Kazakh British Technical University

Web Application Development

Assignment II: Exploring Django with Docker

by Moldir Polat

October 13th 2024

Table of contents

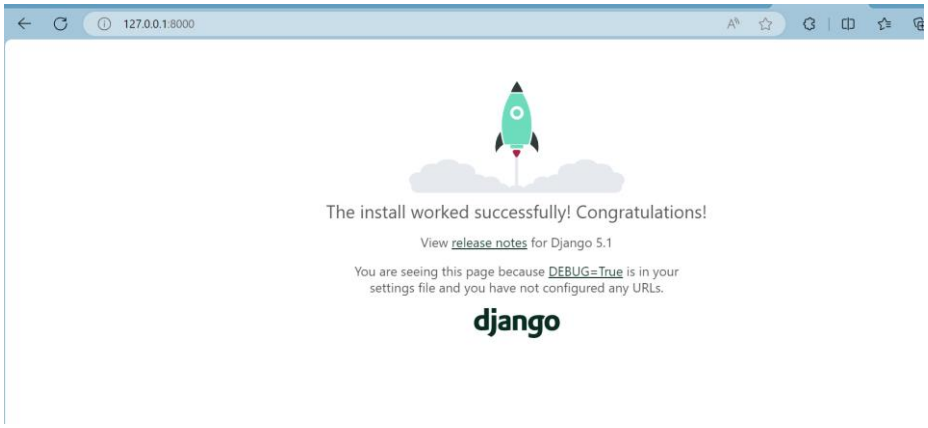
Introduction

Assignment has 3 sections to complete: Docker Compose, Docker Networking and Volumes, Django Application setup.

The goal of this assignment is to gain hands-on experience with Django and Docker, focusing on Docker Compose, Docker networking, and volumes. Students will set up a Django application within a Docker environment and document the process.

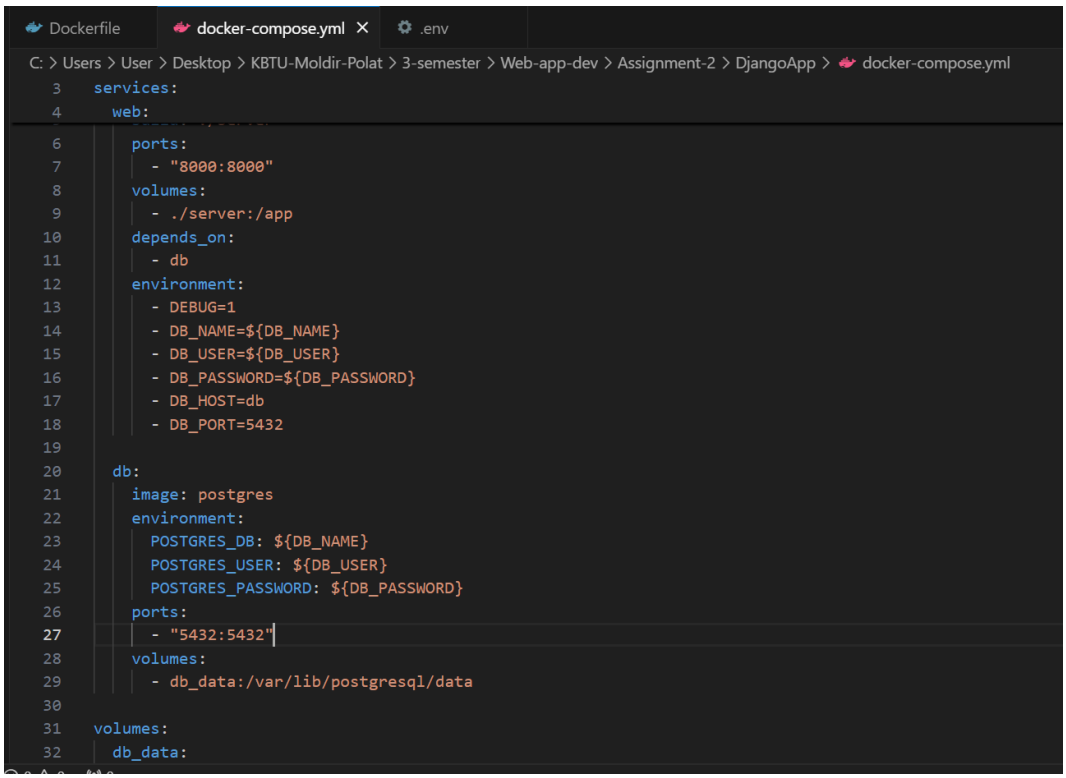
Docker Compose

At first we created Django project with Django app locally.

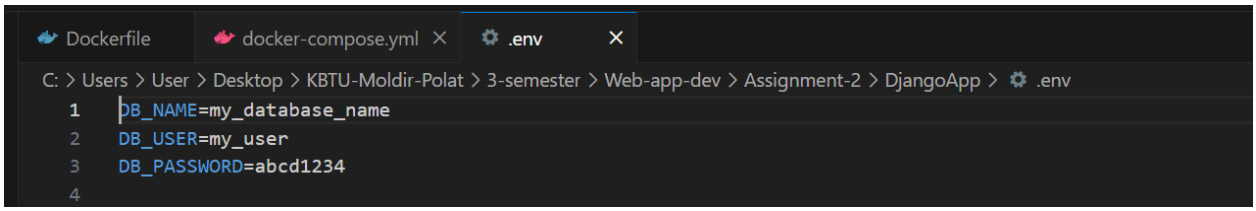


- **Configuration:** We can define and manage multiple Docker containers in docker-compose.yml file. Here we will have web container and database cantainer.

docker-compose.yml file content



.env file content:



- **Build and Run:** We use command docker-compose up --build

It pulls the Python image, installs Django and PostgreSQL inside the containers, sets up the services.

web service builds the Django app from the server directory, runs the development server on port 8000, and uses environment variables for database configuration.

