

Analisis Komparasi Kecepatan Akses Website E-Commerce Antara Arsitektur Web Hosting Konvensional Dan Cloud Serverless Terdistribusi

Guritno Wulandoro Suryosaputro^{1*}, Aris Wahyu Prasetyo², Muhammad Faruq Ghullam Al Khoiri³,
Muhammad lutfi Samsul Al fajar⁴, Agustina Srirahayu⁵

¹Teknik Informatika
Universitas Duta Bangsa
^{1*}240103132@udb.ac.id

²Teknik Informatika
Universitas Duta Bangsa
²240103126@udb.ac.id

³Teknik Informatika
Universitas Duta Bangsa
³240103139@udb.ac.id

⁴Teknik Informatika
Universitas Duta Bangsa
⁴240103141@udb.ac.id

⁵ Teknik Informatika
Universitas Duta Bangsa
⁵Agustina@udb.ac.id

Abstrak—Perkembangan platform e-commerce yang pesat menuntut infrastruktur hosting yang mampu memberikan performa tinggi, respons cepat, dan ketersediaan layanan yang optimal. Penelitian ini bertujuan menganalisis dan membandingkan kecepatan akses website e-commerce antara dua arsitektur infrastruktur hosting, yaitu platform terkelola yang berjalan secara always-on (direpresentasikan oleh Railway) dan cloud serverless terdistribusi dengan edge network global (direpresentasikan oleh Vercel). Pengujian dilakukan menggunakan alat ukur performa web GTmetrix dan Google PageSpeed Insights, dengan parameter yang diukur mencakup Time to First Byte (TTFB), First Contentful Paint (FCP), dan Largest Contentful Paint (LCP) pada kondisi jaringan desktop dan mobile. Hasil analisis menunjukkan bahwa arsitektur serverless unggul pada metrik TTFB dalam simulasi desktop, namun pada simulasi mobile dengan throttling jaringan Slow 4G, platform always-on justru mencatat skor performa dan LCP yang lebih baik, sehingga tidak satu arsitektur pun yang secara mutlak unggul di seluruh skenario pengujian. Pola perbedaan performa yang teramati konsisten dengan penjelasan pada literatur mengenai karakteristik cold start pada arsitektur serverless, meskipun pengukuran cold start secara eksplisit belum dilakukan dalam penelitian ini sehingga keterkaitan tersebut masih bersifat interpretatif. Penelitian ini menyimpulkan bahwa pemilihan arsitektur hosting sebaiknya disesuaikan dengan karakteristik trafik, kebutuhan skalabilitas, dan anggaran operasional masing-masing platform e-commerce.

Kata Kunci—web hosting konvensional, cloud serverless, kecepatan akses, e-commerce, komparasi performa.

Abstract—The rapid growth of e-commerce platforms requires hosting infrastructures capable of delivering high performance, fast response times, and optimal service availability. This study aims to analyze and compare the website access performance of e-commerce platforms deployed on two different infrastructure architectures: an always-on managed hosting platform (represented by Railway) and a distributed serverless cloud architecture with a global edge network (represented by Vercel). Performance testing was conducted using GTmetrix and Google PageSpeed Insights, with evaluated parameters consisting of Time to First Byte (TTFB), First Contentful Paint (FCP), and Largest Contentful Paint (LCP) under desktop and mobile network conditions. The results indicate that the serverless architecture achieves lower TTFB under desktop simulation, whereas under mobile simulation with Slow 4G throttling, the always-on platform recorded better performance scores and LCP, so neither architecture was unambiguously superior across all scenarios. The observed performance pattern is consistent with literature explanations of the cold start phenomenon in serverless architectures, although explicit cold start measurement was not conducted in this study, so this link remains interpretive rather than empirically confirmed. The study concludes that the selection of a hosting architecture should be aligned with traffic characteristics, scalability requirements, and operational budget constraints of the respective e-commerce platform.

Keywords—Conventional Web Hosting, Serverless Cloud Architecture, Web Performance, E-Commerce, Performance Evaluation.

1. PENDAHULUAN

Pertumbuhan industri *e-commerce* di Indonesia dalam beberapa tahun terakhir menunjukkan akselerasi yang signifikan. Berdasarkan data Statista, nilai *Gross Merchandise Value* (GMV) *e-commerce* Indonesia tercatat sebesar USD 40,4 miliar pada tahun 2023 dan diproyeksikan mencapai

USD 63,2 miliar pada 2028 dengan *Compound Annual Growth Rate* (CAGR) sebesar 9,04% [1]. Ekspansi ini turut didukung oleh penetrasi internet yang terus meningkat, di mana Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) mencatat tingkat penetrasi internet nasional sebesar 79,5% pada awal tahun 2024 [2]. Kondisi ini

menempatkan Indonesia sebagai pasar *e-commerce* terbesar di Asia Tenggara, sekaligus menghadirkan tekanan besar terhadap infrastruktur teknologi informasi yang menopangnya.

