Mintzatek, Text-to-Speech Conversion Tool Adapted to Users with Motor Impairments

J. Eduardo Pérez, Myriam Arrue and Julio Abascal
EGOKITUZ: Laboratory of HCI for Special Needs, University of the Basque Country (UPV/EHU)
Informatika Fakultatea, 20018 Donostia, Spain

Keywords: Adapted User Interface, Assistive Technology, Limited Mobility, Speech Disorder, Text-to-Speech Conversion, User-Centered Design.

Abstract: Text-to-speech (TTS) conversion software tools are capable of generating synthetic voice from written text. These tools are essential for some groups of impaired users who have speech difficulties. In some cases, this limitation is caused by some kind of motor impairment. However, current TTS tools are not fully accessible as contain barriers for those users with limited mobility in upper extremities. This paper presents the most significant accessibility barriers detected for this specific user group. In addition, an accessible TTS tool, Mintzatek, has been implemented based on User-Centered Design (UCD) process. The user interface of the developed tool is adapted to users with limited mobility in upper extremities. All the development process has been guided by two real motor impaired users with plenty of experience in the use of assistive technologies.

1 INTRODUCTION

In the last few years, plenty of text-to-speech conversion software tools have been developed. They are capable of generating synthetic voice from written text and are meant to help a wide range of impaired users in their everyday lives.

Most of text-to-speech conversion tools are oriented to users who need speech assistive technology due to any functional diversity limiting their oral communication capacity. In some cases, this limitation is caused by some kind of motor impairment.

However, the development process of the user interfaces of such tools does not always consider universal access paradigm. Consequently, they may contain numerous accessibility barriers for those users with specific functional diversity for example for users with limited mobility in upper extremities.

In general, the developed user interfaces require use of mouse events for activating the different functionalities. This interaction mode implies mobility, precision and strength in upper extremities, something not possible for diverse groups of users.

The objective of this paper is to analyse the accessibility barriers in current text-to-speech conversion tools and to develop an accessible and usable tool for users with motor impairments. User-Centered Design (UCD) (Newell et al., 2000) process has been applied. It is a user interface design approach that takes into consideration final users' characteristics during project development. The products obtained are optimized for those users. This can be only achieved with the participation of final real users at different phases of the iterative design process of a product. Following this approach, we recruited end-users and they were involved during all the process starting at early phases of the analysis and design.

Several accessibility barriers were observed and directly discussed with users. New features were arised from interviews with users and several testing sessions were planned during the development process. As a result, a text-to-speech conversion tool has been developed which considers universal access principles, includes functionalities for improving users' performance and enables motor impaired users to easily interact with the user interface.

The paper has been structured in the following way: several text-to-speech conversion tools are presented in section two. In section three, an accessibility evaluation of selected three tools is presented. The accessibility guidelines and user

Eduardo Pérez J., Arrue M. and Abascal J..
Mintzatek, Text-to-Speech Conversion Tool Adapted to Users with Motor Impairments.
DOI: 10.5220/0004897701120119
In *Proceedings of the 16th International Conference on Enterprise Information Systems* (ICEIS-2014), pages 112-119
ISBN: 978-989-758-029-1

interface adaptation techniques considered in the development of the text-to-speech conversion tool accessible for users with motor impairments are discussed in section four. Section five is devoted to the development process based on UCD. Finally, conclusions and future work are drawn.

2 RELATED WORK

Numerous text-to-speech (TTS) conversion tools have been developed in the last few years. Some of them are devoted to mobile devices while others to personal computers. We have analysed three tools selected based on their popularity in user communities and the features they include in their user interface. All of them are standalone desktop software applications. This type of application provides valuable aid to those users experiencing difficulties for oral communication. The only requirement is to install the software in any computer (for instance, their personal laptop) so there is no need to purchase any speech-generating dedicated device.

