

Uji Performa Aplikasi Sistem Informasi Guru SMK Bali Global Abiansemal Menggunakan Apache JMeter Dan Blazemeter

Indrianto¹, Dimas Nurhamzah²

Fakultas Informatika Dan Ilmu Komputer
Institut Teknologi Dan Bisnis STIKOM Bali
Denpasar, Indonesia

e-mail: ¹indrianto.kuseni@gmail.com, ²hamzahdimas85@gmail.com

ABSTRAK

Sistem informasi Guru SMK Bali Global Abiansemal, merupakan sebuah aplikasi yang dikembangkan untuk melayani administrasi guru di SMK Bali Global Abiansemal. Dengan bertambahnya jumlah guru yang bertugas, maka perlu dilakukan pengujian terhadap kinerja server dalam menangani request setiap guru. Penelitian ini bertujuan untuk melakukan pengujian performa pada Sistem Informasi Guru di SMK Bali Global Abiansemal menggunakan alat pengujian ApacheJMeter dan Blazemeter. Pengujian performa dilakukan dengan menguji empat halaman website, yaitu halaman login, update profil guru, upload foto, dan buat soal. Pengujian dilakukan dengan menggunakan 50 dan 100 sampel dengan periode rampup 10 detik dan loopcount 1. Hasil pengujian menunjukkan bahwa sistem memiliki kinerja yang baik pada fitur-fitur tersebut, dengan rata-rata waktu respons yang konstan, throughput yang meningkat seiring dengan jumlah permintaan, dan deviasi yang menurun. Informasi ini memberikan wawasan tentang kinerja sistem dan dapat digunakan untuk melakukan perbaikan dan optimalisasi guna meningkatkan responsivitas dan kualitas pengalaman pengguna. Pengujian performa menggunakan ApacheJMeter dan Blazemeter dapat menjadi metode yang efektif dalam menguji performa sistem informasi berbasis web.

Kata Kunci: sistem informasi, load testing, performa website, respons waktu, beban pengguna, skalabilitas, beban server, ApacheJMeter, Blazemeter.

ABSTRACT

Teacher Information System of SMK Bali Global Abiansemal is application developed to serve teachers in carrying out their duties. With the increasing number of teachers on duty, it is necessary to test server performance in handling requests for each teacher. This study aims to test the performance of the Teacher Information System at SMK Bali Global Abiansemal using the ApacheJMeter and Blazemeter. Performance testing is done by testing four website pages, namely the login page, updating the teacher's profile, uploading photos, and creating questions. Tests were carried out using 50 and 100 samples with a rampup period of 10 seconds and a loopcount of 1. The test results show that the system has good performance on these features, with a constant average response time, throughput which increase with the number of requests, and a decreasing deviation. This information provides insight into system performance and can be used to make improvements and optimizations to improve responsiveness and quality of the user experience. Performance testing using Apache J Meter and Blazemeter can be an effective method for testing the performance of web-based information systems.

Keywords: *information system, load testing, website performamce, response time, user load, skalability, server load, Apache JMeter, Blazemeter*

1. PENDAHULUAN

Sistem informasi merupakan suatu kombinasi teratur dari orang-orang, hardware, software, jaringan komunikasi, serta sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi [1]. Sistem Informasi Guru SMK Bali Global Abiansemal, merupakan sebuah sistem informasi berbasis web yang dikembangkan oleh SMK Bali Global Abiansemal Badung. Memanfaatkan dukungan teknologi informasi melalui sebuah sistem informasi, proses belajar mengajar, pembuatan ujian, serta input hasil ujian dapat dilakukan dengan lebih cepat dan murah. Sistem informasi yang diperuntukkan bagi guru di SMK Bali Global Abiansemal Badung merupakan solusi agar pelayanan belajar mengajar dan administrasi dapat dilaksanakan dengan lebih efektif dan efisien.

Pengujian performa aplikasi berbasis website adalah proses yang dilakukan untuk mengevaluasi kinerja suatu website dalam menghadapi beban lalu lintas yang tinggi. Tujuan dari pengujian ini adalah untuk memastikan bahwa website mampu memberikan respons yang cepat, ketersediaan yang tinggi, dan kinerja yang optimal kepada pengguna [2].