Dalam ekosistem *e-commerce* yang semakin kompetitif, kecepatan akses *website* menjadi faktor kritis yang secara langsung memengaruhi pengalaman pengguna dan konversi transaksi. Studi kasus industri yang dilakukan oleh van Riet, Malavolta, dan Ghaleb [3] menunjukkan bahwa optimasi performa web yang dilakukan secara sistematis dapat mempercepat waktu muat halaman secara signifikan dan bahwa metrik performa web berkorelasi kuat dengan persepsi pengguna terhadap kecepatan akses, yang menjadi faktor kunci keberhasilan pengalaman pengguna pada platform digital. Temuan ini konsisten dengan data yang banyak dikutip dalam literatur performa web, yaitu bahwa setiap tambahan 100 ms latensi pada platform Amazon secara historis dilaporkan menyebabkan kerugian penjualan sebesar 1% [4]. Di sisi lain, kecepatan *checkout* yang lambat juga berkontribusi pada tingginya angka *cart abandonment*: penelitian Sundjaja et al. terhadap konsumen *e-commerce* di Indonesia menemukan bahwa proses *checkout* yang rumit merupakan faktor paling signifikan yang mendorong pengabaian keranjang belanja [5].

Dalam menjawab tantangan performa ini, pengelola platform *e-commerce* dihadapkan pada dua pilihan utama arsitektur infrastruktur hosting. Pertama, *web hosting* konvensional berbasis server fisik atau *Virtual Private Server* (VPS) yang terpusat, yang menawarkan kontrol penuh dan biaya yang terprediksi, namun memiliki keterbatasan dalam hal skalabilitas otomatis dan distribusi geografis. Kedua, arsitektur *cloud serverless* terdistribusi seperti AWS Lambda, Google Cloud Run, Vercel, dan Cloudflare Workers yang menawarkan skalabilitas elastis, model pembayaran *pay-as-you-go*, serta distribusi konten melalui jaringan *edge* yang lebih dekat dengan pengguna akhir. Meskipun demikian, fenomena *cold start* pada arsitektur *serverless* yakni jeda inisialisasi yang terjadi saat fungsi pertama kali dipanggil setelah periode tidak aktif menjadi tantangan teknis yang perlu diperhitungkan. Golec et al. [8] dalam tinjauan sistematis mereka terhadap lebih dari 100 makalah mencatat bahwa latensi *cold start* dapat mencapai

beberapa detik pada kondisi tertentu, bergantung pada *runtime* dan penyedia layanan yang digunakan. Perlu dicatat bahwa dalam penelitian ini, keterbatasan akses terhadap infrastruktur *shared hosting* atau VPS tradisional mendorong penggunaan definisi operasional yang lebih longgar untuk arsitektur "konvensional", yaitu platform *Platform-as-a-Service* (PaaS) modern yang menjalankan kontainer aplikasi secara *always-on* tanpa fungsi *serverless*, sebagaimana dijelaskan lebih lanjut pada bagian Metodologi.

Sejumlah penelitian terdahulu telah membandingkan performa arsitektur *serverless* dan *server-based*, namun umumnya pada konteks aplikasi generik, layanan pembelajaran mesin, atau platform komputasi awan secara umum [7], [9], [11], [12], sementara studi yang secara khusus mengukur kecepatan akses *website e-commerce* pada kedua arsitektur tersebut, terutama dalam konteks pasar *e-commerce* lokal Indonesia, masih sangat terbatas. Kesenjangan literatur inilah yang mendorong penelitian ini untuk melakukan analisis komparasi yang sistematis dan terukur dengan memanfaatkan alat pengujian performa *web* yang terstandarisasi. Secara spesifik, penelitian ini bertujuan untuk: (1) mengukur dan membandingkan kecepatan akses *website e-commerce* yang dihosting pada arsitektur *always-on* (Railway) dan arsitektur *serverless* terdistribusi (Vercel) berdasarkan parameter Time to First Byte (TTFB), First Contentful Paint (FCP), dan Largest Contentful Paint (LCP); (2) menganalisis pengaruh kondisi jaringan desktop dan mobile (dengan simulasi *Slow 4G*) terhadap performa masing-masing arsitektur; dan (3) menghasilkan data empiris yang dapat dijadikan panduan bagi pengembang dan pengelola platform *e-commerce* dalam menentukan strategi infrastruktur hosting yang optimal sesuai kebutuhan mereka.

