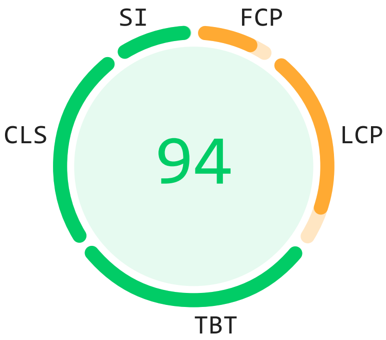
https://rpires71.github.io/milestone1/about-me.html

94 Performance

100 Accessibility

100 Best Practices

100 SEO



Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49 50–89 90–100

Skip to content

About Me

Educator | Technologist | IT Consultant

Hi, I'm **Rob Pires** - a qualified and experienced educator in Computer Science, Mathematics, and Creative iMedia. I'm also a dedicated lifelong learner, always exploring technology and creativity.

From classroom teaching to curriculum design, network administration, and web development, my mission is to empower learners and professionals alike through meaningful technology education and solutions.

METRICS Expand view

First Contentful Paint	Largest Contentful Paint
2.1 s	2.7 s

Total Blocking Time

0 ms

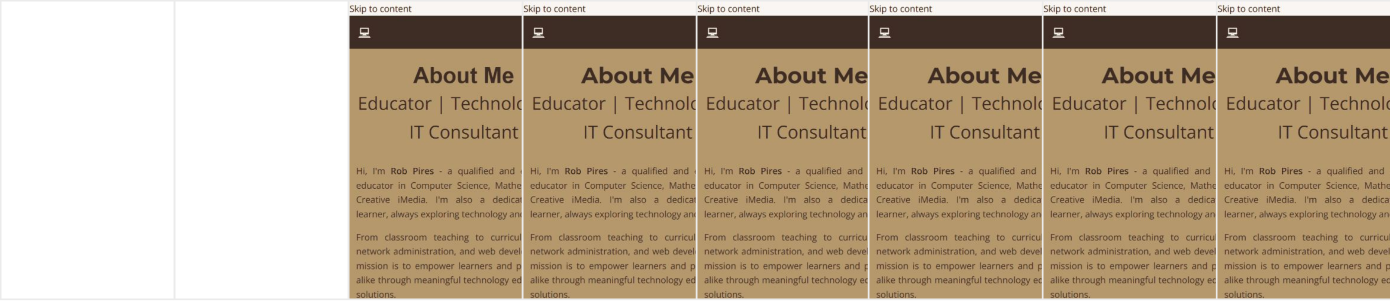
Cumulative Layout Shift

0

Speed Index

2.1 s

View Treemap



Later this year, insights will replace performance audits. Learn more and provide feedback here.

Try insights

Show audits relevant to: All FCP LCP TBT CLS

DIAGNOSTICS

▲ Properly size images — Est savings of 65 KiB

Serve images that are appropriately-sized to save cellular data and improve load time. Learn how to size images. FCP LCP

URL	Resource Size	Est Savings
GitHub Utility 1st Party	**125.3 KiB**	**65.1 KiB**
img.img-fluid ...images/badgev2.webp (rpires71.github.io)	51.0 KiB	18.2 KiB
img.main-illustration.img-fluid ... images/skills_iconsv4.webp (rpires71.github.io)	37.2 KiB	17.6 KiB

	URL	Resource Size	Est Savings
img.section-icon	...images/skill_icon2.webp (rpires71.github.io)	11.9 KiB	8.6 KiB
img.logo.logo-navbar	...images/logodbg2.webp (rpires71.github.io)	8.6 KiB	8.6 KiB
img.section-icon	...images/lifelong_learning_icon2.webp (rpires71.github.io)	10.8 KiB	7.8 KiB
img.section-icon	...images/pro_bio2.webp (rpires71.github.io)	5.8 KiB	4.2 KiB

▲ Eliminate render-blocking resources — Est savings of 970 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

URL	Transfer Size	Est Savings
JSDelivr CDN Cdn	**27.1 KiB**	**1,050 ms**
...css/bootstrap.min.css (cdn.jsdelivr.net)	27.1 KiB	1,050 ms
Google Fonts Cdn	**2.2 KiB**	**870 ms**
/css2?family=... (fonts.googleapis.com)	2.2 KiB	870 ms

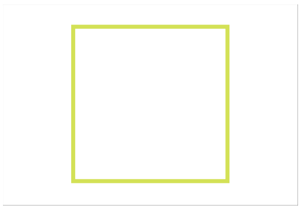
▲ Reduce unused CSS — Est savings of 26 KiB

