

Aula 25 – Métricas de Performance Web

Core Web Vitals



No cenário digital atual, onde a atenção do usuário é um recurso escasso e valioso, a performance de uma aplicação web deixou de ser um diferencial para se tornar uma exigência fundamental. Imagine a frustração de um usuário que tenta acessar um site de e-commerce para uma compra rápida, mas se depara com uma página que demora a carregar, elementos que pulam na tela ou botões que não respondem imediatamente. Essa experiência negativa não apenas afasta o cliente, mas também impacta diretamente a reputação da marca e, em última instância, os resultados de negócio.

O Cenário da Performance Web e a Importância das Core Web Vitals

A velocidade e a fluidez de uma aplicação web são cruciais para garantir uma boa experiência do usuário, aumentar o engajamento e melhorar o posicionamento nos motores de busca. É nesse contexto que as Métricas de Performance Web, e em particular as Core Web Vitals (CWV), ganham destaque. Elas representam um conjunto de indicadores focados na experiência real do usuário, estabelecidos pelo Google, que servem como um termômetro para a saúde e a qualidade percebida de um site.

- ❏ **Objetivo da Aula:** Ao longo desta aula, você será capaz de compreender a importância das Core Web Vitals para o sucesso de aplicações web modernas, identificar as principais métricas – LCP, FID (e seu sucessor INP), e CLS – e entender como cada uma delas afeta a percepção do usuário.

Em um mundo onde a informação está a um clique de distância e a concorrência é acirrada, a paciência do usuário é um luxo que poucos sites podem se dar ao luxo de testar. Pense na sua própria experiência: quantas vezes você já abandonou um site que demorou mais de três segundos para carregar? Essa é a realidade da web moderna, onde a expectativa por velocidade e responsividade é altíssima. Um site lento não apenas frustra o usuário, mas também pode custar vendas, reduzir o tempo de permanência e, crucialmente, prejudicar o ranqueamento nos resultados de busca.



Para endereçar essa questão de forma padronizada e focada no usuário, o Google introduziu as Core Web Vitals (CWV). Elas são um conjunto de métricas que avaliam a experiência de carregamento, interatividade e estabilidade visual de uma página web. Em vez de focar apenas em métricas técnicas abstratas, as CWV traduzem a performance em termos que realmente importam para quem está navegando: o quão rápido o conteúdo principal aparece, o quão rápido a página responde à interação e se os elementos visuais permanecem estáveis.

Podemos pensar nas Core Web Vitals como os "sinais vitais" de um paciente. Assim como um médico avalia a pressão arterial, batimentos cardíacos e temperatura para entender a saúde geral de uma pessoa, as CWV nos dão uma visão clara da "saúde" da experiência do usuário em um site.

Monitorá-las e otimizá-las é essencial não apenas para agradar o usuário, mas também para garantir que sua aplicação seja bem vista pelos motores de busca, que utilizam essas métricas como um fator de ranqueamento.

Largest Contentful Paint (LCP)

A Primeira Impressão Visual

1

O que é LCP?

Mede o tempo que leva para o maior elemento de conteúdo visível na viewport ser renderizado.

2

Por que importa?

