Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования   
**Национальный исследовательский университет «МЭИ»**

**Курсовая работа**

по дисциплине«**WEB-технологии**»

**РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ**

**ПО КУРСУ «WEB-ТЕХНОЛОГИИ»**

Выполнил:

студент группы А-12-19

Доржу Н.Ш.

Вариант 9

Москва  
2022

1. Краткое содержание работы

WEB-приложение база анкет соискателей, где пользователи могут как размещать свои объявления, так и выполнять чужие.

2.1. Часть 1

Установка и конфигурирование веб-сервера Apache в ОС Windows, подключение модуля PHP, СУБД MySQL и PhpMyAdmin.

2.1. Установка Apache

С сайта apachelounge.com/download был скачан архив, содержащий файлы Apache HTTP Server 2.4., содержимое архива было распаковано в директорию C:\Server\bin.

В конфигурационный файл httpd.conf были внесены следующие изменения:

1) Укажем каталог, в котором будут храниться все файлы веб-сервера:



2) Для локального веб-сервера установим имя сервера:



3) Установим путь до каталога, в котором хранятся все веб-документы:





4) Перечислим индексные файлы:



5) Разрешим использование файлы .htaccess:



6) Подключим модуль rewrite:

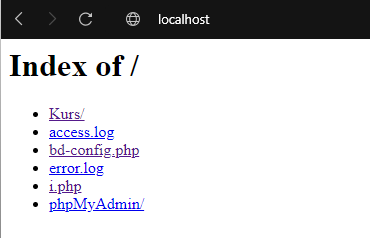


После завершения настройки сервера была произведена установка. Она запускается последовательным вводом следующих команд в Windows PowerShell:





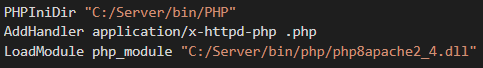
Проверим работу сервера. В адресной строке был введен адрес localhost. Открылась следующая страница:



*Рис.2.1. Подключение к серверу в браузере  
(Скриншот был сделан после полной установки всех компонентов)*

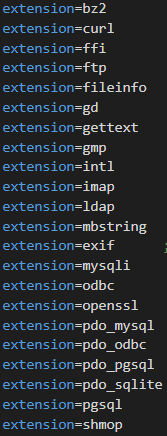
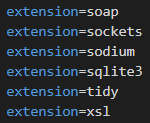
2.2. Подключение модуля PHP

С сайта https://windows.php.net/download/ был скачан архив, содержащий файлы PHP 8, содержимое архива было распаковано в директорию C:\Server\bin. В конфигурационный файл Apache httpd.conf были дописаны следующие строчки:

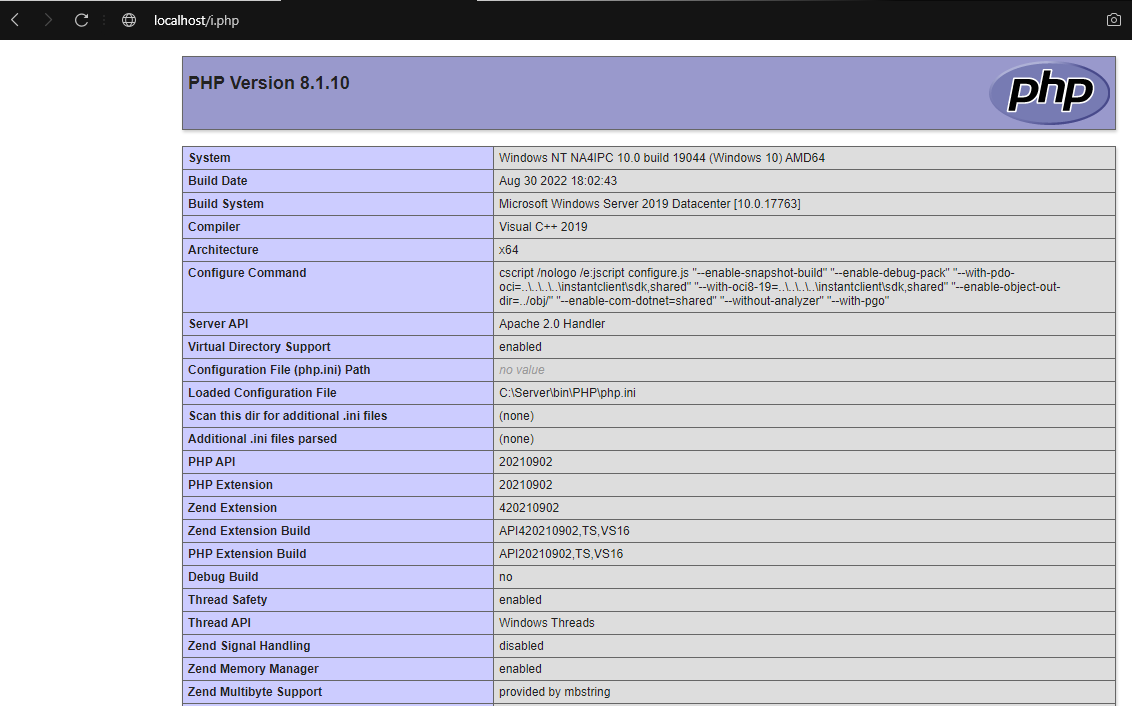


В конфигурационный файл php.ini были добавлены следующие строки:



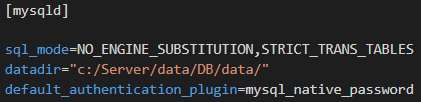
Проверим работу PHP. Был создан файл с PHP-функцией, выводящей информацию о PHP. Откроем его (localhost\i.php), открылась следующая страница:



