

Национальный исследовательский университет ИТМО

Факультет ПИиКТ

Лабораторная работа №4

по дисциплине
«Тестирование программного обеспечения»

Вариант №336767

Работу выполнили:

Тройникова В.Д.

Лушникова А.С.

Группа: Р33302

Преподаватель:

Гаврилов А.В.

Санкт-Петербург

2024

Текст задания

С помощью программного пакета Apache JMeter провести нагрузочное и стресс-тестирование веб-приложения в соответствии с вариантом задания.

В ходе нагрузочного тестирования необходимо протестировать 3 конфигурации аппаратного обеспечения и выбрать среди них наиболее дешёвую, удовлетворяющую требованиям по максимальному времени отклика приложения при заданной нагрузке (в соответствии с вариантом).

В ходе стресс-тестирования необходимо определить, при какой нагрузке выбранная на предыдущем шаге конфигурация перестаёт удовлетворять требованиям по максимальному времени отклика. Для этого необходимо построить график зависимости времени отклика приложения от нагрузки.

- Приложение для тестирования доступно только во внутренней сети кафедры.
- Если запрос содержит некорректные параметры, сервер возвращает HTTP 403.
- Если приложение не справляется с нагрузкой, сервер возвращает HTTP 503.

Параметры тестируемого веб-приложения:

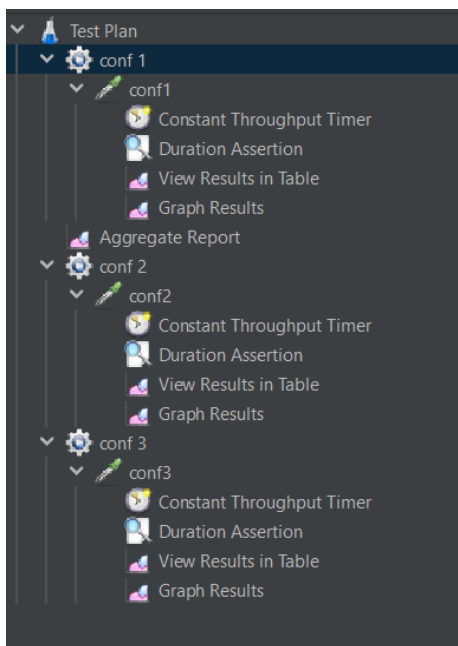
- URL первой конфигурации (\$ 5900)
`http://stload.se.ifmo.ru:8080?token=492557781&user=2129854793&config=1;`
- URL второй конфигурации (\$ 7200)
`http://stload.se.ifmo.ru:8080?token=492557781&user=2129854793&config=2;`
- URL третьей конфигурации (\$ 12800)
`http://stload.se.ifmo.ru:8080?token=492557781&user=2129854793&config=3;`
- Максимальное количество параллельных пользователей - **14**;
- Средняя нагрузка, формируемая одним пользователем - **20 запр. в мин.**;
- Максимально допустимое время обработки запроса - **850 мс.**

Подключение к серверу

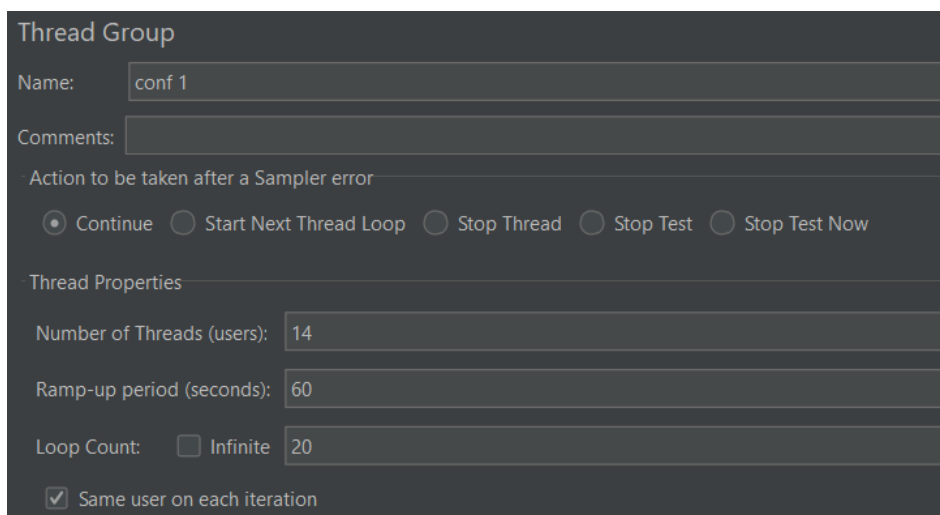
ssh -L 8083:stload.se.ifmo.ru:8080 -p 2222 -N s336767@helios.se.ifmo.ru

