Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет**

**По курсу «Разработка интернет приложений»**

**«Лабораторная работа №8»**

**ИСПОЛНИТЕЛЬ:**

Группа ИУ5-53Б

Рудзинский В. В.

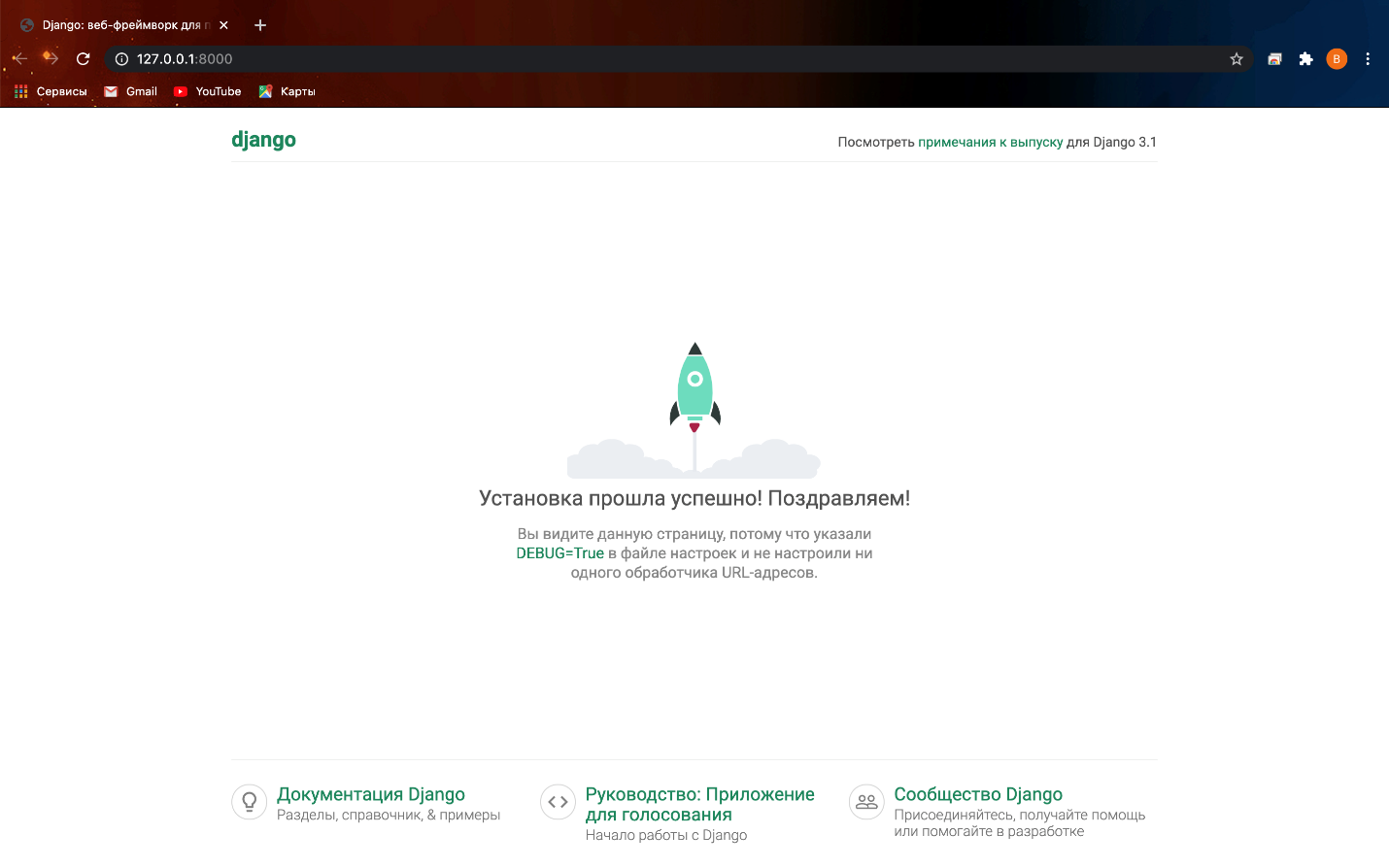
"21" декабря 2020 г.

