

### Задание 3 (Повторная аттестация)

## Составленные API-тест-кейсы по методам для работы с видео YouTube

Для начала работы с методами нам потребовалось привязать к Google-почте разработческий профиль, создать API key и access token

№	Название	Тестовые данные	Шаги	Ожидаемый результат	Приоритет	Статус		
Позитивные функциональные API тест-кейсы								
1	Получить список видео по id jifUJrYPZQQ	Тип запроса : Get      Запрос: <a href="https://www.googleapis.com/youtuube/v3/videos?id={{id}}">https://www.googleapis.com/youtuube/v3/videos?id={{id}}</a>	1. Отправить Get запрос <a href="https://www.googleapis.com/youtuube/v3/videos?id={{id}}">https://www.googleapis.com/youtuube/v3/videos?id={{id}}</a> 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить Response Size 6. Проверить тело ответа от сервера на соответствие документации	Запрос успешно отправлен на сервер HTTP запрос - <b>200</b> OK Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки (695 ms) Ответ пришел в разумных размерах (683 B) Тело ответа    возвращается и имеет вид, соответствующий документации	Passed	Высокий		
2	Поставить like видео c id jifUJrYPZQQ	Тип запроса : <a href="https://www.googleapis.com/youtuube/v3/videos/rate?id=jifUJrYPZQQ&amp;rating=like">https://www.googleapis.com/youtuube/v3/videos/rate?id=jifUJrYPZQQ&amp;rating=like</a>  Раздел Tests    var key = "issue_number" var value = pm.response.json().number	1. Отправить POST запрос <a href="https://www.googleapis.com/youtuube/v3/videos/rate?id=jifUJrYPZQQ&amp;rating=like">https://www.googleapis.com/youtuube/v3/videos/rate?id=jifUJrYPZQQ&amp;rating=like</a>	Запрос успешно отправлен на сервер			Passed	Высокий
			2. Проверить код ответа	HTTP запрос - <b>204</b> OK				
			3. Проверить название	Headers request:				
				Content type: application / json				
			4. Проверить скорость ответа	Accept: application / json				
				Ответ пришел в разумные ожидаемые сроки				
				В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)				
6. Проверить тело ответа от сервера на соответствие документации	Тело ответа    возвращается и имеет вид, соответствующий документации							

3	Получить список самых популярных видео	Тип запроса : Get    Запрос: <a href="https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&amp;chart=mostPopular&amp;regionCode=US">https://youtube.googleapis.com/youtube/v3/videos?part=snippet%2CcontentDetails%2Cstatistics&amp;chart=mostPopular&amp;regionCode=US</a>	1. Отправить Get запрос	Запрос успешно отправлен на сервер	Passed	Высокий
			2. Проверить код ответа	HTTP запрос - <b>200 OK</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
4	Проверить, что API вернул корректный список видео, отсортированный по популярности.	Тип запроса : GET    Запрос: <a href="https://....">https://....</a>	1. Отправить GET-запрос к API с параметром user_id идентификатора пользователя	Получен успешный ответ от API с списком рекомендованных видео для указанного пользователя	Passed	Высокий
			2. Проверить код ответа	HTTP запрос - <b>200 OK</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
5	Получить информацию о пользователе, загрузившем видео с id jifUJrYPZQQ	Тип запроса : GET Запрос: <a href="https://.....">https://.....</a>	1. Отправить GET запрос к API с параметром id=jifUJrYPZQQ и action=user	Запрос успешно отправлен на сервер. Получен успешный ответ от API с информацией о пользователе, загрузившем видео		
			2. Проверить код ответа	HTTP запрос - <b>200 OK</b>		
			3. Проверить название	Headers request:		

		Тело запроса в формате Json { "id": "" }		Content type: application / json	Passed	Высокий
				Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить отображение UI	В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)		
			6. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
6	Добавить комментарий к видео с id jifUJrYPZQQ	Тип запроса : POST Запрос: <a href="https://.....">https://.....</a>  Тело запроса в формате Json { "id": "jifUJrYPZQQ", "body": "Супер", "assignee": "MiraS75", }	1. Отправить POST-запрос к API с параметрами <b>id=jifUJrYPZQQ</b> и текстом комментария <b>"Супер"</b>	Получен успешный ответ от API, подтверждающий успешное добавление комментария	Passed	Высокий
			2. Проверить код ответа	HTTP запрос - <b>204 OK</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		

