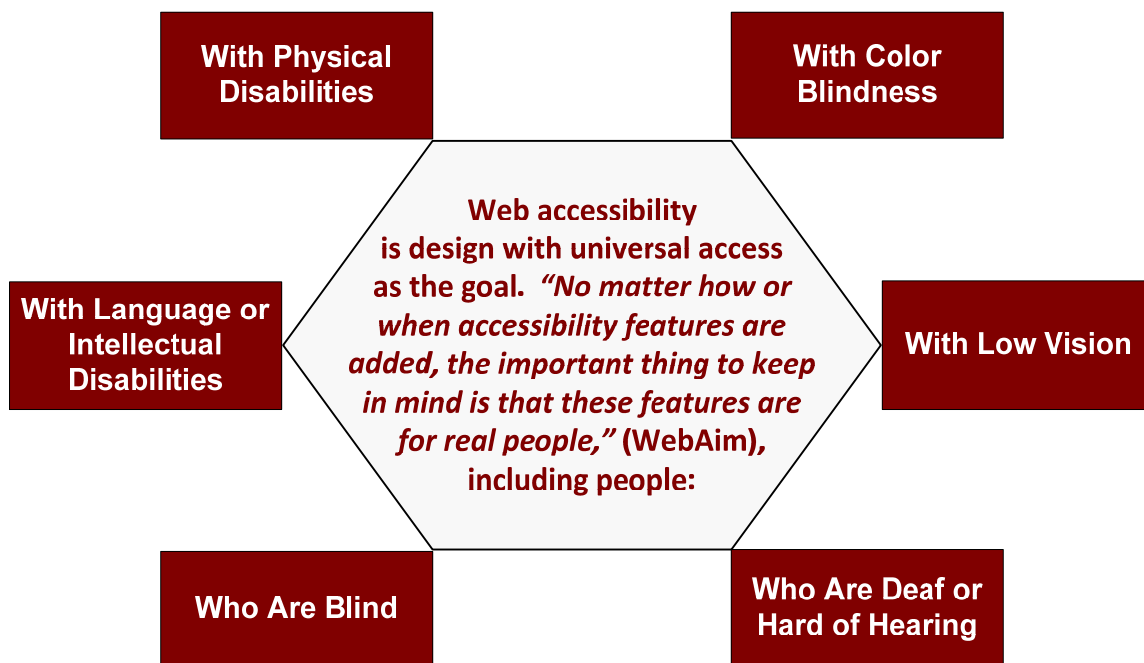
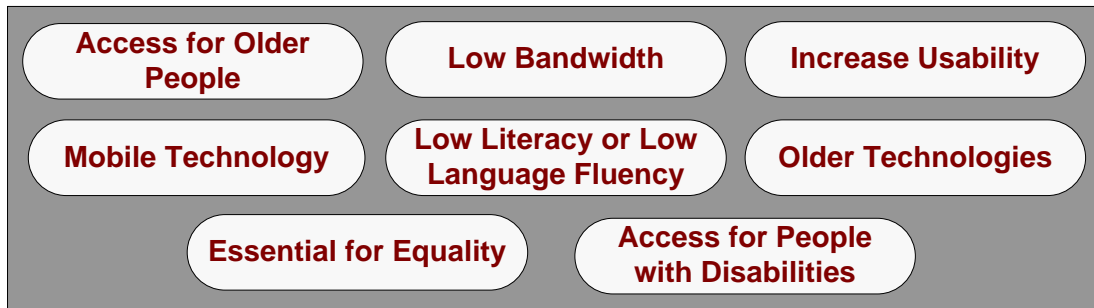


ACCESSIBLE-IZE YOUR WEBSITE

WHY?

SOME GOOD REASONS.....



WebAim

Listed here are some of the accessibility processes that can make the World Wide Web accessible to people with disabilities. For more details and instructions on implementation, see the resources below.

Navigation – (Screen readers can see text even if that text is written to the screen invisibly)

- A process should be employed that allows users to skip repetitive navigation links.
- Provide keyboard equivalents for all menus, and dialog boxes.
- Avoid non-text menu items or, if necessary, integrate visible or invisible text cues for these items.
- Screen readers have the ability to see text even if the text is written to the screen invisibly.

Page Appearance

- Use simple, consistent and predictable screen and dialog layouts.
- Use style sheets (CSS) to control the presentation of the page, though the page should be readable without CSS.
- Avoid specifying fonts (types) directly.
- Avoid auto-refreshing pages.
- Avoid built-in timed responses or lengthen the interval allowed for a user to respond.
- Design pages to avoid causing the screen or part of the screen, such as graphics, to flicker with a frequency greater than 2 Hz and lower than 55 Hz.
- Allow the user to zoom in on or magnify portions of the screen.
- Structure pages using logical heading levels (H1-H6).

