

Analisis pengaruh lokasi geografis dan *browser* terhadap performa *website* UNPERBA menggunakan *automated testing* pingdom

Nur Istiqomah¹, Siti Nasiroh¹, Herjuna Ardi Prakosa¹

¹Program Studi Informatika, Universitas Perwira Purbalingga, Purbalingga

Penulis Korespondensi : Siti Nasiroh (e-mail: sitinasiroh@unperba.ac.id)

ABSTRAK

Website UNPERBA (Universitas Perwira Purbalingga) menjadi salah satu sarana utama untuk komunikasi dalam upaya untuk memastikan performa *website* optimal. penelitian ini bertujuan untuk mengukur *load time*, *page size*, dan *request* dari berbagai lokasi geografis serta menentukan pengaruh *browser* terhadap performa *website*. Pengujian dilakukan dari *server* di Jepang, Australia, San Francisco, Washington D.C, Jerman, London, dan Brazil, menggunakan tiga *browser* yaitu, Google Chrome, Mozilla Firefox, dan Microsoft Edge. Hasil penelitian menunjukkan bahwa lokasi geografis memiliki pengaruh signifikan hingga 64% terhadap performa *website*, sementara jenis *browser* berpengaruh sebesar 36%. Faktor kecepatan *internet* dan cara *browser* merender halaman *website* mempengaruhi hasil pengujian. Berdasarkan temuan ini, direkomendasikan optimasi lebih lanjut untuk meningkatkan kinerja *website* UNPERBA dan kualitas layanan bagi pengguna.

KATA KUNCI Performa *Website*, Lokasi, *Browser*, *Automated Testing*.

1. PENDAHULUAN

Perkembangan teknologi yang semakin berkembang saat ini membuat banyak kegiatan sehari-hari menjadi lebih mudah dan efisien. Seiring dengan berkembangnya teknologi di era saat ini informasi dapat dengan mudah diakses melalui situs yang bisa diakses kapanpun dan dimanapun. Menurut Wardiana, pemanfaatan Teknologi Informasi dan Komunikasi (TIK) pada lembaga pendidikan sangat diperlukan dalam hal pengelolaan data dan pengambilan keputusan. Telekomunikasi digunakan agar data dapat disebar dan diakses secara global. Ditambah lagi dengan hadirnya teknologi *online* seperti layanan internet yang semakin memudahkan penggunaannya dalam mengakses segala macam informasi dari data yang disajikan [1].

Website universitas berfungsi sebagai sarana untuk interaksi antara pihak universitas dengan mahasiswa, calon mahasiswa, staf akademik, dan publik. Universitas Perwira Purbalingga (UNPERBA) telah memiliki *website* yang menyediakan informasi terkait program studi, kegiatan akademik, serta berbagai layanan lainnya. Dengan semakin meningkatnya jumlah pengunjung dari berbagai lokasi geografis dan menggunakan beberapa *browser* berbeda diperlukan analisis mendalam untuk memastikan bahwa *website* UNPERBA memiliki performa yang optimal. Performa

dari sebuah *website* sangat mempengaruhi pengguna dalam menyerap informasi yang ada pada situs tersebut.

