

## ANALISIS PERBANDINGAN SIMPANGAN BAKU WAKTU EKSEKUSI SOLIDJS DAN VUEJS DALAM PENGEMBANGAN WEBSITE INTERAKTIF

Ahmad Fadlilah Ramadlan<sup>1\*</sup>, Nor Wahidman<sup>2</sup>, Jadiaman Parhusip<sup>3</sup>

Universitas Palangka Raya

[ahmadfadlilah09@mhs.eng.upr.ac.id](mailto:ahmadfadlilah09@mhs.eng.upr.ac.id)<sup>\*</sup>, [wahidman948@gmail.com](mailto:wahidman948@gmail.com)<sup>2</sup>,

[parhusip.jadiaman@it.upr.ac.id](mailto:parhusip.jadiaman@it.upr.ac.id)<sup>3</sup>

### ABSTRACT

Selecting the appropriate front-end framework is crucial for developers to achieve optimal performance in modern web application development. SolidJS and VueJS are two popular frameworks, each with their own advantages and challenges. This study focuses on evaluating and comparing the performance of these frameworks, particularly in terms of execution time and consistency, through the calculation of standard deviation across three data creation scenarios (10k, 20k, and 30k rows). Testing was conducted over 15 iterations for each framework to ensure the reliability of the collected data. The findings of this study reveal that SolidJS outperforms VueJS in terms of faster average execution time but exhibits a higher standard deviation, indicating greater variability in its performance. Conversely, VueJS demonstrates more stable execution times, albeit with higher average execution time. This study provides crucial guidance for developers in selecting a framework based on the specific needs of an application, whether prioritizing execution speed or stability under heavy user loads. Ultimately, the choice of framework should align with the application's characteristics to deliver an optimal user experience.

**Keywords:** SolidJS, VueJS, Standard Deviation, Execution Time, Web Development.

### ABSTRAK

Menentukan penggunaan *framework front-end* yang sesuai sangat penting bagi pengembang untuk mencapai kinerja yang optimal dalam pengembangan aplikasi web modern. SolidJS dan VueJS adalah dua *framework* populer yang masing-masing memiliki keunggulan dan tantangan tersendiri. Penelitian ini berfokus pada evaluasi dan perbandingan kinerja kedua *framework*, terutama dalam hal waktu eksekusi dan konsistensinya, melalui perhitungan simpangan baku pada tiga skenario pembuatan data (10k, 20k, dan 30k baris data). Pengujian dilakukan sebanyak 15 iterasi untuk setiap *framework* guna memastikan keandalan data yang dikumpulkan. Hasil penelitian ini menunjukkan bahwa SolidJS lebih unggul dengan waktu eksekusi rata-rata lebih cepat dibandingkan VueJS, tetapi dengan simpangan baku yang lebih tinggi, yang mengindikasikan variasi kinerja yang lebih besar. Sebaliknya, VueJS menunjukkan konsistensi waktu eksekusi yang lebih stabil, meskipun dengan rata-rata waktu yang lebih tinggi. Penelitian ini memberikan pedoman yang sangat penting bagi pengembang dalam menentukan *framework* berdasarkan kebutuhan spesifik aplikasi, baik yang memprioritaskan kecepatan eksekusi maupun stabilitas dalam menghadapi beban pengguna yang besar. Pada akhirnya, pemilihan *framework* harus disesuaikan dengan karakteristik aplikasi untuk memberikan pengalaman pengguna yang optimal.

**Kata Kunci:** SolidJS, VueJS, Simpangan Baku, Waktu Eksekusi, Pengembangan Website.

### Article History

Received: November 2024

Reviewed: November 2024

Published: Desember 2024

Plagiarism Checker No 234

Prefix DOI : Prefix DOI :

