МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра программных систем\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**  
  
по лабораторной работе №7

«Нагрузочное тестирование»

по дисциплине «Тестирование и отладка ПО»

Обучающаяся в группе 6301-020302D Соколова Алёна

Преподаватель Лобанков Антон Алексеевич

Самара 2024

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc185523089)

[ЗАДАНИЕ 1-2 4](#_Toc185523090)

[ЗАДАНИЕ 3 8](#_Toc185523091)

[ЗАКЛЮЧЕНИЕ 11](#_Toc185523092)

[ИСПОЛЬЗУЕМЫЕ ИСТОЧНИКИ 12](#_Toc185523093)

# ВВЕДЕНИЕ

Целью работы является в процессе выполнения заданий ознакомиться с возможностями jMeter для нагрузочного тестирования приложений.

1. Скачать и запустить jMeter;
2. При помощи функции записи создайте сценарий поиска через yandex.ru сайта Самарского университета и перехода на этот сайт. Подробнее с функцией записи сценария можно ознакомиться в https://testsetup.ru/jmeter/record.php. Удалите из сценария лишние запросы (скриншоты сценария до и после удаления должны быть представлены в отчете). Если в сценарии отсутствуют инструменты аналитики – добавьте их вручную. Запустите сценарий с параметрами «number of threads» = 5, «ramp up period» = 30. Объясните график и результаты, полученные в отчете;
3. Вручную создайте сценарий для тестирования конечно точки программного интерфейса зоомагазина для получения списка питомцев https://petstore.swagger.io/#/pet/findPetsByStatus. Добавьте инструменты анализа в ваш сценарий. Запустите сценарий при работе одного пользователя. Прокомментируйте полученные графики и Summary report. Запустите сценарий при работе 50 пользователей. Прокомментируйте полученные графики и Summary report.

# ЗАДАНИЕ 1-2

Скачан Apache JMeter с официального сайта, распакован архив, приложение запущено через ApacheJMeter.jar.

Создание сценария поиска при помощи записи сценария в Jmeter. Полный список HTTP запросов приведен на рисунке 1.

