САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Отчет

Лабораторная работа 3: Микросервисы

Выполнил:

Конев Антон

Группа К33402

Проверил:
Добряков Д. И.

Санкт-Петербург

2023 г.

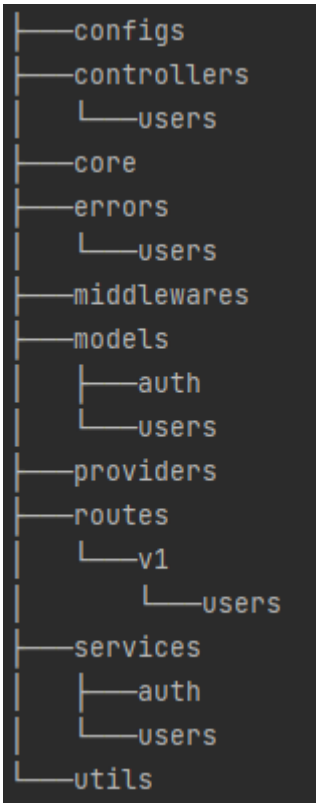
Задача

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

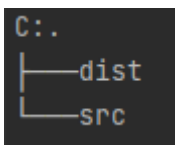
Ход работы

Структура частей приложения

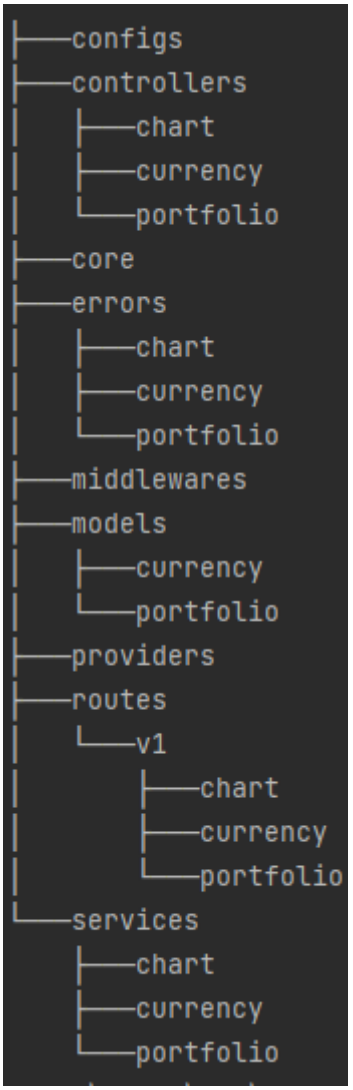
Auth



Gateway



Currency



Для проверки авторизации был добавлен новый эндпоинт

```
router.route( prefix: '/validate')
    .post(controller.validateToken);
```

```
validateToken = async (request: any, response: any) : Promise<void>  => {
    const {accessToken} = request.body;
    try {
        const payload : string | jwt.JwtPayload  = jwt.verify(accessToken, jwtOptions.secretOrKey);
        // @ts-ignore
        const user : User  = await this.userService.getById(payload.id)
        response.send({'valid': true, 'user': user})
    } catch (error: any) {
        response.status(401).send({'valid': false})
    }
}
```

Для проверки авторизации в приложении криптовалюты мы отправляем запрос в микросервис авторизации посредством нового Middleware

```
const customJwtStrategy : Strategy<VerifyFunctions>  = new Strategy( verify: async (token: any, done: any) : Promise<void>  => {
    axios.post(
        url: `http://auth:8001/users/validate`,
        data: {'accessToken': token}
    ).then((resp : AxiosResponse<any, any> ) : void  => {
        if (resp.data.valid) {
            const user = resp.data.user
            done(null, user)
        } else {
            done(null, false)
        }
    }).catch((error) : void  => {
        done(error)
    })
})
```

