МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

кафедра «Вычислительная техника»

**Лабораторная работа №2**

по дисциплине: «Человек машина интерфейс».

Выполнил:

студент гр. ИВТАПбд-21

Кондратьев П.С.

Проверила:

Валюх В. В.

г. Ульяновск, 2018

**Программный продукт сайт - Musfe**

**Цели создания продукта.**

Приложение, позволяющее работать с пользователями (обратный ответ) с использованием баз данных(sql). Сайт представляет собой главную страницу (регистрация) в который нужно пройти 2 этапа регистрации с загрузкой фотографии на сервер. После производиться переадресация на страницу пользователя (админа) где можно будет либо оставить ответ (письмо админу, которое можно будет увидеть в базе данных), либо заняться настройкой пользователей (для админа).

**Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области являются языки программирования php, javascript, также mysql, phpmyadmin.

Предполагается, что с данным приложением будут работать студенты ФИСТ и преподаватели, и они должны иметь возможность решать с её помощью следующие задачи:

1. базовые знания php, javascript, mysql

2. Проверка защиты пользователя

3. Оптимизация сложных sql запросов

4. Расчет среднего времени выполнения запросов

5. Любовь к комментариям в коде и желание разбираться в сложных по архитектуре системах

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложение для базового курса php.

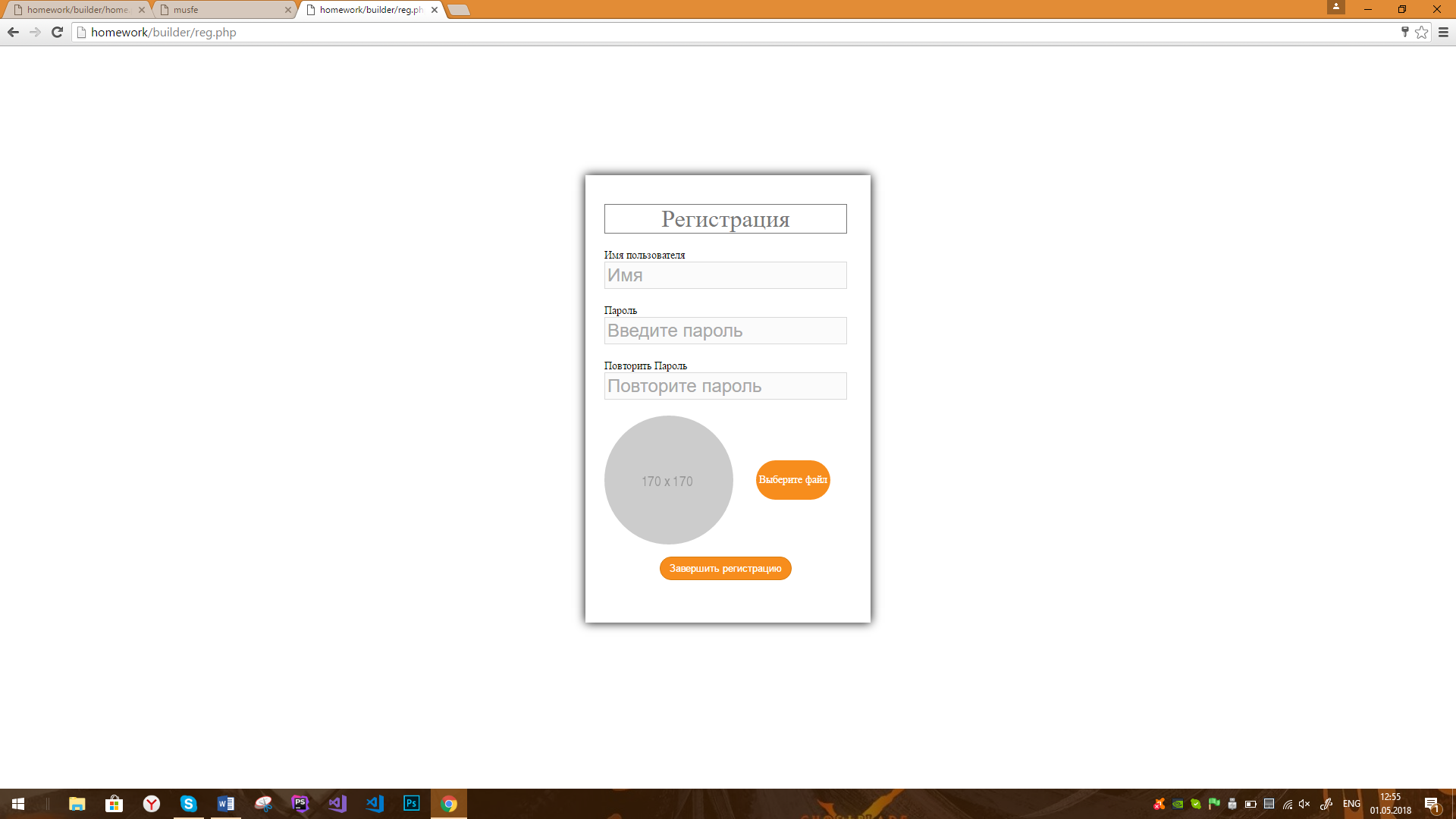
2. Студенты ФИСТ

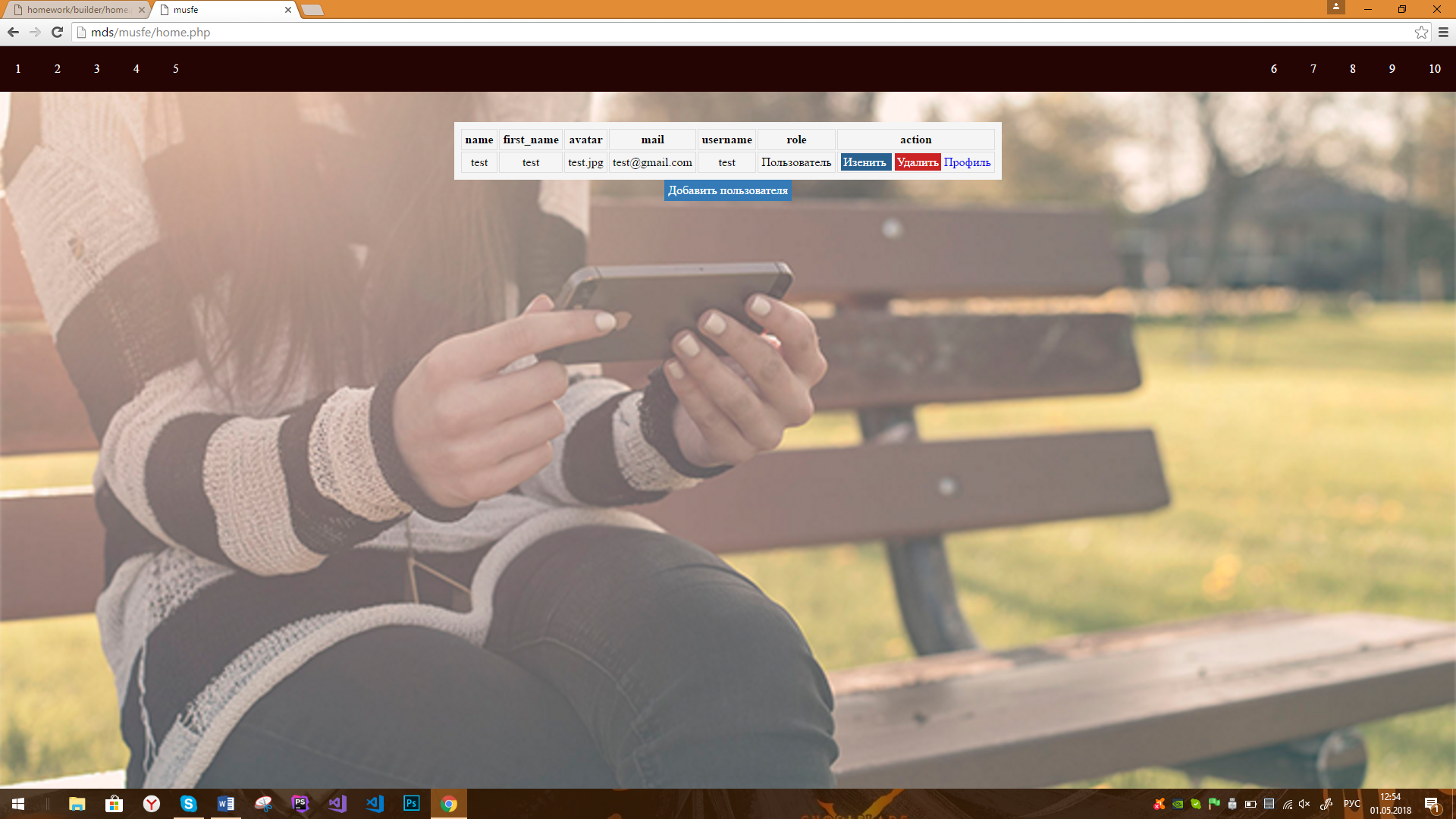
Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «web программирование».

