**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И**

**МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Лабораторная работа №6

«Создание формы и представления регистрации и авторизации»

по дисциплине:

«Web-программирование»

Выполнил:

Студент группы БФИ19

Бардюк Д.В.

Москва, 2021

**Задания:**

1. В директорию lab6 скопируйте ваш прошлый проект blog.
2. Создайте шаблон и настройте адрес для отображения формы регистрации;
3. Создайте представление, которое будет обрабатывать поступающие запросы и регистрировать новых пользователей. Не забудьте сделать проверку на то, что отправленные поля не являются пустыми, а введенное имя пользователя уникально;
4. Создайте стили, подключив CSS-файл к шаблону;
5. Добавьте в шапку страницы всех записей и страницы для определенных статей, ссылку на регистрацию в верхнем правом углу (стиль ссылки сделать такой же, как у ссылки “Все статьи” на собственных страницах постов в предыдущих работах).
6. Создайте шаблон и настройте адрес для отображения формы авторизации;
7. Создайте представление, которое будет обрабатывать поступающие запросы и авторизовывать зарегистрированных пользователей. Не забудьте сделать проверку на то, что отправленные поля не являются пустыми, а введенные имя пользователя и пароль соответствуют одному из зарегистрированных аккаунтов;
8. Создайте стили, подключив CSS-файл к шаблону;
9. Загрузите ваш проект на любой гит-репозиторий (GitHub, GitLab, Google Code, Bitbucket и т.п.).

**2. Ход работы**

Создаём новую директорию для лабораторной работы №6 и копируем в неё проект blog из прошлой лабораторной работы. Добавим в файл urls.py новые пути, как показано на рисунке 1.

