МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Факультет №3 – «Системы управления, информатика и электроэнергетика»

Кафедра 308 - «Информационных технологий»

**ОТЧЕТ**

**О ПРОДЕЛАННОЙ ЛАБОРАТОРНОЙ РАБОТЕ №3**

**«Создание веб-приложения: телефонный справочник»**

Выполнил студент гр. M30-321:

Комашинский Т.В \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверила: Трутнева Н.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата, подпись

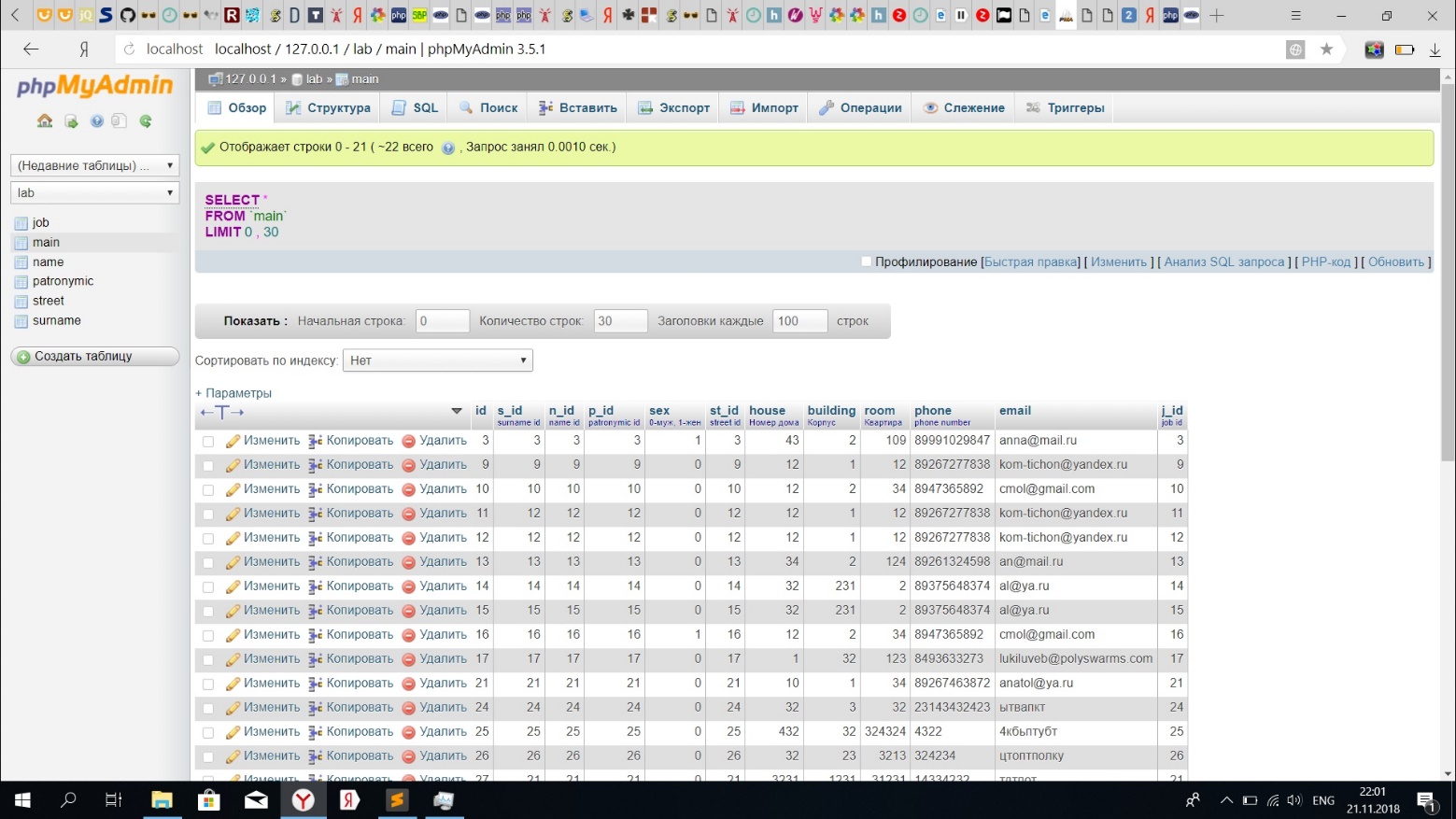
Москва, 2018 г.

