

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Образовательная программа «Программная инженерия»

**СОГЛАСОВАНО**

Старший преподаватель департамента  
больших данных и информационного  
поиска



В.В. Куренков

«13» \_\_\_\_\_ мая \_\_\_\_\_ 2023 г.

**УТВЕРЖДАЮ**

Академический руководитель  
образовательной программы  
«Программная инженерия»  
профессор департамента программной  
инженерии, канд. техн. наук

В.В. Шилов

«\_\_» \_\_\_\_\_ 2023 г.

**Система проверки решений по программированию для Лицея НИУ ВШЭ  
с автоматизированным подсчётом рейтинга**

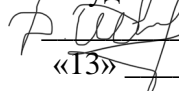
**Руководство оператора**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.05.10-01 34 01-1-ЛУ**

Исполнитель

студент группы БПИ219

 Д. И. Швецов /  
«13» \_\_\_\_\_ мая \_\_\_\_\_ 2023 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

**Москва 2023**

УТВЕРЖДЕН  
RU.17701729.05.10-01 34 01-1-ЛУ

**Система проверки решений по программированию для Лицея НИУ ВШЭ  
с автоматизированным подсчётом рейтинга**

**Руководство оператора**

**RU.17701729.05.10-01 34 01-1**

**Листов 45**

<i>Подп. и дата</i>	
<i>Инв. № дубл.</i>	
<i>Взам. инв. №</i>	
<i>Подп. и дата</i>	
<i>Инв. № подл</i>	

## ОГЛАВЛЕНИЕ

<b>1.</b>	<b>НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ .....</b>	<b>4</b>
1.1.	Функциональное назначение.....	4
1.2.	Эксплуатационное назначение.....	4
1.3.	Описание функций .....	4
<b>2.</b>	<b>УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ .....</b>	<b>6</b>
2.1.	Требования к составу и параметрам технических средств .....	6
2.2.	Минимальный состав программных средств .....	6
2.3.	Требования к оператору .....	6
<b>3.</b>	<b>ЗАПУСК ПРИЛОЖЕНИЯ .....</b>	<b>7</b>
3.1.	Установка программы .....	7
3.2.	Запуск и работа программы.....	7
3.2.1.	Работа с функцией «Регистрация нового пользователя в систему» .....	8
3.2.2.	Работа с функцией «Получение информации о профиле учащегося» .....	10
3.2.3.	Работа с функцией «Получение информации о профиле учителя».....	11
3.2.4.	Работа с функцией «Авторизация пользователя в систему» .....	13
3.2.5.	Работа с функцией «Добавление нового преподавателя в систему с помощью администраторских прав» .....	14
3.2.6.	Работа с функцией «Получение информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания» .....	16
3.2.7.	Работа с функцией «Получение информации о рейтинге, отсортированном по количеству задач, пользователей в порядке убывания» .....	20
3.2.8.	Работа с функцией «Скачивание информации о рейтинге, отсортированном по оценкам, пользователей в порядке убывания в формате .csv» .....	23
3.2.9.	Работа с функцией «Скачивание информации о рейтинге, отсортированном по количеству решенных задач, пользователей в порядке убывания в формате .csv» .....	25

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

# **RU.17701729.05.10-01 34 01–1**

<b>3.2.10. Работа с функцией «Получение информации о всех посылках (т. е. решениях) определённого пользователя» .....</b>	<b>26</b>
<b>3.2.11. Работа с функцией «Добавление решения определенной задачи определённого контекста в систему».....</b>	<b>28</b>
<b>3.2.12. Работа с функцией «Добавление нового контекста с задачами преподавателем в систему».....</b>	<b>30</b>
<b>3.2.13. Работа с функцией «Получение информации о всех контекстах, в которых пользователь принимает участие» .....</b>	<b>32</b>
<b>3.2.14. Работа с функцией «Получение информации о всех контекстах, имеющихся в системе»</b>	<b>34</b>
<b>3.2.15. Работа с функцией «Получение информации о всех посылках всех пользователей системы».....</b>	<b>38</b>
<b>3.2.16. Работа с функцией «Получение информации о роли пользователя системы»</b>	<b>40</b>
<b>3.2.17. Работа с функцией «Получения информации о контексте по его id в системе»</b>	<b>41</b>
<b>4. СООБЩЕНИЯ ОПЕРАТОРУ .....</b>	<b>44</b>
<b>ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....</b>	<b>45</b>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## 1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

### 1.1. Функциональное назначение

Программа содержит в себе систему, тестирующую решения задач по программированию Лицея НИУ ВШЭ и автоматически подсчитывающую рейтинг каждого учащегося, исходя из его оценок и количества решенных задач. Имеет учебно-методическое назначение.

### 1.2. Эксплуатационное назначение

Программа направлена на решение учащимися Лицея НИУ ВШЭ задач по программированию, которые им выдал тот или иной преподаватель, просмотр учащимися своего рейтинга по оценкам и количеству решенных задач среди других учащихся. Также программа направлена на просмотр преподавателями решений задач учащимися, получения их рейтинга в виде файлов.

Конечные пользователи – учащиеся и преподаватели информатики/программирования Лицея НИУ ВШЭ.

### 1.3. Описание функций

API «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга» должен выполнять следующие функции:

1. Регистрация нового пользователя в систему - добавление информации о новом пользователе в базу данных.
2. Получение информации о профиле учащегося - возвращение информации об ученике, имеющейся в базе данных.
3. Получение информации о профиле учителя - возвращение информации об учителе, имеющейся в базе данных.
4. Авторизация пользователя в систему - проверка на валидность email и пароля пользователя и возвращение роли пользователя и JWT-токена для дальнейшей работы с системой.
5. Добавление нового преподавателя в систему с помощью администраторских прав - добавление информации о новом преподавателе в базу данных.
6. Получение информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания - получение информации в виде сортированного списка пользователей из базы данных, представляющего из себя рейтинг по оценкам.
7. Получение информации о рейтинге, отсортированном по количеству задач, пользователей в порядке убывания - получение информации в виде сортированного списка пользователей из базы данных, представляющего из себя рейтинг по количеству задач.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

8. Скачивание информации о рейтинге, отсортированном по оценкам, пользователей в порядке убывания в формате .csv - получение информации в виде сортированного списка пользователей из базы данных, о рейтинге по оценкам в файле формата .csv для загрузки.

9. Скачивание информации о рейтинге, отсортированном по количеству решенных задач, пользователей в порядке убывания в формате .csv - получение информации в виде сортированного списка пользователей из базы данных о рейтинге по количеству задач в файле формата .csv для загрузки.

10. Получение информации о всех посылках (т. е. решениях) определённого пользователя - получение всех посылок (решений), отправленных пользователем, из базы данных.

11. Добавление решения определенной задачи определенного контекста в систему - добавление решения, отправленного пользователем, в базу данных.

12. Добавление нового контекста с задачами преподавателем в систему - добавление нового контекста и задач, относящихся к этому контексту в базу данных.

13. Получение информации о всех контекстах, в которых пользователь принимает участие - получение информации о всех контекстах, в которых принимает участие пользователь из базы данных.

14. Получение информации о всех контекстах, имеющихся в системе - получение информации о всех контекстах из базы данных, которые ранее были добавлены в систему.

15. Получение информации о всех посылках всех пользователей системы - получение информации о всех посылках (решениях) из базы данных, которые ранее были добавлены в систему.

16. Получение информации о роли пользователя системы - получение информации о роли, которая относится к пользователю, из базы данных.

17. Получения информации о контексте по его id в системе - получение информации о контексте из базы данных по id этого контекста.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## 2. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

### 2.1. Требования к составу и параметрам технических средств

Для надёжной работы программы требуется следующий состав технических средств:

#### 1. Сервер приложений.

- 1.1. Процессор: 4 ядра (8 логических потоков), частота – 2 ГГц и больше.
- 1.2. Оперативная память: 8 ГБ и больше.
- 1.3. Свободное дисковое пространство: 256 ГБ и больше.
- 1.4. Пропускная способность сетевого интерфейса: 1 Гбит/с.

#### 2. Сервер баз данных.

- 2.1. Тип накопителя: SSD.
- 2.2. Процессор: 4 ядра (8 логических потоков), частота – 2 ГГц и больше.
- 2.3. Оперативная память: 8 ГБ и больше.
- 2.4. Свободное дисковое пространство: 512 ГБ и больше.
- 2.5. Пропускная способность сетевого интерфейса: 1 Гбит/с.

### 2.2. Минимальный состав программных средств

Для надежной и бесперебойной работы программы на сервере рекомендуется следующий состав программных средств:

- 1. Операционная система: Ubuntu 22.04.1 LTS
- 2. Java Development Kit (JDK) версии, используемой Spring Boot Framework, то есть версии 3.0.2.
- 3. Зависимости Spring Boot Framework.
- 4. Java 17 или более поздней версии.
- 5. Среда выполнения Java (Java Runtime Environment - JRE) для запуска приложения.
- 6. База данных PostgreSQL и драйвер JDBC для нее. PostgreSQL должен быть версии 11 или более поздней.

### 2.3. Требования к оператору

- 1. Пользователь должен свободно владеть русским и английским языком.
- 2. Пользователь должен иметь базовые навыки системного администрирования ПО для установки и запуска Spring Boot приложения и базы данных PostgreSQL на сервере.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

### 3. ЗАПУСК ПРИЛОЖЕНИЯ

#### 3.1. Установка программы

Для установки и настройки базы данных необходимо произвести следующие операции с PostgreSQL:

1. Скачать необходимую версию PostgreSQL на сервер.
1. Собрать скачанные пакеты, то есть произвести установку базы данных и драйвера.
2. Произвести настройку PostgreSQL.
3. Инициализировать новый кластер баз данных.
4. Создать новую базу данных для дальнейшей работы с программой.
5. Запустить database\_init.sql файл для создания таблиц в созданной на шаге 5 базе данных.