**ПРЕПОДАВАТЕЛЬ:**

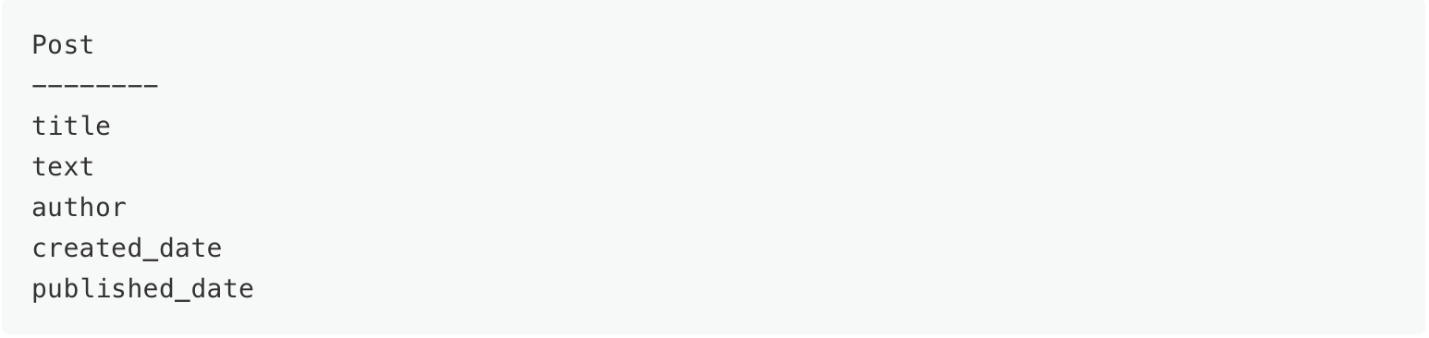
Гапанюк Ю.Е

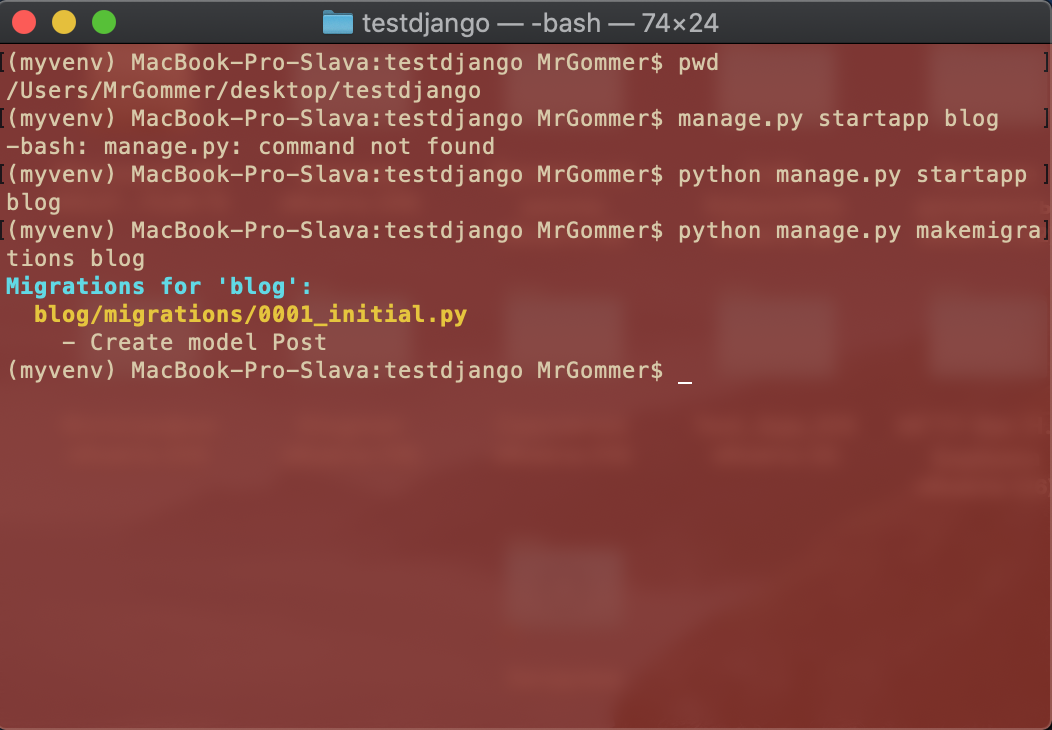
