

What is Cumulative Layout Shift (CLS)

TL;DR: Cumulative Layout Shift (CLS) measures how much your webpage layout unexpectedly moves during page load. To fix it and stabilize your layouts, make sure images and ads have fixed sizes, use font-display: swap for fonts, reserve space for ads, and load styles before scripts.

Having a button you were about to click suddenly jump way down is the ultimate rage-click machine.

It's frustrating, ruins the entire experience, and might cost you a pretty penny if you initiate an express checkout instead of checking your shopping cart.

Luckily, since the introduction of Google's [Core Web Vitals](#) in 2021, site owners have been motivated to monitor real user interactions with their sites and improve buyer journeys based on this data.

One of the core principles is that elements on a page should not randomly move while browsing and interacting with a website.

Let's learn all about it!

What is Cumulative Layout Shift (CLS)?

Cumulative Layout Shift (CLS) is a CWV metric that calculates the visual stability of a webpage during the loading process. It quantifies the amount of unexpected layout shifts that occur as the page loads and is crucial for assessing the user experience.

Cumulative = increasing by successive additions.

Layout Shifts Explained

Layout shifts occur when the visual elements on a webpage unexpectedly move or shift around as the page loads or as a user interacts with the content. They disrupt the entire user experience, especially when they cause unintended actions like clicking on ad banners.



Expected vs Unexpected Layout Shifts

Layout shifts don't necessarily mean a bad user experience.

Any intentional change in layout (such as animations and transitions) that is triggered by user action and happens within 500 milliseconds after the interaction is considered “expected.” Other examples include:

- opening a dropdown menu or expanding a section
- flipping through elements like carousels, slideshows, or dynamic content that users anticipate
- clicking on a button to reveal additional content or scrolling to navigate through a page

Unexpected layout shifts, on the other hand, represent involuntary movement, unconsented actions, and surprise alterations that catch users off guard and disrupt the browsing experience.

What is A Good CLS Score?

Unlike [Largest Contentful Paint \(LCP\)](#) and [Interaction to Next Paint \(INP\)](#), CLS is not measured in seconds and is typically considered to be “good” if it's under 0.1.

CLS is categorized by the following thresholds:

- **Good:** CLS score less than 0.1
- **Needs Improvement:** CLS score between 0.1 and 0.25
- **Poor:** CLS score greater than 0.25



A natural follow-up question is—if not measured in seconds, how is CLS calculated exactly?

How is Cumulative Layout Shift Measured?

Cumulative Layout Shift (CLS) is measured using a scoring system that quantifies the visual stability of a webpage during the loading process. It is calculated based on the movement of unstable elements in that specific viewport between two rendered frames.

Two measures are used to quantify that movement:

- Impact fraction
- Distance fraction

layout shift score = impact fraction * distance fraction

Impact fraction (IF)

The impact fraction represents the area affected by the layout shift as a proportion of the viewport size between two frames. It's basically a calculation of how much an element occupies in the viewport before and after the shifting.

