

Assignment #4 – Performance Testing using JMeter.

Hangsikhak Sin

7909104

Bachelor of Computer Science (Honours)

CSCN72010 – Software Quality 1 – Testing Tools and Methodologies

Professor Prashanth Cheluvasai Ranga

December 08, 2023

LinkedIn learning Course completion certificate

LinkedIn Learning

JMeter: Performance and Load Testing

Course completed by Hangsihak Sin
Nov 30, 2023 at 11:55PM UTC • 32 minutes

Top skills covered

Load Testing **JMeter**



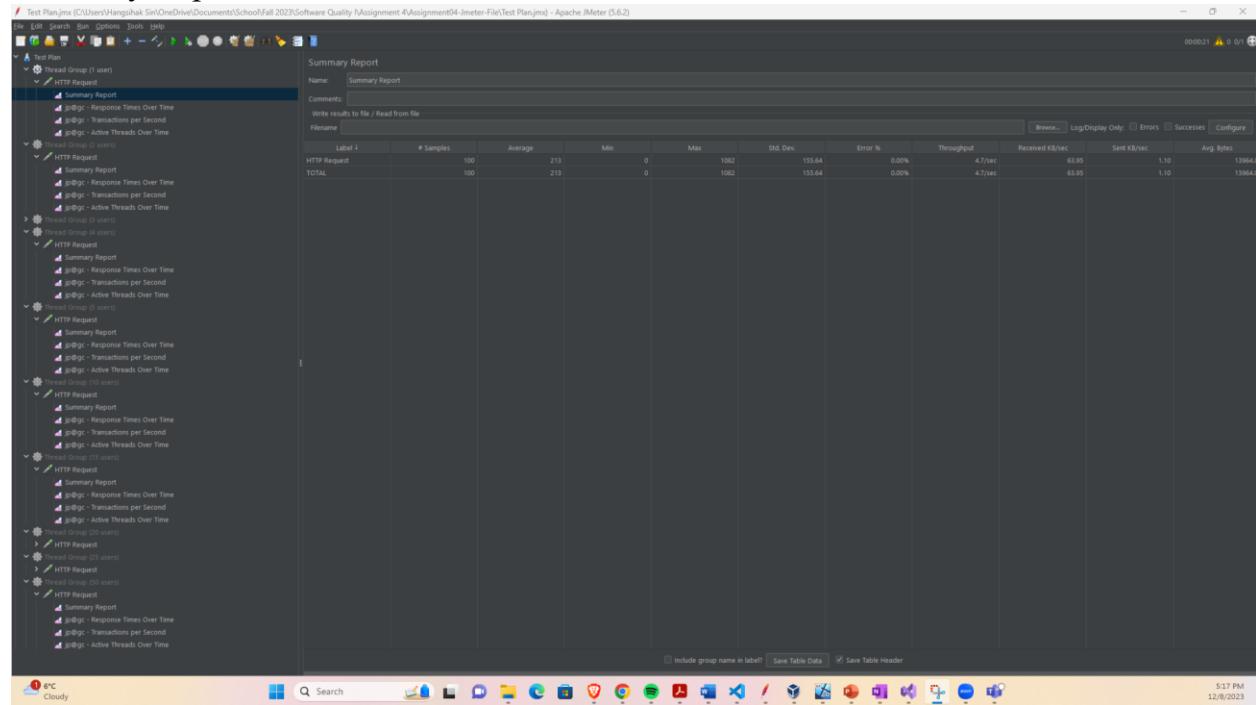
Head of Content Strategy, Learning



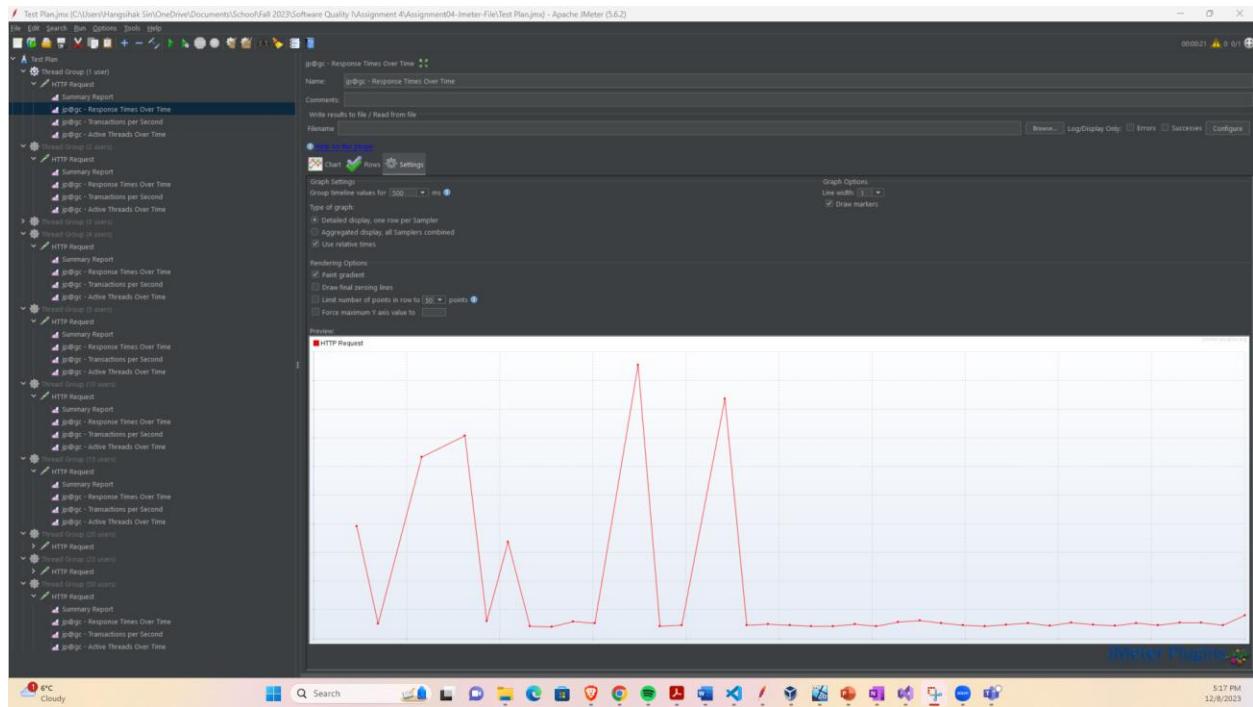
Certificate ID: 0ce961a2cce0c4a7750184a1c03e9cf071b5aad8a381dcc2a62f949329cd0611

