Engineering Accessibility in Web Content Management System Environments

Juan Miguel López, Afra Pascual, and Antoni Granollers

Griho Research Group, University of Lleida, Jaume II St. 69, 25001 Lleida, Spain
{juanmi,apascual,tonig}@diei.udl.cat

Abstract. Law in most countries around the world enforces accessibility requirements in websites, mainly the ones related with public administration. Evaluating accessibility is a long and laborious process that requires manual evaluation. In Web 2.0 environments, the great amount of data generated by users makes necessary further effort in order to validate web content accessibility. This paper introduces an accessibility evaluation methodology based on web content accessibility analysis and the study of web content management by users in Web Content Management System (CMS) environments. The main aim of proposed approach is to optimize the accessibility evaluation process by minimizing the effort it takes to achieve a certain accessibility level. Proposed methodological approach is used as the basis for a generic framework, which is intended to engineer accessibility in all kind of CMS environments. Proposed framework also suggests corrective accessibility maintenance activities for webmasters who are interested in the improvement of the accessibility. The paper how a prototype developed following the framework works in a concrete CMS environment.

Keywords: Web Content Management Systems, Web accessibility evaluation, monitoring, user logs.

1 Introduction

Demographic importance of accessibility is remarkable. For instance, according to the Eurostat [5], from a total population of 362 million people in Europe in year 1996, a 14,8% of the population between 6 and 64 years old had physical, psychological or sensorial disabilities. There are also powerful legal reasons in order to develop accessible web user interfaces. For instance, Section 508 [12] requires Federal agencies in the United States to make their electronic and information technology accessible to people with disabilities. This kind of legislative changes have also been performed in more other countries around the world, such as European Union countries.

Even though software engineering has been a productive research area in the last decades, web engineering is a relatively new one. It refers to specific methods, technologies and models for web application development since these applications have special characteristics. In fact, web applications differ from other traditional software applications in a wide range of aspects, such as the timeframe assigned for

G. Vossen, D.D.E. Long, and J.X. Yu (Eds.): WISE 2009, LNCS 5802, pp. 597–604, 2009.

their development, differences in the characteristics of end-users in terms of age, education and web navigational experience [9]. Because of these special characteristics of web applications, it has been necessary to define specific methods, technologies and models for their development.

In this context, Web 2.0 environments (more participative and with constant changes) enhance the importance for supplying methodologies, mechanisms and tools able to verify accessibility compliance. In this context, Web Content Management Systems (CMS) provide a way for users to manage web content. Aspects such as providing betterment proposals in web content management by users with no technical skills and optimizing the accessibility evaluation process by reusing results from previous accessibility evaluations are main points to be taken into account. This last aspect is especially important given the great increase on the amount of user managed web content that introducing the Web 2.0 paradigm implies.

This paper is as structured as follows. Section 2 describes the background of presented work. Section 3 introduces a methodology to verify accessibility in CMS environments, emphasizing its approach towards solving accessibility problems. In order to provide support for proposed methodology, a framework for monitoring accessibility is proposed. A prototype developed to implement the proposed methodology and tested in a concrete CMS that manages a real website is described in Section 4. Finally, Section 5 shows conclusions and future work.

2 Related Work

Traditionally, accessibility evaluation has been performed revising the fulfilment of WCAG 1.0 guidelines proposed by the Web Accessibility Initiative (WAI) in the World Wide Web Consortium (W3C). In order to fulfil an accessibility evaluation for a website, (X)HTML content and CSS style sheet linked to each web page are reviewed as a first step. Then, automatic accessibility evaluation tools are used [4], which show a revision of automatically detected problems. These tools also point out several possible problems that cannot be automatically revised and require manual revision by accessibility evaluators. Use of assistive technology such as screen readers or text browsers, together with visualization of web pages in different browsers, provide support for manual revisions. All mentioned steps form part of the accessibility evaluation methodology provided by the W3C [6].

