Analysis of an Automatic Accessibility Evaluator to Validate a Virtual and Authenticated Environment

Elisa Maria Pivetta, Carla Flor, Daniela Satomi Saito, Vania Ribas Ulbricht
Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento
Universidade Federal de Santa Catarina (UFSC)
Florianópolis, Brazil

***Abstract*—This article's objective is to analyze an automatic validation software compatible with the guidelines of Web Content Accessibility Guidelines (WCAG) 2.0 in an authenticated environment. To the evaluation it was utilized as a test platform the authenticated environment of Moodle, which is an open source platform created for educational environments. Initially, a brief conceptualization about accessibility and the operation of these guidelines was described, and then the software to be tested was chosen: the WAVE. In the next step, the tool's operation was valued and the study's analysis was made, which allowed the comparison between the testable errors of WAVE with the guidelines of WCAG 2.0. As the results of the research, it was concluded that the tool WAVE obtained a good performance, even though it did not include several guidelines of WCAG 2.0 and did not classified the results within the accessibility's principles of Web Accessibility Initiative (WAI). Also showed itself more adequate to developers than to common users, which have no knowledge of Web programming language.**

Keywords—automatic validation tool; WCAG 2.0; accessibility; WAVE)

I. INTRODUCTION

Web accessibility refers to the capacity of people, regardless of their skills, to perceive, understand and execute activities of navigation and interaction, as well as create contents in web [1]. The accessibility problems in the web affect people with deficiency, being it visual, motor, hearing, cognitive, language, neural system disturbs and others. Although not only people with deficiency need accessibility. Elderly, temporary deficiencies and people in general need accessible environments.

In Brazil, the decree 5.296, published in December 2004, makes mandatory the accessibility in websites of the public administration for the use of people with deficiencies with the objective of ensure full access to the contents available in the web[2].

The inclusion of people with deficiency in the educational, professional and social ways, besides of being mandatory by the Brazilian legislation, is also a social justice act, and provides an independency perspective to the individuals when communication and interaction barriers are diminished.

According to a research made in 2012 by W3C.br/NIC.br, only 2% of government web pages are accessible [3]. In international level, the World Wide Web Consortium [1] has an accessibility working group, which was created to discuss and plan acts in favor of the accessibility in web.

In scope of online education, the IMS Global Learning Consortium [4] conceptualizes accessibility as "the ability of adjust the learning environment to the necessities of all students". This accessibility can be determined by the flexibility of the environment and the availability of contents and alternative activities. In order to guarantee an effective access to Distance Education, the decree n. 5.622 of December 2005, has in the II item of the Article n. 13, that the pedagogical projects of distance courses and programs must offer appropriated treatment to students with special necessities [5]. An appropriated treatment also implies the offer of a technologic structure; in other words, an accessible Virtual Environment of Education.

The Virtual Environments of Education are systems based on a collaborative approach to the creation, application and management of courses that use the Internet. By having these characteristic, they present to have inclusive elements and they are largely utilized also as a support to the presential education. An example of a Virtual Environment of Education is the Moodle platform (Modular Object-Oriented Dynamic Learning Environment), which is the most known and utilized open-source environment in the world [6].

The Web Accessibility Initiative presents guidelines and recommendations to provide access and egalitarian opportunities to people considering the several types of skills in digital environments. Within these guidelines, WAI recommends as a preliminary revision the use of accessibility evaluation tools to identify possible problems that occur in a website. There are several ways to verify the accessibility in Virtual Environments of Education. One of them regards the automatic evaluators, which are softwares that test virtual environments by analyzing the code to verify if those are in conformity with the accessibility guidelines selected to the inspection. Nowadays, the prepositions of WAI, more specifically the Web Content Accessibility Guidelines 2.0 [7], are important references when discussions about web accessibility are raised [8].

Within the accessibility context, searching for an automatic tool that proposes evaluate authenticated environments, this article had as an objective to analyze one of these tools in relation to the virtual environment of education Moodle.

The selected software to the analysis is nominated of WAVE [9] and the choice method warned the observance of the guidelines and recommendations of WCAG 2.0, the philosophy of free code and the validation of an authenticated environment.

II. Web Accessibility Initiative (WAI)

The Web Accessibility Initiative has as proposal to present guidelines and recommendations to provide accessibility [8]. In order to support evaluators, developers and authors of contents in the production of accessible and usable contents by deficient people, WAI articulated the elaboration of the Web Content Accessibility Guidelines (WCAG), which nowadays can be found in the 2.0 version.

According to Reid e Snow-Weaver [10], the referred document has as one of its biggest objectives to describe the requirements to the accessibility of web contents in a neutral language of technologic and in a way that it can be applicable in any technology W3C or not, as CSS, SMIL, SVG, PDF or Flash, in addition to HTML and XHTML.

The accessibility guidelines were built based on four basic principles to a website [11]:

- Perceptible – the information and functionalities must be presented in a way that users can percept them
- Operable – the interactive functionalities must be available to users in a way that users can operate them
- Comprehensible – the information and functionalities must be clear to the understanding of users
- Robust – the contents must be robust enough to be reliably interpreted by a vast variety of agents, including assistive technologies.