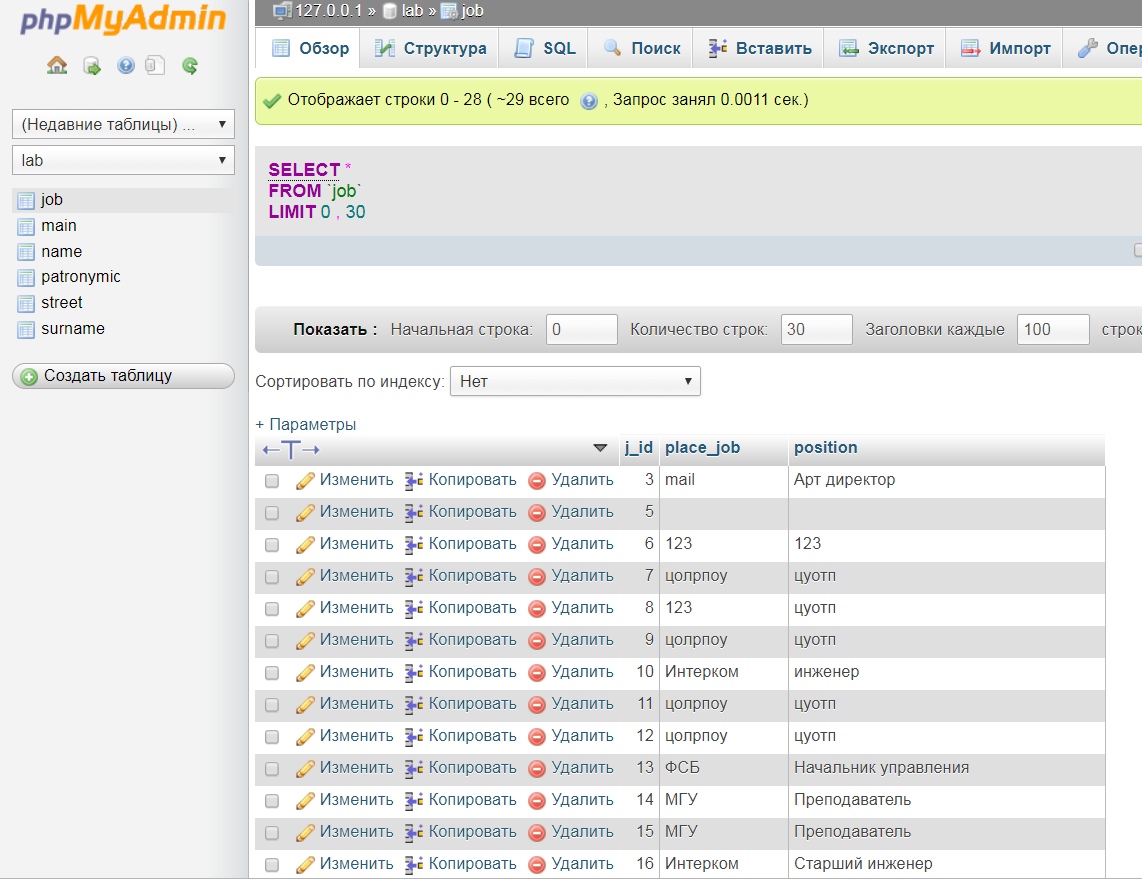
**Цель работы:** Создание приложения в архитектуре “клиент-сервер”, реализующего работу телефонного справочника.