Testing Page Appearance

1. Validate CSS using tools available on the internet (<http://jigsaw.w3.org/css-validator/>).
2. Check page without CSS so that it shows a logical, understandable structure.
3. Scale pages in the browser and check to see that they do not trim or get cut off.
4. Check for paragraphs and form labels.
5. Ensure headings make sense out of context (as a screen reader user would hear them).

Links (Hyperlinks)

- Ensure meaning of the link is contained within the link or can be determined from the context of the link.
- Keep descriptive text short and sweet. (Remember screen readers read every link.)
- Avoid “click here”, “more”, as this means nothing out of context.

Testing Links

1. Some automated tools can flag generic link text.
2. Scan and read links out of context.
3. Download a JAWs (<http://www.freedomscientific.com/downloads/jaws/jaws-downloads.asp>) or Window-Eyes (<http://www.qwmicro.com/Window-Eyes/Demo/>) demo and check links and other page elements for accessibility.

Keyboard

- Keyboard actions should be able to be completed without a mouse.
- Provide keyboard access to all toolbars, menus and dialog boxes.

Testing Keyboard

- Use the tab key to follow tab order.
- Check that Adobe® Flash content responds to the tab key for navigation.
- Ensure a logical flow in forms (tabbing).

Scripts

- 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, or a comparable process.

Color

- Make color coding an unnecessary or secondary means of conveying information.
- Increase the contrast between text and the background.
- Text is easier to read over a solid-color background. Patterned backgrounds can make text harder to distinguish.
- Ensure that the web page will be usable if displayed in monochrome mode (i.e., grayscale).
- Allow users to vary contrast, brightness, and color.

Testing Color

1. Color Contrast Analyzer: downloadable tool available in the Web Accessibility Toolbar.
2. Check forms, graphs, charts, and instructions for reliance on color.

Images

- Every image, applet, embedded media, plug-in, etc. that conveys content has equivalent alternative text (alt, longdesc, or in the element context).
- The alternative text (alt text) concisely describes the content conveyed by the element, without being too wordy or too ambiguous.
- Decorative graphics or CSS background images or have null/empty alt values (alt="").

Testing Images

1. Switch off images via the browser (allows to check existence of alt tags and if they make sense).
2. Check for appropriate "alt text" that makes sense if the picture does not show up.

Image Maps

- Images that have a function (images within links, image buttons, and image map areas) have alternative text which describes the associated function.
- The image map **and** each area in the map must have alternative text.
- Avoid assigning unlabeled hot spots to pictures for use as controls.

Forms

- Clearly label all text fields, text areas, drop-down menus, checkboxes, and radio buttons.
- Label all buttons (Submit, Clear, etc.).
- Associate form labels to form items (i.e., text input, check boxes) using label tag.
- Place label for form adjacent to the form element.
- Allow the user to make a selection, **and then**, press a button to invoke the action.
- Always place the group identifier before the information so that it has proper association.
- Position radio buttons before their labels. This approach ensures that all screen readers associate the proper labels with their respective form controls.
- Allow users to change an input and correct an error with explicit information.

Tables

- Use column and row headers in tables unless they are layout tables.
- Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.
- Use HTML to mark up tables.
- Provide alternative access to static tables or graphic representation of tables.

Access to Multimedia Presentations

- Provide all auditory information visually (text or captions).
- Provide captions with all multimedia presentations.
- Video files and live audio broadcasts have synchronized captions.
- Content presented through video, but not through audio, is provided via a description of the video.

Avoid Using “Pop-Up” Windows

- Avoid the use of “help” or message balloons that disappear whenever the hot spot, or focus of the mouse, changes.
- Locking the help balloon in place lets user move the cursor and continue to read the balloon.

Test Your Website

“No automated accessibility evaluation tool can find all types of accessibility errors. Automated programs can only evaluate a few of the many possible accessibility issues that can arise in a particular web site.” (WebAim)

Testing – Check Your Website

Functional Accessibility Evaluator 1.1 (University of Illinois at Urbana-Champaign): <http://fae.cita.illinois.edu/>

WAVE: Web Accessibility Evaluation Tool: <http://wave.webaim.org/>

List of Web Accessibility Evaluation Tools: <http://www.w3.org/WAI/ER/tools/complete>

Resources

Web Accessibility Initiative (WAI): <http://www.w3.org/WAI/>

WEBAIM: Web Accessibility in Mind: <http://webaim.org/articles/>

U.S. Government Web Accessibility Standards: <http://www.section508.gov/index.cfm>

iCITA: Illinois Center for Information Technology and Web Accessibility:
<http://test.cita.illinois.edu/aria/index.php>

NCAM: National Center for Accessible Media: <http://ncam.wgbh.org/>



Produced by Northwest ADA Center, funded by a grant from U.S. Department of Education,
National Institute on Disability and Rehabilitation Research

Alternate formats available upon request.