СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc533373307)

[ОСНОВНАЯ ЧАСТЬ 6](#_Toc533373308)

[1. Установка и настройка операционной системы с программным обеспечением 6](#_Toc533373309)

[1.1. Подготовка дистрибутива для установки ОС Windows 6](#_Toc533373310)

[1.2. Установка Windows 10 7](#_Toc533373311)

[1.3. Установка и настройка программного комплекса XAMPP 8](#_Toc533373312)

[1.4. Установка и настройка и JetBrains PhpStorm 2017.2 8](#_Toc533373313)

[2. Структура и описание информационной системы 11](#_Toc533373314)

[2.1. Базы данных 11](#_Toc533373315)

[2.2. Описание системы. Функционал 12](#_Toc533373316)

[2.3. Размещение системы на хостинге 18](#_Toc533373317)

[ЗАКЛЮЧЕНИЕ 18](#_Toc533373318)

[СПИСОК ЛИТЕРАТУРЫ 18](#_Toc533373319)

[ПРИЛОЖЕНИЕ 19](#_Toc533373320)

# ВВЕДЕНИЕ

**Наименование программы**

Система кадрового учета - программа, предназначенная для автоматизации работы сотрудников отдела кадров, а также для получения сотрудниками других отделов информации о размере заработной платы, штрафов, коэффициенте трудового участия и других персональных данных. Кроме того, пользователь может отслеживать своё положение в фирме: свою должность, свой статус (устраивается, работает, уволен).

**Функциональное и эксплуатационное назначение**

Система кадрового учета имеет такие функции как: вывод и изменение информации о сотрудниках, добавление новых пользователей, удаление соответствующих данных.

**Актуальность**

Существует возможность внедрения данной программы в какую-либо фирму для автоматизации и удобства взаимодействия начальства и подчинённых.

**Описание и обоснование метода выбора программных средств**

Для успешной работы программы на локальном сервере (localhost) на ПК должны быть установлена операционная система (ОС) Windows 10. Именно эта ОС наиболее удобна для пользователя, благодаря грамотно скомпонованному интерфейсу и информативной простой персонализации.

Кроме того, необходимо установить и настроить программное обеспечение XAMPP и JetBrains PhpStorm 2017.2. Кроме того, программный комплекс может быть размещён в сети Интернет с помощью перехода на сервер. В процессе разработки данной программы были использован языки программирования PHP и SQL, CSS - язык описания внешнего вида документа, написанного с использованием языка разметки HTML, а также СУБД MySQL.

**Постановка задачи**

При разработке программного продукта должны быть выполнены ряд определенных задач:

1) Разработать удобное и понятное для пользователя приложение.

2) Использовать базы данных.

3) Добавить в функционал программы такие функции, как:

а) авторизация в программу через логин и пароль;

б) регистрация нового пользователя в программе;

в) получение информации необходимой информации в зависимости от темы курсового проекта;

г) добавление необходимой информации вручную;

д) удаление ненужной пользователю информации (также вручную).

Важно отметить, что пользователь не может ввести неверные данные в поля для ввода, например, буквы в числовые поля. Также недоступен ввод пустых полей.

Пользователи, не авторизованные в программе, не смогут получить к ней доступ (данный механизм реализован через PHP-сессии).

# ОСНОВНАЯ ЧАСТЬ

## 1. Установка и настройка операционной системы с программным обеспечением

### 1.1. Подготовка дистрибутива для установки ОС Windows

Дистрибутив операционной системы можно разместить на USB-флеш-накопителе или же на оптическом диске. С учетом того, что оптические диски теряют актуальность, удобнее разместить ОС на USB-флеш-накопителе.

Необходимо иметь флеш-накопитель такого объема, который способный вместить дистрибутив операционной системы Windows. Для установочной флешки для Windows 10 минимальный рекомендуемый объём составит 16 гигабайт. Кроме того, понадобится ISO образ данной операционной системы, который можно получить на официальном сайте Microsoft.

Существует несколько способов создания загрузочной флешки, которые в основном можно разделить по классификации:

- ручной;

- автоматический.

Рассмотрим автоматический способ, чтобы адаптировать пользователя к данной операционной системе. Тем самым подготовив его к тому, с чем придется столкнуться при обновлении ОС.

Для автоматического создания загрузочной флешки необходимо специальное программное обеспечение. Воспользуемся одной из самых популярных программ Rufus 3.

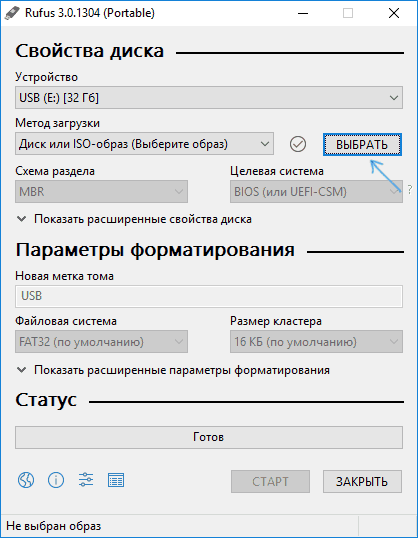


Рисунок 1 - Главное окно программы Rufus 3.

После запуска программы, в поле «Устройство» нужно выбрать USB-флеш-накопитель на который будет производиться запись Windows 10. Далее необходимо нажать кнопку «Выбрать» и указать образ ISO.

В поле «Схема разделов» нужно выбрать схему разделов целевого диска (на который будет устанавливаться система) — MBR (для систем с Legacy/CSM загрузкой) или GPT (для систем UEFI). Настройки в разделе «Целевая система» переключатся автоматически.

Далее в разделе «Параметры форматирования» при желании указать метку флешки.

Также можно указать файловую систему для загрузочной флешки, в том числе возможно использование NTFS для флешки UEFI.

После этого нужно нажать «Старт» и подтвердить, что данные с флешки будут удалены, а затем дождаться завершения копирования файлов из образа на USB накопитель. По завершении процесса нажмите кнопку «Закрыть», чтобы выйти из программы.

### 1.2. Установка Windows 10

Установка операционной системы Windows 10 на ПК начинается после создания дистрибутива данной системы. Перед началом установки необходимо зайти в BIOS/UEFI и настроить очередность запуска запоминающих устройств, установив флэш-накопитель на первое место. Это делается для того, чтобы при загрузке компьютера происходила именно с флэш-накопителя.

После перезагрузки компьютера запускается утилита по установке Windows. В первом окне необходимо выбрать язык, формат времени и язык раскладки клавиатуры. Далее нужно нажать кнопку «Установить» для начала процедуры установки ОС. Затем следует ознакомиться с представленными условиями лицензии и принять их. Далее выбирается тип установки: обновление или выборочная установка. Если до момента установки Windows 10, на компьютере не было системы Windows более старой версии, то следует выбрать «Выборочный» тип. После этого откроется окно настройки разделов жесткого диска. На этом этапе пользователь может разделить физический диск компьютера на несколько логических разделов с определенными размерами или оставить один раздел, в зависимости от своих предпочтений и от требований выполняемых задач. После выбора диска начнется процесс установки Windows 10.

На следующем этапе следует выбрать базовые параметры персонализации и местоположения, а также подключения и отчетов об ошибках. Здесь вы можно выбрать вариант «Использовать стандартные параметры (по умолчанию)» или указать настройки вручную. Далее нужно выбрать режим использования компьютера в сети. Если это домашний компьютер, следует выбрать вариант «Я владею этим компьютером». Если же компьютер нужно подключить к доменной сети, выберите вариант «Моя организация».

Наконец появляется окно создания учетной записи. Пользователю предлагается два варианта: вход в систему с учетной записью Microsoft или же создание локальной учетной записи. Если выполнен вход при помощи учетной записи Microsoft, то станет будет доступна авторизация в Windows при помощи ПИН-кода. Возможно воспользоваться такой возможностью нажав кнопку Использовать ПИН-код или пропустить этот шаг. Далее появится предложение синхронизации с One Drive. В зависимости от предпочтений, выбираем «Далее», если нужна синхронизация или «По умолчанию сохранять новые файлы только на этом ПК» если такой необходимости нет.

После загрузки рабочего стола остается только выбрать режим доступа к компьютеру по локальной сети.

### 1.3. Установка и настройка программного комплекса XAMPP

XAMPP полностью бесплатный и простой в установке дистрибутив Apache, содержащий СУБД MySQL, языки программирования PHP и Perl.

#### 1.3.1. Установка

Запустим файл установки «setup».

Рисунок 2 - Загрузочный файл XAMPP.

Далее убедимся, что выбраны все необходимые компоненты. Затем выберем директорию для установки (рекомендуемый путь C:\xampp). Далее, чтобы начать установку нажмём «Next». По завершении убедимся, что стоит флажок «Do you want to start the Control Panel now?» и нажмём «Finish».

#### 1.3.2. Настройка

Для вызова настроек панели XAMPP в запущенном XAMPP Control Panel нажмем «Config».

Рисунок 3 - Изображение кнопки «Config».

В группе «Autostart of modules» поставим флажок «Apache», «MySQL» и нажмём «Save». Перезапустим XAMPP, чтобы изменения вступили в силу.

Для изменения уровня отображения ошибок PHP откроем файл «C:\xampp\php\php.ini» в любом редакторе и в нем поставим значение:

- error\_reporting = E\_ALL

При этом значении будут отображаться только критические ошибки PHP.

После внесения изменений в файл конфигурации PHP (php.ini) так же нужно перезагружать сервер Apache. В контрольной панели XAMPP напротив «Apache» нажать кнопку «Stop», если сервис уже запущен, дальше после остановки нажать кнопку «Start».

### 1.4. Установка и настройка и JetBrains PhpStorm 2017.2

Для начала скачиваем программу и запускаем файл. Далее выбираем директорию для установки программы, 32-х или 64-х битную пусковую установку, типы создаваемых файлов (в данной работе нам понадобятся файлы с расширениями .php, .css, .html). Затем нажимаем «Next» и запускаем установку.

После установки для корректной работы программы необходимо произвести некоторые настройки:

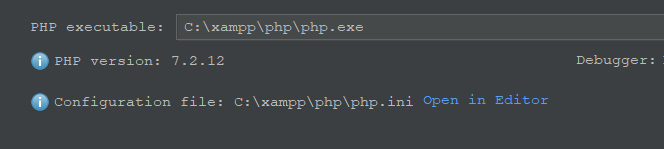
а) Для подключения интерпретатора PHP в настройках в секции «Languages & Frameworks» выберем пункт «PHP». В этой секции необходимо назначить интерпретатор для IDE в пункте «Interpreter» нажав на кнопку «…». В открывшемся окне, в поле «Name» вводим любое имя для удобства дальнейшей идентификации, например: PHP 7.2.12. В поле «PHP executable» указываем путь до интерпретатора.

Рисунок 4 - Окно настройки интерпретатора.

б) Для подключения базы данных в секции «View» выбираем «Tool Windows», «Database». Далее видим окно баз данных.

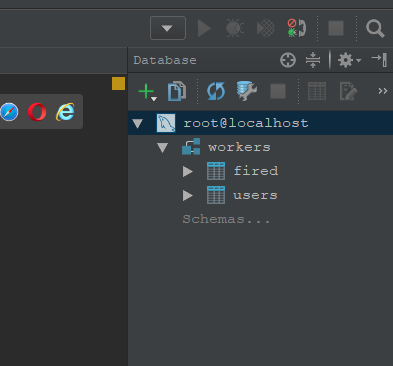


Рисунок 5 - Окно баз данных.

Нажимаем на значок плюса, выбираем «Data Source», «MySQL». В появившемся окне указываем хост, название базы данных, имя пользователя, пароль и URL.

