ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

DESIGN AND IMPLEMENTATION OF PHONETIC ETHIOPIC KEYBOARD SYSTEM FOR ANDROID BASED SMART PHONES

By

MICHAEL SHIFERAW

A Project report submitted to the School of Graduate Studies of Addis Ababa University in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science

June, 2010

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
FACULTY OF INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

DESIGN AND IMPLEMENTATION OF PHONETIC ETHIOPIC KEYBOARD SYSTEM FOR ANDROID BASED SMART PHONES

By
MICHAEL SHIFERAW

Advisor: Solomon Atnafu (PhD)

Name and Signature of members of the Examining Board:

NAME	SIGNATURE
1. Dr. Solomon Atnafu, Advisor	______________________
2.	______________________
3.	______________________

Dedicated to my aunt Yeshumnesh Dejene.

Acknowledgement

First, I thankfully acknowledge the help, guidance and support of God in my whole life, THANK YOU GOD. Next, I would like to thank everybody that helped and supported me in a way or another. I am deeply thankful to my advisor Solomon Atnafu(PhD) for his constructive suggestion and encouragement during this project.

Special thanks to Bezi, Getasew, Masresha, Asma, Tesfaye, Fetiya, Amsale, Abebe, zenebe, Ismael, Andualem, Amany, Neba Tekelay and Selama for their valuable support throughout the course of the project.

Again my special gratitude goes to my best friends Abel, Adam, Aman, Filly, Seny, Mussie, Nati and Neba for helping me through all those moments during my study, good and bad. I am glad to have such wonderful friends.

And finally, I thank all my family my mother, father and brothers. I am pleased to be a member of such a wonderful family. Thank you all for your unconditional love and support. Words won't express what you have sacrificed, in order to bring me up to this point; I always respect and love you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abbreviations

ADT :	Android Development Tools
API:	Application Interface
IDE:	Integrated Development Environment
IME:	Input Method Editor
IMF	Input method Framework
NPD:	National Purchase Diary
OS:	Operating System
PDA:	Personal Digital Assistant
QSAE	Quality and Standard Authority of Ethiopia
SDK:	System Development Kit
SMS:	Short Messaging Service

Abstract

At present day, the computing paradigm is shifting towards hand held devices. Many companies are working together to accelerate innovation in mobile technology and offer customers a richer less expensive and better mobile experience. These handheld devices are becoming widely used in our country. Nevertheless many users were not able to efficiently utilize their devices due to lack of language support. The desire for Ethiopic text entry on mobile devices increased with the advent of the mobile technology in our country.

The objective of this project is to design and implement a phonetic input method for Ethiopic scripts for the widely accepted Android mobile OS. Android OS is selected by considering its current popularity and its open facilities for developing an input method. This work proposes a language neutral and application independent phonetic Ethiopic input method that supports both soft and hard keys of the phones. For this purpose the system is designed based on the Ethiopic symbol mapping principles that is identified from the Ge’ez Frontier Foundation’s specification. The system design is implemented using the Android Development Tools (ADT) plug-in with Eclipse integrated development Environment (IDE). The system is also implemented in such a way that Ethiopic and Latin scripts can be switched to enable multi-script text entry. It was possible to integrate the implemented system with basic mobile application such as messaging and phonebook applications.

Keywords:- Ethiopic input method, Phonetic Ethiopic input method, Android Ethiopic, Ethiopic for Smart Phones.

CHAPTER ONE

INTRODUCTION

1.1 General Background

At present the computing technology is moving into our day to day life. The computing paradigm is shifting towards hand held devices.

Currently, these handheld devices are becoming widely used in our country. The applications of these devices are mostly with foreign languages. If these devices can provide their services in the local languages, they will gain wide acceptance among the users and more application can be developed using the local language.

Due to the large number of alphabets in the Ethiopic scripts, it has been difficult to use these scripts on small mobile devices. However, in the past few years, some efforts have been made to incorporate the Ethiopic scripts with handheld devices [1, 2, 3, 4, 5].

Most of previous works on this area mainly provide Ethiopic input method that is dependent on specific language, specific devices model and application. This project, however, aims to provide a language independent Ethiopic input mechanism that will work for all essential application (with multi-script text entry support) on android based smart phones. Due to the shift of the technology to mobile devices Ethiopic language input support would facilitate the development and use of more local application which will be used by many people.

With the implementation of an appropriate Ethiopic input method it can be possible to develop and use application that can use the Ethiopic scripts on an android based Smart phones. We can save our contacts, write diaries and send text message using the Ethiopic scripts. This project paves a way to other developers to develop an important android application using Ethiopic scripts without worrying about the input mechanism. It will also initiate ideas towards full localization of the android phone. The project can also be implemented for other platforms and this will enhance the use of the technology in parallel with the local languages.

1.2 Statement of the problem

Currently the majority of the latest smart phone devices offer most of the basic functionalities that our laptops or desktops do. However, the input method for some scripts including Ethiopic is still in progress. The desire for Ethiopic text entry on mobile devices is increasing in our country. Some trials have been made to integrate Ethiopic scripts with the handheld devices. But much work is not done towards its implementation. The Ethiopic script by itself also poses problems for exploiting modern communication technology due to the large number of letters it constitutes.

Today's, smart phones came with different type of text entry method. Many of them provide either a soft keyboard which is an on screen keyboard or a hard keyboard, and some have both. Hence it is required to provide an input method which fits into all types.

On the other hand, Ethiopic text entry is possible for personal computers using the different input methods that are available. And Phonetic input methods are the commonly used. However, No previous work is done on design and implementation of phonetic Ethiopic Keyboard for smart phones. Thus it is required to develop such system for smart phones in order to provide the users with an easy Ethiopic input method.

1.3 Objective and Scope

1.3.1 General objective

The general objective of this project is to design and implement a phonetic input method for Ethiopic scripts for both virtual and physical keypads of the android-based smart phones.

1.3.2 Specific objectives

In order to achieve the general objective, the following sets of specific objectives are set.

- Identify the requirements of the system and analyze the available standards.
- Design phonetic based keyboard layout that would be optimum for both virtual and actual keyboard.
- Implement a phonetic Ethiopic keyboard for an android-based smart phone and integrate it to the popular multi-script applications such as editors and email systems.
- Test and evaluate the system

1.3.3 Scope and Limitation of the project

This project focuses on the design and implementation of phonetic Ethiopic keyboard for android-based smart phones and integrating it to the popular multi-script applications such as text editors and e-mail systems. The project does not include the development of multi-script applications. Instead, already existing applications are used. The implemented system supports both Ethiopic virtual and actual phonetic keyboard for an android-based smart phone. Text prediction is out of the scope of this project work.

Moreover, the implementation of this project is limited to only android-based platforms. Other mobile platforms like windows mobile are not considered in this project.

1.4 Methodology

- Literature review will be conducted to study and analyze the existing works, standards and technologies.
- Requirements of the system will be identified and analyzed
- Ethiopic symbol mapping that is optimal for both virtual and actual keyboard is identified from the Ge'ez Frontier Foundation's new specification with slight modification.
- The Ethiopic input method design will be implemented
- The system will be tested to verify that the requirements are meet.

1.5 Organization of the project

The remaining parts of this report are organized as follows. In chapter two, we deal with the literature reviews and related works. In chapters three and four, the analysis and design of the proposed system are presented respectively. Chapters five, presents the implementation, and finally conclusion and recommendations are stated in chapter six.

CHAPTER TWO

LITERATURE REVIEW and RELATED WORK

2.1 Literature Review

2.1.1 Android

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. Android is unique because Google is actively developing the platform but giving it away for free to hardware manufacturers and phone carriers who want to use Android on their devices. It allows developers to write managed code in the Java language, controlling the device via Java libraries. The Android SDK provides the tools and APIs that is necessary for developing applications on the Android platform using the Java programming language [6, 7].

The Android OS is getting wide acceptance in the phone market. According to NPD Group, unit sales for Android OS smart phones ranked second among all smart phone OS handsets sold in the U.S. in the first quarter of 2010. BlackBerry OS and the iPhone OS ranked first and third respectively [8, 9, 10].

The Open Handset Alliance, a group of 71 giant technology and mobile companies which include Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, T-Mobile, ARM Holdings, Atheros Communications, Sony Ericsson, Toshiba Corp, and Vodafone Group Plc have come together to accelerate innovation in mobile technology and offer customers a richer less expensive and better mobile experience. Together they have developed Android the first complete, open, and free mobile platform [11].

2.1.2 Types of Keyboards

Keyboards are input devices that are used for many devices like computers, phones, PDAs and laptops. Nowadays, a wide variety of keyboards are available. **T9**, which stands for Text on 9 keys, is frequently used by mobile phones.

Usually, there are around 110 keys on a computer keyboard. The keyboard is used to enter letters, numbers and other special function keys, which are used for some special functions [12].

There are different types of keyboards available, depending on the layouts of the keyboard, such as QWERTY, AZERTY, etc. The QWERTY keyboards are most commonly used nowadays and have the six alphabets (Q, W, E, R, T, and Y) in the first row of the keyboard. The AZERTY keyboards are used primarily used in the French countries. The types of keyboards vary based on their connection type, the application, the layout of the keys and some special function keyboards [12].

Based on the connection type

There are two types of keyboard based on the connection type the wireless keyboards , which use three basic types of connections, via Bluetooth Keyboards, Infrared (IR) Keyboards and Radio Frequency Keyboards. The PS/2 and USB are the two wired connections that connect the keyboards to desktop computers.

Based on the layout

The ergonomic keyboards are designed considering the ergonomic aspect of the keyboards. It is specially designed as per the comfort of the hands and wrist of the keyboard user. The compact keyboards are slim and usually do not have the numeric keypad that is present on the right side of the other keyboards. Most of the smart phones hard keyboards are also compact keyboard in which there are max 10 keys in a row and there are 4 rows for the keyboard.

The multimedia and gaming keyboards are designed for playing audio and has hot-keys for volume control, play, stop and mute operations. The virtual keyboards are not actually physical keyboards, but they are simulated using software. Usually, the virtual keyboards are used in the PDA and smart phones. There are also other types of text entry methods like handwritten letter recognition, where letters are written on the touch screen, and then "translated" to letters. Voice-to- text systems, which can translate spoken languages into written text, are also available.

Keyboard design theory

The various design tradeoffs and the different approaches used in different keyboarding situations and technologies are discussed below.

Issues to consider during keyboard designing

There are different issues that must be considered related to designing keyboard. Issues of keyboard layout and then issues of large keyboards (those where there are more characters to be typed than keys to type them), typing rule issues are also there [13].

I: Keyboard Layout

There are two ways of designing keyboard layout

Mnemonic

We are using existing information on the keys of a user's keyboard to help the user remember the keying of the character. Mnemonic keyboards are commonly used with Latin-based scripts. If there is a close correspondence between what people want to type and what they see printed on the keyboard in front of them and Phonetic is a mnemonic approach but the phonemic correspondence between the script and printed key is considered for mapping.

Positional

The relative positions of the keys on a QWERTY keyboard are defined positional in relation to each other. What is important is which key is next to which key on which row. This approach is most commonly used when implementing a keyboard based on a typewriter layout or some other standard. The most commonly typed letters are positioned in easy to reach locations, and rarely typed letters are placed in more difficult to reach position. When designing a complete new layout for a keyboard purely in terms of relative positioning and not considering the existing key tops, it is possible to do some analysis to allow typists to type quickly, then by placing commonly typed letters on the keys in the middle of the keyboard, etc.

II: Large Keyboards

In our case, the requirement is to design a keyboard which supports more characters than can be accessed simply from a single press of the 101 or 102 keys on a normal keyboard. There are numerous approaches which are used to extend keyboards and some are present here.

Modifier Keys

The most common approach is to extend a keyboard using modifier keys such as Shift or Ctrl, Alt, Alt-Gr1, Command, Option, etc. depending upon what type of keyboard we have.

The problem is that modifier keys are often used by applications for speed keys or for controlling the application, and requiring their use for typing can preclude their use by the application. Or to the worse, the application may take precedence over the modified key and does not allow the combination to be used for typing a character.

Dead keys

Dead keys work well where there is a very strong mnemonic relationship between the key being pressed and its function. This approach allows the user to type a single character as a sequence of two or more keys on the keyboard.

Operator keys

A different approach to dead keys is to place the modifier after the key it modifies. The major difficulty with this approach is the implementation. We need a system which can go back and edit the document we are typing.

Candidate Window

Other approach to the very large keyboard problem, for such languages as Ethiopic, is to popup a special window as a key is pressed which contains a set of possible characters to be input, which a user can select via the mouse, pressing the initial key or using the arrow keys. As the user types more keys, the possible selection list changes, homing in on appropriate character to input. For example, a Microsoft's Amharic input method (version1.0) results in the following windows as shown in Figure 2.1:

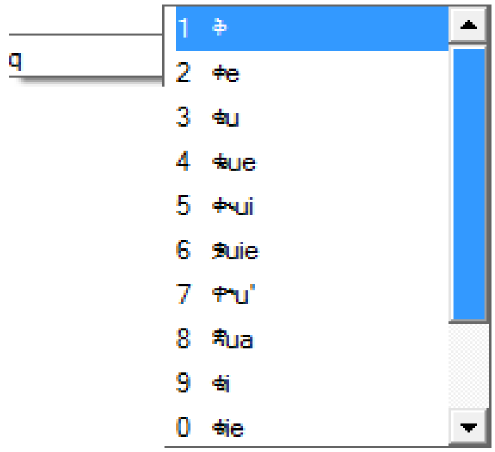

Figure 0-1: xxx

Figure 2.1 Microsoft’s Amharic input method (version1.0) candidate window for ‘q’ or ’ቅ’

The left hand window on Fig 2.1 contains the keystrokes I have pressed and the right hand window contains options for what I might be wanting to type.

III: Typing of Rules

A keyboard may be described in terms of keying sequences. For many keyboards, the sequences are of length 1 in which one keystroke maps to one code input to the application. More sophisticated systems, particularly those with large number of keyboard, allow a keystroke to change the state of this one to one mapping such that a following keystroke is mapped differently.

Different keyboarding requirements require different keyboarding solutions. For some, a simple 1:1 mapping is sufficient. For others, complex in-place editing is needed. Others still require some means of picking from a very long list, and narrowing down that list with each key press, which is true in the case of Ethiopic script.

2.1.3 Ethiopic keyboard standards and specification

The Ethiopic Unicode

In the Ethiopic Unicode Standard version 5.2, there are many Ethiopic characters including the Ethiopic punctuation symbols. It ranges from [1200-137F], [1380-139F] and [2D80-2DDF] hex for the Ethiopic basics, Ethiopic supplementary and Ethiopic extended respectively. The Ethiopic script is used for several related Semitic languages that are spoken in Ethiopia and Eritrea, including Amharic (the national language of Ethiopia), Harari, Gurage, Tigre and Tigrinya.. Ethiopic is written left to right. The writing system is also known as Fidel [14, 15].

