

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бек-энд разработка

Отчет

Лабораторная работа
“Микросервисы”

Выполнили:
Соколовская Арина
Пронина Мария

К333392

Проверил:
Добряков Д. И.

Санкт-Петербург

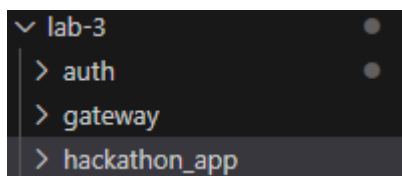
2024 г.

Задание

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

Ход работы

1. Выделено 2 микросервиса: для работы с пользователями и для основного функционала приложения. Структура проекта:



2. gateway для переадресации:

```
import express from "express"
import proxy from "express-http-proxy"
import 'dotenv/config'

const app = express();
app.use(express.json());

app.use('/auth', proxy(`http://localhost:${process.env.auth_port}`));
app.use('/', proxy(`http://localhost:${process.env.app_port}`));

app.listen(process.env.port, () => {
  console.log(`Running server on port ${process.env.port}`);
});
```

You, 59 seconds ago • Uncommitted changes

3. Тестирование микросервиса auth:

Получение пользователя по id:

GET{{api_url}}/auth/users/info/1

Send

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

BodyCookiesHeaders (6)Test Results

PrettyRawPreviewVisualize

Status: 200 OKTime: 64 msSize: 437 BSave as example

```
{
  "id": 1,
  "name": null,
  "surname": null,
  "email": "maria",
  "password": "$2b$95$Jwdtc5xFr6ZdxM0pG.0r1.y7mnv1EbHnX1Y3PYw6b8GhAgwP6dw1K",
  "about": null,
  "role_name": "admin",
  "createdAt": "2024-05-27T08:57:17.554Z",
  "updatedAt": "2024-05-27T09:40:38.981Z"
}
```

Изменение информации о пользователе:

PATCH{{api_url}}/auth/users/info/1

Send

ParamsAuthorizationHeaders (10)BodyPre-request ScriptTestsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1 {

2 | "name": "Maria"

3 | }

BodyCookiesHeaders (6)Test Results

PrettyRawPreviewVisualizeHTML

Status: 200 OKTime: 355 msSize: 440 BSave as example

```
1 {
  "id": 1,
  "name": "Maria",
  "surname": null,
  "email": "maria",
  "password": "$2b$95$Jwdtc5xFr6ZdxM0pG.0r1.y7mnv1EbHnX1Y3PYw6b8GhAgwP6dw1K",
  "about": null,
  "role_name": "admin",
  "createdAt": "2024-05-27T08:57:17.554Z",
  "updatedAt": "2024-05-27T10:14:12.016Z"
}
```

4. Тестирование микросервиса hackthon app

Создание нового хакатона (для проверки прав доступа (создавать хакатон может только пользователь с ролью админ) сервис посылает запрос на микросервис auth):

The screenshot shows a REST client interface for the 'Hackatons API' with the endpoint 'admin / Добавление нового хакатона'. The request is a POST method to '({api_url})/admin/hackathon' with a JSON body. The status is 200 OK, time is 87 ms, and size is 378 B.

Request:

```
POST {{api_url}}/admin/hackathon
```

Body:

```
{
  "name": "test 1",
  "description": "desc for hackthon test 1",
  "task": "task for hackthon test 1"
}
```

Response:

```
{
  "id": 1,
  "name": "test 1",
  "description": "desc for hackthon test 1",
  "task": "task for hackthon test 1",
  "updatedAt": "2024-05-27T18:13:00.289Z",
  "createdAt": "2024-05-27T18:13:00.289Z"
}
```

Получение краткой информации о хакатоне по id:

The screenshot shows a REST client interface for the 'Hackatons API' with the endpoint 'hackathon / Просмотр описания хакатона'. The request is a GET method to '({api_url})/hackathons/1'. The status is 200 OK, time is 21 ms, and size is 386 B.

