

Benchmark Load Testing Multi Skenario Aplikasi SiPADUKA V2 pada Shared Hosting

Muhammad Agus Triawan^{*1}, Husnawati², Ahmad Bahri Joni M³, Ridho Rahmatullah⁴, Bagus Ananta Hidayatullah⁵, Mohd. Ginar Risky⁶

^{1,2,3,4,5,6} Jurusan Teknik Komputer, Politeknik Negeri Sriwijaya

^{*1}matriawan@polsri.ac.id, ²husnawati@polsri.ac.id, ³b.joni@polsri.ac.id, ⁴rdh.rhmtllah@gmail.com,

⁵bagusananta278@gmail.com, ⁶ginarrisky1234@gmail.com

Abstract

Public service Web applications require adequate performance to ensure stable access when handling user load. SiPADUKA V2, an Information System for Reporting Child Abuse, was developed using native PHP MVC, MySQL PDO, Bootstrap, and JavaScript, and deployed on cPanel-based shared hosting. This study aims to establish the baseline performance of SiPADUKA V2 through an empirical benchmark using the load testing (LT) method. The testing was conducted using Apache JMeter 5.6.3 across four scenarios, namely LT1, LT2, LT3, and LT4, with 5, 10, 25, and 50 virtual users, respectively. The test plan consisted of three Thread Groups representing public pages, user pages after login, and admin pages after login. The analyzed metrics included Total Requests/Samples, Average Response Time, Throughput, and Error Rate. The results show that the total number of requests increased from 857 requests in LT1 to 5,180 requests in LT4, while throughput increased from 4.80 req/s to 27.68 req/s. The average response time was recorded at 383.94 ms in LT1, 342.40 ms in LT2, 289.72 ms in LT3, and increased to 589.72 ms in LT4. The error rate was 0.12% in LT1, 0% in LT2 and LT3, and increased to 4.05% in LT4. These findings indicate that SiPADUKA V2 remained relatively stable up to 25 virtual users but began to experience performance pressure at 50 virtual users.

Keywords: Apache JMeter, load testing, response time, shared hosting, SiPADUKA V2.

Abstrak

Aplikasi Web layanan publik perlu memiliki performa yang memadai agar dapat diakses secara stabil ketika menerima beban pengguna. SiPADUKA V2 sebagai aplikasi Sistem Informasi Pengaduan Kekejaman Pada Anak telah dikembangkan menggunakan PHP native MVC, MySQL PDO, Bootstrap, dan JavaScript, serta *dideploy* pada *shared hosting* berbasis cPanel. Penelitian ini bertujuan untuk memperoleh data dasar (*baseline*) performa SiPADUKA V2 melalui *benchmark* empiris menggunakan metode *load testing* (LT). Pengujian dilakukan dengan Apache JMeter 5.6.3 pada empat skenario, yaitu LT1, LT2, LT3, dan LT4 dengan variasi 5, 10, 25, dan 50 *virtual user*. *Test Plan* disusun menggunakan tiga *Thread Group* (TG) yang merepresentasikan halaman publik, halaman pengguna setelah *login*, dan halaman admin setelah *login*. Metrik yang dianalisis meliputi *Total Request/Samples*, *Average Response Time*, *Throughput*, dan *Error Rate*. Hasil pengujian menunjukkan bahwa total *request* meningkat dari 857 *request* pada LT1 menjadi 5180 *request* pada LT4, sedangkan *throughput* meningkat dari 4,80 req/s menjadi 27,68 req/s. *Average response time* tercatat sebesar 383,94 ms pada LT1, 342,40 ms pada LT2, 289,72 ms pada LT3, dan meningkat menjadi 589,72 ms pada LT4. *Error rate* berada pada 0,12% pada LT1, 0% pada LT2 dan LT3, serta meningkat menjadi 4,05% pada LT4. Hasil ini menunjukkan bahwa SiPADUKA V2 relatif stabil hingga 25 *virtual user*, namun mulai mengalami tekanan performa pada 50 *virtual user*.

Kata Kunci: Apache JMeter, load testing, response time, shared hosting, SiPADUKA V2.

1. Pendahuluan

Perkembangan layanan digital mendorong lembaga publik untuk menyediakan sistem informasi yang cepat, terstruktur, dan mudah diakses, termasuk dalam layanan perlindungan anak dan penanganan pengaduan masyarakat. Studi mengenai pemanfaatan teknologi digital menunjukkan bahwa pengelolaan pengaduan kasus kekerasan terhadap anak dapat dilakukan lebih terstruktur dan terdokumentasi melalui penerapan sistem informasi [1], [2], [3]. Selain itu, sistem pelaporan kasus berbasis Web di lingkungan pendidikan dan kajian

perlindungan anak di ruang digital menunjukkan bahwa akses pelaporan, pencatatan, pemantauan, dan respons kelembagaan merupakan aspek penting dalam memperkuat layanan perlindungan anak [4], [5]. Oleh karena itu, pengembangan sistem informasi pengaduan anak menjadi relevan sebagai bagian dari upaya mendukung mekanisme pelaporan dan pengelolaan data kasus agar lebih terstruktur. Dalam konteks ini, aplikasi berbasis web seperti SiPADUKA V2 dapat diposisikan sebagai prototipe sistem informasi pengaduan sederhana

yang mendukung mekanisme pelaporan dan pengelolaan data kekejaman pada anak oleh pemerintah pusat maupun daerah.

Namun, pengembangan aplikasi Web untuk layanan publik tidak cukup hanya dilihat dari keberhasilan fungsionalitasnya, karena aplikasi juga perlu memiliki performa yang memadai ketika menerima beban akses dari pengguna [6], [7]. Performa aplikasi Web umumnya dievaluasi melalui metrik seperti *response time*, *throughput*, dan *error rate*, karena ketiga metrik tersebut dapat menggambarkan waktu *respons* sistem, kemampuan memproses *request*, serta persentase kegagalan *request* selama pengujian [6], [8], [9]. Beberapa penelitian terdahulu menunjukkan bahwa Apache JMeter banyak digunakan untuk melakukan *load testing* aplikasi Web, baik pada Website akademik, sistem informasi sekolah, maupun sistem informasi lainnya [10], [11], [12]. Studi pada Website akademik Universitas XYZ menggunakan Apache JMeter untuk mengevaluasi performa dalam kondisi normal, *peak*, dan *stress usage* [6], [13]. Penelitian lain pada Website Sekolah juga menggunakan JMeter dan BlazeMeter untuk mengukur *response time*, *throughput*, dan kestabilan sistem pada beberapa modul aplikasi [8], [14]. Temuan-temuan tersebut menunjukkan bahwa *load testing* merupakan pendekatan yang relevan untuk memperoleh gambaran empiris terkait performa aplikasi berbasis Web.