Reduce unused rules from stylesheets and defer CSS not used for above-the-fold content to decrease bytes consumed by network activity. [Learn how to reduce unused CSS](). FCP LCP

URL	Transfer Size	Est Savings
JSDelivr CDN Cdn	**26.8 KiB**	**25.8 KiB**
…css/bootstrap.min.css (cdn.jsdelivr.net)	26.8 KiB	25.8 KiB

▲ Largest Contentful Paint element — 2,700 ms

This is the largest contentful element painted within the viewport. [Learn more about the Largest Contentful Paint element]() LCP

Element	
	img.section-icon

Phase	% of LCP	Timing
TTFB	33%	900 ms
Load Delay	0%	0 ms
Load Time	25%	680 ms
Render Delay	42%	1,120 ms

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

	URL
GitHub Utility 1st Party	
img.logo.logo-navbar	...images/logodbg2.webp (rpires71.github.io)
img.main-illustration.img-fluid	...images/skills_iconsv4.webp (rpires71.github.io)
img.logo.logo-header.img-fluid	...images/badgev2.webp (rpires71.github.io)
img.img-fluid	...images/badgev2.webp (rpires71.github.io)
img.section-icon	...images/pro_bio2.webp (rpires71.github.io)
img.section-icon	...images/skill_icon2.webp (rpires71.github.io)
img.section-icon	...images/lifelong_learning_icon2.webp (rpires71.github.io)
img.footer-icon.me-2	...icons/award.svg (rpires71.github.io)

		URL
	img.footer-icon.me-2	...icons/map-pin.svg (rpires71.github.io)
	img.footer-icon.me-2	...icons/send.svg (rpires71.github.io)
	img.footer-icon.me-2	...icons/phone.svg (rpires71.github.io)
	img.footer-icon.me-2	...icons/message-circle.svg (rpires71.github.io)
	img.social-icon.me-2	...icons/linkedin.svg (rpires71.github.io)
	img.social-icon.me-2	...icons/facebook.svg (rpires71.github.io)
	img.social-icon	...icons/instagram.svg (rpires71.github.io)

Minify CSS — Est savings of 5 KiB

Minifying CSS files can reduce network payload sizes. Learn how to minify CSS. FCP LCP

URL	Transfer Size	Est Savings
GitHub Utility 1st Party	**7.3 KiB**	**4.8 KiB**
…css/style.css (rpires71.github.io)	7.3 KiB	4.8 KiB

Serve static assets with an efficient cache policy — 15 resources found

A long cache lifetime can speed up repeat visits to your page. Learn more about efficient cache policies.

URL	Cache TTL	Transfer Size
GitHub Utility 1st Party		**137 KiB**
…images/badgev2.webp (rpires71.github.io)	10m	51 KiB
…images/skills_iconsv4.webp (rpires71.github.io)	10m	37 KiB
…images/skill_icon2.webp (rpires71.github.io)	10m	12 KiB
…images/lifelong_learning_icon2.webp (rpires71.github.io)	10m	11 KiB
…images/logodbg2.webp (rpires71.github.io)	10m	9 KiB
…css/style.css (rpires71.github.io)	10m	7 KiB
…images/pro_bio2.webp (rpires71.github.io)	10m	6 KiB
…icons/phone.svg (rpires71.github.io)	10m	1 KiB
…icons/award.svg (rpires71.github.io)	10m	1 KiB
…icons/message-circle.svg (rpires71.github.io)	10m	0 KiB
…icons/linkedin.svg (rpires71.github.io)	10m	0 KiB
…icons/instagram.svg (rpires71.github.io)	10m	0 KiB
…icons/map-pin.svg (rpires71.github.io)	10m	0 KiB

URL	Cache TTL	Transfer Size
...icons/facebook.svg (rpires71.github.io)	10m	0 KiB
...icons/send.svg (rpires71.github.io)	10m	0 KiB

○ Avoid chaining critical requests — 4 chains found

The Critical Request Chains below show you what resources are loaded with a high priority. Consider reducing the length of chains, reducing the download size of resources, or deferring the download of unnecessary resources to improve page load. [Learn how to avoid chaining critical requests](#).

