

Analisis Perbandingan Kinerja dan Penggunaan Energi pada Framework React dan Vue

Comparative Analysis of Performance and Energy Consumption in React and Vue Frameworks

Indra Wijaya¹, Ahmad Farisi²

^{1,2}Program Studi Sistem Informasi, Universitas Multi Data Palembang

E-mail: ¹indra124@mhs.mdp.ac.id, ²ahmadfarisi@mdp.ac.id

Abstrak

Pengembangan website yang optimal membutuhkan pemilihan framework yang tepat untuk meningkatkan kinerja dan efisiensi. Penelitian ini bertujuan untuk membandingkan kinerja dan konsumsi energi *framework* React dan Vue, menggunakan metrik seperti FCP, LCP, SI, TBT, CLS dengan menggunakan Google Lighthouse. Kinerja rendering time, dan penggunaan memori menggunakan Google DevTools. Konsumsi energi diukur dengan menggunakan tools Globemallow. Penelitian ini menggunakan metode pendekatan kuantitatif dengan metode *true-experimental research*. Metode ini dipilih karena memungkinkan pengukuran sistematis dan objektif melalui variabel-variabel dalam kondisi terkontrol, sesuai tujuan penelitian. Eksperimen dimulai dari mengembangkan halaman *website* kemudian menyiapkan perangkat keras setelah itu melaksanakan skenario pengujian kemudian menganalisis dan menginterpretasikan data yang didapatkan. Hasil penelitian menunjukkan bahwa Vue memiliki kinerja lebih unggul berdasarkan pengujian kinerja *rendering time* dan kinerja FCP, LCP, TBT, CLS, dan SI, sementara React lebih efisien dalam penggunaan memori, dan konsumsi energi. Temuan ini memberikan rekomendasi *framework* yang optimal berdasarkan kebutuhan kinerja dan keberlanjutan untuk pengembangan aplikasi web.

Kata kunci: React, Vue, Kinerja Website, Penggunaan Energi, Google Lighthouse.

Abstract

The development of an optimal website requires selecting the right framework to enhance performance and efficiency. This study aims to compare the performance and energy consumption of the React and Vue frameworks using metrics such as FCP, LCP, SI, TBT, and CLS, measured with Google Lighthouse. Rendering time, and memory usage are evaluated using Google DevTools, while energy consumption is measured with Globemallow. The study employs a quantitative approach using the true-experimental research method, chosen because it allows systematic and objective measurements through variables in controlled conditions, aligned with the research objectives. The experiment begins with developing the website pages, setting up the hardware, executing the test scenarios, and then analyzing and interpreting the collected data. The results show that Vue outperforms React in rendering time performance and metrics such as FCP, LCP, TBT, CLS, and SI, while React is more efficient in memory usage, and energy consumption. These findings provide recommendations for selecting the optimal framework based on performance and sustainability needs for web application development.

Keywords: React, Vue, Website Performance, Energy Consumption, Google Lighthouse.

1. PENDAHULUAN

Di era digitalisasi saat ini, pengembangan *web* terus berkembang, sehingga *website* telah menjadi entitas digital yang mencakup berbagai halaman *web* yang saling terhubung dan menyertakan seluruh berkas yang dibutuhkan. Halaman-halaman tersebut biasanya disebut dengan 'halaman utama' dan bersama-sama membentuk suatu *website* yang lengkap [1].

Selain itu, pengembangan *website* juga menjadi salah satu elemen penting bagi perusahaan, organisasi, dan individu dalam menyediakan layanan berkualitas bagi pelanggan. Di era digital saat ini, *website* memiliki peran signifikan dalam menampilkan identitas dan menyebarkan informasi kepada audiens. Karena itu, pengembangan *website* sebaiknya dilakukan secara optimal dengan menggunakan teknologi terbaru, salah satunya menggunakan *framework* JavaScript [2].

Framework JavaScript adalah kerangka kerja sistematis yang siap pakai pada JavaScript, didukung oleh berbagai *library*, yang memudahkan pengembang dalam membangun *website* berbasis JavaScript. Dengan *framework* ini, pengembang dapat membuat fungsi, prosedur, dan fitur lainnya menggunakan *library* yang tersedia, sehingga tidak perlu menulis ulang kode dari awal [3]. *Framework* Javascript seperti React dan Vue merupakan dua *framework* JavaScript yang sangat populer dan sering dipakai oleh *web developer*. Keduanya memiliki fitur serta keunggulan masing-masing, sehingga sering kali sulit untuk menentukan mana *framework* yang paling sesuai untuk *website development* [2].

React adalah JavaScript *library* yang digunakan untuk membangun komponen antarmuka pengguna yang dapat dipakai ulang. Menurut dokumentasi dari ReactJS, React sangat cocok sebagai dasar untuk mengembangkan *single-page web applications*. Secara keseluruhan, React memungkinkan pengembangan aplikasi berbasis *web* yang besar dan kompleks, di mana data dapat diperbarui tanpa harus memuat ulang halaman sepenuhnya untuk menampilkan halaman berikutnya [4]. Sementara itu, Vue adalah *progressive* JavaScript *framework* yang mudah diakses, fleksibel, dan dapat diintegrasikan dengan pustaka atau proyek yang sudah ada. Dengan pendekatan adopsi bertahap, Vue sangat dihargai atas kesederhanaan dan fleksibilitasnya, menjadikannya pilihan yang sangat baik untuk proyek berskala kecil hingga aplikasi besar dan rumit [5]. Oleh karena itu, pemilihan *framework* dalam pengembangan aplikasi *web* menjadi sangat penting karena akan mempengaruhi kinerja aplikasi secara keseluruhan [1].