Dalam era digital saat ini, pengguna memiliki harapan yang tinggi terhadap waktu respons yang singkat dan pengalaman pengguna yang baik saat mengakses website. Oleh karena itu, pengujian performa website menjadi sangat penting guna memastikan bahwa website dapat memenuhi harapan tersebut.

Pengujian performa website melibatkan simulasi beban lalu lintas dengan menggunakan alat-alat pengujian seperti ApacheJMeter dan Blazemeter. Alat ini memungkinkan pengguna untuk mensimulasikan akses ke website dengan skenario yang berbeda, termasuk jumlah pengguna yang besar dan aktivitas yang intens [2].

Apache JMeter adalah alat pengujian beban yang serbaguna dan open-source yang digunakan untuk melakukan pengujian performa pada aplikasi web. Dikembangkan oleh Apache Software Foundation, JMeter dapat mensimulasikan beban pengguna yang tinggi dengan mengirimkan permintaan HTTP ke server web dan mengukur kinerja serta responsivitasnya. JMeter dirancang untuk mendukung pengujian performa yang komprehensif.

Blaze Meter adalah platform pengujian performa yang digunakan untuk mengukur, menganalisis, dan mengoptimalkan kinerja aplikasi web, situs web, layanan web, dan infrastruktur terkait. Ini memungkinkan pengguna untuk melakukan pengujian beban dan stres yang realistis pada aplikasi web mereka.

Blaze Meter menawarkan fitur-fitur yang kuat untuk pengujian performa, termasuk kemampuan merekam dan memutar kembali skenario pengujian, skala yang fleksibel untuk mensimulasikan beban lalu lintas yang beragam, dan pemantauan kinerja waktu nyata. Platform ini juga menyediakan analisis mendalam tentang metric performa kunci, seperti waktu respons, kecepatan transaksi, jumlah permintaan per detik, dan tingkat kesalahan.

Dalam konteks penelitian ini, BlazeMeter digunakan sebagai salah satu alat untuk merekam skenario pengujian pada sistem informasi guru SMK Bali Global Abiansemal. Dengan menggunakan BlazeMeter, peneliti dapat merekam berbagai aktivitas pengguna, seperti login,

update profil guru, upload foto, dan membuat soal, dan kemudian memutar kembali scenario tersebut untuk mengevaluasi kinerja sistem.

Dalam pengujian performa website, beberapa parameter yang diukur meliputi: 1) Respons Waktu: Waktu yang dibutuhkan oleh server untuk memberikan respons terhadap permintaan dari pengguna. Waktu respons yang lambat dapat mengakibatkan pengalaman pengguna yang buruk dan kemungkinan tingkat peningkatan bouncerate [3]. 2) Beban Pengguna: Jumlah pengguna yang di-simulasikan untuk mengakses website secara bersamaan. Pengujian dilakukan dengan meningkatkan jumlah pengguna untuk melihat bagaimana website menangani beban lalu lintas yang tinggi [4]. 3) Skalabilitas: Kemampuan website untuk mengatasi peningkatan jumlah pengguna atau beban lalu lintas tanpa mengalami penurunan kinerja. Pengujian ini bertujuan untuk menentukan batasan maksimum website dan kapasitasnya dalam menangani lalu lintas yang tinggi [5]. 4) Beban Server: Pengujian ini melibatkan pengukuran penggunaan sumber daya server seperti CPU, memori, dan bandwidth selama berlangsungnya pengujian. Hal ini penting untuk memastikan bahwa server dapat menangani beban lalu lintas yang tinggi tanpa mengalami kegagalan [6].

Hasil dari pengujian performa website memberikan wawasan tentang kinerja dan kemampuan website dalam menangani beban lalu lintas yang tinggi. Informasi ini dapat digunakan untuk melakukan perbaikan dan optimalisasi guna meningkatkan responsivitas dan kualitas pengalaman pengguna.

2. METODE PENELITIAN

Penelitian ini merupakan studi kasus testing pada Sistem Informasi Guru di SMK Bali Global Abiansemal, metode *load testing* digunakan dalam penelitian pada server SMK Bali Global Abiansemal. Tahapan yang dilakukan adalah sebagai berikut.

