

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий
институт

Вычислительная техника
кафедра

УТВЕРЖДАЮ
Заведующий кафедрой
А. И. Легалов
подпись инициалы, фамилия
«___» _____ 2016 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 Информатика и вычислительная техника

код и наименование специализации

Веб-система восстановления хронологии событий. Разработка
пользовательского интерфейса
тема

Пояснительная записка

Руководитель

подпись, дата

ст.преп.

должность, ученая степень

Л. В. Макуха

инициалы, фамилия

Выпускник

подпись, дата

В. П. Потапов

инициалы, фамилия

Нормоконтролер

подпись, дата

доцент, к.т.н

должность, ученая степень

В. И. Иванов

инициалы, фамилия

Красноярск 2016

Федеральное государственное автономное
образовательное учреждение
высшего профессионального образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий
институт

Вычислительная техника
кафедра

УТВЕРЖДАЮ
Заведующий кафедрой
Легалов А.И.
подпись фамилия, инициалы
«___» ____ 2016 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврская работа
бакалаврской работы, дипломного проекта, дипломной работы, магистерской диссертации

Студенту Потапов Вадим Павлович
фамилия, имя, отчество

Группа КИ12-09Б Направление (специальность) 09.03.01
номер код

Информатика и вычислительная техника

наименование

Тема выпускной квалификационной работы

Веб-система восстановления хронологии событий. Разработка
пользовательского интерфейса

Утверждена приказом по университету № _____ от _____

Руководитель ВКР Макуха Л.В., старший преподаватель кафедры ВТ
фамилия, инициалы, должность, учёное звание и место работы

Исходные данные для ВКР задание на проектирование

Перечень разделов ВКР анализ задания на проектирование, разработка архитектуры и основных технических решений

Перечень графического материала демонстрационное видео, презентация в формате Power Point

Руководитель ВКР

подпись

Макуха Л.В.

фамилия, инициалы

Задание принял к исполнению

подпись

Потапов В.П.

фамилия, инициалы

«____» _____ 2016 г.

РЕФЕРАТ

Пояснительная записка содержит 41 страницы, 18 рисунков, 7 источников.

Предметом ВКР является разработка и создание веб-приложение.

При создании сайта были рассмотрены языки веб программирования (PHP, HTML, JavaScript, CSS).

Результатом данной работы является разработка веб-приложения, осуществляющего функцию визуализации информации о преступлении.

Итогом будет являться веб-приложение, которое создавалось с целью упростить работу следователям, благодаря визуальному отображению общей картины с местом преступления.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 Анализ задания на проектирование.....	5
1.1 Выбор картографического сервиса	5
1.1.1 Картографический сервис Google Maps	5
1.1.2 Картографический сервис Яндекс.Карты.....	6
1.1.3 Картографический сервис 2ГИС	6
1.1.4 Обоснования выбора картографического сервиса	7
1.2 Выбор средств разработки	7
1.2.1 Язык программирования PHP	7
1.2.2 Система управления базами данных MySQL	8
1.2.3 Язык программирования JavaScript	9
1.2.4 Обоснование выбора средств разработки	10
1.3 Техническое задание.....	11
1.3.1 Назначение и цели создания веб-приложения.....	11
1.3.2 Требование к веб-приложению	11
1.3.3 Требования к видам обеспечения.....	13
1.4 Вывод по разделу	13
2 Разработка архитектуры и основных технических решений.....	14
2.1 Функциональная модель веб-приложения	14
2.2 Информационная структура веб-приложения	14
2.2.1 Диаграмма прецедентов	14
2.3 Разработка функции авторизации	15
2.3.1 Структура работы с авторизацией	15
2.3.2 Программная реализация функции авторизации.....	16
2.3.3 Результат работы функции авторизации	17
2.4 Разработка функции отображения карты с метками.....	19
2.4.1 Структура отображения карты и меток	19
2.4.2 Программная реализация функции отображения карты и меток	20
2.4.3 Результат работы функции отображения карты и меток.....	23
2.5 Разработка функции администрирования	25
2.5.1 Структура функции администрирования	25

2.5.2 Программная реализация функции администрирования.....	25
2.5.3 Результат работы функции администрирования	32
2.6 Разработка функции выбора дела.....	36
2.6.1 Структура функции выбора дела	36
2.6.2 Программная реализация функции выбора дела.....	36
2.6.3 Результат работы функции выбора дела	39
ЗАКЛЮЧЕНИЕ	40
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	41

ВВЕДЕНИЕ

В наше время информационные технологии развиваются все с большей и большей скоростью. Они имеют свое применение во многих сферах, будь то медицина или образование, но хотелось бы сфокусировать внимание именно на раскрытии преступлений. Уровень их раскрытия, конечно, вырос за предыдущие годы, но все равно большое количество преступлений остается нераскрытыми.

Поэтому система, которая рассматривается в данной работе, будет более чем актуальна. Любой следователь был бы рад системе, которая отображает информацию о местоположении людей в момент совершения преступления, что позволит сэкономить время следователям, так как им будет удобнее расследовать дело с полностью отстроенной картиной произошедшего.

Предмет ВКР заключается в разработке веб-приложения для отображения информации об обстановке на месте преступления на картах Google Maps. Конечная цель данной выпускной работы – разработать эффективное приложение, с помощью которого можно будет считывать данные с трекеров машин и отобразить их на карте, а также визуализировать местоположение людей в момент совершения преступления.

В связи с указанной целью возникают следующие задачи:

- изучить материалы необходимые для разработки веб-приложения;
- научиться работать с картографическим сервисом;
- реализовать функцию добавления маркеров на карту.

Создание такого приложения довольно сильно сэкономит время следователям, так как им будет удобнее расследовать дело с полностью отстроенной картиной произошедшего.

1 Анализ задания на проектирование

Необходимо разработать веб-приложение для визуализации информации об обстановке в момент совершения преступления:

- отображение карты в веб-приложении;
- авторизация;
- импорт gps-трекеров машин на карту;
- добавления маркеров участников на карту;
- “отрисовка” путей маркеров на карте;
- выбор преступления;
- запись координат и информации маркеров в базу данных;
- чтение информации о маркерах с базы данных.

1.1 Выбор картографического сервиса

Существует большое количество картографических сервисов, но только некоторые из них подходят для нашей работы:

- Google Maps;
- Яндекс.Карты;
- 2ГИС.

1.1.1 Картографический сервис Google Maps

Карты Google — набор приложений, построенных на основе бесплатного картографического сервиса и технологии, предоставляемых компанией Google. Созданы в 2005 году.

Достоинства:

- лучшее покрытие всего мира;
- обзорные фотографии улиц;
- интеллектуальный поиск;

- голосовой ввод.

Недостатки:

- большой размер данных при использовании оффлайн;
- нет возможности общения между пользователями [1].

1.1.2 Картографический сервис Яндекс.Карты

Яндекс.Карты — поисково-информационная картографическая служба Яндекса. Открыт в 2004 году.

Достоинства:

- лучшее покрытие России;
- хорошая детализация России;
- отображение пробок в крупных городах.

Недостатки:

- нет ночного режима;
- нет детализации на уровне зданий;
- большой размер данных при использовании оффлайн [1].

1.1.3 Картографический сервис 2ГИС

2ГИС — международная картографическая компания, выпускающая одноимённые электронные справочники с картами городов с 1999 года.

Достоинства:

- одна из лучших детализаций в городах присутствия;
- возможность загрузки и использования оффлайн;
- подробная информация об организациях.

Недостатки:

- уступает конкурентам в покрытии как в России, так и в других странах;
- нет возможности общения между пользователями;

- нет обзорных фотографий улиц;
- нет ночного режима [1].

1.1.4 Обоснования выбора картографического сервиса

Для выполнения данной работы был выбран картографический сервис Google Maps в связи с удобным инструментарием для разработки веб-приложения.

1.2 Выбор средств разработки

На сегодняшний день существует множество языков программирования. Каждый из языков удобен для той или иной ситуации, однако для написания веб-приложения в основном используются следующие языки:

- PHP;
- MySQL;
- JavaScript.