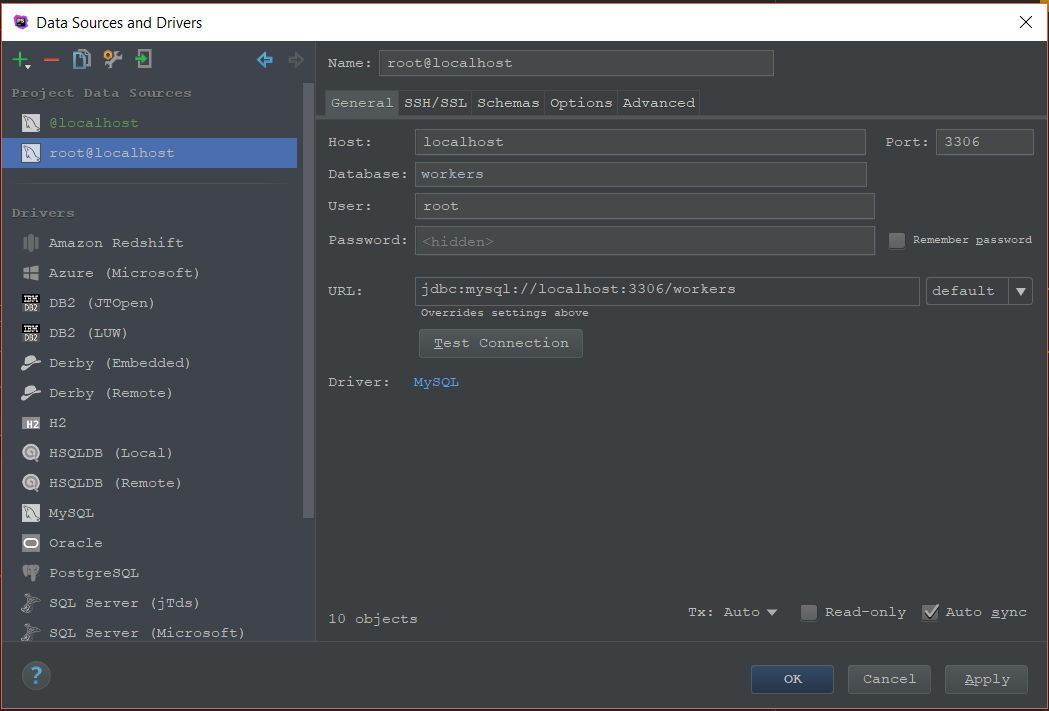


Рисунок 6 - Окно настройки базы данных.

Далее нажимаем «Apply», чтобы изменения вступили в силу. После того, как нажмём «Ок», в окне справа появится схема базы данных.

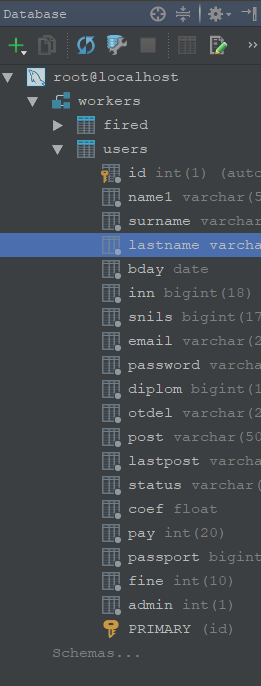


Рисунок 7 - База данных «workers».

2. Структура и описание информационной системы

## 2.1. Базы данных

При проектировании системы была создана 1 база данных с названием «workers» и в ней 2 таблицы «users» и «fired».

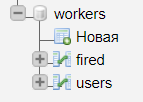


Рисунок 8 - Структура базы данных.

Таблица «users» хранит в себе данные о тех, кто устраивается на работу в фирму и о сотрудниках, а таблица «fired» хранит персональные данные об уволенных.

В каждой из таблиц определено 19 полей:

* Id - уникальный номер пользователя, первичный ключ с параметром AUTO\_INCREMENT;
* name1 - имя пользователя;
* surname - фамилия пользователя;
* lastname - отчество пользователя;
* bday - дата рождение пользователя;
* inn - ИНН пользователя;
* snils - СНИЛС пользователя;
* email - электронная почта пользователя;
* password - пароль пользователя;
* diplom - номер диплома пользователя;
* otdel - отдел, где работает/или куда устраивается пользователь;
* post - должность пользователя;
* status - стастус пользователя;
* coef - коэффициент трудового участия пользователя;
* pay - размер заработной платы пользователя;
* passport - номер и серия паспорта пользователя;
* fine - сумма штрафов пользователя;
* admin - проверка на администратора (0-пользователь,1-администратор).

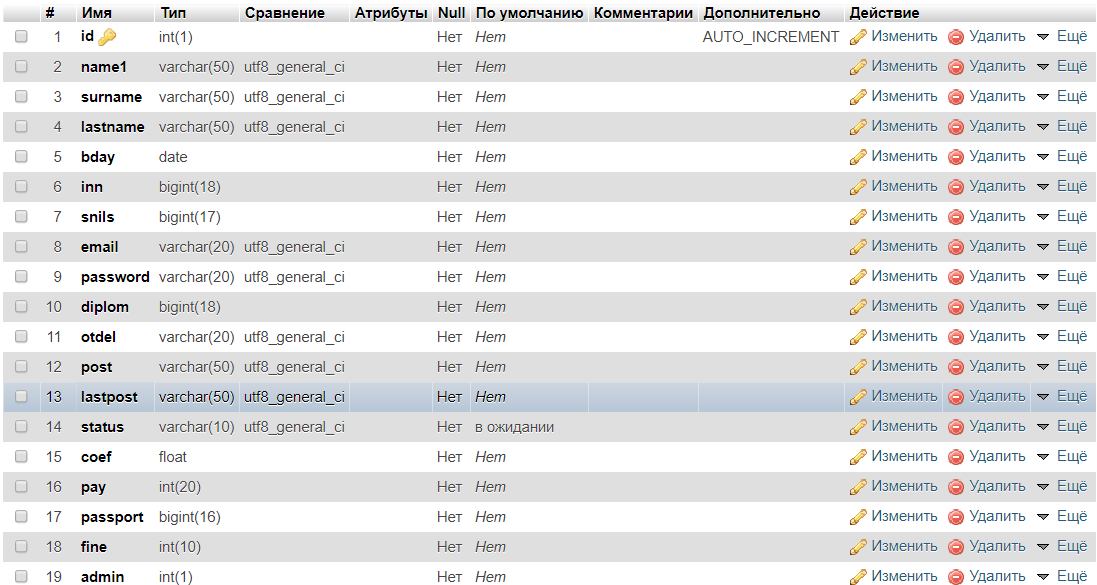
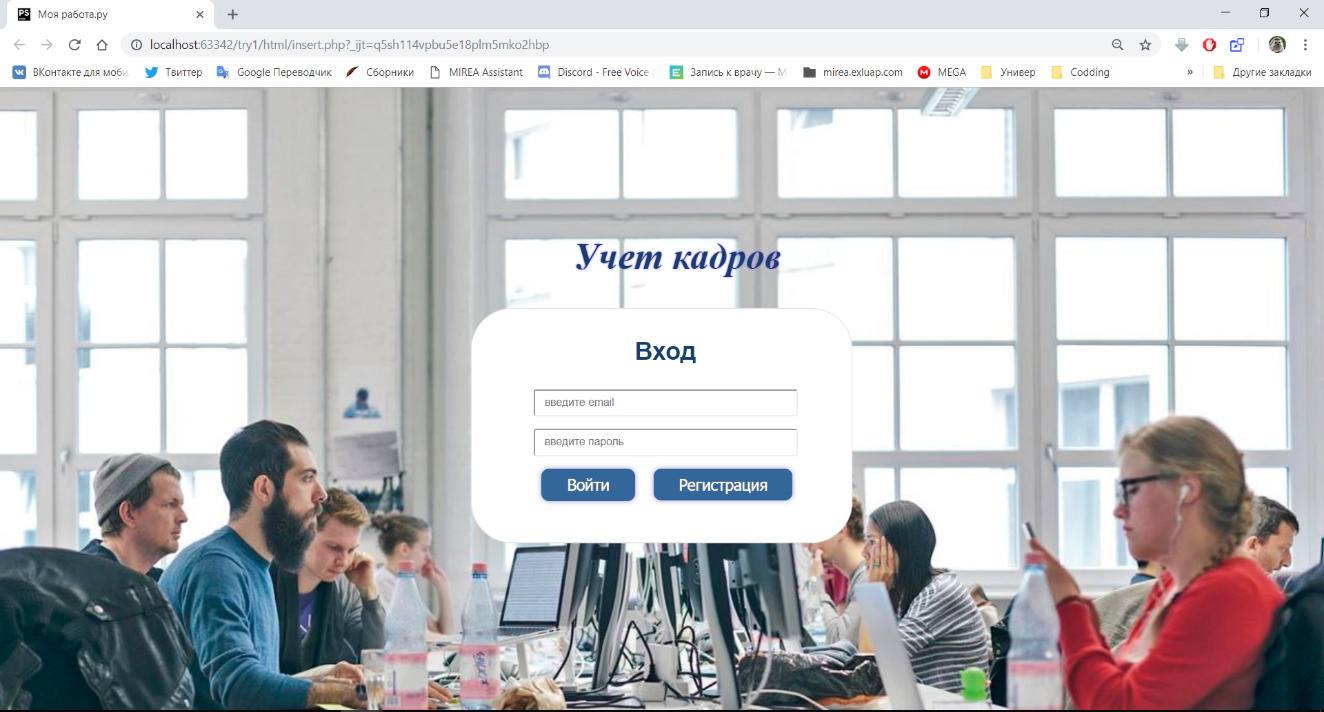


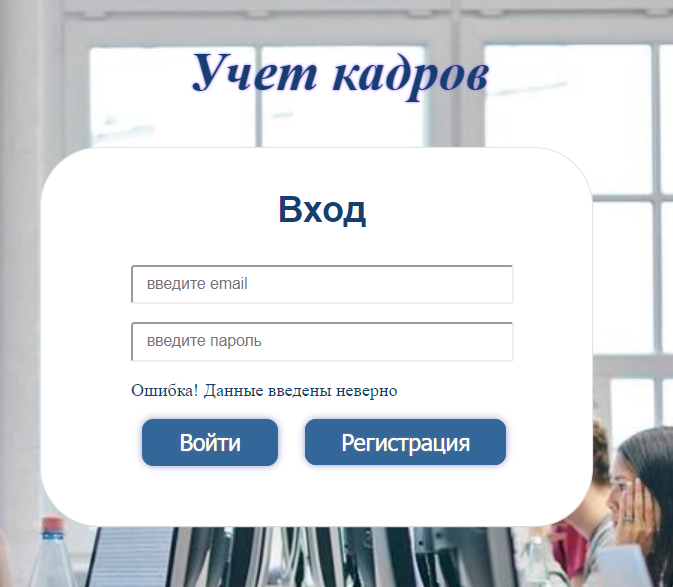
Рисунок 9 - Список полей.

2.2 Описание системы. Функционал

После запуска программного продукта, пользователь попадает на страницу авторизации (insert.php).

 Рисунок 10 - Страница авторизации.

Здесь ему предлагается ввести пароль и e-mail для входа в систему. При этом проверяется, существует ли определённый пользователь. Если нет, то система выдаёт ошибку, иначе создаётся сессия для конкретного пользователя, куда записывается пароль, и осуществляется переход на главную страницу сайта.

Рисунок 11 - Ошибка авторизации.

