

A WYSIWYG Editor to Support Accessible Web Content Production

Hedi Carlos Minin¹, Javier Jiménez Alemán¹,
Carolina Sacramento^{2,3(✉)}, and Daniela Gorski Trevisan¹

¹ Universidade Federal Fluminense, Niterói, Brazil
{hminin, jjimenezaleman, daniela}@ic.uff.br

² Fundação Oswaldo Cruz, Rio de Janeiro, Brazil
carolina.sacramento@uniriotec.br

³ Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, Brazil

Abstract. In a world where lay users on web languages and standards are responsible to produce content to web, it's essential the presence of tools which support the creation of accessible content. This paper proposes to make Web accessibility concepts more understandable to these users with the incorporation of WCAG 2.0 accessibility guidelines in HTML WYSIWYG editors they use. For that we designed and prototyped such Editor and performed preliminaries usability tests with target users. Results shown that accessibility warnings were easy to understand and to apply but difficult to perceive them.

Keywords: Accessibility guidelines · WYSIWYG HTML editor · WCAG 2.0 · ATAG 1.0

1 Introduction

The main advantage of Web 2.0 is the possibility of Web content creation by lay users. In this context, we can find authoring tools such as CMS (Content Management Systems) which allows the publication of content without knowledge about languages, web standards or accessibility guidelines. On the other hand, initiatives such as WCAG 2.0 [1] (Web Content Accessibility Guidelines 2.0) and ATAG 1.0 [2] (Authoring Tool Accessibility Guidelines 1.0) can guide developers in creating authoring tools that produce accessible Web content for disabled people.

Even with these guidelines and specific laws for accessibility in many countries, recent researches indicate that adherence to accessibility standards on web sites is still low [3–5]. According to these, although the version 2.0 of WCAG has been designed to make the accessibility guidelines more understandable and testable, non-experts and even experts in web accessibility have found problems in use them. Brajnik *et al.* [3] demonstrated that experienced evaluators differed in interpreting the success criteria based in the WCAG 2.0.

Our work proposes to make Web accessibility concepts more understandable to lay users that produce Web content. To do that we suggest the incorporation of accessibility guidelines in HTML WYSIWYG editors. The HTML WYSIWYG (pronounced “wiz-ee-wig”) - an acronym for “what you see is what you get” - editor is a tool built-in

the CMS that allows users to create content without HTML code knowledge and shows to these users exactly how the content should appear on screen.

For a better understanding of the objectives and results of this study, this article is organized as follows: Sect. 2 presents the related work and the advantages of the proposed concept in relation to the state-of-the-art in the field. Section 3 describes the methodology used in the design of the proposed HTML WYSIWYG editor interface. Section 4 presents the results of a preliminary evaluation with target users and, finally, in Sect. 5, are shown the conclusions and future works.

2 Related Work

Accessibility guidelines is a theme often approached by researchers of the Human-Computer Interaction area.

Researches show that adherence to accessibility standards is still low due to its complexity [3–5]. Even experts in web accessibility have difficulty understanding the guidelines. Brajnik *et al.* [3] demonstrated that even experienced evaluators differed in interpreting the success criteria based in the WCAG 2.0.

Power *et al.* [6] developed an empirical study of the problems encountered by 32 blind users that evaluated a set of 16 sites. One of evaluation result denoted low degree of guidelines implementation in the evaluated sites, which made the researchers concluded that web developers still have to make great efforts to create accessible sites, possibly due to the low understanding the guidelines or the lack of support tools.

Concerning evaluation of authoring tools, Bittar *et al.* [7] evaluated 5 desktop tools (not Web) about support they offer to Web developers in the implementation of accessibility standards. In this work were evaluated tools as Adobe Dreamweaver, Eclipse and Netbeans, which support the developer on site construction but not in content producer. The authors selected relevant guidelines from ATAG1.0 and WCAG 2.0 and evaluated the adherence of the tools. Although the work is not related to HTML WYSIWYG editors, provides an analysis methodology for the WCAG 2.0 and ATAG 1.0 standards in authoring tools.

