

LITERATURE REVIEW OF PERFORMANCE TESTING ON WEB-BASED APPLICATION USING LOADRUNNER

Vikas Nandal¹, Dr. Kamna Solanki²

¹M. Tech. Student Software Engineering, U.I.E.T., MDU, Rohtak

²H.O.D CSE, Department, U.I.E.T., MDU, Rohtak

Abstract-- Software development life cycle (SDLC) has many phases and testing is one of them. Testing plays an important role in the quality of the product. Testing takes place from the first phase to complete life cycle of the software. Web testing is the process of software testing which describes the testing of the web-based applications. The testing process includes the basic functionality of the application, user interface, accessibility, performances on the heavy traffic etc. There are many testing risks which affecting web applications. To remove these testing risks many testing Tools and Methods are developed by the software companies. In this paper LoadRunner used as web testing tool and methods have been explained which help in the resolving some of the risks in web applications development.

Keywords-- Web application, Web Testing, Tools, Testing Methods, LoadRunner.

I. INTRODUCTION

The software is a collection of related programs and procedures along with their documentation. In a software development, testing consumes a lot of time and it also called the most costly phase of software development. Testing describes that developed software meets its quality standards or not?[1]. As the history of the e-commerce websites is full of software crises. E-commerce websites are designed for business. Information, websites, advertising websites or online shopping websites etc. To deals with such type of challenges testers or developers are stills working on the software testing to ensure the quality or the performance of the website before aired to www(world wide web). Users want the response time or reliability of the website must be best. Because users don't want to compromise with time or money[2].

II. PERFORMANCE TESTING

The performance of a website is affected by many factors such as networks, server, website platform and the end user. Performance testing provides an idea how will application response at different load or stress. Performance is dependent on achieving response times, results and the reliability levels that meet the performance objectives for the website [3]. Load testing describes that if more than one person requesting the same data can the system retrieves the appropriate result to the users or not within the given time? A website should manage user's requests, input data, database connectivity as well as the heavy load on particular data. Stress testing describes that how will any website react beyond its limit? Stress testing is used to distract main functionality of the website.

III. LOADRUNNER

LoadRunner is the product of HP. LoadRunner is used to check the performance of the web applications under various types of load conditions. LoadRunner is used to collect the transaction data from the all the users who are in used. At the same time LoadRunner generate scripts by using the transaction details such as HTTP used to record logging details between server or the client browsers [4]. The component of LoadRunner are-

Load Generator- It is used to generate the load to the application according to the given script.

VuGen- Virtual User Generator is used to create or modify the script to test the application.

Controller- controller is used to monitor real-time status of the data.

Agent process-It is used to manage connections between the controller and load Generators instances. Analysis-assembles logs from various load generators and formats reports for visualization of run result data and monitoring data [5].

IV. COMPARISION WITH OTHER TESTING TOOLS TABLE

Name	Features
LoadRunner	<ul style="list-style-type: none"> Product of HP & license Based application. Windows based system. Supported Applications- Server application, Database, Mobile Application. All Protocols are supported.
Neoload	<ul style="list-style-type: none"> Product of Neotys & license Based application. Support Windows, Linux. Supported Applications- web application servers: (J2EE, .NET, PHP, ASP, AJAX etc) Supported Protocols-HTTP, WebSocket, Java, Oracle, Adobe Flex, GWT, SOAP.

JMeter	<ul style="list-style-type: none"> • Product of Apache & open source application. • Windows, Mac and all UNIX based systems. • Supported Applications-Java, Java Server Pages, Database, Mobile Application. • Protocols- Web(HTTP,HTTPS),Web Services(XML,SOAP),Java based protocols, FTP

V. RELATED STUDIES

Hui-li et. al. “Research of Load Testing and Result Application Based on LoadRunner”(2012)

The heap testing of the web based application framework is the essential test process before discharging the framework, and can discover the bottlenecks of the Web application framework under strain. We Combined with the attributes of the electronic business framework, and did the heap testing and investigation the consequences of load test by methods for the LoadRunner , and found the deformity of the framework when the enormous clients get to the framework, and guided the framework change utilizing the test outcomes. The heap test design and the application strategy are flexible and generally esteem [6].

Kundu “Web Testing: Tool, Challenges and Methods”(2012)

web testing is difficult process which is full of number of challenges along with many risks. On the survey of last few years there are some tools and techniques described which helps to make web testing easier[7].

