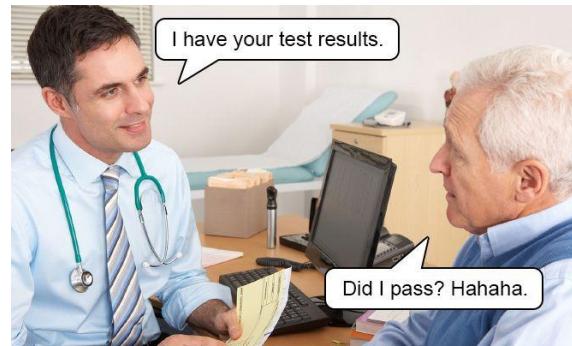


Me when I am waiting for my test results
and knowing I am gonna fail



EATLIVER.COM

Test Report

АВТОРЫ ДОКУМЕНТА
КСЕНИЯ КАМЫШАНСКАЯ
ЕВГЕНИЙ ГУРИН
ПАВЕЛ ЕФАРИНОВ

Принимал участие: сервер



1. Data and Database Integrity Testing

report1.html

Report generated on 17-Jan-2024 at 21:16:55 by [pytest-html](#) v4.1.1

Environment

Python	3.8.10
Platform	Windows-10-10.0.22631-SP0
Packages	<ul style="list-style-type: none">• pytest: 7.2.1• pluggy: 1.3.0
Plugins	<ul style="list-style-type: none">• anyio: 4.2.0• asyncio: 0.23.3• html: 4.1.1• html-reporter: 0.2.9• metadata: 3.0.0
JAVA_HOME	C:\Program Files\Java\jdk-17

Summary

13 tests took 00:00:01.

(Un)check the boxes to filter the results.

Result		Test ▲	Show all details / Hide all details
Passed		tests/database-integration/test_db_schema.py::test_create_get_delete[OrderParamValueService-create_order_param]	118 ms
Passed		tests/database-integration/test_db_schema.py::test_create_get_delete[OrderService-create_order]	531 ms
Passed		tests/database-integration/test_db_schema.py::test_create_get_delete[OrderTypeParamService-create_order_type_param]	41 ms
Passed		tests/database-integration/test_db_schema.py::test_create_get_delete[OrderTypeService-create_order_type]	73 ms
Passed		tests/database-integration/test_db_schema.py::test_create_order_params_not_exists	131 ms
Passed		tests/database-integration/test_db_schema.py::test_create_order_type_all_params	33 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order-expected_table_cols2]	11 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order_confirmation-expected_table_cols4]	14 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order_param_value-expected_table_cols3]	12 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order_status_update-expected_table_cols5]	12 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order_type-expected_table_cols0]	144 ms
Passed		tests/database-integration/test_db_schema.py::test_table_columns[order_type_param-expected_table_cols1]	11 ms
Passed		tests/database-integration/test_db_schema.py::test_tables_exist	46 ms

Название	Описание	Сценарий
test_tables_exist	Проверка существования таблиц	<ol style="list-style-type: none">1. получение списка таблиц бд2. сравнение с ожидаемым списком таблиц
test_table_columns	Проверка типов колонок таблиц	<ol style="list-style-type: none">1. выбор таблицы2. получение названий колонок и типов таблицы бд3. сравнение с ожидаемыми параметрами названия колонки и её типа данных
test_create_get_delete	Проверка создания, получения и удаления элемента	<ol style="list-style-type: none">1. выбор таблицы2. создание элемента сущности3. коммит транзакции4. проверка существования получением элемента по id5. сравнение значений полей с ожидаемыми

		<ol style="list-style-type: none"> 6. проверка существования получением списка всех элементов и наличие в нём созданного 7. удаление элемента по id 8. проверка отсутствия элемента получением по id и поиском в списке всех элементов
test_create_order_type_all_params	Проверка создания заказа со всеми типами параметров	<ol style="list-style-type: none"> 1. Создание шаблона заказа 2. Создание параметров для шаблона заказа, используя все типы для параметров 3. Создание заказа по шаблону 4. Прикрепление параметров заказа к созданному заказу 5. проверка существования созданного заказа 6. проверка соответствия параметров
test_create_order_params_not_exists	Проверка возможности создания заказа с отсутствующим параметром	<ol style="list-style-type: none"> 1. Создание нового заказа 2. Попытка прикрепления параметра к заказу, которого не существует 3. Ожидаемое получение ошибки

2. Function Testing

9 tests took 00:00:07.

(Un)check the boxes to filter the results.

0 Failed, 9 Passed, 0 Skipped, 0 Expected failures, 0 Unexpected passes, 0 Errors, 0 Reruns

Result ▲	Test
Passed	tests/api/test_order.py::test_3_1_5_customer_cannot_view_requests
Passed	tests/api/test_order.py::test_3_1_6_user_can_create_order
Passed	tests/api/test_order.py::test_3_1_7_user_can_create_wood_request
Passed	tests/api/test_order.py::test_3_1_13_user_can_create_defect_request
Passed	tests/api/test_order.py::test_3_1_17_user_can_filter_requests
Passed	tests/api/test_order.py::test_3_1_20_user_can_change_status_of_any_request[new_statuses_updates0]
Passed	tests/api/test_order.py::test_3_1_20_user_can_change_status_of_any_request[new_statuses_updates1]
Passed	tests/api/test_order.py::test_3_1_20_user_can_change_status_of_any_request[new_statuses_updates2]
Passed	tests/api/test_order.py::test_3_1_20_user_can_change_status_of_any_request[new_statuses_updates3]

Название	Описание	Сценарий
test_3_1_5_customer_cannot_view_requests	Заказчик может просматривать только заказ, без детализации	Предварительно создается заказ с заявкой. Заказчик заходит в систему. Ему доступен заказ на баню, но не заявка на брак.
test_3_1_6_user_can_create_order	Пользователь системы может создать заказ на баню	<ol style="list-style-type: none"> Пользователь заходит в систему с правами работника. Пользователь создает заказ на баню.
test_3_1_7_user_can_create_wood_request	Пользователь системы может создать заявку на сруб	<ol style="list-style-type: none"> Пользователь заходит в систему с правами работника. Пользователь создает заказ на баню Пользователь создает заявку на сруб
test_3_1_13_user_can_create_defect_request	Пользователь системы может создать заявку на брак	<ol style="list-style-type: none"> Пользователь заходит в систему с правами работника. Пользователь создает заказ на баню Пользователь создает заявку на сруб Пользователь создает заявку на брак сруба
test_3_1_17_user_can_filter_requests	Пользователь может фильтровать заявки	<ol style="list-style-type: none"> Пользователь заходит в систему с правами работника Пользователь создает несколько заказов на баню Пользователь фильтрует созданные заявки (по типу заявки)
test_3_1_20_user_can_change_status_of_any_request	Пользователь может менять статусы заявок	<ol style="list-style-type: none"> Пользователь заходит в систему с правами работника Пользователь

