

ANALISIS PERBANDINGAN KINERJA ALAT PENGUJIAN BEBAN PERANGKAT LUNAK: APACHE JMETER, GATLING, DAN K6

Qodhi Musyafa'ur Rosul¹, Endang Sulistiyani²

^{1,2} Jurusan Sistem Informasi, Universitas Nahdlatul Ulama Surabaya, Indonesia

¹3130021018@student.unusa.ac.id, ²sulistiyani.endang@unusa.ac.id

Abstrak

Dalam pengembangan perangkat lunak, software load testing memiliki peran sangat penting dalam meningkatnya kinerja suatu sistem. Pemilihan software testing yang tepat akan sangat mempengaruhi kualitas hasil pengujian. Penelitian ini menganalisis perbandingan kinerja tiga software load testing, yaitu apache jmeter, gatling, dan k6, dalam menguji performa load testing 10 website PMB universitas di surabaya. Tujuan utama penelitian adalah mengevaluasi kinerja alat pengujian berdasarkan tiga parameter. Metodologi penelitian ini terdiri dari beberapa tahapan, yaitu perencanaan lingkungan pengujian, implementasi yang mencakup pembuatan skenario uji dan pelaksanaan pengujian, serta menganalisis dan mengevaluasi hasil pengujian. Hasil perbandingan kinerja tiga tools load testing, Dari segi parameter execution time, Apache Jmeter menunjukkan durasi pengujian yang paling cepat dan konsisten di semua tingkat Virtual user. Gatling memiliki durasi pengujian yang jauh lebih lama empat kali lipat dibandingkan Apache Jmeter dan K6. Sementara K6 Menunjukkan durasi pengujian yang mirip dengan Apache Jmeter pada 50 dan 75 VU, tetapi sedikit meningkat pada 100 VU. Dari segi *ease of use*, apache jmeter memperoleh total nilai sebesar 14. masing-masing sebesar 5 poin. Sementara itu, gatling memperoleh nilai tertinggi dengan total 15 poin yang unggul dari seluruh aspek *ease of use*. Sebaliknya, k6 mendapatkan nilai terendah dengan total 10 poin, dengan rata-rata nilai yang lebih rendah dibandingkan kedua tools lainnya. Segi *feature*, apache jmeter memperoleh skor 11, gatling memperoleh skor tertinggi dengan 12 poin, dan k6 juga mencatat total 13 poin, menjadikannya tools dengan nilai tertinggi secara keseluruhan.

Kata kunci: *Software Load Testing, Website_PMB, Apache_Jmeter, Gatling, K6*

1. Pendahuluan

Dalam pengembangan perangkat lunak, peran software *load testing* menjadi sangat penting dalam meningkatnya kinerja suatu sistem. Pengujian *load testing* memiliki peran penting dengan melibatkan penilaian kinerja aplikasi di berbagai tingkat beban untuk mengukur ketahanan, skalabilitas, dan respons aplikasi. (Diantono Abda'u et al., 2024). Pengujian *load testing* salah satu jenis pengujian performansi yang memeriksa bagaimana sistem berfungsi dengan banyaknya user secara bersamaan selama periode waktu tertentu dalam melakukan interaksi pada system (Barus & Siburian, 2019) *Load testing* melibatkan simulasi sejumlah besar pengguna yang mengakses aplikasi secara simultan. Dengan melakukan *load testing* dapat mengukur kapasitas maksimum *website*, mengidentifikasi *bottleneck* atau hambatan kinerja, dan memastikan bahwa aplikasi dapat memberikan pengalaman pengguna yang optimal bahkan dalam kondisi beban yang bervariasi. (Dwinur Andrianto & Fatrianto Suyatno, 2024)

Sistem Penerimaan Mahasiswa Baru (PMB) secara umum merupakan sebuah platform *website* yang kompleks dan penting bagi sebuah institusi pendidikan. Pada periode pendaftaran, sistem PMB seringkali mengalami lonjakan trafik yang signifikan akibat banyaknya calon mahasiswa yang mengakses dan melakukan pendaftaran secara bersamaan. Hal ini

menjadikan sistem PMB dijadikan sebagai objek untuk menguji kemampuan load suatu *website* dalam menangani beban yang tinggi. (Agus Sukmandhani et al., n.d.) Namun pada kenyataannya, pengujian *load testing* belum tentu langsung dapat meningkatkan kinerja suatu sistem. Masalah kinerja alat pengujian yang kurang optimal dikarenakan kompleksitas sistem modern yang terus meningkat, ditambah dengan ekspektasi pengguna yang semakin tinggi (Islam Al Makassar et al., 2022). Masih belum banyak evaluasi yang mengintegrasikan metrik penilaian untuk menghasilkan perbandingan yang objektif. Kurangnya perencanaan yang matang dan analisis data yang mendalam dapat menyebabkan kesalahan dalam pengambilan keputusan terkait optimasi kinerja aplikasi (Nimas Maharani et al., 2023). Selain itu, terbatasnya alat dan infrastruktur pengujian juga menjadi kendala signifikan. Beragamnya alat pengujian *load testing* di pasaran menuntut evaluasi yang cermat untuk memilih alat yang paling sesuai. Penggunaan alat yang tidak memadai atau usang dapat menghasilkan data yang tidak reliabel dan berdampak negatif pada kinerja aplikasi.

Sejumlah penelitian yang telah membahas mengenai pengujian *load testing* antara lain adalah penelitian seperti (Diantono Abda'u et al., 2024). Pada penelitian ini membahas mengenai bagaimana

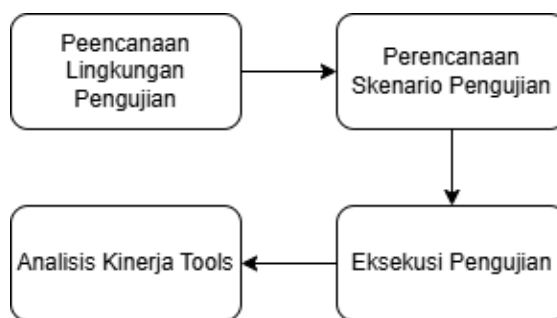
kinerja pada *restful api* yang akan dibandingkan dengan kinerja web untuk menemukan hasil kinerja yang terbaik. Penelitian lain (Barus & Siburian, 2019) yang melakukan perbandingan pengujian di beberapa *tools* tetapi penelitian ini hanya membahas dan fokus terhadap objeknya saja untuk dilakukan evaluasi. Sejumlah penelitian dan studi kasus telah dilakukan untuk membandingkan kinerja beberapa alat pengujian. Penelitian-penelitian ini (Agus Sukmandhani et al., n.d.) dan (Nurazizah et al., 2024) umumnya berfokus pada aspek-aspek seperti waktu respons, throughput, kemudahan penggunaan, dan *scalability* (Lenka et al. 2018) Penelitian (Abbas & Sultan, 2017). melakukan perbandingan beberapa *tools* pengujian dengan berbagai parameter dengan fokus hasil skala tiga poin yaitu baik, rata-rata, dan buruk. Studi lain juga melakukan perbandingan alat pengujian tanpa memberikan analisis komparatif yang mendalam. (Chandrasekhar & Chandran, 2021). Beberapa penelitian juga telah membandingkan alat dalam konteks skenario pengujian tertentu, seperti pengujian api atau pengujian aplikasi web yang kompleks (Diantono Abda'u et al., 2024)

Meskipun telah ada banyak penelitian, masih terdapat beberapa masalah yang belum terselesaikan secara menyeluruh dalam berbagai hasil yakni fokus yang terbatas, perkembangan teknologi yang cepat, dan kurangnya perbandingan secara mendalam sehingga pengguna tidak memiliki pemahaman yang jelas mengenai mekanisme kerja alat-alat tersebut. (Shrivastava, 2020) Banyak studi yang telah dilakukan hanya membandingkan dua alat secara terbatas, sehingga hasil penelitian cenderung kurang spesifik. Studi perbandingan alat dalam banyak penelitian seringkali terfokus pada implementasi objeknya. (Andriansyah, 2019). Hal ini menyebabkan kurangnya pemahaman komprehensif mengenai kelebihan dan kekurangan berbagai alat pengujian yang ada. (Srivastava, 2021)

Perbandingan berbagai macam *tools* software *load testing* dengan kesamaan fungsi dan fitur yang berbeda menjadikan perbandingan yang relevan. (Paz & Bernardino, 2017) dengan melakukan perbandingan alat pengujian yang sesuai, developer dapat meningkatkan efisiensi, mengurangi risiko, dan memastikan bahwa produk yang dihasilkan dapat memenuhi ekspektasi pengguna. Pemilihan alat yang tepat akan sangat mempengaruhi kualitas hasil pengujian, efisiensi proses, dan secara tidak langsung berdampak pada kualitas produk (Barus & Siburian, 2019). Dalam penelitian ini penulis membandingkan beberapa alat yang paling populer.

2. Metode Penelitian

Penelitian ini dilakukan melalui beberapa tahapan penting. Setiap tahapan memiliki peran penting dalam menjamin keberhasilan proses pengujian performa *tools*. Alur tahapan tersebut dapat dilihat pada Gambar 1.



