TESTING THE EFFECTS OF WEB USABILITY CRITERIA FOR VISION IMPAIRED USERS

BARBARA LEPORINI, FABIO PATERNÒ
ISTI - C.N.R.
56124 - PISA, ITALY
{BARBARA.LEPORINI, FABIO.PATERNO}@ISTI.CNR.IT

ABSTRACT
In this paper we show how it is possible to increase usability for vision-impaired users accessing Web applications. To this end, we have developed a number of criteria that allow designers to obtain usable Web sites for users who navigate through screen readers or other similar devices. We present the results of user testing performed to estimate the impact of our criteria on the Web interactions of vision-impaired users. The results indicate that their application can increase Web site usability for such subjects. Indeed, application of the presented criteria improved Web site usability both quantitatively and qualitatively, by reducing the navigation time and making the test Web site easier to use for blind/low vision users.

Categories and Subject Descriptors: H5.m. Information interfaces and presentation (e.g., HCI)
General Terms: Design
Additional Key Words and Phrases: Accessibility, usability, blind users, screen readers, guidelines, user testing

1. INTRODUCTION

In recent years, the interest in accessibility and usability issues has increased. In fact, the use of Web sites has been widening, and the number of users accessing them is steadily increasing. For this reason, it is important that the information be easily reachable by all, including people with disabilities. Moreover, for some user categories (e.g. blind users) the Web is one of their main sources of useful information about educational or cultural purposes. The difficulties in providing such universal access can be addressed through application of the principles of usability and accessibility. Accessibility is aimed specifically at making a Web site more available to a wider population of users (including special categories), by removing the technical barriers that prevent access to the information included in the site. To this end a number of initiatives at governmental level (such as the Section 508 [20]) and standardization level (such as the W3C WAI [25]) has been undertaken to stimulate awareness of such issues in Web interface developers and service providers.

However, accessibility alone cannot be enough to overcome the technological barriers. Usability aspects need to be addressed as well. Indeed, accessibility and usability are frequently addressed as two separate issues, even if vision impaired users need to have both accessible and usable applications. Only recently, have designers and developers become aware of the necessity of integrating these two intertwined aspects of Web site navigation. If accessibility and usability are not properly integrated, Web sites can turn out to be either accessible but barely usable, or usable but barely accessible: in both cases Web site navigation by disabled users is likely to be seriously compromised, since they either may not be able to access the desired information (i.e., it is not accessible) or may find difficulty in arriving at what they need (in this case it is not usable).

The usability of an interactive system is one key factor in its success, for example for increasing user efficiency, organization productivity, acceptance of new systems and safety, or for decreasing errors and the need for training. This aspect is particularly important for disabled people who cannot freely move. For these people Web services are fundamental because through e-commerce they can increase their personal independence, and likewise they need access to public services from their own home, tools for remote education and telework. Besides, other services are becoming more and more available via Web such as home banking services.

In this perspective, our research aims at identifying an appropriate method to overcome all those possible “barriers” preventing the usage of the online services from blind and low vision users. Such a method aims at advancing criteria for an appropriate design and an adequate evaluation of Web sites in order to make available and easy to access the online services to vision impaired users. Such users navigate trough screen readers or magnifying programs. To this end, we have proposed a set of criteria [13, 14] aimed at defining the meaning of usability when Web sites are accessed by visually impaired users. In particular, in order to bridge the gap between visual layout and aural perception, we proposed various designing principles for improving the interaction when using special devices (i.e. screen reader and magnifiers).

In this paper we provide a refined definition of our criteria, report the results of the test used for the empirical validation of such criteria, involving vision-impaired users, and present the outcome of a questionnaire filled in by users with information regarding qualitative aspects of navigation.

Usability testing provides an evaluator with direct information regarding the way people use applications and the problems they encounter when they use the tested interface. In our case, the test was conducted with blind and visual impaired people. The testing procedure adopted was based on two remote evaluation techniques (task-based testing complemented with a remote questionnaire) and was performed by using two Web site prototypes and two automated tests, each one composed of seven assigned tasks. During the usability testing, participants used the system to complete a pre-determined set of tasks while the system recorded (via log files) the results of participants' tests. Our tests revealed that when our criteria are applied, Web navigability is improved in terms of time saved by users, since the time spent looking for information or performing a task can be significantly reduced. Such empirical results provide an interesting feedback on the impact of the application of our criteria on final users.

2. BACKGROUND

2.1 Navigating through screen reader: visual layout vs. different perception

Usually the analysis of digital obstacles for the disabled only addresses accessibility, although usability is fundamental for simplifying both navigation and interaction for users using assistive devices or those with special needs. Blind users navigate the Internet by using a screen reader, which announces the page content (by voice synthesizer) or provides tactile information (through braille display); while low vision users can interact by means of magnifying programs. By using all these assistive technologies, the perception of the Web pages differs a lot from that received through the usual navigation without special devices. When navigating by a screen reader the user perceives the page content in a different way from its rendering on the screen. Such tools convert vocally all the information in the Web page through a linear scansion from top to bottom and from left to right. They also provide some support to access specific elements in the Web pages. This requires a certain expertise in advanced screen reader and browser commands and orientate oneself within the page content can require considerable effort. Blind users do not use the mouse function (i.e. pointing, scrolling, selecting, etc.) for moving around the page; but instead move by means of keyboard commands, such as Tab key, arrow keys, and so on.

Although sites may adhere to accessibility recommendations, users can still experience navigational problems. This is partly due to the fact that Web pages are increasingly designed for parallel or non-serial navigation, offering a variety of options within one page (frames, tables, drop down menus etc). Complex Web pages and parallel design can cause problems for users who are navigating the site using assistive technologies which force them to follow a serial (or linear) route, for example a screen reader reading out every hypertext link on a page one by one.

Our approach is aimed at focusing on main difficulties encountered by the visually impaired when they interact with graphical user interfaces through some assistive technology. In the specific, we concentrated on the interaction with Web sites. Then, starting with the main navigational problems for the blind and low vision users, we investigated for a possible solution by proposing several general design principles. Thus, we proposed various criteria by which designers and developers can appropriately design Web sites and pages.

In brief, the main problems for a blind person navigating through screen reader or magnifier are:

- *Lack of context* – the user may loose the overall context of the current page when navigating through screen reader/magnifier, since they can read only small portions of texts. For example, the blind user who is skipping from link to link with the tab key will read the link text on the braille display or will hear it from the synthesizer (e.g. ".pdf", "more details", etc.); however, the user will not be able to know what is written before and after.
- *Overloading information* – The static portions of the page (links, frames with banners, etc.) may overload the reading process through a screen reader: since the user has to read every thing almost every time, the navigation time can significantly increase.
- *Excess of sequential reading* – the command for navigating and reading can oblige the user to follow the page content sequentially. Thus, it is important to introduce mechanisms to facilitate the identification of precise parts within the page. At the top of result pages generated by a search engine, for example, the user usually finds several links, advertisements, search fields, buttons, etc. that the user should be able to skip to go directly to the search results listed below.

Thus, although the Internet is a precious source of information and offers great availability of services all these drawbacks can discourage blind and visually-impaired users from accessing on-line services.

The screen reader referred to in the following is Jaws for Windows [12] which is the most frequently used by blind people. JAWS gives quick access to the information in a logical format. JAWS also provides more information about the structure and organization of web pages. The information comes right from the Web pages. Thus, features in their implementation allow screen readers to provide detailed information about web pages. Hence, tables, lists, headings, and so forth are all listed vocally. Users can navigate within Web pages by the Virtual Cursor, read web pages using standard screen reader commands, tab through links and easily follow them, display lists of links or headings to quickly find and move to what you need, and easily move to and fill out web based forms. For instance, by pressing a letter the focus moves directly to tables, headings, or lists. Therefore, developers should build Web pages so that users can get advantages from special commands.

Summarizing, the main issues in UI design to consider are:

a) *Page content serialization*. General speaking, the screen reader takes the page source and serializes its content (link, edit field, button, cell, and so on). Also frames or blocks <div> are lined up, without taking into account specific positions assigned by CSS properties. Basically, the screen reader interprets the code as it was written and lines up the page content in the form of a single column. Thus, how the page code is written is very important.

b) *Navigation by Tab key and special commands*. It is important to remember that a blind user usually prefers to visit the page link by link (by Tab key) or using special commands in order to move quickly around the pages. Hence, it is important to support navigation via keyboard by assigning a scale of importance to the links, applying shortcuts to main elements, using specific tags such as <Hn>, and so on. Furthermore, many special screen reader commands operate well only if the developer has applied specific tags or attributes, or appropriate criteria have been followed.

c) *What is offered by a visual layout differs from one provided for aural perception*. Often when developers design a Web page they provide some useful information by means of visual features, such as position, color, separating blank spaces, formatting features, and so forth. For instance, some secondary information is put on the side so that users can recognize it immediately. It is important to provide the same "message" to a blind user by another means (e.g. using a table, a heading, a hidden label, etc.).

Considering all the above issues as well as accessibility and usability difficulties resulting from the screen reader, our proposed criteria attempt to structure and to organize the content and UI elements in order to have a better interaction through screen reader and an adequate layout as well.