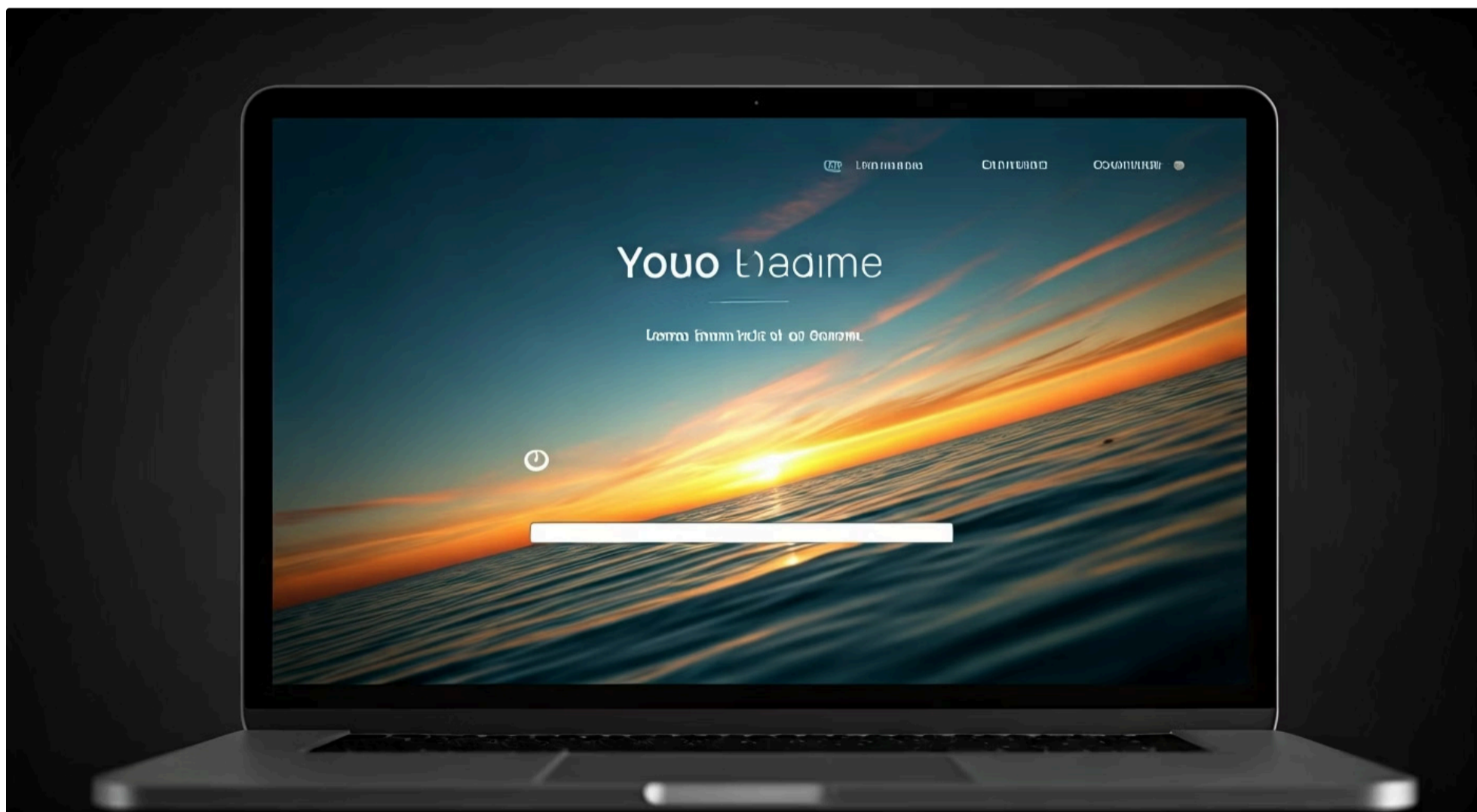
Captura o momento em que o usuário percebe que a página está realmente carregando e que o conteúdo significativo está disponível.

3

Meta ideal

Manter o LCP abaixo de 2.5 segundos é considerado um bom limiar para uma excelente experiência.

A primeira impressão é a que fica, e no mundo digital, essa máxima é ainda mais verdadeira. Quando um usuário acessa uma página web, a primeira coisa que ele espera é ver o conteúdo principal rapidamente. A ansiedade de esperar por uma tela em branco ou com elementos incompletos pode ser um fator decisivo para o abandono. É nesse momento crítico que entra em cena o Largest Contentful Paint (LCP), uma das Core Web Vitals mais importantes.



O LCP mede o tempo que leva para o maior elemento de conteúdo visível na viewport ser renderizado. Esse "maior elemento" pode ser uma imagem grande, um bloco de texto (como um título ou parágrafo principal), um vídeo ou até mesmo um elemento de fundo. A ideia é capturar o momento em que o usuário percebe que a página está realmente carregando e que o conteúdo significativo está disponível para consumo. Um LCP rápido indica que o usuário não precisou esperar muito para ver o que realmente importa.

Imagine que você está em um restaurante e fez um pedido. O LCP seria o momento em que o prato principal, aquele que você mais esperava, é finalmente colocado na sua mesa. Se o prato demora muito para chegar, sua experiência inicial será negativa, mesmo que a comida seja deliciosa.

Da mesma forma, um LCP alto significa que o usuário está esperando demais pelo "prato principal" da sua página, o que pode levá-lo a sair antes mesmo de provar o que você tem a oferecer.

Fatores que Afetam o LCP e Estratégias de Otimização

Compreender o que constitui um bom LCP é apenas o primeiro passo; o desafio real reside em identificar e mitigar os fatores que o elevam. Diversos elementos podem contribuir para um Largest Contentful Paint insatisfatório, e eles geralmente se concentram em como os recursos são carregados e processados pelo navegador. Servidores lentos, recursos que bloqueiam a renderização e imagens não otimizadas são os vilões mais comuns que impedem o conteúdo principal de aparecer rapidamente.

Principais Vilões

- Tempo de resposta do servidor lento
- Recursos render-blocking (CSS e JavaScript)
- Imagens de alta resolução não otimizadas
- Imagens sem atributos de width e height

Estratégias de Otimização

- Utilizar Content Delivery Network (CDN)
- Priorizar carregamento de CSS crítico
- Adiar JavaScript não essencial (defer/async)
- Usar formatos modernos de imagem (WebP)
- Comprimir imagens adequadamente
- Definir dimensões explícitas para imagens



Um dos principais culpados é o tempo de resposta do servidor. Se o servidor demora para enviar os primeiros bytes da página, todo o processo de carregamento é atrasado. Além disso, recursos como folhas de estilo CSS e scripts JavaScript podem ser "render-blocking", ou seja, o navegador precisa processá-los completamente antes de exibir qualquer conteúdo. Se esses arquivos são grandes ou mal otimizados, eles atrasam a renderização do LCP. Imagens de alta resolução que não são otimizadas para a web ou que não possuem atributos de width e height definidos também podem impactar negativamente.

