

# Comparação de Desempenho em Páginas Web: Uma Análise de Client-Side Rendering e Server-Side Rendering

Gabriel Trevisan Damke  
Universidade Tecnológica Federal do  
Paraná (UTFPR)  
Medianeira-PR, Brasil  
gabrieldamke@alunos.utfpr.edu.br

Daniel Mahl Gregorini  
Universidade Tecnológica Federal do  
Paraná (UTFPR)  
Medianeira-PR, Brasil  
danielgregorini@alunos.utfpr.edu.br

**Abstract**—With the advancement of search engines and website ranking systems, the use of search engine optimization (SEO) techniques has become vital for enhancing online visibility, with page performance being one of the main metrics considered by browsers. This study compares two web page rendering approaches: Client-Side Rendering (CSR) and Server-Side Rendering (SSR). Two pages, each employing one approach, were developed using the Next.js framework and evaluated with the Lighthouse tool, simulating various network conditions. The performance and SEO metrics examined include First Contentful Paint (FCP), Largest Contentful Paint (LCP), Total Blocking Time (TBT), Speed Index, and Cumulative Layout Shift (CLS). The results reveal significant variations in page performance under different connectivity conditions, highlighting the advantages and disadvantages of each rendering method in terms of SEO.

**Keywords**—SEO; Search engine optimization; Client-Side Rendering; Server-Side Rendering; Web page performance; Next.js.

**Resumo**—Com o progresso dos mecanismos de busca e classificação de sites, a utilização de técnicas de otimização para motores de busca (SEO) tornou-se vital para aprimorar a visibilidade online, sendo a performance da página uma das principais métricas consideradas pelos navegadores. Este estudo compara duas abordagens de renderização de páginas web: Client-Side Rendering (CSR) e Server-Side Rendering (SSR). Duas páginas, cada qual empregando uma abordagem, foram desenvolvidas com o framework Next.js e avaliadas com a ferramenta Lighthouse, simulando variadas condições de rede. As métricas de desempenho e SEO examinadas incluem First Contentful Paint (FCP), Largest Contentful Paint (LCP), Total Blocking Time (TBT), Speed Index e Cumulative Layout Shift (CLS). Os resultados revelam variações significativas no desempenho das páginas sob diferentes condições de conectividade, evidenciando as vantagens e desvantagens de cada método de renderização em termos de SEO.

**Palavras-chave**—SEO; Otimização de motores de busca; Client-Side Rendering; Server-Side Rendering; Performance de páginas web; Next.js.

## I. INTRODUÇÃO

Com o avanço dos sistemas de busca e ranqueamento de sites, softwares conhecidos como "crawlers" ou

"spiders" realizam buscas em toda a web, visitando páginas públicas. Esses crawlers seguem links de uma página para outra, coletando dados sobre o conteúdo e armazenando essas informações em uma base de dados chamada índice [1]. Após o rastreamento, ocorre o processo de indexação, onde o conteúdo das páginas web rastreadas é analisado e organizado. O índice criado é uma vasta base de dados que armazena informações sobre as páginas da web, como palavras-chave, metadados e a estrutura do conteúdo.

Quando um usuário realiza uma pesquisa em um motor de busca, o sistema consulta o índice e classifica os sites de acordo com uma série de parâmetros, como a relevância do conteúdo, a quantidade e qualidade dos backlinks do domínio, e o país de origem do domínio [2].

Um parâmetro crucial é a otimização do site, no qual os motores de busca realizam uma avaliação, seguindo vários parâmetros para definir um score para a página. O First Contentful Paint (FCP) é um dos parâmetros de otimização, o seu propósito é avaliar a velocidade na qual o primeiro elemento no DOM (Document Object Model) aparece para o usuário [3].

Outro parâmetro importante é o Largest Contentful Paint (LCP), que mede o tempo necessário para que o maior elemento de conteúdo visível na janela de visualização seja completamente renderizado [4].

Para medir o tempo total em que a página foi bloqueada durante o carregamento, nesse momento a página impede que os usuários interajam com ela. Existe a métrica de Total Blocking Time (TBT) [4].

O Speed Index mede a velocidade com que o todo conteúdo visível é carregado da página [4]. Quanto menor o índice, mais rápido o conteúdo aparece na tela, assim ficando mais fluido para o usuário.

Por último, a métrica de Cumulative Layout Shift (CLS), ela mede a estabilidade visual de uma página web durante o carregamento [4]. A quantidade de mudanças inesperadas de um layout é calculada somando todos os "shifts" (mudança de layout).

A Otimização para motores de busca, SEO (Search engine optimization), é um conjunto de práticas,

estratégias e técnicas usadas para melhorar a visibilidade e posicionamento de um site ou página web nos resultados orgânicos dos motores de busca [4]. O objetivo do SEO é atrair público orgânico, ou seja, atrair mais visitantes para as páginas.

O desenvolvimento de aplicações web oferece várias abordagens para o processamento e entrega de conteúdo para o usuário final. Sendo Client-Side Rendering (CSR) e Server-Side Rendering (SSR) [5]. As duas técnicas empregadas têm características únicas que impactam diretamente a performance e no estabelecimento de uma métrica de otimização para o site.

O CSR processa todo o conteúdo no navegador do usuário, permitindo maior interatividade, dinamismo e uma carga menor de processamento para o servidor, no entanto, sofre com um carregamento mais lento, o que impacta o SEO [4]. Na abordagem SSR, ao contrário do CSR, o conteúdo é renderizado no servidor e enviado para o usuário, consequentemente o lado do cliente é mais rápido e há uma melhor otimização para motores de busca, porém, aumenta muito a carga de trabalho do servidor [5].

## II. OBJETIVOS

Neste estudo, o principal objetivo é realizar uma análise comparativa do desempenho e das limitações de duas formas de renderização de páginas web: Client-Side Rendering e Server-Side Rendering [5]. Através da análise dos testes, abordamos as vantagens e desvantagens de cada abordagem, considerando diversos critérios de avaliação, como tempo de carregamento, estabilidade visual e interatividade, que também impactam o desempenho em testes de SEO [6].

## III. METODOLOGIA

Primeiramente foram desenvolvidas duas páginas web em um projeto utilizando o framework Next.js [7]. A página “/csr” foi criada usando a abordagem de Client-Side e a página “/ssr” usando a abordagem Server-Side. As páginas são iguais, mostram o mesmo conteúdo, uma lista de 150 clientes com informações de nome, email e telefone, com a mesma imagem para todos os clientes, a imagem tem 96 pixels de altura e largura, com tamanho de 413,2 kB (Figura 1).

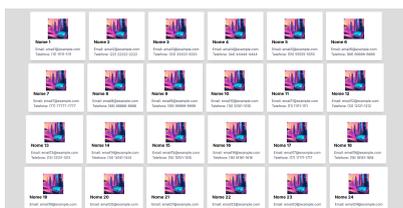


Figura 1. Página do projeto  
Fonte: Autoria própria

Todas as páginas mencionadas neste estudo foram desenvolvidas com estilização adicional utilizando CSS, mas não foram usados scripts JavaScript personalizados. O projeto foi desenvolvido para manter a simplicidade e o foco na comparação do desempenho. O projeto apenas usa os pacotes NPM obrigatórios do Next.js, sem nada adicional [8].

Para as duas abordagens, Client-Side Rendering e Server-Side Rendering, foi criado um build único do projeto para duas páginas, com o objetivo de comparar o desempenho e as limitações de cada uma.

As métricas de desempenho foram coletadas utilizando a ferramenta de análise Lighthouse do Chrome DevTools. Essa ferramenta fornece as informações de: SEO, First Contentful Paint, Largest Contentful Paint, Total Blocking Time, Speed Index, Cumulative Layout Shift.

