

**ADMINISTRATIVE PROCEDURES**

|  |  |
| --- | --- |
| **50.11 Web Accessibility and Usability Standards** | **Revised: October 31, 2013** |

1. GENERAL
   1. This document establishes standards on the design, development, maintenance and testing of software applications (networked, web-based and GIS) and web sites within Texas A&M Forest Service (TFS).
   2. These standards apply to applications that are developed internally or through contracts.
2. DEFINITIONS

2.1 508 Compliance – Using testing / validation tools and procedures to check web pages and web content for compliance with the [Section 508 Standards Guide](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/guide-to-the-section-508-standards) requirements of the Rehabilitation Act. [36 C.F.R. Part 1194]

2.2 W3C – World Wide Web Consortium. Additional information and copies of current standards and recommendations are available at <http://www.w3c.org>.

1. PURPOSE

3.1 Web accessibility encompasses all disabilities that affect access to the web, including visual, auditory, physical, speech, cognitive and neurological disabilities.

3.2 The purpose of these standards is to make applications and web sites accessible and usable to users, regardless of physical status, cognitive ability or technology.

1. SCOPE

This document is intended for software applications developers, web developers, content publishers and contract vendors or programmers. For brevity, henceforth they are referred to as developers.

1. ACCESSIBILITY STANDARDS
   1. Software applications and web sites shall comply with applicable standards:

* Internet Engineering Task Force if using Secure Socket Layer
* W3C if using Cascading Style Sheets (CSS)
  1. Software applications and web sites shall be validated using the W3C CSS validation service.
  2. Required accessibility standards are listed below with a brief description:

1. **Applets and Plug-Ins** – If a web page requires an applet, plug-in or other application to be present on the client system, then provide a link for download and an alternative means of accessing the content.

Examples:

* When applets, plug-ins or applications (Java applets, Java scripts, Acrobat PDF files, Flash, Real Player, Windows Media Player, PowerPoint files) are not accessible to assistive technologies, provide an alternative means of accessing the content within the applications (e.g., a mirror HTML file for a PDF file).
* When an applet, plug-in or application is utilized, provide a link to an accessible page where it can be downloaded.

1. **Color** – If a web page uses color to convey information such as numeric value (such as red to indicate a negative monetary value), hyperlinked text, importance (such as an alert or warning) or special value/meaning (such as red text to show a required field on a form), make sure that the information is equally understandable without color.

Example:

* If color is used to convey information, use alternative indicators such as an asterisk (\*) in conjunction.

1. **Electronic Forms** – If a web page user must complete an electronic form online, design the form so that people using assistive technology are able to complete and submit the form.

Examples:

* Form controls have text labels adjacent to them and keyboard access to control functionality.
* Form elements have explicitly associated labels in the markup (i.e., the id and for HTML elements).
* Dynamic HTML scripting of the form does not interfere with assistive technologies.

1. **Flickering** – If a web page includes images that flicker or flash repeatedly or uses a strobe effect, make sure that the flicker rate is between 2 and 55 Hz.
2. **Frames** – If a web page uses frames, include text and tags that will help users understand the frame’s purpose and function.

Example:

* Each frame has a title that describes its purpose or the type of information contained within the frame.

1. **Image Maps, Client-Side** – If a web page includes a client-side image map, provide alternative links and alt attributes.

Example:

* Use standard HTML client-side image maps with appropriate alt tags for the image and hot spots.

1. **Image Maps, Server-Side** – If a web page includes a server-side image map, provide a descriptive redundant text link for each active area.

Example:

* Separate text links are provided outside of the server-side image map that provides the same description to the content that image map hot spots access.

1. **Multimedia Synchronization** – If a web page includes an audio and/or video presentation, include an equivalent alternative that accommodates users with visual or auditory disabilities.
2. **Navigation** – If a web page contains repetitive navigational elements, allow users of assistive technology to skip over navigational links or filter mechanisms and go directly to content.

Example:

* Provide a link to skip over lengthy lists of links (e.g., navigational menus).

1. **Non-Text Elements**: [**Images**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1A.aspx), [**Multimedia**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1B.aspx), [**Buttons**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1C.aspx), [**Animations**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1D.aspx), [**Applets**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1E.aspx) **and** [**Scripts**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard1F.aspx) – If a web page includes non-text elements (such as graphics, images or input buttons), provide a text equivalent or description of each element.
2. [**Scripting Languages**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard12.aspx) – If a web page uses client-side scripting languages to display content or create interface elements, include functional text that can be read by assistive technology.

