Project Vibl: A Feedback tool for Web developers to improve the accessibility of applications they build

Achin Kulshrestha
University of Wisconsin-Madison
achink@cs.wisc.edu

Chih-Ching Chang
University of Wisconsin-Madison
chih-ching@cs.wisc.edu

Akshay Uttamani
University of Wisconsin-Madison
auttamani@cs.wisc.edu

ABSTRACT

The internet is an indispensable resource in our lives. By law, websites and mobile applications need to be accessible in order to provide tantamount access and opportunity to the visually impaired. However, the percentage of accessible websites are still low. In our research, we studied the application development process of professional web developers to understand their view on accessibility. Based on our findings, we built a tool called - Vibl, which provides feedback to web developers on the accessibility of their application as they build it. We demonstrate the gamification component of Vibl as well, which has been designed to inculcate a sense of competition among developers and motivate them to create accessible websites. We propose a within-participant design to evaluate Vibl's effect on web accessibility. A questionnaire is used for subjective measurement. The result shows that Vibl has significant effect on learning web accessibility guidelines ($t = -7.9148, p < 10^{-4}$) and on easiness of adding accessibility ($t = -5.7664, p < 10^{-3}$).

Author Keywords

accessibility; web application; web standards; visually impaired; accessibility guidelines; tools; developers perception; survey

INTRODUCTION

With the growing use of internet and the wealth of information it provides , the user population of the web is getting more and more diverse. Users of all ages, educational levels, and levels of computing experience are now using it for their day-to-day activity. What we often ignore is the presence of users that have some type of disabilities. We need to understand that users with disabilities can only utilize a web site if it is designed to be compatible with the various assistive technologies. A web site that is sufficiently flexible to be used by all of these assistive technologies is called an accessible web site.

A large number of web accessibility tools exist, but are they meeting the needs of the web application developers? With accessibility standards and the challenges posed by highly dynamic Web-based applications, accessible Web design is as challenging as ever. For developers to be able to make their application accessible, they need to update their skills in coding and testing. Tools have an important role to play in this process. It is observed that the earlier accessibility problems can be identified, the easier they can be fixed. Considering this challenge for the developers we came up with an idea of developing an accessibility tool Vibl which provides feedback to web developers on the accessibility of their application as they build it. This we believe could make it easier for them to make their applications accessible. To analyze the usefulness of our tool, a survey of web application developers was conducted in two phases, and is presented here. The first phase of the survey provides an insight into the problems experienced by developers in creating accessible applications with existing tools, and their opinions on the value of specific tool features. Based on the result of this survey, we built the accessibility tool Vibl. The second phase of the survey was conducted to understand the usefulness of our tool in making their applications accessible.

RELATED WORK

A survey was conducted by Rosson et al. to study the popularity of accessibility among web developers[1]. The survey was based on interviewing more than 300 professional web developers. From this study, while developers were quite aware of accessibility and usability issues, only 5% of respondents tested their applications of accessibility. Rosson et al. gave the hypotheses that developers did not incorporate accessibility because they may lack of awareness they may feel the supporting checking tools were tedious and time consuming. To understand the effectiveness of existing accessibility tools, Petrie et al. [2] conducted a study where they observed the that the usability of these tools were poor and most importantly, these tools did not enhance designers' understanding of accessibility. They concluded the study by saying that it is very important for such tools to fit properly in the web development ecosystem and that the existing tools do not meet this requirement. Similar to the survey of Rosson et al., Lazar et al. [3] also conducted a survey to understand the major challenges in incorporating accessibility features. They surveyed the consciousness towards accessibility of 175 professional web developers who worked in different departments and observed that around 65% of the developers had created an accessible Web site. The respondents reported that balancing accessibility and graphical design, convincing clients(since they were not totally aware of it) and technical challenges to make sites accessible, were the biggest crux of the problem. The respondents also said that there was no effective tool that escalated their process. Within this sample, 61% had used a free web-based accessibility tool, 22% had used a non- Web based tool, and 40% had tested with a screen reader. In the research done by Shari et al. [4], they concluded from their survey that the existing tools are unclear and incomplete as well. The respondents of the survey all agreed that a checklist of automatically detected problems and explanation would help a lot during the design process.

