Object Storage Application with Keycloak and MinIO Integration

Калоян Анастасов 11А клас

**Описание на проекта**

Този проект представлява приложение за обектно съхранение, изградено с Python бекенд, интегрирано с Keycloak за удостоверяване и MinIO за съхранение. Приложението е контейнеризирано с Docker и използва Docker Compose за лесно разгръщане.

**Функционалности**

* **Keycloak интеграция**: Осигурява сигурно удостоверяване и контрол на достъпа.
* **MinIO интеграция**: Предоставя обектно съхранение, съвместимо с API на S3.
* **PostgreSQL база данни**: Използва се като база данни за Keycloak.
* **Flask бекенд**: Имплементира логиката на приложението и API (може да се модифицира за друг бекенд фреймуърк).
* **Контейнеризирана среда**: Улеснява настройката и разгръщането чрез Docker Compose.

**Структура на проекта**

├── docker-compose.yml

├── src/

│ ├── app.py

│ ├── requirements.txt

**Как да стартирате проекта**

За да стартирате проекта, следвайте следните стъпки:

1. **Клониране на репозиторията**  
   Уверете се, че сте клонирали репозиторията на вашата локална машина:

git clone https://github.com/Flychuban/VOT\_11grade/tree/main/ObjectStorageApp

1. **Създаване на .env файл**  
   Създайте файл с име .env в основната директория на проекта. Копирайте и поставете следното съдържание в него, като при необходимост актуализирате стойностите:

POSTGRES\_DB=keycloak

POSTGRES\_USER=keycloak

POSTGRES\_PASSWORD=keycloak

KEYCLOAK\_ADMIN=admin

KEYCLOAK\_ADMIN\_PASSWORD=admin

MINIO\_ROOT\_USER=minioadmin

MINIO\_ROOT\_PASSWORD=minioadmin

S3\_ENDPOINT=http://minio:9000

S3\_ACCESS\_KEY=minioadmin

S3\_SECRET\_KEY=minioadmin

KEYCLOAK\_URL=http://keycloak:8080

KEYCLOAK\_REALM=myrealm

KEYCLOAK\_CLIENT\_ID=myclient

1. **Стартиране на Docker Compose**  
   Уверете се, че Docker и Docker Compose са инсталирани на вашата система. За да стартирате проекта, изпълнете следната команда в основната директория на проекта:

docker-compose up --build

1. **Достъп до услугите**  
   След успешното стартиране, ще имате достъп до следните услуги:
   * **Keycloak:**  
     http://localhost:8080  
     Влезте с потребителско име ${KEYCLOAK\_ADMIN} и парола ${KEYCLOAK\_ADMIN\_PASSWORD}.
   * **MinIO Web Console:**  
     http://localhost:9001  
     Влезте с потребителско име ${MINIO\_ROOT\_USER} и парола ${MINIO\_ROOT\_PASSWORD}.
   * **Приложението:**  
     <http://localhost:5100>
2. **Проверка на състоянието на контейнерите**  
   За да проверите дали всички контейнери работят коректно, изпълнете:

docker ps

Уверете се, че всички необходими услуги (Keycloak, PostgreSQL, MinIO и приложението) са в статус Up.

1. **Спиране на проекта**  
   За да спрете всички контейнери, натиснете Ctrl+C в терминала, където работи docker-compose up. След това изпълнете:

docker-compose down

#### **Качване на файл** Качете файл в MinIO: curl -X POST -F "file=@<path-to-your-file>" <http://localhost:5100/upload> Очакван резултат: { "message": "File uploaded", "file\_id": "example.txt" }

Изтегляне на файл  
curl http://localhost:5100/download/<file\_id> -o <output-file-name>

Актуализиране на файл  
curl -X PUT -F "file=@<path-to-your-updated-file>" [http://localhost:5100/update/<file\_id](http://localhost:5100/update/%3cfile_id)>

Изтриване на файл  
curl -X DELETE http://localhost:5100/delete/<file\_id>