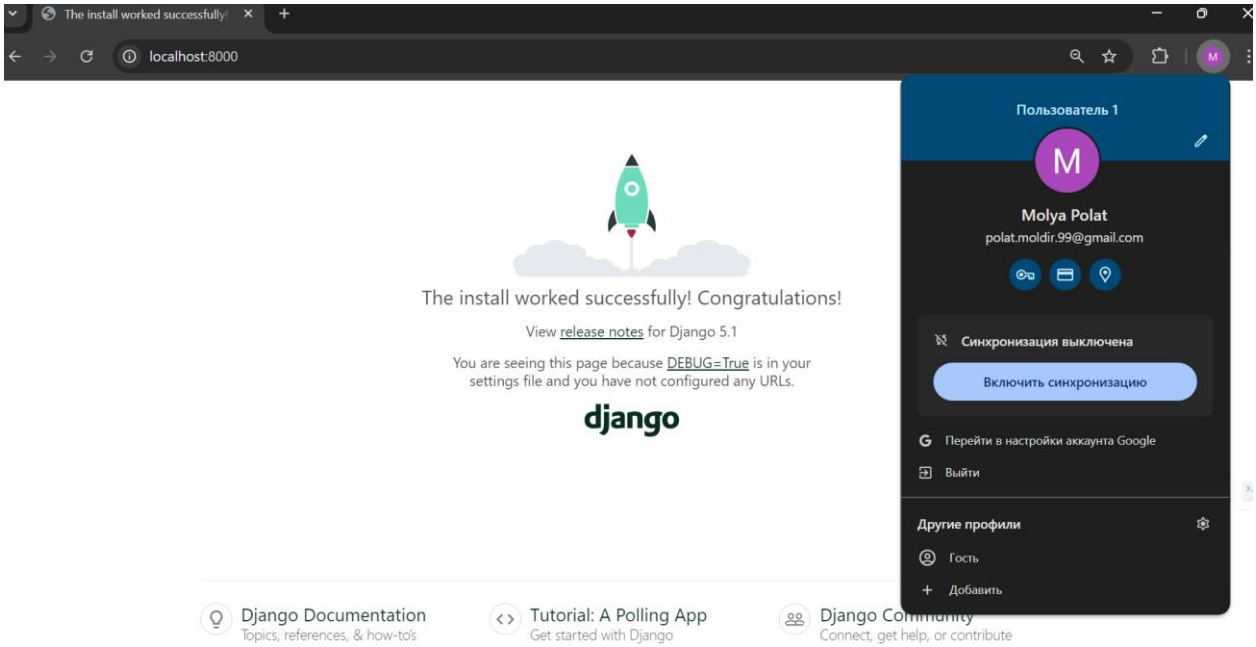
db service uses the official PostgreSQL image, exposing port 5432 and initializing with environment variables from the .env file for database credentials.

volumes persistent storage for PostgreSQL is configured using a named volume (postgres_data).

Built containers (output of *docker-compose up -d –build* command):

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoApp>docker-compose up -d --build
time="2024-10-12T01:59:27+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-2\\DjangoApp\\docker-compose.yml: the attribute 'version' is obsolete, it w
ill be ignored, please remove it to avoid potential confusion"
[+] Building 130.5s (11/11) FINISHED                                                                                                                                                    docker:desktop-linux
 => [web internal] load build definition from Dockerfile                                                                                                                                          0.0s
 => => transferring dockerfile: 224B                                                                                                                                                              0.0s
 => [web internal] load metadata for docker.io/library/python:3.10                                                                                                                                3.7s
 => [web internal] load .dockerignore                                                                                                                                                             0.0s
 => => transferring context: 2B                                                                                                                                                                   0.0s
 => [web 1/5] FROM docker.io/library/python:3.10@sha256:aaacc373f21abcbc67a22d203dab7fb797c016cc4c1daa7d2dec359dcbf4aea1                                                                         105.0s
 => => resolve docker.io/library/python:3.10@sha256:aaacc373f21abcbc67a22d203dab7fb797c016cc4c1daa7d2dec359dcbf4aea1                                                                               0.0s
 => => sha256:c4907bcc004d711316df51048c4c8625e5b38226c8a4dd22f2ec6965da439d18 2.33kB / 2.33kB                                                                                                    0.0s
 => => sha256:c788c1edc88f3c7244e1b75cae5a7f8df8cf33461e25d67e916dc76a3218b2cf 5.94kB / 5.94kB                                                                                                    0.0s
 => => sha256:a173f2aee8e962ea19db1e418ae84a0c9f71480b51f768a19332dfa83d7722a5 64.39MB / 64.39MB                                                                                                 60.8s
 => => sha256:aaacc373f21abcbc67a22d203dab7fb797c016cc4c1daa7d2dec359dcbf4aea1 9.08kB / 9.08kB                                                                                                    0.0s
 => => sha256:cdd62bf39133c498a16f7a7b1b6555ba43d02b2511c508fa4c0a9b1975ffe20e 49.56MB / 49.56MB                                                                                                  7.5s
 => => sha256:a47cff7f31e941e78bf63ca19f0811b675283e2c00ddea10c57f78d93b2bc343 24.05MB / 24.05MB                                                                                                 24.0s
 => => sha256:01272fe8adbacc44afd2b92994b31c40a151f4324ca392050d9e8d580927dd32 211.27MB / 211.27MB                                                                                               57.3s
 => => extracting sha256:cdd62bf39133c498a16f7a7b1b6555ba43d02b2511c508fa4c0a9b1975ffe20e                                                                                                          7.5s
 => => sha256:2503de423c42c91e553f5faf8e7d04db973ea96d6d3865e13ece952d27439b0c 6.16MB / 6.16MB                                                                                                   29.9s
 => => extracting sha256:a47cff7f31e941e78bf63ca19f0811b675283e2c00ddea10c57f78d93b2bc343                                                                                                         2.0s
 => => sha256:3f9262f6ef06204821bf90796cb4b90743cb3dff874b3fb91c11ede1d0094eb4 20.24MB / 20.24MB                                                                                                 51.7s
 => => sha256:fddd6c71e7646d2c20e2b785ddcf2f6b6ec2a0fd76591e838fd7059638ea27fb 248B / 248B                                                                                                       52.9s
 => => extracting sha256:a173f2aee8e962ea19db1e418ae84a0c9f71480b51f768a19332dfa83d7722a5                                                                                                         8.5s
 => => extracting sha256:01272fe8adbacc44afd2b92994b31c40a151f4324ca392050d9e8d580927dd32                                                                                                        23.7s
 => => extracting sha256:2503de423c42c91e553f5faf8e7d04db973ea96d6d3865e13ece952d27439b0c                                                                                                         5.5s
 => => extracting sha256:3f9262f6ef06204821bf90796cb4b90743cb3dff874b3fb91c11ede1d0094eb4                                                                                                         3.4s
 => => extracting sha256:fddd6c71e7646d2c20e2b785ddcf2f6b6ec2a0fd76591e838fd7059638ea27fb                                                                                                         0.0s
 => [web internal] load build context                                                                                                                                                             0.0s
 => => transferring context: 1.24kB                                                                                                                                                               0.0s
 => [web 2/5] WORKDIR /app                                                                                                                                                                        2.1s
 => [web 3/5] COPY requirements.txt .                                                                                                                                                             0.6s
 => [web 4/5] RUN pip install --no-cache-dir -r requirements.txt                                                                                                                                 17.1s
 => [web 5/5] COPY . .                                                                                                                                                                            0.1s
 => [web] exporting to image                                                                                                                                                                      1.7s
 => => exporting layers                                                                                                                                                                           1.6s
 => => writing image sha256:badce551321795bc3bd6ceeb8f8cb9975b1faadab9f5a1cd92ccba2bfea6d788                                                                                                      0.0s
 => => naming to docker.io/library/djangoapp-web                                                                                                                                                  0.0s
 => [web] resolving provenance for metadata file                                                                                                                                                  0.0s
[+] Running 4/4
 ✓Network djangoapp_default    Created                                                                                                                                                            0.4s
 ✓Volume "djangoapp_db_data"   Created                                                                                                                                                            0.1s
 ✓Container djangoapp-db-1     Started                                                                                                                                                            1.8s
 ✓Container djangoapp-web-1    Started                                                                                                                                                            1.7s
```

Checking *localhost:8000*



Postgre sql checking:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoApp>docker exec -it djangoapp-db-1 psql -U my_user -d my_database_name
psql (17.0 (Debian 17.0-1.pgdg120+1))
Type "help" for help.

my_database_name=# \dt
Did not find any relations.
```