- a) Identifikasi Tujuan: Tentukan tujuan pengujian performa yang ingin dicapai. Misalnya, tujuan dapat berupa mengukur waktu respons website, menentukan batasan beban maksimum yang dapat ditangani, atau mengidentifikasi penyebab penurunan kinerja [2].
- b) Penentuan Skenario Pengujian: Definisikan skenario pengujian yang akan digunakan. Skenario pengujian dapat mencakup jumlah pengguna, tipe aktivitas yang dilakukan, dan tingkat beban lalu lintas yang dihasilkan. Skenario ini harus mencerminkan perilaku pengguna yang realistis [3].
- c) Persiapan Lingkungan Pengujian: Siapkan infrastruktur dan lingkungan pengujian yang sesuai. Ini meliputi mengatur server, mengonfigurasi ApacheJMeter, dan memastikan ketersediaan sumber daya yang cukup untuk melakukan pengujian [4].
- d) Konfigurasi ApacheJMeter: Atur ApacheJMeter sesuai dengan skenario pengujian yang telah ditentukan. Konfigurasikan threadgroup untuk mengatur jumlah pengguna dan perilaku pengguna. Definisikan pengukuran performa yang ingin diambil, seperti respons waktu atau beban server [5].
- e) Pelaksanaan Pengujian: Jalankan skenario pengujian menggunakan ApacheJMeter. Monitor dan catat hasil pengujian yang meliputi metrik performa yang diukur, seperti waktu respons, beban server, dan tingkat kesalahan [6].
- f) Analisis Hasil: Analisis hasil pengujian untuk mendapatkan wawasan tentang kinerja website. Identifikasi faktor-faktor yang mempengaruhi kinerja, seperti bottleneck atau penggunaan sumber daya yang berlebihan. Evaluasi apakah kinerja website memenuhi tujuan yang telah ditetapkan [7].

- g) Tindakan Perbaikan dan Optimalisasi: Jika terdapat masalah atau batasan dalam kinerja website, lakukan tindakan perbaikan dan optimalisasi yang diperlukan. Misalnya, meningkatkan kapasitas server, memperbaiki kode aplikasi, atau melakukan caching untuk mengurangi beban server [8].

3. HASIL DAN PEMBAHASAN

3.1 Pengujian Load Testing

Pengujian *load testing* dilakukan dengan merekam web sistem informasi guru SMK Bali Global Abiansemal dengan menggunakan aplikasi *blazemeter* dan *apachejmeter*, pengujian ini dilakukan dengan menggunakan beberapa halaman antara lain halaman login, update profile guru, upload foto, dan create soal.

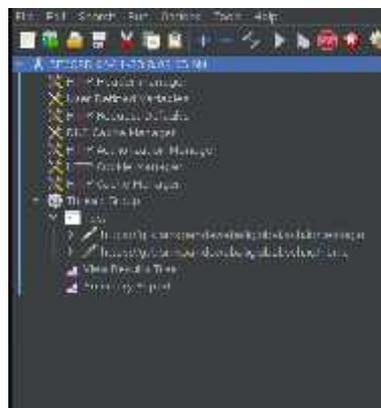
3.1.1 Skenario Login

Pada skenario login, terlebih dahulu dijalankan *blazemeter* untuk merekam proses login, sebagaimana terlihat pada gambar 2.



Gambar 2. Proses merekam halaman login menggunakan *blazemeter*

Hasil rekaman pada *blazemeter* sebagaimana nampak pada gambar 3, merupakan skenario testing yang akan dijalankan dengan Apache JMeter. Skenario testing menggunakan sample 50 dan 100 sample, dengan periode *ramp up* adalah 10 detik dan *loop count* adalah 1 kali.



Gambar 3. Membuka hasil rekaman dari *blazemeter*

Selanjutnya dilakukan pengujian skenario testing pada 50 sample, sebagaimana nampak pada gambar 4. Hasil pengujian menunjukkan waktu tersingkat untuk mengeksekusi adalah 228 ms dan waktuterlama untuk mengeksekusi adalah 1368 ms, sehingga rata-rata waktu untuk bisa mengeksekusi proses ini adalah 440 ms. Standar deviasi 246.59, dengan jumlah error 0%, rata-rata throughput 4.95/sec, rata-rata penerimaan data 3.395 Kb/sec, rata-rata pengiriman data 6.07 Kb/sec.