Lopez *et al.* [8] presents a methodology for identifying and resolving Web accessibility issues in Content Management Systems (CMS). The methodology proposes the production of an HTML document using the CMS and then validate it with rules of WCAG and ATAG. The article mentions that HTML editors do not consider some aspects of accessibility and users can manage content without considering accessibility.

Iglesias *et al.* [9] compares the ability to create accessible content according to WCAG and ATAG guidelines in three different learning environments (Moodle, ATutor and Sakai). They concluded that, in practice, the creation of accessible content depends more of the experience and knowledge of the user on Web accessibility.

Developers of the Portal of Casa de Oswaldo Cruz/Oswaldo Cruz Foundation [10] and winners of Brazilian National Accessibility Award¹ [11], concluded, with their

¹ Awards organized by Brazilian government and W3C's local office to encourage the development of accessible websites.

experience, that is a challenge keeping site content accessible. According to them, more difficult than building an accessible site is to keep the same standard of accessibility over time, especially when the content producer has little or no knowledge of Web languages and standards, including accessibility [12].

This entire scenario reinforces the relevance in development of tools to support the construction of accessible content, especially considering the complexity of the WCAG 2.0 guidelines in contrast to the strong presence of lay users on creating web content.

3 Methodology

The study was conducted in three main stages. The first stage called Context Analysis, focused on the accessibility evaluation of the most used HTMLs Editors and its target audience. In the second, called Design, we developed a prototype in JavaScript language, considering the key features observed in the first stage. Finally, in third step, we conducted a usability evaluation in the prototype developed. All stages will be detailed below.

3.1 Context Analysis

In order to better understand the typical users of WYSIWYG editors, we designed an online questionnaire composed mainly of closed questions.

The questionnaire was answered by 15 people, including 8 women and 7 men. All participants had between 16 and 49 years old. They were asked about the Content Management System (CMS) used and results shown that the Joomla! and Wordpress were the most mentioned (80 % of responses). The user experience on this CMS's was considered medium by 86 % of participants. The main HTML editor features used by participants can be observed in Fig. 1. Although all participants have declared interest in producing Web content accessible, 56 % reported not to know accessibility guidelines. Finally, 87 % of people considered it relevant that the HTML editor could assist and provide help about how to create accessible Web content.

Next, we analyzed the adequacy of the main HTML WYSIWYG editors to WCAG 2.0 and ATAG 1.0 Guidelines. The objective was to know how HTML editors meet the accessibility requirements. We considered from the questionnaire responses that HTML WYSIWYG editors built-in in the CMSs are more accepted. By default, the editor installed in Joomla! is the TinyMCE and Visual Editor is the Wordpress default editor. Besides these, the CKEditor was also evaluated because it can be incorporated in both mentioned CMS and it is considered one of the most accessible WYSIWYG editors according to an informal web search [13–15].

After defining which WYSIWYG editors will be evaluated, we analysed the WCAG 2.0 and ATAG 1.0 Guidelines. In this study, we have identified some accessibility guidelines, those that are considered relevant for the production of Web content. For the WCAG 2.0, we selected only success criteria that the WYSIWYG editor could use to help the user to comply with them. For the ATAG 1.0 were selected guidelines related to the ability to produce accessible content by the WYSIWYG editor.

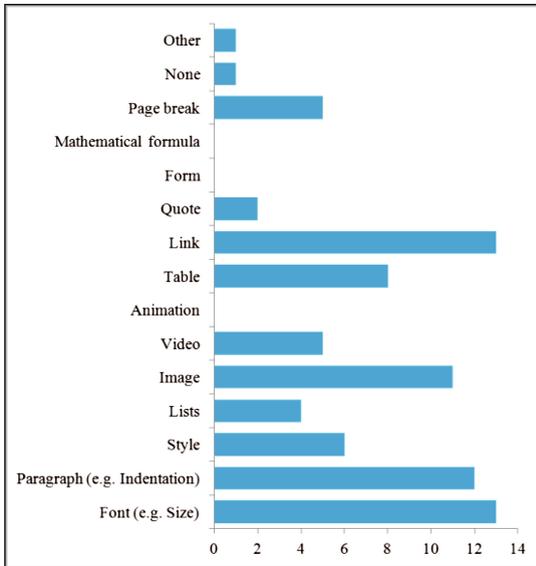


Fig. 1. HTML WYSIWYG editor resources used by participants of survey

For each question, the answers were classified as “No agreed”, “Partially agreed” or “Fully agreed”. This standard was used by [16], when they analyzed the accessibility of computer games. Tables 1 and 2 show the results of the evaluation of WYSIWYG editors.