II. METODOLOGI PENELITIAN

A. Desain Penelitian

Penelitian ini menggunakan pendekatan eksperimental kuantitatif dengan desain komparasi dua kelompok independen. Objek penelitian adalah dua instance *website e-commerce* yang identik secara fungsional namun dihosting pada infrastruktur yang berbeda: satu pada platform hosting terkelola yang berjalan secara *always-on*

(Railway, dengan kontainer aplikasi yang berjalan terus-menerus pada infrastruktur terkelola) dan satu lagi pada platform cloud serverless (Vercel dengan edge network global). Perlu ditegaskan bahwa Railway merupakan platform cloud modern berbasis container orchestration dan bukan representasi shared hosting atau VPS tradisional; penggunaannya dalam penelitian ini dimaksudkan untuk mewakili karakteristik hosting always-on dengan alokasi sumber daya tetap, sebagai pembandingan terhadap model eksekusi serverless yang bersifat on-demand. Website pengujian dibangun menggunakan framework Next.js dengan dataset produk fiktif sebanyak 1.000 item untuk mensimulasikan skenario e-commerce nyata.

B. Alat Pengujian

Pengujian performa website dilakukan menggunakan dua alat pengujian yang umum digunakan dalam evaluasi kualitas dan kecepatan aplikasi web, yaitu GTmetrix dan Google PageSpeed Insights. Kedua alat tersebut dipilih karena mampu memberikan metrik performa yang komprehensif berdasarkan standar industri serta rekomendasi dari Google Lighthouse.

GTmetrix digunakan untuk mengukur performa halaman web melalui berbagai metrik seperti GTmetrix Grade, Largest Contentful Paint (LCP), Interaction to Next Paint (INP), dan Cumulative Layout Shift (CLS), dengan simulasi kondisi desktop pada jaringan broadband. Selain itu, GTmetrix menyediakan fitur waterfall chart yang menampilkan urutan serta durasi pemuatan setiap sumber daya halaman sehingga memudahkan proses analisis bottleneck performa. Google PageSpeed Insights digunakan untuk mengevaluasi performa website berdasarkan data laboratorium (Lab Data) dengan emulasi perangkat mobile dan throttling jaringan Slow 4G. Analisis dilakukan menggunakan teknologi Lighthouse sehingga mampu memberikan skor performa beserta rekomendasi optimasi yang dapat diterapkan untuk meningkatkan kualitas pengalaman pengguna.

Dengan menggunakan kedua alat tersebut secara komplementer pada kondisi desktop dan mobile, penelitian ini memperoleh data performa website dari dua perspektif skenario akses yang berbeda,

sehingga hasil evaluasi yang diperoleh menjadi lebih representatif terhadap kondisi penggunaan nyata.

C. Prosedur Pengujian Perbandingan

Pengujian komparatif dilakukan secara sistematis untuk membandingkan performa website e-commerce yang dihosting pada infrastruktur web hosting konvensional dan cloud serverless. Tahap pertama adalah mempersiapkan dua versi website yang identik, baik dari sisi kode program, konten, maupun basis data, kemudian melakukan deployment pada masing-masing infrastruktur yang akan dibandingkan. Langkah ini bertujuan untuk memastikan bahwa perbedaan hasil pengujian hanya dipengaruhi oleh perbedaan infrastruktur dan bukan oleh faktor aplikasi.

Selanjutnya, dilakukan konfigurasi baseline dengan mendokumentasikan spesifikasi teknis dari masing-masing lingkungan hosting, seperti lokasi server atau region serta penggunaan Content Delivery Network (CDN). Pada tahap ini cache browser dinonaktifkan agar hasil pengukuran merepresentasikan performa aktual dari server.

Protokol awal penelitian ini merencanakan pengujian pada dua kondisi terpisah untuk lingkungan cloud serverless, yaitu cold state (setelah periode idle minimal 15 menit) dan warm state, guna mengukur secara eksplisit waktu inisialisasi awal (cold start latency). Namun, dalam pelaksanaannya pengujian cold state dengan periode idle terukur belum dapat dilaksanakan pada tahap ini; seluruh data yang dilaporkan pada bagian Hasil dan Pembahasan merupakan hasil pengujian pada kondisi layanan yang aktif (warm state). Keterbatasan ini dibahas lebih lanjut pada bagian Limitasi Penelitian.

Pengukuran performa dilakukan menggunakan GTmetrix dan Google PageSpeed Insights. Setiap halaman diuji minimal lima kali pada masing-masing alat dengan menggunakan lokasi server pengujian yang sama guna menjaga konsistensi data. Pengujian beban (load testing) dengan simulasi pengguna konkuren menggunakan Apache JMeter atau k6 sempat direncanakan pada tahap awal penelitian, namun belum dilaksanakan pada penelitian ini dan dicatat sebagai keterbatasan sekaligus arah penelitian lanjutan.

