W3C

List of Checkpoints for Web Content Accessibility Guidelines 1.0

This version:
http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/checkpoint-list
(plain text, postscript, pdf)
This document is an appendix to:
http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505
Latest version of Web Content Accessibility Guidelines 1.0:
http://www.w3.org/TR/WAI-WEBCONTENT
Editors:
Wendy Chisholm, Trace R & D Center, University of Wisconsin -- Madison
Gregg Vanderheiden, Trace R & D Center, University of Wisconsin -- Madison
Ian Jacobs, W3C

Abstract

This document is an appendix to the W3C "Web Content Accessibility Guidelines 1.0". It provides a list of all checkpoints from the Web Content Accessibility Guidelines 1.0, organized by concept, as a checklist for Web content developers. Please refer to the Guidelines document for introductory information, information about related documents, a glossary of terms, and more.

This list may be used to review a page or site for accessibility. For each checkpoint, indicate whether the checkpoint has been satisfied, has not been satisfied, or is not applicable.

A tabular version of the list of checkpoints is also available (e.g., for printing).

This document has been produced as part of the Web Accessibility Initiative. The goal of the WAI Web Content Guidelines Working Group is discussed in the Working Group charter.

Status of this document

This document is an appendix to a document that has been reviewed by W3C Members and other interested parties and has been endorsed by the Director as a W3C Recommendation. This is a stable document and may be used as reference material or cited as a normative reference from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and universality of the Web.

A list of current W3C Recommendations and other technical documents can be found at http://www.w3.org/TR.

This document has been produced as part of the Web Accessibility Initiative. The goal of the Web Content Guidelines Working Group is discussed in the Working Group charter.

Please send comments about this document to wai-wcag-editor@w3.org.

Priorities

Each checkpoint has a priority level assigned by the Working Group based on the checkpoint's impact on accessibility.

[Priority 1]
: A Web content developer **must** satisfy this checkpoint. Otherwise, one or more groups will find it impossible to access information in the document. Satisfying this checkpoint is a basic requirement for some groups to be able to use Web documents.

[Priority 2]
: A Web content developer **should** satisfy this checkpoint. Otherwise, one or more groups will find it difficult to access information in the document. Satisfying this checkpoint will remove significant barriers to accessing Web documents.

[Priority 3]
: A Web content developer **may** address this checkpoint. Otherwise, one or more groups will find it somewhat difficult to access information in the document. Satisfying this checkpoint will improve access to Web documents.

Some checkpoints specify a priority level that may change under certain (indicated) conditions.

Priority 1 checkpoints

In General (Priority 1)

- 1.1 Provide a text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). *This includes*: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video.
- 2.1 Ensure that all information conveyed with color is also available without

color, for example from context or markup.

- 4.1 Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions).
- 6.1 Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document.
- 6.2 Ensure that equivalents for dynamic content are updated when the dynamic content changes.
- 7.1 Until user agents allow users to control flickering, avoid causing the screen to flicker.
- 14.1 Use the clearest and simplest language appropriate for a site's content.

And if you use images and image maps (Priority 1)

- 1.2 Provide redundant text links for each active region of a server-side image map.
- 9.1 Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.

And if you use tables (Priority 1)

- 5.1 For data tables, identify row and column headers.
- 5.2 For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.

And if you use frames (Priority 1)

- 12.1 Title each frame to facilitate frame identification and navigation.

And if you use applets and scripts (Priority 1)

- 6.3 Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.

And if you use multimedia (Priority 1)

- 1.3 Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation.
- 1.4 For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.

And if all else fails (Priority 1)

- 11.4 If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.

Priority 2 checkpoints

In General (Priority 2)

- 2.2 Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].
- 3.1 When an appropriate markup language exists, use markup rather than images to convey information.
- 3.2 Create documents that validate to published formal grammars.
- 3.3 Use style sheets to control layout and presentation.
- 3.4 Use relative rather than absolute units in markup language attribute values and style sheet property values.
- 3.5 Use header elements to convey document structure and use them according to specification.
- 3.6 Mark up lists and list items properly.
- 3.7 Mark up quotations. Do not use quotation markup for formatting effects such as indentation.
- 6.5 Ensure that dynamic content is accessible or provide an alternative presentation or page.
- 7.2 Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off).
- 7.4 Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.
- 7.5 Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects.
- 10.1 Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user.
- 11.1 Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported.
- 11.2 Avoid deprecated features of W3C technologies.
- 12.3 Divide large blocks of information into more manageable groups where natural and appropriate.
- 13.1 Clearly identify the target of each link.
- 13.2 Provide metadata to add semantic information to pages and sites.

- 13.3 Provide information about the general layout of a site (e.g., a site map or table of contents).
- 13.4 Use navigation mechanisms in a consistent manner.

And if you use tables (Priority 2)

- 5.3 Do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized version).
- 5.4 If a table is used for layout, do not use any structural markup for the purpose of visual formatting.

And if you use frames (Priority 2)

- 12.2 Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone.

And if you use forms (Priority 2)

- 10.2 Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned.
- 12.4 Associate labels explicitly with their controls.

And if you use applets and scripts (Priority 2)

- 6.4 For scripts and applets, ensure that event handlers are input device-independent.
- 7.3 Until user agents allow users to freeze moving content, avoid movement in pages.
- 8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]
- 9.2 Ensure that any element that has its own interface can be operated in a device-independent manner.
- 9.3 For scripts, specify logical event handlers rather than device-dependent event handlers.

Priority 3 checkpoints

In General (Priority 3)

- 4.2 Specify the expansion of each abbreviation or acronym in a document where it first occurs.
- 4.3 Identify the primary natural language of a document.
- 9.4 Create a logical tab order through links, form controls, and objects.
- 9.5 Provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls.
- 10.5 Until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links.
- 11.3 Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.)
- 13.5 Provide navigation bars to highlight and give access to the navigation mechanism.
- 13.6 Group related links, identify the group (for user agents), and, until user agents do so, provide a way to bypass the group.
- 13.7 If search functions are provided, enable different types of searches for different skill levels and preferences.
- 13.8 Place distinguishing information at the beginning of headings, paragraphs, lists, etc.
- 13.9 Provide information about document collections (i.e., documents comprising multiple pages.).
- 13.10 Provide a means to skip over multi-line ASCII art.
- 14.2 Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page.
- 14.3 Create a style of presentation that is consistent across pages.

And if you use images and image maps (Priority 3)

- 1.5 Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map.

And if you use tables (Priority 3)

- 5.5 Provide summaries for tables.
- 5.6 Provide abbreviations for header labels.
- 10.3 Until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for *all* tables that lay out text in parallel, word-wrapped columns.

And if you use forms (Priority 3)

- 10.4 Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas.