

**Tab 1**

# Дипломная работа по теме:

## Анализ и сравнение написания web-приложений с использованием разных фреймворков.

### Содержание

Обзор проекта.....	1
Структура проекта Django.....	2
Домашняя страница.....	2
Функционал приложения.....	3
База данных.....	5
Интерфейс админ-панели.....	6
Структура проекта FastAPI.....	7
Функционал приложения.....	7
База данных.....	8
Структура проекта Aiogram.....	9
Функционал чат-бота.....	9
База данных.....	11
Заключение.....	12
Приложение №1 Список необходимых библиотек.....	12

### Обзор проекта.

Целью работы было создание приложения для работы с клиентами условной компании по предложению товаров и услуг, в частности демонстрация товара, описание, цена и краткий экскурс в деятельность компании. Так же предполагалось создание простейшей базы данных для хранения и использования в коммерческих целях хранящихся данных, возможность администрирования и учета товаров.

Для выполнения поставленной задачи принято решение создать полноценное веб-приложение с использованием фреймворка Django. Согласно заданию, необходимо провести анализ и сравнение веб-приложений созданных на основе разных фреймворков. Поэтому дополнительно создадим с использованием фреймворка FastAPI программный интерфейс обеспечивающий взаимодействие

двух программ между собой с возможностью встраивать контент с любого сайта. И применим в работе фреймворк Aiogram, создадим простого чат-бота с возможностью взаимодействия с пользователем, демонстрации каталога, создания базы данных и прочим схожим функционалом.

Для интерфейса полноценного веб-приложения с использованием фреймворка Django будем использовать Bootstrap, Paginator, CSS. Создадим каталог с шаблонами HTML, каталог с медиа файлами, настроим формы обратной связи с пользователями.

## Структура проекта Django.

Проект Django включает в себя следующие компоненты:

### Домашняя страница.

Здесь есть каталог товаров по категориям, при наведении курсора на категорию выбора срабатывает вызов ссылки со сменой фона. В верхнем меню имеется навигация по страницам сайта. Также логотип и “подвал”. (рис.1)

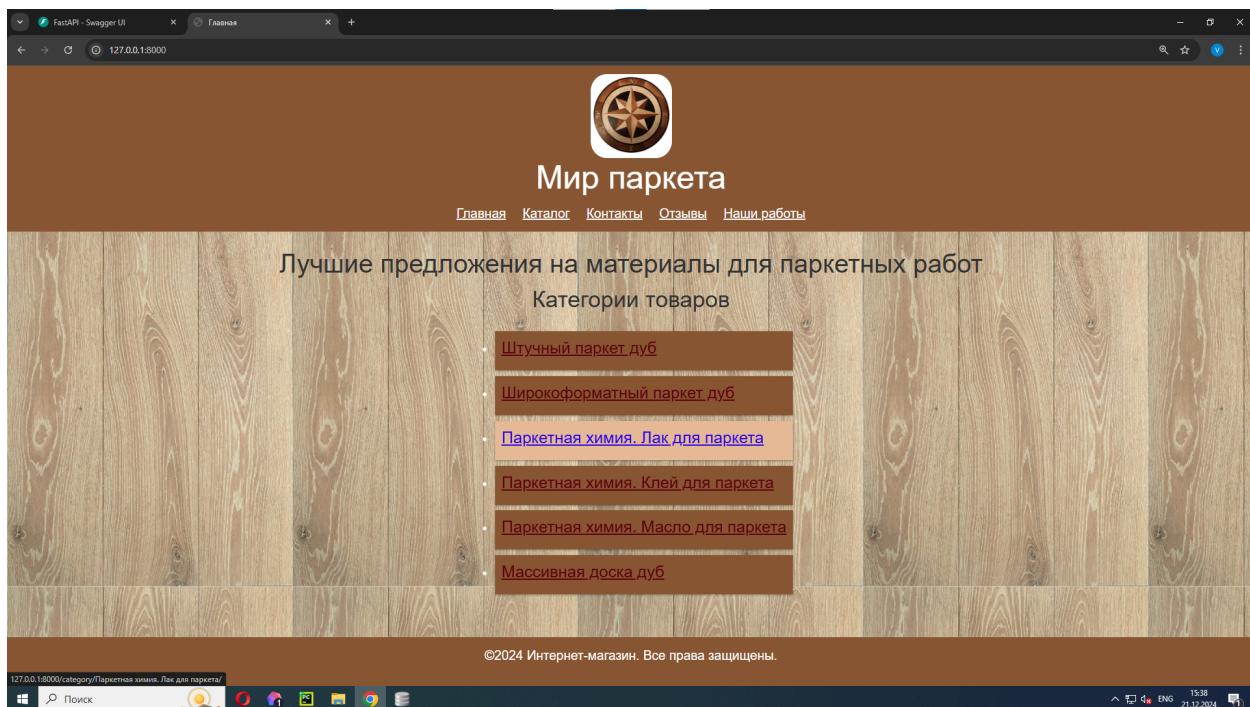
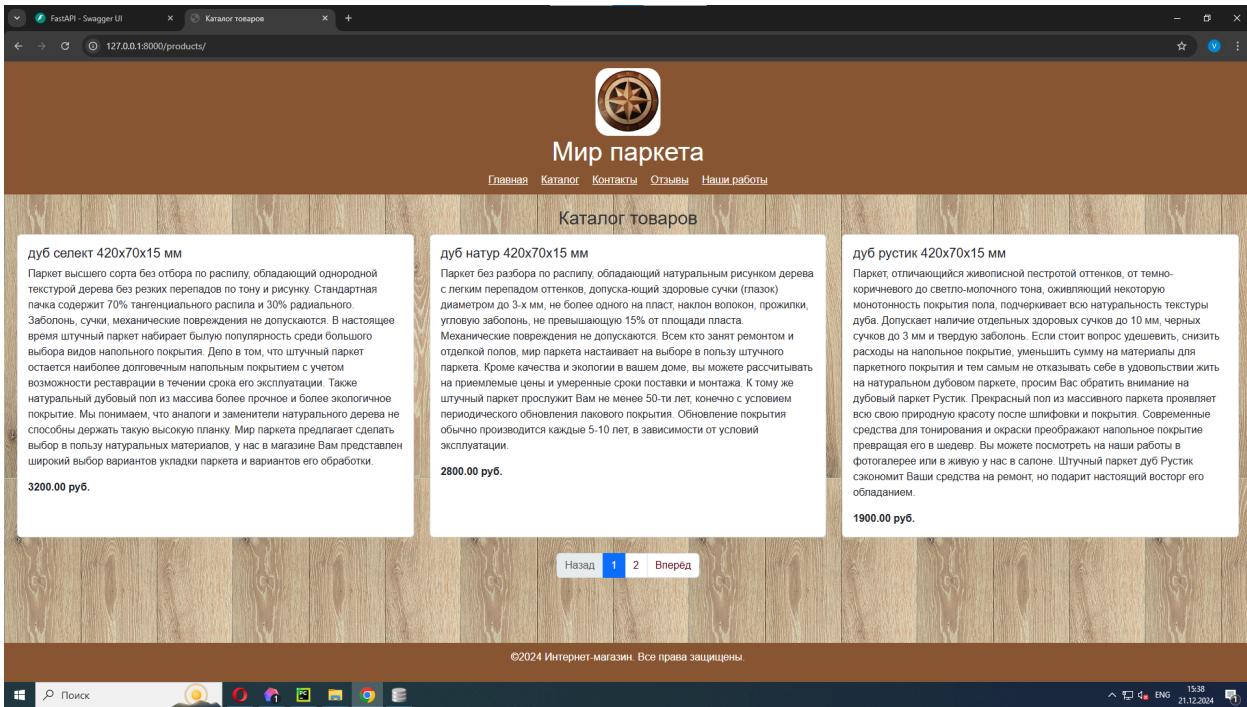


