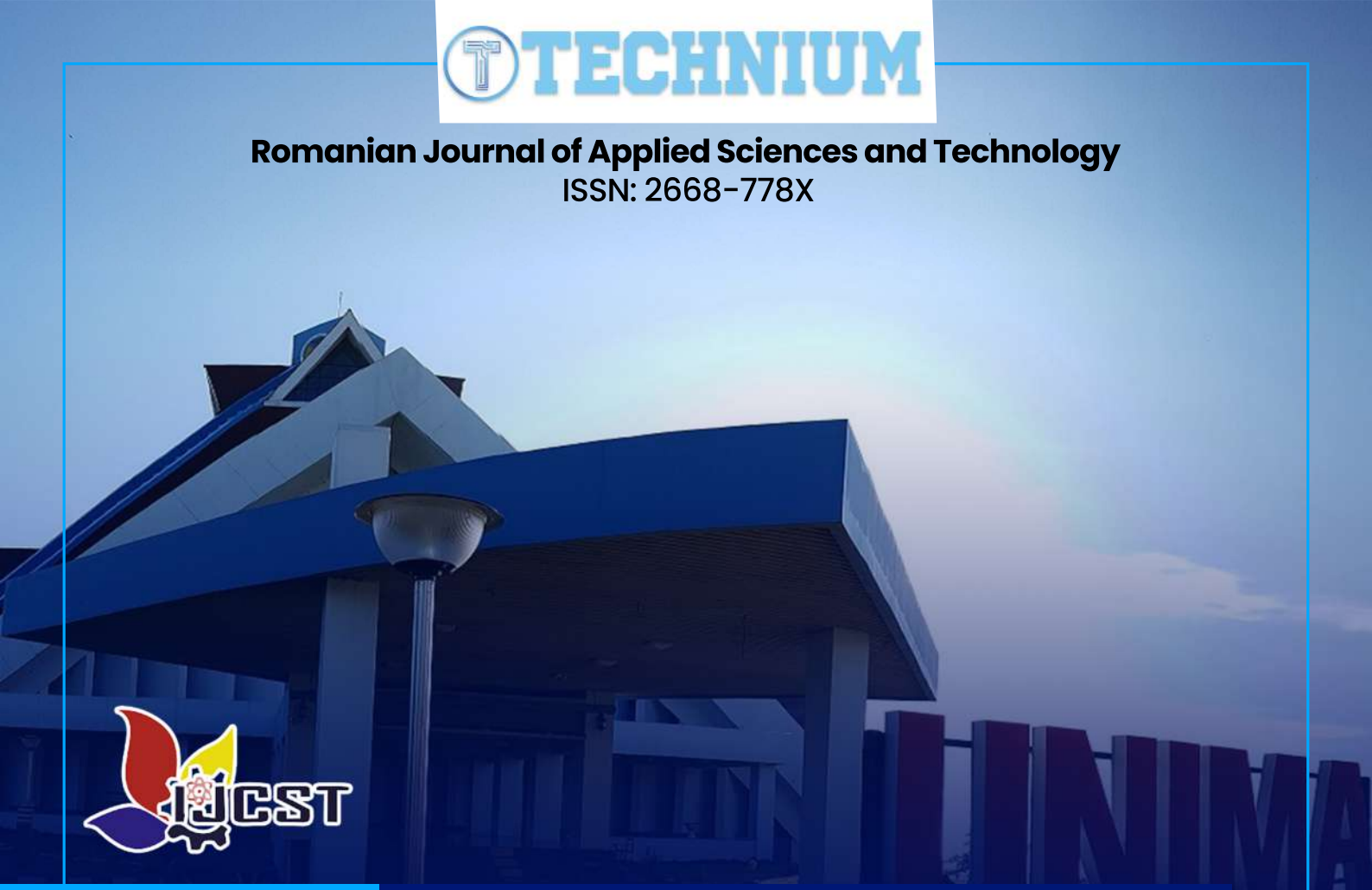
TECHNIUM

Romanian Journal of Applied Sciences and Technology
ISSN: 2668-778X



vol. 16 / 2023

The 7th International Conference on Science Technology

organized by
Faculty of Social Science and
Law Universitas Negeri Manado and
Consortium of International Conference
on Science and Technology

