Министерство науки и высшего образования РФ  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
“Забайкальский государственный университет”  
Энергетический факультет  
Кафедра информатики и вычислительной техники

**Курсовая работа**  
по предмету:  
**Технологии WEB-программирования**

На тему:

**Разработка WEB-приложения с использованием СУБД**

**Выполнил:**  
Студент (ФИО): Банковский Александр Сергеевич   
Группа: ИВТ-20  
Курс: Третий  
Форма обучения: Очная

**Руководитель:**  
 Должность: Доцент кафедры ИВТ и ПМ  
 ФИО: Семигузов Дмитрий Александрович

Чита – 2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ   
Федеральное государственное бюджетное образовательное учреждение высшего образования   
«Забайкальский государственный университет»   
(ФГБОУ ВО «ЗабГУ»)   
Факультет: Энергетический факультет  
 Кафедра: Информатики и вычислительной техники

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА   
к курсовой работе

По дисциплине: Технологии WEB-программирования

На тему «Разработка WEB-приложения с использованием СУБД»

Цели курсовой работы:

* Разработка дизайна веб-приложения
* Разработка функциональных возможностей веб-приложения
* Подключение базы данных к веб-приложению

Руководитель проекта: Семигузов Дмитрий Александрович

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ   
Федеральное государственное бюджетное образовательное учреждение высшего образования   
«Забайкальский государственный университет»   
(ФГБОУ ВО «ЗабГУ»)   
Факультет: Энергетический факультет Кафедра:   
Информатики и вычислительной техники

ЗАДАНИЕ   
на курсовую работу

По дисциплине Технологии WEB-программирования

Студенту

специальности Информатика и Вычислительная техника

1. Тема курсового проекта: Разработка адаптивного веб-приложения

2. Срок подачи студентом законченного проекта: 7.10.2023

3. Исходные данные к работе: cоздать веб-приложение для добавления и сохранения шуток в форме «вопрос-ответ».

4. Перечень подлежащих разработке курсового проекта вопросов: дизайн веб-приложения, функционал веб-приложения, база данных для веб-приложения.

Дата выдачи задания:

Руководитель курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Задание принял к исполнению «\_\_» мая 20\_\_г.  
Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

**Содержание**

[Введение 5](#_Toc147333507)

[Глава 1. Анализ задачи 6](#_Toc147333508)

[Глава 2. Анализ данных 7](#_Toc147333509)

[Глава 3. Программная реализация 8](#_Toc147333510)

[Заключение 25](#_Toc147333511)

[Список использованной литературы 26](#_Toc147333512)

Введение

Задолго то появления развитых технологий и их повсеместного использования, люди пытались изобрести эффективный способ передачи и хранения информации. В текущий момент человечество уже достигло отличного результата в виде всемирной сети Интернет (для обмена); Также различные базы данных на серверах (для хранения). Данные технологии плотно вошли в нашу жизнь, и для поддержки и развития работоспособности технологий было изобретено большое число инструментов для организации связи между пользователем и сервером, для отображения ответа сервера в виде, удобном для пользователя, а также средств разработки, созданных для ускорения и удешевления процесса разработки и поддержки веб-приложений.

Привлекательность веб-технологий как средства доставки информации во многом определяет универсальный интерфейс между человеком и компьютером. Каждому человеку понятны надписи, заголовки, ссылки, картинки. Веб-интерфейс как средство доступа к информации интуитивно понятен. Следствием простоты веб-интерфейса является широкое использование Интернета как канала коммуникации.

Значение веб-технологии, как для разработчиков программного обеспечения, так и для обычных пользователей во многом определяется тем, что это, прежде всего - интеграционная технология. И трудно найти более удачный пример того, как можно интегрировать различные источники информации и различные ее типы. Веб-технологии позволяют создавать простые для освоения, легкодоступные, крайне дешевые, быстро обновляемые информационные, диалоговые, справочные системы.

Глава 1. Анализ задачи

* 1. **Описание предметной области**

Предметная область данной курсовой работы заключается в создании

сайта, предоставляющего пользователю базовый функционал связанный с профилем (регистрироваться, авторизовываться, выходить из профиля и удалять профиль). Также добавлять шутки в базу данных, просматривать, изменять их и удалять данные записи.

Прежде чем приступить к созданию веб-приложения, необходимо определить его функциональные возможности.

Функционал сайта - это то, что делает веб-ресурс полезным, что превращает его из обычного источника информации в полезную программу.

Некоторые программные модули, которые планируется сделать для будущего веб-приложения:

* Форма регистрации пользователя
* Форма авторизации пользователя
* Форма ввода новой шутки и отображение всех добавленных шуток.

После того, как был определен дизайн веб-приложения и его функциональные возможности, можно приступить к его проектированию.

**1.2. Постановка задачи**

В рамках курсовой работы необходимо создать веб-приложение, которое предоставит пользователю возможность добавлять в базу данных через сайт данные, просматривать их, удалять. Особое внимание нужно уделить внешнему виду веб-приложение. Необходимо, что веб приложение было максимально простым, но в то же время функциональным. Внешний вид сайта должен быть выдержан в одной цветовой гамме.

Также необходимо, что бы созданное веб-приложение работало в любом современном браузере.

Глава 2. Анализ данных

Проектирование базы данных (Далее: БД) является очень важным этапом, от которого зависят последующие этапы разработки приложения.

К базовым понятиями модели БД «сущность - связь» относятся: сущности, связи между ними и их атрибуты (свойства).

Сущность - любой конкретный или абстрактный объект в рассматриваемой предметной области (в реляционной БД каждой сущности назначается таблица).

Атрибут - это свойство сущности в предметной области. Его наименование должно быть уникальным для конкретного типа сущности. В реляционной БД атрибуты хранятся в полях таблиц.

Связь - взаимосвязь между сущностями в предметной области. Связи представляют собой соединения между частями БД (в реляционной БД - это соединение между записями таблиц).

Обозначим сущности создаваемого приложения:

* Пользователь
* Шутка

Сущности «Пользователь» соответствуют следующие атрибуты:

* Идентификатор (Далее: ID)
* Имя
* Электронный адрес почты
* Пароль

Сущности «Шутка» соответствуют следующие атрибуты:

* ID пользователя
* ID шутки
* Первая часть шутки (Вопрос)
* Вторая часть шутки (Ответ)

Глава 3. Программная реализация

Созданная программа состоит из следующих страниц:

* index.php
* login.php
* profile.php
* registration.php

Также, к сайту подключены стили из файла «main.css» и логика обращения к БД в следующих файлах:

* DB connect.php
* DB.php
* userJokes.php
* userProfile.php
* userLogin.php
* userRegistration.php

При создании сайта использовался язык программирования php, обеспечивающий внутреннюю логику сайта.

Каждая задача веб-приложения была вынесена в отдельную функцию.

Опишем детально каждую из функций, что используются в программе.

На рис. 1. представлена главная страница созданного веб-приложения (index.php), что представляется пользователю после запуска приложения.

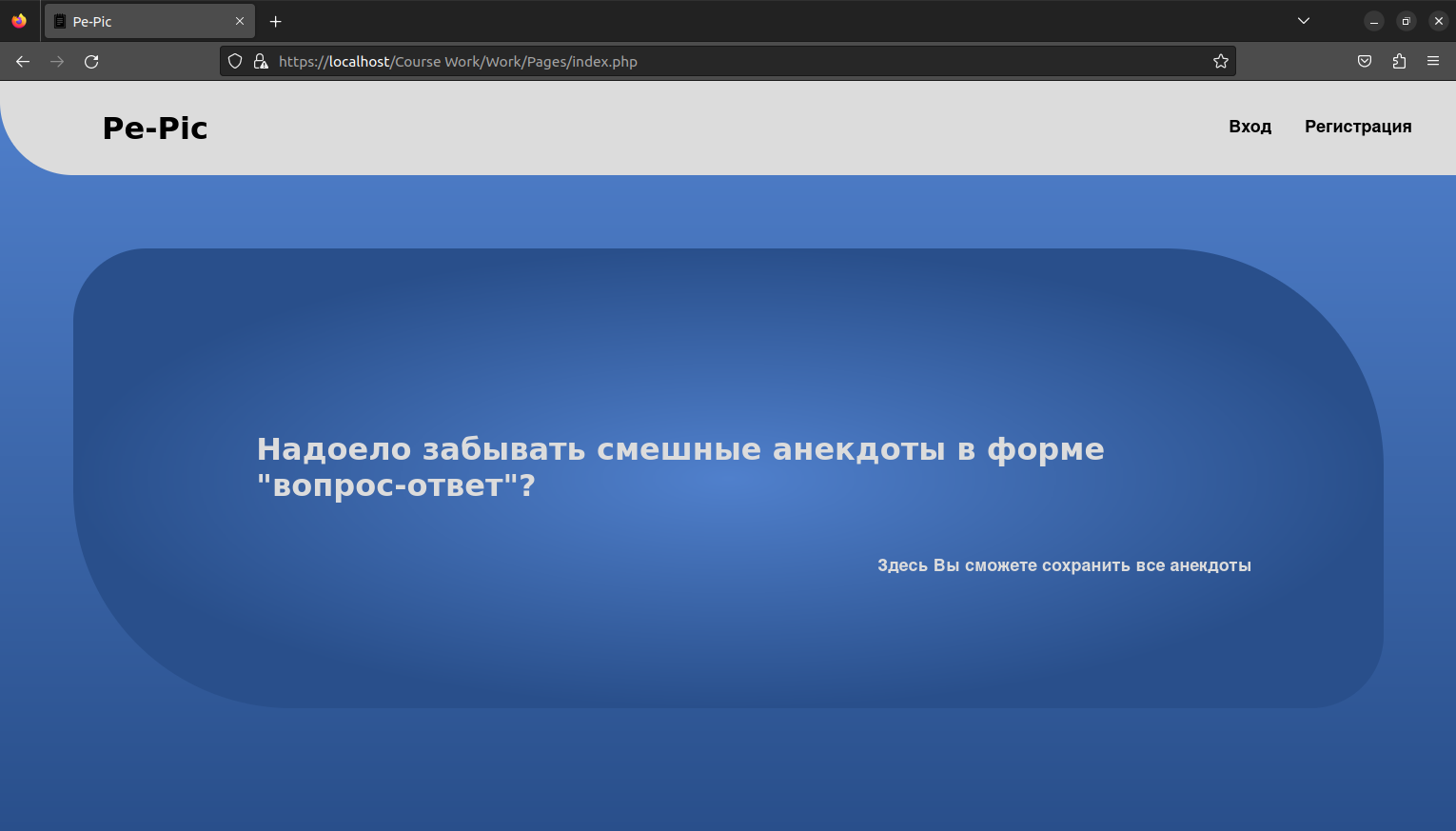


Рисунок . Главная страница сайта (index.php)

На данной странице пользователь имеет возможность просмотреть разделы навигации: переход к регистрации, к авторизации и переход на главную страницу.

Переход по навигации осуществлён через html-тег <a>.

На рисунке 2.1. показана страница регистрации. На рисунке 2.2. показана страница авторизации.