Thread number: 1 user

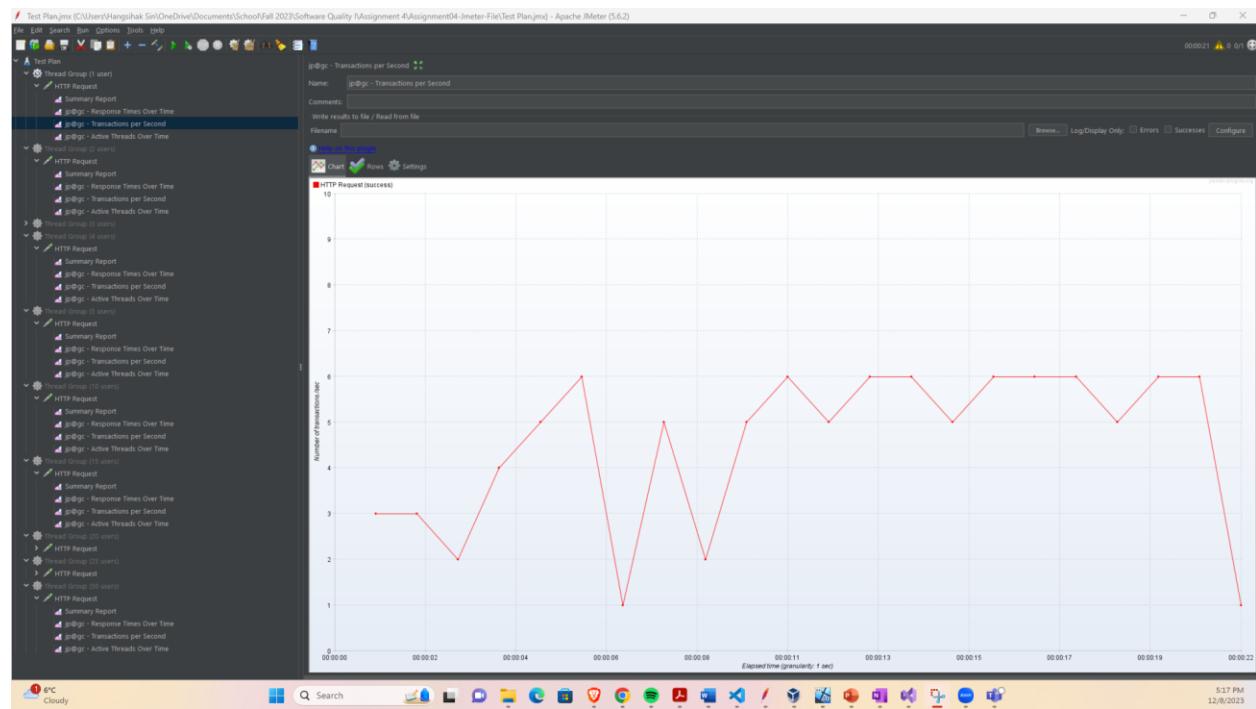
Summary Report



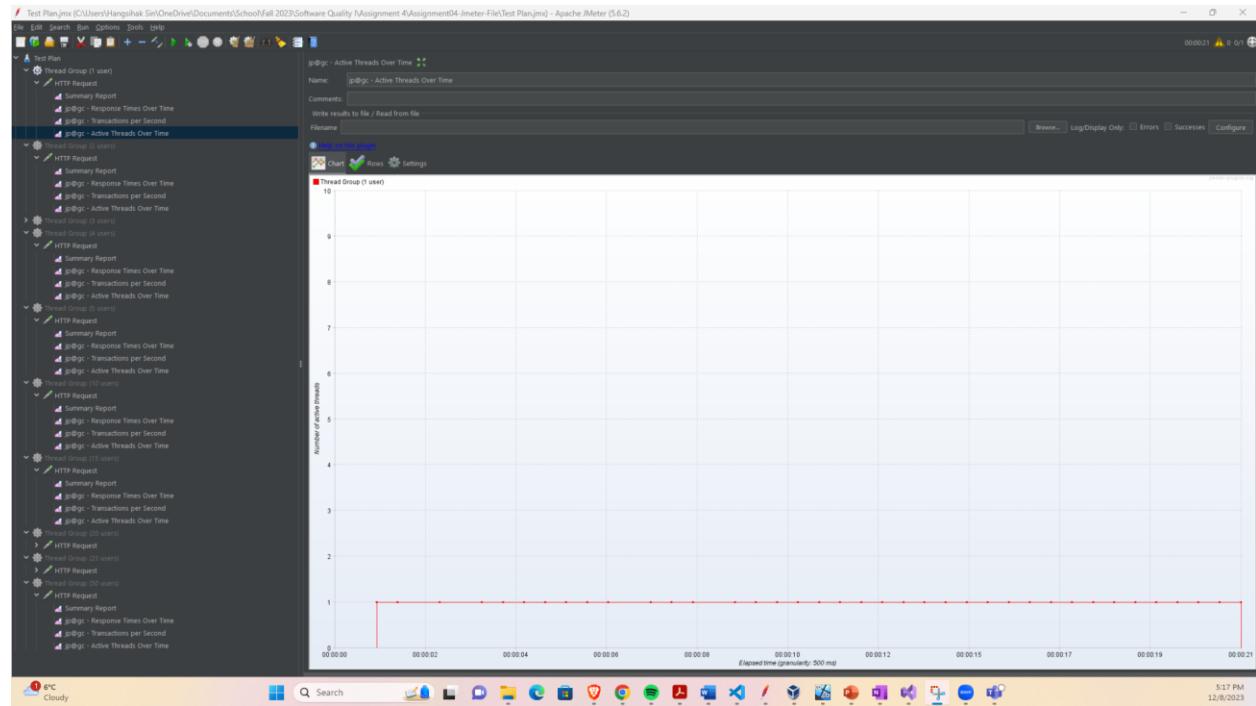
Response Times over Time



Transactions per Second

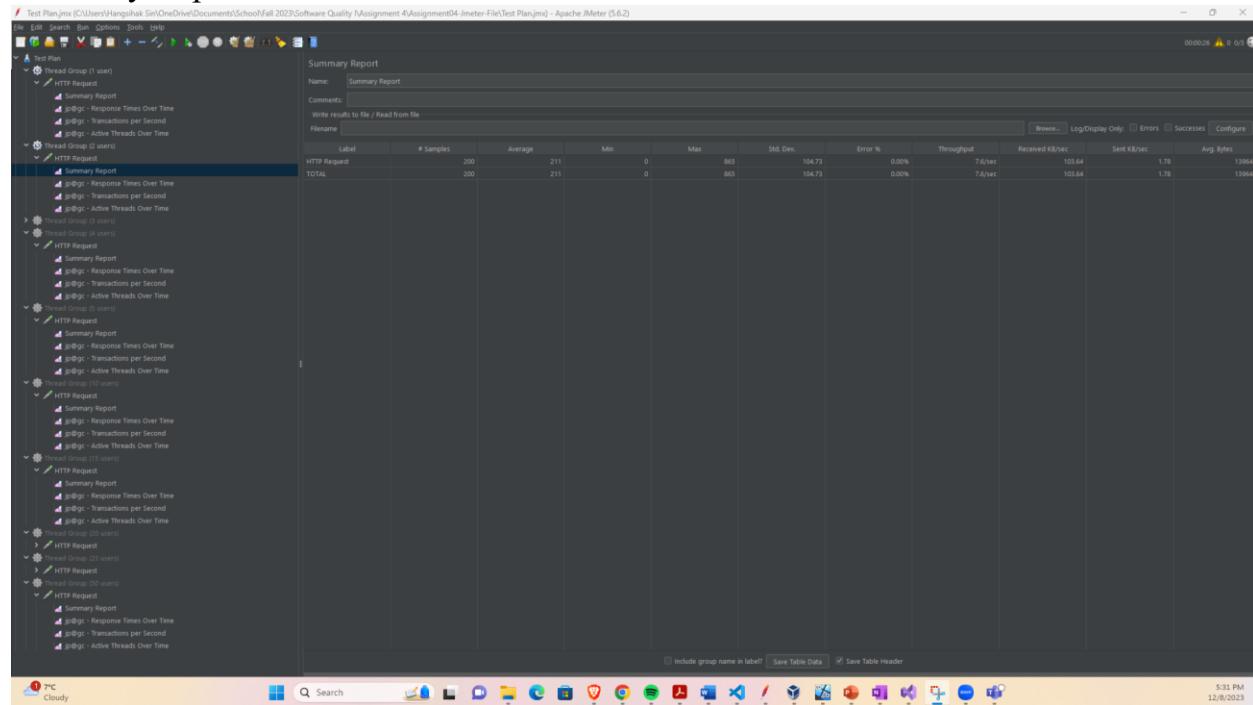


Active Threads over Time

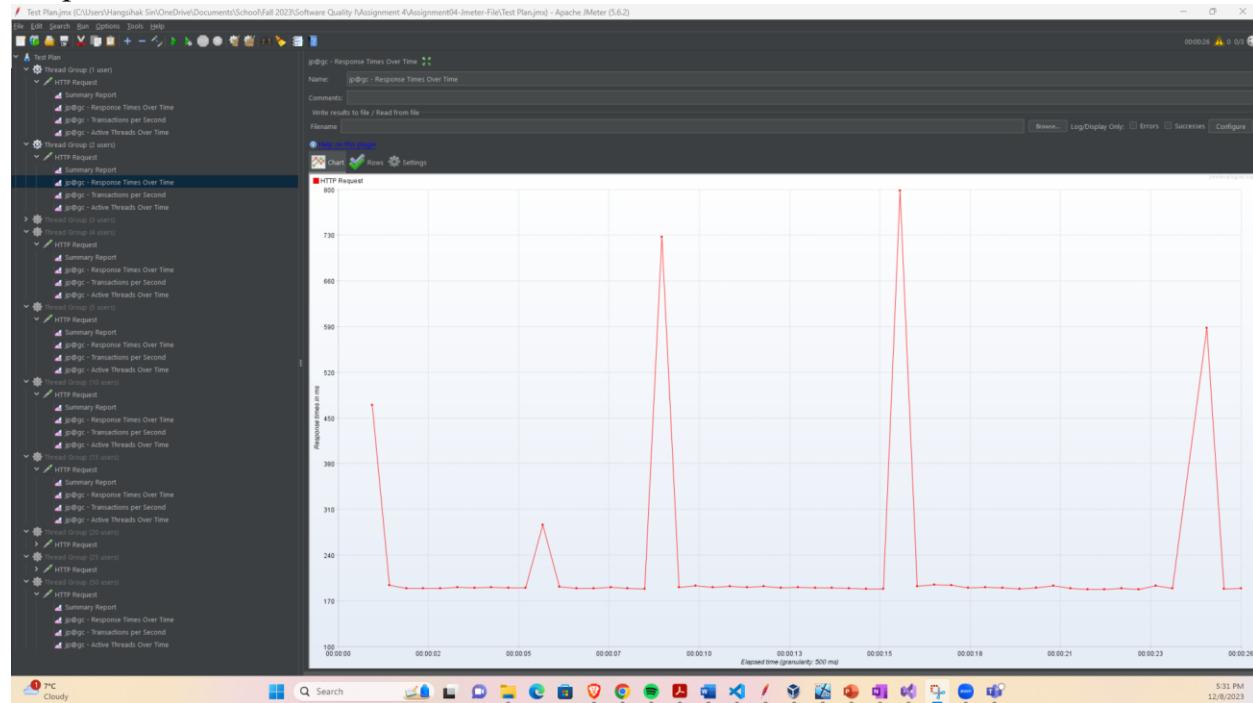


Thread number: 2 users

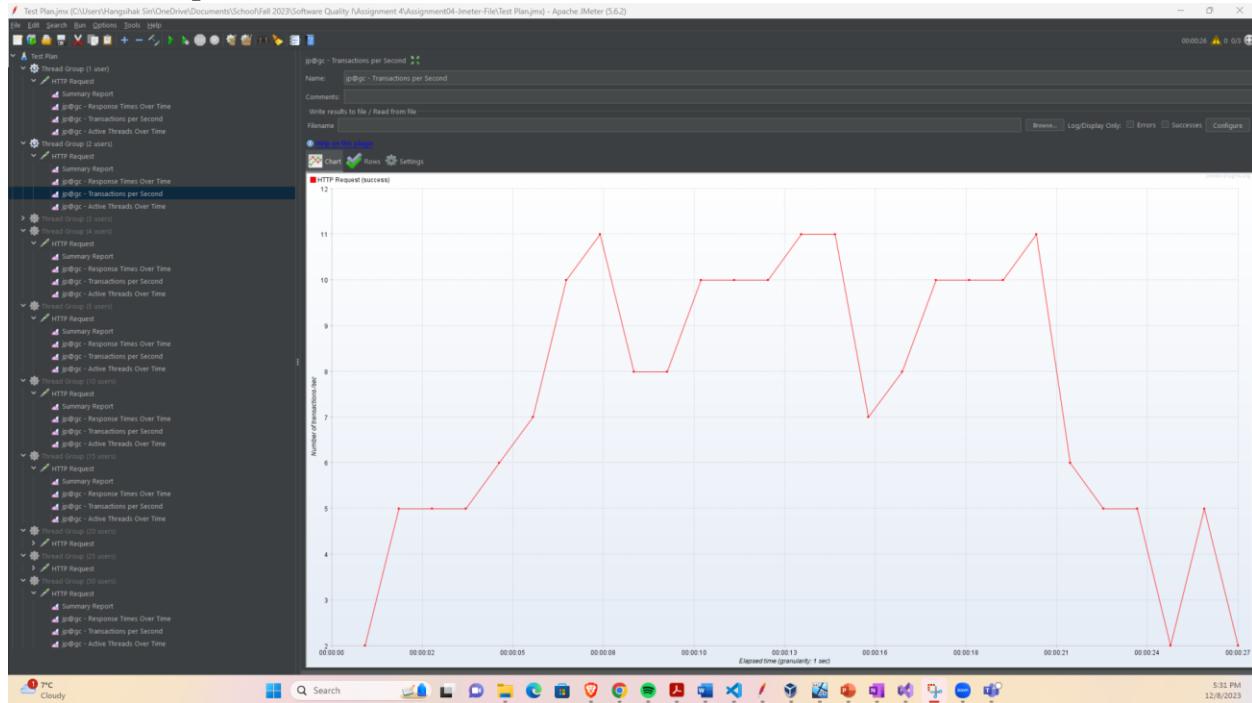
Summary Report



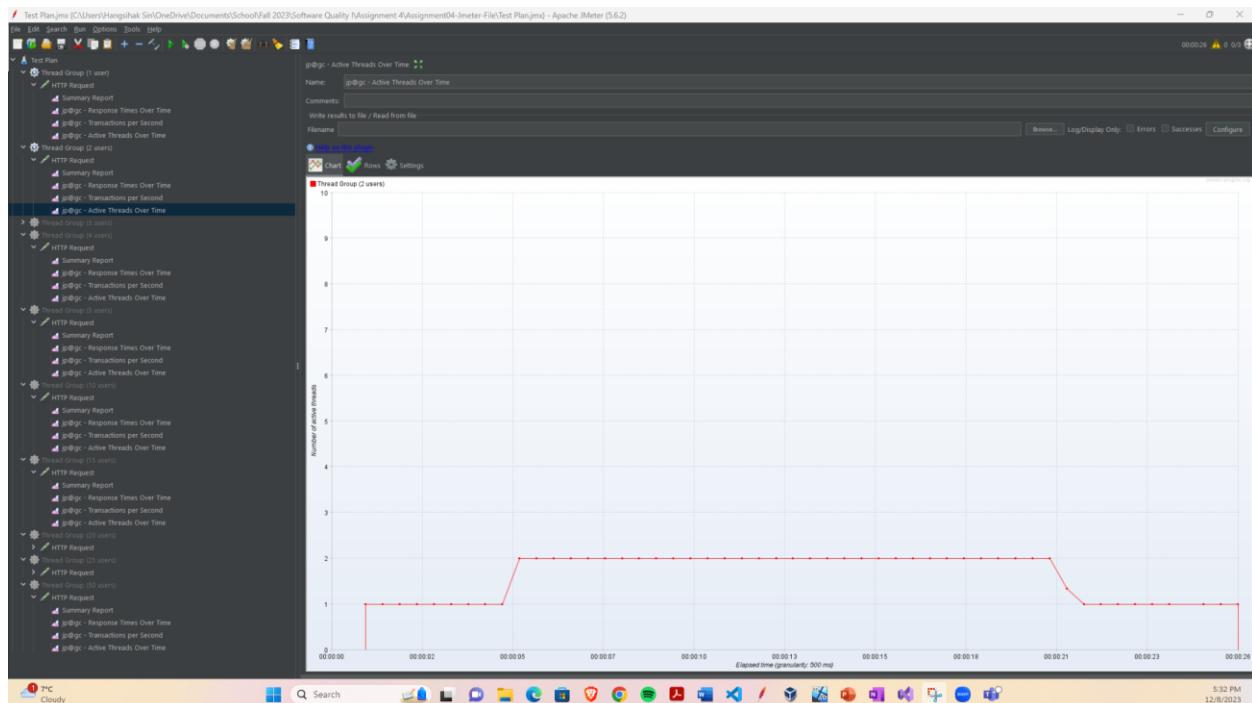
Response Times Over Time



Transactions per Second

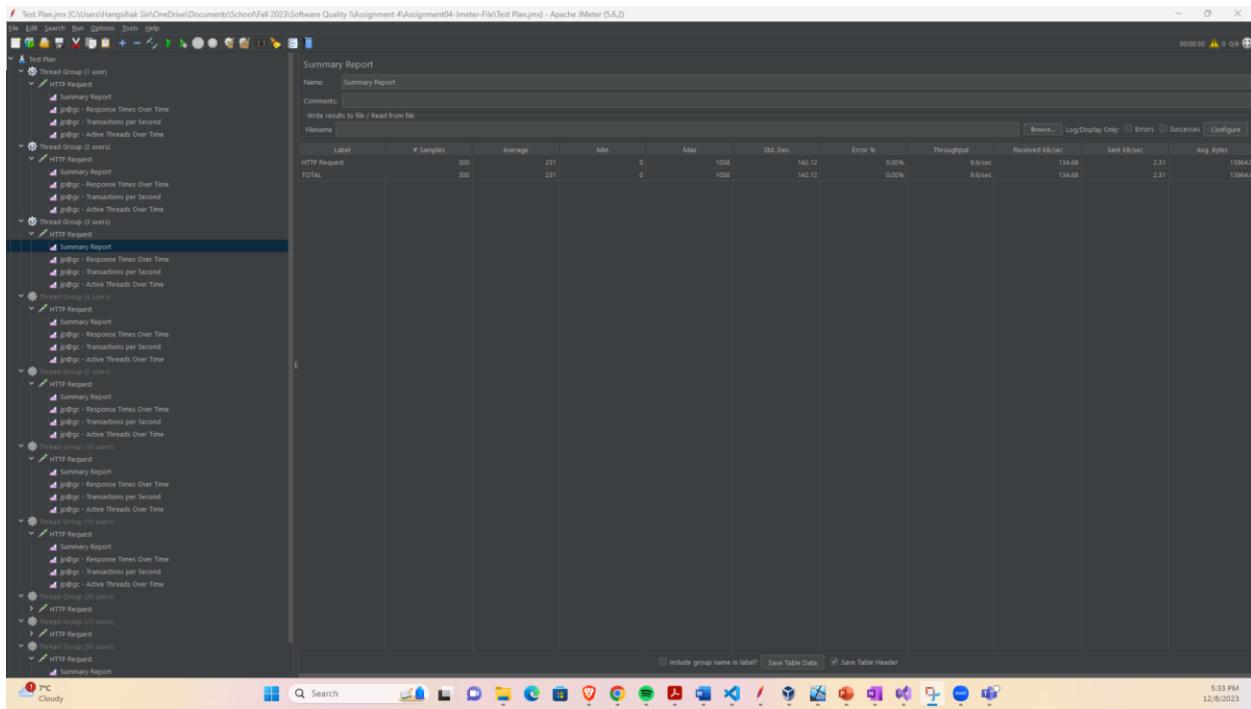


Active Threads Over Time

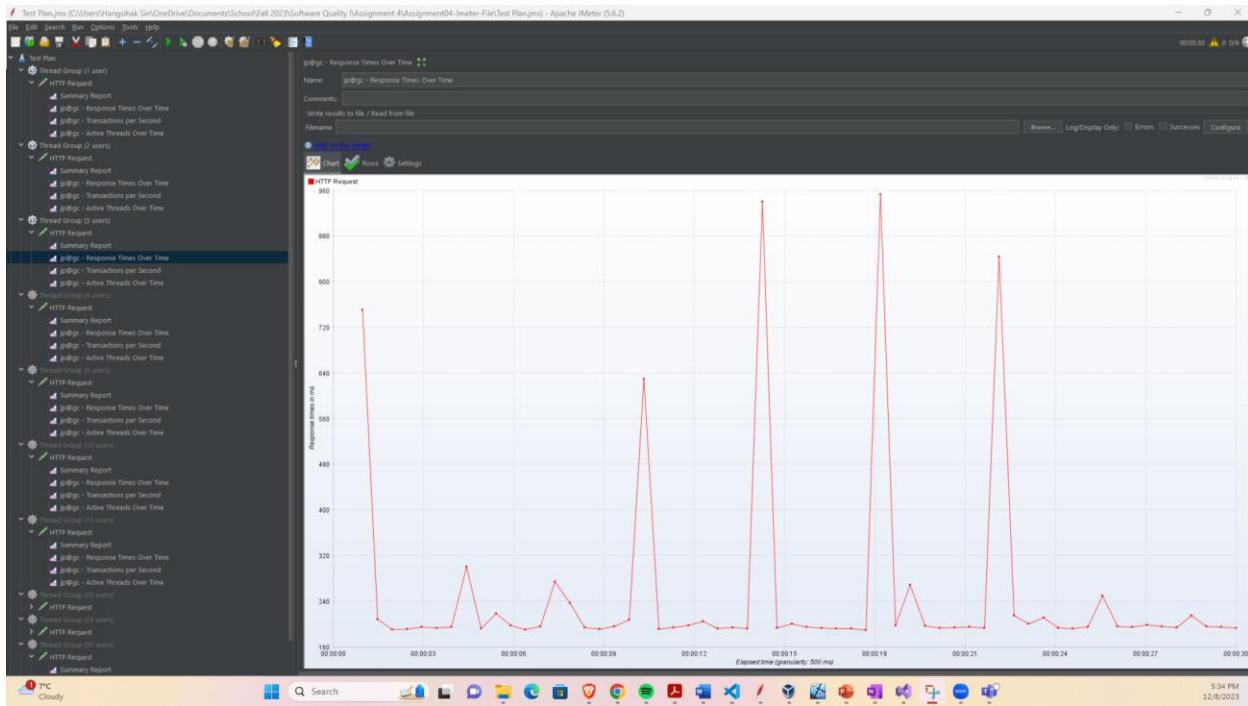


Thread Group (3 users)

