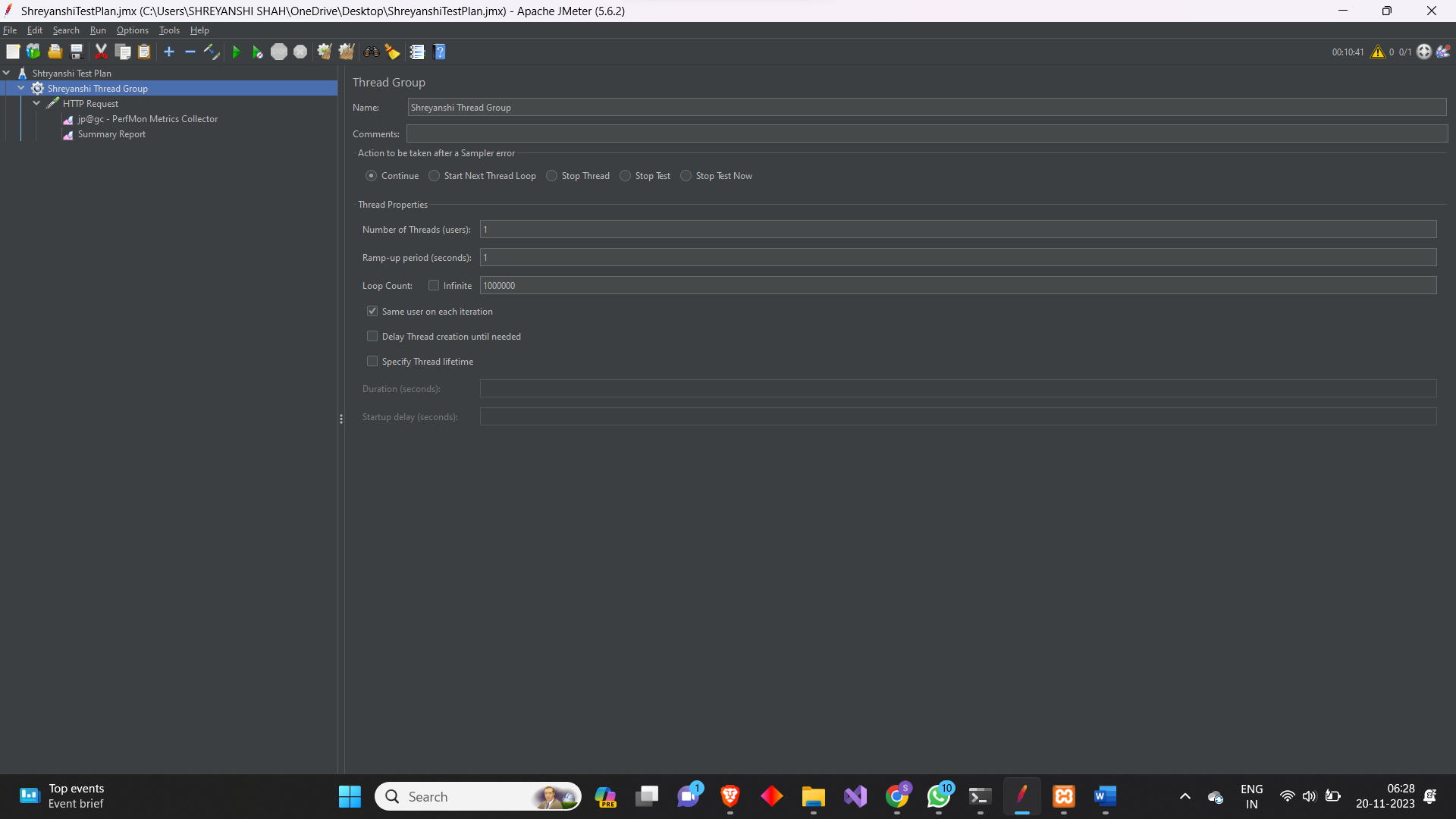
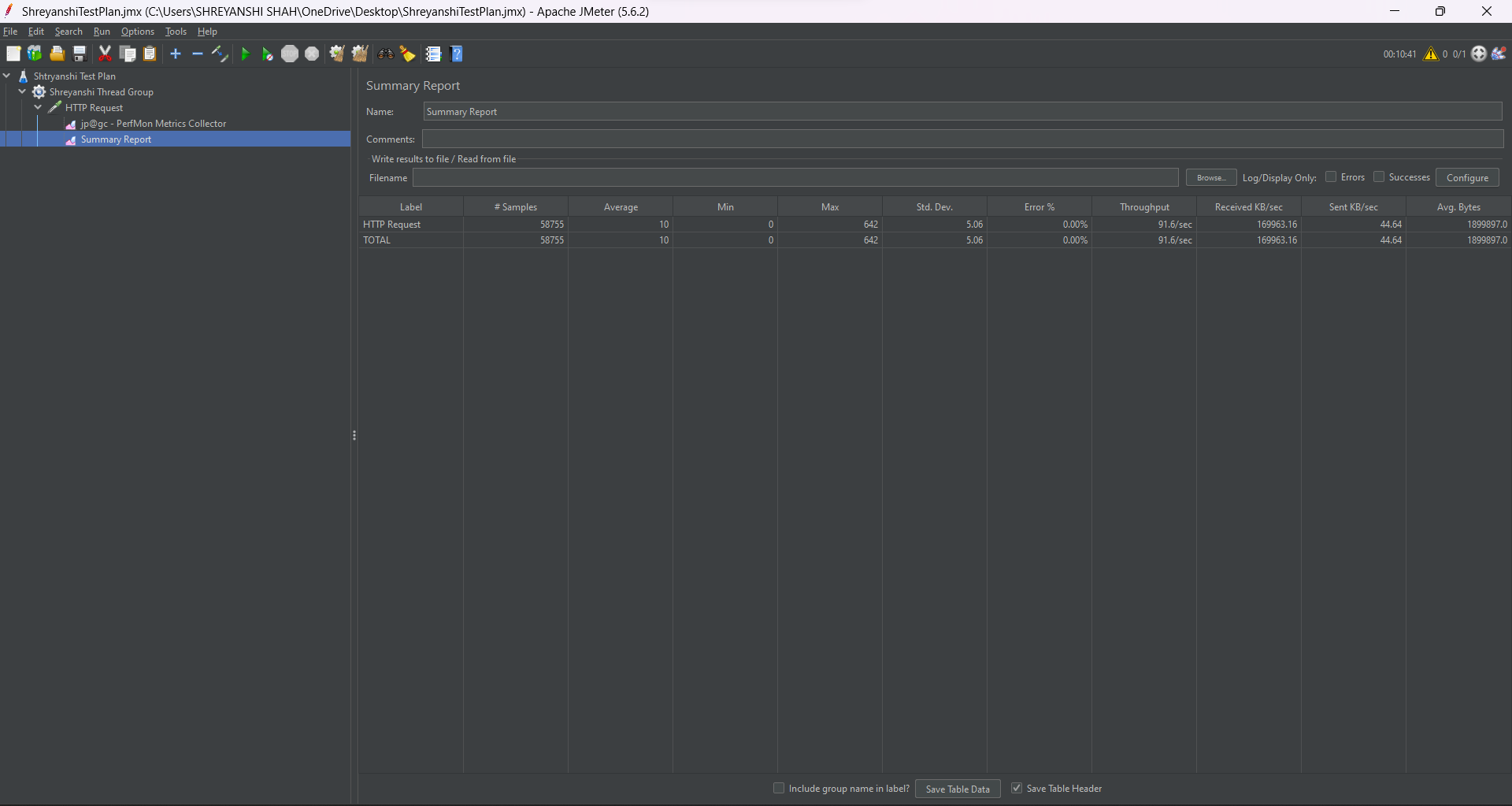
Performance Testing

