ENABLING VISUALLY IMPAIRED PEOPLE TO USE TOUCH SCREEN PHONES

By

ALEXANDER KIVAISI

Supervised by

DR. AUDREY MBOGHO

Thesis presented for the degree of Master of Science

in the Department of Computer Science

University of Cape Town

February 2015

PLAGIARISM DECLARATION

I know the meaning of plagiarism and declare that all of the work in the thesis, save for that which is properly acknowledged, is my own.

Abstract

Most mobile applications are not designed for visually impaired people. The concept of universal design is not adopted by most application developers. Therefore, interaction between a visually impaired user and the mobile application becomes cumbersome. Assistive technologies such as screen readers have been developed for different mobile device manufacturers for visually impaired users. However, they only support the most popular languages spoken in the world such as English. To our knowledge, although research on assistive technologies for the use of mobile devices by visually impaired people has been done, information on that relating to minority languages such as Swahili is virtually nonexistent.

Therefore, this study investigated the challenges faced by visually impaired Swahili speakers when using mobile phones and then developed a mobile phone application which would enable them to use touch screen phones more independently. As a user center design methodology was adopted, visually impaired people were the focal point for this study. The mobile application was designed and implemented based on the feedback from the participants in all iteration phases and thereafter the final evaluation of a complete application was conducted.

The results of the user evaluation showed that most of the visually impaired users were able to complete the tasks independently and were excited to practice even more. Although, the results showed that the sliding-text entry method had more problems than the Braille-method, still users showed preference for the former method. Furthermore, with regard to spoken feedback, the users were willing to use the app even though the quality of the voice was not natural. Overall, the mobile application developed was found to be usable and preferable. While the challenges were explored on the common basic features, it is believed that this work has laid a solid platform for future extensions.

Dedication

*I dedicate this thesis to the memory of my beloved father **Rogath Thomas Kivaisi (1949-2014)** who died recently after battling with cancer for six years. Father, your unconditional love and support will always remain in my heart and memory forever.*

*I also dedicate this thesis to my beloved mother **Amelia Kivaisi.** Your support and encouragement over the last 22 years of formal education and in life in general will never be forgotten.*

*I finally dedicate this thesis to all the visually impaired people in Tanzania**.***

Acknowledgements

Foremost, I am highly grateful to God for His blessing that continue to flow into my life, and because of You, I made this through against all odds.

My thanks and appreciation to Dr. Audrey Mbogho for persevering with me as my supervisor throughout the time it took me to complete this research and write the thesis.

I would like to thank all the visually impaired people for the time and help given throughout the study. Without their participation, this research would not have been possible.

Finally I thank you, Karungi, for your constant encouragement, prayers, love, support and an attentive ear.

Table of Contents

Table of Figures

List of Tables

1 Introduction

In the past decade, the number of people who have used mobile phones has increased tremendously. The usage has shifted from basic features such as making and receiving calls or sending and receiving text message, to advanced features like recording videos or taking pictures, and even to more advanced features such as tracking and locating objects on the map. The use of mobile phones has cut across different sectors such as education institutions, hospitals, agriculture, business companies, government institutions and entertainment sectors.

In the business sector, different companies have taken advantage of the usage of mobile phone applications on smartphones to attract more customers through advertisements. Customers can now purchase products online via a smartphone and delivery can be arranged without much effort. A new business model has developed whereby mobile phone companies, internet providers and other sectors have benefited through the use of smartphones [1].

In the education sector, smart phones can facilitate the interaction between a teacher and a student [1]. At the moment, there are plenty of courses and teaching materials which are available online for free. Khan Academy**,** a company that offers online classes for free and can be accessed through the iPhone and Android platforms is a good example. Therefore, students who miss school or who are far away can easily access lecture notes, do assignments online and submit them using a smart phone from anywhere, at any time as long as the internet is available.

Recently, there has been a rapid increase in the number of people who download and use mobile health apps through their smart phones [2]. Patients seek for medical information through mobile apps instead of going to the hospital. Some use these apps for tracking their health while others use them for maintaining their healthy diets. Some apps are used for making appointments with doctors, while others are used to look for doctor's opinion on a health problem. Continuous Care for health is one of the apps that provide these options.

Therefore, we believe, the trend of mobile phone usage will increase even more as new solutions are required to solve new daily problems faced by all types of people, by people of different age groups, and people of different abilities. However, the usability factor remains unfavorable for the visually impaired people. For instance, the concept of universal design allows users to access and use products regardless of their age or ability [3]. However, the concept has hardly been practiced and implemented by mobile phone application developers. In addition, assistive technologies such as speech output systems (screen readers) have been developed for different device makes [4]. However, they only support the most popular languages spoken in the developed world such as English. To my knowledge, very little research on assistive technologies has been done, and for minority languages such as Swahili it appears to be nonexistent. Considering the fact that a high percentage of the visually impaired (about 90%) live in developing countries [5], there is a great need to develop applications which can assist this group of people who have been abandoned and forgotten in the society. This study investigated the challenges faced by visually impaired Swahili speakers when using mobile phones in their day-to-day activities. Thereafter, the research developed a mobile phone application which addressed those challenges.

1.1 Objectives

The main objective of this work was to implement a mobile application that would enable visually impaired Swahili-speaking people to use touch screen mobile phones more independently.

Specifically, this work sought to address the following questions:

- **Can the proposed mobile application solve the current challenges faced by the visually impaired people when using mobile phones?**

-

  To answer this question, a series of questions were asked to find out challenges faced in making and receiving calls, reading and writing SMS messages, managing contacts, browsing the Internet, listening to music, and performing calculations. Thereafter, an evaluation was conducted to establish whether users can complete task scenarios independently. Note that the word user and participant will be used interchangeably throughout the entire thesis.

- **How usable is the mobile application to visually impaired people?**

  To answer this question, a System Usability Scale (SUS) test was conducted to see whether the application scores high enough to be usable to the visually impaired society.

1.2 Contributions

The main contributions of this thesis are as follows:

- A free Android application that will enable visually impaired people to use Android touch screen phones independently for calling, reading text messages, listening to music, and listening to the local newspaper content.
- Swahili speech recognition resources, the acoustic model and the language model which can be used for voice user interface programs.

1.3 Ethical Issues

Ethical clearance was obtained in order to conduct user testing. The usage of a Text-to-Speech (TTS) engine and third party libraries required other special legal attention. For instance, the TTS engine which lies under GNU General Public License, which is a copyleft license, requires modified versions of it to be distributed using the same license, which implies legal precautions need to be taken seriously. However, the pocketsphinx library lies under the Berkeley Software Distribution (BSD) license, which is a free, permissive license.

1.4 Overview of the thesis

The remainder of this thesis is structured as follows:

Chapter 2 covers the background and related works relevant to this work. It starts with a brief overview of visual impairment and an introduction to Braille, followed by a review on the speech synthesizer and speech recognition. Chapter 3 describes the process taken in designing the interface used to explore the research question, and the proceeding Chapter 4 gives details of how this design was realized as a final application. Chapter 5 presents the results of the evaluation process. Finally, in chapter 6, a summary of this thesis is provided, followed by the conclusion from the findings, and some future work is discussed.

2 Background

This chapter introduces and discusses the existing academic works and related background knowledge relevant to this thesis. It begins by briefly introducing the Swahili language in section 2.1. In section 2.2, we give a background overview about visual impairment. Next, we give an overview about Braille code in section 2.3, mainly describing different versions of Braille code, and about Braille technologies. Text-To-Speech synthesis is discussed in section 2.4, while Speech recognition is discussed in section 2.5 before discussing related works in section 2.6

2.1 About Swahili

The focus language of this study is Swahili, an under-resourced language spoken by over 100 million people in East and Central Africa. It acts as a vernacular (I.e. it is used for non-official communication between members of different language groups) language throughout wide areas of East Africa [6]. In Tanzania, Swahili is used as a national language and a medium of instruction to teach other subjects at Primary school level. It is an official language in all government sectors and used by the media, namely in radio broadcasting and in newspapers. Although, Swahili is spoken in majority of the countries neighboring Tanzania and in countries in central Africa, Tanzania was selected for this case study.

In the global context, Swahili has been used in news programs by international radio stations and TV channels such as BBC, Deutsche Welle, and Voice of America. It has been used in the internet social media such as Facebook, in Google for language translation services and search interface and in websites such as Wikipedia [7]. It has been incorporated in applications such as Microsoft Office, the Windows operating system, and Linux operating system. There is also a Swahili-English internet dictionary called Kamusi [8].

In the next subsection, a brief introduction to visual impairment will be given.

2.2 Visual impairment

Visual impairment can be defined as any type of vision loss that makes an individual either totally or partially blind [9]. To elaborate more on the definition, the World Health Organization (WHO) grouped this vision loss into the following categories [9]:

- Category 1 (Moderate visual impairment).
- Category 2 (Severe visual impairment or low-vision).
- Category 3-5 (Blindness).

Note that this research used the term visual impairment relating only to low-vision and blindness, excluding category 1.

According to WHO, visual impairment is caused by uncorrected refractive errors, un-operated cataracts, and glaucoma [5]. However, cataracts are considered to be the main cause of blindness accounting for more than 50% of all eye diseases [10]. For many years now, in many countries, there has been a dramatic development in preventing and curing visual impairment [10]. However, despite the fact that visual impairment can be either prevented or cured, the problem is massive and efforts to address it fall short. Some causes like cataracts cannot be prevented at old age and require an operation to restore eye sight. Unfortunately the majority of visually impaired people live in low- and middle-income countries as reported in [11], and most of them cannot afford the operation.

According to WHO, the population of visually impaired people around the globe is approximately 285 million people [10]. There is no official statistical data about visually impaired people in Tanzania. On the basis of a strategy report [58] by the Comprehensive Community Based Rehabilitation in Tanzania (CCBRT), the number stands at about 940,000. This is a lot of people to ignore when it comes to enjoying the benefits of modern advances in mobile phone technology. Visually impaired people should be able to operate touch screen phones without any assistance. They should be able to use them both for communication and information access and for entertainment as well.

The following section introduces the Braille code, followed by an overview of Braille technologies.

2.3 About Braille code

Unlike sighted people who use visual alphabets to read and write, visually impaired people use Braille code as their writing system. The Braille code represents six raised dots or cells, arranged in a pattern of three rows to two columns or a group of six dots (see Figure 1). The alphabetical letters and numerals are made up of a combination of these dots. For instance, a letter “a” is represented by a single raised dot, positioned at the first row, in the first column (see Figure 2).

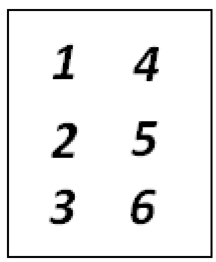

Figure 1: Six dots Braille cells.

There is a number of different versions of Braille code [12] (See Figures, 2 and 3):

- Grade 1: consists of the 26 standard letters of the alphabet and punctuation. It is mainly used by people who just started reading Braille.
- Grade 2: consists of the 26 standard letters of the alphabet, punctuation and contractions. The contractions are employed to save space because a Braille page cannot fit as much text as a standard printed page. Books, signs in public places, menus, and most other Braille materials are written in Grade 2 Braille.

- Grade 3: is used only in personal letters, diaries, and notes. It is a kind of shorthand, with entire words shortened to a few letters.

With these different versions, this thesis selected Grade 1, a grade that seems suitable for most visually impaired people in Tanzania because it is taught in primary school.

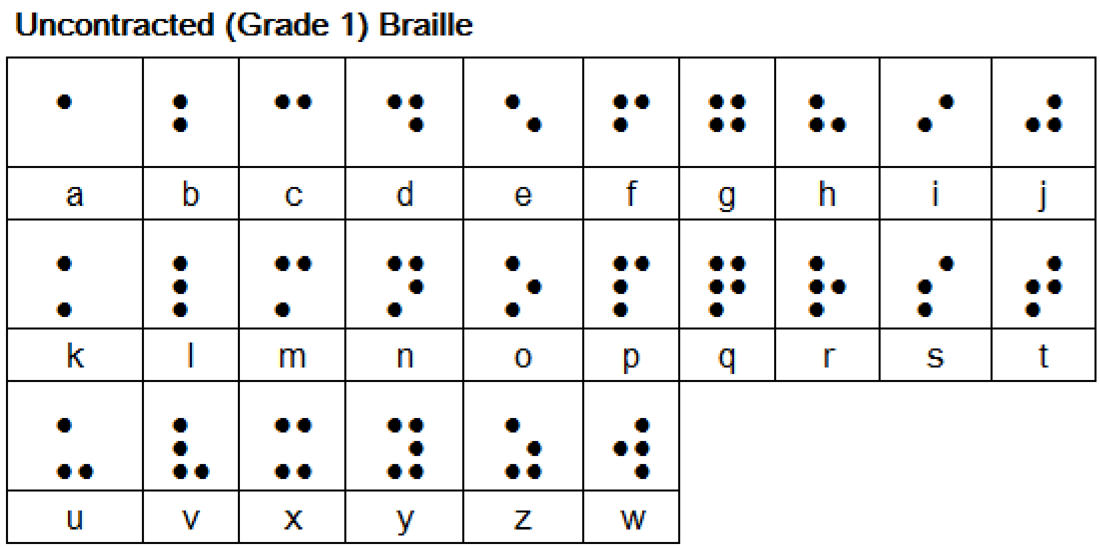

Figure 2: Shows 26 standard Braille code for letters [12]

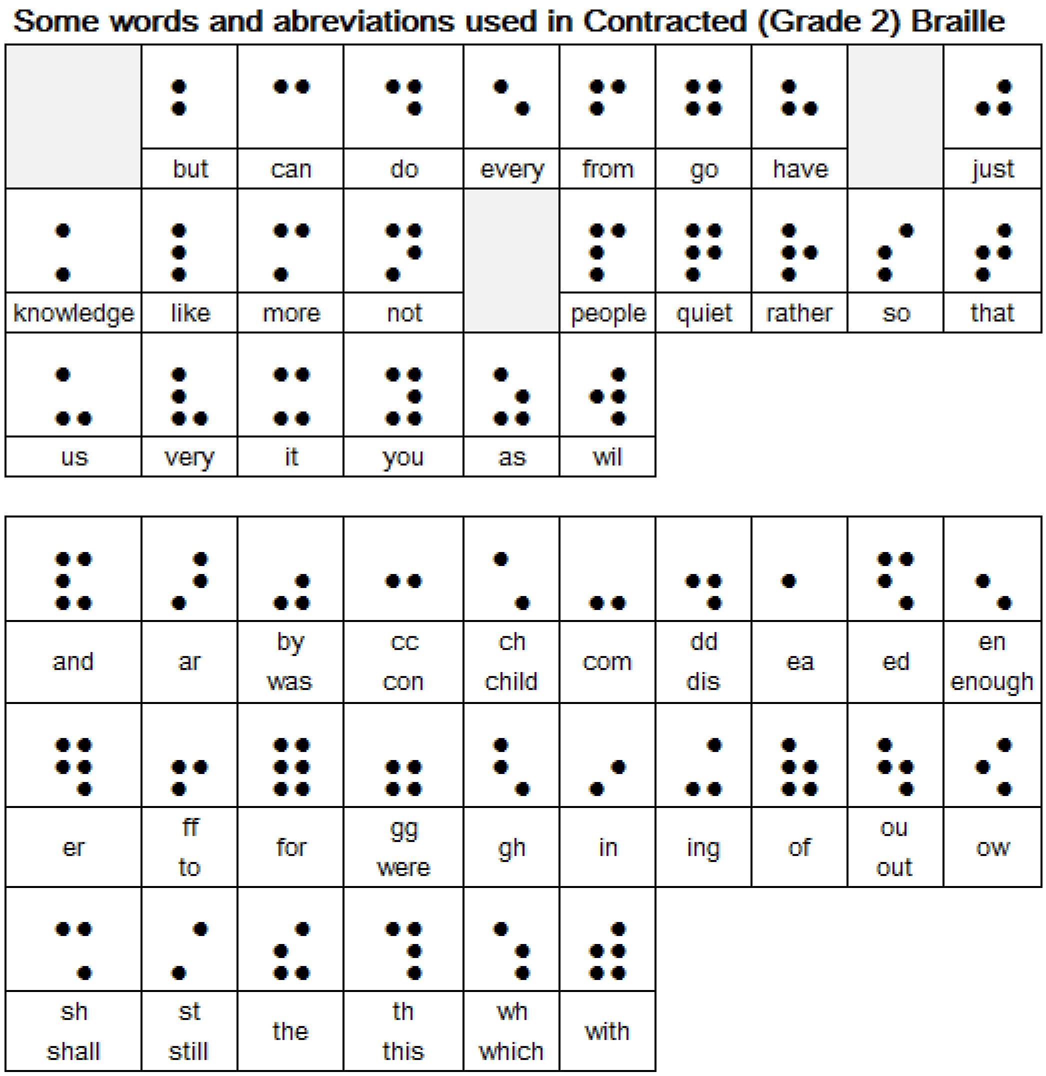

Figure 3: Shows English Braille code contractions [12]

In Tanzania, children learn Braille in primary school, starting from Grade 1 and they use it for the rest of their school life. For those who become blind at later ages, the learning process becomes somewhat difficult, compared to children.

However, there are few printed Braille materials such as brochures, news papers, and books. This means that visually impaired people lack access to information compared to sighted people. Therefore, there is a need for alternative ways of providing information to visually impaired people besides the printed Braille code. This study addressed this problem through developing a mobile application which will enable visually impaired people to access information through web news.

2.3.1 Braille Technologies

Modern technology has made many useful tools for people who read and write Braille. Braille technology allows visually impaired people to do some common tasks such as writing and reading information on computers and from the Internet. Some devices are simple and inexpensive and others are very complicated. The devices below are used by many people who read Braille to complete their work:

- A Slate and Stylus [13] (See Figure 4.a), which are like a pen or pencil which are used to emboss raised dots onto paper.
- Braille Display [14], an electronic device that displays Braille characters on its surface by raising round dots
- Electronic Braille Notetaker [15], an electronic device that acts like a smartphone without the phone capabilities.
- Braille Printers (Embossers) [16], a printer that converts text into tactile Braille cells.
- Braille writers (Perkins braillewriter) [17] (See Figure 4.b), similar to a typewriter for writing texts in Braille.

There is also a screen reader, a software application that reads displayed screen content aloud. It converts text, menu items, symbols or icons into voice when selected. For Braille readers, sometimes, the output of the screen reader can be transferred to a written Braille code. The following are the most popular computer screen readers that are currently available with Braille support:

- JAWS [18]
- NVDA [19]

In related works section, discussion on screen readers will continue, focusing on the most popular Smartphone screen readers. These screen readers, mainly produce text-to-speech output instead. Text-to-speech is discussed in the following section.

Figure 4: (a) a slate and stylus. (b) a Perkins braillewriter [13, 42].

2.4 Text-to-Speech (TTS)

A text-to-speech synthesizer can be defined as computer software that converts computerized text into speech. This speech word is often made up of small natural speech units that are joined together [59]. The quality of the spoken word depends on whether or not the entire word is stored in the synthesizer. For applications that are designed for a specific purpose, storage of entire words is required. For general purpose applications, parts of the spoken word or phones or diphones are required.

The following are the three most common methods used to produce synthesized speech [20]:

- Articulatory synthesis, models the vocal track to produce speech sounds by simulating the movements of the speech articulators.
- Formant synthesis, is based on the source-filter model that models both the glottal pulse train and the formant resonances of the vocal tract.
- Concatenative synthesis, uses recorded human voices to produce natural synthesizing sound after being rearranged or concatenated.

Each of these methods has strengths and weaknesses and the intended uses of a synthesis system determined which approach was used in this research work.

Articulatory synthesis is considered to produce high-quality synthetic speech theoretically compared to other methods [20]. However, it is not that popular based on the fact that its implementation is complicated and a computational burden. Unlike concatenative synthesis, formant synthesis does not use human speech units for creating speech. As a result, the voice is more unnatural. The generated voice is more of an artificial robotic sounding speech. However, at high speeds, formant synthesis is considered to be more useful for applications such as screen readers [60]. In addition, a database is not required for storing speech units which makes formant synthesis an ideal choice for speech synthesis in small devices such as mobile phones.

Speech synthesis has been used mostly as an assistive technology for visually impaired people by integrating it with a screen reader [62, 63]. However, this research focused on speech synthesis which can be used directly by the mobile app as an auditory feedback without a screen reader. A more elaborate discussion on speech synthesis in the Android platform can be found in the related work section.

In the next section, speech recognition which can be seen as an alternative method for text-entry will be introduced.

2.5 Speech recognition

According to Chou and Juang [21] speech recognition is defined as: "*converting the speech waveform, an acoustic signal, into a sequence of words*". In other words, speech recognition can be defined as automatically converting spoken words into text. The following major components make up the so called speech recognition system (see Figure 5) [21]:

- The acoustic model contains statistical representation of the speech segment that makes up a word. It represents speech segments, phones obtained from training speech corpus data.

- The language model constrains search by limiting the computation of possible words that need to be considered in a decoder. The constrained search can either be exact or probabilistic.
- The recognizer lexicon or word lexicon provides pronunciations for words. A single word might have more than one pronunciation. Therefore, depending on the application context, including varieties of a word pronunciation makes a better speech recognizer.

