

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бек-энд разработка

Отчет по
домашней работе “Тестирование”

Выполнил:
Соколовская Арина

Группа:
К33392

Проверил:
Добряков Д. И.

Санкт-Петербург

2024 г.

Задание:

Тестирование API средствами Postman (написать 2-3 теста для конкретных эндпоинтов + 1 общий тест для всех эндпоинтов)

Ход работы:

1. Общий тест:

Время ответа сервера не более 200 мс:

The screenshot shows the Postman interface for a GET request to `http://localhost:8080/hackathons/`. The 'Tests' tab is active, displaying a JavaScript test script:

```
1 pm.test('Server response time less than 200ms', function () {  
2   pm.expect(pm.response.responseTime).to.be.below(200)  
3 })
```


Below the script, the 'Test Results (1/1)' tab is selected, showing a single test result:

All	Passed	Skipped	Failed
<div>PASS Server response time less than 200ms</div>			

2. Тестирование конкретных эндпоинтов:


2.1. Регистрация пользователя

Проверяем данные из запроса и ответа на совпадение:

POST  http://localhost:8080/auth/register

Params Authorization Headers (9) Body ● Pre-request Script **Tests ●** Settings


```
1 pm.test("Valid Response", () => {
2   const request = JSON.parse(pm.request.body.raw)
3   const response = JSON.parse(pm.response.stream.toString())
4   pm.expect(request.name).to.eq(response.name)
5   pm.expect(request.surname).to.eq(response.surname)
6   pm.expect(request.email).to.eq(response.email)
7 })
```

Body Cookies Headers (7) **Test Results (1/1)** 

All Passed Skipped Failed


PASS Valid Response

2.2. Авторизация пользователя

POST  http://localhost:8080/auth/login

Params Authorization Headers (9) Body ● Pre-request Script **Tests ●** Settings


```
1 pm.test("Token is valid", function () {
2   pm.expect(pm.response.json()).to.have.property("token")
3   const token = pm.response.json()["token"]
4   const data = JSON.parse(Buffer.from(token.split('.')[1], 'base64').
5     toString())
6   const request = JSON.parse(pm.request.body.raw)
7   pm.expect(data['email']).to.equal(request['email'])
8 })
```

Body Cookies Headers (7) **Test Results (1/1)** 

All Passed Skipped Failed

PASS Token is valid

2.3. Создание нового хакатона

POST  http://localhost:8080/admin/hackathon

Params Authorization Headers (10) Body ● Pre-request Script Tests ●


```
1 pm.test("Valid Response", () => {
2   const request = JSON.parse(pm.request.body.raw)
3   const response = JSON.parse(pm.response.stream.toString())
4   pm.expect(request.name).to.eq(response.name)
5   pm.expect(request.description).to.eq(response.description)
6   pm.expect(request.task).to.eq(response.task)
7 })
```

Body Cookies Headers (7) Test Results (1/1)

All Passed Skipped Failed

PASS Valid Response

2.4. Назначение куратора на хакатон

POST  http://localhost:8080/admin/curator/2

Params Authorization Headers (10) Body ● Pre-request Script Tests ●

```
1 pm.test("Valid Response", () => {
2   const request = JSON.parse(pm.request.body.raw)
3   const response = JSON.parse(pm.response.stream.toString())
4   pm.expect(request.task_id).to.eq(response.task_id)
5   pm.expect(request.user_id).to.eq(response.user_id)
6 })
```

Body Cookies Headers (7) Test Results (1/1)

All Passed Skipped Failed

PASS Valid Response

2.5. Удаление куратора

DELETE  http://localhost:8080/admin/curator/7

Params Authorization Headers (8) Body Pre-request Script **Tests ●**

```
1  pm.test("Deleted Successfully", () => {
2      const id = pm.request.url.path[2]
3      const token = pm.request.Authorization
4      pm.sendRequest({
5          method: "GET",
6          header: {"Authorization": token},
7          url: `http://localhost:8080/admin/curator/${id}`
8      }, (err, res) => {
9          pm.expect(Number(res.code)).to.eq(Number(403))
10     })
11 })
```

Body Cookies Headers (7) **Test Results (1/1)**

All Passed Skipped Failed

PASS Deleted Successfully

Вывод

В ходе выполнения домашней работы были протестированы некоторые эндпоинты из лабораторной работы 2 при помощи Postman.