WEBSITE ACCESSIBILITY TESTING

T.M. WEISSENBERGER
VERSION 1.0 • JULY 2012

Unit Website Accessibility Assessment ................................................................. 1
Unit-Managed Site Review .................................................................................. 2
Site Testing Tools ............................................................................................... 3
ALT Text/Text Alternative Testing ....................................................................... 4
Heading Level Testing ....................................................................................... 5
Keyboard Accessibility Testing ............................................................................ 6
Color Contrast Testing ....................................................................................... 7
Forms Testing ..................................................................................................... 8
Data Tables Testing ............................................................................................ 9
Document Level Accessibility Testing ............................................................... 10
Compliance Sheriff Testing ............................................................................... 11
Site Assessment Checklist ............................................................................... 13
Unit Website Accessibility Assessment

A website accessibility assessment is a means of evaluating the accessibility of your website. Because many accessibility principles reflect standard design processes and practices, site accessibility assessments can provide value beyond WCAG compliance.

A site accessibility assessment can take place at any point in a website's lifecycle, but may be most useful in the early phases of the design process, when it is relatively easy to develop and manage site features at a global level. Early implementation of accessible development principles can significantly increase access and improve usability for all users. After deployment, site assessment can play a role in identifying maintenance and management tasks, and determining when the site is due for tweaks, high-level maintenance, or a complete reboot.

This document offers a list of tools and techniques for evaluating and improving the accessibility of your website. These suggestions are derived from a variety of well-known sources and describe practices commonly associated with accessibility efforts. Designers, developers, and site owners are encouraged to explore other accessibility resources, as well.

For more information on I.T. accessibility efforts at Iowa, please visit http://itaccessibility.uiowa.edu.
Unit-Managed Site Review

Definition
A unit-managed site review is an option for knowledgeable site owners who wish to perform a preliminary site assessment prior to receiving an assessment through Web Services.

Testing Methodology
Testing methodology is determined by the unit webmaster and site owners. Unit-sponsored site reviews are intended to address known or common accessibility issues in advance of a formal site assessment.

Recommended testing issues include:

- ALT text: alternate text for images and other non-text elements
- HTML heading structure: properly nested H1-H6 elements, describing the outline of a page
- Color contrast: sufficient color contrast for low-vision and color-blind users
- Forms labeling: label or title elements for all interactive form fields
- Tables testing: proper table structure and relationships between table components
- Keyboard accessibility testing: all content and interactivity are available via keyboard-only input

An automated site assessment, using the Functional Accessibility Evaluator or a similar tool, can also expose opportunities to improve accessibility prior to a University-sponsored assessment.

Conformance
- Unit-sponsored site reviews are undertaken at the discretion of the site owner, and can improve the initial results of a formal site assessment.
Site Testing Tools
A variety of tools are available to assist with accessibility evaluation efforts. The IT Accessibility group at Iowa maintains this list as a service to developers and other stakeholders; please contact us at ITAccessibility@uiowa.edu to make suggestions or report changes to items listed here.

Online Assessment Tools
• Functional Accessibility Evaluator
  http://fae.cita.uiuc.edu
• WAVE Online Tool
  http://wave.webaim.org

Assessment Tools for Mozilla Firefox
• WAVE Toolbar
  http://wave.webaim.org
• JuicyStudio Toolbar
  http://www.juicystudio.com
• Web Accessibility toolbar
  http://firefox.cita.illinois.edu
• Web Developer Toolbar
• FANGS

Assessment Tools for Microsoft Internet Explorer
• Web Accessibility Toolbar for Internet Explorer
  http://www.paciellogroup.com/resources/wat/ie
• Colour Contrast Analyser
  http://www.paciellogroup.com/node/18?q=node/20

Assistive Technology /Screen Readers
• NVDA (Non-Visual Desktop Access)
  http://www.nvda-project.org
• JAWS (Job Access With Speech)
  http://freedomscientific.com

You can also assess the functionality of your site using manual tasks, keyboard testing, and other algorithmic methods.
ALT Text/Text Alternative Testing

Definition
Alternative text accessibility means that when non-text elements (e.g., images, form controls, embedded objects, multimedia, players, etc.) are unavailable, they are replaced by alternative text that conveys the same information and provides the same functionality as the non-text element.