An interesting example of such TTS tools is the BJ Hermes commercial software (BJ Adaptaciones, 2011). It was created by a Spanish company specialised in the development of assistive technology tools for people with disabilities. However, it presents serious accessibility problems for users with specific motor impairments. Its interaction model is based on a user who precisely uses the mouse device and keyboard access to the functionalities may be quite difficult for some users.

Another remarkable TTS system is the software called “Verbose Text to Speech” (NCH Software, 2013). Despite the fact that this shareware tool implements some screen reader features (not of interest for our research), it presents other useful ones like the lateral command bar with stretchtext characteristics. However, the tool presents some functionality only reachable by mouse device so keyboard-only users cannot access to them.

TTSReader tool has also been considered (SpheNet, 2011) during this research work. It includes some interesting features such as the tracking of the speech within the typed text. Activating this feature ensures that the cursor position is synchronized with the speech process highlighting the word the tool is reading aloud. Nevertheless, there is not any personalization option in the tool and some features require users to use a mouse device.

3 ACCESSIBILITY BARRIERS ON ANALYSED TTS TOOLS

This section presents an accessibility evaluation of the three different TTS tools described in the previous section (Figure 1 shows their main windows): “BJ Hermes PC” (BJHt), “Verbose Text to Speech” (VTSt) and “TTSReader” (REAt). This evaluation is focused on accessibility issues concerning physically impaired users that suffer from limited movements in upper extremities (causing imprecise use of mouse or even keyboard-only access).

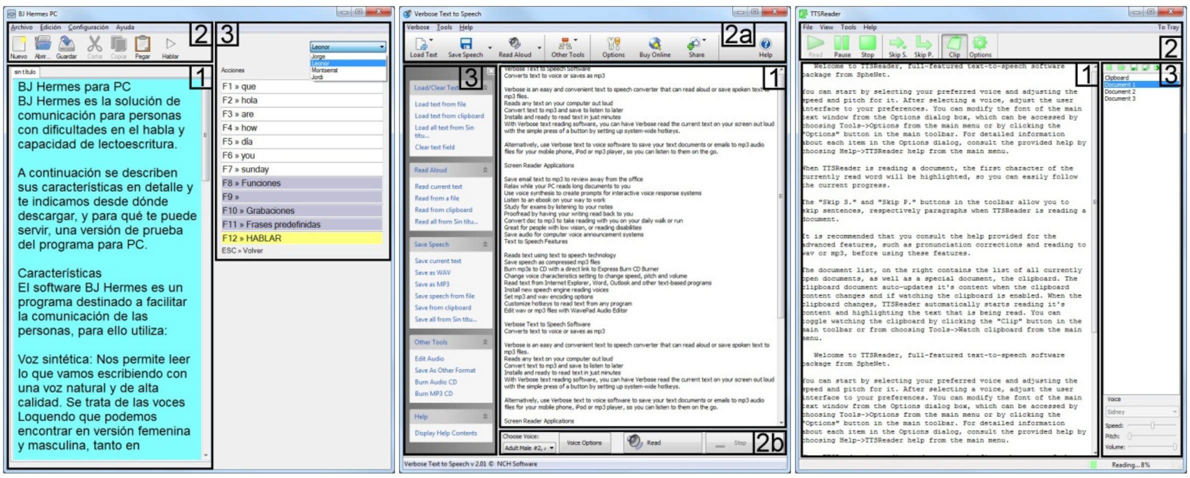

Figure 1: Main windows of three different TTS tools analysed during this work (left: commercial “BJ Hermes PC”, middle: shareware “Verbose Text to Speech” and right: freeware “TTSReader”).

Several accessibility principles and guidelines have been considered in order to detect accessibility barriers in the analysed tools: (Kurniawan et al., 2006), (NIA and NLM, 2002) and (Gajos et al., 2010). It can be noticed that some accessibility guidelines considered in this analysis are related to elderly people who may suffer ageing-related motor impairments as well as users temporarily disabled or interacting in unusual postures.