Pengujian performa aplikasi Web menunjukkan bahwa *load testing* tidak hanya digunakan untuk mengetahui apakah sistem dapat berjalan, tetapi juga untuk mengidentifikasi batas awal penurunan performa ketika beban pengguna meningkat [6], [7], [15], [16]. Penelitian terbaru terhadap Website konferensi menunjukkan bahwa peningkatan *virtual user* dapat meningkatkan *response time* dan *error rate* setelah server melewati batas kemampuan tertentu, meskipun *throughput* masih dapat bertambah pada beban tinggi [7], [17]. Pola tersebut menunjukkan bahwa performa aplikasi tidak dapat dinilai dari satu metrik saja, karena *throughput* yang meningkat tetap perlu dibaca bersama *response time* dan *error rate* [7], [10], [18]. Dalam konteks yang lebih luas, kajian *web performance* modern menempatkan *loading*, *interactivity*, dan *stability* sebagai aspek penting dalam pengalaman pengguna. Dengan demikian, *performa web* dapat dipahami sebagai aspek multidimensi yang berkontribusi terhadap kualitas layanan digital [19], [20], [21], [22].

Meskipun penelitian sebelumnya telah banyak menguji performa Website akademik, sistem informasi sekolah, API, *e-commerce*, Website perpustakaan, dan layanan Web umum, kajian yang secara khusus membahas *benchmark empiris* aplikasi yang berkaitan dengan layanan pengaduan seperti aplikasi kekejaman pada anak pada lingkungan *shared hosting* masih terbatas [7], [8], [23], [24]. *Shared hosting* banyak digunakan sebagai lingkungan *deployment* untuk aplikasi skala kecil hingga

menengah, namun memiliki keterbatasan *resource* (beberapa Website berjalan pada satu server yang sama) dan kontrol server dibandingkan VPS atau *dedicated server* [25], [26], [27]. Selain itu, karakteristik SiPADUKA V2 melibatkan halaman publik, autentikasi pengguna, form pengaduan, dan halaman admin, sehingga pengujian perlu mempertimbangkan *request* autentikasi dan *session* sebagai bagian dari alur akses aplikasi. Oleh karena itu, diperlukan pengujian empiris untuk mengetahui *baseline* atau acuan awal performa SiPADUKA V2 pada lingkungan *shared hosting* sebelum dilakukan optimasi atau pengembangan lanjutan.

Penelitian ini bertujuan untuk melakukan *benchmark* empiris performa aplikasi Web SiPADUKA V2 pada *shared hosting* berbasis cPanel menggunakan Apache JMeter 5.6.3. Pengujian difokuskan pada *load testing* dengan variasi 5, 10, 25, dan 50 *virtual user* untuk mengukur *Total Request/Samples*, *Average Response Time*, *Throughput*, dan *Error Rate*. Halaman uji utama meliputi *Home*, *News*, *Login Form*, *Complaint Form*, *Dashboard*, dan *Data Complaint*, sedangkan *request* autentikasi ditambahkan sebagai prasyarat akses halaman terproteksi sesuai rancangan *Test Plan* JMeter yang telah disusun dalam penelitian ini. Dengan pendekatan tersebut, penelitian ini memberikan gambaran empiris tentang performa awal SiPADUKA V2 pada *shared hosting* serta menyediakan dasar evaluasi bagi pengembangan, optimasi, atau perbandingan *deployment* pada penelitian berikutnya.

2. Metode Penelitian

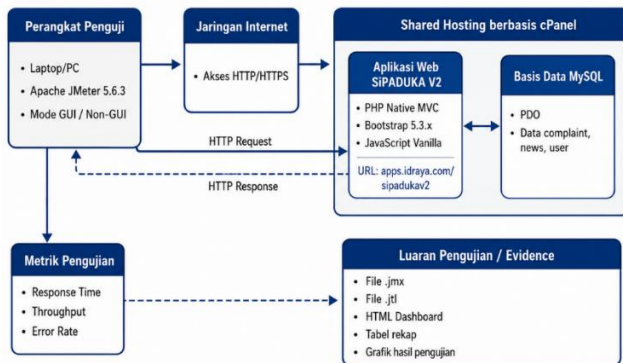
2.1 Research Object and Testing Environment

Objek penelitian ini adalah SiPADUKA V2, yaitu aplikasi web Sistem Informasi Pengaduan Kekejaman Pada Anak yang dikembangkan menggunakan PHP native dengan pola *Model-View-Controller* (MVC), basis data MySQL dengan akses PHP Data Objects (PDO), Bootstrap 5.3.3, dan JavaScript. Aplikasi ini *dideploy* pada *shared hosting* berbasis cPanel dan diakses melalui alamat <https://apps.idraya.com/sipadukav2>. Pengujian dilakukan menggunakan Apache JMeter 5.6.3 dari perangkat penguji melalui jaringan internet untuk memperoleh *baseline performa* aplikasi pada lingkungan *shared hosting*. Diagram blok sistem pengujian ditunjukkan pada Gambar 1.

Tabel 1. Environment Load Testing SiPADUKA V2

Komponen	Keterangan
Aplikasi	SiPADUKA V2
Kepanjangan	Sistem Informasi Pengaduan Kekejaman Pada Anak
Teknologi	PHP native MVC, MySQL PDO, Bootstrap 5.3.3, JavaScript
Deployment	Shared hosting berbasis cPanel
URL	https://apps.idraya.com/sipadukav2
Tools	Apache JMeter 5.6.3,
Rekap dan visualisasi	Microsoft Excel 2019
Mode pengujian utama	Non-GUI

Perangkat pengujian Laptop AMD Athlon Gold 3150U, RAM 32 GB, & Windows 11
 Koneksi internet Hotspot/WiFi



Gambar 1. Diagram Blok Sistem Pengujian Performa SiPADUKA V2 pada Shared Hosting

2.2 Apache JMeter Test Plan Design

Test Plan Apache JMeter disusun untuk merepresentasikan alur akses utama pada aplikasi SiPADUKA V2. Halaman uji utama terdiri atas enam halaman, yaitu *Home*, *News*, *Login Form*, *Complaint Form*, *Dashboard*, dan *Data Complaint*. Pemilihan halaman tersebut didasarkan pada representasi akses publik, proses autentikasi, akses pengguna, dan akses admin. Daftar halaman uji utama ditunjukkan pada Tabel 2 berikut.

Tabel 2. Page Utama Uji Load Testing SiPADUKA V2

Kode	Halaman & Method	URL	Tujuan
H1	Home/GET	https://apps.idraya.com/sipadukav2/index.php?route=home	Mengukur waktu respons halaman awal aplikasi.
H2	News/GET	https://apps.idraya.com/sipadukav2/index.php?route=news	Mengukur waktu respons halaman berita/informasi.
H3	Login Form/GET	https://apps.idraya.com/sipadukav2/index.php?route=login	Mengukur waktu respons halaman autentikasi.
H4	Complaint Form/GET	https://apps.idraya.com/sipadukav2/index.php?route=user/complaints/create	Mengukur waktu respons form pengaduan setelah login user.
H5	Dashboard/GET	https://apps.idraya.com/sipadukav2/index.php?route=admin/dashboard	Mengukur waktu respons dashboard admin.
H6	Data Complaint	https://apps.idraya.com/sipadukav2/index.php?route=admin/complaints	Mengukur waktu respons data pengaduan admin.