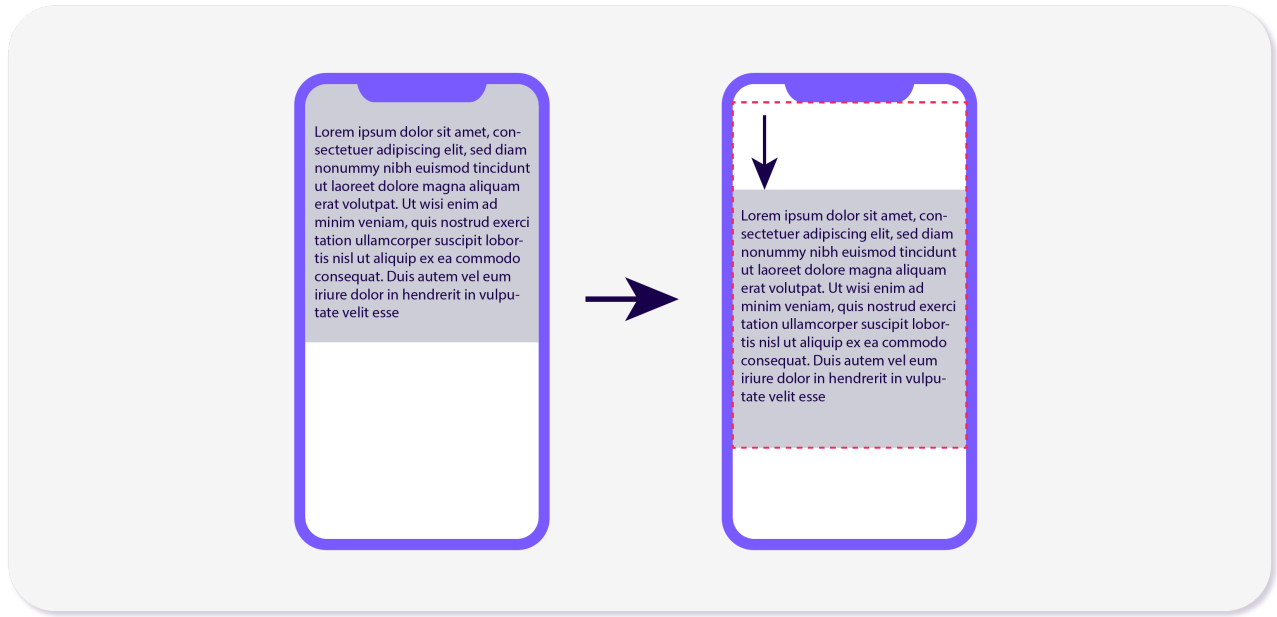


This can also be represented by a simplified formula:

$$IF = \text{Total area of viewport} / \text{Total area of layout shifts}$$

Distance fraction (DF)

The distance fraction, on the other hand, represents the distance that unstable elements have traveled relative to the viewport. It is the greatest horizontal or vertical distance any unstable element has moved in the frame divided by the viewport’s largest dimension (width or height, whichever is greater).



In a simplified formula, the distance fraction would look like this:

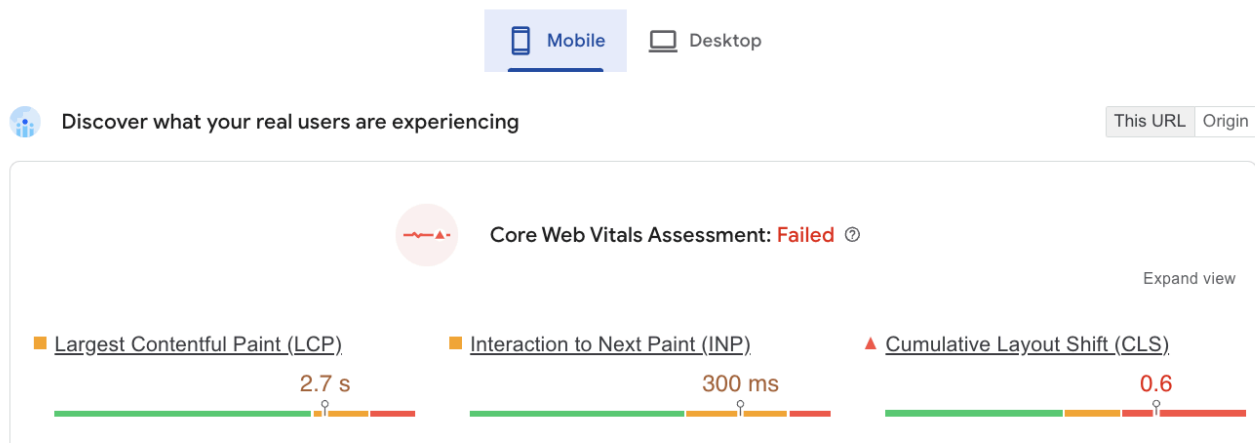
$$DF = \text{Viewport’s largest dimension} / \text{Greatest horizontal or vertical distance traveled by any unstable element}$$

Measuring CLS with lab data vs field data

In your Google PageSpeed Insights report, CLS is present in both your Core Web Vitals assessment and Performance Score breakdown.

