

Accessibility Design Standards for Browser-based systems

V2.1

3 April 2018

Table of Contents

Background.....	6
Purpose	6
Policy	6
Principle 1 - Perceivable non-text information.....	8
1.1 Text alternatives for images Content authors Developers	8
1.2 CSS background images Developers	8
1.3 Alternatives for complex images Content authors Designers	9
1.4 Form field labels Developers	9
Principle 2 - Perceivable video and audio information.....	9
2.1 Captions and text transcript Content authors Designers	10
2.2 Audio that plays automatically Designers	10
Principle 3 - Perceivable structure and presentation	10
3.1 Headings Designers Developers	10
3.2 Lists Designers Developers	11
3.3 Data tables Designers Developers	11
3.4 Data table captions and summarises Content authors Designers Developers	12
3.5 Layout tables Developers	12
3.6 Landmarks Designers Developers	13
3.7 Page title Content authors Developers	13
3.8 Page language Content authors Developers	13

Principle 4 - Perceivable and distinguishable information	14
4.1 Use of colour Designers	14
4.2 Link identification Designers Developers	14
4.3 Colour contrast Designers	15
4.5 Colour customisation Developers	15
Principle 5 - Operable functionality via multiple input methods	15
5.1 Keyboard access Developers	16
5.2 Focus visible Designers Developers	16
5.3 Tabbing order Developers	16
5.4 Reading order Developers	17
5.5 Skip links Designers Developers	17
Principle 6 - Operable timeouts	17
6.1 Timeouts Designers	17
Principle 7 – Controllable animations	18
7.1 Animated content Designers	18
Principle 8 - Operable Navigation	18
8.1 Consistent navigation Content authors Designers	18
8.2 Login link Designers	18
8.3 Size of actionable items Designers	19
8.4 Navigation methods Designers	19
8.5 Step indicator Designers Developers	19

8.6 Accessibility link	Designers	19
Principle 9 - Understandable content		20
9.1 Text formatting	Designers	20
9.2 Text resizing	Designers Developers	20
9.3 Images of text	Designers Developers	20
9.4 Language Content	authors	21
9.5 Content sections	Content authors	21
9.6 Instructions	Content authors	21
Principle 10 - Understandable and predictable behaviour		22
10.1 Coding of user interface components	Designers Developers	22
10.2 Dynamic updates	Developers	22
10.3 Links	Content authors Developers	23
10.4 Unexpected changes	Designers Developers	23
10.5 Frames	Content authors Developers	24
Principle 11 – Understandable forms and instructions		24
11.1 Form labels	Content authors Designers Developers	24
11.2 Label placement	Designers	24
11.3 Error messages	Content authors Designers Developers	25
11.4 Review screen	Content authors Designers	25
Principle 12 - Compatibility with assistive technologies		26
12.1 Hidden content	Developers	26

12.2 Unique id attribute Developers	26
--	----

Background

This document has been produced to ensure that new systems and enhancements to systems developed within the Barclays Group are made accessible to all customers and staff and support Barclays' responsibilities under the Equality Act (EA 2010) in the UK and corresponding laws based on international guidelines globally.

These standards are based on the requirements from the Web Content Accessibility Guidelines (WCAG) 2.0 and best accessibility practice.

Purpose

This document is to be used within the lifecycle of a project or any enhancement to an existing system.

Designs should comply with all elements of the standards within this document.

There is a legal requirement to retain evidence of the activity to meet accessibility requirements therefore completed assessments must be stored to document compliance for each project/enhancement as evidence of the approach taken. If at any time you are unable to incorporate one or more of the standards you must provide evidence as to what approach was taken and the reasoning behind the decision not to comply.

This should also include any staff training systems or customer demonstration systems.