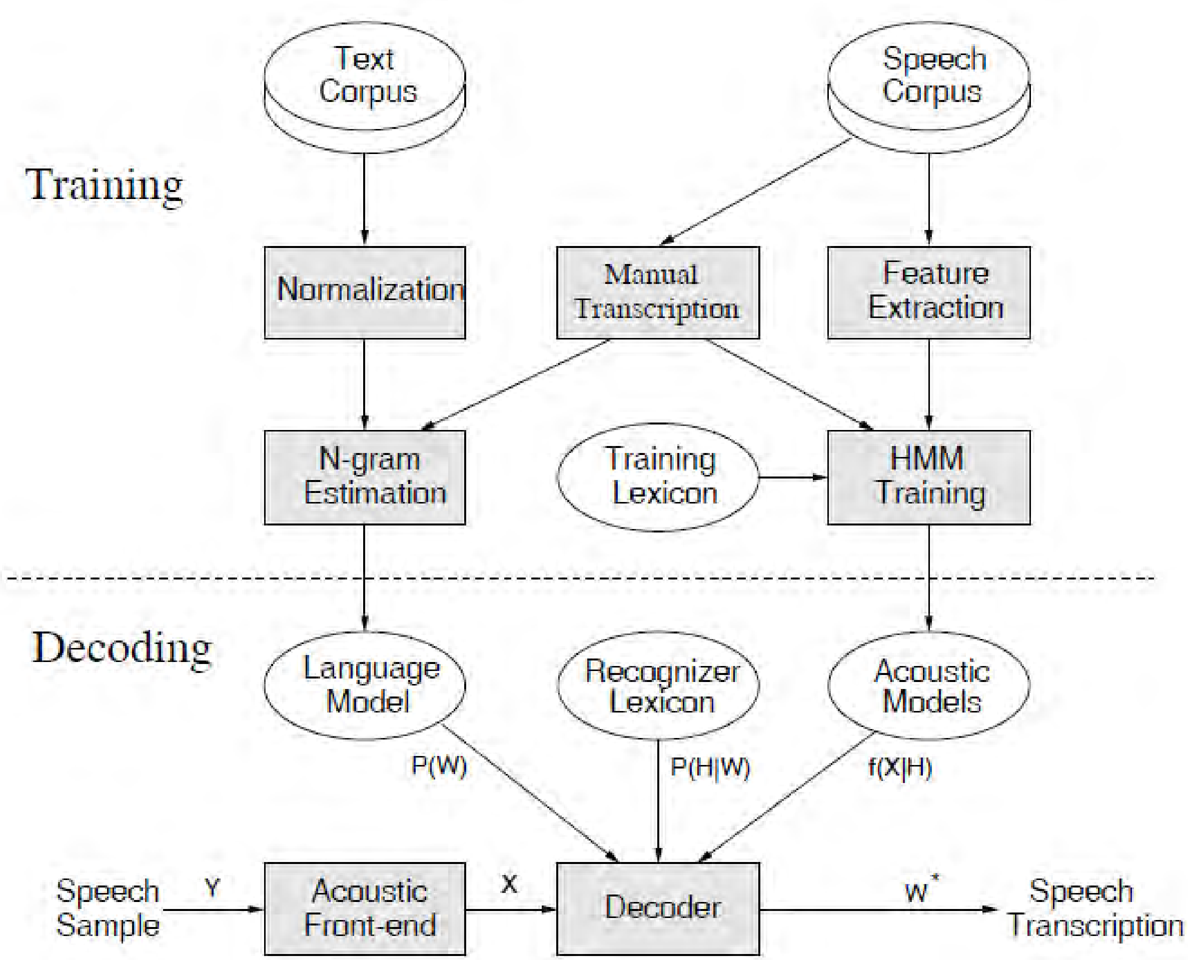

Figure 5: System diagram of a generic speech recognizer based on statistical models, including training and decoding processes and the main knowledge sources [21].

Improvement in speech recognition performance in recent years has prompted it to be used as an alternative input method in some computer and mobile applications. The following are the main speech recognition application tasks according to Rabiner [23]:

- Recognizing single words that perform an action. A good example is for those applications that control a computer by using speech commands like “shutdown”.
- Recognizing two or more connected words that also perform an action. A command such as “Call John” may be used in voice dialing applications.
- Recognizing fluent speech sentences and transforms them into text. Dragon NaturallySpeaking is one of the most popular applications that perform dictation.
- Ability to understand the input speech sentences without recognizing every word. A good example is that of voice control of avionics and displays in single seat aircraft.

Context-free grammars (CFG) and N-gram language models are the two most common language models. Based on voice dial applications task, this research selected to use context-free grammar. These grammars are suitable for spoken dialogue systems. Wang et al. [61] stated that “*A CFG-based system is only good when you know what sentences to speak, which diminishes the value and usability of the system”*. N-gram language models however are generally used in large vocabulary applications, for example to support a dictation method. More information about speech recognition in Android can be found in related works.

Finally to evaluate the recognition system, the Word Error Rate (WER) is used. According to Jurafsky and Martin [24], WER is “*based on how much the word string returned by the recognizer differs from a correct or reference transcription*”. To calculate word error rate given the Levenshtein distance, normalization is performed on the number of editing steps by dividing this by the number of total words in a correct transcript as follows [24]:

$$WER = 100 \frac{Insertion + Substitutions + Deletions}{Total\ Words\ in\ Correct\ Transcript}$$

In the next section, literature related to this thesis, particularly on text-to-speech or screen readers, user interface navigation, text-entry methods and voice user interfaces will be discussed.

2.6 Related Work

2.6.1 Text-to-speech (TTS) and Screen reader

Assistive technology in the form of the screen reader in mobile phones has been a great benefit to visually impaired users. It has enabled a visually impaired user to read and navigate the content of a screen through hearing. Users are able to perceive and navigate the content on the screen, making it possible to perform tasks like word processing, listening to music and surfing the web, by listening to voice communication. In addition, the screen reader can be used to get feedback on the sent and received SMS [25, 26]. These screen readers have been popular for both computers and phones. Nowadays, they are even becoming available for phones with touch screens. Windows Mobile and Symbian OS, for instance, have a screen reader called Mobile Speak. It supports both phones with physical buttons and with touch screen. TalkBack on the other hand, an open source screen reader for Android OS and Voice Over, for iOS, both enable developers to incorporate screen reader functionality into their application.

A screen reader is generally made up of two components: the application that monitors the content on screen, and a synthesizer, which provides the spoken feedback. The limitation of this design is that any application developed does not have full control of the content to be spoken by the synthesizer. Therefore, the screen readers mentioned above are not suitable for the development of the intended application. Hence, eSpeak [27] which is the only open source software speech formant synthesizer that supports majority of minority languages such as Swahili for Linux and Windows OS was integrated. eSpeak has been ported to other platforms, including Android which is an added advantage.

2.6.2 User interface navigation

For these screen readers to work efficiently, applications used in the phones need to follow a common standard so that the content can be integrated and presented correctly [28]. However, this is hardly followed. Although mobile phones include a screen reader as an accessibility feature, still the user interface for a majority of the mobile apps is designed for sighted people. It is clear that design for usability differs depending on whether the final user is sighted or visually impaired. However, currently, the user interface design for mobile apps is not done for visually impaired people. Sanchez et al. [25] proposed specific apps that are needed for visually impaired users in order to obtain the best user experience. They further suggested that, in order to boost the ease of use and learnability as well as satisfaction level for visually impaired users when using mobile phones, specific design requirements for general apps, such as calendars, and contacts, need to be addressed. Therefore, this study took this direction in developing an app, specific for Swahili speaking visually impaired people.

Studies have proposed several solutions to improve the navigation of touch screen for visually impaired users. Karlsen et al. [29], presented Applens and LaunchTile interfaces that enable users to use PDAs and cell phones with one-thumb. In Applens, users can control objects directly using their thumb while in LaunchTile interface, objects are controlled through the cursor. Results showed that the participants liked the way the navigation and selection of applications worked, although they were hesitant with the gestures. Kane et al. [30] proposed an interface that entirely does not rely on visual presentation but depends on auditory feedback and touch screen gestures. The results showed that Slide Rule was significantly faster than a button-based system and was preferred by a majority of the users. Although there were more errors in the earlier method in terms of command gestures, this thesis adapted some of the techniques used. It is in my belief that some of the gesture errors were because of inadequate training for which more time could be provided.

2.6.3 Text-entry methods

The Multi-press system is considered a traditional text-entry method that has been used on the mobile phones for quite sometime on the 12-key pad. One or more key presses are required in order to input a single character in this system. Lagoá, et al. [31] considered this method to be difficult to visually impaired people due to the need to memorize the key letters. Campos et al. [32] also discounted the interface during the preliminary tests, arguing about the size of the keys being very small and also involving a high number of interactions. They further argued that, even if Braille characters were to be printed on the key pads still a blind user would find it difficult to distinguish between characters due to the closeness of the keys in the keypad.

Although the QWERTY keyboard is considered as a standard text-entry method for a majority of touch screen phones, it is considered to be error prone and consumes time for blind people [33, 34]. The introduction of touch screen phones has made it possible for the Braille alphabet system to be implemented and used by the visually impaired. Different research works [34-37] have come up with different types of Braille methods, most of which require to use at least three fingers from each hand for input. This research adopted this design while accommodating Swahili for auditory feedback.

2.6.4 Voice User interface

Speech recognition can be another alternative for text input. However resource limitation, independent users accents, noisy environments and a large vocabulary make the process difficult [32, 35]. Nevertheless, its application on voice user interfaces [23, 38, 39] such as voice dialing or simple data entry has potential for use by the visually impaired when interacting with a mobile app.

PocketSphinx[40] is a lightweight recognizer library that can be imported in the Android platform. This recognizer was chosen because it has clear documentation and a support forum especially on how to prepare resources and training the recognizer. In addition to that, there is clear documentation for integrating PocketSphinx into the Android platform, which is not the case with other speech recognizers.

Swahili as a minority language has very limited resources for building a speech recognizer system. Recently, research done by Gelas et al. [41] created both a Swahili speech corpus and a text corpus. This work used these resources, especially the speech corpus for the acoustic model.

The above discussion led us to conclude that if the development of the mobile app was to succeed in this study, different techniques and tools needed to be adopted. The following chapter describes the methodology and methods used to conduct this research.

2.6.5 Assistive technology smartphone applications.

There has been significant interest in developing smartphone applications for visually impaired people. The following is a list of some of these applications which have tried to focus on specific problems visually impaired people normally face in their daily activities:

- AccessNote allows the visually impaired to take notes as well as search for certain keywords on the phone.
- TapTapSee assists visually impaired people to identify objects by taking a photograph of an item and the app will speak the item back.
- VizWiz enables the visually impaired to be aware of their surroundings by receiving answers after asking questions.
- VisionAssist assists the visually impaired to magnify written things anywhere they go.

- Prizmo enables visually impaired people to scan documents or receipts and export them into text for dictation.
- The Color identifier helps the visually impaired to identify colours in real-time using camera.

AccessNote and TapTapSee are available for Android devices and in iOS devices. The remaining applications in the above list are only available for iOS devices. However, the focus for this research will not be on the specific challenges mentioned on the list above. Therefore, no designs from these assistive technology applications were adopted into the proposed application.

3 Design

3.1 Introduction

The main focus of this thesis is designing a user interface (UI) that is user friendly. In order to achieve that, user centred design was used as a methodology for designing the overall mobile application. This work is a continuation of previous works and hence the design methodology did not include gathering mobile phone features. The focal point here was on the challenges experienced by the users of the current mobile phones features within the context of human computer interaction and usability. Therefore, the design process was done iteratively. Nielsens' usability principles [46] were followed together with design heuristics [42, 43] for mobile development as well as for evaluation and adopted on each iteration. Since the target users were visually impaired people, the number of interactions were to be kept at a minimum to avoid memorizing and confusion in accordance with Brook's [47] recommendation. In addition to that, sound or audio feedback through the speech synthesizer made it possible to interact with the newly designed UI.

The first iteration described in section 3.2 began with gathering user requirements through informal meetings in order to gather early opinions. These opinions helped us in designing and implementing a first prototype which was later evaluated in the second iteration (section 3.3) and changes were made when required. Any changes made were thereafter evaluated in the third iteration (section 3.4). New changes were noted after evaluation and finally implemented. The completed prototype was ready for use in the final evaluation. Consent forms (see appendix C) were provided in each iteration to the participants explaining the objective of the study and other related issues.

3.2 Iteration One

3.2.1 Gathering requirements

To find out the current situation, 20 participants were interviewed, all visually impaired, at the Tanzania League for the Blind organization offices. The interview questions consisted of both closed questions and open questions (Refer to appendix B). Each session lasted about one hour. The main aim was to find out their experiences and challenges strictly when performing the following tasks using mobile phones in their day to day activities:

- Making and receiving calls.
- Reading and writing SMS messages.
- Managing contacts.
- Browsing the Internet.
- Listening to music.
- Performing calculations.

3.2.2 Questionnaire Results

The interview questions were made with an assumption that there would be a possibility to find users who do own mobile phones and some who do not. All 20 participants (only four (4) with low vision and the rest blind) did own a mobile phone. All participants had feature phones except two of them who also had advanced phones which supported screen readers. All of the participants were taught to use a mobile phone by a close friend or a relative or a family member. None were self-taught. The process of learning how to use a mobile phone was all about memorizing the keys' positions as well as memorizing the number of touches to reach into a specific menu item. All participants learned Braille in primary school. However, only three of them use it regularly to write materials and teaching notes using Perkins Brailler machine. The following sections explain feedback provided by users (See Appendix F for more information on questionnaire results).

3.2.2.1 Make and receive Calls Issues

All participants said that they could make and receive calls on their own. However there were some challenges concerning doing both actions. Here are the issues raised:

- Recognizing a caller's identity by name or by number.

  Most of them found it difficult to recognize a caller until may be a conversation was started after accepting a call. Otherwise, most of them normally asked for a caller's identity before beginning the conversation. One participant even put ring tones on some of the important contacts to help identify the caller's identity. So when the phone rang it became easy to recognize who was calling by the ringing tone. However, it became confusing when many contacts were assigned the tones because it became difficult to remember which tone belonged to which contact.
- Searching for a contact in a phonebook.

  Majority had a problem finding a person in the phonebook to call. One participant mentioned that he would call someone thinking it was the correct number and later would find out that it was not. The most common way of calling someone was direct dialling because some of the participants could memorize up to twenty mobile numbers.
- Verifying a dialled number before calling.

  Most participants mentioned making a wrong call simply because there was no method to verify the number.
- Identifying network status.

  Other participants had difficulties during calling especially when there was no network and the mobile phone could not notify them about the situation.
- Memorizing steps

  Other participants mentioned the cumbersomeness of memorizing all the steps for all the programs.

3.2.2.2 Read and Write SMS Issues

Eleven out of the twenty participants could write text messages while the remaining nine required assistance to do so. Eighteen of the participants relied on the assistants to read the SMS for them while the remaining participants had phones with voice reading programs. Here were the issues raised:

- Message verification after finishing writing.

  For those who could write text messages, sometimes the message was not clear. They had no verification of what they were writing until they asked the assistant to help them. One participant said, “Imagine now there is no assistant around; what can you do to correct what you have written?” The other participants said that this was the main problem when one is writing a text message. One of the them added, “Sometimes it may happen that I want to make a correction. Instead of correcting one word for instance, I end up deleting everything and start over.”
- Notification in most parts of the program.

  One participant said, “For instance you might press a button accidentally, and exiting the writing program without knowing, and you will think you are still there writing. This is because I do not know how much I have deleted when pressing the delete key.“
- Message delivery notification

  Another participant said, “Sometimes I may think the message has been delivered after pressing the send message button; unfortunately, it was not the case.”
- Privacy issues

  For those who could not write a text message, it was a difficult situation especially when there was no one to write the message for them. One participant remarked, “Sometimes there are things which are private to you especially family matters or issues of money as we do use M-Pesa”. Another participant added, “There is no privacy. Imagine if you want to send a special message to your girlfriend or your friend, and you do not want anyone to know, it becomes difficult. So even when there is an assistant still it becomes a problem when it comes to trust." Therefore in order to solve this problem some of them normally tell their closest friends to call them instead of texting.

- Limitation of screen readers
    - Two participants said that, sometimes it was difficult to understand the voice during the message reading. Most of the pronunciations were difficult to understand because the synthesizer uses an English voice.
    - Other participants said that the program was only a demo. It worked for fourteen days only, and after that they could not use it. One of them continued to say, "The program does not allow you to read a text message directly or tells you what keypad buttons, letter or number you have pressed during text writing".

3.2.2.3 Internet, Music and Calculation Issues

No participant had used the internet on a mobile phone, however a few of them had used it on a computer before. Three of the participants neither listened to radio nor music on their mobile phones. No participants used calculators on their mobile phone.

3.2.2.4 Issues regarding Touch screen phones

No participants owned a touch screen mobile phone. When they were asked if they would love to own one most of them said no. Here is what they said about touch screen phones:

- "It is not user friendly. It is flat, and for us in order to use it, we need to feel or sense what we are touching in order to navigate. At the moment it is far from that".
- "The structure of the phone is difficult especially when there are no physical buttons. It looks like as if am drawing something on a screen and I do not know exactly what I am drawing".

- “Sometime you have to accept it, we will not be able to use these phones. “
- “I will not be able to buy these phones even if you sell them at a lower price, around 10Tzsh.”
- “I will not be able to use these phones even if someone teaches me. There are no landmarks and the screen is flat. You have to assume or guess where to touch on the screen to select something.”

3.2.3 Recommendation

All of the participants recommended that a program should have auditory feedback. This would notify them at any point in time in the program. It should tell them what they are doing and where they are when navigating different menus. It should tell them what keys are pressed during text writing and it should be able to read the message in a way that is understandable. This auditory feedback should have good Swahili pronunciation and those who are not literate in English should be able to use it.

Most of them suggested that the application should be affordable as the majority of visual impaired people have low incomes. One mentioned that the JAWS program is very expensive.

Other recommendations were that:

- More training and seminars for using phones which have assistive technology be provided.
- Laws which would force mobile companies to design phones for everyone including visually impaired people be put in place and implemented.
- Braille code be included into airtime recharge vouchers.
- Security for the assistive technology application be included to ensure that only the owner is able to use it.

3.2.3.1 Tasks Supported in the Application and Limitations

Looking at the issues raised by the participants with respect to their touch screen experience, we decided to implement a mobile application that will be deployed on the touch screen phones, specifically Android phones. The main task that this new mobile application allows the user to do is to perform the basic mobile tasks that most participants do. All the tasks can be described in terms of the following 5 basic operations: managing calls, managing messages, listen to music, performing some basic calculations and listening to web news.

3.2.3.1.1 Managing Calls

This includes managing of contacts, ability to make and receive calls. The application will enable the user to add, edit or delete a contact. It will also enable the user to choose between two methods for making a call, voice command or by selecting a contact within a phone book. The choice of the voice command method was due to the fact that we thought it would have been easier for users to perform the task more efficiently than with the other methods. The app will also enable the user to simply receive a call with ease.

3.2.3.1.2 Managing Messages

Through managing SMS, the app will enable the user to create new SMS, read incoming SMS or outgoing SMS and delete. The reading will be mainly done using voice. So the user will be able to listen to these text messages by voice. Navigating and typing will also be guided by voice. More details are discussed in the design subsection.

3.2.3.1.3 Listening to Music

As for entertainment, the app will enable the user to listen to music. The user will be able to navigate and select different songs, play them and pause them if required. However, this project limited itself to those features only. Editing, deleting, and importing or exporting songs to other devices are outside the scope of this research.

3.2.3.1.4 Performing basic calculations

The app will enable the users to perform some basic mathematical calculations. This includes performing addition, subtraction, division, and multiplication operations only. Again, voice command technology will be used to perform this task instead of typing. The reason for this choice was the same as that for making calls, namely that it would be efficient.

3.2.3.1.5 Listening to Web News

In order to be informed of what is happening both locally and internationally, the app will enable users to listen to web news. The source of news will be only limited to two local web newspapers, Mwananchi[1] and Raia Mwema[2] due to easy availability. Other web news will come from BBC Swahili.

3.2.3.2 Task analysis summary

The app has capabilities to handle all the tasks identified above. The focus for this research was on managing calls, managing messages and listening to music. The reason for this choice was that these tasks were found to be the ones performed regularly by the majority of the participants. Therefore, these tasks were the ones pursued and evaluated.

---

[1] http://www.mwananchi.co.tz/
[2] http://www.raiamwema.co.tz/

3.2.4 Preliminary Design

3.2.4.1 Introduction: General overview

The app consists of five different main screens or views; where by each of them is supplemented by other views. The first view is the one holding managing of calls, herein referred to as the Simu View. The second view is called Ujumbe View, which manages messages. The third view deals with listening to music (Muziki View), the fourth deals with calculations (Kikakatuo View) and the fifth deals with web news (Habari View). These five views represent the core of the interface and are always available to the user, by performing swipe gesture between them in a manner consistent with many touch screen phones interfaces. Detailed information about different views mentioned above will be provided in the rest of this chapter.

3.2.4.2 Conceptual Overview

Feedback from the interview showed that most of the participants were concerned about being unable to find objects on touch screens or accidentally activating incorrect programs. Therefore, looking at the app interface, most views are considered as screen objects that have plenty of space for interaction. The user can tap anywhere on the screen with any finger. We adopted this technique from [30]. With this, the user does not need to accurately tap on objects. Thus, it is important that touch screen interfaces are easy to explore and minimize the need to search for on-screen items through trial and error.

Flicks, single taps and long taps are the most common events found in touch screens. Flicks for example can be used for navigation and performing shortcuts. Single taps are used more often in selecting objects or activating an event on the screen. Long taps can be used to activate a sub menu which provides more actions on a chosen object. Therefore, for browsing or navigation between views, another technique called a swipe gesture (swipe towards the left, swipe towards the right) was adopted. This technique was applied in most parts of the app, including navigating to other screen view objects, namely browsing the phone book, browsing between different text messages, browsing between songs, and browsing between different news titles.

Furthermore, while navigating through menus, the user will be informed of what is displayed and possible action to take through audio output using the eSpeak synthesizer. The auditory feedback is used in all areas which involves notification and verification from the app. In addition, the eSpeak synthesizer is used to read text messages and web news content.

All participants know how to read and write Braille code. Therefore, Braille text-entry design is used for text-entry. The designs are adopted from [34-37], with a few modifications related to language context. Grade 1 Braille code was selected as it was found to be familiar to the participants.

Besides using swipe gestures for selecting a contact in a phone book, voice command is used in addition. The user will use voice command to dial a number. Furthermore, for calculations, the user can use voice command to perform basic calculations.

For a quick visual summary of how all the various parts of the app interact with one another, see figure 6 below.

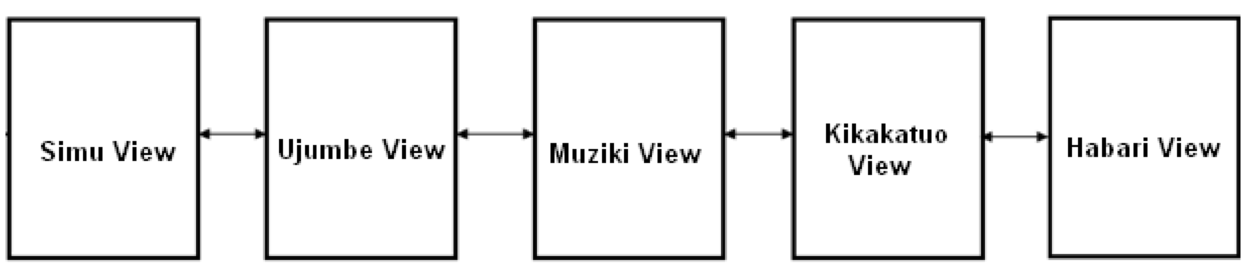

Figure 6: Five main view objects aligned horizontally that represent the baseline of the app.