Web 2.0 paradigm has changed the way the Web is used and perceived. Rather than a mechanism to provide information, the web is now interactive and harnessing the wisdom of many through wikis, blogs, and e-communities [7]. New terms such as social networking and collective intelligence have been coined to explain the new phenomenon. As for web 2.0 technologies, beyond basic HTML the most widely used technologies for implementing web 2.0 are scripting and Cascading Style Sheets (CSS). Scripting can make use of XmlHttpRequest for optimize data transfer over then standard HTTP web protocol. Rich Internet Application (RIA) technology provides more efficient and collaborative web applications. Accessibility is a major concern in web 2.0 [7]. Since most existing accessibility evaluation tools evaluate the resulting HTML pages, as more dynamic server side web development technologies are used, it is difficult for a tool to determine the exact source of the error. Adding

Web 2.0 dynamic updates into the mix and the testing strategy gets much more difficult. In this context, new tools are needed to address Web 2.0 applications. These new interaction models are pushing the limits of web technologies and the ability of assistive technologies to interpret the changing face of the Web [7].

CMS systems do not enforce users in aspects that should be taken into account in order to generate accessible web content. In a website managed by users using a CMS, the templates used to sketch and generate new web pages should be verified in order to eliminate the risk of introducing accessibility faults in web content. In addition, all content introduced by users should be verified for ensuring the fulfilment of accessibility guidelines. In this sense, CMS systems usually do not generate standard accessible code and allow few user modifications towards improving website accessibility. These modifications are often insufficient to provide entirely accessible websites to people with disabilities [10].

Accessibility monitoring tools are systems that constantly monitor accessibility compliance in given websites. They usually notify the website administrator whether accessibility errors are detected or manual evaluations must be performed. Furthermore, existing accessibility monitoring tools provide a constant website analysis that allows knowing the state of the website accessibility on every moment [8]. Nevertheless, for CMS environments they do not analyze causes for not accessible contents, they simply evaluate the evolution of web content accessibility.

3 Methodological Approach and Description of the Framework for Engineering Accessibility in CMS Environments

In order to support accessibility, it must be taken into account in all stages of software to be developed. Several existing methodological approaches include it as a main factor to be taken into account from the beginning of a software lifecycle to the end [3]. Keeping the evolutionary nature of web applications in mind, following the classic Boehm's spiral model [2], and taking modern methods for web and software engineering into account, the development phases should be applied in an iterative and incremental manner, in which the various tasks are repeated and refined until results meet the requirements. At each iteration, current versions of the system should be tested and evaluated, and then extended or modified to meet requirements in a better manner. In this context, accessibility requirements are aimed to be fulfilled. In most countries, accessibility is a legal requirement that developed web applications and systems must fulfil. Besides, evolutionary nature of web applications makes accessibility a key aspect to be considered for maintenance, as modifications on web content have a risk factor related to accessibility compliance. This is especially important in web 2.0 environments, where web content can be managed by a wide range of users.

In order to define proposed methodology, it is departed from classical software engineering phases taking specific features related to web environments into account, but having in mind that CMS environments suppose a major challenge as the involvement of users as web content editors and managers make maintenance phase be even more important than in typical web environments. In this sense, the methodology is rooted on web engineering principles and web application's lifecycle.

On the other hand, existing accessibility guidelines for web content and authoring tools must be followed, in order to check whether the proposed web content management system and generated web content are accessible. Proposed methodology for considering accessibility in CMS environments is organized based on these assumptions. Phases of the methodology are depicted in Figure 1.

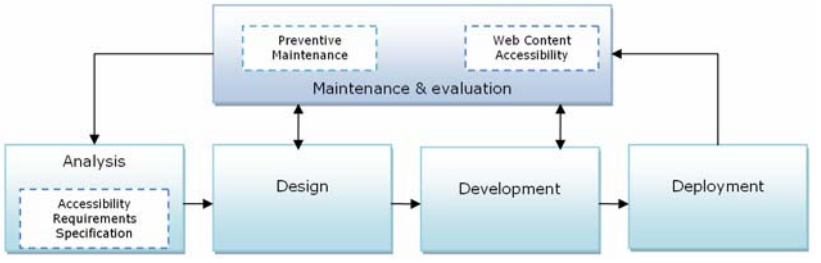

Fig. 1. Schematic approach for Web applications' lifecycle in CMS environments

Accessibility requirements specification is the activity in which the application analyst collects and formalizes the essential information about the application domain and expected functions [9] . This aspect does not significantly differ from requirement collection for traditional applications. Anyway, specification of accessibility requirements is necessary to demonstrate what accessibility issues have to be taken into account and to describe the decisions to be made within each lifecycle phase in order to fulfil them [1] .

