

Accessibility in Non-Professional Web Authoring Tools: A Missed Web 2.0 Opportunity?

Christopher Power, Helen Petrie
Department of Computer Science
University of York
York, UK
YO10 5DD
{cpower,petrie}@cs.york.ac.uk

ABSTRACT

The advent of Web 2.0 technologies, and the increased participation of users in personalized web experiences, has created a need for new web authoring tools intended for use by non-professional web authors. These tools represent a prime opportunity for including accessibility features early in the tool design process. The results from an accessibility evaluation of one of these tools demonstrates that such opportunities could be easily missed.

1. INTRODUCTION

At the beginning of the World Wide Web (WWW), it was clear that the WWW had the promise to eliminate the information gap for people with disabilities[1]. The transfer of information into electronic form, in particular textual information, provided the perfect opportunity to separate content from presentation, enabling the easy generation of alternative format materials that match a user's particular needs and preferences.

Unfortunately, due to the rapid evolution of the WWW, the presentation content in web pages has changed from simple textual information to a wide variety of media, such as graphics, sounds and movies. This change of content has created new gaps between mainstream users and users with disabilities. Techniques on how to close these gaps has been the focus of over a decade of research, and has included the creation of guidelines regarding the accessibility of websites[3, 2] and the tools used to create them[9].

As the WWW evolved, new authoring tools emerged, such as Microsoft Front Page and Netscape Composer. These tools were intended to allow non-technical users to build their own websites. However, as these non-professional web authoring tools were developed before accessibility guidelines were completed. Thus, techniques for accessible content had to be retrofitted into existing tools, if they were included at all.

With the advent of Web 2.0, the presence of content pro-

duced and maintained by non-technical users has dramatically increased. The creation of blogs, photo/video sharing sites and personal web spaces are now arguably the most important uses of the WWW[6]. With an increased demand for personal contributions, a new selection of non-professional web authoring tools are being developed. Packages like iWeb from Apple Inc.[7] and Google Page Creator[5] are powerful tools for creating web content. These tools, which are being developed from the ground up, represent an opportunity to integrate accessibility into the non-professional design world right from the beginning of the tool design process. However, the results of this paper demonstrate that this opportunity could be easily missed.

This paper begins with a review of the existing literature that demonstrates the current poor state of web accessibility, which underlines the need for tools to create accessible content. The paper proceeds to discuss an examination of one of the new non-professional web design tools: iWeb. The interface of iWeb itself, and the contents of the websites it produces, are examined in the context of current accessibility guidelines. The results demonstrate several major failings in addressing accessibility concerns at the most basic of levels. Following these observations are recommendations on solutions to these problems. The paper concludes with a discussion of what types of features can be included in modern non-professional tools to take advantage of this unique opportunity.

2. RELATED RESEARCH

In a recent study performed for the Disability Rights Commission (DRC) in the UK it was determined that the current state of accessibility in the WWW is very poor[4]. In the DRC study 1000 websites were tested using automatic tools. It was found that only 19% of the sites examined achieved the WAI Priority 1 level of conformance[8].

From these 1000 sites, a sample of 100 sites were tested by a panel of 50 disabled users for accessibility problems. On these websites the user panel members performed 913 website tasks, only 76% of which could be completed by people with disabilities.

In light of these statistics, the report had a number of key observations and recommendations for improving the outlook for accessibility on websites. Among the observations was the fact conformance to WCAG guidelines is a necessary step, but not sufficient step, for ensuring the accessibility of websites. Indeed, even though only 55% of the problems identified during the user panel evaluations were addressed

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

W4A2007 - *Communication Paper*, May 07–08, 2007, Banff, Canada. Co-located with the 16th International World Wide Web Conference.
Copyright 2007 ACM 1-59593-590-8/06/0010 ...\$5.00.

by the WCAG guidelines, if these guidelines are ignored the situation worsens significantly.

One obvious solution to accessibility problems is to include automatic tools that would allow non-professional web authors to test their own websites for accessibility violations. However, it is unclear if a non-professional author would know how to properly interpret failures identified in the automatic testing tools without significant understanding of the code base, an understanding which is in conflict with one of the basic principles behind non-professional tools: that the technical aspects of web site design are hidden from the user.

In addition to this, almost 69% of the checkpoint violations in the DRC study require qualitative evaluation by the author. However, non-professional web designers do not receive training in the best practices of accessibility. If this is the case, then how can a novice designer perform such an evaluation?