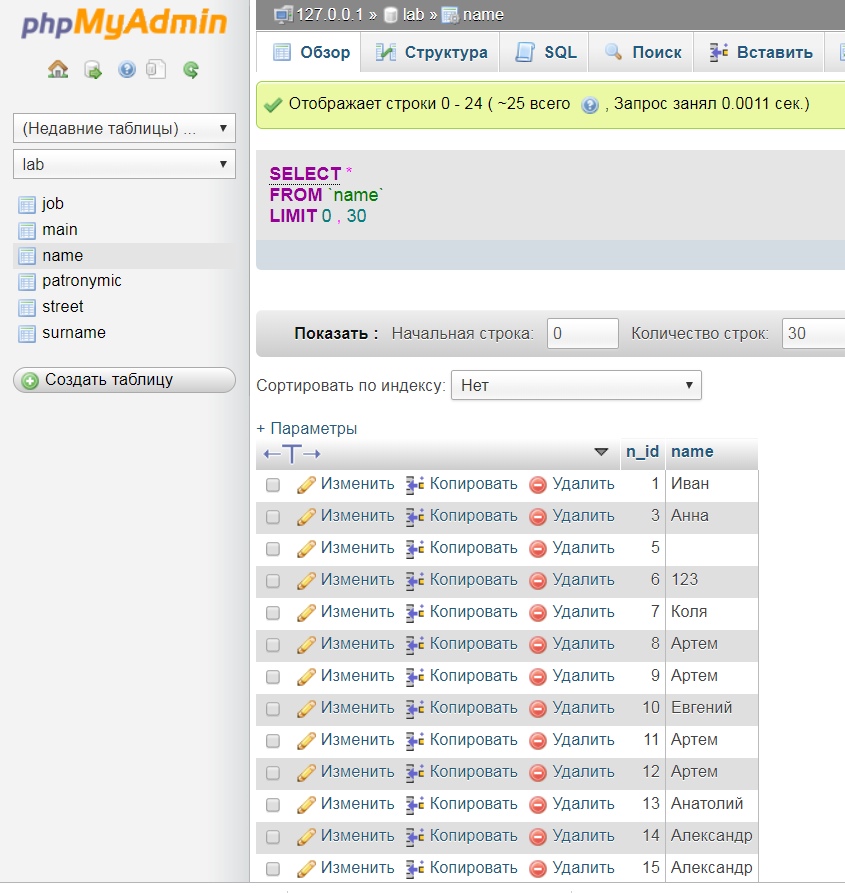
**Порядок проведения работы:**

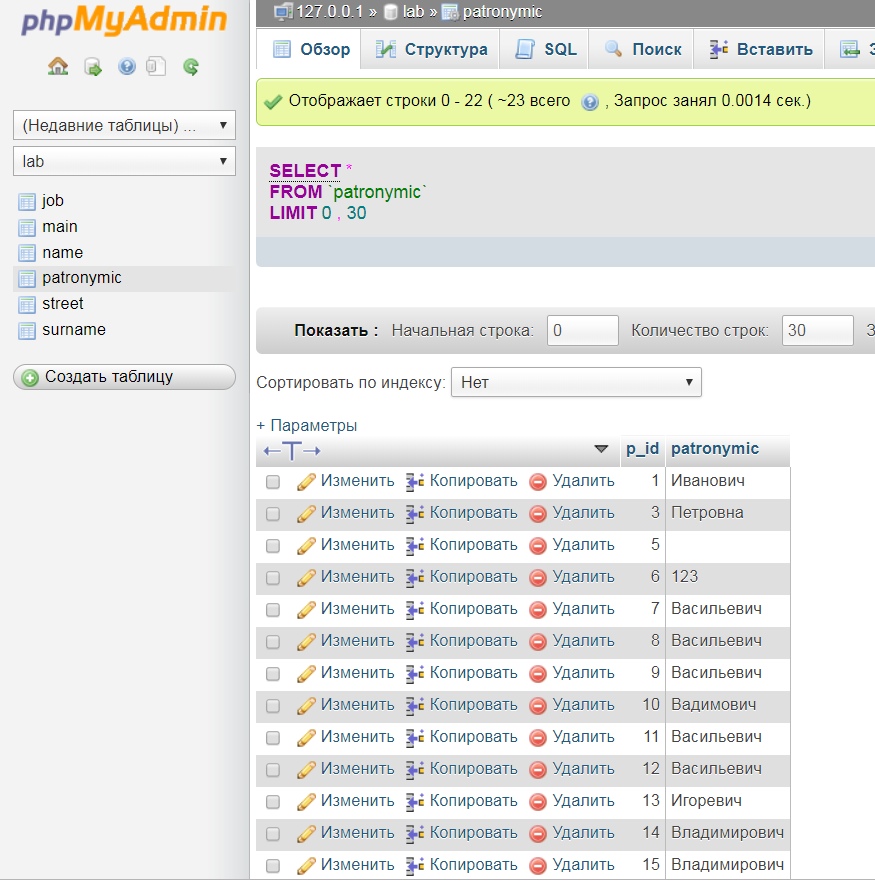
* **Создание БД посредством PhpMyAdmin**

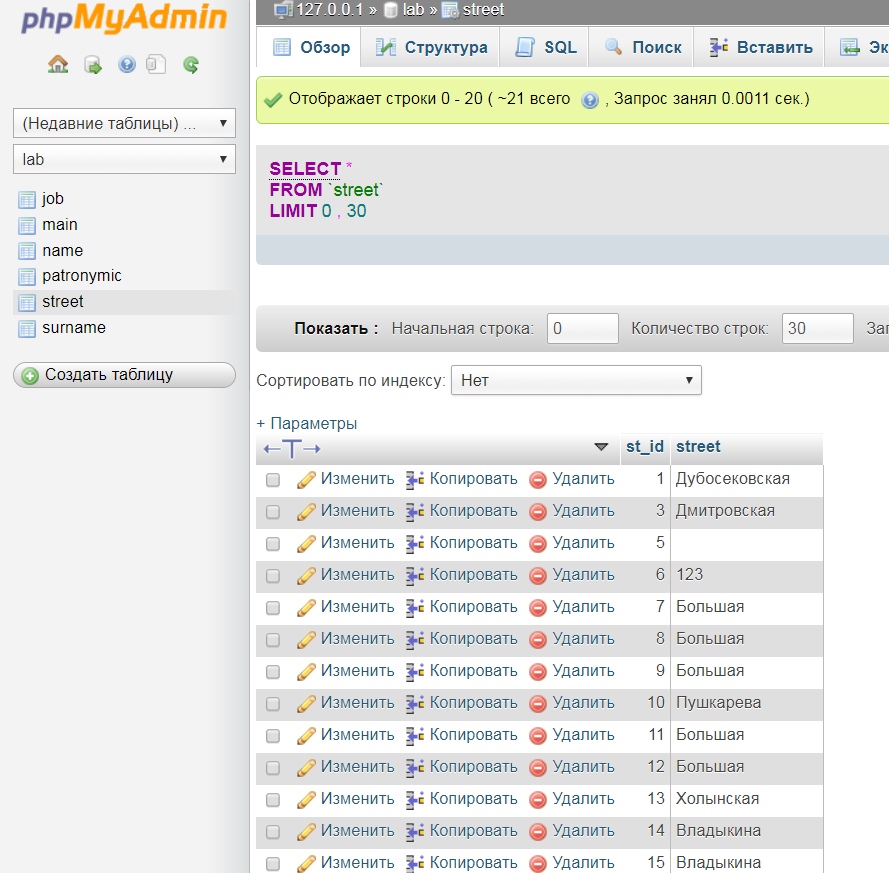
1. Скачиваем Denver
2. Заходим в PhpMyAdmin
3. Создаем БД используя интерфейс PhpMyAdmin согласно ТЗ