Often, you’ll notice differences between the sections, and that’s because they are generated using two sets of incrementally different data:

- The Core Web Vitals assessment section uses “field data,” aka real-user data collected during real browsing sessions on your website



- The Performance score section uses “lab data,” aka simulated data collected in a controlled environment via computers

METRICS		Expand view
▲ First Contentful Paint	5.2 s	
▲ Total Blocking Time	5,980 ms	
▲ Speed Index	15.6 s	
▲ Largest Contentful Paint	33.1 s	
▲ Cumulative Layout Shift	0.569	

Both play a vital role in optimizing for CLS issues:

- **Primary:** Use Field data to gauge how well your website responds to real user interactions
- **Secondary:** Use Lab data to identify and troubleshoot technical issues in specific cases (network strength, location, device type, etc.)

Common Causes Of CLS Issues

Cumulative Layout Shift (CLS) issues can arise due to various factors, often related to how webpages are designed, structured, and loaded. The most common ones include:

Images Without Dimensions

When images on a webpage do not have specified width and height attributes in the HTML, browsers will reserve space for these images based on default dimensions. When the actual image loads and takes up more or less space than the default box, it will cause layout shifts.

Ads, Embeds, or iFrames

Ads, embedded media, or iframes that load asynchronously or without specified dimensions can lead to layout shifts when they render and push other content down or sideways.

Dynamic Content

Content that loads or changes size after the initial render, such as expanding accordions, pop-ups, or dynamic ad placements, can cause layout shifts if not properly managed.

Web Fonts & Font Loading

Loading web fonts asynchronously or using font-display: swap can lead to a flash of invisible text (FOIT) or a flash of unstyled text (FOUT), causing layout shifts when the fonts are applied. Delayed font loading or the use of custom fonts, on the other hand, can cause text to reflow when the font is applied, leading to unexpected shifts in layout as the page loads.

Third-Party Scripts

Scripts from third-party services or analytics tools that manipulate the DOM or load additional content asynchronously can trigger layout shifts if not carefully managed or optimized.

CSS Animations and Transitions

Improperly implemented animations or transitions triggered by user interactions or scripted events can cause layout shifts if they affect the position or size of elements on the page.

