ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

Perpetual Ethiopic and European Calendar and Organizer System for an Android Based Smart Phones

By

Zenebe Nigussie

A Project paper 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

Perpetual Ethiopic and European Calendar and Organizer System for an Android Based Smart Phones

By:

Zenebe Nigussie

Advisor:

Solomon Atnafu (PhD)

Examining Board:

1. Dr. Solomon Atnafu, Advisor __________________________

2. __________________________ __________________________

3. __________________________ __________________________

Acknowledgement

First and foremost I offer my sincerest gratitude to my advisor, Dr Solomon Atnafu, who has supported me throughout my project with his patience and knowledge. He was abundantly helpful and offered invaluable advice, support and guidance and without him this project, too, would not have been completed or written.

My special heartily thankful is to my wife, Mintiwab Mullugeta, whose encouragement and unreserved support helped me to complete my work successfully. She has great place for my success in all my life. My special thanks also address my sweet daughter Abigail, who came to this world at the eve of the completion of my project.

I also would like to extend my warm brotherly feelings of gratitude to my best friend Getch, who has been with me through out all the times of ups and downs. Specially, he has been supporting me with the entire necessary undertaking in my projects.

I also would like to extend my special thanks to my Father Nigussie Alemu, my Mother Yimenashu Gebeyehu, my brother Feleke and my sisters Misgana , Zinash and Mesay who have helped and encouraged me in all the conditions I am in.

Lastly, but not least, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project.

Above all, the Almighty God has been with me in all situations for my dream to come true. Hence, Glory and Praise be for Him forever.

Table of Contents

List of Tables

List of Figures

Abbreviations/Acronyms

AAPT	Android Asset Packaging Tool
ADB	Android Debug Bridge
ADT	Android Development Tools
API	Application Program Interface
APK	Android Package
ARM	Acorn RISC Machine
DDMS	Dalvik Debug Monitor Service
DEX	Dalvik Executable
GPS	Global Positioning System
IDE	Integrated Development Environment
JVM	Java Virtual Machine
OHA	Open Handset Alliance
OS	Operating systems
PC	Personal computer
RAD	Requirement Analysis Document
SDK	Software Development Kit
SQL	Structured Query Language
UML	Unified Modeling Language
VM	Virtual Machine

Abstract

In the growing world of technology, cellular devices have quickly emerged as one of the fastest evolving fields. They have increased greatly in both popularity and complexity, requiring more advanced operating systems and applications to meet the demands of the user. Android is a software stack designed to meet these demands in an open source environment. It includes an operating system, middleware, and key applications, as well as a Software Development Kit (SDK) for developers to create their own applications for the Android environment.

This project is targeted to develop a perpetual Ethiopian and Gregorian calendar system and organizer application that runs on the top of Android mobile phones. Perpetual calendars provide a date over a wide range of years. This project came up with the Ethiopian calendar system with the corresponding Gregorian calendar of any year. The calendar provides the date of the public holidays of National (non religious), Christian and Muslim holidays of any month and year the user prefers to know. In addition to this users are supported to organize their tasks based on the Ethiopian calendar for a specific period of time with Ethiopic script. Once the task is organized or scheduled the users are alarmed when the time of the schedule reached. Besides these, users are supported to keep diaries at any specific date of Ethiopian calendar using Ethiopian scripts. Lastly the system has been tested and the result shows that it works correctly 100%.

Keywords: Perpetual Ethiopian calendar, Ethiopian calendar, Ethiopian and Gregorian calendar, Ethiopic Organizer, Ethiopic Diary, Ethiopic Calendar on Android

CHAPTER ONE: INTRODUCTION

1.1 Overview

Mobile phones are becoming smarter and giving different services for its users beyond communication services due to the advancement in mobile device technologies. One of the advancement in mobile technology is the operating system, which makes mobile phones to run different applications and support diversified services. There are different operating systems for mobile devices. Of these the most popular once are Symbian, iPhone, RIM's BlackBerry, Window mobile, Linux, PalmwebOS. Recently Android OS by Google becomes popular and competent with the other operating systems.

Android is a software platform and operating system for mobile devices, based on the Linux kernel, developed by Google and later by the Open Handset Alliance. It allows developers to write code in the Java language, controlling the device via Google-developed Java libraries. The unveiling of the Android platform on 5 November 2007 was announced with the founding of the Open Handset Alliance, a consortium of different hardware, software and telecom companies devoted to advancing open standards for mobile devices. Open Handset Alliance includes several companies such as Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, Sprint Nextel and NVIDIA [21]. These companies aim to develop technologies that will significantly lower the cost of developing and distributing mobile devices and services. The Android platform is a fully integrated mobile "software stack" that consists of an operating system, middleware, user-friendly interface and applications.

Android is intended to revolutionize the mobile market by bringing the Internet to the cell phone and allowing its use in the same way as on the PC. Android was built from the ground up with the explicit goal to be the first open, complete, and free platform created specifically for mobile devices. The Google Android Operating System for cell phones and wireless mobile devices will allow cell phones to become any device or web appliance by providing services as a web browser, music player, web publisher, editor, instant messenger, camera, map, GPS device, mobile search engine and more [22]. Therefore, the software platform provides a foundation for a number of applications such as calendar just like a real working platform.

Due to the advancement of technology and economic growth in our country, many people have mobile phones. Currently, Android mobile phones are distributed over the market at large number that make many users to have the access to these mobile phones. Besides its communication service, mobile phones have a number of functions to provide for its users. Of these functions, calendar is one of the important applications available on the mobile phones. A calendar is a system of organizing days, months, and years for measuring, recording and keeping of time which can be used for social, religious, commercial, or administrative purposes. It is a system that provides us with the power to manage events throughout our life. Events ranging from holidays to meetings and from external deadlines to inspections can all be scheduled automatically to individual and master diaries, giving the complete picture at all times [4]. Based on the calendar, users can organize their activities and keep diary of a specific year, month and date. Diary is a permanent personal record that is kept of events, thoughts, and ideas associated with an individual. Keeping a diary is an excellent means of documenting experiences and ideas that will have meaning later in life or possibly be of importance to the next generation [5].

Though there are different types of calendar systems, the Gregorian calendar system is the most popular one. Ethiopia has its own ancient calendar system which is different fromh the Gregorian calendar system in terms of years, months and days. The Gregorian calendar system uses 12 months with varying number of days in each month. But the Ethiopian calendar has 12 months of 30 days each, and a 13^{th} month of 5 days or 6 days (in a leap year) at the end of the year [1].

One feature that is associated with Ethiopian calendar is public holidays. Most of the public holidays fall on the same date from year to year. But some of the religious holidays such as Ethiopian Good Friday, Easter, Ed Alfetir, Arefa and Mewulid fall on different dates and months from year to year. To determine the date and month of those religious holidays, it requires calculations based on mathematical techniques used by both religions [2].

1.2 Statement of the Problem

Most of the mobile phones available in the market, including Android phones, are equipped with Gregorian calendar system but do not have Ethiopic calendar [9]. But most users of mobile phones in Ethiopia are not comfortable with the Gregorian calendar as they use Ethiopian calendar in their day to day activities. These users are usually in a trouble when they want to have a reference to Ethiopic date as they had the Gregorian calendar at hand. Most of the users of these mobiles need to convert the Gregorian calendar to Ethiopic calendar in order to use it locally. Though it is possible to convert, only few users know the conversion method and the conversion also takes time. The other option is to carry paper based calendar in their pocket. Rather than buying a paper calendar each year or converting Gregorian to Ethiopic calendar, it is preferable if we have a perpetual Ethiopic and Gregorian calendar in our mobile which provides a calendar service of many years [3]. Since the mobiles do not have Ethiopic calendar, there is no means to know the date of the holidays. Again the mobiles that are available do not support to organize tasks and take diary in Ethiopic text. Therefore, to solve these problems an application should be developed which could generate any year, month and date of the Ethiopian calendar with the corresponding Gregorian calendar and permit to enter text in Ethiopic script for writing notes.

1.3 Objectives of the Project

1.3.1 General objective

The general objective of this project is to design and develop a system that automatically generates Ethiopic calendar with its associated Gregorian calendar for any given year and month in an Android-based mobile phone environment and implement an organizer and a digital diary book that permits Ethiopic text entry.

1.3.2 Specific objective

Specific objectives of the project are:

1. Design algorithms for perpetual Ethiopian calendar with the corresponding Gregorian calendar (Algorithm for conversion of Ethiopian to Gregorian calendar)
2. Implement perpetual Ethiopic calendar with the corresponding Gregorian calendar for Android mobile phone.
3. Design and implement an organizer and a digital diary book based on the Ethiopian calendar system that supports Ethiopic script based text entry.

4. Design an algorithm for automatic computing of religious holidays of Ethiopian calendar.
5. Test that the designed algorithms and the developed system work correctly.
6. Deploy the Ethiopic calendar with the organizer and diary system on Android mobile phone.

1.4 Methodology

For the realization of this project different methodologies are used which are presented as follows.

1.4.1 Data collection

In order to collect data about how the Ethiopic calendar system works different documents and developed calendars are used.

1.4.2 Literature review

For better understanding, besides document analysis, reviewing literatures and assessing related works have been done about the system.

1.4.3 Design of the system

Different algorithms have been designed for the implementation of the system.

1.4.4 Identification of development environment

For software development different tools have been used. The following tools were used for developing the application.