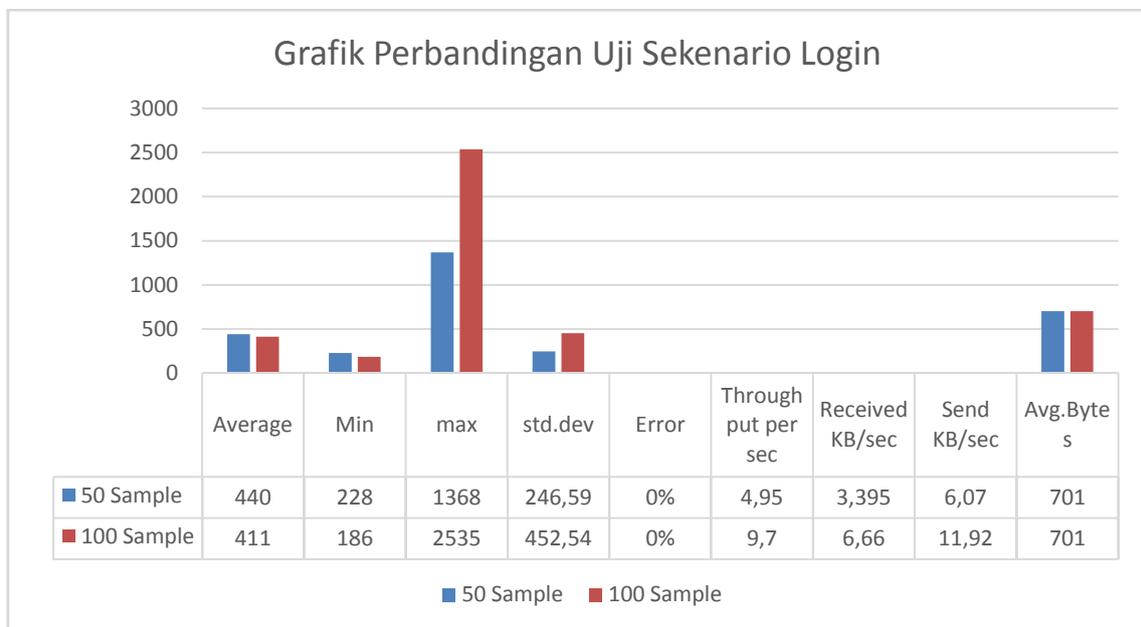
Job	Average	Min	Max	Std. Dev.	Error %	Throughput	Received (KB/sec)	Send (KB/sec)	Avg. Bytes
50	440	228	1368	246.59	0.0%	4.95/sec	3.395	6.07	701

Gambar 4. Hasil pengujian 50 sampel pada apacheJMeter

Selanjutnya dilakukan pengujian dengan menggunakan 100 sample. Hasil pengujian sebagaimana nampak pada gambar 5. Waktu minimal untuk dapat mengeksekusi adalah 186 ms, dan waktuterlama adalah 2535 ms. Rata-rata waktu yang diperlukan untuk mengeksekusi adalah 411 ms. Standar deviasi 452.54, dengan jumlah error 0%, rata-rata throughput 9.7/sec, rata-rata penerimaan data 6.66 Kb/sec, rata-rata pengiriman data 11.92 Kb/sec.

Job	Average	Min	Max	Std. Dev.	Error %	Throughput	Received (KB/sec)	Send (KB/sec)	Avg. Bytes
100	411	186	2535	452.54	0.0%	9.7/sec	6.66	11.92	701

