Performance Testing for API (JMeter & Postman)

Overview

The **Performance Testing for API (JMeter & Postman)** project aims to evaluate the scalability, performance, and responsiveness of the **JSONPlaceholder API** by simulating various load conditions using **Apache JMeter** and **Postman**. The primary goal of this project is to measure critical performance metrics, such as response time, throughput, and error rate, and to ensure the API performs optimally under stress and load conditions.

This project utilizes **JMeter** for stress and load testing, generating comprehensive reports and dashboards for analysis, while **Postman**, coupled with **Newman**, is used for functional API testing and performance benchmarking.

Key Features:

- **Stress Testing**: Simulating high traffic conditions using JMeter.
- **Load Testing**: Evaluating API behavior under normal and peak load conditions.
- **Functional Testing**: Ensuring the API works as expected using Postman.
- **Comprehensive Reporting**: Newman and JMeter generate detailed reports and dashboards to track and analyze performance.

Tools and Technologies

- **Apache JMeter**: An open-source performance testing tool used for load and stress testing. JMeter was used to simulate a high number of requests and evaluate the performance of the JSONPlaceholder API under stress.
- **Postman**: A widely used API testing tool for functional testing. Postman was utilized to create API requests, assertions, and export the collection for use with **Newman**.
- **Newman**: A command-line companion for Postman used to run Postman collections and generate detailed performance reports and dashboards.
- **JSONPlaceholder API**: A mock API that serves as the target for the performance tests, providing endpoints for posts, users, comments, and more.

Project Structure

- **JMeter Test Plan**: Contains all the configuration for load and stress testing, including thread groups, samplers, listeners, and assertions.
- **Postman Collection**: Contains the API requests for functional testing of the JSONPlaceholder API.
- **Postman Environment**: Configuration file to specify environment variables for the API requests (e.g., base URL).
- **Newman Reports**: HTML-based summary report generated by running the Postman collection with Newman, providing insights into API performance.

Test Scenarios

The performance tests were conducted for the following scenarios:

1. Get Users - Stress Test

- **Purpose**: Simulate high traffic by sending a large number of requests to the `GET /users` endpoint to evaluate the API's scalability under heavy load.
- **Metrics Collected**: Response time, throughput, error rate, and server performance under stress.

2. Get Posts - Basic Load Test

- **Purpose**: Simulate typical load on the `GET/posts` endpoint to observe how the API performs under regular usage conditions.
- **Metrics Collected**: Response time, throughput, and error rate.

Performance Metrics

The following key performance metrics were measured:

- **Response Time**: Time taken for the API to respond to each request. Critical for understanding user experience and ensuring fast response times under load.
- **Throughput**: Number of requests processed per second, indicating the capacity of the server to handle requests.
- **Error Rate**: Percentage of failed requests in the test, indicating the reliability of the API.
- **Request Count**: Total number of requests executed during the testing phase, providing an overview of the load imposed on the API.

Key Metrics for Test Scenarios

Get Users - Stress Test

- **Sample Count**: 500 requests
- **Error Count**: 0
- **Error Percentage**: 0%
- **Average Response Time**: 416.95 ms
- **Median Response Time**: 291.5 ms
- **Maximum Response Time**: 1892 ms
- **Throughput**: 29.22 requests/sec
- **Data Received per Second**: 195.92 KB/sec
- **Data Sent per Second**: 3.94 KB/sec

Get Posts - Basic Load Test

- **Sample Count**: 500 requests
- **Error Count**: 0
- **Error Percentage**: 0%
- **Average Response Time**: 448.83 ms
- **Median Response Time**: 337 ms
- **Maximum Response Time**: 2368 ms
- **Throughput**: 35.84 requests/sec
- **Data Received per Second**: 1006.29 KB/sec
- **Data Sent per Second**: 4.83 KB/sec

Total Test Summary

- **Total Requests**: 1000
- **Error Count**: 0
- **Error Percentage**: 0%
- **Average Response Time**: 432.89 ms
- **Median Response Time**: 319.5 ms
- **Maximum Response Time**: 2368 ms
- **Throughput**: 58.43 requests/sec
- **Data Received per Second**: 1016.17 KB/sec
- **Data Sent per Second**: 7.87 KB/sec

Test Execution

Postman Test Execution with Newman

1. **Setup**:
    - Import the Postman collection (`performance-test.json`) and environment file (`performance-env.json`) into Postman.
    - Configure your local or cloud-based Postman instance to target the **JSONPlaceholder API**.

Install **Newman** using npm if not already installed:

```
npm install -g newman
```

    -
2. **Run the Tests**:

To execute the tests and generate the report, run the following command:

```
newman run performance-test.json -e performance-env.json --reporters
html,cli --reporter-html-export newman-report.html
```

    -
    - This will generate a detailed HTML report (`newman-report.html`) with performance metrics and test results.

JMeter Test Execution

1. **Setup**:
    - Download and install **Apache JMeter**.
    - Open the **JMeter Test Plan** (`performance-test.jmx`).
    - Modify the configurations for thread groups, samplers, and listeners as needed.
2. **Run the Tests**:
    - Execute the test plan within JMeter and generate the **JMeter Dashboard Report** to visualize key performance metrics such as response time, throughput, and

error rates.

3. **View the Report**:
    - After running the tests, the results will be available in an HTML format for detailed analysis.

Reporting

JMeter Report

The JMeter dashboard provides a comprehensive analysis of the test results, including:

- **Request Summary**: The total number of requests, response times, and error percentages.
- **APDEX Score**: A measurement of user satisfaction based on response times.
- **Response Times Over Time**: Visual graphs indicating how response times vary over the duration of the test.
- **Error Analysis**: Identification of any errors that occurred during the test.

Newman Report

The **Newman** summary report contains:

- **Total Requests**: 60 requests tested in total.
- **Assertions**: 180 assertions evaluated, all passing with no errors.
- **Test Pass Percentage**: 100% pass rate across all tests.
- **Average Response Time**: 695 ms.
- **Failed Tests**: 0 (No failures observed).

Conclusion

This project demonstrates the effectiveness of **JMeter** and **Postman** as tools for conducting both functional and performance testing of APIs. The JSONPlaceholder API performed well under load, with no failures and response times within acceptable thresholds for both load and stress tests. The combination of **JMeter** for load testing and **Postman/Newman** for functional and performance testing provides a robust solution for comprehensive API evaluation.