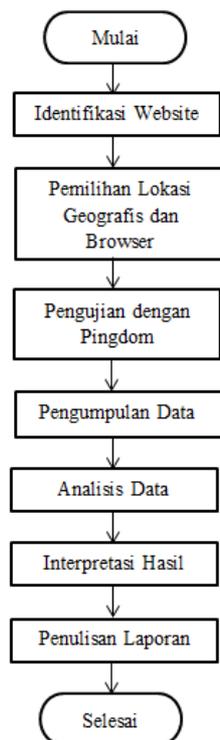
Performa *website* dapat dipengaruhi oleh berbagai faktor termasuk lokasi geografis pengguna dan jenis *browser* yang digunakan. Lokasi berpengaruh signifikan terhadap hasil pengujian kecepatan *website* [2]. Dan setiap *browser* memiliki *rendering* halaman *website* yang berbeda-beda juga dapat mempengaruhi *load time* atau waktu muat halaman [3]. Salah satu yang mempengaruhi kinerja halaman *web* yaitu *loading time*, dimana *loading time* adalah waktu yang dibutuhkan oleh *browser* agar dapat menampilkan halaman *web* secara menyeluruh oleh pengguna ketika pengguna melakukan *request*, selain itu *loading time* merupakan salah satu bagian penting dari optimasi situs *web* [4]. *Load time* yang lambat tidak hanya mengurangi kepuasan pengguna tetapi juga dapat berdampak negatif pada reputasi institusi dan efektivitas layanan yang disediakan. Oleh karena itu, kinerja *website* yang optimal menjadi sangat penting untuk memastikan bahwa pengguna mendapatkan pengalaman yang baik tanpa hambatan teknis yang dapat mengganggu akses informasi. Penelitian ini bertujuan untuk mengukur *load time*, *page size*, dan *request website* UNPERBA dari berbagai lokasi geografis serta menentukan pengaruh jenis *browser* terhadap performa *website*.

2. BAHAN DAN METODE

2.1 Metode Pengumpulan Data

Untuk mencapai tujuan penelitian ini metode yang digunakan adalah metode studi literatur yaitu metode untuk mempelajari teori literatur, buku, karya ilmiah, ataupun situs penyedia layanan yang berkaitan dengan objek penelitian sebagai dasar pemikiran dan referensi. Metode observasi non-partisipan dimana peneliti hanya melakukan pengamatan terhadap *website* UNPERBA (Universitas Perwira Purbalingga) dengan menggunakan *tools* Pingdom.

2.2 Alur Penelitian



Gambar 1. Alur Penelitian

Berdasarkan Gambar 1 dapat dilihat tahapan yang dilakukan oleh peneliti yaitu dengan melakukan identifikasi *website* yang akan diuji. Selanjutnya merupakan tahap pemilihan lokasi geografis dan *browser* yang akan digunakan pada penelitian. Setelah penentuan lokasi dan *browser*, maka dilakukan pengujian menggunakan bantuan *automated testing* Pingdom. Setelah pengujian selesai, data hasil pengujian dikumpulkan untuk selanjutnya di analisis. Tahapan selanjutnya yaitu interpretasi hasil dari pengujian *website* UNPERBA. Dan alur penelitian ditutup dengan penulisan laporan.

2.3 Variabel dan Instrumen Penelitian

Variabel yang akan diuji meliputi, *Load time* (waktu muat) merupakan waktu yang diperlukan untuk memuat sebuah halaman *website*, *page size* (ukuran halaman) merupakan besar ukuran dari seluruh halaman *website*, dan *request* (permintaan) merupakan permintaan yang dibuat oleh pengguna dari *browser* ke

server. Dan instrumen pengujian yang digunakan adalah Pingdom.

2.4 Automated Testing

Metode *automated testing* merupakan sebuah metode pengujian *website* yang menggunakan *software testing* yang terpisah dari sistem yang sedang diuji untuk menjalankan *test case* secara otomatis. Keuntungan utama dari pengujian otomatis adalah kemampuannya untuk membuat proses pengujian menjadi lebih efektif, menghemat waktu, dan memberikan hasil yang konsisten. Pendekatan ini dapat diterapkan untuk berbagai jenis pengujian, termasuk *performance test*, *rest API test*, *end-to-end*, *GUI test*, dan masih banyak lagi, terutama pada aplikasi berbasis *website* dan *mobile* [5]

3. HASIL DAN PEMBAHASAN

Terdapat beberapa tahapan untuk memberikan gambaran umum dari perencanaan hingga eksekusi metode *automated testing* [6] sebagai berikut:

3.1 Identify Test Requirements

Menentukan persyaratan dan cakupan pengujian berdasarkan spesifikasi perangkat lunak. Dengan menentukan variabel pengujian, seperti *load time*, *page size*, dan *request*. Memilih *browser* yang akan digunakan, seperti Google Chrome Version 125.0.6422.113 (*Official Build*) (64-bit), Mozilla Firefox 126.0.1 (32-bit), Microsoft Edge 125.0.2535.92 (*Official build*) (64-bit). Serta menentukan lokasi geografis pengujian, yaitu Tokyo, Sydney, San Fransisco, Washington D.C, London, Frankfurt, dan Sao Paulo. Proses ini melibatkan menentukan apa saja yang perlu diuji untuk memastikan bahwa sistem memenuhi standar kualitas dan fungsionalitas yang diharapkan.

3.2 Select Test Cases for Automated

Automated testing dapat meningkatkan efisiensi, konsistensi, dan kecepatan dalam mendeteksi masalah pada sebuah sistem. Kasus yang diuji adalah yang bersifat berulang dan memerlukan waktu lama jika dilakukan secara *manual* untuk membantu menghemat waktu. yang dalam hal ini kasus yang dipilih untuk *automated testing* adalah pengujian performa *website* UNPERBA

3.3 Choose the Right Automation Tool

Memilih alat pengujian merupakan keputusan penting yang dapat mempengaruhi keberhasilan dan efisiensi proses pengujian *software*. Ada beberapa pertimbangan dalam memilih *automation tool* yaitu, identifikasi kebutuhan pengujian, dengan menentukan jenis pengujian yang akan diotomatisasi (fungsional, non-fungsional, regresi, performa, dll.). Pilih alat yang *user friendly*. Pastikan *tool* yang dipilih dapat menangani *volume* pengujian yang besar dan mampu berjalan efisien pada skenario pengujian yang kompleks. Dan *tool* memiliki kemampuan pelaporan dan analisis komprehensif.

3.4 Design and Develop Test Scripts

Skrip pengujian adalah kode atau serangkaian perintah yang mengotomatiskan proses pengujian untuk memverifikasi bahwa suatu *software* berfungsi sebagaimana mestinya. Menulis skrip pengujian menggunakan alat otomatisasi yang dipilih. Skrip ini mencakup langkah-langkah yang akan dijalankan selama pengujian. Dalam hal ini digunakan alat yang sudah diprogram untuk mengotomatiskan pengujian performa website. Pada tahap ini dilakukan konfigurasi Pingdom untuk menjalankan pengujian tanpa perlu untuk membuat skrip pengujian secara manual.

3.5 Set Up Test Environment

Menyiapkan Tahapan menyiapkan lingkungan pengujian melibatkan konfigurasi alat dan memastikan bahwa semua variabel pengujian telah diatur dengan benar dengan melakukan konfigurasi Pingdom yaitu menambahkan *URL website* UNPERBA ke dalam Pingdom dan mengatur lokasi pengujian (Tokyo, Sydney, San Fransisco, Washington D.C., London, Frankfurt, dan Sao Paulo), serta pemilihan *browser* (Google Chrome, Mozilla Firefox, Microsoft Edge). Serta mengatur jadwal untuk pengujian, pengujian dilakukan pada dua waktu yang berbeda yaitu *before midday* (sebelum tengah hari) mengacu pada waktu dari tengah malam sampai dengan tepat sebelum tengah hari (00:00) – (11:59) dan *after midday* (setelah tengah hari) mengacu pada waktu dari tengah hari hingga waktu sebelum tengah malam (12:00) – (23:59). Waktu pengujian *website* menyesuaikan zona waktu pada masing-masing negara.

3.6 Execute Test Scripts

Pada tahap ini merupakan tahap untuk memverifikasi bahwa sistem berfungsi sesuai yang diharapkan. Tahap ini menjalankan skrip pengujian otomatis menggunakan *tool* yang telah dipilih.

