CoGrammar

Welcome to this session:

Skills Bootcamp - Q&A Session on Responsive Design and Performance Optimization

The session will start shortly...

Questions? Drop them in the chat.
We'll have dedicated moderators
answering questions.

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

Skills Bootcamp Cloud Web Development

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Skills Bootcamp Cloud Web Development

- For all **non-academic questions**, please submit a query: ***www.hyperiondev.com/support***
- **Report a safeguarding incident: *www.hyperiondev.com/safeguardreporting***
- We would love your feedback on lectures: ***Feedback on Lectures.***
- Find all the lecture **content** in your **Lecture Backpack** on GitHub.
- If you are hearing impaired, please kindly use your computer's function through Google chrome to enable captions.

Learning Outcomes

- Clarify foundational concepts of responsive web design, including media queries, flexible grids, and fluid images.
- Address common challenges and best practices in optimising web performance (e.g., lazy loading, minification, CDN usage).
- Provide actionable takeaways for integrating responsiveness and performance optimisation into existing projects.

What is the primary purpose of responsive web design?

A. To make websites look identical on all devices.

B. To adapt the website's layout to different screen sizes and devices.

C. To ensure websites work only on desktops.

D. To optimise loading speed.

Which of the following tools is commonly used to test web page performance?

A. Figma
B. Photoshop
C. Google Lighthouse
D. Visual Studio Code

What does "lazy loading" refer to in web performance optimisation?

A. Reducing CSS size.

B. Loading images and other resources only when they are needed.

C. Delaying the loading of the entire website.

D. Compressing JavaScript files.

Question

- What design principles ensure a website is fully responsive across devices?

CoGrammar

Lecture Overview

- → Introduction to Responsive web design
- → Discussing the Importance of Performance
- → Identifying Bottlenecks and Tools
- → Implementing Optimization Techniques

WHAT IS RESPONSIVE WEB DESIGN?

- Responsive design is a method of creating a web application that is able to adapt to different screen resolutions while maintaining interactivity.
- Responsive design approach combines the following components:
  - Flexible design layouts
  - Responsive images and units
  - Media queries

THE VIEWPORT

- ❖ The viewport is the screen size where the web page is in view.
- ❖ CSS has both absolute and relative units of measuring the viewport dimensions.
- ❖ Relative units or dynamic values depend on the screen's size and resolution or the root element's font sizes.

THE VIEWPORT

- Common relative/responsive units are:
  - **em:** relative unit based on the font size of the parent element
  - **rem:** relative unit based on the font size of the root element
  - **vh; vw:** percent of the viewport's height or width
  - **%:** percentage of the parent element

Media queries

- Different media types are:
  - **All:** default, which matches all devices
  - **Print:** used with printers
  - **Screen:** fits devices with a screen
  - **Speech:** fits devices with text-to-speech functionality

Media queries

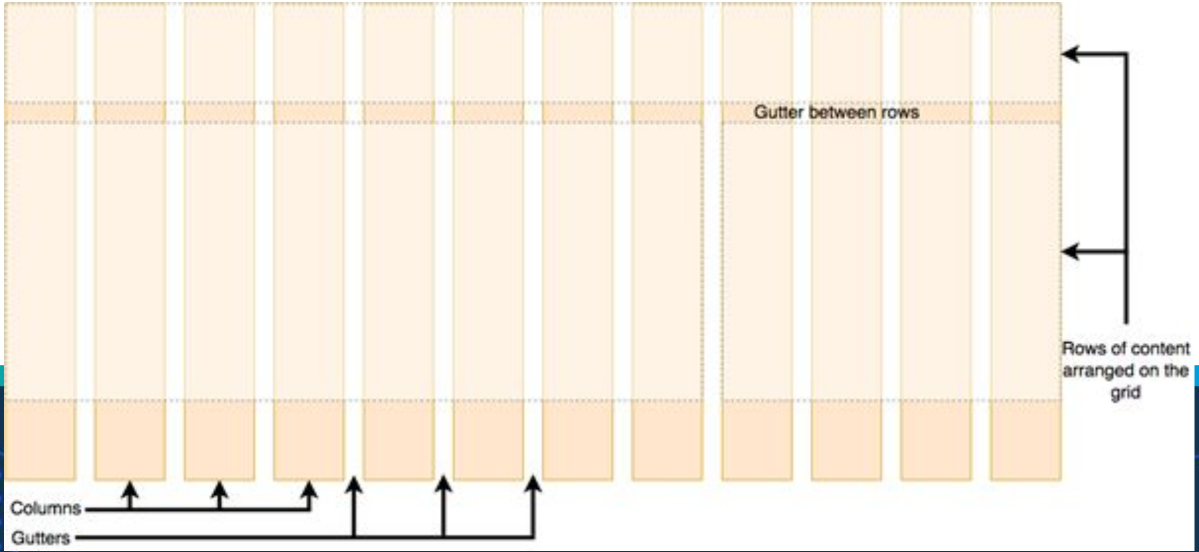
- ❖ Media screen query allows the web page to respond to different screen sizes by applying specific styles based on the screen's viewport dimensions.
- ❖ It helps the page automatically adjust its layout to match the size of the device being used.

