

Is Facebook Really "Open" to All?

Maria Claudia Buzzi IIT-CNR Pisa, Italy Claudia.Buzzi@iit.cnr.it	Marina Buzzi IIT-CNR Pisa, Italy Marina.Buzzi@iit.cnr.it	Barbara Leporini ISTI-CNR Pisa, Italy Barbara.Leporini@isti.cnr.it	Fahim Akhter Zayed University, I.T. Dubai, U.A.E fahim.akhter@zu.ac.ae
---	---	---	---

Abstract

Social interaction is important for everyone, and vital for the differently-abled and elderly. Social network applications are causing us to change our ways of communicating and working. We can now make friends all over the world, share ideas and knowledge, search for others' opinions, and more. Facebook and Twitter are examples of these applications. However, these opportunities are usually meant for people who can walk, see, and talk without difficulty. In this paper we show how blind persons, using screen readers and voice synthesizers to access the Web, can interact with social network applications. Our case study focuses on Facebook, analyzing basic features, explaining electronic barriers and suggesting solutions.

1. Introduction

Social networks such as Facebook and Twitter fuel social interaction and user collaboration. This is very important for anyone, and vital for the differently-abled such as the blind and the elderly, who have mobility problems. However, emerging technologies (in particular Rich Internet Applications [RIAs] that convey efficiently dynamic Web content) if not opportunely designed, may exclude many people from this opportunity.

User experience navigating the Web depends on many factors: finding information rapidly, navigating content efficiently, executing tasks effectively, user privacy and security of personal data, etc.

According to the WHO (World Health Organization) about 314 million people worldwide are visually impaired (87% living in developing countries), and of these, 45 million are blind [21].

Blind people usually access the Web through screen readers, an assistive technology that interprets and announces screen content via voice synthesizer. An alternative is to use a Braille tablet as output, but since this is extremely slow it is rarely used [9].

In today's Web, multimedia and dynamic content, as well as the increasing complexity of Web layouts, may present serious barriers for blind users. Problems range from wasting time to the need for additional cognitive effort (compared to sighted users), needing to ask for help, or even abandoning a task.

Blind persons could enjoy on-line social networks to communicate with friends and socialize with new people, but when accessing the Internet they must overcome a number of obstacles. Some of these are insurmountable, such as mouse-only focusable objects that require Web developer to modify the page source code to match accessibility, while others only make the reading awkward.

In this paper we discuss the main issues involved in interacting via screen reader with Facebook. Specifically, we analyze some basic features (log-in, friendship and some simple interactions with other users). Then we briefly discuss how WAI-ARIA [18], a suite developed by the Web Accessibility Initiative (WAI) group of W3C, can facilitate interaction for screen reader users.

This paper is organized into six sections. After this introduction, Section 2 presents issues of interaction via screen reader and voice synthesizer, Section 3 introduces related works and Section 4 describes the exploring Facebook via screen reader, highlighting potential problems. Section 5 introduces the WAI-ARIA suite, discussing how its application can improve interaction for blind users, for instance by defining the live regions and structuring content in logical sections (landmarks/regions). Last, we present conclusions and describe future work.

2. Interaction via screen reader

We focus on the needs of blind persons who access the computer and navigate the Web via screen reader and voice synthesizer.

A screen reader is an assistive technology that helps the blind to interact with a computer and other electronic devices (palm computers, mobile phones). The screen

reader mediates between the user and the operating system, assisting individuals by interpreting the user interface, which is read aloud sequentially (word by word and line by line) by means of a voice synthesizer. Technically, a screen reader is software that identifies and interprets what is being displayed on the screen (or, more accurately, sends it to a standard output device, whether a video monitor is present or not). This interpretation may be re-presented to the user with voice synthesizer, text-to-speech, aural icons, or a Braille output device (such as refreshable Braille display) [16]. Further details on how a screen reader works with a user interface are described in [1].

Several screen readers are available as free or licensed software including Windows Bridge (<http://www.synthavoice.on.ca/>), Windows Eyes (<http://www.gwmicro.com/>), Hal for Windows (<http://www.dolphincomputeraccess.com/>) and JAWS for Windows (by <http://www.freedomscientific.com/>), which is the most frequently used in Italy [10]. In this study we refer to the latter assistive technology.

Of physical disabilities, blindness involves numerous problems in Website navigation and exploration ([4], [7], [14]). The challenge to web designers is to create websites that are not only informative and visually attractive but also accessible and friendly to the visually impaired. Social networks such as Facebook, which offers an environment that integrates content from different sources (chat, wall, messages, multimedia, etc.), has a complex layout, very difficult for screen reader interaction.

When designing a Web interface for special needs persons it is necessary to consider the three main interacting subsystems of the "Human Processor Model": the perceptual, motor and cognitive systems [2]. Without understanding constraints and mental models of blind users, developers are not aware of the complexity of their situation, nor how to design to meet their needs [6]. Sightless persons perceive page content aurally and navigate via keyboard. This can make the "reading process" time-consuming, difficult and frustrating, if the content is not designed with special attention to their needs.