- Android SDK -provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.
- Eclipse - is an IDE on which the SDK is integrated for developing the application [21].

1.4.5 Implementation of the system

To implement the designed system, android emulator 2.1 and the physical android device have been used.

1.4.6 Testing of the system

For testing of the calendar system, already developed calendar systems for computers have been used. For checking of the holidays different digital and manual calendar systems were referenced. The organizer and diary system is tested on the physical android mobile device with some sample data.

1.5 Organization of the document

The remaining parts of this document are organized as follows. Chapter two presents the literature review and related works. It assesses literatures about the calendar system and identifies the related works with this project. Chapter three deals with the analysis of the system. Chapter four focuses on the design of the system. In chapter five the implementation of the system is presented. Conclusions and recommendations are given in chapter six.

CHAPTER TWO: LITERATURE REVIEW AND RELATED WORKS

2.1 Literature Review

A calendar is a system of organizing days, months, and years for measuring, recording and keeping of time which can be used for social, religious, commercial, or administrative purposes [9]. It allows its users to organize/schedule tasks and keep diaries in some specific period of time. This is done by giving names to periods of time, typically days, weeks, months, and years. The name given to each day is known as a date. Periods in a calendar such as years and months are usually synchronized with the cycle of the sun or the moon. Many civilizations and societies have devised a calendar, usually derived from other calendars on which they model their systems, suited to their particular needs. A calendar is also a physical device, often paper. This is the most common usage in the world. There are also digital calendars that integrated with the operating system of the computer or mobile devices or, which can be integrated into application systems to remind the user of upcoming events and appointments.

2.1.1 Types of Calendar System

There are diverse methods used in creating calendars. Some calendars replicate astronomical cycles according to fixed rules, others are based on abstract, perpetually repeating cycles of no astronomical significance. But the most popular calendars are those that use solar and lunar system to create calendars [4].

Solar Calendars

A solar calendar is a calendar whose dates indicate the position of the earth on its revolution around the sun. Solar calendars assign a date to each solar day. A day may consist of the period between sunrise and sunset, with a following period of night, or it may be a period between successive events such as two sunsets. The length of the interval between two such successive events may be allowed to vary slightly during the year, or it may be averaged into a mean solar day. Solar calendars consist of 365 or 366 (during leap year) days within a year.

Lunar Calendars

Not all calendars use the solar year as a unit. A lunar calendar is one in which days are numbered within each lunar phase cycle. Islamic calendar is one of the most popular calendars that use the lunar system [15]. It consists of 12 months of 29 or 30 days each, for a total of 354 days. The Islamic calendar has no corrective system to align it with the solar calendar. Thus the Islamic holidays do not always fall in the same season, and they occur earlier every year on the solar calendar [34].

The ancient Egyptian calendar was lunar. The solar Coptic calendar, oldest in history, originated three millennia before the birth of Christ [30]. But the exact date of its origin is unknown [2]. Historically, ancient Egyptians initially used a civil calendar based on a solar year that consisted of 365 days or 366 during leap year. The new year of the ancient Egyptians started on **መስከረም** 1 /September 11 which is also the new year of Ethiopians.

The Ethiopian calendar is based on the Coptic calendar, although it differs with regard to the saint's days and names of the months and weekdays [23]. The Coptic, or Egyptian, calendar is seven or eight years behind the Gregorian calendar. This discrepancy resulted from differences between the Orthodox Church and the Roman Catholic Church as to the date of the creation of the world. Like the Ethiopian calendar, the Egyptian calendar has 13 months. The first 12 months have 30 days. The last month, **ጳጉሜን**, is an intercalary/leap month, which has 6 days on leap year and 5 on the others [16]. The year starts on 11th September in the Gregorian calendar or on the 12th in Gregorian Leap Years. The Coptic Leap Year follows the same rule as the Gregorian so that the extra month always has 6 days in a Gregorian Leap Year [13].

The year 46 B.C. marked the introduction of the Julian calendar with 12 months of 30 and 31 days except in February, which had 29 and 30 in a leap year. By the Julian reckoning, the solar year comprised 365 days. The addition of a leap day every four years was intended to maintain correspondence between the calendar and the seasons; however, a slight inaccuracy in the measurement of the solar year caused the calendar dates of the seasons to regress almost one day per century. So, the Julian calendar was 10 days out of synchronization with the seasons. To correct the inaccuracy of the calendar, after a number of changes, in 1582 Pope Gregory XIII who ordered that 4th of October was to be followed by 15th of October by dropping 10 days from October and reestablished the consistency of solar and calendar years [5, 27]. This reformation led to what is known as the Gregorian calendar, which is still used today. This Gregorian reform gave an extremely accurate calendar system and reestablished 1st January as the beginning of the year [24, 26].

The Gregorian calendar also differs from the Julian in that no century year is a leap year unless it is exactly divisible by 400. Every year exactly divisible by four is a leap year, except for years that are exactly divisible by 100; the centurial years that are exactly divisible by 400 are still leap years [27].

The Gregorian solar calendar is an arithmetical calendar. It counts days as the basic unit of time, grouping them into years of 365 or 366 days. The years are divided into 12 months of irregular length. Leap years add 29 days to February, which normally has 28 days. But seven months January, March, May, July, August, October and December have 31 days and four months April, June, September, November have 30 days [27]

According to the beliefs of the Ethiopian Orthodox church, God created the world 5500 years before the birth of Christ. These are referred to as **ዓመተ ዓለም** or "the years of the world". The Ethiopic Enochian calendar has 364 days per year, Enoch 28:11 [28]. From the books of Ehoch, a curious 364 day length of calendar year lends new insight by reserving the last day of the solar year [28].

Since the Ethiopian calendar starts counting after the completion of **ዓመተ ፍዳ** (5500 years) it is seven years behind the Gregorian from September 11 or 12(during leap year) to January 1, and eight years behind from January 1 to end of the year. 5500 + **ዓመተ ምህረት** (year of mercy) years divided by 4 is an Ethiopic leap year if the remainder is 3. Leap years by the Ethiopian calendar are those that end in a Gregorian calendar year preceding a Gregorian calendar leap year. In Ethiopian leap year **ጳጉሜን** will have 6 days.

The Ethiopian calendar is based on the Ethiopian Orthodox Tewahido church calendar. The church used the calendar for a long period of time and still uses it. In order to calculate the fasting holidays the church uses its own calendar system. The calculation of the fasting dates is based on the book "**አቡሻኽር**" which was written by **ዲሜጥሮስ** [2]. How fasting dates are calculated will be discussed in the implementation part.

The Islamic (Hijrah*)* calendar is used in many predominantly Islamic countries. It is a purely lunar calendar that contains 12 months having 29 or 30 days that are based on the motion of the moon [34]. The calendar is used in most Muslim countries to determine the date of the holidays. Since the Ethiopian calendar is based on the Ethiopian Orthodox church calendar, the Muslim holidays date in the Islamic calendar is not the same with the Ethiopian calendar date [25].

2.2 Related works

There are different types of calendar systems in the world. Of these, the Gregorian is the most popular one. But Ethiopia has its own ancient calendar. In the past, Ethiopic calendars are produced in paper format for each year. But these days, Ethiopic calendars are being developed digitally. The Gregorian calendar is integrated with the computer operating system but the Ethiopic calendar is not. Most of the Ethiopic calendar system is developed as an application and installed on the computer. For instance, Power Ge'ez has power Ge'ez calendar which could be used in personal computers [7]. But this calendar is not supported on mobile devices. There is also Ethiopica calendar, which was developed as an application that can be installed on personal computers. It is a perpetual calendar that provides the Ethiopian and Gregorian dates and display in Ethiopic numerals [6]. Since the calendar software is developed for personal computers, it is not supported on mobile devices. The other Ethiopian calendar software, calendar system, is an application which was developed for computer system that displays both the Gregorian calendar and the Ethiopian calendar at one glance [11].

Most Ethiopian calendar softwares have been developed for computer systems. Almost all mobile phones are equipped with the Gregorian calendar system. Android mobile phones have a built-in Gregorian calendar application. This calendar directly synchronizes with the user's Google calendar [10]. Since Android mobile is the latest mobile, which runs on top of Android platform, there is no Ethiopic calendar. Therefore, the goal of this project is to develop a perpetual Ethiopic calendar with its corresponding European calendar which allows its users to schedule their tasks and keeps diary using Ethiopic script.

CHAPTER THREE: SYSTEM ANALYSIS

3.1 Introduction

An important part of software development is to explore the requirements for the system to be developed. A requirement is a feature that the system must have or a constraint that it must satisfy to be accepted by the client [12]. To successfully build the system, requirement elicitation which results in the specification of the system that the client understands, and analysis, which results into an analysis model that the developers can unambiguously interpret what should be done in the RAD [29]. Most of the time, clients and users are experts in their domain and have a general idea of what the system should do but developers have experience in building systems but often have little knowledge of the everyday environment of the users. In order to bridge this gap, use cases are used. A use case is an abstraction that describes a class of scenarios. Requirement elicitation, which focuses on describing the purpose of the system, for the system is done through observation and document analysis [12].

3.2 Functional requirements

Functional requirements describe the interactions between the system and its environment independent of its implementation. The environment includes the user and any other external system with which the system interacts. The users' tasks and the behavior of the system lead to the following functional requirements.