Gambar 1 Metode Penelitian

2.1 Perencanaan Lingkungan Pengujian

Tahap awal yakni perencanaan lingkungan pengujian yang bertujuan untuk menentukan kebutuhan yang dibutuhkan selama proses pengujian. Hal ini mencakup spesifikasi perangkat keras/komputer, objek pengujian, *tools* pengujian, dan parameter kinerja *tools* yang akan diuji.

2.2 Perencanaan Skenario Uji

Setelah perencanaan lingkungan, langkah selanjutnya adalah menyusun skenario pengujian yang akan dilakukan. Pada skenario ini, masing-masing *tools* mensimulasikan jumlah user yang ditentukan agar dapat mengakses web secara bersamaan. Jumlah *virtual user* yang diuji dimulai dari 50, 75, dan 100 user. Setiap skenario pengujian dilakukan satu kali untuk masing-masing jumlah *virtual user*.

2.3 Eksekusi Pengujian

Selanjutnya eksekusi pengujian, tahapan ini merupakan implementasi langsung dari skenario yang telah dirancang sebelumnya. *Tools* pengujian akan menjalankan skrip yang telah disiapkan dan mensimulasikan beban pada *website* PMB.

2.4 Analisis Kinerja Tools

Data yang diperoleh dari hasil pengujian akan dianalisis untuk mengetahui performa masing-masing *tools*. Analisis ini dilakukan dengan menilai kinerja masing-masing *tools* berdasarkan parameter *execution time*, *ease of use*, dan *feature*. (Bath et al., 2018)

Parameter *execution time* dinilai berdasarkan *tools* tercepat dalam melakukan pengujian. Dari *ease of use* dan *feature* dinilai berdasarkan nilai dari skala 3, 2, 1 dengan kategori best, average, and worst respectively. (Abbas & Sultan, 2017)

3. Hasil dan Pembahasan

3.1 Perencanaan Lingkungan Pengujian

Dalam menentukan perangkat pengujian yang memadai untuk pelaksanaan pengujian ini, Berikut adalah spesifikasi perangkat pengujian yang akan digunakan dalam pelaksanaan penelitian ini, yaitu;

- *Processor*: AMD A9-9420 *RADEON R5*, 5 *COMPUTER CORES 2C+3G 3.00 GHz*
- *RAM*: 4 GB
- *SSD*: 128 GB *Hard Disk Size*: 1 TB
- *Sistem Operasi Windows 10 Pro 64-bit*
- *Java version 19.0.1*

Pengujian ini dilakukan menggunakan bantuan tools *Apache Jmeter*, *Gatling*, *Dan K6*. Pengujian ini merupakan pengujian performa yang menggunakan skema *load testing*. Objek yang akan digunakan dalam pengujian ini yakni website PMB 10 universitas di Surabaya. Pemilihan sepuluh website PMB universitas dalam penelitian ini didasarkan pada pemilihan universitas terbaik di Surabaya yang bersumber dari edurank, yang merupakan lembaga pemeringkatan universitas berbasis metrik independen yang melibatkan 14.131 universitas dari 183 negara.

Adapun parameter kinerja tools yang digunakan pada pengujian ini ialah, *Execution time*, *ease of use*, *feature*. Parameter *Execution time* digunakan untuk mengukur durasi waktu yang dibutuhkan oleh masing-masing alat pengujian dalam mengeksekusi pengujian pada sepuluh *website* PMB. Parameter *Ease of use* yang menilai tingkat kemudahan dalam penggunaan suatu tools pengujian. Penilaian dilakukan berdasarkan tiga aspek utama, yaitu

1. Kemudahan instalasi (PI)
2. Kemudahan eksekusi pengujian (PE)
3. Kemudahan membaca hasil laporan (PH)

Dimana setiap aspek penilaian ada 5 poin penilaian seperti yang tertera pada tabel 1

Tabel 1. Poin Penilaian

Easy of use	Poin Penilaian
Kemudahan Instalasi	PI1 (File instalasi mudah ditemukan & diunduh dari website resmi <i>tools</i>) PI2 (Ukuran file <50mb) PI3 (Tidak memerlukan perangkat lunak eksternal.) PI4 (Sangat cepat & mudah, saat menginstal hanya beberapa klik/sentuhan) PI5 (Berjalan lancar tanpa masalah/ <i>Error</i>)

Easy of use	Poin Penilaian
Kemudahan Eksekusi Pengujian	PE1 (Antarmuka pengguna mudah dinavigasi.) PE2 (Fungsi-fungsi mudah ditemukan dan diakses.) PE3 (Alur pengujian logis dan efisien) PE4 (Sistem responsif dan tidak ada gangguan) PE5 (Terdapat panduan singkat yang membantu.)
Kemudahan Membaca Hasil Dokumentasi Laporan Pengujian.	PH1 (Laporan mudah dibaca dan dipahami.) PH2 (Informasi penting (metrik) disajikan dengan jelas.) PH3 (Mudah untuk mengeksport laporan ke format umum misalnya, CSV, PDF.) PH4 (kustomisasi laporan tersedia dan mudah digunakan.) PH5 (Visualisasi data (grafik, tabel) jelas dan informatif.)

