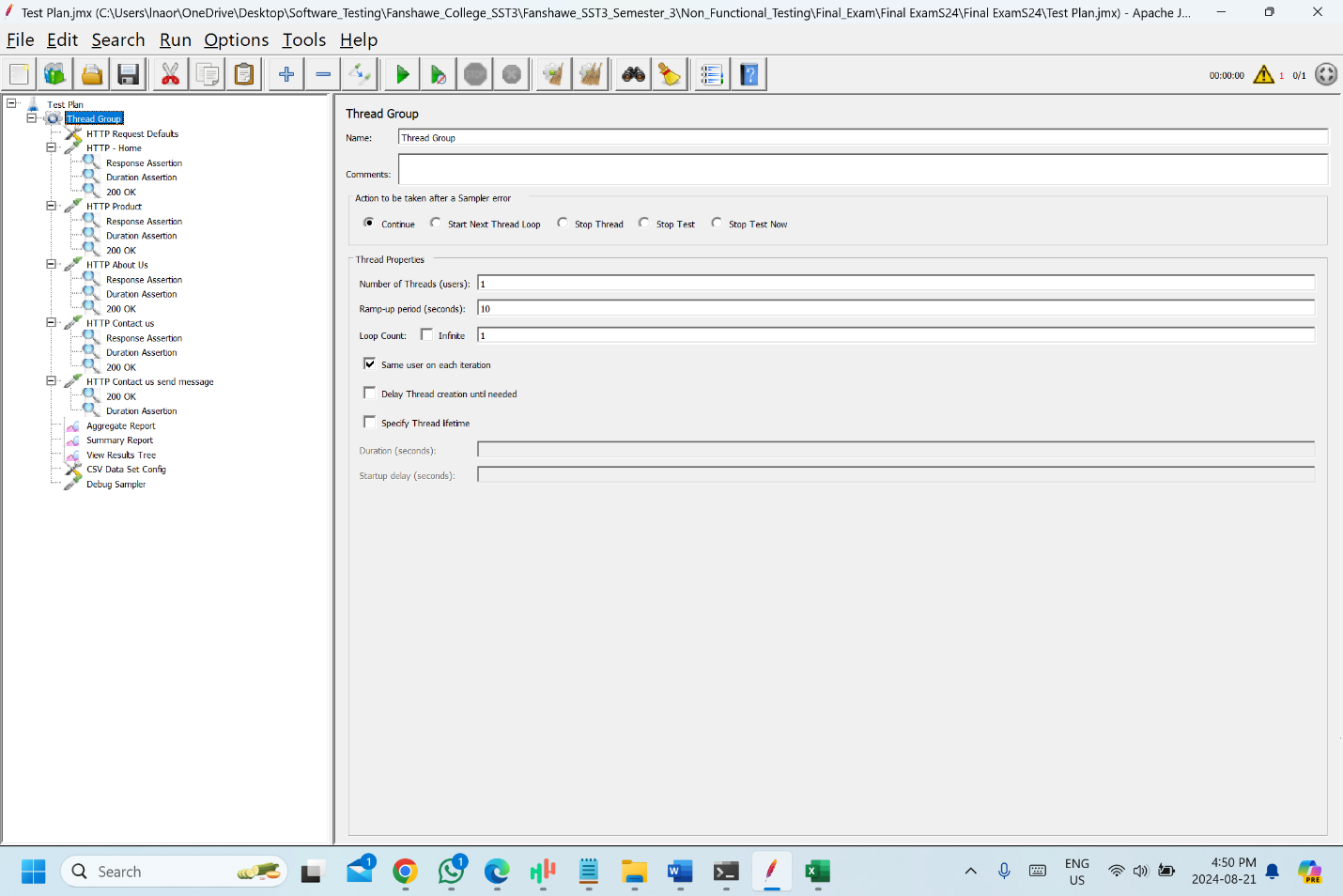
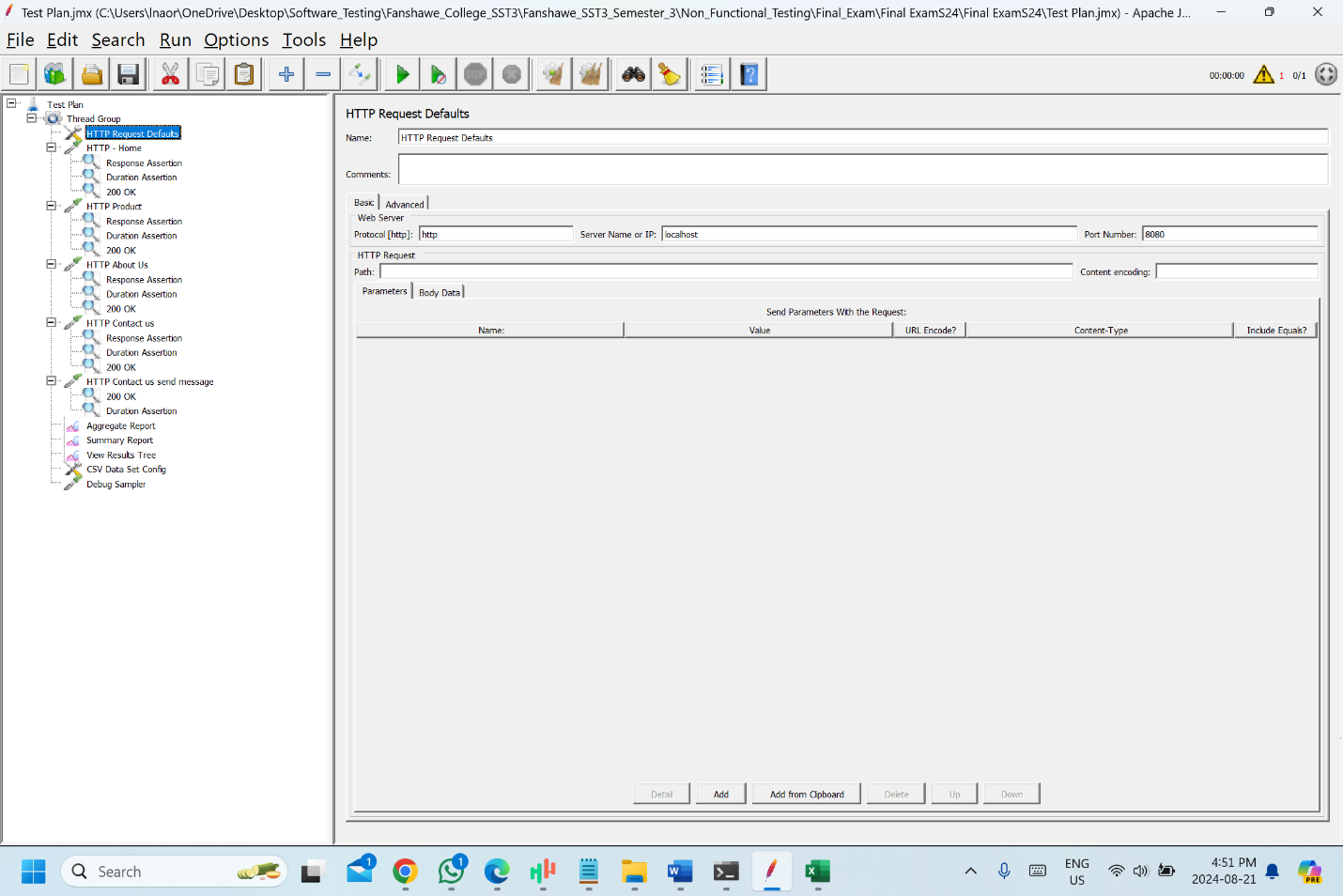
Performance Testing Using JMeter