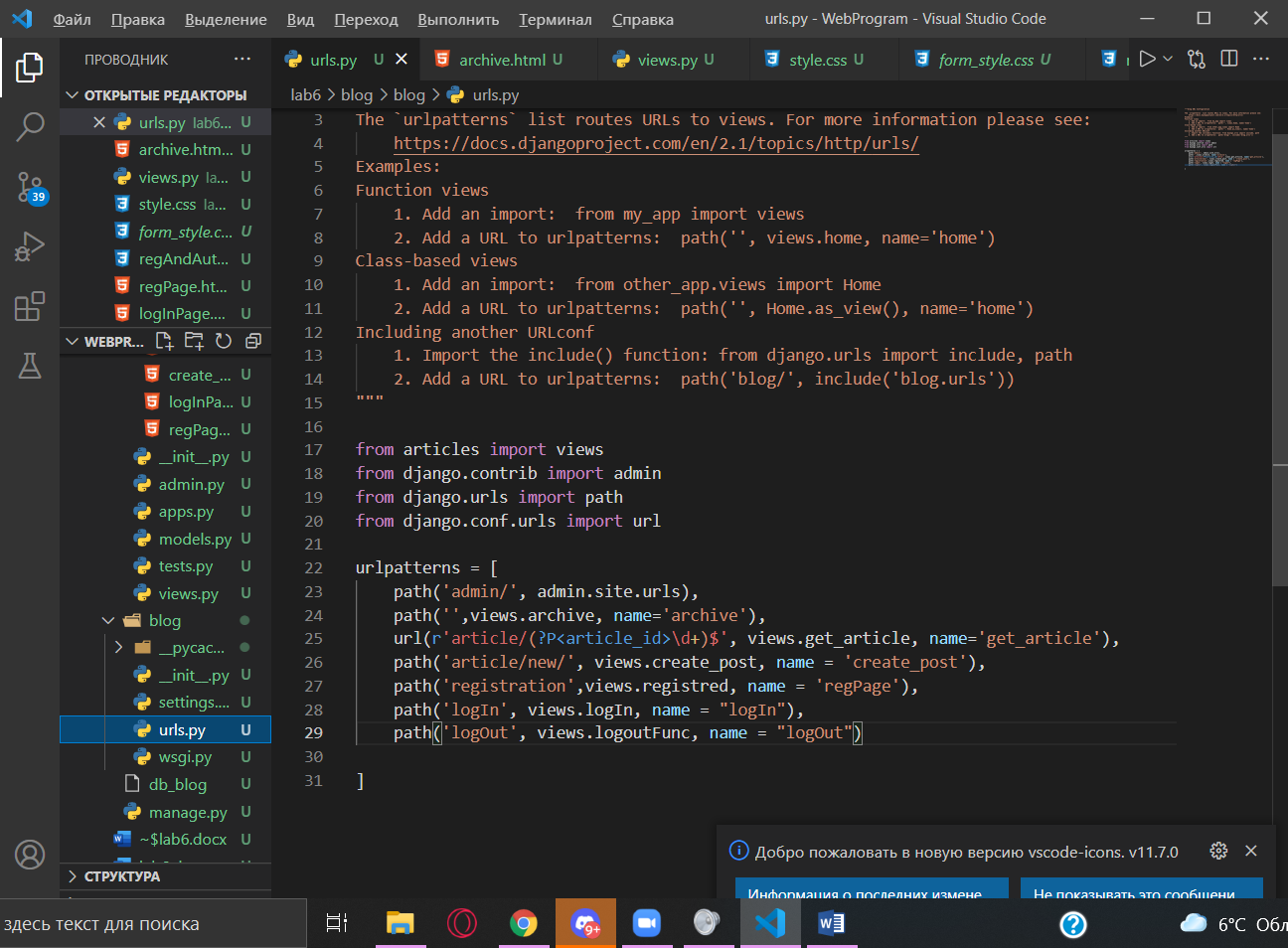


Рисунок 1 – Файл urls.py

Создадим в файле views.py новую функцию registred (рисунок 2) , которая должна будет отображать форму с регистрацией пользователя, и