2.2 RELATED WORK

Well-defined criteria and guidelines must be provided in order to assist designers in handling Web pages and guide them in the development of more usable and accessible Web sites. Up to now, usability and accessibility guidelines have been proposed separately, whereas we propose an integrated approach. Many detailed usability guidelines were formulated for both general user interfaces and Web page design (see for example [17, 24]). Most accessibility issues are currently considered by W3C (World Wide Web Consortium) in the Web Accessibility Initiative (WAI). In the WAI activities, a set of specific guidelines and recommendations has been defined: "Web Content Accessibility Guidelines 1.0" [25]. Currently, a new version 2.0 of Web Content Accessibility Guidelines as a Recommendation is in progress [26]. A number of tools (BOBBY [8], LIFT [23], A-Prompt [3] and WebSat [17]) have been proposed to identify accessibility problems mostly following the guidelines of Section 508 and W3C. LIFT and WebSat also support usability criteria for users without disabilities but do not support usability criteria for users accessing Web sites through screen readers.

Recently, various international projects have addressed issues related to accessibility/usability of interfaces for users with special needs. Stephanidis' group elaborated methods and tools for the development of unified user interfaces [22]. A "Unified Web Browser" was developed as part of the project “AVANTI”: it employs adaptability and adaptivity techniques, in order to provide accessibility and high-quality interaction to users with different abilities and needs (e.g., blind users or those with other disabilities). Even if this aspect is related to the subject of our research (in fact some checkpoints are similar), AVANTI browser mainly focused on accessibility and not on usability of Web sites accessed by people using screen readers. Goble et al. proposed [10] the use of the model of a real world travel in order to classify aspects in Web sites that are important for usability and accessibility. However, they did not provide design guidelines which can be helpful for site and page developers.

The Danish National Library for the Blind (DBB, the abbreviation for Danmarks Blindebibliotek) is an institution which provides services and materials to the blind, the visually impaired, and others whose handicaps prevent them from reading standard printed material. This group in Denmark focuses specifically on teaching electronic content providers how to make their Web sites accessible to people who have impairments of their sight, hearing, or motor skills. The Webcenter (founded by the DBB) on the basis of the WAI guidelines, and on their personal experience from testing hundreds of Web pages with assistive software and from working with blind and visually impaired users, provide some tips and examples of what Web developers need to be aware and to realize [7]. While some accessibility problems have been individuated, and some possible solutions have been suggested, a structured set of usability design criteria for visually impaired users has not been particularly focused.

Brewster's group at Glasgow University has been working in the project Multimodal visualization for blind people [15]. The Multivis project has the purpose to build up "visualization" systems allowing people with visual problems to get access to complex information (tables, graphics, structured information etc.) by means of different representation modes, in addition to the classic visualization technique. The goal is to investigate and solve this issue by the use of virtual reality, so allowing the users to "feel" and "touch" the visualized data. The research leverages previous works about earcons, audio 3D and haptics. Sound was specifically used to aid problems with buttons, scrollbars, menus, progress bars selecting text and files, etc. So, non-speech sounds had been used to improve especially the usability of a graphics package. We think that receiving as output sound signals associated to particular objects or about task status, is a useful output for blind users. Thus, in our work we have also considered non-speech audio in web site design in order to facilitate the navigation.

Regarding problems deriving from the visual layout perceived through special devices by the visually impaired, starting from the hypothesis that it would be most convenient for blind users to work with an auditive representation of a Web page, Donker et al. proposed an aural-based methodology, which has the highest possible resemblance with the visual user interface. The approach introduced in [9] attempts to provide an access not only to page content, but also to other important information coded in the layout of Web pages. To overcome the layout barrier, the authors concentrated on the interaction possibilities in an auditory interaction realm to represent the layout of Web pages and to support the navigation of the users. The auditory user interface of the system prototype was tested by involving seven blind users. The user testing did not produce excellent results. It revealed that the invited experts were not able to process their tasks more effectively and more efficiently with the proposed system compared to their presently used screen reader. Our approach takes into account the difference existing between visual layout and perceived layout through a screen reader, but is not aimed at providing specifically aural representation of the visual layout (like position or dimension); rather our methodology aims at providing the same type of information given through visual layout (position, colour, white spaces, etc.) by using other techniques (e.g. hidden labels, heading levels, etc.).

USERfit [1] is a method for generating usability specifications, purposely created for the assistive technology field. Nevertheless, it aims more at allowing information sharing among remote groups of designers, than at evaluating and repairing usability and accessibility issues in Web sites, as our criteria that mainly address usability aspects for special users who navigate through a screen reader.
The evaluation of Web site accessibility and usability by means of guidelines requires observing, analysing and interpreting Web site characteristics, similarly to other inspection methods used in the usability/accessibility assessment. Since the whole evaluation procedure is time consuming and requires big efforts, it is important to develop tools for automating the process of registration, analysis, and interpretation of accessibility data. Ivory & Hearts [11] made a distinction among automatic capture, analysis, and critique tools. Automatic capture tools assist the process of collecting relevant user and system information. Examples of such tools are Web server logging tools and client-side logging tools (e.g. WebRemUsine [18]), etc.. Many automatic evaluation tools were developed to assist evaluators with guideline review, by automatically detecting and reporting violations (usability, accessibility, etc) and in some cases by making suggestions for fixing them. Kwaresmi [5] and EvalIris [2] are examples of tools that allow designers and evaluators to incorporate new additional accessibility guidelines easily. Although KWARESMI is intended to enable the evaluation of any ergonomic guideline properly expressed in GDL (Guideline Definition Language), but at the present time it does not adopt a set of usability guidelines specifically intended for blind users.
Regarding the usability of Web site for users with disabilities, [4] reports a preliminary testing of the usability of GUI applications for blind and vision-impaired users. However, despite the progress in screen reader development, blind people still encounter many obstacles while using GUI applications.

2.3 USABILITY CRITERIA TO IMPROVE WEB ACCESSIBILITY FOR BLIND USERS

Web site usability depends on many aspects. In order to improve the navigability through screen reader and to make Web sites more accessible and usable, we have defined 16 criteria and 54 checkpoints. Criteria are intended for Web site developers and designers: they are general principles that can be followed to improve Web site accessibility and usability. Each criterion can include one or more checkpoints. Checkpoints are technical solutions that allow the application/evaluation of our criteria and usually correspond to specific implementation constructs that guarantee the satisfaction of the associated criterion. For example, the criterion “Logical partition of interface elements” expresses the concept of well-structuring and organizing the page content. So, the criterion provides the general principle which should be taken into account by developers during the Web site design. Then, developers can decide how to apply that criterion. Usually several solutions can be adopted. For example, the page content could be structured by using frames, or blocks <div> customisable by css properties. Alternatively, the content within the page could be visualised by embedding it in layout or data tables. Besides, long page content could be partitioned through heading levels, or specific page parts could marked with “hidden labels”. So, all these cases apply the same general concept (i.e. partitioning the content) by using different technical solutions. Hence, the general concept is the criterion, whereas all the solutions are given by the possible checkpoints for implementing the criterion. In some cases all checkpoints must be applied to satisfy the criterion; in other cases just one checkpoint is sufficient.

In defining our criteria, we aimed at identifying the main aspects that can cause usability problems in Web site navigability through special devices. The main navigational problems that can arise when using a screen reader or magnifier, as mentioned in the introduction, are lack of context, information overload, and excess of information sequential reading. Our principles attempt to address these usability problems. Criteria are classified and grouped into five logical dimensions based on general issues which should be taken into account during design of Web sites: structure and arrangement, content appropriateness, interactivity, multimodal output, and consistency. In practice, various aspects should be considered for improving the navigation when interacting through keyboard as well as special devices such as screen reader and magnifying programs. Therefore the proposed criteria are based on features such as page structure, user interaction and the content clearness, which address important issues for a good navigation.

Below, the whole set of criteria is summarised grouped by the five logical dimensions above mentioned. The presented criteria intend to be general principles for Web designers/and developers and should be used during the development phases of a Web site. An example of application of such criteria is described in the next section. Many criteria visually affect the Web interface (e.g. coloured areas or element magnifications), whereas other ones can be detected only by the screen reader (e.g. hidden labels or names of frames). We used the format I.J to identify each criterion: I indicates the group to which the criterion belongs (from 1 to 5); J is a progressive number to enumerate the criteria ($j=1..N_i=6|2|1|5|2$).

2.3.1 Structure and arrangement

This set of criteria is aimed at organizing and structuring the user interface elements and the page content so that the navigation is easier. The main difficulties that should be solved through these criteria are the user’s orientation and the capability in getting an overview of the page content. Navigation and structure of the Web site that are easier to access are considered as well. The criteria of this set are:

Logical partition of interface elements (1.1)

this criterion aims at grouping information, links, fields and so on in logical categories, in order to allow users who read the page content through screen reader to localize the essential parts in the page. Moreover, some screen readers allow skipping from section to section. This can be obtained by using markers, frames or headings to group texts, links, forms, and so on, according to a logical division. In addition, in order to reach more easily some location in the page (or in the site) we can insert local navigation links referring to bookmarks in the scope of the page (e.g., 'skip to content', 'go to top', 'go to navigation bar', etc.). And also, in pages containing information of different kinds (paragraphs, news, etc), a local index, such as local page links or a drop-down menu, is suggested to be added in order to navigate more efficiently among the page sections. Another feature particularly critical for blind users who cannot perceive the page overview is the localization of the 'main page content'. Some possible strategies to use in order to better identifying the 'new page content' are suggested.