Beberapa penelitian terdahulu telah melakukan pengujian terhadap *framework-framework* JavaScript, seperti [6] yang melakukan studi komparatif terhadap *framework* React, Angular, Vue, Svelte dan SolidJS dengan studi kasus *Weather App*. Kelima *framework* tersebut diuji berdasarkan metrik pengukuran *First Contentful Paint* (FCP), *Largest Contentful Paint* (LCP), *Time to Interactive* (TTI), *Speed Index* (SI), *Total Blocking Time* (TBT) dan *Cumulative Layout Shift* (CLS). Pengujian yang dilakukan dalam penelitian tersebut dibantu dengan menggunakan *tools* Google Lighthouse. Hasil dari penelitian tersebut menunjukkan bahwa Vue dapat memberikan kinerja terbaik saat diimplementasikan ke *Weather App*.

Penelitian [7] juga melakukan analisis perbandingan kinerja terhadap *framework* React, Vue dan Angular dengan metrik pengukuran FCP, TTI, TBT, *memory usage* dan *rendering*. Adapun *tools* yang digunakan untuk mengukur FCP, TTI, dan TBT adalah Google *Lighthouse*, sementara *memory usage* dan *rendering* diuji menggunakan Chrome DevTools. Hasil dari penelitian tersebut menunjukkan bahwa Vue mendapat penilaian tertinggi untuk kemudahan penggunaan dan produktivitas dengan kurva pembelajaran yang cukup mudah. Penelitian [3] juga melakukan analisis perbandingan terhadap React, Vue dan Angular dengan metrik pengukuran FCP, SI, LCP, TTI, TBT, dan CLS. *Tools* yang digunakan untuk membantu pengukuran adalah Google *Lighthouse*. Hasil dari penelitian tersebut menunjukkan bahwa React lebih direkomendasikan dikarenakan memiliki performa yang baik dibandingkan dengan *framework* Vue dan Angular.

Dari beberapa penelitian sebelumnya, terdapat penelitian yang menunjukkan perbandingan kinerja *framework* JavaScript. Penelitian yang dilakukan oleh [7] menunjukkan bahwa Vue memiliki kinerja yang lebih baik dibandingkan dengan *framework* React dan Angular. Pengujian dilakukan berdasarkan metrik FCP, TTI, TBT, *memory usage* dan *rendering time*. Namun penelitian ini tidak mengukur metrik LCP, SI, CLS seperti yang dilakukan oleh [6], yang juga menunjukkan bahwa Vue lebih baik daripada *framework* React, Angular, Svelte, SolidJS dalam metrik FCP, LCP, TTI, SI, TBT, CLS.

Selain kinerja, penggunaan energi kini menjadi faktor penting dalam pemilihan *framework* *web*, terutama dengan adanya perhatian terhadap keberlanjutan teknologi. Dengan meningkatnya

kebutuhan akan sebuah aplikasi yang ramah lingkungan, penggunaan energi *framework* menjadi salah satu aspek krusial dalam pengembangan *web* berkelanjutan [8]. Penggunaan energi merujuk pada total energi yang digunakan oleh sebuah aplikasi. Satuan yang dipakai untuk mengukur penggunaan energi adalah miliwatt-jam, yang menunjukkan jumlah energi yang dipakai oleh suatu proses atau sistem dalam jangka waktu tertentu [9].

Pengukuran penggunaan energi website pernah dilakukan oleh [8] dengan judul penelitian “*Green and Sustainable JavaScript*” yang mengukur penggunaan energi yang dibantu dengan *tools* yang bernama GreenFrame. Parameter pengukuran penggunaan energi dalam penelitian tersebut adalah jejak karbon dan penggunaan energi. Jejak karbon diukur dalam gram ekivalen karbon dioksida (g eq. CO₂) dan penggunaan energi diukur dalam satuan miliwatt-jam (mWh). Penelitian tersebut mengukur jumlah emisi karbon yang dihasilkan oleh aplikasi dan total energi yang digunakan saat aplikasi berjalan. Sedangkan penelitian [10] meneliti bagaimana cara untuk mengurangi emisi dengan cara meningkatkan efisiensi energi situs, yang dilakukan dengan menggunakan *tools* bernama Beacon, yang dapat mengetimasi emisi karbon berdasarkan kunjungan pertama *website*, ukuran *file* yang dimuat oleh halaman *web*, jumlah *request* yang dikirim browser, elemen visual seperti gambar, *font*, *stylesheet* dan *script*.

Untuk membuktikan beberapa penelitian sebelumnya yang mengungkapkan adanya perbedaan kinerja dan penggunaan energi di antara berbagai *framework web*, penelitian ini menguji dua *framework JavaScript*, yaitu React dan Vue. Melalui pengujian kinerja masing-masing *framework* akan diuji berdasarkan metrik utama dalam waktu muat konten seperti FCP, LCP, SI, TBT, CLS, serta *rendering time*, *memory usage*, dan penggunaan energi.