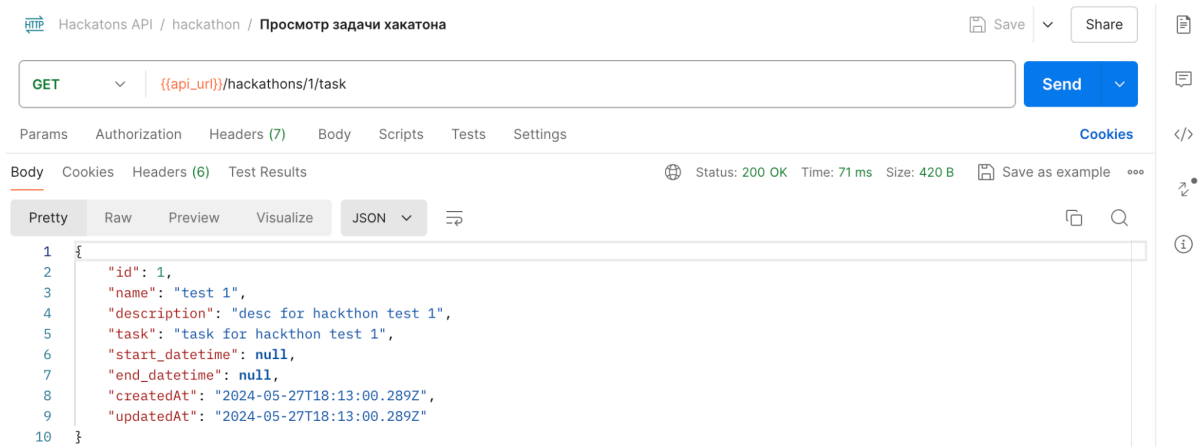
Request:

```
GET {{api_url}}/hackathons/1
```

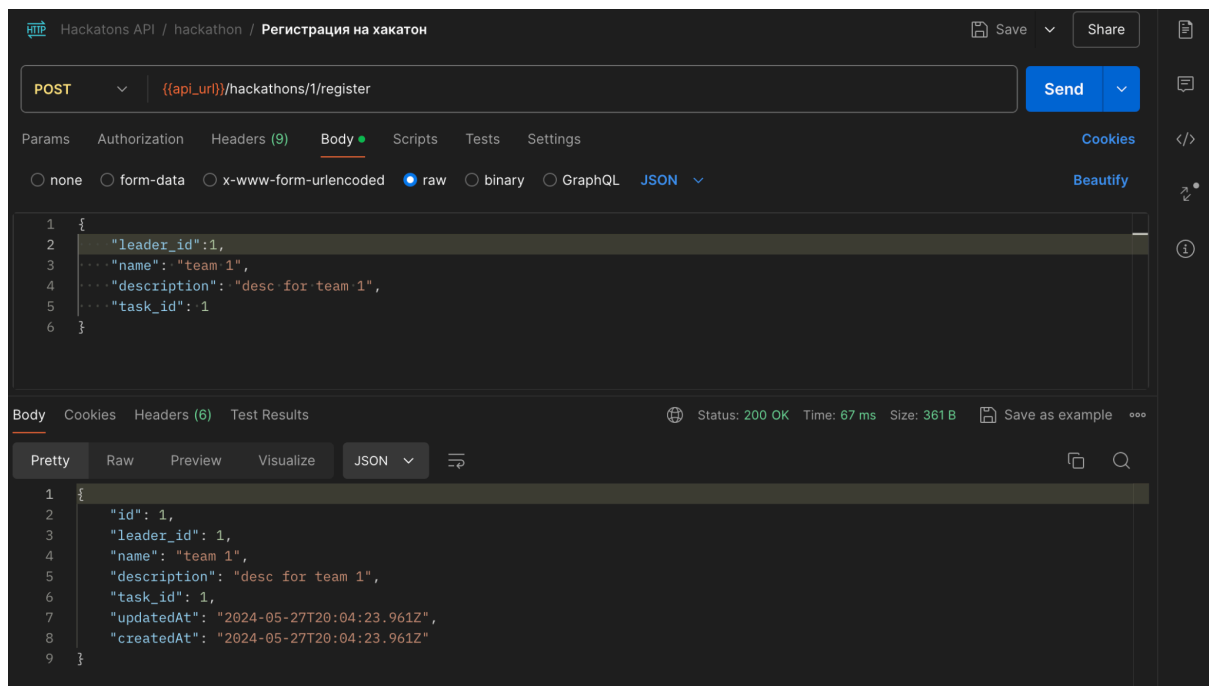
Response:

```
{
  "id": 1,
  "name": "test 1",
  "description": "desc for hackthon test 1",
  "start_datetime": null,
  "end_datetime": null,
  "createdAt": "2024-05-27T18:13:00.289Z",
  "updatedAt": "2024-05-27T18:13:00.289Z"
}
```