Number of links and frames (1.2)

it is important that a page does not contain too many links or frames, as this makes it difficult for the user to skim through them all. Pages should have neither too many nor too few links: lots of links take a long time for readers to get through, too few links may imply too many levels in the Web pages' hierarchy structure.

Location of the navigation bar (1.3)

the so-called navigation links (i.e., the links appearing on each page and enabling users to reach the main parts of the site) represent a source of delay and inefficiency for screen reader users. Since such links appear on each page (and often even twice), the users who are forced to read the contents in an almost sequential way are always compelled to skim them before they can identify the contents of the current page. Therefore, highlighting the navigation bar at the top and/or the bottom of the page, if any, can be useful to make it more understandable to users who are unable to see its visual features (e.g., horizontal or vertical position, colour or font types, etc.) and can increase navigation efficiency for these users.

Importance levels of elements (1.4)

in order to facilitate navigation, especially when using the keyboard, it is possible to assign different importance values to interaction elements such as links, buttons, and fields. This way, when users move through the 'Tab' key (element by element), they visit at first the most important, and later the less important, regardless of their location on the page.

Proper form layout (1.5)

in forms with several groups of data, we must properly lay out group titles and fields to achieve a major clearness. In fact, the way the elements are formatted can cause confusion with the screen reader. For example, in some cases the voice synthesizer or Braille display could read before the 'checkbox', 'combobox' or 'field' item, and after its value, or vice versa. Thus, a correct application of layout elements (e.g., simply by using the label tag in the proper place) is recommended.

Specific sections (1.6)

in sites with frequent information update and/or new resources to download, we can help the user to find more rapidly the new elements by providing a specific section listing the new elements by date, sparing the user the trouble of useless navigation. Furthermore, a specific page listing all short keys associated to the most important links of the Web site should be considered.

2.3.2 Content appropriateness

When perceiving content through a screen reader, especially by using a vocal output, making the content clear and auto-explicative is very important. In fact, the global overview is loss, and usually a blind user moves through Tab key or screen reader special commands in order to obtain a fast navigation. This means that the user reads the content quickly and jumps from link to link, from table to table, and so on. For this reason it is very important providing an appropriate content for textual and graphical links, for table summary, for pictures and images, paying particular attention for those with a functional purpose rather than a decorative scope. Criteria included in this group are:

Proper link content (2.1)

the link labels are important for special users who use screen readers and keyboard commands. Thus, the links should be clear and context independent, and not having general texts such as 'more details', 'download', '.pdf'. We must warn the designer that such texts can lower the site's usability, because they are ambiguous, or not enough informative.

Proper name for frames, tables and images (2.2)

it is important to check that all frames, tables and images have names and descriptions which are appropriate and meaningful; e.g., frames with names such as "top" are not very helpful to the user. On the other hand, names such as "index" can make it easier for users to reach their goals. Similar considerations apply to text summaries of tables and to alternative description for images and graphics.

2.3.3 Interactivity

The interactivity between the user and the interface is important to improve the navigability. Thus, these criteria are aimed at adding design features and providing technical solutions that allows to users to better identify specific sections or to reach links more easily. The criteria of this set are:

Assignment of shortcuts (3.1)

It is advisable to assign hot keys to the most important buttons, links and fields, so that the user is able to reach them quickly through a simple key combination. This feature may be useful especially when users visit the Web site frequently, and learn the key combinations by heart.

2.3.4 Multimodal output

The goal of the criteria belonging to this set is to provide blind or low vision users with several kinds of feedback. Using different colours, changing colours or dimension for focusing the current content, adding different sounds can help the visually impaired user. In fact, particular visual or sound cues can represent good feedbacks for user's orientation. Also, providing various kinds of layouts for different devices, such as for voice synthesizer and Braille display or embossed, is also very important.

Messages and dynamic data management (4.1)

A significant difficulty often encountered through a screen reader is represented by poorly designed system feedback. This may be confirmation or error messages about instructed operations or requested information, which are not presented in a manner that can be rapidly interpreted by screen readers (e.g., messages in the middle of the page or amongst a lot of other information), or they are often so "short" that they are not easy to focus on. This principle suggests paying attention to this aspect.

Loading suitable style sheets (4.2)

Browsers can load specific sheets for different output devices. This style sheet feature allows specifying how a document is to be presented on different media, such as screen, paper, speech synthesizer, Braille device, etc.

Addition of short sounds (4.3)

Associating a short sound to different elements and different kinds of a multimedia environment can give a useful feedback. E.g., providing each page with a short sound indicating when the loading of the page is completed eliminates the need for the user to repeatedly check the status bar.

Colour of text and background (4.4)

For low vision people It is advised to avoid colour combinations giving a poor contrast. Furthermore, changing colours in correspondence to some events, or particular areas, can be a way to get attention.

Magnifying at passing by mouse (4.5)

The use of this feature can help people with a good visual residue to better focus on the pointed object. The idea is to enlarge particular elements such as images, navigation links and buttons, not all text.

2.3.5 Consistency

Keeping the content and layout consistency between Web pages is in general very important for the user. This is particular significant for visually impaired users because they rely on consistent features. When users navigate the Web site, they learn the features common to the various pages. This allows them to not explore the page content, but to look for a specific element without reading the content.

Layout and terminological consistency (5.1)

Consistency is a usability feature that allows users to better understand the context and the available functions. It is important that all the pages of the whole Web site adopt the same labels for buttons performing the same tasks (e.g., OK/Yes, quit/exit), and that all pages have the same layout (e.g., dimension, form and colour).

Page information (5.2)

Defining the beginning and ending of the page by adding useful information can make the navigation more pleasant. For instance, the title of the page, which is read as first line, could contain not only the Web page title, but also additional indication like the current path. Also the last line of the page could contain information so that users are able to understand that is the last line in their sequential reading..

3. WEB SITE PROTOTYPE

For our testing, we considered a Web site containing specific information about the “The Tuscan Association for the Blind” (Unione Italiana Ciechi – Regione Toscana). This testing site was chosen with the intent of providing blind people with a comfortable situation with familiar information, thus reducing navigation difficulties.

We created two versions of the same Web site: one version was implemented according to the 16 criteria and for this reason it will be referred as the “revised site”, while the other version, without criteria, was used as control in our testing protocol (“control site”). The time required for performing the same kind of tasks in both cases was recorded.

The two Web sites had three main sections, i.e. “News”, “Documents&Download”, and “Organization”. Each section was reachable from each page and was composed of a variable number of subsections. The general layout of the page included (an example is shown in Figure 1):

- a navigation bar at the top;
- a submenu or local index at the left;
- the current page content at the right (it is the largest area);
- navigation links (“Go back”, “Go to navigation bar”, etc.) at the bottom.

Figure 1 - A Web page used for the user testing.

The Web site is aimed at collecting several information and services which can be useful especially for blind and low vision people and as well as for all those who like to get information on vision issues. Thus, the Web site provides information like periodical bulletin with several news; download sections containing various documents and local and external files (e.g. utilities, screen readers demo versions, etc.); some catalogues regarding works available in braille format; some information related to the structure and organization of the association and so on. All these data are organized in several sections, which can be accessed by links of the navigation bar.

The application of the criteria to one of the two versions of the testing site included the following aspects:

- *Logical partition.* Heading levels were often used to structure the information in a logical way, but in some cases hidden labels or tables with summary values were also applied. For instance, navigation bar and submenu were marked with appropriate labels (e.g. “navigation bar:” and “submenu:“). Then, in pages containing various file information and links related to downloadable manuals or programs, each data group was placed in a specific table with group names as summary attributes.
- *Proper link content*. For textual links “title” attributes were used when link labels were not appropriate by themselves; in the other cases, graphical and textual links were used together. For instance, a graphic link with an appropriate icon was combined with an appropriate text to obtain a download link: the word “download” was assigned to “alt” attribute to graphical link, and the name of the file to be downloaded was used as label of the link; in this way, the visual link is represented by a symbolic icon close to the name of the downloadable program (e.g., Download Acrobat Reader 5.1).