Parameter *Feature* ini menilai kualitas fitur-fitur standar yang tersedia pada masing-masing *tools* pengujian. Penilaian dilakukan dengan menilai kualitas fitur, yaitu seberapa baik fitur-fitur tersebut berfungsi dalam pengujian. Fitur acuan yang ada dalam masing-masing

1. Time Tracking
2. Result Reporting
3. Generating scripts
4. Load generators
5. Load Management Console

3.2 Perencanaan Skenario Uji

Pengujian ini dilakukan menggunakan masing-masing *tools* untuk mendapatkan nilai *execution time*, *ease of use*, dan *feature*. Dengan skenario pengujian masing-masing *tools* yang disajikan pada tabel 2.

Tabel 2 Rancangan Skenario Pengujian

Parameter Pengujian	Rancangan Skenario
Execution time	Rancangan pengujian ini dilakukan terhadap 10 <i>website</i> PMB dengan menggunakan tiga <i>tools</i> dengan menghitung durasi waktu eksekusi selama proses pengujian berlangsung. Setiap tool diuji dengan tiga variasi jumlah <i>Virtual user</i> (VU), yaitu 50, 75, dan 100 VU. Hasil durasi pengujian dari masing-masing variasi VU akan dihitung rata-ratanya untuk tiap <i>tools</i> , untuk mengetahui tool mana yang memiliki waktu eksekusi terbaik dalam menangani beban pengguna secara bersamaan.
Ease of use	Rancangan pengujian pada parameter ini adalah dengan menilai tingkat <i>ease of use</i> dari tiga <i>tools</i> berdasarkan tiga aspek utama, yakni kemudahan instalasi, kemudahan eksekusi pengujian, dan kemudahan dalam membaca hasil laporan. Setiap aspek terdiri dari lima poin penilaian yang dievaluasi berdasarkan kesesuaian kemudahan saat proses pengujian dilakukan. Total skor dari ketiga aspek tersebut akan dijumlahkan untuk menentukan tingkat keseluruhan

Parameter Pengujian	Rancangan Skenario
Features	kemudahan penggunaan dari masing-masing <i>tools</i> Rancangan pengujian pada parameter ini adalah menilai kualitas dari lima fitur standar pada <i>tools</i> pengujian performa. Penilaian dilakukan menilai kualitas dari fitur-fitur tersebut berdasarkan perbedaan fitur yang dimiliki oleh masing-masing <i>tools</i> terhadap standar fitur yang ditentukan. Total skor dari lima fitur standar akan dijumlahkan untuk memberikan gambaran keseluruhan terkait kelengkapan dan kualitas fitur yang dimiliki oleh masing-masing <i>tools</i>

3.3 Eksekusi Pengujian

Untuk mengetahui performa *tools* dalam menangani beban, dilakukan pengujian menggunakan apache jmeter, gatling dan k6 berdasarkan parameter yang dipilih yakni *execution time*, *ease of use*, dan *feature*.

a. Execution time

Tabel 3 merupakan hasil *execution time* dari pengujian menggunakan jmeter yang dilakukan dengan mensimulasikan jumlah pengguna *virtual* (VU) secara bertahap, mulai dari 50, 75, hingga 100.

Tabel 3. Hasil *Execution time* (Jmeter)

Website	Durasi Waktu (Detik)		
	50 VU	75 VU	100 VU
Website PMB 1	17	16	20
Website PMB 2	10	10	10
Website PMB 3	10	10	10
Website PMB 4	11	10	10
Website PMB 5	10	10	10
Website PMB 6	10	10	10
Website PMB 7	13	10	9
Website PMB 8	17	9	10
Website PMB 9	10	10	11
Website PMB 10	10	10	10

Tabel 4 merupakan hasil *execution time* dari pengujian menggunakan gatling yang dilakukan dengan mensimulasikan jumlah pengguna *virtual* (VU) secara bertahap, mulai dari 50, 75, hingga 100.