All services are running:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoApp>docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED         STATUS         PORTS                    NAMES
640bbb1acdd4   djangoapp-web  "python manage.py ru…"   7 minutes ago   Up 7 minutes   0.0.0.0:8000->8000/tcp   djangoapp-web-1
6ca91b409d5f   postgres       "docker-entrypoint.s…"   7 minutes ago   Up 7 minutes   0.0.0.0:5432->5432/tcp   djangoapp-db-1
```

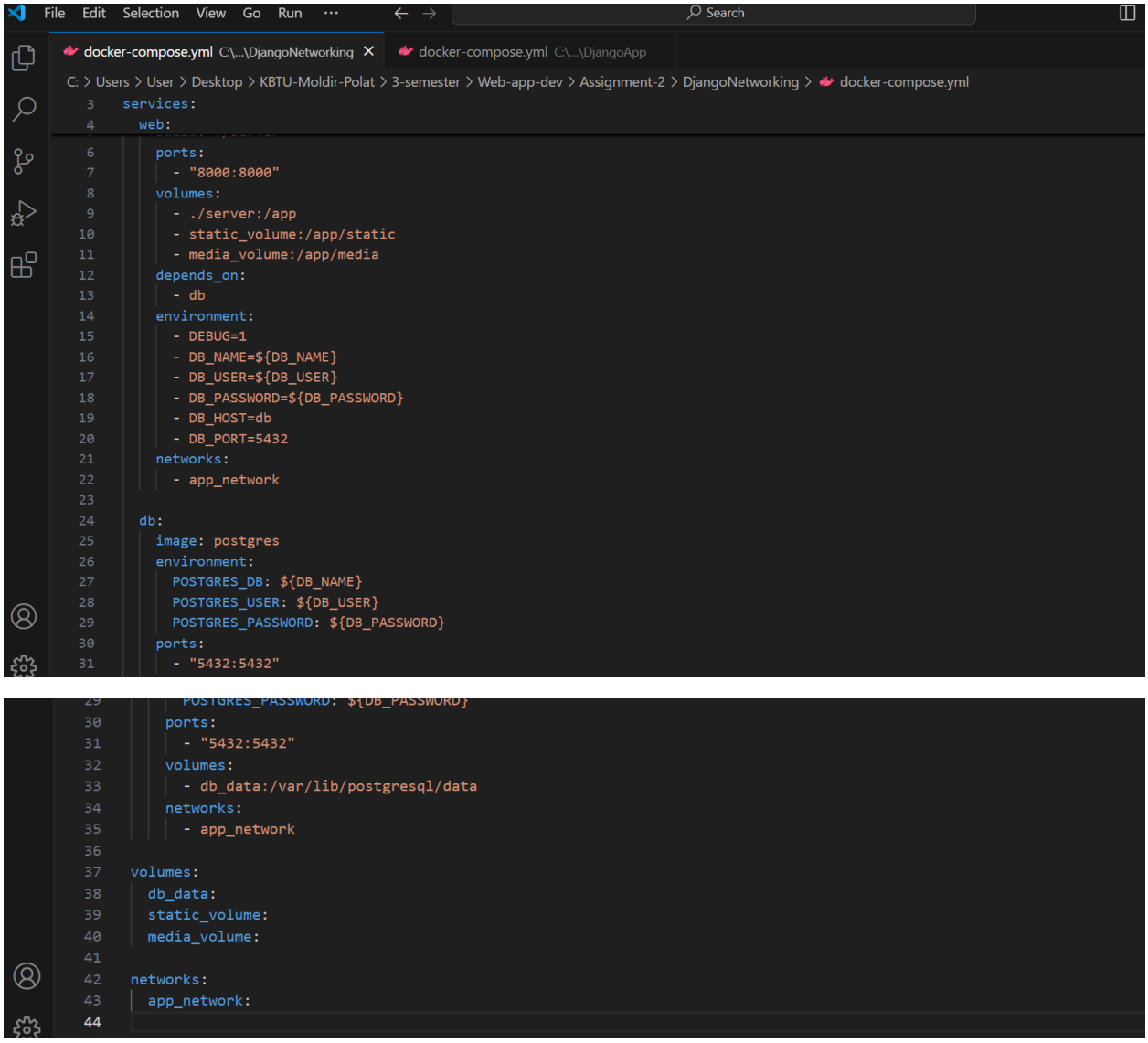
Challenges:

Django required python version 3.10.

Docker Networking and Volumes

- **Set Up Docker Networking**

We modified previous *docker-compose.yml* file as following:



We added *networks:* section to both web and db services to specify a custom network called *app_network*. This will allow these services to communicate with each other.

We updated *setting.py* to connect with Postgresql service:

```
docker-compose.yml   settings.py 1 ×
C: > Users > User > Desktop > KBTU-Moldir-Polat > 3-semester > Web-app-dev > Assignment-2 > DjangoNetworking > server > djando_proj
 75   # Database
 76   # https://docs.djangoproject.com/en/5.1/ref/settings/#databases
 77
 78   DATABASES = {
 79       'default': {
 80           'ENGINE': 'django.db.backends.postgresql',
 81           'NAME': config('DB_NAME'),
 82           'USER': config('DB_USER'),
 83           'PASSWORD': config('DB_PASSWORD'),
 84           'HOST': 'db',
 85           'PORT': '5432',
 86       }
 87   }
 88
 89
```

Started containers with Docker compose:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose up -d
time="2024-10-13T01:26:23+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
[+] Building 0.0s (0/0)  docker:desktop-linux
[+] Building 5.1s (12/12) FINISHED                                                                          docker:desktop-linux
 => [web internal] load build definition from Dockerfile                                                                    0.0s
 => => transferring dockerfile: 224B                                                                                        0.0s
 => [web internal] load metadata for docker.io/library/python:3.10                                                          4.6s
 => [web auth] library/python:pull token for registry-1.docker.io                                                           0.0s
 => [web internal] load .dockerignore                                                                                       0.0s
 => => transferring context: 2B                                                                                             0.0s
 => [web internal] load build context                                                                                       0.0s
 => => transferring context: 5.30kB                                                                                         0.0s
 => [web 1/5] FROM docker.io/library/python:3.10@sha256:aaacc373f21abcbc67a22d203dab7fb797c016cc4c1daa7d2dec359dc           0.0s
 => CACHED [web 2/5] WORKDIR /app                                                                                           0.0s
 => CACHED [web 3/5] COPY requirements.txt .                                                                                0.0s
 => CACHED [web 4/5] RUN pip install --no-cache-dir -r requirements.txt                                                     0.0s
 => [web 5/5] COPY . .                                                                                                      0.1s
 => [web] exporting to image                                                                                                0.1s
 => => exporting layers                                                                                                     0.0s
 => => writing image sha256:e4f4ac2b8e16233f2d59bcbc7f4e105c20f49bc52d02660c89a65c9ad0d578c6                                0.0s
 => => naming to docker.io/library/djangonetworking-web                                                                     0.0s
 => [web] resolving provenance for metadata file                                                                            0.0s
[+] Running 6/6
 ✔Network djangonetworking_app_network     Created                                                                          0.1s
 ✔Volume "djangonetworking_static_volume"  Created                                                                          0.0s
 ✔Volume "djangonetworking_media_volume"   Created                                                                          0.0s
 ✔Volume "djangonetworking_db_data"        Created                                                                          0.0s
 ✔Container djangonetworking-db-1          Started                                                                          1.5s
 ✔Container djangonetworking-web-1         Started                                                                          1.7s
```

Containers are running:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker ps
CONTAINER ID   IMAGE                 COMMAND                  CREATED          STATUS          PORTS                    NAMES
b4d47e7b10af   djangonetworking-web  "python manage.py ru…"   4 seconds ago    Up 3 seconds    0.0.0.0:8000->8000/tcp   djangonetworki
ng-web-1
2f0ec94ae60f   postgres              "docker-entrypoint.s…"   5 seconds ago    Up 4 seconds    0.0.0.0:5432->5432/tcp   djangonetworki
ng-db-1
```

To verify the connection we created models:

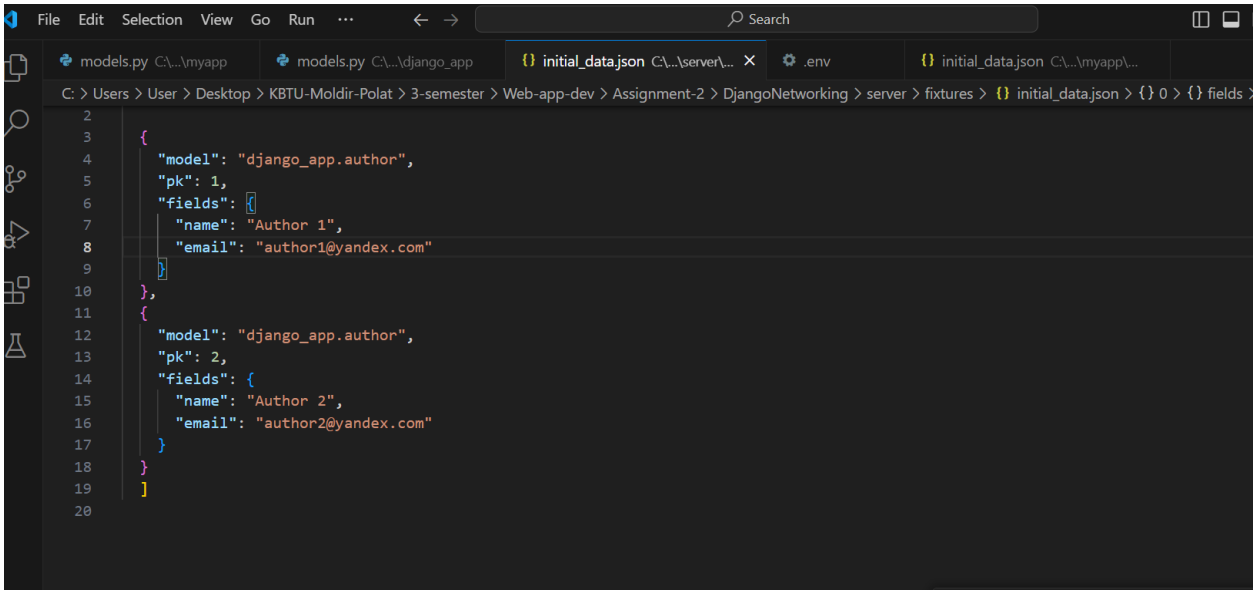
```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose exec web python manage.py
 makemigrations
time="2024-10-13T01:57:53+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
Migrations for 'django_app':
  django_app/migrations/0001_initial.py
    + Create model Author
    + Create model Publisher
    + Create model Book
    + Create model Review
```

Migrated data:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose exec web python manage.py
 migrate
time="2024-10-13T01:58:57+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, django_app, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying django_app.0001_initial... OK
  Applying sessions.0001_initial... OK
```