Также для проксирования запросов был реализован Gateway

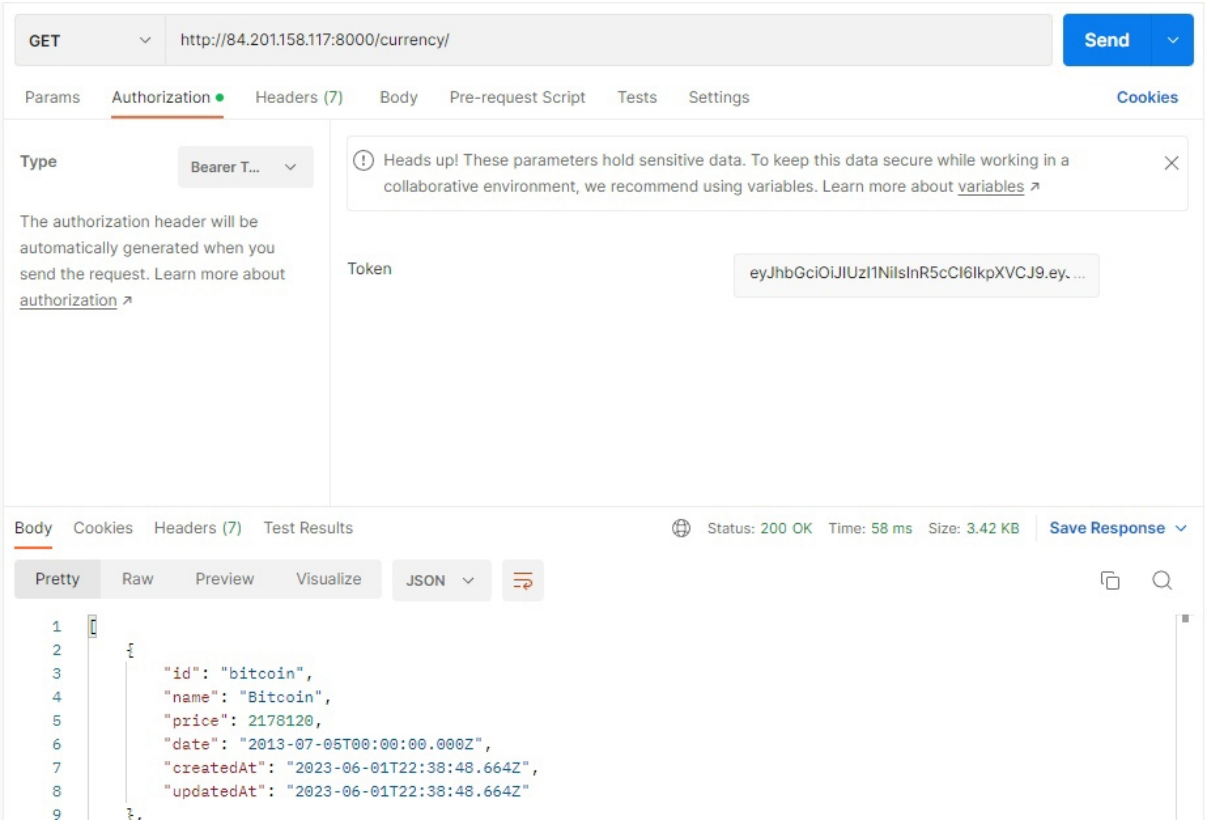
```
const app = express();

app.use('/users', createProxyMiddleware({target: 'http://auth:8001', changeOrigin: true}))
app.use('/currency', createProxyMiddleware({target: 'http://currency:8002', changeOrigin: true}))

Ant
app.listen(8000, () : void  => {
    console.log('Gateway is running on port 8000')
})
```

Эндпоинты сервиса криптовалюты теперь выглядят следующим образом

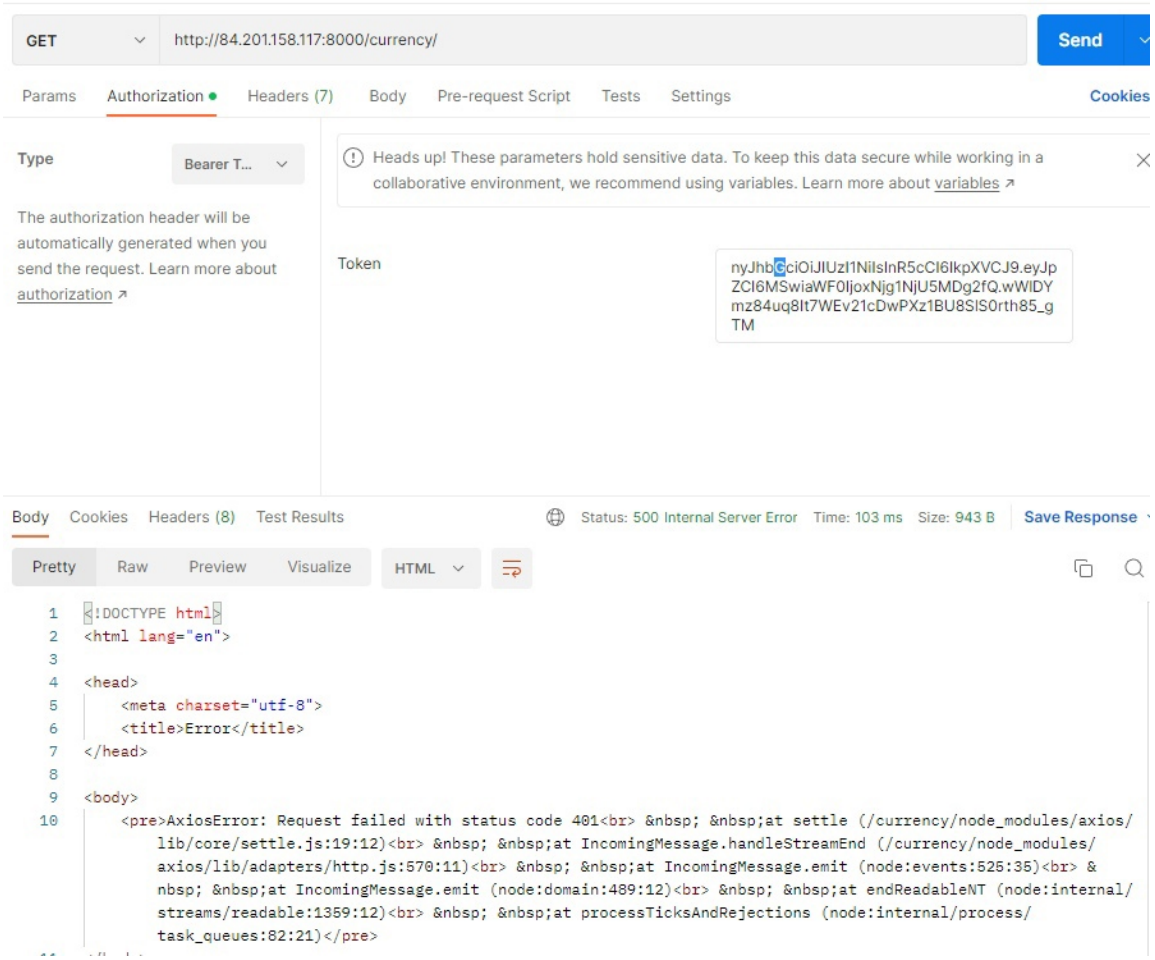
```
router.route( prefix: '/:id')
    .get(passport.authenticate( strategy: 'bearer',  options: { session: false }),controller.findByUserId)

router.route( prefix: '/buy')
    .post(passport.authenticate( strategy: 'bearer',  options: { session: false }),controller.buyCurrency)

router.route( prefix: '/sell')
    .post(passport.authenticate( strategy: 'bearer',  options: { session: false }),controller.sellCurrency)
```

