Volume 8, No. 3, March – April 2017



International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Comparative Analysis of Web Applications using JMeter

Nisha Jha M.Tech Scholar Department of Computer Engineering, YMCAUST, Faridabad,India Rashmi Popli Assistant Professor Department of Computer Engineering, YMCAUST, Faridabad,India

Abstract: The Software testing is done to ensure the quality characteristics of an application like functionality, performance, usability, scalability. Performance testing is a technique for quality assurance. It is a way of testing web applications to guarantee that the applications will perform properly under their expected normal workload. Features and Functionalities supported by a software system is not the sole concern. A software application's performance like its response time, do make a difference. The objective of performance testing is not to discover bugs but to eliminate performance bottlenecks. Performance testing is mainly categorised as Stress and Load testing. JMeter is an Open Source tool, widely used to measure performance parameters of an application. It is a java application for load testing and measuring performance. This paper presents performance testing concepts and a comparative analysis of web applications in terms of its response time, throughput and deviation, proposing a way to increase revenue for web application Industry.

Keywords: software testing; web testing; performance testing; Jmeter; load testing; response time

I. INTRODUCTION

Performance testing is a testing that is performed to determine how fast some aspects of the system perform under a particular workload. In other words, to check functionality under load is called performance testing.

Performance of a system is generally measured in terms of response time for the user activity. For example, when a customer wishes to withdraw money from ATM machine, customer inserts the valid details with card and waits for the disbursal of money. If the system takes more time to deliver the cash, the customer may not appreciate the system.

Performance Testing is designed to test the overall performance of the system at high load and stress conditions. Performance Testing occurs throughout the software development life cycle to isolate performance related constraints. Even at unit level, the performance of the individual module may be assessed as the internal logic of software that is being tested.

When the system gets functionally stable i.e. no more changes are done or required, then performance testing is started.

II. TYPES OF PERFORMANCE TESTING

A. Load Testing

It is done to check whether the server is able to meet its specification or not. It is analogous to volume testing and determines how the application deals with large tasks [10].

B. Stress Testing

It examines the application behavior under peak bursts of activity. It is conducted to evaluate the system or component at or beyond its limit of its specified requirements [2]. It is done to find the end point where the server starts behaving abnormally or starts crashing, this point is called stress point or knee point or crash point. For example, if there a sale or offer on a website on a particular day, then how many users it can handle in a consistent way. This is ensured by stress testing [9].

C. Spike Testing

Spike testing is done to test the fluctuations over a period of time and then find out their reasons. This type of performance testing is done to verify the stability of a system during the bursts of multiple users at the same time or the varying loads over the varying time periods.

D. Capacity Testing

It measures the overall capacity of the system and determines at what point response time and throughput become unacceptable. Capacity testing is conducted with normal load to determine the extra capacity where stress capacity is determined by overloading the system until it fails, which is also called a stress load to determine the maximum capacity of a system [7]. It is done to ensure scalability and stability for future purposes.

III. PERFORMANCE TESTING GOALS

Important performance goals are as follows:

A. Stability

A stability test is designed to determine whether a web application will remain serviceable over an extended time span. It is achieved through soak testing. Server should work consistently. For example, for a constant load of 1000 users per second for 15 days with same environment and configurations, the server's response time should not fluctuate since the load has not been changed.

B. Response time

Response Time is the time gap between the users sending a request and receiving their response [4]. It represents how long a user is waiting for its request to be processed by the application. Two main factors create slow download times: Large web page sizes and slow server performance. It is all about defining the measurements. Performance testing is done to improve performance but in order to know this, the current baseline should be known. It means how fast the application should be to satisfy the specifications.

C. Scalability

Transaction Time is the key determinant of Scalability. Transaction Time doesn't mean good scalability –in fact it's the antithesis of it. Scalability is determined by the number of simultaneous clients served over a given period of time with a limited delay time.

IV. PERFORMANCE TESTING TOOLS

Some popular commercial tools are as follows:

A. Load Runner (HP)

Hewlett-Packard's load/stress testing tool for web and other applications; supports a wide variety of application environments, platforms, and databases. Large suite of network/app/server monitors to enable performance measurement of each tier/server/component and tracing to bottlenecks. Integrates with other Mercury testing and monitoring products [12].

B. Jmeter (Apache)

Apache's Jmeter for load/stress testing of web, database, web Services etc. integration with other tools and plugins provides an in-depth view by monitoring its operating system, database and network components, as well as the application itself.