Tabel 4. Hasil *Execution time* (Gatling)

Website	Durasi Waktu (Detik)		
	50 VU	75 VU	100 VU
Website PMB 1	82	85	86
Website PMB 2	69	73	71
Website PMB 3	68	69	68
Website PMB 4	70	73	71
Website PMB 5	65	67	72
Website PMB 6	74	71	69
Website PMB 7	70	70	70

Website	Durasi Waktu (Detik)		
	50 VU	75 VU	100 VU
Website PMB 8	64	69	67
Website PMB 9	72	74	75
Website PMB 10	71	71	72

Tabel 5 merupakan hasil *execution time* dari pengujian menggunakan k6 yang dilakukan dengan mensimulasikan jumlah pengguna *virtual* (VU) secara bertahap, mulai dari 50, 75, hingga 100.

Tabel 5. Hasil Hasil *Execution time* (K6)

Website	Durasi Waktu (Detik)		
	50 VU	75 VU	100 VU
Website PMB 1	24,4	23,1	29
Website PMB 2	11,3	12,4	13,7
Website PMB 3	11,4	12	13,5
Website PMB 4	11,3	11,3	12,7
Website PMB 5	11,1	11	11
Website PMB 6	11,6	11,6	11,6
Website PMB 7	11	10,6	10,7
Website PMB 8	10,6	10,8	11,1
Website PMB 9	11,3	11,3	15,8
Website PMB 10	10,8	11	18,5

b. Ease of use

Tabel 6, 7, 8 merupakan hasil *ease of use* dari pengujian menggunakan tools jmeter gatling, dan k6 yang dinilai berdasarkan poin penilaian user selama pengujian berlangsung.

Tabel 6. Hasil PI *Ease of use*

Poin Penilaian	Jmeter	Gatling	K6
PI1	v	v	v
PI2	v	v	v
PI3	x	v	v
PI4	v	v	v
PI5	v	v	v
Nilai	4	5	5

Tabel 7. Hasil PE *Ease of use* (Jmeter)

Poin Penilaian	Jmeter	Gatling	K6
PE1	v	v	x
PE2	v	v	x
PE3	v	v	v
PE4	v	v	v
PE5	v	v	v
Nilai	5	5	3

Tabel 8. Hasil PH *Ease of use (Jmeter)*

Poin Penilaian	Jmeter	Gatling	K6
PH1	v	v	v
PH2	v	v	v
PH3	v	v	x
PH4	v	v	x
PH5	v	v	v
Nilai	5	5	3

c. Feature

Tabel 15 menyajikan hasil pengujian kualitas fitur yang dievaluasi menggunakan parameter dari tools Apache Jmeter, Gatling, dan k6. Penilaian kualitas fitur ini dinilai berdasarkan eksekusi selama pengujian berlangsung

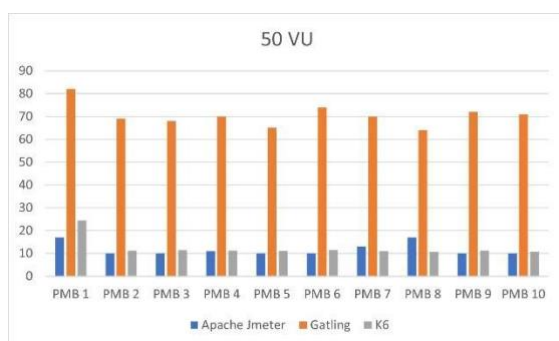
Tabel 15. Hasil *Feature*

Poin Penilaian	Apache Jmeter	Gatling	K6
F1	2	3	2
F2	2	1	3
F3	1	3	2
F4	3	2	3
F5	3	3	3
Total nilai	11	12	13

3.4 Analisis Hasil Pengujian

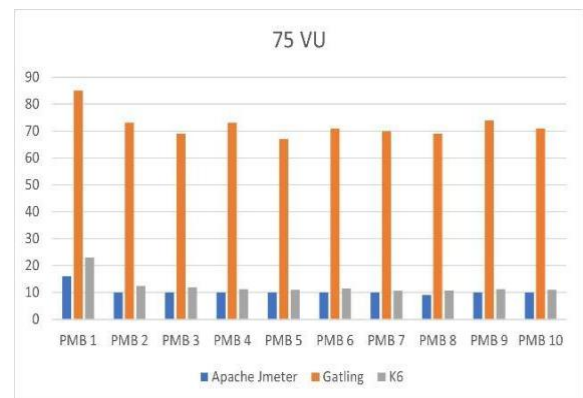
a. Execution time

Berdasarkan hasil pengujian *execution time* dari tiga tools yang disajikan pada gambar 2 menunjukkan bahwa Apache jmeter menunjukkan durasi waktu yang lebih cepat dalam proses pengujian diikuti oleh gatling. Hasil *execution time* yang diperoleh oleh tools ini dalam pengujian ini selaras dengan hasil yang ada dalam literatur terkait yang ditulis oleh. (Chandrasekhar & Chandran, 2021 dan (Shrivastava, 2020) di mana tools ini secara konsisten menunjukkan hasil yang terbaik. Pada penelitian lain (Diantono Abda'u et al., 2024) menunjukkan performa yang sebanding dalam melakukan pengujian pada sebuah sistem.