- *Messages and dynamic data management*. A new page was used to provide information about the sending status of a form.
- *Layout and Terminological Consistency*. All links, buttons and pages had the same features within all pages. In addition, all pages had the same template, such as the navigation bar at the top, the left side submenu (when necessary), the standard size and colour buttons, and so on. Furthermore, the terminology was maintained within the pages, such as "Homepage".
- *Number of links and frames*. The Web site prototype was built with links and without frames.
- *Proper name for frames, tables and images*. In the prototype, style sheet properties were mainly used for organizing the content in the page; tables were used only to render data about download file information (e.g. programs, manuals, documents, etc.). A table was used to group set of files belonging to the same category, in order to obtain a table for each category. In this way, the "summary" content could be represented by the name of the category itself.
- *Location of the navigation bar*. In the prototype, both navigation bar (at the top of the page) and a submenu (at the left size of the page) were added to the page layout and hidden labels "navigation bar" and "submenu" were used to mark bar and menu beginning. In addition, the submenu was built by including item links in a bullet list: in this way users were able to identify its structure more quickly.
- *Different visiting order of elements (*Importance levels of page elements *and* Keyboard shortcuts*).* In every page, except for the home page, the lowest visit order value was assigned to navigation bar links (i.e. the highest tabindex value), whereas greater priority was assigned to the submenu and other links. In this way users could first visit links associated to submenu/more recent information and afterwards go to navigation bar links. Navigation bar/link shortcuts were also associated to navigation links.
- *Proper form layout*. A specific page, built by using appropriate tags, contained a form to fill and send for obtaining information, make suggestions, etc. Besides, CSS properties (and not tables) were used to render the set of pairs <label, edit field>.
- *Specific sections*. In our Web site prototype, "Last updates" and "Key list" sections were added. The "last update" section was reachable from the home page, whereas the "Key list" page can be found in the navigation bar of each page.
- *Indexing of contents*. In the prototype, two types of content page indexes were applied: (1) a list of local links was added when a page contained several blocks of information; (2) a drop down menu was added when a page contained many block of information identifiable by characters from "a" to "z" (e.g., a list of song titles). A similar solution was applied to the page created for collecting a set of documents organised by date: in this case, the index was built by assigning a specific menu item to each group of documents belonging to the same year. In short, the first type of index was used when only few links were necessary to build the local index, while the second type was applied when a high number of links was needed.
- *Navigation links*. Links as "skip to content", "go top of page", "go to navigation bar", "go to submenu", and "go back" were added to each page to facilitate the navigation. Furthermore, the two links "go to navigation bar" and "go to submenu" were made "invisible", since they are useful only for navigation through screen reader or keyboard.
- *Main page content identification.* Three techniques were used to identify the main content of the current page: (1) a link "skip to content" was added to each page; (2) the first line of the main page content was enclosed between `<H1>` and `</H1>`

tags, and (3) the indication of the "#content" bookmark was added to the URLs link in order to move the focus directly to the current page content once the page was loaded.

- *Addition of sound.* Different sounds were associated with the selection of different types of links (i.e. local, internal to the site, external to the site).
- *Visual features (Colour of text and background, and Magnifying at passing by mouse).* We used different colours for distinguishing navigation bar, submenus, and links currently pointed to. The navigation bar was arranged horizontally at the top of the page in a blue area; the submenu links were placed vertically at the left of the page and separated by the page content by a vertical black line. When the mouse hovers over the links, link size and colour change.
- *Additional information.* In each page title, the page path was also included. E.g., for the program downloading page the title was: "Home :: Documents&Download :: Program downloading". Furthermore, each page had the "last update information" on the last page line.

Figure 1 shows an example of page of the Web site prototype: the navigation bar, submenu, and current link magnification are also displayed.

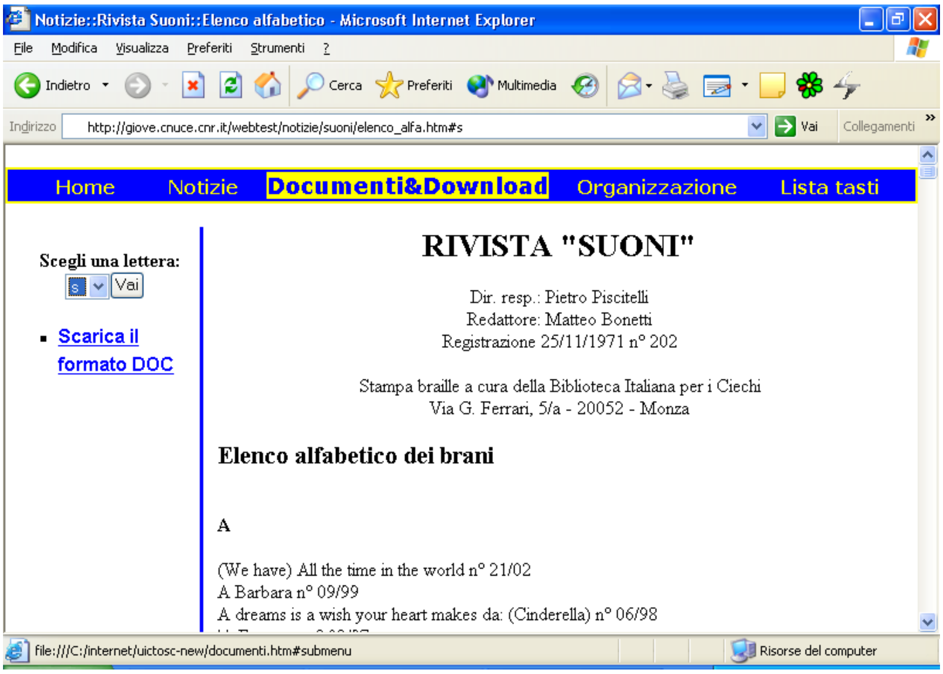


Figure 2– Page showing a list of song titles divided by headings and with a side list to facilitate navigation.

Figure 2 shows a page containing a list of song titles in alphabetical order. The titles are grouped by letter, each of which is associated with a heading level <H3>. This means that blind users can go directly to the titles beginning with any given letter by accessing the headings through a specific screen reader command (see Figure 3). Then, they can select the characters of interest (in this case “s”). For low-vision users, a drop-down menu has been inserted (on the left side), which allows character selection by mouse as well. This solution is preferable to creating a link for each character (as is often done), because the twenty-six links generated would make navigation more difficult for those interacting through a screen reader and Tab control.

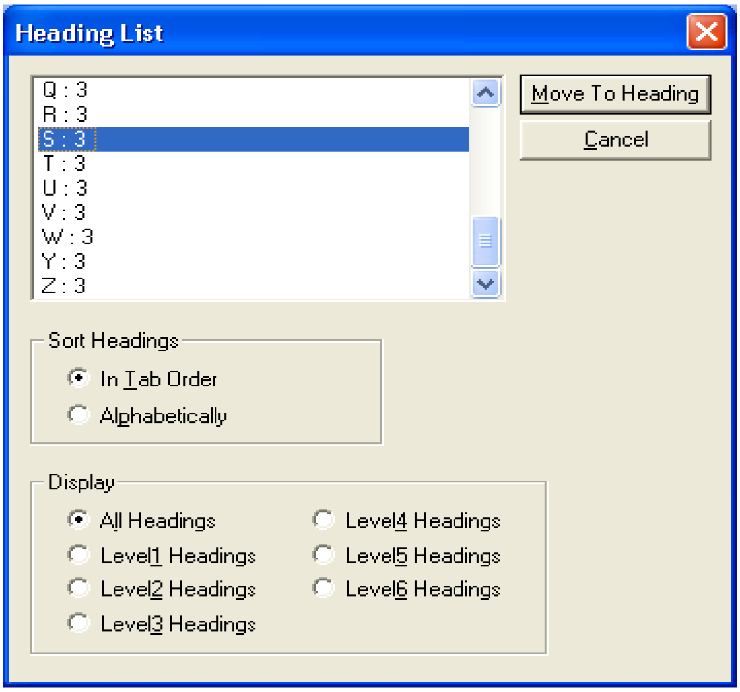


Figure 3– List of heading levels generated by a screen reader command allowing users to jump to the lettered group.

Lastly, Figure 4 shows an excerpt of text generated by the screen reader that highlights some criteria that have been applied. The first row read by the screen reader corresponds to the <title> attribute, which contains the path to the page (e.g. News::Sounds Magazine::Alphabetical List), as well as the title of the page. Such a solution enables users to navigate more easily because the information is immediately read by the screen reader (or in any event, is rapidly found by the screen reader because it corresponds to the first row). The blind user can then check that the screen reader has identified the hidden labels (“navigation bar” and “submenu”) used to identify the various components of the page.

News::Sounds Magazine::Alphabetical List

Graphic Navigation bar:
Link skip to content alt+0
Link Home alt+h
Link News alt+n

Graphic Submenu:
Choose a letter:
Combobox S
Button Go

...
Alphabetical list of musics
Heading level 2 A
(We have) All the time in the world n° 21/02
A Barbara n° 09/99
A dreams is a wish your heart makes da: (Cinderella) n° 06/98
...
Heading level S

Figure 4 – Part of the page read by the screen reader (translated in English). The parts in italics are read by the screen reader but do not appear on the page shown by the browser.

4. EXAMPLES OF USER INTERACTION

In order to show how the criteria affect the user interface, we can consider two scenarios associated with the two Web sites. The goal is to show the main advantages achieved by the proposed principles when the site is accessed through a screen reader.
To this end, the scenarios consider the "Program download page" developed into the two Web site prototypes. From this page, users can select and download various files (e.g. programs, utilities, screen reader applications, music applications, and so on). Practically, the Web page contains several files that could be classified in various categories: utility, multimedia, screen reader, etc. In situations like this, the main problems encountered when navigating through a screen reader can be related to:

- The page contains a very long list of files: the user has to navigate within the page sequentially as there is no mechanism allowing to move more quickly;
- The text of the file links is not clear (e.g. "450nodongle.exe") or all links have the same text (e.g. "download"): although the user moves through Tab key, he/she has to read in the surround of the link for understanding if it is the wanted program;
- All the files are listed without any grouping: the user has to read in sequential way (both through Tab key and arrow keys) because it is not used any technique for grouping the files by categories;
- There are no specific links for reaching more quickly a given category: even if links were organized in various groups, the user has to spend a lot of time for moving among them if any appropriate design is applied.