Berdasarkan halaman uji tersebut, Test Plan dibagi menjadi tiga Thread Group (TG). TG1 digunakan untuk halaman publik yang dapat diakses tanpa autentikasi, TG2 digunakan untuk halaman pengguna setelah login, sedangkan TG3 digunakan untuk halaman

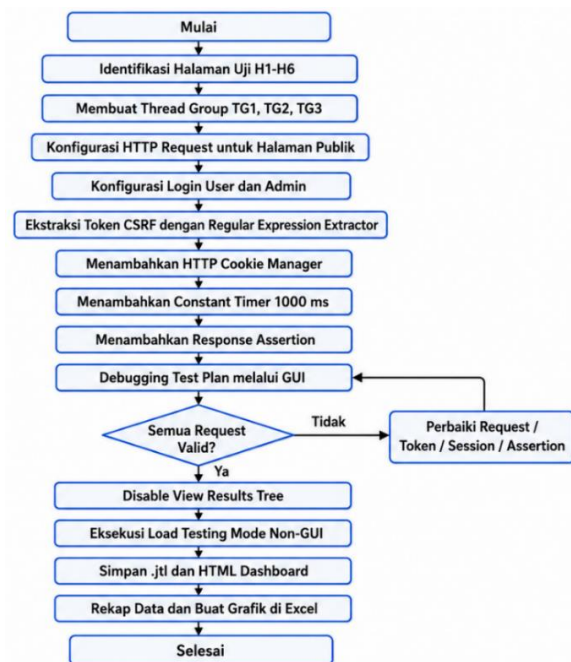
admin setelah login. Struktur Test Plan Apache JMeter ditunjukkan pada Tabel 3.

Tabel 3. Struktur Test Plan Apache JMeter

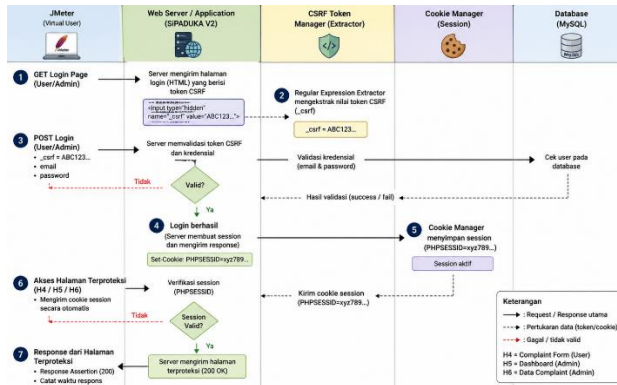
Thread Group	Fungsi	Sampler
TG1 Public Pages	Menguji halaman publik tanpa login	H1, H2, H3
TG2 User Pages	Menguji halaman pengguna setelah login	Get Login Page - User, POST Login as User, H4
TG3 Admin Pages	Menguji halaman admin setelah login	Get Login Page - Admin, POST Login as Admin, H5, H6

Pada rancangan awal, halaman utama yang dianalisis adalah H1 sampai H6. Namun, halaman H4, H5, dan H6 membutuhkan autentikasi sehingga tidak dapat diakses langsung tanpa session login. Oleh karena itu, pada TG2 dan TG3 ditambahkan request Get Login Page dan POST Login sebagai prasyarat autentikasi. Request Get Login Page digunakan untuk mengambil halaman login dan memperoleh token Cross-Site Request Forgery (CSRF), sedangkan request POST Login digunakan untuk mengirim parameter csrf, email, dan password agar session pengguna atau admin terbentuk.

Untuk mempertahankan session selama pengujian, Test Plan menggunakan HTTP Cookie Manager. Nilai token CSRF diekstraksi dari response halaman login menggunakan Regular Expression Extractor. Dengan konfigurasi tersebut, halaman yang membutuhkan autentikasi, yaitu H4, H5, dan H6, dapat diakses secara valid oleh JMeter. Alur penyusunan Test Plan dan alur autentikasi session serta token CSRF ditunjukkan pada Gambar 2 dan Gambar 3.



Gambar 2. Alur Penyusunan Test Plan Apache JMeter untuk Load Testing SiPADUKA V2



Gambar 3. Alur Autentikasi Session dan Token CSRF pada Test Plan JMeter

Dalam proses analisis, halaman uji utama direpresentasikan oleh *main sampler*, yaitu label *sampler* utama tanpa akhiran -0 atau -1. Label dengan akhiran tersebut diperlakukan sebagai *sub-sample* yang dapat muncul akibat *redirect* atau *request* turunan. Oleh karena itu, analisis detail per halaman difokuskan pada main sampler H1 sampai H6, sedangkan *request* autentikasi tetap diperhitungkan dalam statistik global karena menjadi bagian dari alur akses yang diperlukan untuk membuka halaman terproteksi.

2.3 Load Testing Scenario

Pengujian dilakukan dalam empat skenario, yaitu LT1, LT2, LT3, dan LT4 dengan total *virtual user* masing-masing 5, 10, 25, dan 50 yang ditunjukkan pada Tabel 4. *Virtual user* merepresentasikan pengguna *virtual* yang disimulasikan oleh Apache JMeter, sedangkan *Thread Group* digunakan untuk mengelompokkan pola akses halaman publik, pengguna, dan admin. *Ramp-up period* digunakan untuk menaikkan jumlah *virtual user* secara bertahap agar beban tidak diberikan secara tiba-tiba ke *server*. Setiap skenario dijalankan selama 180 detik dengan pengaturan *Loop Count: Forever/Infinite* dan *Scheduler/Specify Thread lifetime* aktif untuk mengontrol durasi pengujian. *Constant Timer* sebesar 1000 ms digunakan untuk memberikan jeda antar-*request*.

Tabel 4. Skenario Load Testing Berdasarkan Variasi VU

Kode	T.V U	TG1	TG2	TG3	Ramp-up Period	Duration
LT1	5	2	2	1	30 s	180 s
LT2	10	4	3	3	60 s	180 s
LT3	25	10	8	7	90 s	180 s
LT4	50	20	15	15	120 s	180 s

LT = Load Testing, T.VU = Total Virtual User, TG = Thread Group.

2.4 Metrics and Data Processing

Metrik yang digunakan dalam pengujian ini meliputi Total *Request/Samples*, *Average Response Time*, *Throughput*, dan *Error Rate*. Keempat metrik tersebut

diperoleh dari bagian Statistics pada HTML Dashboard Apache JMeter setelah setiap skenario *load testing* selesai dijalankan. Ringkasan metrik yang digunakan ditunjukkan pada Tabel 5.