A screen reader reads the contents sequentially, as they appear in the HTML code. This process is annoying when part of the interface (such as the menu or navigation bar) is repeated on every page. As a consequence, blind users often stop the screen reading at the beginning, and prefer to navigate by Tab Keys, from link to link, or explore the content row by row, via arrow keys. Furthermore, the screen reader announces the most important interface elements such as links, images, and window objects as they appear in the code. For the blind user, these elements are important for figuring out the page structure, but require additional cognitive effort to interpret. The user risks developing a different mental model of both the

interaction and the learning processes, so it is crucial to provide an easy overview of the system and contents.

Web developers must be aware of the following problems that interaction via screen reader may produce:

1. Content serialization is time- (and resource-) consuming; thus the user adopts the aforementioned reading strategies.
2. Difficulty processing page content. Content and structure mixing makes reading difficult. If a table's content is organized by columns, the screen reader (which reads by rows) announces the page contents out of order; consequently, the information might be confusing or misleading for the user.
3. Lack of context. When navigating by screen reader the user can access only small portions of text and may lose the overall context of the page; thus it may be necessary to reiterate the reading process.
4. Lack of interface overview. Blind persons do not perceive the overall structure of the interface, so they can navigate for a long time without finding the most relevant contents.
5. Difficulty understanding User Interface (UI) elements. Links, content, and button labels should be context-independent and self-explanatory.
6. Difficulty working with form control elements (see 4.2). Fortunately, new JAWS versions (v. 10 and later) simplify the interaction with forms since it can automatically activate the editing modality (for text input) when the virtual focus arrives at the text box (for instance when the user presses the Tab key).
7. A blind person is unable to fully access multimedia content such as video streaming, video conferencing, and captioning. If an alternative description is not present, the user may lose important content.

These obstacles make it too difficult for blind people to fully understand and perceive the structure of a Web page. However, with appropriate design, blind users can easily perceive the table of contents or arrive at the main content quickly if the Web page is well-structured in logical sections [5], [11], [15].

3. Related Work

Several general accessibility and usability guidelines have been proposed in the literature for both application and Web interfaces. One of the more authoritative sources for the Web is the World Wide Web Consortium (W3C, <http://w3.org/>) which, along with the work of the WAI group (Web Accessibility Initiative), defines accessibility guidelines for web content, authoring tools, and user agent design. The W3C Web Content Accessibility Guidelines (WCAG) are general principles for making Web content more accessible and usable for people with disabilities [20]. However, these general criteria need to be specified within explicit and detailed guidelines that can be assessed simply and rapidly in order to be

concretely applied. The context of use as well as the desired goal must be carefully considered since they impact on the interface design (e.g., the aims of e-commerce are different from those of social networks).

The European Union Communication "Towards an Accessible Information Society" declares it is urgent to achieve a wider and more effective approach to e-accessibility, and in particular web accessibility, promoting the adoption of the Web Content Accessibility Guidelines 2.0 in Europe, with a set of key actions. The objective is to achieve 100% accessibility of member states' public websites by the end of 2010.

In the Middle East, several countries such as the United Arab Emirates (UAE) take into account Web accessibility to structure their official Web sites. To this end, the use of guidelines is required and would help determine how well the different applications offered to people with special needs are used. The Dubai Government (<http://www.dubai.ae>) is fully committed to guaranteeing that all government websites are equally accessible to all Internet users in term of content availability, regardless of any visual, auditory, cognitive, or motor impairment.

Many user studies suggest that the blind encounter more difficulty than those with other sensorial disabilities (such as low vision, motor or hearing impairments) when executing specific tasks [4], [6], [14]. Petrie et al. [14] presented the results of accessibility testing of 100 websites with users with visual, motor and perceptual disabilities, showing that websites that are accessible for differently-abled users can also be visually pleasing. Specifically, 100 websites spread out over five sectors were tested with automated verification and user testing, involving 51 differently-abled users including 10 totally blind users. Results showed a mean task success rate of 76% that fell to 53% if considering only the totally blind (the lowest score of all user categories). Likewise, regarding user satisfaction, the authors recorded that the blind encountered more difficulty than other differently-abled users (4.2 on a 1-7 Likert scale, the lowest score of all the user categories). Researchers at Manchester Metropolitan University [4] highlighted issues of non-visual access by studying a sample of blind and visually impaired users who performed four information-seeking tasks, including the use of search engines. Visually impaired users searching the Web for a specific piece of information took an average of 2.5 times longer than sighted users. The efficiency gap was further quantified by Ivory et al. [6]; when blind subjects executed a set of tasks, they took twice as long as sighted users to explore search results and three times as long to explore the corresponding web pages.

Hailpern et al. compare the way interaction via screen reader changes when moving from static content to Rich Internet Applications. They propose a user workflow design model for improving Web 2.0 accessibility and

suggest design requirements for Web developers to make interaction for the blind more satisfying [6].

Concerning Facebook accessibility for the blind, to the authors' knowledge very few studies have been carried out [8], [12], [17].

Kelly et al. highlight that there are many Facebook groups used by people with disabilities including blind students, and offer a general discussion of accessibility approaches related to the use of social networks [8].

