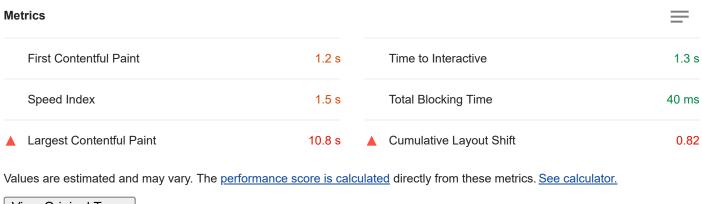




Performance







Opportunities — These suggestions can help your page load faster. They don't directly affect the Performance score.

Estimated Savings Opportunity 9.4 s ^ Properly size images Serve images that are appropriately-sized to save cellular data and improve load time. Learn more. Show 3rd-party resources (3) Resource Potential **URL** Size Savings /public/image-beach1.jpg (127.0.0.1) 4,029.8 KiB 3,971.9 KiB

	URL	Resource Size	Potential Savings
	/public/image-beach3.jpg (127.0.0.1)	3,701.2 KiB	3,648.1 KiB
	/public/image-beach2.jpg (127.0.0.1)	3,464.1 KiB	3,414.3 KiB
▲ Preload	Largest Contentful Paint image		0.89 s ^
Preload	the image used by the LCP element in order to improve your LCP time	ne. <u>Learn more</u> .	
		Show 3rd-party	resources (1)
	URL	Po	tential Savings
	/public/image-beach1.jpg (127.0.0.1)		890 ms
Remove	unused JavaScript		0.28 s ^
Remove	unused JavaScript to reduce bytes consumed by network activity. Le	<u>earn more</u> .	
	If you are not server-side rendering, <u>split your JavaScript bundles</u> a third-party library such as <u>loadable-components</u> .	with `React.lazy()`. Otherwise, cod	de-split using
		Show 3rd-party	resources (0)
URL		Transfer Size	Potential Savings
js/ver	ndors~main.chunk.js (localhost)	869.5 KiB	356.2 KiB
Performance	 More information about the performance of your application. The score. Formous network payloads — Total size was 12,085 KiB 	se numbers don't <u>directly affect</u> the	
Large ne	etwork payloads cost users real money and are highly correlated with	ı long load times. <u>Learn more</u> .	
		✓ Show 3rd-party	resources (3)
URL			Transfer Size
/public/i	image-beach1.jpg (127.0.0.1)		4,029.9 KiB
/public/i	image-beach3.jpg (127.0.0.1)		3,701.4 KiB
/public/i	image-beach2.jpg (127.0.0.1)		3,464.2 KiB
js/ver	ndors~main.chunk.js (localhost)		869.5 KiB
js/ma	nin.chunk.js (localhost)		10.6 KiB
js/bur	ndle.js (localhost)		7.2 KiB

	URL	Transfer Size
	/script.js (localhost)	1.2 KiB
	/home (localhost)	1.2 KiB
	/script.js (localhost)	0.3 KiB
	/sw_udacity.js (localhost)	0.0 KiB
A	Serve static assets with an efficient cache policy — 6 resou	urces found ^
	A long cache lifetime can speed up repeat visits to your page	e. <u>Learn more</u> .
		✓ Show 3rd-party resources (3)
	URL	Cache TTL Transfer Size
	/public/image-beach1.jpg (127.0.0.1)	None 4,030 KiB
	/public/image-beach3.jpg (127.0.0.1)	None 3,701 KiB
	/public/image-beach2.jpg (127.0.0.1)	None 3,464 KiB
	js/vendors~main.chunk.js (localhost)	None 870 KiB
	js/main.chunk.js (localhost)	None 11 KiB
	js/bundle.js (localhost)	None 7 KiB
	Set an explicit width and height on image elements to reduce	e layout shifts and improve CLS. <u>Learn more</u> Show 3rd-party resources (3)
	URL	Failing Elements
	/public/image-beach1.jpg (127.0.0.1)	img.photo
	/public/image-beach3.jpg (127.0.0.1)	img.photo
	/public/image-beach2.jpg (127.0.0.1)	img.photo

Avoid chaining critical requests — 3 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 more.

Maximum critical path latency: 210 ms

Initial Navigation

/home (localhost)

- ...js/bundle.js (localhost) 10 ms, 7.23 KiB
- ...js/vendors~main.chunk.js (localhost) 190 ms, 869.54 KiB
- ...js/main.chunk.js (localhost) 20 ms, 10.56 KiB

Keep request counts low and transfer sizes small — 10 requests • 12,085 KiB

To set budgets for the quantity and size of page resources, add a budget.json file. Learn more.