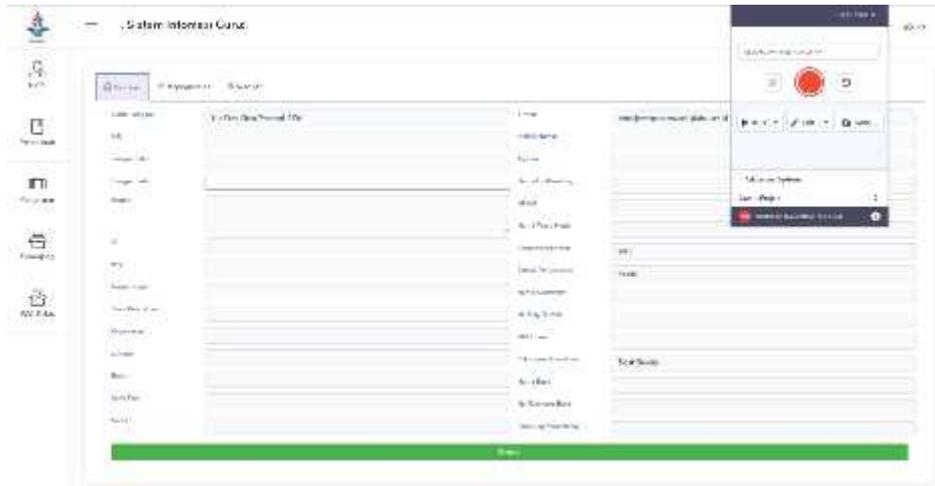
Gambar 5. Hasil pengujian dengan 100 sample pada ApacheJMeter



Gambar 6. GrafikPerbandingan Uji Skenario Login

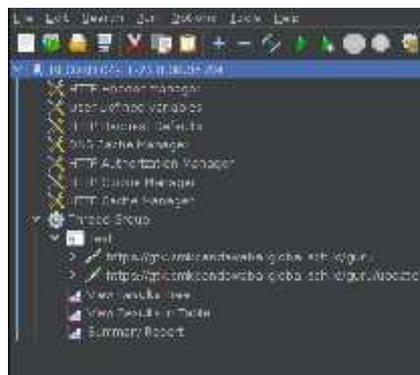
3.1.2 Skenario update guru

Skenario halaman update guru adalah untuk memantau proses update profile guru. Proses perekaman halaman ini sebagaimana terdapat pada gambar7.



Gambar 7. Proses perekaman halaman update guru menggunakan blazemeter

Selanjutnya skenario testing halaman update guru pada Apache JMeter, sebagaimana nampak pada gambar 8. Skenario testing ini menggunakan 50 dan 100 sampel dengan periode *rampup* adalah 100 detik dan *loopcount* adalah 1 kali.



Gambar 8. Skenario testing pada ApacheJMeter

Hasil dari pengujian skenario testing update guru terhadap 50 sampel sebagaimana nampak pada gambar9. Waktu minimal untuk dapat mengeksekusi adalah 137 ms, dan waktuterlama adalah 1211 ms. Rata-rata waktu yang diperlukan untuk mengeksekusi adalah 224 ms. Standar deviasi 182.39, dengan jumlah error 0%, rata-rata throughput 5.0/sec, rata-rata penerimaan data 3.47 Kb/sec, rata-rata pengiriman data 4.65 Kb/sec.

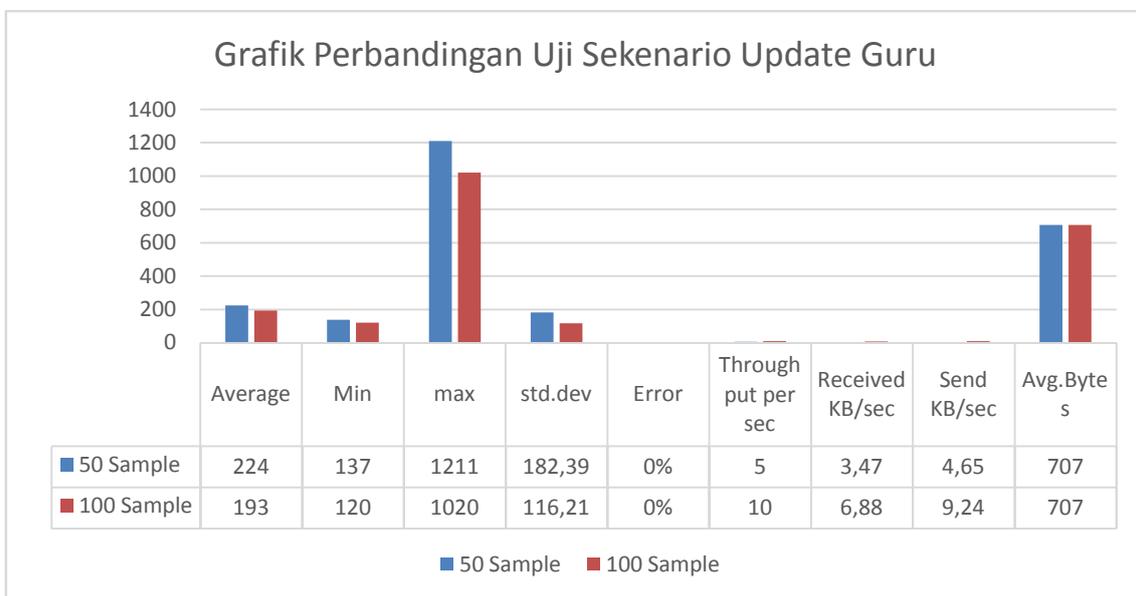
Label	Sample	Success	Fail	Min	Max	Avg	StdDev	Throughput	Bytes Received	Bytes Sent	Errors
http://localhost:8080/	50	100%	0%	137	1211	224	182.39	5.0/sec	3.47	4.65	0
Test	50	100%	0%	137	1211	224	182.39	5.0/sec	3.47	4.65	0
Summary	50	100%	0%	137	1211	224	182.39	5.0/sec	3.47	4.65	0