Для установки и настройки Spring Boot Application необходимо выполнить следующие действия:

1. Установить Java Runtime Environment (JRE) на сервере, а именно Java Development Kit (JDK) версии 3.0.2.
2. Создать скрипт для запуска приложения. В скрипте указать команду для запуска JAR-файла.
3. Настроить брандмауэр сервера, чтобы разрешить входящие запросы на порт, на котором работает приложение.
4. Настроить базу данных. Настроить соединение к ней в файле настроек Spring Boot приложения «application.properties».

#### 3.2. Запуск и работа программы

Для запуска Spring Boot приложения необходимо выполнить созданный ранее скрипт для запуска. Если все настройки при установке, описанные ранее, были правильно выполнены, приложение станет доступно по указанному порту (по умолчанию 8080).

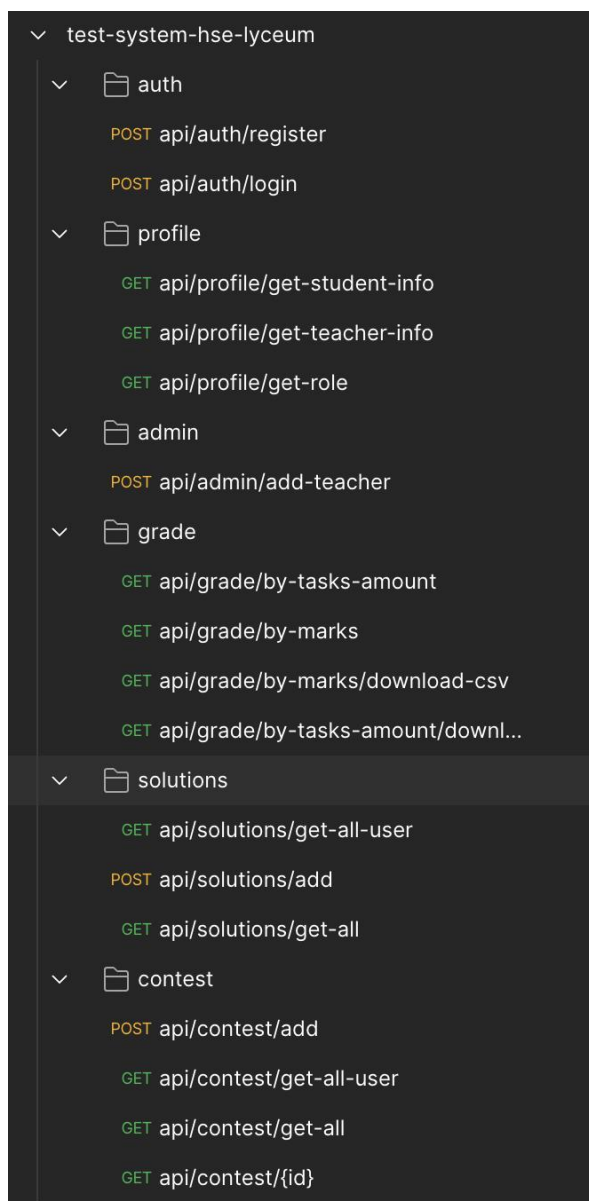
Для наглядности работы программы и каждого endpoint-а была создана коллекция запросов «test-system-hse-lyceum» в инструменте для тестирования API и создания запросов Postman. В созданной коллекции находится 6 папок для удобства разделения запросов по основным endpoint-ам:

- auth
- profile
- admin
- grade
- solutions

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



- contest



Скриншот коллекции в Postman:

### 3.2.1. Работа с функцией «Регистрация нового пользователя в систему»

Триггер: необходимо проверить работоспособность запроса на регистрацию нового пользователя в систему, который поступает со стороны клиента

Название папки коллекции в Postman: auth

Тип запроса: POST

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

Endpoint: `http://{ip}/api/auth/register`

Авторизация: No Auth

Параметры: нет

Тело запроса (body):

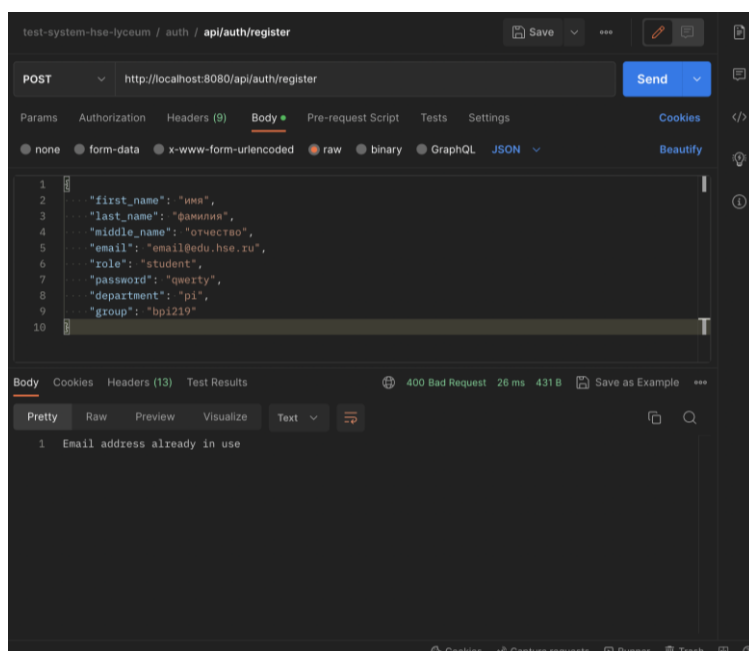
```
{
  "first_name": "имя",
  "last_name": "фамилия",
  "middle_name": "отчество",
  "email": "email@edu.hse.ru",
  "role": "student",
  "password": "qwerty",
  "department": "pi",
  "group": "bpi219"
}
```

Пример ответа в случае успешного выполнения запроса (статус код 200):

User was successfully created and registered

Пример ответа в случае возникновения ошибки (статус код 400):

Email address already in use



Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

`curl --location 'http://{ip}/api/auth/register' \`

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```
--header 'Content-Type: application/json' \
--data-raw '{
  "first_name": "имя",
  "last_name": "фамилия",
  "middle_name": "отчество",
  "email": "email@edu.hse.ru",
  "role": "student",
  "password": "qwerty",
  "department": "pi",
  "group": "bpi219"
}'
```

**3.2.2. Работа с функцией «Получение информации о профиле учащегося»**

Триггер: необходимо проверить работоспособность запроса на получение информации о профиле учащегося, который поступает со стороны клиента

Название папки коллекции в Postman: profile

Тип запроса: GET

Endpoint: `http://{ip}/api/profile/get-student-info`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

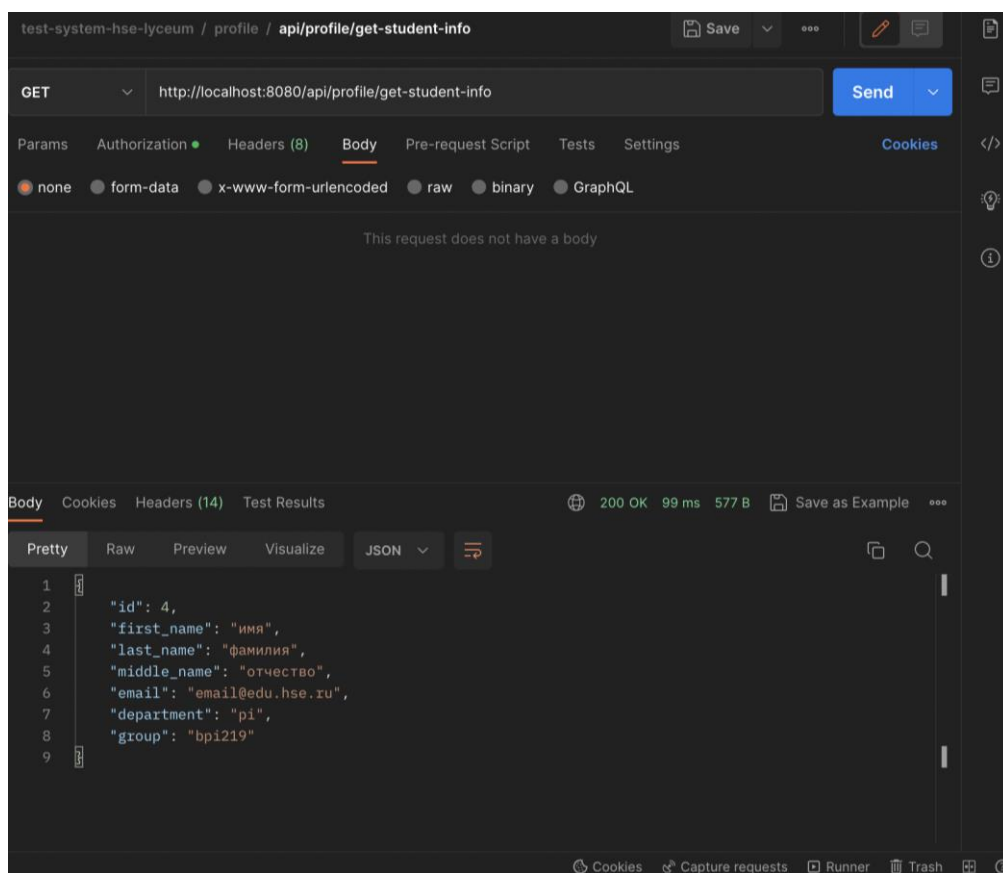
Пример ответа в случае успешного выполнения запроса (статус код 200):

```
{
  "id": 4,
  "first_name": "имя",
  "last_name": "фамилия",
  "middle_name": "отчество",
  "email": "email@edu.hse.ru",
  "department": "pi",
  "group": "bpi219"
}
```

Пример ответа в случае возникновения ошибки (статус код 400):

User with email: example has no education name

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/profile/get-student-info' \
--header 'Authorization: Bearer student_token'
```

### 3.2.3. Работа с функцией «Получение информации о профиле учителя»

Триггер: необходимо проверить работоспособность запроса на получение информации о профиле учителя, который поступает со стороны клиента