3.2.4.3 Simu View

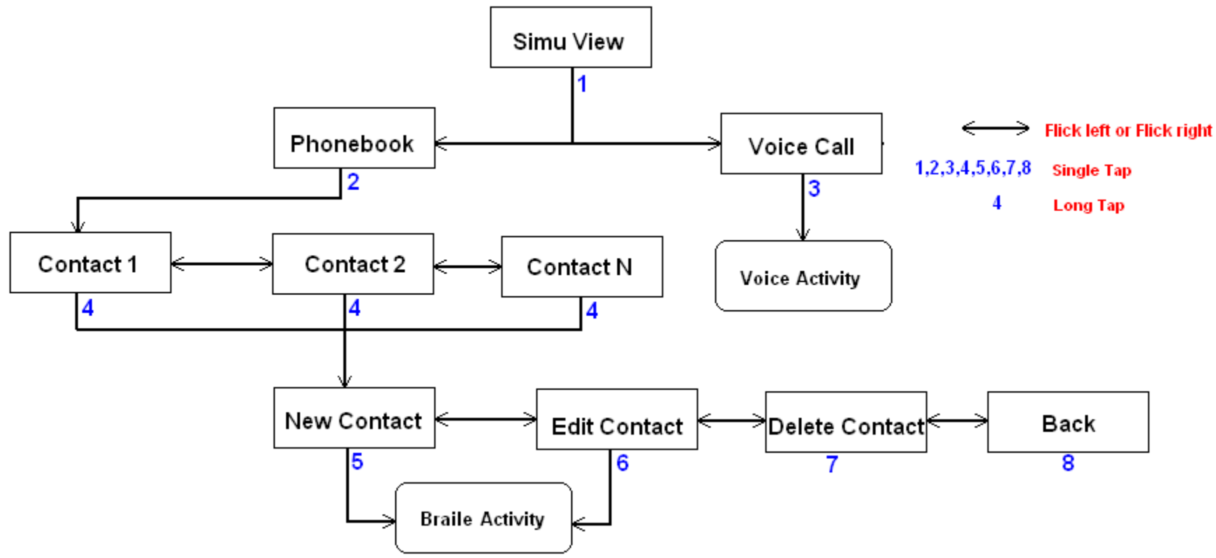

Figure 7: Different levels of hierarchy view or screen objects under the Simu View.

Each rectangle in figure 7 above represents a view or screen object in which a user can interact with. The labels have been translated to English for clarification. However, the real labels are represented in Swahili. The swipe gesture (Flick left or Flick right) is used for browsing the following:

- Choosing between Phonebook view and Voice call view.
- Browsing contacts (contact screen objects) within a Phonebook view.
- Choosing between New Contact view, Edit Contact view, Delete Contact view and Back view

With the swipe gesture performed, audio feedback will inform the user on what is displayed and what action to perform at the same time.

By tapping the screen just once, the user can activate the sub-item or sub-view. For instance, by tapping the “Simu View” (1), the user will activate the two menu views, the voice call (2) and the phonebook (3). Tapping on the phonebook, contact views (4) will be activated while tapping on the voice call, the voice activity (3) will be activated. Tapping on the specific contact view (4), which represents a contact number, will activate a call. Finally, tapping either on the new contact view (5) or edit contact view (6) will activate the Braille activity for text entry. Tapping the delete contact view (7) will remove the respective contact while tapping the back view (8) will go back to the previous view, the contact list view. Similar to the swiping gesture, the auditory feedback will be used to notify the user which views he or she is in or provide instructions on what to do next after opening a new view.

With a long tap gesture triggered for any of the contact view (4) objects, a new menu for adding a new contact, editing, deleting and back view will be activated. The long tap is also used in the voice call view to enable users to speak out voice commands and activate the call.

3.2.4.4 Ujumbe View

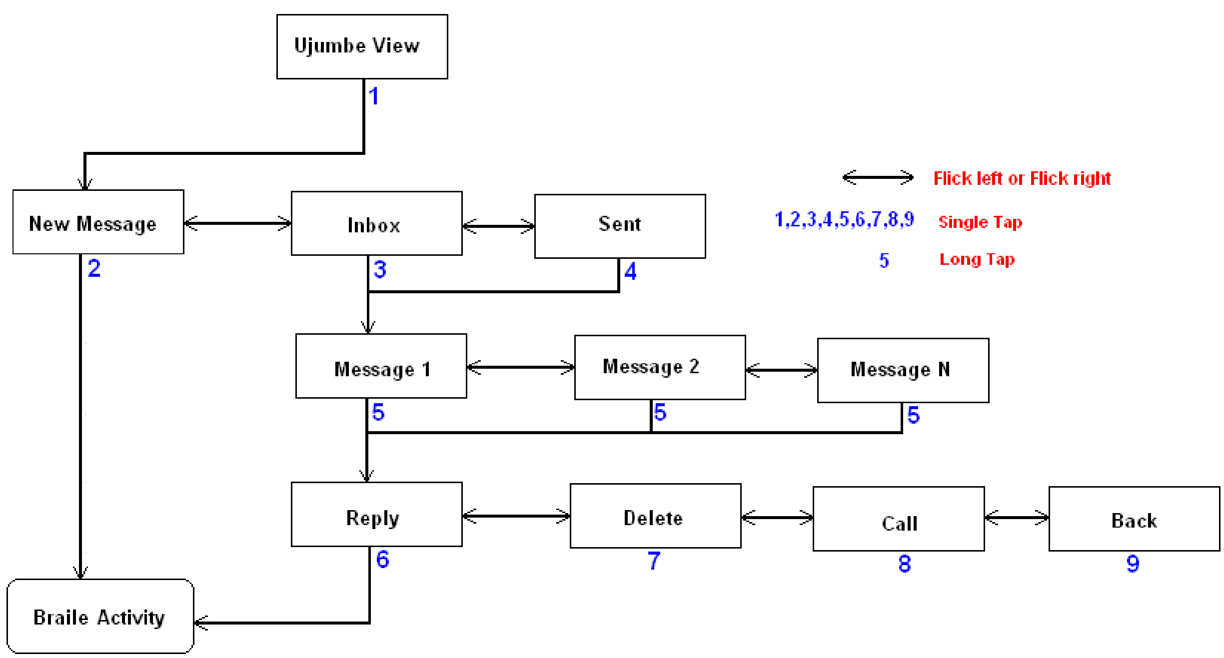

Figure 8: Different levels of hierarchy view under Ujumbe View.

The rectangles in figure 8, can be seen as views objects. The swipe gesture (Flick left or Flick right) is used for the following tasks:

- Swiping between new message view, inbox view and sent view.
- Browsing across different messages within either the inbox view or the sent view.
- Swiping among reply view, delete view, call view and back view.

As in Simu View, audio feedback will inform the user on what is displayed and what action to perform at the same time.

By tapping the screen just once, the user can activate the sub-item or sub-view. For instance by tapping the "Ujumbe View" (1), the user will activate the three menu views, the new message, inbox and sent views. Tapping on the new message (2) view object, the Braille activity view will be activated. Tapping either on the inbox (3) or the sent views (4), the message views will be activated. Tapping on the specific message view (5), which represents a message from a certain contact, will activate to read the message by voice. Finally, tapping the reply view (6) will activate the Braille activity view for text entry. Tapping the delete view (7) will remove the respective text message. Tapping the call view (8) will activate the call while tapping the back view (9) will go back to the previous view, the message list view. Again, the audio will be used to notify the user which views he or she is in or provide instructions on what to do next after opening a new view.

With a long tap of any message view (5), a new menu for replying to a message, editing, deleting, calling and back view will be activated.

3.2.4.5 Muziki View

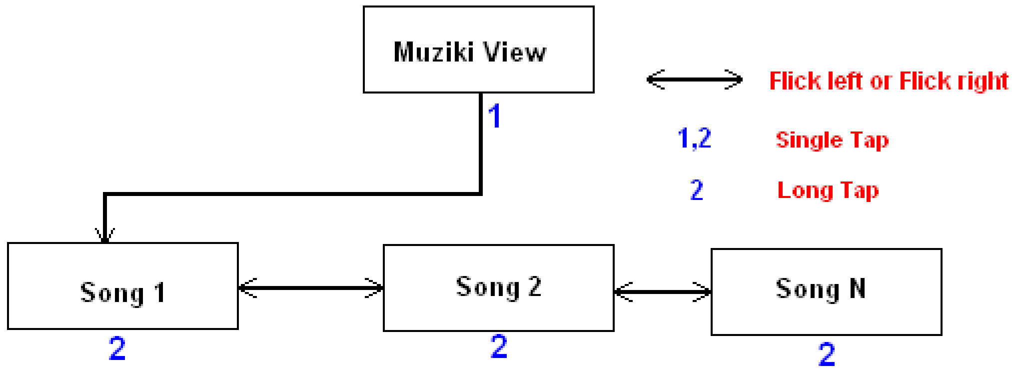

Figure 9: Different levels of hierarchy view under Muziki View

The swipe gesture in figure 9 is used only when browsing across different songs. Selecting a song and playing it is done through this gesture also. So it begins by tapping the Muziki view (1) which will open the song list view. Tapping any song view (2) will simply pause the song, and tapping again for the second time will re-pause. The long tap will provide more information about the song, namely the song name and the artist name. This information will be provided through audio output.

3.2.4.6 Kikakatuo View

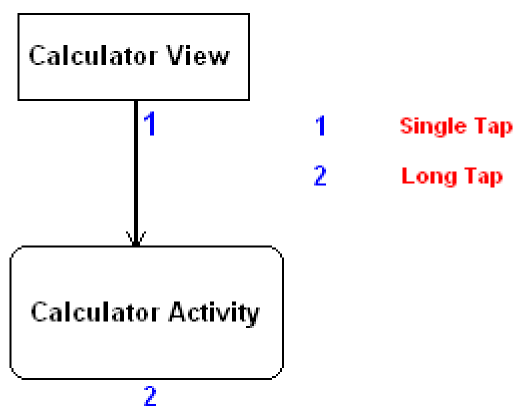

Figure 10: Different levels of hierarchy views under Kikakatuo View

In the calculator view presented in figure 10, only two gestures are involved. The single tap on the calculator view (1) will open the calculator activity view (2). Once the activity is open, instructions are provided using auditory feedback on how to operate the calculator. The long tap in the Calculator Activity view (2) is used similar to that of the voice call. However this is used to activate the calculation by choosing among the four operations: addition, subtraction, multiplication and division.

3.2.4.7 Habari View

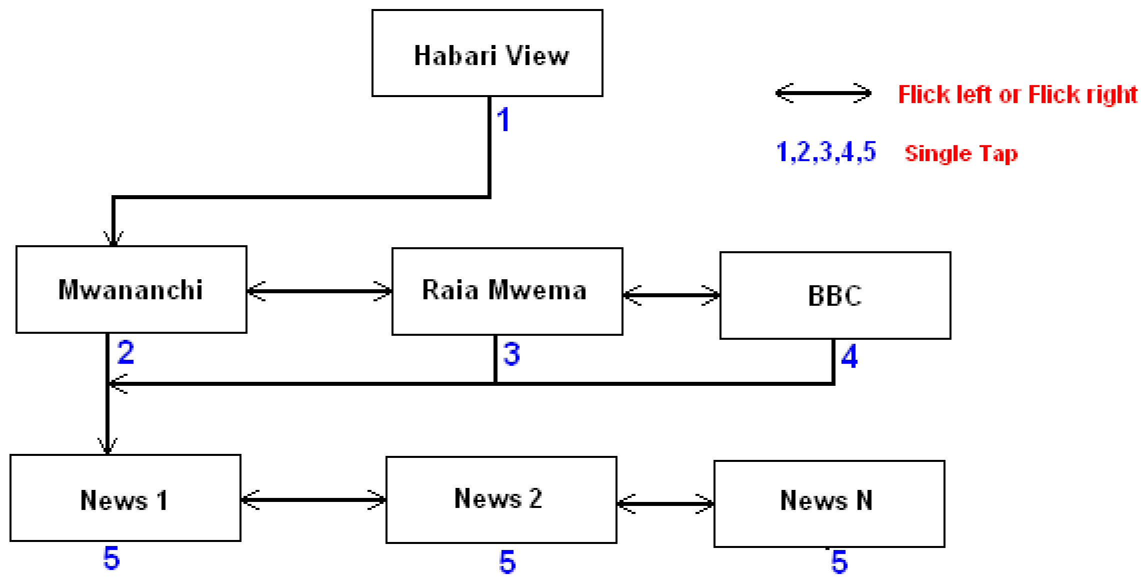

Figure 11: Different levels of hierarchy view under Habari View.

The swipe gesture in figure 11 allows the user to do the following:

- Navigating between Mwananchi view, Raia Mwema view and BBC view.
- Browsing across different web news content.

With the swiping gesture, audio feedback is used again to inform the user on what is displayed. This could be a news source such as BBC. Tapping the screen just once will activate the sub item or sub view. For instance tapping the “Habari View” (1), the user will activate the source news menu views, Mwananchi, Raia Mwema and BBC. Tapping on any of the news sources, be it Mwananchi (2), Raia Mwema (3) or BBC (4) will activate the list of news headlines views(5), while tapping once more on each of them will activate to read the news content.

3.3 Iteration Two

The second iteration began with evaluating the previous iteration. The evaluation result which was based on users' feedback helped us to redesign the app and make changes accordingly. The DECIDE framework described by Rogers [44, 48] was adopted to guide the evaluation. Evaluation techniques and paradigms were chosen on the basis of the goal for the intended iteration.

The following section describes each part of the DECIDE framework in more details relating to the evaluation of the app in the second iteration.

3.3.1 Goals

In this iteration, the goal of the evaluation was more specific on whether the design decisions made for the implemented interface were really in the right direction in terms of assistance and thus they required no alteration. If they did need alterations then changes were made and a similar evaluation was conducted in the next iteration. Evaluating these design decisions was very important not only for the sake of minimizing the cost of redesigning the whole app but also for the sake of the app's success during the final evaluation phase.

3.3.2 Questions

Several questions were chosen in order to verify the design decisions. The following were the main ones:

- Whether or not the synthesizer voice was understandable
- Whether or not the synthesizer speed was too fast or too slow
- Whether or not the auditory feedback or the audio output was understandable in different parts of the app
- Whether or not positioning of the back button was right and the button size was big enough

Besides the above questions, users’ views, opinions and suggestions were also taken into account.

3.3.3 Evaluation method

As the session was not formal, quick and dirty evaluation paradigms were used to get quick feedback from the users. Users were observed while using the prototype and key points were noted relating to the usability. When the observation approach failed to provide enough insights, open questions were asked focusing on users’ opinions. In short, a think-aloud technique was used to get more understanding of the user’s experience while using the app at the early stage.

3.3.4 Practical issues

Five (5) visually impaired people out of the original twenty (20) participants were chosen for participation during the evaluation phase. The sessions were conducted in one of the office rooms at the Tanzania League for the Blind (TLB) premises. The tests lasted about 50 minutes.

For the entire study, Samsung Galaxy S3 GTI9300, with a 4.8" 720x1280 WXGA screen was used. This phone runs Android version 4.0.

3.3.5 Evaluation

Since at this point the evaluation session was not a formal one, no proper documentation was required. Rough notes were written concerning users’ observations, opinions and suggestions and thereafter were considered in the next iteration phase.

3.3.5.1 User issues and Comments

The following is a list of users’ issues and comments gathered during the evaluation phase:

- Too much auditory information

It seemed there was too much information at the beginning of each screen window that appeared, wasting user’s time. One participant said, “The app does not stop talking. It confuses, not knowing what to do and when“. A suggestion was made to shorten these instructions.

- Button location consistency

  It appeared the Back button was inconsistently placed among different views. This caused problems to the users. It was suggested that consistency be followed so as to have easy recall.

- Speech synthesizer speed

  It appeared to the majority of the users that the speed of the synthesizer was fast. They suggested that it should be reduced.

- Small buttons in Braille-method interface

  In the Braille-method interface, the buttons which represent the six Braille code cells were too small for some users for interaction. Users who had long fingers were tapping on unresponsive areas of the interface. So it was suggested that the size of the buttons be increased in order to accommodate users who have long fingers.

- Accidental tapping

  It appeared that the users were pressing the android back or menu button accidentally. A suggestion was made to disable them in order to prevent accidental errors.

- Delete feedback

  In the Braille-method interface, users suggested that, when deleting letters, there should be a notification of what letter is being deleted and whether everything has been deleted or not.

- Ambiguity in Braille-method interface

  It was suggested that when writing the in the Perking machine, go forward or move backward does not apply, so these actions should be remove to avoid ambiguity with the space or delete function.

3.3.5.2 Discussion

Overall, users' reactions to the interface were positive. An encouraging sign was found in the fact that none of the users struggled to understand the synthesized voice or instructions presented; they said it was better than what they are used to because it pronounced Swahili words better. However, there seemed to be general agreement that the voice speed should be reduced. This issue was tackled in this iteration. To increase the size of buttons for Braille-method was another issue that several users agreed on, and was thus resolved in this iteration.

The position for back button was okay. However, in parts of the views, it was inconsistently placed in different positions. This confused users most of the time. Therefore, it was agreed that for all the views, the back button should be consistently placed at the bottom. To avoid the accidental errors, users also agreed to disable both the Android menu button and the back button. Hence only the screen back button will be used. Lastly, there was a word correction from "Kikakatuo" to "Kikokotoa" (Calculator) which was pointed out by a user.

In the Braille-method interface, it was a feasible idea to users that, there should be a voice notification when deleting letters. Users wanted to track and verify what they have deleted using the notification even though there was a button already for reading what was typed. In addition, it was agreed that the forward action and the backward action be removed because they do not exist in the Brailler-machine. Only "space" and "delete" action should remain.

It should also be noted that after having users interact with the app, other issues that they had not commented on became apparent. The first was that of notifying users when they reached the end of sliding the views. It was observed that users keep on repeating the slide gesture in the same direction, believing that the action has not been done correctly.

The second was that of making the view full screen in order to prevent users triggering the Android mini system setting accidentally. It was observed many times especially in the Braille-method interface view where user's fingers would accidentally perform a short slide gesture from the top of the screen when either holding the phone or during typing. Making the view full screen is the strategy which is found in some of the system apps like the camera view.

The third was that of notifying users to wait for some time during news content loading. On observing the users for a few seconds after activating the news content window, it was noted that they would continue tapping the window thinking they have not yet activated the action for reading. In reality, the content was being loaded from the internet which took time to do so. This is bad design, as users were kept in the dark as no notification was presented.

3.3.5.3 Changes To the design

Based on users' feedback and suggestions, changes were implemented to the user interface design. Only observable changes will be highlighted with figures, while the rest will be changed as discussed above but not highlighted here.

To start with, all major five view objects were affected by the notification view object placed both at the beginning and at the end of each view. Figure 12 shows the addition of the "Start View" screen object before the "Simu View" which appears first and the "End View" screen object which appears last after the "Habari View".

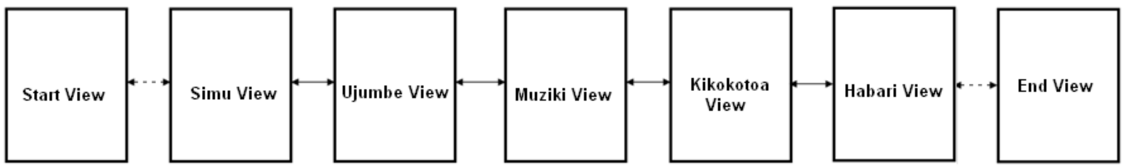

Figure 12: Main view objects aligned horizontally with additional "Start View" and "End View" screen objects.

Looking at figure 13, figure 14, figure 15 and figure 16 both the "Start View" and the "End View" screen objects were also added in each hierarchy level. Note that, the back button in figure 13 after the "Delete contact view" screen object and in figure 14 after the "Call View" was removed for consistency with other view objects. Therefore, the back buttons were positioned at the bottom (see figure 17) in all view objects except for the five (5) main view screen objects.