**4. Сравнительный список произведенных изменений программного продукта**

Программный продукт претерпел множество изменений в плане изменения входа на сайт и главной страницы админа по надстройке пользователей. Такие изменение изменили всю конструкцию сайта по работе с пользователем. Были внесены изменения с работой фотографии пользователя и графическим представлением главной страницы.

**Первоначальный вид пользовательского интерфейса до изменений**





1. ***Регистрация***

Регистрация представляет собой форму с вводом логином и созданием своего пароля для входа на сайт. Кроме того, пользователь может сразу же при регистрации задать своему профилю аватар своей страницы. Стилистика страницы регистрации проста и минималистична. В файле css, который отвечает за стиль страницы, написан стилистически верный, грамотно оформленный код.

Это следует из-за того, что стиль главной страницы регистрации выполнена в белом стиле (который идет по default), с добавлением к белому фону были добавлены маленькие штрихи:

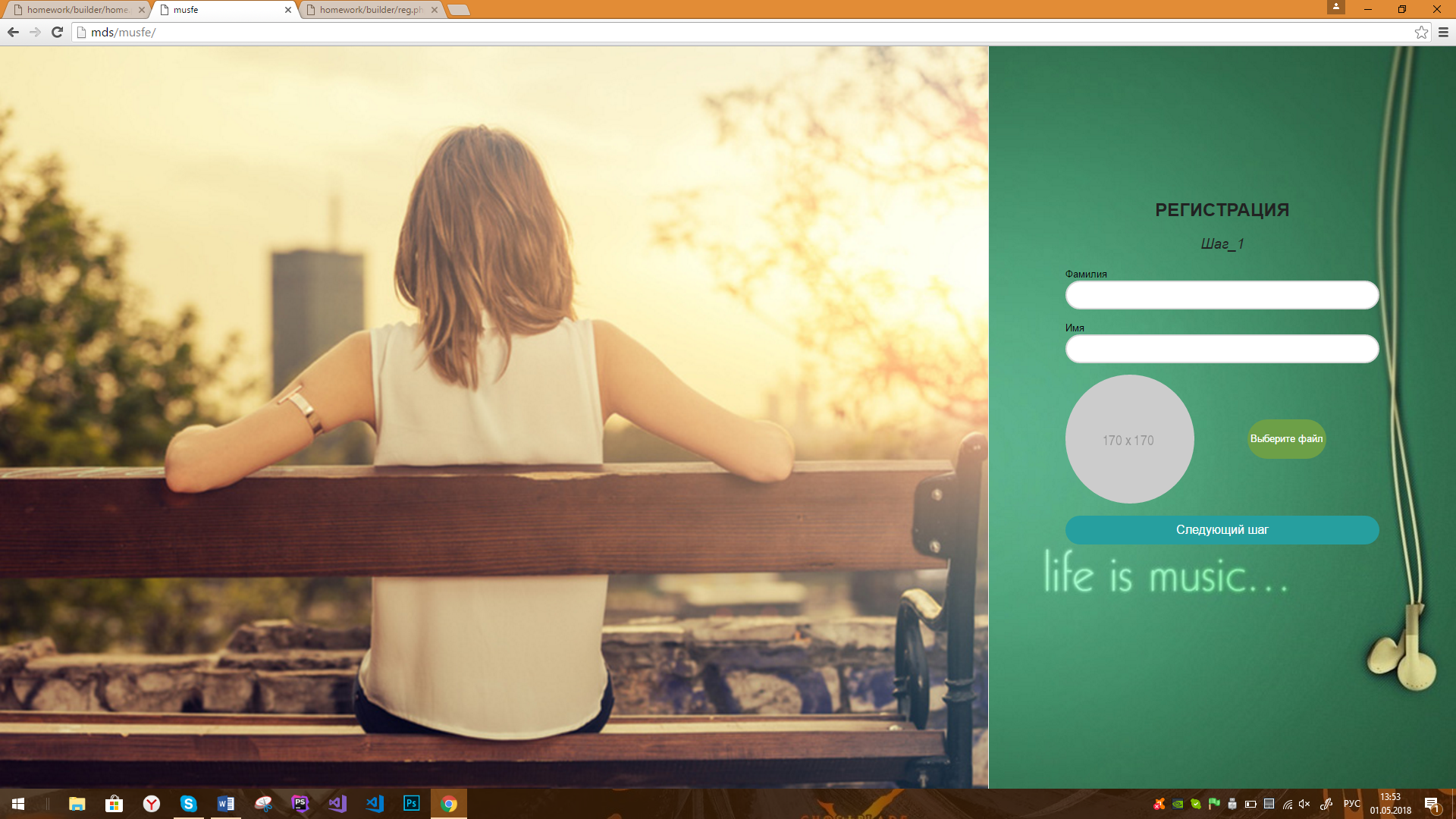
1. Кнопки (Butten) выполнены в форме овала и покрашены в оранжевый цвет, что выгладить намного ярче и красочнее, чем прямоугольные кнопки серого цвета (по default)
2. Форма сделана в виде листа, который находиться по верх белового фона, это повлияло на восприятие страницы, так как форма кажется объемной. Этому поспособствовало использование теней в коде css
3. ***Страница админа***