Код для кнопки «Войти»:

include\_once ('../php/bd\_connection.php');

// начало сессии

session\_start();

$res\_all = $mysqli->query("SELECT \* FROM users");

// авторизация

if(isset($\_GET['login\_button'])) {

while ($sess = $res\_all->fetch\_assoc()) {

foreach ($sess as $key => $value) {

// проверка введённых данных

if ($\_GET['email'] == $sess['email'] && $\_GET['pass'] == $sess['password']) {

$\_SESSION['password']=$sess['password'];

header("Location:http://localhost:63342/try1/html/introduction.php");

break;

} elseif ($\_GET['pass'] != $sess['password']) {

$message = 'Ошибка! Данные введены неверно';

} else {

$message = 'Ошибка! Данные введены неверно';

}

}

}

}

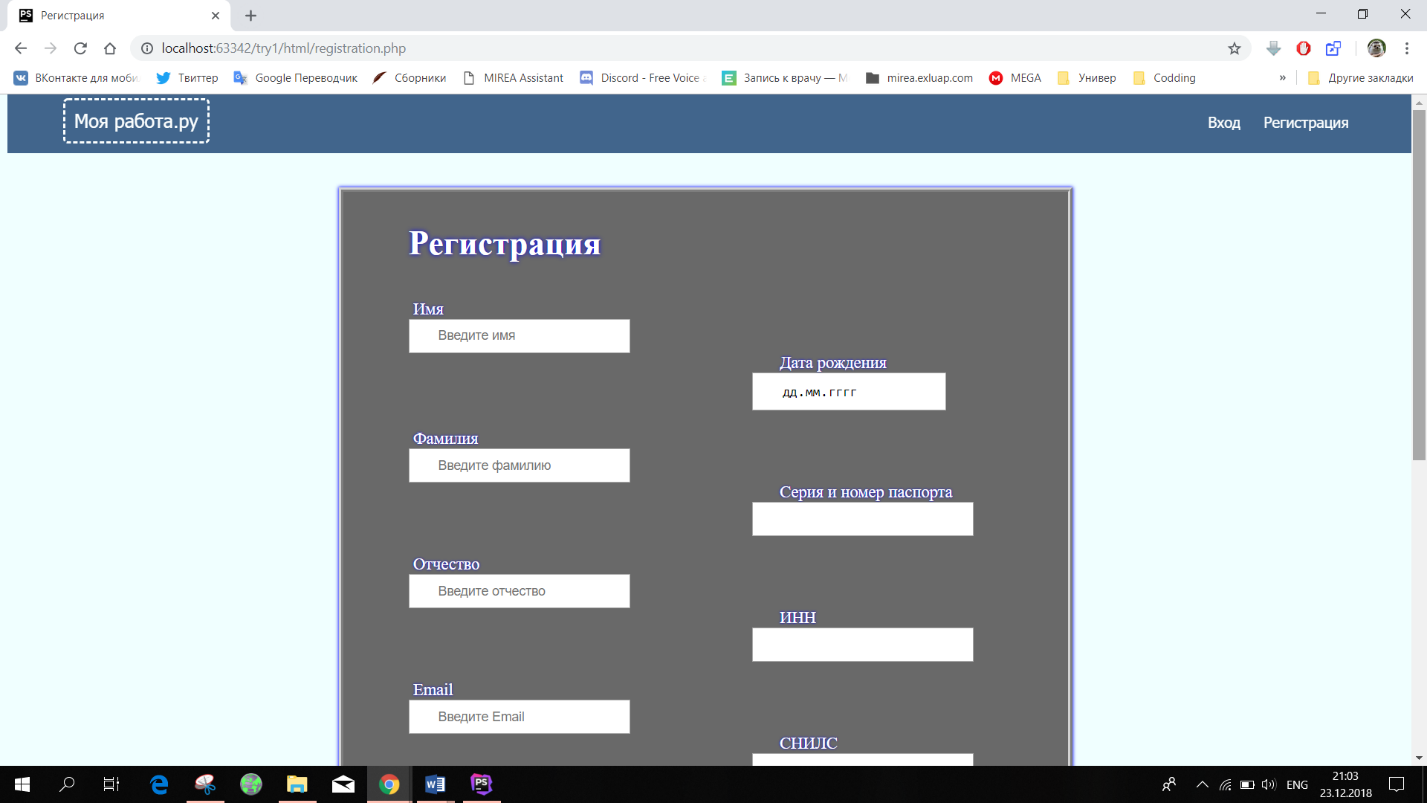
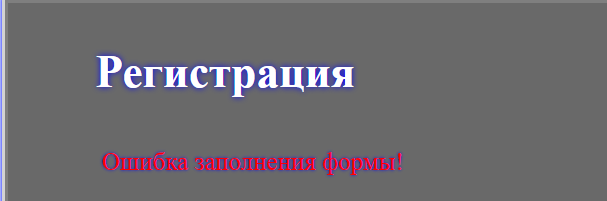
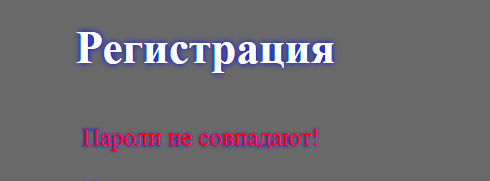
Для того, чтобы использовать систему, нужно зарегистрироваться. Нажав кнопку «Регистрация», осуществится переход на страницу регистрации.

Рисунок 12 - Страница регистрации.

В представленной форме все поля для заполнения обязательны, причем данные о паспорте, ИНН, СНИЛС и номер диплома должны быть представлены последовательностью цифр, иначе возникает ошибка.

Рисунок 13 - Ошибка заполнения формы.

Также, производится проверка e-mail и проверка на совпадение паролей.

Рисунок 14 - Ошибка проверки пароля.

После успешной регистрации пользователь может вернуться на страницу авторизации и войти в систему.

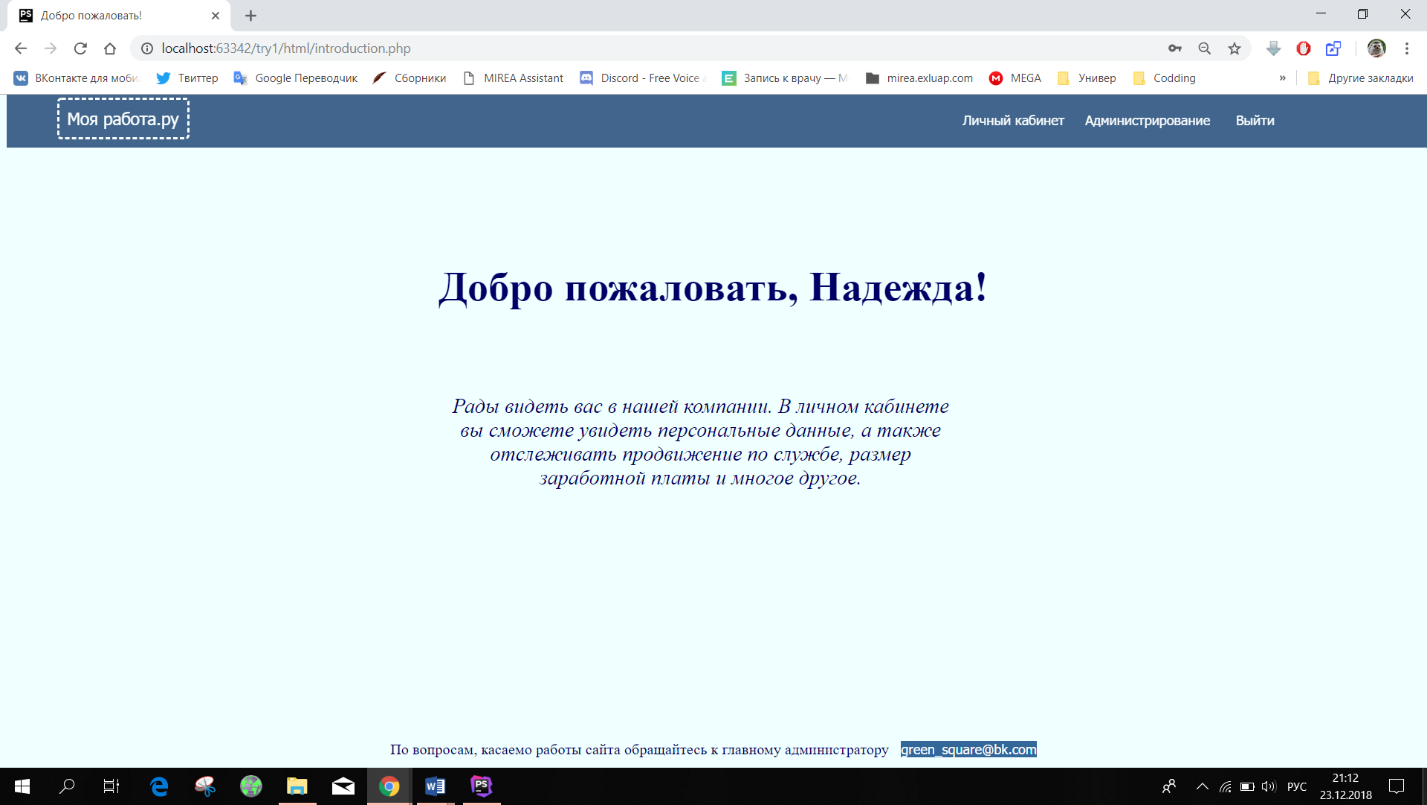
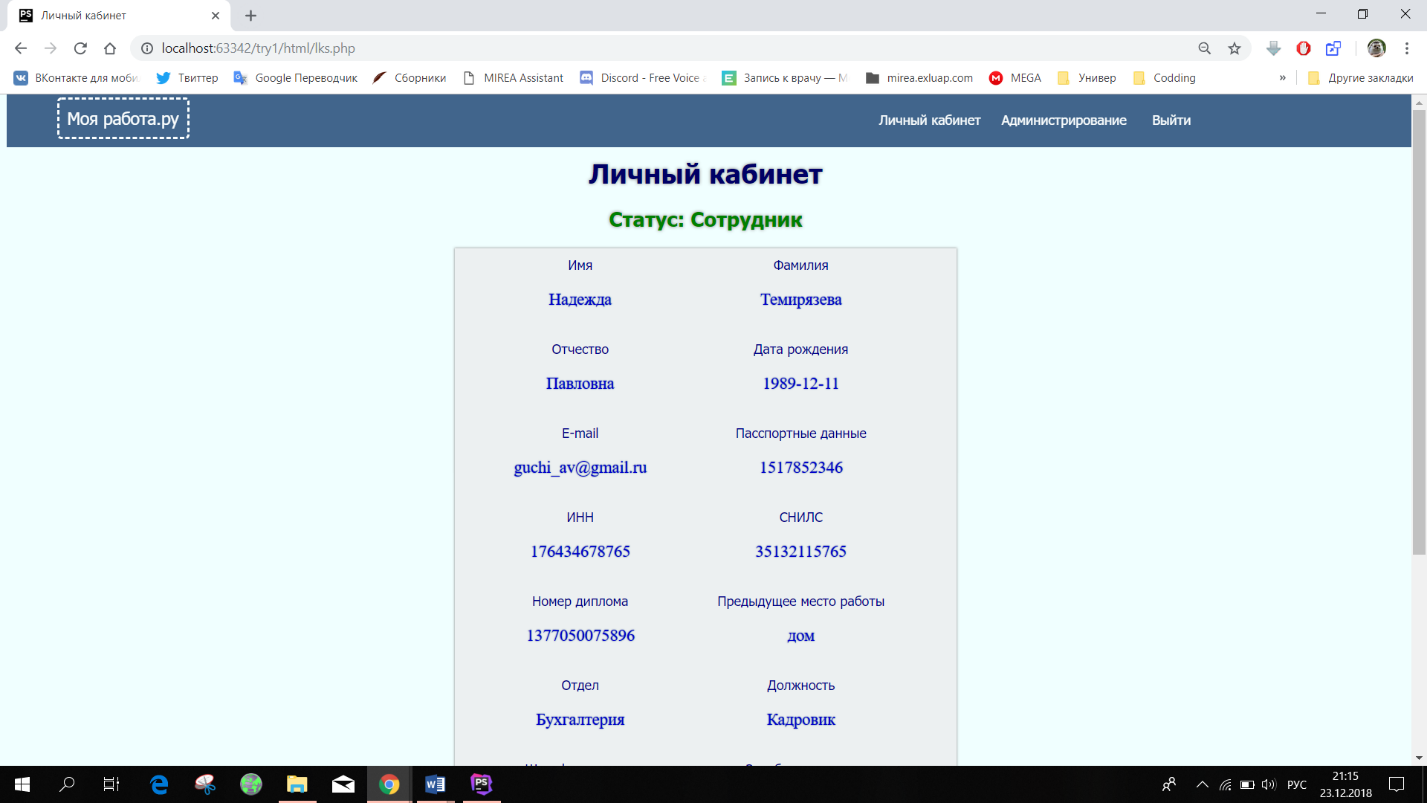
Главная страница встречает пользователя приветствием, а также краткой пояснительной запиской о возможностях системы.