Referring to the four principles, there is a list of twelve guidelines with orientations for the content to be accessible for the biggest amount of people. In the bottom of ever guideline there are success criterion that describe specifically what should be achieved, in order to fulfill the rule. All the success criteria of WCAG 2.0 are written as testable criteria to objectively determinate if the content satisfies those criteria. While some tests are automated by utilizing evaluation software programs, others need human testers in a part or in the whole test.

The guidelines are available in WCAG 2.0 [11], where the accessibility is identified in the following levels:

- Level "A" of conformity: is the minimum criterion of conformity, where all the success criteria categorized as A are satisfied
- Level "AA" of conformity: all the success criteria categorizes as A and AA are satisfied
- Level "AAA" of conformity: all the success criteria categorized as A, AA and AAA are satisfied

It is important to say that the success criteria adopted to the conformity levels are determined having as measure the difficulty level that they present to deficient people, when compared to other publics (by the committee's point of view).

Besides the principles, WAI has non-testable recommendations, but those are ones that give framework global objectives to help the understanding of the success criteria and implement techniques in a better way.

III. Software of Automatic Evaluation

An automatic evaluation program, usually called validator, evaluator or online validator, is a set of tools that evaluate the content of a website according to a set of standards that determinate the accessibility level of the document. To do so, it needs to detect the code of a web page and analyze its content based on guidelines and accessibility recommendations such as W3C [1] and the Section 508 [12].

The validator helps to verify if the analyzed interface was developed by using the web standards of accessibility. In general, these programs are available on the internet by being commercialized or by free distribution, and several differences between them are pointed. Referring to the use of guidelines of W3C, several of them attend only the 1.0 version of WCAG. However, the current guidelines can be found in the document WCAG 2.0 [11].

The evaluations made thru validators are usually fast, but not capable to identify all the accessibility aspects. In general, the utilized tools make the verification based on the W3C recommendations, even if some of them are capable to analyze the submitted document deeper than others. Considering the different criteria that can be adopted to the validation of each one of the accessibility recommendations, the validators present some differences in relation to the answers, warnings and identified problems. According to Faulkner and Arch [13], the automated tools:

- Verify the code's syntax;
- Identify real accessibility problems;
- Identify some potential problems;
- Identify pages that contain elements that might cause problems;
- Search for known standards.

IV. Why WAVE?

There are several accessibility validation softwares available on the internet. Initially it was selected the ones indicated by WAI, although by the moment of this research the WAI's list was outdated, not having indications of any automatic tool that validates the conformity of a document with WCAG in the 2.0 version, but only with the WCAG 1.0.

Then it was considered only the softwares based on the WCAG 2.0 guidelines, considering that some were identified by Al-Khalifa et al [14] and others identified by the authors of this article. Another prerequisite to the choice was that the softwares attend to the open source philosophy, or at least that did not present a cost for acquisition. The table 1 presents the tools that were found and selected.

TABLE I. AUTOMATIC EVALUATION TOOLS

Software	Description	Levels of Conformity
AccessMonitor [15]	Developed by UMIC (Agency to the Society of Knowledge). It had as a starting point the accessibility evaluation tool eXaminator to WCAG 1.0. It emits an accessibility report and a synthesis of the results with an index, that is a valuation unit which the final result synthesizes and quantifies the level of accessibility achieved.	A, AA, AAA
AChecker (Public)[16]	Developed by the Adaptive Technology Feature Centre from the University of Toronto. It presents the results in three categories: known problems, probable problems and potential problems.	A, AA, AAA
ASES 2.0 [17]	Avaliator and Simulator of Accessibility of Sites- its objetive is to provide instruments that make the adoption of accessibility in government sites possible. According to ASES [17], it has tools that evaluate the conformity according to guidelines of WCAG 2.0 and e-MAG 3.0 [18].	A, AA, AAA
TAW3 [19]	It evaluates web pages and stand-alone Java applications. It presents the result in three categories: problems, advertences and non-verified. Based on the accessibility fundaments proposed by WAI.	A, AA (free version) AAA (commercial)
WAAT [20]	Web Accessibility Assessment Tool- Java application developed by the EU FP7 ACCESSIBLE project. Based on the accessibility fundaments proposed by WAI.	A, AA, AAA
WAVE [9]	Web Accessibility Evaluation Tool- avaliable by WebAIM.Provides four kinds of report: mistakes, features and warning; analysis of the page's structure, identifying the sequency of navigation; presentation of the page in the only text mode and, finally, identification of the headers of the page.	A, AA, AAA
Worldspace FireEyes [21]	Conceived as add-on of the Fifefox navigator, tests static and dynamic contents.	A, AA

Source: from the authors

The selected softwares were evaluated by three specialists: two graduated and with a master degree in Computer Sciences; and one graduated in Design with a master degree in Engineering and Knowledge Management. The three evaluators are PhD students and participants of the researching group in digital accessibility.