3.1 Buttons Issues

3.1.1 Reduced Dimensions

The analysed tools contain buttons with inappropriate size. They are too small, producing many difficulties when the mouse is used with a low precision. The VTSt application also implements buttons like links on its lateral command bar (see Area 3 in Figure 1 - middle), reducing even more dimension of clickable areas.

Another problem present in the three tools is that horizontal and vertical sequences of buttons are too close to each other (see Areas 2 and 3 in every capture of Figure 1 - left, middle and right). This issue causes motor impaired users click the incorrect option very often when using the mouse device.

3.1.2 Lack of Textual Information

The REAt tool implements five buttons related with file options that do not incorporate any text (see Figure 2). This issue can prevent users from knowing the function of each button.

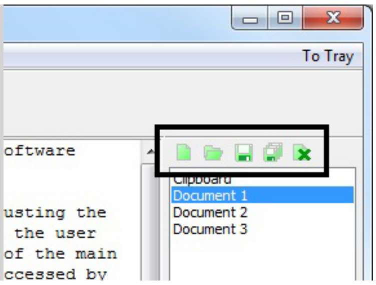

Figure 2: File options related buttons with no label tags on "TTSReader" application.

3.2 Other Components Issues

3.2.1 Cascading Menus

Every studied TTS tool contains hierarchical cascading menus on the upper area of its main window (see Areas 1 in every capture of Figure 3 – up, middle and down). These walking menus introduce accessibility barriers when the user cannot easily move the mouse pointer onto the desired entry (being especially difficult the navigation through secondary menus). In addition, it makes difficult to select an option by keyboard-only users as they have to go through a lengthy sequence of moving keys. Although every entry is reachable with a keyboard, it is recommended to minimise the use of this type of menu bars.

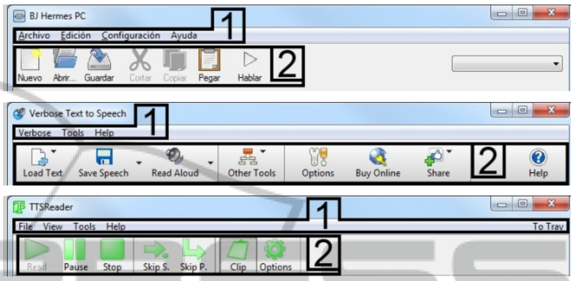

Figure 3: Same distribution of cascading menus above [1] and toolbar below [2], for the three TTS tools analysed (up: “BJ Hermes PC”, middle: “Verbose Text to Speech” and down: “TTSReader”).

3.2.2 Click-and-Drag Components

Every TTS tool studied uses the same spinner component (see Figure 4 – up left, up right and down). This component is provided for altering synthesis parameters like the speed, pitch or volume of the generated speech. These “click-and-drag” components are extremely difficult to use for motor impaired users. These spinner components are strongly not recommended by accessibility guidelines. In addition to that, only BJHt tool (see Figure 4 – up left) implements a spinner component reachable by keyboard-only users.

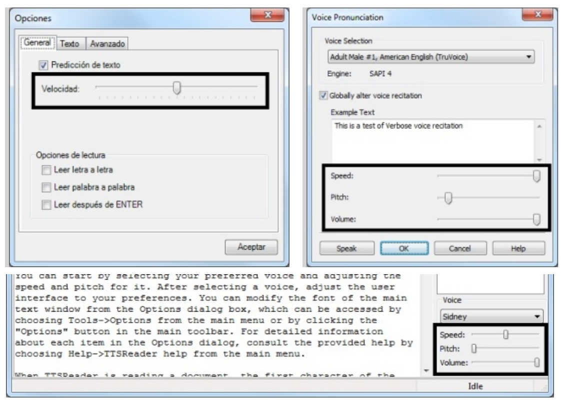

Figure 4: Spinner button components highlighted for each of the three TTS tools analysed (left: “BJ Hermes PC”, right: “Verbose Text to Speech” and down: “TTSReader”).

