

Integration of Web Accessibility into Agile Methods

Sergio Luján-Mora and Firas Masri

Department of Software and Computing Systems, University of Alicante, Alicante, Spain

Keywords: Accessibility, Web, Agile, Evaluation, Testing.

Abstract: In a short period of time, the World Wide Web has had a huge impact on our society and lives. In web sites and web applications, accessibility and usability are essential key requirements. Unfortunately, most web sites are inaccessible to many disabled people and fail to satisfy the most basic standards for accessibility. Many of the barriers people with disabilities face on the Web are completely avoidable and the disadvantage associated with disability can be entirely overcome. To support the accessibility of web sites, different accessibility guidelines and standards have been introduced for the last ten years. Nevertheless, a web site can meet accessibility standards, but it can still be difficult for people with disabilities to use it. Moreover, web accessibility has been often an afterthought in the development process of web sites. In many cases, web developers provide an adaptation or a fix to the interface of a web site after it has been released to the public. In this paper, we argue that the adoption of agile software development methods can help to improve the accessibility of web projects. Besides, the integration of accessibility into agile methods is proposed.

1 INTRODUCTION

The Web has become an essential part of our society and lives. We are continually embracing the Web to better facilitate our business, jobs, and social lives.

According to the World Wide Web Consortium (W3C), *“The Web is fundamentally designed to work for all people, whatever their hardware, software, language, culture, location, or physical or mental ability. When the Web meets this goal, it is accessible to people with a diverse range of hearing, movement, sight, and cognitive ability”* (W3C, 2012). This principle is paramount, because it implies the impact of disability is radically changed on the Web because the Web breaks down accessibility barriers to communication and interaction that many people face in the physical world. However, providing equal access to people with different disabilities (visual, hearing, cognitive, mental, and physical impairments) represents a huge challenge for web designers and developers.

Traditionally, web accessibility for people with disabilities has been an afterthought. In many cases, web developers provide an adaptation or a fix (an “add-on”) to the interface of a web site after it has been released to the public. The basis of this strategy often is built on the perception that a small population of people with disabilities actually use the Web. However, approximately 15% of the

world’s population has a disability that could impact their web surfing experience (WHO, 2011). Moreover, the prevalence of disability is growing due to population ageing and the global increase in chronic diseases.

Many web projects fail to take accessibility into account. It is very common that many web projects start out with the commitment to achieve some level of web accessibility. However, following the schema of traditional software development methods, such as the waterfall model (Royce, 1970), accessibility, in the form of accessibility testing, is postponed to once the web site is built. But at the end of the project, time starts to decrease rapidly and resources start to be assigned to other priorities. Unfortunately, at the end of the project, accessibility is dropped or postponed for a later version of the web site.

In this position paper, we argue that traditional software development methods are not suitable for the achievement of web accessibility and we advocate that the use of agile methods could improve the accessibility of web projects. In the next section, we briefly review how accessibility is tackled in current web projects and we highlight the most important drawbacks. Then, a brief summary of the state of the art in web accessibility evaluation is presented. After this, we present the Agile Manifesto that promotes a new way of software

development. This presentation is followed by our approach which involves adapting the web accessibility requirement to the Agile Manifesto. In this section, the integration of accessibility into agile methods is proposed. Then, we conclude this paper with discussion of future directions for our work.

2 ACCESSIBILITY IN WEB PROJECTS

The waterfall model (Royce, 1970), the traditional software development method for the last 30 years, has been proven to be not suitable for non-trivial projects. The waterfall development model comes from the manufacturing and construction industries (highly structured physical environments) in which processes are formed by sequential phases clearly defined. One should move from one phase to the following phase in a linear fashion only when the phase is completed and perfected.

One of the main problems of the waterfall model is that all possible features of the final product must be planned out in detail prior to the implementation. This planning results in huge amounts of documentation. Moreover, moving back to a previous phase is very expensive. The implication of this is that it is very difficult to respond to changes once the project has started.

Regarding the web development, in the waterfall model, accessibility issues that are detected at the end of the project are very difficult to repair.

The W3C, the main international standards organization for the World Wide Web, is engaged in promoting the creation of accessible web sites. The Web Accessibility Initiative (WAI) is a special working group established by the W3C in April 1997.

