

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»
(БГТУ им. В.Г.Шухова)**

Лабораторная работа по теме
«Серверное программирование. Django»
дисциплина:
Технологии web-программирования

Выполнил: ст. группы ВТ-41
Фаракшин Н. Р.
Проверил: ст. пр. Картамышев С.В.

Лабораторная работа № 3

Серверное программирование. Django

Цель работы: познакомиться с основами backend разработки web-приложений. Научится писать скрипты на языке Python. Познакомиться с основами работы docker. Познакомиться с фреймворком Django и научиться разворачивать проект, производить его настройку. Научится работать с API в приложении Postman.

Задание к лабораторной работе:

1. Развернуть базовое приложение Django.
2. Настроить конфигурацию работы приложения с docker.
3. Добавить модуль для работы с API.
4. Добавить несколько контроллеров со статическими данными.
5. Продемонстрировать работу API в Postman.

Демонстрация работы приложения

```
PS C:\Users\Yanar\Documents\Github\web_labs\backend> docker-compose up
Starting backend_db_1 ... done
Starting backend_web_1 ... done
Attaching to backend_db_1, backend_web_1
db_1 | PostgreSQL Database directory appears to contain a database; Skipping initialization
db_1 |
db_1 | 2020-11-20 22:03:25.190 UTC [1] LOG: starting PostgreSQL 13.1 (Debian 13.1-1.pgdg100+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 8.3.0-6) 8.3.0, 64-bit
db_1 | 2020-11-20 22:03:25.190 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
db_1 | 2020-11-20 22:03:25.190 UTC [1] LOG: listening on IPv6 address ":::", port 5432
db_1 | 2020-11-20 22:03:25.195 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
db_1 | 2020-11-20 22:03:25.202 UTC [25] LOG: database system was shut down at 2020-11-20 22:02:53 UTC
db_1 | 2020-11-20 22:03:25.208 UTC [1] LOG: database system is ready to accept connections
web_1 | Watching for file changes with StatReloader
web_1 | Performing system checks...
web_1 |
web_1 | System check identified no issues (0 silenced).
web_1 |
web_1 | You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
web_1 | Run 'python manage.py migrate' to apply them.
web_1 | November 20, 2020 - 22:03:26
web_1 | Django version 3.1.3, using settings 'blog.settings'
web_1 | Starting development server at http://0.0.0.0:8000/
web_1 | Quit the server with CONTROL-C.
```