Gambar 9. Hasil pengujian 50 sampel pada ApacheJMeter

Hasil dari pengujian skenario testing dengan 100 sampel sebagai nampak pada gambar 10. Waktu minimal untuk dapat mengeksekusi adalah 120 ms, dan waktu terlama adalah 1020 ms. Rata-rata waktu yang diperlukan untuk mengeksekusi adalah 193 ms. Standar deviasi 116.2, dengan jumlah error 0%, rata-rata throughput 10.0/sec, rata-rata penerimaan data 6.88 Kb/sec, rata-rata pengiriman data 9.24 Kb/sec.



Gambar 10. Hasil pengujian dengan 100 sampel pada ApacheJMeter



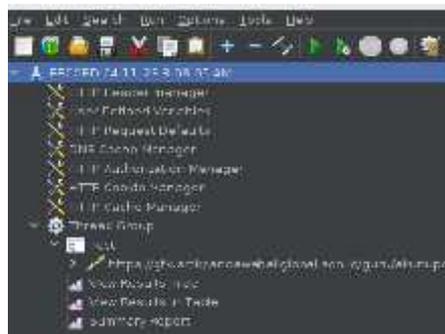
Gambar 11. GrafikPerbandingan Uji Skenario Update Guru

3.1.3 Skenario Upload Foto

Skenario halaman upload foto adalah untuk memantau proses upload foto profile guru. Proses perekaman halaman ini sebagaimana terdapat pada gambar 12.



Gambar 12. Proses perekaman halaman upload foto menggunakan blazemeter



Gambar 13. Skenario testing upload foto pada Apache JMeter

Selanjutnya skenario testing upload foto pada Apache JMeter, sebagaimana nampak pada gambar 13. Skenario testing ini menggunakan 50 dan 100 sampel dengan periode rampup adalah 10 detik dan loopcount adalah 1, hasil dari pengujian skenario testing terhadap 50 sebagai berikut.

Unit	Sample	Min	Max	Avg	Std Dev	Err %	Thru	Data	Recv	Send	Latency	Response
Sample	50	110	1091	371	314.15	0%	23.8	16.44	15.72			