Flexible design layouts

- With a flexible design, the widths of page elements will be proportional to the width of the screen or browser window.
- Flexible design ensures that the layout remains consistent.

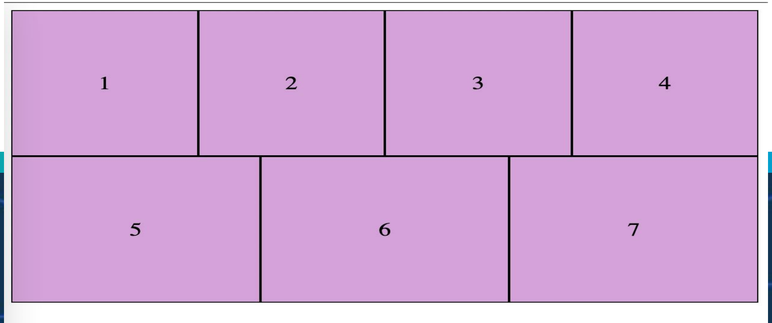
Flexible design layouts

- Grid Layout:
  - A CSS grid is like a table that is designed to make it easier to position elements on a web page
  - The grid usually contains 12 columns, as depicted below:



Flexible design layouts

- Flexbox layout
  - Flexbox is a CSS module designed to more efficiently position multiple elements, even when the size of the contents inside the container is unknown.
  - Items in a flex container expand or shrink to the available space.
- Flexbox container layout:



1	2	3	4
5	6	7	

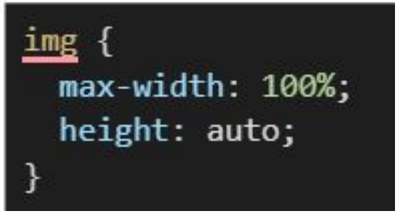
Responsive images

- Responsive images follow the same concept as a fluid layout, using a dynamic unit to control the width or height
- One way to create a responsive image is by setting the img width property to a percentage value
- The percentage unit approximates a single percentage of the viewport's width or height and ensures the image remains in proportion to the screen.

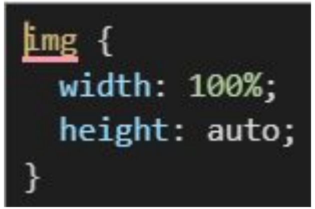
CoGrammar

Responsive images

❖ Examples include the following:



```css
img {
  max-width: 100%;
  height: auto;
}
```

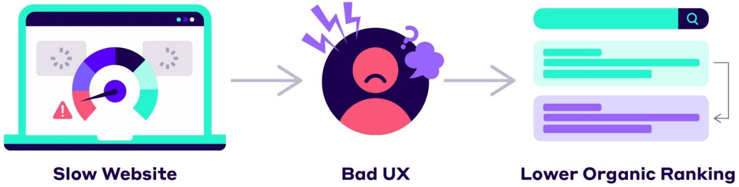


```css
img {
  width: 100%;
  height: auto;
}
```

