Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторным работам №7 и №8**

Дисциплина: «Разработка профессиональных приложений»

Связи в БД. CRUD. Регистрация и авторизация

Вариант № 14

Выполнил:

студент группы ИВТАПбд-21

Мухаметзянов Т. А.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

Для предложенного варианта заданий необходимо реализовать веб-сайт в среде Django, удовлетворяющий следующим требованиям:

1. Количество таблиц в БД - не менее 5.

2. Должны быть реализованы 3 типа связей: one-to-one, one-to-many, man-to-many.

3. Таблицы создаются с использованием моделей в Django и миграций.

4. Как минимум, для 1 сущности должны быть реализованы CRUD операции (Create, read, update, delete) + представление списка сущностей.

5. Остальные таблицы должны быть доступны для редактирования в административной части сайта.

6. Необходимые пять таблиц должны быть разработаны для моделирования предметной области, заданной вариантом. Необходимо придумать эти таблицы, не забывая про здравый смысл.

7. Добавить минимум 2 роли, соответствующих реальным ролям в выбранной предметной области. Добавить пользователей, обладающих этими ролями.

8. Реализовать средства регистрации пользователей, аутенфикации.

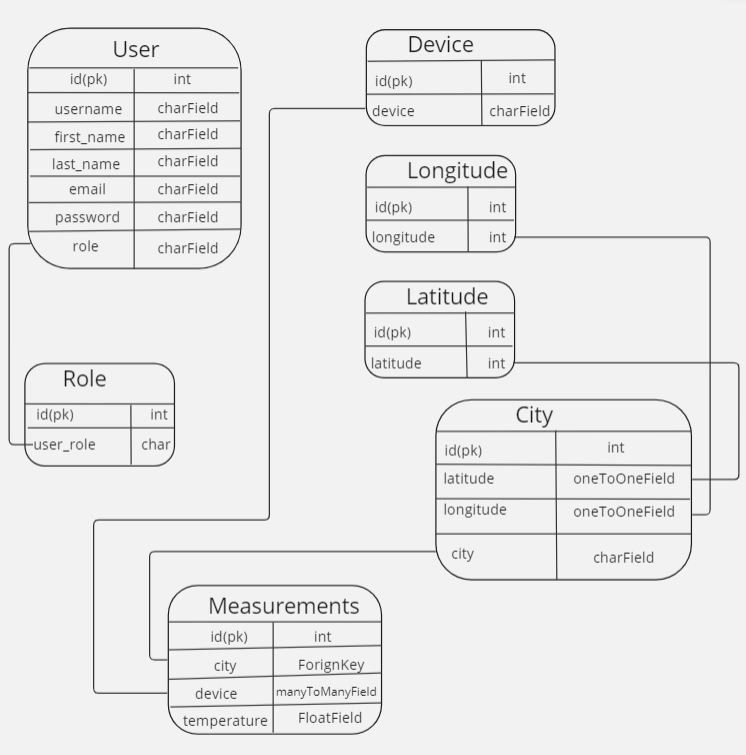
9. Добавить средства авторизации и разграничения прав доступа разным ролям к разным сущностям.

**Описание проекта**

1. Пользователю необходимо зарегистрировать аккаунт на платформе
2. Пользователю необходимо авторизоваться, после авторизации он попадает на страницу профиля.
3. На странице с 7 лабой, юзер может просмотреть все таблицы, главную таблицу со 3 видами связей и имеет возможность удалить или добавить записи
4. Также пользователь может сменить свои пароль и никнейм
5. Юзер может просмотреть данные об аккаунте, посмотреть дату регистрации и последнего изменения профиля

**Описание реализации**

Для того чтобы начать писать код приложения, была разработана структура базы данных



Дизайн web-приложения был придуман самостоятельно во всех лабораторных работах на фрейме Django (6 – 9 лабы).

В файле models.py были созданы модели таблиц бд.

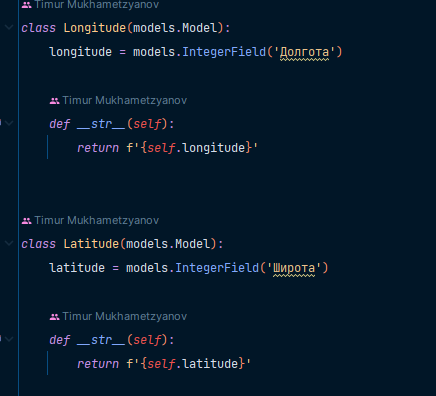


Рис. 1. Таблицы Широты и Долготы



Рис.2. Таблицы города с 3 полями и связью Один-к-одному из двух таблиц



Рис.3. Общая таблица со всеми данными и связями многие-ко-многим и Внешним ключем

В данном файле описываются поля наших таблиц, используя 3 типа связи (one to one, one to many, many to many)

В файле forms.py мы описываем формы для ввода данных (Регистрация, авторизация, обновление информации пользователя, создание записи для 7 Лабы)