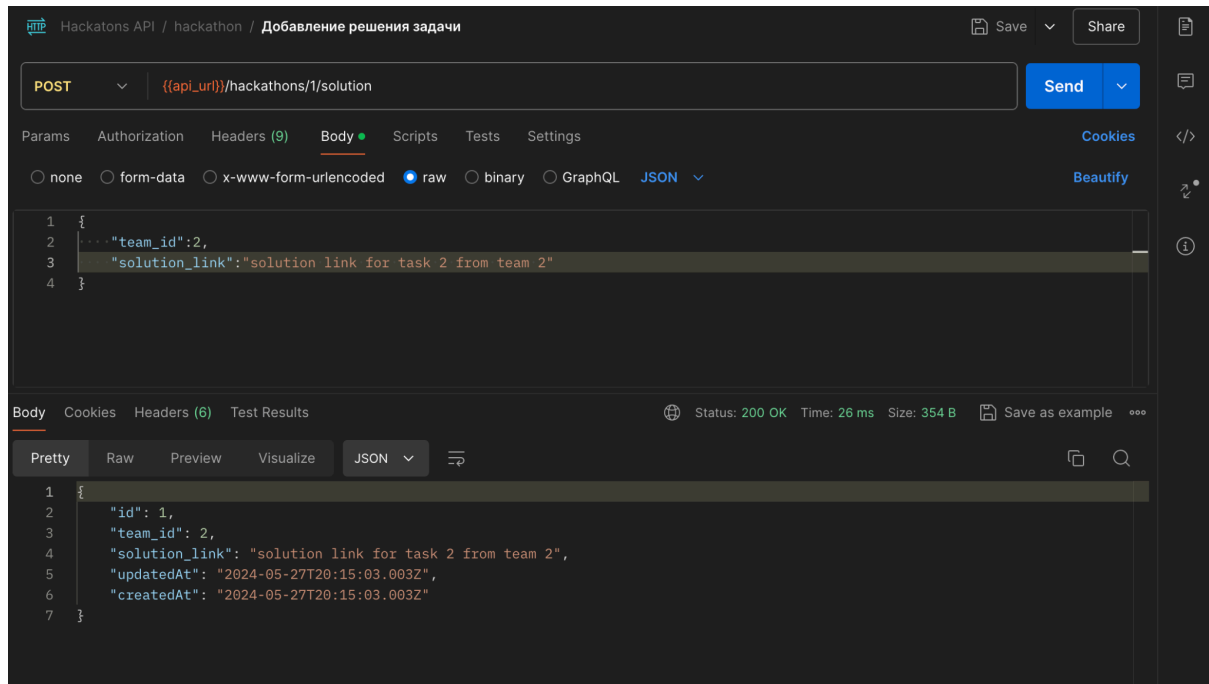
Получение полной информации о хакатоне и задаче хакатона по id (права доступа также проверяются через запрос на auth микросервис):



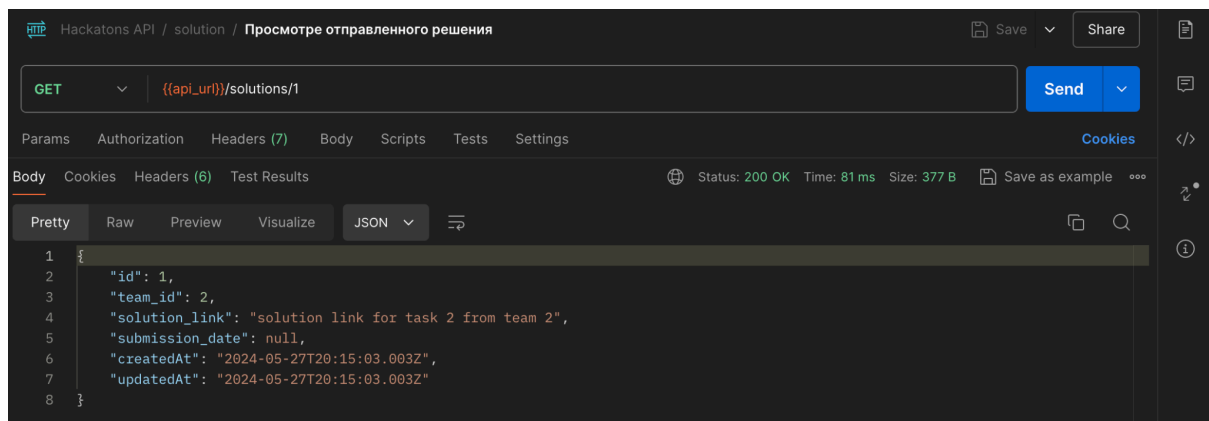
Регистрация пользователя на хакатон (права доступа также проверяются через запрос на auth микросервис):



