Basic Human Computer Interface for the Blind

Raymond Borges Hink[1], borgesraymond@gmail.com
Alcides Alvear Suarez[2] aalvear@suagm.edu
Universidad del Turabo, Gurabo, Puerto Rico

ABSTRACT

The graphical user interfaces of modern operating systems have been improving usability at a good rate, but for blind users the progress has been a much slower rate if not declining. It is for this reason that the focus on accessibility should not be to interpret modern GUI's but to develop an application based on the visually impaired people's general heuristic interaction with software. To design a blind person's user interface based on their behavioral characteristics and provide them an independent and enjoyable environment. Using a user centered design working hand in hand with blind people; a basic blind user interface is to be implemented. This interface provides basic interaction but is used independently from installation to task completion. The process will integrate various open source projects that are available under GNU GPL and Open Source Initiative copyleft licenses, so the resulting design will fall under these licenses as well and be open source free software. Some of the applications that are used are: Non Visual Desktop Access, WebbIE web browser and accompanying apps, and Wikipedia and Project Gutenberg public domain free content. The application is currently in its pre-alpha stage and current user experience is being measured to improve usability and gain feedback for improvements.

Keywords: HCI, GUI, heuristics, usability, TTS, open source

1. INTRODUCTION

The idea to create a simple to use computer interface for the blind stems from the absence of one. In my experience in working with blind people for almost 2 years I have observed some of the obstacles they have to overcome to complete their school work. Available programs are not easily or independently usable by the average blind person due mostly to the fact that they work by reading out graphical user interfaces which are not designed for blind people. There are specific usability requirements that have to be met when developing dual interfaces, that is, graphical user interfaces that are adapted for blind users. These include task adequacy, dimensional trade-off, behavior equivalence, semantic loss avoidance and device independence. Consequently, the development of human-computer interfaces that are based on the task, domain, dialog, presentation, platform and user models has to be modified to take into account these requirements (Fernando Alonso et al., 2006).

Today, more than 6.5 million Americans over age 65 have a severe visual impairment, often as the result of diseases such as age-related macular degeneration, glaucoma, and diabetic retinopathy. By 2030, medical experts and health officials predict that rates of vision loss and severe visual impairment in the United States will double as America's 78 million aging baby boomers reach retirement age and beyond (O'Brien., 2009). Without intervention, the number of blind people worldwide could increase from 37 million in 2008 to 75 million by 2020 (World Health Organization., 2008). Approximately 5,626 legally blind children use Braille as their primary reading medium. Of the 57,696 children who are legally blind, 10% (5,626) are registered with the American Printing House for the Blind as Braille readers, 27% (15,303) as visual readers, 7% (3,942) as auditory readers, 34% (19,793) as non-readers, and 23% (13,032) as pre-readers (American Printing House for the Blind, Inc., 2007). Suggesting the project should be based on Text to speech software (TTS). There are open source programs that are free, but separately these programs are useful but not sufficient, although united they may be much more powerful and used to construct a GUI for Windows operating systems that is understandable and enjoyable to use by the blind.

1.1 OBJECTIVES

There is a need to develop a human computer interface focused on usability for text to speech applications to read and so that blind people can easily interpret and comprehend the information that is presented to them. The following goals were taken into account to provide comparable usability to what GUI's provide sighted people.

1. Provide a simpler more engaging experience to the blind user using a human computer interface.
2. Provide the user a system with high usability, a system that will allow a blind user to accomplish simple tasks efficiently.
3. Provide the user with instructions necessary so that the user will never feel lost in the program.
4. Provide an experience that is equally time consuming or faster in each task as it would be to a sighted person to complete times 1.5, or time and one half. For example if it took an average person 10 minutes to complete a task, said task should be completed by the average blind person in less than 15 minutes.
5. Use open source freely available software such as NVDA which is free text to speech software and serves to read out text presented on screen, WebbIE which is a free text based web browser and other copyleft software which can be integrated into one multifunctional interface that provides all the tools needed to complete many simple tasks and is free.