		<p>создает заказ на баню и заявку на сруб</p> <p>3. Пользователь может изменить статус заявки (см. параметры)</p>
--	--	---

3. Business Cycle Testing

Project: MPI*

Executing ▾

- ✓ Create defect request
- ✓ Create order*
- ✓ Create wood request
- ✓ Login-logout
- ✓ Read order info

Command	Target	Value
1 ✓ open	/realms/MPI/protocol/openid-connect/auth?client_id=frontend-client&redirect_uri=http%3A%2F%2Flocalhost%3A5173%2F&state=1bcf2d38-39bf-4f24-b66f-fc5bc220fa1d&response_mode=fragment&response_type=code&scope=openid&nonce=283862c9-62c6-45ba-a7bc-fef266e9751c	
2 ✓ set window size	784x824	
3 ✓ click	id=username	
4 ✓ type	id=username	
5 ✓ click	id=password	
6 ✓ type	id=password	
7 ✓ click	id=kc-login	
8 ✓ click	css=.card-pf	
9 ✓ type	id=username	
10 ✓ click	id=password	
11 ✓ type	id=password	
12 ✓ click	id=kc-login	
13 ✓ click	id=password	
14 ✓ type	id=password	
15 ✓ click	id=kc-login	
16 ✓ click	css=v-list-item:nth-child(3) .tw--ml-3 linkText=Зарегистрироваться	

Runs: 5 Failures: 0

Log Reference

- click on id=input-search-customer OK
- mouseOver on css=.v-btn--elevated > .v-btn__content OK
- type on id=input-search-customer with value z OK
- click on css=.v-btn--elevated > .v-btn__content OK
- mouseOver on css=.v-card-actions:nth-child(3) > .v-btn:nth-child(1) > .v-btn__content OK
- click on css=.v-card-actions:nth-child(3) > .v-btn:nth-child(1) > .v-btn__content OK
- click on id=input-Название OK

Название	Описание	Сценарий
Login-Logout	Пользователь может зайти в систему	<ol style="list-style-type: none"> Пользователь производит вход в систему После загрузки системы пользователь выходит
Create order	Пользователь системы	<ol style="list-style-type: none"> Пользователь

	может создать заказ на баню	<p>заходит в систему с правами работника.</p> <ol style="list-style-type: none"> 2. Пользователь создает заказ на баню.
Create wood request	Пользователь системы может создать заявку на сруб	<ol style="list-style-type: none"> 1. Пользователь заходит в систему с правами работника. 2. Пользователь создает заказ на баню 3. Пользователь создает заявку на сруб
Create defect request	Пользователь системы может создать заявку на брак	<ol style="list-style-type: none"> 1. Пользователь заходит в систему с правами работника. 2. Пользователь создает заказ на баню 3. Пользователь создает заявку на сруб 4. Пользователь создает заявку на брак сруба
Read order info	Пользователь может фильтровать заявки	<ol style="list-style-type: none"> 1. Пользователь заходит в систему с правами работника 2. Пользователь создает несколько заказов на баню 3. Пользователь фильтрует созданные заявки (по типу заявки)

4. User Interface Testing

Все довольны, всем понравилось **100**

Респондент	Юзкейс	Количество ошибок	Комментарий
Марина	Use-Case 1 “Авторизация”.	0 (0%)	Красиво, но долго
Алексей	Use-Case 2 “Создать заказ на СаБаКо”	2 (15%)	Долго прогружались пользователи, не всегда очевидные виджеты
Алексей	Use-Case 3 “Создать заявку на компонент СаБаКо”	2 (15%)	Не понятно, что заявку можно было создать при оформлении заказа
Олег	Use-Case 4 “Создать заявку на брак”.	1 (7%)	Не сразу нашлась страница с заявкой на которую надо завести брак

5. Performance Profiling

Сценарий

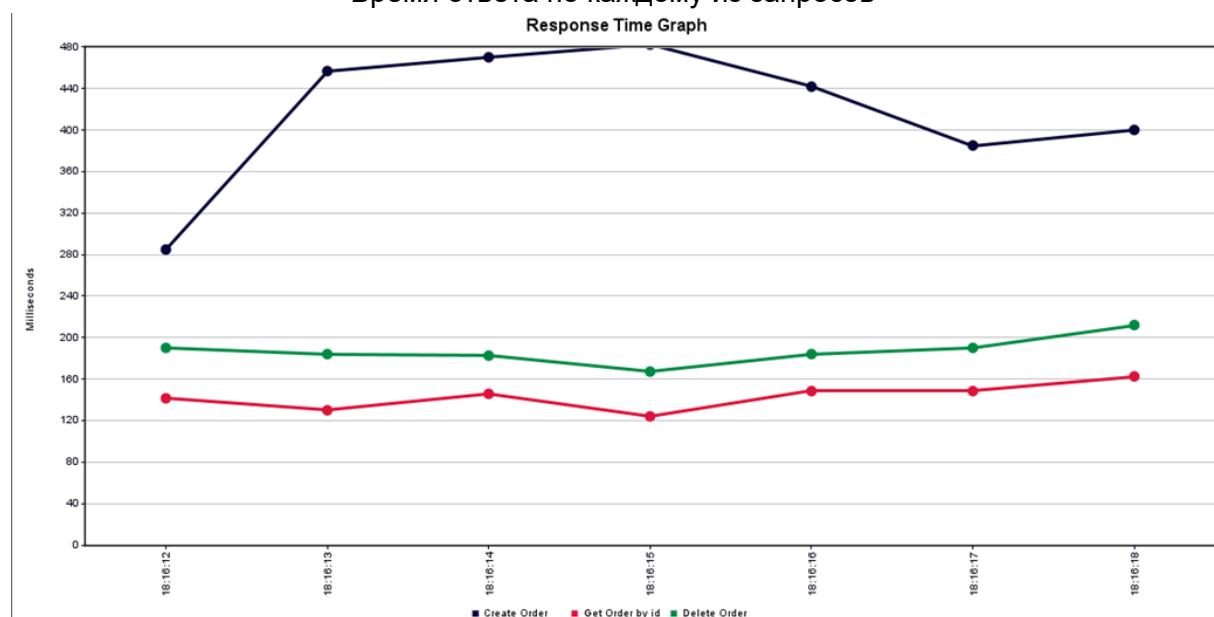
1. Создание заказа
2. Получение заказа
3. Удаление заказа

