

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«ПОЛОЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**ЭЛЕКТРОННЫЙ СБОРНИК
ТРУДОВ МОЛОДЫХ СПЕЦИАЛИСТОВ
ПОЛОЦКОГО ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА**

Выпуск 35 (105)

ПРОМЫШЛЕННОСТЬ

Информационные технологии

Машиностроение

Химические технологии

Обновляется 1 раз в год

Новополоцк
ПГУ
2020

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

Главный редактор:
проф., д.т.н. Д.Н. Лазовский

Заместитель главного редактора:
доц., к.т.н. Ю.П. Голубев

Председатель ред. совета проф., д.т.н. В.П. Иванов
Зам председателя ред. совета проф., д.т.н. Н.Н. Попок

Информационные технологии:

доц., к.т.н. В.Ф. Янушкевич
доц., к.т.н. Р.П. Богуш
доц., к.ф.-м.н. С.А. Вабищевич
доц., к.ф.-м.н. А.А. Козлов

Машиностроение:

доц., к.т.н. А.В. Дудан
доц., к.т.н. Т.В. Вигерина
доц., к.т.н. А.М. Долгих
доц., к.т.н. В.Э. Завистовский

Химическая техника и технология:

доц., к.т.н. А.В. Митинов
доц., х.т.н. С.Ф. Якубовский
доц., к.пед.н. И.В. Бурая
доц., к.т.н. А.А. Ермак

ЭЛЕКТРОННЫЙ СБОРНИК ТРУДОВ МОЛОДЫХ СПЕЦИАЛИСТОВ ПОЛОЦКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА [Электронный ресурс]. – Новополоцк : ПГУ, 2020. – Выпуск 35 (105). Промышленность. – 1 электрон. опт. диск.

Издается с 2014 года (в печатном виде – с 2004 г.)

Сборник включен в Государственный регистр информационного ресурса. Регистрационное свидетельство № 3201404563 от 30.11.2014 г.

В сборнике представлены труды молодых специалистов Полоцкого государственного университета по направлениям прикладных исследований, проводимых в университете: информационные технологии, машиностроение и химические технологии.

Предназначен для работников науки, образования и производства, будет полезен студентам, магистрантам и аспирантам университетов.

Учредитель: учреждение образования «Полоцкий государственный университет»
211440, ул. Блохина, 29, г. Новополоцк, Беларусь
тел. 8 (0214) 59 95 44, e-mail: vestnik@psu.by

Компьютерный дизайн *М. А. Мухоморовой*
Техническое редактирование и компьютерная верстка *О. Ю. Тарасевич*

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.05

ИСПОЛЬЗОВАНИЕ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ДАННЫХ ПРИ РАЗРАБОТКЕ ПРИЛОЖЕНИЯ «АВТОМАТИЗИРОВАННАЯ СИСТЕМА КОММУНИКАЦИЙ СТУДЕНТОВ, АДМИНИСТРАЦИИ И ПОТЕНЦИАЛЬНЫХ РАБОТОДАТЕЛЕЙ»

В.И. БАКЛАН

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)

Рассмотрены особенности фреймворка Spark, а также основные практики и подходы распределённой обработки большого количества данных на примере разработанного приложения «Автоматизированная система коммуникаций студентов, администрации и потенциальных работодателей». Показаны преимущества использования фреймворка Spark в составе интегрированной системы.

Сегодня проблема трудоустройства выпускников после окончания ВУЗов чрезвычайно актуальна, так как молодой специалист сталкивается с довольно жёсткими условиями рынка труда. Проблема трудоустройства выпускников имеет место и для нашего ВУЗа, поэтому создание автоматизированной системы коммуникаций студентов, администрации и потенциальных работодателей, позволяющей установить связь между ними является одной из актуальнейших задач нашего региона.

Для реализации автоматизированной системы коммуникаций студентов, администрации и потенциальных работодателей использовался большой стек современных, пользующихся популярностью среди разработчиков технологий: для реализации backend-части – Scala и Java; для frontend-части – TypeScript; для построения API – Spring Boot Framework на Java, Play Framework и Akka; для построения системы сообщений – Scala, React, Material UI; для хранения данных – СУБД PostgreSQL; для хранения кеша и в качестве брокера сообщений – Redis.

Система развёрнута на платформе Amazon Web Services. База данных размещается в RDS, доступ к Redis осуществляется через ElasticCache, развёртывание кластера производится при помощи ECS. Вся инфраструктура, включая конфигурации сетей, настройки и развёртывания сервисом написана кодом в файлах конфигураций и доступна к развёртыванию с использованием Cloud Formation. Для упаковки приложений в контейнеры для их последующей контейнеризации используется Docker и Docker-Compose.

Исходя из того, что разработанное приложение потенциально может генерировать огромные объёмы данных и мета-информацию, на основании которых можно отслеживать поведение пользователей, сервиса, расставлять приоритеты в выборе последующих подпроектов и анализировать тренды, которые развиваются в рамках сервиса, то такая система не имеет физических ограничений по нагрузке, данные могут генерироваться в любом объёме, следует более подробно остановиться на рассмотрении использования при разработке приложения распределённой обработки данных.

Термин Big Data появился сравнительно недавно. Google Trends показывает начало активного роста употребления словосочетания начиная с 2011 года. Большие данные (англ. big data) – серия подходов, инструментов и методов обработки структурированных и неструктурированных данных огромных объёмов и значительного многообразия для получения воспринимаемых человеком результатов, эффективных в условиях непрерывного прироста, распределения по многочисленным узлам вычислительной сети, сформировавшихся в конце 2000-х годов, альтернативных традиционным системам управления базами данных и решениям класса Business Intelligence. Таким образом, Big Data – это не какой-то конкретный объём данных и не сами данные, а методы их обработки, которые позволяют распределено их обрабатывать. Эти методы применимы не только к огромным массивам данных, но и к маленьким данным.

Существует три основных принципа обработки больших данных:

- горизонтальная масштабируемость;
- отказоустойчивость;
- локальность данных.

В качестве примера обработки большого количества данных можно рассмотреть ClickStream. Clickstream – это поток кликов пользователей, сгруппированный по идентификатору сессии. Так как на начальных этапах объём данных не огромный, обработка будет производиться на кластере с одной нодой и все кликстрим-события будут передаваться в аналитическую систему с помощью Apache Kafka.

Apache Kafka – диспетчер сообщений на Java платформе. В Kafka – есть тема сообщения, в которую издатели пишут сообщения и есть подписчики в темах, которые читают эти сообщения, все сообщения в процессе диспетчеризации пишутся на диск и не зависят от потребителей.

Непосредственным обработчиком будет являться приложение, написанное на языке Python с использованием фреймворка Spark.

Spark – это проект Apache, который позиционируется как инструмент для «молниеносных кластерных вычислений». Проект разрабатывается процветающим свободным сообществом, в настоящий момент является наиболее активным из проектов Apache.

Spark предоставляет быструю и универсальную платформу для обработки данных. По сравнению с Hadoop Spark ускоряет работу программ в памяти более чем в 100 раз, а на диске – более чем в 10 раз.

Spark был выбран в связи с растущим сообществом разработчиков, поддержкой множества систем и языков программирования, большинством встроенных элементов.

Задача аналитического сервиса состоит в генерации поведенческой «карты» пользователя, что в будущем позволит аналитикам и проект-менеджерам продумывать стратегию развития продукта на основании поведенческих данных пользователя в человеко-читаемом формате.

Для уменьшения нагрузки на систему, планируется использование Spark Streaming с Batch-режимом, который позволит обрабатывать поточные данные в небольших батчах, что позволит сократить единоразовую нагрузку и иметь возможность дозировать нагрузку на кластер.

Так как приложение изначально планируется использовать ограниченному количеству пользователей, то для аналитического кластера может использоваться минимальная конфигурация с одной master-нодой и одной worker-нодой, а также минимальным уровнем репликации данных в Kafka. В представленном примере аналитической обработки допускается потеря данных, так как клик является не чувствительной информацией, и потеря допустимой части данных не является критичной.

Таким образом, использование представленного стека технологий позволяет написать систему, которая позволит выполнять обработку любого количества данных, в зависимости от требований и финансовых ограничений. Для последующего увеличения количества данных или времени обработки достаточно расширить кластер, а не переписывать существующий код. Использование языка Python позволяет писать код, не задумываясь о тонкостях языка, что позволяет ускорить процесс разработки и последующей доработки.

ЛИТЕРАТУРА

1. Architectural Styles and the Design. Диссертация Roy Thomas Fielding. [Электронный ресурс]. / ics.uci.edu. – Режим доступа: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf/. – Дата доступа: 20.09.2020.
2. Zaharia M. Spark: The definitive guide // Zaharia M, Chambers B. – O'Reilly Media, Inc, 2018. – 606 с.

УДК 004.05

**ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ CI/CD ПРИ РАЗРАБОТКЕ ПРИЛОЖЕНИЯ
«АВТОМАТИЗИРОВАННАЯ СИСТЕМА КОММУНИКАЦИЙ СТУДЕНТОВ,
АДМИНИСТРАЦИИ И ПОТЕНЦИАЛЬНЫХ РАБОТОДАТЕЛЕЙ»****В. И. БАКЛАН***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)*

Рассмотрены особенности использования CI/CD, а также основные практики и подходы CI/CD непрерывной интеграции и доставки на примере разработанного приложения «Автоматизированная система коммуникаций студентов, администрации и потенциальных работодателей». Показаны преимущества применения данной технологии в процессе разработки программного продукта.

Современная модель получения высшего образования в нашей стране построена на базе коллективного обучения. Это обосновано тем, что студентов гораздо больше, чем преподавателей. В такой системе есть свои плюсы, но основным минусом является то, что она не предоставляет возможности студенту сфокусироваться на конкретном направлении, а взамен – ВУЗ выпускает специалиста широкого профиля, охватывая как можно больший спектр вакансий. Однако получив такого специалиста, работодатель должен затратить дополнительное время и средства на его развитие в зависимости от позиции. Улучшить имеющуюся модель образования можно включив в неё элемент индивидуального обучения (ИО) автоматизировать его. Цель автоматизации – упрощение взаимодействия преподавателя с множеством студентов, при сохранении элемента получения индивидуального образования, что, несомненно, является актуальным.

Таким образом, возникает необходимость создания автоматизированной системы (АИ) коммуникации студентов, администрации и потенциальных работодателей. Разрабатываемая система должна взять на себя решение части задач студента и преподавателя, а в последствии и потенциального работодателя. Причём, со стороны студента система должна: позволять указывать студенту приоритетные направления; мониторить прогресс; фиксировать достижения в той или иной области в результате чего будет формироваться карта обучения, в результате прохождения которой выпускник сможет найти позицию, подходящую приобретёнными знаниями и навыками. Со стороны преподавателя система должна: предоставлять возможность анализа прогресса каждого студента; создавать индивидуальные задачи и планы, на базе заметок; мониторить прогресс одного или группы обучающихся, что позволит реализовать частичное ИО при допустимых затратах сил и времени. Работодателю же, система должна предоставить возможность мониторинга интересующих студентов, просматривая карты обучения конкретного человека, а также возможность корректировки плана обучения на поздних этапах учебного процесса, с целью чтобы после выпуска из учебного заведения специалист был сразу готов работать на определенной позиции.

Непрерывная интеграция (CI, англ. Continuous Integration) – практика разработки программного обеспечения (ПО), которая заключается в постоянном слиянии рабочих копий в общую ветвь разработки и выполнения частых автоматизированных сборок проекта для скорейшего выявления потенциальных дефектов и решения интеграционных проблем.

В обычном проекте, где над разными частями системы разработчики трудятся независимо, стадия интеграции является заключительной. Она может непредсказуемо задержать окончание работ. Переход к непрерывной интеграции позволяет снизить трудоёмкость интеграции и сделать её более предсказуемой за счёт раннего обнаружения и устранения ошибок и противоречий. Основным преимуществом сокращения стоимости исправления дефекта, за счёт раннего её выявления.

Далее подробнее остановимся на рассмотрении основных этапов CI процесса в приложении «Автоматизированная система коммуникации студентов, администрации и потенциальных работодателей»

- 1 Checkout в требуемую ветку (по умолчанию – master);
- 2 Сборка приложения;
- 3 Запуск unit-тестов;
- 4 Запуск integration-тестов;
- 5 Создание docker образа;
- 6 Выгрузка образа в хранилище.

CI процесс построен на базе GitHub для системы хранения и менеджмента кода, Github actions для автоматизации Integration стадии. Для контейнеризации приложения используется Docker Hub, в котором хранятся версионированные образы приложения.

Docker – это платформа, которая предназначена для разработки, развёртывания и запуска приложений в контейнерах. Слово «Docker» в последнее время стало чем-то вроде синонима слова «контейнеризация».

При сборке образа для master ветки создаётся два тега: latest и версионированный тег для того, чтобы развёртывание было проще, так как для данного приложения не критична возможность мгновенного отката к предыдущей версии приложения.

При сборке других веток создаётся только один тег со случайным хеш-кодом, чтобы была возможность развёртывания промежуточных веток в процессе разработки и/или тестирования.

Непрерывная доставка (CD, англ. Continuous delivery) – это практика автоматизации всего процесса релиза ПО. Цель заключается в том, чтобы выполнять CI, плюс автоматически готовить и вести релиз к версии для конечного пользователя. При этом желательно добиться следующего: любой, кто обладает достаточными привилегиями для развёртывания нового релиза может выполнить развёртывание в любой момент, и это можно сделать в несколько кликов. Программист, избавившись практически от всей ручной работы, трудится продуктивнее [1].

Как правило, в процессе непрерывной доставки требуется выполнять вручную как минимум один этап: одобрить развёртывание в окружении конечного пользователя и запустить его. В сложных системах с множеством зависимостей конвейер непрерывной доставки может включать дополнительные этапы, выполняемые вручную либо автоматически.

В рассматриваемом приложении CD реализован на базе AWS Code Deploy, который позволяет разворачивать приложение полностью из заранее собранных контейнеров в несколько кликов. Ограничение привилегий достигается за счёт использования индивидуальных AWS IAM пользователей.

Само приложение развёртывается на базе AWS Web Services, в частности, AWS EC2 (elastic cloud containers) в качестве виртуального сервера, EBS в качестве физически доступной памяти, S3 для хранения ресурсов, RDS для менеджмента базы данных.

Отдельное внимание стоит уделить EC2, так как данный сервис непосредственно предоставляет сервера, на которых разворачиваются контейнеры. EC2 – это облачный сервис, предоставляющий виртуальные сервера (Amazon EC2 Instance), 2 вида хранилищ данных, а также балансировщик нагрузки (Load Balancer). EC2 позволяет запускать уже заранее сконфигурированные серверы с предустановленными ОС. Так же возможно создавать свои образы (AMI – Amazon Machine Image) и использовать любой Linux. Есть возможность настроить защиту доступа к серверам. EC2 инстансы объединяются в группы безопасности (Security Groups) с возможностью ограничения доступа по портам с IP или подсетей. Балансировка нагрузки и автомасштабирование являются очень важными функциями EC2. Можно создать правила, при которых станет возможно автоматически увеличить количество серверов, например, если один или несколько серверов не справляются с нагрузкой. Контроль за здоровьем серверов ведёт ещё один сервис AWS – Amazon Cloudwatch. С помощью данного сервиса можно создавать разного рода проверки – checks, при помощи которых контролируются важнейшие показатели работы ОС. При использовании EC2 осуществляется почасовая оплата, некоторые подсервисы, такие как EBS имеют биллинг с оплатой по месяцам. Для каждого подсервиса предусматривается свой отдельный биллинг по заведомо утверждённой цене в час или в месяц. Так же у EC2 инстансов существует так называемая резервация (Reservation) – оплачивается сразу 3-4 месяца работы сервера, после чего, час работы сервера стоит в ~1,5 раза дешевле. Резервации удобно использовать, если EC2 используется на постоянной основе – экономия на лицо.

Таким образом, использование связки CI/CD и AWS Web Services в данном приложении ускорило разработку и тестирование. Такой подход предоставляет возможность развёртывания новых версий в кратчайшее время, кроме этого, за счёт гибкого использования ресурсов данная связка экономит денежные затраты на каждом шаге интеграции и разработки. Использование сервисов, таких как Github Actions, Cloud Formation предоставляют возможность сохранять конфигурации CI/CD и, в случае заморозки проекта, позволяют в кратчайшие сроки развернуть аналогичную конфигурацию спустя какое-то время без надобности реализации дополнительной конфигурации.

ЛИТЕРАТУРА

1. Architectural Styles and the Design. Диссертация Roy Thomas Fielding. [Электронный ресурс]. / ics.uci.edu. – Режим доступа: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf/. – Дата доступа: 20.09.2020.
2. Философия DevOps. Искусство управления IT // J.Davis – O'Reilly Media, Inc., 2017. – 213 с.
3. Zaharia M. Spark: The definitive guide // Zaharia M, Chambers B. – O'Reilly Media, Inc, 2018. – 606 с.

УДК 004.5; 004.514;

ПОРТФОЛИО КАК ТЕХНОЛОГИЯ ОЦЕНИВАНИЯ ПРОФЕССИОНАЛЬНЫХ КОМПЕТЕНЦИЙ СТУДЕНТОВ

Г.Е. БЕЛЬСКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)

В данной статье рассматривается роль составления личного портфолио и его влияние на повышение профессиональных компетенций выпускников ВУЗа при будущем устройстве на работу.

На сегодняшний день, в ряде стран отмечается снижение качества образования на первой ступени в связи с введением сокращённых сроков обучения. Многие выпускники не соответствуют в полной мере требованиям, которые предъявляются работодателями: вузы выпускают специалистов с общими представлениями о профессиональной деятельности, в большинстве случаев не готовых к оперативному включению в данную область деятельности [1].

В Республике Беларусь массовый переход на дифференцированные сроки подготовки специалистов с высшим образованием осуществлён с 1 сентября 2008 г., однако реализован он одновременно с оптимизацией учебно-программной документации, регламентирующей образовательный процесс, и введением образовательных стандартов нового поколения, реализующих компетентный подход в подготовке специалистов, что позволило избежать снижения качества подготовки специалистов [2].

Сегодня в условиях реализации новых образовательных стандартов вуз должен обеспечить не только процесс развития компетенций специалиста, но и отслеживание и предъявление результатов и достижений его деятельности будущему работодателю. Одной из инновационных технологий оценивания качества обучения и уровня сформированности компетенций студентов в современном вузе является портфолио — собрание личных достижений студента, которое отражает реальный уровень его подготовки и активности в различных учебных и внеучебных видах деятельности в вузе и за его пределами [3].

«Портфолио» в широком смысле слова – это способ фиксирования, накопления и оценки индивидуальных достижений студента в определенный период его обучения. Портфолио активно применяется в зарубежных системах образования, относящих его к разряду «аутентичных» индивидуализированных оценок, ориентированных на новые формы оценивания. Оно позволяет учитывать результаты, достигнутые студентом в разнообразных видах деятельности: учебной, творческой, социальной, коммуникативной и другой, а также является важным элементом практико-ориентированного подхода к образованию. В зарубежной традиции портфолио определяется как «коллекция работ и результатов учащегося, которая демонстрирует его усилия, прогресс и достижения в различных областях» [4].

Портфолио помогает оценить не только когнитивную составляющую компетенций студентов, но и реализовать лично-центрированный подход в профессиональной подготовке будущих специалистов; в Германии приоритетное внимание уделяется накопительному характеру портфолио, который отражается и в широко используемом названии «файлы достижений».

Опираясь на вышеизложенное, был собран ряд критериев, которые в полной мере отражают все компетентные качества студента, которые необходимо отразить в портфолио. Критерии такого портфолио вы можете видеть на рисунке 1.

Краткая информация студента	Его самопредставление Выбранный род деятельности
Специализация Основные навыки которыми владеет студент	Ссылки на сторонние ресурсы студента
Колонка личного прогресса студента (навыки которыми он хотел бы овладеть)	История пройденных курсов, посещенных семинаров или вебинаров

Рисунок 1. – Критерии отображения электронного портфолио студента

Современные исследователи отмечают, что использование портфолио в образовательном процессе ВУЗа позволяет усилить практическую ориентацию и инструментальную направленность образования; достичь оптимального сочетания фундаментальных и практических знаний, направленности образовательного процесса не только на усвоение знаний, но и на развитие способностей мышления, выработку практических навыков; расширить применение интерактивных форм обучения; развить навыки самостоятельной работы; усилить дифференциацию и индивидуализацию образовательного процесса; выявить одарённых студентов и организовать индивидуальный процесс их включению в научно-исследовательскую деятельность; развить у студентов навык саморефлексии.

В качестве наиболее значимых функций портфолио как технологии оценивания профессиональных компетенций студента, отечественные исследователи наряду с контрольно-оценочной функцией выделяют учётно-информационную и контрольно-корректирующую функции, которые позволяют организовать длительное накопление, целенаправленный отбор и отражение специальным образом отбираемых материалов, демонстрирующих уровень сформированности профессиональных компетенций студентов, а также провести коррекцию его профессионально-образовательной деятельности [5]. Кроме того, работа над портфолио позволяет в полной мере выявить склонности и способности студента, помочь осуществить правильный выбор в сфере дальнейшей профессионализации, в частности, между практической и научно-исследовательской деятельностью в рамках избранной специальности. Более детально основные функции портфолио представлены на рисунке 2.



Рисунок 2. – Функции портфолио

В зависимости от целей создания выделяют несколько видов портфолио. Традиционно в образовательной практике используются три типа портфолио:

- портфолио документов,
- портфолио работ
- портфолио отзывов.

Веб портфолио позволяет нам соединить все три типа, показывая студента со всех сторон что обеспечит эффективное взаимодействие студентов с научными руководителями, преподавателями и кураторами в ВУЗе в период обучения, а также с потенциальными работодателями до и после окончания высшего учебного заведения.

К преимуществам электронного портфолио могут быть отнесены высокая степень мобильности его дополнения и коррекции, широкие возможности для художественного оформления, многовариантность выбора средств оформления и структурирования, доступность для внешнего и дистанционного оценивания.

Усилия студентов в данном случае направлены на выявление так называемых точек опоры уже имеющихся знаний, умений, навыков, которые позволяют добиваться успехов в выполняемой деятельности, и точек роста – дефицитов и трудностей, которые тормозят достижение запланированных результатов. Рефлексивное осмысление точек опоры и точек роста позволяет зафиксировать достижения в деятельности относительно ожидаемых результатов, а также значимые изменения, происходящие в самом субъекте деятельности, в частности, в формировании его компетенций.

Пробы в презентации портфолио включают рефлексии и самооценку деятельности на заключительном этапе работы над портфолио и могут быть осуществлены в рамках семинарских занятий, заседаний студенческих научных обществ, тренингов и консультаций по формированию необходимых знаний и навыков студентов по созданию и презентации портфолио. Собственно, презентация портфолио включает представление материалов портфолио-продукта лицам, заинтересованным в информации о достижениях его владельца, планах по его дальнейшему саморазвитию. При этом исследователи отмечают, что презентация

является итоговым, но не финальным действием в рамках портфолио-технологии, так как кроме готового портфолио-продукта в виде папки с материалами могут быть получены и другие результаты работы, не менее значимые для его владельца и других участников.

Таким образом, можно отметить, что внедрение описанных в статье систем в образовательную систему способствует росту конкурентоспособности и собственному развитию студентов как будущих высококвалифицированных кадров.

ЛИТЕРАТУРА

1. Проблемы и перспективы развития образования. [материалы IV Междунар. науч. конф. (г. Пермь, июль 2013 г.).] [Текст] – Дата доступа: 20.09.2019.
2. Новые возможности получения образования за рубежом. [Электронный ресурс] – Режим доступа: <https://www.belta.by/onlineconference/view/bolonskij-protsess.-novye-vozmozhnosti-poluchenija-obrazovaniya-za-rubezhom-540/> – Дата доступа: 23.09.2019.
3. Бельский, Г.Е. Роль портфолио в повышении конкурентоспособности выпускника вуза/ Г.Е. Бельский // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30(100): «Промышленность». – С. 55-57.
4. Григоренко Е. В. Портфолио в вузе: методические рекомендации по созданию и использованию [Электронный ресурс]. – Режим доступа: <https://pandia.ru/text/80/275/40013.php>. – Дата доступа: 15.09.2019.
5. Девисилов, В. А. Портфолио и метод проектов как педагогическая технология мотивации и личностно ориентированного обучения [Электронный ресурс]. – Режим доступа: http://www.mhsts.ru/science/Devisilov/Technologii_motivacii.pdf. – Дата доступа: 10.09.2019.
6. Болонский процесс. [Электронный ресурс] – Режим доступа: <https://www.psu.by/obrazovanie/bolonskij-protsess> – Дата доступа: 23.09.2019.

**ТЕХНОЛОГИЯ «ПОРТФОЛИО»
В КОМПЕТЕНТНОСТНО-ОРИЕНТИРОВАННОМ ОБРАЗОВАНИИ****Б.Г. БЕЛЬСКИЙ***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)*

В данной статье рассматривается технология системы составления портфолио, внедрённой в учебный процесс учащихся вузов.

Традиционные формы и методы подготовки студентов не способствуют оптимизации процесса формирования у таковых компонентов профессиональной культуры [1].

В самом деле, ведущей формой организации подготовки является предметная система, в рамках которой студент решает, как правило, задачи той или иной учебной дисциплины, обычно не связывая их с ведущей целью высшего образования – формированием профессионализма [2].

Самореализация в области профессиональной деятельности требует специфической личностной работы, и ее выполнению, конечно же, необходимо учить. Отсюда следует, что задача проектирования образовательной системы, предоставляющей студенту реальную возможность составить представление о ключевых компетенциях профессионала, должна рассматриваться педагогикой как приоритетная.

Компетентностный подход к организации вузов учебного процесса, разрабатываемый сегодня многими исследователями, открывает принципиально новые педагогические возможности в решении задачи повышения качества подготовки будущих специалистов. Существенным в этом подходе является отказ от ориентации главным образом на предметную подготовку будущего специалиста. Одной из наиболее перспективных педагогических технологий, реализующих основные идеи компетентностного подхода, является внедрение системы составления портфолио среди учащихся высших учебных заведений [3].

Составление портфолио студентов в такой системе максимально разделяет преподавание, учение и оценивание органически интегрируя эти три составляющие в процесс обучения, что позволяет объединить количественную и качественную оценку способностей учащегося посредством анализа разнообразных продуктов учебно-познавательной деятельности. В этой системе поощряется не только оценка, но и самооценка, и взаимооценка учащихся, а также самоанализ и самоконтроль учащегося. А само портфолио направлено на сотрудничество преподавателя и студента с целью оценки достижений, приложенных усилий и прогресса в обучении. И само по себе портфолио – форма непрерывной оценки в процессе непрерывного образования, которая смещает акценты от жестких факторов традиционной оценки к гибким условиям оценки альтернативной [2].

Для университетов республики Беларусь эта тема является как никогда актуальной, именно потому главной задачей стоит разработка веб сервиса, который помогал бы студентам в составлении их будущего портфолио [1]. Прототип разработанного веб сервиса вы можете наблюдать на рисунке 1.

В настоящее время кардинальные изменения вносятся на рынок труда существенные коррективы по отношению к человеку к его профессионально квалификационным характеристикам и выдвигают все новые и более жесткие требования к продукту его профессионального труда. В связи с этим основными морально-психологическими качествами работника становятся инициативность, самостоятельность, креативность, способность сотрудничать и взаимодействовать с другими субъектами общества, высокая мотивация к карьерному росту, повышению квалификации и профессиональному переобучению. Именно эти качества позволяют человеку быть мобильным и конкурентоспособным на рынке труда, а технология «Портфолио» как эффективный инструмент и ведущий показатель системы оценки качества образования позволяет наглядно представить целостную картину о степени сформированности личности как успешного, компетентного и делового человека.

Основная цель формирования «портфолио» – накопить и сохранить документальное подтверждение собственных достижений студента в процессе его обучения в вузе. Портфолио является не только современной эффективной формой самооценивания результатов образовательной деятельности студента, но и способствует:

- мотивации к образовательным достижениям;
- приобретению опыта к деловой конкуренции;
- обоснованной реализации самообразования для развития профессиональных компетентностей;
- выработке умения объективно оценивать уровень своих профессиональных компетентностей;
- повышению конкурентоспособности будущего специалиста.



Рисунок 1. – Макет разработанного электронного портфолио студента

В свою очередь, основные ожидаемые эффекты внедрения новой системы составления портфолио учащихся вузов в высшие образовательные учреждения окажет положительное влияние на организацию и эффективность учебного процесса, мотивируя и поощряя активность студентов, связанную с приобретением профессиональных знаний и умений, их участие в научной работах, в конкурсах, студенческих научных конференциях [1].

Приобретая большую самостоятельность в обучении, студенты нуждаются в консультации, помощи преподавателя больше, чем в его непосредственном руководстве, управлении. В процессе аудиторной и внеаудиторной самостоятельной деятельности студенты определяют и формулируют цели и задачи самостоятельной работы, разрабатывают алгоритмы действий, необходимые для реализации, анализируют полученные результаты, обобщают и систематизируют их, делая выводы [1].

Кроме того, в процессе выполнения учебно-познавательных или учебно-профессиональных заданий студент вырабатывает собственный стиль общения с коллегами, преподавателем. Таким образом, студент демонстрирует степень сформированности собственной компетентности, т.е. готовности и способности к профессиональной деятельности [1].

Портфолио, являясь формой полного и разностороннего представления о выпускнике, определяет образовательный рейтинг будущего специалиста и может использоваться учебным заведением в качестве дополнительной информации о студенте при подведении итогов учебной, научно-исследовательской, социально-творческой деятельности, осуществлении трудоустройства, выдаче рекомендаций при поступлении в магистратуру, аспирантуру и т.д.

Таким образом, комплексное использование указанных средств оценки образовательных результатов в процессе формирования портфолио студента – является эффективной инновационной технологией, позволяющей наилучшим образом объединить в себе потребности образовательного процесса и научной деятельности в вузе, сформировать тягу к творческой и научной самореализации и всестороннему развитию личности обучающихся в процессе обучения.

ЛИТЕРАТУРА

1. Бельский, Г.Е. Роль портфолио в повышении конкурентоспособности выпускника вуза/ Г.Е. Бельский // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30(100): «Промышленность». – С. 55-57.
2. Говюнов, В.С. Информационные системы в образовании. [Журнал «Молодой ученый» №16 (5) – май 2010 г.] [Текст] – Дата доступа: 20.09.2019.
3. Буриев, К.С. Роль портфолио в современном образовании [Образование и воспитание» №4 (9), октябрь 2016 г.] [Текст] – Дата доступа: 20.09.2019.

ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МАГАЗИНА АВТОЗАПЧАСТЕЙ

К.С. БОРИСЁНОК

(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИН)

В статье представлен практический способ создания интерфейса для сайта автозапчастей. Были поставлены основные цели сайта, задачи веб-сервиса и основы построения интерактивного прототипа интерфейса.

Введение. Интернет-магазин (англ. online shop или e-shop) — сайт, торгующий товарами в интернете. Позволяет пользователям сформировать заказ на покупку, выбрать способ оплаты и доставки заказа в сети Интернет.

Выбрав необходимые товары или услуги, пользователь обычно имеет возможность тут же на сайте выбрать метод оплаты и доставки. Совокупность отобранных товаров, способ оплаты и доставки представляют собой законченный заказ, который оформляется на сайте путем сообщения минимально необходимой информации о покупателе. Информация о покупателе может храниться в базе данных магазина если бизнес-модель магазина рассчитана на повторные покупки, или же отправляться разово. В интернет-магазинах, рассчитанных на повторные покупки, также ведется отслеживание возвратов посетителя и история покупок. Часто при оформлении заказа предусматривается возможность сообщить некоторые дополнительные пожелания от покупателя продавцу.

Сайт должен представлять магазин автозапчастей в интернете, поддерживать его положительный и современный имидж, знакомить посетителей с товарами, предоставлять пользователю удобный и понятный интерфейс для облегчения процесса ознакомления с услугами и дальнейшее взаимодействие с сайтом.

Цель сайта – познакомить посетителей с магазином, направлениями его деятельности, товарами, ценами на них. Также к основным целям относятся:

- привлечение количество покупателей;
- экономия времени клиентов и администрации;
- наличие актуальной информации по товарам.

Проектируемый сайт должен удовлетворять следующим требованиям:

- наличие простого и удобного пользовательского интерфейса;
- наличие обширной базы товаров;
- наличие корзины;
- наличие широких функциональных возможностей, а именно: заказ товаров, создания и редактирование новых категорий, добавление и редактирование товаров в созданные категории, оплата товаров через платежные системы и онлайн консультация.

Построение интерактивного прототипа интерфейса.

Прототип – это визуальный каркас сайта, который дает представление о том, как будет выглядеть веб-ресурс [1]. Он отображает структуру будущего сайта, а также его отдельных страниц вне дизайнерских элементов, цветов, изображений и брендинга.

В данном интернет-магазине будет реализован интерфейс, который будет состоять из следующих элементов:

- header это блок в верхней части страницы сайта, который виден на всех страницах сайта. Как правило, содержит логотип, меню, контакты, переключатель языков или корзину, если в интернет-магазине несколько страниц и нужна общая корзина.

- content это все информационное содержание ресурса: тексты, изображения, видео и пр.

- footer это блок в нижней части страницы сайта. Используется в основном как блок дополнительной информации, так же в нем часто дублируют информацию из шапки сайта.

Данный интернет-магазин будет состоять из 5 блоков:

- в первом блоке расположена шапка сайта, где расположены навигационные кнопки (главная, магазин, о нас, контакты), которые позволяют пользователю, перемещаться по интернет-магазину, также в данном блоке находится логотип и контактная информация;

- во втором блоке находится каталог товаров, где пользователь может выбрать нужную ему категорию и далее сам товар;

- в третьем блоке находится список последних добавленных товаров;

- в четвертом блоке расположен слайдер, где пользователь может, просмотреть список рекомендуемых товаров;
- в пятом блоке находится название организации.

Рассмотрим макет пользовательской части сайта. На рисунке 1 представлен макет пользовательской части сайта.

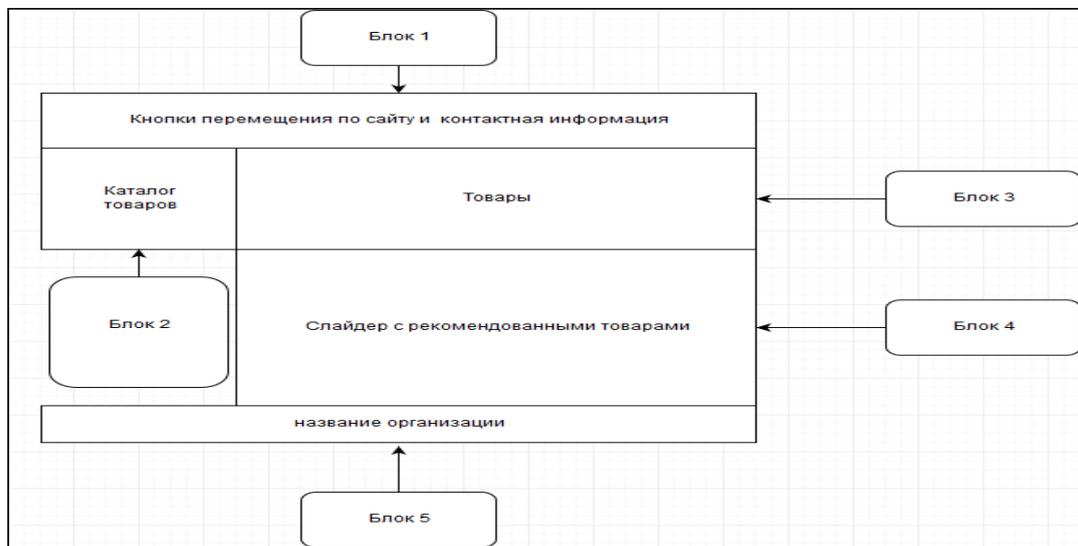


Рисунок 1. – Макет интернет-магазина автозапчастей пользовательской части

Рассмотрим так же макет административной части сайта. На рисунке 2 представлен макет административной части сайта.



Рисунок 2. – Макет интернет-магазина автозапчастей административной части

Данный страница будет состоять из двух блоков:

- в первом блоке расположено сообщение с приветствием администратора и кнопка вернуться обратно на сайт;
- во втором блоке находятся все функции, предоставленные администратору сайта (управление товарами, управление категориями, управление заказами).

Рассмотрим основные функции, реализованные в пользовательском интерфейсе.

Функция «Slider», администратор может добавлять в него рекомендуемые товары реализация данной функции представлена в (листинге 1)

Листинг 1 – Функция «Slider»

```
!!function( $ ) {
var Slider = function(element, options) { this.element = $(element);
this.picker = $('<div
```

```
class="slider">'+ '<div
class="slider-track">'+
'<div class="slider-selec-
tion"></div>'+ '<div
class="slider-handle"></div>'+
'<div class="slider-han-
dle"></div>'+ '</div>'+
'<div class="tooltip"><div class="tooltip-arrow"></div><div
class="tooltip inner"></div></div>'+
'</div>')
.insertBefore(this.element)
.append(this.element);
this.id = this.element.data('slider-id')||op-
tions.id; if (this.id) {
this.picker[0].id = this.id;
}
if (typeof Modernizr !== 'undefined' && Modern-
izr.touch) { this.touchCapable = true;
}
var tooltip = this.element.data('slider-tooltip')||op-
tions.tooltip; this.tooltip = this.picker.find('.tooltip');
this.tooltipInner = this.tooltip.find('div.tooltip-inner');
this.orientation = this.element.data('slider-orientation')||options.orien-
tation; switch(this.orientation) {
case 'vertical':
this.picker.addClass('slider-vertical');
this.stylePos = 'top';
this.mousePos = 'pag-
eY'; this.sizePos = 'off-
setHeight';
this.tooltip.addClass('right')[0].style.left = '100%';
break;
default:
this.picker
.addClass('slider-horizontal')
.css('width', this.element.outer-
Width()); this.orientation = 'hori-
zontal'; this.stylePos = 'left';
this.mousePos =
'pageX'; this.sizePos =
'offsetWidth';
this.tooltip.addClass('top')[0].style.top = -this.tooltip.outerHeight() - 14 +
'px'; break;
}
```

Заключение. В данной статье рассмотрены основные задачи и требования. Были сформулированы необходимые разделы программного средства, для каждого из которых был спроектирован прототип, на основе которого можно создать графическое изображение экрана сайта, а потом и сам веб-сервис.

ЛИТЕРАТУРА

1. Статья о использовании построении сайта [Электронный ресурс] / Прототипирование. Дизайн в цифровой сфере – 2014. – Режим доступа: <http://tilda.education/courses/web-design/prototypes/>. – Дата доступа: 23.08.2020.

УДК 004.051

ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА МАГАЗИНА АВТОЗАПЧАСТЕЙ

К.С. БОРИСЁНОК

(Представлено канд. техн. наук, доц. А.Ф. ОСЬКИН)

В статье представлен практический способ проектирования программного продукта, было описано проектирование диаграммы вариантов использования и проектирование логической структуры базы данных.

Введение. Проектирование программного обеспечения – процесс создания проекта программного обеспечения (ПО), а также дисциплина, изучающая методы проектирования. Проектирование ПО является частным случаем проектирования продуктов и процессов [1].

Основной раздел. Диаграмма вариантов использования является исходным концептуальным представлением или концептуальной моделью системы в процессе её проектирования и разработки. Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества сущностей или актёров, взаимодействующих с системой с помощью так называемых прецедентов. Каждый прецедент определяет некоторый набор действий, совершаемых системой при диалоге с актёром [2].

Для работы приложения определен двумя актёрами, которые выполняет доступные ему действия. Варианты использования описаны в таблице 1.

Таблица 1. – Варианты использования приложения

Варианты использования	Описание
Администратор (актёр)	Главный пользователь ИСИМ, выполняет функции по работе с базой данных;
Авторизация	Вход в систему пользователя с определением его прав доступа
Регистрация товара	Работа с таблицей товары. Расширяется редактированием данных о товаре, изменением и удалением данных
Управление пользовательской системой	Работа с таблицей пользователи. Расширяется добавлением и удалением пользователей, а так же изменением данных о пользователях
Подтверждение заказа	Смена индикатора выполнения заказа в базе данных
Пользователь (актёр)	Пользователь выполняющий основные функции на сайте
Обратная связь	Консультация с администратором по вопросам связанные с сайтом
Выбор товаров	Включает просмотр товаров и поиск по категориям
Добавление товаров	Возможность добавить выбранный товар в корзину для дальнейшей оплаты

На основании выделенных актёров и вариантов использования построена диаграмма вариантов использования, представленная на рисунке 1.

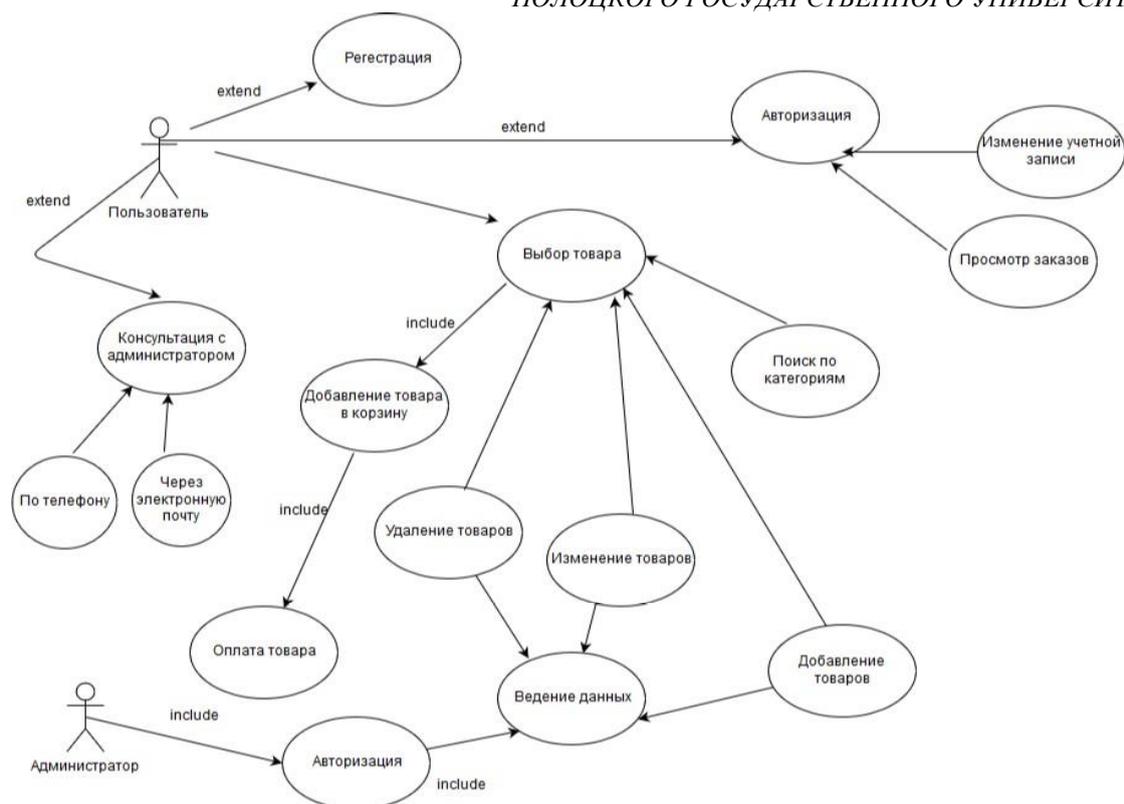


Рисунок 1. – диаграмма вариантов использования

Проектирование логической структуры базы данных

Исходя из предметной области интернет-магазина, можно выделить 3 основные сущности:

- Пользователь [User] – представляет информацию о каждом пользователе.
- Товар [Goods] – представляет информацию о товарах, имеющихся в наличии.
- Заказ [Order] – представляет информацию о всех заказах пользователей.

Для каждой сущности необходимо определить атрибуты. Для сущности Пользователь определены следующие атрибуты:

Сущности, их атрибуты и характеристики атрибутов приведены в таблице 2.

Таблица 2 – Перечень сущностей и их атрибутов

Сущность	Атрибут	Описание атрибута
Пользователь	User ID	Уникальный идентификационный номер
	Логин	Логин пользователя для входа в систему
	Пароль	Пароль пользователя для входа в систему
Товар	Goods ID	Уникальный идентификационный номер
	Имя	Название товара
	Количество	Количество товаров на складе
	Цена	Текущая цена одного товара
Заказ	Order ID	Идентификационный номер заказа
	Order Date	Дата подтверждения заказа пользователем
	Contents	Содержимое заказа
	Exstat	Индикатор выполнения заказа
	Login	Логин пользователя, сделавшего заказ

Заключение. В данном разделе был приведен практический способ проектирования программного продукта, было описано проектирование диаграммы вариантов использования и проектирование логической структуры базы данных.

ЛИТЕРАТУРА

1. Проектирование программного продукта. [Электронный ресурс] – Режим доступа: <https://ru.wikipedia.org/wiki/>. – Дата доступа: 10.08.2020.
2. Диаграмма вариантов использования [Электронный ресурс] // Википедия, свободная энциклопедия. – Электронные данные – Режим доступа: <https://ru.wikipedia.org/wiki/UML>. – Дата доступа 10.08.2020.

УДК 006.9:621.317(075.8)

ПРОЕКТИРОВАНИЕ И ИССЛЕДОВАНИЕ АНТЕННЫ СВЧ ДИАПАЗОНА

Д.Д. БУДЬКО

(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)

В статье рассмотрены вопросы применения приёмно-передающих антенны СВЧ диапазона. Проведено исследование коэффициента стоячей волны по напряжению и входного импеданса. Рассчитаны характеристики антенны. Результаты исследований могут быть использованы для создания антенн, применяемых в различных радиоэлектронных приборах и устройствах, в радиолокации, спутниковом телевидении, радиосвязи, беспроводных компьютерных сетях, спутниковой навигации.

При проектировании антенны в качестве референса использована микрополосковая патч антенна Антрад-4 имеющая излучающую поверхность в форме круга.

Формирование первого эллипса антенны:

$$S_1 = \pi R_1 \cdot r_1, \quad (1)$$

где R_1 – большая полуось;

r_1 – малая полуось.

Формирование второго эллипса антенны:

$$S_2 = \pi R_1 \cdot r_1 - 2 \cdot \left(\frac{\pi r_1^2 + k(z+x) + 2ab \sin a}{2} \right). \quad (2)$$

Из проделанных расчетов, первый эллипс составляет 105,97 мм², второй эллипс составляет 84,49 мм².

В результате получилась антенна, которая показана на рисунке 1.

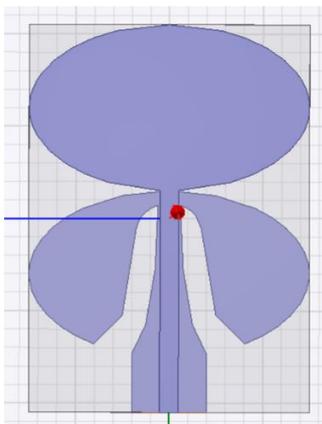


Рисунок 1. – Проектируемая антенна

Результаты исследования необходимо отразить на графиках: коэффициент стоячей волны по напряжению и входной импеданс.

Коэффициент стоячей волны по напряжению вычисляется по формуле:

$$КСВН = \frac{U_{\text{пад}} + U_{\text{отр}}}{U_{\text{пад}} - U_{\text{отр}}}, \quad (3)$$

где $U_{\text{пад}}$ – амплитуда падающей волны;

$U_{\text{отр}}$ – амплитуда отраженной волны.

В идеальном случае $КСВН = 1$, это означает, что отраженная волна отсутствует. Допустимые значения КСВН на рабочей частоте или в полосе частот для различных устройств регламентируются в технических условиях и ГОСТах. Обычно приемлемые значения коэффициента лежат в пределах от 1,1 до 2,0.

Входной импеданс антенны – основная характеристика передающей и приемной антенны, которая определяется как отношение высокочастотного напряжения U_a и тока питания I_a . Входной импеданс антенны Z определяется как сумма сопротивления излучения R_1 и сопротивления потерь антенны R_2 .

$$Z = R_1 + R_2 \quad (4)$$

Сопротивление потерь R_2 , в свою очередь складывается из омических потерь в элементах и проводах антенны, потерь в изоляции (в связи с утечками), сопротивление потерь в земле и тепловые потери в окружающих предметах, лежащих в ближней зоне антенны.

На рисунках 2 и 3 приведены результаты моделирования.

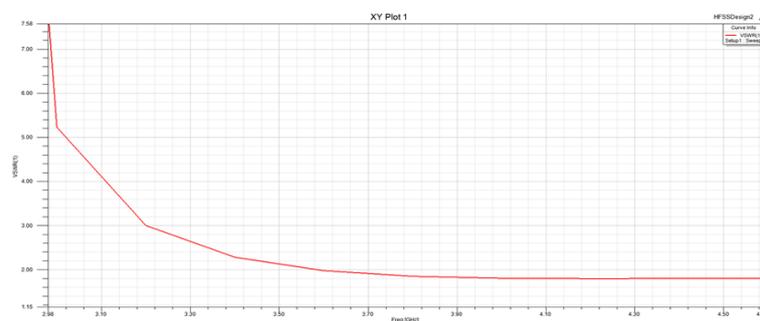


Рисунок 2. – Коэффициент стоячей волны

На диаграмме видно, что величина *КСВН* переходит значение 2. Это указывает на то, что данная разрабатываемая антенна может хорошо работать на высоких частотах [1-3].

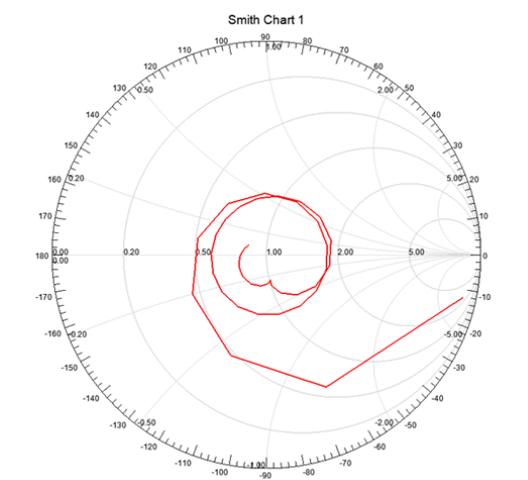


Рисунок 3. – Входной импеданс антенны

Спроектированная антенна может найти применение в различных радиоэлектронных приборах и устройствах, и будет востребована как в военной, так и гражданской промышленности, спутниковом телевидении, радиосвязи, беспроводных компьютерных сетях, спутниковой навигации.

ЛИТЕРАТУРА

1. Филонов А. А. Устройства СВЧ и антенны: учебник / А. А. Филонов, А. А. Фомин, Д. Д. Дмитриев – Красноярск: Сиб. федер. ун-т, 2014. – 492 с.
2. Кременя, К.И. Сравнительный обзор сверхширокополосных антенн: плоской логарифмической спиральной антенны Вивальди/ К.И.Кременя, В.Ф.Янушкевич// Вестник ПГУ. Серия С. Информационные технологии. Электрофизика. - 2014. - №12 С.21-24.
3. Банков С.Е. Решение оптических и СВЧ задач с помощью HFSS – М / Банков С.Е., Гутцайт Э.М., Курушин А.А.; ООО «Оркада», 2012, – 250 с.

УДК 006.9:621.317(075.8)

РАЗРАБОТКА И МОДЕЛИРОВАНИЕ АНТЕННЫ ДЛЯ СВЧ ДИАПАЗОНА

Д.Д.БУДЬКО

(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)

Введение. Для разработки излучающей антенны, использующей СВЧ диапазон нужно определиться с верхней и нижней его границами.

В таблице 1 приведена классификация волн СВЧ диапазона в зависимости от длины волны.

Таблица 1. – Классификация волн СВЧ диапазона

Вид СВЧ-волн	Длина волны	Частота колебаний, Гц
метровые	(1 – 10) м	$3 \cdot 10^7 - 3 \cdot 10^8$
дециметровые	(10 – 100) см	$3 \cdot 10^8 - 3 \cdot 10^9$
сантиметровые	(1 – 10) см	$3 \cdot 10^9 - 3 \cdot 10^{10}$
миллиметровые	(1 – 10) мм	$3 \cdot 10^{10} - 3 \cdot 10^{11}$

Из таблицы 1 следует, что сантиметровые волны генерируются в диапазоне от $3 \cdot 10^9$ до $3 \cdot 10^{10}$ Гц, что в пересчете означает 3 ГГц – 30 ГГц соответственно. Это означает что разрабатываемая антенна должна работать в этом диапазоне [1 – 3].

Определение формы проектируемой антенны

При проектировании антенны в качестве референса использована микрополосковая патч антенна Антрад-4 имеющая излучающую поверхность в форме круга. Разрабатываемая антенна должна иметь необычную форму рисунка, быть универсальной (рассчитанной на приём и передачу) и иметь хорошие данные.

Далее нужно определиться с размерами излучающей поверхности. В ходе испытаний опытным путем были определены размеры эллипсов.

Формирование первого эллипса антенны

$$S_1 = \pi R_1 \cdot r_1, \quad (1)$$

где R_1 – большая полуось;

r_1 – малая полуось.

Формирование второго эллипса антенны

$$S_2 = \pi R_1 \cdot r_1 - 2 \cdot \left(\frac{\pi r^2 + k(z+x) + 2ab \sin \alpha}{2} \right). \quad (2)$$

Из проделанных расчетов, первый эллипс составляет 105,97 мм², второй эллипс составляет 84,49 мм². В результате получилась антенна, которая показана на рисунке 1.

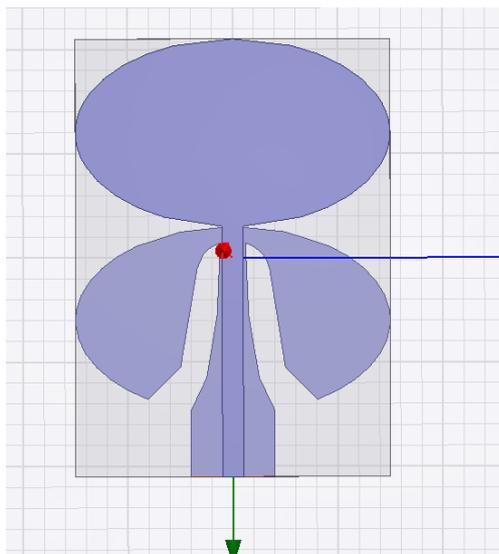


Рисунок 1. – Проектируемая антенны

Результаты моделирования антенны

На данном этапе оптимизации, нужно определить зависимость параметров антенны от изменения толщины излучающей поверхности и выбрать наиболее подходящее значение.

Результаты исследования необходимо отразить на графиках:

- Коэффициент стоячей волны по напряжению;
- Входной импеданс;
- Диаграмма направленности.

Коэффициент стоячей волны по напряжению – отношение наибольшего значения амплитуды напряжения электрического или магнитного поля стоячей волны в линии передачи к наименьшему. Коэффициент стоячей волны по напряжению вычисляется по формуле:

$$КСВН = \frac{U_{\text{пад}} + U_{\text{отр}}}{U_{\text{пад}} - U_{\text{отр}}} \quad (3)$$

где $U_{\text{пад}}$ – амплитуда падающей волны;

$U_{\text{отр}}$ – амплитуда отраженной волны.

В идеальном случае $КСВН = 1$, это означает, что отраженная волна отсутствует. Однако на практике такая величина может быть получена лишь при коротком замыкании на генераторе. Допустимые значения $КСВН$ на рабочей частоте или в полосе частот для различных устройств регламентируются в технических условиях и ГОСТах. Обычно приемлемые значения коэффициента лежат в пределах от 1,1 до 2,0.

Диаграмма направленности антенны – графическое представление зависимости коэффициента усиления антенны или коэффициента направленного действия антенны от направления антенны в заданной плоскости.

Входной импеданс антенны (или входное сопротивление антенны) – основная характеристика передающей и приемной антенны, которая определяется как отношение высокочастотного напряжения U_a и тока питания I_a . Входной импеданс антенны Z определяется как сумма сопротивления излучения R_1 и сопротивления потерь антенны R_2 .

$$Z = R_1 + R_2 \quad (4)$$

Сопротивление потерь R_2 , в свою очередь складывается из омических потерь в элементах и проводах антенны, потерь в изоляции (в связи с утечками), сопротивление потерь в земле и тепловые потери в окружающих предметах, лежащих в ближней зоне антенны.

База данных HFSS включает широкий спектр материалов с разными диэлектрической и магнитной проницаемостью, электрическими и магнитными тангенсами угла потерь. Лучшим материалом для создания антенны является RT/duroid5880 стекловолотно.

На рисунках 2-5 приведены результаты моделирования.

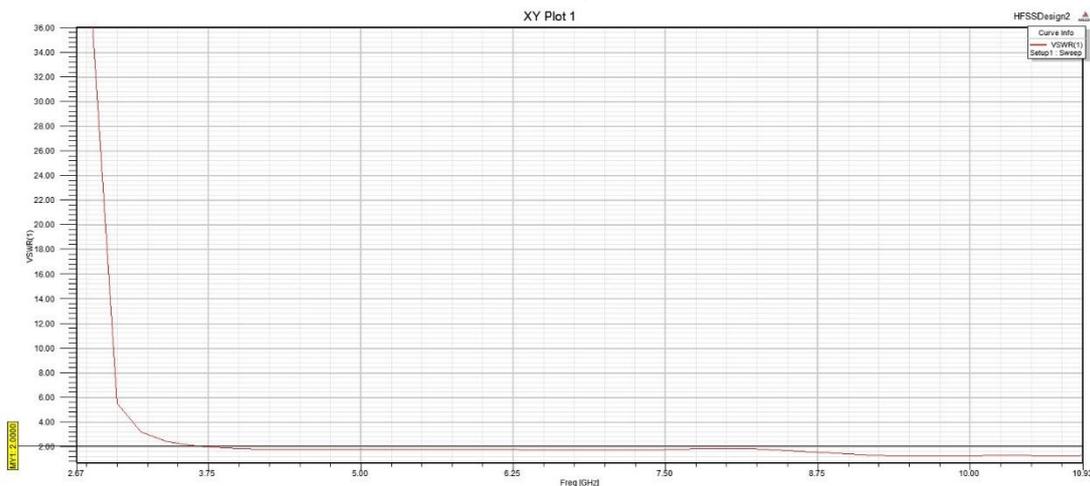


Рисунок 2. – Коэффициент стоячей волны

Значение *KCBH* антенны не превышает 2 при данном материале, что говорит о работоспособности антенны. На участке от 4,52 до 6,54 ГГц *KCBH* равномерен. Минимальное значение *KCBH* равным 1,28 наблюдается на частотах 9,40 и 10,98 ГГц.

Характер излучения антенны в пространстве описывается диаграммой направленности. Кроме излучения в основном (главном) направлении, существуют побочные излучения — задние и боковые лепестки. Диаграмму направленности передающей антенны можно построить, поворачивая ее и измеряя напряженность поля на фиксированном расстоянии и не изменяя частоту передачи. Эти измерения, преобразованные в графическую форму, дают представление в каком направлении антенна имеет максимальный коэффициент усиления, т.е. полярная диаграмма показывает направление, в котором концентрируется энергия, излучаемая антенной в горизонтальной и вертикальной плоскостях. В данной СВЧ антенне коэффициент усиления равен 3,8 дБ.

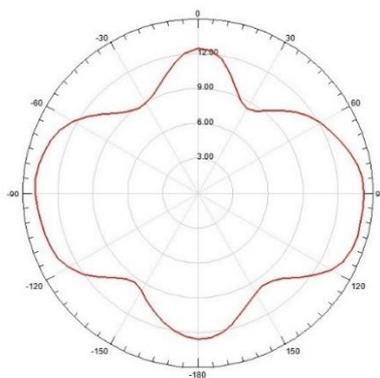


Рисунок 3. – Диаграмма направленности в горизонтальной проекции

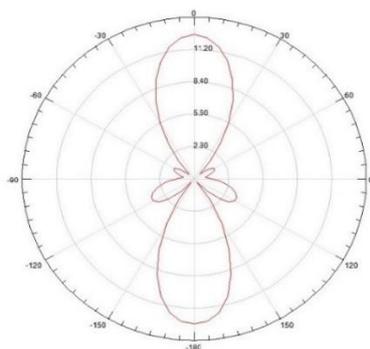


Рисунок 4 – Диаграмма направленности в вертикальной проекции

Коэффициент стоячей волны характеризует степень согласования антенны с фидером, а также согласование выхода передатчика и фидера. На практике всегда часть передаваемой энергии отражается и возвращается в передатчик. Отраженная энергия вызывает перегрев передатчика и может его повредить. В идеале $KCB=1$, значения до 1,5 считаются приемлемым. В разработанной антенне $KCB=1,5$.

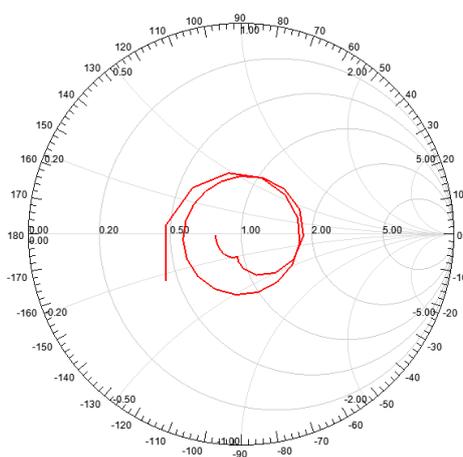


Рисунок 5. – Входной импеданс антенны

Заключение. Импеданс обозначает комплексное (суммарное) сопротивление антенны, и оно изменяется вдоль ее длины. Точка максимального тока и минимального напряжения соответствует наименьшему импедансу и называется точкой возбуждения. Импеданс в этой точке называется входным импедансом. Реактивная составляющая входного импеданса на резонансной частоте теоретически равна нулю. На частотах выше резонансной, импеданс носит индуктивный характер, а на частотах ниже резонансной – емкостной. На практике реактивная составляющая в большинстве случаев меняется от 0 до ± 100 Ом. Сопротивление антенны чаще «подгоняют» под сопротивление линии передачи, чем наоборот [1–3].

ЛИТЕРАТУРА

1. Филонов, А. А. Устройства СВЧ и антенны: учебник / А. А. Филонов, А. А. Фомин, Д. Д. Дмитриев – Красноярск: Сиб. федер. ун-т, 2014. – 492 с.
2. Кременя, К.И. Сравнительный обзор сверхширокополосных антенн: плоской логарифмической спиральной антенны Вивальди/ К.И.Кременя, В.Ф.Янушкевич// Вестник ПГУ. Серия С. Информационные технологии. Электрофизика. – 2014. – №12. – С.21-24.
3. Банков, С.Е. Решение оптических и СВЧ задач с помощью HFSS – М / Банков С.Е., Гутцайт Э.М., Курушин А.А; ООО «Оркада», 2012, – 250 с.

**ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА ВЕБ-ПРИЛОЖЕНИЯ
«ПРОДАЖА АВТОМОБИЛЕЙ «ANGELCAR»****А.Д. ВИНОГРАДОВА***(Представлено: канд. техн. наук, доц. О.Н. ПЕТРОВИЧ)*

В данной статье описано создание графического интерфейса для веб-приложения «Продажа автомобилей «AngelCar». Данный интерфейс включает как простое чтение информации, так и более сложные взаимодействия вроде администраторской части приложения, где можно будет напрямую добавлять и/или редактировать записи в базе данных, обрабатывать заявки клиентов, выгружать отчеты в doc-формате. В статье приведены и обоснованы выбранные средства и методы для разработки интерфейса.

Введение. В последнее время информационные технологии стали неотъемлемой частью нашей жизни. Информационные системы, связанные с предоставлением и обработкой информации для всех уровней управления объектами, приобретают особую важность в общественной жизни. На данный момент невозможно представить какую-либо организацию, не применяющую компьютерных технологий. Это обусловлено и тем, что государственные структуры требуют обязательных отчетов в электронном виде, следовательно, необходима систематизированная информация.

На автомобильном рынке в настоящее время появляется все больше автомобилей различных марок и моделей. Как правило, это новые или поддержанные импортные машины. Дилерам автотехники приходится существовать в условиях довольно жесткой конкуренции с различными производителями: сейчас, когда выбор на рынке очень широк, покупатель может сделать выбор или в пользу нового отечественного автомобиля, или в пользу поддержанного, а может быть, и нового импортного. Продавцу автомобилей необходимо точно знать, почему покупатель выбирает или не выбирает его продукцию, какие требования он к ней предъявляет [1].

В связи с большим количеством марок легковых автомобилей на автомобильном рынке и их технических характеристик, продавцу очень сложно запомнить характеристики каждого автомобиля. Следовательно, использование информационных технологий для автоматизации труда дилера является актуальным и значимым.

Для решения этой задачи будет разработано веб-приложение «Продажа автомобилей «AngelCar», позволяющее автоматизировать процесс подбора легкового автомобиля с учётом требований и возможностей клиента. Разработанное приложение поможет ускорить и упростить работу дилера по продаже автомобилей.

Основная часть. Выбор средств решения поставленной задачи исходил из того, какое программное и аппаратное обеспечение необходимо для разработки.

В качестве среды разработки для реализации проекта была выбрана программа IntelliJ IDEA.

JSP (JavaServer Pages) – технология, позволяющая веб-разработчикам создавать содержимое, которое имеет как статические, так и динамические компоненты. Страница JSP содержит текст двух типов: статические исходные данные, которые могут быть оформлены в одном из текстовых форматов HTML, SVG, WML, или XML, и JSP-элементы, которые конструируют динамическое содержимое. Кроме этого могут использоваться библиотеки JSPтегов, а также EL (Expression Language), для внедрения Java-кода в статичное содержимое JSP-страниц.

Код JSP-страницы транслируется в Java-код сервлета с помощью компилятора JSP-страниц Jasper, и затем компилируется в байт-код виртуальной машины java (JVM). Контейнеры сервлетов, способные исполнять JSP-страницы, написаны на платформонезависимом языке Java. JSP-страницы загружаются на сервере и управляются из структуры специального Java server packet, который называется Java EE Web Application. Обычно страницы упакованы в файловые архивы .war и ear.

JSP является платформонезависимой, переносимой и легко расширяемой технологией для разработки веб-приложений. Потому данная технология прекрасно подходит под концепцию поставленной задачи [2].

Для комфортного восприятия информации пользователями отображение данных из базы оформлено в виде таблиц.

На стартовой странице приложения при гостевом режиме размещены три вкладки: магазин, о нас, контакты. Поддерживается локализация системы – возможность сайта менять язык. Доступны два языка: русский и английский. С данной страницы можно выполнить вход в систему, нажав на значок входа. Представлена карусель с новостями о сайте, а также фотографии поступивших недавно в продажу новых автомобилей. После авторизации пользователя доступны следующие вкладки на стартовой странице: магазин, о нас, контакты, корзина, заказы, оплата. Для администратора – магазин, о нас, контакты, админ.

На рисунках 1 и 2 представлены примеры получившегося графического интерфейса, а точнее страница добавления новых автомобилей и авторизации.

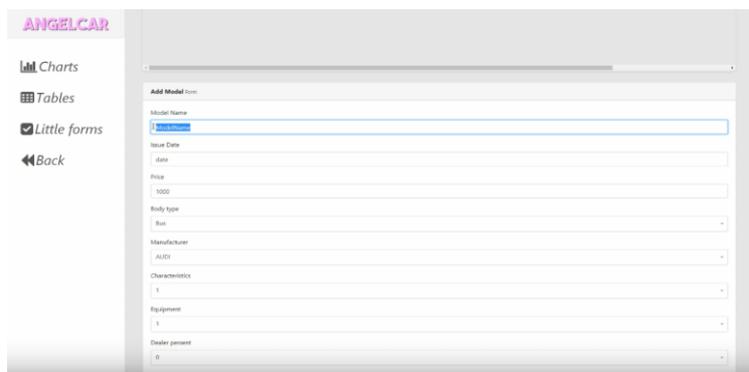


Рисунок 1. – Страница добавления новых автомобилей

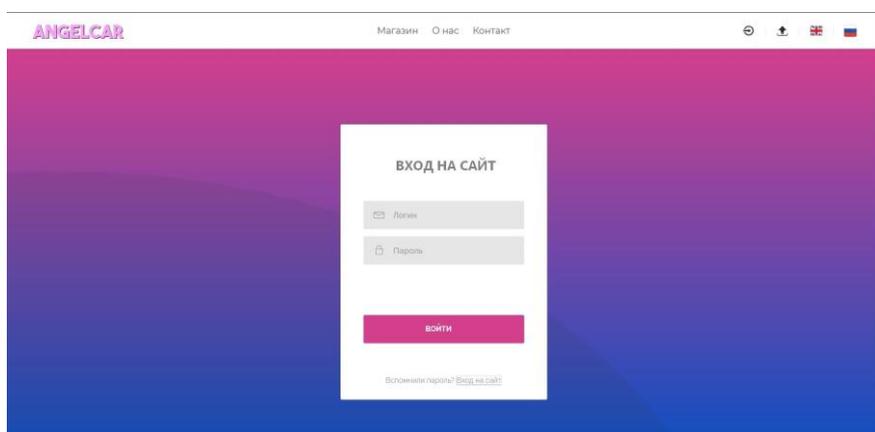


Рисунок 2. – Страница авторизации

Заключение. В результате был разработан графический пользовательский интерфейс для веб-приложения «Продажа автомобилей «AngelCar», отвечающий поставленным требованиям как по функциональности, так и по понятности, простоте и привлекательности. Также данный интерфейс был протестирован на предмет некорректной или нестабильной работы, проблем выявлено не было.

ЛИТЕРАТУРА

1. Автомобильный рынок [Электронный ресурс]. – Режим доступа https://ru.wikipedia.org/wiki/Автомобильный_рынок/. – Дата доступа: 05.03.2020.
2. Обзор технологии Java Server Pages [Электронный ресурс]. – Режим доступа <http://java-online.ru/jsp.xhtml/>. – Дата доступа: 29.03.2020.

ПРЕИМУЩЕСТВА ВЕБ-ПРИЛОЖЕНИЯ «ПРОДАЖА АВТОМОБИЛЕЙ «ANGELCAR»

А.Д. ВИНОГРАДОВА

(Представлено: канд. техн. наук, доц. О.Н.ПЕТРОВИЧ)

В данной статье рассматриваются преимущества веб-приложения «AngelCar» на основе сравнения с другими подобными приложениями.

Введение. В настоящее время существует множество различных сервисов для покупки авто. Каждый из них обладает своими достоинствами и недостатками.

Для сравнения были определены критерии, такие как наличие достаточного количества марок и моделей автомобилей, их описание, понятный и удобный интерфейс сайта, возможность оформления заявок на автомобиль, оплата за авто через Интернет. Данные критерии выбраны не случайно, они позволяют получить наиболее полную картину о качестве автомобильного сервиса.

Преимущества разрабатываемого приложения. Веб-приложение «Продажа легковых автомобилей» – это некий аналог магазина, в котором представлены существующие образцы новых автомобилей. Основными элементами предметной области являются поставляемые автомобили, которые обозначены следующими признаками: номер, производитель, модель, пробег, цвет, цена. Также в отдельные элементы ПО выделены основные составляющие технической части автомобиля – базовая и дополнительная комплектация авто. Данная предметная область содержит информацию о покупателях, продавцах и автодилерах (организациях которые связаны с производителем и отвечают за сбыт автомобилей), поставляющих новые автомобили с заводов-производителей.

Рассмотрим два наиболее популярных сервиса, один российских и один белорусский: Рольф, Autosom.by.

Рольф – крупнейший автомобильный дилер в России. Сотрудничает с 21 маркой автомобилей, включая самые крупные и известные (интерфейс сайта изображен на рисунке 1 и рисунке 2) [1].

Достоинства:

- 1 Приятный и понятный интерфейс.
- 2 Наличие как новых, так и поддержанных автомобилей.
- 3 Наличие специальных подборок автомобилей для конкретных целей.
- 4 Возможность оформления покупки в кредит, оформления страховки.
- 5 Широкий выбор моделей автомобилей и удобный расширенный поиск.
- 6 Возможность купить запчасти.
- 7 Подробнейшее описание каждого автомобиля.

Недостатки:

- 1 Представлены только два крупнейших города России.
- 2 Малое количество марок автомобилей, некоторые из популярнейших марок не представлены вовсе.

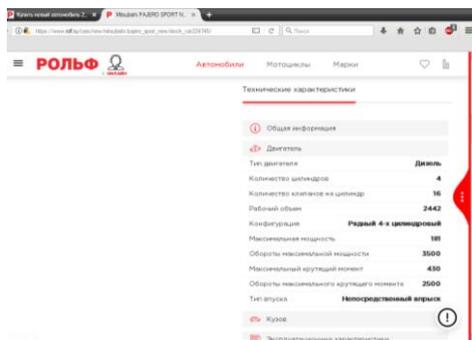


Рисунок 1. – Характеристики выбранного авто на сайте дилера «Рольф»

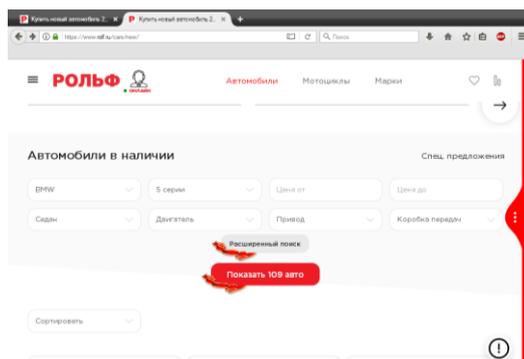


Рисунок 2. – Поиск нужного авто на сайте дилера «Рольф»

Autocom.by – крупный белорусский дилер легковых автомобилей, специализируется как на продаже машин в Минске, так и на их доставку под заказ из России и стран Европы (на рисунке 3 представлен интерфейс сайта данного дилера) [2].

Достоинства:

- 1 Возможность доставки автомобиля под заказ.
- 2 Сравнительно низкие цены.
- 3 Большое количество фотографий автомобиля.
- 4 Наличие возможности покупки новых автомобилей, поддержанных, а также с аукциона.

Недостатки:

- 1 Существует множество недоработок в интерфейсе сайта.
- 2 Малое количество марок и моделей автомобилей.
- 3 Недостаточно подробное описание авто.

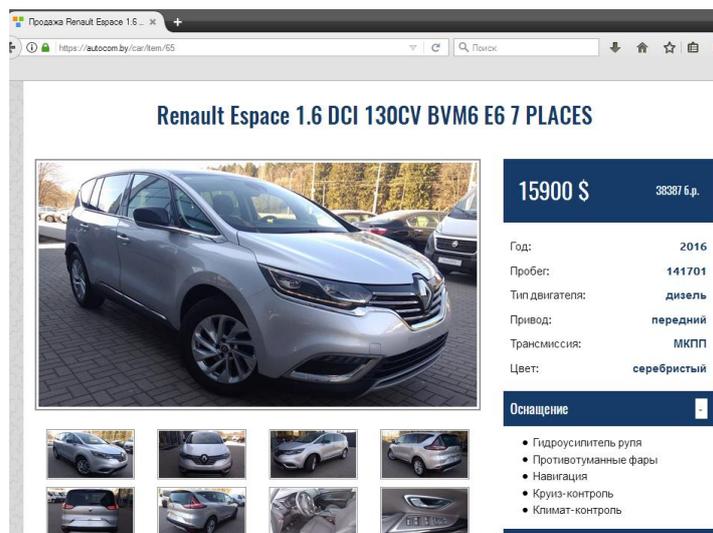


Рисунок 3. – Характеристики выбранного авто на сайте дилера «Autocom.by»

Таким образом, стоит отметить, что главным недостатком аналогов является малое количество марок или моделей автомобилей. Также в ходе сравнительного анализа аналогов и прототипов выявлены проблемы с графическим интерфейсом, описание автомобилей не везде проработано в достаточной мере. Именно эти недостатки следует учесть при разработке данного дипломного проекта.

При начальном анализе также полезно строить диаграмму потоков данных. Такая диаграмма описывает внешние по отношению к системе источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ. Диаграмма потоков данных для предметной области приложения представлена на рисунке 4.

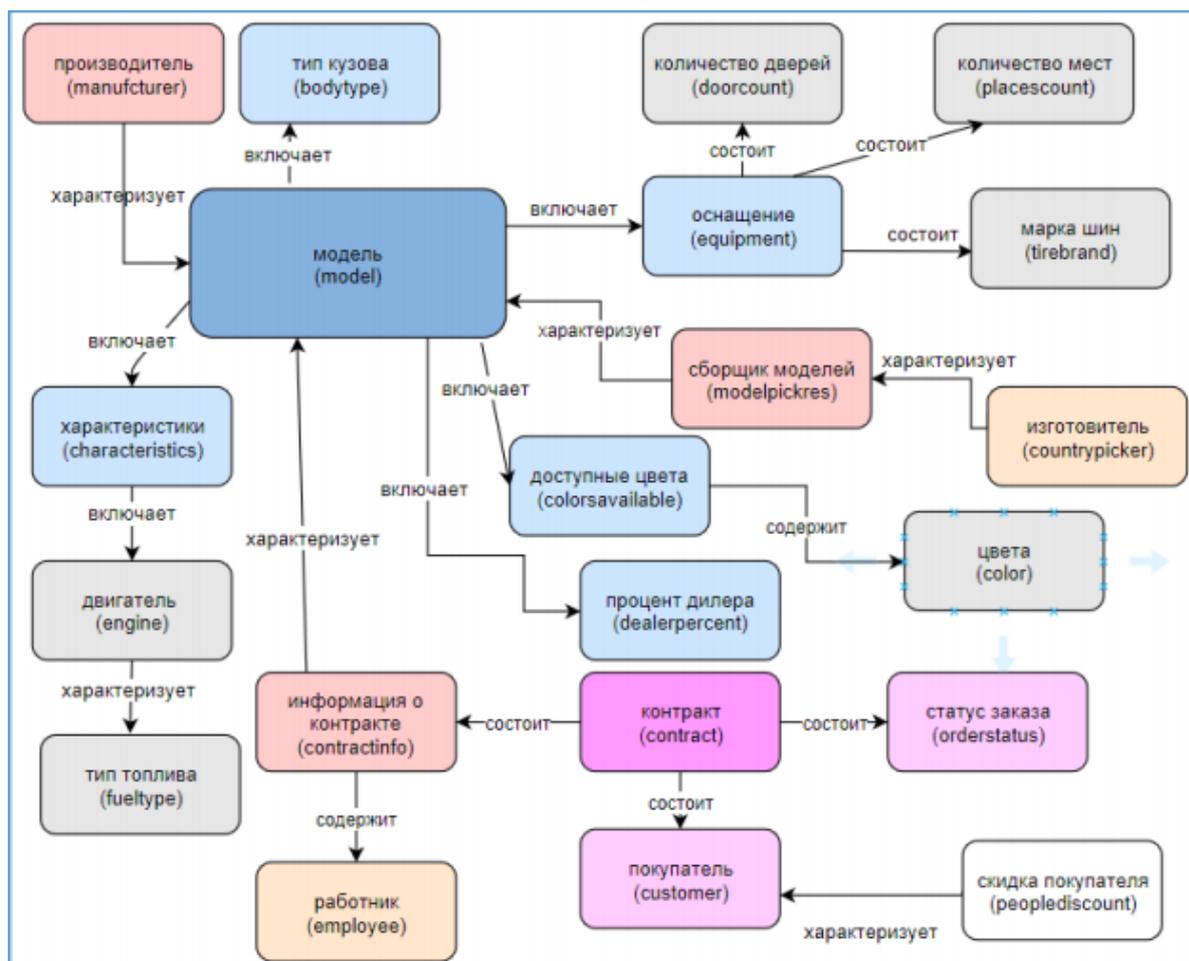


Рисунок 4. – Диаграмма потоков данных

Веб-приложение «Продажа автомобилей AngelCar» предоставляет пользователю следующую функциональность:

- 1 Просмотр сайта (просмотр информации о легковых автомобилях).
- 2 Выбор автомобиля с определенными характеристиками.
- 3 Добавление понравившихся автомобилей в «Избранное» (при условии авторизации клиента).
- 4 Оформление заявки на нужный автомобиль (при условии авторизации клиента).
- 5 Внесение оплаты (частичной или полной) за автомобиль через Интернет (при условии авторизации клиента).
- 6 Добавление информации об автомобилях.
- 7 Изменение информации об автомобилях.
- 8 Просмотр заявок от клиентов на веб-странице.
- 9 Обработка заявок клиентов (подтверждение заказа клиента, оформление заявки).
- 10 Получение отчета в формате doc и на веб-странице с данными о количестве проданных авто и о полученном доходе за определенный период.
- 11 Просмотр справочника клиентов на веб-странице.
- 12 Просмотр внесения клиентом оплаты за автомобиль.
- 13 Предоставление информации о начислении пени в случае просрочки оплаты.

Заключение. Тема продаж и покупок автомобилей очень популярна в наше время, так как сам предмет – автомобиль пользуется большой популярностью у населения. Для облегчения поиска нового автомобиля во многих городах организуются специальные стоянки или закрытые магазины, называемые автосалоном, в которых клиенты могут подобрать себе автомобиль по вкусу. Огромную популярность набирают интернет-сайты, на которых находится информация об автомобилях, представленных к продаже.

Использование данного приложения упрощает поиск информации об автомобиле, как продавцу, так и пользователю. С помощью веб-приложения администратор с легкостью отслеживает динамику

продаж на веб-странице или в doc-формате, начисляет пеню в случае неоплаты, добавляет новые автомобили в базу данных; продавец имеет право просматривать список клиентов, обрабатывать заявки клиентов (подтверждать заказ клиента, оформлять заявки), просматривать данные об оплате за автомобиль клиентом; клиенту же доступно следующее: просмотр сайта, добавление товара в Избранное, оформление заявки на понравившийся товар, оплата через Интернет за автомобиль.

Таким образом, разработанное веб-приложение является эффективной инновационной технологией, позволяющей наилучшим образом объединить в себе потребности клиентов и дилера, упростить работу сотрудников. Следует отметить, что данный программный продукт может быть использован (и даже рекомендуется) в небольших фирмах, которые занимаются закупкой и продажей автомобилей, для совершения более организованной и оперативной работы. При необходимости, данный программный продукт может быть скорректирован под запросы и задачи определенной фирмы. Преимущества данной программы, заключаются в том, что этот продукт позволяет решать конкретные задачи и не требует специальных знаний в программировании, т.к. интерфейс интуитивно понятен.

ЛИТЕРАТУРА

1. Рольф – крупнейший российский дилер России [Электронный ресурс]. Режим доступа <https://www.rolf.ru/>. Дата доступа – 27.09.2020 г.
2. Продажа авто в Минске [Электронный ресурс]. Режим доступа <https://www.autocom.by/>. Дата доступа – 28.09.2020 г.

УДК 004.021

**СОЗДАНИЕ ТРЁХМЕРНЫХ МОДЕЛЕЙ И АУДИОРИЯДА
ДЛЯ ПРОГРАММНО-АППАРАТНОГО VR-ТРЕНАЖЁРА
ПО ОТРАБОТКЕ НАВЫКОВ ПОВЕДЕНИЯ В ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ****Е.Ю. ГАРИСТ, П.В. СИНЯК,
(Представлено: канд. физ.-мат. наук, доц. О В. ГОЛУБЕВА)**

В статье рассмотрены принципы создания графического окружения и звукового сопровождения в среде разработки Unreal Engine 4 (UE4) для VR-тренажёра по отработке навыков поведения в чрезвычайных ситуациях.

Введение. Принципы построения графического окружения для VR-систем базируются на создании 3D-моделей. В UE4 самые распространенные и удобные файлы имеют расширение .fbx. Их можно создавать вручную или воспользоваться готовыми.

Звук используется для управления вниманием пользователя и усиления эффекта присутствия.

Данная статья направлена на подробное описание технологий создания виртуальных объектов и правил закрепления за ними звуков.

Основной раздел. Для создания VR-среды использованы 3D-модели, находящиеся в открытом доступе на сторонних сайтах, но доработанные с точки зрения функциональности и визуальной достоверности. При разработке продукта оказались полезными ресурсы хранилища UE4. Самые значимые 3D-модели тренажёра созданы самостоятельно в среде Autodesk 3ds Max. Именно они позволяют достичь наибольшей степени погружения, так как выполнены с применением хорошо узнаваемых и проработанных текстур, размеров, окрасок, деталей.

Процесс создания 3D-модели состоит из нескольких этапов.

1. Создание собственных или поиск готовых 2D-ракурсов 3D-модели.

Чем сложнее модель, тем больше ракурсов требуется для точного воспроизведения.

2. Создание виртуальной студии.

2D-ракурсы объединяются в пространственный чертёж – виртуальную студию. Она позволяет размещать 2D-ракурсы так, чтобы возник нужный 3D-объект.

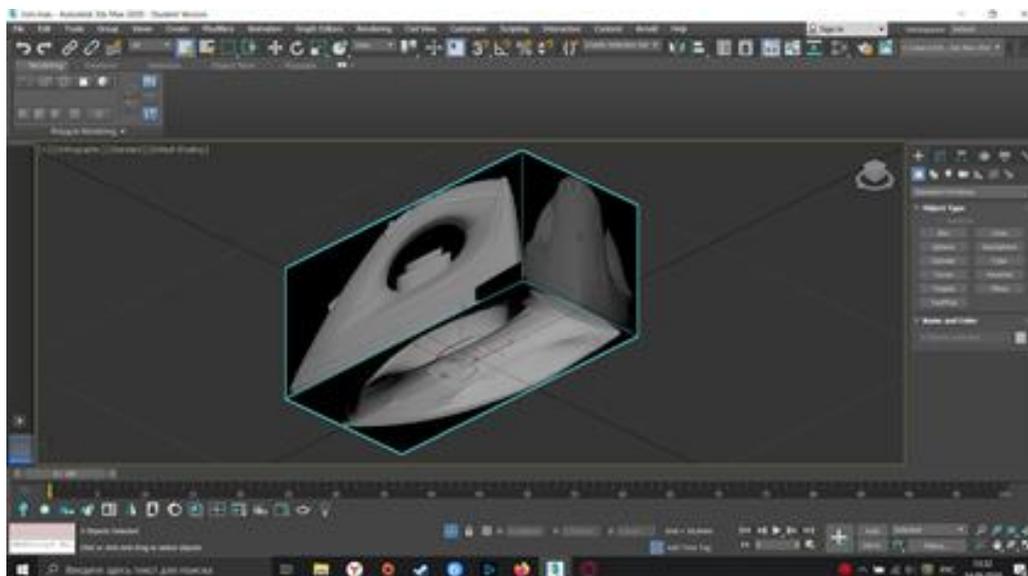


Рисунок 1. – Виртуальная студия

3. Создание и адаптация примитивов.

Силуэт 3D-объекта наполняется заранее созданными деталями.

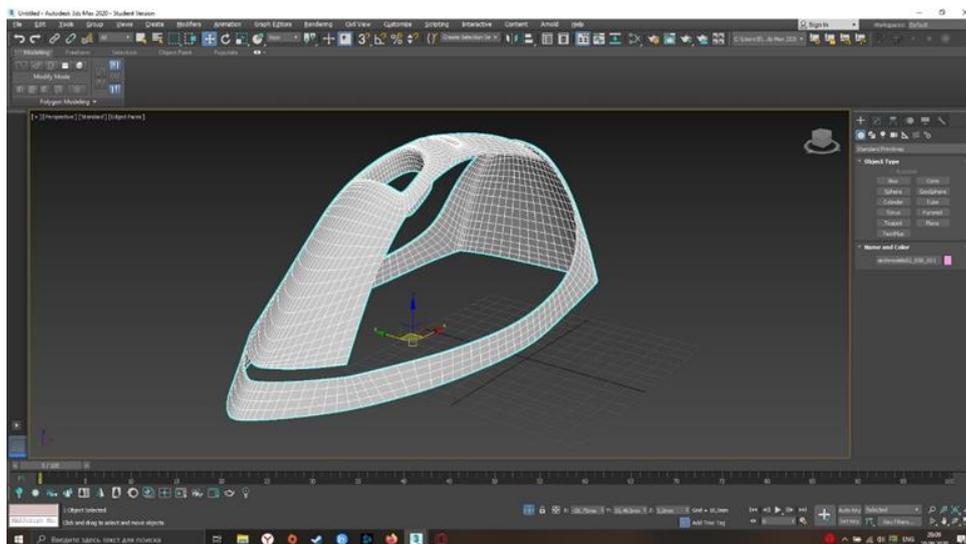


Рисунок 2. – Силуэт модели

4. Детализация.

Добавление мелких частей к основному образу, дополнение уже существующих частей силуэта, «оживление» модели.

Если речь идет о заимствованной модели, то работа над ней начинается с четвертого этапа, поскольку она является либо полноценной, либо силуэтом. Ненужные детали убираются, а дополнительные присоединяются.

5. Настройка текстур и материалов.

Сначала создается материал: задаются характеристики (блеск, отражение) того, как будет выглядеть окрашенный объект. Затем накладывается текстура: картинка, которая будет иметь обозначенные выше характеристики.

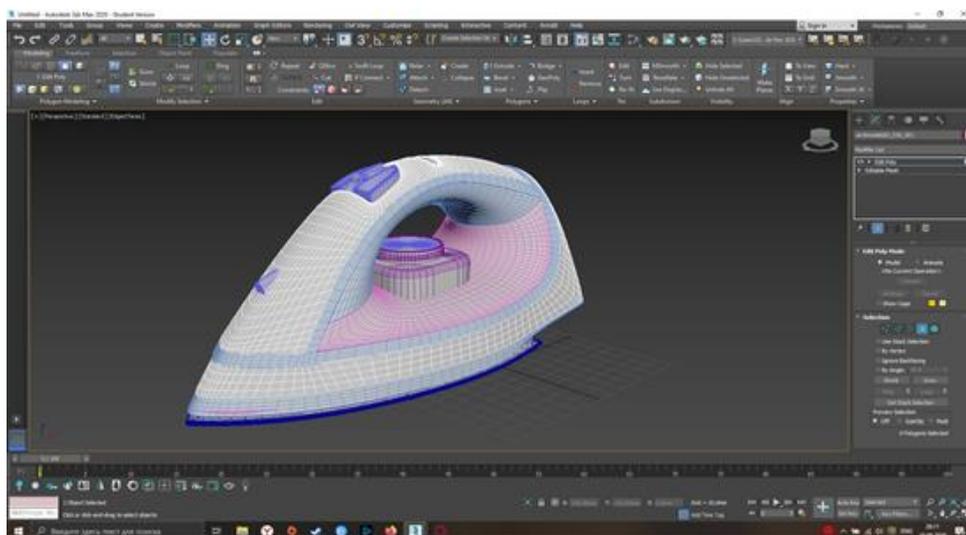


Рисунок 3. – Наложение текстуры

6. Импорт в UE4.

После выполнения всех этапов 3D-модель импортируется в среду разработки UE4 для обработки и масштабирования. В результате получается пригодный к использованию объект.

Для придания графической среде, в данном случае – квартире, реалистичности, помимо основных, разработанных для обучения моделей, добавляются сторонние, чтобы у пользователя складыва-

лось ощущение правдоподобности происходящего. Это предметы интерьера, придающие квартире состояние «обжитости», и фон улицы, похожий на белорусскую местность и дающий ощущение того, что пользователь находится на приличной высоте от земли.

Звуковое сопровождение полностью взято с интернет-платформ в формате mp3. Это характерные звуки, призванные в первую очередь оповещать пользователя о начале чрезвычайных ситуаций и направлять к ним. Также звуки дают понять, что ситуация прогрессирует, решена верно (звук горения, утечки вещества пропадает) или неверно (при тушении водой подключенных к питанию электроприборов появляется новый звук потрескивания; при попадании воды на горящую сковороду – звук взрыва).

В качестве основных использованы звуки длительные, независимые от действий пользователя, и направляющие: звук огня при возгорании масла и электроприборов; звук воды при прорыве трубы отопления; звук утечки газа при неисправности газовой трубы; звук пожара в квартире; звук сирены; звук, сопровождающий короткое замыкание в электроприборах.

Также в симуляторе представлены тематические звуки, призванные усилить эффект присутствия.

Дополнительные звуки – короткие, зависимые (вызываются пользователем) и атмосферные: звук воды при открытии крана; звук улицы при открытии окна; звук жидких и сыпучих материалов для тушения; звук поворота вентиля; звук взрыва кипящего масла; звук переключателей света и электричества.

Каждый звук подобран максимально близко к предмету, поверхности, материалу, издающему этот звук, упор в разработке звукового сопровождения сделан на реализм и качество.

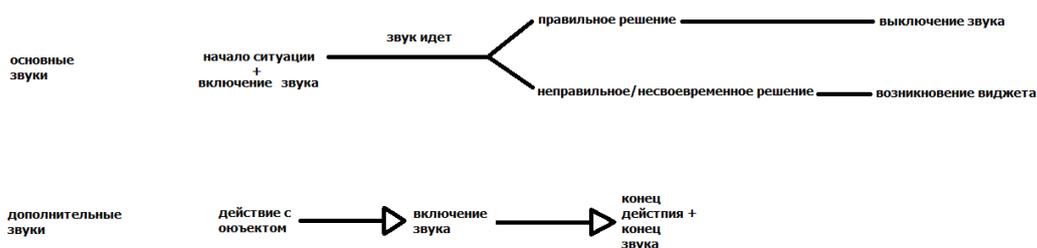


Рисунок 4. – Схема работы звукового сопровождения

Для реализации обратной связи в тренажёре-симуляторе предусмотрены виджеты – упрощенные меню с ситуативным текстом. Для VR-продуктов работает правило «чем крупнее, тем лучше», соответственно инструкции и подсказки располагаются близко к глазам игрока, написаны содержательно и коротко. Чтобы не заставлять пользователя искать виджеты, приближаться к ним, сам игрок перемещается в нужную позицию лицом к тексту с учётом расстояния, удобного для фокусировки – 0,5 метра. Это расстояние, на котором не нужно напрягать глаза для чтения, но если пользователь имеет плохое зрение, то достаточно немного наклониться – не более чем на 10 см.



Рисунок 5. – Схема работы виджета

Заключение. Программно-аппаратный комплекс-тренажёр на основе технологий виртуальной реальности «Квартира опасностей» по отработке навыков поведения в чрезвычайных ситуациях разработан авторами статьи в рамках договора № 2082 от 08 июля 2020 по заказу Представительства Детского фонда Организации Объединенных наций (ЮНИСЕФ) для Центра безопасности МЧС в г.Новополоцке.

ЛИТЕРАТУРА

1. Help4Students [Электронный ресурс] Методология моделирования данных ИС. – Режим доступа: <https://hlp4asp.wordpress.com/2011/10/13/методология-моделирования-данных-ис>. – Дата доступа: 27.09.2020.
2. CitForum [Электронный ресурс] Семантическое моделирование данных, ER-диаграммы. – Режим доступа: http://citforum.ru/database/osbd/glava_24.shtml. – Дата доступа: 27.09.2020.

УДК 004.021

СОЗДАНИЕ ЛОГИКИ ВЗАИМОДЕЙСТВИЙ В ПРОГРАММНО-АППАРАТНОМ VR-ТРЕНАЖЁРЕ ПО ОТРАБОТКЕ НАВЫКОВ ПОВЕДЕНИЯ В ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ В СРЕДЕ РАЗРАБОТКИ UNREAL ENGINE 4

Е.Ю. ГАРИСТ, П.В. СИНЯК

(Представлено: канд. физ.-мат. наук, доц. **О.В. ГОЛУБЕВА**)

В статье рассмотрены принципы построения графического кода в среде разработки Unreal Engine 4 (UE4) для создания VR-тренажёра по отработке навыков поведения в чрезвычайных ситуациях.

Введение. Unreal Engine 4 – широко распространенный игровой движок с многофункциональной базой для создания виртуальных приложений. Визуальный интерфейс blueprint, который используется в UE4, позволяет просто и наглядно прописывать логику приложения.

Данная статья направлена на подробное описание технологий создания визуального кода в UE4 и правил взаимодействия пользователя с созданной средой.

Основной раздел. В VR-разработке отсутствуют шаблоны построения окружения, так как каждая среда служит для уникальных целей. Основой среды VR-тренажёра по отработке навыков поведения в чрезвычайных ситуациях является обычная среднестатистическая квартира, состоящая из двух комнат: кухни и гостиной. Для идентификации игроком чрезвычайной ситуации не требуется много времени и подсказок. Каждый объект, являющийся основой чрезвычайной ситуации, находится в привычном месте узнаваемой для любого человека комнаты и окружен предметами, близкими по смыслу, а также необходимыми средствами решения возникающей чрезвычайной ситуации, что упрощает ориентирование и сокращает время на устранение опасности.

Взаимодействие с виртуальными моделями приближено к реальным условиям, игроку для устранения опасности нужно к ней приблизиться и взаимодействовать с предметами, с помощью которых можно опасность устранить. Это происходит интуитивно. Каждая опасность и предмет, являющийся способом её устранения, расположен не более чем в полуметре от пользователя. Выбор такого интервала обусловлен тем, что человеку неудобно фокусироваться на объектах, находящихся от него на расстоянии большем, чем 0,5 м, так как «далёкие объекты» теряют свою объемность, значимость, и, соответственно, не привлекают внимание. Для сохранения внимания игрока и стимулирования к скорейшему взаимодействию разработчики тренажёра разместили объекты взаимодействия в зоне быстрого доступа.



Рисунок 1. – Схема зон размещения разных видов контента

Главная задача VR-тренажёра – закрепить у пользователя на уровне моторной памяти правильный алгоритм действий. Большинство взаимодействий в тренажёре-симуляторе реализовано именно с этой целью. Однако для придания реалистичности происходящего имеются и побочные взаимодействия: открытие и закрытие крана с водой, включение и выключение света в комнатах, возможность брать фрукты и багет с кухонного стола. Они не имеют отношения к предотвращению чрезвычайных ситуаций, но убеждают

пользователя в реальности происходящего. Реализованные побочные не отвлекают игрока от основной задачи по ликвидации чрезвычайных ситуаций и не мешают принимать важные решения.

Взаимодействия пользователя с объектами построены следующим образом. Выливание воды, высыпание земли и соды происходит при наклоне ёмкости на 89° и больше. Объекты, которые нужно брать, разделены на две категории: с ручкой и без ручки. Если объект имеет ручку, то при взятии она попадает в ту руку игрока, которой он взаимодействует с объектом. Если объект не имеет ручки, то игрок может брать его с любой стороны. Для открытия или закрытия дверей, окон пользователю нужно коснуться ручки на этом объекте. Для включения и выключения чего-либо пользователю нужно коснуться самого объекта.

Взаимодействие объекта с объектом построено на отслеживании коллизии – столкновения. В зависимости от цели можно настроить их отталкивание, сцепление и т. п.

VR-среда имеет преимущество гибкой физики, что означает возможность использования ориентиров реального мира для построения упрощённых взаимодействий. Интуитивно понятные действия, которые выполняются легче, чем в реальности, позволяют быстро двигаться к игровой цели.

Все взаимодействия прописываются в коде среды разработки Unreal Engine 4. Это игровой движок, разработанный для создания игр от первого лица. Blueprint – это скриптовая система в Unreal Engine 4, которая представляет собой визуальный код: блоки, содержащие условные конструкции, действия или события, соединяющиеся линиями переходов (нодами).

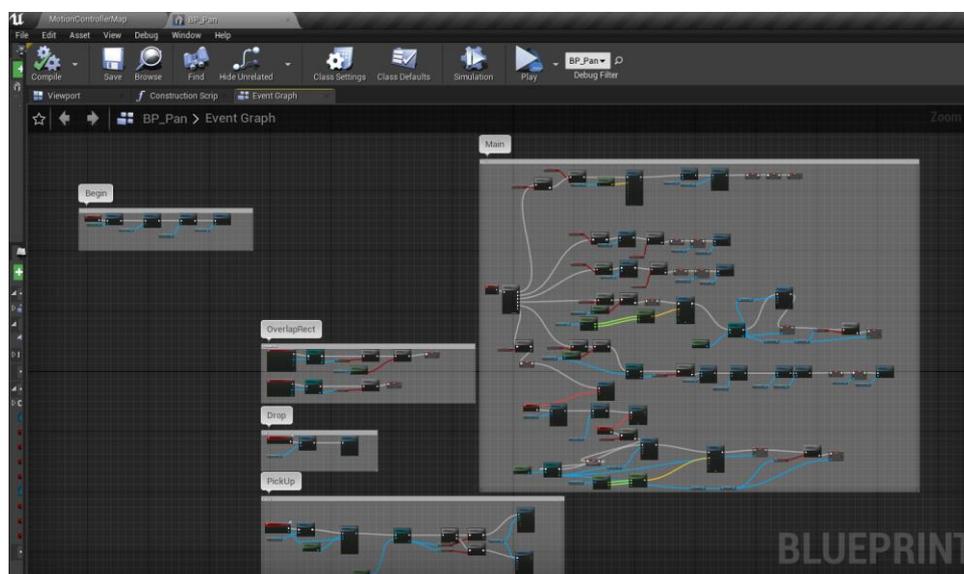


Рисунок 2. – Пример графического кода

Перед написанием кода теоретически продумывается желательная последовательность действий и их конечная цель, а в процессе написания и тестирования изначальный план корректируется, подгоняется под возможности движка. Разрабатываются оптимальные алгоритмы, которые потребляют минимум ресурсов, что позволяет воспроизводить большее количество кадров в секунду. Алгоритмы строятся на базе следующих элементов:

1. Стандартные конструкции: цикл for – for each, оператор условия if – branch и т. п. Их использование гарантирует читабельность кода и легкое изменение его частей, которое не влияет на работу связанных блоков.

2. Действия: участок кода, создающий события. Действия совершаются непосредственно в виртуальной реальности, например, столкновения двух объектов – их пересечение.

3. События: функции или методы, которые вызываются при каком-либо условии. Могут быть стандартными либо созданными для особых целей. Например, обрабатываются столкновения объектов – событие begin overlapped. Для перемещения динамического объекта применяется событие, зависящее от таймера. Динамичность объекта устанавливается свойством этого объекта в окне Details. Чтобы объект мог двигаться и испытывать физику (виртуальное подобие схем движения объектов в зависимости от массы, расстояния и т. п.), устанавливается свойство Movable. Если надо запретить движения и физику, то ставится свойство Static.

Звуковое сопровождение также подключается в коде. Нужная звуковая дорожка заносится в ассет (asset) – материал программы, позволяющий использовать в проекте готовые решения, созданные другими людьми. Созданный ассет называется Sound Cue. Он позволяет манипулировать несколькими звуками

и комбинировать их, регулировать уровень громкости и задавать цикличность – постоянное проигрывание звука или только при вызове. После настройки Sound Cue звуковой эффект используется в проекте.

Заключение. Программно-аппаратный комплекс-тренажёр на основе технологий виртуальной реальности «Квартира опасностей» по отработке навыков поведения в чрезвычайных ситуациях разработан авторами статьи в рамках договора № 2082 от 08 июля 2020 по заказу Представительства Детского фонда Организации Объединенных наций (ЮНИСЕФ) для Центра безопасности МЧС в г. Новополоцке.

ЛИТЕРАТУРА

1. Virtual Reality development [Электронный ресурс] Документация по разработке виртуальной реальности в Unreal Engine 4. – Режим доступа: <https://docs.unrealengine.com/en-US/Platforms/VR/index.html>. – Дата доступа: 19.09.2020.
2. Habr [Электронный ресурс] Тьюториал по Unreal Engine. – Режим доступа: <https://m.habr.com/ru/post/344394/>. – Дата доступа: 19.09.2020.

УДК 004.05; УДК 338.45.01

ОСНОВНЫЕ ПРИНЦИПЫ ВЫВОДА НА РЫНОК МОБИЛЬНОГО ПРИЛОЖЕНИЯ ПО ПОИСКУ ПАРТНЁРОВ ДЛЯ ЗАНЯТИЯ СПОРТОМ

А.Ю. ГЕРЕЦ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)

По всему миру активно продвигается идея здорового образа жизни. А быть профессиональным спортсменом считается престижно. Но, в молодёжной среде спорт не в приоритете, а, наоборот, пытаются увильнуть от физических нагрузок. Для Республики Беларусь здоровье подрастающего поколения важно, в связи с чем предлагается создать и вывести на рынок спортивное приложение.

В таблица 1 представим полученные данные о конкурентах.

Таблица 1 – Анализ конкурентов

Наименование услуги	Team Kraft	Ride with me	Nike+ Run Club	Fitior	Средняя розничная цена конкурентов	Минимальная розничная цена конкурентов	Розничная цена организации
Одно скачивание платной версии, руб.	4,08	3,07	2,95	3,56	3	3	3,15
Коллаборация, руб.		476,41		584,69	531	476	472,50

По данным, представленным в таблице, можно сделать вывод, что все четыре конкурента предоставляют возможность платной подписки на приложение, которое даёт предоставляет возможности. Средняя цена конкурентов 3,0 руб., а предлагаемое к созданию приложение обойдётся в 3,15 руб., что немного больше (+5%). В отношении коллаборации отметим, что такая возможность есть только в двух приложениях. В приложении «Nike+ Run Club» коллаборации нет, т.к. это приложение больше для расширение силы влияния бренда «Nike», популяризации продукции организации. Коллаборация возможна в мобильном приложении «Ride with me» и «Fitior». Средняя цена привлечения для спонсора является 531 руб., а минимальная цена 476 руб., но приложение пока ещё не работает в Республике Беларусь.

Далее осуществим анализ рынка схожих мобильных приложений, отобразив ёмкость и долю рынка. Полученные данные представим в таблице 2.

Таблица 2 – Анализ рынка схожих мобильных приложений

Год	2022	2023	2024	2025	2026	2027	2028	2029
1	2	3	4	5	6	7	8	9
Одно скачивание платной версии								
Ёмкость рынка, тыс. руб.	3553,5	3600,1	3788,2	3939,7	4117,0	4322,9	4539,0	4811,4
Доля рынка, %	3	4	4	4	4	4	4	3
Темп роста рынка, %	3	3	3,5	4	4,5	5	5	6
Объем продаж (нат. ед./ден. ед.)	47 340	66 717	70 424	70 424	70 424	70 424	70 424	70 424
Количество конкурентов	4	4	5	6	6	6	7	7
Количество покупателей, тыс. чел.	1870,3	1926,4	1993,8	2073,5	2166,9	2275,2	2389,0	2532,3
Коллаборация								
Ёмкость рынка, руб.	314421	323854	335189	348596	364283	382497	401622	425719
Доля рынка, %	18	20	23	25	28	30	33	37
Темп роста рынка, %	3,5	4	5	5	6	6	7	7
Объем продаж (нат. ед./ден. ед.)	254160	285930	301815	301815	301815	301815	301815	301815

Окончание таблицы 2

1	2	3	4	5	6	7	8	9
Количество конкурентов	2	2	2	3	3	3	3	4
Количество основных покупателей	4 005	4 126	4 270	4 441	4 641	4 873	5 116	5 423

Из данных, представленных в таблице 2, можно сделать выводы, что ёмкость рынка мобильных приложений к 2022г. составит 3,6 млн. руб., а к 2029г. уже 4,8 млн. руб. Для Республике Беларусь рынок мобильных приложений ещё новый, молодой и с низкой конкуренцией. К этому стоит добавить более дешёвое производство продукта, чем в ближнем зарубежье. Предлагаемое к реализации мобильное приложение по платным установкам составит порядка 3-4% в год, а темп роста рынка колеблется в пределах 3-6% в год. Конкуренция в сфере спорта предполагает 4-7% ежегодного роста, а соответственно, постепенно рынок насытится. По направлению коллаборация ёмкость рынка составляет 314,4-425,7 тыс. руб. В виду того, что коллаборация на отечественном рынке в мобильном сегменте не высока, то станет возможным занять долю рынка 18% в 2027г. И 37% в 2029г. Число возможных партнёров колеблется 4,0 – 5,4 тыс. шт. (физические (блогеры, ИП) и юридические лица).

В предлагаемом спортивном приложении пользователю предоставлена возможность формирования места. Эта функция делает каждое событие уникальным и познавательным даже на уровне смены обстановки, не говоря уже о совершенно новых людях, и, конечно же, новых видах спорта, которые можно открывать для себя хоть каждый день. То есть каждый участник сможет стать организатором мероприятия и при пассивном участии собрать команды для спортивного мероприятия либо совместного занятия спортом от двух человек. TeamMember сможет информировать пользователей о ближайших площадках, на которых можно собираться, также забронировать зал и разделить платёж между участниками мероприятия.

При разработке и поддержании работоспособности приложения используется Cloud с Serverless приложениями. В качестве back-end использованы cloud функции, а также front-end для React.

С целью продвижения, роста числа скачиваний и аккумуляции большой аудитории целесообразен запуск таргетированной рекламы в Инстаграм, ВКонтакте, возможно даже на Ютуб и Телеграмм. Во время рекламы приложения предусмотрена демонстрация динамического проморолика на 15 секунд, в который «зашията» ссылка на скачивание приложения. Целесообразно размещение статей на тематических площадках.

В таблице 3 представлена модель инновационного проекта Канвас.

Приложение первоначально протестировано на работоспособность. Далее планируется учитывать количество скачиваний, число платных подписок на приложение, выручку от рекламодателей.

Маркетинговая стратегия (дифференциация) инновационного продукта заключается в следующем: работать на одном целевом рынке, т. е. сегмент обязательно должен быть связан с занятиями спортом. В приложении доступны несколько возможностей для разных аудиторий:

- собираться единомышленникам для проведения спортивных мероприятий;
 - через приложения для традиционного бизнеса станет возможным собирать на мероприятия свою целевую аудиторию с наименьшими затратами чем использование рекламной компании;
 - любое физическое лицо может стать инициатором проведения спортивного мероприятия и проявить свои лидерские качества;
 - на мероприятия можно привлекать блогеров, которые в результате получают рост своей популярности.
- Для реализации маркетинговой стратегии планируется использовать следующие мероприятия:
- после разработки спортивного мобильного приложения осуществлён запуск таргетинговой рекламы, проанонсировано приложение через СМИ и специализированные интернет-ресурсы;
 - в самом приложении предложена подписка за расширенный функционал приложения;
 - в приложении предусмотрен раздел для сотрудничества с инициативными людьми либо юридическими лицами с предложением поучаствовать в мероприятии как спонсор;
 - ведётся работа с отзывами и предусмотрена обратная связь пользователям, своевременно вносятся изменения и доработки по устранению багов;
 - PRO-версия с полным функционалом на месяц при оплаченном пакете;
 - в соцсети LinkedIn искать бренды, которые заинтересованы в сотрудничестве и готовы к сотрудничеству.

В начале деятельности можно озвучивать цену незначительно ниже расчётной.