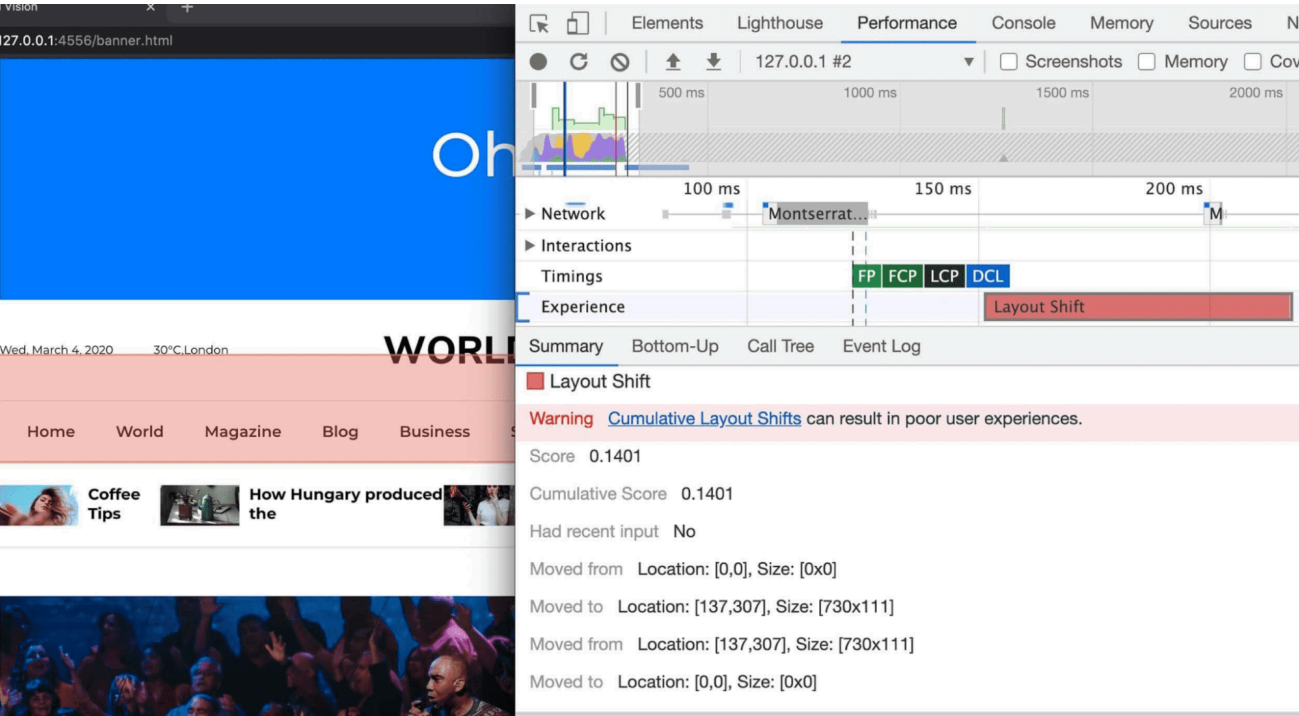
Identifying CLS Issues On Your Own

Here’s a more comprehensive guide on how to identify Cumulative Layout Shift (CLS) issues using both browser developer tools and performance analysis tools like Google PageSpeed Insights (PSI).

Using Google Chrome DevTools

#1 Open DevTools: First, open Google Chrome and navigate to the webpage you want to analyze. Then, right-click anywhere on the page, select "Inspect," and start interacting with the page as a normal user would.

#2 Go to the Performance Panel: In DevTools, navigate to the "Performance" panel and expand the Experience section on the left. The Summary view for a Layout Shift record includes the CLS score and red rectangular boxes indicating affected areas in the viewport.



Using Google PageSpeed Insights

You can find several warnings related to layout shifts in your Google PSI report, categorized under the "Opportunities" and "Diagnostics" sections:

- **Avoid large layout shifts:** This warning indicates instances where elements on the page shift significantly during loading, causing visual instability for users.

▲

Avoid large layout shifts — 2 layout shifts found

^

These are the largest layout shifts observed on the page. Each table item represents a single layout shift, and shows the element that shifted the most. Below each item are possible root causes that led to the layout shift. Some of these layout shifts may not be included in the CLS metric value due to [windowing](#). [Learn how to improve CLS](#) CLS

- **Image elements do not have explicit width or height**

■

Image elements do not have explicit [width](#) and [height](#)

^

Set an explicit width and height on image elements to reduce layout shifts and improve CLS. [Learn how to set image dimensions](#) CLS

- **Eliminate render-blocking resources:** [Render-blocking resources](#) such as CSS and JavaScript files can delay the rendering of content and contribute to layout shifts.

