

## **Accessibility of public library Web sites**

### **Introduction**

Due to constraints in various browsers and multimedia players people can experience difficulty using the World Wide Web. This is particularly true for people with disabilities who use assistive technologies that can be even more restricted in accessing Web content. This can have greater consequences, such as inaccessible education and lack of access to information in general. Also, access to Web content is sometimes more critical for people with disabilities who can make use of particular digital media only because they are print-disabled. Finally, Web sites created accessible are more usable to non-disabled users as they are more easily navigable and can be used by non-graphical desktop browsers.

Particularly affected people with disabilities are visually impaired (blind, short or long sighted, colour blind, tunnel vision sufferers), dyslexic, and people with motor disabilities who are not able to use a mouse. One of disabilities that are most likely to affect Web access is visual impairment. Visually impaired users have difficulties with pages having images without alternative text, especially if images are used as links (as are in image maps). Other common problems are: using structural elements for page layout (using heading mark-ups for making things written in bigger or different type-face), non-described multimedia (no alternative text for sounds and videos), non-existent alternative for finding pages by browsers unable to render frames and scripting languages, non-described tables, poor colour contrast and badly chosen colours for colour blind users.

As libraries are intended to provide equal access to information for all, our intention was to determine to what extent Croatian public libraries really ensure access to content provided on their Web pages. To that purpose, W3C's Web Content Accessibility Guidelines 1.0 have been used to compare the sites with the recommended guidelines. Libraries should provide accessible Web sites, as well as promote Web content accessibility in the community.

### **Advantages of accessible Web sites**

Research has shown that by designing an interface accessible to people with disabilities, you make it more accessible to all users. The Applied Computing Department at the University of Dundee have pioneered the principle that "designing for extra-ordinary users in ordinary

environments is the same as designing for ordinary users in extra-ordinary environments" - if you design for a disabled user, you are likely to produce an interface which is accessible and usable to a non-disabled user working with non-standard browsing equipment and non-standard hardware (Digital Media Access Group).

If proper accessibility techniques are applied, the site will be more accurately indexed by search engines and thus more easily found by users (using meta tags for describing document, using no-frames section in pages with frames, describing pictures and sounds, making self-describing links not based on pictures only etc.). Apart from that, if information is easier to find, users need fewer page requests and thus the Web server runs more efficiently. A moderate quantity of images, audio files and multimedia will help reduce the server overload (fewer users will request pictures, sounds or other space consuming materials if they know what kind of material they will get). Also, people using new technologies such as Personal Digital Assistants, TV-based Web browsing systems, or in-car browsing systems would be able to use the content as well (Why Web accessibility).

The assistive technologies used by disabled users (such as screen readers linked to browsers) rely on valid HTML in order to work correctly. If the code is not valid, a browsers can present the incorrect format of the information contained on the page. W3C provides an HTML validator (2001) for checking the code online. By making judicious use of W3C technologies which promote accessibility, such as Cascading Style Sheets (CSS), Extensible Mark-up Language (XML), Scalable Vector Graphics (SVG), Synchronised Multimedia Integration Language (SMIL), and Extensible Style Language (XSL), you develop sites which are future looking, but will also 'degrade gracefully' - i.e. will still be accessible to users of older browsing technology (or newer technology which is based on text-only layout like WAP browsers in mobile phones) (Why Web accessibility).

Some features of designing accessible Web pages, such as use of style sheets, can reduce maintenance costs (changing colours or typefaces all over the Web site and thus making a "new look and feel" for the entire site). Cost-effectiveness is easily perceived when compared to the broader audience that a site is available to, and greater usability for other users. This can have a large appeal to e-commerce sites – ensuring that they can be accessed by as many potential customers as possible.

Legislation in the European Union, in the United States and Australia provides for accessible design. Under the UK's Disability Discrimination Act it may be illegal for you to include information on your site that is inaccessible to certain users. No such legislation has been enforced in Croatia (Policies relating to Web accessibility).

## Web Content Accessibility Guidelines

The Guidelines have been developed by World Wide Web Consortium's Web Accessibility Initiative, in cooperation with more than 300 international member organisations. World Wide Web Consortium (W3C) is an international consortium, whose mission is to promote the evolution and interoperability of the Web (Fact sheet : Web content accessibility guidelines 1.0).

The document contains fourteen guidelines on how to design Web sites that are accessible to people with disabilities. However, thus created they become more accessible to all, including those using text-only browsers and hand-held or voice-based computers. The current version is 1.0, although a version 2.0 draft has been prepared.

