

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа №4

Выполнил:

Русинов Василий

Группа К3340

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

Для каждого микросервиса был написан Dockerfile. Код на рисунке 1.

```
FROM node:18-alpine

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

RUN npm run build

CMD ["npx", "ts-node", "src/index.ts"]

EXPOSE 3000
```

Рисунок 1 – Dockerfile

Далее был написан docker-compose: подключены сервисы (пример на рисунке 2) и создана внутренняя сеть Docker (пример на рисунке 3).

```
services:
  gateway:
    build:
      context: ./gateway
    ports:
      - "3000:3000"
    depends_on:
      - auth-service
      - job-service
      - application-service
    networks:
      - backend

  auth-service:
    build:
      context: ./auth-service
    ports:
      - "3001:3001"
    environment:
      - DATABASE_URL=postgres://postgres:123@auth-db:5432/search_job
    depends_on:
      - auth-db
    networks:
      - backend

  auth-db:
    image: postgres:15
    restart: always
    environment:
      POSTGRES_USER: postgres
      POSTGRES_PASSWORD: 123
      POSTGRES_DB: search_job
    ports:
      - "5433:5432"
    networks:
      - backend
```

Рисунок 2 – Подключение сервисов в docker-compose

```

networks:
  backend:
    driver: bridge

```

Рисунок 3 – Создание внутренней сети Docker

Также были доработаны ссылки, чтобы контейнеризированный gateway их корректно обрабатывал. Пример на рисунке 4.

```

app.use("/auth", createProxyMiddleware({ target: "http://auth-service:3001", changeOrigin: true }));
app.use("/users", createProxyMiddleware({ target: "http://auth-service:3001", changeOrigin: true }));

app.use("/jobs", createProxyMiddleware({ target: "http://jobs-service:3002", changeOrigin: true }));
app.use("/employers", createProxyMiddleware({ target: "http://jobs-service:3002", changeOrigin: true }));
app.use("/industries", createProxyMiddleware({ target: "http://jobs-service:3002", changeOrigin: true }));

app.use("/applications", createProxyMiddleware({ target: "http://applications-service:3003", changeOrigin: true }));
app.use("/resumes", createProxyMiddleware({ target: "http://applications-service:3003", changeOrigin: true }));

```

Рисунок 4 – Ссылки на сервисы

| <input type="checkbox"/> | Name | Container ID | Image | Port(s) | CPU (%) | Last st | Actions |
|--------------------------|--|--------------|------------|-----------|---------|---------|---------|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> search_job | - | - | - | 64% | 6 secon | |
| <input type="checkbox"/> | application-dl | 4e34cfc1675 | postgres:1 | 5435:5432 | 0% | 3 minut | |
| <input type="checkbox"/> | job-db-1 | bcf2d142e6ed | postgres:1 | 5434:5432 | 0% | 3 minut | |
| <input type="checkbox"/> | auth-db-1 | ffc1f12f64a0 | postgres:1 | 5433:5432 | 0% | 3 minut | |
| <input type="checkbox"/> | auth-service- | 3f5822fbe805 | search_job | 3001:3001 | 8.37% | 10 secc | |
| <input type="checkbox"/> | application-si | 86aa267ef3d1 | search_job | 3003:3003 | 6.04% | 10 secc | |
| <input type="checkbox"/> | job-service-1 | 89ea0218a9fe | search_job | 3002:3002 | 5.9% | 10 secc | |
| <input type="checkbox"/> | gateway-1 | d7230d422fe2 | search_job | 3000:3000 | 43.69% | 6 secon | |

Рисунок 5 – Контейнеры в Docker Desktop

Вывод

В ходе данной лабораторной работы приложение было упаковано в Docker.