The Innovation Breakthrough in Digital and Disruptive Era

Technium: Romanian Journal of
Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

Quality Assurance of Academic Websites using Performance Testing Tools

Fifin Ayu Mufarroha[1*], *Ahmad* Farisul Haq[1], *Arifatul* Maghfiroh[1], *Devie* Rosa Anamisa[1], *Ahmad* Afif Supianto[2], and *Achmad* Jauhari[1]

[1] Informatics Engineering Department University of Trunojoyo Madura, Bangkalan, Indonesia
[2] Research Center for Artificial Intelligence and Cyber Security, National Research and Innovation Agency (BRIN), Indonesia

Abstract. Academic websites play a crucial role in supporting learning activities and academic growth in the digital age. These websites are created and maintained by academic institutions to provide resources, services, and information to the academic community. Academic websites aim to promote the institution, facilitate communication, and disseminate educational content. However, as these websites develop in complexity and draw more visitors, it is crucial to guarantee their efficient operation and satisfying user experience. For academic websites to operate and be reliable, top-notch software development and thorough software testing are required. The performance testing of academic websites using Apache JMeter and the Web Server Stress Tool is the main topic of this study. This study's conclusions are intended to improve software quality and performance for academic website managers and developers. Gathering data, establishing the test environment, planning and creating the test, carrying it out, and evaluating the system in light of the test results are all steps in the performance testing process. The website operates well under heavy traffic, according to the findings of the performance testing done with Apache JMeter. However, the Web Server Stress Tool shows that the website's performance is insufficient under significant user loads. This test gives information about the website's functionality and any potential problems, which can then be used to determine whether speed improvements are required. **Keyword**: academic website, performance, JMeter, Web Server Stress Tools.

* Corresponding author: fifin.mufarroha@trunojoyo.ac.id

Technium: Romanian Journal of
Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

1 Introduction

Academic websites are now a crucial part of supporting learning activities and academic growth in the current digital age. An academic website is an online resource created and maintained by academic institutions or educational organizations to offer resources, services, and information to the academic community, students, faculty, staff, potential students, and members of the general public with an interest in the academic area [1], [2]. This website's goals are to disseminate educational content, encourage communication between users and academic institutions, and provide information. In general, the academic website serves as a hub for communication and engagement among scholars. It makes learning, academic administration, communication, and institution promotion easier. Educational institutions can improve their efficacy and efficiency in offering information and services to all members of the academic community by developing a strong academic website. However, with academic websites becoming more complicated and having more users visit them, it is crucial to make sure that the website can function effectively and offer users a positive experience. In order to guarantee top performance and dependability, academic websites must be developed and maintained using high-quality software.

Comprehensive software testing is a crucial step in assuring software quality [3], [4]. Software testing tries to find flaws, mistakes, and vulnerabilities in software that could affect the functionality and security of websites. Academic websites may encounter issues including poor performance, instability, or even security flaws if they are not properly tested. A system's quality can be tested using a variety of tools. Apache JMeter and Web Server Stress Tool are the two testing tools for performance websites. Testing using JMeter and Web Stress Tools is part of an automated test. The execution of tests, the comparing of actual results to expected results, the creation of test preconditions, and other test control and reporting functions are all controlled by software. Some software project managers firmly believe the notion that testing is expensive, time-consuming, does not aid in the creation of the product, and may even engender animosity between the testing community and the development team. The software has some benefits that can be attained, including: automated tests can be run concurrently on many machines and are helpful in testing a big test matrix. They are cost-effective if tests need to be repeated numerous times. Automation can be quite helpful if a testing team is comfortable with the idea of automating all or a portion of their tests. When used efficiently at the appropriate points in a testing project, it can: save time, and save money.