The W3C develops web accessibility guidelines for the three main components:

- *Authoring Tool Accessibility Guidelines* (ATAG) addresses authoring tools.
- *Web Content Accessibility Guidelines* (WCAG) addresses web content, and is used by developers, authoring tools, and accessibility evaluation tools.
- *User Agent Accessibility Guidelines* (UAAG) addresses web browsers and media players, including some aspects of assistive technologies.

Regarding the web content, the W3C has developed the most important guidelines concerning web accessibility, the WCAG versions 1.0 and 2.0. In recent years, these guidelines have been officially accepted by many countries as the definitive

guidelines on how to create accessible web sites.

These guidelines tend to address the most important accessibility problems and provide guidelines for making web pages capable of being rendered by assistive technology devices, such as screen readers. In order to assure that these guidelines are correctly applied, different web accessibility evaluation methods have been proposed for the last ten years. In the next section, a brief summary of the state of the art in web accessibility evaluation is presented.

3 WEB ACCESSIBILITY EVALUATION

In many projects, web accessibility is considered a requirement that must be accomplished at the end of the web site development. Several studies have suggested numerous evaluation methods (Vigo et al., 2007) as a means to verify, measure and certify the fulfillment of the accessibility guidelines and therefore to supply full accessibility to disabled people. Currently, there are two types of evaluation methods: the qualitative methods (analytical and empirical) and the quantitative methods.

The qualitative methods have been the most used until now, specially the analytical ones, which are characterized by their low cost and ease of use. The other analytical evaluation methods, which are based on the manual heuristic inspection of code, do not guarantee full accessibility (Brajnik, 2008); it depends largely on evaluators' experience and the adopted guidelines. On the other hand empirical methods are generally more expensive, but more accurate, because they clearly show the most catastrophic accessibility faults.

The quantitative methods help to understand, control and improve the final product (Fenton and Pfleger, 1998), thus its main goal is to assure the quality results and monitor the accessibility level by establishing values and summarizing results. These methods and due to their nature aren't sufficient enough to assess accessibility and evaluators can't depend only on them.

These methods consider that the best way to ensure that a web site is accessible is to ensure that accessibility requirements are completely defined and documented at the beginning of a project, and then those accessibility requirements are tested at the end phases of the project. However, at the end of the project accessibility problems are often unable to be repaired.

Agile software development methods are considered to be the antagonist of traditional software development methods, such as the waterfall method. In our previous works (Masri and Luján-Mora, 2011), we have argued that agile methods can help to improve the accessibility of web sites. Deploying web sites and web applications is further more competitive than deploying software in the past. Agile methods are the only ones that can cope with this competitiveness. Other authors have also defended that agile methods can help to integrate accessibility with reduced resistance (Groves, 2011a); (Groves, 2011b). Besides, some authors have developed a process model for agile software development that takes into account accessibility and universal design (Bonacin et al., 2009).

In the next section, we detail the “Agile Manifesto”, the main background work used to delineate our agile accessibility method.

4 THE AGILE MANIFESTO

In February 2001, seventeen software developers published the *Manifesto for Agile Software Development* (Beck et al., 2001). This manifesto sparked the agile software development movement. The Agile Manifesto was also complemented by “Twelve Principles of Agile Software”, which gave more detail behind the Agile Manifesto.

Agile software development methods break tasks into small iterations that typically last from one to four weeks. At the end of each iteration, a full working part of the software has been completed and tested and is ready to be shown to any stakeholder of the project.

Nowadays, there are many agile methods. Koch (Koch, 2005) presents a systematic way to evaluate which method is more adequate to a particular organization.

In the next section, we introduce our approach which involves adapting the web accessibility requirement to the Agile Manifesto.

5 AGILE ACCESSIBILITY

WCAG 1.0 and 2.0 define four ordinal levels of accessibility (none, A, AA, and AAA), and provide a set of checkpoints or success criteria for each level. A web page must satisfy all priority A checkpoints or criteria to be considered minimally accessible. Web developers may implement priority

AA and priority AAA checkpoints or criteria to provide increased accessibility for users. Unfortunately, this system of evaluation does not reflect the real accessibility of a web site: if a web site satisfies many checkpoints in addition to all level A checkpoints, the web site will only conform to level A of WCAG, but the additional efforts to achieve a better level of accessibility will not be visible. Therefore, compliance with the technical guidelines is only the first step towards accessibility.

In the next sections, the web accessibility is related to the Agile Manifesto’s values: individuals and interactions; working software; customer collaboration; and responding to change.