QSAE: ES 3449:2008 standard

I have tried to review two of the standards which have been developed for the desktop QWERTY keyboard it is an appreciated effort, towards standardizing the Ethiopic input system but the standards developed are language dependent on Amharic and Tigrigna. And also it is difficult to use it for the compact 10 row QWERTY keyboard. On each case geometrical based keyboard layout is developed [16].

Geez Frontier foundation

Ge'ez Frontier Foundation have come up with a principle and specifications that have been developed to be language and hardware device independent and sound enough that independent parties could implement them for a target device and arrive at approximately the same result. In this work a language neutral Ethiopic specification is briefed using the QWERTY keyboard. Here the Ethiopic alphabets are mapped based on the phonemes and the punctuations are mapped based on the function. A good numeral mapping is also presented. All Ethiopic character set was kept in to consideration that even the old Geez's tonal marker are included in the specification. The work best feet our project because, it is device dependent, it is language independent, and again it is phonological based [17].

2.2 Related work

Some of the important previous works with regard to the Ethiopic input system on hand held devices are described as follows.

An attempt by Shiferaw Abebe, et. al, [2] have been made to enable a practical and efficient composing of short message texts with Ethiopic letters on wireless devices like mobile phones. In this work, an efficient Ethiopic keyboard mapping model (**ETWirelessKeyBoard**) is proposed. **EasyET**, which is an algorithm that predicts the word, helps to decrease the number of typing required to type a word is also developed. The project uses the ETWirelessKeyBoard and

EasyET as the foundation for developing Ethiopic SMS application. They claimed that their Ethiopic SMS application introduces the Ethiopic writing system to the Wireless revolution. However, the ETwirelessKeyBoard they have proposed is limited to three-key T9 text inputting method which is only applicable to phones that have T9 inputting method. Smart Phones with a touch screen and QWERT keyboard support are not considered.

With this regard, it is worth mentioning some of the available Amharic SMS applications that can be found online. KM software has developed an Amharic SMS sender application. It is a freeware and is available freely on the Internet [18]. Another similar application known as GeezSMS will soon be released [19]. This application can be used for sending SMS texts in Amharic or in Tigrigna languages. Both applications can be used wherever there is messaging service. A java-enabled device with Unicode support is required to use the above applications.

Another work, the Amharic virtual keyboard (AVKB) system is software that renders the Amharic input method on PDA environment. This system allows the user to provide input as a form of Amharic character on PDA systems [1]. This software does not consider the actual keyboard. It is not possible to use it with other applications that have multi- script support. The design is not phonetic based; users are required to have an extensive training with the keyboard layout, in order to make them familiar with the layout.

Other project by the Nokia Company is also one of the contributing efforts toward integration of the Amharic language with the mobile technology. In this project the company was able to develop a mobile phone that has Amharic menus and that support Amharic input method. This is one of the few successful efforts towards localizing the mobile devices .However the problem is the support for the Ethiopic language is limited to few models of the Nokia phones. The products are aimed for those who need only the basic communication resources and are not familiar with foreign language. Thus the models are all low end devices which are specific to some users [20, 21].

As it is discussed above much effort has been made on developing applications that support the Ethiopic input method but most of the input systems that are provided are limited to specific Ethiopic language, application and devices. However our project will make an effort to provide a standard input method which is language and application independent. A phonetic based Ethiopic input method for both the soft/hard keypads of android smart phones will be developed.

CHAPTER THREE

SYSTEM ANALYSIS

3.1 Overview

In this chapter the overview of the requirement of the system is presented. The high-level functionality of the System and the user-level requirements that are not directly related to functionality are also discussed. Moreover the functionalities of the system are further described using the use cases, and dynamic models of the system.

3.2 The Proposed System

Specialized keyboard hardware designed specifically for Ethiopic script does not exist today. Ethiopic scripts are entered by mapping the Ethiopic characters to the symbols of some other keyboard devices which are designed for the users of another script and language. And the mappings are either *positional* based on the geometry of the Ethiopic typewriter, or *phonetic* based on the correspondence in the relationship of the letter sounds with functional mappings of punctuation [17].

The newly proposed system is language independent phonetic Ethiopic input software that allows users to provide input in any kind of language that uses the Ethiopic script on android based mobile phones. This system should also support English character entry in order to provide the user an English text entry without changing their input method. The input method must provide a support for both the soft and hard keys of the phone. The existing QWERTY layout will be considered for the Ethiopic input layout. This layout is also adopted for the English text entry method. The system should also provide a language independent phonetic mapping that will be easy to input any language text that uses the Ethiopic script. For example it should be easy for both Tigrigna and Amharic input method. Users can use the soft keys or the hard keys that is found on the keyboard. Figure 3.1 shows the high level system architecture of the proposed system.

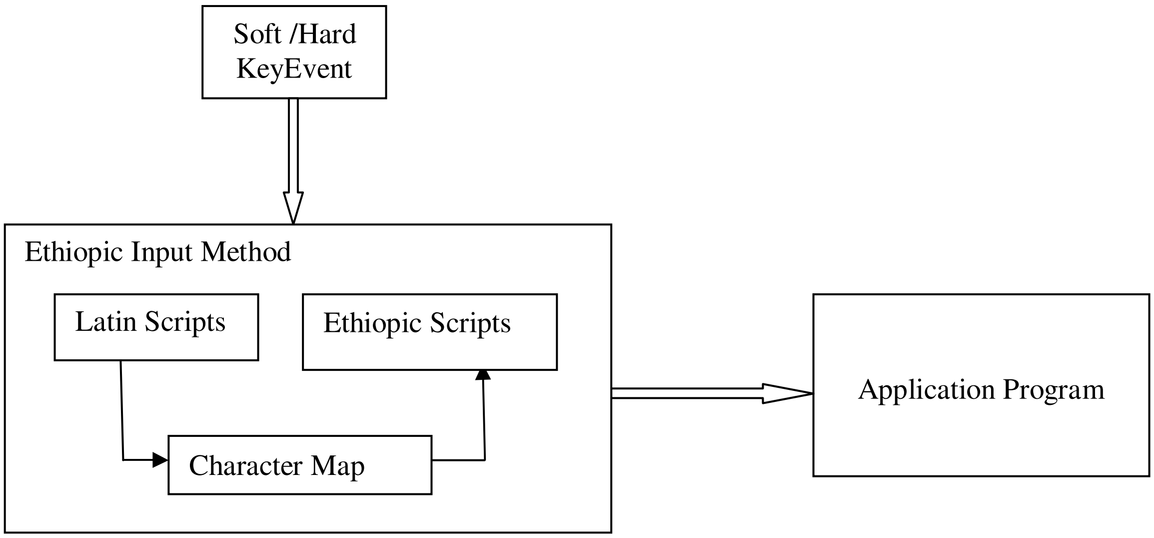

Figure 3.1 High level system Architecture of the proposed system

The user will press either the soft/hard key and this key press event will be handled by Ethiopic IME which will automatically map the Latin key to its corresponding intermediate Ethiopic character and this character will be sent to the application program in which the character will be displayed. A single or a combination of the English character will be mapped with the appropriate Ethiopic character. For example if the user press the key "q" then the system will map the "q" to the Ethiopic "ቅ" and the character will be send to the application for display. And if the user press the character "u" next the character "ቅ" will be modifid to "ቁ" and this character will be sent to the application program to replace the character "ቅ". And if the user wants to write only the character "ቅ" he could press the space char to continue to the other characters.

3.3 Functional requirement

The developed system is expected to provide the following functionalities:

- The system should allow the user to insert an Ethiopic script characters using both the soft and hard keys of the android phone.
- The system should allow the user to insert the Ethiopic scripts using a combination of characters. The combinations are selected in such a way that the sound produced by the Ethiopic character and the phonetic pronunciation of the English characters typed is similar.
- The system should allow entering English characters including numbers and symbols.
- The system should provide a method or shortcut to change the input mode between the Ethiopic and English.
- The system should allow the user to delete the character when the need arises.
- Every Ethiopic script character that is found on the latest unicode Standard table should be supported.

3.4 Non functional requirement

The non-functional requirements expected from the system are also listed below.

- The system must not be complicated and user must be able to use the system easily.
- The system should be working without any error or total failure all the time.
- Since the system is used for inputting characters it should provide a fast response time. Thus response times for the Ethiopic must be the same as the English.
- The system should allow modification if there is a need to add new functionality.
- The system should be secured since it is used for inputting confidential information like passwords.
- The system should run on android based smart phone.

3.5 System Model

3.5.1 Use case model

The functionalities of the proposed system, as mentioned in the previous sections, are explained in terms of system use cases in the following section. The dynamic models of the use cases, the interaction with respect time, are described using sequence diagrams.

Actors

The major actors involved in the system are:

Name: *user*

Description: This actor is responsible to press either a soft or a hard key. He/she is responsible to use the system functionality like change the entry mode or insert a character and delete if it is necessary.

Name*: Application program*

Description: This actor is responsible to display/remove /modify/save the typed characters. Then it can use the character for own purpose afterwards.

Use Cases

The lists of use cases identified in the project are:

1. Switch layout
2. Change input mode
3. Insert character
4. Delete character

Use case diagram

The overall functionalities of the system use cases are modeled using the Fig 3.2 below.

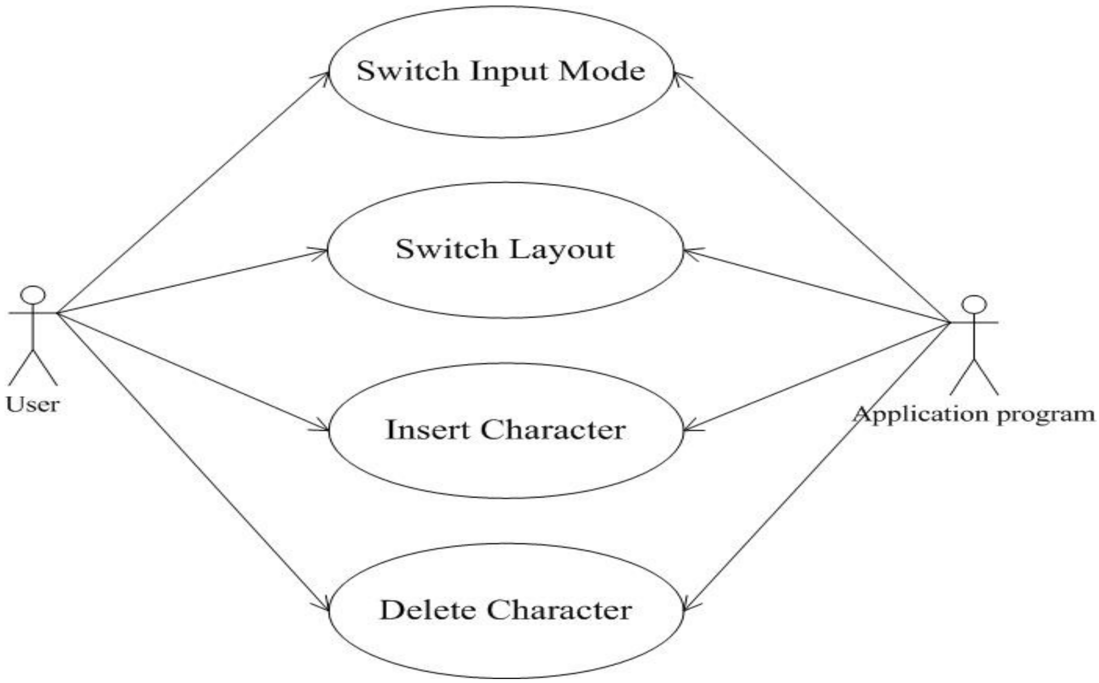

Figure 3.2 Use case diagram of the proposed system

Use case description

Summarized descriptions of each use cases flow are presented as follows.

<u>**Switch input mode use case**</u>

Name: *Switch input mode*

Participating Actor: user, Application program

Description: A use case that allows a user to change the input mode.

(Ethiopic alphabet (ሀለሐ), Ethiopic number and punctuation symbol (፩፪፫), English Alphabet (ABC), English number and punctuation symbol (123)

Pre –condition: user is using Ethiopic as its current input method.

Flow of Events:

1. User press change input mode hard key or key combination.

2. The system update the layout state.
3. The system changes the input mode.

Post- condition : The input mode is changed .

Switch Layout use case

Name :*Switch Layout*

Participating Actor: user, application program

Description: A use case that allows a user to Switch layout.

Pre –condition: user is using Ethiopic as its current input method.

Flow of Events:

1. User press layout changer softkey.
2. The system changes the input mode.
3. The system changes the keyboard layout.

Post condition: The layout is changed

Insert Character use case

Name: *Insert Character*

Participating Actor: user, application program

Description: A use case that allows a user to insert a character

Pre-condition: user is using Ethiopic as its current input method.

Flow of Events:

1. User press the either the soft/hard key.
2. The System map the pressed key code.
3. The System append the character.[A1]

Post-condition: A character is inserted or modified.

Alternative Flows

If the pressed character is an Ethiopic vowel that is coming after an Ethiopic consonant

A1.System replaces the composed character by its new form.

Delete Character use case

Name: *Delete character*

Participating Actor: user, application program

Description: A use case that allows a user to delete a character

Pre-condition: There must be at least one character in the input

Flow of Events:

1. User presses the delete soft or hard key
2. The system removes the a previous character.[A1]

Alternative flow

If there exists a selected text/character

A1. The system removes the selected text/characters.

Post-condition: The previous character or the selected text is removed.

3.5.2 Sequence diagram

The sequence diagram for Switch layout, Change input mode, Insert character and Delete character use cases are presented in Figure 3.3, 3.5, 3.6 and 3.4 respectively.