переводить его на форму авторизации в случае успешной регистрации.

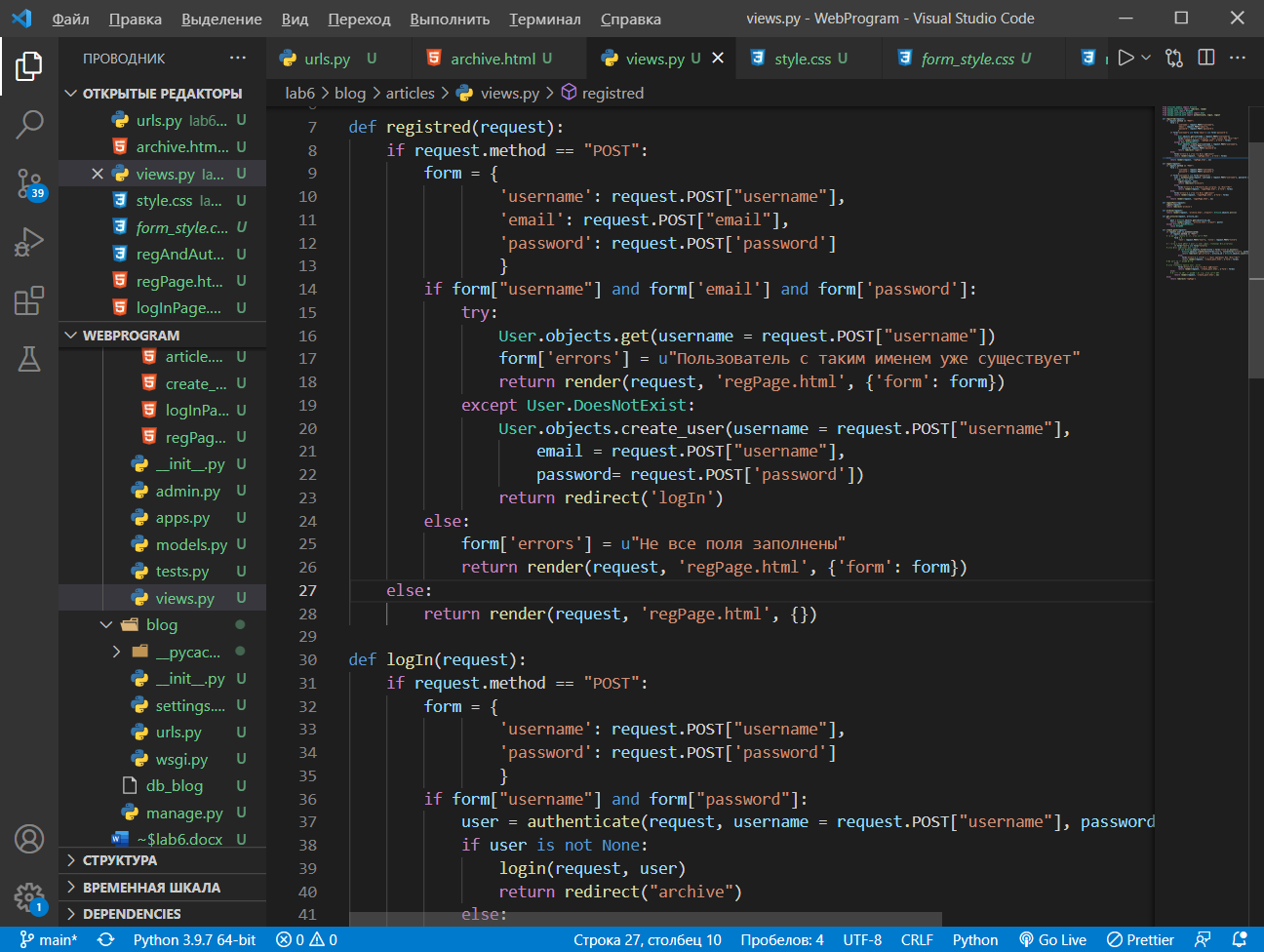


Рисунок 2 – Функция registred()

Создадим файл regPage.html как на рисунке 3.