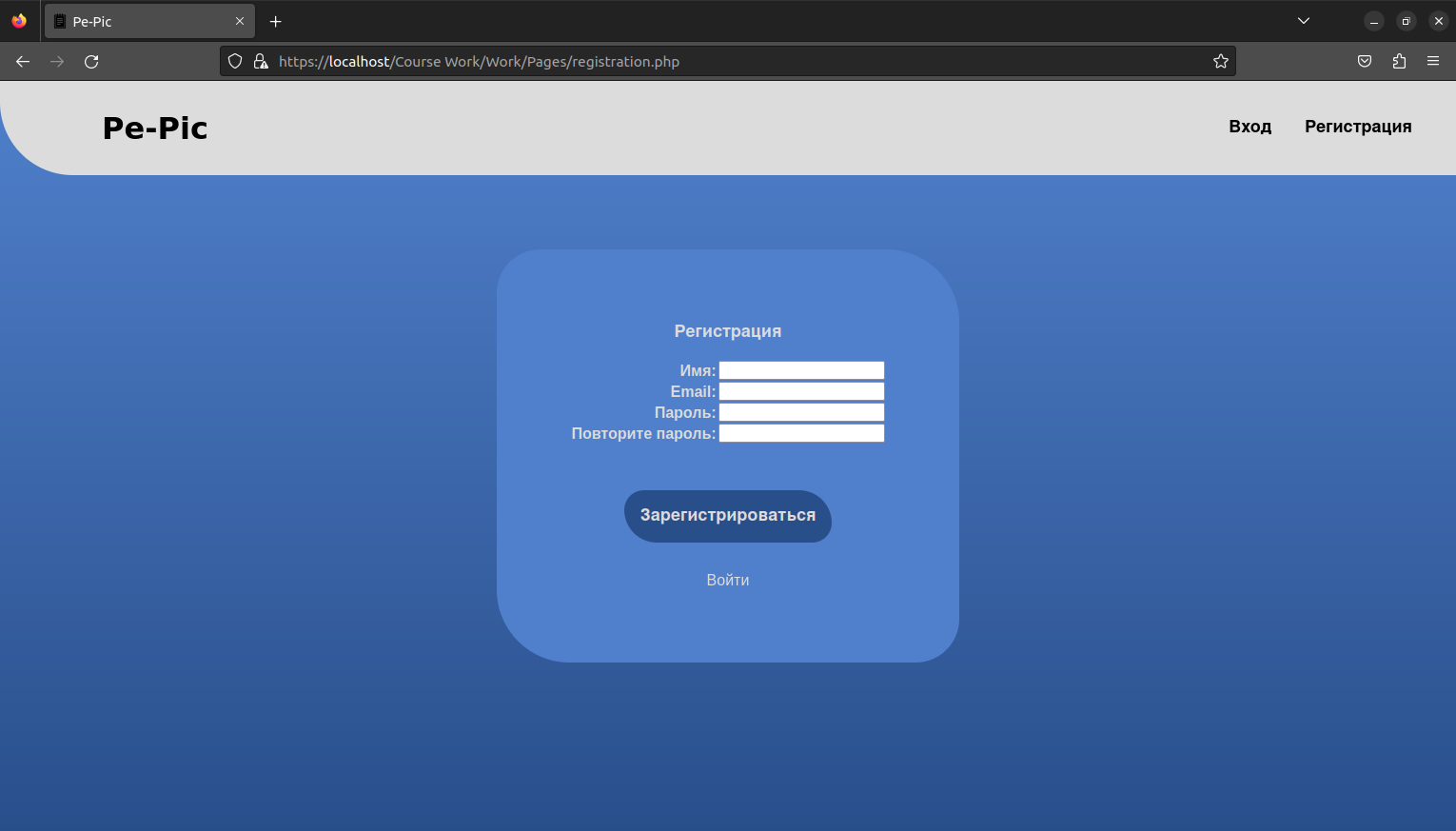


Рисунок .1. Страница регистрации (registration.php)

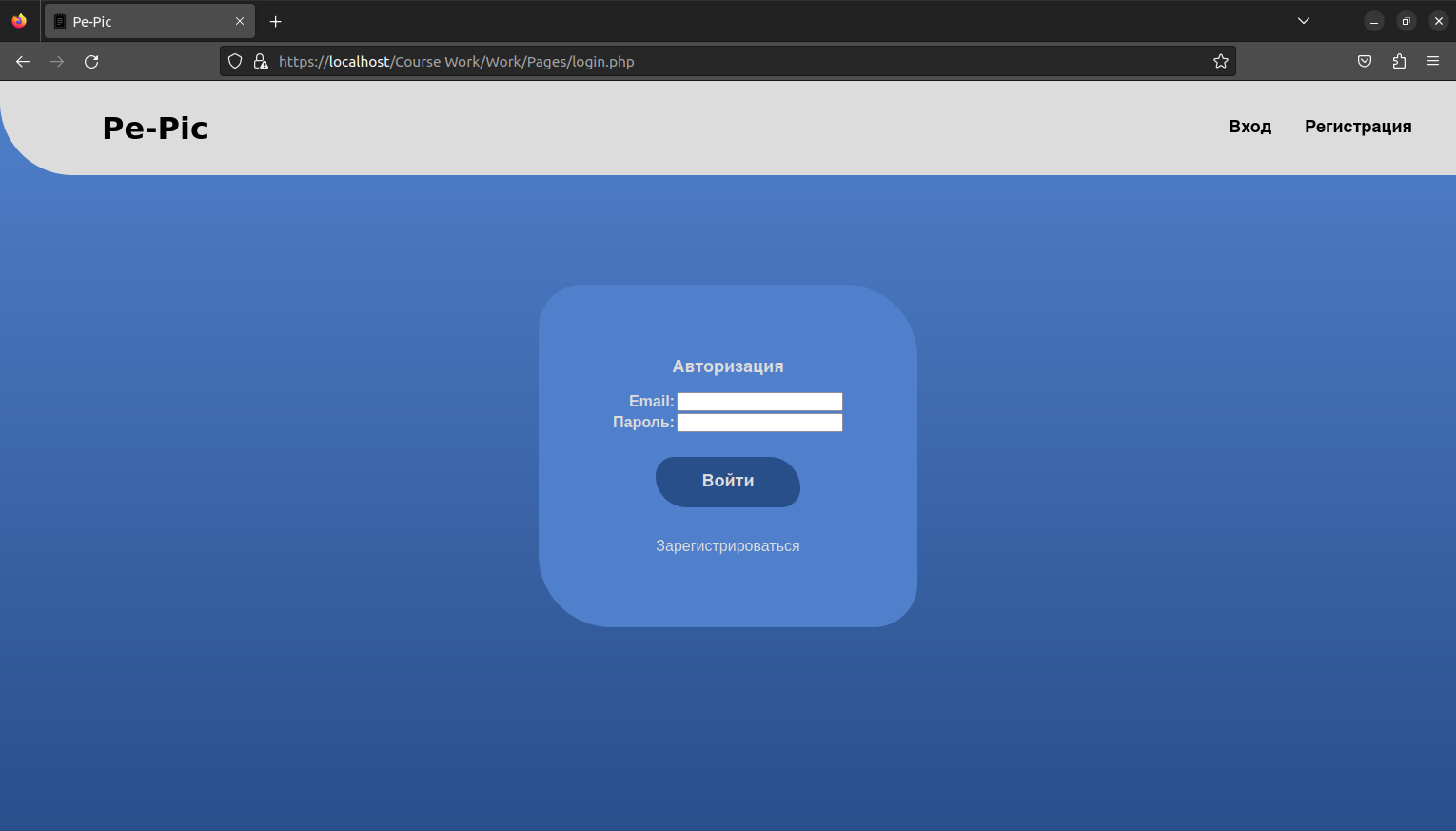


Рисунок 2.2. Страница авторизации (login.php)

Главная страница после авторизации показана на рисунке 3.

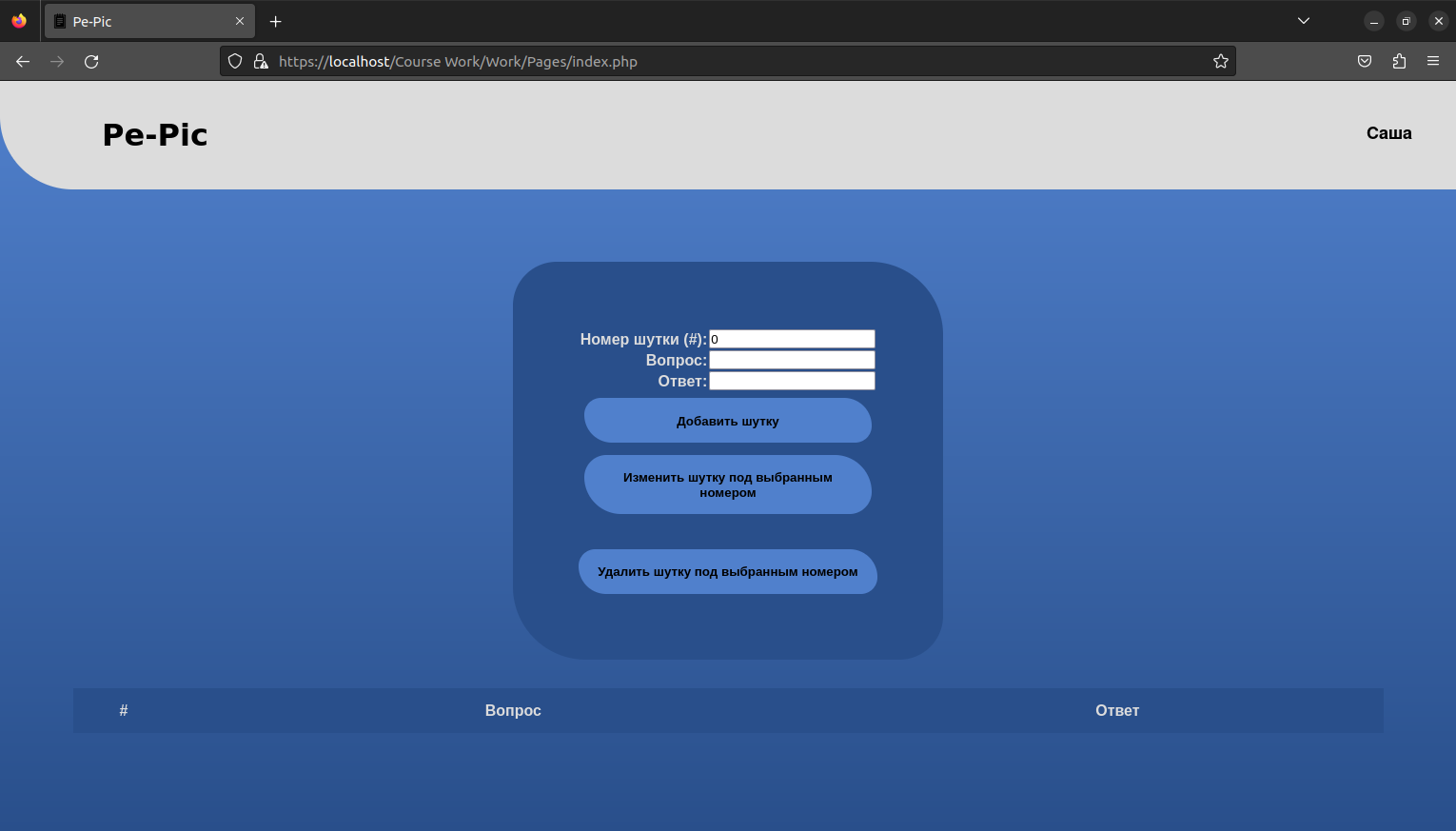


Рисунок . Главная страница (index.php) после авторизации

Далее изображён код со страницы регистрации, в котором показано как именно добавляется пользователь в базу данных.

Код из файла «DB connection.php», в котором осуществляется связь с БД - рисунок 4.1.; Код из файла «DB.php», в котором осуществлены добавление пользователя в БД и получение пользователя из БД по его «id» - рисунки 4.2.1. и 4.2.2. соответственно; На рисунке 4.3. показан код из файла «userRegistration.php» для обработки страницы авторизации;

Обработка действий пользователя на всех страницах осуществляется через html-тег <form> с методом «post».



Рисунок .1. Организация соединения с БД (MySQL) (файл «BD connection.php»)

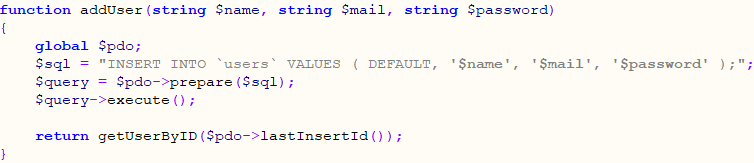


Рисунок 4.2.1. Функция добавления пользователя в БД (файл BD.php)

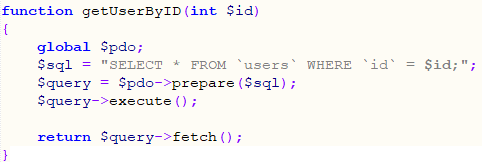


Рисунок 4.2.2. Функция получения пользователя по его id в БД (файл BD.php)



Рисунок 4.3. Код из файла «userRegistration.php» для обработки страницы авторизации

После успешной регистрации или авторизации пользователь может изменить свои данные: имя, почту и пароль; Также пользователь, находясь нас транице профиля, может выйти из профиля (деавторизоваться) или и вовсе удалить свой профиль из базы данных. Страница профиля изображена на рисунке 5.