- 📄 **Exemplo Prático:** Em um site de e-commerce, a imagem do produto principal na página de detalhes é frequentemente o LCP. Otimizar essa imagem e garantir que ela seja carregada rapidamente pode fazer uma grande diferença na percepção do usuário e, conseqüentemente, nas taxas de conversão.

First Input Delay (FID)

A Responsividade da Interação

Após o conteúdo principal da página carregar, a próxima expectativa do usuário é poder interagir com ela sem atrasos. Clicar em um botão, preencher um formulário ou abrir um menu e não obter uma resposta imediata pode ser tão frustrante quanto uma página lenta para carregar. É aqui que o First Input Delay (FID) entra em jogo, medindo a capacidade de resposta da sua aplicação web.



O que mede?

O tempo entre a primeira interação do usuário e o início do processamento pelo navegador



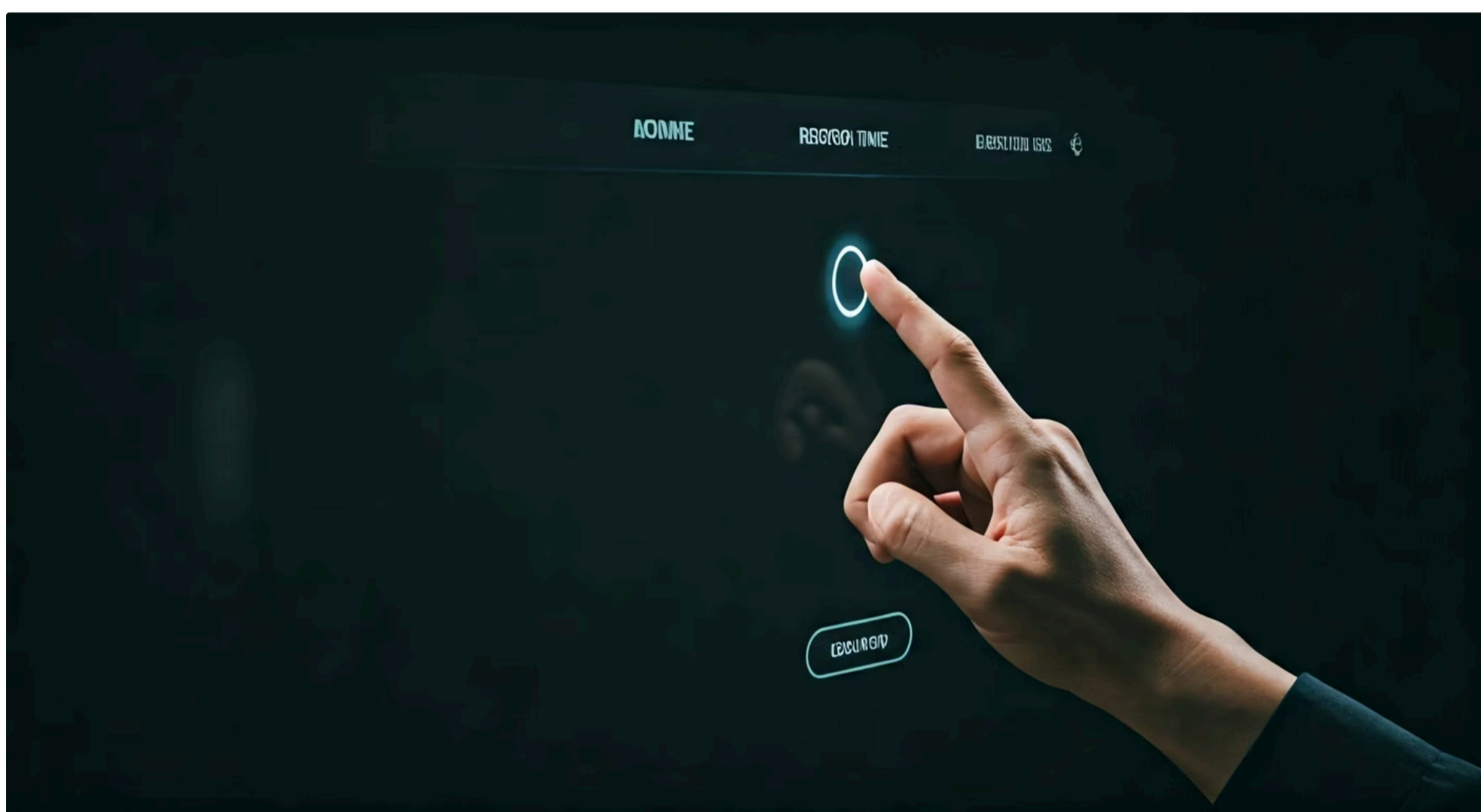
Meta ideal

FID abaixo de 100 milissegundos é considerado uma boa experiência



Principal causa

JavaScript pesado bloqueando o thread principal do navegador



O FID quantifica o tempo que o navegador leva para processar a primeira interação do usuário com a página. Essa interação pode ser um clique em um link, um toque em um botão ou a digitação em um campo de texto. Ele mede especificamente o "atraso" entre o momento em que o usuário interage e o momento em que o navegador consegue começar a processar esse evento. É importante notar que o FID não mede o tempo que leva para o navegador *responder* visualmente à interação, mas sim o tempo que ele leva para *iniciar* o processamento da interação.

Pense no FID como o tempo que um caixa de supermercado leva para reconhecer sua presença e começar a escanear seus produtos depois que você os coloca no balcão. Se o caixa está ocupado com outras tarefas ou com o sistema travado, há um atraso antes que ele possa atender você.

Da mesma forma, um FID alto geralmente ocorre quando o navegador está ocupado executando tarefas pesadas de JavaScript, como parsing, compilação ou execução de scripts, impedindo-o de responder prontamente às interações do usuário.

Otimizando o FID e a Transição para INP

Identificar a causa de um First Input Delay (FID) elevado geralmente nos leva ao JavaScript. Tarefas longas de execução de JavaScript são os principais culpados, pois elas bloqueiam o thread principal do navegador, impedindo-o de responder a qualquer interação do usuário até que essas tarefas sejam concluídas. Isso é particularmente problemático durante a fase inicial de carregamento da página, quando muitos scripts estão sendo processados.



Code Splitting

Divida o código JavaScript em pedaços menores e carregue sob demanda



Defer/Async

Adie scripts não críticos para não bloquear a renderização inicial



Otimização

Otimize o código para execução mais eficiente no thread principal