It is worth to note that in all these cases the page content can be considered accessible (i.e. no particular barriers prevents the access to the resource), but the navigation is not easy for unskilled users. This particular case study was tested through our Web site prototypes. In the "control site" the negative drawbacks described above are present. In the "revised site" a possible improvement is suggested by applying our proposed usability criteria.

4.1 Scenario 1: user interaction with difficulties

Sarah is a blind user who wishes to download the latest version of the screen reader Jaws she uses for interacting with the computer. Instead of visiting the official Web site, which is in foreign language, she prefers to access the available Web site of the Tuscan association for the blind, which has a page containing links to several programs and utilities. She accesses the home page and looks for the “program download page” through “Documents&Download” site section. When the page is loaded by the browser (see Figure 5), the screen reader announces that the page has 20 links and it starts in reading in sequential way the page content: first the page title “download di programmi” (i.e. the <title> tag content appearing as the browser title); next the navigation bar links (i.e. “home”, “news”, “documents&download”, etc) are listed. Then, after the page content title “Download di programmi” (i.e. the first text block under the navigation bar) and the name of the first category “programmi di utilità”, Jaws reads the beginning of a table announcing the content of the summary and the number of the rows and columns:

> Summary: The table is used to organise the information on files containing programs that can be downloaded.
> Table with 3 columns and 4 rows
> Description
> File
> Dimension
> Link Download
> WinZip 8.0
> ...

At this point Sarah stops the Jaws reading, because she realises the page probably contains various program categories and that several tables may have been used for rendering the content. In fact, she has already read first a possible name of category “utility programs”, followed immediately by a table whose columns are “description”, “file name” and “dimension”. So, Sarah supposes that there could be various tables on the basis of available categories. As she does not want to read the whole page content, she decides to try to use some specific jaws commands for moving within the page content in order to reach more quickly the wanted information (i.e. the link for downloading the latest version of jaws). Hence, first she activates the link list through the special Jaws command Insert + f7, but, apart the links belonging to the navigation bar, only links with “download” text are listed. Therefore, Sarah comes back to the top of the page and she begins to move through links using the Tab key. Jaws reads first the links of the navigation bar (i.e. “home”, “news”, and so on), and then it reads only links like “download” and nothing else. However, in this way it is possible to explore also the surrounding link content. Sarah can move using the Tab key and hears “link download”, then she presses again Tab key and again she gets “link download”, and so on. The additional possibility is that when the cursor is on a link, she can also move it using arrow keys and read the adjacent text, i.e. she can explore around the “download” link. In this way she can read the file description associated to the current link. Sarah understands that in this way it is not certainly easy to find the desired link. Therefore, she decides to proceed differently. As she supposes that the program files are grouped by typology, she decides to look for a category which could contain the link for downloading jaws. In practice, Sarah decides to move quickly towards the category section, next she will explore in sequential way just that area content. Thus, she starts to jump from table to table supposing that each category group is embedded in a single table. So, Sarah uses the specific Jaws command “t” for jumping to next table. However, after having pressed “t” for some times, she realises that all tables have the same summary content, i.e. “the table is used for rendering the file data”. For reading the category name she has to move by using the up key. So, again she is not able to find easily the category containing the Jaws download link. Consequently, she can only jump from link to link, and then exploring in sequential way through arrows keys the surrounding content. Unfortunately this activity requires a lot of time and a considerable effort. For Sarah the navigation becomes tedious and she might also decide to give up her task.

If we look at Figure 5 we can understand the reasons for the problems encountered by Sarah. The Figure shows the “program download page” of the “control site”. As we can see, the page contains several links of various typologies. The links appear grouped by application typology, i.e. each group is preceded by the group name like utility programs, screen reader, and so on. Nevertheless, no appropriate methodology has been used for exploiting such categorization when interacting with a screen reader. For a non-blind person is easy to individuate each category by simply scrolling the page. The user who navigates through a screen reader has to behave as if no grouping is available. In fact, if no specific technique is applied, the user is not able to get an overview of the page content as well as of the existing groups. Therefore, the users navigate sequentially throughout the program list unless a search can be made within the page content, provided they already known exactly what they want. Consequently, reading the list requires long time and is not particularly practical. Thus, an appropriate arrangement of the file links in several sets is advisable. However, it is also very important that the grouping is made in a correct way, i.e. by applying specific solutions. In the “revised site” a possible technique is used (see next paragraph).

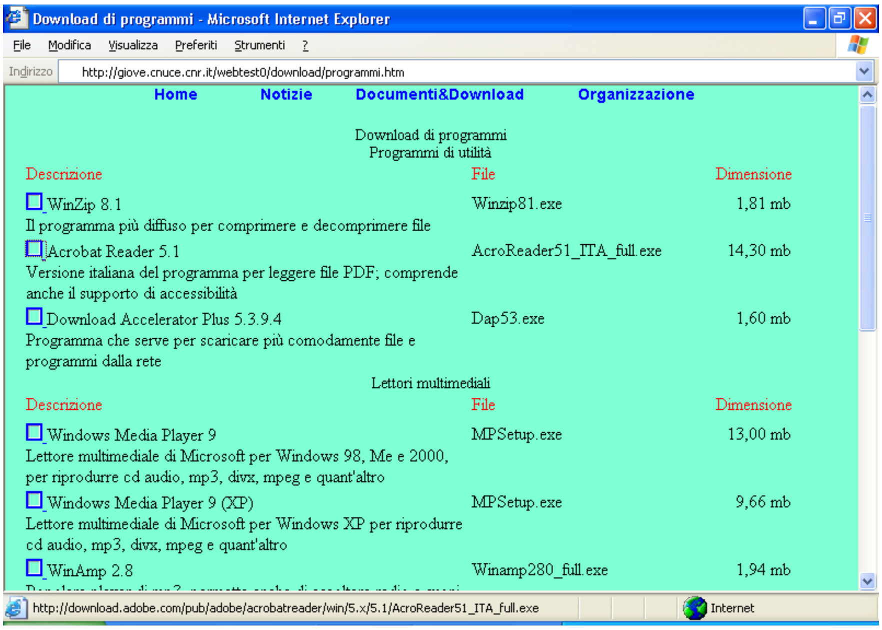
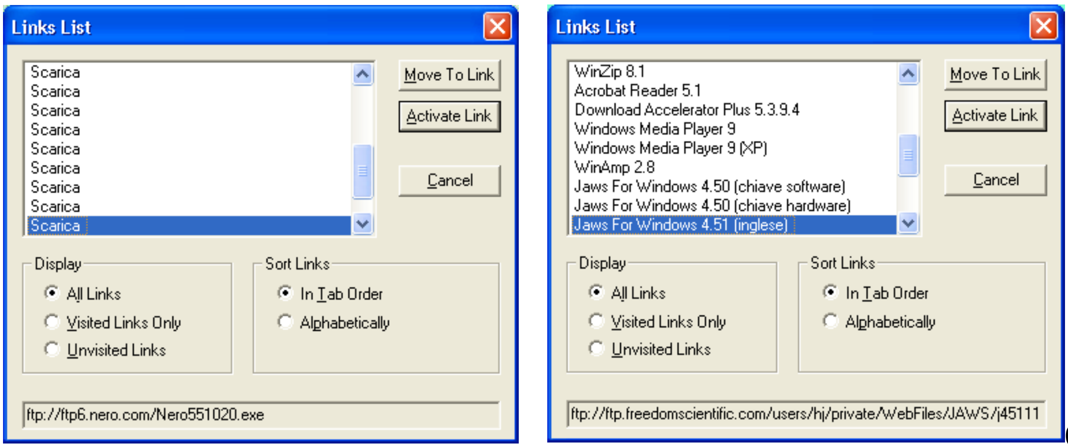


Figure 5 - Control site: program download page where several kinds of files can be downloaded

The other critical aspect occurring when several links are listed in a single page is the text used for each link pointing to the downloadable file, i.e. we refer to the text enclosed between <a> and for textual links, or the content of the attribute alt for graphical links. According to our proposed criteria, the content used as link text should be context-independent and auto-explicative. For example, if all links have the same content (e.g. pdf document), or not clear (e.g. a complicate file name), the user might have difficulties in understanding them. Usually the developer chooses a symbolic picture or the name of the downloadable file as content of the links. If the alt attribute or the name of the file are not clear, the navigation requires long time, because it is necessary to explore the content surrounding each link. In our “control site” all the links are graphical (an image of a “floppy disk” to render the downloadable concept), having all the attributes alt with the same value “scarica” (i.e. “download”). When users move through Tab key or accesses the link list (see Figure 6(a)), they hear just “scarica” repeatedly, without getting any information about the downloadable files. Hence, the navigation has to be carried in sequential way (through arrow keys) in order to read the surrounding content like Sarah did. In fact, reading line by line, information like file description, name and dimension are available. Therefore, users move through Tab key on a given link, they hear just “download”, and at this point, they go down with key-down to read the program description. If it is the wanted link the user has to come back over the link in order to click on it. Otherwise the user continues to move by Tab key on next link and then to explore sequentially.

