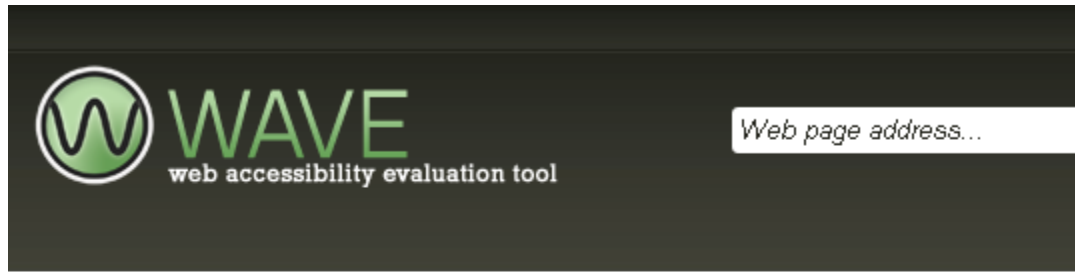# Testing and Creating Accessible Web Pages – a Brief How To

Testing a Web Page for Accessibility

If you want to test a single page, one of the easiest ways is to use the WAVE web accessibility evaluation tool from Webaim.org. Go to http://wave.webaim.org/ and enter the website address of the page you want to have checked.

The output comes back with color-coded icons that show the page's accessibility. See http://wave.webaim.org/icons for more information on the icons.

If you have included a picture with no alt text, WAVE tells you that immediately. Headings and lists are reported. Even if there are no accessibility errors detected by the tool, the fine print instructs you to check the page (ideally with a vision-impaired person using a screen reader) to ensure that it is actually accessible.

Quick Tips from WAI, the Web Accessibility Initiative

The W3C organization developed a list of 10 quick tips for making web pages and websites accessible. For additional information, see http://www.w3.org/WAI/quicktips/ .

1. **Images and animations:** use the **alt** attribute to describe the function of each visual.
2. **Image maps:** Use the **map** and text for hotspots.
3. **Multimedia:** Provide captioning and transcripts of audio, and descriptions of video.
4. **Hypertext links:** Use text that makes sense when read out of context. For example, avoid "click here."
5. **Page organization:** Use headings, lists, and consistent structure. Use CSS for layout and style where possible.
6. **Graphs and charts:** Summarize or use the **longdesc** attribute.
7. **Scripts, applets, and plug-ins:** Provide alternative content in case active features are inaccessible or unsupported.
8. **Frames:** Use the noframes element and meaningful titles.
9. **Tables:** Make line-by-line reading sensible. Summarize.

10. **Check your work:** Validate. Use tools, checklist, and guidelines at
    http://www.w3.org/TR/WCAG/ .

Add Information for Non-Text Elements (e.g. Images, Video, Audio)

If you're adding a picture to a web page in Cascade Server at UC Hastings, Cascade requires that you add **alt** text to describe the picture. The descriptive words should tell briefly what the picture shows.
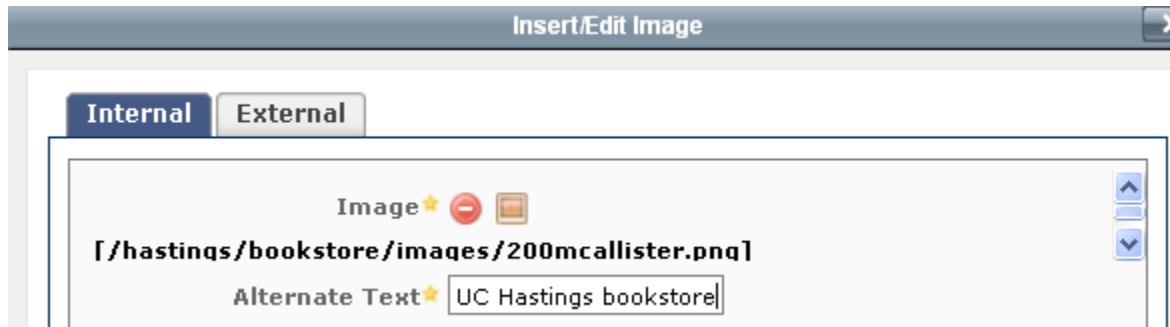

Audio files should have transcripts and video files should have captions. A useful discussion on determining a text equivalent for a graphic that is appropriate to the context where it appears is at http://www.webaim.org/techniques/images/alt_text.php.