Рисунок 1 Домашняя страница

## Функционал приложения

Переходя по ссылкам каталога или меню сайта можно ознакомится с продукцией (рис 2), медиа файлами “наши работы” (рис 3), оставить отзыв о товарах (рис4), отправить сообщение администрации (рис 5).



*Рисунок 2 Название товара, описание и цена. Для удобства восприятия размещены карточки и пагинация.*

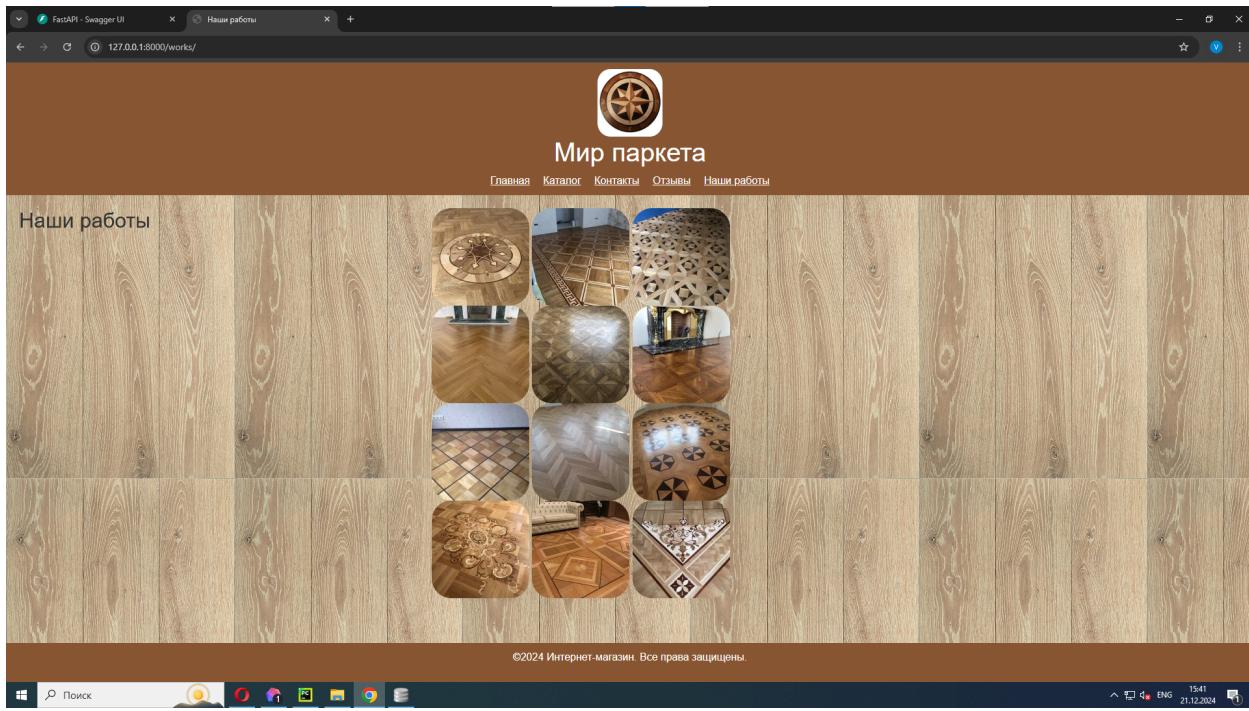


Рисунок 3 Галерея медиафайлов “наши работы”

