

# Quick Reference: Testing Web Content for Accessibility (draft 4/10/07 by Paul Bohman)

1. **Validity/Modern Coding Practices/Structure**
  - a. Validate the code (e.g. XHTML 1.0 strict).
  - b. Specify the character encoding (e.g. UTF-8).
  - c. Use CSS (not tables) for layout.
  - d. Use headings, lists, etc. to organize & structure content.
2. **Enlargement**
  - a. Enlarge the font in Firefox using CTRL + (the plus key). At a minimum, the page needs to be readable when you enlarge it twice using CTRL +. It's ok if some areas don't align quite right or if they overlap slightly, but not if the page is very difficult to read or navigate when enlarged twice.
3. **Styles**
  - a. Test the page with styles turned off. You can do this in Opera by going into "user mode" or in Firefox if you have the Web Developer Toolbar (<https://addons.mozilla.org/en-US/firefox/addon/60>) installed by using the CTRL Shift S keyboard shortcut. You want to make sure that:
    - i. The reading order is still logical when styles are turned off.
    - ii. The content still makes sense. Make sure you haven't used colors to convey meaning, for example.
4. **No mouse**
  - a. Navigate the site using only the keyboard (using the tab key, arrow keys, enter key, page up, page down, etc.)
  - b. Provide a functional "skip navigation" link (see [www.webaim.org/techniques/skipnav/](http://www.webaim.org/techniques/skipnav/)).
  - c. Make sure the link order makes sense, so that when you tab around you're not jumping all over the place.
5. **Browser Compatibility**
  - a. Make sure your browser works in all of the modern browsers, including (at a minimum), Firefox 2.0, Opera 9.0, and Internet Explorer (test it in both 7.0 and 6.0). If you use a Mac, also test in Safari. The design doesn't have to look exactly the same, but it must be functional. (Ideally it would look the same, but sometimes this is difficult to achieve.)
6. **Automated tools**
  - a. Use **at least two** automated tools to check the accessibility of EVERY page individually. Popular tools include WAVE (<http://dev.wave.webaim.org/>), Cynthia Says ([www.cynthiasays.com](http://www.cynthiasays.com)), Hermish ([www.hermish.com](http://www.hermish.com)), WebXact (<http://webxact.com/>) and others. You may want to download the AIS accessibility toolbar (see the info at [www.webaim.org/articles/ais/index.php](http://www.webaim.org/articles/ais/index.php)) which includes links to these and other very useful tools.
7. **Human check**
  - a. **Colors and Contrast**
    - i. Make sure the text is readable against the background, including links, navigation tabs, main content, etc. One way to test this is to use the tools in the AIS Accessibility Toolbar. Another way is to go to the page, hit Prnt Scrn (print screen) on your keyboard, then go to Word (or a graphics program), hit Ctrl V to paste in the screen, then print the screen to a black and white printer. If there are areas of low contrast, adjust the colors appropriately.
    - ii. Simulate color blindness and ensure that your pages are still readable. To do this, use the AIS Accessibility Toolbar (Tools > Simulations > Color blindness) or the the Vischeck web-based tool ([www.vischeck.com/vischeck/](http://www.vischeck.com/vischeck/)).
  - b. **Section 508 compliance.** Check EACH page of your site against the Section 508 checklist (see [www.webaim.org/standards/508/checklist.php](http://www.webaim.org/standards/508/checklist.php)).
  - c. **Images** (see [www.webaim.org/techniques/images/](http://www.webaim.org/techniques/images/)):
    - i. Alt text is short but accurate and useful
    - ii. If the image is just decorative (i.e. if blind people don't need to know what the image is), the image is in the background using CSS (not the <img> tag) or else the alt text is null (alt=" ")
    - iii. Images which require long descriptions are described in more detail, either in the context of the page or on a separate page (and you would provide a link to that separate page).
    - iv. Animations do not strobe or flicker
  - d. **Tables** (see [www.webaim.org/techniques/tables/](http://www.webaim.org/techniques/tables/)):
    - i. Simple tables (i.e. tables with up to two headers per data cell) have scope="col" and/or scope="row", as appropriate
    - ii. Complex tables (i.e. tables with three or more headers per data cell) use the headers and id method of associating data cells with their headers.
  - e. **Links** (see [www.webaim.org/techniques/hypertext/](http://www.webaim.org/techniques/hypertext/))
    - i. Make sure your links describe the link destination, even when read alone, out of context. For example, "click here" is bad because you don't know where the link will go.
  - f. **Forms** (see [www.webaim.org/techniques/forms/](http://www.webaim.org/techniques/forms/))
    - i. Make sure all form elements have <label> elements. You can test this by clicking on the label text with your mouse. If clicking on the text causes the keyboard focus to enter the form element, the label works. If not, it needs to be fixed.
    - ii. Groups of radio buttons or check boxes need to be enclosed in a <fieldset> with a <legend>
8. **Disability Types** ([www.webaim.org/articles/userperspective/](http://www.webaim.org/articles/userperspective/))
  - a. **Blindness** (they can't see; they use keyboard for navigation and screen reader to read content)
  - b. **Low Vision** (they can't see small items; they have difficulty with low contrast; they may use screen enlargement software)
  - c. **Color Blindness** (red-green and red-black confusion are common; contrast may be an issue)
  - d. **Motor Disabilities** (they may have difficulty with small links; may need to use keyboard or keyboard-like devices instead of mouse)
  - e. **Auditory Disabilities** (they can't hear audio; transcripts and/or captions necessary)
  - f. **Cognitive Disabilities** (may experience challenges with memory, attention, complex text and/or instructions, metaphorical language, etc.)
  - g. **Seizure Disorders** (strobe or flicker may cause seizure)
9. **Accessibility Principles** (see [www.webaim.org/articles/pour/](http://www.webaim.org/articles/pour/))
  - a. **Perceivable** (available through sight, hearing, touch)
  - b. **Operable** (compatible with keyboard or mouse)
  - c. **Understandable** (user-friendly, easy to comprehend)
  - d. **Robust** (works in several browsers, assistive technologies, mobile devices, old browsers, etc.)