3.3 Scrolling Issues

3.3.1 In Relation to Dialog Windows

Every TTS tool studied implements the same standard "open file dialog window" (see Figure 5) and "save file dialog window". The scroll bars included to navigate within the file system add an accessibility barrier for people with low precision with mouse device. The scrolling buttons are difficult to access due to their small size.

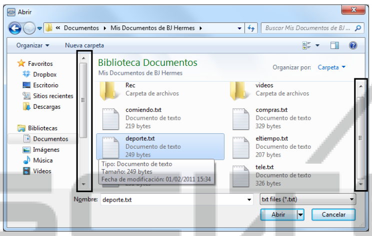

Figure 5: Standard "open file dialog window" as implemented by the three studied TTS tools with scroll bars tagged.

3.3.2 In Relation to Text Editor

None of the three studied TTS tools incorporates specific controls for easily moving the cursor through the text editor content (see Area 1 in every capture of Figure 1 – left, middle and right). Users have to manage with the small sized scroll buttons. Navigating through long texts becomes tedious for some users who experience difficulties with motor coordination and fine movements.

3.4 Keyboard Issues

3.4.1 Out of Reach Controls

The analysed TTS tools include many out of reach controls for a keyboard-only user. It means a total inaccessibility to those functionalities for those users. An analysis has been performed in order to determine the out of reach controls included in each tool. The Areas 2 highlighted in Figure 3 show the inaccessible functionalities for keyboard-only users in the main window. This presents accessibility barriers for some specific group of users even though the functionalities are accessible from other menus.

In addition, VTSt and REAt tools implement not reachable lateral command bars (see Areas 3 in Figure 1 - middle and right) for keyboard-only users. All these not accessible functionalities can be performed in an alternative way. However, the alternative is selecting the option in cascading menus that are not recommended for some groups of users.

Moreover, some components for speech adjustments are completely unreachable for keyboard-only users in VTSt and REAt tools (see Figure 4 - up right and down).

3.4.2 Hotkeys Assignments

VTSt and REAt tools allow the user to define custom hotkeys to perform several commands (read, pause or save text...) by pressing only one key. Both tools present the same accessibility issue related to this functionality: the assigned hotkey is not displayed over the correspondent command button on the GUI. Therefore, users have to investigate by themselves the way to define them and memorize each one in order to get to use them.

On the other hand, BJHt implements not customizable hotkeys. These values are correctly displayed next to the appropriate label (see Area 3 in Figure 1 - left).

Table 1: This table summarizes the results obtained in the accessibility evaluation of the TTS tools (x: accessibility barrier detected, -: not detected, xx: issue detected without alternative way)

	Issue	BJHt	VTSt	REAt
Buttons	Reduced size	X	X	X
	Lack of text	-	-	X
Other components	Cascading menus	X	X	X
	Click-and-drag	X	X	X
Scrolling	Dialog window	X	X	X
	Text editor	X	X	X
Keyboard	Out of reach controls	X	XX	XX
	Hotkeys assignments	-	X	X
	Dialog windows	X	X	X
Other	Lack of text editor personalization	-	X	X

3.4.3 In Relation to Dialog Window

Every analysed TTS tool presents the same accessibility issue regarding dialog windows. Moving to a specific control requires going through a lengthy sequence of tabbing, so the time required to complete tasks increases.

3.5 Other Issues

Related to the GUI personalization, only BJHt implement a customizable option within text editor (font and background features). All the analysed tools implement the possibility of showing or hiding controls (lateral panel, toolbar and status bar).

3.6 Summary Table

Table 1 summarizes the accessibility barriers detected in the analysed TTS tools. The results shown in this table determines the existence of accessibility barriers in each tool and whether the tool provides an alternative way for users with limited mobility in order to avoid those barriers.

4 ACCESSIBLITY ISSUES CONSIDERED IN THE DEVELOPMENT PROCESS

The implemented system called MintzaTek (see Figure 6) is aimed at achieving an accessible TTS tool for motor impaired users with limited movements in upper extremities. The goal is that the implemented functionalities can be accessed and used in an efficient way without using any additional assistive technology.