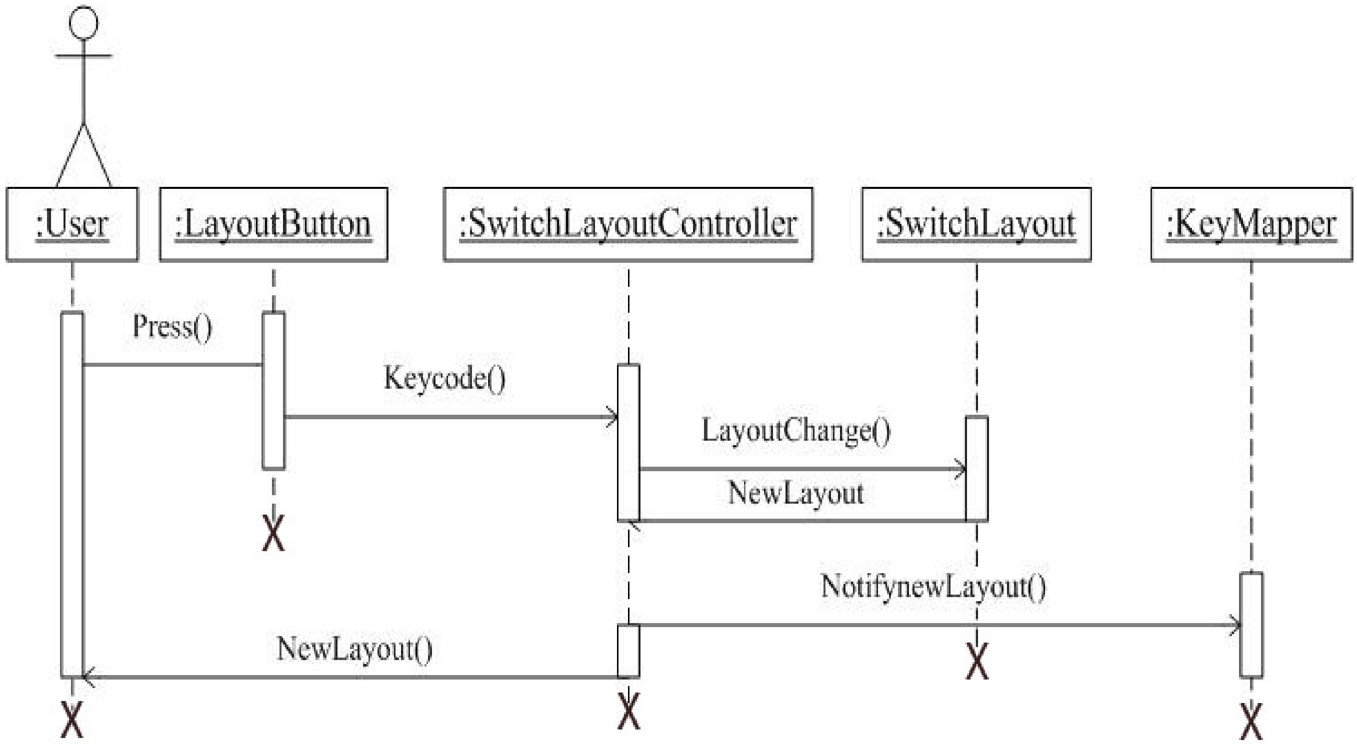

Figure 3.3 Sequence diagram for Switch layout use case

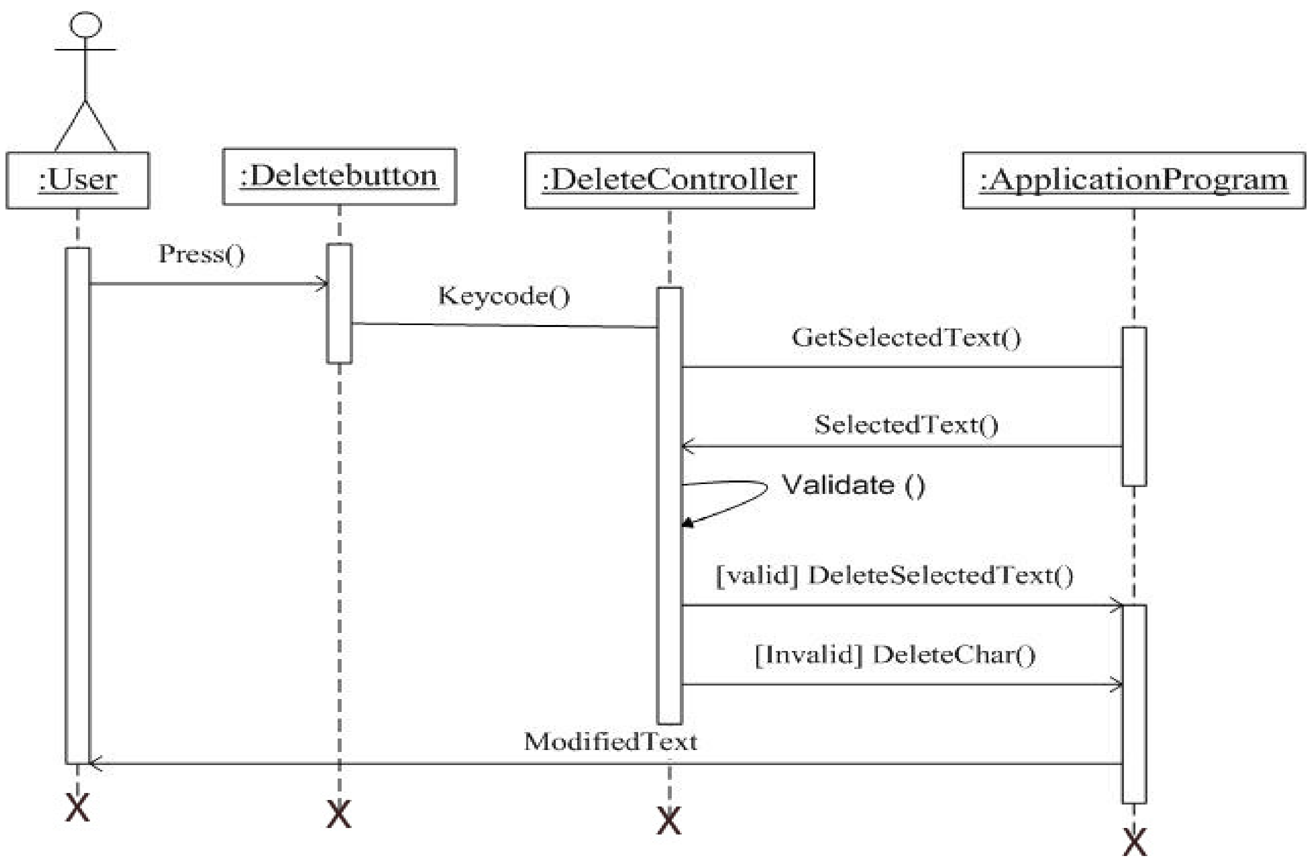

Figure 3.4 Sequence diagram for delete use case

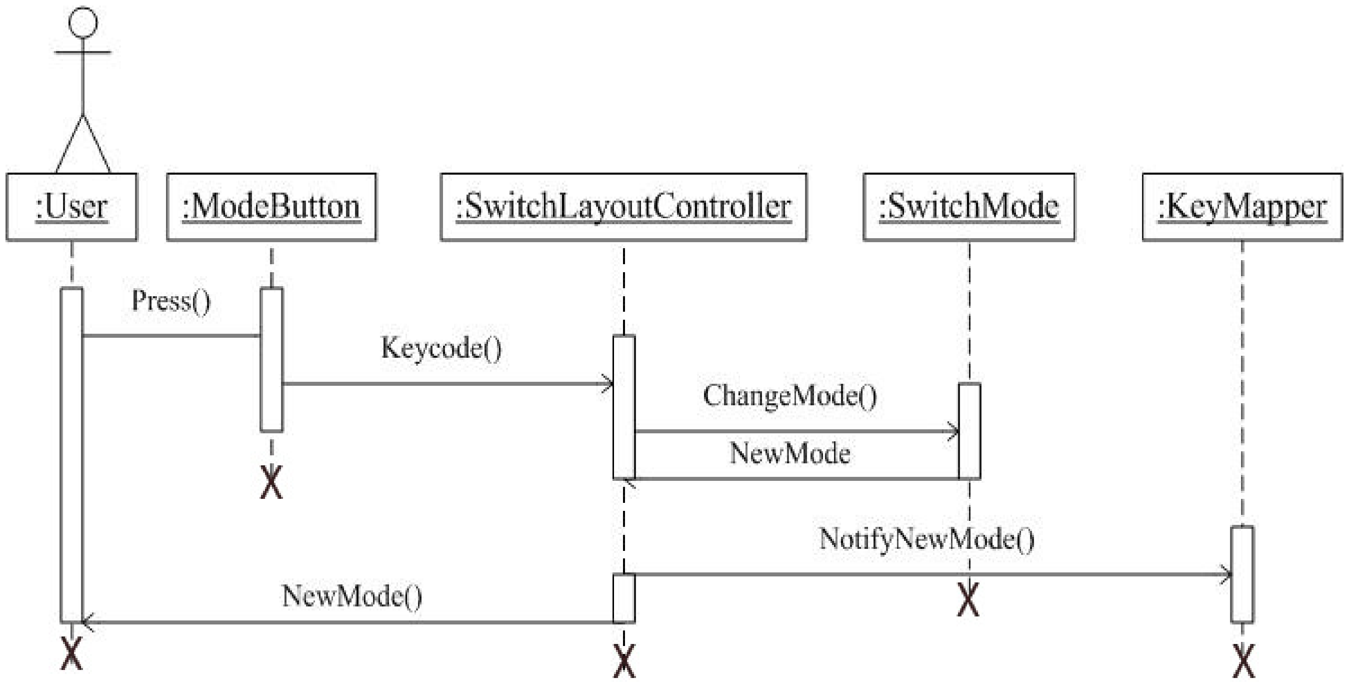

Figure 3.5 Sequence diagram for Switch Input mode use case

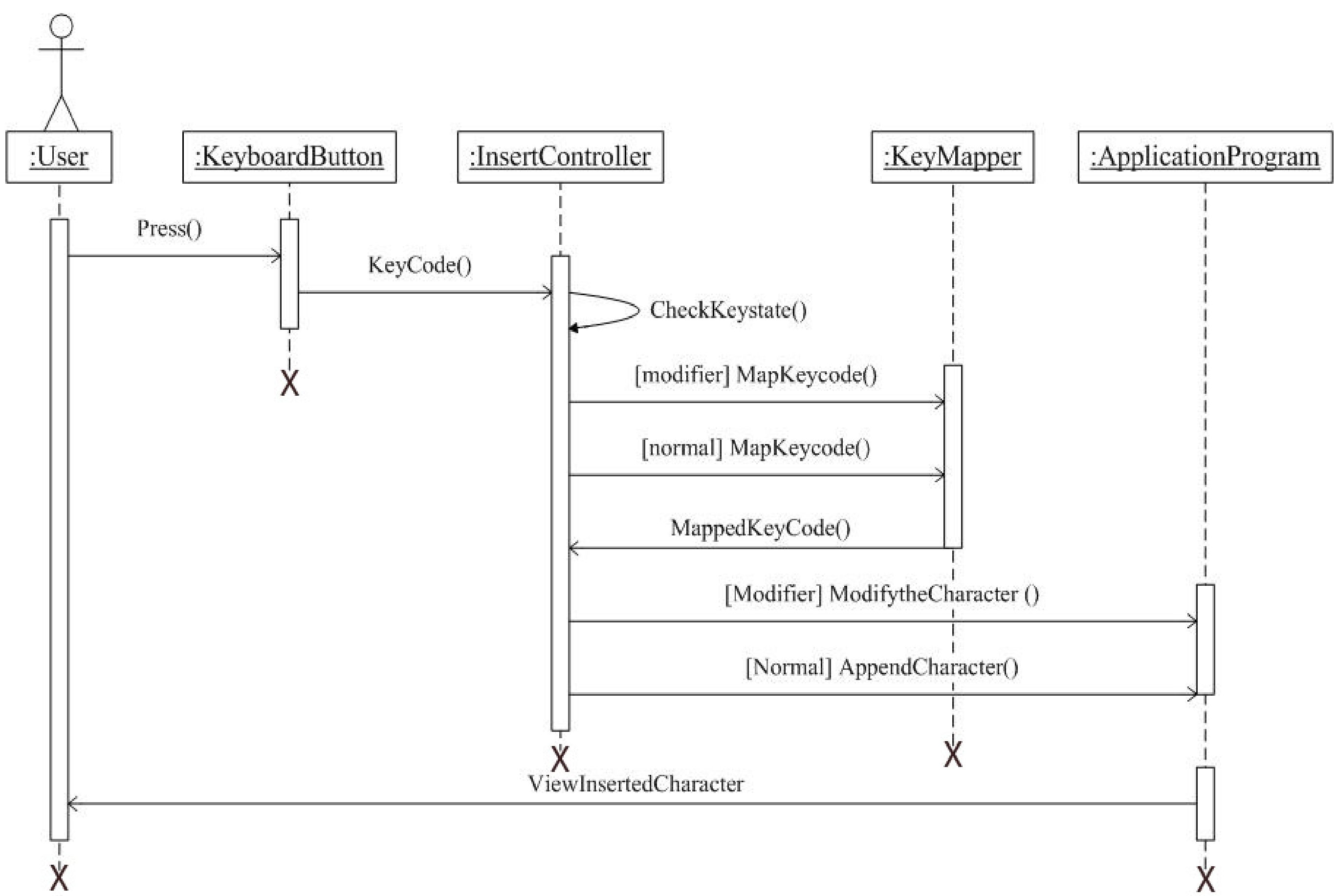

Figure 3.6 Sequence diagram for insert use case

CHAPTER FOUR

SYSTEM DESIGN

The system design deals with transforming of each functional requirement in the System analysis into system components. The main purpose of the design is to specify a solution domain for the analysis. This chapter presents design goals of the system, mapping principles and specifications and the architecture of the system.

4.1 Design Goals

The major design goal of the system includes:

Usability

Different users, ranging from user which is already familiar with a given input device to new user must be able to use system to type in an Ethiopic language easily, as a result the system should provide ease of use by avoiding complexity, unnecessary functionality and by providing candidates for each Ethiopic alphabet. The shortcuts and buttons which are used to switch the layout/modes in the system should be instinctively guessed and remembered.

Reliability

The software should be a system that the users can trust or relay on. Therefore the system should always provide the appropriate and correct character to the single or combination pressed key.

High performance

System should have high performance as much as possible this will be achieved by creating easily loadable interface component for the soft keyboard and optimal mapping specification will be used.

Maintainability

Besides modifications of the input method using the other layouts like the French AZERTY, new functional requirements like text prediction are expected to be added to the system in the near future. Thus the system is developed keeping this in mind so that modification and expansion of the system shall be done easily. Thus xml will be used to implement the layout instead of java.

For example the QWERTY layout can easily be modified by modifying the xml resource file without changing the basic functionality of the system.

Securability

Securability is the ability to provide security to an application and its data. In order to protect the user from the common types of security attack which are related to input methods, like password theft, key logging, the system should never save the characters typed on the permanent storage of the device not to pave a way for unauthorized access of malicious processes.

Reusability

The design of this system can be implemented for other types of smart phones that use different type of platform like the windows mobile and others.

Assumptions and Dependencies

The following subsection deals with describing assumptions or dependencies regarding the Ethiopic input system and its use. These issues include:

Related software or hardware:

Different application software and hardware (phones) may be used in relation to this system, which consequently may create dependencies; end users will most likely use it with contact application, messaging applications, in which most of the applications are assumed to use utf-8 encoding. The system also requires installing an Ethiopic font before use.

