

УДК 004.021

РАЗРАБОТКА КОМПЛЕКСНОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПРОЕКТАМИ ИТ-КОМПАНИЙ

А.А. ЛЕВИЦКИЙ

(Представлено: Т.С. РУДЬКОВА)

Предлагается разработка программного обеспечения, созданного для управления проектами ИТ-компаний, которая позволит компаниям управлять своими сотрудниками, а также регулировать задания заказчиков. Анализируются задачи; показан принцип работы и алгоритмы реализации.

При больших темпах роста информационных технологий появляются все больше компаний, которые недобросовестно относятся к своей работе, а также к заказчикам. Заказчики начали тщательнее выбирать исполнителей, вследствие чего молодым компаниям стало сложнее на рынке. Поэтому для более прозрачной работы появилась необходимость в отслеживании работы непосредственно заказчиком. Для более быстрого решения проблем и вопросов, стоящих перед исполнителями и заказчиками, появилась необходимость в чате между каждым из участников проекта.

Так как справляться с наплывами заказов и постоянного роста компаний обычными средствами стало сложнее, то появилась необходимость в разработке централизованной системы для организации слаженной работы команд над разработкой проектов, а также для взаимодействия с заказчиками. Она позволит оценить вклад конкретного разработчика в проект, а также с большим удобством работать исполнителям на удаленной основе.

Анализ задачи. Основным преимуществом сервиса-конкурента «Trello» можно назвать просмотр нескольких одновременно запущенных проектов и их состояние в данный момент. Для работы с группой разработчиков или иных исполнителей, работающих над проектами с конечной датой выполнения или целью, данная система может дать вам представление о ходе выполнения проектов.

Особенностью разработанной системы является то, что каждый пользователь может быстро связаться с другим пользователем посредством чата в реальном времени без необходимости писать непосредственно на корпоративную почту. Например, если заказчик захочет обсудить какие-либо правки, но сделать это нужно очень быстро вследствие занятости, то чат подойдет как нельзя лучше для таких целей. Также модуль позволит разграничить обязанности каждого пользователя.

Выбор языка программирования – один из важнейших аспектов любого проекта. NodeJS – серверная реализация языка программирования JavaScript, основанная на движке V8; предназначена для создания масштабируемых распределенных сетевых приложений, таких как веб-сервер. NodeJS по целям использования сходен с каркасами Twisted на языке Python и EventMachine на Ruby. В отличие от большинства программ JavaScript, этот каркас выполняется не в браузере клиента, а на стороне сервера. Так как NodeJS использует JavaScript синтаксис, то разработчику веб-сайтов легче переключаться между написанием клиентской части, написанной на JavaScript, и серверной частью, написанной на NodeJS. NodeJS является асинхронным языком программирования, построенный на модели событий, что позволяет выполнять код все зависимости от порядка вызовов функций. NodeJS выполняется в одном потоке, но асинхронность достигается путем подписки на события и вызова кластеров [1].

Выбор СУБД пал на документно-ориентированную систему управления базами данных MongoDB с открытым исходным кодом, не требующая описания схемы таблиц. Данная СУБД классифицирована как NoSQL, она использует JSON-подобные документы и схему базы данных. Скорость обеспечивается также тем, что MongoDB написана на языке C++. Запросы могут возвращать конкретные поля документов и пользовательские JavaScript-функции. Поддерживается поиск по регулярным выражениям.

MongoDB может работать с набором реплик, которые состоят из двух и более копий данных. Каждый экземпляр такого набора может в любой момент выступать в роли основной или вспомогательной реплики. Все операции записи и чтения по умолчанию осуществляются с основной репликой, а вспомогательные поддерживают в актуальном состоянии копии данных. В случае, когда основная реплика дает сбой, данный набор проводит выбор, который из реплик должен стать основным. Таким образом, второстепенные реплики могут дополнительно являться источником для операций чтения [2]. Все рассмотренные выше преимущества объясняют выбор MongoDB в качестве СУБД в разрабатываемом приложении.

Принцип работы и алгоритмы реализации. Система разделена на несколько подмодулей (рис. 1), каждый из которых отвечает за отдельный от остальных функционал.

В редактировании профиля можно изменять такие личные данные, как имя, фамилия, электронная почта, номер телефона, а также загружать фотографию, которая будет отображаться в левой панели сайта текущего пользователя, а также в его профиле (рис. 2). Для загрузки фотографии была использована библиотека «jQuery-Fileuploader», которая позволяет просто управлять процессом загрузки на JavaScript.