CKEditor showed the best results with respect to the support and the production of accessible content (WCAG 2.0: 60 % and ATAG 1.0: 40 %). As this is an open source application, maintained by an international community [17], we considered the possibility of including accessibility features in this editor. However, due to time for implementation and the lack of qualified staff to customize WYSIWYG editor’s source code, we decided to develop an independent prototype. Our proposal is that accessibility features could, subsequently, be incorporated into others HTML WYSIWYG editors.

3.2 Design

The prototype was developed in *JavaScript* and HTML languages considering the key features observed in the previous questionnaire. It can be viewed at: <http://www.ifrocolorado.com.br/uff/taes/> (only in Portuguese).

It was designed to show, in real time, if a specific accessibility guideline was violated. The current prototype version is taking into account images, tables and titles tags checks. When the user is inserting content not according to the expected standard the WYSIWYG editor shows alerts at the interface including contextual examples to help content producers better understand and solve the problem. Figure 2 shows accessibility guidelines provided by the prototype.

Table 1. Editors' evaluation with the WCAG 2.0 success criteria (X = No agreed, V = Partially agreed, VV = Fully agreed).

Level	Success criteria	TinyMCE	CKEditor	Visual editor
A	Has features to insert alternate text (alt) into images? (CS 1.1.1)	V V	V V	V V
	When inserting an image, the alt attribute is inserted with a blank value (alt = "") when its content is not specified? (CS 1.1.1)	V V	V V	X
	Has resources to insert long descriptions (longdesc) in complex images? (CS 1.1.1)	X	X	X
	It checks whether semantic markup titles (H1, H2,...) is properly used? (CS 1.3.1)	X	X	X
	It uses correctly semantic markup (blockquote) in quotes? (CS 1.3.1)	V V	V V	V V
	Has the resources to set headers in tables and relate them to their content? (CS 1.3.1)	X	V	N/A
	Has the resources to enter summary (summary attribute) in tables? (CS 1.3.1)	X	V V	N/A
	Has the resources to incorporate anchors in the content? (CS 2.4.1)	X	V V	X
AA	Has the resources to set the language (lang) in quotes? (CS 3.1.2)	X	V V	X
AAA	Has the resources to insert abbreviations (abbr)? (CS 3.1.4)	X	X	X

Table 2. Editors' evaluation with ATAG 1.0 checkpoints (X = No agreed, V = Partially agreed, VV = Fully agreed).

Level	Checkpoints	TinyMCE	CKEditor	Visual editor
A	Does not the tool generate equivalent alternatives automatically? (PV 3.4)	V V	V V	X
	Does the tool automatically generate markup (HTML) valid? (PV 2.2)	V V	V V	V V
AA	Does the tool support the production of content through context-sensitive help or documentation on creating accessible content, including examples? (PV 6.2)	X	X	X
AAA	Does the tool report the author if the marking is not produced according to the W3C specifications. (PV 2.3)	X	X	X
	Does the tool provide the author a summary of the status of the document accessibility? (PV 4.4)	X	X	X

