

УДК 535.214.6

**КЛИЕНТ-СЕРВЕРНОЕ WEB-ПРИЛОЖЕНИЕ ДЛЯ ОБРАБОТКИ ЗАЯВОК
СОТРУДНИКОВ МАЛОГО ПРЕДПРИЯТИЯ****Владимир Алексеевич Раевский,**ФГБОУ ВО «Калужский государственный университет им. К.Э. Циолковского», доцент
кафедры «Информатики и информационных технологий», г. Калуга, var-77@mail.ru**Ратников Роман Евгеньевич**ФГБОУ ВО «Калужский государственный университет им. К.Э. Циолковского», студент
кафедры «Информатики и информационных технологий», г. Калуга, ratnikovre@studklg.ru**Аннотация**

В статье приводится обзор информационных технологий, применяемых для управления заявками, обосновывается актуальность разработки web-приложения для управления заявками на малых предприятиях. Выбираются средства разработки приложения. Приводится схема данных для реализации реляционной базы данных, являющейся серверной частью приложения. Описывается разработка клиентской части приложения.

Ключевые слова: автоматизация создания и обработки заявок, web-приложение, клиент-серверное приложение, база данных, разработка клиентской части приложения, разработка серверной части приложения.

**CLIENT-SERVER WEB-APPLICATION FOR PROCESSING APPLICATIONS
OF SMALL EMPLOYEES****Vladimir. A. Raevsky,**

Kaluga State University named after K.E. Tsiolkovski, Associate Professor of the Department of Informatics and Information Technologies, Kaluga city, var-77@mail.ru.

Roman. E. Ratnikov

Kaluga State University named after K.E. Tsiolkovski, student of the Department of Informatics and information technologies, Kaluga city, ratnikovre@studklg.ru.

ABSTRACT

The article provides an overview of information technologies used to manage applications, substantiates the relevance of developing a web application for managing applications in small enterprises. Application development tools are selected. The data schema for the implementation of a relational database, which is the server part of the application, is given. Describes the development of the client part of the application.

Keywords: automation of the creation and processing of service request, web application, client-server application, database, development of the client part of the application, development of the server part of the application.

1. Обзор информационных технологий, применяемых для управления заявками. Обоснование актуальности разработки приложения. В настоящее время на современных предприятиях различного рода деятельности активно внедряются системы управления заявками как инструменты документального сопровождения цикла «Подготовка заявки → Согласование заявки → Выполнение заявки».

Ниже приведены наиболее распространенные методы обмена информацией, применимые для реализации цикла:

Средства голосовой связи. Пригодны для постановки и выполнения мгновенных поручений, так как при таком способе организации цикла отсутствует хронология и запись событий. Количество инициаторов/исполнителей заявки два и более приводит к усложнению взаимодействия между ними.

Программы мгновенного обмена сообщениями. Их применение обеспечивает наибольшую оперативность решения задач и проблем, но, тем не менее, является абсолютно непригодным при постановке стратегических или долгосрочных задач и поручений, так как история сообщений хранится одним большим потоком. Количество инициаторов/исполнителей заявки два и более также приводит к усложнению взаимодействия между ними.

Электронная почта. Обеспечивает требуемый уровень хронологии событий, но при постановке задач возникает проблема быстрого осуществления поиска и фильтрации задач.

Специализированные программы учета заявок. Анализ функционала подобных приложений [1-4 и др.]. показывает, что их внедрение позволяет сократить время и упростить процесс инициации заявок, упростить взаимодействие между инициатором и исполнителем заявки, контролировать вышеназванный цикл и при необходимости обмениваться сторонам дополнительной информацией.

Однако, обладая расширенным функционалом, подобные системы в первую очередь ориентированы на сферу обеспечения поддержки продаж и аутсорсинга для крупных компаний, которые имеют в своем штате как системных администраторов, так и администраторов данных систем. Стоимость тарифов может достигать 25000 руб. в месяц; бесплатные тарифы ограничены сравнительно малым числом исполнителей.