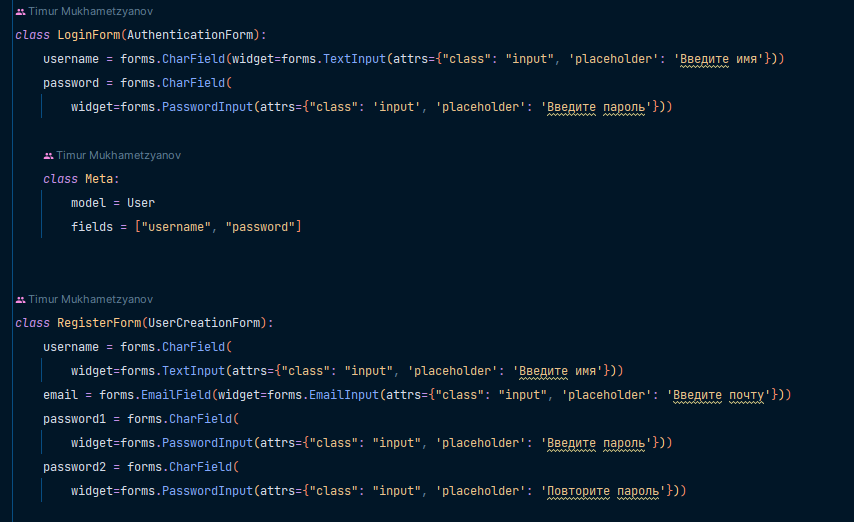


Рис.4. Классы для авторизации и регистрации пользователей

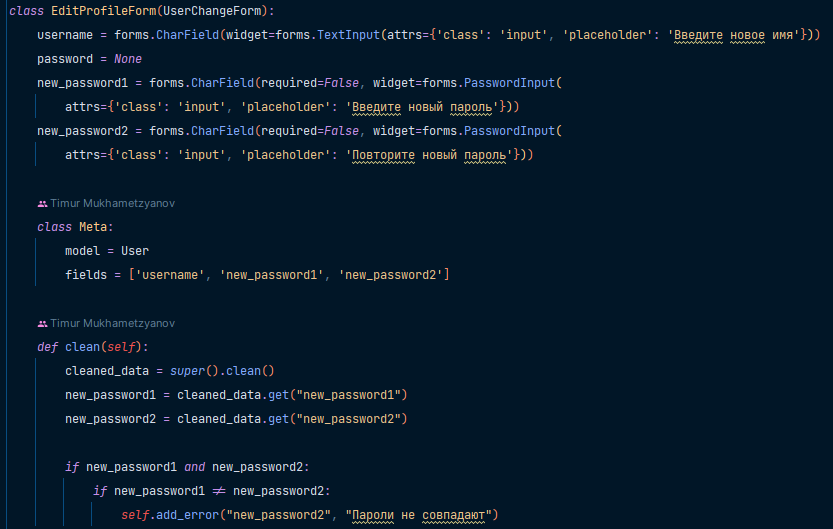


Рис.5. Класс редактирования профиля

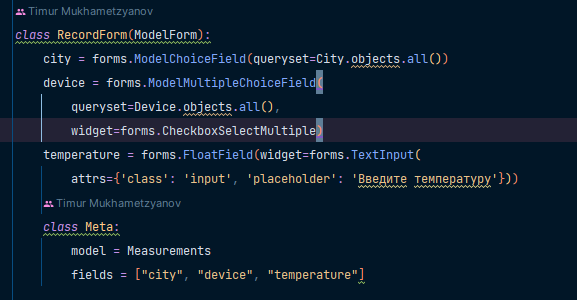


Рис.6. Класс добавления записей в бд с мульти выбором по связям

В файле urls.py мы указываем по каким ссылкам, какие функции будет срабатывать

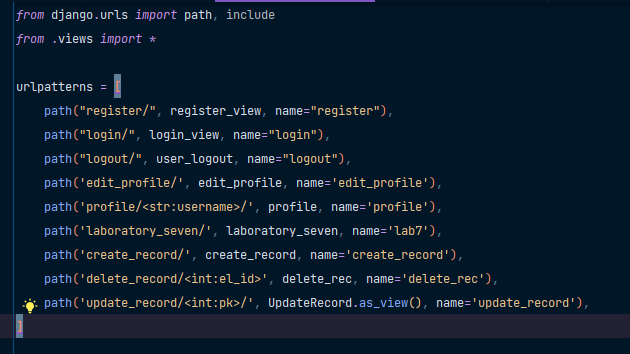


Рис.7. Urls