The implication is that non-professional tools must strive to provide more than just automatic conformance, and instead enhance the authoring process with accessible design happening “out-of-the-box”. The tools *must* provide good accessibility practice examples in their pre-packaged templates and stylesheets. Indeed, this is in line with recommendations of the DRC report that firmly place the responsibility for accessibility concerns at the feet of the developers and vendors of web authoring tools.

Before proper recommendations regarding the development of future non-professional tools can be provided, the potential/existing problems must be identified. In this paper, the lack of conformance to WCAG 2.0 guidelines of the iWeb design tool is demonstrated as a first step in this process.

3. METHODOLOGY

The iWeb tool interface relies heavily on providing the user with templates for creating complete web sites, and further, templates for specific pages. In order to identify the accessibility challenges that are present in iWeb, the authors created 4 different websites with each site being based on a different built-in site template. Within each of these websites 4 pages were created from different page templates, with the page types being: a home page, a blog page, a photo album page and a general information page.

During the creation of these pages, the Mac OSX built-in screen reader, VoiceOver, was activated and used. A log of interface problems that would present challenges to web authors with print disabilities was kept during the web page design process.

When the design of the websites was completed, an expert analysis of the code with reference to the WCAG accessibility guidelines was completed. During this analysis, the presentation of the website was evaluated in the Safari web browser in conjunction with the VoiceOver screen reader to identify further accessibility problems that may be present but not immediately indicated by a code review.

4. PAGE CONTENT ACCESSIBILITY

When investigating the contents of the website pages, it became clear very quickly that the majority of problems that were present originated in the web page templates. For the following discussion, the reader is referred to Figure 1.

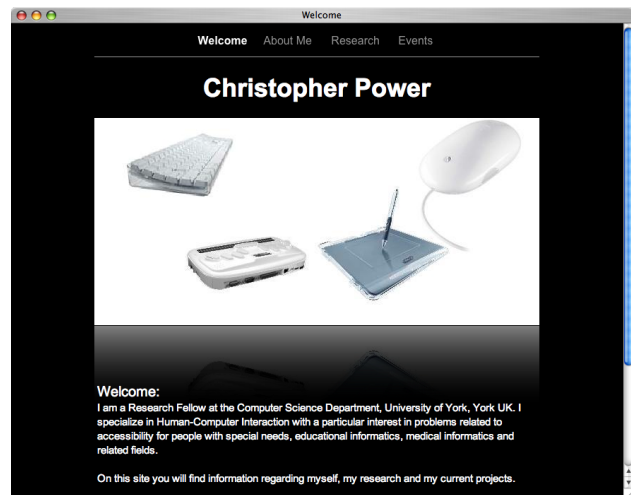


Figure 1: A Welcome home page produced from an iWeb template.

A subset of the accessibility problems found in the page content were as follows:

- Alternative text is not available for any of the pictures or other non-text content.
- Heading tags are not used to semantically mark up sections of the document for easy access with alternative user agents, such as screen readers.
- Text decoration is done through tags which do not provide semantic markup for access through alternative user agents.
- The size of sub-sections within a page are hard-coded, resulting in unpredictable behaviour when text is resized by a web browser.

Each of these points are discussed in detail in the following section.

One of the most commonly found violations of accessibility guidelines is the lack of proper alternative text for non-text components of web pages (WCAG 2.0 Guidelines 1.1, Level 1). In the case of the pages generated in iWeb, outside of a “Photo Page” template where captions can be used, it is impossible to add either alternative text (alt text) or long description text to pages within the iWeb interface for graphics or multimedia items (such as movies). Of course, it is possible to edit the code in an external text editor, but this is contrary to the intention of a tool for non-professional web authors.

Many of the predefined templates for web pages have areas labelled with what appear to be headings. These labels provide a natural way of dividing up areas of a web page so that information can be found easily. The area labels, such as the *Welcome* label near the bottom of Figure 1 can have their contents edited by authors as well as their appearance. The font, colour and position can all be adjusted to conform to the preferences of an author. Unfortunately, these area labels are not marked up semantically so that they can be programmatically determined by alternative user agents, such as screen readers. Such determination could be made

if tags such as *h1*, *h2* were used (Guideline 1.3, Level 1). Indeed, when accessing a website in a browser, the VoiceOver screen reader jumps from hyperlink to hyperlink, skipping these useful semantic markers on the welcome page.

Similarly, authors using iWeb are able to decorate text with a variety of effects such as underlining, italics and bold. According to WCAG guidelines (Guideline 1.3, Level 1), it is necessary to use proper semantic markup, such as the *em* or *strong* tags, to make this information available to alternative user agents when these types of features are used to emphasize information on the page visually. The iWeb code does not contain any of these semantic tags, with the decoration being strictly presentational through Cascading Style Sheet (CSS) markup.