Web applications are subjected to load testing, functional testing, and performance testing using the well-liked and potent testing tool Apache JMeter. The Web Server Stress Tool is a tool created specifically to evaluate a web server's durability and functionality by placing a heavy demand on it. Open-source software called Apache JMeter [5] can be used to evaluate the functionality, load, and performance of both static and dynamic resources [6], [7]. It is possible to simulate user load levels and the system's reaction to such levels thanks to the tool's many capabilities. With the help of this tool, testers can evaluate the performance of additional features as well as the website's core operation [8]. The Web Server Stress Tool is software specifically designed to test the performance and reliability of web servers [9]. Using this tool, we may mimic a web server being under a lot of stress and watch the server respond. The Web Server Stress Tool allows the tester to pinpoint and then address web server performance issues. This study aims to identify the performance constraints of academic websites, trace the causes of instability or failure under high load conditions, and make sure that academic websites can handle heavy user traffic without noticeably degrading in performance by combining these two tools. It is envisaged that the findings of this study would be useful for developers and administrators of academic websites in optimizing performance and assuring the website's software quality.

2 Related work

To ensure their quality and efficiency in supporting academic activities in the field of education, academic websites must be assessed. The functioning of the website can be improved in a number of ways, one of which is by testing. Numerous studies have been done to evaluate the value of academic websites using various methods. Dawam et al[10] uses the ISO/IEC 9126 standard to assess an academic website's quality, using the Kano model to rank each attribute in order of importance for improvement. According to the survey, the website may use some work on its functionality, usability, and dependability. The availability of the information should be guaranteed, it should be updated frequently, the layout and structure should be improved, and the organization and clarity of the information should be improved. The results imply that these enhancements are essential for satisfying user expectations and wants.

Sharmistha et al [11] propose a methodology for evaluating the usability of academic websites using the Analytic Hierarchy Process (AHP) model. To assess the usability of the three academic websites, Institute KGP, Institute K, and Institute D, is the sole purpose of this work. Surveys are used in the study to gather user feedback, and the outcomes are contrasted with those of a standardized Website Analysis and Measurement Inventory (WAMMI) usability evaluation approach. The authors conclude that WAMMI is compatible with their AHP-based questionnaire approach and suggest usability criteria for gauging website performance. The study shows how well the AHP-based questionnaire method works for assessing website usability and pinpointing areas that need improvement.

Handaru [12]conduct the performance of Malaysian university websites is compared to those of top-tier universities. To ascertain if Malaysian university websites adhere to appropriate performance and quality

Technium: Romanian Journal of
Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

criteria, the study takes into account elements including page size, download time, broken links, and link popularity. The results show that websites for Malaysian universities typically fall short of these requirements. The study contends that additional work is required to enhance website design and abide by accepted standards. The study also emphasizes how crucial it is for future studies to take user feedback and a cultural viewpoint into account.

Using the search engine optimization (SEO) techniques, Kamran et al [13] compare the performance of websites for domestic and international universities. The study's objective is to evaluate these websites' advantages and disadvantages in order to identify which ones might be improved and optimized. The study's findings indicate that, in general, overseas universities perform better in SEO than local universities. According to the report, academic websites should adhere to White Hat SEO guidelines to raise their search engine rankings. The analysis emphasizes the value of elements like technology, printability, print quality, and accessibility in deciding website rankings. Overall, this research provides insights into the SEO features necessary for developing academic websites and achieving better search engine ranking.

3 Methodology

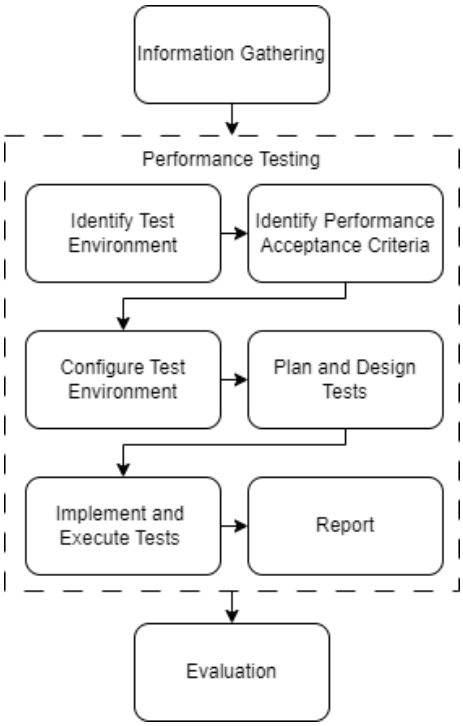


Fig. 1. Overview test model

Figure 1 depicts how the research was conducted in general. In order to conduct the test, information on the study object and system performance test instruments must first be gathered. Performance testing is the next step in this process, which starts with identifying the test environment, identifying the performance acceptance criteria, planning and designing the test, configuring the test environment, implementing and executing the test, and reporting. The sub-chapters of each tool will go into the stages of performance testing in great depth. The process is completed by evaluating the system in light of the performance test findings.