Таблица 3 – Модель Канвас мобильного приложения

<p>8 Ключевые партнеры</p> <p>Разработчики приложения, организаторы спортивных мероприятий, владельцы облачных конструкторов (напр. Lambda)</p>	<p>7 Ключевые действия</p> <p>Приложение планируется возвращать преимущественно органически.</p> <p>Учёт доход и расходов планируется отслеживать на электронных кошельках и переноситься в документацию.</p> <p>Внедрить реферальную систему. Например, за 2-3 приглашённых друга (сказали приложение) один месяц бесплатного пользования сервисом</p>	<p>2 Ключевые ценности</p> <p>Для школьников можно организовывать бесплатные мероприятия либо делать рассылку на такие мероприятия</p> <p>Для студентов и людей 23-30 лет можно предлагать собираться в спортзалах и оплату делить на всех</p> <p>Приложение позволит находить людей по спортивным интересам, также можно стать инициатором события</p>	<p>4 Взаимоотношения с клиентами</p> <p>Удалённые и доброжелательные.</p> <p>Акцент сделан на создание некой соцсети, в которой люди смогут общаться и в онлайн, но с более скудным функционалом, т. к. мероприятия направлены на оффлайн точки</p> <p>Собирать отзывы о пользовании приложением</p>	<p>1 Сегмент потребителей</p> <p>Молодёжь 14-30, которые ищут напарника для занятий спорта. Также эти люди ищут либо бесплатные спортивные мероприятия, либо недорогие</p>
<p>9 Структура расходов</p> <p>Оплата услуг Lambda, оплата услуг разработчиков, оплата дизайна, оплата за размещение на платформе, откуда приложение планируется скачивать, оплата комиссии за приём платежей</p>	<p>5 Поток доходов</p> <p>Реклама от рекламодателей, плата за расширенный функционал приложения</p>	<p>6 Ключевые ресурсы</p> <p>Работа в соцсетях, реклама в соцсетях.</p> <p>Договариваться с блогерами на проведение спортивных мероприятий</p> <p>Мотивировать молодёжь организовывать встречи</p>	<p>3 Каналы</p> <p>Социальные сети, мессенджеры, система рекомендаций на платформе, где планируется рекламировать приложение</p>	

Заключение. В данной статье представлены основные принципы вывода на рынок мобильного приложения, разработана модель Канвас по приложению TeamMember, представлены пути монетизации приложения, а также проанализированы основные и потенциальные конкуренты в сегменте. Маркетинговая стратегия вывода на рынок нового продукта является значимым фактором.

ЛИТЕРАТУРА

1. Система дистанционного бизнес-образования – Инновационный проект: понятия, основные этапы создания и реализации [Электронный ресурс]. – Режим доступа: <http://www.tehnpark.bybusiness/207.html>. – Дата доступа: 18.09.2020.
2. Fitior – социальная сеть для любителей спорта [Электронный ресурс]. – Режим доступа: <https://vc.ru/tribuna/18392-fitior>. – Дата доступа: 19.09.2020.
3. Nike+ Run Club [Электронный ресурс]. – Режим доступа: https://www.nike.com/au/en_gb/c/running/nike-run-club. – Дата доступа: 21.09.2020.
4. Ridewithme – сервис для организации спортивных событий [Электронный ресурс]. – Режим доступа: <https://vc.ru/tribuna/11643-ridewithme>. – Дата доступа: 19.09.2020.
5. TeamKraft – сервис для организации спортивных событий [Электронный ресурс]. – Режим доступа: <https://vc.ru/tribuna/21085-teamkraft>. – Дата доступа: 20.09.2020.

УДК 004.05; 339.137.22

АНАЛИЗ ЦЕЛЕВОЙ АУДИТОРИИ И ЭКОНОМИЧЕСКАЯ ЦЕЛЕСООБРАЗНОСТЬ ВЫХОДА НА РЫНОК МОБИЛЬНОГО ПРИЛОЖЕНИЯ ПО ПОИСКУ ПАРТНЁРОВ ДЛЯ ЗАНЯТИЯ СПОРТОМ

А.Ю. ГЕРЕЦ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)

Рассмотрен экономический аспект целесообразности вывода спортивного приложения на рынок, обоснование стратегии, а также важность занятия спортом молодёжью. Разрабатываемое приложение позволит в удобном для молодёжи формате осуществлять поиск напарников для занятия спортом, организовывать встречи, участвовать в различных спортивных мероприятиях.

Сегодня по всему миру активно продвигается идея здорового образа жизни. Однако в молодёжной среде спорт далеко не в приоритете, наоборот, современные молодые люди всячески пытаются увильнуть от различных физических нагрузок, что приводит к сколиозу, ожирению и иным проблемам со здоровьем. При помощи разработанного спортивного приложения молодые люди получают дополнительную возможность заниматься спортом в группе по интересующему виду спорта, причём, полностью отпадёт необходимость самостоятельно часами искать единомышленников в социальных сетях или среди знакомых для создания группы, а также откроется возможность совместного приобретения абонемента на какое-то мероприятие либо спортивный зал.

Используя предоставленные возможности разработанного приложения, каждый желающий сможет и сам стать организатором какого-либо мероприятия, что позволит мотивировать молодых людей повысить свою активность и в некой степени чувствовать себя лидерами. В виду того, что молодые люди много времени проводят on-line в гаджетах, то им теперь, посредством предлагаемого приложения, достаточно просто будет найти компанию для занятий спортом, велопрогулки, занятий на скейтборде, встрече на природе для занятий йогой и пр.

Для обоснования рыночной целесообразности разработки мобильного приложения в формате интернет-анкетирования проведены «полевые» маркетинговые исследования, результаты которых представлены ниже. Они подтверждают целесообразность реализации проекта с позиций потенциальных потребителей.

В таблице 1 представлены мотивы занятия физической культурой у мужчин и женщин.

Таблица 1 – Мотивы занятия физической культурой

Пол	Укрепление здоровья и физическое совершенствование	Культурно-развлекательный мотив	Мотив спортивного совершенства
Мужчины	60,8	24,3	14,9
Женщины	72,8	21,9	5,3

Для выработки обоснованной маркетинговой стратегии, включая ее рекламную составляющую, в рамках рассматриваемых проблем продвижения продукта была исследована физиологическая составляющая как основа качественной разработки архитектуры мобильного приложения. В современном мире одной из ведущих сторон здорового образа жизни является стремление человека к физическому совершенствованию. Физическое совершенствование органически взаимосвязано с удовлетворением биологических и социальных потребностей личности. Систематическое применение средств физической культуры расширяет возможности организма. Тренированный организм более устойчив к гипоксии, гипокинезии, к резко меняющимся погодным условиям, к действию стрессовых факторов, к вибрации, укачиванию и других.

Далее составим таблицу 2, где отобразим разницу артериального давления у тренированных и нет людей.

Таблица 2 – Влияние физической нагрузки на артериальное давление

Состояние	Тренированные люди	Не тренированные люди
Интенсивная физическая нагрузка	максимальное артериальное давление повышается до 200 мл. рт. ст. и более, может долго держаться	максимальное артериальное давление сначала повышается до 200 мл. рт. ст., а после снижается в результате утомления сердечной мышцы. Может наступить обморок
После работы	максимальное и минимальное артериальное давление быстро приходит в норму	максимальное и минимальное артериальное давление долго остаётся повышенным

Физическая работа способствует общему расширению кровеносных сосудов, нормализации тонуса их мышечных сокращений.

Как отмечалось ранее, чем популярнее будет приложение, тем больше спортивной целевой аудитории будет в ней собираться. Для рекламодателей это выгодно, так как позволит продавать свою продукцию

с наименьшей стоимостью, всего лишь организовав тематическое спортивное мероприятия. На проводимых мероприятиях имеется возможность раздачи промо-кода, а сами мероприятия сделать более масштабными и зрелищными. Целевая аудитория приложения, как правило, платежеспособная в рамках прогнозируемой цены на продукт и находится в большинстве своём и в возрастной категории 18-35 лет.

Обоснование актуальности проекта. В таблице 3 отобразим ёмкость и долю рынка [2].

Таблица 3 – Анализ рынка схожих мобильных приложений

Год	2022	2023	2024	2025	2026	2027	2028	2029
Одно скачивание платной версии								
Ёмкость рынка, тыс. руб.	3553,5	3600,1	3788,2	3939,7	4117,0	4322,9	4539,0	4811,4
Доля рынка, %	3	4	4	4	4	4	4	3
Темп роста рынка, %	3	3	3,5	4	4,5	5	5	6
Объём продаж (нат. ед. / ден. ед.)	47 340	66 717	70 424	70 424	70 424	70 424	70 424	70 424
Количество конкурентов	4	4	5	6	6	6	7	7
Количество покупателей, тыс. чел.	1870,3	1926,4	1993,8	2073,5	2166,9	2275,2	2389,0	2532,3
Коллаборация								
Ёмкость рынка, руб.	314421	323854	335189	348596	364283	382497	401622	425719
Доля рынка, %	18	20	23	25	28	30	33	37
Темп роста рынка, %	3,5	4	5	5	6	6	7	7
Объём продаж (нат. ед. / ден. ед.)	254160	285930	301815	301815	301815	301815	301815	301815
Количество конкурентов	2	2	2	3	3	3	3	4
Количество основных покупателей	4 005	4 126	4 270	4 441	4 641	4 873	5 116	5 423

Согласно прогнозу, ёмкость рынка мобильных приложений к 2022г. составит 3,6 млн. руб., а к 2029г. уже 4,8 млн. руб. Для Республике Беларусь рынок мобильных приложений пока ещё новый, молодой и с низкой конкуренцией. К этому стоит добавить более дешёвое производство продукта, чем в ближнем зарубежье. У мобильного приложения имеется три пути монетизации – установка PRO-версии, коллаборации, продажа рекламы внутри приложения. Рост рынка по PRO-версии составит порядка 3-4% в год, а темп рост рынка колеблется в пределах 3-6% в год. Конкурентов в сфере спорта планируется 4-7 шт. и с каждым годом их число будет возрастать, а, соответственно, рынок будет насыщаться [2].

По направлению коллаборация ёмкость рынка составляет 314,4-425,7 тыс. руб. В виду того, что коллаборация на отечественном рынке в мобильном сегменте не высока, то станет возможным занять долю рынка 18% в 2027г. и 37% в 2029г. Число возможных партнёров колеблется 4,0 – 5,4 тыс. шт. (физические (блоггеры, ИП) и юридические лица).

Принятию решения о реализации инвестиционного проекта предшествовала тщательная оценка условий его реализации, отражённая на рисунке 1.

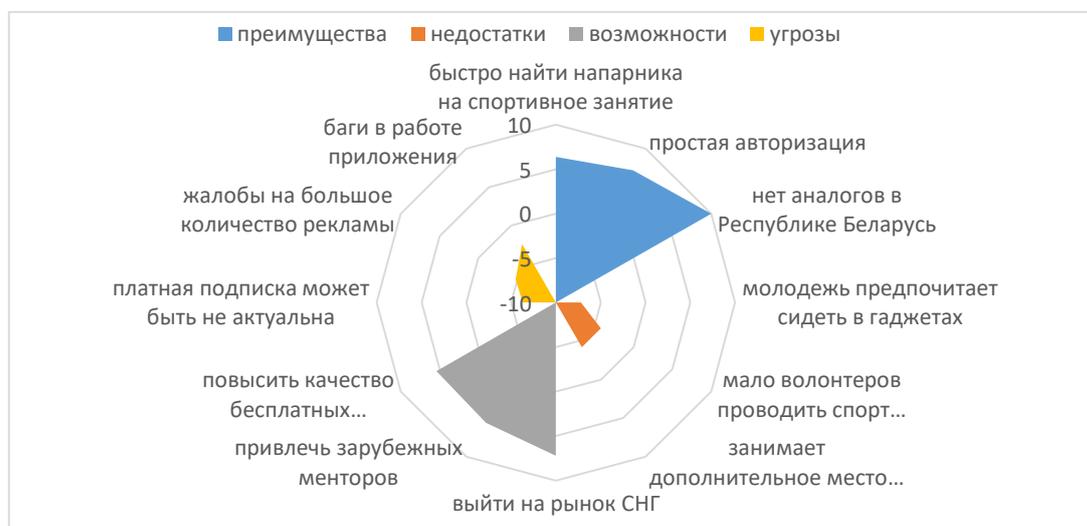


Рисунок 1. – Экспертная оценка преимуществ и недостатков проекта

В связи с отсутствием конкурентов и каких-либо аналогов предлагаемого приложения на отечественном рынке, доступностью и простотой организации процесса, преимущественные позиции получили высокую оценку относительно других факторов, влияющих на реализацию проекта, и, таким образом, послужили ключевым аргументом в принятии решения о старте разработки проекта.

На рисунке 2 представим динамику объёма продаж в натуральном выражении по трём рынкам (отечественный, ближнее и дальнее зарубежье).

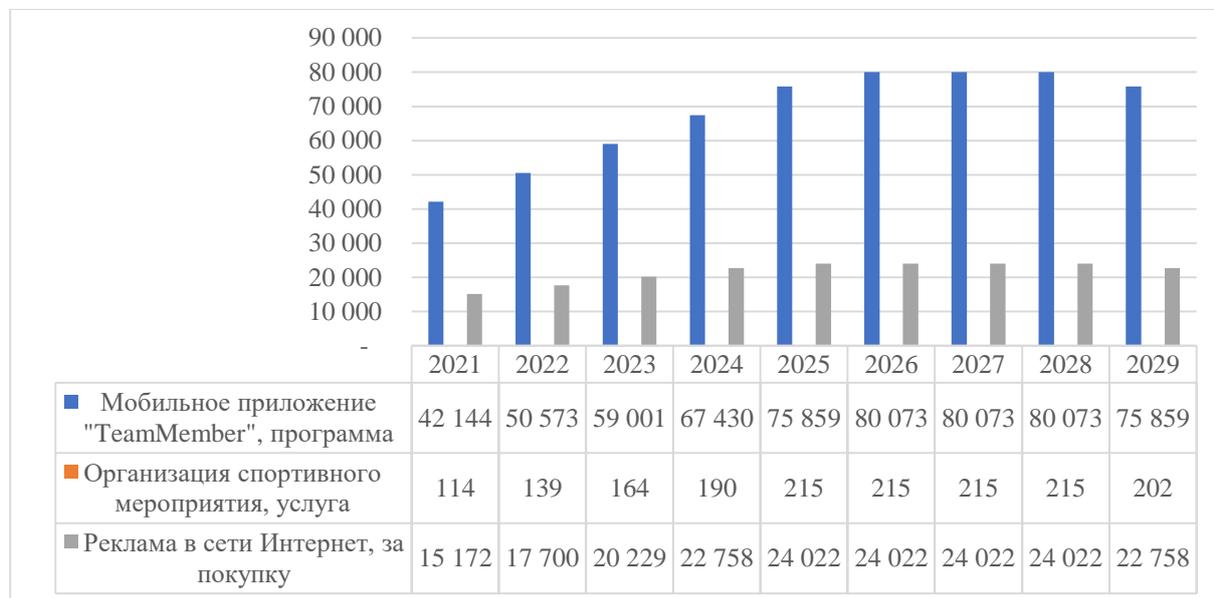


Рисунок 2. – Объем продаж, шт.

Согласно данным, представленным на рисунке 2, в 2021г. планируется организация 114 спонсорских мероприятий, а в 2029г. уже 202 мероприятия. Количество скачиваний возрастёт с 42144 раз. в 2021 г до 75859 раз в 2029г. Реклама в мобильном приложении в 2021г. планируется к размещению 15172 раз, а в 2029г. уже 22758 раз. Рост числа показанной рекламы будет обусловлен двумя факторами: ростом числа пользователей и количеством размещаемой рекламы в приложении.

На рисунке 3 представлена диаграмма Ганта по данным таблицы 3.1. Начало реализации проекта запланировано на 19 мая 2020г., а завершение проекта на 19 декабря 2020г. На рисунке 3.14 представлена диаграмма Ганта.

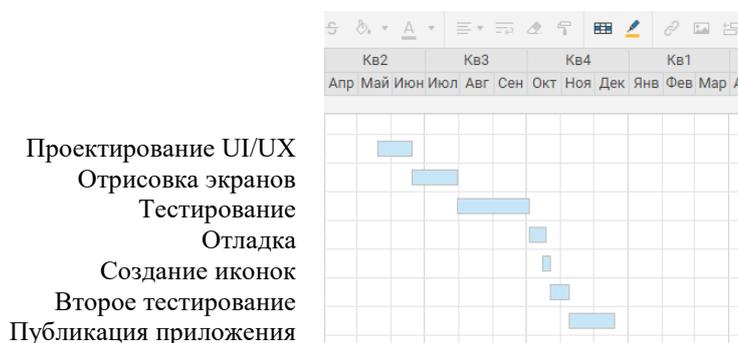


Рисунок 3. – Диаграмма Ганта по разработке мобильного приложения

Самые длительные работы запланированы на май-октябрь, то есть непосредственно на создание приложения: отрисовку экранов, создание дизайна будущего приложения, разработку кликбейковой иконки, отладки приложения. Запуск проекта в рекламу уже не такой длительные и больше зависит от креативности smm-менеджеров и выделенного бюджета на продвижение.

Начиная с первого календарного года деятельности (2021 год), планируется прибыльная работа по всем годам в прогнозируемом периоде (см. рисунок 4).

В 2021 году общая сумма чистой прибыли составит 55 673 руб., а в период с 2021 по 2029 года чистая прибыль будет увеличиваться и находиться в диапазоне от 855 673 руб. до 557 116 руб. В 2029 г прогнозируется уменьшение прибыли до 528 957 руб., что объясняется предполагаемым снижением объёма скачиваемых приложений, привлечение спонсоров и рекламы на 5% в связи с достижением максимального охвата и насыщением рынка.

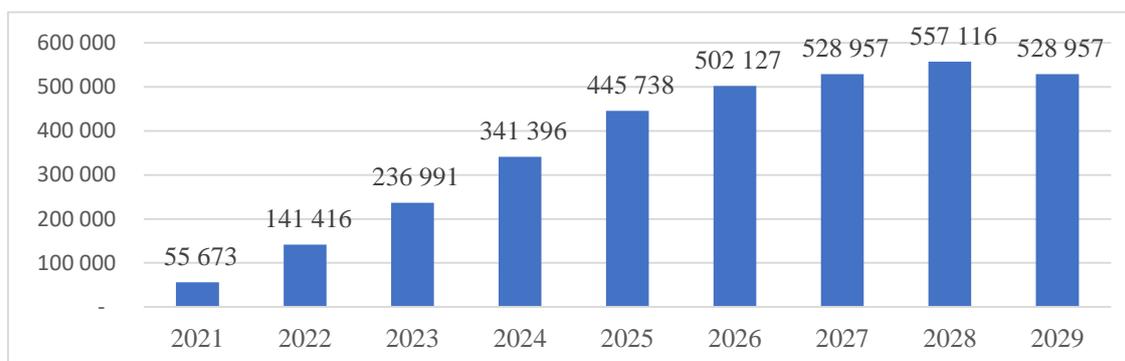


Рисунок 4. – Динамика роста чистой прибыли по годам

Эффективность проекта и его финансовая устойчивость определяются методами сравнения, дисконтирования, имитационного моделирования. Для оценки чувствительности инвестиционного проекта необходимо для начала выделить базовый вариант показателей эффективности, который представлен в таблице 4.

Таблица 4. – Показатели эффективности проекта при базовом варианте его развития

Варианты развития проекта	Значение оцениваемых показателей эффективности
Принятая ставка дисконтирования	3,39
Простой срок окупаемости проекта, лет	3,25
Динамический срок окупаемости проекта, лет	3,33
Физический износ бизнеса, исходя из средневзвешенной нормы амортизации, лет	9,50
Чистый дисконтированный доход на конец горизонта расчёта чистого потока, руб.	2 565 581
Внутренняя норма доходности, %	68,07

Все показатели развития однозначно свидетельствуют об эффективности проекта. Проект следует рекомендовать к внедрению.

Простой срок окупаемости определялся как период по месяцам, в течение которого накопительная разница между разложенными по горизонту расчёта инвестиционными затратами и чистым доходом примет положительное значение. Динамический срок окупаемости определялся аналогично, но для дисконтированных потоков инвестиционных затрат и чистого дохода.

Для расчёта чистого потока наличности путём приведения будущих потоков к начальному периоду времени принималась ставка дисконтирования, равная 3,39% годовых. Ставка дисконтирования в указанном размере принята на основе анализа уровня доходности использования различных финансовых инструментов вложений, процентных ставок представления кредитных ресурсов, существующей на момент расчётов ставки рефинансирования Национального банка Республики Беларусь.

Заключение. Таким образом, разрабатываемое приложение позволит формировать целевые группы по интересам, например, йога, фитнес, силовые тренировки и др. Также будет доступна возможность совместной аренды помещения, а этой в свою очередь замотивирует людей заниматься спортом в группах единомышленников и занятия войдут в привычку. За месяц активный пользователь сможет на 30-40% повысить частоту занятий спортом, тем самым улучшая своё здоровье. Уже через полчаса после начала тренировки наш организм начинает активно вырабатывать эндорфины, настроение резко улучшается, а уже через 4 недели регулярных тренировок легче даётся подъем по лестнице и пробежка. Окупаемость проекта 3,25 года, чистый дисконтированный доход – 2,57 млн. руб., внутренняя норма доходности – 68,07. В самом приложении будет возможна монетизация по трём направлениям – продажа платной подписки, организация спортивных мероприятий, реклама релевантных продуктов.

1. Герец, А.Ю. Организационно-экономические условия реализации мобильного приложения по поиску партнёров для занятия спортом / А.Ю. Герец // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30 (100): «Промышленность». – С. 102-105.
2. Герец, А.Ю. Оценка рынка мобильного приложения по поиску партнёров для занятия спортом / А.Ю. Герец // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30 (100): «Промышленность». – С. 106-110. Nike Training Club App. NTC Premium. Nike.com [Электронный ресурс]. – Режим доступа: <https://www.nike.com/ntc-app>. – Дата доступа: 03.09.2020.
3. Geo-location APIs | Google Maps Platform | Google Cloud [Электронный ресурс]. – Режим доступа: <https://cloud.google.com/maps-platform>. – Дата доступа: 05.09.2020.
4. Yandex.Maps API – Yandex Technologies [Электронный ресурс]. – Режим доступа: <https://tech.yandex.com/maps>. – Дата доступа: 05.09.2020.
5. Система дистанционного бизнес-образования – Инновационный проект: понятия, основные этапы создания и реализации [Электронный ресурс]. – Режим доступа: <http://www.tehnpark.bybusiness/207.html>. – Дата доступа: 02.09.2020.

УДК 004.658

ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ ДЛЯ МУЗЫКАЛЬНОГО СТРИМИНГОВОГО ВЕБ-СЕРВИСА

М.Р. ГРАДЮШКО

(Представлено: О.Н. ПЕТРОВИЧ)

В данной статье рассмотрены основные цели и принципы моделирования базы данных, на основе которых спроектировано хранилище информации музыкального стримингового веб-сервиса «Bro-Music».

Введение. На данный момент для большинства людей трудно представить свою жизнь без музыки – она окружает нас повсюду, её разнообразие огромно, а доступность легка. Музыкальный рынок как никогда актуален и широк, благодаря интернету распространение музыки стало весьма лёгкой задачей. Поэтому необходимы стриминговые веб-сервисы, которые дают возможность пользователям в полной мере насладиться музыкой, именно для одного из таких приложений и необходимо спроектировать базу данных.

Данная статья направлена на описание базы данных для разрабатываемого веб-сервиса и методов его проектирования.

Основной раздел. Разработка базы данных осуществляется с помощью моделирования данных. Цель моделирования данных состоит в обеспечении разработчика ИС концептуальной схемой базы данных в форме одной модели или нескольких локальных моделей, которые относительно легко могут быть отображены в любую систему баз данных. Наиболее распространенным средством моделирования данных являются диаграммы "сущность-связь" (ERD). С помощью ERD осуществляется детализация накопителей данных DFD – диаграммы, а также документируются информационные аспекты бизнес-системы, включая идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их связей с другими объектами (отношений).

Каждая сущность должна обладать уникальным идентификатором. Каждый экземпляр сущности должен однозначно идентифицироваться и отличаться от всех других экземпляров данного типа сущности. Каждая сущность должна обладать некоторыми свойствами:

– иметь уникальное имя; к одному и тому же имени должна всегда применяться одна и та же интерпретация; одна и та же интерпретация не может применяться к различным именам, если только они не являются псевдонимами;

– иметь один или несколько атрибутов, которые либо принадлежат сущности, либо наследуются через связь;

– иметь один или несколько атрибутов, которые однозначно идентифицируют каждый экземпляр сущности.

Каждая сущность может обладать любым количеством связей с другими сущностями модели. [1]

В ходе анализа необходимых возможностей приложения и основ проектирования базы данных были выявлены следующие сущности.

Сущность *Пользователь* описывает пользователя, имеющего доступ к сайту. Характеризуется логином, паролем и ролью. Для него необходимо выделить следующие подсущности: *Роль* (определяет права пользователя), *Посещение сайта* (необходимо для ведения статистики посещения сайта), *Голосование* (предназначено для хранения понравившихся пользователю песен), *Комментарий* (предназначен для хранения комментариев пользователя к песням).

Сущность *Песня* хранит описывает музыкальную композицию. Для нее можно выделить следующие подсущности: *Жанр песни* и *Музыкант*, принимающий участие в создании песни. Для связи музыкантов и песни была добавлена сущность *Музыканты песни*.

Сущность *Группа* является основной и хранит всю информацию о музыкальной группе. Для нее можно выделить следующие подсущности: *Репертуар* (песни, принадлежащие группе), *Состав группы* с указанием *Амплуа музыкантов*, *Тип группы*, *Страна происхождения*.

Отдельной сущностью можно выделить *Тур*, который хранит информацию о концертах группы. Для нее можно выделить следующие подсущности: *Город проведения концерта*, *Концертный зал*, *Концерт*.

Атрибуты сущности служат для уточнения, идентификации, классификации, числовой характеристики или выражение состояния сущности. Его наименование должно быть уникальным для конкретного типа сущности, но может быть одинаковым для различных типов сущности.

Теперь для каждой сущности необходимо определить атрибуты:

1. Сущность «Пользователь»: *Логин*, *Пароль*, *Роль*, *Email*.
2. Сущность «Роль»: *Название роли*.
3. Сущность «Посещение сайта»: *Пользователь*, *Дата входа*, *Дата выхода*.

4. Сущность «Голосование»: *Пользователь, Дата голоса, Песня.*
5. Сущность «Комментарий»: *Пользователь, Дата создания, Песня, Содержимое комментария.*
6. Сущность «Песня»: *Название песни, Дата выпуска, Изображение, Трек, Жанр.*
7. Сущность «Жанр песни»: *Название жанра.*
8. Сущность «Музыкант»: *Имя, Отчество, Фамилия, Дата рождения.*
9. Сущность «Музыканты песни»: *Песня, Музыкант, Статус музыканта.*
10. Сущность «Группа»: *Название группы, Дата начала карьеры, Дата окончания карьеры, Изображение, Тип группы, Страна происхождения.*
11. Сущность «Репертуар»: *Группа, Песня, Дата начала, Дата окончания.*
12. Сущность «Состав группы»: *Группа, Музыкант, Дата начала, Дата окончания, Амплуа музыканта.*
13. Сущность «Амплуа музыканта»: *Название амплуа.*
14. Сущность «Тип группы»: *Название типа.*
15. Сущность «Страна происхождения»: *Название страны.*
16. Сущность «Тур»: *Название тура, Дата начала, Дата окончания, Группа.*
17. Сущность «Город»: *Название тура, Численность населения, Страна.*
18. Сущность «Концертный зал»: *Название зала, Город.*
19. Сущность «Концерт»: *Тур группы, Концертный зал, Дата начала, Дата окончания, Средняя цена билетов.*

Для каждой сущности выделим свой идентифицирующий атрибут, что представлено в таблице 1.

Таблица 1. – Первичные и внешние ключи отношений.

№	Название таблицы	Первичный ключ	Внешние ключи
1	Users	id	roleId
2	Logging	id	userId
3	UserRoles	id	отсутствуют
4	Voices	id	loggingId songId
5	Comments	id	loggingId songId
6	Repertoires	id	groupId songId
7	Songs	id	genId
8	Genres	id	отсутствуют
9	Musicians	id	отсутствуют
10	MapSongMusician	id	songId musicianId
11	Groups	id	groupTypeId countryId
12	GroupTypes	id	отсутствуют
13	CompositionGroup	id	musicianId groupId professionId
14	Professions	id	отсутствуют
15	Tours	id	groupId
16	Countries	id	отсутствуют

Одно из важнейших достоинств реляционных баз данных состоит в том, что можно хранить логически сгруппированные данные в разных таблицах и задавать связи между ними, объединяя их в единую базу. Такая организация данных позволяет уменьшить избыточность хранимых данных, упрощает их ввод и организацию запросов и отчетов.

Существует три разновидности связей между таблицами базы данных:

1. «один-ко-многим». В типе связей один ко многим одной записи первой таблицы соответствует несколько записей в другой таблице.

2. «один-к-одному». Связь один к одному образуется, когда ключевой столбец (идентификатор) присутствует в другой таблице, в которой тоже является ключом либо свойствами столбца задана его уникальность (одно и тоже значение не может повторяться в разных строках).

3. «многие-ко-многим». Если нескольким записям из одной таблицы соответствует несколько записей из другой таблицы, то такая связь называется «многие-ко-многим» и организовывается посредством связывающей таблицы [2].

Связи выполняют более важную роль, чем просто информация размещения данных по таблицам. Прежде всего они требуются для поддержания целостности баз данных.

Правильно настроив связи, можно быть уверенным, что ничего не потеряется, поэтому необходимо ответственно отнестись к этому вопросу. Для реализации информационной системы музыкального сайта необходимо установить связи между таблицами ее базы данных, что представлено на рисунке 1.

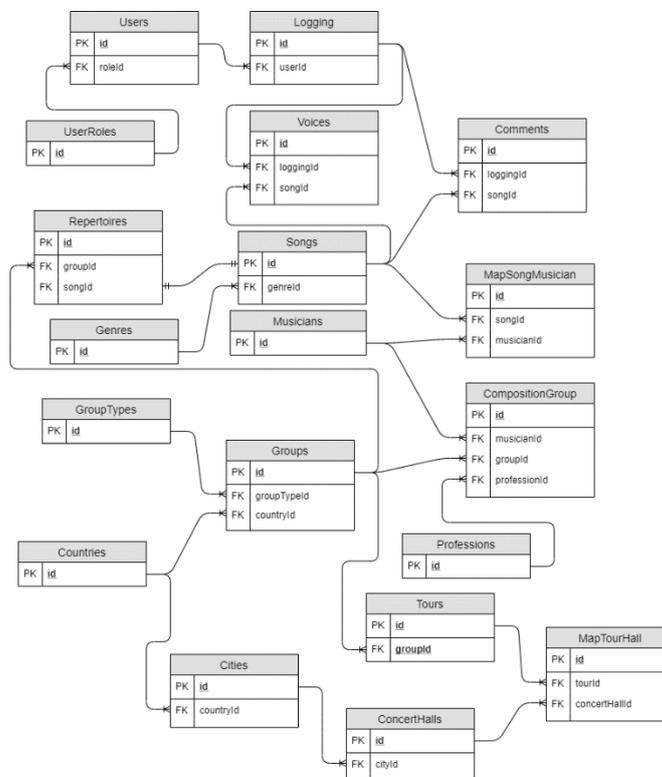


Рисунок 1 – Связи между сущностями базы данных

Заключение. В данной статье были рассмотрены основные цели и принципы моделирования базы данных для стримингового веб-сервиса «Bro-Music». Также были рассмотрены основные сущности спроектированной базы данных.

ЛИТЕРАТУРА

1. НОУ ИНТУИТ [Электронный ресурс]. – Режим доступа: <https://intuit.ru/studies/courses/2195/55/lecture/1636>. – Дата доступа: 20.09.2020.
2. Базы данных [Электронный ресурс]. – Режим доступа: <http://informatic.ugatu.ac.ru/lib/office/Proekt.htm>. – Дата доступа: 21.09.2020.

УДК 004.622

РАЗРАБОТКА МУЗЫКАЛЬНОГО СТРИМИНГОВОГО ВЕБ-СЕРВИСА

М.Р. ГРАДЮШКО*(Представлено: О.Н. ПЕТРОВИЧ)*

В данной статье объектом разработки представлен музыкальный стриминговый веб-сервис «Bro-Music». Рассмотрены основные принципы построения серверной части и графического интерфейса музыкального веб-сервиса.

Введение. Музыкальные сервисы пользуются большим спросом, нужно лишь подкупить пользователя, например, привлекательным внешним видом своего сервиса, уникальными возможностями или богатством музыкальной библиотеки. Тема, конечно, не новая, но совсем не утратила актуальности с течением времени.

Несмотря на довольно большое количество различных веб-сервисов, предоставляющих широкий спектр возможностей, связанных с прослушиванием музыки, все они обладают своими недостатками. К примеру, большинство таких сервисов платные, используют рекламу, не поддерживают оффлайн режим, не оповещают пользователя о выходе новых треков и альбомов любимых исполнителей или же не дают пользователю никакой интерактивности. Именно поэтому можно с уверенностью сказать, что даже сегодня многое можно улучшить в мире музыкальных сайтов.

Поэтому разработка музыкального стримингового веб-сервиса является актуальной.

Основной раздел. Разработанный музыкальный стриминговый веб-сервис предоставляет пользователям возможность прослушивать понравившуюся музыку. Программный продукт представляет собой веб-приложение, имеющее следующий функционал:

1. Прослушивание, комментирование, скачивание музыки.
2. Рассылка оповещений.
3. Предоставление информации о группах, треках, альбомах и т.д.
4. Возможность создания плейлистов.
5. Аутентификация и авторизация.

Для разработки такого веб-сервиса в качестве модульной платформы служит ASP .NET Core 2.2, поскольку она позволяет небольшим проектам и стартапам получить все преимущества платформы корпоративного уровня, при этом предоставляя удобные средства разработки, а также недорогую инфраструктуру. В будущем же неизбежен приход платформы и на крупный корпоративный рынок.

Преимущества платформы ASP.NET Core:

- единое решение для создания пользовательского веб-интерфейса и веб-API;
- страницы Razor делают создание кодов сценариев для страниц проще и эффективнее;
- Blazor позволяет использовать в браузере язык C# вместе с JavaScript. совместное использование серверной и клиентской логики приложений, написанных с помощью .NET;
- возможность разработки и запуска в ОС Windows, macOS и Linux;
- открытый исходный код и ориентация на сообщество;
- встроенное введение зависимостей;
- упрощенный высокопроизводительный модульный конвейер HTTP-запросов [1].

В качестве СУБД был выбран MySQL. Помимо универсальности и распространенности СУБД MySQL обладает целым комплексом важных преимуществ, которые делают её пригодной для решения поставленных задач. Данная система управления базами данных изначально создана таким образом, что множество встроенных функций безопасности в ней работают по умолчанию, что упрощает работу разработчику.

Взаимодействие с базой данных осуществляется с помощью современной технологии Entity Framework. Entity Framework предоставляет возможность работы с базами данных через объектно-ориентированный код C#. Этот подход предоставляет ряд существенных преимуществ: вам не нужно беспокоиться о коде доступа к данным, вам не нужно знать деталей работы СУБД SQL Server и синтаксиса языка запросов T-SQL, вместо этого вы работаете с таблицами базы данных как с классами C#, с полями этих таблиц как со свойствами классов, а синтаксис SQL запросов, который в ADO.NET раньше нужно было вставлять в код C# в виде команд, заменен на более удобный подход с LINQ. Entity Framework берет на себя обязанности по преобразованию кода C# в SQL-инструкции [2].

При работе с Entity Framework вам предоставляются огромные возможности по созданию модели базы данных с помощью интегрированной среды разработки (IDE) Visual Studio. Пример взаимодействия с базой данных представлен в листинге 1.

Листинг 1 – Получение самых популярных песен

```
1. model.AllSongsWithVoices = db.Voices
2. .Where(v => v.SongId != null)
3. .GroupBy(v => v.SongId)
4. .Select(g => new SongVoices { Song = allSongs[(int)g.Key], CountVoices =
g.Count() })
5. .OrderByDescending(v => v.CountVoices)
6. .Take(10);
```

Так как программа, работающая с базой данных, сделана в виде веб-приложения, все её формы выполнены в виде веб-страниц.

Административная часть сайта, где происходит полный контроль над содержимым веб-приложения, обладает теми же стилистическими особенностями, что и часть клиентская. Для удобства использования было принято отображать информацию в табличном виде. Администратор может добавлять, изменять, редактировать информацию и связывать данные из разных таблиц между собой.

Страницы сверстаны на языках HTML, CSS и JS, что позволило придать им понятный и привлекательный вид. Кроме того, стиль придаёт специальный фреймворк Bootstrap, позволяющий быстро стилизовать приложение, наполнив его красивыми элементами и применяя определенные заготовленные стили.

При проектировании приложения были учтены все возможные случаи некорректной работы программы, поэтому большинство нештатных ситуаций сопровождается оповещениями с описанием проблемы.

Ниже представлен скрипт обработки воспроизведения песни из листинга треков.

Листинг 2 – Воспроизведение песни из листинга

```
1. playlist.find('.song').click(function (e) {
2. e.preventDefault();
3. link = $(this);
4. if (link.attr("song-id") == song.id) {
5. playOrPauseSong();
6. } else {
7. $("#playlist.song").attr("src", "/img/listplay.png");
8. currentSong = link.parent().parent().index();
9. playSong();
11. $("#play img").attr("src", "/img/pause.png");
12. link.attr("src", "/img/list-pause.png");
13. }
14. });
```

Также клиенту доступен плеер для более удобного прослушивания музыкальных композиций. В нем доступен ряд возможностей, такие как переключение песен, воспроизведение и приостановка треков, регулирование уровня громкости, перемотка композиции. Скриншот плеера представлен на рисунке 1.

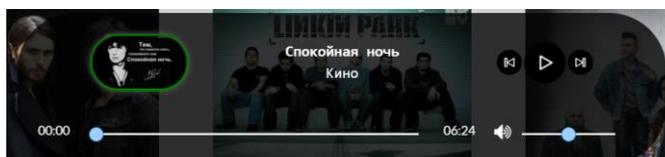


Рисунок 1. – Плеер

Заключение. В данной статье были рассмотрены средства для разработки веб-приложений, при помощи которых был реализован стриминговый веб-сервис, соответствующий современным требованиям.

ЛИТЕРАТУРА

1. Документация по ASP.NET Core [Электронный ресурс]. Режим доступа: <https://docs.microsoft.com/ru-ru/ASPNET/>. – Дата доступа: 22.09.2020.
2. ProfessorWeb [Электронный ресурс]. Режим доступа: <https://professorweb.ru/my/entityframework/6/level1/>. – Дата доступа: 23.09.2020.

**TEST-DRIVEN DEVELOPMENT – РАЗРАБОТКА ЧЕРЕЗ ТЕСТИРОВАНИЕ:
ПРЕИМУЩЕСТВА И НЕДОСТАТКИ****Н.В. ГУРТОВЕНКО***(Представлено: канд. физ.-мат. наук, доц. О.В. ГОЛУБЕВА)*

В статье рассматривается метод разработки TDD. Анализируются его преимущества и недостатки.

Введение. Для создания успешного ИТ-проекта важным шагом является выбор методологии разработки, которая содержит свой подход к процессу создания программ в виде шагов, задач, действий. Таких методов существует немало, каждый имеет свои достоинства и недостатки, и выбор главным образом зависит от поставленной задачи. Один из них – метод Test-Driven Development – разработка через тестирование.

Основной раздел. TDD представляет собой процесс разработки программного обеспечения, требующий от программиста написать автоматизированный тестовый пример для требуемой функциональности (первоначально неисправного), затем добавить минимальный код, необходимый для прохождения теста и, наконец, реорганизовать код и убедиться, что автоматизированные тесты до сих пор проходят. В традиционных методах, вначале пишется код, а затем разрабатываются тестовые примеры для проверки кода. В TDD, тестовые примеры (как правило, называемые юнит-тесты), чтобы удовлетворить требование пишутся до того, как реализация началась.

TDD состоит из цикла «red–green–refactor».

Этап Red. Пишется тест, который представляет собой необходимое поведение системы, затем создается метод-заглушка, чтобы можно было быстро собрать проект. Тест компилируется, но не возвращает нужный результат. Появляется потребность в написании программы.

Этап Green. Пишется минимальный код, который бы позволил тесту выполниться верно. В действительности это значит, что пишется код без структуры, без дизайна, без каких-либо шаблонов проектирования. Теперь появляется потребность придать коду «красивый вид».

Этап рефакторинга. Изменяется внешняя структура кода, без изменения его внешнего поведения. Это разделение на методы, добавление элементов шаблона проектирования, создание дополнительных классов и т. д. Последовательность этапов цикла очень важна. Принцип метода «Test First» подразумевает, что пишется только код, абсолютно необходимый для успешного прохождения всех тестов.

Преимущества методологии:

- Уменьшение времени на отладку. Если не использовать методологию TDD, то потребуется меньше чистого времени на написание кода. Однако потраченное на тесты время окупается при отладке кода и отлове ошибок – а эта часть процесса разработки присутствует всегда, потому что почти невозможно написать код без единой ошибки и это нормально. Благодаря TDD не нужно гадать, где находится ошибка, протестированный код проще поддерживать.

- Удобство изменяемости. Покрытие тестами помогает избежать ситуации, когда при изменении кода в одном месте возникает ошибка в совсем другой части кода. В итоге это позволяет безопасно проводить изменение или рефакторинг кода, потому что если что-то пойдет не так, то тесты сразу обработают нештатную ситуацию.

- Тесты, особенно если они написаны в выразительном стиле, могут служить в качестве технической документации проекта, в которой описывается то, как код должен работать. Такая документация очень полезна для новых разработчиков в проекте, которые для того, чтобы включиться в работу, вначале должны разобраться с кодом. Тест в этом случае служит конкретным примером использования кода. Полное покрытие кода тестами дает огромную практическую документацию кода, отражающую реальное состояние системы.

- Модульность. Один из принципов разработки через тестирование предусматривает, что каждая функция выполняет определенную, небольшую часть работы, потому что просто невозможно протестировать «всемогущий» метод, который выполняет десяток функций в нескольких потоках. Все это вынуждает разделять программу на модули, чтобы удалось протестировать все ветви кода.

- Полноценные тесты. Существует большая разница между написанием обычных тестов и тестов по методу TDD. Когда тесты пишутся после реализации есть вероятность получить неполноценные тесты из-за того, что могут быть учтены не все сценарии работы метода. Например, может считаться, что написанный метод работает, так как он проходит тест, но на самом деле может оказаться, что в методе покрыты тестами не все условия. Использование TDD позволяет не случиться такому, потому что условие не может появиться в коде без теста.

- TDD является лучшим способом, чтобы гарантировать, что тесты на самом деле охватывают все требования, а не только код. Кроме того, TDD помогает не попасть в «ловушку»: добавление функциональности, которой клиенту на самом деле не нужно, но было бы неплохо иметь.

Недостатки:

- Возможность применить TDD имеется не всегда. Существуют задачи, которые невозможно решить только при помощи тестов. Например, это задачи в области безопасности данных и взаимодействия между процессами. Иногда бывает, что сложно сразу представить, как будет выглядеть работа модуля. Кроме того, невозможно решить с помощью TDD разработку баз данных, компиляторов и интерпретаторов языков программирования, невозможно автоматизировать тестирование графического интерфейса и распределенных объектов.

- Начальные требования не всегда могут быть понятны и правильно интерпретированы. В итоге можно получить ошибку в тесте, в коде и в понимании. Опасность ситуации заключается в том, что с виду кажется, что все работает правильно, ведь тесты проходят зеленый этап. На всем этом можно потерять массу времени.

- Необходимость поддержки тестов. База кода при стопроцентном покрытии тестами увеличивается почти в два раза. И кроме того всю эту базу необходимо документировать, поддерживать и проводить рефакторинг.

Заключение. Не смотря на существование большого количества мощных инструментов разработки, программирование по-прежнему остается сложной работой. Методология разработки через тестирование позволяет разделить работу на много маленьких частей и сконцентрировать внимание программиста на единственной задаче. Таким образом, методика TDD позволяет разделить процесс разработки на элементарные режимы, избавляя от монотонности, предлагая быстро переключаться между этими режимами, что повышает эффективность разработки.

ЛИТЕРАТУРА

1. Imprium.ru [Электронный ресурс]. – Режим доступа: <https://imprium.ru/articles/test-based-development>. – Дата доступа: 20.09.2020.
2. Proglib.io. Сайт о программировании [Электронный ресурс]. – Режим доступа: <https://proglib.io/p/test-driven-development/>. – Дата доступа: 20.09.2020.
3. Itvdn.com. Сайт о программировании [Электронный ресурс]. – Режим доступа: <https://itvdn.com/ru/video/test-driven-development>. – Дата доступа: 20.09.2020.

УДК 004.054

ЖИЗНЕННЫЙ ЦИКЛ ТЕСТИРОВАНИЯ ПО. ПРЕИМУЩЕСТВА И ШАГИ**Н.В. ГУРТОВЕНКО***(Представлено: канд. физ.-мат. наук, доц. О.В. ГОЛУБЕВА)*

В статье рассматривается жизненный цикл, преимущества и шаги тестирования при разработке программного обеспечения. Проведен анализ шагов тестирования, важности тестирования на каждом шаге разработки программного обеспечения.

Введение. Тестирование ПО представляет из себя процесс проверки того, насколько поведение реального ПО соответствует требованиям и ожиданиям. Проводимые для этого тесты основаны на задокументированных требованиях заказчика.

Основной раздел. В первую очередь стоит отметить, что процесс тестирования ПО тесно связан непосредственно с процессом разработки. Жизненный цикл разработки состоит из следующих этапов:

1. Анализ требований;
2. Дизайн;
3. Разработка;
4. Тестирование и дебаггинг;
5. Эксплуатация и поддержка.

Как показано в списке выше, тестирование должно проводиться на четвертом шаге жизненного цикла разработки. Но обычно в случае, если главной целью является получить высококачественное ПО и минимизировать затраты на исправление багов, тестирование может проводиться уже на стадии анализа требований. Чем раньше начинается проводиться тестирование, тем лучше будет результат разработки.

Ниже детально рассмотрим какие преимущества может принести проведение тестирования на каждом этапе процесса разработки, начиная с самого первого.

Первый этап. Анализ требований.

Начнем с первого этапа жизненного цикла разработки: анализ требований. Требования к конечному продукту обычно формулируются заказчиком или менеджером проекта. Эти требования могут быть как функциональными, так и нефункциональными. Они формируются в процессе общения с заказчиком или анализа стандартов и нормативной документации. Зачем же необходимо проводить тесты ПО на этом этапе жизненного цикла, и какие преимущества это может дать?

Может возникнуть ситуация, при которой имеющиеся требования не были протестированы, но были использованы на этапе дизайна и разработки. Только после того, как разработка закончена, требования и сам продукт направляются в отдел QA. Как было сказано ранее, в процессе тестирования проверяется, соответствует ли текущее поведение продукта заявленным требованиям. А это значит, что отдел QA может обнаружить ошибки не только в самом продукте, но также и в документации. Следовательно, в этом случае исправление ошибок обойдется гораздо дороже в сравнении с подходом, который предусматривает включение тестов в самые ранние этапы жизненного цикла ПО, такие как фаза анализа требований. Тщательным образом проанализировав требования, разработчик получит информацию, которая поможет оптимизировать процесс работы над проектом с самых первых дней.

Второй этап. Процесс дизайна.

Следующим этапом жизненного цикла разработки ПО является процесс дизайна. Как и тестирование требований на стадии анализа требований, этот этап подразумевает проверку уже созданных прототипов и мокапов на предмет их корректности и соответствия ожиданиям заказчика. Более того, проверка удобства в использовании также должна быть проведена на этом этапе. Также следует начать создание тестовой документации для данного проекта. Эта задача включает в себя подготовку плана тестирования, тест-кейсов, юзкейсов, а также другой документации по требованию заказчика.

Процесс тестирования ПО на этом этапе обеспечивает способность проникновения в суть продукта и понимание ее соответствия требованиям. Важным является точное понимание задач, стоящих перед отделом QA на протяжении всего жизненного цикла разработки.

Третий этап. Разработка.

В течении этапа разработки важно провести модульное, интеграционное и системное тестирование. В самом начале этого шага разработки проводится модульное тестирование. Этот процесс представляет собой проверку отдельного модуля системы или функционала. Интеграционное тестирование проводится после того, как несколько модулей объединены вместе как отдельная часть приложения. В дальнейшем в процессе разработки все больше и больше модулей объединяются воедино. После того, как разработка

закончена, наступает время подготовки к системному тестированию. Эта стадия жизненного цикла разработки ПО подразумевает общий тест системы на предмет интеграции ее компонентов. Это значит, что в случае, если система состоит из различных модулей, разработчик должен проверить, насколько хорошо или насколько плохо каждый из них работает внутри системы. Более того, на этом этапе важно произвести тестирование пользовательского интерфейса.

Четвертый этап. Процесс тестирования и дебаггинга.

На этом шагу необходимо провести тесты независимо от того, проводились ли они на предыдущих этапах. Должны быть проведены полное функциональное тестирование и тестирование пользовательских интерфейсов, а все обнаруженные дефекты должны быть задокументированы в системе баг-трекинга. Помимо этого, применяется регрессионное тестирование. После завершения дебаггинга предоставляется оценка общего качества продукта. После завершения последнего теста процесс тестирования ПО считается законченным.

Пятый этап. Эксплуатация и поддержка.

Даже после достижения стадии релиза продукта, остается необходимость в тестировании, проводимом на этапе эксплуатации и поддержки. Разные пользователи могут работать в абсолютно разных окружениях. Поэтому всегда возможно, что новые ошибки, которые не были выявлены ранее дадут о себе знать. Более того, пользователи могут использовать ПО изначально непредвиденным способом. Это, в свою очередь, может вызвать некоторые непредвиденные проблемы. В таком случае потребуется вмешательство отдела QA.

Заключение. Очевидно, что процесс управления тестированием ПО затрагивает все этапы жизненного цикла разработки. Он подразумевает сравнение действительного состояния продукта и того состояния, которое было запланировано и задокументировано в плане тестирования продукта. Процесс тестирования, анализа и мониторинга помогает спланировать и изменить последующие задачи наилучшим путем. Жизненный цикл тестирования ПО является процессом, которого нельзя избежать. Он непрерывен, продолжителен и требует наличия команды QA, достаточно опытной для того, чтобы произвести полный цикл тестирования. Эта неотъемлемая часть современного процесса разработки ПО помогает заказчику, команде разработчиков, а также конечному пользователю получить продукт высокого качества.

ЛИТЕРАТУРА

4. Habr [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/430128/>. Дата доступа: 20.09.2020.
5. Software-testing. Сайт о тестировании [Электронный ресурс]. Режим доступа: <https://software-testing.ru/library/testing/general-testing/78>. Дата доступа: 20.09.2020.
6. Itsecforu [Электронный ресурс]. Режим доступа: <https://itsecforu.ru/2019/09/17/лучшие-практики-для-разработки-и-те>. Дата доступа: 20.09.2020.

УДК 004.514

**ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА ВЕБ-ПРИЛОЖЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ
ДЛЯ ПОЧТОВЫХ ОТДЕЛЕНИЙ****П.В. ДЕРЕВЯГА****(Представлено: О.Н. ПЕТРОВИЧ)**

В данной статье описано создание графического интерфейса для типичного веб-приложения с автоматизированной информационной системой. Данный интерфейс подразумевает как простое чтение информации, так и более сложные взаимодействия вроде администраторской части приложения, где можно будет напрямую добавлять и/или править записи в базе данных. В статье приведены и обоснованы выбранные технологии для создания интерфейса.

Введение. На сегодняшний день веб-приложения – это, пожалуй, самый распространенный вид приложений. Практически каждая уважающая себя контора, учреждение или даже просто музыкальная группа имеют свой сайт. Не остаются в стороне и государственные структуры, одной из которых является почта. Для грамотного функционирования почтовой системы крайне важен учёт, а также информирование клиентов.

С такими задачами прекрасно может справиться веб-приложение с автоматизированной информационной системой. А одним из важнейших свойств такого приложения будет графический интерфейс. Простота, понятность, функциональность и удобство должны помочь как сотрудникам почты, так и обычным пользователям.

Основная часть. В качестве технологии для разработки веб-приложения была выбрана технология ASP.NET Core в связке с MVC. Такой подход позволяет разрабатывать совместно в одном проекте как графический интерфейс, так и серверную часть приложения.

Так как программа, работающая с базой данных, сделана в виде веб-приложения, все её формы выполнены в виде веб-страниц. Для разметки страниц использован специальный формат веб-страниц в ASP.NET Core – страницы Razor (cshtml). Такие страницы позволяют использовать стандартный html-код одновременно со специальными атрибутами ASP и вставками кода на C#.

Razor - это интеллектуальный обработчик программного кода динамических веб-страниц ASP.NET. Имеет простой, интуитивно понятный, синтаксис встраивания программного кода в веб-страницы. Razor – механизм визуализации (view engine) поддерживаемый .NET Framework, .NET Core в рамках ASP.NET и специально предназначен для создания веб-приложений. [1]

Внешний вид страниц выполнен в стремлении к максимальной понятности интерфейса для пользователей, а также его привлекательности. Все страницы реализованы при помощи специальных классов и шаблонов фреймворка Bootstrap, который значительно облегчает разработку интерфейса веб-приложений при этом позволяя создавать по-настоящему красивые и стильные страницы.

Bootstrap (также известен как Twitter Bootstrap) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения. [2]

Bootstrap является, по сути, ярким лидером среди фреймворков для создания графического интерфейса сайтов. С его помощью можно быстро создать стильный и привлекательный сайт, всего лишь применив готовые шаблоны.

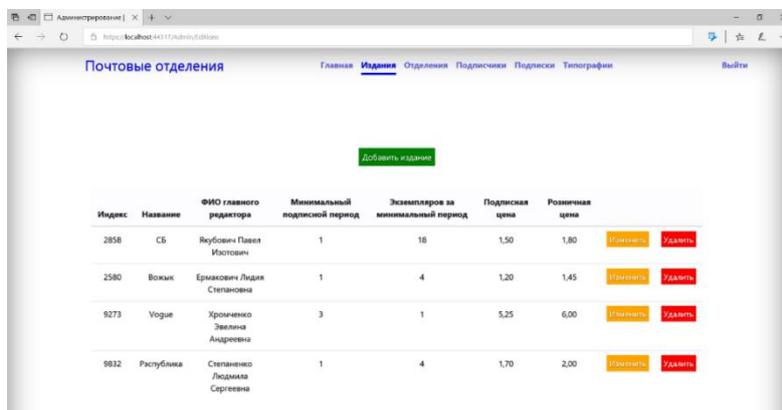
Тип приложения подразумевает под собой наличие как клиентской части, так администраторской, предназначенной для управления контентом. При этом если графический интерфейс клиентской части призван в приятной глазу форме представлять пользователю данные и позволять производить некоторые манипуляции (вход/выход в аккаунт, оформление подписки, просмотр личной статистики), то администратору предоставлен полнофункциональный инструмент для работы с данными. Администратор получил интерфейс, благодаря которому он сможет максимально эффективно добавлять информацию на сайт (в базу данных), изменять и удалять её.

Для комфортного восприятия информации пользователями отображение данных из базы оформлено в виде таблиц. При этом таблицы занимают практически всю ширину страницы, дабы читать записи было максимально легко.

Для администратора приложения существуют специальные страницы, попасть на которые можно только пройдя авторизацию. Интерфейс страниц полностью повторяет обычную часть приложения, за исключением добавления кнопок для добавления, редактирования и удаления записей базы данных. Кроме того, в соответствие каждой из таблиц были реализованы специальные страницы с формами, заполнив которые администратор сможет добавлять или изменять записи. Кроме того, добавление новых записей

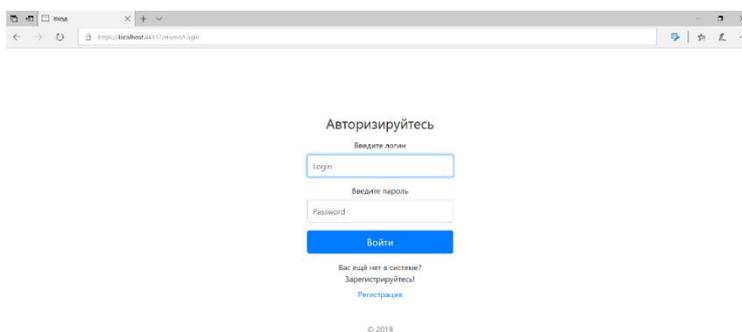
происходит на отдельных страницах, представляющих собой форму с полями для заполнения и кнопку подтверждения заполненных данных.

На рисунках 1 и 2 представлены примеры получившегося графического интерфейса, а точнее страница авторизации и управления списком изданий для администратора.



Индекс	Название	ФИО главного редактора	Минимальный подписной период	Экземпляров за минимальный период	Подписная цена	Розничная цена		
2808	СБ	Якубович Павел Иосифович	1	18	1,50	1,80	Изменить	Удалить
2580	Вожжик	Ермакович Людмила Степановна	1	4	1,20	1,45	Изменить	Удалить
9273	Уголе	Хроменко Звельна Андреевна	3	1	5,25	6,00	Изменить	Удалить
9832	Республика	Степанович Людмила Сергеевна	1	4	1,70	2,00	Изменить	Удалить

Рисунок 1. – Страница управления списком изданий



Авторизируйтесь

Введите логин

Введите пароль

Войти

Вас ещё нет в системе?
Зарегистрируйтесь!

[Регистрация](#)

© 2019

Рисунок 2. – Страница авторизации

Заключение. В результате был разработан графический пользовательский интерфейс для веб-приложения информационной системы почтовых отделений, отвечающий поставленным требованиям как по функциональности, так и по понятности, простоте и привлекательности. Также данный интерфейс был протестирован на предмет некорректной или нестабильной работы и проблем выявлено не было.

ЛИТЕРАТУРА

1. Документация по ASP.NET Core [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/ASPNET>. – Дата доступа: 24.09.2020.
2. Википедия [Электронный ресурс]. Режим доступа: <https://wikipedia.org/>. Дата доступа: 24.09.2020.

УДК 004.624

РАЗРАБОТКА ВЕБ-АР И БАЗЫ ДАННЫХ ВЕБ-ПРИЛОЖЕНИЯ
ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ ПОЧТОВЫХ ОТДЕЛЕНИЙ

П.В. ДЕРЕВЯГА

(Представлено: О.Н. ПЕТРОВИЧ)

В данной статье описывается практическая часть разработки серверной части и базы данных веб-приложения информационной системы для почтовых отделений. Приведен и обоснован выбор технологий для разработки, а также особенности системы.

Введение. Для упрощения жизни практически во всех сферах деятельности стало очень выгодно переходить на автоматизированные информационные системы.

Автоматизированная информационная система (АИС) – это комплекс, который включает компьютерное и коммуникационное оборудование, программное обеспечение, лингвистические средства, информационные ресурсы, предназначенных для сбора, подготовки, хранения, обработки и предоставления информации, а также системный персонал, обеспечивающий поддержку динамической информационной модели некоторой части реального мира для удовлетворения информационных потребностей пользователей и для принятия решений.

Основа любой информационной системы – база данных.

Базы данных уже давно стали основой не только любого солидного приложения, но и любой информационной системы. Они позволяют упорядочить хранение данных о предметной области в одной структуре, предоставляют широкие возможности для взаимодействия с этой информацией. Кроме того, грамотно спроектированная и разработанная база данных – залог успешного приложения. Поэтому сегодня практически в каждой сфере человеческой деятельности необходимы эти самые базы данных.

Не исключение и сфера почтовых услуг – информация о газетах и журналах, подписчиках и подписках, работниках и так далее – все это должно быть сохранено и упорядочено. База данных поможет упростить доступ к информации, организовать её поиск, а также значительно сократить затраты времени и сил на ведение документации и архивов. Кроме того, база данных для почтовых отделений при помощи её возможностей позволит частично автоматизировать некоторые процессы, например, составление квитанций, подсчет финансов, составление накладных и заказов на печать.

Для обычного пользователя такая информационная система позволит просматривать общую информацию об изданиях, отделениях и т.д., для заведующего почтовым отделением – это возможности по учету и анализу ситуации, найму и увольнению сотрудников.

Основная часть. В качестве СУБД для разработки базы данных была выбрана PostgreSQL, которая имеет ряд преимуществ: бесплатное ПО с открытым исходным кодом. Эта СУБД является очень мощной системой; большое количество дополнений - существует очень много дополнений, позволяющих обрабатывать данные для этой СУБД и управлять ими; расширения - существует возможность расширения функционала за счет сохранения своих процедур; объектность – PostgreSQL это не только реляционная СУБД, но также и объектно-ориентированная с поддержкой наследования и много другого.

Реляционная модель представляет собой базу данных в виде множества взаимосвязанных отношений (таблиц), которые используются для хранения информации об объектах, представленных в базе данных.

Исходя из функциональных требований к приложению был определен набор таблиц, описывающий заданную предметную область и позволяющий хранить всю необходимую информацию. Список таблиц представлен в таблице 1.

Таблица 1 – таблицы базы данных приложения

Имя таблицы	Описание
stations	Описывает отделение почтовой связи
employees	Работники почты
districts	Участки, на которые поделен город
subscribers	Подписчики
editions	Печатные издания
subscriptions	Подписки
receipts	Чеки за подписки
waybills	Накладная

Окончание таблицы 2

1	2
typographies	Типографии
print_orders	Заказы на печать для типографий
roles	Роли пользователей
users	Пользователи

Также стоит упомянуть, что обязательная часть любой автоматизированной системы с базой данных являются триггеры. Триггер – это своего рода скрипт, выполняющий SQL-код при наступлении какого-либо события, например, добавление новой записи в таблицу, изменение или удаление записи. Для информационной системы почтовых отделений можно выделить несколько процессов, нуждающихся в подобной автоматизации. Для начала, клиенту необходимо выписать квитанцию после успешного оформления подписки, а также сохранить эту квитанцию в базе данных. Тут как нельзя кстати придется триггер, который при добавлении в базу новой подписки автоматически сформирует квитанцию и добавит в отдельную таблицу. Ещё один похожий процесс – формирование заказов на печать. Заказы одной газеты для одного отделения могут дробиться на несколько штук для разных типографий, поэтому необходим триггер, собирающий количество заказанных газет для каждого почтового отделения. И последний пример – увольнение сотрудника. Если это почтальон – нужно перераспределить участки между оставшимися почтальонами, дабы ни один из участков не остался без обслуживания.

Разработка веб-API данного веб-приложения велась в одном проекте с графическим интерфейсом, такой особенностью обладает выбранная технология разработки – ASP.NET Core 3 MVC.

Платформа ASP.NET Core представляет технологию от компании Microsoft, предназначенную для создания различного рода веб-приложений: от небольших веб-сайтов до крупных веб-порталов и веб-сервисов [1].

Преимущества платформы ASP.NET Core:

- Единое решение для создания пользовательского веб-интерфейса и веб-API.
- Страницы Razor делают создание кодов сценариев для страниц проще и эффективнее.
- Blazor позволяет использовать в браузере язык C# вместе с JavaScript. совместное использование серверной и клиентской логики приложений, написанных с помощью .NET.
- Возможность разработки и запуска в ОС Windows, macOS и Linux.
- Открытый исходный код и ориентация на сообщество.
- Интеграция современных клиентских платформ и рабочих процессов разработки..
- Облачная система конфигурации на основе среды.
- Встроенное введение зависимостей.
- Упрощенный высокопроизводительный модульный конвейер HTTP-запросов [2].

Веб-API при таком подходе будет иметь простую структуру – методы-обработчики будут вызываться при выполнении пользователем GET или POST запросов на соответствующий URL. В самом методе реализуются необходимые действия. Если это действие связано с базой данных, то для взаимодействия с этой самой базой будет использоваться Entity Framework, который весьма удобным образом свяжет API и базу данных.

Заключение. В результате данной разработки были получены такие части веб-приложения с автоматизированной информационной системой как веб-API (или серверная часть) и база данных, служащие для обработки, хранения и доставки пользователям актуальной информации по интересующим их вопросам.

ЛИТЕРАТУРА

1. Сайт о программировании [Электронный ресурс]. – Режим доступа: <https://metanit.com>. – Дата доступа: 24.09.2020.
2. Документация по ASP.NET Core [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/ASPNET>. – Дата доступа: 24.09.2020.

ПРОБЛЕМЫ ПРИМЕНЕНИЯ МАШИННОГО ОБУЧЕНИЯ

П.В. ДМИТРИЧЕНКО

(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)

Изучены наиболее важные технические проблемы, с которыми сталкивается машинное обучение как прикладная наука при внедрении машинного обучения в существующие информационные системы.

Введение. Машинное обучение – обширный подраздел искусственного интеллекта, изучающий методы построения алгоритмов, способных обучаться. Различают два типа обучения. Обучение по прецедентам, или индуктивное обучение, основано на выявлении общих закономерностей по частным эмпирическим данным. Дедуктивное обучение предполагает формализацию знаний экспертов и их перенос в компьютер в виде базы знаний. Дедуктивное обучение принято относить к области экспертных систем, поэтому термины машинное обучение и обучение по прецедентам можно считать синонимами.

Машинное обучение находится на стыке математической статистики, методов оптимизации и классических математических дисциплин, но имеет также и собственную специфику, связанную с проблемами вычислительной эффективности и переобучения. Многие методы индуктивного обучения разрабатывались как альтернатива классическим статистическим подходам. Многие методы тесно связаны с извлечением информации и интеллектуальным анализом данных (Data Mining).

При разработке информационных продуктов, использующих технологии машинного обучения, возникают проблемы, большая часть которых нетипична для экспертных систем или их аналогов, которые машинное обучение и призвано заменить.

Основная часть. Методология сбора данных – представляет собой все методы, связанные со сбором данных, фильтрацией заведомо недостоверных данных и условия в которых эти данные, собираются. Может оказать существенное влияние на наблюдаемые результаты так, например, социологические исследования, которые проводятся в более комфортных для участников условиях могут показывать большую субъективную удовлетворённость продуктом в сравнение с исследованиями, проводимыми в некомфортных. Для получения качественного набора данных в котором присутствуют субъективные признаки или возможно влияние человеческого фактора необходимо сохранять постоянными условия их сбора или максимально диверсифицировать их. Для некоторых находящихся в открытом доступе наборов данных методологии их сбора описаны крайне кратко либо вовсе не описаны. Возможным решением проблемы является сбор данных в условиях максимально приближенных к тем, в которых система будет применяться.

Недостающие детали в данных либо сомнительная точность этих деталей – при построении сложных моделей машинного обучения большую роль играют детали, например, для модели оценки риска раковых заболеваний очень важным является рацион питания человека, однако не все пользователи такой модели ведут журнал потребляемых продуктов, а те люди, которые его ведут используют разный уровень детализации, например некоторые люди могут не вносить данные о продуктах которыми они перекусывали в виде шоколадных батончиков, печенья и т.п. . Сомнительность предоставленных деталей может быть вызвана личными убеждениями, что выливается в попадание в данные только тех деталей, которые характеризуют человека с положительной стороны или выражают его личные убеждения, например, человек может не вносить в базу данных продукты, съеденные на банкетах, которые он считает явно нездоровыми. Исходя из этого можно сделать вывод о необходимости отбора данных с достоверно точными деталями или снижение глубины детализации используемых данных.

Недостаток данных – в зависимости от предметной области получение достаточного количества данных может быть затруднено или вообще недоступно. Очень часто сочетается с недостаточным количеством деталей в существующих данных. Один из путей решения консолидация данных из нескольких источников или собранных с использованием разных методик который в свою очередь приводит к неравномерности их распределения и разному уровню детализации.

Неравномерность распределения – большинство моделей машинного обучения используют классификацию в качестве конечной или промежуточной целей, исключением являются только простые модели типа линейной регрессии. Так модели классификации могут использовать классификацию на промежуточных этапах (например, классификация животное не животное в модели классификатора кот/собака) а модели регрессии могут использовать классификаторы для предварительного разделения данных на группы и применения разных регрессионных моделей к каждой из групп, например, модели типа дерево или лес. Для таких моделей ещё более важным является относительно равномерное распределение данных для предотвращения переобучения для одной группы данных в сочетании с низкой глубиной формируемых

структур для других. В зависимости от используемого алгоритма обучения в лучшем случае результатом обучения на неравномерных данных будет являться узкоспециализированная система.

Ценность данных и их подготовки – не все данные представляют одинаковую ценность для тренировки моделей машинного обучения. При этом данные которые лишены недостатков, описанных выше может быть недостаточно. Подготовка данных является важным процессом, на котором как правило убираются данные достоверность которых вызывает сомнения, а также могут восстанавливаться недостающие детали. Этот процесс довольно трудоёмок и при этом плохо поддаётся автоматизации необходимость использования, которой обусловлена большим объёмом данных.

Проблема отсутствия значимых корреляций. Не все данные могут объяснять видимый результат, а некоторые проблемы могут быть либо детерминированными, либо стохастическими по природе. При этом при наличии больших объёмов данных модели машинного обучения могут демонстрировать точность превышающую точность генератора случайных чисел на 5% что является достаточным для публикации научного исследования, причём цели, преследуемые авторами, могут быть как личными, в виде повышения количества научных статей и публикаций, так и банальный недостаток квалификации и глубины анализа проблемы, чтобы точно установить является ли найденная корреляция случайной или присутствуют слабые причинно-следственные связи. Также возможна ситуация, когда наблюдаемые корреляции существуют не между учитываемыми факторами и наблюдаемыми результатами, а между неучтенным фактором, который может влиять на учтённый в определённых условиях и наблюдаемым результатом. Сложность формируемых моделей при использовании больших данных также усложняет их анализ для определения значимости зависимостей.

Интерпретируемость результатов – для практического применения должна быть четко доказана эффективность иначе бизнес не готов инвестировать в разработку таких систем. Однако учитывая вышенаписанное научное доказательство эффективности сформированных моделей машинного обучения представляется либо очень трудоёмкой, либо вообще невыполнимой задачей (возможно лишь эмпирическое доказательство). Одна из возникающих при этом проблем в то что в отличие от экспертных систем, которые как минимум систематизируют уже имеющиеся знания специалистов, машинное обучение не опирается на них и может демонстрировать более низкую эффективность, которая может быть измерена только с учётом определённого набора данных, а не объяснена логически.

Вывод. Несмотря на наличие большого количества проблем и ограничений машинное обучение прочно входит в повседневную жизнь и тенденции показывает только вероятность более плотной интеграции его в большинство продуктов сферы информационных технологий. Несмотря на существенные различия методологии в сравнении с точными науками машинное обучение позволяет решать проблемы, которые недоступны, например, простому статистическому анализу из-за низких коэффициентов корреляции и/или сложности связей.

ЛИТЕРАТУРА

1. Ограничения машинного обучения [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/462365>. – Дата доступа: 15.09.2020.
2. Проблемы применения машинного обучения для решения реальных задач [Электронный ресурс]. – Режим доступа: <http://robocraft.ru/blog/machinelearning/3710.html>. – Дата доступа: 15.09.2020.
3. 4 способа решить проблемы внедрения Machine Learning в стартапы [Электронный ресурс]. – Режим доступа: <https://datastart.ru/blog/read/4-sposoba-reshit-problemy-vnedreniya-machine-learning-v-startapy>. – Дата доступа: 15.09.2020.
4. Major Challenges for Machine Learning Projects [Электронный ресурс]. – Режим доступа: <https://www.topbots.com/major-challenges-machine-learning-projects>. – Дата доступа: 15.09.2020.
5. Machine learning: 9 challenges [Электронный ресурс]. – Режим доступа: <https://www.kaspersky.com/blog/machine-learning-nine-challenges/23553>. – Дата доступа: 15.09.2020.

УДК 004.051

**УСПЕШНЫЕ ПРИМЕРЫ ПРИМЕНЕНИЯ МАШИННОГО ОБУЧЕНИЯ
И ПРИЧИНЫ ИХ УСПЕХА****П.В. ДМИТРИЧЕНКО***(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

Изучены успешные проекты использующие машинное обучение на основании их анализа сделаны выводы о возможных технических причинах их успеха.

Введение. Деньги являются двигателем прогресса, а отсутствие возврата инвестиций эволюционным механизмом естественного отбора. Успешные проекты открывают дорогу тысячам подражателей и просто проектам которые могут заимствовать элементы успешного. Для понимания причин этого успеха необходим анализ технических условий в которых зарождались и росли наиболее успешные проекты использующие машинное обучение.

Финансовая часть является важной составляющей успеха данных проектов, но ввиду личного интереса автора внимание будет сконцентрировано на технических деталях, которые приводят к их успешному применению машинного обучения на практике.

Основная часть. Для начала кратко рассмотрим наиболее успешные проекты и стартапы использующие машинное обучение.

Финансовые проекты. Одними из наиболее перспективных проектов являются системы применяющие машинное обучение для вычисления счёта кредитования по которому определяются условия выдачи кредитов.

Affirm — предоставляет потребительский кредит для онлайн-покупок.

Upstart – платформа онлайн-кредитования, AI и machine learning применяет для оценки параметров кредита и автоматизации его получения. Проект работает и на b2b-рынке.

Здравоохранение является одной из областей активного развития машинного обучения.

Flatiron Health - оператор облачной платформы, специализированной на онкологии.

Сервисы, предоставляющие платформу для генерируемого пользователями контента или коммерческого. В данном контексте машинное обучение как правило используется для определения некачественного или сворованного контента, ранжирования, и генерации рекомендаций.

Генеральный директор YouTube, Сьюзен Войжитски, рассказала о том, что видео-сервис начнёт гораздо строже наказывать владельцев каналов, которые вредят его репутации. Так, будет реализовано введение штрафных санкций против авторов, которые «делают что-то вопиющее, что наносит значительный вред» всему сообществу. В частности, сервис наймёт для борьбы с нарушениями больше людей и начнёт ещё активнее использовать машинное обучение [1].

Youtube также использует машинное обучение для генерации рекомендаций и ранжирования этих рекомендаций.

Судя по недавнему патенту, Apple Watch смогут рекомендовать плейлисты из iTunes, которые идеально подойдут вашему сердечному ритму. Любая компания, которая работает с интернетом вещей, может использовать подобную технологию. Подключение нескольких устройств с одинаковым набором данных для обучения, сможет улучшить качество анализа, а значит и опыта клиента [4].

Проекты в сфере торговли являются одними из первых к которым было применено машинное обучение как правило в виде генерации рекомендаций, Amazon пошёл дальше этого и вводил персонализированные цены (данная схема также применена Ozon.ru и Dostavka.ru).

Amazon Personalize позволяет в режиме реального времени создавать индивидуально подобранные рекомендации по продуктам и контенту, а также целевые маркетинговые акции, чтобы активнее привлекать клиентов. В основе этого сервиса лежат 20 лет опыта работы компании Amazon в сфере рекомендаций и исследования с использованием машинного обучения. Сервис предоставляет в круглосуточное распоряжение клиентов систему рекомендаций на основе машинного обучения, подобной той, что используется на сайте Amazon.com [3].

Помимо персональных рекомендаций интересной фишкой на Amazon является персонализация цен. Стоимость товаров, которые похожи на те, что пользователь покупал ранее, значительно ниже стоимости других товаров из этой категории [4].

Крупнейшая в мире торговая площадка Alibaba широко использует машинное обучение и другие средства ИИ. В результате ее виртуальные витрины подстраиваются под каждого покупателя, а система поиска выдает для него наилучшие варианты. Чат-бот Ali Xiaomi может самостоятельно справиться с большинством обращений клиентов в техподдержку. [5].

Американская торговая сеть Target обнаружила, что с помощью машинного обучения можно предсказать не только поведение покупателей, но и перемены в их жизни, например, беременность. Алгоритмы Target работают настолько точно, что с помощью данных о покупках могут определить триместр беременности женщины, которая их совершала [4].

Далее будут рассмотрены возможные технические причины успешности вышеописанных и подобных им проектов.

Интеграция системы машинного обучения в уже существующие продукты, которые обладают большим объёмом собранных данных. Сбор данных используемых системами машинного обучения осуществляется уже существующими приложениями и сервисами. Консолидация из нескольких источников как это часто происходит в академических целях не происходит. Это избавляет такие проекты от множества проблем таких как неравномерность используемых данных, неравномерность детализации, а также различия в условиях, в которых данные собраны и используются.

Использование больших объёмов данных. Интеграция машинного обучения в уже существующие успешные и обладающие большими аудиториями продукты сильно упрощает сбор качественных данных если они уже не собраны и подготовку их для использования по методу больших данных. Кроме того, продукты обладающие большой аудиторией могут получать обратную связь в виде опросов пользователей об их удовлетворённости внедряемыми системами, а также получать об этом информацию исходя из поведения пользователей в приложении (например, вышел или продолжил пользоваться после показа контекстной рекламы, общее количество времени, проведённого в приложении после внедрения машинного обучения и т.п.).

Отсутствие рисков при получении неверных результатов. Большинство успешных проектов, использующих машинное обучение используют его для генерации рекомендаций или ранжирования существующего контента. Если подобные системы ошибаются владеющая ими компания не несёт больших финансовых потерь (в сравнении с оценкой экспертов или использования экспертных систем в том числе существующих в виде критериев на бумаге и/или формул) и как правило не может быть привлечена к легальной ответственности (исключения касаются законов о дискриминации и равенстве представления для, например, платформ, использующих машинное обучение для ранжирования контента). Это можно сравнить с экспертными системами, которые могут быть использованы для оценки состояния здоровья людей и выявления у них заболеваний, когда ложноотрицательная ошибка таких систем может привести к ухудшению здоровья пользователей вплоть до летальных исходов.

Вывод. Успешность внедрения машинного обучения в уже существующие информационные системы с большими объёмами данных собранных в этих системах, а не заимствованных из чужих наборов данных для решения задач не имеющих больших последствий при неверных единичных случаях обладает наибольшей перспективностью.

ЛИТЕРАТУРА

1. YouTube введёт санкции и машинное обучение для нейтрализации зловредных авторов [Электронный ресурс]. – Режим доступа: <https://hi-tech.ua/youtube-vvedyot-sanktsii-i-mashinnoe-obuchenie-dlya-neytralizatsii-zlovrednyih-avtorov/>. – Дата доступа: 15.09.2020.
2. Самые перспективные ИИ-стартапы [Электронный ресурс]. – Режим доступа: <https://vc.ru/finance/52042-samy-perspektivnye-ii-startapy>. – Дата доступа: 15.09.2020.
3. Amazon Personalize [Электронный ресурс]. – Режим доступа: <https://aws.amazon.com/ru/personalize/>. – Дата доступа: 15.09.2020.
4. Персонализация в Amazon.com, Ozon.ru и Dostavka.ru [Электронный ресурс]. – Режим доступа: <https://retailrocket.ru/blog/personalizatsiya-v-amazon-ozon-i-dostavka-ru/>. – Дата доступа: 15.09.2020.
4. 5 примеров использования машинного обучения в бизнесе [Электронный ресурс]. – Режим доступа: <https://rb.ru/story/machine-learning-in-business>. – Дата доступа: 15.09.2020.
5. Искусственный интеллект от Alibaba прошел экзамен в Стэнфорде лучше людей [Электронный ресурс]. – Режим доступа: <https://rb.ru/news/ai-alibaba-best/>. – Дата доступа: 15.09.2020.
6. 3 Criteria for a Successful Machine Learning Project [Электронный ресурс]. – Режим доступа: <https://in-datalabs.com/blog/criteria-for-successful-machine-learning-project>. – Дата доступа: 15.09.2020.
7. Frame, Organize and Manage a Successful Machine Learning Project [Электронный ресурс]. – Режим доступа: <https://www.machinelearningmindset.com/organize-machine-learning-project/>. – Дата доступа: 15.09.2020.

УДК 004.4

РАЗРАБОТКА ВЕБ-ИНТЕРФЕЙСА С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА ANGULAR ДЛЯ РЕСУРСА «ВЕБ-ПЛАТФОРМА ДЛЯ СОЦИАЛЬНОГО ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЕЙ»

Н.В. ДУБОВСКИЙ

(Представлено: канд. техн. наук, доц. О.Н. ПЕТРОВИЧ)

В данной статье рассматривается способ создания веб-интерфейса для сайта «Веб-платформа для социального взаимодействия пользователей».

Введение. Понятный для пользователя веб-интерфейс – это основной критерий, на который нужно обращать особенное внимание при разработке веб-приложения. Веб-интерфейс должен удовлетворять основным требованиям пользователя к данному приложению и не иметь перегруженного функционала.

Выбор технологии разработки веб-интерфейса. Каждое веб-приложение имеющее веб-интерфейс должно иметь грамотно спроектированную архитектуру и разделение на частично или полностью независимые модули. Данное разделение модулей облегчает и модификацию, и расширение самого приложения. Суммируя выше сказанное, было принято решение использовать фреймворк Angular для реализации веб-интерфейса и ASP.NET Core для реализации серверной части.

Проектирование. На каждой странице сверху находится блок с логотипом и меню. Данный блок может динамически изменять количество пунктов в зависимости от того авторизован ли пользователь и какую роль имеет, показано на рисунке 1, 2, 3. Нижняя же часть сайта не перегружена излишней информацией.



Рисунок 1. – Виды заголовков страниц



Рисунок 2. – Виды заголовков страниц пользователя



Рисунок 3. – Виды заголовков страниц администратора

На экране, показанном на рисунке 1, находится меню сайта, содержащее в себе ссылки по главной странице, строку поиска и регистрацию, либо ссылку на вход в личный кабинет.

На экране с формой входа и регистрации, представлено на рисунке 4, пользователь должен ввести контактные данные и отправить их, после чего он будет зарегистрирован или авторизован и сможет пользоваться функциями личного кабинета.

На главной странице сайта, представленной на рисунке 5, расположено 3 блока с информацией. Первый блок содержит в себе последние ответы на вопросы. Второй блок – последние заданные вопросы с их рейтингом. Третий блок представлен в виде рейтинга пользователей. В нижней части страницы расположена информация о проекте, содержащая ссылки на главные компоненты приложения.

На странице личного кабинета пользователя, представленного на рисунке 6, так же расположено 3 блока с информацией. Первый блок содержит в себе персональную информацию пользователя. Второй блок – последние заданные вопросы данным пользователем. Третий блок содержит список подписок данного пользователя.

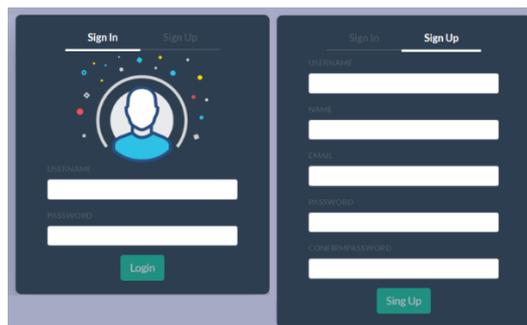


Рисунок 4. – Форма входа и регистрации.

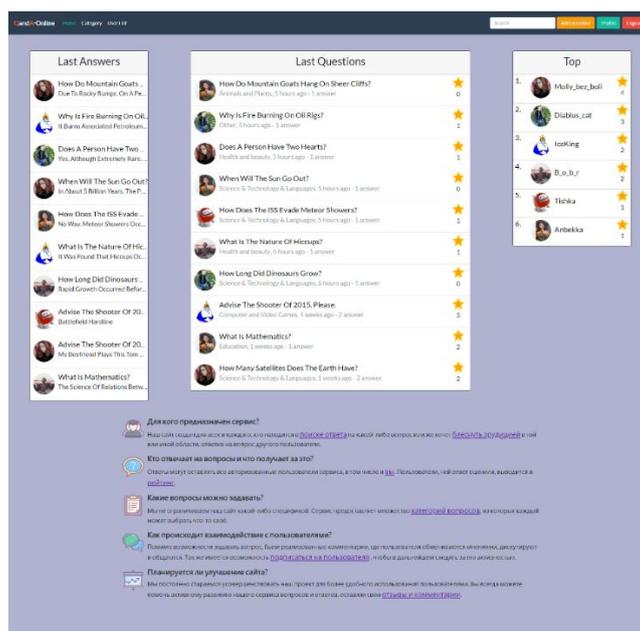


Рисунок 5. – Виды заголовков страниц



Рисунок 6. – Личный кабинет пользователя

Заключение. В данной статье рассмотрена базовая структура интерфейса. Были сформулированы необходимые разделы программного средства, для каждого из которых были спроектированы и созданы страницы сайта.

ЛИТЕРАТУРА

1. Figma.com [Электронный ресурс]. Режим доступа: <https://www.figma.com/>. – Дата доступа: 20.09.2020.
2. Protopie.io [Электронный ресурс]. Режим доступа: <https://www.protopie.io/>. – Дата доступа: 21.09.2020.

УДК 004.4

ВЕБ-ПЛАТФОРМА ДЛЯ СОЦИАЛЬНОГО ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЕЙ**Н.В. ДУБОВСКИЙ***(Представлено: канд. техн. наук, доц. О.Н. ПЕТРОВИЧ)*

В статье представлен объект разработки – «Веб-платформа для социального взаимодействия пользователей». Рассмотрены вопросы выбора и обоснования средств проектирования базы данных.

Введение. Веб-приложение для социального взаимодействия пользователей в первую очередь должно представлять собой удобное, защищенное, с высокой скоростью доступа хранилище данных о вопросах, ответах на них, а также персональных данных пользователей. В статье представлено описание базы данных для разрабатываемой платформы.

Основной раздел. Наиболее распространенным средством моделирования данных (предметной области) является модель «сущность-связь» (ERM). Понятием ERM являются сущность, связь и атрибут.

Сущность (Entity) – реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области.

Каждая сущность должна обладать уникальным идентификатором. Каждый экземпляр сущности должен однозначно идентифицироваться и отличаться от всех других экземпляров данного типа сущности. Каждая сущность должна обладать некоторыми свойствами:

- каждая сущность должна иметь уникальное имя, и к одному и тому же имени должна всегда применяться одна и та же интерпретация. Одна и та же интерпретация не может применяться к различным именам, если только они не являются псевдонимами;

- сущность обладает одним или несколькими атрибутами, которые либо принадлежат сущности, либо наследуются через связь;

- сущность обладает одним или несколькими атрибутами, которые однозначно идентифицируют каждый экземпляр сущности;

- каждая сущность может обладать любым количеством связей с другими сущностями модели.

Связь (Relationship) – поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области. Связь – это ассоциация между сущностями, при которой, как правило, каждый экземпляр одной сущности, называемой родительской сущностью, ассоциирован с произвольным (в том числе нулевым) количеством экземпляров второй сущности, называемой сущностью-потомком, а каждый экземпляр сущности-потомка ассоциирован в точности с одним экземпляром сущности-родителя. Таким образом, экземпляр сущности-потомка может существовать только при существовании сущности родителя.

Связи может даваться имя, выражаемое грамматическим оборотом глагола и помещаемое возле линии связи. Имя каждой связи между двумя данными сущностями должно быть уникальным, но имена связей в модели не обязаны быть уникальными. Имя связи всегда формируется с точки зрения родителя, так что предложение может быть образовано соединением имени сущности-родителя, имени связи, выражения степени и имени сущности-потомка.

Атрибут – любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут представляет тип характеристик или свойств, ассоциированных со множеством реальных или абстрактных объектов (людей, мест, событий, состояний, идей, пар предметов и т.д.). Экземпляр атрибута – это определенная характеристика отдельного элемента множества. Экземпляр атрибута определяется типом характеристики и ее значением, называемым значением атрибута. В ER-модели атрибуты ассоциируются с конкретными сущностями. Таким образом, экземпляр сущности должен обладать единственным определенным значением для ассоциированного атрибута.

В ходе анализа знаний и проектирования базы данных выявлены основные сущности:

- Сущность Users описывает пользователей, а именно личную информацию о них.
- Сущность subscriptions описывает подписки между пользователями.
- Сущность files описывает хранимые файлы, характеризующаяся именем файла, типом файла, датой загрузки.

- Сущность questions описывает вопросы, состоит из заголовка, тела вопроса, даты, категории к которой принадлежит.

- Сущность answers описывает ответ, представлена телом ответа, датой ответа.

- Сущность categories описывает категории, содержит название категории.

- Сущность Recalls описывает отзывы пользователей о сайте, хранит заголовок и тело отзыва.

– Сущность Complaints описывает жалобы пользователей, включающая в себя текст жалобы, отметку о просмотре.

– Сущность Notifications описывает уведомления, включает в себя дату, тип и отметку просмотра.

Одно из важнейших достоинств реляционных баз данных состоит в том, что можно хранить логически сгруппированные данные в разных таблицах и задавать связи между ними, объединяя их в единую базу. Такая организация данных позволяет уменьшить избыточность хранимых данных, упрощает их ввод и организацию запросов и отчетов.

В каждой таблице базы данных может существовать первичный ключ. Под первичным ключом понимают поле или набор полей, однозначно (уникально) идентифицирующих запись. Первичный ключ должен быть минимально достаточным: в нем не должно быть полей, удаление которых из первичного ключа не отразится на его уникальности.

Для организации связей между двумя и более таблицами, используются вторичные ключи. Они служат ограничителями целостности связей нескольких таблиц, т.к. подчиненная таблица не может ссылаться на несуществующие записи главной таблицы (что позволяет строить целостные модели данных). Существует три вида связей между таблицами:

- «один-ко-многим»;
- «один-к-одному»;
- «многие-ко-многим».

Связь «один-ко-многим» – самая распространенная для реляционных баз данных. Для обеспечения ссылочной целостности в дочерней таблице создается внешний ключ, через который организовывается связь с дочерней таблицей. Обязательное условие состоит в том, что внешний ключ по составу полей должен совпадать с первичным ключом родительской таблицы.

Поэтому, для реализации приложения, необходимо рассмотреть всю информационную систему в совокупности, определить взаимное влияние сущностей, установить все связи между сущностями и объектами, что представлено на рисунке 1.

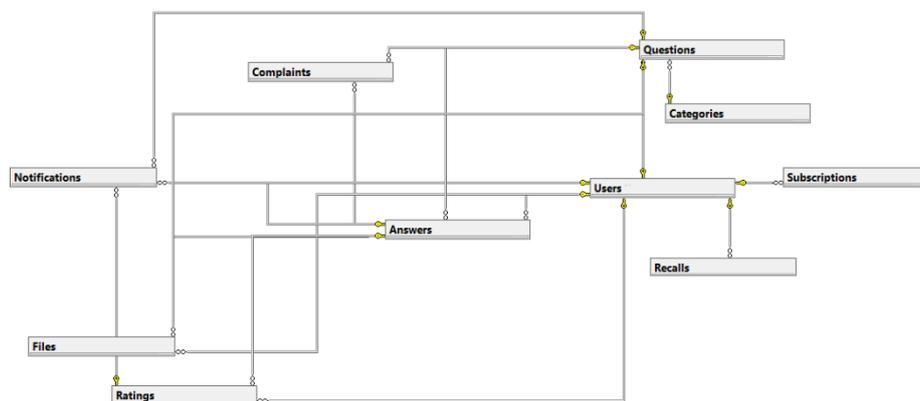


Рисунок 1. – Связи сущностей

Также, важным является выбор системы управления базой данных. Для разработки представленной базы данных удобно использовать MS SQL Server 2018, т.к. данный сервер поддерживает множество типов данных необходимых при разработке и является удобным в использовании.

Заключение. В данной статье были рассмотрены основные цели и принципы моделирования базы данных для веб-приложения. Также были рассмотрены основные сущности спроектированной базы данных.

ЛИТЕРАТУРА

1. Информационный ресурс компании «Интерфейс» [Электронный ресурс]. Режим доступа: <http://www.interface.ru/home.asp?artId=22456>. – Дата доступа: 21.09.2020.
2. Информационный ресурс кафедры информатики УГАТУ [Электронный ресурс]. Режим доступа: <http://informatic.ugatu.ac.ru/lib/office/Proekt.htm>. – Дата доступа: 22.09.2020.

УДК 004.021

ОБЗОР ФРЕЙМВОРКА XAMARIN

А.В. ЕПАНЕШНИКОВ*(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

Представлено описание кроссплатформенного фреймворка Xamarin. Приведен пример использования фреймворка в спутниковой связи. Приведено описание основных частей. Описаны ограничения, которые могут возникнуть при использовании в разных операционных системах.

Введение. На сегодняшний день мобильные приложения находятся на пике своей популярности. Количество разработчиков мобильных приложений увеличивается, количество доступных приложений растет, а также и число их загрузок. В связи с этим развивается большое количество кроссплатформенных мобильных фреймворков для успешного создания приложений. Повсеместно мобильные приложения используют данные со спутников, при этом фреймворк позволяет сделать общение с модулями спутниковой связи намного удобнее, поскольку не требуется писать отдельные нативные приложения для каждой платформы.

Основной раздел. Перед нами стоит задача разработать мобильное приложение, которое будет собирать информацию со всех интернет-магазинов, а также порталов, связанных с книжной индустрией. Для этого нам нужно выбрать кроссплатформенный мобильный фреймворк, в котором создание приложений осуществляется на языке программирования C#. На данный момент существует семь популярных кроссплатформенных мобильных фреймворков, таких как Appcelerator Titanium, Kony Platform, Adobe PhoneGap, IBM Worklight, Telerik Platform, Verivo Akula и Xamarin [1]. Но только Xamarin создает приложения с помощью языка C#.

Xamarin – это фреймворк для кроссплатформенной разработки мобильных приложений (iOS, Android, Windows Phone) с использованием языка C#. Xamarin основан на open-source реализации платформы .NET – Mono. Код программы пишется на широко используемом языке программирования C# с применением всех привычных языковых особенностей, к примеру, LINQ, лямбда-выражений, Generic и Async. При этом имеется полный доступ ко всем возможностям SDK платформы и механизму создания UI (user interface), получая на выходе приложение, которое, строго говоря, ничем не отличается от нативных и не уступает им в производительности.

Фреймворк Xamarin состоит из нескольких основных частей: – Xamarin.iOS – библиотека классов, предоставляющая разработчику доступ к iOS SDK; – Xamarin.Android – библиотека классов, предоставляющая доступ к Android SDK; – компиляторы для iOS и Android; – IDE Xamarin Studio; – плагин для Visual Studio [2]. Ограничения в Xamarin.iOS связаны в том, что в iOS, в отличие от .NET и Mono нет виртуальной машины. Поэтому возникают трудности с поддержкой Generic. Отсюда возникают такие ограничения: – нельзя использовать Virtual Generic методы, так как компилятор не может учесть все возможные варианты использования; – нельзя создавать Generic-наследников от класса NSObject, который является базовым в иерархии Objective-C [3].

Для каждой платформы Xamarin предоставляет возможность использовать нативные средства разработки UI и нативные элементы пользовательского интерфейса. Для Android создание UI может происходить непосредственно в коде или же при помощи декларативного подхода с описанием интерфейса в XML. Для iOS это также либо код, либо использование нативных средств проектирования интерфейса. Для каждой из платформ потребуется реализовать собственный слой UI, т. е. код, который отвечает за внешний вид приложения, придется написать для каждой платформы отдельно.

Если разбивать приложение на слои, то получается такая схема: – Data Layer (DL) – хранилище данных, например, база SQLite или xml-файлы; – Data Access Layer (DAL); – Business Layer (BL) – слой, содержащий бизнеслогику приложения; – Service Access Layer (SAL) – слой, отвечающий за взаимодействие с удаленными сервисами; – Application Layer (AL) – слой, содержащий платформозависимый код; – User Interface Layer (UI) – слой пользовательского интерфейса [4].

Кроссплатформенными являются все слои, расположенные выше Application Layer. Разработчики Xamarin в качестве среды разработки предлагают использовать либо собственную IDE – Xamarin Studio, либо Visual Studio. Xamarin Studio – кроссплатформенная IDE, которая работает как на Mac OS X, так и на Windows [2].

Также Xamarin предлагает возможность вести разработку в Visual Studio после установки специального плагина [5].

Заключение. Таким образом, можно сделать вывод, что фреймворк Xamarin удовлетворяет всем необходимым возможностям и может использоваться при дальнейшей разработке мобильного приложения.

ЛИТЕРАТУРА

1. Корпоративный блог «Центр высоких технологий» [Электронный ресурс]. – Режим доступа: <http://blog.htc-cs.ru/post/nativevsframeworks>. – Дата доступа: 20.09.2020.
2. Официальный сайт Xamarin [Электронный ресурс]. – Режим доступа: <http://xamarin.com/>. – Дата доступа: 20.09.2020.
3. Arstechnica [Электронный ресурс]. – Режим доступа: <http://arstechnica.com/information-technology>. – Дата доступа; 20.09.2020.
4. ПрогХаус [Электронный ресурс]. – Режим доступа: <http://www.proghouse.ru/programming/27-xamarin>. – Дата доступа: 20.09.2020.
5. Блюстайн М. Изучаем MonoTouch. Создание приложений на платформе iOS с помощью C# и .NET. – М. : ДМК Пресс, 2012. 336 с.

УДК 004.432.2

SWIFT – ЯЗЫК, КОТОРЫЙ ИЗМЕНИТ МИР ПРОГРАММИРОВАНИЯ**А.В. ЕПАНЕШНИКОВ***(Представлено: канд. физ-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

Рассматривается язык программирования swift, проанализированы его положительные и отрицательные стороны, приведены комментарии о данном языке от известных разработчиков, от самой компании Apple, а также приведены примеры наиболее популярных приложений и программ.

Введение. Swift – язык программирования, созданный компанией Apple, разработанный в приоритете для разработчиков операционных систем IOS и OS X. Swift представляет собой протоколориентированный, объективно-ориентированный, функциональный и императивный язык программирования. Язык Swift был представлен компанией Apple на конференции WWDC второго июня 2014 года, автором является Крис Латтнер.

Основной раздел. Язык создан на базе следующих языков программирования: CLU, C#, Python, Objective-C, Rust, Haskell, Ruby и другие. На сегодняшний день имеются три версии языка Swift: 1.0, выпущенная в 2014 году, 2.0, выпущенная в 2015 году и версия 3.0, предоставленная в конце 2015 года. Известные программисты, создавшие немало приложений на языке Swift говорят о нем следующее:

– Denis Mikan создатель игры «Blek»: «Язык Swift ликвидирует трудности, которые скопились за двадцать лет в языке Objective-C. Он призван, в первую очередь, облегчить ситуацию тех разработчиков приложений, которым крайне некомфортно трудиться с огромными, сложными конструкциями Objective-C. Язык Swift так же откроет новую эру для непрофессиональных разработчиков, которые сейчас кроме разработки маленьких игр на JavaScript, так же смогут выбирать и этот язык».

– Сама же фирма Apple утверждает, что: «Swift – это надёжный и интуитивно понятный язык программирования от Apple, при помощи которого можно создавать приложения для iOS, Mac, Apple TV и Apple Watch. Он дает разработчикам небывалую свободу творчества, а также придумать и воплотить в жизнь что-то необыкновенное. Благодаря этому простому и удобному языку с открытым кодом вам достаточно просто интересной идеи, чтобы создать нечто невероятное». [1-3]

Как любой язык программирования Swift имеет свои положительные и отрицательные стороны. К плюсам стоит отнести высокую производительность, например, Swift до 2,6 раз быстрее, чем ObjectiveC и до 8,7 раз быстрее, чем Python 2.7. Swift является открытым языком программирования; реализован упрощенный принцип работы с повторяющимися строками и заявлениями; читабелен; полноценное взаимодействие с кодом, написанном на языке Objective-C. Высокая безопасность, выраженная в обработке указателей, «дотошности» компилятора и в том, что в саму компиляцию присутствует возможность встроить опциональную переменную nil для обеспечения обратной связи. [4]

Но не все так идеально, в Swift есть и свои минусы, которые Apple обещает исправить в новой версии: для работы на Swift из-за непереуказанных кодов OS X и IOS требуется минимальное знание языка Objective-C. Разработчику, перешедшему с других языков программирования, покажется, как минимум, странно то, что компилятор Swift выдает излишние и сбивающие с толку ошибки, которые на других языках отсутствовали. [3]

На сегодняшний день создано множество игр, приложений и программ на языке Swift. Самыми известными считаются Duolingo – позволяет самостоятельно изучить иностранные языки за счет игровой формы обучения; Khan Academy – академия, предоставляющая каждому высококачественное образование всегда и везде; Procreate – позволяет реализовать человеку, его творческие навыки в рисовании; VSCO – приложение для обработки снимков; Firefox – один из самых известных браузеров, предоставляющий быстрый доступ к различным поисковым системам; WordPress – позволяет управлять собственным сайтом; Wire – защищенный мессенджер, позволяющий пользователям общаться, не задумываясь о проблемах безопасности. [1] Отдельно стоит выделить приложение Evenbrite – сервис для организации мероприятий, с его помощью можно продавать билеты в режиме он-лайн, собирать информацию о посетителях и управлять записями. Разработчики данного приложения считают, что у каждой организации должен быть инструмент, позволяющий организации планировать мероприятия, и вести свою экономическую характеристику, отчетность. [4]

Заключение. В заключение стоит сказать, что за три года язык Swift набрал огромную популярность среди разработчиков, язык становится массовым. Не только разработчики заметили огромный потенциал языка Swift. Несколько ведущих университетов и учебных заведений уже преподают его на уроках компьютерного программирования и публикуют бесплатные курсы в iTunes U. Следовательно, даже новички могут без особых проблем перейти к программированию на более высоком уровне. К университетам, которые включили язык программирования Swift в учебную программу относятся Политехнический университет штата Калифорния, университет Фулл Сейл, южный методический университет, королевский мельбурнский технологический университет, стэнфордский, мюнхенский университеты и многие другие. Все это подтверждает то, что язык Swift изменяет мир программирования в лучшую сторону, а также способствует простой и эффективной работе компаний, за счет созданных программ.

ЛИТЕРАТУРА

1. Академия Geekbrain [Электронный ресурс]. – Режим доступа: <https://geekbrains.ru/>. – Дата доступа: 20.09.2020.
2. Библиотека Википедия [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Swift>. – Дата доступа: 20.09.2020.
3. Apple [Электронный ресурс]. – Режим доступа: <http://www.apple.com/ru/swift/>. – Дата доступа: 20.09.2020.
4. Электронный журнал Ip iPhones.ru [Электронный ресурс]. – Режим доступа: <https://www.iphones.ru/iNotes/366329>. – Дата доступа: 20.09.2020.

**СОЗДАНИЕ АВТОМАТИЗИРОВАННЫХ УСТРОЙСТВ
ДЛЯ ФИЗИЧЕСКИХ ИЗМЕРЕНИЙ****С.Ю.ЗМИТРОВИЧ***(Представлено: канд. физ.-мат. наук, доц. С.А. ВАБИЩЕВИЧ)*

На современном этапе развития технологий не представляется возможным проведение исследований без использования автоматизированных средств измерений с возможностью наглядного отображения протекающего физического процесса. В настоящей статье рассматривается использование микроконтроллеров, их внутренней периферии для создания автоматизированных устройств физических измерений.

Ключевые слова: автоматизация эксперимента, микроконтроллеры, структурная схема, физические измерения.

Введение. Микроконтроллеры – это производные от микропроцессоров интегральные микросхемы с ЦПУ, памятью, устройством ввода/вывода и собственной периферией. С их изобретением появилась возможность проектирования сложных измерительных систем с относительно небольшими и малыми размерами. Работают данные интегральные микросхемы точно по записанной в их память программе, без каких-либо отклонений от алгоритма, если не учитывать какие-либо сбои в их работе

Микроконтроллеры можно сопрягать с различными видами датчиковой аппаратуры, после чего отображать информацию, полученную от них, на дисплее или передавать для дальнейшей обработки на компьютер. Исходя из полученных данных, микроконтроллерные системы могут принимать различные решения, записанные в их алгоритм работы, по настройке или управлению различных частей их периферии [1,2].

Параметры микроконтроллеров. Микроконтроллеры бывают нескольких разрядностей: 8-битные, 16-битные, 32-битные. Возможности микроконтроллера зависят от используемой архитектуры: RISC, CISC, ARM и др. В современности, большинство компаний делают упор на производительную ARM с ядрами Cortex-M. Использование микроконтроллеров с архитектурой ARM позволяет разработчику перенести код программ между микроконтроллерами различных серий без особых проблем с их адаптацией.

При выборе микроконтроллера для использования в информационно-измерительной технике важно учитывать наличие различной периферии микроконтроллеров, такой как аналого-цифровые преобразователи, различные интерфейсы передачи данных, порты ввода-вывода, таймеров и др. Порты ввода-вывода служат для ввода и вывода цифровой и аналоговой информации, некоторые входы используются для программирования микроконтроллера, для использования таймеров, широтно-импульсного модулирования, компараторов и шин передачи данных.

Аналого-цифровой преобразователь (АЦП) позволяет измерить аналоговую величину на соответствующем входе микроконтроллера, после чего существует возможность математической обработки или передачи её по цифровому интерфейсу другому устройству.

Цифровые интерфейсы, такие как USART, I²C, 1-Wire, USB, позволяют исследователю проводить сопряжения с датчиковой аппаратурой, передавать данные на компьютер или взаимодействовать с другими устройствами.

Разработчику стоит принимать во внимание количество Flash и RAM памяти микроконтроллера, предназначенной для записи программы и хранения переменных.

Счётчики/Таймеры могут быть применены в измерении времени, в подсчёте тактов или внешних импульсов, способны генерировать прерывания при опустошении или переполнении.

Автоматизированные устройства для физических измерений.

В качестве примера использования микроконтроллеров при создании информационно-измерительной техники представим систему измерения показаний физических величин PhyZModule [3], структурная схема которой представлена на рисунке 1. Система построена на основе микроконтроллера STM32F103C8T6 и использует следующую его периферию: АЦП, DMA, USB, I²C, USART, SPI. Программным образом была реализована передача данных по шине 1-Wire. Для отображения полученных измерений, система, посредством двунаправленной шины USB, передаёт данные на компьютер, где в соответствующем программном обеспечении происходит построение графиков изменения физических величин.

Для совместного использования с системой PhyZModule, разработаны установки для исследования теплоизоляционных свойств материалов и анализа параметров источников искусственного света, представленные на рисунках 2а и 2б соответственно.

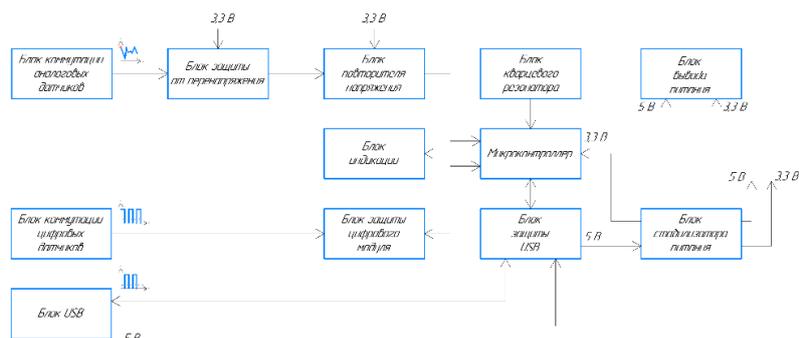


Рисунок 1. – Структурная схема системы измерения физических величин PhyZModule

Проведение опыта по определению теплоизоляционных свойств материала заключается в нагреве внутренней среды установки до определённой температуры с последующим её охлаждением, а также замерами температуры внутри и снаружи установки. Для этого используются датчики, на основе термисторов с отрицательным температурным коэффициентом В57164-К 222-Ј и В57164-К 103-Ј, размещённые как представлено на рисунке 2а: «1» - внутри установки; «2» - внутри стенки; «3» - снаружи на боковой стенке; «4» - инфракрасный обогреватель; «5» - снаружи на верхней стенке; «6» - материал с известными теплофизическими характеристиками; «7» - исследуемый материал [3].

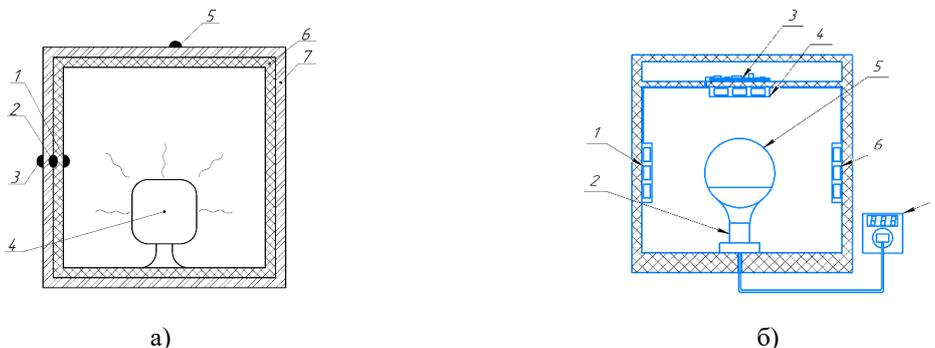


Рисунок 2. – Установки для исследования теплоизоляционных свойств материалов (а) и анализа параметров источников искусственного света (б)

Анализ параметров источников искусственного света (ИИС) происходит следующим образом: ИИС «5» включается в патрон «2», питающие провода которого идут от ваттметра «7», через программное обеспечение настраивается графическое изображение измеряемых величин и микроконтроллерное устройство «3» с подключёнными к нему оптоэлектронными датчиками «1» «4» «6»; ваттметр «7» включается в розетку, начинает считывать потребляемую электрическую энергию от ИИС «5»; свет излучаемый от ИИС «5» падает на оптоэлектронные датчики «1» «4» «6», те в свою очередь начинают фиксировать измеряемые параметры светового потока, силы света, мерцаний ИИС, спектральную характеристику; после испытаний происходит расчёт световой отдачи ИИС «5», анализ полученных параметров, сравнение их с техническими характеристиками на ИИС «5» и регламентируемыми нормами на прибор [4].

ЛИТЕРАТУРА

1. Катцен, С. PIC-микроконтроллеры. Полное руководство / С. Катцен, А.В. Евстифеева. – Москва: ДМК Пресс, 2014 – 651 с.
2. Уилмсхерст, Т. Разработка встроенных систем с помощью микроконтроллеров PIC / Т. Уилмсхерст, Ю.А. Шпак. – Киев: Мк-Пресс, 2015 – 543 с.
3. Змитрович, С.Ю. Автоматизированная система физических измерений / С.Ю.Змитрович, С.А.Вабищевич, Д.Н.Шабанов // Вестник Полоцкого государственного университета. – 2019. – №4. – С.45-49.
4. Змитрович, С.Ю. Установка для исследования параметров источников искусственного света [Электронный ресурс] С.Ю. Змитрович // Электронный сборник трудов молодых специалистов Полоцкого государственного университета. Выпуск 30 (100). Промышленность. – Новополоцк: ПГУ, 2019. С.338-339. – Электронный оптический диск – 1 диск.

УДК 004.622: 53.082.52

ИСПОЛЬЗОВАНИЕ МИКРОКОНТРОЛЛЕРОВ
ДЛЯ ОБРАБОТКИ АНАЛОГОВЫХ СИГНАЛОВ

С.Ю.ЗМИТРОВИЧ

(Представлено: канд. физ.-мат. наук, доц. С.А. ВАБИЩЕВИЧ)

В настоящей статье рассматривается использование микроконтроллеров в качестве измерительного устройства аналоговых величин, схемотехнические решения по защите портов модуля аналого-цифрового преобразователя и программное обеспечение для анализа полученных результатов.

Ключевые слова: микроконтроллер, измерение, датчики, аналого-цифровой преобразователь, программное обеспечение.

В современной практике экспериментальной работы актуальным является использование устройств, позволяющих проводить комплексные исследования физических процессов. Рациональным представляется использование датчиков аналоговых сигналов, параметры которых описываются непрерывной функцией времени и множеством возможных величин. Для этого требуется создать измерительное устройство, которое преобразует показания датчиков физических величин в цифровые сигналы. Наличие такого устройства позволит осуществлять многопараметрические исследования, повысить эффективность и достоверность экспериментальных данных [1].

Осуществить построение эффективных измерительных систем возможно с использованием микроконтроллеров, имеющих в своей архитектуре большое количество модулей, основными из которых являются: аналого-цифровой преобразователь, модули интерфейсов передачи данных, таймеры и т.д.

Аналого-цифровой преобразователь (АЦП) – модуль микроконтроллера, преобразующий аналоговый сигнал в цифровой код. Каждый АЦП модуль характеризуется частотой дискретизации и разрядностью квантования сигнала. Чем выше данные параметры, тем точнее аналоговой форме сигнала будет соответствовать её цифровой код. Суть квантования и дискретизации сигнала представлена на рисунке 1.