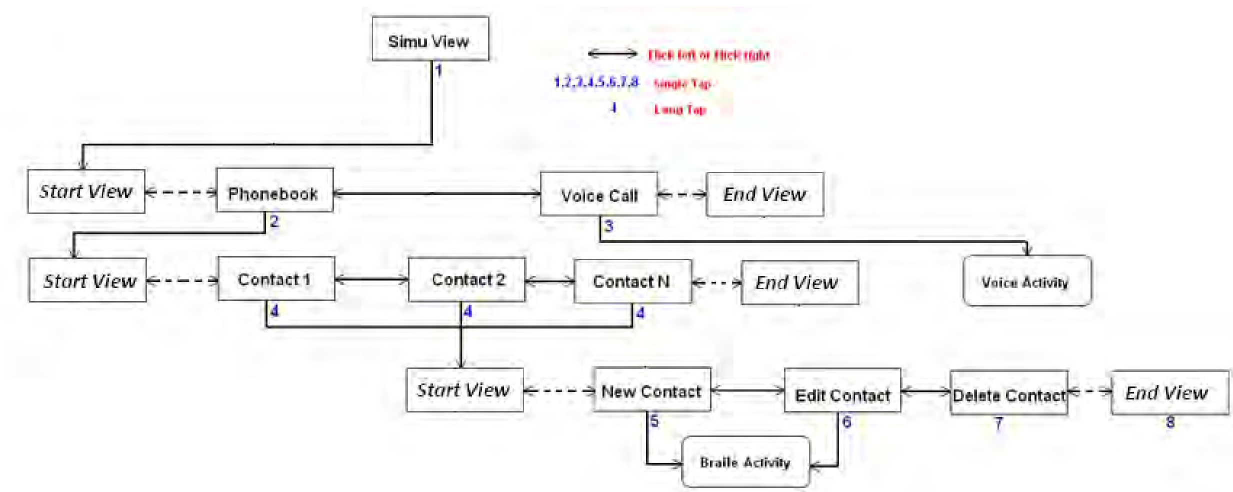

Figure 13: Additional "Start View" and "End View" screen objects in different hierarchy levels for Simu View

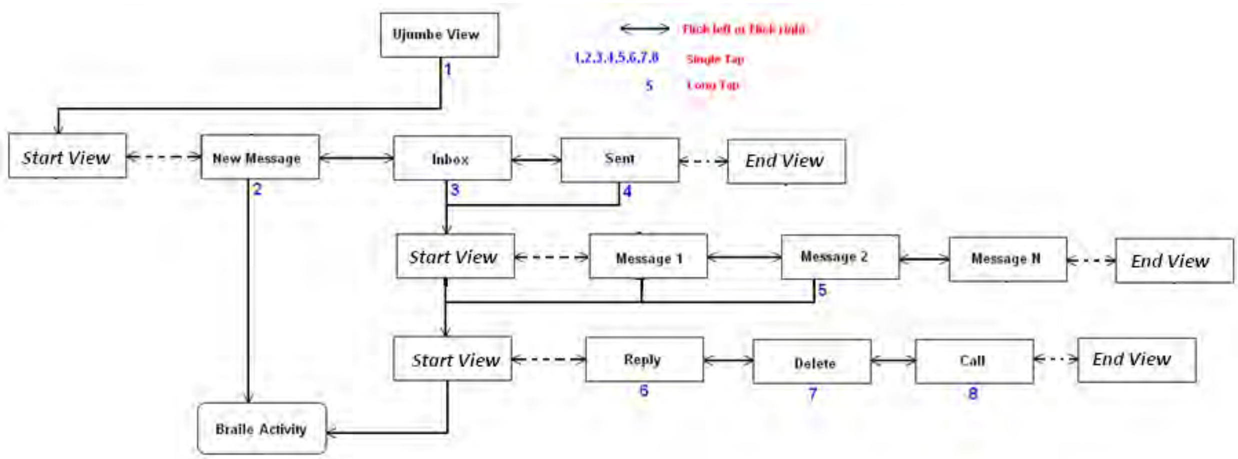

Figure 14: Additional "Start View" and "End View" screen objects in different hierarchy levels for Ujumbe View

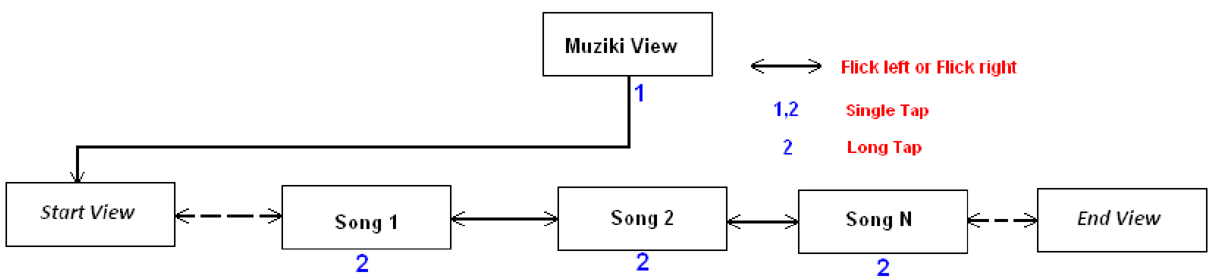

Figure 15: Additional "Start View" and "End View" screen objects in different hierarchy levels for Muziki View

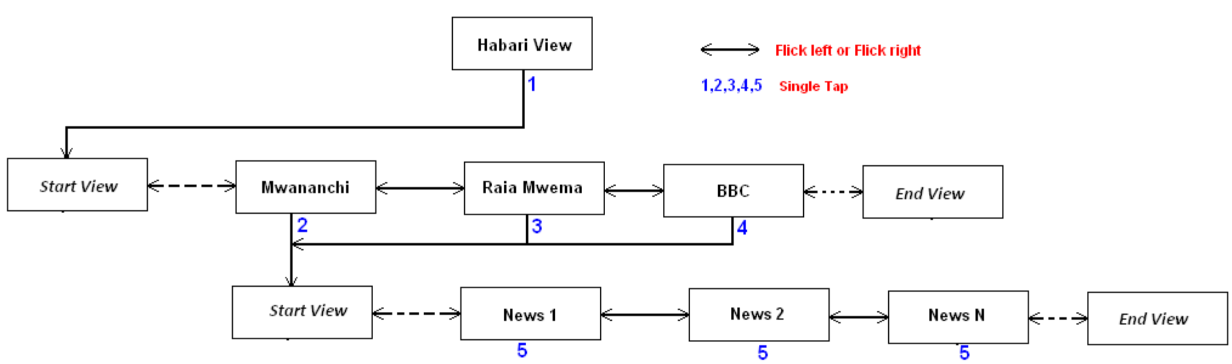

Figure 16: Additional "Start View" and "End View" screen objects in different hierarchy levels for Habari View

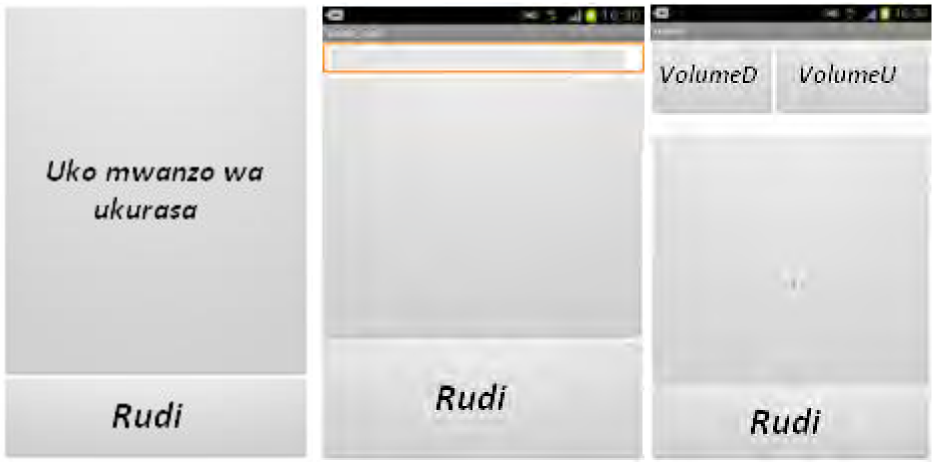

Figure 17: Different view objects, with the back button (label as "Rudi") positioned at the bottom.

The other observable change was inside the Braille-activity view object. Looking at figure 18, the buttons representing the Braille code cell were increased, which decreased the unused area zone. The view was made into full screen view which hides the view title and the phone status bar. Transforming the views into full screen was also made to the rest of the views in the app.

Figure 18: Braille-method interface transformation in design between Iteration 1 and Iteration 2.

With these design changes, the second iteration of implementation was complete. To verify the changes made overall were met with users' approval, a final evaluation was performed. More details will be provided in the next section concerning the final iteration design phase.

3.4 Iteration Three

As was done for the previous iteration, the DECIDE framework was used once again to guide the evaluation. The individual points outlined by DECIDE in this iteration were similar to the previous iteration. The goals remained the same as for the second iteration. Similar questions were asked in this iteration but focusing on the new changes made. The think aloud method was used.

The only exception was the number of participants. Another five participants were invited in the evaluation. The goal again was to verify the new design with new participants before we moved to the final evaluation stage.

3.4.1 User Issues and comments

The following is a list of users' issues and comments gathered during the evaluation phase:

- Ambiguity in audio instructions.
  One participant suggested that the instruction "Upo mwanza wa ukurasa" (You are at the beginning of the page) was ambigious. He thought that this view indicated that he was in the area of composing a message. He suggested that, the instruction be changed to "Uko mwanzo wa program" (You are at the beginning of the program), au "uko mwanzo wa mwongozo mkuu wa simu (You are at the beginning of the main phone's menu) for better understanding.
- Avoiding gesture errors.
  The single click was not accepted by the majority of the participants. It gave them problems by accidentally triggering the event without their intention. Double click was suggested for accepting a choice. Single click was suggested

to be used to provide instructions on what to do at any particular screen object instead.

- Unconscious usage of slide gesture direction

  It was observed that participants always perform a sliding gesture from left towards right unconsciously despite the instruction stating otherwise. According to most of the participants, the main reason was that they are used to read or write Braille code from left to right and hence it became automatic for them to do so.

- Pausing feedback

  One participant suggested the app should notify if it has paused the music.

- Unclear instructions

  Another participant suggested that the instruction, “Uko kwenye sehemu ya kutuma ujumbe” (You are at the sending text message view), should be changed to “Uko kwenye sehemu ya kuandika ujumbe” (You are at the writing text message view).

- Recall issue

  It was observed that participants sometimes forgot what view they were in and what to do next. One participant commented about the instructions not being clear enough to understand what to do exactly. Another participant entered the phonebook view and was presented with instructions on what to do next. She did not understand and thus wanted to hear the instructions again. However at that point, repetition of instructions was not possible unless she started over again.

- Alternative text-entry method

  Participants suggested other means of writing text message. Some participants said they had forgotten how to write Braille code. Other participants felt uncomfortable while handling the phone when using a Braille-method. They would accidentally touch the screen incorrectly and trigger the wrong Braille code. Users suggested that, the design should allow them to align the fingers first even before pressing buttons.

- Adding Swahili Voice

  One participant commented that it would be nice to integrate Swahili voice because so far only English was seen to be popular. Another participant said that the app's voice better than those in computers when pronouncing Swahili words. However, most insisted that the app should have an option to regulate the voice speed, either to increase or decrease it.

3.4.2 Discussion

There was a mixed users' reaction towards the app design. To start, the addition of "Start View" and "End View" screen objects appeared confusing and not useful most users. As there were many conflicting suggestions on how to present the Start View and End View, we decided to remove them all, and make the list of screen objects endless, thus, making the list view objects rotate, so that as the user reaches the end, continuing forward will move to the beginning of the view object. It appeared most of the time, users were given to slide the views from right to left, but still users will unconsciously start to slide the views from left to right. Making the screen objects endless will help users to navigate the views in any direction they want.

The single tapping gesture, was not well accepted by the majority of the users, and therefore we replaced it with a double tapping gesture for accepting the choice. This decision was made after realizing a big number of gesture errors in the second iteration compared to the first iteration and no mechanism was implemented yet for recovering from these errors. Therefore, the single tapping gesture remained to be used to trigger the instructions. This strategy made sense for most of the users and prevented them from accepting a selection accidentally by just touching the screen. As it was observed by the majority of users that they forget or do not know what to do when they enter normally into a view object, using a single tapping gesture to trigger instructions would help to remind them what to do.

There were small changes in the Music view object, where we agreed that the app should notify the user when pausing the song. Another change made was the instruction “Uko kwenye sehemu ya kutuma ujumbe” (You are at the sending text message view), which was changed to “uko kwenye sehemu ya kuandika ujumbe” (You are at the writing text message view).

As some of users struggled with the Braille-method, we decided to design another text-entry method called the Sliding method. We also reached the conclusion that addition view for dialing a number to be added. More details about the new text-entry method are presented in the next section.

3.4.3 Changes To the design

As for iteration 2, only observable changes which affect the design will be highlighted here. Other changes will be updated as per discussion in the previous section but not highlighted. However, the design for the new sliding method will be presented on its own separate sub section.

The change was made by removing the “Start View” and the “End View” objects and making the remaining view screen objects rotate as the user moves between them. This changes the design which can be observed in the figure 19, figure 20, figure 21, figure 22 and figure 23.

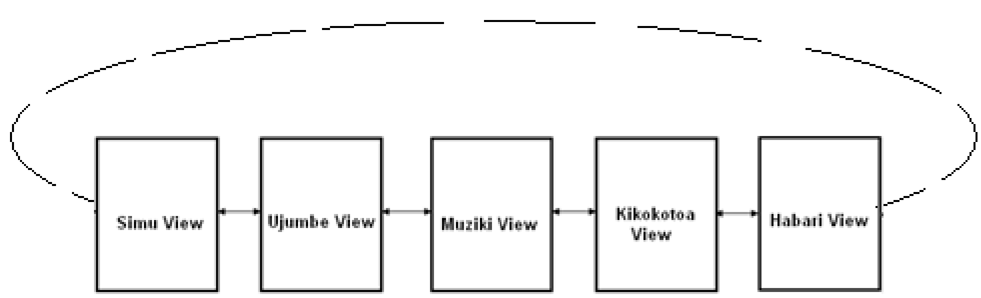

Figure 19: Main view objects aligned horizontally in a circular fashion.

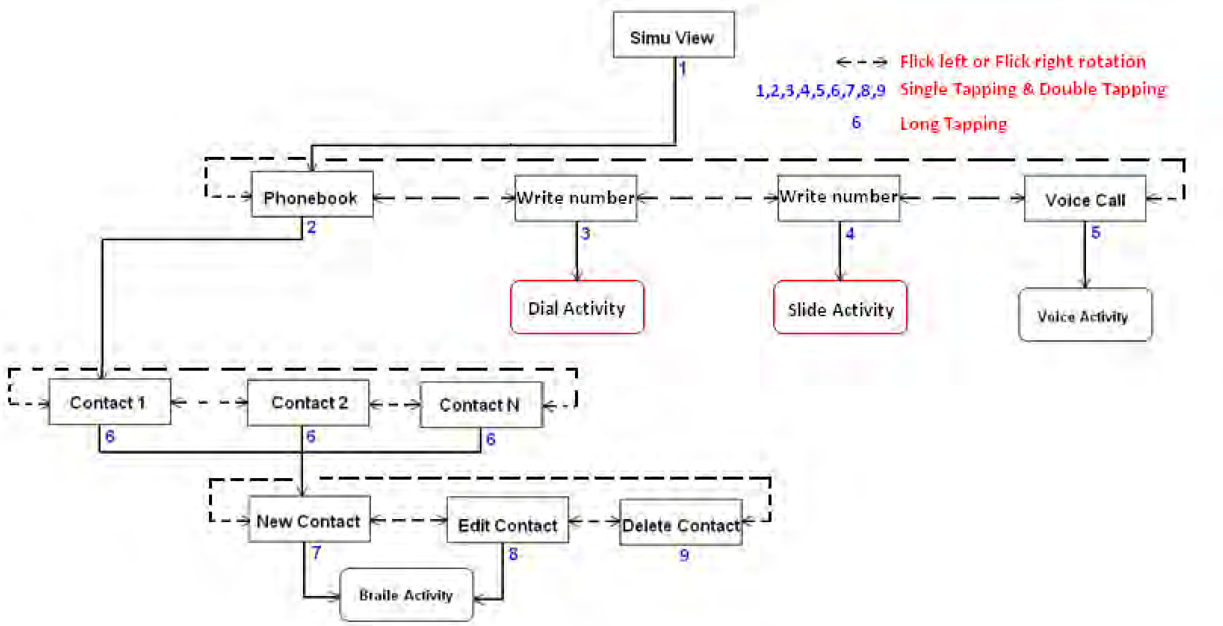

Figure 20: Different levels of hierarchy views in Simu View with an addition of Dial Activity view and Slide Activity view.

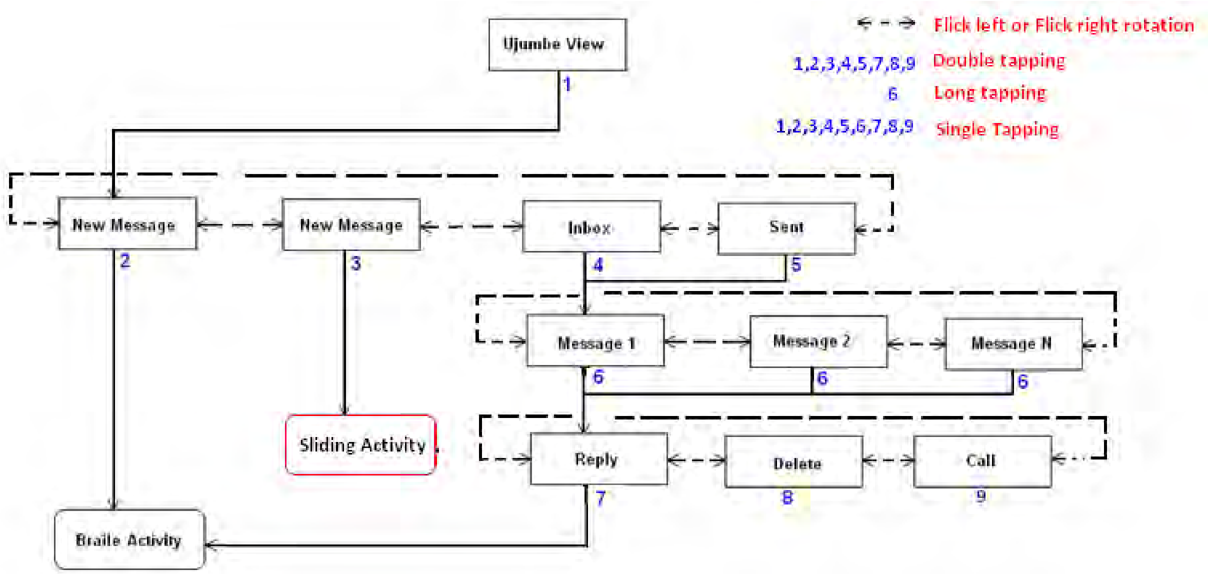

Figure 21: Different levels of hierarchy views in Ujumbe View with an addition of Sliding Activity view.

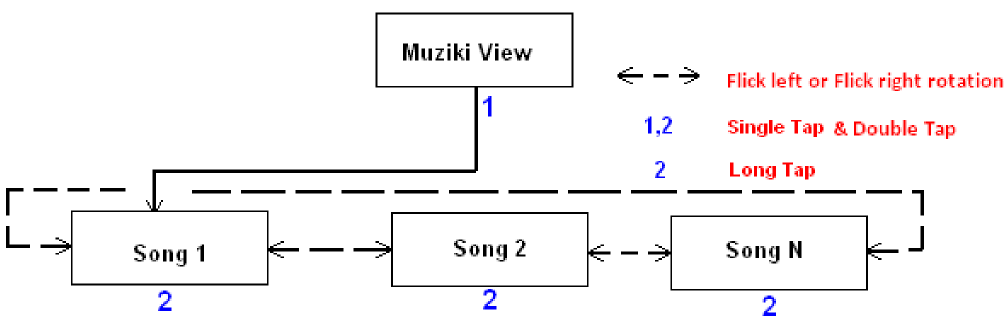

Figure 22: Different levels of hierarchy views in Muziki view which are aligned in a circular fashion.

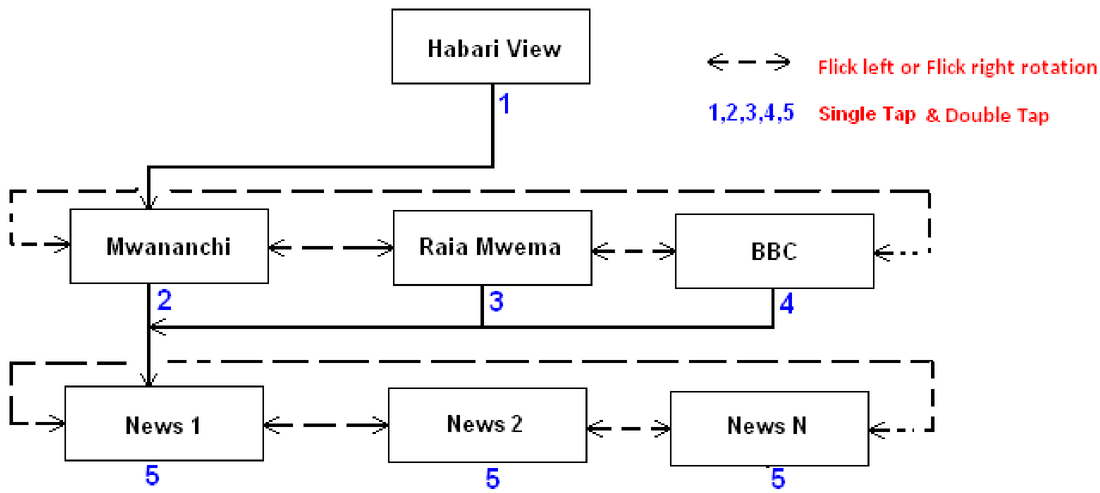

Figure 23: Different levels of hierarchy views in Habari view which are aligned in a circular fashion.

For our new view screen, objects can be observed in figure 20 and figure 21. In figure 20, the Dial Activity view object and the Slide Activity view object were added as part of the design. The Dial Activity design was designed from the idea of the 12-key telephone keypad (for Numbers only), with a small twist. Figure 24, highlights the difference. The star symbol and the hash symbol were removed with an assumption that they were never required. They were merged into a single giant zero button. Delete button was suggested by users to be positioned at the top right corner and the call button to be positioned at the top left corner of the screen. The decision for putting these buttons in that position was adopted from the normal phone design as the majority of the participants found it easy to access them. The back button was positioned at the bottom as usual for consistency.

Figure 24: 12-key telephone keypad implemented in Dial Activity view.

In figure 25, we highlight how the Slide-method for number-entry was implemented. The top buttons and the bottom button are the same as in the dial activity for the same reason. The middle button (labelled "Swipe") is for performing swipe gestures. Users can swipe either up or down for selecting between the decimal numbers from 0-9. In which after selecting the number, the user will perform a double tapping gesture on the same middle button to accept writing the number down.

Figure 25: The Slide-method for number-entry

Lastly, figure 26 presents the final design for the slide-method for text-entry, the sliding Activity. The button layouts are the same as in the slide-method for number-entry with an additional swipe gesture. The steps for writing numbers remain the same, swiping up or down for number selection. However, now, for writing letters of the alphabet after number selection, swiping left or right will select letters (A-Z).

Figure 26: The Slide-method for text-entry

3.5 Summary

After going through three separate iterations, each of which used a slightly different approach to articulating specifications and implementing these as designs, a comprehensive app has been designed. The design itself is a multiple screened application that allows users to navigate left and right between screens using a swipe gesture. Overall, the application has five major screen view objects. The first view, the Simu View, enables users to manage contacts and make calls. The user can make calls using the following ways: making calls through phonebook, making calls by dialing numbers, making calls using the Slide method and making calls using the voice call. The second view, the Ujumbe View, enables users to manage text messages. Here, the user can read inbox messages or sent messages. Users can either write text messages using the Braille-method or the slide-method. The third view, the Muziki View, enables users to simply listen to music. The fourth view, the Kikokotoa view, enables users to perform basic calculations like addition, subtraction, multiplication and division. The last view, the Habari View, enables users to listen to local web news from Mwananchi, Raia Mwema and BBC.