Three levels of conformance to the guidelines have been defined: Single A, Double A and Triple A, each corresponding to checkpoints of three priorities. Priority levels are defined for every checkpoint. Priority one provides for accessibility on the most basic level and ensures that content is accessible at least to a certain degree – this checkpoint *must* be satisfied; if priority two is not respected, information would be hardly accessible – this checkpoint *should* be satisfied; and priority three refers to elements that would make it difficult to use the content - this checkpoint *may* be addressed. Logos can be provided on the site in order to show the (level of) conformance.

The guidelines are the following:

Guideline 1: provide equivalent alternatives to auditory and visual content.

Example: ``

Guideline 2: don't rely on colour alone.

Example: `<p>This should be <em>emphasized</em> in some way</p>`

Guideline 3: use markup and style sheets and do so properly.

Guideline 4: clarify natural language usage.

Examples:

`<p lang="en">English text</p>`

`<p lang="gr">Deutsche tekste</p>`

(for users with speech oriented browsers)

Guideline 5: create tables that transform gracefully.

Guideline 6: ensure that pages featuring new technologies transform gracefully.

Example:

`<script>`

`<!---`

script...

->Alternative description</script>

(hiding the scripting language part from text-based browser)

```
<P><OBJECT codetype="application/java" classid="java:menu.class"
width="500" height="500">
```

```
<a href="mo1.html">Menu option 1</a><br>
```

```
<a href="mo2.html">Menu option 2</a>
```

```
</OBJECT></P>
```

(making alternative way of getting the same result for text browsers)

Guideline 7: ensure user control of time-sensitive content changes.

(page refreshing)

Guideline 8: ensure direct accessibility of embedded user interfaces.

Guideline 9: design for device-independence.

(Example: not making page "best viewed with IE on 800x600")

Guideline 10: use interim solutions.

Guideline 11: use W3C technologies and guidelines.

Guideline 12: provide context and orientation information.

Guideline 13: provide clear navigation mechanisms.

(making self describing links)

Guideline 14: ensure that documents are clear and simple.

Finally, it is important to emphasize that the intention of the Guidelines is not to get boring pages, but to make accessible all kinds of Web sites such as multimedia to all. The fact is that accessible pages are not those that are different, but those that are *flexible*, i.e. operable for use by different technologies, browsers and particular users (Fact sheet : Web content accessibility guidelines 1.0).

The Guidelines do not recommend creating text-only pages except when it can be the only solution. The reason is that text-only versions are usually kept less up-to-date and provide less content than in primary version.

Apart from HTML and Web content, one should also check the accessibility of browsers in User Agent Accessibility Guidelines 1.0 and authoring tools in Authoring Tool Accessibility Guidelines 1.0, all of which influence accessibility.

## **Evaluating accessibility**

There are automated validation tools for checking conformance to the Guidelines, but manual checking needs to be conducted to make to ensure certainty. W3C's Web Accessibility Initiative lists and describes a number of evaluation and repair tools. Such are, for example, CAST's (Centre for Applied Special Technology) Bobby and SNOW's (University of Toronto's project Special Needs Opportunity Window) A-Prompt. As already mentioned, one can also use the W3C's HTML validator in order to check the underlying HTML if the browser could not interpret the page correctly due to errors in HTML code.

One should also test the site to see if there are any problems when graphics is not loaded, when frames, scripts and style sheets are turned off and whether it is possible to browse without a mouse. Try changing text size in different browsers (Internet Explorer, Netscape Navigator, Opera) as well as setting them to ignore colours, font size and style. Apart from checking the conformance to the Guidelines, it is advisable to have a disabled person to browse through the site, as well as to browse the site using a speech browser or screen reader. Viewing the site through a text browser such as lynx can help in determining problems caused by images with no (equivalent) alternative text, confusing navigation systems, reliance on Javascript to provide information or navigation, complex use of frames without adequate help or navigation for no-frames browsing (Sloan 2002).

## **The research**

### **Sample and methodology**

There are currently eighteen Croatian public libraries that have created Web sites. Sixteen of them were included in the research, as two that were not included consist only of one page with several sentences.

The research was conducted with the help of CAST's validator, to the three hypertextual levels, not counting the home page. The Web content design was compared to Priority 1 of the Web content accessibility guidelines 1.0.

### **Results and analysis**

The Bobby validator has for all the researched sites reported three main groups of results:

1. Priority 1 accessibility, which implies that a page does not meet the requirements for priority one of the Web content accessibility guidelines 1.0;

2. Priority 1 user checks, which implies that there are some elements on the page that could be in non-conformance to the priority one level of the Web content accessibility guidelines 1.0, but one needs to check, as the validator can not determine that for a fact; and
3. Three items that can not be checked by the validator, but are required by the Web content accessibility guidelines Priority one level (Identify any changes in the document's language; If you can't make a page accessible, construct an alternate accessible version; and, Use the simplest and most straightforward language that is possible).

