



Accessibility



Best Practices



SEO

▲ 0-49

■ 50-89

● 90-100



## 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

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



Low-contrast text is difficult or impossible for many users to read. [Learn more](#).

#### Failing Elements



button.btn.btn-primary.d-block.d-md-inline.float-md-end



a



div.card.bg-warning.h-100



a

	div.card.bg-warning.h-100
	a
	div.card.border-warning.h-100
	a
	main.m-0.main-wrapper

These are opportunities to improve the legibility of your content.

#### ADDITIONAL ITEMS TO MANUALLY CHECK (10)

Hide

- The page has a logical tab order



Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. [Learn more.](#)

- Interactive controls are keyboard focusable



Custom interactive controls are keyboard focusable and display a focus indicator. [Learn more.](#)

- 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 more.](#)

- 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 more.](#)

- 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 more.](#)

- Custom controls have associated labels ^

Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. [Learn more.](#)

- Custom controls have ARIA roles ^

Custom interactive controls have appropriate ARIA roles. [Learn more.](#)

- Visual order on the page follows DOM order ^

DOM order matches the visual order, improving navigation for assistive technology. [Learn more.](#)

- Offscreen content is hidden from assistive technology ^

Offscreen content is hidden with display: none or aria-hidden=true. [Learn more.](#)

- 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.](#)

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

PASSED AUDITS (22)

Hide

- [aria-\*] attributes match their roles ^

Each ARIA `role` supports a specific subset of `aria-\*` attributes. Mismatching these invalidates the `aria-\*` attributes. [Learn more.](#)

- [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 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-\*] 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.](#)

● 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.](#)

● 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.](#)

● Form elements have associated labels ^

Labels ensure that form controls are announced properly by assistive technologies, like screen readers. [Learn more.](#)

● `[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.](#)

● 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 ^

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.](#)

● `[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.](#)

● `<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.](#)

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

Specifying a valid [BCP 47 language](#) helps screen readers announce text properly. [Learn more.](#)

● 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 more.](#)

● 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.](#)

● List items (`<li>`) are contained within `<ul>` or `<ol>` parent elements ^

Screen readers require list items (`<li>`) to be contained within a parent `<ul>` or `<ol>` to be announced properly. [Learn more](#).

- 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](#).

NOT APPLICABLE (21) 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](#).

- `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 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 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 ^

---

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](#).

---

`<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](#).

---

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](#).

---

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](#).

---

`<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](#).

---

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](#).

---

`<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](#).

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](#).

`<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](#).

`[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 more](#).

`<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

### 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 more](#)

Description	Directive	Severity
No CSP found in enforcement mode		High

GENERAL

○ Detected JavaScript libraries ^

All front-end JavaScript libraries detected on the page. [Learn more.](#)

Name	Version
Bootstrap	5.2.0
jQuery	3.6.0
Vue	

▲ Missing source maps for large first-party JavaScript ^

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.](#)

URL	Map URL
/DependencyHandler.axd?s=L2Fzc2V0c...&t=Javascript&cdv=167... (www.farte.no)	
Large JavaScript file is missing a source map	

PASSED AUDITS (12)

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.](#)

● 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. <a href="#">Learn more.</a>	
 Avoids front-end JavaScript libraries with known security vulnerabilities <span style="float: right;">^</span>	
Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. <a href="#">Learn more.</a>	
 Allows users to paste into password fields <span style="float: right;">^</span>	
Preventing password pasting undermines good security policy. <a href="#">Learn more.</a>	
 Displays images with correct aspect ratio <span style="float: right;">^</span>	
Image display dimensions should match natural aspect ratio. <a href="#">Learn more.</a>	
 Serves images with appropriate resolution <span style="float: right;">^</span>	
Image natural dimensions should be proportional to the display size and the pixel ratio to maximize image clarity. <a href="#">Learn more.</a>	
 Page has the HTML doctype <span style="float: right;">^</span>	
Specifying a doctype prevents the browser from switching to quirks-mode. <a href="#">Learn more.</a>	
 Properly defines charset <span style="float: right;">^</span>	
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. <a href="#">Learn more.</a>	
 Avoids deprecated APIs <span style="float: right;">^</span>	
Deprecated APIs will eventually be removed from the browser. <a href="#">Learn more.</a>	
 No browser errors logged to the console <span style="float: right;">^</span>	
Errors logged to the console indicate unresolved problems. They can come from network request failures and other browser concerns. <a href="#">Learn more</a>	
 No issues in the <b>Issues</b> panel in Chrome Devtools <span style="float: right;">^</span>	
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.	

## NOT APPLICABLE (1)

Hide

- Fonts with `font-display: optional` are preloaded ^

Preload `optional` fonts so first-time visitors may use them. [Learn more](#)



## 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](#).

## 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](#).

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

## PASSED AUDITS (10)

Hide

- 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](#).

- 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](#).

<input checked="" type="radio"/> Links have descriptive text <span style="float: right;">^</span>
Descriptive link text helps search engines understand your content. <a href="#">Learn more.</a>
<input checked="" type="radio"/> Links are crawlable <span style="float: right;">^</span>
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. <a href="#">Learn More</a>
<input checked="" type="radio"/> Page isn't blocked from indexing <span style="float: right;">^</span>
Search engines are unable to include your pages in search results if they don't have permission to crawl them. <a href="#">Learn more.</a>
<input checked="" type="radio"/> Image elements have <code>[alt]</code> attributes <span style="float: right;">^</span>
Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. <a href="#">Learn more.</a>
<input checked="" type="radio"/> Document has a valid <code>hreflang</code> <span style="float: right;">^</span>
hreflang links tell search engines what version of a page they should list in search results for a given language or region. <a href="#">Learn more.</a>
<input checked="" type="radio"/> Document avoids plugins <span style="float: right;">^</span>
Search engines can't index plugin content, and many devices restrict plugins or don't support them. <a href="#">Learn more.</a>

NOT APPLICABLE (4)

Hide

<input type="radio"/> robots.txt is valid <span style="float: right;">^</span>
If your robots.txt file is malformed, crawlers may not be able to understand how you want your website to be crawled or indexed. <a href="#">Learn more.</a>
<input type="radio"/> Document has a valid <code>rel=canonical</code> <span style="float: right;">^</span>
Canonical links suggest which URL to show in search results. <a href="#">Learn more.</a>
<input type="radio"/> Document uses legible font sizes <span style="float: right;">^</span>
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. <a href="#">Learn more.</a>

○ 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.](#)

 Captured at Jan 26, 2023, 4:36 PM GMT+1

 Emulated Desktop with Lighthouse 9.6.8

 Single page load

 Initial page load

 Custom throttling

 Using Chromium 109.0.0.0 with devtools

Generated by **Lighthouse** 9.6.8 | [File an issue](#)