2. USER CENTERED DESIGN

The user centered design model was chosen so that the interface could be correctly designed to be suited for the needs of blind people and based on their heuristic interaction. In a user centered design model every design is based on the feedback of the end user. The feedback is analyzed and reevaluated to produce a better design. The end product of a user centered design is a result of constant feedback and evaluation. The ISO standard for HCI was loosely followed. ISO 13407:1999, titled Human-centered design processes for interactive systems, is an ISO Standard providing Guidance on human-centered design activities throughout the life cycle of interactive computer-based systems (ISO site., 13407:1999").

Choosing an idea or metaphor to design around was the greatest challenge and through extensive interviews and questionnaires that were administered it was possible to come to the conclusion that a radio metaphor could be used since 100% of the blind people that were asked had used some type of radio before or an mp3 device such as an IPod.

2.1 Software Development Process

The incremental and iterative development type of development process was chosen since this type of project of interface design is based on usability and heuristic process that cannot be measured precisely with only decision matrixes, only by constant feedback and reevaluation can good results be obtained. Each process must be developed, tested and improved. The basic idea behind iterative enhancement is to develop a software system incrementally, allowing the developer to take advantage of what was being learned during the development of earlier, incremental, deliverable versions of the system. Learning comes from both the development and use of the system, where possible. Key steps in the process were to start with a simple implementation of a subset of the software requirements and iteratively enhance the evolving sequence of versions until the full system is implemented.In each iteration, design modifications are made along with adding new functional capabilities (Craig Larman et al., 2003).

3. SYSTEM DESIGN

Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. The system design is based on the Use-case scenarios produced by analyzing questionnaires and interviews. Perhaps the biggest advantage of interviewing techniques is that they encourage user participation during interface development (Chris Johnson., 1993.). The following conclusions that were obtained on preferred input and output methods are presented in table 1 and figures 1 and 2.

Users	Users with no answer		Input Methods			Output Methods	
Interviews and questionnaires	Input	Output	Microphone	Qwerty Keyboard	Braille n speak type keyboard	Speakers	Refreshable Braille Display
7	0	2	1	5	1	4	1

Table 1: Results from questionnaires

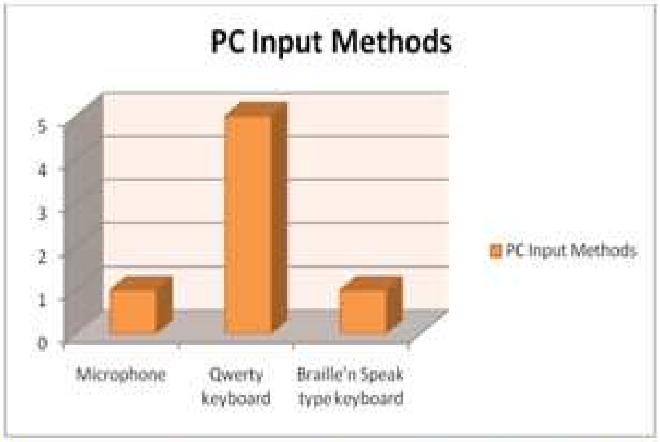


Figure 1: Results for personal computer input methods

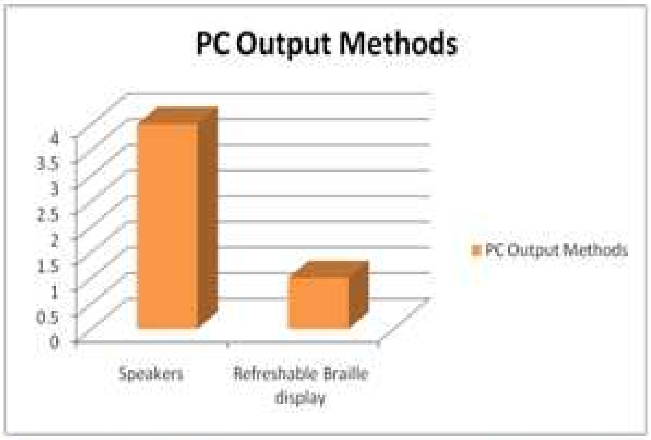


Figure 2: Results for personal computer output methods

Use cases describe the interaction between one or more actors (an actor that is the initiator of the interaction may be referred to as the 'primary actor') and the system itself, represented as a sequence of simple steps. Actors are something or someone which exists outside the system ('black box') under study, and that take part in a sequence of activities in a dialogue with the system to achieve some goal. Actors may be end users, other systems, or hardware devices .Each use case is a complete series of events, described from the point of view of the actor(Ivar Jacobson., 1992). From these results and other interviews the initial specifications and use-case scenarios were obtained. The functional design can be best explained with the following use-case diagram:

3.1 Functional Requirements

The functional requirements analysis for this design was compiled from various interview sessions with blind users and from various questionnaires as well as from observing blind people interacting with computers. In figure 3 we can see a representation of the possible actions one could in the interface and what they involve. We can also appreciate that the interface works hand in hand with the text to speech synthesizer application non visual desktop access which is a freely available open source TTS.