1. *Test Plan 1*

*Number of users: 1*

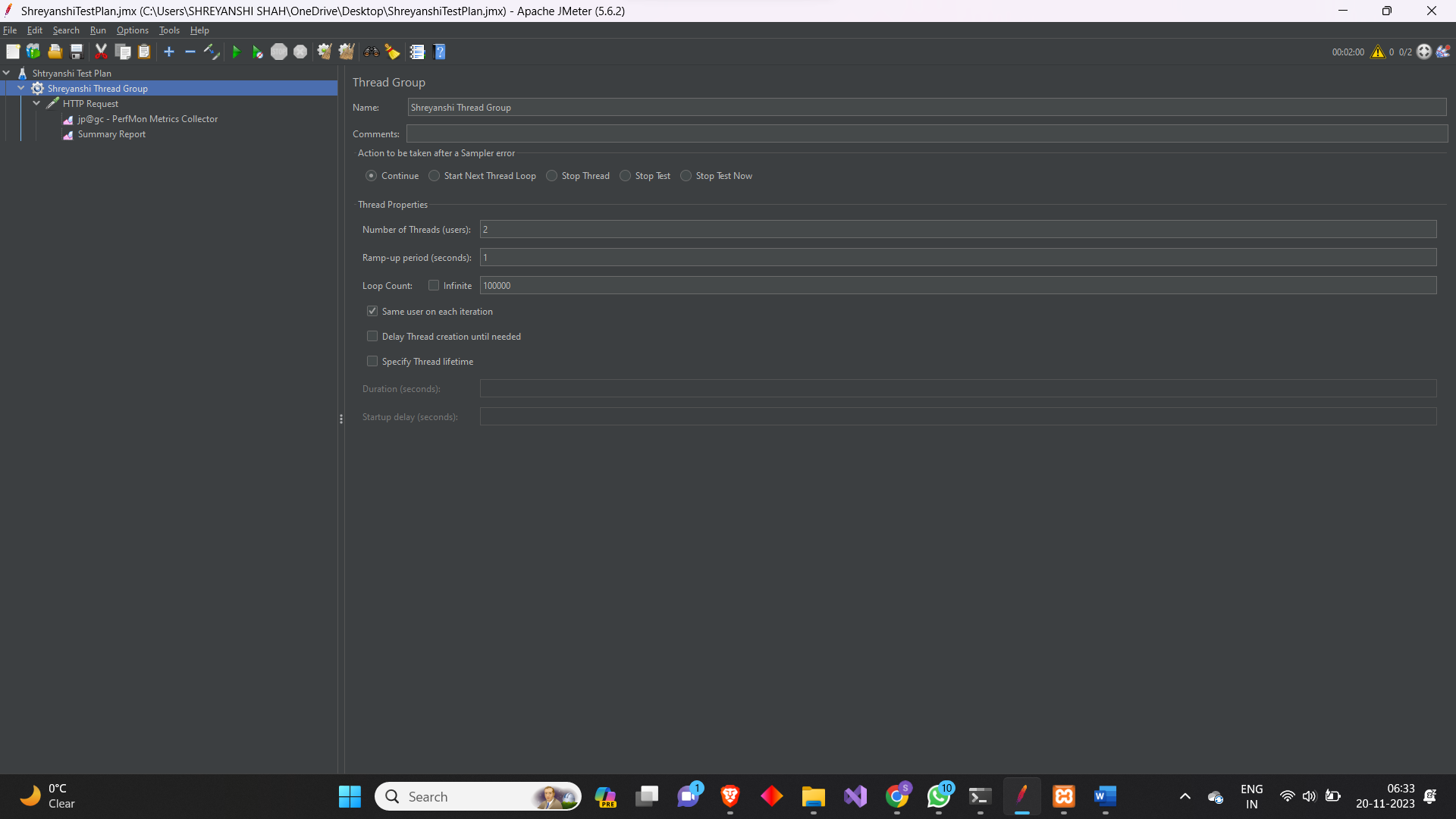
*Loop count: 1000000*

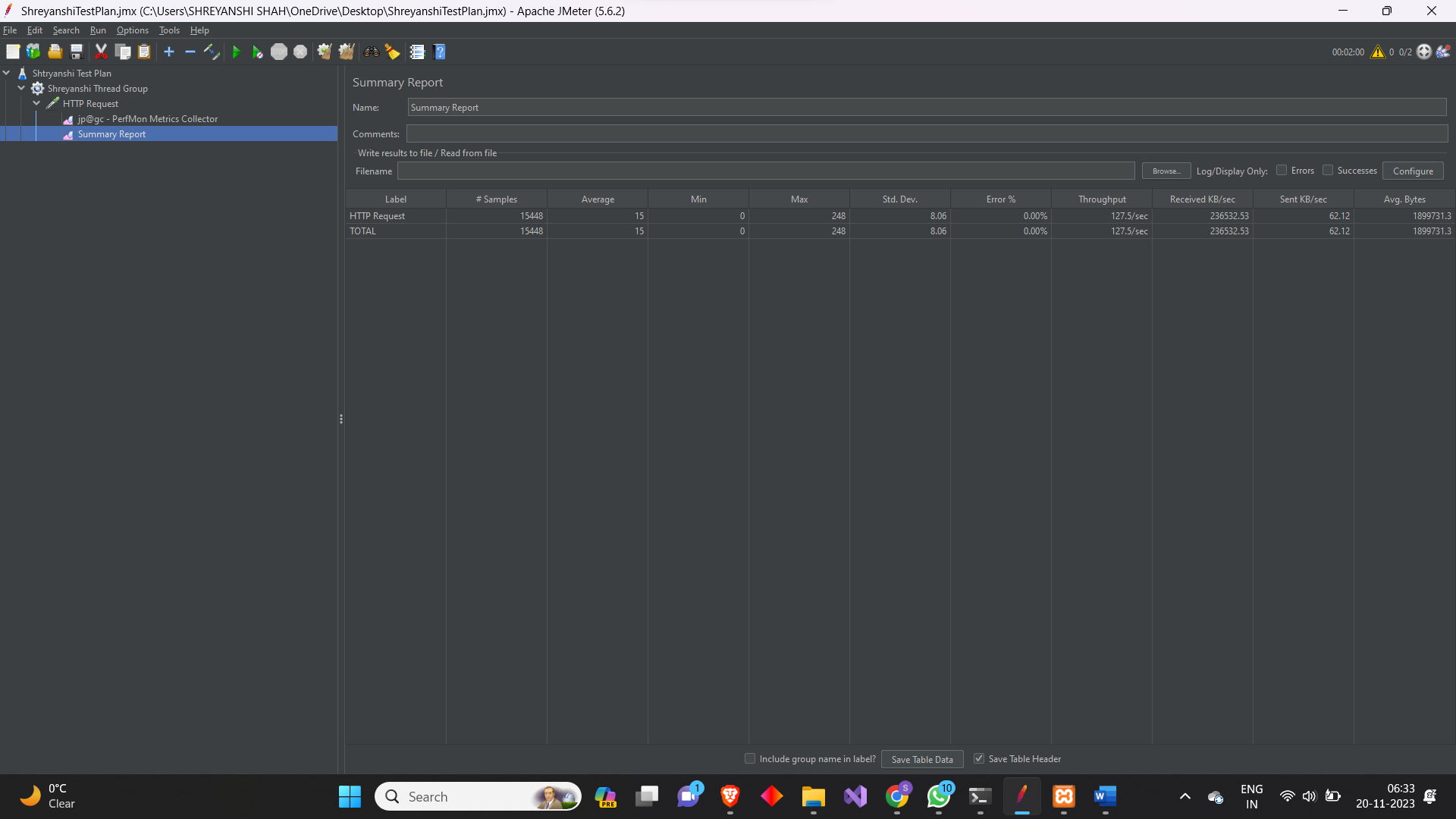
*Summary Report:* **

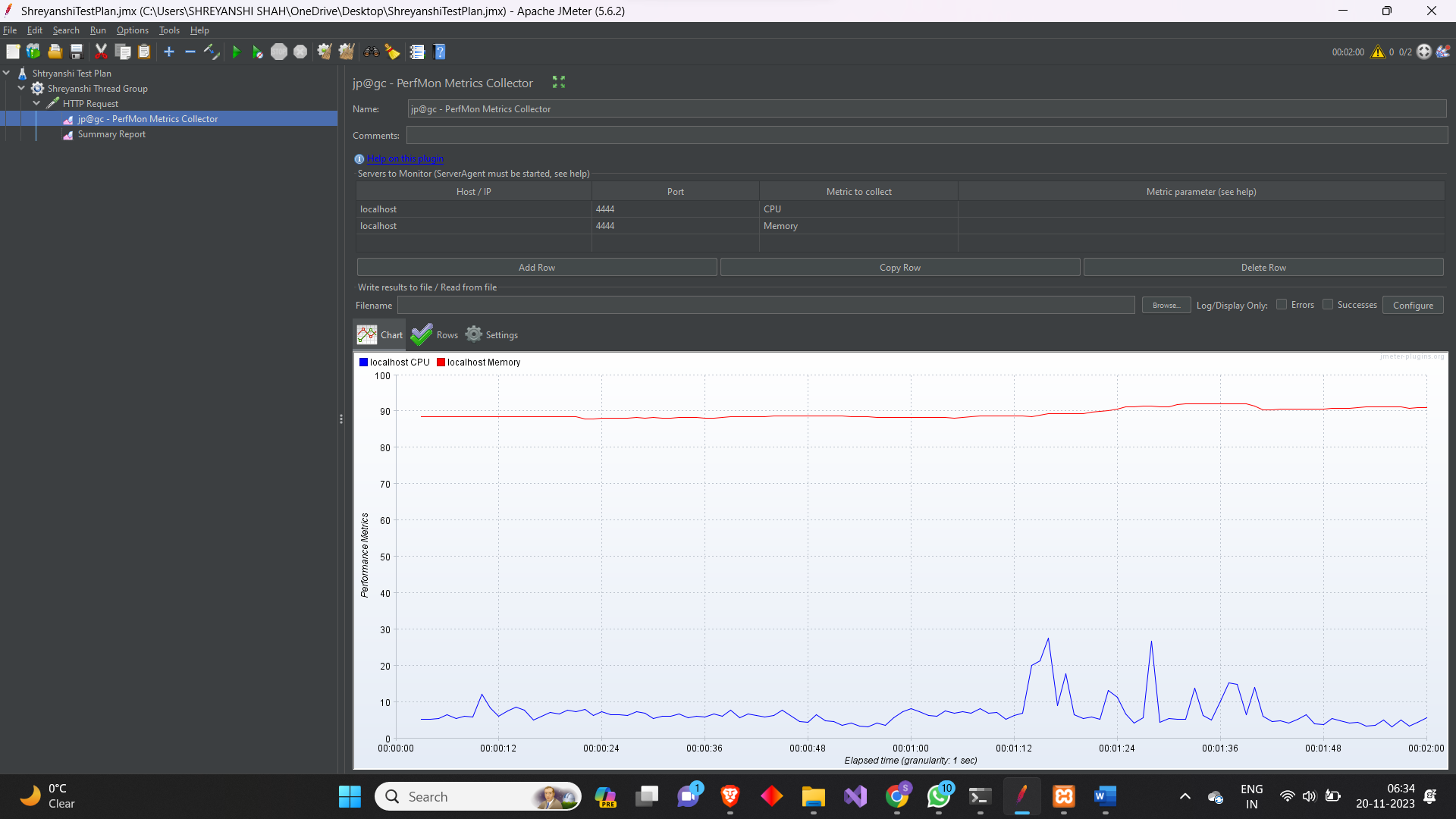
*Plot of CPU and Memory usage:* **

1. *Test Plan 2*

*Number of Users: 2*

*Loop Count: 100000*

*Summary Report:*

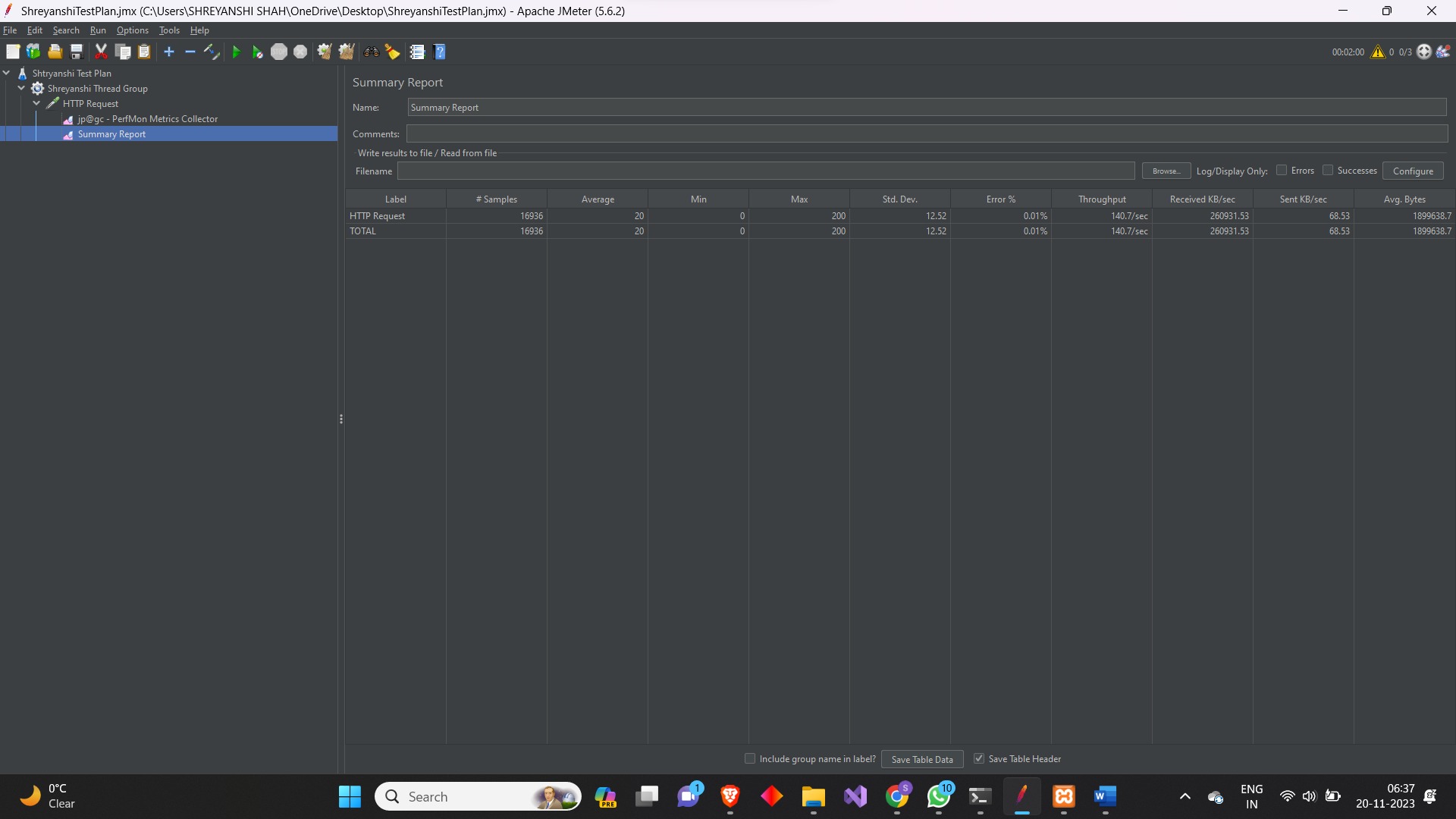
*Plot of CPU and Memory usage:*

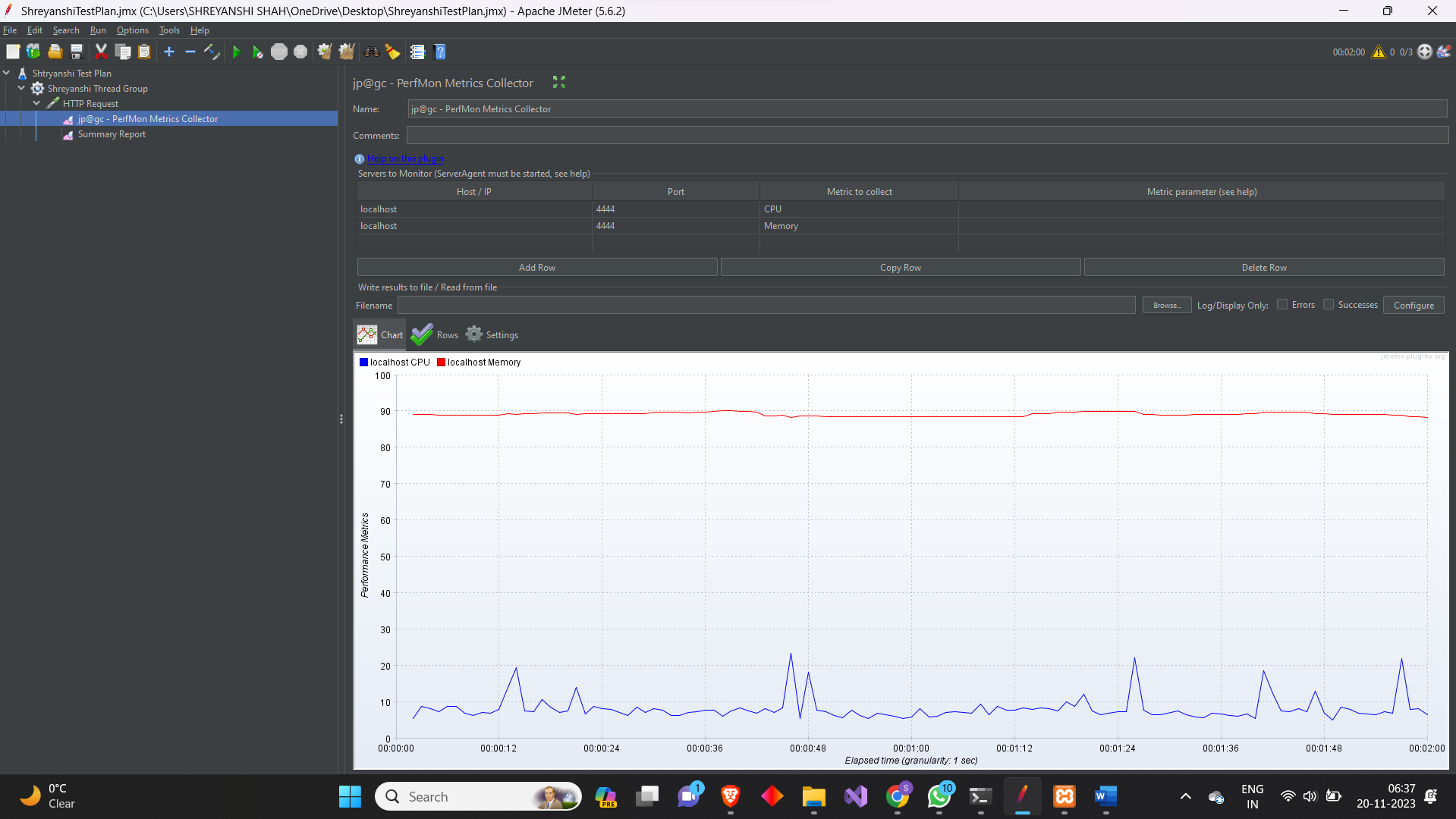
1. *Test Plan 3*

*Number of Users: 3*

*Loop Count: 110000*

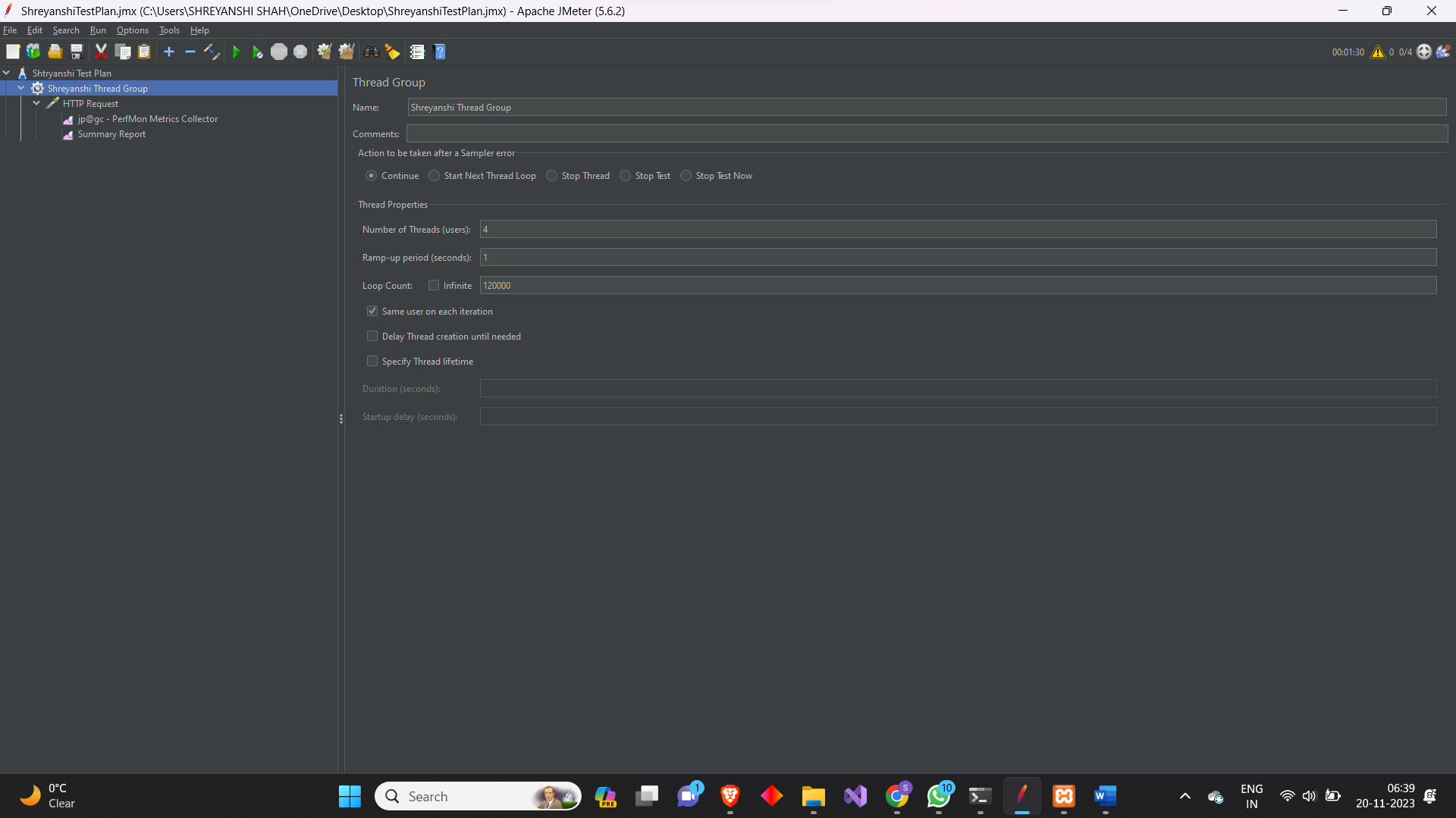
*Summary Report:*

**

*Plot of CPU and Memory usage:*

1. *Test Plan 4*

*Number of Users: 4*

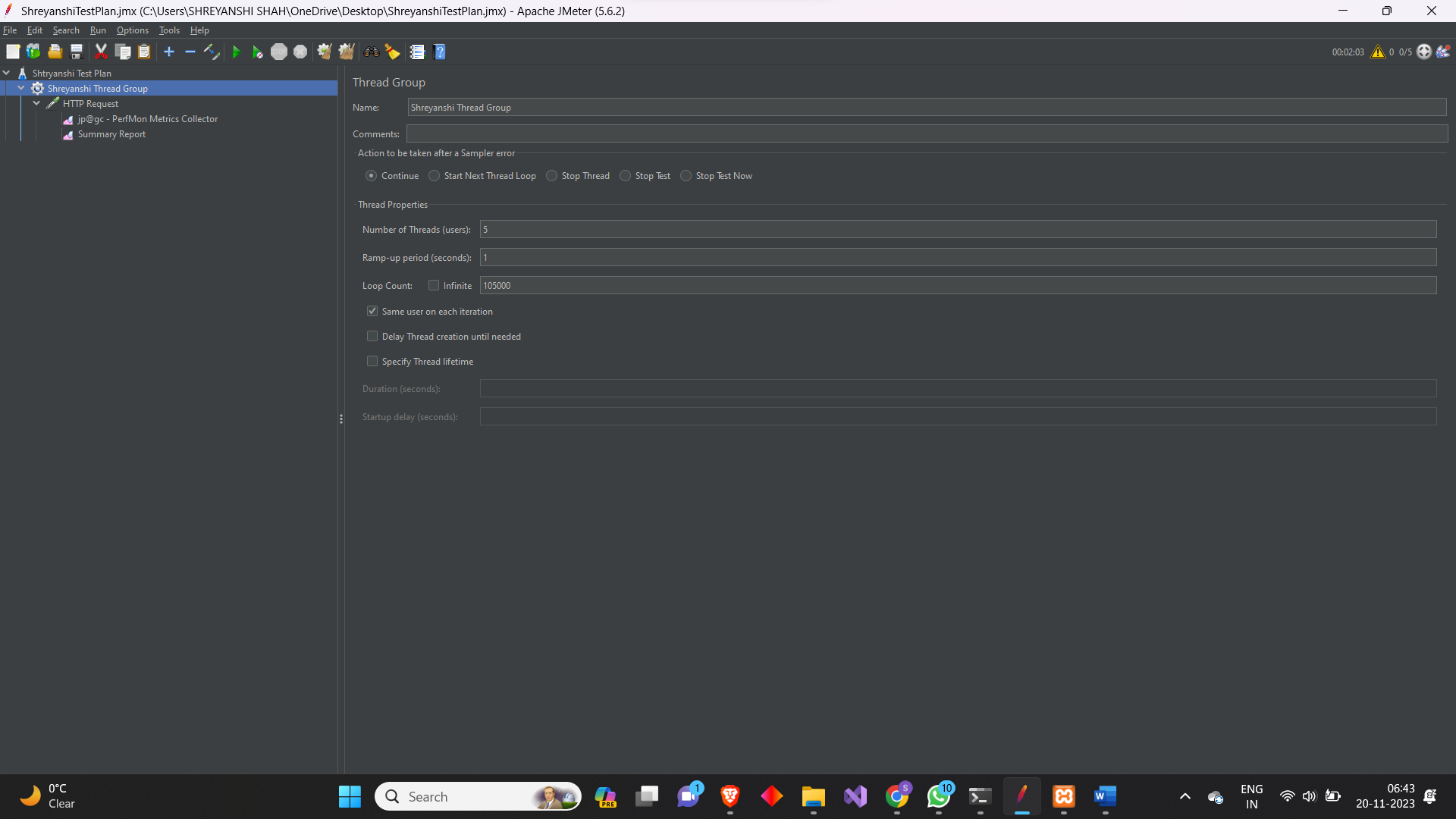
*Loop Count: 120000*

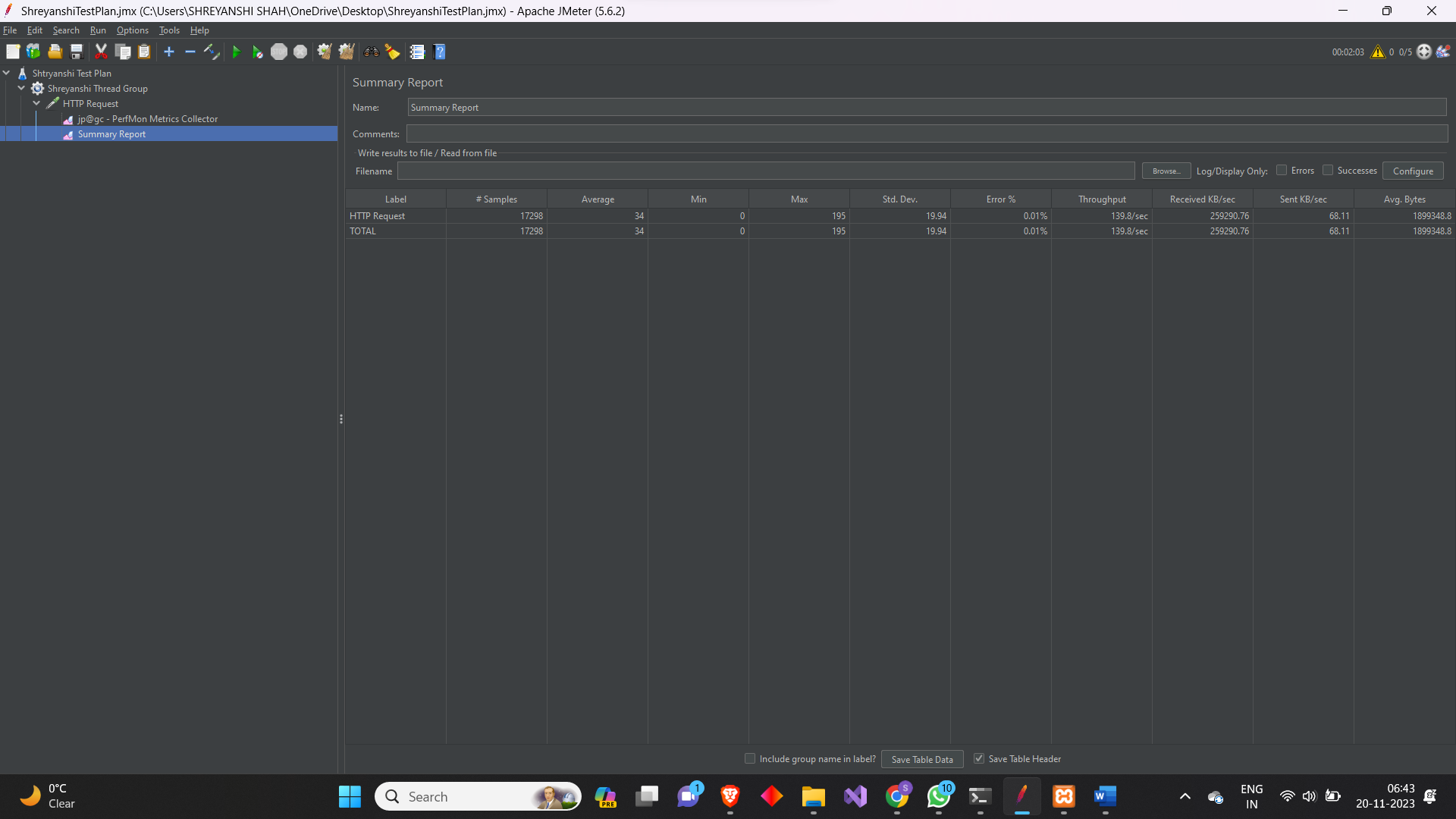
*Summary Report:*

*Plot of CPU and Memory usage:*

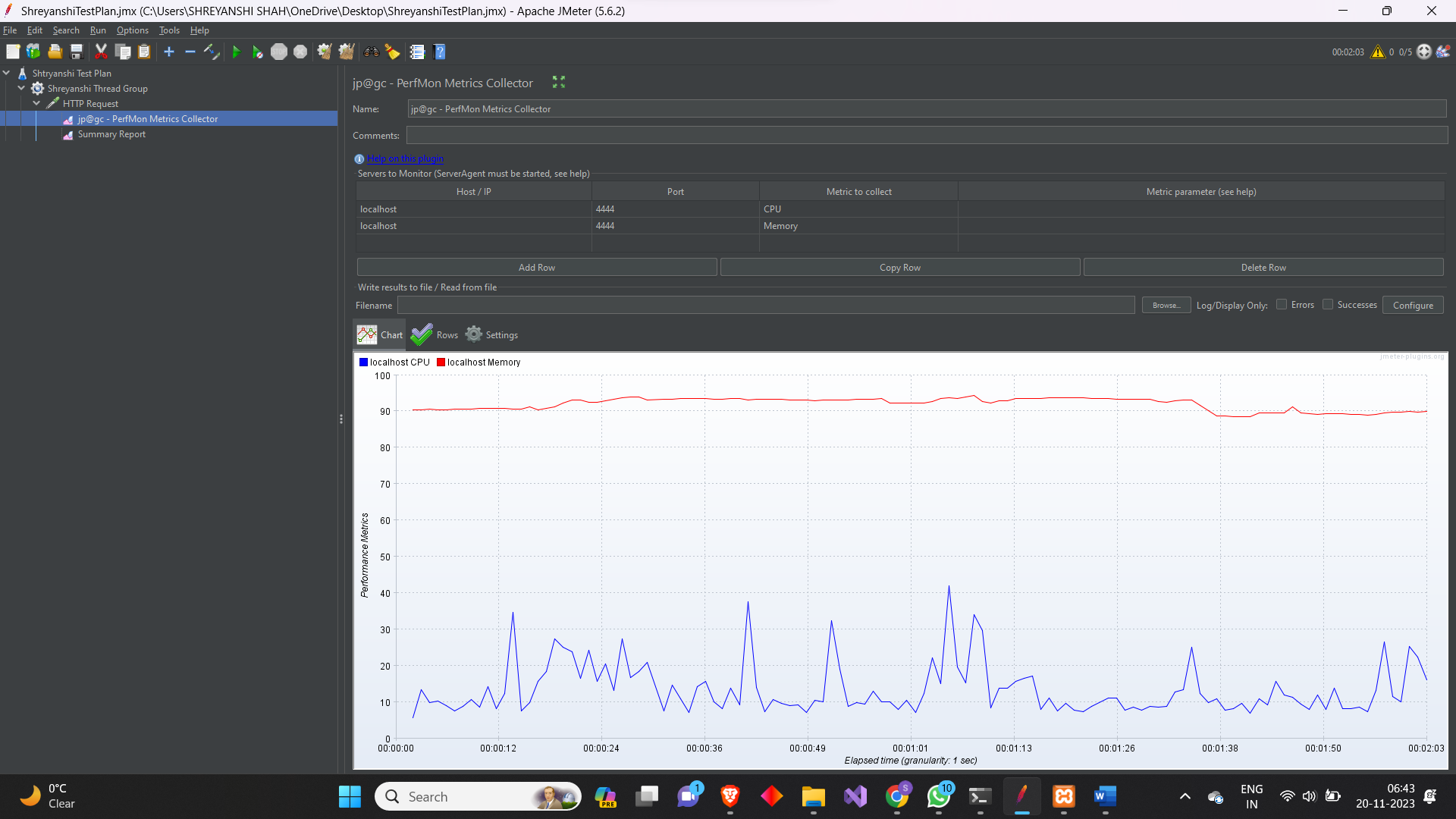
1. *Test Plan 5*

*Number of Users: 5*

*Loop Count: 105000*

*Summary Report:*

