

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

ОТЧЕТ

по Лабораторной работе № 4

по теме: «*Docker*»

по дисциплине: Бек-энд разработка

Специальность:

09.03.03 Мобильные и сетевые технологии

Проверил:

Добряков Д. И. _____

Дата: «__» _____ 202__ г.

Оценка _____

Выполнил:

студент группы К33401

Чернов Е. К.

Санкт-Петербург

2023

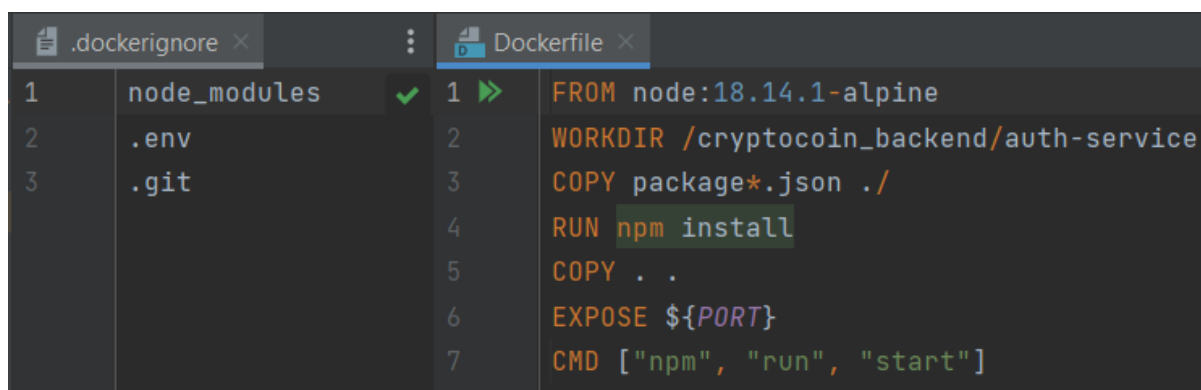
ЦЕЛЬ РАБОТЫ

Получить практические навыки по работе с *Docker* и *Docker Compose*.

ВЫПОЛНЕНИЕ

1 Dockerfile

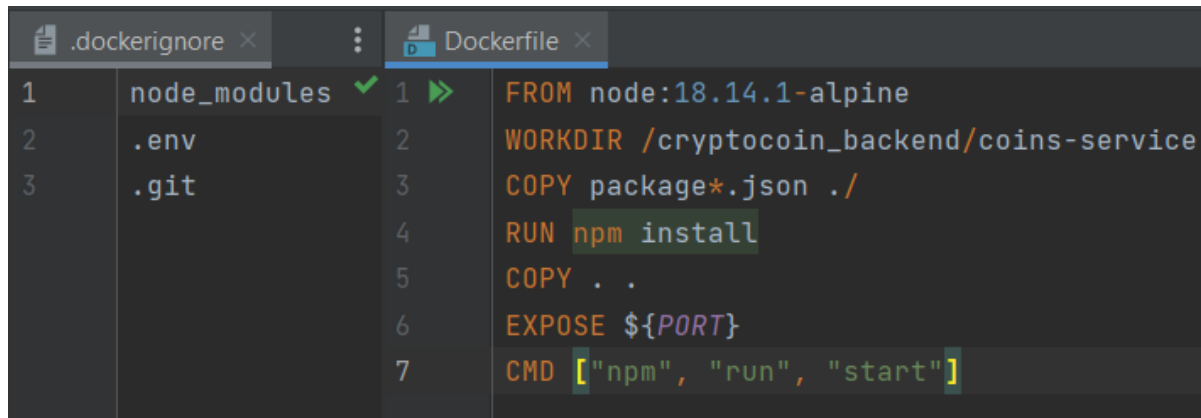
Каждому сервису добавляем в папку файлы «*.dockerignore*» и «*Dockerfile*».