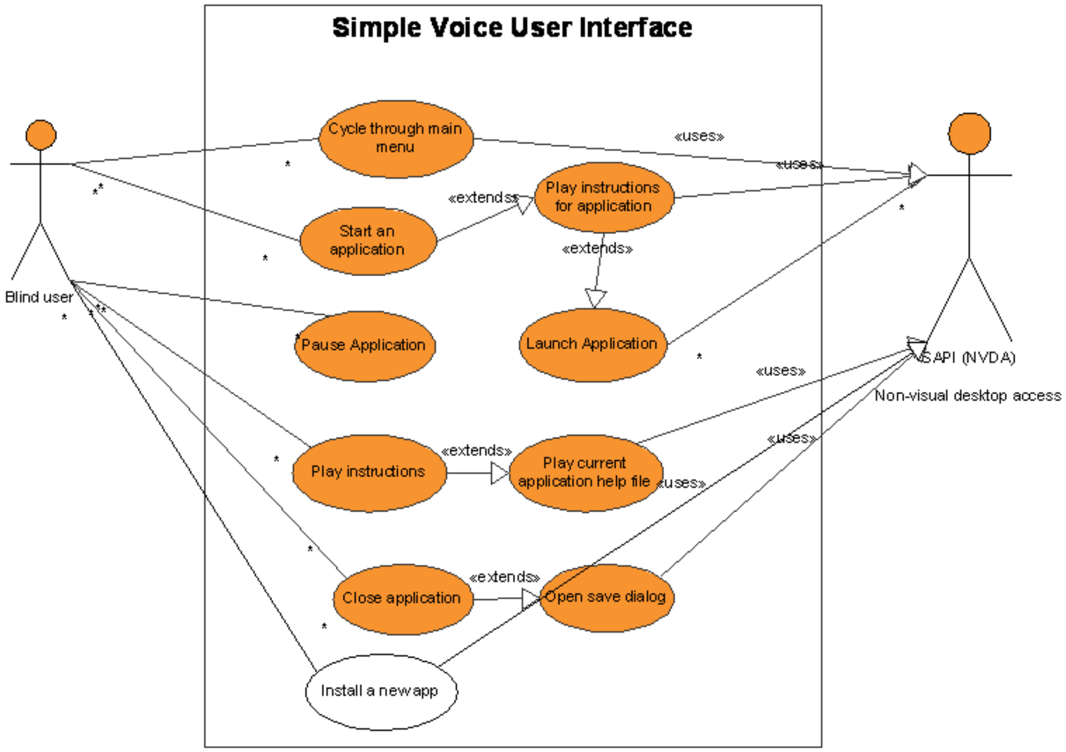


Figure 3: UML Use-Case Diagram for system interface Simple Voice

3.1.1 REQUIREMENTS FOR THE MAIN INTERFACE

- The main application of the system must provide a method for the computer to load into the system or to use the graphical user interface explorer.exe application included in Windows 7. This can be something like a log on screen. Or some other type of small prompt selection screen. This must be made in an audible format and also initialize the voice synthesizer on startup before the interface loads.
- Each application must be linked to its own icon in the main menu. Also there must be a link to the four fundamental folders, my music, my pictures, my videos and my documents. That is the application runs when the icon is pressed.
- Each icon must read out instructions for the application linked to its icon when the icon button is pressed and after it reads instruction it should launch the application. If the icon is a folder then when pressed the pertaining folder must present its contents.
- The system should complement each running application with additional instructions on how to use the application and how to return to the main menu. This will be provided in audible form and based on the applications help file.
- The system should have the capability the pause an application return to the main menu, run a different application and then later return to the same application. This can be accomplished by making the application momentarily invisible to the OS and inaudible to the user.
- The system should provide methods to copy external media to the HDD or to another external media device.
- There should be an icon to shutdown and another for standby.