Create Links that Make Sense when Read Out of Context

"Click here" is an example of a ineffective link. Especially when used repeatedly on a page, "click here" doesn't convey any meaning and can be confusing. Try to describe (briefly and concisely) where you'll go when you follow the link. Some examples are "Download the course catalog" or "more detailed information about applying."

Page Organization - Use Headings and Lists Consistently

Just as a document should have a title whose style is **Heading 1,** a web page should have a <h1> heading for its title. The title on every Hastings web page is automatically made into a <h1> Heading 1. For example, the Bookstore's textbook page has "Textbook Pre-ordering Information" for its title. Inside Cascade, you see
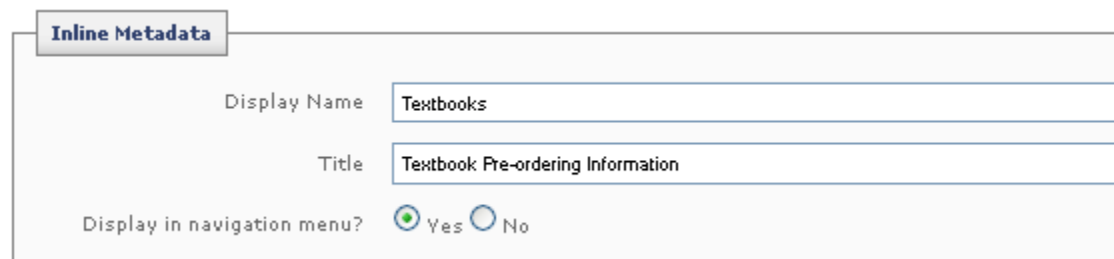
and on the page, the title appears as

Home > Bookstore > Textbooks

Textbook Pre-ordering Information

Pre-order Textbooks - Avoid Long Lines!

Monday, July 26 to Friday, August 13, 2010

The second line on the page (Pre-order Textbooks – Avoid Long Lines!) is a Heading 2 (<h2>) . Use headings just as you would when creating an outline.

WAVE has an "outline" tool what will show the Headings on a web page.
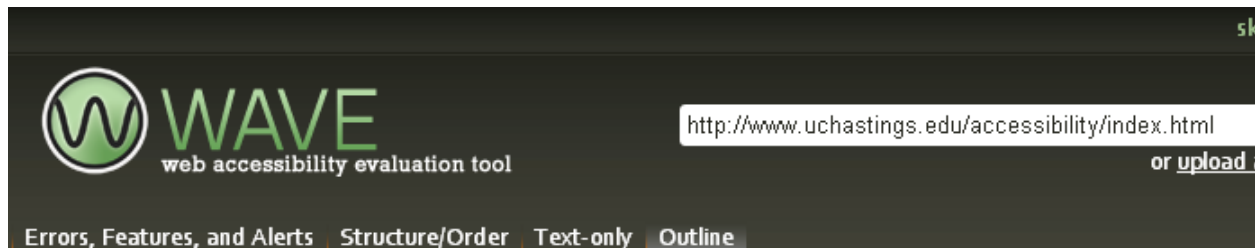
In Cascade Server, to set a Heading 2 or Heading 3, etc. hilight the text and set "Heading 2" in the "Format" drop down box.
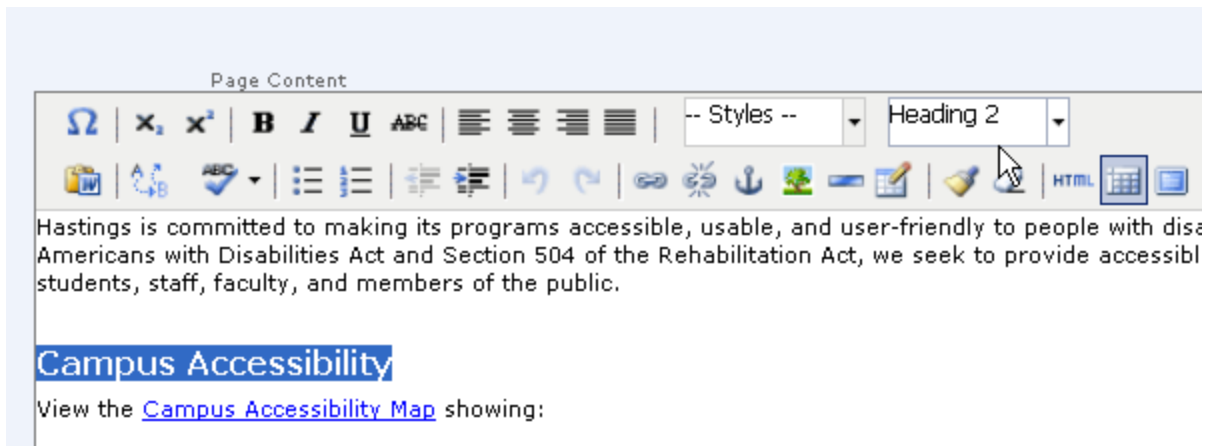
All main headings on the page should be **Heading 2**. If a section is broken into subsections, they would have a **Heading 3** style, etc. The headings show the page's structure, similar to an outline.

