САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа 3

Выполнил:

Литвак Игорь

Группа К33401

Проверил: Добряков Д. И.

Санкт-Петербург

2022 г.

Задача

Необходимо реализовать отдельный микросервис, выполняющий какую-либо содержательную функцию из всего арсенала функций вашего приложения.

Ход работы

Решено выделить механизм аутентификации и модель пользователя в отдельный микросервис.

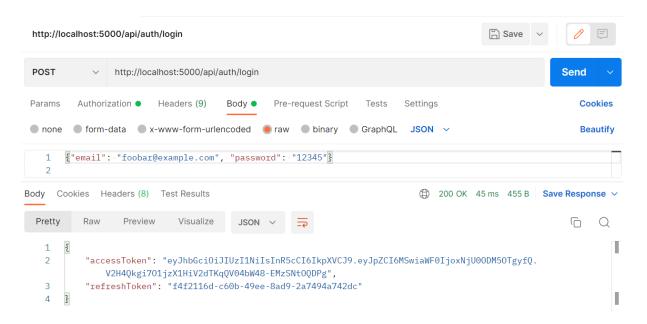
Для проверки того, что пользователь авторизован, добавлен новый эндпоинт:

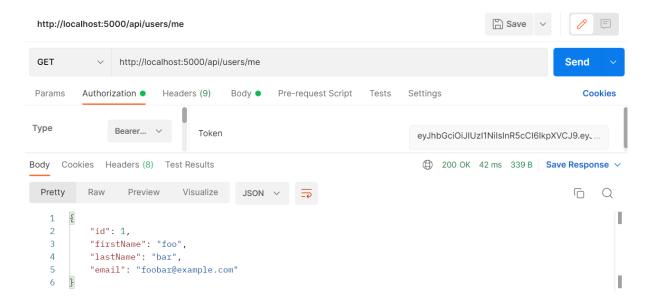
```
validateToken = async (request: any, response: any) => {
    const {body} = request
    const {accessToken} = body
    try {
        const payload = jwt.verify(accessToken, jwtConfig.secret)
        // @ts-ignore
        const user = await this.userService.getById(payload.id)
        response.send({'valid': true, 'user': user})
} catch (e: any) {
        response.status(401).send({'valid': false})
}
```

Для пользователя сохранена структура API из ЛР №2. Перенесенные эндпоинты проксируются с помощью библиотеки Axios:

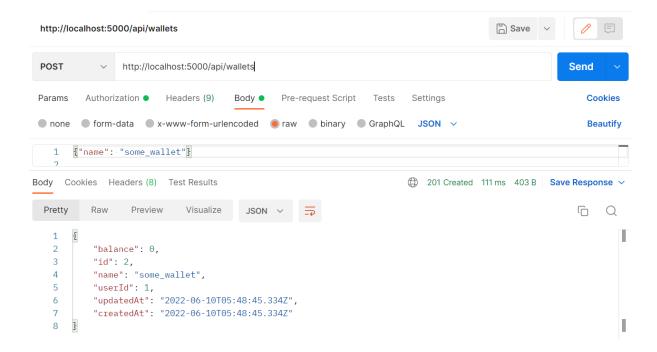
Middleware для авторизации переписан для создания запроса к микросервису авторизации:

Пример работы перенесенных эндпоинтов:





Пример работы эндпоинтов основного сервиса:



Вывод

Я разделил монолитное веб-приложение на node.js на два микросервиса, взаимодействующих при помощи API