Исходя из вышесказанного, разработка клиент-серверного web-приложения для обработки заявок на малом предприятии, является актуальной задачей. Приложение должно обладать простейшим с точки зрения администрирования функционалом (администрирование сотрудниками уровня «пользователь ПК»), иметь возможность адаптации под конкретные задачи организации.

2. Выбор средств разработки web-приложения. Разрабатываемое приложение имеет клиент-серверную архитектуру, где клиентская и серверная части в корне различаются по реализации.

Для создания интернет-ресурсов в настоящее время активно осуществляется переход к средствам разработки RIA-приложений [5]. Однако следует учитывать, что многие малые предприятия имеют парк устаревшего вычислительного оборудования. Исходя из вышесказанного, для разработки web-страниц было решено использовать язык HTML5 [6-8], для оформления – формальный язык описания внешнего вида HTML-документа CSS3 [6-

8]. Разработка непосредственно клиентской части приложения осуществлялась с применением языка программирования JavaScript [7, 8].

Для серверной части авторами был выбран PHP версии 7 [8], что позволит развернуть приложение на web-сервере Apache [9, 10] как в среде ОС Windows, так и Linux-серверов. Функционал полностью удовлетворяет задачам, решаемым при разработке данного приложения.

Для хранения данных было решено использовать систему управления базами данных MySQL [8, 11], которая обладает простой и понятной структурой, позволяя при этом создавать базы данных для хранения больших объемов данных.

3. Разработка серверной части приложения. При реализации базы данных приложения было создано 8 взаимосвязанных таблиц MySQLInnoDB (рис. 1). Тип таблиц «InnoDB» был выбран исходя из тех соображений, что именно этот тип позволяет в полной мере обеспечить ссылочную целостность и ограничения (constraints) на связи между таблицами.

Основными таблицами, которые являются базовыми для разработанной системы, являются «users» и «requests».

Таблица «users» содержит сведения обо всех зарегистрированных в системе пользователях:

Первичный ключ (id).

Имя пользователя (name).

Фамилия пользователя (last-name).

Отчество пользователя (patronymic).

Электронная почта пользователя (E-mail).

Пара аутентификации «Пользователь» (login).

Пара аутентификации «Пароль» (password).

«Соль» к паролю (salt).

Телефон пользователя (phone).

Флаг административного доступа (is_admin).