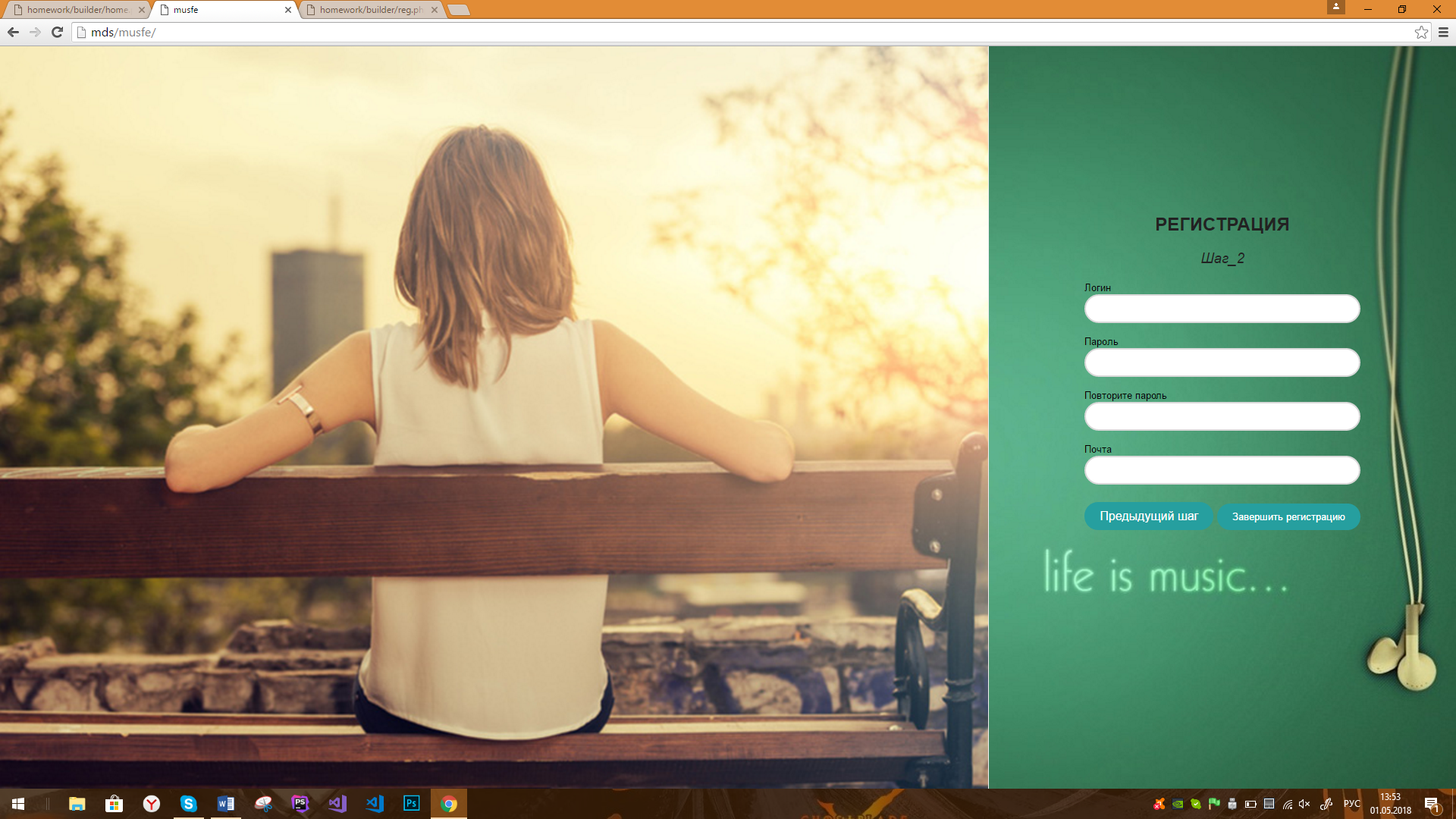
Страница админа представляет собой таблицу пользователей, в которой присутствуют надстройки пользователей. В таблице есть возможность удалять, изменять, добавлять пользователя. Минус этой таблицы в том, что если админ хочет поменять аватар пользователя, то ему нужно писать в таблице имя картинки, которое будет лежать на серве.