By the first step of the process, the selected tools were tabulated having as requisite to make an accessibility evaluation based on the document WCAG 2.0. Because accessibility validation includes generically several accessibility problems and the main-public in question is deficient people, it is considered that to the achievement of an evaluation result with depth, the three levels: A, AA and AAA, were considered relevant in an application designated to this. It is important to say that the success criteria and the conformity levels adopted by the WAI guidelines are determined based on the difficulty level that deficient people present when compared to other publics (by the committee's point of view) [1]. Therefore, of this group, the tools Worldspace FireEyes and TAW3 were disregarded for evaluating only in two levels of conformity WCAG, which are A and AA.

The next step consisted in execute and test the remaining sofwares in relation to the Moodle environment. Some softwares did not execute correctly when the evaluation sceneries required an user authentication using username and password, such as happens in the Moodle configurations.

From the referenced softwares in Table 1, only two were successful in authenticated sceneries: WAVE and ASES. Others, like TAW3, AChecker and WAAT only presented the possibility of evaluation of these environments thru the option of file upload or copy of source-code. In this context, Pivetta, Saito and Ulbricht [22] execute in their work an evaluation of Moodle by utilizing the quoted options. The authors noticed that this evaluation strategy, for being and offline approach, thus a static analysis, the tools could not evaluate completely the codes that were sent, whereas the evaluated pages were making reference to style pages (CSS files) and JavaScript extern files. Even with limitations, the tools could identify a part of the accessibility problems in the evaluated code. The positive factor about these tools is that they present their reports classified within the four principles of WCAG 2.0, presented in the table 2. However within the methodology utilized to execute this study, none of these tools was selected.

TABLE II. MOODLE ANALYSIS WITH NON-AUTHENTICABLE AUTOMATIC TOOLD

Software	Perceptible	Operable	Comprehensible	Robust
TAW	2	1	-	-
AChecker	10	-	-	-
WAAT	222	2	6	7

Source: Pivetta, Saito and Ulbricht [22]

Next, the softwares WAVE and ASES were tested in the Moodle environment. ASES presented execution problems while the tests and, for that reason, this article includes only the evaluation of WAVE, keeping the evaluation report of ASES for a posterior work.

From this choice, a search in the CAPES[1] website was made, and also in the searching website Google.com, to verify the art state in relation to automatic tools evaluations. Some related works were found in the Google[2] website, such as an article by Faulkner and Arch [13], which refers to the evaluation of four tools of automatic validation. In this article, the WAVE tool is quoted, but not evaluated, being the commercial softwares the tests' main objectives of evaluation. For being an older article, the evaluation arguments regarded the guidelines of WCAG 1.0, document that was already overcome by the 2.0 version. Still, Faulkner and Arch [13] related other works of evaluation of automatic tools, but that

[1] http://www.periodicos.capes.gov.br.ez47.periodicos.capes.gov.br/ - access in 12/2012

[2] http://www.google.com.br – access in 02/2013

also refers only to WCAG 1.0, which differs of this proposal. Alexander and Rippon [22], however, utilized WAVE in their work to evaluate an academic website, not including properly an evaluation of the WAVE software, but its application to web environments evaluation.

V. Evaluation Of WAVE

WAVE is a set of accessibility evaluation web tools based on the guidelines of WCAG 2.0 [7] and Section 508 [12], of free access, developed by WebAIM [9]. The validator, instead of providing a technical report like most of automatic evaluation programs, shows the evaluation result on the web page that originated the tests by utilizing embedded icons and indicators that reveal the pages' accessibility. It allows evaluate an URL (Uniform Feature Locator) of a website, although in case the files are not publicly available on the internet, there is still the possibility of making the upload of the files to evaluation WAVE in the tool's website.

Another possibility presented is to copy the HTML code of the website chosen to execute the test and paste in the formulary available on the website. Besides, WAVE offers the download option of a toolbar in Mozilla Firefox, which is installed as a complement to the navigator.

Considering the deficiencies already discussed of an evaluation by upload of source-code, to test the Moodle environment internally the download of the toolbar WAVE Firefox was necessary. Already installed, the Moodle environment was executed with user and authentication password.

The WAVE Firefox toolbar allows evaluating web pages directly in the navigator and test environments that are protected by passwords. According WAVE [9], the toolbar evaluates the contents exhibited locally and dynamically, made from scripts or AJAX. It is composed by four tools that execute the verification of:

- Errors, features and alerts;
- Order of the structures;
- Only texts;
- Visualization of headers.

The toolbar contains other options, such as the option to disable styles, a link to the page that contains explanations about the accessibility icons and the option to "clean" evaluations that were already executed.

Every time a page is submitted to evaluation, colorful icons with different shapes appear as a result of what was evaluated. These are the identified categories:

- Red icons – indicate accessibility errors, which is, contain accessibility problems.
- Yellow icons – indicate alerts and, in this case, can be or not accessibility issues, but generally indicate an area where accessibility is, for several times, a problem, or that can be improved
- Green icons – indicate areas that contain elements with accessibility features and that the author must verify the accuracy.
- Light blue icons – indicate structural, semantic or navigation elements that can help in accessibility. These icons must also be verified.
- Trapezium icons – related to the images available on the website.

