САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа № 4

“Docker, docker compose, docker swarm”

Выполнил:

Ле Хоанг Чыонг

Чан Дык Минь

Группа:

K33392

Проверил:

Добряков Д. И.

Санкт-Петербург

2024 г.

Задача

Необходимо упаковать ваше приложение в docker-контейнеры и обеспечить сетевое взаимодействие между различными частями вашего приложения. Делать это можно как с помощью docker-compose так и с помощью docker swarm. При разумном использовании swirl вы получите дополнительные баллы.

Ход работы

1. **Использование Docker в архитектуре микросервисов:**

Docker обеспечивает легкие, переносимые контейнеры, которые инкапсулируют программное обеспечение и его зависимости, позволяя приложениям работать согласованно в различных средах. В архитектуре микросервисов Docker предлагает несколько преимуществ:

2. **Проект на основе Docker и микросервисов:**

Предоставленный проект Docker является примером реализации архитектуры микросервисов с использованием контейнеров Docker. Проект включает в себя несколько служб, каждая из которых инкапсулирована в своем контейнере Docker, включая:

- Authentication Service
- Activity Service
- Location Service
- Offer Service
- Trip Service
- Review Service
- Gateway Service
- Database Service (PostgreSQL)

3. **Основные компоненты проекта Docker:**

Dockerfiles: Каждая служба определена в своем Dockerfile, где указаны среда, зависимости и конфигурация времени выполнения.

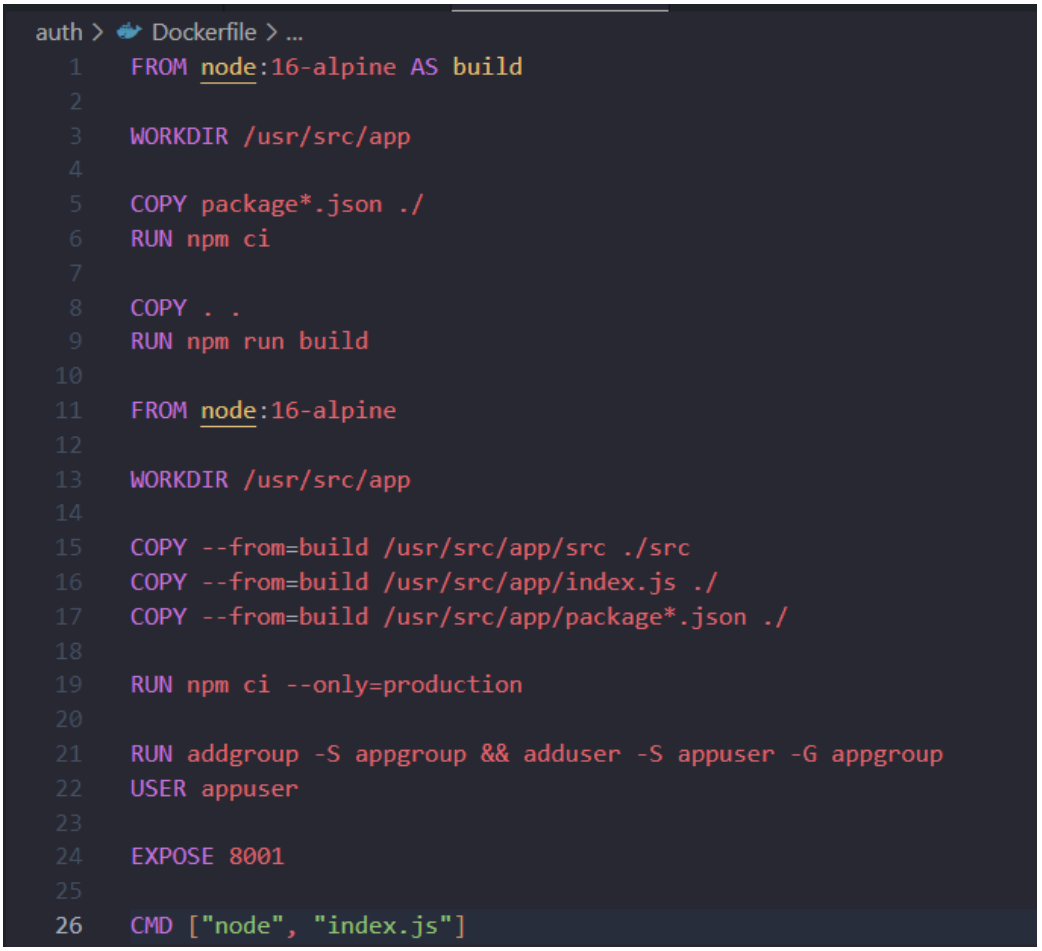

Рисунок 1: Dockerfile для auth-service.

Этот файл Docker делит процесс сборки на два этапа:

На первом этапе он копирует файлы package.json, устанавливает пакеты npm и выполняет процесс сборки приложения.

Затем он создает новый образ из старого, копирует только необходимые файлы и папки с предыдущего этапа сборки и устанавливает пакеты npm, необходимые для производственной среды. Наконец, он создает системного пользователя и объявляет сетевой порт, который должен быть открыт для доступа приложения, и запускает приложение Node.js.

docker-compose.yml: Docker-compose.yml в этом проекте описывает работу таких служб, как аутентификация, активность, местоположение, предложение, поездка, обзор, шлюз и база данных. Эти службы связаны между собой внутренней сетью и доступны извне через сопоставленные порты.