Tabel 5. Metrik Load Testing dan Satuan Pengukuran

Metrik	Definisi	Satuan
Total Request /Samples	Jumlah <i>request</i> yang tercatat selama pengujian	<i>request</i>
Average Response Time	Rata-rata waktu <i>respons</i> aplikasi terhadap <i>request</i>	ms
Throughput	Jumlah <i>request</i> yang diproses per detik	req/s
Error Rate	Persentase <i>request</i> gagal	%

Total *Request/Samples* dihitung sebagai jumlah seluruh *request* yang tercatat pada setiap sampler selama pengujian. Secara matematis, Total *Request/Samples* dinyatakan pada Persamaan (1).

$$TR = \sum_{i=1}^n R_i \quad (1)$$

Keterangan: *TR* adalah Total *Request/Samples*, *R_i* adalah jumlah *request* pada *sampler* ke-*i*, dan *n* adalah jumlah *sampler* yang tercatat dalam pengujian.

Average Response Time digunakan untuk menghitung rata-rata waktu *respons* dari seluruh *request* yang dijalankan selama skenario pengujian. Persamaannya ditunjukkan pada Persamaan (2).

$$ART = \frac{\sum_{i=1}^N RT_i}{N} \quad (2)$$

Keterangan: *ART* adalah *Average Response Time*, *RT_i* adalah *response time* pada *request* ke-*i*, dan *N* adalah total *request/samples*.

Throughput menunjukkan jumlah *request* yang dapat diproses aplikasi per satuan waktu. Pada HTML Dashboard Apache JMeter, nilai *throughput* ditampilkan pada kolom *Transactions/s* dan dalam penelitian ini dinyatakan sebagai *request per second*. Persamaan *throughput* ditunjukkan pada Persamaan (3).

$$TP = \frac{N}{T} \quad (3)$$

Keterangan: *TP* adalah *throughput*, *N* adalah total *request/samples*, dan *T* adalah durasi pengujian dalam detik.

Error Rate digunakan untuk menghitung persentase *request* yang gagal selama pengujian. *Request* gagal dapat berupa *server error*, *timeout*, *connection refused*, atau non-HTTP *response* yang tercatat oleh JMeter. Persamaan *Error Rate* ditunjukkan pada Persamaan (4).

$$ER = \frac{F}{N} \times 100\% \quad (4)$$

Keterangan: *ER* adalah *Error Rate*, *F* adalah jumlah *request* gagal, dan *N* adalah total *request/samples*.

Sebelum pengujian utama dilakukan, *Test Plan* divalidasi menggunakan mode *Graphical User Interface* (GUI)

pada Apache JMeter. Validasi dilakukan dengan *listener View Results Tree* untuk memastikan seluruh HTTP *request, session login*, ekstraksi token *Cross-Site Request Forgery (CSRF)*, dan *Response Assertion* berjalan dengan benar. Setelah seluruh *sampler* tervalidasi, *listener View Results Tree* dinonaktifkan agar tidak menambah beban memori dan pemrosesan pada perangkat pengujian. Pengujian utama kemudian dijalankan menggunakan mode non-GUI untuk mengurangi *overhead* dari antarmuka grafis.

Setiap skenario menghasilkan file hasil mentah dalam format “.jtl” dan laporan HTML *Dashboard*. Data hasil pengujian direkap menggunakan Microsoft Excel. Ringkasan hasil pengujian diambil dari baris Total pada tabel *Statistics* JMeter untuk membandingkan performa antar skenario LT1 sampai LT4. Sementara itu, analisis detail per halaman dilakukan menggunakan *main samplers*, yaitu label *sampler* utama seperti H1 sampai H6 tanpa akhiran -0 atau -1. Label dengan akhiran tersebut tidak digunakan dalam analisis detail karena diperlakukan sebagai *sub-sample* dari *redirect* atau *request* turunan.

3. Hasil dan Pembahasan

3.1 Load Testing Execution

Pengujian utama dijalankan menggunakan Apache JMeter 5.6.3 dalam mode non-GUI setelah *Test Plan* tervalidasi melalui mode GUI. Setiap skenario menghasilkan file hasil mentah dalam format “.jtl” dan laporan HTML *Dashboard*. Data hasil pengujian kemudian diambil dari bagian *Statistics* pada HTML *Dashboard* dan direkap menggunakan Microsoft Excel. Contoh perintah eksekusi *load testing* ditunjukkan sebagai berikut.

```
jmeter -n -t
"... \testplan_sipaduka_v2_LT1_5VU.jmx
-l "... \raw_jt1\LT1_5VU.jt1" -e -o
"... \html_dashboard\LT1_5VU"
```

Parameter -n digunakan untuk menjalankan JMeter dalam mode non-GUI, -t menunjukkan lokasi file Test Plan .jmx, -l menunjukkan lokasi file hasil pengujian .jtl, -e digunakan untuk menghasilkan HTML *Dashboard*, sedangkan -o menunjukkan folder keluaran HTML *Dashboard*. Perintah yang sama digunakan untuk skenario LT2, LT3, dan LT4 dengan menyesuaikan nama file *Test Plan*, file “.jtl”, dan folder HTML *Dashboard* sesuai skenario masing-masing. Konfigurasi jumlah *virtual user*, pembagian *Thread Group*, *ramp-up period*, *duration*, *Loop Count: Forever/Infinite*, *Scheduler/Specify Thread lifetime*, serta *Constant Timer* telah diatur pada masing-masing file .jmx sesuai skenario pada Tabel 4.

3.2 Summary of Load Testing Results

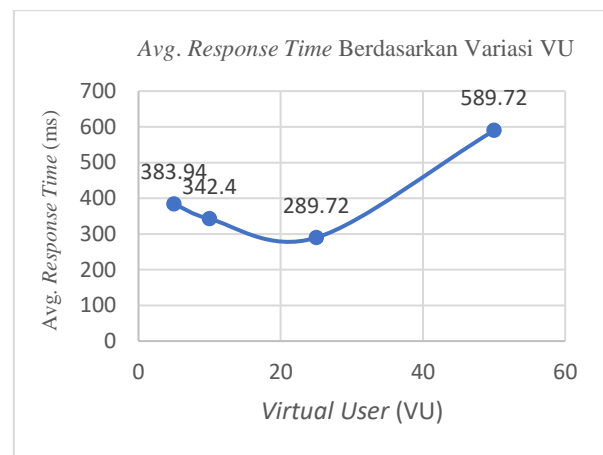
Hasil ringkasan load testing SiPADUKA V2 pada empat variasi *virtual user* ditunjukkan pada Tabel 6. Ringkasan ini diambil dari baris Total pada tabel *Statistics* HTML *Dashboard* Apache JMeter. Oleh karena itu, nilai pada Tabel 6 mencerminkan akumulasi seluruh *request* dalam *Test Plan*, termasuk *request* halaman utama H1–H6 serta *request* autentikasi yang diperlukan untuk mengakses halaman terproteksi.

