# САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

**Дисциплина:** Бэк-энд разработка Отчет

Реализация Контейнеризация написанного приложения средствами docker

Выполнил: Петухов Семён

Группа 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

На основе реализованного ранее микросервисного приложения было реализована контейнеризация в docker.  
Для каждого сервиса был написан dockerfile, с описанием контейнера

# Dockerfile  
FROM node:20-alpine  
  
WORKDIR /app  
  
COPY package\*.json ./  
RUN npm install  
  
COPY . .  
RUN npm run build  
  
EXPOSE 3003  
  
CMD ["npm", "run", "start"]

Далее был разработан docker-compose, который объединяет все системы в одну сеть

version: "3.9"  
services:  
 postgres:  
 image: postgres:15  
 environment:  
 POSTGRES\_USER: postgres  
 POSTGRES\_PASSWORD: 1234  
 POSTGRES\_DB: resume\_finder  
 volumes:  
 - pg\_data:/var/lib/postgresql/data  
 ports:  
 - "5432:5432"  
 networks:  
 - app-network  
  
 user-service:  
 build: ./user-service  
 ports:  
 - "3001:3001"  
 depends\_on:  
 - postgres  
 environment:  
 DB\_HOST: postgres  
 DB\_PORT: 5432  
 DB\_USERNAME: postgres  
 DB\_PASSWORD: 1234  
 DB\_DATABASE: resume\_finder  
 networks:  
 - app-network  
  
 resume-service:  
 build: ./resume-service  
 ports:  
 - "3002:3002"  
 depends\_on:  
 - postgres  
 environment:  
 DB\_HOST: postgres  
 DB\_PORT: 5432  
 DB\_USERNAME: postgres  
 DB\_PASSWORD: 1234  
 DB\_DATABASE: resume\_finder  
 networks:  
 - app-network  
  
 vacancy-service:  
 build: ./vacancy-service  
 ports:  
 - "3003:3003"  
 depends\_on:  
 - postgres  
 environment:  
 DB\_HOST: postgres  
 DB\_PORT: 5432  
 DB\_USERNAME: postgres  
 DB\_PASSWORD: 1234  
 DB\_DATABASE: resume\_finder  
 networks:  
 - app-network  
  
 skill-service:  
 build: ./skill-service  
 ports:  
 - "3005:3005"  
 depends\_on:  
 - postgres  
 environment:  
 DB\_HOST: postgres  
 DB\_PORT: 5432  
 DB\_USERNAME: postgres  
 DB\_PASSWORD: 1234  
 DB\_DATABASE: resume\_finder  
 networks:  
 - app-network  
  
 gateway:  
 build: ./gateway  
 ports:  
 - "3000:3000"  
 depends\_on:  
 - user-service  
 - resume-service  
 - vacancy-service  
 - skill-service  
 networks:  
 - app-network  
  
volumes:  
 pg\_data:  
  
networks:  
 app-network:  
 driver: bridge

Были доработаны ссылки, чтобы контейнеризированный gateway их корректно обрабатывал

app.use("/user-service", createProxyMiddleware({ target: "http://user-service:3001", changeOrigin: true }));  
app.use("/resume-service", createProxyMiddleware({ target: "http://resume-service:3002", changeOrigin: true }));  
app.use("/vacancy-service", createProxyMiddleware({ target: "http://vacancy-service:3003", changeOrigin: true }));  
app.use("/skill-service", createProxyMiddleware({ target: "http://skill-service:3005", changeOrigin: true }));

# Вывод

По результатам работы система была мигрирована в контейнеры Docker