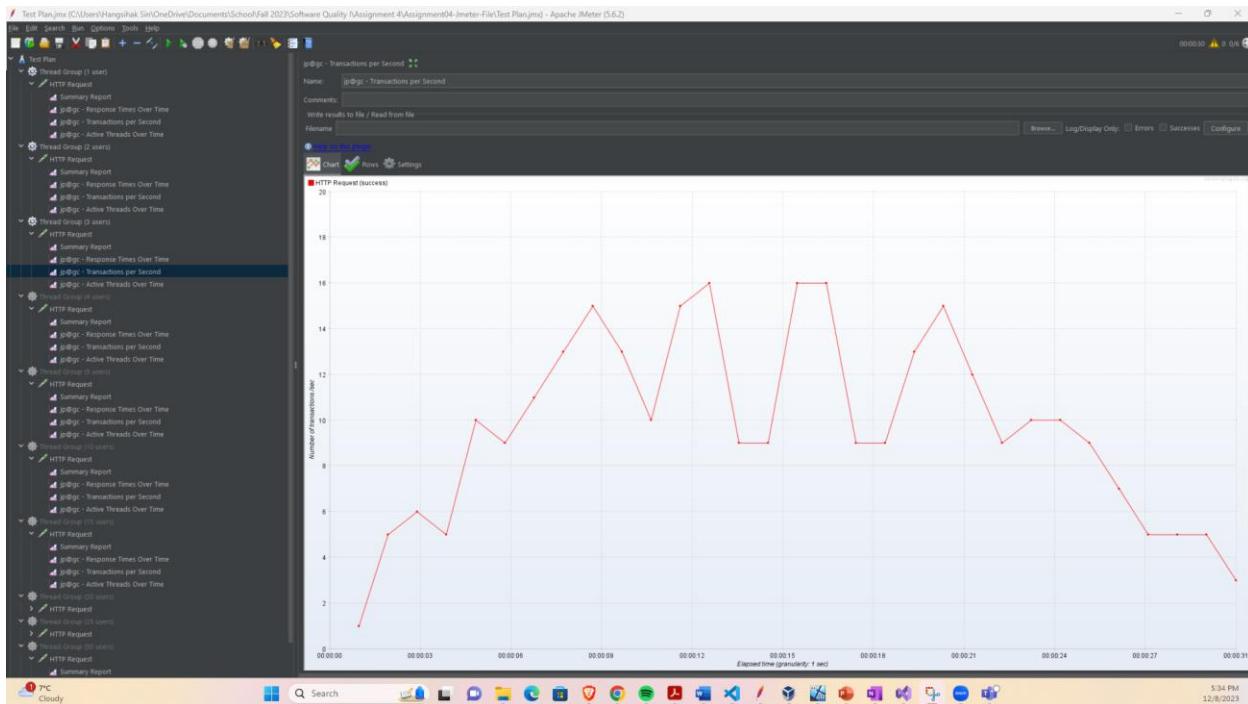
Summary Report



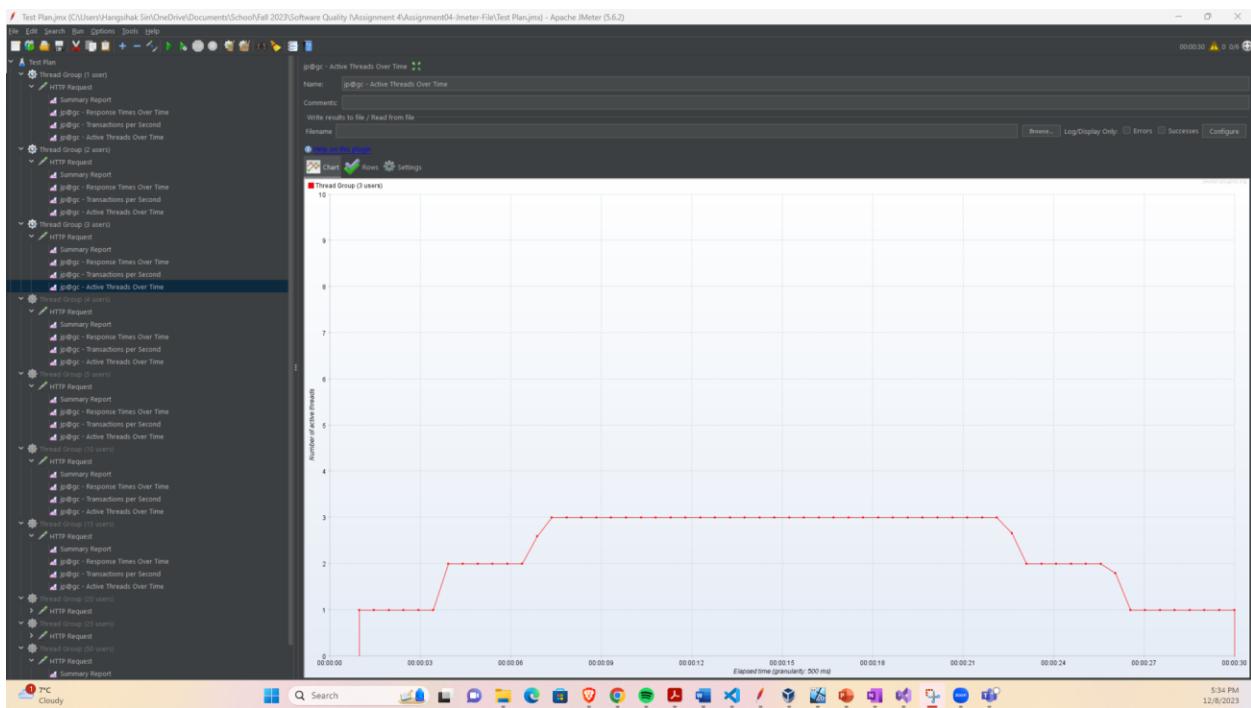
Response Times Over Time



Transactions per Second

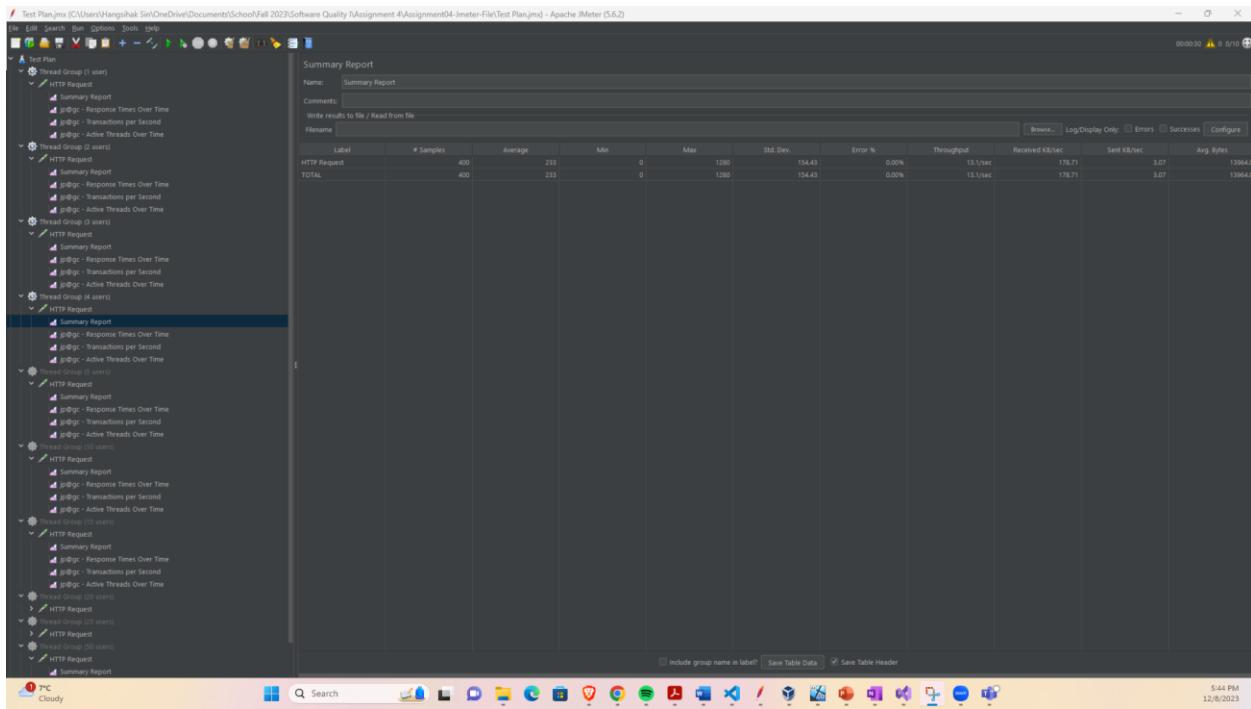


Active Threads Over Time

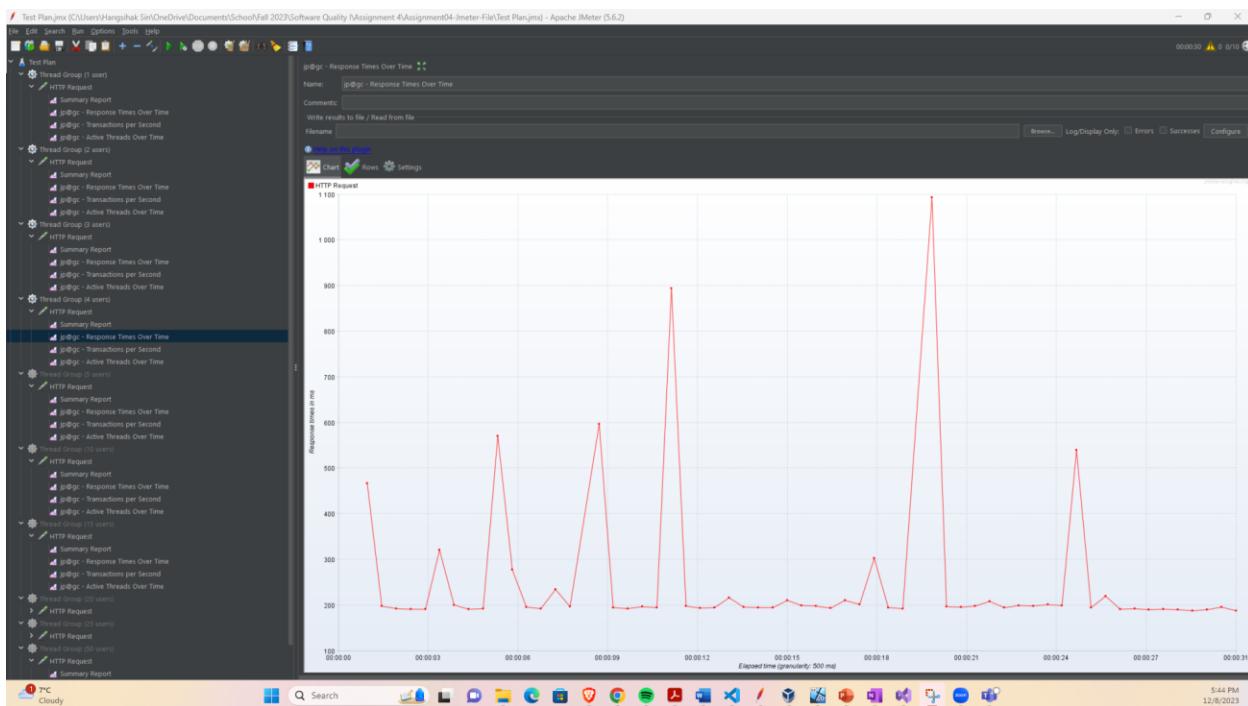


Thread Group (4 users)