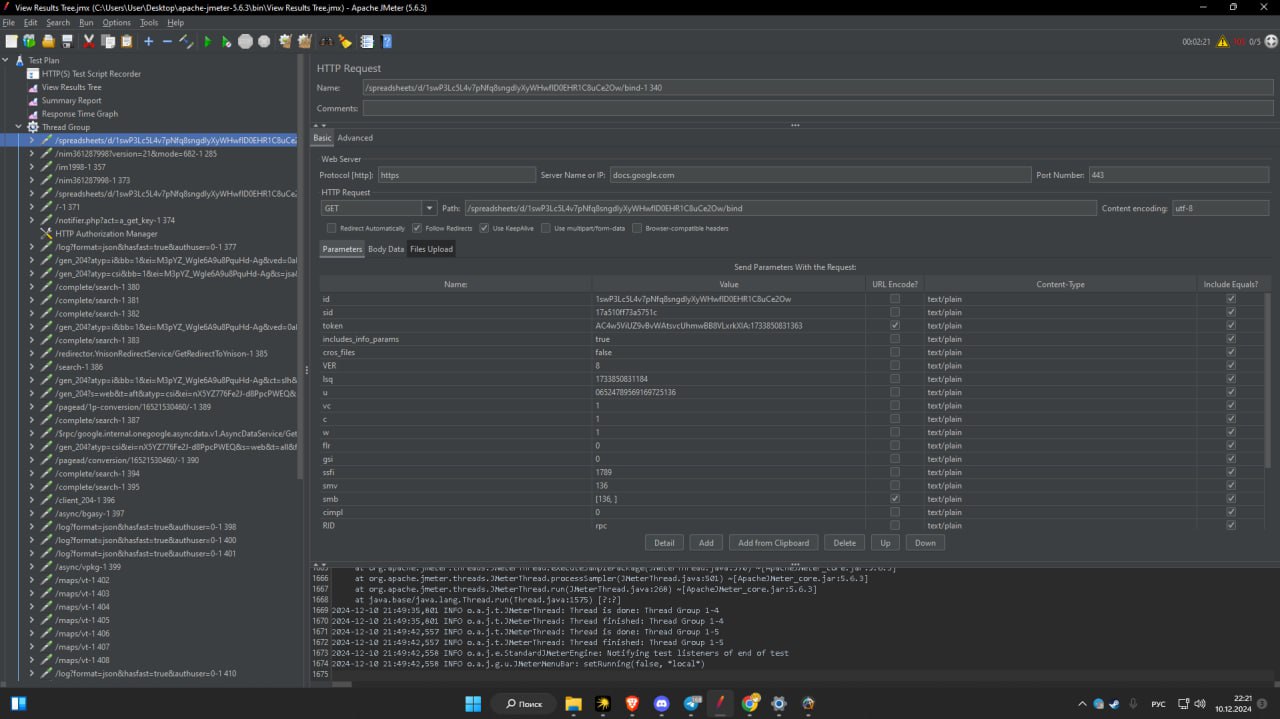


Рисунок 1 – Сценарий поиска

Была проведена очистка данных, в результате которой были удалены большинство лишних запросов. Результат после очистки отображен на рисунке 2.

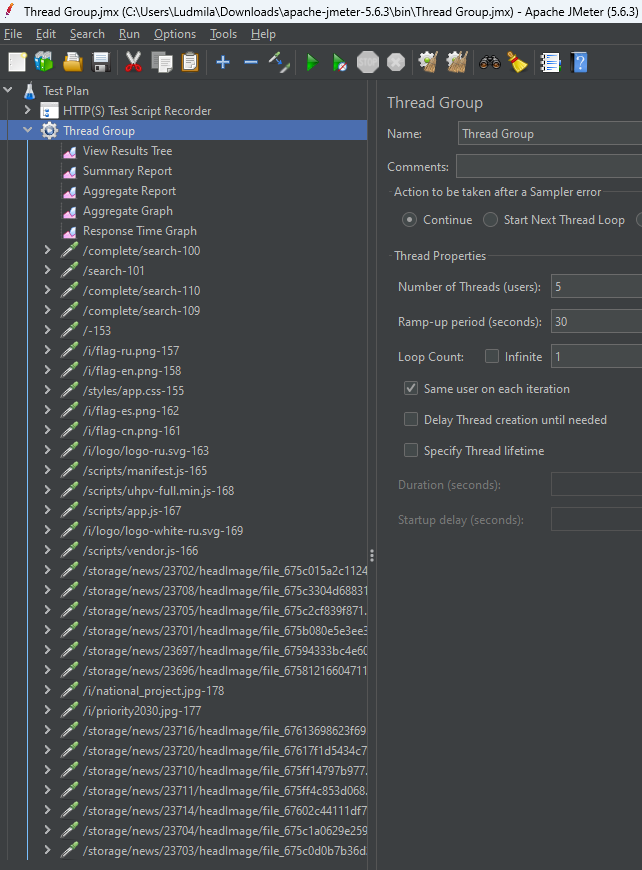


Рисунок 2 – Результат после очистки

При выполнении очищенного сценария с «users» = 5, «period» = 30 открываем инструменты аналитики Aggregate Report, Aggregate Graph и Response Time Graph, откуда будут рассматриваться значения отклонения и пропускной способности. Результаты отображены на рисунках 3, 4 и 5.