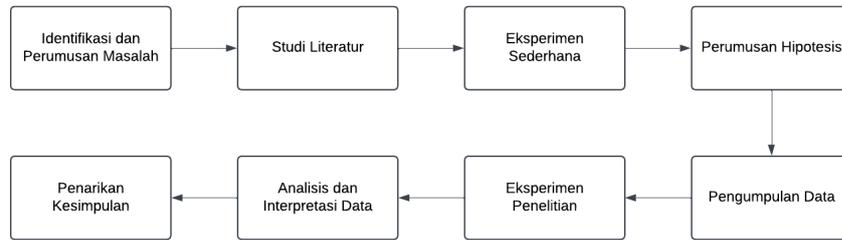
2. METODE PENELITIAN

Penelitian ini menggunakan metodologi *True-Experimental* untuk menentukan *framework* yang paling efektif dalam meningkatkan kinerja website. Metode ini dipilih karena kemampuannya dalam mengontrol variabel-variabel penelitian secara ketat, sehingga memungkinkan pengujian yang lebih akurat terhadap pengaruh berbagai gabungan variabel terhadap metrik kinerja website, termasuk FCP, LCP, SI, TBT, dan CLS. Selain itu, penelitian ini juga mengevaluasi aspek lain seperti *memory usage*, *rendering time* untuk mendapatkan analisis yang komprehensif terkait performa masing-masing *framework*.

Sebagai tambahan, penelitian ini juga menilai efisiensi penggunaan energi dari setiap *framework* dengan mengukur jejak karbon yang dihasilkan serta konsumsi energi dalam satuan kWh. Pengukuran dilakukan menggunakan alat khusus yang dirancang untuk menganalisis konsumsi daya secara mendetail. Dengan pendekatan ini, penelitian tidak hanya berfokus pada kinerja teknis tetapi juga mempertimbangkan dampak lingkungan yang dihasilkan oleh setiap *framework*. Pendekatan berbasis *True Experimental* memungkinkan penelitian ini untuk memberikan hasil yang objektif dan sistematis, sehingga dapat menjadi acuan bagi pengembang dalam memilih *framework* yang optimal, baik dari segi kinerja maupun keberlanjutan lingkungan. Hal ini memberikan kontribusi penting dalam pengembangan aplikasi web yang tidak hanya efektif tetapi juga ramah lingkungan.

2.1 Tahapan Review

Tahapan dari penelitian ini dapat dilihat pada gambar 1 Berikut.



Gambar 1 Tahapan Penelitian

2.2 Ruang Lingkup Penelitian

Penelitian ini memiliki ruang lingkup yang jelas dengan batasan masalah yang telah ditentukan. Fokus utama penelitian ini adalah mengukur aspek kinerja framework dalam pengembangan website, yang meliputi rendering time, memory usage, serta penggunaan energi. Penggunaan energi yang dimaksud mencakup pengukuran *emisi karbon* dan konsumsi energi selama proses pengujian. Data yang digunakan dalam pengujian terdiri dari 100 data dummy produk yang merepresentasikan halaman produk pada sebuah website, dengan informasi berupa nama produk, harga, dan gambar. Penelitian ini juga menetapkan metrik pengukuran utama kinerja website, termasuk *FCP (First Contentful Paint)*, *LCP (Largest Contentful Paint)*, *SI (Speed Index)*, *TBT (Total Blocking Time)*, *CLS (Cumulative Layout Shift)*, serta *rendering time*, *memory usage* sebagai indikator penting dalam evaluasi performa.

Pengujian dilakukan menggunakan satu perangkat keras dengan spesifikasi tertentu untuk memastikan hasil yang konsisten dan terstandarisasi. Perangkat keras yang digunakan memiliki spesifikasi sebagai berikut: prosesor Intel(R) Core(TM) i7-7700HQ CPU @2.80GHz, RAM sebesar 20.00 GB DDR4, dan sistem operasi Linux Ubuntu 24.04.1 LTS. Penelitian ini juga membatasi ruang lingkungannya dengan mengecualikan beberapa variabel tambahan, seperti efisiensi jaringan, keamanan, dan optimasi kode khusus, yang tidak dibahas dalam penelitian ini. Batasan-batasan tersebut ditetapkan untuk menjaga fokus penelitian agar tetap terarah dan mendalam sesuai dengan tujuan yang telah dirumuskan.

2.3 Framework Eksperimen

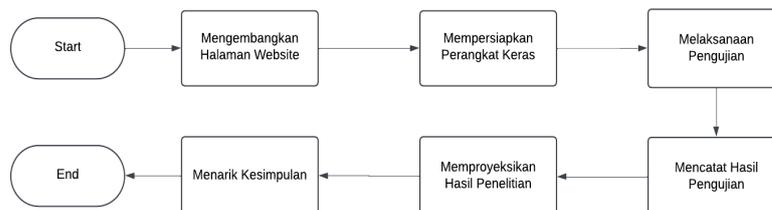
Framework-framework yang digunakan untuk setiap metrik pengujian dalam eksperimen ini dapat dilihat pada Tabel 1 berikut.