3.1.2 REQUIREMENTS FOR THE INSTALLER

- The system must provide details to load the system in a variety of situations where the operating system is configured to auto start software loaded in the portable media device. This can include instructions in Braille in a small booklet and or on the CD. It could also be included in an audio file on the device.

- All the applications included in the system must be installed by the software installer. This requires the installer which is to be designed separately to contain the instructions to install all the other applications that are bundled together.
- The installer must use speech at all times since the first wizard form loads. The installer will initiate a portable version of NVDA to serve as the speech synthesizer upon inserting media device such as a CD or flash drive with the system.
- The inserting should provide the ability to alter the speed of the speech synthesizer and instructions on how to do it. With specific shortcuts the user can alter tone, voice and pitch setting of the voice synthesizer.
- The installation procedure should include the end user license agreement. This includes all the end user license agreement for included application as well.
- After finalizing the installation process the application should load the system to the main menu. This should load the main application and shutdown the explorer.exe.

3.2 NON-FUNCTIONAL REQUIREMENTS

- Initial documentation for installation steps and troubleshooting in Braille and normal text should be short and easy to understand. This will be evaluated by using metrics and timing the use of the software.
- It should be fun and simple to use. This will be evaluated by using certain metrics and questionnaires.
- It will be open source (source code available with product). Source code that is free and open source is available on the web.
- It has to be freeware (free of cost) and free as in open source. Not all free software is open source and open source software is not necessarily free. But this application will be free both ways and it must be since it uses other copyleft software.
- The user should be able to complete tasks independently. This means that the user can install the product and complete regular tasks that can be accomplished with the software.
- The application should be extremely stable and can't have more than one active window open at any given time. The application should take complete control over the interface for the user interacting with internal programs.

3.3 FUNCTIONS

1. The application will serve to facilitate the use of the basic functions of a computer such as: browsing the web, writing, searching the web, copying files, sending emails and playing files for blind people.
2. The application will also be made to be installed by the blind person without external help on a Windows 7 operating system.
3. The application will provide the instructions for each application that is used when the application starts.
4. If additional instructions are needed they will be provided when the shortcut key is pressed.
5. The application will provide a wizard to manage instances when the external media devices are inserted, giving the user the option to play the content, copy to another external media device or to the correct location on the computer hard drive according to file extensions. This means pictures will be copied to my pictures folder, text files to my documents audio files to My music and video files to My videos.

3.4 ARCHITECTURAL DESIGN

The system consists of a 2-tier architectural design which is made up by the main interface presentation layer which in this case is audible and is only presented visibly for the case that if there be someone standing beside the blind person could understand and observe what they are doing and if need be assist them. And the sub layer which consists mainly of the controls which are transparent to the user in both visual and the audible sense, it only exists for interactions with the operating system.

3.4.1 PRESENTATION LAYER

The presentation layer of the architectural design of this project will consist of the appearance in visual form of everything that will be presented to the blind person in both visual and audible output. It will be a representation of the application and its parts. Its purpose is to show you behavior not mechanics which is part of the middleware layer.

Figure 4 is the main form in the application, where the user will spend most of the time interacting with the computer. In the main menu form the blind user will be able to scroll through all the applications installed and also through the folders. The main menu will serve as the focal point for all the application interaction. The user will scroll through each application and or folder and choose which one to activate.