Operating systems:

Since the system is going to be implemented for android smart phone, it can only be deployed and used in android operating system.

4.2 Mapping principles

Due to the huge amount of the Ethiopic character set it is not possible to make one to one mapping of key with character code. Therefore an optimized mapping principle and specification is required in order to develop a good Ethiopic input method.

In the past few years there have been some efforts towards standardization of the Ethiopic mapping for the standard QWERTY keyboard [16]. But these standards best felt the desktop or laptop keyboards. Most of the QWERTY keyboard that are designed for the hand held device like the smart phone have a reduced format that have only 10 keys in a rows which shows the difference between the desktop and smart phone QWERTY keyboards. Hence it is not advantageous to use the already existing standard which is for the desktop and laptops standard keyboard.

We adopted a mapping principle and specification that is optimal for Ethiopic input. In this project the mapping principles and specifications are mainly adopted from the principle and specification for mnemonic Ethiopic keyboard of the Ge'ez Frontier Foundation [17].

This work describes principles for devising Ethiopic input methods based on well established linguistic and mnemonic rules. The principles and Specifications are presented using QWERTY based input devices. These principles are intended to be language and hardware (device) independent. So that independent parties could implement them for a target device and arrive at approximately, if not identically, the same result. This is one of the reasons why we choose to use these principles.

The principles have been developed with hardware keyboards as the primary devices for implementation. But still the principles remain valid for soft keyboards too. The specification focuses on phonetic mapping which is based on the correspondence in the relationship of the letter sounds with functional mappings of punctuation. This adds a point to our choice because it meets our functional requirements. A language neutral input method is specified and a support for those rare instances when Out-of-language or archaic symbols are also discussed [17].

The main principle and specification for mnemonic Ethiopic keyboard of the Ge'ez Frontier Foundation are presented as follows [17].

4.2.1 General Principles

A few guiding general principles are presented and applied to the input method specified [17].

Principle of Standards Conformance: All Ethiopic Unicode standard character specified in a QSAE ES 3449:2008 must be available in an Ethiopic input method.

Principle of Utility: As Ethiopic script is much more frequently mixed with non-Ethiopic punctuation and numbers than with letters. All native punctuation of the underlying keyboard device must be available without a changing to the native input method.

Principle of Ergonomics: The keystrokes must be kept minimal as much as possible. When a Homophonic redundancy occurs, letter frequency will be considered for direct mappings. Therefore the most frequently occurring homophonic symbol will have the fewest keystrokes to render.

4.2.2 Mnemonic Principles

Mnemonic principles are defined so that the results require little or no education on the part of the user. It assumes that the user is familiar with the script that the target device natively supports and that the Ethiopic character set is being mapped onto [17].

Principle of Character Class Continuity: This principle states that, letters are mapped to letters, numbers are mapped to numbers and Punctuations are mapped to punctuation.

Principle of Phonological Continuity: Letters are mapped based on the phonemic correspondence between the scripts. When the target keyboard device is designed for a script where letter case is present, the vowel component (the "V" of a "CV" pattern) is mapped to both the uppercase and lowercase forms of the vowel in the target script.

<u>Principle of Continuity of Quantity</u>: Numbers are mapped based on value between scripts.

<u>Principle of Continuity of Function</u>: Punctuation is mapped between scripts based on their functional role in a body of text.

<u>Principle of Graphical Continuity</u>: the punctuation should be mapped on a graphical basis, if a functional equivalent does not exist for punctuation on the target keyboard device.

4.2.3 Ethiopic symbol Mapping

A language-neutral Ethiopic symbol mapping principle and specification is presented in this sub section. Punctuation and numeral mappings are usually language independent. However the letter mapping will depend on the phonemic behavior of the specific language [17].

Letters

The mapping between the letters of the underlying keyboard and the Ethiopic script is mostly based on the Principle of Phonological Continuity.

Consonant

There are some challenges that come into sight with the Ethiopic consonants. These are phonological redundancy of some Ethiopic letters as used in some but not all languages and the lack of a phonologically corresponding Latin letter to map onto.

Phonological Redundancy: phonological redundancy is common in Amharic for example for “h” (ሀ, ሐ, ኀ, ኸ), “s” (ሠ, ሰ) and “a” (አ, ዐ) have more than one corresponding sounds. Most of the time the redundancies are handled by mapping the Ethiopic homophonic group to the same Latin letter, Requiring the less frequently occurring letter to be entered by a double-strike of the target key (*Principle of Ergonomics)*. For example “ሰ” is entered by a single strike to the “s” key, if the next character struck is another “s”, then “ሰ” is replaced by the less frequent “ሠ”. “h” pose an added problem because it has four homophones. Therefore it is handled by mapping the “ሐ” to uppercase "H" and the "ኸ" to uppercase "K" or to "kh" which both borrow from the mappings used in Tigrinya.

Lack of a phonologically correspondence: some Ethiopic letters (e.g. ጰ, ጸ, etc) lack phonological correspondence with a Latin letters and others are represented by two Latin for example "ቸ" and "ሽ" in Ethiopic would be "ch" and "sh" respectively. In this case the additional keystroke of "h" is required. But this slows the user input .Thus the mapping to "c" for "ቸ" and the unused "x" for "ሽ" thereby avoiding additional keystroke and the complications.

Vowels

"The Ethiopic orders of a syllabic family each exhibit the same consonant base ("C" in a "CV" pattern) with a changing vowel component (the "V" in a "CV" pattern). The exception is for the 6th order syllographs which represents a true consonant ("C" only). Researchers have shown that the sixth order is the most frequently occurring of all orders [3]. "The Ethiopic "መ family" is the only member of the syllabary to realize the full set of 14 syllographic orders. Its keyed entry under QWERTY based hardware is presented here for illustration as the syllabic composition archetype. Note that the regular expressions syntax "[ABC]" employed in the tables is to be read as "one of: A or B or C". One or more bracket expression in the form "[mM][iI][eE]" indicates the valid composition sequences of: "mie", "Mie", "mIe", "Mie", "miE", "MiE", "mIE" and "MIE". The expected, and most probable, sequence is formed from the initial elements in a bracket expression sequence, e.g. "mie", the remaining expansions are not ordered" [17]. Please see Table 4.1 below.

Table 4.1 The Ethiopic 'መ' with full set of 14 syllographic orders [17]

key	[eE]	[uU]	[iI]	[aA]	[i I] [eE]		[oO]	[oO][aA]	[uU] [eE]	[uU] [uU]	[uU] [i I]	[uU] [aA]	[uU][iI][eE]	Y[aA]
[m][M]	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ	ሟ	ᎀ	ᎃ	ᎁ	ᎂ	ፙ	ፙ

Due to the phonemic proximity of the seventh and ninth orders in many Ethiopic languages, the seventh and ninth orders are often used interchangeably.

Lone vowels

Two approaches are used to map the lone vowels (አ, ኡ, ኢ, ኣ, ኤ, እ, ኦ) and (ዐ, ዑ, ዒ, ዓ, ዔ, ዕ, ዖ). This represents the Lone Vowels as Vowels and as Syllables.

Lone Vowels as Vowels: a single keystroke of the vowels is mapped with "አ" and double strike is used for "ዐ". Table 4.1 and table 4.2 shows the mapping of "አ" and "ዐ".

Table 4.2 Mapping of the lone vowel "አ" as a vowel [17]

KEYS	[uU]	[iI]	[iI] [eE]	[eE]	[oO]
VOWEL	ኡ	ኢ	ኤ	እ	ኦ

Table 4.3 Mapping of the lone vowel "ዐ" as a vowel [17]

KEYS	[uU] [uU]	[iI] [iI]	[iI] [iI] [eE]	[eE] [eE]	[oO] [oO]
VOWEL	ዑ	ዒ	ዔ	ዕ	ዖ

Lone Vowels as Syllables: Here vowel syllables are treated like any other syllables and will be entered identically whereby the 6th order appears with the initial keystroke ("e" ⇒"እ") and ("ee" ⇒"ዕ") ,as shown in table 4.4, are the basis for composing all other orders.

Table 4.4 Mapping of the lone vowels "አ" and "ዐ" as Syllables [17]

KEY	+[eE]	+[uU]	+[iI]	+[aA]	+[iI][eE]		+[oO]
[eE]	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ
[eE] [eE]	ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ

Table 4.5 Composition of the Irregular Lone Vowels "ኧ" and "ኣ" [17]

KEY	[aA][eE]	[oO][aA]
VOWEL	ኧ	ኣ

The table above Table 4.5 shows the mapping of the irregular lone vowels “ኧ” and “ኣ” using a combination of vowels.

Punctuation

Some Western Punctuations have become common in modern writing practices. Therefore Punctuations use in Ethiopic writing practice find a modern and classical use. Table 4.6 and 4.7 shows the mapping of Modern Punctuation and Archaic Punctuation respectively.

Table 4.6 Modern Ethiopic punctuation to QWERTY symbol mapping [17]

PUNCTUATION	KEYS	NOTE
፡	:	As per the Principle of Graphical Continuity.
።	::	As per the Principle of Graphical Continuity.
፣	,	As per the Principle of Continuity of Function.
፥	,,	As per the Principle of Continuity of Function.
,	,,,	ASCII comma as per the Principle of Utility.
፤	;	As per the Principle of Continuity of Function.
;	;;	ASCII semicolon as per the Principle of Utility.
፦	:-	As per the Principle of Graphical Continuity.
:	:::	ASCII colon as per the Principle of Utility.
‥	_	Low line (aka "Underscore") is the selected symbol mapping due to the availability of the symbol and its virtual non-use in Ethiopic writing.
_	__	Two ASCII low lines as per the Principle of Utility.
'	''	Two ASCII apostrophes as per the Principle of Utility.

Table 4.7 Archaic Ethiopic punctuation to QWERTY symbol mapping [17]

PUNCTUATION	KEYS	NOTE
፣	:,	Applied for continuity with transliteration.
፠	:# :::	As per the Principle of Graphical Continuity. This mapping also represent key-strike continuity from similar dot based Ethiopic Full Stop (።).
※	:+ ::::	As per the Principle of Graphical Continuity. This mapping also represent key-strike continuity from similar dot based Ethiopic Paragraph Separator (፨).
:	:::::	ASCII colon sign as per the Principle of Utility.
⁝	:? ?	As per the Principle of Continuity of Function.
?	??	ASCII question mark as per the Principle of Utility.

Numerals

Under modern Ethiopic writing the Arabic digits, 0-9, are the default numeral system in use. Hence, the Arabic digits will be the default for an Ethiopic input method and the Ethiopic numerals will be entered by a context change marker or a modifier followed by the appropriate value given in the native numeral system of the keyboard device. For a QWERTY keyboard the context change marker is apostrophe and is chosen for its easy to type position and rare use in Ethiopic practices.

Minimal Support Level: - At the minimal support level direct mappings from Arabic digits onto a single Ethiopic numeral character is required as presented in table 4.8

Table 4.8 the Ethiopic numeral minimal support level mapping [17]

VALUE	1	2	3	4	5	6	7	8	9
KEYS	'1	'2	'3	'4	'5	'6	'7	'8	'9
NUMERAL	፩	፪	፫	፬	፭	፮	፯	፰	፱
VALUE	10	20	30	40	50	60	70	80	90
KEYS	'10	'20	'30	'40	'50	'60	'70	'80	'90
NUMERAL	፲	፳	፴	፵	፶	፷	፸	፹	፺

VALUE	100	1,000	10,000
KEYS	'100	'1000	'10000
NUMERAL	፻	፲፻	፼

Tonal Marks

Mappings for the Ethiopic Tonal Marks are only applicable to the Ge'ez and "Unified Ethiopic" input methods. It is shown in the following table 4.9.

Table 4.9 Ethiopic Tonal Marks mapping [17]

KEYS	_1	_2	_3	_4	_5	_6	_7	_8	_9	_0
SYMBOL	᎐	᎑	᎒	᎓	᎔	᎕	᎖	᎗	᎘	᎙

4.3 Architecture of the System

As shown on Figure 4.1 blow the phonetic Ethiopic input method designed for android smart phones have four main System componet, the key Event manager, keymapper, layout/mode manager, and the application interface.

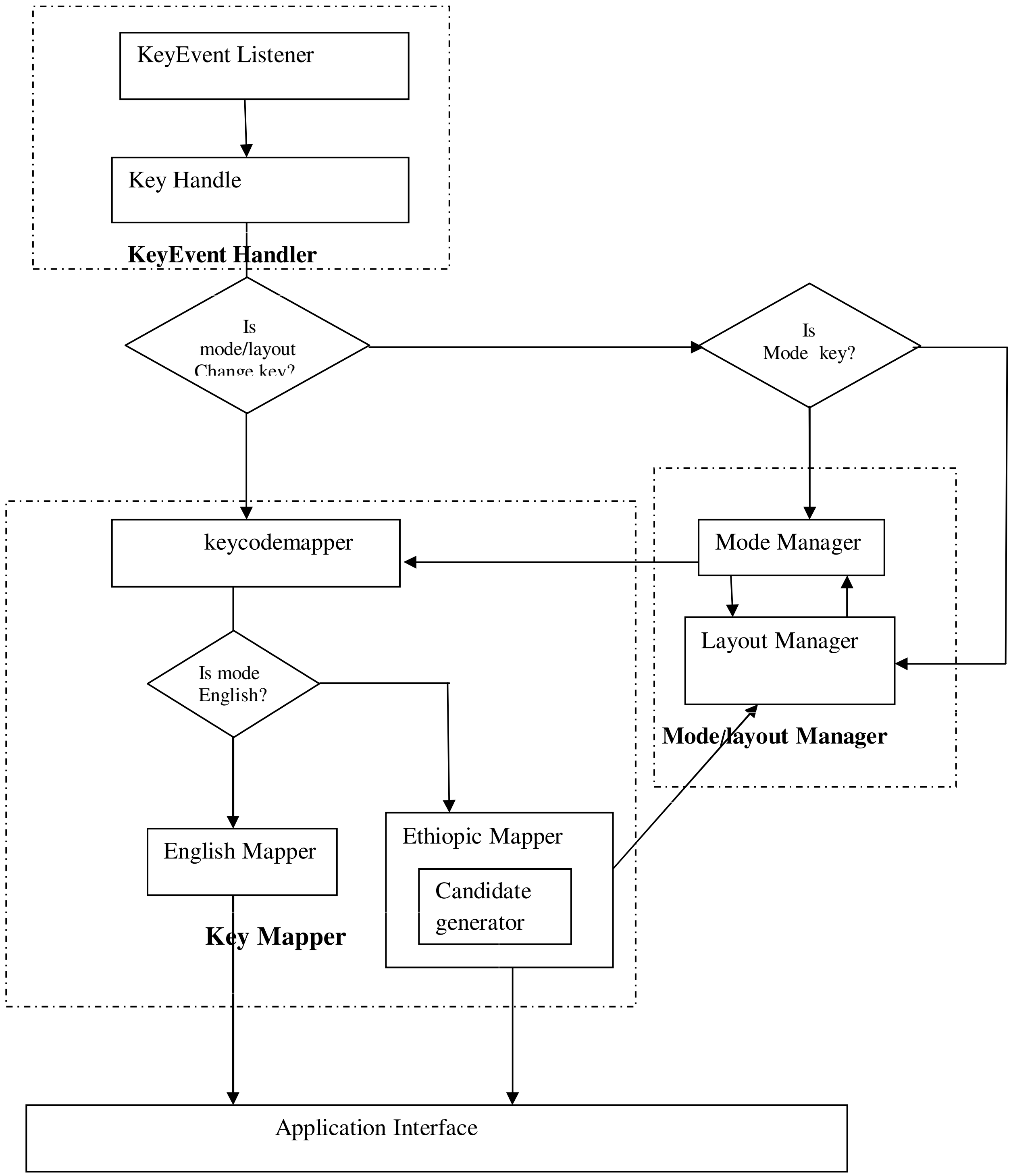

Figure 4.1 The proposed system Architecture of Ethiopic input system

The key event is reponsible for listening keyevents and generate the corresponding keycode.it has two sub parts the key event listener, which is repnosible for listening key press events and generate the corressponding keycode . And the key handler checks the type of the key press and passes the key to the next system componet according to its type. If the key is layout/mode changer it is passed to the layout/mode manager else it will be passed to the keymapper.