Проверка была сделана на 5,10,20 и 30 пользователях

5 пользователей



Время ответа по каждому из запросов

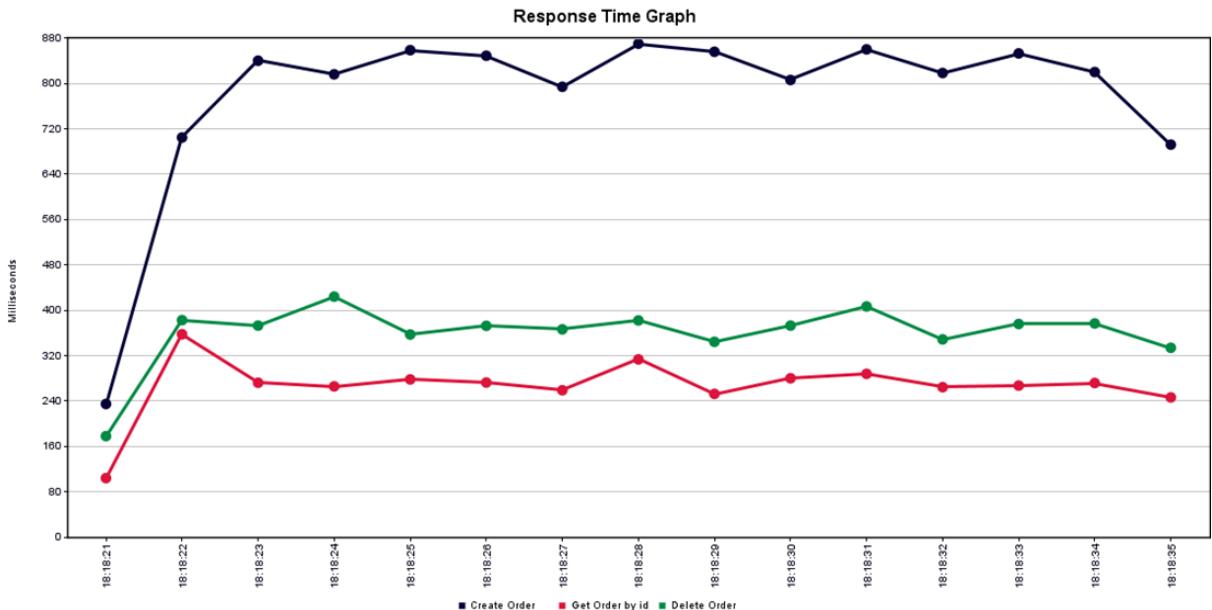


10 пользователей

Код ответа в секунду

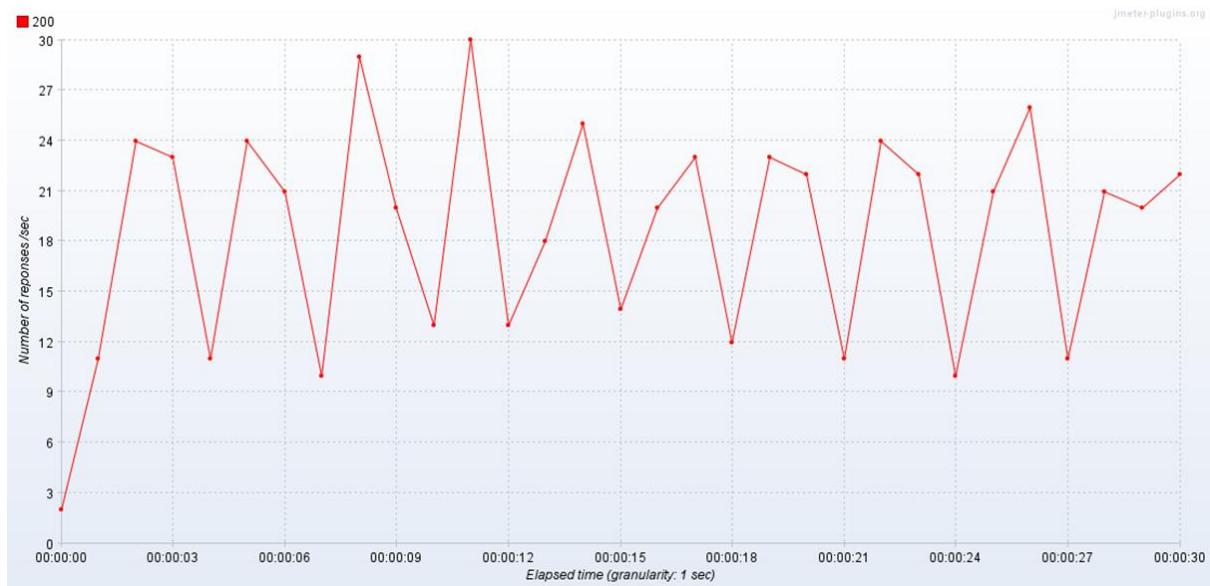


Время ответа по каждому из запросов



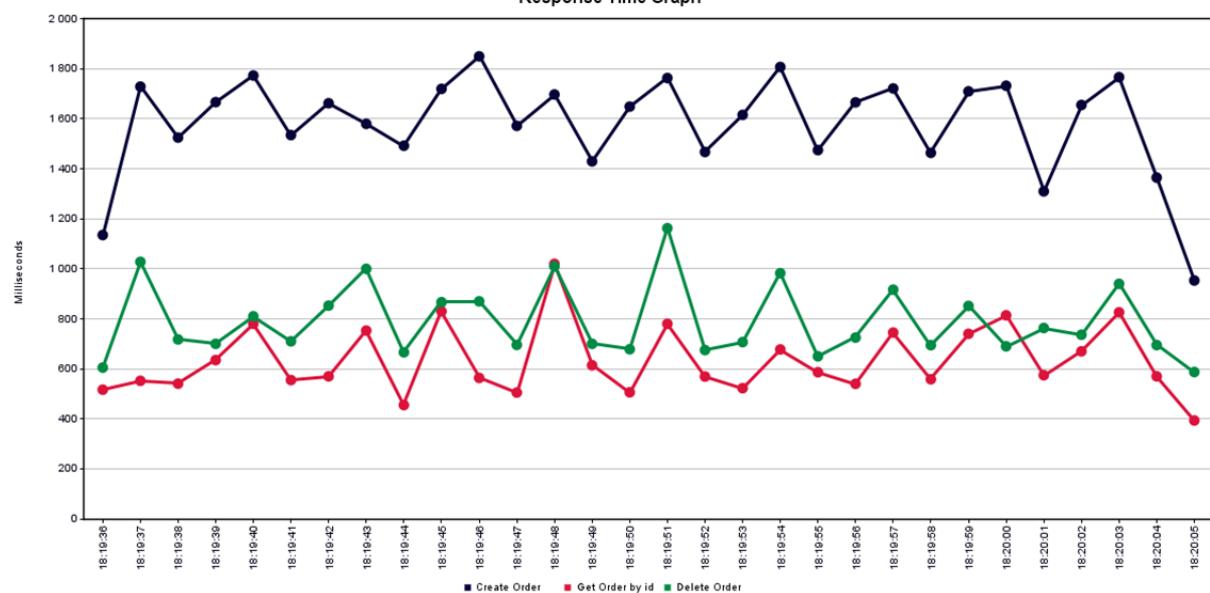