3.1 Apache JMeter

JMeter is a desktop program created to evaluate the functionality and performance of client/server programs such as web applications and FTP programs [14], [15]. It is unquestionably one of the most popular open-source, publicly available testing applications available on the Internet. It is entirely Java-based and has a built-in API (Application Programming Interface) that makes it very flexible. Response time as well as all other server resources, including CPU loads, memory utilization, and resource usage, are measured. JMeter can be used successfully for functional test automation in this regard. The following is a description of the Jmeter performance testing phases:

3.1.1 Identify test environment

A test environment is a configuration that specifies the settings and parameters for carrying out performance tests in Apache JMeter, a well-known open-source performance testing application. May evaluate the performance of web apps, APIs, and other services under diverse loads and conditions by simulating various test scenarios with JMeter. JMeter's test environment includes a variety of elements and parameters that let carry out performance testing successfully. The following are some essential elements of the test environment in JMeter:

- Thread Group: The Thread Group is where test strategy should begin. In order to mimic concurrent users accessing the target application, it specifies the number of virtual users (threads) and the ramp-up time (how long it takes to start all the threads).
- Samplers: Samplers simulate various server requests, including HTTP, FTP, and JDBC database queries.
- Controllers: The flow of test plan can be organized with the use of controllers.
- Listeners: Listeners offer a range of reports and test result visualizations.
- Timers: Timers create delays between thread requests, replicating the behavior of real users who don't always act immediately.
- Assertions: Assertions check that the server's answer satisfies predetermined standards.
- Configuration elements: The behavior of samplers or other test components can be changed by using additional settings provided by configuration elements.
- Can modify requests and replies both before and after they are sent to the server using pre- and post-processors.

To develop accurate and relevant performance tests, these elements must be configured correctly while

setting up the test environment in JMeter. This guarantees that test scenarios adequately reflect actual usage patterns and aids in the discovery of potential application or system performance bottlenecks and problems.

3.1.2 Plan and design test

To run performance tests on web application, API, or service, planning and designing tests in JMeter entails developing a clear and concise test plan. The steps that need to be followed in order to efficiently plan and create tests in JMeter include (1) Determine the test objectives (establish the precise goals for performance tests), (2) Identify test scenarios (the many situations want to test should be defined), (3) Make a plan, (4) Add test elements, (5) Configure samplers, (6) Add assertions, (7) Integrate logic controllers, (8) Set up listeners, (9) Add pre-processors and post-processors, configure thread group and test plan setting, (10) Validation and interative testing, and (11) Analyze result (utilize the different listeners to examine test results, spot performance issues, and make the required adjustments to application).

3.1.3 Configure test environment

Setting up numerous elements and parameters to emulate the necessary test circumstances for performance testing is part of configuring the test environment in JMeter. Create a user-defined variable that reads the environment variable before using it in the following test execution phase.

3.1.4 Implement and Execute Test

The test is currently done on JMeter after the previous stages have been completed. How the test is run is decided by the preceding step. The earlier stages outline a sequence of actions that JMeter will take when it is run. One or more Thread Groups, logic controllers, sample-generating controllers, listeners, timers, assertions, and configuration items make up a full test plan.

3.1.5 Report

To assess the outcomes of performance testing, JMeter offers a variety of report formats. These reports offer insightful information about the application's efficiency, response times, mistakes, and other pertinent parameters. Plugins and built-in listeners provided by JMeter make it possible to see and export test results in a variety of formats. In JMeter, a few methods that are frequently used to produce reports include: view results tree listener, summary report listener, aggregate report listener, aggregate graph listener, view results in table listener, response time graph listener, and JTL log files [16].