7	Получить список рекомендуемых видео для пользователя с определенным id	Тип запроса : <b>GET</b> Запрос: <a href="https://....">https://....</a>	1. Отправить GET-запрос к API с параметром user_id идентификатора пользователя 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить тело ответа от сервера на соответствие документации	Получен успешный ответ от API с списком рекомендованных видео для указанного пользователя HTTP запрос - <b>200 OK</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки Тело ответа возвращается и имеет вид, соответствующий документации	<b>Passed</b>	Высокий
8	Изменить описание видео с id jifUJrYPZQQ	Тип запроса : <b>PUT</b> Запрос: <a href="https://.....">https://.....</a>	1. Отправить PUT-запрос к API с параметрами id=jifUJrYPZQQ и новым текстом описания 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить тело ответа от сервера на соответствие документации	Получен успешный ответ от API, подтверждающий успешное изменение описания видео HTTP запрос - <b>200 OK</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки Тело ответа возвращается и имеет вид, соответствующий документации	<b>Passed</b>	Высокий
9	Удалить комментарий к видео с id jifUJrYPZQQ	Тип запроса : <b>DELETE</b> Запрос: <a href="https://.....">https://.....</a>	1. Отправить DELETE-запрос к API с параметрами id=jifUJrYPZQQ и id комментария 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить отображение UI	Получен успешный ответ от API, подтверждающий успешное удаление комментария HTTP запрос - <b>200 OK</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)	<b>Passed</b>	Высокий

			6. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
<b>Негативные функциональные API тест-кейсы</b>						
10	Передать некорректный идентификатор видео (например, пустую строку или случайную строку) и проверить, что API вернул ошибку или пустой список	Тип запроса : Get Запрос: <a href="https://.....">https://.....</a>	1. Отправить Get запрос <a href="https://.....">https://.....</a> 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить тело ответа от сервера на соответствие документации	Запрос успешно отправлен на сервер HTTP запрос - <b>404 NOT FOUND</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки Тело ответа возвращается и имеет вид, соответствующий документации	<b>Passed</b>	Низкий
11	Передать некорректный тип идентификатора (например, число вместо строки) и проверить, что API вернул ошибку или пустой список	Тип запроса : POST Запрос: <a href="https://....">https://....</a>	1. Отправить POST запрос <a href="https://.....">https://.....</a> 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа 5. Проверить отображение UI 6. Проверить тело ответа от сервера на соответствие документации	Запрос успешно отправлен на сервер HTTP запрос - <b>404 NOT FOUND</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.) Тело ответа возвращается и имеет вид, соответствующий документации	<b>Passed</b>	Низкий
12	Передать несуществующий идентификатор видео и проверить, что API вернул ошибку или пустой список	Тип запроса : POST Запрос: <a href="https://....">https://....</a>	1. Отправить POST запрос <a href="https://.....">https://.....</a> 2. Проверить код ответа 3. Проверить название 4. Проверить скорость ответа	Запрос успешно отправлен на сервер HTTP запрос - <b>404 NOT FOUND</b> Headers request: Content type: application / json Accept: application / json Ответ пришел в разумные ожидаемые сроки	<b>Passed</b>	Низкий

			5. Проверить отображение UI	В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)	Passed	Низкий
			6. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
13	Попытаться поставить like без авторизации и проверить, что API вернул ошибку или отказал в доступе.	Тип запроса : Get Запрос: <a href="https://.....">https://.....</a>	1. Отправить Get запрос <a href="https://.....">https://.....</a>	Запрос успешно отправлен на сервер	Passed	Низкий
			2. Проверить код ответа	HTTP запрос - <b>404 NOT FOUND</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
14	Передать некорректный идентификатор видео (например, пустую строку или случайную строку) и проверить, что API вернул ошибку или отказал в доступе	Тип запроса : <b>POST</b> Запрос: <a href="https://....">https://....</a>	1. Отправить <b>POST</b> запрос <a href="https://.....">https://.....</a>	Запрос успешно отправлен на сервер	Passed	Низкий
			2. Проверить код ответа	HTTP запрос - <b>404 NOT FOUND</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		
			5. Проверить отображение UI	В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)		
			6. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		
15	Передать несуществующий идентификатор видео и проверить, что API вернул ошибку или отказал в доступе.	Тип запроса : <b>POST</b> Запрос: <a href="https://....">https://....</a>	1. Отправить <b>POST</b> запрос <a href="https://.....">https://.....</a>	Запрос успешно отправлен на сервер	Passed	Низкий
			2. Проверить код ответа	HTTP запрос - <b>404 NOT FOUND</b>		
			3. Проверить название	Headers request: Content type: application / json Accept: application / json		
			4. Проверить скорость ответа	Ответ пришел в разумные ожидаемые сроки		

			5. Проверить отображение UI	В репозитории отображается созданная задача с заданными в запросе параметрами (название, автор и т.д.)	Где	Где
			6. Проверить тело ответа от сервера на соответствие документации	Тело ответа возвращается и имеет вид, соответствующий документации		