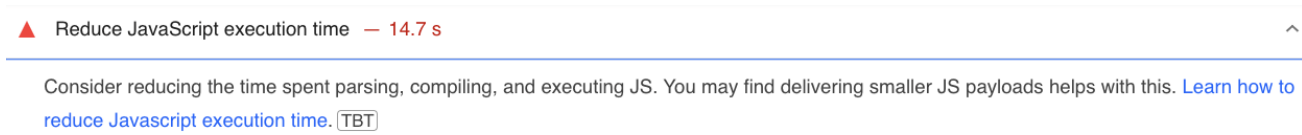
▲

Eliminate render-blocking resources — Potential savings of 1,270 ms

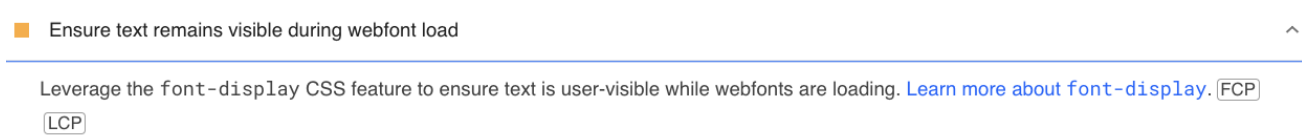
^

Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. [Learn how to eliminate render-blocking resources](#). FCP LCP

- **Reduce JavaScript execution time:** [Long-running JavaScript](#) tasks can delay the rendering of content and may contribute to layout shifts as the browser processes scripts.



- **Ensure text remains visible during web font load:** During asynchronous web font loading, text may initially appear with fallback fonts until the original font fully loads, potentially causing layout shifts due to differences in dimensions or styles.



Now, let's go straight to the most effective techniques for fixing CLS issues.

5 Ways to Improve Your CLS Score

1. Specify Image Dimensions

With the rise of Responsive Web Design, CSS became the norm for resizing images, but this led to reflow issues as images loaded.

To prevent this, modern browsers calculate default aspect ratios based on image dimensions specified in width and height attributes. This allows browsers to reserve space for images before they load, minimizing layout shifts and improving user experience.

To implement this, set `width` and `height` attributes on `img` tags and use CSS to resize images as needed.

```
<!-- set a 16:9 aspect ratio as 640x360 pixels -->

```

Browsers then calculate default aspect ratios based on these attributes, ensuring proper space allocation. The 'auto' value in CSS ensures that image dimensions override default aspect ratios after image loading, reducing layout shifts.

For [responsive images](#), `srcset` specifies available image sizes, and maintaining consistent aspect ratios across images ensures proper resizing.

For example, Chrome, Firefox, and Safari support setting width and height on elements within the `picture` element, enabling more precise image control.

```

```

For a comprehensive read on aspect ratios and responsive images, you can further check Craig Buckler's ["Jank-free page loading with media aspect ratios"](#).

2. Preload Fonts

There are several ways to go about [optimizing font delivery](#) for better CLS:

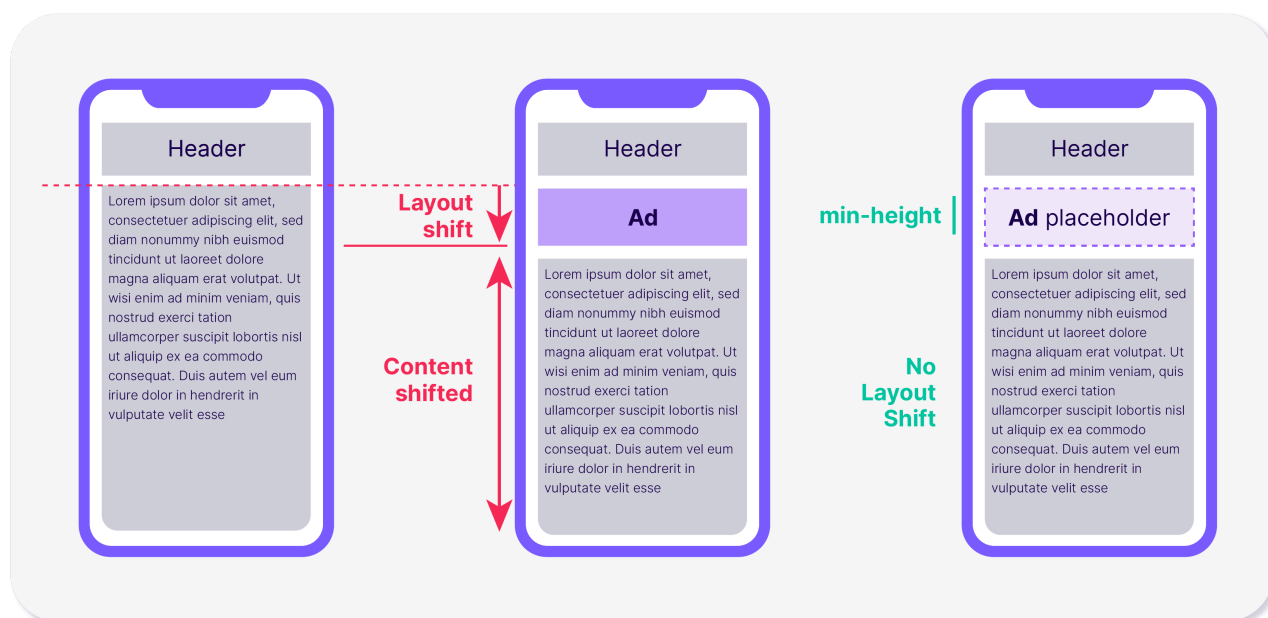
- Loading critical web fonts as early as possible using [link rel=preload](#) will ensure the font is ready for the first paint, reducing the risk of layout shifts.
- Using font-display: optional will help call the web font only if it's available by the time of initial layout.
- Setting fallback fonts that are of similar style to your web font by taking into account the font family, weight, width, etc. This can be done by using the [size-adjust](#), [ascent-override](#), [descent-override](#), and [line-gap-override](#) APIs. For details, see Improved font fallbacks.
- Considering the [Font Loading API](#), which can help reduce font loading time.

3. Reserve Space for Ad Layout Stability

Arguably, the biggest culprit for CLS issues, ads have always been a hot topic.

But when it comes down to revenue vs user experience, there really isn't a win-win situation. Instead, there are ways you can gain more control over how ads are displayed on your website to minimize layout shifts:

- An overall rule is to place late-loading dynamic content lower in the viewport so you have more options on how to reserve space for it.
- Use ad placeholders with fixed dimensions to reserve space for ads before they render to prevent shifts; if you're expecting small as well as bigger ads to be returned, style a larger ad container or choose the most likely size for the ad slot based on previous data. Lastly, avoid collapsing the ad container, even if this means that there's more blank space in the viewport when no ads are returned. Removing it will cause the same amount of CLS.
- Reserve space for late-loading content by adding a min-height styling or by using the new [aspect-ratio](#) CSS property to allow the ad containers to grow as necessary while reducing the risk of layout shifts.



4. Use CSS Transitions Instead of Animations

With animations, you generally want to avoid re-layouts, re-paints, and re-composites. To do that, consider:

- Using CSS transitions over animations whenever possible. CSS transitions typically result in smoother animations and are less likely to cause layout shifts.
- Avoiding to make changes to [top](#) and [left](#) properties of elements, as well as [box-shadow](#) and [box-sizing](#), even when the animated element is moved on its own layer.
- Choosing CSS properties like [transform](#) animations to translate, scale, rotate, or skew elements instead.

Note: Composited animations using [translate](#) and non-composited animation don't cause re-layout and don't count toward CLS. To create [high-performance animations](#), explore all CSS properties that cause layout shifts.

5. Leverage bfcache

[Back/forward cache](#) stores web pages in the browser's memory for a little while after you navigate away to keep them fast-loading if you decide to go back to the page. This makes this page instantly available without causing re-paints and repeated layout shifts.