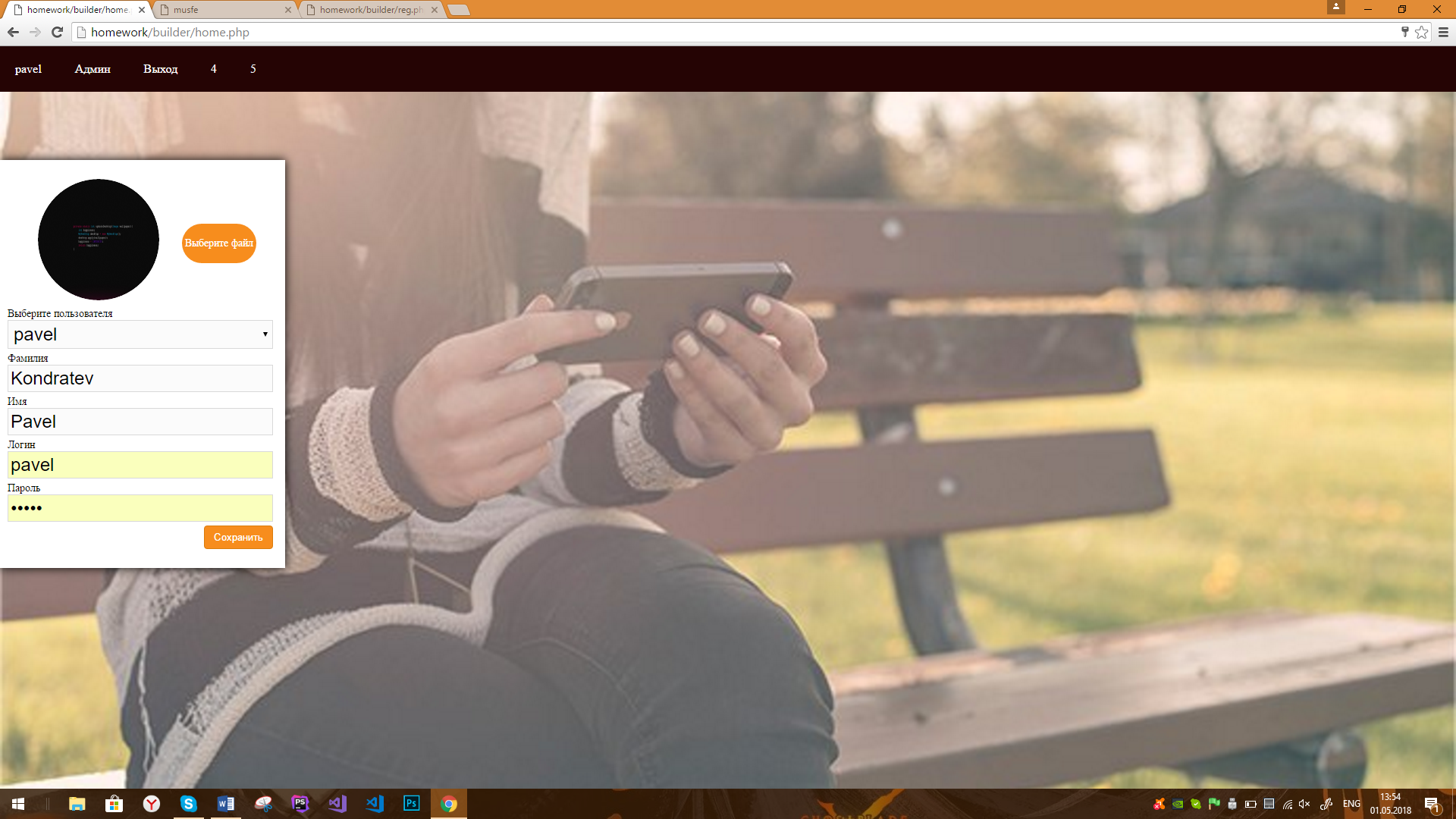
Стилистика страницы:

1. Добавлена картинка на главный экран, нежели как на страницы регистрации
2. Таблица пользователей выполнена в белом цвете с цветовыми вкладками в нее (надстройки пользователя: красная – удалить синяя – изменить голубая - добавить). Кнопка добавления была вынесена отдельной надстройкой для пользователя, так как она является не надстройкой самого пользователя, а добавлением нового админом. Кнопки были сделаны в более просто стиле нежели с страницы регистрации, они было только лишь перекрашены и оставлены в стандартной форме – прямоугольника.

**Вид пользовательского интерфейса после изменений**

В новой версии программного продукта сайт – **Musfe** были добавлены и внесены кардинальные изменения по стилистике сайта и работе с пользователями. Изменения претерпели и регистрацию – был изменён стиль страницы и добавлен 2 шаг регистрации, а страница админа была переделана из табличного варианта в формочную.





1. ***Регистрация***

Кардинально изменен внешний вид страницы и работа с СУБД. Во-первых, регистрация стала в 2 шага, стреми же самыми данными, как и в 1 варианте, но с добавлением новых полей (Фамилия, Имя, Почта). Сама форма стала более красочной и с анимацией перехода на 2 шаг. Была изменена форма кнопок на овальные и изменен цвет на зеленый. Сама форма стала находиться на правой части экрана, те самым разграничивая главный арт сайта с самой регистрацией (2 отдельные детали). Вместо белого фона стали 2 фотообоины подходящие по стилистике сайта.

1. ***Страница админа***

Вместо простой таблицы была создана форма Пользователей. В ней хранятся все данные о пользователе, также в ней доступно изменения, удаление, и создание нового пользователя. Все свелось к тому что в форме присутствует поле select, в котором находятся все пользователи и есть возможность создать пользователя выбрав эту надстройку в select. Главным + стало то что по сравнению с прошлой версией программного продукта появилась возможно удобно и просто менять фотографии у пользователей, не прибегая к тому что каждый раз придется заходить в папку сервера и брать оттуда имя файла, которое в последующем продеться записывать в таблицу. Все свелось к тому что админ может сам выбрать фото со своего компьютера, и она загрузиться (измениться) на серве и у пользователя. Одной из главных надстроек стало то что в панели нельзя удалить самого себя, так как эта форма есть и у пользователя, но она только для изменения своих личных данных. Это стало возможно с помощью ролей на сайте (user, admin). Изменение пользователя осуществляется теперь с помощью переписывания данных в форме, что стало гораздо проще чем дополнительное модальное окно как в 1 версии. Но также в этой версии присутствует -, он заключается в то что если на сайте будет много пользователей, то каждый раз придётся искать нужного пользователя в огромном списке (select), это повлияет на время работы с пользователя и на внешний вид работы с формой. Но так как этот программный продукт лишь тестовым вариантом и у него не будет столько много пользователей, то он является оптимальным по скорости работе и красоте самого сайта.