Therefore, using the link list generated by a specific screen reader command is not practically possible, because all the links have the same description. Thus, the sole practicable possibility is going on in sequential way. Concluding, proceeding in this way means making the page navigation tedious and slow as well. Remembering that for a blind person the reading is done by small text portion, and for getting the surrounding content a sequential exploration is needed. Since this might require some efforts and a lot of time as well, users could not be able to finish their task and therefore to achieve the wanted goal.



(a) b)

Figure 6 - List of the links related to downloadable files: (a) from the "control site (non-correct solution); (b) from the "implemented site" (suggested solution)

4.2 Scenario 2: Easier user interaction

Carlo wants to search and download the latest version of Jaws as Sarah did, and he accesses the "revised site". Carlo opens the home page of the Web site and selects the "documents&download" link in order to access to the download section and in specific the "program download page". As soon as the page is loaded by the Web browser, the screen reader Jaws informs Carlo that the page has 26 links and 5 headings; next Jaws starts reading the sentence "submenu:" followed by links like "utility programs", "multimedia applications" etc.. So, the first useful information that Carlo gets is that the page has 5 headings. This means that the page is partitioned in several sections, which is useful to have a general page overview. Thus, Carlo first opens the dialogue window listing all the headings used for structuring the page. Carlo moves up and down in the heading list through arrows keys and he understands easily that each heading level is associated to a category group. He realises that the first heading level is assigned to the page content "Download di programmi" and the other four correspond to four file groups. Carlo goes down with the down arrow on the heading item "screen readers" and presses enter key. The dialogue window closes and the Jaws focus is on the heading "screen readers". At this point Carlo moves through the Tab key among the available links for downloading different versions of Jaws. More in detail, when Carlo moves through Tab key, Jaws reads the link contents like "Jaws For Windows 4.50 (chiave software)", "Jaws For Windows 4.50 (chiave hardware)", "Jaws For Windows 4.51 (inglese)", and so on. Therefore he localises easily the desired link and clicks on it for activating the download process. Carlo chooses to move by heading levels reading them in the dialogue Jaws window. In any case, Carlo would have two others possibilities for skipping to the "screen readers" section. In the first one, he could move among the local links pointing to each section, which is a local page index by pressing Tab key. These local links form a left-side menu particularly useful for a low vision user. In the second one, Carlo could jump from table to table: the screen reader reads the summary value, i.e. "utility programs section", "multimedia applications section" and so on. So also through this strategy he could reach very quickly the wished file category group. Next he can simply skip link by link through the Tab key since the link content is auto-explicative.

Hence, why did Carlo achieve his goal more quickly than Sarah? Which specific design criteria have been applied to the Web page? In the "revised site" the same "program download page" is structured so that the links are clearer and properly grouped. First of all, the links are arranged in sets according to the associated downloadable program typology. Each set of file links is embedded in a table. Every table has an appropriate attribute summary with the name of the program typology: there is the table "utilities" which contains all links pointing to utility file (e.g. winzip, acrobat reader, etc.); the table "screen reader", which groups all the links regarding the download of the demo version of the most used screen reader (e.g. jaws in various versions), and so on. In this way, the user by simply pressing a particular screen reader command (i.e. key "t") can jump quickly from table to table. The screen reader announces the summary value that is the name of the file category. In this manner the user can rapidly skim through all the categories and find one easier. When the desired group is reached, the navigation can be continued through Tab key or arrow keys. For example, if the "program download page" contains seven file groups and the target is the last, it is sufficient pressing the letter "t" 7 times to reach that group. In the previous situation, i.e. with the "control site", the time required is proportional to the number of links available within the page if the users move through Tab key, or even more if they proceed through arrow keys. Therefore the navigation results simpler. Adding an appropriate summary value to a table is very simply and moreover it does not appear on the screen. In addition, a submenu composed of local links pointing to each table (i.e. group) has been implemented and it is placed as left-side vertical menu (see Figure 7). That submenu is used like a category index, and it is designed especially for low vision users navigation. When a magnifier program is used to explore the Web page, a set of links can be more easily accessed. Thus, the user can click on the link associated to the corresponding category in order to focus the associated table. Clearly those links can be used also when interacting through screen reader, but usually unskilled blind users prefer exploring the page through other commands rather than clicking on local links. In any case, they can use indifferently the two possibilities: jumping from table to table with the letter "t", or clicking on the local links.

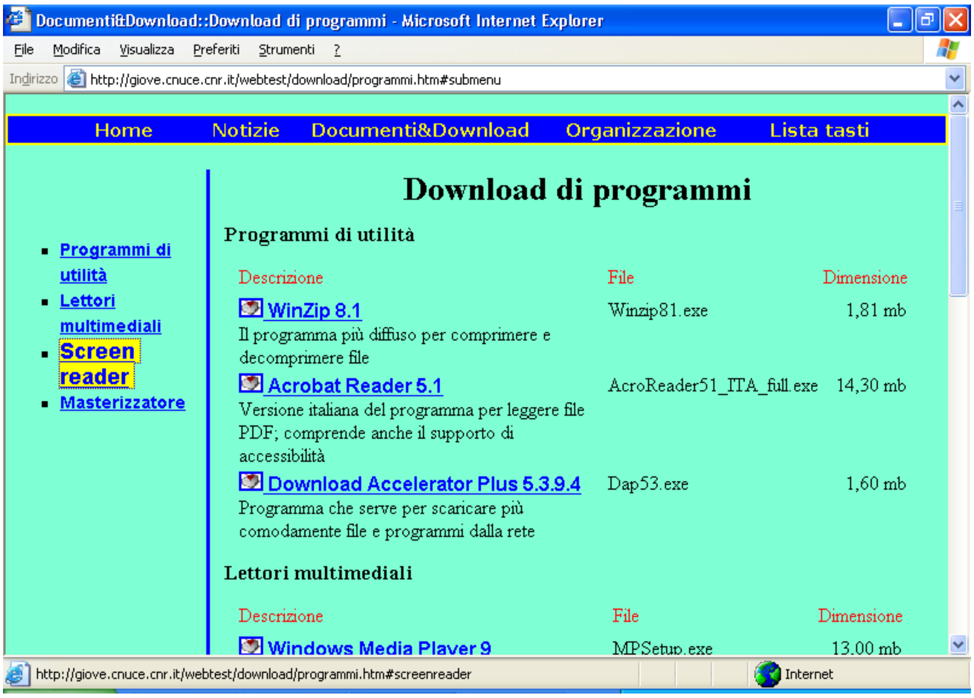


Figure 7 - Revised site: program download page where several kinds of files can be downloaded

Regarding the link content a more descriptive text has been used. Now the user can more speedily locate the desired program. In fact for each link associated to a program file, the content is composed of both the "floppy disk" image and the name of the referred program. Thus, when the user moves through the Tab key, the program name can be heard. Besides, it is now also possible using the link list (see Figure 6(b)) generated by the particular screen reader command (using insert + f7 if Jaws is used). The link list can be explored by the arrow key, link by link; or alternatively a specific link can be found through the first letter of the word. For instance, by pressing few time the letter "a" the "Acrobat reader" program link can be found without difficulty. Consequently the navigation results easier as well as faster. Lastly, note that although this second site version has been implemented by applying proper usability criteria, the visual layout is not particularly different from that of the "control site".

5. TESTING

5.1 Method

Twenty users with total/partial vision deficit were recruited for the testing. All the participants had been using Windows 98/ME and Jaws (as screen reading application) for at least one year at the moment of the testing. Thus, it could safely be assumed that they were adept at using the combination of a screen reader and Windows with the Internet Explorer browser.

Half of the participants were blind and the other half had a partial vision deficit: in any case, no-one could spot elements on the screen without an auxiliary support. The experience with the screen reader was extremely different within the group of participants, their level ranging from beginner to expert. For this reason, two phases were included in the experimental protocol.

Preliminary phase: participants were provided with a summarising list of the most important Jaws commands and were allowed to explore both Web site versions (with and without our criteria) for a week before the testing.

Testing phase: our testing procedure was based on remote evaluation, which is a kind of evaluation performed when evaluators and users are widely separated in time and/or space. Remote evaluation allows designers to analyse the performance of users that interact with applications in their familiar environment.

Two remote evaluation techniques were included in the procedure: a task-based testing and a remote questionnaire.

- The task-based testing was subdivided into two sessions, one conducted on the "control site" ("*session0*") and one on the "revised site" ("*session1*"). The two tests were located online, so that users were able to connect to them from their own computers. Users' interactions were automatically collected during the test run. Half of the users were asked to start from "*session0*" and the other half from "*session1*". Thus, the order of test performance was session0-session1 for 10 users and session1-session0 for the remaining participants.
- The remote questionnaire aimed at evaluating qualitative aspects of Web site usability, such as user satisfaction or enjoyment.

The two different techniques allowed us to collect two kinds of data: *Objective* – Time spent by users performing assigned tasks; *Subjective*– Users' preferences, opinions, and suggestions.

The experimental protocol was designed to avoid two kinds of bias:

- bias of ability: discrepancies in navigation abilities within users, associated to a different degree of individual training, could affect the result of the test: the preliminary phase allowed the participants to start the testing procedure with similar basic skills;
- bias of familiarity: extra navigation familiarity gained by users during the first test could influence the results of the subsequent test (e.g., users could become quicker in performing tasks during the second test). Preliminary navigation on both Web site versions reduced the surplus of experience possibly gained during the first test. Furthermore, the reversed order of test performance in half of the participants (*test0-test1*/*test1-test0*) contributed to smooth the effect of possible residual bias of familiarity.

