

Web Accessibility at Berkeley

On April 21, 2006, Berkeley's CIO, Shel Waggener wrote:

As more of our day-to-day communications, work flow, applications, and collaboration environments move to the Web, full accessibility and access to the broadest possible segment of the Berkeley community as possible is critical.

What's here

- Guiding Principles
- Top 10 tips
 1. Skip navigation
 2. Heading elements (aka "headers")
 3. Cascading Style Sheets (CSS)
 4. Images
 5. Color
 6. Unique names for links
 7. "Go" or "Submit" buttons with combo boxes
 8. Labeling and ordering text fields
 9. Keyboarding and tab order
 10. No more text-only alternatives
- Web resources
- Four steps for redesigning an existing site
- Six steps for developing or designing a new site for accessibility
- The Berkeley Web Accessibility Group

Guiding Principles

- If a website is more accessible, it is also more usable.
- Accessibility and design are complementary: one is not sacrificed for the other.
- Improving the accessibility and usability of websites is an on-going process because of constant changes in both the technology and the law.

Top 10 tips: Tips to keep in mind to make a website both more accessible and more usable.

1. Skip navigation (or “skip to main content” or “jump to...”) links

Why. “Skip nav” allows people using screen readers to move from the top of the page to the beginning of the “main content” of the page (whatever is deemed the “main content” or “most important”). Those who cannot use the mouse have a similar need to use their keyboard to skip over the repetitive navigation links.

What. “Skip nav” is a link with a local target just before the beginning of the main content. Keep it visible and at the top of the page.

For more information. See Jim Thatcher’s article on skip navigation, <http://www.jimthatcher.com/skipnav.htm>, and WebAIM’s article, <http://webaim.org/techniques/skipnav/>.

2. Heading elements, <H1> to <H6>

Note. “Heading” and “header” are often used interchangeably.

Why. Heading elements are more than a quick way to format text. Use them to define your page’s organizational structure and simplify page navigation. Effective use of headings also promotes good indexing and searching using Google or other search engines.

What. Since some users skim through a document by navigating its headings, it is important to use them appropriately to convey document structure. Order header elements properly. For example, in HTML, H2 elements should follow H1 elements, H3 elements should follow H2 elements, and so on. Content developers should not “skip” levels (e.g., H1 directly to H3). Use Cascading Style Sheets (CSS) instead of heading elements to change font sizes and/or styles.

For more information. See the Web Content Accessibility Guidelines (WCAG) version 1.0, section 1.2.1, <http://www.w3.org/TR/WCAG10-HTML-TECHS/#grouping> on structural grouping and W3C’s write-up on heading elements and page structure, <http://www.w3.org/TR/REC-html40/struct/global.html#h-7.5.5>.

3. Use Cascading Style Sheets (CSS) instead of tables to organize page content.

Why. The strength of CSS lies in the ability (a) to separate content from presentation and (b) to allow for more precise control over the layout. With this control, you can change the linear layout of a page without altering its visual layout. Linear layout refers to the order of

elements in a Web page when all styles and formatting are removed. The linear reading order of Web content is important to screen reader users, for example, because that is the order in which screen readers access the content. Screen readers ignore any styles within Web content. All that is left is the content, without any presentation. This stripped-down version of Web pages is all that screen readers ever have access to. The size of any page will change, using CSS, to fit its content. This scalability feature produces cleaner sites.

An additional benefit is that sites created with CSS are easier to maintain than multiple stand-alone pages created with tables: a change to the style sheet is promulgated across all pages. This helps maintain accessibility.

On the other hand, a downside to using CSS vs. tables is that tables tend to be more stable or predictable across browsers.

For more information. See WebAIM's *Overview of CSS*, <http://webaim.org/techniques/css/>, which is helpful and has illustrative examples. Jim Thatcher's chapter on CSS, <http://www.jimthatcher.com/webcourseb.htm>, is also helpful. *CSS Zen Garden*, "a demonstration of what can be accomplished visually through CSS-based design", <http://csszengarden.com/>, exemplifies how CSS can be used to make a website both more accessible and more usable with wonderful design. *Layout Gala*, <http://blog.html.it/layoutgala/>, has 40 CSS layouts with validated HTML and CSS that help with managing headers, footers, nav bars, main content, and other parts of page structure.