Let's take a break

The Case for Speed

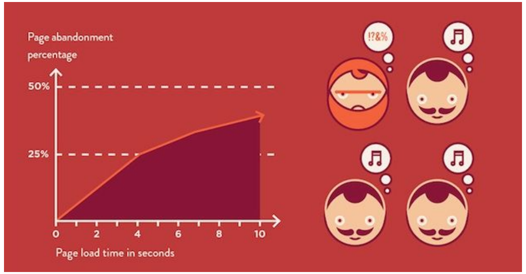
- Statistics:
  - "53% of mobile users abandon a site if it takes longer than 3 seconds to load."
  - "1-second delay reduces conversions by 7%."
- Impact Areas:
  - User Experience
  - SEO and Rankings
  - Conversion Rates and Revenue

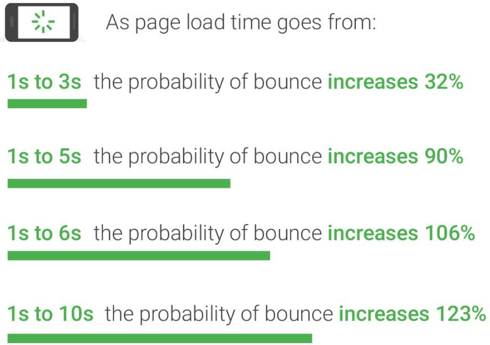


CoGrammar

What Makes a Website Great?

- Amazon’s calculated that a page load slowdown of just one second could cost it $1.6 billion in sales each year.
- Google has calculated that by slowing its search results by just four tenths of a second they could lose 8 million searches per day–meaning they’d serve up many millions fewer online adverts.
- Read more here.





As page load time goes from:

1s to 3s the probability of bounce increases 32%

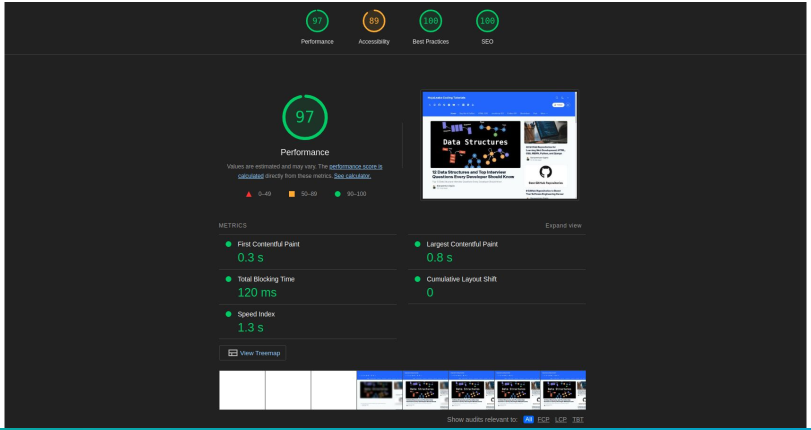
1s to 5s the probability of bounce increases 90%

1s to 6s the probability of bounce increases 106%

1s to 10s the probability of bounce increases 123%

Measuring Website Performance

- Core Web Vitals:
  - Largest Contentful Paint (LCP): < 2.5s.
  - First Input Delay (FID): < 100ms.
  - Cumulative Layout Shift (CLS): < 0.1.
- Additional Metrics:
  - Time to First Byte (TTFB).
  - Speed Index (SI).