Examples:

* Within scripts, information is text-based or a text alternative is provided.
* All scripts (Javascript, pop-up menus, etc.) work with keyboard-accessible alternatives (either within or outside the script) that provide equivalent functionality.

1. [**Style Sheets**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard4.aspx) – Organize a web page so that it is still readable and understandable if the style sheet is removed.

Examples:

* When a document is rendered without associated style sheet, it must still be possible to read the document.
* Provide a text equivalent for any important image or text generated by style sheets (e.g., via the 'background-image', 'list-style' or 'content' properties).

Note: Web Content Accessibility Guidelines (WCAG) 3.3 recommends using style sheets to control layout and presentation.  This method is strongly preferred over the use of tables due to wider compatibility with end user devices

1. [**Tables, Simple**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard7.aspx) – If a web page uses simple tables to display information, use tags to clearly identify row and column headers.

Examples:

* Tables used only for layout do not have header rows or columns.
* In data tables, column and row headers are identified using the <*th*> tag.

1. [**Tables, Complex**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard8.aspx) – If a web page uses complex tables (i.e., two or more levels of row or column headers), use id and header attributes to clearly identify information.

Example:

* Table cells are associated with the appropriate headers (e.g., id, headers, scope and/or axis HTML attributes).

1. [**Testing**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard17.aspx) – Plan for regular usability and accessibility testing of your site.
2. **Text Equivalent** – A text equivalent for every non-text element shall be provided (e.g., via “*alt*”, “*longdesc*” or in element content).

Examples:

* Non-text elements (images, Java applets, flash files, video files, audio files, plug-ins, etc.) have alt tag descriptions that convey the purpose or intended meaning of the object (e.g., alt Tags for images used as links describe the link destination).
* Complex graphics that summarize information (graphs, charts, tables, etc.) are accompanied by text conveying the information providing a meaningful narrative of the information.
* Decorative graphics with no other function have empty alt descriptions (*alt*= "") not missing alt descriptions.
* When descriptions are lengthy or refer to other resources or sites, a longer description is available using a link or supported "*longdesc*".

1. [**Text-Only Pages**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard11.aspx) – If a web page cannot comply with accessibility guidelines in any other way, provide a text-only page with equivalent information or functionality.

Examples:

* A mirror page is acceptable when there is no other way to make the content accessible or when it offers significant advantages for ease of navigation.
* Update the content for primary and mirror pages simultaneously.  For example, use a common database to generate content for multiple versions of the site.
* Instead of static alternative pages, set up server-side scripts that generate accessible versions of a page on demand.
* Mirror pages must be the functional equivalent to primary pages (e.g., provides alternatives for applets, scripts, plug-ins and similar applications that are not directly accessible).
* "Text-only" and "accessible" are not synonymous. Designers must incorporate all of the above standards when designing mirror pages.

1. [**Timed Responses**](http://www2.dir.state.tx.us/management/accessibility/standards/Pages/WebAccessibilityStandard16.aspx) – If a web page requires a timed response (for a timed test or a secure web application), alert the user and provide a way to request more time.

Example:

* Do not automatically forward, refresh, or otherwise alter pages, unless you provide the user with a method to adjust the timing of these content changes.
  1. Other recommended best practices include:
  2. **Audio turnoff** – If any audio plays automatically for more than 3 seconds, provide either a mechanism to pause or stop the audio, or a mechanism to control audio volume that can be set independently of the system volume.
  3. **Color Contrast –** Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on black and white screen.

Example:

* Blue on blue, using different saturations of the same color for background and foreground
  1. **Dynamic Content** – Update equivalents for dynamic content when the dynamic content changes.
  2. **Link Purpose (context)** – On web pages, it must be possible to determine the purpose of each link from the link text or the link text together with its programmatically determined link context.
  3. **Meaningful Sequence** – When the sequence in which the content is presented affects its meaning, a correct reading sequence must be programmatically determined. The sequential navigation of interactive components must be considered with that reading sequence.
  4. **Parsing** – Content implemented using markup languages must have elements with complete start and end tags, except allowed by their specifications, and must be nested according to their specifications.
  5. **Size, Shape, Location** – Instructions provided for understanding and operating content must not rely on shape, size, visual location or orientation of components.
  6. A detail discussion on the above topics is available at

<http://www.dir.texas.gov/management/accessibility/pages/overview.aspx>.