Resource Type	Requests	Transfer Size
Total	10	12,085.4 KiB
Image	3	11,195.4 KiB
Script	4	888.5 KiB
Document	1	1.2 KiB
Other	2	0.3 KiB
Stylesheet	0	0.0 KiB
Media	0	0.0 KiB
Font	0	0.0 KiB
Third-party	3	11,195.4 KiB

Largest Contentful Paint element — 1 element found

This is the largest contentful element painted within the viewport. Learn More

Element



img.photo

Avoid large layout shifts — 5 elements found

These DOM elements contribute most to the CLS of the page.

Element	CLS Contribution
div.container	0.586
div	
	0.101
nav.navbar.bg-dark.container	
	0.061
footer.text-center.text-xs.text-white.p-3.absolute.bottom-0.w-full.bg-dark	
	0.041
p	0.031
Avoid long main-thread tasks — 2 long tasks found	^
Lists the longest tasks on the main thread, useful for identifying worst contributors to input delay. Learn m	
	d-party resources (0)
URL Start Timjs/vendors~main.chunk.js (localhost) 1,104 m	
/home (localhost) 167 m	
ssed audits (26)	^
Eliminate render-blocking resources	^
Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring JS/styles. Learn more.	ng all non-critical

Learn more.

1		
Defer offscreen images		^
Consider lazy-loading offscreen and hidden images after all critical resources have finished load interactive. <u>Learn more</u> .	ding to lower time	to
Minify CSS — Potential savings of 138 KiB		/
Minifying CSS files can reduce network payload sizes. <u>Learn more</u> .		
If your build system minifies CSS files automatically, ensure that you are deploying application. You can check this with the React Developer Tools extension. Learn mo	•	ld of your
	Show 3rd-party r	esources (0)
URL	Transfer Size	Potentia Saving
<pre>/*! * Bootstrap v4.6.0 (https://getbootstrap.com/) * Copyright 2011-2021 The Bootstrap Authors *</pre>	167.2 KiB	138.4 KiB
Minify JavaScript		
Minifying JavaScript files can reduce payload sizes and script parse time. <u>Learn more</u> .		
If your build system minifies JS files automatically, ensure that you are deploying the application. You can check this with the React Developer Tools extension. Learn mo	-	of your
Remove unused CSS — Potential savings of 167 KiB		/
Remove dead rules from stylesheets and defer the loading of CSS not used for above-the-fold unnecessary bytes consumed by network activity. <u>Learn more</u> .	content to reduce	
	Show 3rd-party r	esources (0)
URL	Transfer Size	Potentia Saving
<pre>/*! * Bootstrap v4.6.0 (https://getbootstrap.com/) * Copyright 2011-2021 The Bootstrap Authors *</pre>	167.2 KiB	166.7 KiB
Efficiently encode images		/
Optimized images load faster and consume less cellular data. <u>Learn more</u> .		
Serve images in next-gen formats		/
Image formats like JPEG 2000, JPEG XR, and WebP often provide better compression than PN faster downloads and less data consumption. <u>Learn more</u> .	NG or JPEG, which	n means
Enable text compression		/
Text-based resources should be served with compression (gzip, deflate or brotli) to minimize to more.	tal network bytes.	<u>Learn</u>
Preconnect to required origins		
Consider adding `preconnect` or `dns-prefetch` resource hints to establish early connections to	important third-pa	irty origins.

Total DOM Elements

Initial server response time was short — Root document took 0 ms Keep the server response time for the main document short because all other requests depend on it. Learn more. Show 3rd-party resources (0) **URL** Time Spent /home (localhost) 0 ms Avoid multiple page redirects Redirects introduce additional delays before the page can be loaded. Learn more. If you are using React Router, minimize usage of the `<Redirect>` component for route navigations. Preload key requests Consider using `<link rel=preload>` to prioritize fetching resources that are currently requested later in page load. Learn more. Use HTTP/2 HTTP/2 offers many benefits over HTTP/1.1, including binary headers and multiplexing. Learn more. 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 Remove duplicate modules in JavaScript bundles Remove large, duplicate JavaScript modules from bundles to reduce unnecessary bytes consumed by network activity. 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. For your bundled JavaScript, adopt a modern script deployment strategy using module/nomodule feature detection to reduce the amount of code shipped to modern browsers, while retaining support for legacy browsers. Learn More Avoids an excessive DOM size - 42 elements A large DOM will increase memory usage, cause longer style calculations, and produce costly layout reflows. Learn more. Consider using a "windowing" library like `react-window` to minimize the number of DOM nodes created if you are rendering many repeated elements on the page. Learn more. Also, minimize unnecessary re-renders using <u>`shouldComponentUpdate`, 'PureComponent`,</u> or <u>'React.memo'</u> and <u>skip effects</u> only until certain dependencies have changed if you are using the 'Effect' hook to improve runtime performance. Value Statistic Element