Описание конфигурации JMeter для нагрузочного тестирования

Получившийся тестовый план:



В компоненте Thread Group мы зададим количество потоков (пользователей) - согласно варианту их должно быть 14. Для loop count - количество повторений тестового сценария - устанавливаем значение 20 (так как по заданию каждый пользователь генерирует нагрузку 20 запросов в минуту). Ramp-up period - это время, в течение которого количество пользователей увеличится с 0 до 14. Выбрано время 60 секунд, так как при таком периоде нагрузка увеличивается равномерно и при этом первые потоки не успевают завершиться к моменту начала последних.



Компонент HTTP Request настраиваем в соответствии с вариантом:

HTTP Request

Name:

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP: Port Number:

HTTP Request

Path: Content encoding:

☐ Redirect Automatically ☒ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data ☐ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?
token	492557781	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
user	2129854793	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
config	1	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Detail Add Add from Clipboard Delete Up Down

Компонент Constant Throughput Timer необходим для регулировки частоты отправки запросов на сервер - мы задаем параметр равным 20, чтобы jMeter регулировал кол-во исходящих запросов и создал нагрузку 20 запросов в минуту.

Constant Throughput Timer

Name:

Comments:

Delay before each affected sampler

Target throughput (in samples per minute):

Calculate Throughput based on:

Компонент Duration Assertion осуществляет проверку длительности ответа сервера - в соответствии с вариантом 850 мс.

Duration Assertion

Name:

Comments:

Apply to:

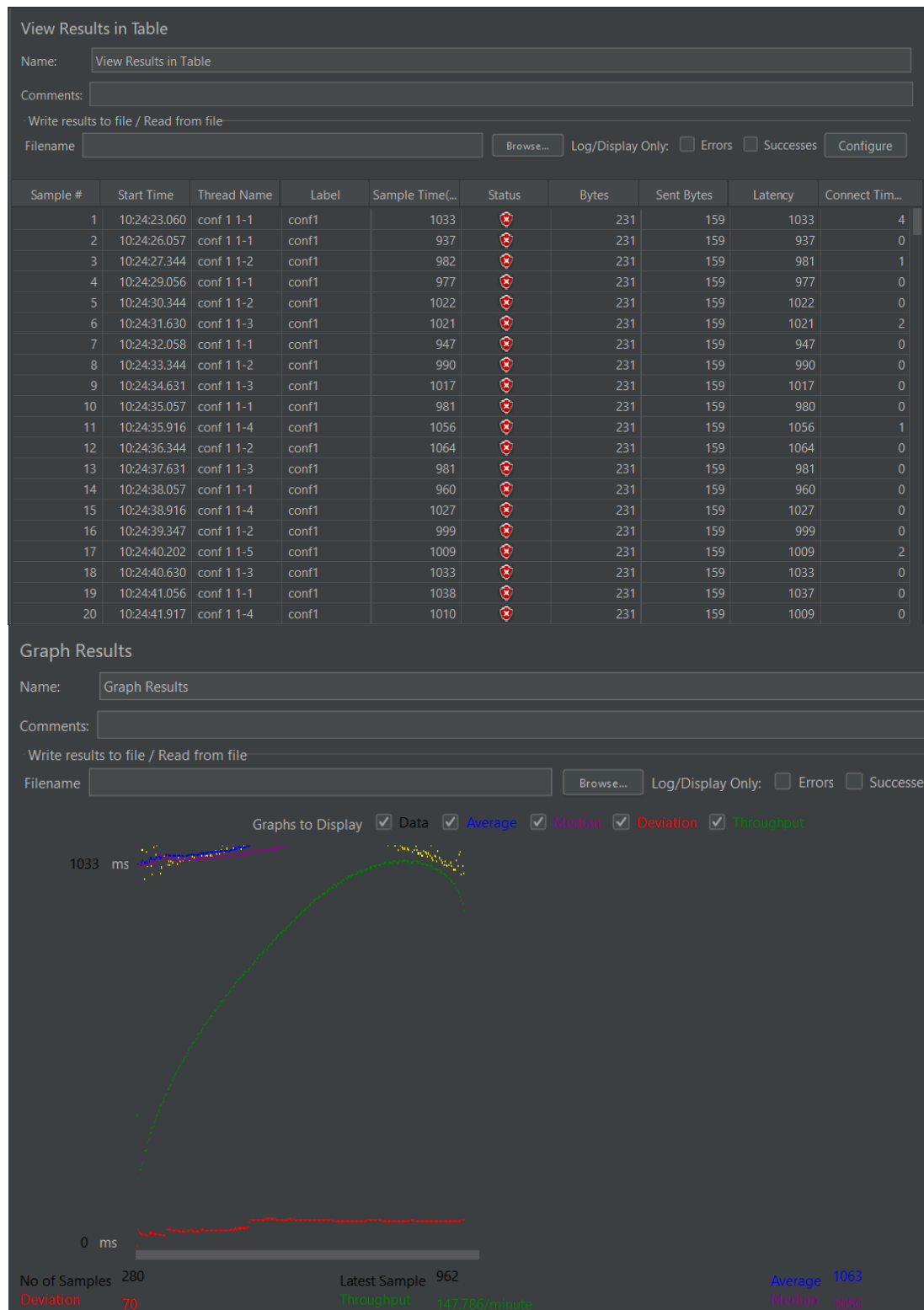
☐ Main sample and sub-samples ☒ Main sample only ☐ Sub-samples only