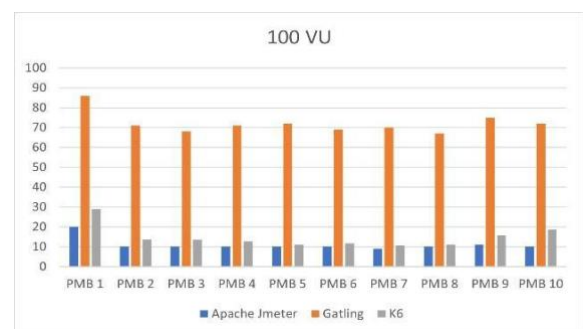
Gambar 2 Execution time 50 VU

Pada tahap awal pengujian dengan simulasi beban 50 VU, hasil analisis *execution time* menunjukkan hasil antara tiga alat yang diuji. Apache Jmeter dan K6 menunjukkan durasi waktu yang tidak jauh berbeda, di mana durasi rata-rata yang tercatat masing-masing hanya 11,8 detik dan 12,7 detik. Sebaliknya tools Gatling mencatat waktu yang jauh lebih lama, dengan rata-rata durasi mencapai 70,5 detik, enam kali lebih lambat dari dua tools lainnya.



Gambar 3 Execution time 75 VU

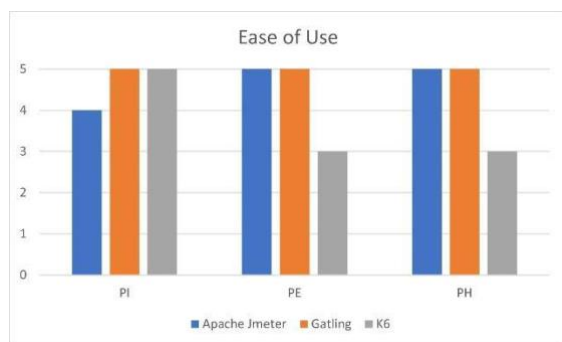
Pada tahap pengujian dengan simulasi beban 75 VU, hasil analisis *execution time* (waktu eksekusi) menunjukkan hasil yang relatif sama seperti pada pengujian di 50 VU. Apache Jmeter dan K6 kembali menunjukkan durasi waktu yang lebih cepat, di mana durasi rata-rata yang tercatat masing-masing hanya 10,5 detik dan 12,7 detik. Kinerja ini membuktikan stabilitas kedua alat meskipun terjadi peningkatan beban dari 50 VU. Sebaliknya, tools Gatling secara konsisten mencatat waktu yang jauh lebih lama, dengan rata-rata durasi mencapai 72,2 detik.



Gambar 4 Execution time 100 VU

Pada pengujian dengan simulasi beban 100 VU, hasil analisis *execution time* menunjukkan bahwa Apache Jmeter mempertahankan kinerja yang paling optimal, dengan durasi rata-rata yang tercatat hanya 11 detik. Di sisi lain, tools K6 mulai menunjukkan dampak dari seiring meningkatnya simulasi beban, di mana durasi rata-ratanya mengalami peningkatan menjadi 15,2 detik, meskipun masih tergolong cepat. Sementara itu, Gatling secara konsisten mencatat waktu yang jauh lebih lama, dengan rata-rata durasi mencapai 72,1 detik.

b. Ease of Use



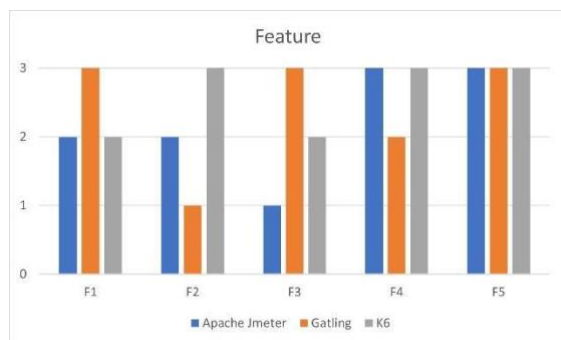
Gambar 5 Ease of use

Berdasarkan hasil analisis terhadap parameter ease of use yang disajikan pada gambar 5, menunjukkan bahwa gatling menunjukkan tingkat kemudahan penggunaan tertinggi dengan mendapatkan total nilai 15 poin di antara ketiga tools pengujian beban yang dievaluasi, diikuti oleh apache jmeter 14 poin, dan terakhir k6 10 poin.

