The Evolution of a Web Accessibility Testing Protocol

Tom Jewett and Wayne Dick

Department of Computer Engineering and Computer Science, California State University, Long Beach
{jewett, wed}@csulb.edu

Abstract. In early 2007, the California State University (CSU) system initiated a program to make all information technology accessible to persons with disabilities; the authors were appointed to lead the Web portion of this effort. Campuses initially hoped to rely on automated testing of Web sites to evaluate their level of accessibility and determine what repairs would be needed. However, we found that automated testing tools were severely limited both in their ability to identify relevant problems and, surprisingly, in their scalability to sites of CSU magnitude. The first manual (non-automated, human intelligence based) testing protocols that we developed proved to be awkward in practice; this paper reports our substantial recent refinements after extensive use and incorporation of the newest accessibility guidelines. We also suggest how this protocol can become part of the Web development workflow, rather than being used solely to check legacy sites.

Keywords: Web accessibility, manual evaluation, testing, Section 508, WCAG.

1 Background

In early 2007, the Chancellor of the 23-campus California State University (CSU) system issued a directive to the campus Presidents that all CSU information technology (IT) resources must be made accessible to persons with disabilities. This effort, called the Accessible Technology Initiative (ATI), was divided into three priority areas: administrative Web sites, instructional materials, and procurement of IT. [1] Of these, the Web was given the most aggressive schedule, with substantial milestones to be realized within one year and all work to be completed by 2012. The Chancellor's directive also specified that success was to be measured by compliance with "Section 508," [2] a part of the United States Rehabilitation Act of 1973 as amended in 1998 to include guidelines for accessibility of computer software and Web sites.

2 Getting Started

The initial ATI management cadre consisted of only two staff members experienced in disabled student services and the second author of this paper. The first author was later added to lead the Web priority area on a part-time basis.

C. Stephanidis (Ed.): Universal Access in HCI, Part III, HCII 2009, LNCS 5616, pp. 701–706, 2009.

A few campuses had established Web accessibility programs in advance of the ATI, relying on one of the leading automatic testing tools to assess compliance with Section 508. Campus Chief Information Officers across the system agreed to adopt the same tool, which was then licensed for use by all campuses. In addition, the ATI Director posted a large number of "508 checklists" and guidelines for the campuses to follow; these came from sources such as U.S. government agencies and universities nationwide.

3 An Unpleasant Surprise

One of the campuses had been particularly aggressive in their automated testing program, and was reporting substantial success in the compliance of their web pages. Naturally, we visited that campus to see what we could learn. After discussing their procedures and reports, we looked at one of the pages that had been highly ranked by the automatic testing software. What we saw was a page that had somehow passed the tests—but when we removed the images (as a voice reader would do), most of the content simply disappeared. Any reader with a visual disability could not have used the page. Additional testing on the same campus and many other campuses revealed that the same types of problems were in fact widespread throughout the system.

The problem was that it is easy for non-technical administrators – and even for many Web developers – to become mesmerized by the automated tool's red-and-green summary charts into believing that a simple yes-or-no answer can characterize the accessibility of any given page or site. It is also easy to overlook the automated tool's own detailed reports which clearly list many aspects of accessibility that cannot be checked by software and must be evaluated further by human specialists. These observations hold true for *any* automated accessibility testing package, not just for the specific one that was used by CSU campuses.

One of the major hopes of automated accessibility testing advocates was that these tools would scale easily to evaluate sites consisting of tens of thousands – even hundreds of thousands – of pages. It was therefore also surprising and perhaps ironic that this did not turn out to be the case. Part of the problem was the complexity of the campuses' Web presence – often distributed over many heterogeneous servers and subject to the control of many different organizational units. More importantly, the limitations of the automatic test reports simply could not be resolved in these quantities, given the campuses' severely constrained personnel and financial resources.

4 Next Steps

Clearly, testing in addition to the automated system would be required. Each campus was to report on the accessibility status of key Web sites by the end of the first year of the ATI program; all were requesting guidance on how this was to be done. Since the previously-posted "508 checklists" had created more confusion than clarity, we decided that a custom-developed "manual evaluation" protocol would be in order.

In the meantime, we had assembled key Web developers from each of the campuses to form a Web Community of Practice or "WebCoP," which met regularly by

conference call. These meetings served both to disseminate information and to provide feedback and guidance for issues such as the manual evaluation.

