

**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

*Куренков*

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**

**Программа и методика испытаний**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.05.10-01 51 01-1-ЛУ**

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

Исполнитель

студент группы БПИ219

*Е.Б. Старцев* / Е.Б. Старцев /  
«13» \_\_\_\_\_ мая \_\_\_\_\_ 2023 г.

студент группы БПИ219

*С.О. Котовский* / С.О. Котовский /  
«13» \_\_\_\_\_ мая \_\_\_\_\_ 2023 г.

студент группы БПИ219

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

**Москва 2023**

УТВЕРЖДЕН  
RU.17701729.05.10-01 51 01-1-ЛУ

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

**Программа и методика испытаний**

**RU.17701729.05.10-01 51 01-1**

**Листов 40**

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

**Москва 2023**

## ОГЛАВЛЕНИЕ

<b>1.</b>	<b>ОБЪЕКТ ИСПЫТАНИЙ.....</b>	<b>4</b>
1.1.	Наименование программы и объекта, в котором программа используется.....	4
1.2.	Краткая характеристика области применения программы и объекта, в котором она используется .....	4
<b>2.</b>	<b>ЦЕЛЬ ИСПЫТАНИЙ .....</b>	<b>5</b>
<b>3.</b>	<b>ТРЕБОВАНИЯ К ПРОГРАММЕ.....</b>	<b>6</b>
3.1.	Требования к функциональным характеристикам .....	6
3.1.1.	Требования к составу выполняемых функций серверной частью.....	7
3.1.2.	Требования к серверной части .....	11
3.1.3.	Требования к взаимодействию клиентской и серверной частей.....	11
3.1.4.	Организация входных данных.....	12
3.1.5.	Организация выходных данных.....	13
3.2.	Требования к надёжности .....	13
3.2.1.	Обеспечение устойчивого функционирования.....	13
3.2.2.	Контроль входной информации .....	13
3.2.3.	Контроль выходной информации .....	13
3.2.4.	Время восстановления после отказа.....	13
<b>4.</b>	<b>ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ.....</b>	<b>14</b>
4.1.	Состав программной документации .....	14
4.2.	Специальные требования к программной документации.....	14
<b>5.</b>	<b>СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ .....</b>	<b>14</b>
5.1.	Технические средства, используемые во время испытаний .....	14
5.2.	Программные средства, используемые во время испытаний.....	15
5.3.	Порядок проведения испытаний .....	15
<b>6.</b>	<b>МЕТОДЫ ИСПЫТАНИЙ.....</b>	<b>16</b>
6.1.	Испытание выполнения требований к программной документации .....	16

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

# **RU.17701729.05.10-01 51 01–1**

<b>6.2.</b>	<b>Испытание выполнения требований к функциональным характеристикам</b>	<b>16</b>
6.2.1.	Испытание функции «Регистрация нового преподавателя» .....	16
6.2.2.	Испытание функции «Авторизация и идентификация пользователя» .....	19
6.2.3.	Испытание функции «Добавление решения задачи в тестирующую систему»	20
6.2.4.	Испытание функции «Получение рейтинга по оценкам учащихся» или «Получение рейтинга по количеству задач учащихся» (работает аналогично) .....	23
6.2.5.	Испытание функции «Получение посылок отправленных пользователем решений»	24
6.2.6.	Испытание функции «Получение списка всех конкурсов, в которых пользователь принимает участие» .....	25
6.2.7.	Испытание функции «Получение информации о конкурсе и его задачах».	26
6.2.8.	Испытание функции «Получение информации о всех конкурсах и их задачах, которые имеются в системе» .....	28
6.2.9.	Испытание функции «Получение персональной информации в личном кабинете (информации о профиле) – это запросы с эндпоинтами /get-student-info, /get-teacher-info (работает аналогично) .....	30
6.2.10.	Испытание функции «Получение информации о роли в системе» .....	31
6.2.11.	Испытание функции «Скачивание рейтинга по оценкам в формате .csv» или «Скачивание рейтинга по количеству задач в формате .csv» (работает аналогично)	32
6.2.12.	Испытание функции «Создание аккаунта для учащегося» .....	34
6.2.13.	Испытание функции «Получение посылок всех пользователей всех решений конкурса» .....	35
6.2.14.	Испытание функции «Добавление нового конкурса с задачами данного конкурса в систему» .....	36
<b>6.3.</b>	<b>Испытание выполнения требований к надёжности .....</b>	<b>39</b>
	<b>ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....</b>	<b>40</b>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## 1. ОБЪЕКТ ИСПЫТАНИЙ

### 1.1. Наименование программы и объекта, в котором программа используется

Наименование темы разработки: «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга».

Наименование темы разработки на английском языке: «HSE Lyceum Programming Testing System with Automated Rating Estimation».

Наименование объекта, в котором программа используется: Лицей НИУ ВШЭ.

### 1.2. Краткая характеристика области применения программы и объекта, в котором она используется

«Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга» - набор программных инструментов для оценивания решений задач на разных языках программирования для учащихся Лицея НИУ ВШЭ с рейтингом для каждого учащегося.

Лицей НИУ ВШЭ – общеобразовательное учреждение города Москва.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## 2. ЦЕЛЬ ИСПЫТАНИЙ

Цель проведения испытаний – проверить разработанные программы на соответствие функциональным требованиям и отдельным требованиям к надежности, изложенных в документе «Техническое задание» к каждой из программ. Основные испытания проводятся на API с эндпоинтами.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

### 3. ТРЕБОВАНИЯ К ПРОГРАММЕ

Все программы должны соответствовать функциональным требованиям, указанным в документе «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга. Техническое задание». Допускается, что в программах, содержащих мобильное приложение и веб-сайт, не будет реализован весь функционал в связи с особенностями каждой из программ. Именно по этой причине далее в данном документе будут рассматриваться требования и испытания непосредственно для программы, содержащей в себе API с эндпоинтами.

#### 3.1. Требования к функциональным характеристикам

Программа должна позволять осуществлять регистрацию и авторизацию в систему для всех пользователей с помощью заранее созданных для аккаунта логина и пароля, различать пользователей, т.е. идентифицировать преподавателя и ученика.

При этом для авторизованного преподавателя системой будут предоставлены возможности:

1. Создание аккаунта учащегося
2. Получать информацию о своем профиле
3. Добавлять решение задачи в тестирующую систему
4. Скачивать рейтинг по оценкам в формате .csv
5. Скачивать рейтинг по количеству задач в формате .csv
6. Получать информацию о рейтинге учеников по оценкам
7. Получать информацию о рейтинге учеников по количеству задач
8. Получать информацию о своих посылках, то есть отправленных решениях
9. Получать информацию о посылках всех участников системы, то есть получать информацию о всех имеющихся в системе решениях
10. Добавлять контекст с задачами данного контекста в систему
11. Получать информацию о всех контекстах, в которых он принимает участие
12. Получать информацию о всех контекстах и их задачах, которые есть в системе
13. Получать информацию о контексте и его задачах в системе
14. Получать информацию о своей роли в системе

Для авторизованного ученика системой будут предоставлены возможности:

1. Получать информацию о своем профиле
2. Добавлять решение задачи в тестирующую систему
3. Получать информацию о своих посылках, то есть отправленных решениях
4. Получать информацию о контекстах, в которых он участвует

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

5. Получать информацию о контексте и его задачах в системе
6. Получать информацию о всех контекстах и их задачах, которые есть в системе
7. Получать информацию о рейтинге всех учеников по оценкам
8. Получать информацию о рейтинге всех учеников по количеству задач
9. Получать информацию о своей роли в системе

**3.1.1. Требования к составу выполняемых функций серверной частью****Для администратора системы:***Процесс регистрации нового преподавателя:*

1. Получение запроса с Basic Auth с заранее выданными логином и паролем для регистрации нового преподавателя в систему с информацией о данном преподавателе.
2. Добавление всей информации о новом преподавателе в базу данных в различные таблицы.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции регистрации нового преподавателя в систему.

**Для пользователя, являющегося преподавателем или учащимся:***Процесс авторизации и идентификации пользователя:*

1. Получение логина и пароля с клиентской части.
2. Проверка наличия логина, т.е. наличия пользователя с указанным логином в базе данных пользователей среди прочих аккаунтов.
3. Проверка совпадения отправленного пароля с паролем, находящимся в базе данных для найденного пользователя.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции авторизации (при успешной авторизации сообщается тип пользователя - учащийся или преподаватель, а также высылается уникальный токен пользователя).

*Процесс добавления решения задачи в тестирующую систему:*

1. Получение информации с клиентской части с решением пользователя и дополнительной информацией о решении, указанной в теле запроса.
2. Добавление всей информации о решении в базу данных в различные таблицы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



**RU.17701729.05.10-01 51 01–1**

3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции добавления решения с соответствующим результатом и необходимой дополнительной информацией прохождения данного решения в систему.

*Процесс получения рейтинга определённого типа (рейтинга по оценкам или рейтинга по количеству задач) учащихся:*

1. Получение с клиентской части запроса о типе рейтинга, который необходимо отобразить на клиентской части.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Подсчёт рейтинга.
4. Формирование необходимого ответа на слое бизнес-логики.
5. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции со списком рейтинга учащихся определённого типа.

*Процесс получения посылок отправленных пользователем решений:*

1. Получение с клиентской части запроса об отправке списка посылок отправленных пользователем решений.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции со списком посылок отправленных пользователем решений.

*Процесс получения списка всех контестов, в которых пользователь принимает участие:*

1. Получение с клиентской части запроса об отправке списка всех контестов, в которых пользователь принимает участие.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции со списком контекстов, в которых пользователь принимает участие.

*Процесс получения информации о контексте и его задачах:*

1. Получение с клиентской части запроса об отправке информации о контексте и его задачах, с которым пользователь хочет взаимодействовать.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции с информацией о контексте и его задачах, с которым пользователь хочет взаимодействовать.

*Процесс получения информации о всех контекстах и их задачах, которые имеются в системе:*

1. Получение с клиентской части запроса об отправке информации о всех контекстах и их задачах, которые имеются в системе.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции с информацией о всех контекстах и их задачах, которые имеются в системе.

*Процесс получения персональной информации в личном кабинете (информации о профиле):*

1. Получение с клиентской части запроса об отправке информации, которая должна содержаться в личном кабинете пользователя.
2. Извлечение необходимых данных из различных таблиц базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции с информацией о пользователе, необходимой для отображения в личном кабинете.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

*Процесс получения информации о роли в системе:*

1. Получение с клиентской части запроса об отправке информации о роли, которая присуща данному пользователю.
2. Извлечение необходимых данных из базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции с информацией о роли, которая присуща данному пользователю.

**Для пользователя, являющегося преподавателем:**

*Процесс скачивания рейтинга определенного типа (рейтинга по оценкам или рейтинга по количеству задач) в формате .csv:*

1. Получение информации с клиентской части о типе рейтинга для скачивания данных в формате .csv.
2. Извлечение необходимых данных из базы данных.
3. Подсчёт рейтинга.
4. Формирование необходимого ответа на слое бизнес-логики.
5. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции (при успешном выполнении операции скачивается файл, содержащий рейтинг необходимого типа в формате .csv).

*Процесс создания аккаунта для учащегося:*

1. Получение с клиентской части запроса создания аккаунта нового учащегося с необходимой информацией о новом учащемся.
2. Добавление нового учащегося в разные таблицы базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции создания аккаунта для учащегося с необходимой информацией об этом учащемся.