Para otimizar o FID, a principal estratégia é minimizar o tempo de execução de JavaScript no thread principal. Isso pode ser alcançado através de técnicas como *code splitting*, que divide o código JavaScript em pedaços menores e os carrega sob demanda, e *deferring* ou *async* para scripts não críticos, garantindo que eles não bloqueiem a renderização inicial. Além disso, é crucial otimizar o código JavaScript para que ele execute de forma mais eficiente e evitar a execução de tarefas complexas no thread principal durante a inicialização da página.

- Importante:** É fundamental mencionar que, embora o FID tenha sido uma métrica valiosa, o Google está evoluindo para uma métrica mais abrangente: o **Interaction to Next Paint (INP)**. A partir de março de 2024, o INP substituirá o FID como Core Web Vital.



FID (First Input Delay)

Mede apenas o atraso da **primeira** interação do usuário com a página

INP (Interaction to Next Paint)

Avalia a latência de **todas** as interações do usuário, desde o clique até a próxima atualização visual

Enquanto o FID mede apenas o atraso da *primeira* interação, o INP avalia a latência de *todas* as interações do usuário com a página, desde o clique até a próxima atualização visual da tela. Isso oferece uma visão mais completa da responsividade geral da página. Para otimizar o INP, as estratégias são semelhantes às do FID, mas com um foco ainda maior na otimização contínua do JavaScript e na minimização de bloqueios do thread principal ao longo de toda a vida útil da página.

Cumulative Layout Shift (CLS)

A Estabilidade Visual

Você já tentou clicar em um botão em um site e, de repente, o botão se moveu, fazendo você clicar em outra coisa por engano? Essa experiência frustrante é um exemplo clássico de um layout instável, e é exatamente isso que o Cumulative Layout Shift (CLS) busca medir. O CLS avalia a quantidade de "mudanças inesperadas de layout" que ocorrem na página, ou seja, o quanto os elementos visuais se movem de forma imprevisível enquanto o usuário está navegando.



Meta Ideal de CLS

Um valor de 0.1 ou menos indica uma página visualmente estável

Um CLS alto indica que a página está "pulando" ou "tremendo", o que não apenas é irritante, mas também pode levar a erros de clique e uma sensação geral de falta de profissionalismo. Essas mudanças de layout geralmente acontecem quando recursos são carregados de forma assíncrona ou quando o navegador não consegue determinar o espaço que um elemento ocupará antes de ele ser renderizado. O Google considera um CLS de 0.1 ou menos como uma boa experiência, indicando que a página é visualmente estável.

Pense no CLS como uma mesa de café que balança. Você está tentando colocar sua xícara de café nela, mas a mesa se move inesperadamente, fazendo com que você derrube a xícara. Ninguém gosta de uma mesa instável, e ninguém gosta de um site onde os elementos visuais não param no lugar. A estabilidade visual é crucial para a usabilidade e para a confiança do usuário na sua aplicação.