20 пользователей

Код ответа в секунду



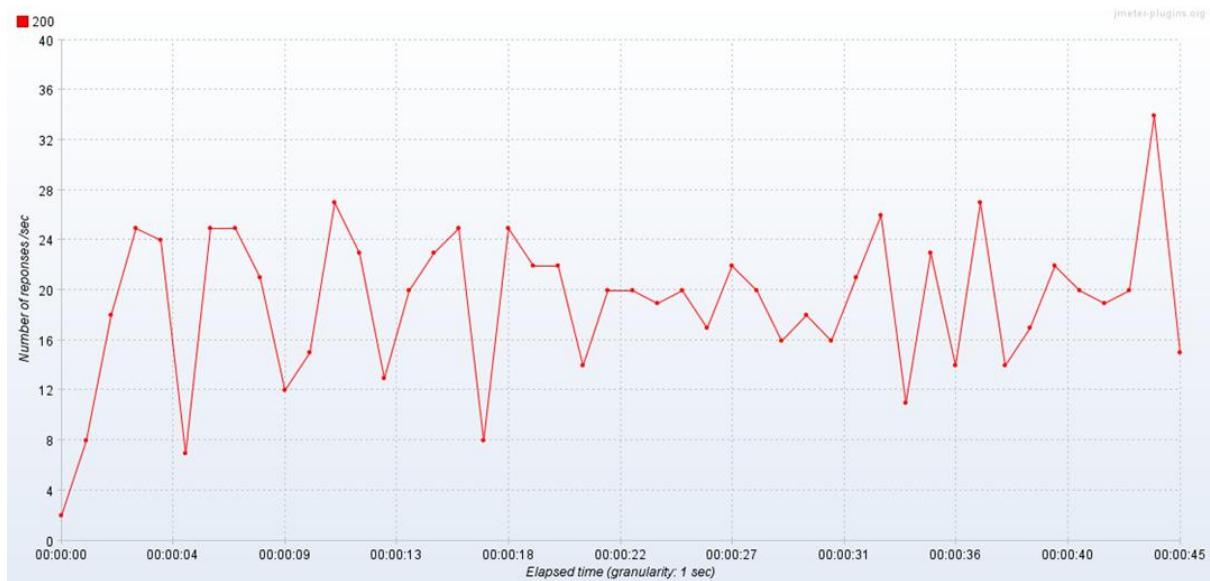
Время ответа по каждому из запросов

Response Time Graph



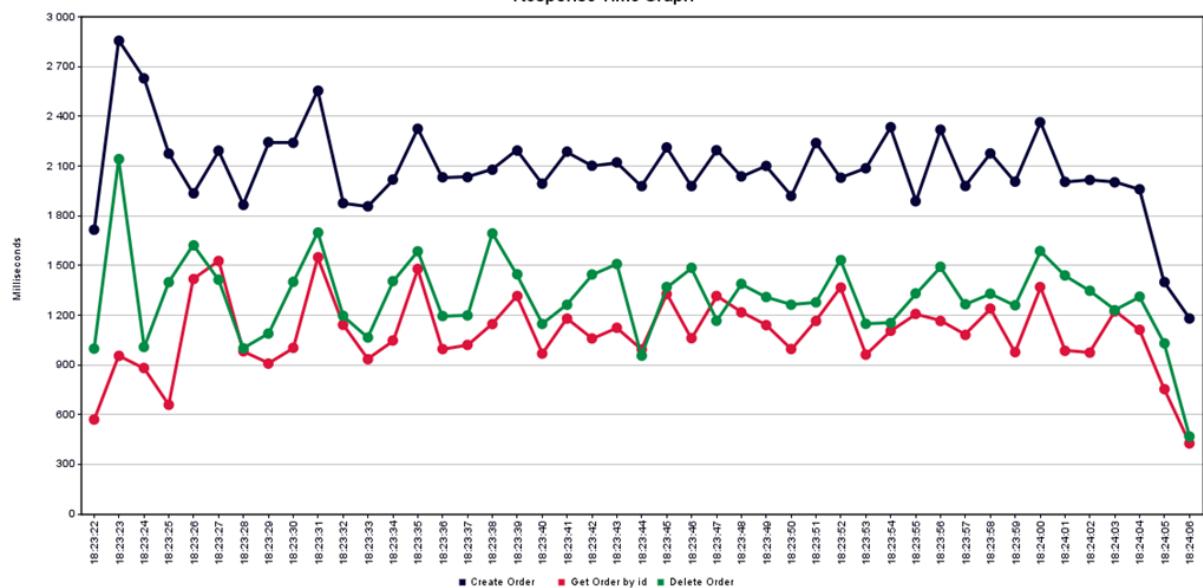
30 пользователей

Код ответа в секунду

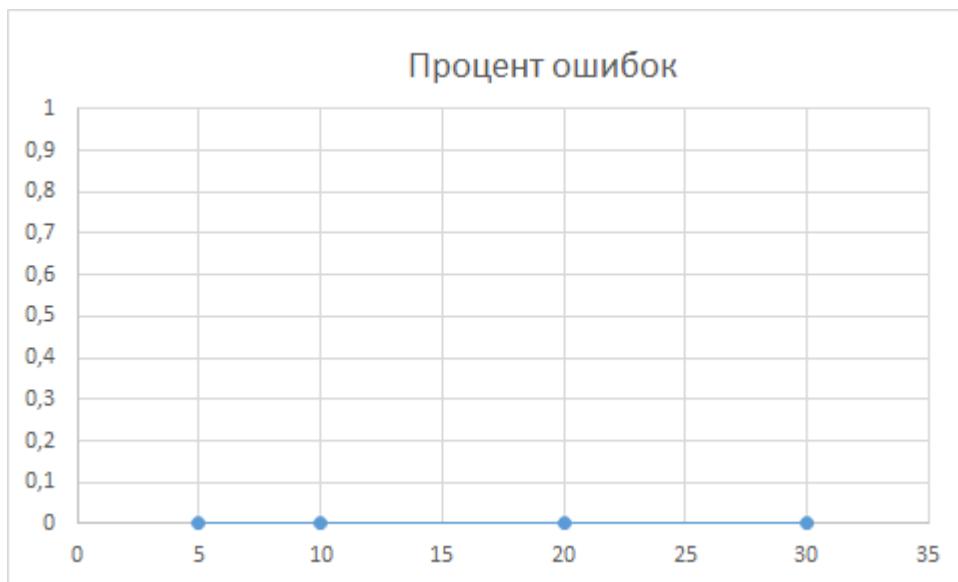
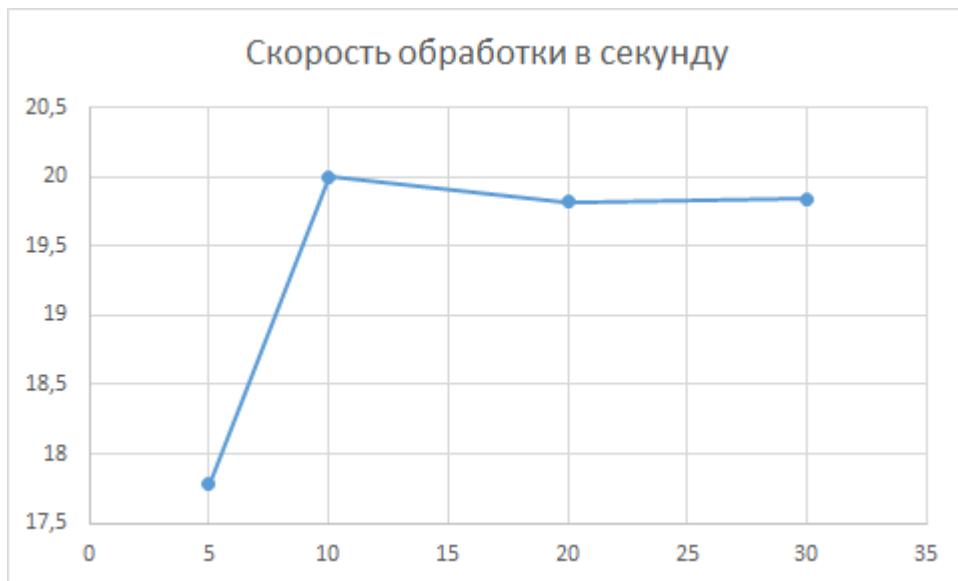