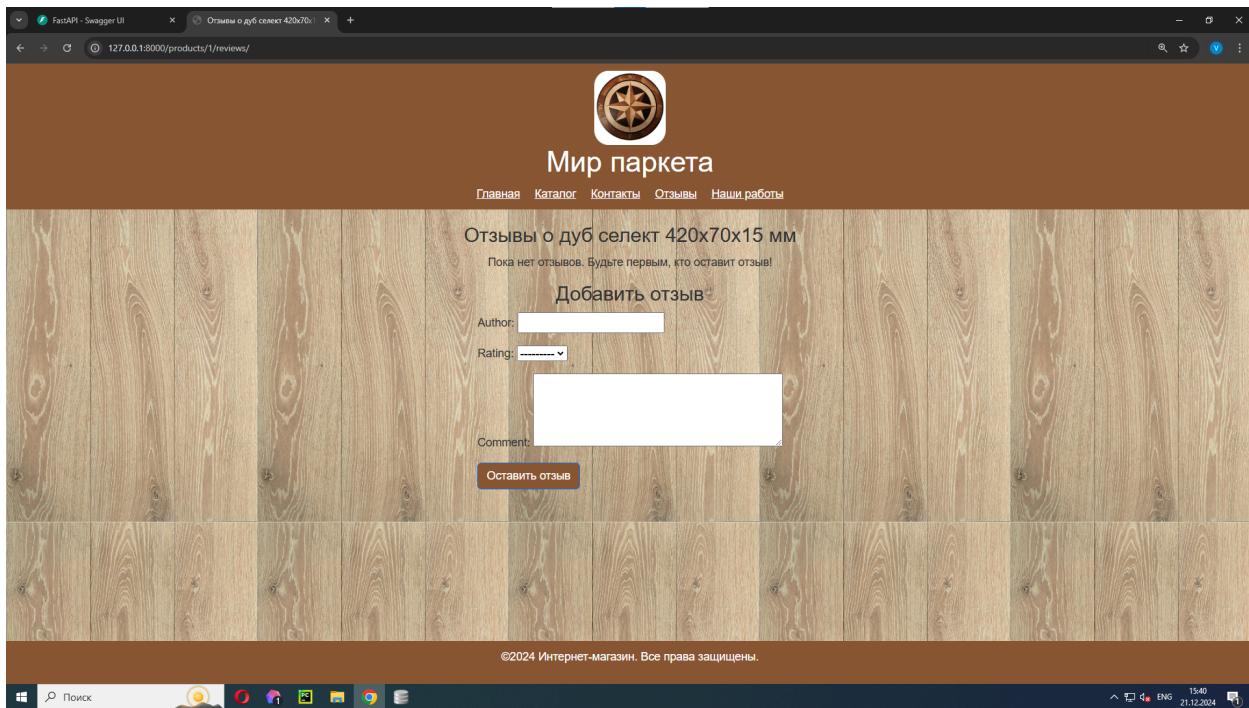


Рисунок 4 Форма с работающим интерфейсом “оставить отзыв”

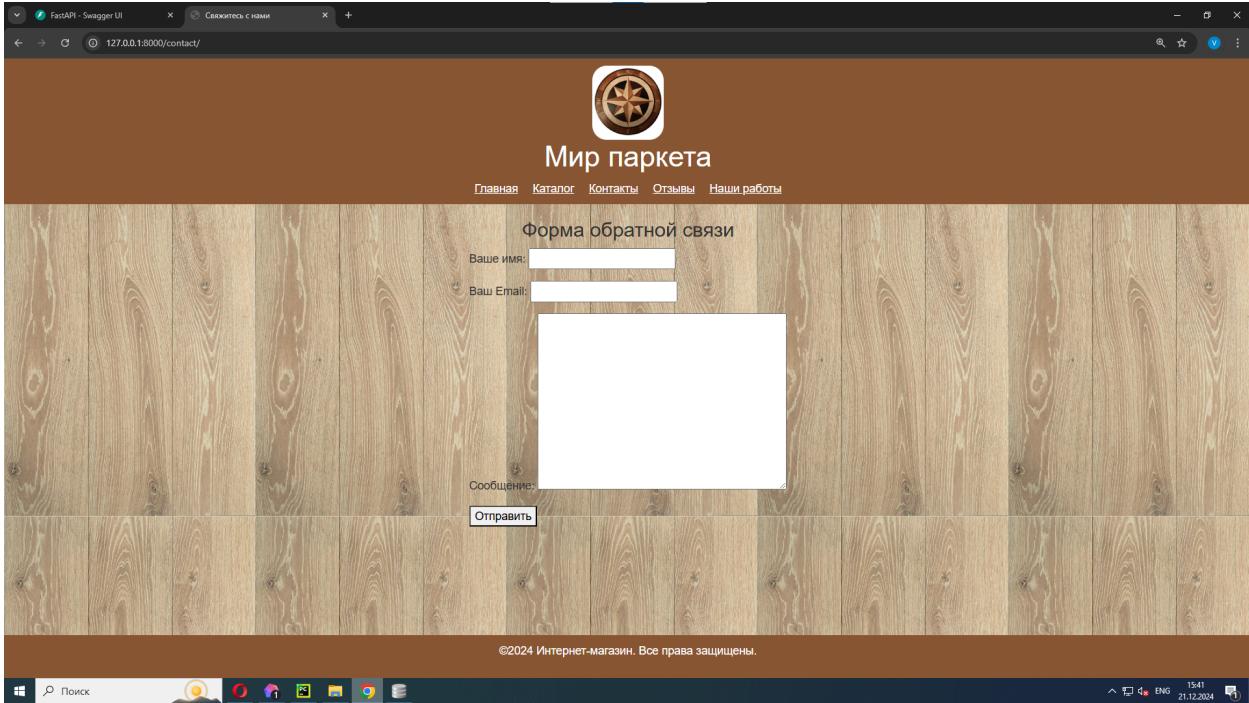


Рисунок 5 Форма с обратной связью.

## База данных.

После создания миграций Django создаст таблицы для моделей (рис 6; 7).