- Calculate the dates of Ethiopian calendar for each month for any year.
- Convert Ethiopian year, month and date to the Gregorian year, month and date.
- Display the Ethiopian and Gregorian calendar together with the specified year.
- Identify the current date in Ethiopic and Gregorian.
- Display the holidays, if any, of the month for the specified year.
- Allow users to organize their tasks and keep diary based on Ethiopian calendar using Ethiopic script.

3.3 Non-functional requirements

Non-functional requirements describe user-visible aspects of the system that are not directly related with the functional behavior of the system.

User interface

The user interface should be easy to use. It must allow users to interact with the system easily.

Documentation

The document about the system must be clearly written to support the users in using the system.

Hardware consideration

The system runs with any specification on top of Android mobile operating system.

Performance characteristics

The system should respond to the user as fast as possible.

Error handling

The program should handle any exceptions to protect the system from failure due to some errors. This will be closely monitored during and after user testing and debugging to ensure that any fault is quickly fixed.

Quality issues

The system should be robust and fault tolerant and easily recoverable after a failure. There should be long mean times between failures.

3.4 System Models

The goal of system model is to understand the business problem that is going to be addressed. Through requirements elicitation, it is possible to identify a problem area and define a system that addresses the problem which brings the system specification. Then the system specification is structured and formalized during analysis to produce an analysis model. They are both models of the system in the sense that they attempt to accurately represent the external aspects of the system.

The analysis model is then extended to describe how the actors and the system interact to manipulate the application domain. It is composed of three individual models: the functional model, represented by use cases, the analysis object model, represented by class and object diagrams, and the dynamic model, represented by activity diagram and sequence diagrams [29].

3.4.1 Use Cases Model

Use cases are used during requirements elicitation and analysis to represent the functionality of the system by focusing on the behavior of the system from an external point of view. A use case is initiated by an actor. After its initiation, a use case may interact with other actors as well. It represents a complete flow of events through the system in the sense that it describes a series of related interactions that result from the initiation of the use case [12]. The use case model is represented in UML with use case diagram. Figure 3.1 shows the use case diagram for the calendar, organizer and diary system.

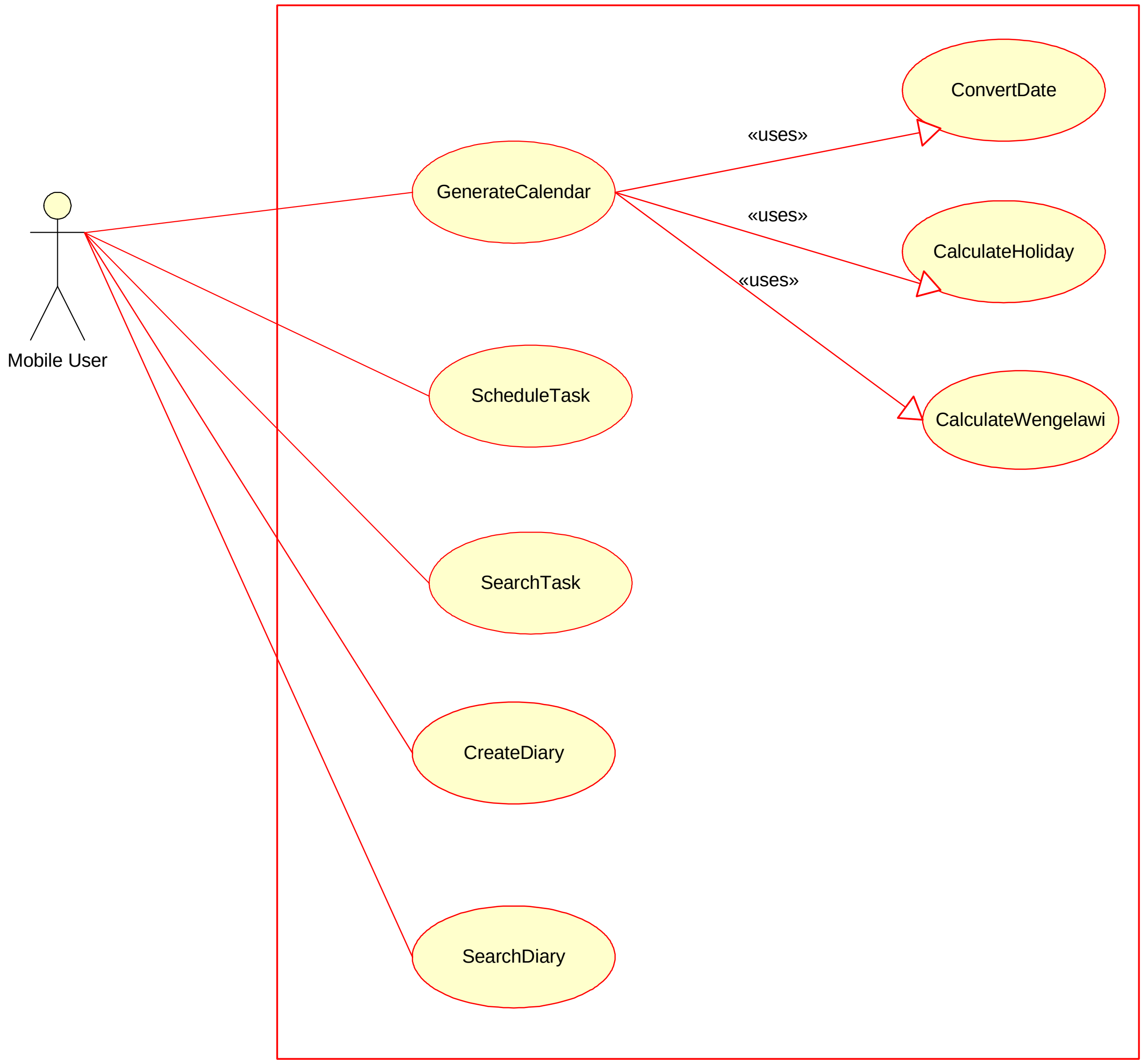


Figure 3.1 Use case diagram for calendar, organizer and diary system

Use case description

The following tables show the description of each use cases modeled in the use case model.

Table 3.1 describes the use case, GenerateCalendar, which generate and display the calendar for the user when the user selects month and enter the year. It also displays the holidays and **ወንጌላዊ** with the calendar.

Table 3.1 Use case description for GenerateCalendar

Use case name	GenerateCalendar
Actor	User
Description	This use case generates the calendar and display for the user. It also calculates the holidays associated with the selected month and display along with the calendar.
Entry condition	The user should decide the month and year for which he/she wants to generate calendar.
Flow of events	1. The user clicks on the month combo box. 2. The system lists the months in the combo box. 3. The user selects one month from the list. 4. The user enter the year on the text box 5. The system takes the selected month and the entered Ethiopian year. 6. The system generates and displays the calendar for the user. 7. The system displays the holiday, if there is any.
Exit condition	The calendar with holiday of the selected month of the year is displayed.

Table 3.2 describes the use case, CalculateHoliyday, which calculates the holiday for the selected month and entered year. This use case triggered when the user selects month or enters the year and clicks on the calendar button. In other words this use case is used by the use cases GenerateCalendar.

Table 3.2 Use case description for CalculateHoliday

Use case name	CalculateHoliday
Actor	User
Description	This use case calculates the date of the holidays for the month and year selected by the user.
Entry condition	Month and year are provided
Flow of events	1. The system takes the month and year. 2. The system calculates the holiday date, if there is any, for the provided year and month. 3. The system displays the calendar with the holidays.
Exit condition	Holiday is calculated and displayed

Table 3.3 describes the use case, CalculateWengelawi, which calculates the **ወንጌላዊ** (Gospel) of the entered year. This use case triggered when the user enters year and clicks on calendar. In other words this use case is used by the use case GenerateCalendar.

Table 3.3 Use case description for CalculateWengelawi

Use case name	CalculateWengelawi
Actor	User
Description	This use case calculates the **ወንጌላዊ** (Gospel) of the year.
Entry condition	Year is selected
Flow of events	1. The system takes the year. 2. The system calculates **ወንጌላዊ**. 3. The system displays the **ወንጌላዊ** of the year.
Exit condition	**ወንጌላዊ** is calculated and displayed

Table 3.4 describes the use case, ConvertDate, which converts the Ethiopian date to Gregorian date for the year and month provided. This use case triggered when the user selects month or clicks on calendar. In other words this use case is used by the use cases GenerateCalendar.

Table 3.4 Use case description for ConvertDate

Use case name	ConvertDate
Actor	User
Description	This use case converts the Ethiopian date to Gregorian date for the month and year provided.
Entry condition	Month and year are provided
Flow of events	1. The system takes the month and year. 2. The system converts the Ethiopian date to Gregorian date for the year and month. 3. The system displays the Ethiopian date with the Gregorian date.
Exit condition	Ethiopian date is converted to Gregorian date and the calendar is displayed

Table 3.5 describes the use case, ScheduleTask, which allows the user to schedule tasks based on the Ethiopian calendar. Since the schedule is based on the Ethiopian calendar, first the calendar should be generated and displayed for the user. Then the user should select any date ahead of current date in order to get the schedule form. The system displays the schedule form with the selected date and the user should enter the task and the time of the schedule for the selected date. Once the task is scheduled, the systems alarm service alarms the user when the system time equals with the time of the scheduled task.