Функционирование некоторых служб, таких как аутентификация и активность, зависит от службы базы данных (db). Объявлено, что службы перезапускаются, когда они перестают работать.

Рисунок 2. Описание сервисов в docker-compose.yml.

Кроме того, этот файл определяет экземпляр службы базы данных PostgreSQL, использует образ postgres:latest и настраивает параметры среды и тома для этой службы.

```yaml
  db:
    image: postgres:latest
    ports:
      - "5432:5432"
    environment:
      POSTGRES_USER: ${POSTGRES_USER}
      POSTGRES_PASSWORD: ${POSTGRES_PASSWORD}
      POSTGRES_DB: ${POSTGRES_DB}
    volumes:
      - postgres_data:/var/lib/postgresql/data

volumes:
  postgres_data:
```

Рисунок 3. Определение экземпляра службы базы данных PostgreSQL.

Зависимости служб: Проект демонстрирует взаимодействие и управление зависимостями между службами, где службы полагаются друг на друга для выполнения определенных функциональностей.

4. **Проверьте развернутые сетевые взаимодействия.**

Выполним команду docker compose up для развертывания системы сервисов

```
PS C:\Users\minhh\OneDrive\Desktop\lab3_be\lab3_be> docker compose up
time="2024-05-15T12:45:34+03:00" level=warning msg="C:\\Users\\minhh\\OneDrive\\Desktop\\
[+] Running 8/0
 ✓ Container lab3_be-db-1        Created
 ✓ Container lab3_be-auth-1      Running
 ✓ Container lab3_be-activity-1  Running
 ✓ Container lab3_be-location-1  Running
 ✓ Container lab3_be-review-1    Running
 ✓ Container lab3_be-trip-1      Running
 ✓ Container lab3_be-offer-1     Running
 ✓ Container lab3_be-gateway-1   Running
Attaching to activity-1, auth-1, db-1, gateway-1, location-1, offer-1, review-1, trip-1
```

Рисунок 4. Службы развернуты успешно.

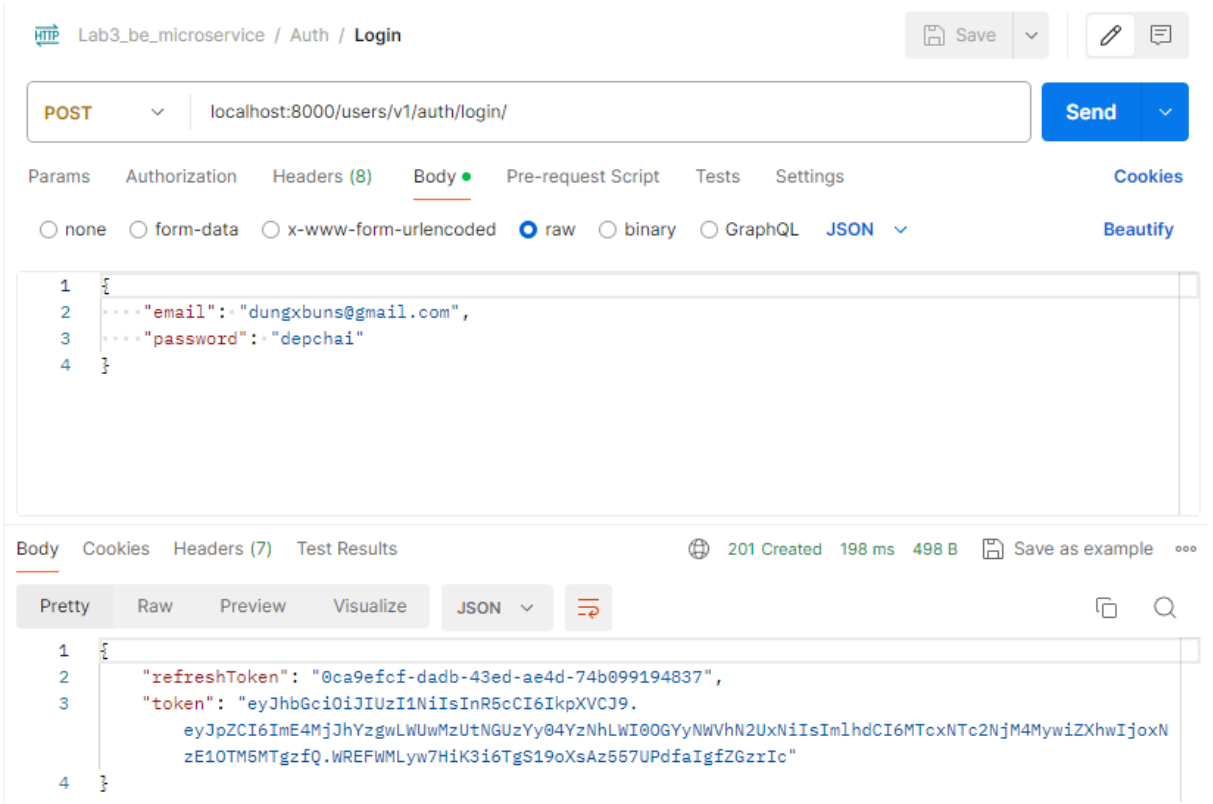

Рисунок 5. Auth-service работает нормально.

Вывод

В ходе лабораторной работы были изучены основы контейнеризации и сетевого взаимодействия контейнеров с использованием docker и docker compose. В проекте используется шаблон микросервисной архитектуры из предыдущей лабораторной работы, который был успешно докеризован.