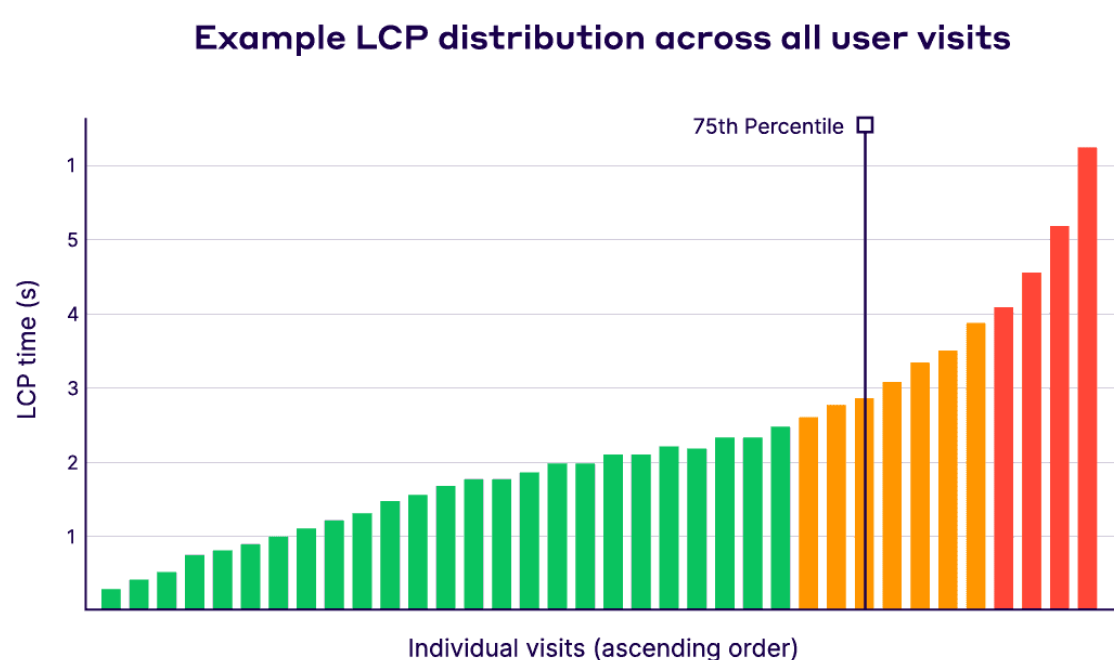
It's available on all browsers by default, but your site might not be eligible for the bfcache usage because of some common blockers, like:

- Complex JavaScript interactions
- User authentication and personalized content
- Non-standard navigation techniques (like SPAs or AJAX-based navigation)
- Heavy resource dependencies
- Cache Control Headers (e.g., Cache-Control: no-store)
- Cross-Origin Resource Sharing (CORS)

Other Tips For Passing Core Web Vitals

To pass Core Web Vitals, it's not only a matter of meeting the requirements of one of the metrics.

The 75th percentile rule refers to the threshold at which 75% of visits to a webpage meet or exceed a certain performance standard. Meaning if you fail one of the Core Web Vitals, you fail the entire assessment.



FAQ

Does CLS impact my SEO?

Yes, CLS can impact your SEO. CLS is used by Google to evaluate user experience, and starting from May 2021, page experience signals, including CLS, are considered in Google's ranking algorithm. A high CLS score can lead to a poor user experience, affecting the quality of your web pages, potentially resulting in lower rankings in search results.

How can I measure layout shifts in JavaScript?

You can measure layout shifts by using the [PerformanceObserver API](#). You can create a new PerformanceObserver instance and specify the entry types you want to observe, including "layout-shift" entries. Then, you can define a callback function to handle the layout shift entries as they are recorded. Within the callback function, you can access information such as the impact fraction and timing of the layout shift events. This approach enables you to measure and analyze layout shifts dynamically in JavaScript, providing valuable insights into your website's performance.

Should I care about post-load CLS shifts?

Yes! Layout shifts can occur after the initial load and are most often the result of user interactions. As long as they happen within 500 milliseconds of that interaction, these shifts are counted toward your CLS score.

However, be mindful of post-load shifts that occur because of lazy-loaded content or interactions between transitions (in SPAs, for example) outside the 500-millisecond timeframe. Even if they occurred after the initial page load, these are included in your CLS score.