Table 3.5 Use case description for ScheduleTask

Use case name	ScheduleTask
Actor	User
Description	This use case allows the user to schedule or organize tasks for a specific date, month and year of Ethiopian calendar.
Entry condition	Date is selected
Flow of events	1. The system takes the selected date, month and year to the schedule form. 2. The user types the task and set the time of schedule on the form. 3. The user click on the save button to save the task.
Exit condition	Task is scheduled

Table 3.6 describes the use case, DisplayTask, which allows the user to see scheduled tasks and allows them to update or delete.

Table 3.6 Use case description for DisplayTask

Use case name	DisplayTask
Actor	User
Description	This use case allows the user to display tasks already scheduled
Entry condition	There should be a scheduled task
Flow of events	1. The user click on the date from the calendar. 2. The user clicks on the display button. 3. The system displays the scheduled task one by one. 4. If the task is not scheduled the system informs the user.
Exit condition	Task is displayed

Table 3.7 describes the use case, CreateDiary, which allows the user to create diary based on the Ethiopian calendar. To create the diary the user click on the diary (**ማስታወሻ**) button from the calendar which leads to the diary form. On the diary form the user write the diary using Ethiopic script.

Table 3.7 Use case description for CreateDiary

Use case name	CreateDiary
Actor	User
Description	This use case allows the user to create diary (**ማስታወሻ**).
Entry condition	The calendar should be displayed
Flow of events	1. The user click on the diary (**ማስታወሻ**) button. 2. The system reads the current date, month and year from the calendar and passes to the diary form. 3. The user writes the diary. 4. The user clicks on save button to save the diary. 5. If the diary is not saved the system informs the user.
Exit condition	Diary is saved

Table 3.8 describes the use case, DisplayDiary, which allows the user to display the diary already created. The user can display the diary one by one or the user can search for a diary created at specific date. In order to display diary of a specific time, the user must enter the data, month and year. Based on these criteria the system searches the diary and display. If there is no diary found at this specific time, the system informs the user about unavailability of the diary.

Table 3.8 Use case description for DisplayDiary

Use case name	DisplayDiary
Actor	User
Description	This use case allows the user to display diary (**ማስታወሻ**).
Entry condition	There should be a diary created
Flow of events	1. The user click on the diary (**ማስታወሻ**) button. 2. The user clicks on the display button. 3. The system displays the diary one by one or the user enters the date, month and year to display a diary at specific time. 4. If the diary is not found, the system informs the user.
Exit condition	Diary is displayed

3.4.2 Object Model

Object model describes the structure of a system in terms of objects, attributes, associations, and operations. It consists of entity, boundary and control objects which are identified from the use cases [29]. It is represented in UML class diagrams. In order to represent the objects using class diagram the different objects are identified as follows.

Entity Object

Entity object represents the persistent information tracked by the system. They will be identified by examining each use cases. For the calendar, organizer and diary system, the following entity objects are identified from the use cases.

1. SelectMonth and SelectYear use cases

- **User –** the person who selects the month for which he/she wants to see the Ethiopian calendar.
- **Calendar –** the Ethiopian and Gregorian date for the selected month and year.

- **Holiday –** the holidays (religious and non-religious) calculated for the selected month and year.
- **ወንጌሳዊ –** the calculated **ወንጌሳዊ** of the selected or default year.

2. ScheduleTask use case

- **User –** the person who schedules his/her task
- **Task –** the activity to be scheduled by the user.
- **Message –** the message displayed for the user about the success or failure of the schedule.

3. DisplayTask use case

- **User –** the person who looks for tasks scheduled.
- **Task –** the scheduled task which is found and displayed for the user.
- **Message –** the message displayed for the user about the failure or success of displaying tasks

4. CreateDiary use case

- **User –** the person who wants to create a diary.
- **Diary –** the note written by the user and saved on the database.
- **Message –** the message displayed for the user about the success or failure of recording of the diary.

5. DisplayDiary use case

- **User –** the person who wants to search for the diary from the database.
- **System –** the system that takes the input from the user and search fro the diary. The input from the user is the date of creation of the diary.
- **Diary –** the note retrieved from the database.
- **Message –** the message displayed for the user about the success or failure of the display. If the diary is found the message will be success otherwise failure.

Boundary Object

Boundary object represents the system interface with the actors. It collects information from the actors and translates it into an interface neutral form that can be used by the entity objects and also by control objects. For the calendar, organizer and diary system the following boundary objects are identified.

1. Boundary objects for the SelectMonth and SelectYear use cases

- **MonthDisplayCombobox –** used by the user to display the Ethiopian months and select one of the month from the list.
- **YearDisplayCombobox –** used by the user to display the Ethiopian years and select one of the year from the list.
- **CalendarForm –** used to display the calendar with the holidays and **ወንጌላዊ** for the user.

2. ScheduleTask use case

- **TaskForm –** form used for input of the task to be scheduled. The form is displayed for the user when the user clicks on the date from the calendar.

3. Record Diary use case

- **Diary Form –** form used to keep diary. The form is displayed when the user select diary on the calendar form.

Control Object

Control object represents the task that is performed by the user and supported by the system. It is responsible for coordinating boundary and entity objects. The following control objects are identified from the use cases for the calendar and organizer system.

1. MonthSelect use case

- **ListMonthControl –** lists the months when the user clicks on the month combo box and pass the selected month for the system to perform the conversion between Ethiopian and Gregorian date. It also passes the month for the function that calculates the holidays for the selected month. Then it displays the calendar and the holidays.

2. YearSelect use case

- **ListYearControl –** Lists the years when the user clicks on the year combo box and passes the selected year for the system to perform the conversion between Ethiopian and Gregorian date. It also passes the selected year for the function that calculates the holiday and wengelawi of the year.

3. ScheduleTask use case

- **ScheduleControl –** accepts the input data and saves to the database.

4. CreateDiary use case

- **DiaryControl –** accepts the input data and saves on the database.

3.4.3 Class diagram

Class diagrams describe the structure of the system in terms of classes and objects. It is an abstraction that specifies the attributes of a set of objects. It shows the classes of the system, their interrelationships, and the operations and attributes of the class [12]. It is depicted as boxes with three sections: the top one indicates the name of the class, the middle one lists the attributes of the class, and the third one lists the methods. Figure 3.2 shows the class diagrams of the system.