Время ответа по каждому из запросов

Response Time Graph



Сводные таблицы по всему тестированию



Вывод:

Система работает исправно вне зависимости от количества пользователей.

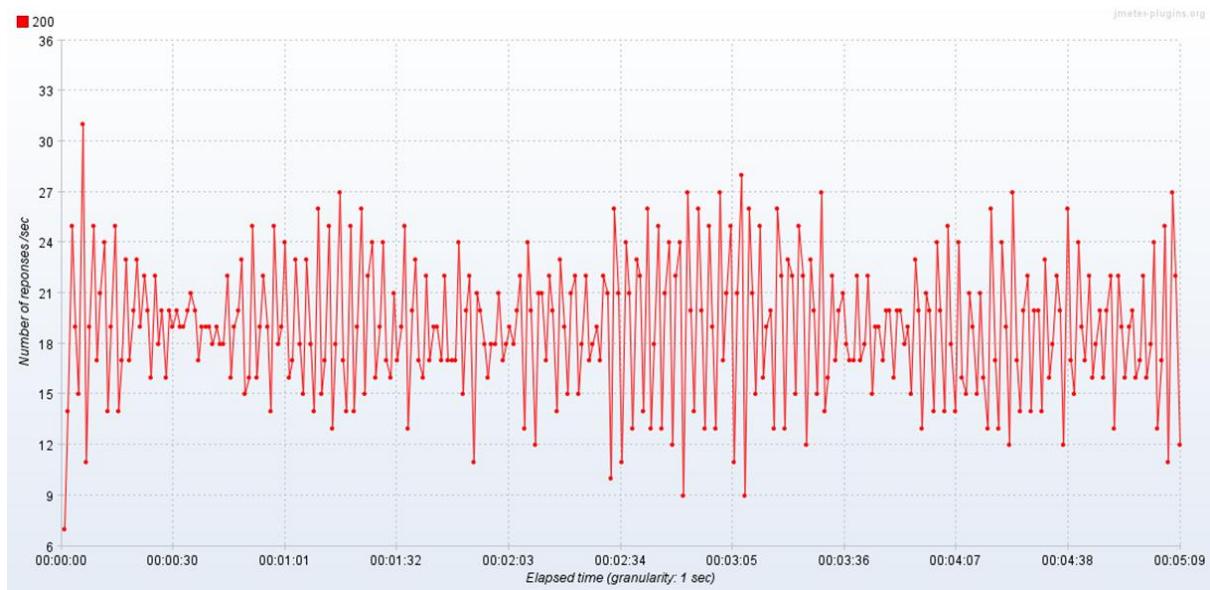
4. Load Testing

Сценарий

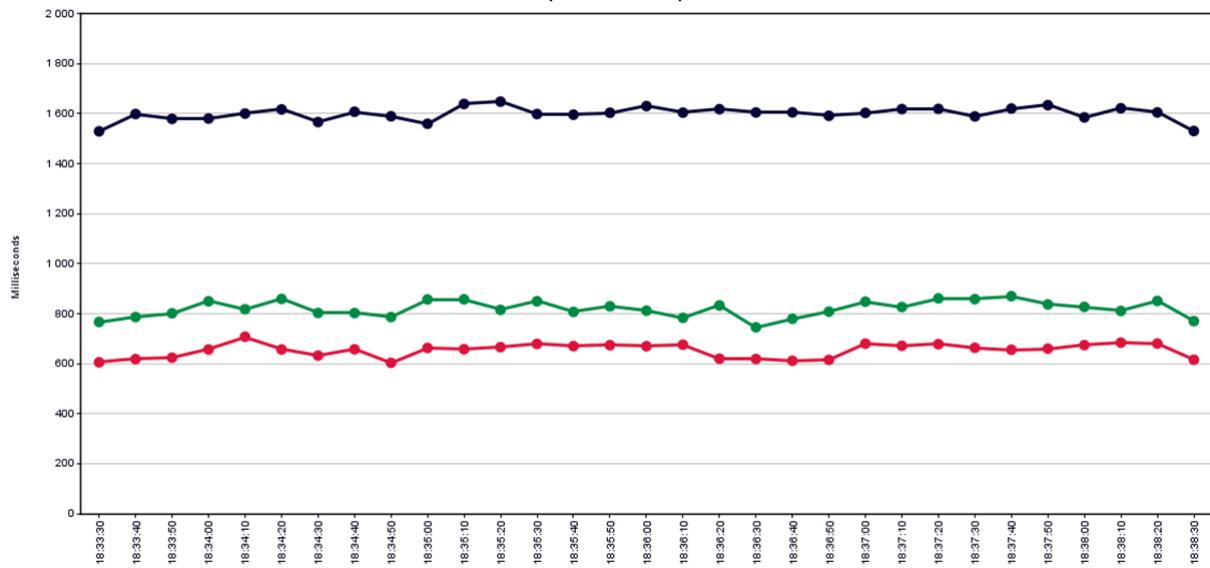
1. Создание заказа
2. Получение заказа
3. Удаление заказа