There are several items that have not been respected in designing Croatian public libraries Web sites within the Priority one group. All images, including image maps, spacers, bullets in lists, graphical buttons, links should contain an alternative text description that represents the function of the graphic, which is easily achieved by using the ALT tag. This also includes all image-type buttons in forms. All the libraries included in the research have at least one page where they do not provide an alternative text description, and one of them also doesn't provide alternative text when image-type buttons in forms are used. Third item, providing alternative text for each applet, is important for people who can not access a Java applet because their browsers don't support them or have disabled Java, as well as the visually impaired, whose assistive technologies do not provide support for Java. Thus, applets should contain a short alternative text description placed inside the APPLET element, or an alternative HTML-based content within the Applet that duplicates the function of the applet. Four libraries included in the research do not provide an alternative text description for applets. The following item refers to provision of alternative text for image map hot-spots (AREAs or MAPs). Image maps contain "hot-spots" where a person can click to follow a link, which can not be used by people using screen readers and text-based browsers. The problem is solved when AREA tags have an ALT attribute. Four out of sixteen library sites do not have AREA tags with an ALT attribute. Another important item is naming each frame, as there are users who can not access more than one frame at a time. Thus, each FRAME element should be given a title attribute to denote the content of the frame. Seven library sites included in the research do not provide frame titles.

Second group of results suggests checking if some visually oriented usage of HTML elements are meaningful for text-oriented users. All libraries have almost the same problems on their pages. Usage of colour or changing typeface for emphasis is something that should be

done by style-sheets and not directly inserting into page. When presenting data in tables, one should describe the table and the columns. This does not mean that tables used for layout must have a description, but it could be useful. Describing columns in tables is sometimes crucial if columns or rows have complicated headers that have hierarchy. The problem with scripting languages and other programmatic objects is that page creators did not include text-oriented solution for such elements, making some pages inaccessible (like Java-based menus which are not duplicated in any other way).

The third group of results has been suggested for all the library sites included in the research. The first item, Identify any changes in the document's language, refers to multilingual Web pages, which helps a browser to display text appropriately for that language, and allows access aids such as speech synthesizers and Braille devices to pronounce text using the right language. This is done with the "lang" attribute of elements in HTML 4.0. Constructing an alternate accessible version is in rare cases when one wants to preserve the designer's intention necessary after all, and then accessible HTML has to be provided. Using simple language is important for people with learning and cognitive disabilities, people with hearing impairment as well as for those who don't speak the language of the site perfectly. One can also create different pages for different target groups, e.g. children, experts etc.

## **Conclusion**

None of the Croatian public library sites included in the research conform to the Priority one of the Web content accessibility guidelines. Providing fully accessible Web sites (e.g. conforming to priority levels two and three), which libraries should as main information centers, has not yet been nearly achieved.

Some accessibility problems are big and should be corrected as soon as possible (inaccessibility due to programmatic elements without alternative). But, generally speaking, the libraries should just make an extra effort to provide alternatives for those that can access only textual information. These would be large improvements achieved with little effort. What should be done? Describe images. Describe links. Make alternatives for dynamic elements (programs). Use style sheets and not <font> elements.

The reason why there are not so many problems is because the Web sites mainly do not use many new technologies. Those which do, use them without thinking of users with

text-based browsers. But there is so little to be done in order to make the sites accessible to a much broader audience.

In the US and UK there are a number of public libraries Web sites that conform to the Web content accessibility guidelines 1.0 and as such they can serve as examples of information centres that are at the same time well designed and accessible to all (see, for example, Buffalo and Erie County Library, Hickory Public Library, The J. Edgar & Louise S. Monroe Library, Minneapolis Public Library or National Library for the Blind (UK)).

## **Bibliography**

1. A-prompt project.  
<http://aprompt.snow.utoronto.ca/> (15-02-2002)
2. Authoring tool accessibility guidelines 1.0.  
<http://www.w3.org/TR/ATAG10/> (17-03-2002)
3. Bobby worldwide : CAST.  
<http://www.cast.org/bobby> (15-02-2002)
4. Buffalo and Erie County Library.  
<http://www.buffalolib.org/> (15-02-2002)
5. Curriculum for Web content accessibility guidelines 1.0.  
<http://www.w3.org/WAI/wcag-curric/> (17-03-2002)
6. Digital Media Access Group : resources.  
<http://www.dmag.org.uk/resources/> (15-02-2002)
7. Fact sheet : Web content accessibility guidelines 1.0.  
<http://www.w3.org/1999/05/WCAG-REC-fact> (17-03-2002)
8. Hickory Public Library.  
<http://www.ci.hickory.nc.us/library/> (15-02-2002)
9. The J. Edgar & Louise S. Monroe Library.  
<http://www.lib.loyno.edu/> (15-02-2002)
10. Milne, S. Implementing skip navigation.  
[http://www.dmag.org.uk/resources/design\\_articles/skip.asp](http://www.dmag.org.uk/resources/design_articles/skip.asp) (17-03-2002)
11. Minneapolis Public Library.  
<http://www.mplib.org/> (15-02-2002)
12. National Library for the Blind (UK).  
<http://www.nlbuk.org/> (15-02-2002)