Causas e Soluções para o CLS

As mudanças inesperadas de layout que resultam em um alto Cumulative Layout Shift (CLS) são frequentemente causadas por elementos que são carregados de forma tardia ou que não têm seu espaço reservado na estrutura da página. Imagens e vídeos sem dimensões explícitas, anúncios e embeds que se redimensionam dinamicamente, e fontes da web que carregam tarde e causam um "flash de texto invisível" (FOIT) ou "flash de texto não estilizado" (FOUT) são os principais culpados.

Imagens sem dimensões

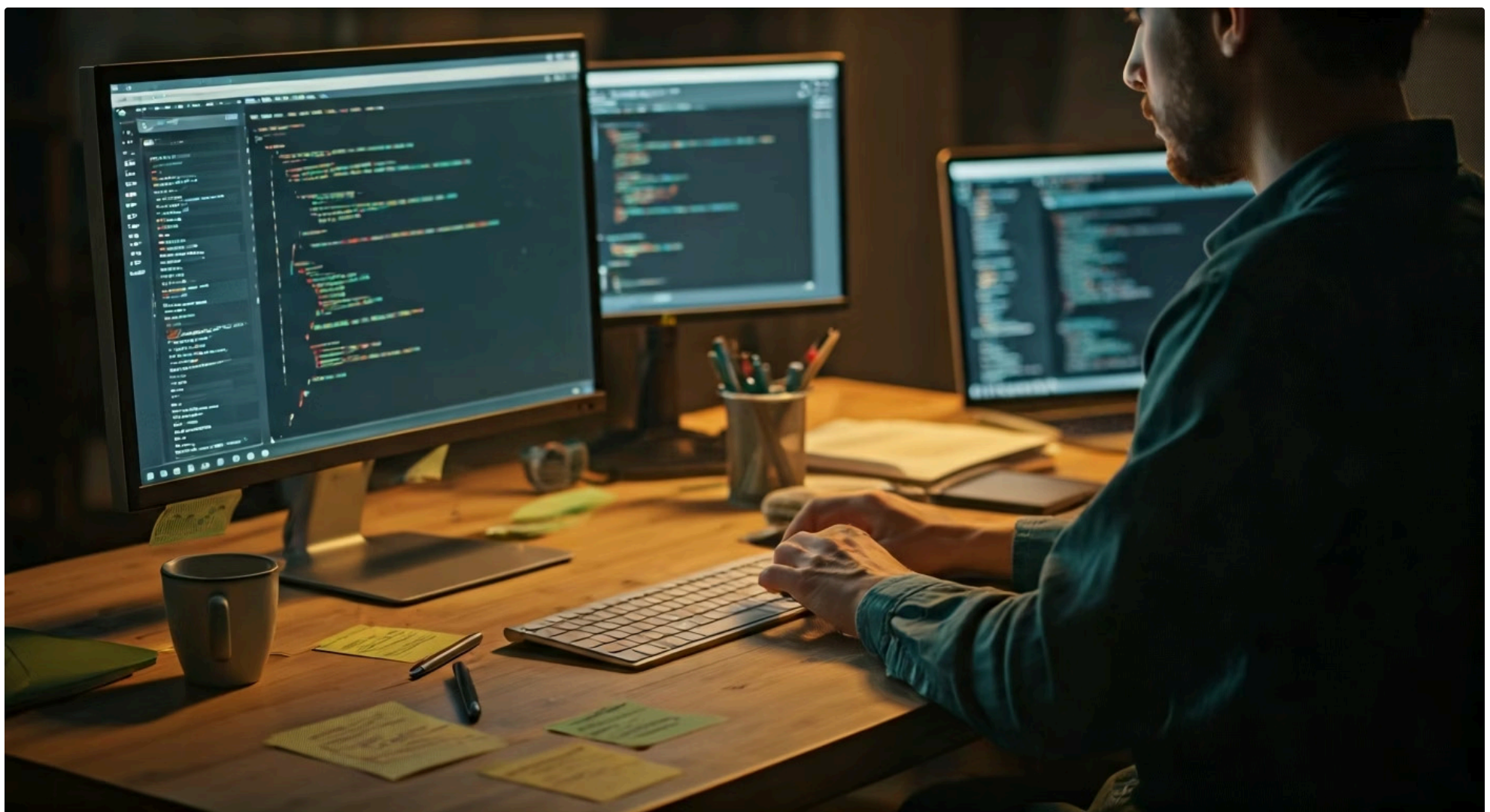
Quando uma imagem é carregada sem que o navegador saiba sua largura e altura, ele não consegue reservar o espaço adequado. Ao carregar, a imagem empurra o conteúdo existente para baixo, causando um *shift*.

Anúncios dinâmicos

Anúncios ou widgets de terceiros que injetam conteúdo dinamicamente e podem ter tamanhos variáveis causam deslocamentos inesperados.