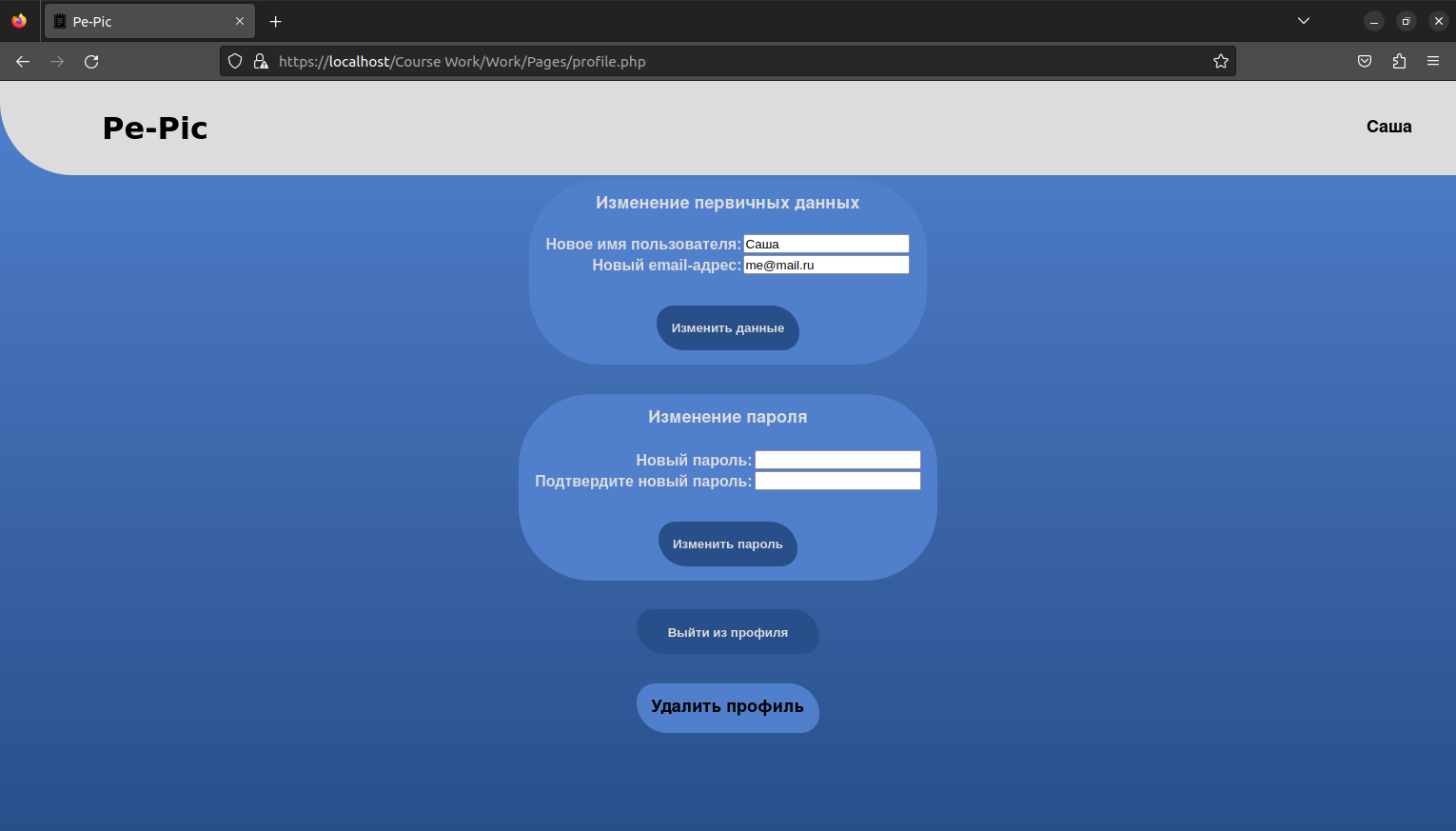


Рисунок . Страница профиля (profile.php)

В случае ввода некорректных данных пользователь будет оповещён об этом текстом, с указанием ошибки, а функция, которая должна была выполниться при вводе корректных данных, не будет выполнена. Попытка ввести неверные данные будет показана на примере страницы регистрации в нескольких вариациях:

1. Заполнение не всех данных пользователя – рисунки 6.1.1-2;
2. Неверное повторение (подтверждение) пароля – рисунки 6.2.1-2;

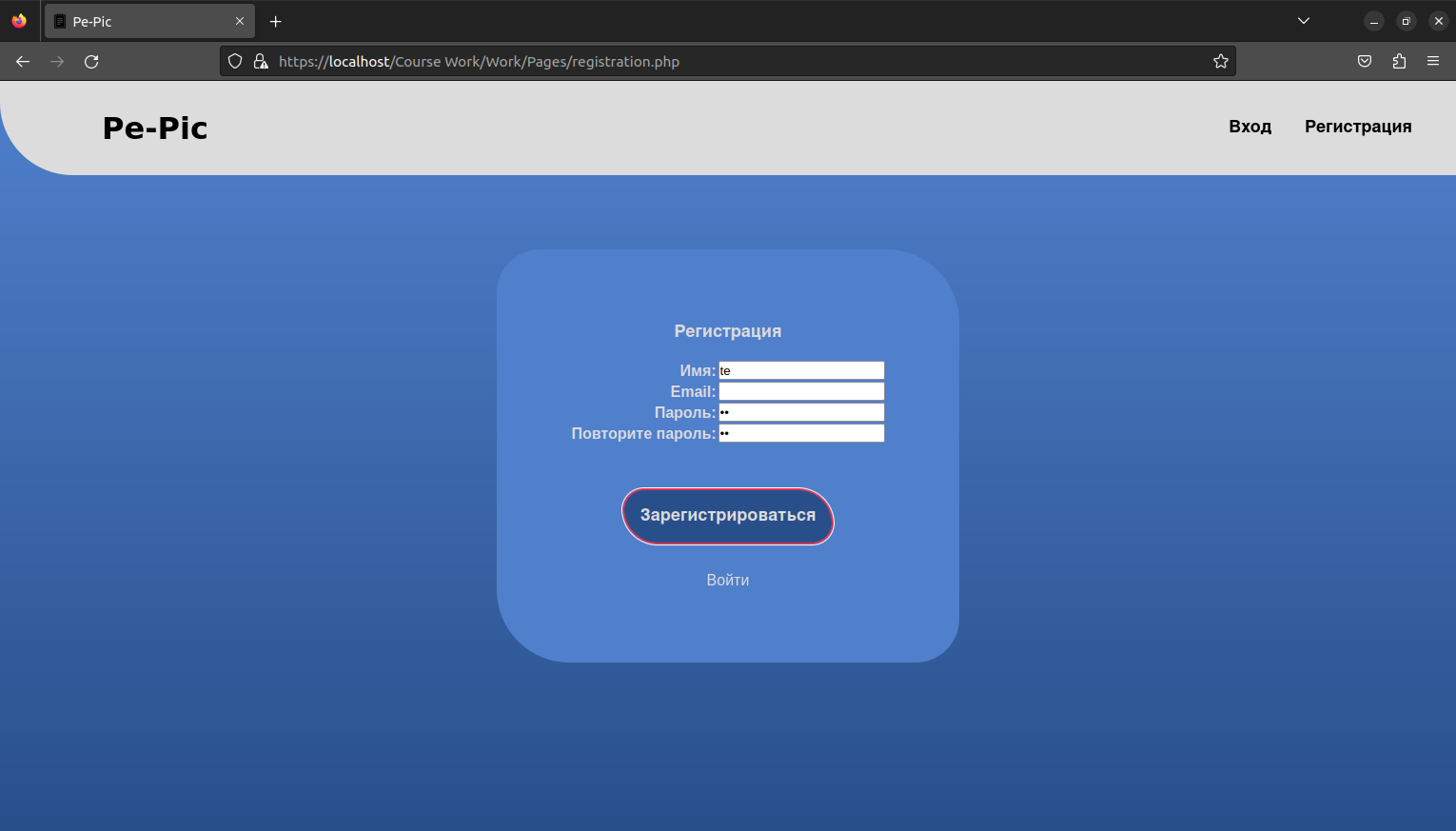


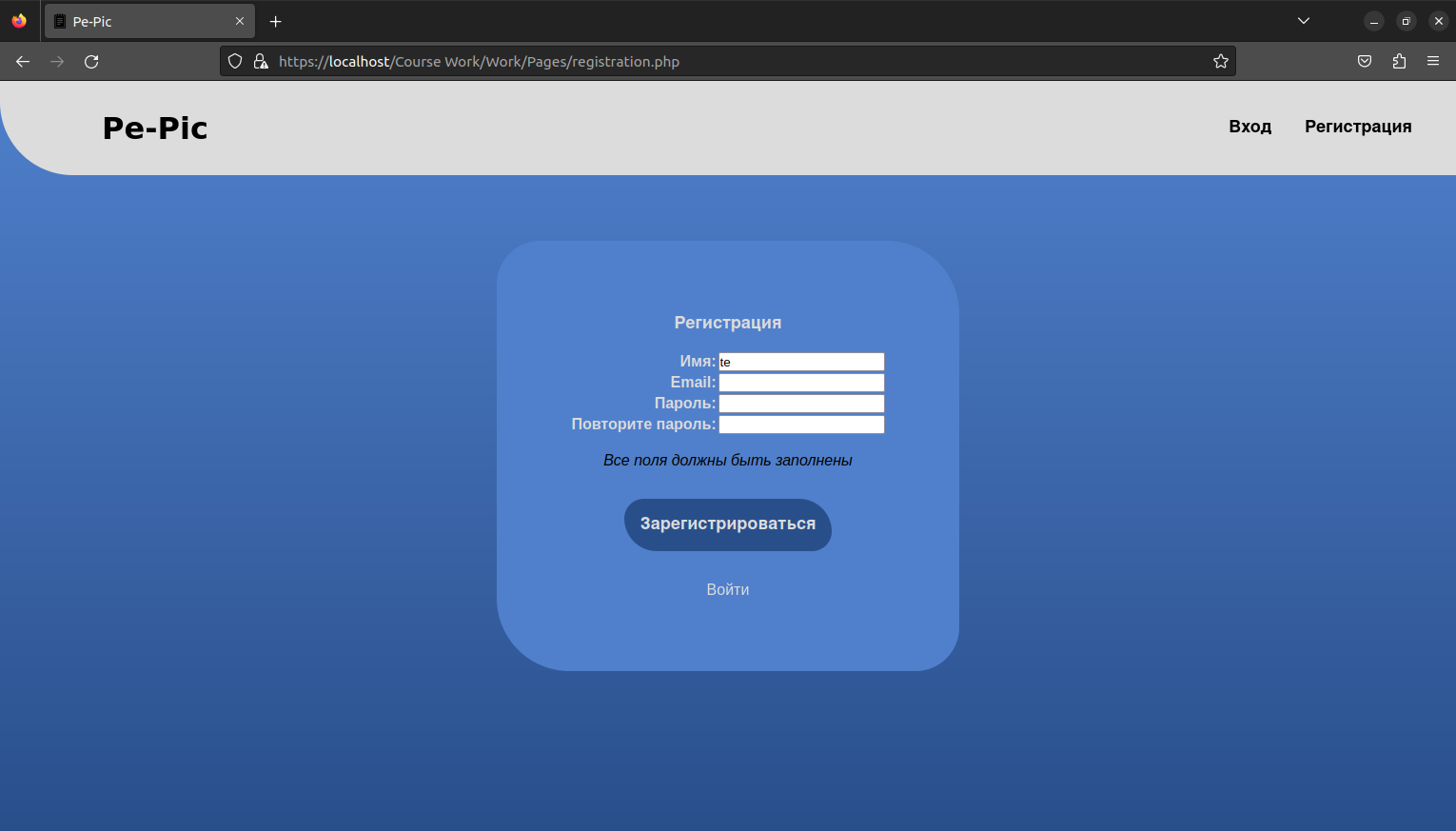
Рисунок .1.1. Заполнение не всех данных при регистрации. 1