3.2 Web Server Stress Tool

In order to find critical performance issues that might limit the best user experience, web server stress tools simulate a large number of users by sending HTTP requests to a web server. Testing a website's stability and performance under high traffic situations is known as website stress testing. This kind of testing enables website owners and developers to comprehend the limitations of their platform and pinpoint places in need of development. Website stress testing helps verify that a website can handle the demands of a large number of users and prevent problems like poor loading times or crashes by simulating high levels of traffic. To make sure a website can handle the demands of a high number of visitors and to pinpoint areas for development, it is crucial to undertake frequent website stress tests. Webserver Stress Tool by Paessler is one of the available web server stress tools. It is a potent HTTP-client/server test tool made to identify serious performance problems in websites or web servers that can impede users from having the best possible experience. It can perform performance, load, and stress tests on nearly any HTTP server.

3.2.1 Identify test environment

In Paessler's Webserver Stress Tool, identifying the test environment is essential for precise testing, performance optimization, resource allocation, and troubleshooting. For numerous reasons, it's crucial to identify the test environment in Paessler's Webserver Stress Tool such as testing precisely, enhancing performance, allocating resources, and troubleshooting.

3.2.2 Identify performance accaptance criteria

The requirements that must be satisfied for a web server or website to be deemed acceptable in terms of performance are known as performance acceptance criteria. Some of the performance acceptance criteria included in Paessler's Webserver Stress Tool include the following:

- The amount of time it takes for the web server to react to a request is known as response time. To provide the best user experience, the response time should fall within a reasonable range.
- The amount of data that can be transferred between a web server and a client in a specific amount of time is known as throughput. The throughput must to be sufficient to deal with anticipated traffic and usage patterns.
- The percentage of requests that encounter errors, such as 404 errors or server timeouts, is known as the error rate.
- Scalability: The capacity of the web server or website to manage rising usage and traffic volumes.
- Stability: This describes a web server's or website's capacity to sustain peak performance over an extended period of time.

Technium: Romanian Journal of
Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

3.2.3 Identify performance accaptance criteria

The requirements that must be satisfied for a web server or website to be deemed acceptable in terms of performance are known as performance acceptance criteria. Some of the performance acceptance criteria included in Paessler's Webserver Stress Tool include the following:

- The amount of time it takes for the web server to react to a request is known as response time. To provide the best user experience, the response time should fall within a reasonable range.
- The amount of data that can be transferred between a web server and a client in a specific amount of time is known as throughput. The throughput must to be sufficient to deal with anticipated traffic and usage patterns.
- The percentage of requests that encounter errors, such as 404 errors or server timeouts, is known as the error rate.
- Scalability: The capacity of the web server or website to manage rising usage and traffic volumes.
- Stability: This describes a web server's or website's capacity to sustain peak performance over an extended period of time.

3.2.4 Plan and design test

The following are the processes to plan and create a test in Paessler's Webserver Stress Tool. (1) Define the test's objectives, establish the test's objective, such as detecting performance problems or evaluating the web server's scalability. (2) Choose the testing environment, indicate the IP address, port number, and protocol of the web server that want to test in detail. (3) Choose the exam scenarios, define the various test scenarios, such as simulating heavy traffic or evaluating the web server's capacity to accommodate ongoing usage. (4) Set up the test parameters, configure the test parameters, including the number of users to simulate, the time between requests, and the length of the test, using the Webserver Stress Tool. (5) Execute the test, launch the test and keep an eye on the web server's performance. Utilize the tool to gather information and produce thorough test logs and graphs. (6) Examine the outcomes, to find any serious performance issues that can be affecting the website or web server, examine the test logs and graphs. (7) Test it one more, repeat the test, if necessary, with various scenarios or setups to learn more about the web server's performance.

3.2.5 Configure test environment

Some considerations to keep in mind when configuring Paessler's Webserver Stress Tool's test environment include changing the tool's settings to replicate realistic user behavior, specifying how many virtual users there will be and how they will behave, and modifying the network settings to reflect actual usage.

3.2.6 Implement and execute test

The phases in the JMeter test are given the same treatment at this point. In this phase, the test is executed in accordance with the preset setup.