Рисунок 15 - Главная страница сайта.

В личном кабинете пользователь может увидеть свои персональные данные, которые указал при регистрации, а также некоторую дополнительную информацию.

Рисунок 16 - Страница личного кабинета.

Данная программа предусматривает несколько видов пользователей: простой пользователь, бухгалтер и администратор. Для двух последний предусмотрен расширенный функционал, доступный во вкладке «Администрирование».

Проверка на администратора:

<?php

include\_once ("bd\_connection.php");

include\_once ('session.php');

include\_once ('bd\_output.php');

// проверка на администратора

if ($row['admin'] == '1') { // если администратор

echo '<a href = "../html/admin.php">Администрирование</a>';

} else if(($row['otdel']=="Бухгалтерия") && ($row['status']!='в ожидании')){

echo '<a href = "../html/booker.php">Администрирование</a>'; // если бухгалтер

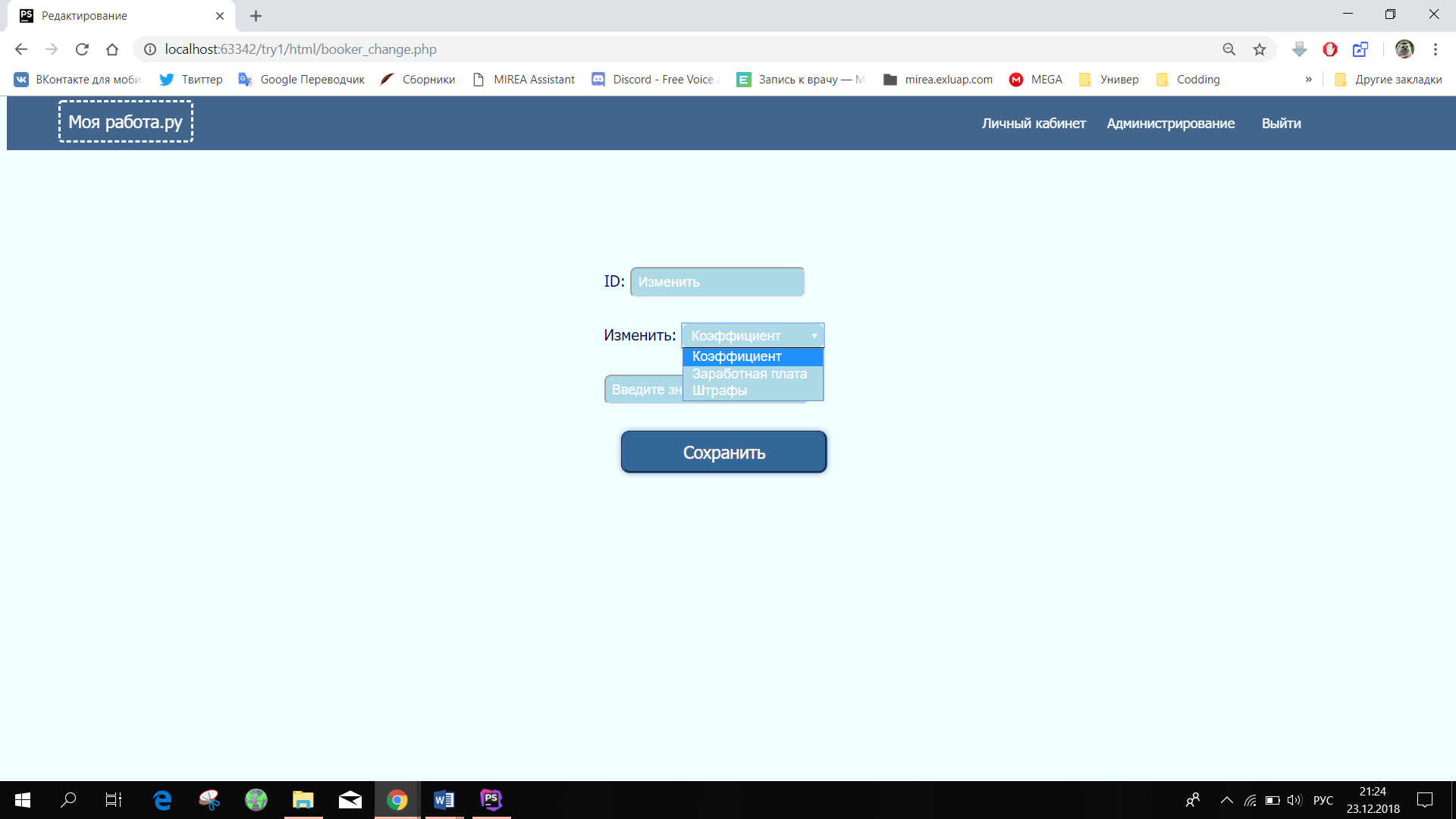
}else echo '<a href = "../html/admin\_error.html">Администрирование</a>'; // если обычный пользователь

?>

Если вход в систему выполнен в качестве бухгалтера, то ему будет доступны такие данные о сотрудниках компании, как: ФИО, статус, отдел, должность, коэффициент трудового участия, заработная плата и сумма штрафов. Для удобства работы с таблицей большого объема, предусмотрена сортировка по имени, зарплате, статусу, коэффициенту.

Рисунок 17 - Страница администрирования для бухгалтера.

Кроме того, бухгалтер имеет право изменять некоторые данные о сотрудниках.

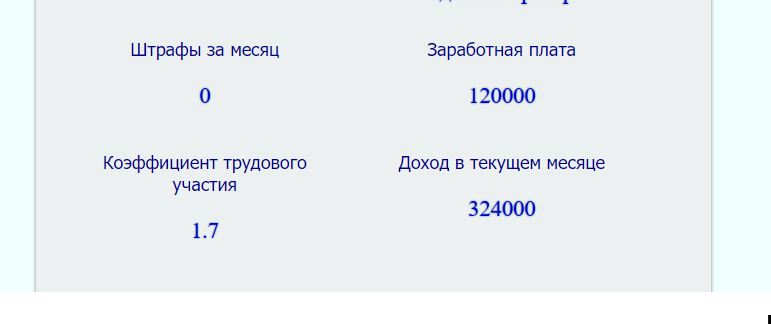
Рисунок 18 - Страница изменения данных для бухгалтера.

После того, как бухгалтер установил для сотрудника соответствующие данные, в личном кабинете автоматически высчитывается доход в текущем месяце.

<?php // расчёт дохода в текущем месяце

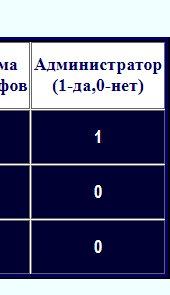
echo $row["pay"]+$row["coef"]\*$row["pay"]-$row["fine"]

?>

Рисунок 19 - Личный кабинет пользователя

Администратор, авторизованный в системе, имеет более расширенный функционал. Ему доступны для просмотра и изменения все персональные данные, о каждом из сотрудников.

Рисунок 20 - Страница администрирования для администратора.

Администратор вправе назначить любого сотрудника администратором, изменив значение определённого поля в таблице.

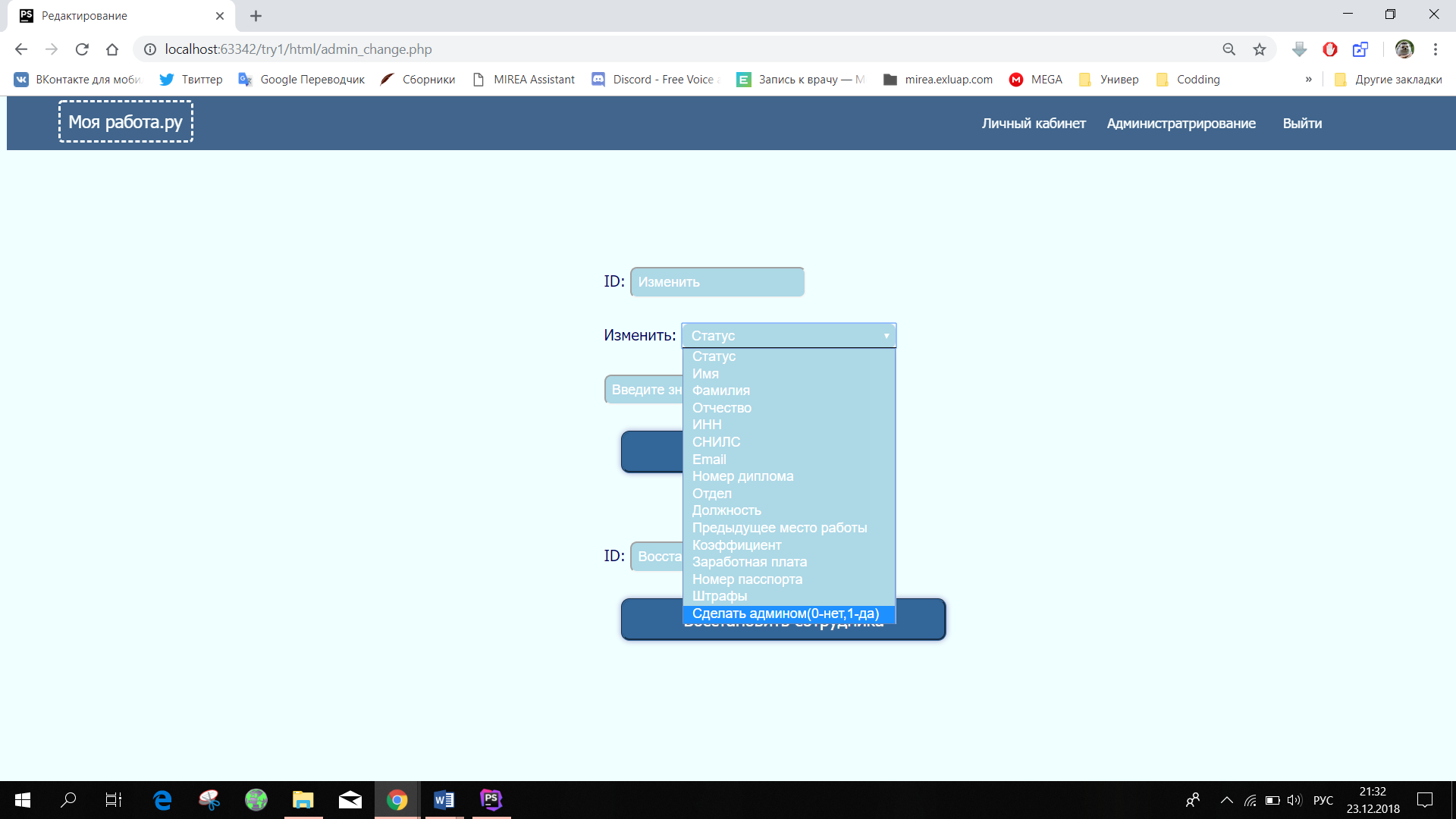
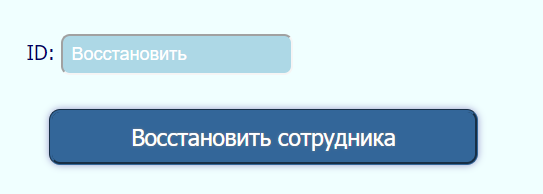


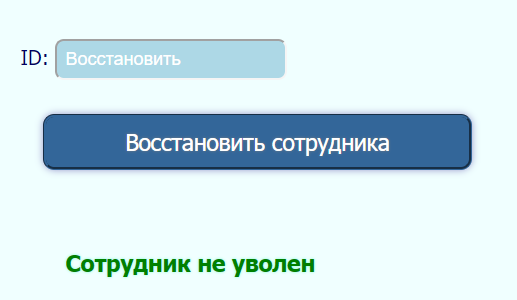
Рисунок 21 - Изменение поля «Администратор».

Если необходимо уволить какого-либо сотрудника, администратору нужно установить для него статус «Уволен», а затем нажать кнопку «фильтр», чтобы уволенный автоматически переместился из одной таблицы в другую.