*Процесс получения посылок всех пользователей всех решений конкурса:*

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

1. Получение с клиентской части запроса об отправке списка посылок всех отправленных решений контеста всеми пользователями.
2. Извлечение необходимых данных из базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции со списком посылок всех отправленных решений контеста всеми пользователями системы.

*Процесс добавления нового контеста с задачами данного контеста в систему:*

1. Получение с клиентской части запроса о добавлении нового контеста в систему с полной информацией об этом контесте, включая задачи данного контеста.
2. Добавление информации о новом контесте и задач данного контеста в соответствующие таблицы базы данных.
3. Формирование необходимого ответа на слое бизнес-логики.
4. Ответ на клиентскую часть, с которой пришел запрос, содержащий уведомление об ошибке или успешной операции добавления нового контеста с задачами данного контеста в систему.

**3.1.2. Требования к серверной части**

Серверная часть должна быть представлена в виде REST API с монолитной архитектурой, принимающей запросы с клиентской части и отправляющей ответы об ошибке / успешной операции на клиентскую часть.

**3.1.3. Требования к взаимодействию клиентской и серверной частей**

Взаимодействие клиентской и серверной частей будет организовано по протоколу HTTP с использованием GET / POST запросов.

С клиентской части принимаются и обрабатываются следующие запросы:

1. POST request: `http://{ip}/api/auth/register` - регистрация нового ученика в систему
2. POST request: `http://{ip}/api/auth/login` - авторизация пользователя в систему
3. GET request: `http://{ip}/api/profile/get-student-info` - получение информации о профиле учащегося
4. GET request: `http://{ip}/api/profile/get-teacher-info` - получение информации о профиле учителя

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

5. GET request: `http://{ip}/api/profile/get-role` - получение информации о роли пользователя системы

6. GET `http://{ip}/api/admin/add-teacher` - добавление нового преподавателя в систему с помощью администраторских прав

7. GET `http://{ip}/api/grade/by-tasks-amount` - получение информации о рейтинге, отсортированном по количеству задач пользователей в порядке убывания

8. GET `http://{ip}/api/grade/by-marks` - получение информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания

9. GET `http://{ip}/api/grade/by-marks/download-csv` - скачивание информации о рейтинге, отсортированном по оценкам пользователей в порядке убывания в формате .csv

10. GET `http://{ip}/api/grade/by-tasks-amount/download-csv` - скачивание информации о рейтинге, отсортированном по количеству решенных задач пользователей в порядке убывания в формате .csv

11. GET `http://{ip}/api/solutions/get-all-user` - получение информации о всех посылках (т. е. решениях) определённого пользователя

12. POST `http://{ip}/api/solutions/add` - добавление решения определенной задачи определённого контекста в систему

13. GET `http://{ip}/api/solutions/get-all` - получение информации о всех посылках всех пользователей системы

14. POST `http://{ip}/api/contest/add` - добавление нового контекста с задачами преподавателем в систему

15. GET `http://{ip}/api/contest/get-all-user` - получение информации о всех контекстах, в которых пользователь принимает участие

16. GET `http://{ip}/api/contest/get-all` - получение информации о всех контекстах, имеющихся в системе

17. GET `http://{ip}/api/contest/info?id={parameter}` - получение информации о контексте по его id в системе

Все запросы отправляются с данными, которые позволяют идентифицировать пользователя, чтобы веб и мобильные приложения получали данные, актуальные для авторизованного в системе пользователя.

**3.1.4. Организация входных данных**

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

Организация входных данных проекта реализована следующим образом: на endpoint-ы API со стороны клиента поступают запросы с заголовками и телом запроса (необязательно) в формате json.

**3.1.5. Организация выходных данных**

Организация выходных данных проекта реализована следующим образом: на поступившие http запросы, приходящие со стороны клиента, отправляются ответы, которые содержат всю необходимую информацию, предоставляемую в теле ответа (необязательно) в формате json или строки.

**3.2. Требования к надёжности****3.2.1. Обеспечение устойчивого функционирования**

Система должна обеспечивать непрерывную работу в течение всего периода проведения различных соревнований, не допускать падения или зависания во время работы с ней.

**3.2.2. Контроль входной информации**

Система должна проверять входные данные на соответствие бизнес-требованиям и уведомлять клиентскую часть об обнаруженных ошибках.

**3.2.3. Контроль выходной информации**

Система должна генерировать корректные ответы, проверять их и уведомлять клиентскую часть об ошибках, если необходимо.

**3.2.4. Время восстановления после отказа**

Система должна иметь возможность восстановления в случае сбоев или отказов в работе в кратчайшие сроки.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## 4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

### 4.1. Состав программной документации

«Реализация «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга». Техническое задание (ГОСТ 19.201-78);

«Реализация «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга». Программа и методика испытаний (ГОСТ 19.301-78);

«Реализация «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга». Пояснительная записка (ГОСТ 19.404-79);

«Реализация «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга». Руководство оператора (ГОСТ 19.505-79);

«Реализация «Система проверки решений по программированию для Лицея НИУ ВШЭ с автоматизированным подсчётом рейтинга». Текст программы (ГОСТ 19.401-78);

### 4.2. Специальные требования к программной документации

Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа (см. п. 5.1.).

Пояснительная записка должна быть загружена в систему Антиплагиат через LMS «НИУ ВШЭ».

Техническое задание и пояснительная записка, титульные листы других документов должны быть подписаны руководителем разработки и исполнителем.

Документация и программа сдаётся в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar.

За три дня до защиты комиссии все материалы курсового проекта: программная документация, программный проект, исполняемый файл, отзыв руководителя, отчёт системы Антиплагиат должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект» в личном кабинете в информационной образовательной среде SmartLMS НИУ ВШЭ.

## 5. СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ

### 5.1. Технические средства, используемые во время испытаний

Для испытаний программы используется сервер, оснащенный следующими техническими характеристиками:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01-1**

description: CPU

product: Intel(R) Xeon(R) CPU E5-2697 v3 @ 2.60GHz

vendor: Intel Corp.

physical id: 400

bus info: cpu@0

version: 6.63.2

slot: CPU 0

size: 2GHz

capacity: 2GHz

width: 64 bits

memory:

description: System Memory

physical id: 1000

size: 2GiB

capacity: 2GiB

capabilities: ecc

configuration: errordetection=multi-bit-ecc

**5.2. Программные средства, используемые во время испытаний**

Для испытаний программы используются следующие программные средства:

1. Операционная система: Ubuntu 22.04.1 LTS.
2. Java Development Kit (JDK) версии 17.
3. Spring Boot 3.0.2.
4. Java 17.
5. База данных PostgreSQL версии 11.

**5.3. Порядок проведения испытаний**

Испытания должны проводиться в следующем порядке:

1. Проверка требований к программной документации.
2. Проверка требований к функциональным характеристикам.
3. Проверка требований к надежности.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



## **6. МЕТОДЫ ИСПЫТАНИЙ**

### **6.1. Испытание выполнения требований к программной документации**

Выполнения требований к составу программной документации проверяется наличием всех подписей и программной документации в системе SmartLMS. Также проверяется соответствие документации требованиям ГОСТ.

Все документы удовлетворяют представленным требованиям.

### **6.2. Испытание выполнения требований к функциональным характеристикам**

#### **6.2.1. Испытание функции «Регистрация нового преподавателя»**

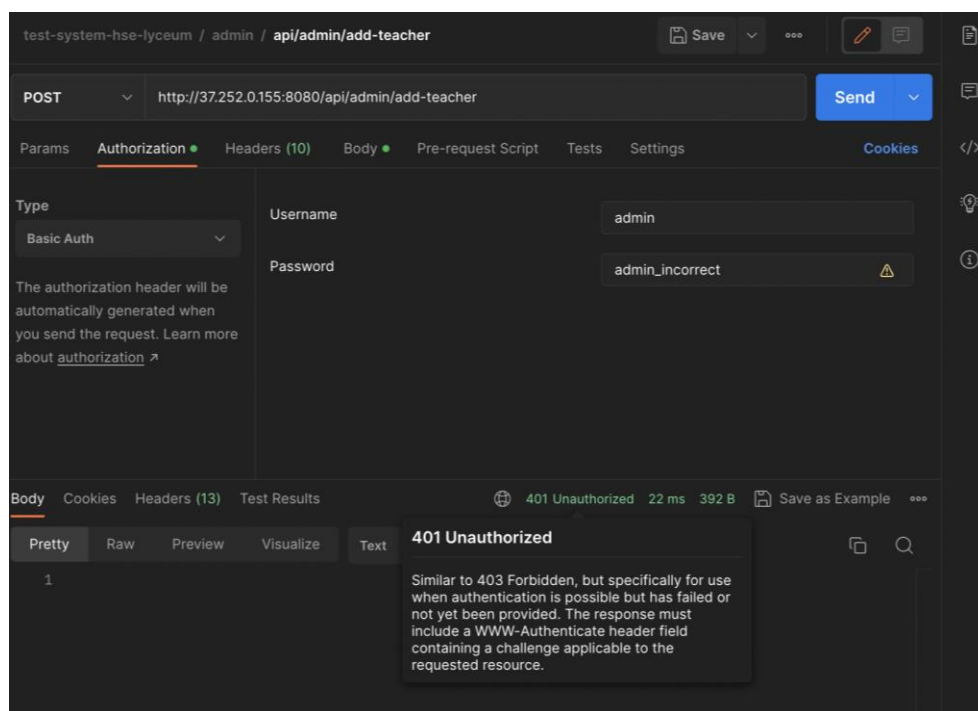
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**RU.17701729.05.10-01 51 01–1**

Для регистрации нового преподавателя в систему используется Basic Auth с заранее выданными администратору логином и паролем.

При некорректном использовании логина или пароля при выполнении запроса на сервер для добавления нового преподавателя будет получен ответ со статус кодом 401 (Unauthorized).

Скриншот из коллекции Postman:

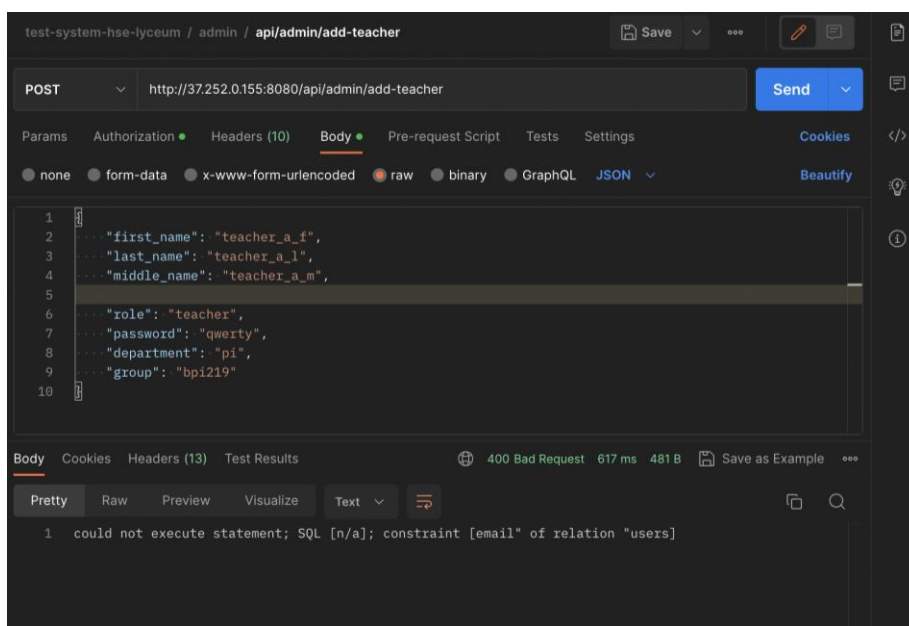


При корректных логине и пароле, но некорректном теле запроса, например, при отсутствии одного из обязательных параметров будет получен ответ, содержащий сообщение об ошибке о