Gatling berhasil unggul secara konsisten dalam semua aspek penilaian, mencakup kemudahan instalasi, kemudahan eksekusi, serta kemudahan dalam membaca laporan hasil pengujian. Sebanding dengan penelitian (Paz & Bernardino, 2017) yang menunjukkan nilai dari karakteristik tools dari segi interface, *script*, dan laporan dokumentasi pengujian. Keunggulan ini menjadikan gatling sebagai pilihan yang sangat direkomendasikan pengguna yang memprioritaskan kemudahan dari tiga aspek yaitu kemudahan instalasi, eksekusi, dan membaca laporan hasil dalam proses pengujian beban.

Selanjutnya, apache jmeter menunjukkan performa baik pada aspek kemudahan instalasi dan interpretasi hasil, yang mana hasil ini sesuai dengan penelitian (Sharma & Singh, 2018). Meskipun terdapat sedikit kompleksitas tambahan pada tahap instalasi. Pada k6, meskipun mendapatkan kemudahan dalam proses instalasi, pengguna membutuhkan tingkat keahlian yang lebih tinggi dalam aspek eksekusi dan interpretasi hasil. Hal ini disebabkan karena k6 menggunakan pendekatan berbasis kode.

c. Feature



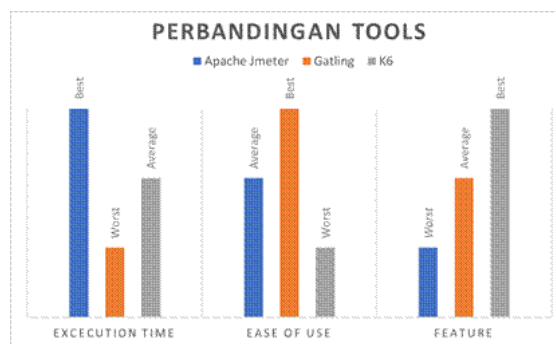
Gambar 6 Feature

Berdasarkan penilaian terhadap fitur yang disajikan pada gambar 6, penelitian ini menunjukkan bahwa K6 muncul sebagai alat dengan kualitas fitur

terbaik secara keseluruhan, memperoleh total nilai 13. K6 unggul terutama dalam kelengkapan pelaporan hasil dan efisiensi load generator, yang penting untuk analisis mendalam dan simulasi beban tinggi. Ini mengindikasikan bahwa k6 menyediakan perangkat fitur yang paling lengkap untuk pengujian load *testing*, yang mana sesuai dengan hasil penelitian oleh (Chandrasekhar & Chandran, 2021). Menunjukkan tools k6 paling baik untuk tim teknik kolaboratif dan fungsi, di mana pengujian mencakup banyak peran fitur.

Gatling menempati posisi kedua dengan total nilai 12, menunjukkan kekuatan yang signifikan dalam pelacakan waktu dan khususnya efisiensi pembuatan skrip, yang menjadikannya pilihan menarik bagi pengembang yang mengutamakan otomatisasi dalam menuliskan skrip. Namun, kelemahan utamanya adalah pada kualitas kelengkapan pelaporan hasil.

Apache jmeter berada di posisi ketiga dengan total nilai 11. Meskipun apache jmeter memiliki fitur yang baik dalam menghasilkan dan mengelola beban, serta pelacakan waktu yang memadai, jmeter menunjukkan kelemahan dalam pembuatan skrip dan kelengkapan pelaporan hasil pengujian.



Gambar 7 Perbandingan Tools

4. Kesimpulan

Secara keseluruhan, penelitian ini menyimpulkan bahwa tidak ada tools load testing yang secara mutlak terbaik dalam segala aspek. Setiap tools memiliki kelebihan dan kelemahan masing-masing. Pemilihan tools load testing yang optimal sangat bergantung pada prioritas dan kebutuhan. Jika kecepatan eksekusi (execution time) adalah yang utama, Apache Jmeter bisa menjadi pilihan yang terbaik. Apabila kemudahan penggunaan (*ease of use*) yang menjadi fokus, Gatling adalah tools terbaik dibidangnya. Sementara itu, jika fitur-fitur (*feature*) yang diutamakan, k6 menjadi pilihan tools yang terbaik.