- a. Pengukuran kecepatan akses internet menggunakan speedtest.net, pengukuran ini bertujuan agar ketika pengujian dilakukan pengujian website dapat berjalan dengan baik



Gambar 1. Pengukuran Kexepatan Akses Internet

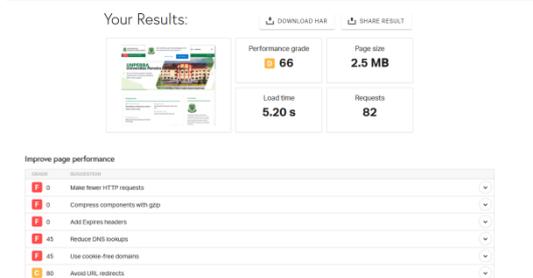
- b. Membuka website Pingdom melalui browser dengan alamat <https://www.pingdom.com/>.
- c. Menyalin alamat *url* dari situs UNPERBA sebagai situs utama ke dalam form *input "URL"*, isi nama dan pilih lokasi pengujian, kemudian klik "*START TEST*". Pingdom akan secara otomatis menganalisis performa dari *website* tersebut dan peneliti perlu menunggu untuk kemudian akan dihubungkan pada

halaman selanjutnya yang memuat hasil analisis dari *website* yang diuji.



Gambar 2. Homepage Pingdom

- d. Setelah data hasil pengujian muncul, peneliti dapat mengumpulkan data hasil pengujian dari Pingdom yang mencakup *load time*, *page size*, dan *request*.



Gambar 3. Tampilan Hasil Pengujian

3.7 Analyze Test Results

Menganalisis hasil pengujian untuk meninjau dan mengevaluasi hasil yang diperoleh dari pengujian, yang bertujuan untuk menemukan permasalahan yang ada pada sistem tersebut.

3.7.1 Analisis Load Time

Tabel 1. Load Time dengan Google Chrome

Google Chrome		<i>Before Midday</i>	<i>After Midday</i>
No.	Negara	<i>Load Time</i>	
1	Tokyo, Jepang	6.44 s	2.59 s
2	Sydney, Australia	2.21 s	4.36 s
3	San Fransisco, USA	2.00 s	2.76 s
4	Washington D.C, USA	2.12 s	1.90 s
5	London, UK	1.40 s	2.63 s
6	Frankfurt, Jerman	1.23 s	1.31 s
7	Sao Paulo, Brazil	4.13 s	3.32 s

Pada tabel 1 pengujian menggunakan Google Chrome Pada tabel 4.4 pengujian menggunakan Google Chrome, pengujian dengan *server* Tokyo, Jepang menunjukkan penurunan signifikan dalam *load time* dari *before midday* ke *after midday* yang mengindikasikan berkurangnya beban jaringan atau *server* pada siang hari pada zona waktu tersebut. Sydney, Australia menunjukan peningkatan *load time* dari *before midday* ke *after midday* yang dapat disebabkan oleh peningkatan aktivitas pengguna atau beban jaringan. Pada *testing* San Fransisco dan Washington D.C menunjukkan variasi kecil dalam *load time* yang dapat disebabkan oleh faktor-faktor yang lebih stabil seperti infrastruktur jaringan yang baik. Testing London dan Jerman mengalami peningkatan *load time after midday* yang dapat disebabkan oleh peningkatan aktivitas pengguna pada siang hari. dan *testing* Sao Paulo, Brazil menunjukkan penurunan *load time* yang dapat mengindikasikan berkurangnya *traffic* pada jaringan atau *server* setelah siang hari.