The navigation bar provided at the top of each web page is generated when the pages are named by the author through the iWeb interface. This navigation bar, despite appearing to be text, is in fact a set of graphics. These graphics have no alternative text, and as a result, when read by a screen reader there is no information presented to the visitor of the website.

Finally, at first glance, it appears that iWeb avoids one of the major pitfalls of accessible web development: the use of tables for layout. In place of tables, the *div* tag is appropriately used to divide the page into various sections. Unfortunately, further investigation revealed that these areas have their size attributes hard coded. This hard coding interferes the resizing of text for people with low vision. When text is resized, it often results in overlapping text, as shown in Figure 2. In other cases, all text disappears from the page.

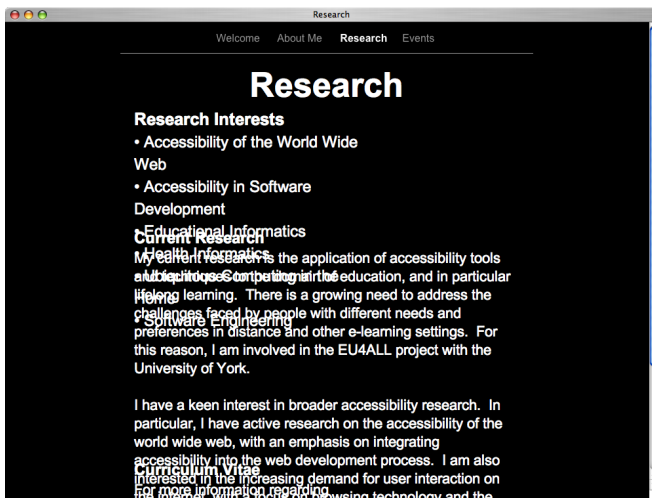


Figure 2: A web page with the text size increased within the browser.

5. INTERFACE ACCESSIBILITY

While a full usability analysis of the iWeb interface is beyond the scope of this paper, anecdotal reports from users of iWeb categorize the iWeb tool as being one of the easier means of authoring websites.

Part of this ease of use is possibly related to the interface hiding the website code from the users. At no time must the user understand HTML code, or use CSS, to create

extremely expressive and decorative designs. Instead, the interface is much like a traditional word processing application. It has familiar controls for adjusting font attributes, assigning colours and for the addition of a wide variety of media.

Unfortunately, this abstraction away from the code base leads to the inability to adjust for some of the shortcomings in the content templates discussed in Section 4. For example, if a user has some knowledge of accessibility, it is impossible to replace area labels with proper heading tags, or to replace text decoration with proper semantic markup tags. As already discussed, there is also no way to provide alternative text for various non-text elements, even if the user wanted to do so.

Given the difficulty in creating accessible content within the iWeb interface, it is perhaps unsurprising that there are deeper accessibility problems that are revealed when the tool is used with the operating system provided screen reader. In Figure 3 the reader will find an example of the iWeb interface. The following significant accessibility problems were identified while building websites in iWeb:

- The configuration of the opening screen of the iWeb application places the user focus in the outline panel located on the left hand side of the window; however this focus is not reported to the user on startup of the application.
- The outline panel itself is voiced relatively well by the screen reader. The tree structure is reported properly and can be navigated via the keyboard. Unfortunately, it is impossible to properly move into the right hand content panel to edit the pages appropriately. The only information reported by the screen reader on entering the outline panel are the words "Scroll Area".
- Within the content panel it is impossible to move between components on the page, such as text areas or other media.
- Addition of all non-text content must be done through mouse interaction, which is difficult if not impossible for some people with visual or physical disabilities.
- The panel of control buttons at the bottom of the interface in Figure 3 is unreachable through anything other than a point-and-click interface.
- The *Inspector Panel* (not shown in Figure 3), which contains alternative access to the options in the control buttons, is also unreachable without the use of a point-and-click device.

These barriers result in a tool that would be very difficult to use, in particular for people with visual disabilities.

6. DISCUSSION

The accessibility evaluation results presented in this paper are clearly not a complete audit of the websites produced by iWeb, or of the tool itself. Indeed, evaluations regarding this and other non-professional tools are currently underway. However, these results do highlight how some of the most well known, and easily provided, accessibility techniques are missing from this tool. How can this situation be addressed before this opportunity is missed?