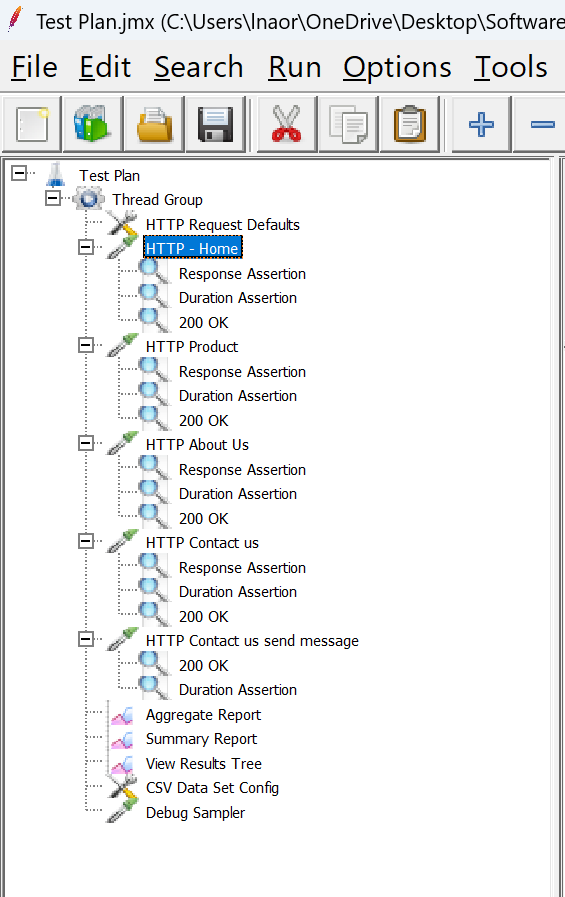
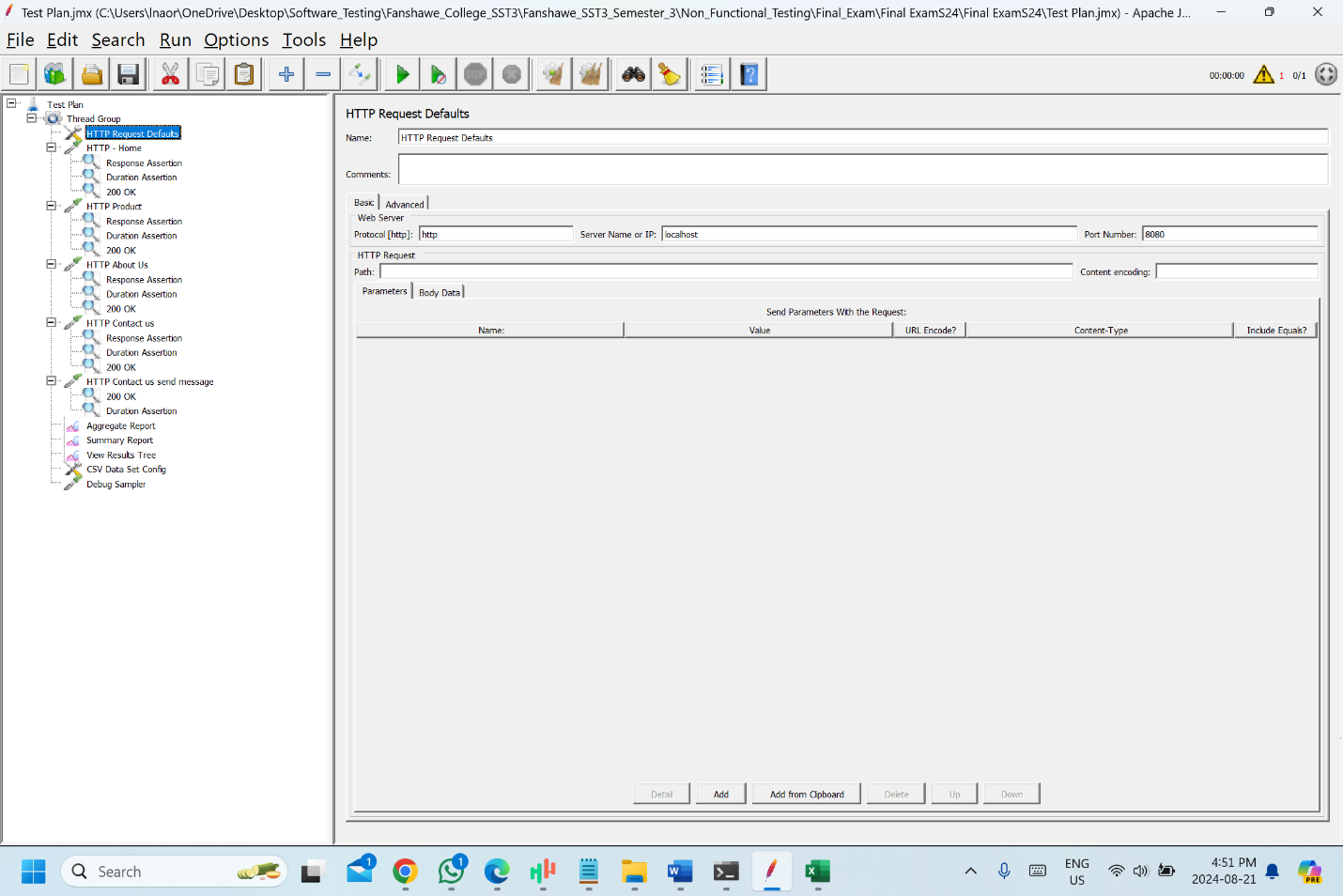
**Thread Group:**