3.2.7 Report

Reports can be generated and analyzed using Paessler's Webserver Stress Tool. The report generating, test documentation, and manual reporting features are summarized here. It is possible to create reports in Word format using report generation. These reports offer a thorough analysis of the exam outcomes, complete with performance indicators, graphs, and data. Paessler offers example performance tests with thorough documentation, including graphs and statistics, in test documentation. These sample tests can be used as a guide for writing our own test documentation. Paessler provides instructions on how to utilize the Webserver Stress Tool in the documentation. The tool's documentation has comprehensive instructions on how to use it, including guidance on creating reports and deciphering the outcomes.

4 Result

4.1 Information gathering

This study's focus is the Trunojoyo University academic web (https://www.trunojoyo.ac.id/). This university website offers numerous details for students, instructors, and the general public on academic programs, campus events, announcements, and services.

4.2 Apache JMeter test results

When an application is being tested for performance, its speed is examined to see if it satisfies the standards set by its users. It can be used to find application bottlenecks that could cause the system as a whole to lag. Load and stress testing also have certain things in common with performance testing. This test involves entering a user simulation of 100 to 500 people with a ramp-up period of 1 second over the course of 5 iteration in order to evaluate the performance of a website. as seen in Table 1.

The results of testing the website can be seen in the Table 1, with the highest average received in iteration 5 with a simulation of 500 people and the lowest average received in iteration 1 with 100 users. The five iterations obtain the most throughput, whereas the one iteration gets the lowest. A rise in incoming requests is depicted in Figure 2's aggregate graph of the website with 100 user simulations in 1 second. 0 to 1.591 milliseconds on a scale represents the outcomes. Figure 3 shows the result graph for the website trunojoyo.ac.id with a time utilized of 1 ms, and it shows the throughput value as a green dotted line with 1,552.795 values being displayed per minute. The reaction time

Technium: Romanian Journal of Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

graph in Figure 4 depicts the response time as the user rises. Whereas more testing time is required the more users there are.

Table 1. Test results with apache JMeter

No	User	Ramp -up	Error [%]	Received [KB/s]	Through put [m]
1	100	1	0.00	1901.68	1,552.795
2	200	1	0.00	2733.37	2,244.669
3	300	1	8.00	2677.54	2,373.731
4	400	1	12.50	2512.09	2,050.756
5	500	1	44.20	2785.55	4,043.127

Fig. 2. Aggregate graph

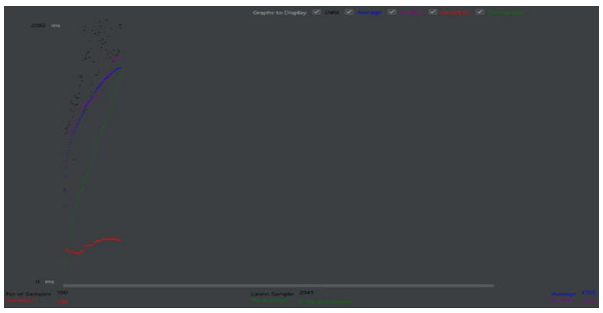


Fig. 3. Graph result

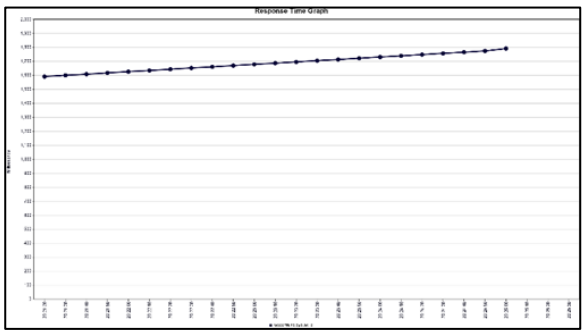


Fig. 4. Response time graph

4.3 Web Stress Tools test results

Table 2 displays the test results from five rounds using the web server stress tool with a simulation of 100 to 500 users. According to the results of the Web Server Stress Tool's rate-of-results analysis, iteration error for iteration 1 was 0%, however iteration error for iteration 2 through iteration 5 was 100%. For average click time, each iteration receives a score of 0. Additionally, starting from the third time the pengujian was run, the result was zero.