Here we loaded initial data fom fixtures:

docker-compose exec web python manage.py loaddata fixtures/initial_data.json




```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose exec web python manage.py
 loaddata fixtures/initial_data.json
time="2024-10-13T02:00:22+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
Installed 2 object(s) from 1 fixture(s)
```

Now let’s check databse directly using Postgre CLI:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose exec db psql -U my_user -
d my_database_name
time="2024-10-13T02:09:31+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
psql (17.0 (Debian 17.0-1.pgdg120+1))
Type "help" for help.

my_database_name=# SELECT * FROM django_app_author;
 id |   name   |       email
----+----------+--------------------
  1 | Author 1 | author1@yandex.com
  2 | Author 2 | author2@yandex.com
(2 rows)

my_database_name=#
```

Exactly 2 objects are present in database which means that aour application successfully connected with db service – successful networking.

- **Implement Docker Volumes**

We have already configured a volume for PostgreSQL data in docker-compose.yml file in previous task. This volume will ensure that PostgreSQL database keeps its data even if the container is stopped or even deleted.

```yaml
  db:
    image: postgres
    environment:
      POSTGRES_DB: ${DB_NAME}
      POSTGRES_USER: ${DB_USER}
      POSTGRES_PASSWORD: ${DB_PASSWORD}
    ports:
      - "5432:5432"
    volumes:
      - db_data:/var/lib/postgresql/data
    networks:
      - app_network
```

For web service we added 2 volumes: *media_volume* – will store uploaded files; *static_volume* – will store static files.

```yaml
services:
  web:
    build: ./server
    ports:
      - "8000:8000"
    volumes:
      - ./server:/app
      - static_volume:/app/static
      - media_volume:/app/media
    depends_on:
      - db
    environment:
      - DEBUG=1
      - DB_NAME=${DB_NAME}
      - DB_USER=${DB_USER}
      - DB_PASSWORD=${DB_PASSWORD}
      - DB_HOST=db
      - DB_PORT=5432
    networks:
      - app_network
```

To verify let’s rebuild all containers:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose down
time="2024-10-13T02:27:23+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
 ✔Container djangonetworking-web-1      Removed                                                                    0.8s
 ✔Container djangonetworking-db-1       Removed                                                                    0.4s
 ✔Network djangonetworking_app_network  Removed                                                                    0.3s
```

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose up -d
time="2024-10-13T02:29:48+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
[+] Running 2/2
 ✔Container djangonetworking-db-1   Started                                                                          1.1s
 ✔Container djangonetworking-web-1  Started                                                                          1.7s
```

Volumes were created (last 3):

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker volume ls
DRIVER    VOLUME NAME
local     66a6f2f88a705acfa9a41d2d99b75e6736576744ac5b03f1e13ad97e75100607
local     795c5dd70d52c64d98f81e17b328da97e67cedfc4a1d5c0eff56ee6ca3e8b3e4
local     b6c943ac4482cc0862f8cd5cd11fbc6275664951194e3341f3b46fe52094ffc6
local     djangoapp_db_data
local     djangoapp_postgres_data
local     djangonetworking_db_data
local     djangonetworking_media_volume
local     djangonetworking_static_volume
```

Let’s check initial data in database:

```
local     djangonetworking_db_data
local     djangonetworking_media_volume
local     djangonetworking_static_volume

C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoNetworking>docker-compose exec db psql -U my_user -
d my_database_name
time="2024-10-13T02:32:37+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoNetworking\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential
 confusion"
psql (17.0 (Debian 17.0-1.pgdg120+1))
Type "help" for help.

my_database_name=# SELECT * FROM django_app_author;
 id |   name   |       email
----+----------+-------------------
  1 | Author 1 | author1@yandex.com
  2 | Author 2 | author2@yandex.com
(2 rows)

my_database_name=#
```

воскресенье, 13 октября 2024 г.
вс 02:32 (Местное время)

They remained there even after stopping the container.

- Networking: we set up custom network by using services’ names, so that they can communicate with each other without port exposing, which is more secure.

- Volumes: PostgreSQL data volumes remain all records of the application safe, and Django volumes save uploaded and static files outside of the container after rebuilds.

- Findings: Networking provided a communication between services improving isolation, security, and scalability. Volumes allowed us to keep all migrated data, files in the storage outside of the container without doing extra additional manipulation with data after restarting containers, so it represents efficient architecture of the application.