Data hasil pengujian kemudian dicatat dan ditabulasi berdasarkan metrik utama, seperti Time to First Byte (TTFB), First Contentful Paint (FCP), Largest Contentful Paint (LCP), dan Total Blocking Time (TBT). Dari data hasil pengujian berulang tersebut dilaporkan nilai serta rentang (range) untuk memperoleh gambaran performa yang lebih representatif.

Analisis statistik inferensial menggunakan uji t-independen atau Mann-Whitney U sempat direncanakan untuk menentukan signifikansi perbedaan performa antara kedua infrastruktur hosting, namun belum dilaksanakan pada penelitian ini mengingat jumlah replikasi pengukuran yang masih terbatas; hasil yang disajikan bersifat deskriptif. Hasil pengujian kemudian disajikan dalam bentuk tabel komparatif, disertai pembahasan mengenai faktor-faktor teknis yang memengaruhi performa, seperti karakteristik eksekusi pada arsitektur serverless, penggunaan CDN, serta kemampuan skalabilitas masing-masing platform.

D. Skenario Pengujian

Pengujian performa dilakukan pada tiga halaman utama yang merepresentasikan fungsi-fungsi penting dalam platform e-commerce. Halaman pertama adalah halaman beranda (*homepage*) yang memuat komponen carousel, banner promosi, serta daftar produk unggulan. Halaman kedua adalah halaman kategori produk yang menyediakan fitur pencarian, penyaringan (*filtering*), dan pengurutan (*sorting*) produk. Halaman ketiga adalah halaman detail produk yang menampilkan informasi lengkap produk, galeri gambar, spesifikasi, serta fitur pendukung lainnya.

Setiap halaman diuji pada dua kondisi jaringan yang berbeda untuk mensimulasikan variasi akses pengguna. Kondisi pertama menggunakan jaringan 4G Mobile yang merepresentasikan akses melalui perangkat seluler dengan keterbatasan bandwidth dan latensi yang lebih tinggi. Kondisi kedua menggunakan jaringan broadband desktop yang merepresentasikan akses melalui koneksi internet berkecepatan tinggi pada perangkat desktop atau laptop. Pengujian pada kedua kondisi tersebut bertujuan untuk memperoleh gambaran performa website pada berbagai skenario penggunaan yang umum terjadi pada lingkungan nyata.

III. HASIL DAN PEMBAHASAN

A. Perbandingan Karakteristik Arsitektur

Pengujian dilakukan terhadap dua instance website e-commerce yang identik secara fungsional, yaitu sebuah platform koperasi digital berbasis Next.js dengan katalog produk yang sama, yang dideploy pada dua arsitektur hosting yang berbeda. Instance pertama dihosting pada Railway, yang dalam penelitian ini diposisikan sebagai representasi arsitektur web hosting konvensional karena menjalankan kontainer aplikasi secara terus-menerus (*always-on*) pada infrastruktur terkelola dengan lokasi data center yang relatif tetap. Instance kedua dihosting pada Vercel, yang merepresentasikan arsitektur cloud serverless terdistribusi dengan model edge function dan content delivery network (CDN) global yang menempatkan aset statis dan fungsi komputasi pada node yang tersebar di berbagai lokasi geografis. Perlu digarisbawahi bahwa Railway secara teknis merupakan platform cloud modern berbasis container orchestration dan managed infrastructure, bukan shared hosting atau VPS tradisional sebagaimana lazimnya dipahami sebagai "web hosting konvensional". Posisinya dalam penelitian ini dibatasi pada representasi model eksekusi *always-on* dengan alokasi sumber daya tetap, sehingga perbandingan yang dihasilkan perlu dimaknai sebagai perbandingan antara dua model eksekusi (*always-on* terkelola vs. serverless terdistribusi), bukan perbandingan hosting konvensional murni versus cloud modern.

Kedua platform diuji menggunakan domain produksi masing-masing, yakni `kopmerahputih.up.railway.app` untuk Railway dan `kop-merahputih.vercel.app` untuk Vercel, dengan kondisi peramban (browser) Chrome versi 142 dan mesin analisis Lighthouse versi 12.6.1 yang konsisten pada kedua pengujian. Pengujian dilakukan dari lokasi server pengujian yang sama, yaitu Seattle, Amerika Serikat, untuk meminimalkan bias geografis terhadap hasil pengukuran. Perbedaan mendasar pada karakteristik kedua arsitektur ini sejalan dengan temuan Wen et al. dalam tinjauan sistematis mereka yang menyatakan bahwa arsitektur serverless mengalokasikan sumber daya komputasi secara dinamis berdasarkan permintaan, sedangkan model komputasi awan konvensional

mengharuskan pengembang menyewa dan mempertahankan sumber daya secara tetap terlepas dari status aplikasi sedang berjalan atau tidak [7].