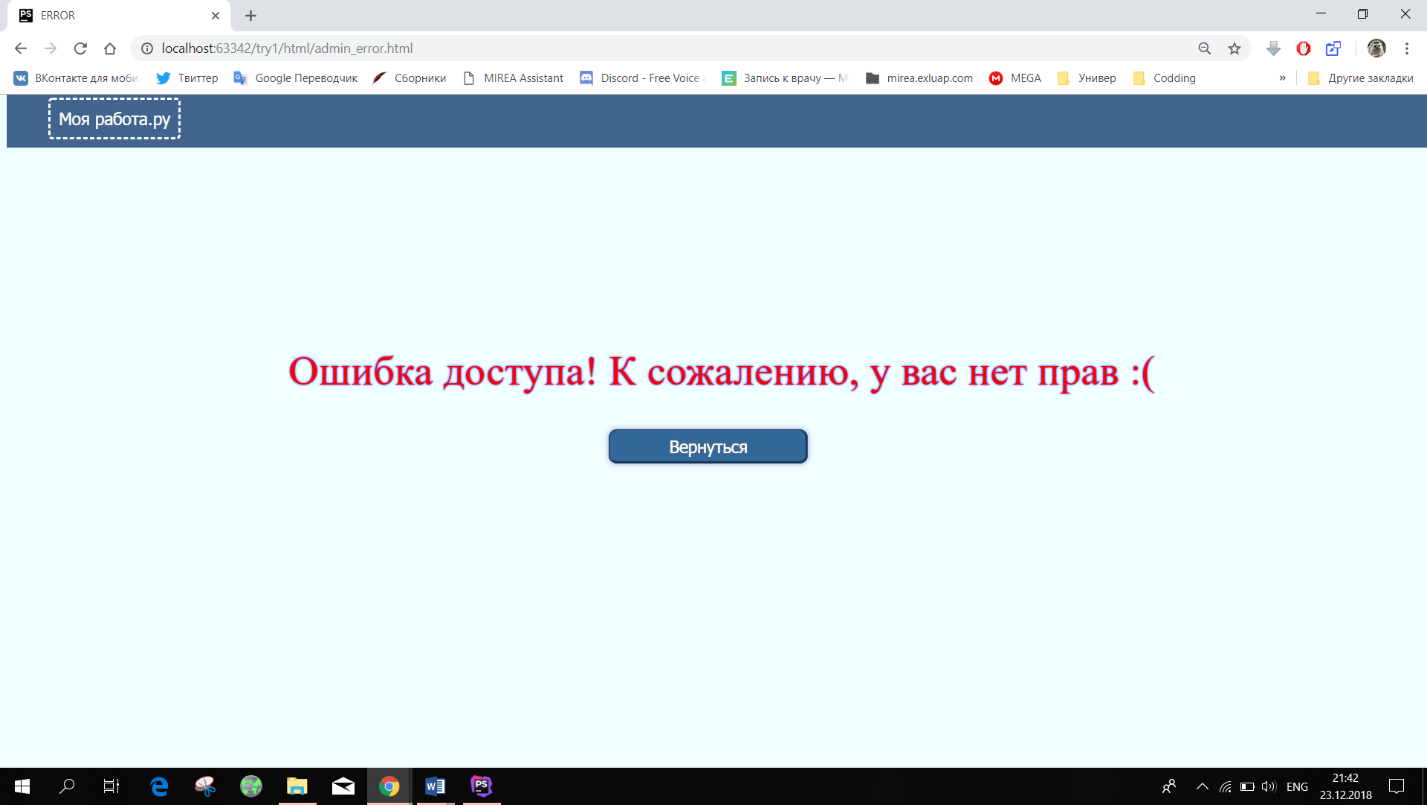
Для восстановления какого-либо уволенного пользователя, необходимо ввести его id и нажать кнопку «Восстановить». Затем нажать кнопку «фильтр», и сотрудник снова окажется в главной таблице.

Рисунок 22 - Восстановление сотрудника.

Предусмотрен случай того, что пользователь с конкретным id не уволен. Если администратор пытается восстановить не уволенного сотрудника, то система вернет ошибку.

Рисунок 23 - Ошибка восстановления сотрудника.

Страница администрирования недоступна простому пользователю. При попытке перехода на неё, возникнет ошибка.

Рисунок 24 - Ошибка доступа.

Выйти из системы можно, нажав кнопку «Выход» в верхнем меню. При этом, сессия завершается, и пользователь попадает на страницу авторизации.

## 2.3. Размещение системы на хостинге

Второй способ доступа к системе кадрового учёта - размещение всех необходимых для работы программы файлов на бесплатном хостинге (www.000webhost.com). После создания аккаунта на сайте был получен доступ к серверу (myworkru.000webhostapp.com). Для хранения файлов на сервере была создана пустая база данных, в которые позже были импортированы созданные ранее на локальном сервере базы данных. Размещение файлов на хостинг было произведено с файлового менеджера (files.000webhost.com).

# ЗАКЛЮЧЕНИЕ

В результате выполнения курсовой работы по дисциплине “Разработка программных приложений” на тему “Разработка системы отдела кадров” была выполнена установка и настройка операционной системы Windows 10 и необходимого программного обеспечения для разработанного программного продукта, произведена разработка и подробное описание структуры программного продукта, проведено тестирование (проверка) программы на локальном сервере, АРМ было размещено на хостинге в сети Интернет, а также составлена пояснительная записка к курсовому проекту.

# СПИСОК ЛИТЕРАТУРЫ

1) Документация PHP [Электронный ресурс] URL: http://php.net/docs.php (Дата обращения: 24.12.2018).

2) Алан Бьюли «Изучаем SQL». Санкт-Петербург-Москва, 2007 г.

3) Елена Верещагина Операционные системы. Учебно-методический комплекс. Проспект, 2015 г.

4) Фрайн Б. Responsive Web Design with HTML5 and CSS3 / HTML5 и CSS3 Разработка сайтов для любых браузеров и устройств. - СПб.: Питер, 2014 г.

# ПРИЛОЖЕНИЕ

Файл insert .php

<!DOCTYPE html>

<?php include\_once ('../php/session.php');?>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Моя работа.ру</title>

<link rel="stylesheet" type="text/css" href="../css/login.css">

</head>

<body>

<div class="zag">

<h2>Учет кадров</h2>

</div>

<div id="log\_in" >

<form method="get" >

<h1 class="login\_header">Вход</h1>

<div id="login\_form">

<input type="text" class="big\_text" name="email" id="email" value="" placeholder="введите email">

<p>

<input type="password" class="big\_text" name="pass" id="pass" value="" placeholder="введите пароль">

</p>

<p class="keeplogin">

<?php // вывод сообщения

if (!empty($message)) {

echo "<p class=\"error\">" . $message . "</p>";

}?>

</p>

<p>

<button name="login\_button" id="login\_button">Войти</button>

<a name="reg\_button" id="reg\_button" href="registration.php" style="" >Регистрация</a>

</p>

</div>

</form>

</div>

</body>

</html>

Файл login.css

#log\_in{  
 margin:0 0 150px 600px;  
 padding: 37px 30px;  
 width: 430px;  
 position: relative;  
 background: #fff;  
 border-radius:52px;  
 box-shadow: 0 1px 0 0 #d7d8db, 0 0 0 1px #e3e4e8;  
}  
  
.login\_header {  
 font-family: "Arial";  
 margin:0 0 30px 180px;  
}  
  
#login\_form {  
 margin: 20px 25px 0 50px;  
 width: 270px;  
}  
  
input.big\_text {  
 font-size: 14px;  
 padding: 6px 12px 8px;  
 box-sizing: border-box;  
 height: 35px;  
 width: 340px;  
 line-height: 270px;  
 border-radius: 3px;  
}  
  
button, a {  
 height: 35px;  
 max-width: 200px;  
 text-decoration: none;  
 font-family: Tahoma;  
 font-size: 20px;  
 padding: 5px 30px;  
 margin: 0px 10px;  
 text-align: left;  
 border: 3px solid whitesmoke;  
 border-radius: 10px;  
 border-color: #336699;  
 box-shadow: 0 0 10px #9999CC;  
 background-color: #336699;  
 color: white;  
 text-shadow: 0 0 3px grey;  
}  
  
#login\_button {  
 height: 42px;  
 cursor: pointer;  
}  
 #reg\_button{  
 width: 120px;  
 }  
 label{  
 font-size: 20px;  
  
 }  
 body{  
 background-size: cover;  
 background-image: url(http://ichef.bbci.co.uk/wwfeatures/wm/live/1280\_720/images/live/p0/4n/fy/p04nfyc7.jpg);  
 color: #153f6f;  
 }  
  
 P{  
 width: 700px;  
 }  
.zag{  
 margin-top: 190px;  
 font-family: "Times New Roman";  
 font-size: 32px;  
 margin-left: 730px;  
 font-style: oblique;  
 color: #153f6f;  
 text-shadow: 0 0 3px #9D60F9;  
 }

Файл session.php

<?php  
include\_once ('../php/bd\_connection.php');  
  
// начало сессии  
session\_start();  
  
// кнопка "Выход"  
if(isset($\_GET['exit'])){  
 header('Location: //localhost:63342/try1/html/insert.php');  
 session\_destroy();  
}  
  
$res\_all = $mysqli->query("SELECT \* FROM users");  
  
// авторизация  
if(isset($\_GET['login\_button'])) {  
 while ($sess = $res\_all->fetch\_assoc()) {  
 foreach ($sess as $key => $value) {  
  
 // проверка введённых данныхч  
 if ($\_GET['email'] == $sess['email'] && $\_GET['pass'] == $sess['password']) {  
 $\_SESSION['password']=$sess['password'];  
 header("Location:http://localhost:63342/try1/html/introduction.php");  
 break;  
 } elseif ($\_GET['pass'] != $sess['password']) {  
 $message = 'Ошибка! Данные введены неверно';  
 } else {  
 $message = 'Ошибка! Данные введены неверно';  
 }  
 }  
 }  
  
}  
?>

Файл registration.php

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Регистрация</title>  
 <link rel="stylesheet" type="text/css" href="../css/reg.css">  
  
</head>  
<body>  
<form class="" method="GET" >  
 <header>  
 <div class="container1">  
  
 <nav>  
 <ul class="menu">  
 <li>  
 <a class="logo" >Моя работа.ру</a>  
 </li>  
 <li>  
 <a href = "insert.php">Вход</a>  
 </li>  
 <li>  
 <a href = "#">Регистрация</a>  
 </li>  
 </ul>  
 </nav>  
 </div>  
 </header>  
 <div class="row main-form">  
<div class="container">  
 <div>  
 <h1>Регистрация</h1>  
 </div>  
 <p> <?php include\_once ("../php/bd\_input.php"); ?> </p>  
 <div class="left\_form">  
 <div class="form-group">  
 <label>Имя</label>  
 <div class="input-group">  
 <input type="text" required name="1name" class="form-control" placeholder="Введите имя"/>  
 </div>  
 <label class="form-control2">Дата рождения</label>  
 <div class="input-group">  
 <input required name="bday" type="date" class="form-control2"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label >Фамилия</label>  
 <div class="input-group">  
 <input type="text" required name="surname" class="form-control" placeholder="Введите фамилию"/>  
 </div>  
 <label class="form-control2">Серия и номер паспорта</label>  
 <div class="input-group">  
 <input required name="passport\_num" type="text" class="form-control2"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Отчество</label>  
 <div class="input-group">  
 <input type="text" required name="lastname" class="form-control" placeholder="Введите отчество"/>  
 </div>  
 <label class="form-control2">ИНН</label>  
 <div class="input-group">  
 <input id="inn" required name="inn" type="text" class="form-control2"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Email</label>  
 <div class="input-group">  
 <input type="text" id="e\_mail" class="form-control" required name="email" placeholder="Введите Email"/>  
  
 </div>  
 <label class="form-control2">СНИЛС</label>  
 <div class="input-group">  
 <input id="snils" required name="snils" type="text" class="form-control2"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Пароль</label>  
 <div class="input-group">  
 <input type="password" class="form-control" required name="password" placeholder="Введите пароль"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Подтверждение пароля</label>  
 <div class="input-group">  
 <input type="password" class="form-control" required name="check\_pas" placeholder="Введите пароль еще раз"/>  
 </div>  
  
