

Maximizando a eficiência de projetos Web: Uma análise da influência dos frameworks no desempenho

Thiago Dal Santo da Luz

Universidade Tecnológica Federal do Paraná – Paraná

Jackson Sieben

Universidade Tecnológica Federal do Paraná – Paraná

Jair José Sequeira

Universidade Tecnológica Federal do Paraná – Paraná

Amoz Emanuel Moitinho Amaral

Universidade Tecnológica Federal do Paraná – Paraná

RESUMO

Com o aumento exponencial da utilização de conteúdos e serviços digitais, as empresas do setor estão cada vez mais focadas em proporcionar uma experiência de alta qualidade aos seus clientes. Hartson e Pyla (2012) destacam que a experiência do usuário é composta por cinco qualidades essenciais: utilidade, funcionalidade, usabilidade, persuasão e design gráfico.

Palavras-chave: Página Web, Framework, Eficiência.

1 INTRODUÇÃO

Com o aumento exponencial da utilização de conteúdos e serviços digitais, as empresas do setor estão cada vez mais focadas em proporcionar uma experiência de alta qualidade aos seus clientes. Hartson e Pyla (2012) destacam que a experiência do usuário é composta por cinco qualidades essenciais: utilidade, funcionalidade, usabilidade, persuasão e design gráfico.

A otimização de desempenho no *front-end* surge como uma questão crucial que influencia diretamente a funcionalidade dos sites, afetando a capacidade dos usuários de desfrutar de uma experiência satisfatória e ágil. Considerando que a interface representa o primeiro ponto de contato do usuário com o conteúdo, é imperativo que as empresas priorizem a otimização das interfaces para oferecer uma experiência eficiente e agradável (NAPOMUCENO; NAPOMUCENO; SADOK, 2020).

Um estudo conduzido pelo *Pinterest* revelou que uma redução de 40% no tempo de carregamento resultou em um aumento de 15% no tráfego (PAVIC; CHRIS; ANSTEY, 2023). Em meio a esse cenário competitivo, torna-se crucial para as empresas implementar técnicas de otimização no *front-end*, visando garantir uma experiência de usuário rápida, eficiente e satisfatória.



À medida que o desenvolvimento de interfaces gráficas busca capturar a atenção do usuário, muitas vezes isso leva à incorporação de múltiplos componentes, potencialmente aumentando os tempos de carregamento. Nesse contexto, este estudo se propõe a explorar algumas das principais técnicas de otimização em *front-end*, como *lazy loading*, *caching* e compressão de dados, com o objetivo de reduzir os tempos de carregamento e compará-las para determinar a mais eficaz na otimização do desempenho em aplicações web.

Um estudo conduzido pela *BBC* demonstrou que a perda de 10% de usuários ocorria para cada segundo adicional de tempo de carregamento do site (PAVIC; CHRIS; ANSTEY, 2023), destacando a importância de abordar o tempo de carregamento em interfaces de aplicações web. A análise e pesquisa dessas técnicas podem contribuir significativamente para o desenvolvimento de interfaces *web* mais eficientes e agradáveis, resultando em uma redução dos tempos de carregamento das páginas e na melhoria da performance geral da aplicação.

O objetivo deste estudo visa explorar o tempo de carregamento de interfaces gráficas em ambientes *web* e como os principais *frameworks web* auxiliam na performance dos mesmos. A análise se concentra em *frameworks* populares como *React* e *Vue.js* e em relação ao desenvolvimento sem *frameworks*. Embora o foco principal recaia sobre a vantagem de utilizar *frameworks* quando se diz respeito a performance geral, esta pesquisa também aborda questões de funcionamento de páginas *web*, visando oferecer *insights* valiosos para desenvolvedores e empresas que buscam otimizar suas aplicações *web*.

2 MATERIAIS E MÉTODOS

A página *web* é um documento acessível por meio da internet que pode conter diferentes tipos de conteúdo, como texto, imagens, vídeos e elementos interativos. No contexto deste estudo, a página *web* é desenvolvida utilizando *TypeScript* para a lógica de programação do lado do cliente. *TypeScript* é uma extensão da linguagem *JavaScript* que adiciona recursos de tipagem estática, permitindo aos desenvolvedores escrever código mais seguro e escalável. Além disso, são empregadas as bibliotecas *React* e *Vue.js*, sendo estruturas de interface de usuários populares para a criação de componentes reutilizáveis e a construção de interfaces de usuário dinâmicas e responsivas.

O *Microsoft Edge* é um navegador da *web* desenvolvido pela *Microsoft*. Ele é construído com base no *Chromium*, o mesmo motor de renderização utilizado pelo navegador *Google Chrome*, oferecendo suporte a tecnologias *web* modernas e padrões abertos. O *Microsoft Edge* é conhecido por sua velocidade, segurança e integração com serviços da *Microsoft*. Neste estudo, o *Microsoft Edge* é utilizado como o navegador padrão para avaliar a constância e desempenho das páginas *web* desenvolvidas.