Meiselwitz and Lazar observed that many teachers incorporate social networking into their lectures. In their study, the authors examined the registration process for 22 social networking Web sites and offer suggestions for improving accessibility [12]. Furthermore Wentz and Lazar studied the topic of e-mail and social networking, carrying out a study on e-mail use by blind users when using both web-based and user agent e-mail systems [17]. However, all of these studies only address a small part of interaction with Facebook (such as login, captcha), while a more systematic approach is still lacking. In this sense, our work is aimed at analyzing overall user interaction with Facebook via screen reader.

4. Interacting with Facebook

4.1. Evaluation Methodology

For testing some Facebook features, we used the screen reader JAWS for Windows (<http://www.freedomscientific.com>) v. 10 and 11 and the Mozilla Firefox (version 3.5 and 3.0.15) and IE (v. 7) browser. The Facebook English version was selected as the user interface to be tested. The test was performed in December, 2009.

The test was carried out by the authors independently, and outcomes were merged. One author has been totally blind since childhood and is skilled at using JAWS (i.e., she uses advanced commands). However, after analyzing the test results we noticed that in spite of her great expertise with JAWS, she was unable to perceive the exact structure of the graphical page layout. The sighted authors carried out the same test using only JAWS basic commands. Covering a variety of screen reader interaction modalities (i.e. basic commands, simulating the level of novice users, and advanced functions for experienced users) led to a more accurate analysis.

4.2. Interacting with the Facebook user interface

In order to evaluate how a blind user may interact with Facebook, and detect accessibility and usability issues encountered when using a screen reader, we analyzed its basic functions. Additional advanced features will be better tested in a future study.

For our purposes, we will consider that Angela, Betty and Clara are three friends who wish to communicate via

Facebook. Betty is skilled at using the screen reader JAWS.

As discussed in section 2, interaction via screen reader is complex, especially when surfing Web pages. One important interaction difficulty is related to how the screen reader has to handle a key pressure, i.e. as a special command or as the text input. To discriminate between command case and text entry, JAWS provides two interaction modalities: exploration and editing. Furthermore a ‘virtual focus’ is provided for the exploration that may not coincide with the system focus. If using an old version of JAWS (v. 9 or previous one) the user might need to switch between editing and navigation modalities several times. Fortunately JAWS 10 automatically enables the editing mode when the virtual focus is in a text box (Auto Forms Mode). This simplifies everything and makes interaction with form fields quicker.

Opening the Facebook source code we may observe that the page implementation relies on Javascript and AJAX programming. AJAX (asynchronous JavaScript and XML) integrate several technologies including: (X)HTML, Cascading Style Sheets, JavaScript, Document Object Model, XML, XSLT, and the XMLHttpRequest object [13]. AJAX enables dynamic applications to make quick updates of UI portions without reloading the entire page, making the application faster and more responsive to user actions. However, if not appropriately coded these applications might offer poor accessibility via screen reader. Bigham et al. introduce a common scripting framework to enable users and developers to collaborate in order to improve accessibility. Using the same underlying technology allows people to share scripts, save resources and thus favor accessible Web content [2].

4.3. Home page

Let us first consider the Facebook Home page where it is possible to log in. Figure 1 reports a portion of this page interpreted by JAWS. Clearly, the content is basically recognizable, so the page can be considered accessible; but does it offer any mechanism for simplifying interaction? Can a skilled user exploit some advanced command, such as pressing the command “t” for navigating and skipping among the tables, to move more quickly and easily understand their content?

Although the page appears accessible, by exploring it in detail via Up/Down arrows, or via Tab key, the UI does not turn out to be quite so usable.

The main issues observed by Angela, Betty and Clara, when navigating the Facebook Home page as beginner and experienced users, can be summarized as follows:

- No logical partitioning mechanism is available, such as headings, hidden labels, landmark/regions, etc.
- Two tables are detected by JAWS, but no summary is recognized: (1) the “log-in table” and (2) “Signing-up

table”, which is used for rendering the data field set to register with the system

- The log-in form is not very accessible. In fact, by navigating both via arrow keys or via Tab key (i.e. in exploring modality), labels are not clearly identified by JAWS, especially the one related to the password. Also, by moving via the Tab key in editing mode, when the focus is on the two fields, JAWS announces no label but just the edit box
- Some links, such as the last two links among those for the available languages, are not read by JAWS.