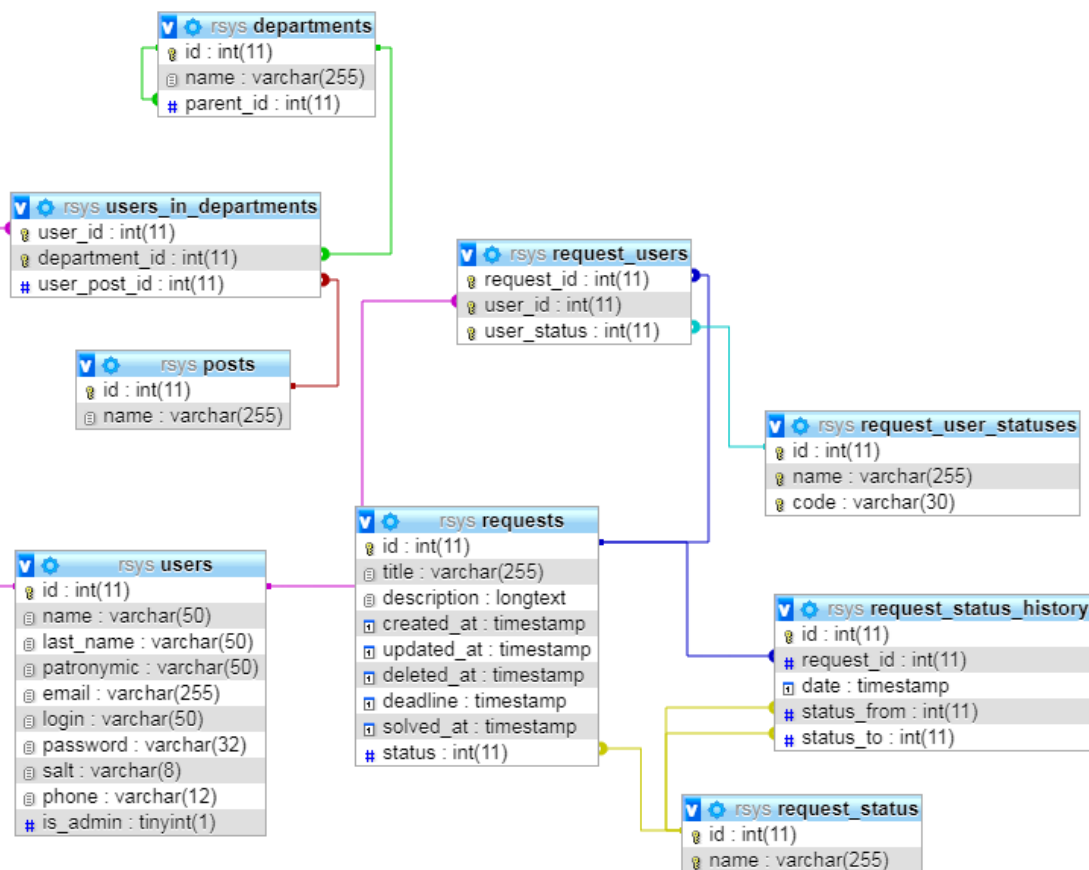


Рисунок 1 – Схема базы данных

Следует отметить, что с целью усиления безопасности хранения паролей, пароли хранятся в зашифрованном виде. Для шифрования используется метод вычисления контрольной суммы md5 с удлинением хэш-кода случайной последовательностью символов («солью» пароля).

Таблица «requests» содержит записи о созданных инициаторами заявках:

Первичный ключ (id).

Заголовок заявки (title).

Описание заявки (description).

Дата и время создания (created_at).

Дата изменения/обновления (updated_at).

Дата удаления (deleted_at).

Дата дедлайна по заявке (deadline).

Дата исполнения заявки (solved_at).

Статус заявки (status).

Поле «deleted_at» реализует фантомное удаление заявки: в реальности не происходит удаления соответствующей записи из таблицы, – таким образом, хранится вся история заявок. Для поля «status», являющегося внешним ключом по отношению к таблице «request_status», реализован триггер для автоматического «запоминания» истории изменения статусов заявки в таблицу «request_status_history». Благодаря такой функции, срабатывающей перед обновлением таблицы «requests» происходит добавление новой строки в таблицу истории статусов, если статус в результате обновления строки поменялся (OLD.status<>NEW.status).

Таблица «requests_status_history» содержит записи изменений статусов по заявке:

Первичный ключ (id).

Внешний ключ к таблице «request» (request_id).

Дата закрытия заявки (date).

Внешний ключ к таблице «request_status» (status_from).

Внешний ключ к таблице «request_status» (status_to).

Для связи с таблицей с заявками создан ключ «request_id», а для связи со справочником статусов – ключи «status_from» и «status_to»

Статусы заявок хранятся в таблице-справочнике «requests_status», содержащей следующие поля:

Первичный ключ (id).

Название заявки (name).

Таблица-справочник «posts» создана для хранения информации о имеющихся должностях сотрудников в организации:

Первичный ключ (id).

Названия должностей (name).

Для хранения данных о структуре предприятия создана таблица-справочник «departments», содержащая поля:

Первичный ключ (id).

Название структурного подразделения (name).

Внешний ключ для «users_in_departments» (parent_id).

Для распределения пользователей по отделам предприятия используется таблица «users_in_departments», содержащая поля:

Внешний ключ для таблицы «users», являющийся и первичным (user_id).

Внешний ключ для таблицы «departments» (department_id).

Внешний ключ для таблицы «post» (user_post_id).

Таблица является связующей сразу трёх сущностей: пользователя, должности и подразделения, – в ней хранится всего три поля, каждое из которых является внешним ключом к соответствующей таблице.