Rodríguez et.al.“Automated Generation of Performance Test Cases from Functional Tests for Web Applications”(2013)

Authors described new module of GXtest which makes workload contents with better quality and less time. JMeter underpins diverse correspondence conventions, permitting the execution of tests against Systems that are gotten to by various interfaces (HTTP, SOAP,FTP),and dealing with the test brought together in one single apparatus. Along these lines, GXtest can be reached out to different conventions notwithstanding creating for a similar load recreation device. In this paper we displayed how to make executable experiments, however not how to consolidate them keeping in mind the end goal to produce the workload and the non-practical approvals to check that the necessities are being reached [8].

Tyagi “A Comparative Study of Performance Testing Tools”(2013)

In this paper NeoLoad, WAPT and Loadster are used as the performance testing tools for a website at various browsers to analyze the performance of the website. Result have been compared with each other to understand which tool is better for the website testing. But the comparison is difficult due to parameters that used every tools have its different features. This research paper provides a help to find the appropriate tool [9].

Proko and Ninka “Analyzing and Testing Web Application Performance”(2013)

Correlation of Ajax electronic applications are more perplexing then the great web applications. Among numerous complexities, execution testing is one of the troublesome exercises which should be contemplated all the more painstakingly. Execution testing and examining apparatuses help to assess an application under a given load. Creators presented two execution testing and examination apparatuses Firebug and YSlow that can be utilized to acknowledge powerful web application execution tests with insignificant overhead. Investigation of test outcomes is valuable for framework creators to settle on the best decision of the ideal stage in which to construct web applications. Execution devices assist programming designers with finding out bottleneck in framework execution, and in addition execution models to help them to create web applications with high performance [10].

Ramadev et. al. “Analysis of Performance Testing on Web Applications”(2014)

Authors explained the concepts, challenges, goals and types of performance testing for different web applications. In the load testing, system performance measured against the simulated user load where as in stress testing the performance of application is measured against the gradually increasing load. In the last section they described few performance testing tools (LoadRunner, WAPT, JMeter, NeoLoad Etc) for web applications [11].

Nabuco et. al.“Model-Based Test Case Generation for Web Applications”(2014)

This paper proposed a test generation and filtering technique for model-based testing of Web applications. The models contain information in the form of UI Test Patterns linked with connectors. Each UI Test Pattern contains specific configurations with the data needed for test execution. Considering that test cases cannot run forever, this paper presented several filters that were applied to provide flexibility and reduce the number of test cases generated. Initially the filters were connected to the Test Paths and afterward to each Test Path component (a UI Test Pattern) configurations. Some test procedures were then made, each with its own particular broadness and scope. This approach was tried on three deferent web applications. The few test systems were thought about between themselves. Additionally, this approach was contrasted and a similar arrangement of tests performed with a catch replay device. Worldview ended up being effective, as the filtering can give better scope, finding more bugs and slaughtering more mutants. By and large, it is conceivable to see clear preferences of this approach:

– Distributing Test Suites —

Since the test procedure can be configured to create deferent test suites, it is conceivable to spread deferent test suites through deferent workstations and run tests simultaneously and in less time.

– Easier troubleshooting —

In the event that a basic blunder happens that may stop the execution of the testing program out and out, it is anything but difficult to see which component or configuration caused that mistake. All the data is put away so the client can replay the means even physically. Likewise, as the outcomes are refreshed after each Test Path, there is no compelling reason to run all the chose tests again in the event of blunder. The highlights got ready for the not so distant future incorporate a more extensive arrangement of filtering alternatives, test techniques and test scope insights [12].

Bhatti et. al. ” Comparative Study of Load Testing Tools”(2015)

Authors described many tools for testing web application. load testing deals with the performance of the web application under various load conditions and analyze which factors affect the performance of the web application and why? Load testing tools are used for monitoring the performance of web applications. After study of the used tools authors found out that Neoload is good for the testing because of easy script and visual programming. This tool has property of automatic detection and handling of application specific parameters. Drag and drop functions for if statements and loops. It records HTTP traffic between server requests and response. It analyzes results using real time graphs and statistics. So from the properties of mentioned tools, Neoload is chosen as best tool for load testing [13].

Dhiman and Sharma” A Comparative Study and Analysis of Web Service Testing Tools”(2016)

Testing a web service is challenging activity that involves many characteristics such as response time, throughput and latency etc. The same web service has been tested for performance with these web service testing tools such as Apache Jmeter, Grinder, HttpRider and results has been compared. The Comparison helps in the selection of the best tool. This research work can be extended to more tools, more web services and different parameters to provide more realistic results [14].