Название папки коллекции в Postman: profile

Тип запроса: GET

Endpoint: http://{ip}/api/profile/get-teacher-info

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

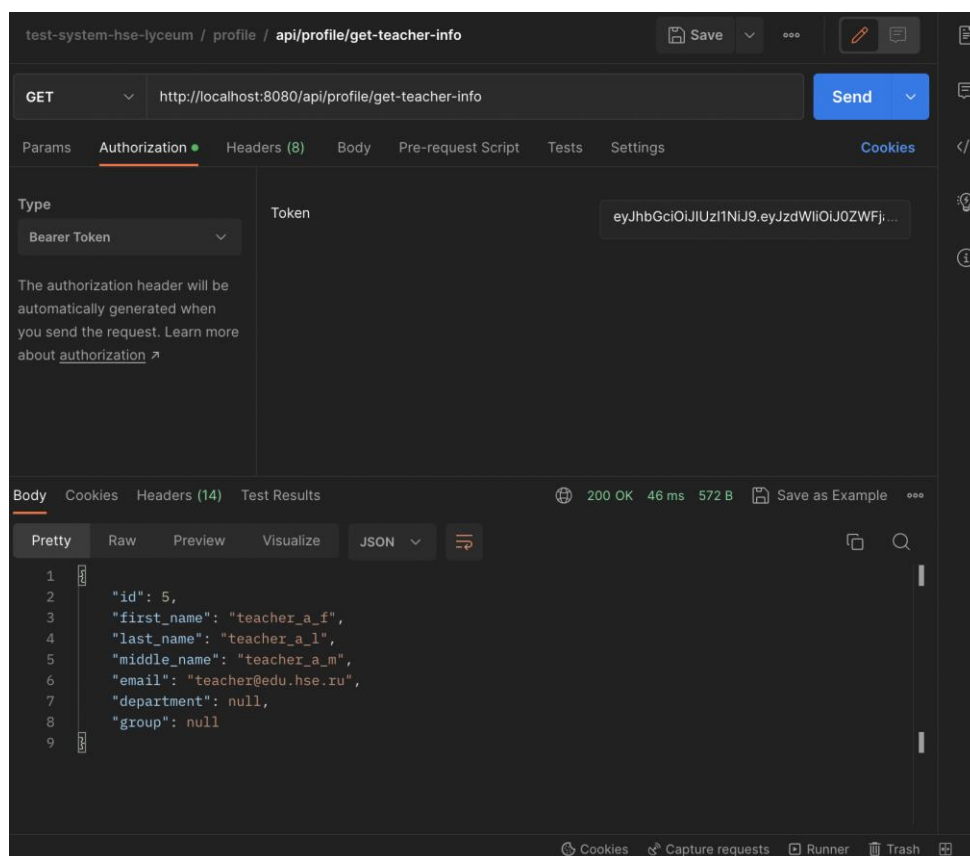
## RU.17701729.05.10-01 34 01–1

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
{
  "id": 5,
  "first_name": "teacher_a_f",
  "last_name": "teacher_a_l",
  "middle_name": "teacher_a_m",
  "email": "teacher@edu.hse.ru",
  "department": null,
  "group": null
}
```

Пример ответа в случае возникновения ошибки (статус код 400):

No user personal info with email: example found



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/profile/get-teacher-info' \
--header 'Authorization: Bearer teacher_token'
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

### 3.2.4. Работа с функцией « Авторизация пользователя в систему»

Триггер: необходимо проверить работоспособность запроса на авторизацию (вход) пользователя в систему, который поступает со стороны клиента

Название папки коллекции в Postman: auth

Тип запроса: POST

Endpoint: `http://{ip}/api/auth/login`

Авторизация: No Auth

Параметры: нет

Тело запроса:

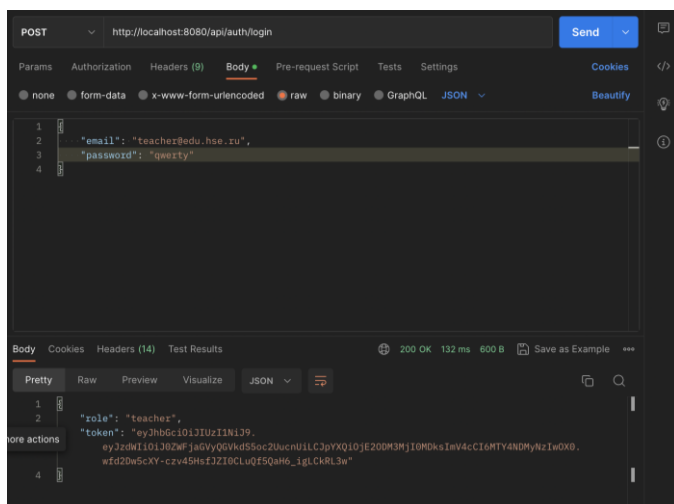
```
{
  "email": "teacher@edu.hse.ru",
  "password": "qwerty"
}
```

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
{
  "role": "teacher",
  "token":
  "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJ0ZWZjaGVyQGVkdS5oc2UucnUiLCJpYXQiOiJlE2ODM3MjE5MDQsImV4cCI6MTY4NDMyNjcwNH0xUT6ee8F1h08lAyOeh8Unbt_o5OV8kfY1ZsT1r69_Qc"
}
```

Пример ответа в случае возникновения ошибки (статус код 400):

Bad credentials



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/auth/login' \
--header 'Content-Type: application/json' \
--data-raw '{
  "email": "teacher@edu.hse.ru",
  "password": "qwerty"
}'
```

### 3.2.5. Работа с функцией «Добавление нового преподавателя в систему с помощью администраторских прав»

Триггер: необходимо проверить работоспособность запроса на добавление нового преподавателя в систему с помощью администраторских прав, который поступает со стороны клиента

Название папки коллекции: admin

Тип запроса: POST

Endpoint: http://{ip}/api/admin/add-teacher

Авторизация: Basic Auth (Username, Password)

Параметры: нет

Тело запроса:

```
{
  "first_name": "teacher_a_f",
  "last_name": "teacher_a_l",
  "middle_name": "teacher_a_m",
  "email": "teacher1@edu.hse.ru",
  "role": "teacher",
  "password": "qwerty",
  "department": "pi",
  "group": "bpi219"
}
```

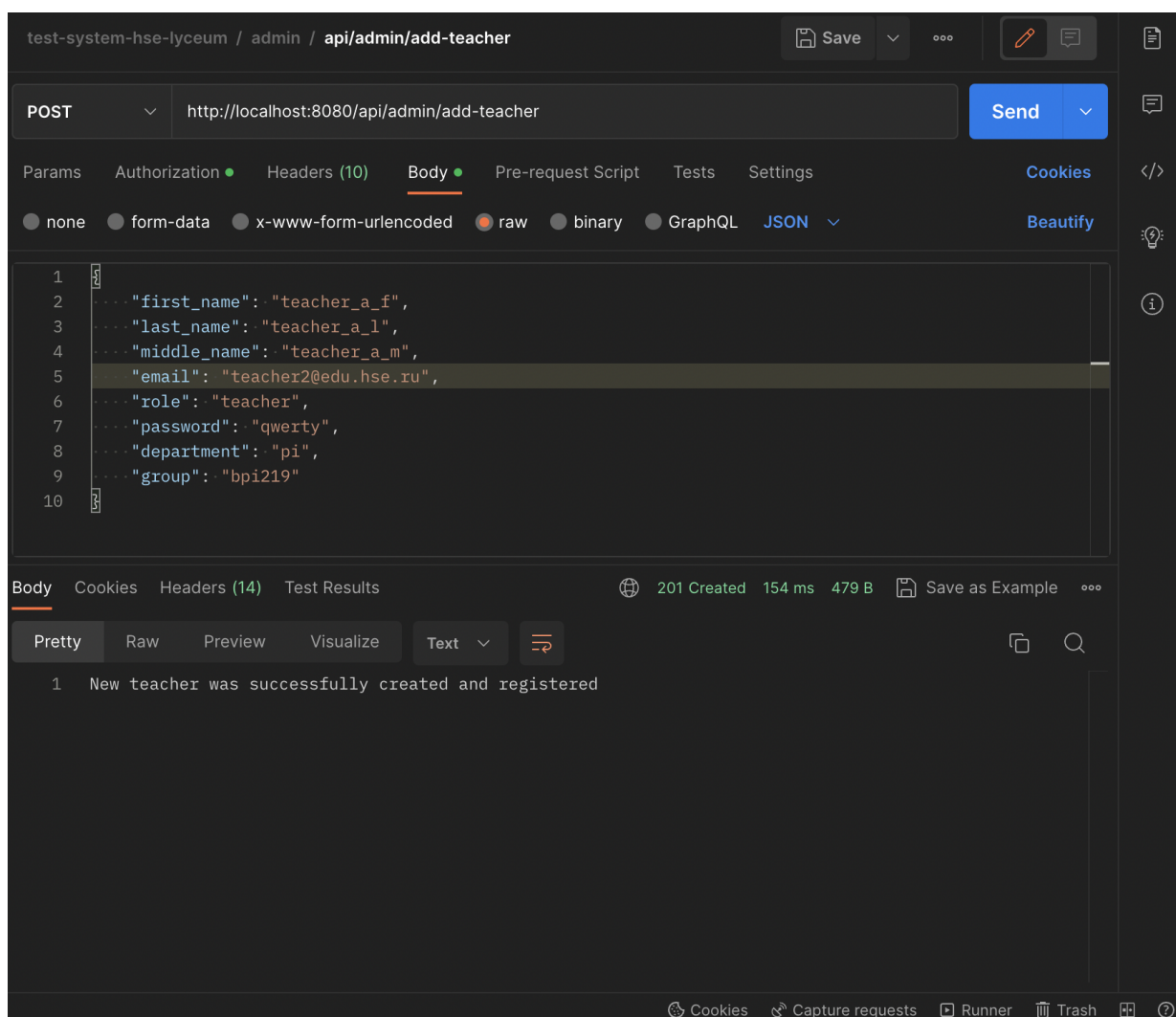
Пример ответа в случае успешного выполнения запроса (статус код 200):

New teacher was successfully created and registered

Пример ответа в случае возникновения ошибки (статус код 400):

Teacher with such email address already exists

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```

curl --location 'http://{ip}/api/admin/add-teacher' \
--header 'Content-Type: application/json' \
--header 'Authorization: Basic YWRtaW46YWRtaW4=' \
--data-raw '{
  "first_name": "teacher_a_f",
  "last_name": "teacher_a_l",
  "middle_name": "teacher_a_m",
  "email": "teacher2@edu.hse.ru",
  "role": "teacher",

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



**RU.17701729.05.10-01 34 01–1**