Maximum critical path latency: **1,124.646 ms**

Initial Navigation

- /milestone1/about-me.html (rpires71.github.io)
  - ...css/bootstrap.min.css (cdn.jsdelivr.net) - **163.402 ms,** 27.11 KiB
  - ...css/style.css (rpires71.github.io)
    - /css2?family=... (fonts.googleapis.com)
      - ...v43/memvYaGs1....woff2 (fonts.gstatic.com) - **162.372 ms,** 42.12 KiB
      - ...v30/JTUSjIg1_....woff2 (fonts.gstatic.com) - **345.764 ms,** 34.67 KiB
  - ...js/bootstrap.bundle.min.js (cdn.jsdelivr.net) - **166.613 ms,** 22.56 KiB

○ Minimize third-party usage — Third-party code blocked the main thread for 0 ms

Third-party code can significantly impact load performance. Limit the number of redundant third-party providers and try to load third-party code after your page has primarily finished loading. [Learn how to minimize third-party impact](#). TBT

Third-Party	Transfer Size	Main-Thread Blocking Time
Google Fonts Cdn	**79 KiB**	**0 ms**
...v43/memvYaGs1....woff2 (fonts.gstatic.com)	42 KiB	0 ms
...v30/JTUSjIg1_....woff2 (fonts.gstatic.com)	35 KiB	0 ms
/css2?family=... (fonts.googleapis.com)	2 KiB	0 ms
JSDelivr CDN Cdn	**50 KiB**	**0 ms**
...css/bootstrap.min.css (cdn.jsdelivr.net)	27 KiB	0 ms
...js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	23 KiB	0 ms

○ Avoid long main-thread tasks — 4 long tasks found

Lists the longest tasks on the main thread, useful for identifying worst contributors to input delay. Learn how to avoid long main-thread tasks TBT

URL	Start Time	Duration
Unattributable		**200 ms**
`Unattributable`	1,115 ms	135 ms
`Unattributable`	1,050 ms	65 ms
GitHub Utility 1st Party		**146 ms**
/milestone1/about-me.html (rpires71.github.io)	1,313 ms	83 ms
/milestone1/about-me.html (rpires71.github.io)	1,250 ms	63 ms

More information about the performance of your application. These numbers don't directly affect the Performance score.

PASSED AUDITS (28) Hide

Defer offscreen images

Consider lazy-loading offscreen and hidden images after all critical resources have finished loading to lower time to interactive. Learn how to defer offscreen images. FCP LCP

Minify JavaScript

Minifying JavaScript files can reduce payload sizes and script parse time. Learn how to minify JavaScript. FCP LCP

Reduce unused JavaScript

Reduce unused JavaScript and defer loading scripts until they are required to decrease bytes consumed by network activity. Learn how to reduce unused JavaScript. FCP LCP

Efficiently encode images

Optimized images load faster and consume less cellular data. Learn how to efficiently encode images. FCP LCP

Serve images in next-gen formats

Image formats like WebP and AVIF often provide better compression than PNG or JPEG, which means faster downloads and less data consumption. Learn more about modern image formats. FCP LCP

Enable text compression

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. Learn more about text compression. FCP LCP

Preconnect to required origins

Consider adding `preconnect` or `dns-prefetch` resource hints to establish early connections to important third-party origins. Learn how to preconnect to required origins. LCP FCP

Initial server response time was short — Root document took 160 ms

Keep the server response time for the main document short because all other requests depend on it. Learn more about the Time to First Byte metric. FCP LCP

URL	Time Spent
GitHub Utility 1st Party	**160 ms**
/milestone1/about-me.html (rpires71.github.io)	160 ms

Avoid multiple page redirects

Redirects introduce additional delays before the page can be loaded. Learn how to avoid page redirects. LCP FCP

Use HTTP/2

HTTP/2 offers many benefits over HTTP/1.1, including binary headers and multiplexing. Learn more about HTTP/2. LCP FCP

Use video formats for animated content

Large GIFs are inefficient for delivering animated content. Consider using MPEG4/WebM videos for animations and PNG/WebP for static images instead of GIF to save network bytes. Learn more about efficient video formats FCP LCP

Remove duplicate modules in JavaScript bundles

Remove large, duplicate JavaScript modules from bundles to reduce unnecessary bytes consumed by network activity. FCP LCP

Avoid serving legacy JavaScript to modern browsers

Polyfills and transforms enable legacy browsers to use new JavaScript features. However, many aren't necessary for modern browsers. Consider modifying your JavaScript build process to not transpile Baseline features, unless you know you must support legacy browsers. Learn why most sites can deploy ES6+ code without transpiling FCP LCP

Preload Largest Contentful Paint image

If the LCP element is dynamically added to the page, you should preload the image in order to improve LCP. Learn more about preloading LCP elements. LCP

Avoids enormous network payloads — Total size was 272 KiB

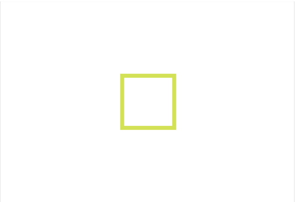
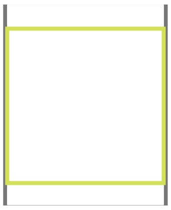
Large network payloads cost users real money and are highly correlated with long load times. Learn how to reduce payload sizes.