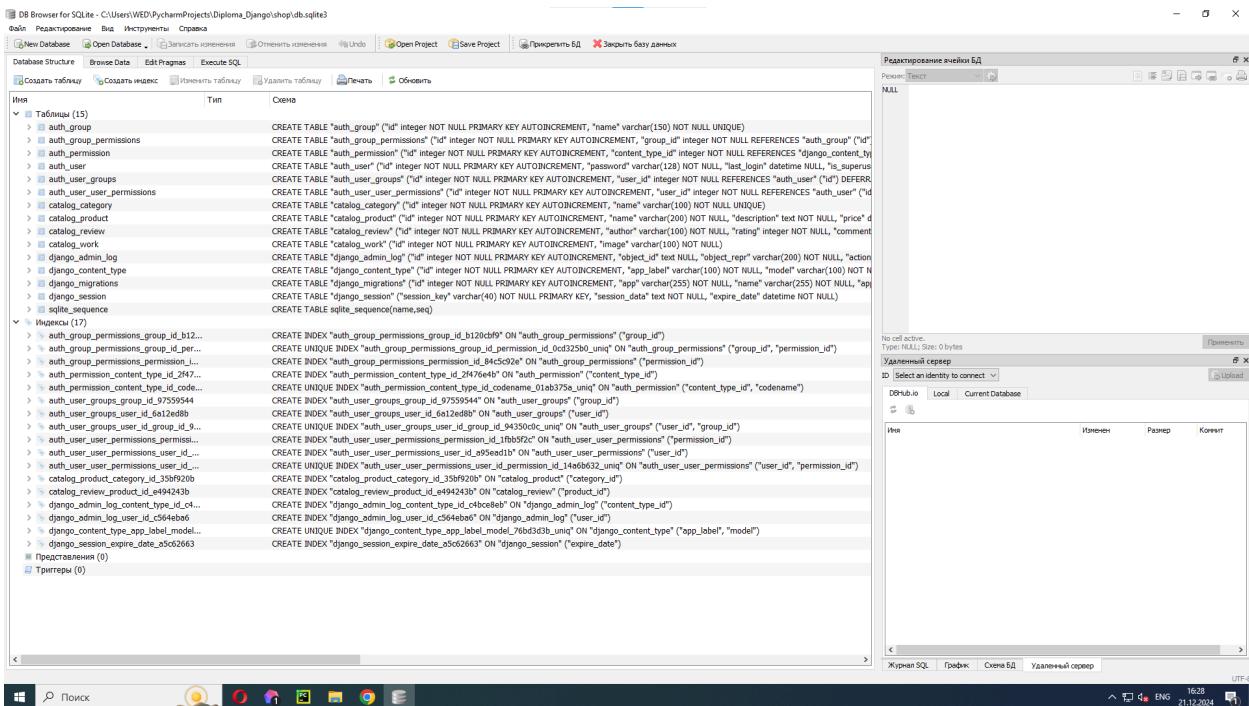


Рисунок 6 Таблицы базы данных

The screenshot shows the DB Browser for SQLite interface with the 'products' table selected. The table has columns: id, name, description, price, created\_at, and category\_id. An edit dialog is open on the right side, showing row 1 with the value '1' in the 'category\_id' field. The status bar at the bottom right shows '16:30 ENG 21.12.2024'.

id	name	description	price	created_at	category_id
Стр.	Выдел.	Выдел.	Выдел.	Выдел.	Выдел.
1	1 дуб селект 420x70x15 мм	Паркет высшего сорта без отбора по распилу, ...	3200	2024-12-17 19:28:46.413120	1
2	2 дуб натур 420x70x15 мм	Паркет без разбора по распилу, облагажий ...	2800	2024-12-17 19:29:29.498117	1
3	3 дуб рустик 420x70x15 мм	Паркет, отличающийся язычковой пестротой ...	1900	2024-12-17 19:30:03.143106	1
4	4 дуб радиок 420x70x15 мм	Отборный паркет, только радиального распила ...	3900	2024-12-17 19:30:47.555065	1
5	5 дуб селект 600-900x90x15 мм	Паркет без отбора по распилу с натуральными ...	3500	2024-12-17 19:32:18.165070	2
6	6 дуб натур 600-900x90x15 мм	Паркет без отбора по распилу и цветовой гамме. ...	3000	2024-12-17 19:33:03.892134	2
7	7 дуб рустик 600-900x90x15 мм	Паркет с сильными цветовыми контрастами и ...	2400	2024-12-17 19:33:42.940040	2
8	8 дуб радиок 600-900x90x15 мм	Отборный паркет старого радиального распила с ...	4200	2024-12-17 19:34:24.866403	2
9	9 KESTO ECO Standard 18мг	Дисперсионный клей для паркета...	5600	2024-12-17 19:36:27.566995	4
10	10 KESTO PRO 2K-FU	Двухкомпонентный полиуретановый клей без ...	5800	2024-12-17 19:38:18.391068	4
11	11 Berger Aqua-Seal 2KPU 5кг	Водоэмульсионный двухкомпонентный паркетный лак м...	26000	2024-12-17 19:39:34.644466	3
12	12 Berger Aqua-Seal CeramicStar 5 кг	Aqua-Seal CeramicStar - это 2-компонентный ...	20000	2024-12-17 19:40:38.926428	3
13	13 Berger Aqua-Seal EcoGold 5 кг	Berger-Заделка Аэра Син ЭкоМол - однокомпонентны...	16000	2024-12-17 19:41:32.352617	3
14	14 Saicos 3200 HardWax oil (2,5 кг)	Сайкос ХардВакс Ойл - масло с гвоздями ...	18000	2024-12-18 16:27:43.363675	5
15	15 Berger Classic Hard Oil (5кг)	Масло с гвоздями Berger Classic Hard Oil" ...	23000	2024-12-18 18:11:12.763916	5
16	16 дуб Натур 400-2000x150x20 мм	Массивная доска без отбора по распилу. ...	4000	2024-12-18 18:26:00.927528	6
17	17 дуб селект 400-2000x150x20 мм	Массивная доска без отбора по распилу. ...	4900	2024-12-18 18:26:56.643760	6
18	18 дуб рустик 400-2000x150x20 мм	Массивная доска без отбора по распилу. Яркие ...	3200	2024-12-18 18:27:32.801243	6

Рисунок 7 Таблица products

## Интерфейс админ-панели.

Administrator имеет полный доступ ко всем функциям админ-панели, в том числе к базе данных через интерфейс админ-панели (рис 8).