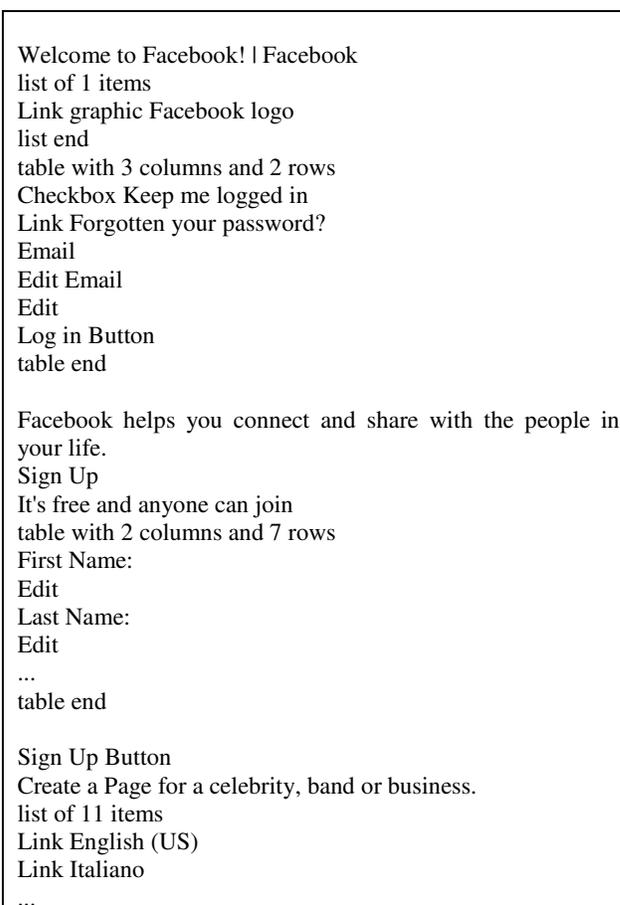


Fig. 1. Facebook homepage: portion of text announced by JAWS

No heading is available to skip quickly from one section to another nor to get a rapid overview of the main section of the page (e.g., log-in, sign up, select language, etc.). A blind user has to read the Web interface in a sequential way to access the content and must imagine a sort of map of the structure. When an expert blind user becomes familiar with a Web site/page, he/she can skip quickly to a specific part even if no mechanism is applied for simplifying the interaction. For example, to search an edit box, the advanced JAWS command “e” can be used or a precise string can be searched. However, to do this, the user has to know the page well. This activity is not

easy for beginners, so interaction is time consuming and often quite frustrating.

Now, let us assume we are already registered, and we have inserted the login data to enter our personal page. Here, the main usability issue is related to the absence of a mechanism (different from visual ones or email notification, if properly configured) for announcing new events, posts, comments, invitation requests, and so on; in other words, it is difficult to perceive whether something has been changed or if new comments have been posted. Navigating around the page content requires a lot of time and the user has to proceed in a sequential way. News feed readers are used to notify of new events, but at present, they are not identifiable by JAWS.

4.4. Signing up

Let us consider now the case of signing up for the platform. If Betty wants to sign up for the system, she opens the first page of Facebook (<http://www.facebook.com>) and by Tab key she reaches the fields to be filled in. Betty does not perceive whether all fields are mandatory or not. If a mistake is introduced – e.g. a mandatory field is not filled in or an invalid e-mail address is written – a short message is shown at the end of the form after the “Sign up” button (Fig. 2).

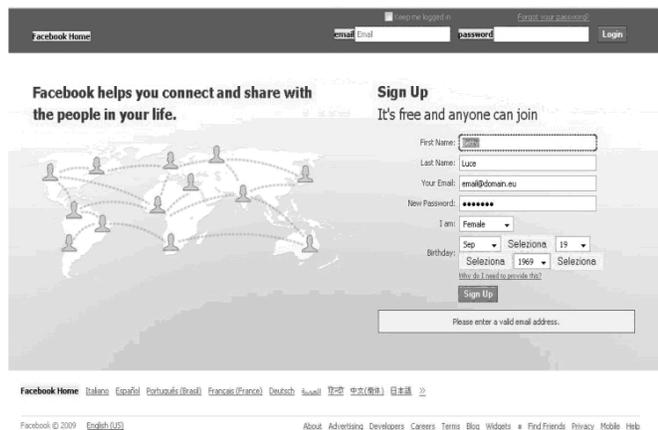


Fig. 2: Facebook signing up phase; the form is not completely filled in

Indeed, when pressing the “Sign up” button, the focus goes onto the first field labeled “First name” and the user is not able to understand why the given action is not carried out. He/she supposes that probably one field is not filled in correctly, but to understand what has happened it is necessary to read the page in a sequential way in the “Exploring mode”, by switching from the Editing to the Exploring mode. This action might be too difficult for beginner users, or tedious for those with more experience. A suitable way to give such a message should be considered in order to make user interaction more

friendly. Unfortunately, no sound, message or other alert is adequately provided to get the user’s attention or to announce what has happened. How to inform a blind user when dynamic messages are visualized is a usability issue previously considered in other studies [11]. To enhance this important aspect the WAI-ARIA suite can help by providing useful information in an automatic way [18].

After filling in all user data and pressing the “Sign up” button, there is the “Security check”: a captcha composed of two words that are used to confirm the subscription. Although an alternative audio version to make this step accessible is provided, it is rather difficult to understand the two words that are read aloud due to the poor quality of the audio file or to the background noise. In addition, the captcha is read in English even when the UI is set for another language. Thus this step is practically inaccessible for blind users.

Therefore, the security checker should be redesigned in order to make the “Signing up” procedure more accessible to any user. In order to proceed, Betty exploits Webvisum (<http://www.webvisum.com/>), which is a unique browser add-on that greatly enhances web accessibility and empowers the blind and visually impaired community. Among other things, Webvisum offers automated and instant captcha image solving, so that signing up to web sites (or making forum posts or blog comments) can be autonomously carried out without asking for help. Although such a tool is available, websites should offer accessible solutions for captcha, because Webvisum is not always able to correctly interpret the graphical captcha, and not all users know it.