Tabel 2. Load Time dengan Microsoft Edge

Microsoft Edge		<i>Before Midday</i>	<i>After Midday</i>
No.	Negara	<i>Load Time</i>	
1	Tokyo, Jepang	4.06 s	4.63 s
2	Sydney, Australia	3.11 s	5.38 s
3	San Fransisco, USA	1.64 s	1.78 s
4	Washington D.C, USA	1.53 s	1.73 s
5	London, UK	1.27 s	1.18 s
6	Frankfurt, Jerman	1.14 s	1.16 s
7	Sao Paulo, Brazil	3.87 s	3.28 s

Pada tabel 2 pengujian menggunakan *Microsoft Edge* pada *testing* Tokyo menunjukkan penurunan *load time* pada *after midday* dari 4.06 s menjadi 4.63 s yang mengindikasikan bahwa ada penurunan *traffic* jaringan atau *server*. *Testing* Sydney mengalami peningkatan pada

after midday sebesar 2.27 s yang dapat disebabkan oleh peningkatan aktivitas pengguna pada rentang waktu tersebut. Pada *testing* San Fransisco dan Washington D.C *load time* mengalami peningkatan pada siang hari yang dapat disebabkan oleh banyaknya aktivitas pengguna. Sedangkan pada *testing* London dan Jerman menunjukkan *load time* paling rendah dibandingkan 5 negara lainnya, hal ini menunjukkan efisensi jaringan yang baik pada dua wilayah tersebut.

Dan pada *testing* Sao Paulo mengalami penurunan setelah siang hari dengan selisih waktu sebesar 0.59 s.

Tabel 3. Load Time dengan Mozilla Firefox

Mozilla Firefox		<i>Before Midday</i>	<i>After Midday</i>
No.	Negara	<i>Load Time</i>	
1	Tokyo, Jepang	4.36 s	5.84 s
2	Sydney, Australia	4.07 s	5.38 s
3	San Fransisco, USA	1.65 s	1.72 s
4	Washington D.C, USA	1.83 s	1.91 s
5	London, UK	1.30 s	1.20 s
6	Frankfurt, Jerman	1.33 s	859 ms
7	Sao Paulo, Brazil	3.31 s	2.33 s

Pada tabel 3 pengujian menggunakan Mozilla Firefox, *testing* Tokyo dan Sydney cenderung mengalami peningkatan 1.3 s.d 1.4 s. Sedangkan *testing* San Fransisco, Washington D.C, dan London mengalami peningkatan *load time* yang relatif kecil sehingga cenderung lebih stabil. Pada *testing* Frankfurt, Jerman mengalami penurunan *load time* yang signifikan dari 1.33 s menjadi 0,859 ms. Dan *testing* Sao Paulo mengalami penurunan *load time* sebesar 0.98 s yang dapat disebabkan oleh berkurangnya *traffic* jaringan.

Pada hasil pengujian *website* UNPERBA menggunakan Pingdom menunjukkan bahwa terjadi perbedaan *load time* dari masing-masing negara. Dari pengujian menggunakan tiga *browser* berbeda, wilayah dengan *load time* terendah adalah London dan Frankfurt kemudian disusul oleh wilayah lainnya. Lokasi geografis dengan *load time* tertinggi adalah Tokyo dan Sydney, secara geografis Tokyo dan Sydney lebih dekat dengan *server website* UNPERBA, hal ini dapat dipengaruhi oleh beberapa faktor sebagai berikut:

1. Kondisi Jaringan Internasional

Jalur fisik yang ditempuh oleh data dapat lebih panjang atau lebih padat, meskipun jaraknya lebih dekat. Terkadang rute jaringan antara Indonesia dan Tokyo atau Sydney dapat melewati banyak titik persimpangan atau jalur yang padat. Kualitas dan kapasitas jalur kabel bawah laut yang menghubungkan Indonesia dengan Tokyo dan Sydney bisa bervariasi. Konektivitas yang lebih buruk atau gangguan pada kabel bawah laut dapat meningkatkan latensi.