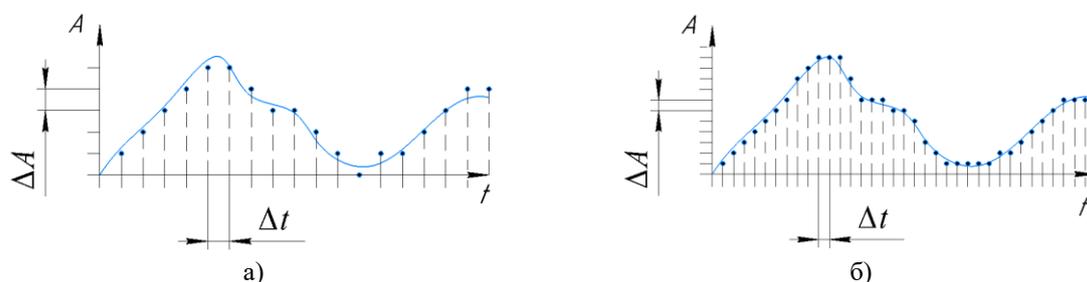


Рисунок 1. – Низкая разрядность квантования, низкая частота дискретизации (а); высокая разрядность квантования, высокая частота дискретизации (б), где Δt – шаг дискретизации, ΔA – шаг квантования

Счётчики/Таймеры могут быть применены в измерении времени, в подсчёте тактов или внешних импульсов, способны генерировать прерывания при опустошении или переполнении. Цифровые интерфейсы, такие как USART, I2C, 1-Wire, USB, позволяют системе передавать данные на компьютер или взаимодействовать с другими устройствами.

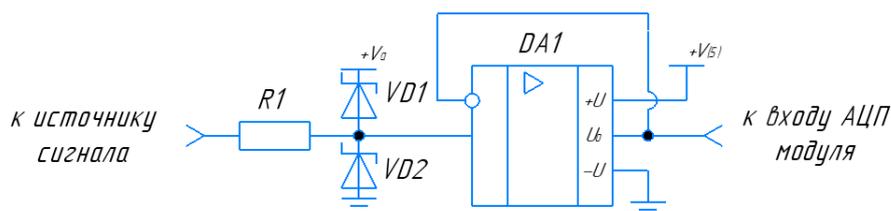


Рисунок 2. – Схемотехническое решение по защите входов АЦП модуля от перенапряжения

Важным в работе с микроконтроллерами является и защита их портов АЦП модуля от воздействия высокого напряжения, схемотехническое решение данной проблемы представлено на рисунке 2.

Принцип работы защиты от перенапряжения: от измеряющего устройства поступает сигнал на резистор R1, который практически не ослабляет его из-за наличия буфера повторителя напряжения построенного на операционном усилителе DA1, имеющего огромное входное сопротивление, протекающий сигнал через буфер попадает на вход АЦП модуля; в случае, если потенциал сигнала превышает (становится меньше) потенциала +Va (GND) на величину падения напряжения на диоде Шоттки, то диод VD1 (VD2) открывается и всё «лишнее» напряжение уходит через него [2].

Полученные микроконтроллером результаты измерений, для наглядности, необходимо отобразить на графике их изменения, после чего следует провести анализ. Примером программного обеспечения, которое может послужить для этих целей является «PhyZModule» [3].

На рисунке 3 представлен интерфейс разработанного программного обеспечения для осуществления измерения физических величин.

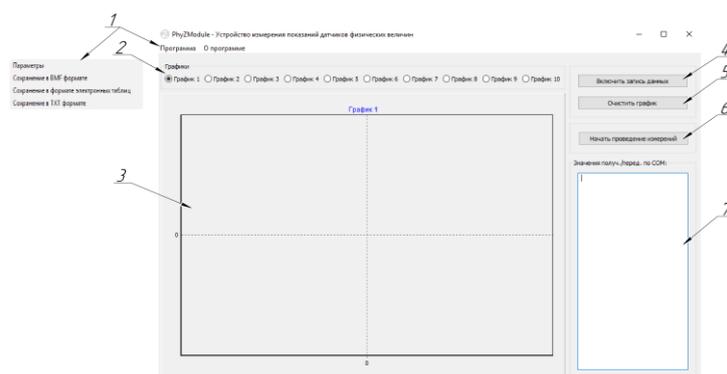
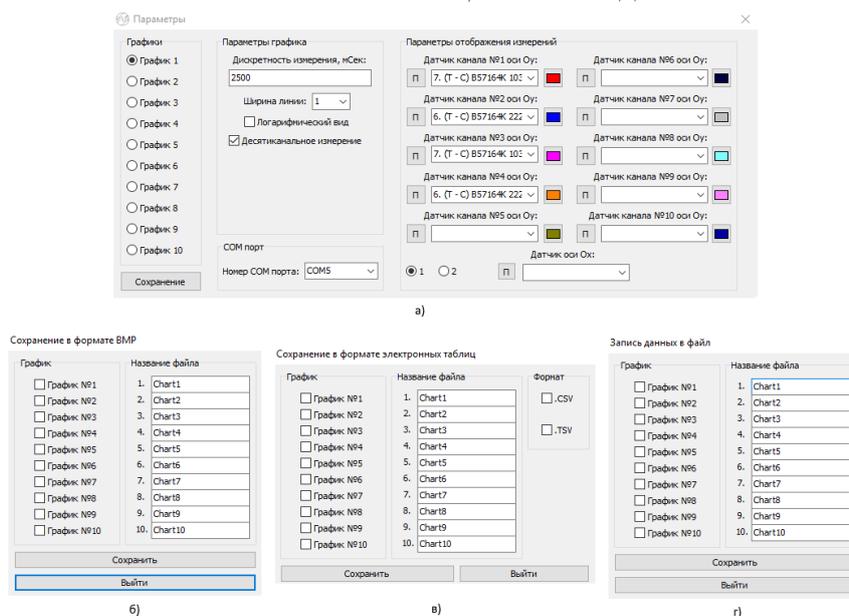


Рисунок 3. – Главное окно программы

Основным полем интерфейса является поле графиков 3, количество которых может достигать десяти. Данное поле используется для отображения всех настроенных для него измерений. Переключение между полями графиков происходит через изменение состояния переключателя 2. Кнопка 4 «Включить запись данных» предназначена для подтверждения использования выбранного поля графиков 3, после чего все измерения будут отображаться на нём. Кнопка 5 «Очистить график» используется при необходимости очистки поля графиков 3. Кнопка 6 «Начать проведение измерений» применяется для того, чтобы инициировать обмен данными между ПО и МК. После ее нажатия, вместо данной появится кнопка «Закончить проведение измерений», которая прерывает передачу данных. Меню 1 позволяет пользователю узнать основную информацию о программе в пункте «О программе», а также зайти в подменю программы в пункте «Программа», где существует возможность выбора формата сохранения и настроек параметров программы (рисунок 4).



**Рисунок 4. – Окно параметров графиков (а); окно сохранения в bmp формате (б);
окно сохранения в форматах tsv и csv (в); окно сохранения в txt формате (г)**

Для каждого из графиков существует возможность изменения параметров: дискретности отображения; использования десятиканального измерения; цвета линии; ширины линии; логарифмического вида, использования датчика и выбора АЦП порта для канала. Произвести изменения можно пройдя по пути «Программа-Параметры», рисунок 4а. Приложение позволяет выводить измеренную информацию в четырёх форматах: txt; bmp; tsv, csv.

Использование таких форматов, как tsv и csv, является универсальным выбором, позволяющим исследователю абстрагироваться от принятия решений об использовании определённого программного обеспечения для построения электронных таблиц и проведения дальнейших действий с выведенными данными. Данные форматы импортируются практически во все офисные пакеты, например, в такие как: «Microsoft Excel», «WPS SpreadSheets».

ЛИТЕРАТУРА

1. Змитрович, С.Ю., Вабищевич, С.А. Обработка аналоговых сигналов датчиковой аппаратуры. Информационно-коммуникационные технологии: достижения, проблемы, инновации (ИКТ-2018) [Электронный ресурс]: электронный сборник статей I международной научно-практической конференции, посвященной 50-летию Полоцкого государственного университета, Новополоцк, 14–15 июня 2018 г. / Полоцкий государственный университет. – Новополоцк, 2018. – 1 электрон. опт. диск (CD-ROM). – С.272-275.
2. Хоровиц, П. Искусство схемотехники / У. Хилл, П. Хоровиц. – Москва: Бинوم, 2015 – 704 с.
3. Змитрович, С.Ю. Автоматизированная система физических измерений/ С.Ю.Змитрович, С.А.Вабищевич, Д.Н.Шабанов// Вестник Полоцкого государственного университета. – 2019. – №4. – С.45-49.

УДК 003.26

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ СИСТЕМЫ «УМНЫЙ ДОМ»**И.В. ИСАКОВ***(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

В данной работе рассмотрим и проанализируем систему «Умный дом», её информационную безопасность.

Концепция умного дома включает в себя объединение различных современных технологий и решений, способных обеспечить комфорт и удобство получения услуг, безопасность граждан, рациональное потребление ресурсов. Вместе с тем из поля зрения приверженцев концепции умного дома часто выпадает безопасность самих элементов умного дома. Зачастую инфраструктура системы «умный дом» развивается намного быстрее, чем средства ее защиты, что вызывает живой интерес и оставляет большой простор для деятельности как исследователей, так и злоумышленников. Исходя из основной концепции системы умный дом, практики ее применения, а также необходимого и должного уровня безопасности и защищенности, как всей системы, так и ее компонентов антивирусное средство «умного дома» должно обеспечивать выполнение следующих важных функций:

- контроль появления на сервере умного дома любых посторонних файлов или программ;
- контроль несанкционированных подключений устройств к сети;
- контроль подключения устройств к беспроводным каналам передачи данных;
- контроль трафика между локальными сетями интеллектуального здания и непосредственно сервером автоматизированного управления;
- контроль взаимодействия сервера с сетью Интернет на предмет проникновения вирусного программного обеспечения;
- контроль сетевого оборудования на предмет DoS-атак;
- обеспечение проверки файлов, передаваемых в проводных и беспроводных сетях;
- выполнение эвристического поиска и наличия на сервере вирусных программ;
- контроль целостности системы умного дома, которая должна заключаться в проверке текущей конфигурации, управляющих процессов и хранимых данных.

На основе проведенного анализа существующих программных антивирусных средств можно сделать выводы, что создание антивирусной системы, способной обеспечивать комплексную защиту системы автоматизированного управления зданием является актуальной и важной задачей в краткосрочный период.

Решить данную задачу возможно с помощью целостного и системного подхода используя средства анализа контента и системы обнаружения атак. Средства анализа контента предназначены для контроля сетевого трафика с целью выявления нарушений политики безопасности. В настоящее время можно выделить два основных вида средств контентного анализа - системы аудита почтовых сообщений и системы мониторинга Интернет-трафика. Системы аудита почтовых сообщений предполагают сбор информации о SMTP-сообщениях, циркулирующих в АС, и её последующий анализ с целью выявления несанкционированных почтовых сообщений, нарушающих требования безопасности, заданные администратором. Так, например, системы этого типа позволяют выявлять и блокировать возможные каналы утечки конфиденциальной информации через почтовую систему. Системы мониторинга Интернет-трафика предназначены для контроля доступа пользователей к ресурсам сети Интернет. Средства защиты данного типа позволяют заблокировать доступ пользователей к запрещённым Интернет-ресурсам, а также выявить попытку передачи конфиденциальной информации по протоколу HTTP. Системы мониторинга устанавливаются таким образом, чтобы через них проходил весь сетевой трафик, передаваемый в сеть Интернет.

Системы обнаружения атак представляют собой специализированные программные или программно-аппаратные комплексы, предназначенные для выявления информационных атак на ресурсы системы посредством сбора и анализа данных о событиях, регистрируемых в системе. Система обнаружения атак включает в себя следующие компоненты:

- модули-датчики, предназначенные для сбора необходимой информации о функционировании системы. Иногда датчики также называют сенсорами;
- модуль выявления атак, выполняющий анализ данных, собранных датчиками, с целью обнаружения информационных атак;
- модуль реагирования на обнаруженные атаки;

– модуль хранения данных, в котором содержится вся конфигурационная информация, а также результаты работы средств обнаружения атак;

– модуль управления компонентами средств обнаружения атак.

Для решения проблемы обеспечения информационной безопасности «Умного дома» необходимо применение законодательных, организационных и программно-технических мер. Пренебрежение хотя бы одним из аспектов этой проблемы может привести к утрате или утечке информации.

ЛИТЕРАТУРА

1. Кусакин, И.И. Программно-аппаратный комплекс автоматизированного контроля целостности инфраструктуры жилых помещений для социального обеспечения. XV Международная телекоммуникационная конференция молодых ученых и студентов «МОЛОДЕЖЬ И НАУКА». Тезисы докладов. В 3-х частях. Ч. 3 – М.: НИЯУ МИФИ, 2012. – С. 156 – 157.

УДК 004.031.2

АВТОМАТИЗАЦИЯ БИЗНЕСА НА ОСНОВЕ CRM-СИСТЕМ

И.В. ИСАКОВ

(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)

В данной работе были проанализированы достоинства и недостатки различных CRM-платформ и разработан ряд программных инструментов, позволяющих автоматизировать малый и средний бизнес.

Введение. В настоящее время стандарты автоматизированных систем управления предприятием относятся не только к управлению потребностями производства: материалами и комплектующими, складскими запасами, производственным и технологическими процессами [1], как это было принято в методологии MRP, но и во всё большей степени к управлению персоналом, продажами, отношениям с клиентами (методология ERP). Поэтому CRM-системы стали важной самостоятельной составляющей корпоративных информационных систем. Потребность в CRM-системах возникла у малого и среднего бизнеса, вовсе не обязательно занимающегося производством. [2]

CRM (Customer Relationship Management, то есть «управление взаимоотношениями с клиентами») при помощи автоматизации процессов помогает эффективнее выстраивать диалог с клиентами и не допускать ошибок в работе, автоматизация помогает минимизировать ошибки «человеческого фактора», систематизировать данные о клиентах и сделках.

CRM берет всю рутину на себя: формирует документы по шаблону, отправляет sms клиентам, создает онлайн-отчеты по необходимым показателям, рассчитывает стоимость услуг, а также отслеживает важные даты и т.п. [3]

CRM-системы накапливают и хранят в себе огромные объемы информации. Поэтому эффективное использование этой информации является ключевой целью менеджеров первого звена, ведь определение текущих показателей в реальном времени и своевременное принятие решений позволит компании отслеживать малейшие изменения в потребностях клиентов, определять зависимости проводимых изменений на финансовые показатели компании.

Ещё одним немаловажным преимуществом CRM - систем является автоматизация рутинной работы и совмещение функций у текущих работников. CRM позволяет быстрее обучать и вводить новых сотрудников в бизнес-процессы компании, систематизировать работу менеджеров и тем самым повысить престиж компании в глазах текущих и новых сотрудников, отслеживает состояние дел, не позволяя сотрудникам забывать о делах и срывать сроки.

На данный момент существует множество различных CRM-платформ. Функционал многих систем подобен, из-за чего выбор подходящего программного продукта достаточно сложен, а разработка и внедрение таких сложных решений занимает длительное время. Также проблемой является то, что компании с опытом внедрения и консультирования по вопросам автоматизации стараются привязать своих клиентов к себе посредством постоянной необходимости в изменении и расширении предоставляемого решения. Таким образом, становятся актуальными гибкие платформы, позволяющие редактировать и изменять функционал без привлечения сторонних разработчиков.

Результаты и обсуждение. С учетом выше сказанного, основными проблемами являются: выбор CRM-платформы, наиболее отвечающей требованиям той или иной формы бизнеса, имеющей подходящий набор инструментов, позволяющей оптимальным образом настроить компоненты и не платить за избыточные и ненужные заказчику компоненты; их оптимальная настройка и интеграция с имеющимся у заказчика функционалом, который может быть реализован в разнородных приложениях. [4]

Данная проблема решалась в рамках научно-практического исследования на основе реальных бизнес-кейсов в практикоориентированном обучении IT-специалистов. [5]

При разработке и настройке систем учитывался набор функций, которые обязательно должны присутствовать в CRM: [6]

1. Модуль учета клиентов, в котором сохраняется вся история взаимодействия с клиентами.
2. Модуль для управления продажами, где указано, на каком этапе находится каждая сделка.
3. Автоматизация бизнес-процессов, которая позволяет не просто ставить задачи, но и отправлять sms-рассылки, менять данные об объектах, напоминать о приближении важных дат.
4. Аналитика и отчеты в реальном времени в виде наглядных графиков и диаграмм, а также таблиц в детальными данными.
5. Управление задачами выстроенное таким образом, чтобы руководитель моментально получал сообщения о выполненных и просроченных сотрудниками делами.

6. Интеграция с почтой, сайтом и IP-телефонией, чтобы все входящие заявки, по какому каналу они бы ни поступили, сразу фиксировались в CRM.

7. Интерфейс программирования API, который позволяет настроить интеграцию с 1С, корпоративным ПО, мобильными и другими приложениями.

В результате научно-практического исследования были решены следующие задачи и разработаны следующие программные средства:

1. Программное средство управления продажами автомобилями на базе CRM-системы. Целью данной работы является формирование требований, проектирование и разработка программного обеспечения по автоматизации процессов управления продажами для автомобильного дилерского центра. В настоящее время конкуренция в сфере продаж высока как никогда. Особенно, когда это касается высокотехнологичных композитных товаров с большой стоимостью – таких как автомобили. Современные продажи изменились абсолютно во всех аспектах. Покупательская способность выросла в разы благодаря развитой экономике и новым возможностям в виде кредитов, рассрочек и лизинга. По каждой модели автомобиля существуют видеозаписи с общей продолжительностью, переваливающей за десятки часов, скомбинированные маркетинговые материалы в виде инфографики, постеров и сравнений разбросаны по всемирной паутине на каждом углу. А условия поставки автомобилей в салоны практически одинаковые и зависят только от объемов продаж. Из-за чего разница в стоимости автомобилей и их комплектующих сравнялась, и современная конкуренция превратилась в борьбу сервиса. Кто быстрее, качественнее и эффективнее преподнесет предложение клиенту, тот и выиграет сделку и продвинет свой центр в лидеры автомобильного бизнеса. Программное средство управления продажами автомобилями было разработано на базе одной из популярных существующих платформ bpm'online.

2. Программное средство управления персоналом на платформе bpm'online. Целью данного проекта являлась разработка модификации раздела Активности для CRM-системы bpm'online. Важной особенностью данного функционала является его нестандартизированность, что позволяет предприятию гибко настроить рабочее время каждого сотрудника на каждый день. В приложении необходимо реализовать блокировку нерабочего временного пространства раздела Активности, для нужного контакта, по тем временным параметрам, которые были для него указаны. В заблокированную область не должны добавляться активности, как новые, так и старые (должна быть предусмотрена блокировка возможности изменения времени, выходящего за пределы рабочего времени). Система с подобным функционалом позволит составлять более точный график работы каждого сотрудника, а, следовательно, и повысит продуктивность, поскольку на каждую задачу выделяется определенное время, которое заполняет свободное пространство в графике, тем самым минимизируется перегруженность работника по задачам. Таким образом, в рамках данного проекта разработано программное средство, позволяющее строить уникальный график работы по каждому сотруднику в компании.

3. Клиентское приложение сервиса по доставке продуктов питания для платформы iOS. Объектом проекта является сфера общественного питания, а предметом реализуемого проекта является – процесс заказа продуктов питания. Актуальность данного проекта заключается в том, что создание сервиса по заказу продуктов питания поможет решить проблему нехватки мест в ресторанах или кафе, увеличить прибыль заведения, также поможет автоматизировать некоторые процессы ресторанов или кафе. Целями проекта являлись разработка мобильного сервиса по заказу продуктов питания ресторанов или кафе, оптимизация работы с клиентами ресторана или кафе посредством доставки, предоставление им возможности удаленного выбора блюд и дальнейшего заказа продуктов питания, а также оплаты с помощью банковской карты. Были сформированы требования проектирования и разработки программного обеспечения клиентского сервиса по доставке продуктов питания. Для достижения поставленных целей в работе определены следующие задачи: моделирование бизнес-процессов; проектирование структуры приложения и интерфейсов реализуемых классов; построение функциональной модели; построение информационной модели, с приведением ее к третьей нормальной форме; анализ и выбор шаблонов проектирования; программная реализация алгоритма приложения; реализация пользовательского интерфейса в виде клиентского мобильного приложения на платформе iOS с использованием Socoa; тестирование разработанной системы.

4. Автоматизация работы менеджера компании-реселлера на предприятии «Реланс-М». Целью данного проекта является модернизация и внедрение CRM-системы bpm'online на предприятие «Реланс-М». В данной системе должны быть автоматизированы все бизнес-процессы предприятия для улучшения качества обслуживания клиентов и увеличения продаж. Для решения поставленной задачи были проанализированы аналоги CRM-систем и сформированы требования к проектируемому программному средству, выполнено моделирование предметной области, разработка функциональных требований, проектирование архитектуры и графического интерфейса программного средства, также выполнено тестирование и разработано руководство пользователя.

Заключение. В результате научно-практического исследования были проанализированы преимущества и недостатки различных CRM-платформ, определены и разработан ряд программных средств, обеспечивающих автоматизацию предприятий малого и среднего бизнеса.

ЛИТЕРАТУРА

1. Бакунова О.М., Бакунов А.М.; Калитеня И.Л.; Образцова О.Н.; Ухналёв Р.Ю., Биркос В.А., Балабко А.В., Пазушко В.В. Моделирование и управление транспортными потоками предприятия в системе 1С International Trends in Science and Technology: Proceedings of the VI International Scientific and Practical Conference, Warsaw, September 30 2018. – Warsaw, 2018. – Vol.1. – P. 65 – 66.
2. Бакунова О.М., Анохин Е.В., Палуйко А.Ф., Александрович Е.Н., Антонов Е.Д., Ситник М.Ю., Гречко И.С. Современные информационные технологии в системе управления. International Journal of Innovative Technologies in Economy. – 2018. – 4(16) – Pp. 52 – 54.
3. Бакунова О.М., Анохин Е.В., Палуйко А.Ф., Александрович Е.Н., Антонов Е.Д., Ситник М.Ю., Гречко И.С., Кабаков Д.М. Применение электронного документооборота в программе 1С. International Journal of Innovative Technologies in Economy. – 2018. – 4(16) – Pp. 64 – 66.
4. Бакунова О.М., Калитеня И.Л., Уласович В.Ю., Тимофеев Д.О., Ухналев Р.Ю., Михаленко В.А. Интеграция информационных систем с использованием Mule ESB. Web of Scholar. – 2018. – 4(22), Vol.1. – Pp. 4 – 7.
5. Образцова О.Н. Бакунова О.М., Кугач Д.М., Хомяков А.В. Практикоориентированное обучение в сфере информационных технологий в БГУИР и сотрудничество вуза с ведущими компаниями IT. Проблемы современного образования: материалы VIII международной научной конференции, 10-11 сентября 2017. – Прага: Vědecko vydavatelské centrum «Sociosféra-CZ», 2017 – С.38 – 41.
6. Электронный ресурс «Онлайн CRM-система SalesapCRM. Автоматизация продаж». – Режим доступа: <https://salesap.ru>. – Дата доступа: 12.06.2019.

УДК 620.92 (07)

СПОСОБЫ МОНТАЖА СОЛНЕЧНОГО КОЛЛЕКТОРА**В.К. КОВАЛЕВСКИЙ***(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)*

В статье рассмотрены способы монтажа солнечного коллектора. Выделены важные параметры, которые необходимо учитывать при установке солнечных коллекторов. Описаны основные рекомендации, которым необходимо придерживаться при монтаже солнечного коллектора. Разработан ряд указаний для правильного монтажа солнечного коллектора.

Солнечные коллекторы – это конструктивно не сложные устройства, но наиболее эффективного режима работы можно добиться лишь при их правильной установке. Монтажные операции включают в себя целый комплекс мероприятий, связанных с расчетом, выбором и установкой солнечных коллекторов. При этом перечисленные критерии во многом зависят от индивидуальных особенностей объекта, его климатического расположения и условий эксплуатации теплогенерирующего оборудования.

Очень важными параметрами при установке солнечных коллекторов являются:

- угол наклона нагревательной поверхности (подбирается в зависимости от времени года и географического расположения), диапазон углов наклона может варьироваться от 30 до 45;

- ориентация прибора относительно сторон света (приоритетным всегда остается южное направление).

Необходимо помнить, что при монтаже солнечного коллектора на крышу строения, нужно учитывать угол наклона самой кровли и соответственно внести необходимые поправки при установке угла наклона коллектора на такой поверхности. При этом прочность кронштейнов, удерживающих гелиоустановку в рабочем положении должна быть достаточно высокой. Дабы надежно фиксировать солнечный коллектор (обладающий большой парусностью) при порывах ветра и атмосферных осадках. На сегодняшний день различают несколько основных способов монтажа солнечных коллекторов:

Наклонный (при установке на скатную конструкцию крыши);

Горизонтальный (осуществляется на плоской конструкции крыши);

Свободный (для установки в любых местах с применением опорной конструкции).

При этом, какому бы методу монтажа не отдавалось бы предпочтение, следует избегать установки подобного оборудования в затененных местах.

Выполняя монтаж солнечных коллекторов желательно придерживаться следующих рекомендаций.

Во-первых, отходящие патрубки гелиоустановки должны располагаться в вертикальном положении, чтобы не создавалось препятствий для циркуляции теплоносителя.

Во-вторых, разрабатывая схему подключения коллекторов к общей гидравлической схеме необходимо учитывать, что движение теплоносителя в коллекторе должно происходить снизу вверх. Очень важно, для исключения перегрева сухого коллектора (способного вызвать его деформацию), производить все монтажные и пуско-наладочные операции при накрытии панели плотным непрозрачным материалом.

В-третьих, без применения компенсаторов теплового расширения рекомендуется монтировать в один ряд не более трех солнечных коллекторов. При этом необходимо избегать взаимного затенения коллекторов, устанавливая их на минимальном расстоянии друг от друга.

Для эффективной работы солнечного коллектора в режиме естественной циркуляции теплоносителя, рекомендуется производить укладку трубопроводов с уклоном в 0.01° по ходу движения антифриза. Если же предполагается установка циркуляционного насоса, то этот показатель должен составлять 0.002° . Но какой бы длины не были трубопроводы, их максимальный уклон не должен превышать 10мм на всю линию. Трубопроводы должны монтироваться таким образом, чтобы имелась возможность производить слив теплоносителя, и его закачку используя запорную арматуру, а также выполнять сброс воздуха в самой верхней точке.

Технология заправки коллектора теплоносителем имеет некоторые особенности. Итак, после того, как солнечный коллектор распакован и очищен от пыли и грязи, может быть произведена его фиксация и подключение к гидравлической схеме. Заполнение коллектора теплоносителем может производиться только в холодном состоянии (дабы избежать внутреннего теплоудара). Скорость подачи воды при заполнении коллектора должна быть невысокой, причем поступление должно происходить снизу. Это делается с той целью, чтобы обеспечить выход всех воздушных пробок из системы. После запуска гелиоустановки в работу, в течение первого времени рекомендуется контролировать уровень теплоносителя в системе (он должен полностью заполнять всю установку) и при необходимости производить его добавление.

После того как произведены все монтажные и пусконаладочные операции необходимо выполнить ряд диагностических мероприятий:

- проверить герметичность всех соединений и состояние теплоизоляции трубопроводов;
- проконтролировать и в случае необходимости очистить поверхность коллектора от загрязнения, чтобы обеспечить максимальное светопропускание;
- замену элементов производить согласно рекомендаций изготовителя, не допуская наличия царапин, сколов и трещин в приборе, также недопустимо проявления масляных и жирных пятен на светопоглощающей поверхности элемента.

ЛИТЕРАТУРА

1. Ширман, Я. Д. Теоретические основы радиолокации / Я.Д. Ширман, В.Н. Голиков, И.Н. Бусыгин, Г.А. Костин. – М.: Советское радио, 1970. – С. 559.
2. Саплин, Л.А. Энергоснабжение сельскохозяйственных потребителей с использованием возобновляемых источников: Учебное пособие / Л.А. Саплин, С.К. Шерьязов, О.С. Пташкина-Гирина, Ю.П. Ильин. – Челябинск: ЧГАУ, 2000.
3. Метрология и радиоизмерения. Учебно-методический комплекс для студ. спец. 1-39-01-01 «Радиотехника» / В.Ф. Янушкевич. – Новополец, ПГУ, 2010. – С. 304.

УДК 620.92 (07)

УСТРОЙСТВО СЛЕЖЕНИЯ ЗА СОЛНЕЧНЫМИ ЛУЧАМИ

В.К. КОВАЛЕВСКИЙ*(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)*

В статье описана необходимость внедрения системы слежения за солнечными лучами в систему управления солнечным коллектором. Определена схема электрическая-принципиальная устройства слежения за солнечными лучами. Описана элементная база схемы, а также описан принцип её работы.

Очень часто солнечные коллекторы устанавливаются жестко и стационарно. Это просто, но не очень эффективно, так как в течение дня солнце перемещается по небу и количество тепловой энергии, получаемое солнечным коллектором, меняется. Максимальная отдача солнечного коллектора возможна только когда свет падает на панель перпендикулярно ее плоскости. Для того, чтобы солнечные лучи падали на коллектор всегда перпендикулярно используются системы слежения за солнечными лучами разной конструкции и разного уровня сложности. Такая система хоть и усложняет монтаж солнечного коллектора, но позволяет использовать его максимальный КПД. Солнечные коллекторы все еще не дешевы, и мы, вкладывая в них свои средства, наверняка хотим реализовать весь их потенциал.

Часто простейшие системы слежения могут поворачивать панель только по одной оси. Такие устройства имеют в своем составе пару фотодатчиков и пару компараторов, которые управляют одним электродвигателем, включенным в диагональ простейшего Н-моста. Такие устройства конечно тоже повышают эффективность работы солнечного коллектора, но делают это не совсем эффективно, так как коллектор работает только в пределах одной оси. В этой статье описывается очень простая система слежения за солнечными лучами, которая, тем не менее, умеет перемещать панель по двум осям, что повышает эффективность работы солнечного коллектора. Во время работы система непрерывно пытается расположить коллектор так, чтобы ее плоскость всегда была перпендикулярна падающим солнечным лучам и получить максимальную отдачу тепла от солнца. Устройство слежения следует за солнцем с рассвета до последних лучей заката и автоматически возобновляет свою работу на следующее утро. Схема электрическая-принципиальная устройства слежения за солнечными лучами представлена на рисунке 1.

Схема устройства слежения за солнечными лучами содержит микросхему LM339, которая представляет собой четыре аналоговых компаратора в общем корпусе. Двумя моторами управляет специализированная микросхема L293D - сдвоенный Н-мост. Кроме микросхем принципиальная схема содержит несколько дискретных элементов. В качестве датчиков интенсивности освещения используются фоторезисторы LDR1 – LDR4. Эти фоторезисторы определяют положение солнечного коллектора относительно лучей солнца.

Рассмотрим алгоритм работы системы слежения за солнечными лучами на примере ее части, отвечающей за перемещения по оси X. Если фоторезистор LDR2 получает больше света чем LDR1, то сопротивление LDR2 становится меньше сопротивления LDR1. На входах компараторов A1 и A2 (4, 7) появляется более высокий уровень напряжения. При этом на выходе компаратора A2 (1) появляется высокий уровень напряжения. Мотор M1 начинает вращаться в одном из направлений (скажем, против часовой стрелки), поворачивая солнечную панель.

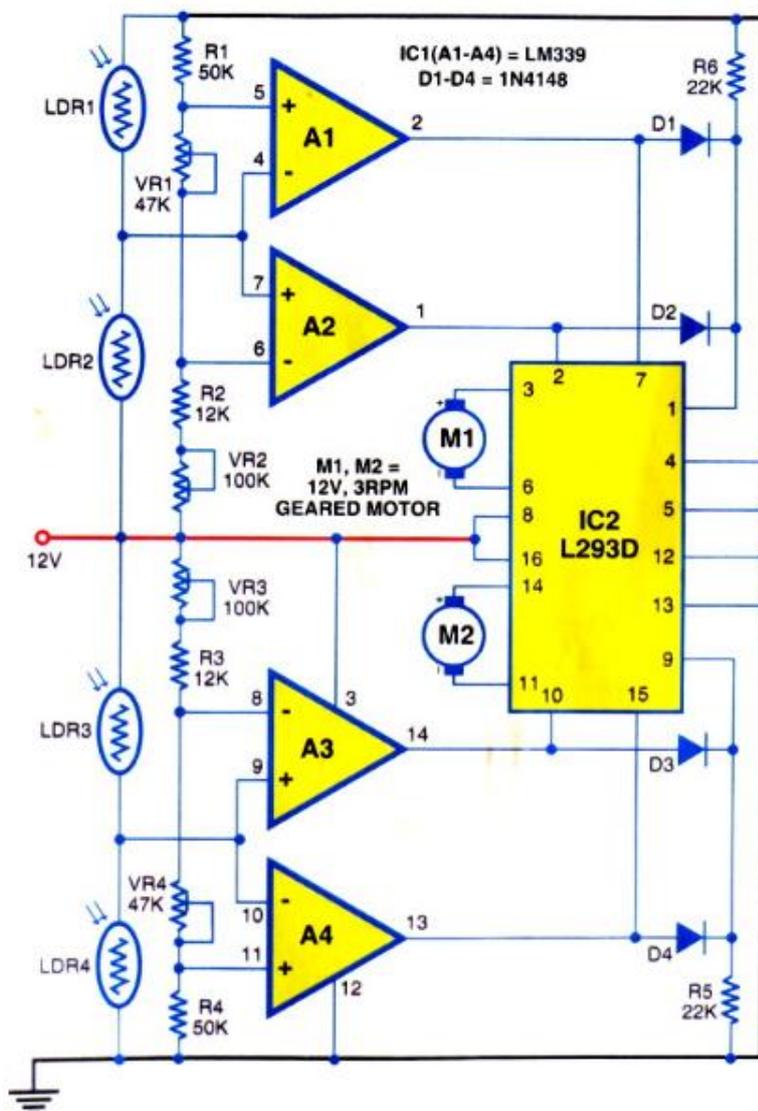


Рисунок 1. – Схема электрическая принципиальная устройства слежения за солнечными лучами

Если LDR1 получает света больше чем LDR2, то его сопротивление становится меньше, чем сопротивление LDR2, тем самым уменьшая напряжение на входах компараторов (4, 7). На выходе (2) компаратора A2 появляется высокий уровень и мотор начинает вращаться в противоположном направлении (скажем, по часовой стрелке). Точно таким же образом работает слежение по оси Y.

ЛИТЕРАТУРА

1. Функциональные схемы и их виды [Электронный ресурс]. –Режим доступа: <http://www.moysatapparat.ru/ustanovka/ustanovka.html>
2. Ширман, Я. Д. Теоретические основы радиолокации / Я. Д. Ширман, В. Н. Голиков, И. Н. Бусыгин, Г. А. Костин. – М.: Советское радио. – 1970. – С. 559.
3. Метрология и радиоизмерения. Учебно – методический комплекс для студ. спец. 1 – 390101 «Радиотехника» / В. Ф. Янушкевич. – Новополоцк, ПГУ. – 2010. – С. 304.
4. Петровский, А. А. Микропроцессорная техника. Учебное пособие по курсовому проектированию / А. А. Петровский, Г.В. Таранов. – Минск, БГУИР. – 2005. – 51 с.
5. Микроконтроллеры AVR [Электронный ресурс]. – Режим доступа: <http://www.electromonter.info/handbook/13/ElectromagneticRelay>.

УДК 004.413.2

**ПРИНЦИПЫ РАЗРАБОТКИ И ШАБЛОНЫ ПРОЕКТИРОВАНИЯ ПРИЛОЖЕНИЙ
ПОД ОПЕРАЦИОННУЮ СИСТЕМУ IOS****Е.В. КОНЯЕВ***(Представлено: И.С. РУСЕЦКИЙ)*

В статье рассматриваются основные принципы разработки приложений под операционную систему iOS, подробно рассмотрены распространенные шаблоны проектирования и их особенности.

Ключевые слова: мобильное приложение, разработка, архитектура, Swift, Apple.

С развитием современных технологий, которые позволяют создавать различные мобильные устройства, рынок программных продуктов получает мощнейший стимул к расширению и развитию. Современную мобильную технику люди носят с собой всегда и везде. За основу взята мобильная ОС iOS и рассмотрение принципов разработки программных продуктов для мобильных устройств будет производиться на ее примере. Для грамотного создания приложения необходимо учитывать принципы простоты, удобства и интуитивно-понятного управления. Исходя из входных параметров выбирается наиболее подходящий шаблон архитектуры, влияющий на последующее использование приложения конечным пользователем.

Написание приложения под ОС iOS осуществляется на компьютерах и ноутбуках производства компании Apple в рабочей среде X-Code, имеется возможность написания приложения на двух языках: Objective-C или Swift. Остановимся подробнее на среде разработки Xcode — это приложение для Mac, предназначенное для разработки других приложений для Mac и iOS. Его можно загрузить бесплатно из App Store для Mac. Xcode тесно интегрирован с framework Cocoa, который состоит из библиотек, API, и сред, которые формируют слой разработки для всех Mac OS X. Данный набор инструментов включает в себя: Xcode IDE (для написания кода, создания и отладки приложений), Interface Builder (для разработки пользовательского интерфейса), инструменты для анализа поведения и производительности и другое.

Рассмотрим подробнее языки программирования [1]. Начнем с Objective-C, который является расширением языка программирования C и предоставляет объектно-ориентированные возможности, а также динамическую систему времени выполнения. В нем имеются привычные элементы, например, типы данных (int, float и т.д.), структуры, функции, указатели и управляющие конструкции (while, if...else и оператор for). Также имеется доступ к функциям стандартной библиотеки программирования C, например, к тем, которые объявлены в `stdlib.h` и `stdio.h`. По аналогии с интерфейсами в Java, в Objective C есть протоколы. Протокол определяет набор селекторов, которые должен поддерживать объект, реализующий протокол.

Swift является более молодым и, на данный момент, стремительно развивающимся языком программирования. Он разрабатывается компанией Apple, как будущая замена Objective-C [2]. Swift задумывался как быстрый и эффективный язык программирования с откликом в реальном времени, который легко можно вставить в готовый код Objective-C. Swift берет довольно многое из Objective-C, однако он определяется не указателями, а типами переменных, которые обрабатывает компилятор. Swift работает с фреймворками Cocoa и Cocoa Touch и совместим с основной кодовой базой Apple, написанной на Objective-C. Swift разрабатывается как более безопасный язык в сравнении с Objective-C, к тому же Swift, в отличие от Objective-C, легко читаемый язык, как и Python. На данный момент эти языки взаимно дополняют друг друга, но основная часть приложений написана на Objective-C.

Важным аспектом разработки мобильного приложения является выбор правильного шаблона проектирования. В первую очередь рассмотрим особенности хорошей архитектуры приложения: сбалансированное распределение функций между действующими объектами, тестируемость приложения и удобство использования.

В настоящее время существует много вариантов архитектуры шаблонов проектирования, наиболее популярными из которых являются:

- Model-View-Controller (MVC);
- Model-View-Presenter (MVP);
- Model-View-ViewModel (MVVM).

MVC – схема использования нескольких модулей, с помощью которых модель приложения, пользовательский интерфейс и взаимодействие с пользователем разделены на три отдельных компонента таким образом, чтобы модификация одного из компонентов оказывала минимальное воздействие на остальные [3]. Традиционное представление MVC модели представлено на рисунке 1.

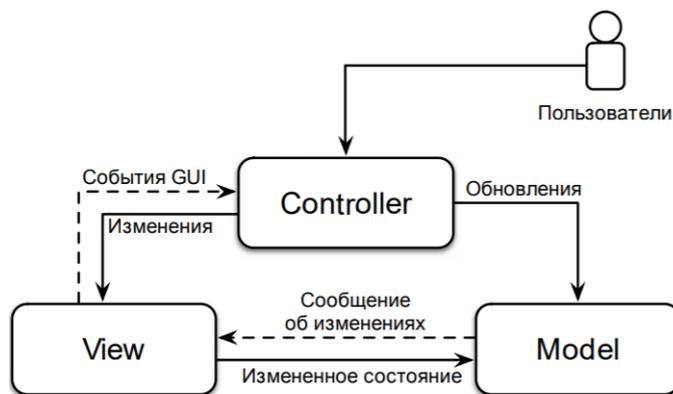


Рисунок 1. – Традиционное представление MVC модели

Модель (Model) содержит основные данные, например, переменные, подключения к внешним RSS-каналам или изображения, подробные функции и числовую информацию. Вид (View) отвечает за стиль отображения информации на дисплее. В качестве представления выступает экран с графическими элементами на котором, например, можно изменить стиль или удалить элементы. Контроллер (Controller) управляет запросами пользователя и использует модель для реализации необходимой реакции.

MVP является шаблоном проектирования производным от MVC, структура шаблона показана на рисунке 2. Основной задачей MVP является сделать вид (view) повторно используемым [4].

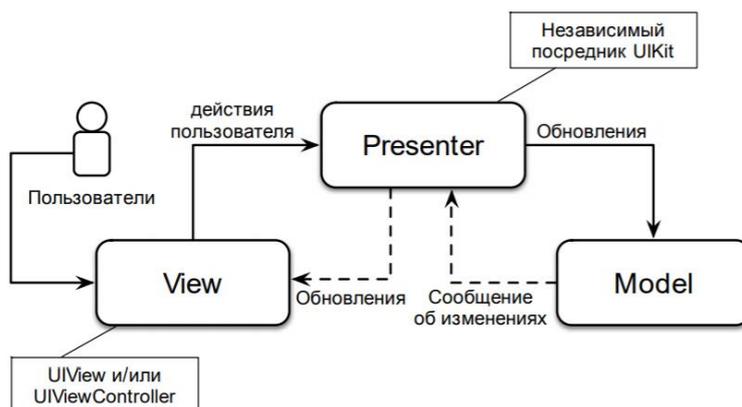


Рисунок 2. – Структура шаблона MVP

MVC и MVP парадигмы очень похожи друг на друга, но их применение зависит от условий использования. Для MVC — это там, где вид (view) обновляется каждый раз по какому-либо событию, а для MVP, когда вид (view) не нужно каждый раз пересоздавать.

Последний из выбранных шаблонов проектирования, MVVM, появился для обхода ограничений паттернов MVC и MVP, и объединяющий некоторые из их сильных сторон. Эта модель впервые появилась в составе фреймворка Small Talk в 80-х, и была позднее улучшена с учетом обновленной модели MVP. Схема работы MVVM представлена на рисунке 3.

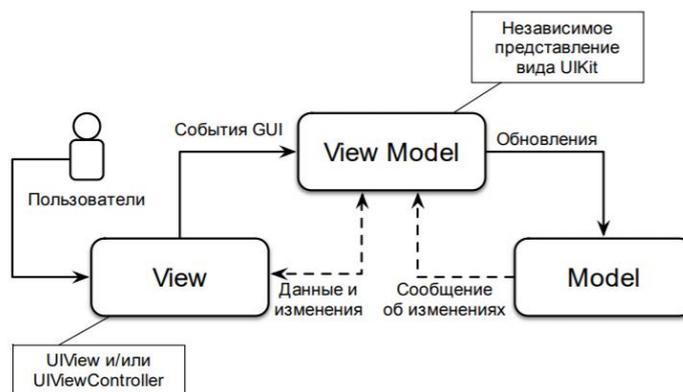


Рисунок 3 – Модель MVVM

MVVM является очень привлекательным вариантом, так как он сочетает в себе преимущества вышеупомянутых подходов, и, кроме того, он не требует дополнительного кода для вида (view) и тестируемость находится на хорошем уровне [5].

Все шаблоны проектирования являются актуальными на данный момент. Выделим некоторые моменты для выбора шаблона проектирования: MVVM используется в ситуации, когда возможно связывание данных без специальных интерфейсов, MVP используется, когда невозможно связывание данных, MVC используется, когда связь между видом (View) и другими частями приложения невозможна.

Заключение. В данной статье были рассмотрены основной инструментарий создания приложения на платформе iOS, используемые для этого языки программирования и их особенности, а также основные шаблоны проектирования приложений и их структурные отличия.

ЛИТЕРАТУРА

1. Live Typing [Электронный ресурс]. Режим доступа: <https://livetyping.com/ru/blog/na-chem-pishut-prilozhenija-pod-ios>. – Дата доступа: 28.09.2020.
2. GeekBrains. [Электронный ресурс]. Режим доступа: https://geekbrains.ru/posts/swift_vs_obj_c. Дата доступа: 28.09.2020.
3. Stfalcon [Электронный ресурс]. Режим доступа: <https://stfalcon.com/ru/blog/post/ios-patterns>. Дата доступа: 28.09.2020.
4. Habr [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/215605/> Дата доступа: 29.09.2020.
5. ProfessorWEB [Электронный ресурс]. Режим доступа: https://professorweb.ru/my/WPF/documents_WPF/level36/36_5.php Дата доступа: 20.09.2020.

УДК 004.41

СРАВНЕНИЕ БЫСТРОДЕЙСТВИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ПОД ПЛАТФОРМУ IOS

Е.В. КОНЯЕВ

(Представлено: И.С. РУСЕЦКИЙ)

В статье рассматриваются показатели быстродействия языков программирования Swift и Objective-C под платформу iOS.

Ключевые слова: Objective-C, Swift, тестирование, сортировка.

Одной из главных составляющих проектирования под платформу iOS является выбор языка программирования для создания мобильных приложений. Существует два таких языка, со своими характерными особенностями, преимуществами и недостатками. Одним из ключевых параметров является быстродействие – скорость совершения операций с помощью кода написанного на данных языках программирования.

Сравнение быстродействия:

Быстродействие языков тестировалось путем замера скорости сортировок, алгоритмы которых были написаны на всех трех языках. Главный интерес представляют результаты по Swift и Objective-C, классический Си использовался как своего рода критерий. Скорость выполнения написанных на нем программ благодаря низкому, приближающемуся к машинному, уровню абстракции, является очень высокой по сравнению с языками высокого уровня (к категории которых относятся Swift и Objective-C) [1]. Те в меньшей степени учитывают особенности компьютерного железа, удобнее в написании прикладных программ, но по этой же причине медленнее работают.

Результаты испытаний варьируют в зависимости от заданных уровней оптимизации при компиляции исходного кода. Swift значительно уступает Objective-C при отключенной оптимизации, но превзошел его на стандартной оптимизации от 6 до 18 раз, а на высокой — от 7 до 35 раз, в зависимости от типа сортировки [2]. Это очень впечатляющий прирост быстродействия, демонстрирующий явное превосходство Swift по сравнению с Objective-C. На рисунке 1 представлено окно с установкой уровней оптимизации.

▼ Apple LLVM 9.0 - Code Generation	
Setting	FoodTracker
Link-Time Optimization	No ↕
▼ Optimization Level	<Multiple values> ↕
Debug	None [-O0] ↕
Release	Fastest, Smallest [-Os] ↕
Optimization Profile File	/Users/yar/Documents/Xcode/
Relax IEEE Compliance	No ↕
Use Optimization Profile	No ↕

Рисунок 1. – Настройка уровней оптимизации.

Для получения точных данных о быстродействии языков программирования используются различные методы сортировки, такие как:

- Сортировка пузырьком – сравниваются соседние значения и если нужно неравенство не выполняется, то элементы меняются;
- Сортировка выборкой – выполняем поиск наименьшего элемента массива, вставляя его в начало, повторяем эти действия для оставшегося массива;
- Быстрая сортировка – берется случайный элемент массива, удаляется из массива, и массив разбивается на 2 – в первом те элементы что меньше выбранного, во втором остальные. И к каждому из полученных массивов применяются те же действия, пока размер массива не станет нулевым;

— Сортировка слиянием – массив делится пополам, и к каждой половине применяется сортировка слиянием, и после происходит слияние отсортированных половинок за линейное время;

— Стандартная сортировка, которая предполагается стандартной библиотекой.

Быстродействие языков тестировалось путем замера скорости сортировок, алгоритмы которых были написаны на всех трех языках. Главный интерес представляют результаты по Swift и Objective-C, классический Си использовался как своего рода критерий. Скорость выполнения написанных на нем программ благодаря низкому, приближающемуся к машинному, уровню абстракции, является очень высокой по сравнению с языками высокого уровня (к категории которых относятся Swift и Objective-C). Те в меньшей степени учитывают особенности компьютерного железа, удобнее в написании прикладных программ, но по этой же причине медленнее работают.

Таблица 1. – Сравнение быстродействия языков

		Сортировка пузырьком	Сортировка выборкой	Быстрая сортировка	Сортировка слиянием	Стандартная сортировка
Swift	None [-O0]	7,572	2,531	0,077	0,033	0,009
	Fast [-O]	0,326	0,172	0,0222	-	0,0007
	[-Ofast]	0,339	0,180	0,0214	-	0,00007
Obj-C	None [-O0]	2,636	1,189	0,030	0,035	0,0068
	Fastest [-O3]	2,205	0,972	0,029	0,034	0,0069
	[-Ofast]	2,203	0,971	0,030	0,035	0,007
C	None [-O0]	0,457	0,155	0,0044	0,0035	0,0015
	Fastest [O3]	0,163	0,188	0,0038	0,0027	0,0013
	[-Ofast]	0,163	0,186	0,0038	0,0030	0,0013

На Swift при переходе с нулевого уровня оптимизации на оптимизацию внутри каждого файла (средний уровень оптимизации) время работы уменьшается в десятки раз, и в сортировке выборкой он близок к C [3], в то время как на Objective-C время практически не меняется. Здесь явно видно время исполнения, что Swift быстрее чем Objective-C, а C быстрее чем Swift, разве что при использовании стандартной сортировки Swift в 2 раза быстрее.

Заключение. В данной статье было рассмотрено быстроедействие трёх языков программирования, из них два ключевых для создания приложений на платформе iOS. Произведя измерения и анализ полученных результатов, стало понятно, что наиболее производительным и оптимизированным для использования при создании мобильных приложений является язык Swift.

ЛИТЕРАТУРА

1. An analysis of sorts between Swift and Objective-C [Электронный ресурс] // jessesquires.com: [сайт]. URL: <https://www.jessesquires.com/blog/apples-to-apples/>. - Дата доступа: 27.09.2020
2. An analysis of sorts between Swift and Objective-C [Электронный ресурс] // jessesquires.com: [сайт]. URL: <https://www.jessesquires.com/blog/apples-to-apples-part-two/>. - Дата доступа: 27.09.2020.
3. An analysis of sorts between Swift and C [Электронный ресурс] // jessesquires.com: [сайт]. URL: <https://www.jessesquires.com/blog/apples-to-apples-part-three/>. - Дата доступа: 27.09.2020.

УДК 621.371.39

РАСПРОСТРАНЕНИЕ РАДИОИМПУЛЬСНЫХ СИГНАЛОВ
НАД УГЛЕВОДОРОДНЫМИ ЗАЛЕЖАМИ

А. П. КОЧАНОВ

(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)

В статье проведен анализ комбинационных составляющих тензора диэлектрической проницаемости в режиме взаимодействия радиоимпульсных сигналов с анизотропными средами над углеводородными залежами. Используется вертикальная поляризация электромагнитных волн. Исследовано влияние удельной проводимости среды на распространение радиоимпульсных сигналов. Даны рекомендации по использованию оптимальных характеристик зондирующих сигналов. Результаты исследований могут быть использованы в поисковой геофизике.

Ключевые слова: электромагнитная волна, анизотропная среда, углеводородные залежи.

Введение. Актуальность рассматриваемых в настоящей работе задач заключается в усовершенствовании существующих электромагнитных методов георазведки и разработке новых методов поиска, идентификации месторождений нефти и газа (углеводородов), являющихся стратегическим видом полезных ископаемых. Методы поиска углеводородных залежей (УВЗ) основаны на анализе электрофизических и электрохимических процессов в анизотропных средах (АС) над залежами углеводородов и оптимизации частотных, поляризационных, амплитудных, модуляционных и других параметров зондирующих сигналов с учетом специфических характеристик УВЗ [1-4].

Аналогия исследуемой среды над залежью нефти и газа с плазмоподобным образованием позволяет использовать для изучения взаимодействия электромагнитных волн (ЭМВ) с УВЗ существующие решения при изучении плазмы и плазмоподобных сред. Установлено, что наиболее оптимальным для исследования сред над УВЗ является квазигидродинамический подход с использованием многочастичных электронно-ионных токов и феноменологически заданных частот столкновений, ввиду отсутствия необходимости учета точных пространственно-временных перемещений частиц [5-7].

Результаты и их обсуждение. В статье проведен анализ дисперсионных характеристик диэлектрической проницаемости в режиме взаимодействия радиоимпульсных сигналов с анизотропными средами над углеводородными залежами. Методика исследований заключается в облучении исследуемого профиля электромагнитной волной на фиксированной частоте, приеме отраженного сигнала. Измеряют напряженность электрического поля отраженного сигнала в точках измерения исследуемого профиля и по аномальным значениям напряженности электрического поля отраженного сигнала определяют границу углеводородной залежи.

Представляет интерес анализ частотных характеристик комбинационных составляющих

$$\begin{aligned}\dot{\mathcal{E}}_R(\omega) &= \dot{\mathcal{E}}_1 + \dot{\mathcal{E}}_2 = \text{Re } \mathcal{E}_R + j \text{Im } \mathcal{E}_R \\ \dot{\mathcal{E}}_L(\omega) &= \dot{\mathcal{E}}_1 - \dot{\mathcal{E}}_2 = \text{Re } \mathcal{E}_L + j \text{Im } \mathcal{E}_L.\end{aligned}\quad (1)$$

На рисунке 1 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с правой поляризацией $\mathcal{E}_R = \Psi(\sigma)$.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характера сред над УВЗ.

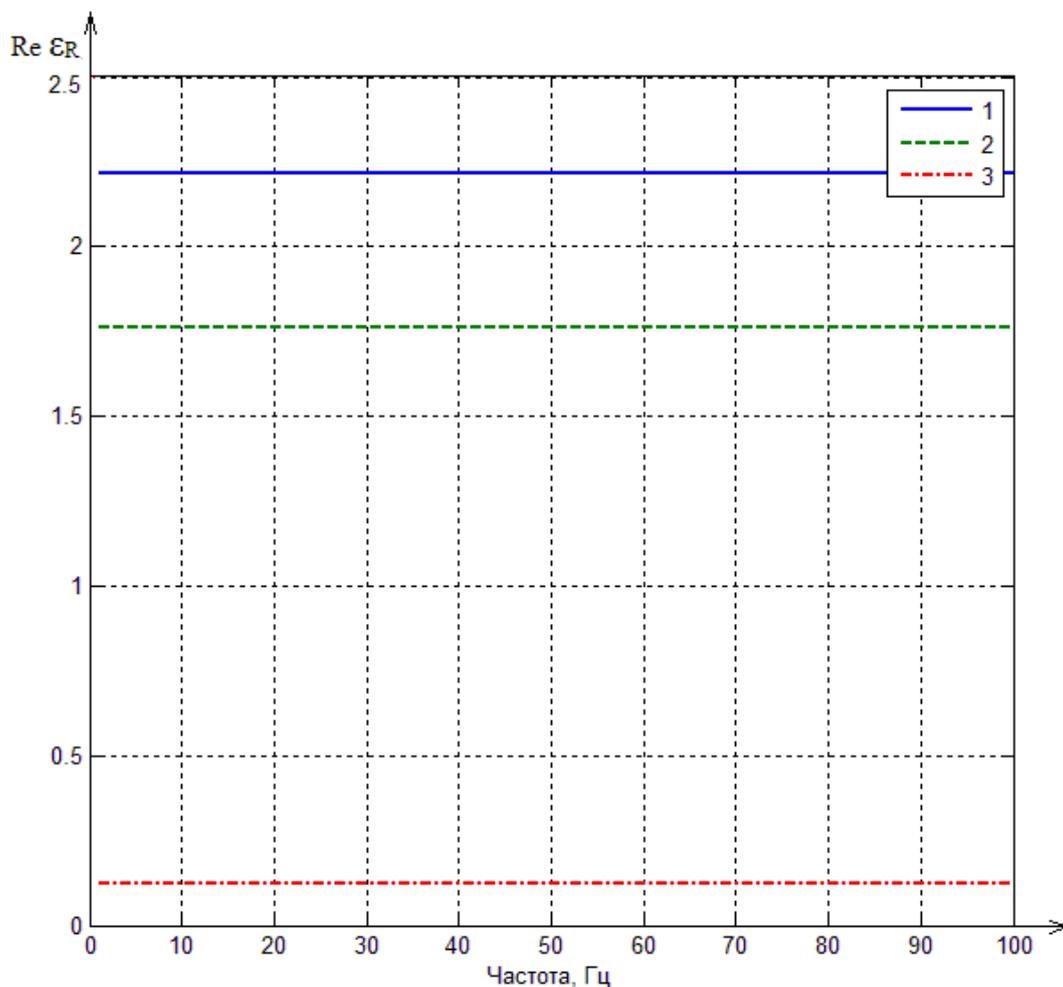


Рисунок 1. – Зависимости $Re \epsilon_R = \psi(\sigma)$. 1 – для $\epsilon_r = 25$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 2$

На рисунке 2 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с левой поляризацией $\epsilon_L = \Psi(\sigma)$.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характера сред над УВЗ.

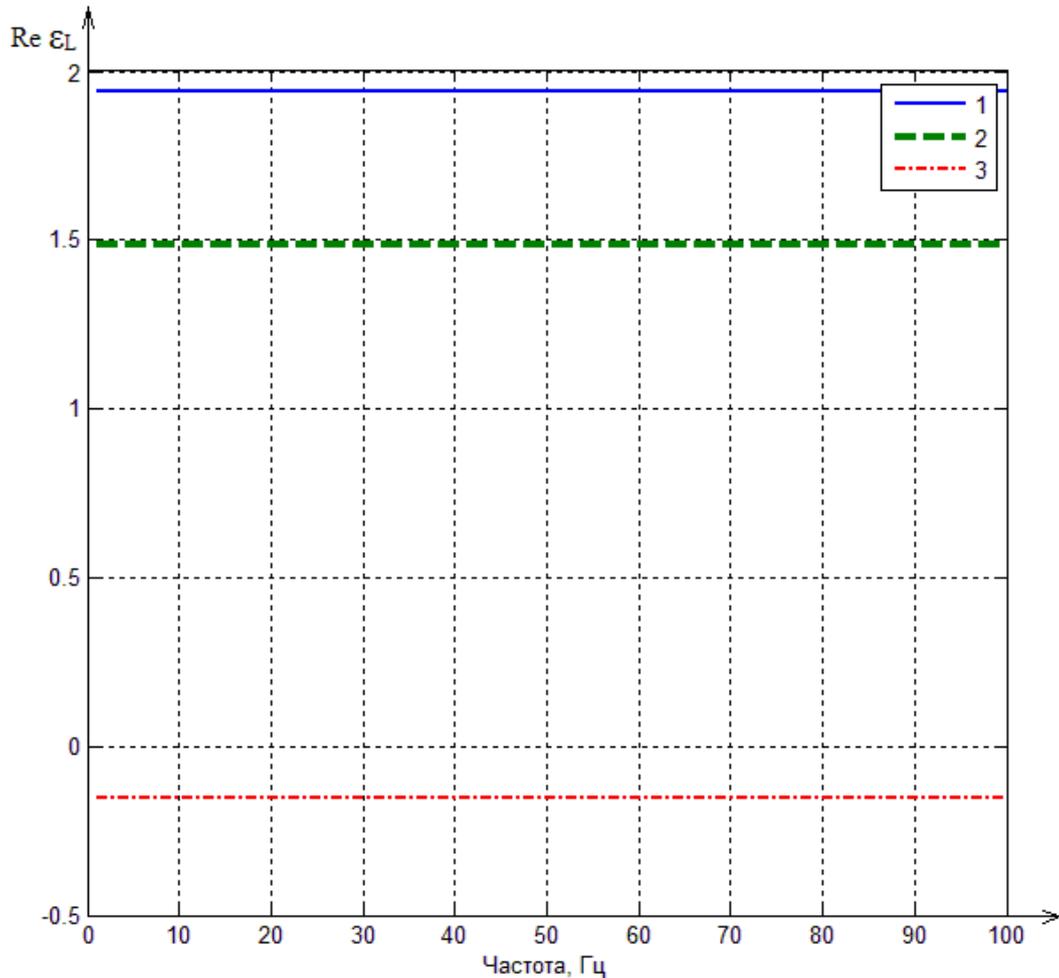


Рисунок 2. – Зависимости $Re \epsilon_L = \Psi(\sigma)$. 1 – для $\epsilon_r = 25$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 2$

На рисунке 3 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с правой поляризацией $\epsilon_R = \Psi(V_C)$.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характера сред над УВЗ.

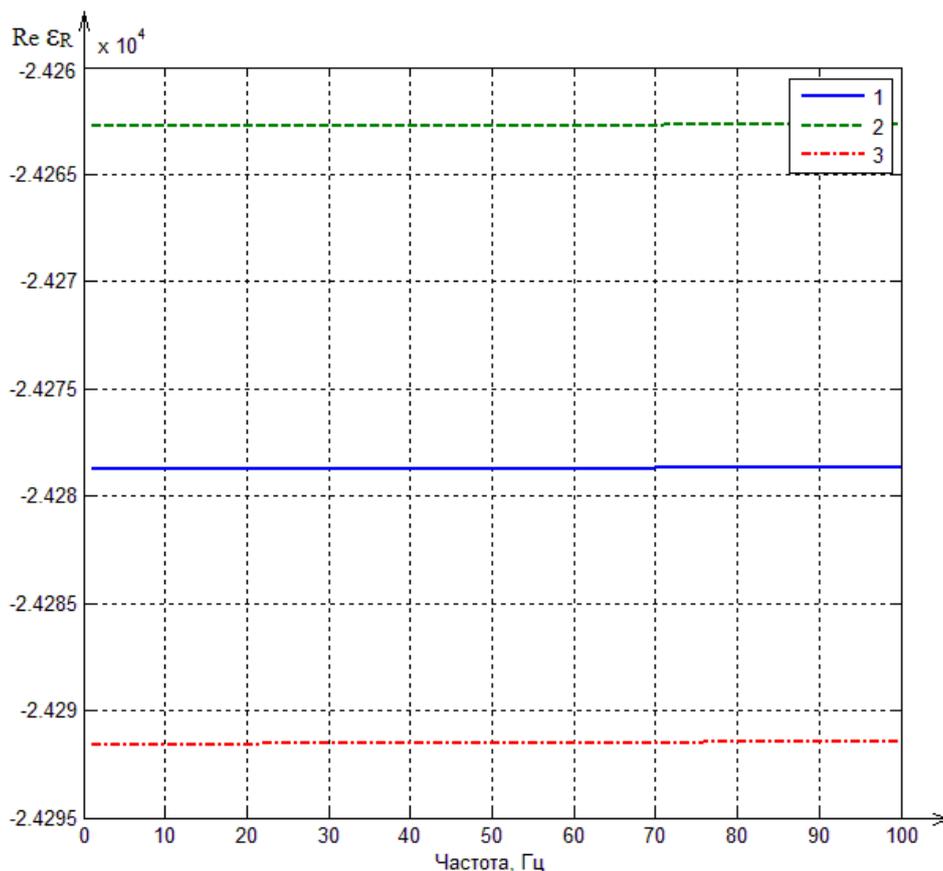


Рисунок 3. – Зависимости $Re \epsilon_R = \psi(Vc)$. 1 – для $\epsilon_r = 10$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 2$

Заключение. Проведен анализ взаимодействия анизотропного слоя над углеводородами с ЭМВ в режиме радиоимпульсных сигналов. Исследования могут быть применены для определения характеристик среды над залежью при распространении ЭМВ с правой и левой круговыми поляризациями, что повышает информативность методов оконтуривания и выделения УВЗ;

ЛИТЕРАТУРА

1. Huang R.H. Zow temperature (-80 C) termionic electron emission from alkaldes and electroles / R.H. Huang, I.Z. Dyl // Chem Phys Zett. 1990. Vol.166, № 2. P. 133 – 136.
2. Levashov, S.P. Electric-resonance sounding method and its application for, geological-geophysical and engineering-geological investigations / S. P. Levashov // 66nd EAGE Conference and Technical Exhibition, Paris, France, 7–10 June 2003 (CD-ROM Abstracts volume).
3. Adamovskiy, E., Yanushkevich, V. Simulation of electromagnetic waves interaction with hydrocarbon deposits / E. Adamovskiy, V. Yanushkevich // 8 Junior researchers conference European and national dimension in research. In 3 Parts. – Part 3. TECHNOLOGY. – PSU, Novopolotsk, 2016. – V. 179 – 183.
4. Янушкевич, В.Ф., Кременя, К.И., Иванов, М. М., Молодечкина, Т. В. Взаимодействие электромагнитных волн с анизотропными средами над углеводородными залежами в режиме радиоимпульсных сигналов. / В.Ф. Янушкевич, К.И. Кременя, М.М. Иванов, Т.В. Молодечкина. Вестник ПГУ. Серия С. Фундаментальные науки. Физика. Новополоцк, 2016. – №12. – С. 45 – 50.
5. Иванова, К.И., Янушкевич, В.Ф. Способ геоэлектроразведки углеводородной залежи с использованием радиоимпульсных сигналов. Фундаментальные и прикладные исследования в современном мире / К.И. Иванова, В.Ф. Янушкевич // Материалы XV Международной НТК. 4 октября 2016г.– Том 1. – Санкт – Петербург. – С.107 – 111.
6. Копейкин, В.В. Обратная задача георадиолокации: [Электронный документ]. (<http://www.georadar.ru/articles/article4.php>). Проверено 15-01-2013.
7. Янушкевич, В.Ф. Электромагнитные методы поиска и идентификации углеводородных залежей / В.Ф.Янушкевич. – Новополоцк, ПГУ, 2017. – 232с.

УДК 621.371.39

ВЛИЯНИЕ ЧАСТОТЫ РАДИОИМПУЛЬСНОГО СИГНАЛА, ДИЭЛЕКТРИЧЕСКОЙ ПРОНИЦАЕМОСТИ СРЕДЫ НА КОМПОНЕНТЫ ТЕНЗОРА ДИЭЛЕКТРИЧЕСКОЙ ПРОНИЦАЕМОСТИ

А. П. КОЧАНОВ

(Представлено: канд. техн. наук В.Ф. ЯНУШКЕВИЧ)

Используется вертикальная поляризация электромагнитных волн. Рассмотрен широкий диапазон используемых частот. Исследовано влияние диэлектрической проницаемости, частоты на распространение радиоимпульсных сигналов. Даны рекомендации по использованию оптимальных характеристик зондирующих сигналов. Результаты исследований могут быть использованы в поисковой геофизике.

Ключевые слова: электромагнитная волна, анизотропная среда, углеводородные залежи.

Введение. Для выделения материальных объектов на фоне окружающей среды, используются отражательные характеристики, которые служат инструментом для оптимизации электрических параметров зондирующего сигнала [1–3]. Исследуемую залежь углеводородов можно представить в виде анизотропной неоднородности на трассе связи [4–7].

В общем случае пространственная ориентация внешней нормали к границе раздела сред и волнового вектора \vec{k} является произвольной (рисунок 1) и процесс взаимодействия ЭМВ с локальным включением на трассе РРВ можно представить в виде режима наклонного падения плоской волны с вертикальной поляризацией на безграничную поверхность (в приближении больших характерных размеров неоднородности по сравнению с длиной волны зондирующего сигнала) [8–10].

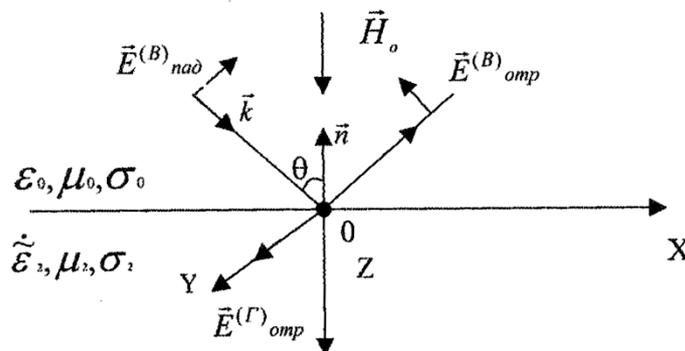


Рисунок 1. – Геометрия задачи для ЭМВ с вертикальной поляризацией поля

Объекты и методы исследования. На рисунке 2 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с правой поляризацией $\epsilon_R = \Psi(n)$.

На частоте $f = 100$ Гц величина $\text{Re } \epsilon_R$ примерно составляет 2860 единиц для диэлектрической проницаемости $\epsilon_f = 5$, для $\epsilon_f = 20$ она составляет 2870 единиц, а для $\epsilon_f = 10$ составляет 2880 единиц.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характера сред над УВЗ.

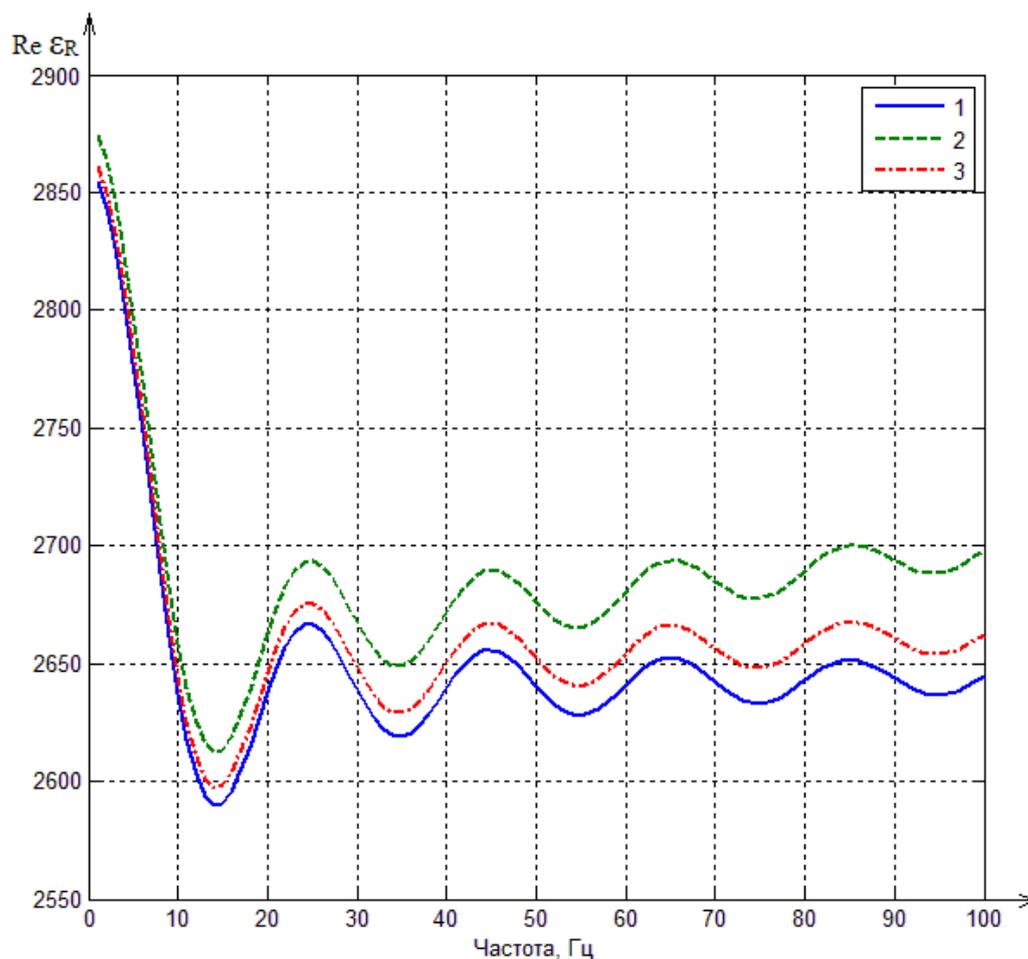


Рисунок 2. – Зависимости $Re \epsilon_R = \psi(n)$. 1 – для $\epsilon_r = 5$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 10$

На рисунке 3 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с левой поляризацией $\epsilon_L = \psi(n)$. Установлено, что с ростом частоты разностная компонента увеличивается.

На частоте $f = 100$ Гц величина $Re \epsilon_R$ примерно составляет -281 единиц для диэлектрической проницаемости $\epsilon_r = 5$, для $\epsilon_r = 20$ она составляет -276 единиц, а для $\epsilon_r = 10$ составляет -262 единиц.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости.

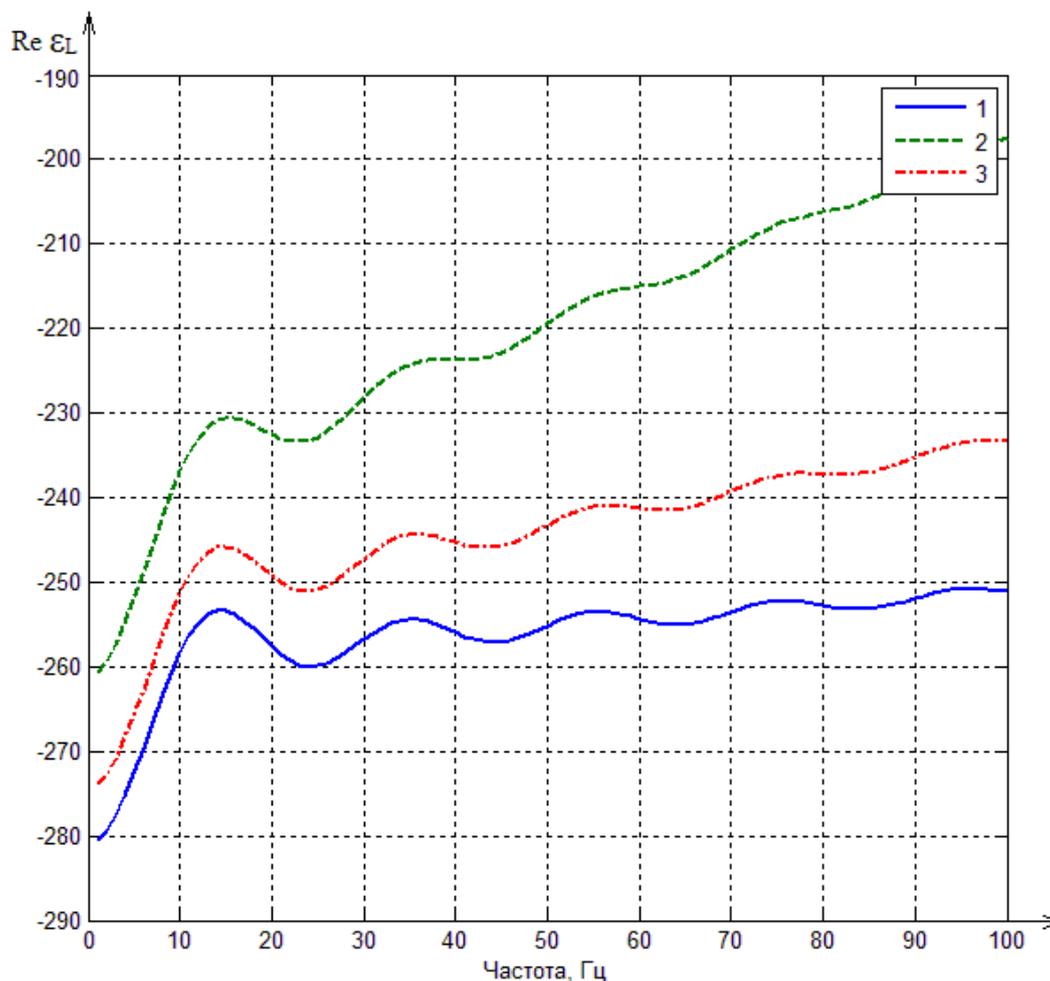


Рисунок 3. – Зависимости $Re \epsilon_L = \psi(n)$. 1 – для $\epsilon_r = 5$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 10$

На рисунке 4 представлены зависимости вещественной части диэлектрической проницаемости среды над УВЗ для электромагнитных волн с правой поляризацией $\epsilon_R = \Psi(Ni)$. Установлено, что с ростом частоты суммарная компонента увеличивается.

Как видно из приведенного графика, при изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характера сред над УВЗ.

Проведен анализ взаимодействия анизотропного слоя плазмоподобного типа с ЭМВ в режиме радиоимпульсных сигналов. Результаты исследования могут быть применены в поисковой геофизике. При этом следует отметить:

- дисперсионные характеристики в режиме радиоимпульсных сигналов для среды над залежью определены в широком диапазоне частот зондируемых сигналов, диэлектрических проницаемостей и концентраций сред;

- по сравнению с существующими аналогами данный способ обнаружения УВЗ основывается на определении дисперсионных характеристик диэлектрической проницаемости среды для правой и левой поляризаций ЭМВ в режиме радиоимпульсных сигналов, построении двухканальной измерительной системы, позволяющей повысить точность определения границ углеводородов.

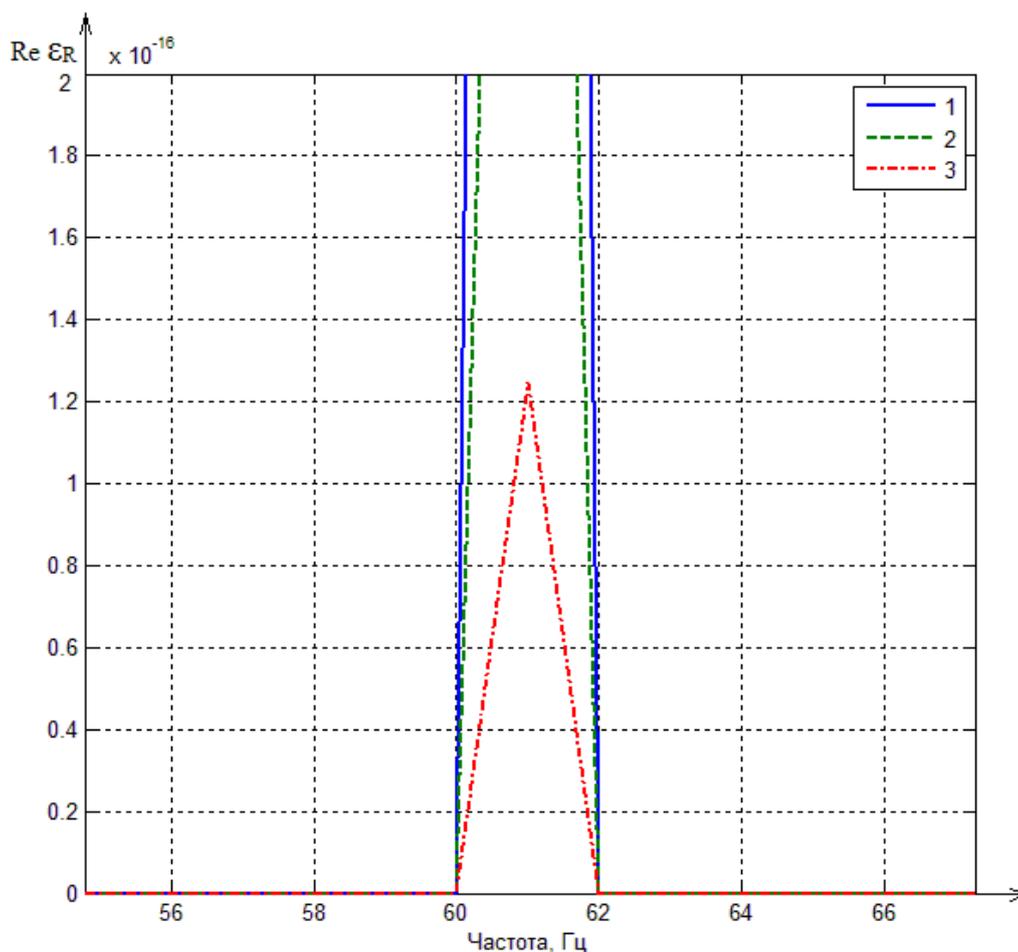


Рисунок 4. – Зависимости $Re \epsilon_R = \psi(Ni)$. 1 – для $\epsilon_r = 25$; 2 – для $\epsilon_r = 20$; 3 – для $\epsilon_r = 2$

Заключение. При изменении диэлектрической проницаемости среды над УВЗ происходит существенное изменение комбинационной компоненты тензора диэлектрической проницаемости, что может быть использовано для определения характеристик сред над УВЗ.

ЛИТЕРАТУРА

1. Гололобов, Д.В. Радиотехнические системы поиска и идентификации углеводородных залежей в режиме двухчастотного взаимодействия / Д.В.Гололобов, В.Ф.Янушкевич // Весці НАН Беларусі. Сер.фіз. тэхн. – 2002. – № 1. – С.49 – 54.
2. Moskvichew, V.N. Interaction of electromagnetic waves (EMW) with anisotropic inclusion in communication line / V.N. Moskvichew // 9-th Microw. Conf. NICON – 91, Rydzyna, May 20-22, 1991. – Vol. 1. – P. 240-244.
3. Гололобов, Д.В. Поверхностный импеданс среды над углеводородными залежами в режиме частотно-модулированных сигналов / Д.В.Гололобов, С.В.Калинцев, В.Ф.Янушкевич // Весці НАН Беларусі. Сер.фіз. тэхн. 2010. – № 4. – С.98 – 101.
4. Gaikovich K.P., Gaikovich P.K. Inverse problem of near-field scattering in multilayer media // Inverse Problems, vol.26, no. 12, pp. 125013, 2010.
5. Макаров А.М., Лунёва Л.А., Макаров К.А. Об особенностях волновых процессов в устройствах радиоэлектроники с анизотропной рабочей средой. Наука и образование: научное издание. 2016. № 11. С. 88-98.
6. Александров Ю.М., Яцышен В.В. Расчет элементов тензора комплексной диэлектрической проницаемости для анизотропных материалов. Физика волновых процессов и радиотехнические системы. 2015. Том 18. № 1. С. 23 – 27.

7. Иванова К.И., Янушкевич В.Ф. Способ геоэлектроразведки углеводородной залежи с использованием радиоимпульсных сигналов. *Фундаментальные и прикладные исследования в современном мире. Материалы XV Международной НТК. 4 октября 2016г. Том 1. Санкт – Петербург. С.107 – 111.*
8. Янушкевич, В.Ф. Экранирование радиоэлектронных устройств и систем. Новые принципы проектирования антенн: учеб. пособие для аспирантов / В.Ф. Янушкевич, М.Е. Капралов, К.И. Кременя. – Вильнюс: Циклонас, 2016. – 156 с.
9. Янушкевич, В.Ф. Электромагнитные методы поиска и идентификации углеводородных залежей / В.Ф.Янушкевич. – Новополоцк, ПГУ, 2017. – 232с.
10. Гололобов, Д.В. Электродинамические параметры подводных источников в поле поверхностной электромагнитной волны / Д.В.Гололобов, В.Н. Москвичев, Г.П.Турук, В.Ф.Янушкевич / Тез. докл. 35 Всероссийской Межвузовской НТК. –Владивосток ,1992. – Т1,ч.1. –С.59 – 62.

УДК 004.8

МАШИННОЕ ОБУЧЕНИЕ – ОСНОВА ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

В.С. ЛЕСНИЦКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)

В статье рассмотрены принципы работы машинного обучения на простых примерах. Актуальность данной темы связана с повсеместным внедрением машинного обучения в разные сферы человеческой жизни, начиная от автоматических роботов-пылесосов, до мобильных телефонов, которые сейчас в 99% случаев не обходятся без машинного обучения. Цель данной статьи объяснить человеку, не связанному с миром ИТ что такое машинное обучение и как оно работает.

Артур Сэмюэл, первопроходец в области искусственного интеллекта и компьютерных игр, ввёл термин «машинное обучение». Он определил машинное обучение как «область обучения, которая даёт компьютерам возможность учиться без явного программирования». Проще говоря, машинное обучение (МО) можно объяснить, как автоматизацию и улучшение процесса обучения компьютеров на основе их опыта без фактического программирования, то есть без какой-либо помощи человека. Процесс начинается с подачи данных хорошего качества, а затем обучения наших машин (компьютеров) путём построения моделей МО с использованием данных и различных алгоритмов. Выбор алгоритмов зависит от того, какой тип данных у нас есть и какую задачу мы пытаемся автоматизировать. [3]

Пример: обучение студентов во время экзамена. При подготовке к экзаменам студенты не заучивают, а стараются выучить его с полным пониманием. Перед экзаменом они загружают свою машину (мозг) большим количеством высококачественных данных (вопросы и ответы из разных книг, заметок учителей или онлайн-видео лекций). Фактически, они тренируют свой мозг как на входе, так и на выходе, то есть какой подход или логика у них есть для решения различных вопросов. Каждый раз, когда они решают практические тестовые задания и верность выполнения (точность/балл), сравнивая ответы, постепенно производительность продолжает расти, приобретая больше уверенности в правильности действий. Вот как на самом деле строятся модели, обучить машину с данными (как входные, так и выходные данные передаются модели), и когда придет время протестировать данные (только с входными данными) и получить оценки нашей модели, сравнив свой ответ с фактическим выходом, который не был загружен во время обучения. Исследователи прилагают все усилия, чтобы улучшить алгоритмы и методы, чтобы эти модели работали ещё лучше. На рисунке 1 показаны отличия работы обычной программы и машинного обучения.



Рисунок 1. – Особенности работы обычной программы и машинного обучения

Далее определим основные отличия МО от традиционного программирования. Традиционное программирование: вводим ДАННЫЕ (ввод) + ПРОГРАММА (логика), запускаем его на машине и получаем вывод. Машинное обучение: вводим данные (ввод) + вывод, вводим их на машине во время обучения, и машина создает свою собственную программу (логику), которую можно оценить во время тестирования.

Что именно означает обучение для компьютера? Считается, что компьютер учится на опыте в отношении некоторого класса задач, если его производительность в данной задаче улучшается с увеличением опыта, а компьютерная программа учится на опыте E в отношении некоторого класса задач T и показателя производительности P , если её производительность в задачах в T , измеренная с помощью P , улучшается с опытом E . [3]

Пример: игра в шашки.

E = опыт игры во многие шашки;

T = задача игры в шашки;

P = вероятность того, что программа выигрывает в следующей игре.

В общем, любую проблему МО можно отнести к одной из двух широких классификаций: обучение с учителем и обучение без учителя. [3]

Посмотрим, как все работает на самом деле. Говоря об онлайн-покупках, есть миллионы пользователей с неограниченным кругом интересов в отношении брендов, цветов, ценового диапазона и многого другого. При совершении покупок в Интернете покупатели обычно ищут несколько товаров. Теперь частый поиск продукта заставит покупателя веб-страницы, поисковую систему или этот интернет-магазин начать рекомендовать или показывать предложения по этому конкретному продукту. Никто не сидит там, чтобы кодировать такую задачу для каждого пользователя, вся эта задача полностью автоматическая. Здесь МО играет свою роль. Исследователи, специалисты по обработке данных, специалисты по МО создают модели на машине, используя хорошее качество и огромное количество данных, и теперь их машина автоматически работает и даже улучшается с увеличением опыта и времени. Традиционно реклама делалась только с использованием газет, журналов и радио, но теперь технологии сделали нас достаточно умными, чтобы делать таргетированную рекламу (система онлайн-рекламы), которая является гораздо более эффективным методом нацеливания на наиболее восприимчивую аудиторию.

В здравоохранении МО также отлично справляется со своей задачей. Исследователи и учёные подготовили модели для обучения машин обнаружения рака, просто глядя на изображения слайд-клеток. Для выполнения этой задачи людям потребовалось бы много времени. Но теперь, больше никаких задержек, машины предсказывают вероятность наличия или отсутствия рака с некоторой точностью, и докторам просто нужно сделать уверенный звонок, вот и все. Ответ на вопрос – как это возможно? – очень прост. Все, что требуется, это высокая вычислительная машина, большой объем данных изображения хорошего качества, модель МО с хорошими алгоритмами для достижения самых современных результатов. Врачи используют МО для диагностики пациентов по различным параметрам.

Ещё один из примеров применения МО. Использование Google Фото, где он распознает лица, Google Lens, где модель распознавания текста МО может извлекать текст из изображений, которые вы загружаете, Gmail, который категоризирует электронную почту как социальную, продвижение, обновления или форум классификация текста, которая является частью МО.

Как работает МО? Сбор прошлых данных в любой форме, подходящей для обработки. Чем выше качество данных, тем больше они подходят для моделирования. Обработка данных. Иногда собранные данные находятся в необработанном виде, и их необходимо предварительно обработать. Пример: в некоторых кортежах могут отсутствовать значения для определенных атрибутов, и в этом случае он должен быть заполнен подходящими значениями для выполнения МО или любой формы интеллектуального анализа данных. Отсутствующие значения для числовых атрибутов, таких как цена дома, могут быть заменены средним значением атрибута, тогда как отсутствующие значения для категориальных атрибутов могут быть заменены атрибутом с наивысшим режимом. Это неизменно зависит от типов используемых нами фильтров. Если данные представлены в форме текста или изображений, тогда потребуется преобразование их в числовую форму, будь то список, массив или матрица. Просто данные должны быть актуальными и последовательными. Он должен быть преобразован в формат, понятный для машины. Разделите входные данные на наборы для обучения, перекрёстной проверки и тестирования. Соотношение между соответствующими наборами должно быть 6: 2: 2. Построение моделей с подходящими алгоритмами и методами на обучающей выборке. Тестирование нашей концептуальной модели с данными, которые не были переданы в модель во время обучения, и оценка её производительности с использованием таких показателей, как оценка F1, точность и отзыв.

Таким образом, машинное обучение работает по принципу человеческого мозга, в самом начале это как новорождённый ребёнок, но чем больше мы его обучаем, тем умнее он становится. Благодаря этому мы можем создавать программы под разные сферы жизнедеятельности человека, и она будет сама обучаться, адаптироваться под конкретного пользователя. Сейчас это очень плотно используется в различных стриминговых сервисах музыки, видео и многих других, для подборки контента для конкретного пользователя.

ЛИТЕРАТУРА

1. Habr.com [Электронный ресурс]. – Режим доступа: <https://www.habr.com>. – Дата доступа: 27.09.2020.
2. Wikipedia.org [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/> – Дата доступа: 27.09.2020.
3. Рашка, Н.С. «Python и машинное обучение: машинное и глубокое обучение с использованием Python, scikit-learn и Ten». / Н.С. Рашка. – М.: Вильямс, 2019 г. – 656 с.

УДК 004.7

ТИПЫ КОМПЬЮТЕРНЫХ СЕТЕЙ – LAN, MAN И WAN

В.С. ЛЕСНИЦКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)

В данной статье рассмотрены основные виды сетей, которые используются каждый день человеком. Цель данной статьи является объяснение рядовому пользователю в чем разница между различными сетями LAN, MAN и WAN.

Сеть позволяет компьютерам подключаться и взаимодействовать с различными компьютерами через любую среду.

Локальная сеть не может охватывать города или посёлки, и для этой сети требуется городская сеть, которая может соединять город или группу городов вместе. Кроме того, для подключения страны или группы стран требуется глобальная сеть.

Наиболее распространённые типы сетей включают: локальную сеть Local Area Network (LAN), городскую вычислительную сеть (Metropolitan Area Network (MAN) и глобальную вычислительную сеть Wide Area Network (WAN). LAN, MAN и WAN – это три основных типа компьютерных сетей, предназначенных для работы в зоне покрытия. Между ними есть некоторые сходства и различия строящиеся на основе их размера, покрытия расстояния, скорости передачи данных и их охвата. Одно из основных отличий – это зона, которую они покрывают, т. е. LAN покрывает наименьшую площадь; MAN занимает площадь больше, чем LAN, а WAN – самая большая из всех.

Существуют и другие типы компьютерных сетей, например:

- Personal Area Network (PAN) – персональная сеть;
- Storage-Area Network (SAN) – сеть хранения данных;
- Enterprise Private Network (EPN) – корпоративная частная сеть;
- Virtual Private Network (VPN) – виртуальная частная сеть.

Локальная сеть (LAN). LAN или локальная сеть соединяет сетевые устройства таким образом, что персональный компьютер и рабочие станции могут совместно использовать данные, инструменты и программы. [3] Группа компьютеров и устройств соединена коммутатором или стекком коммутаторов с использованием частной схемы адресации, определенной протоколом Transmission Control Protocol/Internet Protocol (TCP/IP). Частные адреса уникальны по сравнению с другими компьютерами в локальной сети. Маршрутизаторы находятся на границе LAN, соединяя их с более крупной WAN. Данные передаются с очень высокой скоростью, так как количество подключённых компьютеров ограничено. По определению, соединения должны быть высокоскоростными и относительно недорогим оборудованием (например, концентраторами, сетевыми адаптерами и кабелями Ethernet). Локальные сети охватывают меньшую географическую территорию (размер ограничивается несколькими километрами) и находятся в частной собственности. Её можно использовать для офисного здания, дома, больницы, школы и т. д. LAN легко проектировать и поддерживать. Среда связи, используемая для LAN, состоит из кабелей витой пары и коаксиальных кабелей. Он покрывает небольшое расстояние, поэтому погрешность и шум сведены к минимуму. Ранние LAN имели скорость передачи данных от 4 до 16 Мбит/с. Сегодня скорость обычно составляет 100 или 1000 Мбит/с. Задержка распространения в локальной сети очень мала. В самой маленькой локальной сети можно использовать только два компьютера, в то время как в более крупной локальной сети могут разместиться тысячи компьютеров. Локальная сеть обычно полагается в основном на проводные соединения для повышения скорости и безопасности, но беспроводные соединения также могут быть частью локальной сети. Отказоустойчивость LAN выше, и в этой сети меньше перегрузок. Например, группа студентов играет в Counter Strike в одной комнате (без интернета).

Городская сеть (MAN). MAN или городская сеть покрывает большую площадь, чем у LAN, и меньшую по сравнению с WAN. [3] Он соединяет два или более компьютеров, которые находятся отдельно, но находятся в одном или разных городах. Он охватывает большую географическую область и может выступать в качестве поставщика услуг Интернета, т. е. Интернет-провайдера (англ. Internet Service Provider, ISP). MAN разработана для клиентов, которым нужна высокоскоростная связь. Скорости MAN варьируются в Мбит/с. Трудно спроектировать и поддерживать городскую сеть.

Отказоустойчивость MAN меньше, а также больше перегрузка в сети. Это дорого, и может принадлежать или не принадлежать одной организации. Скорость передачи данных и задержка распространения MAN умеренные. Для передачи данных через MAN используются следующие устройства: модем

и провод/кабель. Примерами MAN являются часть сети телефонной компании, которая может предоставить абоненту высокоскоростную цифровую абонентскую линию (англ. Digital Subscriber Line, DSL) или сеть кабельного телевидения в городе.

Глобальная сеть (WAN). Глобальная сеть или глобальная сеть – это компьютерная сеть, которая простирается на большую географическую область, хотя может быть ограничена пределами области или страны. WAN может представлять собой соединение LAN, соединяющееся с другими LAN через телефонные линии и радиоволны, и может быть ограничено предприятием (корпорацией или организацией) или быть доступным для общественности. Технология высокоскоростная и относительно дорогая. [3]

Существует два типа WAN: коммутируемая глобальная сеть и точка-точка. WAN сложно проектировать и поддерживать. Как и в случае с MAN, отказоустойчивость глобальной сети меньше, а в сети больше перегрузок. Среда связи, используемая для WAN – это коммутируемая телефонная сеть общего пользования (англ. Public Switched Telephone Network, PSTN) или спутниковая связь. Из-за передачи на большие расстояния в WAN больше шума и ошибок. Скорость передачи данных WAN ниже 10-й скорости LAN, так как это связано с увеличением расстояния и количества серверов и терминалов и т. д. Скорость WAN варьируется от нескольких Кбит/с до нескольких Мбит/с. Задержка распространения – одна из самых больших проблем, с которыми здесь сталкиваются. Для передачи данных через WAN используются следующие устройства: оптические провода, микроволновые печи и спутники. Примером коммутируемой глобальной сети является сеть с асинхронным режимом передачи (ATM), а двухточечная глобальная сеть – это коммутируемая линия, соединяющая домашний компьютер с Интернетом.

Таким образом, компьютерная сеть LAN имеет много преимуществ перед MAN и WAN. Согласно проведённого анализа сети LAN обеспечивают превосходную надёжность, высокую скорость передачи данных, ими легко управлять, а также они совместно используют периферийные устройства.

ЛИТЕРАТУРА

1. Habr.com [Электронный ресурс]. – Режим доступа: <https://www.habr.com>. – Дата доступа: 27.09.2020.
2. Wikipedia.org [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/> – Дата доступа: 27.09.2020.
3. В. Олифер, Н. Олифер «Компьютерные сети. Принципы, технологии, протоколы. Учебник», 2016

УДК 004.771

ПОДХОДЫ К РАЗРАБОТКЕ ИНТЕРФЕЙСА АССИСТЕНТА ДЛЯ УПРАВЛЕНИЯ УМНЫМ ДОМОМ

К.В. МАРКЕВИЧ

(Представлено: канд. техн. наук, доц. А. Ф. ОСЬКИН)

В статье представлен практический способ создания интерфейса ассистента для управления умным домом. Проведён анализ технологий, наиболее подходящих, для разработки данной системы. Задача, реализация прототипа интерфейса с использованием веб-компонентов.

Введение. Автоматизация здания – это автоматическое централизованное управление системами отопления, вентиляции и кондиционирования, освещением, контролем доступа, системами безопасности и другими взаимосвязанными системами через системы автоматизации зданий (BMS) или систему централизованного контроля и надзора технических установок в здании (BAS). Целями автоматизации здания являются повышение комфорта жильцов, эффективная работа систем здания, снижение энергопотребления, снижение эксплуатационных расходов, повышение безопасности, документация производительности, удаленный доступ / управление / эксплуатация, а также улучшение жизненного цикла оборудования и связанных с ним коммунальных услуг.

Автоматизация зданий является примером распределенной системы управления – компьютерной сети электронных устройств, предназначенных для мониторинга и управления системами в здании.

Здание, управляемое BAS, часто называют интеллектуальным зданием «умным зданием» или (если это жилой дом) «умным домом» [1].

Выбор технологий разработки интерфейса.

Выбор средств решения поставленной задачи исходил из того, каким программное и аппаратное обеспечение необходимо для разработки. Здесь выбор был сделан в пользу веб-приложения.

В качестве среды разработки для реализации проекта было выбрано Visual Studio Code.

Основные возможности и преимущества программы:

- Visual Studio Code поддерживает работу с TypeScript, JavaScript, Node.js и Mono.
- Имеются встроенные отладчик и командная строка.
- Поддержка практически всех языков программирования.
- Наличие встроенной библиотеки элементов кода.
- Автозавершение при вводе кода.
- Добавление в библиотеку собственных сниппетов.
- Подсветка синтаксиса.
- Одновременная работы с несколькими проектами.
- Поддержка многооконного и двухпанельного режимов.
- Расширение функционала с помощью плагинов.
- Интеграция с Visual Studio Team Services, GitHub и GIT.
- Наличие встроенных средств для тестирования, сборки, упаковки и развертывания приложений.
- Публикация созданных программных продуктов в Microsoft Azure (через посредство Visual Studio Team Services).
- Интегрированная система подсказок.
- Командная работа над проектами.
- Широкий набор настроек и кроссплатформенность.

В качестве языка программирования для написания приложения был выбран Python. Python, как и любой другой язык программирования, имеет свои отличительные особенности. Итак, можно выделить следующие:

Кроссплатформенность. Python – это интерпретируемый язык, его интерпретаторы существуют для многих платформ. Поэтому с запуском его на любой ОС не должно возникнуть проблем.

С Python доступно огромное количество сервисов, сред разработки, и фреймворков. Легко можно найти подходящий продукт для работы.

Возможность подключить библиотеки, написанные на C. Это позволяет повысить эффективность, улучшить быстродействие.

Наличие самых разных источников информации о Python. Не составит труда найти ответ на любой возникший вопрос, так существует много бесплатной литературы, обучающих видео-пособий, готовых исходников и шаблонов для работы в открытом доступе [2].

Проектирование интерфейса.

Интерфейс можно разделить на 4 части:

- **Bootstrap.** Это очень крошечный скрипт, который загружается на страницу в первую очередь. Он отвечает за проверку учетных данных для аутентификации и настройку соединения веб-сокета с серверной

частью. Скрипт позволяет нам начать загрузку данных, одновременно загружая остальную часть пользовательского интерфейса.

- **Оболочка приложения.** Это все, что требуется для рендеринга боковой панели и обработки маршрутизации.

- **Панели.** Каждая страница представляет собой панель. Компоненты могут регистрировать дополнительные панели, которые будут отображаться пользователю. Примерами панелей являются «состояния», «карта», «бортовой журнал» и «история».

- **Диалоги.** Определенная информация и ввод данных предоставляются пользователям в потоках. Диалоги можно запускать на любой странице. Наиболее распространенным является диалоговое окно с дополнительной информацией о сущности, которое позволяет пользователям копаться в состоянии сущности, истории и настройках.

Интерфейс использует API Websocket и Rest API для взаимодействия ассистентом.

Данные доступны в виде `has` свойства, которое передается каждому компоненту. `Has` Свойство содержит ядро состояния и имеет методы для интерфейсов вызова.

Компоненты могут подписаться на информацию, которая недоступна в основном состоянии. Подписки выполняются через API-интерфейс `websocket`, который поддерживает синхронизацию данных с серверной частью.

Используется однонаправленный поток данных. Когда вносятся изменения в бэкэнд (например, включается свет), `has` объект будет обновлен в корне приложения и станет доступным для каждого компонента, который в нем нуждается.

Поток данных. Интерфейс использует децентрализованную маршрутизацию. Каждый компонент знает о маршрутизации достаточно, чтобы знать, как обращаться с той частью, за которую он отвечает. Дальнейшая маршрутизация передается вниз по дереву компонентов.

Маршрутизация. Например, главный компонент будет смотреть на первую часть URL-адреса, чтобы решить, какая панель должна быть загружена. Каждая панель может иметь собственное сопоставление между URL-адресом и отображаемым контентом.

Заключение. Правильным подходом при разработке приложения является использование современных технологий, которые позволяют решать свои задачи. Это, в первую очередь, экономит время при разработке интерфейса, а также ресурсы на обработку данных. Это является очень актуальным, при большом количестве информации, которая подлежит постоянному изменению.

Рассмотренная технология позволяет создать адаптивный интерфейс, любого уровня сложности, при этом сохраняя его функциональные возможности и привлекательный внешний вид. По итогам данной работы, была описана возможность их применения к построению интерфейса.

ЛИТЕРАТУРА

1. Официальный сайт справочной информации. [Электронный ресурс]. Режим доступа <http://asupro.com/building/technology/structure-building-automation-system-bms.html>. Дата доступа – 20.09.2020 г.
2. Официальный сайт справочной информации. [Электронный ресурс]. Режим доступа <https://ipar.ru/poleznye-stati/4-useful/yazyk-programmirovaniya-python>. Дата доступа – 19.09.2020 г.

УДК 004.771

ПРОЕКТИРОВАНИЕ АССИСТЕНТА ДЛЯ УПРАВЛЕНИЯ УМНЫМ ДОМОМ

К.В. МАРКЕВИЧ

(Представлено канд. техн. наук, доц. А. Ф. ОСЬКИН)

В статье рассматривается проектирование системы для управления умным домом. Проведён анализ технологий, наиболее подходящих, для разработки данной системы. Проведены исследования по актуальности разработки данной системы.

Введение. Сегодня развитие автоматизированных систем является общепризнанным трендом, как в строительстве, так и мебельной промышленности. Тенденциям развития умных домов уделяется много внимания на ежегодных выставках, форумах и площадках. Данная статья направлена на описание разработанного приложения.

Основной раздел. Автоматизация окружающей среды, а точнее систем, которые окружают нас повсюду, во многом упрощает жизнь, делая ее комфортной и удобной. Система «Умный дом» — это сеть со своим интеллектуальным программным обеспечением, которое позволяет управлять инженерными системами в доме или квартире, как дистанционно, так и, находясь непосредственно на месте.

Главная характеристика системы «Умный дом» для дома это набор инструментов программы, благодаря которым осуществляется настройка и приведение в действие конкретных систем или приборов. Современные производители предлагают уже полные готовые комплекты с пакетом опций. Однако по желанию каждый может собрать себе систему из отдельных модулей, которые необходимы для оснащения конкретного дома или квартиры.

По набору опций все «умные» системы для дома делятся на несколько категорий:

- обеспечивающие безопасность в доме и на приусадебной территории — системы пожарной безопасности и видеонаблюдения, контроль газовых утечек и т. д.;
- гарантирующие экономию — тепловые, световые датчики и датчики движения, сенсоры на водные смесители;
- для поддержания комфортных условий окружающей среды — с помощью умных систем осуществляется управление шторами (открытие/закрывание), освещением, электроприборами и розетками, шлагбаумом и воротами. Поддержание комфортных климатических условий. Позволяет программировать кормление животных, полив газона и пр.;
- развлекательные — включение/выключение видео или аудиозаписей в доме или в его отдельных комнатах.

Оборудование «Умным домом» своего жилища способствует большой экономии не только времени и сил, но и бюджета семьи. Для каждой отдельной системы дома обычно требуется свой пульт или монитор, тогда как умное программное обеспечение позволяет управлять полным набором функций всего с одной сенсорной панели. Также функционал дома доступен и на планшете или смартфоне, благодаря чему контролировать «Умный дом» и настраивать его можно, находясь далеко за пределами своего жилища.

У системы есть много плюсов и минусов, с которыми каждому желающему установить «умный дом» у себя в доме следует ознакомиться заранее.

С системой «Умного дома» даже самые скрупулезные собственники могут быть уверены в безопасности своего жилища, так как у технологии широкий спектр возможностей, затрагивающих огромное количество вещей в доме. В результате установки «Умного дома» простые повседневные вещи становятся более функциональными и простыми, а на обслуживание коммуникационных систем и оборудования тратится меньше времени и сил.

Плюсы системы начинаются с обеспечения безопасности в жилище. В это понятие входит и охрана помещения, и видеонаблюдение, и противопожарный контроль. Любой оборудованный системой дом постоянно находится под видеонаблюдением со стороны своих хозяев, которые при этом могут находиться за сотни километров. Чтобы появилась возможность 24 часа получать видео в режиме онлайн, достаточно установить камеры в доме. Подключаясь к серверу, оборудованному датчиком WI-FI, можно получать изображение на свой гаджет удаленно.

Еще одна способность «Умного дома» заключается в охране собственности от несанкционированного проникновения сторонних лиц. Помимо передачи изображения с камер, система также может информировать собственников о вторжении в дом. В этот момент включается сигнализация и световой прожектор, а хозяину и службам безопасности умной системой отправляется уведомление о произошедшем. Помимо этого, интеллектуальное программное обеспечение способно имитировать в доме присутствие человека, включая свет в комнатах, воссоздавая звук работы телевизора или другой техники, а с помощью прожектора на окна могут отбрасываться силуэты людей.

Еще одна функция «умного дома» — противопожарная безопасность, которая заключается не только в определении возникновения проблемы и оповещения жильцов, но и в непосредственной борьбе с огнем. Чтобы оснастить дом такой системой, устанавливаются датчики задымления, тепла и газа. При возникновении опасности (утечке газа или задымлении), работает «Умный дом» так: прекращается подача кислорода в помещения, то есть отключается вентиляционная система, электричество и подача газа, включается сигнализация. Для безопасного передвижения по дому в этот момент остается работать лишь аварийное освещение помещений. После этого в специальные шахты вентилятором удаляется дым, а

«умный дом» отправляет сообщение в пожарную службу и владельцу о случившемся. Следующий этап — активация системы пожаротушения. Это может быть порошок или газ.

Следующая функция безопасности, которой можно оснастить дом — умный контроль протечек. Его выводят на панель управления. Если владелец решил установить такую систему, то он может выбрать

«Умный дом» с уже встроенной такой функцией, а может приобрести отдельный комплект, состоящий из датчиков (Wi-Fi или проводных), контроллера, крана перекрытия жидкости. Все места в доме, где могут возникнуть протечки (около унитаза, стиральной машинки, раковины, под кухонной раковиной и пр.), оснащаются датчиками, которые в случае фиксации протечки отправляют смс-уведомление об аварии хозяину, а также сигнал контроллеру, который перекрывает подачу воды.

Установив интеллектуальную систему дома, каждый собственник должен понимать, что окупится она в среднем через 5-8 лет, в зависимости от своей первоначальной стоимости. Однако, что сразу станет заметно, так это экономия денежных средств на платежах по оказанию коммунальных услуг в среднем на 30%. Экономия доступна, благодаря тому, что «Умный дом» контролирует расход электроэнергии, отключая все электроприборы, которые не используются домочадцами в данный момент. Помимо этого, установленные датчики движения включают свет только в тех местах, где находится человек, а затем выключает его.

Это же касается и статьи по расходам на отопление. Программное обеспечение выстраивает процесс отопления дома таким образом, что во время отсутствия жильцов на протяжении всего дня устанавливается минимальная температура (около 10 градусов по Цельсию), а за определенное время до прихода хозяев система поднимает температуру и начинает нагревать комнаты.

Комфортная температура внутри помещений, включение и выключение электроприборов и света по заданному времени, автоматическое открывание и закрывание штор, ворот или жалюзи — все это возможно, благодаря умному программному обеспечению системы.

Помимо настройки автоматической работы приборов, можно запрограммировать «Умный дом» на автоматическое кормление домашних животных. Это очень удобно в отсутствие хозяев длительное время, собачки, кошечки и даже рыбки не останутся голодными. Не смотря на многочисленные положительные аспекты использования интеллектуальной системы в доме, есть у нее и свои минусы, о которых не стоит забывать.

Минусов у системы для обеспечения комфортного проживания в доме немного, но каждый из них следует изучить прежде, чем устанавливать ее.

Первый минус — стоимость комплекта. Производители сегодня предлагают, как уже готовые системы с набором необходимого оборудования и приборов, так каждую составляющую по отдельности. В среднем цена на комплект для установки в доме небольшой квадратуры или квартире варьируется от 5 до 15 тысяч Евро. В «дешевый» набор умного комплекта войдут: датчики задымления, движения с контролем протечек, камеры и пр., которые приобретаются и устанавливаются по отдельности.

Второй недостаток заключается в человеческом факторе, а именно в мастерстве и профессионализме людей, которые будут устанавливать систему в доме. Действительно опытных мастеров, которые бы смогли собрать все компоненты в работающую сеть, особенно если речь идет о системах, где приходится совмещать разное оборудование, сегодня мало. Основной риск для хозяина жилища, устанавливающего интеллектуальное программное обеспечение, заключается в ошибках мастеров.

Еще один немаловажный отрицательный факт — чувствительность к перепадам питания в сети, от которых могут выходить из строя не только отдельные части, но и вся система в целом. Чтобы не столкнуться с подобной проблемой собственникам следует задуматься об обеспечении бесперебойного питания, для чего приобретается резервное оборудование. Также весь комплект состоит из большого количества приборов и оборудования, которое постоянно шумит и может доставлять неудобство жильцам. Из-за этого целесообразно оборудовать в доме отдельную комнату, где будет размещаться вся аппаратура.

Для большинства современных моделей систем характерна проводная связь между основными узлами, то есть устанавливать технологию придется либо на стадии черновых отделочных работ, либо полностью менять ремонт только для того, чтобы внедрить систему в дом.

Заключение. Любые технологии, гаджеты и системы входят в современную жизнь молниеносно, меняют ее к лучшему, упрощают и делают более комфортной. Однако продолжительность их жизненного цикла не велика, так как все быстро устаревает, а новые разработки занимают лидирующие позиции рынка. Эта же закономерность характерна и для систем «Умного дома».

УДК 004.05; 004.921

МЕТОДЫ ПРИВЛЕЧЕНИЯ И ПОДДЕРЖКИ ПОЛЬЗОВАТЕЛЕЙ В ИНТЕРНЕТ-СЕРВИСЕ**Н.А. МЕДВЕДЕВ***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)*

В данной статье рассматриваются основные методы по привлечению и осуществлению поддержки пользователей в интернете. Рассмотрены основные принципы и правила по регулированию процесса посещения пользователями любого сервиса.