METHOD

Our study design included three parts. First, we interviewed professional web application developers to understand their view on accessibility. We interviewed the developers at their workspace face-to-face or on the Internet, taking notes and recording the interview simultaneously for reference. Second, based on the qualitative data we collected from the developers and previous related work, we summarized the current challenges of web accessibility and designed our tool referring to the current problems. Last, in order to test the effectiveness of our tool, we asked the same developers to use our tool and gave them an online questionnaire to collect their subjective feedback. Our design for the evaluation is a single factor within-participant design with two levels: before using Vibl and after using Vibl. There was no time restriction for participants to answer the questionnaire.

PROCEDURE

In this section, the procedure of our experiment is described in detail. Based on the interview of professional web developers, we decided to implement the tool as a chrome extension. All participants were asked to fill the questionnaire first without seeing any function or interface of Vibl (no matter how much experience about web accessibility they had). The questions in the survey provided link to w3c website guidelines that they could refer to, if they needed them to answer the questions. The questionnaire had three different categories: learning, motivation, and easiness of web accessibility (guidelines). After finishing the first part, participants were asked to watch a short video about Vibl. The content of the video was a demonstration of all functions in Vibl. A screen record of us using Vibl and human voice explanation of the functions was provided in the video. The second phase of the survey was conducted to understand the usefulness of Vibl after they finished watching the video. Participants could go to the previous page to watch the video again if they needed to. They would answer the same questions as in the first part but under the condition of knowing Vibls functions and features. A few helping messages of Vibls functions were provided alongside the questions for their reference. This whole process had no time restriction. All questions asked about participants subjective opinions in the scale of 1 to 5. See appendix for our questionnaire.

PARTICIPANTS

Professional web developers (7 males , 1 female) were recruited from online and personal invitation for the interview part. Participants were either our friends with professional experience or web developers that work in Madison, WI. Some participants already had knowledge and experience of implementing web accessibility; others had knowledge but did not have much experience in it. Ten web developers, including the original interviewees, were recruited to do the survey for Vibls evaluation.

MEASUREMENT

Measurements were in the format of questionnaire. Participants gave their personal opinion in the scale of 1 to 5. There were 18 questions to evaluate the effectiveness of Vibl, in terms of the easiness of learning accessibility guidelines, the motivation of adding accessibility, and the easiness of implementing accessibility.

RESULT

Phase-I: Web Developers' Interview

After interviewing 8 different professional web developers, we inferred that lack of awareness was not a major issue. All of them were aware of it but not all had an experience of developing a completely accessible web application. Few of the problems faced by them are listed below :

- Lack of proper feedback: the developers felt that even if they implemented the accessibility features, they could not be sure if they got it right or not, since their was hardly any feedback from the disabled users.
- Lack of tools to check for accessibility issues: most tools they felt, do not fit well into their web development process.
- Lack of motivation to add extra effort in making the application accessible: they felt that there was no satisfactory outcome even when they tried incorporating the features.
- Lack of awareness about Legal obligations: 7 out of 8 developers did not know about the Section 508 law, which makes it mandatory for web developers to incorporate accessibility.
- Difficulty in understanding the guidelines: since there are so many of them, they find it hard to figure out which rule/guideline they are missing in their application.
- Hard deadlines and lack of time in an industry setup : since most of their projects have a strict deadline, they felt there was no requirement of adding extra features.
- No proper Lint rule or Chrome extension for accessibility checks: they were unsatisfied by the existing tools.

On being asked how we can make people more aware, one of the respondents said "I think that we need to hold more events/conferences that are regarding accessibility to let more people aware of the need in this field." Some of them also suggested that adding gamification in the tool might help them motivate to make their application accessible. Considering all the stated problems and based on existing Google developer tools for accessibility[4], we built our tool- Vibl , which fits easily in the web development cycle and includes features like gamification (for motivation) and easy check for accessibility issues. It also provides a proper feedback to the developer to fix the issues.

Phase-II: Tool evaluation

Factor analysis and paired t-test were conducted to analyze the collected data. We first obtained the most important questions and scales from factor analysis. After that, we used paired t-test to compare the results, categorized in three different scales, before and after seeing Vibls functions.

Questionnaire result

Ten data sets were gathered throughout the experiment, and a factor analysis was conducted. Four out of eighteen eigenvalues were above the value of 1. The number of factors extracted could be seen in the Screeplot in Figure 1. The infection point was about five in the Screeplot. Thus, the number of factors chosen for further analysis was five.

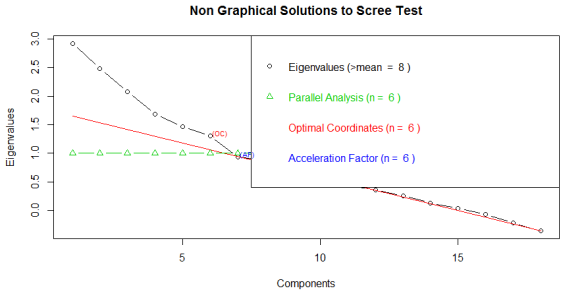

Figure 1. Screeplot result: 5 factors were chosen to retain.