1.2.1 Язык программирования PHP

PHP — является распространенным интерпретируемым языком общего назначения с открытым исходным кодом. PHP создавался специально для ведения web-разработок и код на нем может внедряться непосредственно в HTML-код. Синтаксис языка берет начало из C, Java и Perl, и является легким для изучения. Основной целью PHP является предоставление web-разработчикам возможности быстрого создания динамически генерируемых web-страниц, однако область применения PHP не ограничивается только этим [2].

Синтаксис PHP подобен синтаксису языка Си.

Для работы программы не требуется описывать какие-либо переменные, используемые модули и т. п. Любая программа может начинаться непосредственно с оператора PHP.

Достоинства:

- подключается к любым БД;
- свободный язык программирования и не требует лицензий на разработки на нем;
- можно запускать из консоли как обычно так и через планировщик cron что позволяет управлять рассылками и другими событиями.

Недостатки:

- процедурный стиль работы со строками/массивами;
- порядок параметров;
- скорость выполнения [3].

1.2.2 Система управления базами данных MySQL

MySQL — свободная реляционная система управления базами данных.

Разработку и поддержку MySQL осуществляет корпорация Oracle, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQL AB. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP, VertrigoServ. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако

в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц [4].

Достоинства:

- обеспечивается поддержкой большого количества типов таблиц;
- СУБД MySQL поставляется со специальным типом таблиц EXAMPLE демонстрирующим принципы создания новых типов таблиц;
- благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

Недостатки:

- наиболее быстрый тип таблиц MyISAM имеет невысокую надежность;
- таблицы типа MyISAM не поддерживают транзакции [4].

1.2.3 Язык программирования JavaScript

JavaScript — прототипно-ориентированный сценарный язык программирования.

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java, но при этом лёгким для использования непрограммистами. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке [5].

Достоинства:

- объекты, с возможностью интроспекции;
- функции как объекты первого класса;
- автоматическое приведение типов;
- автоматическая сборка мусора;
- анонимные функции.

Недостатки:

- модульная система: JavaScript не предоставляет возможности управлять зависимостями и изоляцией областей видимости;
- стандартная библиотека: в частности, отсутствует интерфейс программирования приложений по работе с файловой системой, управлению потоками ввода-вывода, базовых типов для бинарных данных;
- стандартные интерфейсы к веб-серверам и базам данных;
- система управления пакетами, которая бы отслеживала зависимости и автоматически устанавливала их [6].

1.2.4 Обоснование выбора средств разработки

Для выполнения данной работы был выбран PHP вместе с базой данных MySQL, так как они хорошо взаимодействуют друг с другом и довольно просты в написании веб-приложения, а JavaScript был выбран для

удобной работы с Google Maps. Их минусы не являются критичными для данной работы.

1.3 Техническое задание

1.3.1 Назначение и цели создания веб-приложения

1.3.1.1 Назначение веб-приложения

Веб-приложение — это вспомогательные программные средства, которые предназначены для автоматизированного выполнения каких-либо действий.

1.3.1.2 Цель создания веб-приложения

Основной целью создания веб-приложения является визуализация информации о преступлениях на картах Google Maps. Это позволит следователям проще и эффективнее изучить информацию по делу.

1.3.2 Требование к веб-приложению

1.3.2.1 Требования к структуре и функциональной части веб-приложения

Функциональная структура веб-приложения должна включать в себя:

- запускаться в любом браузере;
- дружелюбный интерфейс;
- малую ресурсоемкость;
- быстроту работы;
- загрузка информации из БД;
- простое и понятное отображение информации на карте.

1.3.2.2 Перспективы развития веб-приложение

Добавление новых функций в веб-систему должно проходить с минимальными добавлениями в основной код.

1.3.2.3 Требования к надежности

Показатель надежности

Время восстановления работоспособности веб-приложения при любых сбоях и отказах не должно превышать одного рабочего дня, исключая случаи неисправности серверного оборудования.

1.3.2.4 Требования по эргономике и технической эстетике

Требования к внешнему оформлению

При разработке визуальной части программы должны учитываться психофизиологические особенности человека:

- не слишком яркая цветовая палитра;
- визуальная часть веб-приложения должно сочетаться с визуальной частью сайта;
- информация должно выводится простым и понятным способом;
- положение визуальной составляющей веб-приложения должно соответствовать структуре сайта.

1.3.3 Требования к видам обеспечения

1.3.3.1 Требования к лингвистическому обеспечению

Языки программирования

Для удобной работы с Google Maps были использованы следующие языки программирования:

- PHP;
- MySQL;
- JavaScript.

Языки взаимодействия пользователей и системы

Основным языком взаимодействия пользователей и системы является русский язык.

1.3.3.2 Требования к программному обеспечению

Программное обеспечение должно быть представлено в виде веб-приложения.

1.4 Вывод по разделу

В результате анализа задания на дипломное проектирование были сформулированы четкие требования к разрабатываемой системе, было принято важное техническое решение - разрабатываемая система будет получать информацию из БД для дальнейшего занесения на карту Google Maps.

2 Разработка архитектуры и основных технических решений

2.1 Функциональная модель веб-приложения

Цель работы - является разработка веб-приложения для автоматизации процесса поиска свидетелей. С помощью данного веб-приложения следователи смогут визуализировать местоположение людей в момент совершения преступления, что позволит сэкономить время следователям, так как им будет удобнее расследовать дело с полностью отстроенной картиной произошедшего.

Перед тем как работать с веб-приложением будет предложено авторизоваться. Если ввести данные администратора, будет доступно добавление и удаление дел и пользователей. При авторизации обычного пользователя будет можно выбрать дела для дальнейшей работы с веб-приложением.

2.2 Информационная структура веб-приложения

Для начала необходимо составить информационную модель.

2.2.1 Диаграмма прецедентов

В разрабатываемом веб-приложении выделены 2 роли пользователь (например, следователь) и администратор. Диаграмма прецедентов представлена на рисунке 1.

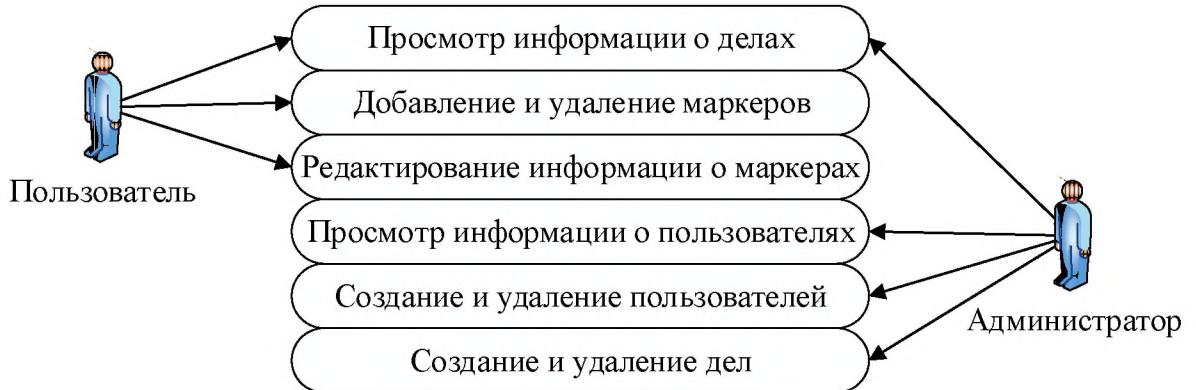


Рисунок 1 - Диаграмма прецедентов

Функции пользователя:

- просмотр информации о делах;
- добавление и удаление маркеров;
- редактирование информации о маркерах.

Функции администратора:

- просмотр информации о пользователях;
- создание и удаление пользователей;
- просмотр информации о делах;
- создание и удаление дел.