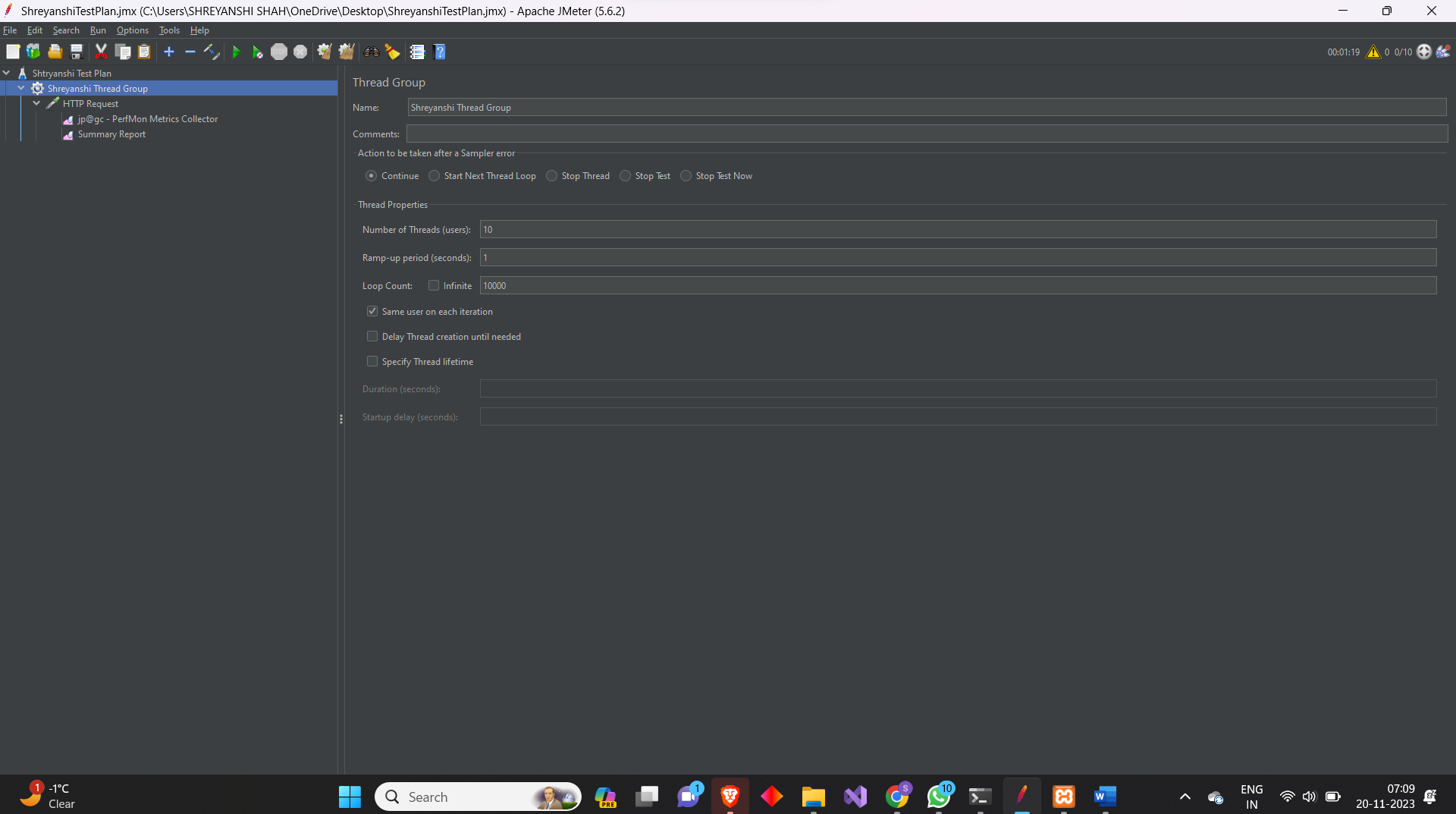
*Plot of memory and CPU usage:*

**

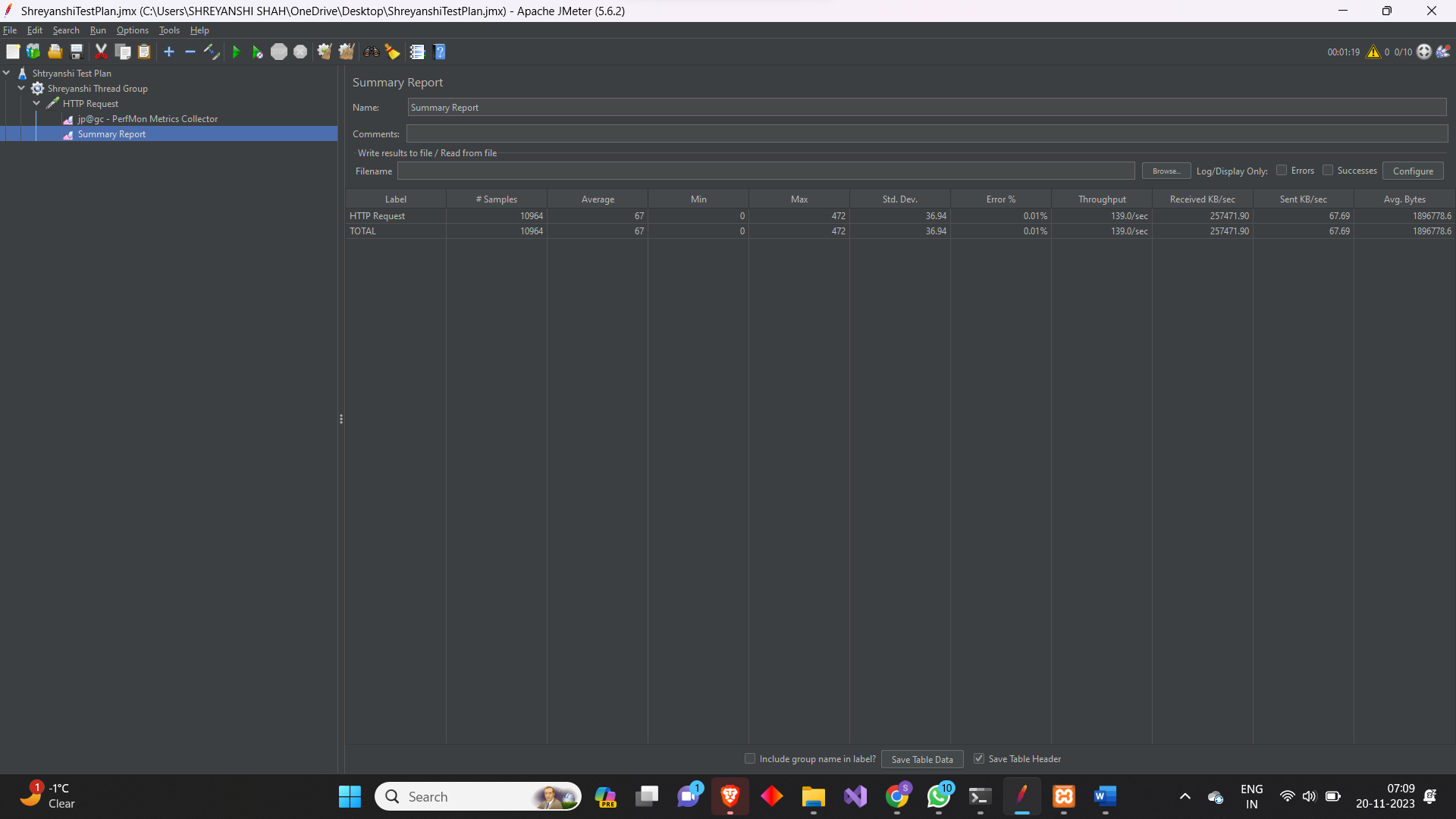
1. *Test Plan 6*

*Number of Users: 10*

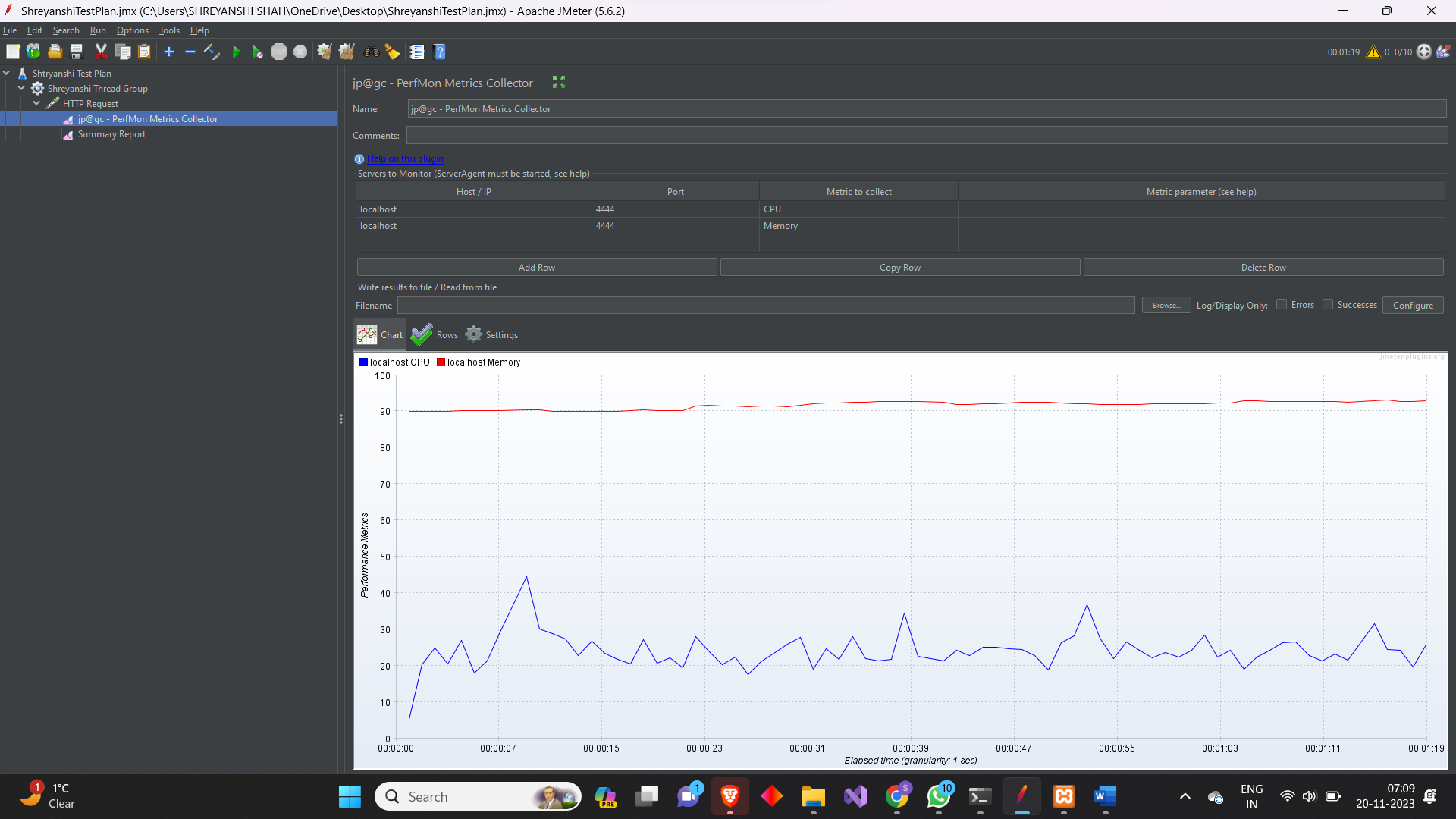
*Loop Count: 10000*

**

*Summary Report:*

**

*Plot of CPU and Memory Usage:*

**

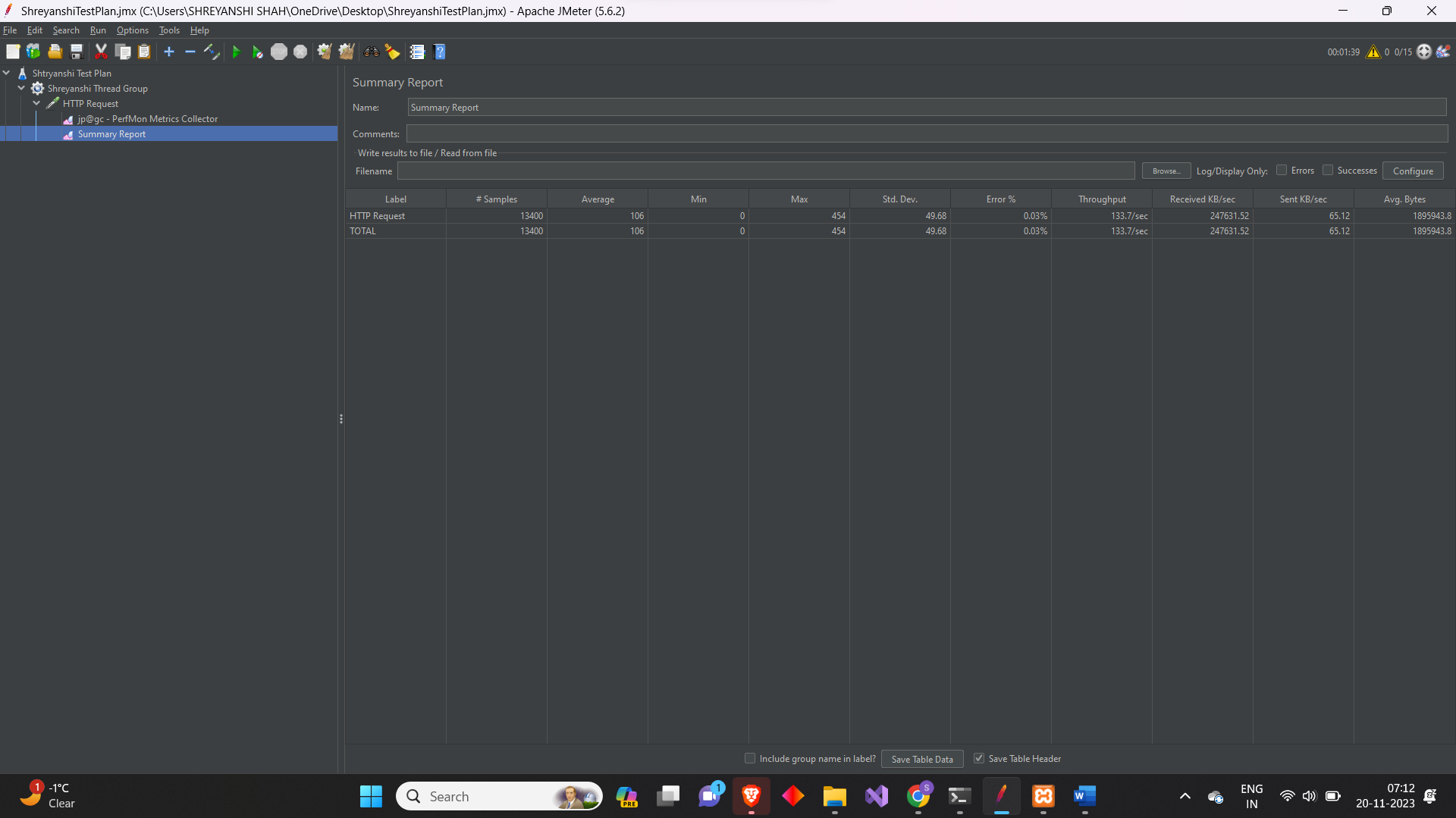
1. *Test Plan 7*

*Number of Users: 15*

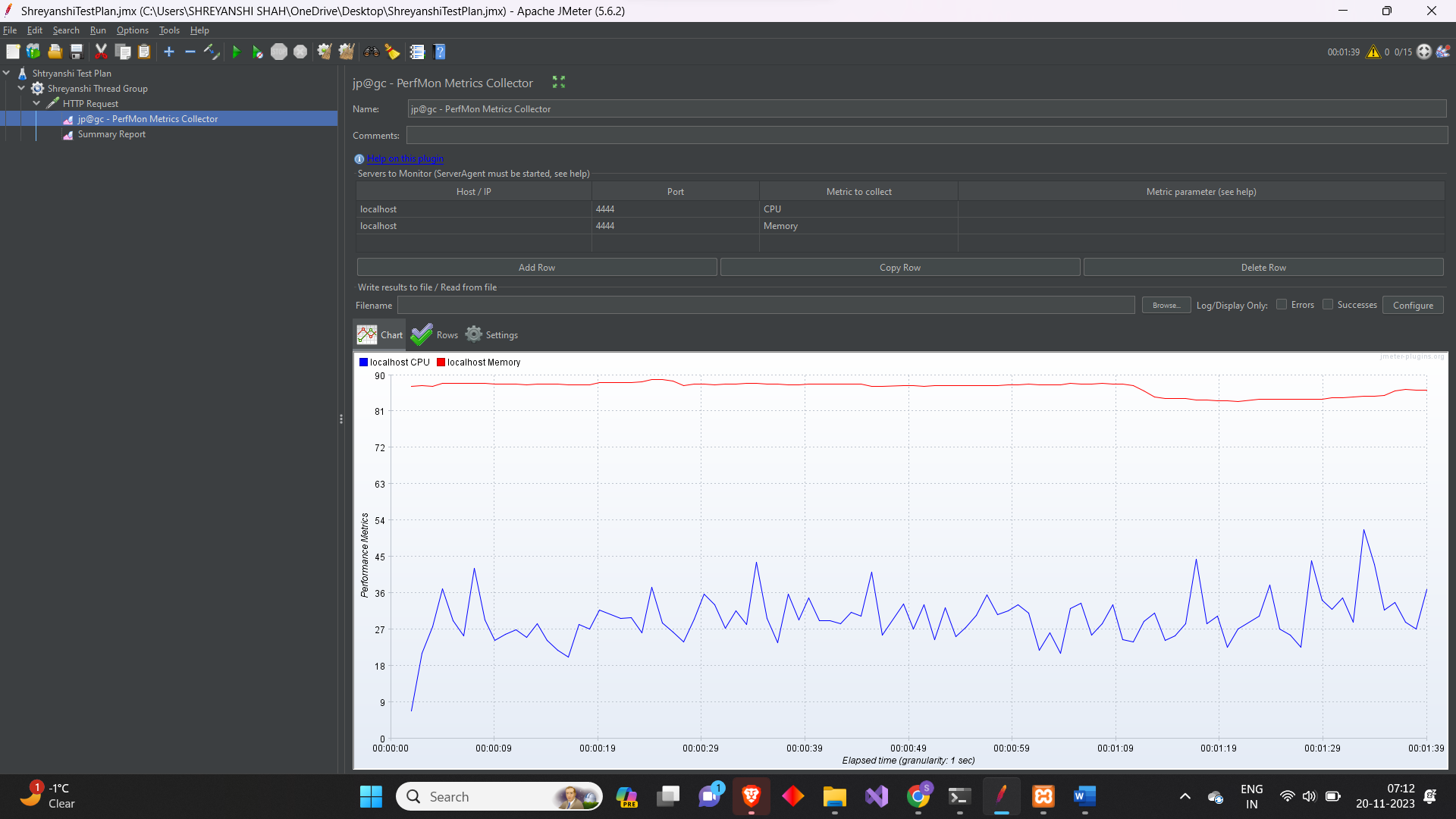
*Loop Count: 10000*

**

*Summary Report:*

**

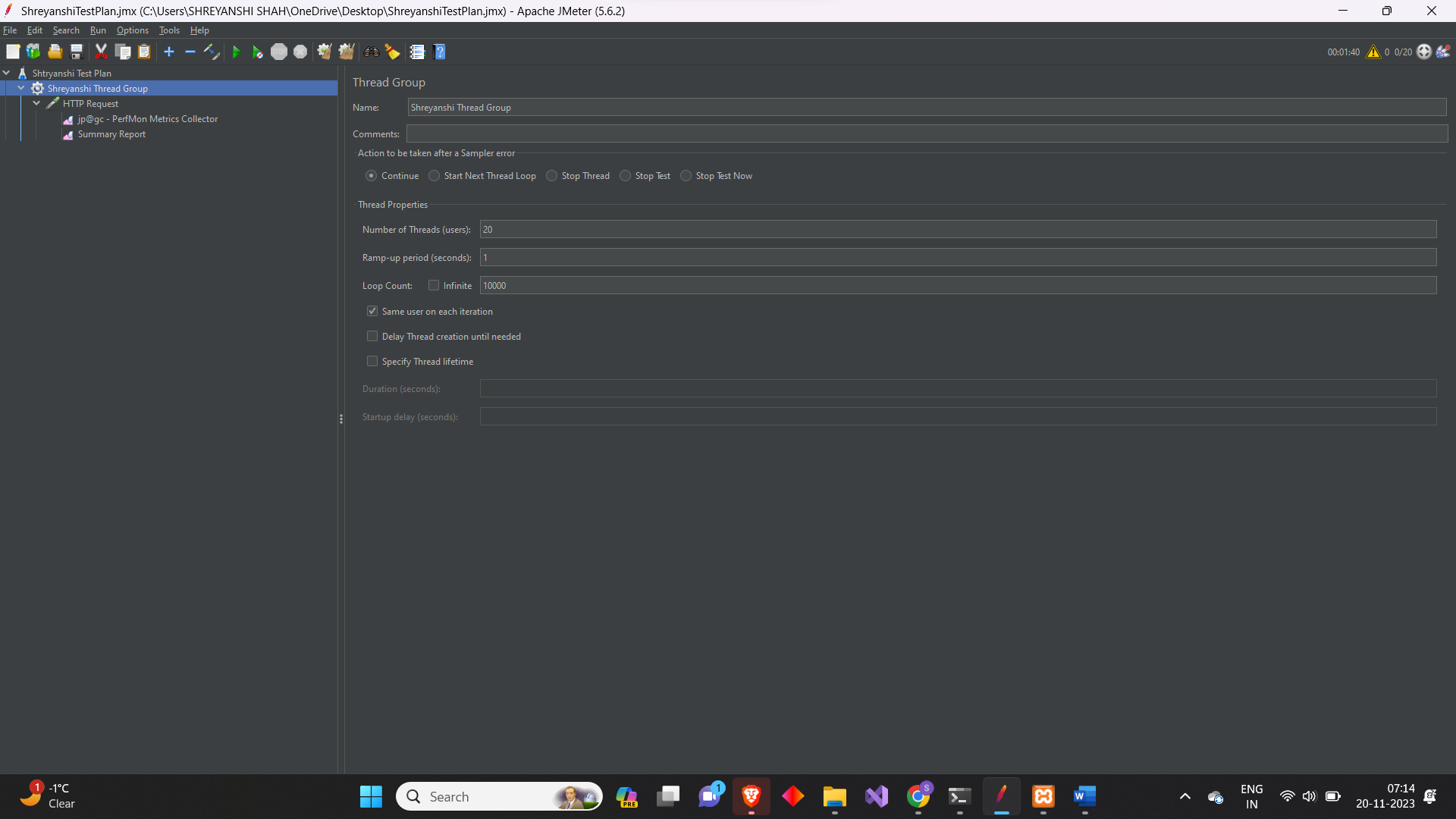
*Plot of CPU and Memory Usage:*

**

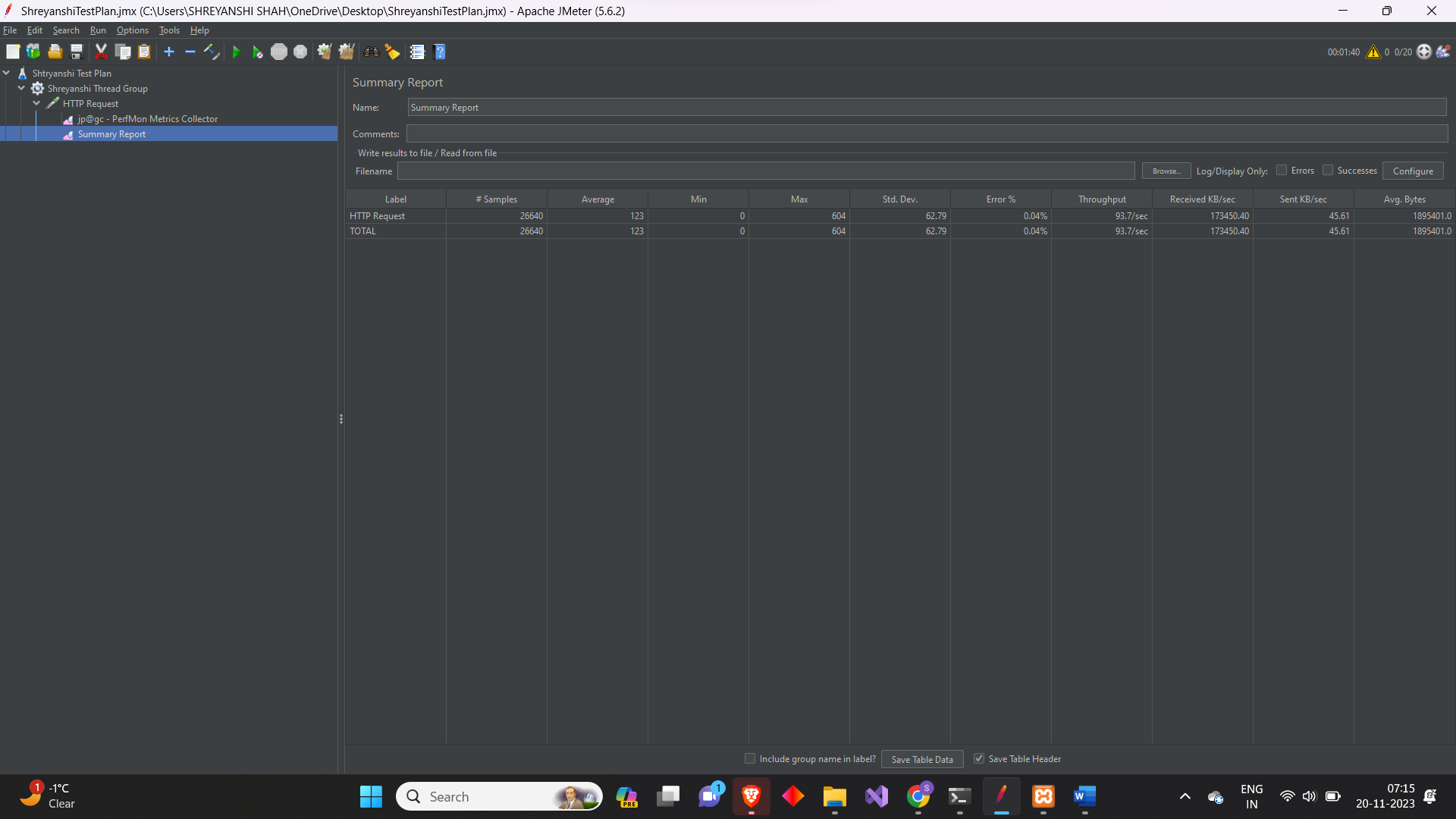
1. *Test plan 8*

*Number of Users: 20*

*Loop Count: 10000*

**

*Summary Report:*

**

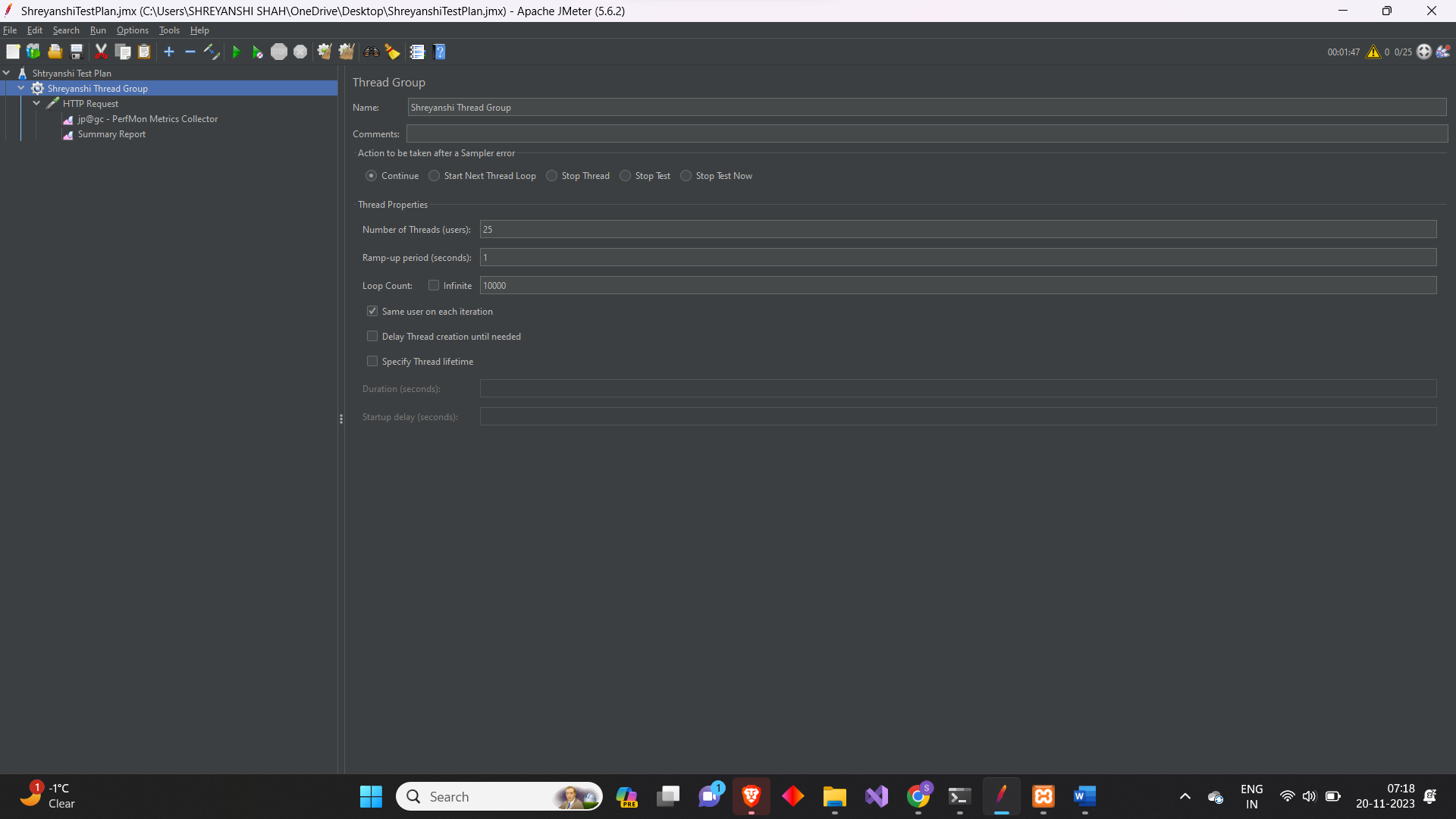
*Plot of CPU and Memory Usage:*

**

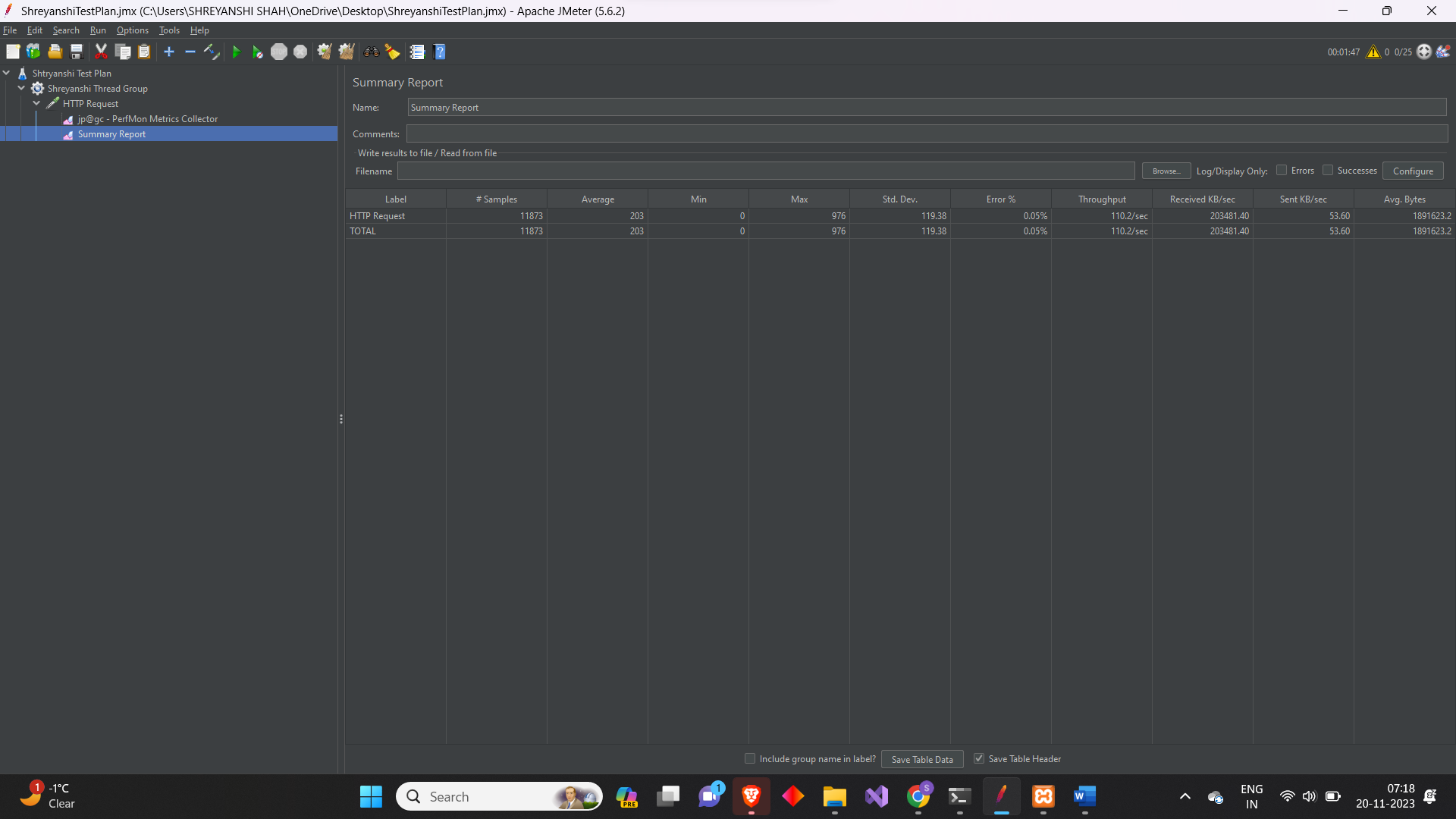
1. *Test Plan 9*

*Number of Users: 25*

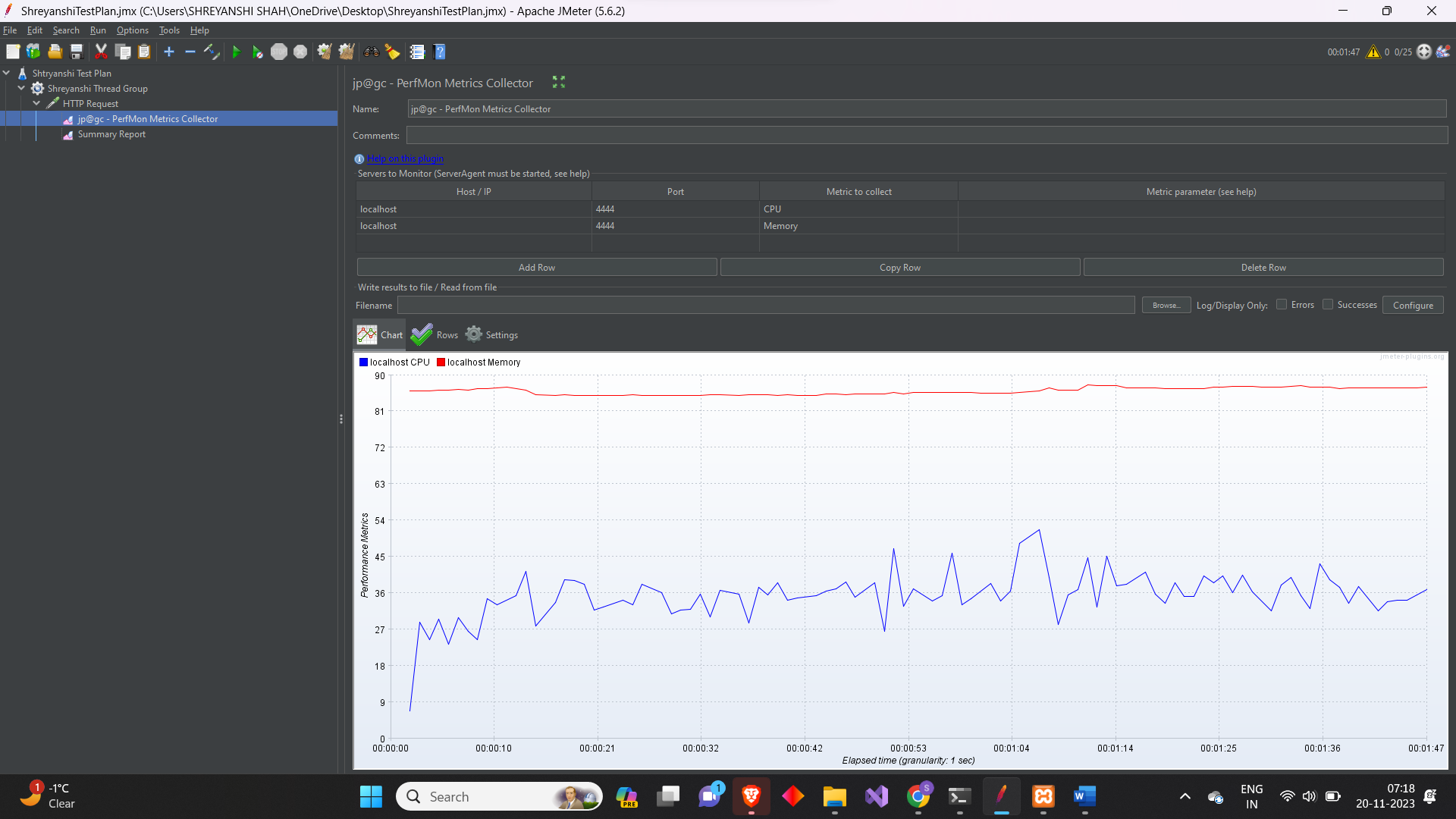
*Loop Count: 10000*

**

*Summary Report:*