In developing our approach to evaluation, we started with existing checklists and input from "WebCoP" members, but relied most heavily on the then-newly-released book, *Web Accessibility*, by Jim Thatcher and colleagues. [3] This comprehensive reference also became "required reading" for campus developers as they learned to evaluate their Web sites.

The protocol itself evolved in two stages:

- First was a series of steps using an assortment of tools such as browser add-ins; this proved to be confusing to some evaluators since it did not follow the structure of Section 508.
- The protocol was then expanded, keeping the browser add-in tools but ordering the steps to match each paragraph of the Section 508 guidelines and adding specific questions to be answered – along with a detailed reference guide – for each step.

The second version has remained in use by most campuses, although it has perhaps been more valuable as a training aid than as a production tool. As a result of feedback from the campuses and from our own use (discussed below) we identified a number of problems with it:

- There were too many questions and too much detail for easy use after an evaluator had mastered the concepts involved.
- For the same reasons, it proved extremely difficult to document the findings from each checkpoint and to update the protocol to accommodate newer standards and our own evolving understanding of accessible development techniques.
- Our attempt to accommodate evaluators who wanted to use a variety of browsers and a variety of add-ins increased the complexity of the protocol for everyone.
- Finally, sequencing the tests in Section 508's arbitrary checkpoint order – as well-meaning as it may have been – made the protocol much more time-consuming and redundant that it needed to be.

5 Another Challenge

In addition to administrative Web sites, each campus also uses a Web-based learning management system (LMS) for posting course materials, instructor-student communication, and other instructional tasks. These are large-scale, complex systems that are far beyond the ability of any campus to maintain, modify or even to evaluate for accessibility compliance. Therefore, the Chancellor's Office issued a competitive request for proposal, seeking one or more LMS vendors who could provide an accessible product for system-wide licensing.

The vendor proposals were narrowed down by Chancellor's Office procurement staff to five finalists, based on functional requirements and other criteria. The authors were then called in to evaluate the accessibility of those five systems, all of which were existing products (four commercial, one open-source).

5.1 Simplifying and Customizing the Protocol

Time and cost constraints dictated a simple standards-based approach, using a shorter version of our previously-developed manual evaluation protocol. We selected seven steps or questions to ask of each page; these were generalized to cover key areas such as text enlargement, page structure, images, and so on. In each step, we were looking for major barriers that would be encountered by readers with a variety of disabilities including visual, motor skills, and more. Already familiar with browser add-ins and other evaluation aids, we did not need to write these specifically into this version of the protocol.

We then selected 11 pages of each LMS to evaluate; these represented typical functions that a user would perform, including login, home page navigation, discussion board, grade book, and so on. The simplified protocol was thus run 110 times; 5 systems * 11 pages * 2 evaluators. We also developed a fairly complex weighted numerical scoring system which was independent of the protocol itself (although it did add time to our evaluation). For our report on each LMS, we provided an overall evaluation and a separate professional judgment score in addition to the page-by-page results. An executive overview and a detailed description of our methodology completed the package that we delivered to the procurement team.

5.2 Justifying and Demonstrating the Protocol

Of the five candidate systems, we found one to be generally accessible, with a few possible improvements that could be made to help persons with certain disabilities. Another one of the systems was somewhat accessible, especially to experienced users of assistive technology who could "work around" the problems they encountered. The remaining three systems, unfortunately, were profoundly inaccessible even to the most skilled users of assistive technology.

This is not what anyone wanted to hear – especially not those campuses that had already made major investments in any of the inaccessible systems, let alone the vendors of those systems.

No amount of documentation would have been sufficient to answer questions and resolve doubts about our results. We were thus called upon to present out findings at four separate meetings of campus academic technology and information technology directors and in one telephone-plus-computer briefing to a disappointed vendor. Our presentations consisted of only a brief summary of the methodology, followed by an extensive demonstration of the barriers we had encountered, showing a selection of the exact protocol steps and evaluation tools that we had used. Although these sessions proved to be an important learning experience for our audiences, they also provided us with important validation in at least two respects:

- Even a simple standards-based evaluation protocol proved to be robust in its ability to identify actual barriers that would be encountered by persons with disabilities using assistive technology.
- More importantly, those barriers were effectively demonstrable to audiences whose members were neither accessibility specialists nor in many cases even familiar with the types of assistive technology that might be used by persons with disabilities. For these audiences, demonstrated barriers were also easier to understand than legalistic Section 508-based explanations.