Функции описываются в файле views.py



Рис.8. Функции для регистрации и авторизации

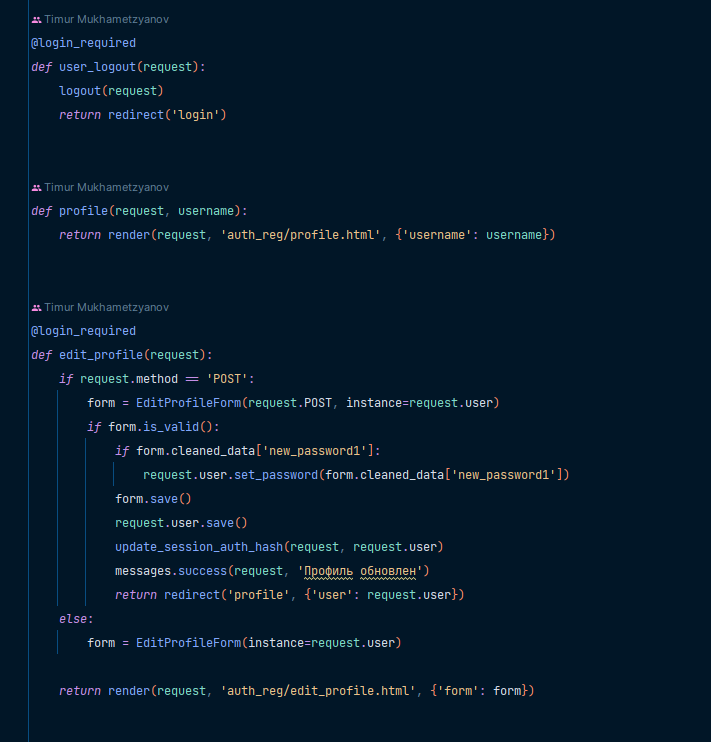


Рис.9. Функции редактирования профиля, выхода из аккаунта и редиректа на страницу профиля

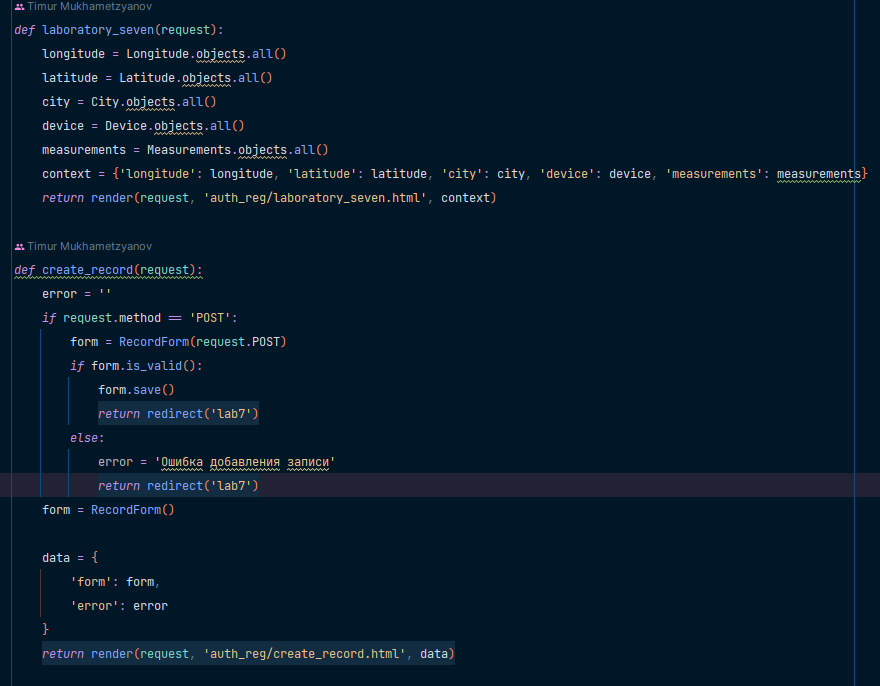


Рис.10. Функции вывода данных на страницу с 7 лабой и создания записи

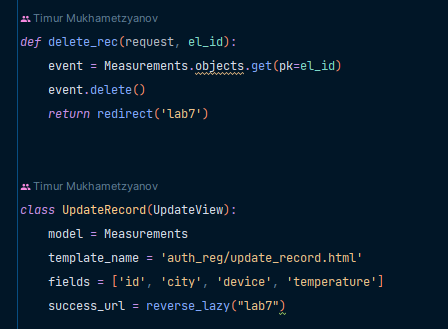
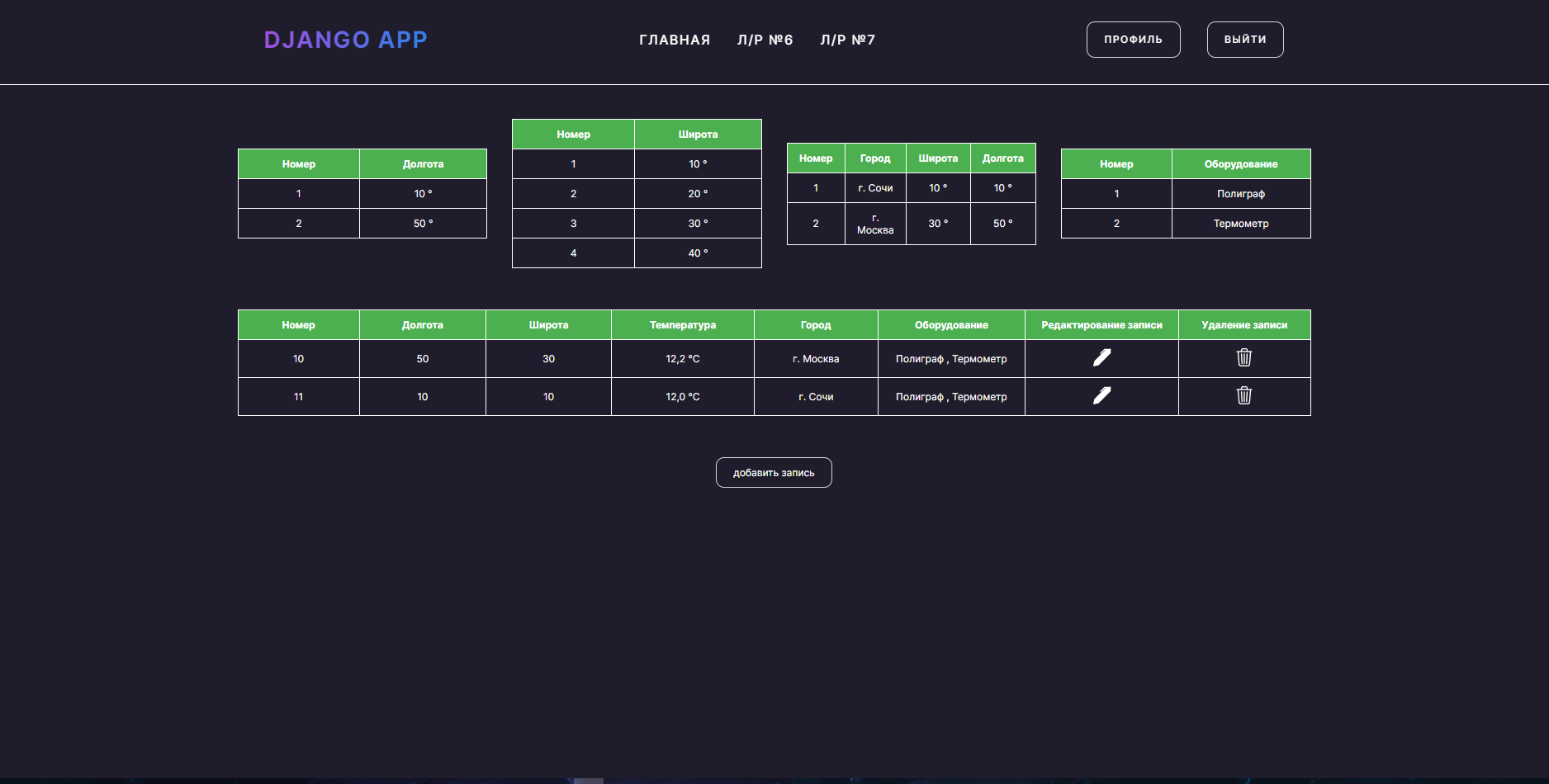


