САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа №4

Выполнил:

Гнеушев Владислав

К3339

Проверил:
Добряков Д. И.

Санкт-Петербург

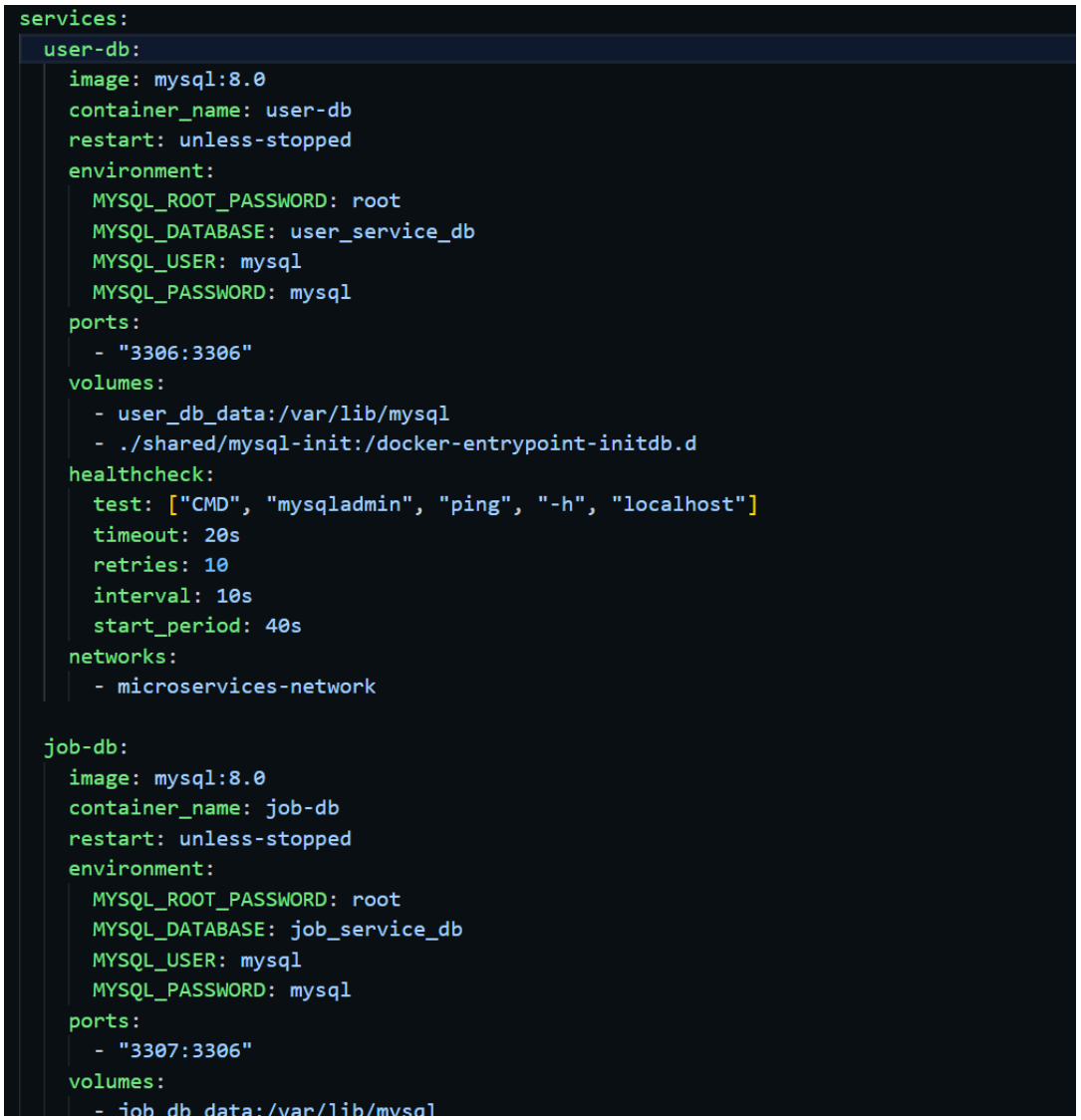
2025 г.

Задача

- реализовать Dockerfile для каждого сервиса;
- написать общий docker-compose.yml;
- настроить сетевое взаимодействие между сервисами.