C. Rational Performance Tester (IBM)

Rational Performance Tester is a performance testing tool by IBM used to identify the presence and cause of system performance bottlenecks. Provides testers with automated performance testing capabilities to validate the scalability of web and server based applications. Automates test data variation and enables insertion of custom Java code for flexible test customization [11].

D. Silk Performer(Borland)

Enterprise class load testing tool from Borland can simulate thousands of users working with multiple protocols and computing environments. Allows prediction of behavior of ebusiness environment before it is deployed, regardless of size and complexity. Silk Performer Lite version also available for up to 100 simulated users.

E. WebLoad(Radview)

Load testing tool from Radview software, also available as part of their test view web testing suite. Capabilities include over 75 performance metrics; can view global or detailed account of transaction success or failures on individual virtual client level, assisting in capturing intermittent errors; allows comparing or running tests vs. past test metrics. Test scripting via visual tool or JavaScript. Wizard for automating non-GUI based services.

V. PERFORMANCE TESTING ACTIVITIES

Performance testing is a testing that is performed to determine how fast some aspects of the system perform under a particular workload. In other words, to check functionality under load is called performance testing.

The seven core performance testing activities [1]:

- 1. Identify Test Environment
- 2. Identify Performance Acceptance Criteria

- 3. Plan and Design Tests
- 4. Configure Test Environment
- 5. Implement Test Design
- 6. Execute Tests
- 7. Analyze, Report, and Retest

These seven core activities do not in themselves constitute an approach to performance testing; rather, they represent the foundation upon which an approach can be built that is appropriate for the project.

It generally makes sense to start identifying, or at least estimating, the desired performance characteristics of the application early in the development life cycle. This can be accomplished most simply by noting the performance characteristics that your users and stakeholders equate with good performance. The notes can be quantified at a later time. Classes of characteristics that frequently correlate to a user's or stakeholder's satisfaction typically include:

- 1. Response time: The time required by a server to process one transaction. For example, the product catalog must be displayed in less than three seconds.
- 2. Latency: The time required to process a request, as perceived by the user. Latency includes network delay as well as response time.
- 3. Throughput: The number of transactions processed per unit time. For example, the system must support 25 book orders per second.
- 4. Workload: The number of transactions that need to be processed per unit time. For example, a server receives 500 transactions per minute, but is able to process only 80% of that load. Workload=500 transactions per minute Throughput=400 transactions per minute Ideally throughput=workload When workload exceeds capacity, throughput < workload and transactions are delayed or lost.</p>
- 5. Resource Utilization: For example, processor utilization is not more than 75 percent. Other important resources that need to be considered for setting objectives are memory, disk input/output (I/O), and network I/O.
- 6. Standard Deviation: By definition, one standard deviation is the amount of variance within a data set.

VI. AN INTRODUCTION TO JMETER

Apache Jmeter is a 100% pure java desktop application designed to load test client/server software (such as a web application). It may be used to test performance both on static and dynamic resources such as static files, Java Servlets, CGI scripts, Java objects, database, FTP servers, and more. Jmeter can be used to stimulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types.

Additionally, Jmeter can help to regression test application by letting to create text scripts with assertions to validate that the application is returning the results that are expected. For maximum flexibility, Jmeter allows to create these assertions using regular expressions.

JMeter's capabilities rapidly expand as developer stake advantage of its pluggable architecture. The primary goal of further developments will be:

- 1. Addition of web sockets
- 2. Addition of FTPS and SFTP protocols

- 3. Enhancements to web services protocols(SOAP architecture)
- 4. Enhancements to JMS protocol implementation

Figure 1 shows the overall architecture of Apache JMeter.



Figure 1. JMeter Architecture.

Companies need to go all out to ensure that their online assets run optimally. Web administrators run a whole gamut of performance and functional tests to ensure that the websites perform optimally. While there are extensive testing tools for such tasks, many of them are too complex. Apache Jmeter open source software fills a big void in this space by making such tests easy.

VII. CASE STUDY: SIMULATION RESULTS AND DISCUSSION

The very high cost of business tools for performance testing has triggered the development of open source testing tools for testing online applications. These tools are fully featured, utterly reliable and relevant to most of the load testing applications. The tools simulates an exceptional many clients connecting with Web, application and database servers; make practical models simulating diverse enterprise situations; and deliver a vault of information to investigate and report consequences during development of software program. Open source tools are manifestly on hand at no fee which introduces a big cost savings without obtaining testing tools commercially.