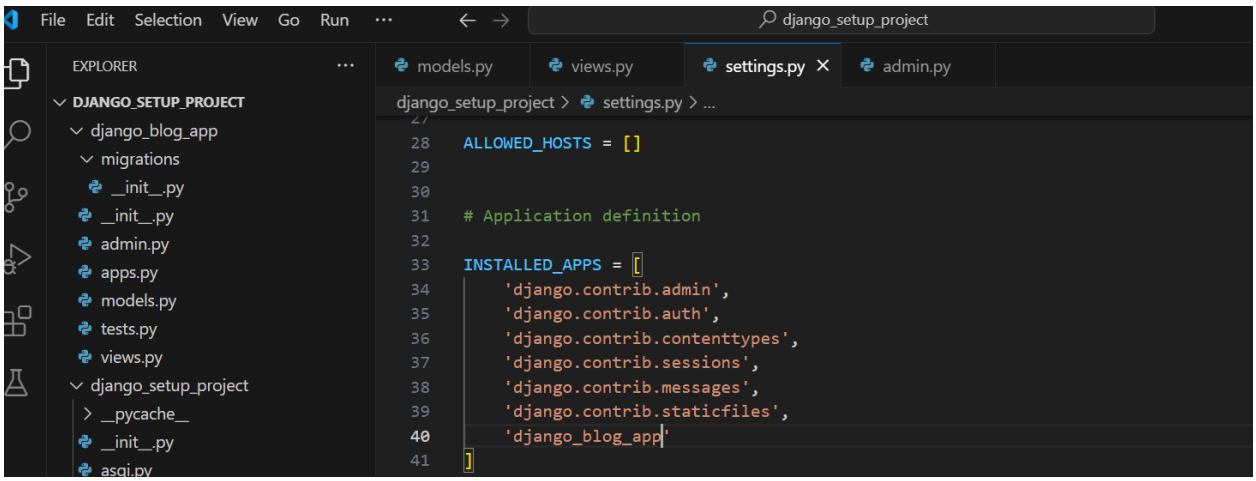
Django Application Setup

● Create a Django Project

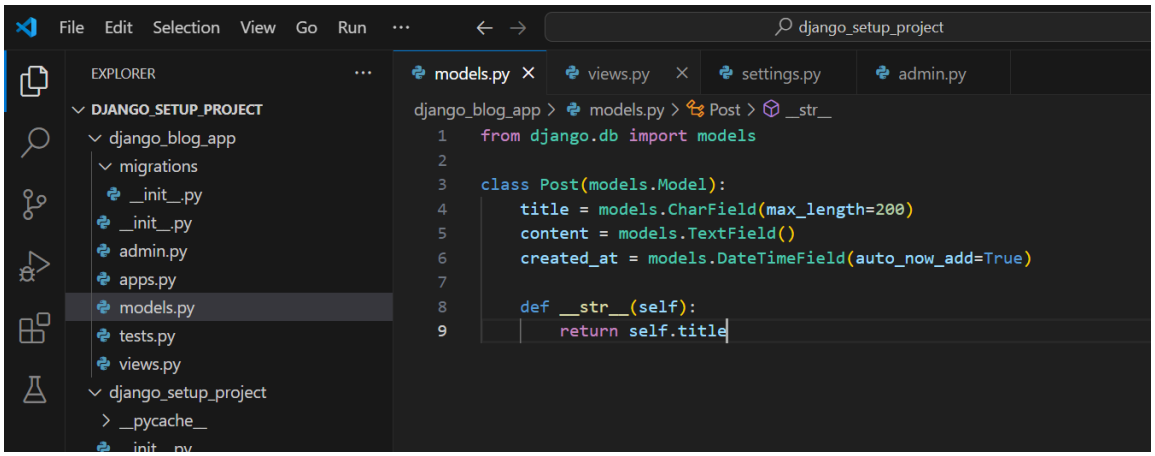
Created the project and the application:

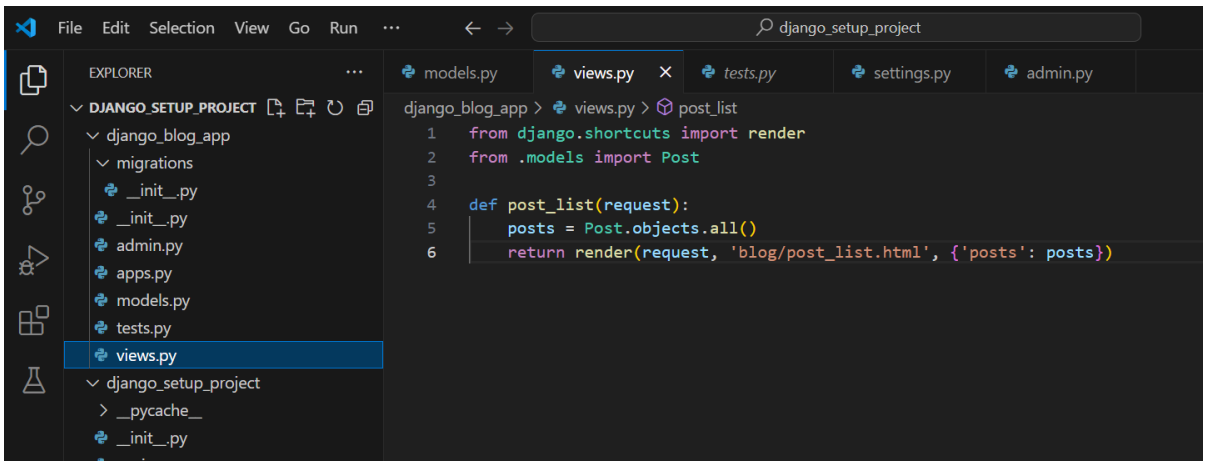
```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup>django-admin startproject django_setup_pro
ject

C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup>cd django_setup_project

C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>python manage.py star
tapp django_blog_app

C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>
```

Registered the application:



Our simple model Post and simple view to display a list of posts:

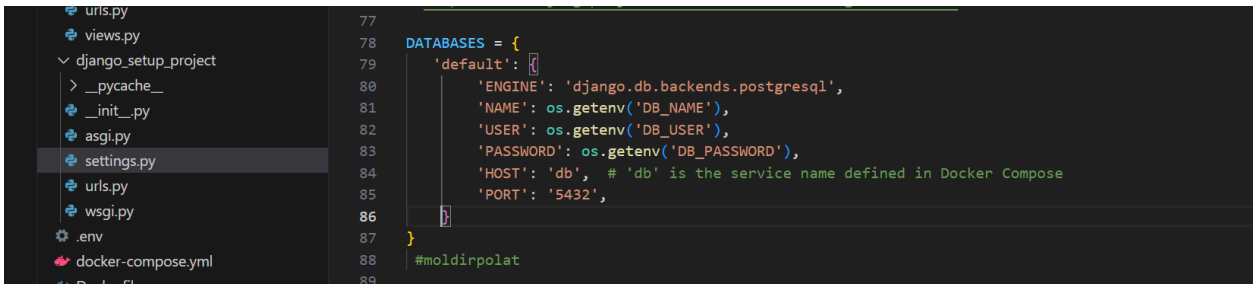




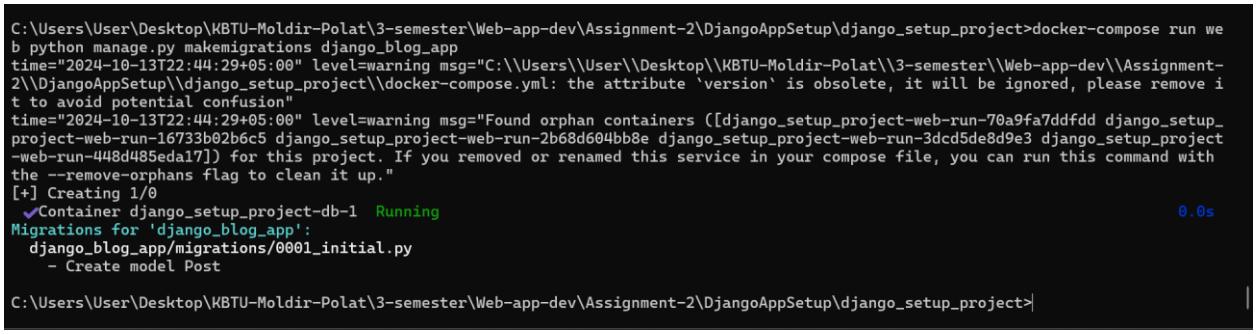
- **Configure the Database**

Updated settings.py to use the PostgreSQL database that we set up with Docker Compose, integrating it to our Django application.