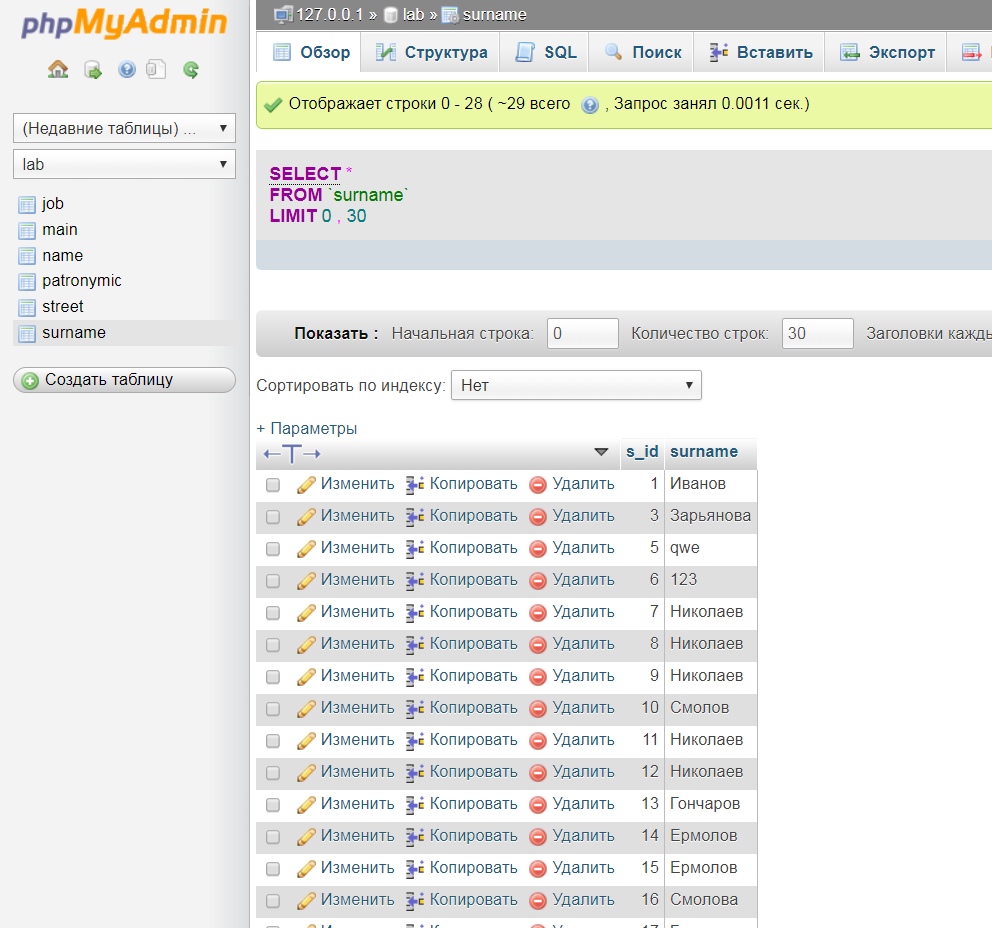






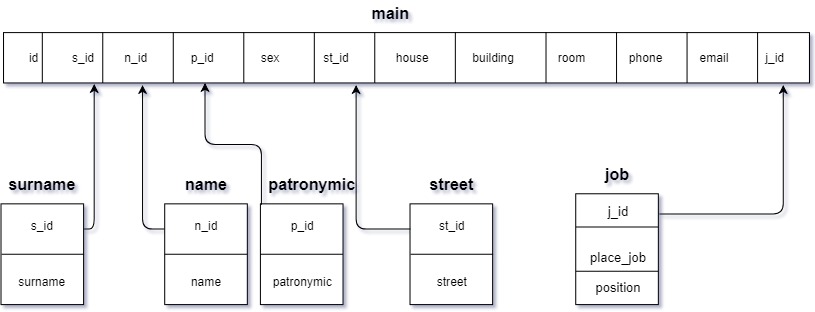






В БД в таблице main в полях s\_id, n\_id, p\_id, st\_id содержаться ссылки на таблицы surname, name, patronomic, street, в которых находятся значения фамилии, имени, отчества и улицы.

**Схема таблиц в БД**

****

**Общая схема взаимодействия файлов**

js

www

script.js

script.add.phpphp

script.panel.php

bootstrap.js

index.php

script.php

Jquery-3.3.1.js

model.php

css

bootstrap-theme.css

bootstrap.css

font-awesome.css

style.css

*Папка www:*

* Файл **script.php** отвечает за поиск контакта посредством SQL запроса SELECT. В БД в таблице main в полях s\_id, n\_id, p\_id, st\_id содержаться id ссылающиеся на таблицы surname, name, patronomic, street, в которых находятся значения фамилии, имени, отчества и улицы. В SQL запросе мы посредством INNER JOIN объединяем эти таблицы. В итоге выбираются из таблицы те строки, в которых находится не строгое соответствие с переменной $word, значение которого мы прописываем через input в файле index.php. Найденное соответствие выделяется синим цветом.
* Файл **script.panel.php** отвечает за работу панели просмотра (открывается при нажатии на любую строку в таблице), в которой можно изменить выбранный элемент или удалить его. Изменение осуществляется с помощью SQL запроса UPDATE, а удаление с помощью SQL запроса DELETE.
* Файл **model.php** отвечает за соединение с БД.
* Файл **Index.php** представляет из себя основу клиентской части всего проекта. В нем происходит отправка форм в файлы серверной части и вывод ответа из нее . Опишем его поподробнее:

Всё приложение условно можно разделить на 4 блока.