All the existing icons can be visualized in Fig 1. Also in the WAVE toolbar there is a link with explanations about the meaning and the recommended actions for each element.

To execute a WAVE report, it is necessary to select one of the four tools that compose the toolbar. The first test was with the option ERRORS, FEATURES AND ALERTS. The names of the items of the WAVE tool will always be referenced in capital letters to be differentiated from the text. In this first evaluation, the validator was applied to Moodle's homepage, where the authenticated user, in this case a student, can visualize courses, disciplines, his profile and other things. The screen referred to this first evaluation can be visualized in the Fig. 2, that shows how WAVE presents the results when submitted to validation of a website.

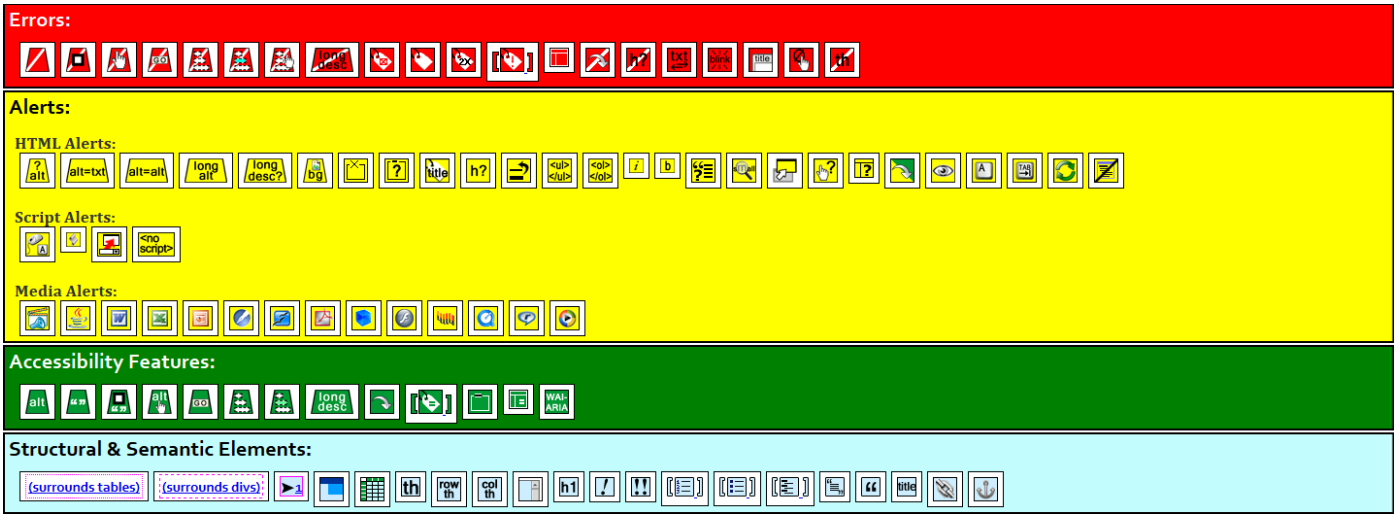

Fig. 1. Icons to accessibility indication (source: http://wave.webaim.org/icons)