B. Analisis Perbandingan Metrik

Pengujian performa dilakukan menggunakan dua alat ukur utama, yaitu GTmetrix yang mensimulasikan kondisi desktop dengan jaringan broadband, dan Google PageSpeed Insights (PSI) yang mensimulasikan kondisi perangkat mobile (emulasi Moto G Power) dengan throttling jaringan Slow 4G. Kombinasi kedua alat ini memungkinkan evaluasi performa pada dua skenario akses yang berbeda, sehingga hasil yang diperoleh lebih representatif terhadap kondisi penggunaan nyata. Rekapitulasi hasil pengujian kedua platform disajikan pada Tabel 1.

Tabel 1. Perbandingan Metrik Performa Railway dan Vercel

Metrik	Railway (Konvensional)	Vercel (Serverless)
TTFB (GTmetrix)	325 ms	220 ms
FCP (GTmetrix, desktop)	836 – 973 ms	979 ms
LCP (GTmetrix, desktop)	758 – 836 ms	507 – 979 ms
GTmetrix Grade / Performance	A / 97–98%	A / 96–97%
Structure Score (GTmetrix)	94–95%	95–96%
TBT (GTmetrix)	96 – 114 ms	96 – 142 ms
PSI Performance Score (mobile)	91	87
FCP (PSI, mobile, Slow 4G)	2,7 s	2,7 s
LCP (PSI, mobile, Slow 4G)	2,8 s	3,4 s
Speed Index (PSI, mobile)	3,2 s	3,0 s
CLS (PSI, mobile)	0,005	0,005
Accessibility	95 / 100 / 100	95 / 100 / 100

Sumber: Hasil pengujian GTmetrix dan Google PageSpeed Insights, 22 Juni 2026.

Pada pengujian GTmetrix, Time to First Byte (TTFB) Vercel tercatat sebesar 220 ms, lebih rendah dibandingkan Railway yang mencapai 325 ms. Selisih sebesar 105 ms ini sejalan dengan karakteristik arsitektur edge serverless yang menempatkan titik respons lebih dekat dengan lokasi pengujian, sebagaimana dikemukakan oleh Setiawan dan Setiyadi dalam analisis komparatif performa server hosting [6]. Namun pada metrik First

Contentful Paint (FCP) dan Largest Contentful Paint (LCP) di kondisi desktop, kedua platform menunjukkan nilai yang relatif berimbang, dengan Railway bahkan mencatat LCP yang sedikit lebih stabil pada beberapa pengujian berulang (758–836 ms) dibandingkan Vercel yang menunjukkan variasi lebih lebar (507–979 ms).

C. Pembahasan Perbedaan Kinerja

Temuan yang menarik justru muncul pada pengujian Google PageSpeed Insights dengan emulasi perangkat mobile dan throttling jaringan Slow 4G. Pada kondisi ini, Railway memperoleh skor performa sebesar 91, sementara Vercel memperoleh skor 87. Largest Contentful Paint pada Railway tercatat 2,8 detik, sedangkan pada Vercel mencapai 3,4 detik. Hasil ini berbeda dari ekspektasi umum bahwa arsitektur serverless dengan CDN global akan selalu mengungguli hosting konvensional pada seluruh skenario pengujian.

Fenomena ini dapat dijelaskan melalui beberapa faktor teknis. Pertama, keunggulan TTFB pada arsitektur serverless umumnya paling terasa pada kondisi warm state, sementara pada simulasi mobile dengan throttling jaringan yang ketat, overhead pemrosesan rendering di sisi klien dan ukuran payload memiliki pengaruh yang lebih dominan terhadap LCP dibandingkan kecepatan respons awal server. Golec et al. menegaskan bahwa performa fungsi serverless sangat dipengaruhi oleh kondisi eksekusi, termasuk kemungkinan inisialisasi ulang (cold start) yang dapat menimbulkan variasi latensi meskipun fungsi telah pernah dipanggil sebelumnya, bergantung pada kebijakan idle timeout penyedia layanan [8]. Kedua, hosting konvensional yang berjalan secara always-on memberikan konsistensi eksekusi server-side rendering yang lebih dapat diprediksi karena tidak ada proses inisialisasi tambahan yang harus dilalui pada setiap permintaan.