1. SPECIAL SOFTWARE AND OPERATING SYTEM STANDARDS

When an application uses special vendor software (or operating system) that is not industry standard, but that needs to be installed on the client system, then the software shall meet the 508 Compliance.

1. USABILITY STANDARDS

Recommended usability standards based on W3C guidelines are listed below with a brief description:

**Blinking** –Design web pages so that content does not blink for more than three seconds, or provide a method to stop all blinking content in the web page.

**Bypass Option** – Provide a mechanism that permits users to bypass blocks of content or navigation links repeated on multiple web pages.

**Consistent Identification** – Identifycomponents that have the same functionality within a set of web pages consistently.

**Consistent Navigation** –Navigational mechanisms repeated on multiple web pages within a set of web pages shall occur in the same relative order each time they are repeated, unless a change is initiated by the user.

**Error Identification** –If an input error is automatically detected, identify the item that is in error and describe it to the user in text.

**Focus Order** –If a web page can be navigated sequentially, focusable components shall receive focus in an order that follows information and relationships conveyed through presentation.

**Language of Page** – Changes in the natural language of a document's text and any text equivalents (e.g., captions) must be clearly identified. For example, in HTML use the "lang" attribute; in XML, use "xml:lang".

**On Focus** –When a component receives focus it must not initiate a change of context.

**On Input** – Changing the setting of any user interface component must not automatically cause a change of context unless the user is advised of the behavior before using the component.

**Page Title** – Web pages must have descriptive titles.

**Pausing** – Moving, blinking, scrolling or auto-updating information must allow pausing by the user unless it is part of an activity where timing or movement is essential. Content that is pure decoration must at least allow stopping by the user.

**Three Flashes or Less** – Web pages must not contain anything that flashes more than three times in any one second period (i.e., has a frequency greater than 2 Hz and lower than 55 Hz), or the flashing is below the general flash and red flash thresholds.

**Timing** – For each time limit that is set by the content, at least one of the following must be true:

* Turn off – Allow the user to turn off the time limit before encountering it.
* Adjust – Allow the user to adjust the time limit before encountering it over a wide range that is at least ten times the length of the default setting.
* Extend – Warn the user before time expires, give at least 20 seconds to extend the time limit with a simple action (for example, "hit any key"), and allow the user to extend the time limit at least ten times.
* Real-time Exception – The time limit must be a required part of a real-time event (for example, an auction), and no alternative to the time limit is possible.
* Essential Exception – The time limit must be part of an activity where timing is essential (for example, time-based testing) and time limits cannot be extended further without invalidating the activity.

1. HELPFUL RESOURCES FOR DEVELOPERS
   1. Training materials – can be found in <http://training.section508.gov/index.cfm?fuseAction=Courses>. Some recommended training courses are:

* Designing Accessible Web Sites
* Accessible Conferences
* Accessible Video and Multimedia
* Building and Buying Accessible Software
  1. Firefox Toolbar and Add-ons:
* Web developer toolbar

<https://addons.mozilla.org/en-US/firefox/addon/toggle-web-developer-toolbar/?src=ss>

* HTML Validator is a Mozilla extension that adds HTML validation inside Firefox and Mozilla. The number of errors of a HTML page is seen on the form of an icon in the status bar when browsing. The extension can validate the HTML sent by the server or the HTML in the memory (after AJAX execution).

<https://addons.mozilla.org/en-US/firefox/addon/html-validator/?src=ss>

* Total Validator performs multiple validations and takes screen shots. This 5-in-1 validator works with external, internal or local web pages using the Total Validator service or local copy of the desktop tool, will email a report if opted, will validate to WCAG 1, 2 or 3, and has a help link.

<https://addons.mozilla.org/en-US/firefox/addon/total-validator/?src=search>

* 1. Internet Explorer Toolbar:
* AIS Web Accessibility toolbar has been developed to aid manual examination of web pages for a variety of aspects of accessibility.

<http://www.visionaustralia.org/digital-access-wat>

* 1. Accessibility Testing Software :
* <http://jimthatcher.com/testing2.htm>
* <http://www.w3.org/WAI/ER/tools/>
  1. Forums:
* USA.gov – <https://www.usa.gov/>
* Web Content Managers Forum – <https://digital.gov/communities/web-content-managers/>
* Refresh Austin – <http://www.refreshaustin.org>
* National Association of Government Webmasters – <http://www.nagw.org>

CONTACT: [Information Resources Department Head](mailto:cbryan@tfs.tamu.edu), 979/458-6609