☑ Show 3rd-party resources (4)

URL	Transfer Size
GitHub Utility 1st Party	**127.6 KiB**
...images/badgev2.webp (rpires71.github.io)	51.2 KiB
...images/skills_iconsv4.webp (rpires71.github.io)	37.3 KiB
...images/skill_icon2.webp (rpires71.github.io)	12.0 KiB
...images/lifelong_learning_icon2.webp (rpires71.github.io)	11.0 KiB
...images/logodbg2.webp (rpires71.github.io)	8.8 KiB
...css/style.css (rpires71.github.io)	7.3 KiB
Google Fonts Cdn	**76.8 KiB**
...v43/memvYaGs1....woff2 (fonts.gstatic.com)	42.1 KiB
...v30/JTUSjIg1_....woff2 (fonts.gstatic.com)	34.7 KiB
JSDelivr CDN Cdn	**49.7 KiB**
...css/bootstrap.min.css (cdn.jsdelivr.net)	27.1 KiB
...js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	22.6 KiB

Avoids an excessive DOM size — 121 elements

A large DOM will increase memory usage, cause longer style calculations, and produce costly layout reflows. Learn how to avoid an excessive DOM size. TBT

Statistic	Element	Value
Total DOM Elements		121
Maximum DOM Depth	 img.social-icon.me-2	8
Maximum Child Elements	 div.footer-tile	10

○ User Timing marks and measures

Consider instrumenting your app with the User Timing API to measure your app's real-world performance during key user experiences. Learn more about User Timing marks.

JavaScript execution time — 0.0 s

Consider reducing the time spent parsing, compiling, and executing JS. You may find delivering smaller JS payloads helps with this. Learn how to reduce Javascript execution time. TBT

URL	Total CPU Time	Script Evaluation	Script Parse
Unattributable	**479 ms**	**14 ms**	**0 ms**
`Unattributable`	479 ms	14 ms	0 ms
GitHub Utility 1st Party	**408 ms**	**26 ms**	**4 ms**
/milestone1/about-me.html (rpires71.github.io)	408 ms	26 ms	4 ms

Minimizes main-thread work — 1.0 s

Consider reducing the time spent parsing, compiling and executing JS. You may find delivering smaller JS payloads helps with this. Learn how to minimize main-thread work TBT

Category	Time Spent
Other	575 ms
Style & Layout	231 ms
Script Evaluation	61 ms
Rendering	52 ms
Parse HTML & CSS	43 ms
Script Parsing & Compilation	19 ms

All text remains visible during webfont loads

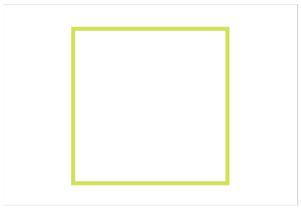
Leverage the `font-display` CSS feature to ensure text is user-visible while webfonts are loading. [Learn more about `font-display`].

○ Lazy load third-party resources with facades

Some third-party embeds can be lazy loaded. Consider replacing them with a facade until they are required. [Learn how to defer third-parties with a facade]. TBT

Largest Contentful Paint image was not lazily loaded

Above-the-fold images that are lazily loaded render later in the page lifecycle, which can delay the largest contentful paint. [Learn more about optimal lazy loading]. LCP

Element
 img.section-icon

○ Avoid large layout shifts

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

Uses passive listeners to improve scrolling performance

Consider marking your touch and wheel event listeners as `passive` to improve your page's scroll performance. Learn more about adopting passive event listeners.

Avoids `document.write()`

For users on slow connections, external scripts dynamically injected via `document.write()` can delay page load by tens of seconds. Learn how to avoid document.write().

○ Avoid non-composited animations

Animations which are not composited can be janky and increase CLS. Learn how to avoid non-composited animations CLS

Has a `<meta name="viewport">` tag with `width` or `initial-scale`

A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents a 300 millisecond delay to user input. Learn more about using the viewport meta tag.

