Performance Testing Report – Week 4

1. Objective

The objective of this performance testing activity is to evaluate the behavior of the application under load using **Apache JMeter**. The goal is to measure response time, throughput, and error rate when multiple users access the application simultaneously.

2. Tool Used

- Apache JMeter
- Type of Testing: Load Testing

3. Test Environment

- Operating System: Windows
- Tool: Apache JMeter
- Application Type: Web Application
- Test Type: Performance / Load Testing

4. Test Scenario

A load test was performed to simulate multiple users accessing the application at the same time. The test focused on sending HTTP requests to the application to analyze how it performs under load.

The following JMeter components were used:

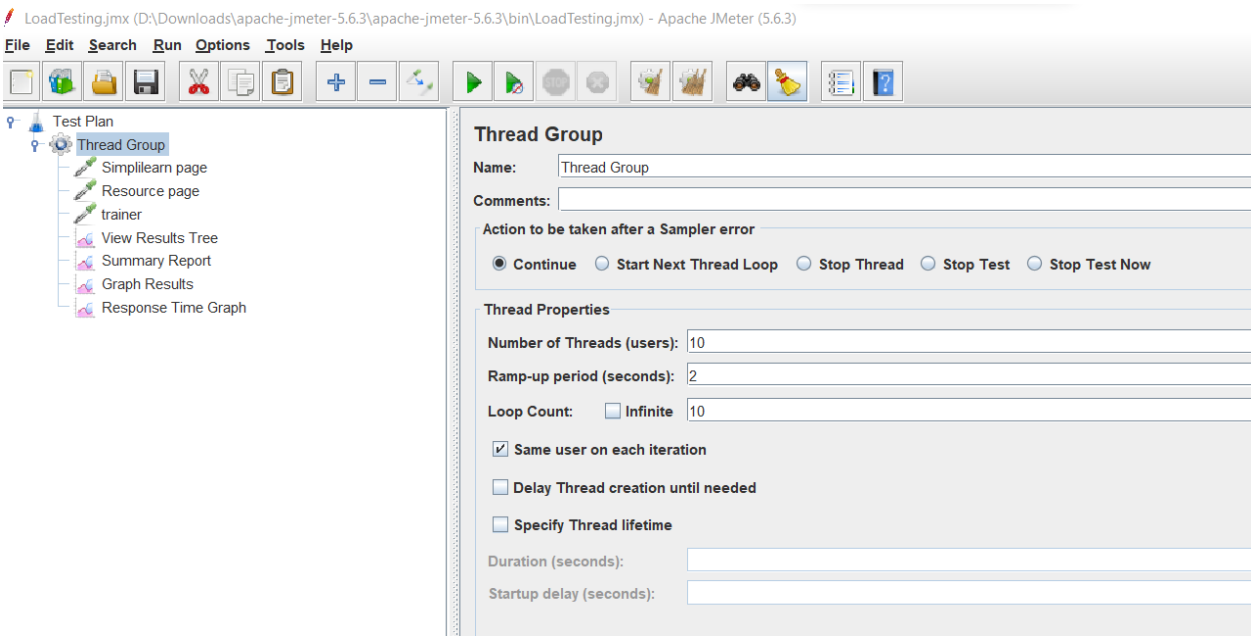
- Test Plan
- Thread Group
- HTTP Request Sampler
- Listeners (View Results Tree, Summary Report, Graph Results, Response Time Graph)

5. Load Test Configuration

Thread Group

- Virtual Users: Configured using Thread Group
- Ramp-up Period: Gradual user increase
- Loop Count: As configured in JMeter

This setup simulates real users accessing the application concurrently.