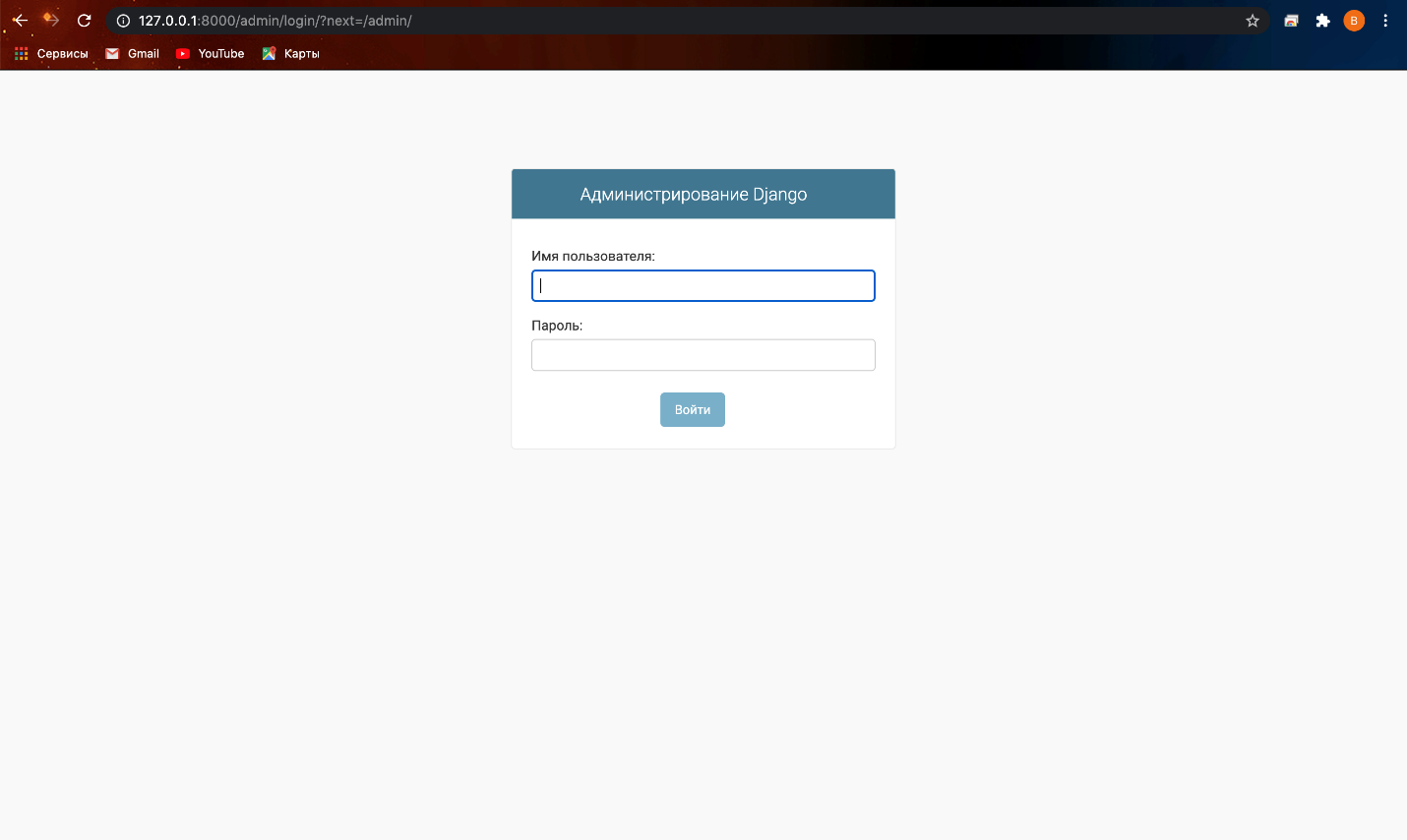
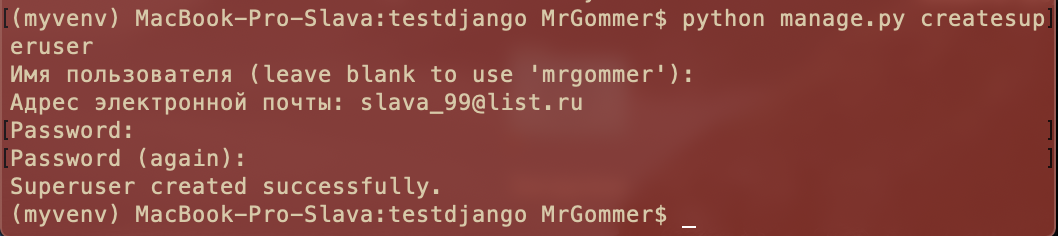
Москва 2020