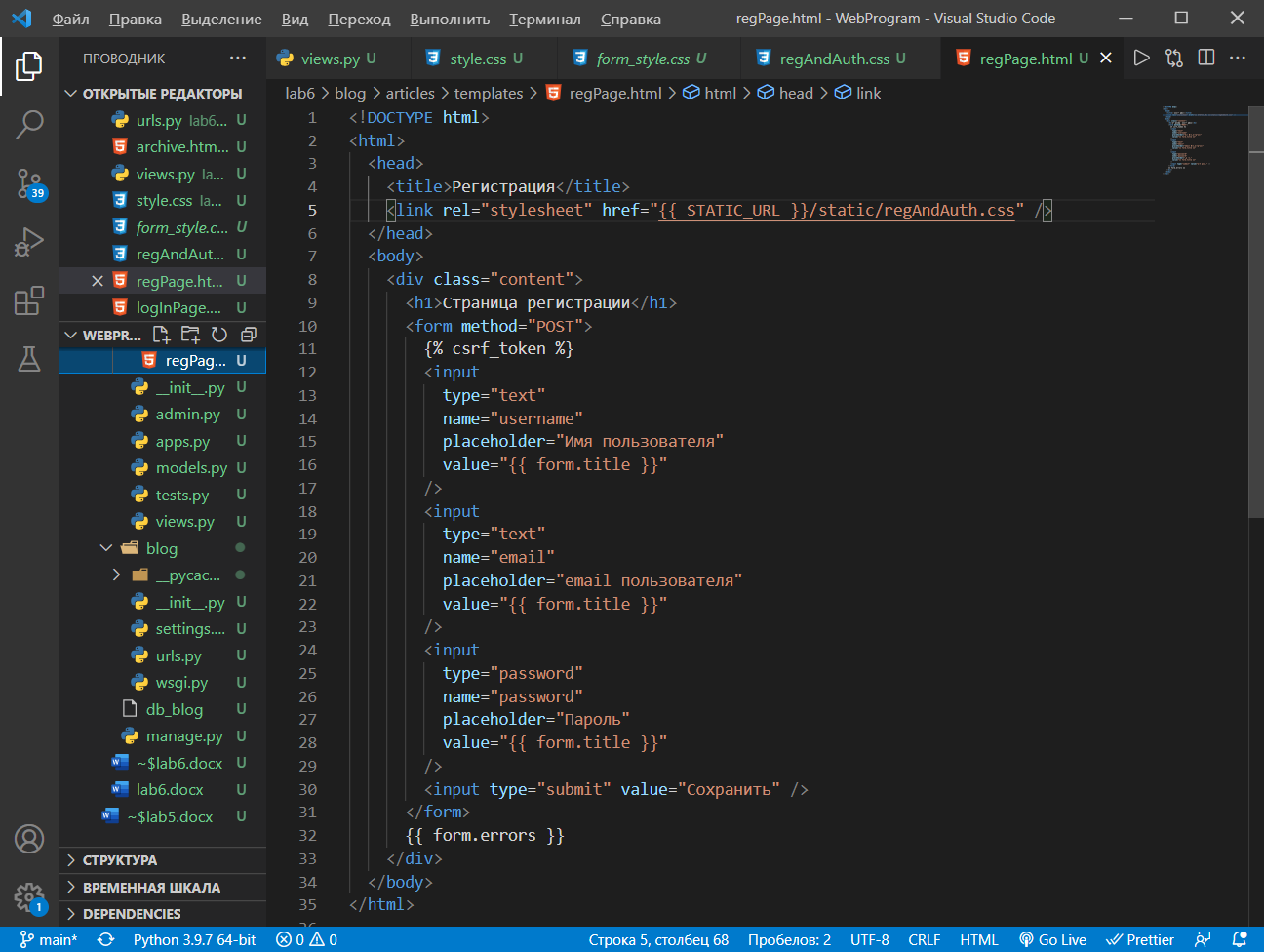


Рисунок 3 – Файл regPage.html

Затем добавим функцию logIn, которая будет выполнять авторизацию пользователя. Результат представлен на рисунке 4. И создадим файл представления logInPage, как показано на рисунке 5.