Table 2. Test results with the Web Server Stress Tool

No	User	Time	Error [%]	Average Click Time [ms]	Time Spent[ms]
1	100	1	0	0	0
2	200	1	100.00	0	0
3	300	1	100.00	0	0
4	400	1	100.00	0	0
5	500	1	100.00	0	0

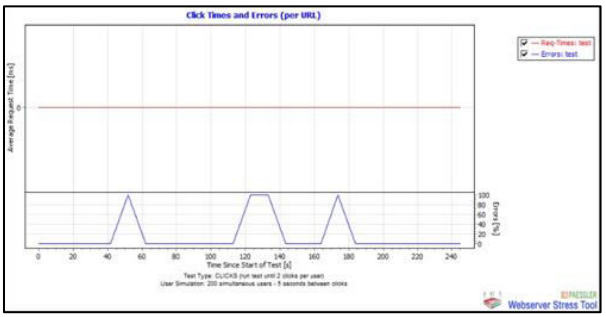


Fig. 5. Graph click time & error

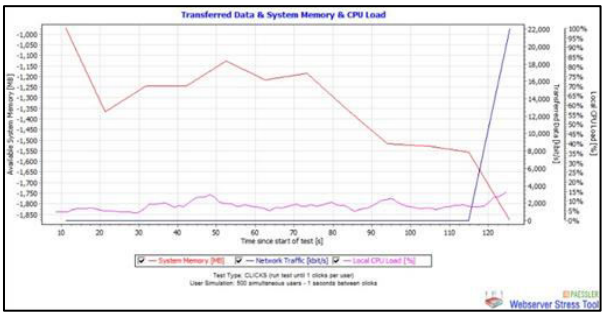


Fig. 6. Test client health

Get the average performance by graphing click times and errors on a simulation of 200 users in Figure 5. Click time is 0 milliseconds, but the percentage of errors that occur increases by 100% in the ranges of 40–60, 120–140, and 160–180. Figure 6 shows that the system memory of the website has grown to 1,850 MB, even if the CPU load is only 15% and network traffic has gone from 0 kbit/s to 22,000 kbit/s at the 120th second. Figure 7 shows how unstable network traffic and data delivered to the server always increase and decrease by a significant amount, and how performance falls from 200 to 0 in the 180 second. Figure 8 further shows that while the hits/s and clicks/s graphic fluctuate, they do not do so for longer than 9.5s, while the click time is consistently stable.

Technium: Romanian Journal of Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

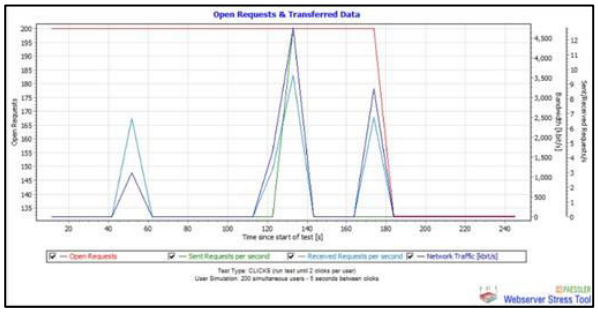


Fig. 7. Open request & transferred data

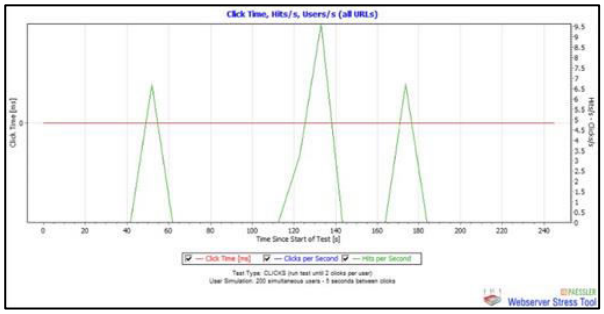


Fig. 8. Click time, hits/s, and click/s

5 Conclusion

According to the Apache JMeter test findings, the test with the largest user simulation, or 500 users, yielded the highest Throughput value. The most crucial factor in determining how well a server can manage severe loads is throughput. Whereas server performance improves as the Throughput value rises. In contrast, from the second to the fifth cycle of testing utilizing the Web Server Stress Tool, the error results reached 100%. In all iterations, the average click time and time spent values are equal to 0. As a result, the writers come to two conclusions on the Trunojoyo University Madura website's quality. First, when tested with Apache JMeter, it receives a high Throughput value, indicating that the website performs well when a large number of users access it. Second, the Web Server Stress Tool's website performance is insufficient when dealing with significant user loads since mistakes keep growing as the number of users rises. It is intended that by conducting this test, it would be possible to gain a better knowledge of how well the Trunojoyo Madura University website functions and help detect any issues that may arise. Additionally, it is anticipated that this data will serve as the foundation for any necessary website speed upgrades.