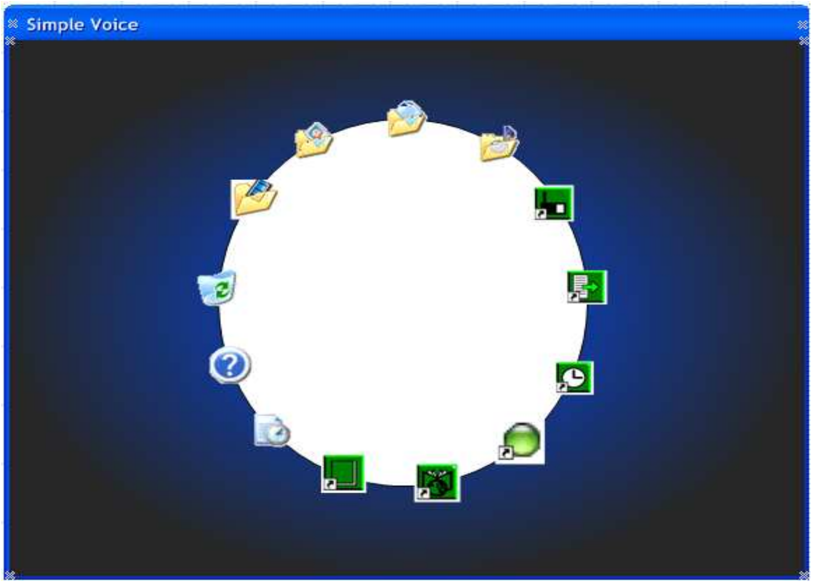


Figure 4: Main menu form layout.

From topmost icon clockwise:

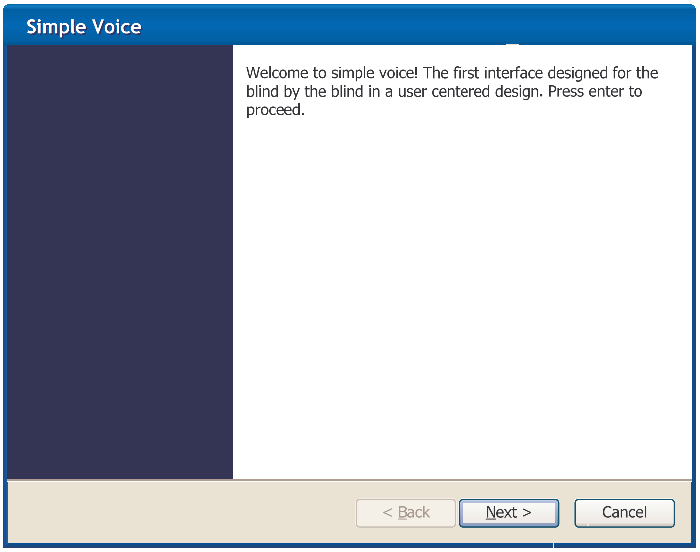
1. My documents
2. My music
3. Internet live radio
4. PDF reader
5. Clock
6. WebbIE web browser
7. Web directory
8. Gutenberg library
9. My recent documents
10. Help
11. Recycle Bin
12. My videos
13. My pictures

There will be an option to add other applications as well. In this case the icon would appear in the main interface once the application is installed. Each icon in the main menu will activate a bundled application, one of its components, a folder or a help menu. Only one icon representing an application may be active at one time with the exception of some small gadgets such as the system time or clock icon among others.

3.5.1.1 Installation Wizard

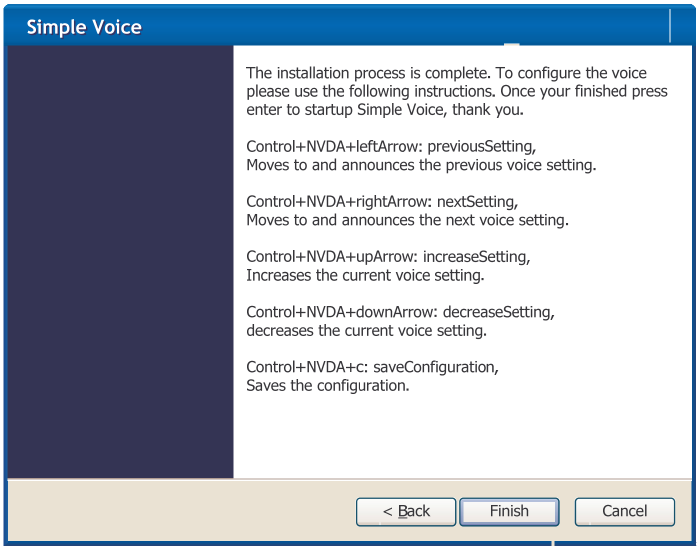
The installation wizard which the first few steps we can observe in figures 5 and 6 has the following goals:

- Install Simple Voice main application.
- Install accompanying applications that will be bundled with Simple Voice
- Explain the end user license agreement.
- Have the user accept or cancel installation of Simple Voice and bundled apps.
- Simplify the installation by automatizing the process.



Welcome to simple voice! The first interface designed for the blind by the blind in a user centered design. Press enter to proceed.