Testing Methodology
1. Ensure that all `<img>` elements include an ALT attribute
2. Ensure that ALT attributes provide content and functionality (i.e., linking, focusability) equivalent to the `<img>` element
3. Avoid using background images for content or functionality
4. Ensure that images used for decoration only include empty ALT text (alt="")
5. Ensure that all embedded objects (e.g., multimedia, applets, Flash components, etc.) include appropriate alternative text when those objects are not available
6. Ensure that long text descriptions are provided for non-text elements that require detailed explanation
7. Pages with visual CAPTCHAs include CAPTCHAs that use a different modality

Tools
- WAVE Toolbar (Errors, Features, and Alerts)
  - Places overlay on current page, revealing images without ALT text
- Web Developer Toolbar (Images/Disable Images/All Images)
  - Hides embedded images, revealing ALT text
  - Hides background images
- Web Developer Toolbar (Images/Display ALT Attributes)
  - Generates overlay with ALT text

Conformance
Non-text elements meet accessibility criteria if:

- All information contained in the non-text element is perceivable by the user
- All functionality contained in the non-text element is operable by the user
- Decorative images contain empty ALT text
- Adjacent text and image ALT does not repeat, or "stutter"
Heading Level Testing

Definition
Headings 1-6 should be used to provide a hierarchical structure to web documents. Each major section or topic within a document should be introduced with an HTML heading, and headings should be properly nested. No levels should be skipped in a heading sequence: for example, <h2> is an appropriate child of <h1>.

Testing Methodology
1. Ensure that pages begin with a descriptive heading or title, marked up as <h1>;
2. Examine pages for logical and hierarchical heading structure;
3. Determine whether any logical heading levels are skipped;
4. Determine whether headings are used for stylistic, rather than structural purposes

Tools
- WAVE Toolbar (Outline)
  - Reveals document outline; highlights missing headings
- Web Developer Toolbar
  - Reveals document outline; highlights missing headings

Conformance
A website or application meets structured heading standards if:

- All page sections begin with a heading;
- All headings are nested appropriately, according to heading level;
- Headings are not used for formatting, sizing, or other stylistic purposes
Keyboard Accessibility Testing

Definition
Keyboard-only access means that all site navigation and functionality are available using only the standard keyboard, and that the user can move freely through the page using only the standard keyboard without becoming caught in a "keyboard trap" (in which the cursor is "trapped" in one section, widget, or functional region of the document, and cannot move to another section).

Testing Methodology
1. Access all site navigation, form controls and other functionality on a device with no mouse;
2. Determine that all content and function can be keyboard controlled;
3. Visually confirm that objects indicate focus with a visual cue;
4. Ensure that no keyboard traps exist to isolate the user in one area of the interface;
5. Review any custom key commands for conflict with the browser or operating system

Tools
- Keyboard (View page with mouse disconnected)
  - Allows tester to tab through page to determine accessibility
- Web Developer toolbar (CSS/View Style Information)
  - Tester can select an element and check CSS (e.g., for `:focus` pseudo-class) in Style Information tab
- Web Developer toolbar (CSS/Disable Styles)
  - Reveals skip links that may be positioned off-screen

Conformance
A web site or application meets keyboard accessible criteria if:

- All navigation and interface elements are operable through keyboard controls only;
- Headings and focusable elements permit interface navigation for screen readers;
- Elements change visually to indicate focus;
- No keyboard traps are present;
- Accessibility key commands do not conflict with key commands in the native application;
- No other keyboard-specific barriers are present
**Color Contrast Testing**

**Definition**
Sufficient color contrast means that for text content, all foreground/background color combinations provide sufficient contrast for low-vision users or users with color blindness.