**HTTP request default:**



**Detailed steps for the test plan**



HTTP request home page:

* I made the HTTP request for the home page with the URL “/index.html” and GET method
* The response assertion is making sure that the exact page has been opened with asserting text “Ecommerce HTML Template”
* The duration assertion is making sure that the request is being completed in 10 seconds
* The 200 OK assertion is making sure that the page returns 200 Ok

HTTP request product page:

* I made the HTTP request for the product page with the URL “/products.html” and GET method
* The response assertion is making sure that the exact page has been opened with asserting text “Featured Items”
* The duration assertion is making sure that the request is being completed in 10 seconds
* The 200 OK assertion is making sure that the page returns 200 Ok

HTTP request about us page:

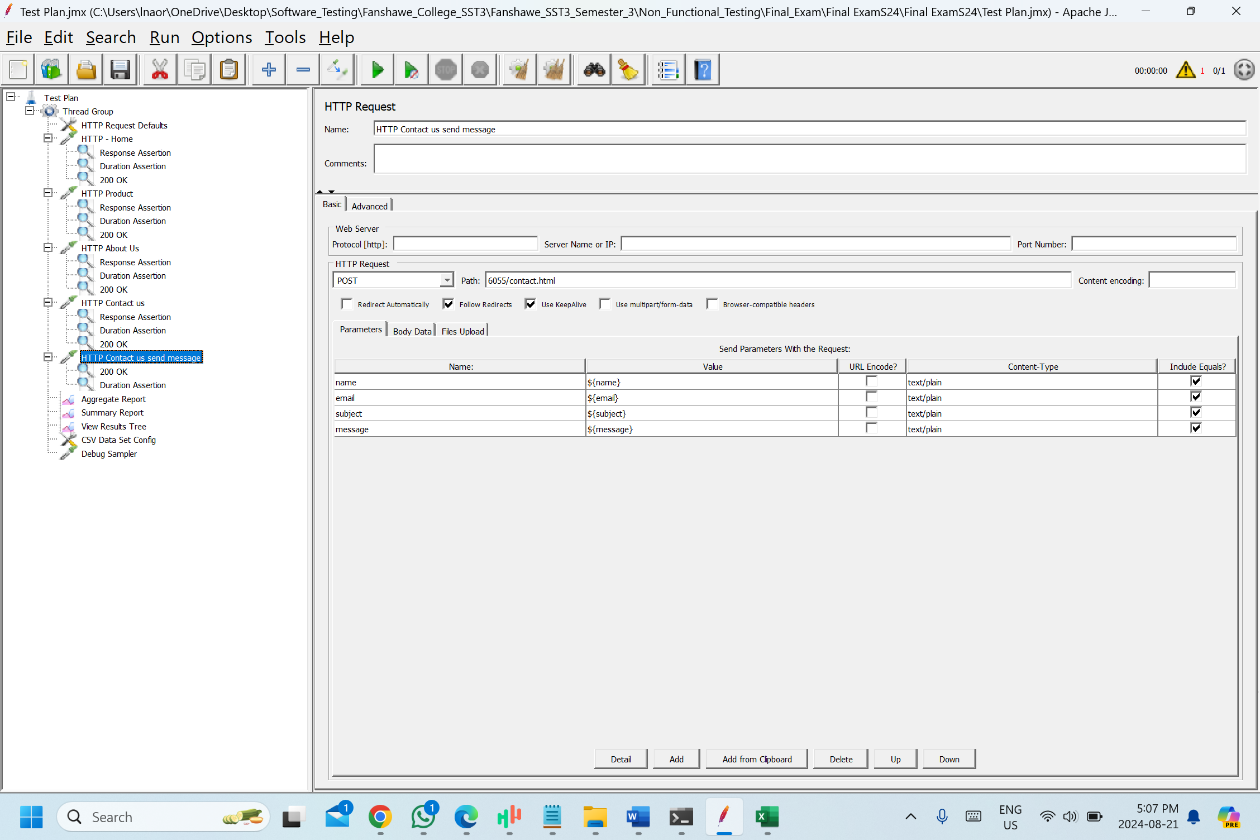
* I made the HTTP request for the about us page with the URL “/about.html” and GET method
* The response assertion is making sure that the exact page has been opened with asserting text “About Us”
* The duration assertion is making sure that the request is being completed in 10 seconds
* The 200 OK assertion is making sure that the page returns 200 Ok