Рисунок 6.1.2 Заполнение не всех данных при регистрации. 2

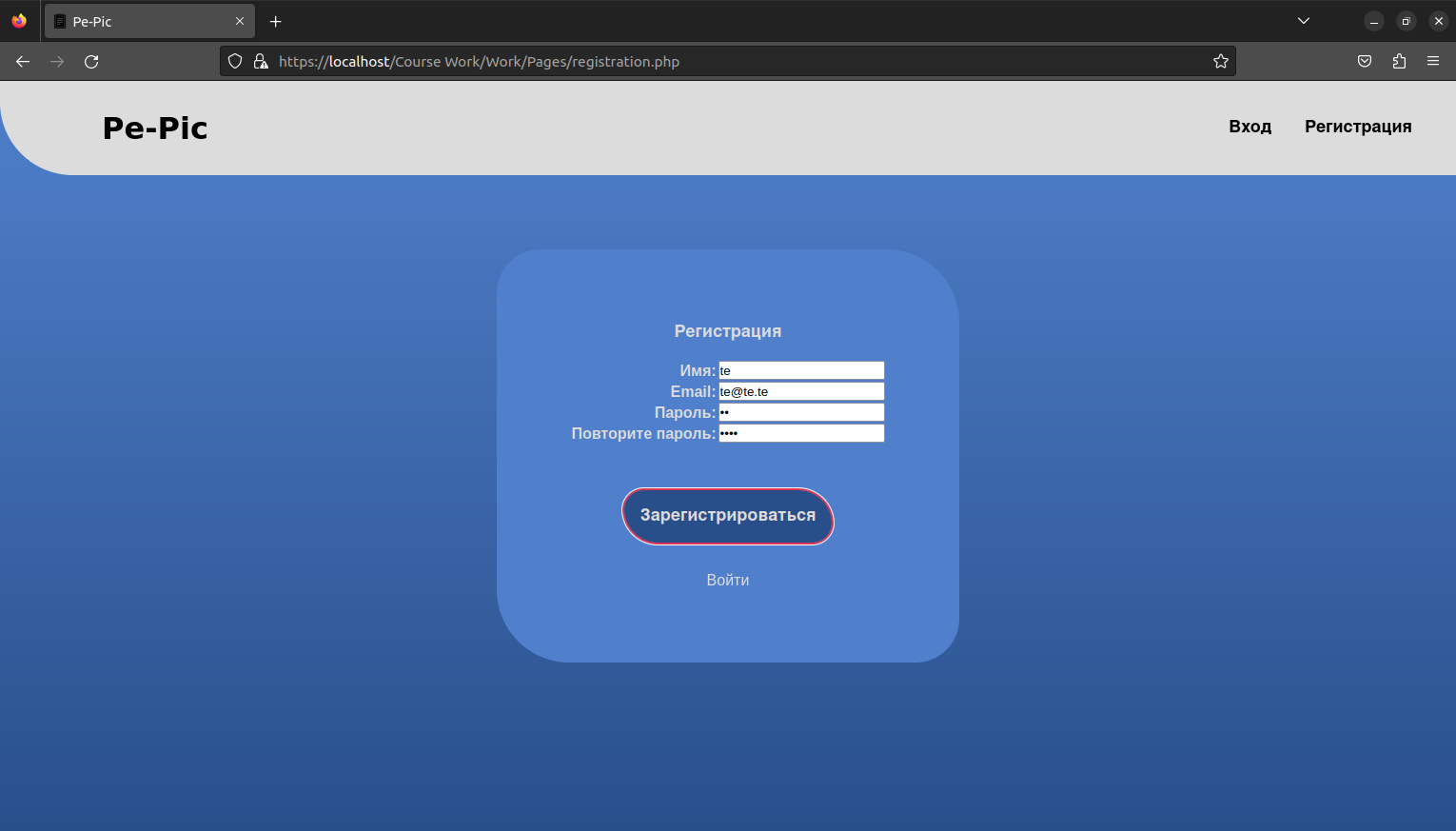


Рисунок 6.2.1. Неверное повторение пароля при регистрации. 1

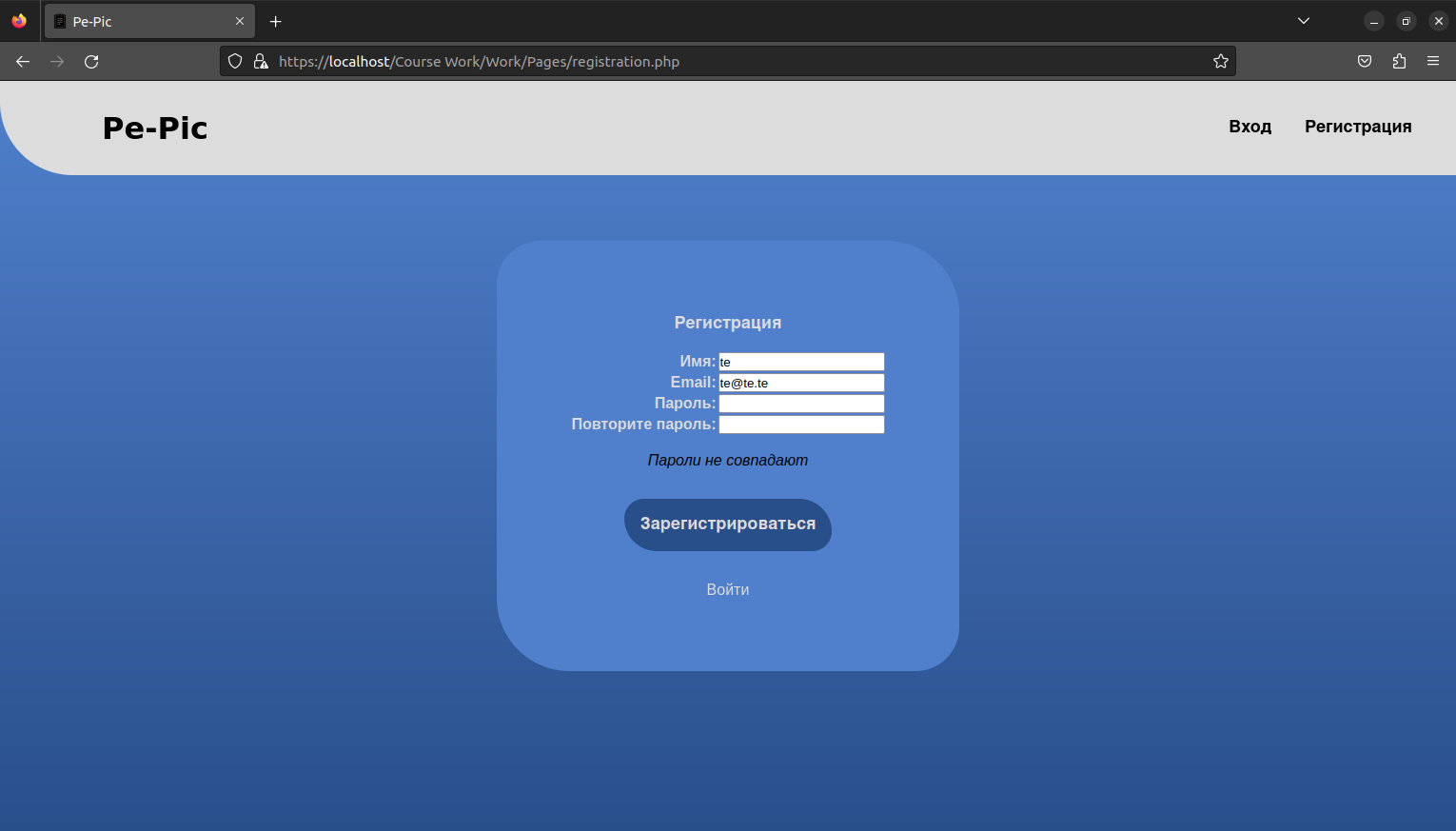


Рисунок 6.2.2. Неверное повторение пароля при регистрации. 2

Главная задача сайта: возможность зарегистрированным пользователям добавлять, редактировать и удалять шутки в виде «вопрос-ответ». Пример данного функционала изображён на рисунках 7.1-3.

Код, реализующий вышеуказанную возможность для пользователя, изображён на рисунках 7.4 и 7.5.1, 7.5.2;