4.5. Accepting a friend invitation request

Let us suppose that Angela sent a friend invitation to Betty. When Betty is logged in, on her personal Homepage a friend invitation request is available by means of the link “1 Friend request” to indicate that a request is pending. The main usability issue is that this event (i.e. a pending request) is not automatically announced by the screen reader. Moreover, no alert is provided that something new is available. This issue is a general usability problem encountered when opening the Facebook Home page. Thus, Betty needs to try exploring the page. She gets the heading list and perceives that an invitation request is waiting, because a specific heading level has been used to mark it. Once that part is reached, by selecting that heading Betty explores the page via the Down arrow, finds the link “1 Friend request” and clicks on it to open the related page (Fig. 3).

In order to find the right point in the page, Betty needs to explore via arrow key or try to search a specific string (e.g. “you have invitation”). The latter method is usually employed by skilled users or when a person knows the page very well and so can search for a specific string.

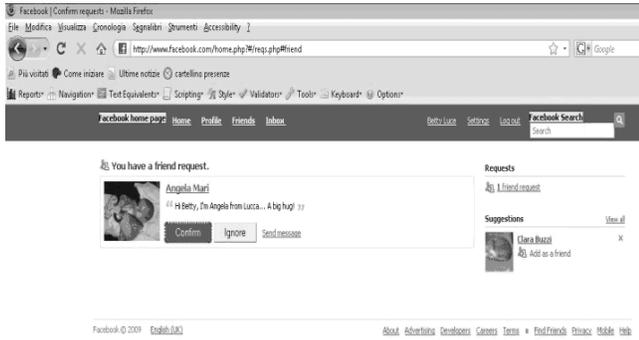


Figure 3: Facebook - confirm friend invitation

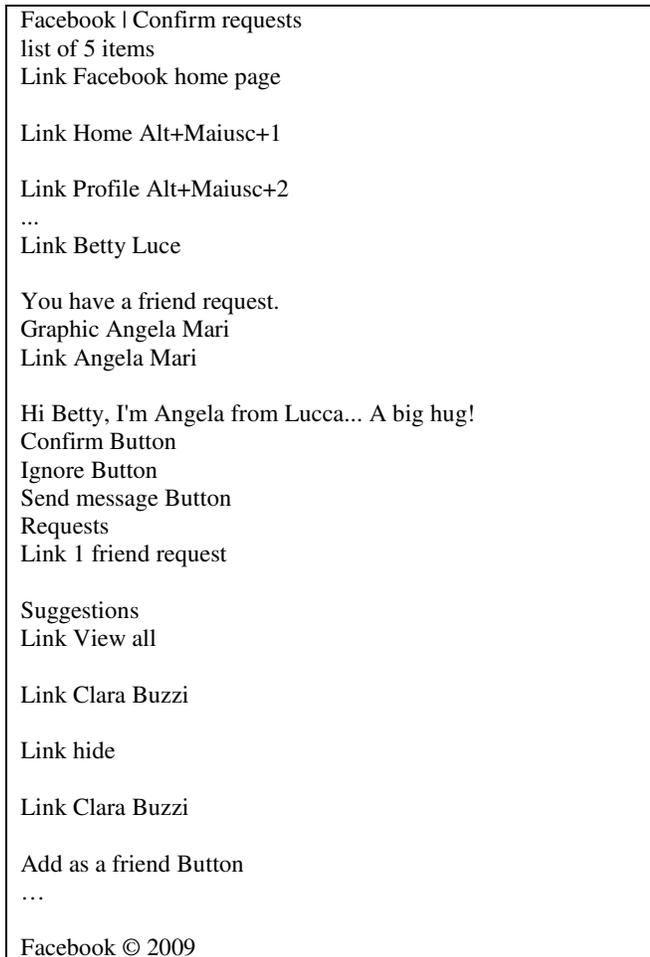


Fig. 4. Portion of Confirm invitation read by JAWS

Figure 4 reports a portion of content read by JAWS. By looking at this portion, it is possible to observe that some links are separated by blank lines (Fig. 4). When links are not consecutive – i.e. there are blank lines between them – the sequential reading could be tedious and frustrating for the blind user, who has to listen many empty lines. In this sense, no blank line should be left between links (Fig. 4)

4.6. Inviting friends

Let us suppose now that Betty wishes to invite some friends to join her contacts. She can look for a friend by using the Facebook search function, or on the other hand, she can decide to accept the suggestion given by the system to invite Clara (a friend of Angela). Indeed, by exploring the Facebook Home page, Betty finds the suggestion and clicks on the “Add as a friend” button in order to proceed (Fig. 3). When clicking on that button, unfortunately the page is not loaded again, but a new portion is updated by the Facebook system. This implies that Betty is not able to understand right away that the page has changed. She decides to read the page sequentially via arrow keys, but she soon realizes that there is too much information in the page. Thus, Betty tries to use the command to obtain the headings available in the page. Fortunately, headings are used to mark each message, including that regarding the friend invitation (Fig. 5).

After selecting that heading through the Down-Arrow and having confirmed via Enter key, Betty explores the page in order to find a suitable link to proceed. She clicks on “Add a personal message” and an edit box appears. After writing a short message, the “Send invite” button is activated (Fig. 6). The next step is the “Security checker” (Fig. 7).