97 Performance

89 Accessibility

100 Best Practices

100 SEO

97

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100

METRICS

Expand view

● First Contentful Paint
0.3 s

● Largest Contentful Paint
0.8 s

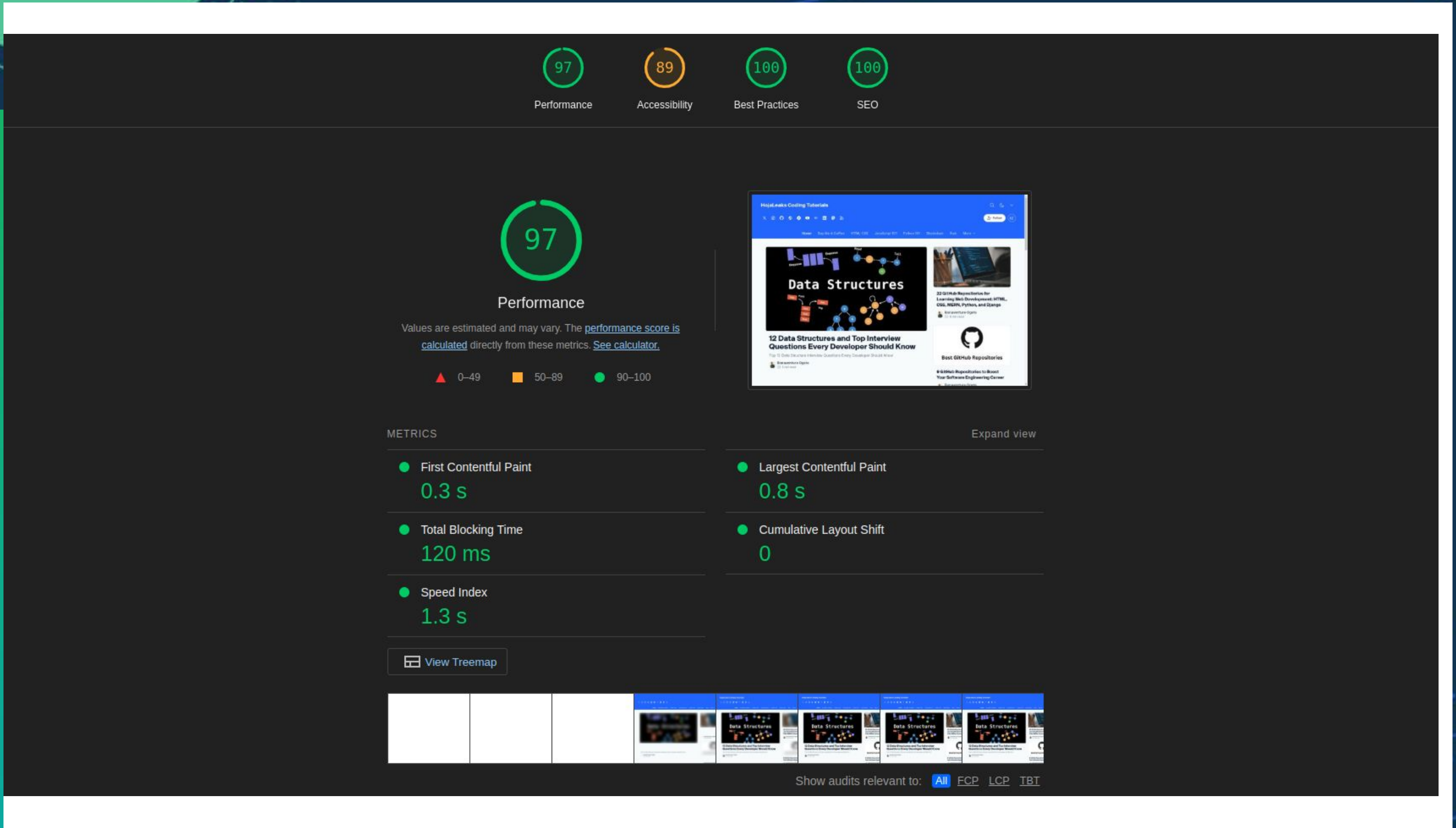
● Total Blocking Time
120 ms

● Cumulative Layout Shift
0

● Speed Index
1.3 s

View Treemap
HojaLeaks Coding Tutorials
Data Structures
12 Data Structures and Top Interview Questions Every Developer Should Know
Best GitHub Repositories
Show audits relevant to: All FCP LCP TBT