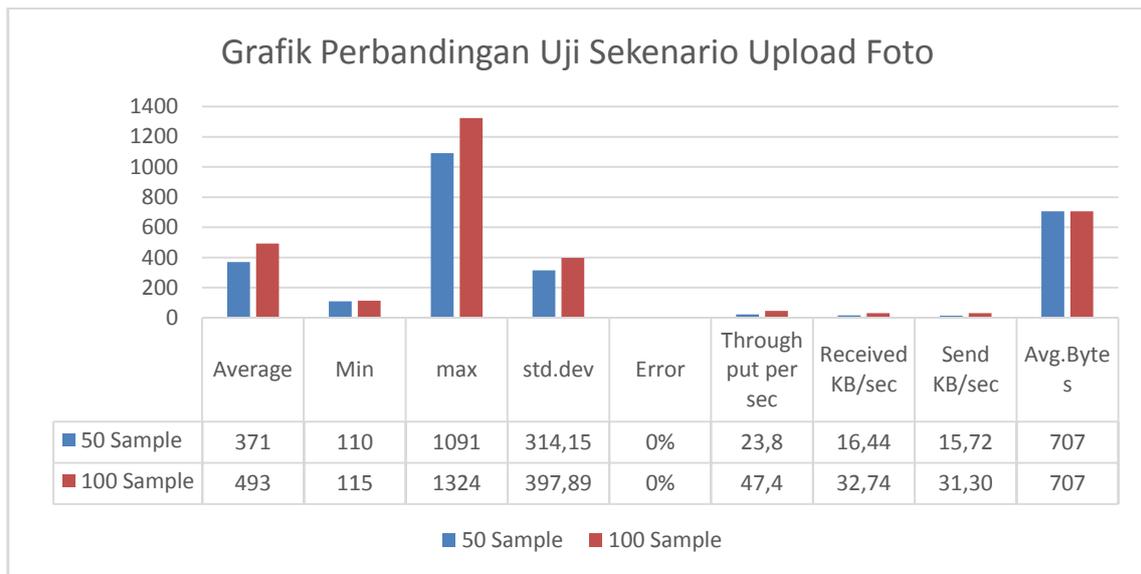
Gambar 14. Hasil pengujian 50 sampel pada Apache JMeter

Hasil pengujian dengan menggunakan 50 sampel pada Apache JMeter, sebagaimana terdapat pada gambar 14, menyatakan bahwa waktu minimal untuk dapat mengeksekusi adalah 110 ms, dan waktuterlama adalah 1091 ms. Rata-rata waktu yang diperlukan untuk mengeksekusi adalah 371 ms. Standar deviasi 314.15, dengan jumlah error 0%, rata-rata throughput 23.8/sec, rata-rata penerimaan data 16.44Kb/sec, rata-rata pengiriman data 15.72Kb/sec.

Unit	Sample	Min	Max	Avg	Std Dev	Err %	Thru	Data	Recv	Send	Latency	Response
Sample	100	110	1091	371	314.15	0%	23.8	16.44	15.72			

Gambar 15. Hasil pengujian 100 sampel pada Apache JMeter

Hasil pengujian 100 sampel pada ApacheJMeter, sebagaimana terdapat pada gambar 15. Waktu minimal untuk dapat mengeksekusi adalah 115ms, dan waktuterlama adalah 1324ms. Rata-rata waktu yang diperlukan untuk mengeksekusi adalah 493ms. Standar deviasi397.89, dengan jumlah error 0%, rata-rata throughput 47.4/sec, rata-rata penerimaan data 32.74Kb/sec, rata-rata pengiriman data 31.3Kb/sec.



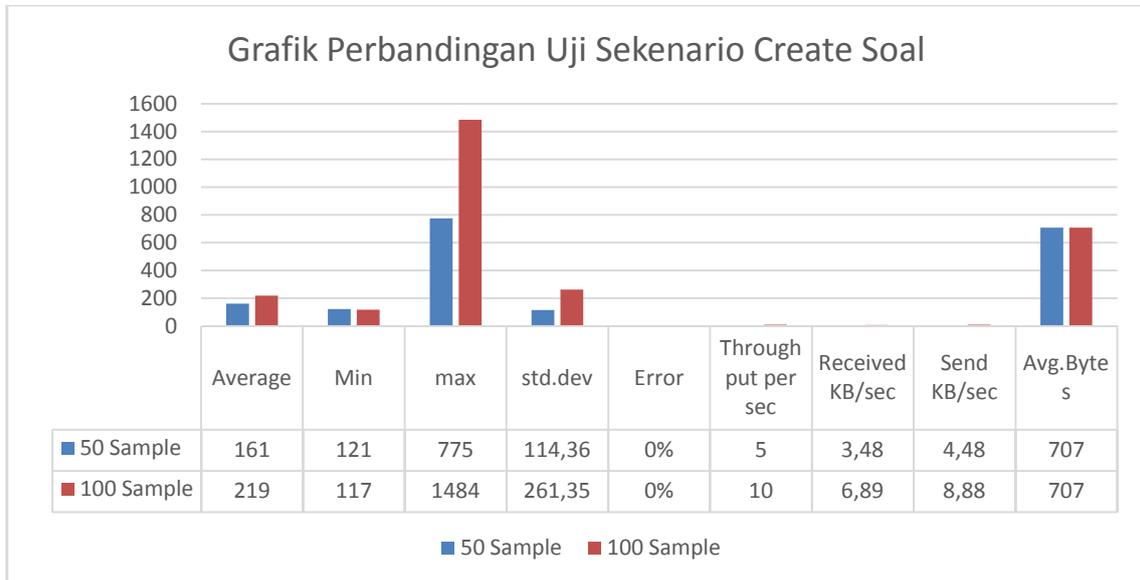
Gambar 16. Grafik Perbandingan Uji Skenario Upload Foto