 <label class="form-control2">Номер диплома</label>  
 <div class="input-group">  
 <input id="education" type="text" class="form-control2" required name="diplom"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Отдел</label>  
 <div class="input-group">  
 <select name="otdel">  
 <option value="Коммерческий">Коммерческий</option>  
 <option value="Юридический">Юридический</option>  
 <option value="Отдел кадров">Отдел кадров</option>  
 <option value="Финансовый">Финансовый</option>  
 <option value="Бухгалтерия">Бухгалтерия</option>  
 <option value="Информационный">Информационный</option>  
 </select>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label class="form-control2">Должность</label>  
 <div class="input-group">  
 <input type="text" required name="post" class="form-control2" placeholder="Введите должность"/>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <label>Предыдущее место работы</label>  
 <div class="input-group">  
 <input type="text" required name="lastpost" class="form-control" placeholder="Введите название"/>  
 </div>  
 </div>  
 </div>  
  
 <div class="form-group">  
  
 <input class="button" id="button" name="but" type="submit" value="Зарегистрироваться">  
  
 </div>  
  
</div>  
 </div>  
  
</form>  
</body>  
</html>

Файл reg.css

.form-group{  
 margin-bottom: 15px;  
 margin-top: 20px;  
}  
  
input, select{  
 font-size: 14px;  
}  
.form-control {  
 padding: 8px 30px !important;  
 margin-left: 30px;  
}  
  
.form-control2 {  
 padding: 8px 30px !important;  
 margin-left: 400px;  
}  
  
header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
li,a {  
 text-decoration: none;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
}  
  
a.logo{  
 margin-right: 1050px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
 display: inline;  
}  
  
  
  
#button {  
 text-decoration: none;  
 border: 3px solid whitesmoke;  
 margin-top: 28px;  
 margin-left: 30px;  
 padding: 8px 40px;  
 color: #666;  
 text-shadow: 0 1px #fff;  
 cursor: pointer;  
 border-radius: 8px;  
 background: #f5f5f5;  
 font-size: 18px;  
 box-shadow: 0 0 15px blue;  
}  
.main-form{  
 margin-top: 30px;  
 margin: 0 auto;  
 max-height: 1600px;  
 max-width: 700px;  
 padding: 10px 40px;  
 background-color: dimgray;  
 color: #FFF;  
 text-shadow: 0 0 3px blue;  
 border: 5px ridge lightgray;  
 box-shadow: 0 0 5px blue;  
 font-size: 18px;  
}  
.input-group {  
 color: #009edf;  
 font-size: 16px;  
}  
.login-button{  
 margin-top: 5px;  
}  
label {  
 margin-left: 35px;  
}  
select{  
 margin-left: 30px;  
 padding: 10px 60px;  
 font-family: Arial;  
 color: #666;  
 background: #f5f5f5;  
}  
  
form{  
 margin-top: 100px;  
}  
  
.left\_form{  
 max-width: 1000px;  
}  
h1{  
 margin-left: 30px;  
 font-family: "Times New Roman";  
 margin-bottom: 40px;  
 text-shadow: 0 0 8px blue;  
}  
body{  
 background-color: azure;  
}  
  
p{  
 font-size: 20px;  
 color: red;  
 margin-left: 35px;  
}

Файл check\_point.php

<?php  
  
// проверка e-mail  
function mail1($mail1){  
 if (!filter\_var($mail1, FILTER\_VALIDATE\_EMAIL)) {  
  
 return false;  
 } else return true;  
}  
  
// проверка введенных данных на присутствие только цифр  
function numbers ($num){  
 if (!ctype\_digit ($num)){  
 return false;  
 } else return true;  
}  
  
// проверка на непустоту  
function ne\_nul($x){  
 if($x!=NULL){  
 return true;  
 } else return false;  
}  
  
?>

Файл bd\_input.php

<?php  
  
include\_once ("bd\_connection.php");  
include\_once ("check\_point.php");  
  
// заненсение информации в базу данных из формы  
  
if(isset($\_GET['1name'])) {  
 $bday = trim($\_GET['bday']);  
 $passport = ($\_GET['passport\_num']);  
 $lastname = ($\_GET['lastname']);  
 $inn = ($\_GET['inn']);  
 $snils = ($\_GET['snils']);  
 $diplom = ($\_GET['diplom']);  
 $otdel = trim($\_GET['otdel']);  
 $surname = trim($\_GET['surname']);  
 $post = ($\_GET['post']);  
 $lastpost = ($\_GET['lastpost']);  
 $email = trim($\_GET['email']);  
 $name = trim($\_GET['1name']);  
 $password = $\_GET['password'];  
 $password2 = $\_GET['check\_pas'];  
  
 // проверка ввода  
 if (numbers($passport)&&numbers($inn)&&numbers($snils)&&numbers($diplom)) {  
  
 // проверка совпадения пароля  
 if ($password == $password2) {  
  
 // проверка e-mail  
 if (mail1($email)) {  
  
 // создание новой записи в БД  
 $query = $mysqli->query("INSERT INTO `users`(name1,surname,lastname,bday,inn,snils,email,password,diplom,otdel,post,lastpost,passport)  
 VALUES ('" . $name . "','" . $surname . "','" . $lastname . "','" . $bday . "','" . $inn . "','" . $snils . "','" . $email . "','" . $password . "','" . $diplom . "','" . $otdel . "','" . $post . "','" . $lastpost . "','" . $passport . "')");  
  
 if ($query) {  
 $message = "Запись успешно добавлена! Можете авторизоваться.";  
 } else {  
 $message = "Запись не добавлена.";  
 }  
 } else {  
 $message = "Ошибка ввода e-mail";  
 }  
 } else {  
 $message = "Пароли не совпадают!";  
 }  
 } else {  
 $message = "Ошибка заполнения формы!";  
 }  
}  
  
// вывод сообщения  
if (!empty($message)) {  
 echo "<p class=\"error\">" . $message . "</p>";  
}  
?>

Файл introduction.php

<!DOCTYPE html>  
<?php include\_once ("../php/session.php");  
 include\_once ("../php/bd\_output.php")?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Добро пожаловать!</title>  
 <link rel="stylesheet" type="text/css" href="../css/introduction.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
 <nav>  
 <ul class="menu">  
 <li>  
 <a class="logo" href = "#">Моя работа.ру</a>  
 </li>  
 <li>  
 <a href = "lks.php">Личный кабинет</a>  
 </li>  
 <li>  
 <?php include\_once("../php/admin\_error.php"); ?>  
 <!-- <a href = "admin.php">Администратор</a> -->  
 </li>  
 <li>  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
 <div class="main\_container">  
 <h1>Добро пожаловать, <?php echo $row['name1'] ?>! </h1>  
 <p><i>Рады видеть вас в нашей компании. В личном кабинете вы сможете увидеть персональные данные, а также отслеживать продвижение по службе, размер заработной платы и многое другое.</i></p>  
 <div class="info">По вопросам, касаемо работы сайта обращайтесь к главному администратору <a class="mail">green\_square@bk.com</a> </div>  
 </div>  
</form>  
</body>  
</html>

Файл introduction.css

header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
li,a, input {  
 text-decoration: none;  
 background-color: #153f6f;  
 border-color: #153f6f;  
 border: none;  
 font-size: 16px;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 cursor: pointer;  
}  
  
a.logo{  
 margin-right: 900px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
}  
  
body{  
 background-color: azure;  
}  
  
.main\_container{  
 margin: 200px 300px 0 300px;  
 text-align: center;  
 font-size: 25px;  
 color: #000066;  
  
}  
  
.info{  
 margin-top: 300px;  
 font-size: 18px;  
}  
.mail{  
 background-color: #336699;  
}  
P{  
 margin-left: 230px;  
 max-width:600px;  
 margin-top: 100px;  
}

Файл bd\_output.php

<?php  
include\_once ("bd\_connection.php");  
include\_once ("session.php");  
  
// вывод информации об авторизованном пользователе из базы данных  
  
if (isset($\_SESSION['password'])) {  
 while ($rows = $res\_all->fetch\_assoc()) {  
 if ($\_SESSION['password'] == $rows['password']) {  
 $row = $rows;  
 }  
 }  
}  
  
?>

Файл bd\_connection.php

<?php  
  
// подключение к базе данных  
  
$mysqli = new mysqli('localhost','root','',"workers");  
if ($mysqli->connect\_errno) {  
 echo "Не удалось подключиться к базе данных: (" . $mysqli->connect\_errno . ") " . $mysqli->connect\_error;  
}  
  
// изменение набора символов на utf8  
if (!$mysqli->set\_charset("utf8")) {  
 printf("Ошибка при загрузке набора символов utf8: %s\n", $mysqli->error);  
 exit();  
}  
?>

Файл lks.php

<!DOCTYPE html>  
<?php include\_once ('../php/session.php');?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Личный кабинет</title>  
 <link rel="stylesheet" type="text/css" href="../css/lks.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
  
 <nav>  
 <ul class="menu">  
 <li>  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
 <li>  
 <a href = "#">Личный кабинет</a>  
 </li>  
 <li>  
 <?php include\_once("../php/admin\_error.php"); ?>  
 <!-- <a href = "admin.php">Администратор</a> -->  
 </li>  
 <li>  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
 </ul>  
 </nav>  
 </div>  
 </header>  
 <div class="main\_container">  
 <div>  
 <h1>Личный кабинет</h1>  
 <?php include\_once ("../php/bd\_output.php")?>  
 <h2>Статус: <?php echo $row["status"]?></h2>  
 <div class="catalog">  
 <div class="item">Имя  
 <p><?php echo $row["name1"]?></p>  
 </div>  
  
 <div class="item">Фамилия  
 <p><?php echo $row["surname"]?></p>  
 </div>  
 <div class="item">Отчество  
 <p><?php echo $row["lastname"]?></p>  
 </div>  
 <div class="item">Дата рождения  
 <p><?php echo $row["bday"]?></p>  
 </div>  
 <div class="item">E-mail  
 <p><?php echo $row["email"]?></p>  
 </div>  
 <div class="item">Пасспортные данные  
 <p><?php echo $row["passport"]?></p>  
 </div>  
 <div class="item">ИНН  
 <p><?php echo $row["inn"]?></p>  
 </div>  
 <div class="item">СНИЛС  
 <p><?php echo $row["snils"]?></p>  
 </div>  
 <div class="item">Номер диплома  
 <p><?php echo $row["diplom"]?></p>  
 </div>  
 <div class="item">Предыдущее место работы  
 <p><?php echo $row["lastpost"]?></p>  
 </div>  
 <div class="item">Отдел  
 <p><?php echo $row["otdel"]?></p>  
 </div>  
 <div class="item">Должность  
 <p><?php echo $row["post"]?></p>  
 </div>  
 <div class="item">Штрафы за месяц  
 <p><?php echo $row["fine"]?></p>  
 </div>  
 <div class="item">Заработная плата  
 <p><?php echo $row["pay"]?></p>  
 </div>  
 <div class="item">Коэффициент трудового участия  
 <p><?php echo $row["coef"]?></p>  
 </div>  
 <div class="item">Доход в текущем месяце  
 <p><?php // расчёт дохода в текущем месяце  
 echo $row["pay"]+$row["coef"]\*$row["pay"]-$row["fine"] ?></p>  
 </div>  
  
 </div>  
 </div>  
 </div>  
</form>  
</body>  
</html>

Файл lks.css

.main\_container {  
 width: 1000px;  
 margin: 75px auto;  
}  
.catalog {  
 width: 600px;  
 height: 900px;  
 background-color: #ecf0f1;  
 box-shadow: 0 0 3px #999999;  
 margin-left: 200px  
  
}  
.item {  
 display: inline-block;  
 vertical-align: top;  
 width: 200px;  
 min-height:90px;  
 margin-top: 10px;  
 margin-left: 50px;  
 margin-right: 10px;  
 color: darkblue;  
 text-shadow: darkgrey;  
 text-align: center;  
 font-family: Tahoma;  
  