As duas páginas foram submetidas a 5 rodadas de teste do Lighthouse simulando 5 tipos de redes: 2G, 3G, 4G lento, 4G rápido e sem throttling [9]. Essas simulações permitem avaliar o desempenho das páginas em diferentes condições de conectividade, refletindo a experiência dos usuários em diversas situações. Na rede 2G, com uma velocidade extremamente baixa de 80 kbit/s e latência de 800 ms, o foco foi observar como a página lida com condições de rede precárias. Em seguida, com a rede 3G, que oferece 1.6 Mbit/s de download e latência de 300 ms, a performance da rede é melhor, mas ainda há desafios de latência e largura de banda limitadas. Ao testar nas condições de 4G lento, com uma velocidade de 4 Mbit/s e latência de 170 ms, e de 4G rápido, com 12 Mbit/s e latência de 70 ms, a expectativa é que as páginas tenham um desempenho ótimo, especialmente em termos de carregamento de recursos pesados como imagens [8]. Por fim, a simulação sem throttling (sem limitação de rede) serve como um benchmark ideal, mostrando como as páginas funcionam em uma conexão de rede sem restrições. O conjunto de teste fornece um esboço grande, simulando vários tipos de usuários possíveis, desde usuários com uma péssima conexão e usuários com uma conexão ideal.

Para garantir que não houvesse interferência de rede nos resultados, tanto o servidor de teste quanto o cliente foram executados no mesmo computador. A máquina utilizada para os testes é equipada com um processador AMD Ryzen 5 3550H, que possui 4 núcleos e 8 threads, operando a uma frequência base de 2,1 GHz. O sistema conta com 16 GB de memória RAM, operando a 2666 MHz. O armazenamento é fornecido por um SSD ATA WDC WD10SPZX-21. As versões do software utilizadas foram o Node.js v20.15.0 e o Next.js v14.2.5. O sistema operacional utilizado foi o Linux Ubuntu 22.04.4 LTS, e os testes foram conduzidos utilizando o navegador Chrome versão 128.0.6613.84.

IV. RESULTADOS PRELIMINARES

A média das rodadas dos testes realizados com o Lighthouse em diferentes condições de rede demonstraram variações significativas na performance das páginas para diferentes perfis de rede, começando pelas métricas de performance e SEO que mostram uma visão geral dos testes (Quadro 1). A medida de SEO se manteve a mesma em todos os testes, pois a página é a mesma em todos os testes e o SEO avalia aspectos como descrição, links de navegação, metadados e outras características da página que não se alteram de acordo com a performance.

Métrica / Condição de Rede	Performance	SEO
2G CSR	74	100
2G SSR	98	100
3G CSR	95	100
3G SSR	100	100
4G lento CSR	99	100
4G lento SSR	100	100
4G rápido CSR	100	100
4G rápido SSR	100	100
No throttling CSR	100	100
No throttling SSR	100	100

Quadro 1. Comparativo de Métricas de Performance e SEO  
Fonte: Autoria Própria.

Já a performance da página teve resultados significativos. Os resultados indicam uma vantagem da renderização Server-Side em redes mais lentas, isso ocorre porque a renderização em Server-Side envia ao cliente um documento já pronto, isso significa em termos práticos um arquivo mais leve. Em redes mais rápidas, como 4G lento e 4G rápido, bem como em um cenário sem limitação de rede (no throttling), tanto Client-Side quanto Server-Side alcançam pontuações ideais. Já em conexão 4G rápida e No Throttling não existe diferença. A

menor performance no Server-Side foi de 98 pontos na pior conexão testada, no mesmo caso a renderização de Client-Side teve 74 pontos, isso demonstra a importância de considerar a rede ao escolher a abordagem de renderização, especialmente em aplicações voltadas para um público com acesso variado à internet.