10.8734/Kohesi.v1i2.365

**Copyright : Author**

**Publish by : Kohesi**



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



## PENDAHULUAN

Perkembangan teknologi web terus mendorong pengembang untuk mencari *framework* yang paling sesuai untuk aplikasi yang digunakan. Website adalah entitas digital yang terdiri dari berbagai halaman web yang saling terhubung dan menyertakan semua berkas terkait (Khoirurrizal dkk, 2024). *Framework* adalah kumpulan intruksi dalam suatu class dengan memiliki fungsi masing masing agar mempermudah dan mempercepat menyelesaikan program (Irawan & Novianto, 2020). SolidJS dan VueJS merupakan dua *framework front-end* yang sering dibandingkan karena keduanya memiliki keunggulan dan tantangan masing-masing. JavaScript adalah sebuah bahasa pemrograman yang diterapkan pada pengembangan web untuk menciptakan situs web yang interaktif, dinamis, dan responsif (Khoirurrizal dkk, 2024). SolidJS adalah sebuah *framework front-end* yang bertujuan untuk menyediakan solusi yang sangat efisien dan performa yang baik dalam pengembangan aplikasi web (Siahaan & Vianto, 2022). VueJS adalah *framework front-end* yang banyak dikenal dan mudah digunakan. VueJS menawarkan pendekatan pengembangan yang ramah pengguna dan mudah dipelajari (Sofi'ie & Qoiriah, 2023). SolidJS dikenal dengan efisiensi kinerja karena menggunakan kompilasi langsung ke JavaScript murni, sementara VueJS lebih stabil dan memiliki kurva pembelajaran yang lebih mudah.

Namun, ketika memilih *framework*, penting untuk mempertimbangkan lebih dari sekadar kecepatan rata-rata. Variasi dalam waktu eksekusi, yang dapat diukur melalui simpangan baku, memberi gambaran lebih dalam tentang konsistensi *framework* dalam menghadapi beban pengguna yang berbeda. Simpangan baku adalah salah satu teknik statistik yang lazim digunakan untuk menjelaskan homogenitas kelompok. Simpangan baku merupakan nilai statistik yang biasa digunakan untuk menentukan bagaimana sebaran data dalam sampel, serta seberapa dekat titik data individu ke mean atau rata-rata nilai sampel (Rahmawati dkk, 2020). Penelitian ini bertujuan untuk mengukur waktu eksekusi dan simpangan baku pada *framework* SolidJS dan VueJS dalam tiga skenario pembuatan baris data.

### Metode Penelitian

Penelitian ini memanfaatkan data sekunder yang diambil dari hasil pengujian kinerja/performa *framework* SolidJS dan VueJS. Data yang digunakan mencakup waktu eksekusi untuk tiga skenario utama, yaitu pembuatan 10k, 20k, dan 30k baris data. Setiap skenario diuji sebanyak 15 kali untuk masing-masing *framework* guna mendapatkan data yang representatif. Data yang terkumpul mencakup nilai waktu eksekusi minimum, maksimum, dan rata-rata pada setiap iterasi pengujian. Penelitian ini berfokus pada pengukuran dan analisis simpangan baku untuk menilai stabilitas performa kedua *framework*. Simpangan baku digunakan untuk mengukur variasi data waktu eksekusi di setiap skenario pengujian. Perhitungan dilakukan menggunakan rumus berikut:

$$\text{Simpangan Baku} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$$



Keterangan:

- $x_i$ : Nilai waktu eksekusi individu.
- $\bar{x}$ : Nilai rata-rata waktu eksekusi.
- $n$ : Jumlah iterasi pengujian (15 iterasi per *framework*).

Langkah perhitungannya melibatkan penghitungan rata-rata waktu eksekusi, dilanjutkan dengan menghitung selisih kuadrat setiap nilai waktu terhadap rata-rata, kemudian menjumlahkan hasil tersebut dan membaginya dengan jumlah data, dan akhirnya mengambil akar dari hasil perhitungan tersebut untuk memperoleh simpangan baku. Data diolah menggunakan perangkat lunak seperti Python dan Excel untuk memastikan akurasi perhitungan statistik. Analisis statistik deskriptif ini dilakukan untuk menghitung rata-rata, nilai minimum (ms), nilai maksimum (ms), dan simpangan baku (ms) dari waktu eksekusi masing-masing *framework* pada setiap skenario. Selanjutnya, hasil simpangan baku dianalisis untuk menilai stabilitas waktu eksekusi, sedangkan nilai rata-rata dibandingkan untuk mengevaluasi efisiensi *framework*. Dengan menggunakan pendekatan ini, penelitian ini mampu menjelaskan secara terperinci perbedaan antara stabilitas dan efisiensi yang dimiliki oleh SolidJS dan VueJS.

### Hasil dan Pembahasan

Hasil pengujian kinerja *framework* SolidJS dan VueJS pada tiga skenario (pembuatan 10k, 20k, dan 30k baris data) dirangkum dalam Tabel 1. Data tersebut mencakup waktu eksekusi minimum, maksimum, rata-rata, dan simpangan baku untuk setiap *framework* di setiap skenario.