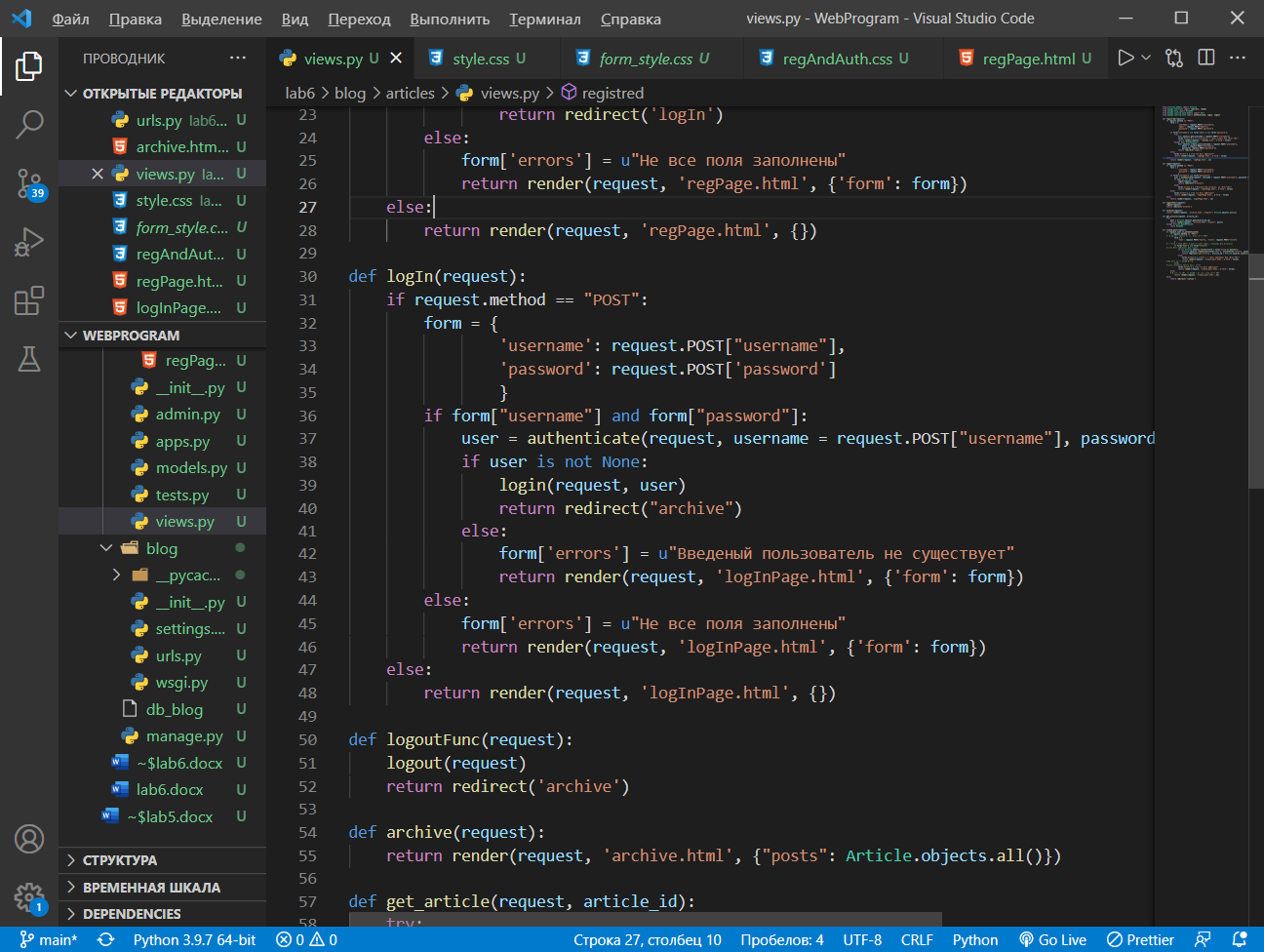


Рисунок 4 – Функция logIn

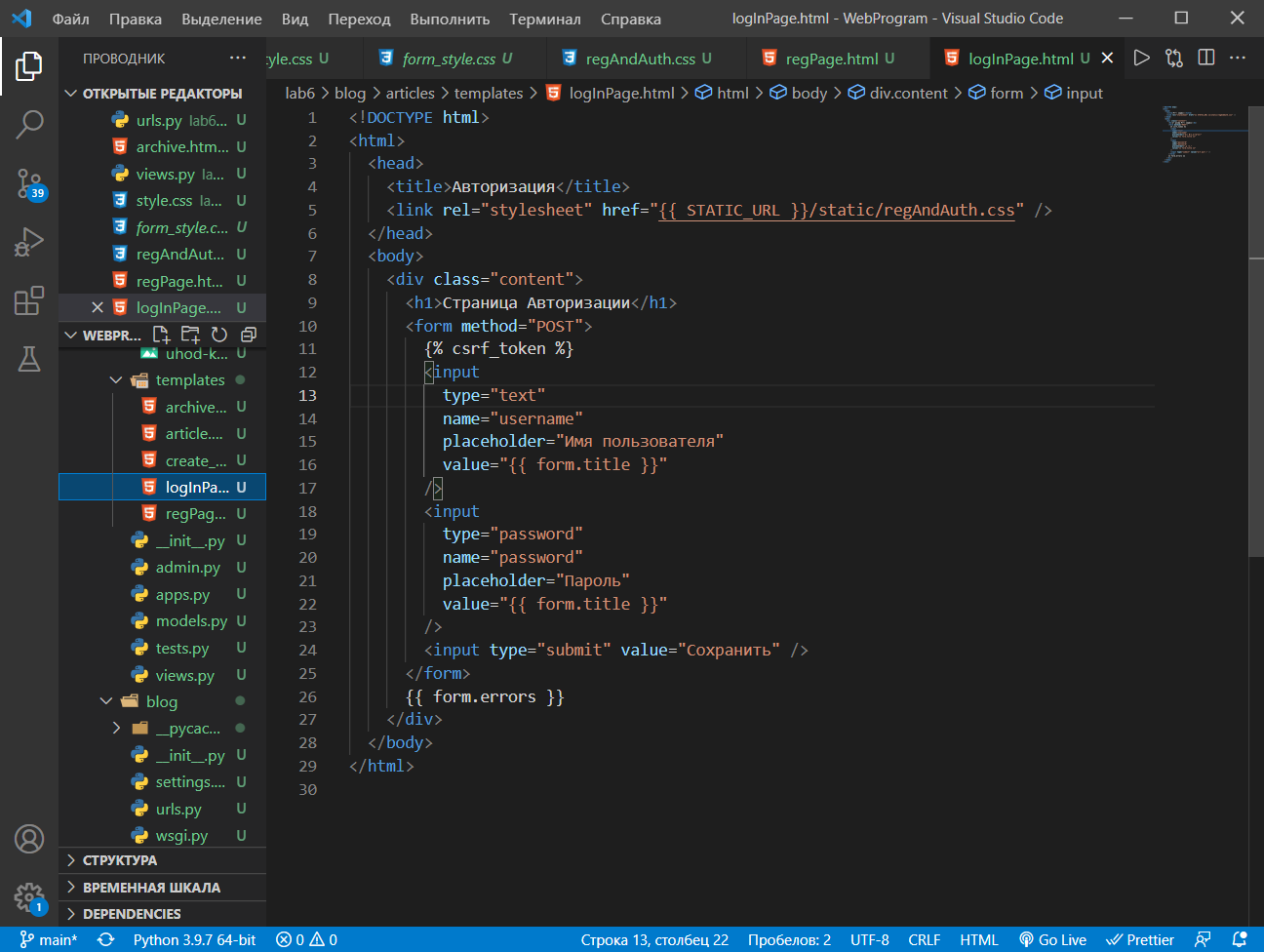


Рисунок 5 – Файл logInPage

Создадим так же функцию logOut для выхода из аккаунта, как показано на рисунке 6.