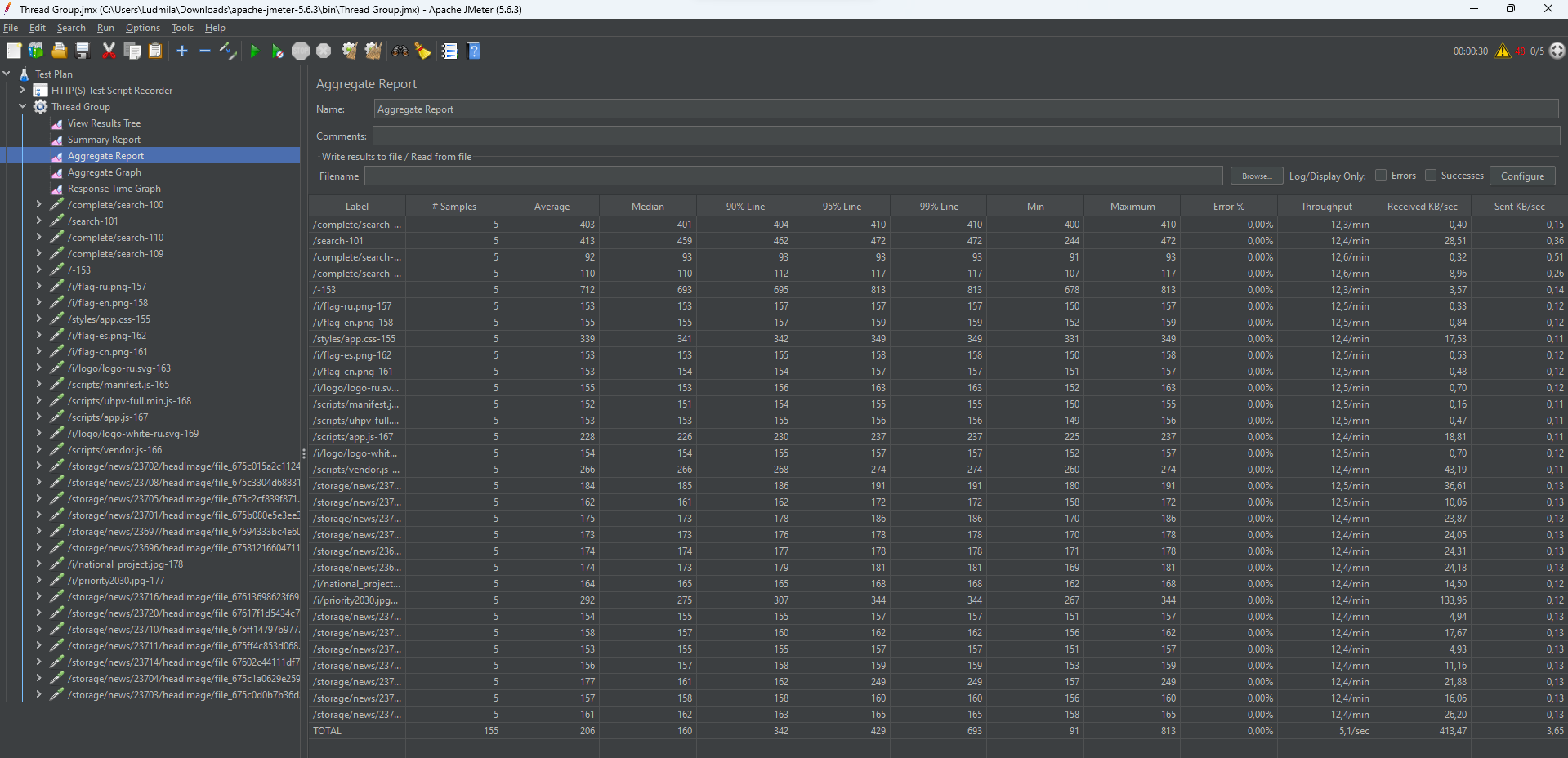


Рисунок 3 – Результат Aggregate Report с «users» = 5, «period» = 30

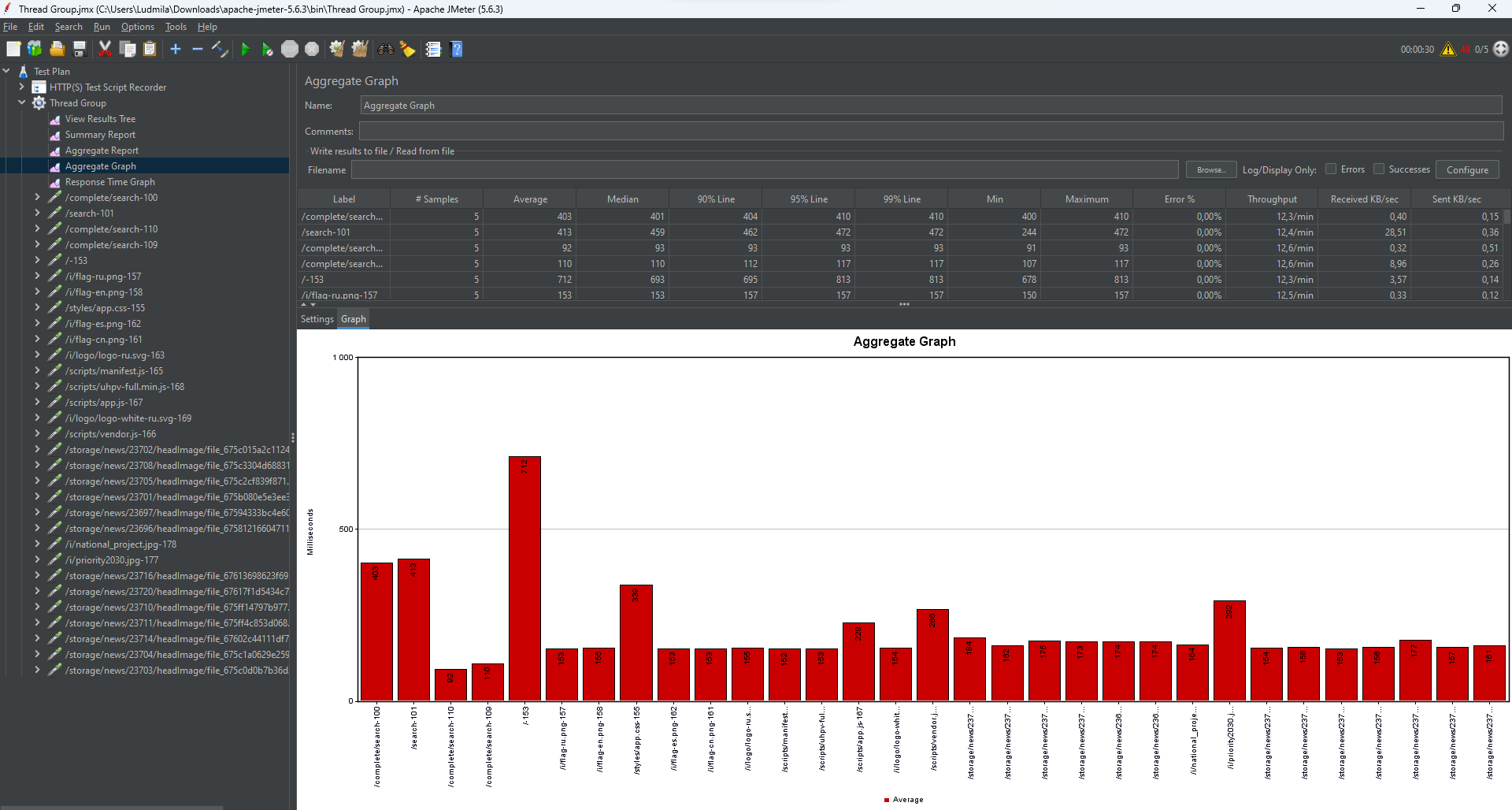


Рисунок 4 – Результат Aggregate Graph с «users» = 5, «period» = 30

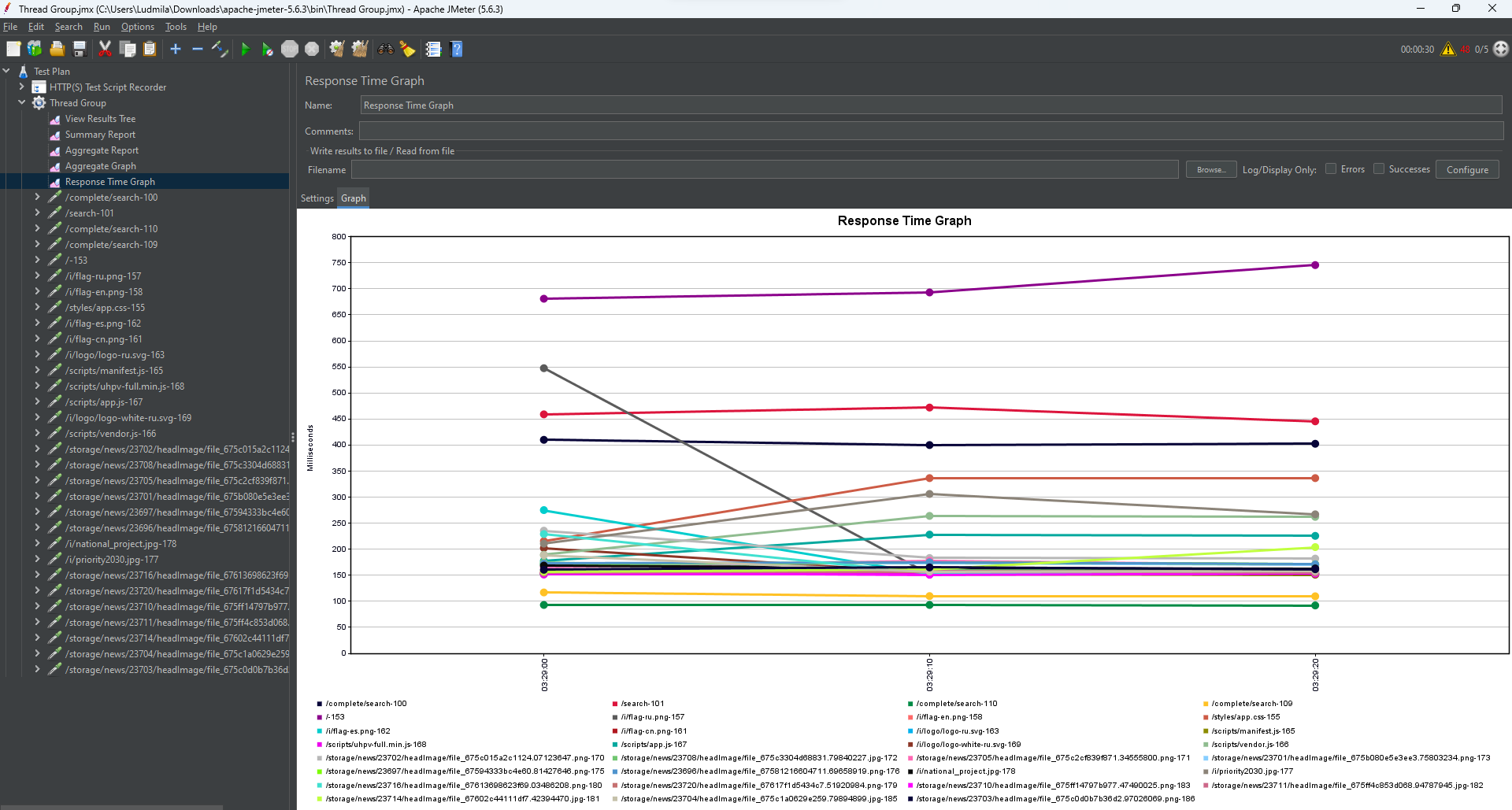


Рисунок 5 – Результат Response Time Graph с «users» = 5, «period» = 30

Среднее время отклика (206 мс) относительно высокое, что указывает на умеренную производительность сервера при текущей нагрузке. Несмотря на это, сервер обрабатывает запросы с небольшой задержкой, что может быть нормальным для текущего уровня нагрузки.

Минимальное время отклика составляет 91 мс, а максимальное – 813 мс. Отклонение в 722 мс указывает на наличие некоторого разброса во времени отклика, что может свидетельствовать о незначительных колебаниях производительности, но в целом разница не критична. Такие вариации могут быть вызваны различными факторами, такими как нагрузка на сервер или внешние воздействия, однако они не вызывают значительных проблем.

Суммарное количество выборок и средней пропускной способности можно оценить как нормальные для указанной нагрузки. Важно отметить, что, несмотря на некоторые колебания в максимальных значениях, сервер продолжает стабильно обрабатывать запросы в рамках допустимых параметров. В целом, производительность сервера выглядит приемлемой для текущего теста.