1. List\_contacts - вывод на страницу списка всех контактов из БД.
2. Search – отвечает за поиск контакта по введенному в input слову.
3. Add - этот блок представляет из себя форму из input’ов, значения которых передаются по нажатию кнопки в файл script.add.php.
4. Panel - блок отвечает за вывод панели просмотра индивидуальной для каждого контакта с возможностью редактирования и удаления выбранного контакта.

* Файл **Script.add.php** представляет из себя элемент добавления в БД данных ранее полученных через форму из файла Index.php посредством sql запроса INSERT. А также проверяет успешность выполнения операции и уведомляет о результате выводом HTML блока с картинкой “Успешно” или “Ошибка”.

*Папка js:*

* Файл **script.js** содержит весь функционал клиентской части, обработку событий при нажатии на кнопки.
* Файл **bootstrap.js** отвечает за переключение между вкладками, за показ панели просмотра.
* Файл **Jquery-3.3.1.js** – библиотека, которая упрощает получение доступа к элементам.

*Папка css* отвечает за оформление сайта.

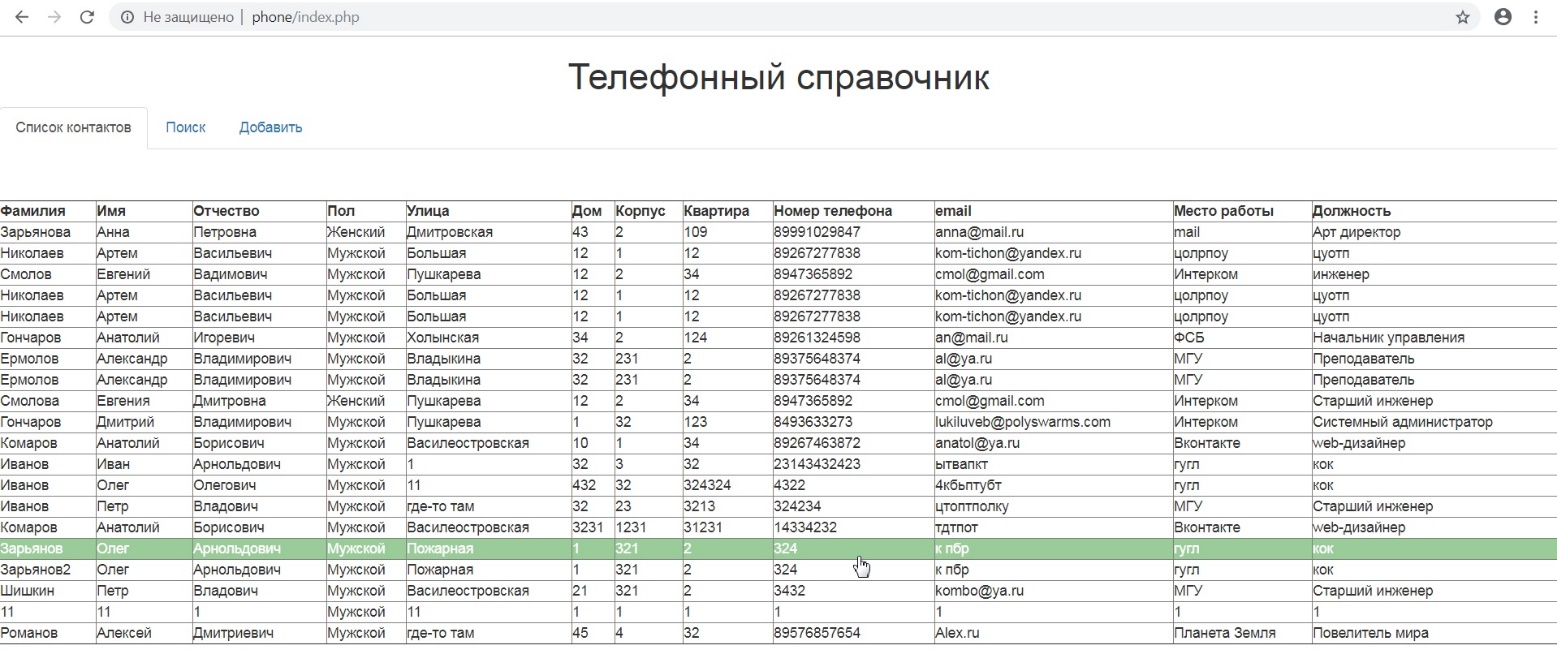
**Описание работы основных функций приложения:**