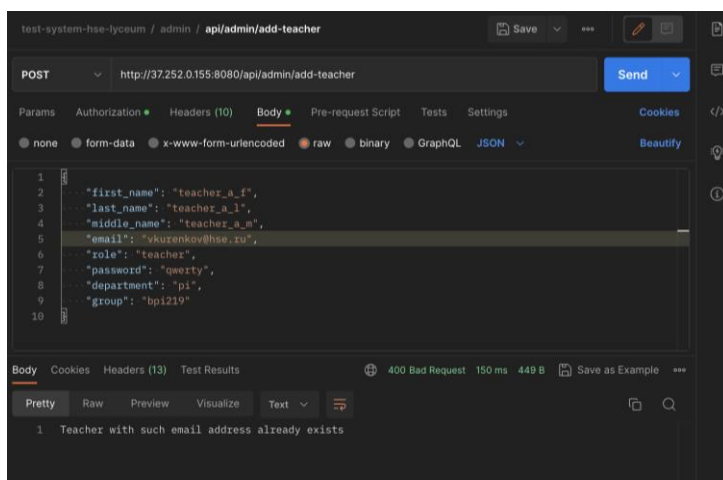
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

невозможности выполнить добавление преподавателя в систему при отсутствии данного параметра. Скриншот из коллекции Postman:



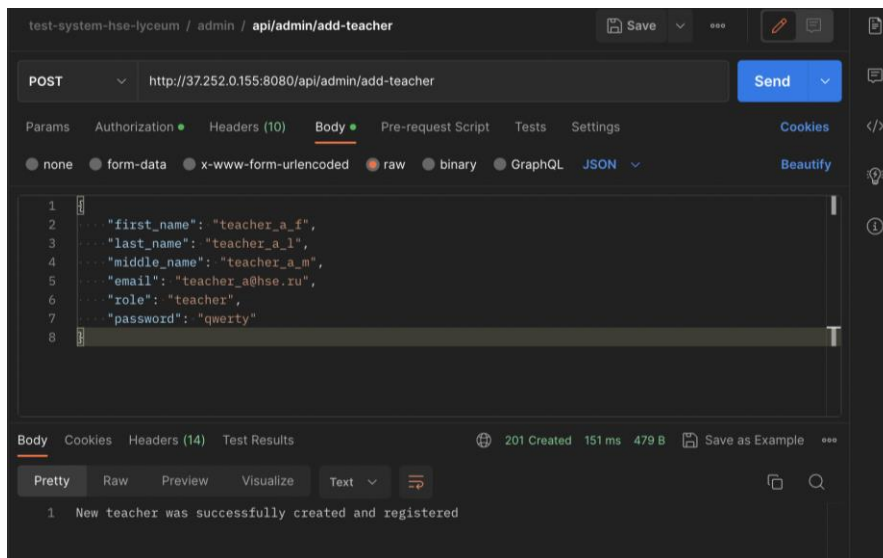
При корректных логине и пароле, но некорректном теле запроса, например, при попытке добавить преподавателя с уже существующим email, будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить добавление преподавателя в систему, так как



преподаватель с таким email уже зарегистрирован в системе. Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

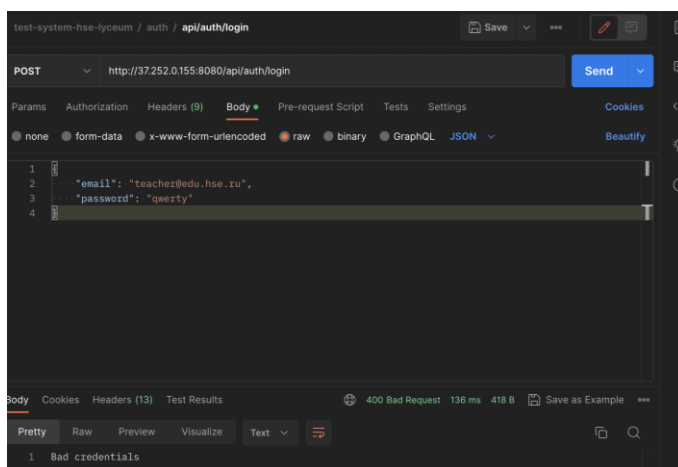
При корректных логине и пароле, а также верном теле запроса с информацией о новом преподавателе будет получен ответ об успешном создании и добавлении нового преподавателя в систему. Скриншот из коллекции Postman:



### 6.2.2. Испытание функции «Авторизация и идентификация пользователя»

Для авторизации и идентификации пользователя в систему не требуется использование аутентификации, происходит сравнение введенных пользователем email и пароля.

При некорректно введенном email или пароле будет получен ответ, содержащий информацию о неверных данных, поступивших в теле запроса во время авторизации в систему.

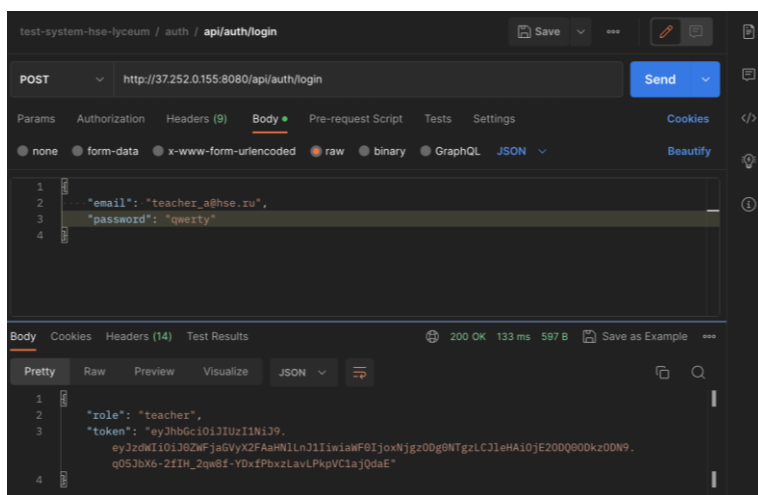


Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

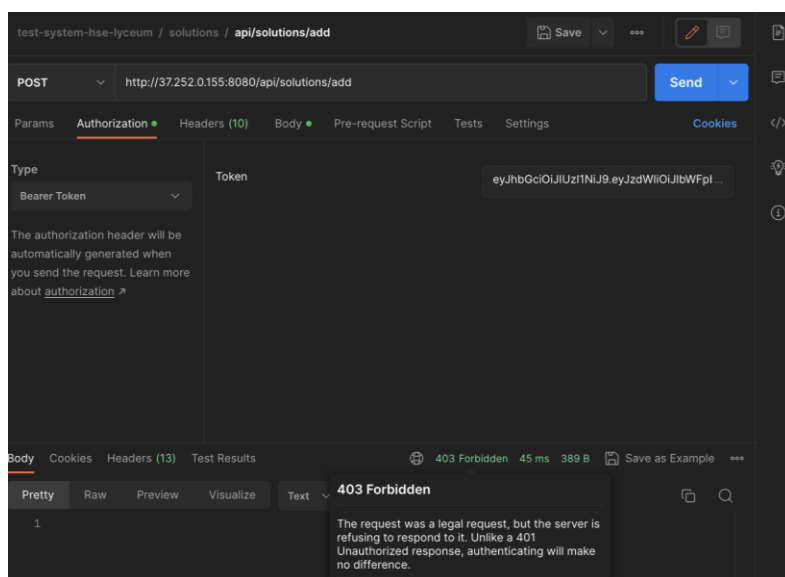
При корректно введенных email и пароле будет получен ответ об успешной авторизации пользователя в систему, а также будет возвращена роль пользователя в системе и сгенерированный JWT-токен для работы пользователя с системой. Скриншот из коллекции Postman:



### 6.2.3. Испытание функции «Добавление решения задачи в тестирующую систему»

Для добавления решения задачи в тестирующую систему требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

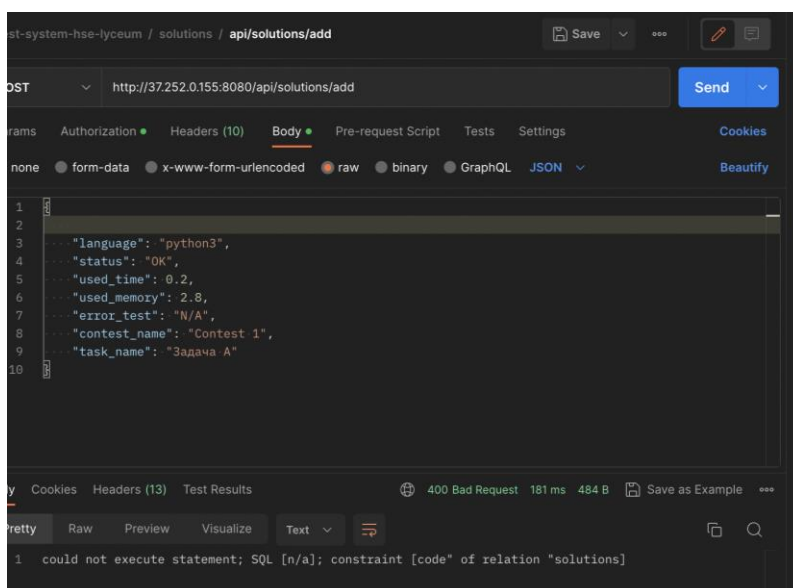


При верной аутентификации, но отсутствии какого-либо из обязательных полей в теле запроса будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить

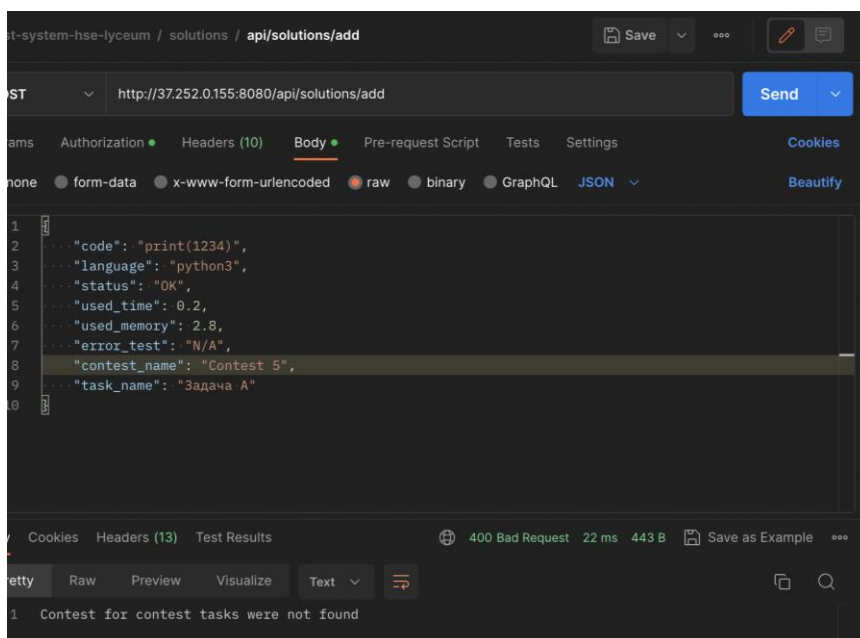
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

добавление решения задачи в тестирующую систему при отсутствии данного параметра.