Tabel 1 Framework Eksperimen

Framework	Performance	Rendering Time	Memory Usage	Energy Usage
React	Google Lighthouse	Google DevTools	Google DevTools	Globem allow
Vue	Google Lighthouse	Google DevTools	Google DevTools	Globem allow

2.4 Alur Eksperimen

Alur dari keseluruhan dari penelitian ini dapat dilihat pada Gambar 2 berikut ini.



Gambar 2 Alur Eksperimen

Eksperimen ini dimulai dengan mengembangkan halaman *website* dengan menggunakan *framework* pengujian yaitu React dan Vue. Setelah halaman *website* selesai dikembangkan, kemudian menyiapkan perangkat keras yang akan digunakan untuk pengujian. Setelah menyiapkan perangkat keras, pengujian dilakukan sebanyak 10 kali untuk mengevaluasi kinerja dari *framework* serta melakukan pengujian penggunaan energi. Kemudian data dari pengujian tersebut dikumpulkan sebagai hasil pengujian, yang kemudian di proyeksikan dalam bentuk diagram-diagram yang mudah dibaca untuk memberikan wawasan terhadap perbandingan dari kinerja dan penggunaan energi dari *framework* yang diuji. Langkah terakhir adalah penarikan kesimpulan berdasarkan hasil data tersebut.

3. HASIL DAN PEMBAHASAN

3.1 Identifikasi dan Perumusan Masalah

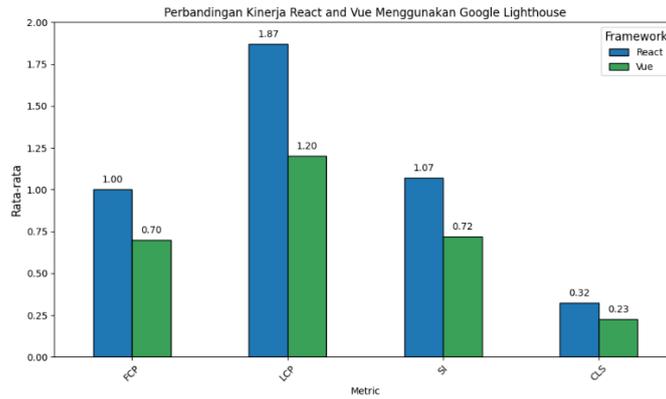
Pada tahap ini, dilakukan identifikasi masalah seperti pemilihan *framework* yang dapat mempengaruhi kinerja *website*, penggunaan sumber daya, dan penggunaan energi. Oleh karena itu, penelitian ini berfokus menganalisis perbandingan kinerja dan penggunaan energi dari dua *framework* JavaScript, React dan Vue terutama dalam hal, kinerja *website menggunakan* Google Lighthouse, *rendering time*, *memory usage*.

3.2 Studi Literatur

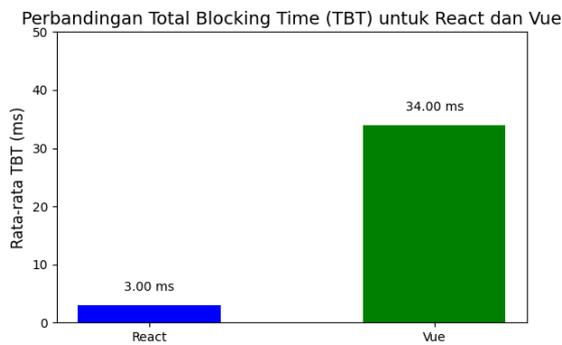
Tahapan ini melakukan kajian literatur dari artikel jurnal dan buku-buku yang berasal dari sumber-sumber yang ilmiah. Kajian ini membantu mengidentifikasi *framework* JavaScript yang digunakan untuk mengukur kinerja situs web dalam pengembangan aplikasi web saat ini, serta metode pengujian yang digunakan untuk memperoleh hasil yang relevan dan valid sesuai dengan permasalahan yang telah diidentifikasi dan dirangkum dalam rumusan masalah.

3.3 Eksperimen Sederhana

Tahapan ini melakukan eksperimen sederhana dengan mengembangkan aplikasi *website* yang menggunakan *framework* React dan Vue untuk dilakukan pengujian. Hasil dari eksperimen sederhana ini digunakan untuk merumuskan hipotesis penelitian. Berikut hasil dari eksperimen sederhana yang telah dilakukan dapat dilihat dari Gambar 3 sampai Gambar 8.

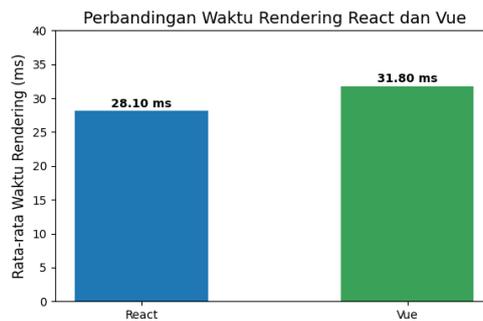


Gambar 3 Perbandingan Kinerja Metrik FCP, LCP, SI, CLS Framework React dan Vue Menggunakan Google Lighthouse



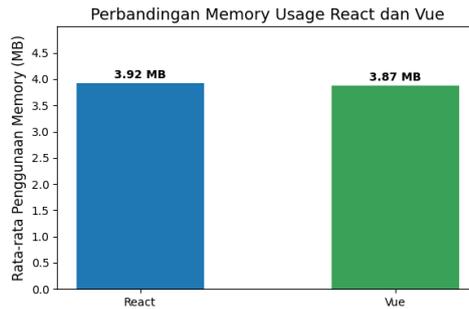
Gambar 4 Perbandingan Kinerja TBT Framework React dan Vue Menggunakan Google Lighthouse

Pada Gambar 3 dan Gambar 4 menunjukkan hasil pengujian kinerja metrik FCP, LCP, SI, CLS dan TBT menunjukkan bahwa *framework* Vue lebih unggul dari *framework* React.