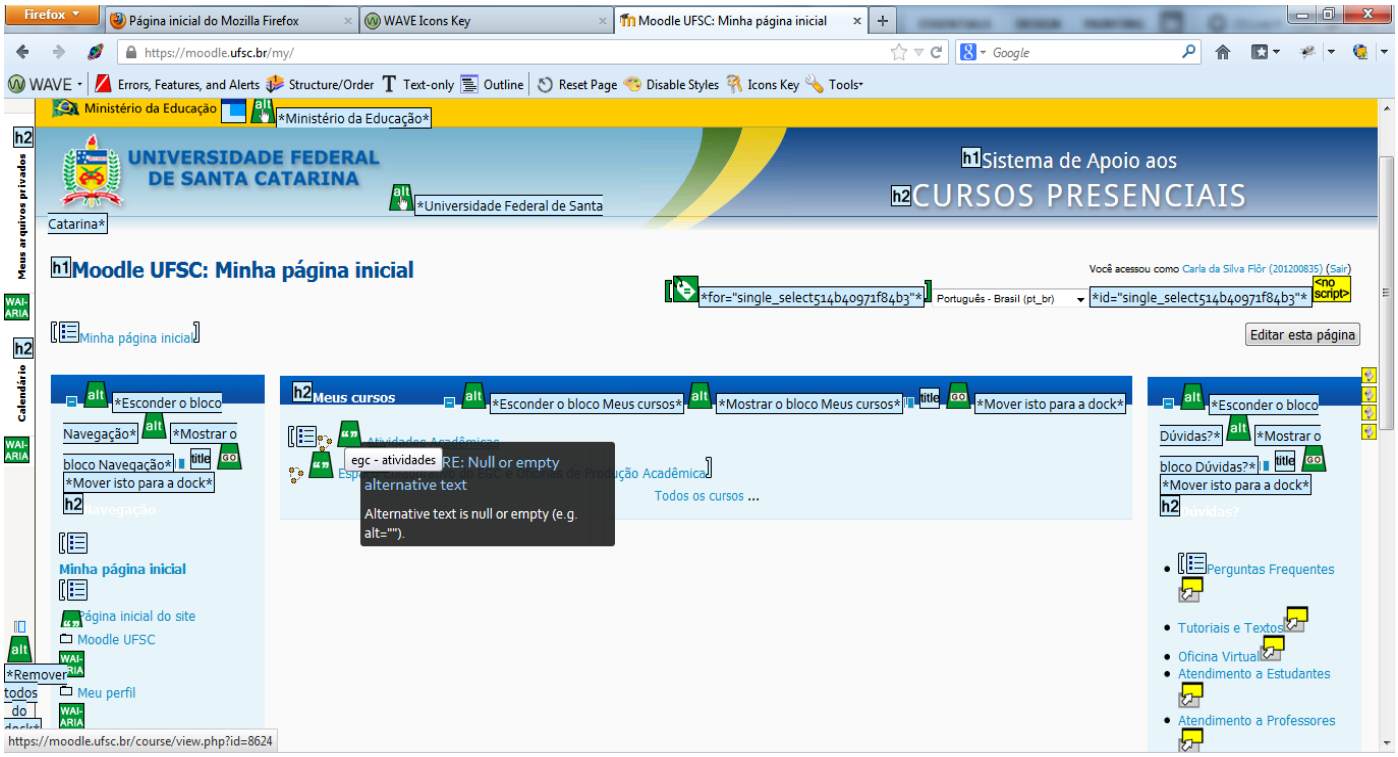

Fig. 2. WAVE test – ERROS, FEATURES, and ALERTS

The page is modified by the presence of accessibility icons in different areas of the website. Every time that the mouse cursor is positioned over one of these icons a brief description about the icon is presented, as you can see in a black rectangle in the Fig. 2. Or still, the user can see a detailed description of each icon in the option "Icons Key" that can be found in the right superior area of the page. The result of the tool in the homepage was showed thru the graphic presentation of icons:

- Yellow icons – alert about alternative texts to near images, links to new windows, CSS to occult content that were not read by screen readers, alternative texts for non-executed scripts on the navigator, JavaScript and others.
- Green icons – show existing accessibility features, like alternative contents for images and buttons.
- Blue icons – indicates non-enumerated lists and titles to the sections.

The evaluated page did not present any red icon, which is the icon that points errors, even when the use of CSS (Cascade Style Sheet) is disabled as WAVE [9] suggests.

As WAVE tests one page at a time, from the several screens tested, the error indication occurred only in pages that contained formularies. The registered errors were "image without textual alternative content" and "orphan labels", in other words, without an associated entry. In the other pages the result was similar to the one obtained with the homepage.

According to WAVE [9], the functionalism of the WAVE 1.1.8 toolbar is also available in the tool menu, with the acceleration keys (ALT + T). It allows the accessibility of the keyboard to all the tool's functions, even when the toolbar is not visible. In tests executed in three computers, those acceleration keys did not work. Usually the presentation of an underlined letter in the menu indicates its use as an acceleration feature, and when the underlined letter is pressed at the same time as the key ALT, its functionality is activated. In this case, ALT functioned normally to all the other options in the Firefox menu.

The next tool of WAVE is the STRUCTURE ORDER, which allows the visualization of the structural organization of the website. In this tool, the indicators show a reading sequence that corresponds to the order of navigation in the page. To determine if the reading and the order of navigation of the page make sense and are logical, the numbers must be followed. The tool also indicates the presence of lists, headers, tables and alerts in relation to the structure and functionality of these elements. The submission's result can be observed in the picture 3.

In sequence, the presented tool is the TEXT ONLY. This tool provides the option to visualize only textual information of the page. The tool removes the page's visual style and provides a verification of what is read by a screen reader, including alternative texts to images and bottoms. Besides, with this tool, other occult information to the user becomes visible, such as the "skip navigation" and "skip main menu" links.

The DISABLE STYLES tool has a similar effect. However, it is different from TEXT ONLY just for removing the page's styles, maintaining the images and not showing the alternative contents to images and bottoms like TEXT ONLY.