Detalhando os indicadores que compõem a métrica de performance, vemos que os resultados de todos os testes no indicador de First Contentful Paint são de 0,2 segundos, isso ocorre porque o primeiro elemento renderizado da página é simples (Quadro 2). Entretanto, a renderização do último elemento, medido pelo indicador de Largest Contentful Paint, mostra diferenças significativas. Em velocidades mais baixas, o resultado mais lento para a página SSR foi de 0,5 segundos, enquanto da página CSR foi de 2,1 segundos (Quadro 2). Todos os resultados do SSR são iguais. Por conta da forma que a renderização é feita, o servidor envia todo o documento pronto para o cliente. Enquanto o CSR requisitou inicialmente a página do servidor, ele posteriormente buscou os dados necessários e construiu a página completa no lado do cliente. Essa abordagem em conexões lentas, como 2G, causou uma demora a mais para mostrar o último elemento da página.

Para finalizar a análise das métricas de performance apresentadas, podemos destacar alguns pontos principais observados (Quadro 2). O Total Blocking Time é significativamente reduzido em condições onde o Server-Side Rendering é utilizado, especialmente em redes mais lentas como 2G, o tempo maior do CSR em conexões lentas é explicado pelos dados requisitados do servidor serem maiores, enquanto o SSR recebe apenas o documento, o CSR recebe o documento, o script e os dados. A métrica Cumulative Layout Shift é 0 em todos testes, pois essa métrica mede a variação de tamanho que a layout tem durante o carregamento, essa métrica não é afetada pela velocidade da conexão, apenas pela forma como o layout foi construído na aplicação

Métrica / Condição de Rede	First Contentful Paint	Largest Contentful Paint	Total Blocking Time	Cumulative Layout Shift	Speed Index
2G CSR	0,2 s	2,1 s	360 ms	0	0,9 s
2G SSR	0,2 s	0,5 s	140 ms	0	0,3 s
3G CSR	0,2 s	0,8 s	160 ms	0	0,6 s
3G SSR	0,2 s	0,5 s	50 ms	0	0,2 s
4G lento CSR	0,2 s	0,8 s	40 ms	0	0,4 s
4G lento SSR	0,2 s	0,5 s	40 ms	0	0,2 s
4G rápido CSR	0,2 s	0,5 s	40 ms	0	0,4 s
4G rápido SSR	0,2 s	0,5 s	40 ms	0	0,2 s
No throttling CSR	0,2 s	0,5 s	40 ms	0	0,4 s
No throttling SSR	0,2 s	0,5 s	40 ms	0	0,2 s

Quadro 2. Comparativo de Métricas de Renderização: First Contentful Paint, Largest Contentful Paint, Total Blocking Time, Cumulative Layout Shift e Speed Index  
Fonte: Autoria Própria

## V. CONSIDERAÇÕES FINAIS

As análises realizadas neste estudo evidenciam a importância de se considerar a abordagem de renderização adequada de acordo com a qualidade da conexão do usuário da página. O projeto desenvolvido para os testes se encontra em um repositório no Github, livre para uso <[https://github.com/DanielGregorini/teste\\_comparacao\\_d\\_esempenho](https://github.com/DanielGregorini/teste_comparacao_d_esempenho)>. A renderização Server-Side demonstrou uma clara vantagem em relação a renderização Client-Side nos testes em conexões lentas, como 2G e 3G. Em redes rápidas, como 4G, as diferenças tornam-se menos pronunciadas, com ambas as abordagens conseguindo pontuações de performance excelentes. A escolha pela renderização SSR pode ser decisiva para garantir uma experiência de usuário eficiente e consistente.

Os resultados obtidos mostram a importância da otimização, até mesmo em páginas simples. A diferença pode ser vista nas pontuações de desempenho, que são usadas por buscadores como o Google para melhorar o posicionamento da página. Em sistemas mais complexos, onde o volume de dados e o número de componentes a serem renderizados são maiores, essa diferença na performance se torna ainda maior. A falta de otimização pode resultar em tempos de carregamento significativamente mais longos, afetando negativamente tanto a experiência do usuário quanto o SEO da página. Assim, investir em estratégias de otimização, se possível, utilizar a abordagem Server-Side em mais páginas e componentes, a minimização de scripts, e o gerenciamento eficiente de recursos, se torna ainda mais essencial para garantir a eficiência e o sucesso de uma aplicação na web.