Fontes da web

Se o navegador renderiza o texto com uma fonte padrão e depois a substitui pela fonte da web, o texto pode ocupar um espaço diferente, gerando um *shift*.



Estratégias de Mitigação

Para Imagens e Vídeos

- Sempre incluir atributos width e height
- Utilizar propriedade CSS aspect-ratio

Para Anúncios e Embeds

- Reservar espaço fixo
- Usar *placeholder* com maior dimensão possível

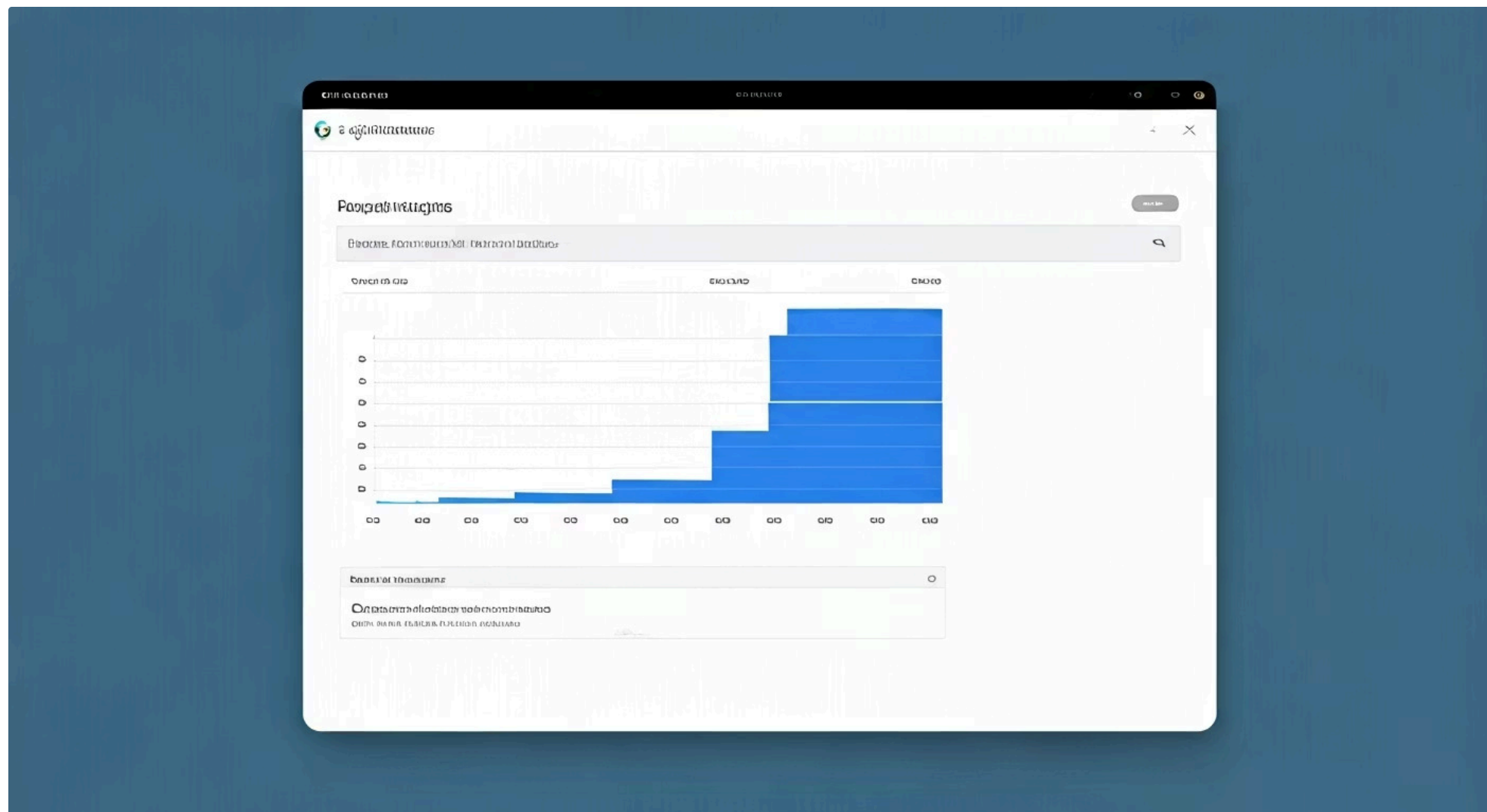
Para Fontes da Web

- Usar font-display: optional ou swap
- Pré-carregar fontes com <link rel="preload">

Exemplo Prático: Em um site de notícias, garantir que as imagens e anúncios não causem shifts inesperados ao longo do artigo é fundamental para a leitura contínua e a satisfação do usuário.

