DAFTAR PUSTAKA

Adeyeye, M., Makitla, I., & Fogwill, T. (2013). WebRTC using JSON via XMLHttpRequest and SIP over websocket initial signalling overhead findings. *WEBIST 2013 - Proceedings of the 9th International Conference on Web Information Systems and Technologies*. https://doi.org/10.5220/0004317901190124

Aditi, Prasad, V. K., Gerogiannis, V. C., Kanavos, A., Dansana, D., & Acharya, B. (2024). Utilizing convolutional neural networks for resource allocation bottleneck analysis in cloud ecosystems. *Cluster Computing*, *28*(1), 22. https://doi.org/10.1007/s10586-024-04720-z

Agten, P., Van Acker, S., Brondsema, Y., Phung, P. H., Desmet, L., & Piessens, F. (2012). JSand: Complete client-side sandboxing of third-party JavaScript without browser modifications. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/2420950.2420952

Amin Azad, B., Starov, O., Laperdrix, P., & Nikiforakis, N. (2020). Web Runner 2049: Evaluating Third-Party Anti-bot Services. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12223 LNCS*. https://doi.org/10.1007/978-3-030-52683-2_7

Basques, K. (2024). *Analyze runtime performance*. Google Developers. Diambil 29 Juni 2025, dari https://developer.chrome.com/docs/devtools/performance

Brinkmann, M. (2019, Februari 15). *Third Party Web: an analysis of third-party script costs*. https://www.ghacks.net/2019/02/15/third-party-web-an-analysis-of-third-party-script-costs/. Diambil 1 Juni 2024, dari https://www.ghacks.net/2019/02/15/third-party-web-an-analysis-of-third-party-script-costs/

Cao, B., Shi, M., & Li, C. (2017). The solution of web font-end performance optimization. *Proceedings - 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2017*, *2018-January*. https://doi.org/10.1109/CISP-BMEI.2017.8302083

Chrome Developers. (2024). *Lighthouse performance scoring*. Chrome for Developers. Diambil 29 Juni 2025, dari https://developer.chrome.com/docs/lighthouse/performance/performance-scoring

Dwivedi, D. (2023, Juli 26). *The Cost of 3rd Party Scripts: Understanding and Managing the Impact on Website Performance*. https://crystallize.com/blog/cost-of-3rd-party-scripts. Diambil 26 Agustus 2024, dari https://crystallize.com/blog/cost-of-3rd-party-scripts

Feng, Z., Yang, L., & Zhang, Y. (2024). Pomelo: Alternative Mechanism of Threads Communication for Accelerating Convolution on SIMT Based Processor. *2024 9th International Conference on Intelligent Computing and Signal Processing (ICSP)*, 1357–1360. https://doi.org/10.1109/ICSP62122.2024.10743993

Flanagan, D. (2011). JavaScript: The Definitive Guide 6th Edition. Dalam *Chemistry*.

Goel, U., Steiner, M., Na, W., Wittie, M. P., Flack, M., & Ludin, S. (2016). Are 3rd parties slowing down the mobile web? *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, *03-07-October-2016*. https://doi.org/10.1145/2987354.2987359

Google Developer. (2024, Desember 27). *Introduction to Google Analytics*. https://developers.google.com/analytics/devguides/collection/ga4. Diambil 11 Agustus 2024, dari https://developers.google.com/analytics/devguides/collection/ga4

Grosskurth, A., & Godfrey, M. W. (2005). A reference architecture for web browsers. *IEEE International Conference on Software Maintenance, ICSM*, *2005*. https://doi.org/10.1109/ICSM.2005.13

Hickson, I. (2014). HTML5: A vocabulary and associated APIs for HTML and XHTML. *W3C Recommendation*.

Hou, T., Bi, S., Wei, M., Wang, T., Lu, Z., & Liu, Y. (2022). When Third-Party JavaScript Meets Cache: Explosively Amplifying Security Risks on the

Internet. *2022 IEEE Conference on Communications and Network Security, CNS 2022*. https://doi.org/10.1109/CNS56114.2022.9947247

Ikram, M., Kaafar, M. A., Masood, R., Loizon, N., Tyson, G., & Ensafi, R. (2019). The chain of implicit trust: An analysis of the web third-party resources loading. *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019*. https://doi.org/10.1145/3308558.3313521

Karltorp, J. D., & Skoglund, E. (2020). *Performance of Multi-threaded Web Applications using Web Workers in Client-side JavaScript*. www.bth.se

Kosaka, M. (2023, Agustus 27). *Inside Look at Modern Web Browser, Part 1*. Chrome Developers. Diambil 27 Agustus 2024, dari https://developer.chrome.com/blog/inside-browser-part1

MDN Contributors. (2024). *Call Stack*. Mozilla Developer Network. Diambil 29 Juni 2025, dari https://developer.mozilla.org/en-US/docs/Glossary/Call_stack

Meta Developer. (2024, Desember 27). *Meta Pixel*. https://developers.facebook.com/docs/meta-pixel. Diambil 11 Agustus 2024, dari https://developers.facebook.com/docs/meta-pixel

Nakhaei, K., Ansari, F., & Ansari, E. (2020). JSSignature: eliminating third-party-hosted JavaScript infection threats using digital signatures. *SN Applied Sciences*, *2*(1). https://doi.org/10.1007/s42452-019-1805-5

Peng, S., Yin, B., Xia, Y., Yang, Q., & Wang, L. (2024). A Semi-supervised crowd counting method based on patch crowds statistics. *Pattern Analysis and Applications*, *27*(4), 147. https://doi.org/10.1007/s10044-024-01359-9

Ralph, B. (2001). Cascading Style Sheets: Designing for the Web. *Technical Communication*, *48*(3).