## VI. AGRADECIMENTOS

Agradeço primeiramente a Deus e a minha família, que sempre esteve ao meu lado em todos os momentos, oferecendo apoio incondicional e sendo a base sólida que me sustenta.

## VII. REFERÊNCIAS

- [1] A. Verzbickas, E. F. Mocelin, M. B. de Souza Neto, e R. T. Siega, "Relatório Web Crawlers," Universidade Federal de Santa Catarina, Centro Tecnológico, Departamento de Informática e Estatística, Curso de Sistemas de Informação, INE5644 - Data Mining, Florianópolis, Brasil, 2013. [Online]. Available: [https://www.inf.ufsc.br/~luis.alvares/INE5644/G1\\_WebCrawlers.pdf](https://www.inf.ufsc.br/~luis.alvares/INE5644/G1_WebCrawlers.pdf)
- [2] A. Jain e M. Dave, "The Role of Backlinks in Search Engine Ranking," International Journal of Advanced Research in Computer Science and Software Engineering, vol. X, no. X, pp. 1-6, 2024.

- Disponível em: [www.ijarsse.com](http://www.ijarsse.com)
- [3] T. Rascia, \*Understanding the DOM — Document Object Model\*. New York City, NY, USA: DigitalOcean, 2020. ISBN: 978-0-9997730-9-3.
  - [4]. "Lighthouse Overview," Chrome Developers, Aug. 24, 2024. [Online]. Available: <https://developer.chrome.com/docs/lighthouse/overview?hl=pt-br>. Acesso em: 23 de agosto de 2024.
  - [5] A. H. Schneider, "Desenvolvimento web com Client Side Rendering: combinando Single Page Application e serviços de backend," Bacharelado, Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil, 2016. [Online]. Disponível: <https://lume.ufrgs.br/bitstream/handle/10183/150910/001009680.pdf>
  - [6] D. Mladenovic, A. Rajapakse, N. Kožuljevic, e Y. Shukla, "Otimização de mecanismos de busca (SEO) para profissionais de marketing digital: explorando determinantes de visibilidade online para serviços de banco de sangue," Online Information Review, vol. X, no. X, pp. 1-20, 2024. [Online]. Disponível em: <https://www.emerald.com/insight/content/doi/10.1108/OIR-05-2022-0276/full/html>
  - [7] "Next.js - The React Framework for Production," Next.js, [Online]. Disponível em: <https://nextjs.org/>. Acessado em: 23 de agosto de 2024.
  - [8] "npm - Node Package Manager," npm, [Online]. Disponível em: <https://www.npmjs.com/>. Acessado em: 23 de agosto de 2024.
  - [9] Í. L. da C. S. Silva, "Do 1G ao 5G: Evolução das Redes de Telefonia Móvel," Bacharelado em Ciências Exatas e Tecnológicas, Universidade Federal do Recôncavo da Bahia, Cruz das Almas, Brasil, 2016. [Online]. Disponível em: [https://www2.ufrb.edu.br/bcet/components/com\\_chronoforms5/chronoforms/uploads/tcc/20190327163532\\_2015.2\\_-\\_TCC\\_Itala\\_Liz\\_-\\_Do\\_1g\\_Ao\\_5g\\_Evolucao\\_Das\\_Redes\\_de\\_Telefonia\\_Movel.pdf](https://www2.ufrb.edu.br/bcet/components/com_chronoforms5/chronoforms/uploads/tcc/20190327163532_2015.2_-_TCC_Itala_Liz_-_Do_1g_Ao_5g_Evolucao_Das_Redes_de_Telefonia_Movel.pdf).