

# TEXT CHECKLIST

WHAT TO LOOK FOR	WHAT TO DO
<b>Font size</b>	<p>Work with design to find a default font size that is readable.</p> <p>A default of 16px is a good baseline minimum.</p>
<b>Heading level</b>	<p>Nest headings by their rank (or level).</p> <p>The most important heading has the rank 1 (&lt;h1&gt;), the least important heading rank 6 (&lt;h6&gt;).</p> <p>Headings with an equal or higher rank start a new section, headings with a lower rank start new subsections that are part of the higher ranked section.</p>
<b>Colour contrast</b>	<p>Check colour contrasts on various online tools such as:</p> <p><a href="https://webaim.org/resources/contrastchecker/">https://webaim.org/resources/contrastchecker/</a></p> <p>Work with design on what colours to use if colours fail tests.</p>
<b>Font legibility</b>	<p>Best practice is to use the most readable fonts. Unfortunately, this is more easily said than done.</p> <p>It is generally best to use standard fonts that are available on the end users device. Keep in mind that documents with only one, or only a few font faces are usually easier to read.</p> <p>Using too many font faces can create a confusing visual layout, which is bad for all users, but may be especially difficult for users with reading disorders, learning disabilities, or attention deficit disorders.</p>
<b>Font scalability</b>	<p>It is best to use relative units (such as percent or ems) to specify font sizes rather than absolute units (such as pixels or points).</p> <p>This provides more flexibility in modifying the visual presentation using CSS.</p> <p>For accessibility, because modern browsers adequately resize text regardless of how the size has been defined, it is not vital that text sizes be defined in relative sizes.</p>
<b>Capitalisation</b>	<p>Typing sentences or phrases IN ALL CAPITALS is rarely a good idea.</p> <p>It may make sense under some circumstances, but only rarely. Lengthy segments of capitalised content are more difficult to read. They also may give the impression that the author is shouting.</p> <p>If the author really does want to convey a shout, an exclamation point and/or use of &lt;em&gt; or &lt;strong&gt; may be better.</p>

# LINKS CHECKLIST

WHAT TO LOOK FOR	WHAT TO DO
<b>Title attributes being applied</b>	<p>Title attributes should not be used:</p> <p><a href="https://developer.paciellogroup.com/blog/2012/01/html5-accessibility-chops-title-attribute-use-and-abuse/">https://developer.paciellogroup.com/blog/2012/01/html5-accessibility-chops-title-attribute-use-and-abuse/</a></p>
<b>Ensure links are recognisable</b>	<p>Ensure links don't just have a different colour than the surrounding text, also add visual indicators (underline etc.).</p>
<b>Links that contain no text</b>	<p>For example: An image wrapped in an anchor tag.</p> <p>You should make sure the link has text. For this example you could add a hidden span with a description.</p>
<b>Missing focus states</b>	<p>Check all links for a focus state. Add a focus state where applicable.</p>
<b>Missing skip link</b>	<p>Create a skip link. This should be the first tab-able link on the page and should take the user to the start of the main content.</p>
<b>Navigation/ Breadcrumbs/ Pagination</b>	<p>Follow best practices when creating components such as navigation, pagination and breadcrumbs.</p>

# SIGNAL11Y

●●● SIGNAL | ACCESSIBILITY



## KEYBOARD CHECKLIST

WHAT TO LOOK FOR	WHAT TO DO
<b>TabIndex</b>	TabIndex should flow correctly down the page, the best approach is to try and not change it in any way and use the defaults.
<b>Make sure user can get down the page correctly using the keyboard</b>	Ensure modals, tabs and other custom components can be interacted with properly and check the user can get past those components.

## USEFUL TOOLS

WHAT TO LOOK FOR	WHAT TO DO
<b>Axe</b>	Axe is an extension created by deque, web accessibility specialists. Axe scans your pages and highlights a wide range of accessibility issues. Some examples are invalid semantics, poor colour contrast ratios, and elements missing important attributes such as alt tags on non-decorative images.
<b>WebAIM</b>	Wave is a tool by WebAIM, web accessibility specialists. Wave is a similar tool to Axe in that it will scan your page and highlight detected issues, as well as offering suggestions on how to fix these problems, and a link to the relevant part of the WCAG specification. WebAIM also offer a great collection of tools and resources you can find online.
<b>Eslint Jsx Accessibility Plugin</b>	Eslint-plugin-jsx-a11y is a great plugin which extends eslint to scan your code for a range of defined accessibility issues, when pairing this with a modern editor such as vscode, this can give you real time feedback as you're coding, which can help to prevent inaccessible components.
<b>Lighthouse</b>	Lighthouse is a tool by the Google Chrome team which audits website performance, accessibility, and various other metrics. You can find this under the audits tab built into Chrome. We often use this as a supplementary tool after we have checked the page with WebAIM and or Axe.

## IMAGES CHECKLIST

WHAT TO LOOK FOR	WHAT TO DO
<b>Alt attributes</b>	Ensure all images have an alt attribute, if the image is decorative then give it an empty alt attribute.  Make sure the description of the image is useful.  Don't begin alternative text with "photo of..." or "picture of...". Assistive technologies already indicate the role of the element as an "image" or "graphic".
<b>Legibility</b>	Ensure that images do not contain important text, as this is not accessible when presented through images.

## USEFUL RESOURCES

WHAT TO LOOK FOR	WHAT TO DO
<b>WebAIM</b>	Beyond the tools offered, WebAIM curate a very useful list of accessibility information which you can find on their page:  <a href="https://webaim.org/resources/">https://webaim.org/resources/</a>
<b>Inclusive Components</b>	Heydon Pickering is an accessibility expert who maintains the Inclusive Components blog, sharing tips and practical examples of how to create accessible front end components.
<b>Smashing Magazine</b>	Smashing Magazine offer great accessibility insights, thanks to their Accessibility Editor, the aforementioned Heydon Pickering.
<a href="https://www.w3.org/TR/wai-aria-practices/">https://www.w3.org/TR/wai-aria-practices/</a>	W3 publish a wealth of technical information as how to adhere to WCAG guidelines.