Selakovic, M., & Pradel, M. (2016). Performance issues and optimizations in Java script: An empirical study. *Proceedings - International Conference on Software Engineering*, *14-22-May-2016*. https://doi.org/10.1145/2884781.2884829

Sinha, M., Bera, P., & Satpathy, M. (2025). SYN-Monitor: An Energy Efficient Defense System against TCP-SYN Flooding Attacks in SDN. *Proceedings of*

the 26th International Conference on Distributed Computing and Networking, 346–351. https://doi.org/10.1145/3700838.3703695

Słodziak, W., & Nowak, Z. (2016). *Performance Analysis of Web Systems Based on XMLHttpRequest, Server-Sent Events and WebSocket*. https://doi.org/10.1007/978-3-319-28561-0_6

Sun, H., Wang, T., & Zhang, X. (2019). Optimization of internal thread structure of force sensor considering fatigue performance. *Sensor Review*, *39*(6). https://doi.org/10.1108/SR-05-2019-0137

Team, R. T. (2024). *Parsing: What is it & How is it Implemented?* https://botpenguin.com/glossary/parsing. Diambil 26 April 2025, dari https://botpenguin.com/glossary/parsing

Thomas Hunter II, & Bryan English. (2021). Multithreaded JavaScript. Dalam C. C. (Development E. D. E. (Production E. Amanda Quinn (Acquisitions Editor) (Ed.), *Multithreaded JavaScript Concurrency Beyond the Event Loop* (First Edition, Vol. 1, hlm. 1–38). O'Reilly Media, Inc.

van Riet, J., Malavolta, I., & Ghaleb, T. A. (2023). Optimize along the way: An industrial case study on web performance. *Journal of Systems and Software*, *198*. https://doi.org/10.1016/j.jss.2022.111593

Verdu, J., & Pajuelo, A. (2016). Performance Scalability Analysis of JavaScript Applications with Web Workers. *IEEE Computer Architecture Letters*, *15*(2). https://doi.org/10.1109/LCA.2015.2494585

Villaespesa, E., & Stack, J. (2015). Finding the motivation behind a click: Definition and implementation of a website audience segmentation. *Museums and the Web 2015: Proceedings*, *June*.

Voitsechov, D., Port, O., & Etsion, Y. (2018). Inter-Thread communication in multithreaded, reconfigurable coarse-grain arrays. *Proceedings of the Annual International Symposium on Microarchitecture, MICRO*, *2018-October*. https://doi.org/10.1109/MICRO.2018.00013

Walton, P., & Pollard, B. (2025a). *Largest Contentful Paint (LCP)*. web.dev. Diambil 29 Juni 2025, dari https://web.dev/articles/lcp

Walton, P., & Pollard, B. (2025b). *Total Blocking Time (TBT)*. web.dev. Diambil 29 Juni 2025, dari https://web.dev/articles/tbt

Wang, W., Kim, I. L., & Zheng, Y. (2019). AdJust: Runtime Mitigation of Resource Abusing Third-Party Online Ads. *Proceedings - International Conference on Software Engineering*, *2019-May*. https://doi.org/10.1109/ICSE.2019.00105

X Developer. (2024, Desember 27). *Embedded Tweets*. Diambil 11 Agustus 2024, dari https://developer.x.com/en/docs/x-for-websites/embedded-tweets/overview

Zhang, J., Liu, W., Zhao, W., Ma, X., Xu, H., Gong, X., Liu, C., & Yu, H. (2018). A Webpage Offloading Framework for Smart Devices. *Mobile Networks and Applications*, *23*(5). https://doi.org/10.1007/s11036-018-1009-z

Zhang, J., Qiao, X., & Lin, B. (2023). VTD-XML Parsing Performance Optimization based on Helper Thread Sampling Prefetching. *International Conference on Ubiquitous and Future Networks, ICUFN*, *2023-July*. https://doi.org/10.1109/ICUFN57995.2023.10200816

Zhang, X., Jeon, W., Gibbs, S., & Kunjithapatham, A. (2012). Elastic HTML5: Workload offloading using cloud-based web workers and storages for mobile devices. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, *76 LNICST*. https://doi.org/10.1007/978-3-642-29336-8_26

Zhong, Y., Huang, Y., Hu, J., Zhang, Y., & Ji, R. (2025). Towards Accurate Post-Training Quantization of Vision Transformers via Error Reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–18. https://doi.org/10.1109/TPAMI.2025.3528042

Zigisova, J., Pollard, B., Farrugia, K., N. Jose, A., & Hantsis, S. (2022, September 26). *Part I Chapter 8 Third Parties*. https://almanac.httparchive.org/en/2022/third-parties. Diambil 26 Agustus 2024, dari https://almanac.httparchive.org/en/2022/third-parties