1. **Поиск.** В файле index.php находится текстовое поле, куда пользователь вводит искомое слово. При нажатии на клавишу клавиатуры в заданном диапазоне (прописывается в файле script.js) происходит событие onkeyup(отпускании клавиши), при котором запускается функция search(e). Эта функция прописана в файле script.js, она отсылает полученное слово на сервер ajax-запросом методом POST. В этом файле сначала подключается файл model.php, вызывается функция startup(), которая произведет соединение с БД. Далее с помощью функции isset() проверяем на существование принятый post-запрос. Переменная $q содержит результат выполнения SQL-запроса SELECT, который посредством INNER JOIN объединяет таблицу main и таблицы job, name, surname, patronymic, street, при условии, что id из таблиц job, name, surname, patronymic приравнивается к соответствующим id из таблицы main . После чего пользователю выводятся из таблицы те строки, в которых находится не строгое соответствие с переменной $word. Найденное соответствие выделяется синим цветом.
2. **Добавление.** В файле Index.php задается блок, состоящий из текстовых полей(данные о пользователе). По нажатию кнопки “добавить” со стороны клиента считываются данные полей и ajax-запросом отправляются на сервер в файл script.add.php. Далее в этом файле сначала подключается файл model.php и вызывается функция startup(), которая произведет соединение с БД. Потом создаются переменные, которые примут в себя значения переданных данных. И начнется проверка на уникальность этих данных, в ходе которой, при условии, что в БД нет такого же значения, данное значение будет внесено в БД и ему присвоится уникальный ID, если же найдется такое же значение, то ID возьмут у него. С помощью этого алгоритма мы получаем все записи, которые будут впоследствии добавлены в соответствующие поля БД. После того как определены значения для таблицы main, происходит внесение нового контакта в БД, посредством sql-запроса INSERT и отобразится HTML страница, сообщающая об успешном добавлении.
3. **Редактирование/Удаление.** В файле Index.phpесть блок list-contacts, который реализует вывод всех контактов из БД, с помощью sql-запроса select. При нажатии на любую строку появившейся таблицы, происходит передача ID этой строки(записи) в функцию view\_panel(id), которая в свою очередь отправляет на сервер script.panel.php переданную ему переменную Id, спец-слово action, открывая при этом панель просмотра, состоящую из формы, которая повторяет все поля контакта, и две функциональные кнопки “удалить” и “сохранить изменения”. В этом файле сначала подключается файл model.php и вызывается функция startup(), которая произведет соединение с БД. Далее производится проверка на существование переданного ранее post запроса из функции view\_panel(id) и равенства записи action необходимому значению. После прохождения проверки, с помощью sql-запроса select и переданного значения id, в БД находим значения полей контакта, который соответствует записи id, и выводим все поля этого контакта в ранее упомянутые формы, давая возможность изменять их для реализации в дальнейшем функции редактирования в приложении. Далее само редактирование и удаление происходит по нажатию на соответствующую кнопку.

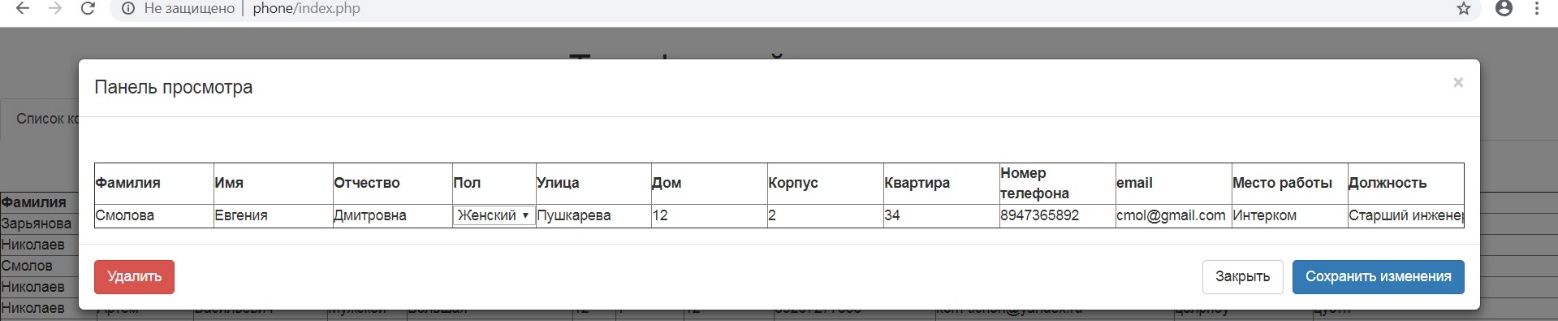
* **Редактирование** происходит по нажатию на кнопку сохранить изменения. Нажатие на кнопку запускает функцию save\_change(id), которая принимает тот самый id, который мы получили еще при нажатии на строку в блоке list\_contacts. Функция берет значения из форм на панели просмотра и формирует из них post запрос, внося туда так же ID и спец-слово action, который будет передан в script.panel.php. В файле script.panel.php, пройдя такую же проверку, как и до этого, на существование переданного id и соответствия спец-слова заданному значению, начинается алгоритм редактирования. Этот алгоритм заключается в том, чтобы взять переданные ему записи посредствам post запросов и применить sql-запрос UPDATE на каждую необходимую запись в таблицах, где встретилось соответствие c переданным id.
* **Удаление** происходит по нажатию на кнопку “Удалить” Нажатие на кнопку запускает функцию delete\_record(id). Функция формирует post запрос, внося туда ID и спец-слово action, который будет передан в script.panel.php. В файле script.panel.php, пройдя такую же проверку, как и до этого, на существование переданного id и соответствия спец-слова заданному значению, осуществляется удаление: имея переданный нам id, применяем sql-запрос DELETE на каждую запись в БД, где обнаружилось соответствие с id.

