Functional Web Accessibility Techniques and Tools from the University of Illinois

Jon Gunderson, Ph.D.
University of Illinois at Urbana/Champaign
1207 S. Oak Street
Champaign, IL 61820
(217) 244-5870
jongund@uiuc.edu

Hadi Bargi Rangin, M.S.
University of Illinois at Urbana/Champaign
1207 S. Oak Street
Champaign, IL 61820
(217) 244-0518
hadi@uiuc.edu

Nicholas Hoyt
University of Illinois at Urbana/Champaign
1207 S. Oak Street
Champaign, IL 61820
(217) 244-0518
nhoyt@uiuc.edu

ABSTRACT
For web developers to create functionally accessible web resources they need more than general guidelines and tools that provide them with lists of manual accessibility checks. Web developers need specific web accessibility techniques and tools that help them verify they have correctly implemented the techniques. The techniques also need to support the wider concepts of the web of interoperability and device independence. The CITES/DRES Functional Web Accessibility Best Practices provide developers with specific techniques and requirements to implement Section 508 and W3C WCAG 1.0 requirements. The use of the Functional Web Accessibility Evaluation (FAE) Tool and the Mozilla/Firefox accessibility extension provide free and open source tools to allow developers to verify they have used the best practices.

Categories and Subject Descriptors
Design, Human Factors, Standardization, Legal Aspects

General Terms
Web, Design, Accessibility, Tools, Evaluation

Keywords
Web, Accessibility, Evaluation, Mozilla, Firefox, Best Practices, Tools, Dynamic HTML, Scripting and Techniques

1. CITES/DRES Best Practices
The Section 508 [1] and W3C WCAG 1.0 [2] provide principles for accessible web design, but do not have any specific markup implementation requirements. The CITES/DRES Functional Accessibility Best Practices [3] provide specific markup requirements for common html web page and web site design. The best practices are organized based on the principles of Navigation and Orientation, Text Equivalents, Scripting and Automation, Styling and Standards.

1.1 Navigation and Orientation
Navigation and orientation information relates of using the structural markup capabilities of HTML to uniquely title web resources, assign header markup to section titles, defining navigation bars, labeling form controls and in general using html to indicate the semantics of the web resource. Navigation and orientation is an area overlooked by many web developers, since most current web evaluation tools [4] report these requirements as manual checks. Since manual checks are a labor intensive process these requirements are often overlooked by most developers.

1.2 Text Equivalents
Text equivalents are need to provide text representations of audio, video and graphic content in web resources. Ideally the text descriptions represent the same information as the non-text content. Text equivalents are the poster child of accessibility since missing ALT attributes on IMG elements are very easy for automated tools to detect.

1.3 Scripting and Automation
Scripting and automation are becoming more a part of how web resources are being developed and a whole new generation of web applications are based upon scripting and XML technologies. There are accessibility issues related to device independence and generated content that are critical for accessibility.

1.4 Styling
The use of Cascading Style Sheets (CSS) to separate styling from structural content is an important principle of the web and even more important to people with disabilities to be able to restyle content to meet their own perceptual needs. The use of CSS makes it easier for web resources to adapt to a wide range of technologies and user styling preferences.

1.5 Standards
The use of HTML, xhtml, CSS, Document Object Model (DOM) and other W3C Standards allows the widest range of web technologies to access web resources. People with disabilities use a wider range of technologies than the general public so the use of web standards improves the reliability of their access to information.

2. Functional Accessibility Evaluator
The Functional Accessibility Evaluation (FAE) Tool [5] provides a means to estimate the functional accessibility of web resources by analyzing web pages and estimating their use of the CITES/DRES web accessibility best practices. The tool does not determine if a resource or a collection of resources is accessible or not, but provide summary and detailed reports on the use of accessible markup categorized by the best practices principles. FAE uses rules for testing each of the functional accessibility features of navigation, text descriptions, styling, scripting and the use of standards. The test results are linked to the CITES/DRES

Copyright is held by the author/owner(s).
ASSETS'06, October 22–25, 2006, Portland, Oregon, USA.
ACM 1-59593-290-9/06/0010.

best practices resources [3] for web developers to find out more information about the evaluation results. FAE is being extended to test for DHTML accessibility features.

