

DAFTAR PUSTAKA

- Addy Osmani, & Barry Pollard. (2020, May 5). *Optimize Cumulative Layout Shift*. <https://Web.Dev/Articles/Optimize-Cls>.
- Anh Do. (2019). *React server-side rendering with Scala.js and GraalVM*.
- Ardiyanto, R., Ardianto, E., & Author, C. (2024). Analisa Peformasi Metode Rendering Website: Client Side, Server Side, dan Incremental Static Regeneration. In *Computer Science (CO-SCIENCE* (Vol. 4, Issue 1). <http://jurnal.bsi.ac.id/index.php/co-science>
- Cagan Sevensan. (2024). *OPTIMIZING WEB DELIVERY: THE IMPACT OF RENDERING METHODS ON USER EXPERIENCE ACROSS NETWORK CONDITIONS*.
- Chen, K. (2025). *Improving Front-end Performance through Modular Rendering and Adaptive Hydration (MRAH) in React Applications*. arXiv. <https://doi.org/10.48550/arXiv.2504.03884>
- Duric, Z. (2025). *Comparative Analysis of Server-Side Rendering: React vs. Next.js*. https://www.theseus.fi/bitstream/handle/10024/885455/Duric_Zoran.pdf?sequence=2
- Fadhilah Iskandar, T., Lubis, M., Fabrianti Kusumasari, T., & Ridho Lubis, A. (2020). Comparison between client-side and server-side rendering in the web development. *IOP Conference Series: Materials Science and Engineering*, 801(1). <https://doi.org/10.1088/1757-899X/801/1/012136>
- Google. (2020, May 28). *Evaluating page experience for a better web*. <https://Developers.Google.Com/Search/Blog/2020/05/Evaluating-Page-Experience>.
- Hanafi, R., Haq, A., & Agustin, N. (2024). Comparison of Web Page Rendering Methods Based on Next.js Framework Using Page Loading Time Test. *Teknika*, 13(1), 102–108. <https://doi.org/10.34148/teknika.v13i1.769>
- Jani, Y. (2020). The Critical Importance of React Performance Optimization: Strategies for Success. In *North American Journal of Engineering and Research Etd* (Vol. 1). <https://najer.org/najer>
- Jeremy Wagner. (2024, May 22). *Interaction to Next Paint (INP)*. <https://Web.Dev/Articles/Inp>.
- Kirill Konshin. (2018). *Next.js Quick Start Guide*. Packt Publishing Ltd.
- Malek Hakim. (2016). *Speed index and critical path rendering performance for isomorphic single page applications*.
- Mathias Beke. (2018). *On the Comparison of Software Quality Attributes for Client-side and Server-side Rendering* [Master]. University of Antwerp.
- Milica Mihajlija, & Philip Walton. (2023, April 12). *Cumulative Layout Shift (CLS)*. <https://Web.Dev/Articles/Cls>.
- Nefe Emadamerho-Atori. (2024, May 2). *What is Client-side Rendering (CSR)?* <https://Prismic.Io/Blog/Client-Side-Rendering>.

- Ollila, R., Mäkitalo, N., & Mikkonen, T. (2022). Modern Web Frameworks: A Comparison of Rendering Performance. *Journal of Web Engineering*, 21(3), 789–813. <https://doi.org/10.13052/jwe1540-9589.21311>
- Philip Walton. (2023, August 2). *User-centric performance metrics*. <https://web.dev/articles/user-centric-performance-metrics>.
- Philip Walton, & Barry Pollard. (2024, February 19). *Largest Contentful Paint (LCP)*. <https://web.dev/articles/lcp>.
- Salma Alam-Naylor. (2023, April 13). *The Acronyms of Rendering on the Web*. [https://www.netlify.com/blog/the-acronyms-of-rendering/#:~:text=much%20needed%20R%26R,-,What%20is%20rendering%3F,Search%20Engine%20Optimization%20\(SEO\)](https://www.netlify.com/blog/the-acronyms-of-rendering/#:~:text=much%20needed%20R%26R,-,What%20is%20rendering%3F,Search%20Engine%20Optimization%20(SEO).).
- Sanity. (2024, August 23). *Rendering definition*. <https://www.sanity.io/glossary/rendering>.
- te Dorsthorst, J. A. (2024). *Analysis of the Impact of JavaScript Design Patterns on Page Performance in Web Applications*. https://pure.tue.nl/ws/portalfiles/portal/347245412/Dorsthorst_J.pdf