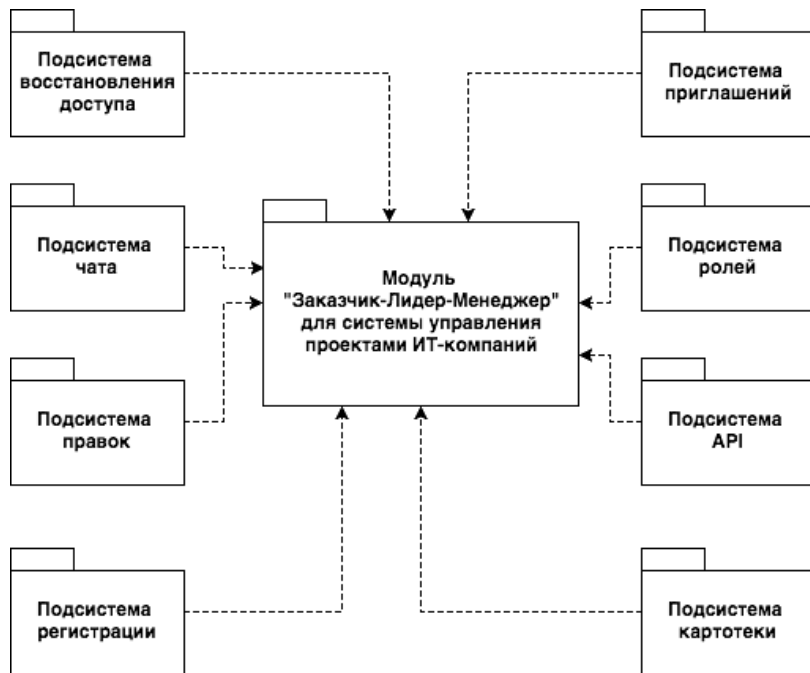


Рисунок 1. – Модули системы управления проектами

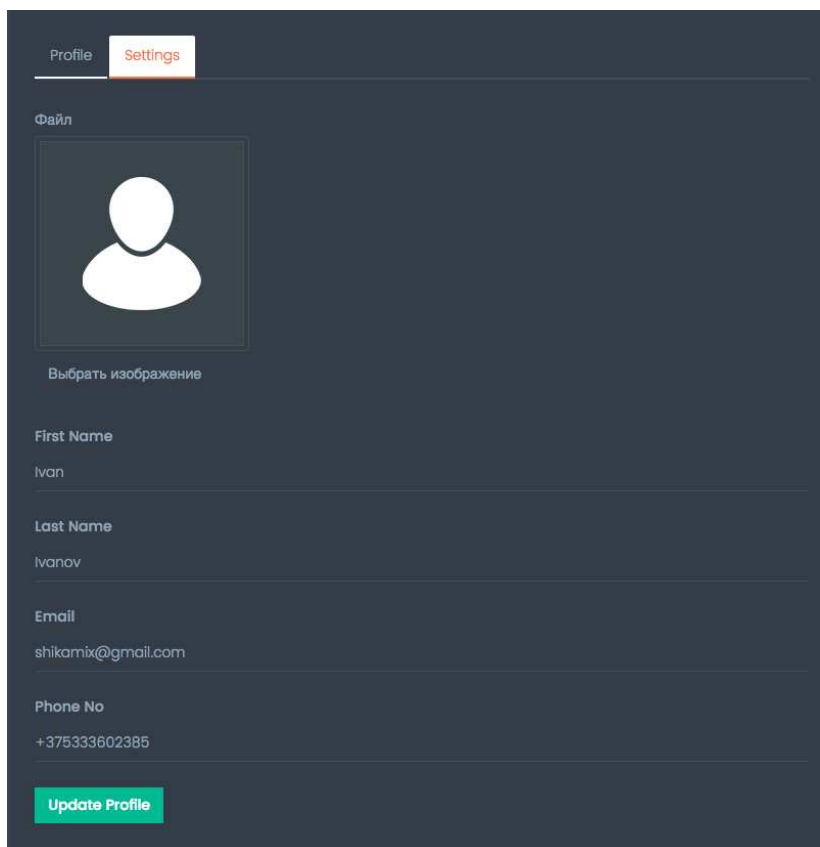


Рисунок 2. – Редактирование профиля

Страница личных сообщений реализована с помощью технологии веб-сокетов. Она позволяет отправлять и принимать любые пакеты данных мгновенно без перезагрузки страницы. Для этого помимо клиентской части также требуется серверная реализация, которая позволит перенаправлять сообщения определенному пользователю в зависимости от полученной информации. Для реализации чата подобного типа была использована библиотека «SocketIO», которая включает в себя реализацию для клиентской