5.1 Individuals and Interactions

In agile methods, self-organization and motivation are important, as are interactions between different stakeholders. The Agile Manifesto states that “*the most efficient and effective method of conveying information to and within a development team is face-to-face conversation*”. Therefore, the cooperation between people participating in the design and implementation must be encouraged. Everyone in a development team must be engaged with web accessibility.

Moreover, interaction between developers and final users must be also encouraged. User-centered design, which focuses attention on the needs and limitations of end users, is a keystone of our approach. Final users should participate as external reviewers from the beginning of the project. Besides, the best way to achieve a user-centered design is the application of user testing. Accessibility user testing is an irreplaceable technique, since it provides direct information on how real users use web sites and which real problems they face. Accessibility user testing is explained with more detail in the following section “Customer collaboration”.

5.2 Working Software

In agile methods, working software is the primary measure of progress and is delivered frequently (weeks rather than months). From the beginning of a project, web site prototypes should be delivered with accessibility in mind. Therefore, accessibility testing must be done from the early stages of a project and must not be postponed to the end of the project.

Accessibility problems are often caused by underlying incorrect markup or compatibility issues. Accessibility tests must be run continuously and must be automated in order to detect these

accessibility issues from the beginning of a project. This ensures accessibility problems are caught and eliminated during the development. This is not possible for the waterfall method, since the final product is tested only at the very end, which means any accessibility problem found will result in the entire product having to be re-written or the problem not being resolved.

5.3 Customer Collaboration

The satisfaction of the customer is the highest priority behind the Agile Manifesto. In the case of a web site, although the customer is the person who orders and owns the web site, regarding the web accessibility we consider the end user as the customer.

Many web developers think only blind people face accessibility. Although *“the most serious accessibility problems given the current state of the Web probably relate to blind users and users with other visual disabilities since most web pages are highly visual”* (Nielsen, 1996), other people with different disabilities also face accessibility problems. The WCAG 1.0 states that many users may be accessing a web site in contexts very different from the designers' context:

- They may not be able to see, hear, move, or may not be able to process some types of information easily or at all.
- They may have difficulty reading or comprehending text.
- They may not have or be able to use a keyboard or mouse.
- They may have a text-only screen, a small screen, or a slow Internet connection.
- They may not speak or adequately understand the language in which the document is written.
- They may be in a situation where their eyes, ears, or hands are busy or interfered with (for example, driving to work, or working in a loud environment).
- They may have an early version of a browser, a different browser entirely, a voice browser, or a different operating system.

Accessibility user testing is the best technique to identify (and later correct) accessibility problems. We propose to carry out accessibility user testing frequently and from the beginning of the web project. It is very important to have instant access to feedback from disabled people (Edwards, 2010). These testing sessions can be conducted with a small amount of users: according to some studies (Nielsen,

2000), a usability test with only a single user helps to detect almost a third of all the problems a web site has. When adding more users, less and less is learnt because following users do the same things as previous users. However, some authors (Faulkner, 2003) state there is some risk of using a short number of users. As a first approach, we recommend performing accessibility testing with less than five users from each disability category.

Accessibility user testing highlights important accessibility problems and leads to rate the severity of the problems correctly. Therefore, this testing allows developers to prioritize the impact of accessibility problems and helps to detect problems whose solution makes a difference in accessibility. However, accessibility user testing presents some drawbacks: they imply higher cost than experts' conformance testing, they mix accessibility and usability problems, and their logistics is complicated (Brajnik, 2008).

5.4 Responding to Change

Traditionally, user interfaces have been created assuming that users have concrete tasks or goals in mind. However, when users surf the Web, their goals shift and change as they find their way through the Web. Therefore, conventional user testing method, where one user is put on one computer to see how they surf through a web site, should be rethought. Besides, some researches argue that users do not operate in the real world in the same way as they are asked to act in user research and usability testing (O'Brien, 2012).

Nowadays, majority of web sites are created using content management systems (CMS). Thanks to this, the work that used to take up 80% of a web development project is automated by CMS. Therefore, there is a clear shift in the effort of a web project: whereas in the past (five or ten years ago), the main part of the working effort was invested in programming, nowadays the main effort is put on the maintenance and the adaptation to the new requirements and functionalities. This is not possible when the waterfall method is employed, since any changes to be made means the project has to be started all over again.