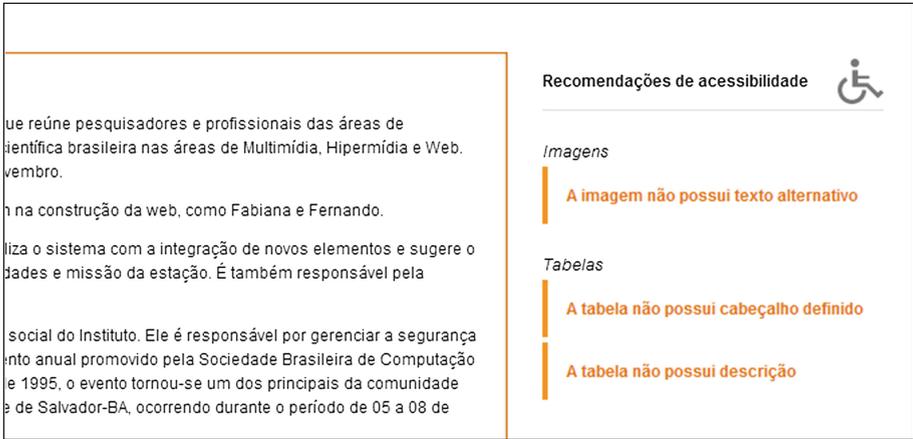


Fig. 2. Example of accessibility guidelines provided by WYSIWYG editor prototype (in Portuguese).

For the images, our prototype analyses if there is an alternative text, which is the text alternative for non-textual elements on a Web page. Every time an image is added without alternative text, the editor displays a warning to fill the text.

Tables of data should have headers and description. For non-blind users, the information contained in tables of data are easily understood visually, across columns and rows. However, for blind people, understanding and obtaining data from a table isn't an easy task [18]. In this context, headers to identify the contents of a given table cell containing a description of its purpose and general structure are fundamental. The editor prototype alerts the user to the absence of these information.

And, finally, for the title tags, our prototype analyses whether the levels of title were specified correctly and alert if there is any hierarchy violation. For example, if a user selects a level 3 title tag in the middle of a text only to create a paragraph with letters larger than the other, without having defined title tags levels 1 and 2 before, the prototype will warning.

For all three situations, help resources will be provided with examples of how to meet the guidelines. Figure 3 illustrates the help feature relating to the image description.

At the end of the creating content process, if user decided for not adjust the items indicated by the editor and try to save the work, a warning appears on the screen asking if the user really wants to save without correcting accessibility issues detected.

3.3 Evaluation

Usability tests was used as evaluation methodology, from Think-Aloud technique proposed by [19]. The objective of the evaluation was assess whether the accessibility features provided by the prototype helped the content producers on creation of accessible Web content.

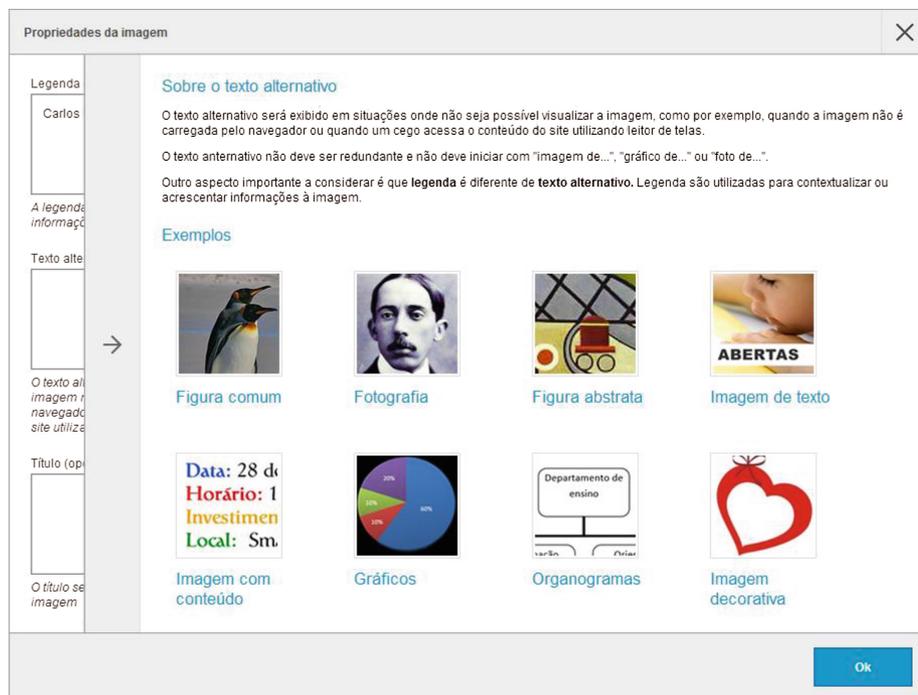


Fig. 3. Help feature with examples of alternative text (in Portuguese)

Five persons participated of usability test: 3 men and 2 women with an average age of 33 years, among them journalists and secretaries. Four of five participants self-declared as graduate degree and reported having more than 5 years of experience with web content production. The participants highlighted that “font”, “tables” and “images” are the more used HTML WYSIWYG editor’s resources by them. Three of five participants indicated do not have knowledge of Web content accessibility guidelines.