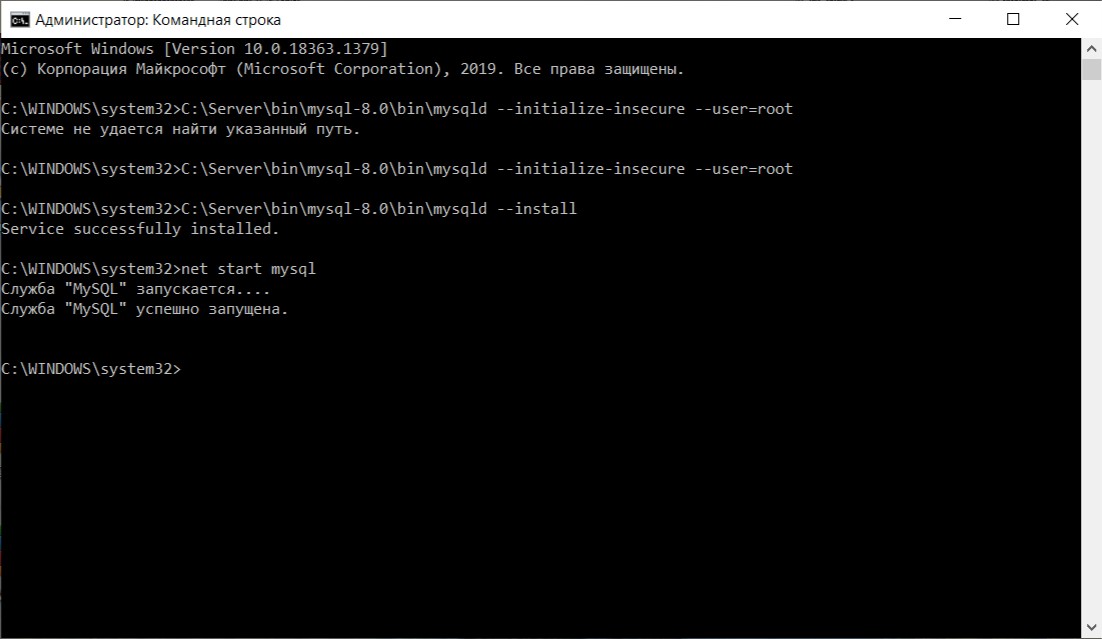
*Рис.2.2. Информация о PHP (i.php)*

2.3. Подключение модуля СУБД MySQL

С сайта https://dev.mysql.com/downloads/mysql/ был скачан архив, содержащий файлы MySQL 8.0, содержимое архива было распаковано в директорию C:\Server\bin. В конфигурационном файле my.ini были добавлены следующие строки:



Инициализация, установка и запуск:



*Рис 2.3. Установка и запуск MySQL*

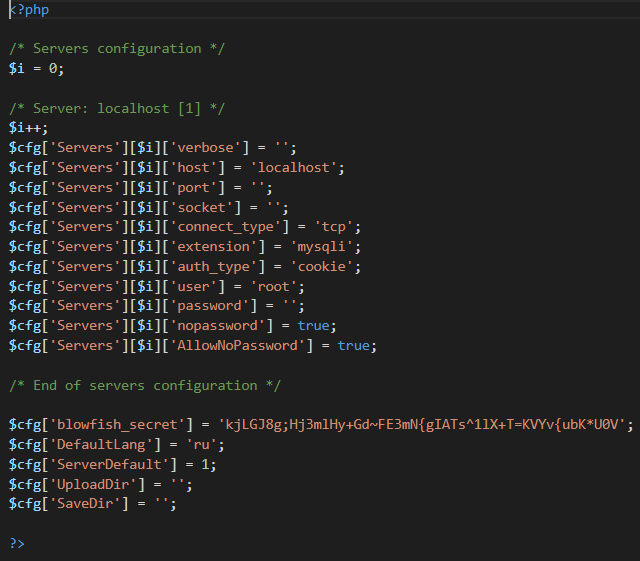
После установки в списке служб появилась служба MySQL.