**Testing Methodology**
1. Document background/foreground color combinations in the application;
2. Using a color contrast tool, evaluate the color combinations for compliance, OR
3. Calculate color contrast

**Tools**
- Colour Contrast Analyser
  - Evaluates contrast between two colors (eyedropper or HEX values)
- JuicyStudio toolbar (Colour Contrast Analyser)
  - Yields element by element, pass/fail grid of contrast

**Conformance**
A website or application meets color contrast accessibility criteria if:

- All background and foreground color combinations contrast at a 4.5:1 ratio for 14 point text or smaller;
- All background and foreground color combinations contrast at a 3:1 ratio for text larger than 14 points
**Forms Testing**

**Definition**
Accessible forms are fully keyboard-operable, with well-labeled fields that change visually when they receive focus. Forms also should include detection and recovery mechanisms that alert the user of invalid input, required fields, and other changes in context.

**Testing Methodology**
1. Keyboard-operability: complete and submit the form without using a mouse or stylus;
2. Labeled fields
   - Click or send focus to a field's corresponding `<label>` element; the field should take focus, OR;
   - Evaluate form `<label>` items using the Forms tool in the Web Developer toolbar; OR
   - Fields without corresponding `<label>` tags use the TITLE attribute to provide contextual information
3. Review error detection and recovery mechanisms to ensure that:
   - Cursor focuses to the message in the alert mechanism;
   - The user can navigate by keyboard within the alert interface;
   - When the user closes the alert mechanism, focus returns to the point of the error, or another appropriate location within the document

**Tools**
- Web Developer Toolbar (Forms >> View Form Information)
  - Generates a grid of fields and labels
- WAVE (Errors, Features and Alerts)
  - Reveals unlabeled form controls
- WAVE (Text-Only)
  - Exposes labels within form controls

**Conformance**
Forms meet accessibility guidelines if:
- They are fully keyboard-operable
- All fields are accompanied by meaningful `<label>` elements, or use the TITLE attribute
- Error recovery mechanisms (OS and modal) are interactive and navigable for users of screen readers and other AT
- No other barriers exist in the form
Data Tables Testing

Definition
Accessible tables are semantically sound, using <th> and <td> to distinguish between headings and data. Heading and data cells should include SCOPE, and ID or HEADERS to associate data with the correct headings. Tables should also include a meaningful SUMMARY attribute, and a <caption> element to aid in describing the purpose of the table. Tables should be used for data presentation only.

Testing Methodology
1. Ensure that the <table> tag includes an appropriate SUMMARY attribute;
2. Check for the presence of a <caption> tag inside the <table>;
3. Review row and column headers for use of the <th> tag, with SCOPE attribute;
4. Review data cells for use of the <td> tag;
5. If <th> elements use ID, ensure that <td> uses corresponding HEADERS attribute;
6. Examine the table in a screen reader (or linearize the table) to ensure that table content makes sense when presented linearly;

Tools
• NVDA/JAWS (Screen reader)
  o Yields text-to-speech rendering of table
• JuicyStudio Toolbar (Table Inspector)
  o Creates overlay to expose table headers and summary
• Web Developer Toolbar (Information/Display Table Information)
  o Displays table headers and summary
• WAVE Toolbar (Structure/Order)
  o Reveals scope and reading order of table
• FANGS and/or WAVE Text-Only
  o Yields text-only rendering of table content
  o Does not expose table structure

Conformance
A table meets accessibility guidelines if:

• The purpose of the table is the tabular presentation of data;
• Headers and data cells are marked up correctly;
• All of the component parts include appropriate, descriptive attributes;
• The SUMMARY attribute and <caption> tag are present, and descriptive;
• No other barriers are presented by the use of the table
Document Level Accessibility Testing

Definition
Document-level accessibility testing covers a number of issues in the construction of a page at the document level. Document-level tags should be properly nested, and should include specific meta-information to provide context to the user and user agent.