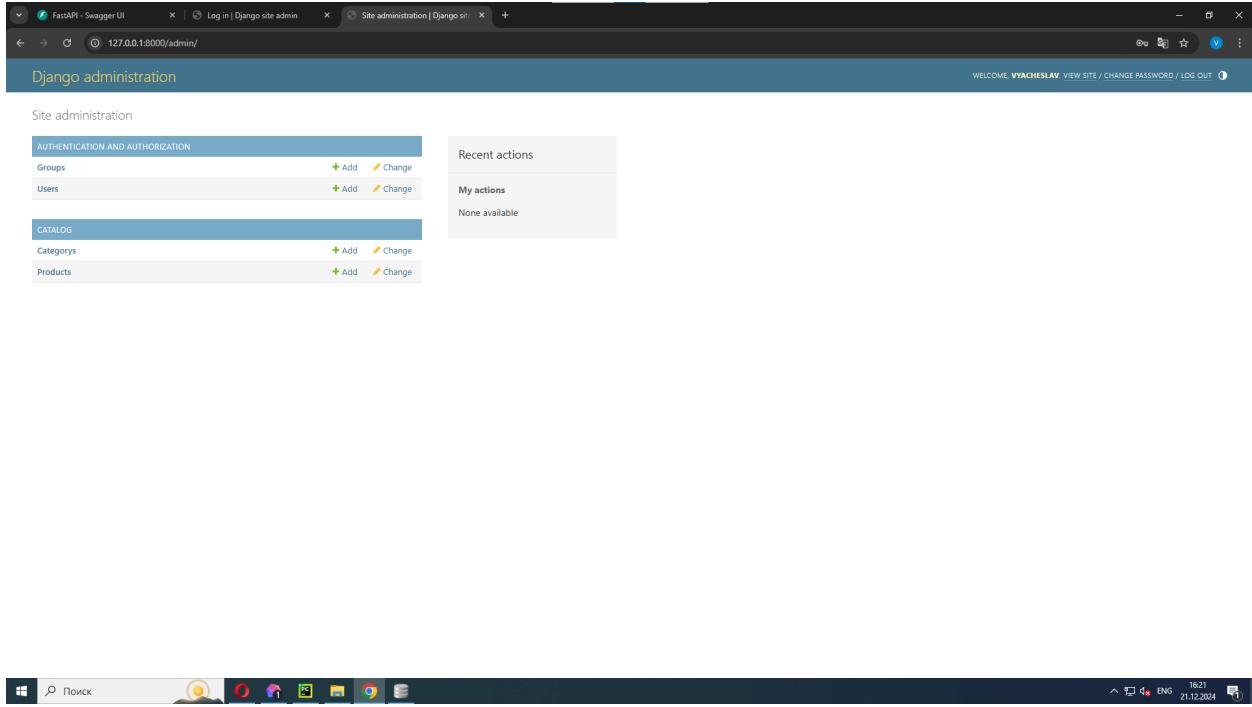


Рисунок 8 Админ-панель

## Структура проекта FastAPI

Проект FastAPI включает в себя следующие компоненты.

### Функционал приложения.

Основной функционал приложения FastAPI, это применение CRUD-операции, такие как Create (создание), Read (чтение), Update (редактирование) и Delete (удаление). Это основные методы работы с базами данных. Операции CRUD предназначены для редактирования данных программы (рис 9).

JSON (JavaScript Object Notation) используется для представления данных на сервере в текстовом формате. Он легко читается как людьми, так и машинами (рис 10).

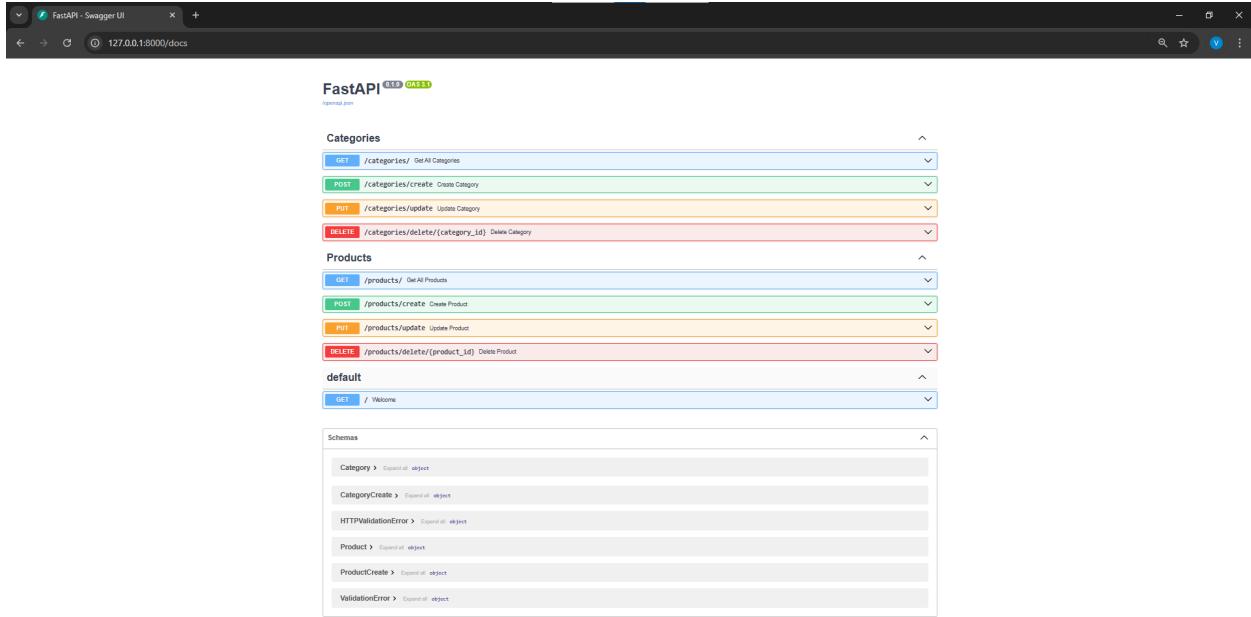
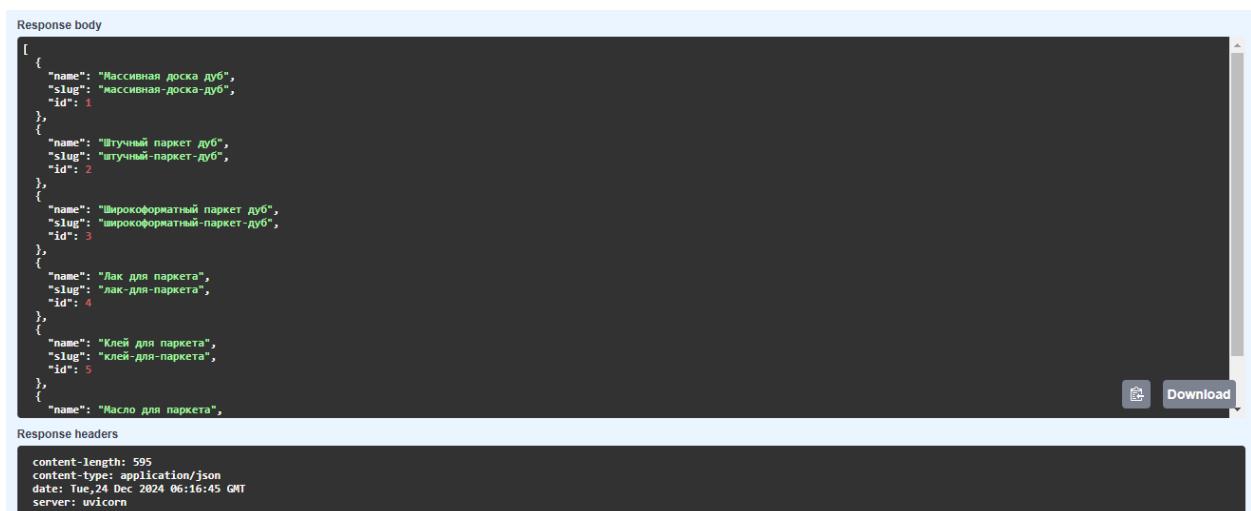