The screenshot shows a code editor with two tabs: `.dockerignore` and `Dockerfile`. The `.dockerignore` tab is active, showing a list of files to be ignored: `node_modules`, `.env`, and `.git`. The `Dockerfile` tab is also visible, showing the following content:

```
1 FROM node:18.14.1-alpine
2 WORKDIR /cryptocoin_backend/auth-service
3 COPY package*.json ./
4 RUN npm install
5 COPY . .
6 EXPOSE ${PORT}
7 CMD ["npm", "run", "start"]
```

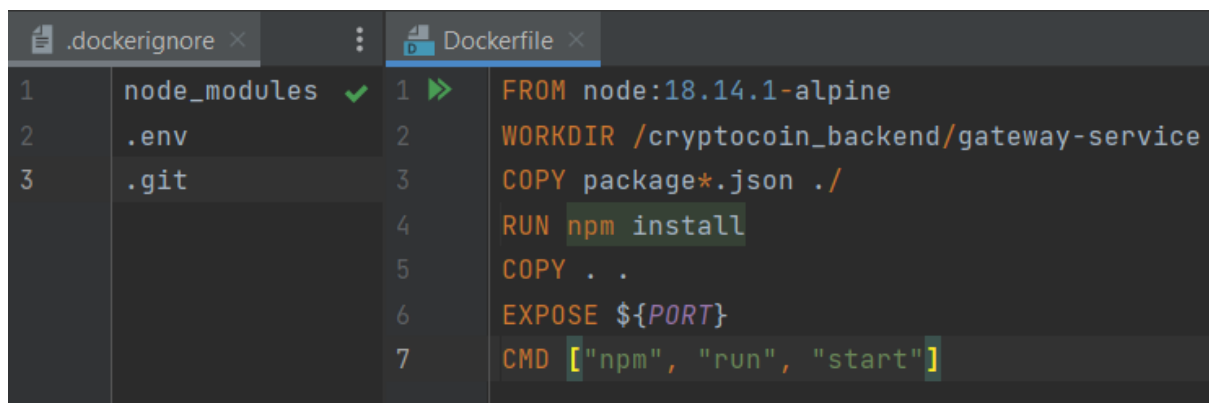
Рисунок 1 - auth-service



The screenshot shows a code editor with two tabs: `.dockerignore` and `Dockerfile`. The `.dockerignore` tab is active, showing a list of files to be ignored: `node_modules`, `.env`, and `.git`. The `Dockerfile` tab is also visible, showing the following content:

```
1 FROM node:18.14.1-alpine
2 WORKDIR /cryptocoin_backend/coins-service
3 COPY package*.json ./
4 RUN npm install
5 COPY . .
6 EXPOSE ${PORT}
7 CMD ["npm", "run", "start"]
```

Рисунок 2 - coins-service



```
.dockerignore x
1 node_modules ✓
2 .env
3 .git

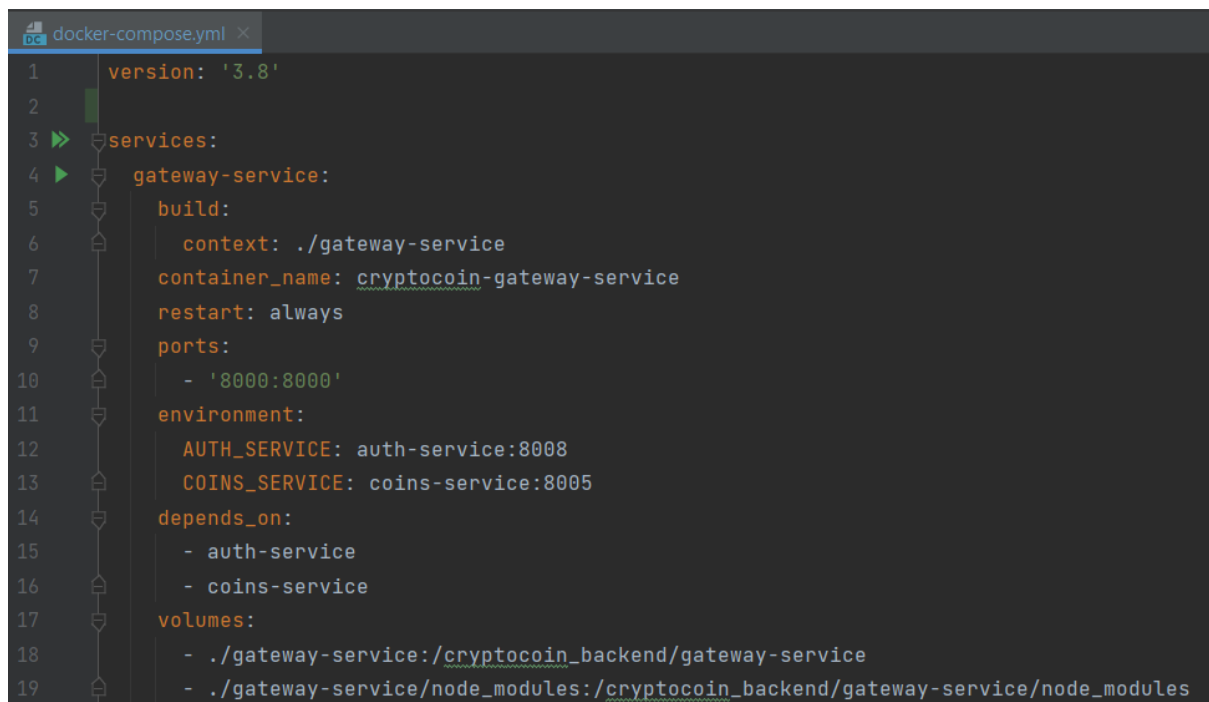
Dockerfile x
1 FROM node:18.14.1-alpine
2 WORKDIR /cryptocoin_backend/gateway-service
3 COPY package*.json ./
4 RUN npm install
5 COPY . .
6 EXPOSE ${PORT}
7 CMD ["npm", "run", "start"]
```