Pada sisi structure score dan kepatuhan terhadap best practices, kedua platform menunjukkan hasil yang hampir identik, dengan Best Practices dan SEO masing-masing mencapai skor 100 pada kedua platform. Hal ini mengindikasikan bahwa kualitas implementasi kode aplikasi memberikan kontribusi yang signifikan terhadap skor performa, dan bukan semata-mata ditentukan oleh pilihan infrastruktur hosting. Total Blocking Time (TBT) pada kedua

platform juga berada pada rentang yang sebanding, yaitu 96–114 ms untuk Railway dan 96–142 ms untuk Vercel, yang menunjukkan bahwa beban pemrosesan JavaScript di sisi klien tidak banyak dipengaruhi oleh pemilihan arsitektur backend. Temuan ini mendukung pandangan Abraham dan Yang yang menekankan bahwa perbandingan performa antara layanan serverless dan server-based perlu memperhitungkan konteks beban kerja spesifik, bukan generalisasi tunggal [9], serta selaras dengan temuan tinjauan sistematis Ghattas, Odeh, dan Mora yang menunjukkan bahwa kurangnya standardisasi protokol evaluasi performa web menyebabkan variasi hasil pengukuran antar alat, meskipun umumnya mengacu pada kerangka kerja Lighthouse dan Core Web Vitals yang sama [10].

D. Implikasi untuk E-Commerce

Hasil pengujian ini memberikan implikasi praktis bagi pengelola platform e-commerce dalam menentukan strategi infrastruktur hosting. Bagi platform dengan basis pengguna yang terkonsentrasi secara geografis dan trafik yang relatif stabil, hosting konvensional seperti Railway tetap menjadi pilihan yang kompetitif, karena terbukti memberikan performa mobile yang lebih baik dalam pengujian ini serta konsistensi TTFB yang dapat diprediksi tanpa risiko cold start. Sebaliknya, arsitektur serverless seperti Vercel tetap relevan untuk skenario dengan pengguna yang tersebar secara global, mengingat keunggulan TTFB pada pengujian desktop berbasis lokasi pengujian yang sama, serta kemampuan skalabilitas otomatis yang sangat bermanfaat pada kondisi lonjakan trafik mendadak, sebagaimana ditunjukkan oleh Fujiyanti et al. dalam evaluasi performa layanan serverless pada Google Cloud Platform [11] dan Pratama et al. dalam perbandingan layanan container serverless lintas penyedia cloud [12].

Mengingat bahwa selisih skor performa antara kedua platform pada penelitian ini relatif tidak besar, faktor non-teknis seperti efisiensi biaya operasional, kemudahan pengelolaan deployment, serta dukungan terhadap teknologi front-end yang digunakan (dalam hal ini Next.js) tetap perlu menjadi pertimbangan utama. Bagi platform e-commerce skala kecil hingga menengah dengan anggaran terbatas, model pay-as-you-go pada

arsitektur serverless dapat lebih efisien dibandingkan biaya tetap bulanan pada hosting konvensional, meskipun harus memperhitungkan potensi penurunan skor performa mobile sebagaimana ditemukan dalam penelitian ini.

E. Limitasi Penelitian

Penelitian ini memiliki beberapa limitasi yang perlu diperhatikan dalam menginterpretasikan hasil. Pertama, penggunaan Railway sebagai representasi "hosting konvensional" memiliki keterbatasan konseptual, karena Railway merupakan platform cloud modern berbasis container orchestration dan managed infrastructure, bukan shared hosting atau VPS tradisional yang menjadi acuan pembahasan pada bagian Pendahuluan; perbandingan yang dihasilkan lebih tepat dimaknai sebagai perbandingan antara model eksekusi always-on dan serverless, bukan antara hosting konvensional klasik dan cloud modern. Kedua, pengujian hanya dilakukan pada satu lokasi server pengujian (Seattle, Amerika Serikat), sehingga belum merepresentasikan variasi latensi yang mungkin terjadi pada lokasi geografis lain, khususnya wilayah Indonesia yang menjadi konteks utama industri e-commerce yang dibahas. Ketiga, pengujian dilakukan pada kondisi layanan yang aktif (warm state); pengujian cold start secara eksplisit dengan periode idle yang terukur belum dilaksanakan, sehingga penjelasan mengenai pengaruh cold start terhadap perbedaan performa pada bagian Pembahasan bersifat interpretatif berdasarkan kajian literatur dan belum didukung oleh data pengukuran langsung. Keempat, pengujian beban (load testing) dengan jumlah pengguna konkuren yang bervariasi serta analisis statistik inferensial (uji t atau Mann-Whitney U) belum dilakukan dalam tahap ini, sehingga perbandingan skalabilitas kedua arsitektur pada kondisi trafik tinggi masih bersifat teoritis berdasarkan kajian literatur dan signifikansi statistik dari perbedaan metrik yang diamati belum dapat dipastikan. Penelitian lanjutan disarankan untuk memperluas lokasi pengujian, menggunakan representasi shared hosting/VPS yang lebih sesuai untuk arsitektur konvensional, menambah jumlah replikasi pengukuran disertai uji statistik formal, serta melibatkan pengujian beban dan kondisi cold

start yang terkontrol guna memperoleh gambaran performa yang lebih komprehensif.