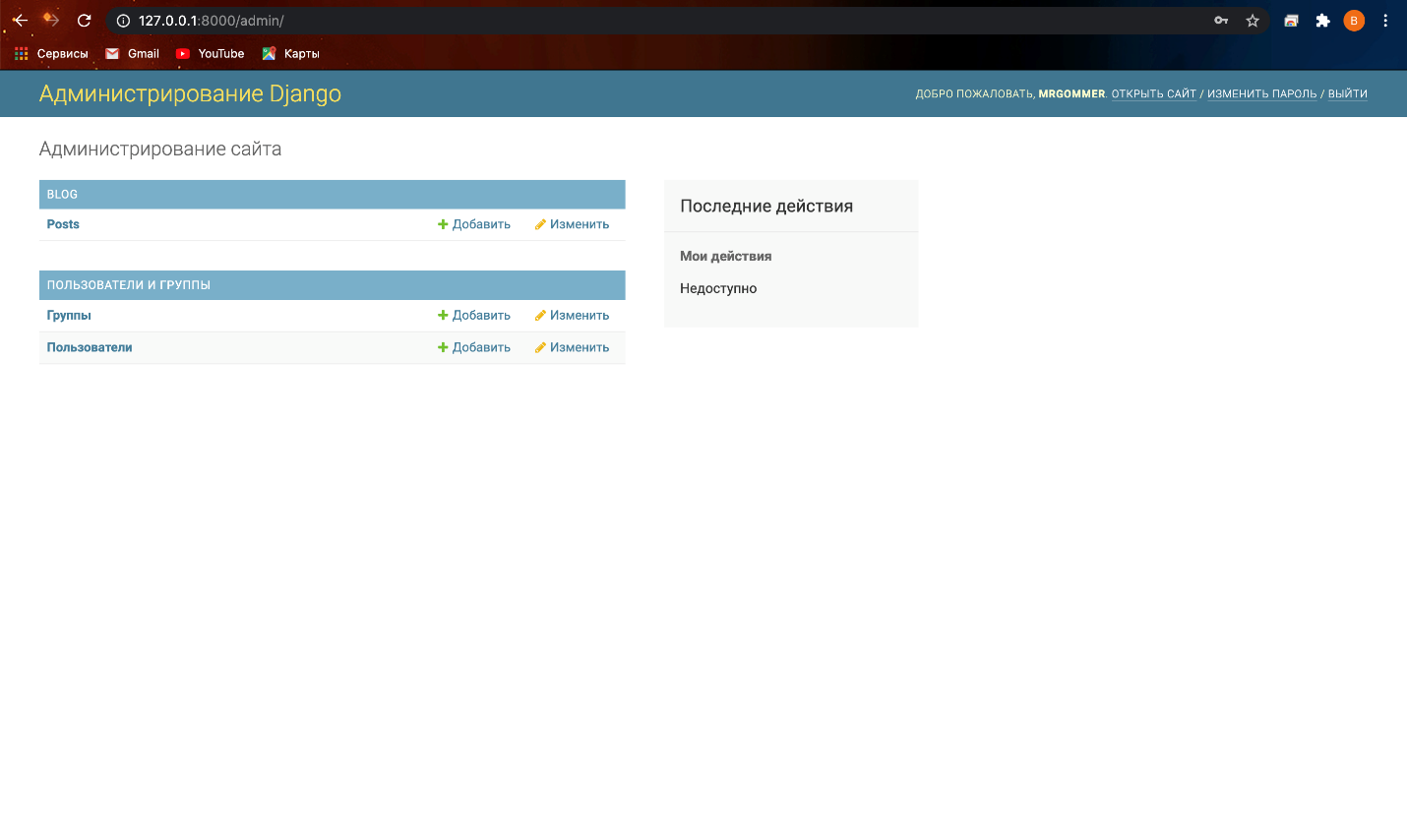
После установки Django, создания виртуального окружения и нового проекта в среде разработки PyCharm, а также корректной настройки **manage.py** и **Edit configuration** имеем вот такой работающий сервер