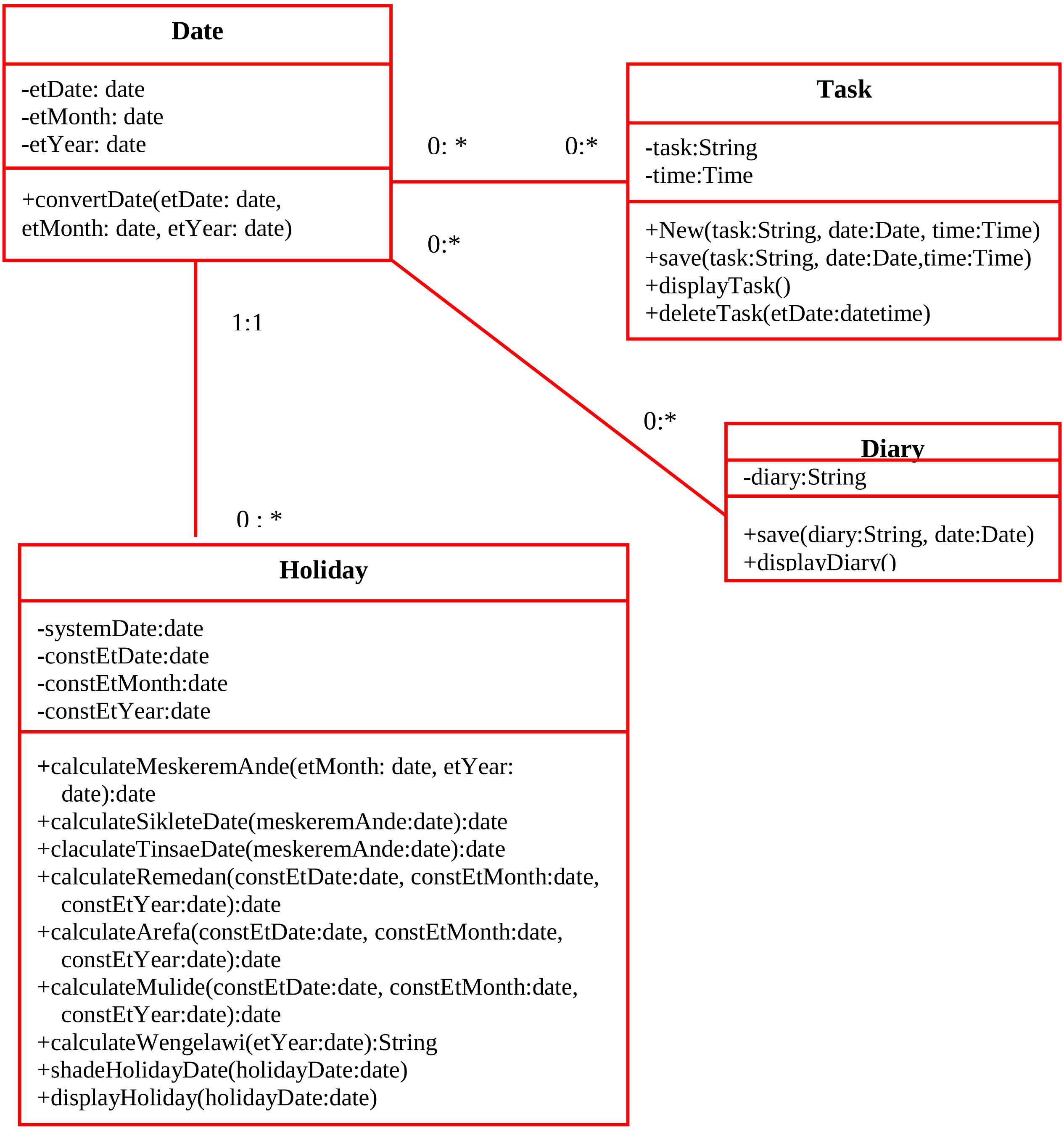


Figure 3.2 Class diagram for calendar, organizer and diary system

3.4.4 Sequence Diagram

A sequence diagram ties use cases with objects. It shows how the behavior of a use case is distributed among its participating objects and the sequence of interactions among objects needed to realize the use case [12]. The columns of a sequence diagram represent the objects that participate in the use case. The leftmost column is the actor who initiates the use case. Horizontal arrows across columns represent messages, or stimuli, which are sent from one object to the other. Time proceeds vertically from top to bottom. Figure 3.3 shows the sequence diagram of GenerateCalendar use case identified in the requirement elicitation activities. The use case triggered when the user selects the month and entered the year and then clicks on the calendar button.

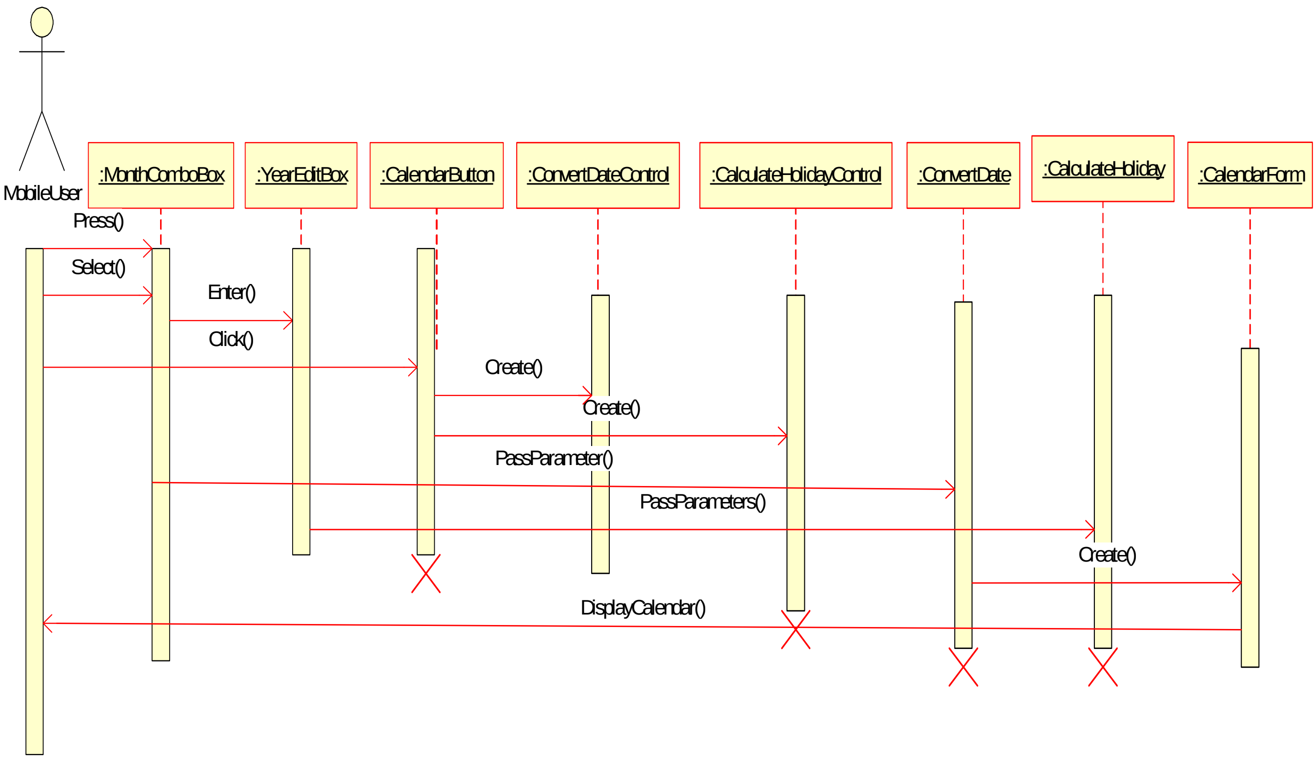


Figure 3.3 Sequence diagram for GenerateCalendar use case

Figure 3.4 shows the sequences of actions occurred when the users activate the schedule task use case.

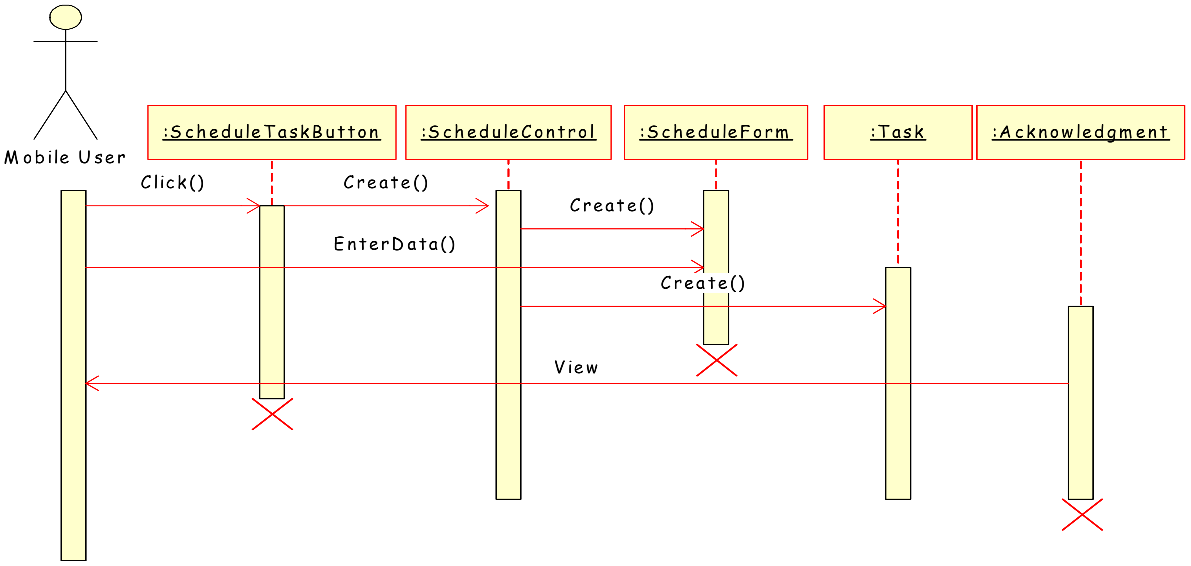


Figure 3.4 Sequence diagram for ScheduleTask use case

Figure 3.5 shows the sequence of actions in order to display tasks scheduled for some specific period.

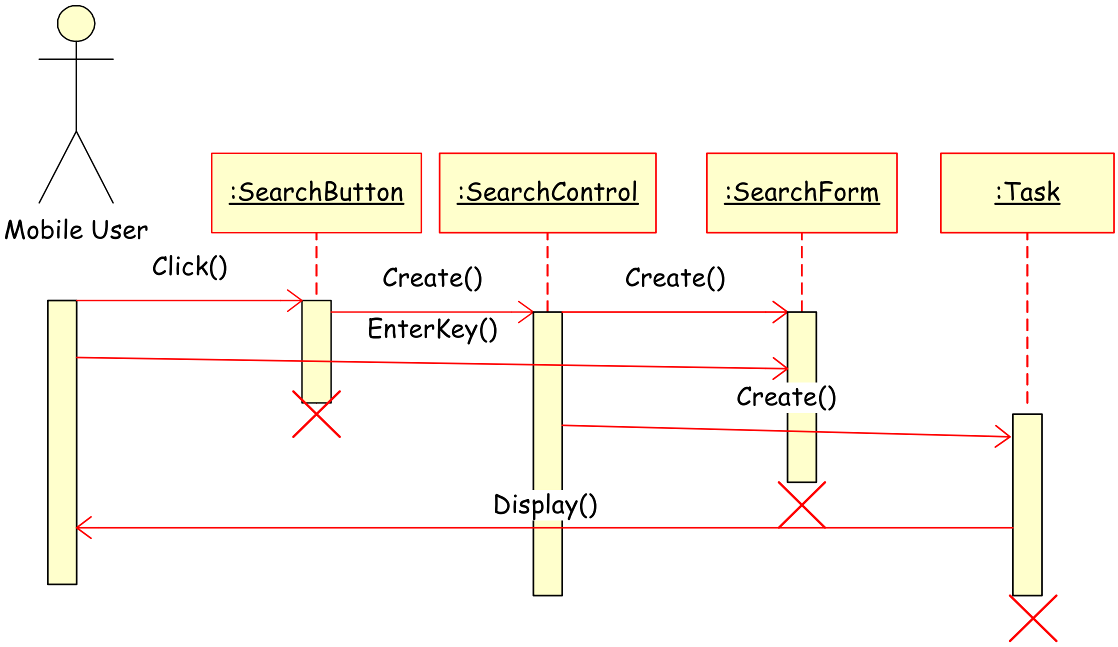


Figure 3.5 Sequence diagram for DisplayTask use case

Figure 3.6 shows the sequence diagram for diary creation use case that depicts the interaction of objects in order to allow the user to keep diary.