O *Google Lighthouse* é uma ferramenta de código aberto desenvolvida pelo *Google* para avaliação da qualidade e desempenho de páginas da *web*. Ele analisa diversos aspectos das páginas, incluindo



desempenho, acessibilidade, boas práticas de desenvolvimento e *SEO* (*Search Engine Optimization*). O *Google Lighthouse* fornece relatórios detalhados e sugestões de melhorias, permitindo que os desenvolvedores otimizem suas páginas para oferecer uma experiência de usuário superior e melhorar sua classificação nos mecanismos de busca. Neste estudo, o *Google Lighthouse* é utilizado para realizar avaliações objetivas da qualidade e desempenho das páginas web desenvolvidas com *React* e *Vue.js*, utilizando a linguagem de programação *TypeScript*.

Para os testes, foi realizado o desenvolvimento de três páginas web, em um formato simples, utilizando *TypeScript* e *HTML*, a segundo utilizando *TypeScript* e *React* e a terceira utilizando *TypeScript* e *Vue.js*. O desenvolvimento foi feito utilizando os mesmos padrões para as três páginas, contendo uma imagem de alta resolução, chamadas de *API* simples e importação de arquivos de fontes.

Os testes foram realizados em três etapas, buscando a mínima, mediana e máxima pontuação encontrada pelo *Google Lighthouse*, sendo assim, um formato mais consistente de resultados foi encontrado, colocando a média das três baterias de teste como resultado.

3 RESULTADOS

Os resultados da análise utilizando o *Google Lighthouse* oferecem uma visão detalhada das vantagens que os frameworks podem proporcionar em termos de performance para projetos *web*. Ao comparar com o desenvolvimento sem o uso de *frameworks*, onde apenas *HTML* e *TypeScript* foram empregados, com projetos que incorporaram frameworks como *React* e *Vue.js*, as diferenças são notáveis.

Para o desenvolvimento sem *frameworks*, as pontuações obtidas nas três baterias de teste foram relativamente estáveis, mas ainda assim inferiores em comparação com os projetos que utilizam *frameworks*. Isso sugere que, embora seja possível desenvolver páginas *web* funcionais sem *frameworks*, a ausência dessas estruturas adicionais pode resultar em uma performance limitada.

Por outro lado, os projetos desenvolvidos com *frameworks* demonstraram uma melhoria considerável na performance. As pontuações mais altas obtidas nessas configurações indicam uma melhor otimização do código, uma gestão mais eficiente dos recursos e uma experiência de usuário mais rápida e fluida. O uso de *TypeScript* em conjunto com *frameworks* como *React* e *Vue.js* permitiu alcançar pontuações mais altas de forma consistente, demonstrando a capacidade dessas ferramentas em impulsionar a performance de páginas *web*.

Além disso, ao analisar a mediana das pontuações, fica evidente que o desenvolvimento com *frameworks* oferece uma vantagem significativa em termos de performance. Com uma diferença de 17 pontos na pontuação mediana entre os projetos desenvolvidos com e sem *frameworks*, fica claro que a incorporação dessas estruturas pode resultar em uma experiência de usuário substancialmente melhor.



Em resumo, os resultados destacam a importância dos *frameworks* no desenvolvimento de projetos *web* e como sua utilização pode contribuir significativamente para a melhoria da performance, garantindo que as páginas *web* sejam mais rápidas, eficientes e responsivas para os usuários.

4 CONSIDERAÇÕES FINAIS

Este estudo destaca a importância dos *frameworks* na performance de projetos *web*, demonstrando que o uso de *frameworks* como *React* e *Vue.js* resultou consistentemente em melhorias significativas em comparação com projetos desenvolvidos sem *frameworks*. A diferença substancial nas pontuações, especialmente na mediana, ressalta o impacto positivo dos *frameworks* na qualidade e na experiência do usuário. Ao optar por *frameworks*, os desenvolvedores podem garantir uma performance superior e uma experiência de usuário mais satisfatória em suas aplicações *web*.



REFERÊNCIAS

HARTSON, R.; PYLA, P. S. The UX Book: Process and guidelines for ensuring a quality user experience. [S.l.]: Elsevier, 2012.

NEPOMUCENO, K. T. C.; NEPOMUCENO, T. C. C.; SADOK, D. F. H. Measuring the internet technical efficiency: A ranking for the world wide web pages. IEEE Latin America Transactions, v. 18, n. 6, p. 1119–1125, May 2020.

PAVIC, B.; ANSTEY, C.; WAGNER, J. Learn: Why does speed matter?. [S. l.], 2023.