Tabel 6. Ringkasan Hasil Load Testing SiPADUKA V2 Berdasarkan Variasi *Virtual User*

SK	T.VU	T.R/S	Avg. R.T. (ms)	Tp. (req/s)	E.R (%)
LT1	5	857	383,94	4,8	0,12
LT2	10	1686	342,4	9,41	0
LT3	25	3873	289,72	21,56	0
LT4	50	5180	589,72	27,68	4,05

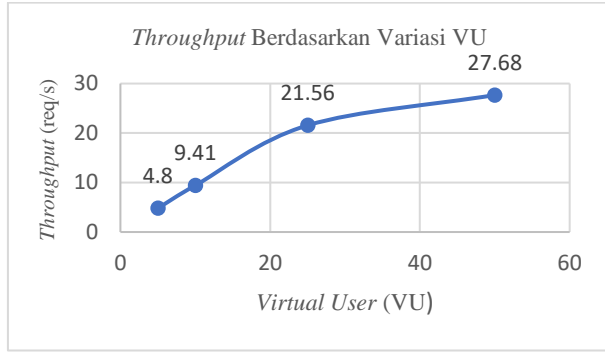
SK = Skenario, T = Total, T.R/S = Request/Samples, Avg. R.T. = Average Response Time, Tp = Throughput, E.R. = Error Rate

Berdasarkan Persamaan (1), Total *Request/Samples* merupakan akumulasi request yang tercatat selama pengujian. Tabel 6 menunjukkan bahwa total *request* meningkat dari 857 *request* pada LT1 menjadi 5180 *request* pada LT4. Peningkatan tersebut terjadi karena jumlah *virtual user* bertambah, sehingga jumlah HTTP *request* yang dikirim oleh JMeter selama durasi pengujian juga meningkat.



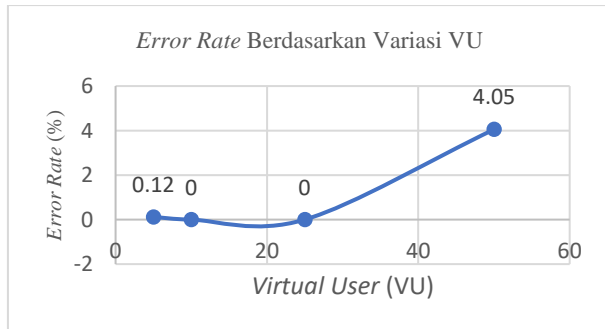
Gambar 4. Average Response Time SiPADUKA V2 pada Setiap Skenario Load Testing

Berdasarkan Persamaan (2), *Average Response Time* dihitung dari rata-rata waktu *respons* seluruh *request* pada setiap skenario. Hasil pada Tabel 6 dan Gambar 4 menunjukkan bahwa *average response time* pada LT1 sebesar 383,94 ms, menurun menjadi 342,40 ms pada LT2 dan 289,72 ms pada LT3, kemudian meningkat menjadi 589,72 ms pada LT4. Penurunan pada LT2 dan LT3 dapat dipengaruhi oleh variasi kondisi jaringan, *cache*, serta karakteristik *shared hosting*. Namun, peningkatan *response time* pada LT4 menunjukkan bahwa beban 50 *virtual user* mulai memberikan tekanan lebih besar terhadap aplikasi dan lingkungan *hosting*.



Gambar 5. Throughput pada Setiap Skenario Load Testing

Berdasarkan Persamaan (3), *Throughput* menunjukkan jumlah *request* yang dapat diproses per detik. Hasil pada Tabel 6 dan Gambar 5 menunjukkan peningkatan *throughput* secara konsisten dari 4,80 req/s pada LT1 menjadi 9,41 req/s pada LT2, 21,56 req/s pada LT3, dan 27,68 req/s pada LT4. Peningkatan *throughput* ini menunjukkan bahwa aplikasi mampu memproses lebih banyak *request* ketika jumlah *virtual user* meningkat. Namun, peningkatan *throughput* pada LT4 disertai dengan kenaikan *average response time* dan *error rate*, sehingga skenario tersebut menunjukkan adanya tekanan *performa* yang lebih tinggi dibandingkan skenario sebelumnya.



Gambar 6. Error Rate pada Setiap Skenario Load Testing

Berdasarkan Persamaan (4), *Error Rate* dihitung dari persentase *request* gagal terhadap total *request*. Hasil pada Tabel 6 dan Gambar 6 menunjukkan bahwa *Error rate* pada LT1 tercatat sebesar 0,12%, sedangkan LT2 dan LT3 tidak menghasilkan *error*. Pada LT4, *error rate* meningkat menjadi 4,05%. Dengan demikian, skenario LT1 sampai LT3 menunjukkan kondisi pengujian yang relatif stabil, sedangkan LT4 (skenario 50 *virtual user*) mulai menunjukkan sebagai kondisi awal penurunan stabilitas lingkungan *shared hosting* yang ditandai dengan peningkatan *error rate*.

3.3 Per-Page Load Testing Results

Analisis detail per halaman dilakukan menggunakan *main sampler*, yaitu label *sampler* utama H1 sampai H6 tanpa akhiran -0 atau -1. Label dengan akhiran tersebut tidak digunakan dalam analisis detail karena diperlakukan sebagai *sub-sample* yang muncul akibat *redirect* atau *request* turunan. *Request* autentikasi seperti

Get Login Page dan *POST Login* tetap diperhitungkan dalam statistik global karena menjadi bagian dari alur akses halaman terproteksi, tetapi analisis detail halaman difokuskan pada halaman uji utama H1 sampai H6. Grafik detail per skenario difokuskan pada *Average Response Time* per halaman karena metrik tersebut paling sesuai untuk menunjukkan perbedaan waktu *respons* antarhalaman uji. Berdasarkan Persamaan (2), *average response time* dihitung dari rata-rata waktu *request* pada masing-masing *sampler*. Sementara itu, metrik Total *Request/Samples*, *Throughput*, dan *Error Rate* tetap disajikan dalam tabel detail sebagai data kuantitatif pendukung.