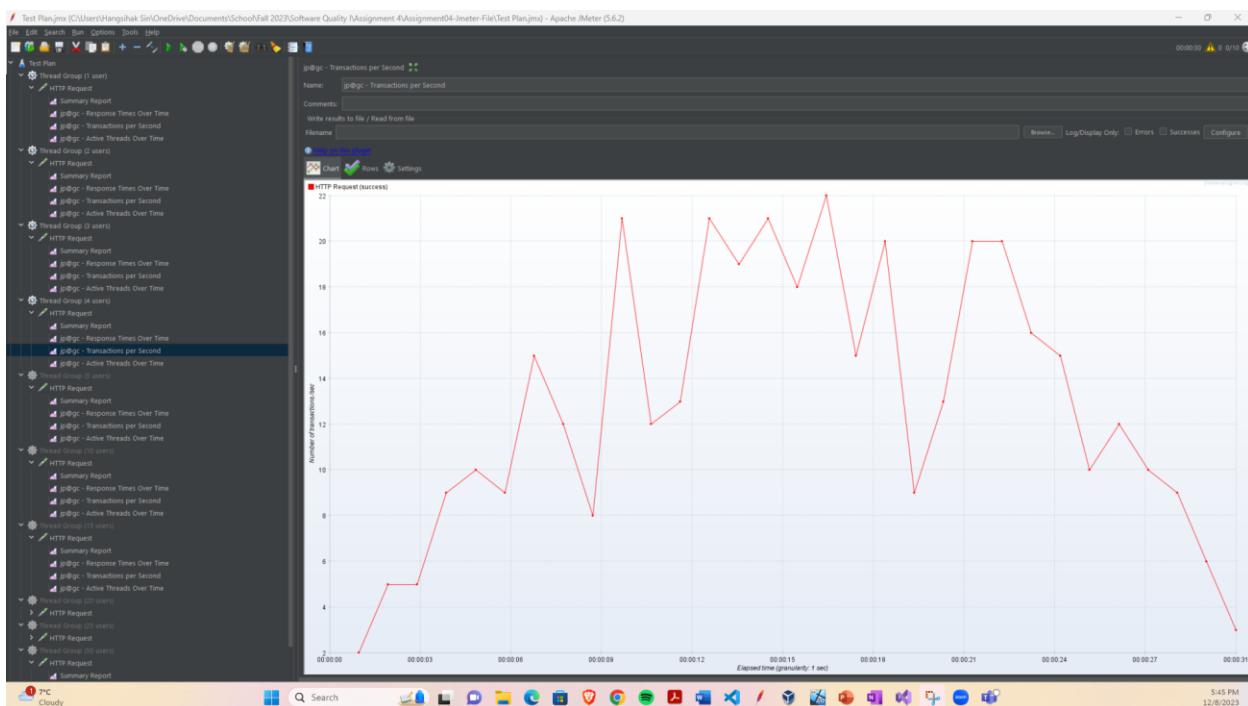
Summary Report



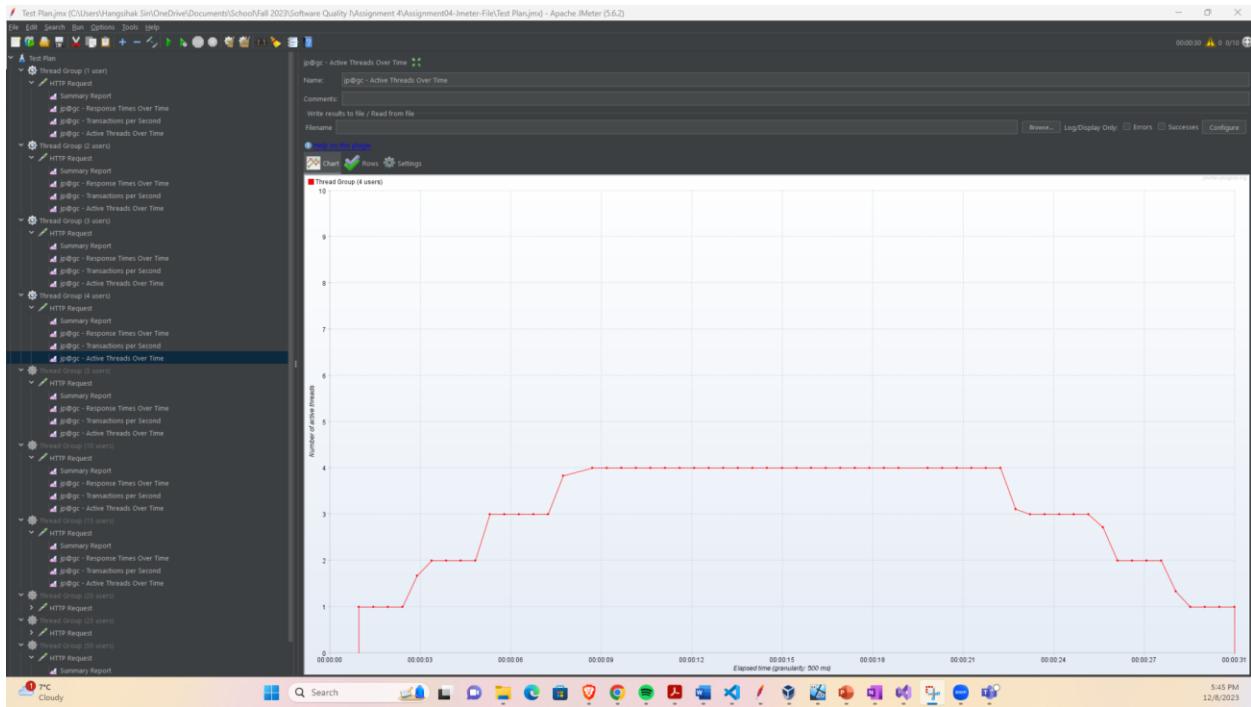
Response Times Over Time



Transactions per Second

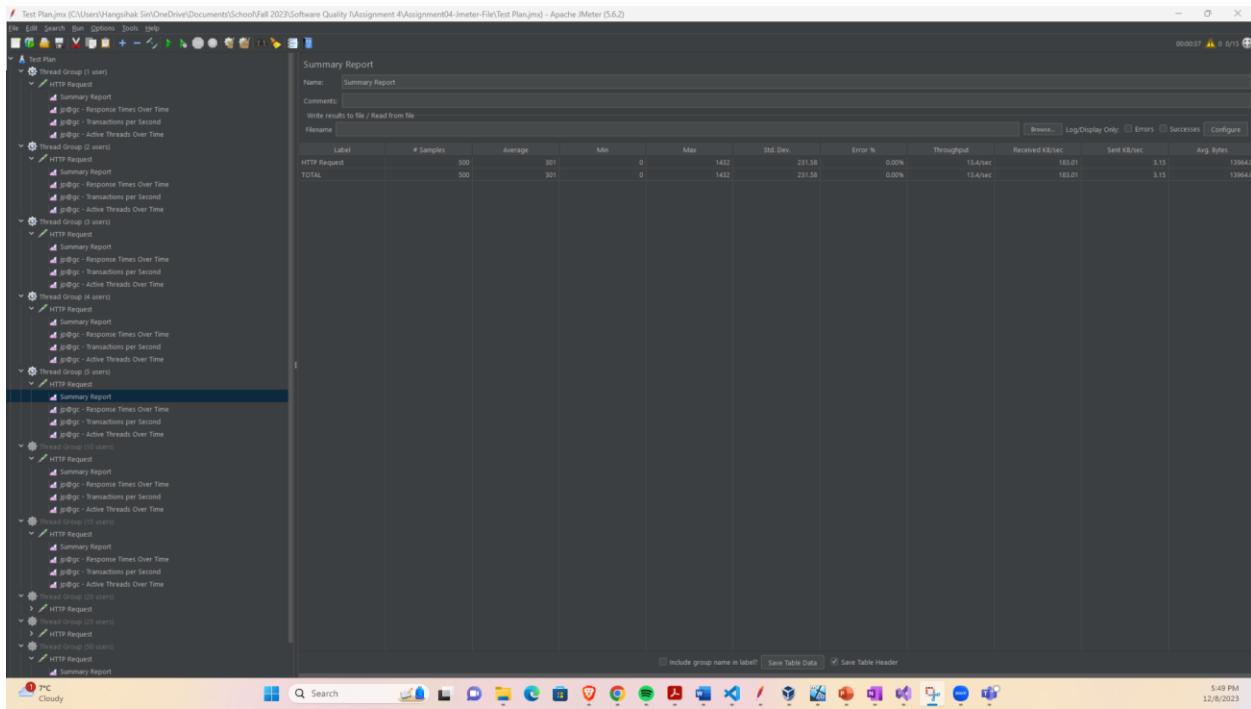


Active Threads Over Time

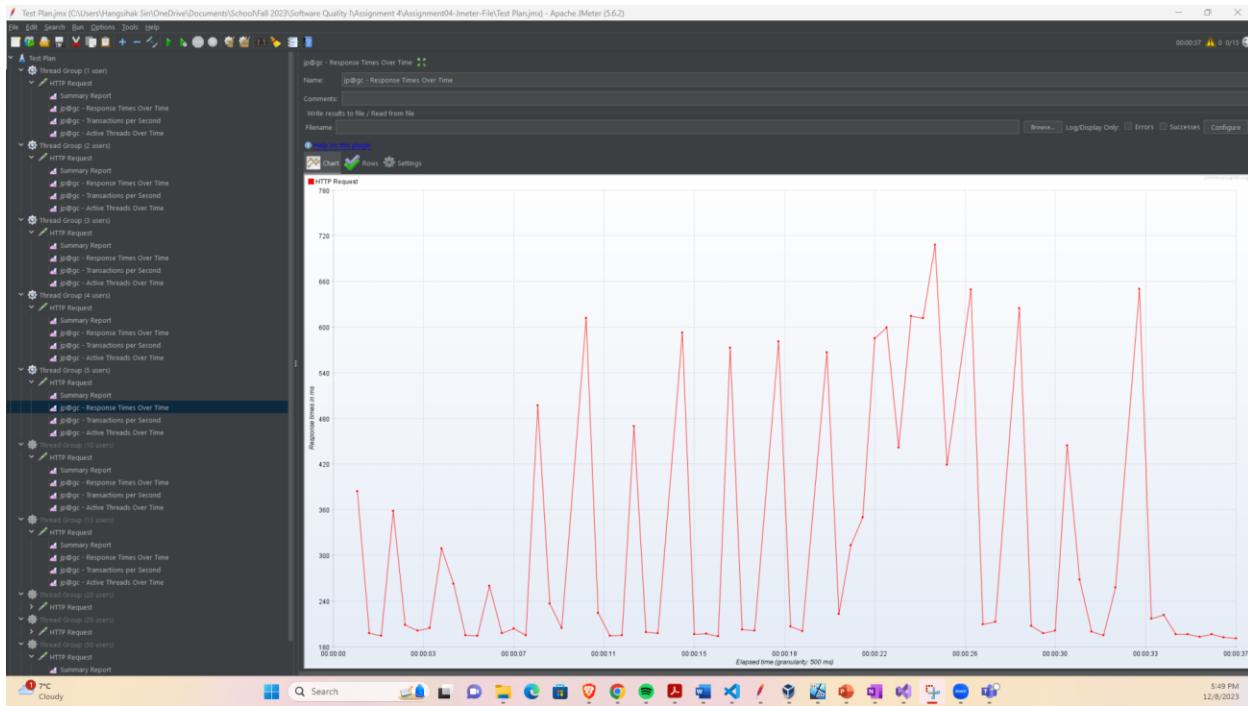


Thread Group (5 users)