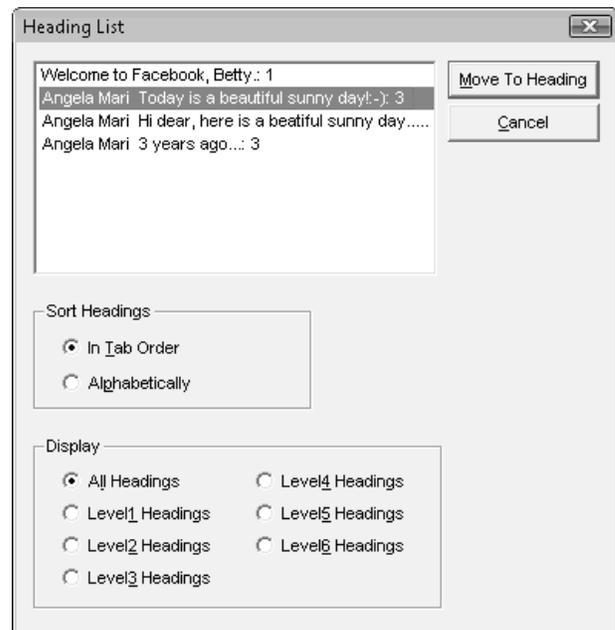


Fig. 5. List of headings available on the Facebook Homepage

The problem is the same previously described in the “Signing up” section (4.3.1): listening to the alternative audio version is rarely useful, especially for non-English-

speaking people. So, at this time, also the “Add as a friend” feature cannot be effectively enjoyed by a blind user.

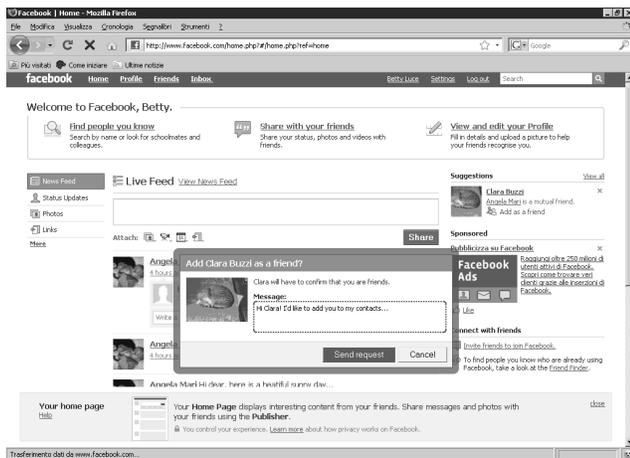


Fig. 6. Facebook Home page with Friend invitation message

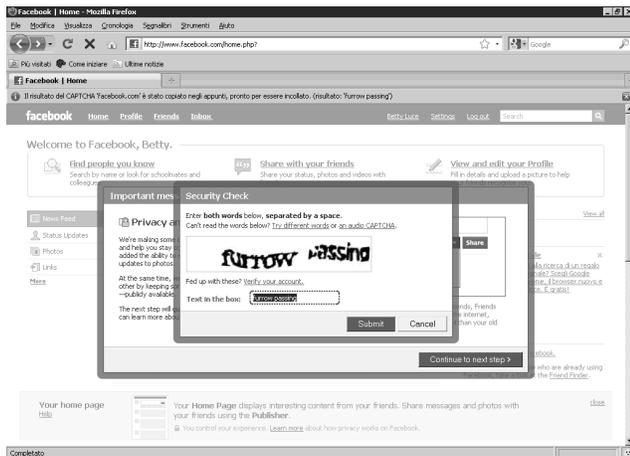


Fig. 7. Facebook Home page with Friend invitation message and security check request

4.7. Answering a message

Let us suppose that Betty wants to answer a message by Angela. When exploring the page, Betty encounters some difficulties identifying the button to use for replying to Angela. By showing the list of headings available on the page (i.e. by using the Insert+F6 JAWS command), Betty selects the first message with the arrow key and presses the Enter key. Now, Betty proceeds to explore with the virtual cursor and the Up/Down arrows in a sequential way in order to find the control for answering. The screen reader detects the following segment:

Link Do you wish a tea?
 22 minutes ago
 .
 Comment
 .
 Link See Wall-to-Wall
 Write a comment...
 Link Angela Mari
 Link Hide

Fig. 8. Portion of Facebook message read by JAWS

No button is detected by JAWS as a potential “answering” control element. Only links are available to click, but the texts used are not clear or meaningful or intuitive. The labels “comment” or “Write a comment...” are available, but they are neither links nor buttons. After some attempts to click the links available for the message to answer, Betty (a skilled user) tries to press the space bar – i.e. the alternative modality for left mouse click via keyboard when interacting via JAWS – in order to see to see if anything happens. At this point the message portion has been changed: an edit box as well as “Comment” button are added. To become aware of this dynamic update, the user must explore the portion in a sequential way; otherwise he/she is not able to perceive this change. When the focus is on the Edit box, Betty can activate the field (i.e. the Editing modality) and write her comment. At the end, by moving via Tab key and pressing the space bar (or Enter key) on the “Comment” button, the answering message is added to the page (Fig. 9).