2.3 Разработка функции авторизации

2.3.1 Структура работы с авторизацией

Одной из важных функций является функция авторизации пользователя. При запуске веб-приложения будет предложено авторизоваться. При вводе данных администратора будут доступны функции управления делами и пользователями. После ввода данных обычного пользователя будет предложен выбор дела для дальнейшей работы с картой (рисунок 2). Если данные были введены неверно, веб-приложение выдаст сообщение об ошибке.

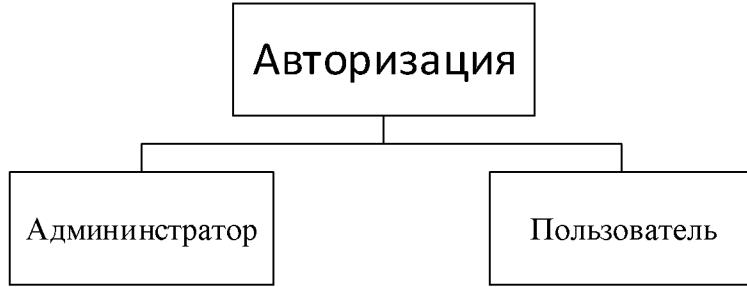


Рисунок 2 - Структура функции авторизации

2.3.2 Программная реализация функции авторизации

Проанализировав структуру функции регистрации, был разработан файл «login.php». Рассмотрим содержимое данного файла.

Для начала инициализируем данные сессии.

```

<?php session_start();?>
<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
    <title>Вход на сайт</title>

```

Подгружаем стили дизайна веб-приложения.

```

<link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-
awesome/3.2.0/css/font-awesome.min.css'>
<link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
</head>
<body>

```

Создаем окно с авторизацией.

```

<div class="box">
<div class="containerWrapper2">
    <div id="containerLogin" class="tabContainer2">
        <form action="login.php" method="POST">
            <h2 class="loginTitle">Авторизация</h2>
            <div class="loginContent">
                <div class="inputWrapper">

```

```
<input type="text" name="login" placeholder="Логин" />
</div>
<div class="inputWrapper">
    <input type="password" name="password" placeholder="Пароль"/>
</div>
</div>
<button class="blueBox" name="submit"><span class="iconLogin"></span>
ВОЙТИ</button>
<div class="clear"></div>
</form>
```

2.3.3 Результат работы функции авторизации

При авторизации под логином обычного пользователя (рисунок 3) появится окно с выбором доступных дел, как это видно на рисунке 4.

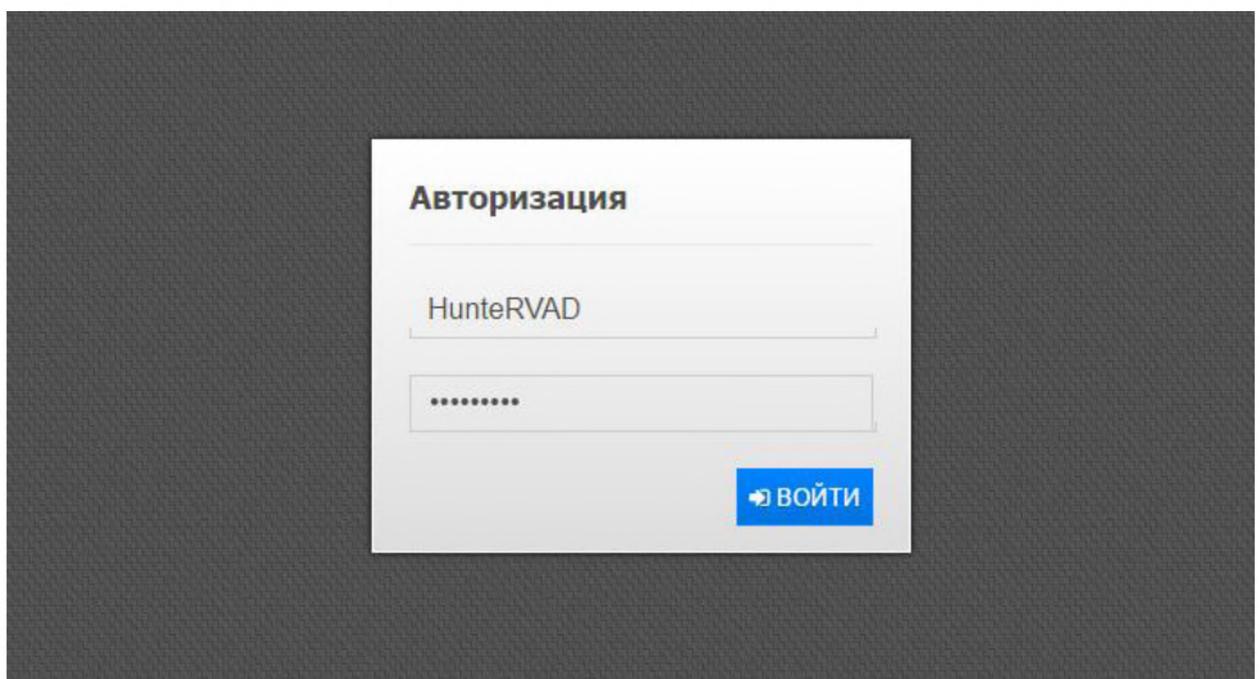


Рисунок 3 - Авторизация пользователя

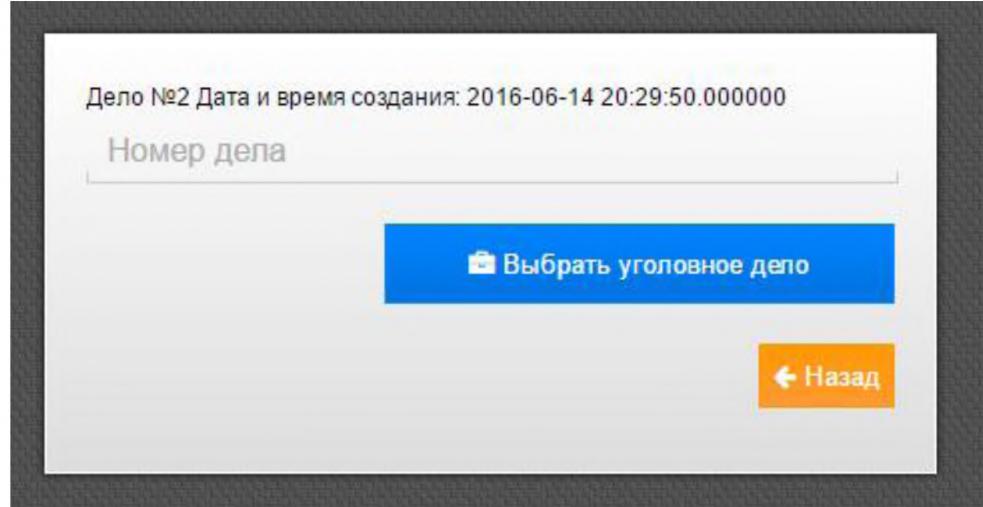


Рисунок 4 - Выбор дела

При авторизации в роли администратора (рисунок 5) будет показано окно с управлением пользователями и делами. Результат представлен на рисунке 6.