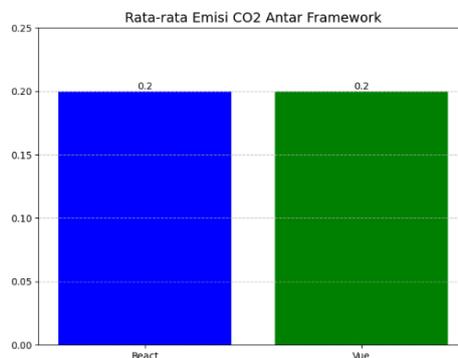
Gambar 5 Perbandingan Kinerja Metrik *Rendering Time* Framework React dan Vue Menggunakan Google DevTools

Pada Gambar 5 menunjukkan hasil pengujian kinerja *rendering time* menunjukkan bahwa *framework* Vue lebih unggul dari *framework* React.

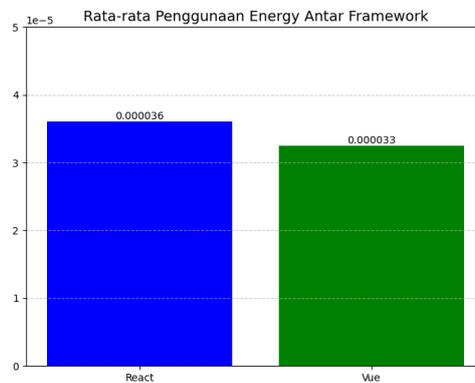


Gambar 6 Perbandingan Kinerja Metrik *Memory Usage Framework* React dan Vue Menggunakan Google DevTools

Pada Gambar 6 menunjukkan hasil pengujian kinerja *memory usage* menunjukkan bahwa *framework* Vue lebih unggul dari *framework* React.



Gambar 7 Perbandingan Metrik Emisi CO₂ *Framework* React dan Vue Menggunakan Globemallow



Gambar 8 Perbandingan Metrik Penggunaan Energi *Framework* React dan Vue Menggunakan Globemallow

Pada Gambar 7 dan Gambar 8 menunjukkan hasil pengujian dari Penggunaan energy dan emisi karbon yang dihasilkan. Rata-rata dari penggunaan energi menunjukkan bahwa *framework* Vue lebih unggul dari *framework* React. Namun pada rata-rata metrik emisi karbon menunjukkan bahwa *framework* React dan Vue itu setara.

3.4 Perumusan Hipotesis

Dari hasil eksperimen sederhana yang telah dilakukan sebelumnya, penelitian ini membangun hipotesis sebagai berikut.

- Hipotesis 1 (H₁): Framework Vue memiliki kinerja yang lebih unggul dibandingkan React berdasarkan metrik utama seperti FCP, LCP, SI, CLS, dan TBT.
- Hipotesis 2 (H₂): Framework Vue memiliki efisiensi yang lebih unggul dalam *memory usage* dibandingkan React.
- Hipotesis 3 (H₃): Framework React memiliki *rendering time* yang lebih cepat dibandingkan dengan Vue.
- Hipotesis 4 (H₄): Framework Vue lebih efisien dalam penggunaan energi dibandingkan dengan React.

3.5 Pengumpulan Data

Data yang akan digunakan dalam penelitian ini merupakan data dummy. Data dummy ini mencakup beberapa atribut penting sebagai berikut:

1. id: berupa string yang berfungsi sebagai identitas unik untuk setiap produk,
2. name: berupa string yang merepresentasikan nama atau label dari produk,
3. category: berupa string yang menunjukkan kategori utama dari produk,
4. sub_description: berupa string yang memberikan deskripsi tambahan mengenai subkategori produk.
5. price: berupa integer yang merepresentasikan harga produk.
6. id_gambar: berupa string yang digunakan sebagai referensi untuk menghubungkan produk dengan gambar terkait.

3.6 Eksperimen Penelitian

Hasil dari eksperimen yang telah dilakukan ditunjukkan pada Tabel 2 hingga Tabel 5 .

Tabel 2 Pengujian Kinerja Framework React pada FCP, LCP, TBT, CLS, SI Menggunakan Google Lighthouse

React					
Tes	FCP	LCP	TBT	CLS	SI
1	1.0s	1.9s	120ms	0	1.2s
2	1.1s	2.0s	100ms	0	1.2s
3	1.1s	2.0s	120ms	0	1.2s
4	1.1s	2.0s	110ms	0	1.2s
5	1.1s	2.0s	110ms	0	1.2s
6	1.0s	1.9s	100ms	0	1.2s
7	1.0s	2.0s	100ms	0	1.2s
8	1.1s	2.0s	110ms	0	1.2s
9	1.0s	1.9s	100ms	0	1.2s
10	1.1s	2.0s	110ms	0	1.2s