Duration to Assert

Duration in milliseconds:

Graph Results и View Results in Table компоненты отвечают за отображение результатов.

Первая конфигурация даже при небольшом кол-ве пользователей не справляется с требуемым временем ответа:



Вторая конфигурация при небольшом кол-ве пользователей справляется с требуемым временем ответа, однако при увеличении кол-ва пользователей данная конфигурация также не справляется с требуемыми параметрами:

View Results in Table

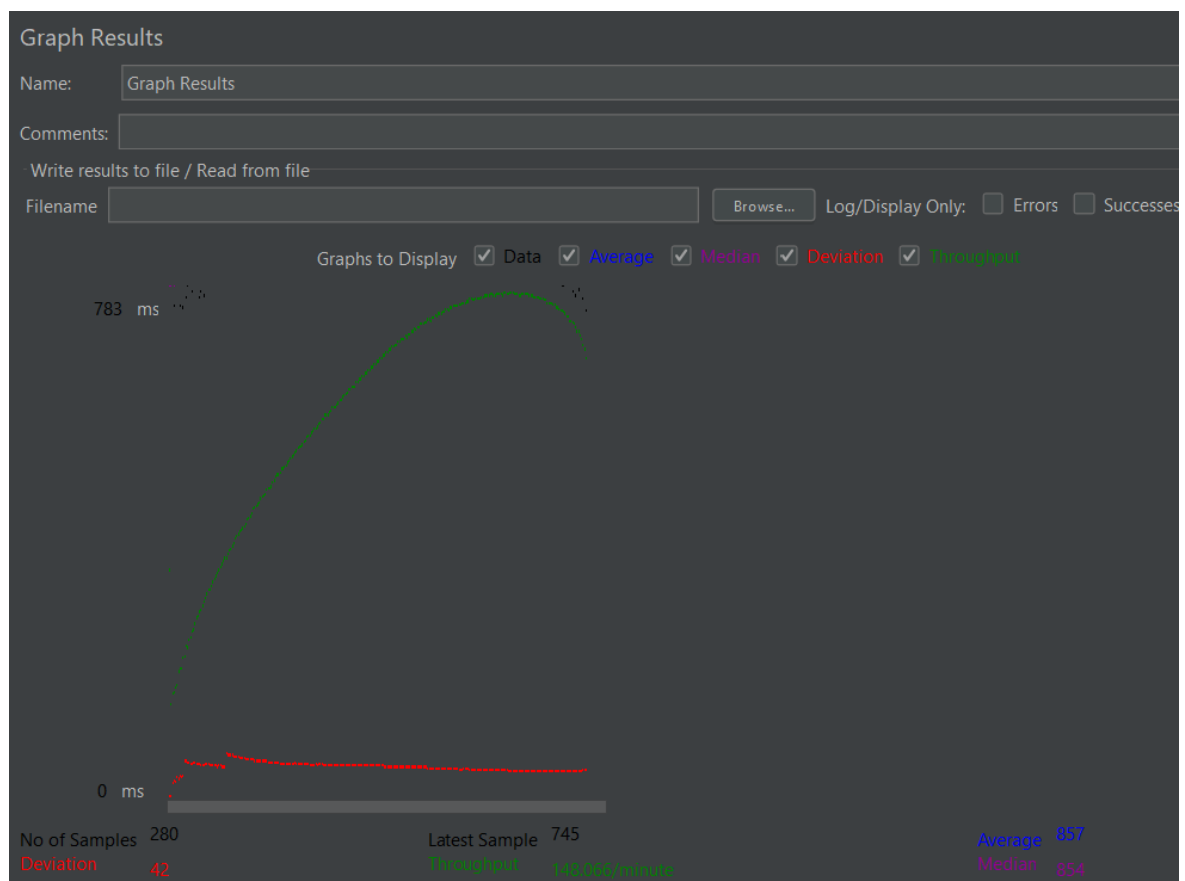
Name:

Comments:

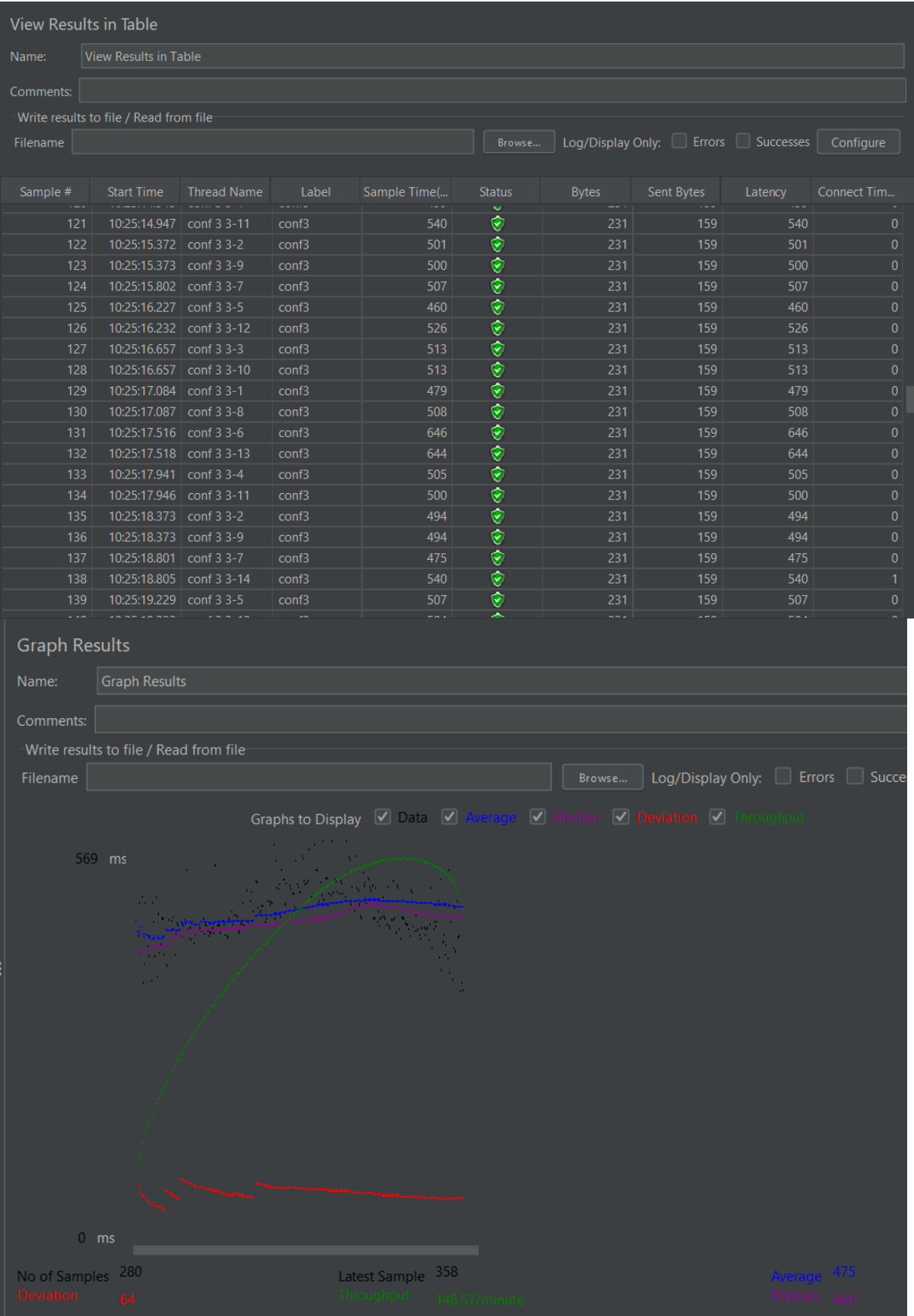
Write results to file / Read from file

Filename Log/Display Only: ☐ Errors ☐ Successes

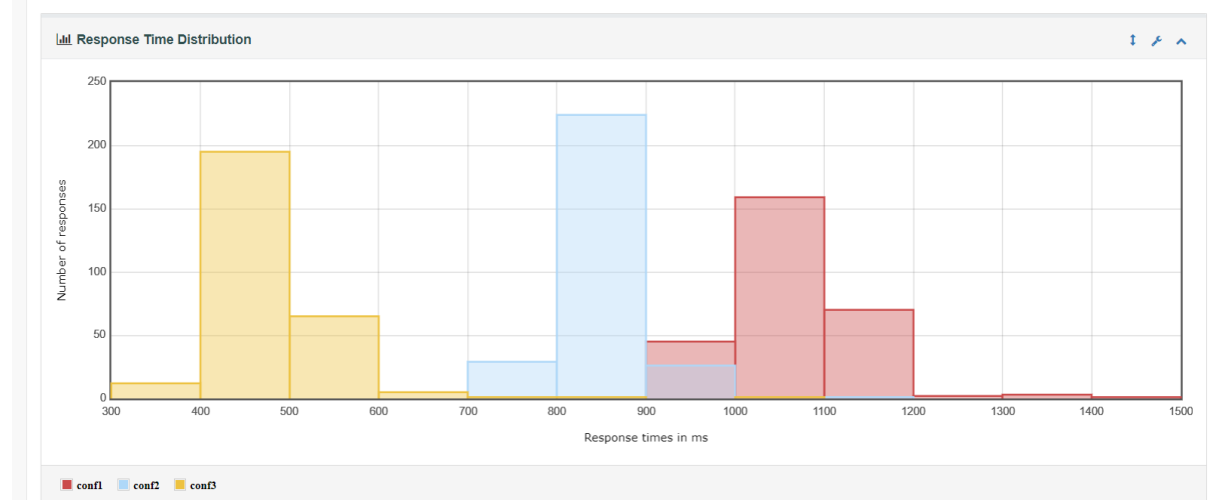
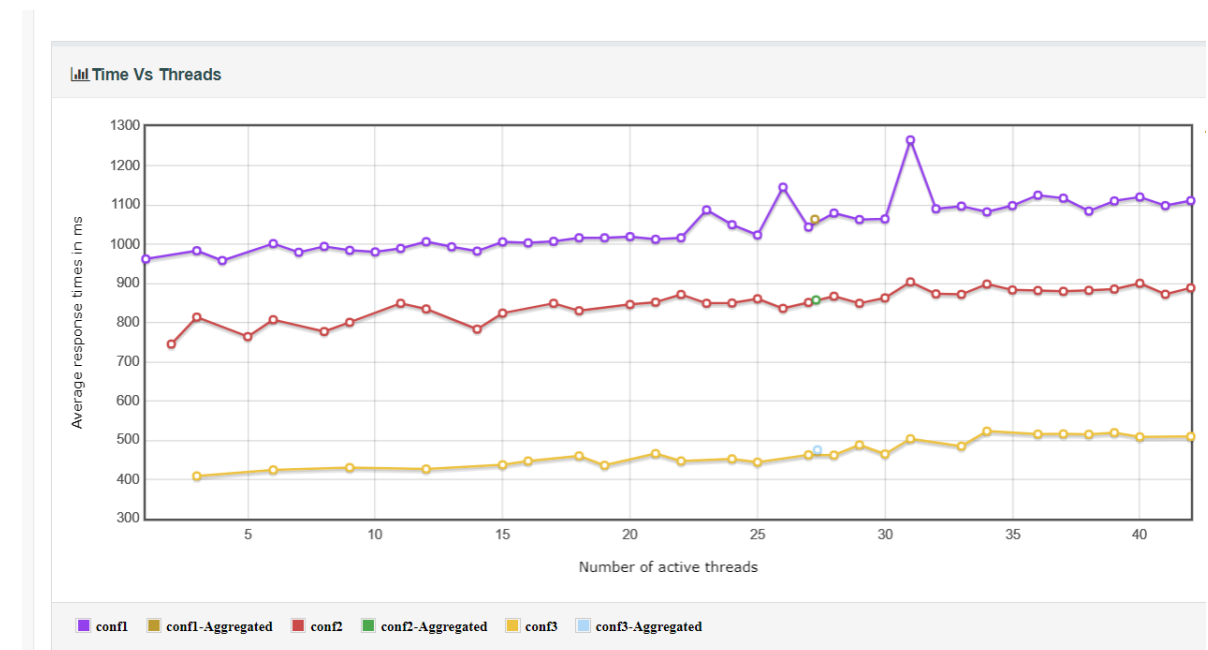
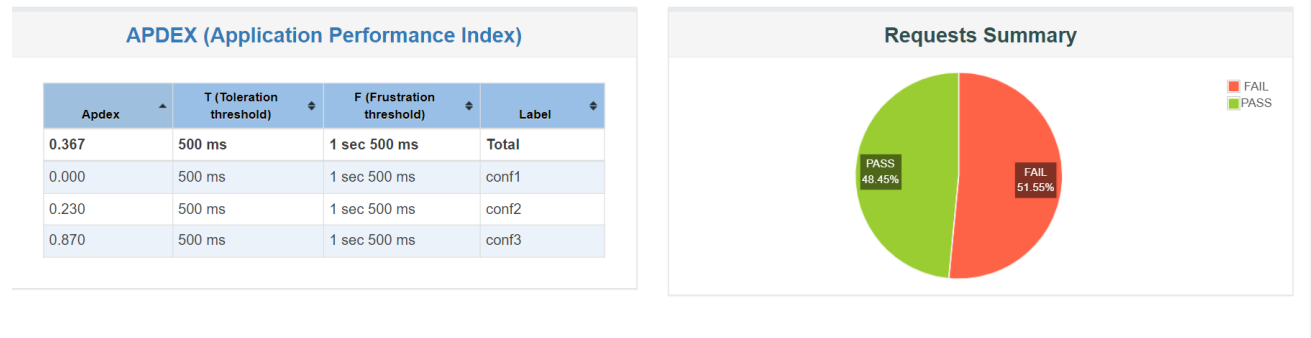
Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Tim...
117	10:25:14.074	conf 2 2-8	conf2	872		231	159	872	0
118	10:25:14.501	conf 2 2-6	conf2	860		231	159	860	0
119	10:25:14.505	conf 2 2-13	conf2	982		231	159	982	1