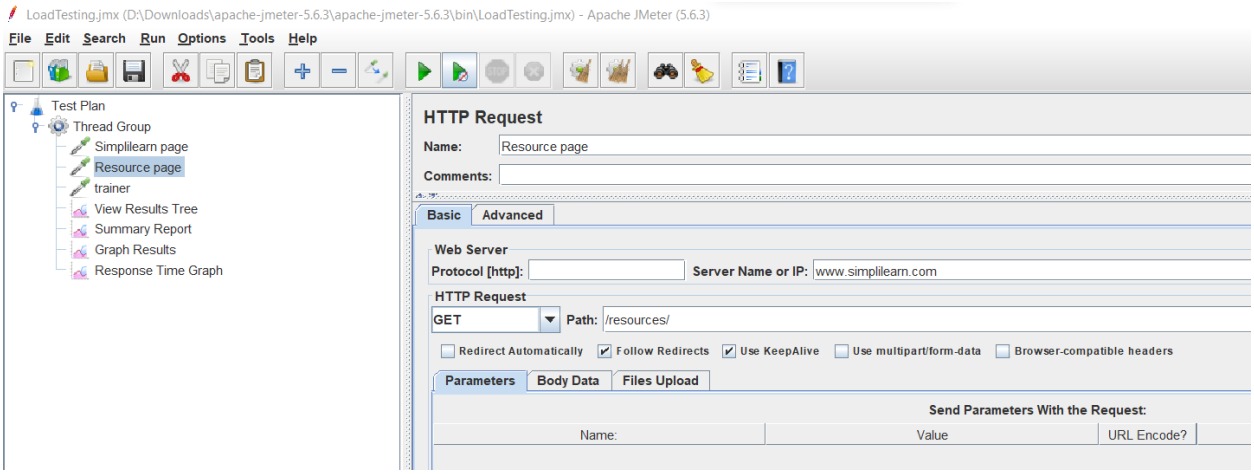
6. Request Details

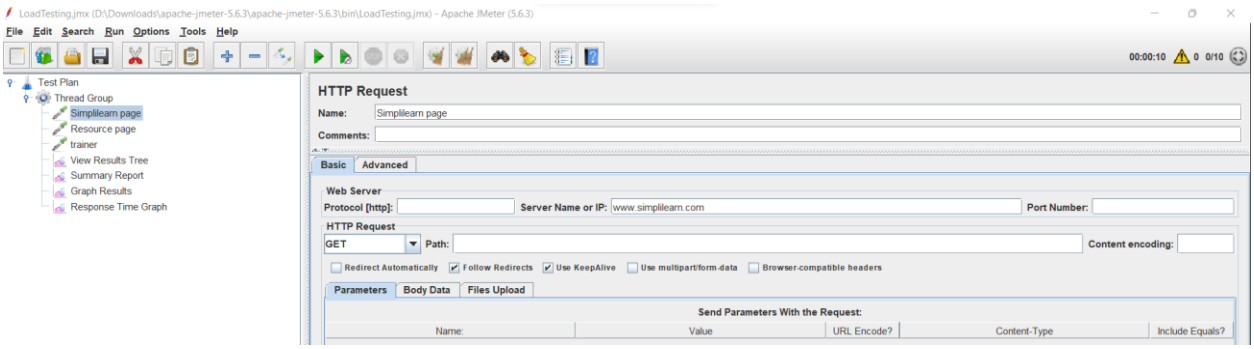
HTTP Request

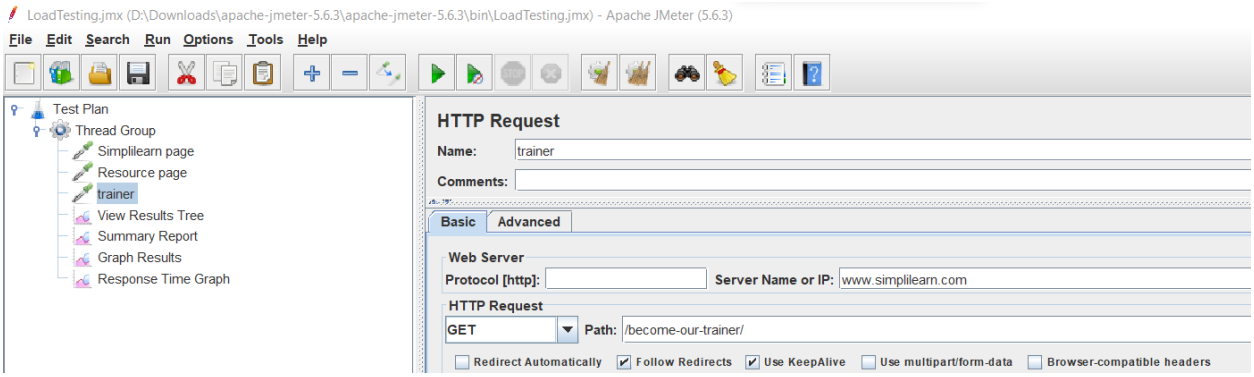
- Protocol: HTTP
- Method: GET / POST
- Server Name and Path configured as per the application

The HTTP Request sampler sends requests to the application and records performance data.