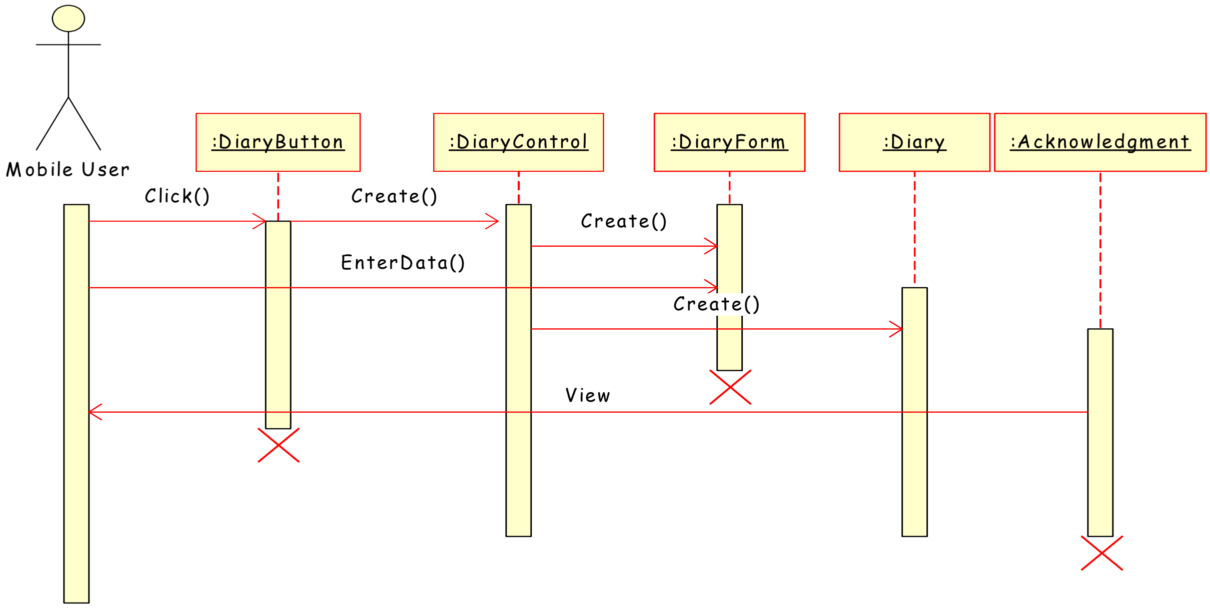


Figure 3.6 Sequence diagram for CreateDiary use case

Figure 3.7 shows the sequence diagram of display diary. The diagram shows the interactions between objects to display the diary kept by the user one by one.

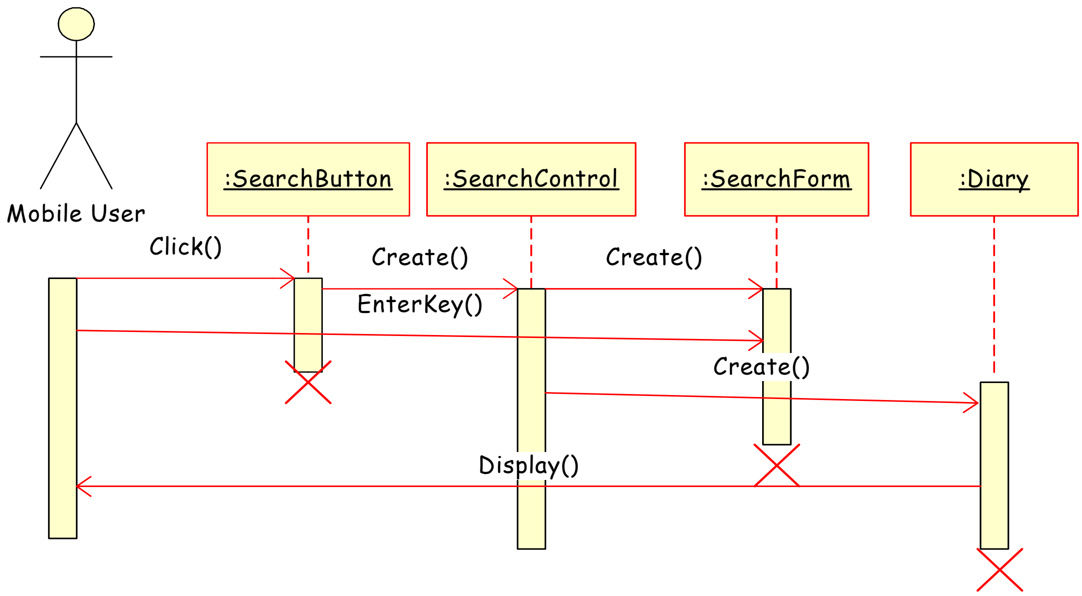


Figure 3.7 Sequence diagram for DisplayDiary use case

3.4.5 Activity diagram

Activity diagrams are typically used for business process modeling, for the logic captured by a single use case or scenario, or for modeling the detailed logic of a business rule [12]. Figure 3.8 shows the activity diagram for the system and figure 3.9 shows the activity diagram of the calendar and organizer system.

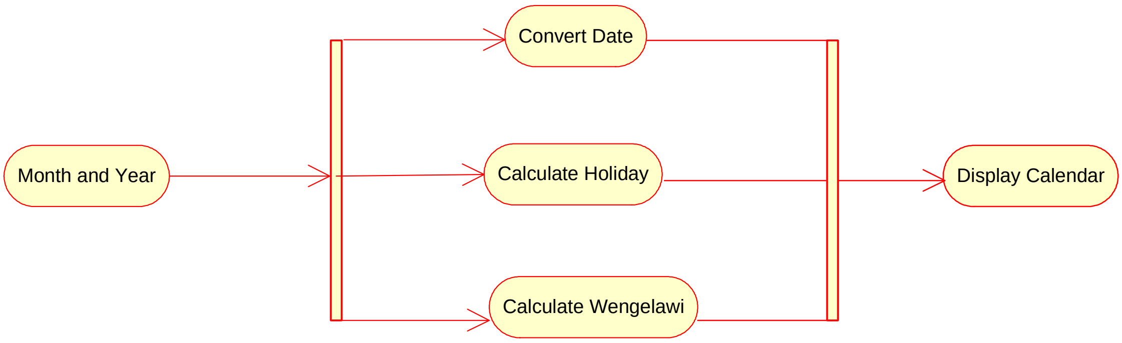


Figure 3.8 Activity diagram for calendar system

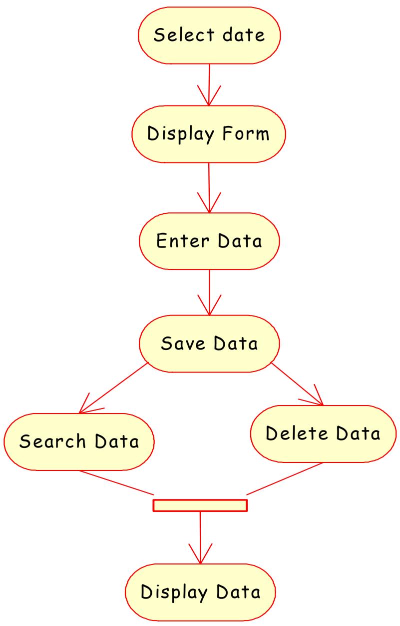


Figure 3.9 Activity diagram for organizer

CHAPTER FOUR: SYSTEM DESIGN

4.1 Overview

System design is the transformation of the analysis model into a system design model. It is concerned with defining the design goals of the project and decompose the system into smaller subsystems, select strategies for building the system, such as the hardware/software platform on which the system will run, the persistent data management strategy, the global control flow, the access control policy, and the handling of boundary conditions [29]. The result of system design is a model that includes a clear description of each of these strategies, subsystem decomposition, and a UML deployment diagram representing the hardware/software mapping of the system. There are two important products of systems design [12]:

Design goals - describing the qualities of the system that should be optimized in the system which are derived from the nonfunctional requirements and,

Software architecture - describing the subsystem decomposition in terms of subsystem responsibilities, dependencies among subsystems, subsystem mapping to hardware, and major policy decisions such as control flow, access control, and data storage.

In this chapter the design goals, which are derived from non-functional requirements, the software architecture, system decomposition and database management system are discussed.

4.2 Design goals

Design goal focuses on identifying the qualities that the system should comprise. Most of the design goals are inferred from the non-functional requirement of the requirement analysis document (RAD) and discussed as follows.

4.2.1 Performance criteria

Performance may include the speed and memory requirements of the system. Processing speed and available memory are the main constraints of handheld devices.

Response time: the system should calculate the calendar and holiday and display within fraction of seconds. Although this factor is not only imposed by the system, there are other factors like speed and the type of device also contribute to the response time. But the system by itself must be fast enough to respond to the user's requests.