According to [3] four types of maintenance exist related to website engineering: corrective maintenance (i.e. fixing problems with the website behaviour or inserting missing contents), adaptive maintenance (i.e. upgrading the site with respect to new technologies, like new browsers' capabilities), perfective maintenance (i.e. improving the site behaviour or content), and preventive maintenance (i.e. fixing problems in behaviour or content before they affect users). In this context, CMS accessibility compliance heavily lays on preventive maintenance, as a big amount of accessibility errors can be solved by providing proper web content management environments. Monitoring web content changes performed by users can be a good indicator for finding potential web content management user interface issues related with accessibility problems. In the context of CMS environments, web content accessibility evaluation must be carried out after maintenance activities. This aspect is due to the fact that evaluating web content accessibility provides little value if corrective measures are not taken in the maintenance phase in order to correct detected CMS environment related accessibility issues. Otherwise, same accessibility errors would be repeated again and again, as the causes related to their appearance would not be solved.

To put proposed methodological approach into practice in a trustworthy way, a generic framework has been designed with the aim of engineering the accessibility in all kind of CMS environments. In order to support preventive maintenance, data from all the different sources available was examined in order to infer possible causes for lack of accessibility. Available data sources in CMS environments include system

change logs, system user logs, historic data about previous accessibility evaluations, data about the examined CMS environment and how it manages web content. Figure 2 shows the architecture for the proposed framework.

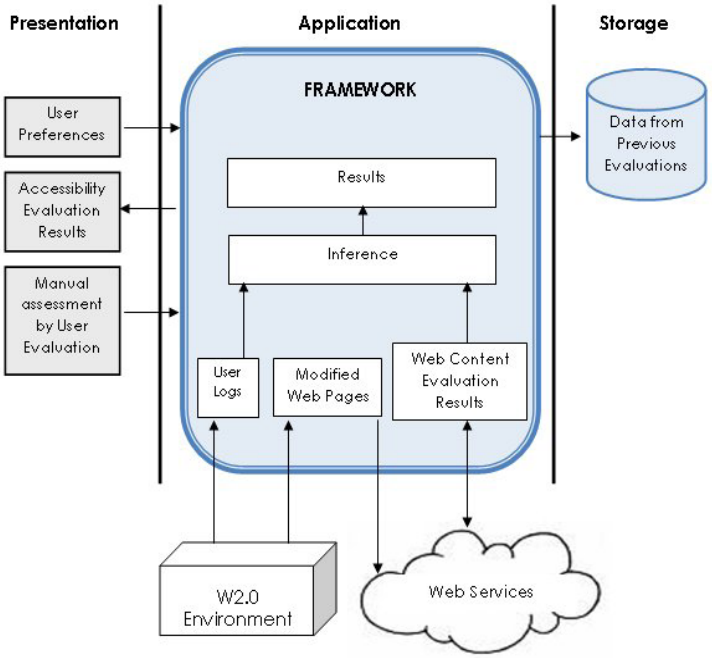

Fig. 2. Architecture for proposed framework

User log data for analysis is received from the CMS environment analyzed. Accessibility related changes from previous evaluation are obtained from the administration area of the framework, where changes among the different versions of modified web pages are stored. HTML code of web pages to be analyzed is obtained from the CMS environment. Selected ones are only the ones which have been modified from the last accessibility evaluation. This is made with the aim of optimizing time and resources. It eliminates a great amount of redundancy from the accessibility evaluation.