One of the common 508 sayings is – "If it looks like a heading, it should be tagged as a heading." In otherwords, don't use just Bold and Font size to make something look like a heading. Use Styles.

Frames

The Hastings website doesn't use frames.

Tables

Using tables on a web page can be problematic for screen readers which process information from left to right, then top to bottom.

For example, in the salary table below, the heading identifiers go with the values below.

average salary	minimum salary	maximum salary
$122,378	$30,000	$165,000

A screen reader would probably read this as "average salary, minimum salary, maximum salary" and then "$122,378, $30,000, $165,000." There is no grouping of "average salary - $122,378" and so forth.

In this case, we don't need a table and could format it as straight text:

Average salary: $122,378

Minimum salary: $30,000

Maximum salary: $165,000

Data tables should have row and column headers identified.

If data tables have two or more logical levels of row or column headers, use markup to associate data cells and header cells. For in the following table, the first row and the first column are headers for the data in the rest of the rows and columns.

Category	#	# reporting salary	average salary	minimum salary	maximum salary
Law Firms (all sizes)	210	147	$135,337	$45,000	$165,000
Self-Employed	8	1	$80,000	$80,000	$80,000
2 to 10 Attorneys	54	19	$72,568	$45,000	$135,000
11 to 25 Attorneys	14	9	$75,556	$55,000	$115,000
26 to 50 Attorneys	8	3	$103,333	$70,000	$160,000

In html, you would use <th> and "scope" to explain each header. This would be

<tr>

<th scope = "col">Category</th>
<th scope = "col"> # </th>
. . .
<th scope = "row">Law Firms (all sizes)</th>

. . .

Cascade Server checks each label to see if it has a "summary" – a brief explanation of the table's contents. If none is defined, you will see a message each time you edit the page and will be asked to provide a summary. For the table above, you might say "table contains salary ranges for various firm sizes."

For additional information about tables and accessibility, consult the WAI at http://www.w3.org/WAI/wcag-curric/chk6-0.htm .

Use Appropriate Font Size and Text Scaling
Most current browsers (Internet Explorer, Firefox, Safari, Chrome) and assistive technologies can enlarge text independent of how the page was coded. However, some older or non-standard browsers can only enlarge text if the page has been coded properly. So font size should be coded so that it will be responsive to the older browser setting. This is best done via definitions within cascading style sheets (CSS), to facilitate uniform implementation across a site.

The default text sizes for Hastings web pages are set for each page by the design and style sheet used. The Center for Accessible Technology (CforAT) recommends that default text sizes be at least .75em (the equivalent of 12pt in most browsers) and layouts should be designed to allow for text-scaling to 200% of the default without having text overlap or disappear.