5. **Примечание**

Демонстрация php, javascript, css кода по изменениям главной страницы админа

1. **css**

*Кнопки:*

.button{

border: solid 1px #da7c0c;

background: #f78d1d;

color: #fff;

padding: 7px 12px;

border-radius:4px;

float: right;

cursor: pointer;

}

.button\_center {

border: solid 1px #da7c0c;

background: #f78d1d;

color: #fff;

padding: 7px 12px;

border-radius: 50px;

display: block;

margin: 0 auto;

cursor: pointer;

}

.button\_center:hover {

transition: 1s;

box-shadow: 0 0 10px rgb(0, 0, 0);

}

Использовалось border-radius (скругление кнопки), color (цвет), box-shadow (анимация тени).

*Javascript*

function myFunction() {

var val = document.getElementById("mySelect").value;

$.post("getUser.php", {id: '' + val}).done(function (res) {

object = JSON.parse(res);

$('#first\_name').val(object.first\_name);

$('#name').val(object.name);

$('#username').val(object.username);

var scr = 'images/profile/' + object.avatar;

$('#delFile').val(object.avatar);

if (val == "Создать пользователя") {

var per = 'http://placehold.it/160x160';

$('#images').attr("src", per);

} else {

$('#images').attr("src", scr);

}

});

var myid = document.getElementById("delete").value;

if (val == myid) {

$("#del").css("display", "none");

} else {

$("#del").css("display", "block");

}

if (val == "Создать пользователя") {

$("#del").css("display", "none");

$("#insert").css("display", "block");

$("#button").css("display", "none");

} else {

$("#insert").css("display", "none");

$("#button").css("display", "block");

}

}

if (window.File && window.FileReader && window.FileList && window.Blob) {

function renderImage(file) {

var reader = new FileReader();

reader.onload = function (event) {

the\_url = event.target.result;

$('#preview').html("<img style=\"width: 170px; height: 170px; border-radius: 200px; border: 1px solid white;\" src='" + the\_url + "' />")

}

reader.readAsDataURL(file);

}

$("#the-photo-file-field").change(function () {

renderImage(this.files[0])

});

} else {

alert('The File APIs are not fully supported in this browser.');

}

$(function () {

var val = document.getElementById("mySelect").value;

var myid = document.getElementById("delete").value;

if (val == myid) {

$("#del").css("display", "none");

} else {

$("#del").css("display", "block");

}

})

Была использован объект FileReader(), который позволил работать с фото на нашем серве, он поспособствовал тому что пользователь сможет посмотреть preview своей фотографии на странице, и последующему сохранению ее на серве. Также была написана функция для того чтобы было проще работать с самой форой, и обработаны мелочи по тому что нельзя было удалять себя и тд. Этого ы добились с помощь нашего скрипта, который проверяет наш выбор в select и, если выбран сам пользователь (который выполнил вход на сайт), что с помощью скрипта мы меняем css (стиль) нашей кнопки удалить на display: none, который в свою очередь до этого был display: block, после чего наша кнопка исчезает.

**Программный продукт Работа с коллекциями с использованием WPF**

**Цели создания продукта.**

Приложение, позволяющее работать с коллекцией, в котором реализовано визуальное окно для работы с коллекцией. Приложение также имеет анимацию и имеет возможность производить действия с коллекцией (добавление, удаление, изменение)

**Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области являются язык программирования c# (реализация самой коллекции и работы с ней) и DoubleAnimation (анимация для модальных окон).

Предполагается, что с данным приложением будут работать студенты ФИСТ и преподаватели, и они должны иметь возможность решать с её помощью следующие задачи:

1. базовые знания с#

2. знание библиотеки DoubleAnimation

3. работа с коллекцией (list)

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «WPF».

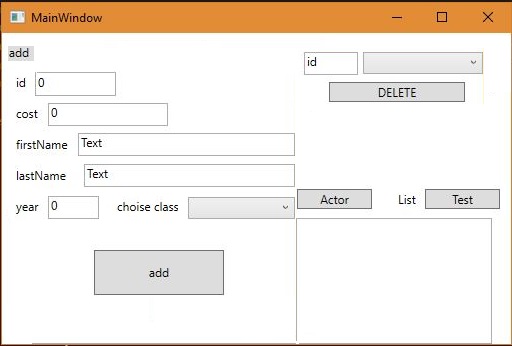
2. Студенты ФИСТ

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «основы WFP».