4 Case Studio: Web Site Managed by a CMS

A prototype implementing proposed framework was performed and integrated in a CMS. Concretely, it was integrated in the OpenCMS environment that manages the http://griho.udl.cat website. The prototype had to provide feedback about the accessibility compliance on the workflow of the CMS. It also had to be flexible enough to be integrated in the normal workflow of the system, and be able to support

different roles. Under these circumstances, it would be able to focus on analyzing concrete problems related to accessibility in CMS environments. Accessibility requirements specification was performed by taking the legal framework into account. In this case, the fulfilment of Spanish law regarding accessibility [11] made necessary some adjustments from WCAG, as there are four priority level changes among the guidelines. In this sense, maintenance regarding accessibility evaluation also had to be adapted. To follow the W3C methodology [6] for evaluation accessibility, two automatic accessibility evaluators were used (TAW1 and EvalAccess2). Analysed web pages results were integrated in a single view inside the prototype. At the same time, they were adapted to express their result regarding the accessibility requirements for the concrete case. The accessibility of web contents was automatically re-evaluated in a daily basis. The evaluation requires an expert human evaluator to complete the analysis of those aspects not automatically detectable, as the W3C methodology states [6].

To detect the potential problems related to accessibility, it was necessary to acquire information only available in CMS user's logs registers. We needed to analyze the information related to the management carried out by the users. This information is automatically managed by the CMS and stored in its internal logs registers. Related to this aspect, it is necessary to have in mind that each CMS has its own particular presentation and storage systems. This is important because to enable data examination, corresponding CMS module must be installed and its data available for evaluators. In this sense, it must also be taken into account that information providers, in this case the CMS users, usually are not aware of the need for creating accessible content, and the tools do not do provide support to enable them in this feature.

Accessibility improvement recommendations were determined by analysing the data obtained in user logs, previously performed accessibility evaluations and the result of web content accessibility evaluations performed automatically and manually on current web pages. Results were statistically treated to detect high accessibility errors percentages related to concrete HTML elements. This way, it can be determined which errors are persistent in the system. Changes record and user log analysis allowed determining the origin of the error, which can be on the content or on the template. The system analysed the possible causes for the error and the factors that may have influence on them. This way, the system proposed recommendations to solve the persistent accessibility problem.

System's user interface is divided in three parts. The main menu, placed on the top, includes navigational actions to accede to the evaluation results, whether incremental, current or last evaluation results. Every evaluation displayed evaluation results for each evaluated page show all the information about the guidelines that have been evaluated as accessibility errors. The page related to the CMS evaluation displayed the problems found that would have influence in all web pages stored in the CMS.

In this sense, Figure 3 depicts how the system found a problem regarding the alternative text of images in the website. As most images had no alternative text, the system performed an analysis to infer possible causes for this situation. As a result, the system provided a recommended action for the webmaster of the web site, which

[1] http://www.tawdis.net/taw3/cms/es
[2] http://sipt07.si.ehu.es/evalaccess2/index.html

was to edit the template of the CMS so the images appeared with alternative text. This action could also suppose the necessity of changing the way users include images in the web site, but it required further analysis regarding the web editor used in the system, which is not currently developed in the prototype.

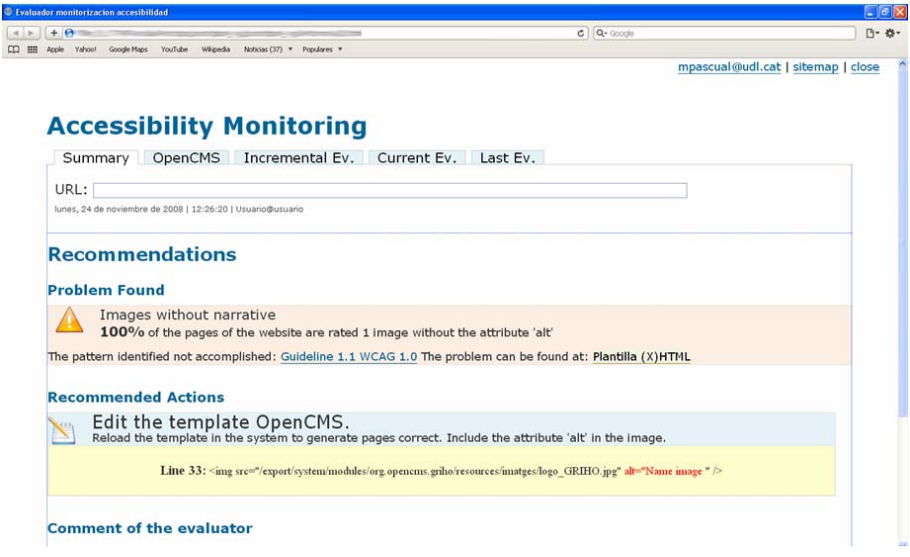

Fig. 3. Screenshot of the developed prototype

5 Conclusions and Future Work

This paper introduces a methodological approach and a framework for implementing accessibility evaluations for Web 2.0 contexts. It is based on Web content accessibility analysis and the study of web content management by users in CMS environments. The framework here presented can be used to achieve four main objectives. First, to improve the accuracy of accessibility evaluation including results up to three different automatic accessibility evaluators, thus fulfilling the methodological background for accessibility evaluations provided by the W3C [6]. Second, to optimize manual accessibility evaluation process. It is performed by taking into account accessibility guidelines checked in previous manual evaluations and that still have validity when the web elements analyzed do not change from one evaluation to the next. Third, the result of the accessibility evaluation is not focused only on checking web content, it also addresses aspects related to web content management in evaluated systems. In this sense, the framework provides information about possible corrective accessibility maintenance activities for webmasters based on logs from the CMS environment. Finally, to minimize the amount of time and resources involved in evaluating accessibility. Provided benefits are expected to be useful for webmasters and human accessibility evaluators.

Presented prototype is built on a concrete CMS scenario, in this case OpenCMS. As future work it is foreseen to complete it by including more inference rules and

aspects to be taken into account for inferring possible causes for accessibility errors, like the use of external components such as web editors. It is also expected to provide support for more CMS environments.

Acknowledgements

The work described in this paper has been partially supported by Spanish Ministry of Science and Innovation through the Open Platform for Multichannel Content Distribution Management (OMediaDis) research project (TIN2008-06228).

References

[1] Abascal, J., Arrue, M., Vigo, M.: A Methodology for Web Accessibility Development and Maintenance. In: Zaphiris, P., Kurniawan, S. (eds.) HCI Research in Web Design and Evaluation. Idea Group Publishing (2006) ISBN. 1-59904-248-7

[2] Boehm, B.A.: Spiral Model of Software Development and Enhancement. IEEE Computer 21(5), 61–72 (1988)

[3] Brajnik, G.: Web Accessibility Testing: When the Method is the Culprit. In: Miesenberger, K., Klaus, J., Zagler, W.L., Karshmer, A.I. (eds.) ICCHP 2006. LNCS, vol. 4061, pp. 156–163. Springer, Heidelberg (2006)

[4] Ceri, S., Fraternali, P., Bongio, A.: Web Modeling Language (WebML): a modeling language for designing Web sites. In: Proceedings of the World Wide Web WWW9 Conference, Amsterdam (May 2000)

[5] Eurostat. Health Stadistics. Luxembourg: Office for Oficial Publications of the European Communities (2002), http://ec.europa.eu/health/ph_information/dissemination/reporting/eurostat_health2002_3_en.pdf (retrieved on June 2009) ISBN 92-894-3730-8

[6] Evaluating Web Sites for Accessibility, http://www.w3.org/WAI/eval/Overview.html (retrieved on June 2009)

[7] Gibson, B.: Enabling an Accessible Web 2.0.W4A2007 - Keynote. In: Co-located with the 16th International World Wide Web Conference, Banff, Canada, May 7-8 (2007)

[8] IBM Rational Policy Tester Accessibility (2009), http://www-01.ibm.com/software/awdtools/tester/policy/accessibility/ (retrieved on June 2009)

[9] Mendes, E., Mosley, N., Counsell, S.: The need for Web Engineer: an introduction. In: Web Engineering, pp. 1–27. Springer, Heidelberg (2006)

[10] Rainville-pitt, S., D'Amour, J.M.: Using a CMS to create fully accessible websites. In: W4A 2007- ACM International Conference Proceeding Series (2007)

[11] Royal decree number 1494/2007, by the one the regulation about basic terms for the access to technologies, products and services related with information society and social media networks by users with disabilities are approved. In: Presidency Ministry. Spanish Official State Bulletin, vol. 279, pp. 47567–47572 (2007) (in spanish)

[12] Section 508, http://www.section508.gov/ (retrieved on June 2009)