```
"password": "qwerty",
"department": "pi",
"group": "bpi219"
}'
```

### **3.2.6. Работа с функцией «Получение информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания»**

Триггер: необходимо проверить работоспособность запроса на получение информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания, который поступает со стороны клиента

Название папки коллекции в Postman: grade

Тип запроса: GET

Endpoint: `http://{ip}/api/grade/by-marks`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
[
  {
    "id": 1,
    "userShortInfo": {
      "email": "a@edu.hse.ru",
      "role": "student",
      "firstName": "A_F",
      "lastName": "A_L",
      "middleName": "A_M",
      "departmentName": "pi",
      "groupName": "bpi219"
    }
  },
  {
    "id": 2,
    "userShortInfo": {
      "email": "t_a@edu.hse.ru",
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```

    "role": "teacher",
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
},
{
  "id": 3,
  "userShortInfo": {
    "email": "b@edu.hse.ru",
    "role": "student",
    "firstName": "B_F",
    "lastName": "B_L",
    "middleName": "B_M",
    "departmentName": "pi",
    "groupName": "bpi219"
  }
},
{
  "id": 4,
  "userShortInfo": {
    "email": "email@edu.hse.ru",
    "role": "student",
    "firstName": "имя",
    "lastName": "фамилия",
    "middleName": "отчество",
    "departmentName": "pi",
    "groupName": "bpi219"
  }
},
{

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01-1**

```

"id": 5,
"userShortInfo": {
  "email": "teacher@edu.hse.ru",
  "role": "teacher",
  "firstName": "teacher_a_f",
  "lastName": "teacher_a_l",
  "middleName": "teacher_a_m",
  "departmentName": "null",
  "groupName": "null"
}
},
{
  "id": 6,
  "userShortInfo": {
    "email": "teacher1@edu.hse.ru",
    "role": "teacher",
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
},
{
  "id": 7,
  "userShortInfo": {
    "email": "teacher2@edu.hse.ru",
    "role": "teacher",
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
}

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

```

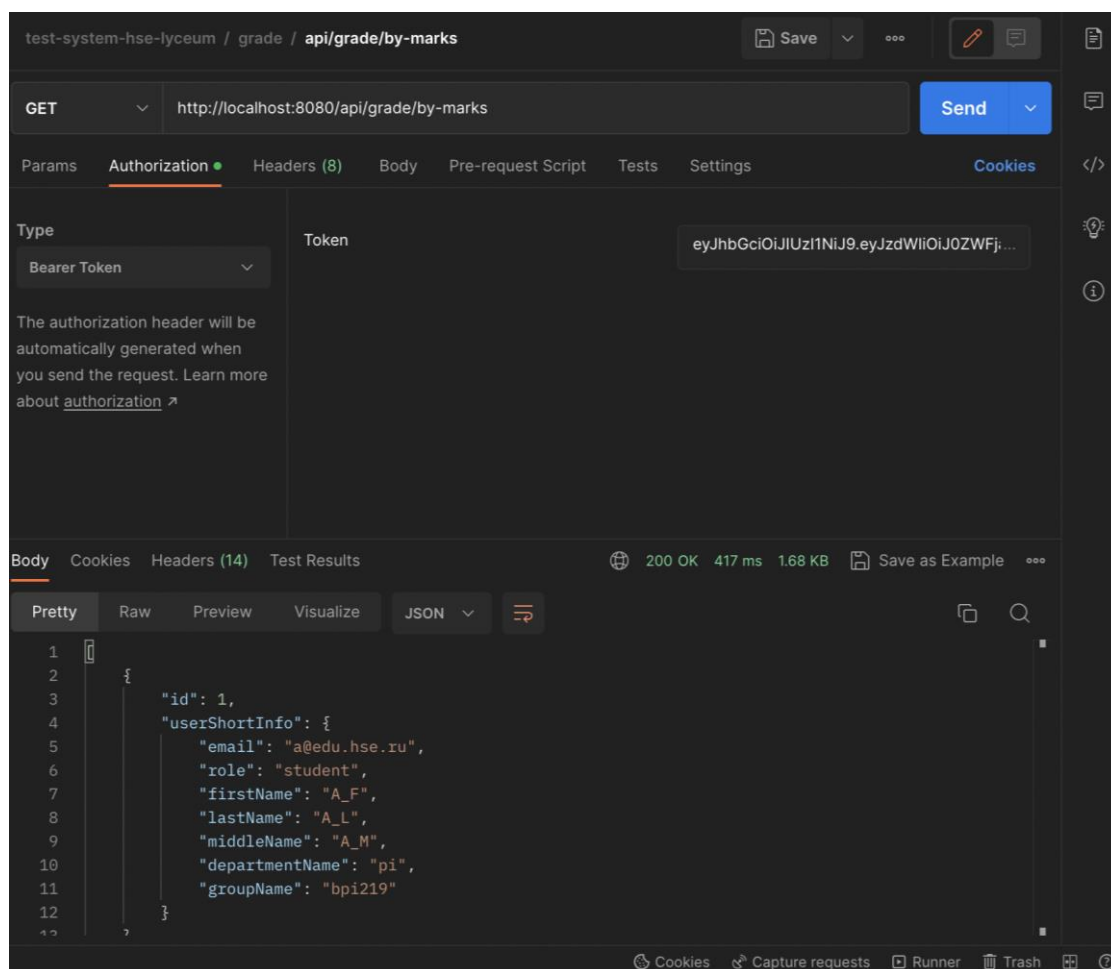
    }
  }
]

```

Пример ответа в случае возникновения ошибки (статус код 400):

User info was not found

Скриншот с коллекции «test-system-hse-lyceum» в Postman



Подпись

cURL code snippet для выполнения запроса с сервера:

```

curl --location 'http://{ip}/api/grade/by-marks' \
--header 'Authorization: Bearer user_token'

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1****3.2.7. Работа с функцией «Получение информации о рейтинге, отсортированном по количеству задач, пользователей в порядке убывания»**

Триггер: необходимо проверить работоспособность запроса на получение информации о рейтинге, отсортированном по количеству задач пользователей в порядке убывания, который поступает со стороны клиента

Название папки коллекции в Postman: grade

Тип запроса: GET

Endpoint: `http://{ip}/api/grade/by-tasks-amount`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
[
  {
    "id": 1,
    "userShortInfo": {
      "email": "a@edu.hse.ru",
      "role": "student",
      "firstName": "A_F",
      "lastName": "A_L",
      "middleName": "A_M",
      "departmentName": "pi",
      "groupName": "bpi219"
    }
  },
  {
    "id": 2,
    "userShortInfo": {
      "email": "t_a@edu.hse.ru",
      "role": "teacher",
      "firstName": "teacher_a_f",
      "lastName": "teacher_a_l",
      "middleName": "teacher_a_m",

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```

    "departmentName": "null",
    "groupName": "null"
  }
},
{
  "id": 3,
  "userShortInfo": {
    "email": "b@edu.hse.ru",
    "role": "student",
    "firstName": "B_F",
    "lastName": "B_L",
    "middleName": "B_M",
    "departmentName": "pi",
    "groupName": "bpi219"
  }
},
{
  "id": 4,
  "userShortInfo": {
    "email": "email@edu.hse.ru",
    "role": "student",
    "firstName": "имя",
    "lastName": "фамилия",
    "middleName": "отчество",
    "departmentName": "pi",
    "groupName": "bpi219"
  }
},
{
  "id": 5,
  "userShortInfo": {
    "email": "teacher@edu.hse.ru",
    "role": "teacher",

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```

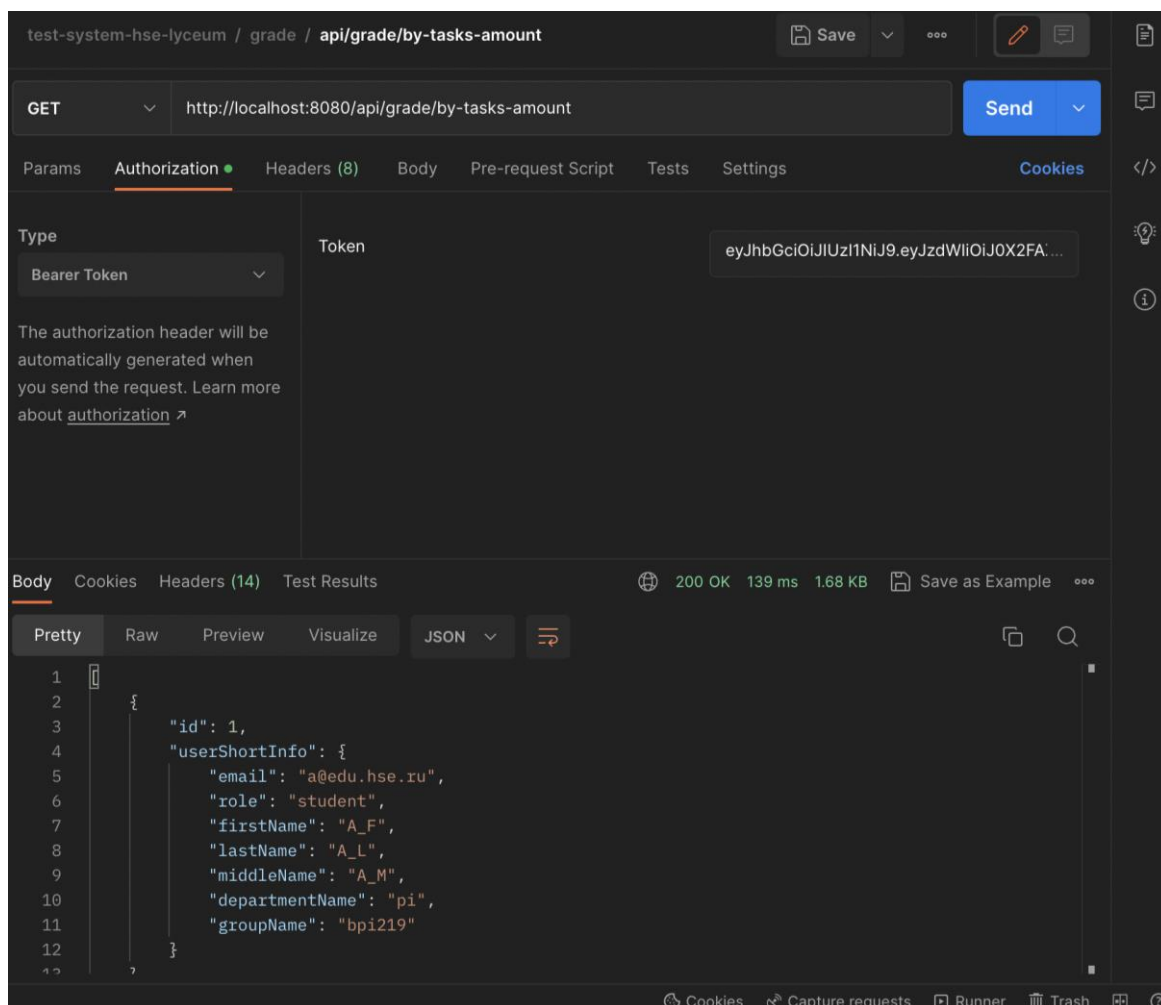
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
},
{
  "id": 6,
  "userShortInfo": {
    "email": "teacher1@edu.hse.ru",
    "role": "teacher",
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
},
{
  "id": 7,
  "userShortInfo": {
    "email": "teacher2@edu.hse.ru",
    "role": "teacher",
    "firstName": "teacher_a_f",
    "lastName": "teacher_a_l",
    "middleName": "teacher_a_m",
    "departmentName": "null",
    "groupName": "null"
  }
}
]