**Результаты работы**

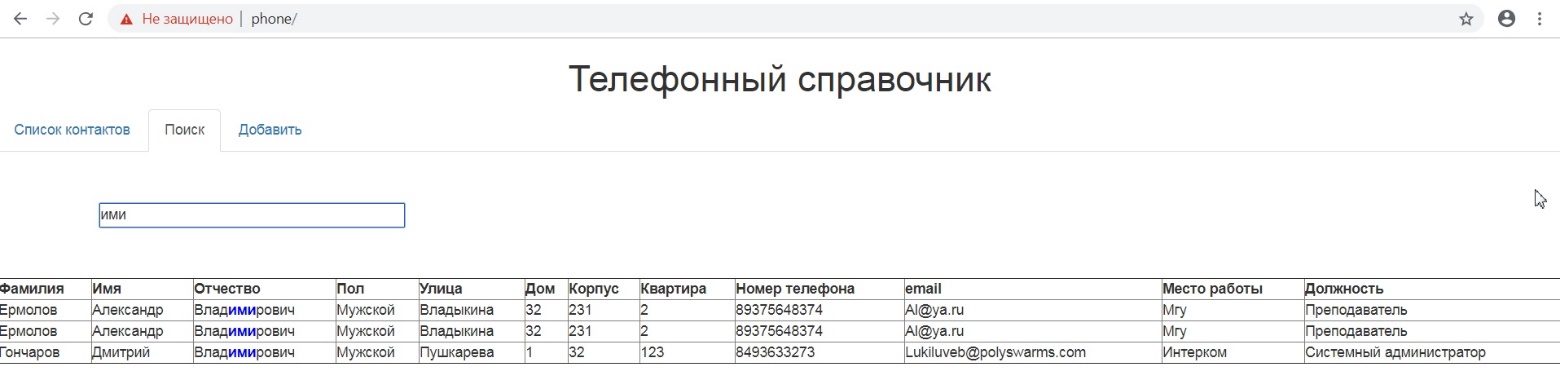
После выполнения данной работы мы получаем полностью функционирующее web-приложение, которое представляет собой телефонный справочник с возможностью поиска, добавления, редактирования и удаления контактов.

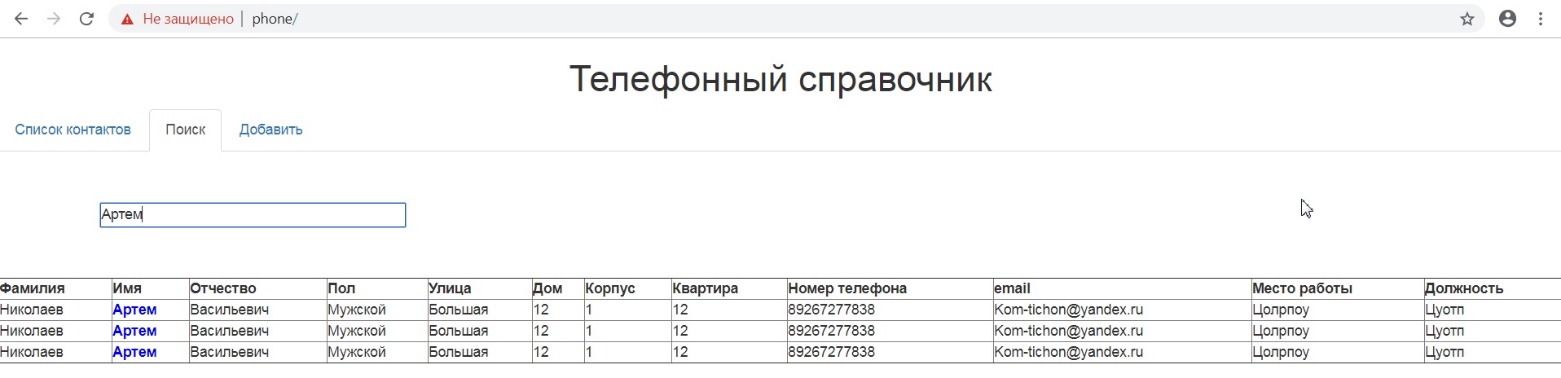
Выбираем элемент в таблице

Затем появляется «Панель просмотра», в которой мы можем изменить данный элемент таблицы (или какую-то его часть) или удалить его.