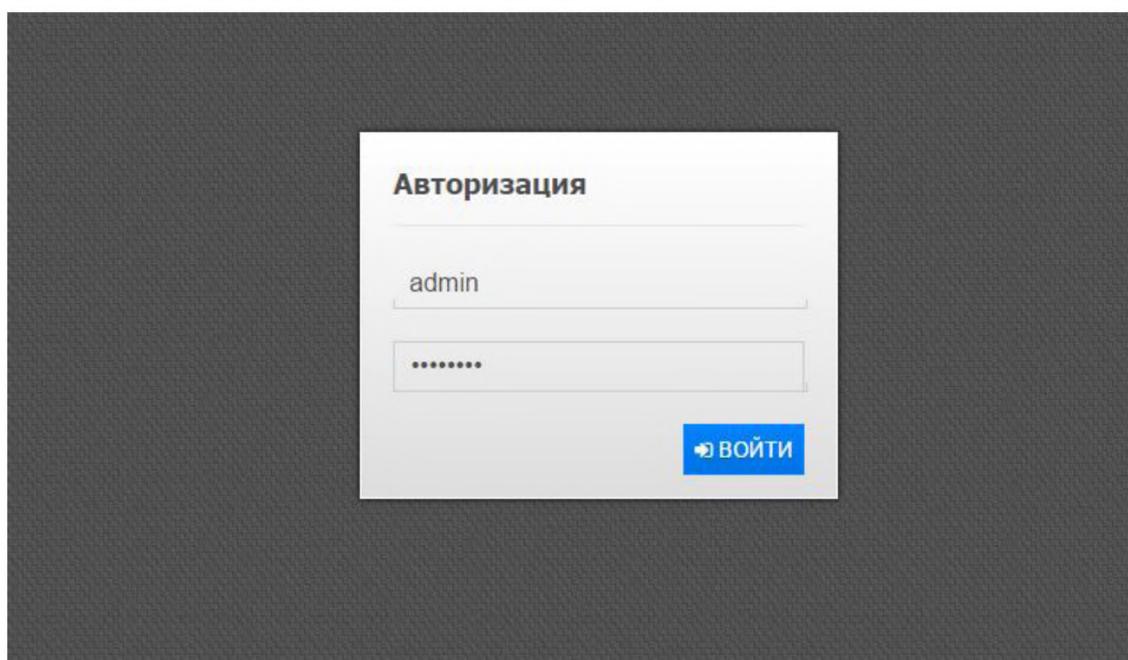


Рисунок 5 - Авторизация администратора

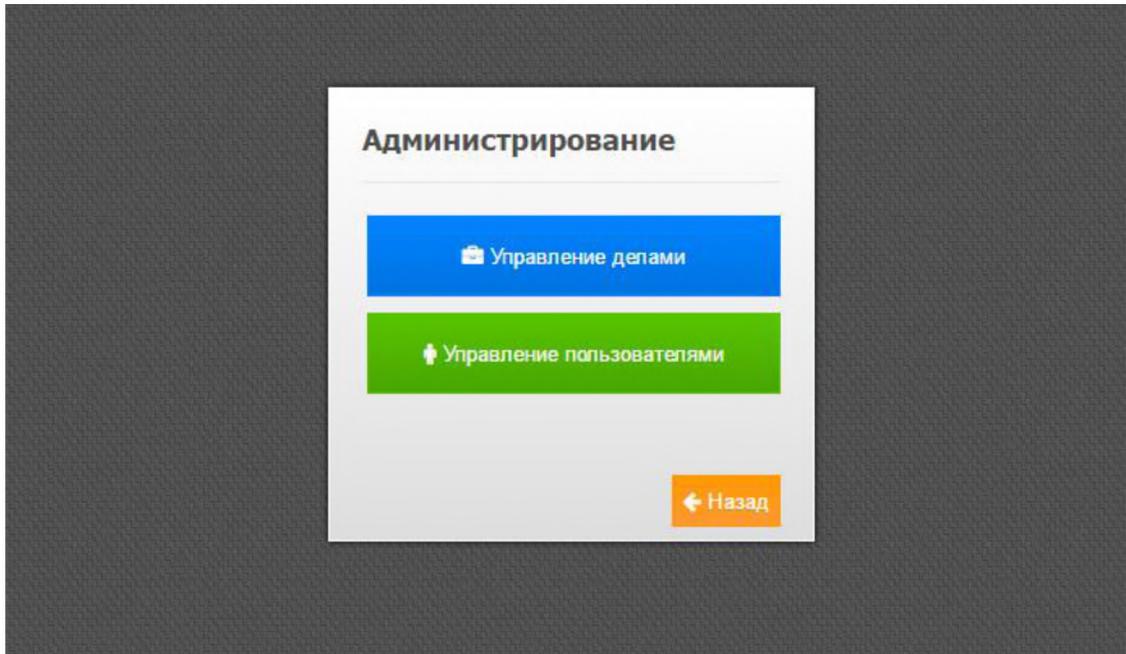


Рисунок 6 - Выбор управления пользователями или делами

2.4 Разработка функции отображения карты с метками

2.4.1 Структура отображения карты и меток

При выборе дела происходит переход к окну с картой и отрисовываем существующие в базе данных маркеры. Также присутствует функция расставления маркеров вручную. При нажатии на карту открывается “окно редактирования информации”. После нажатия кнопки “Save and Close” информация сохраняется в базу данных. Структура представлена на рисунке 7.



Рисунок 7 - Структура отображения карты

2.4.2 Программная реализация функции отображения карты и методов

Для отображения карты Google Maps в веб-приложении был создан файл «Map.php». Рассмотрим его содержимое.

Инициализация сессии.

```
<?php session_start();?>
```

```
<html>
```

```
  <head>
```

```
    <title>Map</title>
```

Подгружаем стили дизайна веб-приложения.

```
    <link      rel='stylesheet'      prefetch'      href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
```

```
    <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
```

```
    <meta name="viewport" content="initial-scale=1.0">
```

```
    <meta charset="utf-8">
```

```
    <style>
```

```
        html, body {
```

```
            height: 100%;
```

```
            margin: 0;
```

```
            padding: 0;
```

```
}
```

Выделение места под карту

```
#map {
```

```
    height: 100%;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
    <div id="map"></div>
```

```
    <form action="Map.php" method="POST">
```

```
        <button class="backMap" name="back"><span class="iconBack"></span>
```

Назад</button>

```
</form>
```

```
<?php
```

Работа кнопки “Назад”

```
if (isset($_POST['back']))  
{  
    header("Location: choosecase.php");  
}  
?>
```

```
<script>
```

```
p=0;  
var infoedit;  
var mark;
```

Выбор иконок.

```
var customIcons = {  
    crime: {  
        icon:
```

```
'http://s8.hostingkartinok.com/uploads/images/2016/04/4c134e45d8aad9d69b5bc769c5a910e4.  
png'  
},
```

```
witness: {  
    icon:
```

```
'http://s8.hostingkartinok.com/uploads/images/2016/05/b6098b344f7b20135e5040d0b9536520.  
png'  
}
```

```
};
```

Функция инициализации карты.

```
function initMap() {
```

Установка центра карты.

```
    var dot = new google.maps.LatLng(56.012219, 92.875246);
```

```
    downloadUrl("phpxml.php", function(data) {
```

```
        var xml = data.responseXML;
```

```
        var dots = xml.documentElement.getElementsByTagName("entity");
```

```
        var type = dots[0].getAttribute("type");
```

```
        var point = new google.maps.LatLng(
```

```
            parseFloat(dots[0].getAttribute("lat")),
```

```

parseFloat(dots[0].getAttribute("lng")));
if(type=='crime')
{
    p++;
    dot=point;
}
);
var mapOptions = {
zoom: 15,
center: dot
};

```

Инициализация карты.

```

var map = new google.maps.Map(document.getElementById("map"),
mapOptions);
poly = new google.maps.Polyline({
strokeColor: '#000000',
strokeOpacity: 1.0,
strokeWeight: 3
});
poly.setMap(map);
var html = "<table>" +
"<tr><td>Name:</td> <td><input id='name' /> </td> </tr>" +
"<tr><td>Info:</td> <td><input id='info' /></td> </tr>" +
"<tr><td>Type:</td> <td><select id='type'>" +
"<option value='crime' SELECTED>crime</option>" +
"<option value='witness'>witness</option>" +
"</select> </td></tr>" +
"<tr><td></td><td><td><input type='button' value='Save' & Close' onclick='saveData()' /></td></tr>";
infoedit = new google.maps.InfoWindow({
content: html
});

```

Добавление маркера.

```

google.maps.event.addListener(map, "click", function(event) {
var path = poly.getPath();

```

```
path.push(event.latLng);
mark = new google.maps.Marker({
  position: event.latLng,
  title: '#' + path.getLength(),
  map: map
});
```

Добавление окна редактирования информации.

```
google.maps.event.addListener(mark, "click", function() {
  infoedit.open(map, mark);
});
var infoWindow = new google.maps.InfoWindow;
```

2.4.3 Результат работы функции отображения карты и меток

После авторизации в качестве пользователя и выбора дела, появится окно с картой и размещенными метками (рисунок 8). На карте можно размещать свои маркеры и добавлять в них информацию, как представлено на рисунке 9, а также проложить путь людей.