Рисунок 9 браузер Swagger и CRUD запросы

### *Рисунок 9 браузер Swagger и CRUD запросы*



*Рисунок 10 Данные в JSON формате*

## База данных.

Интеграция базы данных (рис 11):

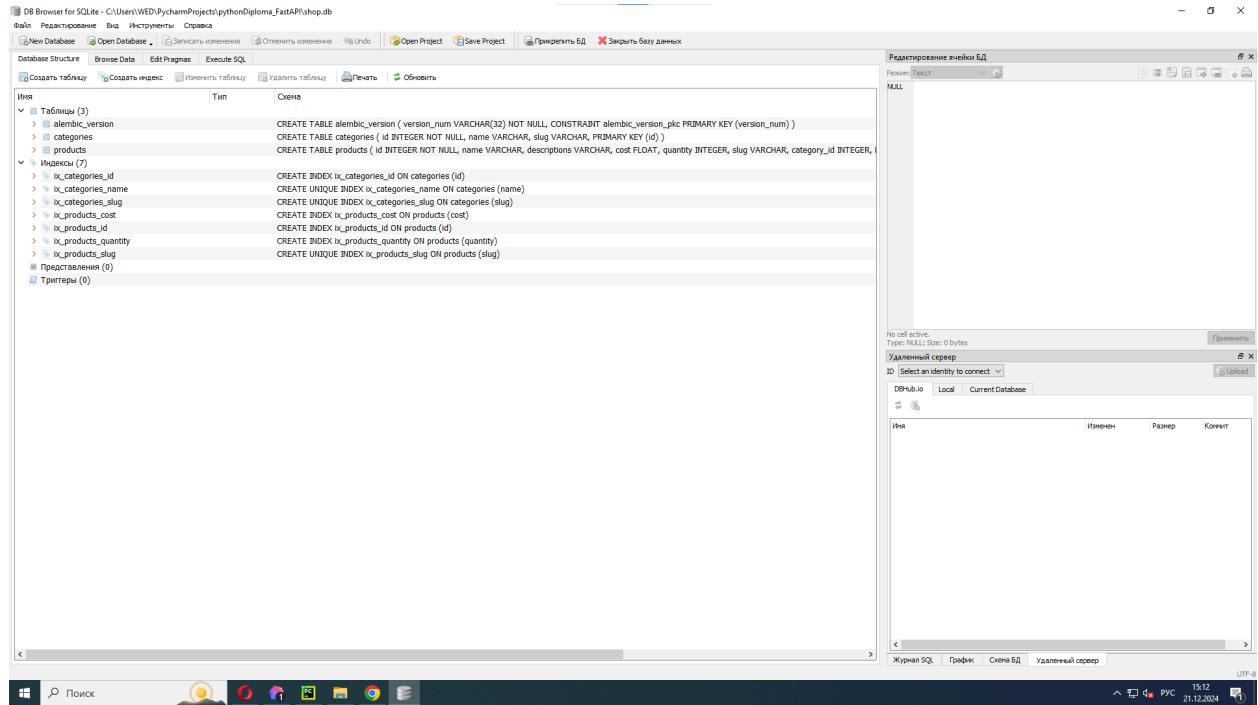


Рисунок 11 sqlite3 вид таблиц из базы данных

## Структура проекта Aiogram

Проект по созданию чат-бота содержит в себе следующие компоненты:

### Функционал чат-бота.

Чат-бот – это специальный программный агент, который может взаимодействовать с пользователями путем автоматизированного обмена сообщениями. Реализован старт проекта с приветствием пользователя по nic (рис 12; 13; 14):