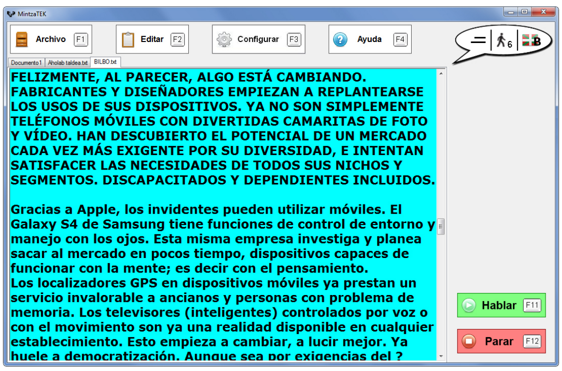

Figure 6: Main window of the MintzaTek TTS tool implemented within this work.

User interface adaptation techniques have been considered in the development in addition to accessibility guidelines. According to Knutov et al. (Knutov et al., 2009), adaptation techniques can be classified in three main groups: content adaptation techniques, presentation adaptation techniques and navigation adaptation techniques. This section presents the techniques considered in the development of MintzaTek.

4.1 Content Adaptations

These techniques involve changes in the content display within the interface. Some interaction components have been replaced with other more accessible.

4.1.1 Avoiding Cascading Menus

We have implemented independent static menus (see Figure 7) avoiding the use of walking menus.

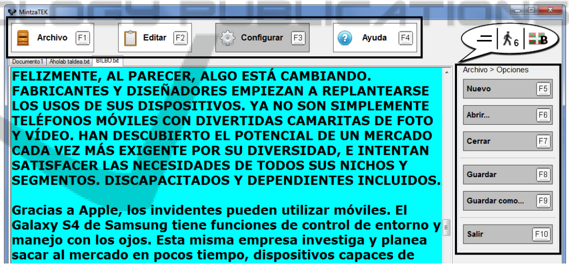

Figure 7: Alternative menu design.

4.1.2 Avoiding Click-and-Drag Components

We have replaced the spinner components with two buttons (see Figure 8): one of them to increase the corresponding parameter and the other one to decrease it.

Figure 8: Two buttons design instead of spinner control.

4.2 Presentation Adaptations

This group of techniques involve style changes in the interface layout. Some aspects of the layout have been changed in order to avoid inaccessible interfaces.

4.2.1 Avoiding Scrolling Buttons

Accessibility issues related with scrolling (see Section 3.3.1) have been solved by applying an alternative design. The total information to present in the user interface has been broken into data chunks (see Figure 9). Additional buttons (previous page and next page) are incorporated when necessary so users can navigate backward and forward through the content.

Figure 9: Dictionary dialog window.

4.2.2 Accessible Buttons

The buttons included in MintzaTek fulfil these necessary characteristics: minimum space between consecutive buttons, minimum dimensions, textual description (see visible buttons in Figures 6 to 10).

4.2.3 Personalization Features

The implemented system allows each user to localize the user interface in two possible languages, to change within different colours schemes (for texts and backgrounds), to resize texts and buttons, or to alter the distribution of the interface main group of elements.

Although this group of adaptations are not related with physical disabilities needs, users with low vision or interacting with computer in unusual postures can benefit from them.

4.3 Navigation Adaptations

This group of techniques involve changes dealing with interface structure and behaviour in order to facilitate the navigation of users. The following aspects have been included in MintzaTek tool in order to facilitate user interaction.

4.3.1 Fast Navigation within Text Editor

To move faster through texts within the editor when only using a keyboard, several commands have been implemented. These are aimed at jumping forward and backward variable amount of characters through text (next/previous word, phrase, paragraph and all text). Also another command has been included to navigate to next text editor tab.