Policy

Whilst the reasonableness of adjustments will be determined by the Courts as case law is developed in the UK, the EHRC Code of Practice in the UK recommends that all reasonable steps are taken to ensure that systems are accessible to people with disabilities. In the case of service providers, the needs of disabled people at large should be anticipated and good practice measures introduced to meet those needs. Accessibility of websites is highlighted specifically in the Code. When designing systems and enhancements, Barclays will also have regard to its obligations towards vulnerable customers. As highlighted in [Barclays Customers in Vulnerable Circumstances Global Policy](#), physical and cognitive disabilities are amongst the circumstances which may result in a customer being especially susceptible to detriment.

In order for Barclays to meet its legal obligations in the countries in which it operates, it is a mandatory requirement for all designs to comply with these Standards. These requirements are regulatory and have been endorsed by Group Diversity and Inclusion. From the publication of this document it is the Barclays' policy that designs over time adopt adjustments to comply with all of the standards. These standards will be kept under review to ensure that they remain fit for purpose, in the light of legal, regulatory and/or technological developments.

Any areas of non-compliance must be identified and agreed by adopting an appropriate policy. It is recommended that non-compliance handling be based on appropriate Non-Financial Risk



assessment and acceptance of risk at appropriate levels of authority. This approach is embedded into the breach process followed in the Barclays Group.

Principle 1 - Perceivable non-text information

This principle covers the requirement to provide text alternatives for non-text content such as images, charts and graphs. There is also a requirement for labels to be associated with interface objects e.g. form fields.

1.1 Text alternatives for images **Content authors** **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	All HTML images have the <i>alt</i> attribute in their source code.	1.1.1
Content authors	A meaningful and accurate textual description is provided for all informative images (i.e. images that convey important information).	1.1.1
Content authors	A textual description of their function or destination content is provided for all functional images (i.e. images that act as links or buttons).	1.1.1
Content authors	Textual descriptions do <u>not</u> need to communicate the type of element (e.g. "image of", "button for" do <u>not</u> need to be included).	1.1.1
Content authors Developers	The null alt attribute (<i>alt=""</i>) is included in the source code of decorative images (i.e. images that have a solely decorative purpose), or CSS images are used for decorative images.	1.1.1

Resources:

- <http://webaim.org/techniques/alttext/>
- <https://www.w3.org/WAI/tutorials/images/>

1.2 CSS background images **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	CSS background images are only used for decorative images.	1.1.1

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/F3.html>

1.3 Alternatives for complex images **Content authors** **Designers**

Who is responsible	Requirement	WCAG 2.0
Content authors Designers	Charts, graphs, diagrams and other complex images are described in text on the same page or in a page linked from the page containing the image.	1.1.1
Designers	When charts or graphs are used to display tabular information (i.e. interrelated data), the same data is shown in a correctly marked-up table.	1.1.1

Resources:

- <https://www.w3.org/WAI/tutorials/images/complex/>

1.4 Form field labels **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	Form fields that have a visible label are associated with their label in the HTML code (via the <i>id</i> and <i>for</i> attributes).	1.3.1
Developers	Form fields with no visible label are described via the <i>aria-label</i> attribute or are associated with a label positioned off-screen. (Note: Placeholder text does not replace the need for labels).	1.3.1
Developers	Radio buttons and checkboxes are associated with their label via the <i>id</i> and <i>for</i> attributes, are grouped together with <i><fieldset></i> and the question preceding each group is coded with <i><legend></i> .	1.3.1

Resources:

- <http://webaim.org/techniques/forms/controls>
- <https://www.w3.org/WAI/tutorials/forms/labels/>
- <https://www.w3.org/WAI/tutorials/forms/grouping/>

Principle 2 - Perceivable video and audio information

This principle covers the requirement to provide captions and transcripts for audio and video content. The desired approach is to use captioning for video and audio content with transcripts being additional to captions rather than replacing them.