In the following chapter some technical implementation details on how all of this design was put into operation as a working app will be given. Thereafter, a final evaluation will follow in chapter 5.

4 Implementation

The User interface for the mobile app was implemented iteratively for each of the design iterations. The Android platform was used to develop the app through the Eclipse Kepler service release 1 IDE. Although the IDE has a built in simulator for testing the app, it was not used because it did not support the speech recognition module. Instead, the app was run directly on the mobile device and debugging was carried out there. With the help of BitBucket website and Eclipse, version control was achieved in which dated versions of the app were kept as well as the backup code.

The mobile app was implemented through the model-view-controller design pattern. This strategy helps to develop the app from the small functional modules and finally combine them into a complete app. Integration of pocketsphinx speech recognizer and the eSpeak synthesizer into the Android platform was considered before development of the app commenced. The biggest challenge in the app was envisioned to be whether or not the app can speak out the written Swahili text and it can also recognize Swahili words. In the next sections, details about different main parts (see figure 27 for pre-existing and figure 28 for newly developed components) of the app which were then combined to complete the app are described.

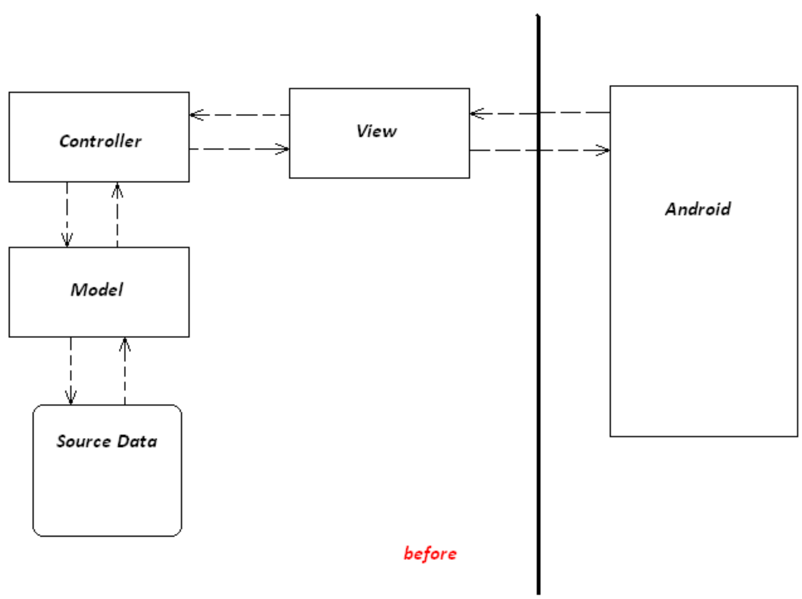

Figure 27: Pre-existing application components

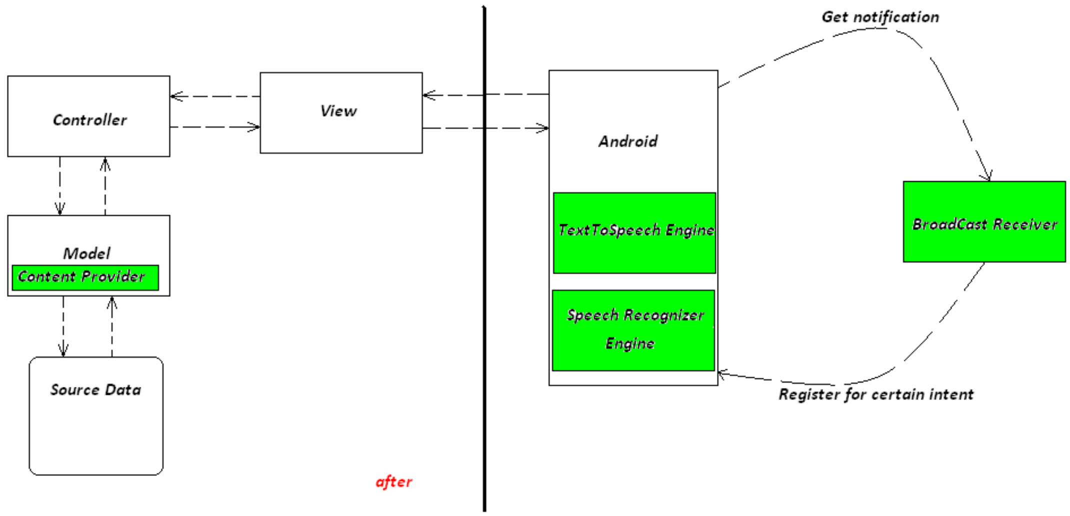

Figure 28: Additional developed application components

4.1 Database (Source Data)

Since contacts, SMS and audio SQLite databases were already created when the Android OS was deployed into the device, the major task was to manage the data and display them. The content provider facilitated this exercise by managing accessibility to this structured set of data. Normally the content provider presents data to external applications as one or more tables that are similar to the tables found in relational database. In order for an application to access data from a content provider, a ContentResolver client object is used. The ContentResolver methods provide the basic CRUD functions of persistent storage. The Uniform Resource Identifier (URI), maps the table in the provider name to a table name. In other words, it identifies data in the provider. Therefore, to retrieve data, the provider will construct a query (ContentResolver.query()) with a specified URI (see table 1) as one of the parameters. The Cursor is returned containing columns specified by the query's projection for the rows that match the query's selection criteria. Iterating over the rows in the results, these were then populated into the respective data source classes which encapsulate each of the data. In the end, each encapsulated data is displayed across fragments.

To modify data from the provider, interaction between a provider client and the provider's ContentProvider was required. To insert data, a ContentResolver.insert()

method was used. For updating, a client method, ContentResolver.update() was used. While for deleting data, ContentResolver.delete() was used. All CRUD operations are only performed in the contact database. For SMS database, only create, read and delete operations were performed. While in the audio database, only read operation was performed. The control was done mainly in their respective Activity classes.

Table 1: Source data with their respective controllers and the model source (URI).

Source Data	Source Data Class	Activity Class (controller)	URI (model source)
Contacts	ContactBean	Nambasimu, AddedSimu, SimuManage	ContactsContrast.Contacts.CONTENT_URI
SMS	ContactSms	listiujumbe, andikaujumbe, SmsManage	Contet://sms/inbox or Content://sms/sent
Audio	ZikiClass	miziki	MediaStore.Audio.Media.EXTERNAL_CONTENT_URI

4.2 ViewPager

Most of the work and all the visuals in the app are controlled by different ViewPagers. The ViewPager is the widget that allows users to use the swipe gesture to browse between screen objects. ViewPager incorporates both Fragments and PageAdapters. In this case, Fragments are the view screens objects while PageAdapter can be regarded as a supplier which supplies fragments to the UI for drawing. Each view that the ViewPager allows the user to scroll to is really a Fragment. This can be any collection of views.

The main class in the application is the MenuActivity. This is the class which contains the important onCreate method which is always the first to be called when an Android application is run. The following are the processes that happen when this method is run:

- The list of fragments containing each view(Simu View, Ujumbe View, Muziki View, Kikokotoa View, and Habari View) are created.
- The custom page adapter is created using the list of fragments.
- The pager assigns the custom pageAdapter.
- The speech recognizer is initialized.
- The TextToSpeech engine is also initialized and started.

Other important classes in the application are the NambaSimu, the listiujumbe, the miziki and the NewsFeed. NambaSimu is the class where the lists of contacts are populated into the fragments for viewing. ListiUjumbe is the class where the lists of either inbox sms or sent sms are populated into the fragments for viewing. Songs are populated into fragments embedded into the miziki class while news contents are populated into the fragments embedded inside the NewsFeed class. To avoid any inconsistency during any modification of any data, the query method was better placed within the onResume method and not inside the onCreate method. Table 2 below summarizes each ViewPager with its corresponding fragments (views).

Table 2: A list of ViewPagers (FragmentActivity) with their corresponding views (Fragments)

FragmentActivity	No of fragments	Fragments
MenuActivity	5	Simu, Ujumbe, Miziki, Kikokotoa, Habari
SimuKupiga	4	Phonebook, Write (1), Write (2), VoiceCall
NambaSimu	Number of contacts	-
SimuManage	3	New, Edit and Delete Contact
UjumbeMenu	4	New Message (1), New Message (2), Inbox, Sent
listiujumbe	Number of messages	-

SmsManage	3	Reply, Delete Sms, Call back
Miziki	Number of songs	-
NewsMenu	3	Mwananchi, Raia Mwema, BBC
NewsFeed	Number of headlines	-

4.3 Broadcast Receiver

In some parts of the application, the app is required to listen to incoming or outgoing phone calls or incoming text messages. This can be achieved by using Broadcast receiver. Broadcast receiver is an Android component that listens for a system event or application events. These events can be intercepted by other applications for their personal use. For instance, it is possible to redirect incoming calls to our application, by opening the receive call dialog. This can be seen as well when the outgoing call is made; the end call dialog is displayed instead of the system dialog. Through the app, the user is notified of any new message or incoming SMS. Furthermore, the user is given confirmation on whether the SMS has been delivered or not. Table 3 summarizes the broadcast receiver used in the application.

Table 3: Broadcast receivers with their corresponding tasks.

Receiver name class	Action	Tasks
IncomingCallReceiver	Android.intent.action.PHONE_STATE	Listening for incoming calls
OutgoingCallReceiver	Android.intent.action.NEW_OUTGOING_CALL	Listening for outgoing calls
SmsReceiver	.ndroid.provider.Telephony.SMS_RECEIVED	ning for incoming sms
deliverSMSReceiver	Com.example.idesign4.SMS_DELIVERED	Listening for sms report status
SendSMSReceiver	Com.example.idesign4.SMS_SENT_ACTION	Listening for outgoing sms status

4.4 TextToSpeech (TTS)

Most notification in this app is through voice or audio output. To facilitate this, eSpeak for Android, a port of the eSpeak Text-to-Speech synthesizer is used. The initial stage before allowing the app to speak, is to check for the presence of the TTS resources. This is normally done within the onCreate() method inside the MenuActivity class. A successful check will be marked by a CHECK_VOICE_DATA_PASS result code; otherwise downloading and installing the data will take place through firing off the ACTION_INSTALL_TTS_DATA intent. Once the android.speech.tts.TextToSpeech instance is properly initialized and configured, the app will invoke the speak() method anywhere there is a text to speak out. The TextToSpeech instance is made public and static so it can be accessed easily from anywhere in the app while avoiding initiating the TTS instance every time. Therefore, the shutdown() method is never invoked in any Activity. Furthermore, in most cases the Stop() method is used before the speak() method. This is useful where there are cases which required interrupting the previous utterance to speak out the new one.

4.5 Speech Recognizer

Speech recognizer is used for voice call and performing calculations. It all begins by importing the CMUSphinx library, using the loadLibrary() method. Both a Recognizer task instance which runs in a worker thread and the Thread in which the recognizer task runs are created inside the onCreate() method on both the VoiceCall class and Kikokotoamenu class. The recognizer task then sets a listener for any speech voice once the button is pressed and released. onResults() method is invoked when full results are generated from the recognizer. The result is then split in an array of strings. Different commands are executed depending on usage of the recognizer, whether within the VoiceCall class or Kikokotoamenu class. In the next sub section, details about the speech recognizer data source are described.

4.5.1 Speech recognition source data

The speech corpus [41] created from the web broadcast news was used as the acoustic data. We used approximately 6 hours and 20 minutes (6022) for training, and about 3 minutes (78 transcribed sentences or 199 words) as a test set of 16 kHz sample rate. The language model was created from the same 78 transcribed sentences and words with a closed dictionary of 82 words. The transcribed sentences and words are related to the voice commands to be used in the app. The WER for the test set was 19.1%. However, we thought of doing adaptation to improve the model base on the context (test data).

The acoustic model was then adapted to the test set. This gave us an improvement of 19.1% WER. This means that the recognizer was able to recognize all 199 words correctly without any errors. The adapted acoustic model is the one that was used in the app. The same words were used as a dictionary. The pronunciation file used, which is a list of words with the corresponding sequence of phones, is shown in Appendix D. A grammar language model was used instead of a statistical model as it can easily describe very simple types of languages for command and control. The grammar was written in JSGF format. The JSGF commands were based on making a call, reading SMS, playing songs, listening to the web news, and performing basic calculations (See Appendix E for the list of JSGF commands).

4.6 Interaction with the app

The four major interactions gestures the app supports are single tapping, double tapping, long tapping and swipe. Both single tap and double tap were created using MalibuCountDownTimer class which extends CountDownTimer class together with the onClickListener() handler to identify single or double event taps. The onLongClickListener() handler identifies any long tap. The Swipe gesture was implemented differently depending on the task. In ViewPagers, no implementation was required because by default it allowed the user to change the screen objects by using the swipe gesture. However, in the Slide-method view interface for both writing numbers and text messages, onTouchListener() was used to detect the motion event of the touch either horizontally or vertically. This results in four (4) types of swipe gestures:

- left-to-right swipe
- right-to-left swipe
- up-to-down swipe
- down-to-up swipe

Lastly, the onTouchListener is used again in detecting multiple tap events for the Braille-code method. This listener detects multiple touches from any of the six (6) Braille code buttons in order to execute letters or numbers.

5 Evaluation

5.1 Introduction

This chapter explains the evaluation of the app and discusses the results. Referring to the research questions in chapter 1, the evaluation revolved around the following goals:

- It should find whether or not the app can solve the current challenges faced by the visually impaired when using mobile phones. Therefore, the metric used was the task completion rate. This was able to determine how much effort it took for users to interact with the app. In addition, a comparison was made between the two text-entry methods based on performance, and types of errors encountered in completing the tasks were identified.

- It should determine usability level. A formal System Usability Scale (SUS) questionnaire [51-53] and formal interview were used to obtain the results.

Usability testing as an evaluation paradigm was adopted in this final evaluation phase. 10 users, all visually impaired, from ages of 20-53, were involved during this phase. 80% were male while 20% were female. Only two (2) users came from the previous design iterations. The remaining users were new. To deal with ethical issues, consent forms (see appendix C) were provided to the participants together with procedures of the tests which were explained to them. Finally, during the testing, the user would attempt to complete the following 10 tasks:

1. Asha ni rafiki yako na unataka kumpigia simu. Utafanyaje kwa kutumia orodha? (Imagine Asha is your friend and you want to call her. How will you do it using the phonebook?)
2. Fikiria simu inaita, utafanyaje kama unataka kuzungumza. (Imagine the phone is ringing, how will do it if you want to accept the call?)
3. Utafanyaje ukitaka kupiga namba yako kwa njia ya kuandika namba. (How will you do it if you want to call your number using dial number method?)

4. Utafanyaje ukitaka kupiga namba yako kwa njia ya Kuslaaidi. (How will you do it if you want to call your number using sliding method?)
5. Utafanyaje ukitaka kupiga namba yako kwa njia ya Kutamka namba. (How will you do it if you want to call your number by pronouncing the digits?)
6. Fikiria unataka kusikiliza muziki, utafanyaje? (Imagine you want to listen to music, how will you do it?)
7. Fikiria unataka kuandika ujumbe kwa kutumia njia ya Maandishi ya nukta nundu, utafanyaje? (Imagine you want to write a text message using Braille code method, how will you do it?)
8. Fikiria unataka kuandika ujumbe kwa kutumia njia ya KuSlaaidi, utafanyaje? (Imagine you want to write a text message using sliding method, how will you do it?)
9. Fikiria umepokea ujumbe kutoka kwa Asha na unataka kuusoma, utafanyaje? (Imagine you have received a text message from Asha and you want to read it, how will you do it?)
10. Utafanyaje ukitaka kupiga namba yako kwa njia ya Kutamka majina. (How will you do it if you want to call your number by pronouncing the contact name?)

These tasks were chosen reflecting the tasks which would be performed in reality.

In section 5.2, detailed procedures will be explained describing how the evaluation took place. Thereafter results will be presented in section 5.3 and a discussion will follow in section 5.4.

5.2 Procedure

Upon meeting with each user, an introduction was given to them together with a consent form explaining what the evaluation was all about. If there was anything unclear on the consent form, then it was repeatedly explained till it was clear. It was emphasized to the users that the evaluation was about the app and not them.

Since all of the participants were using the touch screen mobile phones for the first time, they were given a quick tutorial about how to interact with these types of phones. Specific areas to interact with the application on the phone were pointed out and demonstrated, and different ways to interact were also explained. Finally, the participants were trained on the four interactions gestures (Single-tap, double-tap, long-tap and the left-right sliding) before starting the evaluation.

After the participants seemed comfortable performing the four gestures, they were then introduced to the app features. At this point they were assigned a task set to start with. From here the participants were given a tour of the app they would be using, illuminating points of interest and important features. The tour remained fairly constant, although never formally scripted.

After this the evaluator read the tasks out to the user, who then attempted to complete them on the mobile phone device (See figure 29). Three attempts for each task were performed to compensate for the learning effect. However, task 5 and task 10 required five (5) attempts because the learning effect took much longer. Since the nature of mobile tasks dictates easily repeatable, simple interactions, it was not felt that recording the tasks on video was important. Instead, the entire conversation was audio recorded and every interaction made by the user was noted down by the evaluator on paper. The only interaction between the evaluator and the user throughout this stage of the test occurred when the user was unable to complete a task. Any questions/points for clarification brought up by the user were also noted down. An important point to note at this stage was that, the user was not allowed to be assisted in any way till unable completely to complete a task and even then, no direct assistance was provided, just clues from the earlier tutorial session. The intention of doing this was to try to evaluate the app to see whether it could be used independently in the early stage without any assistance.

Figure 29: Participants during the evaluation session performing different tasks

After both sets of tasks were completed, the participants would then be handed a SUS questionnaire (appendix A) and asked to complete it. Lastly, thoughts on the app would be taken. It was at this point that any suggestions for future improvements would be taken. The tests lasted on average about 3 hours per user for five days.

5.3 Results

As stated above, there are two parts to the evaluation - the SUS to test for the design's overall success and user testing, assess task completion rate and types of errors. First, user testing will be reported followed by the SUS results. Thereafter, both results will be discussed at the end.

5.3.1 Effectiveness

Effectiveness in this study was measured by how many tasks the user was able to complete. Figure 30 shows the number of users who were able to complete a particular task after all three trials. Note that for task five (5) and task (10), the last three trials out of five were considered during the evaluation. Most of them were able to complete all tasks at least once in all three trials. Two users were observed to complete all tasks without any assistance for all the trials except task 7 which for they required assistance in the first trial only. The majority of the participants required assistance for most of the tasks during the first trial but were able to complete subsequent trials independently. Tasks 5, 7 and 8 were experienced to be difficult by 2 users, 1 user and 3 users, respectively. These users were unable to complete the tasks even after getting assistance in all 3 trials. With a Cochran's Q test (Q=20.348, df=9, asymp sg=0.016) it was concluded that there was a statistically significant success difference among the 10 tasks which were performed by the users.

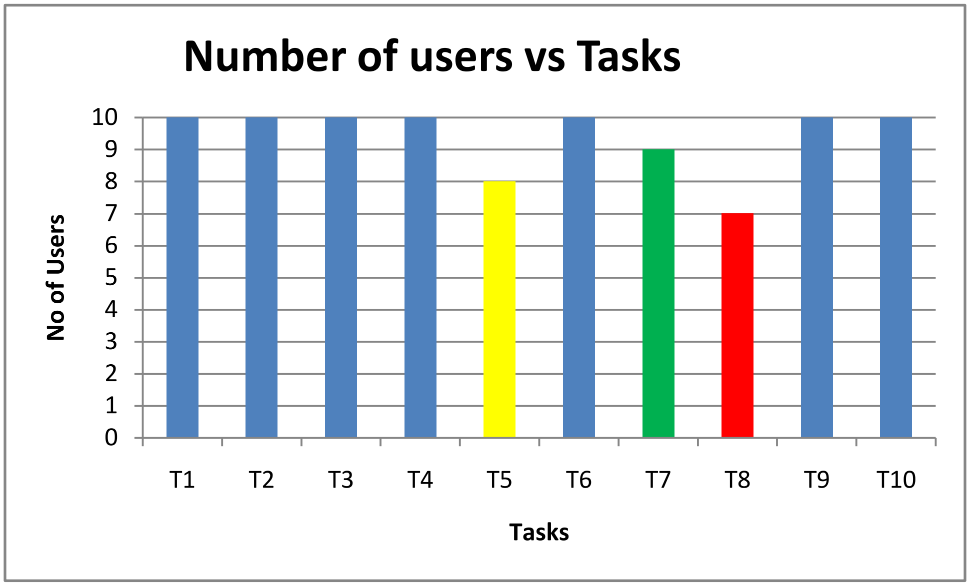

Figure 30: A bar graph for different number of users who completed the tasks.

Detailed results for task 5 and task 10 can be seen in figures 31 and figure 32 respectively. The mean accuracy for recognizing numbers (figure 31) is 1.5 after 5 trials while that of recognizing names (figure 32) after 5 trials is 4.2. The failures in figure 31 and figure 32 are due to both miss-recognition of either names or numbers and interface errors. Note that results started to be counted after assistance was provided. Thus, the first trial was meant for learning and then usability testing commenced after that.

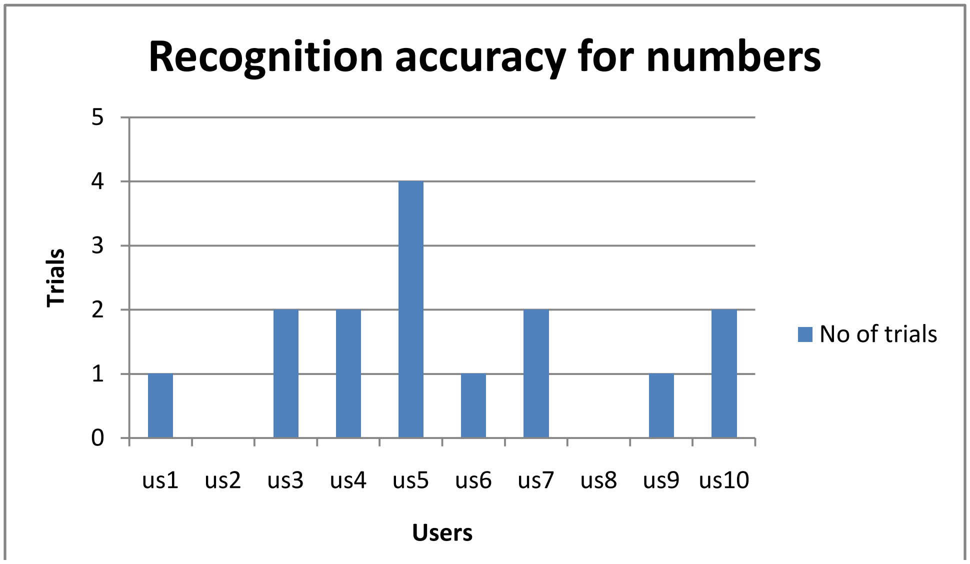

Figure 31: The number of times the app was able to recognize phone numbers during voice call per user.