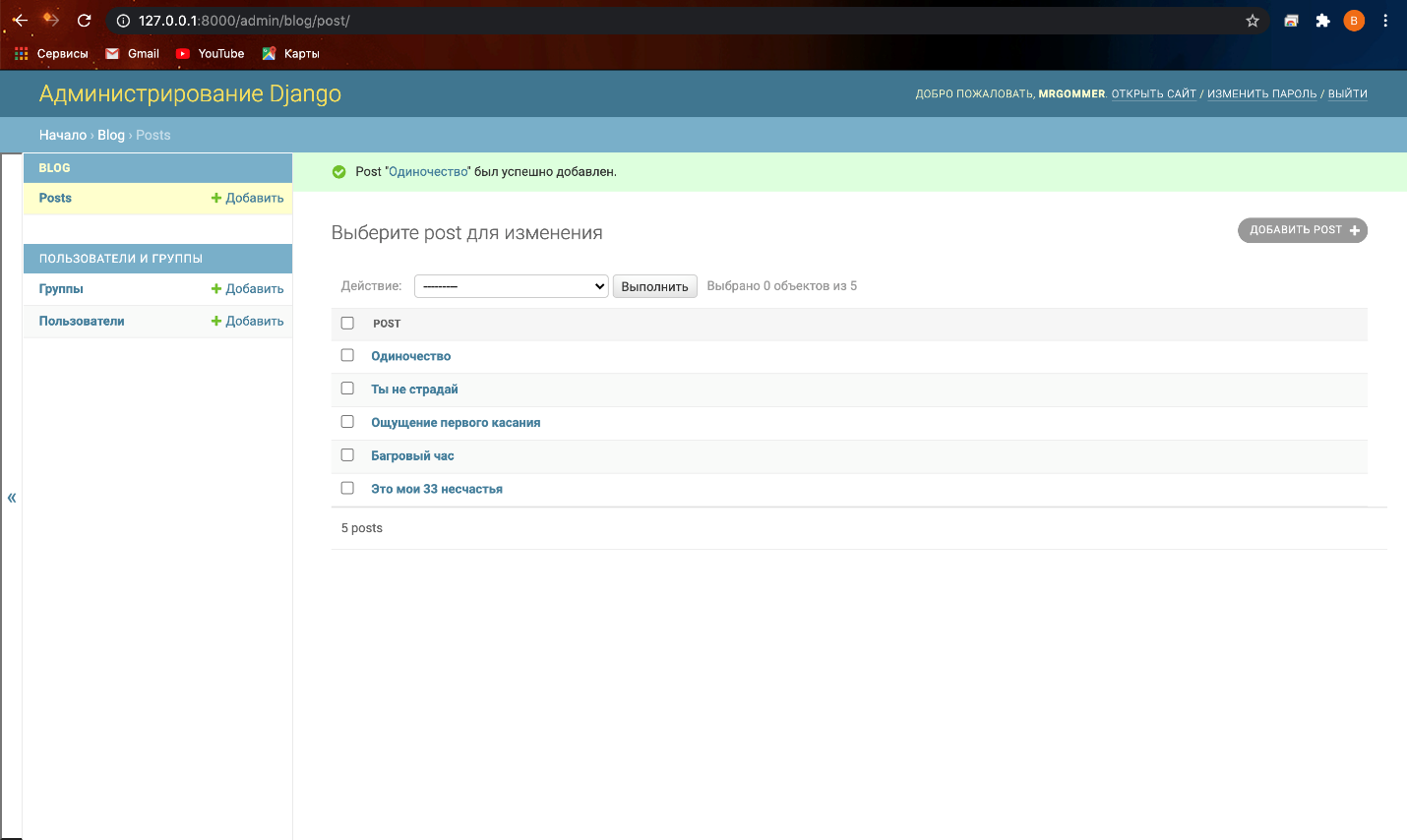
Далее создаем модель Post на основе придуманной нами схемы, также создаем суперпользователя и авторизуемся по этим данных через «админку» сайта