# ЗАДАНИЕ 3

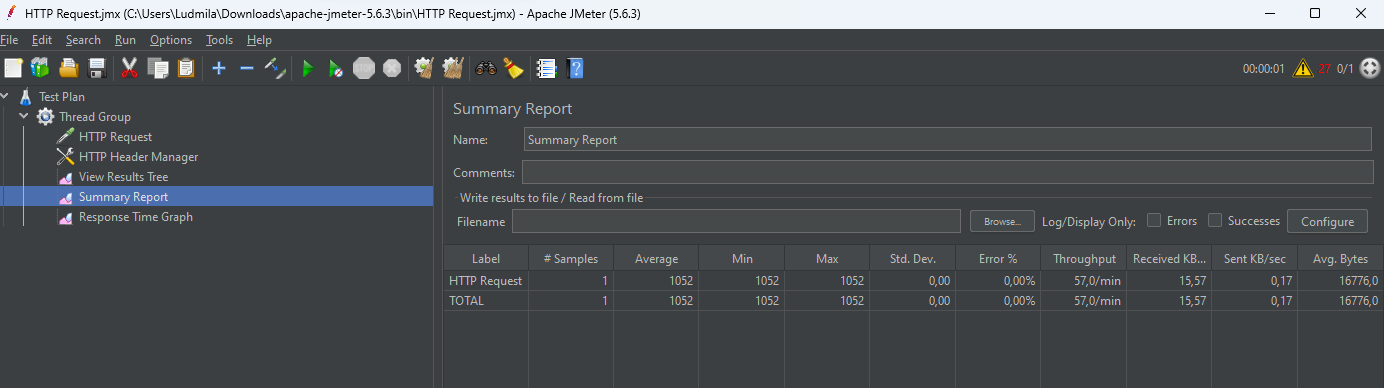


Рисунок 6 – Результат Summary Report с «users» = 1, «period» = 1

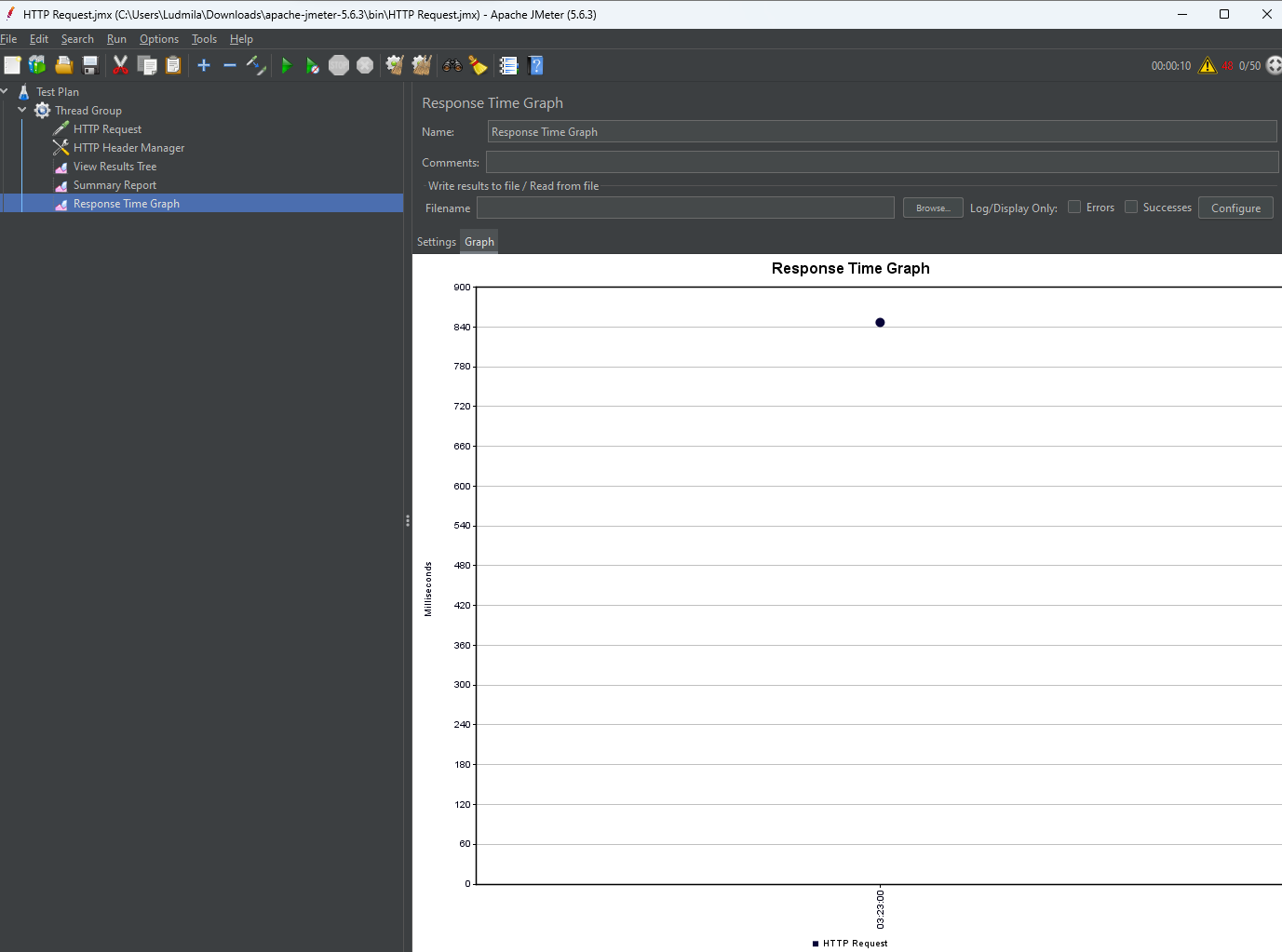


Рисунок 7 – Результат Graph Results с «users» = 50, «period» = 1

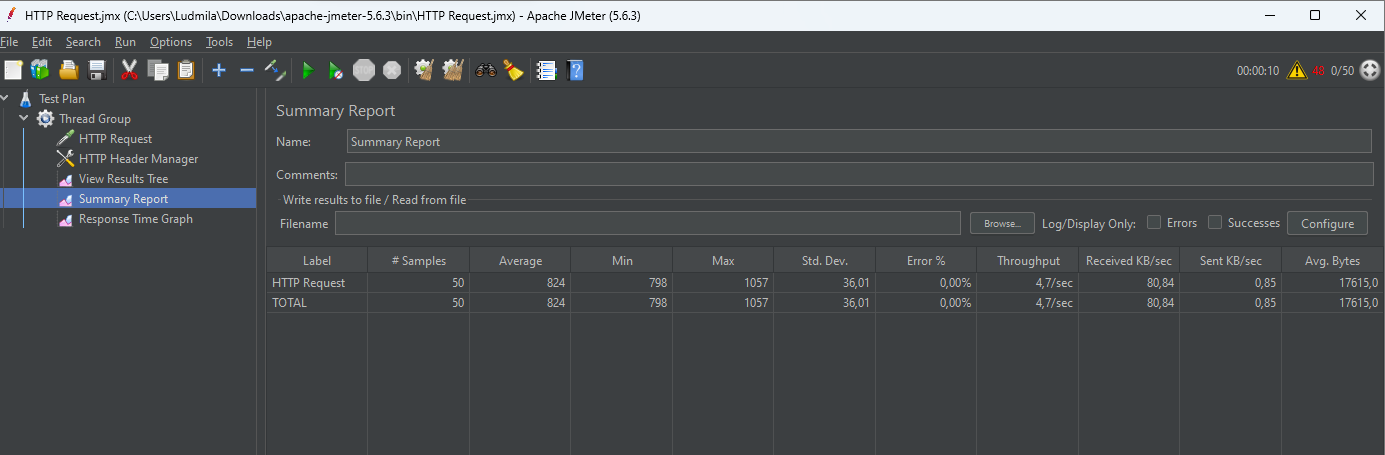


Рисунок 8 – Результат Summary Report с «users» = 50, «period» = 1

Среднее время отклика (824 мс) улучшилось по сравнению с тестом одного пользователя (1052 мс → 824 мс). Это может говорить о том, что сервер справляется с увеличением нагрузки, при этом время отклика не ухудшается значительно, что указывает на хорошую производительность.

Минимальное время отклика составляет 758 мс, а максимальное – 1057 мс. Отклонение от среднего времени в мс указывает на вариативность в обработке запросов, что характерно для тестов с несколькими запросами, но такие колебания не являются критичными.

Сервер способен обрабатывать 50 запросов в тесте, что свидетельствует о стабильной пропускной способности и хорошей производительности при текущем уровне нагрузки.