Проверка была сделана на 20 пользователях, которые непрерывно отправляли запросы в течении 5 минут

Код ответа в секунду



Время ответа по каждому из запросов
Response Time Graph



Вывод: Система справляется с требуемой нагрузкой в течении длительного времени

4. Stress Testing

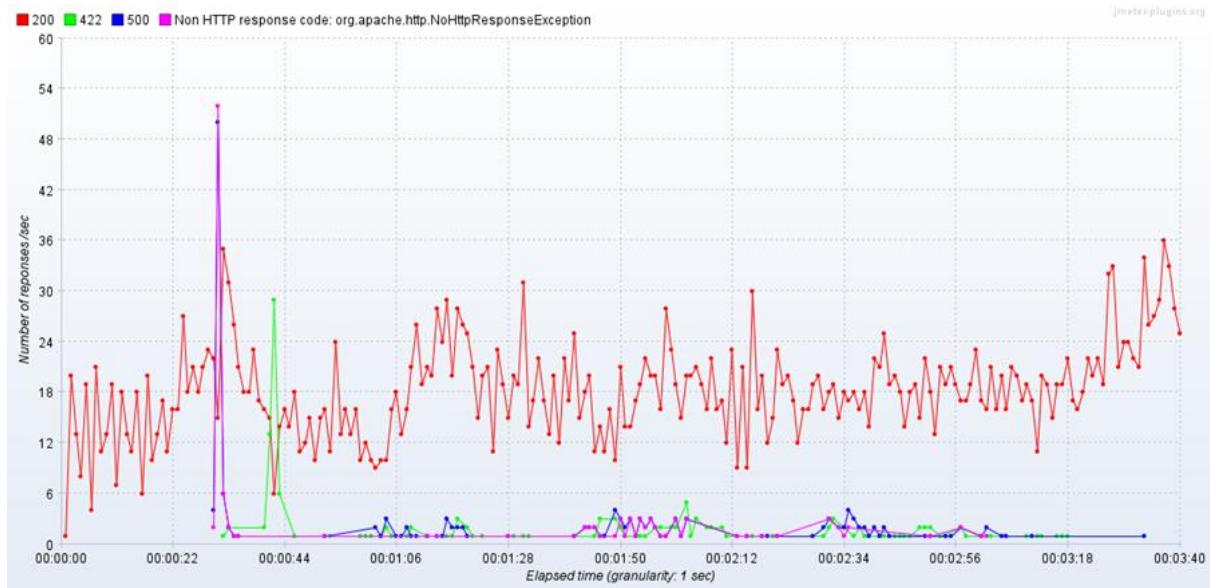
Сценарий

1. Создание заказа
2. Получение заказа
3. Удаление заказа

Проверка была сделана на 100,150,300 и 400 пользователях.