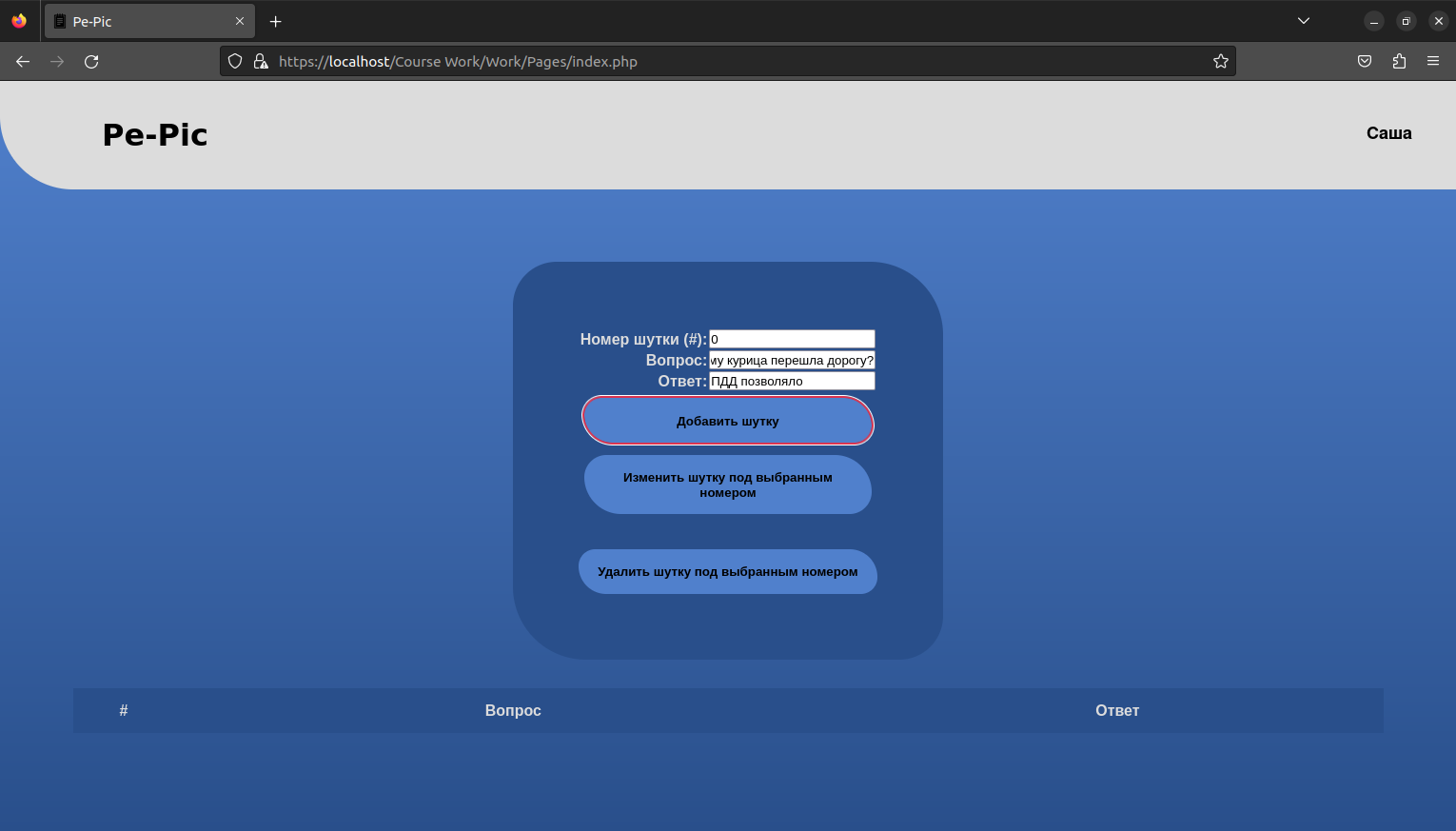


Рисунок .1.1. Добавление шутки. 1

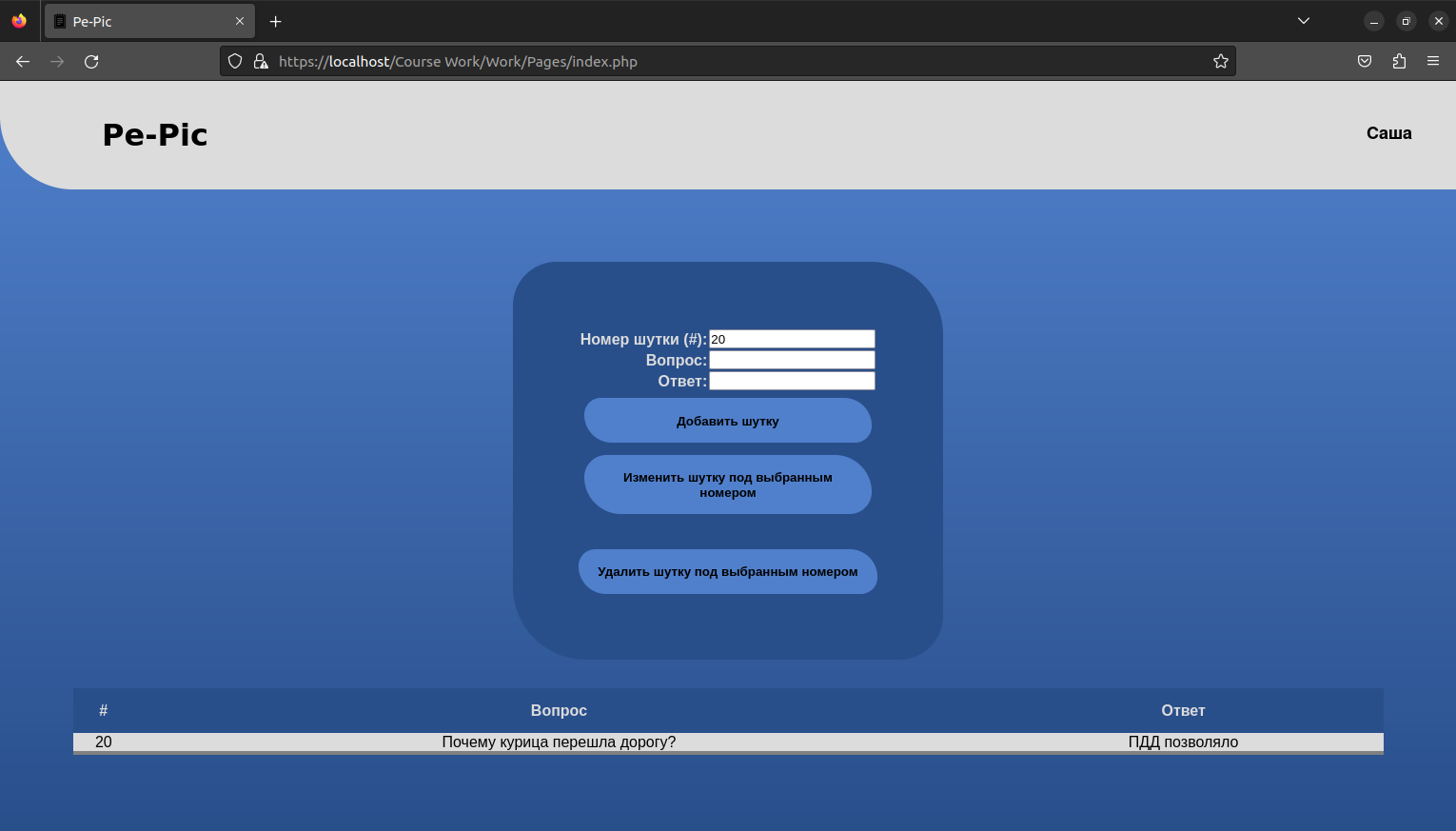


Рисунок 7.1.2. Добавление шутки. 2

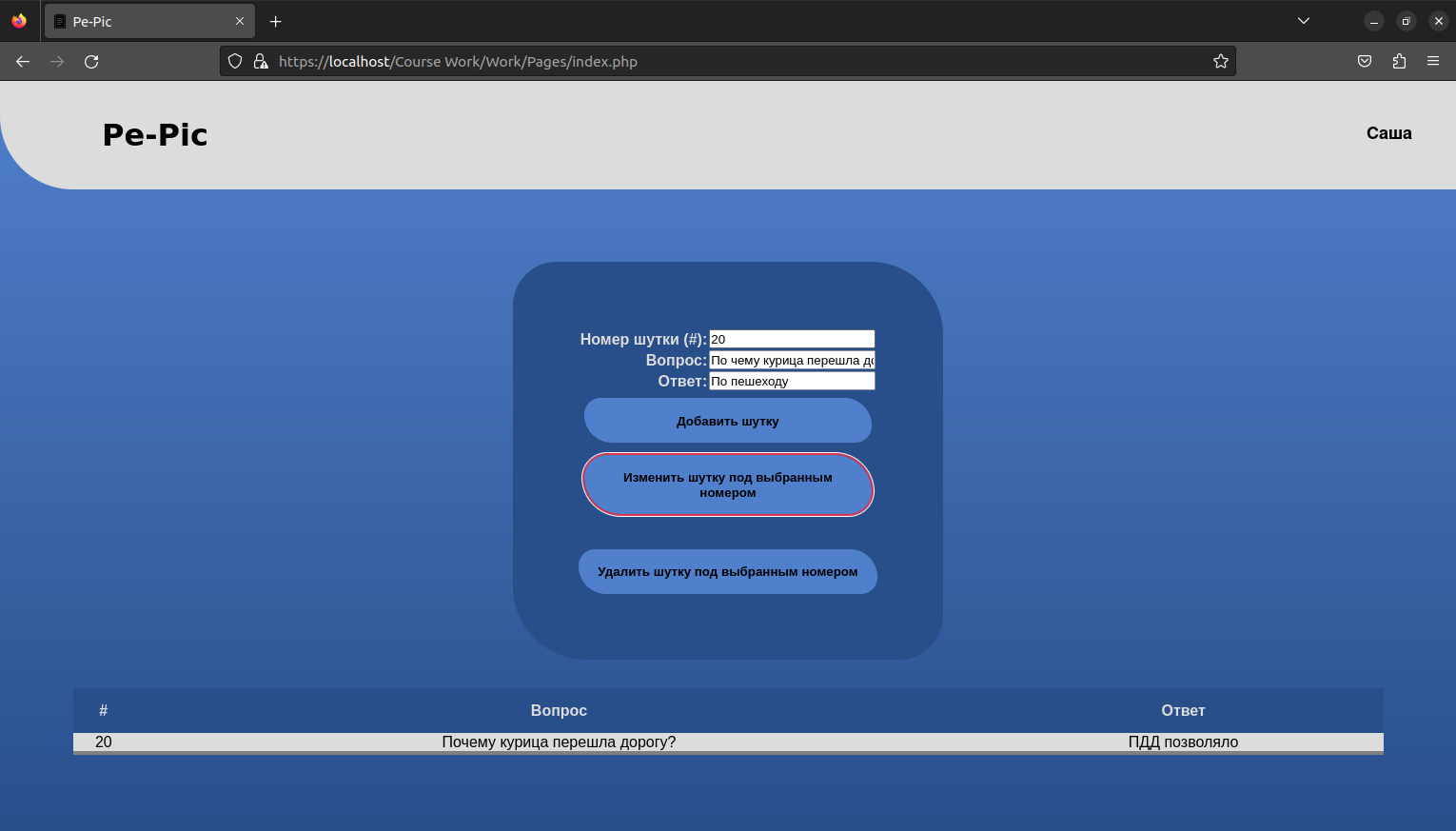


Рисунок 7.2.1. Изменение шутки. 1

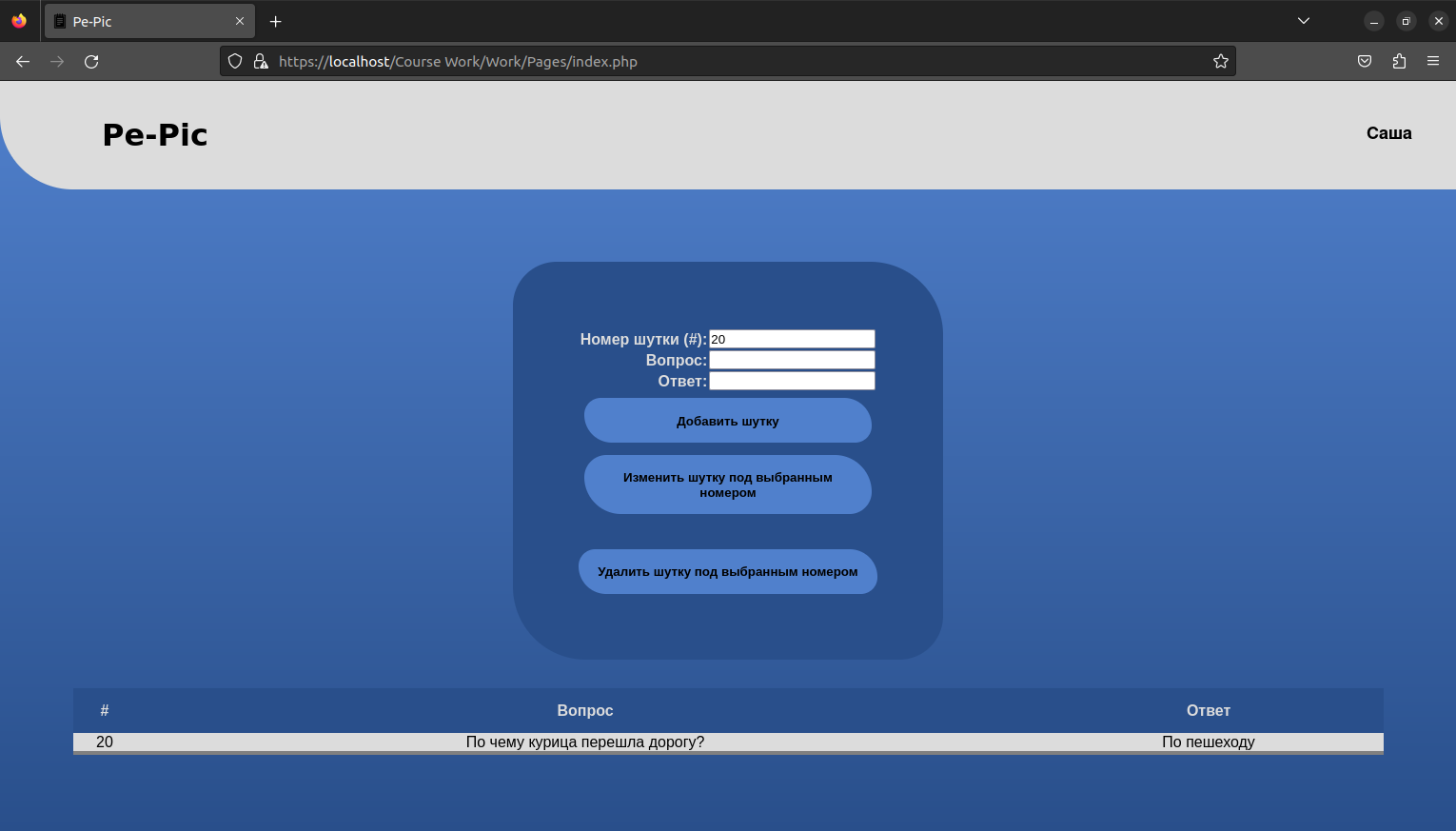


Рисунок 7.2.2. Добавление шутки. 2

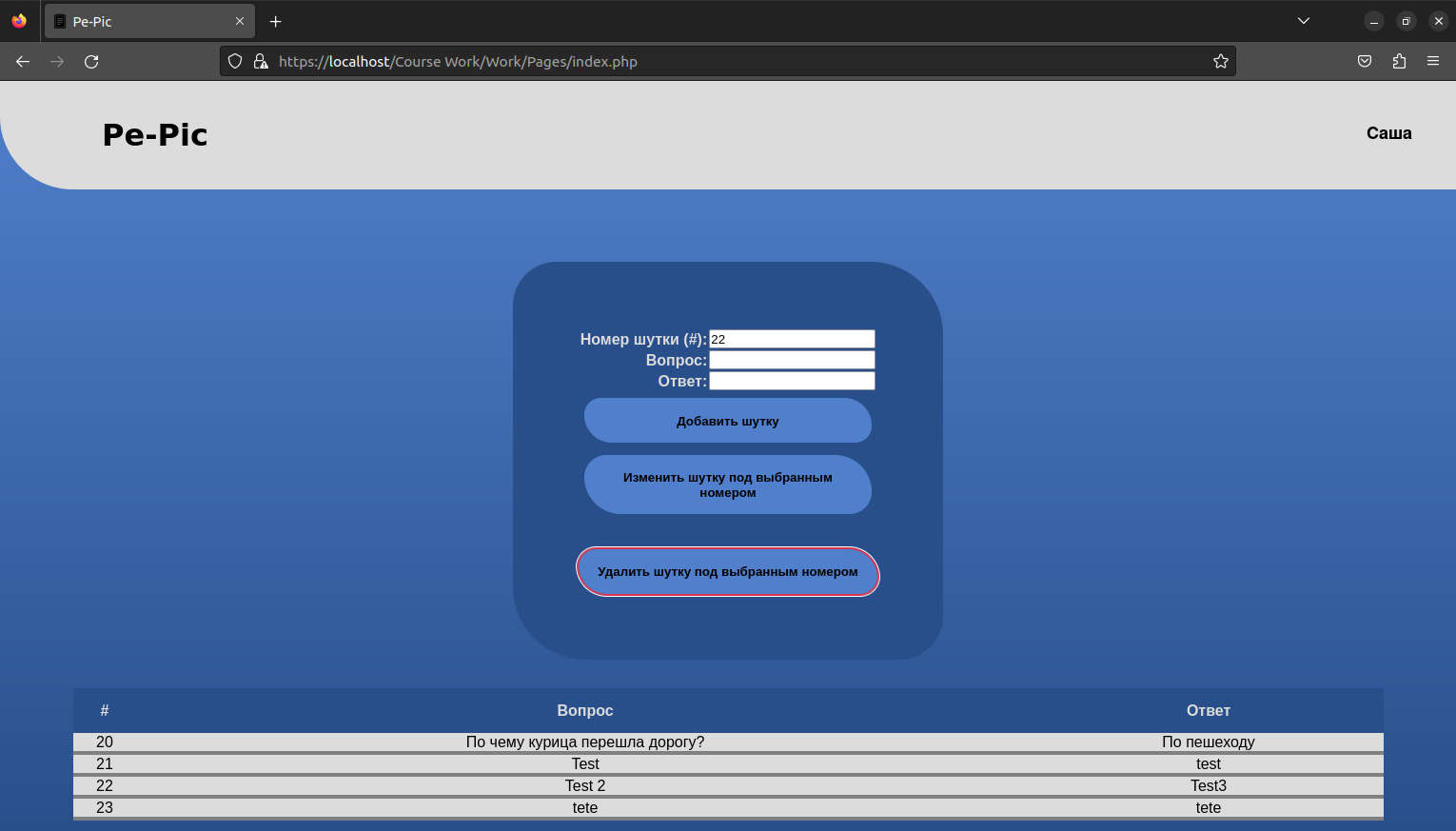


Рисунок 7.3.1. Удаление шутки. 1

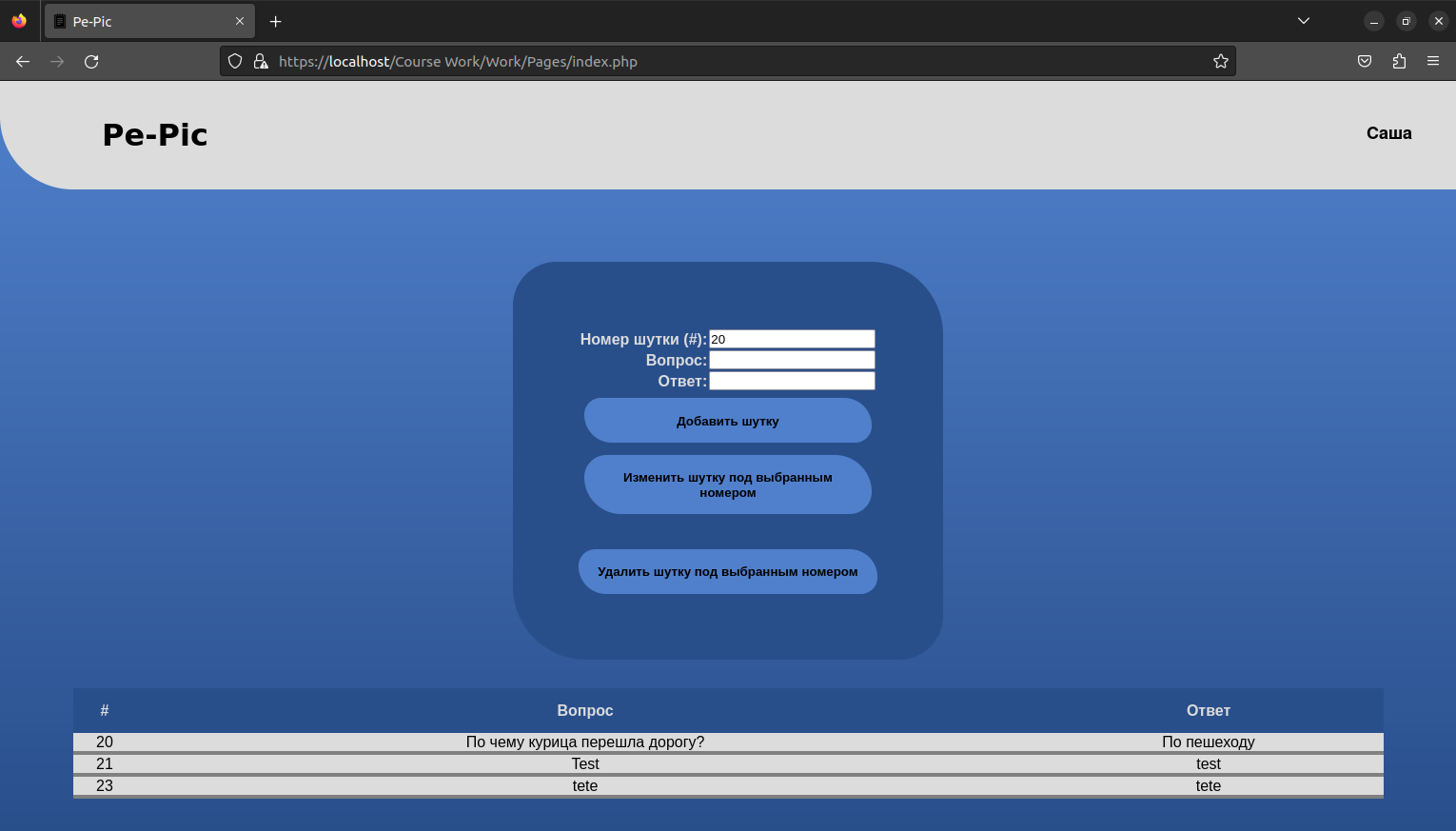


Рисунок 7.3.2. Удаление шутки. 2

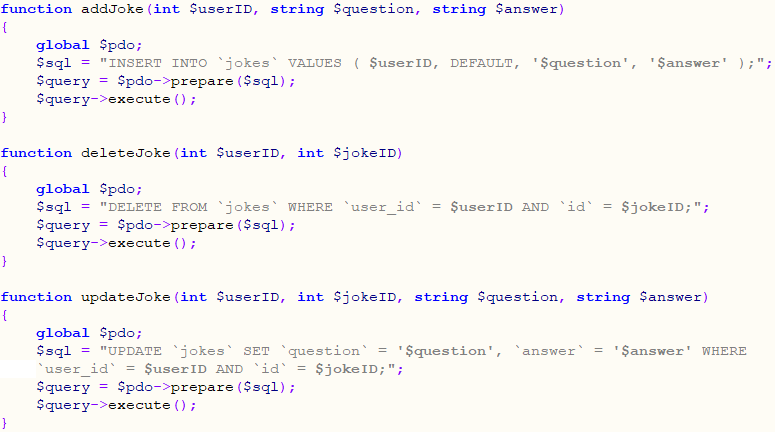


Рисунок 7.4. Фрагменты кода, реализующего добавление, удаление и изменение шутки в БД (файл BD.php)



Рисунок 7.5.1. Фрагмент кода из файла index.php, который реализует получение всех шуток авторизованного пользователя

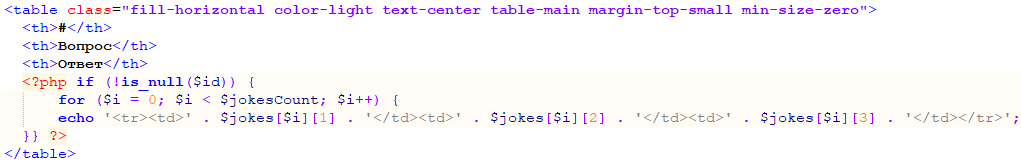


Рисунок 7.5.2. Фрагмент кода из файла index.php, который реализует вывод всех шуток авторизованного пользователя

Глава 4. Руководство пользователя

При первом входе на сайт открывается данная страница (рисунок ):