Рисунок 8 - Отображение карты и меток

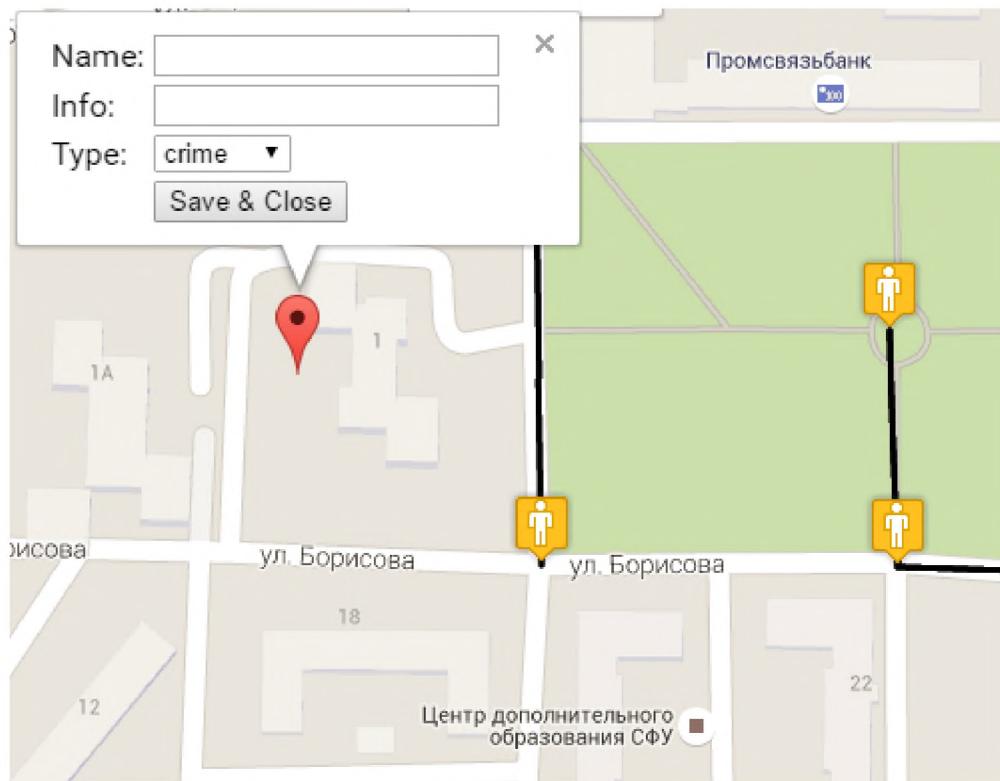


Рисунок 9 - Добавление маркера на карту

2.5 Разработка функции администрирования

2.5.1 Структура функции администрирования

Администратор может удалять, создавать дела и пользователей.
Структура представлена на рисунке 10.

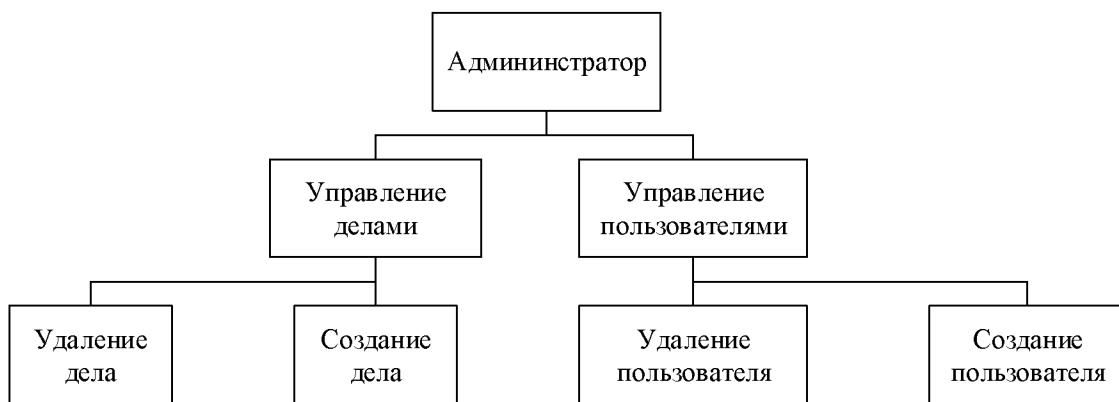


Рисунок 10 - Функция администрирования

2.5.2 Программная реализация функции администрирования

Были разработаны семь файлов «admin.php», «users.php», «adduser.php», «addcase.php», «deleteuser.php», «delet case.php» и «cases.php».

Рассмотрим содержимое первого файла «admin.php».

```
<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
    <title>Администрирование</title>
```

Подключение стилей.

```
<link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
<link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
</head>
```

Меню администратора.

```

<div class="box">
    <div class="containerWrapper">
        <div id="containerAdmin" class="tabContainer">
            <form action="admin.php" method="POST">
                <h2 class="loginTitle">Администрирование</h2>
                <div class="registerContent">
                    <button class="caseBox" name="cases"><span class="iconCase"></span>
                        Управление делами</button>
                    <button class="userBox" name="users"><span class="iconUser"></span>
                        Управление пользователями</button>
                    <button class="backBox" name="back"><span class="iconBack"></span> Назад</button>
                    <div class="clear"></div>
                </form>
            </div>
        </div>
    </div>

```

Рассмотрим файл «users.php».

С помощью этого файла отображается меню управления пользователями.

```

<html>
    <head>
        <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
        <title>Пользователи</title>
        <link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
        <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
    </head>
    <body>
        <div class="box2">
            <div class="containerWrapperUsers">
                <div id="containerUsers" class="tabContainerUsers">
                    <?php
                        $hostname = 'localhost';
                        $username = 'pma';

```

```

$spasswordname = '1';
$basename = 'wsrce';
$conn = new mysqli($hostname, $username, $passwordname,
$basename) or die ('Невозможно открыть базу');
$sql = "SELECT * FROM `users`";
$result = $conn->query($sql);
while ($row = $result->fetch_assoc())
{
    echo 'Логин: '.$row['login'];
    echo ' ';
    echo 'Электронная почта: '.$row['email'];
    echo '<br>';
}
?>
<form action="users.php" method="POST">
<div class="registerContent">
    <button class="deleteBox" name="delete"><span
class="iconDelete"></span> Удалить пользователя</button>
    <button class="addBox" name="add"><span
class="iconAdd"></span> Добавить пользователя</button>
    <button class="backBox" name="back"><span
class="iconBack"></span> Назад</button>
    <div class="clear"></div>
</div>
</div>
</div>

```

Рассмотрим файл «adduser.php».

Файл добавления пользователей.

```

<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
    <title>Добавление пользователя</title>

```

Подключение стилей.

```

<link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
    <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
</head>
<body>
<div class="box">
    <div class="containerWrapper">
        <div id="containerRegister" class="tabContainer">
            <form action="adduser.php" method="POST">
                <h2 class="loginTitle">Добавление пользователя</h2>
                <div class="registerContent">
                    <div class="inputWrapper">
                        <input type="text" name="login" placeholder="Логин" />
                    </div>
                    <div class="inputWrapper">
                        <input type="text" name="email" placeholder="Электронная почта" />
                    </div>
                    <div class="inputWrapper">
                        <input type="password" name="password" placeholder="Пароль"/>
                    </div>
                </div>
                <button class="redBox" name="back"><span
class="iconBack"></span> Назад</button>
                <button class="greenBox" name="submit"><span class="iconAdd"></span>
                    Добавить пользователя</button>
                <div class="clear"></div>
            </form>
        </div>
    </div>

```

Рассмотрим файл «addcase.php».