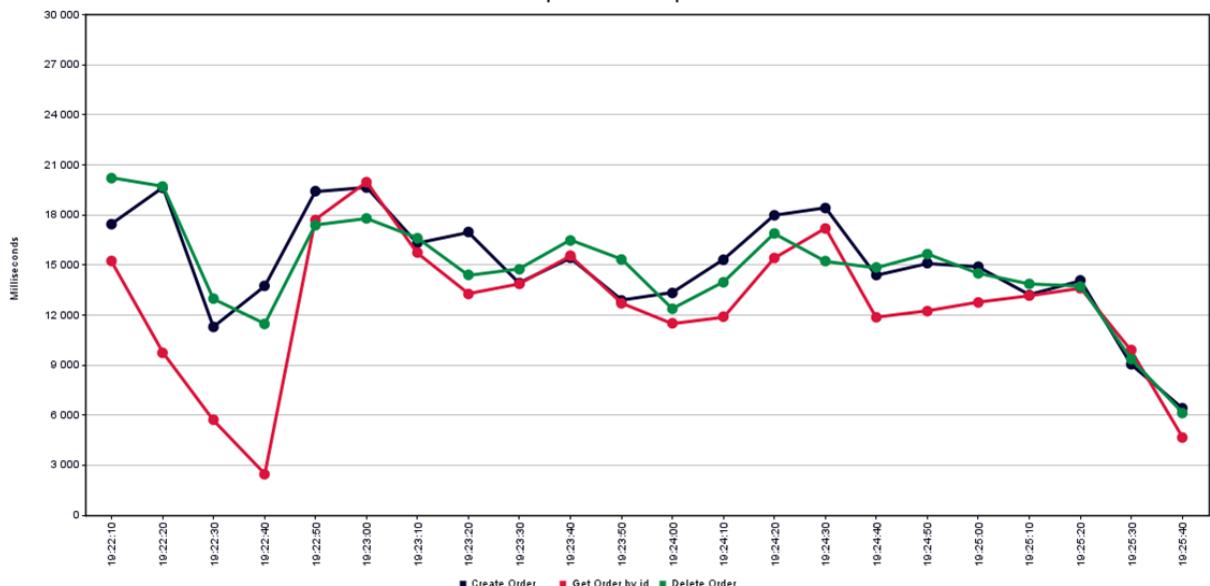
300 пользователей

Код ответа в секунду



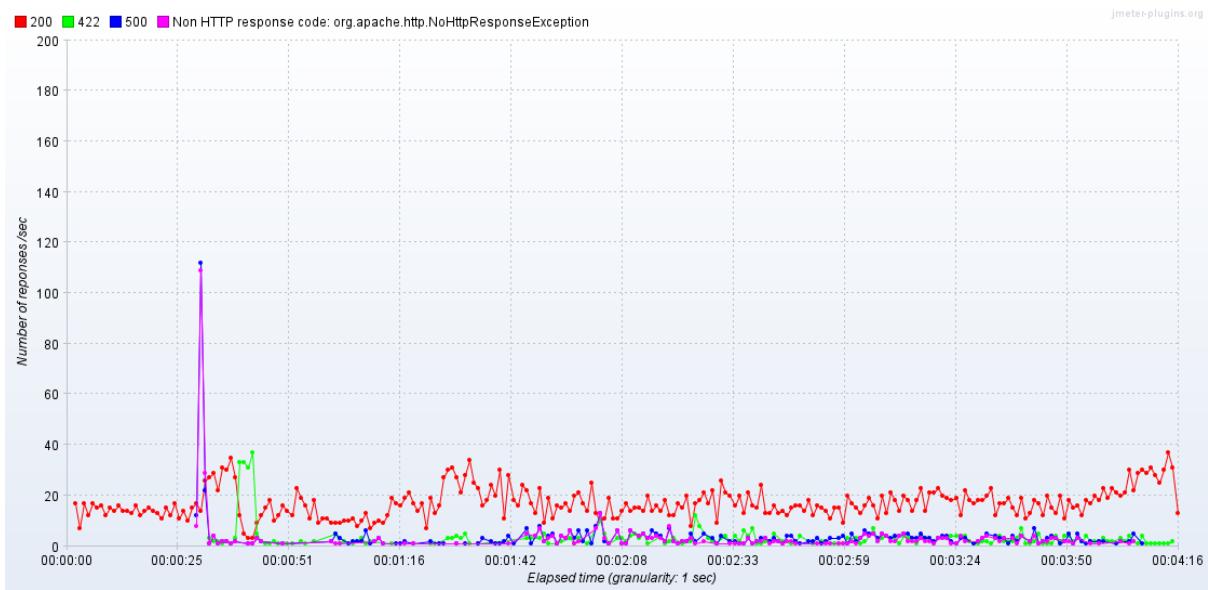
Время ответа по каждому из запросов

Response Time Graph



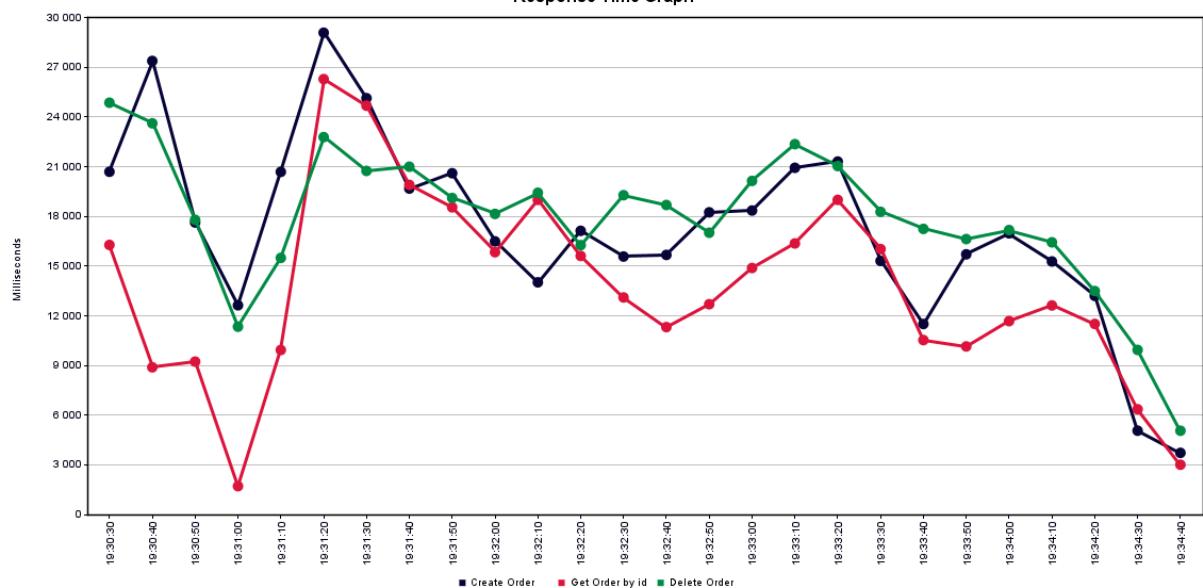
400 пользователей

Код ответа в секунду

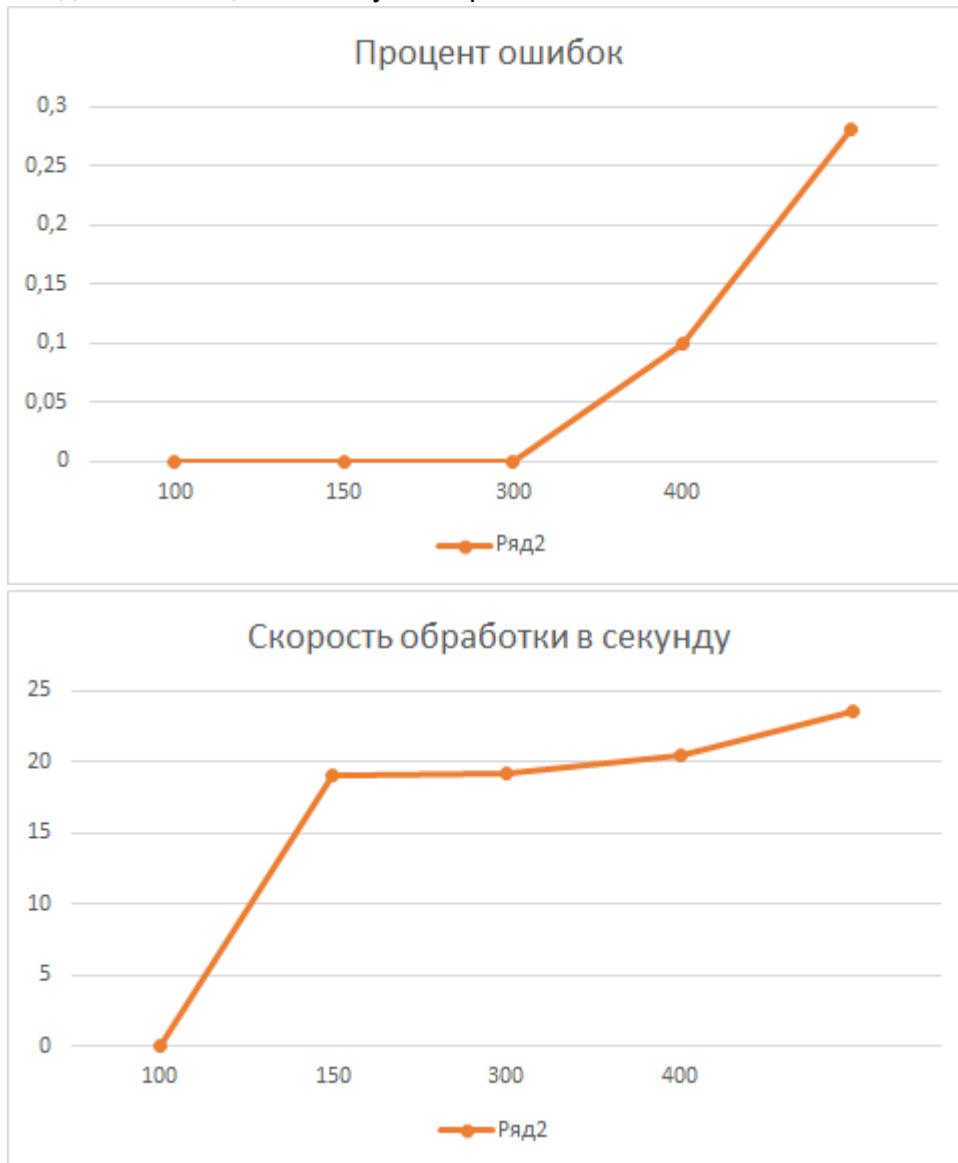


Время ответа по каждому из запросов

Response Time Graph



Сводные таблицы по всему тестированию



Вывод:

Когда количество пользователей стало превышать 150 система перестала справляться с нагрузкой.

