

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа 4

Выполнила:

Афанасьева Ирина Максимовна

Группа:

К33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2023 г.

Задание

Необходимо упаковать ваше приложение в docker-контейнеры и обеспечить сетевое взаимодействие между различными частями вашего приложения.

Ход работы

Пропишем докерфайлы для микросервисов:

```
FROM node:16
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 5001
CMD [ "npm", "start" ]
```

















```
FROM node:16
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 5000
CMD [ "npm", "start" ]
```

Обеспечим сетевое взаимодействие между частями приложения с помощью docker compose, в docker-compose.yml:









```
version: '3.9'
services:
  auth_service:
    container_name: auth_service
    build:
      context: ./auth_service
    ports:
      - "5001:5001"
    command: npm run start

  main_service:
    container_name: main_service
    build:
      context: ./main_service
    ports:
      - "5000:5000"
    command: npm run start
    depends_on:
      - auth_service
```

Контейнеры:

<input type="checkbox"/>	Name	Image	Status	Port(s)	Last started	Actions
<input type="checkbox"/>	 lw4		Running (2/2)		41 seconds ago	  
<input type="checkbox"/>	 auth_service b4e8a5ce0693 	lw4-auth_service	Running	5001:5001 	42 seconds ago	  
<input type="checkbox"/>	 main_service a407c9c8ba05 	lw4-main_service	Running	5000:5000 	41 seconds ago	  

Образы:

<input type="checkbox"/>	Name	Tag	Status	Created	Size	Actions
<input type="checkbox"/>	lw4-main_service 51217e305880 	latest	In use	4 minutes ago	1.03 GB	  
<input type="checkbox"/>	lw4-auth_service 3aa43e0bb64b 	latest	In use	4 minutes ago	1.03 GB	  

Запуск:

```
[+] Running 3/3
✓ Network lw4_default      Created
✓ Container auth_service   Created
✓ Container main_service   Created
auth_service | [INFO] 23:07:25 ts-node-dev ver. 2.0.0 (using ts-node ver. 10.8.1, typescript ver. 4.7.3)
main_service |
main_service | > lw2@1.0.0 start
main_service | > ts-node-dev src/index.ts
main_service |
main_service | [INFO] 23:07:25 ts-node-dev ver. 2.0.0 (using ts-node ver. 10.8.1, typescript ver. 4.7.3)
auth_service | Added models
auth_service | Running server on port 5001
auth_service | Executing (default): SELECT name FROM sqlite_master WHERE type='table' AND name='Users';
auth_service | Executing (default): SELECT 1+1 AS result
auth_service | Executing (default): PRAGMA INDEX_LIST('Users')
auth_service | Connection has been established successfully.
auth_service | Executing (default): PRAGMA INDEX_INFO('sqlite_autoindex_Users_1')
auth_service | Executing (default): SELECT name FROM sqlite_master WHERE type='table' AND name='RefreshTokens';
auth_service | Executing (default): PRAGMA INDEX_LIST('RefreshTokens')
auth_service | Executing (default): PRAGMA INDEX_INFO('sqlite_autoindex_RefreshTokens_1')
auth_service | Synced models
main_service | Added models
main_service | Running server on port 5000
main_service | Executing (default): SELECT name FROM sqlite_master WHERE type='table' AND name='Wallets';
main_service | Executing (default): SELECT 1+1 AS result
main_service | Executing (default): PRAGMA INDEX_LIST('Wallets')
main_service | Connection has been established successfully.
main_service | Executing (default): SELECT name FROM sqlite_master WHERE type='table' AND name='Coins';
main_service | Executing (default): PRAGMA INDEX_LIST('Coins')
main_service | Executing (default): PRAGMA INDEX_INFO('sqlite_autoindex_Coins_1')
main_service | Executing (default): SELECT name FROM sqlite_master WHERE type='table' AND name='CoinWallets';
main_service | Executing (default): PRAGMA INDEX_LIST('CoinWallets')
main_service | Executing (default): PRAGMA INDEX_INFO('sqlite_autoindex_CoinWallets_1')
main_service | Synced models
```

Вывод

В ходе работы я упаковала микросервисное приложение в контейнеры Docker и настроила их взаимодействие при помощи docker-compose.