2.1 Captions and text transcript **Content authors** **Designers**

Who is responsible	Requirement	WCAG 2.0
Content authors Designers	A text transcript is available (on the same page or on a page linked from the page containing the media) for any audio-only (e.g. podcasts), video-only (e.g. videos with no audio track) or multimedia content (e.g. videos with video and audio content).	1.2.1 , 1.2.3
Content authors Designers	Captions (closed or open) are available for any multimedia content whose audio track conveys important information.	1.2.2

Resources:

- <https://www.w3.org/TR/UNDERSTANDING-WCAG20/media-equiv-captions.html>
- <https://www.w3.org/TR/UNDERSTANDING-WCAG20/media-equiv-audio-desc.html>

2.2 Audio that plays automatically **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	Audio content does not start playing automatically, or stops after 3 seconds.	1.4.2
Designers	Alternatively, a keyboard accessible and clearly-labelled Pause button is available at the very beginning of the page.	1.4.2

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/G170.html>

Principle 3 - Perceivable structure and presentation

This principle deals with the need to ensure that the structure of a page - including page titles, headings, lists and tables - is clearly communicated to assistive technologies.

3.1 Headings **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	Headings are coded with heading mark-up (<code><h1></code> - <code><h6></code>).	1.3.1

Who is responsible	Requirement	WCAG 2.0
Designers Developers	Headings are assigned the correct heading level (i.e. the main page heading is coded with <code><h1></code> , its subheadings are coded with <code><h2></code> , subheadings of any <code><h2></code> heading are coded with <code><h3></code> , etc.).	1.3.1
Developers	Text that does not represent a heading is not coded with heading mark-up.	1.3.1

Resources:

- <https://www.w3.org/WAI/tutorials/page-structure/headings/>
- <http://webaim.org/techniques/semanticstructure/>

3.2 Lists **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	Lists are coded with the most appropriate HTML list element (<code></code> for unordered lists, <code></code> for ordered lists and <code><dl></code> for definition lists).	1.3.1
Designers Developers	List mark-up is used to accurately describe the structure of lists (e.g. do not split items of a same list into two separate <code></code> elements).	1.3.1
Developers	Content that does not represent a list is not coded with list mark-up.	1.3.1

Resources:

- <https://www.w3.org/WAI/tutorials/page-structure/content/#lists>

3.3 Data tables **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	In data tables, cells containing row/column headers are coded with <code><th></code> ; cells containing data are coded with <code><td></code> .	1.3.1

Who is responsible	Requirement	WCAG 2.0
Designers Developers	Complex data tables (i.e. tables containing several rows and/or columns of headers, irregular and multi-level headers) are avoided when possible. When complex data tables must be used, each data cell is associated with all its headers via the <i>scope</i> , <i>id</i> and <i>headers</i> attributes.	1.3.1

Resources:

- <https://www.w3.org/WAI/tutorials/tables/>
- <http://webaim.org/techniques/tables/data>

3.4 Data table captions and summarises **Content authors** **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Content authors Designers Developers	A title is provided for all data tables via the <i><caption></i> element.	1.3.1
Content authors Designers Developers	For complex tables, information on the content and structure of the table is also provided immediately before the table, either as visible or hidden text.	1.3.1

Resources:

- <https://www.w3.org/WAI/tutorials/tables/caption-summary/>

3.5 Layout tables **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	The use of layout tables (i.e. tables solely used to position items on the screen) is avoided when possible; CSS are used to position the content instead.	1.3.1
Developers	When layout tables cannot be avoided, their source code contains the ARIA role <i>presentation</i> and does not include the <i><th></i> and <i><caption></i> elements, and the <i>headers</i> , <i>axis</i> , <i>scope</i> and <i>summary</i> attributes.	1.3.1

Resources:

- <http://webaim.org/techniques/tables/>
- https://www.w3.org/TR/wai-aria-practices-1.1/#presentation_role