5.1.1 Logging Tool

An automatic logging tool was used to record the time spent by users carrying out the assigned tasks. Our tool is an adaptation of the logging tool which was developed for the WebRemUsine tool [18]. The main interaction activities performed by each user during the testing procedure were captured and logged.

The tool included Javascript functions, java applet, and java servlet. The tool component in Javascript was able to detect all user interactions with the browser (client side). Then, all the events detected were passed to a Java applet (client side). The applet allowed gathering all the interactions provided by the Javascript component; at the end of the testing procedure, the applet sent all data to a servlet that created relative logs file (server side).

Such log files contained a wide variety of user actions (such as mouse clicks, text typing, link selections…) as well as browsing behaviour, such as page loading start and end. In particular, the tool logged the time when a specific interaction was performed. Time was expressed in milliseconds and then converted into seconds. During the testing procedure, users had to click on the "Next task" link when they switched to the following task.

The logging tool recorded task switching in the log file, followed by the time (since the beginning of the test). When the analysis was performed, it was possible to calculate the time spent performing each task as the difference between the time recorded at the beginning and at the end of the task.

5.1.2 The Questionnaire

After the end of both tests, participants were asked to fill in a questionnaire composed of 18 questions. Such questionnaires allowed collecting information about the operating system and the assistive technologies used by the participants, and to obtain other qualitative data not obtainable by the logging tool. Subjective information was also considered. For example, users could express opinions and ideas about the usefulness of sounds, shortcuts, etc. In addition, a section was specifically designed for low vision users, in order to register opinions on colour contrast and link magnification. Indications about the level of difficulty of each task were also considered. Finally, suggestions and comments were elicited.

5.2 The Wizard Test

The Wizard just assigned a standardized pattern of tasks (necessary for the subsequent evaluation) without constraining participants' behaviour.

Participants were asked to carry out a set of seven tasks per test. The tasks included common navigation operations, such as page opening, content reading, and information search. The participants were also required to download files, fill in a form, and so forth. *Session0* and *session1* included the same types of tasks, which differed only in some minor aspects (e.g. the file to download, the information to find, etc.).

The following list of tasks were assigned to the users, both in *session0* and *session1*:

1. Visit the bulletin page of Tuscany (Home::Notizie::Toscana_oggi). This was a reading content page where the information was organised by using 'heading levels'. The user was asked to look for a specific piece of information.
2. Visit the Web page "Departments and Commissions" (Home::Organizzazione::Dipartimenti e commissioni). This page was a content reading page where 'heading levels' was used to logically group the different pieces of information. A page index was composed

of a small set of local links. The user was asked to find the name of the coordinator of the commission of a specific association.

3. Visit the alphabetically ordered list of “Sound Magazine” (Home::Documenti&Download::Rivista Suoni::Elenco alfabetico). This page was a content reading where both ‘heading levels’ and an index combobox were used to organize page information. The user had to search for a specific song title.
4. Downloading a document (Home:: Irifor::Corso di formazione "Homerus"). This page was a content reading page containing a specific link for downloading an application form. The user had to search and download a specific application form.
5. Download a zip file (Home::Documenti&Download::Download di manuali) This was a ‘downloading page’ where groups of files were placed in several tables: each table summary contains the name of a specific group. The users had to download a specific file.
6. Open an external URL(Home:: Organizzazione::Sezioni UIC Toscana) This was an ‘index page’ containing brief information about local chapters of the Tuscan Association for the Blind and related Web site links. The users had to search for the telephone number and open the Web site of a given chapter.
7. Fill in a form (Home::Invia un messaggio).
   This was a Web page with a form to fill in for sending comments, suggestions, or questions. The users had to fill in the form and send it.

While users carried out the *test*, they were free to navigate as they pleased. A “Next task” button was available to start the performance of the following task when the user decided that the goal had been reached or chose to give up. The indication of the next task to acomplish appeared in a popup menu, which contained a “Continue the test” button to proceed with the test. When users clicked on the “Stop” button, all interactions collected by the logging tool (see 5.1.1) were recorded in a log file.

6. RESULTS

All data gathered through the testing procedure were analysed in order to evaluate the overall improvement of the Web site after the application of our criteria. Such improvement was measured in terms of navigation time saved by users in accomplishing given tasks.

The difference between the time spent performing each task in *session0* and *session1* (performed respectively in “control site” and “revised site”) was used to verify if and to what extent the application of our criteria had improved navigability. So, the time saved by users was taken as an indicator of Web site improvement.

Considering size and type of data, non-parametric statistic tests were applied to raw data; α was fixed at 0.05 (significance) and 0.01 (high significance) [21].

We found a significant difference between the total time spent by all users performing each task in *session0* and *session1* (Wilcoxon matched pairs test; $n=7$; $z=2.37$; $p<0.05$). For each task, the total time was calculated by summing the time spent by each user (from user1 to user20). Figure 8 shows the time saved performing each task (1-7), averaged over 20 users.

We also found a highly significant difference between the total time spent by each user performing all the given tasks in *session0* and *session1* (Wilcoxon matched pairs test $n=20$; $z=3.93$; $p<0.01$). For each user, the total time required in *session0* and *session1*

was computed by summing the time required for each task, from task1 to task7. Figure 6 shows the time saved by each user (1-20) performing all the tasks, averaged over 7 tasks.
The wide range of time difference shown in Figure 8 and 9 (for tasks and users respectively) is possibly due to the different ability of users/difficulty of tasks. However, on average, the application of our criteria to the Web site has led to a significant/highly significant time saving for all users and tasks.
As can be noted in Figure 8, Task 1 and 6 were the less influenced by our criteria, likely because the main criterion involved in these two tasks was the application of heading levels, which are not so crucial for low vision users, unless the content of the page includes a very high number of links and/or is overloaded by confused information. In Task2 (i.e. looking for information in a long page) turned out to be the most influenced by our criteria, likely due the fact that low vision users could considerably reduce their navigation by using side submenus (e.g. local links or list boxes) to move quickly to a specific section of the page.

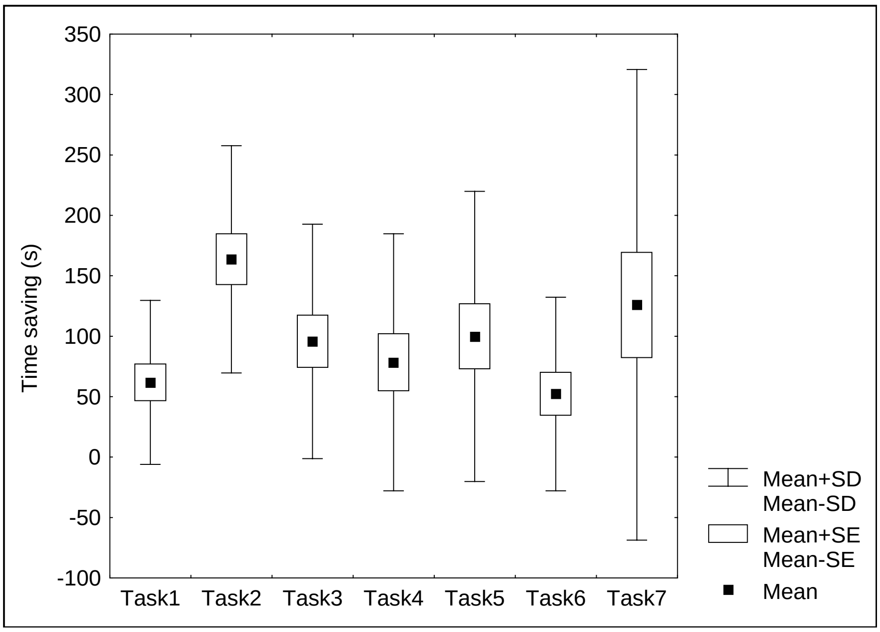


Figure 8 – Time (s) saving/task, expressed as the difference between the time spent performing each task in *session0* and *session1*, averaged over 20 users. Whiskers: mean ± SD, boxes: mean ± SE.

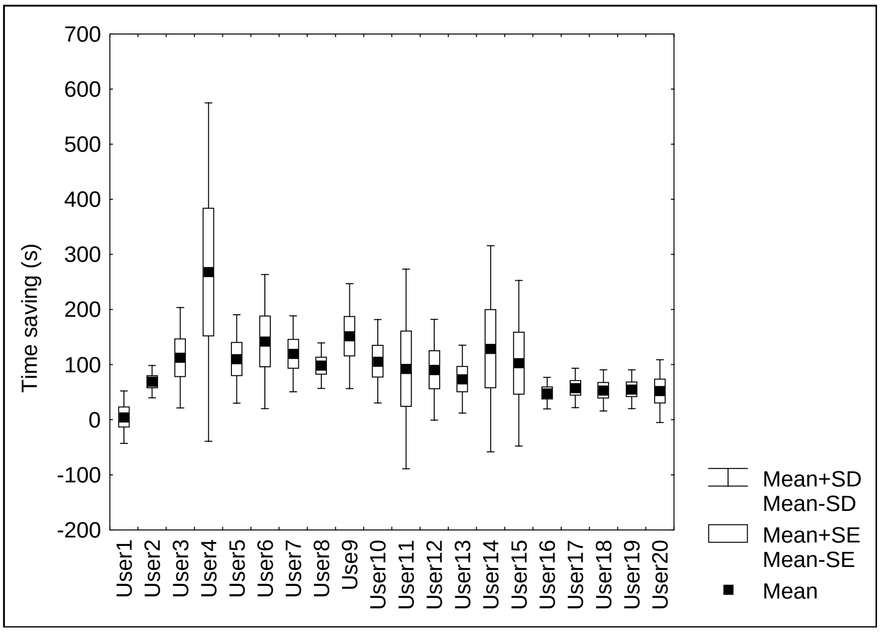


Figure 9 – Time (s) saving/user, expressed as the difference between the time spent performing all the tasks in *session0* and *session1*, averaged over 7 tasks. Whiskers: mean ± SD, boxes: mean ± SE.