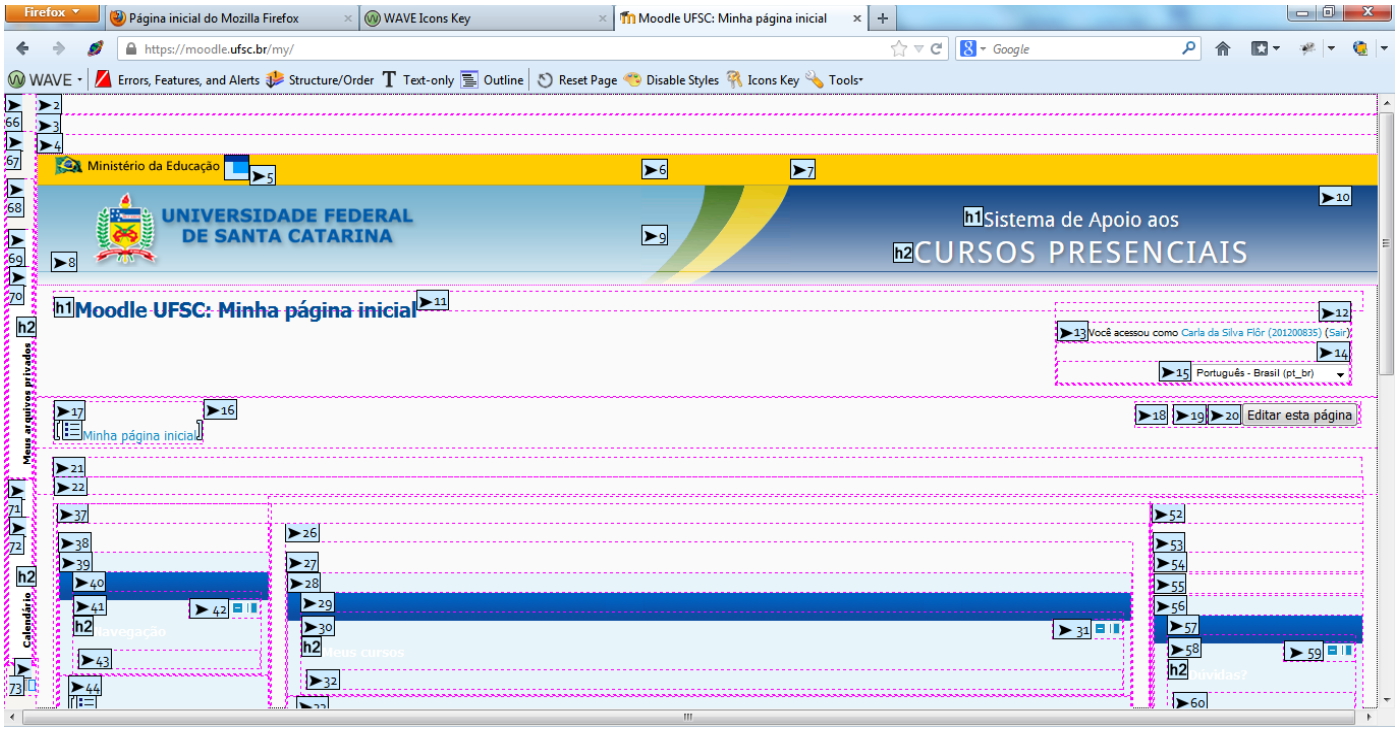

Fig. 3. Test with STRUCTURE ORDER

Lastly, the OUTLINE tool allows the visualization of headers and its levels, verifying if the structure is logical and adequate.

VI. Analysis of the Study

The WAVE, object of study of this work, was efficient when it comes to evaluation. However, it was observed that the results presented a few accessibility errors when compared with other analysis with the Moodle environment, such as the WAAT tool, described in Pivetta, Saito and Ulbricht [21]. Due to these results, it was made an analysis of tests and criteria utilized by WAVE and its tools to confront them with what is proposed by WCAG 2.0.

The W3C, thru WCAG 2.0, establishes as a base to the validation tests 60 testable success criteria that are included in the conformity levels A, AA and AAA. A great part of these criteria can be automatically tested to the presentation of results in reports to accessibility analysis.

In the WAVE analysis, it was verified the existence of 20 tests to accessibility evaluation in web pages in contrast with the 60 Success criteria of WCAG 2.0. From the 20 tests, 8 make reference to errors with image insertions, 4 make reference to errors of labels in formularies and only 8 test other accessibility errors of pages. These 8 are divided in:

- 3 errors to title: no title in tables, in page, in header;
- 2 errors to links;
- 1 error for having an HTML <*marquee*> (moving texts);
- 1 error to header of table without text.

Because these 20 errors are not identify within the four principles of WCAG 2.0 this work tried to analyze these errors and classify them within the principles. The table 3 and 4 shows the result of the classification.

TABLE III. WAVE ERRORS AND COMPARATIVE WITH THE SUCCESS CRITERIA WCAG 2.0

ERROR Wave	**Success Criterion WCAG 2.0**
ERROR: Missing alternative text;	1.1.1 (Non-text Content)
ERROR: Spacer image missing alternative text;	1.1.1 (Non-text Content)
ERROR: Linked image missing alternative text;	1.1.1 (Non-text Content)
ERROR: Image button missing alternative text	1.1.1 (Non-text Content)
ERROR: Image map missing alternative text	1.1.1 (Non-text Content)
ERROR: Image map area missing alternative text	1.1.1 (Non-text Content) 2.4.4 (Link Purpose (In Context)) 2.4.9 (Link Purpose (Link Only))
ERROR: Server-side image map	No match in WCAG 2.0. Reference to Section 508
ERROR: Invalid longdesc	1.1.1 (Non-text Content)
ERROR: Form label missing	1.1.1 (Non-text Content) 1.3.1 (Info and Relationships) 3.3.2 (Labels or Instructions) 4.1.2 (Name, Role, Value)
ERROR: Empty form label	1.3.1 (Info and Relationships) 3.3.2 (Labels or Instructions)
ERROR: Multiple form labels	4.1.1 (Parsing)