Figure 5: First step of installation wizard.



The installation process is complete. To configure the voice please use the following instructions. Once your finished press enter to startup Simple Voice, thank you.

Control+NVDA+leftArrow: previousSetting,
Moves to and announces the previous voice setting.

Control+NVDA+rightArrow: nextSetting,
Moves to and announces the next voice setting.

Control+NVDA+upArrow: increaseSetting,
Increases the current voice setting.

Control+NVDA+downArrow: decreaseSetting,
decreases the current voice setting.

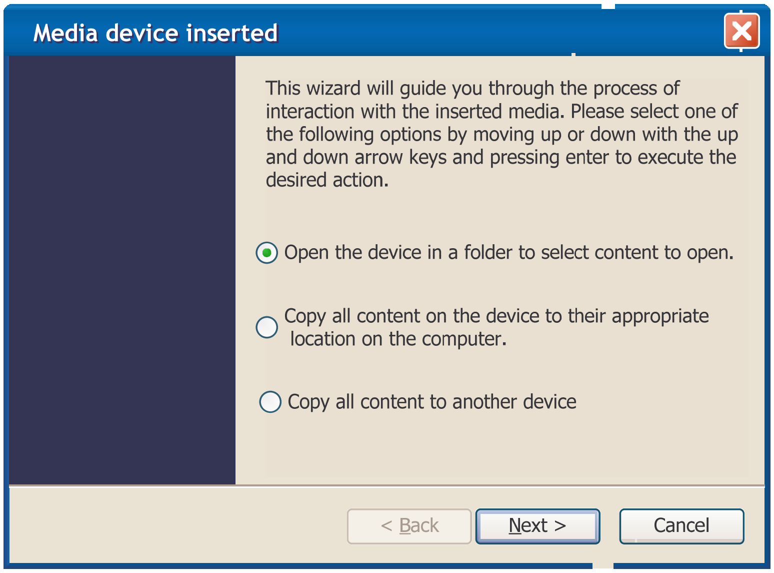
Control+NVDA+c: saveConfiguration,
Saves the configuration.

Figure 6: Fourth and final step of installation wizard.

3.5.1.2 Inserted media device wizard

The inserted media device wizard form which is presented in figure 7 has the following goals:

- Provide the user simple access to content on the inserted device.
- Provide the user simple steps to copy the content to hard drive.
- Provide the user simple steps to copy the content to another external device.



This wizard will guide you through the process of interaction with the inserted media. Please select one of the following options by moving up or down with the up and down arrow keys and pressing enter to execute the desired action.

● Open the device in a folder to select content to open.

○ Copy all content on the device to their appropriate location on the computer.

○ Copy all content to another device

Figure 7: First step of inserted media device wizard.

This is the first step in the process of interacting with an inserted media device such as a CD, DVD, flash drive or external hard drive. It displays 3 options and instructions. It will also deliver the message through the speech synthesizer. There are 3 options to be selected. The first option displays the contents of the device as individual files which can scrolled through and activated individually.

3.4.2 MIDDLEWARE OR SUB LAYER

The sub layer is composed of the mechanics of the application, classes and object relationships. This layer consists of class, objects and methods and is of little importance in the concern of usability since it is transparent to the user.

3.5 Heuristic analysis and usability testing

This interface's main concern is on usability for which it is very necessary to talk about heuristics and testing. To perform testing on any type of application it is necessary to utilize metrics so here they are discussed along with the end goals of the usability rates for the interface.

3.6.1 Metrics

The metrics will consist of the following three areas:

Effectiveness

- Percent of tasks completed
- Ratio of successes to failures
- Workload
- Number of features or commands used

Efficiency

- Time to complete a task
- Time to learn

- Time spent on errors
- Percent or number of errors
- Frequency of help or documentation use
- Number of repetition or failed commands

User satisfaction

- Rating scale for usefulness of the product or service
- Rating scale for satisfaction with functions and features
- Number of times user expresses frustration or anger
- Rating scale for user versus technological control of task
- Perception that the technology supports tasks as needed by the user

3.6.2 Goals

The user experiences have not been measured since we have not yet reached this phase in the project. The development is still under way but taking into account the statistics provided earlier we can deduce that no application will satisfy 100% of the blind populations needs so as a goal the following may be a good end result of user experience when specific metrics are finally chosen:

- All top user tasks must have a success rate of at least 75 per cent.
- User satisfaction should be at least 5 of 7 for all top tasks.