Скриншот из коллекции Postman:



При верной аутентификации, но получении неверного названия конкурса (то есть получении решения для несуществующего конкурса) будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить добавление решения задачи в тестирующую систему, так как указанный конкурс в запросе не существует. Скриншот из коллекции Postman:

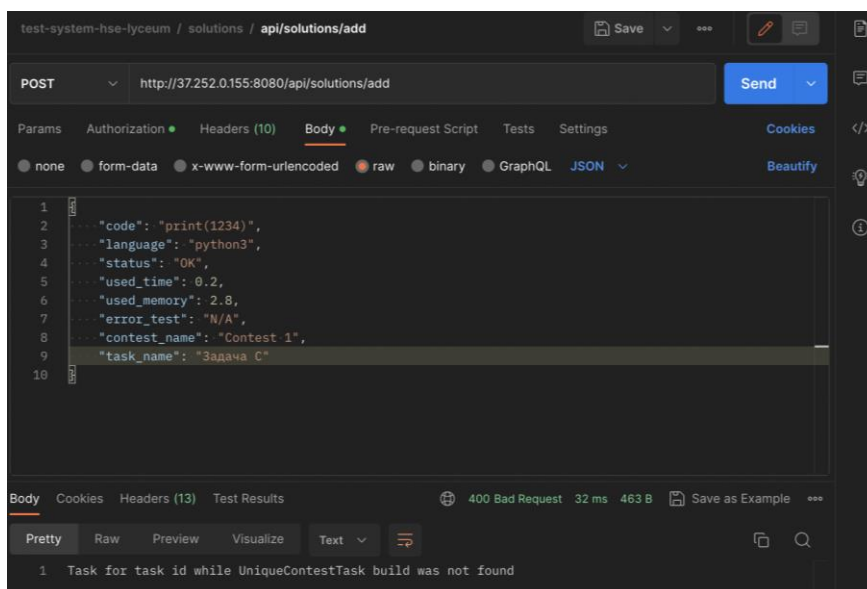


Подпись

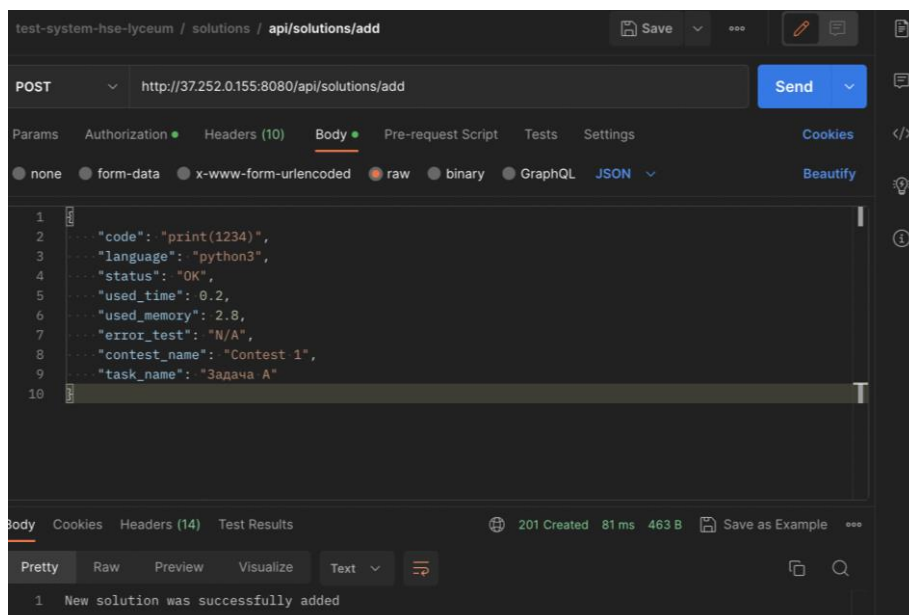
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

При верной аутентификации, но получении неверного названия задачи (то есть получении решения для несуществующей задачи) будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить добавление решения задачи в тестирующую систему, так как указанная задача в запросе не существует. Скриншот из коллекции Postman:



При верной аутентификации и верном теле запроса, удовлетворяющем всем условиям, будет получен ответ об успешном добавлении нового решения пользователя в систему.



Подпись

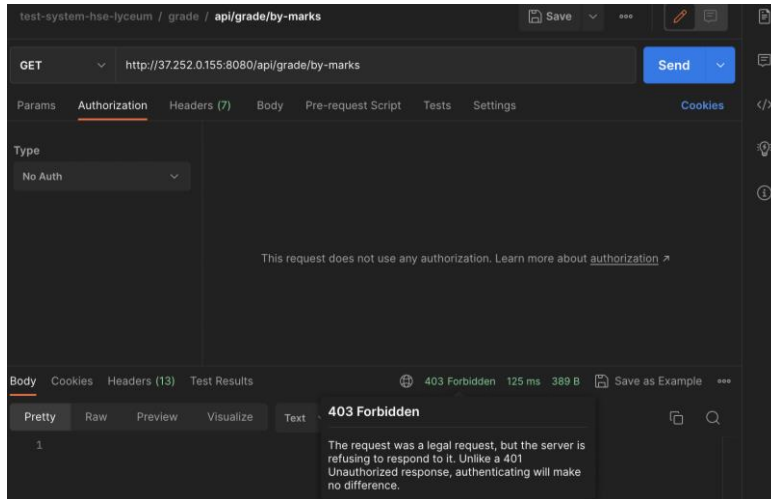
Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

#### 6.2.4. Испытание функции «Получение рейтинга по оценкам учащихся» или «Получение рейтинга по количеству задач учащихся» (работает аналогично)

Для получения рейтинга по оценкам учащихся в системе требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет



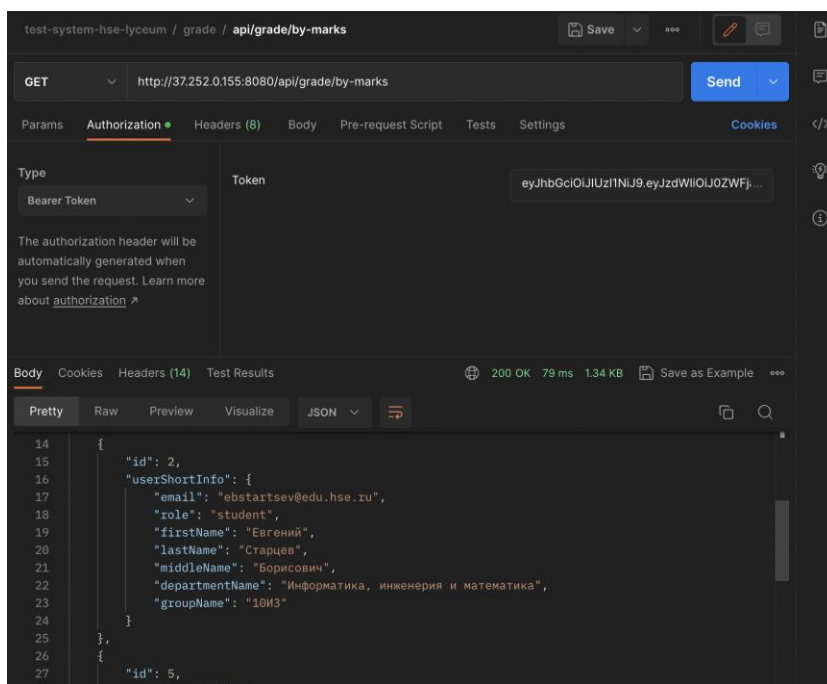
получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



## RU.17701729.05.10-01 51 01–1

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий отсортированный текущий рейтинг определенного типа пользователей. Скриншот



Подпись

из коллекции Postman:

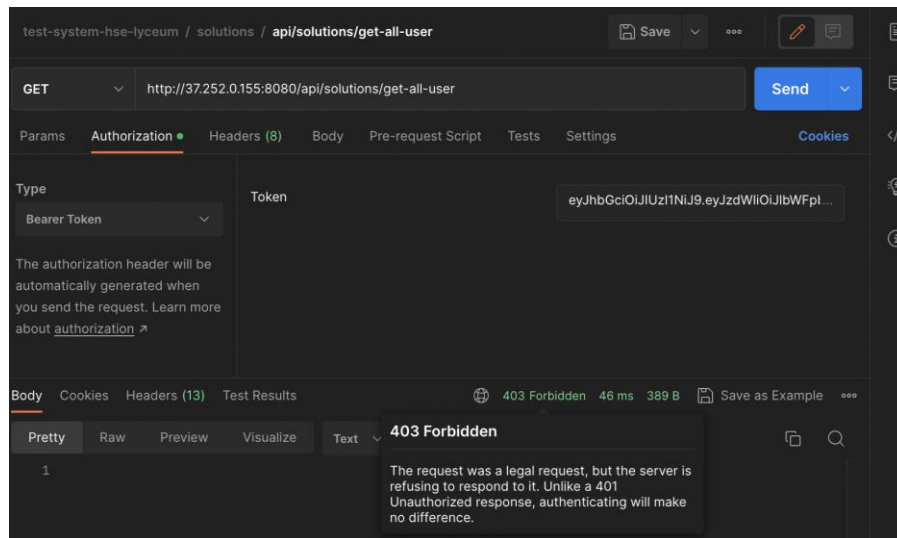
### 6.2.5. Испытание функции «Получение посылок отправленных пользователем решений»

Для получения посылок отправленных пользователем решений в тестирующую систему требуется использование аутентификации с помощью JWT-токена.

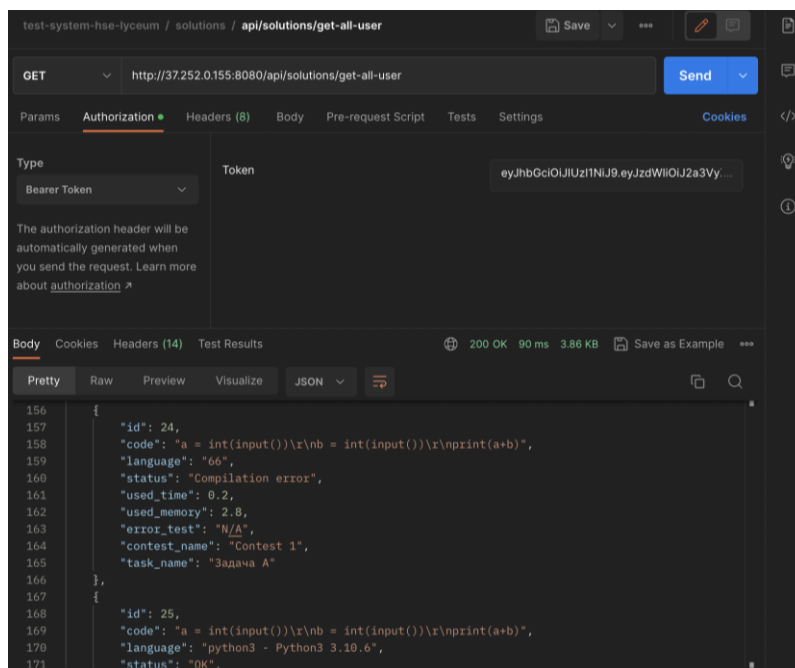
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:



При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий все послышки данного пользователя. Скриншот из коллекции Postman:



Подпись

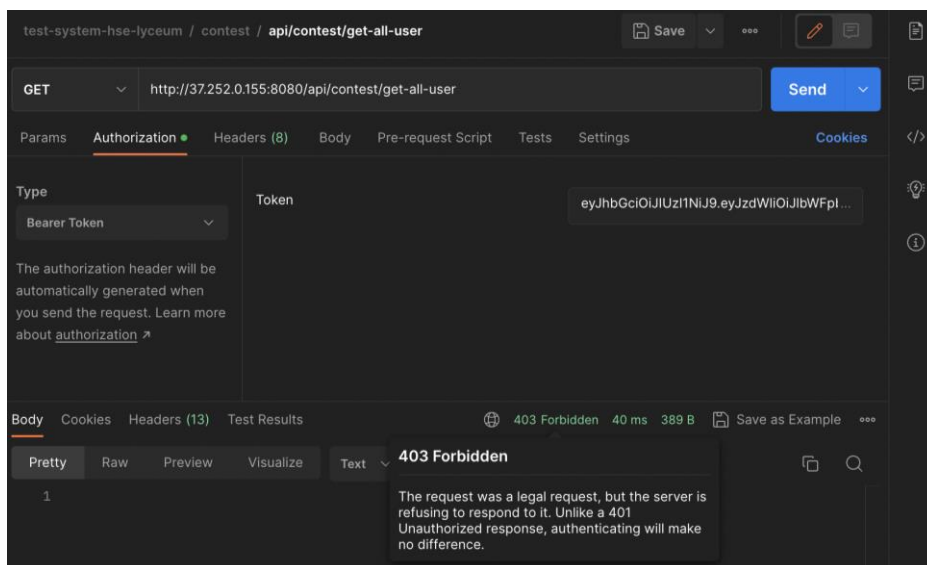
### 6.2.6. Испытание функции «Получение списка всех контестов, в которых пользователь принимает участие»

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

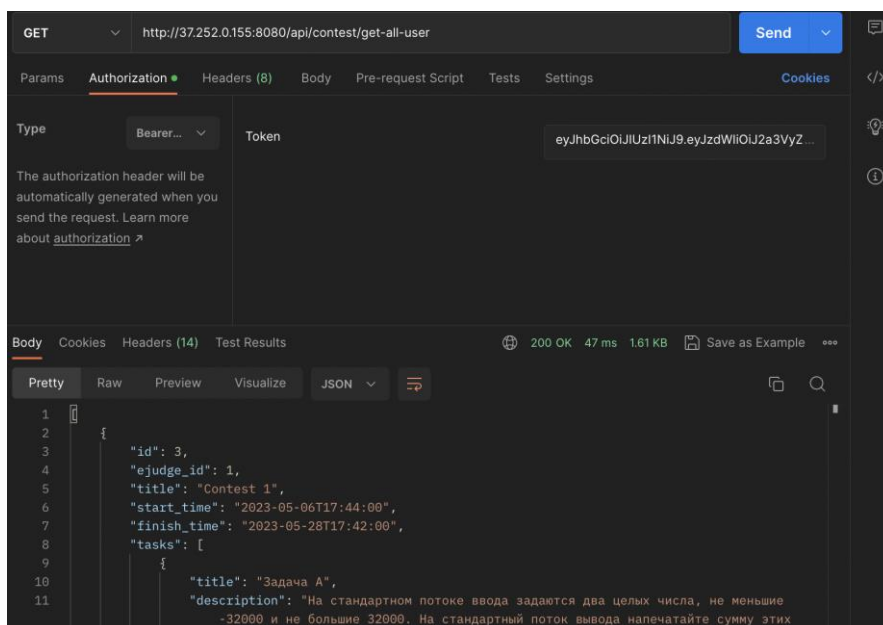
## RU.17701729.05.10-01 51 01–1

Для получения списка всех контестов, в которых пользователь принимает участие, требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:



При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий список всех контестов, в которых пользователь принимает участие. Скриншот из



коллекции Postman:

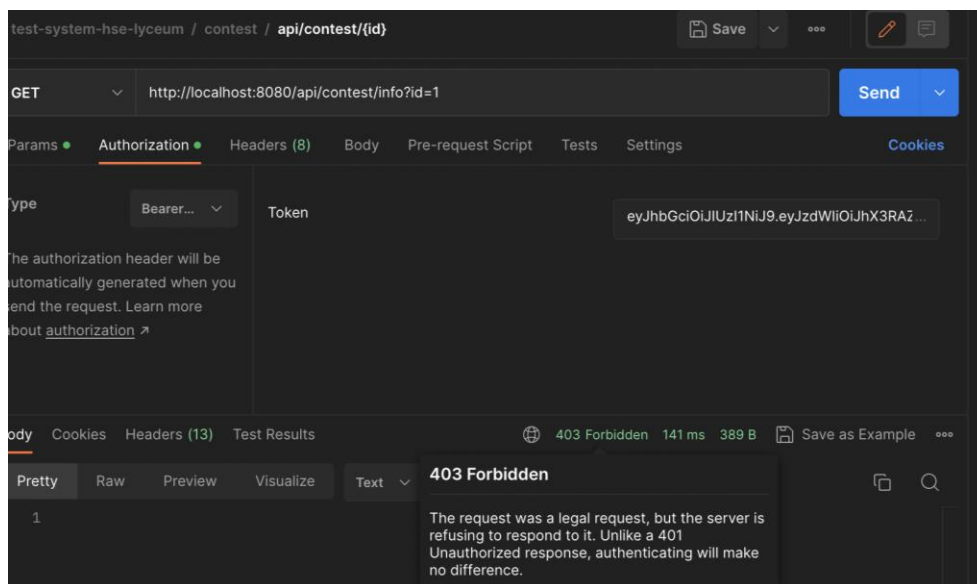
### 6.2.7. Испытание функции «Получение информации о контесте и его задачах»

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

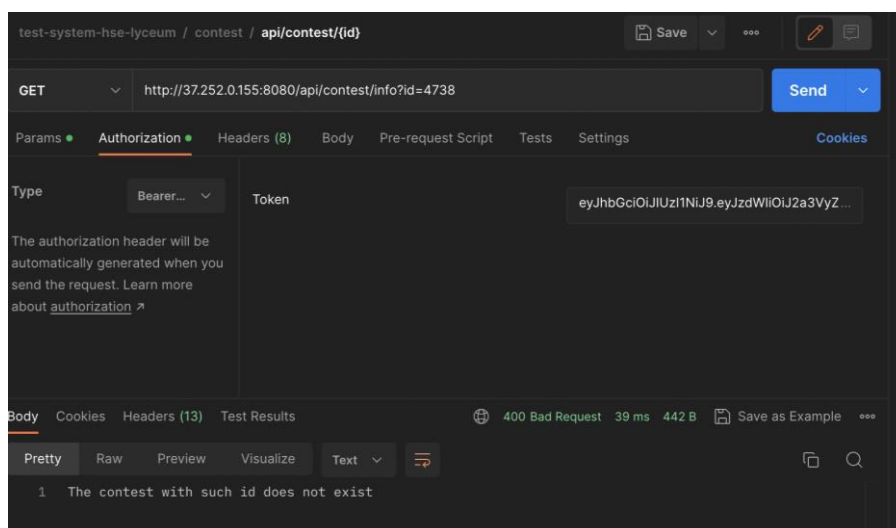
## RU.17701729.05.10-01 51 01–1

Для получения информации о конкурсе и его задачах, имеющихся в тестирующей системе, требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:



При верной аутентификации, но неверном вводе айди конкурса, о котором необходимо получить информацию будет получен ответ, содержащий сообщение об ошибке о несуществовании конкурса с указанным айди в тестирующей системе. Скриншот из коллекции

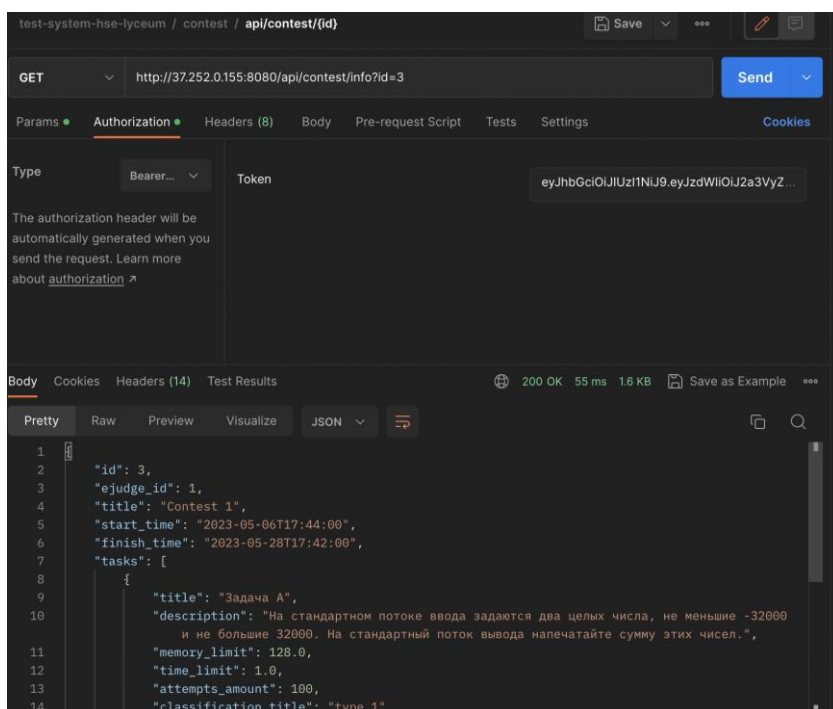


Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий информацию о констесте и его задачах. Скриншот из коллекции Postman:



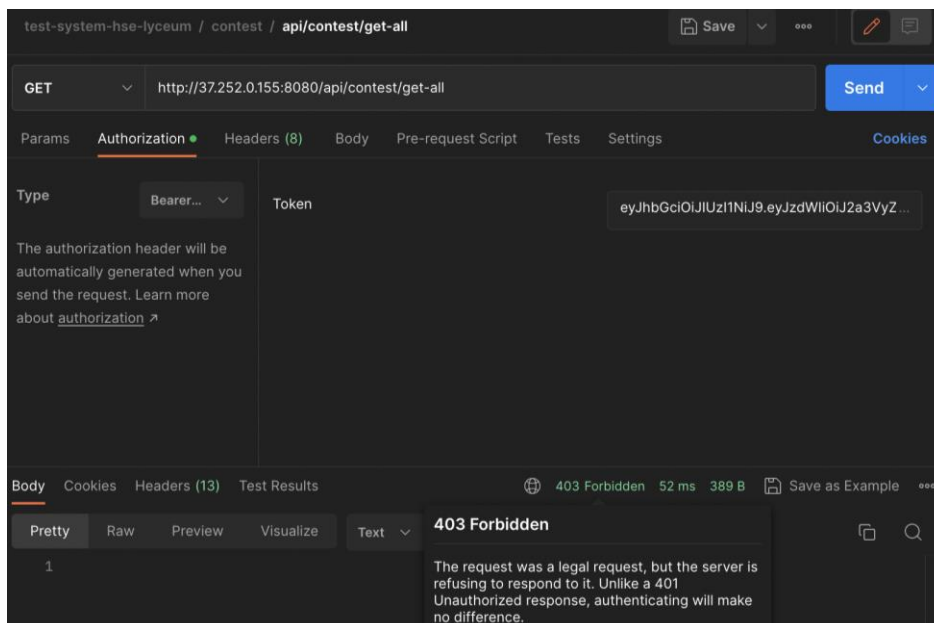
### 6.2.8. Испытание функции «Получение информации о всех констестах и их задачах, которые имеются в системе»