Добавление дела.

```

<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />

```

```

<title>Создание уголовного дела</title>
    <link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-
awesome/3.2.0/css/font-awesome.min.css'>
        <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
    </head>
    <body>

        <div class="box">
            <div class="containerWrapper">
                <div id="containerRegister" class="tabContainer">
                    <form action="addcase.php" method="POST">
                        <h2 class="loginTitle">Создание уголовного дела</h2>
                        <div class="registerContent">
                            <div class="inputWrapper">
                                <input type="text" name="case" placeholder="Номер дела" />
                            </div>
                            <button class="redBox" name="back"><span
class="iconBack"></span> Назад</button>
                            <button class="greenBox" name="submit"><span class="iconAdd"></span>
Создать уголовное дело</button>
                            <div class="clear"></div>
                        </form>
                    </div>
                </div>
            </div>
        
```

Рассмотрим файл «deleteuser.php».

Удаление пользователей.

```

<html>
    <head>
        <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
        <title>Удаление пользователя</title>
        <link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-
awesome/3.2.0/css/font-awesome.min.css'>
            <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
        </head>

```

```

<body>
<div class="box">
    <div class="containerWrapper">
        <div id="containerDelete" class="tabContainer">
            <form action="deleteuser.php" method="POST">
                <h2 class="loginTitle">Удаление пользователя</h2>
                <div class="registerContent">
                    <div class="inputWrapper">
                        <input type="text" name="login" placeholder="Логин " />
                    </div>
                </div>
                <button class="deleteBox" name="delete"><span class="iconDelete"></span> Удалить пользователя</button>
                <button class="redBox2" name="back"><span class="iconBack"></span> Назад</button>
            </div>
        </div>
    </div>

```

Рассмотрим файл «deletecase.php».

Удаление дел.

```

<html>
    <head>
        <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
        <title>Удаление уголовного дела</title>
        <link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
        <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
    </head>
    <body>

        <div class="box">
            <div class="containerWrapper">
                <div id="containerRegister" class="tabContainer">
                    <form action="deletecase.php" method="POST">
                        <h2 class="loginTitle">Удаление уголовного дела</h2>

```

```

<div class="registerContent">
    <div class="inputWrapper">
        <input type="text" name="case" placeholder="Номер дела" />
    </div>
    <button class="deleteBox" name="delete"><span
class="iconDelete"></span> Удалить уголовное дело</button>
    <button class="redBox2" name="back"><span
class="iconBack"></span> Назад</button>
    <div class="clear"></div>
</form>
</div>
</div>
</div>

```

Рассмотрим файл «cases.php».

Управление уголовными делами.

```

<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
    <title>Дела</title>
    <link rel='stylesheet' prefetch' href='http://netdna.bootstrapcdn.com/font-
awesome/3.2.0/css/font-awesome.min.css'>
    <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
</head>
<body>
<div class="box2">
    <div class="containerWrapperUsers">
        <div id="containerUsers" class="tabContainerUsers">
            <?php
                $hostname = 'localhost';
                $username = 'pma';
                $passwordname = '1';
                $basename = 'wsrce';
                $conn = new mysqli($hostname, $username, $passwordname,
$basename) or die ('Невозможно открыть базу');
                $sql = "SHOW TABLES FROM $basename";

```

```

$result = $conn->query($sql);
while ($row = $result->fetch_row())
{
    if($row[0]=='users') {
        echo 'Дело: '.$row[0];
        echo '<br>';
    }
}
?>
<form action="cases.php" method="POST">
<div class="registerContent">
<button class="deleteBox" name="delete"><span
class="iconDelete">Удалить уголовное дело</span>
<button class="addBox" name="add"><span
class="iconAdd">Создать уголовное дело</span>
<button class="backBox" name="back"><span
class="iconBack">Назад</button>
<div class="clear"></div>
</div>
</div>
</div>

```

2.5.3 Результат работы функции администрирования

В окне управления пользователями, которое представлено на рисунке 11, доступно удаление (рисунок 12) и добавление пользователей (рисунок 13).