3.6 Interaction

The basic method used to date by the blind for interaction is the keyboard through the use of arrow keys, tab, and other shortcuts and keys but an alternate method could be the one presented in figure 8 which describes the possible blind user interaction with the interface by using the mouse to execute the basic tasks illustrated.

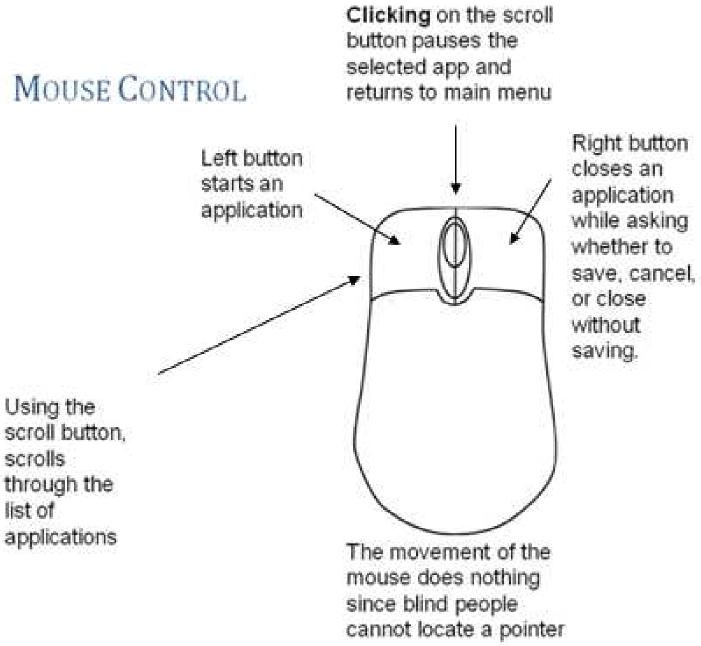

MOUSE CONTROL
Clicking on the scroll button pauses the selected app and returns to main menu
Left button starts an application
Right button closes an application while asking whether to save, cancel, or close without saving.
Using the scroll button, scrolls through the list of applications
The movement of the mouse does nothing since blind people cannot locate a pointer

Figure 8: How a blind person could use the mouse to interact with the interface.

4. Conclusion

The design of this interface was completed and is currently in the implementation phase of the software development model. All parts of this initial conceptual design were constructed based on interviews and questionaires conducted with various blind people in the area of Puerto Rico. The application is still pre-alpha and not a release candidate but further usability evaluation is being conducted and will provide feedback from live recordings of usage and questionaires as well as interviews.

REFERENCES

American Printing House for the Blind, Inc. (2007, September 30). American Printing House for the Blind. Retrieved September 2009, from About: http://www.aph.org/about/ar2007.html

Craig Larman, Victor R. Basili (June 2003). "Iterative and Incremental Development: A Brief History". IEEE Computer (IEEE Computer Society) 36 (6): 47–56. doi:10.1109/MC.2003.1204375. ISSN 0018-9162. http://c2.com/cgi/wiki/wiki?HistoryOfIterative. Retrieved 2009-01-10.

Chris Johnson, 1993. Interactive Systems Design. http://www.dcs.gla.ac.uk/~johnson/teaching/isd/course.html

Fernando Alonso, José L. Fuertes, Ángel L. González and Loïc Martínez (2006). "A Framework for Blind User Interfacing". Facultad de Informática, Universidad Politécnica de Madrid, Spain. Lecture Notes in Computer Science, Volume 4061/2006, pp. 1031-1038. http://www.springerlink.com/content/l40721855mu72030/

Ivar Jacobson (1992). Object-Oriented Software Engineering. Addison Wesley Professional. ISBN 0-201-54435.

ISO 13407:1999

O'Brien, S. (n.d.). About. Retrieved October 2009, from Senior Living: http://seniorliving.about.com/od/visionproblems/a/vision_loss_stu.htm

World Health Organization. (2008, October 9). World Health Organization. Retrieved October 2009, from Regional Office for the Eastern Mediterranean: http://www.emro.who.int/pressreleases/2008/no18.htm

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.