Daftar Pustaka:

- Abbas, R., Sultan, Z., & Bhatti, S. N. (2017). Comparative Analysis of Automated Load Testing Tools: Apache JMeter, Microsoft Visual Studio (TFS), LoadRunner, Siege. In

- 2017 International Conference on Communication Technologies (ComTech) (pp. 39–44).
- Agus Sukmandhani, A., Gaesela, Y. M., & Fernando, N. (n.d.). *Testing dan Evaluasi Per for ma Website E-Commer ce Testing and Performance Evaluation of E-Commerce Web Sites*. 6(2), 34–45. <https://doi.org/10.30813/jbase.v6i1.4673>
- Andriansyah, D. (2019). Computer Based Information System Journal PERFORMANCE DAN STRESS TESTING DALAM MENGOPTIMASI WEBSITE. *CBIS JOURNAL*, 07(01). <http://ejournal.upbatam.ac.id/index.php/cbishtt> [p://ejournal.upbatam.ac.id/index.php/cbis](http://ejournal.upbatam.ac.id/index.php/cbis)
- Barus, A. C., & Siburian, L. (2019). *STUDI PERBANDINGAN ALAT PENGUJIAN OTOMATIS UNTUK APLIKASI ANDROID*. 6(6), 645–654. <https://doi.org/10.25126/jtiik.20196953>
- Bath, G., Black, R., Podelko, A., Pollner, A., & Rice, R. (2018). *Certified Tester Foundation Level Syllabus-Performance Testing*.
- Chandrasekhar, A. K., & Chandran, A. S. (2021). *COMPARATIVE ANALYSIS OF LOAD TESTING TOOLS*. <https://opensource-demo.orangehrmlive.com/index.php/dashboard>
- Costa, V., Girardon, G., Bernardino, M., MacHado, R., Legramante, G., Neto, A., Basso, F. P., & De MacEdo Rodrigues, E. (2020). Taxonomy of performance testing tools: A systematic literature review. *Proceedings of the ACM Symposium on Applied Computing*, 1997–2004. <https://doi.org/10.1145/3341105.3374006>
- Diantono Abda'u, P., Susanto, A., Supriyono, A. R., Prasetyanti, D. N., Studi, P., Informatika, T., Cilacap, P. N., & Soetomo, J. (2024). *Perbandingan Kinerja Antara Gatling dan Apache JMeter pada Uji Beban RESTful API*. 15(01). <https://doi.org/10.35970/infotekmesin.v15i1.2176>
- Dwinur Andrianto, L., & Fatrianto Suyatno, D. (2024). *Analisis Performa Load Testing Antara Mysql Dan Nosql MongoDB Pada RestAPI Nodejs Menggunakan Postman*.
- Islam Al Makassar, S. M. J., Budiman, T., & Yulianto, A. B. (2022). Rancangan Program Otomatisasi Otentikasi Pengguna Untuk Otorisasi Pada Website Dengan Python Dan Selenium Web Driver. *Jurnal Manajemen Informatika Jayakarta*, 2(4), 326. <https://doi.org/10.52362/jmijayakarta.v2i4.917>
- Lenka, R. K., Mangain, S., Kumar, S., & Barik, R. K. (2018). Performance Analysis of Automated Testing Tools: JMeter and TestComplete. In *International Conference on Advances in Computing, Communication Control and Networking (ICACCCN2018)* (pp. 399–407)
- Nimas Maharani, C., Darwis, D., Penulis, N., Dedi, K. :, & Submitted, D. (2023). *Analisis Perbandingan Kualitas Perangkat Lunak Pada Website Perguruan Tinggi Menggunakan Metode Webqual, Apache J-Meter, Dan Web Server Stress Tool*. 4(1), 34–41. <https://doi.org/10.33365/jtsi.v4i1.2436>
- Nurazizah, R. A. S., Yustiana, I., & Hermanto, H. (2024). EVALUASI WEBSITE OPEN DATA KOTA SUKABUMI BERDASARKAN STANDAR ISO/IEC 9126. *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 9(1), 79–89. <https://doi.org/10.29100/jipi.v9i1.4281>
- Paz, S., & Bernardino, J. (2017). Comparative analysis of web platform assessment tools. *WEBIST 2017 - Proceedings of the 13th International Conference on Web Information Systems and Technologies*, 116–125. <https://doi.org/10.5220/0006308101160125>
- Sharma, V., & Singh, M. (2018). *Proceedings, IEEE 2018 International Conference on Advances in Computing, Communication Control and Networking : (ICACCCN) : on 12th-13th Oct, 2018*. IEEE.
- Shrivastava, S. (2020). Comprehensive Review of Load Testing Tools. *International Research Journal of Engineering and Technology*. www.irjet.net
- Srivastava, N. (2021). Software and Performance Testing Tools. *Journal of Informatics Electrical and Electronics Engineering (JIEEE)*, 2(1), 1–12. <https://doi.org/10.54060/jieec/002.01.001>

Halaman ini sengaja dikosongkan