

On the application of W3C Guidelines in Website Design from scratch

Diamantino Freitas, Helder Ferreira

Faculty of Engineering of the University of Porto
LPF-ESI / DEEC / FEUP / Portugal
dfreitas@fe.up.pt, hfilipe@fe.up.pt

Abstract

The Web Content Accessibility Guidelines (WCAG1.0) [1], from the World Wide Web Consortium (W3C) [2], became a recommendation in May 1999. However, up to now, only a few websites took them into full consideration. In spite of the governmental rules that some countries enforce applying to official websites, web designers still don't comply with the guidelines, building inaccessible web pages for about 240 million persons in the world [3]. Therefore, in a spirit of initiative and divulgation, the Laboratory for Speech Processing, Electro-acoustics, Signals and Instrumentation of the Faculty of Engineering of the University of Porto, developed a website (<http://lpf-esi.fe.up.pt>) with a simple information system database, that complies to all WCAG1.0 in the predicted cases used in the site. A triple-A level design [4] was developed to make sure that a larger number of people with disabilities were included. Without great costs of layout it was possible to build an enjoyable and accessible site to every user; what proves that it is feasible to make a small effort to improve the accessibility of web contents. We defend the point of view that it's not the layout that is important but the information it presents to us. Some libraries, scripts and tools were developed in PERL to help build web pages that comply with the WCAG1.0. The concepts and application of those tools will be explained in the paper. We will describe a few essential steps of a website preparation project, which are found essential to follow a good path and to avoid time waste. We will also provide a few solutions to apply the W3C Guidelines in practice. The result and the intermediate phases were preliminarily evaluated by users using standard browsers and text browsers equipped with screen-readers, showing promising results.

1 Introduction

Before starting to build an accessible website from scratch, a few questions come up, such as: "What are the main accessibility issues? Why should we fore plan before we start programming a website? What are the solutions? Where should we start?". WYSIWYG (What You See Is What You Get) editors are often the general solution adopted to build a website, however those editors don't produce pages with compliance to all the accessibility rules. Sometimes the information we provide in a website is context dependent and needs our personal attention to become accessible. The most convenient approach will be the use of tools and methods built specifically for accessibility compliance, combined with a profound study of accessibility issues. A "do it yourself" and "always validate" philosophy. One possible method, adopted by us, consists in using a normal text editor, and building the web page using Hyper Text Mark-up Language (HTML) and Cascading Style Sheets (CSS). To reduce the amount of programming work, a few tools in

Practical Extraction and Report Language (PERL) were developed to produce accessible blocks of code.

2 The Web Design Process

Several points should be considered when fore planning and developing a website. Accessibility issues should be one of the main priorities. The process followed for building this website was:

- Selecting the raw contents and getting contact with the accessibility issues.
- The layout design.
- Developing and using tools and libraries.
- Working the content information
- Other aspects to consider: format, metadata, language, security.
- Validate and test the website.

Different web applications demand different layouts and different approaches to become accessible. However, an important rule to always keep in mind is: Information is the most important thing, layout comes in second! The following mistakes are very common: the abuse of tables for layout, lots of animations, auto-refresh or auto-redirect of information and multimedia presentations without text-version alternatives. The use of the previous techniques often compromises the accessibility solutions, therefore they should be avoided. A few considerations about the layout design should be fore planned: a navigation bar must be always available and easily accessible, a search engine might be a useful tool, there should be enough contrast between the background and the text colour, the text should be simple and concise, images should always have descriptions, data tables should have the proper mark-up and the layout should be constant through all the web pages of the site. We suggest the use of CSS to improve the beauty of the layout, instead of other multimedia presentation applications such as shockwave or flash. In this website we used only HTML and CSS. The chosen layout consists in: on the top of the pages, an organization logotype, followed by a title. In the middle, the content information composed by text, images and links. On the bottom, a navigation menu. Using this layout, we provide to the visiting user the following cadence of information:

- Organization logo
- Title of information.
- The information itself
- Navigational menu

Since our navigational menu makes use of keyboard shortcuts, in long web pages, the user can change its choice without making scroll. Also remember: the majority of accessible websites use simple contents applying simple solutions! In our website, we used only text and images.

WCAG1.0 states that content layout should be constant along the website (Guideline 14 – Priority 3). This introduces the reason why we have developed a few tools and libraries: the use of layout templates already validated is important. These tools allow a reduction in the amount of work of programming a website. Almost every web page in this website makes use of those libraries and tools to create dynamic pages with HTML code sections previously validated in accessibility issues. However, automatic and personal validation is always required along the process of developing a website! Every web designer is encouraged to study and apply the W3C's Web Content Accessibility Guidelines.