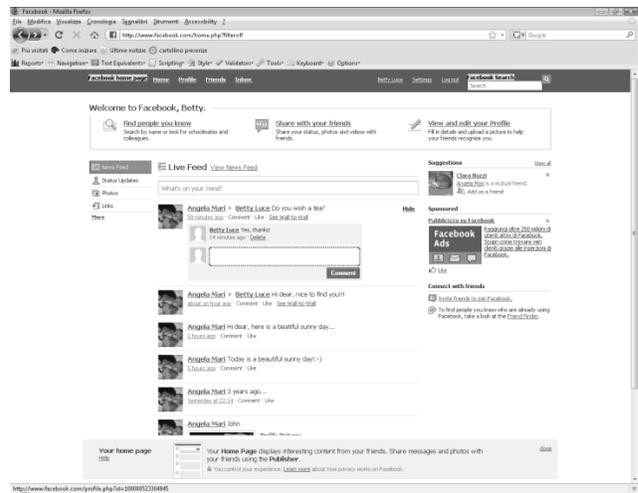


Fig. 9. Screenshot of Facebook with Betty's answer to Angela's message

4.8. Posting a message

Now Betty wants to post a new message. When Betty is on her personal Facebook Home page, she has to find the edit box to use for writing a new comment. As a skilled user, Betty presses the letter “E” (i.e., an advanced JAWS command) to move towards the edit boxes available in the page. The first edit box encountered by pressing “E” has no label and the corresponding button has no label either. Betty tries to press the “E” key again and she moves onto the edit box containing the string “What’s on your mind?”. By activating the Edit mode, Betty can write her comment. Unfortunately, the way JAWS works with this kind of function depends on the browser. With IE the Edit box is clearly visible and simple to use. With Mozilla Firefox, the edit box is not immediately available and when the user clicks on the written “What’s on your mind?” to use it, the user is unable to write inside it: all typed letters are interpreted erroneously, and no text is written in the edit box. Thus, this function is not accessible with Mozilla Firefox.

5. Discussion

Generally speaking, the Facebook environment is not easy to navigate for a blind user who interacts via screen reader. The greatest accessibility and usability issue is related to the announcement of new events (i.e. notifications provided to the user). Although “News feed live readers” are used to inform the user about new events/messages, to get this information via screen reader is in practice very difficult or impossible. This means that when a user connects to his/her own Facebook Home page, new comments or messages are not immediately perceived, unless the user explores all messages in a sequential way each time, by using the Exploring JAWS mode. This can require great effort and is stressful for the user; therefore one of the most important features of Facebook is practically inaccessible or difficult to perceive via screen reader. In this sense, the main goal of this kind of social network risks being unfulfilled: i.e. the environment does not provide a comfortable and easy way to stay in contact with other people. Therefore, a more usable user interface would offer new opportunities for blind people to fully participate in social networks.

In order to make the user interface more usable via assistive technologies, specific criteria and techniques should be applied to simplify the communication between the screen reader and the UI. WAI-ARIA (ARIA) allows delivery of accessible RIAs to screen reader users [18]. Specifically, parts of a Web page that change dynamically, such as Facebook notifications, may become accessible by using ARIA live regions. Defining roles, states and properties for UI elements, ARIA adds semantics to (X)HTML objects. In this way, changes in

the user interface can be automatically captured by the screen reader and communicated to the user. In general, Web developers may set appropriate Live Region Properties according their application design. By means of live region properties (set by developer), the screen reader knows how to process dynamic updates, for instance if and when to communicate the change (live property: “off”, “polite” or “assertive”). The ARIA best practices describe step-by-step how to define live regions [19].

There are several other additional accessibility and usability issues. Most of them are related to the form elements, such as when the focus moves inappropriately from a field to another one; several labels are not identified or some edit boxes are not detected by JAWS or are very difficult to use.

Another important accessibility issue is related to the “security checker”: the used captcha is not appropriately made accessible even though an audio version is provided to the users. In fact, those audio versions are unclear.

A general guideline for simplifying interaction for the blind is to structure the UI in logical sections, which may be marked as ARIA regions, specifying standard XHTML landmarks (main, navigation, search, etc.) or defining customized regions. No logical partitioning criteria is applied, except for one’s own homepage where each post is marked with the headings (Fig. 5). This certainly makes the perception of available posts clearer (although ARIA regions should be preferred to headings [19]), whereas new comments cannot easily be detected.

Structuring the Facebook UI, the user can obtain a page overview (i.e. the list of UI regions (“INS+CTRL+;” JAWS command), to move around a specific region (by arrow key) and also to jump from one region to the next/previous (“;”/“Shift+;” JAWS command).

Other general recommendations include making all content accessible by keyboard, and delivering the same content in a different way (visual and auditory channels).

6. Conclusion

Social networks are spreading rapidly throughout our society. Thus, by penetrating peoples’ lives, social networks track new and existing social connections. For blind individuals who have mobility problems, community networks can reinforce connections and encourage new social relationships. This opportunity is very stimulating, so new studies are needed to guarantee a fully satisfying access for them as well.

In this paper we illustrated the interaction via screen reader with basic Facebook features: sign up, log-in, friendship and some simple interactions with other users, showing several accessibility and usability issues that confirm that the Facebook environment is difficult to use for a blind user.