Для получения информации о всех констестах и их задачах, которые имеются в тестирующей системе, требуется использование аутентификации с помощью JWT-токена.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

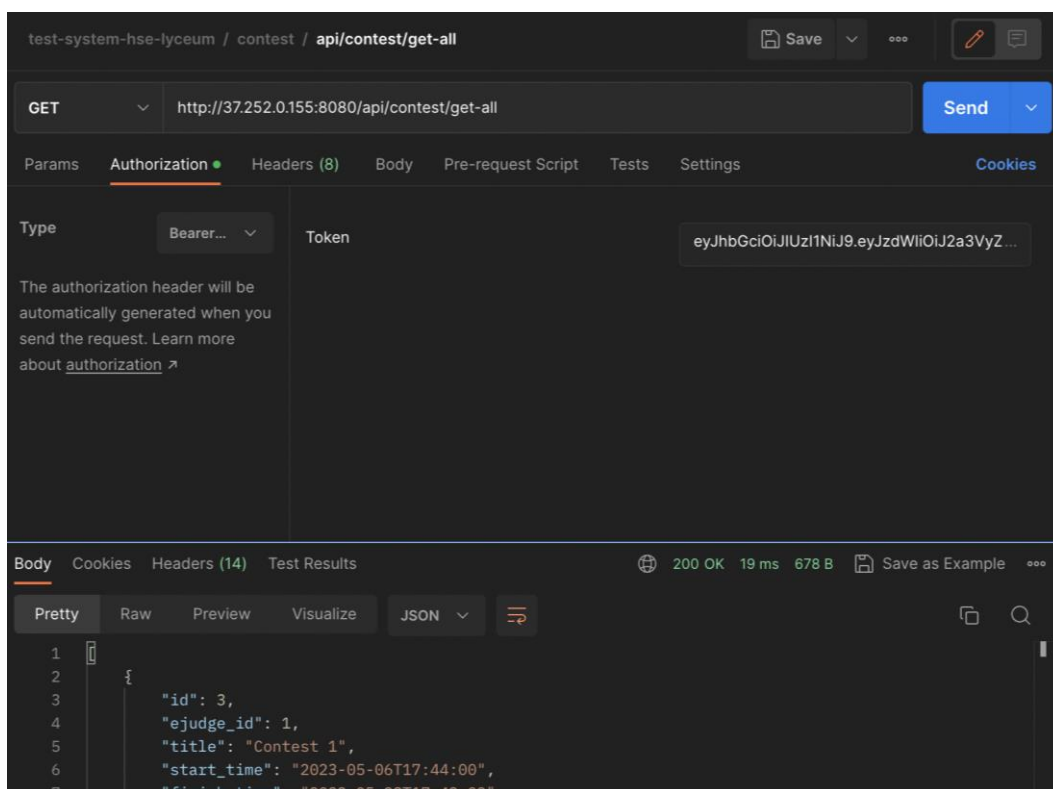
## RU.17701729.05.10-01 51 01–1

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет



получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий информацию о всех контестах и их задачах, имеющихся в тестирующей системе. Скриншот из коллекции Postman:



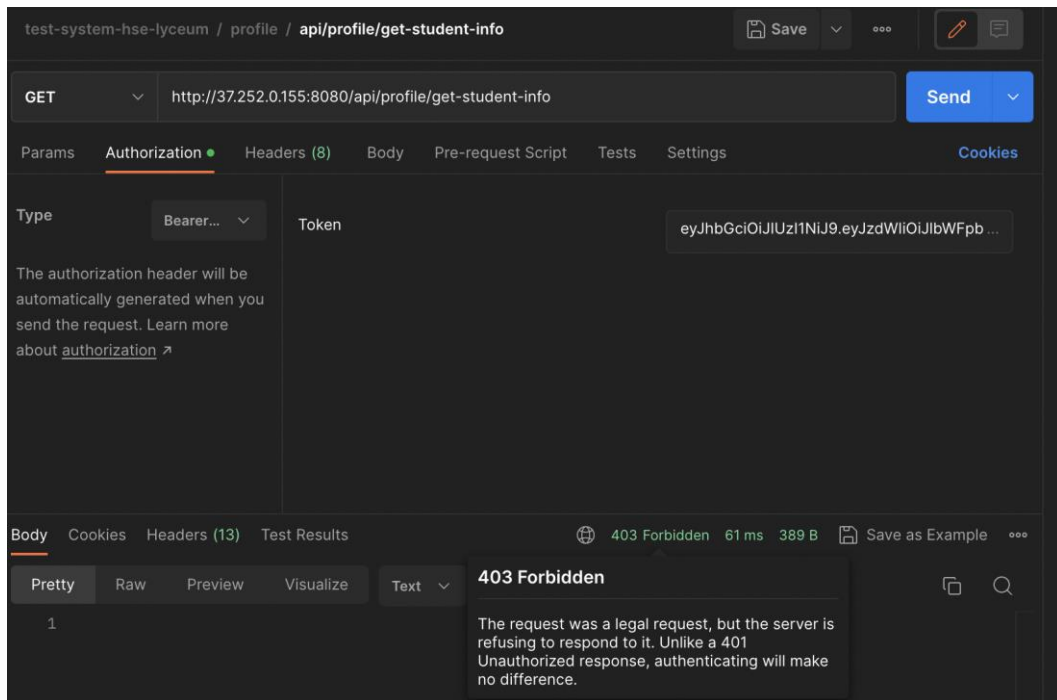
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

### 6.2.9. Испытание функции «Получение персональной информации в личном кабинете (информации о профиле) – это запросы с эндпоинтами /get-student-info, /get-teacher-info (работает аналогично)

Для получения персональной информации в личном кабинете (информации о профиле) требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет

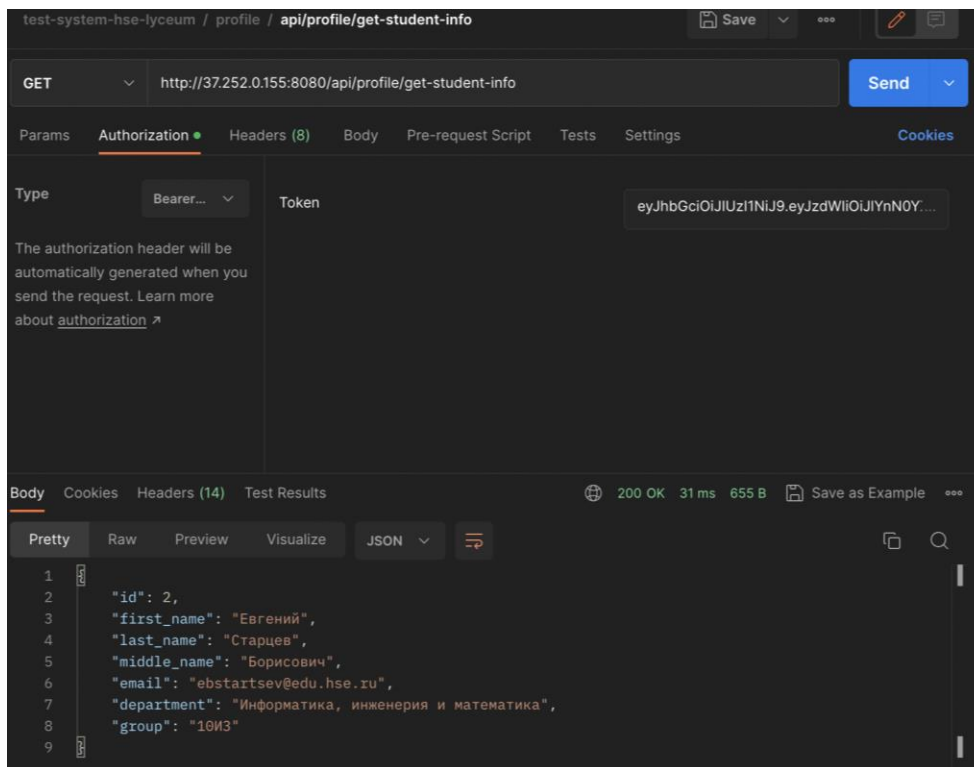


получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий персональную информацию, которая содержится в личном кабинете, о



пользователе. Скриншот из коллекции Postman:

### 6.2.10. Испытание функции «Получение информации о роли в системе»

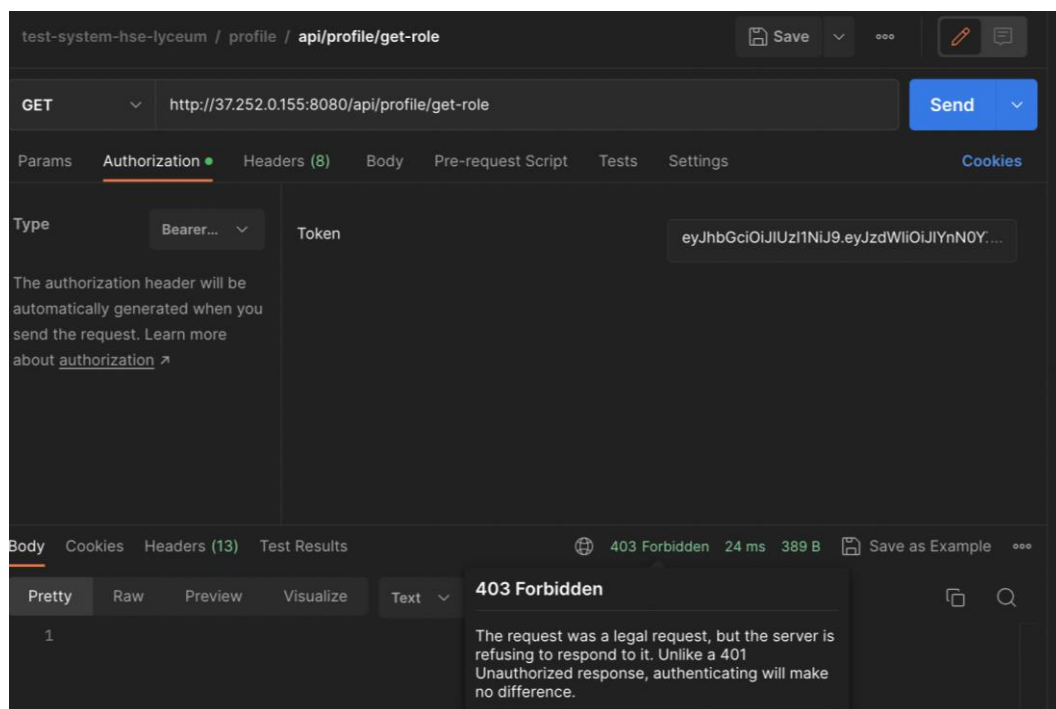
Для получения информации о роли в системе требуется использование аутентификации с помощью JWT-токена.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