Рис.11. Класс редактирования записи в 7 лабе и удаления записи

**Тестирование web приложения**

  
Рис.12. Страница с таблицами из 7 лабы

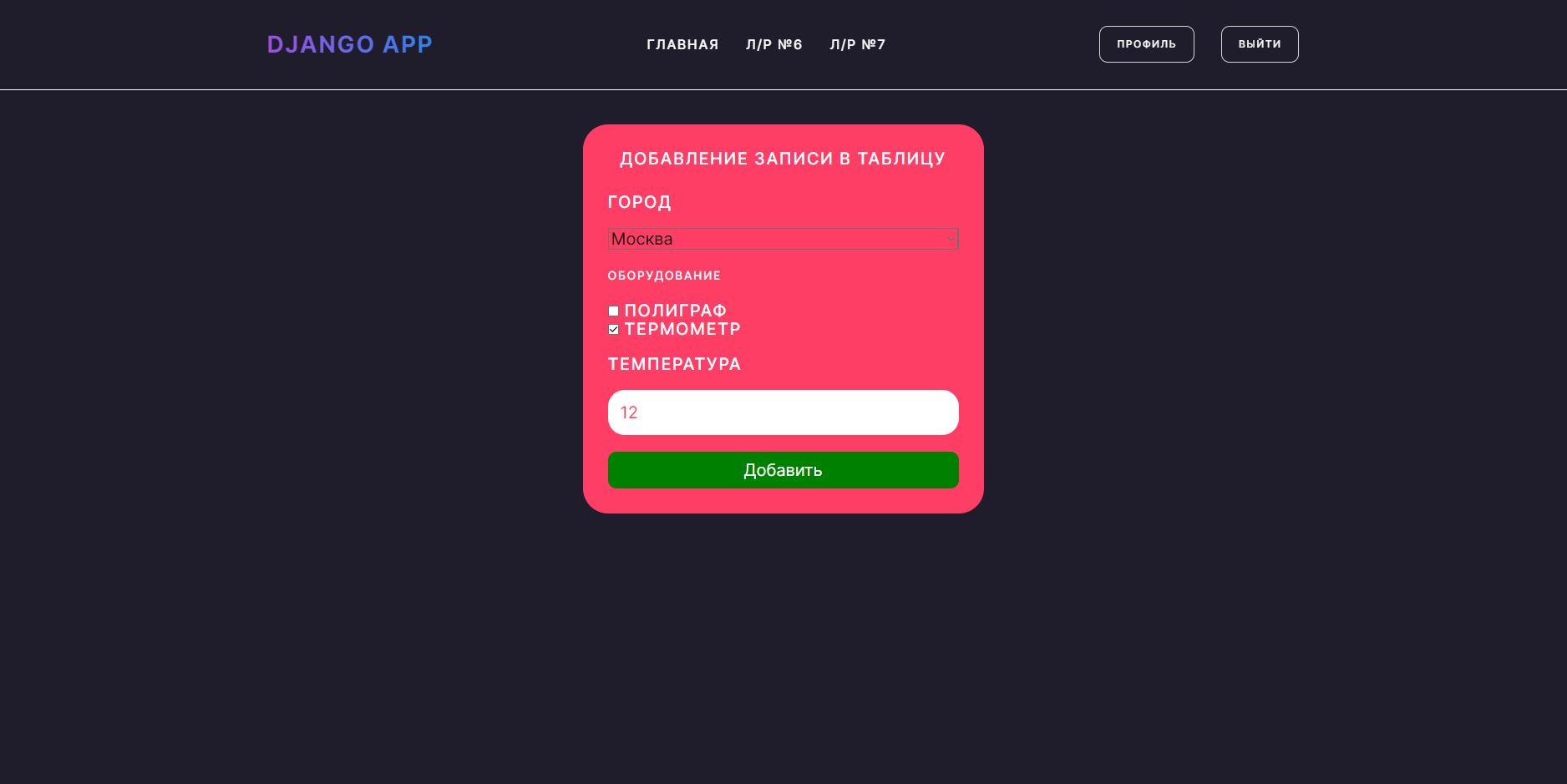


Рис.13. Страница с добавлением записи в главную таблицу, выбор устройства определения осадков по выбору (виды связей)