4. **Tag images that are meaningful and integral to the content of the page with alt text** (aka "The Campanile Guideline," where Campanile is pronounced cam-pa-nile – like the river Nile – the way most screen readers pronounce it)

Why. If an image does not add meaning, then its use is superfluous to a screen reader user; it simply slows the process down with no benefit. If an image does add meaning, then use alt-text – a tag that describes the image and/or its purpose.

What. Provide a text equivalent for every non-text element that is meaningful or adds value to the content of the page. These include 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. When a short text equivalent does not suffice to adequately convey the

function or role of an image, provide additional information in a file designated by the longdesc attribute.

If an alt tag is not used, simply leave the quote marks with nothing between them, e.g., ``.

For more information. Dive into Accessibility's, <http://diveintoaccessibility.org/>, 30-day course on making websites accessible has a section (Day 23) on using alt tags, http://diveintoaccessibility.org/day_23_providing_text_equivalents_for_images.html. The examples of bad and good alt tags are especially useful. Caveat: At the top of Day 23, there is a statement that all images need alt text. Keep your alt texting to meaningful images.

5. **Use color with care.**

Why and what. If color is used to impart meaning, e.g., green always means "go," while red means "stop," that meaning will be lost to a person with vision impairment – including the estimated 9% of men who are color deficient or color blind. (The greatest color deficiency is around red and green.) Color contrast is also important: the eye must work harder (and therefore becomes tired more easily) if low-contrast background colors are used, for example, to separate columns in a page. Higher-contrast color combinations, or a simple line, are more effective in differentiating sections of a Web page.

Dueling disabilities. A "dueling disability" is when the "fix" for one type of disability cancels out a benefit to another disability type. A perfect example of dueling disabilities related to Web accessibility is that associating color with meaning actually can benefit people who are learning disabled, if color is used consistently. Use color with care – and awareness.

For more information. See Joe Clark's comprehensive discussion of color deficiency (aka "color blindness") and other issues related to color contrast, <http://joeclark.org/book/sashay/serialization/Chapter09.html>. Juicy Studio's, <http://juicystudio.com/>, *Colour Contrast Analyzer*, <http://juicystudio.com/services/colourcontrast.php>, provides more examples of color contrasts that are effective. The University of Washington's *Disabilities, Opportunities, Internetworking, and Technology* (DOIT) website has links to other helpful articles on color contrast: <http://www.washington.edu/doit/Resources/color.blind.html> and http://www.lighthouse.org/color_contrast.htm. And the *Ishihara Test for Color Blindness* on Frank Dutton's site about color deficiency is another helpful resource, <http://www.toledobend.com/colorblind/Ishihara.html>.

6. **Give links unique names** and make them descriptive (aka "The Click-here Guideline" or "The More...Guideline").

Why. Screen readers recognize links and make them searchable. Unique names facilitate that search.

What. If, for example, a page consists of links to pdf files that are all related to the same subject, e.g., graduate students, avoid naming them all "Graduate students: [Topic A]," then "Graduate students: [Topic B]." Instead, use the topic name itself, e.g., "A," then "B," and so on. Avoid repetitions.

In addition to making a link's name unique, make it descriptive so that it is understood out of context. Screen reader users can adjust their software to read only the links on a page. For this reason, links should provide enough information when read out of context. Use a more descriptive phrase than "click here" or "more" as a link or next to a graphic used as a link. For example:

```
<a href="about.htm">Click here</a> for information about our company.
```

will present "Click here" as the link. However,

```
<a href="about.htm">Information about our company. </a>
```

will present "Information about our company" to a screen reader user. This type of displays descriptive text as the link.

For more information. Jim Thatcher's tutorial (see Section 4, Point 4.3, <http://www.jimthatcher.com/webcourse4.htm>) describes some of the unintended consequences of link naming.

7. **Use "Go" or "Submit" buttons, with select menus, combo boxes and drop-down menus** (all the same).

Why. If there is no separate "Go" or "Submit" (or any other clear title) button that activates the chosen item in a combo box, then whichever one is encountered first by a screen reader will be activated. This has the unfortunate effect of either sending the screen reader in endless loops or breaking the page.

What. Place a "Go" or "Submit" button immediately after combo boxes so that the user can make her all selections first, then "submit" it.

For more information. Jim Thatcher's tutorial on accessible forms, <http://www.jimthatcher.com/webcourse8.htm>, goes into even more

detail on forms, including suggestions for using radio buttons and checkboxes.