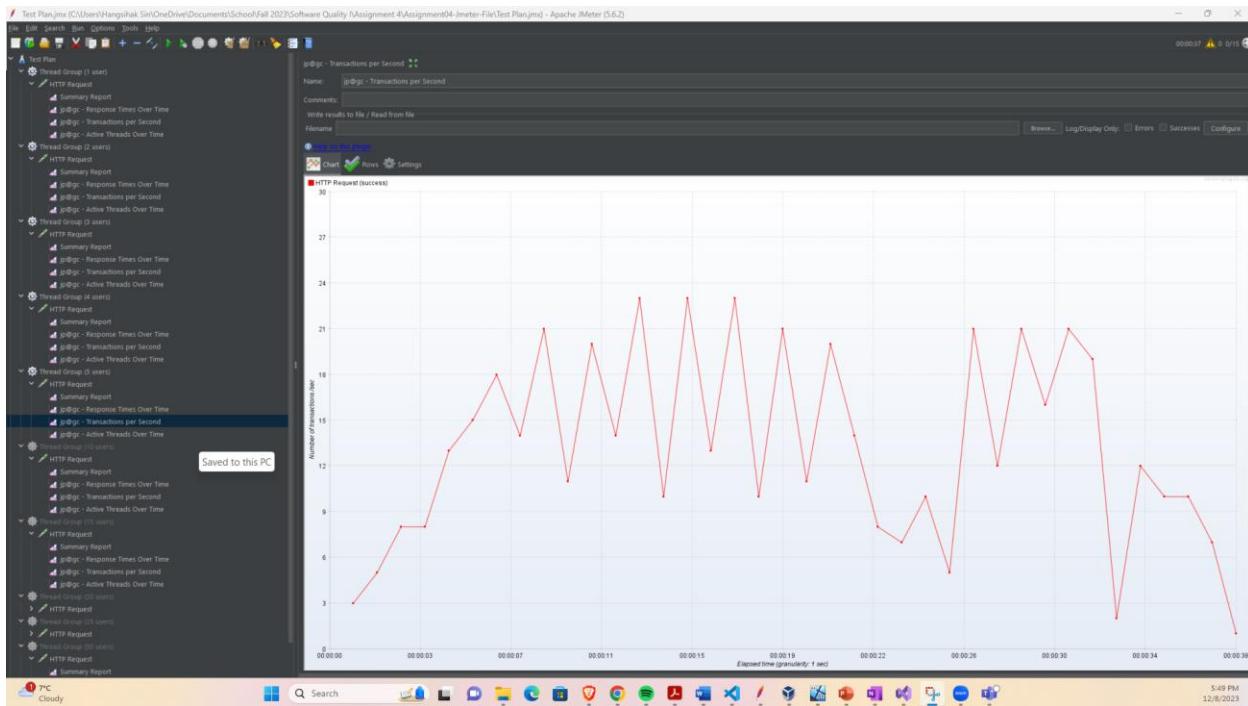
Summary Report



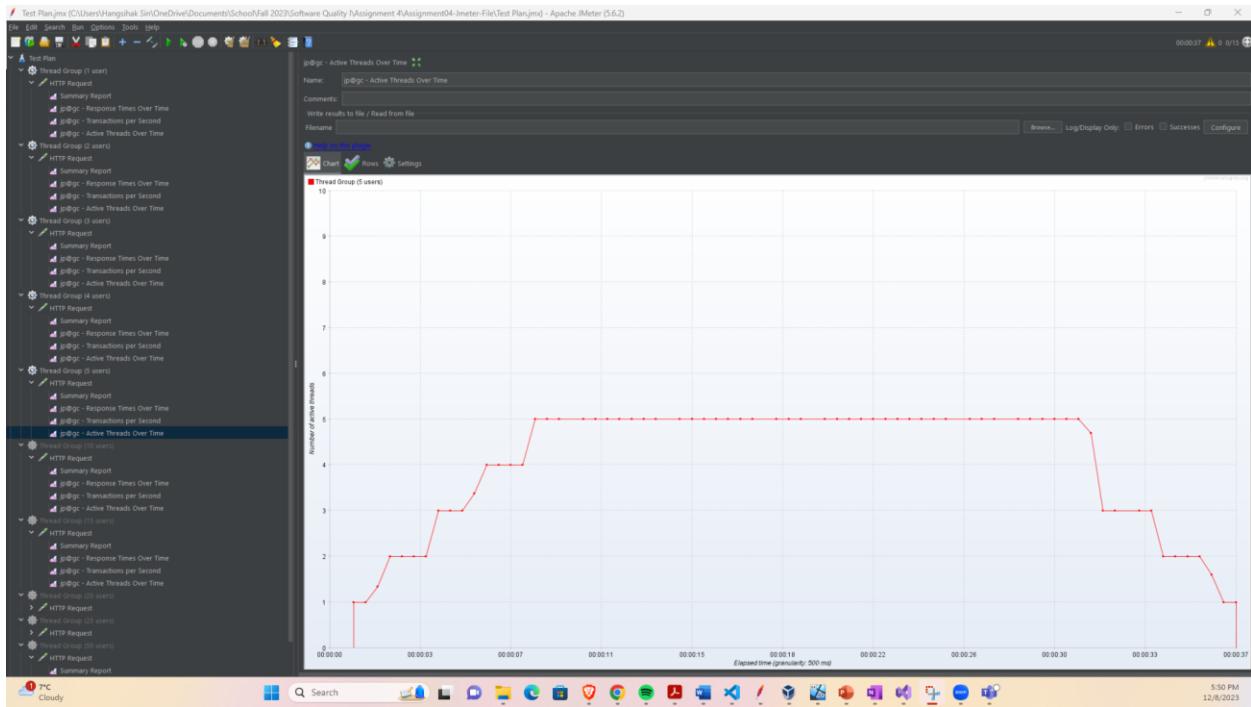
Response Times Over Time



Transactions per Second

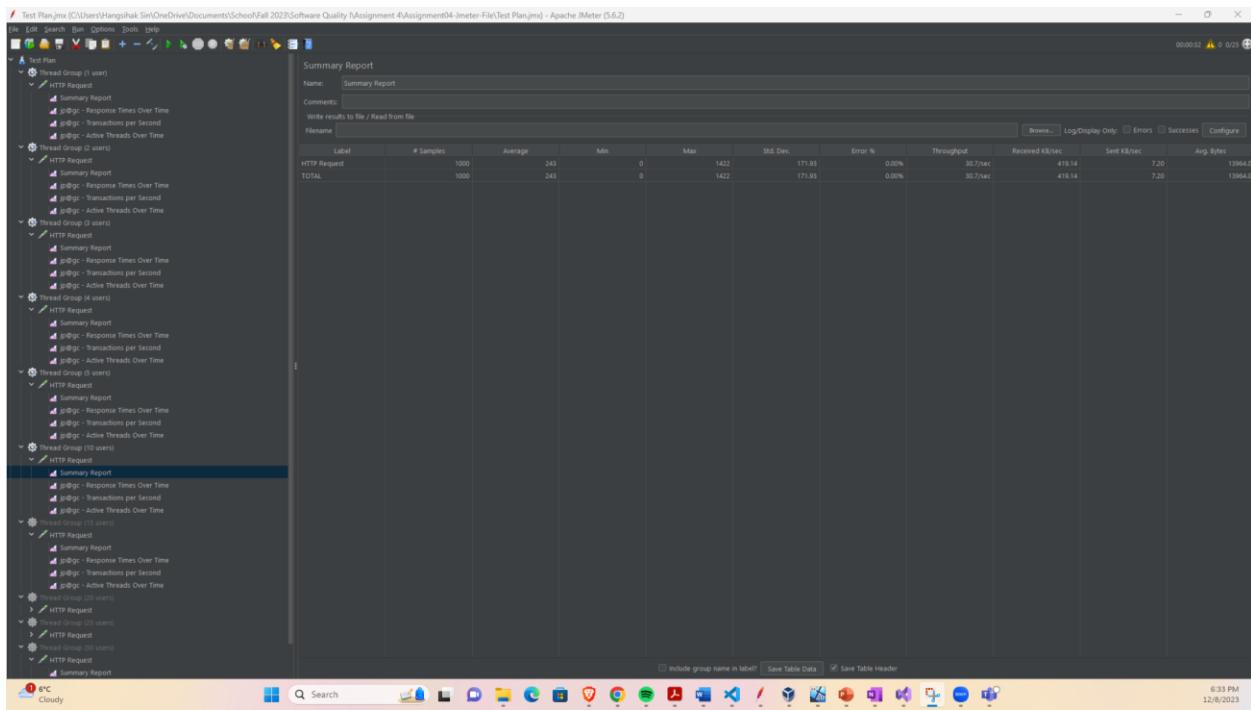


Active Threads Over Time

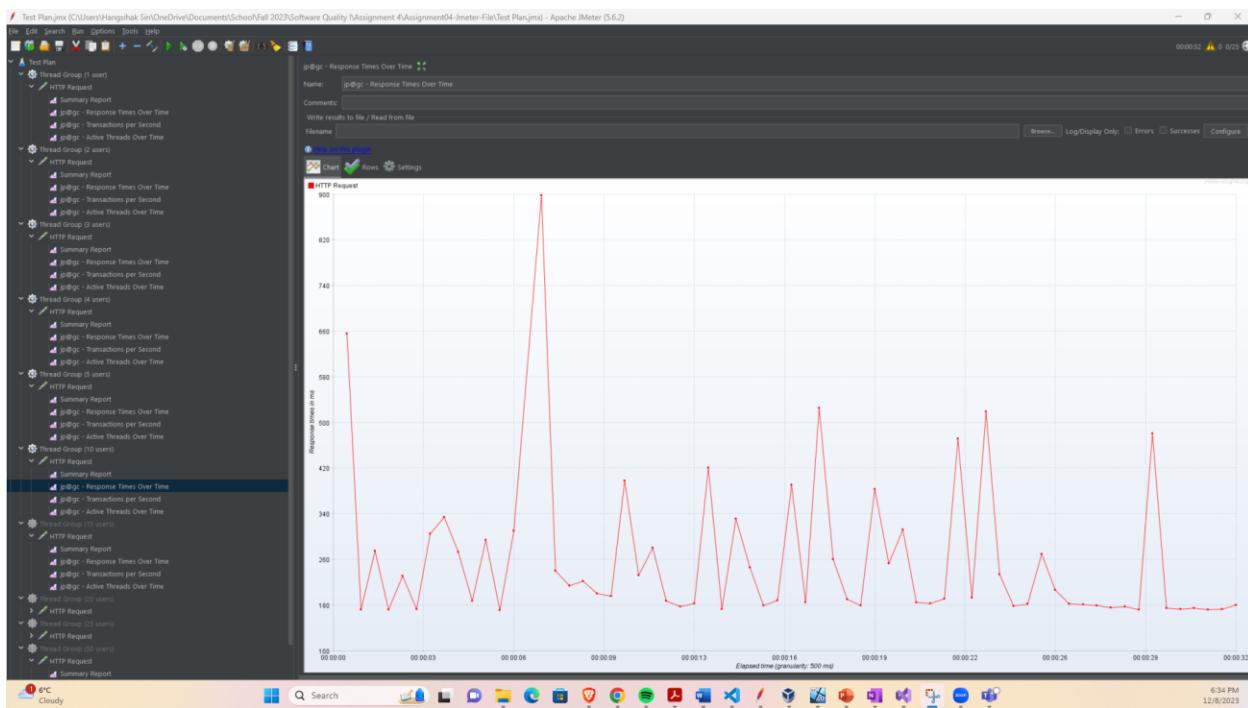


Thread Group (10 users)

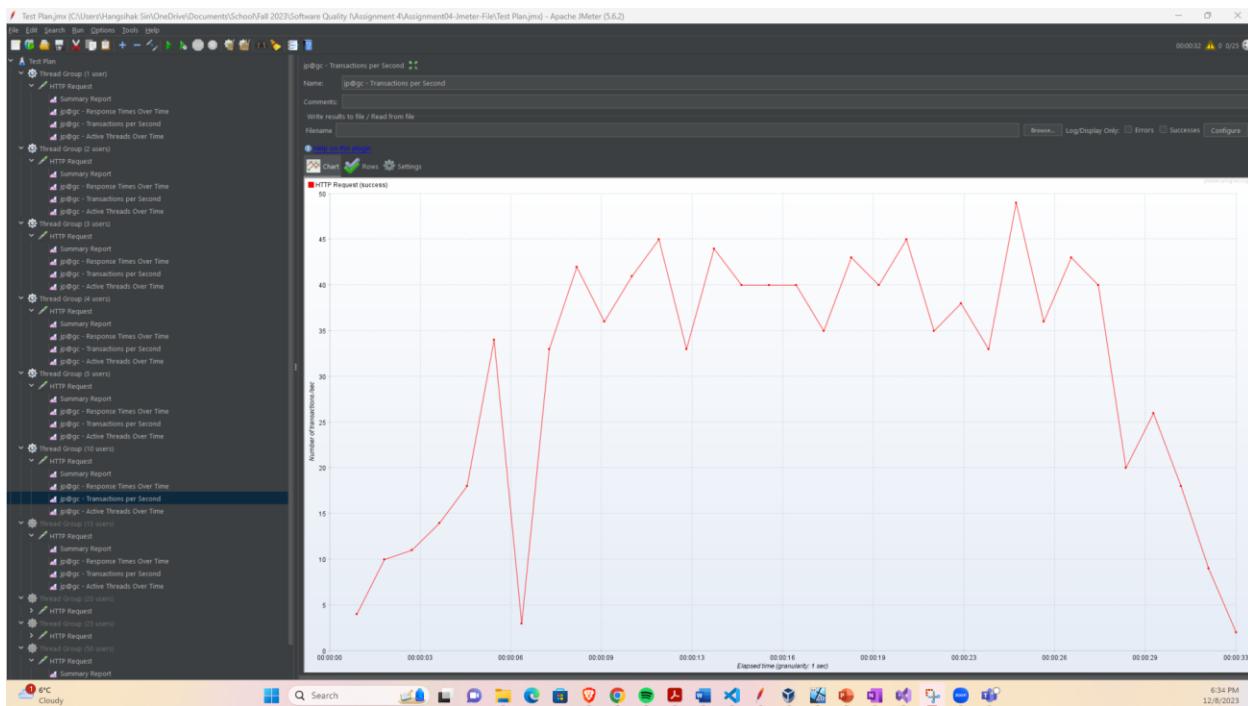
Summary Report



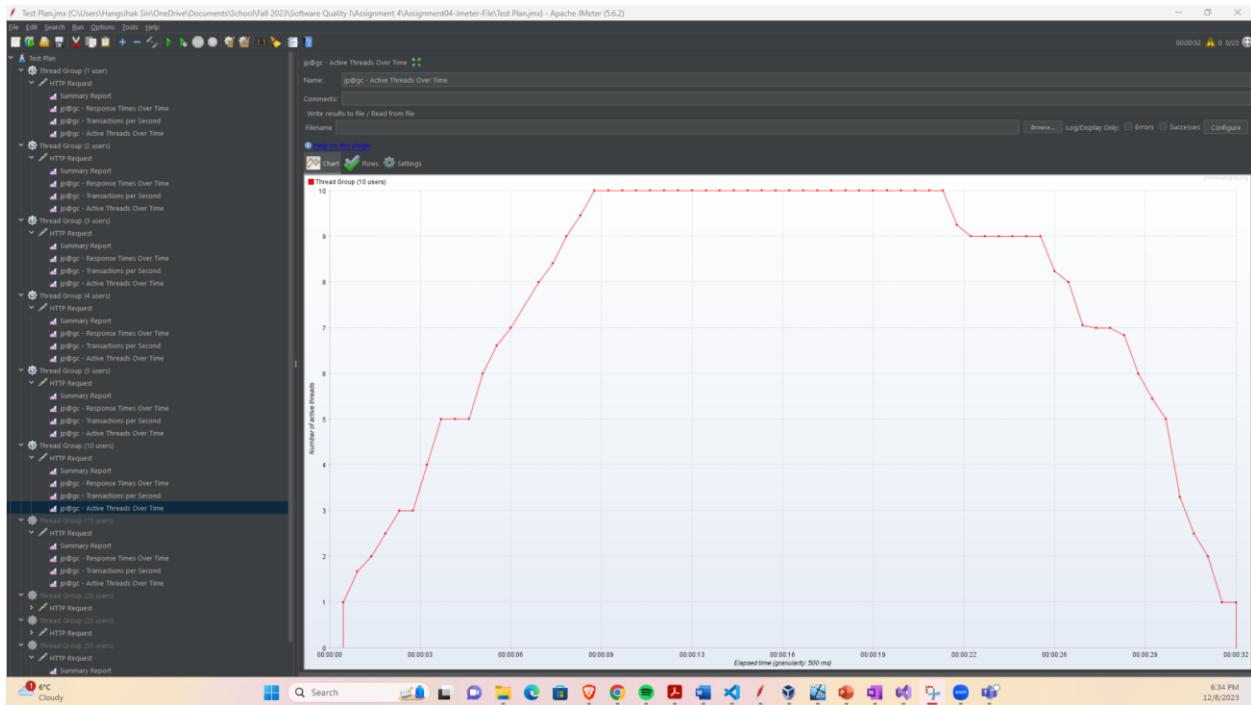
Response Times Over Time



Transactions per Second

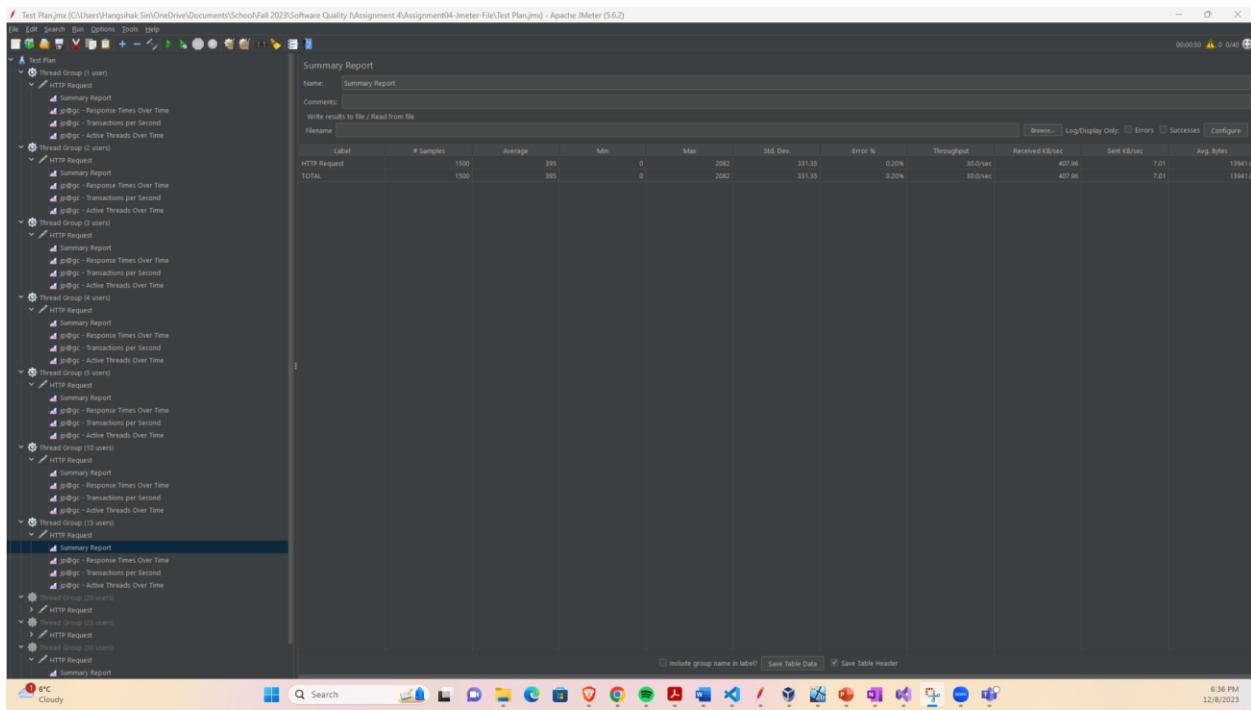


Active Threads Over Time

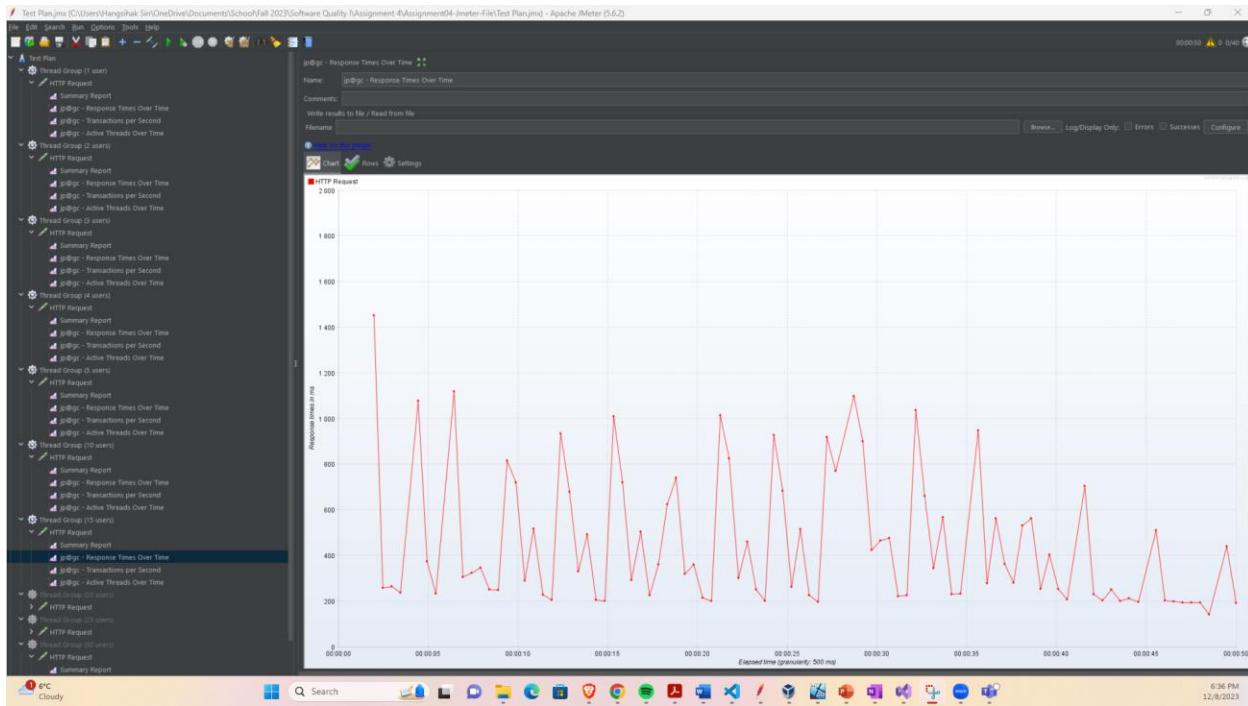


Thread Group (15 users)