Введение. В связи с высокой конкуренции в просторах интернета привлечение пользователей является важной задачей, а привлечение их в места для совершения купли-продажи – одна из наиболее важных, так как численность пользователей и их активность являются главнейшими показателями популярности и узнаваемости интернет-сервиса. Помимо привлечения внимания пользователя к сервису необходима и его поддержка для формирования доверия и лояльности к сервису.

Целью работы является исследование методов привлечения и поддержки пользователей в интернет-сервисе.

Вначале опишем приёмы привлечения пользователей. Для привлечения пользователей можно выделить следующие методы, которые далее рассмотрим подробнее.

1 Ценный материал. Нахождение ценного материала привлекает большой объем траффика от поискового сервера, что само собой подразумевает увеличение количества входящих внешних ссылок и социальных сигналов, что дополнительно расширяет охват сайта с помощью социальных сетей, в которых он используется совместно. Это помогает создать имидж бренда.

2 Оптимизация сервиса. Хорошая структура веб-сайта позволяет упростить задачу для понимая пользователем работу сервиса и увеличение скорости работы сервиса в целом. Можно уделить внимание использованию настроек для установления совместимости с мобильными устройствами (требуется сделать адаптивный дизайн). Дополнительно мобильный интернет-трафик генерируют в среднем от 15% всего трафика.

3 Социальные сети. Правильное использование социальной сети позволяет заметно привлечь внимание пользователей на свой сервис, а возможность иметь страницу в какой-нибудь сети повышает её популярность из-за отсутствия необходимости напрямую заходить на сервис. Это помогает отдельным пользователям, которые могут находиться только в социальной сети.

4 Халява. Людям нравится, когда им дают что-то бесплатно. И поэтому просто раздача бесплатного продукта или услуги в сервисе уже является хорошим методом продвижения сервиса ведь весьма вероятно, что такая «бесплатная раздача» не только увеличит конверсию, но и заставит пользователей рассказать своим друзьям о вашем сайте. Возможно даже, что об этом упомянет блогер или журналист.

5 Ценовая политика. Для интернет-магазина ценовая политика является наиболее важной частью хорошей политики компании. Для удачной политики рекомендуется организовать региональную политику, которая решит большинство вопросов по ценам и доступ к товарам, ведь далеко не каждый издатель захочет выставлять свой товар не в том месте. Также можно устроить бандлы (комплект, состоящий из нескольких товаров, продаваемых как единое целое) [2]. Особенно распространены в игровой индустрии.

6 Гостевое блогерство. Гостевой блог – это техника, которую можно использовать двумя различными способами. Первое, публикация своего контента в других блогах, получается ценными входящими ссылками и установка диалога с новыми пользователями. Второе, это получение хорошего контента от авторов, которые хотят опубликовать его в сервисе, что также принесёт вам дополнительный трафик, потому что автор всегда продвигает свои гостевые тексты, используя собственные каналы. Этот тип сотрудничества является наиболее естественным между сайтами, которые уже находятся в отношениях. Это означает, что лучше начинать с партнёров или, возможно, даже клиентов.

7 Маркетинг комментариев. Маркетинговые комментарии представляют комментатора как компетентного человека, представляющего серьёзный бренд, принимая участие в деловом обсуждении. Такие комментарии, которые добавляют реальную ценность дискуссии, часто заставляют читателей сделать клик по имени автора и посетить их веб-сайт. Другая сторона маркетинга комментариев – это участие в дискуссиях под вашим собственным контентом. Очень важно реагировать на все замечания и делать это как можно быстрее. Статья, в соответствии с которой пользователи задают вопросы и никогда не получают ответа, действительно может испортить весь эффект.

8 Вести себя как эксперт. Еще один способ использовать имеющиеся у вас знания – это активное присутствие на дискуссионных форумах, на сайтах типа вопросов и ответов и в тематических группах в бизнес-социальных сетях, таких как LinkedIn. Читать, какие проблемы есть у людей, и принимать участие в дискуссиях. Оказание помощи позволяет создать высокий рейтинг интернет сервиса.

9 Продвижение веб-адреса. Показывание URL-адреса где только можно даёт заметные плюсы в узнавании, что сильно повышает популярность сервиса в просторах интернета и известность у обычных людей.

10 Партнёрский маркетинг. Хорошо подготовленная партнёрская программа может стать недорогим источником достаточно высокого трафика для сервиса. Но она не всегда будет актуальным трафиком. Все зависит от того сколько партнёров будет задействовано. Имеются ли сайты с похожей тематикой, которых может быть мало. Чтобы программа приносила ожидаемую прибыль необходимо обеспечить следующее: хорошее предложение, легко продвигаемое партнёрами; привлекательные творения которые будут привлекать внимание пользователей и убеждать их кликнуть кнопку; справедливая комиссия (20-50%); чёткие правила; быстрые платежи и активный уход.

11 Публичные отношения. Отношения с СМИ (средства массовой информации) являются важным элементом продвижения своей системы. В случае отсутствия связи с известным брендом крупной корпорации можно дать им свои новости, но они должны быть действительно новаторскими, чтобы быть опубликованными. Но чтобы эти новости действительно были опубликованы где-нибудь, то необходимо строить отношения с журналистами. Свяжитесь с ними ещё до того, как у вас появится что-то важное для публикации. Предложите свою помощь в качестве эксперта, готовность быстро высказать своё мнение в случае необходимости. Если вы разговариваете с главными редакторами или редакторами разделов, предложите какой-нибудь интересный контент для их названия. Потому что, когда вам нужно сообщить рынку о чем-либо, отправьте хорошо подготовленную информацию своим контактам в прессе вместе с вопросом, есть ли шанс включить её в новости. Они предпочли бы использовать ваш текст, а не Пресс-релиз, взятый из массовой электронной почты, отправленной человеком, которого они совершенно не знают.

12 Реклама. Для многих людей это самое очевидное, что нужно сделать. Когда реклама хорошо организована, она действительно может работать. Только рекламодатели с действительно толстыми кошельками могут извлечь выгоду из классических рекламных баннеров, взимаемых за просмотр. Веб-порталы или медиа-дома готовят для них такое предложение, что кампания, по крайней мере, окупает себя. Но они также отходят от стандартной рекламы и все чаще заставляют издателей совершать нетипичные действия, такие как запуск специальных сайтов, организация конкурсов и т. д. [1].

Теперь чтобы понять как обычно ведут себя пользователи в интернет определим на какие категории их можно разделить.

Посетитель – это пользователи, которые посещают ваш сайт, но не регистрируются и не прилагают никаких усилий для предоставления учётных данных на нём. И обычно информация, которую можно найти это IP и cookies для информации вокруг него.

Зарегистрированный пользователь – посетитель, который регистрирует свою личность на вашем веб-сайте через ваш собственный процесс регистрации. Зарегистрированные пользователи не обязательно активны и на самом деле подавляющее большинство пользователей находятся далеко не на всех сайтах. Соотношение активного пользователя к зарегистрированному – это очень хороший показатель интереса пользователя к вашему продукту. Чем ближе это соотношение к 1.0, тем больше пользователей вовлечено в работу вашего приложения, даже если их доходность растягивается на месяцы.

Активный пользователь – это зарегистрированный пользователь, который использует ваше приложение по крайней мере один раз в течение 30 дней. Использование – чем чаще активный пользователь использует ваш продукт, тем более привлекательным будет ваш сайт или приложение. Деривативы (это производные финансовые инструменты) – ставка возврата является производной от вашего активного пользовательского расчёта. Если активный пользователь возвращается в среднем каждые 10 дней, то это также ваша средняя ставка возврата.

Платный пользователь – его можно заметить в социальных играх, где они тратят огромное количество денег и обычно являются около 1% всех пользователей, хотя такие компании как Zynga (американский разработчик онлайн-игр) утверждает, что на самом деле 3% являются платными и в целом даже пойдёт увеличение в 2 раза [3].

Заключение. В результате были исследованы основные виды привлечения пользователей в интернет-систему. Изучены возможность и необходимость применения для разрабатываемого интернет-сервиса. Проведён анализ иерархии пользователей для выбора подхода для реализации поддержки пользователей. Решено использовать вышеуказанные принципы для реализации проекта. Определены основные показатели удовлетворённости пользователей, удобный регламент. Продуманы основные правила.

ЛИТЕРАТУРА

1. 12 ways to attract users to your website. [Электронный ресурс] / SITEIMPULSE Blog. – Режим доступа: <https://www.siteimpulse.com/blogen/12-ways-to-attract-users-to-your-website/> – Дата доступа: 18.09.2020.
2. Бандл. [Электронный ресурс] / Wikipedia. – Режим доступа: <https://ru.wikipedia.org/wiki/Бандл>. – Дата доступа: 18.09.2020.
3. Техническая поддержка. [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Техническая_поддержка. – Дата доступа: 19.09.2020.
4. Модерация. [Электронный ресурс] / Calltouch. – Режим доступа: <https://www.calltouch.ru/glossary/moderatsiya-cto-eto-takoe-i-gde-primenyaetsya/>. – Дата доступа: 19.09.2020.
5. Internet User Defined: A User by Any Other Name... [Электронный ресурс] / Guidedlaunch. – Режим доступа: <http://guidedlaunch.com/blog/2012/internet-user-defined/> – Дата доступа: 20.09.2020.

УДК 004.05

ЗАЩИТА ИНТЕРНЕТ-СИСТЕМЫ ОТ ВЗЛОМА ПОСТОРОННИМИ ЛИЦАМИ**Н.А. МЕДВЕДЕВ***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)*

В данной статье рассматриваются методы взлома интернет-системы для получения информации и подробное описание представляющих наибольшую угрозу. Рассмотрены методы защиты интернет-сервиса от подобных несанкционированных атак.

Введение. Сегодня одной из самых серьёзных проблем в сфере информационно-вычислительных систем является защита информации в интернете. Современные люди не только разговаривают и переписываются посредством интернета, но и осуществляют различные финансовые операции, заказывают товары, услуги, пользуются кредитными карточками, проводят платежи, т. е. совершают множество действий, требующих обеспечения конфиденциальности и защиты. Попав в руки злоумышленника конфиденциальные данные способны привести к опасным последствиям. Поэтому к безопасности интернет-систем предъявляются высокие требования и рассмотрение методов по обеспечению защиты такой системы является актуальным.

Целью представленной статьи является выбор и обоснование наиболее эффективных методов защиты интернет-сервисов от несанкционированных атак.

Вначале, проанализируем основные методы взлома интернет-систем и выявим предоставляющие наибольшую угрозу.

Взлом пароля – специальная процедура методичного угадывания зашифрованного слова или фразы, которую злоумышленник пытается получить из централизованной базы данных [1]. Данные действия обычно применяются в 2 случаях, когда необходимо:

- восстановить забытый пароль;
- узнать пароль другого пользователя системы без его ведома для незаконных действий с его учетными данными.

Что касается сферы Quality assurance (QA), то процесс взлома пароля обычно используется с целью, осуществления проверки безопасности приложения, отыскав максимальное количество существующих уязвимостей в его системе [1].

В сегодняшних реалиях развития IT-сообщества, многие программисты поставили себе за цель создать особые алгоритмы, которые могли бы взламывать установленные пароли за минимальные временные промежутки. Больше половины инструментов, представленных в данном сегменте программирования, ориентируются на вход в систему на основе максимального количества допустимых словесных и буквенных комбинаций. Если перед хакером очень сложный пароль (структура которого состоит из особой комбинации цифр, букв и специальных символов), то его взлом может занять от нескольких часов до нескольких недель. Имеются и особые программы со встроенными словарями паролей, однако успешность применения подобных инструментов несколько ниже, так как с одновременным подбором комбинаций ключевые запросы сохраняются в приложении, а это занимает определенное время [1].

Существует 15 способов взлома пароля интернет-системы [2].

1. Перебор по словарю –полный перебор всех предполагаемых паролей.
2. Брутфорс –полный перебор всех символов в указанном диапазоне.
3. Фишинг – получение данных через поддельный сайт, где пользователь вводит персональные данные для входа.
4. Вирусы – использование специализированной программы по получению данных отправкой на другой компьютер.
5. Плечевой серфинг – получение данных методом просмотра другого пользователя.
6. Сканер портов – просмотр незащищённых портов и их использование для проникновения в систему жертвы.
7. Радужная таблица – метод использования для вскрытия паролей, преобразованных при помощи сложно обратимой хеш-функции, а также для атак на симметричные шифры на основе известного открытого текста.
8. Оффлайнный взлом – взлом путём восстановления пароля из кэша браузера. Требуется физическое использование системой жертвы.
9. Социальная инженерия – получение данных путём убеждения выдачи самим пользователем.
10. Гибридная атака – взлом методом комбинации брутфорса и словаря.
11. Взлом секретных вопросов – взлом путём угадывания ответа по секретным вопросам.

12. Цепь Маркова – последовательность случайных событий с конечным или счётным числом исходов, где вероятность наступления каждого события зависит от состояния, достигнутого в предыдущем событии.

13. Гибридный словарь – взлом методом словаря, а затем комбинации с брутфорсом.

14. Спайдеринг – поиск всей информации жертвы с последующим использованием брутфорса.

15. Keyloggers – занесение вируса-трояна для отлова всех символов, нажатых пользователем.

Если рассмотрим подробнее следующие виды вышеприведённых атак:

Брутфорс – взлом пароля путём перебора всех возможных вариантов ключа. Особенностью данного метода является возможность применения данного способа против любого практически используемого шифра. Однако такая возможность существует лишь теоретически, порой требуя нереалистичные временные и ресурсные затраты. Если пространство решений очень большое, то такой вид атаки может не дать результатов в течение нескольких лет или даже веков [3].

Радужная таблица – специальный вариант таблиц поиска для обращения криптографических хеш-функций, использующий механизм разумного компромисса между временем поиска по таблице и занимаемой памятью. Радужные таблицы используются для вскрытия паролей, преобразованных при помощи сложнообратимой хеш-функции, а также для атак на симметричные шифры на основе известного открытого текста. Использование функции формирования ключа с применением «соли» (об этом рассмотрим далее) делает эту атаку неосуществимой [4].

Цепь Маркова – последовательность случайных событий с конечным или счётным числом исходов, где вероятность наступления каждого события зависит от состояния, достигнутого в предыдущем событии. Характеризуется тем свойством, что при фиксированном настоящем – будущее независимо от прошлого [5]. В Марковских цепочках атак хакеры собирают определённую базу паролей. Вначале они разбирают пароли на 2-3 длинных слога символов, а затем разрабатывают новый алфавит. Таким образом, метод в основном основан на сопоставлении различных наборов паролей, пока он не найдёт исходный пароль. Это очень похоже на словарную атаку, но более продвинуто.

Вышеперечисленные атаки показывают, что возникает необходимость в применении мер по защите интернет-систем, выполнив определённую последовательность действий. В первую очередь, необходимо произвести в базе данных полное разделение пользователей на группы с разграничением уровня доступа. Затем использовать для пользователей следующие методы для осложнения взлома системы и пароля:

Хеширование – функция, осуществляющая преобразование массива входных данных произвольной длины в (выходную) битовую строку установленной длины, выполняемое определённым алгоритмом. Преобразование, производимое хеш-функцией, называется хешированием. Исходные данные называются входным массивом, «ключом» или «сообщением». Результат преобразования (выходные данные) называется «хешем», «хеш-кодом», «хеш-суммой», «сводкой сообщения» [6]. Этот способ позволяет в случае взлома не раскрыть базу паролей и значительно повышает надёжность системы. Однако в связи с использованием метода радужной таблицы рекомендуется использовать хеширование с «солью».

«Соль» – это случайная составляющая, добавляемая в конец или в начало пароля перед операцией хеширования [7]. При этом, как показано в примере выше, всякий раз получаются совершенно разные строки из одного и того же хеш-кода пароля. Соль не обязательно держать в секрете. Просто при использовании случайной величины для построения хеш-кода таблицы поиска, обратные таблицы поиска и радужные таблицы становятся неэффективными. Злоумышленник не узнает заранее, какая будет «соль», поэтому он не может предварительно вычислить таблицу поиска или радужную таблицу. Если пароль каждого пользователя хешируется с помощью разной «соли», атака с использованием обратной таблицы поиска также не будет работать [7].

Многофакторная аутентификация (МФА, англ. multi-factor authentication, MFA) – расширенная аутентификация, метод контроля доступа к компьютеру, в котором пользователю для получения доступа к информации необходимо предъявить более одного «доказательства механизма аутентификации». К категориям таких доказательств относят:

- знание – информация, которую знает субъект, например, пароль, ПИН-код.
- владение – вещь, которой обладает субъект.
- свойство, которым обладает субъект, например, биометрия, природные уникальные отличия: лицо, отпечатки пальцев, радужная оболочка глаз, капиллярные узоры, последовательность ДНК [8].

Среди видов аутентификации отдельно рассмотрим двухфакторную аутентификацию.

Двухфакторная аутентификация (ДФА) – тип многофакторной аутентификации [9]. ДФА представляет собой технологию, обеспечивающую идентификацию пользователей с помощью комбинации двух различных компонентов. Примеры ДФА: авторизация, используемая крупнейшей в мире поисковой системой интернета Google и авторизация – на уровне приложений Microsoft. Когда пользователь осуществляет вход с нового устройства, помимо аутентификации по имени-паролу его просят ввести шестизначный (Google) или восьмизначный (Microsoft) код подтверждения. Абонент может получить его по SMS оповещению, с помощью голосового звонка на его телефон, причем, код подтверждения может быть взят из

заранее составленного реестра разовых кодов или новый одноразовый пароль может быть сгенерирован приложением-аутентификатором за короткий промежуток времени. Метод выбирается в настройках аккаунта Google или Microsoft соответственно [9].

ReCAPTCHA – система, разработанная в университете Карнеги – Меллона для защиты веб-сайтов от интернет-ботов и одновременной помощи в оцифровке текстов книг. Принцип работы ReCAPTCHA кроется в недопущении податься обману программой распознавания текста. Второе слово берется из источника, требующего распознавания, например, книги. Проверка и прохождение «капчи» осуществляется по тому слову, которое известно системе. Неизвестное второе слово вводить не обязательно. Второе слово, введенное пользователем, сохраняется в системе и используется в качестве возможного варианта распознавания. Окончательное распознавание слова производится путём выбора слова, наиболее часто используемого для ввода. Система reCAPTCHA предоставляет пользователям изображения для распознавания и собирает результаты, после чего передаёт их организаторам оцифровки материалов [9].

Заключение. В результате проведённого исследования были проанализированы основные виды взлома интернет-системы и методы её защиты от подобных действий. На основании проведённого анализа планируется внедрение защиты для проекта, основной задачей которого является предоставление условий распространения неигрового ПО с возможностью запустить купленные приложения с помощью специального лаунчера.

ЛИТЕРАТУРА

1. 10 самых популярных программ для взлома паролей в 2019 году. [Электронный ресурс] / TestMatick. – Режим доступа: <https://testmatick.com/ru/10-samyh-populyarnyh-programm-dlya-vzloma-parolej-v-2019-godu/>. – Дата доступа: 15.09.2020.
2. Password Cracking Techniques Used By Hackers in 2020. [Электронный ресурс] / TechViral. – Режим доступа: <https://techviral.net/top-password-cracking-techniques-used-by-hackers/>. – Дата доступа: 18.09.2019.
3. Полный перебор [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Полный_перебор – Дата доступа: 16.09.2020.
4. Радужная таблица [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Радужная_таблица – Дата доступа: 16.09.2020.
5. Цепь Маркова [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Цепь_Маркова – Дата доступа: 16.09.2020.
6. Хеш-функция. [Электронный ресурс] / Wikipedia. – Режим доступа: <https://ru.wikipedia.org/wiki/Хеш-функция>: 17.09.2020.
7. «Соленое» хеширование паролей: делаем правильно [Электронный ресурс] / Интернет-технологии. – Режим доступа: <https://www.internet-technologies.ru/articles/solenoe-heshirovanie-paroley-delaem-pravilno.html> – Дата доступа: 17.09.2020.
8. Многофакторная аутентификация [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Многофакторная_аутентификация – Дата доступа: 17.09.2020.
9. ReCAPTCHA [Электронный ресурс] / Wikipedia. – Режим доступа: <https://ru.wikipedia.org/wiki/ReCAPTCHA> – Дата доступа: 17.09.2020.

УДК 004

**ОПИСАНИЕ ОБЩИХ ПРИНЦИПОВ И РЕКОМЕНДАЦИЙ ПО ЗАЩИТЕ
ОТ НЕКОТОРЫХ ВИДОВ ВИРУСНЫХ АТАК****И.В. МИСЕВИЧ***(Представлено канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

В данной статье описаны общие рекомендации для защиты от некоторых видов вирусных атак.

Целью данной статьи является проведение анализа поведения типичного вирусного ПО, поиск решений для устранения последствий с наименьшими усилиями, а также написание общих рекомендаций для быстрого восстановления работоспособности ОС Windows после вирусной атаки.

Актуальность данной темы обусловлена тем, что ОС Windows наиболее широко используется, и как следствие, чаще всего подвергается атаке. В данной статье описаны общие принципы работы и закономерности в поведении некоторых классов вредоносного ПО, а также способы устранения последствий их работы. Здесь не будут описываться алгоритмы работы реальных вирусов, но будут общие рекомендации по профилактическим и экстренным восстановительным мерам.

Задача состояла в анализе изменений реестра, которые вносились вирусным ПО. В основном изучались группы вирусов, направленных на остановку работы ОС (например, WinLock), которые делают невозможной дальнейшую работу ОС Windows, а, следовательно, лишают пользователя возможности устранить последствия в штатном режиме, находясь в среде штатно установленной ОС. Также рассмотрены некоторые профилактические меры общего характера, актуальные не только при использовании ОС Windows.

В любой версии ОС Windows (начиная с Windows 95) имеется реестр, представляющий из себя базу данных, в которой хранятся записи параметров, необходимых для работы различного ПО – как установленного, так и входящего в состав самой ОС. Для любых ОС Windows имеются типичные секции реестра, которые не изменяются от версии к версии, для обеспечения совместимости при работе ПО в различных версиях Windows. Например, существуют типичные места для автоматического старта программ.

В реестре имеется множество секций, которые могут быть – как созданными изначально (при установке Windows), так и создаваемыми другим ПО, для хранения его параметров.

В реестре Windows содержатся не только параметры старта, но и множество других параметров, необходимых для работы различного ПО. Например, сведения о лицензировании, персональных настройках и так далее.

Типичные секции реестра Windows, в том числе и те, которые отвечают за автозагрузку, как правило часто используются при работе различного вирусного ПО. Вирусы прописываются в эти секции, так как любая ОС Windows отреагирует на них одинаково без дополнительных настроек, что обеспечит наибольшую поражающую силу. Некоторые вирусы, например – Trojan WinLock, загружаются при старте любого пользователя Windows, поэтому приходится выполнять аварийное восстановление ОС, изменяя параметры её реестра, и при этом находиться за её пределами (например, при загрузке с различных дисков аварийного восстановления). На многих дисках могут содержаться антивирусные средства. Есть и такие диски, которые специально предназначены для этого (например – Kaspersky Rescue Disk). Однако, в базе сигнатур конкретного антивирусного ПО может не быть данной модификации вируса и/или могут отсутствовать драйвера сетевой карты в самой ОС, которая содержится на данном диске. Это лишает пользователя возможности подключиться к Интернету, и как следствие – не даёт обновить вирусную базу.

Несмотря на то, что коды различных вирусов постоянно совершенствуются и шифруются, поведение многих из них остаётся схожим, так как алгоритмы для реализации тех или иных команд Windows часто остаются неизменными. Зачастую, можно обойтись очисткой всех секций реестра, отвечающих за автозагрузку программ. Это даёт возможность отменить старт множества вирусов, затем штатно удалить сами файлы, содержащие их.

Ниже приведён пример простейшего пакетного файла сценария Windows, содержащего вредоносный код:

```
@echo off
reg add "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\poli-
cies\system" /v DisableTaskMgr /t reg_dword /d 1 /f>nul
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\poli-
cies\system" /v DisableTaskMgr /t reg_dword /d 1 /f>nul
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon"
/v Shell /t reg_sz /d %windir%\TestVirus.bat /f>nul
taskkill /f /im explorer.exe>nul
chcp 1251
shutdown /s /f /t 10 /c "Тестовый вирус">nul
xcopy /y TestVirus.bat %windir%\>nul
```

Данный код выполняет отключение диспетчера задач у всех пользователей. После этого прописывает себя в секцию реестра Windows, вместо стандартного проводника. Потом даёт команду на выключение компьютера с отсрочкой в 10 секунд, выводит предупреждающую надпись и копирует себя в папку Windows.

Так как в стандартном случае интерфейс Windows стартует автоматически, то прописанный в его секцию вирус будет стартовать вместо него у всех пользователей. Но даже после удаления файла с вирусом пользователь не получает интерфейс, так как вместо стандартной команды Explorer.exe в секции Shell прописан вирус, и проводник не стартует. Для возврата работоспособности ОС требуется вернуть стандартные значения в секции Shell и те, которые отвечают за диспетчер задач.

Конечно, этот вирус легко обойти. Например, зайдя в безопасном режиме с поддержкой командной строки. Плюс ко всему, код из BAT-скрипта легко читаем. Однако, это - не пример реального блокирующего вируса, а тестовый код, который показывает приблизительную логику работы такого класса вирусов.

Многие вирусы оставляют после себя неверные значения реестра, а некоторые из них и вовсе удаляются сами, после изменения вышеупомянутых. Как уже было сказано выше, полная очистка секций автозагрузки в большинстве случаев даёт положительный эффект. Очевидным преимуществом такого подхода являются простота и быстрое достижение результата.

Конечно, данный подход имеет множество недостатков. Например, все программы, которые загружались автоматически (за исключением тех, ссылки на которые помещены в папку автозагрузки), не будут загружаться сами, и пользователю нужно будет заново войти в эти программы и выставить соответствующие опции (например, в Skype и других программах). Но, чаще всего это не является проблемой. В большинстве случаев ОС не подвергается негативным изменениям, а блокировка снимается.

Также учтены и некоторые другие изменения, которые часто производятся вирусами, но которые можно вернуть к стандартным значениям, не навредив пользователю. Например, можно удалить файл HOSTS, и скачать новый – со стандартными значениями. Это нужно сделать для отмены работы вирусов, которые “блокируют” попадание на конкретный сайт, перенаправляя пользователя на другой. Так как файл hosts является локальной таблицей соответствия между доменным именем сайта и его IP-адресом (выполняет функцию базы DNS-сервера на конкретном компьютере), то при нахождении IP-адреса к нужному доменному имени Windows попадает именно по адресу, который написан в этом файле, а не отправляет запрос по цепочке DNS-серверов (для снижения нагрузки на них). Зачастую, вирусы типа “блокираторов соц. сетей” действуют именно так, меняя адрес конкретного сайта на подставной или локальный (127.0.0.1). Аналогично действуют и некоторые мошеннические программы: подменяют адрес сайта (например, банковского) на адрес сайта-имитатора, который полностью повторяет внешний вид оригинала, но на котором встроены скрипты, ворующий персональные данные. Такой ложный сайт может также не только отправить введённые данные в свои базы, но и передать их оригинальному сайту, перенаправив пользователя на него. Ничего не подозревающий пользователь набирает верное доменное имя, попадает на идентичный по внешнему виду сайт, вводит свои данные и авторизуется уже на оригинальном сайте, при этом не догадываясь, что его данные продублированы злоумышленникам и переданы третьим лицам.

Для исключения таких проблем можно удалить файл hosts, а также не лишним будет очистка кэшей браузеров и удаление сомнительных расширений (дополнений). Также следует отключить разрешение на автоматическую установку и обновление расширений в настройках браузера для всех сайтов. Аналогично стоит поступить и с авто перенаправлениями на другие страницы.

Также, в качестве меры предосторожности можно создать несколько закладок с часто посещаемыми сайтами, в которых прописать их прямой IP-адрес, а не доменное имя. В этом случае соответствие не потребуются, и запрос пользователя будет попадать прямо на прописанный конкретный адрес. Есть множество сервисов, на которых можно узнать IP-адрес конкретного сайта. Например, на сайте 2ip.ru имеется сервис для определения своего IP, IP других сайтов, измерения скорости интернет соединения и так далее.

Также следует отключить авто-сохранение паролей, и вообще авторизоваться на сайтах в «приватном» режиме, когда браузер не запоминает историю посещения и не сохраняет файлы cookie, так как эта информация может быть украдена вирусом.

Также следует создать отдельного пользователя с ограниченными правами, а лучше – виртуальную машину с отдельной гостевой ОС. Это нужно в случаях, когда есть необходимость переходить по сомнительным ссылкам. В гостевой ОС не следует оставлять своих данных, иначе она станет бесполезна для вышеприведённой ситуации. На виртуальной машине, которая создана для этих целей, не следует авторизоваться на проверенных сайтах, оставлять какие-либо конфиденциальные данные, пароли и так далее. Существует множество программ для работы с виртуальными машинами. Например, Oracle Virtual Box (бесплатная), VMWare (платная) и так далее. Общий принцип создания и настройки виртуальной машины достаточно прост. Все программы подобного рода эмулируют работу различных основных устройств (видеокарты, звуковой платы и т.д.). Под эти виртуальные устройства имеются пакеты драйверов, которые идут в комплекте с самой программой для работы с ВМ. Гостевая ОС работает не с реальными устройствами, а с эмулированными, а сама программа для работы с ВМ преобразует команды реальных устройств в команды для работы с виртуальными. Также эмулируется работа отдельного ПЗУ, который является файлом (или набором файлов), реальная разметка накопителей и файловые системы разделов – не затрагиваются. Такой подход делает возможной изоляцию гостевой ОС от основной, а также делает гостевую ОС программно-независимой от реального оборудования. Однако, стоит помнить, что ресурсы для работы виртуальной машины даёт основная ОС, а сама она берёт их от реального оборудования, распределяя ресурсы между приложениями.

Если работает виртуальная машина, то вирусы, проникшие на неё, не поразят основную ОС, так как гостевая ОС работает на более высоком уровне абстракции, чем основная, и программа для запуска виртуальных машин работает как обычное приложение в основной ОС.

В заключение хотелось бы отметить, что даже соблюдение всех вышеупомянутых мер не является панацеей, однако существенно снижает вероятность атаки на пользователя. Ещё раз подведем краткий список рекомендаций:

- При блокировке Windows вирусным ПО, очищать все места автозагрузки;
- Запретить изменения файла hosts (все изменения вносить только вручную, при необходимости);
- Запретить автоматическую установку расширений (дополнений) браузеров;
- Запретить автоматические перенаправления на другие страницы;
- Не использовать авто-сохранение паролей;
- Периодически очищать кэши браузеров;
- Авторизоваться на сайтах в “приватном” режиме;
- Не сохранять/удалять файлы cookie, для сайтов, требующих авторизации;
- Использовать виртуальную машину с гостевой ОС, для посещения сомнительных ресурсов в Интернете, а также для работы с неизвестным ПО.

ЛИТЕРАТУРА

1. Официальный сайт компании Microsoft [Электронный ресурс] URL: <https://www.microsoft.com> дата доступа: 26.09.2020
2. Официальный сайт лаборатории Касперского [Электронный ресурс] URL: <http://www.kaspersky.ru> дата доступа: 26.09.2020
3. Официальный сайт компании Dr.Web [Электронный ресурс] URL: <http://www.drweb.ru> дата доступа: 26.09.2020
4. Официальный сайт компании AvastSoftware [Электронный ресурс] URL: <https://www.avast.ru/> дата доступа: 26.09.2020
5. Официальный сайт компании VBA [Электронный ресурс] URL: <http://www.anti-virus.by> дата доступа: 26.09.2020
6. Официальный сайт компании Symantec [Электронный ресурс] URL: <https://www.symantec.com>. Дата доступа: 26.09.2020.

УДК 004.04

БОЛЬШИЕ ДАННЫЕ И МАШИННОЕ ОБУЧЕНИЕ

И.В. МИСЕВИЧ

(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)

Статья посвящена изучению Больших данных и их особенностей, актуальности применения и значение для нашей повседневной жизни. Особое внимание уделяется способам обработки такого вида данных, в частности, рассматривается машинное обучение как наиболее эффективный. На конкретных примерах показано то, как работают методы машинного обучения.

Актуальность данной работы обусловлена появлением нового типа данных – Больших данных, которые открывают новые возможности для практически каждой сферы общественной жизни. Проблема заключается в сложности обработки таких объёмов данных, для чего и было создано, большое количество устройств и программ, в частности, машинное обучение.

Цель – рассмотреть, что такое большие данные, машинное обучение и выяснить принцип работы машинного обучения на конкретном примере.

Задачи:

1. Поиск актуальной информации на заданную тему в различных письменных источниках и ресурсах Интернета;
2. Анализ найденной информации, её обобщение и систематизация, добавление собственных наблюдений;
3. Структурирование и изложение полученных данных;
4. Формулировка соответствующего вывода, ответ на поставленный в цели вопрос.

Методологическая основа данной работы включает в себя: анализ и обобщение специальной литературы, публикаций в периодических изданиях, наблюдение, описание, иллюстрирование примерами, формулировка собственного мнения и вывода.

У понятия «большие данные» нет однозначного определения, можно встретить множество трактовок и версий. Их объединяет одно — под большими данными подразумевается совокупность специальных технологий. Их используют для обработки значительно большего объёма данных (от петабайта: 1015 байт), чем это было до появления «больших данных». Данные определяют как множество объектов и множество соответствующих им ответов (откликов). Кроме того, большие данные должны работать с этими поступающими в большом количестве данными быстро, а также обрабатывать как структурированные, так и плохо структурированные данные.

Большие данные непрерывно накапливаются практически в любой сфере человеческой жизни. Это и социальные медиа, и медицина, и банковская сфера, и реклама, а также системы устройств, получающие многочисленные результаты ежедневных вычислений. Например, астрономические наблюдения и метеорологические сведения.

Информация с разнообразных систем слежения в режиме реального времени поступает на сервера компаний, использующих большие данные.

Технологии больших данных неотрывны от научно-исследовательской деятельности и коммерции. Более того, они начинают захватывать и сферу государственного управления – везде требуется внедрение все более эффективных систем хранения и детерминирование информации.

Существует большое количество техник и методов для анализа и обработки такой информации. Среди основных можно выделить следующие:

- **Методы класса или глубинный анализ (Data Mining).** Данные методы основаны на использовании особого математического инструментария в совокупности с достижениями из сферы информационных технологий.
- **Краудсорсинг.** Данная методика позволяет получать данные одновременно из практически неограниченного числа источников.
- **А/В-тестирование.** Из всего объема данных выбирается итоговая совокупность элементов, которую поочередно сопоставляют с другими похожими совокупностями, где был изменён один из элементов, что помогает определить, изменения какого из параметров оказывают наибольшее влияние на совокупность.
- **Прогнозная аналитика.** Данный метод направлен на предугадывание и планирование того, как будет вести себя подконтрольный объект, чтобы принять наиболее выгодное в этой ситуации решение.
- **Машинное обучение (искусственный интеллект).** Метод основан на эмпирическом анализе информации и последующем построении алгоритмов самообучения систем.

• **Сетевой анализ.** После получения статистических данных анализируются созданные в сетке узлы, то есть взаимодействия между отдельными пользователями и их сообществами.

Как уже было отмечено выше, машинное обучение является одним из методов обработки больших данных. Рассмотрим его подробнее.

Машинное обучение – это математическая дисциплина, в рамках которой решается задача поиска закономерностей в эмпирических данных; на их основе алгоритм может давать определенные прогнозы. Машинное обучение можно отнести к методам искусственного интеллекта, так как оно не решает задачу напрямую, а обучается применять решение для множества похожих задач.

Машинное обучение находится между математической статистикой, методов оптимизации и классических математических дисциплин, но имеет также и отличительную особенность, связанную с проблемами эффективности вычислений и переобучения. Многие методы также тесно связаны с извлечением информации и интеллектуальным анализом данных (Data Mining).

Машинное обучение избавляет программиста от необходимости подробно объяснять компьютеру, как именно решать проблему. Вместо этого компьютер обучают самостоятельно находить решение.

Алгоритм получает набор необходимых данных, а затем использует их для обработки запросов. К примеру, вы можете загрузить в машину код нескольких фотографий со следующим описанием: «на этом фото изображена собака» и «на этом фото нет собаки». Если после этого загрузить в компьютер большое число новых изображений, он начнёт самостоятельно сортировать снимки.

Машины учатся видеть изображения и классифицировать их, как в примере с фото. Они могут распознавать текст, числа, людей и местность на этих изображениях. Компьютеры не просто выявляют отличительные особенности для сортировки, но и учитывают контекст их употребления.

Конечно, не стоит ожидать 100% правильного результата, случаются и ошибки. Верные и ошибочные результаты распознавания попадают в базу данных, тем самым давая возможность программе учиться на своих ошибках и лучше справляться с поставленной задачей. Теоретически, процесс совершенствования может развиваться бесконечно долго. В этом и состоит суть процесса обучения.

Существует несколько общих методов машинного обучения: обучение с учителем (самый распространённый и рабочий), обучение без учителя и обучение с подкреплением.

Концепция первого метода состоит в том, что в систему загружается тренировочный набор данных – «обучающая выборка», в которой информация разбита на пары: данные ввода и данные вывода. Задача компьютера – понять логику, связывающую пары, создать алгоритм и с его помощью объединять в пары новые данные. Система постоянно совершенствуется, проанализированные данные становятся ее «опытом», она учитывает ошибки и стремится минимизировать их допущение в будущем. Так происходит обучение.

Пример: имеются данные о 1 000 000 квартир в Москве; о каждой известна площадь, количество комнат, этаж, месторасположение, наличие парковки, и так далее. Кроме того, известна стоимость каждой квартиры. Задача – построение модели, которая на основе данных признаков будет определять стоимость квартиры. Это классический пример обучения с учителем, где у нас есть данные (различные параметры для каждой квартиры) и отклики (стоимость квартиры). Такая задача называется задачей регрессии.

Обучение без учителя – тренировка без подсказок. Обучающая выборка состоит только из данных ввода. Задача программы – выявить всевозможные зависимости и связи между заданными параметрами. Так как данные не парны, у системы нет «шпаргалки» с правильными ответами. Выводы о наличии связей система делает самостоятельно. Ожидаемый результат – разделение информации на кластеры или обнаружение отклонений от введённых параметров. С каждым новым процессом система будет учиться более точно классифицировать данные.

Пример: допустим нам известны данные о росте и весе некоторого числа новорождённых. Необходимо сгруппировать данные на 3 группы, чтобы для каждой выпустить детские ползунки соответствующего размера. Это задача кластеризации. Нужно отметить, что разделение на кластеры является не таким явным и часто нет «правильного» разделения.

Ещё один метод – обучение с подкреплением. Оно подразумевает взаимодействие системы со средой. Такое взаимодействие вызывает отклик, положительный или отрицательный. Это помогает программе постепенно обнаружить оптимальные пути для стабилизации отклика.

После тысяч часов вычислений и операций, обучаемая любым из методов система готова к неизвестности. Ее чётко отработанные алгоритмы способны на прогнозирование, классификацию, кластеризацию принципиально новых данных. В процессе обработки модель продолжит учиться и совершенствоваться. Процесс обучения идет до тех пор, пока пополняется база.

Методы машинного обучения позволяют лучше понять клиента, облегчить поиск товаров, повысить конверсию, оценить риски, связанные с теми или иными инвестициями. Согласно опросу, проведённому MIT Review Custom и Google Cloud, 60% опрошенных, представляющих самые разнообразные компании, заявили, что машинное обучение для них является основным способом обработки массивов данных. Среди основных мотивов участники опроса называли стремление извлечь новые знания из своих данных, приобрести конкурентное преимущество, ускорить анализ информации и выпуск продукции нового поколения.

Два года назад Facebook, алгоритмы которого построены с помощью машинного обучения, открыл код используемого им программного обеспечения. В прошлом году открытой стала библиотека ПО для машинного обучения Google'a TensorFlow. Она помогает IT-специалистам понять, как работают модели машинного обучения, и встраивать их в свои продукты. Twitter, Uber.

Эти платформы значительно снизили важность понимания алгоритмов, на которых основывается машинное обучение. С помощью открытых инструментов и коммерческих облачных решений использование машинного обучения стало более доступно. Эксперты прогнозируют, что очень скоро любой обычный человек вместо нагромождений кода и алгебраических выкладок, сможет применять механизмы машинного обучения, пользуясь понятным интерфейсом.

Таким образом, большие данные становятся неотъемлемой частью нашего повседневного существования. Каждый день огромное количество информации о нас и о наших предпочтениях помогает создавать сложные концепции умных домов, умных городов, интернета вещей и так далее. Тем самым, анализ больших данных помогает улучшить и преобразовать почти все стороны нашей жизни.

Big Data открывает перед нами новые горизонты в планировании производства, образовании, здравоохранении и других отраслях. Если и дальше будет продолжаться их развитие, то технологии больших данных смогут поднять информацию, как фактор производства, на совершенно новый качественный уровень.

ЛИТЕРАТУРА

1. Воронцов, К. В. Лекции по машинному обучению. www.MachineLearning.ru. 2004-2016.
2. Панышин И. Машинное обучение [Электронный ресурс] – Режим доступа: <https://newtonew.com/tech/machine-learning-novice>. Дата доступа: 25.09.2020.
3. Соколов Е. Введение в машинное обучение и анализ данных [Электронный ресурс] – Режим доступа: <http://docplayer.ru/73701440-Vvedenie-v-mashinnoe-obuchenie-i-analiz-dannyh.html>. Дата доступа: 25.09.2020.
4. Машинное обучение: искусственный интеллект помогает упорядочить хаос больших данных [Электронный ресурс] – Режим доступа: <http://sap-technology.rbc.ru/mashinnoe-obuchenie.html>. Дата доступа: 25.09.2020.
5. Технологии Big Data: как использовать большие данные в маркетинге [Электронный ресурс] – Режим доступа: <https://www.uplab.ru/blog/big-data-technologies/>. Дата доступа: 25.09.2020.
6. Большие данные: новая теория и практика [Электронный ресурс] – Режим доступа: <https://www.osp.ru/os/2011/10/13010990/>. Дата доступа: 25.09.2020.
7. Big Data от А до Я. Часть 1: Принципы работы с большими данными, парадигма MapReduce [Электронный ресурс] – Режим доступа: <https://habrahabr.ru/company/dca/blog/267361/>. Дата доступа: 25.09.2020.
8. Веретенников А. В. BigData: анализ больших данных сегодня // Молодой ученый. – 2017. – №32. – С. 9-12 [Электронный ресурс] – Режим доступа: <https://moluch.ru/archive/166/45354/>. Дата доступа: 25.09.2020.

УДК 004

**РАЗРАБОТКА ОБУЧАЮЩЕГО ПРОГРАММНОГО СРЕДСТВА
«ОСНОВЫ ПРЕДПРИНИМАТЕЛЬСТВА».****О.С. МИХНОВИЧ***(Представлено: О.В. МИХНОВИЧ)*

В статье представлен объект разработки – десктопное обучающее программное средство «Основы предпринимательства». Рассмотрена актуальность разработки данного программного средства. Проведен анализ используемых для разработки технологий.

В современном мире образовательное программное обеспечение достаточно востребовано. Информационные технологии в образовании играют всё более существенное значение. Так альтернативой учебной литературе, доске и конспектам становятся программные средства обучения, которые предоставляют большие возможности для усовершенствования учебного процесса, повышают значимость самостоятельной образовательной деятельности студентов. Преподаватель перестает быть единственным источником знания, становясь руководителем и помощником студентов в образовательном процессе, обеспечивая педагогическое руководство и контроль результатов обучения, нацеливая студентов на самостоятельный поиск знаний.

Разработанное программное средство «Основы предпринимательства» позволяет студентам получать знания в области инновационной и инвестиционной деятельности, бизнес-планирования, риск-менеджмента, регулирования предпринимательской деятельности, этики предпринимательства, а также по другим актуальным аспектам предпринимательской деятельности. Данный программный продукт помогает не только изучить материал, но предоставляет возможность выполнять практические работы, закрепив тем самым полученные знания, а также проходить тестирование с целью контроля и измерения уровня знаний по данной дисциплине. Стоит отметить, что разработанное программное средство могут использовать не только студенты учебных заведений, но и все желающие, которые хотят расширить свои знания и навыки в области предпринимательства

Для удобства восприятия информации весь теоретический материал разбит по темам. Если у студента возникнут вопросы, он в любой момент может обратиться к разделу с теорией. Выполнение заданий по практическим работам происходит поэтапно: получить доступ к следующему практическому заданию можно только в случае прохождения предыдущего. Выполненные практические работы студентов хранятся в формате JSON, который является текстовым форматом обмена данными. Проверить знания помогут тесты. Тест на знание материала студент может пройти в любое время, независимо от выполненных практических работ и прочитанных лекций, во время выполнения теста доступ к теоретическому материалу блокируется. Аналитическая правовая система, доступная непосредственно из программы позволяет студентам воспользоваться актуальными аналитическими и справочными материалами для получения дополнительных сведений по изучаемому материалу. Также в программе реализована возможность выполнения контрольной работы по завершению курса. Примечательно, что аналоги данного приложения отсутствуют, делая его уникальным и привлекательным.

При реализации программного средства были выбраны наиболее подходящие инструменты для разработки. Разработка серверной части приложения осуществлялась на языке программирования Ruby, являющийся одним из самых молодых языков программирования. Это мощный и динамический объектно-ориентированный язык с открытыми исходниками. Ruby работает на многих платформах, включая Linux и другие реализации Unix, MS-DOS, Windows 9x/2000/NT, BeOS и MacOS [1]. Поскольку разрабатываемое приложение должно выполняться на современных версиях операционной системы Microsoft Windows, то наиболее целесообразно выглядит использование Microsoft .NET в качестве базовой технологии разработки. Для реализации клиентского приложения был выбран язык разработки C# – основной язык программирования платформы .NET [4]. Это высокоуровневый объектно-ориентированный, статически строго типизированный язык программирования с С-образным синтаксисом. Язык поддерживает полиморфизм, перегрузку операторов, делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, исключения, асинхронные методы [5]. Инструмент для разработки .NET-приложений – это Visual Studio, являющаяся наиболее функционально насыщенной IDE-средой, в которой можно разрабатывать приложения на C#. Данный продукт поставляется с конструкторами графических пользовательских интерфейсов, с возможностями работы с базами данных, с поддержкой фрагментов кода, с возможностями для просмотра всего проекта в целом, с просмотром свойств объектов [2].

Для реализации задачи по разработке вышеописанного программного средства была выбрана система управления базами данных (СУБД) PostgreSQL 9.5. Это обусловлено тем, что PostgreSQL – бесплатное программное обеспечение с открытым исходным кодом. Кроме того, эта СУБД является очень мощной системой. PostgreSQL не просто реляционная, а объектно-реляционная СУБД. Это даёт ей некоторые преимущества над другими SQL базами данных с открытым исходным кодом, такими как MySQL, MariaDB и Firebird [3]. Плюсы использования СУБД PostgreSQL в данном проекте следующие: ее совместимость с технологиями языка Ruby, легковесность (при наличии всех необходимых функций), свободное распространение, а платформа Microsoft Windows на данный момент пока останется самой популярной.

Таким образом, при помощи современных инструментов для разработки клиент-серверного приложения удалось создать программное средство, которое позволяет изучать дисциплину «Основы предпринимательства» на высоком уровне. Данное программное средство уже внедрено в учреждение образования «Полоцкий экономический колледж».

ЛИТЕРАТУРА

1. Язык программирования Ruby [Электронный ресурс] /Интернет –технологии. – Режим доступа: <http://www.internet – technologies.ru/articles/yazyk-programmirovaniya-ruby.html> – Дата доступа: 10.04.2020.
2. Visual-studio [Электронный ресурс] – Режим доступа: [https:// alekseygulynin.ru /visual-studio/](https://alekseygulynin.ru/visual-studio/)– Дата доступа: 06.04.2020
3. Хабрахабр – Чем PostgreSQL лучше других SQL БД с открытым исходным кодом – Режим доступа: <https://habrahabr.ru/post/282764/>. – Дата доступа: 27.04.2020.
4. .NET Development [Electronic Resource] / Microsoft Corporation. – Mode of access: <https://msdn.microsoft.com/en-US/library/ff361664.aspx>. – Дата доступа: 14.04.20120.
5. Википедия [Электронный ресурс] Sharp. – Режим доступа: https://ru.wikipedia.org/wiki/C_Sharp. – Дата доступа: 25.03.20.

УДК 004

ВОЗМОЖНОСТИ ЯЗЫКА ПРОГРАММИРОВАНИЯ CRYSTAL

О.С. МИХНОВИЧ

(Представлено: Е.В. ДАНЧЕНКО)

В статье обсуждаются возможности языка программирования Crystal и его преимущества перед языками программирования C и Ruby на примере реализации задачи Ханойской башни.

На сегодняшний день существует множество различных языков программирования. Одни из них позволяют разработчику писать хорошо читабельный и эффективный код, другие же позволяют сосредоточить внимание на производительности будущего продукта. Языки программирования по виду типизации принято делить на две категории – статические и динамические. К первому, например, относятся: C/C++, Java, C#, а ко второму – Ruby, Python, JavaScript. Одним из популярных динамических интерпретируемых языков программирования является Ruby.

Ruby обладает независимой от операционной системы реализацией многопоточности, сильной динамической типизацией, сборщиком мусора и многими другими возможностями. По особенностям синтаксиса он близок к языкам Perl и Eiffel, по объектно-ориентированному подходу – к Smalltalk. Также некоторые черты языка взяты из Python, Lisp, Dylan и Клу. На сегодняшний день Ruby используется в связке с фреймворком Rails. Однако у Ruby есть и недостатки. Наиболее существенным недостатком Ruby является его производительность. Рассмотрим язык программирования Crystal, как более производительную альтернативу Ruby [1].

Crystal – это объектно-ориентированный язык общего назначения, спроектированный и созданный Ary Borenszweig и Juan Wajnerman. Crystal статически типизирован и имеет Ruby-подобный синтаксис. Первый официальный релиз языка произошел в июне 2014 года. Изначально компилятор языка был написан на Ruby, пока в 2013 году не был переписан на Crystal. Язык находится в активной разработке. Несмотря на схожесть синтаксиса, Crystal намного эффективнее, чем Ruby, компилируется в машинный код, используя LLVM, жертвуя при этом динамическими аспектами языка. По результатам тестов Crystal показывает схожую с языком C производительность. Язык использует Boehm garbage collector, обладает системой макросов, поддерживает дженерики, а также перегрузку методов и операторов[2].

При проектировании и разработке Crystal создатели ставили перед собой следующие цели:

1. Ruby-подобный синтаксис.
2. Статическая типизация, но без указания типа переменных или аргументов метода.
3. Возможность вызывать код на C путем написания биндингов внутри Crystal.
4. Есть оценка времени компиляции и генерации кода, дабы избежать шаблонности кода.
5. Эффективная компиляция в машинный код.

Для сравнения производительности Crystal возьмем два языка программирования: Ruby и C. Выбор пал на данные языки в связи с тем, что Crystal позиционируется как сбалансированная между данными языками технология. Для сравнения производительности реализуем решение задачи о Ханойской башне.

Ханойская башня[3] является одной из популярных головоломок XIX века. Даны три стержня, на один из которых нанизаны восемь колец, причём кольца отличаются размером и лежат меньшее на большем. Задача состоит в том, чтобы перенести пирамиду из восьми колец за наименьшее число ходов на другой стержень. За один раз разрешается переносить только одно кольцо, причём нельзя класть большее кольцо на меньшее.

Существует несколько подходов к решению (рекурсивно, «треугольное» решение, циклическое решение), все они дают идентичные результаты. Реализуем рекурсивное решение.

Реализация решения на Ruby представлена в листинге ниже.

```
require 'benchmark'
```

```
N=20
```

```
A=Array.new
```

```
B=Array.new
```

```
C=Array.new
```

```
LOG = false
```

```
def move(n, source, target, auxiliary)
```

```
  if n > 0
```

```
    move(n - 1, source, auxiliary, target)
```

```
    target.append(source.pop)
```

```
    print "*****\n#{A}\n#{B}\n#{C}\n" if LOG
```

```
    move(n - 1, auxiliary, target, source)
```

```
  end
```

```

end
def run()
  (1..N).each { |v| A << v }
  A.reverse
  move(N, A, C, B)
end
puts "\nRuby work time: #{Benchmark.measure { run() } }"

```

Реализация решения на Crystal представлена в листинге ниже.

```

require "benchmark"
N = 20
A = Array(Int32).new
B = Array(Int32).new
C = Array(Int32).new

LOG = false

def move(n, source, target, auxiliary)
  if n > 0
    move(n - 1, source, auxiliary, target)
    target << source.pop
    print "*****\n#{A}\n#{B}\n#{C}\n" if LOG
    move(n - 1, auxiliary, target, source)
  end
end

def run()
  (1..N).each { |v| A << v }
  A.reverse
  move(N, A, C, B)
end

puts "\nCrystal work time: #{Benchmark.measure { run() } }"

```

Листинги с решениями задачи «Ханойская башня» на языках Crystal, C, Ruby находятся на GitHub [4]. После запуска решений на выбранных языках получаем следующие результаты (в секундах):

Таблица 1 – Результаты тестов

Язык	Тест 1	Тест 2	Тест 3	Среднее значение
C	0.000895	0.000957	0.000955	0.000936
Crystal	0.021738	0.021611	0.013851	0.019067
Ruby	0.364303	0.359521	0.359140	0.360988

Проанализировав полученные результаты можно сделать вывод: язык программирования Crystal сочетает в себе достаточно высокую, как в C, производительность, и читабельный, и лаконичный, как в Ruby, синтаксис и может применяться для разработки программных продуктов под операционные системы семейства Unix. Из минусов данного языка можно отметить пока слабое комьюнити и малое количество библиотек.

ЛИТЕРАТУРА

1. Ruby [Электронный ресурс] – Режим доступа: <https://www.ruby-lang.org/>. – Дата доступа: 20.01.2020.
2. Crystal [Электронный ресурс] – Режим доступа: <https://crystal-lang.org/>. – Дата доступа: 20.01.2020.
3. The Tower of Hanoi [Электронный ресурс] – Режим доступа: https://en.wikipedia.org/wiki/Tower_of_Hanoi/. – Дата доступа: 20.01.2020.
4. Crystal benchmark [Электронный ресурс] – Режим доступа: <https://github.com/olegmikhnovich/cr-rb-c-benchmark/>. – Дата доступа: 20.01.2020.

УДК 004.716

VPN И ЕГО ПРИМЕНЕНИЕ

Д.Д. МОРОЗОВ

(Представлено: канд. физ.-мат. наук, доц. О.В. ГОЛУБЕВА)

В статье рассказывается, что такое виртуальная частная сеть и для чего она используется в различных сферах.

Ключевые слова: виртуальная частная сеть, VPN-сети, защита трафика.

VPN сейчас актуален, как никогда прежде. Даже в домашних роутерах стали появляться не просто VPN-серверы, а еще и с аппаратным ускорением шифрования. Что же такое VPN и для чего он вообще нужен?

Что такое VPN?

Аббревиатура VPN расшифровывается как Virtual Private Network, то есть виртуальная частная сеть. Что такое «сеть» — на примитивном уровне это объединение двух и более узлов каким-либо видом связи для того, чтобы они могли обмениваться информацией. Естественно, наиболее удобным способом и с поддержкой всех необходимых сервисов.

Что такое «частная» — не публичная, поэтому и частная. То есть такая, в которой только дозволенные узлы. Именно эта составляющая VPN и является самой главной, так как она определяет ряд требований к этой самой «частности».

Во-первых, надо как-то маркировать участников этой сети и ту информацию, которой они обмениваются, чтобы она не смешивалась с чужой. Во-вторых, определенно полезно эту информацию защитить от посторонних глаз. Ну хотя бы зашифровать, что снова накладывает следующий круг ограничений, связанных со стойкостью этого шифрования.

В-третьих, надо сохранять целостность такого способа передачи информации — не пускать посторонних в частную сеть, проверять источник передаваемых сообщений и следить за тем, чтобы информация нигде не просачивалась в «голом виде».

Понятие «виртуальная» значит, что такая сеть абстрагирована от физической составляющей — ей не важно, по каким и скольким каналам связи она проложена, так как для участников этой сети она работает прозрачно. Или же, с другой стороны, физическая сеть чаще всего просто не принадлежит пользователю виртуальной.

Например, в серьезных организациях сотрудников при подсоединении рабочего ноутбука к любым проводным или беспроводным сетям, находящимся за пределами стен этой самой организации, обязуют сразу же задействовать VPN-подключение до офисной сети. При этом не важно, через какие именно сети будет установлено это соединение, но можно не сомневаться, что пойдет оно по публичным, чужим сетям связи. Такое соединение принято называть туннелем, впоследствии этот термин нам встретится еще не раз.

Для чего нужен VPN?

Приведенный выше пример подключения удаленного пользователя к корпоративной сети — один из наиболее типичных сценариев использования VPN. Злоумышленник не сможет просто так узнать, чем конкретно занят этот пользователь, какие данные он передает и получает. Более того, в компаниях, озабоченных собственной безопасностью, на всех используемых работниками устройствах принудительно включается обязательное использование VPN-подключений где бы то ни было. Даже использование Интернета в таком случае идет сквозь корпоративную сеть и под строгим надзором службы безопасности.

Второй по распространенности вариант использования схож с первым, только подключаются к корпоративной сети не отдельные пользователи, а целые офисы или здания. Цель та же — надежно и безопасно объединить географически удаленные элементы одной организации в единую сеть.

Это могут быть как крупные представительства корпораций в разных странах или даже просто камеры, сигнализации и прочие охранные системы. При такой простоте создания VPN виртуальные частные сети могут создавать и внутри компаний для объединения и изоляции тех или иных отделов или систем.

Не менее часто организуются VPN-сети и между серверами или целыми вычислительными кластерами для поддержания их доступности и дублирования данных. Частота их использования напрямую связана с ростом популярности облачных технологий. Причем все вышеперечисленное — это не какие-то временные решения: такие подключения могут поддерживаться (и поддерживаются) годами.

У России свой путь применения VPN на практике. Когда-то крупные ISP строили свои сети на основе простых неуправляемых коммутаторов — очевидно, в целях экономии. Для разделения трафика клиентов стали использовать различные варианты VPN-подключений к серверу провайдера, через который и выдавали доступ в Интернет.

Такой метод используется до сих пор, а производители домашних роутеров для российского региона все еще вынуждены добавлять поддержку таких подключений в прошивку своих устройств. Так что, в каком-то смысле, Россия была лидером по числу одновременных VPN-подключений среди пользователей Сети.

Контрпример таких постоянных VPN-соединений — это сессионные подключения. Они нередко используются при предоставлении клиентского доступа к различным сервисам, которые, как правило, связаны с обработкой очень чувствительной информации в области финансов, здравоохранения, юриспруденции.

Впрочем, для обычного пользователя гораздо важнее другой вариант практического использования VPN. В советах по безопасности Android и iOS настоятельно рекомендуется применять защищенное VPN-соединение до надежного узла (будь то домашний роутер или специальный VPN-провайдер) при подключении к любым публичным сетям, чтобы защитить свой трафик от возможного вмешательства злоумышленников.

Наконец, последний вариант применения VPN в частном порядке — это обход разнообразных сетевых ограничений. Например, для получения доступа к ресурсам, которые заблокированы или не предоставляют свои услуги на определенной территории. Согласно отчету GlobalWebIndex, только в 2014 году для доступа к социальным сетям VPN использовали около 166 млн. человек.

ЛИТЕРАТУРА

1. A Framework for IP Based Virtual Private Networks [Электронный документ] / B. Gleeson, A. Lin, J. Heinanen. — <http://www.ietf.org/rfc/rfc2764.txt>.
2. VPN и IPSec на пальцах [Электронный документ] / Dru Lavigne. — <http://www.nestor.minsk.by/sr/2005/03/050315.html>.
3. Сергей Петренко Защищённая виртуальная частная сеть: современный взгляд на защиту конфиденциальных данных // Мир Internet. — 2001. — № 2.
4. Маркус Файльнер Виртуальные частные сети нового поколения // LAN. — 2005. — № 11.

УДК 004.423.42

САЙТ И ЕГО СОСТАВЛЯЮЩИЕ

Д.Д. МОРОЗОВ

(Представлено: канд. физ-мат. наук, доц. О.В. ГОЛУБЕВА)

В статье рассказывается, что такое сайт и его структура.

Ключевые слова: сайт, верстка, веб-сервер, контент, домен.

Сайт, или **веб-сайт** (от англ. *website*: *web* — «паутина, сеть» и *site* — «место», буквально «место, сегмент, часть в сети»), — одна или несколько логически связанных между собой веб-страниц также место расположения контента сервера. Обычно сайт в интернете представляет собой массив связанных данных, имеющий уникальный адрес и воспринимаемый пользователями как единое целое. Веб-сайты называются так, потому что доступ к ним происходит по протоколу HTTP.

Веб-сайт как система электронных файлов может принадлежать частному лицу или организации и быть доступным в компьютерной сети под общим доменным именем и IP-адресом или локально на одном компьютере. В статье журнала «Хозяйство и право» также было высказано мнение, что каждый сайт имеет своё название, которое при этом не следует путать с доменным именем. С точки зрения авторского права сайт является составным произведением, соответственно название сайта подлежит охране наряду с названиями всех прочих произведений.

Все сайты в совокупности составляют Всемирную паутину, где коммуникация (паутина) объединяет сегменты информации мирового сообщества в единое целое — базу данных и коммуникации планетарного масштаба. Для прямого доступа клиентов к сайтам на серверах был специально разработан протокол HTTP.

Рассмотрим основные технические компоненты и строение сайта.

Дизайн сайта отвечает за визуальное представление и организацию информации, способствует взаимодействию ресурса с его посетителями. Отвечающим за создание дизайна специалистом обычно является веб-дизайнер. Именно он должен спроектировать логическую структуру всех страниц сайта, разработать способы подачи материала и проработать внешнее оформление ресурса.

Для каждой страницы дизайн разрабатывается отдельно в зависимости от ее функций. Например, у сайта интернет-магазина есть страница, где представлен товар, а также страница с пользовательской корзиной. Естественно, что эти директории по логике своего оформления должны быть разными. Цель размещения корзины — способствовать быстрому и удобному совершению покупок, а страницы, представляющей товар, — ознакомление со свойствами продукта, сравнение его с другими товарами.

Верстка представляет собой процесс написания особого кода для браузеров. У такого файла расширение html, и отображается он только в браузерах. Нужно следить, чтобы сверстанный макет одинаково выглядел во всех браузерах, будь то Opera, Mozilla Firefox, Google Chrome или Internet Explorer.

Программирование составляет примерно 50-70% работы над сайтом. Программная часть ресурса — довольно обширное понятие. Программирование отвечает за то, чтобы многочисленные страницы сайта отображались с одинаковым дизайном, но разным содержимым. Также программный код ответственен за визуальные эффекты и многое другое. Благодаря ему сайт становится «живым» и динамичным.

Веб-сервер — это компьютер с установленным на нем программным обеспечением, которое предназначено для того, чтобы отвечать на запросы веб-клиента круглосуточно в режиме реального времени. Под веб-клиентом подразумевается браузер, который отправляет серверу запрос на определенную страницу, и если она доступна, то пользователь может увидеть ее. Чтобы пользователь увидел нужную ему страницу, его браузер должен получить от сервера соответствующий html-код. После чего код и все визуальные элементы распознаются и предстают перед пользователем в виде готовой понятной любому странице.

Клиентская часть представляет собой код, загружаемый вместе с кодом html. Это может быть CSS, JavaScript, ActionScript.

Контент сайта, то есть все содержимое сайта. Сюда входят тексты, картинки, видео, flash и прочие файлы. Контент часто путают с элементами дизайна, но это совершенно разные вещи. Оформление сайта отвечает на вопрос «как разместить», а контент отвечает на вопрос «что разместить». Что касается текстового наполнения, то именно оно наполняет любой веб-ресурс смыслом, благодаря которому пользователи могут найти сам сайт в поисковых системах. Интересный контент — это ключ к увеличению посещаемости, читаемости и коммерческих показателей, таких как звонки, продажи или подписки на e-mail рассылки.

CMS тоже можно отнести к одному из элементов сайта. Система управления содержимым позволяет управлять всеми элементами ресурса через административную часть. Использование CMS не обязательно,

однако оно необходимо всем, кто хочет самостоятельно контролировать и редактировать содержимое сайта. Существует целый ряд разнообразных CMS, отличающихся друг от друга удобством пользования, набором свойств и принципами работы.

Доменное имя и хостинг. Доменное имя – это уникальный адрес сайта в Сети. А хостинг – это услуга, которая предоставляется специальными компаниями, хранящими на своих серверах все необходимые для работы сайта данные. Без этих компонентов строение было бы неполным, а сам сайт - недоступен для пользователей и посетителей. Хостинг предполагает выделение свободного места на сервере для размещения и хранения какого-либо сайта. Чтобы владелец был уверен в том, что его сайт всегда виден пользователям Интернета, он обязательно должен обзавестись уникальным доменным именем, а также надежным хостингом.

ЛИТЕРАТУРА

1. Воройский, Ф. С. Информатика. Энциклопедический систематизированный словарь-справочник. — Физматлит, 2006. — С. 432. — 945 с.
2. Бобкова, О., Давыдов, С. К. вопросу о соотношении понятий «доменное имя» и «название сайта» // Хозяйство и право. — М., 2014, № 6. — С. 102—106.
3. info.cern.ch.
4. Исследование безопасности коммерческих сайтов за 1 кв 2015 года (19 мая 2015).
5. Сервис защиты сайтов SiteSecure.ru.

УДК 004.414.3

ПРОЕКТИРОВАНИЕ WEB-ПРИЛОЖЕНИЯ «FANTASY» ДЛЯ РАЗВИТИЯ КРЕАТИВНОСТИ У ДЕТЕЙ ШКОЛЬНОГО ВОЗРАСТА**М. НЕПЕСОВ***(Представлено: Т.М. ГЛУХОВА)*

В статье проанализированы требования, предъявляемые к разработке Web-приложения «Fantasy». Приложение является частью проекта Stimey (финансируется программой Горизонт 2020), стимулирующего развитие STEM-навыков у детей школьного возраста.

Задачи и цели проекта. Целью проекта Stimey является стимулирование интереса детей школьного возраста к техническим дисциплинам (развитие STEM-навыков). Перед нами была поставлена задача разработки отдельного web-модуля, который бы отвечал за развитие креативности у детей.

Как отмечают многие исследователи, адекватная самооценка учащегося, вера в свои способности, потребность в их реализации – личностно значимый мотив креативной деятельности [1].

Термин «креативность» происходит от латинского creare, что означает «порождать, создавать, творить». Под креативностью понимается относительно устойчивая характеристика личности, сущность которой составляют умственные процессы, ведущие к решениям, идеям, осмысленному созданию художественных форм, теорий или любых других уникальных и новых продуктов. Креативность связана с особенностями мышления, совокупностью способностей, позволяющих генерировать идеи, отличающиеся от общепризнанных, стереотипных, умением создавать новые. Креативность – это способность реагировать на необходимость новых подходов и новых творческих продуктов, создание которых во многом зависит от личности творца и силы его внутренней мотивации [2].

Таким образом, перед нами была поставлена задача создать инструмент, мотивирующий детей к созданию новых продуктов, генерированию идей, применению нестандартных методов. Разрабатываемый инструмент должен быть направлен не на выполнение детьми заранее подготовленных заданий, а скорее как инструмент учителя для обогащения образовательного процесса новыми методами организации самостоятельной работы учащихся.

Конечной целью образовательной программы является формирование креативной личности, способной к развитию, обогащению и реализации своего потенциала в существующих условиях [3].

Таким образом среди целей разработки web-приложения «Fantasy» можно выделить:

– с точки зрения освоения и закрепления материала преподавателя:

- наглядное представление материала;
- применение разных источников информации;
- разные способы представления информации (текст, графический материал, аудио, видео);
- оперативный контроль знаний;
- мотивирование путем назначения баллов;

– с точки зрения подготовки материала для своих сверстников:

- научить ребенка выбирать источники информации;
- научить структурировать информацию и преподнести ее доступно для своих сверстников, используя выбранные источники информации;
- представлять результаты работы в нестандартной форме;
- уметь формулировать проблемную ситуацию;
- уметь подготовить тестовые задания для контроля освоения материала.

Проектирование программного обеспечения. Для достижения подобных целей в программном обеспечении должны быть представлены 2 роли пользователей: Ученик и Учитель.

Ученик имеет ряд прав в приложении. Для прохождения фантазий, которые были ранее созданы учителем или другими учениками, ему достаточно выбрать интересующую его фантазию по любым критериям: название, тематика, уровень сложности (прецедент Поиск фантазии). После выбора фантазии ученику открывается рабочее поле с активной точкой (прецедент Пройти фантазию). Каждая активная точка – это набор информации на определенную тему. Информация представляется в разных формах: текст, иллюстрации, медиа-информация (видео). После ознакомления с информацией ученик может пройти тест. После успешного прохождения теста на рабочем поле будет открыта следующая активная точка (прецедент Пройти активную точку). Проходя все активные точки, ученик зарабатывает баллы. В описании фантазии будет опубликован прогресс прохождения учеником этой фантазии. Дополнительно ученик может

просмотреть статистику по пройденным им фантазиям, чтобы узнать полученные баллы в активных точках (прецедент Просмотреть статистику).

Диаграмма прецедентов для роли Ученик (перечень функций, доступный Ученику) приведена на рисунке 1.

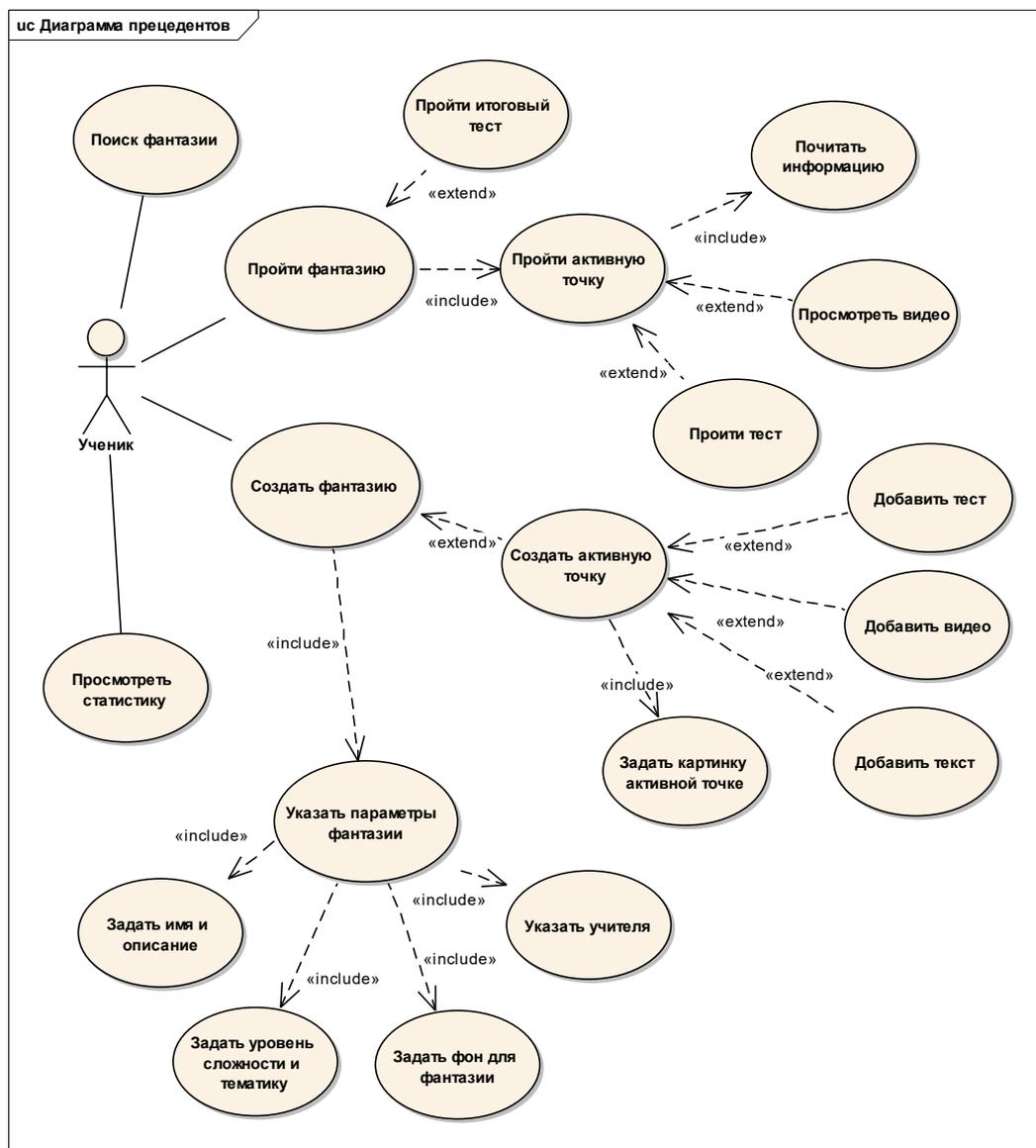


Рисунок 1. – Диаграмма прецедентов для роли Ученик

Кроме функций, отображенных на диаграмме для роли Ученик, приложение также должно выполнять методы передачи информации о баллах на платформу Stimey. Баллы ученику начисляются за правильные ответы на тесты в активных точках, а также за создание своей собственной фантазии. Интеграция с платформой не отображается на представленных диаграммах. Переданные на платформу баллы повышают рейтинг ученика, что дополнительно способствует его мотивации в прохождении или создании фантазий.

В целях безопасности и обеспечения качества материала для роли Учитель заданы более расширенные права. В качестве дополнительных функций у Учителя присутствует возможность публиковать фантазии учеников (прецедент Опубликовать фантазию ученика). При этом на учителя направляются только те фантазии, где сам ученик указал этого учителя в настройках своей фантазии.

Для удобства учителя фантазии может быть задан пароль, что обеспечивает ограничение для доступа к прохождению конкретной фантазии (прецедент Задать пароль для фантазии). Таким образом, учитель регулирует состав группы, кто может проходить ту или иную фантазию. Пароль может быть задан как для фантазии, где автором является сам учитель, так и для фантазии, созданной его учеником.

Расширенная статистика позволяет учителю просмотреть данные о том, кто из учеников проходил конкретную фантазию (прецедент Просмотреть статистику прохождения фантазий) и какие баллы на тестах в активных точках ими были получены (прецедент Просмотреть статистику по каждому ученику).

Диаграмма прецедентов для роли Учитель представлена на рисунке 2.

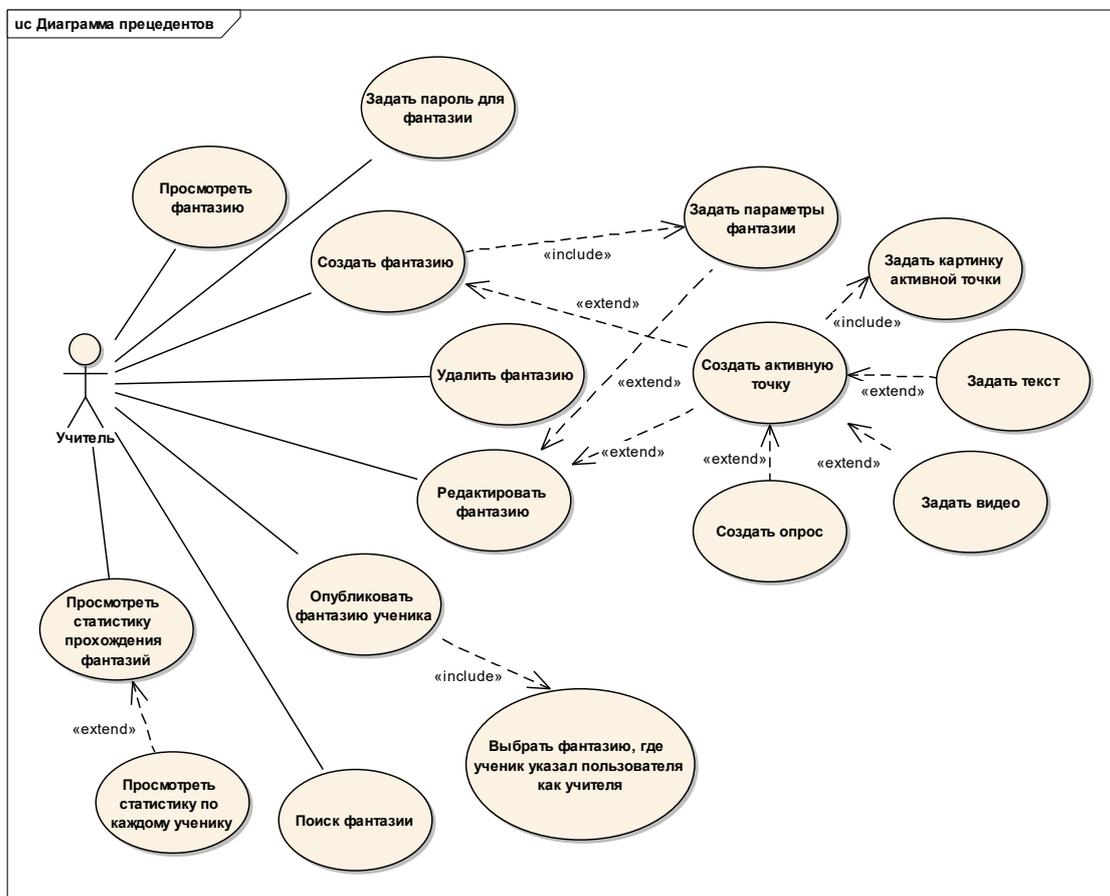


Рисунок 2. – Диаграмма прецедентов для роли Учитель

Полученные диаграммы прецедентов показывают функции приложения «Fantasy» разных ролей. По диаграмме видно, что роли имеют ряд пересекающихся функций, а у роли Учитель более расширенные права.

Заключение. В результате работы было спроектировано программное обеспечение, удовлетворяющее всем требованиям проекта. «Fantasy» поддерживает два вида пользователей: Ученик и Учитель, каждый из которых обладает разным набором прав. Одна из главных целей разработки – развитие креативности – достигается в приложении за счет возможности создавать учеником свои «фантазии», применяя множество различных инструментов и представлять их в новой форме. Ребенок может в полной мере применить все свои творческие способности: от создания графического материала до подготовки каверзных опросов. Работа в конечном счете проверяется и публикуется учителем. В результате «фантазия» – это не просто реферат ребенка на определенную тему, а полноценный ресурс, который может применяться в учебном процессе.

ЛИТЕРАТУРА

1. Гаврилова, Г.Н. Улучшение качества жизни детей формированием флексибельности и креативности / Гаврилова Г.Н. – Фундаментальные исследования. 2008. № 2. С. 64.
2. Куриленко, И.В. проблема развития креативности детей с помощью института дополнительного образования / Куриленко И.В. – Власть. 2012. № 9. С. 128–130.
3. Романцов, М.Г. креативность и педагогический процесс / Романцов М.Г., Рыбалкин А.С. – Современные проблемы науки и образования. 2006. № 1. С. 86-87.

УДК 004.9

**РАЗРАБОТКА И ОРГАНИЗАЦИЯ РАЗРАБОТКИ
ОБУЧАЮЩЕГО WEB-ПРИЛОЖЕНИЯ «FANTASY»****М. НЕПЕСОВ, Р. АБДУЛЛАЕВ**
(Представлено: Т.М. ГЛУХОВА)

В статье описаны технологии, применяемые при разработке Web-приложения «Fantasy», а также рассмотрены вопросы, связанные с организацией процесса разработки программного обеспечения.

Задачи и цели проекта. В современном мире всё большее распространение получают онлайн-курсы. Они представляют собой одну из форм дистанционного образования, когда взаимодействие ученика и учителя происходит на расстоянии. Однако несмотря на это, сохраняются все основные компоненты образовательного процесса: цели, содержание, методы, организационные формы и средства обучения. Онлайн-курс определяется, как “вид электронного обучения, то есть организованный целенаправленный образовательный процесс, построенный на основе педагогических принципов, реализуемый на основе технических средств современных информационных технологий и представляющий собой логически и структурно завершённую учебную единицу, методически обеспеченную уникальной совокупностью систематизированных электронных средств обучения и контроля”. В данный момент широкое распространение получил частный вид онлайн-курсов – массовые открытые онлайн-курсы (МООК). Особенностью данного вида является открытый доступ к учебным и тестовым материалам [1].

Одной из главных трудностей массовых онлайн-курсов является формирование у слушателей практических навыков. Очевидно, что в условиях ограниченных возможностей для прямого взаимодействия учителя и ученика формирование практических навыков становится нетривиальной задачей, требующей поиска инновационных решений [2].

Таким образом, одной из главных задач на современном этапе при создании обучающих web-приложений является применение нестандартных подходов. Такая цель достигалась и при создании приложения «Fantasy». Но главная цель приложения: развитие креативности. Материал преподносится в нестандартной форме – в виде так называемых фантазий. Фантазия – это некоторое задание, которое включает теоретический материал и тестовое задание, а отображается она в виде рабочей области с активными точками (активной точкой является изображение, которое реагирует на нажатие мышкой). В этих активных точках находятся теоретический материал: текст, аудио и видео, а также тестовые задания (рисунок 1).

Преподаватель создает фантазию таким-образом что при прохождении ученика более легкого теста для него открывается следующая активная точка, которая содержит следующий материал и тест по заданной теме.

Одна из главных идей приложения – сам ученик имеет возможность создавать фантазию. Эта фантазия будет иметь те же функции, что и фантазия учителя, но при одном условии: общий доступ к этой фантазии может открыть только выбранный преподаватель после проверки. Такая функция позволяет заставлять учеников креативно мыслить: искать и подбирать материалы по нужной теме, нестандартно оформлять материал и создавать тесты по теме.

Порядок работы с фантазией. На основании ранее разработанных диаграмм прецедентов была спроектирована следующая логика работы приложения.

Для того, чтобы создать фантазию требуется выполнить следующие шаги:

1. Нажать кнопку Создать;
2. На первом шаге ввести имя создаваемой фантазии и выбрать файл картинки для отображения его на карточке фантазии в общем списке фантазий;
3. На втором шаге выбрать тематику фантазии, сложность и задать краткое описание;
4. На втором шаге также требуется указать один из параметров в зависимости от того, какая роль у пользователя, который работает с фантазией:
 - Роль Учитель: указывается уровень доступа к фантазии. Существует 2 уровня открытый (public – открыто для всех пользователей) и закрытый (private – открыто только для автора и преподавателя), а также возможность задать пароль для ограничения доступа к фантазии;
 - Роль Ученик: указывается учитель, который может проверить и открыть доступ для фантазии другим пользователям.

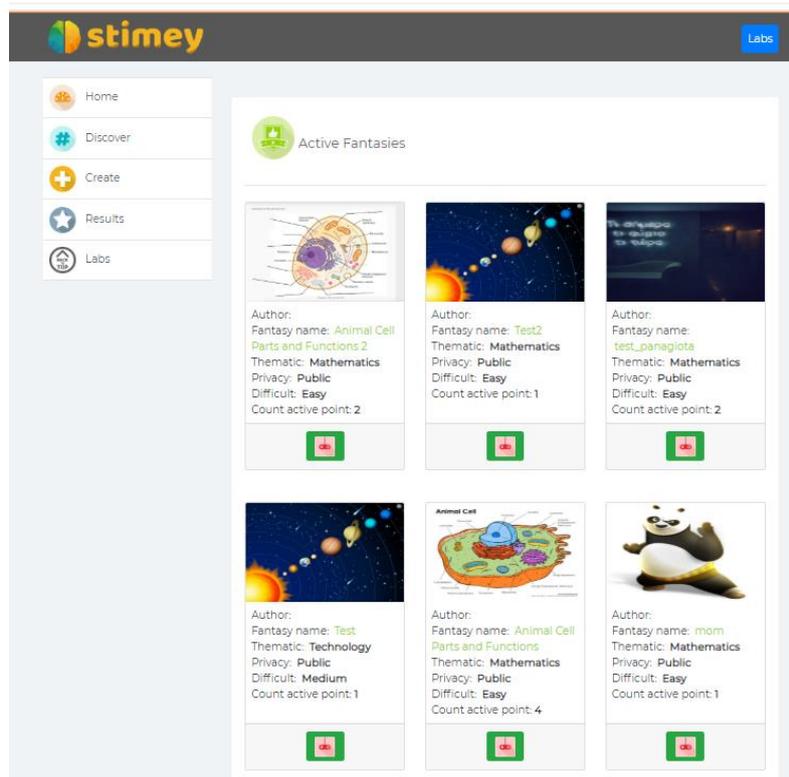


Рисунок 1. – Список фантазий приложения Fantasy

5. На третьем шаге указывается фон (картинка, background) для рабочей области фантазии;
6. После отображения рабочей области можно создавать активные точки (рисунок 2).

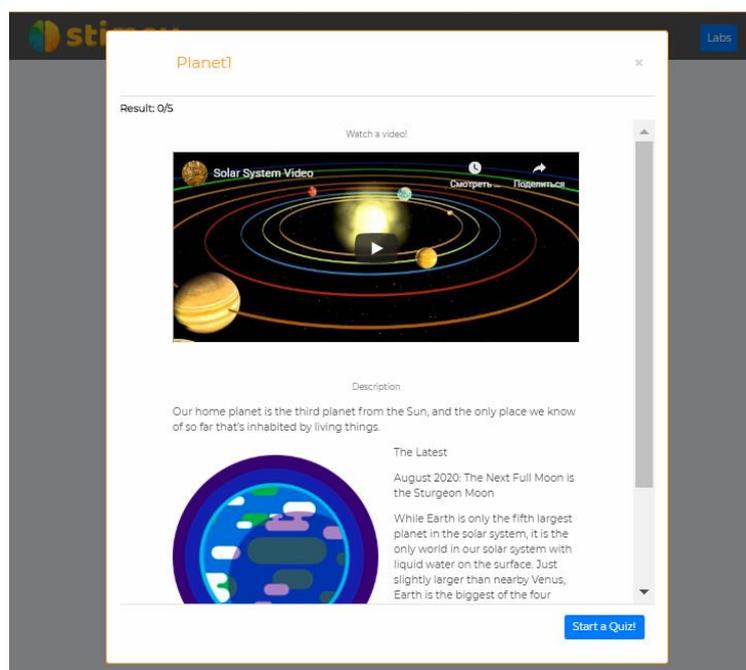


Рисунок 2. – Информация по активной точке с возможностью пройти тест и посмотреть результат

7. После появления активной точки на рабочей области можно задать ее размеры (растянуть по краям прямоугольной области с помощью мыши), сместить в другую точку на рабочей области (перетащить мышкой) и указать ее характеристики (двойной клик на активной точке):

- Для задания характеристик активной точке на первом шаге требуется ввести имя активной точки, выбрать изображение для активной точки и выбрать порядок следования активной точки;
- На втором шаге требуется ввести текстовое описание активной точки (обучающий материал для других пользователей) и URL на видео для дополнительной информации;
- На третьем шаге создается тест для проверки знаний о пройденном материале: выбрать тип вопроса, количество баллов за правильный ответ и правильные/неправильные ответы.

8. Сохранить активную точку.

После создания фантазии в зависимости от указанного уровня доступа она может быть доступна для других пользователей.

Технологии разработки. Для разработки web-приложения были выбраны следующие технологии:

- Node JS — кроссплатформенная среда выполнения для JavaScript, которая выполняется на серверах. Одной из основных особенностей Node.js является скорость выполнения, что является ее преимуществом по сравнению с C или Java;
- React JS - это библиотека JavaScript, которая используется для создания пользовательского интерфейса. React - инструмент для создания масштабируемых веб-приложений;
- PostgreSQL - свободная объектно-реляционная система управления базами данных (СУБД).

Организация процесса разработки. В процессе разработки программного обеспечения применялись современные подходы к организации и применяемым программным средствам. Так, для планирования, управления и коммуникации применялась система управления проектами Битрис 24, которая помогла разбить проект на отдельные подзадачи, следить за ходом выполнения работ, общаться между разработчиками и планировать проект по средствам диаграммы Ганта.

Для работы с исходными кодами проекта применялся Git, с помощью которого осуществлялся контроль версий исходного кода с возможностью восстановления версий в случае обнаружения проблем в ходе разработки.

Для тестирования готовой системы применялась методика на основе ранее составленных вариантов использования (Use case). Само тестирование осуществлялось вручную, без применения средств автоматизации тестирования.

Интеграция с внешней платформой Stimey осуществлялась по средствам взаимодействия по протоколу REST API.

Заключение. В результате работы было реализовано web-приложение «Fantasy». Приложение выполнено с помощью современных средств проектирования и реализации, а также с применением средств организации процесса разработки и тестирования программного обеспечения.

ЛИТЕРАТУРА

1. Диканская, Ю.В. Онлайн-платформа stepik для создания массовых открытых онлайн-курсов / Диканская Ю.В. – В сборнике: Европейские научные исследования. сборник статей V Международной научно-практической конференции. 2020. С. 183-185.
2. Белоглазов, А.А. Образовательные технологии онлайн-обучения: анализ массовых открытых онлайн-курсов российских вузов / Белоглазов А.А., Белоглазова Л.Б., Белоглазова И.А., Мальцев О.Л., Трубащев Е.В., Никифорова С.А., Попенко В.В. – Вестник Московского городского педагогического университета. Серия: Информатика и информатизация образования. 2018. – № 4 (46). С. 50-57.

УДК 004.021

ИССЛЕДОВАНИЕ ТЕХНОЛОГИИ ВЗАИМОДЕЙСТВИЯ С БАЗОЙ ДАННЫХ ДЛЯ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ТУРИСТИЧЕСКОГО ЦЕНТРА

А.И. ПОПОВ*(Представлено: канд. техн. наук, доц. И. Б. БУРАЧЕНОК)*

В данной статье рассмотрены принципы взаимодействия с реляционными базами данных на языке программирования Java посредством технологии JDBC. Приведены преимущества разделения архитектуры приложения непосредственно на саму реализацию приложения и отделинный от него драйвер. Обосновано применение представленной технологии для реализации интерфейса для разрабатываемого приложения.

В каждом современном языке программирования существует собственный набор методов и средств для разработки системы взаимодействия с базами данных (БД). В языке программирования Java, для данной цели, используется такая технология как – Java DataBase Connectivity (JDBC) – платформенно-независимый промышленный стандарт взаимодействия Java-приложений с различными СУБД, реализованный в виде пакета java.sql, входящего в состав Java SE.[1]

Далее подробнее остановимся на особенностях взаимодействия между реализованным приложением и базой данных с использованием стандартного Application Programming Interface (API) JDBC. Если говорить в целом, то JDBC – это библиотека, которая обеспечивает целый набор интерфейсов для доступа к различным БД для решения поставленных задач. Наиболее распространённые цели, для которых может использоваться JDBC:

- создание соединения с БД;
- создание SQL выражений;
- выполнение SQL – запросов;
- просмотр и модификация полученных записей.

В общем виде JDBC состоит из двух слоёв: JDBC API – Обеспечивает соединение «приложение – JDBC Manager». JDBC Driver API – Обеспечивает соединение «JDBC Manager – драйвер». JDBC API использует менеджер драйверов и специальные драйверы БД для обеспечения подключения к различным базам данных. JDBC Manager проверяет соответствие драйвера и конкретной БД. Он поддерживает возможность использования нескольких драйверов одновременно для одновременной работы с несколькими видами БД.

Схематично, JDBC можно представить в виде архитектуры, представленной на рисунке 1:

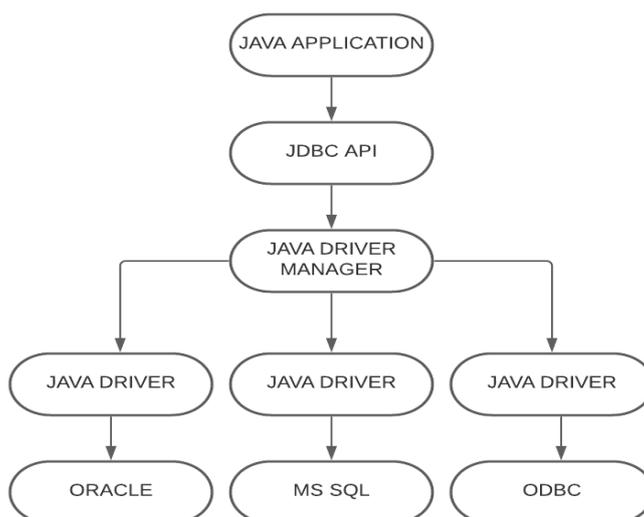


Рисунок 1. – Архитектура взаимодействия JDBC с базой данных

Рассмотрим элементы JDBC по отдельности.

– **Менеджер драйверов (Driver Manager)**. Этот элемент управляет списком драйверов БД. Каждый запрос на соединение требует соответствующего драйвера. Первое совпадение даёт нам соединение.

– **Драйвер (Driver)**. Этот элемент отвечает за связь с БД. Работать с ним нам приходится крайне редко. Вместо этого чаще используют объекты DriverManager, которые управляют объектами этого типа.

– **Соединение (Connection)**. Этот интерфейс обеспечивает нас методами для работы с БД. Все взаимодействия с БД происходят исключительно через Connection.

– **Выражение (Statement)**. Для подтверждения SQL-запросов используют объекты, созданные с использованием этого интерфейса.

– **Результат (ResultSet)**. Экземпляры этого элемента содержат данные, которые были получены в результате выполнения SQL – запроса. Он работает как итератор и «пробегают» по полученным данным.

– **Исключения (SQL Exception)**. Этот класс обрабатывает все ошибки, которые могут возникнуть при работе с БД. [2]

Пример создания подключения к базе данных с использованием JDBC представлен в листинге 1.

Листинг 1 – Пример создания подключения к базе данных с использованием JDBC.

```
public class Connect {
    public static Connection conn = null;
    public static Statement stmt = null;
    public static Connection connectToDatabase() {
        Locale.setDefault(Locale.ENGLISH);
        String url = "jdbc:sqlserver://localhost:1433";
        try {
            if (conn == null) {
                try {
                    Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
                } catch (ClassNotFoundException e) {
                    System.out.println(e.getMessage());
                }
                Properties props = new Properties();
                props.setProperty("databaseName", "TyrCentre");
                props.setProperty("user", "tmp");
                props.setProperty("password", "619");
                conn = DriverManager.getConnection(url, props);
                return conn;
            }
        } catch (SQLException e) {
            e.printStackTrace();
            System.out.println(e.getMessage());
        }
        return conn;
    }
    public static Statement getStatement() {
        try {
            if (stmt == null) {
                stmt = conn.createStatement();
                return stmt;
            }
        } catch (SQLException e) {
            System.out.println(e.getMessage());
        }
        return stmt;
    }
}
```

После установки соединения с базой данных можно взаимодействовать с таблицами при помощи объекта Result. Получение данных из БД с использованием JDBC представлено в листинге 2.

Листинг 2 – Получение данных из БД с использованием JDBC.

```
try {
    ResultSet rs = Connect.stmt.executeQuery("SELECT name\n" +
        "FROM Place");
    while (rs.next()) {
        list.add(rs.getString(1));
    }
} catch (SQLException ex) {
    System.out.println(ex.getMessage());
}
```

Таким образом, разделение архитектуры приложения непосредственно на саму реализацию приложения и отделённый от него драйвер имеет ряд преимуществ:

- лёгкость разработки: разработчик может не знать специфики базы данных, с которой работает;
- код практически не меняется, если приложение переходит на другую базу данных (количество изменений зависит исключительно от различий между диалектами SQL);
- нет необходимости устанавливать громоздкую клиентскую программу;
- к любой базе можно подсоединиться через легко описываемый унифицированный указатель ресурса URL (Uniform Resource Locator).

Следовательно, используя данную технологию доступа к базам данных можно организовать взаимодействие приложения с SQL-сервером с минимальными затратами ресурсов. Так как JDBC является частью языка Java, то не потребуются использование сторонних ресурсов. Кроме того, использование данной технологии нивелирует привязанность к какой-то определённой базе данных, приложение может взаимодействовать с другой базой данных с минимальными изменениями в настройках подключения и заменой используемого драйвера подключения при этом основной код приложения останется неизменным.

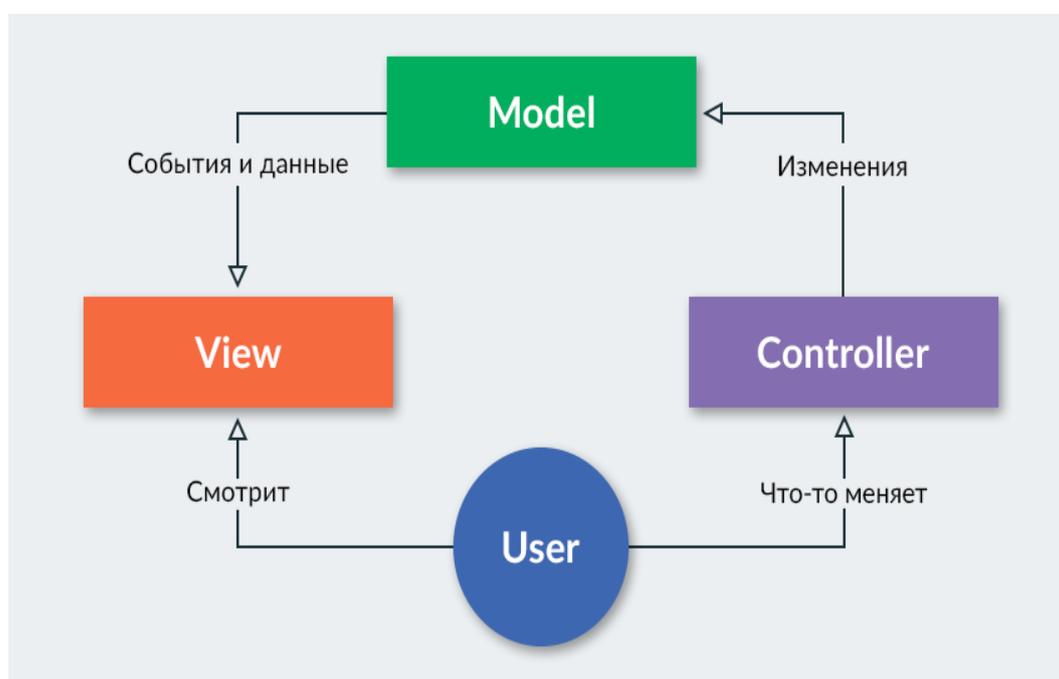
ЛИТЕРАТУРА

1. Java Database Connectivity. [Электронный ресурс] / Wikipedia. – Режим доступа: https://ru.wikipedia.org/wiki/Java_Database_Connectivity/. – Дата доступа: 19.09.2020.
2. Введение в JDBC. [Электронный ресурс] / Codeflow. – Режим доступа: <https://www.codeflow.site/ru/article/java-jdbc/>. – Дата доступа: 19.09.2020.

**ИСПОЛЬЗОВАНИЕ АРХИТЕКТУРНОГО ПАТТЕРНА MVC
ПРИ ПРОЕКТИРОВАНИИ СИСТЕМЫ ТУРИСТИЧЕСКОГО ЦЕНТРА****А.И. ПОПОВ***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)*

В данной статье рассмотрены принципы построения взаимодействия компонентов в приложении на языке программирования Java. Рассмотрим основные идеи паттерна MVC и задачи, которые нужно реализовать для следования паттерну. Представлены преимущества при разработке системы по модели MVC.

Как правило, современные приложения строятся по шаблону MVC (Model-View-Controller или Модель-Вид-Контроллер). MVC – это не паттерн проектирования. MVC – это именно набор архитектурных идей и принципов для построения сложных систем с пользовательским интерфейсом. [1] Но для удобства его обычно называют шаблоном или паттерном. Идея данного шаблона проста – разделение обязанностей. Таким образом, любая разрабатываемая система должна быть разделена на три составные части, каждая из которых отвечает за свои задачи как показано на рисунке 1.

**Рисунок 1. – Структура MVC приложения**

Задача контроллера – обработка действий пользователя (нажатие по кнопкам, обработка запросов к серверу и т. д.); модель – предоставляет контроллеру представление данных, запрашиваемых пользователем, содержит всю бизнес-логику приложения; вид – обеспечивает представление данных, полученных из модели. Данный модуль отвечает за отображение данных пользователю. Все, что видит пользователь, генерируется видом. [2]

Для того чтобы добиться реализации MVC нужно проделать несколько обязательных шагов.

Первый шаг – это разделение бизнес-логики приложения от пользовательского интерфейса. Таким образом приложение уже разбивается на два блока. Первый отвечает за реализацию бизнес-логики и будет являться ядром системы. В этом модуле и содержится реализация модели предметной области приложения. Второй модуль отвечает за интерфейс, который отображается пользователю и логику взаимодействия пользователя с приложением. Основная идея данного шага заключается в том, чтобы ядро системы, это то, что мы выделили как Model, могло быть разработано и протестировано независимо. Архитектура приложения после такого разделения показана на рисунке 2.

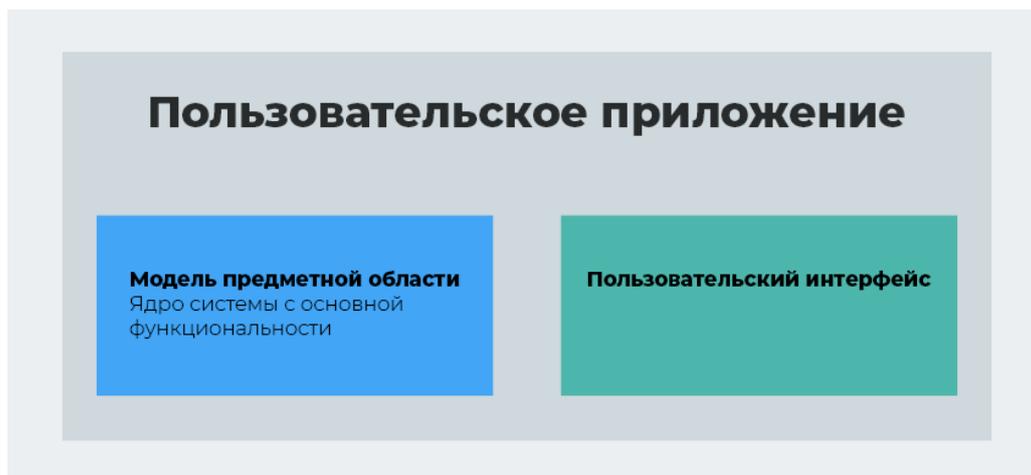


Рисунок 2. – Разделение приложения на модель и представление

Вторым шагом будет разделение приложения на ещё более низком уровне иерархии, то есть мы разделим представление, на два уровня, а именно на вид и контроллер. Вид – это то, что видит пользователь, а контроллер – это механизм, посредством которого пользователь может взаимодействовать с системой. Важно понимать, что элементы управления, например, кнопки на странице – это, по сути, часть контроллера. Но они также видны пользователю, как и любая часть вида. Поэтому при этом разделении подразумевается функциональное разделение интерфейса на части как показано на рисунке 3. Так как у интерфейса можно выделить две основные функции:

- выводить и удобно отображать пользователю информацию о системе
- вводить данные и команды пользователя в систему (передавать их системе) то по этим функциям мы и проводим разделение.

Первую функцию будет выполнять вид, а вторую контроллер.

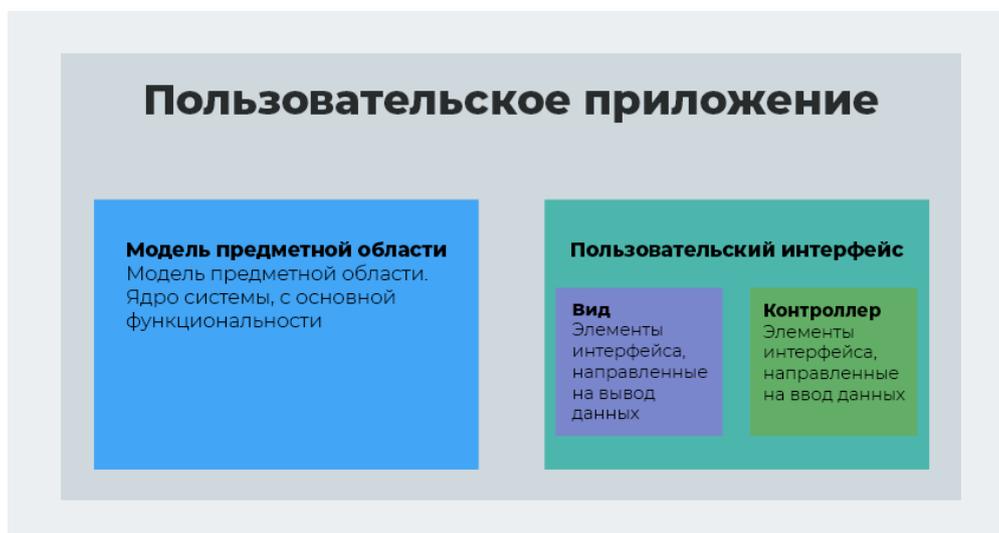


Рисунок 3. – Разделение приложения на модель, вид и представление

Когда пользователь вводит информацию через контроллер, он тем самым вносит изменения в модель. Точнее, пользователь вносит изменения в данные модели. Когда пользователь получает информацию через элементы интерфейса (через Вид), пользователь получает информацию о данных модели.

Следование данным правилам позволяет добиться следующих преимуществ: основная цель следования принципам MVC – отделить реализацию бизнес-логики приложения (модели) от ее визуализации (вида). Такое разделение повысит возможность повторного использования кода. Польза применения MVC наиболее наглядна в случаях, когда пользователю нужно предоставлять одни и те же данные в разных формах, например, в виде таблицы, графика или диаграммы, используя различные виды. При этом, не затрагивая реализацию видов, можно изменить реакции на действия пользователя (нажатие мышью на

кнопке, ввод данных). Также, если следовать принципам MVC, можно упростить написание программ, повысить читаемость кода, сделать легче расширение и поддержку системы в будущем. [3]

Рассмотрев принципы реализации MVC, построим примерную модель разбиения проектируемой системы для соблюдения требований паттерна. Модель структуры системы представлена на рисунке 4.

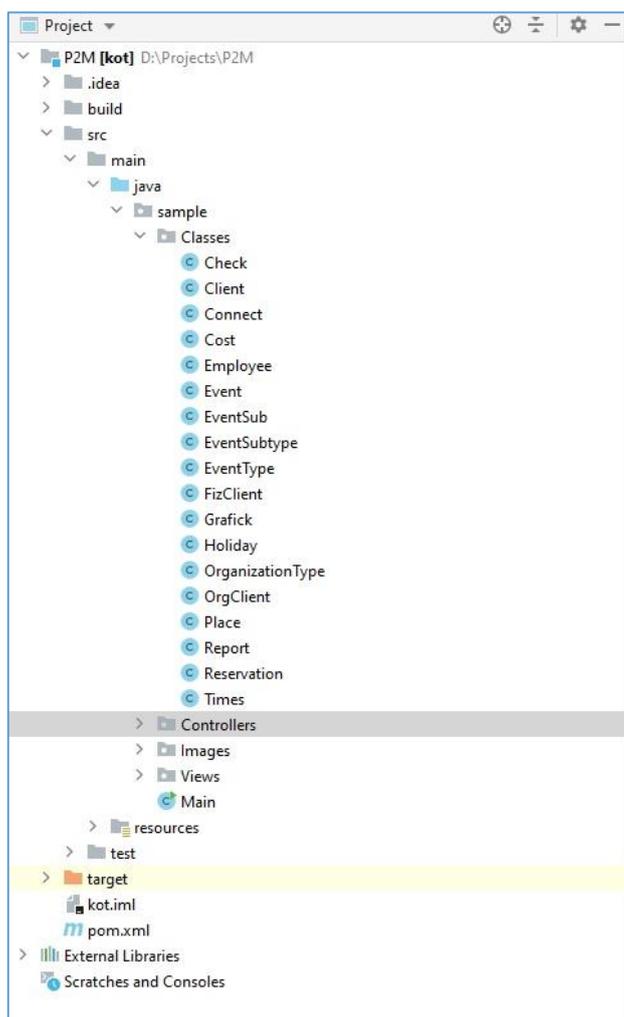


Рисунок 4. – Схема модели проекта

Проектирование системы туристического центра с использованием архитектурного паттерна MVC позволит упростить дальнейшее масштабирование системы. Кроме того, разделение приложения на составные части позволит разделить ответственность между различными компонентами системы и упростит дальнейшую разработку и реализацию новых функциональных решений.

ЛИТЕРАТУРА

1. Model-View-Controller. [Электронный ресурс] / Wikipedia. – Режим доступа: <https://ru.wikipedia.org/wiki/Model-View-Controller>. – Дата доступа: 19.09.2020.
2. Что такое паттерн проектирования MVC в Java. [Электронный ресурс] / Pro-java. – Режим доступа: <https://pro-java.ru/patterny-proektirovaniya-java/chto-takoe-pattern-proektirovaniya-mvc-v-java/>. – Дата доступа: 18.09.2020.
3. MVC [Электронный ресурс] / Javarush. – Режим доступа: <https://javarush.ru/quests/lectures/questcollections.level06.lecture01/>. – Дата доступа: 18.09.2020.

УДК 004.415.25

РЕАЛИЗАЦИЯ ИНТЕРАКТИВНОГО ИЗМЕНЕНИЯ МАТЕРИАЛА ОБЪЕКТА В ИГРОВОМ ПРИЛОЖЕНИИ

М.А. СЕРГЕЕВ

(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИНЫМ)

В статье представлен практический способ создания механики интерактивного изменения текстур объектов в приложениях, разрабатываемых на Unreal Engine 4. Задача, реализация интерактивного изменения текстуры объекта в игровом приложении.

Ключевые слова: информационные технологии, текстуры, материалы, Unreal Engine 4.

Введение. Практически любые игровые приложения основаны на принципе интерактивности пользователя с игровой средой. Наиболее же яркий эффект отзывчивости для пользователя достигается визуальным путём. Так как объекты игровой среды представлены не только формой, но и материалом, сконструированным из текстур, изменение материалов сделает игровой мир «живым».

В данной статье будет представлена и рассмотрена реализация интерактивного изменения материала объекта в игровом приложении.

Реализация взаимодействия игрока с объектом.

Игровым движком для реализации приложения был выбран Unreal Engine 4. Таким образом, всё написание приложения будет вестись при помощи Blueprints, графического языка программирования.

В данной реализации взаимодействие пользователя с объектом представлено в общем виде через объект BP_GLURP, наследующийся от объекта BP_BUTTON. То есть, с точки зрения пользователя в игровой среде взаимодействие представлено в виде нажатия на объект кнопки.

В родительском объекте BP_BUTTON создаётся событие Button Action (Рисунок 1), реализация которого в различных дочерних объектах будет отличаться. Запуск этого события происходит в момент, когда виртуальное отображение игрового контроллера игрока пересекается коллизиями с объектом кнопки. Валидация такого пересечения происходит через функцию «Cast to...». Альтернативным решением в данном случае было бы проверить объект на наличие тега функцией «Actor Has Tag», однако такая реализация скорее применима к взаимодействию с разными видами объектов на игровом уровне, в данном же случае кнопка взаимодействует только с контроллером пользователя.



Рисунок 1. – Реализация взаимодействия игрового контроллера с объектом BP_Button

Реализация изменения материала объекта.

Объект, материал которого изменяется, в данном проекте представлен блюпринтом BP_Canvas (англ. холст). Интерактивное изменение материала реализуется в данном случае не через полную смену материала функцией «Set Material», так как при больших количествах разных материалов размер проекта может стать чрезмерно большим, а через изменение, то есть дополнение, либо же перерисовку текстур в динамическом материале. На рисунке 2 изложено создание такого динамического материала, который всамом объекте BP_Canvas представлен переменной Brush Material.