```

Пример ответа в случае возникновения ошибки (статус код 400):

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

User info was not found



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

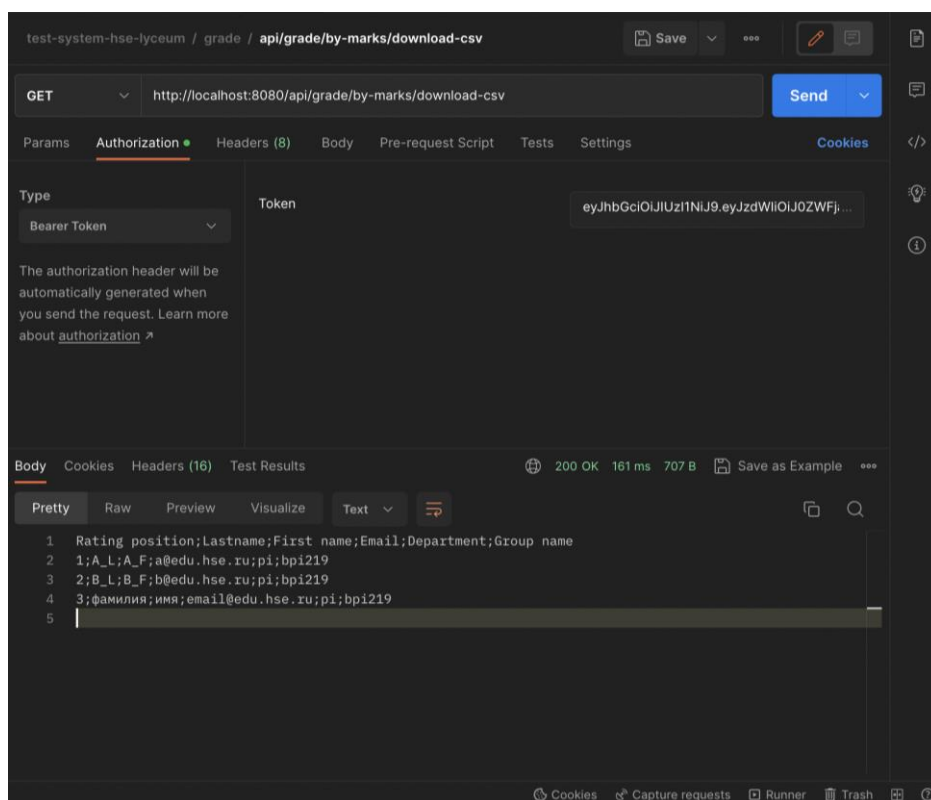
```
curl --location 'http://{ip}/api/grade/by-tasks-amount' \
--header 'Authorization: Bearer user_token'
```

### 3.2.8. Работа с функцией «Скачивание информации о рейтинге, отсортированном по оценкам, пользователей в порядке убывания в формате .csv»

Триггер: необходимо проверить работоспособность запроса на скачивание информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания, который поступает со стороны клиента

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и





Название папки коллекции в Postman: grade

Тип запроса: GET

Endpoint: `http://{ip}/api/grade/by-marks/download-csv`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200) (содержимое файла):

Rating position;Lastname;First name;Email;Department;Group name

1;A\_L;A\_F;a@edu.hse.ru;pi;bpi219

2;B\_L;B\_F;b@edu.hse.ru;pi;bpi219

3;фамилия;имя;email@edu.hse.ru;pi;bpi219

Пример ответа в случае возникновения ошибки (статус код 400):

User info of user with email: <email> was not found

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```
curl --location 'http://{ip}/api/grade/by-marks/download-csv' \
--header 'Authorization: Bearer user_token'
```

### **3.2.9. Работа с функцией «Скачивание информации о рейтинге, отсортированном по количеству решенных задач, пользователей в порядке убывания в формате .csv»**

Триггер: необходимо проверить работоспособность запроса на скачивание информации о рейтинге, отсортированном по количеству задач пользователей в порядке убывания, который поступает со стороны клиента

Название папки коллекции в Postman: grade

Тип запроса: GET

Endpoint: `http://{ip}/api/grade/by-tasks-amount/download-csv`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200) (содержимое файла):

Rating position;Lastname;First name;Email;Department;Group name

1;A\_L;A\_F;a@edu.hse.ru;pi;bpi219

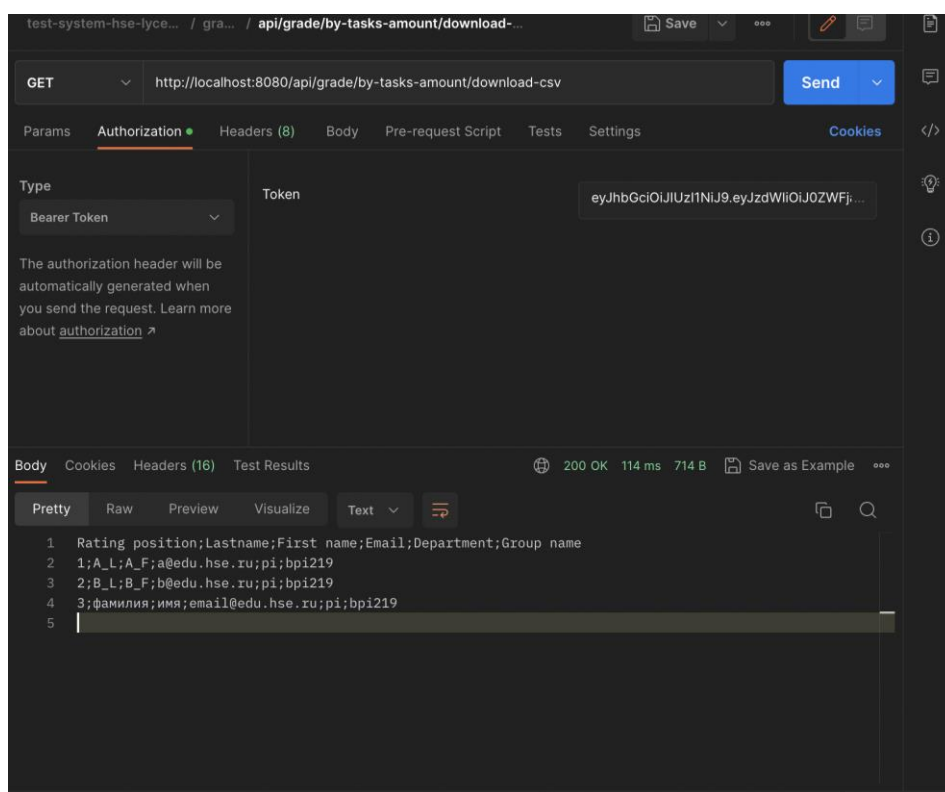
2;B\_L;B\_F;b@edu.hse.ru;pi;bpi219

3;фамилия;имя;email@edu.hse.ru;pi;bpi219

Пример ответа в случае возникновения ошибки (статус код 400):

User info of user with email: <email> was not found

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/grade/by-tasks-amount/download-csv' \
--header 'Authorization: Bearer user_token'
```

### 3.2.10. Работа с функцией «Получение информации о всех посылках (т. е. решениях) определённого пользователя»

Триггер: необходимо проверить работоспособность запроса на получение информации о всех посылках пользователя, который поступает со стороны клиента

