

Website Accessibility: An Overview of Issues and Solutions
Hawai'i Student Success Institute, March 3, 2017
Ralph Toyama, Systems Librarian, Leeward Community College
rtoyama@hawaii.edu

When designing a web page, there are things you should do to make it more accessible to people with sensory and physical disabilities.

To learn more about web accessibility in general:

<https://www.w3.org/WAI>

<http://webaim.org>

UHCC websites should follow Web Content Accessibility Guidelines (WCAG) Level AA.

Also be aware of Accessible Rich Internet Applications. ARIA attributes can be used to address accessibility problems that standard HTML doesn't address.

Accessibility Tools

Web authoring tools can try to provide accessibility features, but they can't guarantee accessibility because there comes a point where a human has to make a decision about how to describe something.

For similar reasons, accessibility checkers aren't completely accurate. Sometimes they miss things, get fooled, and flag things that aren't really problems. You still need to carefully consider your content and what you can and need to do to communicate it.

One popular accessibility checker is at <http://wave.webaim.org/>. They also have a Chrome extension you can install that puts a button on your browser that lets you quickly check a web page. A fairly stringent checker is at <http://cynthiasays.com/>

Technology Overview

What is a Web Page?

The heart of a web page is an HTML file. At its most basic, it consists of the words of the web page, marked up with codes that indicate the function of the words within the document. It can also have codes that make graphic images appear within the web page. The HTML codes that mark up the page content are called "tags". Tags can contain additional instructions called "attributes".

A web page uses styling codes (called CSS) to control the appearance and position of things on the page, and scripting to add functionality to the page. Styling and scripting might be in separate files linked to the HTML file, or they may be included in the HTML file. HTML, CSS, and script files are text files.

What is a Text File?

A text file is a string of binary codes that represent individual text characters – letters, numbers, & symbols – as well as a handful of control codes. For example, the code 1000001 represents the capital letter “A”.

You can open a text file with text editing program like Notepad (Windows) or TextEdit (Mac). Depending on where your web page is hosted, you may or may not be able to directly edit the file with a text editor, but within web authoring software there is usually a “source” mode. There are situations where editing the source code is the easiest – and sometimes only – way to fix an accessibility problem.

When words are encoded as text, they are easily shared between computer programs. Adaptive technology like a screen reader can take the text from a web page and convert it to speech.

You can tell something is encoded as text when you can highlight it with your mouse.

Arranging and Marking Content

First, try to write your content in a way that you don’t have to rely on visual formatting to communicate your message. This can be hard to do because of competing design considerations, but the closer you can get to this ideal, the easier it is for someone who can’t see the formatting to understand the content.

Use HTML tags to indicate the function of the different parts of your document. This gives semantic context that helps adaptive technology to effectively present the context.

Common tags:

- Mark your page’s main heading with <h1> tags

- Mark your subheadings with <h2> tags

- If your page is organized into deeper layers, you can use <h3> through <h6>

- Mark your paragraphs with <p> tags

- Create HTML lists when appropriate

- Use tables only for presenting tabular data. Don’t use tables to control layout.

What you should not do is rely on styling the text. You can make something look like a heading by making it large and bold, but those visual cues are not available to someone who is listening to a screen reader.

For more information on using HTML, CSS, JavaScript, and other web authoring tools:

<https://www.w3schools.com>

Declaring the Primary Language of the Page

This can help adaptive technology properly present the page content. Within the <html> tag near the top of the file, include this attribute – lang="en-US"

Using Graphics to Convey Information

Words that are part of a graphic image are not accessible.

You can often use CSS to add visual impact without relying on graphic images. For example, if you want to emphasize a point by putting a statement in a colored box, you can use CSS to create a box with a colored background and rounded corners, and put your statement in the box as a line of text. Not only is this the best way to present information for screen reading technology, but when your information is on the screen in the form of text, it enlarges well when you use the browser's size control.

When you do use graphics, you may have to provide alternative text that either describes the image, describes its function, or provides the information given in the image.

The tag is used to specify a graphic image to be displayed on a web page. All tags should contain an "alt" attribute. For purely decorative graphics, the alt attribute may be empty – alt="". An empty alt attribute tells a screen reader that the image can be ignored.

```
  
  

```

If an image can't be adequately described briefly, a "longdesc" attribute in the tag may be used to provide a link to longer description, either on another web page or somewhere else on the same page. However, because longdesc support is inconsistent between browsers, some recommend that a conventional link to the description also be given near the graphic.

```
  
<a href="festivaltext.html">Accessible text</a>
```

A long description formatted as a web page with appropriate markup can be communicated more effectively than a long string of words.

For guidance in composing appropriate alternative text, see <http://webaim.org/techniques/alttext>

Flashing , Flickering, or Strobing Images

Graphic images or videos that flash prominently at a rate of 2 or 3 times per second or faster can cause seizures in those with photosensitive epilepsy. Also be aware that some moving images can induce nausea and dizziness.

Navigation

You can put a “skip navigation” link near the top of a page that allows a user of a screen reader to skip over a long navigation menu and go directly to the content of the page. You can do this anywhere you think it might be helpful. To minimize visual impact on your page, you can make these links inconspicuous by making them small and in a color that blends in with the background.

The traditional way to do this was to place a name anchor tag where you want the link to jump to. The current recommended way is to assign an ID name to the object you want to jump to.

The link would be coded as:

```
<a href="#theimportantstuff">skip navigation</a>
```

The traditional method:

```
<a name="theimportantstuff"></a><p>The actual page content starts here...</p>
```

The currently recommended method:

```
<p id="theimportantstuff">The actual page content starts here...</p>
```

Linking to Other Pages

It is recommended that when you link to other pages within your website, that you open the new page in the same tab (which is usually the default behavior). You can have pages open in new tabs, but that makes it harder to trace your steps backward if you can't see the tabs or use a mouse.

Focus

Things that can accept user input on a web page can have a state known as “focus”. These can be things like links, buttons, form fields, and other designated objects. Only one object has focus at a time. You can use the Tab key to step through them, generally in the order in which they occur in the page code (or Shift-Tab to step through them in the other direction). This is important for people who have difficulty using a mouse, and have to rely on the keyboard or other adaptive device.

You can control how something that has focus appears on the screen. By default, browsers will usually indicate focus with a faint outline around the object. Some platforms or designers disable this for aesthetic reasons, but it's important to be able to see which object has focus. If you have put hidden navigation links on your page to help screen readers, you should make sure that when these links have focus, they are displayed visibly.

Scripts

A scripting language like JavaScript can add functionality to a web page. But there are accessibility concerns to be aware of:

Scripts That Cause Something on the Screen to Change

Look up the attribute `aria-live`, which you can use to tell the browser that a particular part of the screen might change, and specify what to do when it does.

Controls That Are Only Workable with a Mouse

Just about anything on the screen can be made to trigger some action when you click on it, but many of these objects cannot be activated with the keyboard. Adding the `tabindex="0"` attribute can allow the object to receive keyboard focus.

Scripts That Do Something on a Timed Basis

Ideally, the user should be able to pause or alter the timing of an action that occurs on a timed basis. This can be a difficult thing to deal with, especially if you're trying to put a web widget on your page that you have little or no control over.

Forms

Web forms are often laid out in a manner in which the labels are to the right of the thing being labelled, which is understandable visually, but confusing when a screen reader reads the form from left to right. Therefore, form elements should be linked to their labels using the `<label>` tag or the `aria-label` and `aria-labelledby` attributes.

Using JavaScript to manage form data can cause accessibility problems. You should test forms to make sure you can fill them out using only the keyboard.

For more information, see <http://webaim.org/techniques/forms/>

YouTube Closed Captioning

YouTube now attempts automatic captioning on all uploaded videos, but the results are not perfect. Wait several hours or a day after you upload a video and check and correct the automatically-generated captions. Alternately, you can paste a text file of captions into their editing interface, and their system will automatically set the timing of the captions.

Providing Alternative Web Pages

Providing Alternate Web Pages

Sometimes the best solution to an accessibility problem is to provide an alternative web page that provides the same information or function in a different manner. Sometimes even when you solve the problems, the solution is not as user friendly as having something presented in a more straightforward layout.

Sample Web Page Code

An example of HTML markup, language declaration, a link to an external stylesheet, a skip navigation link, a list used for website navigation, and a graphic image with alt and longdesc attributes.

```
<!DOCTYPE html>
<html lang="en-US">
<head>
<link rel="stylesheet" type="text/css" href="mysite.css" media="screen" />
</head>
<body>

<h1>Title of My Document</h1>
<a href="#pagecontent">skip navigation</a>

<ul>
<li><a href="page1.html ">Menu Item 1</a></li>
<li><a href="page2.html ">Menu Item 2</a></li>
<li><a href="page3.html ">Menu Item 3</a></li>
</ul>

<h2 id="pagecontent">The First Section</h2>
<p>This is the content of this section.</p>
<p>This is more content of this section.</p>

<h2>The Second Section</h2>
<p>This is the content of this section.</p>
<p>This is more content of this section.</p>


<a href="infographictext.html">Accessible text</a>

</body>
</html>
```