42

Statistic	Element		Value
Maximum DOM Depth	a.Link		5
Maximum Child Elements	nav.navbar.bg-darl	k.container	7
User Timing marks and measures			^
Consider instrumenting your app with experiences. <u>Learn more</u> .	h the User Timing API to measure your app's rea	al-world performance durir	ng key user
Use the React DevTools your components. Learn	Profiler, which makes use of the Profiler API, to more.	measure the rendering pe	rformance of
JavaScript execution time — 0.2 s			^
Consider reducing the time spent pa with this. <u>Learn more</u> .	rsing, compiling, and executing JS. You may find	d delivering smaller JS pay	rloads helps
		Show 3rd-part	ry resources (0)
URL	Total CPU Time	Script Evaluation	Script Parse
Unattributable	571 ms	5 ms	0 ms
/home (localhost)	241 ms	1 ms	0 ms
js/vendors~main.chunk.js (localhe	ost) 125 ms	5 ms	65 ms
js/main.chunk.js (localhost)	104 ms	102 ms	2 ms
Minimizes main-thread work — 1.0	S		^
Consider reducing the time spent pa with this. <u>Learn more</u>	rsing, compiling and executing JS. You may find	delivering smaller JS pay	loads helps
Category			Time Spent
Other			617 ms
Rendering			219 ms
Corint Evaluation			
Script Evaluation			116 ms

Category	Time Spent
Garbage Collection	12 ms
Style & Layout	9 ms
Parse HTML & CSS	5 ms
All text remains visible during webfont loads	^
Leverage the font-display CSS feature to ensure text is user-visible while webfonts are loading. <u>Learn more</u>	<u>e</u> .
Minimize third-party usage	^
Third-party code can significantly impact load performance. Limit the number of redundant third-party provious load third-party code after your page has primarily finished loading. <u>Learn more</u> .	ders and try to
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	d. <u>Learn more</u> .
Uses passive listeners to improve scrolling performance	^
Consider marking your touch and wheel event listeners as `passive` to improve your page's scroll performa	ance. <u>Learn more</u> .
Avoids document.write()	^
For users on slow connections, external scripts dynamically injected via `document.write()` can delay page seconds. <u>Learn more</u> .	load by tens of
Avoid non-composited animations	^
Animations which are not composited can be janky and increase CLS. <u>Learn more</u>	



Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

Contrast — These are opportunities to improve the legibility of your content.

▲ Background and foreground colors do not have a sufficient contrast ratio.

Low-contrast text is difficult or impossible for many users to read. <u>Learn more</u>.

Failing Elements

I			
Failing Elements	a.Link		
	a.Link		

Navigation — These are opportunities to improve keyboard navigation in your application.

▲ Heading elements are not 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. <u>Learn more</u>.

Failing	Elements
---------	----------



Additional items to manually check (10) — These items address areas which an automated testing tool cannot cover. Learn one in our guide on conducting an accessibility review.

The page has a logical tab order	^
Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. <u>Learn more</u> .	
Interactive controls are keyboard focusable	^
Custom interactive controls are keyboard focusable and display a focus indicator. <u>Learn more</u> .	
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. <u>Learn more</u> .	
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. <u>Learn more</u> .	
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. <u>Learn more</u> .	
Custom controls have associated labels	^
Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. <u>Learn more</u> .	
Custom controls have ARIA roles	^
Custom interactive controls have appropriate ARIA roles. <u>Learn more</u> .	
Visual order on the page follows DOM order	^
DOM order matches the visual order, improving navigation for assistive technology. <u>Learn more</u> .	
Offscreen content is hidden from assistive technology	^
Offscreen content is hidden with display: none or aria-hidden=true. <u>Learn more</u> .	
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. <u>Learn more</u>.