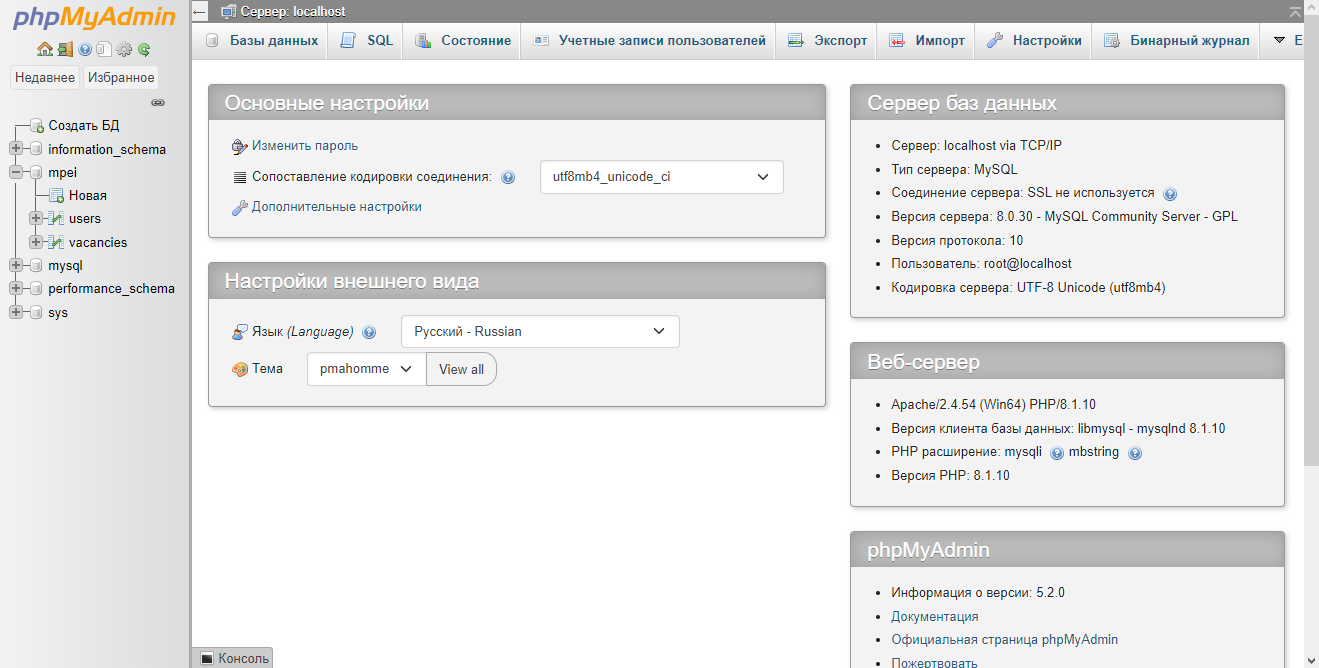
*Рис 2.4. MySQL в списке служб Windows*

2.4. Подключение PhpMyAdmin

С сайта phpmyadmin.net был скачан архив, содержащий файлы PhpMyAdmin, содержимое архива было распаковано в директорию C:\Server\data\htdocs. В конфигурационном файле config.inc.php были добавлены следующие строки:



При переходе по адресу localhost\phpmyadmin после авторизации появляется следующая страница:



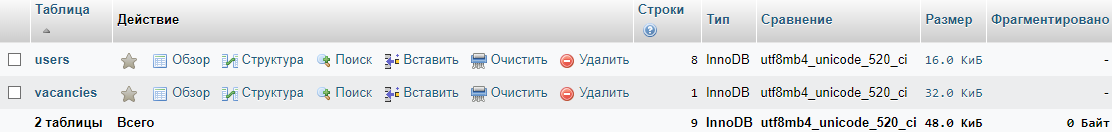
*Рис 2.5. Окно phpMyAdmin в браузере*

1. Часть 2

Разработка веб-документов (скриптов) проекта и структуры баз данных под управлением СУБД MySQL, сборка и тестирование веб-приложения.

2.1. Описание структуры базы данных

База данных **MPEI** содержит 2 таблицы: **users** и **vacancies**, предназначенные для хранения данных о пользователях и их анкетах соответственно.



*Рис 2.1. Таблицы в TMV*

Таблица **users** содержит следующие поля:

id (int) — идентификационный номер пользователя (первичный ключ, автоинкемент);

firstname (varchar(191)) — имя пользователя;

lastname (varchar(191)) — фамилия пользователя;

email(varchar(191)) — почта пользователя;

password (varchar(191)) — пароль, соответствующий почте пользователя;

avatar(varchar(191)) — фото пользователя;

info(text) — информация о пользователе;

skill\_coding (tinyint) — навык программирования пользователя;

skill\_photoshop (tinyint) — навык работы в фотошопе пользователя;

skill\_videoediting (tinyint) — навык монтирования видео пользователя;

status(varchar(191)) — статус пользователя (обычный / администратор);

complete\_count(int) — количество выполненных заказов пользователя;



*Рис 2.2. Поля в таблице users*

Таблица **vacancies** содержит следующие поля:

id (int)— идентификационный номер анкеты (первичный ключ, автоинкремент);

user\_id (int)— идентификационный номер пользователя, который оставил анкету;

title (varchar(191)) — название вакансии;

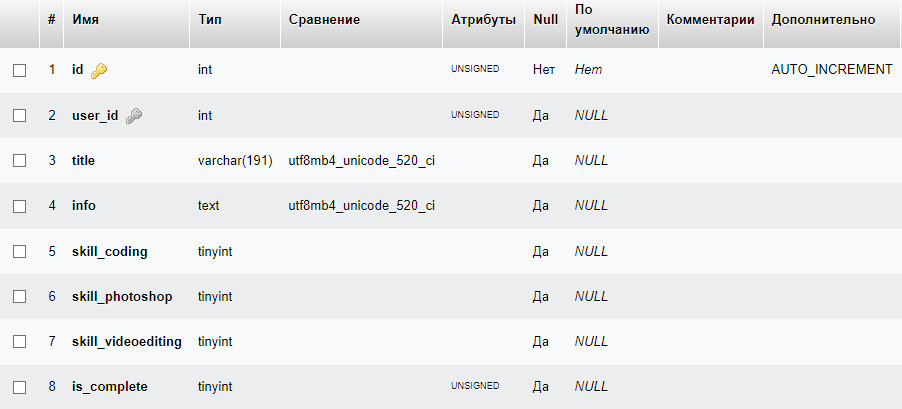
info (text) — основная информация вакансии;

skill\_coding (tinyint) — навык программирования пользователя;

skill\_photoshop (tinyint) — навык работы в фотошопе пользователя;

skill\_videoediting (tinyint) — навык монтирования видео пользователя;

is\_complete (tinyint) — статус анкеты;



*Рис 2.3. Поля в таблице vacancies*

Создание БД, ее конфигурация и заполнение данными создавалось вручную. Также была добавлена учетная запись Администратора (логин: admin, пароль: admin). Этот аккаунт имеет расширенный доступ при использовании сайта.