4.3.2 Keyboard-only Full Accessibility

In order to create an accessible tool for both keyboard-only users and low precision mouse users, every button present within the interface can be accessed via direct keyboard button and mouse.

4.3.3 Fast Keyboard Interaction

Every command within all dialog windows also implements direct access via keyboard (see Figure 10). This feature makes more dynamic the interaction of keyboard-only users through dialog windows, reducing the time required to complete any task.

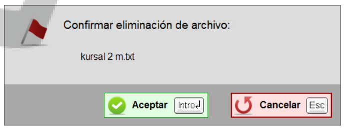

Figure 10: Alert dialog window for deleting file confirmation.

5 TOOL DEVELOPMENT BASED ON USER-CENTERED DESIGN

During the development of MintzaTek we have collaborated with two real end-users. These users experience reduced mobility, precision and strength in upper extremities. In addition, they have difficulties in oral communication due to their motor impairments. They both are usual users of text-to-speech conversion tools and have huge experience with assistive technology. Therefore, both users were involved in our UCD process so we benefit from their knowledge and experience. The objective was to develop a tool accessible for them but also for other users in a similar situation.

5.1 Collaborations with Users

The UCD process applied in the development of MintzaTek was totally oriented to the collaboration of both real users. Every feedback obtained in one step of the process was immediately adopted in next steps. The main steps of the process with direct interaction with users were the following:

1. Preliminary analysis of existing TTS tools
2. Gathering Initial Requirements of MintzaTek
3. User Interface Design
4. User Testing

The first step is related to analyse existing TTS tools. The involved users are experienced in using "BJ Hermes PC" tool. We had the opportunity of observing direct interaction of both users with the system. They both use different strategies for interacting with the interface: one of them emulates mouse device by keyboard and the other uses direct access keyboard buttons for accessing the functionalities. We simulate an interview using the TTS tool and annotate any strategy they applied for the interaction.

Table 2: Initial requirements gathered from users involved in UCD process.

	Initial requirements
Interaction	Full keyboard access to all implemented functionalities
	Simplified GUI that allows reaching any function with the fewer steps
Speech synthesis	Speech synthesis of text in Basque language
	Customizable synthetic communication with several voices, and speech speeds
	Customizable amount of text send to the speech generator (word, phrase, paragraph or everything)
Others	Allow saving texts typed before closing the tool
	Customizable GUI (resolution, language, colors, distribution,...) and text editor (font, font size, font color and background color)

The second step was determinant in order to define the functionalities MintzaTek should provide. We carried out interviews with both users and perform a brainstorming technique in order to detect essential functionalities for the new TTS tool as well as other complementary functionalities which could be interesting in order to facilitate the interaction. For this purpose, results from our TTS tools analysis were used. We presented functionalities found in other tools which were not included in "BJ Hermes PC" to users (for example, the speech tracking feature of "TTSReader"). In addition, we discuss functionalities only implemented in "BJ Hermes PC" which were not in other tools (for example, personalization issues).

Table 3: Improvements and corrections detected after user testing sessions for prototypes evaluation in UCD process.

	Improvements/corrections
Interaction	Allow defining abbreviations to speed up the typing process
	Add keyboard navigation assistance to help moving faster through text editor, and keyboard selection and edition text options
Speech synthesis	Stop the synthetic speech in any moment
	Generate the synthesizer output (synthetic speech) faster
	Add more speed levels to choose from
Others	Allow changing directly a text selection between capital letters and lowercase letters
	Rename several textual information from the GUI

The third step was intended to test the first interface designs of the MintzaTek prototypes. We implemented just the user interface and perform a user evaluation with both users. We observed the interaction and gathered all comments aroused in the sessions. Results from this evaluation sessions were applied in next iterations (for example, a new functionality for defining abbreviations to speed up the typing process was proposed by a user). This step was repeated several times until the design was satisfactory for both users.