Measuring Website Performance



97 Performance
89 Accessibility
100 Best Practices
100 SEO

97

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100

METRICS

Expand view

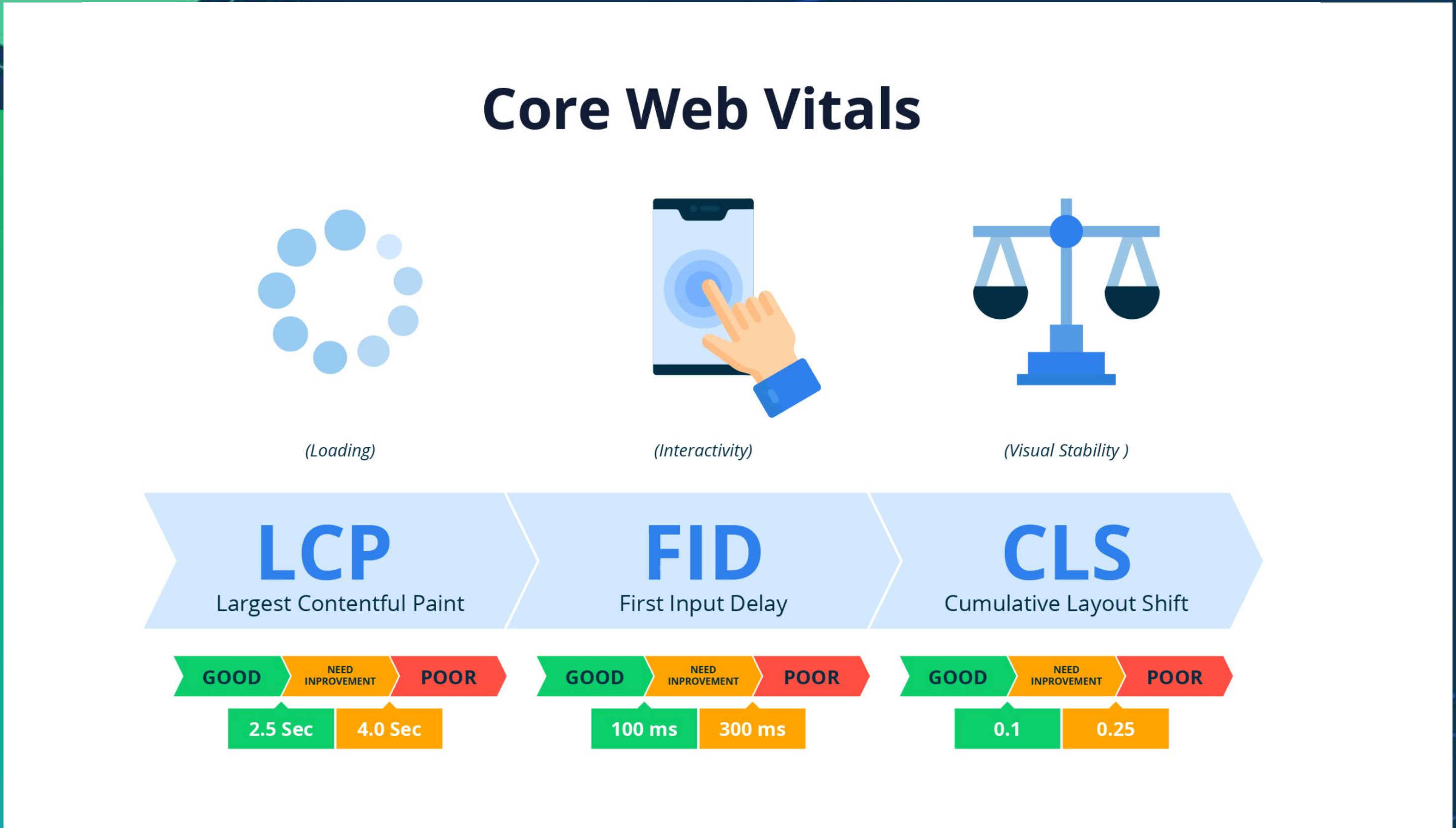
● First Contentful Paint 0.3 s	● Largest Contentful Paint 0.8 s
● Total Blocking Time 120 ms	● Cumulative Layout Shift 0
● Speed Index 1.3 s	