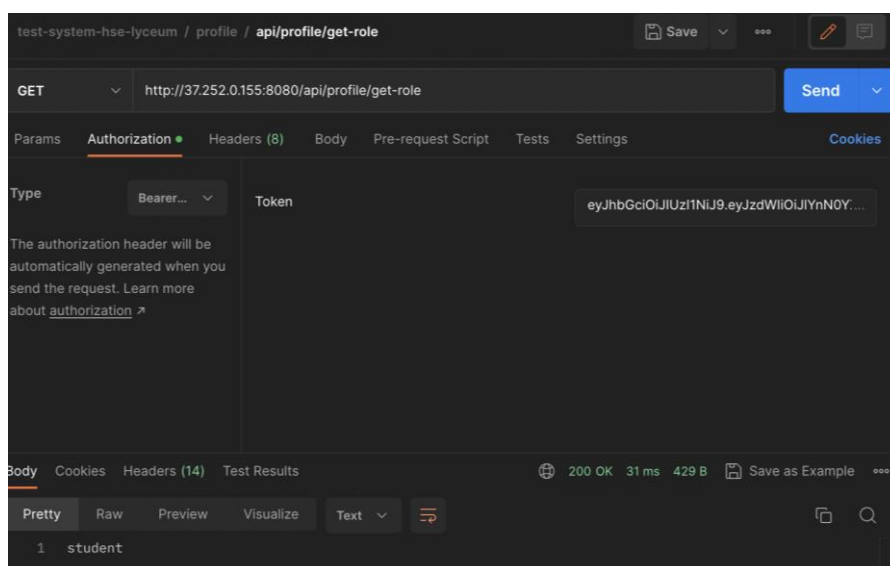
## RU.17701729.05.10-01 51 01–1

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет



получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий информацию о роли пользователя в тестирующей системе. Скриншот из коллекции Postman:



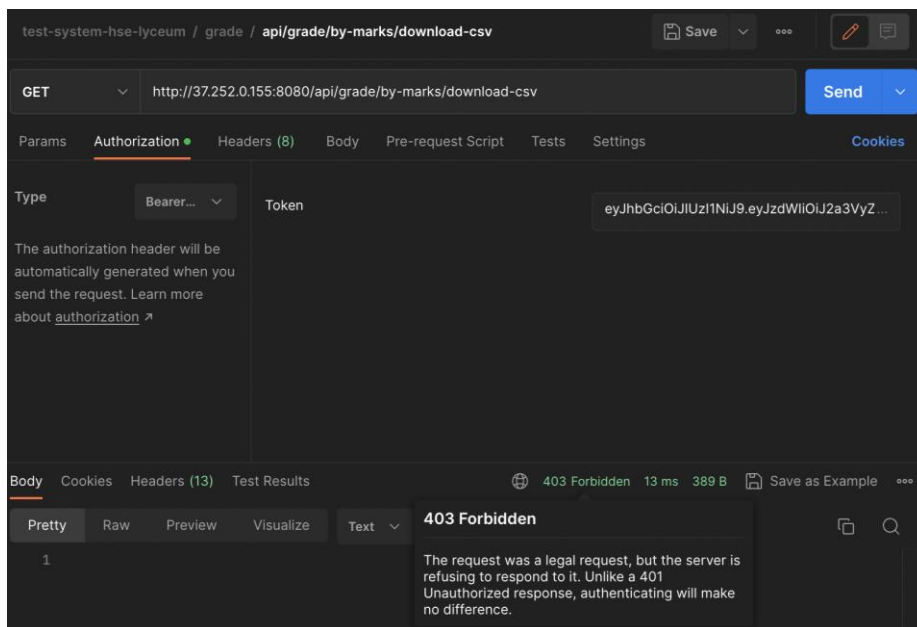
### 6.2.11. Испытание функции «Скачивание рейтинга по оценкам в формате .csv» или «Скачивание рейтинга по количеству задач в формате .csv» (работает аналогично)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

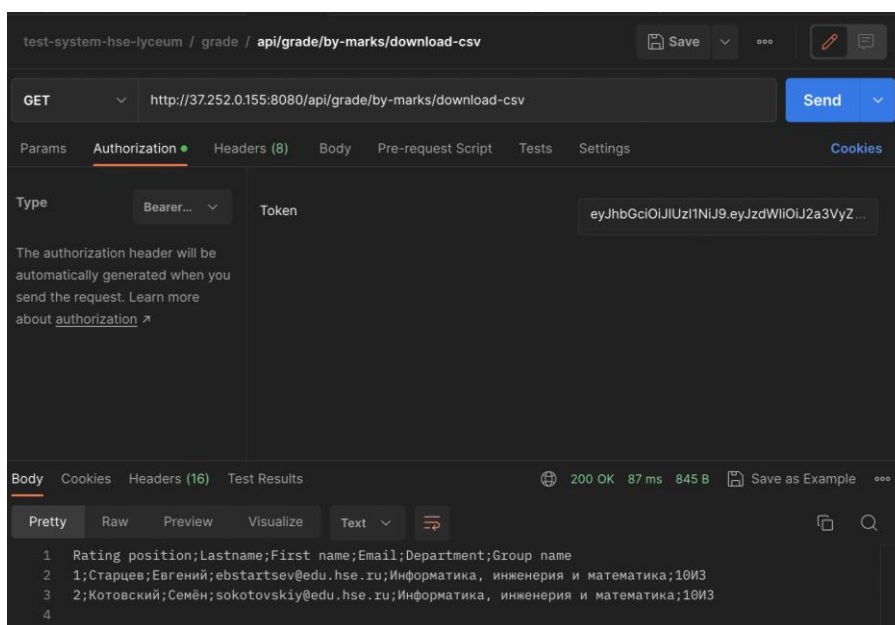
## RU.17701729.05.10-01 51 01–1

Для скачивания рейтинга определенного типа в формате .csv из тестирующей системы требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:



При верной аутентификации и отсутствии возникновения ошибок будет скачан файл с отсортированным рейтингом пользователей определённого типа в формате .csv. Скриншот из



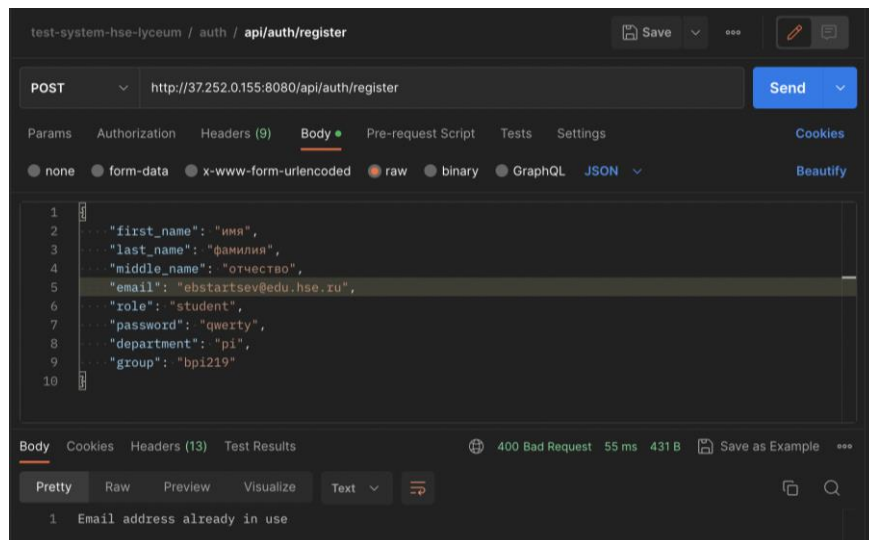
коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

### 6.2.12. Испытание функции «Создание аккаунта для учащегося»

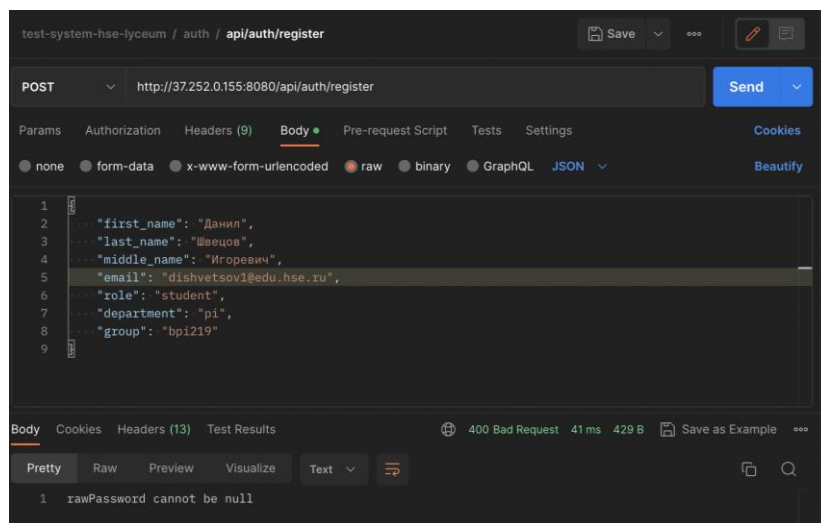
Для создания аккаунта для учащегося в тестирующей системе не требуется использование аутентификации.

При некорректном теле запроса, например, при попытке добавить учащегося с уже существующим email, будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить добавление учащегося в систему, так как учащийся с таким email уже



зарегистрирован в системе. Скриншот из коллекции Postman:

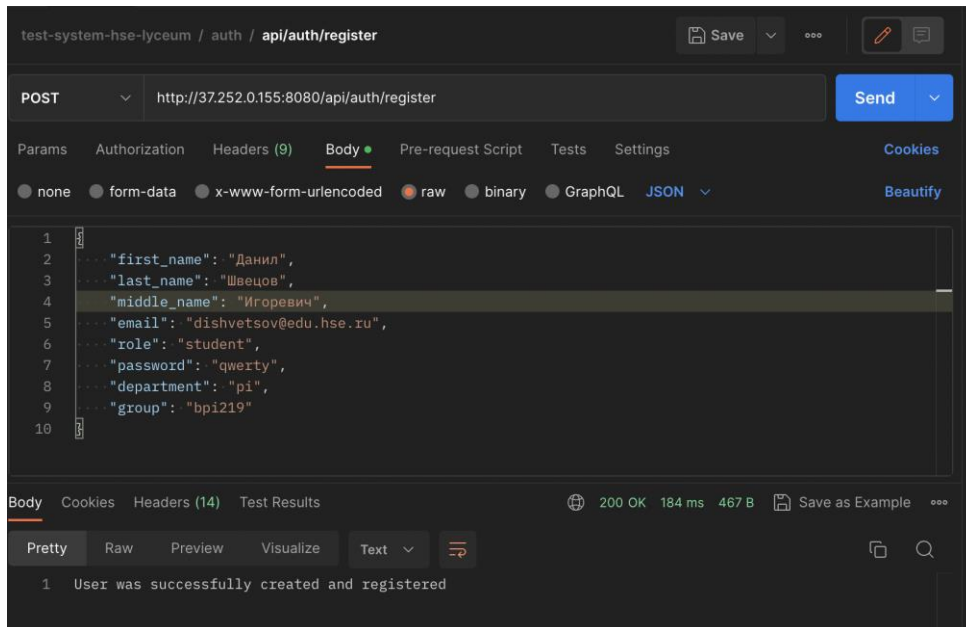
При верном email (то есть использовании email, которого нет в системе), но отсутствии какого-либо из других обязательных полей в теле запроса будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить создание аккаунта учащегося в тестирующую



систему при отсутствии данного параметра. Скриншот из коллекции Postman:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

При отсутствии возникновения ошибок будет получен ответ об успешном создании аккаунта для учащегося в тестирующей системе. Скриншот из коллекции Postman:



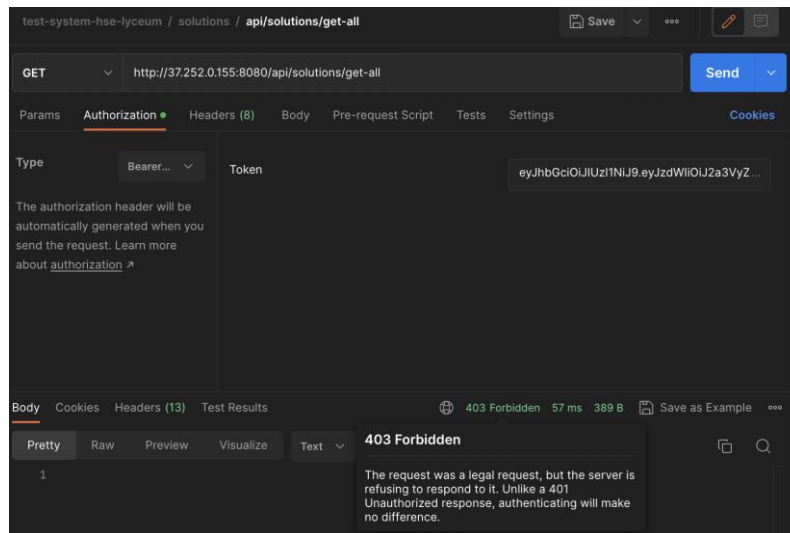
### 6.2.13. Испытание функции «Получение посылок всех пользователей всех решений контеста»

Для получения посылок всех пользователей всех решений контестов в тестирующей системе требуется использование аутентификации с помощью JWT-токена.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

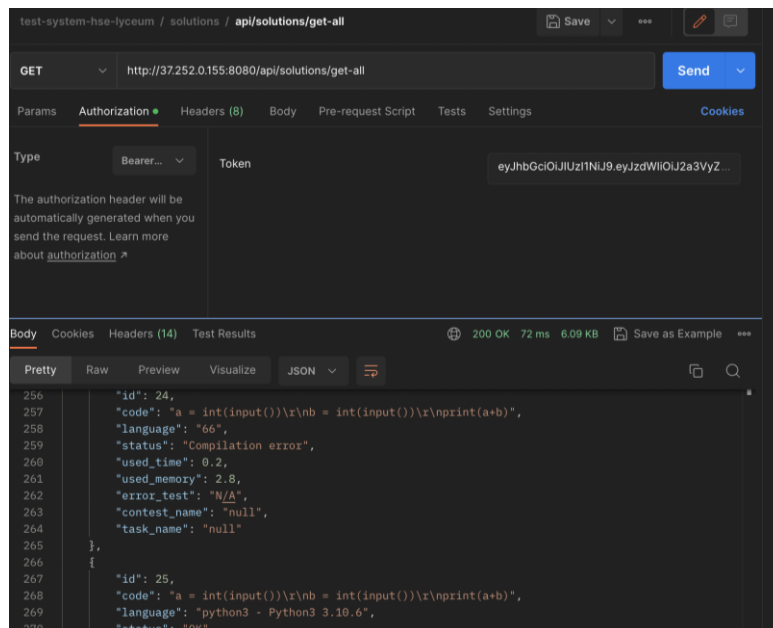
## RU.17701729.05.10-01 51 01–1

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет



получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:

При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий информацию о всех послылках всех пользователей, имеющих в тестирующей



системе. Скриншот из коллекции Postman:

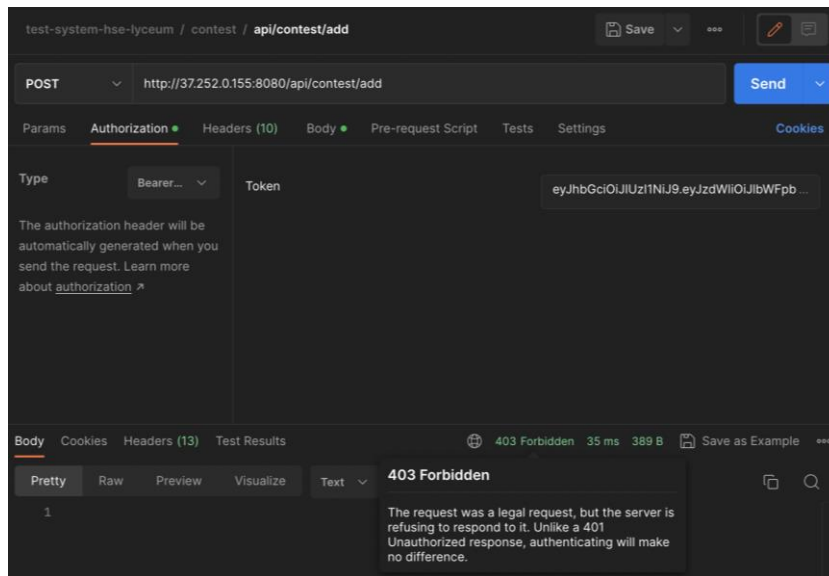
### 6.2.14. Испытание функции «Добавление нового контеста с задачами данного контеста в систему»

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

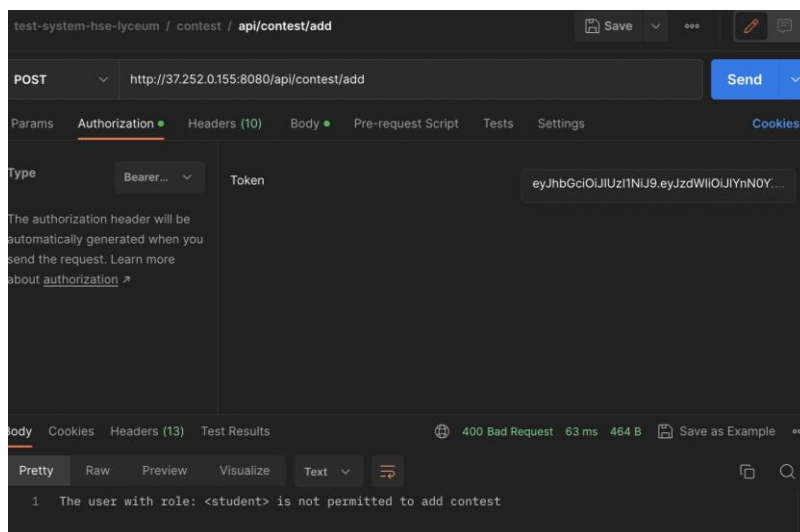
## RU.17701729.05.10-01 51 01–1

Для добавления нового конкурса с задачами данного конкурса в тестирующую систему требуется использование аутентификации с помощью JWT-токена.

При отсутствии Bearer Token при аутентификации или его некорректном вводе будет получен ответ со статус кодом 403 (Forbidden). Скриншот из коллекции Postman:



При попытке добавить конкурс с задачами в тестирующую систему пользователем, имеющим роль учащегося, будет получен ответ, содержащий информацию о том, что права доступа пользователя, имеющего роль учащегося, не позволяют ему осуществить данную

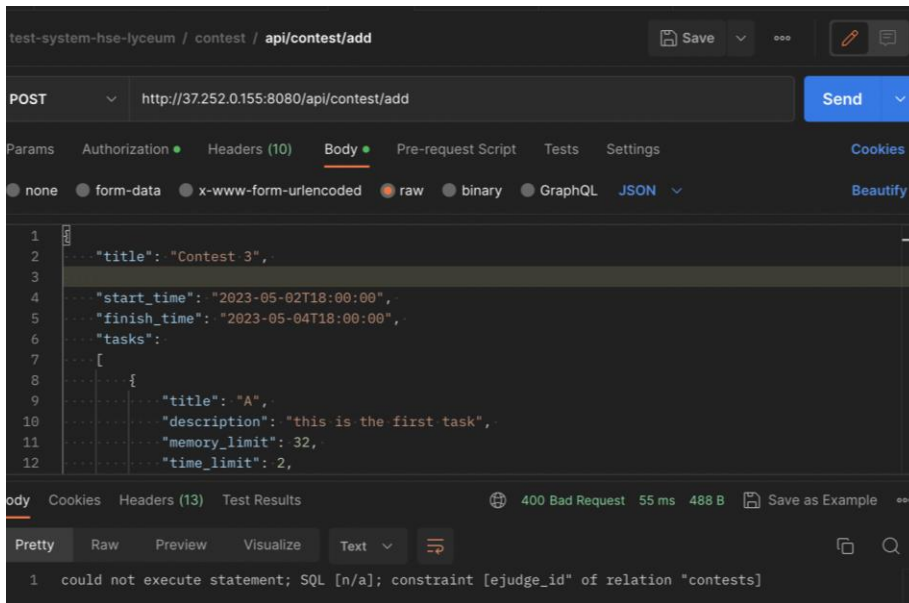


операцию. Скриншот из коллекции Postman:

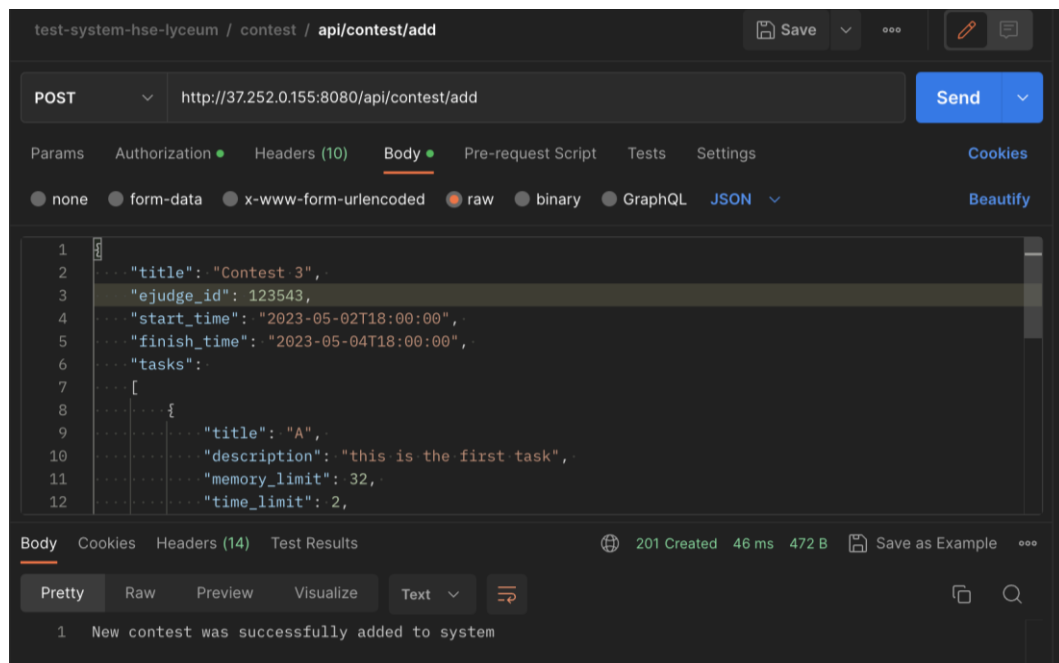
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

## RU.17701729.05.10-01 51 01–1

При верной роли пользователя, но отсутствии какого-либо из других обязательных полей в теле запроса будет получен ответ, содержащий сообщение об ошибке о невозможности выполнить добавление нового контекста с задачами данного контекста в тестирующую систему при отсутствии данного параметра. Скриншот из коллекции Postman:



При верной аутентификации и отсутствии возникновения ошибок будет получен ответ, содержащий информацию об успешном добавлении нового контекста с задачами данного контекста в тестирующую систему. Скриншот из коллекции Postman:



Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и

**6.3. Испытание выполнения требований к надёжности**

В случае возникновения ошибки или исключения в программе на сторону клиента возвращается данная ошибка с сообщением о возникшей ошибке в теле запроса. На стороне клиента отображается либо ошибка с сервера, либо текст, демонстрирующий, что на сервере произошла ошибка.

На протяжении всех испытаний программа работала корректно и не завершалась аварийно.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.10-01 51				
Инв. № подл.	Подп. и	Взам. инв.	Инв. №	Подп. и