Memory Requirement: the system should take less memory space since mobile devices have small storage devices.

Throughput: The system should display the calendar, holiday and wengelawi all at a time when the user selects the month or year.

4.2.2 Dependability criteria

This factor determines how much effort should be expected in minimizing system crashes and their consequences. A question to be raised in this section includes: how often can the system crush, availability of the system to the user and security risks associated with the system.

Robustness

Robustness is the ability of the system to survive for some errors so that system functioning will not be disrupted. If there is any error the system displays the appropriate error message.

Reliability

The system should provide a service for the user consistently and correctly. It should also provide the correct calendar for any year and month the user selects.

4.2.3 Maintenance Criteria

Items discussed in this section include: how easy it is to add new functionality to the system, and how easy it is to understand the code.

Extensibility

The system must be hospitable to accommodate new functionality in the future. For instance if there might be any new holiday that should be incorporated to the calendar, the system must allow to add it to the system. Adding new functionality or class is easier, since object oriented approach is used to structure modules and classes, thereby tracing the code and adding new functionality should be fairly easy.

Readability

The code should be commented and documented for future maintenance and upgrade.

4.2.4 End User Criteria

The end user criterion includes qualities that are desirable from users' point of view and have not yet been covered under the performance and dependability criteria.

Usability: The system shall be developed so as to be easy for user understanding. By providing the calendar using Ethiopic script and display both the Ethiopian and Gregorian calendar any user can use it easily.

4.3 Architecture of the System

The architecture of the system shows the general structure of the system using different components. For calendar, organizer and diary system, three important components are identified to create the architecture. The first component is the interface part through which the users have access to the system. The second component is the controller that accepts the users request and passes to the system that performs the intended operation. The last component is the implementer that performs the operation of the user. Figure 4.1 shows the architecture of the system.

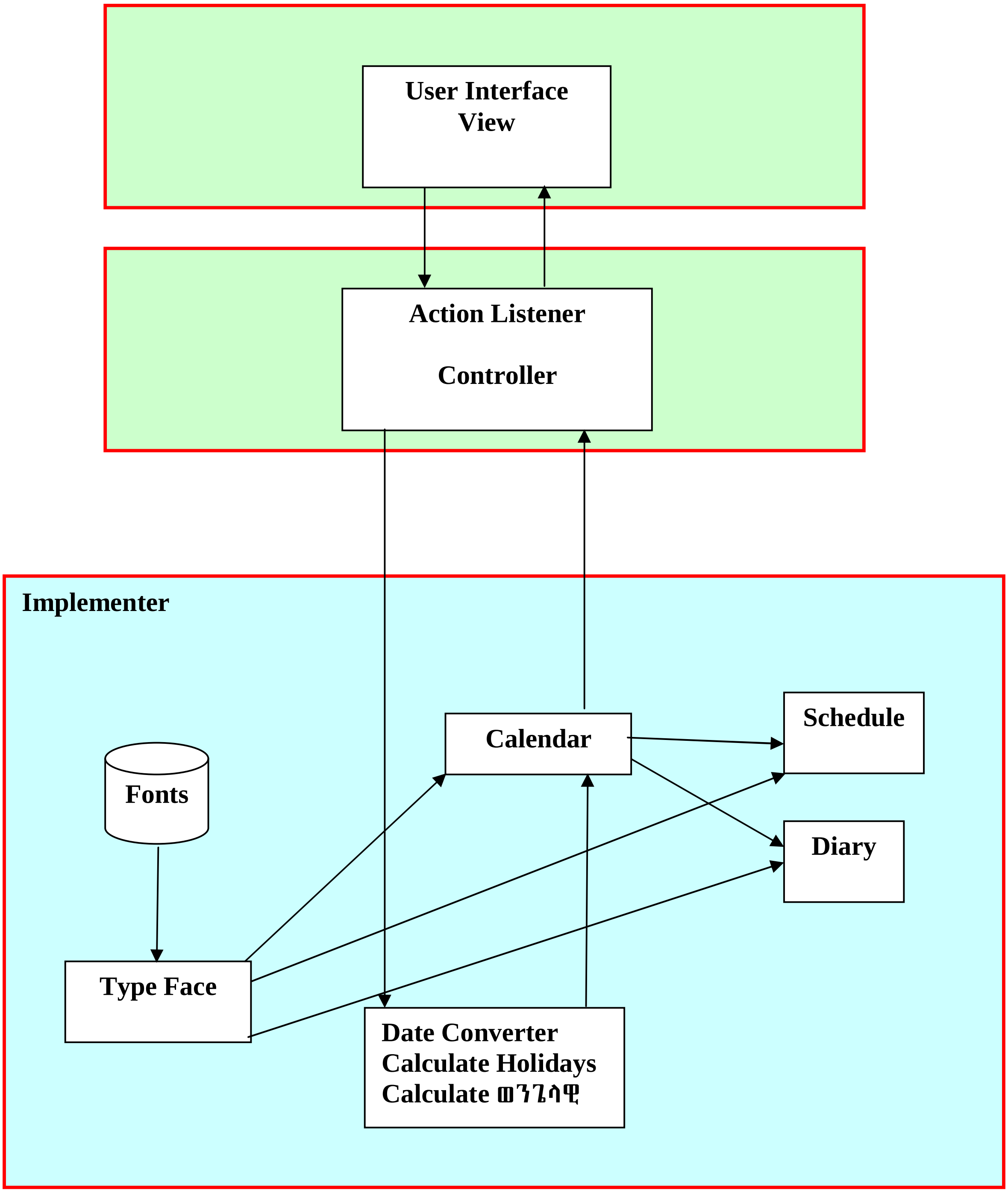


Figure 4.1 Architecture of the calendar, organizer and diary system

As depicted in figure 4.1 the architecture of calendar and organizer system has three parts. The first part is the Interface through which the user communicates with the system. The user selects the month or enters year for which the system convert the Ethiopian date to the Gregorian date, calculate the holiday date and **ወንጌሳዊ**. The communication of the user with the system is handled by a middle interface called controller.

The controller takes any input from the interface and passes it to the implementer that performs the calculation. Specifically the controller takes the month and the year that the user selects. Then it passes to the system to perform the necessary operation.

The implementer part takes the input from the controller and performs the required operation. The first task the system performs is just to convert the Ethiopian date to the Gregorian date. After the Ethiopian date is found it should be in Ethiopic script. To display the calendar in Ethiopic script the system uses the font database which is provided through asset manger. Once the calendar is generated, it is possible to keep diary and schedule tasks for specific date based on the Ethiopic calendar. The schedule uses the alarm services to provide alarm for the user.

4.4 Subsystem decomposition

In order to reduce the complexity of the solution domain, the system is decomposed into simpler parts, called subsystems, which are made of a number of solution domain classes. In the case of complex subsystems, it is possible to apply this principle and decompose a subsystem into simpler subsystems. System decomposition is the process of breaking down the whole system into manageable subsystems. Figure 4.2 shows the subsystem of the calendar, organizer and diary system.

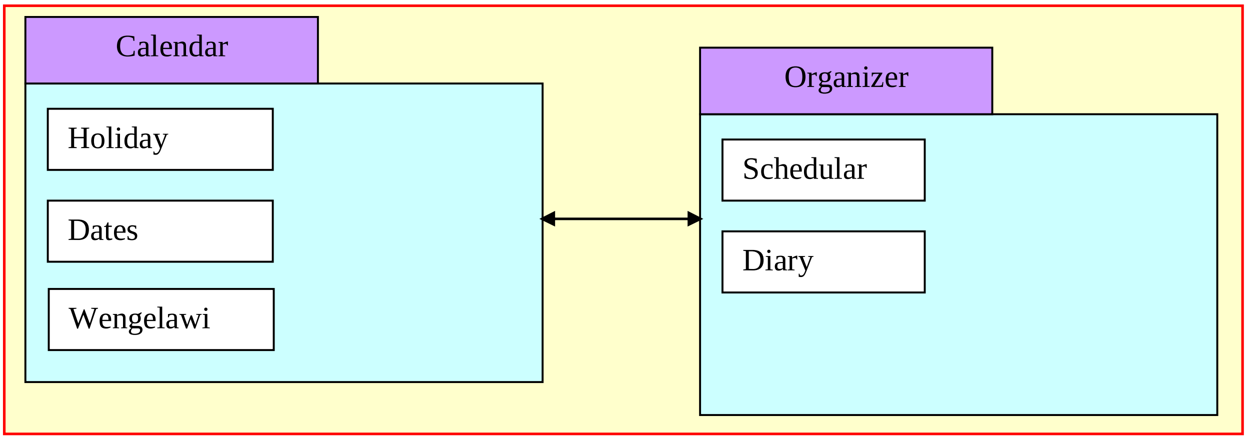


Figure 4.2 Sub system decomposition of calendar, organizer and diary system

4.5 Persistent Data Management

Persistent data management describes the persistent data stored by the system and the data management option required for it. Organizer and diary system maintains data about tasks organized and diaries kept by the user. Relational database management system is used to manage the data. Android uses SQLite for its database needs. SQLite is a very fast and lightweight database. It is file based which makes it highly portable and an ideal candidate for use as an embedded database. The data should be stored in a way that facilitates reliable retrieval making it easily to find specific information. There are two tables in the system: Task table and diary table. Table 4.1 depicts the task table that handles data of the task to be scheduled by the user for specific Ethiopian date. Table 4.2 depicts the table that handles data of the diary the user keeps.

