

The Usability of Accessibility Evaluation Tools

Helen Petrie¹, Neil King², Carlos Velasco³, Henrike Gappa³, and Gaby Nordbrock³

¹ Department of Computer Science, University of York, Heslington,
York YO10 5DD United Kingdom,

² DesignedforAll, Prince Consort House, 109 - 11Farringdon Road, London EC1R 3BW
United Kingdom

³ Fraunhofer-Institut für Angewandte Informationstechnik (FIT)
Schloss Birlinghoven 53754 Sankt Augustin Germany
Helen.Petrie@cs.york.ac.uk, Neil@DesignforAll.com,
carlos.velasco@fit.fraunhofer.de

Abstract. A range of accessibility evaluation tools are available to support Web developers in producing accessible Web resources. It is important that these tools support web developers very well, so that the developers can concentrate on the tasks of Web development. In addition, the tools should enhance Web developers understanding of accessibility issues. This paper presents a usability evaluation of five entry level accessibility evaluations tools. A group heuristic evaluation was conducted, with 5 experts in usability and accessibility working through each tool together, but rating usability problems separately. The results showed that the usability of these tools is surprisingly poor and that they do not support web developers adequately in checking the accessibility of their web resources.

Keywords: Web accessibility, accessibility evaluation, accessibility evaluation tools, usability.

1 Introduction

Testing the accessibility of a website is an important component of its overall development. Testing should include both checking whether the site conforms to the Web Content Accessibility Guidelines (WCAG) [8] and evaluation with disabled people to ensure that they can effectively use the site with different assistive technologies [9].

There are over 100 different tools to assist web developers in testing the conformance of their sites to WCAG [10]. At the most specific level, these include tools which provide information to a developer about a particular accessibility success criterion, but may still require human judgement to decide whether a web resource conforms to WCAG on this success criterion. For example, a tool might display a resource as it would appear to individuals with different colour vision deficiencies; then the developer must check whether the colours used in the resource convey information that is not conveyed by other means (and hence violate WCAG2 Success Criterion (SC) 1.3.2 “Any information that is conveyed by colour is also visually

evident without colour”). Such a tool is the Accessibility Colour Wheel [4]¹. At the most general level, a tool might conduct a range of automatic tests on a website and inform a developer whether the site conforms to WCAG on success criteria related to these tests. Such a tool is WebXACT from Watchfire [7].

It has been argued [1, 2] that the usability of accessibility evaluation tools should not be considered until the functionality and scope of these tools is clear. Although we agree there are serious questions about the functionality issues, we argue that usability issues always need to be considered from the outset of the development of any interactive system. Studies in human-computer interaction have repeatedly shown that if usability is not considered from the outset, it is extremely difficult to retro-fit and a system that is difficult and frustrating to use is likely to occur [6]. It is important that accessibility evaluation tools are very easy for web developers to use, so that the developers can concentrate on the tasks of Web development and not on how to use the tool and interpret its results.

The accessibility evaluation tools also have an important role in clarifying accessibility issues to developers. Many web developers are still relatively inexperienced about accessibility issues [3], so the more the tools can illustrate how the accessibility evaluation works, the more they will be able to learn in an incidental manner as they interact with the tool. In particular, there are issues that the tools may be conducting a test related to a particular WCAG Checkpoint or SC, but this is not a complete test of the checkpoint or SC. For example, consider Checkpoint 1.1 of WCAG1, “Provide text alternatives for all non-text content”. A particular tool might test this by checking whether each IMG element has an ALT attribute. This is a good initial test, but the fact that a particular IMG element passes this test does *not* mean that it has passed Checkpoint 1.1. It might have a null ALT (ALT = “”) or a meaningless or incorrect description in the ALT attribute (we have found descriptions such as “blah blah blah” and an image of a cat that had a description saying it was an elephant). The tool should make clear exactly what tests it is making, how comprehensive they are and whether the Checkpoint of SC is completely covered by the tests.

To start to investigate these issues, an evaluation of the usability of a range of “entry level” accessibility evaluation tools was undertaken; tools that a novice web developer, who needs a highly usable and informative tool might well use.

2 Method

A group expert evaluation was conducted on five accessibility evaluation tools. The experts worked as a group, spending a maximum of 2 hours using the tool, exploring all the possible functionality. Any expert was allowed to propose a usability problem about the tool, but then all the experts privately noted whether a proposed problem was indeed a problem, and if they thought it was a problem, rated its severity. Thus different experts might identify different numbers of problems for a particular tool, and rate them differently.