3.6 Landmarks **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	The purpose of each section of content on a page is identified via HTML 5 semantic elements (e.g. <code><nav></code> , <code><aside></code> , <code><footer></code>) and/or ARIA landmark roles (e.g. <i>banner</i> , <i>main</i> , <i>content info</i>).	1.3.1
Designers Developers	When a same HTML 5 element and/or ARIA landmark is used multiple times on a page, the content of each is described via <i>aria-label</i> (e.g. "main navigation", "secondary navigation").	1.3.1

Resources:

- <https://www.w3.org/WAI/tutorials/page-structure/regions/>
- <https://www.w3.org/WAI/tutorials/page-structure/labels/>

3.7 Page title **Content authors** **Developers**

Who is responsible	Requirement	WCAG 2.0
Content authors Developers	Each page has a unique and descriptive title, provided via the <code><title></code> element.	2.4.2
Content authors Developers	In single page applications, the title is updated every time a new view is loaded.	2.4.2

Resources:

- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G88>

3.8 Page language **Content authors** **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	The <code><html></code> element of every page with English content includes <code>lang="en"</code> .	3.1.1

Who is responsible	Requirement	WCAG 2.0
Content authors Developers	If a page contains any block of content in a language other than English, the language is specified in the source code of this content via the <i>lang</i> attribute.	3.1.2

Resources:

- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/H57>
- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/H58>

Principle 4 - Perceivable and distinguishable information

All users – including those with visual and colour deficiencies - must be able to perceive and distinguish information on a website. This principle covers the requirement not to use colour alone to differentiate between elements, to use colours with a strong contrast ratio and to allow users to change the colours.

4.1 Use of colour **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	Colour alone is not used to convey information (e.g. to identify form fields with invalid entries or the current step in a step indicator) or to distinguish between areas of a map or graph; additional visual cues are used to provide the same information (e.g. icons, shapes, patterns, text).	1.4.1

Resources:

- <https://www.deque.com/blog/inclusive-design-tips-presenting-information-multiple-ways/>

4.2 Link identification **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers	All links within blocks of text are underlined, and no other content on the page is underlined.	1.4.1

Who is responsible	Requirement	WCAG 2.0
Designers Developers	If links cannot be underlined and are identified via colour alone, the contrast ratio between their colour and the colour of the surrounding text is at least 3, and their visual appearance changes (e.g. they become underlined) when they receive the mouse and keyboard focus.	1.4.1

Resources:

- http://webaim.org/techniques/hypertext/link_text#underlining

4.3 Colour contrast **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	Text that is 18 points or larger has a contrast ratio with the background colour of at least 3; text that is smaller than 18 points has a contrast ratio with the background colour of at least 4.5.	1.4.3

Resources:

- <https://accessibility.blog.gov.uk/2016/06/17/colour-contrast-why-does-it-matter/>
- <http://juicystudio.com/services/luminositycontrastratio.php>

4.5 Colour customisation **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	When users modify text and background colours in the accessibility settings of their computer or browser, the colours on the web page change accordingly.	1.4.8

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/G156>
- <https://mcmw.abilitynet.org.uk/changing-your-colours/>

Principle 5 - Operable functionality via multiple input methods

The main consideration for this principle is to ensure that content can be accessed without the need for a pointing device. This should be done in a logical way- generally from the top left of the screen and ending up in the bottom right. Shortcuts to allow users to skip over large

amounts of navigation or to complete common tasks should also be implemented. Keyboard focus and text cursors should be clearly visible.