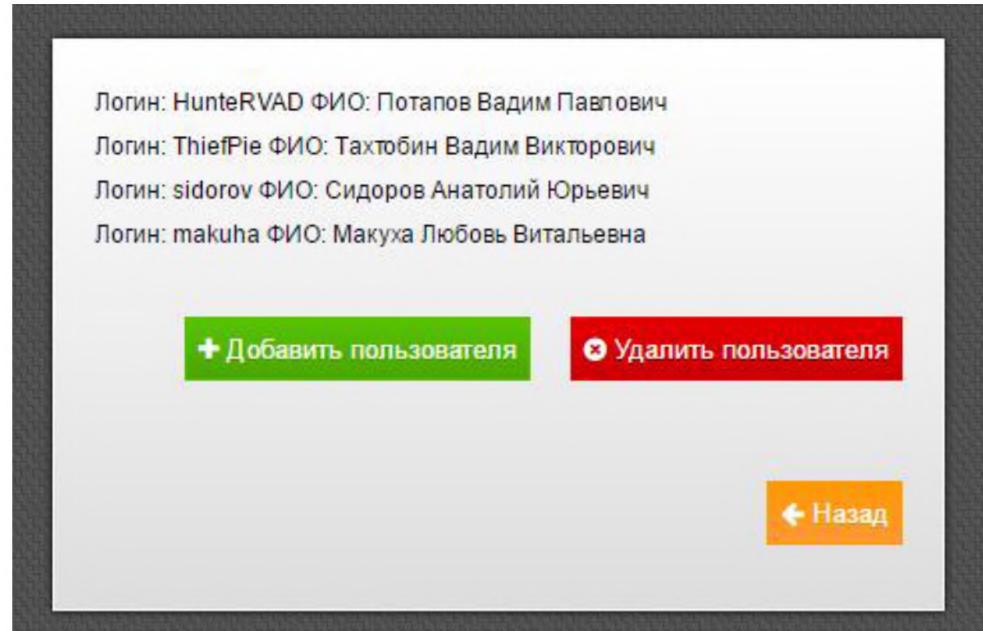


Рисунок 11 - Управление пользователями

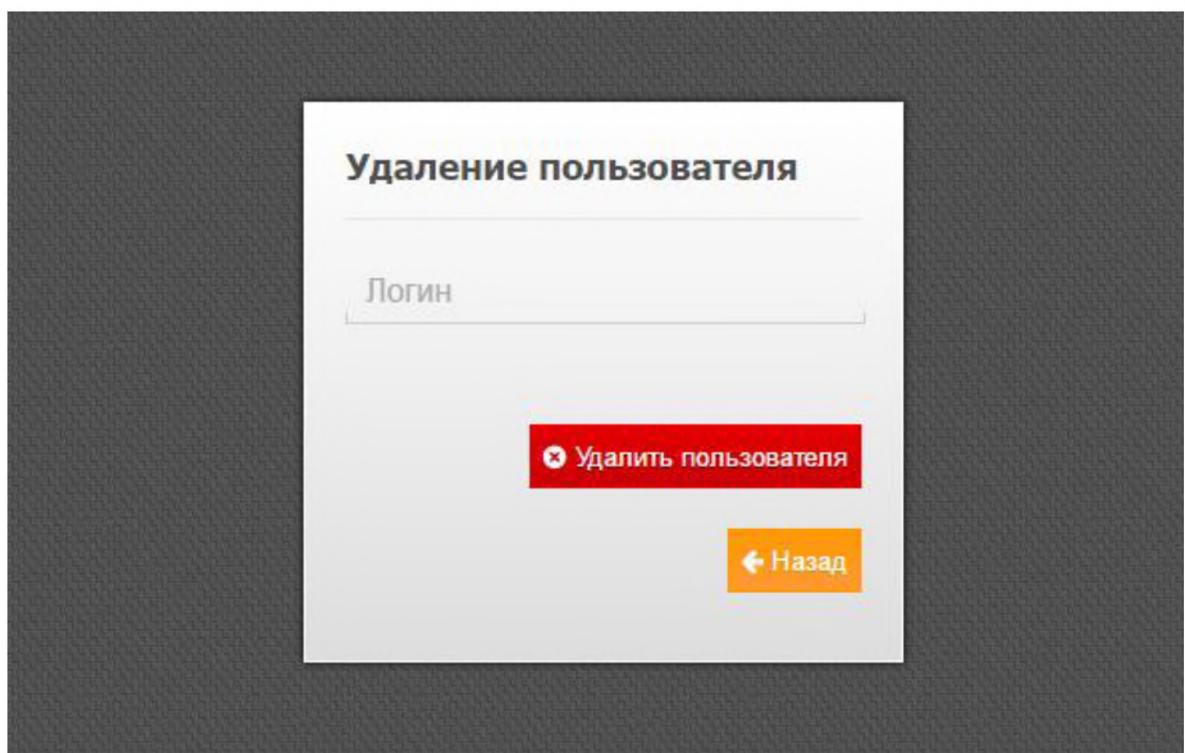


Рисунок 12 - Удаление пользователя

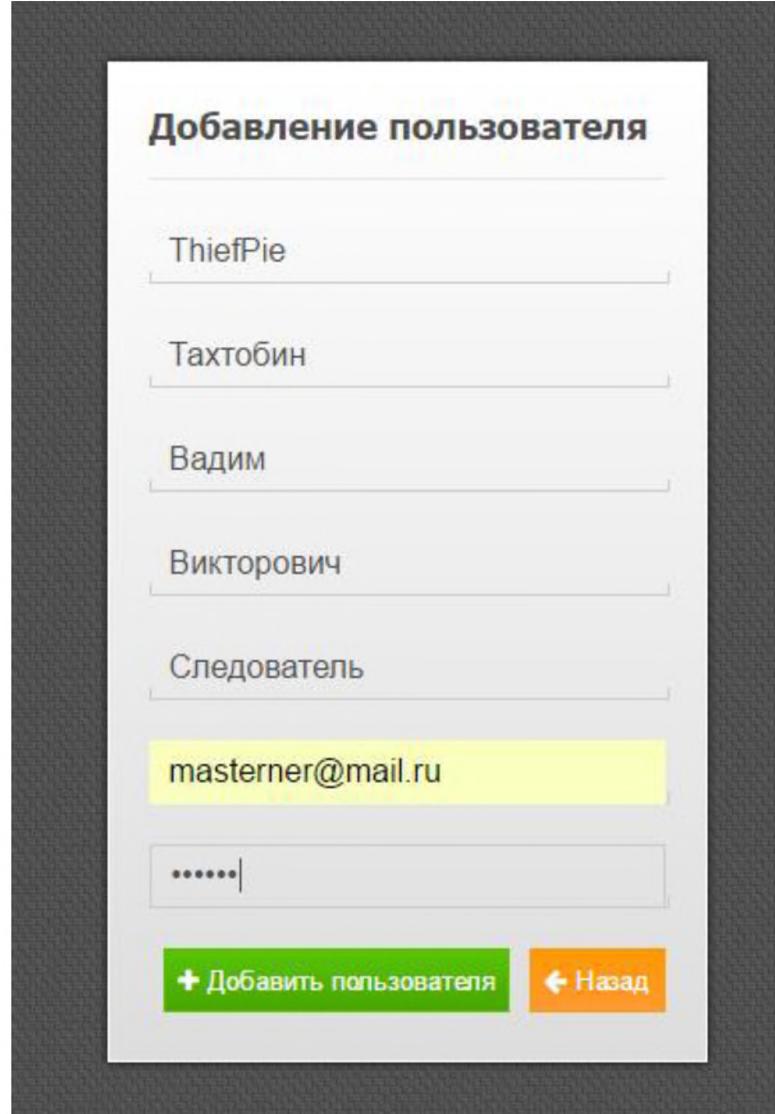


Рисунок 13 - Добавление пользователя

В окне управления делами, показанного на рисунке 14, доступно удаление (рисунок 15) и создание дел (рисунок 16).