¹ Examples of products, commercial or otherwise, are provided only as illustrative of particular points, and are not endorsements of these products.

2.1 Participants

Five experts in both the usability and accessibility fields participated. 2 were female, 3 were male. Ages ranged from 31 to 49. Years of experience in the usability field ranged from 2 to 15 years. Years of experience in the accessibility field ranged from 1 to 13 years.

2.2 Tools Evaluated

The evaluations were conducted in July 2005. Tools may have changed since then. The following information describes the tools when they were tested.

Five evaluation tools were evaluated:

- Site Valet (<http://valet.webthing.com/>)
- WAVE 3.0 (<http://www.wave.webaim.org/>)
- Cynthia Says (<http://www.cynthiasays.com>)
- Bobby 5.2 (www.watchfire.com/default.aspx)
- Deque Ramp (<http://www.deque.com/products/ramp/index.php>)

2.2.1 Site Valet by WebThing

The online service of Site Valet offered by WebThing evaluates a single web resource. The service claims to check the page according to WCAG2 (this seems

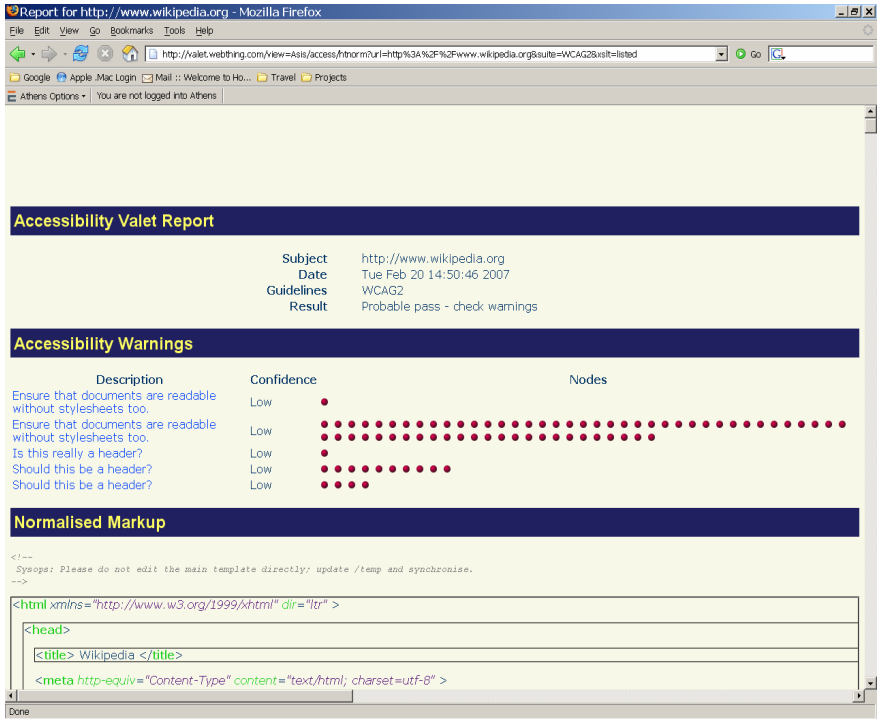


Fig. 1. Site Valet accessibility reporting page

extremely unlikely, although it has no bearing on the results of the present evaluation). It is not possible to customize the coverage of the accessibility check, so a page cannot be inspected for conformance to other accessibility standards such as WCAG 1.0, Section 508 or particular priority levels of these guidelines. No information about the scope of the service, terms of use or other such information is provided. The service is free.

The Site Valet accessibility reporting page is divided into three parts (see Figure 1). The first provides a summary of what standards have been checked, the URL of the Web page evaluated and a result in terms of “pass” or “fail”. The second part displays all relevant accessibility warnings with a description, for example “table cells should be associated with headers”, a column indicating the level of confidence about this warning (high, medium, or low) and the instances of this accessibility warning indicated by a red circle under the header “node”. The nodes are directly linked to the occurrence in the markup which is displayed in the third part of the report.

2.2.2 Wave 3.0 by WebAIM

The WAVE Accessibility Tool Online Web Service was originally developed at Temple University Institute on Disabilities in Pennsylvania and has now been taken over by WebAIM (Web Accessibility in Mind) at the Center for Persons with Disabilities (CPD, Utah State University).