Соответственно изменение переменной Brush Material приведёт к изменению отображения материала объекта, которое существует только текущий момент отображения и опирается лишь на текстуру, либо текстуры, которые накладываются на двумерную карту материала Render Target 2D.

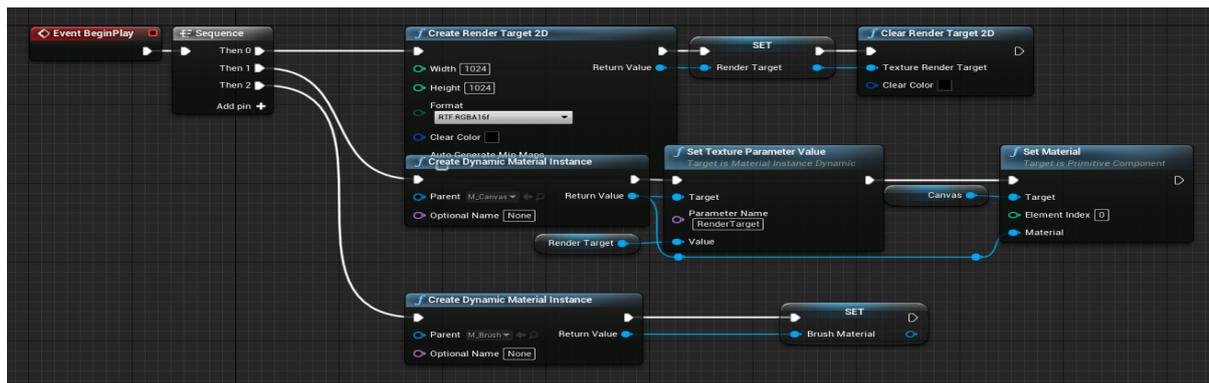


Рисунок 2. – Реализация создания и применения динамического материала к объекту VR_Canvas

Функция нанесения текстуры на динамический материал вызывается в объекте кнопки, сами текстуры хранятся в массиве в этом же объекте. На рисунке 3 представлена реализация события Button Action объекта VR_Glurp. При взаимодействии с объектом, в текстурной переменной Glurp присваивается значение из массива текстур. Эта текстура применяется к переменной динамического материала Brush Material, после чего, вызывается функция Draw Brush (Рисунок 4), которая и отвечает за обновление материала на объекте. В случае с данным проектом, к материалу применяются две разные текстуры, накладывание которых предупреждено тем, что одна из текстур вырезана по альфа-каналу, то есть имеет прозрачный фон.

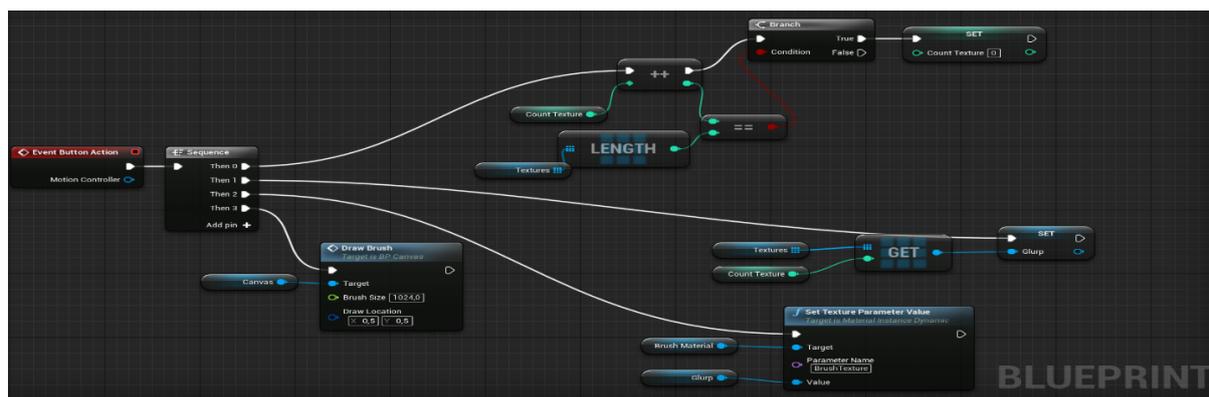


Рисунок 3. – Реализация события Button Action

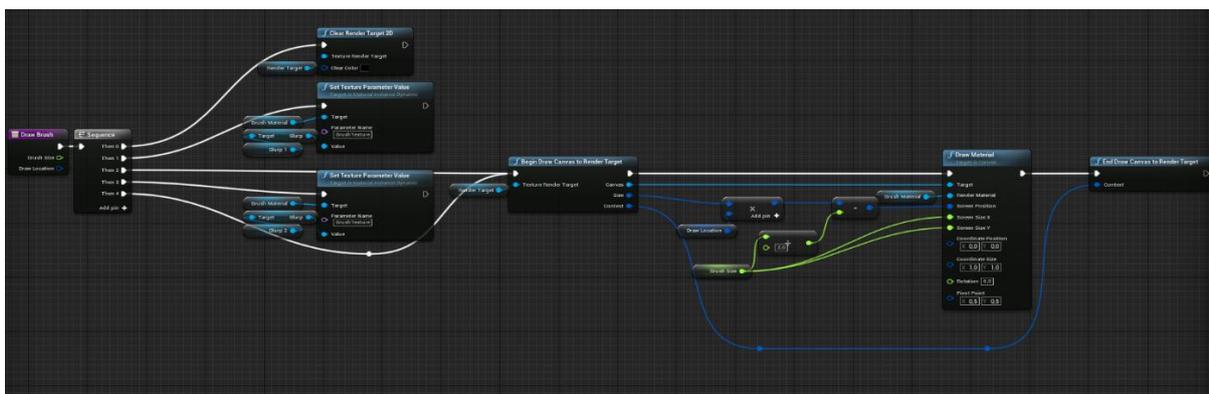


Рисунок 4. – Реализация функции Draw Brush

Заключение. В данном проекте реализован один из вариантов упрощенного изменения материалов объектов в игровой среде. Данный метод позволяет хранить в проекте только текстуры некоторых изменяемых объектов, используя при рендеринге в реальном времени только динамический материал, что уменьшает размер итогового проекта.

ЛИТЕРАТУРА

1. Unreal Engine VR для разработчиков / Митч Маккефри; [пер. с англ. Н.И. Веселко, О.В. Максименковой, А.А. Незнанова]. – Москва : Эксмо, 2019.

РЕАЛИЗАЦИЯ ГОЛОВОЛОМКИ «ПАЗЛ» В ИГРОВОМ ПРИЛОЖЕНИИ ВИРТУАЛЬНОЙ РЕАЛЬНОСТИ

М.А. СЕРГЕЕВ

(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИН)

В статье представлен практический способ создания пазла в приложениях виртуальной реальности разрабатываемых на Unreal Engine 4. Задача, реализация пазла с возможностью смены изображений.

Ключевые слова: информационные технологии, текстуры, материалы, Unreal Engine 4.

Введение. Один из простейших в реализации видов игровых приложений – пазл. Сбор некоторой картинки зачастую реализуется в формате двухмерной игры. Однако, механики, которые подходят под реализацию такой головоломки могут широко применяться в различных игровых проектах.

В данной статье будет представлена и рассмотрена реализация пазла с возможностью смены картинок.

Разработанная головоломка состоит из нескольких сущностей: BP_Motion Controller (контроллер игрока), BP_Stand (поле, на котором лежат элементы головоломки), BP_Canvas (оригинальное изображение), BP_Puzzle (элемент головоломки), BP_Glurp (объект, взаимодействие с которым сменяет изображения как на BP_Canvas, так и на всех 12-ти элементах BP_Puzzle).

Реализация перемещения элемента головоломки.

Проект, рассматриваемый в статье разработан на игровом движке Unreal Engine 4 для систем виртуальной реальности. Благодаря этому перемещение самих элементов можно реализовать привязкой («захватом») к контроллеру игрока, что доступна в базовом шаблоне VR приложений.

Расположение же элементов на поле фиксируется по триггерам Box-Box11, расположенным на модели объекта BP_Stand (Рисунок 1). При пересечении коллизии элемента с триггером, позиция элемента модифицируется функцией SetActorLocationAndRotation, придавая расположению элемента значение расположения триггера. Функция AttachComponentToComponent же закрепляет элемент в этом положении, чтобы симуляция гравитации перестала влиять на элемент. Фрагмент кода представленный на рисунке 1 аналогичен для остальных одиннадцати элементов.

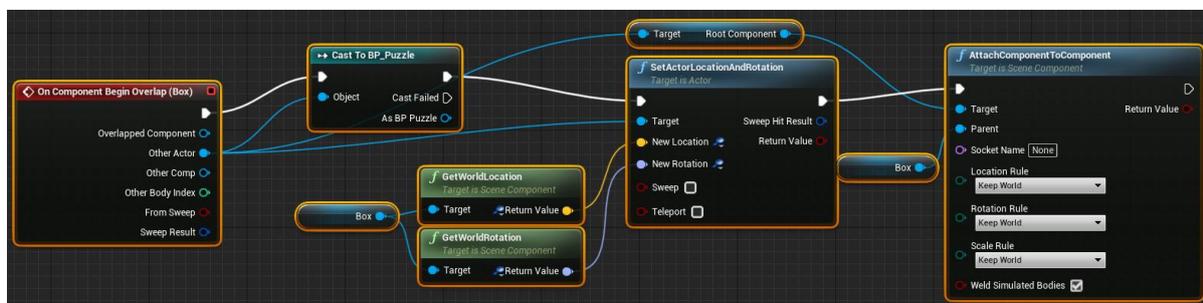


Рисунок 1. – Реализация привязки элемента головоломки к позиции на поле

Реализация изменения изображения.

Изменение изображения происходит при пересечении коллизий контроллера игрока и объекта BP_Glurp. В BP_Glurp содержится несколько массивов с текстурами, которые применяются на динамические материалы, отображающиеся на BP_Stand и 12-ти экземплярах BP_Puzzle (Рисунок 2,3).

На рисунке 2 представлен выбор одного из изображений в зависимости от порядка взаимодействия игрока с объектом BP_Glurp, то есть линейное переключение по счётчику. При выборе одного из паттернов происходит вход в цикл For, в котором текстуры из массива направляются в функцию “Draw” для каждого из элементов головоломки. Применение текстуры к BP_Stand происходит аналогичным образом до представленного фрагмента графического кода.

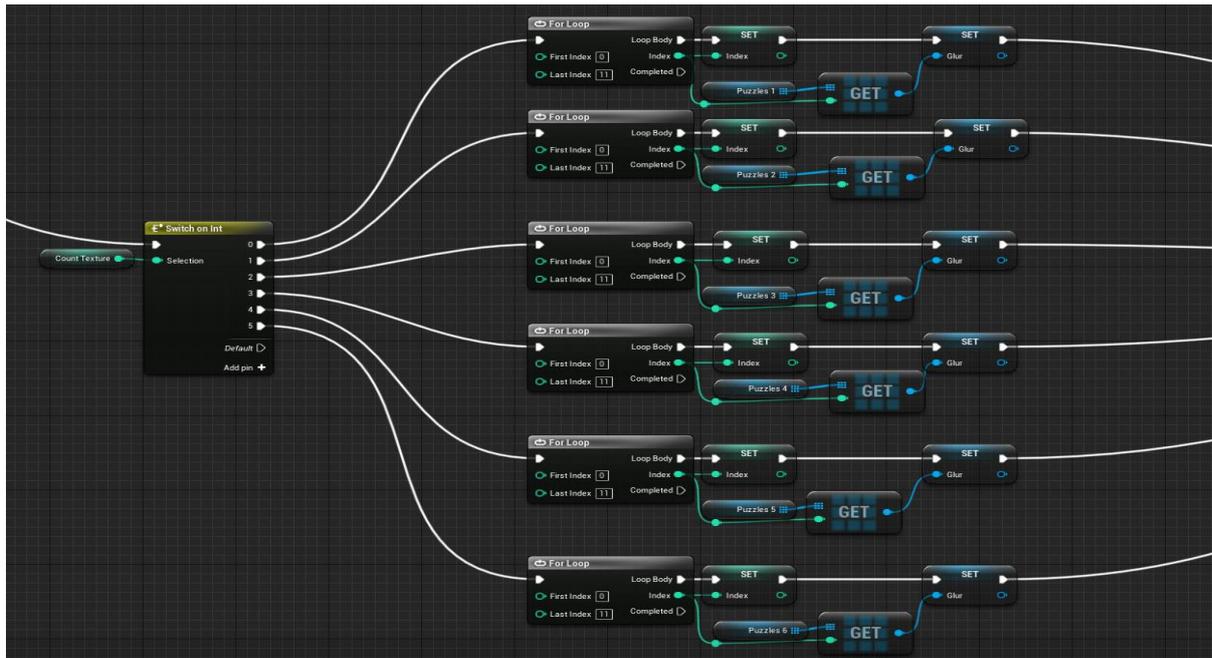


Рисунок 2. – Разветвление в зависимости от выбранного изображения

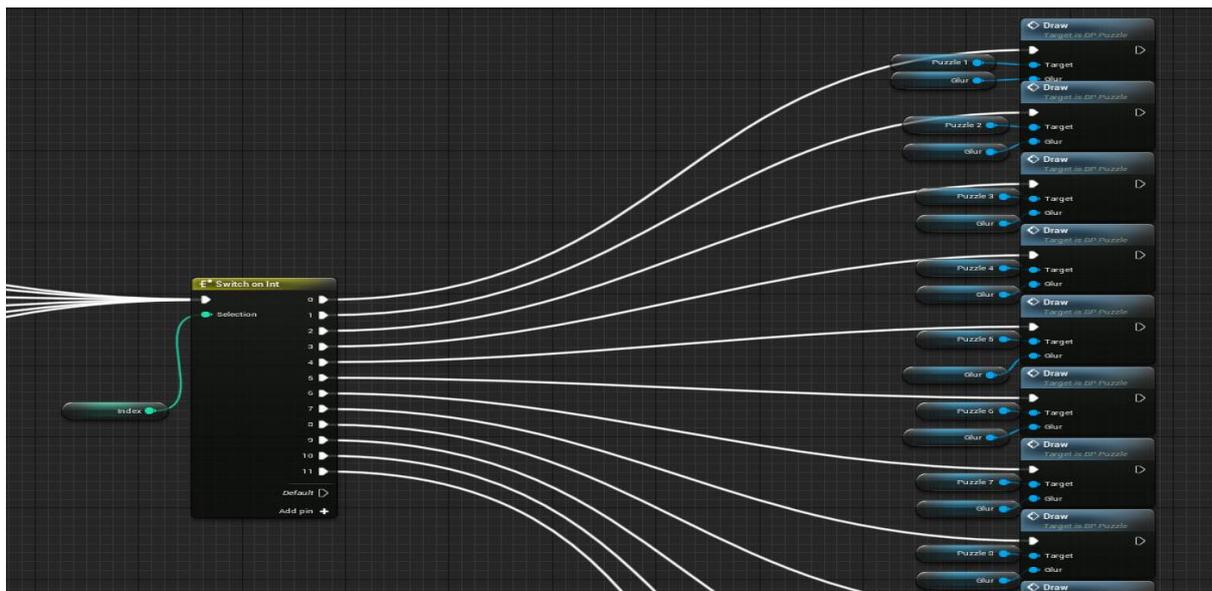


Рисунок 3. – Применение текстур, выбранных на предыдущем шаге к динамическим материалам элементов головоломки

Заключение. В данной реализации головоломки «пазл» для приложений виртуальной реальности продемонстрирован подход синхронного изменения материалов объектов с использованием динамических материалов. Подобный подход позволяет хранить в проекте только текстуры, не создавая материал для каждого варианта объекта, для облегчения объёма проекта.

Однако данный подход рационально использовать в более камерных проектах, которые не полагаются на сложные фотореалистичные материалы.

ЛИТЕРАТУРА

1. Unreal Engine VR для разработчиков / Митч Маккефри; [пер. с англ. Н.И. Веселко, О.В. Максименковой, А.А. Незнанова]. – Москва : Эксмо, 2019.

УДК 004.056

**ТЕХНОЛОГИИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ ИНФОРМАЦИИ
ПРИ ПЕРЕДАЧЕ ДАННЫХ МЕЖДУ НЕСКОЛЬКИМИ ПРЕДПРИЯТИЯМИ
ПО СЕТИ ИНТЕРНЕТ****А.С. СИВОГРАКОВ***(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

В данной работе рассмотрим и проанализируем основные технологии обеспечения безопасности информации при передаче данных по сети Интернет.

В настоящее время главным инструментом управления бизнесом и фактически важнейшим средством производства становятся корпоративные информационные системы. Основными видами деятельности для многих компаний становятся электронная коммерция, продажа информации в режиме on-line и другие услуги. Так же из-за того, что большинство компаний не ограничивают себя территориально им приходится реализовывать каналы для передачи данных между офисами. Для этого удобно использовать уже существующую сеть Интернет. Но передача данных по открытым сетям не гарантирует их безопасность.

Информация, обрабатываемая в корпоративных сетях, является особенно уязвимой, чему способствуют:

1. увеличение объемов обрабатываемой, передаваемой и хранимой в компьютерах информации;
2. сосредоточение в базах данных информации различного уровня важности и конфиденциальности;
3. расширение доступа круга пользователей к информации, хранящейся в базах данных, и к ресурсам вычислительной сети;
4. увеличение числа удаленных рабочих мест;
5. широкое использование глобальной сети Интернет и различных каналов связи;
6. автоматизация обмена информацией между компьютерами пользователей.

Для комплексной защиты от угроз и гарантии экономически выгодного и безопасного использования коммуникационных ресурсов для электронного бизнеса необходимо:

1. проанализировать угрозы безопасности для системы электронного бизнеса;
2. разработать политику информационной безопасности;
3. защитить внешние каналы передачи информации, обеспечив конфиденциальность, целостность и подлинность передаваемой по ним информации;
4. гарантировать возможность безопасного доступа к открытым ресурсам внешних сетей и Internet, а также общения с пользователями этих сетей;
5. защитить отдельные наиболее коммерчески значимые ИС независимо от используемых ими каналов передачи данных;
6. предоставить персоналу защищенный удаленный доступ к информационным ресурсам корпоративной сети;
7. обеспечить надежное централизованное управление средствами сетевой защиты.

Рассмотрим наиболее распространённые способы обеспечения безопасности корпоративной информации на предприятиях.

SSL. Протокол SSL применяется в качестве протокола защищенного канала, работающего на сеансовом уровне модели OSI. Этот протокол использует криптографические методы защиты информации для обеспечения безопасности информационного обмена. [4] Протокол SSL выполняет все функции по созданию защищенного канала между двумя абонентами сети, включая их взаимную аутентификацию, обеспечение конфиденциальности, целостности и аутентичности передаваемых данных. Ядром протокола SSL является технология комплексного использования асимметричных и симметричных криптосистем.

Взаимная аутентификация обеих сторон в SSL выполняется путем обмена цифровыми сертификатами открытых ключей пользователей (клиента и сервера), заверенными цифровой подписью специальных сертификационных центров. Протокол SSL поддерживает сертификаты, с помощью которых организуется выдача и проверка подлинности сертификатов.

Конфиденциальность обеспечивается шифрованием передаваемых сообщений с использованием симметричных сессионных ключей, которыми стороны обмениваются при установлении соединения. Сессионные ключи передаются также в зашифрованном виде, при этом они шифруются с помощью открытых ключей, извлеченных из сертификатов абонентов. Использование для защиты сообщений симметричных ключей свя-

зано с тем, что скорость процессов шифрования и расшифрования на основе симметричного ключа существенно выше, чем при использовании несимметричных ключей. Подлинность и целостность циркулирующей информации обеспечивается за счет формирования и проверки электронной цифровой подписи.

Такой способ защиты широко используется в мире Веб для приложений, в которых важна безопасность соединения, например, в платежных системах.

Протокол шифрования SSL также часто используется для шифрования каналов связи с базами данных.

Криптопровайдеры. Не возможность применения протокола шифрования SSL в некоторых случаях привела к появлению криптопротоколов. Криптопровайдер — это независимый модуль, позволяющий осуществлять криптографические операции в операционных системах, управление которым происходит с помощью функций CryptoAPI. То есть, это посредник между операционной системой, которая может управлять им с помощью стандартных функций CryptoAPI, и исполнителем криптографических операций.

Криптопровайдер должен обеспечивать:

1. реализацию стандартного интерфейса криптопровайдера;
2. работу с ключами шифрования, предназначенными для обеспечения работы алгоритмов, специфичных для данного криптопровайдера;
3. невозможность вмешательства третьих лиц в системы работы алгоритмов.

Использование криптопровайдера позволяет сделать шифрование данных прозрачным для любого приложения и любого протокола передачи данных.

Электронная цифровая подпись. Электронная цифровая подпись — реквизит электронного документа, позволяющий установить отсутствие искажения информации в электронном документе с момента формирования электронной цифровой подписи и проверить принадлежность подписи владельцу сертификата ключа электронной цифровой подписи. [2] Значение реквизита получается в результате криптографического преобразования информации с использованием закрытого ключа электронной цифровой подписи.

Цифровая подпись предназначена для аутентификации лица, подписавшего электронный документ. Кроме этого, использование цифровой подписи позволяет осуществить:

1. Контроль целостности передаваемого документа: при любом случайном ли преднамеренном изменении документа подпись станет недействительной, потому что вычислена она на основании исходного состояния документа и соответствует лишь ему.
2. Защиту от изменений документа: гарантия выявления подделки при контроле целостности делает поддельывание нецелесообразным в большинстве случаев.
3. Невозможность отказа от авторства. Так как создать корректную подпись можно, лишь зная закрытый ключ, а он должен быть известен только владельцу, то владелец не может отказаться от своей подписи под документом.
4. Доказательное подтверждение авторства документа: Так как создать корректную подпись можно, лишь зная закрытый ключ, а он должен быть известен только владельцу подписи под документом.

Все эти свойства электронной цифровой подписи позволяют использовать её для следующей целей:

1. Декларирование товаров и услуг (таможенные декларации).
2. Регистрация сделок по объектам недвижимости.
3. Использование в банковских системах.
4. Электронная торговля и госзаказы.
5. Контроль исполнения государственного бюджета.
6. В системах обращения к органам власти.
7. Для обязательной отчетности перед государственными учреждениями.
8. Организация юридически значимого электронного документооборота.
9. В расчетных и трейдинговых системах.

Крипто-Про CSP. КриптоПро CSP представляет собой криптопровайдер, средство криптографической защиты, предназначенное для обеспечения целостности программных приложений при помощи методов шифрования. [3] Также программное обеспечение позволяет защитить конфиденциальную информацию при обмене данными через интернет и обеспечить юридическую достоверность электронных документов.

В современном мире необходимо рассматривать вопросы обеспечения безопасности информации при передаче данных по открытым сетям, особенно это касается внутренней корпоративной информации на предприятиях, конфиденциальность которой стоит особо остро. Постоянное усовершенствование технологий угроз защите информации ведет к технологическому развитию средств защиты и наиболее перспективны на этом пути комплексные средства обеспечения доверенного сеанса связи, которые были представлены в этой статье.

1. Электронный ресурс «VPN». Режим доступа: <http://ru.wikipedia.org/wiki/VPN>. Дата доступа: 12.08.2020
2. Веденьёв Л.Т., Леонтьев С.Е. и Попов В.О. Вопросы повышения безопасности ключей пользователей в среде вычислительной системы. Доклад на конференции РусКрипто 2009.
3. Электронный ресурс «Крипто-Про CSP». – Режим доступа: http://www.cryptostandart.ru/showtopic/index.php?id=program_cryptopro. – Дата доступа: 12.08.2020
4. Шаньгин В. Ф. Информационная безопасность компьютерных систем и сетей. Москва ИД «ФОРУМ» – ИНФРА-М 2011.

УДК 003.26

ЭЛЕКТРОННАЯ ПОДПИСЬ: ПРЕИМУЩЕСТВА И НЕДОСТАТКИ**А.С. СИВОГРАКОВ***(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)*

В данной работе были проанализированы преимущества и недостатки электронной подписи.

В настоящее время что бы упростить подписание документов и заключение договоров без личного присутствия лиц, заключающих различного рода акты, используют электронные подписи. Но не смотря на преимущества данной технологии имеется и ряд её недостатков.

Эта тема востребована, ввиду широким внедрением в повседневную жизнь новых электронных технологий, в том числе внедрение технологии электронной цифровой подписи (ЭЦП) идет активно в различных государственных структурах. Само определение электронной подписи имеет следующую формулировку: это информация в электронно-цифровой форме, с помощью которой можно идентифицировать физическое или юридическое лицо без его личного присутствия [1].

Применение электронной цифровой подписи. Наиболее активное применение электронной цифровой подписи можно встретить в таких областях как:

1. система делопроизводства в электронном формате;
2. цифровая торговля;
3. ведение бухгалтерии;
4. бизнес;
5. платежные системы.

Назначение ЭП:

1. позволяет осуществить контроль целостности документа в электронном виде;
2. обеспечивает защиту данных от подделки и внесения изменений;
3. обеспечивает возможность подтверждения авторства владельца ЭП.

Существуют различные способы сформировать электронную подпись. Самым удобным и распространенным из них является создание ЭП с помощью электронного ключа.

Электронный ключ – ключевая пара, которая состоит из двух частей: открытой и закрытой. Оба этих ключа выдаются и создаются удостоверяющими центрами с помощью специальной программы шифрования (например, «Крипто-про»).

Закрытый ключ – или «Ключ электронной подписи» – уникальная последовательность символов, предназначенная для создания ЭП и для расшифровки сообщений. Это частная, приватная информация, которая известна только ее владельцу.

Закрытый ключ генерируется на рабочем месте пользователя с помощью средства криптографической защиты информации и сохраняется (только у пользователя) на съемный носитель (дискета, токен, смарт-карта) или в реестр Windows. Такой закрытый ключ необходимо хранить в секретном месте со всеми мерами предосторожностей.

На основе закрытого ключа создается открытый ключ (стоит сказать, что обратный процесс здесь невозможен, так как подобрать закрытый ключ по открытому ключу нельзя).

Открытый ключ – он же «Ключ проверки электронной подписи» – уникальная последовательность символов, предназначенная для проверки подлинности ЭП. Это открытая, общеизвестная информация доступна любому пользователю системы электронного документооборота.

Открытый ключ вычисляется из закрытого ключа и отправляется в Удостоверяющий центр в виде запроса на сертификат.

Сертификат является электронным (и/или бумажным) документом:

1. выдаётся на ФИО конкретного человека (должностного лица) - содержит персональные данные;
2. подписывается ЭП Удостоверяющего центра, который тем самым подтверждает его действительность;
3. сертификат в себе содержит открытый ключ Пользователя (поэтому открытый ключ называют сертификатом).

Определяются следующие виды ЭП: простая и усиленная электронные подписи. В свою очередь усиленная электронная подпись подразделяется на усиленную неквалифицированную электронную подпись (именуемую неквалифицированная электронная подпись) и усиленную квалифицированную электронную подпись (именуемую квалифицированная электронная подпись).

Простая электронная подпись. Простая ЭП – это максимально упрощенный вариант электронной подписи. Для ее создания не используются криптографические модули. Наиболее ярким примером, попадающим под определение простой электронной подписи, являются SMS-пароли, а также данные для доступа к различным информационным сервисам.

Часто используется вариант простой электронной подписи в банковских системах для подтверждения платежей и других операций. Она может также применяться в следующих случаях:

1. при получении госуслуг через официальный портал (доступны не все операции);
2. во внутренних системах документооборота (обмен между сотрудниками компании);
3. во внешних системах при наличии дополнительного соглашения;
4. при входе на различные сайты.

Данный вариант цифровой подписи недопустимо применять в системах, где приходится сталкиваться с гостайной. Это ограничение установлено на законодательном уровне.

Усиленная неквалифицированная электронная подпись. Усиленная неквалифицированная ЭП создается уже с помощью криптографического ПО. При этом используется закрытый ключ ЭП. Она позволяет проверить личность владельца, а также отсутствие изменений в подписанном файле.

При получении подписи владельцу передаются открытый и закрытый ключи ЭП. Открытый ключ ЭП нужен для проверки подлинности подписи, а закрытый используется в момент подписания файла (документа). Фактически открытый ключ доступен всем. Выдавать НЭП могут различные УЦ. Аккредитация для этого не требуется.

Довольно часто НЭП используется для участия в различных торгах.

Усиленная квалифицированная электронная подпись. Это самая защищенная разновидность ЭП; это ЭП, которая создается с применением криптографических средств, подтвержденных компетентными органами. Гарантом подлинности такой подписи служит специальный сертификат, который выдается аккредитованным удостоверяющим центром.

Применение КЭП значительно шире. Она применяется при участии в различных торгах в качестве поставщика и заказчика, с использованием нее может проводиться документооборот внутри компании и с партнерами. А также с помощью нее сдаются отчеты в различные госорганы.

Существуют специальные криптопрограммы для работы с ЭП:

1. КриптоПро CSP
2. ViPNet CSP
3. КриптоАРМ

Схема ЭП предусматривает следующие процессы:

1. генерация ключей ЭП и ключей проверки ЭП;
2. формирование ЭП;
3. проверка ЭП.

Механизм ЭП определяется реализацией двух важных процессов:

1. формирование ЭП;
2. проверка ЭП.

Применение ЭП позволяет осуществить следующие функции при передаче в системе подписанного ЭП сообщения:

1. осуществление контроля целостности передаваемого подписанного ЭП сообщения;
2. доказательная идентификация лица, подписавшего сообщение при помощи ЭП;
3. защита сообщения от возможной подделки.

Взлом ЭП на практике представляет собой взлом алгоритма шифрования. Для полной защиты своих данных недостаточным является защита выполнения алгоритма RSA и применение мер математической безопасности, иными словами, использование ключа достаточной длины, поскольку на практике наилучший результат имеют те атаки, которые производятся на незащищенные этапы управления ключами системы RSA.

Вывод. В результате научно-практического исследования были проанализированы преимущества и недостатки электронной цифровой подписи. Несмотря на имеющиеся недостатки ЭП, можно сделать вывод о подавляющем количестве преимуществ её использования.

ЛИТЕРАТУРА

1. Электронный ресурс «Подводные камни» простой электронной подписи». – Режим доступа: <https://habr.com/ru/post/313982>. Дата обращения: 25.08.2020.
2. Электронный ресурс «Применение электронной подписи». – Режим доступа: <http://elektronnayapodpis.ru/wiki/primenenie>. – Дата доступа: 25.08.2020.
3. Электронный ресурс «Виды электронной подписи (ЭЦП)». – Режим доступа: <https://ca.kontur.ru/articles/vidy-ehlektronnoj-podpisi-esp>. – Дата доступа: 25.08.2020.
4. Электронный ресурс «Неквалифицированная электронная подпись». – Режим доступа: <http://elektronnayapodpis.ru/wiki/nekvalifitsirovannaya>. – Дата доступа: 25.08.2020.
5. Электронный ресурс «Программы и приложения для ЭЦП». – Режим доступа: <https://esesp.ru/programmy-i-prilozheniya-dlya-esp>. – Дата доступа: 25.08.2020.

УДК 004.03

СИСТЕМА «УМНЫЙ ДОМ»

А.А. СКУКОВСКАЯ, А.Ф. АСТУКЕВИЧ
(Представлено: Т.С. СТРУК)

В статье рассматриваются общие концепции системы «Умный дом», ее структура и актуальность разработки данной системы.

Ключевые слова: информационные технологии, гаджеты.

На сегодняшний день концепция умного дома - это совокупность технологий, позволяющая создавать различные системы автоматизации жилого пространства, обеспечивающие возможность взаимодействия различных устройств, удаленного управления ими, а также, дружественный графический интерфейс для максимально простого «общения» с хозяином. Условно все устройства системы можно поделить на несколько типов: контроллер, устройства-исполнители, устройства управления и прочее интегрируемое оборудование и веб-сервисы. Все эти устройства общаются между собой посредством сигналов, каждое устройство умного дома одновременно и принимает, и передает сигнал.

Средства решения задачи:

Рассмотрим каждый тип устройств умного дома подробнее.

Контроллер – это «мозг» умного дома. Этот прибор контролирует работу сети и всех входящих в нее устройств, хранит в своей памяти сложные сценарии (определенные наборы действий) и обеспечивает связь системы умного дома с вашим смартфоном, планшетом или компьютером. Различные датчики, определяющие движения, задымления, протечки, – это «органы чувств» умного дома. Благодаря им система непрерывно получает информацию о том, что происходит в доме.

Главный критерий при выборе контроллера — протокол беспроводной связи, по которому он работает. Wi-Fi для этих целей, как правило, не подходит. Модуль слишком энергозатратен — приборы и датчики не смогут с ним подолгу работать автономно, их постоянно придется подзаряжать. Кроме того, сама технология довольно дорогая. Также у Wi-Fi могут быть проблемы с безопасностью и стабильностью покрытия.

Поэтому чаще всего встречаются контроллеры с протоколами Z-Wave и ZigBee. Принцип работы у них схож: в обоих ячеистая схема сети, когда каждое устройство умного дома становится своеобразным передатчиком. Это увеличивает зону покрытия и делает сеть стабильнее. Если какой-то элемент выходит из строя, его функции по передаче данных просто перераспределяются между остальными. Так как команды идут не по строго определенному маршруту, ниже риск, что они не дойдут до адресата [1].

Актуаторы (устройства-исполнители) – компоненты умного дома, которые по входящему сигналу (от сенсора или контроллера) и в соответствии с запрограммированным сценарием обеспечивают соответствующее действие. Примером простейшего актуатора может быть диммер, который по сигналу от датчика освещенности изменяет яркость накала лампы. Другие примеры актуаторов: всевозможные реле, блоки открывания/закрывания жалюзи и другие.

Сенсоры (датчики) – «глаза» и «уши» системы «Умный дом», определяющие изменения контролируемых параметров. Это могут быть датчики присутствия, времени, освещенности, температуры и другие. Сенсоры передают в инсталляционную шину сигнал об изменении значения параметров [2].

Чаще всего для реализации умного дома используют определенный набор датчиков. В этот набор входят датчики температуры и влажности воздуха, датчики освещения, датчики движения, датчики, обеспечивающие безопасность (датчики горючих газов и дыма, датчики пламени). Далее рассмотрим подробнее некоторые датчики.

К датчикам температуры и влажности относятся: AM2320, АНТ10, ВМЕ280, SHT31, HDC1080, Si7021. В таблице 1 приведены основные характеристики перечисленных датчиков.

Главным преимуществом датчика AM2320 является доступность цены и наличие корпуса. Датчик AM2320 представлен на рисунке 1.

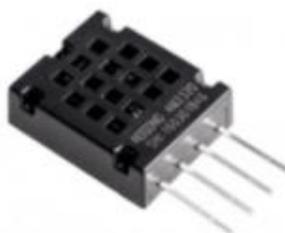


Рисунок 1. – Датчик AM2320

Таблица 1 – Датчики температуры и влажности

Характеристика/ Наименование дат- чика	AM2320	АНТ10	ВМЕ280	SHT31	HDC1080	Si7021
Диапазон темпера- тур	-40°C... +80°C	-40°C... +85°C	-40°C... +85°C	-40°C... +125°C	-40°C... +125°C	-40°C... +85°C
Погрешность температуры	±0,5°C	± 0,3 °C	± 0,5 °C	± 0,3 °C	± 0,2 °C	± 0,4 °C
Диапазон влажности	0 ... 99,9%	0...100%	0...100%	0...100%	0...100%	0...100%
Погрешность влажности	±3%	± 2%	±3%	± 2%	± 2%	±3%
Напряжение питания	3,1-5,5 В	1,8-6,0 В	1,8-5,0 В	2,5-5,5 В	2,7-5,5 В	1,9-3,6 В
Габариты	15x12,1x4,5 мм	16x11 мм	15x12x3 мм	40x20 мм	16x16 мм	3x3x1,5 мм
Корпус	есть	нет	нет	нет	нет	нет
Стоимость	10,5	6,65	78,4	46,2	53,9	14

К датчикам освещения относятся: ВН1750, МАХ44009, ОРТ3001. Характеристики перечисленных датчиков представлены в таблице 2.

Согласно документации, датчик ВН1750 чувствителен к видимому свету и практически не подвержен влиянию инфракрасного излучения, т.е. реагирует примерно на тот же спектральный диапазон, что и человеческий глаз. Данный датчик представлен на рисунке 2.

Таблица 2 – Датчики освещения

Параметр/датчик	ВН1750	МАХ44009	ОРТ3001
Диапазон освещенности	0-65535 лк.	0.045 – 188000лк	0.01 до 83886 лк
Напряжение питания	3.3-5 В	3,3 - 5 В	1.6...3.6 В
Размер	18.5 x 13.9 x 2 мм	20 x 10 мм	2×2×0.65 м
Стоимость	3,78	3,5	3,33

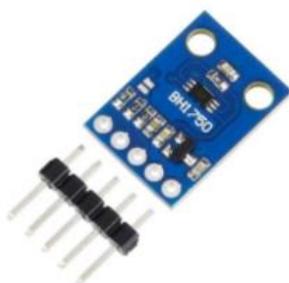


Рисунок 2. – Датчик BH1750

Как видим, организация умного дома – довольно трудоемкий, но занимательный процесс, но главное это то, что система умных домов на данный момент набирает все большую популярность в связи с удобством ее использования и легкостью обучения. Современные разработки в данном направлении позволяют установить систему умного дома как посредством специальных организаций, так и самостоятельно.

Заключение. В данной статье были рассмотрены основные компоненты системы «Умный дом», принципы их взаимодействия друг с другом и актуальность данной системы на данный момент. Было выявлено, что для установки данной системы есть все условия и компоненты.

ЛИТЕРАТУРА

1. Lifehacker [Электронный ресурс]. Режим доступа: <https://lifehacker.ru/smart-controllers/>. Дата доступа: 17.09.2020.
2. Система снабжения. [Электронный ресурс]. Режим доступа: <http://snabsystem.ru/catalog/umnyy-dom/moduli-i-komponenty-umnogo-doma>. Дата доступа: 17.09.2020.
3. 3DiY [Электронный ресурс]. Режим доступа: <https://3d-diy.ru/product/cifrovoj-datchik-temperatury-i-vlazhnosti-am2320>. Дата доступа: 20.09.2020.
4. Вольтик [Электронный ресурс]. Режим доступа: <https://voltiq.ru/bme280-and-arduino/>. Дата доступа: 20.09.2020.
5. Chipdip [Электронный ресурс]. Режим доступа: <https://www.chipdip.ru/>. Дата доступа: 20.09.2020.
6. Компэл [Электронный ресурс]. Режим доступа: <https://www.compel.ru/lib/71297/>. Дата доступа: 20.09.2020.

УДК 004.03

СЕРВЕРНАЯ ЧАСТЬ СИСТЕМЫ «УМНЫЙ ДОМ»**А.А. СКУКОВСКАЯ, А.Ф. АСТУКЕВИЧ**
(Представлено: Т.С. СТРУК)

В статье рассматриваются аппаратные и программные средства для организации серверной части системы умного дома.

Ключевые слова: информационные технологии, гаджеты, сервер, сеть.

Одним из самых главных компонентов умного дома является сервер, который связывает все остальные составляющие данной системы, собирает данные с других компонентов и передает команды остальным. Для создания серверной части умного дома используются различные аппаратные и программные средства, которые мы рассмотрим в данной статье.

Средства решения задачи:

Физическую часть сервера умного дома обычно выбирают по нескольким критериям: цена, мощность, надежность, размер.

Промышленные сервера в системах «Умный дом» обычно не используются, так как это достаточно дорого. В качестве сервера для умного дома можно использовать любой компьютер на Windows или Linux. Основное достоинство такого варианта — можно использовать любой компьютер в качестве «Умного дома». Системы для создания умного дома обычно не требовательны к железу, поэтому запустятся на многих старых компьютерах и будут стабильно функционировать. Но если покупать новый персональный компьютер специально в качестве сервера — дорого и кроме того, использование компьютера/нетбука в качестве сервера — это его габариты.

Самым лучшим вариантом для сервера умного дома является одноплатный компьютер (рис.1.).



Рисунок 1. – Одноплатный компьютер

К основным достоинствам использования одноплатных компьютеров в качестве сервера умного дома относятся:

- небольшая цена — от 20 USD в Китае до 50-70 USD в магазинах СНГ;
- по техническим характеристикам отлично подходит для умного дома;
- надежность;
- небольшой по размерам.

Самые популярные одноплатные компьютеры для создания Умного дома:

- Raspberry Pi;
- Orange Pi;
- Banana Pi;
- CubieBoard [1].

Для программной составляющей сервера умного дома уже существуют готовые программы, помогающие пользователям настроить свой «Умный дом». Одной из таких систем является MajorDoMo.

Система «Умный дом» Majordomo — проект по адаптации и автоматизации жилого помещения.

Системы Majordomo максимально адаптированы для начинающих пользователей и поэтому практически не требуют опыта или знаний в таких областях, как программирование или электроника.

Majordomo — это бесплатная общедоступная программа, которая поможет быстро обучиться комплексному управлению «Умным домом». Благодаря поддержке двух основных платформ (Windows или Linux) такая система может быть установлена фактически на любой персональный компьютер, поскольку потребляет относительно мало ресурсов. Даже имея малопроизводительный ПК, можно позволить себе пользоваться этой системой. [2]

Также для более продвинутых пользователей имеются системы с расширенными настройками и открытым кодом, такие как OpenHAB, Domoticz и Home assistant.

Home assistant - это ПО с открытым кодом для автоматизации умного дома, ориентирующееся на локальное управление и конфиденциальность. Он отлично подходит для работы на Raspberry Pi или локальном сервере. Проекту более пяти лет, он использует python и лицензию Apache 2. Для управления устройствами Home assistant использует отдельные модули (integrations, или components). Создать такой довольно просто. На их сайте можно найти каталог основных (одобренных и поддерживаемых сообществом) модулей. Среди общего их количества (1485 штук) попадаются совершенно разнообразие. [3]

Что касается openHAB, - это очень гибкий конструктор. Основанный на технологии OSGi, он позволяет конфигурировать каждый отдельный плагин (Binding) «налету», не перезагружая весь сервер.

OpenHAB реализует единую шину, т.е. позволяет объединить все устройства с разными протоколами в единую сеть, абстрагируя пользователя от каждого конкретного протокола. Таким образом, можно пользоваться единственным средством управления (скажем, приложением на смартфоне) и реализовать сколь угодно сложную логику взаимосвязи между устройствами.

OpenHAB — это специальный сервер, который может работать на любом компьютере под управлением любой ОС (включая RaspberryPi). Вся установка заключается в распаковывании дистрибутива сервера и установке Java машины. Далее начинается процесс настройки.

Третья система, Domoticz - это мультиплатформенное ПО с открытым кодом ориентированное на создание системы управления умным домом. Поддерживает большое количество различных устройств разных вендоров, в том числе работает с устройствами Xiaomi.

Система имеет более гибкие возможности по настройке сценариев — например проверку активности устройства, то чего нет в MiHome, или создание переменных — которые позволяют по одному условию — например нажатие клавиши — выполнять различные действия, в зависимости от значения переменной.

Также, если разработкой сервера системы «Умный дом» занимается программист, есть возможность запрограммировать сервер с нуля, используя лишь свой собственный код.

Данный подход является самым сложным, но и самым гибким, так как уже готовые программы всегда имеют ограничения в возможностях.

При выборе языков программирования на одноплатные компьютеры, такие как Raspberry Pi, обычно останавливаются на таких популярных языках как C, C++, JavaScript и Python. Однако, лучше всего для этого подходят именно C и Python. Так как Raspbian – ОС на базе Unix, в качестве базового языка здесь выступает C. С его помощью можно получить максимальную производительность без использования машинных команд. А IDLE, стандартная среда разработки программ на Raspberry, работает как раз на языке Python.

Исходя из анализа вышеуказанных фактов были сделаны выводы о том, какие аппаратные и программные средства лучше всего подходят для создания сервера умного дома. В качестве аппаратной части был выбран одноплатный компьютер Raspberry Pi, так как с помощью этого микрокомпьютера можно собрать множество интересных решений, и не только для умного дома. А встроенные сетевой интерфейс и WiFi модуль позволяют легко подключить плату к локальной сети или Интернету. Удобно, что ее наборы уже включают все необходимое, в том числе грамотное руководство, позволяющее легко познакомиться с основами работы с Raspberry Pi. [4] Также Raspberry Pi имеет доступную стоимость. В качестве языка программирования целесообразно будет использовать Python в связи с его отличной совместимостью с выбранной аппаратной частью.

Заключение. В данной статье были рассмотрены основные средства, которые используются для создания серверной части системы «Умный дом». Все полученные при исследовании данные были проанализированы и в итоге были выбраны оптимальные средства для создания сервера для умного дома.

ЛИТЕРАТУРА

1. MAJORDOMO [Электронный ресурс]. Режим доступа: <https://mjdm.ru/kakoy-server-vibrat-dlya-umnogo-doma/>. Дата доступа: 21.09.2020.
2. Умный дом. [Электронный ресурс]. Режим доступа: <https://proumnyjdom.ru/sistemy-avtomatizacii/sistema-umnij-dom-majordomo.html>. Дата доступа: 21.09.2020.
3. Хабр. [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/471822/>. Дата доступа: 21.09.2020.
4. AGEOFGEEKS [Электронный ресурс]. Режим доступа: <https://ageofgeeks.com/technology/raspberry-pi-smart-home/>. Дата доступа: 21.09.2020.

УДК 004.55; 004.514; 364.042

ГРАФИЧЕСКИЕ СИМВОЛЫ В СОЦИАЛИЗАЦИИ ЛЮДЕЙ
С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ

А.Ю. СОЛОЦКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)

Рассмотрены особенности отображения основных графических символов для обозначения важной информации для людей с ограниченными возможностями. Приведены примеры использования графической информации в повседневной жизни для ускорения восприятия на примере символов доступности для людей ограниченными возможностями. Показано, что художественно-визуальные образы современных знаков коммуникации способствуют упрощению жизни инвалидов обеспечивая их равенство среди людей.

Символика играет огромную роль в жизни общества. Будь то дорожная разметка, рекламный постер или открытка, увидев знакомый знак, мы сразу понимаем, о чем идет речь. Часто символ понятнее многих слов. И его предназначение – информировать о проблеме, заявлять о том явлении, которое он обозначает, служить сигналом к действию или призывом изменить что-либо.

Всем известна «Красная ленточка» – международный символ борьбы со СПИДом, «Земной шар в окружении стилизованных человечков» – символ Международного дня детей, символы опасности – яд, радиация, биологическая опасность – легко распознаваемые на подсознательном уровне каждым. И все они быстро концентрируют внимание на том, что есть особенности, с которыми нужно считаться.

Потребность применения визуальных знаков коммуникации в интерактивной среде для людей с ограниченными возможностями сегодня актуальна и для Республики Беларусь, где на сегодняшний день насчитывается более полумиллиона инвалидов (6,6% среди взрослого населения и 1,5% среди детей) [1], при этом, количество одних инвалидов-колясочников только в городе Новополоцке превышает 200 человек. Поэтому в рамках проекта по разработке электронного путеводителя по городу Новополоцку для людей с ограниченными возможностями [2] проработка, рассмотрение и использование символики доступности для людей с ограниченными способностями является важной задачей.

Далее подробнее остановимся на знаковых средствах, передающих информацию о степени доступности объектов окружающей среды – предупреждающие знаки, таблички и наклейки, пиктограммы и т.п., которые размещаются при входе в помещения, в здания и сооружения общественного назначения, на транспортных средствах и т.д. Это необходимо для того, чтобы использовать наиболее известные знаки для отображения необходимой информации для людей с ограниченными возможностями в своём программном продукте.

За многие годы повышенной заботы о людях с ограниченными возможностями разработано и широко используется большое число специальных символов. Примером может быть контрастный жёлтый круг, разработанный с учётом физиологических особенностей человеческого зрения и размещаемый на обширном остеклении дверных проёмов. Особенно важную роль он играет для слабовидящих людей.

Примеры некоторых наиболее часто используемых знаков представлены в таблице 1.

Таблица 1. – Основные знаки доступности

	Знак «Осторожно! Крутой подъем» размещается на стенах и других вертикальных поверхностях на путях постоянного передвижения инвалидов в креслах-колясках по дорогам и внутри доступных для инвалидов в креслах-колясках зданий и сооружений перед подъемами.
	Информационный знак «Лифт для инвалидов» размещается рядом с дверьми лифта, оборудованного специально для инвалидов.
	Знак «Туалет для инвалидов» используют для обозначения входа в санузел, который оборудован специально для людей с ограниченными возможностями.
	Знак «Кнопка вызова» предназначен для использования в поликлиниках, больницах и аптеках, магазинах и банках, гостиницах, кафе и других объектах социальной инфраструктуры и помогает людям с ограниченными возможностями без труда увидеть месторасположение звонка для помощи или обслуживания персоналом заведения.

Окончание таблицы 1

	Знак «Парковка для инвалидов» размещается на общественных автомобильных стоянках у мест, наиболее доступных для въезда и выезда транспорта. На них могут припарковаться только инвалиды I и II групп или перевозящие их.
	Знак «Место для инвалидов, пожилых людей с детьми» располагается во всех видах общественного транспорта у мест, доступ к которым требует минимальных усилий.
	Знак «Пандус» устанавливается у пандусов – конструкций, обеспечивающих подъём или спуск и дублирующих или заменяющих лестницы внутри и снаружи зданий, например, на цокольный этаж.

Все эти изображения с небольшими вариациями оформления общеизвестны, каждый встречает их в своей повседневной жизни с раннего возраста, они значительно упрощают навигацию людей с ограниченными возможностями, понятны даже детям, не умеющим читать. Кроме того, большая часть получаемой людьми информации поступает по зрительному каналу, и это позволяет точно и без усилий их идентифицировать.

Важность и необходимость таких символов подтверждает неоднократное проведение в Беларуси конкурсов на создание знака доступности. Первый конкурс проводился офисом по правам людей с инвалидностью, который занимается вопросами доступности в области прав людей с ограниченными возможностями в рамках «Недели доступности» [3]. Второй – в рамках проекта «Доступный город» при поддержке ОО «Республиканская ассоциация инвалидов-колясочников» [4]. Оба некоммерческих конкурса привлекли множество неравнодушных дизайнеров и обратили дополнительное внимание общества на проблему недостаточной оборудованности городов для всех людей с ограниченными возможностями передвижения. Работы-победители данных конкурсов представлены на рисунке 1.



Рисунок 1. – Разработанные знаки мест повышенной доступности в Республике Беларусь

В разработанном электронном путеводителе по городу Новополоцку для людей с ограниченными возможностями описанные в таблице знаки использовались для максимально понятного и содержательного указания параметров доступности объектов социальной инфраструктуры города. Это позволило тем числе сократить занимаемую этим описанием полезную площадь экрана, что особенно важно для пользователей мобильных устройств.

Таким образом, проведённые исследования основных графических символов для обозначения важной информации на примере визуальных образов современных знаков коммуникации для людей с ограниченными возможностями и опыта их применения в повседневной жизни с целью упрощения восприятия широкими массами можно сделать вывод, что использование такого рода изображений во всём мире глубже укореняет художественно-визуальные образы современных знаков коммуникации в сознании людей и способствует упрощению жизни инвалидов и тем самым, обеспечивает им возможность свободного

доступа к объектам социальной инфраструктуры в приоритетных сферах жизнедеятельности: жилым, общественным, производственным зданиям, сооружениям и помещениям, местам отдыха и досуга, тем самым обеспечивает равенство возможностей.

ЛИТЕРАТУРА

1. Численность инвалидов, получающих пенсию в органах по труду и социальной защите | Министерство труда и социальной защиты Республики Беларусь. Официальный сайт [Электронный ресурс]. – Режим доступа: <http://www.mintrud.gov.by/ru/chislinv>. – Дата доступа: 06.09.2020.
2. Солоцкий, А.Ю. Электронный путеводитель по городу Новополоцку для людей с ограниченными возможностями / А. Ю. Солоцкий. // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30(100): «Промышленность». – С. 13-14.
3. Офис по правам людей с инвалидностью: Подведены итоги конкурса «Создание символа Доступность» | Офис по правам людей с инвалидностью [Электронный ресурс]. – Режим доступа: <http://www.disright.org/ru/news/ofis-po-pravam-lyudey-s-invalidnostyu-podvedeny-itogi-konkursa-sozdanie-simvola-dostupnost>. – Дата доступа: 11.09.2020.
4. «Доступный город» подводит результаты конкурса на создание знака доступности | ОО «РАИК» [Электронный ресурс]. – Режим доступа: <https://raik.by/novosti/dostupnyj-gorod-podvodit-rezultaty-konkursa-na-sozdanie-znaka-dostupnosti/>. – Дата доступа: 12.09.2020.

УДК 004.55; 004.62; 364.042

РАЗРАБОТКА АДМИНИСТРАТИВНЫХ ИНТЕРФЕЙСОВ

А.Ю. СОЛОЦКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)

В статье рассматриваются особенности разработки административного интерфейса веб-ресурса на примере путевода по городу Новополюцку для людей с ограниченными возможностями. Основная цель рассмотреть некоторые принципы разработки таких интерфейсов в современных реалиях и серверной части для них.

Развитие технологий привнесло множество изменений в обычный образ жизни человека. Огромное число сервисов перемещаются полностью или предоставляют варианты доступа через интернет, на веб-сайтах. Соответственно, обеспечение возможности поддержания данных в актуальном состоянии очень важная задача. Такие интерфейсы (их называют административные) должны быть защищены, оказывать максимальную помощь пользователю и быть построены на современных технологиях для дальнейшей поддержки. Административные интерфейсы обретают всё большую важность с ростом количества сервисов, доступных в сети Интернет, а продуманная их проработка является основой стабильной работы, поэтому рассмотрение основных принципов и подходов при разработке административных интерфейсов является актуальной задачей.

Далее подробно остановимся на рассмотрении основных этапов создания административного интерфейса на примере реализации путевода по городу Новополюцку для людей с ограниченными возможностями.

Разработка интерфейса для управления информацией администратором проводилась на первом этапе реализации путевода, после проектирования подсистемы хранения информации об объектах города. На втором этапе создан пользовательский интерфейс для отображения актуальной информации об объектах городской инфраструктуры конечным пользователям в удобном виде с возможностью информирования конечных пользователей [1]. Для построения системы использовалась клиент-серверная модель. Для размещения базы данных (БД) используется реляционная система управления базами данных MySQL, а для получения данных из постоянного хранилища использовалась другая открытая технология – Hypertext Preprocessor (PHP). Это скриптовый язык общего назначения, в нашем случае он будет основой серверной части приложения и отвечать за генерацию промежуточного представления информации из базы данных и безопасного и защищённого внесения изменений туда с использованием средств извлечения параметров запросов автоматически. Также он будет в некоторой степени ответственен за безопасность, блокируя соединения извне, позволяя редактировать данные только локально. Это наиболее действенный из прямолинейных подходов, не требующий от администратора дополнительных действий. Для доступа к ресурсу в сети Интернет необходимо ещё одно программное обеспечение – веб-сервер. Он принимает HTTP-запросы от клиентского браузера и предоставляет им ответы с веб-страницами, изображениями и другими данными. Он также не будет предоставлять доступа к БД снаружи.

Диаграмма вариантов использования разработанной системы представлена на рисунке 1.

Как видно из диаграммы, администратору предоставляются возможность полного управления маркерами: созданием, удалением и изменением всех его параметров. Для этого и был разработан специальный интерфейс.

Слева располагается карта. Были использованы карты Яндекс как качественно поддерживающие наш регион, а также предоставляющие бесплатное использование для некоммерческих проектов [2]. Карта используется для позиционирования маркеров, при этом записывается в базу положение центра карты, отмеченное крестом. Так как это стандартная карта, доступно перемещение и масштабирование.

Далее расположен блок описания маркера. Здесь необходимо указать название объекта, его адрес, телефон (в определённом формате), выбрать из списка тип, указать график работы и отметить соответствующие параметры доступности нажатиями на необходимые изображения. График работы настраивается выбором дня недели, указанием времени работы, а также можно выставить обед и отметить его время. При этом, при деактивации дня также происходит очистка времени и обеда. Ввод в поля настроек возможен только во время конфигурации маркера, а иначе они заблокированы. Снизу доступны кнопки подтверждения введённых данных и отмены. При пропущенных или не верно заполненных полях выводится ошибка у первого некорректного поля, его фон становится оранжевым и сохранения информации не происходит.

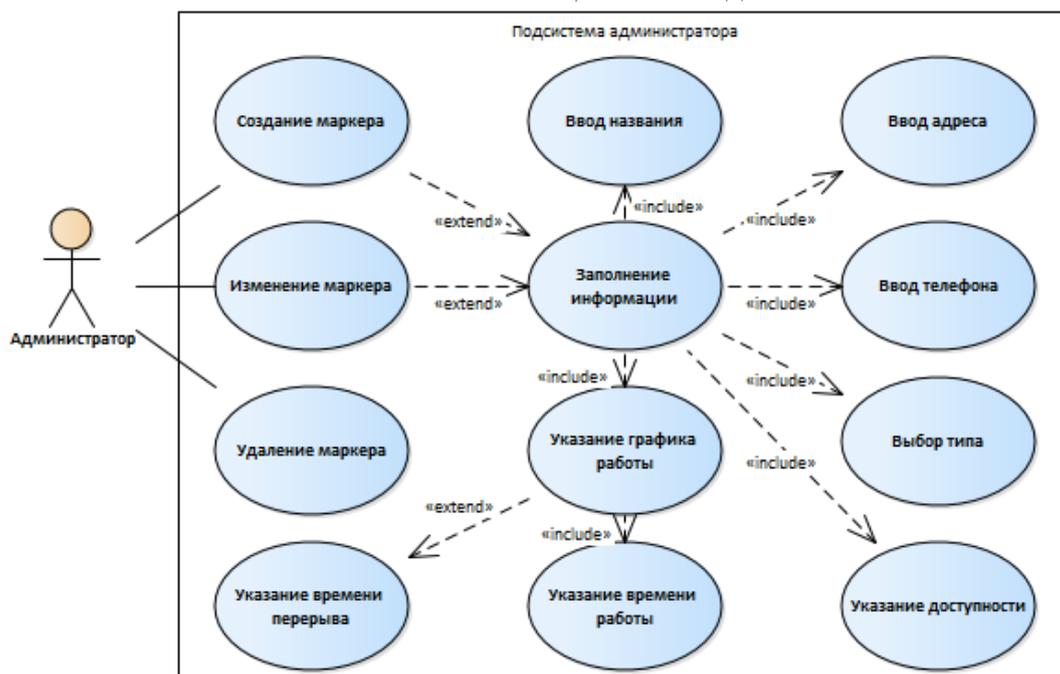


Рисунок 1. – Структура административной системы

Крайним справа является блок с данными в процессе настройки и из базы. Сверху расположен список настроенных маркеров в базе. Их можно изменить или удалить, при этом, как повергнувшиеся правкам они оказываются во втором списке. Здесь их так же можно повторно изменить или отменить редактирование. Если в результате правок маркер становится идентичным изначальной версии, считается, что он не изменён и отметка возвращается в первоначальный список. В случае, если запись помечается как удалённая, в области редактирования рядом с названием отображается соответствующая пометка. Ниже расположен список добавленных маркеров с кнопками создания нового, его изменения и отказа от его добавления. Последним является список данных, помеченных в базе как удалённых. Маркер из этого списка можно восстановить по нажатию на кнопку под списком. По кнопке выбора файла загружается информация из json-файла с промежуточным представлением данных. При нажатии на кнопку отправить, в зависимости от версии интерфейса данные сохраняются в файл, который автоматически скачивается, или отправляются на сервер, где PHP сохраняет их в базу и инициирует обновление файла промежуточного представления.

При разработке интерфейса использовались адаптивные подходы к позиционированию блоков, в том числе с учётом соотношения сторон окна, и определению их ширины и масштаба относительно базового размера шрифта [3]. Интерфейс разработанной системы представлен на рисунке 2.

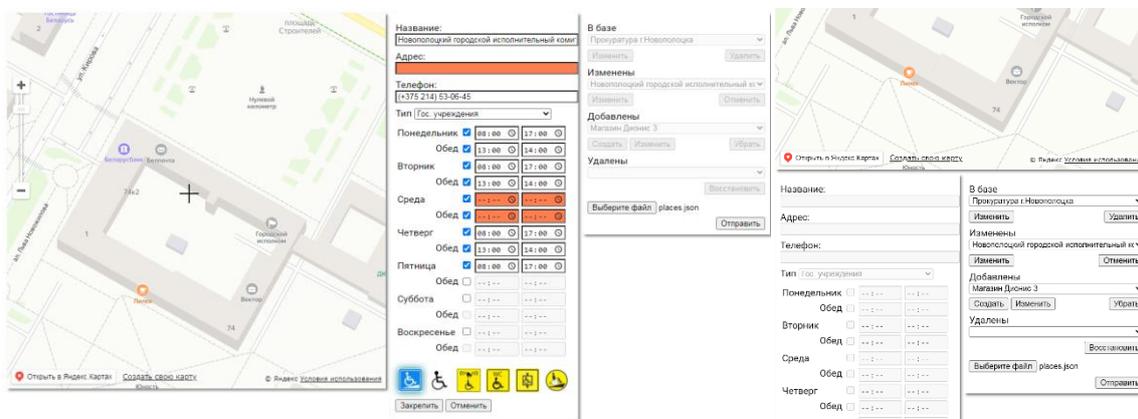


Рисунок 2. – Административный интерфейс

Следует также отметить, что в разработке приняли активное участие и работники государственного учреждения «Новополоцкий территориальный центр социального обслуживания населения», которые подготовили объём данных об объектах города представив их в виде путеводителя, доступного в бумажном и электронном вариантах. Это позволило получить достоверные данные для маркеров и выделить критерии доступности объектов социальной инфраструктуры города. Они же и являются пользователями административного интерфейса.

Таким образом, разработанное программное обеспечение отвечает всем заявленным требованиям предметной области и помогает поддерживать коммуникации людей с ограниченными возможностями, а рассмотренные принципы и подходы разработки административных интерфейсов позволили учесть все функциональные возможности системы и вынести на первый план требуемые функции, обеспечив наглядность и неперегруженность интерфейса путеводителя по городу Новополоцку для людей с ограниченными возможностями.

ЛИТЕРАТУРА

1. Солоцкий, А.Ю. Электронный путеводитель по городу Новополоцку для людей с ограниченными возможностями / А. Ю. Солоцкий. // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30(100): «Промышленность». – С. 13-14.
2. Условия использования API Яндекс.Карт. Статьи [Электронный ресурс]. – Режим доступа: https://yandex.ru/dev/maps/jsapi/doc/2.1/terms/index.html/#index_conditions. – Дата доступа: 11.09.2020.
3. Солоцкий, А.Ю. Разработка пользовательского интерфейса веб-сайта, с учетом требований отзывчивости, адаптивности и повышения доступности для людей с ограниченными возможностями / А. Ю. Солоцкий. // Труды молодых специалистов Полоцкого государственного университета. – 2019. – № 30(100): «Промышленность». – С. 15-16.

УДК 004.055

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА WEB-САЙТА В СЕТИ ИНТЕРНЕТ НА ПРИМЕРЕ ГОРОДА БЕШЕНКОВИЧИ

А.И. СТРЕЛЬЧЕНКО

(Представлено: канд. техн. наук., доц. **О.Н. ПЕТРОВИЧ**)

В данной статье рассматривается проектирование и разработка веб-сайта на примере г.Бешенковичи, особенности и специфика сайта.

Введение. Интернет технологии развиваются стремительно, и в настоящее время большинство компаний уже имеют собственные сайты. Тем не менее, не только компании, но и частные лица, у которых есть те или иные услуги, а также собственные сайты появляются и у городов.

Основной раздел. На современном этапе развития человечества Интернет выступает в роли мощного инструмента по поиску и представлению информации. По статистике, более половины жителей планеты имеют доступ к сети Интернет. Как следствие, разработка Web-сайта в сети Интернет позволит использовать современные технологии для развития информационной поддержки и рекламы сайта.

Для создания сайта города потребуются такие технологии как:

- CMS (Content Management System) – это система управления контентом, набор скриптов для создания, редактирования и управления контентом сайта. Примерами CMS являются WordPress, Joomla, OpenCart;
- HTML (Hypertext Markup Language) – язык разметки для создания шаблонов страниц сайтов;
- CSS (Cascading Style Sheets) – инструмент для оформления внешнего вида страниц;
- PHP (Hypertext Preprocessor) – язык программирования, используемый для разработки программных функций сайта. На этом языке написано большинство современных CMS (Content Management System);
- Хостинг – от правильного выбора хостинга и понимания нюансов его работы будет зависеть работоспособность, быстродействие и надежность работы сайта.

Также к ним понадобятся основы копирайтинга, создание контекстной рекламы, запуска таргетинга в социальных сетях и оптимизация текстов под требования поисковиков SEO (Search Engine Optimization). Покупка домена и хостинга. Домен – зарегистрированный адрес в интернете, по которому городской сайт доступен пользователям. Хостинг – услуга по предоставлению места на сервере для хранения файлов сайта.

На городском сайте должны присутствовать такие информационные разделы как:

- городские новости;
- раздел об истории города;
- справочная информация (телефоны, адреса городских служб и компаний, расписание транспорта и т.п.);
- карта города;
- погода, курсы валют банков;
- фотогалерея;
- туристический потенциал;
- городская афиша;
- разделы авто, работа, недвижимость;
- статьи и обзоры на различную тематику.

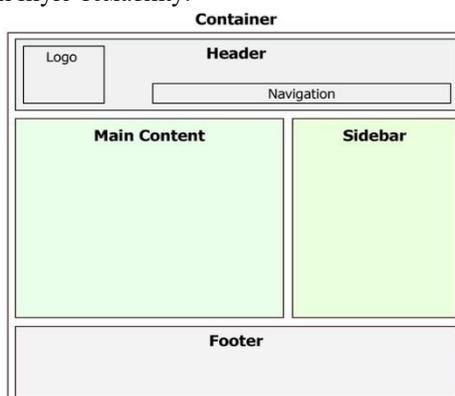


Рисунок 1. – Макет главной страницы

Раздел «Logo» представляет собой изображение-герб города.

Header – название города и дата основания.

Navigation – навигация по страницам, важная часть веб-страницы; она нужна посетителям, чтобы пользоваться сайтом. Навигацию должно быть легко найти и использовать, поэтому она почти всегда размещается в заголовке или, по крайней мере, в верхней части страницы. Иногда оба типа навигации используются одновременно на сайтах с высоким содержанием контента.

Типы навигации:

– по горизонтали: ряд ссылок, отображаемых в строке, которую обычно называют «навигацией»;

– по вертикали: ряд ссылок, отображаемых в вертикальном порядке, обычно называемый «меню».

Здесь находятся ряд ссылок (разделов сайта): городские новости, раздел об истории города, карта города и т.д.

Main Content – основной контент, который чаще всего ищут посетители сайта. Контент должен быть в центре внимания страницы, чтобы посетители могли быстро найти то, что они ищут.

Sidebar – боковая панель, раздел в котором находится вторичный контент, такой как реклама, поиск по сайту, ссылки для подписки (RSS, Twitter, электронная почта и т.д.), этот раздел не всегда необходим, хотя многие веб-сайты его используют. Обычно боковая панель расположена в правом ряду, но боковую панель можно также разместить слева или даже с двух сторон, если боковая панель не мешает внешнему виду основного контента.

Footer – в конце веб-сайта должен быть «подвал». Как и Header, Footer – это не основная часть веб-страницы, а скорее раздел содержащий информацию. В «подвале» будут указаны авторские права, юридические и основные контакты. Также можно разместить несколько ссылок на важные части веб-сайта, такие как верхняя страница, главная страница, страница контактов и др. Другие веб-сайты используют этот раздел как возможность упомянуть другие связанные материалы или другую важную информацию.

Заключение. В данной статье приводятся технологии, которые позволяют создать веб-сайт. Перечислены основные разделы, которые должны присутствовать на сайте города, и описаны составные части сайта (Logo, Header, Navigation, Sidebar, Footer).

ЛИТЕРАТУРА

1. Tilda.com [Электронный ресурс]. Режим доступа: <http://tilda.education/#website/>. – Дата доступа: 27.09.2020.
2. BESHANKOVICHY.BY [Электронный ресурс]. Режим доступа: <http://beshankovichy.by/>. – Дата доступа: 27.09.2020.
3. Habr. Сайт интересных статей по программированию. [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/53498/>. –Дата доступа: 27.09.2020.
4. Руководство по HTML5 и CSS3 [Электронный ресурс]. – Режим доступа: <https://metanit.com/web/html5/>. –Дата доступа: 27.09.2020.

ПОИСКОВАЯ ОПТИМИЗАЦИЯ WEB-САЙТА SEO (SEARCH ENGINE OPTIMIZATION)

А.И. СТРЕЛЬЧЕНКО

(Представлено: канд. техн. наук., доц. О.Н. ПЕТРОВИЧ)

В данной статье представлены основные способы SEO-оптимизации и раскрутки сайта, которые могут поднять позицию сайта в результатах выдачи поисковых систем.

Введение. Поисковая оптимизация (Search engine optimization) – это набор мер для поднятия позиций сайта в результатах выдачи поисковых систем для конкретных запросов пользователей. Как правило, чем выше позиция сайта в результатах поиска, тем больше заинтересованных посетителей переходит на него с поисковых систем.

Основной раздел. Сама суть SEO-оптимизации заключается в определении ключевых слов, часто используемых при поиске информации по теме, которой посвящен сайт, и в дальнейшем использовании этих слов в подходящих местах на сайте с целью увеличения вероятности того, что поисковые системы включат ссылку на него в свои выдачи.

Поисковая SEO-оптимизация сайта начинается еще в самом начале его создания. Разработка системы управления, дизайна и пользовательского интерфейса, равно как и написание текстового контента, должны подчиняться строгим правилам SEO, только тогда сайт считается оптимизированным для поисковых систем и конкурентоспособным на рынке.

Существуют несколько методов оптимизации сайта:

1. Белый метод оптимизация включает в себя:

- разработка семантического ядра (подбор определенных слов с помощью которых будет продвигаться сайт);
- разработка структуры сайта и работа с юзабилити;
- оптимизация html (hypertext markup language) кода сайта (мета-теги);
- создание новых веб-страниц для смыслового покрытия тематики сайта;
- оптимизация текста сайта.

Белая оптимизация требует больше времени, чтобы можно было увидеть результат продвижения сайта. Зато перечисленные выше процедуры создают базу для качественного продвижения сайта в дальнейшем.

2. Серый метод оптимизации включает в себя:

- добавление большого количества ключевых слов в тексте страницы, обычно в ущерб читабельности для человека. При этом, SEO-оптимизация заключается в подборе ключевых запросов для конкретной веб-страницы. Широко применяется также включение ключевого запроса в HTML-теги title, h1, атрибут meta keywords.

3. Черный метод оптимизации включает в себя:

- текстовый «спамдексинг» (насыщение веб-страницы ключевыми словами), использование невидимого текста, забивка текста под картинкой на сайте, пропись текста очень мелким шрифтом;
- свопинг – оптимизация веб-страницы сайте для достижения высокой позиции в поисковых системах, а после достижения этих позиций – вносятся значительные изменения страницы вплоть до полной замены тематики страницы;
- клоакинг – метод сокрытия поискового спама, с помощью которого пользователю сети выдается в браузере одна информация, а поисковым системам совсем другая;
- дорвей – специальные веб-страницы, при попадании на которые пользователь принудительно перенаправляется с помощью автоматического перенаправления или ссылок на целевую страницу. При этом сам «дорвей» не содержит полезной информации для пользователя;
- редирект – принудительный, не запрошенный пользователем переход на совсем другой адрес или загрузка новой вкладки. То есть пользователь попадает на совсем другую страницу.

Google, Yandex, Mail, Yahoo и другие поисковые машины предупреждают, что когда подобные методы обнаруживаются, на сайты налагаются санкции вплоть до блокировки в поисковых системах.

Поисковые машины научились изучать почти все черные методы продвижения сайтов.

Если раньше методы «редирект» и «дорвей» были весьма успешным способом оптимизации, то в настоящий момент такие сайты быстро удаляются из результатов выдачи в поисковых системах.

Для поднятия позиции сайта в результатах выдачи поисковых систем можно использовать специальные теги, например, мета-тег description который предоставляет поисковой системе короткое резюме, чему именно посвящена данная страница, сам тег располагается внутри тега <head>.

Существует еще один тег – keywords, он подсказывает поисковым системам по каким именно поисковым запросам должен находиться сайт.

Листинг 1. – Использование мета-тегов description и keywords

```
<html>
  <head>
    <meta name="description" content="Описание
сайта">
      <meta name="keywords" content="ключевое слов 1,
ключевое слово 2, ключевое слово 3" />
    </head>
  </html>
```

Заключение. В данной статье были рассмотрены основные способы оптимизации и раскрутки сайта. Методов продвижения довольно много, все они имеют различный эффект, и применяя только лишь один из них, результата не получится добиться. Также следует с осторожностью относиться к «черным» методам раскрутки, неумелое их применение может пагубно сказаться на продвижение сайта, и сайт может и вовсе потерять доверие поисковых систем.

ЛИТЕРАТУРА

1. Search engine optimization Wikipedia [Электронный ресурс]. Режим доступа: https://en.wikipedia.org/wiki/Search_engine_optimization– Дата доступа: 28.09.2020.
2. Habr. Сайт интересных статей по программированию. [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/480100/>. – Дата доступа: 28.09.2020.
3. SEO оптимизация или поисковая оптимизация сайта [Электронный ресурс]. Режим доступа: <https://convertmonster.ru/blog/seo-blog/seo-optimizacija-ili-poiskovaja-optimizacija-sajta/>– Дата доступа: 28.09.2020.
4. Микроразметка: что нужно знать SEO-специалисту о словарях и синтаксисе. [Электронный ресурс]. Режим доступа: <https://netpeak.net/ru/blog/mikrorazmetka-cto-nuzhno-znat-seo-spetsialistu-oslovaryakh-i-sintaksise/>. – Дата доступа: 28.09.2020.
5. Что такое SEO [Электронный ресурс]. Режим доступа: <https://netpeak.net/ru/blog/cto-takoye-seo>.– Дата доступа: 28.09.2020.
6. Основы SEO-оптимизации. [Электронный ресурс]. Режим доступа: <https://fireseo.ru/blog/seo-cto-eto-takoe>.– Дата доступа: 28.09.2020.

ENUM И SWITCH, И ПРОБЛЕМЫ ИХ ИСПОЛЬЗОВАНИЯ

Д.В. СУЩЕВСКИЙ

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)

Представлен способ решения проблемы использования напрямую типов перечислений в операторе `switch` для разработчиков, использующих C-подобные языки программирования.

Очень часто разработчикам использующих C-подобные языки, приходится сталкиваться с проблемой не обработанных CASE-условий в `switch`. Такие ошибки могут нести огромные убытки для компаний и надёжности самого программного обеспечения (ПО). С появлением функциональных языков программирования люди начали понимать, что на многие вопросы стоит посмотреть под другим углом, чтобы увидеть проблему или преимущества.

Для решения проблем использования типов перечислений в операторе `switch`, в функциональных парадигмах используются размеченные объединения (“discrimination union”) и код просто не будет компилироваться, пока не будут рассмотрены все возможные варианты из типа, в то время как, например, C# вынуждает разработчика использовать условие `DEFAULT` чтобы точно обработать все условия, так как `enum` не строго типизирован и можно любое число преобразовать в `enum`, что приносит много проблем в дальнейшей перспективе.

С такими проблемами можно столкнуться. Где возникает такая необходимость? Чаще всего это работа с каким-либо `web Application Programming Interface (API)`. В ситуации, когда к нам приходит сущность с фиксированным количеством полей, а уже понимать какие поля заполнены или актуальны для данной сущности, можно лишь по какому-то типу, который находится в объекте. Тогда код будет обрабатываться оператором `switch` и соответствующей логикой обработки актуальных типов на данный момент. Всё может усугубить то, что не весь код обработки таких сущностей находится в одном проекте и по банальной человеческой невнимательности можно забыть доработать код, в других проектах, что повлечет ошибочные ситуации и трудно обнаруживаемые ошибки.

В данной статье хотелось бы поделиться возможными решениями данной проблемы. Ниже рассмотрим реальные примеры на языке программирования C#. Если проблема состоит только в `switch` операторе и отслеживании новых типов, тогда необходимо избавиться от них. Идея состоит в том, чтобы заменить использование `switch` паттерном `visitor`.

Пример 1. Предположим, что у нас есть какой-то `API (Application Programming Interface)` для работы с документами, от которого мы получаем необходимые данные и определяем его тип, а далее в зависимости от этого типа, необходимо делать различные операции (см. рисунок 1).

```
public enum DocumentType
{
    Invoice,
    PrepaymentAccount
}

public interface IDocumentVisitor<out T>
{
    T VisitInvoice();
    T VisitPrepaymentAccount();
}

public static class DocumentTypeExt
{
    public static T Accept<T>(this DocumentType self, IDocumentVisitor<T> visitor)
    {
        switch (self)
        {
            case DocumentType.Invoice:
                return visitor.VisitInvoice();
            case DocumentType.PrepaymentAccount:
                return visitor.VisitPrepaymentAccount();
            default:
                throw new ArgumentOutOfRangeException(nameof(self), self, null);
        }
    }
}
```

Рисунок 1. – Пример файла `DocumentType.cs`

Предложено определять все связанные типы в одном файле, как показано на рисунке 2, что не является идиоматичным для .Net разработчика, однако иногда это очень упрощает понимание кода.

```
public class DatabaseSearchVisitor : IDocumentVisitor<IDocument>
{
    private ApiId _id;
    private Database _db;

    public DatabaseSearchVisitor(ApiId id, Database db)
    {
        _id = id;
        _db = db;
    }

    public IDocument VisitInvoice() => _db.SearchInvoice(_id);
    public IDocument VisitPrepaymentAccount() => _db.SearchPrepaymentAccount(_id);
}
```

Рисунок 2. – Visitor, который производит поиск документа в базе

Пример использования DatabaseSearchVisitor показан на рисунке 3.

```
public void UpdateStatus(ApiDoc doc)
{
    var searchVisitor = new DatabaseSearchVisitor(doc.Id, _db);

    var databaseDocument = doc.Type.Accept(searchVisitor);

    databaseDocument.Status = doc.Status;

    _db.SaveChanges();
}
```

Рисунок 3. – Пример использования DatabaseSearchVisitor.cs

Пример 2. Предположим, что у нас имеется список событий по электронному кошельку. Их определяем из очереди в виде типа события и json с дополнительными полями. Определение этих типов показано на рисунке 4.

```
public enum PurseEventType
{
    Increase,
    Decrease,
    Block,
    Unlock
}

public sealed class PurseEvent
{
    public PurseEventType Type { get; }
    public string Json { get; }

    public PurseEvent(PurseEventType type, string json)
    {
        Type = type;
        Json = json;
    }
}
```

Рисунок 4. – Описание события по счету

Например, появилась необходимость отправлять уведомления пользователю на определенный тип событий. Тогда реализуем visitor следующим образом, показанным на рисунке 5.

```
public interface IPurseEventTypeVisitor<out T>
{
    T VisitIncrease();
    T VisitDecrease();
    T VisitBlock();
    T VisitUnlock();
}

public sealed class PurseEventTypeNotificationVisitor : IPurseEventTypeVisitor<Missing>
{
    private readonly INotificationManager _notificationManager;
    private readonly PurseEventParser _eventParser;
    private readonly PurseEvent _event;

    public PurseEventTypeNotificationVisitor(PurseEvent @event, PurseEventParser eventParser, I
NotificationManager notificationManager)
    {
        _notificationManager = notificationManager;
        _event = @event;
        _eventParser = eventParser;
    }

    public Missing VisitIncrease() => Missing.Value;

    public Missing VisitDecrease() => Missing.Value;

    public Missing VisitBlock()
    {
        var blockEvent = _eventParser.ParseBlock(_event);
        _notificationManager.NotifyBlockPurseEvent(blockEvent);
        return Missing.Value;
    }

    public Missing VisitUnlock()
    {
        var blockEvent = _eventParser.ParseUnlock(_event);
        _notificationManager.NotifyUnlockPurseEvent(blockEvent);
        return Missing.Value;
    }
}
```

Рисунок 5. – Visitor по отправке уведомлений

Для примера, ничего не будем возвращать после отправки уведомления. Для этого используется тип Missing из System.Reflection или же это можно написать, используя тип Unit. В реальном проекте возвращался бы Result, например, с информацией об ошибке, если такие имеются (см. рис. 6).

```
public void SendNotification(PurseEvent @event)
{
    var notificationVisitor = new PurseEventTypeNotificationVisitor(@event, _eventParser, _noti
ficationManager);
    @event.Type.Accept(notificationVisitor);
}
```

Рисунок 6. – Пример использования visitor по отправке уведомлений

Если нужно использовать такой подход там, где важна производительность, в качестве visitor можно использовать структуры (ключевое слово struct). Тогда код изменится следующим образом, как показано на рисунке 7.

```
public static T Accept<TVisitor, T>(this DocumentType self, in TVisitor visitor)
    where TVisitor : IDocumentVisitor<T>
{
    switch (self)
    {
        case DocumentType.Invoice:
            return visitor.VisitInvoice();
        case DocumentType.PrepaymentAccount:
            return visitor.VisitPrepaymentAccount();
        default:
            throw new ArgumentOutOfRangeException(nameof(self), self, null);
    }
}
```

Рисунок 7. – Изменение в методе расширении

Сам `visitor` остаётся прежним, достаточно изменить `class` на `struct`. Сам код обновления документа будет выглядеть не так удобно (см. рисунок 8), но работает быстро.

```
public void UpdateStatus(ApiDoc doc)
{
    var searchVisitor = new DatabaseSearchVisitor(doc.Id, _db);

    var databaseDocument = doc.Type.Accept<DatabaseSearchVisitor, IDocument>(searchVisitor);

    databaseDocument.Status = doc.Status;

    _db.SaveChanges();
}
```

Рисунок 8. – Изменение в методе обновления документа

При таком использовании `generic`, необходимо уточнять типы самому, так как компилятор не способен вывести их автоматически.

Если нужно реализовать логику только в одном месте, то часто `visitor` – громоздко и не удобно. Поэтому есть альтернативное решение `match` (см. рисунок 9).

```
public static T Match<T>(this DocumentType self, Func<T> invoiceCase, Func<T> prepaymentAccount
Case)
{
    var visitor = new FuncVisitor<T>(invoiceCase, prepaymentCase);
    return self.Accept<FuncVisitor<T>, T>(visitor);
}
```

Рисунок 9. – Пример метода `match` для структуры

На рисунке 10 представлена реализация `visitor`, который принимает реализацию логики обработки типа в виде делегата.

```
public readonly struct FuncVisitor<T> : IDocumentVisitor<T>
{
    private readonly Func<T> _invoiceCase;
    private readonly Func<T> _prepaymentAccountCase;

    public FuncVisitor(Func<T> invoiceCase, Func<T> prepaymentAccountCase)
    {
        _invoiceCase = invoiceCase;
        _prepaymentAccountCase = prepaymentAccountCase;
    }

    public T VisitInvoice() => _invoiceCase();
    public T VisitPrepaymentAccount() => _prepaymentAccountCase();
}
```

Рисунок 10. – Реализация FuncVisitor

Пример использования метода расширения match (см рисунок 11).

```
public void UpdateStatus(ApiDoc doc)
{
    var databaseDocument = doc.Type.Match(
        () => _db.SearchInvoice(doc.Id),
        () => _db.SearchPrepaymentAccount(doc.Id)
    );

    databaseDocument.Status = doc.Status;

    _db.SaveChanges();
}
```

Рисунок 11. – Пример использования Match

Таким образом. При добавлении нового значения в enum необходимо.

- добавить метод в интерфейс;
- добавить его использование в метод расширение.

Для всех реализаций представленного интерфейса необходимо реализовать логику обработки нового типа и пока это не будет сделано, проект просто не соберется. Таким образом можно избавиться от проблемы забытого case в switch.

ЛИТЕРАТУРА

1. Ploeh blog. [Электронный ресурс] / Ploeh blog. – Режим доступа: <https://blog.ploeh.dk/2018/06/25/visitor-as-a-sum-type/> – Дата доступа: 16.09.2020.
2. Wikipedia. [Электронный ресурс] / Wiki. – Режим доступа: https://en.wikipedia.org/wiki/Unit_type – Дата доступа: 10.09.2020.
3. Microsoft documentation. [Электронный ресурс] / Result – F#. – Режим доступа: <https://docs.microsoft.com/en-us/dotnet/fsharp/language-reference/results> – Дата доступа: 13.09.2020.

УДК 004.05

ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ MICROSOFT TEAMS**Д.В. СУЩЕВСКИЙ***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)**Рассмотрены преимущества платформы Microsoft Teams для общения и совместной работы.*

В современном мире, невозможно представить себе жизнь без компьютера, интернета и возможности связаться с друзьями, родными из любой точки мира. Благодаря интернету, люди не только развлекаются, но и получают необходимые знания, участвуют в онлайн конференциях, проходят онлайн обучение и прочее. Сегодня вопрос о коммуникации и взаимодействии удалённо является несомненно актуальным. Наиболее острым он стал в условиях пандемией. Чтобы замедлить распространение коронавируса многие компании вынуждены были переводить своих работников на удалённую работу на период эпидемии. Всё больше лидеров бизнеса поддерживают идею работы на дому и формирования гибких графиков. По мере того, как офисы постепенно открываются после карантина, все больше работодателей рассматривают новые схемы занятости. Эксперты по работе с персоналом приветствуют «удалёнку» и сокращённую неделю как альтернативу массовому возвращению людей в офисы. Сдвиг в сторону дистанционной работы также позволит компаниям оценить, нужно ли им столько дорогой офисной площади. Всё больше лидеров бизнеса поддерживают идею работы на дому и гибких графиков. Этому способствуют новые технологии, такие как: Microsoft Teams, Zoom, Google Meet, Skype и пр., позволяющие проводить виртуальные веб-конференции, вебинары и видеоконференции, а также выступать в роли мессенджеров.

Для комфортной работы удалённо, необходима возможность простого общения онлайн, использование текстовых сообщений и видеозвонки, делиться необходимой информацией, устраивать групповые встречи. Некоторые люди для этого используют совершенно разные сайты, сервисы и программы, что вносит большой разлад в рабочий процесс и делает его неэффективным. Хорошей альтернативой такому подходу является Microsoft Teams.

Microsoft Teams – это центр командной работы, который позволяет объединить все, что необходимо для группы: чат и беседы с несколькими ветками, виртуальные собрания и видеоконференции, звонки и совместную работу над контентом с помощью приложений Microsoft 365, а также возможность создавать и интегрировать приложения и рабочие процессы, лежащие в основе вашего бизнеса. Разработана компанией Microsoft как конкурент популярного корпоративного решения Slack.

Основные особенности сервиса:

- возможность бесплатно использовать Microsoft Teams в командах, насчитывающих до 300 человек;
- неограниченное количество сообщений в чате и поиск по ним;
- встроенные аудио и видеозвонки для индивидуальных, групповых и командных совещаний;
- общее хранилище 10 ГБ плюс дополнительные 2 ГБ на каждого пользователя;
- интеграция с другими продуктами компании Microsoft, в том числе: Word, Excel, PowerPoint и OneNote;
- интеграция со 140 сторонними сервисами и приложениями, включая: Adobe, Evernote и Trello;
- возможность общения и взаимодействия с любыми участниками, даже не входящими в состав вашей команды.

Далее рассмотрим плюсы Microsoft Teams.

1 Повышение производительности и расширение возможностей общения.

Широкий спектр инструментов совместной работы, предлагаемых в Microsoft 365, часто может дезориентировать пользователей. Команды увеличивают продуктивность, делая все ваше сотрудничество – разговоры, чаты, онлайн-встречи, общие файлы, задачи и т. д. доступными в одном приложении и едином интерфейсе. В дополнение к широко используемым приложениям от Microsoft, имеется широкий выбор сторонних приложений, которые интегрируют в команды

2 Лучше сосредоточенность на своей работе.

С Teams можно сосредоточить своё внимание на рабочих темах и лучше расставить приоритеты. Вместо того, чтобы тратить время на просмотр почтового ящика, содержащего сообщения от всех возможных источников. Teams разделяет информацию по каналам и помещает её в контекст. Канал содержит только разговоры, встречи и документы, относящиеся к теме этого конкретного канала. Таким образом

вместо того, чтобы отвлекаться на случайную электронную почту или тратя время, пытаясь очистить почтовый ящик, можно сосредоточить внимание и тратить время на каналы, по которым вы действительно выполняете свою работу.

3 Повышение прозрачности.

Большая проблема для многих организаций – держать всех в курсе и заставить всех чувствовать, что они знают, что происходит. С Teams это становится намного проще. Вам больше не нужно беспокоиться о том, кому поставить копию в электронном письме или кого пригласить на встречу. С помощью Teams вы публикуете своё сообщение в соответствующем канале, упоминая людей с помощью символа @, от которых хотите получать уведомления и, оставляете его для чтения всем остальным.

Беспроблемный переход к цифровому рабочему месту.

Переход на цифровое рабочее место в облаке даёт так много преимуществ для бизнеса. Так как Teams – это полностью облачное решение, переход в облако происходит без проблем. Возможность доступа к информации из любого места на любом устройстве – огромное преимущество для бизнеса. Кроме того, разместив рабочее место в облаке, вы сможете воспользоваться преимуществами обработки больших данных и искусственного интеллекта, чтобы работать эффективнее.

4 Если вы работаете в многоязычной организации, вы можете воспользоваться автоматическим переводом разговоров.

5 Если вы пропустили встречу, вы можете легко настроиться на ту часть встречи, которая вас интересует, выполнив поиск определенных слов, которые были сказаны в видеозаписи.

6 Новые члены команды могут быстрее освоиться.

На сегодняшнем гибком рабочем месте, где мы можем легко найти таланты из любой точки мира, очень важно быстро научить людей работать. С Teams вы можете быстро обучить новых членов команды и вовлечь их в командную работу – независимо от того, в какой точке мира они находятся.

Одним из самых крупных конкурентов Teams, является Zoom. Каждая из этих относительно новых платформ быстро развивалась за последние пару лет накопив впечатляющее количество функций и поклонников. На очень высоком уровне Microsoft Teams и Zoom пересекаются и конкурируют друг с другом в том, что они предлагают набор услуг для видеоконференцсвязи (включая конференц-системы) и телефонии. Более глубокое изучение более тонких функций, UX (User Experience), ценообразование и интеграции позволяют организациям оценивать компромиссы и принимать наилучшее решение о том, какая платформа лучше всего подходит.

Teams представляет собой более сложную задачу, поскольку пользователям необходимо быстро освоить способы взаимодействия в разных каналах и командах, включить общий доступ к файлам, а также все другие приложения Office 365, встроенные в Teams. Полный набор встроенных в Teams функциональных возможностей для совместной работы в рамках рабочего процесса явно позволяет ему предлагать более широкую область использования и сценарии (и, следовательно, лучшую ценность), чем Zoom.

Microsoft Teams и Zoom предлагают бесплатные версии платформ с более продвинутыми функциями, предлагаемыми с платными планами. Бесплатная версия Microsoft Teams включает ограниченный чат и совместную работу, приложения и службы для повышения производительности, собрания и звонки, а также безопасность. В бесплатной версии отсутствуют две большие части, включая инструменты администрирования или поддержку Microsoft.

Бесплатная версия Zoom включает в себя собрания, в которых могут принимать участие до 100 участников (с 40-минутным ограничением для групповых встреч), неограниченное количество встреч один на один, онлайн-поддержку, а также функции видео и веб-конференций, функции групповой совместной работы и безопасность. План Microsoft Premium немного дешевле для пользователя, чем аналогичный план Zoom Pro, но для их корпоративных планов они аналогичны.

Возможность совершать звонки на уровне предприятия имеет решающее значение, особенно для бизнес-коммуникаций с помощью видео, голоса, конференц-связи и обмена сообщениями. Эта категория функциональных возможностей изначально была оплотом Microsoft, так как Zoom изначально не предлагала телефонных систем. Microsoft обращается к UC Telephony с помощью телефонной системы Microsoft и тарифных планов или Microsoft Phone System с прямой маршрутизацией. В значительной степени из-за способности Teams «переносить» всю эту подкатегорию непосредственно из своего наследия Skype для бизнеса, корпоративная голосовая связь была серьезным направлением для Teams, поскольку Microsoft планирует использовать Teams для замены Skype для бизнеса в качестве основного инструмента компаний.

Кроме сказанного, Microsoft Teams – это тесная встроенная интеграция приложений Office 365. Существует более 100 интеграций для Microsoft Teams, которые включают опции для управления билетами, опросами, прогнозом погоды, новостей и т.д.

Таким образом, Microsoft Teams имеет возможность настройки и поддержки широкого спектра приложений и служб для интеграции использования всех основных функций, которые можно ожидать от приложения для организации рабочего процесса. Вся команда может подключиться к одной панели инструментов, таким образом, легко общаться с другими участниками. Имеются возможности делиться важными документами, заметками и т. п. Со своей командой планировать задачи и проводить конференции. Все действия команды видны, что очень полезно и выгодно для руководителя группы или руководителей. Можно вести видеосвязь внутри и вне организации из любого места и любой платформы. Microsoft Teams можно безупречно использовать в смартфонах, что действительно делает его очень привлекательным продуктом особенно во время пандемии COVID-19. Можно совместно использовать и удалённо управлять устройством по запросу и с одобрения клиентов или поставщиков, чтобы делиться решениями.

ЛИТЕРАТУРА

1. Microsoft. Общие сведения о безопасности. [Электронный ресурс] / MDN. – Режим доступа: <https://msdn.microsoft.com/>. – Дата доступа: 18.09.2020.
2. Wikipedia. [Электронный ресурс] / Wiki. – Режим доступа: https://en.wikipedia.org/wiki/Microsoft_Teams. – Дата доступа: 18.09.2020.

УДК 004.05

**МИКРОСЕРВИСНЫЕ АСИНХРОННЫЕ АРХИТЕКТУРЫ
КАК РЕШЕНИЕ СЛОЖНЫХ БИЗНЕС-ПРОЦЕССОВ****П.Д. ТАЛАЙКО***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)*

Рассмотрены особенности взаимодействия программных компонентов в асинхронных микросервисных архитектурах. Произведено знакомство и спецификой работы с современными Message Brokers. Обоснована целесообразность использования асинхронного подхода в распределённых системах. Рассмотрен основной принцип работы одной из самых популярных очередей сообщений Kafka.

Часто при разработке программных продуктов приходится сталкиваться с реализацией каких-то бизнес-задач в рамках сложных процессов. IT-сообщество (от англ. Information Technology) придумало множество различных решений основанных на декомпозиции таких процессов. С появлением инструментов и вычислительных возможностей в построении высоконагруженных архитектур, многие разработчики разбили свои огромные приложения (монолиты) на модули, работающие над определенным подпроцессами, но столкнулись с рядом проблем в ходе создания микросервисной архитектуры (МА), а именно:

- усложненное взаимодействие модулей системы;
- управление транзакциями между модулями;
- дорогая стоимость поддержки (каждый модуль должен иметь свой набор ресурсов);
- усложнение разработки и запуска ПО (CI/CD (англ. Continuous Integration/Continuous delivery), Jenkins, TeamCity, Docker, OpenShift и т. д.).

Правильно обозначенные задачи, позволят правильно выстроить процессы и подпроцессы. Декомпозиция должна происходить строго по линии данных и функций, за которые отвечает данный программный компонент. Для примера можем провести модуль, отвечающий за авторизацию и аутентификацию пользователя в системе. Данный модуль назовём «security-service» и возложим на него определенные функции и данные. Данными, которыми будет оперировать модуль – эта информация о пользователе и его роли в системе. Может возникнуть справедливый вопрос, как нам получать данные пользователя в рамках другого сервиса и как взаимодействовать с «security-service» модулем? Ответ очень прост – вынести логику авторизации и аутентификации в общий инфраструктурный компонент и подключать его к различным модулям в зависимости от необходимости получения данных о пользователе и его ролях.

Инфраструктура микросервисной архитектуры – это набор сервисов или библиотек, которые направлены на вынесение реализации общей функциональности. К сервисам инфраструктура можно отнести сервис конфигураций «config-service» и сервис регистрации сервисов архитектуры «discovery-service». Основная концепция направлена на централизацию конфигураций с помощью «config-service» и обращение к сервису по его имени, независимо от места физического запуска сервиса, за это отвечает «discovery-service». Для создания инфраструктуры существуют готовые решения, и одно из таких представлено во framework Spring в модуле Cloud или коротко Spring Cloud [1].

После того как централизована конфигурация и к сервису можно обратиться по имени, не зная где данный сервис физически запущен, перейдём к главной теме асинхронному взаимодействию микросервисов между собой. Долгое время закладывалась концепция о клиент-серверном взаимодействии – «совершая запрос в синхронном режиме получаем ответ (результат)». Представим, когда необходимо выполнить ряд операций по одному запросу, допустим, пользователь в личном кабинете нажимает кнопку «Получать уведомления» и сервис получает событие о начале отправки уведомлений для конкретного пользователя. Уведомления могут приходиться пользователю на почту, в телеграм канал, мобильный телефон и все источники должны уметь отправлять пользователю уведомление в соответствующий канал. Следует отметить, что после нажатия пользователем кнопки «Получать уведомления», отправится запрос на изменение статуса уведомления в систему, после чего, его данные попадут в выборку участвующих в рассылке уведомлений, а конкретный сервис, умеющий работать с мессенджерами, почтой, мобильными рассылками будет сообщать о происходящих событиях.

Каждый сервис может иметь собственный компонент представления (JSP, JS, HTML, CSS), через который происходит взаимодействие с пользователем.

Далее подробнее рассмотрим архитектуру, представленную на рисунке 1. Пользователь отправляет HTTP запрос и получает синхронно ответ HTTP 200 OK о включении его в список подписчиков уведомлений, сервис «user-service» изменяет в схеме <project_name>_USER базы данных информацию о статусе подписки, а другой модуль высылающий уведомления сервисам рассылки в фоновом процессе ожидает сообщение от message broker. При поступлении нового уведомления производит создание сообщений по

средствам выборки пользователей, нуждающихся в уведомлениях, и собственно контента уведомления. Последним этапом формируется запрос через другую очередь сообщений для конкретных рассылщиков (телеграм, почта, мобильный сервис).

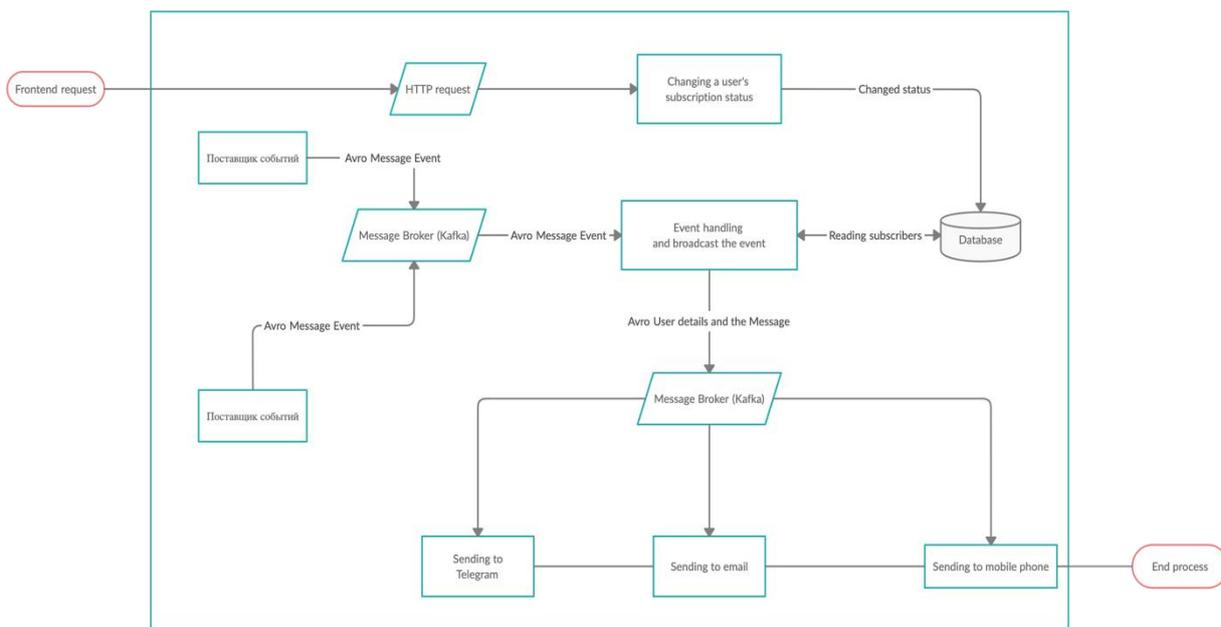


Рисунок 1. – Архитектура рассылки уведомлений пользователю

Далее рассмотрим, как могут взаимодействовать различные сервисы в микросервисной архитектуре. Выше было описано одно из возможных взаимодействий посредством общей базы данных, но в большинстве случаев необходимо просто передать другому сервису информацию, не отображая и не храня её.

Рассмотрим одну из самых известных очередей сообщений (message broker) Kafka. Схематичный принцип работы данного инструмента представлен на рисунке 2.

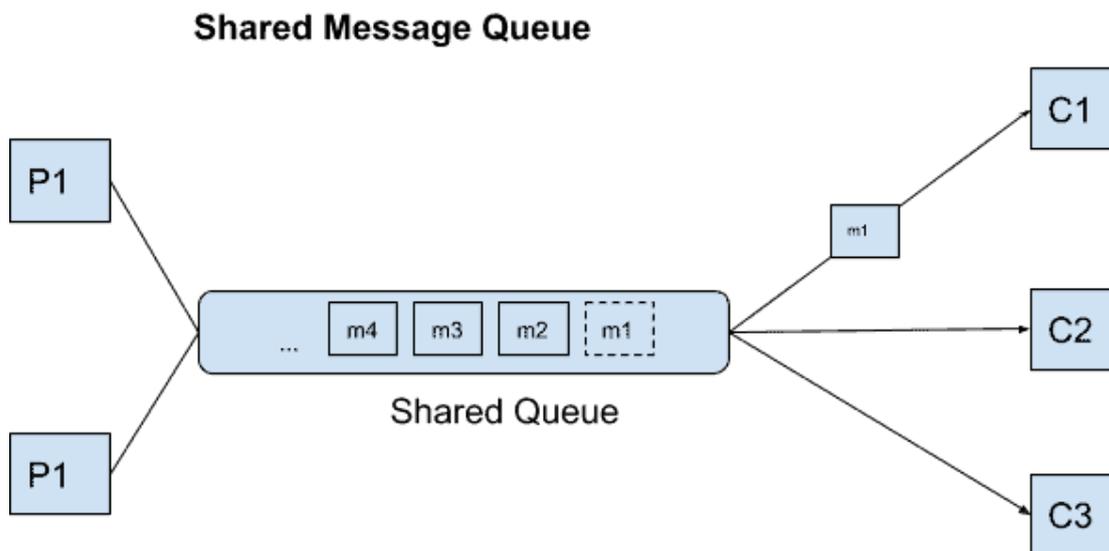


Рисунок 2. – Схематичный прототип работы Kafka

Kafka позволяет обмениваться сообщениями в больших объёмах и при этом хранить состояние очереди. Главная концепция заключается в наличии поставщиков, потребителей и групп потребителей. Лёгкая интеграция с любой платформой делает данную очередь сообщений инструментом по умолчанию во многих проектах. Необходимо чётко понимать то, какие сообщения необходимы каким сервисам и правильно

определять группы сервисов, так как от этого зависит получат ли они из очереди сообщения. Если обратиться к примеру архитектуры выше и объединить наших отправителей сообщений пользователю в одну группу, то сообщение будет отправляться только одним из сервисов, так как остальным сообщение из очереди не будет доступным. Элемент очереди представлен в виде набора byte, поэтому если работаем с определенным видом данных как String, Number, Object, то необходимо определить сериализаторы и десериализаторы для того или иного типа данных. Многие реализации producer-consumer под различные платформы для Kafka имеют встроенные инструменты сериализации и десериализации, поэтому работать с данным брокером становится только приятнее. Стоит отметить, что существует возможность передавать объекты по каналу, и в мире Java данным протоколом общения выступает Avro. Apache Avro позволяет определять протокол Java-объекта, который будет передаваться в очередь Kafka. Таким образом можно обмениваться целыми Java-объектами, что делает работу проще и понятнее. Kafka отлично подойдет для архитектур где важно хранить состояние очереди, иметь возможность управлять стратегией получения сообщения и передачи их большего количества между сервисами.

В ходе данного исследования были выявлены основные преимущества асинхронных архитектур, а именно: обработка пользовательских запросов в фоне, горизонтальное масштабирование системы, большие возможности интеграции со сторонними системами. Микросервисные асинхронные архитектуры позволяют исполнить процессы, где важно эффективно выполнить задачу, но при этом не ожидать ответ, пока тот или иной подпроцесс выполняется. Важное преимущество такой архитектуры – это подключение к расширению или созданию новой функциональности другими разработчиками, так как большие задачи, как правило, реализуются с большими затратами человеческих ресурсов. Модули системы слабосвязанные и имеют протоколы связи, которые безболезненно могут быть заменены. Были исследованы возможные взаимодействия в распределённых системах.

ЛИТЕРАТУРА

1. Синхронные архитектуры. Основы Spring Cloud. / Создание микросервисов Сэм Ньюмен [Электронный ресурс]. – Режим доступа: https://vk.com/doc7608079_449808456?hash=330da3096bb49d5549 – Дата доступа: 29.08.2020.
2. Асинхронные архитектуры. Инфраструктурные сервисы архитектуры. Spring Cloud Config. Spring Cloud Foundry. Взаимодействие компонентов в микросервисных архитектурах. / Java в облаке Джош Лонг, Кеннет Бастанни [Электронный ресурс]. – Режим доступа: <https://vk.com/doc26879026473369384?hash=191b74beba25cb6661> – Дата доступа: 13.06.2020.
3. Основные понятия и быстрый старт с Kafka / Документация Kafka [Электронный ресурс]. – Режим доступа: <https://kafka.apache.org/documentation/#ecosystem> – Дата доступа: 14.06.2020.
4. Spring Cloud Netflix Microservices / Статья Kirill Sereda [Электронный ресурс]. – Режим доступа: <https://goo-gl.su/yQczFctA> – Дата доступа: 15.06.2020.

УДК 004.05

МИГРАЦИЯ БАЗ ДАННЫХ В СОВРЕМЕННОЙ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

П.Д. ТАЛАЙКО

(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЕНОК)

Даны понятия миграции баз данных. Рассмотрен популярный инструмент миграции и примеры его использования в современной разработке программного обеспечения.

В крупных проектах необходимо организовать администрирование баз данных (БД), которые содержат большое количество различных таблиц, представлений, индексов и пр. Так как процесс отслеживания изменений объектов БД или данных очень сложный, то на помощь приходят миграторы, которые помогают решить проблемы с версионированием и инициализацией схем БД и тех данных, которые в них сосредоточены.

Основной целью исследования является обзор понятия миграции и одного из популярных миграторов Liquibase.

Архитектурная модель современного web-проекта, как правило, состоит из трёх компонентов:

- клиента;
- сервера приложения;
- сервера БД.

Наиболее детально рассмотрим только последний компонент, на который возлагается очень важная функция – предоставления и хранения данных. Частный случай предполагает у одного экземпляра приложения одну БД или схему, поэтому в качестве примера рассмотрим миграцию БД на одном из серверов БД.

Миграция БД подразумевает переход от одной структуры БД к другой без потери консистентности. Консистентность данных – это согласованность данных друг с другом, целостность данных, а также внутренняя непротиворечивость [4]. Множество всех условий, налагаемых на данные, определяется структурой данных. Миграция подразумевает как прямое изменение схемы БД или данных, так и обратное изменение – откат изменений.

Использование инструментов миграции позволяет разработчику избежать проблем при смене БД. Не составляет никакого труда изменить систему управления базами данных (СУБД), так как такие инструменты подстраиваются под диалект выбранной СУБД, используя свой формат описания схем и возможностей загрузки данных (например, при переходе с СУБД Oracle на PostgreSQL). Инструменты, созданные под разные платформы, языки программирования и разработки позволяют без затруднений агрегировать свой функционал в любой проект.

Далее рассмотрим реально пользующийся популярностью у разработчиков инструмент разработки под названием Liquibase, который берёт на себя функциональные возможности мигратора БД. Библиотека Liquibase написана на языке Java и отлично интегрируется с таким знаменитым фреймворком как Spring. Liquibase можно использовать как отдельно так и в сборке с проектом в виде зависимости Maven или Gradle. Если упростить описание, то инструмент представлен в виде jar-файла, который при запуске принимает параметры подключения и путь к файлу описания схемы базы данных в формате XML, JSON или YAML. Пример запуска мигратора представлен в листинге 1.

Листинг 1 – Запуск мигратора через командную строку

```
java -jar liquibase.jar \  
  --driver=oracle.jdbc.OracleDriver \  
  --classpath=\path\to\classes;jdbcdriver.jar \  
  --changeLogFile=com/example/db.changelog.xml \  
  --url="jdbc:oracle:thin:@localhost:1521:oracle" \  
  --username=scott \  
  --password=tiger \  
Update
```

При запуске миграции обычно не возникает никаких сложностей за исключением наличия драйвера необходимого под ту или иную БД. К сожалению, из описания, представленного в листинге 1 сложно представить, насколько облегчает Liquibase работу в создании схемы БД, отслеживания изменений и загрузки в таблицы данных, однако на практике – это так.

В Liquibase представлены ключевые понятия как:

- databaseChangeLog,
- changeset.

Databasechangelog – тег или элемент, который включает в себя наборы changeSet.

Changeset – тег или элемент, описывающий создаваемый при исполнении объект БД (таблица, индекс, грант и т. д.). Пример такого файла представлен в листинге 2.

Листинг 2 – Databasechangelog файл

```
<databaseChangeLog ...>
  <changeSet author="talayko" id="1585407925310-6">
    <createTable tableName="USERS">
      <column name="ID" type="UUID">
        <constraints nullable="false"/>
      </column>
    </createTable>
  </changeSet>
</databaseChangeLog>
```

Тег databaseChangeLog содержит атрибуты подключения схем валидации xsd и один changeSet, который имеет важные атрибуты, определяющие уникальность данного изменения – AUTHOR и ID. Ниже рассмотрим с какой целью должна присутствовать уникальность каждого changeSet. После запуска мигратора в БД создается таблица USERS с единственным столбцом с именем ID и типом raw(16) – если БД Oracle или UUID – если БД PostgreSQL. В зависимости от выбранного драйвера базы данных используют необходимый диалект Liquibase.

Liquibase создает две технические таблицы в схеме БД:

- DATABASECHANGELOCK,
- DATABASECHANGE.

DATABASECHANGELOCK содержит информацию о общем состоянии миграции. В данной таблице очень важно знать о столбце LOCK, который имеет булевый тип и позволяет понять идёт ли сторонняя миграция в данную таблицу. Текущий процесс миграции будет находиться в состоянии ожидания пока значение LOCK в значении TRUE.

DATABASECHANGE таблица описывающая изменения применённые при миграциях. Liquibase накатывает изменения только те, которые ранее не были записаны в таблицу DATABASECHANGE, а ищет данные изменения по атрибутам AUTHOR и ID, которые были указаны в changeLog. Изменения в ранее применённых changeSet невозможны, мигратор сравнивает check sum MD5 найденных по ID и AUTHOR, который записан в DATABASECHANGE в столбце MD5 и check sum из changeSet файла и если сумма не совпадает, то происходит ошибка миграции. Чтобы избежать данной ошибки, можно создать новый changeSet или удалить строку о старом changeSet из таблицы DATABASECHANGE.