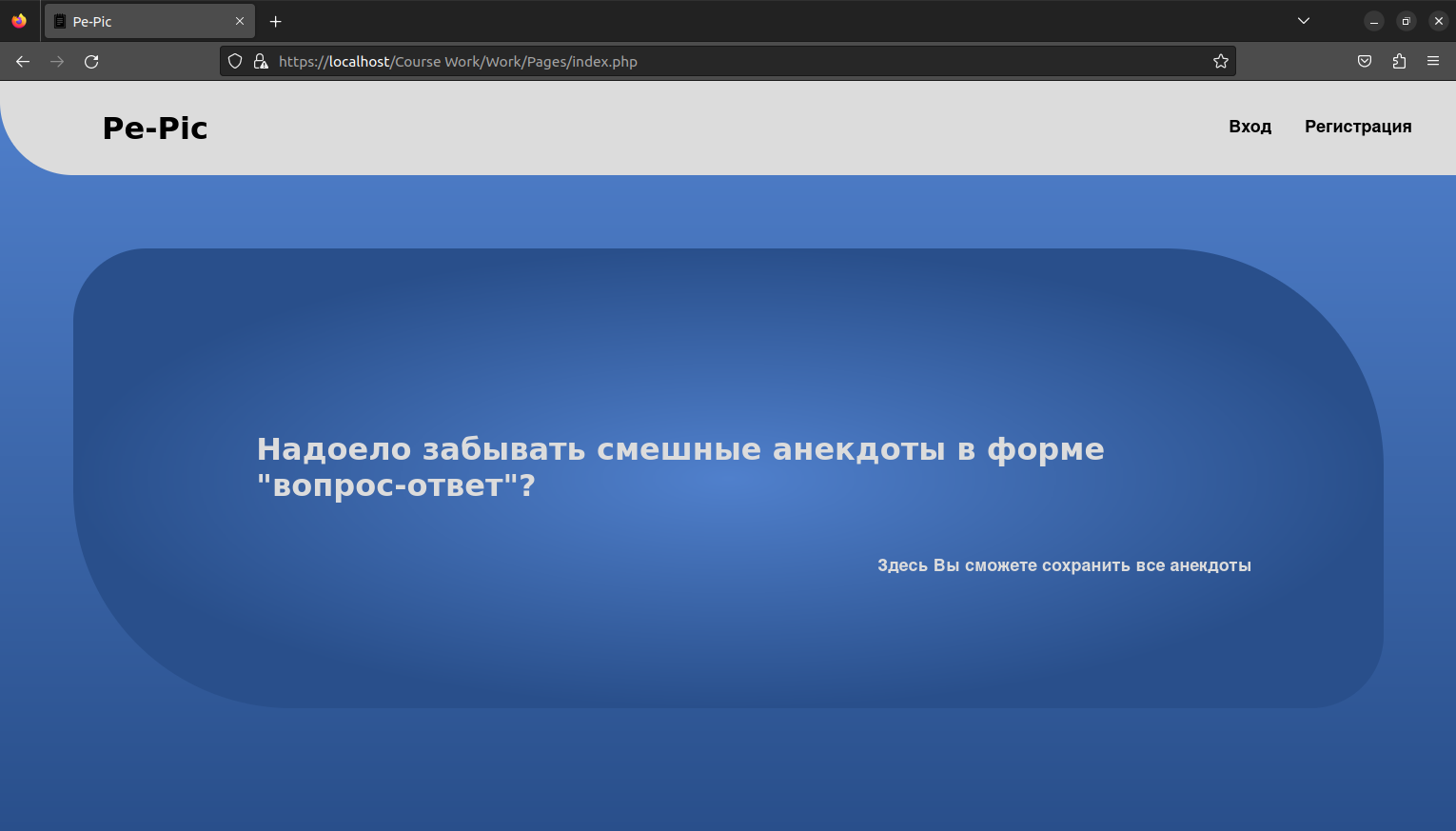


Рисунок . Главная страница сайта, при входе в первый раз

Страница предоставляет возможность посетить все страницы из списка навигации для просмотра данных.

Нажав на кнопку «Регистрация», пользователю предоставиться возможность зарегистрироваться на сайте (рисунок 9), внеся тем самым данные о своём профиле в БД. Также, уже зарегистрировавшись, пользователь сможет авторизоваться (рисунок 10), через соответствую кнопку «Вход», чтобы продолжить работу со своего имеющегося профиля.

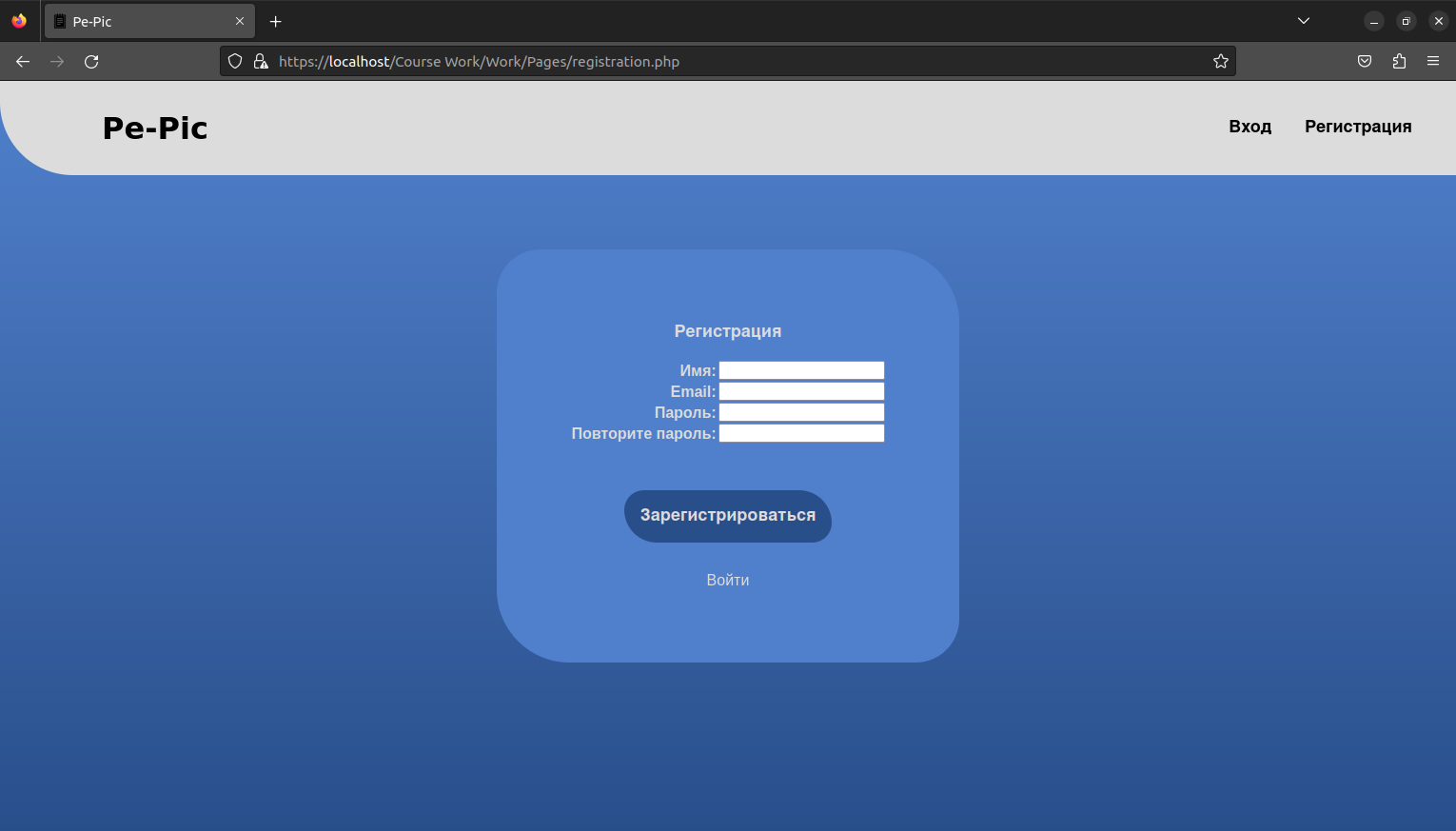


Рисунок . Страница регистрации нового профиля

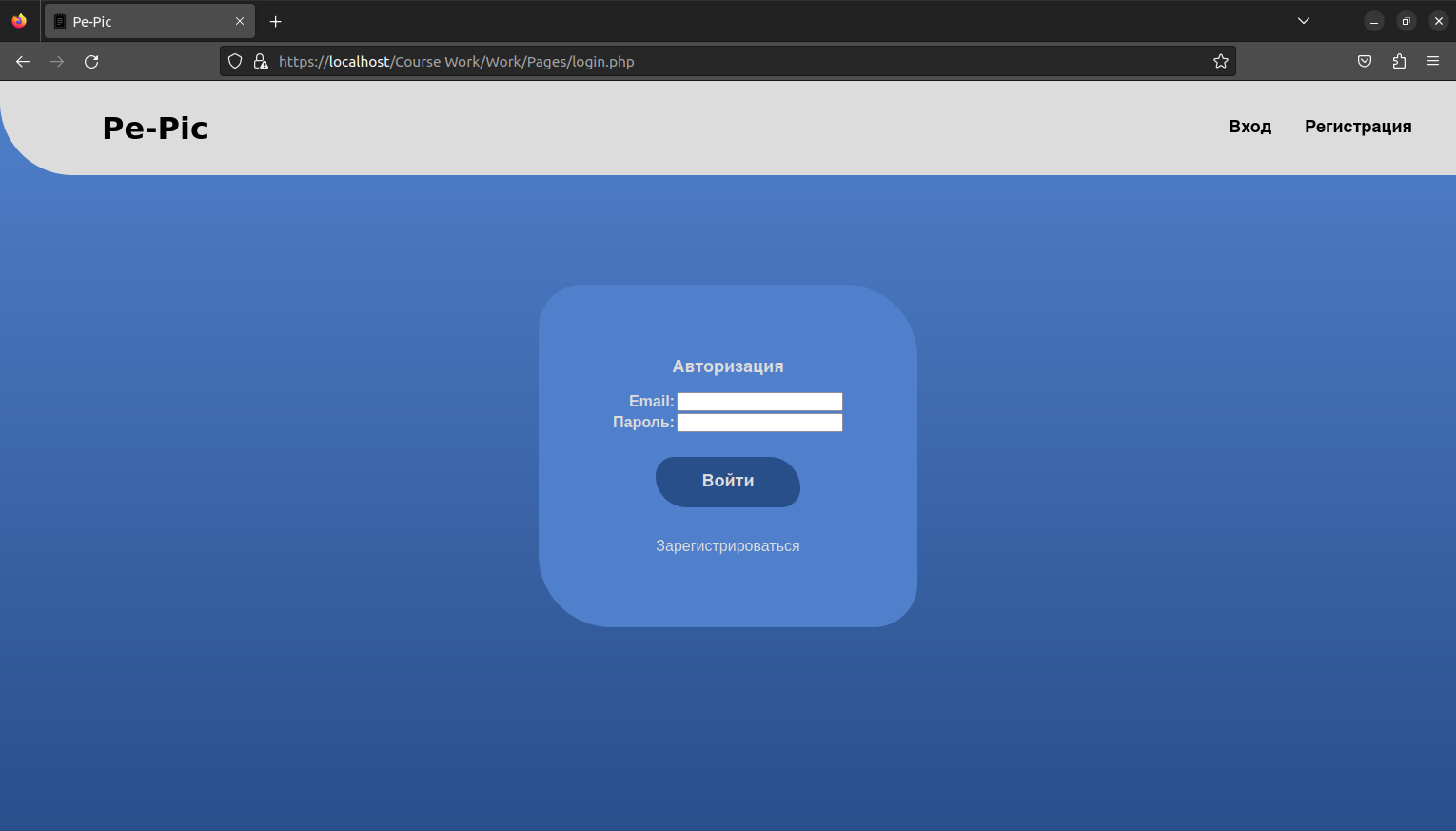


Рисунок . Страница входа в профиль

После успешной авторизации или регистрации пользователя переносит на главную страницу, которая будет иметь другой вид (рисунок 11); Теперь пользователь имеет возможность добавлять, изменять и удалять шутки.