Page didn't prevent back/forward cache restoration

Many navigations are performed by going back to a previous page, or forwards again. The back/forward cache (bfcache) can speed up these return navigations. Learn more about the bfcache

100

Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so manual testing is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (10) Hide

○ Interactive controls are keyboard focusable

Custom interactive controls are keyboard focusable and display a focus indicator. Learn how to make custom controls focusable.

○ Interactive elements indicate their purpose and state

Interactive elements, such as links and buttons, should indicate their state and be distinguishable from non-interactive elements. Learn how to decorate interactive elements with affordance hints.

○ The page has a logical tab order

Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. Learn more about logical tab ordering.

○ Visual order on the page follows DOM order

DOM order matches the visual order, improving navigation for assistive technology. Learn more about DOM and visual ordering.

○ User focus is not accidentally trapped in a region

A user can tab into and out of any control or region without accidentally trapping their focus. Learn how to avoid focus traps.

○ The user's focus is directed to new content added to the page

If new content, such as a dialog, is added to the page, the user's focus is directed to it. Learn how to direct focus to new content.

○ HTML5 landmark elements are used to improve navigation

Landmark elements (`<main>`, `<nav>`, etc.) are used to improve the keyboard navigation of the page for assistive technology. Learn more about landmark elements.

○ Offscreen content is hidden from assistive technology

Offscreen content is hidden with display: none or aria-hidden=true. Learn how to properly hide offscreen content.

○ Custom controls have associated labels

Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. Learn more about custom controls and labels.

○ Custom controls have ARIA roles

Custom interactive controls have appropriate ARIA roles. Learn how to add roles to custom controls.

These items address areas which an automated testing tool cannot cover. Learn more in our guide on conducting an accessibility review.

PASSED AUDITS (19) Hide

`[aria-*]` attributes match their roles

Each ARIA `role` supports a specific subset of `aria-*` attributes. Mismatching these invalidates the `aria-*` attributes. Learn how to match ARIA attributes to their roles.

`[aria-hidden="true"]` is not present on the document `<body>`

Assistive technologies, like screen readers, work inconsistently when `aria-hidden="true"` is set on the document `<body>`. Learn how `aria-hidden` affects the document body.

`[aria-*]` attributes have valid values

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. Learn more about valid values for ARIA attributes.

`[aria-*]` attributes are valid and not misspelled

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. Learn more about valid ARIA attributes.

Buttons have an accessible name

When a button doesn't have an accessible name, screen readers announce it as "button", making it unusable for users who rely on screen readers. Learn how to make buttons more accessible.

Image elements have `[alt]` attributes

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. Learn more about the `alt` attribute.

`[user-scalable="no"]` is not used in the `<meta name="viewport">` element and the `[maximum-scale]` attribute is not less than 5.

Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. Learn more about the viewport meta tag.

ARIA attributes are used as specified for the element's role

Some ARIA attributes are only allowed on an element under certain conditions. Learn more about conditional ARIA attributes.

`[aria-hidden="true"]` elements do not contain focusable descendents

Focusable descendents within an `[aria-hidden="true"]` element prevent those interactive elements from being available to users of assistive technologies like screen readers. Learn how `aria-hidden` affects focusable elements.

Elements use only permitted ARIA attributes

Using ARIA attributes in roles where they are prohibited can mean that important information is not communicated to users of assistive technologies. Learn more about prohibited ARIA roles.

Background and foreground colors have a sufficient contrast ratio

Low-contrast text is difficult or impossible for many users to read. Learn how to provide sufficient color contrast.

Document has a `<title>` element

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. Learn more about document titles.

`<html>` element has a `[lang]` attribute

If a page doesn't specify a `lang` attribute, a screen reader assumes that the page is in the default language that the user chose when setting up the screen reader. If the page isn't actually in the default language, then the screen reader might not announce the page's text correctly. Learn more about the `lang` attribute.

`<html>` element has a valid value for its `[lang]` attribute

Specifying a valid BCP 47 language helps screen readers announce text properly. Learn how to use the `lang` attribute.

Links have a discernible name

Link text (and alternate text for images, when used as links) that is discernible, unique, and focusable improves the navigation experience for screen reader users. Learn how to make links accessible.

No element has a `[tabindex]` value greater than 0

A value greater than 0 implies an explicit navigation ordering. Although technically valid, this often creates frustrating experiences for users who rely on assistive technologies. Learn more about the `tabindex` attribute.