A factor analysis was conducted using principal axis factor extraction and varimax for rotation method. The result for factor analysis can be seen in Figure 2. The reported Cronbachs Alpha shows good internal consistency that all three scales have value over 0.7. The scales constructed were statistically viable, and the questions loaded on the scales were shown to be reasonable. The result can be seen in Table 1.

Table 1. Cronbachs Alpha for questions chosen in each of the three scales.

Scale	Questions chosen	alpha
Learning	Q11,Q14,Q16	0.9
Motivation	Q21,Q22,Q24	0.7
Easiness	Q32,Q35,Q36	0.86

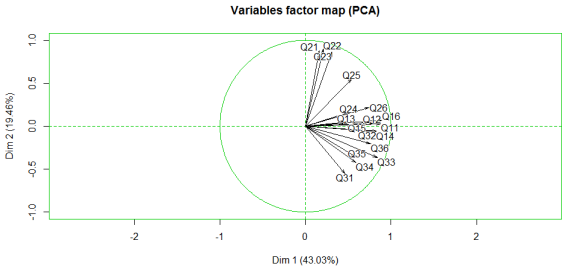

Figure 2. Result of factor analysis. The meaning of each label: Q11 to Q16 for questions in "Learning", Q21 to Q26 for questions in "Motivation" and Q31 to Q36 for questions in "Easiness".

From the factor analysis, we chose three most important items for each scale for further inferential statistical analysis. The descriptive statistical result of each scales we constructed can be seen in Figure 3. Figure 3 gives the overall score distribution for both before and after cases. The detailed number of the result can be seen in Table 2.

A Pearson correlation coefficient was computed to assess the relationship among the three dependent variables. There was a significant positive correlation between the two variables,

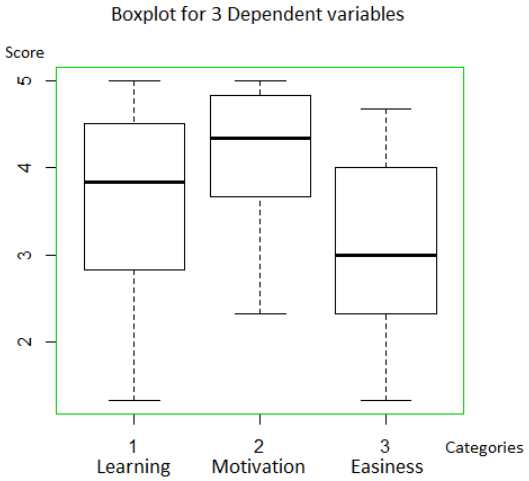

Figure 3. Boxplot for the score distribution of each dependent variable.

Table 2. Descriptive Analysis: the distribution of each variables.

	mean	sd	median	min	max
Vibl*	1.5	0.51	1.5	1	2
Learning	3.57	1.07	3.83	1.33	5
Motivation	4.23	0.69	4.33	2.33	5
Easiness	3.07	0.99	3	1.33	4.67

Learning and Easiness, r = 0.72, n = 20, p = 0.005. The result of Pearson correlation can be seen in Table 3.

Table 3. Pearson correlation of each scale.

		learning	motivation	easiness
Learning	r	1	0.29	0.72
	p		0.2101	0.0004
	N	20	20	20
Motivation	r	0.29	1	0.12
	p	0.2101		0.611
	N	20	20	20
Easiness	r	0.72	0.12	1
	p	0.0004	0.611	
	N	20	20	20

A within-subjects t test was conducted to determine the effect of Vibl on learning, motivation and easiness respectively. Using a one-tailed .05 criterion, we succeeded to reject the null hypothesis in learning and easiness. There is significant difference between learning without Vibl (M = 2.8, SD = 0.89) and with Vibl (M = 4.33, SD = 0.57), t(9) = -7.9148, p = 2.4e-5. The difference between easiness without Vibl (M = 2.33, SD = 0.75) and with Vibl (M = 3.80, SD = 0.55) is also significant, t(9) = -5.7664, p = 0.00027. The detailed result for t test is shown in Figure 4 and Table 4

Table 4. Result for paired t-test, comparing each dependent variable under the condition "without" and "with" Vibl.