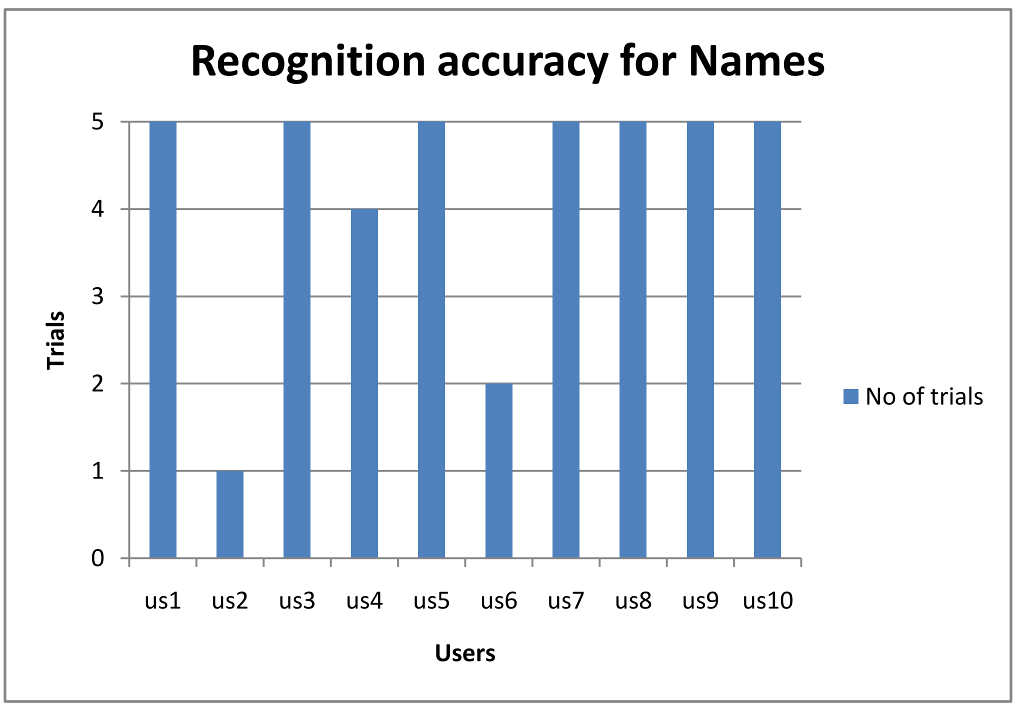

Figure 32: The number times the app was able to recognize names during voice call per user

5.3.2 Text-entry methods comparison

Comparison of the two text-entry methods was made based on efficiency. Efficiency here means how much time it takes to complete a task correctly. If the user failed to complete the task, the time spent is not counted in the calculation. Eighty percent (80%) completion rate on both methods was obtained in the last trial which was considered for comparison.

As indicated in figure 33, users were able to write faster using the Braille text-entry (Mean 90.9) method than using the sliding text-entry method (Mean=194.1). The ANOVA test ($F(1, 13) = 15.705$, $p < .005$) showed that there were statistically significant differences between the two text-entry methods.

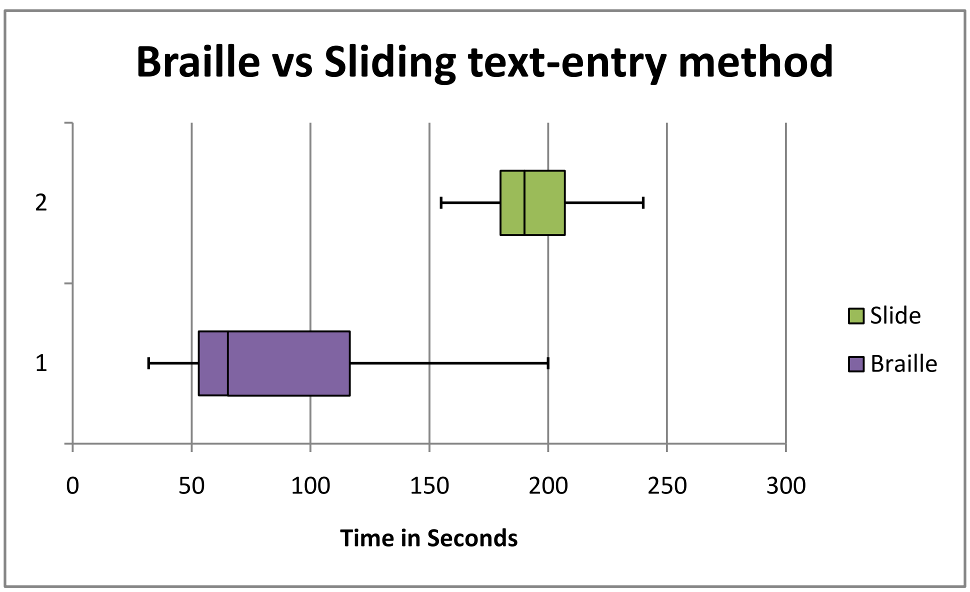

Figure 33: Box-and-whisker plots of usability measures per text-entry method

5.3.3 Errors

Table 4 shows all the error types that occurred in the app and their severity. The severity is based on Molich and Jeffries' [55] 3 point-scale in table 5. Table 6 highlights these error types on each task. Task 4 and Task 8 appear to contain most error types. Task 3 follows second, Task 5 and Task 7 follow third and the rest finish last.

Table 4: List of error types

No	errors	level
E1	Problem with double tapping gesture accidentally triple tapping which result to single tap	1
E2	Double tapping any of the main view assuming it is a back button	1
E3	Double tapping in the wrong position when tapping the back button	2
E4	Double tapping the delete key accidentally	2
E5	Selecting the wrong number during dialing for a number	2
E6	trial and error in activating the instructions	1
E7	Problem with sliding event-accidentally single tapping or unfinished the sliding event	2
E8	Initiate the Slide finger in the wrong area zone. Mostly the upper zone or the lower zone.	3
E9	Forgetting to say "Piga" (Call) during voice call. Instead, start by pronouncing the numbers. Pronounce the numbers in groups like "hamsini" (fifty) instead of "tano" "sifuri" (five zero).	3
E10	Selecting the wrong Braille combination number due to uncomfortable handling of the phone.	3
E11	Assuming the numbers were written when she was browsing across the numbers	3
E12	Assumption that he has made a mistake when accidentally activating the instructions, so he taps the delete key or taps the go back to start over.	2

Table 5: Error rating scales

Level	Details
1	Minor (delays user briefly)
2	Serious (delays user significantly but eventually)
3	Catastrophic (prevents user from completing their task

Table 6: Error types with respect to task

TASKS	ERROR CATEGORY NO
T1	E1,E2,E3,E7
T2	E1,E2,E3,E7
T3	E1,E2,E3,E4,E5,E6,E7

T4	E1,E2,E3,E4,E6,E7,E8,E11,E12
T5	E1,E2,E3,E7,E9
T6	E1,E2,E3,E7
T7	E1,E2,E3,E7,E10
T8	E1,E2,E3,E4,E6,E7,E8,E11,E12
T9	E1,E2,E3,E7
T10	E1,E2,E3,E7

5.3.4 Preference

Although the performance of the Braille text-entry method was better than the sliding method, the sliding method outperforms that of the former by 20% in terms of preference. Figure 34 shows this result.

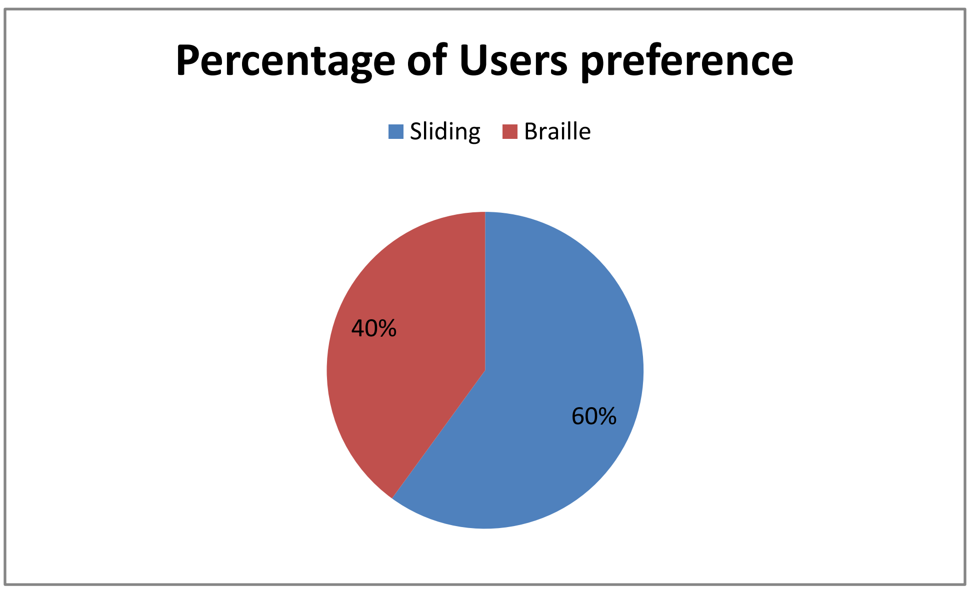

Figure 34: User preference on the two text-entry methods

5.3.5 System Usability Scale Results

The average score of the SUS questionnaires was 81 and standard deviation was 13.95. In figure 35, the grading curve, shows that this score is equivalent to grade A.

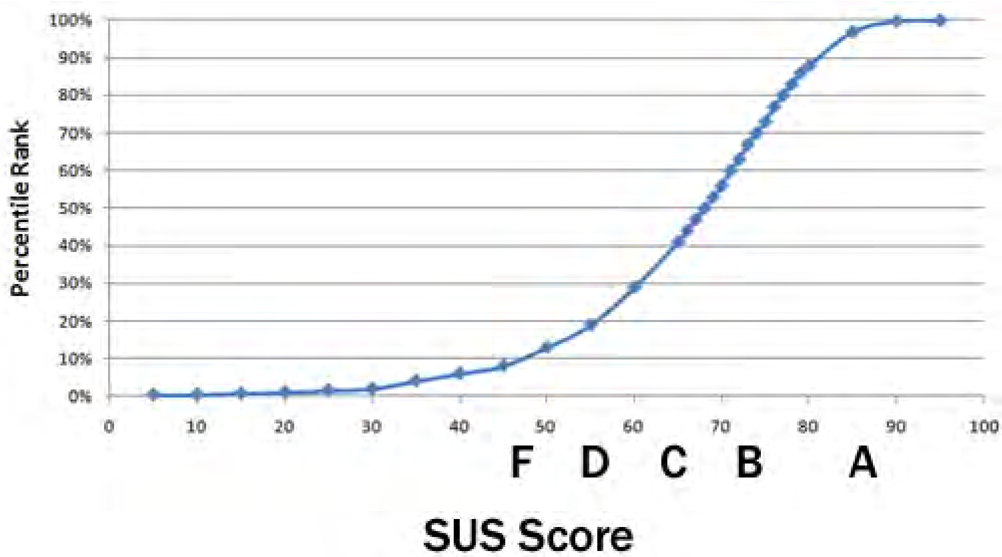

Figure 35: SUS grading curve [53]

5.4 Discussion

The results stated above clearly show that 100% of the users did manage to complete the tasks independently except for task 5 (80%), task 7 (90%) and task 8 (70%). Task 8 appeared to be difficult enough that 30% of the users failed to complete it. The two (2) male users had secondary education while one female user had primary education. It seems, therefore, that the reason for failing to complete task 8 was neither the educational background nor the gender. This result was observed to be from a combination of error types, namely E1, E4, E8, E11 and E12. Most of these errors caused users to start the task over. Although error types, E11 and E12 are considered to be serious or catastrophic, still they can be avoided after the user gets used to the method or understands well how the method works. These error types were mostly observed in the early stage of learning (first trial) when users were still getting familiar with the new text-entry method.

Error types E1 and E8 can be avoided or reduced by providing more training on how to interact with the application. One of the users recommended more training on how to perform the double tapping and the slide gesture. Another user confessed that she was scared when using the program for the first time, so tapping or sliding across the screen was a very tense exercise for her. Although, with audio feedback, users could recover from most error types, one user still suggested that keypad tones should have been included. This will provide additional sensitivity and awareness in different parts of the app and gestures performed.

As for E4, long tapping the button should be implemented instead of double-tapping the button. This was recommended by one user who said “I think delete button gesture should change to be the long tap gesture as we are used to”. It was observed that most users appeared to imitate the long press on the button thinking it would delete everything which was not the case. They ended up tapping the delete button multiple times hoping it would delete the text faster as they are used to, but in reality the double tapping move deleted everything. Although it was clearly explained in the instructions that double tapping would delete everything, users apparently never paid attention to this and instead performed part of the task as they were used to with their phones.

Task 5 appeared to be the second difficult task to the users. Looking at its performance, recognition accuracy for names was far better than recognition accuracy for numbers. Two users’ voices were unable to be recognized in the voice calling view when dictating numbers. They were male and female, hence gender was possibly not a factor. This shortcoming can be due to the fact that there was too much variation of pronunciations of digits, the pause in between, the speed of pronunciations and too many words (digits) to pronounce which affected the accuracy. Some users tended to hesitate during pronunciations trying to recall the number correctly and so pausing for a long time or mispronouncing the number. The users were trained to perform in a way the recognizer would have recognized them but still for some users it did not work. This was different when it came to dictating names. Even for those users who failed to be recognized, at least one trial was successful in the five (5) trials performed. We think, as few words were pronounced, the recognizer tended to perform better. Otherwise, to improve the recognizer accuracy, more voices are required for adaptation, from male to female voices. With this, we are confident of convincing more users to use this feature, since only one user did not prefer it.

Six users know and use Braille machine regularly while the remaining know but do not use it. However, in Task7, only one user among those six users was unsuccessful. Error type, E10, was one of the major reasons for this result. The user would select a wrong combination, accidentally using two fingers to tap or select one Braille code. Some users suggested more training on handling the phone. Others suggested redesigning the interface so that the alignment of fingers is almost like that of the Perking machine, and instead of holding the phone with two hands the phone would lie on the table. This suggestion seemed to be possible in touch phones which have bigger screens because there is space to align all six fingers horizontally on the screen. For smaller screens, we believe, it would be difficult.

Observing the results of the two text-entry methods, the Braille-method was faster than the sliding method. This result could be due to the fact that the sliding method by design appears to take longer for letter selection. The user would require tapping multiple times in order to select just one letter while for the former method, it is just one combination tapping. Furthermore, the sliding method appeared to contain more error types than the Braille-method. Error types, E4 and E8, were observed to be the major ones reoccurring. However, and surprisingly, more users prefer the sliding method than the Braille-method. Most of these users, opted for it because, it is in some ways similar to the writing method they are used to in their current phones. One user said, “I like this method because of assurance that you go precisely to the right letter”. Another user mentioned that she had already forgotten how to write in Braille code, and therefore she prefers the alternative method. Another user preferred it just simply because it was something new. He actually said, “I like this method because it is something new, something fresh”. For the majority who prefer the Braille code, it was because it is their language. He said, “It feels nice to write the text message using your own language”. One user recommended both methods to remain in the app in order to enable those people who do not know how to write Braille to use the sliding method.

The scoring on the SUS questionnaire shows clearly that overall, the users found the app to be usable. In addition to that, the result showed that, there was a great possibility that users would recommend the product to their friends. This was also pointed out in the users' general view about the app. Besides these good promising results, other comments and suggestions were as follows:

- One user commented "I believe now the end of our problem has arrived after using the app. Now we will be able to write a text message alone without any assistance"
- Another user was thankful for creativity in the work, as he already accepted the defeat of using these touch screen mobile phones but after using this app, he said he now he knows that he can use them without too many problems.
- Another user said "if I have money, right then I will go to buy the phone which supports this app immediately".
- One recommended that donors should donate mobile phones which support the app.
- One user was impressed by the fact that the research focused on the visually impaired people in Tanzania.
- Another user remarked that in order to use this app, one should have at least a primary school education or know how to read and write.
- One suggested that the app should enable users to regulate voice speed and provide a variety of speaker voice similar to the JAWS software on the computer.

Apart from these issues, it was encouraging to see that users understood the synthesized voice giving the instructions for most parts of the app, although in the beginning for some users it was confusing and they had to listen more than once to understand what was said. For instance, pronunciation of letter "i" when navigating letters in sliding-method was a bit confusing, which sounded like letter “e”. One user said that this is one of the parts that needed to be improved. However, he said at the moment it was fine to use. For some who have used JAWS, they said, the synthesized voice sounded much better because it pronounced in Swahili and not in English. However, it fails when it comes to pronouncing English words.

In the end, we were generally pleased with the performance of the app for our visually impaired participants; especially when we keep in mind that for all of them it was the first time using the touch screen mobile phone and the majority of them at the end managed to perform the tasks independently. Despite participants’ initial scepticism about touch screen mobile phones, they wanted to own them more than before, after recognizing the potential these phones have. Based on the observations, we think if participants spend enough time with the app, they will not have problems with interacting with the phone as errors would likely be reduced.

6 Conclusion

The emergence of touch screen phones in the current generation, raises questions whether the assistive technologies which these phones come with, do really support all visually impaired. Consider a great percentage of visually impaired living in developing countries, very little research on assistive technologies is known from our knowledge relating to minority languages such as Swahili because reports on previous research are nowhere to be found. Therefore, this study was aimed at implementing a mobile application that would enable Swahili visually impaired people to use touch mobile phones more independently.

In implementing the app, explicit questions that the research revolved around were as follows:

- Can a proposed mobile app solve the current challenges faced by the visually impaired people when using the mobile phones?
- How usable is the app to the visually impaired people?

This thesis addresses these issues by building an app in the current popular Android mobile platform. Different iterations were developed using a User-centered design methodology where users are involved from the requirements gathering stage to the design process and all the way to the evaluation process. In addition, the app usability came mainly as the result of fastidious adherence to design rules and heuristics. Working with visually impaired users in the development of the app was a challenging task due to the sighting problem, but it made it easier to understand what the users needed from the app. The most common way of calling someone was by direct dialling because some of the participants could memorize as many as twenty mobile numbers. Speech synthesis was integrated into the app to be provide auditory feedback.

In order to answer the first research question, the app was evaluated using user testing based on task completion rate. The user testing with the visually impaired users was able to highlight aspects of the design that needed to be revised. The single click was not accepted by the majority of the participants. It gave them problems by accidentally triggering the event without their intention. Double click was suggested for accepting a choice. Single click was suggested to be used to provide instructions on what to do at any particular screen object instead.

In the Braille-method interface, the buttons which represent the six Braille code cells were too small for some users for interaction. Users who had long fingers were tapping on unresponsive areas of the interface. So it was suggested that the size of the buttons be increased in order to accommodate such users. The auditory feedback is used in all areas which involve notification and verification from the app.

The results of the user evaluation showed that majority of the visually impaired users were able to complete the tasks independently and excited to practice more. Although, the results showed that the sliding-text entry method had more problems than the Braille-method, still users showed preference in former method. The users were willing to use the app even though the quality of the voice was not natural. The synthesized voice was good enough for users to understand the instructions.

In order to answer the second research question, the app was evaluated using a System Usability Scale (SUS). The scoring on the SUS questionnaire showed clearly that users found the app overall to be usable and preferable. The results also showed that there was a great possibility that users would recommend the app to their friends.

In conclusion, this thesis has shown that it is quite feasible to build an app that solves challenges faced by the visually impaired when using touch screen phones. While the challenges were explored on some common basic features, namely call management, SMS management, listening to music, performing calculations and listening to web news, not all of them were explored. There is scope for more work in adding more features to the app, and it is believed that this work has laid a solid platform for future extensions.

7 Future Work

7.1 Deploying the app on mobile devices

At the end of the interview, users asked about the type of phones the app can be installed to, so they can buy. At the moment, only one phone type has been tested to be usable. However, it is expensive. Therefore, more testing is required in this direction. The test will be conducted on low end, cheaper phones available currently in the market and that are affordable to the visually impaired, most of whom have low income. Through our quick survey, visiting local mobile shops, the following phone brands that support the Android platform (version 2.3 and above) in time will be tested:

- Huawei phones
- ZTE phones

7.2 Refining the text-entry method

At present, the slide-method is preferred to the Braille-method. However, the slide-method is slower than the Braille-method when it comes to writing a text message. Therefore, more work is required to either refine and improve the method or explore new text-entry methods. There was a suggestion about assigning a long tap gesture on the delete button for all the text-entry methods. This change is straightforward and can be implemented immediately.

7.3 More Voices adaptation

Currently, the speech recognizer is not very reliable for recognizing phone numbers. The acoustic model has been trained with a single male voice whose performance is poor. To improve recognition performance, further adaptation is required. Thus, more voices are required for adaptation which must include a variety of both male and female voices.

7.4 More features

7.4.1 Integrating Mobile money system (M-Pesa, Airtel Money, Tigo pesa)

There is potential for further mobile money adoption in Tanzania based on the current rate of access to mobile phones [56]. This platform has enabled users to use their mobile phones for domestic and international money transfers, mobile payments, and mobile banking. At the moment, the visually impaired people are relying on assistants to use this platform which raises security issues as was mentioned by the majority of the participants. Therefore, addition of this feature to the app reported on in this thesis could help the visually impaired to use the platform independently and securely.

7.4.2 Music features

At the moment, the app allows users to only browse and play all songs. There is still room for improvement by adding more music features such as enabling users to add songs via Bluetooth, deleting the songs, creating favourite playlists, etc.

7.5 Integrating a natural voice synthesizer

At present, the app makes use of a voice generated through formant synthesis for auditory feedback which sounds robotic. However, the app could make use of concatenative synthesis to make the voice sound more natural. This could be highly beneficial for reading long text content. There are a number of speech synthesizer engines available which support this. One example is Flite (Festival-lite) [57], which will require further investigation.