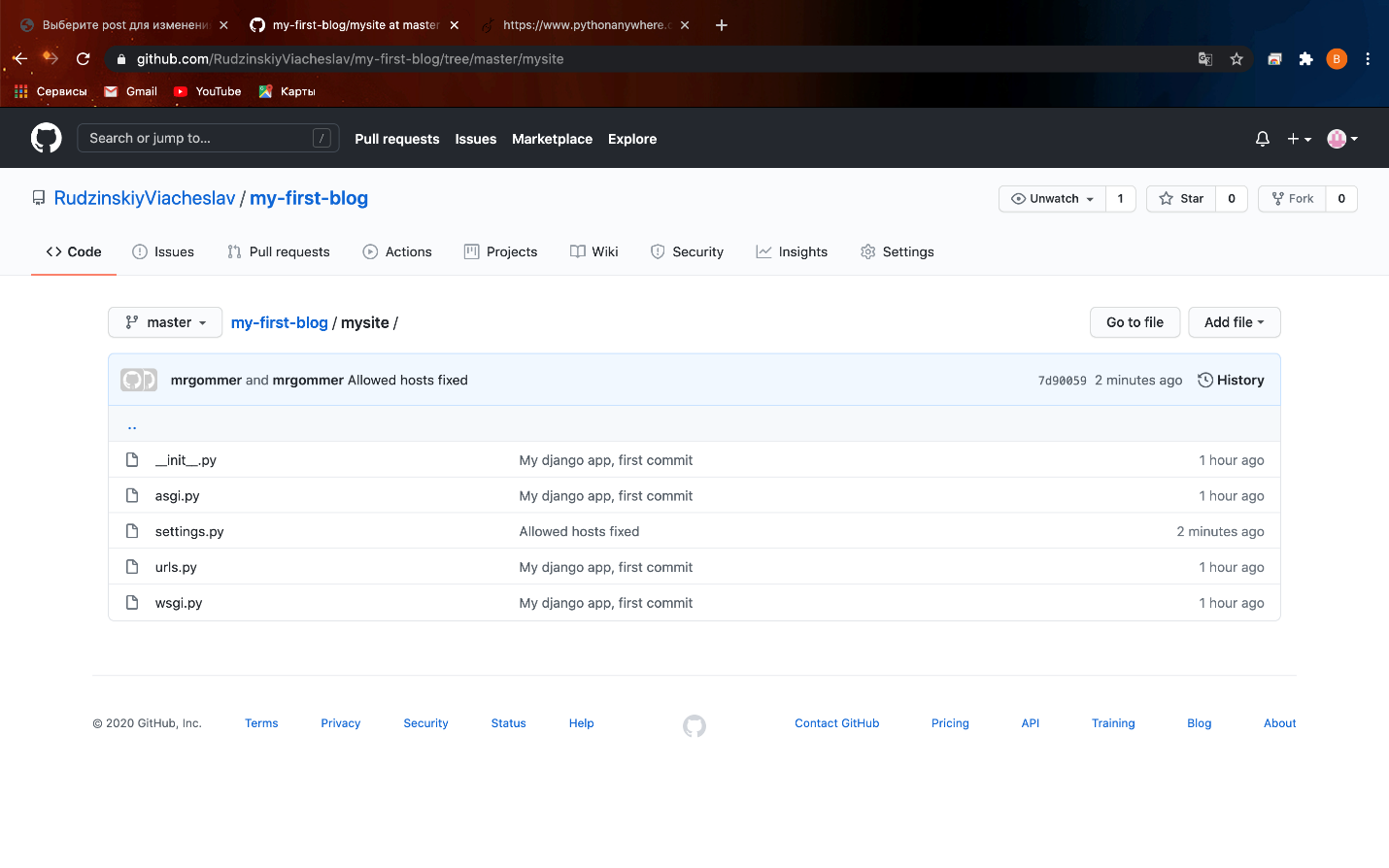




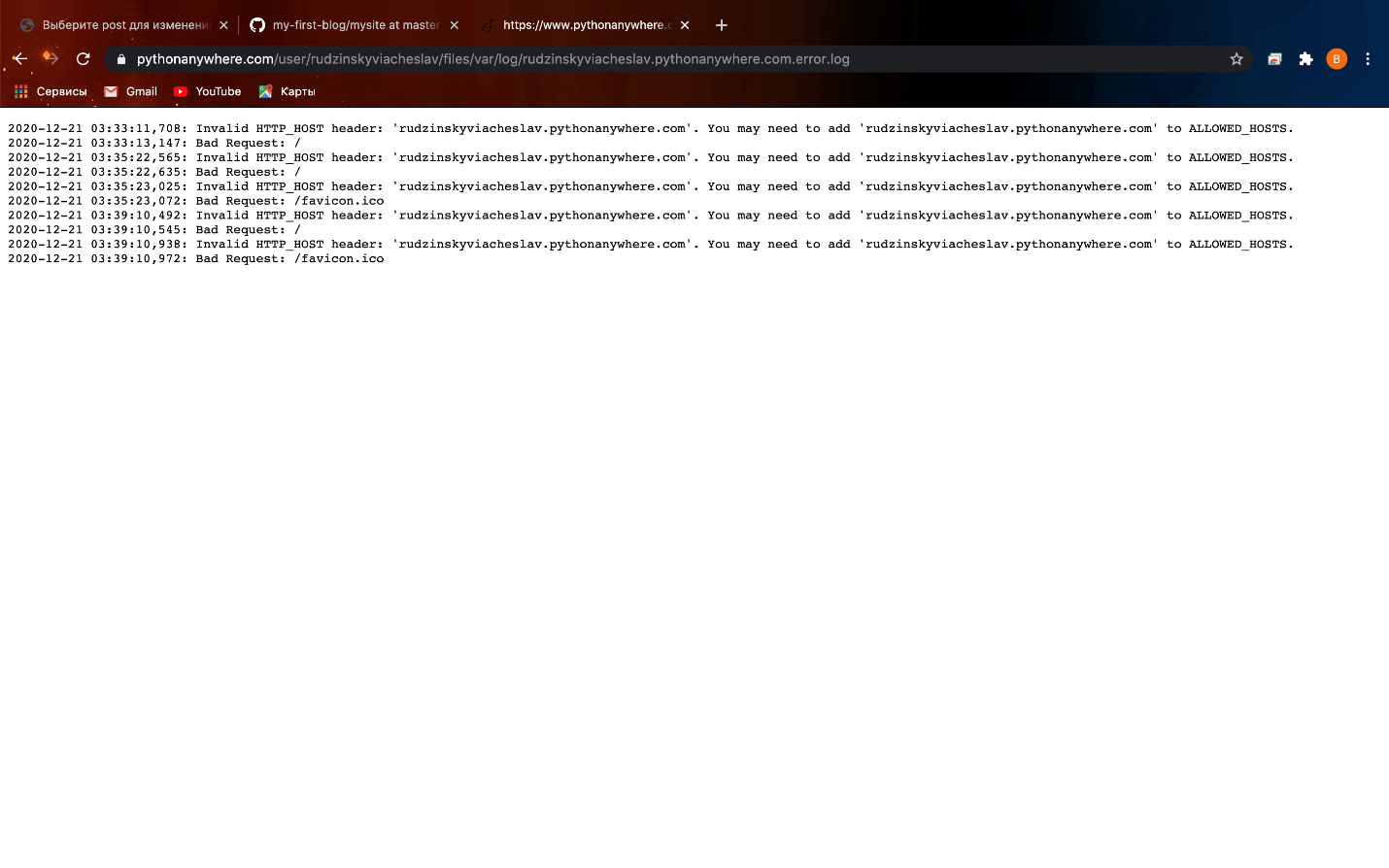
Создаем несколько постов через «админку»