120	10:25:14.932	conf 2 2-11	conf2	857		231	159	857	0
121	10:25:14.932	conf 2 2-4	conf2	887		231	159	887	0
122	10:25:15.358	conf 2 2-2	conf2	859		231	159	859	0
123	10:25:15.360	conf 2 2-9	conf2	885		231	159	885	0
124	10:25:15.788	conf 2 2-7	conf2	861		231	159	861	0
125	10:25:16.216	conf 2 2-5	conf2	855		231	159	855	0
126	10:25:16.220	conf 2 2-12	conf2	879		231	159	879	0
127	10:25:16.646	conf 2 2-3	conf2	857		231	159	857	0
128	10:25:16.646	conf 2 2-10	conf2	885		231	159	885	0
129	10:25:17.076	conf 2 2-8	conf2	872		231	159	872	0
130	10:25:17.073	conf 2 2-1	conf2	938		231	159	938	0
131	10:25:17.501	conf 2 2-6	conf2	866		231	159	866	0
132	10:25:17.506	conf 2 2-13	conf2	894		231	159	894	0
133	10:25:17.933	conf 2 2-11	conf2	867		231	159	867	0
134	10:25:17.933	conf 2 2-4	conf2	880		231	159	880	0
135	10:25:18.359	conf 2 2-9	conf2	870		231	159	870	0
136	10:25:18.358	conf 2 2-2	conf2	882		231	159	882	0