The tool is free of charge and is intended for personal, non-commercial use. The terms of use state that the tool is offered as an “as is” product, meaning no warranty is provided and that compliance to a certain Web accessibility standard cannot be ensured by usage of the service. A navigation link leading to the terms of use is displayed clearly next to the submit button for an accessibility evaluation on the home page.

WAVE evaluates a single Web resource. The accessibility reporting page displays the tested Web resource with icons indicating problems on the resource in the following colours (see Figure 2):

- red icons for errors that need repair
- yellow icons for alerts that need to be checked for possible errors
- green icons for accessibility features that need to be checked for accuracy
- blue icons for structural and semantic elements that may support accessibility and should be checked for accuracy

The meaning of the icons is explained in a separate document which can be opened from the accessibility reporting page in a new window. In this document, the relevance of each icon in relation to WCAG 1.0 and Section 508 is explained.

The reporting page does not provide an evaluation result in terms of “pass” or “fail”. WAVE allows the user to customize the accessibility check for WCAG1.0 Priorities 1, 2 and 3, and Section 508, the elements to be displayed in the report (e.g. tables, images, frames) and special WAVE features such as the implications of linearised reading order by an arrow to the right, borders on all tables, etc.

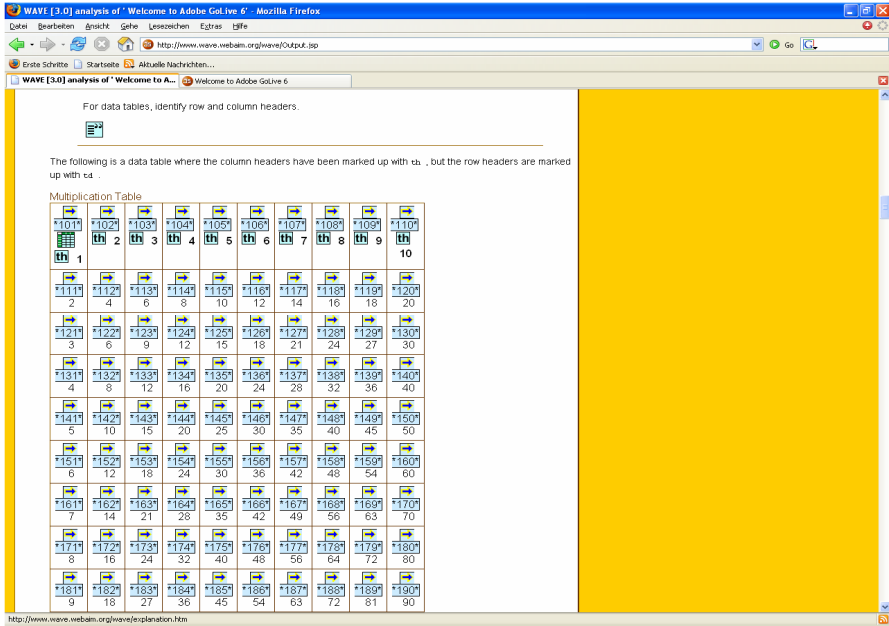


Fig. 2. Wave 3.0 accessibility reporting page

2.2.3 Cynthia Says by HiSoftware

Cynthia Says is a single web resource evaluation tool from HiSoftware. As the company states, Cynthia Says is designed to offer a service for all users who wish to create accessible websites, including those with little or no knowledge. The section “about this site” explains that Cynthia Says evaluates a web resource against tests for Section 508 and WCAG1 by employing HiSoftware’s AccMonitor Server technology and conducting accessibility verification on its central server with the result being sent back to the user’s browser. This service is offered free of charge.

HiSoftware states in its terms of use that Cynthia Says shall only be used for personal, non-commercial purposes. It is offered on an “as is” basis, meaning that no guarantee for accuracy of the results is given. It is also stated that the test results need to be reviewed by a human and that manual checks as listed in the accessibility report need to be performed.

On the accessibility reporting page, the resource that has been evaluated is indicated and whether it has passed or failed automated verification. Results are presented in the form of the tabular summary of WCAG1 checkpoints. Below each checkpoint the tested that have been evaluated are listed with the result returned by Cynthia Says: “Rule: 1.1.2 - All INPUT elements are required to contain the alt attribute or use a LABEL” with the result “No invalid INPUT elements found in document”. In the column next to this statement, fulfillment of this checkpoint is rated. This can be a “Yes”, “No”, “Not Verified” (N/V), “No related elements were found in the document” (N/A) or a “Warning”. Cynthia Says also provides the line number in the source code of the tested page where a problem has been detected.