8 References

1. M. Sarwar, and R. T. Soomro, "Impact of Smartphone's on Society", European Journal of Scientific Research, Vol. 98(21), pp.216-226, 2013.
2. S. Laird, "How Smartphones Are Changing Health Care?" September 26, 2012, http://mashable.com/2012/09/26/smartphones-health-care-infographic/.
3. R. L. Mace, "Universal Design, "http://www.ncsu.edu/ncsu/design/cud/about_us/usronmace.htm.
4. N. Narasimhan, "Making mobile phones and services accessible for persons with disabilities", 2012.
5. WHO, "Visual Impairment and blindness," August, 2014, http://www.who.int/mediacentre/factsheets/fs282/en/.
6. E. C. Polomé, "SWAHILI LANGUAGE HANDBOOK," 1967.
7. Wikipedia Contributors, "Mwanzo," December 2014http://sw.wikipedia.org, Last Accessed: 2nd September, 2014
8. Kamusi Project, http://kamusi.org/.
9. WHO, October 2006, http://www.who.int/classifications/icd/2006updates.pdf.
10. WHO, "Global Data on Visual Impairments", www.who.int/blindness/GLOBALDATAFINALforweb.pdf.
11. WHO, "VISION 2020 THE RIGHT TO SIGHT," 2007, www.who.int/blindness/Vision2020_report.pdf.
12. Braille Works, "What is Braille?" http://brailleworks.com/braille-resources/what-is-braille/.
13. Wikipedia Contributors, "Slate and stylus", July, 2014, http://en.wikipedia.org/wiki/Slate_and_stylus.
14. Wikipedia Contributors, "Refreshable Braille display,"August 2014, http://en.wikipedia.org/wiki/Braille_display.
15. Wikipedia Contributors, "BrailleNote," July 2013, http://en.wikipedia.org/wiki/BrailleNote.

16. Wikipedia Contributors, “Braille embosser,” April 2014, http://en.wikipedia.org/wiki/Braille_embosser.
17. Wikipedia Contributors, “Perkins Brailler,” November 2014, http://en.wikipedia.org/wiki/Perkins_Brailler.
18. Wikipedia Contributors, “JAWS (Screen reader),” December 2014, http://en.wikipedia.org/wiki/JAWS_(screen_reader).
19. Wikipedia Contributors, “NonVisual Desktop Access, “December 2014, http://en.wikipedia.org/wiki/NonVisual_Desktop_Access.
20. S. Lemmetty, “Review of Speech Synthesis Technology. Master’s Thesis, Helsinki University of Technology,” 1999.
21. W. Chou, and B. H. Juang (Eds), “Pattern recognition in speech and language processing,” CRC Press, 2003.
22. L. R. Rabiner, and B. H. Juang, “*Speech recognition: Statistical methods,”* The Encyclopedia of Language and Linguistics, 1-18, 2006.
23. L. Rabiner, “Applications of speech recognition in the area of telecommunications,” In Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on (pp. 501-510). IEEE.
24. D. Jurafsky, and J. H. Martin, “Speech and language processing : an introduction to natural language processing,” computational linguistics, and speech recognition. International ed. Prentice Hall series in artificial intelligence, Upper Saddle River (N.J.): Prentice Hall. Xxvi, 934 s, 2000.
25. J. Sanchez, and J. Selva Roca de Togores, “Designing Mobile Apps for Visually Impaired and Blind Users,” The Fifth International Conference on Advances in Computer-Human Interactions, (pp. 47-52), 2012.
26. S. Henny, “Talk is cheap – screen reader testing on mobile,” December 2011. http://www.iheni.com/talk-is-cheap-screen-reader-testing-on-mobile/.
27. D. Jonathan, “eSpeak text to speech”, 2006, http://espeak.sourceforge.net.
28. T. Babinszki, “How Do Blind People Use Mobile Phones,” 2010, http://evengrounds.com/blog/how-do-blind-people-use-mobile-phones.

29. A. K. Karlson, B. B. Bederson, and J. SanGiovanni, "AppLens and launchTile," Proceedings of the SIGCHI conference on Human factors in computing systems – CHI '05, p.201, 2005.
30. S. K. Kane, J. P. Bigham, and J. O. Wobbrock, "Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques," In Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility (pp. 73-80). ACM, 2008.
31. P. Lagoá, P. Santana, T. Guerreiro, D. Gonçalves, D, and J. Jorge, "Blono: a new mobile text-entry interface for the visually impaired," In Universal Access in Human-Computer Interaction. Ambient Interaction (pp. 908-917), Springer Berlin Heidelberg, 2007.
32. A. Campos, P. Branco, and J. A. Jorge, "Design and Evaluation of a Spoken-Feedback Keyboard," 2007.
33. S. Mascetti, C. Bernareggi, and M. Belotti, "TypeInBraille: a braille-based typing application for touchscreen devices," In The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (pp. 295-296), ACM, 2011.
34. N. Paisios, "Mobile Accessibility Tools for the Visually Impaired," New York University, 2012.
35. M. Romero, et al, "BrailleTouch: designing a mobile eyes-free soft keyboard," in Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services. ACM, 2011.
36. C. Southern, J. Clawson, B. Frey, G. Abowd, and M. Romero, "An evaluation of BrailleTouch: mobile touchscreen text entry for the visually impaired," In Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services (pp. 317-326), ACM, 2012.
37. B. Frey, C. Southern, and M. Romero, "*Brailletouch: mobile texting for the visually impaire,".* Universal Access in Human-Computer Interaction, Context Diversity: p. 19-25, 2011.

38. T. Paek, and D. M. Chickering, “Improving command and control speech recognition on mobile devices: using predictive user models for language modeling,” User modeling and user-adapted interaction: 17(1-2), 93-117, 2007.
39. K. Sharma, K, T. Suryakanthi, and T. V. Prasad, “Exploration of Speech enabled System for English, arXiv preprint arXiv:1304.8013, 2003.
40. Carnegie Mellon University (CMU), “Pocketsphinx–Sphinx for handhelds, http://www.speech.cs.cmu.edu/pocketsphinx/.
41. H. Gelas, L. Besacier, and F. Pellegrino, “Developments of Swahili resources for an automatic speech recognition system,” In SLTU (pp. 94-101), 2012.
42. G. K. Zipf, “Human behaviour and the principle of least effort.”, Oxford, England: Addison-Wesley Press, 1949.
43. E. L. Hutchins, J. D. Hollan, and D. A. Norman, “Direct manipulation interfaces. Human–Computer Interaction,” 1(4), 311-338, 1985.
44. I. Rogers, H. Sharp, J. Preece, “*Interaction Design: Beyond Human Interaction Design*”, published by John Wiley & Sons, 2011.
45. D. Norman, “Affordance, Conventions and Design,” Interactions, 1999.
46. J. Nielsen, “*Usability engineering* 1st ed., Morgan Kaufmann, 1993.
47. F. P. Brooks, “The Mythical Man-Month: Essays on Software Engineering Anniversar,” Addison-Wesley Professional, 1995.
48. J. Preece, “Interaction Design: Beyond Human Computer Interaction,” 2002, http://www.sharritt.com/CISHCIExam/preece.html
49. K. Ericsson, H. Simon, “Verbal reports as data” , Psychological Review, vol. 87, pp. 215-251, 1980.
50. J. Dumas, J, J. Redish, “A practical guide to usability testing,” Intellect Books, 1999.
51. J. Brooke, “SUS-A quick and dirty usability scale. Usability evaluation in industry,” 189, 194, 1996.
52. J. Kirakowski, 1994, “The use of questionnaire methods for usability assessment, http://sumi.ucc.ie/sumipapp.html.

53. J. Sauro, “Measuring Usability With The System Usability Scale (SUS),” February 2011, http://www.measuringusability.com/sus.php
54. J. R. Lewis, and J. Sauro, “The factor structure of the system usability scale,” In Human Centered Design (pp. 94-103), Springer Berlin Heidelberg, 2009.
55. R. Molich, and R. Jeffries, “Comparative usability evaluation (CUE-4), accepted for publication in Behaviour & Information Technology,” www.dialogdesign.dk/CUE.html.
56. Inter Media, February 2013, “Mobile Money in Tanzania: Use, Barriers and Opportunities,”http://www.intermedia.org/wp-content/uploads/FITS_Tanzania_FullReport_final.pdf
57. A. W. Black, “Flite: a small, fast run time synthesis engine,” 2001, http://www.festvox.org/flite/
58. CCBRT, “Strategy,” http://www.ccbrt.or.tz/uploads/media/CCBRT_Strategy_2008-2012_02.pdf.
59. D. Sasirekha, and E. Chandra, "Text to Speech: A simple Tutorial,"International Journal of Soft Computing and Engineering 2.1 (2012): 275-278, 2012.
60. Wikipedia Contributors, “Speech synthesis,” December 2014, http://en.wikipedia.org/wiki/Speech_synthesis.
61. Y. Y. Wang, M. Mahajan, and X. Huang, “A unified context-free grammar and n-gram model for spoken language processing,” In Acoustics, Speech, and Signal Processing, ICASSP'00. Proceedings. 2000 IEEE International Conference on (Vol. 3, pp. 1639-1642). IEEE.
62. F. R. Sharma, & S. G. Wasson, “Speech recognition and synthesis tool: assistive technology for physically disabled persons, Int. J. Comp. Sc. Telecom, 3(4), 86-91, 2012.
63. A. S. Kurian, B. Narayan, N. Madasamy, A. Bellur, R. Krishnan, G. Kasthuri, and K. Prahallad, “Indian language screen readers and syllable based festival text-to-speech synthesis system,” In Proceedings of the Second Workshop on

Speech and Language Processing for Assistive Technologies(pp. 63-72). Association for Computational Linguistics, 2011.

Appendices

Appendix A

SUS- Swahili version

No.	System Usability Scale	Nakataa kabisa			Nakubali kabisa	
		0	1	2	3	4
1	Ninavyodhani ningependelea kutumia mara nyingi hii program.					
2	Nimeona hii program inachanganya.					
3	Nafikiri hii program ilikuwa ni rahisi kutumia.					
4	Nafikiri nahitaji msaada kutoka kwa mtaalamu ili niweze kutumia hii program.					
5	Naona baadhi ya matumizi ya simu yamejumuishwa vizuri kwenye hii programu					
6	Nafikiri kuna mambo mengi kwenye program haya wiani vizuri au yana kinzana/haya fuati utaratibu.					
7	Nafikiria watu wengi wataweza kujifunza kutumia hii program kwa uwepesi/haraka					
8	Nimeona hii program inasumbua kutumia					
9	Nilikuwa na uhakika/kujiamini sana wakati natumia hii program.					
10	Nahitaji kujifunza mambo mengi sana kabla sijaanza kuitumia hii programu					

SUS- English version

No.	System Usability Scale	Strongly disagree			Strongly agree	
		0	1	2	3	4
1	I think that I would like to use this system frequently.					
2	I found the system unnecessarily Complex.					
3	I thought the system was easy to use.					
4	I think that I would need the support of a technical person to be able to use this system.					
5	I found the various functions in this system were well integrated.					
6	I thought there was too much inconsistency in this system.					
7	I would imagine that most people would learn to use this system very quickly.					
8	I found the system very cumbersome to use					
9	I felt very confident using the system.					
10	I needed to learn a lot of things before I could get going with this system					

Appendix B

Interview Questionnaire

Questionnaire number ____

Date of interview ____

1. Age:…………………………………………
2. Gender:
    a. Male
    b. Female
3. Education?
    a. College/University
    b. Secondary Education
    c. Primary Education
    d. No formal Education
4. Occupation?
    a. Employed
    b. Unemployed
    c. Business
    d. Student
    e. Peasant
    f. Other………………………………..
5. Do you own a mobile phone?
    a. Yes
    b. No
6. If No to question 5, then
    a. How do you communicate with your relatives/friends who live very far?
        i. Letter
        ii. Email
        iii. Others……………………………………………………………………………………
7. How do you entertain yourself?

……

8. How do you receive news?

……

9. How do you do simple calculations?

……

10. If NO in question 5, then are you planning to own one?
    a. Yes
    b. No
11. If No to question 10, then why?
    a. Expensive
    b. Difficult to Use
    c. Others…………………………………
12. If Yes to question 10, when/why?

………

If yes, then what type of Mobile phone do you own?

a. Featured phone(Old goodie)
b. Smart-Phone(Touch screen?)
c. Special phones(Visually impaired people)
d. Others……………………………..

13. What do you use a mobile phone for?
    a. Making ▢ or receiving calls ▢
    b. Writing ▢ and reading sms ▢
    c. Listening to music/radio ▢
    d. Performing calculations ▢
    e. Browsing the internet ▢
    f. Other usage………………………………………………………………………………………………
14. How do you do that?(*For each sub item in question 14*)
    a. ………
    b. ………
    c. ………
    d. ………
    e. ………
    f. ………
15. Do you find it difficult? If yes, how difficult it is?(*For each sub item in question 14 including challenges facing during navigation/accessing an application/feature*)
    a. ………
    b. ………
    c. ………
    d. ………
    e. ………
    f. ………

16. Do you know how to write and read in Braile code?
    a. Yes
    b. No
17. If yes for question 17, then what exactly do you read/write?

………

18. Where and when did you learn braile code? (in School? At what level?)

………

How did you learn?(Any Challenges faced in the process of learning?)

……………………………………………………………………………………………Have you ever use touch screen phone?

……

19. Others (Suggestion, Comments)

……

Mahojiano ya dodoso (Interview Questionnaire)

Dodoso namba ______________________________________

Tarehe ya mahojiano ____

1. Umri:………………………
2. Jinsia:
    a. Kiume
    b. Kike
3. Elimu?
    a. Chuo/Chuo kikuu
    b. Elimu ya sekondari
    c. Elimu ya msingi
    d. Sijasoma
4. Kazi?
    a. Nimeajiriwa
    b. Sijaajiriwa
    c. Mfanyabiashara
    d. Mwanafunzi
    e. Mkulima
    f. Mengineyo………………………………….
5. Unamiliki simu ya mkononi?
    a. Ndio
    b. Hapana
6. Kama "HAPANA" kwenye swali namba 5 basi:
    a. Unawasilianaje na ndugu zako/marafiki ambao wanaishi mbali na wewe?
        i. Barua
        ii. Barua pepe
        iii. Mengineyo…………………………………………………………………………………
        ………………………………………………………………..
7. Unapataje burudani?

……

8. Unapataje taarifa za habari?

………

9. Unafanyaje hesabu za kawaida?

………

10. Kama “HAPANA” kwenye swali namba 5 basi, unampango wa kumiliki simu ya mkononi?
    a. Ndio
    b. Hapana
11. Kama “HAPANA”, kwenye swali namba 10, basi nisababu ipi imekufanya usitake kumiliki simu?
    a. Gharama
    b. Ngumu kutumia
    c. Sioni umuhimu
    d. Mengineyo…………………………………

………

12. Kama “NDIO” kwenye swali namba 10 basi nisababu gani imekufanya utake kumiliki simu ya mkoni?

………

13. Kama “NDIO” kwenye swali namba 5 basi, ni aina gani ya simu unayomiliki?
    a. Simu za tochi(Feature phone)
    b. Simu za kuslaidi(Touch screen/Smartphone)
    c. Simu Special phones(Watu wenye matatizo ya kuona)
    d. Mengineyo……………………………..

14. Unatumia simu ya mkononi kwa matumizi gani?
    a. Kupiga ☐ Kupokea ☐ simu
    b. Kuandika ☐ Kusoma ☐ ujumbe mfupi wa simu
    c. Kusikiliza mziki/redio ☐
    d. Kufanya mahesabu ☐
    e. Kutumia mtandao wa intaneti ☐
    f. Matumizi mengine……………………………………………………………………………………………
15. Unafanyaje?(Kwa kila kipengele kwenye swali namba 14)
    a. ………
    b. ………
    c. ………
    d. ………

e. ………

16. Je, unaona kuna ugumu wowote? Kama”NDIO”, ni ugumu gani?(Kwa kila kipengele cha swali namba 14, pamoja na changamoto zake)
    a. ………
    b. ………
    c. ………………………………………………………………………………………
    d. ………………………………………………………………………………………
    e. ………………………………………………………………………………………
    f. ………
17. Je, unajua jinsi ya kuandika na kusoma braille code?
    a. Ndio
    b. Hapana
18. Kama ndio kwenye swali namba 17 basi kama ni kusoma unasoma nini? Na kama ni kuandika unaandika nini? Kwa kutumia nini?

………

19. Ni mahali gani na niwakati gani ulijifunza kuandika/kusoma Braille code? (Shuleni? Kwenye daraja gani?)

………

Ulijifunzaje?(Orodhesha changamoto zilizukukumba wakati unajifunza)

………

Umeshawai kutumia “touch screen phones”/”Simu za ku-slide”?

……

20. Mengineyo(Mapendekezo au Mahitaji):

………

Appendix C

CONSENT FORM FOR PARTICIPATION IN A STUDY

Title of Project: DEVELOPING A MOBILE APPLICATION FOR BLIND PEOPLE IN A DEVELOPING COUNTRY

Foreword

Greetings, my name is Alexander Kivaisi, a postgraduate student at the University of Cape Town, South Africa. At the moment, I am carrying out a study to investigate the challenges faced by Swahili-speaking blind and visually impaired people when using a mobile phone.

Purpose of the study

The purpose of this study is to investigate the challenges faced by Swahili-speaking blind and visually impaired people when using a mobile phone and come up with a mobile application that will assist them in using it. You are being asked to participate in this study because you have particular knowledge and experiences that may be important to the study.

Procedures

The study will consist of a one-time interview session which will take approximately 60 minutes to complete. Interview notes will be used as instruments to collect the data from the participants. With your permission, I will audiotape the interview solely for the purposes of accurately transcribing the conversation. With your permission I would also like to take photographs. The audiotapes and photographs, as well as the transcriptions will be stored securely at the University of Cape Town. The photographs will be used to present our research at various conferences and at the University of Cape Town.

Additional data will come from evaluation sessions which will also take approximately 60 minutes each to complete. You will interact with the application at various stages during development and at the end. You will be observed and your interactions will be documented and usability indicators will be noted. You will also be asked for feedback and recommendations from one iteration to the next, and will take part in a final usability testing session.

Confidentiality

I assure you that all the information collected from you will be kept confidential. Only people working in this research will have access to the information. We will not put your name or other identifying information on the records of the information you provide. Access to interview notes will be limited. Interview notes will be properly dispose of, destroyed, or deleted after the research period.

Risks

Some questions could potentially make you feel uncomfortable. You may refuse to answer any particular question and stop the interview at any time. We do not expect any harm to come to you because of participation in this study.

Right to withdraw and alternatives

Taking part in this study is completely your choice. If you choose not to participate in the study or if you decide to stop participating in the study you will not be harmed. You can stop participating in this study at any time, even if you have already given your consent. Refusal to participate or withdrawal from the study will not involve penalty or loss of any benefits to which you are otherwise entitled.

Benefits

The information you provide will help us to develop a mobile application that will assist the blind people in using mobile phones more efficiently and more independently.

Who to contact

If you have questions about this study, please don't hesitate to contact:

Mr. Alexander Kivaisi, the investigator, E-mail: regak2010@gmail.com, Mobile number: +255684399516.
Dr Richard C Hill, Chairman of Faculty of Science Research Ethics Committee, E-mail: richard.hill@uct.ac.za, Telephone: + 27 21 650 2786

Dr Audrey Mbogho, E-mail: Audrey.mbogho@uct.ac.za, Telephone: +27 21 650 5108, the supervisor of this study.

Agreement

I confirm that the purpose of the research, the study procedures, the possible risks and discomforts as well as benefits have been explained to the participant. All questions have been answered. The participant has agreed to participate in the study.

(**Investigator**)

…………………………………………… ………………………………………..
Signature of Person Obtaining Consent Date:

The participant agrees to be audio-taped YES NO Initial:…………………………

The participant agrees to be photographed YES NO Initial:…………………………

(Witness)

I confirm that the researcher has explained the elements of informed consent to the participant. The subject knows that their participation is voluntary, and that they do not need to answer all questions. The participant understands that they can withdraw from the study at any time without penalty. The purposes of the research as well as the risks and benefits have been explained. The participant understands issues of confidentiality. The participant has agreed voluntarily to participate in the study.

……………………………………….. …………………………………….
Witness signature Date:

………………………………………..
Witness Name

FOMU YA RIDHAA YA KUSHIRIKI KATIKA UTAFITI (CONSENT FORM FOR PARTICIPATION IN A STUDY)

Kichwa cha habari: KUTENGENEZA PROGRAMU YA SIMU MKONONI KWA AJILI YA WATU WASIOONA KATIKA NCHI ZINAZOENDELEA.

Utangulizi

Hujambo? Ninaitwa Alexander Kivaisi, mwanafunzi wa digrii ya pili katika chuo kikuu cha Cape Town, kilichopo Afrika ya Kusini. Kwa sasa ninafanya utafiti wa kuchunguza changamoto zinazowapata watu wasioona wanaozungumza Kiswahili wakati wanatumia simu za mkononi.

Madhumuni ya utafiti

Dhumuni ya utafiti huu ni kuchunguza changamoto zinazowapata watu wasioona wanaozungumza Kiswahili wakati wa kutumia simu za mkononi na baadaye kutengeneza programu ya simu ya mkononi ambayo itawasaidia wakati wanatumia simu. Unaombwa kushiriki kwenye huu utafiti kwa sababu una ujuzi na ufahamu ambao unaweza ukawa na umuhimu katika utafiti huu.

Taratibu

Utafiti utahusisha mahojiano ya ana kwa ana ambayo yatachukua muda usio pungua dakika sitini/saa moja kumalizika. Maelezo ya mahojiano hayo yatatumika kukusanya taarifa kutoka kwa washiriki (watu wasioona). Kwa ruhusa yako, nitarekodi haya mazungumzo kwenye kifaa cha kurekodia sauti kwa dhumuni la kunakili kwa ufasahaa mazungumzo haya. Kwa ruhusa yako pia, nitafurahi kuchukua picha. Picha na mkanda wa sauti pamoja na maandishi yake yatahifadhiwa katika eneo salama la chuo. Picha zitatumika wakati wa kuwasilisha utafiti wetu katika mikutano mbalimbali ya utafiti na chuoni.