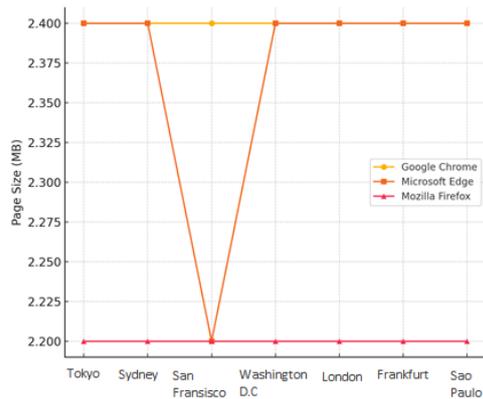
2. Pengaturan CDN

Meskipun ada *server* CDN di Tokyo dan Sydney, tidak semua konten di-cache dengan baik pada lokasi tersebut. Jika konten yang diminta tidak tersedia di *server* lokal CDN dan harus diambil dari lokasi yang lebih jauh yang dapat meningkatkan *load time*.

3. Kepadatan Traffic dan Beban Server

Server di Tokyo atau Sydney sedang mengalami beban tinggi atau tingginya *traffic* yang dapat memperlambat respon. Jaringan yang padat dan permintaan tinggi pada *server* tertentu dapat mempengaruhi kecepatan pengiriman konten.

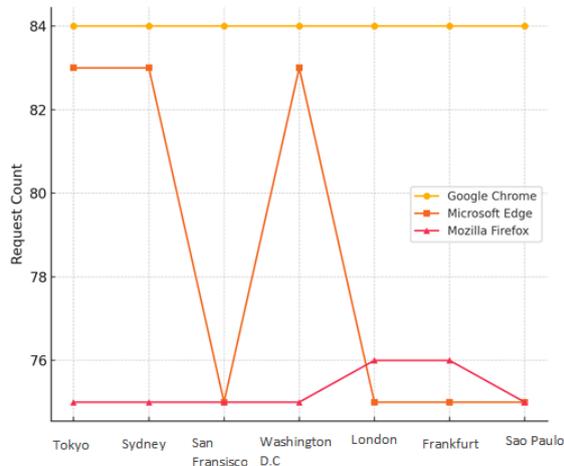
3.7.2 Analisis Page Size



Gambar 4. Grafik Page Size

Pada hasil pengujian dengan *browser* Google Chrome menunjukkan bahwa *page size* untuk semua lokasi geografis konstan 2,4 MB. *Page size* pada Microsoft Edge untuk sebagian lokasi memiliki ukuran yang sama yaitu 2.4 MB, kecuali San Fransisco yang memiliki *page size* 2.2 MB. Dan pada Mozilla Firefox memiliki *page size* konstan 2.2 MB untuk seluruh lokasi geografis. Pada tabel diatas terdapat perbedaan *page size* pada tiga *browser* (Google Chrome, Microsoft Edge, dan Mozilla Firefox).

3.7.3 Analisis Request



Gambar 5. Grafik Request

Pada Gambar 4.2 bahwa Google Chrome memiliki jumlah *request* konstan 84 disemua lokasi geografis. Pada *browser* Microsoft Edge memiliki jumlah *request*

umumnya pada angka 83, dengan beberapa variasi di San Fransisco (76), Washington D.C (84), dan Frankfurt (75). Jumlah *request* pada Mozilla Firefox konstan di angka 75, kecuali di Washington D.C yang memiliki *request* 76.

3.8 Reporting

Hasil pengujian dapat digunakan oleh tim *developer* untuk evaluasi terhadap sistem yang diuji. laporan hasil pengujian berisi informasi dalam bentuk statistik atau grafitk yang memungkinkan mereka mengambil keputusan untuk melakukan evaluasi terhadap sistem mereka.

3.8.1 Pengaruh Lokasi Geografis

Pengaruh lokasi geografis terhadap performa *website* sangat signifikan ketika diakses dari lokasi geografis yang berbeda. Semakin jauh server dari lokasi pengguna, semakin tinggi tingkat latensi yang dapat meningkatkan *load time* halaman *website*. Hal ini sering kali disebabkan oleh jumlah *hop* atau perantara yang harus dilalui data, serta kualitas infrastruktur jaringan di wilayah tersebut. Oleh karena itu, dalam beberapa kasus, lokasi geografis dapat mempengaruhi performa *website* hingga 64%.