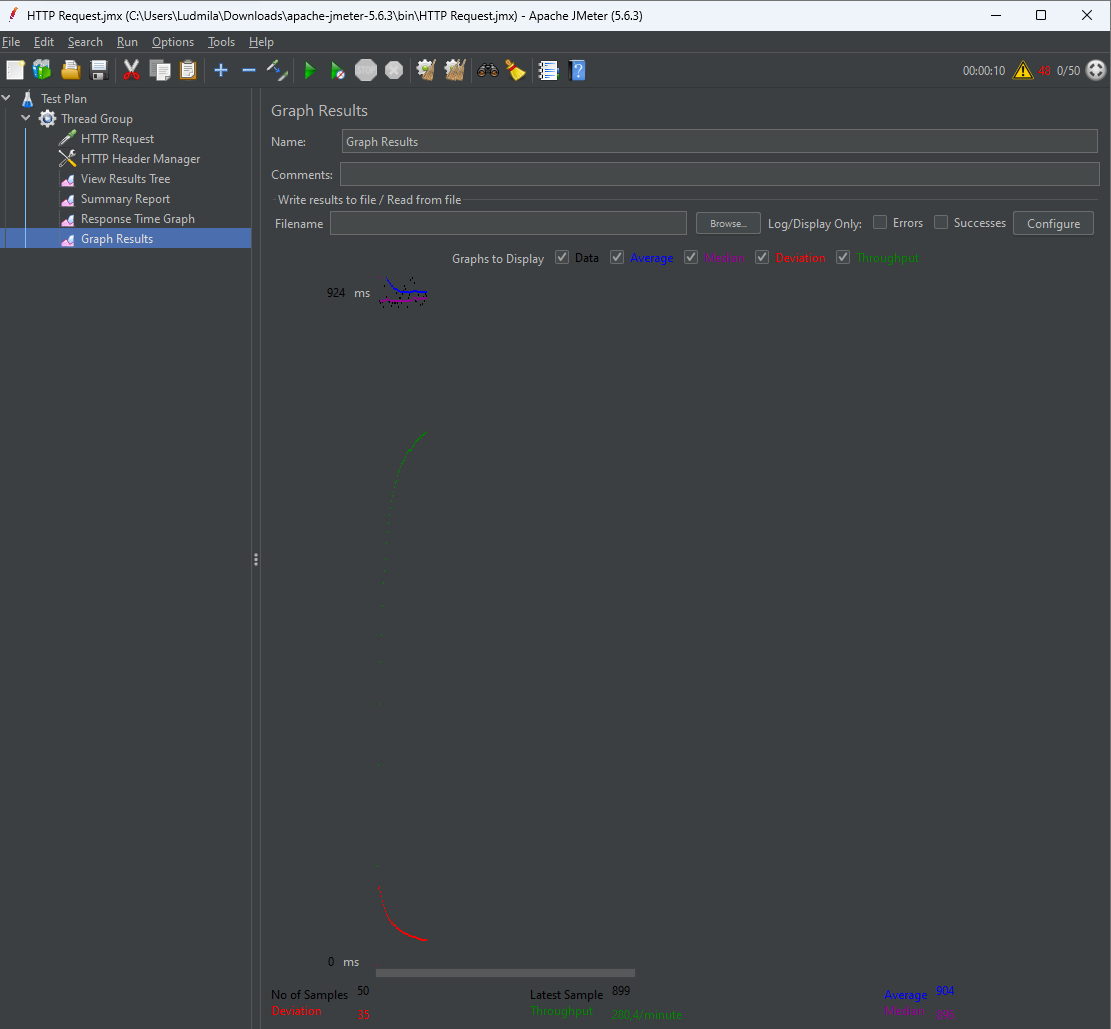


Рисунок 9 – Результат Graph Results с «users» = 50, «period» = 1

Исходя из графика, среднее время отклика для сценария составляет 904 мс. Медианное значение и стандартное отклонение равны 896 и 35 соответственно, что указывает на нестабильность времени отклика. График показывает, что время отклика быстро снижается в начале теста (вероятно, из-за периода Ramp-up period), а затем стабилизируется. Пропускная способность системы около 281 запросов в минуту, что является достаточно хорошим показателем и показывает, что система способна обрабатывать значительное количество одновременных запросов без серьезных задержек.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторных заданий был изучен инструмент jMeter для нагрузочного тестирования веб-приложений. В процессе работы с jMeter были созданы сценарии для тестирования поиска через сайт Самарского университета и для взаимодействия с API зоомагазина. В результате использования функции записи сценариев удалось эффективно настроить сценарий для поиска и перехода на сайт, а также удалить лишние запросы для оптимизации. Дополнительно были добавлены инструменты аналитики для более точного анализа работы системы. Проведенное тестирование с различным количеством пользователей позволило получить графики и отчеты, на основе которых было возможно оценить производительность системы и выявить потенциальные узкие места. Результаты тестирования способствовали лучшему пониманию работы приложения под нагрузкой, что позволяет оптимизировать его производительность.

# ИСПОЛЬЗУЕМЫЕ ИСТОЧНИКИ

1. Лекция 6. Нагрузочное тестирование. Лобанков А.А.
2. Запись теста JMeter [Электронный ресурс] / URL: https://testsetup.ru/jmeter/record.php. (дата обращения: 12.12.2024).