Aiming to overcome certain limitations for blind people, we briefly introduce how the WAI-ARIA suite could enhance Facebook accessibility via screen reader and greatly improve user experience for the blind. By adding semantics to UI elements, and managing the focus of any customized widget by using ARIA, the Web developer makes screen reader users able to fully control keyboard navigation,

To improve UI usability, it is recommended to organize the page in logical sections marked with ARIA regions, to obtain an overview of the page and allow the blind user to move rapidly from one region to another. Especially customized regions could offer many advantages. The blind user has more control over the interface, can navigate the page efficiently and select which text to read, and thus has a better interaction experience. Furthermore, ARIA live regions enable the communication of dynamic interface changes to the user, making new events/messages more detectable and accessible. This feature favors the full integration of blind users in social networks, enabling one to have real-time control of social activities with the assurance of not missing any content.

In future work we hope to extend this study by implementing a more systematic analysis, including Facebook advanced features and suggesting specific guidelines for developers. A comparison of the degree of accessibility with other social networks such as Twitter also deserves to be investigated.

7. References

- [1] C. Asakawa, and B. Leporini, "Screen readers", Chapter of the book *Universal Access Handbook*, Stephanidis C. (Ed), Taylor & Francis, June 11, 2009.
- [2] J.P. Bigham, and R.E. Ladner, "Accessmonkey: a collaborative scripting framework for web users and developers", *Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, ACM, pp. 25-34.
- [3] S.K. Card, T.P. Moran, and A. Newell, *The Psychology of Human-computer Interaction*, Lawrence Erlbaum Associates, London, UK, 1983, pp.29-97.
- [4] J. Craven, and P. Brophy, "Non-visual access to the digital library: the use of digital library interfaces by blind and visually impaired people." Technical report, Manchester: Centre for Research in Library and Information Management (CERLIM), 2003, <http://www.cerlim.ac.uk/pubs/index.php>
- [5] K. Fukuda, H. Takagi, J. Maeda, and C. Asakawa, "An assist method for realizing a Web page structure for blind people". *Proceedings of UAHCI 2003*, pp. 960-964.
- [6] J. Hailpern, L. Guarino Reid, R. Boardman, and S. Annam, "WEB 2.0: Blind to an Accessible New World", *Proceedings of the 18th International conference on World Wide Web*, 2009, ACM, pp. 821-830.
- [7] M. Y. Ivory, S. Yu, and K. Gronemyer, "Search result exploration: a preliminary study of blind and sighted users' decision making and performance". *Extended abstracts of CHI 2004*, pp. 453-1456.
- [8] B. Kelly, L. Nevile, E.A. Draffan, and S. Fanou. "One World, One Web ... But Great Diversity". *Proceedings W4A Conference 2008*, pp. 141-147.
- [9] S. Lee. "Universal access to mobile computing by modular i/o design", *Proceedings of Technology and Persons with Disabilities Conference 2004*, Los Angeles, USA, <http://www.csun.edu/cod/conf/2004/proceedings/122.htm>
- [10] B. Leporini, P. Andronico, M. Buzzi, and C. Castillo, "Evaluating a modified Google user interface via screen reader". *Universal Access in the Information Society*, issue 7/1-2, Springer, 2008.
- [11] Leporini B., and Paternò F., "Applying web usability criteria for vision-impaired users: does it really improve task performance?" *Proceedings IJHCI 2008*, Vol. 24, issue 1, January 2008, pp. 17-47.
- [12] G. Meiselwitz, and J. Lazar, "Accessibility of Registration Mechanisms in Social Networking Sites", *LNCS Vol 5621/2009 Book Online Communities and Social Computing* 10.1007/978-3-642-02774-1_9, pp. 82-90.
- [13] Mozilla Developer Center. AJAX. <http://developer.mozilla.org/en/docs/AJAX>
- [14] H. Petrie, F. Hamilton, and N. King, "Tension, what tension?: Website accessibility and visual design", *Proc. 2004 International Cross-disciplinary Workshop on Web Accessibility (W4A)*, pp. 13 – 18.
- [15] C. Senette, M. C. Buzzi, M. Buzzi, and B. Leporini, "Enhancing Wikipedia Editing with WAI-ARIA" *USAB 2009 - Usability & HCI Learning from the Extreme. LNCS Vol. 5889/2009*, pp. 159-177.
- [16] J. Thatcher, "Assistive Technology: Screen Readers and Browsers". Chapter 5 in *Web Accessibility*, CA, USA: Apress, 2006, pp. 103-124.
- [17] B. Wentz, and J. Lazar. "Email Accessibility and Social Networking". *LNCS Volume 5621/2009. Online Communities and Social Computing*, DOI:10.1007/978-3-642-02774-1_15, pp. 134-140.
- [18] W3C. Accessible Rich Internet Applications (WAI-ARIA) 1.0, Working Draft W3C Working Draft 15 December 2009, <http://www.w3.org/TR/wai-aria/>
- [19] W3C. WAI-ARIA Best Practices. W3C Working Draft 24 February 2009; Available from: <http://www.w3.org/TR/wai-aria-practices/>

[20] W3C. Web Content Accessibility Guidelines 2.0. <http://www.w3.org/TR/WCAG20/>, 5 Dec 2008.

[21] WHO. Blindness. Fact sheet: visual impairment and blindness, <http://www.who.int/topics/blindness/en/>