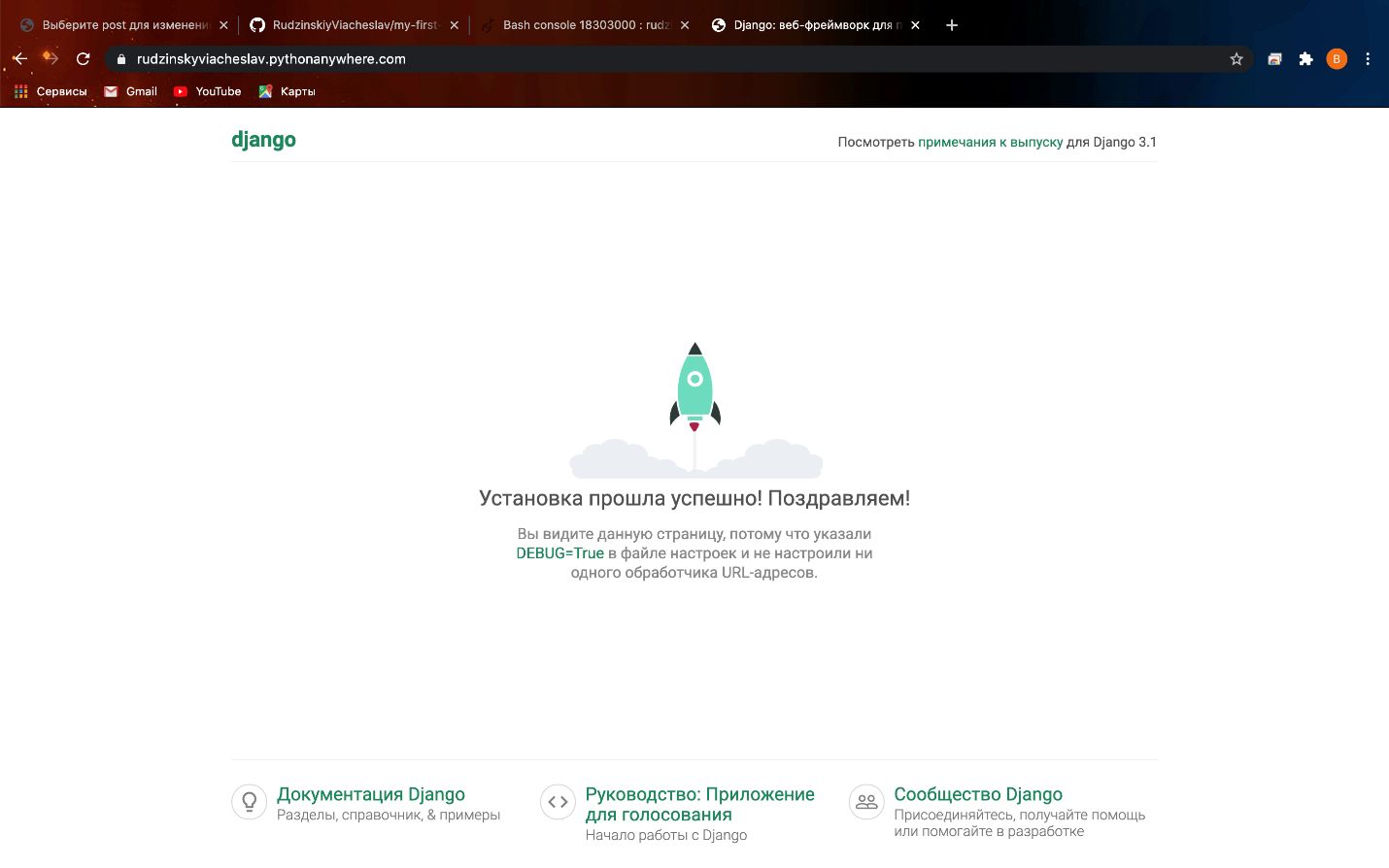


Далее я создал учетную запись на PythonAnywhere и загрузил свой проект blog в новый репозиторий на GitHub. Также я запустил удаленную консоль на PythonAnywhere и настроил сервер так, чтобы он получал данные с моего GitHub для запуска сервера удаленно