**

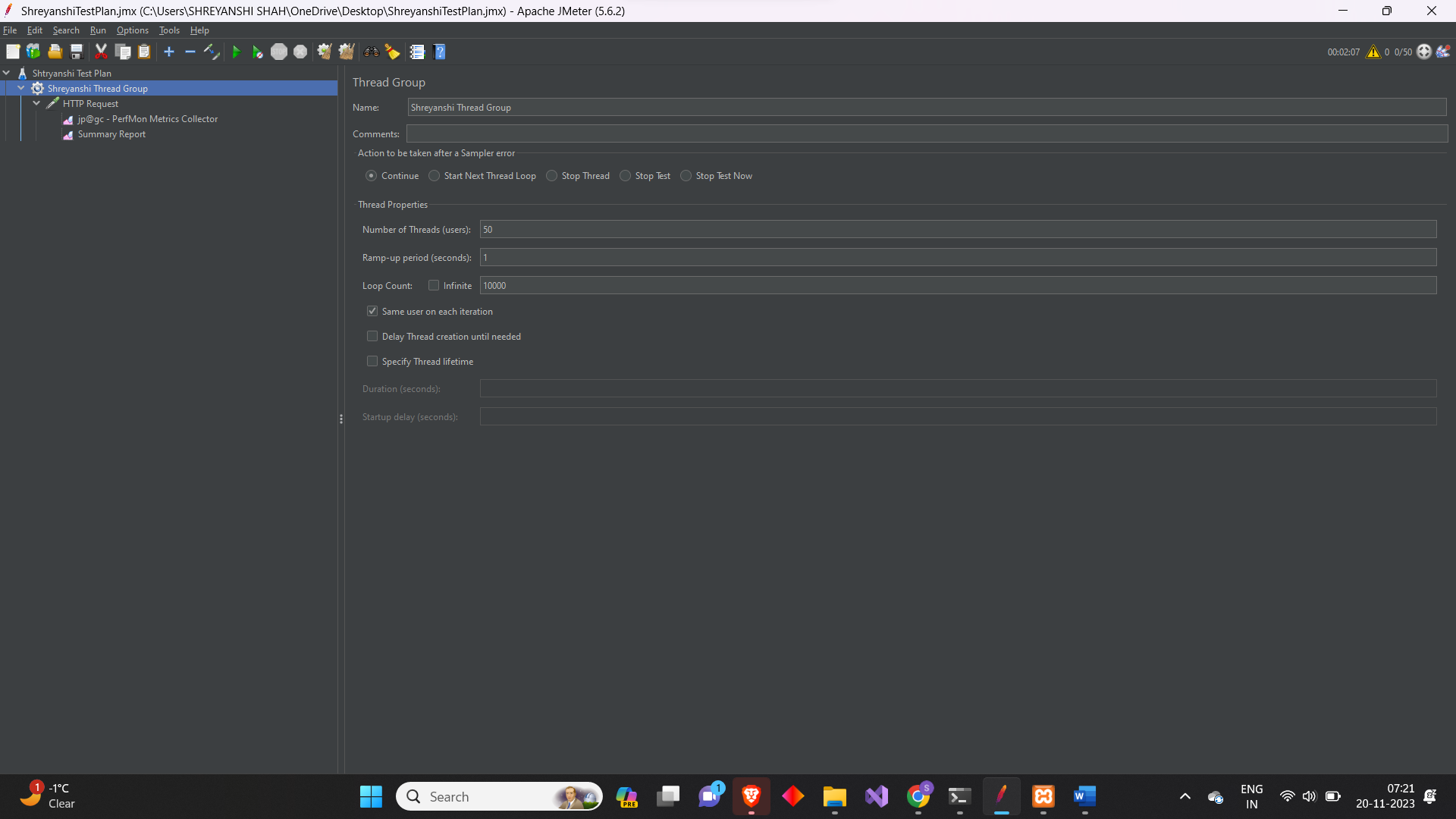
*Plot of CPU and Memory Usage:*

**

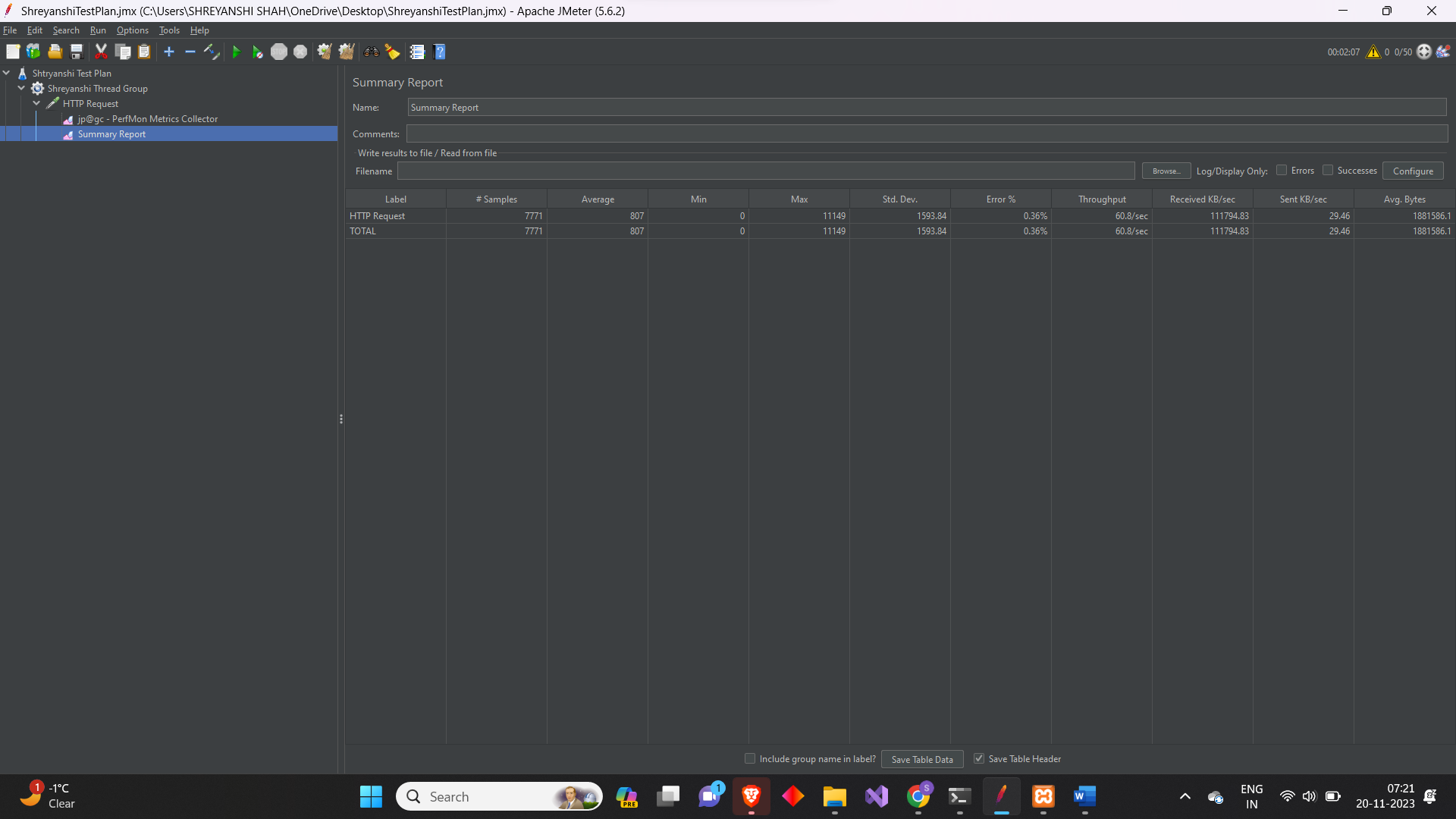
1. *Test Plan 10*

*Number of Users: 50*

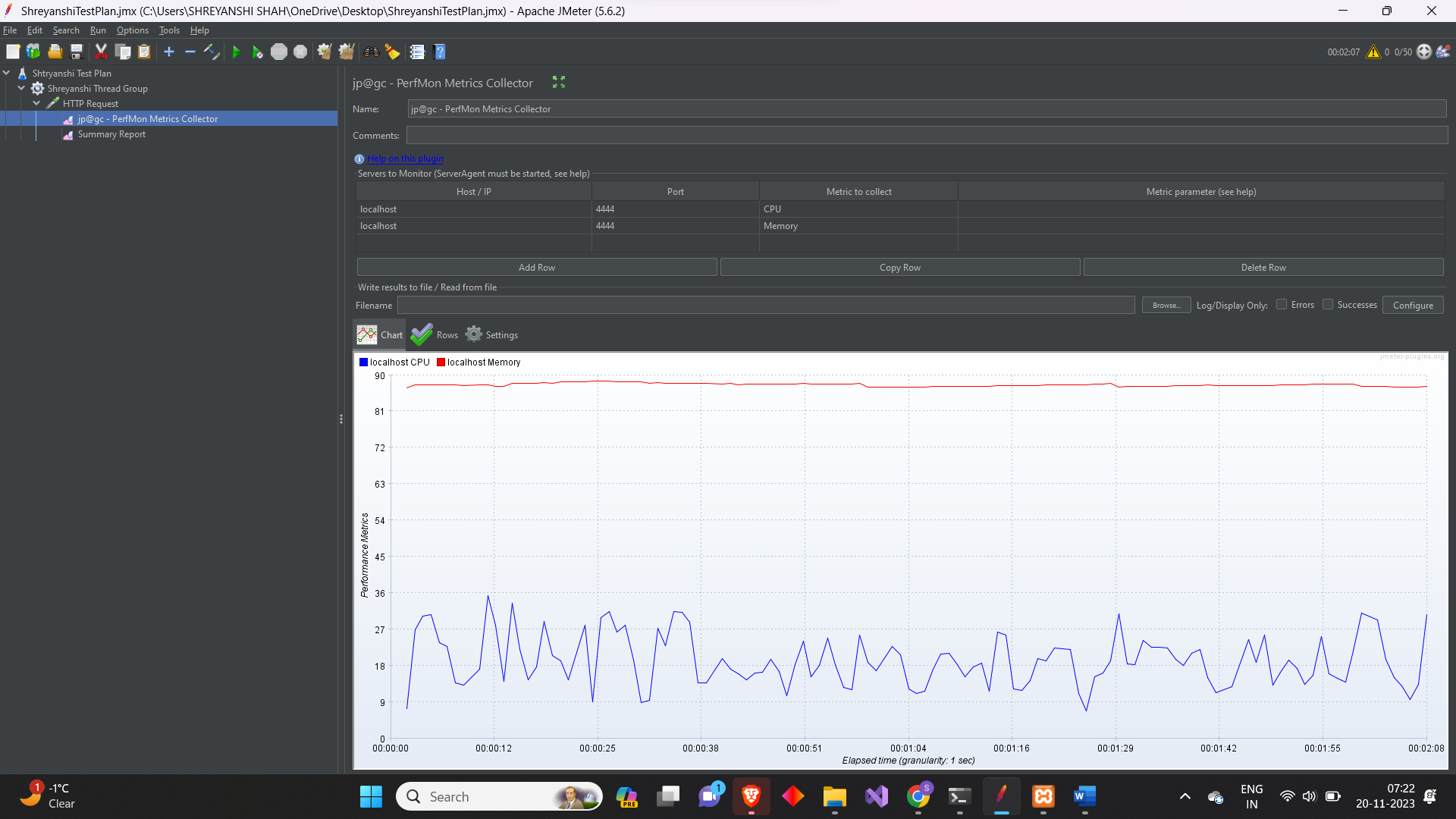
*Loop Count: 10000*

**

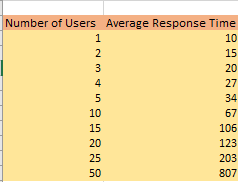
*Summary Report:*

**

*Plot of CPU and Memory Usage:*

**

***2. A MS excel chart of Average Response time (ms) Vs. Number of Users:***

******

*3****. Explanation:***

*Based on the result obtained, it seems that as the number of users increases, the CPU and Memory usage also increases which leads to the increase in response time. This is a common observation and can be indicative of the system’s performance under load.*

*Key points from the result:*

1. *As the number of users goes from 1 to 5, the response time increases gradually from 10ms to 34ms respectively which shows the system can efficiently handle a small number of users.*