Testing Methodology
1. Ensure that pages begin with the appropriate Document Type Declaration for the HTML version used to author the page
2. Check each page for an appropriate, descriptive <TITLE>
3. Make sure pages contain <meta> information required by the HTML version used to author the page
4. Ensure that the <HTML> tag includes a LANG attribute to identify the human language of the page
5. Ensure that any changes in human-readable language are indicated by using the LANG attribute in the element that contains the language change
6. Ensure that pages are equally legible and functional when stylesheets are omitted

Tools
- Validate site (validator.w3.org)
  - Reveals absence of DTD, TITLE, LANG, CHARSET
- FAE Toolbar (Reports/Accessibility Issues/FAE Rule Set)
  - Reveals absence of DTD, TITLE, LANG, CHARSET
- WAVE Toolbar (Disable Styles)
  - Yields visual inspection of page with CSS disabled
- Web Designer Toolbar (CSS/Disable Styles)
  - Yields visual inspection of page with CSS disabled

Conformance
A web page meets document-level accessibility standards if:

- All of the above criteria are met
- No other barriers exist in the page construction
Compliance Sheriff Testing

Definition
Compliance Sheriff assessment is performed by Central Web Services through the Web Accessibility Coordinator. This assessment programmatically assesses conformance with multiple guidelines within the WCAG 2.0, AA specification.

Testing Methodology
1. Unit provides Web Accessibility coordinator with URL(s) of website(s) to be tested
   - Pages with a pending accommodation request
   - Top 20% of unit pages
   - Pages that are essential to the core function of the unit
2. Web Accessibility coordinator conducts assessments on static and dynamic HTML/CSS content per applicable checkpoints and checkpoint groups
3. Web Accessibility coordinator provides unit with Compliance Sheriff report and interpretation

Tools
Compliance Sheriff assessments are performed by Central Web Services on behalf of University of Iowa colleges, departments, and programs.

Conformance
A website or application meets Compliance Sheriff criteria if:

- The Compliance Sheriff score is > 80%;
- No priority issues exist in the Compliance Sheriff report;
- Accessibility issues identified in the Compliance Sheriff report are corrected;
- Other functional barriers exposed by the Compliance Sheriff report are corrected
### Site Assessment Checklist

This checklist is intended to assist site owners and other stakeholders in assessing a website for accessibility.