Рис. 1 Запущено два контейнера Docker с помощью Docker-Compose

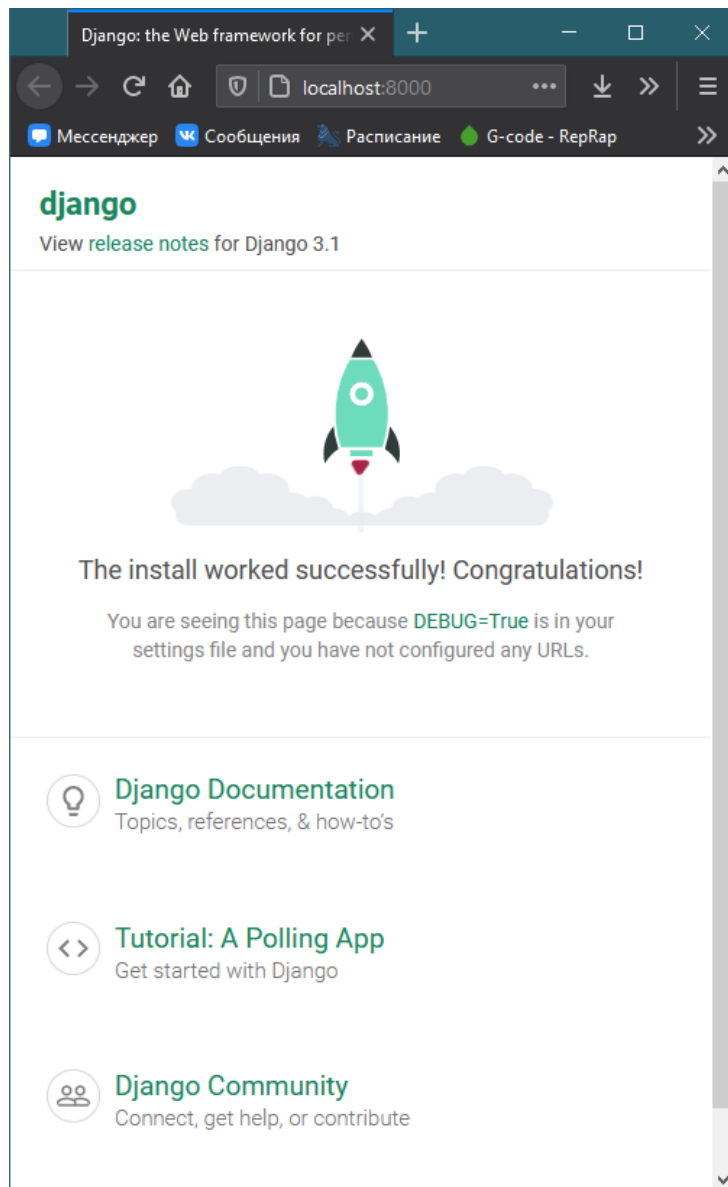


Рис. 2 Приложение Django корректно работает

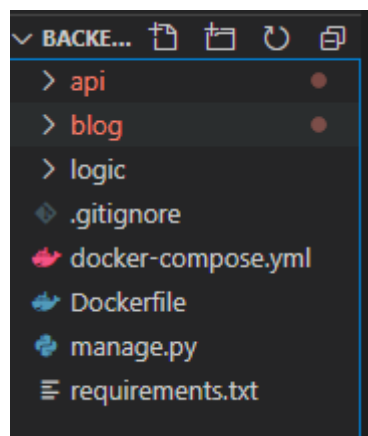


Рис. 3 Структура проекта

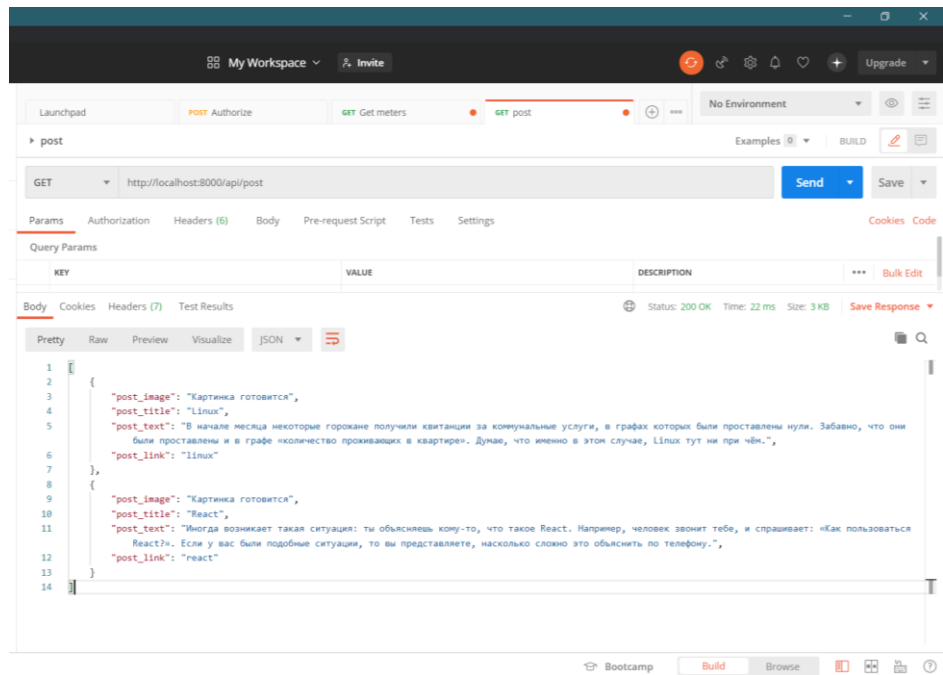


Рис. 4 Запрос к API в приложении Postman успешно получает данные

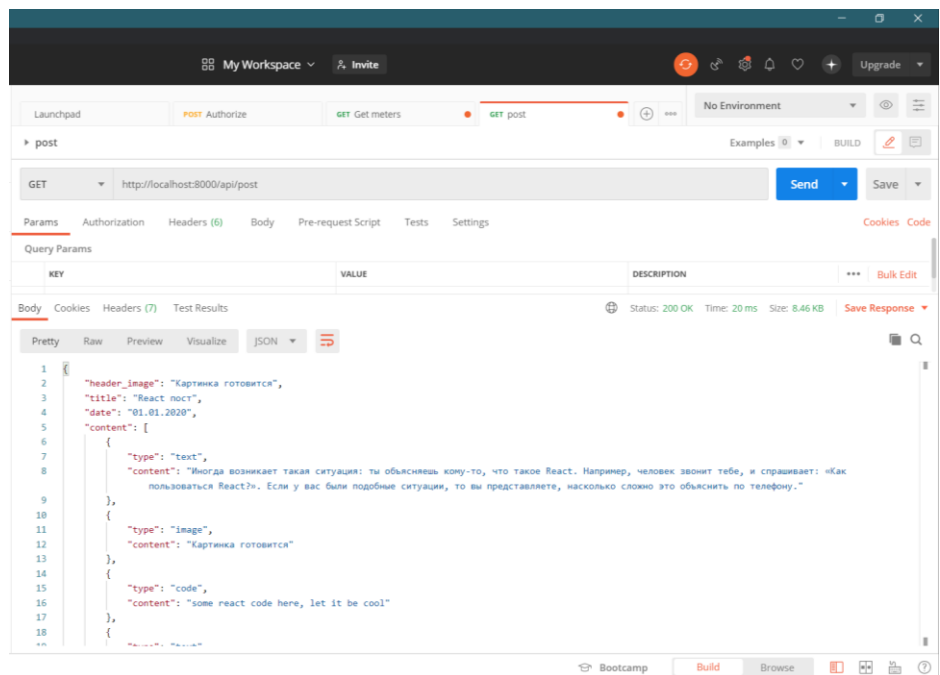


Рис. 5 Запрос к другой конечной точке API

Приложение

Исходный код файлов Dockerfile и docker-compose.yml. Код отдающего данные контроллера.

docker-compose.yml

```
version: "3.8"

services:
  db:
    image: postgres
    environment:
      - POSTGRES_DB=postgres
      - POSTGRES_USER=postgres
      - POSTGRES_PASSWORD=postgres
  web:
    build: .
    command: python manage.py runserver 0.0.0.0:8000
    volumes:
      - ./code
    ports:
      - "8000:8000"
    depends_on:
      - db
```

Dockerfile

```
FROM python:3
ENV PYTHONUNBUFFERED=1
WORKDIR /code
COPY requirements.txt /code/
RUN pip install -r requirements.txt
COPY . /code/
```

views.py

```
def static_post_list(request):
    response = json.dumps(
        [
            {
                "post_image": "Картинка готовится",
                "post_title": "Linux",
                "post_text": "В начале месяца некоторые горожане получили..."
                "post_link": "linux"
            },
            {
                "post_image": "Картинка готовится",
                "post_title": "React",
                "post_text": "Иногда возникает такая ситуация: ... ",
                "post_link": "react"
            },
        ]
    )
```

```

    ]
)
return HttpResponse(response)

def static_post(request):
    response = json.dumps(
        {
            "header_image": "Картинка готовится",
            "title": "React пост",
            "date": "01.01.2020",
            "content": [
                {
                    "type": "text",
                    "content": "Иногда возникает такая ситуация: ..."
                },
                {
                    "type": "image",
                    "content": "Картинка готовится"
                },
                {
                    "type": "code",
                    "content": "some react code here, let it be cool"
                },
                {
                    "type": "text",
                    "content": "Нелегка в России судьба React: ..."
                }
            ]
        }
    )
    return HttpResponse(response)

```