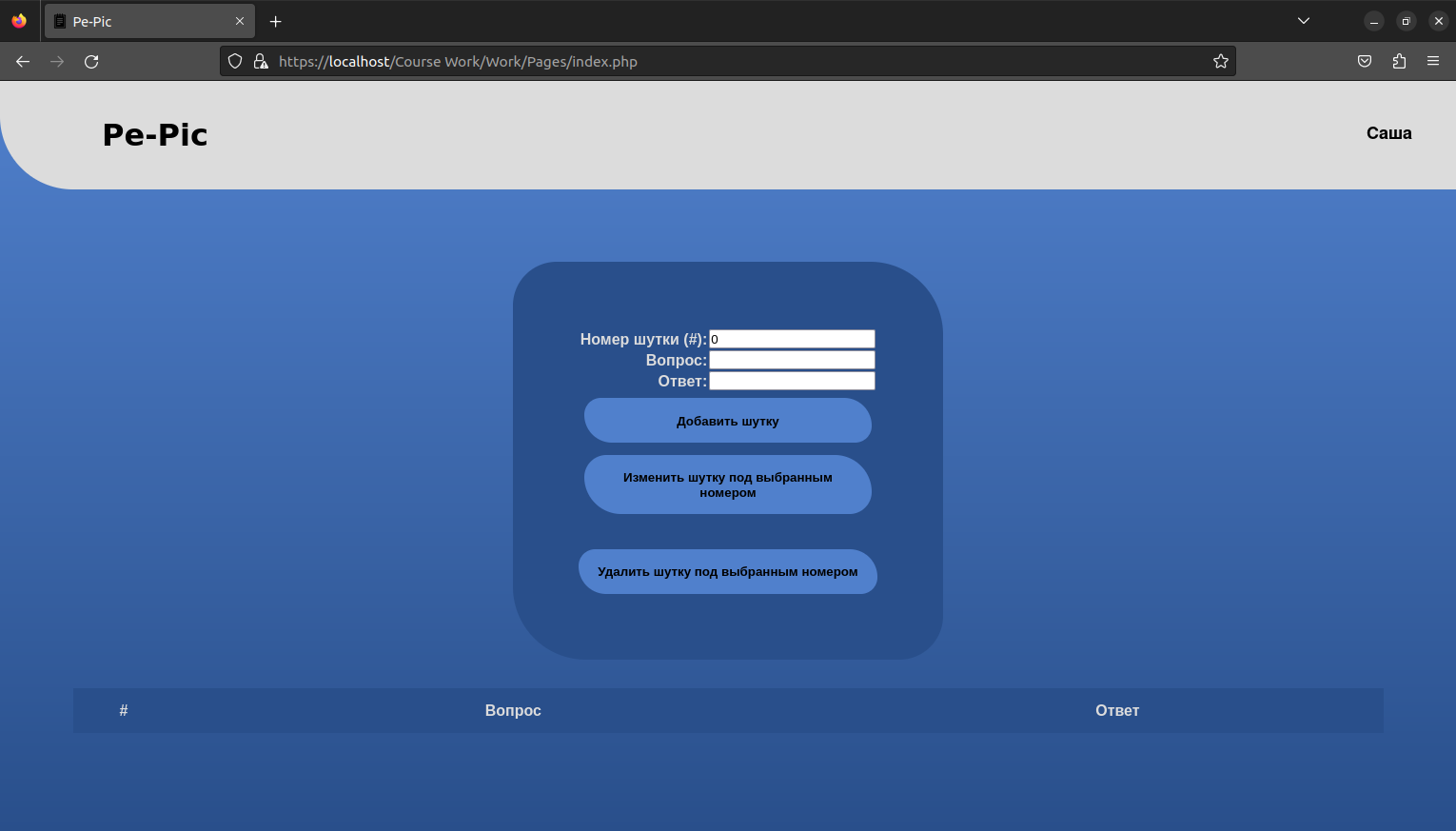


Рисунок . Главная страница после авторизации

Также, в случае необходимости, пользователь может перейти на страницу профиля (рисунок 12) и изменить необходимые данные (имя, адрес электронной почты и пароль). Для удобства пользователя на странице профиля предоставлен функционал полного удаления профиля или выход (деавторизация) из профиля.

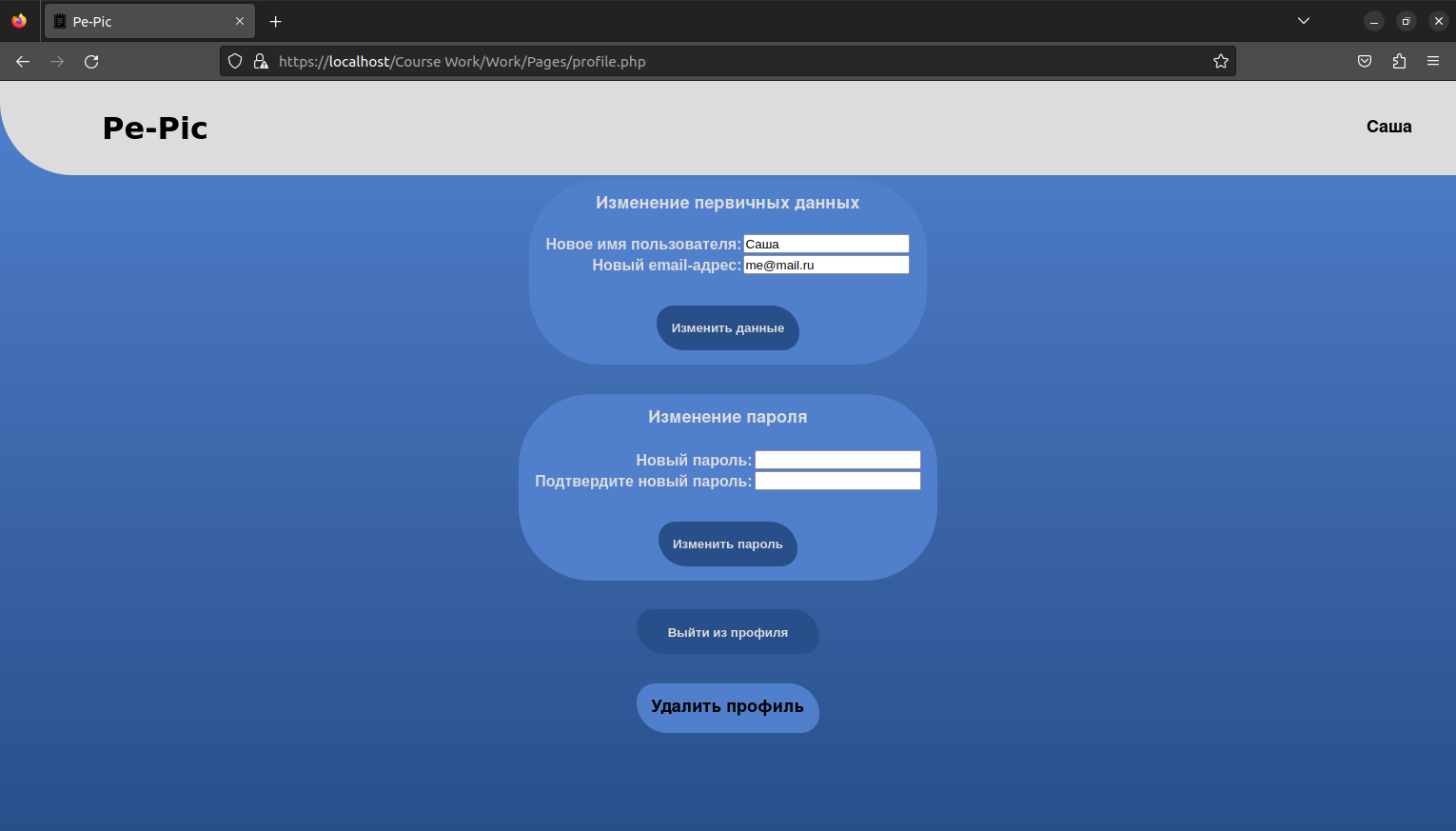


Рисунок . Страница профиля

Для взаимодействия со страницами, пользователь должен воспользоваться полями и кнопками на самих страницах. В случае некорректного ввода данных, пользователь будет оповещён о проблеме вместе с её описанием. Оповещение производится за счёт текста рядом с местами, где пользователь может вызвать проблему. Пример оповещения о проблеме показан на рисунке 13.

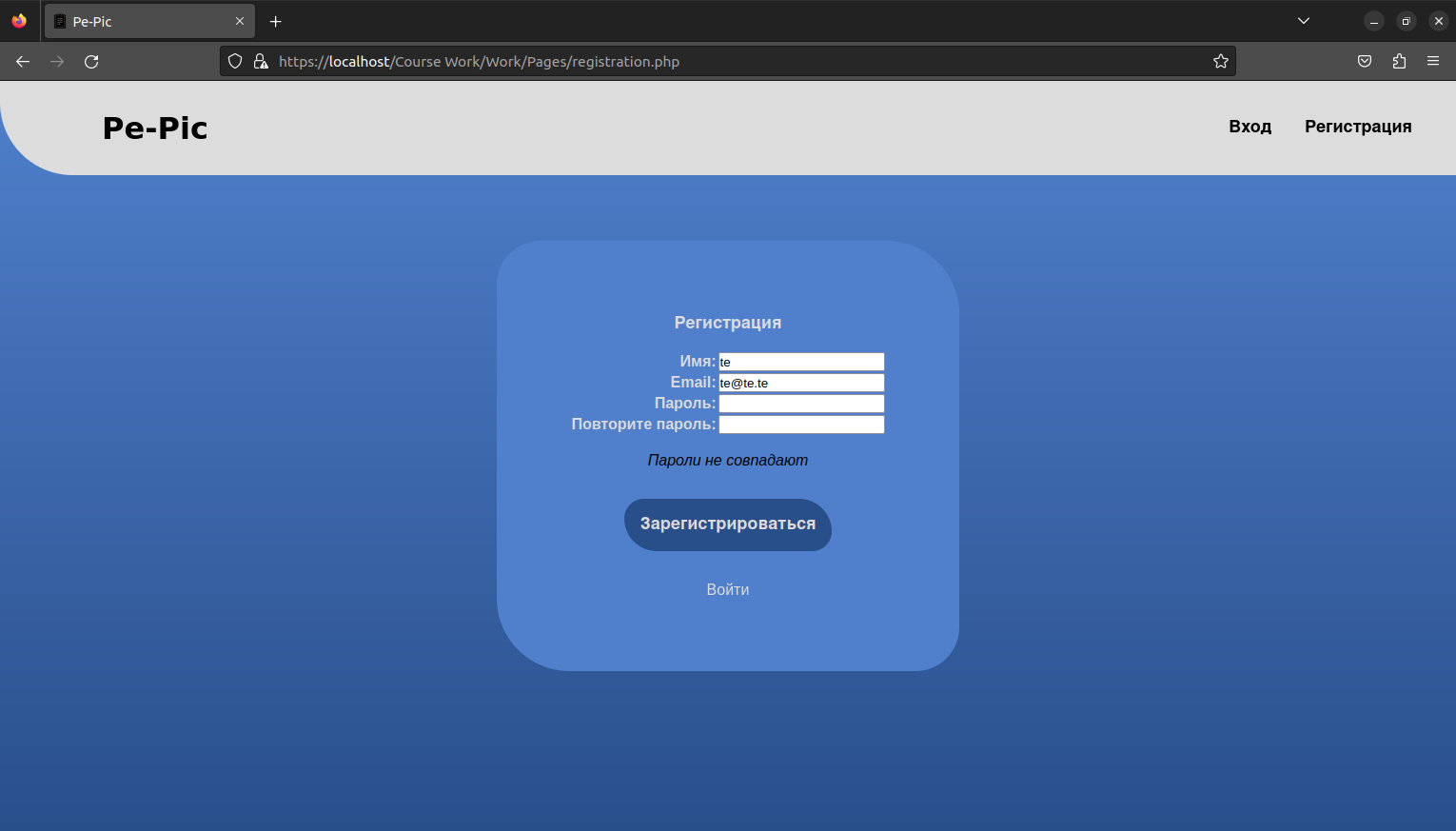


Рисунок . Пример оповещения о проблеме при попытке регистрации нового пользователя

Заключение

В результате выполнения курсовой работы было создано веб-приложение, которое позволяет пользователю регистрироваться, авторизовываться, выходить из профиля, удалять профиль и изменять данные профиля; добавлять шутки, изменять их и удалять их. Все данные хранятся в созданной базе данных с помощью MySQL. Также веб- приложение можно развить дальше в полноценный сайт, дополнив различны функционалом.

В пояснительной записке детально описан процесс создания веб-приложения. Также описана структура базы данных, представлено описание результатов работы приложения и подробно описано руководство пользователя.

Список использованной литературы

* [Электронный ресурс] <https://www.w3schools.com>
* [Электронный ресурс] <https://www.basicweb.ru>
* [Электронный ресурс] <http://htmlbook.ru/>
* [Электронный ресурс] <https://html5book.ru/>
* [Электронный ресурс] <https://www.youtube.com/watch?v=_GTMOmRrqkU>
* [Электронный ресурс] <https://www.youtube.com/playlist?list=PLMB6wLyKp7lVtihXKn-D3yHSLCjaIxliH>
* [Электронный ресурс] <https://selectel.ru/blog/tutorials/how-to-create-databases-in-mysql/>
* [Электронный ресурс] <https://selectel.ru/blog/tutorials/how-to-create-tables-in-mysql/>
* [Электронный ресурс] <https://ru.hexlet.io/courses/php-basics/lessons/intro/theory_unit>