Тестирование в Postman

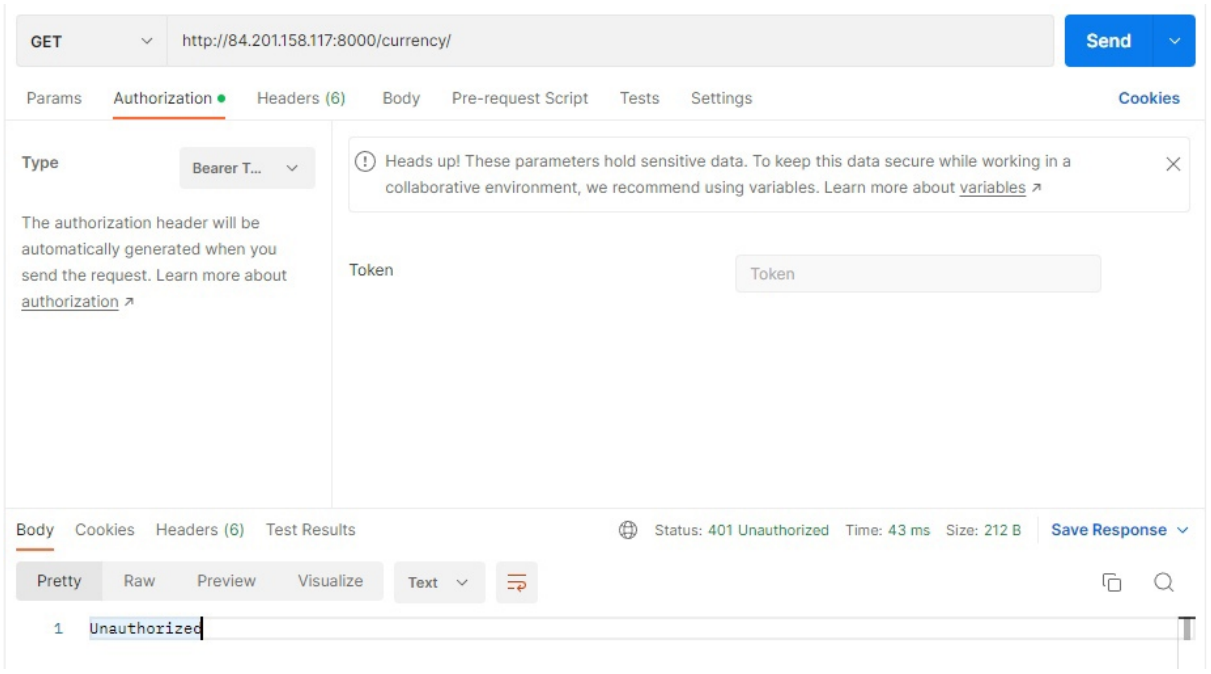
Валидный токен



Протухший токен



Отсутствие токена



Вывод

В ходе лабораторной работы монолитный проект из лабораторной работы 2 был разделен на микросервис авторизации, сервис криптовалюты и gateway их проксирующий.