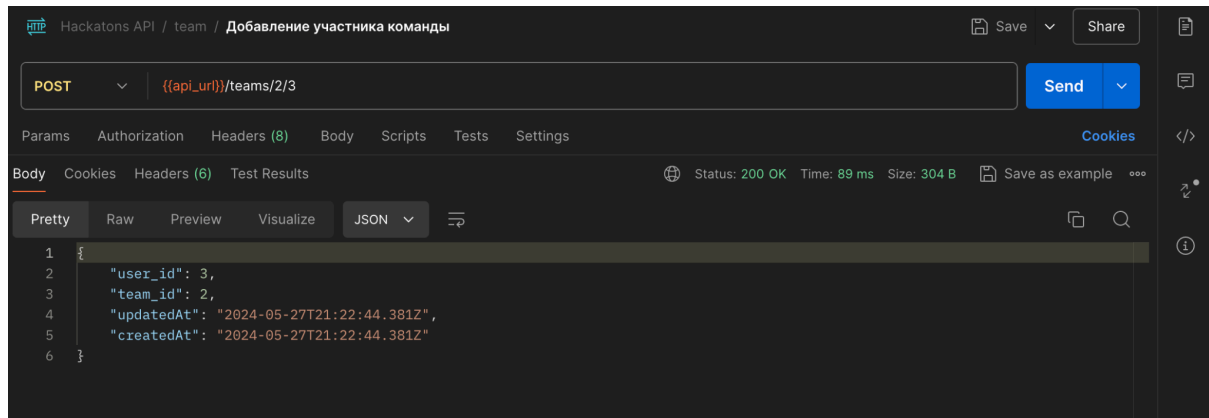
Добавление решения для задачи хакатона (права доступа также проверяются через запрос на auth микросервис):



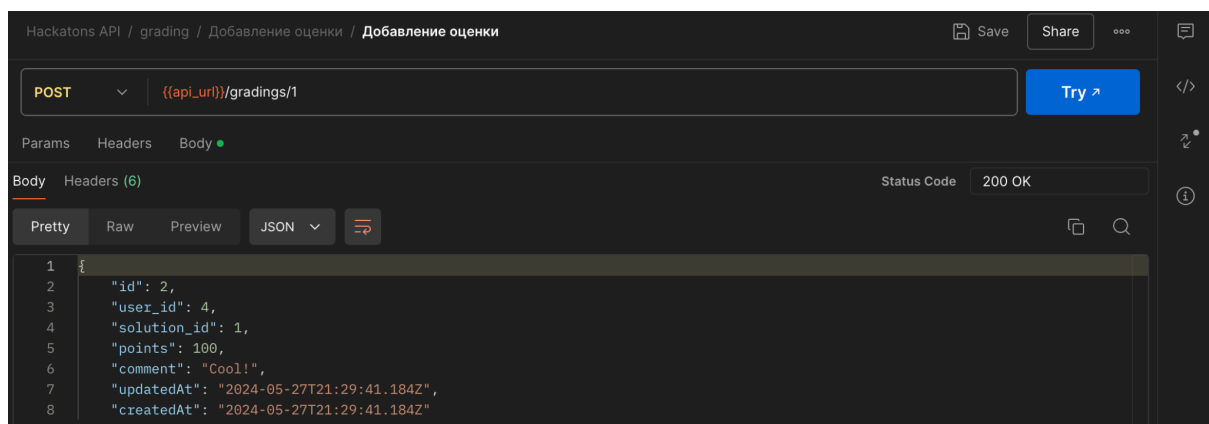
Просмотр решения на задачу хакатона (права доступа также проверяются через запрос на auth микросервис):



Добавление участника в команду (права доступа также проверяются через запрос на auth микросервис):



Выставление оценки за решение задачи хакатона (права доступа также проверяются через запрос на auth микросервис):



Вывод:

В ходе выполнения лабораторной работы были реализованы отдельные микросервисы для работы с пользователями и основного функционала, также был настроен gateway для маршрутизации запросов.