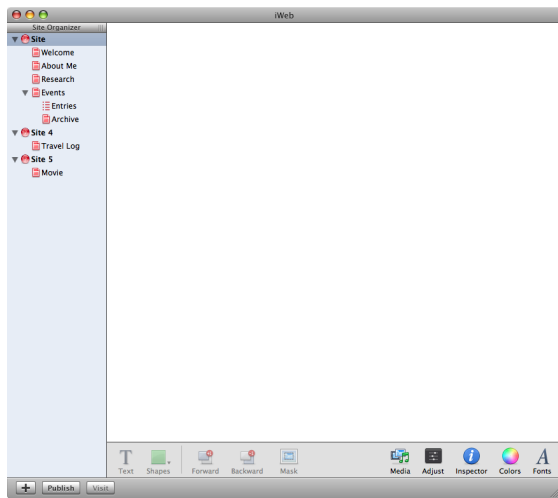


Figure 3: The iWeb main interface window.

6.1 Developer Education

It is essential that developer knowledge on accessibility be addressed at the training level. There is a desperate need to integrate accessibility guidelines, standards and testing into the training that tool developers receive. With such training in place, the examples provided by these developers can serve to demonstrate best-practice to those who do not have an interest in the technical aspects of website creation.

6.2 Best Practice Integration

It is essential that best practices be integrated into the content templates provided, as well as in the interface of the authoring tools.

The majority of the content issues discussed in the results of the accessibility evaluation can be addressed by very minor changes to the code base without changing the iWeb interface. For example, it is clear that authors will want to be able to change the appearance of the headings on their website. This can be accomplished just as easily as it is now, with the code for the area headings being actual HTML heading tags augmented by CSS. This would provide the flexibility the authors desire while simultaneously addressing the accessibility of the content they produce.

In some cases, the interface of iWeb must be adjusted. For example, in order to add alternative text to a picture, one could use a dialogue box to request a text alternative for a picture, or provide a context menu.

6.3 Non-Professional Education

With best practices in place, developers could pass on their knowledge about accessibility to the non-professional users of their tools.

Consider the situation discussed above, where a dialogue box is used for entering a text alternative. Imagine if there was a hyperlink that said *Click here to find out why this is important*, which would take the user to a discussion about why alternative text is important to people with visual disabilities. All of a sudden the interface becomes a powerful educational tool for reaching people who typically would not have the opportunity to learn about accessibility.

7. CONCLUSIONS

This paper examines the accessibility of one of a collection of new non-professional authoring tools. The evaluation that was completed has demonstrated that there are several missed opportunities in these new tools for providing accessible content on what is becoming one of the biggest uses of the WWW: personal websites. However, it has also demonstrated that many of these accessibility problems can be addressed with relatively little effort. Finally, this paper has demonstrated that there is a prime opportunity to include accessibility in the early designs of new products being created for Web 2.0; an opportunity that could be easily missed.

8. REFERENCES

- [1] Bart Bauwens, Jan Engelen, Filip Evenepoel, Chris Tobin, and Tom Wesley. Increasing access to information for the print disabled through electronic documents in sgml. In *ASSETS*, pages 55–61. ACM, 1994.
- [2] Ben Caldwell, Wendy Chisholm, John Slatin, and Gregg Vanderheiden. Web Content Accessibility Guidelines 2.0 (draft). Online Website at <http://www.w3.org/TR/WCAG20/> Retrieved February, 2007, 2007.
- [3] Wendy Chisholm, Gregg Vanderheiden, and Ian Jacobs. Web Content Accessibility Guidelines 1.0. W3C Website; Retrieved in March, 2007 from <http://www.w3.org/TR/WCAG10/>, 1999.
- [4] Disability Rights Commission. The Web: Access and Inclusion for Disabled People, 2004.
- [5] Google. Page creator. Product Website; Retrieved in March, 2007 from <http://pages.google.com>, 2007.
- [6] Lev Grossman. Time’s Person of the Year: You. *Time Magazine*, 168(26), 2006.
- [7] Apple Inc. iWeb. Product Website; Retrieved in March, 2007 from <http://www.apple.com/ilife/iweb/>, 2007.
- [8] Web Accessibility Initiative. Level A Conformance to Web Content Accessibility Guidelines 1.0. W3C Website; Retrieved in March, 2007 from <http://www.w3.org/WAI/WCAG1A-Conformance>, 1999.
- [9] Jutta Treviranus, Charles McCathieNevile, Ian Jacobs, and Jan Richards. Authoring tool accessibility guidelines 1.0. W3C Website; Retrieved in March, 2007 from <http://www.w3.org/TR/WAI-AUTOOLS/>, 1999.