Maelezo ya ziada yatatoka kutoka kwenye vikao vya tathmini ambavyo navyo vitachukua muda wa dakika sitini/saa moja. Utatumia hiyo programu ya simu wakati wa vipindi mbalimbali vya utengenezaji wake pamoja na mwishoni. Utachunguzwa na mahosiano yako na programu hiyo yatanakiliwa pamoja na viashiria vya utumikaji. Utaulizwa kutoa maoni na mapendekezo yako ndani ya hivyo vipindi ambayo yatatumika kwenye tathmini ya utumikaji kwenye kikao cha mwisho.

Usiri

Nakuhakikishia kuwa taarifa zote zilizokusanywa kutoka kwako zitakua siri na hakuna mtu yeyote ambaye hafanyikazi kwenye utafiti huu atakayeambiwa ulichosema. Ni watu ambao wanafanya kazi na huu utafiti tu ndio wataruhusiwa kupata hizi taarifa. Hatutaweka jina lako au utambulisho wako mwingine kwenye taarifa ambazo utatupatia. Upatinakaji wa maelezo ya mahojiano yatawekewa vikwazo. Maelezo ya mahojiano yataondolewa vizuri, yataharibiwa au kufutwa kwenye kumbukumbu zetu baada ya muda wa utafiti kuisha.

Hatari

Baadhi ya maswali huenda yakakufanya uwe na wasiwasi. Unaweza ukakataa kujibu swali lolote na kuacha kabisa mahojiano muda wowote ule utakaoamua. Hakuna hatari yeyote itakayotokea kwako kutokana na ushiriki wako kwenye utafiti huu.

Haki ya kujitoa au vingenevyo

Ushiriki katika utafiti huu ni hiari. Kutokushiriki au kujitoa kutoka kwenye utafiti hakutakua na adhabu yeyote na hutapoteza stahili zako endapo utaona ni vema kufanya hivyo.

Faida

Kama utakubali kushiriki kwenye utafiti huu itakua ni fanasa na faraja kwetu kwa vile utafiti huu una lengo la kuwezesha watu wasioona kutumia simu za mkononi kwa urahisi zaidi na pia kutokuwa tegemezi kwa wale wanaotumia wasaidizi kutumia simu.

Nani wa kuwasiliana naye

Kama kuna swali kuhusiana na utafiti huu usisite kuwasiliana na:

Mr. Alexander Kivaisi, *Mtafiti mkuu*, Barua pepe: regak2010@gmail.com, Namba ya simu ya mkononi: +255787212898.
Dr Richard C Hill, *Mwenyekiti wa Kitivo cha Sayansi na Kamati ya Utafiti wa Maadili*, Barua pepe: richard.hill@uct.ac.za, Namba ya simu: + 27 21 650 2786

Dr Audrey Mbogho, Barua pepe: Audrey.mbogho@uct.ac.za, Namba ya simu: +27 21 650 5108, *Msimamizi wa Utafiti*.

Makubaliano

Ninathibitisha ya kwamba dhumuni la utafiti huu, taratibu za tafiti, hatari zinazoweza kutokea na pamoja na faida zake vyote hivyo imeelezwa kwa mshiriki. Maswali yote yamejibiwa. Mshiriki amekubali mwenyewe kushiriki kwenye utafitii huu.

(**Mtafiti**)

…………………………………………… ………………………………………..
Sahihi ya Mtu anayetaka kupata ridhaa Tarehe:

Mshiriki amekubali kurekodiwa kwa njia ya sauti: Ndio au Hapana:……………………….

Mshiriki amekubali kupigwa picha: Ndio au Hapana:……………………….

(Shahidi)

Nadhibitisha ya kuwa mtafiti amemuelezea mshiriki dondo/mambo yanayohusu ushiriki wa ridhaa katika utafiti. Mshiriki anajua ya kuwa ushiriki wake ni wa hiari na hahitaji kujibu maswali yote. Mshiriki anaelewa kuwa anaweza kujitoa muda wowote bila adhabu. Mshiriki ameelezwa dhumuni la utafiti huu pamoja na hasara na faida zake. Mshiriki anaelewa masuala yote yanayohusu usiri wa taarifa atazotoa. Mshiriki amekubali kwa ridhaa yake kushiriki kwenye utafiti huu.

……………………………………….. …………………………………….
Sahii ya Shahidi Tarehe:

………………………………………..
Majina ya Shahidi

Appendix D (Lexicon)

barua b aa r uw aa

bonyeza b ow nh y eh z aa

cha ch aa

kichwa k ih ch w aa

cheza ch eh z aa

chini ch ih nh ih

elfu eh l f uw

funga f uw nh g aa

fungua f uw nh g uw aa

futa f uw t aa

gazeti g aa z eh t ih

magazeti mh aa g aa z eh t ih

habari hh aa b aa r ih

hali hh aa l ih

hapana hh aa p aa nh aa

hewa hh eh w aa

hifadhi hh ih f aa dhh ih

huu hh uw uw

jana jh aa nh aa

juu jh uw uw

kesho k eh sh ow

kikakatuo k ih k aa k aa t uw ow

kisura k ih s uw r aa

kitabu k ih t aa b uw

kulia k uw l ih aa

kupigana k uw p ih g aa nh aa

kushoto k uw sh ow t ow

kwanza k w aa nh z aa

laki l aa k ih

leo l eh ow

lini l ih nh ih

mbili mh b ih l ih

mia mh ih aa

michezo mh ih ch eh z ow

milioni mh ih l ih ow nh ih

moja mh ow jh aa

msaada mh s aa aa d aa

mtandao mh t aa nh d aa ow

muda mh uw d aa

mwaka mh w aa k aa

mwananchi mh w aa nh aa nh ch ih

mwanzo mh w aa nh z ow

mwezi mh w eh z ih

mwimbo mh w ih mh b ow

mwisho mh w ih sh ow

mziki mh z ih k ih

muziki mh uw z ih k ih

albamu aa l b aa mh uw

miziki mh ih z ih k ih

mwanamuziki mh w aa nh aa mh uw z ih k ih

yote y ow t eh

na nh aa

nane nh aa nh eh

ndio nh d ih ow

nenda nh eh nh d aa

ni nh ih

niambie nh ih aa mh b ih eh

nne nh nh eh

nyuma nh y uw mh aa

ongeza ow nh g eh z aa

onyesha ow nh y eh sh aa

pepe p eh p eh

piga p ih g aa

mpigie mh p ih g ih eh

pili p ih l ih

punguza p uw nh g uw z aa

rekebisha r eh k eh b ih sh aa

rudi r uw d ih

rudia r uw d ih aa

saba s aa b aa

sauti s aa uw t ih

sifuri s ih f uw r ih

simamisha s ih mh aa mh ih sh aa

sita s ih t aa

soma s ow mh aa

tafuta t aa f uw t aa

nisomee nh ih s ow mh eh eh

tano t aa nh ow

tarehe t aa r eh hh eh

tatu t aa t uw

tisa t ih s aa

ufuatao uw f uw aa t aa ow

ukubwa uw k uw b w aa

ukurasa uw k uw r aa s aa

uliopita uw l ih ow p ih t aa

unaofuata uw nh aa ow f uw aa t aa

vichekesho v ih ch eh k eh sh ow

video v ih d eh ow

wa w aa

ya y aa

dogo d ow g ow

badilisha b aa d ih l ih sh aa

sikiliza s ih k ih l ih z aa

mbele mh b eh l eh

jumlisha jh uw mh l ih sh aa

gawanya g aa w aa nh y aa

zidisha z ih d ih sh aa

kutoa k uw t ow aa

simu s ih mh uw

ujumbe uw jh uw mh b eh

juma jh uw mh aa

asha aa sh aa

hamisi hh aa mh ih s ih

peta p eh t aa

maria mh aa r ih aa

joni jh ow nh ih

kumi k uw mh ih

ishirini ih sh ih r ih nh ih

thelathini t hh eh l aa t hh ih nh ih

arubaini aa r uw b aa ih nh ih

hamsini hh aa mh s ih nh ih

sitini s ih t ih nh ih

sabini s aa b ih nh ih

themanini t hh eh mh aa nh ih nh ih

tisini t ih s ih nh ih

andika aa nh d ih k aa

raia r aa ih aa

mwema mh w eh mh aa

bbc b ih b ih c

kutoka k uw t ow k aa

la l aa

Appendix E (JSGF grammar file)

```
#JSGF V1.0;


/**
 * JSGF Grammar
 */


grammar helloalex;


public <actionA> = Andika|Ongeza| Badilisha| Futa| Tafuta;
public <actionB> = Piga|Mpigie;
public <actionC> = Soma|Nisomee;
public <actionD> = Sikiliza | Cheza;
public <actionE> = Uliopita| Unaofuata| Nyuma| Mbele;
public <actionF> = Jumlisha| Zidisha| Gawanya| kutoa;
public <names> =  (Asha| Hamisi| Maria|Peta|juma|joni);
public <maelfu>= Elfu <digitsMoja> [na <mamia>];
public <makumi>= <digitsKumi>[na<digitsMoja>];
public <mamia>= <digitsMamia> [na <makumi>];


public <actionSimu> = (<actionA>|<actionB>|<actionC>) [simu][ya] [<names>];
public <actionMsg> = (<actionA>|<actionC>) ujumbe [wa] [<names>|<actionE>];
public <actionMusic> = (<actionA>|<actionD>)(mziki|albamu)(wa|ya)<names>;
```

```
public <actionCalc> =
(<maelfu>|<mamia>|<makumi>|<digitsMoja>)<actionF>(<maelfu>|<mamia>|<maku
mi>|<digitsMoja>);

public <simu>
=<digitsMoja><digitsMoja><digitsMoja><digitsMoja><digitsMoja><digitsMoja><digits
Moja><digitsMoja><digitsMoja><digitsMoja>;

public <digitsMoja> = moja|mbili|tatu|nne|tano|sita|saba|nane|tisa|sifuri ;

public <digitsKumi> =
kumi|ishirini|thelathini|arobaini|hamsini|sitini|sabini|themanini|tisini;

public <digitsMamia> = Mia<digitsMoja>;
```

Appendix F (Questionnaire results)

User 1- User 5

No/QnNo	U1	U2	U3	U4	U5
Qn1	39 (Partial Blind)	25 (Totally Blind)	53 ially Blind)	26 (Totally Blind)	46 (Partially Blind)
Qn2	Male	Female	Male	Female	Female
Qn3	Chuo	Secondary	Secondary	Secondary	Secondary
Qn4	Employee	Not employee	Employee	Not employee	Employee
Qn5	Yes	Yes	Yes	Yes	Yes
Qn7	Phone	Phone	Radio,Phone	Radio, Phone	Radio, TV
Qn8	Radio	TV, Radio	TV, Radio	TV, Radio	TV
Qn9	No calculator	No calculator, Abacus(rarely)	Sometime calculator	No Calculator	No Calculator
Qn13	Featured Phone	Feature phone	Feature phone	Feature phone	Feature phone
Qn14	Call/Receive phone calls, Writing/Reading need assistance.	Call/Receive phone calls, Writing/Reading need assistance, Radio	Call/Receive ne calls, Writing, Reading need assistance, Radio, Calculator	Call/Receive phone calls, Writing, Reading need assistance, Radio	Call/Receive phone calls, Writing/Reading need assistance, Radio
Qn15	Familiar with the receiving button(hear for vibration. Memorizing all the phone buttons(Memorizing the phone	Memorizing the phone numbers sometime though asking for assistance. Familiar with the receiving	Memorizing the phone keypads,	Memorizing the phone keypads, phone number, ask assistance to look for numbers.	Memorizing the keypads for calling and receiving phone call

	numbers)	button.			
	Using assistance	Using assistance	Memorizing the phone keypads	Voice message, assistance	Memorizing for keypads on writing SMS. Assistance on reading SMS.
		morizing how to play songs			
Qn16	Pressing the wrong button	No challenge	Don't know who is calling. Keypads not user friendly because they are flat and cannot distinguish between buttons. Call the wrong number	No network connection.	-Don't know who is calling. -Wrong call.
	No privacy.	No Privacy	Verify the written message. Use assistance. No privacy	No privacy, when entering a voucher number	Don't know what has been deleted. No privacy.
Qn17	Yes	Yes	Yes	Yes	Yes
Qn18	Novel, subject notes(Social science), Newspaper, Bibles	Novel, books	Letters, Books, Articles,	Story, Subject books	Books, Stories
	Notes, News	Notes	Notes, Mathematics, Letters,	Letters	Subject notes

			Minutes meeting		
	Perkings Braille	Perkings Braille, A4 frame	Perkings Braille	Perkings Braille, A4 frame	Perkings Braille, A4 frame
Qn19	Primary School	Primary School	Primary School	Primary School	Primary School
Qn20	Voice speech for: identifying numbers, Reading message	Voice program for Sending message, reading message	Cheap price, Voice program	Voice program	Voice program for reading SMS
Qn21	Never used	Never used	Never used	Never used	Never used

User 6-User 10

No/QnNo	U6	U7	U8	U9	U10
Qn1	30 (Totally Blind)	28 (Totally Blind)	45 (Partially Blind)	39 (Totally Blind)	27 (Totally Blind)
Qn2	Male	Female	Male	Male	Male
Qn3	Primary	Primary	University	Primary	Secondary
Qn4	Business	Business	Employee	Not employee	Not Employee
Qn5	Yes	Yes	Yes	Yes	Yes
Qn7	Phone	Radio phone, memory card, TV, computer	Radio, TV, CD/DVD, phone radio	Radio	Phone, Radio, TV
Qn8	Radio, call news people	Radio phone	Radio, Someone read for him(Newspaper, Email)	Radio	Radio

Qn9	No calculator	No calculator	Calculator ,Pen with large alphabets	No Calculator	No Calculator
Qn13	Featured Phone	Feature phone	Feature phone	Feature phone	Feature phone
Qn14	Call/ Receiving phone calls, writing, Radio	Call/ Receiving phone calls, writing, Radio	Call/ Receiving phone calls, Radio	Call/ Receiving phone calls	Call/ Receiving phone calls, writing, assistance on reading SMS, Radio
Qn15	Memorizing all the phone buttons (Memorizing the phone numbers)	Memorizing all the phone ons (Memorizing the phone numbers)	Memorizing the phone keypads and steps to go to phonebook.	Memorizing the phone keypads, phone number,	Memorizing the keypads for calling and receiving phone call
	-Writing no assistance -Reading requires assistance	-Writing no assistance -Reading requires assistance	-Writing requires assistance	-assistance	Memorizing for keypads on writing SMS.
	-Listening songs - no assistance. Can transfer songs bluethooth of the familiar phone	Memorizing how to play songs however requires assistance to copy the songs inside the			Memorizing all the steps

	set	memory card			
Qn16	-No network connection. -Keypads don't give sound.	-Recognizing a miss call	Searching for a phone number Recognizing a person who is making a call	No challenge	Wrong number -Recognizing a caller
	-Entering into wrong composing message area -No privacy.	-Requires high concentration (quiet environment). -No privacy	-No privacy	-Delay to get a assistance thus delay to take actions. -Give feedbacks airtime balance	Requires high concentration otherwise you have to start a fresh.
Qn17	Yes	Yes	Yes	Yes	Yes
Qn18	Books, Brochures, By laws	Novel, books	-Books, Letter, Bible	Books, Quran	Books, Stories
	Daily business expenses,	Books	Notes, Letters	Notes	Subject notes
	A4 frame, Perkings Braille	Perking Braille, A4 frame	Perkings Braille (Regularly)	Perkings Braille	Perkings Braille, A4 frame
Qn19	Primary School	Primary School	Primary School	Primary School	Primary School
Qn20	Voice speech for Swahili, cheap price, special training on how to use assistive technology	Voice speech for: Sending message, reading message	Voice program Swahili, cheap, embossed dots in phone keypads.	Acquire freedom of using the phone independenty .	Voice program

Qn21	Never used	Never used	Never used	Never used	Never used

User 11-User 15

No/QnNo	U11	U12	U13	U14	U15
Qn1	36 (Totally blind)	34 (Totally Blind)	28 (Totally Blind)	35 (Totally Blind)	34 (Totally Blind)
Qn2	Male	Male	Male	Male	Male
Qn3	Primary	University	Secondary	Secondary	Secondary
Qn4	Not Employee	Employee	Not Employee	Business	Business
Qn5	Yes	Yes	Yes	Yes	Yes
Qn7	Phone, radio, TV	Phone, computer, home thetre	Radio, TV, phone radio	Phone using memory card, Radio	Phone, computer,
Qn8	Radio, TV	Radio, TV,	Radio	Phone, radio	Radio
Qn9	No calculator	No calculator, abacus.	No calculator	No Calculator	No Calculator
Qn13	Featured Phone	Feature phone, smartphone nokia E63	Feature phone	Feature phone	Feature phone
Qn14	Call/Receive calls , writing SMS	Call/ Receive calls , writing SMS, Radio	Call/ Receive calls , writing SMS, Radio	Call/ Receive calls, Radio	Call/ Receiving phone calls, writing, assistance on reading SMS, Radio, Bluetooth requires assistance
Qn15	Memorizing all the phone Buttons	Memorizing all the phone buttons (Memorizing	Memorizing the phone keypads and steps to go to phonebook.	Memorizing the phone keypads, phone	Memorizing the keypads for calling and receiving

	(Memorizing the phone numbers)	the phone numbers). -Using Jaws no memorizing		number,	phone call
	-Writing no assistance -Reading requires assistance	-Writing no assistance -Reading no assistance. Use Jaws software	-Writing and reading requires assistance	-Writing no assistance	Memorizing for keypads on writing SMS. Reading requires assistance
	-Listening songs - no assistance.	Memorizing how to play songs . Using Jaws sotware no memorizing			Memorizing all the steps
Qn16	-wrong number by pressing the wrong button. -Recognizing a person who is making a call. -Using alternative method using assign tone.	-Recognizing a miss call. -How to put into silence mode. -wrong number -Using Jaws no challenge.	-Keytones on/off -Unlock the keylock	No challenge	-Wrong number -Recognizing a caller
	-Wrong message. -Requires high concentration.	-Wrong message -No privacy - Using Jaws no challenge except during reading	-No privacy -Making corrections during writing	-No privacy.	-Wrong message. -No privacy

		Swahili messages especially in pronunciations.	-Verify whether message is sent or not.		
	-Saving the radio stations. -Locating a file to delete is challenging.				-Memorizing the steps
Qn17	Yes	Yes	Yes	Yes	Yes
Qn18	Articles, Books	Novel, books, articles	-Books, articles, By laws	Story books, By laws, Brochures	Articles, Brochures, Subjects
	Notes	Teaching notes, action plan, stories	Letters		Business notes
	A4 frame, Perkings Braille	Perking Braille, Computer (Laptop)	Perkings Braille (Regularly)		A4 frame
Qn19	Primary School	Primary School	Primary School	Primary School	Primary School
Qn20	No privacy. Voice program in Swahili.	Cheap Voice programs, Swahili pronunciations.	Voice program No Privacy	Device that read messages -No Privacy	-Voice program -Telling time -Access the internet
Qn21	Never used	Never used	Never used	Never used	Never used

User 16-User 20

No/QnNo	U16	U17	U18	U19	U20
Qn1	29 (Totally blind)	31 (Totally Blind)	30 (Totally Blind)	40 (Totally Blind)	34 (Totally Blind)

Qn2	Male	Male	Male	Female	Male
Qn3	Secondary	Secondary	Primary	Secondary	Secondary
Qn4	Not Employee	Not Employee	Not Employee	Employee	Employee
Qn5	Yes	Yes	Yes	Yes	Yes
Qn7	Phone, radio	Radio, phone, computer, TV	Phone	Radio	Phone
Qn8	Radio, TV	Radio, TV, internet	Radio, TV, Phone	Radio, TV, Newspaper	Radio, Newspaper
Qn9	No calculator	No calculator	No calculator	No Calculator	No Calculator
Qn13	Featured Phone	Feature phone, smartphone nokia E63	Feature phone	Feature phone	Feature phone
Qn14	Call/Receive phone calls, riting SMS, Radio, Email and facebook requires assistance	Call/Receive phone calls, writing/reading SMS, Radio, internet	Call/Receive phone calls, Radio, internet	Call/Receive phone calls, telling the time.	Call/Receiving phone calls, Radio
Qn15	Memorizing all the phone buttons(Memorizing the phone numbers)	Memorizing all the phone buttons (Memorizing the phone numbers). -Using Toxi no memorizing	Memorizing the phone keypads and steps to go to phonebook.	Memorizing the phone keypads, phone number mostly used	Memorizing the keypads for calling and receiving phone call

	-Writing no assistance -Reading requires assistance	-Writing no assistance -Reading no assistance. Use Toxi	Memorizing for keypads on writing SMS. Reading requires assistance	-Writing no assistance	-Require assistance in both writing and reading
	-Listening songs - no assistance.	Use facebook, Download and Play music using Toxi	Memorizing all the steps		
Qn16	-wrong number by pressing the wrong button. -Recognizing a on who is making a call.	-Recognizing a miss call. -wrong number -Using Toxi no assistance	-Recognizing a caller. -wrong number	-Recognizing if there is a network. -Stack keypads -Low volume	No challenge
	-No privacy -send the message to the wrong number	-Wrong message -No privacy - Using Toxi no challenge except during reading Swahili messages especially in pronunciations.	-Wrong message. -No privacy	-Wrong message -No privacy -Recognizing who sent you a message	-No privacy
	-Locating a file to play is		-requires assistance in chatting	-Setting up time/alarm clock	

	challenging.		through the internet. Use the phone for recording.		
Qn17	Yes	Yes	Yes	Yes	Yes
Qn18	Books, By laws	By laws, News paper, Bible	-story books, By laws	Books, Bible, articles, brochures	History books, Civics books, Bible, Newspaper(Torch Trust)
	Meetings notes	Letters, headlines	-stories, songs	Notes subject	Teaching notes, minutes notes, Business records
	Perkings Braille	Perking Braille, A4 frame	A4 frame		Perkings Braille (Regularly), A4 frame
Qn19	Primary School	Primary School	Primary School	Primary School	Primary School
Qn20	Voice program in every phone. Special phones at cheaper price	-Voice program should be free like Non Visual desk assistance (NVDA). -The voice program should exist in all phones. -The voice program should pronounce	Voice program that is user friendly.	Voice program that provide notification at different sections on the phone. -Simplicity in accessing music songs -Swahili pronunciations.	-Voice program to read messages. -Cheap price handsets. -Training in ICT -Put Braille codes in keypads

		well Swahili words.			
Qn21	Never used	Never used	Never used	Never used	Never used