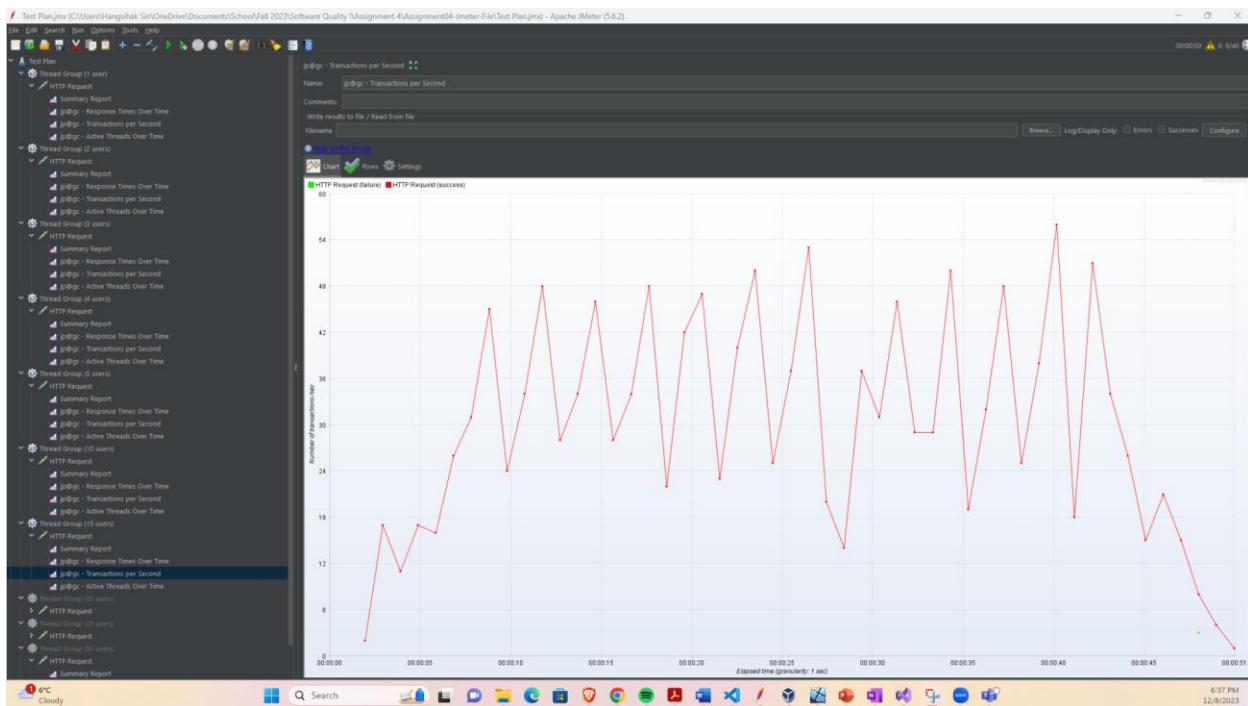
Summary Report



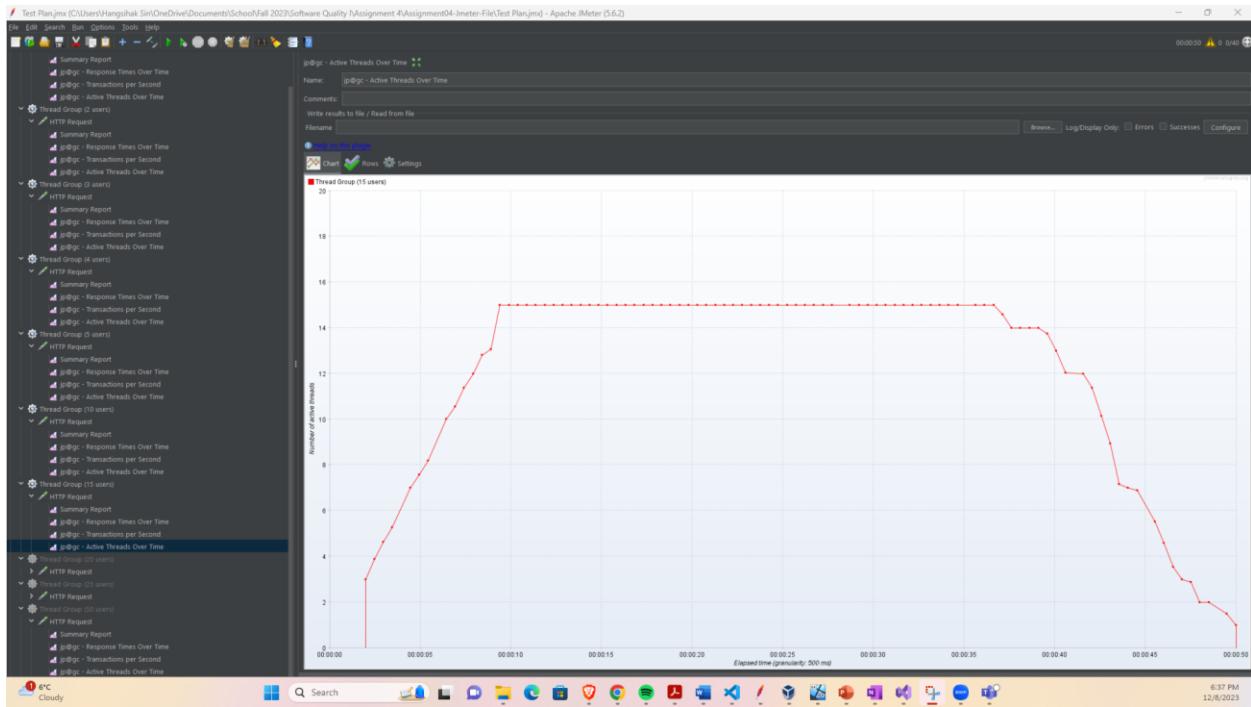
Response Times Over Time



Transactions per Second

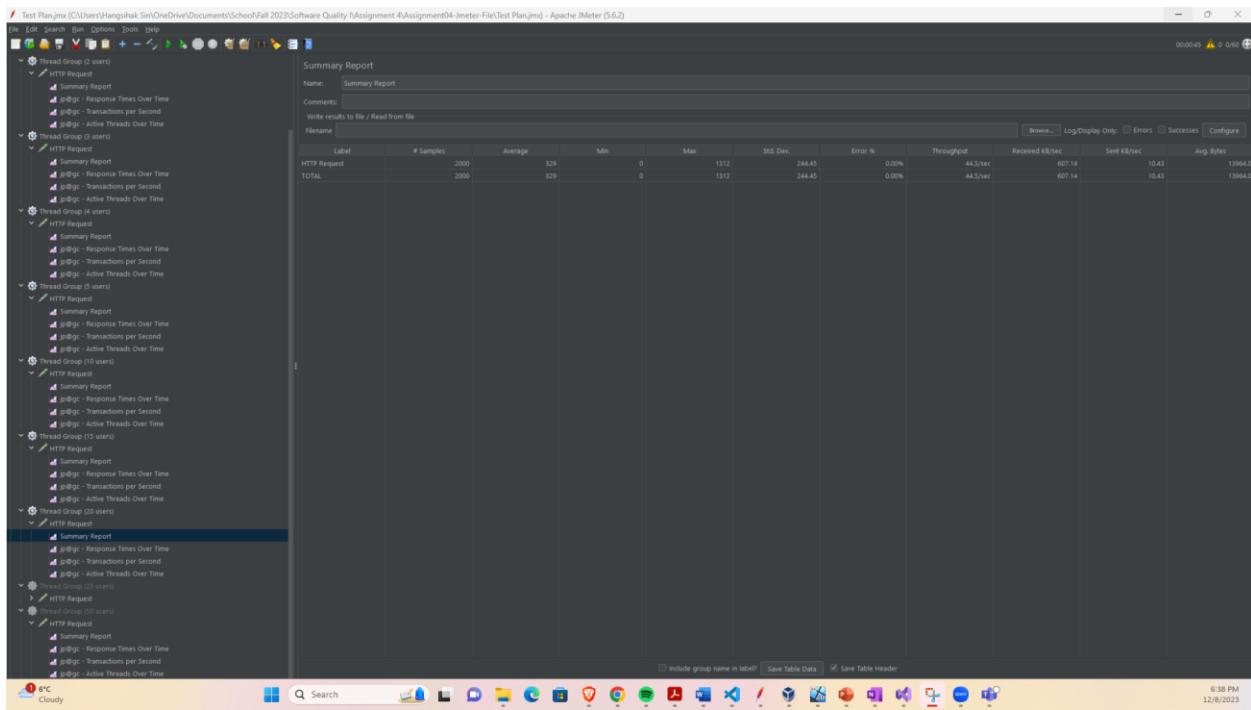


Active Threads Over Time

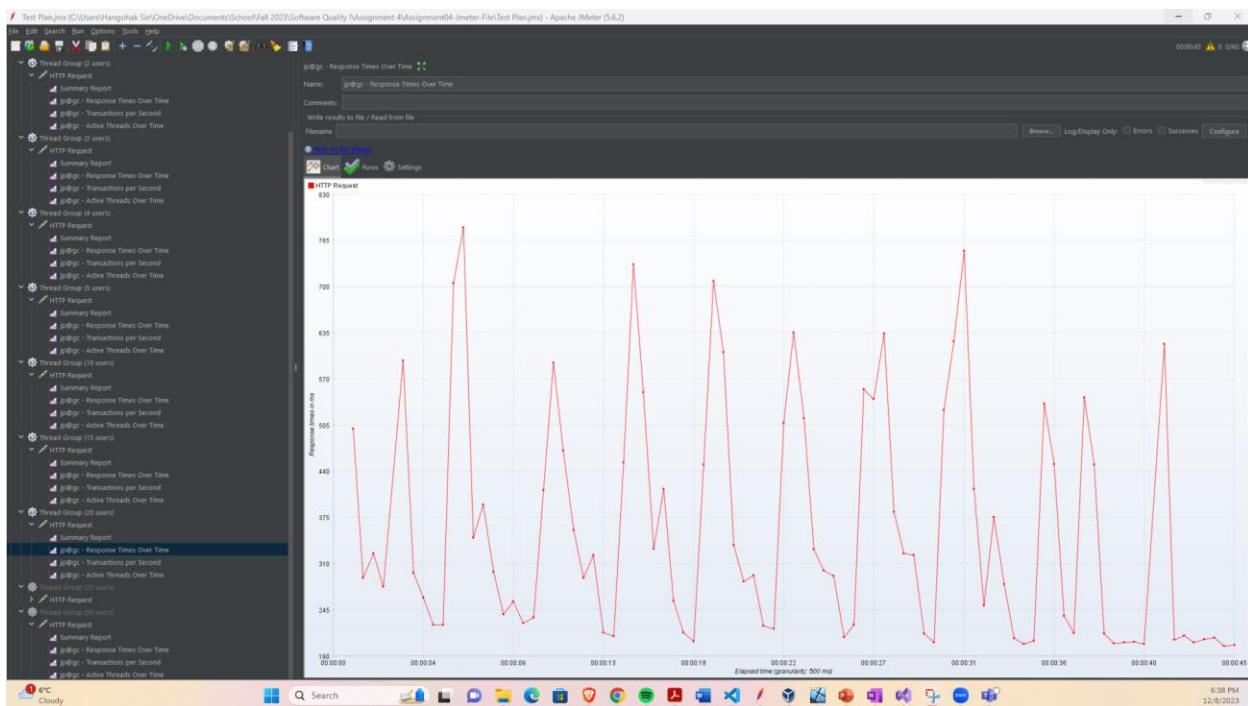


Thread Group (20 users)