The test consisted of three tasks related to creation of textual content, content with images, texts with table of data and texts with well defined titles hierarchy. In every task, the participant should insert a pre-defined content in the editor. When the texts were long or had associated images, it was supplied by an appraiser to facilitate the activity execution. It was expected that participants would use the accessibility guidelines supplied by the editor to create accessible content, so these guidelines were not included in the material provided by appraiser. Upon completion of each task, the user was directed to press the “Save” button and call the appraiser.

After the usability testing, users completed a satisfaction questionnaire which contained a set of questions based on the System Usability Scale methodology (SUS) [20] and other questions related to the editor accessibility features.

4 Discussion of Results

Usability tests have shown low rate of tasks conclusion. On average, only 53 % of the tasks performed were completed with the desired effect, that is, taking into account the accessibility guidelines applied to content.

The main cause of this result was the low perception of accessibility guidelines by users in the Editor. Those tasks where accessibility guidelines were displayed on the screen (images and tables) only two users perceived the warnings, demonstrating the need for review of how warnings are shown on interface.

Regarding the understanding of recommendations, only one of the two users that perceived the warning performed the accessibility recommendation on image insertion task. In the task of editing table data, the two users who perceived the warning on the editor made corrections. It was possible to see a trend of people more familiarized with the accessibility guidelines (two users) have less difficulty in recognizing and adjust the questions posed by the editor. However it is necessary to expand the tests to users with different profiles to prove this trend (users who are familiarized with accessibility guidelines and users who are not).

The tests also showed that users tend to follow the same procedure that are familiarized in the HTML WYSIWYG editors that use in their day-to-day. We observed only 2 hits, on the entire test, to access the editor help feature.

We also noted that usability problems in the editor prototype have influenced negatively the tasks performance. In total were perceived 2 noises, e.g. problems that cause decreased user performance on the task, 5 obstacles, e.g. problems that the user experiences a few times but eventually overcome them and 2 barriers, that means problems that user can't overcome. Because of these problems, 20 % of the task executions were not completed.

The post-test questionnaire indicated a level of satisfaction of 68.1 points in the SUS scale, which is considered slightly below the average of 68.2 points to Web interfaces, according to the Usability Professionals Association [21]. Participants who perceived accessibility guidelines classified as satisfactory the quality of guidelines and stated that the recommendations brought much contribution to their learning on Web accessibility theme.

5 Final Remarks and Future Works

Taking into account that 40 % of users did not perceive the accessibility recommendations it is necessary to rethink the way that the editor is delivering the warnings at the interface. The authors believe due the fact that the equipment used in the test has widescreen may have contributed to this result. At this point, is needed a further investigation of how the user's visual attention behaves in HTML WYSIWYG editors.

Even with the low percentage of recommendations perception, 75 % of users who noticed the warnings performed the corrections proposed. This finding indicates that the editor tends to meet its goal of supporting the accessible Web content creation. However, we point out that these corrections were made by users who knew, even superficially, the WCAG 2.0 recommendations.

In order to improve the tests results we are adjusting the usability problems detected to submit it to new tests, increasing the number of participants and dividing them into two distinct groups: those without knowledge about WCAG 2.0 recommendations and those that know the recommendations.

As future work, we want to expand the items checked by the editor, incorporating new accessibility guidelines and trying different kinds of alerts, with sounds and blinks for instance. Also, we intended to make the editor available as a plugin for others HTML WYSIWYG Editor and to expand the HTML editor accessibility for disabled people, since the current prototype does not consider the use by this public.

Acknowledgement. The Brazilian Students-Agreement Program Graduate (PEC-PG).