Tabel 7. Detail Hasil Load Testing pada Skenario LT1 dengan 5 Virtual User

LHS	T.R/S	Avg. R.T. (ms)	Tp. (req/s)	E.R. (%)
H1 Home	86	367,81	0,49	0
H2 News	86	386,98	0,52	0
H3 Login Form	84	242,48	0,52	0
H4 Complaint Form	86	351,96	0,52	0
H5 Dashboard	22	1052,95	0,16	0
H6 Data Complaint	22	706,23	0,16	0

LHS = Label/Halaman/Sampler, T = Total, T.R/S = Request/Samples, Avg. R.T. = Average Response Time, Tp = Throughput, E.R. = Error Rate

Berdasarkan Tabel 7, pada skenario LT1 dengan 5 *virtual user*, halaman H5 - *Dashboard* memiliki *average response time* tertinggi, yaitu 1052,95 ms, diikuti oleh H6 - *Data Complaint* sebesar 706,23 ms. Halaman dengan *average response time* terendah adalah H3 - *Login Form* sebesar 242,48 ms. Berdasarkan Persamaan (2), nilai tersebut menunjukkan bahwa *request* pada halaman admin membutuhkan rata-rata waktu *respons* lebih tinggi dibanding halaman publik dan halaman user pada skenario beban rendah. Seluruh *main sampler* pada LT1 memiliki *error rate* 0%, sehingga kegagalan pada *summary* LT1 sebesar 0,12% tidak berasal dari halaman utama H1 sampai H6, tetapi dari request lain dalam alur *Test Plan*.

Tabel 8. Detail Hasil Load Testing pada Skenario LT2 dengan 10 Virtual User

LHS	T.R/S	Avg.R.T. (ms)	Tp. (req/s)	E.R. (%)
H1 Home	154	417,4	0,87	0
H2 News	152	347,69	0,92	0
H3 Login Form	151	325,27	0,92	0
H4 Complaint Form	114	488,82	0,7	0
H5 Dashboard	82	409,34	0,5	0
H6 Data Complaint	81	527,84	0,5	0

LHS = Label/Halaman/Sampler, T = Total, T.R/S = Request/Samples, Avg. R.T. = Average Response Time, Tp = Throughput, E.R. = Error Rate

Berdasarkan Tabel 8, pada skenario LT2 dengan 10 *virtual user*, *average response time* tertinggi terdapat pada H6 - *Data Complaint* sebesar 527,84 ms, diikuti oleh H4 - *Complaint Form* sebesar 488,82 ms. Halaman H3 - *Login Form* memiliki *average response time* terendah, yaitu 325,27 ms. Seluruh halaman utama pada LT2 memiliki *error rate* 0%, yang menunjukkan bahwa

request pada main sampler H1 sampai H6 berhasil diproses tanpa kegagalan teknis. Jika dibandingkan dengan LT1, halaman admin tidak lagi menunjukkan lonjakan response time setinggi H5 pada LT1, tetapi H6 tetap termasuk halaman dengan waktu respons terbesar.

Tabel 9. Detail Hasil Load Testing pada Skenario LT3 dengan 25 Virtual User

LHS	T. R/S	Avg.R.T. (ms)	Tp. (req/s)	E.R. (%)
H1 Home	351	350,98	10,31	0
H2 News	349	350,54	16,72	0
H3 Login Form	345	268,52	6,6	0
H4 Complaint Form	276	365,84	5,97	0
H5 Dashboard	179	416,51	3,83	0
H6 Data Complaint	178	408,49	3,85	0

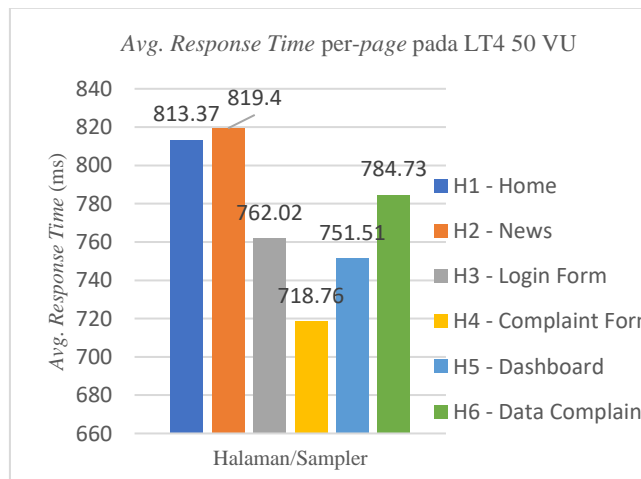
LHS = Label/Halaman/Sampler, T = Total, T.R/S = Request/Samples, Avg. R.T. = Average Response Time, Tp = Throughput, E.R. = Error Rate

Berdasarkan Tabel 9, pada skenario LT3 dengan 25 virtual user, average response time seluruh halaman berada pada rentang 268,52 ms sampai 416,51 ms. Nilai tertinggi terdapat pada H5 - Dashboard sebesar 416,51 ms, sedangkan nilai terendah terdapat pada H3 - Login Form sebesar 268,52 ms. Seluruh main sampler memiliki error rate 0%. Berdasarkan Persamaan (3), throughput per halaman pada LT3 juga meningkat dibandingkan LT1 dan LT2, khususnya pada H1 dan H2. Kondisi ini menunjukkan bahwa pada beban 25 virtual user, aplikasi masih dapat mempertahankan respons yang relatif stabil pada halaman utama yang diuji.

Tabel 10. Detail Hasil Load Testing pada Skenario LT4 dengan 50 Virtual User

LHS	T.R/S	Avg.R.T. (ms)	Tp. (req/s)	E.R. (%)
H1 Home	466	813,37	2,49	5,58
H2 News	454	819,4	2,46	5,73
H3 Login Form	449	762,02	2,53	5,57
H4 Complaint Form	371	718,76	2,12	5,39
H5 Dashboard	255	751,51	1,46	5,88
H6 Data Complaint	252	784,73	1,45	5,56

LHS = Label/Halaman/Sampler, T = Total, T.R/S = Request/Samples, Avg. R.T. = Average Response Time, Tp = Throughput, E.R. = Error Rate



Gambar 7. Average Response Time per Halaman pada Load Test 4 dengan 50 Virtual User

Berdasarkan Tabel 10 dan Gambar 7, pada skenario LT4 dengan 50 virtual user seluruh halaman utama mengalami peningkatan average response time. Nilai tertinggi terdapat pada H2 - News sebesar 819,40 ms, diikuti oleh H1 - Home sebesar 813,37 ms dan H6 - Data Complaint sebesar 784,73 ms. Halaman dengan average response time terendah adalah H4 - Complaint Form sebesar 718,76 ms, tetapi nilainya tetap lebih tinggi dibanding seluruh halaman pada LT3. Pola pada Gambar 7 menunjukkan bahwa pada beban 50 virtual user, peningkatan waktu respons terjadi secara merata pada halaman publik, halaman login, halaman user, maupun halaman admin. Berdasarkan Persamaan (4), error rate pada main sampler H1 sampai H6 pada LT4 berada pada rentang 5,39% sampai 5,88%. Nilai error rate tertinggi terdapat pada H5 - Dashboard sebesar 5,88%, sedangkan error rate terendah terdapat pada H4 - Complaint Form sebesar 5,39%. Meskipun perbedaannya tidak terlalu besar, munculnya error pada seluruh halaman utama menunjukkan bahwa beban 50 virtual user mulai memengaruhi stabilitas respons aplikasi pada lingkungan shared hosting.