Название папки коллекции в Postman: solutions

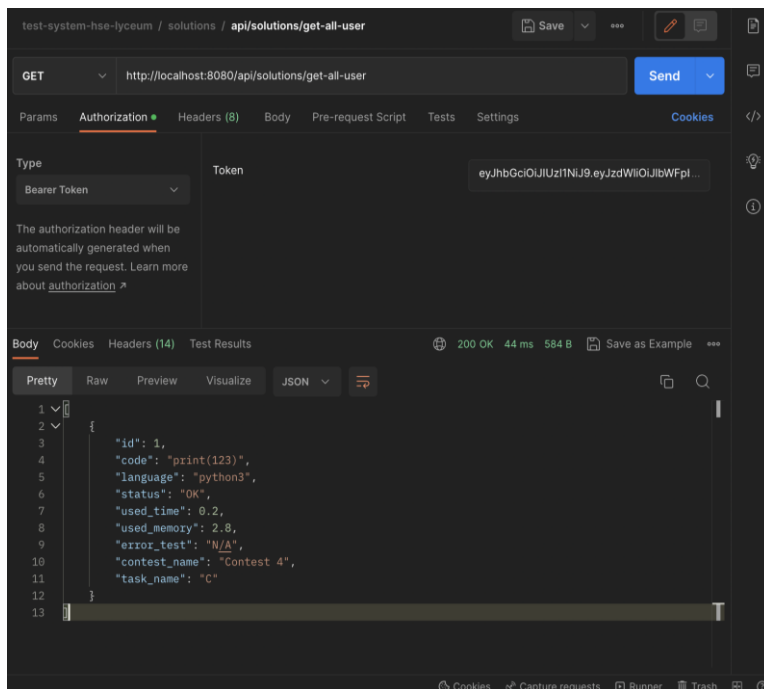
Тип запроса: GET

Endpoint: `http://{ip}/api/solutions/get-all-user`

Авторизация: Bearer Token

Параметры: нет

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

```

[
  {
    "id": 1,
    "code": "print(123)",
    "language": "python3",
    "status": "OK",
    "used_time": 0.2,
    "used_memory": 2.8,
    "error_test": "N/A",
    "contest_name": "Contest 4",
    "task_name": "C"
  }
]

```

Пример ответа в случае возникновения ошибки (статус код 400):

User solution entity has id that doesn't exist in database

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/solutions/get-all-user' \
```

```
--header 'Authorization: Bearer user_token'
```

### **3.2.11. Работа с функцией «Добавление решения определенной задачи определенного контекста в систему»**

Триггер: необходимо проверить работоспособность запроса на добавление решения задачи контекста, который поступает со стороны клиента

Название папки коллекции в Postman: solutions

Тип запроса: POST

Endpoint: http://{ip}/api/solutions/add

Авторизация: POST

Параметры: нет

Тело запроса:

```
{
  "code": "print(1234)",
  "language": "python3",
  "status": "OK",
  "used_time": 0.2,
  "used_memory": 2.8,
  "error_test": "N/A",
  "contest_name": "Contest 4",
  "task_name": "A"
}
```

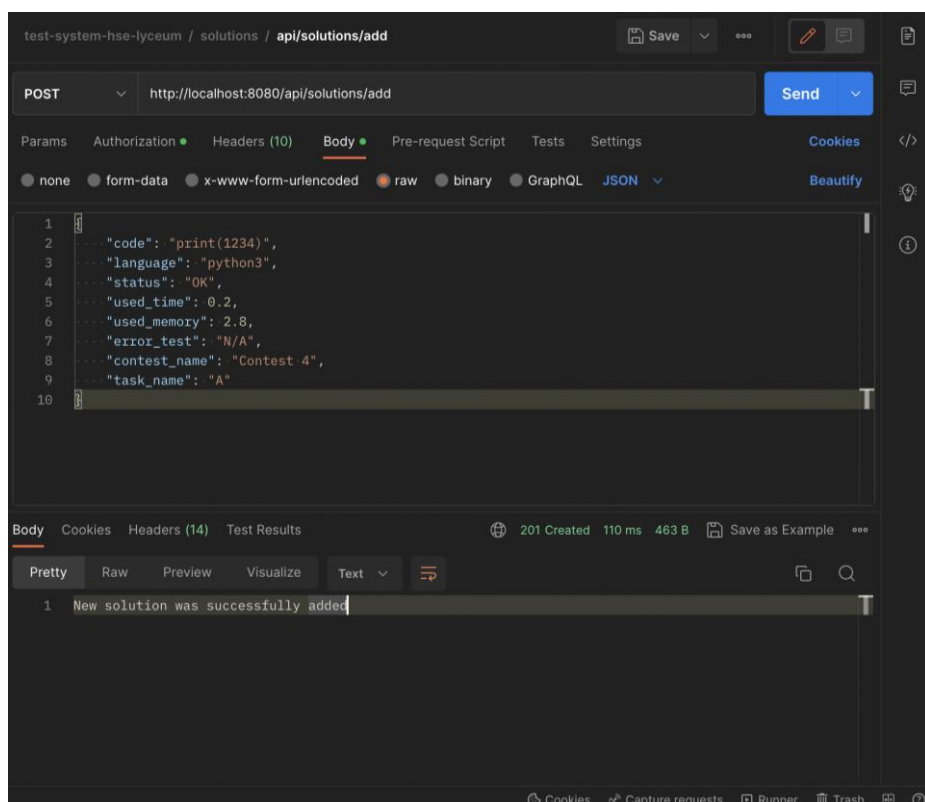
Пример ответа в случае успешного выполнения запроса (статус код 200):

New solution was successfully added

Пример ответа в случае возникновения ошибки (статус код 400):

Contest for contest tasks were not found

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```

curl --location 'http://{ip}/api/solutions/add' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer user_token' \
--data '{
  "code": "print(1234)",
  "language": "python3",
  "status": "OK",
  "used_time": 0.2,
  "used_memory": 2.8,
  "error_test": "N/A",
  "contest_name": "Contest 4",
  "task_name": "A"
}'

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

},

### 3.2.12. Работа с функцией «Добавление нового конкурса с задачами преподавателем в систему»

Триггер: необходимо проверить работоспособность запроса на добавление нового конкурса с задачами преподавателем в систему, который поступает со стороны клиента

Название папки коллекции в Postman: contest

Тип запроса: POST

Endpoint: http://{ip}/api/contest/add

Авторизация: Bearer Token

Параметры: нет

Тело запроса:

```
{
  "title": "Contest 4",
  "ejudge_id": 123543,
  "start_time": "2023-05-02T18:00:00",
  "finish_time": "2023-05-04T18:00:00",
  "tasks":
  [
    {
      "title": "A",
      "description": "this is the first task",
      "memory_limit": 32,
      "time_limit": 2,
      "attempts_amount": 100,
      "classification_title": "type_1"
    },
    {
      "title": "B",
      "description": "this is the second task",
      "memory_limit": 64,
      "time_limit": 2,
      "attempts_amount": 100,
      "classification_title": "type_2"
    },
    {
      "title": "C",
      "description": "this is the third task",
      "memory_limit": 32,
      "time_limit": 1,
      "attempts_amount": 100,
      "classification_title": "type_1"
    }
  ]
}
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

]

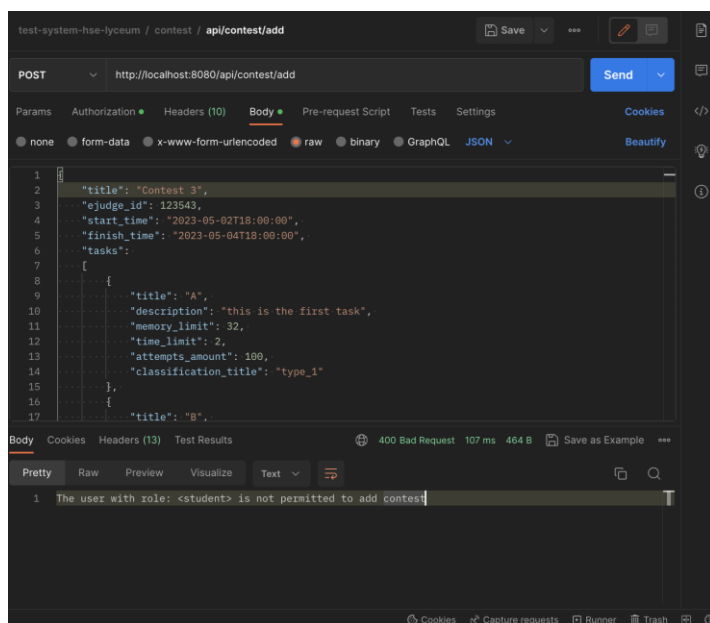
}

Пример ответа в случае успешного выполнения запроса (статус код 200):

New contest was successfully added to system

Пример ответа в случае возникновения ошибки (статус код 400):

The user with role: <student> is not permitted to add contest



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/contest/add' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer teacher_token' \
--data '{
  "title": "Contest 3",
  "ejudge_id": 123543,
  "start_time": "2023-05-02T18:00:00",
  "finish_time": "2023-05-04T18:00:00",
  "tasks":
  [
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



**RU.17701729.05.10-01 34 01–1**

```
{
  "title": "A",
  "description": "this is the first task",
  "memory_limit": 32,
  "time_limit": 2,
  "attempts_amount": 100,
  "classification_title": "type_1"
},
{
  "title": "B",
  "description": "this is the second task",
  "memory_limit": 64,
  "time_limit": 2,
  "attempts_amount": 100,
  "classification_title": "type_2"
},
{
  "title": "C",
  "description": "this is the third task",
  "memory_limit": 32,
  "time_limit": 1,
  "attempts_amount": 100,
  "classification_title": "type_1"
}
]
```

### **3.2.13. Работа с функцией «Получение информации о всех контекстах, в которых пользователь принимает участие»**

Триггер: необходимо проверить работоспособность запроса на получение информации о всех контекстах пользователя в системе, который поступает со стороны клиента

Название папки коллекции в Postman: contest

Тип запроса: GET

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01-1**

Endpoint: `http://{ip}/api/contest/get-all-user`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
[
  {
    "id": 1,
    "ejudge_id": 123543,
    "title": "Contest 4",
    "start_time": "2023-05-02T18:00:00",
    "finish_time": "2023-05-04T18:00:00",
    "tasks": [
      {
        "title": "A",
        "description": "this is the first task",
        "memory_limit": 32.0,
        "time_limit": 2.0,
        "attempts_amount": 100,
        "classification_title": "type_1"
      },
      {
        "title": "B",
        "description": "this is the second task",
        "memory_limit": 64.0,
        "time_limit": 2.0,
        "attempts_amount": 100,
        "classification_title": "type_2"
      },
      {
        "title": "C",
        "description": "this is the third task",
        "memory_limit": 32.0,
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