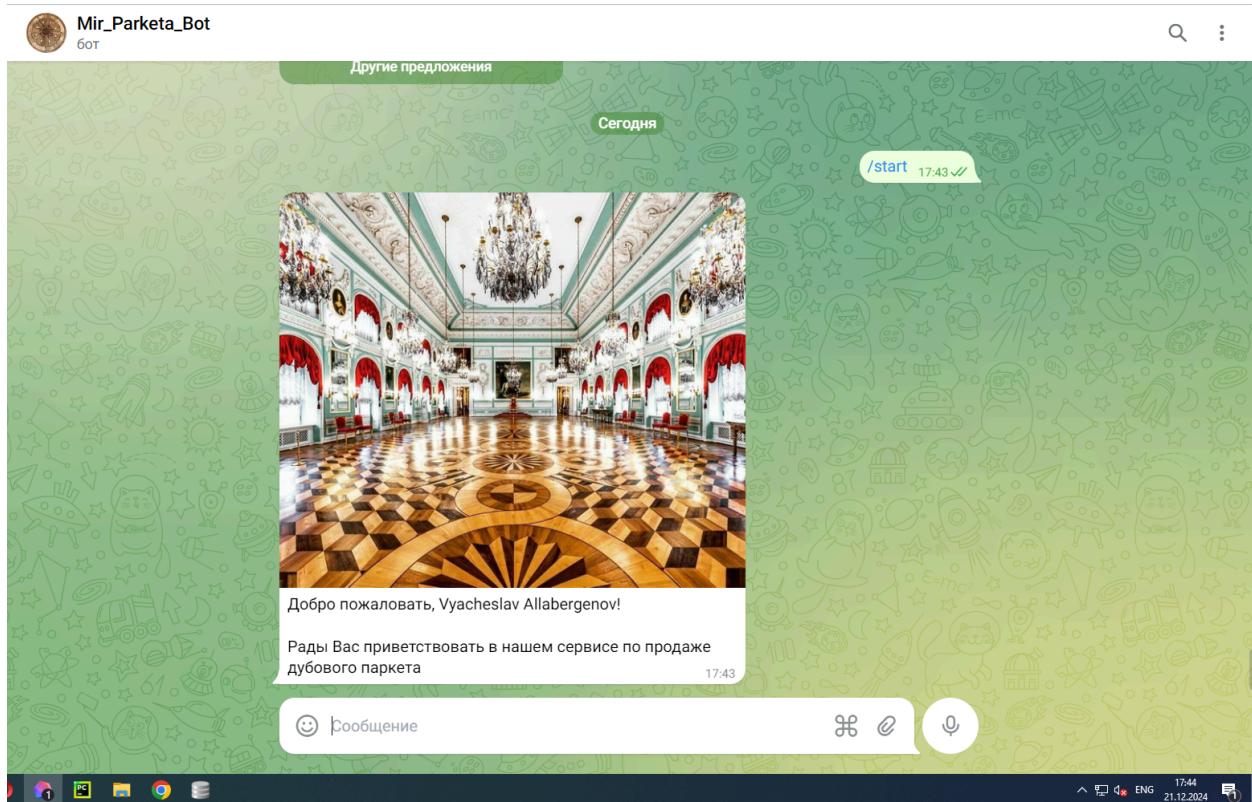


Рисунок 12 Блок приветствия

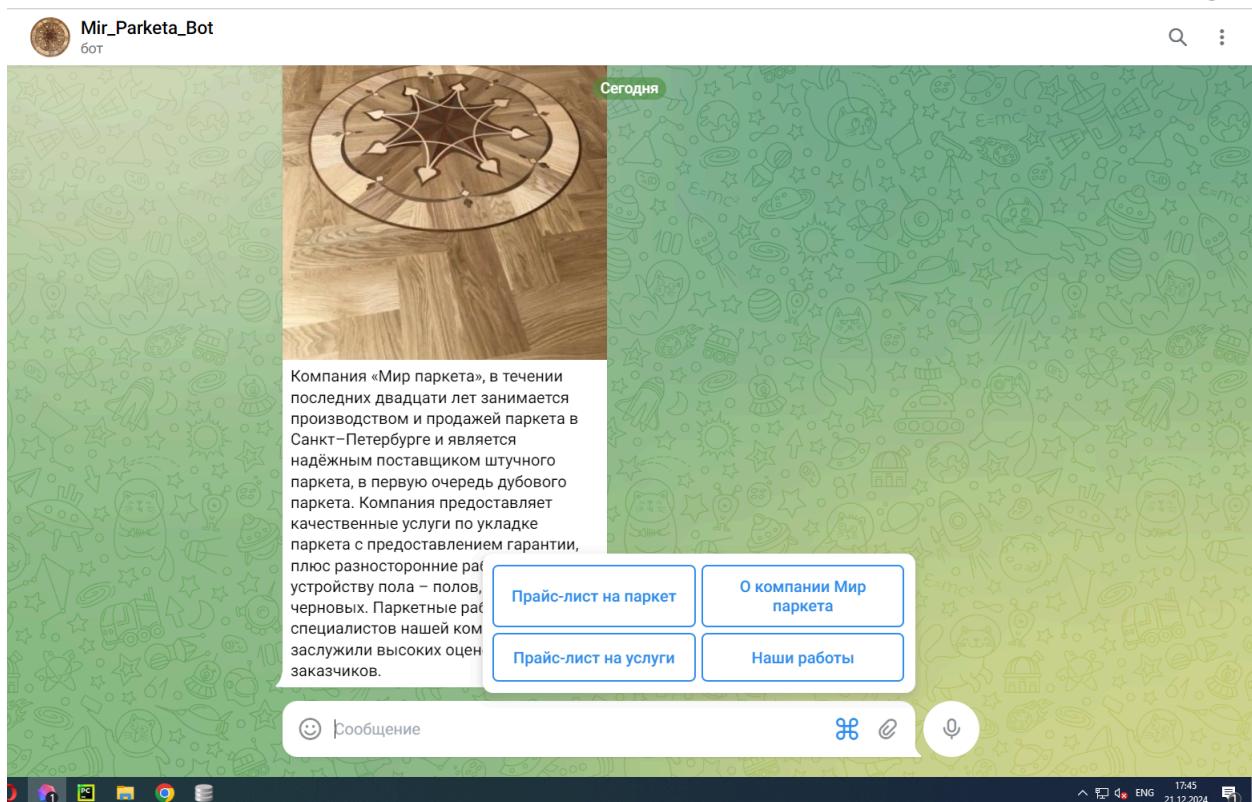


Рисунок 13 Клавиатура меню

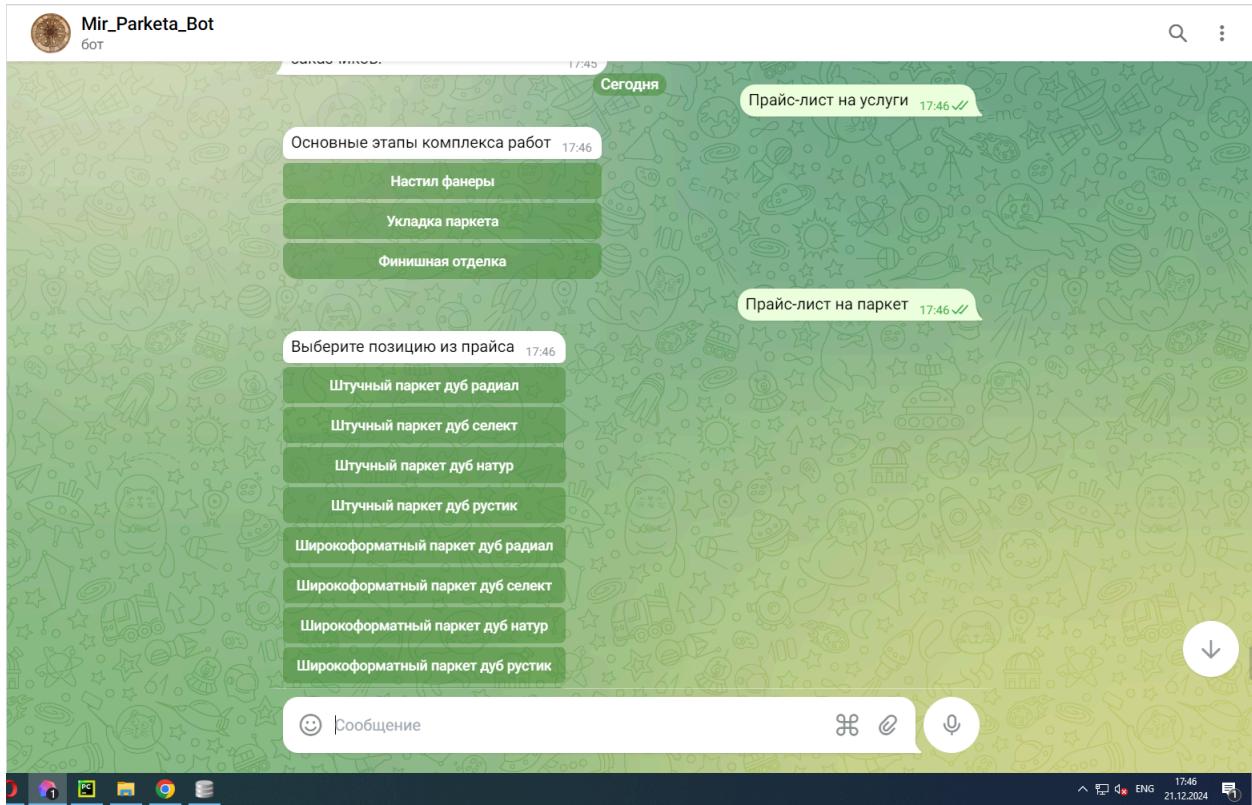


Рисунок 14 Каталог

## База данных.

Реализовано создание базы данных администратора для работы с пользователями, такие как регистрация, статус, блокировка и разблокировка (рис 15).

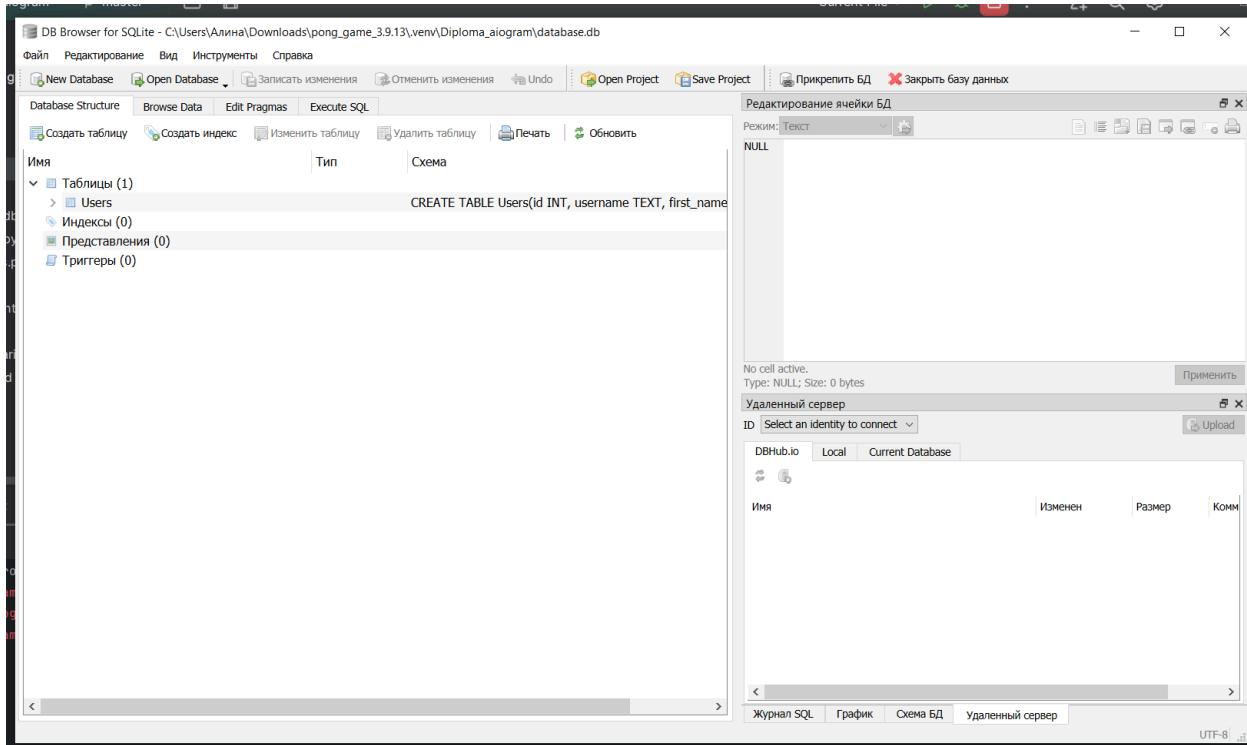


Рисунок 15 Таблица в базе данных

## Заключение.

Обзор выполненной работы. Проектирование и разработка веб-приложений успешно завершены в соответствии с техническими требованиями. Приложения содержат начальный функционал основополагающих задач. Применялся простой и удобный способ хранения и администрирования данных.

## Приложение №1 Список необходимых библиотек

### Django:

```
asgiref==3.8.1
Django==4.2.17
pillow==11.0.0
sqlparse==0.5.3
typing_extensions==4.12.2
tzdata==2024.2
```

## FastApi:

```
alembic==1.14.0
annotated-types==0.7.0
anyio==4.7.0
certifi==2024.8.30