6 CONCLUSIONS

The evolution of the Internet and the World Wide Web has grown from a novelty to become fully integrated into our lifestyles. Web accessibility

involves making web content available to all individuals, regardless of any disabilities or environmental constraints they experience. However, several web sites often fail to meet minimum web accessibility standards. Part of the problem lies on the way web accessibility is tackled: web accessibility is considered a requirement that must be checked at the end of the web site development. Besides, testing accessibility at the end of a project is more expensive, difficult, and time-consuming than taking into account from the beginning.

In this paper, we have discussed the adoption of agile methods as a means to achieve real web accessibility. In our approach, web accessibility should be tested during development. Therefore, user testing with people with disabilities should be done early in the development process.

More work is needed to provide additional evidence of advantages and disadvantages of the integration of web accessibility into agile methods. We plan to evaluate the advantages and disadvantages of our approach in real web projects and we also plan to specifically tune up our agile method for the development of accessible web applications.

ACKNOWLEDGEMENTS

This paper has been partially financed by the MESOLAP (TIN2010-14860) project from the Spanish Ministry of Science and Tehenology (currently Ministry of Economics and Competivity).

REFERENCES

- Beck, K., Beedle, M., van Bennekum, A., et al., 2001. Manifesto for Agile Software Development. Internet: <http://agilemanifesto.org/> (visited 20/12/2011)
- Bonacin, R., Baranauskas, M., and Rodrigues, M. A., 2009. An Agile Process Model for Inclusive Software Development. In *11th International Conference on Enterprise Information Systems (ICEIS 2009)*, Lecture Notes in Business Information Processing 24, p. 807-818, Milan (Italy), May 6-10.
- Brajnik, G., 2008. Beyond Conformance: The Role of Accessibility Evaluation Methods. In *WISE'08, Proceedings of the 2008 International Workshops on Web Information Systems Engineering*, Lecture Notes in Computer Science 5175, p. 63-80, Auckland (New Zealand), September 1-3.
- Edwards, P., 2010. Accessibility Testing in an Agile Process. Internet: <http://pauledwards.posterous.com/accessibility-testing-in-an-agile-process> (visited 18/12/2011)
- Faulkner, L., 2003. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. In *Behavior Research Methods, Instruments, & Computers*, 35 (3), p. 379-383.
- Fenton, N. E., and Pfleeger, S. L., 1998. *Software Metrics: A Rigorous and Practical Approach*. Boston, USA: PWS Publishing Co.
- Groves, K., 2011a. Agile & Accessibility. Internet: <http://www.karlgroves.com/2011/09/28/agile-accessibility/> (visited 10/01/2012)
- Groves, K., 2011b. It's time to let go of the waterfall model of accessibility. Internet: <http://www.karlgroves.com/2011/10/03/it-is-time-to-let-go-of-the-waterfall-model-of-accessibility/> (visited 10/01/2012)
- Koch, A. S., 2005. *Agile Software Development: Evaluating the Methods for Your Organization*. Norwood, USA: Artech House.
- Nielsen, J., 2000. Why You Only Need to Test with 5 Users. Internet: <http://www.useit.com/alertbox/20000319.html> (visited 10/09/2011)
- Masri, F., and Luján-Mora, S., 2011. A Combined Agile Methodology for the Evaluation of Web Accessibility. In *IHCI 2011, IADIS International Conference Interfaces and Human Computer Interaction 2011*, p. 423-428, Rome (Italy), July 24-26.
- Nielsen, J., 1996. *Accessible Design for Users With Disabilities*. Internet: <http://www.useit.com/alertbox/9610.html> (visited 08/01/2012)
- Nielsen, J., 2000. Why You Only Need to Test with 5 Users. Internet: <http://www.useit.com/alertbox/20000319.html> (visited 08/01/2012)
- Royce, W., 1970. Managing the Development of Large Software Systems. In *Proceedings of IEEE WESCON*, p. 1-9, August 26.
- Vigo, M., Arrue, M., Brajnik, G., Lomuscio, R., and Abascal, J., 2007. Quantitative metrics for measuring web accessibility. In *Proceedings of the 2007 International Cross-Disciplinary Conference on Web Accessibility*, p. 99-107, Banff (Canada), May 7-8.
- World Health Organization (WHO), 2011. World Report on Disability 2011. Internet: http://www.who.int/disabilities/world_report/2011/en/index.html (visited 10/12/2011)
- World Wide Web Consortium (W3C), 2012. Accessibility. Internet: <http://www.w3.org/standards/webdesign/accessibility> (visited 06/03/2012)