8. **Include a well-positioned label for each field on a form** (aka “The-What-am-I- supposed-to-fill-in-here Guideline”)

Why. For blind or low-vision Web users, it is really frustrating to not know what information to type in a text field on a form or if, for a checkbox or radio button, the box that is checked or button that is chosen is actually the one associated with the correct label. This happens because of the misplacement of the associated labels or because of labels that are completely missing (think of a form asking for address information where there are three lines provided for street address and only the first one is labeled, e.g., “Street address.” Label placement, likewise, is critical.

What. Label each field and place the label close to the field, usually immediately to the left or immediately to the right (although left is preferred) of the field to be filled in. Using the <label> tag and the title attribute for form elements can help. Avoid a text label that is visually next to the input field, but is separated from the field in the code, thereby confusing the screen reader and the screen reader user. The linearized view that the screen reader uses may not help the user understand whether the label applies to the field before it or the field after it.

For more information. Jim Thatcher’s tutorial on accessible forms, <http://www.jimthatcher.com/webcourse8.htm>, goes into even more detail on forms, including suggestions for using “Go” or “Submit” buttons, radio buttons, and checkboxes.

9. **Ensure that all content can be accessed with the keyboard alone in a logical way by using tab order.**

Why. WebAim describes the “why” well:

Keyboard accessibility is one of the most important aspects of disability access. Blind people generally cannot use a mouse because they cannot see where to click. They use their keyboard almost exclusively. (Some legally blind people have limited residual vision which allows them the use of a mouse if the page is enlarged and the contrast is high.)

Some individuals with neuromuscular impairments cannot use a mouse either. Some have tremors which don't allow for fine muscle control. Others have little or no use of their hands, due to a spinal cord injury, brain damage, or some other cause. Some people simply do not have hands,

whether due to a birth defect, an accident, or amputation. There is an almost limitless list of conditions that could make mouse usage difficult or impossible.

Those who cannot use a mouse may not be able to use a keyboard either. Some use "puff and sip" devices activated by airflow from the mouth. Others use single-switch devices - consisting of a piece of hardware with one button that can be pushed. These devices are sometimes configured to interpret Morse code, which the person performs by tapping on it. One thing that all of these devices have in common is that they interact with the computer in a way that mimics the functionality of the keyboard. If a Web site is not keyboard-accessible, you will shut many people out.

What. Assign the tab order to form controls and/or links. Test the tab order by tabbing (not mousing) through your page.

For more information. WebAim has a useful discussion on keyboarding that includes information about tab order, <http://www.webaim.org/techniques/keyboard/>.

10. Instead of providing a "text-only" alternative, work to make your primary website accessible.

Why. Most "text-only" alternatives to websites are out of date. This, along with the increasing effectiveness of assistive technologies such as screen readers, reduces the value and effectiveness of "text-only" alternatives.

What. Instead of using limited resources to maintain mirror sites, concentrate on the previous top nine tips!

For more information. See W3C's information on tab order, <http://www.w3.org/TR/WCAG10-HTML-TECHS/#forms-keyboard-access> and <http://www.w3.org/TR/WCAG10-HTML-TECHS/#link-accesskey>.

Bonus tip. "Printer-friendly" options can be helpful, though, with the twin benefits of improving both accessibility and usability.

Other resources

- UC Berkeley's "campus commitment" to technology accessibility:
http://acads.chance.berkeley.edu/technology_access.shtml.
- Section 508 of the Rehabilitation Act.
 - For the official U.S. government standards, see section 1194.22,
<http://www.section508.gov/index.cfm?FuseAction=Content&ID=12>.
 - For a pass-fail perspective, see
<http://webaim.org/standards/508/checklist>.
- World-Wide Web Consortium (W3C), Web Content Accessibility Guidelines (WCAG).
 - Version 1.0: <http://www.w3.org/TR/WAI-WEBCONTENT/>.
 - Version 2.0: <http://www.w3.org/TR/2005/WD-WCAG20-20051123/>.
- The Wave, an on-line accessibility evaluation tool,
<http://www.wave.webaim.org/index.jsp>.
- WebAIM, Web Accessibility in Mind,
<http://www.wave.webaim.org/index.jsp>, at the Center for Persons with Disabilities at Utah State University – an all-around helpful resource.
- Jim Thatcher's website, <http://jimthatcher.com>, is a wonderful and comprehensive resource. His list of links to other on-line resources is exhaustive.