и серверной части веб-приложения. Если у адресата и адресанта открыты чаты друг с другом, то сервер понимает, что эти два пользователя общаются в данный момент. Для большей информативности еще до отправки, при наборе сообщений одним пользователем, другой пользователь видит, что его собеседник набирает сообщение. Это может быть полезно в том случае, если пользователь ждет моментального ответа, но не уверен, пользователь в сети или нет. Для того чтобы увидеть сообщения собеседника, необходимо кликнуть на его имя в левой панели чата. После нажатия будут загружены прошлые сообщения, а также непрочитанные. Все сообщения отсортированы по времени отправления так, что снизу выводятся последние (рис. 3). Слева выводятся сообщения собеседника, справа – сообщения текущего пользователя. По нажатию на фотографию собеседника, пользователь будет переадресован на личную страницу собеседника.

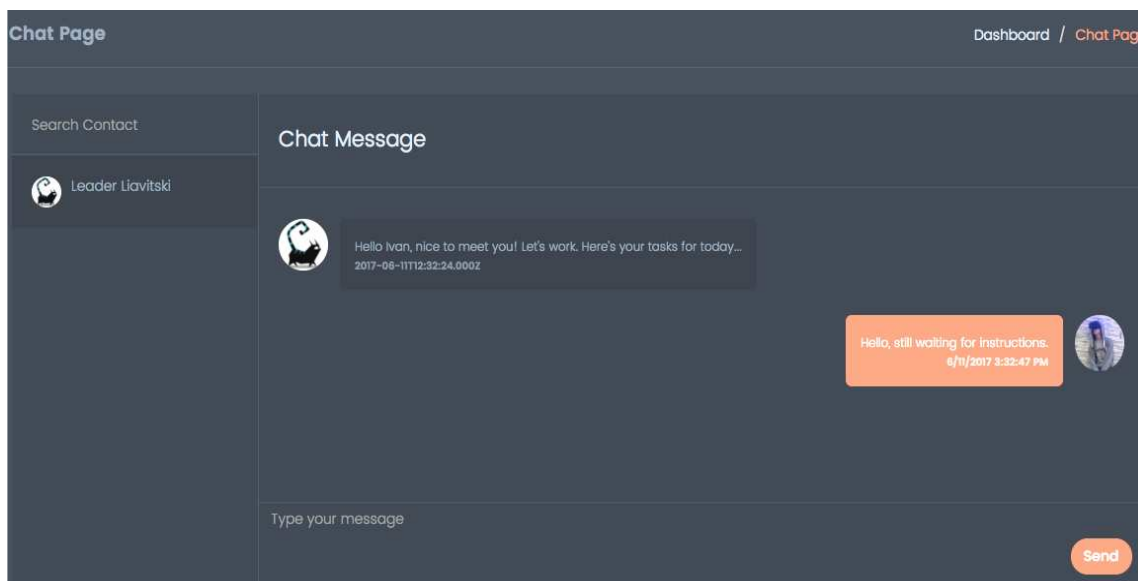


Рисунок 3. – Чат на WebSocket

В разделе «Invites» показаны текущие приглашения, которые еще не были активированы и по которым еще не зарегистрировались приглашенные пользователи. Если пользователь с введенным почтовым ящиком уже существует, то ему не будет отправлено приглашение повторно. При отправлении приглашения генерируется уникальная ссылка, при переходе на которую пользователь сможет зарегистрироваться. За один раз можно добавить неограниченное число сообщений, нажимая кнопку «Add more invites», при необходимости добавить очередное приглашение, при нажатии на которую добавится еще одна строка для ввода новых данных о пользователе. В таблице указать роль приглашаемого пользователя.

Заключение

Разработанная программная система с дружественным пользовательским интерфейсом как для исполнителей, так и для заказчиков позволит компаниям эффективно управлять заданиями заказчиков, эффективно вести переписку, назначать исполнителей, а также точно знать, какую именно задачу выполняет работник.

ЛИТЕРАТУРА

1. Node Package Manager [Электронный ресурс]. – Режим доступа: <https://npmjs.com>. – Дата доступа: 17.09.2017.
2. Базы данных [Электронный ресурс]. – Режим доступа: <http://www.bazydannyh.ru/>. – Дата доступа: 18.09.2017.