References

1. W3C: Web Content Accessibility Guidelines 2.0 (2008). <http://www.w3.org/TR/WCAG20/>
2. W3C: Authoring Tool Accessibility Guidelines 1.0 (2000). <http://www.w3.org/TR/WAI-AUTOOLS/>
3. Brajnik, G., Yesilada, Y., Harper, S.: Is accessibility conformance an elusive property? a study of validity and reliability of WCAG 2.0. *ACM Trans. Access. Comput.* **4**(2), 28 (2012). Article 8
4. Lazar, J., et al.: Potential pricing discrimination due to inaccessible web sites. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) *INTERACT 2011*, Part I. LNCS, vol. 6946, pp. 108–114. Springer, Heidelberg (2011)
5. Hanson, V.L., Richards, J.T.: Progress on website accessibility? *ACM Trans. Web* **7**(1), 30 (2013). Article 2
6. Power, C., Freire, A., Petrie, H., Swallow, D.: Guidelines are only half of the story: accessibility problems encountered by blind users on the web. In: *SIGCHI Conference on Human Factors in Computing Systems (CHI 2012)*, pp. 433–442. ACM, New York (2012)
7. Bittar, T.J., Amaral, L.A., Faria, F.B., Fortes, R.P.M.: Supporting the developer in an accessible edition of web communications: a study of five desktop tools. In: *Workshop on Information Systems and Design of Communication (ISDOC 2012)*, pp. 3–9. ACM, New York (2012)
8. López, J.M., Pascual, A., Menduiña, C., Granollers, T.: Methodology for identifying and solving accessibility related issues in web content management system environments. In: *The International Cross-Disciplinary Conference on Web Accessibility (W4A 2012)*, Article 32, 8 p. ACM, New York (2012)
9. Iglesias, A., Moreno, L., Martínez, P., Calvo, R.: Evaluating the accessibility of three open-source learning content management systems: a comparative study. *Comput. Appl. Eng. Educ.* **22**(2), 320–328 (2014)
10. Casa de Oswaldo Cruz. <http://www.coc.fiocruz.br/>
11. Todos @Web: Prêmio Nacional de Acessibilidade. <http://premio.w3c.br/2013/>
12. Sacramento, C.C., Silva, C.S., Conceição, F.I., Moraes, F.L.D., Castro, I.M., Zorzanelli, L. P., Barros, Y.S.: Portal da Casa de Oswaldo Cruz e o Prêmio Nacional de Acessibilidade na Web. In: *13th Brazilian Symposium on Human Factors in Computing Systems. Brazilian Computer Society, Foz do Iguaçu* (2014)
13. 10 Best WYSIWYG Text and HTML Editors for Your Next Project. <http://www.1stwebdesigner.com/design/10-best-wysiwyg-text-and-html-editors-for-your-next-project>

14. Top 5 Latest Javascript and HTML5 WYSIWYG Advanced HTML Editors. <http://www.typeerror.com/index.php/top-5-latest-jquery-wysiwyg-advanced-text-editors>
15. 10 Best Free Javascript/jQuery WYSIWYG HTML Editors. <http://medleyweb.com/resources/10-best-free-javascript-or-jquery-wysiwyg-html-editors>
16. Porter, J.R., Kientz, J.A.: An empirical study of issues and barriers to mainstream video game accessibility. In: The 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2013), Article 3, 8 p. ACM, New York (2013)
17. About CKEditor. <http://ckeditor.com/about>
18. Modelo de Acessibilidade em Governo Eletrônico. <http://emag.governoeletronico.gov.br/>
19. Lewis, C.: Using the 'thinking aloud' method in cognitive interface design. Technical report, IBM, RC-9265 (1982)
20. Brooke, J.: SUS: a quick and dirty usability scale. In: Jordan, P.W., Thomas, B., McClelland, I.L., Weerdmeester, B. (eds.) Usability Evaluation in Industry, pp. 189–194. CRC Press, London (1996)
21. Bangor, A., Kortum, P., Miller, J.: Determining what individual SUS scores mean: adding an adjective rating scale. *J. Usability Stud.* **4**(3), 114–123 (2009)