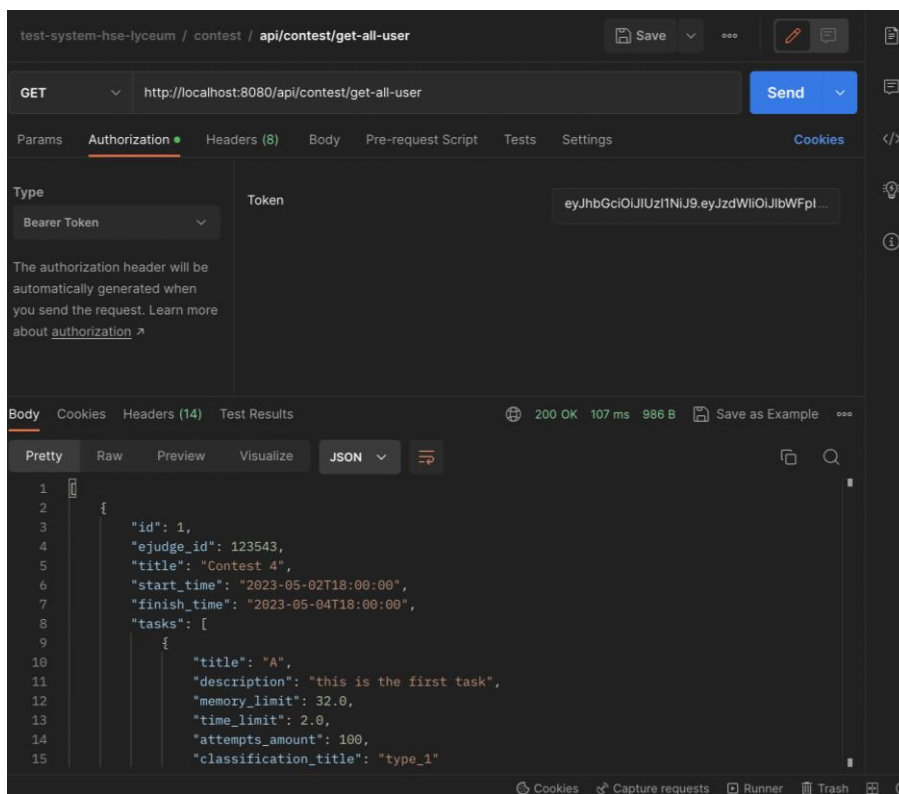
```

    "time_limit": 1.0,
    "attempts_amount": 100,
    "classification_title": "type_1"
  }
]
}
]

```

Пример ответа в случае возникновения ошибки (статус код 400):

The user with such email does not exist



Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/contest/get-all-user' \
--header 'Authorization: Bearer user_token'
```

### 3.2.14. Работа с функцией «Получение информации о всех контестах, имеющихся в системе»

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

Триггер: необходимо проверить работоспособность запроса на получение информации о всех контестах, имеющихся в системе, который поступает со стороны клиента

Название папки коллекции в Postman: contest

Тип запроса: GET

Endpoint: `http://{ip}/api/contest/get-all`

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

```
[
  {
    "id": 1,
    "ejudge_id": 123543,
    "title": "Contest 4",
    "start_time": "2023-05-02T18:00:00",
    "finish_time": "2023-05-04T18:00:00",
    "tasks": [
      {
        "title": "A",
        "description": "this is the first task",
        "memory_limit": 32.0,
        "time_limit": 2.0,
        "attempts_amount": 100,
        "classification_title": "type_1"
      },
      {
        "title": "B",
        "description": "this is the second task",
        "memory_limit": 64.0,
        "time_limit": 2.0,
        "attempts_amount": 100,
        "classification_title": "type_2"
      },
      {
        "title": "C",
        "description": "this is the third task",
        "memory_limit": 32.0,
        "time_limit": 1.0,
        "attempts_amount": 100,

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01-1**

```

    "classification_title": "type_1"
  }
]

},
{
  "id": 2,
  "ejudge_id": 123543,
  "title": "Contest 3",
  "start_time": "2023-05-02T18:00:00",
  "finish_time": "2023-05-04T18:00:00",
  "tasks": [
    {
      "title": "A",
      "description": "this is the first task",
      "memory_limit": 32.0,
      "time_limit": 2.0,
      "attempts_amount": 100,
      "classification_title": "type_1"
    },
    {
      "title": "B",
      "description": "this is the second task",
      "memory_limit": 64.0,
      "time_limit": 2.0,
      "attempts_amount": 100,
      "classification_title": "type_2"
    },
    {
      "title": "C",
      "description": "this is the third task",
      "memory_limit": 32.0,
      "time_limit": 1.0,
      "attempts_amount": 100,
      "classification_title": "type_1"
    }
  ]
}

},
{
  "id": 3,
  "ejudge_id": 123543,

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01-1**

```

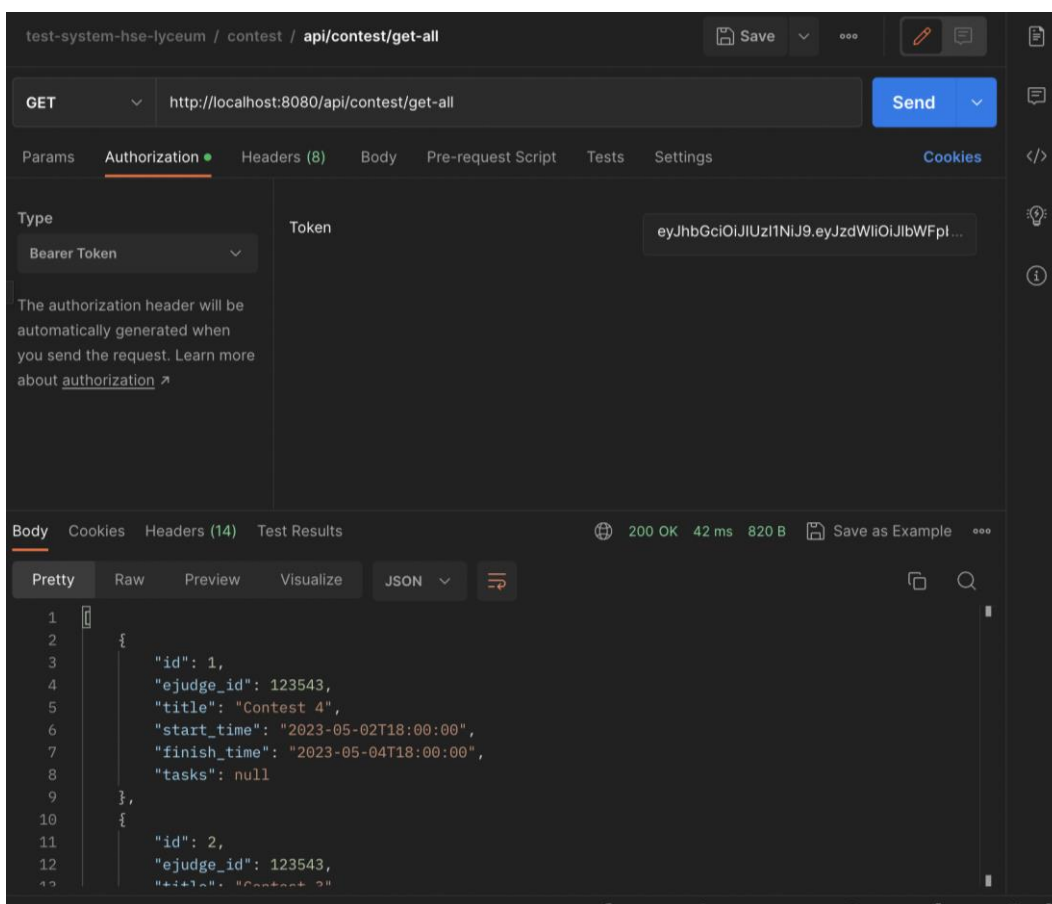
"title": "Contest 3",
"start_time": "2023-05-02T18:00:00",
"finish_time": "2023-05-04T18:00:00",
"tasks": [
{
  "title": "A",
  "description": "this is the first task",
  "memory_limit": 32.0,
  "time_limit": 2.0,
  "attempts_amount": 100,
  "classification_title": "type_1"
},
{
  "title": "B",
  "description": "this is the second task",
  "memory_limit": 64.0,
  "time_limit": 2.0,
  "attempts_amount": 100,
  "classification_title": "type_2"
},
{
  "title": "C",
  "description": "this is the third task",
  "memory_limit": 32.0,
  "time_limit": 1.0,
  "attempts_amount": 100,
  "classification_title": "type_1"
}
]
}
]

```

Пример ответа в случае возникновения ошибки (статус код 400):

Contests are not found

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/contest/get-all' \
--header 'Authorization: Bearer user_token'
```

### 3.2.15. Работа с функцией «Получение информации о всех посылках всех пользователей системы»

Триггер: необходимо проверить работоспособность запроса на получение информации о всех посылках, имеющихся в системе, который поступает со стороны клиента