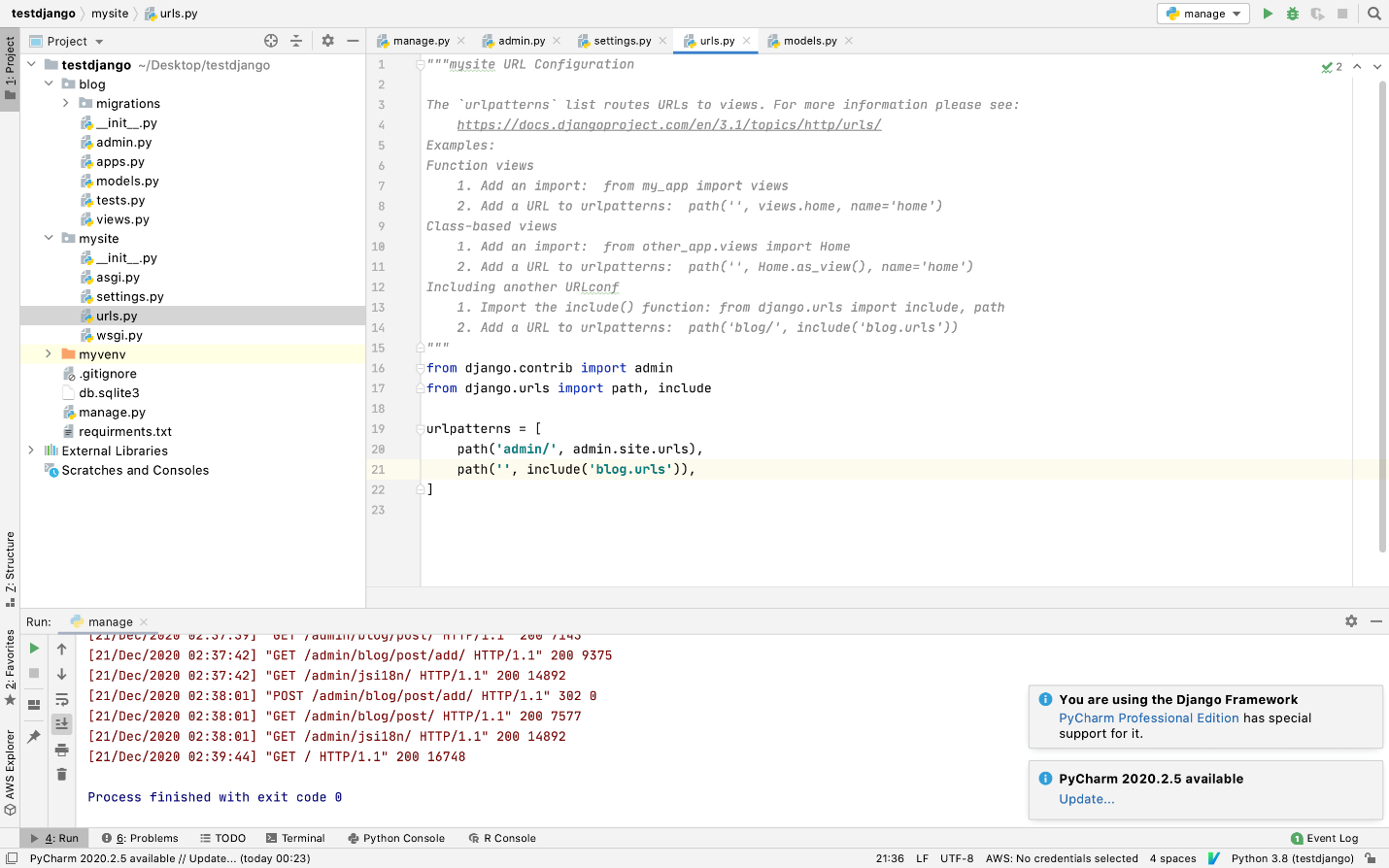


После проделанной работы выявилась ошибка, которую надо было исправить в файле **url.py,** добавив новый адрес сервера в допустимые хосты. После всего удаленный сервер работает исправно

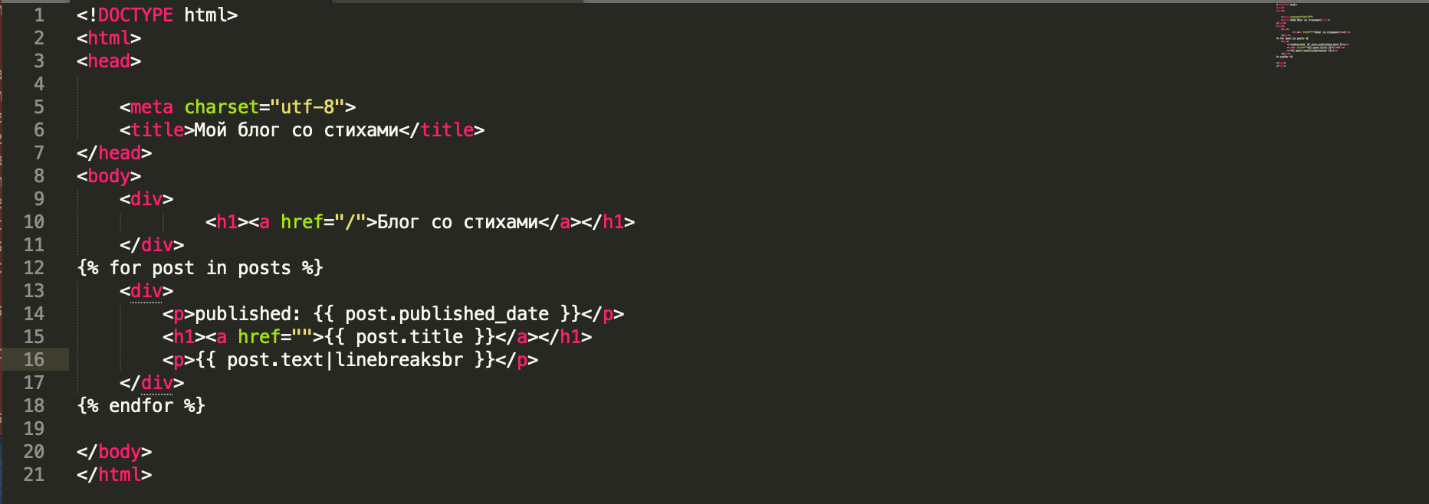




Далее создаем «урлы» и тестируем QuerySet, добавляя вывод своих данных из базы также и в HTML-шаблон.







Далее вставляем теги шаблонов Django для создания динамичного сайта.

После добавляем красивые каскадные таблицы стилей CSS, используя bootstrap и подключаем все это через статические файлы Django.

После этого расширяем шаблон, добавляя **базовый шаблон** для удобства использования его в других шаблонах сайта.

В конце добавляем и, тем самым, расширяем приложение, добавляя ссылку на отдельный пост, а также на создание и изменение записи, используя **Django формы,** позволяющие сохранять и отображать все в реальном времени**.**

В конце все еще раз загружаю через **push** на GitHub и перезапускаю сервер PythonAnywhere, чтобы все работало корректно и удаленно, а не только локально.

Как итог, имею готовый проект для ведения блога стихотворений удаленно с возможностью добавлений поста, а также его редактирования удаленно. При этом вэб-приложение учитывает безопасность и не даст произвести действия над блоками без авторизации.

Ниже привожу скриншоты работы вэб-приложения. Весь исходный код доступен в репозитории. Также приложение доступно по адресу: <https://rudzinskyviacheslav.pythonanywhere.com/>