Изучив основные задачи, решаемые мигратором, а также то каким образом он применяет изменения к БД, далее рассмотрим, как можно подключить Liquibase к проекту построенному на основе Spring Boot. Для подключения к проекту необходимо в сборщике проекта указать зависимость или добавить liquibase.jar в lib classpath для дальнейшего использования. Подключение зависимости в pom.xml maven отображено в листинге 3.

Листинг 3 – Зависимость Liquibase в Maven проекте

```
<dependency>
  <groupId>org.liquibase</groupId>
  <artifactId>liquibase-core</artifactId>
</dependency>
```

Далее необходимо прописать местоположение в файле контекста приложения местоположения файла миграции (листинг 4).

Листинг 4 – Property liquibase changelog в файле контекста Spring Boot

```
spring.liquibase.change-log=classpath:/db/changelog/changelog-master.xml
```

Теперь при старте будет происходить миграция схемы и данных, прописанных в файле changelog-master.xml.

Таким образом, рассмотренные основные принципы работы миграторов БД и популярный инструмент миграции Liquibase, а также приведённые примеры его использования в современной разработке программного обеспечения, анализ внутреннего устройства мигратора определили важные составляющие

в виде технических таблиц и принципов определения новых изменений, что позволило осуществить выбор данного инструмента для дальнейшей разработки.

ЛИТЕРАТУРА

1. Официальный сайт Liquibase [Электронный ресурс] / Liquibase | Open source version control for Your Database – Режим доступа: <https://www.liquibase.org/get-started/how-liquibase-works> – Дата доступа: 14.08.2020.
2. Major functionality [Электронный ресурс] / Liquibase Wikipedia – Режим доступа: <https://en.wikipedia.org/wiki/Liquibase> – Дата доступа: 15.08.2020.
3. Откат изменений Liquibase [Электронный ресурс] / Версионирование структуры БД с помощью Liquibase – Режим доступа: <http://easy-code.ru/lesson/database-versioning-liquibase> – Дата доступа: 15.08.2020.
4. Консистентность данных [Электронный ресурс] / Консистентность данных – Режим доступа: <https://dic.academic.ru/dic.nsf/ruwiki/147082> - Дата доступа: 15.08.2020.

УДК 004.021

**ПРИМЕНЕНИЕ ДИСКРЕТНОГО КОСИНУСНОГО ПРЕОБРАЗОВАНИЯ
И ГЕНЕРАТОРА ПСЕВДОСЛУЧАЙНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ
В АЛГОРИТМАХ ЦИФРОВОЙ СТЕГАНОГРАФИИ****А.С. ТАНАНА****(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)**

В статье рассматривается стеганографический алгоритм встраивания текстовой информации в мультимедийный объект на основе дискретного косинусного преобразования и генератора псевдослучайной последовательности.

В современном мире одной из важнейших задач является защита информации, например, обеспечение безопасности хранения криптографических ключей или передачи конфиденциальных данных. Существует множество криптографических методов для решения этой задачи, однако большинство алгоритмов шифрования данных не обладают нужной стойкостью. К каждому алгоритму можно найти обратный метод, который дешифрует закодированное слово. Рациональным решением задачи обеспечения защиты информации является сокрытие факта передачи данных. Стеганография – специальная наука, изучающая методы сокрытия информации, которая должна оставаться не обнаружимой как статистически, так и для человеческого восприятия. В настоящее время такая задача актуальна для мультимедийных объектов: цифровые изображения, звуковые файлы, видеозаписи.

Современная стеганография разделяется на два направления: компьютерная и цифровая стеганография. Первое направление основано на особенностях компьютерной платформы и широко применяется в файловых системах. Второе направление, цифровая стеганография, основано на встраивании дополнительной информации в цифровые объекты. Алгоритмы цифровой стеганографии могут быть направлены как на полное сокрытие внедренной информации от посторонних глаз, так и на наложение специальных знаков на исходный сигнал. Таким образом, цифровую стеганографию можно разделить на следующие группы [1]:

- встраивание информации в непосредственно цифровой сигнал;
- встраивание цифровых водяных знаков;
- встраивание идентификационных номеров;
- встраивание заголовков.

Наибольший интерес представляют алгоритмы первой группы. Совокупность средств и методов, которые используются для формирования скрытого канала передачи, формируют стеганографическую систему, или стегосистему.

Каждый стеганографический метод должен обладать набором качественных характеристик. К ним можно отнести невидимость, устойчивость и объем встраиваемого сообщения. Невидимость скрытой информации достигается за счет особенностей человеческого восприятия. Например, система человеческого зрения наименее чувствительна к изменениям канала синего цвета в изображениях, а система человеческого слуха практически нечувствительна к изменениям фазы звукового сигнала. Оценка стойкости алгоритма основана на его устойчивости к различного рода модификациям и атакам. Под модификацией понимается применение фильтров, изменение объема контейнера, аналогово-цифровое и цифро-аналоговое преобразование. Под атакой на стегосистему понимается попытка обнаружить, извлечь или повредить стеганографическое сообщение. Объем встраиваемого сообщения является немаловажным параметром оценки стеганографического алгоритма, т.к. наибольший интерес представляют алгоритмы, способные внедрить большое количество информации в контейнер. Таким образом, контейнеры большего размера позволяют внедрить большее количество информации.

На сегодняшний день существует множество различных алгоритмов и методов стеганографического внедрения информации, однако большинство из них уже устарели и имеют большую вероятность обнаружения данных. Безусловно, разработаны методы, обеспечивающие высокую надежность, однако их реализация слишком сложна в вычислительном процессе. Анализ существующих методов встраивания информации показывает, что задача разработки стеганографических методов остается актуальной. На основе существующих стеганографических алгоритмов можно создать усовершенствованный алгоритм, который будет обладать достаточной степенью устойчивости к различным преобразованиям и хорошо противостоять методам стегоанализа.

Один из самых распространенных методов сокрытия информации основан на замене наименее значимых бит (LSB) потока данных на биты встраиваемого сообщения. Однако большинство мультимедий-

ных объектов хранятся в сжатом виде. Алгоритмы сжатия построены так, что на этапе квантования происходит округление спектральных коэффициентов, полученных после косинусного преобразования. Идею метода LSB можно применить и в спектральной области [2].

Для получения спектра используют не только дискретное преобразование Фурье, но и дискретное косинусное преобразование. Известно, что коэффициенты, полученные после применения ДКП, упорядочены от более низкочастотных к более высокочастотным (рис. 1). Низкочастотные коэффициенты содержат самую важную информацию для восстановления исходных данных. Высокочастотные коэффициенты можно занулить без существенных потерь после обратного применения ДКП.

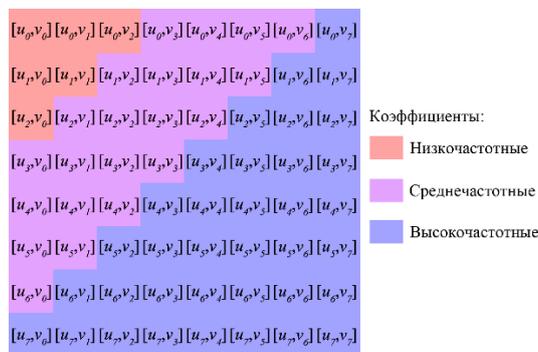


Рисунок 1. – Матрица коэффициентов ДКП размером 8x8

Самый высокочастотный коэффициент массива ДКП выступает в роли наименее значащего бита. Этот коэффициент можно заменить другим значением без риска существенного искажения исходного сигнала. В основе алгоритма будет лежать принцип замены заранее определённого высокочастотного или среднечастотного коэффициента.

Рассмотрим алгоритм внедрения информации. В качестве стекоконтейнера может быть использован как видео-, аудиофайл, так и изображение. Встраиваемое сообщение представляет собой обычное текстовое сообщение, преобразованное в битовую последовательность. Для увеличения стойкости алгоритма полученную битовую последовательность можно закодировать любым помехоустойчивым кодом. Разделим исходный сигнал на блоки размером 8x8 и получаем массив номеров блоков. При этом следует учитывать то факт, что многие контейнеры содержат специальные блоки заголовков, содержащие специальные данные согласно формата контейнера. Встраивание сообщения будем производить не в каждом блоке, а в блоках с номерами, соответствующими номерам из псевдослучайной последовательности чисел (ПСП).

В качестве генератора псевдослучайной последовательности выберем генератор Блюма-Шуба (BBS). Это простейший и наиболее эффективный генератор, использующий сложный теоретический подход разложения на множители [3].

Пусть есть два простых числа p и q . Их произведение n является целым числом Блюма. Выберем другое случайное число x , взаимно простое с n . Тогда

$$x_0 = x^2 \bmod n$$

есть стартовое число генератора. Последующие значения последовательности будут вычисляться по формуле:

$$x_i = x_{i-1}^2 \bmod n$$

Безопасность этой схемы основана на сложности разложения n на множители. Более того, генератор BBS непредсказуем как в левом направлении, так и в правом направлении. Это означает, что, получив последовательность, криптоаналитик не сможет предугадать ни следующий, ни предыдущий номер последовательности.

После того, как контейнер и сообщение преобразованы, можно приступить непосредственно к внедрению информации:

1. Для каждого блока, соответствующему номеру из полученной ПСП, применяем дискретное косинусное преобразование.
2. Извлекаем первый бит из битовой последовательности, полученной после кодирования сообщения.
3. Выбираем значение уровня шума ($s > 0$). Чем выше уровень шума, тем выше уровень искажения сигнала.
4. Наиболее высокочастотный коэффициент заменяется значением уровня шума следующим образом:

$$F[N - 1] = \begin{cases} -s, & b = 1 \\ s, & b = 0 \end{cases}$$

где F – массив коэффициентов ДКП; N – размерность массива коэффициентов; s – значение уровня шума, b – встраиваемый бит сообщения.

5. К текущему блоку применяется обратное дискретное косинусное преобразование.

Процесс повторяется до тех пор, пока в битовой последовательности есть значения.

Предложенный алгоритм обладает высокой надежностью, а также приемлемой пропускной способностью и сложностью реализации. Заполненный контейнер не будет отличаться от оригинального контейнера в силу человеческого восприятия.

ЛИТЕРАТУРА

1. Садов, В.С. Компьютерная стеганография / В.С. Садов – Минск: БГУ, 2010. – 211 с.
2. Blackledge, J. Resilient Digital Image Watermarking for Document Authentication / J. Blackledge, O. Iakovenko // IAENG International Journal of Computer Science. – 2014. – № 41(1). – С. 1-17.
3. Горелкина Д.А., Дорошенко Н.С., Осипов Д.Л. Применение методов цифровой стеганографии для внедрения конфиденциальной информации в растровые изображения // Вестник Ставропольского государственного университета. – 2011. – №75. – С. 75-76.
4. Грибунин В.Г., Оков И.Н., Туринцев В.И. Цифровая стеганография. – М. : СОЛОН-Пресс 2002. 272 с.

УДК 004.021

**ПРОЕКТИРОВАНИЕ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ
ДЛЯ ВСТРАИВАНИЯ ИНФОРМАЦИИ В АУДИОФАЙЛ
НА ОСНОВЕ МОДИФИКАЦИИ СПЕКТРАЛЬНЫХ КОЭФФИЦИЕНТОВ.
ОЦЕНКА СТЕГАНСТОЙКОСТИ СИСТЕМЫ**

А.С. ТАНАНА

(Представлено: канд. физ.-мат. наук, доц. Ю.Ф. ПАСТУХОВ)

В статье рассматриваются вопросы разработки и создания графического интерфейса пользователя для встраивания текстовой информации в звуковой файл. Выбор технологии разработки, определение основных функциональных возможностей. Оценка стеганостойкости разработанной системы.

Защита информации, в том числе конфиденциальных данных, становится актуальной задачей с развитием информационных технологий. Решение такой задачи даёт стеганография – наука, изучающая алгоритмы и методы сокрытия данных в мультимедийных объектах, которая должна оставаться не обнаружимой ни статистически, ни для человеческого восприятия. Среди возможностей стеганографии можно выделить встраивание цифровых водяных знаков, идентификационных номеров, заголовков.

Каждый стеганографический алгоритм должен удовлетворять ряду качественных характеристик. К ним можно отнести надёжность, устойчивость, объём встраиваемых данных. Большинство алгоритмов лишь частично соответствуют заданным требованиям, поэтому задача разработки стеганостойкого алгоритма встраивания данных остается актуальной задачей и в настоящее время.

Алгоритм встраивания данных на основе замены высокочастотных коэффициентов с применением генератора псевдослучайной последовательности является усовершенствованным методом существующих методов. Для оценки стеганостойкости алгоритма необходимо провести тестирование функциональности внедрения и извлечения сообщения, а также провести ряд модификаций и атак на стегоконтейнер.

В рамках данной статьи в качестве стегоконтейнера выберем звуковой файл формата WAVE. Данный формат содержит аудиоданные в несжатом виде, что позволяет хранить информацию в наилучшем качестве. В качестве встраиваемого сообщения будем использовать текстовые данные.

Для разработки программного обеспечения будем использовать возможности платформы .Net Framework: язык программирования C#, графический интерфейс пользователя Windows Forms, среду разработки Visual Studio 2015. Разработанная программа должна решать следующие задачи:

- встраивание текстовых данных в аудиофайл;
- извлечение данных из стегоконтейнера в целостном виде.

Дизайн пользовательского интерфейса должен быть удобным, интуитивно понятным и привлекательным.

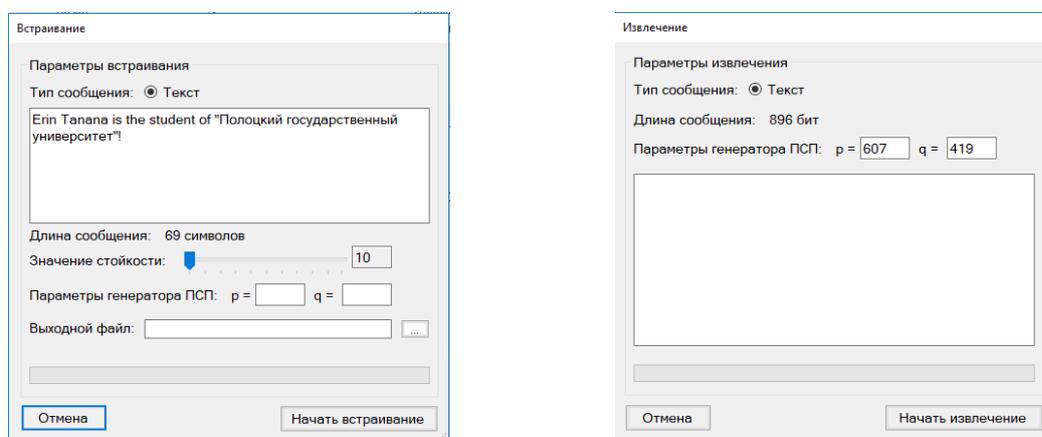
В основе алгоритма встраивания сообщения лежит дискретное косинусное преобразование. Рассмотрим программную реализацию преобразования на языке C#:

Листинг 1. Дискретное косинусное преобразование

```
1: public static void DCT(double[] data){
2: double[] result = new double[data.Length];
3: double c = Math.PI / (2.0 * data.Length);
4: double scale = Math.Sqrt(2.0 / data.Length);
5: for (int k = 0; k < data.Length; k++) {
6: double sum = 0;
7: for (int n = 0; n < data.Length; n++)
8: sum += data[n] * Math.Cos((2.0 * n + 1.0) * k * c);
9: result[k] = scale * sum;
10: }
11: data[0] = result[0] / Math.Sqrt(2.0);
12: for (int i = 1; i < data.Length; i++)
13: data[i] = result[i];
14: }
```

Интерфейс пользователя включает в себя следующие окна: главное окно приложения, окно встраивания сообщения, окно извлечения сообщения (рис. 1). На главном окне приложения пользователь выбирает файл и тип операции: встраивание или извлечение. Здесь же отображаются основные данные о выбранном аудиофайле. Перед встраиванием пользователь вводит сообщение, выбирает значение стойкости

и начальные параметры генератора псевдослучайной последовательности. После указания пути выходного файла и при корректности введённых данных можно начать процесс встраивания сообщения. Для извлечения сообщения необходимо знать длину скрытого сообщения и начальные параметры генератора, а также исходный аудиофайл.

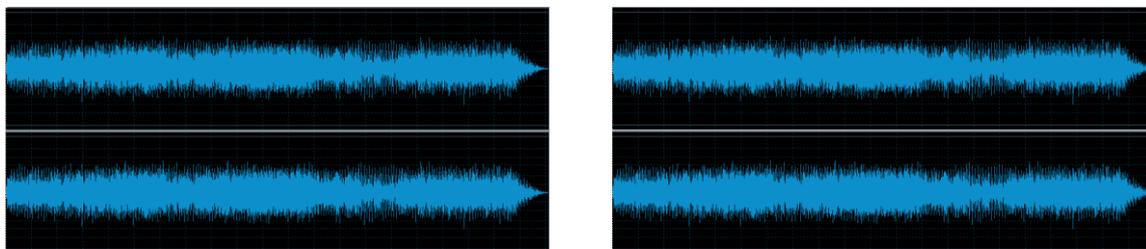


а

б

Рисунок 1. – Интерфейс пользователя:
а – окно встраивания сообщения; б – окно извлечения сообщения

Для демонстрации стеганографической модификации контейнера выберем аудиоcontainer в формате WAVE. В нашем случае, это файл `audiofile.wav` (размер 35.5МБ). В качестве встраиваемого сообщения будем использовать текстовое сообщение «Erin Tanana is the student of “Полоцкий государственный университет”!» длиной 896 бит. Выберем значение стойкости $s = 156$, в качестве входных параметров генератора $p = 607$, $q = 419$. Зададим путь выходного файла и встроим сообщение. Сравним полученный стегоcontainer с исходным containerом:



а

б

Рис. 2. Сравнение исходного и заполненного контейнера:
а – исходный контейнер; б – заполненный контейнер

Визуальных отличий волновой формы нет, при прослушивании файла заметных искажений нет, значит выбранный контейнер пригоден для стеганографической модификации. Визуальные отличия не были обнаружены и при выборе контейнеров, содержащих монотонный звук. Таким образом, для встраивания сообщения пригоден любой контейнер. Отметим, что для встраивания сообщений большого объёма целесообразно использовать аудиофайлы большего размера, однако это приводит к увеличению вычислительного процесса.

Для определения стойкости разработанной стеганосистемы проведём несколько атак и модификаций: применение эффекта искажения звука, удаление 25% данных в конце файла, удаление 50% данных. В качестве исходного контейнера положим рассмотренный ранее контейнер `audiofile.wav`. Входные параметры также оставим неизменными. Как уже отмечалось, визуальных изменений волновой формы замечено не было, как и искажений звука при прослушивании. Результаты проведённых испытаний отображены в таблице.

Таблица 1. Результаты модификаций и атак

Название атаки	Результат
Эффект искажения звука	Сообщение успешно извлекается
Удаление 25% данных в конце файла	Сообщение успешно извлекается
Удаление 50% данных	Сообщение извлечь не удалось

Делаем вывод, что разработанный алгоритм обладает приемлемой надежностью и устойчивостью. В перспективе дальнейшей разработки планируется расширить функционал приложения и перевести приложение на веб-технологии.

ЛИТЕРАТУРА

1. Грибунин В.Г., Оков И.Н., Туринцев В.И. Цифровая стеганография. – М. : СОЛОН-Пресс 2002. 272 с.
2. Blackledge, J. Resilient Digital Image Watermarking for Document Authentication / J. Blackledge, O. Iakovenko // IAENG International Journal of Computer Science. – 2014. – № 41(1). – С. 1-17.
3. Горелкина Д.А., Дорошенко Н.С., Осипов Д.Л. Применение методов цифровой стеганографии для внедрения конфиденциальной информации в растровые изображения // Вестник Ставропольского государственного университета. – 2011. – №75. – С. 75-76.
4. Садов, В.С. Компьютерная стеганография / В.С. Садов – Минск: БГУ, 2010. – 211 с.

УДК 004.05

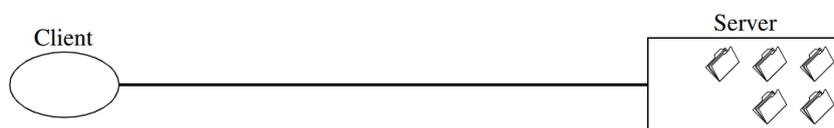
**ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ АРХИТЕКТУРНОГО СТИЛЯ ВЗАИМОДЕЙСТВИЯ
REPRESENTATIONAL STATE TRANSFER****Д.С. ТАТАРИН***(Представлено: канд. техн. наук, доц. И.Б. БУРАЧЁНОК)*

Рассмотрены особенности архитектурного стиля взаимодействия компонентов распределённого приложения в сети – REST и примеры его использования. Показаны основные преимущества при использовании набора ограничений, учитываемых при проектировании (разработке) распределённой системы с использованием компонентов REST.

Representational State Transfer или же как его часто сокращают REST – это архитектурный стиль взаимодействия компонентов распределённого приложения в сети. REST представляет собой согласованный набор ограничений, учитываемых при проектировании распределённой гипермедиа-системы [1]. В определённых случаях (интернет-магазины, поисковые системы, прочие системы, основанные на данных) применение распределённой системы с использованием компонентов REST приводит к повышению производительности и упрощению архитектуры. В широком смысле компоненты в REST взаимодействуют наподобие взаимодействия клиентов и серверов во Всемирной паутине. REST является альтернативой RPC (Remote Procedure Call – вызов удалённых процедур).

Так же очень часто наряду с аббревиатурой REST используется понятие RESTful. Им называют веб-сервис, который следует всем шести ограничениям, накладываемым на взаимодействие между сервером и клиентом. Ниже рассмотрим их более подробно.

Клиент-серверная архитектура. Принцип, лежащий в основе ограничений клиент-сервер, заключается в разделении ответственности [1]. Отделение проблем пользовательского интерфейса от проблем хранения данных улучшает переносимость пользовательских интерфейсов на нескольких платформах. Это также улучшает масштабируемость за счет упрощения серверных компонентов. Возможно, наиболее важным для Интернета является то, что разделение позволяет компонентам развиваться независимо, таким образом поддерживая требования в масштабе Интернета для нескольких доменов организации. Пример клиент-серверной архитектуры можно увидеть на рисунке 1.

**Рисунок 1. – Клиент-серверная архитектура**

Отсутствие состояния. Связь клиент-сервер ограничена отсутствием клиентского контекста, хранящегося на сервере между запросами [1]. Каждый запрос от любого клиента содержит всю информацию, необходимую для обслуживания запроса, а состояние сеанса сохраняется в клиенте. Состояние сеанса может быть передано сервером другой службе, такой как база данных, для поддержания постоянного состояния в течение определенного периода времени и обеспечения аутентификации. Клиент начинает отправлять запросы, когда он готов к переходу в новое состояние. Пока один или несколько запросов не выполнены, считается, что клиент находится в процессе перехода. Представление каждого состояния приложения содержит ссылки, которые можно использовать в следующий раз, когда клиент решит инициировать новый переход между состояниями. Отсутствие состояния представлено на рисунке 2.

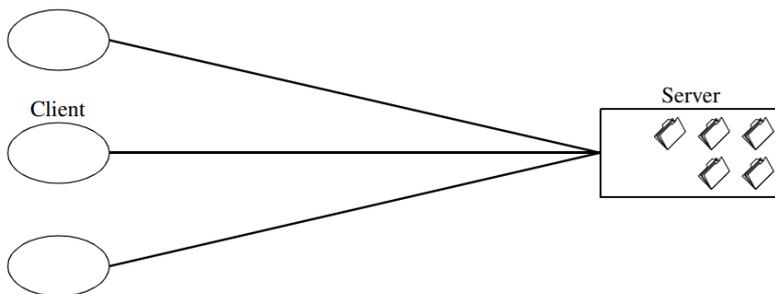


Рисунок 2. – Отсутствие состояния

Кешируемость. Как и во всемирной паутине, клиенты и посредники могут кэшировать ответы [1]. Ответы должны, неявно или явно, определять себя как кэшируемые или некешируемые, чтобы клиенты не предоставляли устаревшие или несоответствующие данные в ответ на дальнейшие запросы. Хорошо управляемое кэширование частично или полностью исключает некоторые взаимодействия клиент-сервер, дополнительно улучшая масштабируемость и производительность. Пример кешируемости можем представить на рисунке 3.

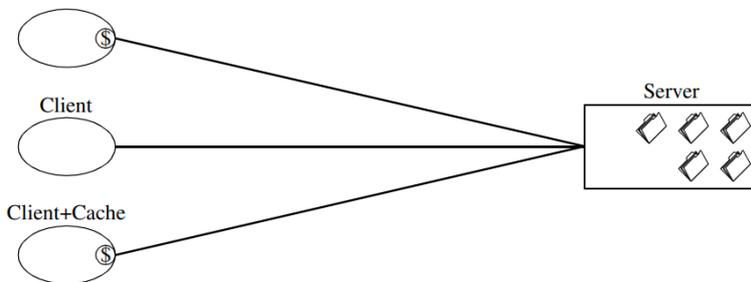


Рисунок 3. – Кешируемость

Многоуровневая система. Обычно клиент не может определить, подключен ли он напрямую к конечному серверу или к посреднику по пути [1]. Если между клиентом и сервером размещен прокси-сервер или балансировщик нагрузки, это не повлияет на их взаимодействие, и не потребуются обновлять код клиента или сервера. Промежуточные серверы могут улучшить масштабируемость системы за счет включения балансировки нагрузки и предоставления общих кешей. Кроме того, безопасность может быть добавлена как слой поверх веб-служб, а затем четко отделить бизнес-логику от логики безопасности. Добавление безопасности в качестве отдельного уровня обеспечивает соблюдение политик безопасности. Наконец, это также означает, что сервер может вызывать несколько других серверов для генерации ответа клиенту. Пример многоуровневой системы можем представить на рисунке 4.

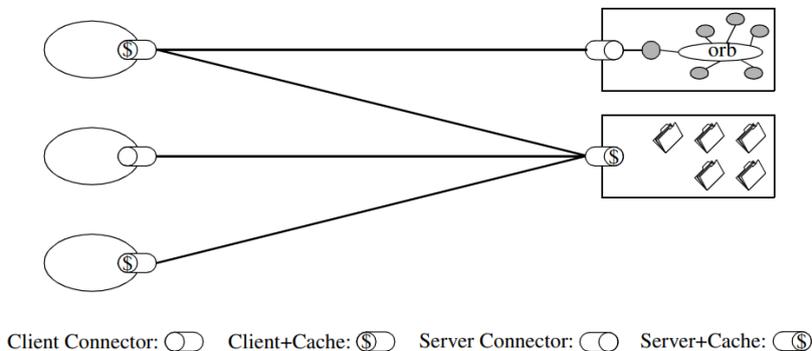


Рисунок 4. – Многоуровневая система

Код по запросу, это правило является не обязательным. Серверы могут временно расширять или настраивать функциональность клиента, передавая исполняемый код: например, скомпилированные компоненты, такие как апплеты Java, или сценарии на стороне клиента, такие как JavaScript.

Uniform interface. Ограничение унифицированного интерфейса является фундаментальным для проектирования любой системы RESTful [1]. Оно упрощает и разделяет архитектуру, что позволяет каждой части развиваться независимо. Далее рассмотрим четыре ограничения для этого унифицированного интерфейса.

– Идентификация ресурса в запросах. Отдельные ресурсы идентифицируются в запросах, например, с помощью URI (Uniform Resource Identifier) – унифицированный идентификатор ресурса в веб-службах RESTful. Сами ресурсы концептуально отделены от представлений, возвращаемых клиенту. Например, сервер может отправлять данные из своей базы данных в формате HTML, XML или JSON, ни один из которых не является внутренним представлением сервера.

– Управление ресурсами посредством представлений. Когда клиент содержит представление ресурса, включая любые прикрепленные метаданные, у него достаточно информации для изменения или удаления состояния ресурса.

– Самодостаточные сообщения. Каждое сообщение содержит достаточно информации, чтобы описать, как его обрабатывать. Например, анализатор для вызова может быть определен типом носителя.

– Гипермедиа как двигатель состояния приложения (HATEOAS). Получив доступ к начальному URI для приложения REST – аналогично доступу обычного пользователя Web к домашней странице веб-сайта – клиент REST должен иметь возможность динамически использовать предоставляемые сервером ссылки для обнаружения всех необходимых ему доступных ресурсов. По мере доступа сервер отвечает текстом, который включает гиперссылки на другие доступные в настоящее время ресурсы. Клиенту не нужно жестко запрограммировать информацию о структуре или динамике приложения [2].

Таким образом, придерживаясь архитектурного принципа REST получаем легко масштабируемые приложения, простую архитектуру и снижаем нагрузку на вычислительные ресурсы. Принцип REST предлагается использовать при реализации проекта, а именно при построении бэкенда с использованием RESTful веб-сервиса для удобного взаимодействия с различными фронтэндными приложениями.

ЛИТЕРАТУРА

1. Architectural Styles and the Design. Диссертация Roy Thomas Fielding. [Электронный ресурс] / ics.uci.edu. – Режим доступа: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf – Дата доступа: 20.09.2020.
2. RESTful. Статья про рестфулл приложения. [Электронный ресурс] / habr.com. – Режим доступа: <https://habr.com/ru/post/319984/> – Дата доступа: 20.09.2020.

УДК 004.05

ОСНОВНЫЕ ТЕХНОЛОГИИ SPRING FRAMEWORK

Д.С. ТАТАРИН

(Представлено: канд. техн. наук, доц., доц. И.Б. БУРАЧЁНОК)

Рассмотрены особенности популярного фреймворка для разработки приложений на Java – Spring Framework, его актуальные компоненты и их возможности. Проанализированы основные преимущества применения Spring для разработчика по разработке J2EE-приложений.

Главное, что позволяет сделать Spring – упростить разработку J2EE приложений для разработчика. Поскольку J2EE-приложения в основном очень массивные и имеют долгий срок поддержки, то нужно было какое-то стандартизированное решение, которое позволяло бы перенести ответственность за инициализацию объектов и уменьшению связанности кода, что в будущем намного бы облегчало поддержку этого приложения, с разработчиков на фреймворк.

Spring Framework имеет несколько модулей, визуализацию модулей спринга можно увидеть на рисунке 1. В основе лежат модули основного контейнера, включая модель конфигурации и механизм внедрения зависимостей. Помимо этого, Spring Framework обеспечивает фундаментальную поддержку для различных архитектур приложений, включая обмен сообщениями, транзакционные данные и постоянство, а также веб. Он также включает веб-фреймворк Spring MVC на основе сервлетов и, параллельно, реактивный веб-фреймворк Spring WebFlux.

По своей сути Spring Framework представляет собой просто контейнер внедрения зависимостей, с несколькими удобными слоями (например, доступ к базе данных, прокси, аспектно-ориентированное программирование, RPC (Remote Procedure Call), веб-инфраструктура MVC (Model-View-Controller). Это все позволяет быстрее и удобнее создавать Java-приложения [1].

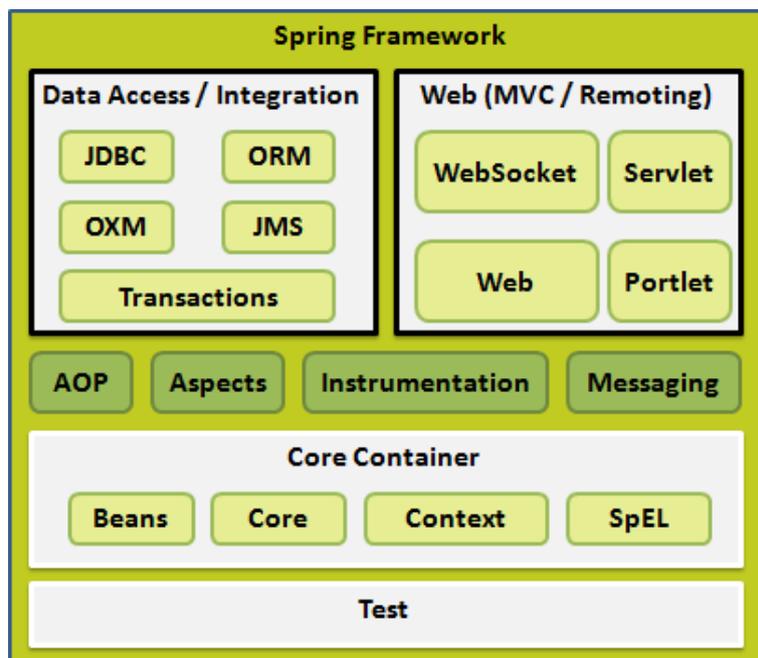


Рисунок 5. – Визуализация компонентов Spring

Далее рассмотрим основные принципы разработки, которые заложены в Spring Framework.

– Обеспечивать выбор на каждом уровне. Spring Framework позволяет откладывать дизайнерские решения как можно позже. Например, вы можете переключать поставщиков технологии программирования Object-Relational Mapping (ORM) через конфигурацию без изменения кода. То же самое верно и для многих других инфраструктурных проблем и интеграции со сторонними Application Programming Interface (API) [1].

– Учитывать разные точки зрения. Spring поддерживает гибкость и не имеет мнения о том, как все должно быть сделано. Он поддерживает широкий спектр приложений с разных точек зрения [1].

– Поддерживать обратную совместимость. Эволюция Spring Framework тщательно контролировалась, чтобы между версиями было мало критических изменений. Spring поддерживает тщательно подобранный диапазон версий Java Development Kit (JDK) и сторонних библиотек для облегчения обслуживания приложений и библиотек, зависящих от Spring [1].

– Заботится о дизайне API. Команда Spring вкладывает много времени и усилий в создание интуитивно понятных API-интерфейсов, которые выдерживают многие версии и многие годы [1].

– Установить высокие стандарты качества кода. Spring Framework уделяет большое внимание содержательной, актуальной и точной документации javadoc. Это один из очень немногих проектов, которые могут претендовать на чистую структуру кода без циклических зависимостей между пакетами [1].

Невозможно рассматривать спринг Spring Framework без понимания термина Dependency Injection (DI) – один из видов инверсии управления (Inversion of Control – IoC) [2].

При написании действительно крупных и сложных проектов разработчики сталкиваются с необходимостью делать классы приложения как можно более независимыми друг от друга для возможности повторного использования и юнит-тестирования. Именно DI устанавливает связи между этими классами, при этом сохраняя их независимость друг от друга.

Core Container (основной контейнер) включает в себя: Beans, Core, Context и SpEL (expression language). Beans отвечает за BeanFactory которая является сложной реализацией паттерна Фабрика (GoF) [1]. Бины – это объекты, которые являются основой приложения и управляются Spring IoC контейнером [1]. Эти объекты создаются с помощью конфигурационных метаданных, которые указываются в контейнере (например, XML- `<bean>...</bean>`). Модуль Core обеспечивает ключевые части фреймворка, включая свойства IoC и DI. Модуль Core можно найти на рисунке 1.

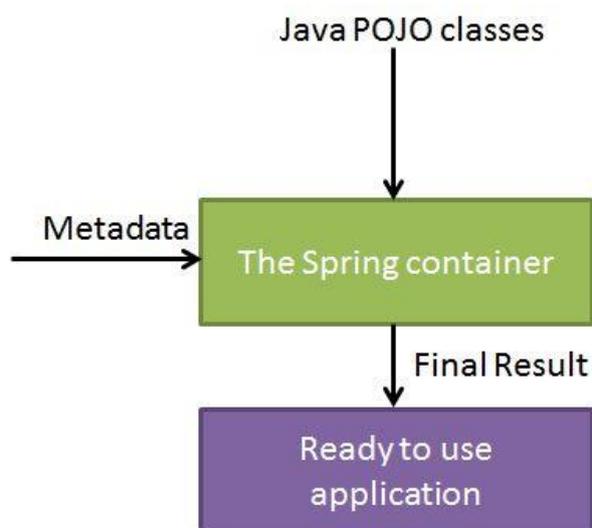


Рисунок 6. – Упрощенная работа спринга

Context построен на основе Beans и Core и позволяет получить доступ к любому объекту, который определён в настройках. Ключевым элементом модуля Context является интерфейс ApplicationContext. ApplicationContext является более сложным и более продвинутым Spring Container-ом. Так же, как BeanFactory, ApplicationContext загружает бины, связывает их вместе и конфигурирует их определённым образом. Но кроме этого, ApplicationContext обладает дополнительной функциональностью: распознавание текстовых сообщений из файлов настройки и отображение событий, которые происходят в приложении различными способами. Этот контейнер определяется интерфейсом `org.springframework.context.ApplicationContext`.

Модуль SpEL обеспечивает мощный язык выражений для манипулирования объектами во время исполнения [2].

Контейнер Data Access/Integration состоит из: Java DataBase Connectivity (JDBC), ORM, Object/XML Mapping (OXM), Java Message Service (JMS) и модуля Transactions.

JDBC – это стандартный API для независимого соединения языка программирования Java с различными базами данных, он обеспечивает абстрактный слой JDBC и избавляет разработчика от необходимости вручную прописывать монотонный код, связанный с соединением с БД.

ORM обеспечивает интеграцию с такими популярными ORM, как Hibernate, JDO, JPA и т.д. Модуль OXM отвечает за связь Объект/XML – XMLBeans, JAXB и т.д.

Модуль JMS (Java Messaging Service) – стандарт промежуточного программного обеспечения (ПО) для рассылки сообщений. Позволяет приложениям, выполненным на платформе Java EE, создавать, посылать, получать и читать сообщения [2].

Transactions поддерживает управление транзакциями для классов, которые реализуют определённые методы.

Веб слой состоит из Web, Web-MVC, Web-Socket, Web-Portlet. Модуль Web обеспечивает такие функции, как загрузка файлов и т. д. Web-MVC содержит реализацию Spring MVC для веб-приложений.

Web-Socket обеспечивает поддержку связи между клиентом и сервером, используя Web-Socket-ы в веб-приложениях.

Web-Portlet обеспечивает реализацию MVC в среде портлетов (элементов веб-страницы).

Spring также включает в себя ряд других важных модулей, таких как Aspect-oriented programming (AOP), Aspects, Instrumentation, Messaging и Test.

AOP реализует аспектно-ориентированное программирование (АОП) и позволяет использовать весь его арсенал возможностей.

Модуль Aspects обеспечивает интеграцию с AspectJ, которая также является мощным фреймворком АОП.

Instrumentation отвечает за поддержку Class Instrumentation и Class Loader, которые используются в серверных приложениях.

Модуль Messaging обеспечивает поддержку Simple (or Streaming) Text Oriented Message Protocol (STOMP).

Модуль Test обеспечивает тестирование с использованием TestNG или JUnit Framework.

Рассмотрев основные компоненты, можно сделать вывод, насколько Spring Framework упрощает разработку ПО, а именно берет на себя управление зависимостями и облегчает работу с компонентами. Поэтому планируется использовать Spring Framework при реализации проекта в бэкенд части, для написания легко поддерживаемого приложения с удобным взаимодействием с архитектурой REST и базой данных.

ЛИТЕРАТУРА

1. Spring Core Technologies. Общие сведения о спринге. [Электронный ресурс] / spring. – Режим доступа: <https://docs.spring.io> – Дата доступа: 20.09.2020.
2. Proselyte. Руководство по спринг. [Электронный ресурс] / proselyte. – Режим доступа: <https://proselyte.net/tutorials/spring-tutorial-full-version/ioc-containers> – Дата доступа: 20.09.2020.

УДК 004.021

**ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ
ДЛЯ МОБИЛЬНОГО ПРИЛОЖЕНИЯ «МУЗЫКАНТЫ РОССИИ»****Н.О. ШЕРШНЕВ***(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИНЫМ)*

В данной статье рассматривается современный способ проектирования базы данных для мобильного приложения «Музыканты России» с использованием документ-ориентированной базы данных MongoDB. Приводится пример концептуальной схемы базы данных.

Введение. База данных – это организованная структура, предназначенная для хранения, изменения и обработки взаимосвязанной информации. В базе данных может содержаться различная информация: личные данные пользователей, записи, даты, заказы, список клиентов и т.д. Очень важно организовать правильную структуру, которая позволит максимально эффективно использовать выделенные ресурсы. В данной статье рассматривается пример проектирования базы данных для мобильного приложения с использованием MongoDB.

MongoDB реализует абсолютно новый подход к построению баз данных, где нет таблиц, схем, запросов SQL, внешних ключей и многих других вещей, которые присущи объектно-реляционным базам данных.

В отличие от реляционных баз данных MongoDB предлагает документ-ориентированную модель данных, благодаря чему MongoDB работает быстрее, обладает лучшей масштабируемостью, ее легче использовать. Вся система MongoDB может представлять не только одну базу данных, находящуюся на одном физическом сервере. Функциональность MongoDB позволяет расположить несколько баз данных на нескольких физических серверах, и эти базы данных смогут легко обмениваться данными и сохранять целостность [1].

Основной раздел. Проект «Музыканты России» предназначен для облегчения коммуникации между музыкантами и участниками музыкальной индустрии, обмена полезной информацией, поиска единомышленников для совместного творчества. Проект является уникальным на территории СНГ, поэтому он нашел поддержку у множества спонсоров.

Основополагающей идеей мобильного приложения «Музыканты России» является предоставление пользователям подробной информации о музыкальных партнёрах по всей стране. Под музыкальными партнерами понимаются: концертные площадки и репетиционные базы, музыкальные лейблы и интернет-магазины, учебных базы, а также концертные агентства. Разработка собственной программа лояльности, которая подразумевает предоставления клиентам персональных бонусных карт со скидками на услуги музыкальных партнеров. Также пользователь всегда в курсе всех важных музыкальных событий.

Мобильное приложение «Музыканты России» обладает следующими функциональными особенностями:

- возможность обработки информации, поступающей в разрабатываемое приложение, что включает в себя обеспечения пользователя удобными средствами ввода, а также возможность редактирования, добавления и удаления информации;
- наличие визуальной среды для просмотра и обработки информации, важными характеристиками которых являются дизайн пользовательского интерфейса, интуитивность и удобство навигации;
- возможность получения информации о музыкальных магазинах, репетиционных базах и концертных площадках. Предоставление этой информации в виде списков и при помощи карт Google Maps;
- возможность получения последней актуальной информации о предстоящих мероприятиях в мире музыки;
- возможность выставления оценки для музыкального партнёра;
- возможность редактирования личной информации пользователя;
- возможность подачи частного объявления о поиске музыкантов или музыкальных коллективов для дальнейшего сотрудничества.

При проектировании базы данных необходимо установить все связи между сущностями, рассмотреть всю информационную систему в совокупности и определить взаимное влияние сущностей.

Документ-ориентированная система управления базами данных MongoDB использует JSON-подобные документы для описания используемых коллекций. Описание хранимых коллекций в базе данных представлено ниже в таблице 1.

Таблица 1. – Перечень коллекций базы данных

№ п/п	Название коллекции	Описание
1	Partners	Список музыкальных партнеров
2	Users	Список пользователей
3	Vacancies	Список вакансий
4	Partnerships	Список предложений о сотрудничестве

Для обеспечения целостности данных мы создаем дополнительные ключи для каждой коллекции. Это позволит нам в дальнейшем написать запросы к базе данных, используя данные, которые находятся в нескольких коллекциях. Идентификаторы и ссылки на другие коллекции представлены в таблице 2.

Таблица 2. Первичные и внешние ключи отношений

№ п/п	Название коллекции	Идентификатор	Ссылки на другие коллекции
1	Partners	partnerId	userId
2	Users	userId	partnerId
3	Vacancies	vacancyId	userId
4	Partnerships	partnershipId	userId

Концептуальная схема базы данных представлена на рисунке 1.

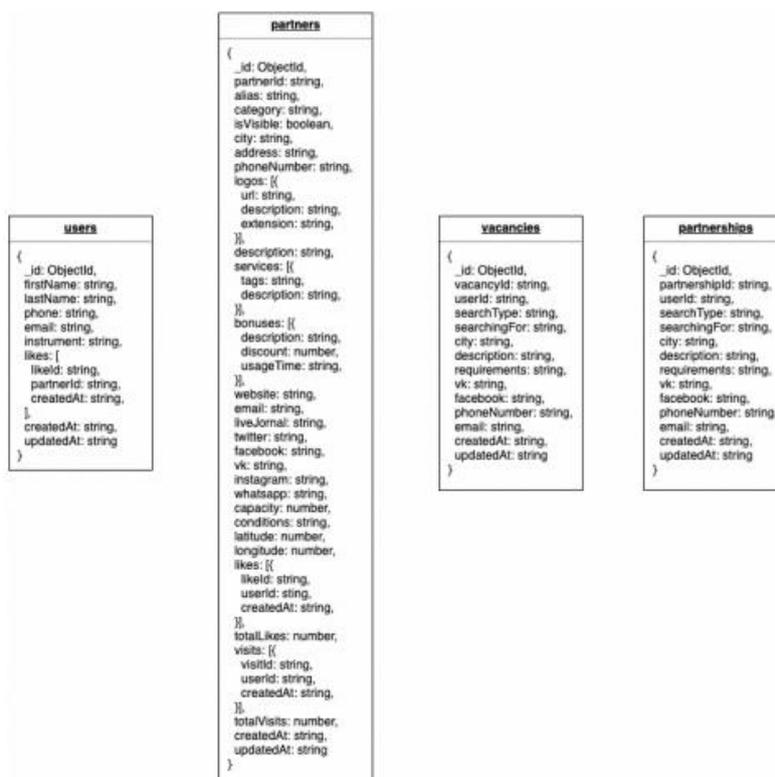


Рисунок 1. – Концептуальная схема базы данных

Можно выделить следующие преимущества использования MongoDB:

- динамическая схема. СУБД позволяет гибко работать со схемой базы данных без необходимости изменять сами данные;
- масштабируемость. MongoDB горизонтально масштабируема, что позволяет уменьшить нагрузку на сервера при больших объемах данных;
- скорость. Высокая производительность при выполнении простых запросов;

- гибкость. В MongoDB можно без вреда для существующих данных, их структуры и производительности СУБД добавлять поля и новые коллекции [2].

Заключение. В данной статье были рассмотрены основные особенности документ-ориентированной базы данных MongoDB. Выделены основные сущности модели данных для мобильного приложения «Музыканты России». Спроектирована концептуальная схема базы данных.

ЛИТЕРАТУРА

1. Трогер [Электронный ресурс] / SQL против NoSQL на примере MySQL и MongoDB. – Режим доступа: <https://tproger.ru/translations/sql-vs-nosql>. – Дата доступа: 24.09.2020.
2. Metanit.com – сайт о программировании [Электронный ресурс] / Введение в MongoDB. – Режим доступа: <https://metanit.com/nosql/mongodb/1.1.php>. – Дата доступа: 24.09.2020.

УДК 004.021

ИСПОЛЬЗОВАНИЕ СЕРВИСА УВЕДОМЛЕНИЙ PUSHWOOSH В МОБИЛЬНОМ ПРИЛОЖЕНИИ «МУЗЫКАНТЫ РОССИИ»

Н.О. ШЕРШНЕВ

(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИНЫМ)

В данной статье рассматривается современный способ обработки всплывающих уведомлений в мобильном приложении. Приведен подробный пример интеграции сервиса PushWoosh в мобильное приложение.

Введение. Push-уведомления – это краткие всплывающие оповещения, которые появляются на экране мобильного телефона или компьютера и сообщают о важных событиях и обновлениях. При эффективном использовании эти краткие информативные сообщения являются мощным маркетинговым инструментом. Основной целью push-уведомлений является информирование пользователей об обновлениях веб-сайтов или приложений, добавлении нового контента, либо о каких-либо других новостях [1].

Согласно статистике, приведенной всемирно известной компанией Localytics, которая имеет огромный опыт работы в аналитике и маркетинге приложений, можно отметить следующее:

- push-уведомления повышают вовлеченность пользователей на 88%;
- при включенной функции уведомлений 65% пользователей возвращается в приложение в течение 30 дней;
- более 50% пользователей подключают push-уведомления и считают эти сообщения полезными;
- push-уведомления увеличивают количество запусков приложения на 27% [1].

Приведенные статистические данные дают четко понять, что для успешного продвижения приложения необходимо использовать систему оповещений. По этой причине push-уведомления нашли широкое применение в разработке мобильных приложений.

Основной раздел. PushWoosh является отличным бесплатным решением для реализации системы оповещения пользователей. Данный сервис является кроссплатформенным, что позволяет использовать его одновременно более чем на 20 различных платформах.

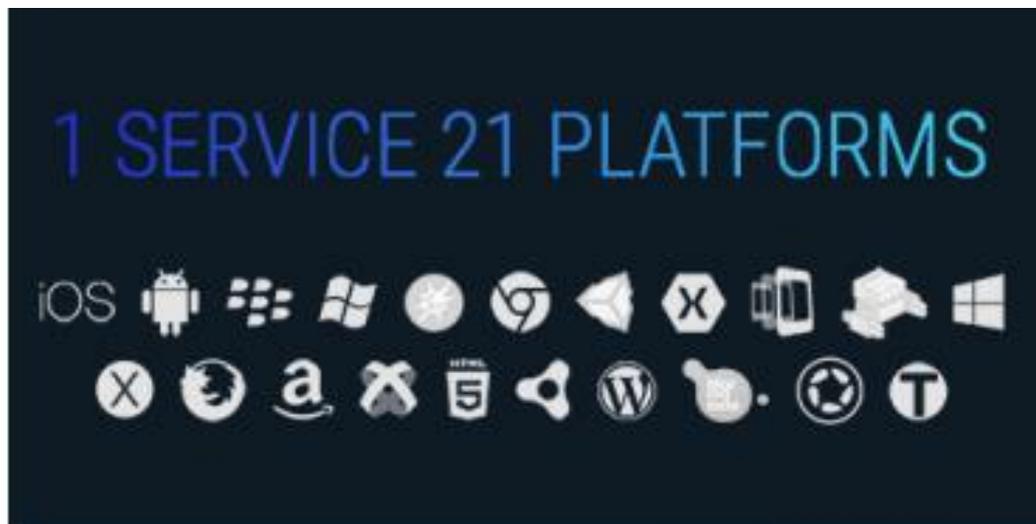


Рисунок 1. – Платформы для работы с сервисом PushWoosh

PushWoosh имеет много отличительных особенностей, которые значительно выделяют этот сервис на фоне своих конкурентов. Среди наиболее значимых можно отметить следующие:

- private Cloud. Каждому пользователю предоставляется защищенное облачное хранилище данных. Сервис PushWoosh внимательно следит за конфиденциальностью ваших личных данных;
- high Speed. Сервис работает с невероятно быстрой скоростью. Каждый день PushWoosh отправляет более 250 миллионов push-уведомлений;
- multi-language. PushWoosh всегда знает, какой язык установлен на устройстве, которое подписано на обновления. Это позволяет создавать уникальный набор оповещающих сообщений, адаптированных под каждый язык, что с легкостью позволяет преодолевать языковой барьер;

- geo-zones. Данной функцией сервиса PushWoosh можно воспользоваться на устройствах под управлением операционной системы IOS, Android или Windows Phone. Она позволяет создавать области на карте, с точностью до 50 метров, чтобы отсылать уведомления пользователям в конкретно заданном месте;
- timezone Sensitive. Сервис PushWoosh предоставляет возможность отсылать уведомления пользователю в соответствии с его часовым поясом. Это позволяет сделать ваши push-сообщения своевременными для каждого конкретного пользователя и, как результат, более привлекательными [2].

Сервис PushWoosh отлично себя зарекомендовал и является лучшим решением для работы с push-уведомлениями. По этой причине, при создании мобильного приложения «Музыканты России», было принято решение об использовании именно данного сервиса. Далее будет рассмотрен пример использования данного сервиса в мобильном приложении, написанном на языке программирования Java. Процесс интеграции данного сервиса достаточно прост, на официальном сайте можно найти подробную инструкцию о том, как использовать PushWoosh на любой платформе. Важную роль в приложении «Музыканты России» играют Push-уведомления. Ведь именно они позволяют вовремя узнать о скидках и акциях в музыкальных магазинах, об изменениях в расписании посещения репетиционных баз и музыкальных лейблов, о приближающихся важных событиях в мире музыки. В основе работы системы уведомлений лежит сервис PushWoosh. Далее будет приведен пример, как именно можно использовать данный сервис в своем приложении.

В первую очередь необходимо наследовать MainActivity от FragmentActivity class и реализовать интерфейс PushEventListener (см. листинг 1).

Листинг 1 – Реализация интерфейса PushEventListener

```
public class MainActivity extends FragmentActivity implements PushEventListener
```

В методе onCreate вызываем метод PushFragment.init(this); (см. листинг 2).

Листинг 2 – Инициализация PushFragment

```
@Override protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    PushFragment.init(this);
}
```

Далее создаем метод onNewIntent и вызываем функцию PushFragment.onNewIntent (this, intent); (см. листинг 3).

Листинг 3. – Создание метода onNewIntent

```
@Override public void onNewIntent(Intent intent) {
    super.onNewIntent(intent); // check if we have a new intent with push notification
    PushFragment.onNewIntent(this, intent);
}
```

После этого необходимо реализовать методы PushEventListener интерфейса. Самым важным для нас будет метод doOnMessageReceive (String message), который срабатывает, когда мы получаем новое уведомление от сервиса PushWoosh. В данном методе реализована обработка и вывод всплывающего уведомления на экран мобильного устройства (см. листинг 4).

Листинг 4 – Обработка и вывод уведомления на экран пользователя

```
@Override public void doOnMessageReceive(String message) {
    String show = parsePushNotification(message); // сообщение приходит в формате JSON, поэтому
    его необходимо распарсить
    NotificationCompat.Builder builder = new NotificationCompat.Builder(getApplicationContext()) //
    создаем уведомление
        .setSmallIcon(R.drawable.info) // устанавливаем небольшую иконку для нашего сообщения
        .setContentTitle("Музыканты России") // задаем заголовок
        .setContentText(show) // помещаем текст уведомления
        .setVibrate(new long[] {10, 1000}) // добавляем вибрацию к нашему оповещению
        .setColor(Color.argb(255,45,192,233))
```

```
.setAutoCancel(true); // сообщение автоматически исчезает из меню уведомлений, когда пользо-
ватель к нему прикасается Intent
intent = new Intent(getApplicationContext(), MainActivity.class); // создаем новый экземпляр Intent
PendingIntent pendingIntent = PendingIntent.getActivity(getApplicationContext(), 0, intent, 0); // предо-
ставляем право сервису PushWoosh для выполнения фрагмента кода
Notification notificationManager = (NotificationManager) getSystemService(Context.NOTIFICATION_
SERVICE); // Notification Manager – системный сервис Ан-
дроид, который управляет всеми уведомлениями builder.setContentIntent(pendingIntent); // предо-
ставляем ответ, который будет получен при нажатии на уведомление
Notification notification = builder.build(); // объединяем все свойства, которые были заданы для
уведомления и возвращаем экземпляр класса Notification
notificationManager.notify(1, notification); // объект notification передается в систему путем вы-
зова метода notificationManager.notify()
}
```

Как можно легко заметить, процесс обработки push-уведомлений в мобильном приложении «Музыканты России» достаточно прост и не требует каких-либо дополнительных знаний. Все что вам необходимо, это внимательно ознакомиться с инструкцией, которая представлена на официальном сайте, и выполнить все необходимые действия шаг за шагом. Такой простой способ интеграции push-уведомлений заметно выделяет сервис PushWoosh, который является несомненно одним из лучших современных решений для реализации push-уведомлений в своем приложении.

Заключение. В данной статье был представлен современный способ работы с всплывающими уведомлениями на примере мобильного приложения «Музыканты России». Были представлены основные преимущества и функциональные возможности сервиса уведомлений PushWoosh, а также разобран пример интеграции данного сервиса в мобильное приложение.

ЛИТЕРАТУРА

1. Блог App Global [Электронный ресурс] / Push-уведомления: взрывная статистика. – Режим доступа: <http://app-global.ru/blog/push-uedomleniya-vzrivnaya-statistika>. – Дата доступа: 24.09.2020.
2. PushWoosh [Электронный ресурс] / Features. – Режим доступа: [https:// www.pushwoosh.com/features](https://www.pushwoosh.com/features). – Дата доступа: 24.09.2020.

УДК 004.055

**ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ТОРГОВОЙ ПЛОЩАДКИ ДЛЯ ПРОДАЖИ ПРОДУКТОВ****И.А. ШУМЯНЦЕВ***(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИН)*

В статье представлен практический способ создания интерфейса для торговой площадки для продажи продуктов. Были поставлены основные цели ресурса, задачи и основы построения интерактивного прототипа интерфейса.

Введение. Торговые площадки в интернете – современная альтернатива онлайн-магазинам. Они удобны как для покупателей, так и для продавцов. Первым нравится возможность сравнить цены и условия продажи. Ну, а для продавцов площадки являются относительно недорогим и эффективным способом продвижения товаров.

По своей сути электронные торговые площадки (ЭТП) – это сайт в интернете, который позволяет взаимодействовать заказчиком и поставщиком. Первые с помощью площадки создают и проводят конкурентные закупки. Вторые в них участвуют.

Заказчики используют ЭТП, чтобы сделать процесс выбора поставщиков прозрачным. Ведь победитель всегда выбирается по заранее известным критериям, поэтому исключается человеческий фактор. Недобросовестный сотрудник не может выбрать своего поставщика в обход конкурентной процедуры.

Для поставщиков площадки полезны тем, что на них можно найти подходящую закупку и поучаствовать в ней удалённо. Не надо искать клиента по традиционным каналам и готовить пачки бумажных документов для очного аукциона.

Выбрав необходимые товары или услуги, пользователь имеет возможность выбрать метод оплаты и доставки. Также пользователь может оценить товар, тем самым повлияв на рейтинг продавца. Пользователю также доступна возможность оставлять отзывы о товаре. Совокупность отобранных товаров, способ оплаты и доставки представляют собой законченный заказ, который оформляется на сайте путём сообщения необходимой информации о покупателе. Информация о заказе и покупателе должна храниться в базе данных площадки, так как подобные сервисы должны быть рассчитаны на повторные заказы со стороны покупателей. Так же на торговой площадке должна быть реализована возможность оставлять «пожелания к заказу» для продавца, это необходимо если покупатель захочет сообщить какую-либо дополнительную информацию для продавца

Торговая площадка должна представлять из себя сайт в интернете, на которой будут продаваться продукты от различных продавцов, предоставлять пользователь удобный и понятный интерфейс для обеспечения минимального порога вхождения.

Цель торговой площадки – познакомить посетителей с наибольшим количеством продавцов или магазинов, направлениями их деятельности, товарами и ценами на них. Очень важно привлекать и продавцов на торговую площадку, это создаст конкурентоспособный рынок и расширит диапазон товаров. Также к основным целям относятся:

- привлечение максимального количества покупателей и продавцов;
- экономия времени покупателей и продавцов;
- наличие актуальной информации по товарам;
- предоставление точной и актуальной статистики о продавце;
- быстрая и безопасная организация сделок между покупателем и продавцом.

Проектируемый сайт должен удовлетворять следующим требованиям:

- наличие простого и удобного пользовательского интерфейса;
- наличие корзины;
- наличие окна обратной связи;
- наличие широких функциональных возможностей, а именно: регистрация на сайте, возможность оставлять комментарии под товаром, возможность оценивать продавца, делать заказ с онлайн-оплатой через различные платёжные системы, изменять (добавлять, удалять, редактировать) список товаров и их характеристики продавцом.

Построение интерактивного прототипа интерфейса. Прототип – это модель, позволяющая продемонстрировать интерфейс, поведение проектируемой системы или архитектуру [1]. Он отображает структуру будущего сайта, а также его отдельных страниц вне дизайнерских элементов, цветов и изображений.

Реализация интерфейса торговой площадки будет состоять из следующих элементов:

- header – это блок в верхней части страницы сайта, который виден на всех страницах сайта. Как

правило, содержит основной функционал сайта: вкладки основных разделов, поиск, выдвигаемое меню с категориями, корзина.

– main – это все информационное содержание ресурса: информативные блоки, видео, изображения или любой другой контент сайта.

– footer – это блок в нижней части страницы сайта. Используется в основном как блок дополнительной информации (номер телефона, почта, часы работы организации, ссылки на полезные ресурсы).

Данная торговая площадка будет состоять из 5 блоков:

– в первом блоке расположена шапка сайта, где расположены навигационные кнопки (главная, о нас, доставка, помощь, переход на оптовою/розничную площадку), которые позволяют пользователю, перемещаться по сайту, так же в данном блоке находится логотип;

– во втором блоке находится каталог товаров и поиск, где пользователь может выбрать нужную ему категорию и далее сам товар;

– в третьем блоке находятся кнопки сортировки товаров (по популярности, по цене, по городу);

– в четвертом блоке расположены сами товары (по 8 штук на странице), для перехода к следующему товаром в самом низу четвертого блока имеются кнопки пагинации ;

– в пятом блоке находится информационно-справочная информация.

Рассмотрим макет пользовательской части ресурса. На рисунке 1 представлен макет пользовательской части ресурса.



Рисунок 1. – Макет торговой площадки для продажи продуктов пользовательской части

Также рассмотрим макет административной части сайта (рисунок 2).

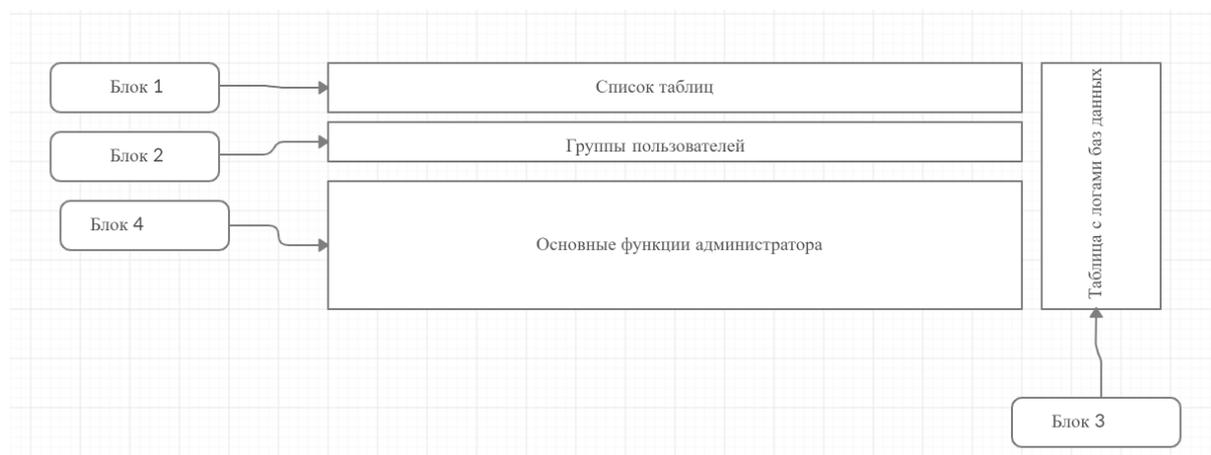


Рисунок 2. – Макет торговой площадки для продажи продуктов административной части

Данная страница будет состоять из четырёх блоков:

- в первом блоке расположен список таблиц из всех баз данных (кроме вложенных);
- во втором блоке расположены группы пользователей (администраторов), где главный администратор может задавать определённые права;
- в третьем блоке находится отчёт о всех совершённых действиях с базами данных, где администратор может отменить какое-либо из действий;
- В четвёртом блоке находятся все функции, предоставленные администратору сайта (управление товарами, управление категориями, управление заказами и т.д.).

Рассмотрим основные функции, реализованные в пользовательском интерфейсе.

Функция «sort», предоставляет возможность пользователю сортировать товары по заданным условиям (листинг 1).

Листинг 1 – Функция «sort».

```
1: function ready() {
2: if (localStorage.getItem('rating') == '-rating') {
3: document.getElementById('sortRate').innerHTML = 'По популярности &#9660;';
4: document.getElementById('sortPrice').innerHTML = 'По цене';
5: document.getElementById('sortRate').className = 'btn-info btn btn-sm m-0 p-2';
6: document.getElementById('sortPrice').className = 'btn-primary btn btn-sm m-0 p-2';
7: document.getElementById('sortCity').className = 'btn-primary btn btn-sm m-0 p-2';
8: }
9: else if (localStorage.getItem('rating') == 'rating') {
10: document.getElementById('sortRate').innerHTML = 'По популярности &#9650;';
11: document.getElementById('sortPrice').innerHTML = 'По цене';
12: document.getElementById('sortRate').className = 'btn-info btn btn-sm m-0 p-2';
13: document.getElementById('sortPrice').className = 'btn-primary btn btn-sm m-0 p-2';
14: document.getElementById('sortCity').className = 'btn-primary btn btn-sm m-0 p-2';
15: }
16: else if (localStorage.getItem('price') == '-price') {
17: document.getElementById('sortRate').innerHTML = 'По популярности';
18: document.getElementById('sortPrice').innerHTML = 'По цене &#9660;';
19: document.getElementById('sortRate').className = 'btn-primary btn btn-sm m-0 p-2';
20: document.getElementById('sortPrice').className = 'btn-info btn btn-sm m-0 p-2';
21: document.getElementById('sortCity').className = 'btn-primary btn btn-sm m-0 p-2';
22: }
23: else if (localStorage.getItem('price') == 'price') {
24: document.getElementById('sortRate').innerHTML = 'По популярности';
25: document.getElementById('sortPrice').innerHTML = 'По цене &#9650;';
26: document.getElementById('sortRate').className = 'btn-primary btn btn-sm m-0 p-2';
27: document.getElementById('sortPrice').className = 'btn-info btn btn-sm m-0 p-2';
28: document.getElementById('sortCity').className = 'btn-primary btn btn-sm m-0 p-2';
29: }
30: else {
31: document.getElementById('sortRate').className = 'btn-primary btn btn-sm m-0 p-2';
32: document.getElementById('sortPrice').className = 'btn-primary btn btn-sm m-0 p-2';
33: document.getElementById('sortCity').className = 'btn-info btn btn-sm m-0 p-2';
34: }
35: }
36: const sendAjax = (sort) => {
37: $.ajax({
38: type: "POST",
39: url: "ajax/add/",
40: dataType: "html",
41: data: { "item": sort },
42: success: function (data) {
43: location.reload();
44: }
45: });
46: }
```

```
47: function setActive(value) {
48:   if (value == 1) {
49:     localStorage.setItem('price', null);
50:     localStorage.setItem('rating', null);
51:     let sort = localStorage.getItem('city');
52:     sendAjax(sort);
53:   }
54:   else if (value == 2) {
55:     if (localStorage.getItem('rating') == null ) localStorage.setItem('rating', '-rating');
56:     let sort = localStorage.getItem('rating');
57:     if (sort == '-rating') {
58:       sort = 'rating';
59:       localStorage.setItem('rating', sort);
60:       localStorage.setItem('price', null);
61:     }
62:     else {
63:       sort = '-rating';
64:       localStorage.setItem('rating', sort);
65:       localStorage.setItem('price', null);
66:     }
67:     sendAjax(sort);
68:   }
69:   else {
70:     if (localStorage.getItem('price') == null ) localStorage.setItem('price', '-price');
71:     let sort = localStorage.getItem('price');
72:     if (sort == '-price') {
73:       sort = 'price';
74:       localStorage.setItem('price', sort);
75:       localStorage.setItem('rating', null);
76:     }
77:     else {
78:       sort = '-price';
79:       localStorage.setItem('price', sort);
80:       localStorage.setItem('rating', null);
81:     }
82:     sendAjax(sort);
83:   }
84: }
85: document.addEventListener("DOMContentLoaded", ready);
```

Заключение. В данной статье рассмотрены основные задачи и требования. Были сформулированы необходимые разделы программного средства, для каждого из которых был спроектирован прототип, на основе которого можно создать графическое изображение торговой площадки, а потом и сам веб-сервис.

ЛИТЕРАТУРА

1. Статья о использовании построении сайта [Электронный ресурс] / Прототипирование - как метод сбора требований о ПО. – Режим доступа: <https://clck.ru/R8MKr>. – Дата доступа: 22.09.2020.

УДК 004.051

ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА ТОРГОВОЙ ПЛОЩАДКИ ДЛЯ ПРОДАЖИ ПРОДУКТОВ

И.А. ШУМЯНЦЕВ*(Представлено: канд. техн. наук, доц. А.Ф. ОСЬКИН)*

В статье представлен практический способ проектирования программного продукта, было описано проектирование диаграммы вариантов использования и проектирование логической структуры базы данных.

Введение. Проектирование программного обеспечения — процесс создания проекта программного обеспечения (ПО), а также дисциплина, изучающая методы проектирования.

Проектирование ПО является частным случаем проектирования продуктов и процессов.

Целью проектирования является определение внутренних свойств системы и детализации её внешних (видимых) свойств на основе выданных заказчиком требований к ПО (исходные условия задачи). Эти требования подвергаются анализу. Проектирование ПО включает следующие основные виды деятельности:

- выбор метода и стратегии решения;
- выбор представления внутренних данных;
- разработка основного алгоритма;
- документирование ПО;
- тестирование и подбор тестов;
- выбор представления входных данных.

Первоначально программа рассматривается как чёрный ящик. Ход процесса проектирования и его результаты зависят не только от состава требований, но и выбранной модели процесса, опыта проектировщика [1].

Модель предметной области накладывает ограничения на бизнес-логику и структуры данных.

В зависимости от класса, создаваемого ПО, процесс проектирования может обеспечиваться как «ручным» проектированием, так и различными средствами его автоматизации. В процессе проектирования ПО для выражения его характеристик используются различные нотации — блок-схемы, ER-диаграммы, UML-диаграммы, DFD-диаграммы, а также макеты.

Проектированию обычно подлежат:

- архитектура ПО;
- устройство компонентов ПО.

Основной раздел.

Диаграммы вариантов использования описывают взаимоотношения и зависимости между группами вариантов использования и действующих лиц, участвующими в процессе.

Важно понимать, что диаграммы вариантов использования не предназначены для отображения проекта и не могут описывать внутреннее устройство системы. Диаграммы вариантов использования предназначены для упрощения взаимодействия с будущими пользователями системы, с клиентами, и особенно пригодятся для определения необходимых характеристик системы. Другими словами, диаграммы вариантов использования говорят о том, что система должна делать, не указывая сами применяемые методы [2]. Варианты использования описаны в таблице 1.

Таблица 1 – Варианты использования приложения

Варианты использования	Описание
Администратор	Главный пользователь ресурса, основная функция - работа с БД
Работа с пользователями	Изменение, удаление и добавление данных пользователя
Работа с данными	Удаление, изменение и добавление данных о товаре и поставщиках
Авторизация	Процесс входа в учётную запись или создание новое
Пользователь	Пользователь выполняющий основные функции на сайте
Добавление товара в корзину	Помещение товара (нескольких товаров) во временное хранилище данных для совершения покупки в «один клик»
Заказ товара	Процесс оформления доставки и оплаты
Поиск товара	Процесс при котором пользователь может найти необходимый ему товар на сайте
Оценка товара	Пользователь может оставить комментарий к товару, а также поставить ему оценку

На основании выделенных актёров и вариантов использования построена диаграмма вариантов использования, представленная на рисунке 1.

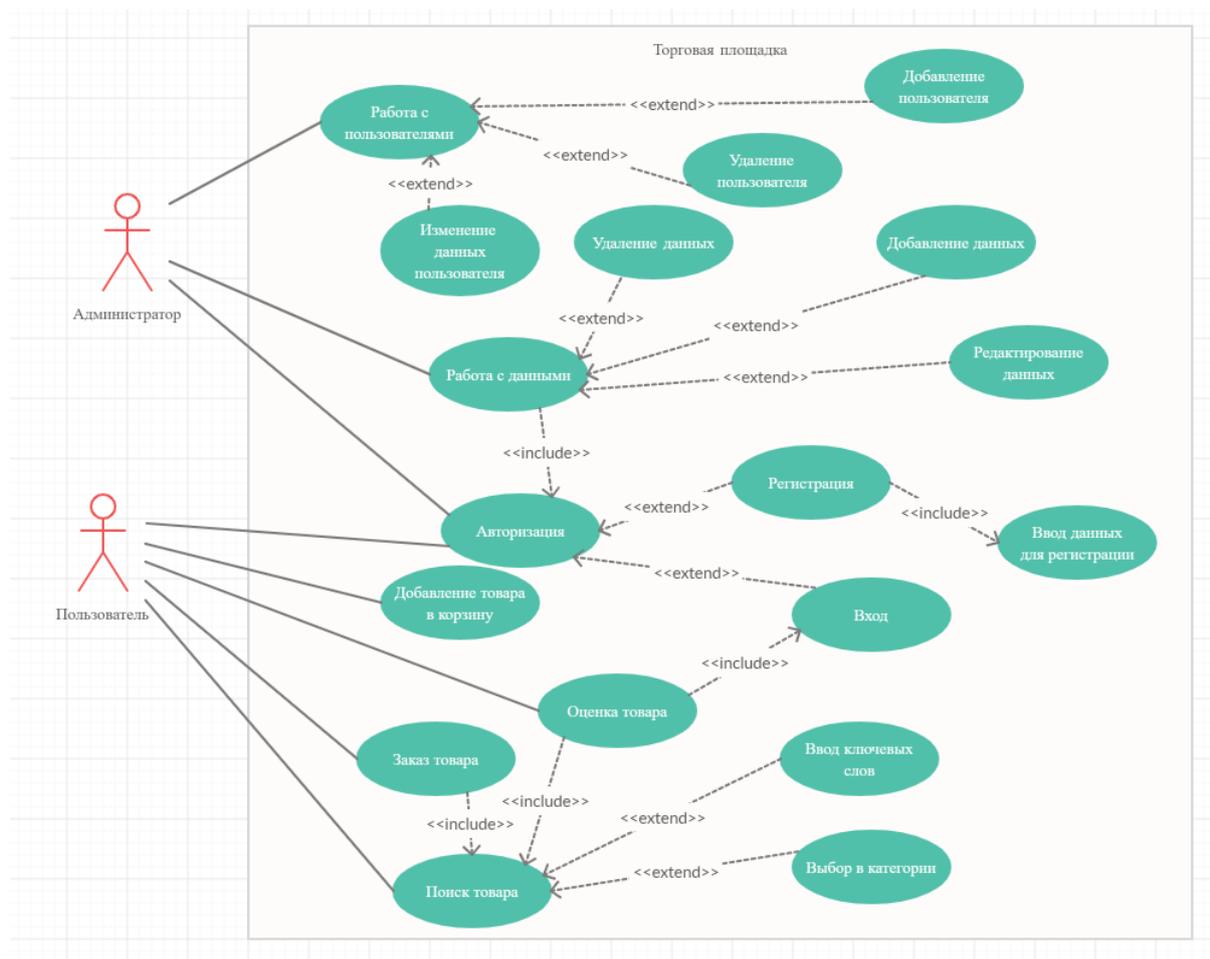


Рисунок 1. – Диаграмма вариантов использования

Проектирование логической структуры базы данных.

Исходя из предметной области торговой площадки, можно выделить 4 основные сущности:

- Пользователь [User] – представляет информацию о каждом пользователе.
- Продукт [Product] – представляет информацию о товарах, имеющихся в наличии.
- Заказ [Order] – представляет информацию о всех заказах пользователей.
- Комментарий [Comment] - предоставляет информацию о комментариях пользователя

Для каждой сущности необходимо определить атрибуты. Сущности, их атрибуты и характеристики атрибутов приведены в таблице 2.

Таблица 2 – Перечень сущностей и их атрибутов

Сущность	Атрибут	Описание атрибута
Пользователь	id	Уникальный идентификационный номер
	Логин	Логин пользователя для входа в систему
	Пароль	Пароль пользователя для входа в систему
Продукт	id	Уникальный идентификационный номер
	Название	Название товара
	Категория	Категория товара

1	2	3
Продукт	Цена	Текущая цена одного товара
	Поставщик	Поставщик товара
	Описание	Краткое описание товара
	Поставщик	Поставщик товара
Заказ	id	Идентификационный номер заказа
	Дата	Дата подтверждения заказа пользователем
	Содержимое	Содержимое заказа
	Выполнение	Индикатор выполнения заказа
	Пользователь	Логин пользователя, сделавшего заказ
	Сумма	Сумма заказа
Комментарий	id	Уникальный идентификационный номер
	Пользователь	Логин пользователя, оставивший комментарий
	Содержание	Содержание комментария
	Продукт	Ключ продукта, к которому оставлен комментарий

Заключение. В данном разделе был приведён практический способ проектирования программного продукта, было описано проектирование диаграммы вариантов использования и проектирование логической структуры базы данных

ЛИТЕРАТУРА

1. Проектирование программного обеспечения. [Электронный ресурс] – Режим доступа: <https://kartaslov.ru/карта-знаний/Проектирование+программного+обеспечения>. – Дата доступа: 13.08.2020.
2. Диаграмма вариантов использования [Электронный ресурс] – Режим доступа: <https://habr.com/ru/post/47940>. – Дата доступа 10.09.2020.

МАШИНОСТРОЕНИЕ

УДК 621.91.01

ВЛИЯНИЕ УГЛА НАКЛОНА РЕЖУЩЕЙ КРОМКИ ИНСТРУМЕНТА НА ХАРАКТЕРИСТИКИ ОБРАБОТКИ ПЛОСКОЙ ПОВЕРХНОСТИ ДЕТАЛИ НА СТАНКЕ С ЧПУ

В.С.АНИСИМОВ, Н.В.КУРАШ, Г.И.ШВЕЦОВ
(Представлено: д-р техн. наук, Н.Н. ПОПОК)

Приведены схемы обработки плоской поверхности вращающейся детали вращающимся инструментом, рассмотрены основные векторы линейных скоростей движения, возможные положения режущей кромки инструмента и проанализировано их влияние на процесс резания.

Введение. Современные металлорежущие станки с ЧПУ позволяют реализовывать трехосевую обработку поверхностей деталей с различными сложными траекториями резания. При реализации таких траекторий рекомендуется использовать режущий инструмент с крутоспиральными режущими кромками, который обеспечивает безударное касательное движение резания. Преимущества применения касательного движения и крутоспирального режущего инструмента ранее исследовались и описывались ведущими отечественными учеными Г.И. Грановским, Ю.А. Новоселовым, В.В. Шварцем, В.Ф. Бобровым и другими [1,2]. Однако результаты их исследований не были реализованы в полной мере из-за отсутствия компактных высокоскоростных и высокомоментных приводов.

Основная часть. Для определения влияния угла наклона режущей кромки инструмента и ее касательного движения резания на параметры (скорость резания, подача) и характеристики (сила резания, усадка стружки) процесса резания рассмотрим схему обработки плоской поверхности детали на токарно-фрезерном станке с ЧПУ (рисунок 1) [3].

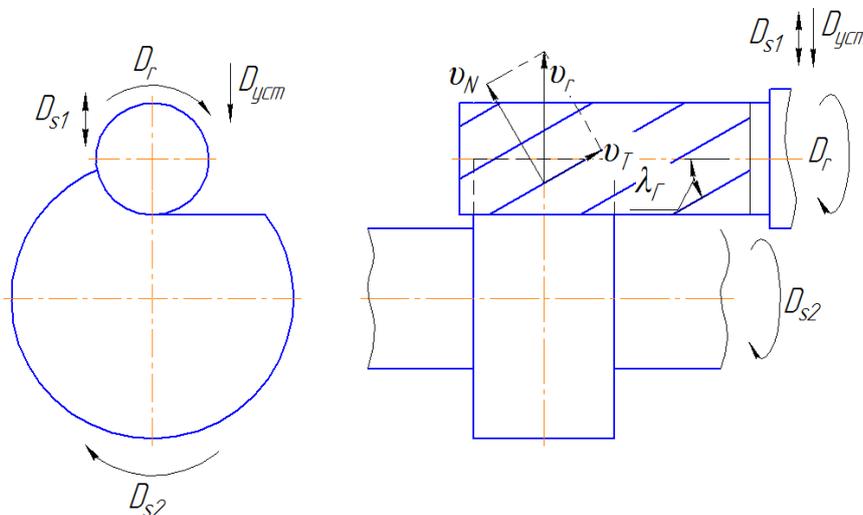


Рисунок 1. – Схемы обработки плоской поверхности детали на токарно-фрезерном станке с ЧПУ

При такой обработке поверхности угол наклона режущей кромки λ может рассматриваться в инструментальной, статической и кинематической системах координат. В инструментальной системе координат угол λ_r является углом заточки режущей кромки инструмента, который определяется между режущей кромкой и отсчетной осью или поверхностью инструмента. Например, для концевой фрезы угол λ_r – это угол между режущей кромкой и геометрической осью (рисунок 2, а).

В статической системе координат угол λ_c является углом установки инструмента на станке только при учете главного движения резания и за отсчетную плоскость принимается основная плоскость, перпендикулярная вектору линейной скорости главного движения. Например, для концевой фрезы λ_c – это угол в статической плоскости резания между режущей кромкой и основной плоскостью $P_{v_c}-P_{v_c}$ (рисунок 2, б).

В кинематической системе координат учитываются все элементарные движения резания и отсчетную основную плоскость проводят перпендикулярно вектору скорости резания. Например, для концевой фрезы угол λ_K – это угол между режущей кромкой и кинематической основной плоскостью Pv_K – Pv_K (рисунок 2, в).

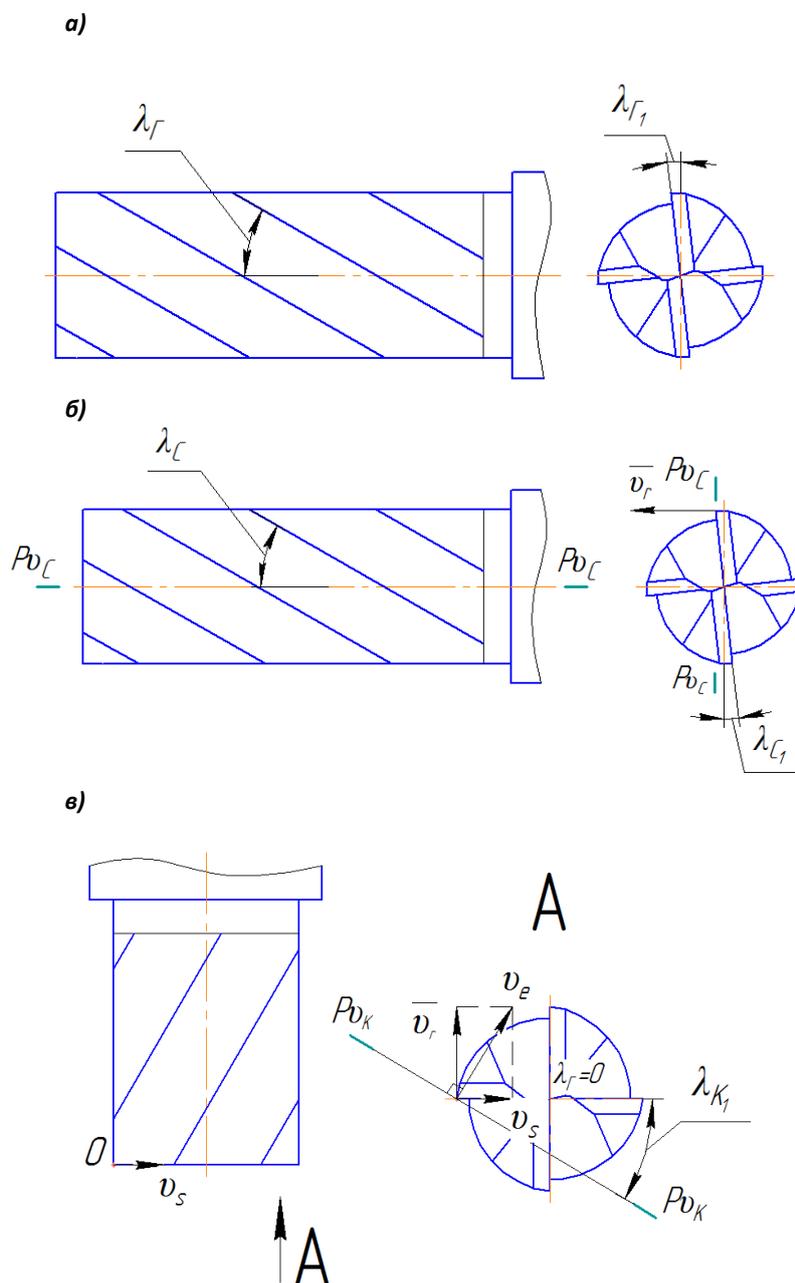


Рисунок 2. – Схема для определения угла наклона режущей кромки инструмента в инструментальной (а), статической (б) и кинематической (в) системах координат

При обработке плоской поверхности вращающейся детали вращающимся инструментом (рисунок 3, а) возможные варианты расположения режущей кромки на цилиндрической и торцовой поверхностях фрезы представлены на рисунках 3, б и 3, в.

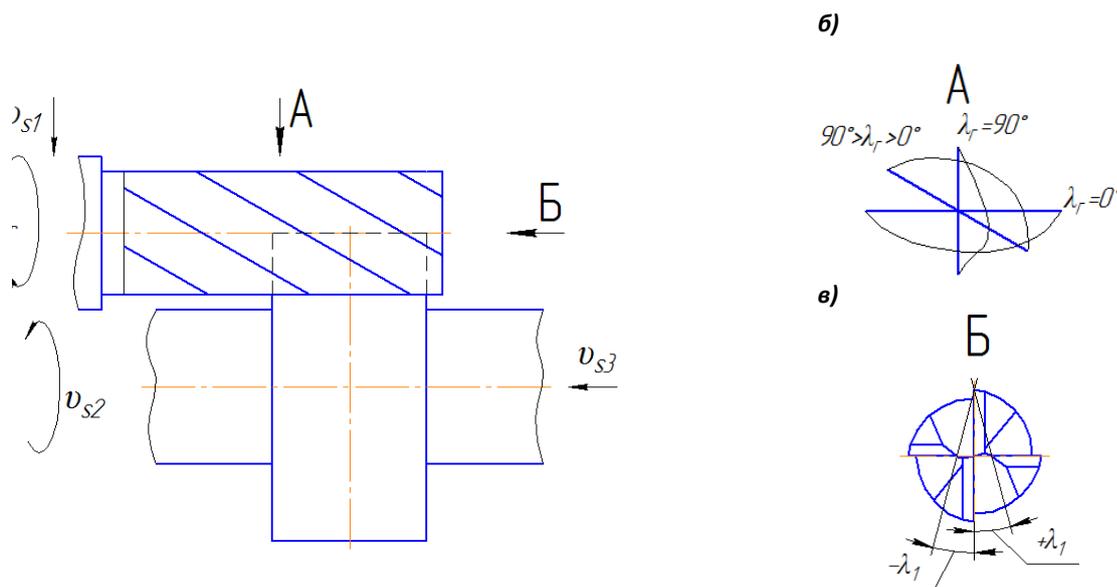


Рисунок 3. – Схема обработки плоской поверхности вращающейся детали вращающимся инструментом (а) и возможные углы наклона режущей кромки на цилиндрической (б) и торцевой (в) поверхностях концевой фрезы

Если $\lambda_r = 0^\circ$ (рисунок 3, в), то режущая кромка инструмента контактирует по максимальной ширине b , что приводит к росту сил и мощности резания. При $0^\circ < \lambda_r \leq 90^\circ$ резание возможно при подаче инструмента v_{s3} вдоль оси главного вращательного движения. В этом случае реализуется ротационное и плунжерное резание, для которых характерно снижение мощности резания за счет касательного движения.

При касательном движении важно соотношение линейных скоростей вращения инструмента и заготовки (передаточное отношение i_k). Если реализуется кинематическая схема резания точения, то $\frac{v_H}{v_3} \leq 1$

и оптимальным считается значение кинематического коэффициента $i_k = 0,5 \dots 0,6$, что соответствует изменению $60^\circ > \lambda_k > 30^\circ$. Если реализуется кинематическая схема фрезерования, то $\frac{v_3}{v_H} < 1$ и в этом случае $i_k =$

$0,01 \dots 0,1$, то есть влияние изменения направления вектора линейной скорости заготовки на положение основной плоскости незначительно и им можно пренебречь.

Выводы. При обработке плоской поверхности детали на токарно-фрезерном станке с ЧПУ имеет место движение режущей кромки по касательной к поверхности резания или обрабатываемой поверхности детали. Наличие касательного движения и изменения угла наклона режущей кромки в пределах от 0° до 90° приводят к существенному изменению силы и мощности резания.

ЛИТЕРАТУРА

1. Грановский, Г.И. Кинематика резания / Г.И. Грановский – М.: Машгиз, 1948. – 199с.
2. Бобров, В.Ф. Влияние угла наклона главной режущей кромки инструмента на процесс резания металлов / В.Ф. Бобров – М.: Машгиз, 1962. – 152с.
3. Попок, Н.Н. Кинематика обработки поверхностей деталей вращающимся режущим инструментом с касательным движением режущей кромки/ Н.Н. Попок, В.С. Анисимов// Вестник ПГУ, сер. В., № 11, 2019. – с. 31-38.

УДК 629.3.018.2

**ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ДИАГНОСТИРОВАНИЯ ЭЛЕКТРОННЫХ СИСТЕМ
УПРАВЛЕНИЯ ДВИГАТЕЛЕМ****Ю.А. ГАСС, Д.В. ЛОПАТИН**
(Представлено: В.В. КОСТРИЦКИЙ)

Введение. Уровень автомобилизации мирового современного общества предъявляет повышенные требования к надежности автотранспортных средств, к обеспечению технико-экономических свойств и к снижению техногенного воздействия, прежде всего - выбросов вредных веществ в окружающую среду. Требования к экологической безопасности автомобиля в мировой практике сформулированы рядом законодательных актов. Мировое автомобилестроение в настоящее время отказалось от использования несовершенных систем питания двигателей внутреннего сгорания карбюраторного типа и использует электронные системы управления режимами двигателя.

Электронная система автоматического управления двигателем состоит из датчиков для постоянного контроля за параметрами его и окружающей среды, электронного блока управления (ЭБУ) на основе микропроцессора и исполнительных устройств, с помощью которых ЭБУ управляет двигателем по заложенной в его память программе и в соответствии с информацией от датчиков.

Электронное управление необходимо для удовлетворения высоких требований по экологичности, топливной экономичности, эксплуатационным характеристикам, удобству обслуживания и диагностики, предъявляемым к современным автомобильным двигателям техническими регламентами и потребителями.

Усложнение системы привело к увеличению функциональных и параметрических отказов. Это предопределяет необходимость разработки новых методов и средств диагностирования ЭСУД.

Доля затрат на поддержание работоспособности и обеспечение надежности в эксплуатации автомобилей, оснащенных электронными системами управления двигателем (ЭСУД), в значительной мере зависит от точности и оперативности поставленного диагноза при поиске неисправности и качества последующего её устранения.

Таким образом, проблема создания надежных методов диагностирования ЭСУД с использованием современных информационных технологий является актуальной. [1]

Цель работы: повышение эффективности диагностирования электронных систем управления двигателем.

Основная часть. В современном мире диагностики электронных систем двигателя простой мультиметр уступил место сканерам и мотортестерам как более точным и многофункциональным устройствам.

При помощи сканера диагност может просмотреть коды ошибок, хранящиеся в памяти электронного блока управления (ЭБУ), произвести тест исполнительных механизмов, просмотреть значение выходных сигналов некоторых датчиков в режиме реального времени.

Мотортестер используется для определения правильности работы конкретных датчиков, исполнительных механизмов и систем двигателя. Мотортестер выдаёт информацию в виде осциллограмм, соответствующих сигналам датчиков или командам от ЭБУ к исполнительным механизмам.

Однако даже при помощи этих приборов не всегда удаётся правильно и быстро определить причину неисправности электронной системы двигателя. Так, например, ошибка считанная сканером через диагностический разъём, относящаяся к автоматической коробке передач, может быть вызвана неравномерностью работы двигателя, а мотортестер не даёт полной информации о состоянии всех систем двигателя одновременно, а только конкретной системы, датчика, механизма. [2]

Для получения точного результата диагностирования недостаточно знать код и название ошибки или текущий сигнал тестируемого элемента. Часто неисправность является временной, т. е. возникает в определённый момент времени или при определенных параметрах работы двигателя (работа на холостом ходу или под нагрузкой, на "холодную" или на "горячую" и т. д.). Важно определить при каких именно обстоятельствах появляется проблема, или что способствовало её появлению если проблема не временная, а постоянная. Для этого автомобиль прогревают до рабочей температуры либо выполняют пробную поездку. Такая диагностика не даёт точных обстоятельств и конкретного времени возникновения ошибки.

Полная информация о работе всех систем двигателя за определённый период времени содержится в файлах, называемых «логами». Блоки управления обычных автомобилей не имеют возможности сохранять такие файлы в памяти, но могут быть получены путем записи через сканер за промежуток времени выбранный диагностом.

На автомобилях, участвующих в автоспорте, устанавливаются ЭБУ, в которых «логи» записываются постоянно. Такие блоки имеют флэш-карту, на которой сохраняется информация о последних 11 часах работы двигателя. Для считывания логов с такого блока не нужно дополнительных адаптеров, достаточно извлечь флэш-карту из блока и вставить в ноутбук. Расшифровка «логов» производится с помощью специальных программ (рисунки 1,2).

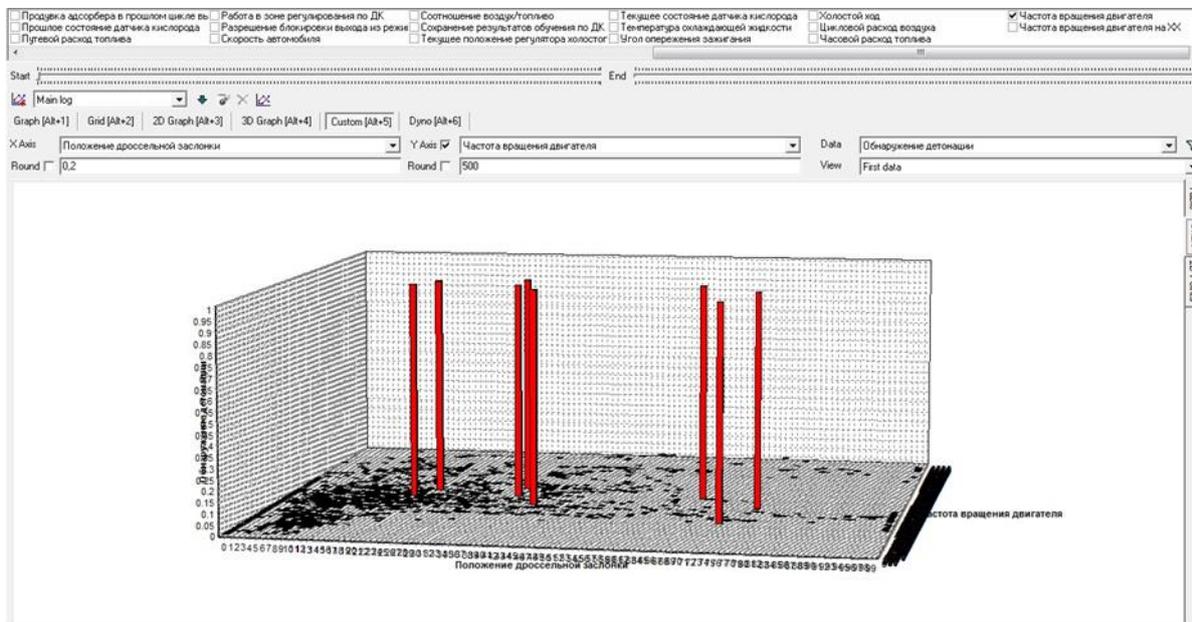


Рисунок 1. – График детонации в зависимости от положения дроссельной заслонки и частоты вращения коленчатого вала



Рисунок 2. – Одновременное отображение параметров работы двигателя

Данный метод получит более широкое распространение, если автопроизводители будут устанавливать ЭБУ с возможностью записи и хранения логов в памяти. Это позволит уменьшить время диагностики, т. к. диагност сможет прочитать уже имеющиеся данные без необходимости тестовой поездки. [3]

Заключение. Таким образом, использование «логов» при диагностике позволяет определить обстоятельства и время возникновения неисправностей, адекватность датчиков, а также влияние элементов двигателя друг на друга. При содействии автопроизводителей, данный метод позволит значительно уменьшить время диагностирования автомобиля.

ЛИТЕРАТУРА

1. Гончаров А.А., Гончаров П.А. Определение состояния элементов электронных систем управления двигателем // Прогрессивные технологии в транспортных системах: Сборник докладов шестой Российской научно-технической конференции. Оренбург: ИПК ОГУ, 2003.
2. Назначение и основные возможности комплекса USB Autoscope // InjectorService.com.ua [Электронный ресурс]. – Режим доступа: https://injectorservice.com.ua/html/main_features_v4.x.html. – Дата доступа: 22.09.2020.
3. Снятие и чтение логов ЭБУ // DRIVE2.RU [Электронный ресурс]. – Режим доступа: <https://www.drive2.ru/c/479741113528221827/>. – Дата доступа: 22.09.2020.

УДК 629.3.018.2

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ИСПОЛЬЗОВАНИЯ ОСЦИЛЛОГРАФА ПРИ ДИАГНОСТИРОВАНИИ ДВИГАТЕЛЯ СОВРЕМЕННОГО АВТОМОБИЛЯ**Ю.А. ГАСС, Д.В. ЛОПАТИН**
(Представлено: В.В. КОСТРИЦКИЙ)

Введение. Интенсивный рост использования автотранспортных средств вызывает дальнейшее повышение затрат на их техническое обслуживание и ремонт. В связи с этим роль своевременной и точной диагностики в снижении эксплуатационных затрат будет только возрастать. Техническая диагностика автомобиля - это совокупность целей и задач, связанных с поиском неисправностей механизмов и систем автомобиля, для их дальнейшего устранения. Для точного определения технического состояния механизмов, систем и агрегатов автомобиля диагностика должна проводиться квалифицированными специалистами, имеющими в своем распоряжении современное диагностическое оборудование. Для проведения работ по диагностированию автомобиля создаются специальные участки диагностики автомобиля.

Современные автомобили получили настолько обширный перечень функций и инструментов, что уже с легкостью можно приравнять их к роботу. За выполнение поставленных перед ним задач отвечает множество систем и датчиков, за которыми должен осуществляться контроль. [1]

Техническое диагностирование обеспечивает значительную экономию средств на содержание автомобилей за счет сокращения их расхода во время обслуживания и ремонта, выполнения действительно необходимых регулировочных и ремонтных операций, сокращения потребления запасных частей и топлива. Это достигается путем своевременного обнаружения и устранения неисправностей в системах зажигания, питания, а также в агрегатах трансмиссии и тормозной системы.

Перед специалистами по технической диагностике автомобилей, стоит ответственная задача — диагностика, техническое обслуживание и ремонт автомобилей с помощью современного диагностического оборудования.

Наиболее сложным узлом автомобиля является двигатель. Число неисправностей, связанных с этим агрегатом, достигает 80% от всех неисправностей автомобиля. Неправильная работа двигателя может влиять и на другие системы автомобиля.

Диагностика двигателя включает:

1. Внешний осмотр двигателя, прослушивание на не характерные шумы;
2. Проверка эксплуатационных жидкостей;
3. Проверка системы управления двигателем;
4. Диагностика системы зажигания двигателя.
5. Диагностика КШМ двигателя автомобиля;
6. Диагностика ГРМ двигателя автомобиля;
7. Диагностика систем питания (топливной, воздушной систем, системы выпуска) двигателя;
8. Проверка наполняемости цилиндров, анализ оборотов и т.д.
9. Диагностика системы охлаждения двигателя;
10. Диагностика системы смазки двигателя. [2]

Цель работы: повышение эффективности использования осциллографа при диагностировании двигателя современного автомобиля.

Основная часть. Использование осциллографа при диагностике современного автомобиля позволяет наиболее точно определить неисправность, т. к. использование сканера только локализует область поиска. Однако работа с осциллографом требует профессионального подхода.

Для уменьшения времени и повышения качества диагностирования автомобильных двигателей используется мотортестер. Он представляет собой осциллограф с расширенным функционалом. Увеличение возможностей диагностирования достигается за счёт использования компьютерной программы.

Использование специальных подпрограмм (скриптов) позволяет существенно повысить скорость диагностирования. Анализ осциллограммы, полученной при диагностировании элемента, происходит по заданному алгоритму. Перед началом анализа диагност выбирает параметры, наиболее подходящие к диагностируемому двигателю (например, порядок работы цилиндров, наличие системы изменения фаз газораспределения и т.д.). Это позволяет диагностировать различные типы двигателей практически всех производителей.

Наиболее часто используемыми скриптами являются:

- Rx;
- CSS;
- EIPower.

Скрипт Rx при помощи датчика давления, вкрученного на место свечи зажигания, позволяет проверить характеристики цилиндра, впускного и выпускного трактов, системы регулирования угла опережения зажигания и оценить взаимное влияние этих систем друг на друга (рис.1). Таким образом он намного информативнее в сравнении с компрессометром.

ЗАКЛЮЧЕНИЕ

-  Обнаружены нетипичные фазы газораспределения
-  Недостаточное наполнение цилиндра при 1000...6000 RPM (вероятно из-за неоптимальных фаз газораспределения или геометрии впускного тракта)

ВПУСКНОЙ ТРАКТ



Наполнение цилиндра на холостом ходу при 870 RPM (20...45), %	25
Максимальное наполнение цилиндра при 1000 RPM (80...101), %	66
при 1500 RPM (83...104), %	70
при 2000 RPM (86...107), %	73
при 2500 RPM (89...110), %	77
при 3000 RPM (92...113), %	78
при 3500 RPM (95...116), %	87
при 4000 RPM (98...119), %	88
при 4500 RPM (101...122), %	90
при 5000 RPM (104...125), %	97
при 5500 RPM (107...128), %	100
при 6000 RPM (110...131), %	97

Рисунок 1. – Пример скрипта Rx

Скрипт CSS предназначен для выявления причин неравномерности работы цилиндров двигателя путём сравнения их эффективности на различных режимах работы двигателя. Позволяет выяснить, какой именно цилиндр работает хуже остальных и в какой именно системе находится неполадка: в механической части двигателя, в системе подачи топлива или в системе зажигания.

Скрипт EIPower предназначен для комплексной диагностики системы пуска двигателя и электропитания автомобиля. В результате анализа графика тока в цепи аккумуляторной батареи, получаемого при помощи токовых клещей, скрипт предоставляет подробную информацию о состоянии аккумулятора, стартера, генератора и мощных электрических потребителей автомобиля (рис.2).

 12 V Аккумулятор	Начальное напряжение (12.2...12.7) Уровень заряженности (50...100) Просадка напряжения до (9.0...10.0) Фактический пусковой ток EN (580...680) Пусковой ток аккумулятора недостаточен для цепи стартера Ресурс по пусковому току от заявленных 540 EN (85...105) Аккумулятор изношен - рекомендуется заменить	12.3 V 60 % 6.6 V 256 A !!! 47 % !!!
 Стартер	Ток втягивающего реле (10...35) Время включения втягивающего реле (10...45) Фактический пиковый ток Требуемый пусковой ток аккумулятора EN (270...770)	23 A 23 ms 476 A 646 A
 Генератор	Максимальное напряжение (13.8...15.0) Действующее напряжение зарядки (13.8...14.8) Пульсации напряжения (20...80)	14.42 V 13.79 V 127 mV

Рисунок 2. – Пример скрипта EIPower

Использование скриптов позволяет получить комплексную оценку состояния двигателя. [3]

Заключение. Использование мотортестера при диагностировании двигателя позволяет получить наиболее точный результат при минимальных затратах времени. Применение специальных алгоритмов не требует высокой квалификации специалиста, что делает диагностику более доступной.

ЛИТЕРАТУРА

1. Кокорев Г.Д. Тенденции развития системы технической эксплуатации автомобильного транспорта / Г.Д. Кокорев, И.А. Успенский, И.Н. Николотов // Сборник статей II международной научно-производственной конференции «Перспективные направления развития автотранспортного комплекса». – Пенза, 2009. С. 135-138.
2. Кокорев Г.Д. Стратегии технического обслуживания и ремонта автомобильного транспорта / Г.Д. Кокорев, И.А. Успенский, И.Н. Николотов // Вестник МГАУ. Москва -2009 -№3. -С. 72-75.
3. Назначение и основные возможности комплекса USB Autoscope // InjectorService.com.ua [Электронный ресурс]. – Режим доступа: https://injectorservice.com.ua/html/main_features_v4.x.html. – Дата доступа: 22.09.2020.

УДК 629.331(075)

МЕТОДЫ ПРЯМОГО ВПРЫСКА ВОДЫ В РАБОЧЕЕ ПРОСТРАНСТВО ДВИГАТЕЛЯ**Е.А. ЕФРЕМЕНКО, А.С. ПРУДНИКОВА**
(Представлено: В.В. КОСТРИЦКИЙ)

В статье рассмотрены системы впрыска воды в рабочее пространство двигателя. Выявлены плюсы данной системы, а также влияние на количество вредных выбросов автомобиля.

Введение. Автомобиль на данный момент является одним из основных средств передвижения. Потребитель заинтересован в снижении расхода топлива и повышении мощности двигателя. А также из-за большого количества автомобилей возникает необходимость в снижении вредных выбросов. Поставленные задачи поможет решить впрыск небольшого количества воды в камеру сгорания. Испарение водяного аэрозоля сопровождается снижением температуры в камере сгорания, а наличие большого количества водяного пара в заряде приводит к повышению его теплоемкости. Совместное действие этих двух факторов ведет к снижению температур в локальных очагах горения и сокращению образования оксидов азота.

Прямой впрыск воды обеспечивал оптимальную комбинацию хорошей экономичности и снижение вредных выбросов с выхлопными газами. Это достигнуто одноразовой, относительно невысокой инвестицией и незначительным увеличением эксплуатационных затрат без снижения надежности и фактически без дополнительных требований по размещению оборудования. При этом возможно снижение потребления топлива на несколько процентов (до 13%) без потери мощности и крутящего момента. Экономия возможна на высоких оборотах: например, при резком ускорении или движении по шоссе с высокой скоростью. При этом вредные выбросы CO_2 снижаются на 4%. Наиболее эффективным будет внедрение водяного впрыска для четырёхцилиндровых двигателей. В случаи с двигателями с турбонаддувом возможно увеличение мощности до 5% [1].

В исследовательской работе были рассмотрены методы прямого впрыска воды в рабочее пространство двигателя с целью изучения влияния данной системы на расход топлива и вредные выбросы автомобиля.

Принцип работы системы. При распылении воды в двигатель вместе с впускным воздухом охлаждается впускной коллектор и воздух, попадающий в двигатель. Частицы воды позволяют сделать смесь бензина и топлива более однородной, что повышает КПД. Попадая в горячую камеру сгорания маленькие капли воды моментально испаряются, превращаясь в пар, который очищает камеру сгорания, днище поршня и свечи, а также «давит» на поршень. Вода создает паровой эффект в двигателе внутреннего сгорания, который выражается в повышении крутящего момента двигателя. А также вода вступает в химическую реакцию с выхлопными газами, что сильно снижает количество вредных выбросов. В результате реакции образуется CO_2 и H_2O .

При прямом впрыске воды в рабочее пространство происходит:

- Снижение температуры впускного воздуха
- Снижение температуры в камере сгорания
- Повышение детонационной стойкости топлива
- Снижение вредных выбросов
- Повышение мощности
- Снижение расхода топлива

Способы впрыска воды в камеру сгорания. На данный момент существуют такие способы впрыска воды как: впрыск через отдельные сопла, в одной форсунке; впрыск через отдельную форсунку, установленную в крышку цилиндра; расслоенным впрыском воды и топлива через сопла одной форсунки.

Система расслоенного впрыска топлива и воды через один распылитель, оборудована специальным золотниковым устройством, позволяющим в периоды между впрысками заполнять водой часть внутренней полости, прилегающей к игле форсунки.

При закрытом игольчатом клапане канал подвода воды в корпусе распылителя совпадает с каналом в теле игольчатого клапана. Давление подводимой воды несколько выше, чем остаточное давление в топливной магистрали, поэтому часть воды попадает в надыголочное пространство, вытесняя топливо и формируя водяную прослойку в надыголочной камере. При нагнетании топлива ТНВД канал подвода воды перекрывается обратным клапаном, а при поднятии игольчатого клапана происходит дополнительное разъединение водяной и топливной полостей за счет перекрытия канала подвода воды в корпусе распылителя образующей игольчатого клапана.

Впрыск топлива и воды через один распылитель, оборудованный специальным клапанном устройством, позволяющим в периоды между впрысками заполнять водой часть внутренней полости трубопровода высокого давления [2].

В потоке топлива, подводимого к форсунке, формировались четко выраженные зоны чередования воды и топлива. При впрыске последовательно вытеснялись топливо, вода и снова топливо, формируя в факеле распыливания четко выраженные слои (рисунок 1).

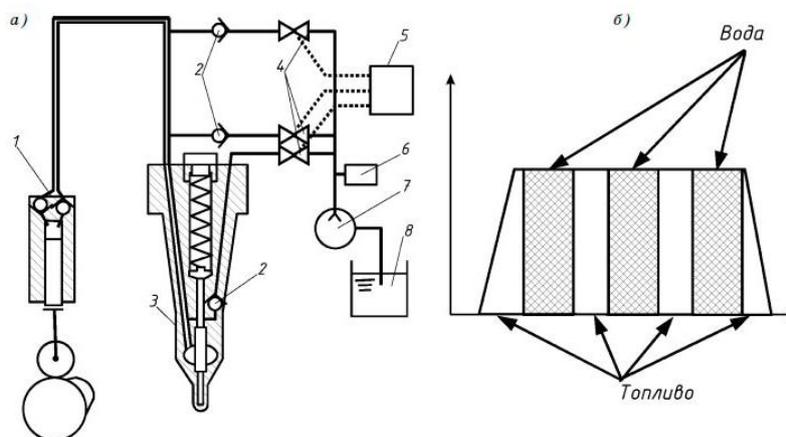


Рисунок 1. – 1-ТНВД, 2-обратные клапаны подвода воды в форсунку и линию высокого давления, 3-форсунка, 4-электроуправляемые запорные устройства, 5-электронный блок управления, 6-аккумулятор давления, 7-подкачивающий насос, 8-цистерна с водой.

Впрыск через отдельные сопла, в одной форсунке. Форсунка с двумя сопловыми наконечниками для отдельного подвода топлива и воды, а также система подачи воды к форсунке. Для впрыска воды использовался топливный насос высокого давления, выполняющий роль генератора импульсов для поршневого разделителя, который и осуществляет подачу воды к форсунке под давлением 21 МПа [3].

Электронная система управления позволяет изменять подачу воды в зависимости от режима работы двигателя в широких пределах и обеспечить аварийное отключение ее подачи в случае зависания иглы распылителя (рисунок 2).