Третья конфигурация поддерживает требуемое время ответа сервера даже при большом кол-ве пользователей:



Графики пропускной способности приложения, полученные в ходе нагрузочного тестирования.





Выводы по выбранной конфигурации аппаратного обеспечения.

По результатам тестирования наиболее подходящая конфигурация - это конфигурация под номером 3. Только она при требуемых 14 пользователях укладывается в отведенное время ответа сервера 850 мс (причем с довольно большим запасом).

Первая конфигурация превышает требуемое время ответа даже при небольшом кол-ве пользователей. Вторая конфигурация укладывается в допустимые рамки, но лишь до +- 6 активных пользователей.

Описание конфигурации JMeter для стресс-тестирования.

Для стресс-тестирования была выбрана третья конфигурация. Увеличим максимальное количество пользователей и установим ramp-up period равным 150 секунд. Тогда мы сможем отследить изменение времени отклика сервера при постепенном увеличении количества пользователей от 1 до 80.

Thread Group

Name:

Comments:

Action to be taken after a Sampler error

☒ Continue ☐ Start Next Thread Loop ☐ Stop Thread ☐ Stop Test ☐ Stop Test Now

Thread Properties

Number of Threads (users):

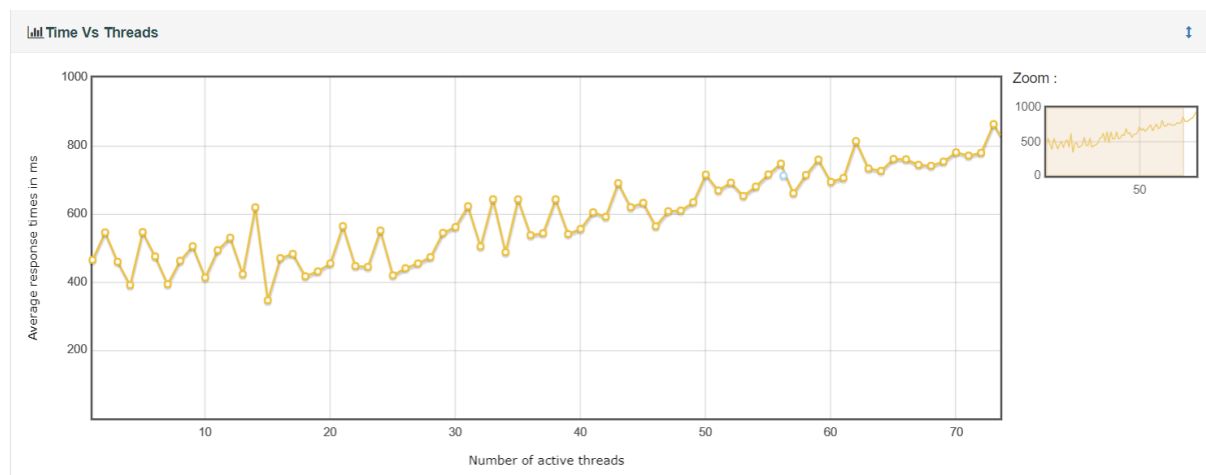
Ramp-up period (seconds):

Loop Count: ☒ Infinite

☒ Same user on each iteration

☐ Delay Thread creation until needed

График изменения времени отклика от нагрузки для выбранной конфигурации, полученный в ходе стресс-тестирования системы.



Допустимое время ответа сервера достигается до увеличения количества пользователей до 72. При дальнейшем увеличении время ответа сервера уже выходит за границы допустимых значений.

Выводы по работе

В ходе выполнения данной лабораторной работы были проведены нагрузочное и стресс тестирования, проведена аналитика полученных данных. Для проведения тестирований использовался [Apache JMeter](#).