IV. KESIMPULAN

Penelitian ini berhasil mencapai tujuannya untuk menganalisis dan membandingkan kecepatan akses *website e-commerce* antara arsitektur *web hosting* konvensional (Railway) dan *cloud serverless* terdistribusi (Vercel) melalui pengujian sistematis menggunakan GTmetrix dan Google PageSpeed Insights. Hasil pengujian menunjukkan bahwa tidak ada satu arsitektur yang secara mutlak superior di seluruh skenario, melainkan keduanya menunjukkan keunggulan yang bersifat kontekstual.

Pada simulasi kondisi desktop (GTmetrix), arsitektur *serverless* (Vercel) menunjukkan keunggulan pada metrik TTFB (220 ms dibanding 325 ms pada Railway), konsisten dengan karakteristik distribusi *edge network* yang lebih dekat dengan lokasi pengujian. Namun pada metrik FCP dan LCP, kedua platform menunjukkan performa yang relatif seimbang, dengan Railway justru mencatat stabilitas LCP yang sedikit lebih baik pada pengujian berulang. Temuan yang lebih mengejutkan muncul pada simulasi kondisi mobile dengan *throttling* Slow 4G (Google PageSpeed Insights), di mana hosting konvensional (Railway) justru mengungguli arsitektur *serverless*, dengan skor performa 91 berbanding 87 dan LCP 2,8 detik berbanding 3,4 detik pada Vercel. Hasil ini membantah asumsi umum bahwa *cloud serverless* dengan CDN global akan selalu mengungguli hosting konvensional pada seluruh skenario, dan menunjukkan bahwa keunggulan TTFB tidak selalu berbanding lurus dengan performa *rendering* akhir, terutama pada kondisi jaringan yang terbatas.

Dari sisi *best practices*, SEO, dan *Total Blocking Time*, kedua platform menunjukkan hasil yang hampir identik, mengindikasikan bahwa kualitas implementasi kode aplikasi turut berperan signifikan terhadap performa keseluruhan, tidak semata-mata ditentukan oleh pilihan infrastruktur hosting.

Berdasarkan temuan tersebut, pemilihan arsitektur hosting bagi platform *e-commerce* sebaiknya disesuaikan dengan karakteristik trafik dan basis pengguna. Hosting konvensional seperti Railway tetap menjadi pilihan kompetitif bagi platform dengan basis pengguna terkonsentrasi

secara geografis serta trafik yang relatif stabil, mengingat konsistensi performa mobile dan TTFB yang dapat diprediksi tanpa risiko *cold start*. Sebaliknya, arsitektur *serverless* seperti Vercel tetap relevan bagi platform dengan pengguna yang tersebar secara global serta kebutuhan skalabilitas otomatis pada lonjakan trafik mendadak, meskipun fenomena *cold start* tetap menjadi faktor risiko yang perlu dimitigasi melalui strategi seperti *scheduled warming* atau *provisioned concurrency*.

Penelitian ini memiliki sejumlah keterbatasan, antara lain pengujian yang hanya dilakukan dari satu lokasi server (Seattle, Amerika Serikat), belum dilakukannya pengujian *cold start* secara eksplisit dengan periode *idle* terukur, serta belum dilaksanakannya pengujian beban (*load testing*) pada berbagai jumlah pengguna konkuren. Oleh karena itu, perbandingan skalabilitas kedua arsitektur pada kondisi trafik tinggi masih bersifat teoritis berdasarkan kajian literatur. Penelitian lanjutan disarankan untuk memperluas lokasi pengujian khususnya pada wilayah Indonesia sebagai konteks utama industri *e-commerce* yang dibahas menambah jumlah replikasi pengukuran, melibatkan pengujian beban dan kondisi *cold start* yang terkontrol, serta mengeksplorasi pendekatan arsitektur hibrid yang menggabungkan konsistensi hosting konvensional dengan skalabilitas elastis arsitektur *serverless*.

UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih yang sebesar-besarnya kepada Ibu Agustina Srirahayu, M.Kom. selaku Dosen Pengampu mata kuliah Sistem Terdistribusi atas bimbingan, arahan, dan masukan yang sangat berharga selama proses penelitian ini berlangsung. Ucapan terima kasih juga disampaikan kepada seluruh dosen, staf, dan civitas akademika Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Duta Bangsa yang telah memberikan dukungan akademik dan fasilitas selama perkuliahan. Tidak lupa, penulis mengucapkan terima kasih kepada keluarga dan rekan-rekan mahasiswa atas doa, motivasi, serta kontribusi pemikiran yang membantu kelancaran penelitian ini. Semoga hasil penelitian ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan praktik di bidang sistem terdistribusi.

REFERENSI

- [1] ResearchAndMarkets.com, "Indonesia E-Commerce Business Report 2024: Gross Merchandise Value will Increase from \$40.4 Billion in 2023 to Reach \$63.2 Billion by 2028," BusinessWire, Sep. 2024. [Online]. Available: <https://www.businesswire.com/news/home/20240904648772>
- [2] Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), "Survei Penetrasi Internet Indonesia 2024," APJII, Jan. 2024. [Online]. Available: <https://apjii.or.id>
- [3] J. van Riet, I. Malavolta, and T. A. Ghaleb, "Optimise Along the Way: An Industrial Case Study on Web Performance," *Journal of Systems and Software*, vol. 198, Article 111593, 2023. [Online]. Available: <https://doi.org/10.1016/j.jss.2022.111593>
- [4] S. Pati and Y. Zaki, "Evaluating the Efficacy of Next.js: A Comparative Analysis with React.js on Performance, SEO, and Global Network Equity," *arXiv preprint arXiv:2502.15707*, Feb. 2025. [Online]. Available: <https://arxiv.org/abs/2502.15707>
- [5] A. M. Sundjaja, A. V. Tatuil, D. V. Scholus, and Y. D. Restiani, "The Determinant Factors of Shopping Cart Abandonment Among E-commerce Customers in Indonesia," *CommIT (Communication and Information Technology) Journal*, vol. 18, no. 1, pp. 29–38, Apr. 2024. [Online]. Available: <https://doi.org/10.21512/commit.v18i1.9308>
- [6] E. B. Setiawan and A. Setiyadi, "Comparative Analysis of Web Hosting Server Performance," *International Journal of Engineering*, vol. 36, no. 3, pp. 558–564, 2023. [Online]. Available: <https://doi.org/10.5829/ije.2023.36.03c.16>
- [7] J. Wen, Z. Chen, X. Jin, and X. Liu, "Rise of the Planet of Serverless Computing: A Systematic Review," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 5, Article 131, Jul. 2023. [Online]. Available: <https://doi.org/10.1145/3579643>
- [8] M. Golec, G. K. Walia, M. Kumar, F. Cuadrado, S. S. Gill, and S. Uhlig, "Cold Start Latency in Serverless Computing: A Systematic Review, Taxonomy, and Future Directions," *ACM Computing Surveys*, vol. 57, no. 3, pp. 65:1–65:36, 2025. [Online]. Available: <https://doi.org/10.1145/3700875>
- [9] J. Abraham and J. Yang, "A Comparative Analysis of Performance and Usability on Serverless and Server-Based Google Cloud Services," in *Proc. Int. Conf. Advances in Computing Research (ACR'23)*, Lecture Notes in Networks and Systems, pp. 408–422, 2023. [Online]. Available: https://doi.org/10.1007/978-3-031-33743-7_33
- [10] M. Ghattas, S. Odeh, and A. M. Mora, "Predicting Website Performance: A Systematic Review of Metrics, Methods, and Research Gaps (2010–2024)," *Computers*, vol. 14, no. 10, Article 446, Oct. 2025. [Online]. Available: <https://doi.org/10.3390/computers14100446>
- [11] V. Fujiyanti, G. M. Suranegara, and I. N. Ichsan, "Comparative Analysis of Server-Based and Serverless Service Performance on Google Cloud Platform (GCP) (Case Study: Machine Learning Model Deployment)," *Journal of Information Systems and Informatics*, vol. 6, no. 2, pp. 1172–1194, 2024. [Online]. Available: <https://doi.org/10.51519/journalisi.v6i2.773>
- [12] M. A. Pratama, O. Nurdianawan, A. R. Dikananda, D. Pratama, and D. A. Kumia, "Comparative Analysis of Serverless Container Service Performance Between Google Cloud Run and AWS App Runner in Cross-Cloud Architecture," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 5, no. 2, pp. 2531–2539, 2026. [Online]. Available: <https://ioinformatic.org/index.php/JAIEA/article/view/1919>