2.2. Описание разработки скриптов на PHP

Приложение разрабатывалось в 2 этапа:

1). Верстка всех страниц, применение к ним CSS и создание интерактивных элементов с помощью JS.

2). Добавление функционала с помощью языка PHP, создание переходов между страницами, соединений с БД и т.д.

К каждому документу подключен один и тот же CSS-файл *stile.css* и JS-файл *script.js*.

Все файлы проекта представлены на рисунке ниже:

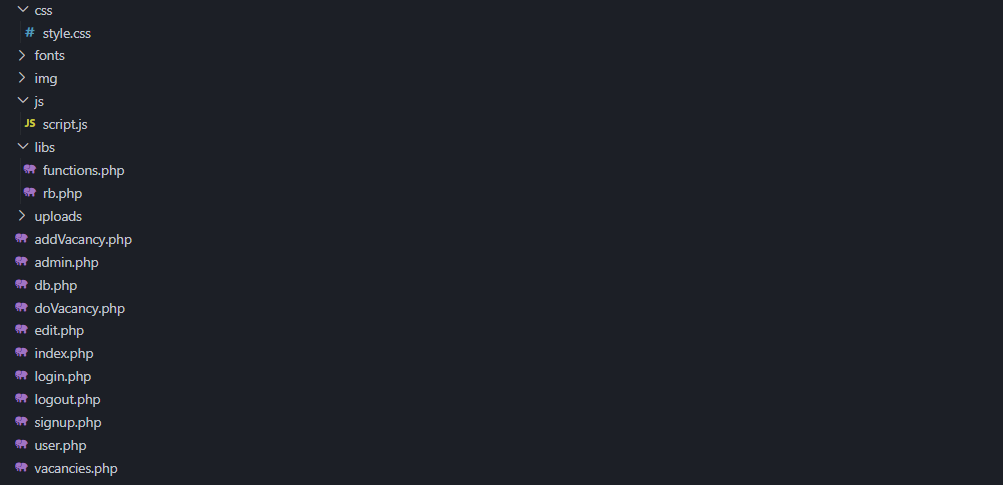


Рис. 2.4. Файлы проекта

**Документы проекта и связи между ними**

***Index.php*** — главная страница проекта, на которой коротко описан функционал приложения. Здесь же есть окно с именем пользователя и кнопка для выхода из аккаунта.

***signup.php*** — страница регистрации. При регистрации отправляет пользователя на главную страницу *index.php.*

***login.php*** — страница авторизации. При входе в учетную запись отправляет пользователя на страницу со статусом user на главную страницу *index.php*, а пользователя со статусом *admin* – на административную панель *admin.php*. Есть возможность перехода на страницу *signup.php*.

***logout.php*** — файл выхода из сессии.

***user.php*** — Личный кабинет. Здесь отображается вся информация о пользователе, а также его анкеты без отклика. Есть возможность перехода на страницы *edit.php* для изменения данных пользователя и *addVacancy.php* для добавления нового объявления. Уже имеющиеся объявления можно удалить.

***edit.php*** — страница изменения данных о пользователе.

***vacancies.php*** — страница со всеми не выполненными объявлениями, которые не принадлежат авторизованному пользователю.

***addVacancy.php*** — страница добавления нового объявления.

***doVacancy.php*** — скрипт «отклика» на вакансию.

***admin.php*** — административная панель. Доступ имеют только пользователи со статусом *admin*. Есть возможность просматривать все существующие учетные записи пользователей, их данные и активные объявления. Можно удалять как вакансии, так и самих пользователей.

***db.php*** — файл для настройки при подключении к БД. Подключен ко всем файлам, где имеется обращение к БД.

***functions.php*** — файл с используемыми функциями.

***rb.php*** — библиотека Red Bean.

**Описание работы системы пользователей (кратко)**

Для сохранения информации о текущем пользователе используется суперглобальный массив $\_SESSION. Все страницы, которым необходимо «знать», вошел ли пользователь, обращаются к массиву $\_SESSION к записи «user».

При регистрации пользователя на странице *signup.php* устанавливается соединение с БД, отправляется запрос для получения выборки из таблицы users с почтой, введенной пользователем. Если выборка не пуста (т.е. содержит хотя бы 1 запись), то пользователю предлагается заполнить форму заново (т.к. пользователь с таким email уже существует). Если выборка пуста, то отправляется запрос для добавления данных пользователя в таблицу *users*. Пользователь перенаправляется на страницу *index.php.* В массив $\_SESSION вносится запись «user» со значением логина пользователя.

При авторизации пользователя на странице *login.php* устанавливается соединение с БД, отправляется запрос для получения выборки из таблицы *users* с данными, введенными пользователем. Если выборка не пуста (т.е. содержит 1 запись), то пользователь перенаправляется на страницу *index.php*. В массив $\_SESSION вносится запись «user» со значением логина пользователя. Если выборка пуста, то пользователю предлагается заполнить форму заново.

При выходе пользователя из аккаунта сессия удаляется из базы данных.

Все поля форм при входе и регистрации проверяются скриптом на языке PHP.

**Описание работы системы объявлений (кратко)**

В личном кабинете авторизованный пользователь может просмотреть свои объявления, удалить их, а также добавить новые, перейдя на страницу *addVacancy.php* и указав там все необходимые данные: название объявления, список навыков, основную информацию.