5.1 Keyboard access **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	All actionable components can be reached and activated using the keyboard alone.	2.1.1
Developers	No item traps the keyboard focus; upon reaching any item on the page, it is possible to move to the item that precedes or follows it using the keyboard.	2.1.2

Resources:

- <https://webaim.org/techniques/keyboard/>
- <https://webaim.org/techniques/javascript/eventhandlers>

5.2 Focus visible **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	When actionable items receive the keyboard focus, their visual appearance changes (e.g. a box appears around them, links become underlined, the colour of the text or of the background changes).	2.4.7

Resources:

- <https://www.deque.com/blog/give-site-focus-tips-designing-usable-focus-indicators/>

5.3 Tabbing order **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	Actionable items receive the keyboard focus in a logical order.	2.4.3

Resources:

- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G59>

5.4 Reading order **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	All content on the page is announced by screen readers in a logical order (in most cases this should be top-to-bottom and left-to-right).	1.3.2

Resources:

- <http://webaim.org/techniques/tables/#linearization>

5.5 Skip links **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	The page contains one or more links that allow users to jump to the different sections of content (e.g. the secondary navigation, the main content area).	2.4.1
Designers Developers	They are the first links on the page and (if hidden) become visible when they receive the keyboard focus.	2.4.1

Resources:

- <http://webaim.org/techniques/skipnav/>

Principle 6 - Operable timeouts

Timeouts may prevent users with a disability from completing actions on a website. This principle deals with this issue by requesting developers to inform users in advance of a timeout. The user should also be given the opportunity to extend this time, or where practical, to switch off the timeouts.

6.1 Timeouts **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	If it is necessary to implement timeouts (e.g. for security reasons), users are informed when a timeout is about to occur and are given the option to extend it.	2.2.1

Principle 7 – Controllable animations

This principle covers the requirement to ensure that animated content is not overly used and is controllable.

7.1 Animated content **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	Any content that scrolls, moves or blinks (e.g. carousels, news ticker) automatically stops after 5 seconds. Alternatively, users can easily pause or hide the animated content.	2.2.2

Resources:

- <https://www.w3.org/WAI/tutorials/carousels/animations/>

Principle 8 - Operable Navigation

This principle deals with the need for a consistent and useful navigation system. To ensure that users are able to navigate and find content, it is important that our websites have clear and consistent design and navigation. Users should be able to confidently predict where interface elements can be found. Also, elements such as buttons, form controls and links should be large enough to be selected without the possibility of selecting adjacent controls.

8.1 Consistent navigation **Content authors** **Designers**

Who is responsible	Requirement	WCAG 2.0
Content authors Designers	Navigational items that are repeated on multiple pages of a website (e.g. Search field, social media links, related links panels) are positioned and labelled consistently across the website.	3.2.3 , 3.2.4

8.2 Login link **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	For pages containing a login link, this link is one of the first links on the page.	-

8.3 Size of actionable items **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	The selectable area of actionable items is large enough for all users to select with ease.	-
Designers	There is some empty space in between actionable items (e.g. in between links on a navigation menu), to reduce the chances of users selecting the wrong item.	-

8.4 Navigation methods **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	In addition to navigation menus, the website contains a Search functionality and/or a site map (for large websites) and/or an A-Z index of pages (for small websites).	2.4.5

8.5 Step indicator **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	On pages that constitute a multi-step process, a step indicator indicates how many steps are in the process and which step the user is at (using text or icons associated with a textual description).	-

Resources:

- <https://www.w3.org/WAI/tutorials/forms/multi-page/>

8.6 Accessibility link **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	The page contains a link to the Accessibility page of the website (e.g. http://www.barclays.co.uk/accessibility/ for Barclays.co.uk).	-

Principle 9 - Understandable content

This principle covers the requirement to write content that is easy to read and understand for all users. Ensuring that you use a clear typeface (i.e. non serif font) will make content easier to read by users. All Barclays Brand fonts comply with this principle. In addition to using clear fonts, it is important that the content itself is clear and concise and is written in an easy to understand way. Using simple language for information, instructions, prompts and outputs will allow all users to understand the content. As a financial services organisation, it's important that we have clear, jargon free content.