	before (without)		after (with)			
	M	SD	M	SD	t-test	p-value
Learning	2.8	0.89	4.33	0.57	-7.91	2.41E-05
Motivation	4.13	0.79	4.33	0.61	-1.15	0.2789
Easiness	2.33	0.75	3.8	0.55	-5.77	0.000271

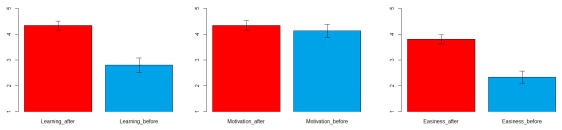

Figure 4. Result of t-test. From left to right are "Learning", "Motivation", and "Easiness". Red color indicates the result with (after) Vibl. Blue color indicates the result without (before) Vibl.

CONCLUSION

This survey of experienced and professional web developers showed that while developers are aware of web accessibility, not all of them try to incorporate accessibility features in their web applications. There are a lot of factors that make accessibility less popular among web developers but we observed that the main cause was a lack of properly structured guidelines, which are less verbose and easy to understand. Developers depend on accessibility tools to guide them but using the existing tools is difficult and time-consuming. Using these tools reliably and effectively requires accessibility expertise. Another highly influencing factor was lack of motivation and feedback. Even when developers are motivated to provide accessible applications, they are hampered by new standards, pressure to meet deadlines and inconsistencies in assistive technologies.

Keeping all these factors in mind, we developed our tool 'Vibl' and found that it significantly improved learning through simple guidelines and ease of use. Developers found it easy to find and fix the accessibility issues in their application using Vibl. Vibl also had a gamification component: each devloper was given a score based on the number of guidelines they had correctly follwed and the ones that they missed. A marginal improvement in motivation was seen with this gamification component. This hints towards improving the gamification component, that would eventually motivate the developers to make their application more accessible. We also got a couple of useful feed-backs from the developers: they demanded documentation along side every error raised by Vibl. They also suggested that having a lint rule to support development in a team based setup, will make their process much smoother and easier. This survey indicates that while developers are ready to incorporate the accessibility features, a tool like 'Vibl' helps and motivates them to make the applications accessible. Vibl easily fits in their development cycle and allowed them to fix the issues in a very short time.

FUTURE WORK

While our tool 'Vibl' showed improvement in ease of use over existing tools, there is still a lot work to be done to help the developers make their applications accessible. Ease of understanding is important but the information provided by the tool is equally important. Features like displaying line of code for each accessibility failure might make it more useful. As inferred, gamification can help increase the motivation, so having intelligent gamification support in the tool might prove helpful. Examples for each guideline can be used in the tool which will make it easier for the developers to fix the issue. Finally, a lint rule for developers along with this tool might make it extremely easy for them to meet the accessibility requirements. Future research should also examine the use of each feature in more depth, and also examine the perceptions of web accessibility held by other stakeholders, such as managers and clients.

ACKNOWLEDGMENTS

Firstly, we would like to express our sincere gratitude to Prof. Bilge Mutlu for his continuous support, motivation, and immense knowledge. We thank all the developers who participated in the survey, and those who helped to disseminate the survey. We would specially like to thank the people at Practicum for letting us interview their awesome team of professional web developers and their valuable feedback for the Vibl tool.

REFERENCES

[1] Rosson, M., Ballin, J., Rode, J. and Toward, B. Designing for the Web. Revisited. A survey of informal and experienced Web developers. In D. Lowe and M. Gaedke (Eds) ICWE 2005. LNCS 3579. Springer-Verlag Berlin Heidelberg, 2005, 522-532.

[2] Petrie, H., King, N., Velasco, C., Gappa, H. and Nordbrock, G. The usability of accessibility evaluation tools. In C. Stephanidis (Ed.) Universal Access in HCI, Part III, HCII 2007, LNCS 4556,. Springer-Verlag Berlin Heidelberg, 2007, 124-132.

[3] Lazar, J., Dudley-Sponaugle, A. and Greenidge, K. Improving Web accessibility: A study of webmaster perceptions. Computers in Human Behavior 20 (2004), 269- 288.

[4] Trewin, S., Cragun, B., Swart, C., Brezin, J., Richards, J. Accessibility Challenges and Tool Features: An IBM Web Developer Perspective. In Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A) (W4A '10).

[5] Google developer tools for accessibility, Google accessibility team `https://github.com/GoogleChrome/accessibility-developer-tools`

APPENDIX

[1] Questionnaire: `http://goo.gl/forms/iyJrmhZT6K`