Pa	ssed audits (8)	^
	[aria-hidden="true"] is not present on the document <body></body>	^
	Assistive technologies, like screen readers, work inconsistently when `aria-hidden="true"` is set on the document ` <body>`. <u>Learn more</u>.</body>	
	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.	
	Document has a <title> element</td><td>^</td></tr><tr><td></td><td>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. <u>Learn more</u>.</td><td>е</td></tr><tr><td></td><td><html> element has a [lang] attribute</td><td>^</td></tr><tr><td></td><td>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.</td><td></td></tr><tr><td></td><td><html> element has a valid value for its [lang] attribute</td><td>^</td></tr><tr><td></td><td>Specifying a valid <u>BCP 47 language</u> helps screen readers announce text properly. <u>Learn more</u>.</td><td></td></tr><tr><td></td><td>Image elements have [alt] attributes</td><td>^</td></tr><tr><td></td><td>Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. <u>Learn more</u>.</td><td></td></tr><tr><td></td><td>Links have a discernible name</td><td>^</td></tr><tr><td></td><td>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. <u>Learn more</u>.</td><td></td></tr><tr><td></td><td>[user-scalable="no"] is not used in the <meta name="viewport"> element and the [maximum-scale] attribute is not less than 5.</td><td>^</td></tr><tr><td></td><td>Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. <u>Learn more</u>.</td><td>ì</td></tr><tr><td>No</td><td>ot applicable (34)</td><td>^</td></tr><tr><td></td><td>[accesskey] values are unique</td><td>^</td></tr><tr><td></td><td>Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. Learn more</td><td></td></tr><tr><td></td><td>[aria-*] attributes match their roles</td><td>^</td></tr></tbody></table></title>	

Each ARIA 'role' supports a specific subset of 'aria-*' attributes. Mismatching these invalidates the 'aria-*' attributes. Learn 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 more. [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 more. 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. ARIA meter 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 more. ARIA progressbar 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 more. [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. 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. [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. [role] values are valid ARIA roles must have valid values in order to perform their intended accessibility functions. Learn more 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 ARIA tooltip 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 more.

ARIA treeitem elements have accessible names

users who rely on screen readers. Learn more. [aria-*] attributes have valid values Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. Learn more. [aria-*] attributes are valid and not misspelled Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. Learn more. 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 more. <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 more. 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 more. [id] attributes on active, focusable elements are unique All focusable elements must have a unique 'id' to ensure that they're visible to assistive technologies. Learn more. ARIA IDs are unique The value of an ARIA ID must be unique to prevent other instances from being overlooked by assistive technologies. Learn more. 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 more. <frame> or <iframe> elements have a title Screen reader users rely on frame titles to describe the contents of frames. Learn more. <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 more. Form elements have associated labels Labels ensure that form controls are announced properly by assistive technologies, like screen readers. Learn more. Lists contain only 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.

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for

List items () are contained within or parent elements

Screen readers require list items (`') to be contained within a parent `' or `' to be announced properly. <u>Learn</u> more.

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. <u>Learn more</u>.

<object> elements have [alt] text

Screen readers cannot translate non-text content. Adding all text to `<object>` elements helps screen readers convey meaning to users. <u>Learn more</u>.

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. <u>Learn more</u>.

Cells in a element that use the [headers] attribute refer to table cells within the same table.

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

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. <u>Learn more</u>.

[lang] attributes have a valid value

Specifying a valid <u>BCP 47 language</u> on elements helps ensure that text is pronounced correctly by a screen reader. <u>Learn more</u>.

<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.



Best Practices

General

▲ Uses deprecated APIs — 1 warning found

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

Show 3rd-party resources (0)

Deprecation / Warning Source SharedArrayBuffer will require cross-origin isolation as of M91, around May 2021. See vendors~main.ch https://developer.chrome.com/blog/enabling-shared-array-buffer/ for more details. unk.js:87136 Browser errors were 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 Show 3rd-party resources (0) Source Description script.js:10 TypeError: btnAdd.addEventListener is not a function at http://localhost:3000/script.js:10:12 Passed audits (15) **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. Links to cross-origin destinations are safe Add `rel="noopener"` or `rel="noreferrer"` to any external links to improve performance and prevent security vulnerabilities. Learn more. 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. 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. Avoids front-end JavaScript libraries with known security vulnerabilities Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more. Allows users to paste into password fields Preventing password pasting undermines good security policy. Learn more. Displays images with correct aspect ratio Image display dimensions should match natural aspect ratio. Learn more. Serves images with appropriate resolution

Page has the HTML doctype Specifying a doctype prevents the browser from switching to quirks-mode. Learn more. 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. Avoids unload event listeners The 'unload' event does not fire reliably and listening for it can prevent browser optimizations like the Back-Forward Cache. Consider using the 'pagehide' or 'visibilitychange' events instead. Learn more Avoids Application Cache Application Cache is deprecated. Learn more. **Detected JavaScript libraries** All front-end JavaScript libraries detected on the page. Learn more. Name Version React 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. Show 3rd-party resources (0) URL Map URL ...js/vendors~main.chunk.js.map (localhost) ...js/vendors~main.chunk.js (localhost) Error: Timed out fetching resource. ...js/main.chunk.js (localhost) ...js/main.chunk.js.map (localhost) Error: Timed out fetching resource. ...js/bundle.js (localhost) ...js/bundle.js.map (localhost) Error: Timed out fetching resource. 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.