3.8.2 Pengaruh Jenis Browser

Pengujian menggunakan tiga *browser* menunjukkan performa berbeda dalam hal *load time*. Microsoft Edge cenderung memiliki memiliki waktu muat lebih cepat dibandingkan dengan Google Chrome dan Mozilla Firefox, hal ini disebabkan oleh optimasi *internal browser* tersebut serta teknik *rendering* yang digunakan oleh masing-masing *browser*. Pengujian *website* menggunakan Mozilla Firefox menunjukkan *page size* konsisten 2.2 MB dan *request* rata-rata 75 yang lebih konsisten dan efisien dibandingkan dengan dua *browser* lainnya. Oleh karena itu, jenis *browser* berpengaruh positif terhadap performa *website* dapat berkisar antara 36%.

3.8.3 Rekomendasi Peningkatan Performa dari Pingdom



Gambar 6. Improve Page Performance Pingdom

a) Mengurangi Permintaan HTTP

Mengurangi jumlah komponen pada halaman mengurangi jumlah permintaan HTTP yang diperlukan untuk merender halaman, sehingga mempercepat *load time* halaman. Beberapa cara untuk mengurangi jumlah komponen termasuk menggabungkan *file*, menggabungkan beberapa skrip menjadi satu skrip, menggabungkan beberapa *file* CSS menjadi satu *style sheets*, dan menggunakan CSS *Sprites* dan *image maps*.

b) Mengompresi Komponen dengan gzip

Kompresi mengurangi waktu respons dengan mengurangi ukuran respons HTTP. Gzip adalah metode kompresi paling populer dan efektif yang tersedia saat ini dan umumnya mengurangi ukuran respons sekitar 70%. Sekitar 90% lalu lintas Internet saat ini melewati *browser* yang mengklaim mendukung gzip.

c) Menambahkan *Header Expires*

Halaman web semakin kompleks dengan lebih banyak skrip, *style sheets*, gambar, dan *Flash* di dalamnya. Kunjungan pertama ke halaman mungkin memerlukan beberapa permintaan HTTP untuk memuat semua komponen. Dengan menggunakan *header Expires*, komponen-komponen ini menjadi dapat di-*cache*, yang menghindari permintaan HTTP yang tidak perlu pada tampilan halaman berikutnya. *Header Expires* paling sering dikaitkan dengan gambar, tetapi dapat dan harus digunakan pada semua komponen halaman termasuk skrip, *style sheets*, dan *Flash*.

d) Mengurangi Pencarian DNS

Sistem Nama Domain (DNS) memetakan nama *host* ke alamat IP. Ketika mengetik URL www.unperba.ac.id ke dalam *browser*, *browser* menghubungi *resolver* DNS yang mengembalikan alamat IP server. DNS memiliki biaya; biasanya membutuhkan waktu 20 hingga 120 milidetik untuk mencari alamat IP untuk nama *host*. *Browser* tidak dapat mengunduh apa pun dari *host* sampai pencarian selesai.

e) Menggunakan Domain Bebas Cookie

Ketika *browser* meminta gambar statis dan mengirimkan cookie dengan permintaan, server mengabaikan *cookie* tersebut. *Cookie* ini adalah lalu lintas jaringan yang tidak perlu. Untuk mengatasi masalah ini, pastikan bahwa komponen statis diminta dengan permintaan bebas *cookie* dengan membuat subdomain dan menempatkannya di sana.

f) Menghindari Pengalihan URL

Pengalihan URL dilakukan menggunakan kode status HTTP 301 dan 302. Mereka memberi tahu browser untuk pergi ke lokasi lain. Menyisipkan pengalihan antara pengguna dan dokumen HTML akhir menunda semuanya di halaman karena tidak ada yang dapat dirender dan tidak ada komponen yang dapat diunduh sampai dokumen HTML tiba.

g) Menghindari *src* atau *href* kosong

IE membuat permintaan ke direktori tempat halaman berada; Safari, Chrome, Firefox 3 dan sebelumnya membuat permintaan ke halaman itu sendiri. Perilaku ini bisa merusak data pengguna, membuang siklus komputasi *server* untuk menghasilkan halaman yang tidak akan pernah dilihat, dan dalam kasus terburuk, melumpuhkan *server* dengan mengirimkan sejumlah besar lalu lintas yang tidak terduga.