Ferramentas de Medição: Lighthouse e PageSpeed Insights

Compreender as Core Web Vitals é essencial, mas como podemos realmente medir e diagnosticar a performance de nossas aplicações? Felizmente, o Google oferece ferramentas robustas e acessíveis que nos permitem analisar a saúde de um site em relação a essas métricas. Duas das mais populares e eficazes são o Lighthouse e o PageSpeed Insights, cada uma com suas particularidades e casos de uso ideais.



Lighthouse

Ferramenta de auditoria de código aberto integrada ao Chrome DevTools. Realiza testes em ambiente controlado ("lab data") e gera relatórios detalhados sobre performance, acessibilidade, SEO e PWAs.

PageSpeed Insights

Ferramenta online que combina dados de "lab" (Lighthouse) com dados de "campo" (Chrome User Experience Report). Oferece visão realista da experiência de usuários reais em diferentes condições.

Comparação das Ferramentas

Conceito	Lighthouse	PageSpeed Insights
Tipo de Dados	Lab data (ambiente controlado)	Lab data + Field data (usuários reais)
Onde Usar	Chrome DevTools, durante desenvolvimento	Online, para análise de produção
Melhor Para	Identificar problemas e testar otimizações	Entender performance real dos usuários
Relatórios	Detalhado com sugestões específicas	Combina métricas reais com recomendações

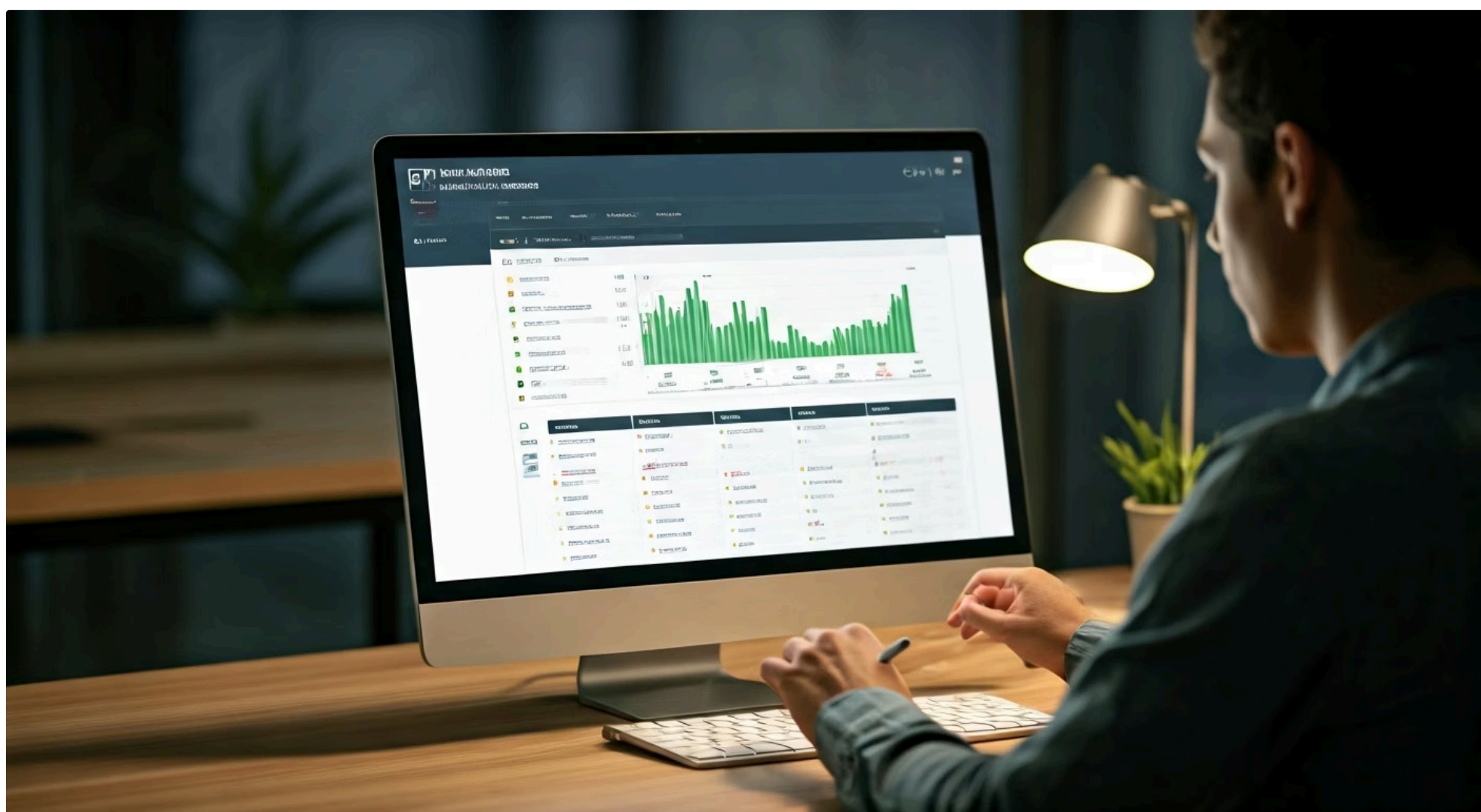
O **Lighthouse** é uma ferramenta de auditoria de código aberto e automatizada, integrada diretamente no Chrome DevTools. Ele realiza uma série de testes em uma página web e gera um relatório detalhado sobre performance, acessibilidade, melhores práticas de SEO e Progressive Web Apps (PWAs). Para a performance, o Lighthouse simula o carregamento da página em um ambiente controlado (o "lab data"), o que é excelente para identificar problemas e testar otimizações antes mesmo de elas serem implantadas em produção. Ele fornece pontuações e sugestões específicas para melhorar cada Core Web Vital e outras métricas importantes.