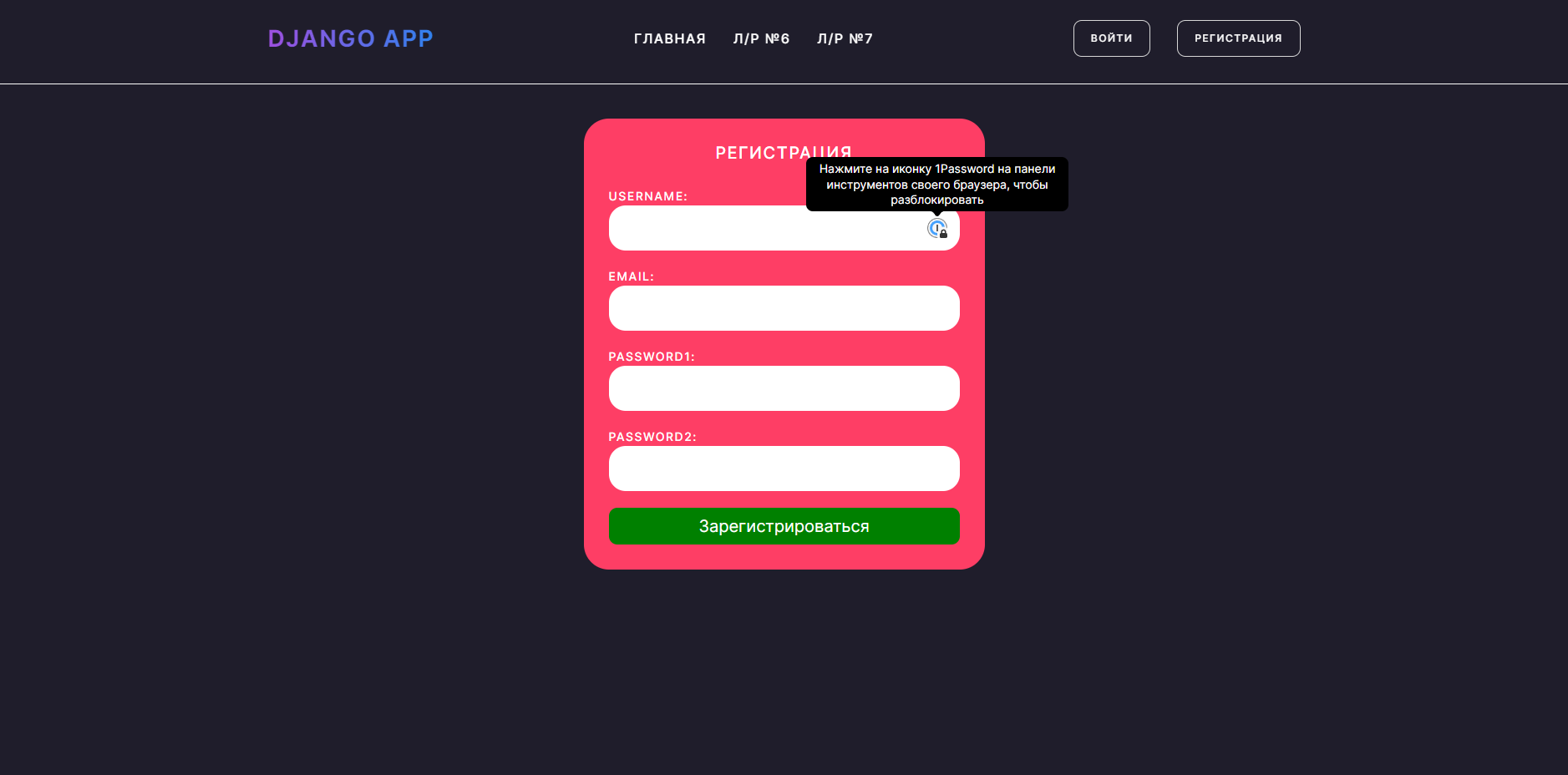


Рис.14. Страница с регистрацией аккаунта

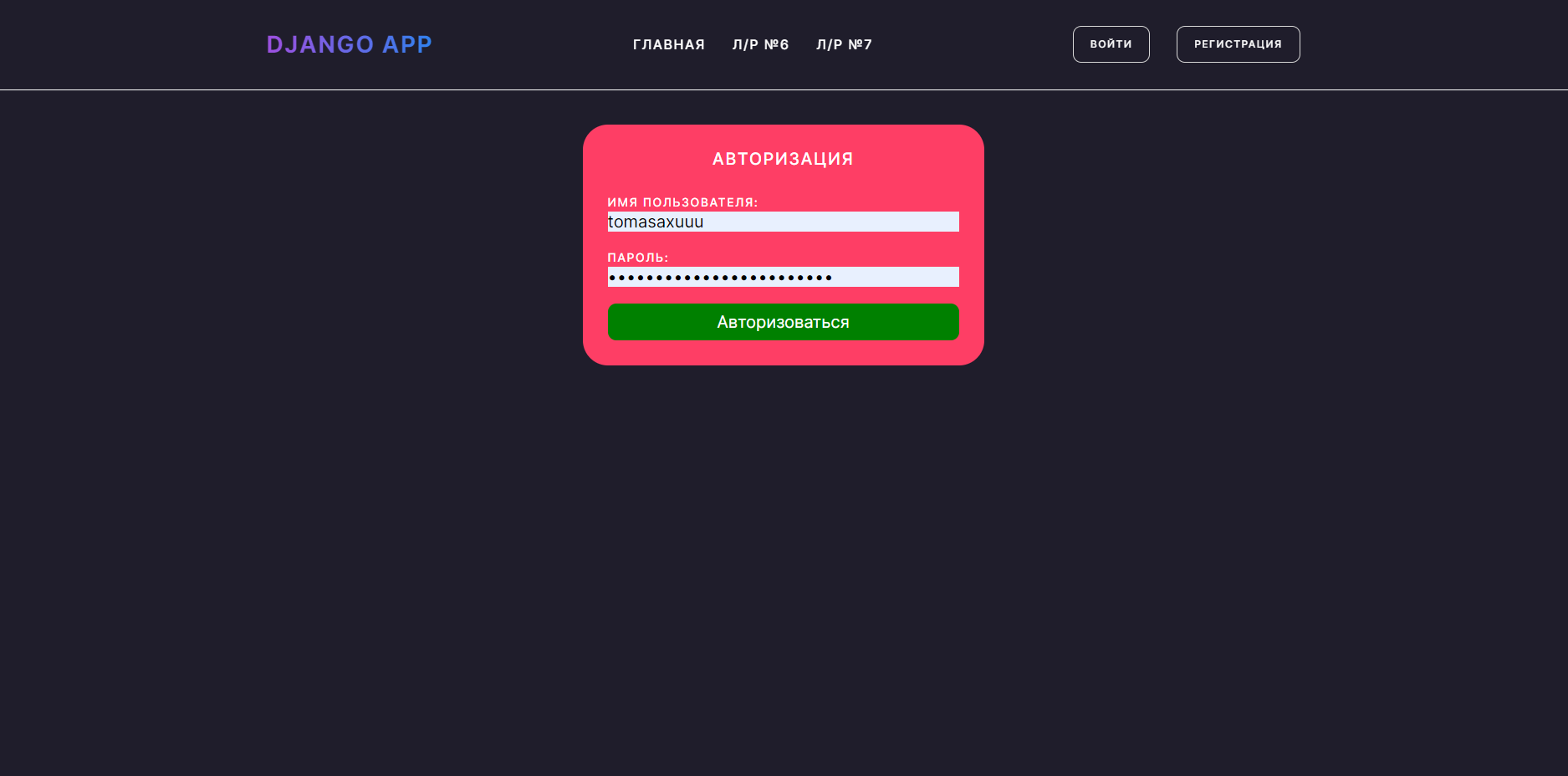


Рис.15. Страница с авторизацией

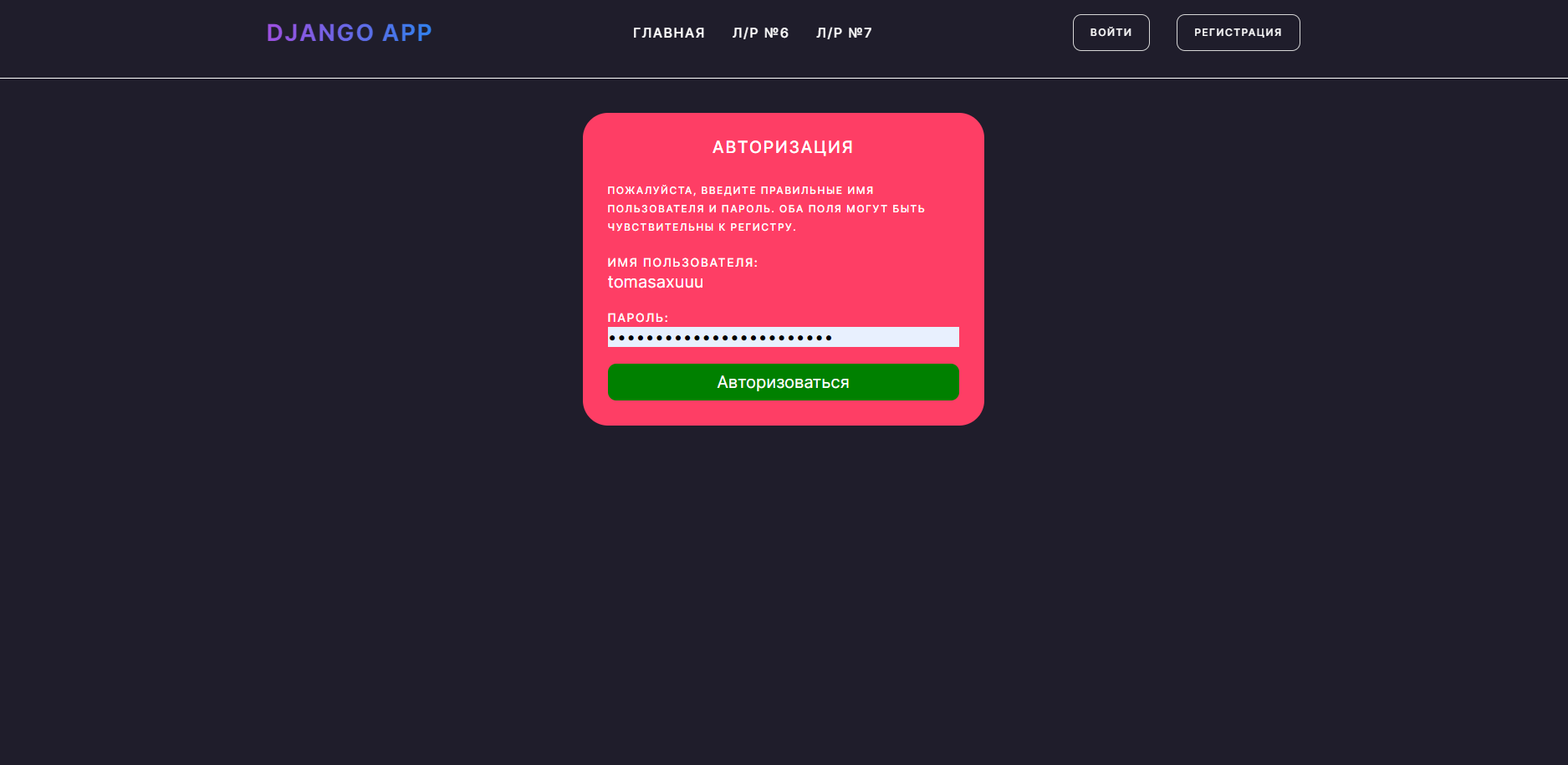