The key mapper's task is to map the keycode to its corresponding unicode. The key mapper saves the current inputmode state and either Ethiopic mapper or English mapper is used based on the state. English mapper is simple, it make a one -to-one mapping of the keycode with the unicode. But the Ethiopic mapper is a bit complex and the mappings are made based on the Ethiopic mapping specification which was pressented in the above section. The Ethiopic mapper has the candidate generator subpart, which generates the possible candidate forms which are related to the pressed key.

The layout/mode manager has two sub parts which are highly interrelated with each other. These are the layout manger and the mode manager. Layout magnager is responsible for displaying the softkeyboard layout and updating it when needed. The mode manager tracks the current mode and notifies the change to the keymaper.

The Application interface is the interface between the application and the ethiopic input method. It is mainly used to send the mapped unicode to the application program which is currently in use.

A layout key – is a key that is used to modify the current displayed layout. these keys are deligated keys that are found on the softkeyboard.

"123" key-change the softkeyboard layout to English number and punctuation layout.

"፩፪፫" key- change the softkeyboard layout to Ethiopic number and punctuation layout.

"abc" key- change the softkeyboard layout to English alphabet

"ሀለሐ" key – change the softkeyboard layout to Ethiopic alphabet.

An input mode key- is a key that is used to change the current input mode for the hard keyboard.

There are only two types of input mode the Ethiopic and English input mode.

"ALT+Shift+ E" hard key combination will be used to toggle between the input modes.

To summerize, the user key press event is the main input of this system. When a key (soft/hard key) is pressed by a user the keypress event is listend by the Key event listener and this key event listener will generate the coresponding keycode and pass it to the key handeler component, which check the key type, if the pressed key is a layout changer keycode the key is passed to the layout manager or else if the pressed key is mode changer key, the keycode is passed to the mode manager. Since the layout and the mode are interrelated with each other the change in one will update the state of the other and also the change will be notifed to the key mapper. On the other case if the pressed key is not a mode/layout changer key then it is passed to the keymapper. Here the key mapper will check its current input mode and pass it either to the ethiopic mapper or to the english mapper. At the mappers the key code will be mapped to the the approprate unicode character and the unicode character code is passed to the application interface the application interface will send the unicode to the application.

4.4 Proposed Layout for Ethiopic soft keyboard

There are two main visual elements in the Proposed Layout of the Ethiopic soft keyboard. The input view and the candidate view. The proposed layout is based on compact QWERTY layout for mobile phones as shown in Figure 4.2.

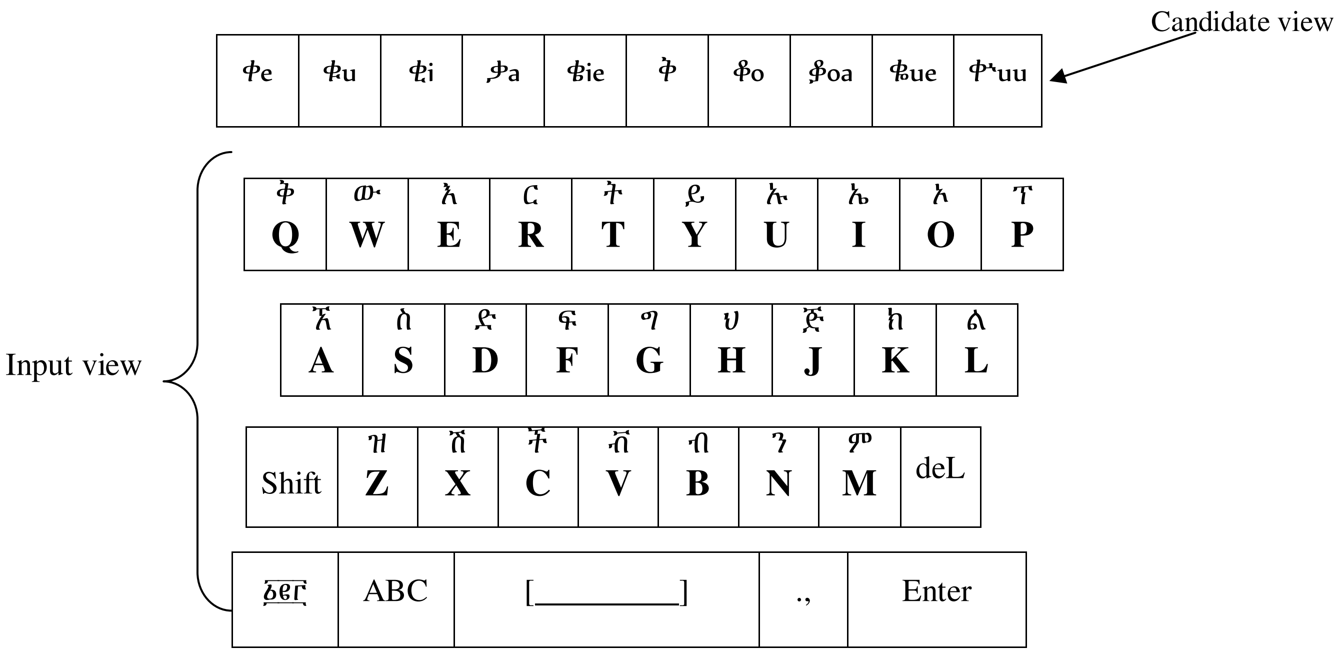

Figure 4.2 The Proposed Layout for Ethiopic soft keyboard

Input View

This is where the user can input character in the form of soft key presses.these view is used for softekey input only

***Candidates* View**

This is where the possible forms of the Ethiopic character are suggested to the user. Candidates are displayed when the inputmode is Amharic and a key is pressed. Incase of a hard keyboard the candidate view will only be displayed.

The layout is designed based on the QWERTY English keyboard and the Ethiopic keymapping specification. Please refere Appendix A.

CHAPTER FIVE

IMPLEMENTATION

The design of the proposed system is successfully implemented for android platform. This chapter will present the system development environment, the development tools and the developed system.

5.1 System development Environment

For the implementation of the systems Eclipse editor with android development tool plugin is used. Android provide a built-in input framework that facilitates the development of a new input method.

5.1.1 Android Input Method Frame work (IMF)

The IMF arbitrate the interaction between applications and the current input method chosen by the user. The motivation behind this framework is the realization that as Android matures, more hardware/software devices, and input techniques will appear in user's applications like the virtual keyboards, voice recognition, hand writing etc…[22].

The IMF is aware of the available hardware and its current state. If there is no a readily available hardware keyboard, an input method editor (IME) will be made available to the user when they tap on an enabled EditText. To create an input method for entering text into text fields and other Views, We need to extend theInputMethod Service Class. This class provides much of the basic implementation for an input method, in terms of managing the state and visibility of the input method and communicating with the currently visible activity [22, 23]. The typical life-cycle of an InputMethodService is described in the following figure 5.1.

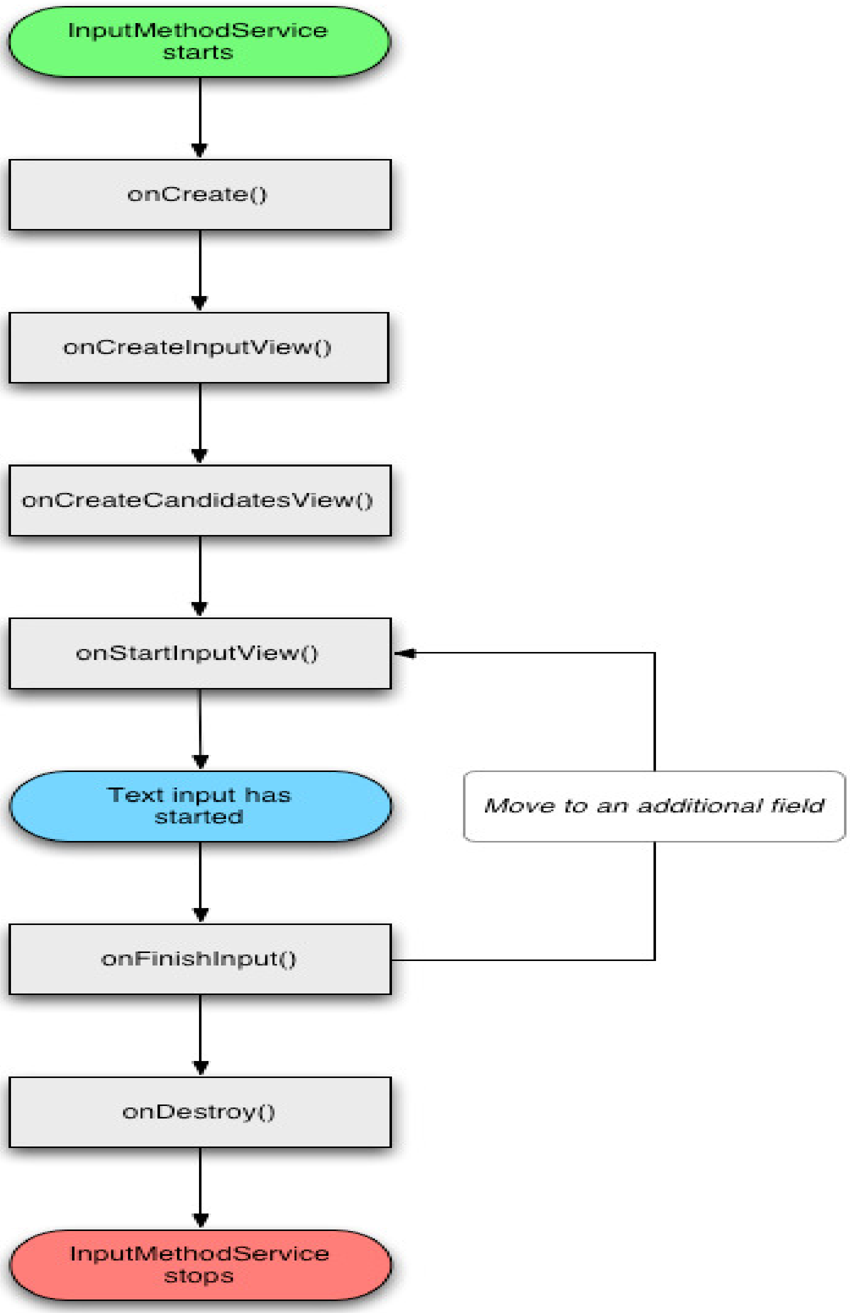

Figure 5.1 - Life-cycle of an InputMethodService [23]

There are two main visual elements for an input method. The input view and the candidates view.

The ***Input View*** is where the user can input text either in the form of hard or soft key presses, handwriting or other gestures. When the input method is displayed for the first time, *InputMethodService.onCreateInputView()* will be called. Create and return the view hierarchy that you would like to display in the input method window.

The ***Candidates View*** is where potential word/character corrections or completions are presented to the user for selection. *InputMethodService.onCreatet*Candidate*View()* is called to create the candidate view.

Composing text before committing

Since Ethiopic input method requires multiple steps to compose a character, the progress is shown in the text field until the user commits it, then the replace the partial Composition (intermediate character) with the correct character. The character that is being composed will be highlighted in the text field in some fashion, such as an underline. The following figure 5.2a and 5.2b illustrate the composing state for ‘m’ types “ም” and when ‘e’ is typed it changes the composing state of ‘ም’ to ‘መ’ which will be commited because there is no other form of ‘መ’ that is be typed by extending “me ”.

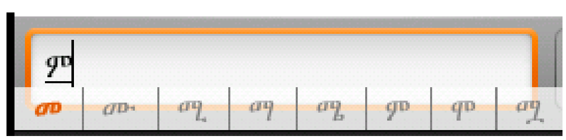

Figure 5.2a - Composing state for ‘ም’

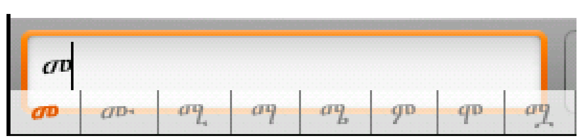

Figure 5.2b - Committed state of ‘መ’

5.2 Development Tools

The Android includes custom tools that help to develop mobile applications on an Android platform. The most important of these are the Android Emulator and the Android Development Tools plug-in for Eclipse. Eclipse java editor is also used.

5.2.1 The Android Emulator

The Android SDK includes a mobile device emulator which is a virtual mobile device that runs on our computer. The emulator lets us prototype, develop, and test Android applications without using a physical device. The Android emulator mimics all of the typical hardware and software features of a typical mobile device, except that it cannot receive or place actual phone calls. It provides a variety of navigation and control keys, which we can "press" using your mouse or keyboard to generate events for your application. It also provides a screen in which your application is displayed, together with any other Android applications running [24].see Figure 5.3.