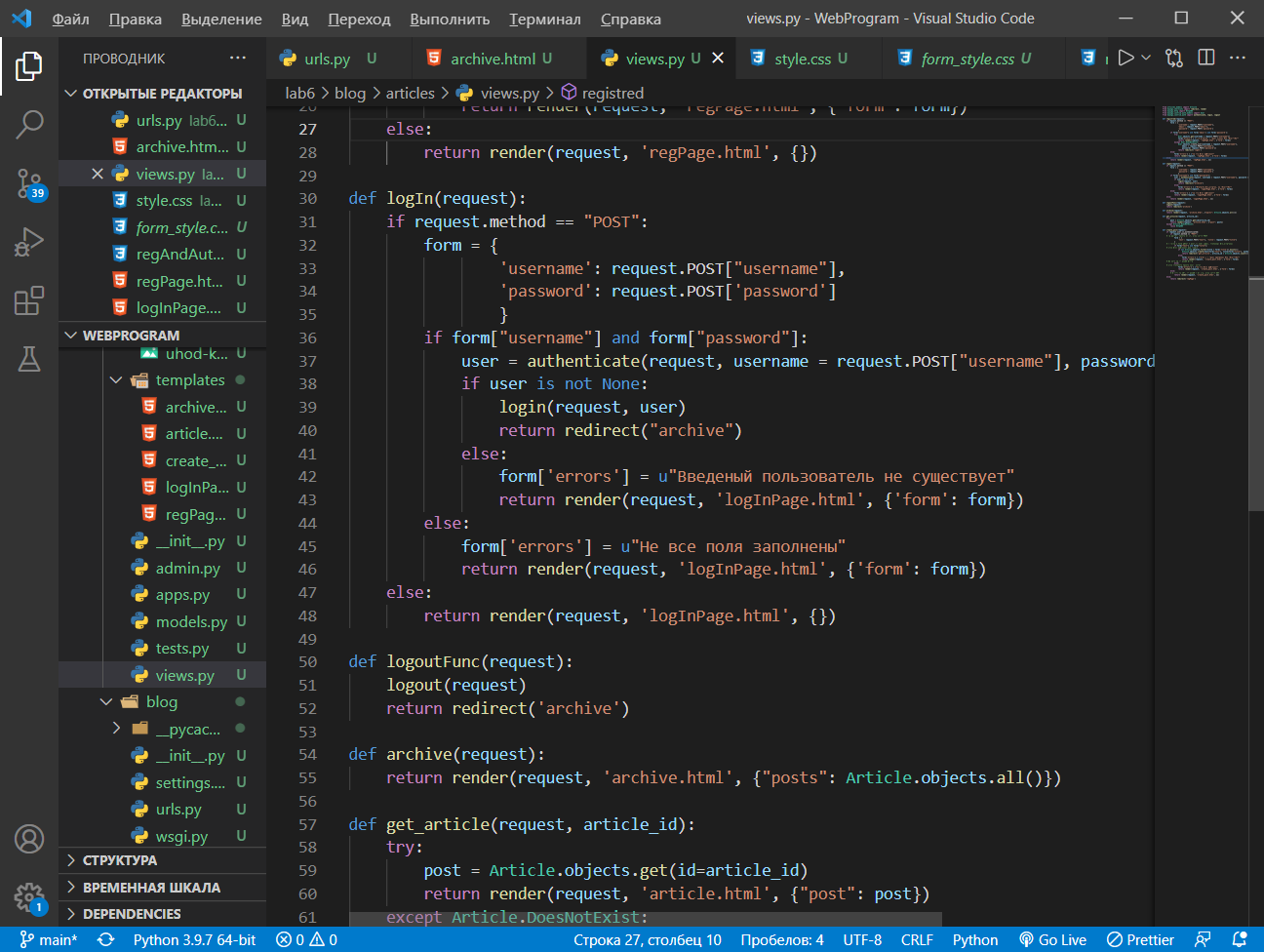


Рисунок 6 – Функция logout

Теперь добавим стилей нашим формам(листинг 1).

**Листинг 1**

body {

  background: #1abc9c;

  font-family: Tahoma, Arial, sans-serif;

  color: #ffffff;

  width: 960px;

  margin: 0 auto;

}

.content {

  background-color: rgba(133, 187, 173, 0.877);

  margin-top: 50px;

  padding: 20px;

  border-radius: 10px;

  border-color: white;

  border: 1px solid;

}

.content h1 {

  text-align: center;

}

form {

  height: 16rem;

  display: flex;

  flex-direction: column;

  justify-content: space-evenly;

  align-items: center;

}

input[type="text"] {

  height: 1, 2rem;

  width: 200px;

  text-align: center;

  border-radius: 10px;

  padding: 5px;

}

textarea {

  height: 100px;

  width: 300px;

  border-radius: 10px;

  padding: 5px;

}

input[type="submit"] {

  color: white;

  border-radius: 10px;

  height: 2rem;

  width: 8rem;

  padding: 5px;

  background-color: rgb(32, 182, 32);

}

Добавим ссылки на формы авторизации и регистрации, а также на функцию выхода из аккаунта. Результат представлен на рисунке 7.