2.2.4 Bobby 5.2 by Watchfire

The Bobby 5.2 is the desktop version of the well-known Bobby tool and is designed to conduct accessibility audits encompassing an entire website. It is a commercial product that aims to identify barriers to accessibility in accordance with WCAG1 and Section 508 guidelines. It spiders through a website page by page conducting 90 accessibility checks per page. Several types of accessibility reports are offered. There is a summary statement that gives an overview of all pages reviewed with the status (pass/fail) and the number of accessibility errors and possible accessibility issues. It offers the possibility to link to a particular page to access an accessibility report for that page. The individual page report follows the same outline as the summary report. The only difference is that here for some of the listed accessibility errors and issues, a line number is provided indicating the location in the source code where they have been identified. All accessibility findings specified in the reports are linked to a brief explanation on what is meant by this finding, how this issue can be avoided, and a rationale for this finding including which user groups are affected. At the end of the explanation a link to the according reference in WCAG1 and Section 508 is offered.

2.2.5 Deque Ramp

Deque Ramp is a commercial web accessibility tool that brands itself as being both 'powerful and comprehensive' to sighted and non-sighted web developers. Deque Ramp promotes itself as a system that is platform independent and designed to test and retest a site's accessibility.

Deque Ramp evaluates a website either page-by-page, or by spidering through the website as dictated by the author of the evaluation. This enables developers and analysts to enter the website's URL, and then select how deep into the website they wish to test. Unlike basic one-page tools, Deque Ramp states it can navigate the most complex web sites and produce highly usable reports. The tool produces an array of reports including an overview of problems that directly relate to missing images or code violations in one click and more through in-depth reports relating to checkpoint violations.

The tool also includes a 'Wizard' function, to assist the web developers in displaying the results relevant to their particular criteria.

2.3 Procedure

Each expert was provided with a pile of Usability Problem Report Sheets and a numbered list of Nielsen's heuristics (Nielsen and Mack, 1994). Prior to each evaluation session, one member of the group of experts familiarised themselves with the tool to be evaluated. This person led the group through a number of accessibility audits using the tool and a general exploration of the tool's features. The group worked its way through the tool systematically as far as possible, looking for all potential usability problems.

Each time a potential usability problem was discovered (and all the experts were encouraged to propose such problems), a description of the potential problem was agreed between the experts and noted on the Usability Problem Report Sheets. Each expert then privately rated whether he/she thought it was a problem or not. Thus it was not necessarily the case that all the experts would have the same number of

reported usability problems, as they had the option of indicating that they did not think a proposed problem was actually a problem. If they did indicate that it was a problem, they also noted the severity of the problem using the 4 point scale proposed by Nielsen and Mack (1994): usability catastrophe, major usability problem, minor usability problem, cosmetic usability problem. They also indicated which of Nielsen's usability heuristics would account for the problem, if any.

This procedure was repeated until the experts were confident that all the usability problems in the tool had been uncovered. Each evaluation took approximately 2 hours to complete.

3 Results

3.1 Results for Site Valet

For Site Valet, a total of 16 usability problems were identified. These consisted of 1 catastrophe, 12 major, 2 minor and 1 cosmetic usability problem. Table 1 provides a summary of the catastrophe problem found in Site Valet (space permits us only to list catastrophic problems for each tool).

3.2 Results for Wave

For the Wave 3.0, a total of 12 usability problems were identified. These consisted of 7 catastrophe, 4 major, and 1 minor usability problems. Table 2 provides a summary of the most critical usability problems found on the Wave. This tool generated the least number of unique usability problems of all those tested, but over half of those issues identified were awarded the highest mean severity rating (catastrophe).

Table 1. Catastrophic usability problem found with Site Valet

Description	Mean severity rating	Relevant heuristics
Functionality of buttons "Accessibility", "Validate" and "links" are unclear – does "Accessibility" refer to WCAG P1, P2 and P3 or Section 508?	3.6 Catastrophe	Clarity of information

3.3 Results for Cynthia Says

For Cynthia Says, a total of 16 problems were identified, consisting of 13 major and 3 minor usability problems. This tool generated no catastrophe usability issues, but the majority of problems (81%) were defined as major usability problems. The experts also found it particularly hard to agree on which heuristics were relevant to these problems.

