

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бек-энд разработка

Отчет

Лабораторная Работа №4

Выполнил:

Кобелев Л.К.

К33401

Проверил:

Добряков Д. И.

Санкт-Петербург

2023 г.

## Задача

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

## Ход работы

Предположим, что у нас есть:

1. Клиентская часть (vue.js);
2. Серверная:
  - a. Public API (ts + express + typeorm);
  - b. ML API (fastapi)

Следовательно, каждый из них должен иметь свой Dockerfile.

Фронт:

---

```
FROM node:lts-alpine

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

EXPOSE 3000

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

Бэк (да, он такой же, потому что нода):

---

```
FROM node:lts-alpine

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

EXPOSE 3000

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

ML микросервис:

```
FROM python:3.8-slim

COPY requirements.txt /app/requirements.txt

RUN pip install --no-cache-dir --upgrade -r /app/requirements.txt

COPY . /app

WORKDIR /app

EXPOSE 3000

CMD ["uvicorn", "api:app", "--reload", "--host", "0.0.0.0", "--port", "3000", "--log-level", "critical"]
```

Теперь объединяем всё это в компоуз, который будет находиться в репозитории основного сервиса бэка (хотя при желании под компоуз можно было создать отдельный репозиторий).

```
version: '3'

services:
  backend:
    restart: always
    build: ./app
    ports:
      - "3000:3000"
    container_name: backend
  ml:
    restart: always
    build: ../Steam-Better-RecSys-ML/app
    ports:
      - "3001:3000"
    container_name: ml
  frontend:
    restart: always
    build: ../Steam-Better-RecSys-Frontend
    ports:
      - "3002:3000"
    container_name: frontend
```

Теперь, например, мы можем обращаться по адресу контейнера:

ML\_HOST=http://ml:3000/ -- строка .env, которая используется в RecommendationsController

## Вывод

Изучена основы Docker и docker-compose.