HTTP request Contact Us page:

* I made the HTTP request for the contact us page with the URL “/contact.html” and GET method
* The response assertion is making sure that the exact page has been opened with asserting text “Contact Us”
* The duration assertion is making sure that the request is being completed in 10 seconds
* The 200 OK assertion is making sure that the page returns 200 Ok

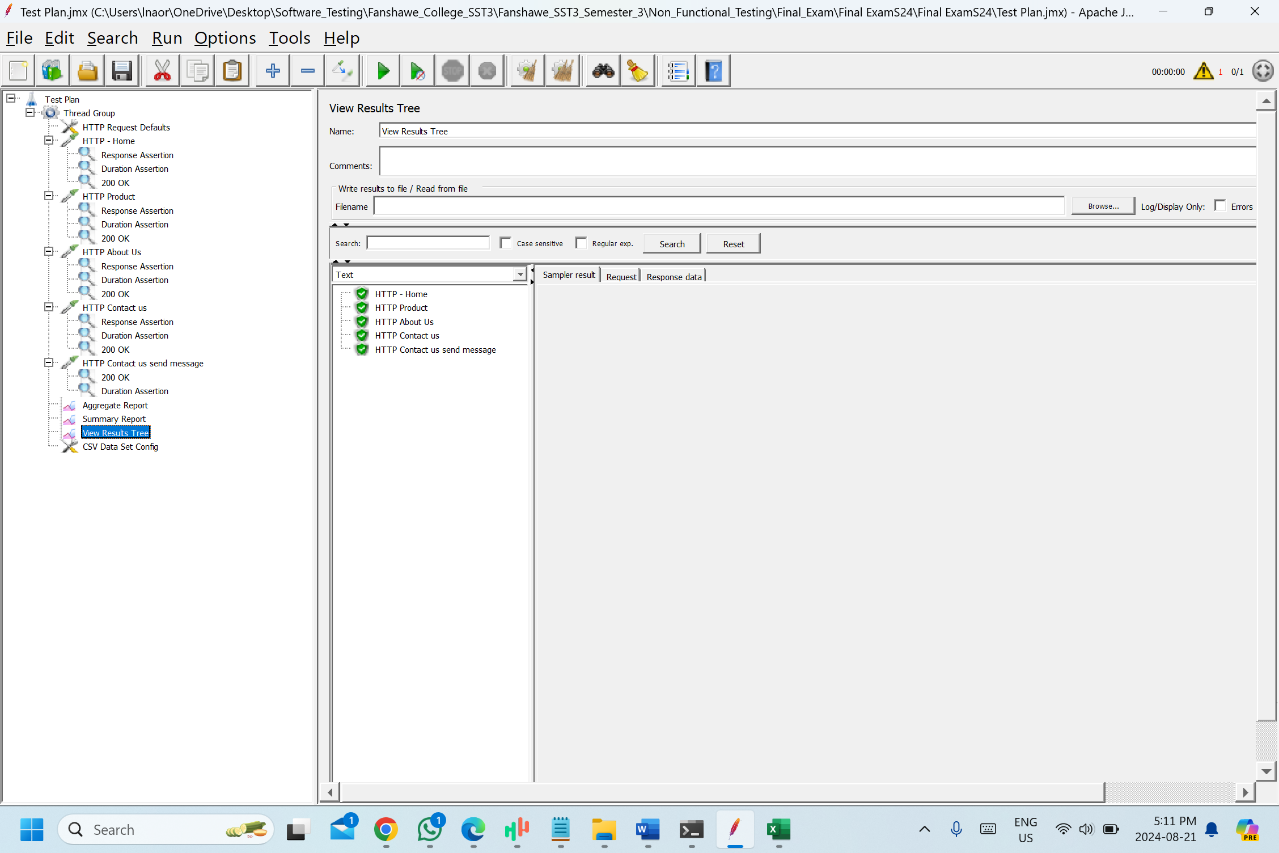
HTTP request contact us send message page:

* I made the HTTP request for the contact page with the URL “/contact.html” and POST method.
* I added the parameters to test the send message functionality by using the CSV file which has different data sets to test “send message”
* The duration assertion is making sure that the request is being completed in 10 seconds
* The 200 OK assertion is making sure that the page returns 200 Ok

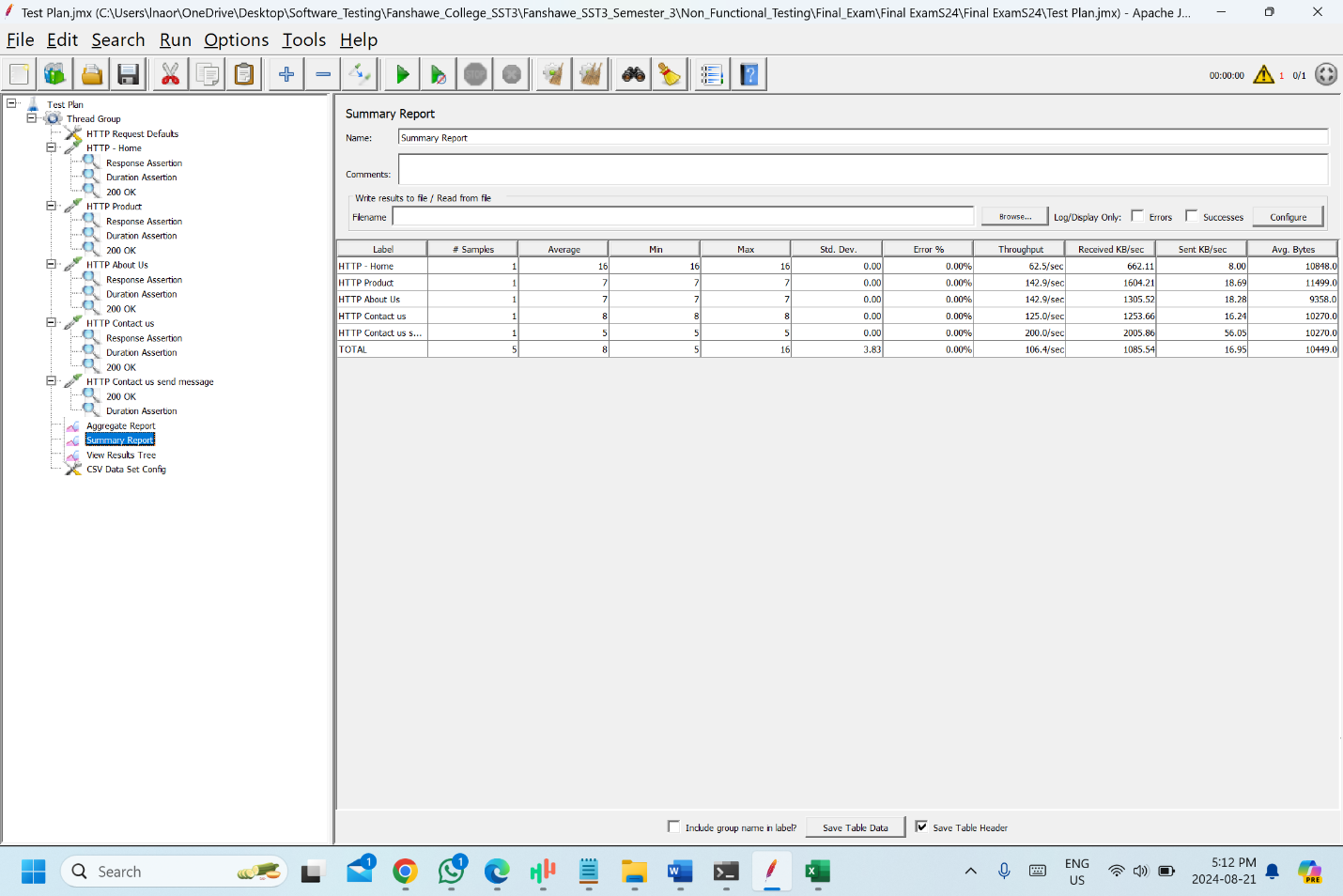
****

**Load Testing:**

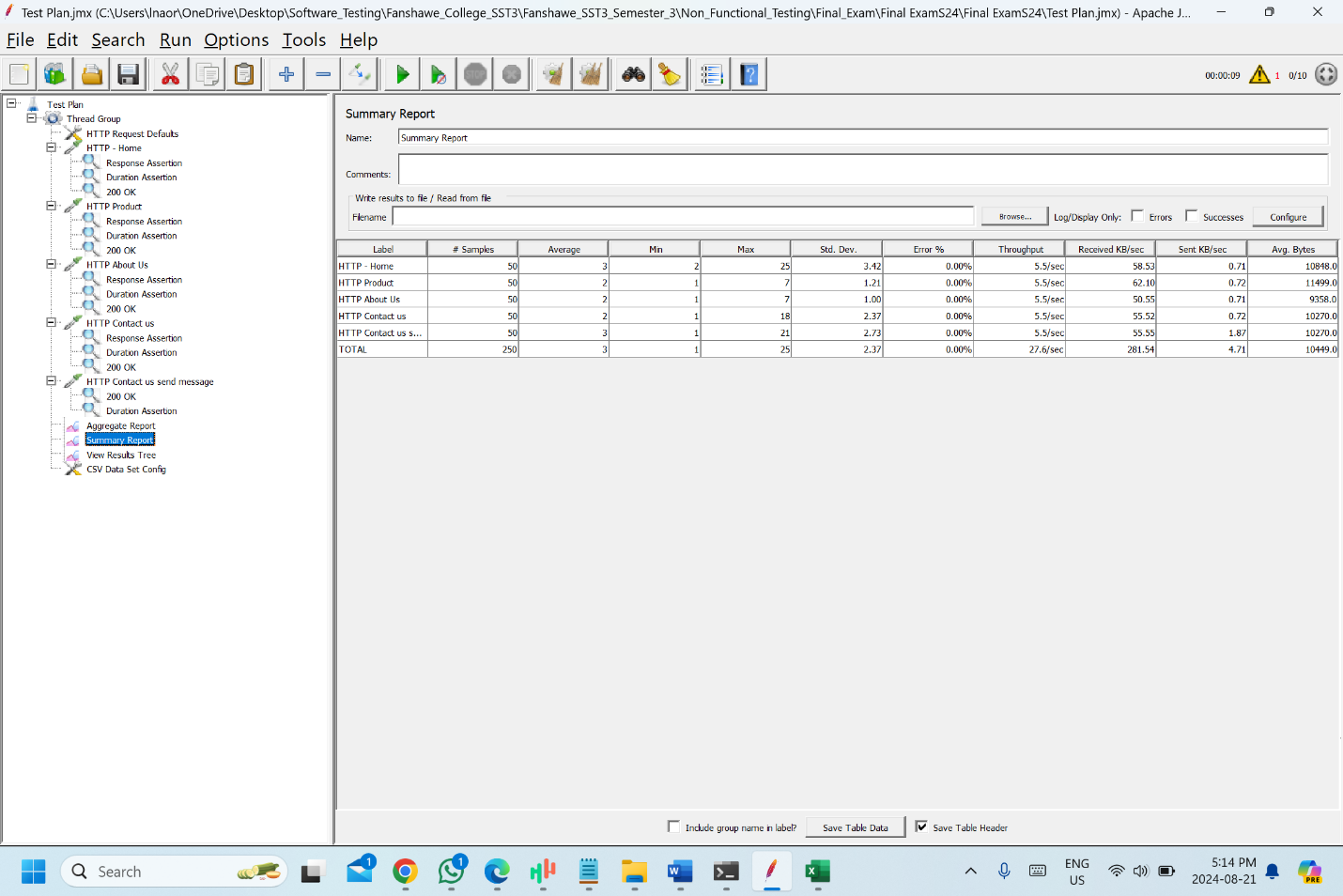
Result tree with 1 user



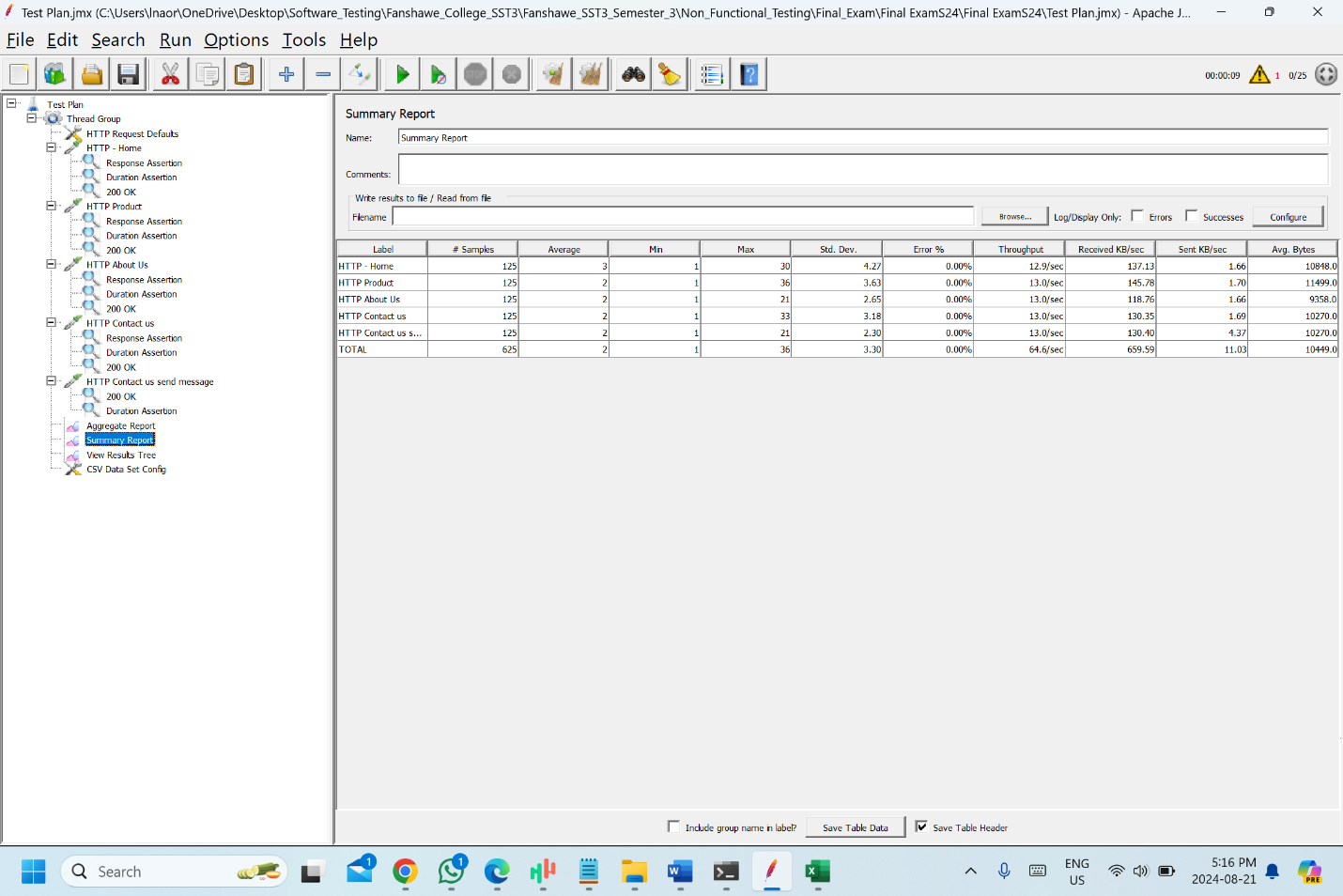