Four steps for redesigning an existing site for accessibility

1. Evaluate the current site.
 - Review the site using an automated tool.
 - Review the site by using a screen reader and a screen magnifier – used by someone who knows how to use the software.
 - Review the site with the monitor set to greyscale to ensure color blind accessibility.
 - Review the site with images turned off in the browser.
 - Attempt to navigate the site using the keyboard alone (i.e., with no mouse or tracking device).
 - Attempt to navigate the site using the mouse and your non-dominant hand.
2. Decide whether you will modify the current pages or start from scratch.
3. Modify the code to address the issues found in Step 1.
4. Review the site again using at least two methods listed in Step 1.

Six steps for developing or designing a new site for accessibility

1. Set the level of accessibility you are aiming to achieve (e.g., fully compliant with Section 508 standards and Priority 1 of the Web Content Accessibility Guidelines – both referenced in the Berkeley campus commitment to technology accessibility).
2. Consider the technology you will use: Does it support or assist in producing accessible content (e.g., check the use of Flash or Web authoring tools that support the development of accessible code)?
3. If you are not producing the pages yourself, make sure that the person who is is familiar with the campus accessibility guidelines and tools.
4. [If the site is being developed by a vendor, the requirements must include compliance with the campus accessibility guidelines.]
5. As pages are coded, they should be reviewed with an automated tool (as a minimum) by the person coding them.
6. When the site is completed and before release, review using the evaluation process described in Step 1 of *Four steps for redesigning an existing site to make it more accessible* just above.

The Berkeley Web Accessibility Group

The Berkeley Web Accessibility Group (aka Web Access) has been meeting regularly since January 2005. Comprised of staff from throughout the campus, Web Access has also included students, faculty, and post-docs. Guests, who sometimes join the group, have been from both campus and outside, e.g., from other universities such as San Jose State and UC Davis; from vendors such as Macromedia and InterWrite; and from other organizations such as the IBM's Human Ability and Accessibility Center, <http://www-306.ibm.com/able/index.html>, and Berkeley's own Center for Accessible Technology, <http://www.cforat.org/>.

Web Access is devoted to:

- Learning about disabilities that present barriers to full use of the Web and the technologies that can promote Web accessibility;
- Collaborating with and on behalf of all representatives of the Berkeley community – students, faculty, staff, and the public; and
- Working directly with campus Web developers to improve their sites.

Interested in learning more? Contact Dianne Walker, Project Engineer at IST, 643-0290, dwalker@berkeley.edu or Lucy Greco, Assistive Technology Specialist, Disabled Students Program, 643-7591, lgreco@berkeley.edu.

~~~~~  
With special acknowledgement to those who've been part of the learning...

- |                      |                          |                    |
|----------------------|--------------------------|--------------------|
| o Abigail Tilden     | o Ian Crew               | o Paola Timiras    |
| o Adam Hochman       | o Jack McCredie          | o Patrick Civello  |
| o Allison Bloodworth | o Jane Berliss-Vincent   | o Patrick McMahon  |
| o Andre Porter       | o Jerry Yoon             | o Paul Nozicka     |
| o Arnold Yip         | o Jim Betinol            | o Rick Hill        |
| o Aron Roberts       | o Jim Gammon             | o Rob Pedlow       |
| o Bernie Rossi       | o Josh Marcus            | o Sara Leavitt     |
| o Betsy Livak        | o Kalle Nemvalts         | o Sarah Hawthorne  |
| o Bill Sprenger      | o Karen de Valois        | o Sarah Jones      |
| o Bob Regan          | o Karen Eft              | o Scot Hacker      |
| o Caroline Boyden    | o Karin Bliman           | o Shel Waggener    |
| o Cathy Taruskin     | o Kathleen Phillips Satz | o Shelley Glazer   |
| o Chris Ashley       | o Kathryn Bader          | o Steve McConnell  |
| o David Willson      | o Kelly Haberer          | o Stuart Fryer     |
| o Debra Goldentyer   | o Kimberly Choo          | o Sue Schweik      |
| o Derek Coates       | o Lilianna Denizot       | o Susan Hedgpath   |
| o Diane Hill         | o Lisa Harrington        | o Susan McKeehan   |
| o E. Marie Robertson | o Lucy Greco             | o Tom Holub        |
| o Ed Rogers          | o Marc Buchalter         | o Tony Christopher |
| o Elise Jok          | o Melani King            | o Tony Roybal      |
| o Gary Thackeray     | o Meriah Hudson          | o Tu Tran          |
| o Gina Blednyh       | o Mimi Mugler            | o Vam Makam        |
| o Helena Tso         | o Natalie Kato           | o Victor Edmonds   |
- ~~~~~