9.1 Text formatting **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	A sans-serif font is used for all text on the page; Barclays guidelines state that Expert Sans Light, Expert Sans Regular, Verdana and Arial should be used on Barclays websites.	-
Designers	Capitalisation and italic text are used sparingly.	-
Designers	Text is left aligned.	-

Resources:

- <http://webaim.org/techniques/fonts/#readability>

9.2 Text resizing **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	When the whole page, or its text content only, is enlarged, all information and functionality remains available.	1.4.4

9.3 Images of text **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	With the exception of text in logos, graphs and diagrams, the page does not contain any image of text; text is HTML text.	1.4.5

Resources:

- <https://www.w3.org/WAI/tutorials/images/textual/>

9.4 Language **Content authors**

Who is responsible	Requirement	WCAG 2.0
Content authors	The language used on the page is as simple and clear as possible, taking into account the nature of the content and the website audience.	3.1.5
Content authors	Jargon is avoided as much as possible.	3.1.3
Content authors	Acronyms and abbreviations are used sparingly, and their full extension is displayed next to them the first time they are used on the page.	3.1.4

Resources:

- <http://webaim.org/techniques/writing/>

9.5 Content sections **Content authors**

Who is responsible	Requirement	WCAG 2.0
Content authors	The content on the page is divided into small blocks, each preceded by a descriptive heading.	2.4.6

Resources:

- <http://4syllables.com.au/articles/accessibility-writers-labels/>

9.6 Instructions **Content authors**

Who is responsible	Requirement	WCAG 2.0
Content authors	The instructions on the page do not rely upon shape, size, visual location or sound. Examples of inaccessible instructions are "Click the square icon to continue", "Instructions are in the right-hand column" and "A beeping sound indicates you may continue".	1.3.3

Resources:

- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G96>

Principle 10 - Understandable and predictable behaviour

This principle deals with how dynamic components are conveyed to assistive technologies and how they behave. It should be clear to all users what the functionality of these components is, how to interact with them and what the result of this interaction is. No component should behave in an unexpected way.

10.1 Coding of user interface components **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	Links, buttons and form fields are implemented as native HTML controls (e.g. <code><a></code> , <code><button></code> , <code><input></code> , <code><select></code>), rather than as custom controls (e.g. using <code><div></code> , <code></code> elements and scripting).	4.1.2
Designers Developers	The HTML code of custom controls (e.g. sliders, tabs, accordions) includes WAI-ARIA attributes to describe their role and status (e.g. <code>tab</code> , <code>slider</code> , <code>selected</code> , <code>expanded</code> , <code>collapsed</code>).	4.1.2

Resources:

- <https://www.w3.org/TR/wai-aria-practices-1.1/>

10.2 Dynamic updates **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	WAI-ARIA live regions are used to inform screen reader users of dynamic changes to the page content (e.g. error messages that appear while users fill in a form, an alert message that is displayed when a session is about to expire, some content that is changed/removed/added as a result of a user's action).	3.3.1

Resources:

- https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Live_Regions

10.3 Links **Content authors** **Developers**

Who is responsible	Requirement	WCAG 2.0
Content authors	All links accurately describe the content that they load; the page does not contain generic links such as "click here" and "read more".	2.4.4
Content authors Developers	If the use of generic links cannot be avoided, information on their destination content is included in the HTML code, either as hidden text or via the <i>aria-label</i> or <i>aria-labelledby</i> attributes (e.g. "about credit card" is added to the end of a "read more" link as hidden text).	2.4.4
Content authors	Links that open a new tab/window or load content other than a web page (e.g. a PDF document) clearly indicates this in text or via an icon associated with a textual description (e.g. "Accessibility Academy (new window)", "Prospectus 2011 (PDF 2.3Mb)").	2.4.4

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/ARIA7>
- <https://www.w3.org/TR/WCAG20-TECHS/ARIA8>
- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G201>

10.4 Unexpected changes **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	No unexpected changes of context occur while users are interacting with the page. Unexpected changes of context include the loading of a new page, the opening of a new window or the sudden change of location of the focus on the page. Such changes only occur when initiated by the user (e.g. upon activation of a link or a button).	3.2.1 , 3.2.2