click==8.1.7
colorama==0.4.6
dnspython==2.7.0
email_validator==2.2.0
fastapi==0.115.6
fastapi-cli==0.0.6
greenlet==3.1.1
h11==0.14.0
httpcore==1.0.7
httplib==0.6.4
httpx==0.28.1
idna==3.10
itsdangerous==2.2.0
Jinja2==3.1.4
Mako==1.3.8
markdown-it-py==3.0.0
MarkupSafe==3.0.2
mdurl==0.1.2
orjson==3.10.12
pydantic==2.10.3
pydantic-extra-types==2.10.1
pydantic-settings==2.6.1
pydantic_core==2.27.1
Pygments==2.18.0
python-dotenv==1.0.1
python-multipart==0.0.19
PyYAML==6.0.2
rich==13.9.4
rich-toolkit==0.12.0
shellingham==1.5.4
sniffio==1.3.1
SQLAlchemy==2.0.36
starlette==0.41.3
typer==0.15.1
typing_extensions==4.12.2
ujson==5.10.0
```

```
uvicorn==0.32.1
watchfiles==1.0.0
websockets==14.1
```

## Aiogram:

```
aiogram==2.25.1
aiohttp==3.8.6
aiosignal==1.3.2
async-timeout==4.0.3
attrs==24.3.0
Babel==2.9.1
certifi==2024.12.14
charset-normalizer==3.4.0
frozenlist==1.5.0
idna==3.10
magic-filter==1.0.12
multidict==6.1.0
propcache==0.2.1
pytz==2024.2
typing_extensions==4.12.2
yarl==1.18.3
```