HTTP Request Configuration



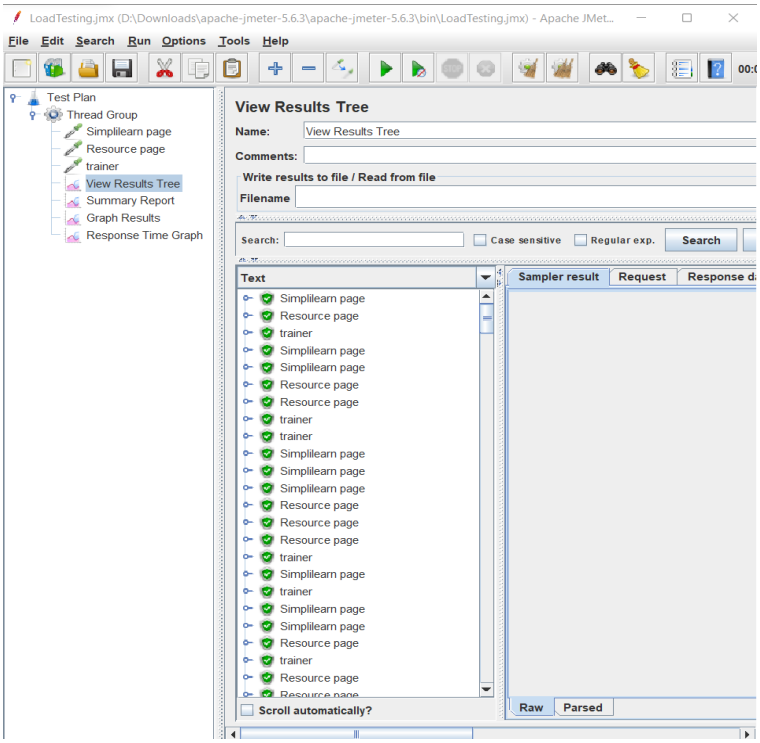




7. Test Execution Evidence

View Results Tree

The View Results Tree listener confirms that requests were successfully sent and responses were received. Most responses were successful, indicating stable application behavior.

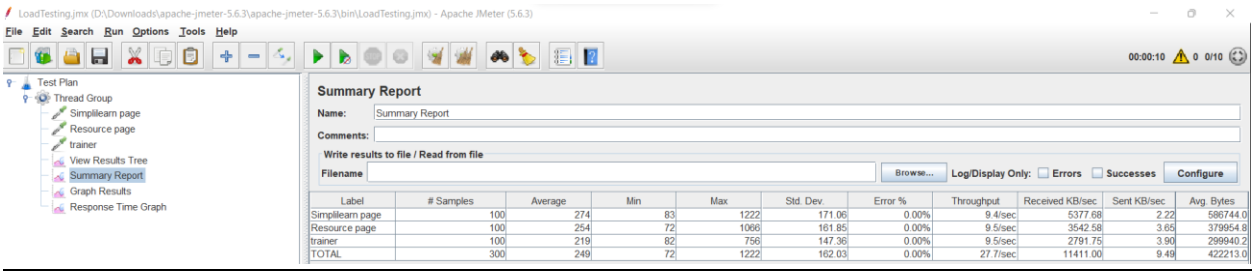


Summary Report

The Summary Report provides key performance metrics such as:

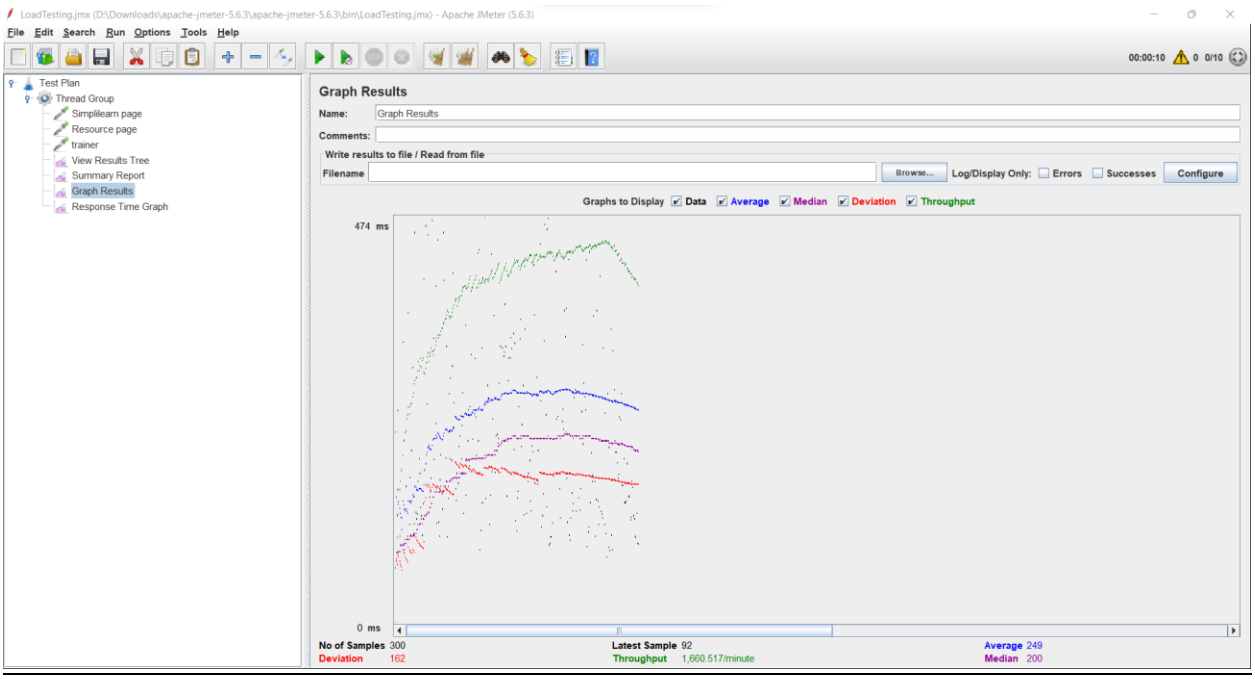
- Average response time
- Minimum and maximum response time
- Throughput
- Error percentage