Вся информация объявлений хранится в отдельной таблице *vacancies.*

На странице *vacancies.php* пользователь может просмотреть вакансии других пользователей, а также откликнуться на них, но только на те, для которых достаточно навыков. Если пользователь не обладает умениями, указанными в объявлении, то такое объявление ему не удастся выполнить, кнопка будет заблокирована.

Код всех файлов прикреплен в Пункте 3 «Приложение».

**Учетная запись администратора**

Для осуществления модерации сайта предусмотрена панель администратора, доступ к которой имеют пользователи со статусом - admin. Администратор имеет возможность просматривать, а также удалять объявления всех пользователей и их учетные записи.

2.3. Описание функционала разработанного приложения

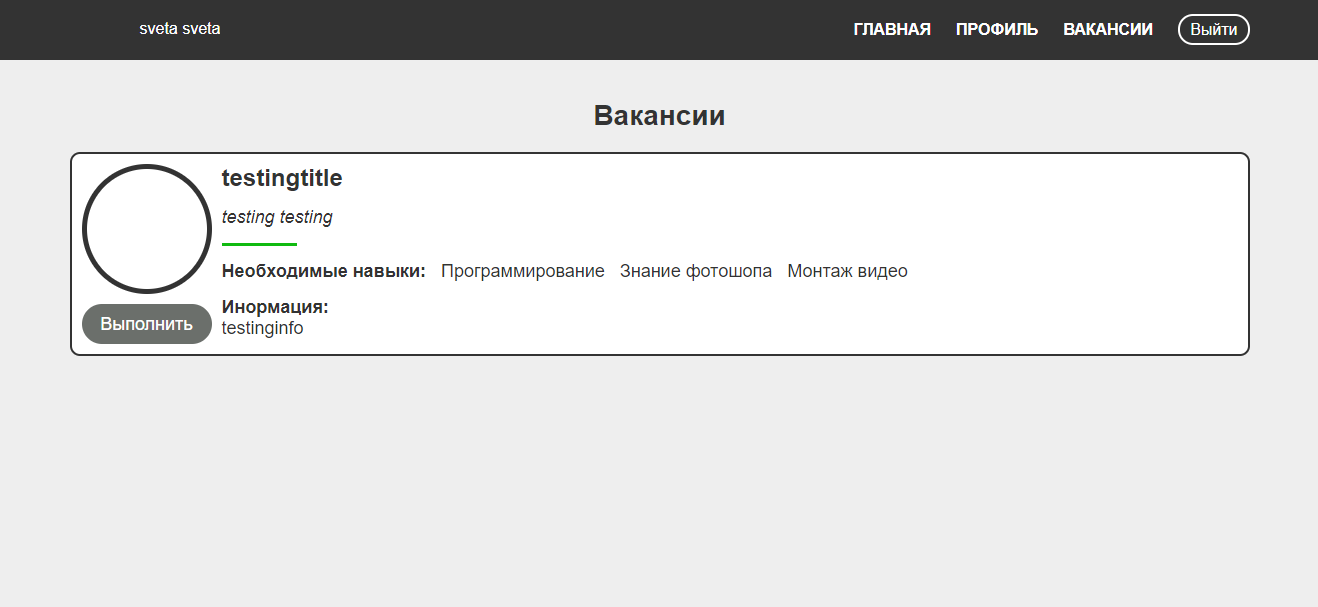
На каждой из страниц имеется возможность перехода на главную страницу, в личный кабинет и на страницу вакансий, на страницу регистрации и авторизации, а при уже имеющейся сессии – на файл logout.

При входе на сайт пользователь попадает на главную страницу. На ней можно увидеть имя текущего пользователя, его фото, есть возможность выйти из аккаунта при нажатии кнопки «Выйти».