Secara keseluruhan, hasil detail per halaman menunjukkan bahwa LT3 merupakan skenario paling stabil pada level main samplers karena seluruh halaman memiliki error rate 0% dengan average response time yang relatif terkendali. Sebaliknya, LT4 menunjukkan peningkatan response time dan error rate pada seluruh halaman utama. Temuan ini mendukung hasil summary pada Tabel 6, yaitu bahwa aplikasi masih menunjukkan performa stabil hingga 25 virtual user, tetapi mulai mengalami tekanan teknis pada skenario 50 virtual user.

4. Kesimpulan

Penelitian ini telah melakukan benchmark empiris terhadap aplikasi Web SiPADUKA V2 pada lingkungan shared hosting berbasis cPanel menggunakan Apache JMeter 5.6.3. Pengujian dilakukan melalui empat skenario load testing dengan variasi 5, 10, 25, dan 50 virtual user. Hasil pengujian menunjukkan bahwa jumlah request dan throughput meningkat seiring bertambahnya beban akses, yaitu dari 857 request dan 4,80 req/s pada LT1 menjadi 5180 request dan 27,68 req/s pada LT4. Average response time berada pada 383,94 ms pada LT1, menurun menjadi 342,40 ms pada LT2 dan 289,72 ms pada LT3, kemudian meningkat menjadi 589,72 ms pada LT4. Error rate tercatat rendah pada LT1 sebesar 0,12%, tidak muncul pada LT2 dan LT3, tetapi meningkat menjadi 4,05% pada LT4.

Berdasarkan hasil tersebut, tujuan penelitian untuk memperoleh baseline (nilai acuan awal) performa SiPADUKA V2 pada shared hosting telah tercapai. Aplikasi menunjukkan performa yang relatif stabil sampai skenario LT3 dengan 25 virtual user, ditunjukkan

oleh *error rate* 0% dan *average response time* yang masih terkendali. Namun, pada skenario LT4 dengan 50 *virtual user*, peningkatan *average response time* dan *error rate* menunjukkan bahwa beban tersebut mulai memberikan tekanan terhadap aplikasi dan lingkungan *shared hosting*. Analisis per halaman juga menunjukkan bahwa pada beban tertinggi, peningkatan *response time* dan *error rate* terjadi pada seluruh halaman utama, baik halaman publik, halaman *login*, halaman pengguna, maupun halaman admin.

Pengembangan berikutnya dapat diarahkan pada pengujian lanjutan dengan variasi durasi, jumlah *virtual user* yang lebih bertahap, serta pengulangan skenario untuk memperoleh hasil yang lebih stabil secara statistik. Selain itu, evaluasi performa dapat diperluas dengan membandingkan hasil pada *shared hosting*, VPS, dan server lokal, serta mengkaji optimasi *query*, *caching*, konfigurasi *database*, dan arsitektur *deployment*. Pengujian tambahan menggunakan *Web performance testing* seperti Lighthouse atau PageSpeed Insights juga dapat digunakan untuk melengkapi hasil *load testing* dari sisi pengalaman *browser* pengguna.

Ucapan Terimakasih

Penulis menyampaikan terima kasih kepada Pusat Penelitian dan Pengabdian kepada Masyarakat (P3M) Politeknik Negeri Sriwijaya (POLSRI) atas dukungan pada kegiatan penelitian sebelumnya—dibiayai oleh Dana PNPB POLSRI pada tahun 2024—yang menjadi dasar pengembangan aplikasi SiPADUKA V2 berbasis Web. Artikel ini merupakan pengembangan lanjutan yang difokuskan pada *benchmark* pengujian *performa* aplikasi SiPADUKA V2 berbasis Web serta penyusunan maupun publikasi artikel yang dilakukan secara mandiri oleh penulis.