These metrics help evaluate overall system performance under load.



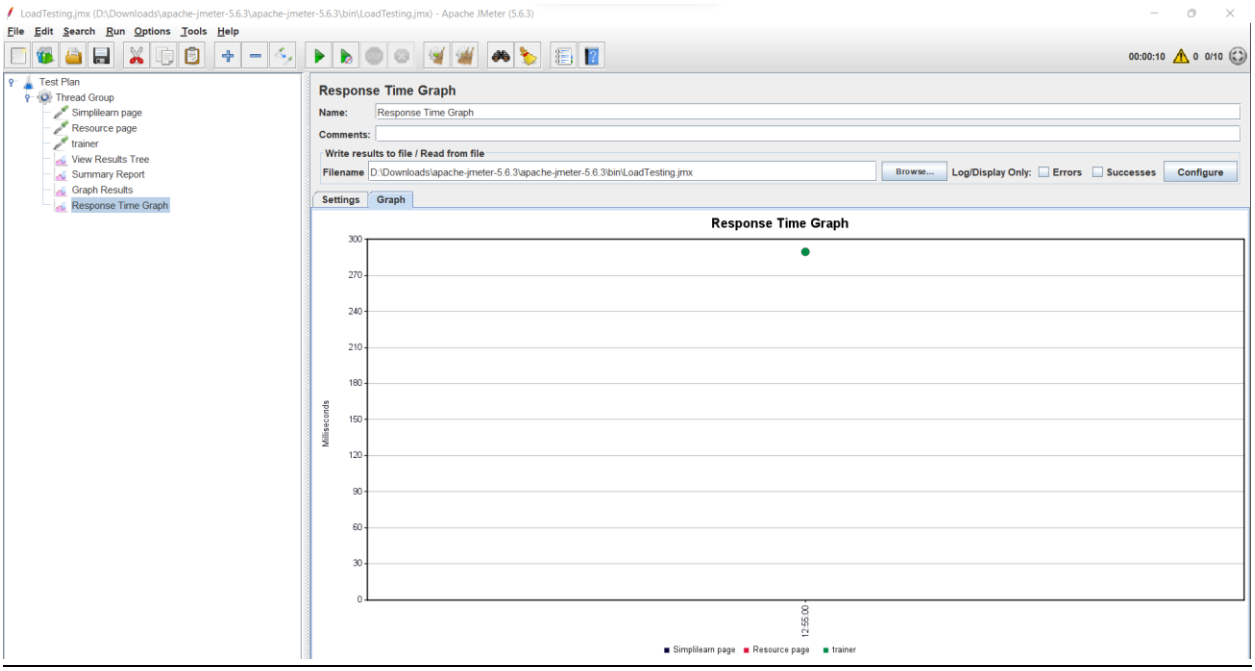
Graph Results

The Graph Results listener visually represents performance metrics, including response time and throughput, making it easier to analyze trends during test execution.



Response Time Graph

The Response Time Graph shows how response time changes with user load. As the number of users increases, response time also increases gradually, which is expected in load testing.



8. Observations

- The application handled multiple user requests successfully
- Response time increased as user load increased
- Throughput remained stable under moderate load
- No major errors were observed during execution

9. Conclusion

The performance test results indicate that the application performs well under moderate user load. The system was able to process multiple requests with acceptable response times and minimal errors.

10. Recommendations

- Optimize server-side performance for higher user load
- Monitor response time under peak traffic
- Conduct stress testing to identify system breaking points