}  
  
h1,h2{  
 text-align: center;  
 font-family: Tahoma;  
 color: #000066;  
 text-shadow: 0 0 5px #8A8787;  
 }  
h2{  
 color: green;  
}  
  
body{  
 background-color: azure;  
}  
  
header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
li,a, input {  
 text-decoration: none;  
 background-color: #153f6f;  
 border-color: #153f6f;  
 border: none;  
 font-size: 16px;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 cursor: pointer;  
}  
  
a.logo{  
 margin-right: 800px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
}  
  
p{  
 font-family: "Times New Roman";  
 font-size: 20px;  
 color: mediumblue;  
 text-shadow:0 0 1px #153f6f;  
}

Файл admin\_error.php

<?php  
include\_once ("bd\_connection.php");  
include\_once ('session.php');  
include\_once ('bd\_output.php');  
  
// проверка на администратора  
  
 if ($row['admin'] == '1') { // если администратор  
 echo '<a href = "../html/admin.php">Администрирование</a>';  
 } else if(($row['otdel']=="Бухгалтерия") && ($row['status']!='в ожидании')){  
 echo '<a href = "../html/booker.php">Администрирование</a>'; // если бухгалтер  
 }else echo '<a href = "../html/admin\_error.html">Администрирование</a>'; // если обычный пользователь  
  
?>

Файл admin\_error.html

<!DOCTYPE html>  
<html lang="en">  
<link>  
 <meta charset="UTF-8">  
 <title>ERROR</title>  
 <link rel="stylesheet" type="text/css" href="../css/error.css">  
</head>  
<body>  
<form>  
 <header>  
 <div class="container">  
  
 <nav>  
 <ul class="menu">  
 <li>  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
  
 <div class="txt">  
 Ошибка доступа! К сожалению, у вас нет прав :(  
 </div>  
 <p>  
 <a class="return" href="lks.php">Вернуться</a>  
 </p>  
</form>  
</body>  
</html>

Файл error.css

body{  
 background-color: azure;  
}  
  
header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
  
.logo{  
 margin-right: 900px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
 text-decoration: none;  
 color: white;  
}  
  
.return{  
  
 text-decoration: none;  
 font-family: Tahoma;  
 font-size: 20px;  
 padding: 5px 70px;  
 margin: 0px 20px;  
 text-align: left;  
 border:3px groove whitesmoke;  
 border-radius: 10px;  
 border-color: #336699;  
 box-shadow: 0 0 10px #9999CC;  
 background-color: #336699;  
 color: white;  
 text-shadow: 0 0 3px grey;  
}  
  
p{  
 margin: 50px 700px;  
}  
.txt{  
 margin: 300px 0 0 20px;  
 text-align: center;  
 font-size: 50px;  
 text-shadow: 0 0 3px blue;  
 color: red;  
}  
  
li {  
 text-decoration: none;  
 background-color: #153f6f;  
 border-color: #153f6f;  
 border: none;  
 font-size: 16px;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
}

Файл booker.php

<!DOCTYPE html>  
<?php include\_once ('../php/session.php');?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Страница бухгалтера</title>  
 <link rel="stylesheet" type="text/css" href="../css/admin.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
 <nav>  
 <ul class="menu">  
 <li class="up">  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "lks.php">Личный кабинет</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "#">Администрирование</a>  
 </li>  
 <li class="up">  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
  
 <div class="buttons">  
 <a class="button" href="booker\_change.php">Изменить</a>  
 <input class="button" name="but0" type="submit" value="Показать">  
 <p> Сортировать по:</p>  
 <input class="button" name="but1" type="submit" value="имени">  
 <input class="button" name="but2" type="submit" value="зарплате">  
 <input class="button" name="but3" type="submit" value="статусу">  
 <input class="button" name="but4" type="submit" value="коэффициенту">  
  
 </div>  
  
 <div class="BD">  
 <table>  
 <tr>  
 <th class="headers">ID</th>  
 <th class="headers">Статус</th>  
 <th class="headers">Имя</th>  
 <th class="headers">Фамилия</th>  
 <th class="headers">Отчество</th>  
 <th class="headers">Отдел</th>  
 <th class="headers">Должность</th>  
 <th class="headers">Коэффициент трудового участия</th>  
 <th class="headers">Заработная плата</th>  
 <th class="headers">Сумма штрафов</th>  
  
 </tr>  
 <?php include\_once("../php/bd\_half\_output.php"); ?>  
  
 </table>  
 </div>  
  
</form>  
</body>  
</html>

Файл bd\_half\_output.php

<?php  
include\_once ("bd\_connection.php");  
include\_once ("bd\_output.php");  
  
// сортировка таблицы на странице Администрирование для бухгалтера  
$ask\_name=$mysqli->query("SELECT \* FROM users ORDER BY name1");  
$ask\_otdel=$mysqli->query("SELECT \* FROM users ORDER BY pay");  
$ask\_status=$mysqli->query("SELECT \* FROM users ORDER BY status");  
$ask\_c = $mysqli->query("SELECT \* FROM users ORDER BY coef");  
$res\_all = $mysqli->query("SELECT \* FROM users ORDER BY id");  
  
  
if(isset($\_GET['but1'])){  
  
 // сортировка по имени  
 while($row\_all=$ask\_name->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
   
 </tr>';  
 }  
} else if(isset($\_GET['but2'])) {  
  
 // сортировка по зарплате  
 while($row\_all = $ask\_otdel->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
   
 </tr>';  
 }  
} else if(isset($\_GET['but3'])) {  
  
 // сортировка по статусу  
 while($row\_all = $ask\_status->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>   
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
   
 </tr>';  
 }  
} else if(isset($\_GET['but4'])) {  
  
 // сортировка по коэффициенту  
 while($row\_all = $ask\_c->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
   
 </tr>';  
 }  
} else if(isset($\_GET['but0'])) {  
  
 // вывод таблицы  
 while ($row\_all = $res\_all->fetch\_assoc()) {  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
   
 </tr>';  
  
 }  
}  
?>

Файл booker\_change.php

<!DOCTYPE html>  
<?php include\_once ('../php/session.php');?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Редактирование</title>  
 <link rel="stylesheet" type="text/css" href="../css/admin\_change.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
  
 <nav>  
 <ul class="menu">  
 <li class="up">  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "lks.php">Личный кабинет</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "booker.php">Администрирование</a>  
 </li>  
 <li class="up">  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
  
 <div class="form">  
  
 <div class="input-group">  
 <label>ID:</label>  
 <input type="text" name="id" class="form-control" placeholder="Изменить"/>  
 </div>  
  
 <div class="input-group">  
 <label>Изменить:</label>  
 <select name="value">  
 <option value="coef">Коэффициент</option>  
 <option value="pay">Заработная плата</option>  
 <option value="fine">Штрафы</option>  
  
 </select>  
 </div>  
  
 <div class="input-group">  
 <input type="text" name="new\_val" class="form-control1" placeholder="Введите значение"/>  
 </div>  
  
  
 <input class="button" id="button" name="but" type="submit" value="Сохранить">  
 <p><b> <?php include\_once ("../php/bd\_change.php"); ?></b> </p>  
 </div>  
</form>  
</body>  
</html>

Файл bd\_change.php

<?php

include\_once ("bd\_connection.php");

include\_once ("check\_point.php");

if(isset($\_GET['id'])&& isset($\_GET['new\_val'])&&isset($\_GET['but'])) {

$id = $\_GET['id']; // id пользователя

$value = $\_GET['value']; // поле для изменения

$new\_val=$\_GET['new\_val']; // новое значение

// изменяем соответствующее поле

switch($value){

case "name1": $ask= $mysqli->query("UPDATE users SET name1='$new\_val' WHERE id=".$id);break;

case "surname": $ask= $mysqli->query("UPDATE users SET surname='$new\_val' WHERE id=".$id);break;

case "lastname": $ask= $mysqli->query("UPDATE users SET lastname='$new\_val' WHERE id=".$id);break;

case "inn": $ask= $mysqli->query("UPDATE users SET inn='$new\_val' WHERE id=".$id);break;

case "snils": $ask= $mysqli->query("UPDATE users SET snils='$new\_val' WHERE id=".$id);break;

case "email": $ask= $mysqli->query("UPDATE users SET email='$new\_val' WHERE id=".$id);break;

case "diplom": $ask= $mysqli->query("UPDATE users SET diplom='$new\_val' WHERE id=".$id);break;

case "otdel": $ask= $mysqli->query("UPDATE users SET otdel='$new\_val' WHERE id=".$id);break;

case "post": $ask= $mysqli->query("UPDATE users SET post='$new\_val' WHERE id=".$id);break;

case "lastpost": $ask= $mysqli->query("UPDATE users SET lastpost='$new\_val' WHERE id=".$id);break;

case "status": $ask= $mysqli->query("UPDATE users SET status='$new\_val' WHERE id=".$id);break;

case "coef": $ask= $mysqli->query("UPDATE users SET coef='$new\_val' WHERE id=".$id);break;

case "pay": $ask= $mysqli->query("UPDATE users SET pay='$new\_val' WHERE id=".$id);break;

case "passport": $ask= $mysqli->query("UPDATE users SET passport='$new\_val' WHERE id=".$id);break;

case "fine": $ask= $mysqli->query("UPDATE users SET fine='$new\_val' WHERE id=".$id);break;

case "admin": $ask= $mysqli->query("UPDATE users SET admin='$new\_val' WHERE id=".$id);break;

default: $message="No match!"; break;

}

$message = "Успешно изменено.";

}

// вывод сообщения

if (!empty($message)) {

echo "<p class=\"error\">" . $message . "</p>";

}?>

Файл admin.php

<!DOCTYPE html>  
<?php include\_once ('../php/session.php');?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Страница администратора</title>  
 <link rel="stylesheet" type="text/css" href="../css/admin.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
 <nav>  
 <ul class="menu">  
 <li class="up">  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "lks.php">Личный кабинет</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "#">Администратрирование</a>  
 </li>  
 <li class="up">  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
  
 <div class="buttons">  
 <a class="button" href="admin\_change.php">Изменить</a>  
 <input class="button" name="but0" type="submit" value="Показать">  
 <input class="button" name="fired\_but" type="submit" value="Показать уволенных">  
 <p> Сортировать по:</p>  
 <input class="button" name="but1" type="submit" value="имени">  
 <input class="button" name="but2" type="submit" value="отделу">  
 <input class="button" name="but3" type="submit" value="статусу">  
 <p> Отфильтровать уволенных сотрудников:</p>  
 <input class="button" name="del\_but" type="submit" value="фильтр">  
  
  
 </div>  
  
 <div class="BD">  
 <table>  
 <tr>  
 <th class="headers">ID</th>  
 <th class="headers">Статус</th>  
 <th class="headers">Имя</th>  
 <th class="headers">Фамилия</th>  
 <th class="headers">Отчество</th>  
 <th class="headers">Дата рождения</th>  
 <th class="headers">Номер паспорта</th>  
 <th class="headers">ИНН</th>  
 <th class="headers">СНИЛС</th>  
 <th class="headers">Email</th>  
 <th class="headers">Пароль</th>  
 <th class="headers">Номер диплома</th>  
 <th class="headers">Отдел</th>  
 <th class="headers">Должность</th>  
 <th class="headers">Предыдущее место работы</th>  
 <th class="headers">Коэффициент трудового участия</th>  
 <th class="headers">Заработная плата</th>  
 <th class="headers">Сумма штрафов</th>  
 <th class="headers">Администратор  
 (1-да,0-нет)</th>  
 </tr>  
 <?php include\_once("../php/bd\_output.php");  
 include\_once ("../php/bd\_sort.php");  
 include\_once ("../php/bd\_del.php");  
 ?>  
  
 </table>  
 </div>  
  
</form>  
</body>  
</html>

Файл admin.css

body{  
 background-color: azure;  
}  
  
header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
.up {  
 text-decoration: none;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 background-color: #153f6f;  
 border: none;  
 font-size: 16px;  
 cursor: pointer;  
}  
  
input.up{  
 background-color: #153f6f;  
}  
  
.logo{  
 margin-right: 900px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 cursor: pointer;  
 text-decoration: none;  
}  
  
table {  
 border: 4px ridge #000066;  
 margin: 40px 0px;  
 width: 100%;  
 background: white;  
 color: #000066;  
 border-spacing: 1px;  
}  
  