13. Policies relating to Web accessibility.  
<http://www.w3.org/WAI/Policy/> (17-03-2002)
14. Selfish reasons for accessible Web authoring : AWARE Center : HTML writers guild.  
<http://aware.hwg.org/why/selfish.html> (17-03-2002)
15. Sloan, D. How to uncover a Web site's accessibility barriers.  
[http://www.dmag.org.uk/resources/design\\_articles/howtojudge.asp](http://www.dmag.org.uk/resources/design_articles/howtojudge.asp) (17-03-2002)
16. Techniques for Web content accessibility guidelines 1.0.  
<http://www.w3.org/TR/WCAG10-TECHS/> (03-04-2002)
17. User agent accessibility guidelines 1.0.  
<http://www.w3.org/TR/UAAG10/> (17-03-2002)
18. W3C HTML Validation Service.  
<http://validator.w3.org> (17-03-2002)
19. Web Content Accessibility Guidelines 1.0.  
<http://www.w3.org/TR/WCAG10/> (10-04-2002)
20. Why Web Accessibility?  
<http://www.dmag.org.uk/resources/why/default.asp> (17-03-2002)

## **Appendix 1: Quick tips to make accessible web sites**

(taken from W3C's site <http://www.w3.org/WAI/References/QuickTips/>)

Images & animations. Use the alt attribute to describe the function of each visual.

Image maps. Use the client-side map and text for hotspots.

Multimedia. Provide captioning and transcripts of audio, and descriptions of video.

Hypertext links. Use text that makes sense when read out of context. For example, avoid "click here."

Page organization. Use headings, lists, and consistent structure. Use CSS for layout and style where possible.

Graphs & charts. Summarize or use the longdesc attribute.

Scripts, applets, & plug-ins. Provide alternative content in case active features are inaccessible or unsupported.

Frames. Use the noframes element and meaningful titles.

Tables. Make line-by-line reading sensible. Summarize.

Check your work. Validate. Use tools, checklist, and guidelines at

<http://www.w3.org/TR/WCAG>

## Appendix 2: Table of results

Library (number of pages in three levels of a site)	Accessibility					User checks												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Dubrovnik (11)	-					*	*	*	*	*	*							
Karlovac (5)	-					*	*	*	*	*	*							
Križevci (4)	-	-				*	*	*	*	*	*	*	*	*	*			
Marin Držić, Zg (10)	-					*	*	*	*	*	*							
Bogdan Ogrizović, Zg (6)	-					*	*	*	*	*	*							
Medveščak, Zg (39)				-		*	*	*	*	*	*							
Split (11)	-		-			*	*	*	*	*	*						*	
Varaždin (15)	-			-		*	*		*	*	*							
Slavonski Brod (13)	-				-	*	*	*	*	*	*							*
Vinkovci (10)	-					*	*		*	*								
Čakovec (32)	-				-	*	*	*	*	*	*							
KGZ, Zg (77)	-			-	-	*	*	*	*	*	*	*	*	*	*			
Solin (13)	-				-	*	*				*							
Sisak (18)	-	-			-	*	*		*	*	*							
Bjelovar (21)	-	-			-	*	*	*	*	*	*	*	*	*	*			
Koprivnica (18)	-	-		-	-	*	*	*	*	*	*	*	*	*	*			
Zadar (116)	-					*	*	*	*	*	*							

### Meaning of numbers in column headings

- 1 Provide alternative text for all images
- 2 Provide alternative text for each APPLET
- 3 Provide alternative text for all image-type buttons in forms
- 4 Provide alternative text for all image map hot-spots (AREAs)
- 5 Give each frame a title
- 6 If you use color to convey information, make sure the information is also represented another way
- 7 If an image conveys important information beyond what is in its alternative text, provide an extended description
- 8 Provide alternative content for each SCRIPT that conveys important information or functionality.
- 9 If this is a data table (not used for layout only), identify headers for the table rows and columns
- 10 If a table has two or more rows or columns that serve as headers, use structural markup to identify their hierarchy and relationship
- 11 If style sheets are ignored or unsupported, are pages still readable and usable?
- 12 Provide accessible alternatives to the information in scripts, applets, or objects
- 13 Make sure pages are still usable if programmatic objects do not function
- 14 Synchronize equivalent alternatives with multimedia presentations
- 15 Make sure that the page does not cause the screen to flicker rapidly
- 16 If the submit button is used as an image map, use separate buttons for each active region

17 If ASCII art is present, consider substituting it with an accessible image