

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа 4

Выполнил:
Комиссаров Александр
К33402

Проверил:
Добряков Д. И.

Санкт-Петербург

2022 г.

Задача

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

Ход работы

Для каждого микросервиса создадим Dockerfile. Порты оставим прежними, 8000 для основного приложения и 8100 для микросервиса аутентификации.

Пример Dockerfile для микросервиса аутентификации:

```
FROM node:16

WORKDIR /app/auth

COPY package.json .

RUN npm install

COPY . .

EXPOSE 8100

CMD [ "npm", "start" ]
```

В корень проекта добавим файл docker-compose

```
version: '3.8'

services:
  auth:
    build:
      context: ./auth
    ports:
      - '8100:8100'
    restart: always
  main:
    build:
      context: ./main
    ports:
      - '8000:8000'
    restart: always
```

Теперь контроллер аутентификации обращается уже не по адресу localhost:8000, а по адресу контейнера main, то есть main:8000.

```
import axios from "axios"

export default class AuthController {

  post = async (request: any, response: any) => {
    const { email, password } = request.body
    const secretOrKey = 'test123'
    if (email && password) {
      let user = await axios.get("http://main:8000/v1/user", {
        params: {
          email: email
        }
      })
      if (!user) {
        response.status(401).json({ msg: 'No such user found', user })
      }
      if (user!.data.password === password) {
        let payload = { id: user!.data.id }
        const jwt = require('jsonwebtoken')
        let token = jwt.sign(payload, secretOrKey)
        response.json({ msg: 'ok', token: token })
      } else {
        response.status(401).json({ msg: 'Password is incorrect' })
      }
    }
  }
}
```

Вывод

В результате выполненной работы наше приложение было упаковано в docker-контейнеры, а также было обеспечено сетевое взаимодействие между ними.