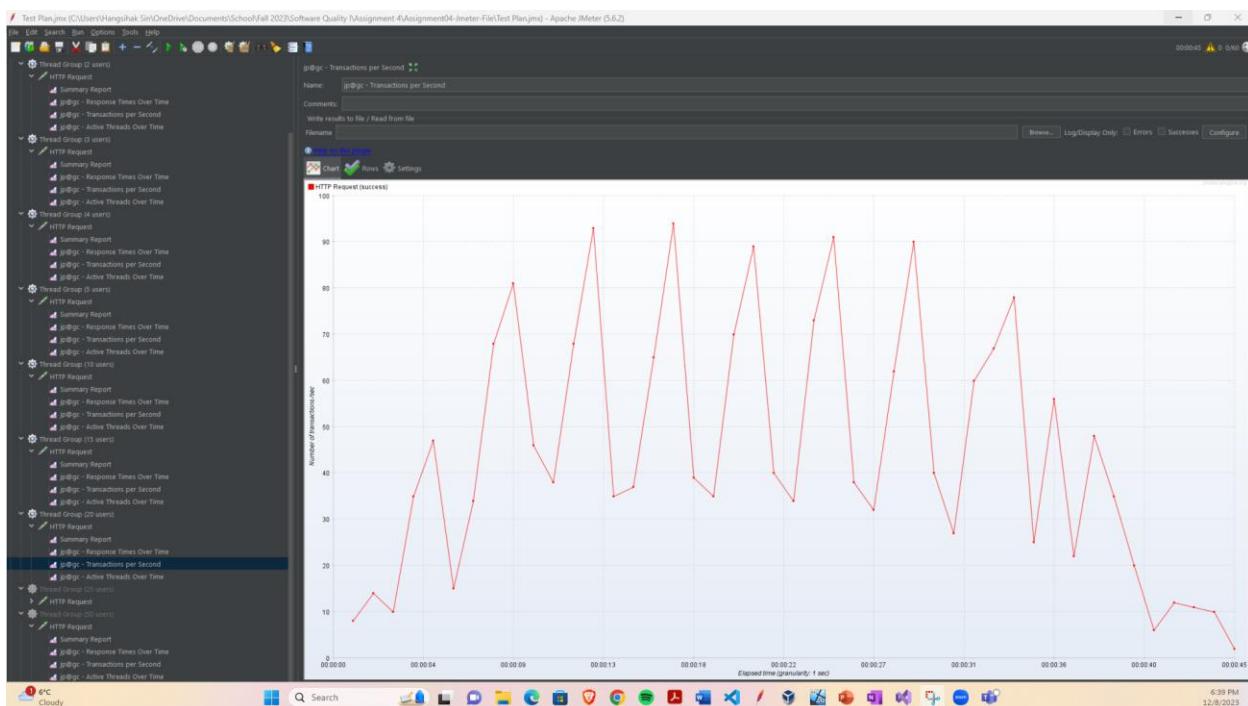
Summary Report



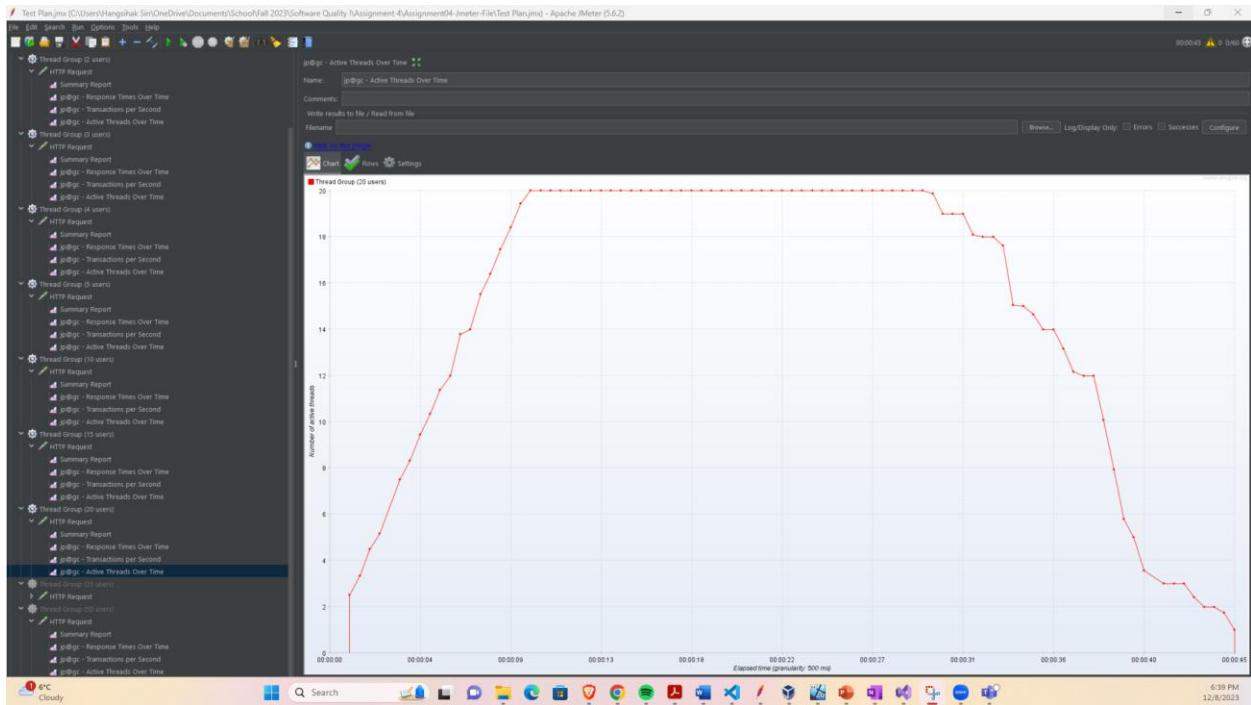
Response Times Over Time



Transactions per Second

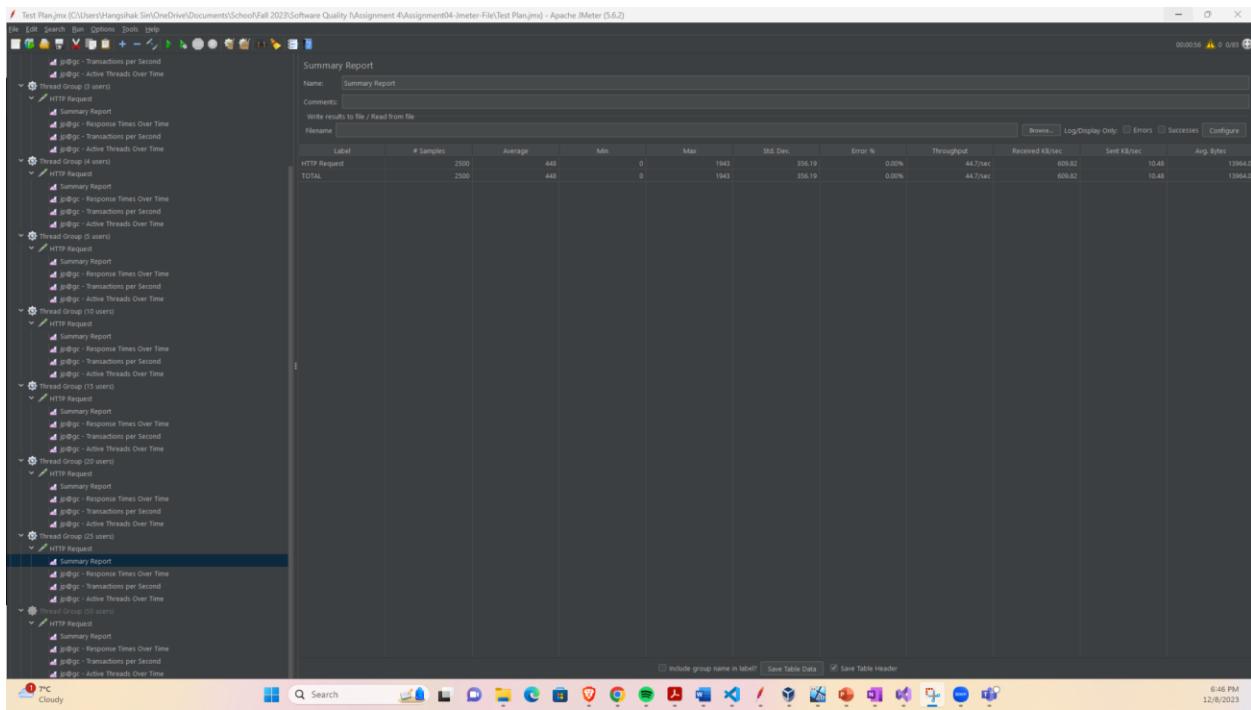


Active Threads Over Time

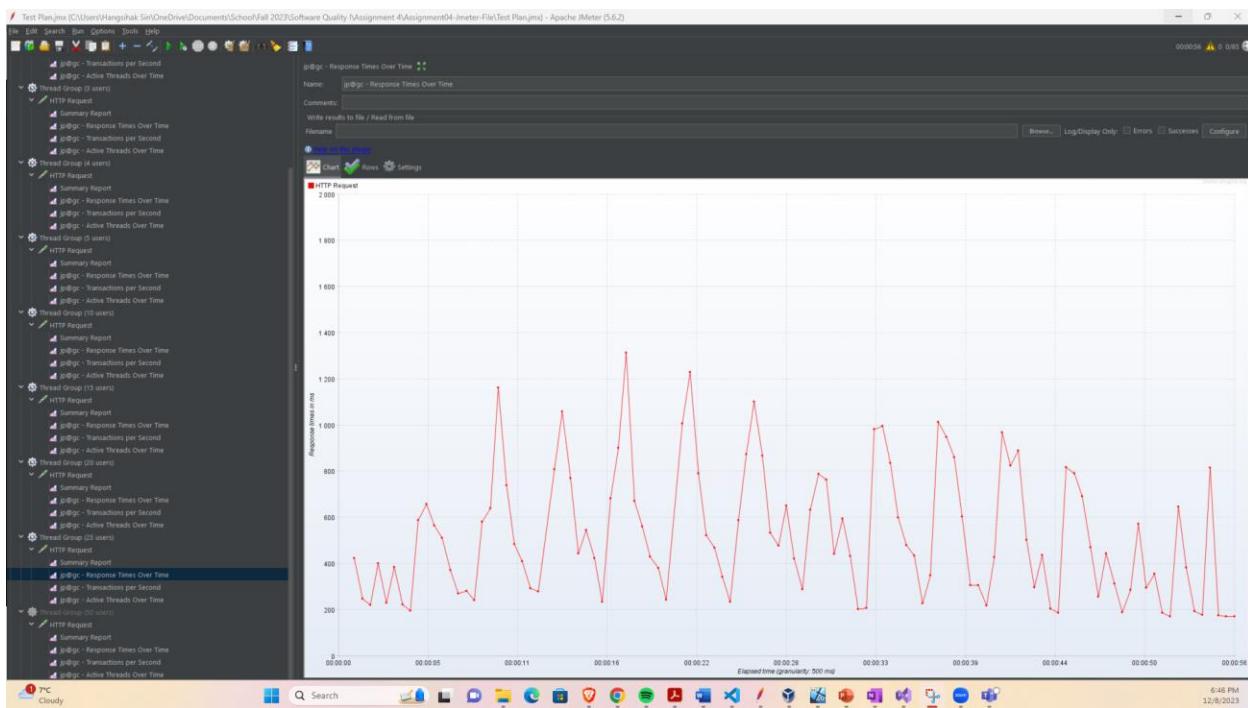


Thread Group (25 users)