ERROR: Orphaned form label	1.1.1 (Non-text Content) 1.3.1 (Info and Relationships) 3.3.2 (Labels or Instructions) 4.1.2 (Name, Role, Value)
ERROR: Frame missing title	2.4.1 (Bypass Blocks) 4.1.2 (Name, Role, Value)
ERROR: Broken skip navigation link	2.4.1 (Bypass Blocks)
ERROR: Empty heading	2.4.6 (Headings and Labels)
ERROR: Marquee	2.2.2 (Pause, Stop, Hide: (Moving, blinking, scrolling))
ERROR: Blinking content	2.2.2 (Pause, Stop, Hide)
ERROR: <title> is missing or not informative	2.4.2 (Page Titled)
ERROR: Empty link	2.4.4 (Link Purpose (In Context)) 2.4.9 (Link Purpose (Link Only))
ERROR: Empty table header	1.3.1 (Info and Relationships)

Source: the authors

TABLE IV. WAVE CLASSIFICATION AND COMPARATIVE WITH WCAG 2.0

Principles	**WCAG 2.0**		**WAVE**
	Guidelines	***Success criteria***	***Errors that correspond to the success criteria WCAG 2.0***
Perceptible	4	22	2 (1.1.1 - 1.3.1)
Operable	4	19	6 (2.4.4 - 2.4.9 - 2.4.1 - 2.4.6 - 2.2.2 - 2.4.2)
Comprehensible	3	17	1 (3.3.2)
Robust	1	2	2 (4.1.2 - 4.1.1)
Total	12	60	11

Source: the authors

Analyzing the Tables 3 and 4 it is noticeable that eleven from all the errors that were treated by Wave are in the perceptible principle, but are equal to only two from the twenty-two success criteria to this principle (it occurs because each success criterion can be tested by more than one Wave error).

From the eleven errors in the perceptible level, nine are related to the 1.1.1 criterion (Non-text Content), while three (one error is in both criteria) are related to the 1.3.1 criterion (Info and Relationships). In the operable level, eight Wave errors correspond to six success criteria, and WCAG 2.0 foresee nineteen success criteria to this principle. By the same way, in the comprehensible level, three Wave errors are related to only one success criterion of this level, the 3.3.2 (Labels or Instructions), while WCAG 2.0 foresee seventeen criteria. The only totally satisfied principle is the lustiness, that has only two success criteria and both are testable by Wave errors.

Therefore, it is noticeable that Wave does not contemplate big part of the success criteria of Wave 2.0, covering only eleven from the sixty success criteria foreseen by WCAG 2.0.Besides, the WAVE tool does not categorize the errors according to the four principles of WCAG 2.0 for presenting a different report, composed by graphic icons within the page.

The report formats vary a lot, depending on the target-public, on familiarity with Web Design and Web accessibility standards. In case of web designers, developers and evaluators that know which better format answer its necessities, are able to choose one appropriated tool. According to WebAIM [24] the evaluation tools include six report formats:

- Based in text – errors listed by line number.
- Based in text – errors listed by linked line number, which is, links errors with the source-code.
- Based in text – errors listed within source-code.
- Based in text – errors listed within source-code and GUI (graphic user interface) – uses tables to show the users three pages in once: the report based on text, the graphic interface of the user of the webpage, and the error cases detached in the code.
- Graphic – based on icons.
- EARL – the EARL reports are a W3C attempt to standardize accessibility reports and help the users to compare the efficiency of the accessibility tools [1].

However, thru this graphic approach of icons, the developer can verify: alerts, accessibility features and where they can be found. To each identified item, WAVE has recommendations, such as:

- HTML alerts – 25 recommendations;
- Script alerts – 18 recommendations;
- Media alerts – 14 recommendations;
- Accessibility features – 13 recommendations;
- Structural and semantic elements – 20 recommendations.

The alerts are necessarily accessibility errors, although present HTML, scripts or media points that deserve more attention and that could be improved. As an example, a text alternative named “image” could be created; even though it is present in the code, it is not representative to the user because does not describe the image content.

On the other hand, the accessibility features highlight the present features for the developer to be able to verify if these are correct. The structure and semantic elements indicate the structure, the navigation and the semantic of the page, in a way that a correct read of the order and the hierarchy of the information can be made.

VII. CONCLUSION

This work evaluated automatic software of accessibility evaluation in relation to an authenticated virtual environment:

the Moodle. Softwares of this category present different approaches, shapes, characteristics and benefits, considering that some of them can present a large quantity of tests while others sub estimate the existing problems in a website. The choice of the ideal tool depends on a set of abilities and how the evaluator defines the responsibilities of the site that he wants to test. An important efficiency measure of an automated product is the capacity of produce results without the necessity of a more human interpretation. In this sense, were adopted the following validation criteria of the program about to be evaluated: freeware tools, the WCAG 2.0 guidelines, the possibility of evaluation of authenticated environments and the absence of execution errors in tests. Within the identified and selected programs, the one with better characteristics was WAVE.