Название папки коллекции в Postman: solutions

Тип запроса: GET

Endpoint: `http://{ip}/api/solutions/get-all`

Авторизация: Bearer Token

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

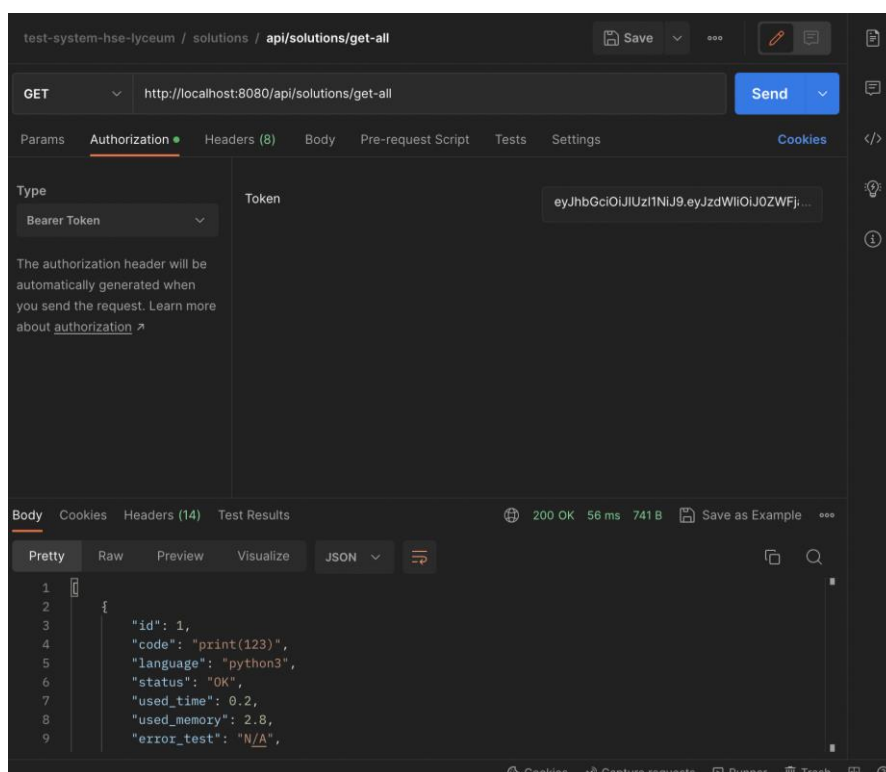
```
[
  {
    "id": 1,
    "code": "print(123)",
    "language": "python3",
    "status": "OK",
    "used_time": 0.2,
    "used_memory": 2.8,
    "error_test": "N/A",
    "contest_name": «Contest 3»,
    "task_name": "A"
  },
  {
    "id": 2,
    "code": "print(1234)",
    "language": "python3",
    "status": "OK",
    "used_time": 0.2,
    "used_memory": 2.8,
    "error_test": "N/A",
    "contest_name": «Contest 1»,
    "task_name": "B"
  }
]
```

Пример ответа в случае возникновения ошибки (статус код 400):

User with email: <email@edu.hse.ru> is not a teacher and has role: <student>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и





Подпись

Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/solutions/get-all' \
--header 'Authorization: Bearer user_token'
```

### 3.2.16. Работа с функцией «Получение информации о роли пользователя системы»

Триггер: необходимо проверить работоспособность запроса на получение информации о роли пользователя в системе, который поступает со стороны клиента

Название папки коллекции в Postman: profile

Тип запроса: GET

Endpoint: http://{ip}/api/profile/get-role

Авторизация: Bearer Token

Параметры: нет

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

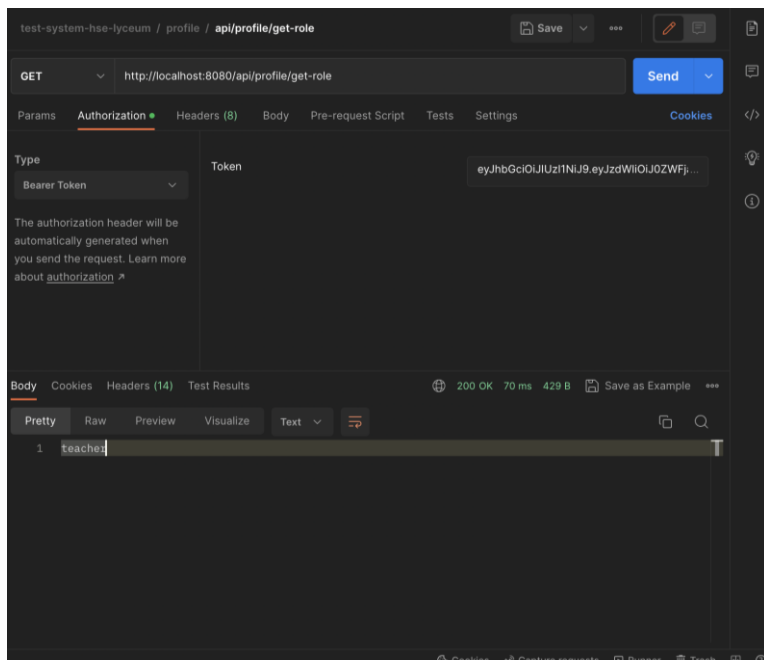
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

teacher

Пример ответа в случае возникновения ошибки (статус код 400):

Role of user with email: <email> was not found

Скриншот с коллекции «test-system-hse-lyceum» в Postman:



Подпись

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/profile/get-role' \
--header 'Authorization: Bearer user_token'
```

### 3.2.17. Работа с функцией «Получения информации о конкурсе по его id в системе»

Триггер: необходимо проверить работоспособность запроса на получение информации об определенном конкурсе в системе, который поступает со стороны клиента

Название папки коллекции в Postman: contest

Тип запроса: GET

Endpoint: `http://{ip}api/contest/info?id=number`

Авторизация: Bearer Token

Параметры: id

Тело запроса: нет

Пример ответа в случае успешного выполнения запроса (статус код 200):

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 34 01–1**

```

{
  "id": 1,
  "ejudge_id": 123543,
  "title": "Contest 4",
  "start_time": "2023-05-02T18:00:00",
  "finish_time": "2023-05-04T18:00:00",
  "tasks": [
    {
      "title": "A",
      "description": "this is the first task",
      "memory_limit": 32.0,
      "time_limit": 2.0,
      "attempts_amount": 100,
      "classification_title": "type_1"
    },
    {
      "title": "B",
      "description": "this is the second task",
      "memory_limit": 64.0,
      "time_limit": 2.0,
      "attempts_amount": 100,
      "classification_title": "type_2"
    },
    {
      "title": "C",
      "description": "this is the third task",
      "memory_limit": 32.0,
      "time_limit": 1.0,
      "attempts_amount": 100,
      "classification_title": "type_1"
    }
  ]
}

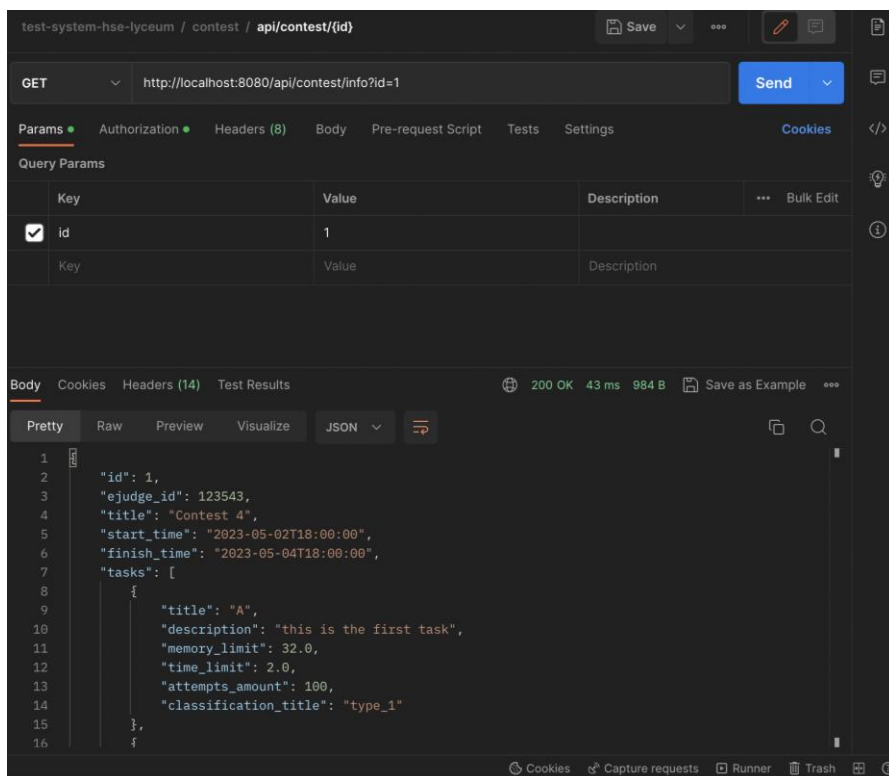
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 34 01–1

Пример ответа в случае возникновения ошибки (статус код 400):

The contest with such id does not exist



Скриншот с коллекции «test-system-hse-lyceum» в Postman:

cURL code snippet для выполнения запроса с сервера:

```
curl --location 'http://{ip}/api/contest/info?id=1' \
--header 'Authorization: Bearer user_token'
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

#### 4. СООБЩЕНИЯ ОПЕРАТОРУ

При некорректных действиях оператора программой предусмотрен вывод предупреждающих сообщений. Во всех случаях программа работает корректно и не завершается аварийно.

Предусмотрены следующие сообщения оператору:

1. Вывод сообщения о некорректной регистрации нового пользователя в систему
2. Вывод сообщения о некорректном получении информации о профиле учащегося
3. Вывод сообщения о некорректном получении информации о профиле учителя
4. Вывод сообщения о некорректной авторизации пользователя в систему
5. Вывод сообщения о некорректном добавлении нового преподавателя в систему с помощью администраторских прав
6. Вывод сообщения о некорректном получении информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания
7. Вывод сообщения о некорректном получении информации о рейтинге, отсортированном по количеству задач, пользователей в порядке убывания
8. Вывод сообщения об ошибке в процессе скачивания информации о рейтинге, отсортированном по оценкам, пользователей в порядке убывания в формате .csv
9. Вывод сообщения об ошибке в процессе скачивания информации о рейтинге, отсортированном по количеству решенных задач, пользователей в порядке убывания в формате .csv
10. Вывод сообщения о некорректном получении информации о всех посылках (решениях) определённого пользователя
11. Вывод сообщения о некорректном добавлении решения определенной задачи определённого контекста в систему
12. Вывод сообщения о некорректном добавлении нового контекста с задачами преподавателем в систему
13. Вывод сообщения о некорректном получении информации о всех контекстах, в которых пользователь принимает участие
14. Вывод сообщения о некорректном получении информации о всех контекстах, имеющихся в системе
15. Вывод сообщения о некорректном получении информации о всех посылках всех пользователей системы
16. Вывод сообщения о некорректном получении информации о роли пользователя системы
17. Вывод сообщения о некорректном получении информации о контексте по его id в системе

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и