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

Not applicable (1)

Fonts with font-display: optional are preloaded

Preload 'optional' fonts so first-time visitors may use them. Learn more



These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. <u>Learn more</u>.

Additional items to manually check (1) — Run these additional validators on your site to check additional SEO best practices. Structured data is valid Run the Structured Data Testing Tool and the Structured Data Linter to validate structured data. Learn more Passed audits (10) Has a <meta name="viewport"> tag with width or initial-scale Add a `<meta name="viewport">` tag to optimize your app for mobile screens. Learn more. 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 Document has a meta description Meta descriptions may be included in search results to concisely summarize page content. Learn more. Page has successful HTTP status code Pages with unsuccessful HTTP status codes may not be indexed properly. Learn more. Links have descriptive text Descriptive link text helps search engines understand your content. Learn more. 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. <u>Learn More</u>

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. 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. 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. Document avoids plugins Search engines can't index plugin content, and many devices restrict plugins or don't support them. Learn more. Not applicable (4) 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. Document has a valid rel=canonical Canonical links suggest which URL to show in search results. Learn more. Document uses legible font sizes 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. Tap targets are sized appropriately Interactive elements like buttons and links should be large enough (48x48px), and have enough space around them, to be easy enough to tap without overlapping onto other elements. Learn more.



Progressive Web App

These checks validate the aspects of a Progressive Web App. Learn more.

Installable

Web app manifest and service worker meet the installability requirements

Service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. With proper service worker and manifest implementations, browsers can proactively prompt users to add your app to their homescreen, which can lead to higher engagement. <u>Learn more</u>.

PWA Optimized Registers a service worker that controls page and start url The service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. Learn more. Redirects HTTP traffic to HTTPS If you've already set up HTTPS, make sure that you redirect all HTTP traffic to HTTPS in order to enable secure web features for all your users. Learn more. Configured for a custom splash screen A themed splash screen ensures a high-quality experience when users launch your app from their homescreens. Learn more. Sets a theme color for the address bar. The browser address bar can be themed to match your site. Learn more Content is sized correctly for the viewport If the width of your app's content doesn't match the width of the viewport, your app might not be optimized for mobile screens. Learn more. Has a <meta name="viewport"> tag with width or initial-scale Add a `<meta name="viewport">` tag to optimize your app for mobile screens. Learn more. Provides a valid apple-touch-icon For ideal appearance on iOS when users add a progressive web app to the home screen, define an 'apple-touch-icon'. It must point to a non-transparent 192px (or 180px) square PNG. Learn More. Manifest doesn't have a maskable icon A maskable icon ensures that the image fills the entire shape without being letterboxed when installing the app on a device. Learn more. Additional items to manually check (3) — These checks are required by the baseline PWA Checklist but are not automatically checked by Lighthouse. They do not affect your score but it's important that you verify them manually. Site works cross-browser To reach the most number of users, sites should work across every major browser. Learn more. Page transitions don't feel like they block on the network

Transitions should feel snappy as you tap around, even on a slow network. This experience is key to a user's perception of performance. <u>Learn more</u>.

Each page has a URL

Ensure individual pages are deep linkable via URL and that URLs are unique for the purpose of shareability on social media. <u>Learn more</u>.

Runtime Settings

URL http://localhost:3000/home

Fetch Time Apr 24, 2021, 11:07 PM GMT+2

Device Emulated Desktop

Network throttling 40 ms TCP RTT, 10,240 Kbps throughput (Simulated)

CPU throttling 1x slowdown (Simulated)

Channel devtools

User agent (host) Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like

Gecko) Chrome/90.0.4430.85 Safari/537.36

User agent (network) Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML,

like Gecko) Chrome/90.0.4420.0 Safari/537.36 Chrome-Lighthouse

CPU/Memory Power 1101

Axe version 4.1.2

Generated by Lighthouse 7.2.0 | File an issue