Tabel 1. Hasil Statistik Deskriptif

Skenario	Framework	Min (ms)	Max (ms)	Rata-rata (ms)	Simpangan Baku (ms)
10k	SolidJS	1488	1818	1551	193.98
10k	VueJS	1587	1756	1676	84.62
20k	SolidJS	2969	3743	3288	392.93
20k	VueJS	3360	3481	3411	61.24
30k	SolidJS	4492	4992	4782	253.18
30k	VueJS	5054	5454	5148	226.35

Hasil pengujian menunjukkan bahwa SolidJS memiliki waktu eksekusi rata-rata lebih cepat di semua skenario, tetapi dengan simpangan baku lebih tinggi dibandingkan VueJS. Ini menunjukkan bahwa SolidJS memiliki variasi waktu eksekusi yang lebih besar. Sebaliknya, VueJS menunjukkan performa lebih stabil dengan simpangan baku lebih rendah, meskipun waktu rata-rata eksekusinya lebih lama. Pada skenario pembuatan 10k baris data, SolidJS menghasilkan waktu eksekusi rata-rata 1551 ms, dengan simpangan baku sebesar 193.98 ms. Sebaliknya, VueJS memiliki waktu rata-rata 1676 ms dan simpangan baku hanya 84.62 ms. Meskipun VueJS lebih stabil, SolidJS memberikan hasil yang lebih efisien dalam hal waktu rata-rata. Pada skenario pembuatan 20k dan 30k baris data, tren yang sama tetap terlihat. SolidJS tetap unggul dalam hal efisiensi waktu, namun variasi waktunya lebih besar. Ini menunjukkan bahwa SolidJS lebih rentan terhadap fluktuasi dalam kinerja, yang bisa menjadi pertimbangan penting dalam aplikasi yang membutuhkan konsistensi kinerja.

Pemilihan *framework* harus disesuaikan dengan kebutuhan aplikasi. SolidJS lebih cocok untuk aplikasi yang memprioritaskan kecepatan, sementara VueJS lebih ideal untuk aplikasi



yang membutuhkan stabilitas performa. Analisis ini memberikan pemahaman yang penting bagi pengembang dalam menentukan *framework* yang paling tepat untuk kebutuhan spesifik aplikasi yang digunakan.

### Kesimpulan

Hasil penelitian ini menunjukkan kesimpulan bahwa SolidJS memiliki kinerja yang lebih baik pada efisiensi waktu rata-rata di setiap skenario pengujian. Namun, tingginya simpangan baku pada SolidJS menunjukkan bahwa *framework* ini memiliki variasi waktu eksekusi yang lebih besar dibandingkan VueJS. Sebaliknya, VueJS menunjukkan performa yang lebih stabil dengan simpangan baku yang lebih rendah, meskipun waktu eksekusinya lebih lama dibandingkan SolidJS. Dengan demikian, pemilihan *framework* harus disesuaikan dengan kebutuhan spesifik aplikasi, lebih memprioritaskan kecepatan waktu eksekusi atau stabilitas performa dalam menghadapi beban pengguna yang besar.

### Daftar Pustaka

- Irawan, D., & Novianto, Z. (2020). Perancangan E-Learning Pada Sman 1 Kota Lubuklinggau Menggunakan *Framework* Codeigniter (CI) E-Learning Design In Sman 1, Lubuklinggau City Using *Framework* Codeigniter (CI) Pendahuluan Perkembangan teknologi informasi saat ini sangat pesat seiring den. *Jurnal Digital Teknologi Informasi*, 3, 53-60. <http://dx.doi.org/10.23960/jitet.v12i2.4106>
- Khoirurrizal, M. F., Hidayat, C. R., & Ruuhwan, R. (2024). ANALISIS PERBANDINGAN FRAMEWORK FRONT-END JAVASCRIPT SOLIDJS DAN VUEJS PADA PENGEMBANGAN WEBSITE INTERAKTIF. *Jurnal Informatika dan Teknik Elektro Terapan*, 12(2). <http://dx.doi.org/10.23960/jitet.v12i2.4106>
- Rahmawati, D., Ali, E. P., Nurvia, M., & Harahap, E. (2020). Aplikasi simpangan baku menggunakan Microsoft excel. *Matematika: Jurnal Teori dan Terapan Matematika*, 19(2), 47-54.
- Siahaan, M., & Vianto, V. O. (2022). Comparative Analysis Study of Front-end JavaScript Frameworks Performance Using Lighthouse Tool. *Jurnal Mantik*, 6(3), 2462-2468. <https://doi.org/10.35335/mantik.v6i3.3131>
- Sofi'ie, F. A. F., & Qoiriah, A. (2023). Analisis Perbandingan Framework Front-end Javascript React dan Vue Pada Pengembangan Website. *Journal of Informatics and Computer Science (JINACS)*, 5(02), 157-164. <https://doi.org/10.26740/jinacs.v5n02.p157-164>