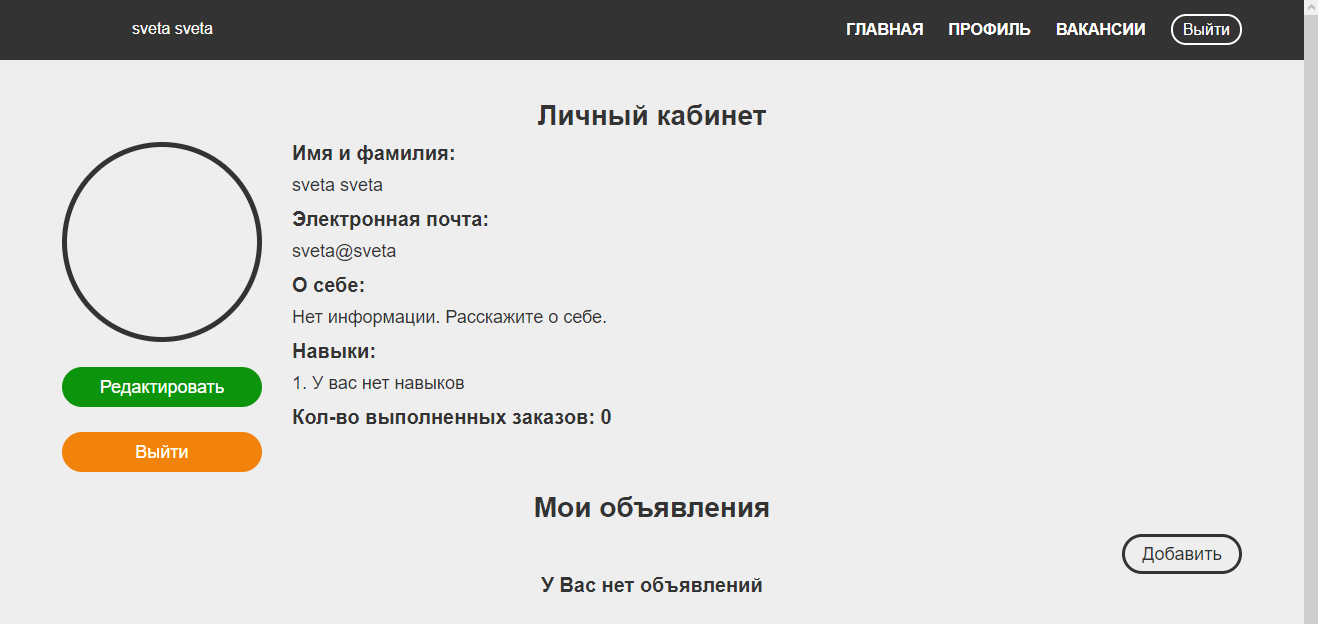
*Рис 2.5. Главная страница*

На странице «Вакансии» представлен список доступных анкет других пользователей с их заголовком, описанием, именем автора, его фотографией и перечнем навыков. При нажатии на кнопку «Выполнить» сработает скрипт, который поменяет статус вакансии в таблице на «1», что будет означать, что данная вакансия выполнена, поэтому она перестанет отображаться в списке. Также количество выполненных заказов пользователя увеличится на 1. После чего осуществится переход в личный кабинет пользователя, где можно будет убедиться в выше сказанном. При переходе на страницу «Вакансии» неавторизированного пользователя перебросит на страницу авторизации.



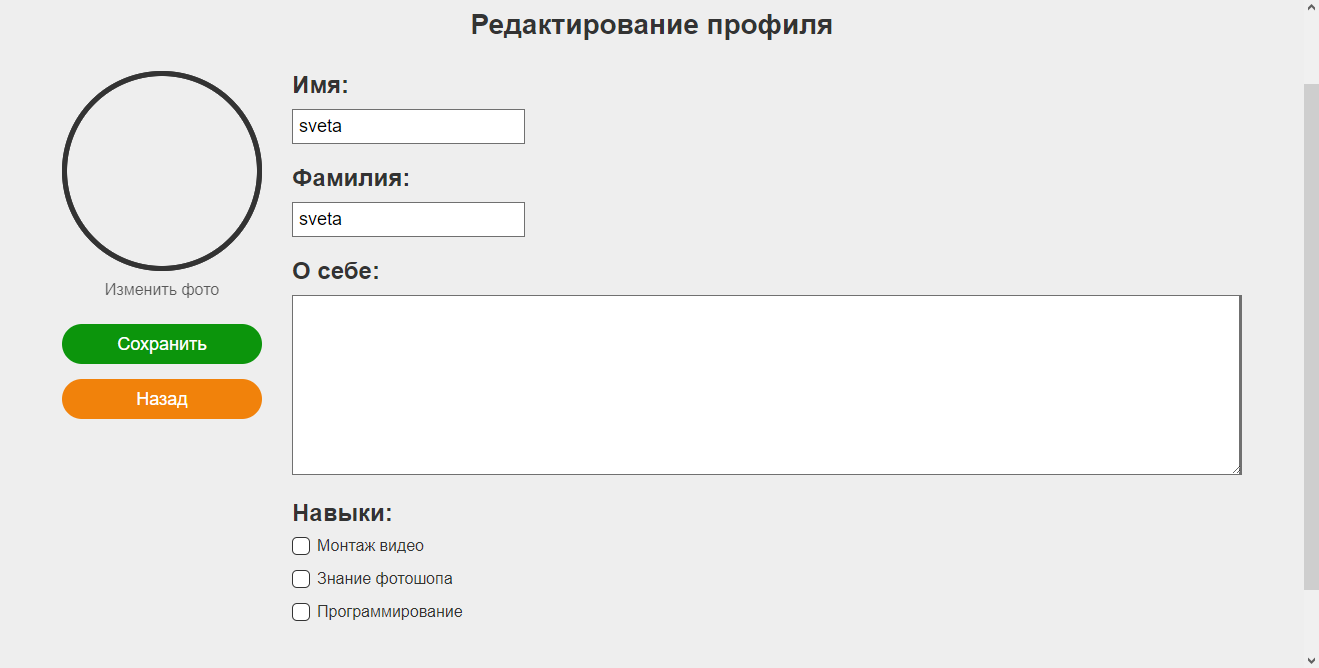
*Рис 2.6. Страница со списком анкет*

В личном кабинете отображается информация о пользователе: его имя, фамилия, почта, на которую зарегистрирована учетная запись, его фотография, навыки. Нажав кнопку Данные можно поменять, нажав на кнопку «Редактировать» и перейдя на страницу *edit.php*. При переходе личный кабинет неавторизированного пользователя перебросит на страницу авторизации.



