

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Домашняя работа №4
Тестирование API

Выполнил:
Оспельников Алексей

Группа К3340

Проверил:
Добряков Д. И.

Санкт-Петербург

2025 г.

Задача

Задание:

- 1) реализовать тестирование API средствами Postman;
- 2) написать тесты внутри Postman.

Ход работы

- 1) Тест на SLI в 200 мс. Он будет общим для всех запросов

```
pm.test("Response time < 200 ms", function () {  
    pm.expect(pm.response.responseTime).to.be.below(200);  
});
```

Рисунок 1 - Тест на SLI

- 2) Тест на аутентификацию по JWT токenu

```
pm.test("Authentication Test - Status code is 401/403", function () {  
    pm.expect(pm.response.code).to.be.oneOf([401, 403]);  
});
```

Рисунок 2 - Тест на аутентификацию

- 3) Тест на правильное автозаполнение поля created_at

```
pm.test("Validate that 'created_at' field is a valid date and not in the future", function () {  
    const responseData = pm.response.json();  
  
    pm.expect(responseData).to.be.an('object');  
    pm.expect(responseData.data).to.have.property('created_at').that.satisfies(function(dateString) {  
        const date = new Date(dateString);  
        if (isNaN(date)) {  
            return false;  
        }  
        const now = new Date();  
        const dateOnly = new Date(date.getFullYear(), date.getMonth(), date.getDate());  
        const nowOnly = new Date(now.getFullYear(), now.getMonth(), now.getDate());  
  
        return dateOnly <= nowOnly;  
    }, "The 'created_at' field must be a valid date format and not in the future");  
});
```

Рисунок 3 - Тест на автозаполнение поля created_at

4) Тест на наличие полей при создании

```
pm.test("Response contains required fields", function () {
  const responseData = pm.response.json();

  pm.expect(responseData).to.be.an('object');
  pm.expect(responseData.data).to.exist.and.to.be.an('object');

  const requiredFields = ['fullname', 'phone', 'city', 'email', 'username',
    'password', 'id', 'created_at'];

  requiredFields.forEach(field => {
    pm.expect(responseData.data).to.have.property(field);
  });
});
```

Рисунок 4 - Тест на наличие полей при создании

5) Тест на правильный формат почты

```
pm.test("Email is in a valid format", function () {
  const responseData = pm.response.json();

  pm.expect(responseData).to.be.an('object');
  pm.expect(responseData.data).to.have.property('email').that.is.a('string').and.matches(
    /^[^\s@]+@[^\s@]+\.[^\s@]+$/, "Email format is invalid"
  );
});
```

Рисунок 5 - Тест на правильный формат почты

Вывод

За время работы были созданы 5 тестов для тестирования API средствами Postman