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/F36>
- <https://www.w3.org/TR/WCAG20-TECHS/F37>

10.5 Frames **Content authors** **Developers**

Who is responsible	Requirement	WCAG 2.0
Content authors Developers	Any <code><frame></code> or <code><iframe></code> element used in the source code of the page includes a <code>title</code> attribute that accurately describes the frame content (e.g. "Travel form").	4.1.2

Resources:

- <http://webaim.org/techniques/frames/>

Principle 11 – Understandable forms and instructions

Principle 11 covers the need to ensure that forms, instructions and error messages are clear and accurate. This will help users to avoid and correct mistakes.

11.1 Form labels **Content authors** **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Content authors	All labels in a form clearly and precisely describe the data required from users.	3.3.2
Content authors Designers	The labels of mandatory fields include an asterisk or the text "Required"; alternatively, the labels of non-mandatory fields include the text "(Optional)" when most fields are mandatory.	3.3.2
Content authors Designers Developers	Any additional useful information around the data to enter (e.g. the expected format, example data) is included in the label or are displayed next to the field and associated with it using <code>aria-labelledby</code> or <code>aria-describedby</code> .	3.3.2

Resources:

- <https://www.w3.org/WAI/tutorials/forms/instructions/>

11.2 Label placement **Designers**

Who is responsible	Requirement	WCAG 2.0
Designers	Labels are displayed to the left or above text fields and drop-downs, and to the right of checkboxes and radio buttons.	1.3.1

Who is responsible	Requirement	WCAG 2.0
Designers	There is not a lot of empty space between labels and fields.	1.3.1

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/G162>

11.3 Error messages **Content authors** **Designers** **Developers**

Who is responsible	Requirement	WCAG 2.0
Designers Developers	When forms are submitted with errors, error messages are listed on the screen and the keyboard focus is set to the list of error messages.	3.3.1
Developers	If error messages are displayed inline, <i>aria-describedby</i> is used to associate them with the fields in error. When the error messages are added to forms as the users fill them in, ARIA live regions are used to ensure they are immediately announced by screen readers.	3.3.1
Content authors	The error messages clearly and accurately explain the errors that have occurred and provide suggestions on how to fix them. When possible, suggestions for corrections are offered.	3.3.3

Resources:

- <https://www.w3.org/WAI/tutorials/forms/notifications/>

11.4 Review screen **Content authors** **Designers**

Who is responsible	Requirement	WCAG 2.0
Content authors Designers	On multi-page forms, a review screen showing all entered data and offering users the option to modify them (if possible) is provided before the data is submitted.	3.3.4

Resources:

- <https://www.w3.org/TR/WCAG20-TECHS/G98>

Principle 12 - Compatibility with assistive technologies

The final principle focuses on the need to ensure that websites are compatible with assistive technologies. Meeting all the requirements listed on this document will help you to ensure that assistive technologies can access and correctly convey the content on your websites. In addition to this, good coding and design principles - including using the latest versions of HTML, CSS and WAI-ARIA correctly - should be followed.

12.1 Hidden content **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	With the exception of decorative images and other visual content that would be of no value to visually impaired users (e.g. legend) , all content that is visible on the page is announced by screen readers.	-
Developers	With the exception of hidden instructions/information deliberately added to the HTML code for the benefit of screen reader users, content that is not visible on the page and should not be accessible to any users (including visually impaired users) is not announced by screen readers. E.g. content on the collapsed sections of an accordion, a message that has been dismissed by users, form fields that should only appear when a specific option in a drop-down list is selected.	-

12.2 Unique id attribute **Developers**

Who is responsible	Requirement	WCAG 2.0
Developers	All <i>id</i> attributes in the HTML code of a page have a unique value.	4.1.1

Resources:

- <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/H93>