Rata-rata hasil dari pengujian dari *framework* React yang telah dilakukan sebagai berikut:

- FCP (First Contentful Paint): 1.08s
- LCP (Largest Contentful Paint): 1.96s
- TBT (Total Blocking Time): 108ms
- CLS (Cumulative Layout Shift): 0
- SI (Speed Index): 1.2s

Tabel 3 Pengujian Kinerja Framework Vue pada FCP, LCP, TBT, CLS, SI Menggunakan Google Lighthouse

Vue					
Tes	FCP	LCP	TBT	CLS	SI
1	0.8s	1.5s	160ms	0	1..0s
2	0.8s	1.5s	170ms	0	1..0s
3	0.8s	1.5s	170ms	0	1..0s
4	0.8s	1.5s	180ms	0	1..0s
5	0.8s	1.6s	150ms	0	1..0s
6	0.8s	1.5s	170ms	0	1..0s
7	0.8s	1.5s	170ms	0	1..0s
8	0.8s	1.5s	180ms	0	1..0s
9	0.8s	1.5s	170ms	0	1..0s
10	0.8s	1.6s	170ms	0	1..0s

Rata-rata hasil dari pengujian dari *framework* Vue yang telah dilakukan sebagai berikut:

- FCP (First Contentful Paint): 0.8s
- LCP (Largest Contentful Paint): 1.52s
- TBT (Total Blocking Time): 171ms
- CLS (Cumulative Layout Shift): 0
- SI (Speed Index): 1.0s

Tabel 4 Pengujian Kinerja *Memory Usage* Framework React dan Vue Menggunakan Google DevTools

Tes	React	Vue
1	7mb	7.2mb
2	7mb	7.2mb
3	7mb	7.2mb
4	7.1mb	7.2mb
5	7.1mb	7.5mb
6	7.1mb	7.3mb
7	7.1mb	7.3mb
8	7.4mb	7.3mb
9	7.4mb	7.3mb
10	7.4mb	7.3mb

Rata-rata dari hasil rendering time yang didapatkan Framework React adalah 7.16 m sedangkan Vue 82.8ms.

Tabel 5 Pengujian Kinerja *Rendering Time* Framework React dan Vue Menggunakan Google DevTools

Tes	React	Vue
1	103ms	71ms
2	106ms	83ms
3	109ms	88ms
4	101ms	84ms
5	99ms	90ms

6	115ms	91ms
7	111ms	82ms
8	105ms	83ms
9	98ms	86ms
10	107ms	70ms

Rata-rata dari hasil rendering time yang didapatkan Framework React adalah 7.16 m sedangkan Vue 82.8ms.

Tabel 6 Pengujian Penggunaan Energi Framework React dan Vue Menggunakan Globemallow

Tes	React	Vue
1	0.000046	0.000049
2	0.000037	0.000036
3	0.000035	0.000038
4	0.000036	0.000021
5	0.000034	0.000034
6	0.000037	0.000034
7	0.000034	0.000021
8	0.000034	0.000036
9	0.000034	0.000035
10	0.000034	0.000021

Rata-rata dari hasil penggunaan energi yang didapatkan Framework React adalah 0.000021 kWh sedangkan Vue 0.0000501 kWh.

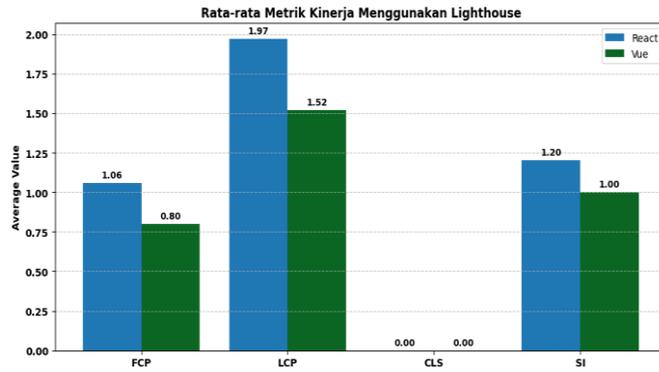
Tabel 7 Pengujian Emisi Karbon Framework React dan Vue Menggunakan Globemallow

Tes	React	Vue
1	0.02	0.02
2	0.02	0.02
3	0.02	0.02
4	0.02	0.02
5	0.02	0.02
6	0.02	0.02
7	0.02	0.02
8	0.02	0.02
9	0.02	0.02
10	0.02	0.02

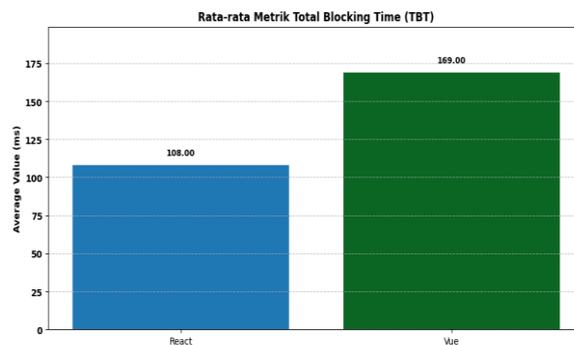
Rata-rata dari hasil perbandingan emisi karbon yang didapatkan Framework React adalah 0.01g CO₂a sedangkan Vue 0.022g CO₂.