View Treemap

Show audits relevant to: All FCP LCP TBT

CoGrammar

Measuring Website Performance



Core Web Vitals

(Loading)

LCP
Largest Contentful Paint

GOOD | NEED INPROVEMENT | POOR

2.5 Sec | 4.0 Sec

(Interactivity)

FID
First Input Delay

GOOD | NEED INPROVEMENT | POOR

100 ms | 300 ms

(Visual Stability)

CLS
Cumulative Layout Shift

GOOD | NEED INPROVEMENT | POOR

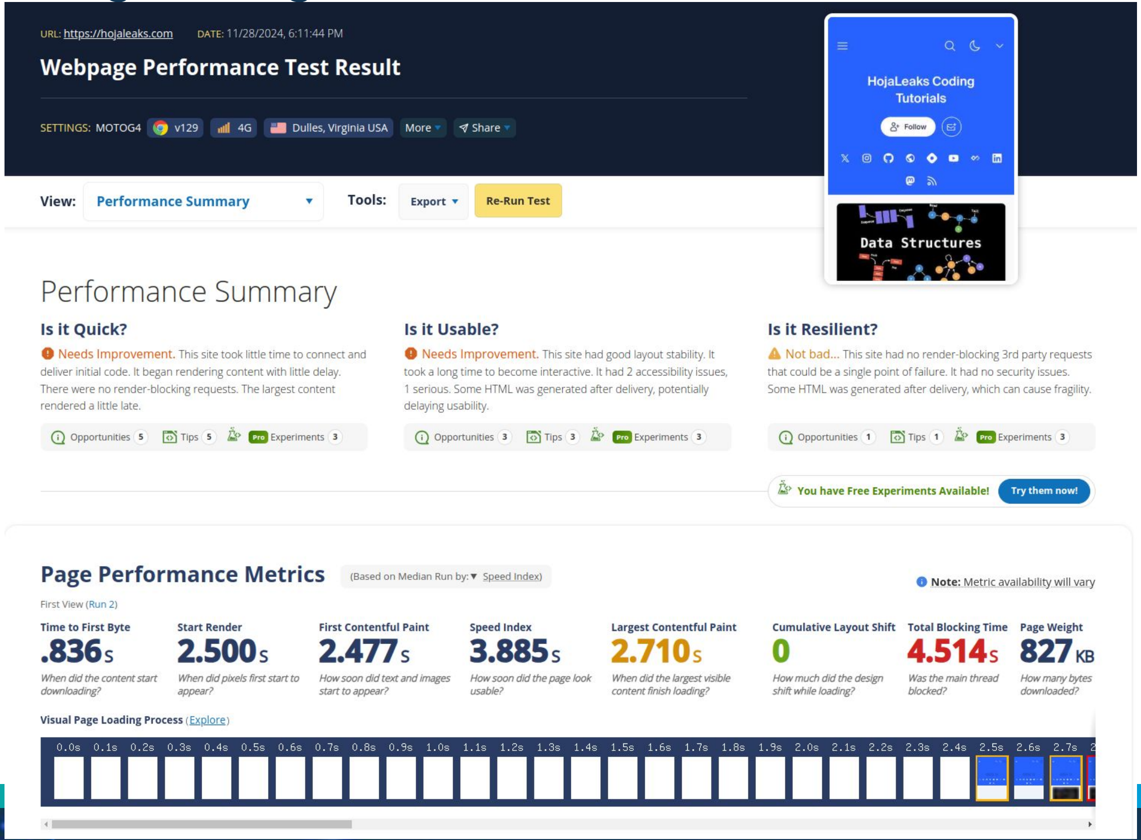
0.1 | 0.25

How to Measure Performance

- Tools Overview:
  - Google Lighthouse (built into Chrome DevTools).
  - WebPageTest.org for deeper analysis.
  - GTmetrix for detailed insights.