**Summary report with 1 user:**



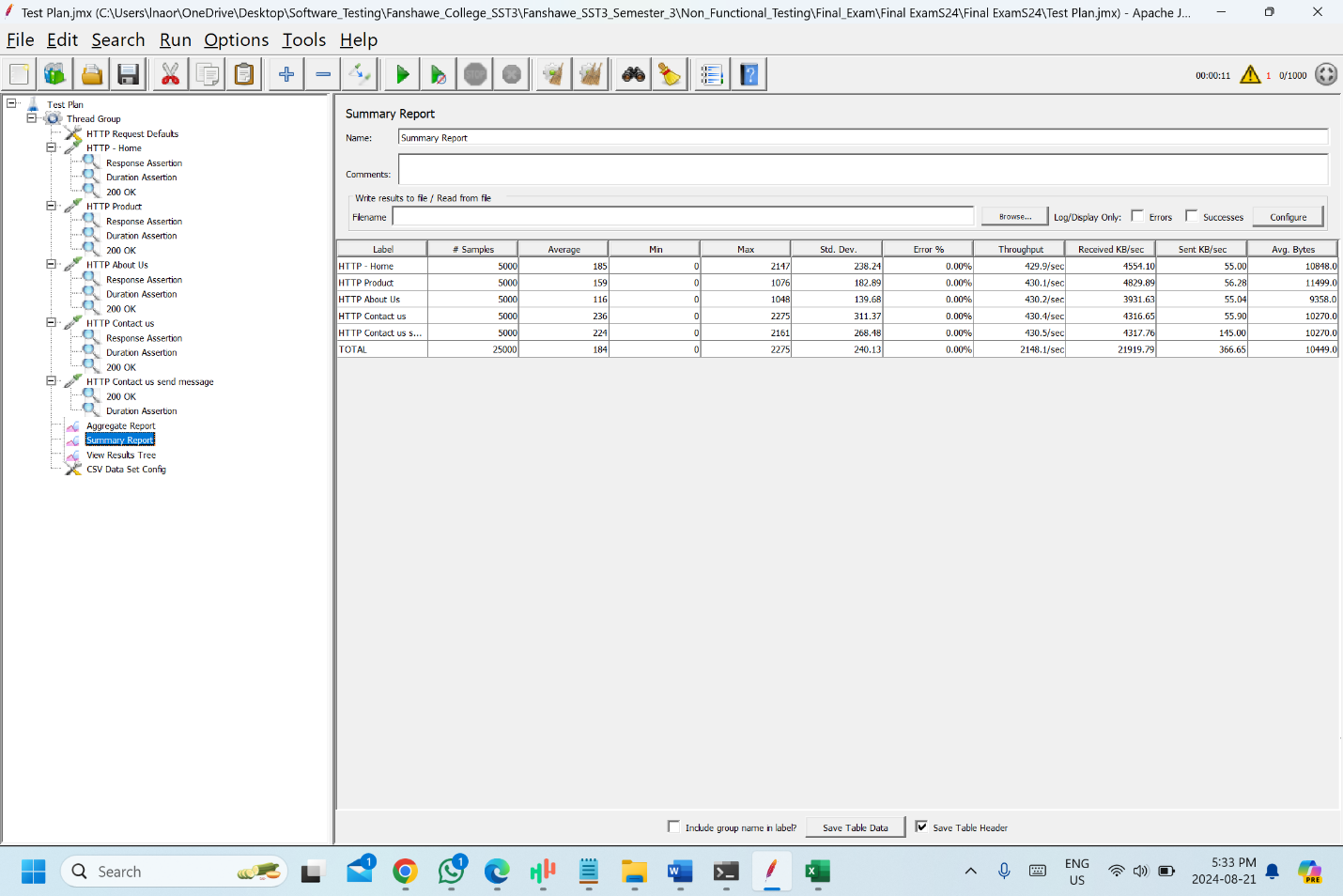
**Summary report with 10 users**



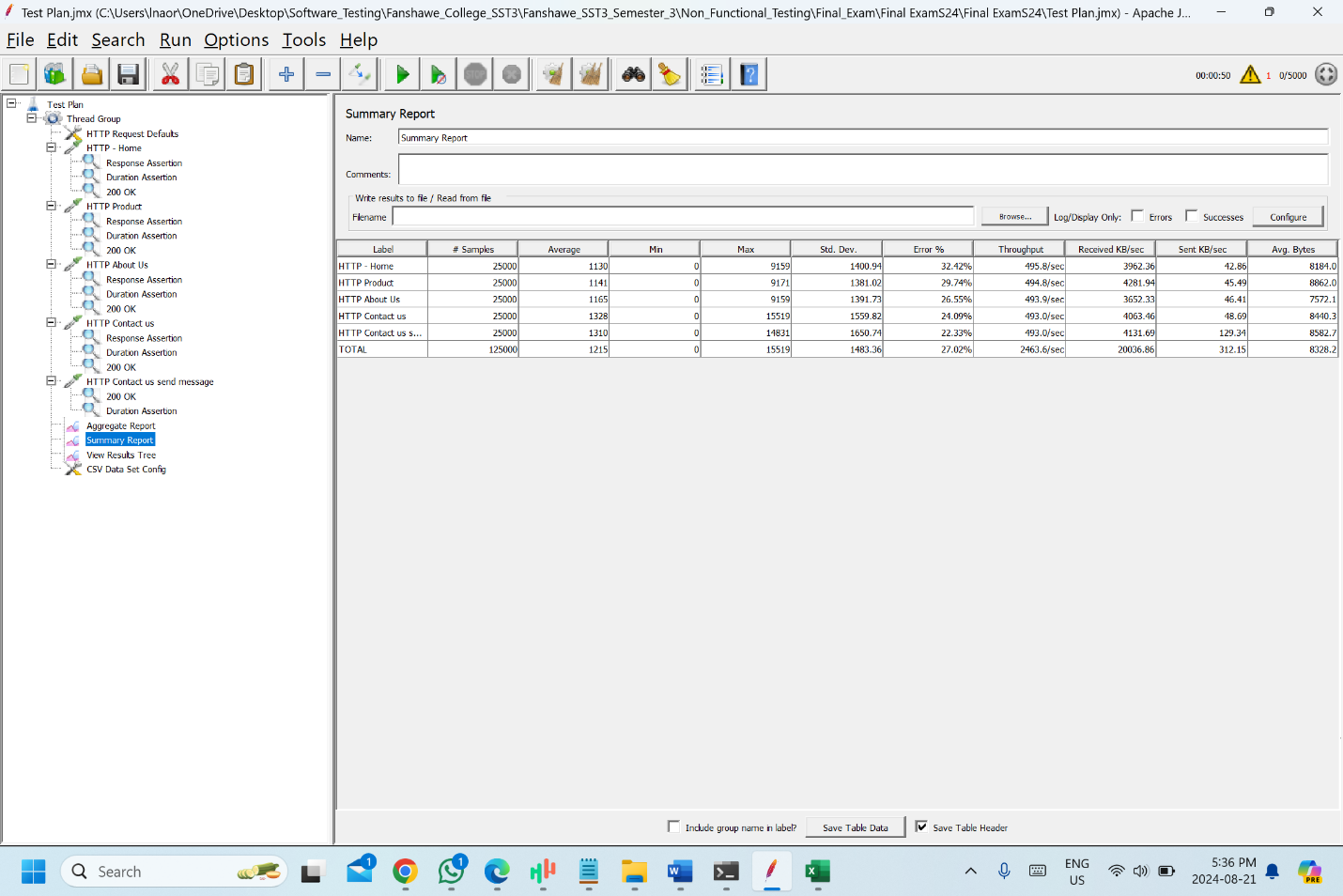
**Summary report with 25 users:**



**Summary report with 1000 users**



**Summary report with 5000**



**How many users can your system support before the request time increases significantly?**

The system supports less than 5000 users before the request time increase significantly, as we can see in the above screenshots. There are some errors when we are testing the system with 5000 users and 5 loop counts.

**Conclusion:**

* The response time increases almost linearly as the number of users increases (doubling the users nearly doubles the response time). This suggests that the system has consistent but poor scalability under increased load.
* The sharp increase in response time when moving from 100 to 1000 users (50 seconds to 430 seconds) indicates a tipping point where the system's performance starts degrading exponentially. This suggests that the system is hitting significant bottlenecks or resource saturation at higher loads.
* The system likely reaches its effective capacity somewhere between 1000 and 5000 users. Beyond this point, performance degrades drastically, indicating the system is struggling to handle the load efficiently.

Summary:

* Implement load balancing strategies or scale the application to distribute the load more effectively.
* Run stress tests to determine the exact point of failure or bottleneck, and focus on optimizing those specific areas
* Reevaluate the application architecture, especially if handling high concurrency is critical