Auditing Cheat Sheet

Auditing your web-based systems is a vital part of ensuring good practice in terms of building good user experience. In the same way that a developer tests their website’s functionality or security, the level of accessibility should be tested and compared against the [Web Content Accessibility Guidelines (WCAG 2.1 AA)](https://www.w3.org/TR/WCAG21/) to ensure compliance with UK legislation and to provide users with the best experience. This cheat sheet will provide an overview of how to approach testing a web page in terms of accessibility and how to improve websites by thinking about them from a different perspective. For more detail on what to look for, please see the [Supplier Accessibility Guidance Document](https://lexdis.boatwif.co.uk/wp-content/uploads/2019/07/Supplier-Accessibility-Guidance.docx) on LexDis.

## Requirements

**Platforms** - Windows 10

**Browsers** - Internet Explorer/Edge, Chrome/Firefox/Safari

**Screen Readers** - [NVDA](https://www.nvaccess.org/)/[JAWS](https://www.freedomscientific.com/products/software/jaws/)

**Magnifiers** - Windows Magnifier/Zoomtext/OS X Magnifier

**Automated Tools** - [WebAim WAVE](https://wave.webaim.org/)

## 1. Structural Auditing

Structural auditing involves looking at the markup of the page and the structure of its elements to ensure that the page is friendly to assistive tools and users. A comprehensive solution is to manually check through the HTML of the site before using an automated tool to reinforce your results.

### Manual

Using inspect element, analyse the markup (e.g. HTML). Common things to look out for include:

* All markup tags are closed/complete
* Semantic markup are used appropriately
* Labels and attributes are associated with elements
* Page language is declared
* Markup is not used for styling (e.g. tables not used for structure, HTML bold elements not used)
* ARIA attributes and alt-tags present on relevant elements.

### Automated

After a manual audit, use an automated tool to confirm your findings and pick up on things you may have missed. Tools such as [WAVE](https://wave.webaim.org/) will analyse a page to produce a report on the structural errors it has found. These tools will only pick up structural errors however, which only covers off 20-30% of WCAG guidance.

## 2. Content Auditing

After identifying structural issues, the content should be analysed to ensure that it is perceivable, operable, understandable and robust.

### Keyboard & Screen Reader

Reload the page and reach the bottom of the page using a keyboard-only (Tab, Shift+Tab, Enter and the arrow keys). Meanwhile, have a screen reader on and ensure that all elements (e.g. text, images) are being read by the screen reader with clear and useful descriptions. Look out for things such as:

* The very first element should be a button to bypass the navigation called a “skip to content” button
* Images have clear and descriptive alt tags
* Text (including titles and text on top of other elements) is read aloud
* The web page and its elements are navigable with a keyboard-only
* There should be no images of text
* Links are clear and descriptive in context
* Keyboard focus is highlighted
* Contrast of elements is adequate and perceivable
* Videos have captions, and transcripts (if applicable).

### No Styles

Some users remove the CSS from a web page so they can interact directly with elements and information on a page. Internet Explorer has a ‘No Style’ option to do remove the CSS. Using the keyboard and screen reader again, go through all elements and content to ensure there are no artifacts or broken functionality when CSS has been removed.

### Additional Checks

A number of additional checks should be performed on the page to ensure that it is widely compatible with assistive tools and different devices. It could be useful to do these tests on another device/OS (e.g. a mobile), with things to look out for including:

* Content displaying properly when text spacing or line height is increased
* No horizontal scrolling if the window’s width is decreased to 320px
* The page is readable and functional when zoomed to 200%.

## Next Steps

If you’ve found accessibility issues after auditing, ensure to log them for developers to make fixes and/or to produce a report for the information of stakeholders to aid their decision making. Ensure that pages are re-audited when a substantial change has been made.