How to Measure Performance

❖ WebPageTest.org



URL: https://hojaleaks.com DATE: 11/28/2024, 6:11:44 PM

Webpage Performance Test Result

SETTINGS: MOTOG4 | v129 | 4G | Dulles, Virginia USA | More ▾ | Share ▾

View: Performance Summary ▾ Tools: Export ▾ | Re-Run Test

Performance Summary

Is it Quick?

Needs Improvement. This site took little time to connect and deliver initial code. It began rendering content with little delay. There were no render-blocking requests. The largest content rendered a little late.

Opportunities 5 | Tips 5 | Pro Experiments 3

Is it Usable?

Needs Improvement. This site had good layout stability. It took a long time to become interactive. It had 2 accessibility issues, 1 serious. Some HTML was generated after delivery, potentially delaying usability.

Opportunities 3 | Tips 3 | Pro Experiments 3

Is it Resilient?

Not bad... This site had no render-blocking 3rd party requests that could be a single point of failure. It had no security issues. Some HTML was generated after delivery, which can cause fragility.

Opportunities 1 | Tips 1 | Pro Experiments 3

You have Free Experiments Available! Try them now!

Page Performance Metrics

(Based on Median Run by: ▾ Speed Index)

Note: Metric availability will vary

First View (Run 2)

Time to First Byte	Start Render	First Contentful Paint	Speed Index	Largest Contentful Paint	Cumulative Layout Shift	Total Blocking Time	Page Weight
.836s	2.500s	2.477s	3.885s	2.710s	0	4.514s	827KB
When did the content start downloading?	When did pixels first start to appear?	How soon did text and images start to appear?	How soon did the page look usable?	When did the largest visible content finish loading?	How much did the design shift while loading?	Was the main thread blocked?	How many bytes downloaded?

Visual Page Loading Process (Explore)

0.0s 0.1s 0.2s 0.3s 0.4s 0.5s 0.6s 0.7s 0.8s 0.9s 1.0s 1.1s 1.2s 1.3s 1.4s 1.5s 1.6s 1.7s 1.8s 1.9s 2.0s 2.1s 2.2s 2.3s 2.4s 2.5s 2.6s 2.7s

Why Websites Slow Down

- Top Performance Issues:
  - Large, unoptimized images.
  - Too many or blocking scripts (e.g., JavaScript).
  - Lack of caching strategies.
  - Inefficient server response times.

Improving Website Speed

- Quick Wins:
  - Compress images (e.g., TinyPNG, WebP).
  - Minify CSS, JavaScript, and HTML.
  - Use lazy loading for images and iframes.
- Advanced Techniques:
  - Implement Content Delivery Networks (CDNs).
  - Defer non-critical JavaScript.
  - Optimize server response times with caching and gzip.

Optimizing for Accessibility

- Why Accessibility Matters:
  - Faster sites help users with disabilities.
  - WCAG Guidelines (e.g., color contrast, alt text).
- Tips for Optimized Accessibility:
  - Use semantic HTML.
  - Test with screen readers.

Design for Everyone

- Key Web Content Accessibility Guidelines (WCAG):
  - Ensure sufficient color contrast for text and backgrounds.
  - Provide alt text for all images.
  - Enable keyboard navigation for all interactive elements.
- Common Pitfalls:
  - Hard-to-read text, missing alt text, and inaccessible forms.

Wrap-Up

- What is the one thing you learned?

Which of these strategies is most effective for reducing image size on a website

A. Using JPEG instead of WebP.
B. Using SVG for all images.
C. Serving appropriately sized images for different screen resolutions.
D. Avoiding images altogether.

CoGrammar

What is the purpose of using a Content Delivery Network (CDN)?

A. To increase website loading speed by serving resources from servers closer to users.
B. To compress HTML and CSS files.
C. To protect the website from hacking attempts.
D. To create backups of a website.

When using flexbox for responsive layouts, which property is best for controlling the direction of items on different screen sizes?

A. flex-direction
B. align-items
C. justify-content
D. flex-wrap

Questions and Answers

CoGrammar

Thank you for attending

CoGrammar

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education