Так как пользователь в рамках одного подразделения может занимать только одну должность, было принято решение использовать составной первичный ключ по полям, идентифицирующим пользователя и отдел. Таким образом важно, чтобы пара «user_id» – «department_id» была всегда уникальна.

Таблица «request_users» содержит информацию о пользователе в заявке и его роли:

Внешний ключ для таблицы «request», являющийся и первичным (request_id).

Внешний ключ для таблицы «users» (user_id).

Внешний ключ для таблицы «request_user_status» (user_status).

Структура аналогична той, что использовалась в распределении пользователей по отделам; исключением является, что в рамках одной заявки пользователь может иметь не одну роль, поэтому первичный ключ в данной таблице состоит не из двух, а из всех трёх полей.

Таблица «request_user_statuses» является справочником для связи по идентификатору и обеспечивает человеко-машинное отображение статусов:

Первичный ключ (id).

Название статуса (name).

Код статуса (code).

База данных серверной части реализована с помощью утилиты phpMyAdmin.

3. Разработка клиентской части приложения. Интерфейс клиентской части разрабатывался с использованием библиотеки Bootstrap 4 [12] – это инструментария с открытым исходным кодом для разработки с помощью HTML, CSS и JS. Позволяет

использовать переменные Sass и миксины, содержит гибкую систему сеток, множество готовых компонентов и плагинов, основанных на jQuery.

Отметим, что в рамках проекта не стояла задача создать фирменный дизайн, основная цель – реализация простейшего необходимого функционала.

Реализация осуществлялась с применением инструментариев Wamp [13].

Файловая структура клиентской части приложения показана на рисунке 2.

Папка «bootstrap» содержит CSS- и JS-файлы подключенной библиотеки. Файл «style.css» – набор стилей, разработанных для функционирования приложения (собственные стили и изменения стандартных стилей bootstrap).

Файл «jquery-3.3.1.min.js» содержит минимизированный код библиотеки jQuery, необходимой для работы bootstrap и значительно упрощающей взаимодействие с DOM.

Файл «common.js» содержит обработку событий и функции, реализуемые на клиентской части системы.

Следует отметить, что система использует множество обращений типа XMLHttpRequest – технология AJAX. Это позволяет передавать запросы на сервер и получать ответ без обновления страницы в браузере. Таким образом, например, происходит валидация (проверка) правильности заполнения форм, добавление заявки, добавление пользователя и множество другого функционала.

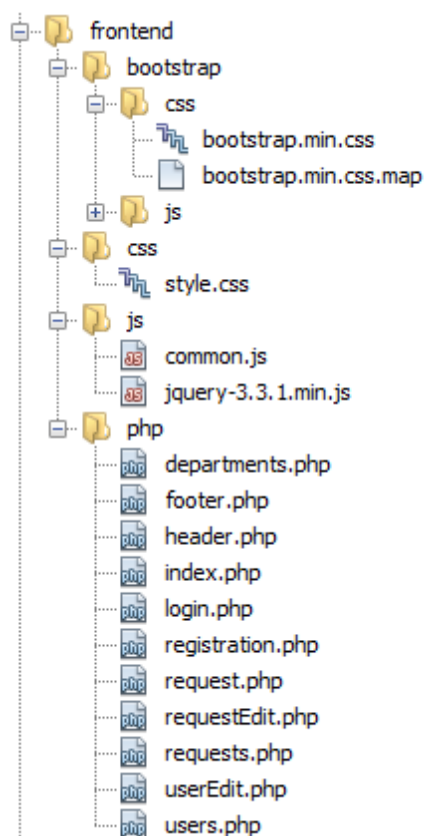


Рисунок 2 – Файловая структура клиентской части приложения

В папке «php» содержатся файлы с интерфейсами, представляющие собой шаблоны со вставками PHP-кода.

4. Описание пользовательского интерфейса приложения. Ниже приведены некоторые страницы разработанного web-приложения.