4. KESIMPULAN

Pengaruh lokasi geografis terhadap performa website UNPERBA memiliki presentase sebesar 64%

berdasarkan pengujian yang telah dilakukan. Sedangkan pengaruh jenis browser memiliki presentase sebesar 36%.

Saran bagi pihak UNPERBA untuk perbaikan performa website sebagai berikut:

- Dengan menggunakan CDN (*Content Delivery Network*), dengan memanfaatkan CDN dapat mengurangi load time dengan menyajikan konten dari *server* yang lebih dekat dengan pengguna.
- Kompresi gambar dan media untuk mengurangi *page size* dan *load time*.
- Minifikasi *file* CSS, JavaScript, dan HTML dapat mengurangi ukuran *file*, sehingga memperkecil *page size*.
- Perlu dilakukan survei kepuasan pengguna guna mendapatkan masukan mengenai aspek-aspek yang perlu diperbaiki. serta perlunya optimasi *website* secara berkala untuk meningkatkan performa *website* UNPERBA.

Saran bagi penelitian selanjutnya adalah dapat melakukan pengujian dengan berbagai jaringan dan provider berbeda, dapat menggunakan alat pengujian lain seperti Gtmetrix, Katalon, PageSpeed Insight, serta menggunakan lokasi pengujian yang lebih dekat dengan lingkup Universitas Perwira Purbalingga. Keterbatasan pada penelitian ini karena terbatasnya testing server yang tersedia pada Pingdom yang hanya menyediakantujuh server yaitu Tokyo, Sydney, San Fransisco, Washington D.C, London, Frankfurt, dan Sao Paulo.

5. DAFTAR PUSTAKA

- [1] M. Irfan Nur, M. S. Lamada, dan M. Riska, "Rancang Bangun Sistem Informasi Kegiatan Kemahasiswaan Jurusan Teknik Informatika dan Komputer Berbasis Website," hal. 1–12, 2020.
- [2] D. Muriyatmoko dan A. Musthafa, "Website Performance Testing Using Speed Testing Model: A Case of Reputable Indonesian Journals," *J. Ilmu Tek. dan Inform.*, vol. 2, no. 1, hal. 40–45, 2022, doi: 10.51903/teknik.v2i1.120.
- [3] M. B. Purba, I. M. S. Putra, dan A. A. K. O. Sudana, "Pengujian Performa Sistem Single Sign On SRUTI pada Universitas Hindu Indonesia Menggunakan Tools GTmetrix," *JITTER J. Ilm. Teknol. dan Komput.*, vol. 2, no. 3, hal. 578, 2021.
- [4] A. Y. Fadli, I. Nuryasin, dan Z. Sari, "Optimasi Kecepatan Loading Time Web Template dengan Implementasi Teknik Front-End," vol. 2, no. 11, hal. 1456–1463, 2020.
- [5] S. Sri Rahayu, D. Aris Firmansyah, dan S. Susanti, "Analisis Penggunaan Tools Automation Testing pada Aplikasi: Systematic Literature Review," vol. 8, hal. 106–116, 2024.
- [6] B. Homès, *Fundamentals of Software Testing*.

Amerika Serikat: Wiley, 2024. [Daring].
Tersedia pada:
<https://www.google.co.id/books/edition/Funda>

[mentals_of_Software_Testing/knUOEQAAQB
AJ?hl=id&gbpv=0](#)