Our results show that both blind and low vision users benefited from the application of our criteria saving around 40% of their navigation time. However, blind people saved more time than low vision users (50% vs. 30%). This result is consistent with our expectations because, even if usability criteria are thought both for blind and low vision users, they are mainly aimed at improving accessibility through screen readers rather than through magnifying programs.

Subjective information was gathered through a questionnaire filled by all users. In particular, users were asked to express their opinions about sounds, colours, magnifiers, and so on.

85% of the users considered the sound added to page loading to be helpful, however, 60% would also associate different sounds to distinct types of links (internal, external, local). All low vision users found that the association of different colours to distinct areas helped spotting important section in the page (e.g. area of application form, navigation bar, etc.). Colour contrast was considered unhelpful by 60% of low vision users, who suggested to use the combination white text/dark background, instead of the one used in the prototype (black text/white background). About 2/3 of all users used shortcuts and specific commands of the screen reader during the navigation.

Users were asked to write some remarks or differences observed between the two Web site versions. Almost all of the blind users judged particularly helpful (1) the application of headings to group information, (2) the usage of hidden labels for marking navigation bar and side submenu, and (3) the repartition of several files into various tables (one per category), obtained by using the “summary” attribute together with the name of each category. Low vision users particularly appreciated enlargement and colour change of background/text appearing when mouse cursor hovered over links.

Some blind users suggested the application of different sounds (1) to distinguish between failure and success (of a process); (2) to let the user know when they are writing in an edit box or when, in a field, there are no more characters to cancel using the backspace key. Low vision users suggested to better show the separation among information blocks, to facilitate the identification of each block.
Finally, the difficulty in finding blind users with a sufficient familiarity with a screen reader and available to perform an user testing like the one proposed here, should be considered when approaching similar experimental procedures.

7. CONCLUSIONS

In this paper, we first discuss a set of usability criteria created to improve Web navigation for vision impaired people, and then we present the empirical results of a user testing designed to evaluate the impact of such criteria on the Web interface.
The user testing was conducted by using remote evaluation techniques and two groups of people were involved: blind and visual impaired subjects. Two version of the same Web site were considered: a "control version" without criteria and a "revised version" created according to our criteria. A questionnaire was used to collect information on navigation quality from users. User testing and questionnaire results reveal that our usability criteria improved Web site navigability both in quantitative and qualitative terms, by reducing the navigation time of 40% and by making the Web site more friendly to blind/low vision users.
We are currently working on the development of a tool that should be able to automatically check whether the design criteria proposed are supported by the Web site considered in order to facilitate their application and support designers since the development phase.

8. ACKNOWLEDGMENTS

The authors wish to thank to Ivan Norscia for the help to improve the paper presentation.

REFERENCES

1. Abascal, J., Arrue, M., Garay, N., Tomás J. USERfit Tool. *A tool to facilitate Design for All*. In Proceedings of the 7th ERCIM Workshop "User Interfaces for All", Paris (Chantilly), 2002, in Springer-Verlag, Lecture Notes in Computer Science n.2615 (2003), pp. 141-152.
2. Abascal J., Arrue M., Fajardo I., Garay N., Tomás J., *Use of Guidelines to automatically verify Web accessibility*. Universal Access in the Information Society. Special Issue on "Guidelines, standards, methods and processes for software accessibility" (Guest editors: Gulliksen J., Harker S., Vanderheiden G.). Springer Verlag, 2003.
3. A-Prompt: Web Accessibility Verifier, Adaptive Technology Resource Center (University of Toronto) and Trace Center (University of Wisconsin), Canada & USA, accessible at http://www.snow.utoronto.ca
4. Barnicle, K. *Usability Testing with Screen Reading Technology in a Windows Environment*. Proceedings of the 2000 Conference on Universal Usability (CUU-00), pp. 102-109, ACM Press, November 16-17 2000.
5. Beirekdar, A., Vanderdonckt, J., Noirhomme-Fraiture, M., *A Framework and a Language for Usability Automatic Evaluation of Web Sites by Static Analysis of HTML Source Code*, Proc. of 4th Int. Conf. on Computer-Aided Design of User Interfaces CADUI'2002, Valenciennes, 2002, Kluwer Academics Pub., Dordrecht, 2002, pp. 337-348.
6. Brewster, S.A. (1998). Using earcons to improve the usability of a graphics package. In Proceedings of BCS HCI'98, (Sheffield, UK), Springer-Verlag, pp 287-302.
7. Christensen S. (2001) *How we work to make the Web speak*. Computers in libraries, vol. 21 (9), (October 2001), pp. 30-34.
8. Clarck D., Dardailler D. (1999) *Accessibility on the Web: Evaluation and repair tools to make it possible*. In proceedings of the CSUN Technology and Persons with disabilities Conferences, Los Angeles, CA. Available at http://www.cast.org/bobby.

9. Donker H., Klante P., Gorny P. The Design of Auditory User Interfaces for Blind Users. In: Proceedings of the Second NordiCHI, Tradition and Transcendence, Edited by O. W. Bertelsen, S. Boedker and K. Kuutti, October 19-23, 2002, Aarhus Denmark. New York 2002 : ACM Press pp. 149-155.
10. Goble C., Harper S., Stevens R., The Travails of Visually impaired Web Travellers, Proceedings ACM Hypertext 2000, ACM Press.
11. Ivory, M and Hearts, M (2001) The State of the Art in Automating Usability Evaluation of User Interfaces. ACM Computing Surveys, Vol, 33, No 4: 470-516.
12. Jaws for Windows (2004), Freedom Scientific at http://www.freedomscientific.com.
13. Leporini, B., *Criteria to improve Web site usability and accessibility when interacting through screen readers: definition, application, and evaluation*. PhD thesis, Dipartimento di Informatica, Università di Pisa, September 2003.
14. Leporini, B., Paternò, F., *Increasing Usability when Interacting through Screen Readers*, International Journal Universal Access in the Information Society (UAIS), Springer Verlag, Vol.3, N.1.
15. Multimodal visualisation for blind people (Multivis). Department of Computing Science, University of Glasgow. http://www.dcs.gla.ac.uk/~lorna/home.html
16. Nicolle, C., Abascal J. *Inclusive design guidelines for HCI*, p. 285, Taylor & Francis, 2001.
17. Nist Web Metrics: http://zing.ncsl.nist.gov/WebTools/ tech.HTML
18. Paganelli, L., Paternò, F., Tools for Remote Usability Evaluation of Web Applications through Browser Logs and Task Models, Behavior Research Methods, Instruments, and Computers, The Psychonomic Society Publications, 2003, 35 (3), pp.369-378, August 2003.
19. Scapin, D.L., Leulier, C., Vanderdonckt, J., Mariage, C., Bastien, Ch., Farenc, Ch., Palanque, Ph., and Bastide, R., *A Framework for Organizing Web Usability Guidelines*, in Ph. Kortum and E. Kudzinger (eds.), Proc. of 6th Conf. on Human Factors and the Web HFWeb'2000, Austin, 2000, accessible at http://www.tri.sbc.com/hfweb/scapin/Scapin.html .
20. Section 508 standards available at http://www.section508.gov/
21. Seagle S., (1956). *Non parametric statistics: for the behavioral science*. New York. Mc Graw - Hill Book Company, Inc.
22. Stephanidis, C., Akoumianakis, D., Sfyrakis, M., & Paramythis, A. (1998). *Universal accessibility in HCI: Process-oriented design guidelines and tool requirements.* In C. Stephanidis & A. Waern (Eds.), Proceedings of the 4th ERCIM Workshop on "User Interfaces for All", Stockholm, Sweden, 19-21 October .
23. USABLE NET (2000) LIFT ON LINE. Available at http://www.usablenet.com/
24. Vanderdonckt, J., Development Milestones towards a Tool for Working with Guidelines, Interacting with Computers, Vol. 12, N° 2, 1999, pp. 81- 118. Accessible at http://belchi.qant.ucl.ac.be/publi/1999/Milestones. pdf
25. WAI Accessibility Guidelines. Web Accessibility Initiative, World Wide Web Consortium, 1999. Accessible at http://www.w3.org/wai
26. Web Content Accessibility Guidelines 2.0, W3C Working Draft 1 March 2004available at http://www.w3.org/WAI/GL/WCAG20/