Рисунок 3 - gateway-service

2 Docker Compose

В корневой папке добавляем файл «*docker-compose.yml*» в котором определяем сервисы и их взаимосвязь.

Настройки для *gateway-service* (рисунок 4):



```
docker-compose.yml x
1 version: '3.8'
2
3 services:
4   gateway-service:
5     build:
6       context: ./gateway-service
7     container_name: cryptocoin-gateway-service
8     restart: always
9     ports:
10      - '8000:8000'
11     environment:
12       AUTH_SERVICE: auth-service:8008
13       COINS_SERVICE: coins-service:8005
14     depends_on:
15       - auth-service
16       - coins-service
17     volumes:
18       - ./gateway-service:/cryptocoin_backend/gateway-service
19       - ./gateway-service/node_modules:/cryptocoin_backend/gateway-service/node_modules
```

Рисунок 4 - gateway-service

Настройки для *auth-service* (рисунок 5):

```
auth-service:
  build:
    context: ./auth-service
  container_name: cryptocoin-auth-service
  restart: always
  expose:
    - 8008
  environment:
    DB_HOST: auth-db
    DB_PORT: 5432
  depends_on:
    - auth-db
  volumes:
    - ./auth-service:/cryptocoin_backend/auth-service
    - /cryptocoin_backend/auth-service/node_modules
```

Рисунок 5 - auth-service

Настройки для *coins-service* (рисунок 6):

```
coins-service:
  build:
    context: ./coins-service
  container_name: cryptocoin-coins-service
  restart: always
  expose:
    - 8005
  environment:
    DB_HOST: coins-db
    DB_PORT: 5432
  depends_on:
    - coins-db
  volumes:
    - ./coins-service:/cryptocoin_backend/coins-service
    - /cryptocoin_backend/coins-service/node_modules
```

Рисунок 6 - coins-service

Настройки для *auth-db* (рисунок 7):

```
auth-db:
  image: postgres
  container_name: cryptocoin-auth-db
  restart: always
  env_file:
    - .env
  expose:
    - 5432
  environment:
    POSTGRES_USER: ${DB_USERNAME}
    POSTGRES_PASSWORD: ${DB_PASSWORD}
    POSTGRES_DB: ${DB_AUTH}
  volumes:
    - ./cryptocoin_backend/auth_data:/var/lib/postgresql/data
```

Рисунок 7 - auth-db

Настройки для *coins-db* (рисунок 8):

```
coins-db:
  image: postgres
  container_name: cryptocoin-coins-db
  restart: always
  env_file:
    - .env
  expose:
    - 5432
  environment:
    POSTGRES_USER: ${DB_USERNAME}
    POSTGRES_PASSWORD: ${DB_PASSWORD}
    POSTGRES_DB: ${DB_COINS}
  volumes:
    - ./cryptocoin_backend/coins_data:/var/lib/postgresql/data
```

Рисунок 8 - coins-db

ВЫВОД

В ходе работы получили практические навыки по работе с *Docker* и *Docker Compose*.