Touch targets have sufficient size and spacing.

Touch targets with sufficient size and spacing help users who may have difficulty targeting small controls to activate the targets. Learn more about touch targets.

Heading elements appear in a sequentially-descending order

Properly ordered headings that do not skip levels convey the semantic structure of the page, making it easier to navigate and understand when using assistive technologies. Learn more about heading order.

Image elements do not have `[alt]` attributes that are redundant text.

Informative elements should aim for short, descriptive alternative text. Alternative text that is exactly the same as the text adjacent to the link or image is potentially confusing for screen reader users, because the text will be read twice. Learn more about the `alt` attribute.

NOT APPLICABLE (38) Hide

○ `[accesskey]` values are unique

Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. Learn more about access keys.

○ Uses ARIA roles only on compatible elements

Many HTML elements can only be assigned certain ARIA roles. Using ARIA roles where they are not allowed can interfere with the accessibility of the web page. Learn more about ARIA roles.

○ `button`, `link`, and `menuitem` elements have accessible names

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn how to make command elements more accessible.

○ Deprecated ARIA roles were not used

Deprecated ARIA roles may not be processed correctly by assistive technology. Learn more about deprecated ARIA roles.

○ Elements with `role="dialog"` or `role="alertdialog"` have accessible names.

ARIA dialog elements without accessible names may prevent screen readers users from discerning the purpose of these elements. Learn how to make ARIA dialog elements more accessible.

○ ARIA input fields have accessible names

When an input field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more about input field labels.

○ ARIA `meter` elements have accessible names

When a meter element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn how to name `meter` elements.

○ ARIA `progressbar` elements have accessible names

When a `progressbar` element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn how to label `progressbar` elements.

○ `[role]`s have all required `[aria-*]` attributes

Some ARIA roles have required attributes that describe the state of the element to screen readers. Learn more about roles and required attributes.

○ Elements with an ARIA `[role]` that require children to contain a specific `[role]` have all required children.

Some ARIA parent roles must contain specific child roles to perform their intended accessibility functions. Learn more about roles and required children elements.

○ `[role]`s are contained by their required parent element

Some ARIA child roles must be contained by specific parent roles to properly perform their intended accessibility functions. Learn more about ARIA roles and required parent element.

○ `[role]` values are valid

ARIA roles must have valid values in order to perform their intended accessibility functions. Learn more about valid ARIA roles.

○ Elements with the `role=text` attribute do not have focusable descendents.

Adding `role=text` around a text node split by markup enables VoiceOver to treat it as one phrase, but the element's focusable descendents will not be announced. Learn more about the `role=text` attribute.

○ ARIA toggle fields have accessible names

When a toggle field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more about toggle fields.

○ ARIA `tooltip` elements have accessible names

When a tooltip element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn how to name `tooltip` elements.

○ ARIA `treeitem` elements have accessible names

When a `treeitem` element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more about labeling `treeitem` elements.

○ The page contains a heading, skip link, or landmark region

Adding ways to bypass repetitive content lets keyboard users navigate the page more efficiently. Learn more about bypass blocks.

○ `<dl>`'s contain only properly-ordered `<dt>` and `<dd>` groups, `<script>`, `<template>` or `<div>` elements.

When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. Learn how to structure definition lists correctly.

○ Definition list items are wrapped in `<dl>` elements

Definition list items (`<dt>` and `<dd>`) must be wrapped in a parent `<dl>` element to ensure that screen readers can properly announce them. Learn how to structure definition lists correctly.

○ ARIA IDs are unique

The value of an ARIA ID must be unique to prevent other instances from being overlooked by assistive technologies. Learn how to fix duplicate ARIA IDs.

○ No form fields have multiple labels

Form fields with multiple labels can be confusingly announced by assistive technologies like screen readers which use either the first, the last, or all of the labels. Learn how to use form labels.

○ `<frame>` or `<iframe>` elements have a title

Screen reader users rely on frame titles to describe the contents of frames. Learn more about frame titles.

○ `<html>` element has an `[xml:lang]` attribute with the same base language as the `[lang]` attribute.

If the webpage does not specify a consistent language, then the screen reader might not announce the page's text correctly. Learn more about the `lang` attribute.

○ Input buttons have discernible text.

Adding discernable and accessible text to input buttons may help screen reader users understand the purpose of the input button. Learn more about input buttons.

○ `<input type="image">` elements have `[alt]` text

When an image is being used as an `<input>` button, providing alternative text can help screen reader users understand the purpose of the button. Learn about input image alt text.