.headers {  
 border: 1px solid #666666;  
 background: white;  
}  
  
.tbl{  
 border: 1px solid #666666;  
 background-color: #000033;  
 color: white;  
 font-family: "Arial Narrow";  
 padding: 5px;  
  
}  
  
 .buttons{  
 margin-top: 120px;  
 margin-bottom: 0px;  
 font-family: Tahoma;  
 font-size: 25px;  
}  
 .button{  
 text-decoration: none;  
 font-family: Tahoma;  
 font-size: 20px;  
 padding: 5px 70px;  
 margin: 0px 20px;  
 text-align: left;  
 border:3px groove whitesmoke;  
 border-radius: 10px;  
 border-color: #336699;  
 box-shadow: 0 0 10px #9999CC;  
 background-color: #336699;  
 color: white;  
 text-shadow: 0 0 3px grey;  
 cursor: pointer;  
 }  
  
 p{  
 margin-left: 30px;  
 color: #000066;  
 }

Файл bd\_sort.php

<?php  
  
include\_once ("bd\_connection.php");  
include\_once ("bd\_output.php");  
  
// сортировка таблицы на странице Администрирование для администратора  
$ask\_name=$mysqli->query("SELECT \* FROM users ORDER BY name1");  
$ask\_otdel=$mysqli->query("SELECT \* FROM users ORDER BY otdel");  
$ask\_status=$mysqli->query("SELECT \* FROM users ORDER BY status");  
$ask\_fired=$mysqli->query("SELECT \* FROM fired");  
$res\_all = $mysqli->query("SELECT \* FROM users ORDER BY id");  
  
  
if(isset($\_GET['but1'])){  
  
 // сортировка по имени  
 while($row\_all=$ask\_name->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["bday"] . '</th>  
 <th class="tbl">' . $row\_all["passport"] . '</th>  
 <th class="tbl">' . $row\_all["inn"] . '</th>  
 <th class="tbl">' . $row\_all["snils"] . '</th>  
 <th class="tbl">' . $row\_all["email"] . '</th>  
 <th class="tbl">' . $row\_all["password"] . '</th>  
 <th class="tbl">' . $row\_all["diplom"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["lastpost"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
 <th class="tbl">' . $row\_all["admin"] . '</th>  
 </tr>';  
 }  
} else if(isset($\_GET['but2'])) {  
  
 // сортировка по отделу  
 while($row\_all = $ask\_otdel->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["bday"] . '</th>  
 <th class="tbl">' . $row\_all["passport"] . '</th>  
 <th class="tbl">' . $row\_all["inn"] . '</th>  
 <th class="tbl">' . $row\_all["snils"] . '</th>  
 <th class="tbl">' . $row\_all["email"] . '</th>  
 <th class="tbl">' . $row\_all["password"] . '</th>  
 <th class="tbl">' . $row\_all["diplom"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["lastpost"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
 <th class="tbl">' . $row\_all["admin"] . '</th>  
 </tr>';  
 }  
} else if(isset($\_GET['but3'])) {  
  
 // сортировка по статусу  
 while($row\_all = $ask\_status->fetch\_assoc()){  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["bday"] . '</th>  
 <th class="tbl">' . $row\_all["passport"] . '</th>  
 <th class="tbl">' . $row\_all["inn"] . '</th>  
 <th class="tbl">' . $row\_all["snils"] . '</th>  
 <th class="tbl">' . $row\_all["email"] . '</th>  
 <th class="tbl">' . $row\_all["password"] . '</th>  
 <th class="tbl">' . $row\_all["diplom"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["lastpost"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
 <th class="tbl">' . $row\_all["admin"] . '</th>  
 </tr>';  
 }  
} else if(isset($\_GET['but0'])) {  
  
 // вывод таблицы без сортировки  
 while ($row\_all = $res\_all->fetch\_assoc()) {  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["bday"] . '</th>  
 <th class="tbl">' . $row\_all["passport"] . '</th>  
 <th class="tbl">' . $row\_all["inn"] . '</th>  
 <th class="tbl">' . $row\_all["snils"] . '</th>  
 <th class="tbl">' . $row\_all["email"] . '</th>  
 <th class="tbl">' . $row\_all["password"] . '</th>  
 <th class="tbl">' . $row\_all["diplom"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["lastpost"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
 <th class="tbl">' . $row\_all["admin"] . '</th>  
 </tr>';  
  
 }  
}else if(isset($\_GET['fired\_but'])) {  
  
 // вывод таблицы уволенных сотрудников  
 while ($row\_all = $ask\_fired->fetch\_assoc()) {  
 echo '<tr class=info>  
 <th class="tbl">' . $row\_all["id"] . '</th>  
 <th class="tbl">' . $row\_all["status"] . '</th>  
 <th class="tbl">' . $row\_all["name1"] . '</th>  
 <th class="tbl">' . $row\_all["surname"] . '</th>  
 <th class="tbl">' . $row\_all["lastname"] . '</th>  
 <th class="tbl">' . $row\_all["bday"] . '</th>  
 <th class="tbl">' . $row\_all["passport"] . '</th>  
 <th class="tbl">' . $row\_all["inn"] . '</th>  
 <th class="tbl">' . $row\_all["snils"] . '</th>  
 <th class="tbl">' . $row\_all["email"] . '</th>  
 <th class="tbl">' . $row\_all["password"] . '</th>  
 <th class="tbl">' . $row\_all["diplom"] . '</th>  
 <th class="tbl">' . $row\_all["otdel"] . '</th>  
 <th class="tbl">' . $row\_all["post"] . '</th>  
 <th class="tbl">' . $row\_all["lastpost"] . '</th>  
 <th class="tbl">' . $row\_all["coef"] . '</th>  
 <th class="tbl">' . $row\_all["pay"] . '</th>  
 <th class="tbl">' . $row\_all["fine"] . '</th>  
 <th class="tbl">' . $row\_all["admin"] . '</th>  
 </tr>';  
 }  
}  
?>

Файл bd\_del.php

<?php  
include\_once ("bd\_connection.php");  
  
// восстановление уволенного сотрудника  
if(isset($\_GET['restore'])){  
 $roww=$mysqli->query("SELECT \* FROM fired");  
 $id2 = $\_GET['id2'];  
 while ($arr=$roww->fetch\_assoc()) {  
  
 // проверка на наличие сотруднека в таблице fired  
 if ($id2 == $arr['id']) {  
 $asking = $mysqli->query("UPDATE fired SET status='Сотрудник' WHERE id=" . $id2);  
 $mess="Успех";  
 }else $mess=NULL;  
 }  
 if(empty($mess)){  
 $mess="Сотрудник не уволен";  
 }  
  
 // вывод сообщения  
 if (!empty($mess)) {  
 echo "<p class=\"error\">" . $mess . "</p>";  
 $mess=NULL;  
 }  
}  
  
// фильтр уволенных сотрудников  
if(isset($\_GET['del\_but'])) {  
 $asking1 = $mysqli->query("INSERT INTO fired SELECT \* FROM users WHERE status='Уволен'");  
 $asking2=$mysqli->query("DELETE FROM users WHERE status='Уволен'");  
 $asking3=$mysqli->query("INSERT INTO users SELECT \* FROM fired WHERE status!='Уволен'");  
 $asking4=$mysqli->query("DELETE FROM fired WHERE status!='Уволен'");  
}  
  
?>

Файл admin\_change.php

<!DOCTYPE html>  
<?php include\_once ('../php/session.php');?>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Редактирование</title>  
 <link rel="stylesheet" type="text/css" href="../css/admin\_change.css">  
</head>  
<body>  
<form class="" method="GET">  
 <header>  
 <div class="container">  
  
 <nav>  
 <ul class="menu">  
 <li class="up">  
 <a class="logo" href = "introduction.php">Моя работа.ру</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "lks.php">Личный кабинет</a>  
 </li>  
 <li class="up">  
 <a class="up" href = "admin.php">Администратрирование</a>  
 </li>  
 <li class="up">  
 <input class="up" name="exit" type="submit" value="Выйти">  
 </li>  
  
 </ul>  
 </nav>  
 </div>  
 </header>  
  
 <div class="form">  
  
 <div class="input-group">  
 <label>ID:</label>  
 <input type="text" name="id" class="form-control" placeholder="Изменить"/>  
 </div>  
  
 <div class="input-group">  
 <label>Изменить:</label>  
 <select name="value">  
 <option value="status">Статус</option>  
 <option value="name1">Имя</option>  
 <option value="surname">Фамилия</option>  
 <option value="lastname">Отчество</option>  
 <option value="inn">ИНН</option>  
 <option value="snils">СНИЛС</option>  
 <option value="email">Email</option>  
 <option value="diplom">Номер диплома</option>  
 <option value="otdel">Отдел</option>  
 <option value="post">Должность</option>  
 <option value="lastpost">Предыдущее место работы</option>  
 <option value="coef">Коэффициент</option>  
 <option value="pay">Заработная плата</option>  
 <option value="passport">Номер пасспорта</option>  
 <option value="fine">Штрафы</option>  
 <option value="admin">Сделать админом(0-нет,1-да)</option>  
 </select>  
 </div>  
  
 <div class="input-group">  
 <input type="text" name="new\_val" class="form-control1" placeholder="Введите значение"/>  
 </div>  
  
 <input class="button" id="button" name="but" type="submit" value="Сохранить">  
 <p><b> <?php include\_once ("../php/bd\_change.php"); ?></b> </p>  
 <div class="input-group">  
 <label>ID:</label>  
 <input type="text" name="id2" class="form-control" placeholder="Восстановить"/>  
 </div>  
 <input class="button" id="button" name="restore" type="submit" value="Восстановить сотрудника">  
 <p><b> <?php include\_once ("../php/bd\_del.php"); ?></b> </p>  
 </div>  
</form>  
</body>  
</html>

Файл admin\_change.css

body{  
 background-color: azure;  
}  
  
header {  
 background-color: #153f6f;  
 padding-bottom: 3px;  
 height:60px;  
 position:fixed;  
 top: 0;  
 opacity: 0.8;  
 overflow:hidden;  
 width:100%;  
}  
.up {  
 text-decoration: none;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 background-color: #153f6f;  
 border: none;  
 font-size: 16px;  
}  
  
input.up{  
 background-color: #153f6f;  
 cursor: pointer;  
}  
  
.logo{  
 margin-right: 900px;  
 font-size: 20px;  
 border-radius: 5px;  
 border: dashed;  
 padding: 10px 10px;  
 color: white;  
 font-family: Tahoma;  
 display: inline;  
 max-width: 100%;  
 margin-left: 10px;  
 cursor: pointer;  
 text-decoration: none;  
}  
  
.form{  
 margin: 200px 0px 50px 700px;  
}  
  
.input-group{  
 margin:30px 0px;  
 width: 600px;  
 font-family: Tahoma;  
 font-size: 18px;  
 color: #000066;  
  
}  
  
input{  
 font-size: 16px;  
 background-color: lightblue;  
 height: 30px;  
 border-color: whitesmoke;  
 border-radius: 8px;  
 color: white;  
 padding-left: 7px;  
  
}  
  
::placeholder{  
 color: white;  
}  
  
:optional{  
 font-size: 16px;  
 background-color: lightblue;  
 height: 30px;  
 border-color: whitesmoke;  
 border-radius: 8px;  
 color: white;  
 padding-left: 7px;  
  
}  
  
.button{  
 height: 50px;  
 text-decoration: none;  
 font-family: Tahoma;  
 font-size: 20px;  
 padding: 5px 70px;  
 margin: 0px 0px 50px 20px;  
 text-align: left;  
 border:3px groove whitesmoke;  
 border-radius: 10px;  
 border-color: #336699;  
 box-shadow: 0 0 10px #9999CC;  
 background-color: #336699;  
 color: white;  
 text-shadow: 0 0 3px grey;  
 cursor: pointer;  
}  
p{  
 margin-left: 40px;  
 font-family: Tahoma;  
 font-size: 20px;  
 color: green;  
 text-shadow: 0 0 3px aquamarine;  
}  
.form-control1{  
 width: 230px;  
}