3.5 Analisis dan Interpretasi Data

Berdasarkan data yang diperoleh dari eksperimen penelitian pada tahapan sebelumnya, dilakukan analisis dalam bentuk diagram untuk mempermudah pembacaan dan interpretasi data tersebut.

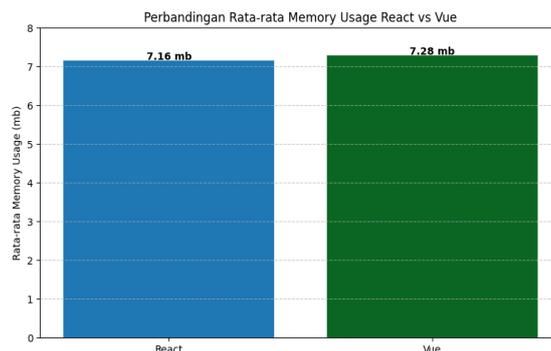


Gambar 9 Perbandingan Kinerja Metrik FCP, LCP, CLS, SI pada *framework* React dan Vue Menggunakan Google Lighthouse



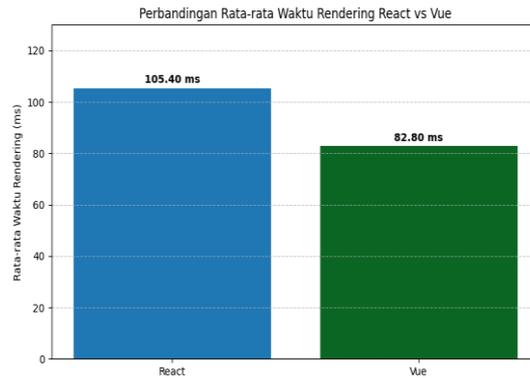
Gambar 10 Perbandingan Kinerja Metrik TBT pada *framework* React dan Vue Menggunakan Google Lighthouse

Berdasarkan Gambar 9 dan Gambar 10, *framework* Vue menunjukkan kinerja yang lebih baik dibandingkan React dalam hal metrik FCP, LCP, dan SI. Hal ini menandakan bahwa Vue lebih unggul dalam memberikan pengalaman pemuatan halaman yang lebih cepat dan responsif kepada pengguna. Namun, pada metrik TBT, React menunjukkan performa yang lebih baik dibandingkan Vue, dengan selisih waktu sebesar 51 ms.



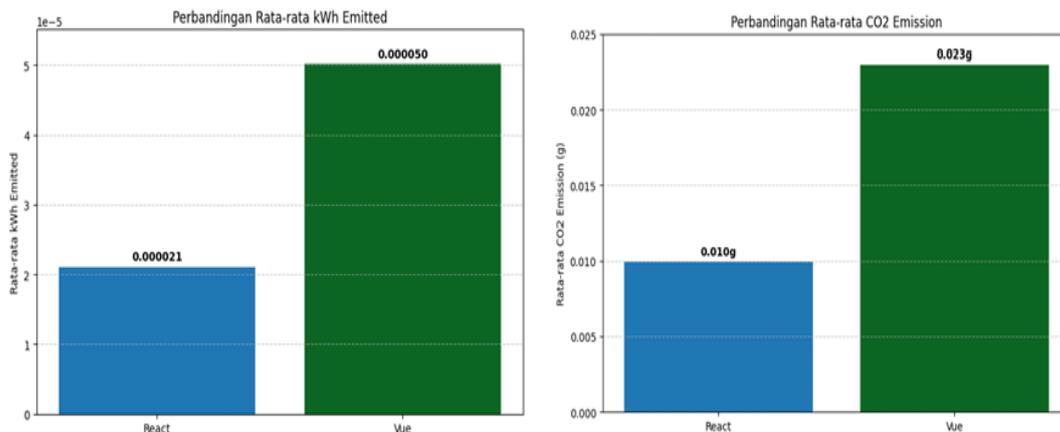
Gambar 11 Perbandingan Kinerja *Memory Usage* pada *framework* React dan Vue Menggunakan Google DevTools

Berdasarkan Gambar 11, terlihat bahwa React lebih efisien dalam penggunaan memori, dengan konsumsi memori sebesar 7.16 MB. Sementara itu, Vue menggunakan memori sebesar 7.28 MB. Meskipun perbedaannya tergolong sangat kecil, yaitu hanya sebesar 0.12 MB, hasil ini tetap menunjukkan bahwa React memiliki sedikit keunggulan dalam pengelolaan memori.



Gambar 12 Perbandingan Kinerja *Rendering Time* pada *framework* React dan Vue Menggunakan Google DevTools

Berdasarkan Gambar 12, dapat disimpulkan bahwa Vue memiliki waktu rendering yang lebih cepat, yaitu sebesar 82.80 ms, dibandingkan dengan React yang membutuhkan waktu 105.40 ms. Selisih waktu antara kedua *framework* tersebut mencapai 22.60 ms, menunjukkan keunggulan Vue dalam hal kecepatan rendering. Kecepatan rendering ini dapat memberikan pengalaman pengguna yang lebih baik, terutama pada aplikasi yang memerlukan pemrosesan elemen visual secara cepat dan responsif.