References

1. M. H. Brannen, S. Milewski, and T. Mack, “Providing staff training and programming to support people with disabilities: An academic library case study,” *Public Services Quarterly*, vol. 13, no. 2, pp. 61–77, 2017.
2. J. C. Moore, “A Synthesis of Sloan-C Effective Practices, December 2008.,” *Journal of Asynchronous Learning Networks*, vol. 12, pp. 99–118, 2008.
3. P. Bourque, R. Dupuis, A. Abran, J. W. Moore, and L. Tripp, “The guide to the software engineering body of knowledge,” *IEEE Softw*, vol. 16, no. 6, pp. 35–44, 1999.
4. M. H. Fallah and A. M. Jrad, “SQA—A proactive approach to assuring software quality,” *AT&T Technical Journal*, vol. 73, no. 1, pp. 26–33, 1994.
5. R. K. Lenka, S. Mamgain, S. Kumar, and R. K. Barik, “Performance analysis of automated testing tools: JMeter and TestComplete,” in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, IEEE, 2018, pp. 399–407.
6. E. N. Alam and F. Dewi, “PERFORMANCE TESTING ANALYSIS OF BANDUNGTANGINAS APPLICATION WITH JMETER,” *International Journal of Innovation in Enterprise System*, vol. 6, no. 02, pp. 157–166, 2022.
7. P. Memon, T. Hafiz, S. Bhatti, and S. S. Qureshi, “Comparative study of testing tools Blazemeter and Apache JMeter,” *Sukkur IBA Journal of Computing and Mathematical Sciences*, vol. 2, no. 1, pp. 70–76, 2018.
8. R. Hidayanto and P. Sawitri, “Performance Testing of e-Payment Website Using JMeter,” *International Research Journal of Advanced Engineering and Science*, vol. 4, no. 3, pp. 350–352, 2019.
9. H. Sarojadevi, “Performance testing: methodologies and tools,” *Journal of Information Engineering and Applications*, vol. 1, no. 5, pp. 5–13, 2011.
10. Institute of Electrical and Electronics Engineers. Indonesia Section and Institute of Electrical and Electronics Engineers, *2015 3rd International Conference on Information and Communication Technology (ICoICT)*.
11. S. Roy, P. K. Pattnaik, and R. Mall, “Quality assurance of academic websites using usability testing: an experimental study with AHP,” *International Journal of System Assurance Engineering and Management*, vol. 8, no. 1, pp. 1–11, Mar. 2017, doi: 10.1007/s13198-016-0436-0.
12. H. Jati, “Performance Evaluation of Malaysian University Website.” [Online]. Available: http://www.websiteoptimization.com
13. K. Rasheed, M. Noman, M. Imran, M. Iqbal, Z. M. Khan, and M. M. Abid, “PERFORMANCE COMPARISON AMONG LOCAL AND FOREIGN UNIVERSITIES WEBSITES USING SEO TOOLS,” *ICTACT JOURNAL ON SOFT COMPUTING*, p. 2, 2018, doi: 10.21917/ijsc.2018.0223.
14. E. H. Halili, *Apache JMeter*. 2008.
15. N. Jha and R. Popli, “Comparative Analysis of Web Applications using JMeter.,” *International*

Technium: Romanian Journal of Applied Sciences and Technology
Vol. 16, 226-233, October, 2023
ISSN: 2668-778X
www.techniumscience.com

Journal of Advanced Research in Computer Science, vol. 8, no. 3, 2017.
16. S. Kaur, S. Hooda, and H. Deo, “Software Quality Management by Agile Testing,” *Agile Software Development: Trends, Challenges and Applications*, pp. 221–233, 2023.