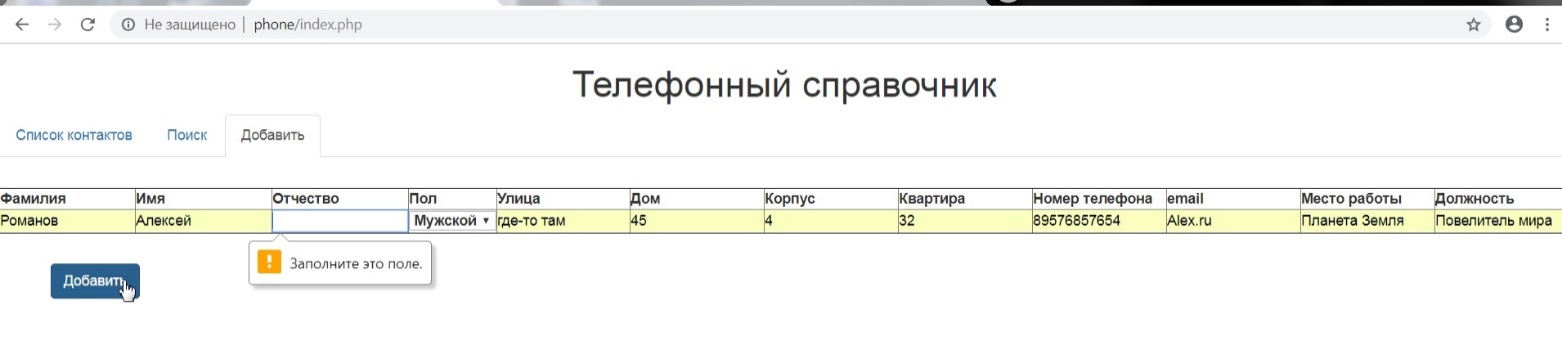
****

Так же имеется возможность поиска нужного контакта:

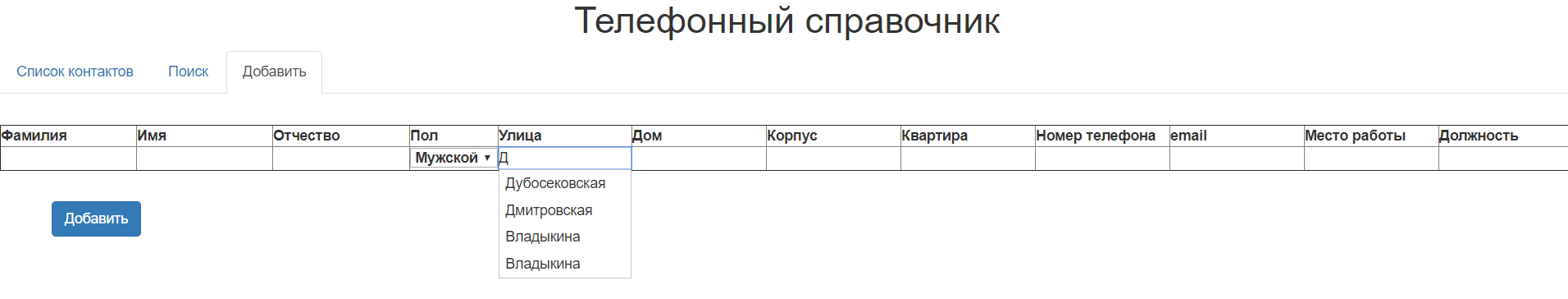




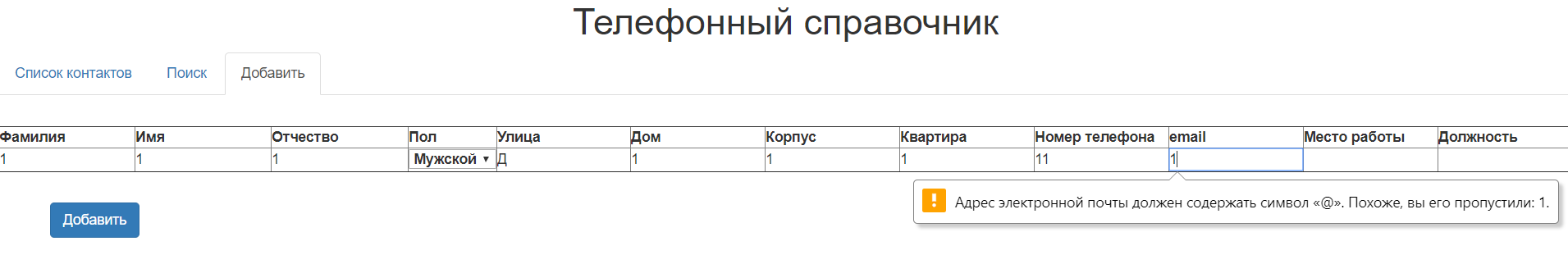
Имеется функция добавления нового контакта. Если пользователь пропустил какое-либо поле для заполнения, то система укажет ему на это.



Добавлена функция автозаполнения в таких полях, как улица, место работы и должность.



Также реализована «защита от дурака» в т.ч. для поля “email”



**Вывод:** Получен полезный опыт разработки Web-приложения, работы с CУБД PHPMyAdmin

и взаимодействием приложения с БД посредством SQL-запросов.

Получен опыт работы с языками программирования PHP , JavaScript , библиотеками Bootstrap, JQuery.