A screenshot of the website we have developed is provided in figure1, in the next page.

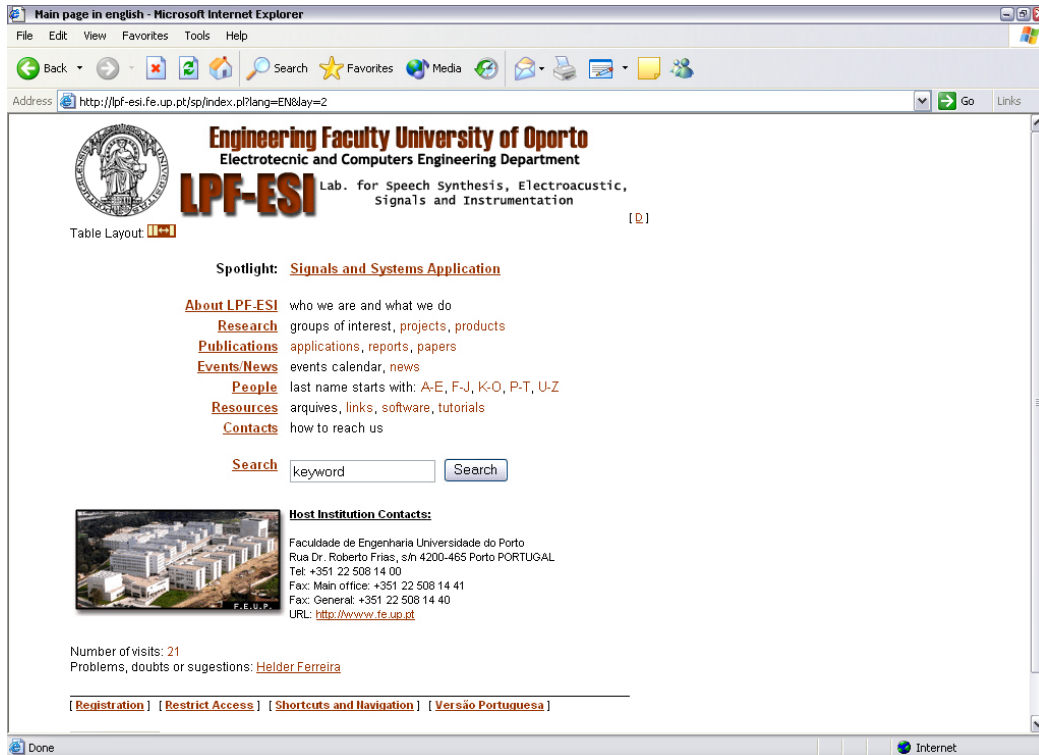


Figure 1: Screenshot taken from the developed website. Figure shows the main web page composed by a title, a practical menu, a search engine, a few contact informations and ends with a navigational menu.

3 Developed Libraries

To reduce the repetition of work, and to avoid a few errors in the accessibility design, some libraries were developed containing functions that create dynamic web pages already validated in WCAG1.0. The created libraries are:

- *HTML.pm*: Functions in this library allow the creation of accessible HTML block codes that are used to build web pages. For example, there are functions to generate HTML code for the navigation menu, the top identification of a web page and for messages to the visiting user. Since these are repetitive elements along the website, once we validate them, they will always produce accessible HTML.
- *BIBLIOTECA.pm*: Tools for security and input tests.
- *CONFIG.pm*: Declarations and configurations. This is a very useful library, since it allows the website to change its URL, name of machine, etc.. In short, to be portable to other systems.

However, the construction of these libraries does not release the use of personal and automatic validation.

4 On the application of W3C guidelines

Different types of media can be found in a website, such as: text, images, audio, video and object scripts. Before start developing, it is very important that the web designer studies profoundly the accessibility issues each type of media carries and the options available to solve them. W3C provides several techniques and guidelines with accessibility solutions to almost every situation (Techniques for WCAG1.0 [5] and WCAG1.0 Recommendation [1]). However some guidelines require substantial interpretation.

For example, on *Guideline 1 – “Provide equivalent alternatives to auditory and visual content”*, the text equivalent used in the ALT tag on an image can be very subjective and its necessary a test with a blind person to ensure that the descriptions used are enough and lead to good interpretation.

On *Guideline 2 – “Use mark-up and style sheets and do so properly”*, the use of relative units in CSS is essential to achieve a higher accessibility, however introduces a problem in higher screen resolutions (the text can became too small).

On *Guideline 5 – “Create tables that transform gracefully”*, it’s not very clear how a screen reader would read a table with several cells within, and cells within cells.