Kamble and Kharade “ Quantitative Analysis of Manual and Automation Testing and Comparative Study of Selenium and Load Runner Automated Testing Tools“(2016)

Selenium and Load Runner are both effective tools for automation testing. But Selenium is open source tool whereas Load Runner is not open source. Selenium simulates user interaction with the interface whereas Load Runner simulates significant usage. Selenium simulates a user by recording its actions on the interface whereas Load Runner doesn't hassle concerning the interface and records the commands through a proxy. Selenium is mainly use for functional testing whereas Load Runner is use mainly for performance testing [15].

Bhatia and Ganpati. “In Depth Analysis of Web Performance Testing Tools”(2016)

Performance evaluation of a web application requires knowledge of its users and their aggregated behaviour – how many users use the application, when, and which operations they perform, at what frequency In this paper performance testing tools namely Neoload, LoadImpact, Loadster, LoadUI, Webload are evaluated on the basis of properties and characteristics of the tools individually, since the tools only partially overlap in feature sets, and each one offers something the other ones don't. The tools are compared side by side according to the expectations and requirements set out for an ideal performance testing tool. From this observation it is concluded that these tools shows different behaviour under different parameters. Webload is better in terms of response time and throughput. Neoload is better in terms of both memory utilization and CPU utilization. Therefore from the above observation we can conclude the Webload has better performance from LoadUI, LoadImpact, Loadster and Neoload. In future more work can be performed by developing other performance tests such as stress testing, spike testing, stability testing, etc [16].

Thakur et. al.”Web Performance Testing Tools”(2017)

Web application performance testing is a rising and an imperative field of software engineering. As web applications turn out to be more common and complex, the requirement for performance testing is likewise expanding. There are a wide variety of performance testing tools available in the market. Selecting a tool in this area, it is very important to consider the parameters rather than cost. The study provides information to select the best tool for performance testing of web applications according to performance requirement. It is difficult to compare tools because many parameter values are not considered in all tools. JMeter is easy to install, has friendly GUI and it is platform independent, while other tools don't provide the same features. Loadster and LoadUI provide automatic bottleneck detection, while other testing tools don't have this particular feature. So it is concluded that testing tools can be selected according to the requirements. This research work can be extended to more tools with new versions of the tools and different parameters to provide more information about the web performance testing tools[17].

VI. CONCLUSION

Web application performance testing is a rising and an imperative field of software engineering. As web applications turn out to be more common and complex, the requirement for performance testing is likewise expanding. There are a wide variety of performance testing tools available in the market. Selecting a tool in this area, it is very important to consider the parameters rather than cost. LoadRunner provide automatic bottleneck detection, while other testing tools don't have this particular feature. So it is concluded that testing tools can be selected according to the requirements.

REFERENCES

- [1] G. J. Myers, T. M. Thomas, and C Sandler, “the Art of Software Testing”, John Wiley & Son 2004.
- [2] Samaroo, A., Allott, S., & Hambling, B.. “Effective Testing for E-Commerce” 1999.
- [3] Kunhua Zhu, Junhui Fu “Research the performance testing and performance improvement strategy in web application”(ICETC) 2010.
- [4] Mar, Wilson” LoadRunner Architecture ”Wilsonmar.com. 2015.
- [5] Krazit, Tom. "HP snaps up Mercury Interactive". CNET. CBS Interactive Inc. Retrieved 2 April 2015.
- [6] Zhang Hui-li et. Al. “Research of Load Testing and Result Application Based on LoadRunner” 2012.
- [7] Shakti Kundu, “Web Testing: Tool, Challenges and Methods”2012.
- [8] Federo Toledo Rodríguez et.al.“Automated Generation of Performance Test Cases from Functional Tests for Web Applications”2013.
- [9] Sanjay Tyagi,“A Comparative Study of Performance Testing Tools”, International Journal of Advanced Research in Computer Science and Software Engineering 2013.
- [10] Eljona Proko, Ilija Ninka “Analyzing and Testing Web Application Performance” 2013.
- [11] Ramadev et. al. “Analysis of Performance Testing on Web Applications” 2014.
- [12] Miguel Nabuco et. al.“Model-Based Test Case Generation for Web Applications” 2014.
- [13] Sandeep Bhatti, Raj Kumari,” Comparative Study of Load Testing Tools” 2015.
- [14] Shikha Dhiman, Pratibha Sharma,” Performance Testing: A Comparative Study and Analysis of Web Service Testing Tools” 2016.
- [15] Krutika Kamble, Jyoti Kharade,“ Quantitative Analysis of Manual and Automation Testing and Comparative Study of Selenium and Load Runner Automated Testing Tools” 2016.
- [16] Reenu Bhatia Anita Ganpati,“ In Depth Analysis of Web Performance Testing Tools” 2016.
- [17] Shalini, Jawahar Thakur ,”Web Performance Testing Tools” 2017.