**4. Сравнительный список произведенных изменений программного продукта**

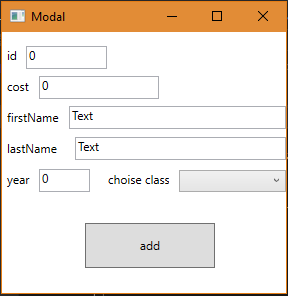
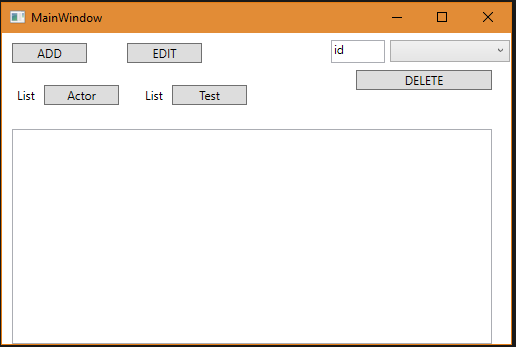
Программный продукт претерпел изменение в плане изменения главного WPF окна, в котором первоначально все действия с коллекцией находились в одно окне. Все textbox были в одном окне разбросанные по разным частям формы, что мешало лучшему восприятию по работе с коллекцией.

**Первоначальный вид пользовательского интерфейса до изменений**



1. ***Форма коллекции***

Представлена собой одну большую надстройку по коллекции, в которой присутствуют поля для удаления, добавления и вывода коллекции. Стиль формы был сделан в минималистическом стиле с использованием белых и серых цветов. Так же для удобства удаления пользователя из коллекции было сделано объект select, для удобного нахождения пользователя чтобы его удалить.

**Вид пользовательского интерфейса после изменений**

1. ***Форма коллекции***

Полностью изменили окно коллекции, разбив его на отдельные части для более удобного использования надстроек коллекции (оставили только удаление коллекции по id с помощью выбора из select). Был изменен размер textbox для вывода, сделали его больше чтобы уместить всю нужную информацию о коллекцию. Были добавлены новые кнопки (button ADD и EDIT) в которых как раз размещены наши методы с коллекциями (добавление и изменение).

1. ***Метод ADD***

Был вынесен в отдельное модальное окно, в которые перенесли все textbox их старой версии продукта с добавлением автоинкремента для поля id. Новшеством также стала анимация загрузки модального окна и его скрытые при завершении надстроек над пользователем коллекции. При использовании метода ADD появляется модальное окно с эффектом жалюзи (плавно спуская вниз до полного открытия) и с тем же самым эффектом при завершении работы, но в обратном направлении.

5. **Примечание**

Демонстрация с# кода по изменениям анимации главного и модально окна коллекций

public NewWindow1(string title)

{

InitializeComponent();

DoubleAnimation da = new DoubleAnimation();

da.From = 0;

da.To = 300;

da.Duration = TimeSpan.FromSeconds(2);

BeginAnimation(NewWindow1.HeightProperty, da);

Title = title;

textBox.Text = Convert.ToString(test.MainWindow.it++);

}

С помощью объекта DoubleAnimation(), удается добиться плавной и красивой анимации главного и модального окна (ADD) с эффектом жалюзи. Также здесь генерируется автоинкремент для поля id в textbox (ADD).

private void button\_Click(object sender, RoutedEventArgs e)

{

if (comboBox.Text == "Actor")

{

test.Lists.AddActor(Convert.ToInt32(textBox.Text), Convert.ToInt32(textBox1.Text), textBox2.Text, textBox3.Text);

} else

{

test.Lists.AddTest(Convert.ToInt32(textBox.Text), Convert.ToInt32(textBox4.Text));

}

DoubleAnimation da1 = new DoubleAnimation();

da1.To = 0;

da1.Duration = TimeSpan.FromSeconds(2);

da1.Completed += Animation\_Completed;

BeginAnimation(NewWindow1.HeightProperty, da1);

}