Daftar Pustaka

- [1] P. P. Ardini *et al.*, "Mobile Solution for Child Abuse Reporting: Enhancing Accessibility and Protection through SI-ADU," *J. Pendidik. Anak Usia Dini Undiksha*, vol. 12, no. 2, pp. 365–373, Dec. 2024, <https://doi.org/10.23887/paud.v12i2.62641>
- [2] O. Viani, A. Suherman, and P. Utami, "Optimalisasi Penerapan E-Government Pada Program SISABAR Dalam Upaya Meningkatkan Perlindungan Anak (Studi Kasus Dinas Pemberdayaan Perempuan Dan Perlindungan Anak Kabupaten Tangerang)," *J. Ilm. ILMU Adm.*, vol. 13, no. 2, pp. 83–93, Dec. 2023, <https://doi.org/10.33592/jiia.v13i2.3822>
- [3] M. G. P. Syahtoni and O. Reviandani, "PENERAPAN E-GOVERNMENT SISTEM INFORMASI PELAYANAN PERLINDUNGAN PEREMPUAN DAN ANAK DI KELURAHAN PUTAT JAYA," *J. Publicuho*, vol. 7, no. 1, pp. 43–56, Feb. 2024, <https://doi.org/10.35817/publicuho.v7i1.313>
- [4] K. A. P. Astuti, M. M. Putra, M. Murni, and B. Joshi, "Web-Based Bullying Case Reporting Information System at Indonesian Schools in Kota Kinabalu, Malaysia," *J. Abdimas Cendekiawan Indones.*, vol. 3, no. 1, pp. 9–17, Jan. 2026, <https://doi.org/10.56134/jaci.v3i1.160>
- [5] D. Radovanović, "Navigating Child Online Protection in Indonesia: International Norms, Local Realities, and the TikTok Factor," *Digit. Soc. 2026 51*, vol. 5, no. 1, pp. 6–, Jan. 2026, <https://doi.org/10.1007/s44206-025-00244-0>
- [6] S. S. Raweyai and I. R. Widiyarsari, "Performance Testing of Academic Website Using Load Testing Method Supported by Apache JMeter at XYZ University," *J. Tek. Inform.*, vol. 5, no. 3, pp. 721–730, 2024, <https://doi.org/10.52436/1.jutif.2024.5.3.1796>
- [7] Annisa, R. F. Putra, and D. A. Pratama, "Performance Evaluation of Web Applications Using JMeter Load Testing for Server Capacity and Response Efficiency," *J. Appl. Informatics Comput.*, vol. 10, no. 01, pp. 585–590, 2026, <https://doi.org/10.30871/jaic.v10i1.11840>
- [8] Indrianto, "Performance Testing on Web Information System Using Apache JMeter and BlazeMeter," *J. Ilm. Ilmu Terap. Univ. Jambi*, vol. 7, no. 2, pp. 138–149, 2023, <https://doi.org/10.22437/jiutuj.v7i2.28440>
- [9] H. Alnuhait, W. Alzyadat, A. Althunibat, H. Kahtan, B. Zaqaibeh, and H. A. Al-Khawaja, "Web Application Performance Assessment: A Study of Responsiveness, Throughput, and Scalability," *Int. J. Adv. Appl. Sci.*, vol. 11, no. 9, pp. 214–226, 2024, <https://doi.org/10.21833/ijaas.2024.09.023>
- [10] D. I. Permatasari, "Pengujian Aplikasi Menggunakan Metode Load Testing dengan Apache JMeter pada Sistem Informasi Pertanian," *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 8, no. 1, pp. 135–139, 2020, <https://doi.org/10.26418/justin.v8i1.34452>
- [11] "Optimalisasi Performa Website LSP SMK N 2 Kraksaan melalui Pengujian Beban dengan Apache JMeter," *ELSAINS*, vol. 7, no. 1, pp. 9–14, 2025, <https://doi.org/10.30996/elsains.v7i1.13183>
- [12] "PENGUJIAN WEBSITE INVITEES MENGGUNAKAN METODE LOAD TESTING DENGAN APACHE JMETER," *KHARISMATECH*, vol. 18, no. 01, pp. 99–112, 2023, <https://doi.org/10.55645/kharismatech.v18i1.305>
- [13] B. Sabella, R. R. Tina, F. Achmad, A. Lestari, and F. Fathurrahmani, "Pengujian Aplikasi SIHARAPAN Menggunakan Metode Stress Testing," *J. Teknol. Inf. dan Terap.*, vol. 11, no. 1, pp. 7–14, Jun. 2024, <https://doi.org/10.25047/jtit.v11i1.359>
- [14] D. B. Indrianto and D. Nurhamzah, "Uji Performa Aplikasi Sistem Informasi Guru SMK Bali Global Abiansemal Menggunakan Apache JMeter Dan Blazemeter," *e-Jurnal JUSITI (Jurnal Sist. Inf. dan Teknol. Informasi)*, vol. 13, no. 1, pp. 1–12, Jun. 2024, <https://doi.org/10.36774/jusiti.v13i1.1551>
- [15] P. Diantono Abda'u *et al.*, "Perbandingan Kinerja Antara Gatling dan Apache JMeter pada Uji Beban RESTful API," *Infotekmesin*, vol. 15, no. 1, pp. 211–215, Jan. 2024, doi: 10.35970/INFOTEKMESIN.V15I1.2176.
- [16] M. Garjitno, H. Angriani, Y. Saharaeni, P. Studi Sistem Informasi, and S. Kharisma Makassar, "ANALISIS DAN OPTIMASI KINERJA APLIKASI POTIO MENGGUNAKAN LOAD TESTING JMETER," *JTRISTE*, vol. 12, no. 2, pp. 56–68, Oct. 2025, <https://doi.org/10.55645/jtriste.v12i2.625>
- [17] "Development of a Web-Based Point of Sale (POS) Information System Using the Agile Method," *JSAI*, vol. 08, no. 02, pp. 379–386, 2025, <https://doi.org/10.36085/jsai.v8i2.8451>
- [18] H. Setiawan and D. Enda, "Pengujian Load Testing Website Jurusan Teknik Informatika Politeknik Negeri Bengkalis," *JEKIN - J. Tek. Inform.*, vol. 5, no. 1, pp. 1–12, Jan. 2025,

- <https://doi.org/10.58794/jekin.v5i1.820>
- [19] N. Cahyono and Kamarudin, "Perbandingan Gtmetrix, Lighthouse, Pingdom dan Pagespeed Insight dalam evaluasi Performa Website," *J. Ilm. Media Sisofo*, vol. 18, no. 2, pp. 201–210, Oct. 2024, <https://doi.org/10.33998/mediasisfo.2024.18.2.1901>
- [20] R. Susanti and N. Cahyono, "ANALISIS DAN PERBANDINGAN PERFORMA WEBSITE PENERIMAAN MAHASISWA BARU PERGURUAN TINGGI SWASTA YOGYAKARTA," *JATI (Jurnal Mhs. Tek. Inform., vol. 8, no. 6, pp. 12044–12050, Nov. 2024, <https://doi.org/10.36040/jati.v8i6.11607>*
- [21] F. Fuad and T. Haryanti, "Analisis Penggunaan Website Dinas Sumber Daya Air Provinsi DKI Jakarta dalam Menyampaikan Informasi Publik," *Infomatek*, vol. 28, no. 1, pp. 71–78, Apr. 2026, <https://doi.org/10.23969/infomatek.v28i1.43295>
- [22] A. Widi, E. Sedyono, and H. Hendry, "Analisa Performa Website Organisasi Akuatik Menggunakan Automated Software Testing GTmetrix," *J. Teknol. Sist. Inf.*, vol. 5, no. 2, pp. 25–33, Sep. 2024, <https://doi.org/10.35957/jtsi.v5i2.5925>
- [23] A. Ismail, A. Y. Ananta, S. N. Arief, and E. N. Hamdana, "Performance Testing Sistem Ujian Online Menggunakan JMeter pada Lingkungan Virtual," *J. Inform. Polinema*, vol. 9, no. 2, pp. 159–164, 2023, <https://doi.org/10.33795/jip.v9i2.1190>
- [24] M. D. Adila and T. Y. Hadiwandura, "Peningkatan Kinerja dan Skalabilitas Website E-Commerce Menggunakan Load Balancing," *J. Teknol. Inform. dan Komput.*, vol. 10, no. 2, pp. 428–442, Sep. 2024, <https://doi.org/10.37012/jtik.v10i2.2183>
- [25] A. M. N. Hidayat, A. Rizaldy, N. Hartono, and H. Harwalis, "PENGUJIAN KINERJA WEB SERVER ELASTIC CLOUD COMPUTE (EC2) FREE TIER PADA AMAZON WEB SERVICE (AWS) MENGGUNAKAN JMETER," *J. Sist. Inf. dan Inform.*, vol. 7, no. 1, pp. 82–94, Mar. 2024, <https://doi.org/10.47080/simika.v7i1.3208>
- [26] D. Uga Perceka, T. Aditya, and P. Korespondensi, "Pembuatan Sistem Informasi Irigasi dan Kebutuhan Air dengan Standar dan Mekanisme Akses Berbasis Infrastruktur Data Geospasial untuk Irigasi Pertanian di Kabupaten Karanganyar," *JGISE J. Geospatial Inf. Sci. Eng.*, vol. 4, no. 1, pp. 15–29, Apr. 2021, <https://doi.org/10.22146/jgise.61438>
- [27] M. Bagus, T. Cahyo, H. Setiawan, I. Ratna, and I. Astutik, "Implementation and Analysis of Cloud-Native Approaches to Enhance Scalability and Performance in Dwi Jaya Understel Workshop Information System," *J-INTECH*, vol. 13, no. 02, pp. 365–374, Dec. 2025, doi: 10.32664/J-INTECH.V13I02.2083.