2. *When the users doubled from 5 to 10, the response time also doubled to 67ms keeping the loop count at 10000.*
3. *The response time plunged considerably when the user increases from 15 users to 20 users and from 25 users to 50 users.*

*It’s visible that the system may be experiencing performance bottlenecks due to the increase in the number of users.*

*Possible factors contributing to performance bottlenecks:*

1. *Server Capacity: As the number of concurrent users increases, the server may not have enough memory, CPU, or other resources to accommodate them, which would result in slower response times.*
2. *Scalability Issues: When more people use the website at once, performance may suffer due to ineffective scalability design.*
3. *Network latency: When the user base grows, the network may get congested, delaying data transit from the users to the server.*
4. *Inefficient Code: The code of the website may not be performance-optimized, resulting in execution bottlenecks. This could encompass inefficient algorithms, unnecessary computations, or resource-heavy processes that become more noticeable with increased user loads.*

*Validity of the results based on this setup:*

1. *Testing Environment: The findings are derived from a regulated testing environment, and actual conditions may vary. Elements such as network status, user activity, and geographical distribution can influence performance in a live setting.*
2. *Data Distribution: The data does not offer insights into the distribution of user actions. Certain actions or webpages may have a greater impact on response time than others.*
3. *Caching: The effects of caching mechanisms may not be reflected in the results. In a live environment, caching strategies can substantially enhance response times for recurring requests.*
4. *Load Type: The data does not indicate the nature of the load (e.g., concurrent logins, page requests, transactions). Different types of loads can put different components of the system under stress.*
5. *Real-world Variability: The performance of the system may be influenced differently due to the actual user behaviour and patterns, which may deviate from the test scenarios.*
6. *External Factors: Changes in network conditions, modifications to the server infrastructure, or updates to the website can affect performance in a live setting.*

*Factors Affecting Production Performance:*

1. *User Load Variability: The actual usage can fluctuate significantly, and the website may encounter abrupt surges or peaks in user activity. Conducting tests with diverse user loads, including stress tests with loads surpassing the anticipated maximum, can help determine the system’s behaviour under extreme circumstances.*
2. *Infrastructure Scaling: The production environment might have a different infrastructure setup compared to the test environment. Scaling factors, such as load balancing, server clustering, and database sharding, may be applicable in production and can affect the overall system performance*.
3. *Caching and Content Delivery: Production configurations often incorporate caching mechanisms and content delivery networks (CDNs) to streamline the delivery of static and dynamic content. These elements can considerably influence response times and should be factored into performance testing.*

*To sum up, the data indicates potential performance bottlenecks as the user load escalates, necessitating further exploration to pinpoint and rectify specific issues. Enhancing server capacity, refining code efficiently, and deploying effective caching strategies are potential avenues to investigates for improving system performance in a live setting.*