Ход работы

Был написан общий docker-compose для всех сервисов. Он включает 3 микросервиса (пользователей, вакансий и откликов), а также базы данных для каждого из них и реверс-прокси в виде nginx.



```yaml
services:
  user-db:
    image: mysql:8.0
    container_name: user-db
    restart: unless-stopped
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: user_service_db
      MYSQL_USER: mysql
      MYSQL_PASSWORD: mysql
    ports:
      - "3306:3306"
    volumes:
      - user_db_data:/var/lib/mysql
      - ./shared/mysql-init:/docker-entrypoint-initdb.d
    healthcheck:
      test: ["CMD", "mysqladmin", "ping", "-h", "localhost"]
      timeout: 20s
      retries: 10
      interval: 10s
      start_period: 40s
    networks:
      - microservices-network

  job-db:
    image: mysql:8.0
    container_name: job-db
    restart: unless-stopped
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: job_service_db
      MYSQL_USER: mysql
      MYSQL_PASSWORD: mysql
    ports:
      - "3307:3306"
    volumes:
      - job_db_data:/var/lib/mysql
```

Рисунок 1 — Базы данных микросервисов

```yaml
job-service:
  build:
    context: ./microservices/job-service
    dockerfile: Dockerfile
  container_name: job-service
  restart: unless-stopped
  ports:
    - "3002:3002"
  env_file:
    - ./microservices/job-service/.env
  depends_on:
    job-db:
      condition: service_healthy
  volumes:
    - ./microservices/job-service:/app
    - /app/node_modules
    - /app/dist
  healthcheck:
    test: ["CMD", "wget", "--no-verbose", "--tries=1", "--spider", "http://localhost:3002/health"]
    interval: 30s
    timeout: 10s
    retries: 5
    start_period: 40s
  networks:
    - microservices-network

application-service:
  build:
    context: ./microservices/application-service
    dockerfile: Dockerfile
  container_name: application-service
  restart: unless-stopped
  ports:
```

Рисунок 2 — Микросервисы

```yaml
  nginx:
    image: nginx:alpine
    container_name: api-gateway
    restart: unless-stopped
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - ./shared/nginx/nginx.conf:/etc/nginx/nginx.conf:ro
      - ./shared/nginx/conf.d:/etc/nginx/conf.d:ro
    depends_on:
      - user-service
      - job-service
      - application-service
    networks:
      - microservices-network

volumes:
  user_db_data:
    driver: local
  job_db_data:
    driver: local
  application_db_data:
    driver: local
```

Рисунок 3 — Nginx

Далее были написаны Dockerfile для каждого сервиса.

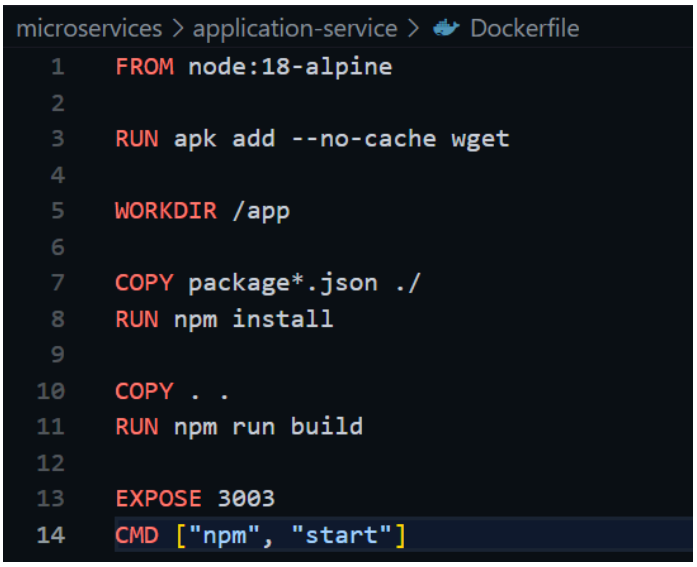


Рисунок 4 — Пример докерфайла микросервисов

Вывод

В ходе данной лабораторной работы система была упакована в Docker, с использованием Dockerfile-ов и файла docker-compose.yml. В качестве реверс-прокси был добавлен Nginx, который перенаправляет запросы в нужный микросервис.