3.4 Results for Bobby 5.2

For Bobby 5.2, a total of 19 usability problems were identified. These consisted of 1 catastrophe, 10 major, 6 minor and 2 cosmetic usability problems. Table 3 provides a summary of the catastrophe problem found in Bobby 5.2.

Table 2. Catastrophic usability problems found with Wave 3.0

Description	Mean severity rating	Relevant heuristics
Tool does not provide a fluid route to effectively compare results and source code	3.8 Catastrophe	Lack of task support
No summary overview of test results e.g how many errors, what level the page has reached (A, AA etc)	3.8 Catastrophe	Lack of task support
Icons are non-intuitive	3.8 Catastrophe	Match between system/real world
Information explaining the icons is unclear	3.8 Catastrophe	Help and documentation
Lack of clear mapping between results page and actual web page	3.6 Catastrophe	Match between system/real world
Terminology used for preferences is unclear	3.6 Catastrophe	Match between system/real world
Nothing about what guidelines/level checked against	3.6 Catastrophe	Help and documentation

3.5 Results for Deque Ramp

For Deque Ramp, a total of 21 usability problems were identified. These consisted of 1 catastrophe, 16 major, and 4 minor usability problems. Table 4 provides a summary of the catastrophe problem found in Deque Ramp.

Table 3. Catastrophic usability problem found with Bobby 5.2

Description	Mean severity rating	Relevant heuristics
Disassociation between hitting “scan now” button and system status at bottom of page	3.6 Catastrophe	Visibility of system status

Table 4. Catastrophic problem found with Deque Ramp

Description	Mean severity rating	Relevant heuristics
No relationship between “interactive checks” and “violations”	3.8 Catastrophe	Consistency and standards

4 Discussion and Conclusion

The results of this evaluation are disappointing. All the tools evaluated had considerable numbers of very obvious and serious usability problems that could easily have been avoided. The home page for these tools requires only a field to enter a url, information about the tests to be conducted and possible options to choose, but even there problems often occurred. It was often not clear whether only the single resource was being tested, or child resources as well. The experts were sometimes forced to

conduct test evaluations to establish this. In addition, it was not always clear what guidelines/levels of guidelines would be tested.

But the bulk of problems occurred in the accessibility reporting pages. As one expert commented “if I was unsure about accessibility before I started using this tool, I would be completely baffled after using it”. The presentation of results was almost always unclear. Information such as what tests had been made, what guidelines/levels of guidelines these related to, what priority level a resource had passed, should all be very clear to a user when viewing an accessibility report, but was rarely the case. Experts felt that web developers would want to move between information about the overall conformance levels, specific accessibility violations, the code where the violations occurred and information about how to repair that code. Several of the tools have begun to made attempts to provide those links, but there were many usability problems in this area.

Overall, these entry level accessibility evaluation tools are not making the vital task of checking the accessibility of web resources easy for web developers. Those who are experts in web accessibility need to work to provide better support for the wider community of web developers to encourage them to consider accessibility in their development process.

References

1. Brajnik, G.: Automatic web usability evaluation: what needs to be done? In: Proceedings of 6th Conference on Human Factors and the Web. Austin, Texas. Available at (2003) <http://www.tri.sbc.com/hfweb/brajnik/hfweb-brajnik.html>
2. Brajnik, G.: Comparing accessibility evaluation tools: a method for tool effectiveness. *Universal Access in the Information Society* 3(3 – 4), 252–263 (2004)
3. Disability Rights Commission, *The Web: Access and Inclusion for Disabled people*. London: The Stationery Office (2004)
4. Mazzocato, G.: Accessibility Colour Wheel (Accessed: 15/02/07) Available at: <http://gmazzocato.altervista.org/colorwheel/wheel.php>
5. Nielsen, J., Mack, R.L.: *Usability inspection methods*. John Wiley, New York (1994)
6. Preece, J., Rogers, Y., Sharp, H.: *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, New York (2002)
7. Watchfire. WebXACT (Accessed: 15/02/07) Available at: <http://webxact.watchfire.com/>
8. Web Accessibility Initiative. Web Content Accessibility Guidelines Overview (Accessed: 15/02/07) Available at: <http://www.w3.org/WAI/intro/wcag.php>
9. Web Accessibility Initiative. Involving Users in Web Accessibility Evaluation (Accessed: 15/02/07) Available at <http://www.w3.org/WAI/eval/users.html>
10. Web Accessibility Initiative. Complete list of web accessibility evaluation tools (Accessed: 15/02/07) Available at: <http://www.w3.org/WAI/ER/tools/complete>