On *Guideline 9 – “Design for device-independence”*, the creation of a tab order it’s very subjective. The web designer must consider which links are the most important in a web page. The keyboard control shortcuts appear to be browser-dependent. They don’t work every time. The tab order seems to be more important, since it works on every browser.

On *Guideline 10 – “Use interim solutions”*, it is stated the use of default characters in empty edit boxes and text areas, as being a level 3 priority rule. However, we found this very confusing for visiting users, especially when using visual browsers, because the INPUT tags can already use the TITLE attribute that indicates a brief description of the field in a form. In text browsers, we can’t see the tool tip text of the fields so we don’t have information about it. Still, it is confusing to see a form already filled with descriptions or default characters. Sometimes it’s not possible to include default characters in a field, for example, in a password input.

On *Guideline 14 – “Ensure that documents are clear and simple”*, it’s stated that web pages should use clear and simple language and the use of graphic or auditory presentations to facilitate comprehension. However we found very difficult to validate this. How should we establish what is clear or not? How do we know if we need more graphical elements to help the comprehension?

Nevertheless, we would like to state that the *W3C’s Web Contents Accessibility Guidelines* (WCAG1.0) are very useful and absolutely necessary to give the first steps to change the way that websites are created nowadays.

5 Performed Tests and Validation

This website was tested in three different ways: using checklists, web validators and users with standard browsers (graphical and text) equipped with screen-readers.

The checklist used was the WCAG1.0 Checklist [6]; this is a good example of validate yourself by check-marking the website’s implemented guidelines. However, sometimes is difficult to validate some checkpoints such as 5.3 (*Do not use tables for layout unless the table makes sense when linearized*), 12.3 (*Divide large blocks of information into more manageable groups where natural and appropriate*), 14.1 (*Use the clearest and simplest language appropriate for site’s content*) and 14.2 (*Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page*), by the reasons already discussed in the previous chapter.

The web validators used were:

- Bobby [7]

- W3C HTML Validator [8]
- W3C CSS Validator [9]

We found the W3C HTML and CSS validators excellent tools to prove the mark-up compliance of the web document. However, they don't provide much information about accessibility. Therefore we used Bobby validator that allows a validation on W3C guidelines or US Section 508.

However, there are still many checkpoints that Bobby isn't able to evaluate, requiring web designers' checking, before declaring an accessibility level priority compliance.

A few preliminary tests were made along the development phase with screen-readers. JAWS 4.50 was used. The main problems identified were: a few ALT descriptions in logotype stamps and the lack of mechanisms that provide jumping between two columns in a table, in spite of the capability to gracefully transform the table in a linearized lecture form. The rest of the website was approved.

6 Conclusion and Future Perspectives

This website uses an accessible compliant HTML and CSS code, with AAA priority accessibility level. However we still would like to improve that, allowing an even greater accessibility, and providing alternative forms of mark-up information to be read by accessible browsers or user agents. Nowadays audio and video files run everywhere on the Internet, most of them in a non-accessible way. We would like to include these two elements in our website, presenting further solutions to their accessibility. Speech Synthesis Mark-up Language (SSML) [10] and Aural Cascading Style Sheets (ACSS) [11] will also be one of the future requirements we want to provide. Formulas and mathematical representations will be marked with Mathematical Mark-up Language (MathML) [12].

7 Acknowledgments

The authors wish to thank Rui Almeida for the precious comments and suggestions along the performed tests.

References

- [1] W3C, Web Content Accessibility Guidelines 1.0 from <http://www.w3.org/TR/WCAG10/>.
- [2] World Wide Web Consortium from <http://www.w3c.org>.
- [3] Roe, P. (Editor). (2001). Bridging the Gap?. COST219bis, European Commission, 2001.
- [4] W3C, WCAG 1.0 Priority Types from <http://www.w3.org/TR/WCAG10/#priorities>.
- [5] W3C, Techniques WCAG1.0 from <http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/>.
- [6] W3C, WCAG 1.0 Checklist from <http://www.w3.org/TR/WCAG10/full-checklist.html>.
- [7] Bobby Online Accessibility from <http://bobby.watchfire.com/bobby/html/en/index.jsp>.
- [8] W3C, HTML Online Validator from <http://validator.w3.org/>.
- [9] W3C, CSS Online Validator from <http://jigsaw.w3.org/css-validator/>.
- [10] W3C, SSML from <http://www.w3.org/TR/speech-synthesis/>.
- [11] W3C, Aural CSS from <http://www.w3.org/TR/WD-acss>.
- [12] W3C, MathML 2.0 Specification from <http://www.w3.org/TR/MathML2/>.