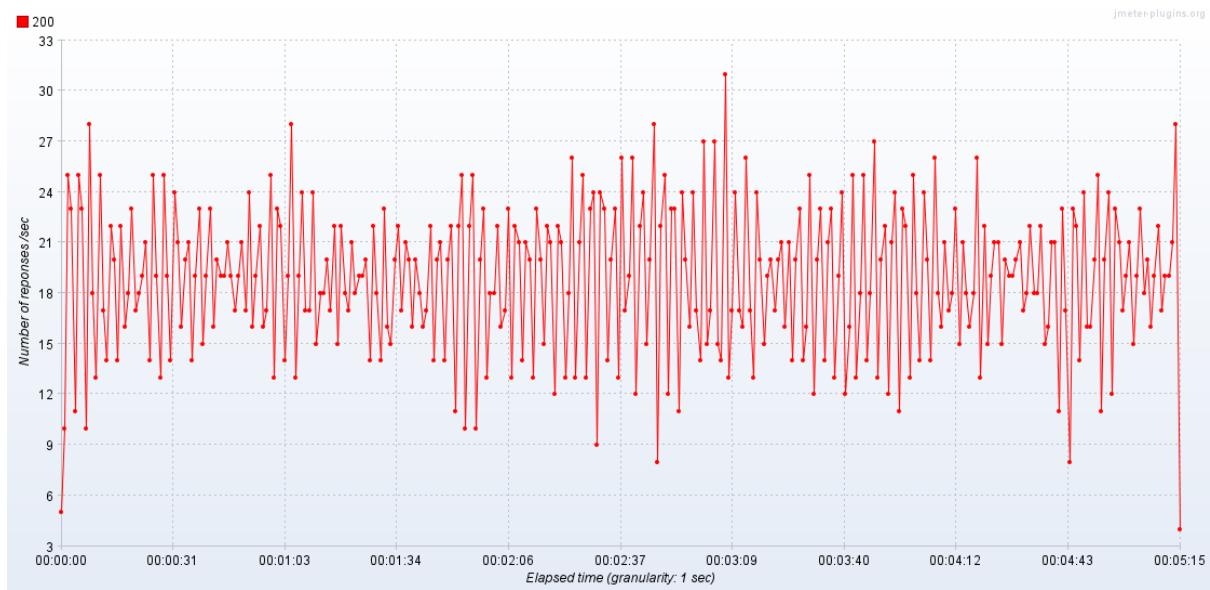
4. Volume Testing

Сценарий

1. Создание заказа
2. Получение заказа
3. Удаление заказа

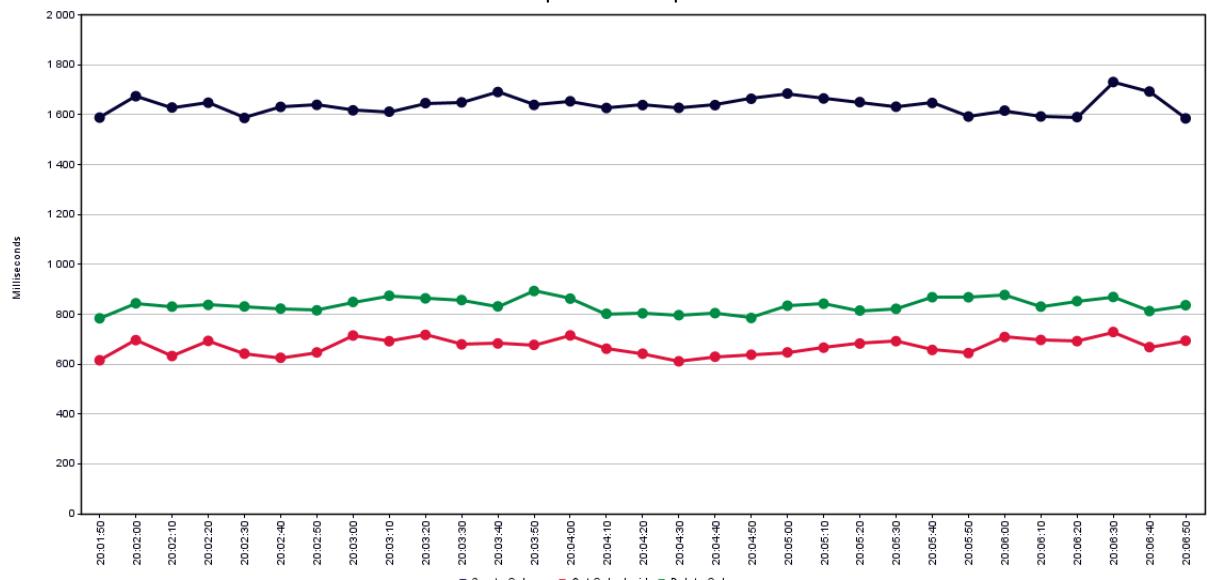
Проверка была сделана на 20 пользователя с заполненной базой в 1000 заказов

Код ответа в секунду



Время ответа по каждому из запросов

Response Time Graph



Вывод:

Система справляется с требуемой нагрузкой вне зависимости от заполнения БД

4. Security and Access Control Testing

1. Проверки доступа к сущностям для разных ролей (пользователь, работник)
2. Проверки смены статуса заявки для разных ролей (менеджер клиентов, менеджер заявок, исполнитель)

Summary

16 tests took 00:00:41.

(Un)check the boxes to filter the results.

0 Failed, 16 Passed, 0 Skipped, 0 Expected failures, 0 Unexpected passes, 0 Errors, 0 Fluruns

Result ▲	Test
Passed	tests/api/test_security.py::test_security_healthcheck
Passed	tests/api/test_security.py::test_customer_can_only_view_orders[user_roles0]
Passed	tests/api/test_security.py::test_customer_can_only_view_orders[user_roles1]
Passed	tests/api/test_security.py::test_customer_can_only_view_orders[user_roles2]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_requests[user_roles0]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_requests[user_roles1]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_requests[user_roles2]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_defects[user_roles0]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_defects[user_roles1]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_defects[user_roles2]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_defects[user_roles3]
Passed	tests/api/test_security.py::test_customer_cannot_work_with_defects[user_roles4]
Passed	tests/api/test_security.py::test_user_cannot_change_status[start_statuses0-new_statuses_updates0-user_roles0]
Passed	tests/api/test_security.py::test_user_cannot_change_status[start_statuses1-new_statuses_updates1-user_roles1]
Passed	tests/api/test_security.py::test_user_cannot_change_status[start_statuses2-new_statuses_updates2-user_roles2]
Passed	tests/api/test_security.py::test_user_cannot_change_status[start_statuses3-new_statuses_updates3-user_roles3]