Web Accessibility Checklist

MARKUP

- Separate structure from presentation and use proper markup for that structure. For example, mark up lists as lists (`<ul>`, `<ol>`, `<dl>`) rather than text with a `<br>` tag after each list item.

- HTML headings (e.g. `<h1>`) are very helpful for blind users. Properly mark up the sections of a page and body copy with HTML headings rather than something such as a `<p>` tag with CSS styling that makes it look like a heading.

- Give pages meaningful and accurate titles using the `<title>` tag.

- Indicate the primary human language of the document using the `lang` attribute in the `<html>` tag, and indicate any passages in a secondary language using the `lang` attribute on other tags wrapping the relevant text (e.g. "`<span lang="es">Hola</span>` means Hello").

- Provide "Skip to content" links at the top of the markup order in pages with large numbers of navigational links before the main content.

- Always indicate headers in data tables using `<th>` tags, and associate all data cells with their header.

- Be sure tab order is logical using `tabindex`, if necessary. (If your HTML is in the proper order, then using `tabindex` isn't necessary.)

VISUAL APPEARANCE & CONTENT

- Be sure your page is still usable when images are turned off. (This may include making sure that contrast is still sufficient if you happen to be using a background image and that image is removed.)

- Be sure pages remain usable when users enlarge text up to twice its original size.

- Be sure each element on a page is reachable and can be manipulated via the keyboard.

- Whenever possible, write descriptive headings and link texts which can be understood when read out of context (e.g. no "click here" links).

- For color-blind and low-vision users, be sure your content and background have sufficient contrast.

- Do not use content that flashes or blinks more than three times a second.

- Do not hide the focus indicator. When a user uses the keyboard to tab from element to element, it should always be apparent where they are.

- Do not require users to perceive font, color, or other styling changes in order to understand meaning. For instance, don't say, "The highlighted word in the previous paragraph is the most important," or "Items marked in red are errors and need to be corrected," unless the word or items are indicated in some other way.

DYNAMIC CONTENT

- Do not use JavaScript events that radically alter the page or load a new page when fired.

IMAGES & MULTIMEDIA

- Be sure **all** images have an alt attribute, leaving the text for decorational images blank (e.g. `alt=""`).

- Always add `alt` text when images are also links.

- In general, be brief with `alt` text (e.g. "Notre Dame Cathedral"), but provide detail when it conveys meaning (e.g. "Son standing at his father's graveside with mother in arms").

- Provide a transcript, captions, and/or sign language translation for all audio and video with speech.

- Provide a "described" version of a video when description is necessary for unsighted users to understand content. (The described audio track can either be distributed with the video content, or as an audio only file.)

- Be sure that all videos, if they don't autoplay, have, at the very least, an accessible Play control.

- When text can be rendered just as well by the browser as it can in an image, avoid using images for text. (Image replacement techniques are often an acceptable alternate, but also consider translation requirements when using text in or as images.)

- Avoid CAPTCHAs unless you have no other choice, and even then they should be avoided. However, if you must use them, provide an audio CAPTCHA alternative.

FORMS

- Always label **all** form fields with the `<label>` tag. If a form field has no specific text label on the page, add one, and hide it with CSS or use the `title` attribute.

- Use fieldsets (`<fieldset>`) with legends (`<legend>`) to associate prompts with radio buttons and check boxes. For instance, a form asks "Gender:" and offers radio buttons that say "Male" or "Female". Then "Gender:" should be enclosed in a `<legend>` tag, and all three elements (`<legend>` and the two radio buttons with their label text) should be enclosed in a `<fieldset>` tag.

- Identify all input errors in text (in addition to any images or icons), and place the error notification either next to the affected field or in a prominent location such as the top of the page with an anchor link to the affected field.

- Provide help links or inline instructions for completing fields when necessary.

- Do not permit users to complete important actions without a confirmation or a way to undo.

- Avoid using HTML elements in nonstandard ways (e.g. form elements for navigation, links for form submission, etc.).

TESTING

- Test all pages for markup validation (http://validator.w3.org). If your page does not pass validation, there should be a good reason for it.

- Test all pages for color blindness using simulators or browser plug-ins. (Recommended: http://colororacle.cartography.ch or http://www.vischeck.com)

- Test all pages for accessibility using an evaluation tool (http://wave.webaim.org), but only after you have done all you can to ensure it is accessible using the recommendations here.

- Have your pages reviewed by an accessibility expert.