For experimental setup, two websites named labqashop (http://labqashop.qacampus.com) [5] and nopcommerce (http://demo.nopcommerce.com) [6] have been considered and for simulation, online Performance testing tool JMeter is used. We have analyzed the websites by considering three performance Parameters such as throughput, deviation and average response time. During testing, the following test results as a graph using JMeter Performance testing tool are observed. By seeing this, we can come to know that labqashop website is designed with high throughput, less request error rate and high average response time when compared with other websites.

No of virtual users: 100 Ramp up Period: 100 Loop Count: 10

The number of threads represent the virtual users. Loop count represents the number of times to execute the test.

Ramp up period tells JMeter how long to take to "ramp-up" to the full number of threads chosen. If 100 threads are used and the ramp up period is 100 seconds, then JMeter will take 100 seconds to get all 100 threads up and running [8]. Each thread will start 1(100/100) second after the previous thread was begun. If there are 30 threads and a ramp up period of 120

seconds, then each successive thread will be delayed by 4 seconds. Ramp up period should be long enough to avoid too large a workload at the start of a test and short enough that the last threads start running before the first ones finish.

The following Figure 2 shows the Graph Results with throughput (574) and deviation (122) for the online shopping website labqashop.



Figure 2. labqashop's graph results.

The following Figure 3 shows the Graph Results with throughput (562) and deviation (385) for the online shopping website nopcommerce.



Figure 3. nopcommerce's graph results.

The following Figure 4 shows the labqashop's Response Time graph with the above mentioned parameters.



Figure 4. labqashop's Average Response Time Graph

The following Figure 5 shows the nopcommerce's Response Time graph with the above mentioned parameters.



Figure 5. nopcommerce's Average Response Time Graph

This study has highlighted some of the main aspects of online shopping websites performance and it discussed how web application performance testing is going on and how it has been evaluated. It discussed and illustrated the overall achievement in the area of performance testing of websites. Cloud-based performance testing can provide enormous benefit to business by reducing development cost, effort, and time [3]. As a result, organizations can progress focus their resources and energy in foundation business areas.

With this evaluation results, it is observed that the labqashop would get more customer satisfaction than other website because of higher throughput, lower deviation and quick average response time.

VIII.CONCLUSION

The purpose of this work is to discuss modern web application performance testing from a theoretical and a practical perspective, with an emphasis at the latter.

The study has focused on some of the principle aspects of performance measures of online shopping websites and it also discussed how web applications are tested for performance and the way it is evaluated. It discussed and illustrated the overall achievement within the vicinity of performance testing of websites throughout the peak load time. As a result, businesses can development focus their assets and strength in basis commercial enterprise regions. This paper has highlighted some of the essential issues, tools, challenges which can be encountered in web performance testing. It has additionally supplied information about various performance testing tools and shopping website's performance report based on average response time graphs, throughput and deviation. With these evaluations and graphical reports it is been observed that on an offer day how the websites are loaded with customer's requests and gain the customer's pride and satisfaction.

IX. REFERENCES

- Ms. S. Sharmila, Dr. E. Ramadevi," Analysis of Performance Testing on Web Applications", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 3, March 2014.
- [2] Shakti Kundu. "Web Testing: Tool, Challenges and Methods", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012.
- [3] V. Janani, K. Krishnamoorthy, "Evaluation of Cloud based Performance Testing for Online Shopping Websites", Indian Journal of Science and Technology, Vol 8(35), December 2015.
- [4] Wikipedia. Software performance testing Wikipedia, https://en.wikipedia.org/wiki/Software_perfor mance_ testing, 2017. [Online; accessed 11-October-2016].
- [5] Qashop. http://labqashop.qacampus.com/, 2017. [Online; accessed 08-December-2016].
- [6] nopcommerce. http://demo.nopcommerce.com, 2017. [Online; accessed 08-December-2016].
- [7] Performance testing tools available at http://www.softwaretestinghelp.com/what-isperformance-testing-load-testing-stress-testing/
- [8] Performance Testing: Types, Process & Important Metrics available at http://www.guru99.com/performancetesting.html
- [9] R.S.Pressman, "Software Engineering A Practitioner's Approach", Oxford University Press International Edition, ISBN-13:978-0-19-80-6184-7.
- [10] Naresh Chauhan, "Software Testing Principles and Practices", McGraw-Hill International Edition, ISBN 007-124083-7.
- [11] Isha Arora, Vikram Bali, "A Brief Survey on Web Application Performance Testing Tools Literature Review", International Journal of Latest Trends in Engineering and Technology, Vol. 5, Issue 3, May 2015.
- [12] Dr. Niranjanamurthy M, Kiran Kumar S, Anupama Saha, Dr. Dharmendra Chahar, "Comparative Study on Performance Testingith JMeter", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Special Issue 2, October 2016.