6 Subsequent Development

Reflecting on both the campus ATI activities and the LMS evaluation, we are convinced that automated tools and manual protocols must be used to compliment each other, acknowledging the strengths and weakness of each. Our own work has focused on the manual evaluation component, which more accurately could be described as human intelligence based analysis. From the experiences described above, we realized that an evaluation protocol could be either too complex (as was our second try for the CSU system) or too abbreviated (as was our personal-use protocol for the LMS evaluation). Further work was needed.

6.1 Setting Goals

Even without doing a formal requirement analysis as we would for software development, it was clear that our testing protocol should meet a number of goals:

- It should readily identify the most important barriers to use of a Web page by persons with a wide range of disabilities. These barriers should be demonstrable to a wide range of audiences, as we discovered in the LMS project.
- It should be useable by developers with a wide range of experience, but does not need to function (in fact, should not function) as a complete accessibility tutorial.
- It should be consistent with both legal standards (such as Section 508) and best development practices, but again does not need to provide a tutorial on these topics.
- It should compliment, not duplicate, the use of automatic testing tools – effectively filling in the information that cannot be provided by those systems.
- It should be adaptable both to various organizational requirements and to rapid changes in Web technology and standards.
- It should be designed for use every day in new Web site development, not just for occasional checking of legacy sites. (In fact, most campuses have realized that prioritized replacement of sites is far more cost-effective than repair of old, inaccessible ones.)

We were fortunate that as we refined our evaluation methods, the World-Wide Web Consortium (W3C) finalized their newest accessibility recommendation, the Web Content Accessibility Guidelines (WCAG) 2.0. [4] We mapped the majority of WCAG 2.0 success criteria [5] to Section 508 guidelines, and included in our protocol the few additional ones that are needed to meet their "A" level of accessibility.

6.2 Current Status

For most of our own work, and for the majority of developers whom we train, we have found that the most useful and effective protocol employs one browser (Firefox), one add-in toolbar (the Firefox Accessibility Extension, developed at the University of Illinois [6]), and 13 simple tests.

Tests are ordered for efficient workflow while using the toolbar. Each one is formatted with its topic, a brief description of what to look for, a simple reminder of how the test is performed, and another reminder of which legal requirements are being evaluated. Here is an example of one test in its current form:

- **Headings:** Headings should match the actual semantic structure of the document and should be properly nested by level. Headings should also be used to identify and navigate between groups of related links. Proper use of headings is now more accessible than adding "skip navigation" links.

– How: Navigation -> Headings; Navigation -> Menu and Navigation Bars.
– Addresses Section 508 paragraphs: (d), (o).

The 13 tests, taken together, are designed to cover all of the Section 508 criteria plus a few additional best practices from WCAG 2.0. However, we have also realized that a subset of only five tests (visual check, headings, links, images, and styles) can often identify the most significant barriers on a page. Since these can be performed in a matter of seconds, we refer to them (only slightly jokingly) as a "stinker test" – until a page passes at least this much, further development and testing would not be productive.

7 Conclusions

Based on our work with the California State University campuses, our experience in the Learning Management System procurement process, and our concurrent teaching and lecturing activities, we believe that a manual accessibility evaluation protocol should be part of any Web development program. We have traced the evolution of one such protocol over a period of two years, but realize that it can never be a static, "finished" document. We welcome comments on the online current version [7], and hope that others will be able to adapt and improve it to meet their own needs.

References

1. Accessible Technology Initiative, http://www.calstate.edu/accessibility
2. Section 508: the Road to Accessibility, http://www.section508.gov
3. Thatcher, J., et al.: Web Accessibility: Web Standards and Regulatory Compliance. Friendsofed, Berkeley (2006)
4. Web Content Accessibility Guidelines (WCAG) 2.0, http://www.w3.org/TR/WCAG20
5. How to Meet WCAG 2.0, http://www.w3.org/WAI/WCAG20/quickref
6. Firefox Accessibility Extension, http://firefox.cita.uiuc.edu
7. Smart Analysis Simplified, http://www.theenabledweb.com/smart-analysis.html