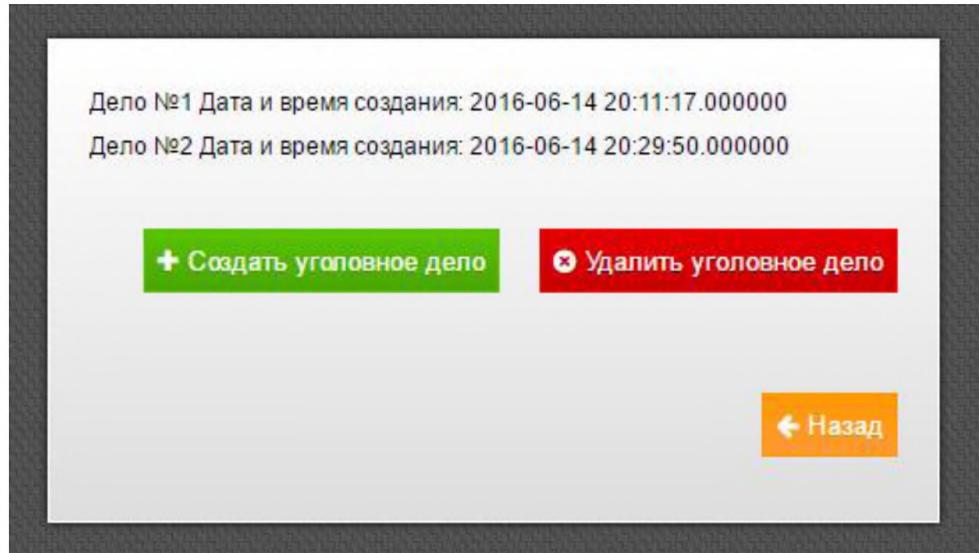


Рисунок 14 - Управление делами

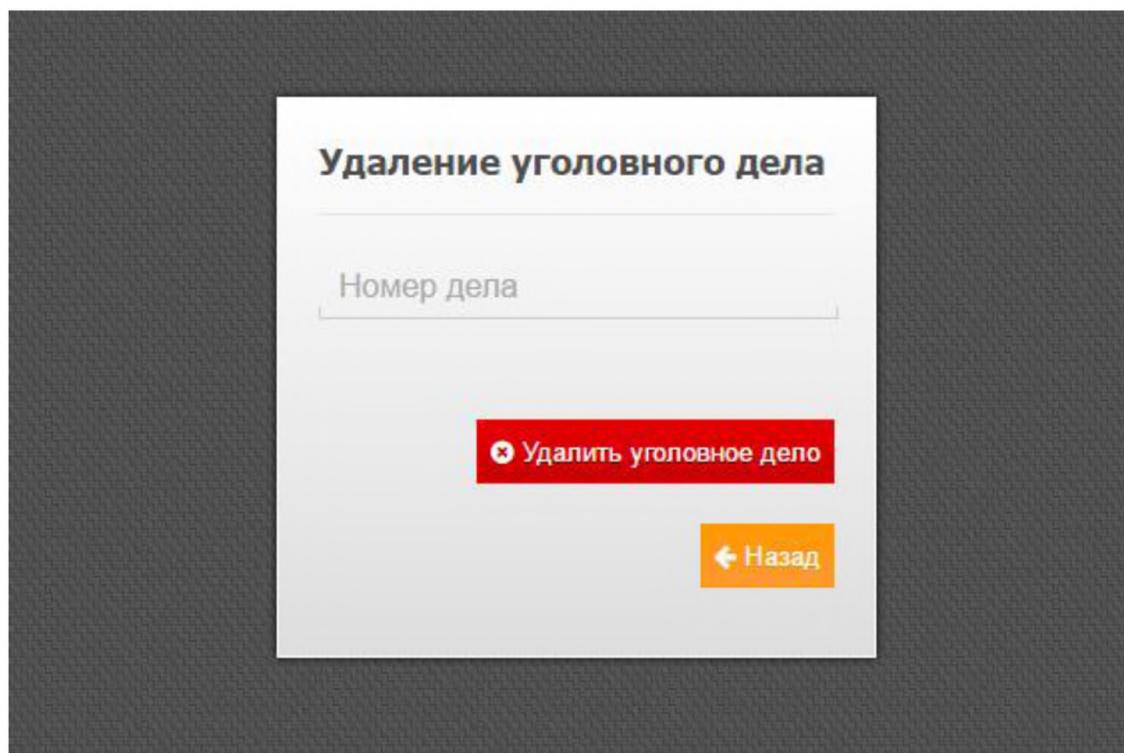


Рисунок 15 - Удаление уголовного дела

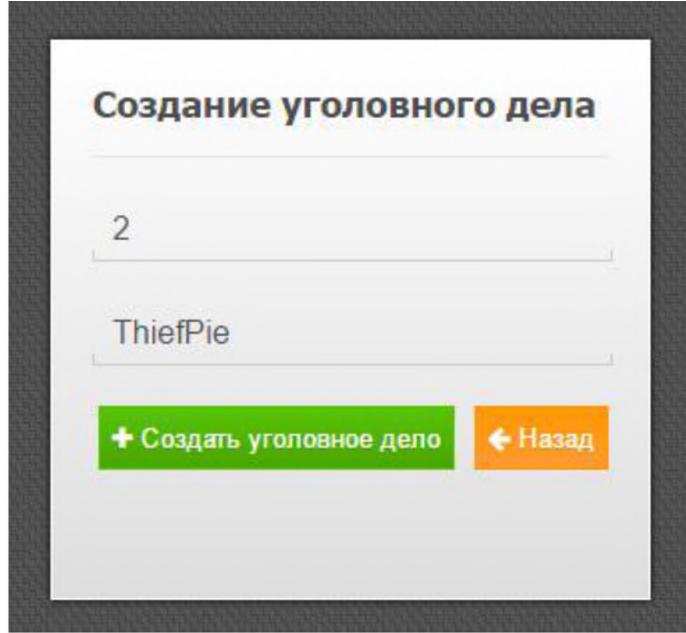


Рисунок 16 - Создание уголовного дела

2.6 Разработка функции выбора дела

2.6.1 Структура функции выбора дела

В данном окне будут доступны на выбор уже созданные дела. Структура выбора дела представлена на рисунке 17.



Рисунок 17 - Структура выбора дел

2.6.2 Программная реализация функции выбора дела

Для реализации данной функции был разработан файл «choosecase.php». Рассмотрим его содержимое.

Начало сессии.

```
<?php
session_start();
?>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
<title>Дела</title>
```

Подключение стилей.

```
<link rel='stylesheet prefetch' href='http://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css'>
<link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />
</head>
```

Инициализация окна выбора дел.

```
<body>
<div class="box2">
<div class="containerWrapperUsers">
<div id="containerUsers" class="tabContainerUsers">
```

Отображение списка дел.

```
<?php
$hostname = 'localhost';
$username = 'pma';
$passwordname = '1';
$basename = 'wsrce';
$conn = new mysqli($hostname, $username, $passwordname,
$basename) or die ('Невозможно открыть базу');
$sql = "SHOW TABLES FROM $basename";
$result = $conn->query($sql);
while ($row = $result->fetch_row())
{
    if($row[0]=='users') {
        else
    {
        echo 'Дело: '.$row[0];
        echo '<br>';
```

```

        }
    }
?>

<form action="choosecase.php" method="POST">
<div class="registerContent">
<div class="inputWrapper">
<input type="text" name="case" placeholder="Номер дела" />
</div>
<button class="caseBox2" name="choose"><span
class="iconCase">Выбрать уголовное дело</span>
<button class="back2Box" name="back"><span
class="iconBack">Назад</button>
<div class="clear"></div>
</div>
</div>
</div>

```

Обработка кнопок.

```

<?php
if (isset($_POST['choose']))
{
    $choice=$_POST['case'];
    $_SESSION['choice']=$choice;
    header("Location: Map.php");
}
if (isset($_POST['back']))
{
    header("Location: login.php"); } ?>

```

2.6.3 Результат работы функции выбора дела

При авторизации в качестве пользователя будет доступен выбор между доступными делами для последующей работы на карте, как показано на рисунке 18.

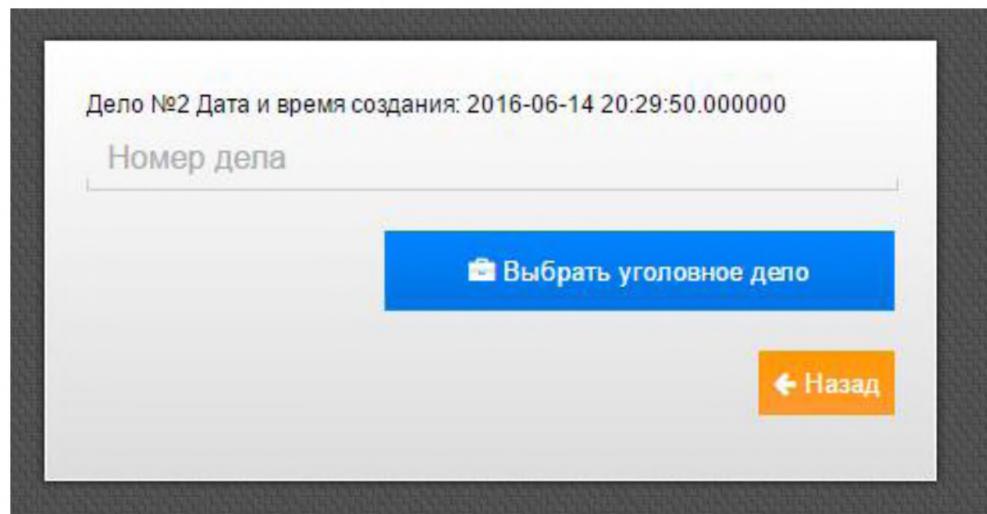


Рисунок 18 - Функция выбора дел

ЗАКЛЮЧЕНИЕ

В результате выполнения ВКР было разработано веб-приложение. Данная веб-система создана для следователей. С его помощью они смогут визуализировать местоположение людей в момент совершения преступления, что позволит сэкономить время следователям, так как им будет удобнее расследовать дело с полностью отстроенной картиной произошедшего.

Также для полной работы веб-приложения потребуются администраторы, которые будут предоставлять доступ следователям и создавать новые дела.

При разработке веб-приложения было рассмотрено исчерпывающее количество современных технологий веб-разработки. Разработанное веб-приложение соответствует всем требованиям, приведенным в техническом задании.

В будущем возможно добавление большого количества функций и для этого поможет Google Maps API. Возможна оптимизация кода и доработка интерфейса для уменьшения времени загрузок и улучшения интуитивности.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Habrahabr.ru [Электронный ресурс]: Руководство по PHP Яндекс.Карты, 2ГИС или всё же Google Maps? - Режим доступа: <https://habrahabr.ru/post/242015>
2. PHP.net [Электронный ресурс]: Руководство по PHP - Режим доступа: <http://php.net/manual/ru/preface.php>
3. Клуб программистов [Электронный ресурс]: PHP, его минусы и плюсы - Режим доступа: <http://www.programmersclub.ru/php-его-минусы-и-плюсы>
4. Свободная энциклопедия [Электронный ресурс]: MySQL- Режим доступа: <https://ru.wikipedia.org/wiki/MySQL>
5. Свободная энциклопедия [Электронный ресурс]: JavaScript- Режим доступа: <https://ru.wikipedia.org/wiki/JavaScript>
6. J-S [Электронный ресурс]: Уроки по JavaScript - Режим доступа: <http://www.j-s.ru>
7. СТО 4.2-07-2014 Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. – Введ. 22.12.2014. – Красноярск: ИПК СФУ, 2014. – 41 с.