Environment variables were used to keep sensitive information secure.



Run the migrations inside the Docker container to set up the database schema before starting the services:



```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>docker-compose run we
b python manage.py migrate
time="2024-10-13T22:46:35+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoAppSetup\\django_setup_project\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove i
t to avoid potential confusion"
time="2024-10-13T22:46:35+05:00" level=warning msg="Found orphan containers ([django_setup_project-web-run-543e5de8e4a5 django_setup_
project-web-run-70a9fa7ddfdd django_setup_project-web-run-16733b02b6c5 django_setup_project-web-run-2b68d604bb8e django_setup_project
-web-run-3dcd5de8d9e3 django_setup_project-web-run-448d485eda17]) for this project. If you removed or renamed this service in your co
mpose file, you can run this command with the --remove-orphans flag to clean it up."
[+] Creating 1/0
 ✔Container django_setup_project-db-1  Running                                                                          0.0s
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, django_blog_app, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying django_blog_app.0001_initial... OK
```

Started both services:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>docker-compose up --b
uild -d
time="2024-10-13T22:49:26+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-
2\\DjangoAppSetup\\django_setup_project\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove i
t to avoid potential confusion"
[+] Building 4.3s (12/12) FINISHED                                                                    docker:desktop-linux
 => [web internal] load build definition from Dockerfile                                                              0.0s
 => => transferring dockerfile: 234B                                                                                  0.0s
 => [web internal] load metadata for docker.io/library/python:3.9-slim                                                4.0s
 => [web internal] load .dockerignore                                                                                 0.0s
 => => transferring context: 2B                                                                                       0.0s
 => [web 1/6] FROM docker.io/library/python:3.9-slim@sha256:49f94609e5a997dc16086a66ac9664591854031d48e375945a9dbf4d1d53abbc         0.0s
 => [web internal] load build context                                                                                 0.0s
 => => transferring context: 13.83kB                                                                                  0.0s
 => CACHED [web 2/6] RUN apt-get update && apt-get install -y gcc libpq-dev                                           0.0s
 => CACHED [web 3/6] WORKDIR /app                                                                                     0.0s
 => CACHED [web 4/6] COPY requirements.txt /app/                                                                      0.0s
 => CACHED [web 5/6] RUN pip install --no-cache-dir -r requirements.txt                                               0.0s
 => [web 6/6] COPY . /app/                                                                                            0.0s
 => [web] exporting to image                                                                                          0.1s
 => => exporting layers                                                                                               0.0s
 => => writing image sha256:ff5070cb5a051f4cc4a0bc5a90a6b49a72668224868df28f6fe7ae7d40f26d40                          0.0s
 => => naming to docker.io/library/django_setup_project-web                                                           0.0s
 => [web] resolving provenance for metadata file                                                                      0.0s
time="2024-10-13T22:49:31+05:00" level=warning msg="Found orphan containers ([django_setup_project-web-run-01128dc77b09 django_setup_
project-web-run-543e5de8e4a5 django_setup_project-web-run-70a9fa7ddfdd django_setup_project-web-run-16733b02b6c5 django_setup_project
-web-run-2b68d604bb8e django_setup_project-web-run-3dcd5de8d9e3 django_setup_project-web-run-448d485eda17]) for this project. If you
removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up."
[+] Running 2/2
 ✔Container django_setup_project-db-1   Running                                                                       0.0s
 ✔Container django_setup_project-web-1  Started                                                                       0.4s
```

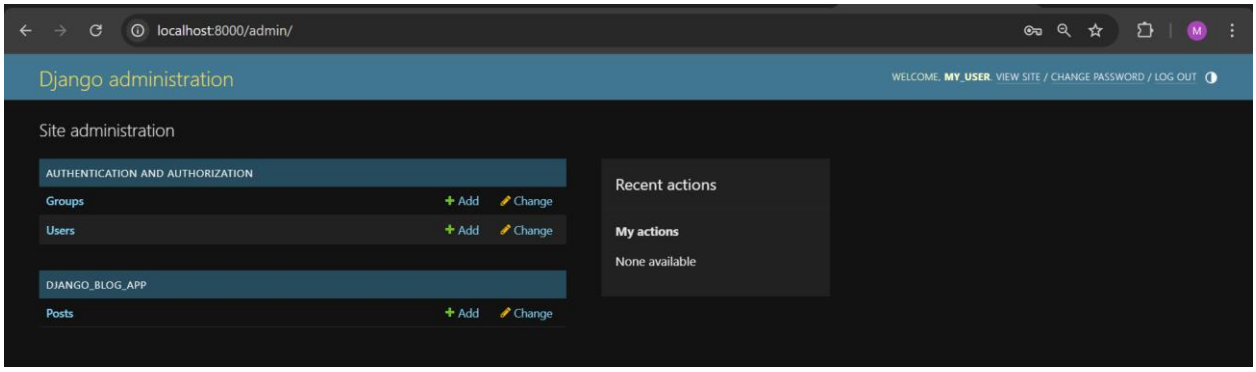
Our site:

Posts of Moldir Polat

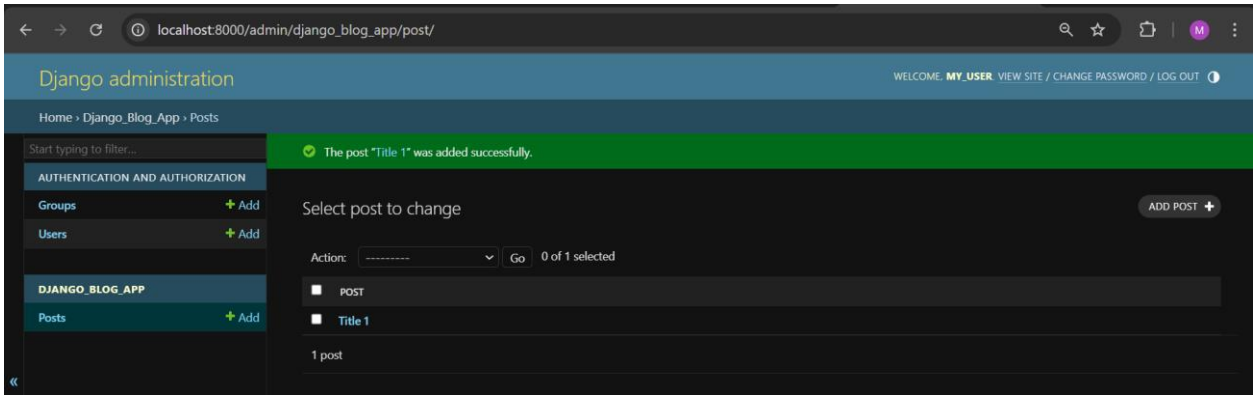
Created superuser to add some data:

```
C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>docker-compose run web python manage.py crea
tesuperuser
time="2024-10-13T23:01:13+05:00" level=warning msg="C:\\Users\\User\\Desktop\\KBTU-Moldir-Polat\\3-semester\\Web-app-dev\\Assignment-2\\DjangoAppSetup\\djan
go_setup_project\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
time="2024-10-13T23:01:13+05:00" level=warning msg="Found orphan containers ([django_setup_project-web-run-01128dc77b09 django_setup_project-web-run-543e5de
8e4a5 django_setup_project-web-run-70a9fa7ddfdd django_setup_project-web-run-16733b02b6c5 django_setup_project-web-run-2b68d604bb8e django_setup_project-web
-run-3dcd5de8d9e3 django_setup_project-web-run-448d485eda17]) for this project. If you removed or renamed this service in your compose file, you can run thi
s command with the --remove-orphans flag to clean it up."
[+] Creating 1/0
 ✔Container django_setup_project-db-1  Running                                                                                                    0.0s
Username (leave blank to use 'root'): my_user
Email address: moldir.polat@yandex.com
Password:
Password (again):
This password is too common.
Bypass password validation and create user anyway? [y/N]: N
Password:
Password (again):
Superuser created successfully.

C:\Users\User\Desktop\KBTU-Moldir-Polat\3-semester\Web-app-dev\Assignment-2\DjangoAppSetup\django_setup_project>
```

Admin page:



Added data:



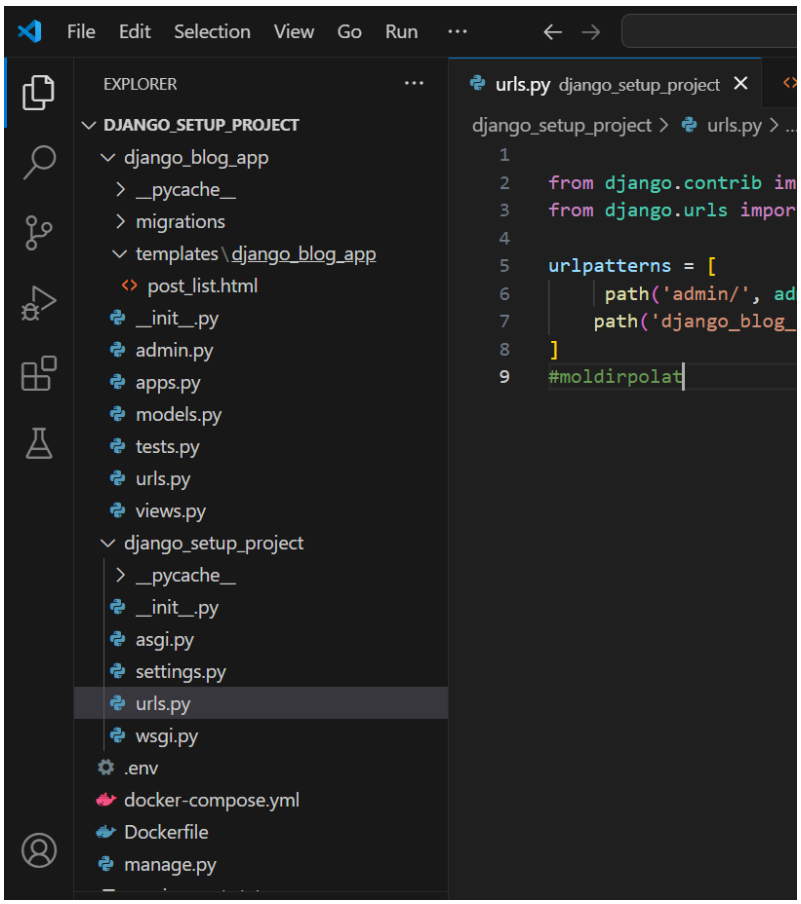
Posts of Moldir Polat

- Title 1 - Oct. 13, 2024, 6:06 p.m.
- Title 2 - Oct. 13, 2024, 6:06 p.m.

- **Document the Process**

All process, models, views, and mingrations were captured above.

Final project structure:



The Django application consists of a main project folder (**django_setup_project**), which contains the core settings and configuration files, and an app folder (**django_blog_app**) that handles specific features like models, views, and templates. Docker was used to create isolated containers for the web application and the PostgreSQL database. The **Dockerfile** defines the environment for the Django application, including dependencies, and sets up the working directory. **Docker Compose** (docker-compose.yml) manages the orchestration of multiple services. The **web** service uses the configuration provided in the **Dockerfile** to build the application, while the **db** service uses the official PostgreSQL image.

Our findings:

It is easier to work with Docker because it can handle dependencies and database configuration. The challenge was to ensure all services communicated correctly, but Docker Compose simplified linking the database to the application.

Conclusion

Docker Compose can define and manage multiple services, such as a Django web application and a PostgreSQL database, with ease.

Docker Networking makes managing services efficient, provides secure communication between them.

Docker volumes allow data persistence.

The most important takeaway from the assignment is ability to configure files correctly in order to containerize the Django application in Docker, along with databse settings, environment variables, volumes, migrations.