○ Form elements have associated labels

Labels ensure that form controls are announced properly by assistive technologies, like screen readers. Learn more about form element labels.

○ Links are distinguishable without relying on color. ⌃

Low-contrast text is difficult or impossible for many users to read. Link text that is discernible improves the experience for users with low vision. Learn how to make links distinguishable.

○ Lists contain only `<li>` elements and script supporting elements (`<script>` and `<template>`). ⌃

Screen readers have a specific way of announcing lists. Ensuring proper list structure aids screen reader output. Learn more about proper list structure.

○ List items (`<li>`) are contained within `<ul>`, `<ol>` or `<menu>` parent elements ⌃

Screen readers require list items (`<li>`) to be contained within a parent `<ul>`, `<ol>` or `<menu>` to be announced properly. Learn more about proper list structure.

○ The document does not use `<meta http-equiv="refresh">` ⌃

Users do not expect a page to refresh automatically, and doing so will move focus back to the top of the page. This may create a frustrating or confusing experience. Learn more about the refresh meta tag.

○ `<object>` elements have alternate text ⌃

Screen readers cannot translate non-text content. Adding alternate text to `<object>` elements helps screen readers convey meaning to users. Learn more about alt text for `object` elements.

○ Select elements have associated label elements. ⌃

Form elements without effective labels can create frustrating experiences for screen reader users. Learn more about the `select` element.

○ Skip links are focusable. ⌃

Including a skip link can help users skip to the main content to save time. Learn more about skip links.

○ Tables have different content in the summary attribute and `<caption>`. ⌃

The summary attribute should describe the table structure, while `<caption>` should have the onscreen title. Accurate table mark-up helps users of screen readers. Learn more about summary and caption.

○ Cells in a `<table>` element that use the `[headers]` attribute refer to table cells within the same table. ⌃

Screen readers have features to make navigating tables easier. Ensuring `<td>` cells using the `[headers]` attribute only refer to other cells in the same table may improve the experience for screen reader users. Learn more about the `headers` attribute.

○ `<th>` elements and elements with `[role="columnheader"/"rowheader"]` have data cells they describe.

Screen readers have features to make navigating tables easier. Ensuring table headers always refer to some set of cells may improve the experience for screen reader users. Learn more about table headers.

○ `[lang]` attributes have a valid value

Specifying a valid BCP 47 language on elements helps ensure that text is pronounced correctly by a screen reader. Learn how to use the `lang` attribute.

○ `<video>` elements contain a `<track>` element with `[kind="captions"]`

When a video provides a caption it is easier for deaf and hearing impaired users to access its information. Learn more about video captions.

100

Best Practices

TRUST AND SAFETY

○ Ensure CSP is effective against XSS attacks

A strong Content Security Policy (CSP) significantly reduces the risk of cross-site scripting (XSS) attacks. Learn how to use a CSP to prevent XSS

Description	Directive	Severity
No CSP found in enforcement mode		High

○ Use a strong HSTS policy

Deployment of the HSTS header significantly reduces the risk of downgrading HTTP connections and eavesdropping attacks. A rollout in stages, starting with a low max-age is recommended. Learn more about using a strong HSTS policy.

Description	Directive	Severity
No HSTS header found		High

○ Ensure proper origin isolation with COOP

The Cross-Origin-Opener-Policy (COOP) can be used to isolate the top-level window from other documents such as pop-ups. Learn more about deploying the COOP header.

Description	Directive	Severity
No COOP header found		High

○ Mitigate clickjacking with XFO or CSP

The `X-Frame-Options` (XFO) header or the `frame-ancestors` directive in the `Content-Security-Policy` (CSP) header control where a page can be embedded. These can mitigate clickjacking attacks by blocking some or all sites from embedding the page. Learn more about mitigating clickjacking.

Description	Severity
No frame control policy found	High

PASSED AUDITS (15) Hide

Uses HTTPS

All sites should be protected with HTTPS, even ones that don't handle sensitive data. This includes avoiding mixed content, where some resources are loaded over HTTP despite the initial request being served over HTTPS. HTTPS prevents intruders from tampering with or passively listening in on the communications between your app and your users, and is a prerequisite for HTTP/2 and many new web platform APIs. Learn more about HTTPS.

Avoids deprecated APIs

Deprecated APIs will eventually be removed from the browser. Learn more about deprecated APIs.

Avoids third-party cookies

Third-party cookies may be blocked in some contexts. Learn more about preparing for third-party cookie restrictions.