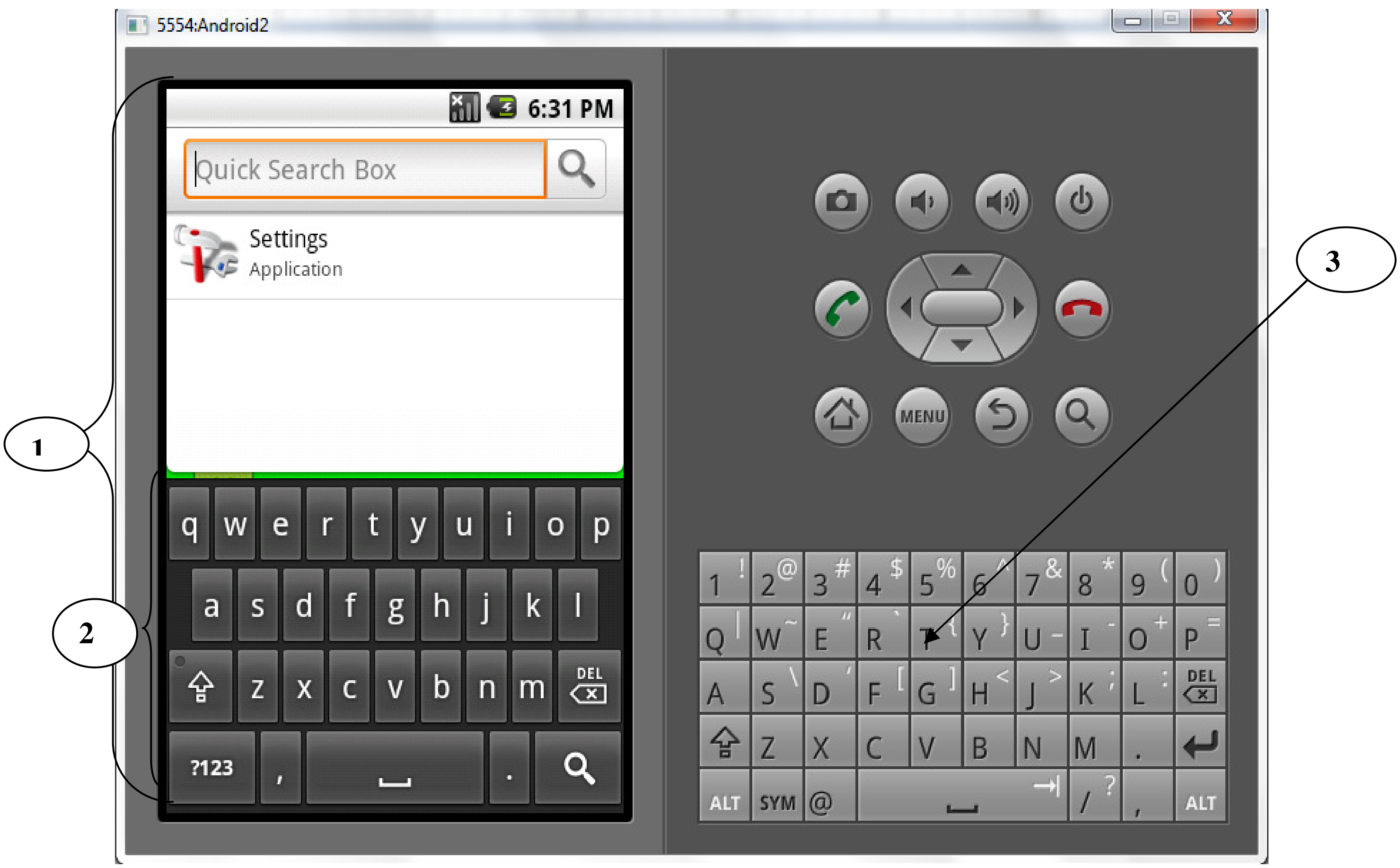

Figure 5.3 - Android Emulator

Reference

1. Android Emulator screen area
2. Android soft keyboard
3. Android hard keyboard

5.2.2 Android Development Tools (ADT) plug-in for Eclipse

The ADT plug-in adds powerful extensions to the Eclipse integrated environment, making creating and debugging Android applications easier and faster. If Eclipse is used as a development environment, the ADT plug-in gives an enhancement for developing Android applications.

5.2.3 Eclipse

Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. Hence Eclipse for RCP/Plug-in Developers is used with android plug-in to code the system.

5.3 Yeshi- Ethiopic input Method

The implemented system is named as "Yeshi-Ethiopic input method" or"የሺ ኢትዮፒክ"in amharic.It is named in the memory of my aunt.

The implementation of the system design makes it possible to type using Ethiopic on the built in applications. Users can send an Ethiopic SMS-message, contact can be saved using Amharic, Tigrigna or in any other language that is using Ethiopic scripts. From now on android users will enjoy the Ethiopic scripts without a problem.

To use the Yeshi-Ethiopic input method users must first install it on their device. After installing the software onto the Android Emulator or Android smart phone, Yeshi - Ethiopic will be add to the OS as one type of the input method option. Then users must first add the "Yeshi Ethiopic" from menu->Setting->language & keyboards. In order to added it to the input method options list as shown in the Figure 5.4.

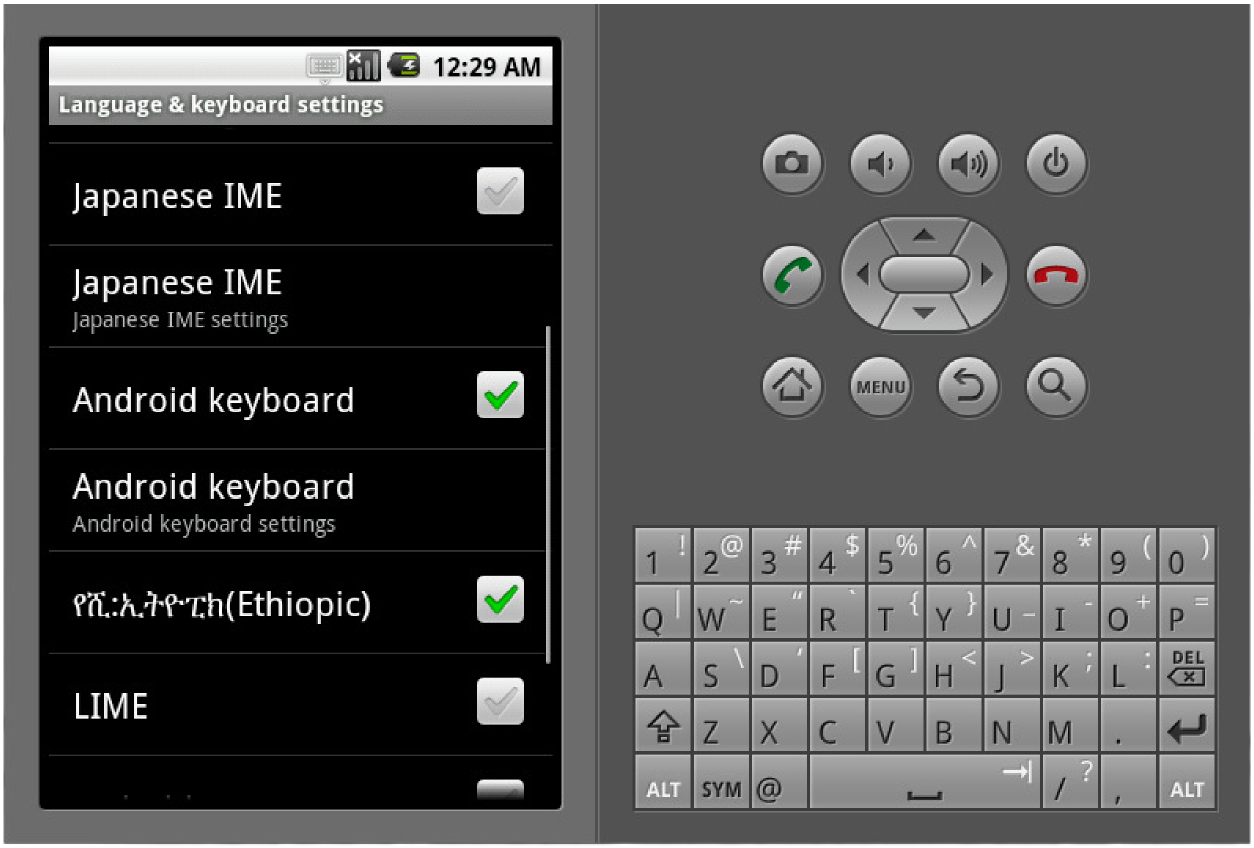

Figure 5.4 - Activating Yeshi- Ethiopic input method

Then when the user wants to use our input method he/she must select the Yeshi-Ethiopic from the input method list to make it the current input method. To get the input method list click and hold on the text area and the input method will pop up as shown in Figure 5.5.

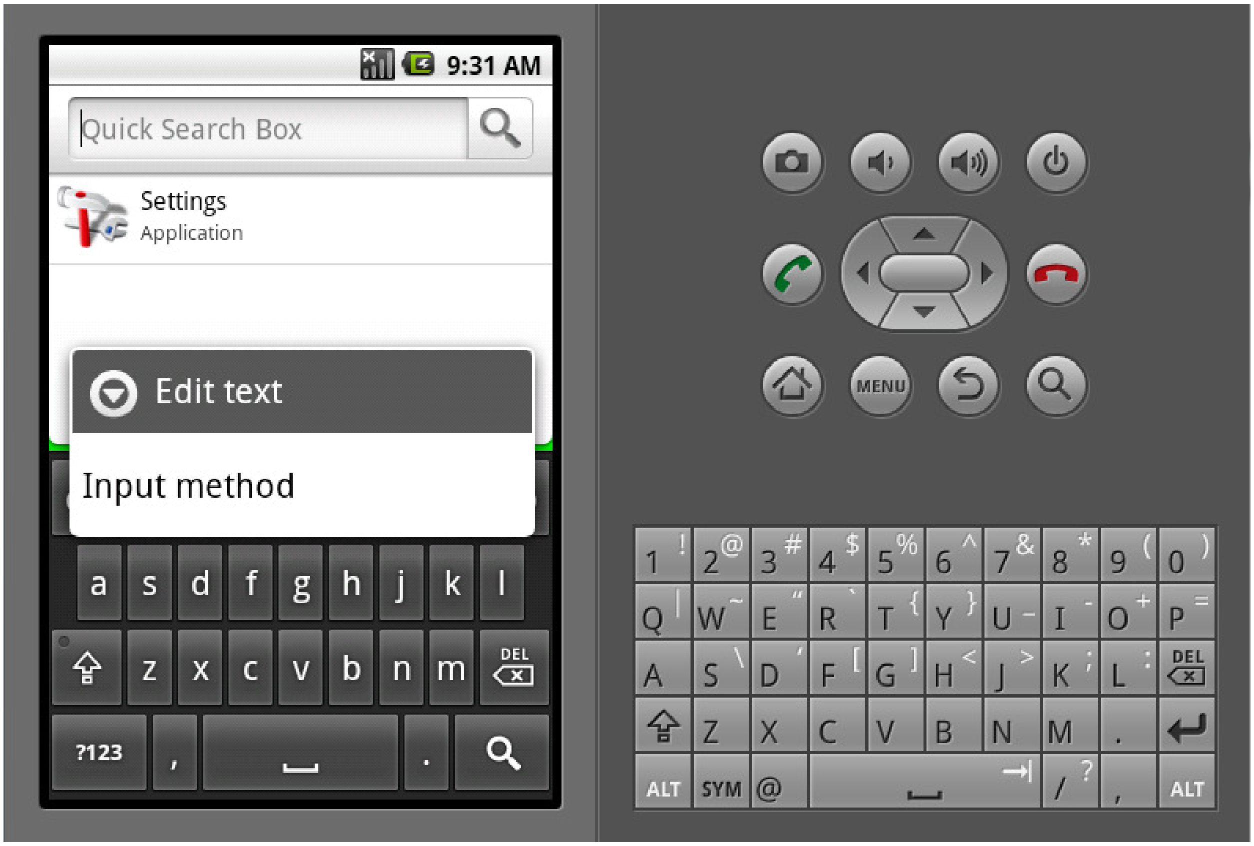

Figure 5.5 - Changing the input method

User gets the available input method lists. He/she make sure that Yeshi-Ethiopic is selected from the available lists see Figure 5.6.

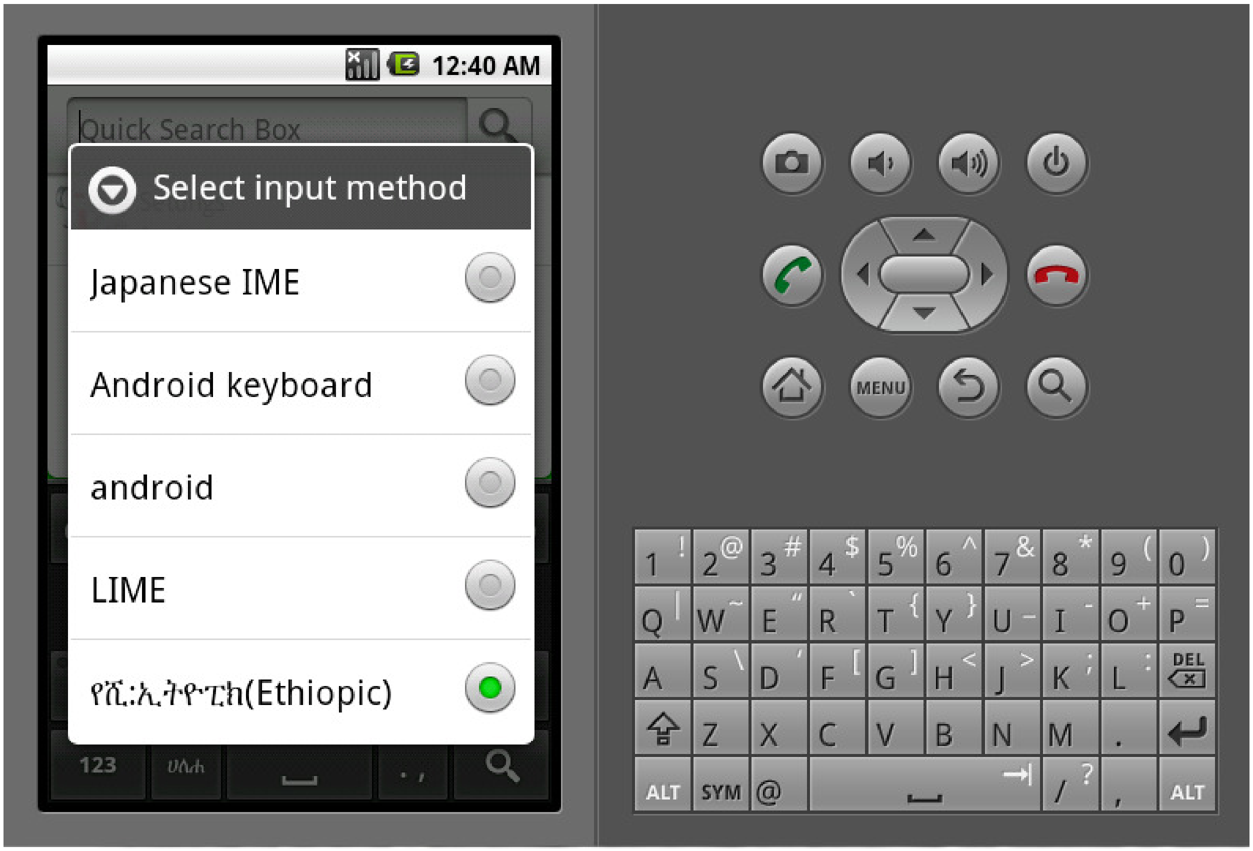

Figure 5.6 - Available input method list

Now the current input method is changed to Yeshi- Ethiopic. users can type Ethiopic character using either using soft/hard keys of the system. Figure 5.7, 5.8, 5.9 shows the Ethiopic “ሀለሐ”, “፩፪፫” And the English “abc” and “123” layouts of the soft keyboard.