<table>
<thead>
<tr>
<th>TEXT ALTERNATIVES TO NON-TEXT ELEMENTS</th>
<th>Check or Practice</th>
</tr>
</thead>
<tbody>
<tr>
<td>All <code>&lt;img&gt;</code> elements include an ALT attribute</td>
<td>✓</td>
</tr>
<tr>
<td>Images used for decoration only include empty ALT text (alt=&quot;&quot;) and no TITLE attribute</td>
<td></td>
</tr>
<tr>
<td>Long text descriptions are provided for non-text elements that require detailed explanation</td>
<td></td>
</tr>
<tr>
<td>Where images and adjacent text link to the same resource, combine both in a single link or <code>&lt;a&gt;</code></td>
<td></td>
</tr>
<tr>
<td>Images included using CSS (e.g., background images, list-style-images) do not convey important information or functionality</td>
<td></td>
</tr>
<tr>
<td>All embedded objects (e.g., multimedia, applets, Flash components, etc.) include appropriate alternative text when those objects are not available</td>
<td></td>
</tr>
<tr>
<td>EMBED is accompanied by NOEMBED</td>
<td></td>
</tr>
<tr>
<td>Pages with visual CAPTCHAs include CAPTCHAs that use a different modality</td>
<td></td>
</tr>
<tr>
<td>Alternate text provides content and functionality (i.e., linking, focus) equivalent to the <code>&lt;img&gt;</code> or non-text element</td>
<td></td>
</tr>
<tr>
<td>FRAME and IFRAME elements include meaningful TITLE attribute</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KEYBOARD ACCESSIBILITY</th>
<th>Check or Practice</th>
</tr>
</thead>
<tbody>
<tr>
<td>All content and functionality are available using only the keyboard</td>
<td></td>
</tr>
<tr>
<td>Where mouse event handlers are in use, keyboard event handlers are also functional</td>
<td></td>
</tr>
<tr>
<td>No keyboard traps exist (i.e., the keyboard user can move freely throughout the page)</td>
<td></td>
</tr>
<tr>
<td>Objects are highlighted by the browser when they receive focus</td>
<td></td>
</tr>
<tr>
<td>Custom key commands do not conflict with the browser or operating system</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DOCUMENT STRUCTURE</th>
<th>Check or Practice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pages begin with the appropriate Document Type Declaration</td>
<td></td>
</tr>
<tr>
<td>Each page includes an appropriate, descriptive <code>&lt;TITLE&gt;</code></td>
<td></td>
</tr>
<tr>
<td>Pages contain <code>&lt;meta&gt;</code> information required by the HTML version used to author the page</td>
<td></td>
</tr>
<tr>
<td>The <code>&lt;HTML&gt;</code> tag includes a LANG attribute to identify the human language of the page</td>
<td></td>
</tr>
<tr>
<td>Elements that include a change in human language are use the LANG attribute</td>
<td></td>
</tr>
<tr>
<td>HEADING LEVELS</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Pages begin with a top-level heading, marked up as <code>&lt;h1&gt;</code></td>
<td></td>
</tr>
<tr>
<td>Pages maintain a logical and hierarchical heading structure</td>
<td></td>
</tr>
<tr>
<td>No logical heading levels are skipped</td>
<td></td>
</tr>
<tr>
<td>Headings are used for structural purposes only; style is handled by CSS</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CSS/ELEMENT STYLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background and foreground color combinations contrast at a 4.5:1 ratio for 14 point text or smaller;</td>
</tr>
<tr>
<td>Background and foreground color combinations contrast at a 3:1 ratio for text larger than 14 points</td>
</tr>
<tr>
<td>Font size is described in em, percent, or named sizes, and not in pixels or points</td>
</tr>
<tr>
<td>Text within form elements is scalable</td>
</tr>
<tr>
<td>Pages are equally legible and functional when style sheets are omitted</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FORMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forms are fully keyboard-operable</td>
</tr>
<tr>
<td>Form fields are associated with <code>&lt;label&gt;</code> elements, or use the TITLE attribute</td>
</tr>
<tr>
<td>Users can review and correct responses before submitting a form</td>
</tr>
<tr>
<td>Form validation and alert (error detection and recovery) occur client-side</td>
</tr>
<tr>
<td>Form validation and alert mechanisms (OS and modal) are interactive and navigable for users of screen readers and other assistive technologies</td>
</tr>
<tr>
<td>Form validation and alert include text descriptions of errors and affected fields</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables use <code>&lt;th&gt;</code> and <code>&lt;td&gt;</code> to present data</td>
</tr>
<tr>
<td>Table headers and table data cells are associated by SCOPE, ID and HEADERS</td>
</tr>
<tr>
<td>Data tables include meaningful <code>&lt;caption&gt;</code> element and SUMMARY attribute</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OTHER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in page or element context require activation, and not only focus</td>
</tr>
<tr>
<td>Lists of links are presented in the same order on different pages</td>
</tr>
<tr>
<td>Content yields no errors in Compliance Sheriff</td>
</tr>
<tr>
<td>Content yields no errors in FAE</td>
</tr>
<tr>
<td>Content yields no errors in WAVE</td>
</tr>
</tbody>
</table>