WAVE's proposal indicates that the tool is appropriated to help web developers to make the available content more accessible. The software in question does not describe of a content is accessible or not, but helps the evaluator to verify accessibility aspects of this content. According to information from the site [9] the use of the tool demands experience of the evaluator user, which is, it is important that the person that is analyzing the site has the knowledge in computer sciences to a better understanding of the alerts and errors.

Considering that WAVE is a software that proposes the accessibility evaluation of websites, the tools appear to be a little limited. One example refers to the tools TEXT ONLY and OUTLINE, which are not very useful in a accessibility validation level, only help the developer to verify the structure of the site in the text more and in the header structure mode, respectively.

As an accessibility evaluation tool, it could be rearranged in a way to be in conformity with WAI guidelines, considering the four basic principles: perceptible, operable, comprehensible and robust, in a way to available error reports, alerts and other items classifying them within these principles. On the other hand, the used methodology, which is the icons insertion within the website is interesting due to the facility to identify the area and the accessibility item that is present or not.

Lastly, to provide accessibility tools is a great step to web accessibility. However, it is important to say that the use of tools to verify accessibility is only the first step. Besides, the evaluator must be warned about the tool's limitations and have knowledge of accessibility subjects and its implications to deficient people, in order to interpret the reports about signalization of alerts and errors.

REFERENCES

[1] W3C, Evaluation and Report Language (EARL) 1.0 Schema, 2011, http://www.w3.org/TR/EARL10-Schema/

[2] BRASIL, Decreto nº 5.296. Presidência da República - Casa Civil - Subchefia para Assuntos Jurídicos, 2004. http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2004/decreto/d5296.htm.

[3] W3CGT, GT Acessibilidade. http://www.w3c.br/GT/GrupoAcessibilidade#w3c_inicio_conteudo

[4] IMS, Global Learning Consortium. IMS Access For All v2.0 Final Specification. http://www.imsglobal.org/accessibility/

[5] BRASIL, Decreto nº 5.622. Presidência da República - Casa Civil - Subchefia para Assuntos Jurídicos. 2005. http://portal.mec.gov.br/seed/arquivos/pdf/dec_5622.pdf

[6] Moodle, Modular Object-Oriented Dynamic Learning Environment. http://www.moodle.org

[7] WCAG 2.0, Web Content Accessibility Guidelines 2.0. http://www.w3.org/TR/WCAG20/.

[8] WAI, Web Accessibility Initiative. http://www.w3.org/WAI/

[9] WAVE, Web Accessibility Evaluation Tool. http://wave.webaim.org.

[10] L. G. Reid, A. Snow-Weaver, "WCAG 2.0: Web Accessibility Standard for the Evolving Web". Proceedings of the 2008 International Cross-Disciplinary Conference on Web Accessibility, W4A. Beijing, China: ACM Press, 2008.

[11] WCAG 2.0 – Web Content Accessibility Guidelines. http://www.w3.org/TR/WCAG/

[12] Section 508. https://www.section508.gov

[13] S. Faulkner, A. Arch, Accessibility Testing Software Compared, 2003. AusWeb03, University website accessibility revisited. http://ausweb.scu.edu.au/aw03/papers/arch/paper.html

[14] H. S. Al-Khalifa et al. "A Pilot Study for Evaluating Arabic Websites Usign Automated WCAG 2.0 Evaluation Tools", Proceedings of the 2011 International Conference on Innovations in Information Technology. Riyadh, Saudi Arabia: IEEE Computer Society, 2011.

[15] UMIC, http://www.acessibilidade.gov.pt/accessmonitor/

[16] ACHECKER, Web Accessibility Checker. http://achecker.ca/checker/index.php.

[17] ASES, Avaliador e Simulador para a Acessibilidade de Sítios. http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/ases-avaliador-e-simulador-de-acessibilidade-sitios.

[18] EMAG30, e-MAG – Modelo de Acessibilidade de Governo Eletrônico – versão 3.0, 2011. http://www.governoeletronico.gov.br/biblioteca/arquivos/e-mag-3.0/download

[19] TAW, Accesibilidad de Sitios Web. http://www.tawdis.net/

[20] WAAT, Web Accessibility Assessment Tool. http://www.accessible-eu.org/.

[21] WORLDSPACE, Worldspace fireEyes. http://www.deque.com/products/worldspace-fireeyes.

[22] E. Pivetta. D. S. Saito and V. R. Ulbricht. "WCAG e a Acessibilidade de Ambientes Virtuais de Ensino Aprendizagem na Perspectiva de Alunos Surdos". Proceedings of II Conferência Internacional de Integração do Design, Engenharia e Gestão para a Inovação, 2012, IDEMi 2012. Florianópolis, 2012.

[23] D. Alexander, S. Rippon, "University website revisited", AUSWEB07, 2007. http://ausweb.scu.edu.au/aw07/papers/refereed/alexander/paper.html.

[24] WebAIM, "Accessibility Evaluation Tools" http://webaim.org/articles/tools/