Рис.16. Вывод ошибки

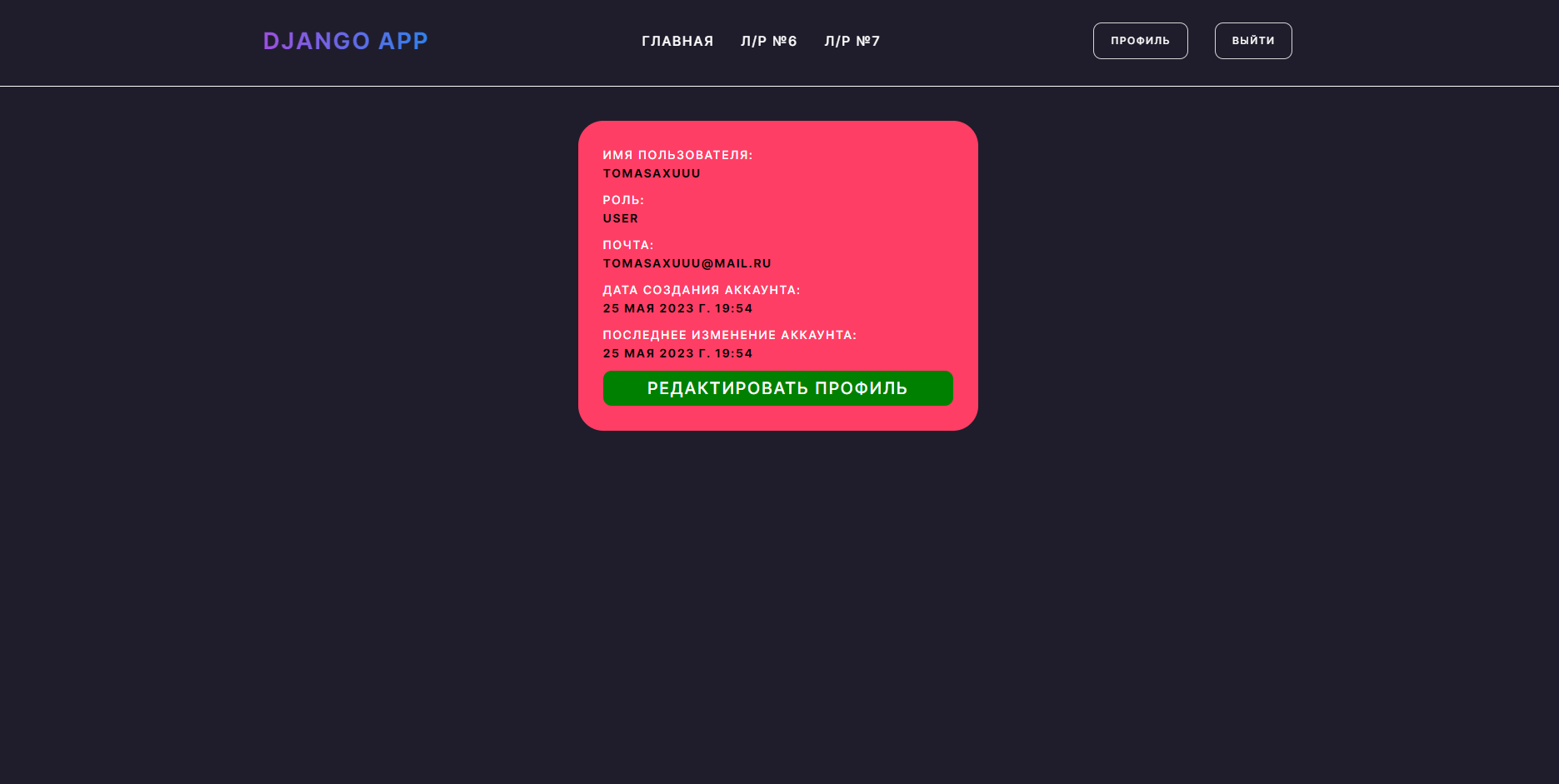


Рис.17. Профиль после авторизации

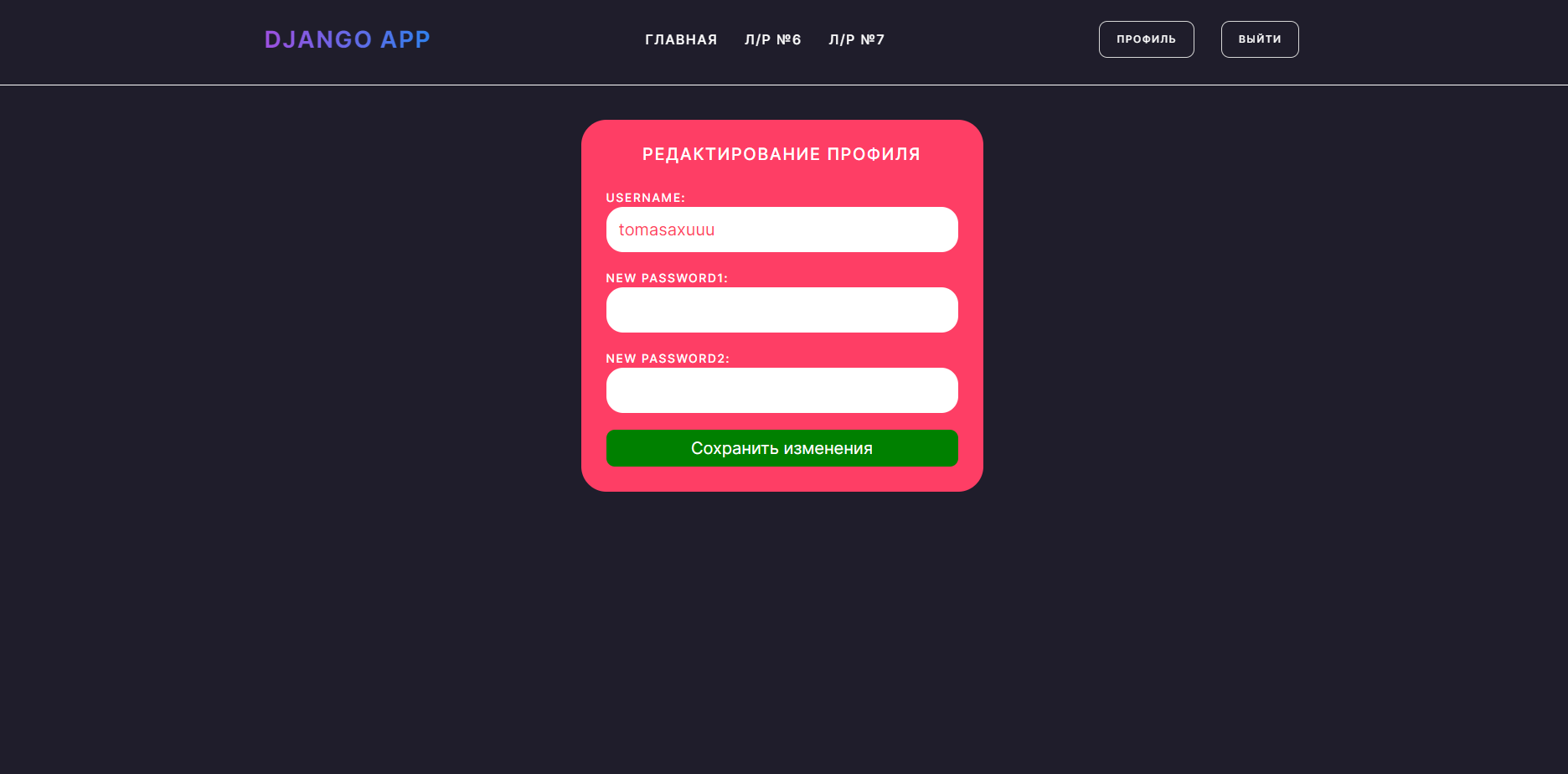


Рис.18. Редактирования профиля

**Памятка для разработчика:**

1. Скачать проект на официальном **Github** разработчиков по ссылке ниже
2. Запустить проект в PyCharm или любой другой удобной IDLE
3. Открыть терминал
4. Установить Django командой «pip install Django»
5. Перейти в директорию «app» командой «cd .\app\»
6. Запустить проект командой «py manage.py runserver»
7. После запуска откройте **url** сайта <http://127.0.0.1:8000/>