Figure 5.7 - Yeshi- Ethiopic “ሀለሐ” layout of the soft keyboard

In the above layout to change the keyboard layout user should use “፩፪፫” and the “ABC” key to change the keyboard to Ethiopic number and Punctuation and English alphabet respectively.

User can also use the Shortcut cut Hard key “alt +shift+e” to toggle the input mode between Ethiopic and English.

If the user presses “፩፪፫” the layout is changed to Ethiopic number and Punctuation as shown in Figure 5.8.

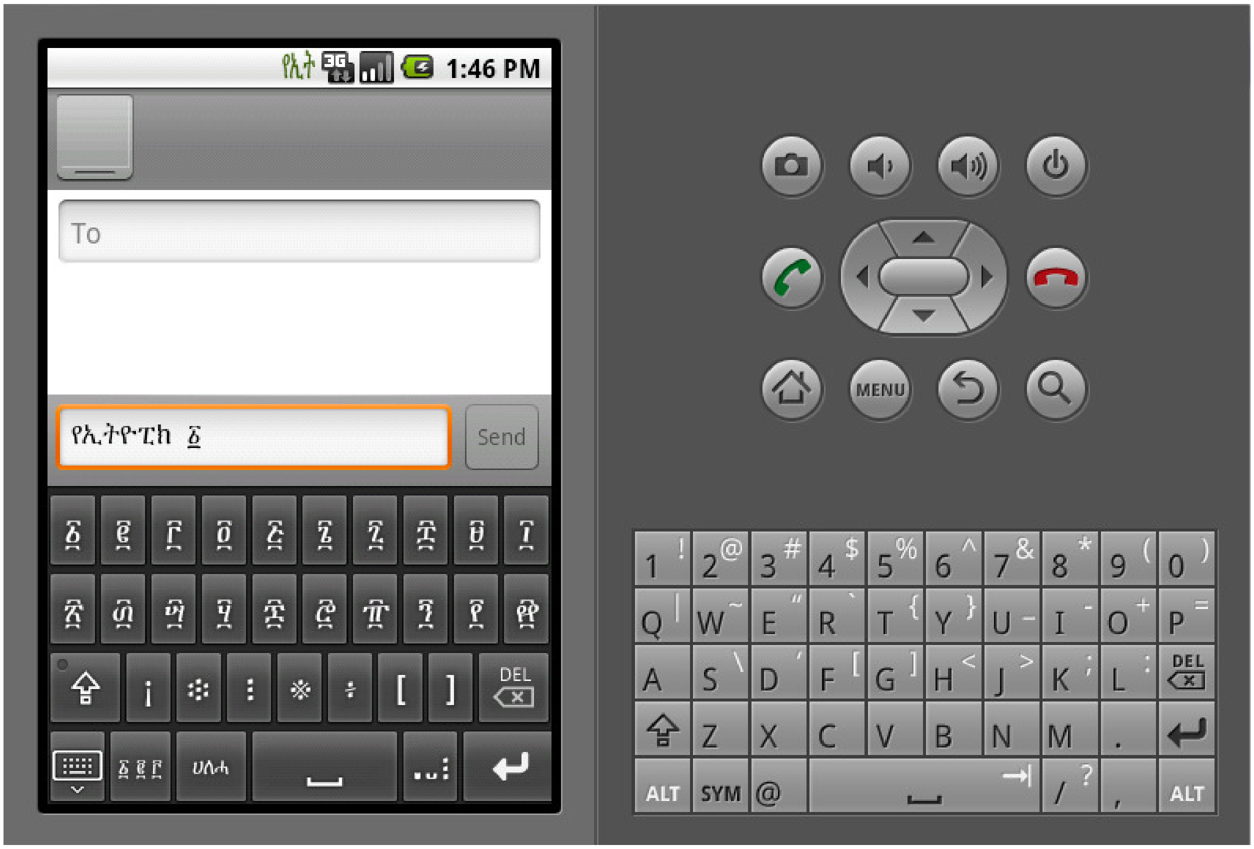

Figure 5.8 - Yeshi- Ethiopic “፩፪፫” layout of the soft keyboard

If the user has pressed “abc” the compacted Latin QWERTY layout is load. See Figure 5.9.

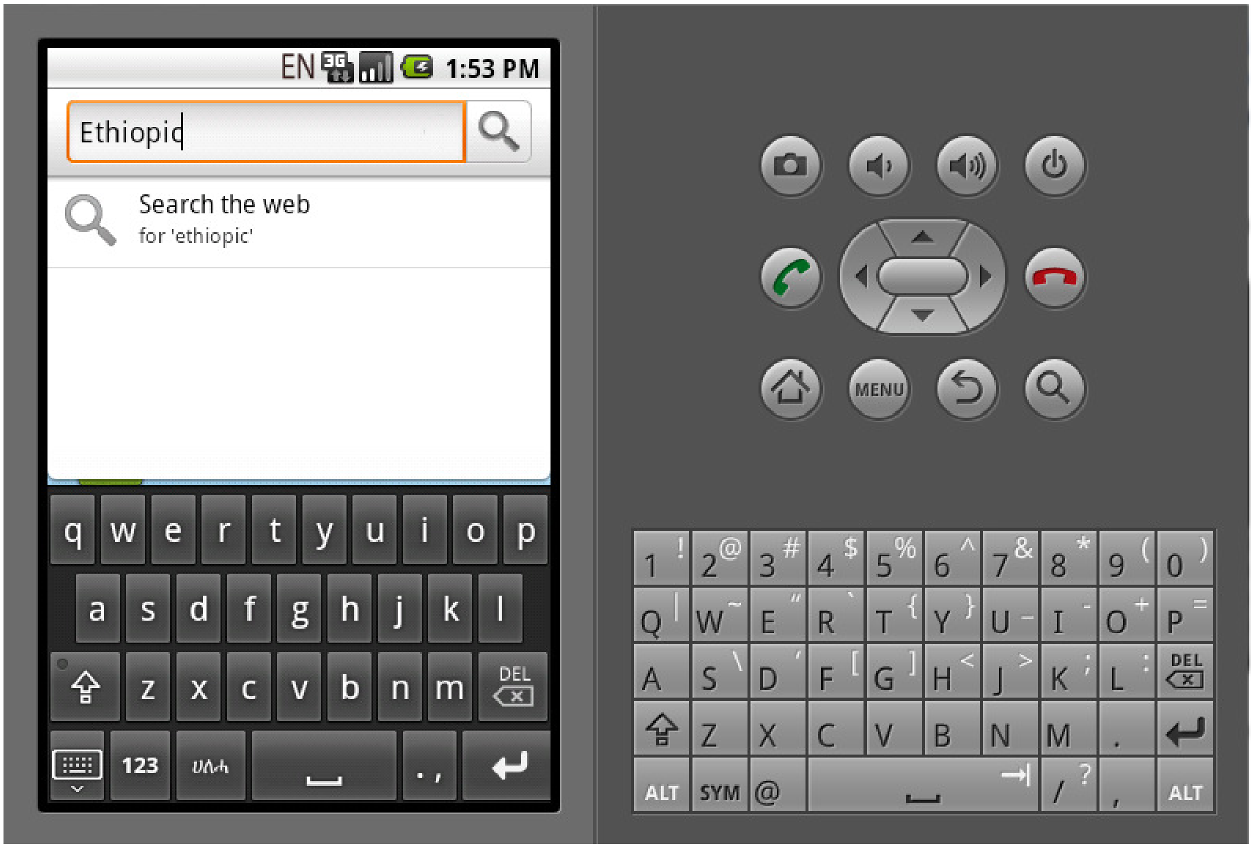

Figure 5.9 - Yeshi- Ethiopic “abc” Latin layout of the soft keyboard

User must change the layout to English number and Punctuation in order to type Arabic numbers and English punctuation symbols as shown in figure 5.10.

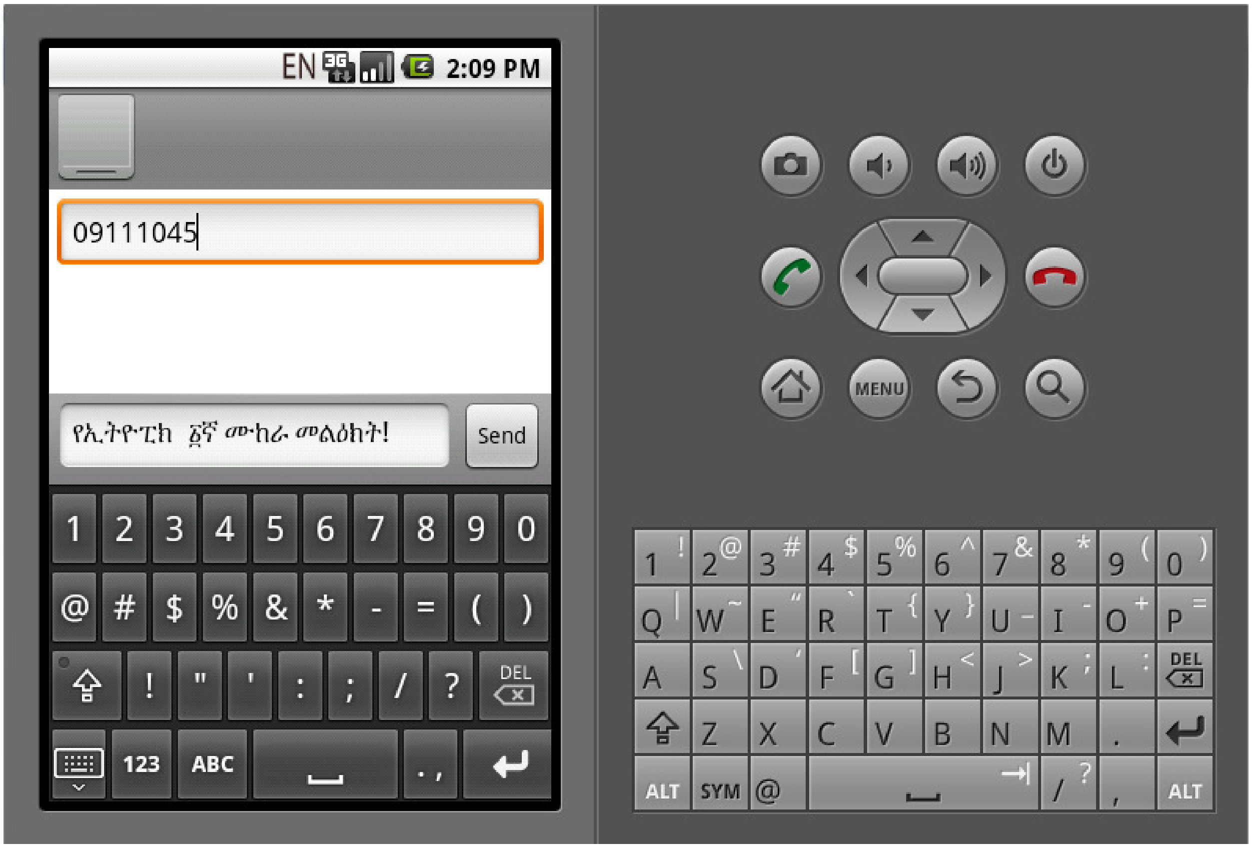

Figure 5.10 -Yeshi- Ethiopic “123” Latin layout of the soft keyboard

5.4 Testing

Yeshi- Ethiopic input method is tested using the native android applications. The contacts and messages applications are mainly used to test the integration of the system with other applications. And it was possible to successfully enter Ethiopic scripts on these applications using the implemented system. Please see the following figure 5.11 and figure 5.12.

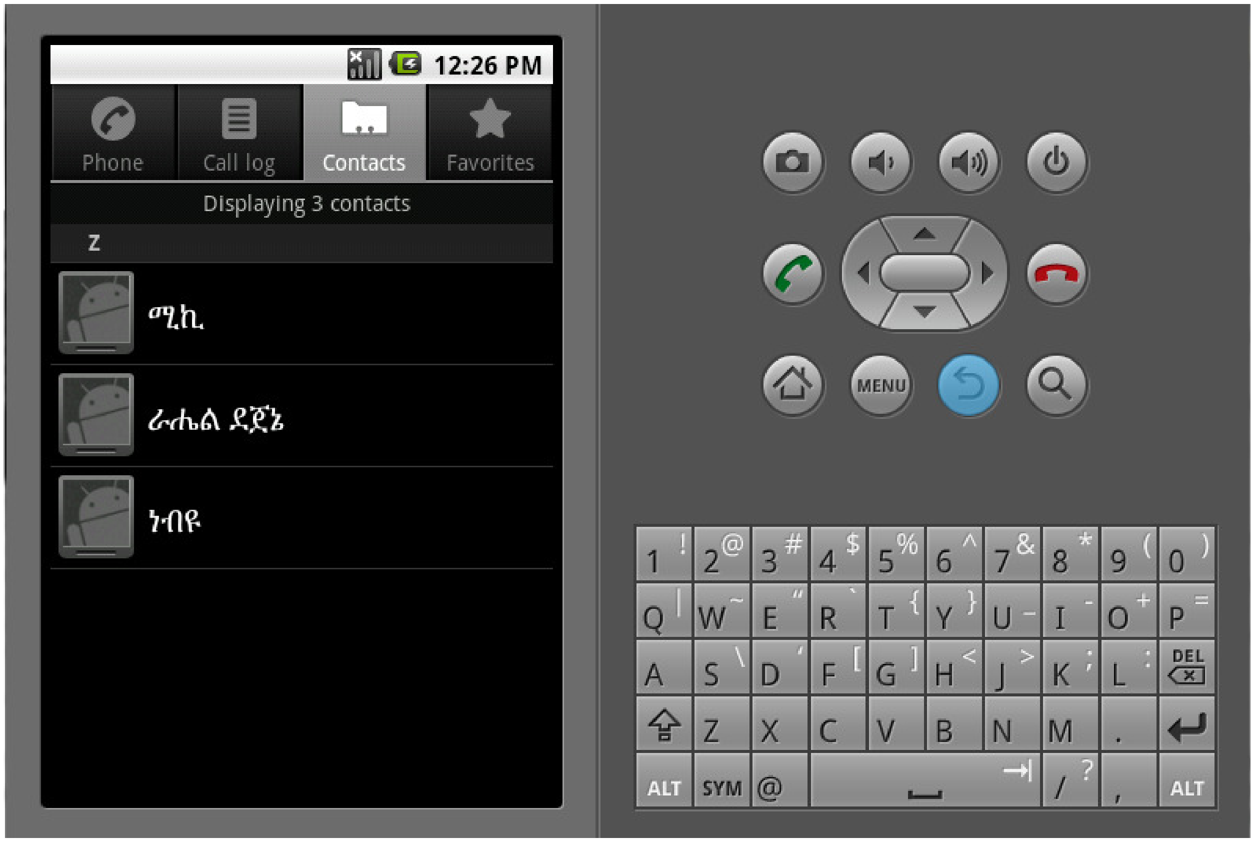

Figure 5.11 - Saved contact using the Ethiopic script.