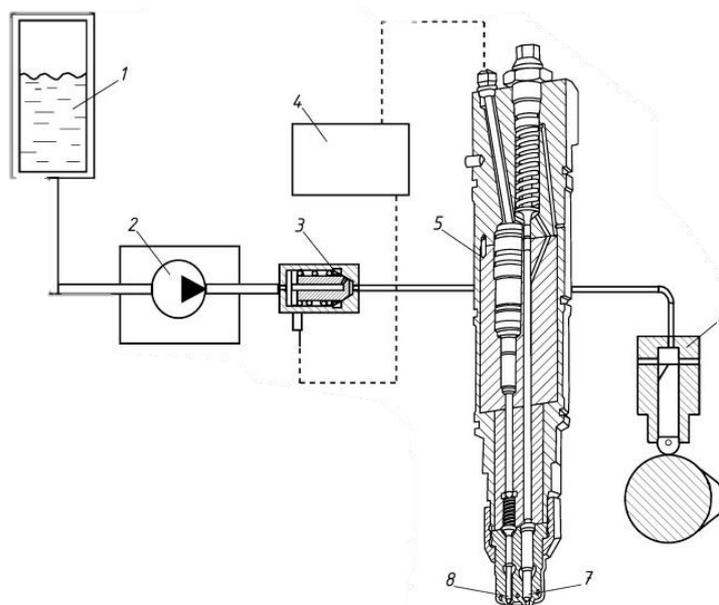


Рисунок 2. 1-цистерна с водой, 2-подкачивающий насос, 3-поршневой разделитель, 4-электронный блок управления, 5-форсунка, 6-ТНВД, используемый в качестве генератора импульсов, 7-наконечник для распыливания жидкого топлива, 8-наконечник для распыливания воды

Впрыск через отдельную форсунку, установленную в крышку цилиндра. Осуществляется через насос-форсунка, содержащая корпус, закрепленный на крышке цилиндра двигателя внутреннего сгорания, ступенчатую гильзу, закрепленную в корпусе, установленный в последней ступенчатый плунжер, рабочую и газовую камеры, образованную со стороны торца плунжера большего диаметра и сообщенную с камерой сгорания двигателя через газовый канал, рабочую водяную рубашку, сообщенную с камерой сгорания через водяной канал и через выполненное в корпусе отверстие для подвода воды с трубопроводом, обратный клапан, установленный в водяном канале, и пружину, установленную соосно с плунжером с возможностью взаимодействия с последним и с возможностью регулирования.

Закключение. По итогам исследования можно сделать следующие вывод. На данный момент прямой впрыск воды в рабочее пространство двигателя является перспективным направлением, которое позволит уменьшить затраты на топливо и увеличить экологичность автомобилей.

ЛИТЕРАТУРА

1. Коваленко, О. Л. Электронные системы автомобилей: учебное пособие / О.Л. Коваленко; Сев. (Арктич.) федер. ун-т им. М.В. Ломоносова. – Архангельск: ИПЦ САФУ, 2013. – 80 с.
2. Режим доступа: <http://chiptuner.ru/content/sensor/> – Дата доступа: 01.07.2020.
3. Режим доступа: http://injectorservice.com.ua/docs/temperature_sensor_diagnostics.pdf – Дата доступа: 13.08.2020.

УДК 629.331(075)

ПРЯМОЙ ВПРЫСК ВОДЫ В РАБОЧЕЕ ПРОСТРАНСТВО ДВИГАТЕЛЯ

Е.А.ЕФРЕМЕНКО, А.С.ПРУДНИКОВА
(Представлено: В.В. КОСТРИЦКИЙ)

В статье рассмотрены системы впрыска воды в рабочее пространство двигателя. Выявлены достоинства данной системы, а также влияние на количество вредных выбросов автомобиля.

Введение. Автомобиль на данный момент является одним из основных средств передвижения. Потребитель заинтересован в снижении расхода топлива и повышении мощности двигателя. А также из-за большого количества автомобилей возникает необходимость в снижении вредных выбросов. Поставленные задачи поможет решить впрыск небольшого количества воды в камеру сгорания. Испарение водяного аэрозоля сопровождается снижением температуры в камере сгорания, а наличие большого количества водяного пара в заряде приводит к повышению его теплоемкости. Совместное действие этих двух факторов ведет к снижению температур в локальных очагах горения и сокращению образования оксидов азота.

Прямой впрыск воды обеспечивал оптимальную комбинацию хорошей экономичности и снижение вредных выбросов с выхлопными газами. Это достигнуто одноразовой, относительно невысокой инвестицией и незначительным увеличением эксплуатационных затрат без снижения надежности и фактически без дополнительных требований по размещению оборудования. При этом возможно снижение потребления топлива на несколько процентов (до 13%) без потери мощности и крутящего момента. Экономия возможна на высоких оборотах: например, при резком ускорении или движении по шоссе с высокой скоростью. При этом вредные выбросы CO_2 снижаются на 4%. Наиболее эффективным будет внедрение водяного впрыска для четырёхцилиндровых двигателей. В случаи с двигателями с турбонаддувом возможно увеличение мощности до 5% [1].

В исследовательской работе были рассмотрены методы прямого впрыска воды в рабочее пространство двигателя с целью изучения влияния данной системы на расход топлива и вредные выбросы автомобиля.

Принцип работы: во впускной коллектор двигателя устанавливается форсунка, через которую поступает вода. При работе мотора происходит следующее: вначале во впускной коллектор поступает топливно-воздушная смесь, затем туда же впрыскивается вода, которая охлаждает топливно-воздушную смесь, поступающую в цилиндры. Благодаря тому, что частицы бензина обволакивают микрокапли воды, массовая доля горючего увеличивается, а из-за неиспарившейся жидкости возрастает степень сжатия в камерах сгорания. Скорость горения бензина, смешанного с водой, существенно падает, следовательно, условия, способствующие детонации рабочей смеси возникнуть не могут. Следует помнить, что измененный состав рабочей смеси в цилиндрах двигателя влияет на состав отработавших газов. Так, концентрация углерода и окислов азота существенно снижается, однако увеличивается доля углеводородов. Форсированный таким способом ДВС может периодически работать нестабильно. Чаще всего это происходит при движении с малой скоростью при полностью открытой дроссельной заслонке. Причина в том, что система впрыска настроена неправильно, вследствие чего во впускной коллектор попадает избыточное или недостаточное количество жидкости.

Первоначально впрыск воды применялся для борьбы с детонацией. Однако, как правило, использовался раствор воды и метилового спирта в различных пропорциях. Опытным путем было установлено, что оптимальное соотношение составляет 50/50. Сам по себе раствор играет роль антидетонационной добавки, а форсирование двигателя изначально было побочным эффектом, о котором узнали не сразу. Кроме того, вода является антиоксидантом и препятствует образованию углеродистых отложений в камерах сгорания [2].

В камерах сгорания при впрыске водного раствора метанола происходит:

– Вода обладает высокой теплоемкостью, благодаря чему существенно снижается температура в цилиндрах ДВС.

– Поскольку более холодный воздух сжать намного легче, энергии во время такта сжатия затрачивается значительно меньше, т. е. повышается КПД двигателя.

– Помимо этого, появляется возможность загнать в цилиндры больше воздуха, а вода, испаряясь, создает дополнительное давление, повышая степень сжатия.

– Жидкость попадает в цилиндры в распыленном состоянии, и мгновенно обволакивается частицами бензина, в результате рабочая смесь становится более однородной, хорошо заполняет все доступное пространство, и сгорает более равномерно. Это обеспечивает дополнительный рост КПД и снижает вероятность детонации. Таким образом, мощность ДВС возрастает примерно на 10%.

Что касается метилового спирта, процесс его сгорания протекает с меньшей скоростью, чем у бензина, поэтому рост давления в цилиндрах протекает более плавно, и максимальное значение достигается позже. В результате увеличивается крутящий момент и мощность (рисунок 1).

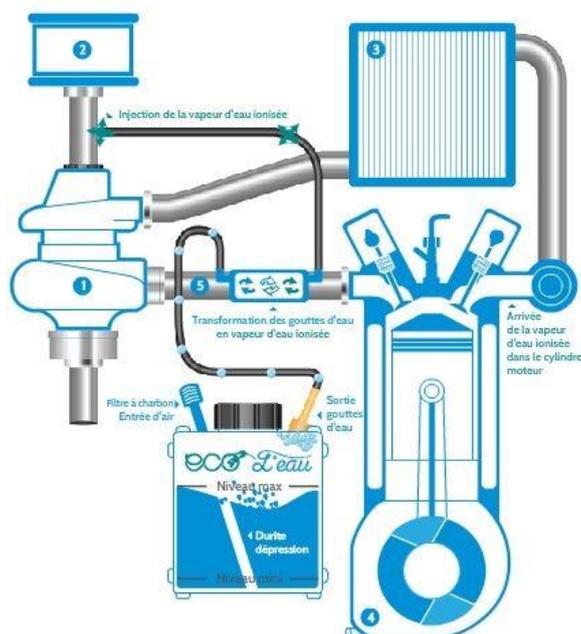


Рисунок 1. – Схема устройства подачи воды в цилиндры двигателя

В идеальном случае наибольшее количество воды должно впрыскиваться на пике момента. Соотношение воды и воздуха должно находиться в пределах от 1/10 до 1/14. При меньшем количестве воздуха рабочая смесь будет сгорать не полностью, о чем просигнализируют «выстрелы» в глушителе, а при недостатке воды может возникнуть детонация [3].

Достоинства:

- снижение температуры впускного воздуха;
- снижение температуры в камере сгорания;
- резкое повышение детонационной стойкости топлива (в том числе некачественного и низкооктанового);
- снижение вредных выбросов на 60-80%;
- повышение мощности на 15-20% и крутящего момента на 25-30%;
- снижение расхода топлива (при правильной настройке ЭБУ);
- очистку камеры сгорания и свечей зажигания.

Недостатки:

- стоимость системы;
- необходимость периодически заправлять дополнительный бачок дистиллированной водой или водно-спиртовой смесью.

Заключение. По итогам исследования можно сделать следующие выводы.

На данный момент прямой впрыск воды в рабочее пространство двигателя является перспективным направлением, которое позволит уменьшить затраты на топливо и увеличить экологичность автомобилей.

ЛИТЕРАТУРА

1. Коваленко, О. Л. Электронные системы автомобилей: учебное пособие / О.Л. Коваленко; Сев. (Арктич.) федер. ун-т им. М.В. Ломоносова. – Архангельск: ИПЦ САФУ, 2013. – 80 с.
2. Режим доступа: <http://chiptuner.ru/content/sensor/> – Дата доступа: 01.07.2020.
3. Режим доступа: http://injectorservice.com.ua/docs/temperature_sensor_diagnostics.pdf – Дата доступа: 13.08.2020.

УДК 621.924.1

ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ ПРИМЕНЕНИЯ ПОКРЫТИЙ ИЗ ИНСТРУМЕНТАЛЬНЫХ СТАЛЕЙ ДЛЯ ПОВЫШЕНИЯ СТОЙКОСТИ РЕЖУЩЕГО ИНСТРУМЕНТА

Е.В. ПУЙМАН

(Представлено: д-р техн. наук Н.Н. ПОПОК)

Исследованы структуры и фазовый состав упрочняющих покрытий высоколегированных инструментальных сталей, определено распределение модифицирующих примесей по глубине упрочненных слоев инструментальных материалов, изучена износостойкость покрытия, разработаны технологические рекомендации по режимам ионно-плазменной обработки высоколегированных инструментальных сталей.

Введение. Проблема повышения износостойкости деталей машин, работающих в условиях фрикционного взаимодействия, а также режущих инструментов, функционирующего при повышенных температурах и давлениях, становится все более значимой в связи с постоянным увеличением уровня действующих нагрузок, скоростей и рабочих температур, а также применением агрессивных сред. Одним из наиболее перспективных способов существенного повышения физико-механических, триботехнических характеристик и антикоррозионных свойств, а также выносливости сталей и сплавов является модифицирование поверхностных слоев этих материалов концентрированными потоками ионов металлов и газов [1, 2], интенсивно развиваемый в Полоцком государственном университете [3]. Важной особенностью метода является сохранение прочностных свойств подложки за счет пониженной температуры ионно-плазменной обработки, что весьма существенно для повышения ресурса работоспособности материалов с упрочненными поверхностными слоями. Весьма перспективно использование для покрытий высоколегированных инструментальных сталей мартенситного типа. Дополнительные возможности для повышения прочности и износостойкости газотермических покрытий с метастабильными фазами открывает использование технологии ионно-плазменной обработки их поверхностных слоев.

Основная часть. На рисунке 1 представлена характерная микроструктура напыленного газотермического покрытия из инструментальной стали 95ХС.

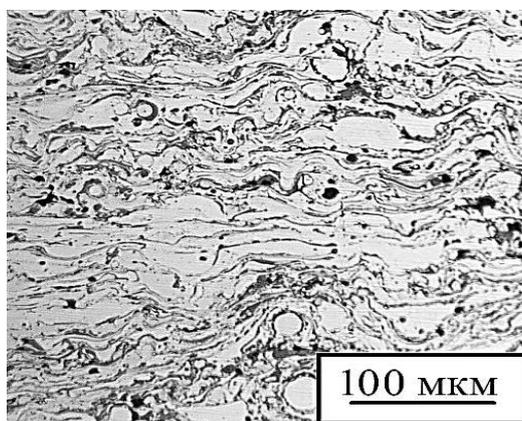


Рисунок 1. – Микроструктура газотермического покрытия из проволоочной стали 95ХС

Газотермическое покрытие из инструментальной стали 95Х18 подвергают ионно-плазменной обработке с осаждением нитрида титана. Образцы с покрытием 95Х18 перед нанесением ионно-плазменного покрытия из олова находились в неполированном и основном состояниях (рис. 2, А, Б). Ионно-плазменное покрытие осуществляли на установке ПВМ-0,5 ФН при следующих условиях: вакуум в камере установки – $3 \cdot 10^{-3}$ МПа, температура – $150-250$ °С, ток – $85-95$ А, давление азота $0,097$ Па и пропана – 100 Па, время нанесения покрытия $20-30$ мин. Образец отчета изготовлен из быстрорежущей стали марки П6М5. Износ образцов определяют путем взвешивания на аналитических весах модели ВК-1500.1.

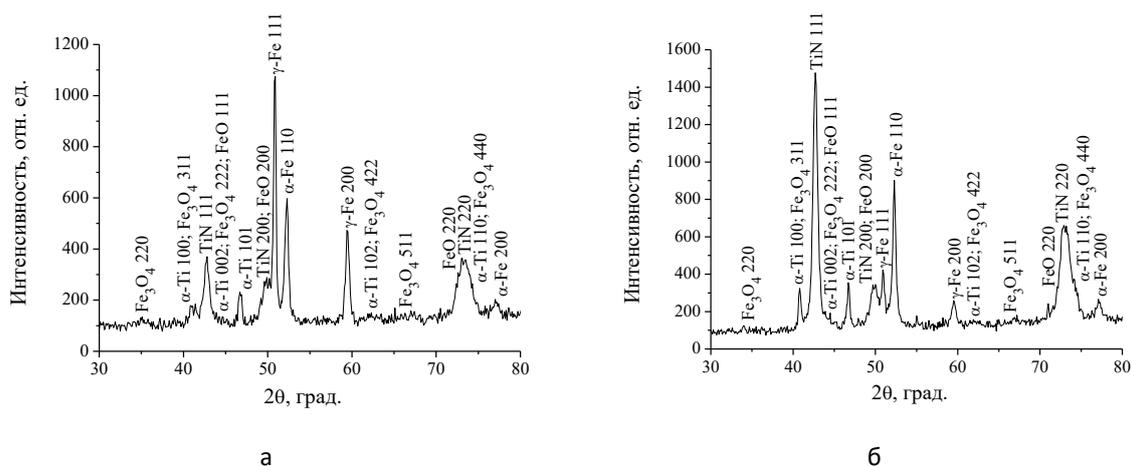


Рисунок 2 – Фрагмент рентгеновской дифрактограммы (CoKa) покрытия из стали 95XSi TiN

Сцелью оценки износостойкости образцов с комбинированным покрытием были проведены экспериментальные исследования шлифованных и нешлифованных образцов на машине трения МТ-393, при следующих режимах: усилие прижима – 50Н, скорость – 200мин⁻¹, время – 200сек.

На рисунке 3 показана машина трения МТ-393 и внешний вид образцов при проведении исследований износостойкости.



Рисунок 3. – внешний вид образцов после испытания на машине трения МТ-393:
а) до испытания б) после испытания

Результаты исследования представлены в таблице 1 и на рисунке 4.

Таблица 1 – Износ шлифованных и нешлифованных образцов

Номер образца	Вес до, г./мг.	Вес после, г./мг.	Весовой износ, г./мг.	Материал
1	4,510	4,505	0,005	95XC+TiN шлифованный
2	4,470	4,440	0,030	95XC+TiN нешлифованный

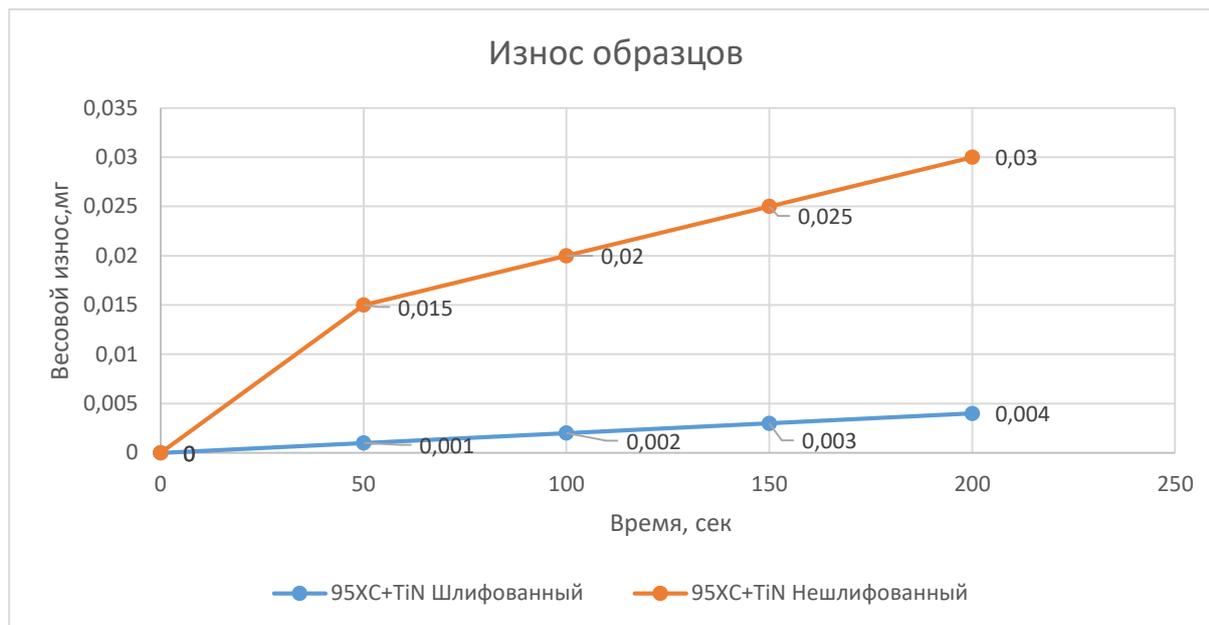


Рисунок 4. – зависимости износостойкости образцов от времени трения:
1-шлифованный, 2- нешлифованный

Как видно из графика, износостойкости образцов полученных нанесением покрытия из нитрида титана на шлифованную поверхность газотермического покрытия 95XС выше, чем не шлифованных.

Выводы:

1. Микроструктура покрытия из инструментальной стали 95X18 по пористости (менее 3%) и содержанию оксидов (~ 25%) соответствует требованиям, предъявляемым к материалам для режущего инструмента.
2. Нанесение тонкопленочного покрытия (3-5 мкм) ионно-плазменным методом на полированную поверхность образца из стали 95X18 способствует увеличению содержания олова и снижению содержания γ -фазы. Микротвердость поверхности значительно возрастает (до 4 раз).
3. Износ образцов с комбинированным покрытием из стали 95X18 и олова снижается в 2 раза по сравнению с неполированными образцами при условии, что образцы нагружены в пределах 20-40 Н и их линейная скорость не превышает 20-40 м / мин. Это значение линейной скорости соответствует режущим инструментам из быстрорежущей стали, рекомендованным для обработки материалов.
4. Рекомендуется ионно-плазменная обработка образцов в течение 30-40 мин при токе 90 а, температуре 200 ° С, вакууме $5 \cdot 10^{-3}$ МПа, давлении азота 0,097 па и пропана - 100па.

ЛИТЕРАТУРА

1. Хирвонен, Дж.К. Ионная имплантация. М.: Металлургия, 1985. 285 с.
2. Х. Риссел, И. Руге. Ионная имплантация: Пер. с нем. В.В. Климова, В.Н. Пальянова. / Под ред. М.И. Гусевой. - М.: Наука. Главная редакция физико-математической литературы, 1983. - 360с.
3. Попок, Н.Н. Ионные источники: виды, конструкции, применение = Ionsources: variety, construction, application: пособие для студентов и магистрантов машиностроительной специальности/ Н.Н.Попок, С.В. Дербущ, А.Н. Попок. – Новополоцк: Полоц. гос. ун-т, 2017. – 72с.

УДК 621.91.01

МОДЕЛИРОВАНИЕ ВЛИЯНИЯ ТЕПЛООВОГО ВОЗДЕЙСТВИЯ НА ОСНОВНЫЕ ЭЛЕМЕНТЫ
РЕЖУЩЕГО ИНСТРУМЕНТА В SOLIDWORKSС.А. СТОТИК, Р.Г. ДМИТРИЕВ., Е.С. БЕДРИЦКАЯ, В.А. КУДРЯКОВА,
Е.А. ГЛИНСКИЙ, Д.Е. МАТВЕЕНКОВ, А.А. ПОРТЯНКО
(Представлено: д-р техн. наук Н. Н. ПОПОК; С.А. ПОРТЯНКО)

Представлены результаты экспериментальных исследований влияния теплового воздействия на основные элементы режущего инструмента в программной среде SolidWorks.

Введение. В процессе резания происходит нагрев режущего инструмента, поэтому требуется прогнозировать его поведение при различных температурных режимах работы. Были проведены экспериментальные теоретические исследования. Экспериментальные исследования заключаются в реальном нагреве 3D-модели режущего инструмента, а теоретические – в моделировании нагрева режущих инструментов с использованием программной среды SolidWorks. Исследовались основные конструктивные составляющие блочно-модульной торцевой фрезы: винт, блок и корпус [1]. Надежность разработанной системы закрепления режущей пластины в блоке резцовом и блока резцового в корпусных модулях рассмотрена в [2].

Основная информация по применяемым материалам и инструментам. В ходе эксперимента для нагрева использовался промышленный фен «Stern HG2000ACN». В таблице 1 указаны его технические характеристики.

Таблица 1. – Технические характеристики промышленного фена «Stern HG2000ACN»

	
Мощность	2000 Вт
Число температурных режимов	2
Диапазон температур	350 – 550°C
Поток воздуха	300 – 500 л/мин
Количество скоростей потока воздуха	2
Защита от перегрева	есть

Температура в первом положении достигает 350°C. Во втором положении – 550°C.

3D-модели винта, блока и фрезы были напечатаны на 3D-принтере модели Mass Portal Pharaoh XD 30. Их изготовили из пластика ABS Plus. Характеристики пластика указаны в таблице 2.

Таблица 2. – Технические характеристики пластика ABS Plus

Плотность	1,05 г/см ³
Предел прочности на разрыв	30 МПа (2400МПа (23°C))
Ударная прочность	130 (при 23°C), 100 (при 130°C) КДж/м ²
Модуль упругости при растяжении	1627 МПа
Модуль упругости при растяжении при 23°C	1700 – 2930 МПа
Модуль упругости при изгибе	1834 МПа
Коэффициент удлинения	6%
Электрическая прочность	12-15 МВ/м
Влагопоглощение	0,2-0,4 %
Температура размягчения	~ 100°C
Температура плавления	~ 220°C
Температура самовоспламенения	~ 395°C

Классификация пламени	НВ
-----------------------	----

Исследование теплового влияния на основные элементы режущего инструмента. В лаборатории проводился нагрев винта, блока и корпуса фрезы до температуры плавления и исследовалось их поведение при воздействии температуры. На рисунке 1 показан порядок проведения эксперимента.

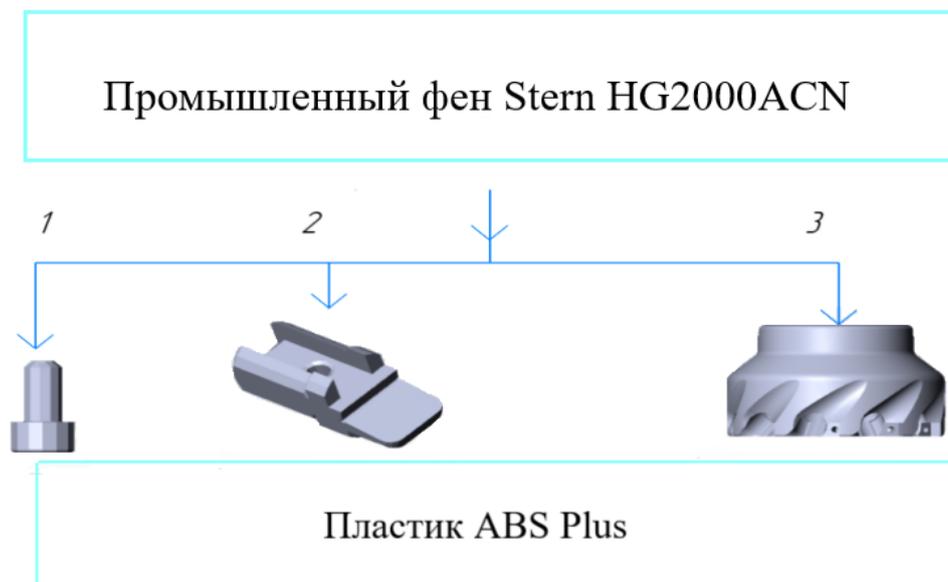


Рис. 1. Схема проведения эксперимента: 1 – винт; 2 – корпус блока; 3 – корпус фрезы торцовой

Расчетная формула температуры от точечного источника тепла представлена ниже [3]:

$$\theta(x, y, z, \tau) = \frac{q_T}{\lambda \sqrt{\omega} (4\pi\tau)^{\frac{3}{2}}} \exp \frac{-(x_u - x)^2 + (y_u - y)^2 + (z_u - z)^2}{4\omega\tau} =$$

$$= \frac{20}{0,2 \times \sqrt{1,1} \times (4 \times \pi \times 20) \times \frac{3}{2}} \times \exp \frac{-(0-0)^2 + (24-0)^2 + (0-0)^2}{4 \times 1,1 \times 20} = 175,96^\circ\text{C} \quad (1)$$

где: $\theta(x, y, z, \tau)$ – температура любой точки тела;

x, y, z – координаты точки тела;

x_u, y_u, z_u – координаты источника тепла;

τ – время действия источника;

λ и ω – коэффициенты соответственно теплопроводности и теплопроводности материала тела;

Далее было проведено моделирование нагрева винта, блока и корпуса фрезы в программной среде SolidWorks.

Температура окружающей среды 20°C . Нагрев винта, блока и корпуса фрезы проводился до температуры 150°C , так как дальнейший нагрев не целесообразен и приводит к самовоспламенению (см. табл.2). Параметры моделирования представлены в таблице 3.

Таблица 3. – Параметры моделирования в SolidWorks

Материал моделей	Пластик ABS Plus
Температура окружающей среды	20°C
Конечная температура	150°C
Время нагрева	3 с (для винта), 5 с (для блока и корпуса фрезы)

Результаты исследования представлены в таблице 4.

Таблица 4. – Результаты исследования в SolidWorks

Винт				
0с ($T_{\max}=20^{\circ}\text{C}$)	2,25с ($T_{\max}=52,5^{\circ}\text{C}$)	2,5с ($T_{\max}=95,83^{\circ}\text{C}$)	2,75с ($T_{\max}=128,33^{\circ}\text{C}$)	3с ($T_{\max}=150^{\circ}\text{C}$)
Блок				
0с ($T_{\max}=20^{\circ}\text{C}$)	2с ($T_{\max}=43,23^{\circ}\text{C}$)	3с ($T_{\max}=84,9^{\circ}\text{C}$)	4с ($T_{\max}=126,42^{\circ}\text{C}$)	5с ($T_{\max}=150^{\circ}\text{C}$)
Корпус фрезы				
0с ($T_{\max}=20^{\circ}\text{C}$)	2с ($T_{\max}=45,5^{\circ}\text{C}$)	3с ($T_{\max}=89,57^{\circ}\text{C}$)	4с ($T_{\max}=128,33^{\circ}\text{C}$)	5с ($T_{\max}=150^{\circ}\text{C}$)

Из таблицы 1 видно, что все модели в течении 1-5 с начали деформироваться под воздействием температуры, изменяя геометрические параметры.

Заключение. Если произвести сравнение практического и теоретического исследований, то видно, что они различаются по признакам проявления деформаций и температур. Практический и теоретический опыт не совпали, т.к. при реальном нагреве после отключения источника тепла температура продолжает расти.

В результате моделирования тепловых деформаций можно отметить следующее [4]:

1. При нагреве крепежного винта он удлиняется, что необходимо учитывать при проектировании резьбового соединения.
2. Режущая пластина увеличивается в «диаметральном» размере, что необходимо учитывать при определении размера паза для её размещения;
3. Корпус блока резового увеличивается в размерах, наибольшей тепловой деформации подвергаются боковые стенки продольного паза для размещения прихвата, что необходимо учитывать при назначении допусков на ширину паза и прихвата.
4. Тепловые деформации отдельных конструктивных элементов блока резового сказываются на общей деформации блока резового в сборе – режущая пластина «врастает» в корпус и прихват, размеры

прихвата изменяются по высоте, ширине и длине, деформируется форма отверстия под штифт, увеличивается диаметр штифта и диаметр винта зажимного. Эти тепловые деформации должны быть учтены при размерном анализе сборки резцового блока.

5. Качественные изменения тепловых деформаций торцевой фрезы свидетельствуют о максимальных их значениях в режущей пластине и блоке резцовом, однако ими нельзя пренебрегать и при проектировании корпуса фрезы, особенно в части торцевой шпонки, с помощью которой фреза закрепляется в хвостовике.

ЛИТЕРАТУРА

1. Попок, Н. Н. 3D-моделирование конструкций блочно-модульных торцевых фрез / Н. Н. Попок, С. А. Портянко, Г. И. Гвоздь // Информационно-коммуникационные технологии: достижения, проблемы, инновации (ИКТ-2018): электронный сборник статей I международной научно-практической конференции, посвященной 50-летию Полоцкого государственного университета, Новополоцк, 14–15 июня 2018 г. / Полоцкий государственный университет ; пред. редкол.: Д. Н. Лазовский. – Новополоцк, 2018. – С. 190
2. Попок, Н.Н., Максимчук, А.С., Портянко, С.А. Совершенствование системы закрепления пластин режущих и блоков резцовых в блочно-модульных режущих инструментах/ Вестник ПГУ, сер. В – 2015 – №2, с. 16-22.
3. Попок, Н.Н. Теория резания: учебное пособие / Н.Н. Попок. – Минск : ИВЦ Минфина, 2019 г. – 372 с.
4. Методология исследования работоспособности фрезерных и осевых режущих инструментов на основе 3D прототипирования / Н.Н. Попок, С.А. Портянко – Вестник ПГУ, сер. В. Промышленность. Прикладные науки. 2020 г. – с. 29-39.

УДК 621.91.01

МОДЕЛИРОВАНИЕ РАБОТЫ ТОРЦОВЫХ ФРЕЗ В ПРОГРАММЕ CINEMA 4D

С.А. СТОТИК, Р.Г. ДМИТРИЕВ., Е.С. БЕДРИЦКАЯ, В.А. КУДРЯКОВА,
Е.А. ГЛИНСКИЙ, Д.Е. МАТВЕЕНКОВ, А.А. ПОРТЯНКО

(Представлено: д-р техн. наук Н.Н. Попок; С.А. Портянко)

Для успешной работы с фрезами важное значение имеет понимание взаимодействия их конструкции с аэро-гидродинамической средой. Цель исследовательской работы состоит в исследовании воздействия аэро- и гидродинамических сред на конструкции фрез и их работоспособность. Предложены различные методы измерения поверхностных напряжений трения. Разработаны пневмометрические методы косвенного определения вектора поверхностных напряжений трения в пространственных течениях. Показано моделирование динамики работы фрез в вязко-жидкостной среде. Так же приведены результаты исследований. Получены новые научные результаты и методические обоснования имеют научно-практическую ценность для дальнейшего развития конструирования торцовых фрез и режущего инструмента.

Введение. Применение сборных режущих инструментов в промышленности достигает 70% от общего количества инструмента, причем одну треть этих инструментов составляют сборные торцовые фрезы. Все большее распространение получают сборные торцовые фрезы, состоящие из взаимозаменяемых модулей, применение которых значительно сокращает затраты на производство.

В настоящее время практически единственным методом определения обтекаемости конструкций является метод плавающего весового элемента. Однако его применение на поверхностях с большой кривизной или при действии значительных знакопеременных инерционных нагрузок, затруднено.

Из косвенных пневматических методов более распространено использование трубок Престона, насадки Стантона. Эти методы так же используются и при измерении напряжений сдвига в пространственных течениях, при этом направление вектора напряжений сдвига предварительно определяется с помощью метода масляных капель или теплового измерителя.

Использование данных пневмометрических методов затруднено или вообще невозможно в сильно изменяющихся по направлению течениях, а также при действии на исследуемый объект значительных инерционных нагрузок.

К косвенным методам определения напряжений сдвига в стенке относятся метод флуоресцирующих покрытий, метод измерения касательных напряжений по длине штрихов в результате растекания масляных капель, наносимых на поверхность, а также методы, использующие аналогию Рейнольдса между температурным и динамическим пограничными слоями (рисунок 1) [1].

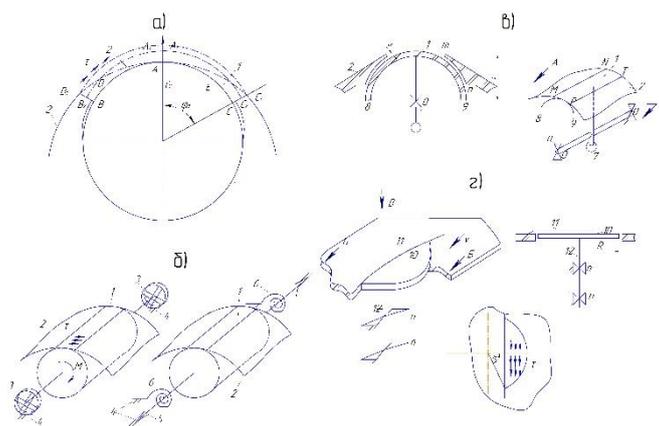


Рисунок 1. – Описание: а – цилиндрический элемент касается поверхности круга;
б – модификация с тензометрическими весами; в – часть цилиндрической поверхности;
г – чувствительный элемент в виде плоского диска

Описание программного продукта Cinema 4D. На 2020 год Cinema 4D является самой простой в использовании программой для 3D художников, по крайней мере, так заявлено на сайте разработчиков.

Все основные преимущества также присутствуют: удобство в использовании, всеми любимым интуитивный интерфейс, стабильность, интегрированная справка, процедурный рабочий процесс и доступность программы в разных вариантах в зависимости от целей использования.

Cinema 4D – универсальная программа для 3D моделирования, редактирования объектов и создания эффектов. Также выполняет рендеринг объектов по методу Гуро (Метод закраски, который основан на интерполяции интенсивности и известен как метод Гуро (по имени его разработчика), позволяет устранить дискретность изменения интенсивности). Рендеры могут быть как “родные”, так и встраиваемыми непосредственно в саму программу с помощью плагинов и коннекторов.

Эту программу обожают использовать студенты, так как не потребуется много времени и усилий на изучение принципов работы. Именно в этом редакторе были исследована работа фрез в вязко-жидкостной среде.

Плагины и настройка Cinema 4D. Для моделирования работы фрез в жидкостной среде был использован плагин «RealFlow». Этот плагин предназначен для моделирования и симуляции разнообразных физических тел в динамике и нацелен для использования прежде всего в индустрии компьютерной графики, анимации и спецэффектов, а не в научных расчётах и исследованиях. В основе своей использует системы частиц для просчета динамики формы физических тел. Алгоритм расчетов «RealFlow» использует метод гидродинамики сглаженных частиц (англ. Smoothed Particle Hydrodynamics — SPH), который позволяет моделирующим среду частицам взаимодействовать между собой. Суть SPH состоит в том, что в зависимости от расстояния между двумя частицами эти частицы могут «слипаться» или «расталкиваться». Каждая частица может иметь свою массу и так называемую «длину сглаживания», которая определяет, на каком расстоянии эта частица будет притягиваться к другим, а на каком — отталкиваться. Все тела, моделируемые «RealFlow», построенные на точечных частицах, механизм работы которых описан выше. Изначально «RealFlow» был предназначен для моделирования только текучих и вязких тел, в первую очередь жидкостей, однако впоследствии набор поддерживаемых физических тел существенно расширился. В результате «RealFlow» 5-й версии поддерживает твёрдые и деформируемые тела, газы и «мешы».

«RealFlow» может работать как полностью отдельное (standalone) приложение, а может использоваться в качестве компонента (плагины) к другим программам, таким как 3D Max, Maya, LightWave, Softimage, Cinema 4D и Houdini.

В проекте, где проводилось исследование были использованы следующие элементы (рисунок 2):

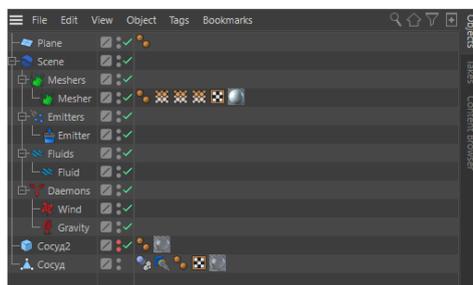


Рисунок 2. – Элементы проекта

Плагин создает раздел «Scene», который содержит все элементы и настройки жидкости.

«Meshers» является текстурой самой воды. Объект «Emitters» хранит все настраиваемые свойства воды, которые возможно изменить в любой момент (рисунок 3). Где «Телом» указан объект «Сосуд», чтобы вода не проливалась сквозь него.

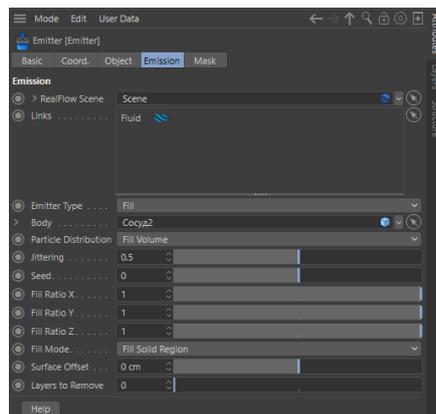
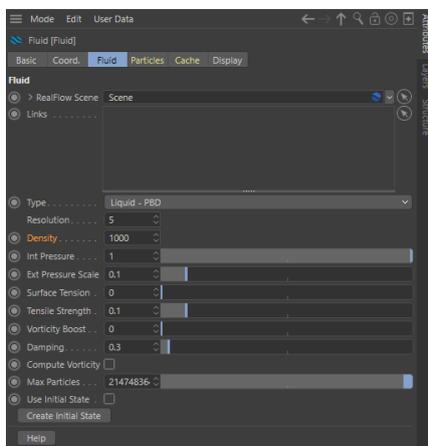


Рис.3. Настройка «Emitter»

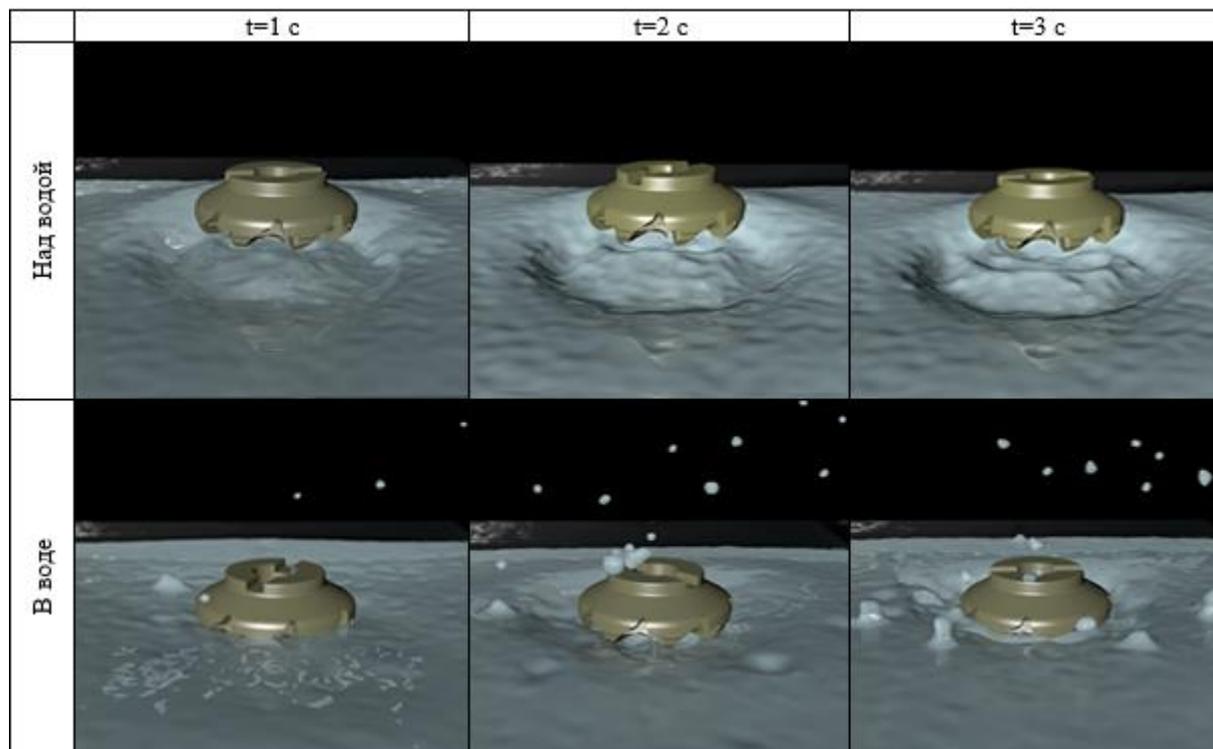
«Fluids» отвечает за качество и разрешение гранул, из которых состоит вода. В данном случае, разрешение воды выставлено на 5 (рисунок 4).

**Рисунок 4. – Настройка «Fluid»**

В разделе «Daemons» хранятся вся физика воды («Wind», «Gravity»). Где «Wind» отвечает за ветер, создаваемый вращением фрезы, а «Gravity» отвечает за гравитацию воды.

Исследование

Таблица 1. – Исследование динамики работы первой фрезы в среде жидкости



В таблице 1 и 2 представлены покадрово (фрагментально) изменения жидкости (волны, потоки, брызги) в течение определенного промежутка времени $t=1, 2, 3$ с.

Таблица 2. – Исследование динамики работы второй фрезы в среде жидкости

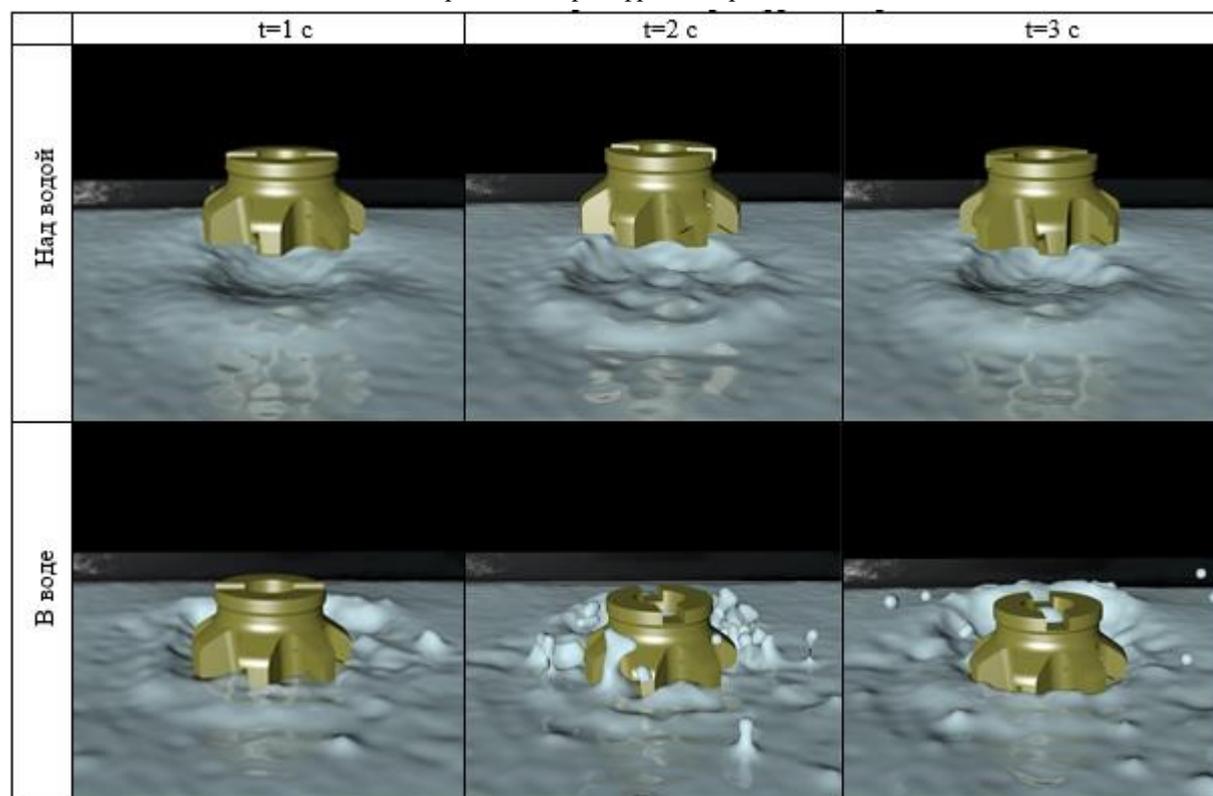
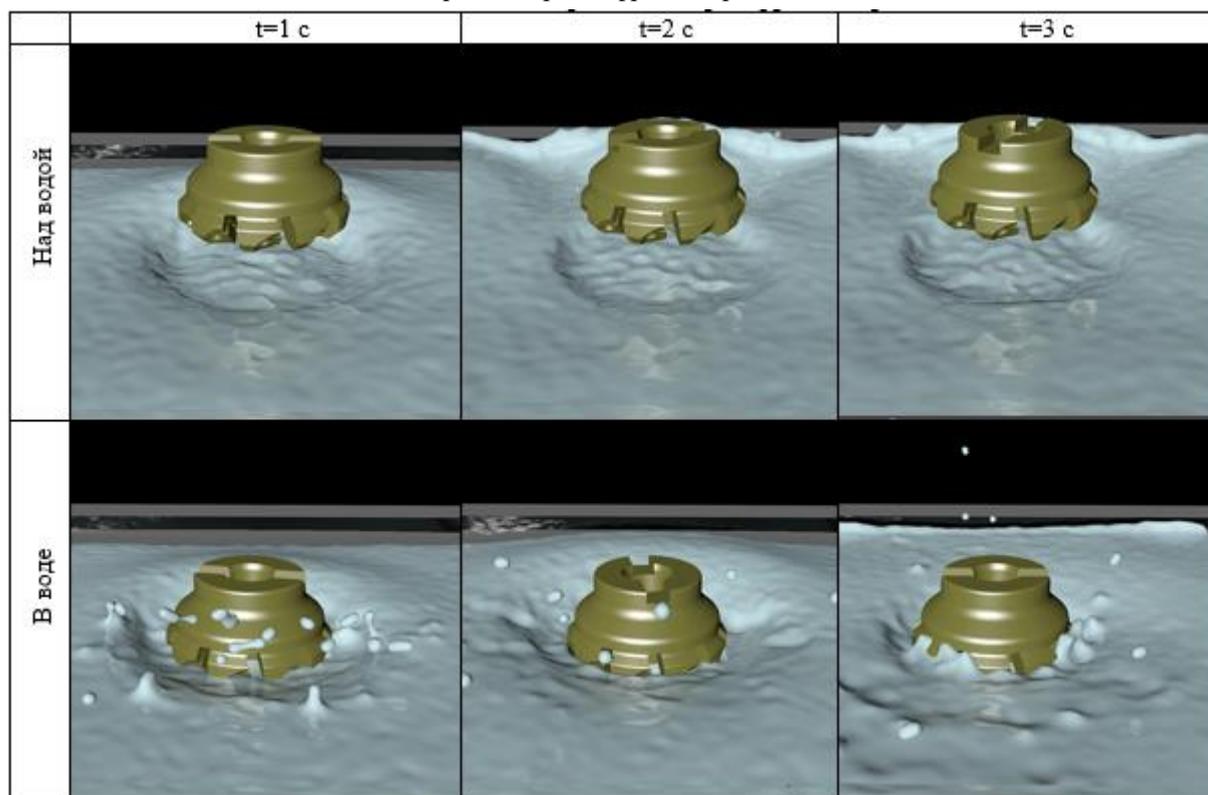


Таблица 3. – Исследование динамики работы третьей фрезы в среде жидкости



Описание. Фрезерование является одним из наиболее распространенных методов обработки. По уровню производительности фрезерование превосходит строгание и в условиях крупносерийного производства уступает лишь наружному протягиванию. Кинематика процесса фрезерования характеризуется быстрым вращением инструмента вокруг его оси и медленным движением подачи. Движение подачи при фрезеровании может быть прямолинейно-поступательным, вращательным, либо винтовым. При прямолинейном движении подачи фрезой производится обработка всевозможных цилиндрических поверхностей: плоскостей, всевозможных пазов и канавок, фасонных цилиндрических поверхностей.

Экспериментальные исследования проводились в программном обеспечении Cinema 4D, с использованием дополнительного плагина, позволяющего симулировать поведение жидкости и взаимодействия с ней. Результаты исследования представлены в таблицах, из которых видно воздействие фрез на поверхность жидкости. В нашем случае поверхностью служит поверхность жидкости, на которую воздействует фреза потоком воздуха. Каждая фреза, вращалась вокруг своей оси над поверхностью, воздействуя на жидкость. частота вращения шпинделя станка n : 1000 об/мин. Из-за различных размеров канавок на конструкции фрез, они имеют различные свойства прохождения через них потоков воздуха.

Фреза 2 воздействует на поверхность жидкости меньше всех, так как она имеет большую площадь. В случае с 1 и 3 фрезой они воздействуют на меньшую площадь, но с большей силой, что видно на изображениях в таблицах.

Если поместить вращающиеся фрезы в жидкость, то мы увидим, что в процессе вращения, фрезы будут отводить воду через канавки и формировать волны, всплески на поверхности жидкости. У фрезы с больше площадью при вращении, образуется большее количество волн, когда у фрезы 1 и 3, при меньшем размере канавок отводится больше жидкости, что вызывает большее количество всплесков.

Заключение. В ходе исследования выявили возможности методики изучения аэро-гидрадинамической обтекаемости, воздействия потоков воздуха на формы конструкции фрез. Так же с помощью программного обеспечения Cinema 4D и дополнения «RealFlow» смоделировали симуляцию жидкостной среды, для взаимодействия с 3D моделью различных фрез, соответственно изучили воздействия стружкоотводящих канавок на жидкостную среду, как с помощью проходящего через них воздуха, так и прямого взаимодействия фрезы с водой.

ЛИТЕРАТУРА

1. Методы определения вектора аэрогидродинамических напряжений трения на поверхности тел, обтекаемых пространственным или плоским потоком. [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/metody-opredeleniya-vektora-aerogidrodinamicheskikh-napryazheniy-treniya-na-poverhnosti-tel-obtekaemyh-prostranstvennym-ili-ploskim/viewer/>. – Дата доступа: 20.09.2020.
2. Методология исследования работоспособности фрезерных и осевых режущих инструментов на основе 3D прототипирования / Н.Н. Попок, С.А. Портянко – Вестник ПГУ, сер. В. Промышленность. Прикладные науки. 2020 г. – с. 29-39.

УДК 621.91.01

**СРАВНИТЕЛЬНЫЙ АНАЛИЗ 3D ПРИНТЕРОВ
НА ОСНОВЕ FDM И SLA ТЕХНОЛОГИЙ**

**С.А. СТОТИК, Р.Г. ДМИТРИЕВ., Е.С. БЕДРИЦКАЯ, В.А. КУДРЯКОВА,
Е.А. ГЛИНСКИЙ, Д.Е. МАТВЕЕНКОВ, А.А. ПОРТЯНКО**
(Представлено: д-р техн. наук Н. Н. ПОПОК; С.А. ПОРТЯНКО)

Введение. Моделирование методом слитого осаждения (FDM) и стереолитография (SLA) являются двумя наиболее популярными технологиями 3D-печати. Обе технологии адаптированы и усовершенствованы, что делает их более доступными, простыми в использовании и более функциональными.

Моделирование методом послойного нанесения материала является наиболее широко используемой технологией 3D-печати на уровне потребителей. FDM работает путем экструзии термопластов, таких как ABS, PLA, через нагретый экструдер, плавления материала и нанесения пластика слой за слоем. Эти типы 3D-принтеров хорошо подходят для базовых моделей, проверки концепции, а также для быстрого и недорогого создания прототипов.

Стереолитография была первой в мире технологией 3D-печати, изобретенной в 1980-х годах, и до сих пор остается одной из самых популярных технологий для профессионалов. SLA использует лазер для отверждения жидкой смолы в затвердевшую пластмассу в процессе, называемом фотополимеризацией.

3D-принтеры SLA стали чрезвычайно популярными благодаря своей способности производить высокоточные, изотропные и водонепроницаемые прототипы и детали из самых современных материалов с прекрасными характеристиками и гладкой поверхностью. Смолистые составы SLA предлагают широкий спектр оптических, механических и термических свойств, соответствующих свойствам стандартных, технических и промышленных термопластов.

SLA – отличный вариант для высокодетализированных прототипов, требующих жестких допусков и гладких поверхностей, таких как формы, узоры и функциональные детали. SLA широко используется в различных отраслях промышленности: от машиностроения и проектирования изделий до производства, стоматологии, ювелирных изделий, создания моделей и образования.

Сравнение технологий 3D-печати FDM и SLA. Качество и точность печати. Когда процессы 3D-печати производят детали слой за слоем, каждый слой также вносит возможность неточности. Процесс формирования слоев влияет на качество поверхности, уровень точности, а, следовательно, и на общее качество печати.

3D-принтеры FDM слои могут не полностью прилипать друг к другу, слои, как правило, хорошо видны на поверхности, и в процессе отсутствует способность воспроизводить сложные, мелкие детали, которые могут предложить другие технологии.

В 3D-печати SLA, позволяет получать более мелкие детали и является более надежным для многократного достижения высококачественных результатов. В результате, SLA 3D-печать известна своими прекрасными характеристиками, гладкой поверхностью, высочайшей точностью деталей.

Использование света вместо тепла для печати - это еще один способ, которым машины SLA гарантируют надежность. При печати деталей при температуре, близкой к комнатной, они не страдают от теплового расширения и артефактов сжатия, которые могут возникнуть в процессе печати FDM.

В то время как FDM создает механическую связь между слоями, 3D-принтеры SLA создают химические связи путем поперечной сшивки фотополимеров между слоями, в результате чего получаются полностью плотные, водонепроницаемые и воздухонепроницаемые детали. Эти связи обеспечивают высокую степень боковой прочности, что приводит к изотропным деталям, а это означает, что прочность деталей не изменяется с ориентацией. Это делает SLA особенно идеальным для машиностроения и производства, где важны свойства материала.

Материалы. 3D-принтеры FDM работают с рядом стандартных термопластов, таких как ABS, PLA и их различными смесями. Популярность FDM в пространстве любителей привела к изобилию вариантов цвета. Существуют также различные экспериментальные смеси нитей для создания деталей с поверхностью, подобной дереву или металлу.

Инженерные материалы, такие как нейлон, PETG, PA или TPU и высокопроизводительные термопластики, такие как PEEK или PEI, также доступны, но часто ограничиваются избранными профессиональными принтерами FDM, которые их поддерживают.

Смолы SLA обладают преимуществом широкого спектра конфигураций составов: материалы могут быть мягкими или твердыми, сильно заполненными материалами, такими как стекло и керамика, или наполненными механическими свойствами, такими как высокая температура отклонения тепла или ударопрочность.

Различные составы смол предлагают широкий спектр оптических, механических и термических свойств, чтобы соответствовать свойствам стандартных, технических и промышленных термопластов.

Описание и технические характеристики 3D принтеров 3D-принтер Premier3D N1 (FDM)



Рисунок 1. – 3D-принтер Premier3D N1

Принтер выполнен из высококачественного металла и имеет изолированную камеру печати, закрытую ударопрочным стеклом. Несомненным преимуществом данного принтера является система фильтрации воздуха, которая позволит сократить до минимума количество вредоносных испарений, выделяемых при печати.

Внутренние размеры области печати, составляющие 200x200x200 мм., вкпе с компактным корпусом и малым весом, позволят использовать Premier3D N1 в домашних условиях без каких-либо проблем, совершенно не ограничивая пространство.

Скорость печати составляет внушительные 70 мм/с при толщине всего 0,1 мм. Такая производительность была достигнута путем применения специальных алгоритмов печатающей головки. LCD панель, отвечающая за настройку принтера, обладает высокой информативностью и большим количеством функций, позволяя настроить Premier3D для печати любых моделей, используя необходимый вид пластика [1].

Таблица 1. – Технические характеристики

Основные	
Тип	FDM принтер
Количество экструдеров	1
Корпус	закрытый
Материал рамы	сталь
Минимальная толщина слоя	100 мкм
Технические характеристики	
Длина области печати	200 мм
Ширина области печати	200 мм
Высота области печати	200 мм
Поддерживаемые материалы	PLA, ABS
Толщина материала	1,75 мм
Подогрев стола	✓
Температура стола	60 – 120 °С
Температура экструдера	190 – 260 °С
Максимальная скорость печати	70 мм/с
Дисплей	✓
Потребляемая мощность	360 Вт
Комплектация	
Требуется сборки	×
Размеры и вес	
Длина	480 мм
Ширина	505 мм
Высота	450 мм
Вес	19,9 кг

3D-принтер Stratasys Mojo (FDM)

3D-принтер **Mojo** от компании Stratasys – это надежная и быстрая система трехмерной печати, которая легко поместится на вашем столе.



Рисунок 2. – 3D-принтер Stratasys Mojo

Простой в использовании и качественный в результатах – принтер Mojo специально создан для быстрой печати прототипов с помощью прочного пластика ABSplus. Небольшая стоимость, надежность и простота в использовании делают Mojo примером доступного 3D-принтера с хорошими характеристиками. Это лучший выбор для тех, кто хочет познакомиться с возможностями трехмерной печати. Технология работы принтера – FDM – является самой популярной технологией 3D-печати в мире. Стабильность результатов печати также обеспечивают продвинутое программное обеспечение и автоматический контроль температуры внутри камеры построения. Это позволит вам получать качественные прототипы высокой четкости, не отходя от рабочего места [2].

Таблица 2. – Технические характеристики

Технология печати		Параметры печати	
FDM (Fused Deposition Modeling)	послойное нанесение расплавленной пластиковой нити	Толщина слоя	0,178 мм
Размер области построения (максимальный размер получаемой модели)		Режимы печати	настраиваемый (заполняемость модели материалом, алгоритм выравнивания поддержки)
Длина	127 мм	Параметры печатающего блока	голова – 1, сопел – 2
Ширина	127 мм	Скорость вращения	
Высота	127 мм	Максимальная производительность	
Размер рабочей камеры		Поддержка цветной печати	9 цветов материала
Длина	127 мм	Поддерживаемые материалы	
Ширина	127 мм	Общее количество	1 материал
Высота	127 мм	Материалы	ABSplus-P430 — инженерный пластик (доступны 9 цветов)
		Материал поддержки модели	SR-30 Soluble — специальный растворимый пластик
		Картриджи	2 катушки по 655 см ³

3D-принтер Form3 (SLA)

Form 3 использует технологию стереолитографии низкой мощности (LFS) с гибким резервуаром и линейной подсветкой, которая обеспечивает безупречное качество печати и надежность принтера. Парabolicкое зеркало гарантирует, что лазер печатает перпендикулярно плоскости, обеспечивая равномерное качество печати по всей платформе [3].



Рисунок 3. – 3D-принтер Form3

Form 3 постоянно отслеживает производительность печати, встроенные датчики помогают поддерживать идеальные условия и отправляют оповещения о состоянии принтера. Оптические датчики постоянно корректируют масштаб и мощность, и даже могут обнаруживать пыль. Автоматизированная система подачи смолы позволяет печатать больше изделий, оставляя меньше отработанных материалов, а также увеличить срок жизни ванны для печати. Печатающая камера с воздушным автоподогревом до 35 °С.

Таблица 3. – Технические характеристики

Основные	
Тип	SLA принтер
Корпус	закрытый
Минимальная толщина слоя	25 мкм
Технические характеристики	
Длина области печати	145 мм
Ширина области печати	145 мм
Высота области печати	185 мм
Точность позиционирования оси X, Y	25 мкм
Точность позиционирования оси Z	25 мкм
Поддерживаемые материалы	фотополимер
Подогрев стола	×
Полноцветная печать	×
Дисплей	✓ (5.5", 1280×720)
Управление	сенсорный дисплей
Интерфейсы	USB, Wi-Fi, Ethernet
Поддерживаемые форматы файлов	.stl, .obj
Камера	×
Автоматическая калибровка платформы	×
Возможность паузы при печати	✓
Мощность лазера	250 мВт
Потребляемая мощность	220 Вт

3D принтер Mass Portal Pharaon

Качество 3D-принтеров Mass Portal – как качество их сборки, обеспечивающее надежность и бесперебойность, так и качество, и точность работы, дающее лучший результат при печати, – позволило им распространиться по нескольким континентам и привлечь к себе заслуженное внимание.

Особенность серии XD, которую представляют XD 20, XD 30 и XD 40 – возможность печатать несколькими материалами одновременно. [4]



Рис. 4. 3D принтер Mass Portal Pharaon

Подробнее о них всех – в таблице ниже. В таблице есть ещё и цены, но они ориентировочные – цена на любой 3D-принтер может как вырасти, так и снизиться, точную лучше узнавать на сайте.

Таблица 4. – Технические характеристики

Основные	
Тип	FDM принтер
Количество экструдеров	1
Корпус	закрытый
Материал рамы	алюминий
Минимальная толщина слоя	10 мкм
Технические характеристики	
Длина области печати	300 мм
Ширина области печати	300 мм
Высота области печати	300 мм
Точность позиционирования оси X, Y	6 мкм
Точность позиционирования оси Z	5 мкм
Поддерживаемые материалы	PLA, ABS, PVA, Nylon, PET, NIPS, FLEX, PC, TPU, PP, PETG
Толщина материала	1,75 мм
Материал стола	стекло

Подогрев стола	✓
Температура стола	120 °C
Температура экструдера	300 °C
Максимальная скорость печати	300 мм/с
Дисплей	✓
Управление	сенсорный дисплей
Интерфейсы	USB, Wi-Fi, Ethernet
Поддерживаемые форматы файлов	.stl, .obj, .gcode
Автоматическая калибровка платформы	✓
Возможность паузы при печати	✓
Потребляемая мощность	360 Вт
Цвет корпуса	черный
Комплектация	
Карта памяти	✓ (8 Гб)
Сопла	0.4 мм (опционально 0.1, 0.2, 0.6, 0.8, 1.0, 1.2, 1.4 мм)
Требуется сборки	×
Размеры и вес	
Длина	483 мм
Ширина	425 мм
Высота	875 мм
Вес	39.8 кг

Заключение. В процессе анализа 3D принтеров на основе FDM, SLA, мы выявили их различия. Влияние основы принтера на температуру экструдера, подогрева стола, потребляемой мощности, поддерживаемые материалы, влияние различных факторов на скорость печати. Так же изучили толщину материалов, минимальную толщину слоев. Проанализировали габариты каждого принтера, размеры области печати. Область печати у рассматриваемых принтеров на основе FDM от 145 до 300мм, а у принтера на основе SLA, значительно меньше 127мм. В принтерах, работающих по технологии FDM минимальная толщина слоя – 100мкм, когда на основе SLA – 127 мкм. Также существует разница в технологии печати 3D-принтеры FDM формируют слои путем нанесения линий расплавленного пластика. При этом процессе разрешение детали определяется размером сопла экструдера, и между закругленными линиями возникают

пустоты, когда сопло осаждает их. В 3D-печати SLA, жидкая смола отверждается высокоточным лазером для формирования каждого слоя, что позволяет получать более мелкие детали и является более надежным для многократного достижения высококачественных результатов.

ЛИТЕРАТУРА

1. 3D – принтер PREMIER 3D N1. [Электронный ресурс]. – Режим доступа: <https://easycraft.by/3d-printer-premier-3d-n1/>. – Дата доступа: 20.09.2020.
2. 3D принтер Stratasys Mojo [Электронный ресурс]. – Режим доступа: <https://3d-m.ru/3d-printer-stratasys-mojo/>. – Дата доступа: 22.09.2020.
3. Form 3. [Электронный ресурс]. – Режим доступа: <https://formlabs.com/ru/3d-printers/form-3/>. – Дата доступа: 28.09.2020.
4. Mass Portal. [Электронный ресурс]. – Режим доступа: <https://massportal.com/>. – Дата доступа: 29.09.2020.

УДК 621.91.01

МОДЕЛИРОВАНИЕ ВОЗДУШНЫХ ПОТОКОВ В SOLIDWORKS FLOWSIMULATION

**С.А. СТОТИК, Р.Г. ДМИТРИЕВ., Е.С. БЕДРИЦКАЯ, В.А. КУДРЯКОВА,
Е.А. ГЛИНСКИЙ, Д.Е. МАТВЕЕНКОВ, А.А. ПОРТЯНКО**
(Представлено: д-р техн. наук Н. Н. ПОПОК; С.А. ПОРТЯНКО)

Создание модели распространения воздушных потоков торцовых фрез для проведения последующего анализа аэродинамических свойств при обработке материалов резанием.

Для создания модели распространения воздушных потоков нам понадобится: 3D – модель исследуемого объекта; 3D – модель корпуса, фиксирующего объект. Следующим шагом будет создание сборки исследуемой модели (рисунок 1).

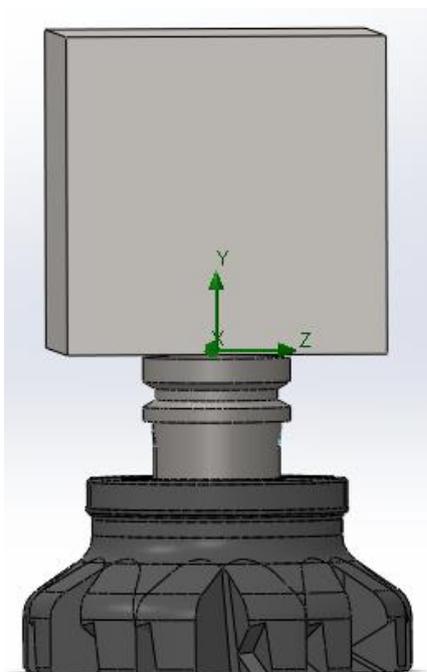


Рисунок 1. – Сборка исследуемой модели

После чего нужно добавить трехмерную модель, при помощи эскиза и выдавливания, выделяем область вращения исследуемой модели (рисунок 2).

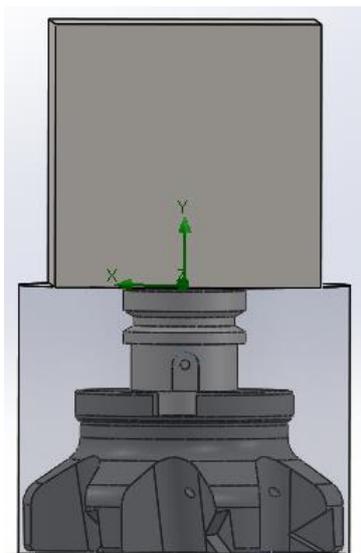


Рисунок 2. – Сборка с областью вращения

Далее с помощью FlowSimulation запускаем мастер проекта с помощью соответствующей кнопки. После чего в открывшемся окне выбираем метрическую систему, вид моделирования (в нашем случае вращение), вид газа или жидкости. После чего в новом проекте задаем область вращения и скорость вращения, при помощи созданной детали, и граничное условие, фиксирующее нашу деталь (рисунок 3).

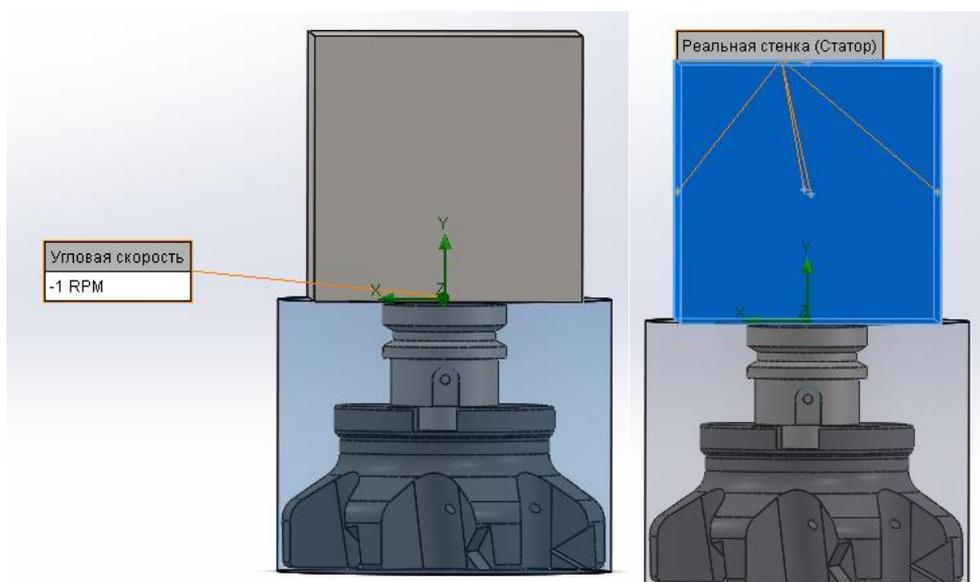


Рисунок 3. – Область вращения и граничные условия.

После чего наш проект готов к моделированию. Запускаем проведение расчетов с помощью соответствующей кнопки «Запустить». После завершения расчетов открываем раздел результаты и добавляем траекторию потока для визуализации (рисунок 4).

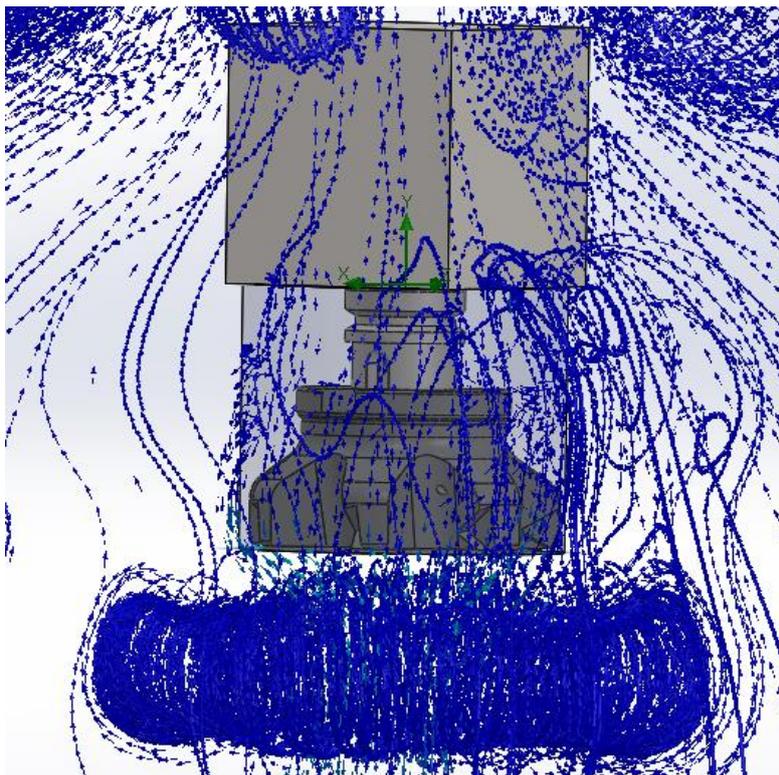


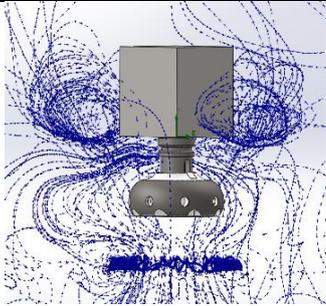
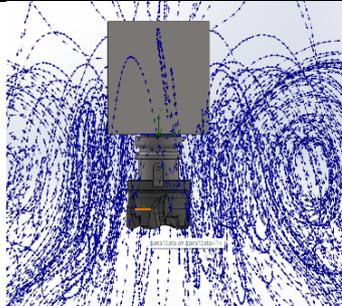
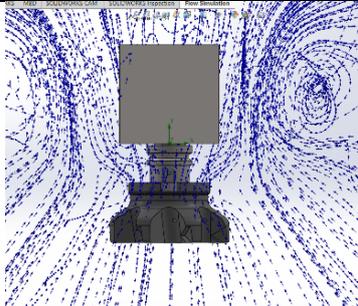
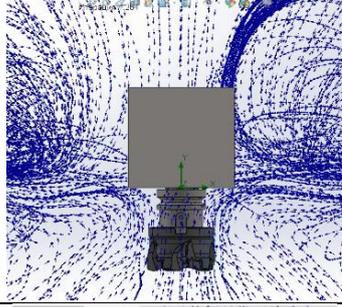
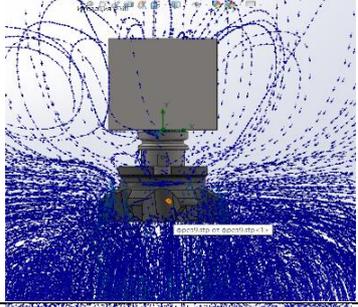
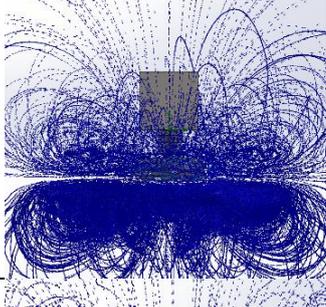
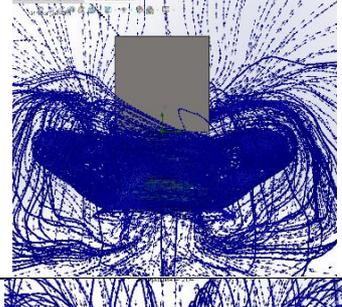
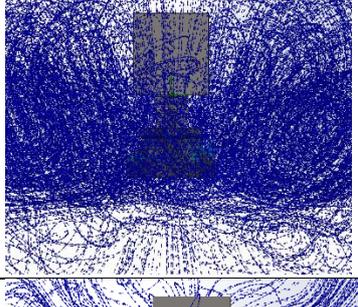
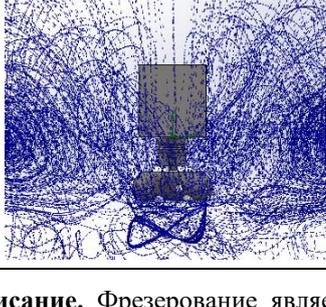
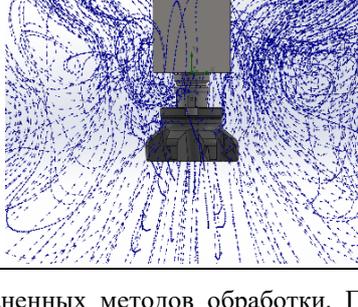
Рисунок 4. – Распространение воздушных потоков

С помощью FlowSimulation мы можем произвести анализ и сравнение воздушных потоков, создаваемых различными фрезами. Для опыта выбираем фрезы: 1-Торцовая фреза, 2-Фреза FF FWX D066-05-22-08, 3-Фреза HP F90AT D50-4-22-19.

Для испытаний выбираем угловые скорости: 1 об/мин, 10 об/мин, 100 об/мин, 1000 об/мин.

Составим таблицу с результатами всех опытов. В таблице показаны направления воздушных потоков в зависимости от угловой скорости.

Таблица 1. – Исследование динамики работы фрез в воздухе

Ско- рость враще- ния	Фреза 1	Фреза FF FWX D066-05-22-08	Фреза HP F90AT D50-4-22-19.
1 об/мин			
10 об/мин			
100 об/мин			
1000 об/мин			

Описание. Фрезерование является одним из самых распространенных методов обработки. По уровню производительности превосходит строгание и в условиях серийного производства уступает лишь наружному протягиванию. Кинематика процессов фрезерования характеризуется быстрым вращением вокруг его оси и медленным движением подачи. Движение подачи при фрезеровании может быть прямолинейно-поступательным, вращательным, винтовым.

Для фрезы 1: при скорости 1 об/мин заметно небольшое количество воздушных потоков, большинство проецируется на плоскость обработки, однако значительная часть хаотически разлетается. При повышении скорости мы видим увеличение количества воздушных потоков, в основном они расположены на проецируемой плоскости, их траектория напоминает тор. Однако при скорости 1000 об/мин мы видим, что большая часть воздушных потоков рассеивается, а у основания образуется отчетливый контур фрезы.

Для фрезы FF FWX D066-05-22-08: при скорости 1 об/мин наблюдается всасывание воздушных потоков, после чего они создают вихри по бокам фрезы. При увеличении скорости до 10 об/мин радиус вихрей увеличивается, при этом количество воздушных потоков изменяется незначительно. Однако при

повышении скорости до 100 об/мин количество воздушных потоков существенно увеличивается. Потоки сильно сконцентрированы вокруг фрезы. Происходит это из-за формы зубьев. Увеличив скорость до 1000 об/мин мы наблюдаем колоссальное количество воздушных потоков. Они образуют огромный тор вокруг фрезы.

Для фрезы HP F90AT D50-4-22-19: при угловой скорости равной 1 об/мин образуется достаточно небольшое количество потоков, относительно остальных фрез, однако после повышения скорости до 10 об/мин картина существенно меняется. Количество воздушных потоков возрастает, направление при этом остается тем же, потоки поднимаются сверху вниз расплываясь в вертикальной плоскости по параболической траектории. Когда скорость увеличивается до 100 об/мин, наблюдается перенаправление воздушных потоков. Происходит поднятие с завихрением, достаточно сконцентрированное, относительно других фрез. Форма завихрений – тор. Однако при увеличении скорости вращения до 1000 об/мин, увеличивается скорость движения самих потоков, тем самым распределяя их менее плотно, но на расстояние существенно большее, чем в остальных случаях, количество потоков, в свою очередь уменьшается.

Заключение. В ходе работы была изучена методика исследования направления, количества, скорости воздушных потоков, аэродинамической обтекаемости торцевых фрез. Все расчеты, создание модели движения воздушных потоков выполнялись в программе SolidWorks с установленным плагином FlowSimulator.

ЛИТЕРАТУРА

1. Методология исследования работоспособности фрезерных и осевых режущих инструментов на основе 3D прототипирования / Н.Н. Попок, С.А. Портянко – Вестник ПГУ, сер. В. Промышленность. Прикладные науки. 2020 г. – с. 29-39.

УДК 656.11

**КООРДИНИРОВАННОЕ УПРАВЛЕНИЕ ДОРОЖНЫМ ДВИЖЕНИЕМ
«ЗЕЛЕНАЯ ВОЛНА» В ГОРОДЕ НОВОПОЛОЦКЕ****Е.С. ЦАРЬКОВА****(Представлено: М.В. СЕМЕНЧЕНКО)**

Аннотация: в данной статье приведены виды светофорного регулирования задачи, которые сможет решить светофорное регулирование, принцип работы светофоров с вызывными устройствами и их преимущества, краткое описание работы контроллера СИДК, используемых для координированного управления дорожным движением «Зеленая волна» в г.Новополоцке.

С каждым годом в Республике Беларусь все больше развивается автомобильный транспорт. Помимо роста количества транспортных средств в собственности у людей, увеличивается их ассортимент, улучшаются характеристики. Из-за возросшего количества транспортных средств в транспортном потоке увеличивается нагрузка на улично-дорожную сеть. Для разгрузки строятся новые магистрали, расширяется улично-дорожная сеть. Если город старой застройки, такой подход не всегда становится возможным из-за особенностей планировки и необходимости значительных финансовых инвестиций. В такой ситуации наиболее доступным решением данной проблемы является совершенствование организации дорожного движения.

В городе Новополоцке на данный момент вводится «Зеленая волна» - один из элементов интеллектуальной транспортной системы. Для реализации режима зеленой волны в Новополоцке на протяжении всей улицы Молодежной были внедрены светофорные объекты и закрыты некоторые нерегулируемые пешеходные переходы. На новых внедренных светофорных объектах для пешеходов были установлены светофоры с вызывными устройствами.

Установка новых контроллеров и «умных» светофорных объектов является реализацией мероприятий Планов зеленого градостроительства и Единого плана устойчивой городской мобильности – проекта «Зеленые города», разработанного для Полоцка и Новополоцка.

Всего можно выделить два вида светофорного регулирования:

1. Постоянное регулирование

1.1. Режим постоянный и не меняется с течением времени

1.2. Режим постоянный, но меняется в зависимости от времени суток или дня недели

2. Адаптивное регулирование. Делится на два типа в зависимости от влияния транспортного потока на регулирование.

2.1. Частично зависящее (меняется либо продолжительность, либо количество, либо последовательность фаз).

Светофорное регулирование позволяет решить следующие основные задачи:

- увеличение уровня безопасности дорожного движения. В случае наличия ДТП, которые можно избежать при введении светофорного регулирования либо, когда другие мероприятия не имеют эффективности (запрет обгона, ограничение скорости движения, нерегулируемые пешеходные переходы)

- улучшение качества движения. Светофорное регулирование защищает от перегрузки участка улиц и дорог, освобождает для приоритетного проезда и ограничивает въезд и выезд. Также светофорные объекты, как правило, меньше по площади (круговое движение).

- улучшение экологической ситуации. Светофорное регулирование позволяет поддерживать максимально возможную скорость движения, сократить количество остановок, двигаться равномерно через несколько пересечений. Данные обстоятельства существенно снижают расход топлива, что в свою очередь снижает уровень шума от транспорта и выделение выхлопных газов.

Данные задачи выполняются максимально эффективно при внедрении «Зеленой волны».

Принцип работы светофоров с вызывными устройствами следующий: если нет пешеходов, желающих перейти дорогу, то транспортный поток проезжает данный участок без снижения скорости и без остановки, для транспортных средств горит зеленый сигнал светофора. Если есть пешеходы, желающие перейти дорогу, то им необходимо нажать кнопку или коснуться вызывного устройства и тогда через небольшой промежуток времени для пешеходов загорается зеленый, а для транспортных средств красный сигнал светофора.

Стоимость установки оборудования и самого оборудования высокая, поэтому нецелесообразно вводить светофорные объекты с кнопками на всех переходах. Наиболее эффективно применять данное оборудование на переходах с пешеходным потоком, явно выраженным в утренние и вечерние часы. Такое решение позволяет:

- улучшить качество движения
- улучшить экологическую ситуацию (водителям не нужно останавливаться при отсутствии пешеходов, в связи с этим меньше выброс выхлопных газов)
- обезопасить пешеходов (на красный сигнал заранее среагировать проще, чем заметить пешехода, который переходит дорогу по нерегулируемому пешеходному переходу)
- сгруппировать пешеходов (На нерегулируемом пешеходном переходе пешеходы, как правило, переходят дорогу по одному. Если одному пешеходу уступили дорогу, следующий старается ускориться и пройти пока стоят автомобили, так иногда образуются цепочки из пешеходов)

В городе Новополюцк внедряется координированное управление дорожным движением «Зеленая волна» со следующим принципом работы: контроллеры СИДК, которые устанавливаются на перекрестках, связываются между собой и обеспечивают включение зеленого сигнала светофоров к моменту подхода групп транспортных средств. Эти мероприятия в первую очередь рассчитаны на общественный транспорт, который сможет передвигаться с минимальными остановками и задержками.

«СИДК» является современным дорожным контроллером, сочетающим в себе новаторские идеи и надежность. Он обладает инновационным дизайном европейского промышленного оборудования, использующего мультипроцессорность мощных процессоров Cortex-M3 с обширным числом интерфейсов. Архитектура безопасности обеспечивает выполнение европейских, а также местных директив и стандартов.

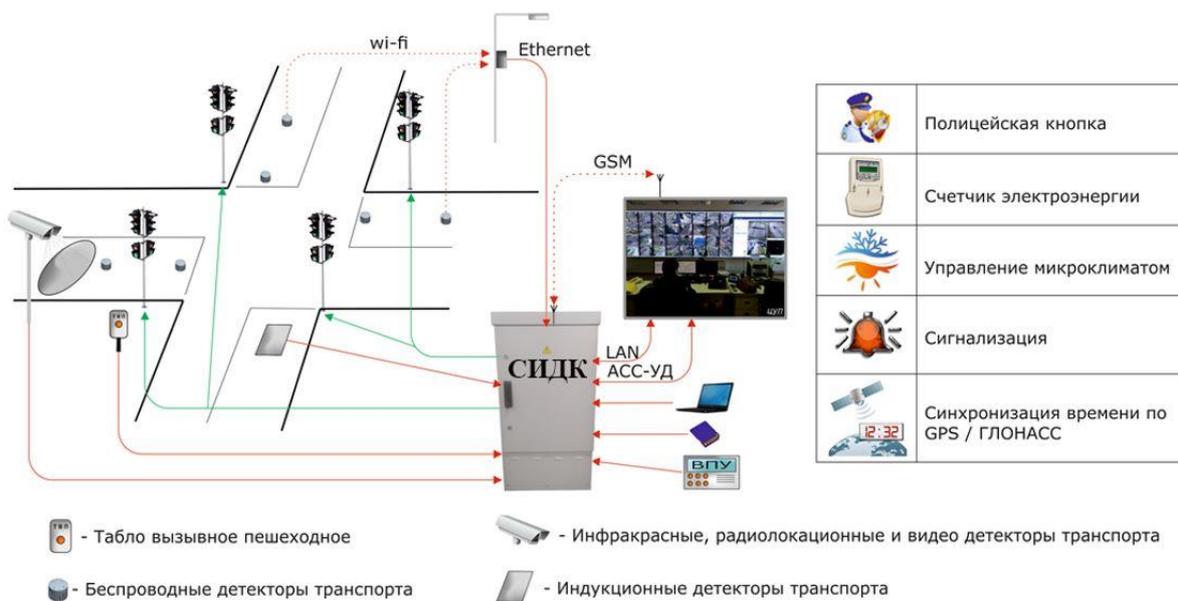


Рисунок 1. – Схема работы контроллера СИДК

Как правило, на всех перекрестках, скоординированных по принципу «зелёная волна», задаётся один и тот же цикл. Зелёный сигнал светофора включается на всех перекрестках не одновременно, а с заранее определенным сдвигом, сдвиги определяются согласно скорости движения на перегонах и расстоянию между перекрестками.

Заключение. Координированное управление дорожным движением «Зеленая волна» повышает безопасность дорожного движения, улучшает качество движения, улучшает экологическую ситуацию за счет уменьшения количества торможений перед светофорами и начала движения со светофора.

Светофорное регулирование с вызывными устройствами позволяет улучшить качество движения, экологическую ситуацию, обезопасить и сгруппировать пешеходов.

ЛИТЕРАТУРА

1. Моделирование движения в интеллектуальной транспортной системе: сб. науч. Ст. / Вестник Самарского государственного аэрокосмического университета, № 2. – Самара, 2004. – 344 с.
2. Зеленая волна в Полоцке и Новополоцке– [Электронный ресурс]. – 2020. – Режим доступа: <http://greencities.by/skoro-zelenaya-volna-v-polotske-i-novopolotske/>-
Дата доступа: 27.01.2020.

УДК 656.11

**ТРАНСПОРТНОЕ МОДЕЛИРОВАНИЕ. ЧЕТЫРЕХШАГОВЫЙ АЛГОРИТМ
ТРАНСПОРТНОГО МОДЕЛИРОВАНИЯ****Е.С. ЦАРЬКОВА***(Представлено: М.В. СЕМЕНЧЕНКО)*

Аннотация: в данной статье указаны статистические данные для Республики Беларусь по росту числа автомобилей, принцип работы четырехшагового алгоритма транспортного моделирования, программные продукты, применяемые в транспортном моделировании.

В связи с возрастающим объемом передвижений пешеходов и транспортных средств существует необходимость в моделировании и дальнейшем прогнозировании транспортной инфраструктуры. Предварительное транспортное моделирование помогает оценить, как будет работать транспортное сооружение до рабочего проектирования и строительства. Также помимо роста объема передвижений растет и количество автомобилей имеющих в собственности граждан (рис.1). Как видно из графика количество легковых автомобилей за 20 лет увеличилось в два раза.



**Рисунок 1. – Наличие транспортных средств в личной собственности граждан
(по данным Министерства внутренних дел Республики Беларусь)**

Всего можно выделить три уровня транспортного моделирования:

- Микроскопический уровень – моделирование нескольких пересечений на уровне транспортного средства.
- Мезоскопический уровень – моделирование сети на уровне транспортного средства.
- Макроскопический уровень – моделирование сети на уровне потоков.

Транспортная модель в целом представляет собой программный комплекс, состоящий из информационных и расчетных блоков. Информационные блоки составляют единую базу данных, предназначенную для хранения и обработки информации, необходимой для прогноза транспортных потоков. Расчетные блоки реализуют алгоритмы решения задач математического программирования, ориентированных на прогноз потребности в передвижениях и расчет реализующих ее транспортных потоков.

Для транспортного моделирования используются следующие программные продукты:

- «PTV Vision® VISUM» - стратегическое планирование дорожной инфраструктуры (Германия);
- «PTV Vision® VISSIM» – микромоделирование – оценка текущего и проектного состояний организации дорожного движения (Германия);
- «СВЕТОФОР» – расчет режима регулирования и задержек светофорных объектов (Иркутск, РФ);
- MXURBAN – реконструкция и ремонт городских улиц и дорог с учетом инженерных сетей;
- «AIMSUN» (Испания), Civil 3D, Credo, Robur, и др.

Данные программные продукты различаются уровнем проектирования и некоторыми характерными особенностями. Наиболее удобным из всех перечисленных является VISUM/VISSIM PTV Vision®. VISUM/VISSIM PTV Vision® - это пакет программ, который использует в своей основе современные математические инструменты, программные продукты в данном пакете интегрированы между собой. Также сильной стороной является успешное применение этих программ для анализа транспортных проектов международных мероприятий.

При построении транспортной модели наиболее трудоемкий и продолжительный по времени этап при построении транспортных моделей – сбор исходных данных.

В конечном итоге алгоритм решает основную задачу в соответствии с существующим спросом и имеющимся транспортным предложением. Таким образом можно выделить два основных этапа: транспортный спрос и транспортное предложение.

При сформированных параметрах транспортного спроса и предложения необходимо усовершенствовать алгоритм распределения транспортного спроса по транспортному предложению и откалибровать модель по собранным натурным данным, которые характеризуют основные параметры транспортного движения.

В первую очередь необходимо сформировать транспортное предложение, далее необходимо создать или рассчитать транспортный спрос.

Транспортное предложение - совокупность данных о транспортной инфраструктуре моделируемой территории. Включает информацию о расположении и протяженности автомобильных дорог и их характеристиках, железнодорожных путей, сетей других видов транспорта, информацию о подвижном составе, расписании движения общественного транспорта, расположении остановок общественного транспорта и организации дорожного движения на перекрестках.

Транспортное предложение - это набор имеющихся в распоряжении городского сообщества средств транспортировки.

Транспортное предложение отображает маршрутную сеть всех видов транспорта:

- Пешеходное движение
- Индивидуальный автомобильный транспорт
- Общественный транспорт
- Другие виды транспорта (водный, велосипедный, железнодорожный)

Транспортное предложение даёт представление о возможностях существующей транспортной инфраструктуры планируемой территории с требуемой степенью точности. Степень детализации данных о транспортном предложении исходя из объекта моделирования и круга задач транспортного планирования.

Исходные данные для создания модели транспортного предложения:

1. Исходные данные для создания модели транспортного предложения индивидуального транспорта.

1.1. Создание геометрии улично-дорожной сети

Транспортное предложение индивидуального транспорта состоит из перекрестков (узлов) и участков дорог между перекрестками (перегонов). Для сбора данных о перекрестках и перегонах необходимы геоинформационные системы (ГИС).

1.2. Исходные данные о существующей организации дорожного движения.

Необходимыми данными в организации дорожного движения являются количество полос движения, пропускная способность перегона и скоростной режим на участке УДС, также для перекрестков необходимо провести учет светофорного регулирования с направлениями и продолжительность фаз.

Источники исходных данных для задания параметров УДС:

- Проекты по организации дорожного движения
- Панорамные видеосъемки

На основе данных о видеосъемках можно посчитать количество полос движения.

1.3. Светофорное регулирование

Исходными данными о направлении движения и фазах светофорного регулирования служат паспорта светофорных объектов и натурный сбор данных о светофорном регулировании.

2. Исходные данные для создания модели транспортного предложения общественного транспорта.

2.1 Информация о действующей маршрутной сети городского пассажирского транспорта общего пользования (ГПТОП).

Маршрутная сеть ГПТОП состоит из:

- Остановочных пунктов и маршрутов движения городского пассажирского транспорта общего пользования.

- Расписание движения маршрутов городского пассажирского транспорта общего пользования.

Транспортный спрос количественно и качественно определяет потребность жителей города в перемещении.

Исходные данные для создания модели транспортного спроса:

- Исходные статистические данные.
 - ✓ Население и трудящееся население.
 - ✓ Рабочие места и рабочие места в сфере услуг.
 - ✓ Количество студентов и учебных мест.
 - ✓ Количество школьников и учебных мест в школах.
- Исходные данные о функционировании транспортной системы.
 - ✓ Данные об интенсивности транспортных потоков.
 - ✓ Данные о пассажирских потоках.
 - ✓ Данные о пассажиропотоке на маршрутах городского пассажирского транспорта общего пользования (ГПТОП);
 - ✓ Данные о пассажирообороте на остановочных пунктах ГПТОП.
- Исходные данные о транспортной подвижности населения.

Расчет прогноза в такой транспортной модели осуществляется по четырехшаговому алгоритму, вследствие чего такие прогнозные транспортные модели называют «четырёхшаговыми». Классическая четырехшаговая модель, позволяющая определить транспортный спрос на основе данных о распределении жителей, рабочих и учебных мест, а также рабочих мест в сфере обслуживания по транспортным районам города. В создании и последующей работе такой модели можно выделить четыре этапа (шага):

1. Генерация спроса (Trip Generation)
2. Распределение спроса (Trip Distribution)
3. Выбор режима (Mode Choice)
4. Перераспределение (Traffic Assignment)

Генерация спроса. При построении модели транспортного спроса необходимо исследуемую территорию представить из набора областей, они называются транспортными районами. Каждому району требуется присвоить свой номер и уникальную конфигурацию, которая повторяет элементы городской среды.

Распределение спроса (Trip Distribution). Исходными данными для решения задачи распределения спроса являются

- матрицы затрат,
- потоки источника/цели (истоки/стоки) для районов по слоям спроса,
- необходимое для калибровки модели транспортного спроса среднее время поездки.

На первом этапе моделирования определяются величины корреспонденций различных типов между транспортными районами, на которые разбита моделируемая территория. Для расчета используется модель максимизации энтропии, позволяющая определять наиболее вероятные значения корреспонденций.

Подход в расчете матриц корреспонденций основан на эквивалентности трех теоретических концепций, которые используются для обоснования методов моделирования распределения транспортного спроса:

- гравитационного,
- энтропийного,
- байесовского подходов.

Эквивалентность означает, что три указанных подхода приводят к одному и тому же практическому итерационному методу расчета элементов матрицы корреспонденций.

Выбор режима (Mode Choice). После получения матриц корреспонденций по слоям спроса, необходимо разделить эти матрицы по режимам движения. Режим движения будет определять способ реализации корреспонденции – на индивидуальном или на общественном транспорте.

Перераспределение (Assignment). После того, как получены матрицы корреспонденций по слоям спроса, они разделяются на виды транспорта, которыми они будут реализованы. Необходимо провести перераспределение полученных матриц корреспонденций по транспортному предложению для выбора того или иного пути.

Заключение. Транспортное моделирование позволяет прогнозировать ситуацию по разным параметрам. Итогом выполнения четырехшагового алгоритма после перераспределения являются матрицы затрат для общественного и индивидуального транспорта, далее итерация возвращает к пересчету взвешенной матрицы затрат и распределению спроса. Распределение спроса и является конечной задачей транспортного моделирования.

Итогом результата работы является рассчитанные интенсивности транспортных потоков индивидуального транспорта, пассажиропотоки на общественном транспорте, скорости движения транспортных потоков и т.д.

Все значения связаны с улично-дорожной сетью города и в итоге их можно использовать для принятия управленческих решений в сфере транспортного планирования и организации дорожного движения.

ЛИТЕРАТУРА

1. Якимов М.Р. Транспортное планирование: Создание транспортных моделей городов / М.Р. Якимов. – М.: Логос, 2013. – 188 с.
2. Общий алгоритм работы четырехшаговой транспортной модели: сб. науч. Ст. /Вестник ИрГТУ №1 (48). – Иркутск, 2011. – 343.
3. Постнов С.Н. Методика построения матрицы корреспонденций новых планируемых районов // Международная конференция «Современные технологии стратегического и оперативного транспортного планирования PTV Vision®»: тез. докл. 6-ой МК. С-П., 2010. С. 19-20.

УДК 621.91.01

**АНАЛИЗ СПОСОБОВ ОБРАБОТКИ СФЕРИЧЕСКИХ ПОВЕРХНОСТЕЙ ДЕТАЛЕЙ
НА ТОКАРНЫХ СТАНКАХ С ЧПУ****М.М. ШАУРКО, А.А. ДЁМКИН***(Представлено: д-р техн. наук Н.Н. ПОПОК, канд. техн. наук, доц. Р.С. ХМЕЛЬНИЦКИЙ)*

Приведен анализ способов обработки неполных сферических поверхностей деталей на токарных станках с ЧПУ. Рассмотрены кинематические схемы резания данных способов. Выявлены их преимущества и недостатки.

В машиностроении существуют большое количество деталей, в которых встречаются следующие разновидности сферических поверхностей:

- сфера (др.-греч. σφαῖρα – мяч, шар) – геометрическое место точек в пространстве, равноудаленных от центра сферы (например, шарик подшипника);
- сферический сегмент – часть сферы, отсеченная от нее какой-нибудь плоскостью (например, шар рукоятки, линза);
- сферические пояс – часть сферы, заключенная между двумя параллельными секущими плоскостями (например, пробка шарового крана).

Сферические сегмент и пояс иногда называют неполными сферическими поверхностями.

Анализ способов обработки неполных сферических поверхностей деталей показывает, что зачастую приходится использовать специальные станки, специальное технологическое оснащение или специальный режущий инструмент [2]. В настоящее время при обработке деталей в металлообработке все чаще используются станки с числовым программным управлением, которые позволяют за счет движений получать сложные геометрические формы деталей.

Рассмотрим более подробно способы обработки неполных сферических поверхностей деталей, реализуемых на токарных станках с ЧПУ (таблица).

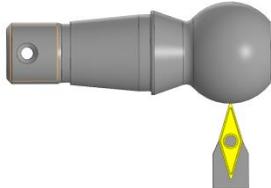
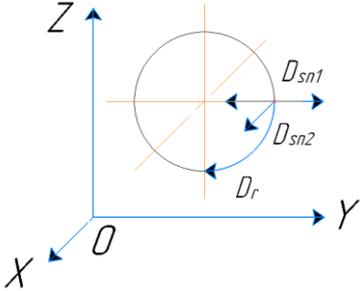
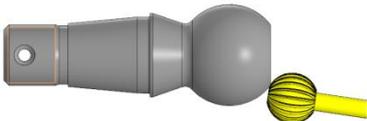
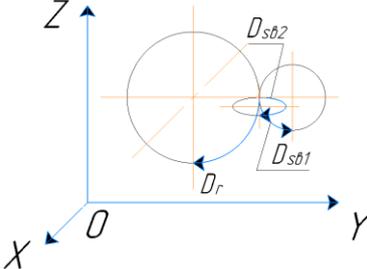
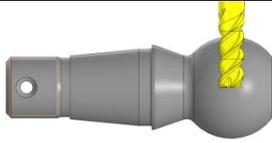
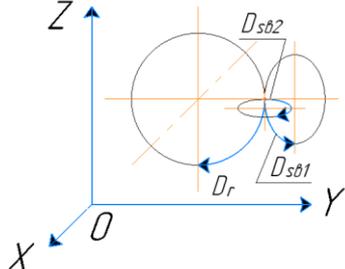
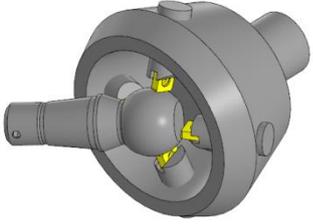
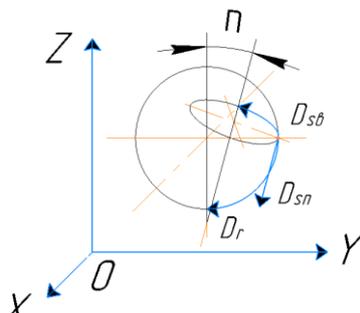
Точение резцом (см. таблицу 1, п.1) наиболее распространенный способ обработки неполных сферических поверхностей на токарных станках с ЧПУ. Кинематическая схема обработки неполной сферической поверхности состоит из трех элементов: главного вращательном движении заготовки D_g и двух вспомогательных поступательных движений D_{s1} и D_{s2} режущего инструмента. Анализируя кинематическую схему резания видно, что все эти движения полностью совпадают с движениями, реализуемым на двухосевых токарных станках с ЧПУ: вращение шпинделя и перемещение револьверной головки по осям X и Z станка. В качестве преимуществ данного способа можно отметить, что для его реализации достаточно использование токарного резца для контурного точения. Недостатком данного способа является приближение скорости резания к нулю у полюса сферической поверхности.

В настоящее время на токарных станках с ЧПУ помимо обработки тел вращения резцами применяют обработку фрезами [3, 4]. Далее рассмотрим возможности обработки неполных сферических поверхностей деталей фрезами.

Фрезерование сферической фрезой (см. таблица п.2). Кинематическая схема фрезерования сферической фрезой состоит из трех элементов: главного вращательном движении шаровой фрезы D_g , вспомогательного вращательного движения заготовки D_{sv1} и перемещения режущего инструмента по дуге D_{sv2} . Причем следует отметить, что перемещение режущего инструмента по дуге D_{sv2} реализуется на станке с ЧПУ при помощи круговой интерполяции за счет двух согласованных поступательных движений по осям X и Z . И при разложении движения D_{sv2} на два поступательных движения D_{sp1} и D_{sp2} кинематическая схема резания примет вид четырехэлементной. Данная кинематическая схема также легко реализуется на токарном станке с ЧПУ, оснащенным револьверной головкой с возможностью установки приводного инструмента. Согласно рекомендациям [3] фирмы Sandvik Coromant при работе фрезой со сферическим концом наиболее критической зоной режущей кромки является центр инструмента, где скорость резания близка к нулю, что неблагоприятно для процесса резания. А при обработке сферической поверхности детали ее скорость вращения будет также приближаться к нулю на полюсе. Поэтому рекомендуется наклонять ось вращения режущего инструмента по отношению к оси заготовки на 10–15 градусов для смещения зоны резания от центра инструмента, что приведет к увеличению минимальной скорости резания, стойкости режущего инструмента и улучшению стружкообразования, и, как следствие, улучшению качества обработанной поверхности. К недостаткам данного способа можно отнести высокую стоимость режущего инструмента, а также необходимость применения специальной поворотной приводной головки.

Фрезерование концевой фрезой (см. таблица п.3) отличается от предыдущего способа тем, что ось главного вращательном движении концевой фрезы D_r перпендикулярна оси вспомогательного вращательного движения заготовки D_{sb1} . Это отчетливо видно на кинематической схеме резания. Так как обработка проводится периферией концевой фрезы скорость резания в любой точке, обрабатываемой неполной сферической поверхности остается постоянной. Что благоприятно сказывается на процессе резания и качестве обработанной поверхности. Из недостатков данного способа следует отметить необходимость применения станка с радиальной приводной головкой со смещением для установки режущего инструмента с возможностью перемещения револьверной головки вдоль оси Y .

Таблица 1. – Способы обработки сферических поверхностей деталей на токарных станках и ЧПУ

№ п/п	Способ обработки	Процесс обработки	Кинематическая схема резания
1	2	3	4
1	Точение резцом		
2	Фрезерование сферической фрезой		
3	Фрезерование концевой фрезой		
4	Скоростная комбинированная обработка специальным инструментом		

Скоростная комбинированная обработка специальным инструментом (см. таблица п.3). Кинематической схемы резания данного способа обработки может быть двух- (два вращательных движения) и трех- (два вращательных и поступательное движения) элементной, реализуемой на трех переходах операции формообразования неполной сферической поверхности [5]. На первом переходе происходит врезание вращающегося инструмента

во вращающуюся заготовку (черновая обработка), причем направление поступательного движения врезания (подача) D_{sp} происходит по оси главного вращательного движения инструмента D_r . Ось вращательного движения подачи заготовки D_{sv} направлена под углом η к оси главного вращательного движения. На втором переходе происходит выхаживание поверхности (получистовая обработка) и реализуется двухэлементная кинематическая схема резания с главным вращательным движением инструмента D_r и вращательным движением подачи заготовки D_{sv} . На третьем переходе, когда лезвия инструмента выходят на требуемый размер сферы, реализуется выглаживание (чистовая обработка) поверхности сферы. На этих переходах наряду со встречным движением D_r и D_{sv} возможно осуществление их попутного вращения, что будет способствовать более качественному выглаживанию поверхности [6]. Скоростная обработка неполных сферических поверхностей деталей машин первоначально была реализована на универсально-заточном станке модели 3М642, но анализируя кинематическую схему резания, видно, что ее можно реализовать на токарном станке с ЧПУ при помощи поворотной приводной головки. К недостаткам данного способа можно отнести необходимость точной настройки угла поворота приводной головки.

На основе анализа кинематических схем резания рассмотренных способов обработки неполных сферических поверхностей деталей машин можно сделать вывод, что они все реализуемы на токарных станках с ЧПУ. Выбор же конкретного способа обработки будет зависеть, как от конфигурации неполной сферической поверхности, так и от возможностей токарного станка и технологического оснащения.

ЛИТЕРАТУРА

1. Иллюстрированный определитель деталей общемашиностроительного применения: руководящий техн. материал: кл. 40 и 50 общесоюз. классификатора пром. и с.-х. продукции. – Москва : Изд-во стандартов, 1976. – 238 с.
2. Анализ способов обработки сферических поверхностей / Н.Н. Попок [и др.] // Вестн. Полоц. гос. ун-та: Сер. В. Прикладные науки. Промышленность. – 2006. – № 12. – С. 42–45.
3. Sandvik Coromant [Электронный ресурс] // Старт/Техническая информация/Фрезерование/Профильное фрезерование: <https://www.sandvik.coromant.com/ru-ru/knowledge/milling/pages/profile-milling.aspx>. – Дата доступа: 10.09.2020
4. Анализ особенностей высокоскоростного точения фрезерованием / А.М. Марков [и др.] // Актуальные проблемы в машиностроении: Том 4. Инновационные технологии в машиностроении. – 2017. – № 3. – С. 39–44.
5. Технологическое обеспечение обработки неполных сферических поверхностей деталей машин / Н.Н. Попок, Р.С. Хмельницкий, В.С. Анисимов // Вестн. Полоц. гос. ун-та. Серия В, Промышленность. Прикладные науки. – 2018. – № 3. – С. 54–60.
6. Кинематическое и технологическое обеспечение формирования регулярного микрорельефа сферических поверхностей деталей комбинированной обработкой резанием / Н.Н. Попок, Р.С. Хмельницкий, В.С. Анисимов, Г.И. Гвоздь, В.А. Кукареко // Вестн. Полоц. гос. ун-та. Сер. В, Прикладные науки. Промышленность. – 2017. – № 11. – С. 16–25.

УДК 629.331

ТЕХНОЛОГИИ ДИАГНОСТИРОВАНИЯ ДВИГАТЕЛЕЙ ВНУТРЕННЕГО СГОРАНИЯ

Д.С. ШЕВЧЕНКО*(Представлено: канд. техн. наук, доц. Т.В. ВИГЕРИНА)*

В статье приведен анализ методов диагностирования двигателей внутреннего сгорания и их классификация в зависимости от характера и физической сущности распознаваемых признаков и измеряемых параметров технического состояния. Рассмотрена технология поиска неисправностей по схеме поэлементного диагностирования двигателя. Приведен вывод о производительности и качестве выполняемых диагностических работ путем традиционных методов диагностирования.

Введение. Основным агрегатом автомобиля, на долю которого приходится наибольшее число отказов, является двигатель внутреннего сгорания (ДВС). Надежность работы двигателей зависит не только от их конструкции, технологии изготовления, условий эксплуатации автомобилей, но в большой степени от организации и качества их обслуживания. Совершенство любого метода обслуживания и ремонта определяется тем, насколько полно он обеспечивает взаимодействие между объективно существующим процессом изменения технического состояния объекта и процессом его технической эксплуатации. Традиционный планово-предупредительный метод обслуживания и ремонта, основанный на выполнении профилактических работ определенных объемов через заранее запланированные интервалы времени или наработки независимо от состояния систем и деталей, обеспечивает слабое взаимодействие между указанными процессами. Более тесную связь между ними обеспечивают методы обслуживания и ремонта по состоянию. Основным принципом данных является принцип предупреждения отказов систем автомобиля и их отдельных наиболее важных узлов и деталей при обеспечении максимально возможной их наработки до замены. Методы обслуживания и ремонта по состоянию предусматривают непрерывный или периодический контроль и измерение параметров, определяющих техническое состояние функциональных систем и узлов, то есть осуществление непрерывного или периодического диагностирования данных объектов. При проведении технического обслуживания легковых автомобилей на диагностирование отводится около 15 % от общей трудоемкости работ. Внешние признаки различных неисправностей ДВС могут быть одинаковы, для более точного определения технического состояния механизмов и систем двигателей используется ряд диагностических технологий и стандартного оборудования, позволяющих при последовательной проверке выявлять неисправность. [1,2]

Основными причинами изменения технического состояния элементов являются: конструктивные; технологические; эксплуатационные.

Исходная неравномерность рабочих процессов, обусловленная конструктивными и технологическими факторами, усиливается эксплуатационными факторами.

Низкая температура пуска, низкое качество топлива и масла, низкая очистка воздуха, низкое качество регулировочных операций – все эти факторы имеют место в эксплуатации и усугубляют процесс нарушения технического состояния ДВС.

Состояние отремонтированной техники характеризуется этими же факторами. Однако они проявляются в большей степени, что определяется использованием в процессе ремонта деталей с уже накопленными различиями. Из этого следует, что для машин с различным исходным состоянием во время эксплуатации потребуется разное восстановительно-профилактическое вмешательство через неодинаковое время. Оптимальным будет являться индивидуальное обслуживание каждого экземпляра с целью восстановления как функциональных, так и ресурсных параметров.

На практике не всегда удается провести замеры, непосредственно характеризующие техническое состояние машины. В таких случаях используют косвенные показатели. В современных сложных машинах предусмотрена возможность доступа к местам контроля или имеются приспособления для монтажа контролирующих устройств. Новые модели оснащены встроенными преобразователями, соединенными либо с сигнализирующими приборами на панели управления, либо при помощи разъема на период контроля – со специальной аппаратурой.

Диагностирование машин, проводимое с использованием внешних и встроенных средств контроля, позволяет определять техническое состояние агрегатов, механизмов и систем машины без их разборки, прогнозировать сроки службы узлов, фактически управлять их техническим состоянием, назначая соответствующие предупредительные работы и выполняя их в процессе технического обслуживания и ремонта. Это снижает время простоя машины, обеспечивает значительную экономию средств на ее обслуживании.