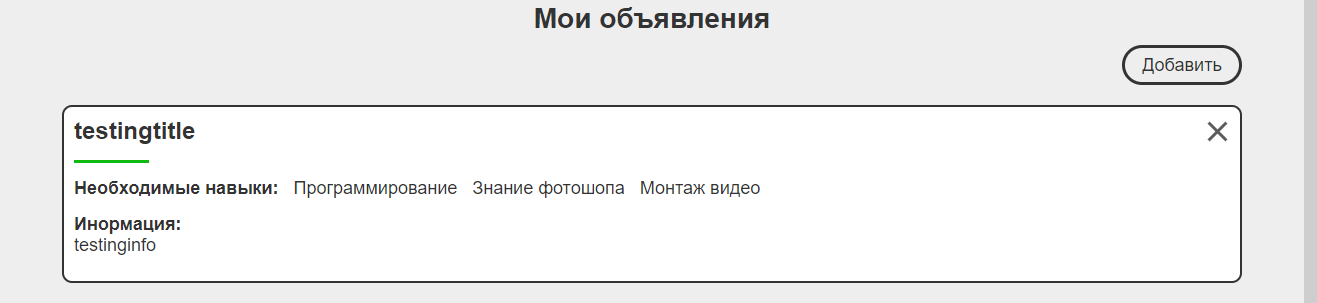
*Рис 2.7. Личный кабинет пользователя, информация*

На странице редактирования профиля пользователь может поменять свои имя, фамилию, фотографию, дополнить рассказ о себе, указать новоприобретенные навыки. Если изменения не требуются, можно вернуться в Личный кабинет, щелкнув по кнопке «Назад».



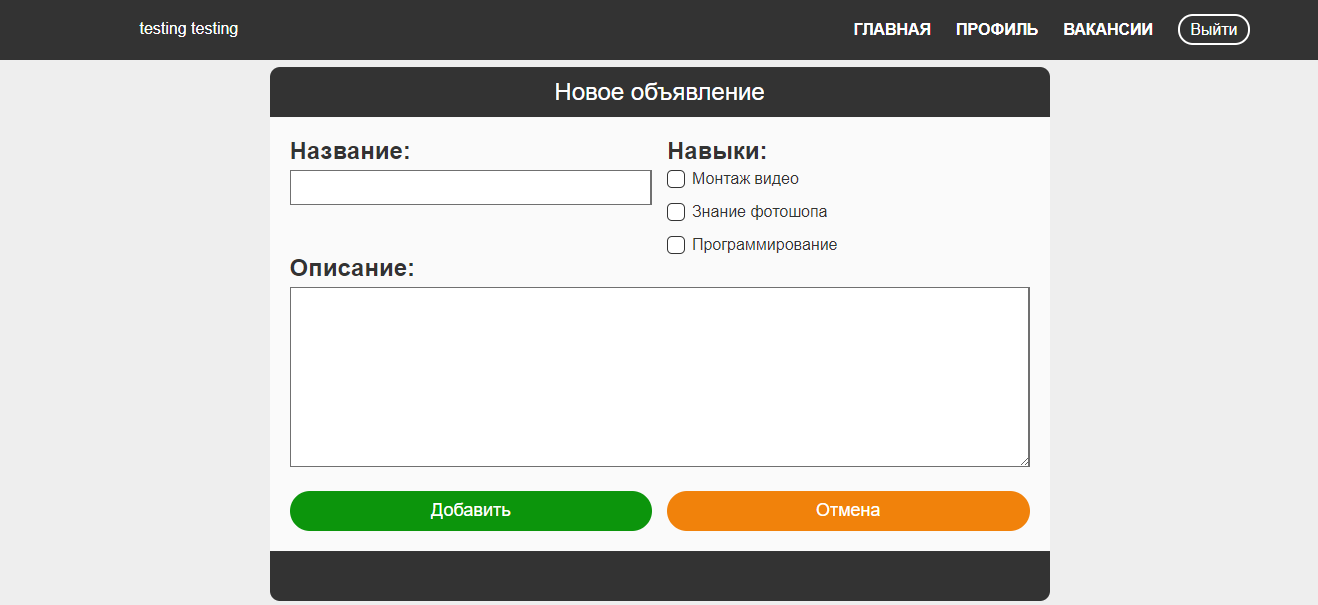
*Рис 2.8. Редактирование профиля*

Также в личном кабинете отображаются все активные вакансии пользователя, на которые еще никто не откликнулся. Существует возможность добавить новое объявление, нажав на кнопку «Добавить» и перейдя на страницу *addVacancy.php*, и удалить имеющиеся: на каждой анкете предусмотрена кнопка-крестик.



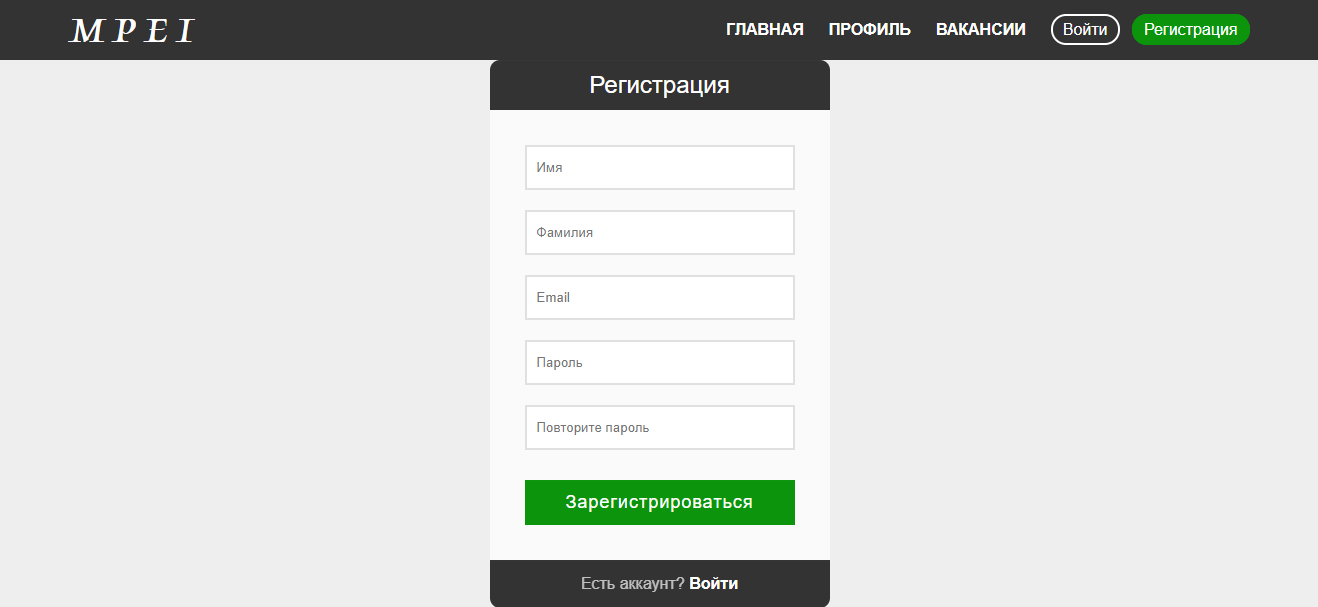
*Рис 2.9. Личный кабинет пользователя, объявления*