3.1.4 Skenario createsoal

Skenario pengujian halaman create soal adalah untuk memantau proses createsoal. Proses perekaman halaman ini sebagaimana terdapat pada gambar 17.



Gambar 17. Proses perekaman halaman create soal menggunakan blazemeter



Gambar 21. Grafik Perbandingan Uji Skenario Create Soal

Analisis Hasil Pengujian

Hasil pengujian menunjukkan rata-rata waktu proses uji untuk 50 sample adalah 299 ms, sedangkan untuk 100 sample adalah 329 ms. Rata-rata *standard deviation* untuk uji 50 sample adalah 214.37 sedangkan untuk uji 100 sample adalah 306.997. Rata-rata *throughput* untuk uji 50 sample adalah 5.6875 per second, sedangkan pada uji 100 sample rata-ratanya sebesar 19.275 per second. Rata-rata data yang diterima (*Received*) untuk uji 50 sample adalah 6.69625 KB/sec, sedangkan pada 100 sample didapatkan 13.2925 KB/sec. Rata-rata data yang bisa dikirimkan (*Send*) untuk uji 50 sample adalah 7.73 KB/sec, sedangkan untuk uji 100 sample adalah 15.3334 KB/sec.

4. KESIMPULAN

Pengujian yang dilakukan pada penelitian ini yaitu loadtesting sistem informasi guru dengan menggunakan aplikasi Apache JMeter dan Blazemeter, loadtesting yang dilakukan dengan menguji 4 halaman website yaitu, login, update profile guru, update foto guru, buat soal. Load testing menggunakan 50 dan 100 sampel dengan ramp up period 10 detik dan loop count adalah 1, hasil yang didapat cukup beragam, pada bagian rata-rata waktu pengujian (average) mendapatkan nilai yang konstan, pada bagian throughput mendapatkan nilai yang meningkat sebanding dengan jumlah request sampelnya, dan pada bagian deviation mendapatkan nilai yang menurun sehingga ini membuktikan bahwa sistem mempunyai ketahanan yang baik dimana hasil grafik nilai average, throughput, dan deviation mendapat nilai yang baik.

DAFTAR PUSTAKA

- [1] Rainer, R. K., Prince, B., & Cegielski, C. G. (2018). *Introduction to Information Systems: Supporting and Transforming Business* (6th ed.). Wiley.
- [2] A. M. Nadeem, N. T. Islam, and A. Ali, "A Systematic Literature Review on Performance Testing of Web Applications," 2020 IEEE 4th International Conference on Information Systems and Computer Networks (ISCON), 2020, pp. 54-59.
- [3] N. Gupta, A. Gupta, and S. Tiwari, "Performance Evaluation of Web Applications: A Comprehensive Review," 2019 IEEE 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 1105-1110.
- [4] P. Jain and G. Singh, "Performance Testing Methodologies for Web Applications: A Review," 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), 2018, pp. 478-483.
- [5] S. Chandra and R. Maheshwari, "Performance Testing of Web Applications Using Apache JMeter," 2018 3rd International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 838-843.
- [6] A. Singh and S. K. Singh, "Performance Testing of Web Applications: An Overview," 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence, 2017, pp. 266-270.
- [7] M. M. Ghatol and S. T. Rao, "Performance Testing of Web Applications: A Review," 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2015, pp. 1-6.
- [8] J. A. Thampi, M. R. N. Moorthy, and V. M. Nistala, "Performance Testing Methodologies for Web Applications," 2014 IEEE International Advance Computing Conference (IACC), 2014, pp. 784-789.