живание и ремонт. Выполнение только действительно необходимых операций по ремонту и регулированию сокращает расход запасных частей и топливо-смазочных материалов. Так, своевременное обнаружение и устранение значительных неисправностей в системах питания или зажигания двигателя, агрегатов трансмиссии или ходовой части улучшает на 5...10% топливо-экономические показатели, повышает безопасность эксплуатации машины. [3]

Методы диагностирования классифицируют в зависимости от характера и физической сущности распознаваемых признаков и измеряемых параметров технического состояния объектов. Обслуживанием выявляют места и характер ненормальных стуков, шумов, перебоев в работе двигателя, отказов в трансмиссии и ходовой системе (по скрежету и шуму), неплотность (по шуму прерывающегося воздуха) и т.п. Осмотром устанавливают места подтекания воды, масла, топлива, цвет отработавших газов, дымления из сапуна, биение вращающихся частей, натяжение цепных передач и т.п. Осязанием определяют места и степень ненормального нагрева, биения, вибрации деталей, вязкость, липкость жидкости и т.п.

Инструментальные, или объективные, методы применяют для измерения и контроля всех параметров технического состояния, используя при этом диагностические средства. По характеру измерения параметров методы диагностирования машин подразделяются на прямые и косвенные.

Прямые методы основаны на измерении структурных параметров технического состояния непосредственно прямым измерением: зазоров в подшипниках, прогиба цепных и ременных передач, размеров деталей и т.д. Из-за своей простоты прямые методы нашли широкое практическое применение при контроле и регулировании механизмов и устройств расположенных снаружи агрегатов машины, доступных и удобных для проверки, не требующих разборки механизмов (приводные механизмы, ходовая часть, рулевое управление, тормозная система и др.). Применение прямых методов измерения находящихся внутри агрегатов (ЦПГ, подшипниковые узлы коленчатого вала двигателя) ограничено большой трудоемкостью и необходимостью разборки агрегата.

Косвенные методы диагностирования основаны на определении структурных параметров технического состояния агрегатов машин по косвенным (диагностическим) параметрам при установке датчика или диагностического устройства снаружи агрегата без разборки механизмов машины. Косвенные методы основываются на измерении непосредственно физических величин, характеризующих техническое состояние механизмов, систем и агрегатов машин: давление, перепад давления, температура, перепад температуры рабочего тела в системе, расход газа, топлива, масла параметры вибрации составных частей машин, ускорение при разгоне двигателя и др. Многие методы осуществляются на основе преобразования механических величин в электрические с применением электронных диагностических приборов и установок. [4]

Методы диагностирования технического состояния автомобилей, агрегатов характеризуются физической сущностью и способом измерения диагностических параметров, наиболее приемлемых для использования в зависимости от задачи диагностирования. В настоящее время выделяют три основные группы методов диагностирования.

Методы первой группы базируются на имитации скоростных и нагрузочных режимов работы автомобиля, определении при заданных условиях выходных параметров и сравнении их количественных значений с эталонными. Диагностирование проводится с использованием стендов с беговыми барабанами или непосредственно в процессе работы автомобиля. Методы широко применяются для общей оценки технического состояния автомобилей и агрегатов.

К методам диагностирования по параметрам сопутствующих процессов относятся:

– методы диагностирования по герметичности рабочих объемов. Сущность процесса диагностирования заключается в создании в контролируемом объеме избыточного давления (разряжения) и в оценке интенсивности их падения. Этим методом диагностируются цилиндропоршневая группа двигателя, пневматические приводы тормозов и др.;

– тепловой метод, заключающийся в определении параметров, характеризующих количество тепла, выделяемого в результате протекания процессов сгорания, работы сил трения при заданных скоростном и нагрузочном режимах. Такими параметрами могут быть температура нагрева, скорость ее изменения. Метод может применяться для диагностирования двигателя, агрегатов трансмиссии, подшипниковых узлов, однако широкого применения на автотранспорте пока не нашел;

– методы диагностирования узлов, систем по параметрам колебательных процессов широко используются при создании средств технического диагностирования автомобилей и их можно разделить на три подвиды: методы, оценивающие колебания напряжения в электрических цепях (на этой основе созданы мотор-тестеры), по параметрам виброакустических сигналов, получаемых при работе зубчатых зацеплений, клапанных механизмов, подшипников и т. д.), по параметрам, оценивающим пульсацию давления в трубопроводах (на этой основе созданы дизель-тестеры для диагностирования дизельной топливной аппаратуры);

– методы, оценивающие состояние узлов и агрегатов по физико-химическому составу отработавших эксплуатационных материалов. Например, простейший экспресс-анализ отработанного масла на загрязне-

ние, спектральный анализ проб масел, в результате проведения которого по наличию и концентрации различных химических элементов в масле можно поставить диагноз работоспособности отдельных узлов и сопряжений агрегата. Если в пробе картерного масла двигателя имеется высокое содержание свинца, это говорит об износе вкладышей шатунных и коренных подшипников, если высокое содержание железа - об износе гильз цилиндров, если высокое содержание кремния - о засорении воздушного фильтра и т. д.

Третья группа методов основывается на объективной оценке геометрических параметров (зазор, люфт, свободный ход, смещение и т. д.). Метод применим, когда указанные параметры легкодоступны для непосредственного измерения.

В настоящее время проводятся исследования по разработке новых и совершенствованию имеющихся методов диагностирования применительно к усложняющимся конструкциям автомобилей, изменению элементной базы микроэлектроники и микропроцессорной техники.

Один и тот же диагностический признак чаще всего может быть установлен с помощью нескольких методов диагностирования. Вопрос выбора наиболее целесообразного из них в каждом конкретном случае решается с учетом: уровня информативности и точности, степени универсальности метода диагностирования, трудоемкости диагностирования, различных организационно-экономических факторов. [5]

Во время работы двигателя подвергаются износу, поэтому после определенного пробега возникает потребность в ремонте или замене деталей. Износ двигателей проявляется, в первую очередь, в нарушении герметичности поршневых колец, клапанов и уплотнении цилиндров. При этом снижается давление сжатия, ухудшается наполнение цилиндров, возрастает расход масла, снижаются разрежение во впускном трубопроводе и мощность, повышается расход топлива, изменяется объемный КПД, крутящий момент и т.д.

Технология поиска неисправностей осуществляется в три этапа (рисунок 1).



Рисунок 1. – Схема поэтапного диагностирования дизельного двигателя.

На первом этапе выполняются подготовительные операции, без которых нельзя проверять техническое состояние двигателя по комплексным параметрам. На втором этапе осуществляется проверка двигателя по комплексным параметрам. Если они соответствуют допустимым значениям, диагностирование прекращается. Третий этап – поиск неисправностей; необходим в том случае, когда все или один из комплексных параметров не соответствуют допустимым значениям. Проверка технического состояния систем и узлов осуществляется в определенной последовательности, которая зависит от комбинаций количественных характеристик комплексных показателей. После регулировочных работ или работ, связанных с заменой детали, повторяют контроль комплексных показателей. В зависимости от их значений диагностирование продолжается или прекращается. По количественным значениям комплексных параметров можно сделать заключение об исправности двигателя. Для определения характера неисправности и ее местонахождения используются частные параметры: разрежение во впускном трубопроводе; герметичность надпоршневого пространства цилиндров двигателя (величина компрессии в цилиндрах; относительная утечка воздуха, подаваемого в цилиндры под давлением; разрежение); количество газов, прорывающихся в картер двигателя; интенсивность выгорания масла; характер и место шумов и стуков, определяемых стетоскопом или фонендоскопом; акустические сигналы, анализируемые специальной виброакустической аппаратурой; количество оксида углерода (СО) и углеводородов (С_nH_m) в отработавших газах; давление масла в

системе смазки и др. С помощью перечисленных параметров можно оценить техническое состояние цилиндропоршневой группы (ЦПГ), кривошипно-шатунного и газораспределительного механизмов, степень исправности систем зажигания и питания. Механизмы и узлы двигателя функционально взаимосвязаны, параметры одного из них могут характеризовать состояние другого. Однако они неодинаково влияют на работу двигателя, поэтому очень важно знать рациональную последовательность определения частных параметров в процессе диагностирования, чтобы быстрее найти неисправность. [6]

В процессе поэлементного диагностирования можно довести до нормы значения некоторых частных параметров путем регулировки соответствующих систем и механизмов двигателя. Выполняя регулировки в рациональной последовательности и продолжая измерять другие частные параметры, можно определить неисправности, вызванные износами или поломками деталей узлов и механизмов.

Диагностирование всех систем и механизмов бензинового двигателя внутреннего сгорания является процессом с высокой трудоемкостью при использовании традиционных методик диагностирования, которые предполагают последовательную проверку всех систем и механизмов двигателя. Проведенный анализ показал, что существенное увеличение производительности технического обслуживания легковых автомобилей, качества выполняемых работ, возможны при внедрении более совершенных технологий диагностирования.

ЛИТЕРАТУРА

1. Кузнецов, А.С. Техническое обслуживание и диагностика двигателей внутреннего сгорания: учеб. Пособие / А.С. Кузнецов. – М.: Изд. Центр «Академия», 2011. – 80с.
2. Хабардин, В.Н. Повышение эффективности использования машинотракторного парка путем диагностирования на основе новых методов и средств / В.Н. Хабардин // Вестник ИрГСХА. – 2007. - №29. – с.98-113.
3. Крамаренко, Г.В. Техническая эксплуатация автомобилей: учебник для вузов / под ред. Г.В. Крамаренко. – М.: Транспорт, 1983. -488с.
4. Карташевич, А.Н. Диагностирование автомобилей. Практикум: учебное пособие / А.Н. Карташевич, В.А. Белоусов, А.А. Радушко и др.; под ред. А.Н. Карташевича – Минск: Новое знание. – М.: Инфра-М, 2011. – 208с.
5. Дмитриевский, А.В. Автомобильные бензиновые двигатели / А.В. Дмитриевский. М.: ООО «Издательство АСТ», 2003. – 80с.
6. Шадюль, Р. Методология диагностирования машин на всех этапах их существования: монография / М. Воропай, А. Карташевич. Быдгощ, 2003. 325 с.

УДК 629.331

**СНИЖЕНИЕ ТРУДОЕМКОСТИ ДИАГНОСТИРОВАНИЯ
ДВИГАТЕЛЕЙ ВНУТРЕННЕГО СГОРАНИЯ****Д.С. ШЕВЧЕНКО***(Представлено: канд. техн. наук, доц. Т.В. ВИГЕРИНА)*

Одни и те же признаки неисправности автомобиля могут случаться по разным причинам. Поэтому в некоторых случаях диагностика традиционными методами может занимать до несколько дней. Это необходимо, чтобы выявить все проблемы и правильно произвести ремонт. Снизить количество затраченного времени и трудоемкость выполнения диагностических работ позволяет методика диагностирования двигателей с использованием современных информационных технологий.

Для определения технического состояния цилиндропоршневой группы, кривошипно-шатунного и газораспределительного механизмов двигателя существует несколько методов, большинство которых реализуется с помощью приборов и диагностических стендов.

На практике наиболее распространенным методом определения технического состояния ЦПГ является метод определения давления в цилиндрах в конце такта сжатия, т.е. определение компрессии. Основные признаки снижения компрессии: затрудненный пуск двигателя, что особенно характерно для бензиновых двигателей; неустойчивая работа двигателя во всех режимах; отказ одного или нескольких цилиндров; хлопки во впускной или выпускной тракт; увеличение расхода топлива; появления дыма в отработавших газах.

Причины, приводящие к снижению компрессии: износ стенок цилиндра и компрессионных колец; закоксовывание или разрушение поршневых колец; сквозное прогорание или частичное разрушение поршня; сквозная трещина в головке блока цилиндров; коробление посадочной поверхности головки блока; неправильная регулировка клапанов; повреждение гидротолкателей; износ направляющих втулок; деформация стержня клапана; прогорание клапана; нагар на стенках камеры сгорания и на днище поршня. Компрессию измеряют с помощью компрессометра или компрессографа, которые представляют собой манометр с рукояткой, трубкой, наконечником и золотниковым устройством. [1]

Основные недостатки проверки технического состояния двигателя с использованием величины компрессии следующие:

- зависимость показаний частоты вращения коленчатого вала двигателя от режима нагрузки. Частота вращения при прокрутке стартером (250...350 об/мин) существенно отличается даже от частоты вращения в режиме холостого хода (700...900 об/мин), не говоря уже о режимах частичных и полных нагрузок;
- неудовлетворительная информативность теста для выявления не только проблемных цилиндров, но и первопричины недостаточного давления;
- невозможность проведения теста на демонтированном или частично разобранном двигателе либо двигателе с неработающим стартером.

Метод проверки состояния ЦПГ пневмотестером позволяет устранить недостатки метода определения состояния ЦПГ с помощью компрессометра.

При использовании пневмотестера: во-первых, анализируется непосредственно герметичность надпоршневого пространства (частота вращения коленчатого вала не оказывает никакого влияния на измерения, так как он при проведении теста неподвижен); во-вторых, имеется возможность локализации неисправностей; в-третьих, возможно проведение теста на снятом или частично разобранном двигателе либо на двигателе с неработающим стартером.

Герметичность надпоршневого пространства цилиндра двигателя проверяют путем измерения в цилиндре давления воздуха, подаваемого через калиброванное отверстие корундовой втулки. При утечке воздуха из надпоршневого пространства наблюдается снижение давления, которое пропорционально потерям воздуха через неплотности в цилиндре. Так как давление воздуха до корундовой втулки поддерживается редуктором на постоянном уровне (0,16 МПа), снижение давления, наблюдаемое по манометру, будет характеризовать износ ЦПГ, состояние клапанов и прокладки головки блока. Герметичность надпоршневого пространства бензиновых и дизельных двигателей считается удовлетворительной, если давление будет не ниже 0,11 МПа. [2]

За последние годы в сферу диагностирования двигателей внедряются нестандартные методики, которые призваны повысить эффективность и снизить трудоемкость работ. Диагностирование осуществляется при помощи мотортестера USB AUTOSCOPE IV, с целью выявления неисправностей в различных электронных системах автомобиля, в системах зажигания, а также для диагностики системы газораспределения и механики двигателя

Скрипт Rx предназначен для автоматического анализа графика давления в цилиндре без воспламенения. Генерирует несколько вкладок отчёта с рядом дополнительных параметров и характеристик двигателя и системы управления. Рассчитываются пневматические и геометрические характеристики цилиндра, перечень обнаруженных отклонений выводится в форме текстовых сообщений. Для улучшения удобства и наглядности исследования фаз газораспределения, график преобразуется скриптом в диаграмму количества газа в цилиндре и выводится в двух формах представления. Кроме того, строится подробная диаграмма циклового наполнения цилиндра на такте впуска, характеризующая работу всего впускного тракта двигателя. Выводится диаграмма затрат энергии на очистку цилиндра от отработавших газов. При наличии дополнительного сигнала синхронизации с искрой зажигания, строится диаграмма угла опережения зажигания (УОЗ). Для получения графика давления в цилиндре применяется датчик, преобразующий давление в напряжение, который может использоваться с удлинителем.

Классическим инструментом для оценки состояния цилиндро-поршневой группы является компрессометр, предназначенный для измерения пикового давления в цилиндре при прокрутке двигателя стартером, ещё называемого «компрессией». Это величина комплексная, и, соответственно, зависит от потерь газа через неплотности в цилиндре, от степени сжатия и др. Но ухудшение показаний компрессометра по одному из цилиндров зачастую трактуется именно как ухудшение его герметичности. Скрипт Rx позволяет отличить неплотность цилиндра от низкой степени сжатия, так как рассчитывает отдельно величину потерь газа, и отдельно величину степени сжатия. [4,5]

В результате выполнения анализа скриптом, его результаты были представлены в окне программы «USB Oscilloscope» в нескольких вкладках отчёта.

На вкладке «Количество» отображается диаграмма количества газа в цилиндре в зависимости от положения поршня и от такта работы цилиндра.

Для оснащённых ременным приводом распределительного вала двигателей, где и впускные и выпускные клапана цилиндра обслуживаются одним распределительным валом, ошибка установки приводного ремня на 1 зуб в сторону "позже" приводит к запаздыванию моментов открытия и закрытия клапанов на $\sim 15^\circ$. На диаграмме количества газа в цилиндре это отражается как смещение момента закрытия впускного клапана на $\sim 15^\circ$ влево, а открытия выпускного – на $\sim 15^\circ$ вправо. При этом получаются, что характерные точки отдаляются друг от друга на $\sim 30^\circ$.

Дополнительным признаком поздней установки фаз газораспределения может служить характерное искажение формы левой части красного и зелёного фрагментов диаграммы в районе фазы перекрытия клапанов.

Если же распределительный вал установлен на 1 зуб приводного ремня в сторону «раньше», то это приводит к опережению моментов открытия и закрытия клапанов. На диаграмме это отражается как смещение момента закрытия впускного клапана вправо, а открытия выпускного – влево. Характерные точки при этом отдаляются друг от друга на те же $\sim 30^\circ$.

Характерное искажение формы левой части красного и зелёного фрагментов диаграммы количества газа в цилиндре возникает в результате ранней установки фаз газораспределения. При ранней установке фаз газораспределения, красный и зелёный фрагменты диаграммы в районе фазы перекрытия клапанов друг на друга не накладываются вовсе, и располагаются под заметным углом.

Вкладка «Фазы газораспределения».

В этой вкладке отображается та же диаграмма количества газа в цилиндре, что и во вкладке «Количество», но уже в зависимости от угла поворота коленчатого вала. Количество газа в цилиндре выражается как расстояние от центра диаграммы: чем больше в цилиндре газа, тем дальше от центра расположен фрагмент диаграммы.

Для оснащённых ременным приводом распределительного вала двигателей, где и впускные и выпускные клапана цилиндра обслуживаются одним распределительным валом, ошибка установки приводного ремня на 1 зуб в сторону «позже» приводит к «повороту» диаграммы фаз по часовой стрелке на $\sim 15^\circ$. При этом фазы уже получают несимметричными относительно ВМТ.

Дополнительным признаком поздней установки фаз газораспределения может служить характерное искажение формы диаграммы в центре.

Вкладка «Опережение». Если дополнительно к графику давления в цилиндре был записан ещё и сигнал синхронизации с искрой зажигания, то скрипт Rx строит диаграмму угла опережения зажигания в зависимости от оборотов и нагрузки на двигатель.

Вкладка снабжена маской, которая отражает оптимальные зоны расположения фрагментов диаграммы соответствующих цветов. На рисунке 1 отчёт скрипта Rx показывает пример, когда красный и зелёный фрагменты диаграммы расположились вне пределов маски. Причиной этого является неправильная работа блока управления двигателем, который не регулировал УОЗ ни по оборотам, ни по нагрузке.

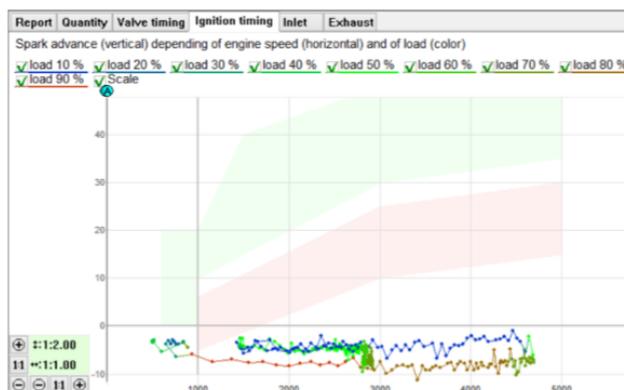


Рисунок 1. – Вкладка «Опережение» из отчёта скрипта Rx. УОЗ регулируется неправильно

Вкладка «Впуск». В этой вкладке отображается диаграмма циклового наполнения цилиндра воздухом в зависимости от оборотов и от нагрузки. Высота графика отражает количество газа, оставшегося в цилиндре после закрытия впускного клапана. Цвета фрагментов диаграммы отражают нагрузку на двигатель, рассчитываемую по величине разрежения в цилиндре на такте впуска (рисунок 2).



Рисунок 2. – Вкладка «Впуск» из отчёта скрипта Rx

Красный фрагмент диаграммы позволяет оценить влияние компонентов всего впускного тракта на наполняемость цилиндра. Чем выше расположен фрагмент, тем больше максимальное наполнение цилиндра, и тем выше отдача цилиндра. Кроме прочего, на наполняемость цилиндра влияют система изменения фаз газораспределения, система изменения геометрии впускного коллектора, максимальное сечение дроссельной заслонки и пропускная способности воздушного фильтра. Также, можно оценить влияние нагнетателя воздуха, если он установлен. По расположению фрагмента диаграммы, соответствующего работе двигателя на холостом ходу, можно оценить цикловое наполнение цилиндра в этом режиме. Чем оно меньше, тем эффективнее и экономичнее работа двигателя в этом режиме.

Вкладка «Выпуск». Вкладка отображает диаграмму затрат мощности двигателя на очистку цилиндра от отработавших газов.

Чем меньше затраты на выпуск, тем диаграмма располагается ниже. Наклонная красная линия определяет максимально допустимый предел высоты расположения диаграммы. Положение этой линии было определено экспериментально, путём анализа множества диаграмм с исправных и неисправных автомобилей. Следует заметить, что затраты на выпуск зависят от величины противодавления в выпускном тракте не напрямую.

Сама по себе величина пика давления в цилиндре на такте выпуска не является прямой характеристикой динамического сопротивления выпускной системы. Здесь важно учитывать ещё и угол поворота коленчатого вала, на который приходится этот пик. Если на такте выпуска в цилиндре возникает существенное повышение давления и его пик приходится на момент подхода поршня к ВМТ, то это практически не создает сопротивления движению коленчатого вала, поскольку поршень уже почти остановился. Если же этот пик приходится на середину такта выпуска, то возникает значительное сопротивление вращению коленчатого вала, поскольку поршень в этот момент движется с максимальной скоростью. [6]

В ходе исследования было проведено диагностирование двигателей различных марок автомобилей по традиционным методам.

Таблица 1. – Трудоемкость диагностирования различных автомобилей с использованием традиционных методов и с использованием современных информационных технологий.

Марка автомобиля	Признаки неисправностей	Неисправность	Трудоемкость с использованием традиционных методов, чел·ч	Трудоемкость с использованием предложенного метода чел·ч
Mazda 626 2,0 2000 г.	Потеря мощности и нестабильная работа двигателя под нагрузкой	Пробой изолятора свечи зажигания;	0,5	0,25
VW Polo 1,6 2015 г.		Увеличенный зазор свечи зажигания	0,5	0,3
VW Golf IV 2002 г.		Загрязнение изолятора свечи зажигания	0,6	0,3
Fiat Marea 1,6 1996 г.	Неравномерность работы двигателя	Неисправность форсунки 1-го цилиндра	3,8	0,5
Renault Logan 2009г.		Загрязнение форсунок	1	0,6
Lada Vesta 2016 г.		Обрыв цепи управляющего провода форсунки	0,6	0,3
VW Tiguan 1,6 TFSI 2008 г.		Неисправность форсунки 3-го цилиндра	1	0,4
Hundai Sonata 2.0 1997г.	Нестабильная и неравномерная работа двигателя на всех режимах	Сломанное компрессионное кольцо	0,8	0,6
Peugeot 406 2003 г.		Износ поршневых колец	1	0,5
Audi 100 C3 1986 г.		Неправильная установка ремня ГРМ	1,5	0,6
Fiat Bravo 1,4 2010 г.		Прогар клапана ГРМ	1	0,6

Таким образом существенное увеличение производительности технического обслуживания легковых автомобилей, качества выполняемых работ, возможны при внедрении более совершенных технологий диагностирования. Проводимые при этом мероприятия отличаются относительно низким уровнем затрат, а время на их реализацию минимально.

ЛИТЕРАТУРА

1. Бабошин, А.А. Методика диагностирования поршневых двигателей внутреннего сгорания по результатам их косвенного индицирования: автореф. дис. канд. тех. наук / А.А. Бабошин, Санкт-Петербург, 2013, 23с.
2. Геращенко, В.В. Методы и средства диагностирования и повышения эксплуатационных свойств автомобилей и их агрегатов: [монография] / В.В. Геращенко, Н.А. Коваленко, В.П. Лобах. – Могилёв: Беларус.-Рос.ун-т, 2017.-170 с. : ил.
3. Карташевич, А.Н. Техническое диагностирование машин: лекция./ А.Н. Карташевич, Г.Н. Сапьяник, А.Ф. Скадорва, – Горки: Белорусская государственная сельскохозяйственная академия, 2009. 32 с.
4. Ананьин, А.Д. Диагностика и техническое обслуживание машин: учебник для студентов высш. учеб. заведений / А.Д. Ананьин, В.М. Михлин, И.И. Гибитов и др. – М. : Издательский центр «Академия», 2008. – 432с.
5. Борц А.Д., Диагностика технического состояния автомобиля. / А.Д. Борц, Я.Х. Закин, Ю.В. Иванов. - М.: Транспорт, 1979. - 160 с.
6. Кострицкий В.В. Диагностирование и то двигателя атомобия : метод. указания / Полоцкий гос. ун-т. сост.: В.В. Кострицкий. – Новополоцк : ПГУ, 2015. – 123с.

ХИМИЧЕСКИЕ ТЕХНОЛОГИИ

УДК 665.777.4:665.637:614.7

СПОСОБЫ ПОДАВЛЕНИЯ ПЫЛЕОБРАЗОВАНИЯ ПРИ ТРАНСПОРТИРОВКЕ УГЛЕРОДСОДЕРЖАЩИХ МАТЕРИАЛОВ

В.А. ЛЯХОВИЧ

(Представлено: канд. техн. наук, доц. Ю.А. БУЛАВКА)

Актуальной задачей для белорусского нефтеперерабатывающего предприятия является борьба с пылеобразованием при получении и транспортировке нефтяного кокса. Запыленность воздуха может стать причиной аварийных ситуаций, а длительное пребывание человека в запыленной атмосфере вызывает профессиональные легочные заболевания. Рассмотрены способы борьбы с пылеобразованием при транспортировке нефтяного кокса.

Стратегическим объектом, с экономической точки зрения, для Республики Беларусь является установка замедленного коксования (УЗК), которая позволит увеличить на ОАО «Нафтан» выход топливных фракций и наладить выпуск новой продукции – нефтяного кокса.

Сегодня процесс замедленного коксования является одним из наиболее бурно развивающихся и перспективных процессов глубокой переработки нефтяного сырья, поскольку введение установки замедленного коксования в эксплуатацию на НПЗ приводит к увеличению основного показателя – глубины переработки нефти до 95 %.

Нефтяной кокс является побочным продуктом нефтепереработки (в результате совершенствования технологий согласно международным требованиям к качеству) и используется в качестве топлива в цементных печах и на электростанциях. Рост населения и развитие экономики способствует увеличению спроса на сырой нефтяной кокс, особенно в развивающихся странах, таких как Индия и Китай, что обусловлено качественными характеристиками продукта (небольшое количество сырого нефтяного кокса достаточно, чтобы сгенерировать значительное количество электроэнергии).

На мировом рынке отмечается тенденция по увеличению производства нефтяного кокса. До 2020 г. прогнозируется ежегодный рост рынка нефтяного кокса на 3 %, следовательно, производство данного продукта белорусскими НПЗ, а именно ОАО «Нафтан», соответствует текущей международной тенденции [1].

Топливный кокс является основным топливом, которое используется в промышленности и производственных процессах, включая производство стали, алюминия и других специальных продуктов. С момента своего производства в США в 30-х годах нефтяной кокс безопасно транспортировался морскими, железнодорожными и автомобильными перевозками [2].

Углезагрузочные вагоны являются основными машинами для обслуживания коксовых печей. Они работают в комплексе с другими коксовыми машинами – коксовыталкивателями, двересъемными машинами и тушилными вагонами. Угольная и абразивная коксовая пыль вызывают коррозию металла вагонов и обслуживающих машин [3].

Важным фактором условий труда рабочих на погрузочно-разгрузочных площадках является опасность загрязнителей воздушной среды. До настоящего момента основным неблагоприятным вредным производственным фактором является пыль. Особенно большое количество пыли образуется при работе очистных комбайнов и при погрузке-выгрузке углеродсодержащего сырья. Пыль может стать причиной заболевания пневмококоном, хроническим пылевым бронхитом [4].

Коксохимические заводы занимают ведущее место по выбросам вредных веществ среди предприятий черной металлургии. К ним относят пыль, оксиды азота, диоксид серы, оксид углерода, аммиак, фенол, бензол, нафталин, сероводород, цианистый водород и графит. Поэтому важнейшей социальной задачей, стоящей перед коксохимическими предприятиями, является повышение экологической безопасности производства, техническое переоснащение коксохимических предприятий, осуществление крупномасштабной программы строительства природоохранных объектов, внедрение современных экологически безопасных технологических процессов.

Борьба с выбросами при выталкивании кокса из печных камер – одна из наиболее сложных задач. Над раскаленным коксом, падающим в тушилный или коксовозный вагоны, возникает интенсивное восходящее течение нагретого воздуха, которое увлекает в движение значительные массы окружающего атмосферного воздуха. Этот подсосываемый (эжектируемый) из атмосферы поток подхватывает образующиеся при разрушении коксового пирога частицы пыли и увлекает их вверх. В результате возникает окрашенное пылевое облако значительных размеров, в котором кроме пыли могут содержаться и газообразные

вредные вещества, выделяющиеся из кокса; объем этих газов сравнительно невелик и обычно не превышает нескольких десятков кубометров.

Радикальным мероприятием по снижению выбросов при складировании углей и кокса является сооружение закрытого склада с эффективными системами аспирации и пылеулавливания. Запыленность воздуха рабочей зоны может быть уменьшена путем следующих мероприятий: герметизации пылящего оборудования; сооружения аспирационных систем, предназначенных для удаления запыленного воздуха; систем с эффективным пылеулавливающим оборудованием; устройством приточной вентиляции; блокировкой технологического оборудования с аспирационными системами; регулярной уборкой помещений и оборудования от осевшей пыли; систематическим контролем за состоянием воздуха производственных помещений.

Одним из факторов, влияющих на запыленность воздуха производственных помещений, является уборка осевшей пыли с поверхностей полов, стен и оборудования. На коксохимических предприятиях наибольшее распространение получила мокрая уборка, что обусловлено взрывоопасными свойствами угольной пыли. Однако опыт сухого пылеулавливания на ряде предприятий показывает, что в принципе можно производить и сухую уборку. В этом случае целесообразно использовать централизованные пылесосные установки с водокольцевыми вакуум-насосами. Это позволит избавиться от водношламового хозяйства, упростить эксплуатацию системы, но потребует устройства разветвленной сети коммуникаций, очистки воздуха перед выбросом в атмосферу и решения вопросов утилизации уловленной сухой пыли.

До настоящего времени основным способом улавливания угольной пыли остается мокрый. В связи с этим на предприятиях наиболее распространены центробежные скрубберы, скоростные промыватели, циклоны с водяной пленкой; в ряде случаев применяются оригинальные конструкции, разработанные предприятиями. В последнее время появились ротоклоны. Сухие коллекторы, используемые в качестве первой ступени очистки, в большинстве случаев оборудуют устройствами для подачи и распыления воды. Эффективность улавливания угольной пыли в мокрых аппаратах весьма различна, что связано, по-видимому, как с плотностью орошения и качеством распыления жидкости, так и с дисперсностью улавливаемых частиц [5].

Использование антипылевых химических агентов, вместо воды, может дать следующие преимущества:

- иметь остаточный эффект до 45 дней или более для целостности покрытия, в зависимости от погодных условий и типа используемого химического вещества;
- минимизировать необходимость повторного применения воды;
- разрешить дополнить высоту груды кокса, если необходимо, при хранении;
- могут быть уменьшены текущие требования к очистке коксовых конвейеров;
- может быть уменьшена потребность в техническом обслуживании на объектах по обработке кокса;
- препятствует смерзанию кокса при отрицательных температурах.

Поверхностно-активное вещество усиливает смачивание, снижая поверхностное натяжение воды, тем самым позволяя каплям воды проникать глубже в нефтяной кокс. Увлажнитель помогает замедлить испарение влаги [2].

Потребность в профилактических средствах нефтяного происхождения для пылеподавления, против прилипания, примерзания и смерзания горных пород и дисперсных материалов сохраняется и в настоящее время [6-19], однако их производство значительно сокращается в связи с различными преобразованиями в нефтяной отрасли, связанными с увеличением доли углубляющих процессов и изменением качества получаемых продуктов. Существует необходимость в разработке технологии получения этих средств из продуктов переработки современных нефтеперерабатывающих заводов.

Таким образом, пыль от углеводородного материала оказывает большое влияние на окружающую среду, следовательно, отражается на здоровье рабочих и состоянии машин и аппаратов, которые необходимы для обработки, перевозки сырья. Применение пылеподавляющих процедур и установок помогает сократить выбросы в окружающую среду, что приведет к улучшению условий труда работников коксохимической и угольно добывающей промышленности, сокращению износа обслуживающих аппаратов и транспорта.

ЛИТЕРАТУРА

1. Бизнес план ОАО «Нафтан» на 2015 год. – Новополоцк: ОАО «Нафтан», 2014. –160 с.
2. Guidance Document for the Storage and Handling of Petroleum Coke – API guidance document PC1 – first edition, DECEMBER 2014
3. Непомнящий, И.Л. Коксовые машины, их конструкция и расчет / И. Л. Непомнящий – М.: Металлургиздат, 1957. – 263 с.
4. Рогалис, В.С. Сочетание воздействия угольной пыли и радиации на здоровье шахтеров / В. С. Рогалис, М. В. Павленко, А. А. Шилов // Труды международного научного симпозиума «Неделя Горняка – 2016». Горный информационно-аналитический бюллетень (научно-технический журнал), М.: Издательство «Горная книга», 2019. – №1 (специальный выпуск 1) – 552 с.

5. Пыриков, А.Н. Защита окружающей среды на коксохимических предприятиях / А. Н. Пыриков, С. К. Васнин, Б. Н. Баранбаев. – М.: Интермет – инжиниринг, 2000. – 176 с.
6. Снижение воздействия коксовой пыли на работников путем использования пылеподавляюще-противосмерзающего средства /Булавка Ю.А., Ляхович В.А.// Вестник Полоцкого государственного университета. Серия В. Промышленность. Прикладные науки. – 2020. – № 3. – С. 83-87.
7. Reduction of airborne particulate matters emissions reduction associated with petroleumcoke productions/ V. A. Liakhovich, Y.A. Bulauka// Scientific conference abstracts of XVI International Forum-Contest of Students and Young Researchers «Topical Issues of Rational Use of Natural Resources», St.Petersburg Mining University on June 17-19, 2020 Volume 2. pp. 319.
8. Reduced dust pollution in the processes of production and transportation of petroleum coke /Liakhovich V., Bulauka Y. //European and national dimension in research. technology: Electronic collected materials of XII Junior Researchers' Conference, Novopolotsk, May 13-14, 2020 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2020. – pp.71-73
9. Способ получения пылеподавляюще-противосмерзающего средства для перевозки сыпучих углеродсодержащих материалов // Ю.А. Булавка, В.А. Ляхович/Инновационные материалы и технологии: материалы Международ.науч.-техн. конф. молодых ученых, г. Минск, 9–10 янв. 2020г. – Минск : БГТУ, 2020. –С. 232-235.
10. Пылеподавляющий и противосмерзающий состав для кокса из продуктов вторичных процессов переработки нефтяного сырья /В.А. Ляхович, Ю.А. Булавка//Наука. Технология. Производство – 2019: материалы Международной науднотехнической конференции, посвященной 100-летию Республики Башкортостан / редкол.: Н.Г. Евдокимова и др. – Уфа: Изд-во УГНТУ, 2019. – С.59-61.
11. Method to reduce freezing and improve dust suppression when receiving and transporting petcoke/V. Liakhovich, Y. Bulauka// Scientific Conference Abstracts of XV International Forum-Contest of Students and Young Researchers « Topical issues of rational use of natural resources», St. Petersburg, May 13-17, 2019.- Saint-Petersburg Mining University. - St. Petersburg,2019.-P. 71
12. Способ улучшения пылеподавления и снижения смерзания при получении и транспортировке нефтяного кокса / В.А. Ляхович, Ю.А. Булавка// Актуальные вопросы современного материаловедения: материалы VI Международной молодежной научно-практической конференции (г. Уфа, 30 октября 2019 г.) / отв. ред. ОС. Куковинец. – Уфа : РИД БашГУ, 2020. – С.205-214.
13. Method to reduce freezing and improve dust suppression when receiving and transporting petcoke/ V. Liakhovich, Y.Bulauka //European and national dimension in research. Technology = Европейский и национальный контексты в научных исследованиях: Electronic collected materials of XI Junior Researchers' Conference, Novopolotsk, May 23-24, 2019 / Polotsk State University ; ed. D. Lazouski [et al.]. – Novopolotsk, 2019. – P.104-105.
14. Способ улучшения пылеподавления при получении и транспортировке нефтяного кокса /В.А. Ляхович, Ю.А. Булавка// Системы обеспечения техносферной безопасности: материалы VI Всероссийской конференции и школы для молодых ученых (с международным участием) (Таганрог, Россия, 4-5 октября 2019 г)/Южный федеральный университет. – Ростов-наДону; Таганрог: издательство Южного федерального университета. 2019. – С.45-46.
15. Способ снижения смерзания и улучшения пылеподавления при получении и транспортировке нефтяного кокса/Ляхович В.А., Булавка Ю.А.// Сборник тезисов докладов 73-й Международной молодежной научной конференции «Нефть и газ – 2019» (22-25 апреля 2019 г. Москва). – Том 5. – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2019. – С.468–469.
16. Способы подавления пылеобразования при транспортировке углеродсодержащих материалов /Ляхович В.А., Булавка Ю.А. //Промышленная безопасность и охрана труда: практика, обучение, инновации: сб. материалов Международной заочной научно-практической конференции. – Минск: УГЗ, 2019. – С.26-29.
17. Снижение воздействия пылевого фактора на работников при получении и транспортировке нефтяного кокса /Ляхович В.А., Булавка Ю.А.// Безопасность – 2019: материалы докладов XXIV Всероссийской студенческой научно–практ. конф. с междунар. участием «Проблемы экологической и промышленной безопасности современного мира» (г. Иркутск, 16–19 апр. 2019 г.). – Иркутск : Изд-во ИРНИТУ, 2019. – С.129-131.
18. Receiving an antifreezing agent for transporting coke by rail /V. Liakhovich, V. Yemelyanova, Y. Bulauka// European and national dimension in research. technology = европейскийи национальный контексты в научных исследованиях: Electronic collected materials of X Junior Researchers' Conference, Novopolotsk, May 10-11, 2018 / Polotsk State University ; ed. D. Lazouski [et al.]. – Novopolotsk, 2018. -P.153-155.
19. Противосмерзающего средства из отходов нефтехимии для транспортировки топливного кокса /Ляхович В.А., Емельянова В.А., Булавка Ю.А.// Сборник докладов 72-й Международной молодежной научной конференции «Нефть и газ – 2018» (23-26 апреля 2018 г. Москва). – Том 2.– М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. – С.366.

УДК 665.777.4:665.637:614.7

СПОСОБ ПОЛУЧЕНИЯ ПЫЛЕПОДАВЛЯЮЩЕ-ПРОТИВОСМЕРЗАЮЩЕГО СРЕДСТВА, СНИЖАЮЩЕГО ВОЗДЕЙСТВИЕ ПЫЛИ НЕФТЯНОГО КОКСА НА РАБОТНИКОВ**В.А. ЛЯХОВИЧ***(Представлено: канд. техн. наук, доц. Ю.А. БУЛАВКА)*

В ходе исследования разработано пылеподавляюще-противосмерзающее средства, которое позволит снизить пылевое воздействие на работников установки замедленного коксования при получении нефтяного кокса и улучшить морозоустойчивость продукта. Предлагаемый способ позволяет решать проблемы транспортировки в условиях отрицательных нефтяного топливного кокса, и расширяет ассортимент товарной продукции, увеличивает эффективность производства и снижает затраты на закупку импортного противосмерзающего средства.

Перевозка полезных ископаемых, рыхлых вскрышных пород, сыпучих углеродсодержащих материалов, обладающих повышенной влажностью, сопровождается в осенний период интенсивным прилипанием, а в зимний период - смерзанием груза и примерзанием к металлической поверхности полувагонов и думпкаров, гондол, хопперов, кузовов автосамосвалов. В результате этого до 50 % горной массы остается не выгруженной, что вызывает необходимость в дополнительной очистке и приводит к ухудшению технико-экономических показателей работы карьерного транспорта на 15...20 %. Стоимость перевозки увеличивается на 10...18% [1].

Для обеспечения безопасного ведения технологического процесса необходимо строгое соблюдение требований промышленной и пожарной безопасности, охраны труда, кроме того, изменяются условия труда работающих завода, в связи с тем, что появляется дополнительный вредный производственный фактор – пылевой, обусловленный выделением в воздух рабочей зоны коксовой пыли, которая обладает способностью к тлению, самовозгоранию и самовоспламенению. Превышение предельно допустимых концентраций по пыли нефтяного кокса в воздухе рабочей зоны производственных помещений (ПДК р.з. составляет 5 мг/м³) может стать причиной развития профессиональных легочных заболеваний. По степени воздействия на организм человека пыль нефтяного кокса относится к 4 классу опасности. Согласно ГОСТ 22898 температура тления при самовозгорании пыли с размером частиц 50-100 мкм 205-235°C, температура самовоспламенения 535-625 °С. Пыль с размером частиц 50-160 мкм не взрывоопасна: нижний предел воспламенения взрывзвеси отсутствует до 500 г/м³.

Коксовое производство относится к отраслям промышленности, представляющим собой потенциальную опасность вследствие высокой вероятности контакта работающих с сырьевыми материалами, коксопыльными продуктами и вспомогательными реагентами в процессе труда и неблагоприятными физическими факторами (шума, вибрации, теплового и электромагнитного излучений, наличие источников радиоактивного излучения от радиоизотопных уровнемеров на коксовых камерах и колонне фракционирования и другие). Основными источниками загрязнения воздуха являются: блок трубчатых печей, блок теплообменников, блок коксовых камер, блок насосных, блок ректификации и стабилизации и блок места перегрузки нефтяного кокса с конвейера на конвейер, отделение грохочения, бункеры в грузки нефтяного кокса из силосов, линия погрузки нефтяного кокса в вагоны, узел дробления нефтяного кокса, транспортировки и складирования кокса. Вместе с тем, высокая влажность нефтяного кокса создает трудности при обработке кокса на УЗК, приводя к низкой эффективности распада на грохоте. Перевозка топливного нефтяного кокса, обладающего повышенной влажностью, сопровождается в осенний период интенсивным прилипанием, а в зимний период – смерзанием груза и примерзанием к металлической поверхности думпкаров, гондол, хопперов, кузовов автосамосвалов. В результате этого до 50 % горной массы остается не выгруженной, что вызывает необходимость в дополнительной очистке, при этом стоимость перевозки увеличивается до 20%. Для предупреждения пылеуноса, прилипания, смерзания, примерзания горных пород имеющих повышенную влажность применяются различные средства, наиболее широкое промышленное применение за последние годы получили профилактические средства нефтяного происхождения – Ниогрин и Универсин [2-19].

В настоящее время является актуальной для нефтеперерабатывающей отрасли задача борьбы с пылеобразованием, потерями от выдувания, прилипанием, смерзанием и примерзанием к рабочим металлическим поверхностям автомобильных и железнодорожных транспортных средств при транспортировке в условиях отрицательных температур нефтяного топливного кокса, что и определило цель настоящего исследования.

Основным требованием к качеству профилактических средств, предназначенных для защиты металлической поверхности горнотранспортного оборудования от примерзания влажных сыпучих материалов, является показатель температуры застывания.

Низкотемпературные свойства обеспечивают требуемую подвижность профилактических средств при отрицательных температурах и возможность нанесения вещества без дополнительных операций подогрева. Для эффективного применения температура застывания профилактического средства Ниогрин не должна быть выше -35°C , для Универсина марки «С» – не выше -45°C . Профилактические средства, используемые для предотвращения пылеобразования на временных автодорогах при добыче полезных ископаемых открытым способом, а также для защиты поверхностей подвижного состава от примерзания и выдувания сыпучих материалов (угля, кокса, торфа и т.д.), должны соответствовать требованиям пожарной безопасности, которые регламентируют значение температуры вспышки. Для безопасного хранения и нанесения в полузакрытых помещениях и на открытом воздухе профилактическое средство Ниогрин должно иметь температуру вспышки не ниже 75°C , а профилактическое средство Универсин марки «Л» – не ниже 150°C . Также важными характеристиками профилактических средств являются вязкость и содержание воды и механических примесей. Для возможности нанесения средств с помощью форсунок и равномерного распределения на обрабатываемых поверхностях профилактические средства должны обладать определенными значениями вязкости. Минимальным содержанием либо отсутствием воды и механических примесей. Содержание воды негативно влияет на эксплуатационные свойства профилактических средств в условиях отрицательных температур. Образование кристаллов воды способствует усилению связей между частицами насыпного материала, а также материала с металлическими поверхностями транспортного оборудования и приводит к снижению подвижности профилактического средства.

В качестве дисперсионной среды, предлагаемых профилактических средств выбран следующий продукт вторичных процессов ОАО «Нафтан» – дизельное топливо с установки «Висбрекинг-Термокрекинг» (далее ДТ Висбрекинга). В качестве загущающих добавок улучшающих низкотемпературные свойства использовали высокомолекулярные тяжелые нефтяные остатки: гудрон с установки ВТ-1 ОАО «Нафтан», мазут с установки АВТ-6 ОАО «Нафтан», остаток с установки «Висбрекинг-Термокрекинг» ОАО «Нафтан», нефтеполимерную смолу из тяжелой смолы пиролиза (НПС ТСП) производства ОАО «Лесохимик».

Профилактических средств получали компаундированием при помощи механической лопастной мешалки высокомолекулярных тяжелых нефтяных остатков с дисперсионными средами и последующей термической обработки при температуре $100\pm 5^{\circ}\text{C}$ в течение 1,5...2 ч. Остатки добавляли в концентрациях до 30 % масс. на дисперсионную среду.

Для полученных профилактических средств определены: условная вязкость при 50°C ГОСТ 6258, температуру застывания по ГОСТ 20287, температуру вспышки по ГОСТ 6356, массовую долю воды по ГОСТ 2477, массовая доля механических примесей по ГОСТ 6370 и провели испытание на медной пластинке по ГОСТ 6321. Для лучших по эксплуатационным свойствам провели моделирование процесса примерзания и прилипания к стенкам вагона на примере каменного угля. Для моделирования процесса три образца профилактических средств с минимальными температурами застывания – ниже минус 70°C :

- ~ образец 1: дизельное топливо Висбрекинга с 5% масс. мазута с установки АВТ-6;
- ~ образец 2: дизельное топливо Висбрекинга с 3% масс. гудрона с установки ВТ-1.

Задача испытаний состояла в определении профилактирующей способности новых низкозастывающих смазочных составов предотвращать прилипание, смерзание и примерзание влажного каменного угля к поверхности лабораторных металлических лотков, при отрицательных температурах - естественных климатических условиях, максимально приближенных к условиям применения профилактических смазок.

Испытания проводились в лабораторных условиях при искусственном обводнении угля в морозильных камерах при температуре -30°C . На поверхность, заранее охлажденных металлических лотков (30x20x15 см), кисточкой, равномерным тонким слоем, не допуская натеков средства на металлической поверхности, наносилось охлажденное профилактическое средство. По истечении 5 - 10 минут после нанесения профилактического средства, во все емкости загрузалось одинаковое количество сыпучего материала – каменного угля.

После при помощи пульверизатора уголь обводнялся в количестве воды 5% масс. от массы угля. Для создания условий максимально приближенных к условиям погрузки, после наполнения емкостей углем, дополнительно проводили утрамбовку сыпучей породы. Наполненные лотки размещались в морозильных камерах на двое суток при температуре -30°C . По истечении времени испытаний, лотки опустошались путем опрокидывания, затем проводился осмотр металлической поверхности лабораторных думпкаров, оценка степени выгрузки сыпучей породы в процентном соотношении. Последовательность моделирования процесса примерзания и прилипания каменного угля к стенкам полувагонов и думпкаров приведена на рисунке 1.

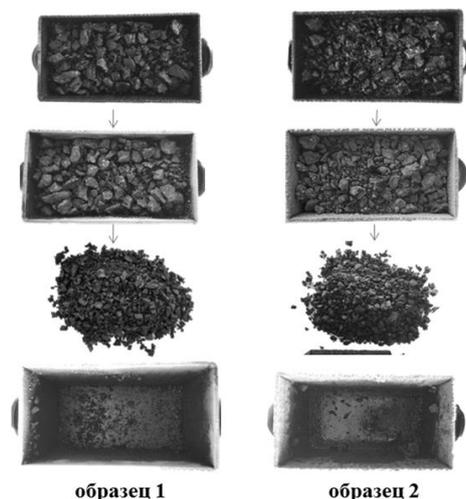


Рисунок 1. – Последовательность моделирования процесса примерзания и прилипания каменного угля к стенкам полувагонов и думпкаров

Лабораторное моделирование процесса примерзания и прилипания каменного угля к стенкам полувагонов и думпкаров показали, что:

- при обработке лабораторных думпкаров образцами профилактических средств 1 и 2 были получены положительные результаты;
- после выгрузки породы на стенках лотка, предварительно обработанного образцами профилактических средств 1 и 2, не было обнаружено примерзшего угля, и дополнительная ручная выгрузка не требовалась. На дне лотка так же не наблюдалось примерзания угля.

Технико-экономические показатели предлагаемых профилактических средств в сравнении промышленными аналогами приведены в таблице 1. Анализ технико-экономических показателей предлагаемых профилактических средств (образец 1 и 2) в сравнении промышленными аналогами показал, что они не уступают по эксплуатационным свойствам «Ниогрину-ПС 35С» и при этом дешевле более чем в три раза.

Таблица 1. – Технико-экономические показатели профилактических средств

Показатели	Ниогрин ПС-35С	Уни-версин-С	Северин-2	образец 1	образец 2
Условная вязкость при 50°C, °ВУ, ГОСТ 6258, в пределах	1,0 – 3,0	1,1 - 3,5	1,1 - 1,5	1,12	1,11
Температура застывания по ГОСТ 20287, °С, не выше	- 35	- 40	- 50	< -65	< -65
Температура вспышки по ГОСТ 6356, °С, не ниже	40	80	80	70	70
Массовая доля воды по ГОСТ 2477, в %, не более	2,0	0,5	0,5	0,01	0,01
Массовая доля механических примесей по ГОСТ 6370, в %, не более	1,0	0,3	0,2	следы	следы
Испытание на медной пластинке по ГОСТ 6321	выдерживает	-	-	выдерживает	
Цвет, визуально	от св. коричневого до черного				
Стоимость \$/тонну	180-200	180-200	210-1100	55	54

Для моделирования процесса пылеуноса нефтяного кокса был отобран образец 1 пылеподавляюще-противосмерзающее средства, полученный из дизельного топлива Висбрекинга с 5% масс. мазута с установки АВТ-6. Испытание опытного образца пылеподавляюще-противосмерзающего средства проводилось в условиях лаборатории кафедры технологии и оборудования переработки нефти и газа. Задача состояла в определении способности образца предотвращать пылеунос и выдувание при транспортировке нефтяного кокса.

Скорость ветра в лабораторной аэродинамической трубе определяли с помощью механического чашечного анемометра. Сухая и обработанная навеска нефтяной коксовой пыли помещалась в аэродинамическую трубу на 30 минут при скорости ветра 7,5 км/ч.

Взвешивание навески без обработки профилактическим средством до и после испытания показали, что пылеунос составляет 58%: 10,578г до проведения испытания и 4,351г по окончании испытания, выдув составил 6,227г. Взвешивание навески обработанной образцом 1 до и после испытания показали, что пылеунос составляет 12%: 10,247г до проведения испытания и 9,028г по окончании испытания, выдув составил 1,219г.

Таким образом, мы получили опытные образцы профилактических средств, которые снижают смерзаемость углеродсодержащих пород при транспортировке. Предлагаемые профилактические средства не проявляют коррозионной агрессивности по отношению к металлическим поверхностям, не содержат механических примесей и воды, имеют достаточно высокие температуры вспышки, соответствующие требованиям пожарной безопасности, характеризуются низкими температурами застывания, позволяющими их использовать при температурах окружающей среды ниже минус 40 °С; имеют хорошую смачиваемость, адгезионную способность и реологические свойства; высокую способность к пылеподавлению; небольшой расход 1...1,5% масс. на массу нефтяного кокса.

Предлагаемые образцы не уступают по эксплуатационным свойствам аналогу «Ниогрину-ПС 35С» и при этом дешевле его более чем в три раза [20].

Результаты лабораторных испытаний показали, что предлагаемые пылеподавляюще-противосмерзающих средства из местного сырья, а именно на базе остаточных нефтепродуктов ОАО «Нафтан» могут быть рекомендованы к использованию для борьбы с пылеобразованием и предотвращением примерзания и прилипания горных пород и сыпучих углеродсодержащих грузов к металлическим поверхностям и полостям автомобильного и железнодорожного транспорта.

ЛИТЕРАТУРА

1. Учитель, А.Д., Кормер, М.В., Шмельцер, Е.О. Методы предотвращения смерзания углей при их транспортировке. // Вестник Криворожского национального университета. -2013. - № 4 (33). - С. 113-117.
2. Снижение воздействия коксовой пыли на работников путем использования пылеподавляюще-противосмерзающего средства /Булавка Ю.А., Ляхович В.А.// Вестник Полоцкого государственного университета. Серия В. Промышленность. Прикладные науки. – 2020. – № 3. – С. 83-87.
3. Reduction of airborne particulate matters emissions reduction associated with petroleumcoke productions/ V. A. Liakhovich, Y.A. Bulauka// Scientific conference abstracts of XVI International Forum-Contest of Students and Young Researchers «Topical Issues of Rational Use of Natural Resources», St.Petersburg Mining University on June 17-19, 2020 Volume 2. pp. 319.
4. Reduced dust pollution in the processes of production and transportation of petroleum coke /Liakhovich V., Bulauka Y. //European and national dimension in research. technology: Electronic collected materials of XII Junior Researchers' Conference, Novopolotsk, May 13-14, 2020 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2020. – pp.71-73.
5. Способ получения пылеподавляюще-противосмерзающего средства для перевозки сыпучих углеродсодержащих материалов // Ю.А. Булавка, В.А. Ляхович/Инновационные материалы и технологии: материалы Междунар.науч.-техн. конф. молодых ученых, г. Минск, 9–10 янв. 2020г. – Минск : БГТУ, 2020. – С. 232-235.
6. Пылеподавляющий и противосмерзающий состав для кокса из продуктов вторичных процессов переработки нефтяного сырья /В.А. Ляхович, Ю.А. Булавка//Наука. Технология. Производство – 2019: материалы Международной научнотехнической конференции, посвященной 100-летию Республики Башкортостан / редкол.: Н.Г. Евдокимова и др. – Уфа: Изд-во УГНТУ, 2019. – С.59-61.
7. Method to reduce freezing and improve dust suppression when receiving and transporting petcoke/V. Liakhovich, Y. Bulauka// Scientific Conference Abstracts of XV International Forum-Contest of Students and Young Researchers « Topical issues of rational use of natural resources», St. Petersburg, May 13-17, 2019.- Saint-Petersburg Mining University. - St. Petersburg,2019.-P. 71
8. Способ улучшения пылеподавления и снижения смерзания при получении и транспортировке нефтяного кокса / В.А. Ляхович, Ю.А. Булавка// Актуальные вопросы современного материаловедения: материалы VI Международной молодежной научно-практической конференции (г. Уфа, 30 октября 2019 г.) / отв. ред. ОС. Куковинец. - Уфа: РИД БашГУ, 2020. – С.205-214.
9. Method to reduce freezing and improve dust suppression when receiving and transporting petcoke/ V. Liakhovich, Y.Bulauka //European and national dimension in research. Technology = Европейский и

- национальный контексты в научных исследованиях: Electronic collected materials of XI Junior Researchers' Conference, Novopolotsk, May 23-24, 2019 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2019. – P.104-105.
10. Способ улучшения пылеподавления при получении и транспортировке нефтяного кокса /В.А. Ляхович, Ю.А. Булавка// Системы обеспечения техносферной безопасности: материалы VI Всероссийской конференции и школы для молодых ученых (с международным участием) (Таганрог, Россия, 4-5 октября 2019 г.)/Южный федеральный университет. –Ростов-наДону; Таганрог: издательство Южного федерального университета. 2019.- С.45-46.
 11. Способ снижения смерзания и улучшения пылеподавления при получении и транспортировке нефтяного кокса/Ляхович В.А., Булавка Ю.А.// Сборник тезисов докладов 73-й Международной молодежной научной конференции «Нефть и газ – 2019» (22-25 апреля 2019 г. Москва). – Том 5. – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2019. – С.468-469.
 12. Способы подавления пылеобразования при транспортировке углеродсодержащих материалов /Ляхович В.А., Булавка Ю.А. //Промышленная безопасность и охрана труда: практика, обучение, инновации: сб. материалов Международной заочной научно-практической конференции. – Минск: УГЗ, 2019. – С.26-29.
 13. Снижение воздействия пылевого фактора на работников при получении и транспортировке нефтяного кокса /Ляхович В.А., Булавка Ю.А.// Безопасность – 2019: материалы докладов XXIV Всероссийской студенческой научно–практ. конф. с междунар. участием «Проблемы экологической и промышленной безопасности современного мира» (г. Иркутск, 16–19 апр. 2019 г.). – Иркутск : Изд-во ИРНИТУ, 2019. – С.129-131.
 14. Receiving an antifreezing agent for transporting coke by rail /V. Liakhovich, V. Yemelyanova, Y. Bulauka// European and national dimension in research. technology = европейский и национальный контексты в научных исследованиях: Electronic collected materials of X Junior Researchers' Conference, Novopolotsk, May 10-11, 2018 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2018. –P.153-155
 15. Инновационный подход к переработке тяжелой смолы пиролиза углеводородного сырья /Булавка Ю.А., Якубовский С.Ф., Хохотов С.С., Ляхович В.А.// Сборник трудов XII Всероссийской научно-технической конференции «Актуальные проблемы развития нефтегазового комплекса России». – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. –с.23-26.
 16. Противосмерзающего средства из отходов нефтехимии для транспортировки топливного кокса /Ляхович В.А., Емельянова В.А., Булавка Ю.А.// Сборник докладов 72-й Международной молодежной научной конференции «Нефть и газ – 2018» (23-26 апреля 2018 г. Москва). – Том 2. – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. – С.366.
 17. Получение товарных продуктов из тяжелой смолы пиролиза/Булавка Ю.А., Якубовский С.Ф., Ляхович В.А.// Актуальные вопросы современного химического и биохимического материаловедения: материалы V Международной молодежной научно-практической школы-конференции (г. Уфа, 4-5 июня 2018 г.) / отв. ред. О.С. Куковинец. – Уфа: РИЦ БашГУ, 2018. – С. 54-57.
 18. Современные направления переработки тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка, В.А. Ляхович, А.С. Москаленко// Новые технологии– нефтегазовому региону: материалы Международной научно-практической конференции/ отв. ред. П. В. Евтин. – Тюмень: ТИУ, 2018. – С.33-35.
 19. Рациональная переработка тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка., В.А. Ляхович, С.Ф.Якубовский // Tatarstan UpExPro 2018: материалы II Международной молодежной конференции (14–17 февраля 2018 г., Казань). – Казань: Изд-во Казан. ун-та, 2018. –С.120-121.
 20. Пыриков, А.Н. Защита окружающей среды на коксохимических предприятиях / А. Н. Пыриков, С. К. Васнин, Б. Н. Баранбаев. – М.: Интернет – инжиниринг, 2000. – 176 с.

УДК 662.758.2

**ОПРЕДЕЛЕНИЕ ФРАКЦИОННОГО СОСТАВА ДИЗЕЛЬНЫХ ТОПЛИВ
В СМЕСИ С ПОДСОЛНЕЧНЫМ МАСЛОМ****Я.В. ПРАШКОВИЧ, В.А. ТРОЦ***(Представлено: канд. техн. наук А.В. СПИРИДОНОВ)*

В данной статье рассматриваются влияние добавки подсолнечного масла в дизельное топливо на изменение температуры кипения смеси. Проведены исследования эффективности применения и свойств биодизельных топливных композиций. Представлен фракционный состав этих топлив. Приведены основные характеристики и мировой опыт использования отходов растительных масел.

Идея использовать растительные масла в качестве топлив для дизельных двигателей была выдвинута еще при создании первых таких моторов. Однако с освоением нефтяных запасов в XX веке более выгодным оказалось топливо из нефти. Сейчас биодизельное топливо часто отождествляют с рапсовым маслом, которое действительно стало основным сырьевым источником "биосоляры" в Европе. Однако биодизельное топливо можно получать и из других масел, например, подсолнечного, пальмового или соевого, что и делают за пределами Европы.

Важно иметь в виду, что сами по себе растительные масла в качестве топлив не используются. Любая «биосоляра» представляет собой смесь растительных масел. В растительном содержатся жиры – эфиры жирных кислот с глицерином. В Европе основным биодизельным топливом стал метиловый эфир рапсового масла.

Растительные масла и их эфиры, как и спирты, отличаются агрессивностью ко многим материалам, традиционно используемым в двигателях и топливной системе автомобилей. В последние годы большинство европейских производителей выпускают машины, допускающие использование смесей нефтяного топлива с "биосолярой" в количестве 5-20%, а иногда и 100% биотоплива. Добавление биодизельного компонента в количестве до 5% обычно считается приемлемым для любых двигателей, неадаптированных к биотопливу. Достаточно активно биодизельное топливо внедряется и в США, где в качестве сырья используют чаще всего соевое масло. Еще один перспективный источник "биосоляры" - отработанные пищевые масла.

Истощение нефтяных месторождений и продолжающийся рост цен на нефть и нефтепродукты делают неизбежным все более широкое использование в дизельных двигателях биотоплив на основе растительных масел. Перевод дизелей на биотоплива позволит не только обеспечить замещение топлив нефтяного происхождения топливами, производимыми из возобновляемых сырьевых ресурсов, но и заметно снизить токсичность отработавших газов (ОГ) и улучшить экологическую ситуацию в крупных городах и населенных пунктах сельской местности [1–3].

Для централизованного снабжения топливом автотранспорта больших городов в значительной степени пригодны сложные эфиры растительных масел — метиловые, этиловые и бутиловые, получаемые путем обработки растительных масел спиртами (соответственно метанолом, этанолом и бутанолом) [4, 5]. Из перечисленных эфиров более дешевыми являются метиловые эфиры. Причем сложные эфиры растительных масел можно получать из различных растительных масел — рапсового, подсолнечного, соевого, пальмового и др. Возможно использование для этих же целей животных жиров. Производство биодизельных топлив на основе растительных масел и животных жиров постоянно расширяется. Данные о динамике производства биодизельного топлива в некоторых странах представлены на рис. 1-2.

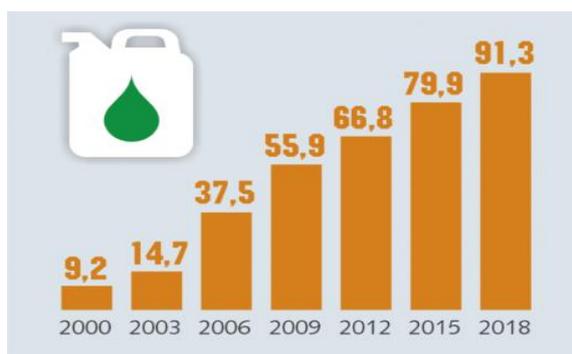


Рисунок 1. – Динамика роста производства биодизельного топлива (эферы растительных масел и животных жиров) в мире в млн. т.

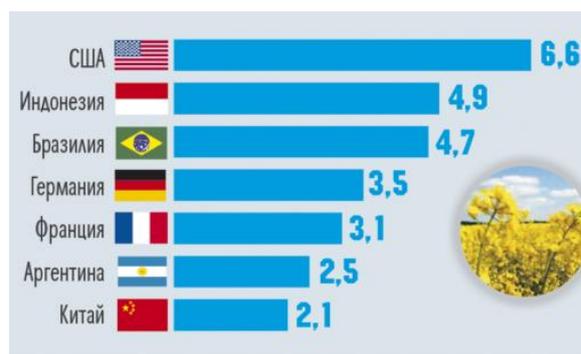


Рисунок 2. – Мировые лидеры по производству биодизельного топлива.

Биодизельные топлива, предназначенные для использования в дизельных двигателях, должны за минимальный период сформировать в камере сгорания топливно-воздушную смесь, обеспечивающую ее легкое воспламенение, плавное и достаточно полное сгорание с минимальным содержанием сажи, токсичных и канцерогенных веществ. Не допускается образование осадков в топливоподающей системе и нагароотложений на деталях двигателя. В связи с изложенным одной из основных проблем является организация рабочего процесса дизельных двигателей, работающих на биодизельных топливах. Обеспечение современных требований к показателям топливной экономичности и токсичности возможно лишь при дальнейшем совершенствовании процессов распыливания топлива и смесеобразования. При этом скорость формирования топливно-воздушной смеси предопределяется как длиной и шириной топливных струй, так и их структурой (качеством распыливания), интенсивностью испарения топлива и диффузии его в сжатый воздух.

Для достижения требуемого качества процесса смесеобразования топливо должно обладать заданным фракционным составом, который является одним из важнейших показателей эксплуатационных свойств топлив для двигателей внутреннего сгорания. Фракционный состав характеризует содержание в топливе различных фракций, выкипающих в определенных температурных пределах.

При снятии характеристик фракционного состава для традиционного дизельного топлива обычно выделяют температуру начала перегонки (начала кипения), температуры перегонки 10, 50, 90 % топлива и температуру окончания перегонки (конца кипения), соответствующую перегонке 96 или 98 % топлива. Температура перегонки 10 % топлива характеризует склонность топлива к образованию паровых пробок в системе питания дизеля. Наличие в топливе большого количества легких фракций приводит к увеличению жесткости сгорания (скорости нарастания давления и максимального давления сгорания) и снижению ресурса работы дизельного двигателя.

Среднюю испаряемость топлива устанавливают по температуре выкипания 50 %, которая обуславливает испаряемость топлива и период задержки воспламенения. Наличие в топливе тяжелых трудноиспаряющихся фракций можно определить по температуре перегонки 90 %. По температуре окончания перегонки, соответствующей перегонке 96 или 98 % топлива, также выявляют присутствие в нем тяжелых (выкипающих при высоких температурах) фракций, чрезмерное содержание которых в топливе приводит к ухудшению испарения и усложнению образования однородной топливно-воздушной смеси. Высокая температура выкипания 96 % топлива свидетельствует о наличии тяжелых фракций, которые не только ухудшают смесеобразование, но и увеличивают нагарообразование на деталях цилиндропоршневой группы и тем самым снижают экономичность и надежность работы двигателя. При использовании топлива с большим содержанием тяжелых фракций некоторая их часть догорает в такте расширения, вызывая дымность ОГ и повышение их температуры, рост удельного расхода топлива, загрязнение моторного масла и деталей двигателя продуктами неполного сгорания топлива, прежде всего сажей.

Перечисленные выше факторы указывают на необходимость учитывать фракционный состав при выборе топлива для дизелей. Это также относится и к биодизельному топливу. Фракционный состав топлив определяется методом перегонки — путем разделения смеси жидкостей на компоненты, основанном на разности температур их кипения. Данный метод заключается в нагревании жидкости до кипения с последующей конденсацией паров в холодильнике. В настоящее время существуют способы и устройства автоматического определения фракционного состава топлив. Проведём фракционную перегонку дизельного топлива с подсолнечным маслом.

Нанесем на график (рис. 3) зависимости разности температур кипения смесей дизельного топлива с подсолнечным маслом и чистого дизельного топлива разной концентрации подсолнечного масла по интервалу кипения (согласно ГОСТу 2177-99).

Эти точки можно адекватно описать полиномами. Введём обозначения величин: $\Delta t = t_{дт} - t_{см}$ – разность температур кипения дизельного топлива (чистого) и смеси дизельного топлива с растительным маслом; $v = V/V_0$ – объёмная доля отгона нефтепродукта (по ГОСТу), φ – объёмная доля растительного масла в дизельном топливе.

Рассмотрим зависимость разности температур кипения смесей дизельного топлива с подсолнечным маслом и чистого дизельного топлива от концентрации подсолнечного масла в дизельном топливе (рис. 3). Кривые в интервале от начала кипения до 40%-й точки и от 93%-й точки до конца кипения имеют монотонный характер. Кривые в интервале 50% - 90% имеют точки экстремума. Эти кривые можно представить полиномиальными уравнениями.

Графики на рис. 3-4 представляют собой проекции этой поверхности на координатные плоскости. Таким образом можно представить зависимость разности температур кипения смесей дизельного топлива в виде поверхности 3-го порядка (рис. 4).

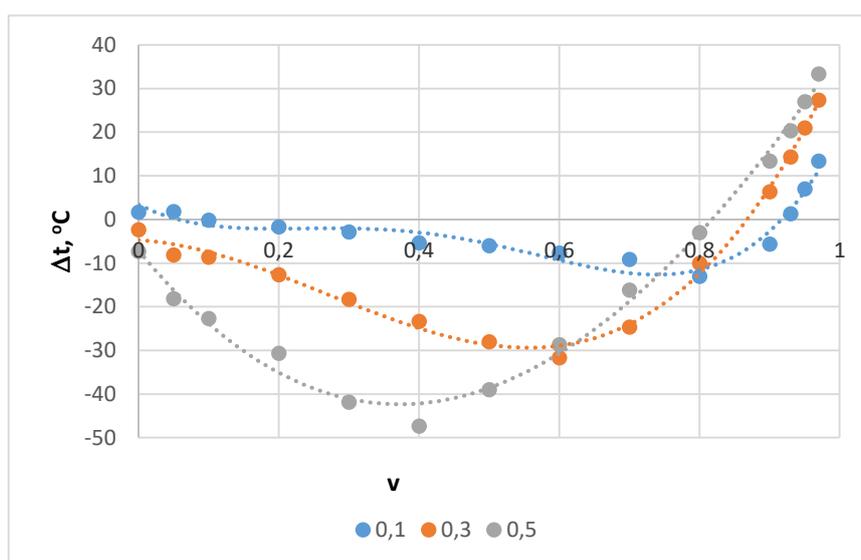


Рисунок 3. – График зависимости разности температур кипения в контрольных точках (по ГОСТу) дизельного топлива и смесей дизельного топлива с подсолнечным маслом разной концентрации подсолнечного масла по интервалу кипения

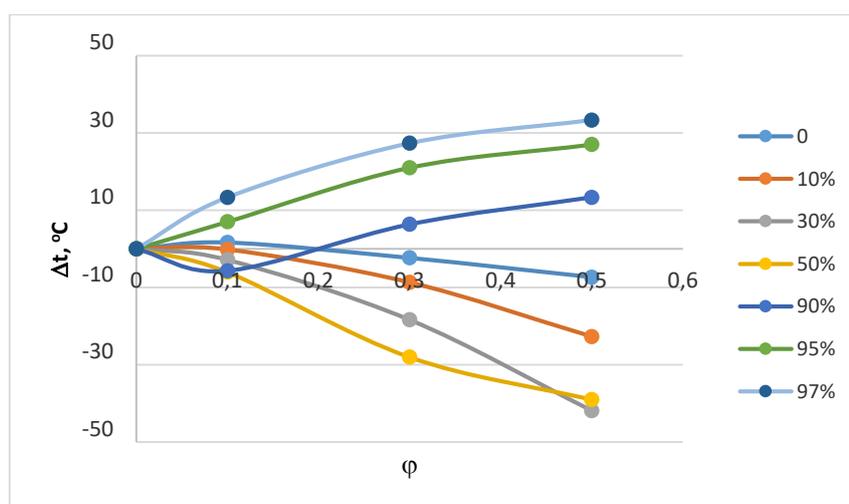


Рисунок 4. – График зависимости разности температур кипения дизельного топлива и смесей дизельного топлива с подсолнечным маслом от концентрации подсолнечного масла (цветные линии – контрольные точки кипения по ГОСТу).

Обработав массив экспериментальных данных получили значения коэффициентов в приведенном выше уравнении, со средним квадратичным отклонением 3 °С. Проведём оценку значимости полученных коэффициентов уравнения регрессии. После оценки значимости мы пришли к выводу, что некоторые переменные оказывают незначительное влияние на результаты расчёта, поэтому их количество можно сократить. В результате получим новое уравнение регрессии и произведем уточнённый расчёт коэффициентов, со средним квадратичным отклонением 3 °С. Т.е. значение среднего квадратичного отклонения осталось прежним.

$$\Delta t = 42,2 \cdot \varphi - 127 \cdot \varphi^2 + 20,1 \cdot v - 128 \cdot v^3 + 112 \cdot v^4 - \\ - 669 \cdot \varphi \cdot v + 720 \cdot \varphi \cdot v^2 + 668 \cdot \varphi^2 \cdot v - 562 \cdot \varphi^2 \cdot v^2$$

Таким образом, можно считать данный вид уравнения приемлемым для дальнейших расчётов. Данное уравнение позволяет определить изменение температуры кипения дизельного топлива в различных точках по ГОСТу в зависимости от количества добавленного растительного масла. Это актуально, так как каждая контрольная точка дизельного топлива, как отмечалось выше, влияет на его эксплуатационные показатели.

В заключение следует отметить, что приведенные данные экспериментальных исследований подтверждают возможность использования в дизелях биодизельных топлив, получаемых из различных сырьевых ресурсов. Наибольшее приближение к свойствам нефтяных дизельных топлив обеспечивает применение смесей нефтяного дизельного топлива и метиловых эфиров растительных масел. При этом удастся получать показатели токсичности отработавших газов, которые заметно лучше аналогичных показателей дизеля, работающего на чистом дизельном топливе. Причем улучшения показателей токсичности отработавших газов достигают даже при небольшом содержании биодизельного топлива в смеси. Это позволяет использовать исследованные биодизельные топлива в качестве экологических добавок к нефтяному дизельному топливу.

ЛИТЕРАТУРА

1. Использование биотоплив на основе растительных масел в дизельных двигателях / В.А. Марков, С.Н. Девянин, В.Г. Семенов, А.В. Шахов, В.В. Багров. М.: ISSN 0236-3941. Вестник МГТУ им. Н.Э. Баумана. Сер. "Машиностроение". 2012.
2. Использование растительных масел и топлив на их основе в дизельных двигателях / В.А. Марков, С.Н. Девянин, В.Г. Семенов, А.В. Шахов, В.В. Багров. М.: ООО НИЦ "Инженер", 2011. 536 с.
3. Электронный документ от 13.04.2017: <http://computerra.ru/> Зачем нужны биотоплива - Компьютерра-Онлайн.htm
4. Топлива, смазочные материалы, технические жидкости. Ассортимент и применение: Справочник. / ред. Школьников В.М. – М. «Техноинформ», 1999, с. 13-94.

УДК 661.715.6

**ПОЛУЧЕНИЕ НЕФТЯНОГО КОКСА
НА ОСНОВЕ ТЯЖЕЛОЙ СМОЛЫ ПИРОЛИЗА ЗАВОДА «ПОЛИМИР» ОАО «НАФТАН»****Е.А. ШУЛЬГА***(Представлено: канд. хим. наук, доц. С.Ф. ЯКУБОВСКИЙ)*

Тяжелая смола пиролиза углеводородного сырья является побочным продуктом пиролиза углеводородного сырья, представляет собой смесь различных групп углеводородов, преимущественно ароматических, с температурой кипения выше 160°C. В настоящее время этот продукт не находит рационального применения, вынужденно вовлекается в котельное топливо. В данной статье приведены результаты по исследованию возможного применения тяжелой смолы пиролиза углеводородного сырья в качестве сырья для получения нефтяного кокса.

В Республике Беларусь, в частности, на заводе «Полимир» ОАО «Нафтан» объемы выработки отхода производства – тяжелой смолы пиролиза (ТСП) – достигают 12000-16000 тонн в год [1-17]. Вопрос рационального использования ТСП остро актуален в связи с приближением крупной модернизации завода «Полимир» ОАО «Нафтан», одной из целей которой является увеличение мощности предприятия, что приводит к увеличению количества побочных продуктов и обострению проблем, связанных с их сбытом. Получаемая при пиролизе жидких углеводородов на установках этиленового производства тяжелая смола пиролиза содержит конденсированные ароматические углеводороды и их алкилпроизводные. Обладая высокой степенью ароматичности и уникальным химическим составом, тяжелая смола пиролиза не содержит гетероциклические соединения, что выгодно отличает ее от тяжелых остатков нефтепереработки, которые используются в качестве сырья коксования.

Вместе с тем, создание сырьевой базы для производства электродов является одним из актуальных направлений в промышленности, поскольку кокс с содержанием серы до 1% используется в производстве графитируемых электродов, закупается по импорту. В частности, игольчатый кокс используется для получения высококачественных графитовых электродов. Электроды должны иметь высокую механическую прочность, электропроводность, низкое содержание серы и низкий коэффициент теплового расширения.

В настоящем исследовании выполнена оценка пригодности использования тяжелой смолы пиролиза завода «Полимир» ОАО «Нафтан» для получения нефтяного кокса и нефтяного пека.

Тяжелая смола пиролиза завода (ТСП) была исследована по нескольким показателям, которые используются при оценке пригодности сырья для получения нефтяного кокса. Основными показателями качества сырья для производства нефтяного кокса являются состав и молекулярная структура всех его составляющих. От состава сырья и строения молекул его компонентов зависят реакционная способность сырья, количественные и качественные показатели процесса коксования и основные эксплуатационные свойства получаемого кокса.

Вопрос о химическом составе сырья решается методами комплексного химического анализа и не представляет особых трудностей. Однако данных о молекулярной структуре пока еще крайне недостаточно. Но уже имеющиеся данные по структуре высокомолекулярных соединений нефти в ряде случаев дают ориентировочную оценку при решении отдельных научных и практических вопросов.

Весьма важной характеристикой при анализе сырья для коксования является его коксуемость по Конрадсону, определяемая обычно в стандартном приборе с газовым обогревом [1-4,16]. Так же были определены вещества, нерастворимые в толуоле (α -фракция), установлено содержание асфальтенов по методу Маркусона. Проведена экстракция пека смолы пиролиза смесью алифатических и ароматических растворителей с целью изучения его пригодности для сырья коксования.

На первом этапе исследования проведена разгонка тяжелой смолы пиролиза по Энглери. Сущность метода заключается в разделении нефтепродукта на составляющие его фракции по температурам выкипания последних. Полученные результаты разгонки показывают, что до температуры 180°C выкипает примерно 1-1,5% углеводородов, содержащихся в ТСП. Наибольшее количество углеводородов выкипает в интервале температур 180-210°C – это так называемое «сине-зеленое масло». Фракции 210-220°C, 210-230°C, 210-240°C составляют порядка 15%. Установлено, при разгонке ТСП около 50% приходится на твердый кубовый остаток (пек).

На втором этапе исследования определены вещества нерастворимые в толуоле (α -фракция). Метод основан на использовании различной растворимости компонентов пека и заключается в обработке навески его толуолом с последующим определением массы отфильтрованного и высушенного нерастворившегося остатка. Для проведения анализа была взята три навески пека с интервалами температур кипения 220 – к.к., 230 – к.к., 240 – к.к. Результаты исследований показали, что содержание веществ по результатам

исследования, нерастворимых в толуоле в исследуемых объектах минимально и составила 0,2% масс. для каждой исследуемой фракции. Подтверждено, что исследуемое сырье является пригодным для получения нефтяного кокса, в том числе игольчатого, по действующим нормативам, содержание α -фракции должно быть не более 8...11% масс.

На третьем этапе исследования проведена экстракция пеков смесью растворителей. Сущность метода заключается в экстракции пека смесью алифатического и ароматического растворителей с целью последующего коксования экстракта. В качестве смеси алифатических и ароматических растворителей были взяты смесь Уайт-спирита и сольвента, выкипающую в пределах 130-185°C при соотношении смесь Уайт-спирита и сольвента: остаток смолы по массе 2:1 – 3:1. В результате исследования было установлено, что с увеличением содержания в смеси растворителей процентного содержания сольвента и при увеличении времени экстракции количество нерастворимого остатка увеличивается, следовательно, качество получаемого кокса будет снижаться. При увеличении температуры экстракции будет увеличиваться выход нерастворимого остатка, что негативно скажется на выходе возможно получаемого кокса.

На четвертом этапе исследования проведено выделение асфальтенов по методу Маркусона, которое показало, что масса асфальтенов составила 15 % масс. от навески для фракции 220-кк, 17% масс для навески 230-кк и 20% масс 240-кк. Полученные данные согласуются с нормой содержания последних для сырья для получения игольчатого кокса.

В целом, в результате комплексных исследований подтверждено, что тяжёлая смола пиролиза завода «Полимир» ОАО «Нафтан» по изученным показателям может быть использована в качестве сырья для получения нефтяного кокса для электродной промышленности.

ЛИТЕРАТУРА

1. Инновационный подход к переработке тяжелой смолы пиролиза углеводородного сырья /Булавка Ю.А., Якубовский С.Ф., Хохотов С.С., Ляхович В.А.// Актуальные проблемы развития нефтегазового комплекса России: Сборник тезисов XII Всероссийская научно-техническая конференция (12-14 февраля 2018 г., г. Москва). – РГУ нефти и газа (НИУ) имени И.М. Губкина.- Москва, 2018.-С.209.
2. Рациональная переработка тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка, В.А. Ляхович, С.Ф.Якубовский // Tatarstan UpExPro 2018: материалы II Международной молодежной конференции (14–17 февраля 2018 г., Казань). – Казань: Изд-во Казан. ун-та, 2018. –С.120-121.
3. Инновационный подход к переработке тяжелой смолы пиролиза углеводородного сырья /Булавка Ю.А., Якубовский С.Ф., Хохотов С.С., Ляхович В.А.// Сборник трудов XII Всероссийской научно-технической конференции «Актуальные проблемы развития нефтегазового комплекса России». – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. –С.23-26.
4. Получение товарных продуктов из тяжелой смолы пиролиза / Булавка Ю.А., Якубовский С.Ф., Ляхович В.А.// Актуальные вопросы современного химического и биохимического материаловедения: материалы V Международной молодежной научно-практической школы-конференции (г. Уфа, 4-5 июня 2018 г.) / отв. ред. О.С. Куковинцев. - Уфа: РИЦ БашГУ, 2018-С. 54-57.
5. Современные направления переработки тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка, В.А. Ляхович, А.С. Москаленко// Новые технологии – нефтегазовому региону: материалы Международной научно-практической конференции/ отв. ред. П. В. Евтин. – Тюмень: ТИУ, 2018. – С.31-33.
6. Rational refining of heavier cut of pyrolysis gas oil of hydrocarbon feed/Yu.A. Bulauka, V.A. Liakhovich, D.S. Yukhno, S.F. Yakubouski // Сборник докладов 72-й Международной молодежной научной конференции «Нефть и газ – 2018» (23-26 апреля 2018 г. Москва). – Том 3. – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. –С.293.
7. Получение нафталина - нового для белорусского рынка продукта малотоннажной химии /Ю.А. Булавка, С.Ф. Якубовский, В.А. Ляхович// Сборник материалов 4-го Белорусско-Балтийского форума «Сотрудничество – катализатор инновационного роста», Минск, 31 мая-1 июня 2018 года, г. – Минск: БНТУ, 2018. – С.62-63.
8. Направления использования тяжёлой смолы пиролиза /Е.А. Шульга, Н.С. Вашкова//Электронный сборник трудов молодых специалистов Полоцкий государственный университет. Сер. Промышленность. – Выпуск 30(100). – С.336-337
9. Сравнительная оценка растворяющей способности углеводородов и спиртов по отношению к нафталину/Якубовский С.Ф., Булавка Ю.А., Казак Е.В.//Вестник Полоцкого государственного университета. Серия В, Промышленность. Прикладные науки. - 2016. - № 3. - С. 160-163.
10. Сольватирующая способность растворителей различной природы по отношению к нафталину/ Е.В. Казак, С.Ф. Якубовский, Ю.А. Булавка//Химия и жизнь: сб. тез. и докл. междунар. науч. -практ. конф. / Новосибир. гос. аграр. ун-т.– Новосибирск, 2016. – С. 206-209.

11. Азеотропная ректификация нафталинсодержащей фракции тяжелой смолы пиролиза /Е.В. Казак, С.Ф. Якубовский, Ю.А. Булавка// Сборник научных работ студентов Республики Беларусь «НИРС 2016» / В. Л. Богуш (председатель) [и др.]. – Минск : Издательский центр БГУ, 2017. - С. 97-98
12. Получение нового для белорусского рынка продукта нефтехимии – нафталина/ Ю.А. Булавка, С. Ф. Якубовский, С. С. Хохотов//Горизонты и перспективы нефтехимии и органического синтеза: материалы Международной научной конференции-Уфа: Изд-во «Реактив», 2018. – С.138-139.
13. Разработка эффективной технологии извлечения нафталина из тяжелой смолы пиролиза /Ю.А. Булавка, С.Ф. Якубовский// Инновационные материалы и технологии: материалы докладов Международной научно-технической конференции молодых ученых. – Минск: БГТУ, -2019. - С.211-214
14. Rational refining of heavier cut of pyrolysis gas oil / Bulauka Yu., Yakubouski S.// Abstract book of 10th International Youth Scientific and Practical Congress «Oil and Gas Horizons», Moscow, November 19-22, 2018.- Gubkin Russian State University of Oil and Gas (National Research University).-Moscow, 2018.-p.61
15. Process to extract high purity naphthalene from the heavier gas oil fraction from naphtha crackers producing ethylene/Y.A. Bulauka, S.F. Yakubouski// Scientific Conference Abstracts of XV International Forum-Contest of Students and Young Researchers « Topical issues of rational use of natural resources», St. Petersburg, May 13-17, 2019.- Saint-Petersburg Mining University. - St. Petersburg,2019.- P. 24
16. PGO Processing with azeotropic rectification to extract naphthalene /Y.A. Bulauka, S.F. Yakubouski// Topical Issues of Rational Use of Natural Resources 2019- Litvinenko (Ed), 2020 Taylor & Francis Group, London, ISBN 978-0-367-85720-2, CAT# 349509 <https://doi.org/10.1201/9781003014638>, .-Volume 2 - P.495-501.
17. Variety of heavy residual oil applications /Shulha A., Vashkova N., Jakubovskij S.//European and national dimension in research. Technology = Европейский и национальный контексты в научных исследованиях: Electronic collected materials of XI Junior Researchers' Conference, Novopolotsk, May 23-24, 2019 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2019. -P.96-97

УДК 661.715.6

**ПУТИ РАЦИОНАЛЬНОГО ИСПОЛЬЗОВАНИЯ ТЯЖЁЛОЙ СМОЛЫ ПИРОЛИЗА.
ПОЛУЧЕНИЕ НЕФТЯНОГО КОКСА И СУПЕРПЛАСТИФИКАТОРА
ДЛЯ ЦЕМЕНТНЫХ СИСТЕМ**

Е.А. ШУЛЬГА

(Представлено: канд. хим. наук, доц. С.Ф. ЯКУБОВСКИЙ)

Выполнен комплексный анализ физико-химических свойств остаточных фракций тяжелой смолы пиролиза завода «Полимир» ОАО «Нафтан». Определено, что рассматриваемые фракции характеризуются комплексом свойств достаточных для их использования в качестве сырья коксования, а также для использования сырья при получении пластификаторов цементных систем. Выполнен в лабораторных условиях процесс коксования и изучена микроструктура полученных остатков коксования.

Тяжелая смола пиролиза (ТСП), образующаяся в процессе промышленного производства низших алкенов термическим пиролизом газов, прямогонного и легкого бензинов, атмосферного и вакуумного газойлей, представляет собой смесь конденсированных алкил- и алкенилароматических углеводородов с двумя и более циклами, содержит также олигомеры алкенилароматических углеводородов, асфальтены и другие высокомолекулярные соединения, выкипающих при температуре выше 170°C. Только в России производство тяжелой смолы пиролиза составляет от 243 000 до 325 000 тонн в год [1-15]. В настоящее время рационального применения не находит, в большинстве случаев используется неэффективно, как компонент котельного топлива [16].

Разработка новых научно обоснованных способов рационального использования тяжелой смолы пиролиза и получение технических материалов, таких как нефтяной кокс, пластификатор для цементных систем с комплексом требуемых свойств является актуальной научной задачей и положительно скажется на эффективности процесса переработки тяжелой смолы пиролиза, и, как следствие, позволит повысить рентабельность этиленовых производств [16-20].

На начальном этапе исследования выполнена разгонка тяжелой смолы пиролиза производства завода «Полимир» ОАО «Нафтан» по Энглеру в соответствии с ГОСТ 2177-82 выделены углеводородные фракции: н.к.-220°C и 220-кк; н.к.-230°C и 230-кк; н.к.-240°C и 240-кк.

Проведен анализ физико-химических свойств остаточных фракций 220-кк, 230-кк, 240-кк тяжелой смолы пиролиза производства завода «Полимир» ОАО «Нафтан»: определены вещества нерастворимых в толуоле (α -фракция), содержание асфальтенов по методу Маркусона по ГОСТ 11858, проведена экстракция каменноугольного пека смесью растворителей, определена глубина проникновения иглы и степень размягчения каменноугольного остатка, коксуемость ГОСТ ISO 10370-2015, а так же был определен газохроматографический состав исследуемых фракций.

Анализ результатов разгонки тяжелой смолы пиролиза по Энглеру показал, что около 50% приходится на твердый кубовый остаток,

Состав остаточного продукта оптимален для его использования в качестве сырья коксования для получения нефтяного кокса. Содержание веществ, нерастворимых в толуоле (α -фракция) в исследуемых объектах минимально (не более 0,2% мас.), таким образом сырье является пригодным для получения нефтяного кокса. Коксуемость, определенная микрометодом составила: для фракции ТСП 220-кк – 16,2 %; для фракции ТСП 230-кк - 20%; для фракции ТСП 240-кк - 23%. Общеизвестно, что повышенная коксуемость остатков – это основное требование, предъявляемое к качеству сырья установок коксования: коксуемость должна составлять не менее 14...20%. Таким образом, остаточные фракции тяжелой смолы пиролиза производства завода «Полимир» ОАО «Нафтан» характеризуются комплексом свойств достаточных для их использования в качестве сырья коксования.

Полученные фракции тяжелой смолы пиролиза были проанализированы с помощью газовой хроматографии в ЦЗЛ ОАО «Нафтан» согласно европейскому стандарту по ИОР 744-86 «Определение ароматических соединений в углеводородах методом газовой хроматографии». Анализ приведённых данных показывает, что с увеличением температуры отбора фракции наблюдается уменьшение содержания н-парафиновых углеводородов и инданов. Такие углеводороды как изопарафины и нафтены присутствуют во всех исследуемых фракциях, однако наибольшее их количество сконцентрировано во фракции 180 – 210°C. Таким образом данные дают сведения о том, что можно использовать фр. 210-230°C в качестве сырья для получения суперпластификатора для цементных систем.

Выполнен в лабораторных условиях процесс коксования остаточных фракции 220-кк, 230-кк, 240-кк ТСП производства завода «Полимир» ОАО «Нафтан» при температуре коксования 500°C и продолжительности 30 минут. Проведен анализ микроструктуры полученного остатков коксования при помощи

атомно-силового микроскопов микроскопа NT-206 при общем увеличении до 30 нм. Результаты микро-структурного анализа поверхности показали (рис.1), что структура нефтяного кокса изотропная, с утяжелением сырья появляется анизотропные включения. Наблюдаются единичные элементы похожие на трубки, возможно, фрагменты игольчатого кокса.

Так же в лабораторных условиях был получен суперпластификатор для цементных систем из тяжёлой смолы пиролиза. Была выбрана технология смешения алкилбензолов и ТСП, нагрев до 160°C, после чего к ним была добавлена серная кислота. Оптимальное время сульфирования 30 минут. Результаты исследования представлены в таблице 1.

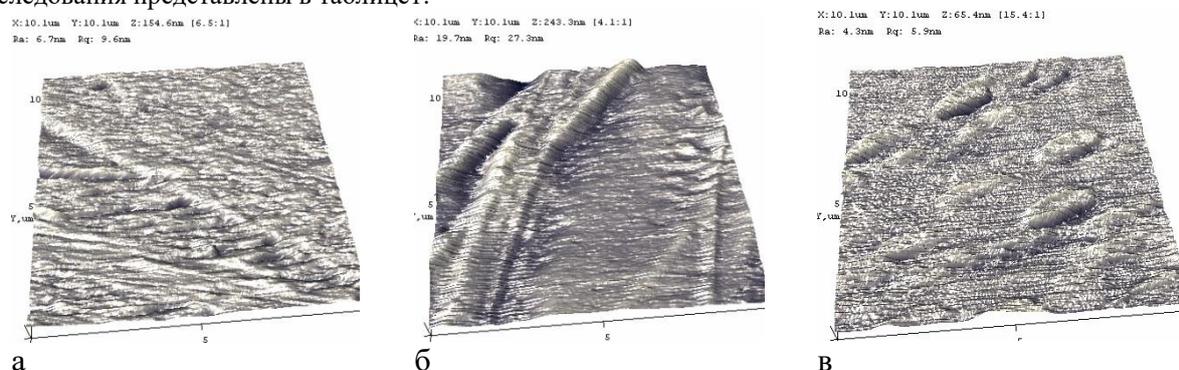


Рисунок 1. – Поверхность остатков коксования фр. ТСП: а–220-кк, б–230-кк, в–240-кк

Таблица 1. – Анализ полученного скперпластификатора для цементных систем

№ опыта	Сульфлируемая фракция, °С	Алкилбензолы С ₁₀₊ , мл	Кислота H ₂ SO ₄ , мл	Расплав, мм	
				на 0,2	на 0,4
1	Исходная ТСП, 10 мл	5	12	35x44	85x70
2	Исходная ТСП, 10 мл	-	8	45x50	53x48
3	180 – 210, 10 мл	5	12	45x45	50x50
4	180 – 210, 10 мл	-	8	52x60	40x40
5	210 – 220, 10 мл	5	12	40x42	52x55
6	210 – 220, 10 мл	-	8	45x40	35x35
7	210 – 230, 10 мл	5	12	66x69	70x65

Проведенные исследования позволили прийти к выводу, что эффективным альтернативным направлением рационального использования тяжелой смолы пиролиза производства завода «Полимир» ОАО «Нафтан», является его использование в качестве сырья коксования для получения нефтяного кокса, а также при получении суперпластификатора для цементных систем.

ЛИТЕРАТУРА

1. Инновационный подход к переработке тяжелой смолы пиролиза углеводородного сырья /Булавка Ю.А., Якубовский С.Ф., Хохотов С.С., Ляхович В.А.// Актуальные проблемы развития нефтегазового комплекса России: Сборник тезисов XII Всероссийская научно-техническая конференция (12-14 февраля 2018 г., г. Москва). – РГУ нефти и газа (НИУ) имени И.М. Губкина. – Москва, 2018.-С.209.
2. Рациональная переработка тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка, В.А. Ляхович, С.Ф.Якубовский // Tatarstan UpExPro 2018: материалы II Международной молодежной конференции (14–17 февраля 2018 г., Казань). – Казань: Изд-во Казан. ун-та, 2018. –С.120-121.
3. Инновационный подход к переработке тяжелой смолы пиролиза углеводородного сырья /Булавка Ю.А., Якубовский С.Ф., Хохотов С.С., Ляхович В.А.// Сборник трудов XII Всероссийской научно-технической конференции «Актуальные проблемы развития нефтегазового комплекса России». – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. –С.23-26.
4. Получение товарных продуктов из тяжелой смолы пиролиза / Булавка Ю.А., Якубовский С.Ф., Ляхович В.А.// Актуальные вопросы современного химического и биохимического материаловедения: материалы V Международной молодежной научно-практической школы-конференции (г. Уфа, 4-5 июня 2018 г.) / отв. ред. О.С. Куковинцев. - Уфа: РИЦ БашГУ, 2018-С. 54-57.
5. Современные направления переработки тяжелой смолы пиролиза углеводородного сырья /Ю.А. Булавка, В.А. Ляхович, А.С. Москаленко// Новые технологии – нефтегазовому региону: материалы Международной научно-практической конференции/ отв. ред. П. В. Евтин. – Тюмень: ТИУ, 2018. – С.31-33.

6. Rational refining of heavier cut of pyrolysis gas oil of hydrocarbon feed/Yu.A. Bulauka, V.A. Liakhovich, D.S. Yukhno, S.F. Yakubouski // Сборник докладов 72-й Международной молодежной научной конференции «Нефть и газ – 2018» (23-26 апреля 2018 г. Москва). – Том 3. – М.: Издательский центр РГУ нефти и газа (НИУ) имени И.М. Губкина, 2018. – С.293.
7. Получение нафталина - нового для белорусского рынка продукта малотоннажной химии /Ю.А. Булавка, С.Ф. Якубовский, В.А. Ляхович// Сборник материалов 4-го Белорусско-Балтийского форума «Сотрудничество – катализатор инновационного роста», Минск, 31 мая-1 июня 2018 года, г. – Минск: БНТУ, 2018. – С.62-63.
8. Направления использования тяжелой смолы пиролиза /Е.А. Шульга, Н.С. Вашкова//Электронный сборник трудов молодых специалистов Полоцкий государственный университет. Сер. Промышленность. – Выпуск 30(100). – С.336-337.
9. Сравнительная оценка растворяющей способности углеводородов и спиртов по отношению к нафталину/Якубовский С.Ф., Булавка Ю.А., Казак Е.В.//Вестник Полоцкого государственного университета. Серия В, Промышленность. Прикладные науки. - 2016. - № 3. - С. 160-163.
10. Сольватирующая способность растворителей различной природы по отношению к нафталину/ Е.В. Казак, С.Ф. Якубовский, Ю.А. Булавка//Химия и жизнь: сб. тез. и докл. междунар. науч. -практ. конф. / Новосибир. гос. аграр. ун-т.– Новосибирск, 2016. – С. 206-209.
11. Азеотропная ректификация нафталинсодержащей фракции тяжелой смолы пиролиза /Е.В. Казак, С.Ф. Якубовский, Ю.А. Булавка// Сборник научных работ студентов Республики Беларусь «НИРС 2016» / В. Л. Богуш (председатель) [и др.] - Минск : Издательский центр БГУ, 2017. - С. 97-98.
12. Получение нового для белорусского рынка продукта нефтехимии – нафталина/ Ю.А. Булавка, С. Ф. Якубовский, С. С. Хохотов//Горизонты и перспективы нефтехимии и органического синтеза: материалы Международной научной конференции-Уфа: Изд-во «Реактив», 2018. – С.138-139.
13. Разработка эффективной технологии извлечения нафталина из тяжелой смолы пиролиза /Ю.А. Булавка, С.Ф. Якубовский// Инновационные материалы и технологии: материалы докладов Международной научно-технической конференции молодых ученых. – Минск: БГТУ, -2019. – С.211-214.
14. Rational refining of heavier cut of pyrolysis gas oil / Bulauka Yu., Yakubouski S.// Abstract book of 10th International Youth Scientific and Practical Congress «Oil and Gas Horizons», Moscow, November 19-22, 2018.- Gubkin Russian State University of Oil and Gas (National Research University). – Moscow, 2018. – p.61.
15. Process to extract high purity naphthalene from the heavier gas oil fraction from naphtha crackers producing ethylene/Y.A. Bulauka, S.F. Yakubouski// Scientific Conference Abstracts of XV International Forum-Contest of Students and Young Researchers « Topical issues of rational use of natural resources», St. Petersburg, May 13-17, 2019.- Saint-Petersburg Mining University. - St. Petersburg,2019. – P. 24
16. PGO Processing with azeotropic rectification to extract naphthalene /Y.A. Bulauka, S.F. Yakubouski// Topical Issues of Rational Use of Natural Resources 2019- Litvinenko (Ed) , 2020 Taylor & Francis Group, London, ISBN 978-0-367-85720-2, CAT# 349509 <https://doi.org/10.1201/9781003014638>, -Volume 2 - P.495-501.
17. Variety of heavy residual oil applications /Shulha A., Vashkova N., Jakubovskij S.//European and national dimension in research. Technology = Европейский и национальный контексты в научных исследованиях: Electronic collected materials of XI Junior Researchers' Conference, Novopolotsk, May 23-24, 2019 / Polotsk State University; ed. D. Lazouski [et al.]. – Novopolotsk, 2019. – P.96-97.
18. Перспектива расширения сырьевой базы для получения игольчатого кокса / И.Р. Хайрудинов, А.А. Тихонов, М.М. Ахметов // Башкирский химический журнал. – 2011. – №3.
19. Нефтяной кокс. / Красюков А.Ф. – М. Химия, 1966. – 264 с.
20. Нефтяной кокс. / Гимаев Р.Н. – М.: Химия, 1992. – 80 с.

СОДЕРЖАНИЕ

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Баклан В.И. Использование распределенной обработки данных при разработке приложения «автоматизированная система коммуникаций студентов, администрации и потенциальных работодателей»	3
Баклан В.И. Использование технологии ci/cd при разработке приложения «автоматизированная система коммуникаций студентов, администрации и потенциальных работодателей»	5
Бельский Б.Г. Портфолио как технология оценивания профессиональных компетенций студентов	7
Бельский Б.Г. Технология «Портфолио» в компетентно-ориентированном образовании	10
Борисенок К.С. Проектирование интерфейсов программного обеспечения магазина автозапчастей	12
Борисенок К.С. Проектирование программного продукта магазина автозапчастей	15
Будько Д.Д. Проектирование и исследование антенны СВЧ диапазона	18
Будько Д.Д. Разработка и моделирование антенны для СВЧ диапазона	20
Виноградова А.Д. Проектирование интерфейса веб-приложения «Продажа автомобилей «Angelcar»	24
Виноградова А.Д. Преимущества веб-приложения для автоматизированного учета продаж и поставок автомобилей «Angelcar»	26
Гарист Е.Ю., Сияк П.В. Создание трёхмерных моделей и аудиоряда для программно-аппаратного VR-тренажёра по отработке навыков поведения в чрезвычайных ситуациях	30
Гарист Е.Ю., Сияк П.В. Создание логики взаимодействий в программно-аппаратном VR-тренажёре по отработке навыков поведения в чрезвычайных ситуациях в среде разработки Unreal Engine 4	34
Герец А.Ю. Основные принципы вывода на рынок мобильного приложения по поиску партнеров для занятия спортом	37
Герец А.Ю. Анализ целевой аудитории и экономическая целесообразность выхода на рынок мобильного приложения по поиску партнеров для занятия спортом	40
Градюшко М.Р. Проектирование базы данных для музыкального стримингового веб-сервиса	45
Градюшко М.Р. Разработка музыкального стримингового веб-сервиса	48
Гуртовенко Н.В. Test-Driven Development – разработка через тестирование: преимущества и недостатки	50
Гуртовенко Н.В. Жизненный цикл тестирования ПО. Преимущества и шаги	52
Деревяга П.В. Проектирование интерфейса веб-приложения информационной системы для почтовых отделений	54
Деревяга П.В. Разработка веб-API и базы данных веб-приложения информационной системы для почтовых отделений	56
Дмитриченко П.В. Проблемы применения машинного обучения	58
Дмитриченко П.В. Успешные примеры применения машинного обучения и причины их успеха	60
Дубовский Н.В. Разработка веб-интерфейса с использованием фреймворка Angular для ресурса «веб-платформа для социального взаимодействия пользователей»	62
Дубовский Н.В. Веб-платформа для социального взаимодействия пользователей	64
Епанешников А.В. Обзор фреймворка Xamarin	66
Епанешников А.В. Swift – язык, который изменит мир программирования	68
Змитрович С.Ю. Создание автоматизированных устройств для физических измерений	70
Змитрович С.Ю. Использование микроконтроллеров для обработки аналоговых сигналов	72
Исаков И.В. Информационная безопасность системы «Умный дом»	75
Исаков И.В. Автоматизация бизнеса на основе CRM-систем	77

<i>Ковалевский В.К.</i> Способы монтажа солнечного коллектора.....	80
<i>Ковалевский В.К.</i> Устройство слежения за солнечными лучами	82
<i>Коняев Е.В.</i> Принципы разработки и шаблоны проектирования приложений под операционную систему iOS	84
<i>Коняев Е.В.</i> Сравнение быстродействия языков программирования под платформу iOS	87
<i>Кочанов А.П.</i> Распространение радиоимпульсных сигналов над углеводородными залежами	90
<i>Кочанов А.П.</i> Влияние частоты радиоимпульсного сигнала, диэлектрической проницаемости среды на компоненты тензора диэлектрической проницаемости.....	94
<i>Лесницкий В.С.</i> Машинное обучение – основа искусственного интеллекта.....	99
<i>Лесницкий В.С.</i> Типы компьютерных сетей – LAN, MAN и WAN.....	101
<i>Маркевич К.В.</i> Подходы к разработке интерфейса ассистента для управления умным домом.....	103
<i>Маркевич К.В.</i> Проектирование ассистента для управления умным домом	105
<i>Медведев Н.А.</i> Методы привлечения и поддержки пользователей в интернет-сервисе	107
<i>Медведев Н.А.</i> Защита интернет-системы от взлома посторонними лицами	110
<i>Мисевич И.В.</i> Описание общих принципов и рекомендаций по защите от некоторых видов вирусных атак	113
<i>Мисевич И.В.</i> Большие данные и машинное оборудование.....	116
<i>Михнович О.С.</i> Разработка обучающего программного средства «Основы предпринимательства».....	119
<i>Михнович О.С.</i> Возможности языка программирования Crystal	121
<i>Морозов Д.Д.</i> VPN и его применение	123
<i>Морозов Д.Д.</i> Сайт и его составляющие	125
<i>Непесов М.</i> Проектирование web-приложения «Fantasy» для развития креативности у детей школьного возраста	127
<i>Непесов М., Абдуллаев Р.</i> Разработка и организация разработки обучающего web-приложения «Fantasy»	130
<i>Попов А.И.</i> Исследование технологии взаимодействия с базой данных для автоматизированной информационной системы туристического центра	133
<i>Попов А.И.</i> Использование архитектурного паттерна MVC при проектировании системы туристического центра	136
<i>Сергеев М.А.</i> Реализация интерактивного изменения материала объекта в игровом приложении	139
<i>Сергеев М.А.</i> Реализация головоломки «Пазл» в игровом приложении виртуальной реальности	142
<i>Сивограков А.С.</i> Технологии обеспечения безопасности информации при передаче данных между несколькими предприятиями по сети Интернет.....	144
<i>Сивограков А.С.</i> Электронная подпись: преимущества и недостатки	147
<i>Скуковская А.А., Астукевич А.Ф.</i> Система «Умный дом».....	150
<i>Скуковская А.А., Астукевич А.Ф.</i> Северная часть системы «Умный дом»	153
<i>Солоцкий А.Ю.</i> Графические символы в социализации людей с ограниченными возможностями.....	156
<i>Солоцкий А.Ю.</i> Разработка административных интерфейсов	159
<i>Стрельченко А.И.</i> Проектирование и разработка WEB-сайта в сети Интернет на примере города Бешенковичи	162
<i>Стрельченко А.И.</i> Поисковая оптимизация WEB-сайта SEO (Search Engine Optimization).....	164
<i>Сущевский Д.В.</i> Enum и Switch, и проблемы их использования.....	166
<i>Сущевский Д.В.</i> Преимущества использования Microsoft Teams	171

<i>Талайко П.Д.</i> Микросервисные асинхронные архитектуры как решение сложных бизнес-процессов	174
<i>Талайко П.Д.</i> Миграция баз данных в современной разработке программного обеспечения	177
<i>Танана А.С.</i> Применение дискретного косинусного преобразования и генератора псевдослучайной последовательности в алгоритмах цифровой стеганографии	180
<i>Танана А.С.</i> Проектирование графического интерфейса пользователя для встраивания информации в аудиофайл на основе модификации спектральных коэффициентов. оценка стеганостойкости системы.....	183
<i>Татарин Д.С.</i> Преимущества использования архитектурного стиля взаимодействия Representational State Transfer	186
<i>Татарин Д.С.</i> Основные технологии Spring Framework.....	189
<i>Шершнев Н.О.</i> Проектирование базы данных для мобильного приложения «Музыканты России».....	192
<i>Шершнев Н.О.</i> Использование сервиса уведомлений PushWoosh в мобильном приложении «Музыканты России».....	195
<i>Шумянец И.А.</i> Проектирование интерфейсов программного обеспечения торговой площадки для продажи продуктов	198
<i>Шумянец И.А.</i> Проектирование программного продукта торговой площадки для продажи продуктов	202

МАШИНОСТРОЕНИЕ

<i>Анисимов В.С., Кураш Н.В., Швецов Г.И.</i> Влияние угла наклона режущей кромки инструмента на характеристики обработки плоской поверхности детали на станке с ЧПУ	205
<i>Гасс Ю.А., Лопатин Д.В.</i> Повышение эффективности диагностирования электронных систем управления двигателями	208
<i>Гасс Ю.А., Лопатин Д.В.</i> Повышение эффективности использования осциллографа при диагностировании двигателя современного автомобиля	211
<i>Ефременко Е.А., Прудникова А.С.</i> Методы прямого впрыска воды в рабочее пространство двигателя	214
<i>Ефременко Е.А., Прудникова А.С.</i> Прямой впрыск воды в рабочее пространство двигателя	217
<i>Пуйман Е.В.</i> Исследование возможностей применения покрытий из инструментальных сталей для повышения стойкости режущего инструмента	219
<i>Скотик С.А., Дмитриев Р.Г., Бедрицкая Е.С., Кудрякова В.А., Глинский Е.А., Матвеенков Д.Е., Портянко А.А.</i> Моделирование влияния теплового воздействия на основные элементы режущего инструмента в SolidWorks.....	222
<i>Скотик С.А., Дмитриев Р.Г., Бедрицкая Е.С., Кудрякова В.А., Глинский Е.А., Матвеенков Д.Е., Портянко А.А.</i> Моделирование работы торцовых фрез в программе Cinema 4D	226
<i>Скотик С.А., Дмитриев Р.Г., Бедрицкая Е.С., Кудрякова В.А., Глинский Е.А., Матвеенков Д.Е., Портянко А.А.</i> Сравнительный анализ 3D принтеров на основе FDM и SLA технологий	232
<i>Скотик С.А., Дмитриев Р.Г., Бедрицкая Е.С., Кудрякова В.А., Глинский Е.А., Матвеенков Д.Е., Портянко А.А.</i> Моделирование воздушных потоков в SolidWorks FlowSimulation	237
<i>Царькова Е.С.</i> Координированное управление дорожным движением «Зеленая волна» в городе Новополоцке.....	242
<i>Царькова Е.С.</i> Транспортное моделирование, четырехшаговый алгоритм транспортного моделирования	244

<i>Шаурко М.М., Демкин А.А.</i> Анализ способов обработки сферических поверхностей деталей на токарных станках с ЧПУ	249
<i>Шевченко Д.С.</i> Технологии диагностирования двигателей внутреннего сгорания	252
<i>Шевченко Д.С.</i> Снижение трудоемкости диагностирования двигателей внутреннего сгорания	256

ХИМИЧЕСКИЕ ТЕХНОЛОГИИ

<i>Ляхович В.А.</i> Способы подавления пылеобразования при транспортировке углеродсодержащих материалов	260
<i>Ляхович В.А.</i> Способ получения пылеподавляюще-противосмерзающего средства, снижающего воздействие пыли нефтяного кокса на работников	263
<i>Прашкович Я.В., Троц В.А.</i> Определение фракционного состава дизельных топлив в смеси с подсолнечным маслом	268
<i>Шульга Е.А.</i> Получение нефтяного кокса на основе тяжелой смолы пиролиза завода «Полимир» ОАО «Нафтан»	272
<i>Шульга Е.А.</i> Пути рационального использования тяжелой смолы пиролиза. Получение нефтяного кокса и суперпластификатора для цементных систем	275