На странице добавления анкеты, пользователь может создать новое объявление. Для этого необходимо указать название объявления, необходимые для этого навыки, а также дать краткое описание. Хотя бы один навык должен быть выбран, а текстовые поля не должны быть пустыми, в противном случае выведутся соответствующие сообщения об ошибке. Если пользователь передумал размещать новую вакансию, он может вернуться в личный кабинет, нажав кнопку «Отмена».



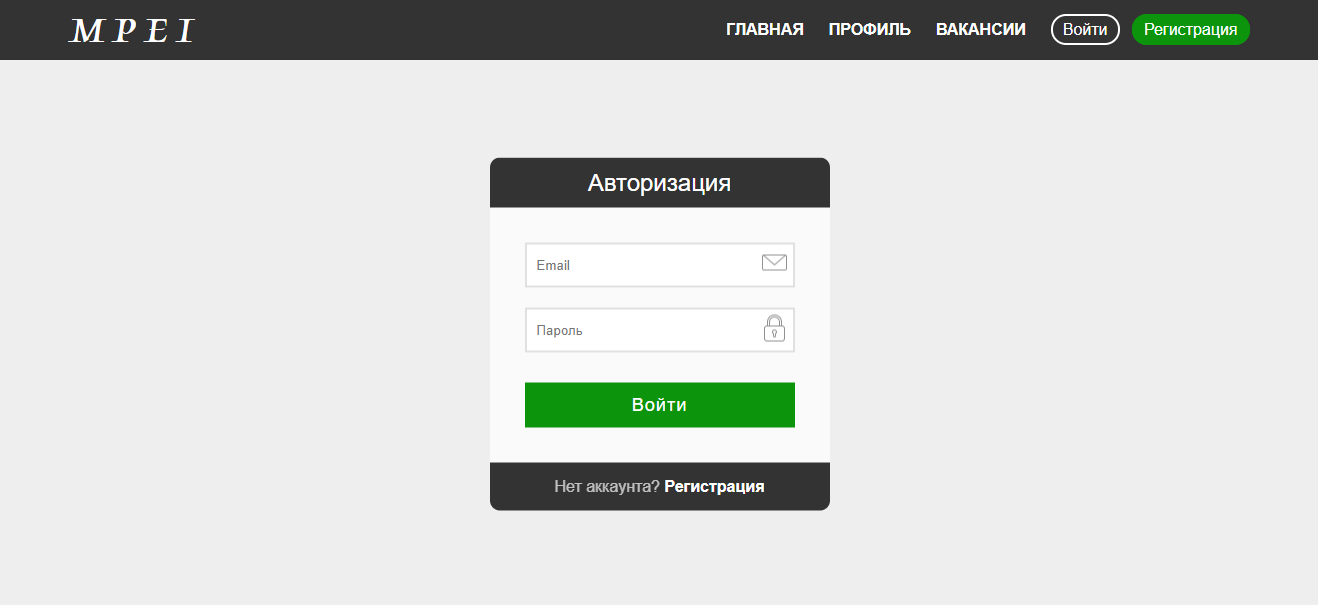
*Рис 2.10. Добавление нового объявления*

На странице «Регистрация» предлагается заполнить форму для регистрации или перейти на страницу «Авторизация» по нажатию кнопки «Войти», если у пользователя уже имеется аккаунт. Если уже имеется сессия, пользователя будет перебрасывать на главную страницу. При вводе неверных данных будет выведено соответствующее предупреждение. (Изображения ниже обрезаны для компактности). После успешного входа пользователь попадает на главную страницу.



*Рис 2.12. форма регистрации*

На странице «Авторизация» предлагается заполнить форму для авторизации или перейти на страницу «Регистрация» по нажатию кнопки «Регистрация», если у пользователя еще нет аккаунта. Если уже имеется сессия, пользователя будет перебрасывать на главную страницу. При вводе неверных данных будет выведено соответствующее предупреждение. (Изображения ниже обрезаны для компактности). После успешного входа пользователь попадает на главную страницу.



*Рис 2.14. форма авторизации*

При авторизации пользователей со статусом *admin* перебрасывает на администаривнуб панель *admin.php*, где они могут простатривать информацию о всех пользователях со статусом *user* и удалять их, просматривать объявления всех пользователей и удалять их. Предусмотрен выход по нажатию кнопки «Выйти».



*Рис 2.16. Административная панель, пользователи*