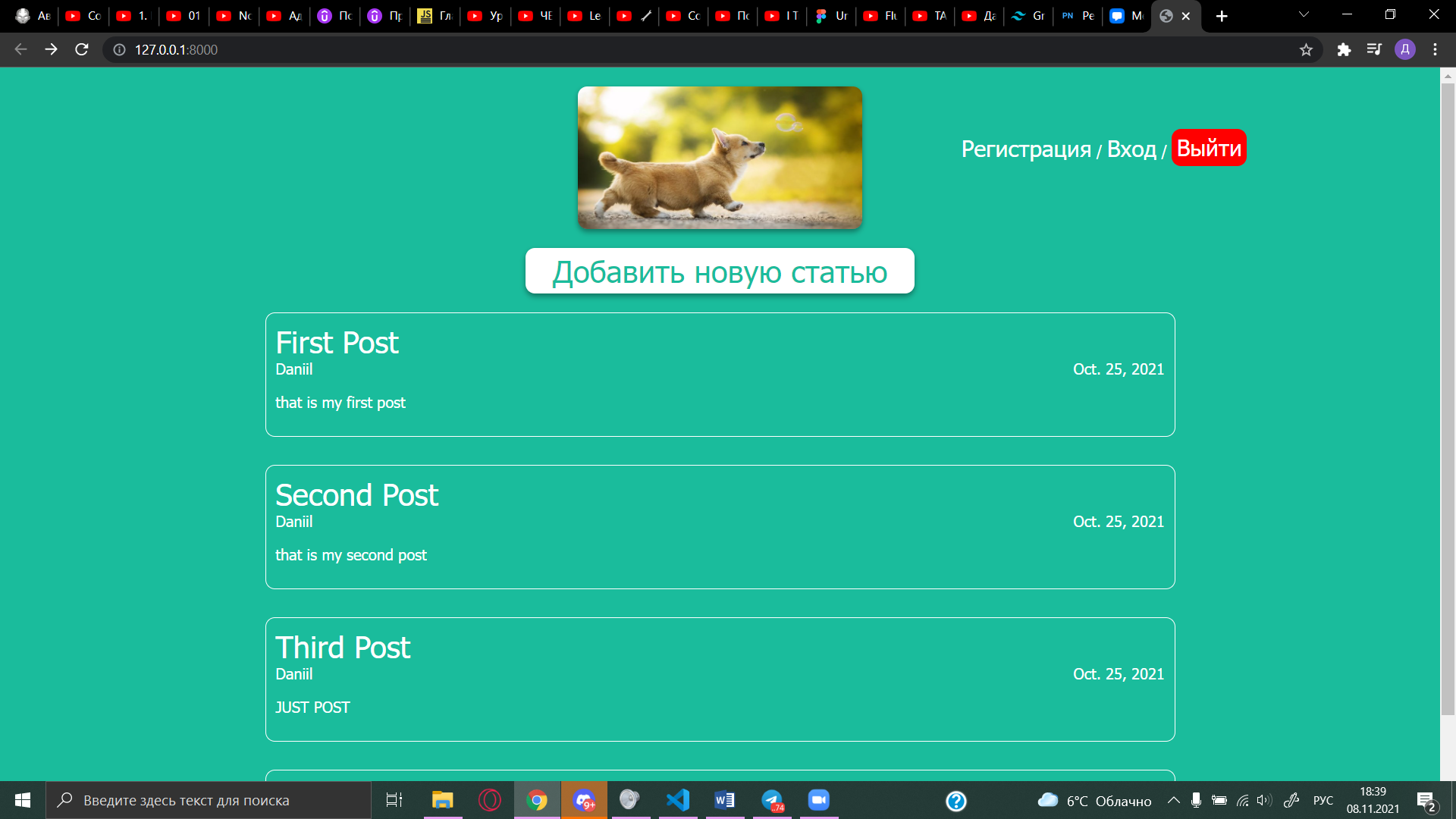


Рисунок 7 – Основная страница с ссылками.

Стилизованные формы представлены на рисунке 8 и 9.