Доступ к приложению предоставляется исключительно авторизованным пользователям/администратору на странице ввода пары «Пользователь/пароль» (рис. 3).

Имя пользователя

Пароль

Войти

Рисунок 3 – Страница входа в систему

После ввода корректной пары пользователь перенаправляется на главную страницу приложения (рис.к 4), содержащую сводную информацию о заявках.

RSys Заявки Пользователи Подразделения

Создать заявку...

Вам назначено задач: **4**

Наблюдение за заявками: **1**

Ваши заявки: **7**

Все заявки: **10**

Рисунок 4 – Главная страница приложения

На каждой странице приложения (за исключением страницы авторизации) присутствует главное меню:

Заявки – страница с заявками в приложении.

Пользователи – управление пользователями (только входе в приложение в режиме администратора).

Подразделения – управление структурой организации (только входе в приложение в режиме администратора).

Кнопка «Создать заявку» перемещает пользователя на страницу создания заявки (рис. 5).

R Sys Заявки Пользователи Подразделения

Создание новой заявки

Заголовок заявки

Описание заявки

Срок исполнения

Исполнитель

Выберите исполнителя

Наблюдатели

Рисунок 5 – Страница создания заявки

После завершения формирования заявки пользователь переходит на страницу списка заявок, который изображён на рисунке 6. Здесь отображаются заявки в порядке даты последнего обновления. Цвет плашки обозначает статус заявки:

Жёлтый – «Новая».

Синий – «В работе».

Бирюзовый – «Решена».

Зелёный – «Закрыта».

Для редактирования и/или смены статуса необходимо нажать на блок с нужной задачей.

Интерфейс редактирования аналогичен интерфейсу создания заявок. Новые заявки формируются со статусом «Новая».

RSys Заявки Пользователи Подразделения

Список всех заявок

Назначены мне Наблюдаю Созданы мной Все Создать...

Обновлено: 09.06.2019 21:30 Решить до: 12.06.2019 **Убедить кого-то сделать что-то**
Создал: Администратор Администратор
Назначена: Фёдоров Иван Николаевич
Описание того, что именно надо сделать

Обновлено: 20.05.2019 21:04 **Проверить создание заявок**
Создал: Администратор Администратор
Назначена: Администратор Администратор
Надо нажать на кнопку "Создать", ввести данные (можно срок исполнения и наблюдателей не задавать) и нажать на кнопку "Сохранить"

Обновлено: 19.05.2019 12:09 Решить до: 17.05.2019 **Проба дедлайна**
Создал: Администратор Администратор
Назначена: Фёдоров Иван Николаевич

Обновлено: 11.05.2019 21:29 **Подключить кабинет 141 к Интернету**
Создал: Администратор Администратор
Назначена: Фёдоров Иван Николаевич
Проложить кабель витой пары в кабинет 141 и настроить подключение к сети Интернет

Рисунок 6 – Страница списка заявок

5. Заключение. Достоинством разработанного web-приложения является простота его использования и сопровождения сотрудниками организаций даже при отсутствии администраторов в штате. В настоящее время ведется доработка с целью взаимодействия с некоторыми ERP и CRM системами.

Список литературы:

1. INTRADESK – системы для поддержки клиентов. [Электронный ресурс]. Режим доступа: <https://www.intradesk.ru/>.
2. Еадеск – инструмент командной работы с входящими сообщениями. [Электронный ресурс]. Режим доступа: <https://yeahdesk.ru/>.
3. Directum – Интеллектуальные цифровые процессы и электронный документооборот. [Электронный ресурс]. Режим доступа: <https://www.directum.ru/solution/6754076>.
4. ITSM365 – ITIL Service Desk (Help Desk). Российское SaaS решение. [Электронный ресурс]. Режим доступа: <https://itsm365.ru/>.
5. Раевский В.А., Зуева З.И. Обоснование выбора модели данных и средств разработки интернет-ресурса бронирования железнодорожных билетов «Оригинальные исследования» (ОРИС). 2020. № 05. С. 120-125.
6. Фрейн Б. HTML 5 и CSS 3. Разработка сайтов для любых браузеров и устройств. [Текст] / Б. Фрейн. – СПб.: Питер, 2014. – 304 с.
7. Роббинс Дж. HTML 5, CSS 3 и JavaScript. Исчерпывающее руководство. [Текст] / Дж. Роббинс. – 4-е издание. – М.: Эксмо, 2014. – 528 с.
8. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML 5. [Текст] / Р. Никсон. – 4-е издание. – СПб.: Питер, 2016. – 768 с.