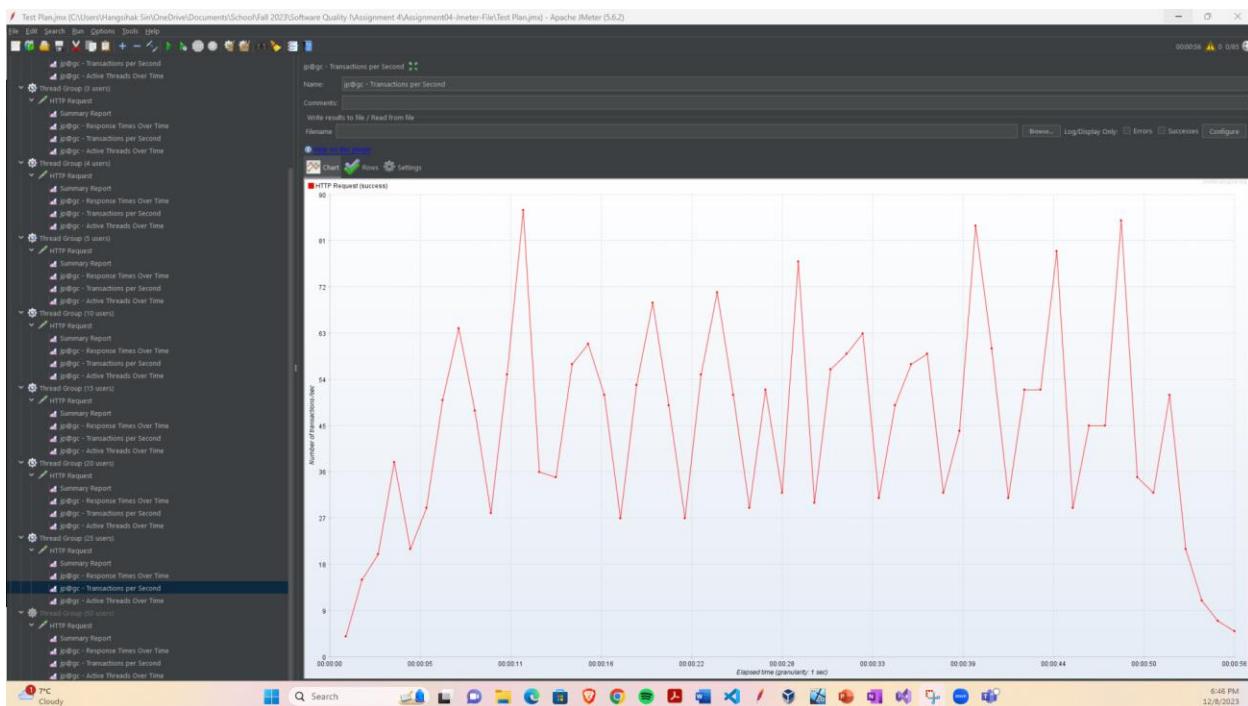
Summary Report



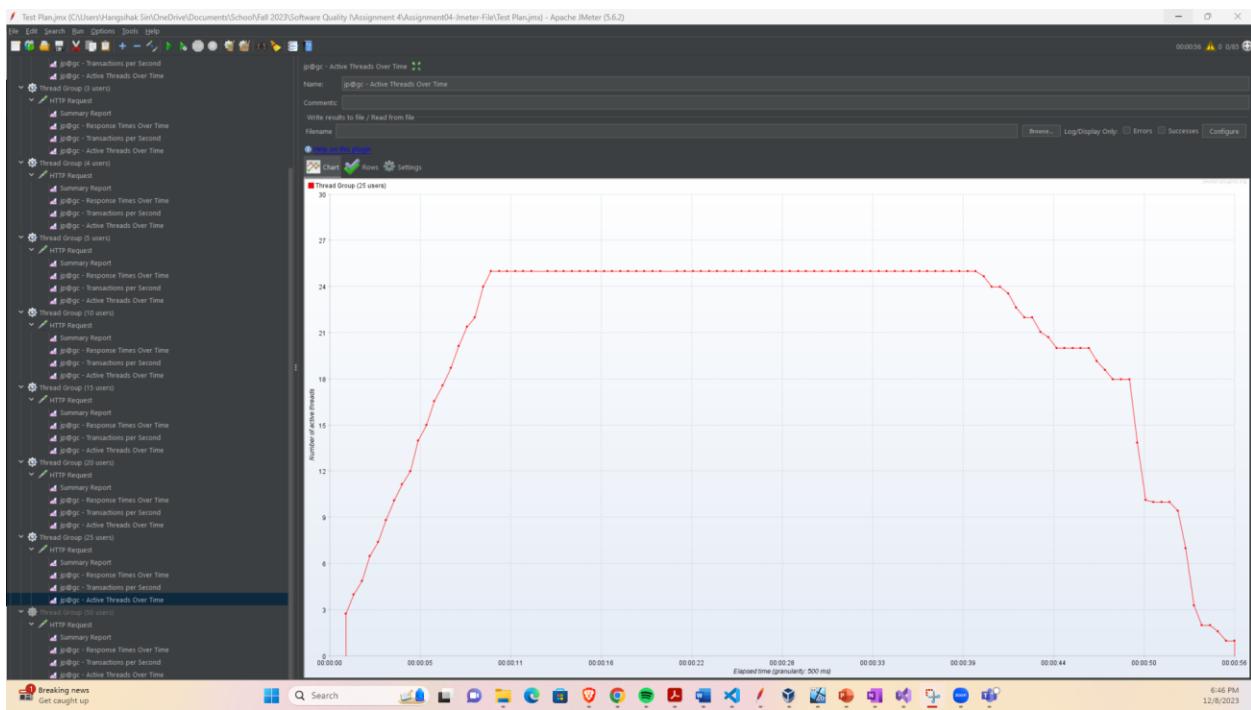
Response Times Over Time



Transactions per Second

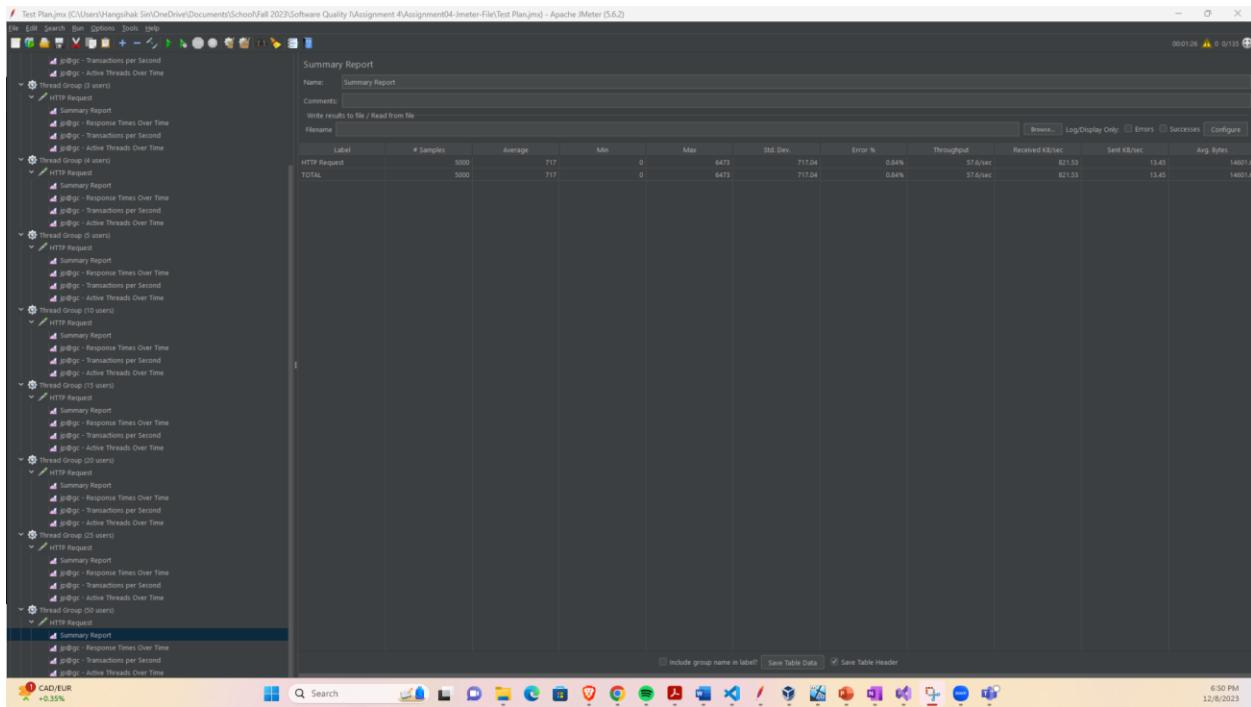


Active Threads Over Time

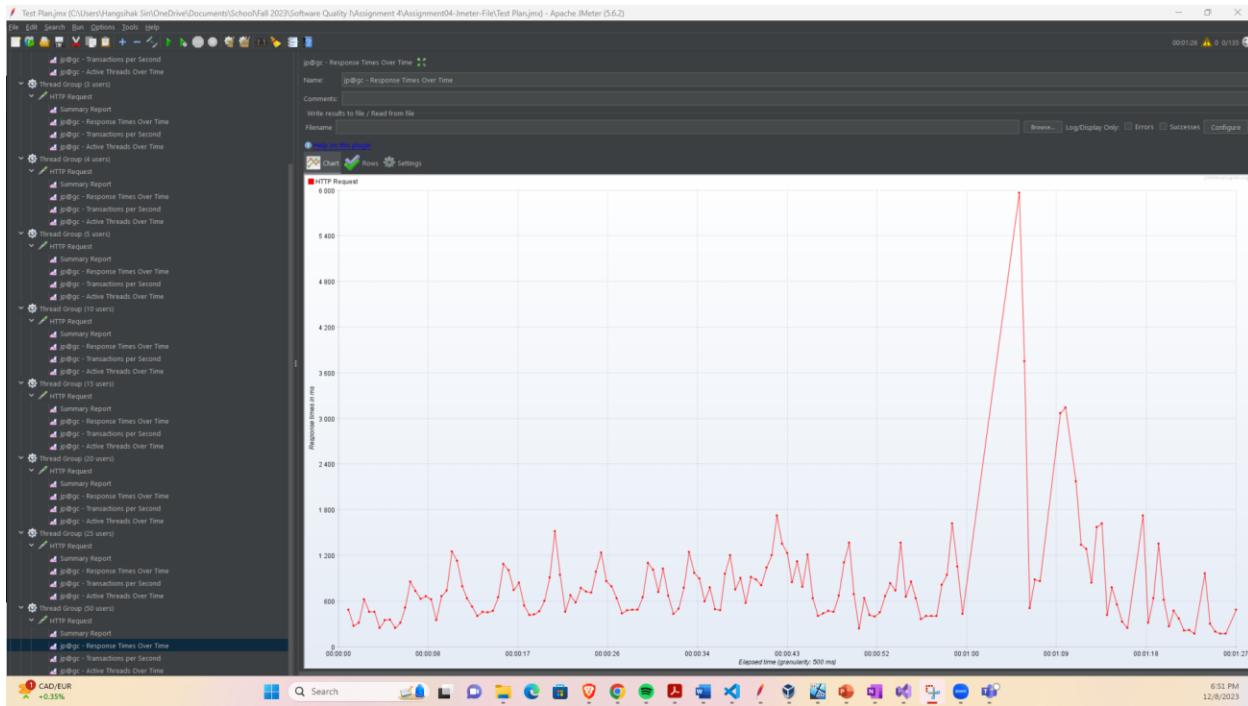


Thread Group (50 users)

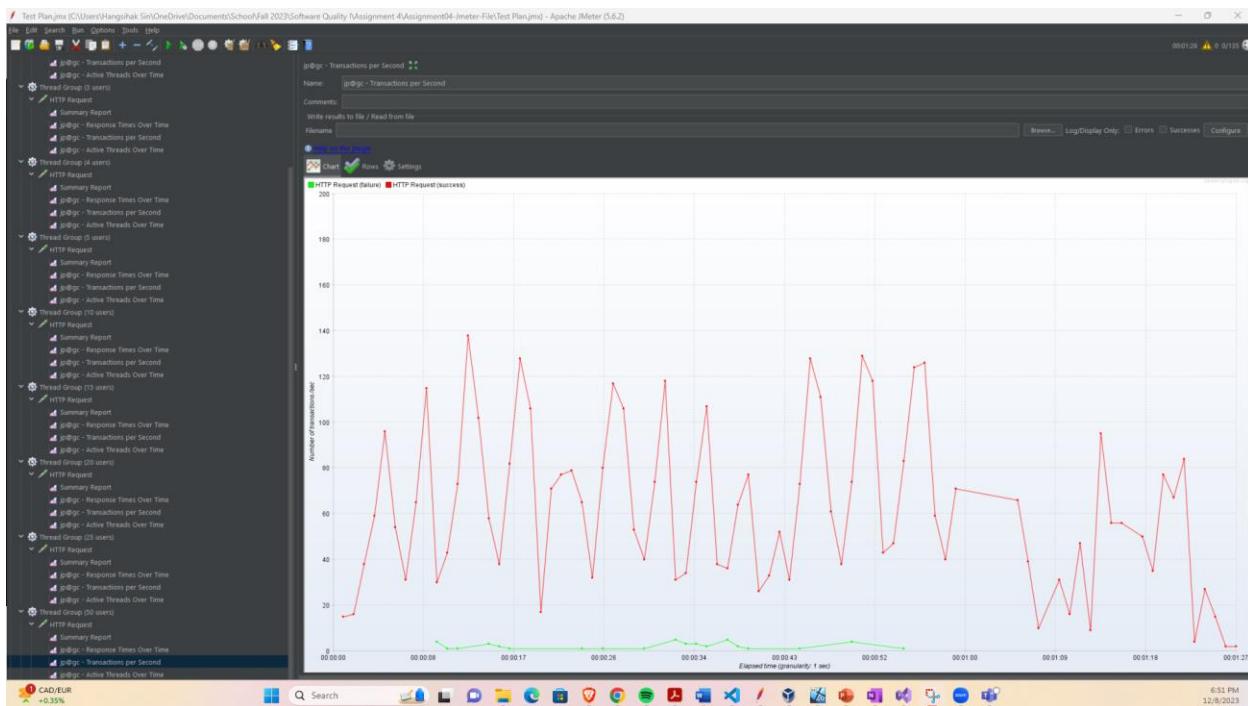
Summary Report



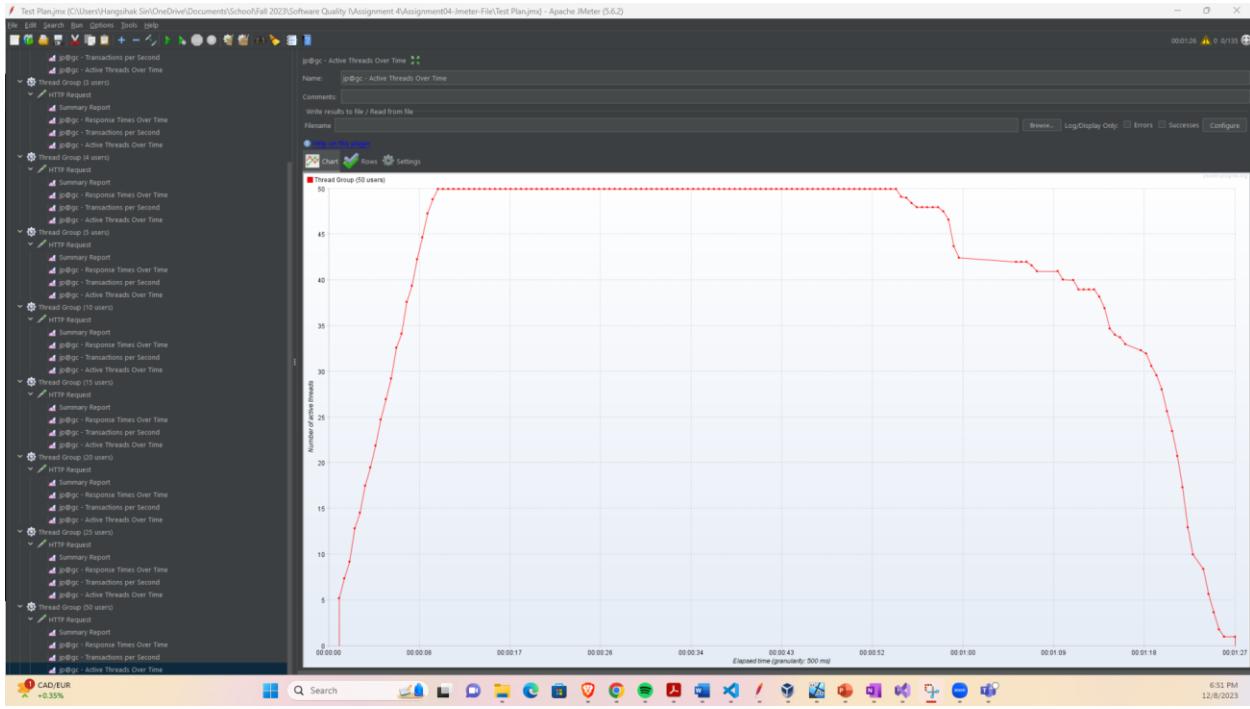
Response Times Over Time



Transactions per Second



Active Threads Over Time



Final Observation

Response times:

Based on the lecture presentation, response time refers to the time it takes – usually not but not always elapsed time – for the system to respond to a user's request after an interaction. In the response time graph on the y-axis presented in milliseconds and x-axis presented in the elapsed time. When conducting the JMeter test, it is apparent that as the number of threads (users) increases from 1 to 50. Initially, with a low number of threads (let's say 1 user), the response time tends to be low. This is because the system has fewer requests to handle. However, as the number of threads increases (1 to 50), the response times show an upward trend, indicating a greater load and stress on the system which may be due to factors such as resource limitations or bottlenecks.

Transactions per second:

Transaction per second accounts for the number of processes completed in one second. In other words, it is calculated based on how many transactions are being executed over a certain time period of the test and then calculate it for a second (Loadium Wiki, n.d.). There is a correlation between Transaction per second and response times (Loadium Wiki, n.d.). This correlation applies to both virtual users and response times, which can be calculated using the formula below:

$$\text{TPS} = \frac{\text{VirtualUser}}{\text{responsetime} + \text{thinktime}}$$

In the JMeter test for Transactions per second (TPS), a recurring pattern became obvious. Initially, TPS graph with a single thread (1 user) climbs steadily in the first few seconds, reaching a peak. However, there are an intermittent drop, possibly due to resource limitations. As the test progresses through different thread groups (2 to 50), TPS gradually increases until it peaks, indicating system capacity. Beyond this peak, TPS decreases possibly due to resource overload and increased competition among threads.

Observation on the Active threads over time:

Active Threads over Time is a listener that provides a representation of the number of active threads in each thread group throughout the test run (JMeter, n.d.).

In the JMeter test for Active threads over time, the graph would gradually increase during the ramp-up period, reflecting the growing number of threads. At the end, it would drop signifying that it has finished. Notably, this observed pattern remains to be consistent throughout most of the thread groups, from 1 to 50 users.

It is important to note that the ramp-up period and loop count have an impact on the metrics. A longer ramp-up period results in smoother load increases but in our case all the thread groups are set to 10 sec long, whereas the loop count determines how long threads remain active are set to 100. These factors contribute to the overall performance of the system.

