1. Министерство образования и науки Российской Федерации
2. Санкт-Петербургский Политехнический Университет Петра Великого
3. —
4. Институт кибербезопасности и защиты информации

**ЛАБОРАТОРНАЯ РАБОТА № 2**

**Реализация backend и аутентификации**

по дисциплине «Безопасность интернет приложений»

1. Выполнили Берко А.С.
2. студенты гр. 4851003/90801 Кулеева А.Г.

Реденко И.Д.



1. Руководитель
2. ассистент Климшин И.И.
4. Санкт-Петербург
5. 2023

# Задание

1. Регистрация в Интернет-приложении. Двухфакторная аутентификация в Интернет-приложении. Второй фактор в аутентификации выбирается самостоятельно, главное требование исходит из определения — второй фактор подтверждение не должен доставляться по тому же каналу, что и первый.
2. Бэкэнд с готовым API, оформленный в Swagger. API должен покрывать функционал, который будет доступен пользователю приложения. Для этого необходимо изначально прикинуть, какие функции будут в вашем Интернет-приложении, и на основе функций спроектировать API.
3. Интерфейс не обязателен, даже в аутентификациях.

# Ход работы

## Создание и начальная настройка проекта

Было принято решение разрабатывать интернет-приложение с помощью фреймворка Python Django. Для этого был использован интерпретатор Python версии 3.10.5, а также сам фреймворк Django версии 4.1.7 (на данный момент является крайней версией). После установки фреймворка в терминале была введена команда python -m django startproject bip, после чего был создан проект с некоторыми начальными параметрами. Разберем созданные файлы:

**manage.py** — является ссылкой на скрипт django-admin, но с уже предустановленными переменными окружения, указывающими на проект, как для чтения настроек оттуда, так и для управления им при необходимости;

**settings.py** — здесь находятся настройки проекта. Файл уже содержит несколько разумных настроек:

* Секретный ключ;
* Режим отладки (DEBUG = True);
* Установленные приложения;
* Некоторые ссылки;
* Параметры работы с базами данных. В качестве СУБД была выбрана SQLite, установленная по умолчанию с фреймворком;
* Поддерживаемый язык (был изменен на русский);
* Часовой пояс (был оставлен по умолчанию);

**urls.py** — содержит URL'ы для отображения представлений;

**wsgi.py** — это WSGI обёртка для приложения. Этот файл используется сервером разработки Django и возможно другими контейнерами, такими как mod\_wsgi, uwsgi и др. на «боевом» сервере;

**models.py** будет содержать Django ORM-модели для приложения;

**views.py** будет содержать код представлений;

**tests.py** будет содержать написанные разработчиками модульные и интеграционные тесты.

установить python Django, создать проект

начальная настройка проекта (язык, бд, зависимости)

создать страницы сайта (шаблоны html, бд, минимальные кнопки)

список всех предложений

подробный разбор каждого предложения

профиль организации (рест + благ, комменты, оценка (пока хз как))

написать отчет по своей части

1. Представитель ресторана и представитель благотворительной организации регистрируются в системе. При регистрации обе стороны указывают такие данные, как ОГРН, ИНН, КПП, дата регистрации, юр адрес и т.д. Эти данные проверяются через госуслуги (такая информация должна быть открыта). Пока было решено проверять данные вручную администратором для удобства тестирования.
2. Благотворительная организация выкладывает объявление о необходимости ресурсов. Все объявления организованы в виде доски. Добавляем также фильтрацию по городам. Объявление краткое типа «требуется 50 порций супа для бездомных». При открытии объявления высвечивается более подробная информация, а именно сроки проведения, геопозиция, вся информация об организации. Также может быть дополнительная информация, например, «Акция «Соберем помощь ветеранам» приурочена к 9 мая…»
3. Ресторан может откликнуться на объявление по соответствующей кнопке. Благотворительная организация сама выбирает исполнителя из откликнувшихся. После этого сделка считается открытой. Далее обе стороны общаются в чате, обсуждают орг моменты, обмениваются номерами телефонов. Для этого добавляем шифрование.
4. Есть система репутации для ресторанов. После выполнения сделки обе стороны ставят друг другу звезды и могут написать отзыв. Закрытие сделки ставится благотворительной организацией через нажатие соответствующей кнопки.