9. Хокинс С. Администрирование Web сервера Apache и руководство по электронной коммерции. [Текст] / С. Хокинс – М.: Издательский дом «Вильямс», 2001. – 336 с.
10. Apache HTTP Server версия 2.4 Документация. [Электронный ресурс]. Режим доступа: <https://httpd.apache.org/docs/2.4/ru/>.
11. MySQL Documentation. [Электронный ресурс]. Режим доступа: <https://dev.mysql.com/doc/>.
12. Bootstrap RUS. [Электронный ресурс]. Режим доступа: <https://getbootstrap.ru/?ysclid=lblvdu4bpj281863918>.
13. Wampserver – plate-forme de développement Web sous Windows. [Электронный ресурс]. Режим доступа: <https://www.wampserver.com/>.

References:

1. INTRADESK – sistemy dlya podderzhki klientov. [Electronic resource]. Access mode: <https://yeahdesk.ru/>. (in Russian).
2. Eadesk – instrument komandnoj raboty s vbodyashchimi soobshcheniyami. [Electronic resource]. Access mode: <https://yeahdesk.ru/>. (in Russian).
3. Directum – Intellektual'nye cifrovye processy i elektronnyj dokumentooborot. [Electronic resource]. Access mode: <https://www.directum.ru/solution/6754076>. (in Russian).
4. ITSM365 – ITIL Service Desk (Help Desk). Rossijskoe SaaS reshenie. [Electronic resource]. Access mode: <https://itsm365.ru/>. (in Russian).
5. Raevskij V.A., Zueva Z.I. Obosnovanie vybora modeli dannyh i sredstv razrabotki internet-resursa bronirovaniya zheleznodorozhnyh biletov «Original'nye issledovaniya» (ORIS). (ОРИС). 2020. № 05. P. 120-125. (in Russian).
6. Frejn B. HTML 5 i CSS 3. Razrabotka sajtov dlya lyubyh brauzerov i ustrojstv. [Text] / B. Frejn. – Saint Petersburg: Piter, 2014. – 304 p. (in Russian).
7. Robbins Dzh. HTML 5, CSS 3 i JavaScript. Ischerypyvayushchee rukovodstvo. [Text] / Dzh. Robbins. – 4-e izdanie. – Moscow: Eksmo, 2014. – 528 p. (in Russian).
8. Nikson R. Sozdaem dinamicheskie veb-sajty s pomoshch'yu PHP, MySQL, JavaScript, CSS i HTML 5. [Text] / R. Nikson. – 4-e izdanie. – Saint Petersburg: Piter, 2016. – 768 p. (in Russian).
9. Hokins S. Administrirovanie Web servera Apache i rukovodstvo po elektronnoj kommercii. [Text] / S. Hokins – Moscow: Izdatel'skij dom «Vil'yams», 2001. – 336 p. (in Russian).
10. Apache HTTP Server versiya 2.4 Dokumentaciya. [Electronic resource]. Access mode: <https://httpd.apache.org/docs/2.4/ru/>.
11. MySQL Documentation. [Electronic resource]. Access mode: <https://dev.mysql.com/doc/>. (in Russian).
12. Bootstrap RUS. [Electronic resource]. Access mode: <https://getbootstrap.ru/?ysclid=lblvdu4bpj281863918>.
13. Wampserver – plate-forme de développement Web sous Windows. [Electronic resource]. Access mode: <https://www.wampserver.com/>. (in Belgian).