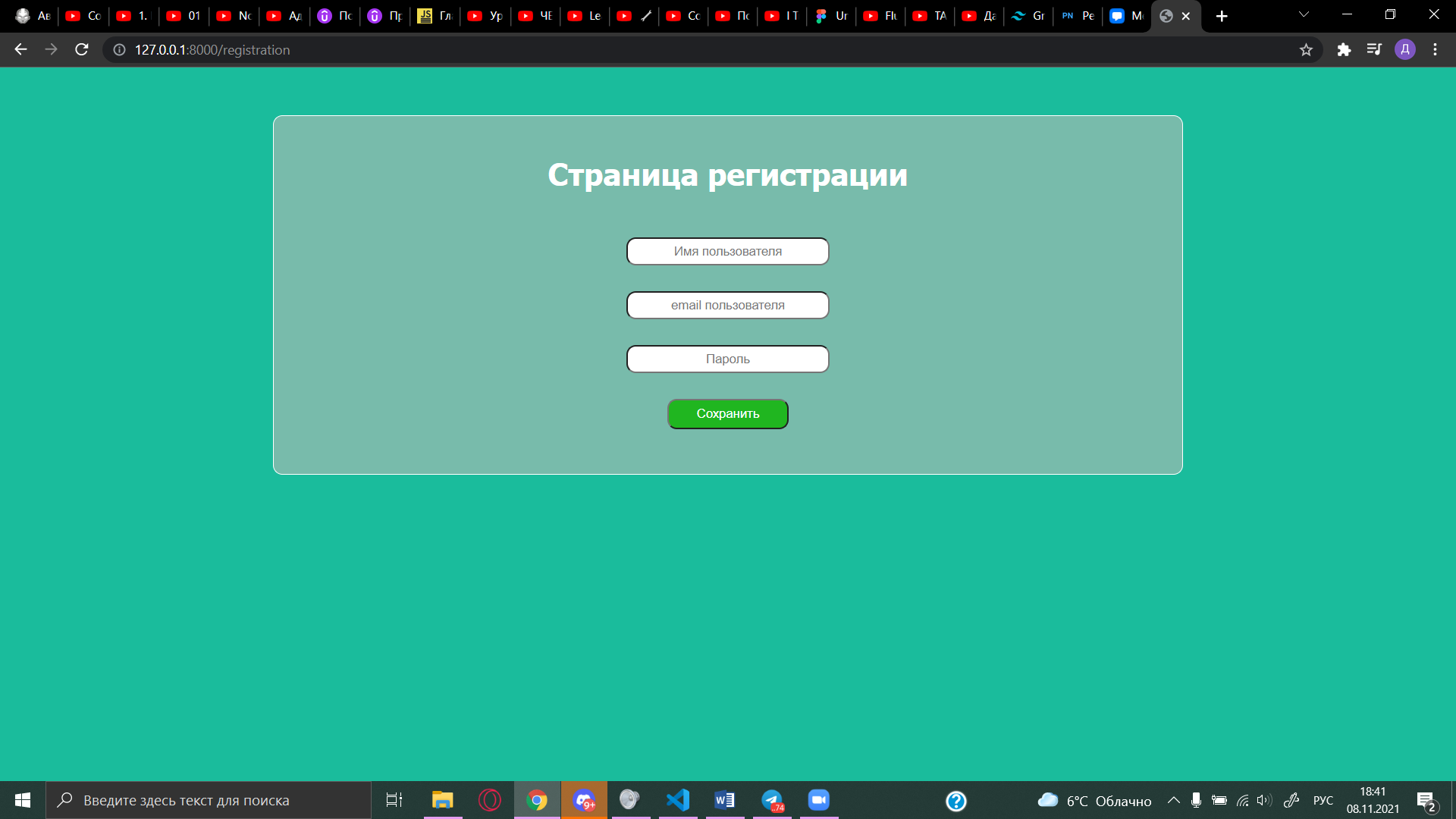


Рисунок 8 – Страница регистрации

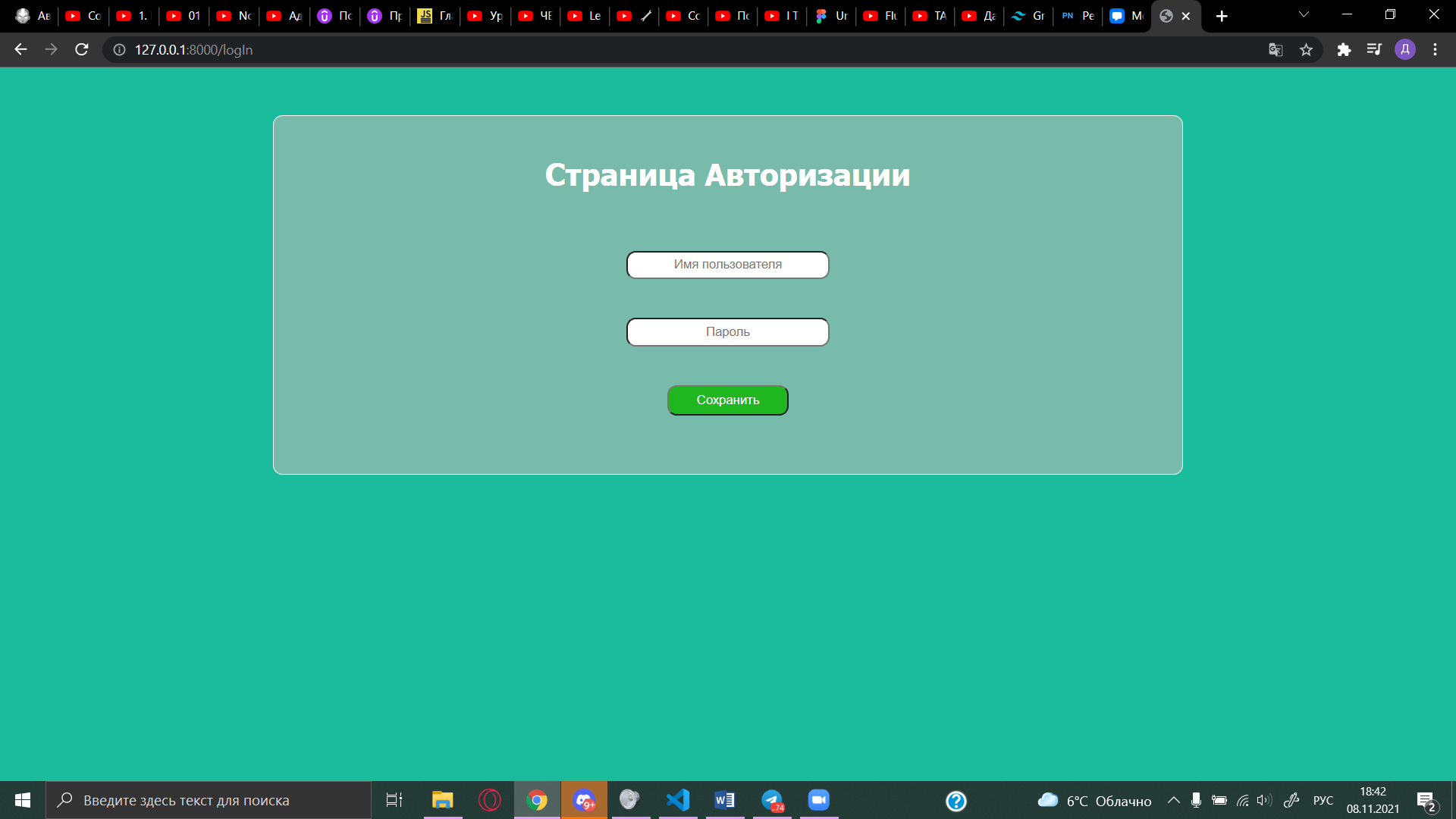


Рисунок 9 – Страница авторизации

**Вывод:** я установил язык программирования «Python» и фреймворк Django, ознакомился с синтаксисом и выполнил задания.