Já o **PageSpeed Insights (PSI)** é uma ferramenta online que combina dados de "lab" (do Lighthouse) com dados de "campo" (do Chrome User Experience Report - CrUX). Os dados de campo são métricas de performance coletadas de usuários reais que visitaram a página, oferecendo uma visão mais realista da experiência do usuário em diferentes condições de rede e dispositivos. O PSI é particularmente útil para entender como sua página está performando para seus usuários reais e para comparar seu desempenho com o de concorrentes. Ele apresenta as Core Web Vitals de forma clara e oferece recomendações de otimização tanto para dispositivos móveis quanto para desktop.

Consolidação e Próximos Passos

Chegamos ao final de nossa jornada pelas Métricas de Performance Web, as Core Web Vitals. Vimos que, em um ambiente digital cada vez mais exigente, a velocidade, a responsividade e a estabilidade visual de uma aplicação não são meros detalhes técnicos, mas sim pilares fundamentais para a experiência do usuário, o engajamento e o sucesso de qualquer negócio online. Compreendemos o papel crucial do LCP na primeira impressão visual, do FID (e seu sucessor INP) na interatividade e do CLS na estabilidade do layout. Mais importante, exploramos as ferramentas essenciais, Lighthouse e PageSpeed Insights, que nos permitem diagnosticar e otimizar esses aspectos.

- 📌 **Em prática:** Para aplicar o conhecimento adquirido, comece auditando suas próprias aplicações com o PageSpeed Insights. Identifique os pontos fracos nas Core Web Vitals e utilize as recomendações para criar um plano de otimização. Priorize a otimização de imagens, a minimização do JavaScript que bloqueia a renderização e a reserva de espaço para elementos dinâmicos. Lembre-se que a performance é um processo contínuo, não um evento único.



Autoavaliação

01

Qual das Core Web Vitals mede o tempo que o maior elemento de conteúdo leva para ser renderizado na tela?

- a) FID
- b) CLS
- c) LCP
- d) INP

02

Um alto valor de Cumulative Layout Shift (CLS) geralmente indica:

- a) Que o servidor está lento para responder.
- b) Que há muitos scripts JavaScript bloqueando a interação.
- c) Que elementos da página estão se movendo inesperadamente.
- d) Que a primeira imagem da página está demorando a carregar.

03

Qual das seguintes ferramentas combina dados de "lab" e "campo" para fornecer uma visão abrangente da performance?

- a) Chrome DevTools
- b) Lighthouse
- c) PageSpeed Insights
- d) WebPageTest

04

A partir de março de 2024, qual métrica substituirá o First Input Delay (FID) como Core Web Vital, oferecendo uma avaliação mais completa da responsividade?

- a) Largest Contentful Paint (LCP)
- b) Cumulative Layout Shift (CLS)
- c) Interaction to Next Paint (INP)
- d) Time to Interactive (TTI)

05

Explique a importância de otimizar as Core Web Vitals para o sucesso de uma aplicação web moderna, considerando tanto a experiência do usuário quanto o impacto nos motores de busca.

Gabarito

Questão 1

c) LCP

Questão 2

c) Que elementos da página estão se movendo inesperadamente.

Questão 3

c) PageSpeed Insights

Questão 4

c) Interaction to Next Paint (INP)

Próxima Aula

Estratégias de Cache

Parte 1: Cache no Cliente e no Gateway

Na próxima aula, mergulharemos em "Aula 26 – Estratégias de Cache – Parte 1: Cache no Cliente e no Gateway".

Entenderemos como o cache pode ser um poderoso aliado na otimização da performance, reduzindo a carga do servidor e acelerando a entrega de conteúdo para o usuário.

Recursos Adicionais

- **Web.dev/vitals:** Para a documentação oficial e mais aprofundada sobre as Core Web Vitals.
- **Google Developers Blog:** Para acompanhar as últimas atualizações e tendências em performance web.
- **Artigos sobre otimização de imagens e JavaScript:** Para aprofundar nas técnicas de otimização discutidas.

📌 **NOTA IMPORTANTE:** As informações regulatórias/legais/técnicas desta aula estão atualizadas até 2025. Consulte sempre fontes oficiais para verificar alterações.