Gambar 13 Perbandingan Penggunaan Energi dan Emisi Karbon pada *framework* React dan Vue Menggunakan Globemallow

Berdasarkan Gambar 13, *framework* Vue menunjukkan konsumsi energi yang lebih tinggi dan menghasilkan emisi karbon yang lebih besar dibandingkan dengan *framework* React. Selisih penggunaan energi antara kedua *framework* tersebut adalah sebesar 0.000029 kWh, sedangkan perbedaan dalam emisi karbon yang dihasilkan mencapai 0.013 gram.

4. KESIMPULAN DAN SARAN

Penelitian ini mengevaluasi perbandingan kinerja dan penggunaan energi dari dua *framework* populer, yaitu React dan Vue, untuk menentukan *framework* yang lebih optimal dalam pengembangan aplikasi web. Dalam perbandingan kinerja, digunakan berbagai metrik, termasuk FCP (First Contentful Paint), LCP (*Largest Contentful Paint*), TBT (*Total Blocking Time*), CLS (*Cumulative Layout Shift*), SI (*Speed Index*), *memory usage*, dan *rendering time*. Pengujian kinerja dilakukan dengan memanfaatkan Google Lighthouse dan Google DevTools untuk mendapatkan hasil yang terukur dan objektif. Selain itu, penelitian ini juga mengevaluasi penggunaan energi dengan mengukur konsumsi daya dalam satuan kWh dan emisi karbon yang

dihasilkan selama proses pengujian, menggunakan tools Globemallow untuk memastikan akurasi dalam pengukuran.

Berdasarkan serangkaian eksperimen yang telah dilakukan, hasil penelitian menunjukkan bahwa framework Vue lebih unggul dalam beberapa aspek kinerja seperti FCP, LCP, TBT, CLS, SI, dan rendering time, menjadikannya pilihan yang baik untuk aplikasi yang mengutamakan kecepatan pemuatan dan *rendering* halaman. Di sisi lain, *framework* React menunjukkan keunggulan dalam *memory usage*, dan penggunaan energi, sehingga lebih sesuai untuk aplikasi yang memprioritaskan efisiensi memori dan keberlanjutan energi. Hasil ini memberikan wawasan penting bagi pengembang dalam memilih framework yang sesuai dengan kebutuhan proyek, baik dari segi kinerja maupun efisiensi energi.

Sebagai saran untuk penelitian selanjutnya, disarankan untuk memperluas cakupan dengan membandingkan framework lain, seperti React dan Angular, untuk memberikan gambaran yang lebih komprehensif tentang keunggulan dan kelemahan masing-masing framework. Selain itu, disarankan untuk menggunakan perangkat dengan spesifikasi yang bervariasi, seperti *low-end*, *mid-range*, dan *high-end*, agar hasil penelitian dapat mencerminkan kinerja framework pada berbagai kondisi perangkat keras. Hal ini penting untuk memahami bagaimana setiap framework beradaptasi terhadap keterbatasan atau kelebihan sumber daya perangkat yang digunakan, sehingga pengembang dapat memilih framework yang paling sesuai untuk target pengguna mereka.

DAFTAR PUSTAKA

- [1] M. F. Khoirurrizal, C. R. Hidayat, and Ruuhwan, "ANALISIS PERBANDINGAN FRAMEWORK FRONT-END JAVASCRIPT SOLIDJS DAN VUEJS PADA PENGEMBANGAN WEBSITE INTERAKTIF," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4106.
- [2] F. F. S. Axza and A. Qoiriah, "Analisis Perbandingan Framework Front-End Javascript React dan Vue Pada Pengembangan Website," *Journal of Informatics and Computer Science*, vol. 05, 2023.
- [3] A. Christopher, S. Anjelina, and Lina, "ANALISIS PERFORMA FRAMEWORK FRONTEND JAVASCRIPT BAGI PROGRAMMER," vol. 7, Dec. 2022.
- [4] E. Yhuni Pratama, "IMPLEMENTASI FRAMEWORK REACT JS PADA SISTEM PENJUALAN FURNITURE BERBASIS WEBSITE RESPONSIVE (STUDI KASUS : TOKO D.A PUTRA TUNGGAL)," Bandar Lampung, Nov. 2023.
- [5] A. Ridow Johanis M *et al.*, "Pengantar Teknologi Digital: Web dan Mobile Teknologi," Jun. 2024. [Online]. Available: <https://www.researchgate.net/publication/381291236>
- [6] M. Siahaan and V. O. Vianto, "Comparative Analysis Study of Front-End JavaScript Frameworks Performance Using Lighthouse Tool," Online, Nov. 2021.
- [7] O. S. I. Anggraeni, L. Sugiarto, and T. Agustin, "Studi Komparatif Performa Framework Javascript Modern dalam Pengembangan Aplikasi Web," *Jurnal Informatika dan Sains Teknologi*, vol. 2, no. 4, pp. 162–177, 2024, doi: 10.62951/modem.v2i4.239.
- [8] M. Wadholm, "Green and Sustainable JavaScript a study into the impact of framework usage," 2023. [Online]. Available: www.bth.se
- [9] W. Gren and F. D. Mander, "Comparison of Energy Usage and Response Time for Web Frameworks," 2023, [Online]. Available: www.bth.se
- [10] H. Bebinger, "Sustainability in Web Development through Energy Efficiency," Apr. 2022. [Online]. Available: <https://scholarworks.bgsu.edu/honorsprojects>