Allows users to paste into input fields

Preventing input pasting is a bad practice for the UX, and weakens security by blocking password managers.Learn more about user-friendly input fields.

Avoids requesting the geolocation permission on page load

Users are mistrustful of or confused by sites that request their location without context. Consider tying the request to a user action instead. Learn more about the geolocation permission.

Avoids requesting the notification permission on page load

Users are mistrustful of or confused by sites that request to send notifications without context. Consider tying the request to user gestures instead. Learn more about responsibly getting permission for notifications.

Displays images with correct aspect ratio

Image display dimensions should match natural aspect ratio. Learn more about image aspect ratio.

Serves images with appropriate resolution

Image natural dimensions should be proportional to the display size and the pixel ratio to maximize image clarity. Learn how to provide responsive images.

Has a `<meta name="viewport">` tag with `width` or `initial-scale`

A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents a 300 millisecond delay to user input. Learn more about using the viewport meta tag.

Document uses legible font sizes — 100% legible text

Font sizes less than 12px are too small to be legible and require mobile visitors to “pinch to zoom” in order to read. Strive to have >60% of page text ≥12px. Learn more about legible font sizes.

Source	Selector	% of Page Text	Font Size
`Legible text`		100.00%	≥ 12px

Page has the HTML doctype

Specifying a doctype prevents the browser from switching to quirks-mode. Learn more about the doctype declaration.

Properly defines charset

A character encoding declaration is required. It can be done with a `<meta>` tag in the first 1024 bytes of the HTML or in the Content-Type HTTP response header. Learn more about declaring the character encoding.

No browser errors logged to the console

Errors logged to the console indicate unresolved problems. They can come from network request failures and other browser concerns. Learn more about this errors in console diagnostic audit

No issues in the `Issues` panel in Chrome Devtools

Issues logged to the `Issues` panel in Chrome Devtools indicate unresolved problems. They can come from network request failures, insufficient security controls, and other browser concerns. Open up the Issues panel in Chrome DevTools for more details on each issue.

Page has valid source maps

Source maps translate minified code to the original source code. This helps developers debug in production. In addition, Lighthouse is able to provide further insights. Consider deploying source maps to take advantage of these benefits. Learn more about source maps.

URL	Map URL
JSDelivr CDN Cdn	
…js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	…js/bootstrap.bundle.min.js.map (cdn.jsdelivr.net)

NOT APPLICABLE (2) Hide

○ Redirects HTTP traffic to HTTPS

Make sure that you redirect all HTTP traffic to HTTPS in order to enable secure web features for all your users. Learn more.

○ Detected JavaScript libraries

All front-end JavaScript libraries detected on the page. Learn more about this JavaScript library detection diagnostic audit.

100

SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more about Google Search Essentials.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)

Hide

○ Structured data is valid

Run the Structured Data Testing Tool and the Structured Data Linter to validate structured data. Learn more about Structured Data.

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (8)

Hide

Page isn’t blocked from indexing

Search engines are unable to include your pages in search results if they don't have permission to crawl them. Learn more about crawler directives.

Document has a `<title>` element

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. Learn more about document titles.

Document has a meta description

Meta descriptions may be included in search results to concisely summarize page content. Learn more about the meta description.

Page has successful HTTP status code

Pages with unsuccessful HTTP status codes may not be indexed properly. Learn more about HTTP status codes.

Links have descriptive text

Descriptive link text helps search engines understand your content. Learn how to make links more accessible.

Links are crawlable

Search engines may use `href` attributes on links to crawl websites. Ensure that the `href` attribute of anchor elements links to an appropriate destination, so more pages of the site can be discovered. Learn how to make links crawlable

Image elements have `[alt]` attributes

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. Learn more about the `alt` attribute.

Document has a valid `hreflang`

hreflang links tell search engines what version of a page they should list in search results for a given language or region. Learn more about `hreflang`.

NOT APPLICABLE (2) Hide

○ robots.txt is valid

If your robots.txt file is malformed, crawlers may not be able to understand how you want your website to be crawled or indexed. Learn more about robots.txt.

○ Document has a valid `rel=canonical`

Canonical links suggest which URL to show in search results. Learn more about canonical links.

Captured at Aug 19, 2025, 10:44 PM GMT
Initial page load

Emulated Moto G Power with Lighthouse 12.6.1
Slow 4G throttling

Single page session

Using Chromium 139.0.0.0 with devtools

Generated by **Lighthouse** 12.6.1 | File an issue