FAE works similar to other web accessibility evaluation tools. Users go to the website and enter a URL and the depth of checking of the web site. Users then request the resources be checked for accessibility based on the coding techniques outlines in the CITES/DRES web accessibility best practices. This is where FAE differs from current evaluation tools. Other tools do tag matching to determine known accessibility problems, like missing ALT text from an IMG element, and report missing ALT attributes as a known accessibility problem. For other accessibility issues, like proper use of headers (H1-H6), current tools tell the user that they need to perform a manual evaluation. The number of manual checks is based on the types of tags found in the resource. There is only a small set of accessibility problems that can absolutely identified in this tag matching approach and the reports usually require between 20 – 30 manual checks. Manual checks are quite tedious and therefore ignored by many web developers due to limitations in time or understanding of the requirements. Since FAE is looking for best practices, items like missing headers or resources not being properly titled can be reported as known problems, not as part of some list of manual checks. Developers are motivated to eliminate known accessibility problems so they can report their resources are highly accessible. The power of FAE therefore is automating these manual checks reported by current evaluation tools and therefore encouraging web developers to use more accessible web design techniques. FAE is free service of the University of Illinois and anyone can request an evaluation using the web interface. FAE currently only checks HTML based resources, but other formats may be supported in the future.

3. Mozilla/Firefox Accessibility Extension

Web browsers can play a critical role in testing web accessibility if they can highlight the accessibility features of a web resource to developers. The Mozilla/Firefox Accessibility Extension [6] provides navigation, styling and conditional rendering features that are important in improving access to web content for people with disabilities and evaluating web resources for functional accessibility by developers. The features are based on the W3C User Agent Accessibility Guidelines [7]. The value of the Mozilla/Firefox Accessibility Extension is the ability to make information that is hidden in a graphical rendering of content visible to developers and people with disabilities. For example, when developers use of headers (h1-h6) or use of labels for form controls the graphical rendering typically does not disclose this information. The accessibility extension provides information on headers, labels and many other types of structural information by querying the Document Object Model (DOM) of the resource and extracting structural information and displaying it in dialog boxes or by providing keyboard navigation commands. People with disabilities and developers can then use this information to access and functionally test the structural markup of web resources. Other features include the ability to test for the inclusion and functional use of text equivalents for non-text content like images, audio and video. The extension provides the ability to disable author supplied CSS styling, in-line tag styling and tables used for layout. Users can apply user style sheets and includes two built-in options for high contrast style sheets. The extension also implements features to support the new Dynamic HTML accessibility features [8] being developed by the W3C Protocols and Formats group.

4. Dynamic Web Application Evaluation

Dynamic HTML Web Applications cannot be evaluated with traditional web based accessibility evaluation tools. Current tools rely on the state of the web resource to be defined as part of the URL. The state of web resources in web applications is defined on a server and communication of state information is handled through behind the scenes transactions with the server. FAE is being improved to provide a web services interface to allow HTML markup to be sent to FAE for evaluation and to generate page and site wide reports based on markup sent by other services. The Mozilla/Frefox accessibility extension will be the first tool to use the web services interface to help automate the generation of accessibility reports related to tasks within a web application. The Mozilla/Firefox extension can take code samples from the Mozilla DOM and send the code to FAE for the evaluation and the generation of reports. The user therefore only needs to move through a web application and at each step tell the toolbar to add the current page of the web application to the FAE report. This greatly reduces the time and effort needed to generate accessibility reports and allows developers to replicate the reports sent to them.

5. Conclusions

For functional accessibility to be achieved more automated tools need to be available to help web developers to review and redesign their web resources. Without automation most web developers will only be checking for a limited scope of web accessibility requirements. Automated tools cannot solve all accessibility problems, and usability testing with people with disabilities is a critical aspect of accessible web design. But tools like FAE and the Mozilla/Firefox Accessibility Extension allow evaluation resources to be devoted to usability issues, rather than the more basic requirements of having accessible markup.

6. REFERENCES

[1] Section 508 Information Technology Accessibility Standards http://www.access-board.gov/sec508/standards.htm

[2] W3C Web Content Accessibility Guidelines http://www.w3.org/TR/WCAG

[3] DRES/CITES Web Accessibility Best Practices http://html.cita.uiuc.edu

[4] Web Accessibility Evaluation Tools http://www.w3.org/WAI/ER/tools/

[5] Functional Accessibility Evaluator http://fae.cita.uiuc.edu

[6] Mozilla/Firefox Accessibility Extension http://firefox.cita.uiuc.edu

[7] W3C User Agent Accessibility Guidelines http://www.w3.org/TR/UAAG/

[8] W3C DHTML Accessibility Roadmap http://www.w3.org/WAI/PF/roadmap