The fourth step is related to the user testing of the implemented tool. Users directly interacted with MintzaTek while we observed them and annotated any comment for improving the system. Results from this step were considered in next iterations (for example, in an evaluation session it was detected the need of adding navigation assistance functionality so they could easily navigate through the text). This step was repeated several times until the implemented prototype was satisfactory for both users.

5.2 Results

After all the meetings we had with both users, we can summarize the following results as part of the contribution of the UCD process to our research.

5.2.1 Initial Requirements

After the first couple of interviews with both users, we established the next requirements for the user interface of our TTS tool (see Table 2).

5.2.2 Improvements and Corrections

After several meetings in which we have analysed the way both users responded and interacted with every prototype designed, the following improvements and corrections have been done (see Table 3).

6 CONCLUSIONS AND FUTURE WORK

The existing TTS conversion tools present several accessibility barriers for users with limited mobility in upper extremities. The motor impairments causing these mobility limitations sometimes cause also speech disabilities. Therefore, there is a specific user group who could greatly benefit from TTS conversion tools if they were accessible. The accessibility barriers detected in this research work make user interaction difficult in many cases even impossible in others.

User interface adaptation techniques have been studied for creating alternative interaction components and techniques for solving the detected accessibility barriers. Several content adaptation techniques, presentation adaptation techniques and navigation techniques have been considered for the development of MintzaTek user interface.

The development process of the TTS conversion tool has been an iterative one and based on UCD. Two real motor impaired users with large interaction experience with this type of assistive technology have been involved in the process. The initial requirements were gathered with the aid of these two users and applying techniques such as observations, interviews and discussion activities. User interface design have been tested and commented with both users and prototypes of the system have been evaluated. Feedback obtained from this UCD activities have been always considered in the iterative development process.

We are currently contacting more users with motor impairments in order to conduct formal user evaluation of the developed TTS conversion tool. We also plan to apply eye-tracking methods for obtaining additional information from user interaction. This comprehensive user evaluation will serve to demonstrate the utility of the adapted user interface components included in the tool.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the Spanish Ministry of Science and Innovation through Project ModelAccess (TIN2010-15549). The EGOKITUZ Research Laboratory is supported by the Department of Education, Universities and Research of the Basque Government (Eusko Jaurlaritza/Gobierno Vasco) through Grant# IT-395-10.

REFERENCES

BJ Adaptaciones, 2011. BJ Hermes PC, version 1.1.2 [software]. *Trialware available from: http://www.bj-adaptaciones.com/descargas/BJ_Hermes/BJHermesSetupDemo.exe* (Retrieval: October 2, 2013).

Kurniawan, S. H., King, A., Evans, D. G., Blenkhorn, P. L., 2006. Personalising web page presentation for older people. *Interacting with Computers.* ELSEVIER SCIENCE INC. Volume 18. Issue 3. Pages 457-477.

Newell, A.F., Gregor, P., 2000. User Sensitive Inclusive Design: in search of a new paradigm. CUU 2000 First ACM Conference on Universal Usability, 2000.

National Institute on Aging and National Library of Medicine, 2002. Making You Web Site Senior Friendly: A Checklist. *Available from: http://www.nlm.nih.gov/pubs/checklist.pdf* (Retrieval: October 2, 2013). NIH & NLM.

Gajos, K.Z., Weld, D.S. and Wobbrock, J.O. 2010. Automatically Generating Custom User Interfaces for Users with Physical Disabilities. Journal Artificial Intelligence, Vol. 174, N.12-13, 910-950. Elsevier.

Knutov, E., De Bra, P. and Pechenizkiy, M. AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques. New Review of Hypermedia and Multimedia 5 (1), Pp. 5-38. 2009

SpheNet, 2011. TTSReader, version 1.30 [software]. *Available from: http://www.sphenet.com/Files/TTSReader.exe* (Retrieval: October 2, 2013).

NCH Software, 2013. Verbose Text to Speech, version 2.01 [software]. *Shareware available from: http://www.nch.com.au/verbose/vbsetup.exe* (Retrieval: October 2, 2013).