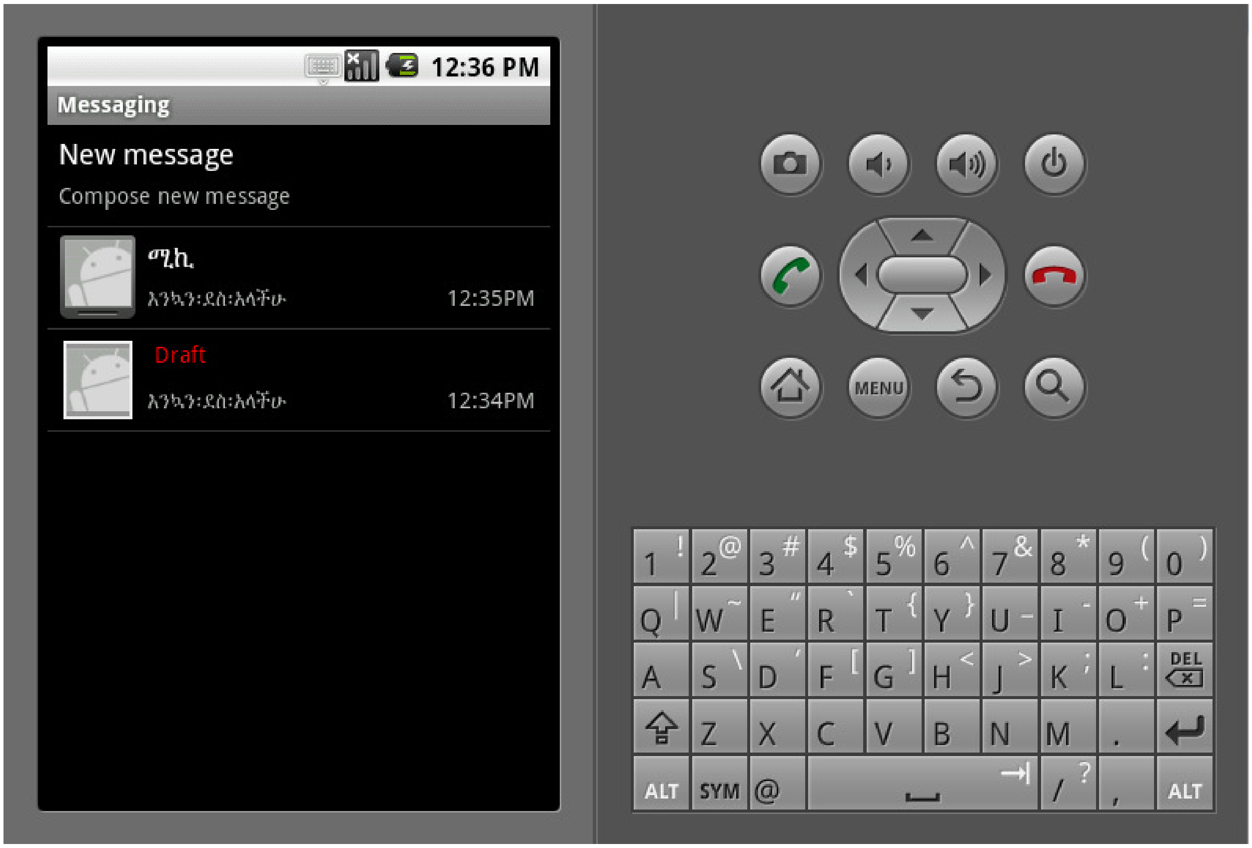

Figure 5.12 - Sent and draft messages using the Ethiopic script.

Chapter Six

Conclusion and Recommendation

The design and implementation of an Ethiopic input method is a complex task due to the huge number of the Ethiopic characters set. Furthermore, trying to develop a language neutral input method for small handled device that have a reduced screen size and hard keyboard adds to the existing complexity. Though complex and difficult we implemented a language neutral Ethiopic input method for android based smart phone based on Ge'ez Frontier Foundation specification. However, it can be observed that there is much work to be done towards standardization of the Ethiopic input method.

The contribution of this work can be listed as follows:

- Contributed a review of the Ethiopic input methods and available standards.
- Analyze the requirements for Ethiopic input method.
- Design a phonetic input method for Ethiopic scripts. The design is made in such a way that it is language independent with respect to languages that use Ethiopic scripts.
- The design of Ethiopic input method is made for both soft and hard keypads. Switching between Ethiopic and QWERTY keyboards has been simple.
- Implemented the designed system for use on android based smart phones.
- Tested the usability of the developed input method for different applications on android phone platforms.

And for the future this system can be extend as listed below:

- It should be extended to support the available standard keyboard layouts other than the QWERTY.(such as AZERTY, Arabic, etc)
- A word prediction could be incorporated in order to facilitate a better and faster text entry.
- A language dependent input method will definitely ease the text entry method.

At last we recommend to other developers to develop similar software for the other mobile platforms so that it would be possible to type and exchange text information using the Ethiopic scripts.

References

[1]. Gedion Tamene, "Design and Implementation of virtual Keyboard for Amharic text entry on PDA system", MSc Project , Addis Ababa University department of Computer Science, June 2008.

[2]. Shiferaw Abebe, Solomon Atnafu, Tewodros Seyum, Samuel Kinde Kassegne, "Ethiopic Keyboard Mapping and Predictive Text Inputting Algorithm in a Wireless Environment", Proceedings of International Symposium on ICT Education and Application in Developing Countries(ICTES2004), Addis Ababa, Ethiopia, October 2004.

[3]. Samuel Kinde Kassegne, Ted Kidane, Mieso K. Denko, "Mobile text and instant messaging solutions in non-latin scripts for commodity price information management in emerging markets", proceedings of international symposium on wireless communications and information technology in developing countries (WCITD2008), Pretoria, South Africa, OCTOBER 2008.

[4]. Abera Abebaw, "Implementation of online handwriting recognition system for Ethiopic character set", MSc project, Addis Ababa University department of Computer Science, May 2007.

[5]. Daniel Kefale, "Modeling multi-script text editing based on online handwriting recognition", MSc thesis, Addis Ababa University department of Computer Science, June 2008.

[6]. Android developers," Developer Announcements", http://developer.android.com, Last visited on June 6, 2010.

[7]. Android, "What is Android ?",http://www.android.com/about/, Last visited on June 6, 2010.

[8]. Tarmo Virki, "Google beats Microsoft in smartphones", http://www.reuters.com/article/idUSLDE64I03620100519/, Last visited on June 6, 2010.

[9]. Peter Kafka, " Is Android Really Outselling Apple?", http://mediamemo.allthingsd.com/20100510/is-android-really-outselling-apple, Last visited on June 6, 2010

[10]. Donald Melanso, NPD: Android ousts iPhone OS for second place in US smartphone market,http://www.engadget.com/2010/05/10/npd-android-ousts-iphone-os-for-second-place-in-us-smartphone-m/, Last visited on June 6, 2010.

[11]. Open Handset Alliance, "What would it take to build a better mobile phone?", http://www.openhandsetalliance.com/, Last visited on June 6, 2010.

[12]. Aarti R, "Different Types of Keyboards", http://www.buzzle.com/articles/different-types-of-keyboards.html, Last visited on June 6, 2010.

[13]. Martin Hosken, "An introduction to keyboard design theory: What goes where?" , http://scripts.sil.org/KeybrdDesign., Last visited on June 6, 2010.

[14]. The Unicode standard version 5.2 , page [116-122]&[281 – 282], http://www.unicode.org/charts/PDF/U1200.pdf, Last visited on June 6, 2010.

[15]. Omniglot, "*Ge'ez* (Ethiopic) syllabic script and the Amharic language", www.omniglot.com/writing/ethiopic.htm, Last visited on June 6, 2010.

[16]. QSAE ES3449:2008, "Ethiopic keyboard layout", www.eictda.gov.et/Downloads/.../ES%203449_2008%20keyboard.pdf, Last visited on June 6, 2010.

[17]. Ge'ez Frontier Foundation ,"Principles and Specification for Mnemonic Ethiopic Keyboards", http://keyboards.ethiopic.org/specification/GFF-MnemonicEthiopicKeyboardSpecification.pdf, Last visited on June 6, 2010.

[18]. KM software, "Amharic SMS Sender for your phone", http://www.mobango.com/swarea/index.php/md/24637?area=apps&standardFilter=top_rated&word=nokia&idSubcategory=508&title=Amharic+SMS+Sender&listNum=1, Last visited on June 6, 2010.

[19]. GeezSMS, "Introducing - Geez SMS V1.0", http://www.geezsms.com/, Last visited on June 6, 2010.

[20]. Elizabeth Blunt," Ethiopians get texting in Amharic", http://news.bbc.co.uk/2/hi/africa/7138128.stm, Last visited on June 6, 2010.

[21]. Elena Balan, “Nokia Brings Amharic Keypad to Mobile Phones”, http://news.softpedia.com/news/Nokia-Brings-Amharic-Keypad-to-Mobile-Phones-66590.shtml, Last visited on June 6, 2010.

[22]. Android developers, “Input method Manager”, http://developer.android.com/reference/android/view/inputmethod/InputMethodManager.html, Last visited on June 6, 2010.

[23]. Android developers, “Creating an Input Method”, http://developer.android.com/resources/articles/creating-input-method.html, Last visited on June 6, 2010.

[34]. Android developers, “android emulator”, http://developer.android.com/guide/developing/tools/emulator.html, Last visited on June 6, 2010.

Appendix A: Ethiopic (Language Neutral) Input Method

	IPA	+^{w}	+u	+i	+a	+e		+o	+^{w}	+^{w}ə	+^{w}u	+^{w}i	+^{w}a	+^{w}e	+ya
IPA	**KEY**	**+[eE]**	**+[uU]**	**+[iI]**	**+[aA]**	**+[iI][eE]**		**+[oO]**	**+[oO][aA]**	**+[uU][eE]**	**+[uU][uU]**	**+[uU][iI]**	**+[uU][aA]**	**+[uU][iI][eE]**	**+Y[aA]**
h	**h**	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ	ሇ	ኈ	ኍ	ኊ	ኋ	ኌ	
l	**[lL]**	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ	ⶀ				ሏ		
ħ	**H**	ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ					ሗ		
m	**[mM]**	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ	ⶁ	ᎀ	ᎃ	ᎁ	ሟ	ᎂ	ፙ
s	**ss**	ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ					ሧ		
r	**[rR]**	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ	ⶂ				ሯ		ፘ
s	**s**	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ	ⶃ				ሷ		
ʃ	**[xX]**	ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ	ⶄ				ሿ		
ʆ	**[xX][xX]**	ⶠ	ⶡ	ⶢ	ⶣ	ⶤ	ⶥ	ⶦ							
k’	**q**	ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቇ	ቈ	ቍ	ቊ	ቋ	ቌ	
ʁ	**Q**	ቐ	ቑ	ቒ	ቓ	ቔ	ቕ	ቖ		ቘ	ቝ	ቚ	ቛ	ቜ	
k'ʸ	**qY**	ⷀ	ⷁ	ⷂ	ⷃ	ⷄ	ⷅ	ⷆ							
b	**[bB]**	በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ	ⶅ	ᎄ	ᎇ	ᎅ	ቧ	ᎆ	
v	**[vV]**	ቨ	ቩ	ቪ	ቫ	ቬ	ቭ	ቮ					ቯ		
t	**t**	ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ	ⶆ				ቷ		
ʧ	**c**	ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ	ⶇ				ቿ		

ʧ	cc	ⶨ	ⶩ	ⶪ	ⶫ	ⶬ	ⶭ	ⶮ							
ŋ	hh	ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ	ኇ	ኈ	ኍ	ኊ	ኋ	ኌ	
n	n	ነ	ኑ	ኒ	ና	ኔ	ን	ኖ	ⶈ				ኗ		
ɲ	N	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ	ⶉ				ኟ		
ʔ	a	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ	ⶊ						
k	[kK]	ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኯ	ኰ	ኵ	ኲ	ኳ	ኴ	
kʸ	kY	ⷈ	ⷉ	ⷊ	ⷋ	ⷌ	ⷍ	ⷎ							
xʸ	KY	ⷐ	ⷑ	ⷒ	ⷓ	ⷔ	ⷕ	ⷖ							
w	[w W]	ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ	ዏ						
ʕ	A	ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ							
z	z	ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ	ⶋ				ዟ		
ʒ	Z	ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ					ዧ		
ʐ	ZZ	ⶰ	ⶱ	ⶲ	ⶳ	ⶴ	ⶵ	ⶶ							
y	[yY]	የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ	ዯ						
d	d	ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ	ⶌ				ዷ		
d’	D	ዸ	ዹ	ዺ	ዻ	ዼ	ዽ	ዾ	ⶍ				ዿ		
ʤ	[jJ]	ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ	ⶎ				ጇ		
g	g	ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጏ	ጐ	ጕ	ጒ	ጓ	ጔ	
ŋ	G	ጘ	ጙ	ጚ	ጛ	ጜ	ጝ	ጞ		ⶓ	ⶖ	ⶔ	ጟ	ⶕ	
gʸ	gY	ⷘ	ⷙ	ⷚ	ⷛ	ⷜ	ⷝ	ⷞ							
t’	T	ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ	ⶏ				ጧ		
ʧ’	C	ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ	ⶐ				ጯ		
ʧ’	CC	ⶸ	ⶹ	ⶺ	ⶻ	ⶼ	ⶽ	ⶾ							
p’	P	ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ	ⶑ				ጷ		

s’	**S**	ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ					ጿ		
s’	**SS**	ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ	ፇ						
f	**[fF]**	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ		ᎇ	ᎈ	ᎉ	ፏ	[illegible]	ፚ
p	**p**	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ	ፗ	ᎊ	ᎋ	ᎌ	ፗ	[illegible]	
ə	**ae**	ኧ													

Declaration

I, the undersigned, declare that this project is my original work and has not been presented for a degree in any other university, and that all source of materials used for the project have been duly acknowledged.

Declared by: Name: ____

Signature: ____

Date: ____

Confirmed by advisor: Name: ______________________________________

Signature: ____

Date: ____