Table 4.1 Task table

Task : Table

	Field Name	Data Type
🔑	_id	Number
	task	Text
	date	Date/Time
	hour	Number
	minute	Number
	time	Text

Table 4.2 Diary table

Diary : Table

	Field Name	Data Type
🔑	_id	Number
	note	Text
	date	Date/Time

4.5 Ethiopic Calendar application

Ethiopia has its own ancient calendar system which is similar to the Coptic calendar system but it is different from the Gregorian calendar system. Although the Ethiopian, Coptic, Julian and Gregorian calendars have the same number of days in a year, (365 days and 366 days in a leap year), the counting systems giving the number of days in each month, and number of months in a year, of the Julian and Gregorian calendars differ from the Ethiopian and Egyptian Coptic calendars. The Ethiopian and Coptic calendars consist of 13 months where the first 12 months have 30 days each, and the last (thirteenth) month has 5 days (6 days in Ethiopic leap year). The Gregorian calendar consists of 12 months with varying number of dates between months.

- January, March, May, July, August, October and December have 31 days;
- April, June, September and November have 30 days and
- February have 28 days (29 days in a leap year).

The Ethiopian New year starts on **መስከረም** 1 in Ethiopian calendar. This is on September 11 in the Gregorian Calendar (G.C.) or on the 12th in Gregorian Leap Years. The Ethiopian calendar is 7 or 8 years behind the Gregorian calendar. In Gregorian calendar, the new year starts on January 1. In the Gregorian calendar every year that is exactly divisible by 4 is a leap year. But (5500 + **ዓመተ ምህረት**) divided by 4 is an Ethiopic Leap year if the remainder is 3 or leap years by the Ethiopian Calendar are those that end in a Gregorian calendar year preceding a Gregorian calendar leap year. The Ethiopic leap year have 6 days in **ጳጉሜን**. Table 4.3 shows the difference between the Ethiopian and Gregorian date. Based on this difference the algorithm is designed which is implemented for the system.

Table 4.3 Date difference between Ethiopian and Gregorian calendar

Ethiopian Month	Gregorian Month	Gregorian Equivalent Dates	Start Date	Start Date Leap Year
መስከረም (month 1)	September (month 9)	September 11 - October 10	11 Sept	12 Sept
ጥቅምት (month 2)	October (month 10)	October 11 - November 9	11 Oct	12 Oct
ህዳር (month 3)	November (month 11)	November 10 - December 9	10 Nov	11 Nov
ታህሳስ (month 4)	December (month 12)	December 10 - January 8	10 Dec	11 Dec
ጥር(month 5)	January (month 1)	January 9 - February 7	9 Jan	10 Jan
የካቲት (month 6)	February (month 2)	February 8 - March 9	8 Feb	9 Feb
መጋቢት (month 7)	March (month 3)	March 10 - April 8	10 Mar	
ሚያዝያ (month 8)	April (month 4)	April 9 - May 8	9 Apr	
ግንቦት (month 9)	May (month 5)	May 9 - June 7	9 May	
ሰኔ (month 10)	June (month 6)	June 8 - July 7	8 Jun	
ሐምሌ(month 11)	July (month 7)	July 8 - August 6	8 Jul	
ነሐሴ (month 12)	August (month 8)	August 7 - September 5	7Aug	
ጳጉሜን (month 13)	September 6 - September 10 (ends September 11, during Ethiopic leap years)			

The difference between the Ethiopian and Gregorian calendar is shown below.

ከመስከረም 1 እስከ ጥቅምት 21 የ 10 ቀን ልዩነት

ሲፕ (Leap year) **ሲሆን የ** 11 **ቀን ልዩነት**

ከጥቅምት 22 እስከ ታህሳስ 22 የ 9 ቀን ልዩነት

ሲፕ (Leap year) **ሲሆን የ** 10 **ቀን ልዩነት**

ከታህሳስ 23 እስከ ጥር 23 የ 8 ቀን ልዩነት

ሲፕ (Leap year) **ሲሆን የ** 9 **ቀን ልዩነት**

ከጥር 24 እስከ የካቲት 21 የ 7 ቀን ልዩነት

ሲፕ (Leap year) **ሲሆን የ** 8 **ቀን ልዩነት**

ከየካቲት 22 እስከ መጋቢት 22 የ 9 ቀን ልዩነት

ከመጋቢት 23 እስከ ግንቦት 23 የ 8 ቀን ልዩነት

ከግንቦት 24 እስከ ሐምሌ 24 የ 7 ቀን ልዩነት

ከሐምሌ 25 እስከ ነሐሴ 25 የ 6 ቀን ልዩነት

ከነሐሴ 26 እስከ ጳጉሜን 5 የ 5 ቀን ልዩነት

The Gregorian calendar exceeds by 7 or 8 years from Ethiopian calendar. From **መስከረም** 1 to **ታህሳስ** 22 the difference is by 7 years and from **ታህሳስ** 23 to the end of the year the difference is by 8 years. Based on this difference the perpetual Ethiopic calendar system is designed that display both the Ethiopian and Gregorian calendar together. Based on the difference given in Table 4.3, the following algorithm is generated.

Algorithm for date conversion from Ethiopian to Gregorian date

```
If Gregorian month is between January and August
     Ethiopian year= Gregorian year -8
Else
     Ethiopian year = Gregorian year -7

If (Gregorian month is January)
     If (remainder of Ethiopian year divided by 4 is 0)
          If (Gregorian date is less than 10)
               Ethiopian month is ታህሳስ (December)
               Ethiopian date = (Gregorian date + 31) – 10
          Else
               Ethiopian month is ጥር (January)
               Ethiopian date = Gregorian date – 9
     Else
          If (Gregorian date is less than 9)
               Ethiopian month is ታህሳስ (December)
               Ethiopian date = (Gregorian date + 31) – 9
          Else
               Ethiopian month is ጥር (January)
               Ethiopian date = Gregorian date – 8
Else if (Gregorian month is February)
     If (remainder of Ethiopian year divided by 4 is 0)
          If (Gregorian date is less than 10)
               Ethiopian month is ጥር (January)
               Ethiopian date = (Gregorian date + 31) – 9
          Else
               Ethiopian month is የካቲት (February)
               Ethiopian date = Gregorian date – 8
     Else
          If (Gregorian date is less than 8)
               Ethiopian month is ጥር (January)
               Ethiopian date = (Gregorian date + 31) – 8
          Else
               Ethiopian month is የካቲት (February)
               Ethiopian date = Gregorian date – 7
Else if (Gregorian month is March)
     If (Gregorian date is less than 10)
```

```
          Ethiopian month is የካቲት (February)
          Ethiopian date = (Gregorian date + 29) – 8
     Else
          Ethiopian month is መጋቢት (March)
          Ethiopian date = Gregorian date – 9
Else if (Gregorian month is April)
     If (Gregorian date is less than 9)
          Ethiopian month is መጋቢት (March)
          Ethiopian date = (Gregorian date + 30) – 8
     Else
          Ethiopian month is ሚያዝያ (April)
          Ethiopian date = Gregorian date – 8
Else if (Gregorian month is May)
     If (Gregorian date is less than 9)
          Ethiopian month is ሚያዝያ (April)
          Ethiopian date = (Gregorian date + 30) – 8
     Else
          Ethiopian month is ግንቦት (May)
          Ethiopian date = Gregorian date – 8

Else if (Gregorian month is June)
     If (Gregorian date is less than 8)
          Ethiopian month is ግንቦት (May)
          Ethiopian date = (Gregorian date + 31) – 8
     Else
          Ethiopian month is ሰኔ (June)
          Ethiopian date = Gregorian date – 7
Else if (Gregorian month is July)
     If (Gregorian date is less than 8)
          Ethiopian month is ሰኔ (June)
          Ethiopian date = (Gregorian date + 30) – 7
     Else
          Ethiopian month is ሐምሌ (July)
          Ethiopian date = Gregorian date – 7
Else if (Gregorian month is August)
     If (Gregorian date is less than 7)
          Ethiopian month is ሐምሌ (July)
          Ethiopian date = (Gregorian date + 31) – 7
     Else
          Ethiopian month is ነሐሴ (August)
          Ethiopian date = Gregorian date – 6
Else if (Gregorian month is September)
     If (remainder of Ethiopian year divided by 4 is 0)
          If (remainder of Gregorian year divided by 4 is 0)
               If (Gregorian date is less than 6)
                    Ethiopian month is ነሐሴ (August)
```

```
                    Ethiopian date = (Gregorian date + 31) – 6
               Else
                    If (Gregorian date is less than 12)
                         Ethiopian month is ጳጉሜን
                         Ethiopian date = Gregorian date – 5
                    Else
                         Ethiopian month is መስከረም (September)
                         Ethiopian date = Gregorian date – 10
          Else
               If (Gregorian date is less than 6)
                    Ethiopian month is ነሐሴ (August)
                    Ethiopian date = (Gregorian date + 31) – 7
               Else
                    If (Gregorian date is less than 12)
                         Ethiopian month is ጳጉሜን
                         Ethiopian date = Gregorian date – 6
                    Else
                         Ethiopian month is መስከረም (September)
                         Ethiopian date = Gregorian date – 10
     Else
          If (remainder of Gregorian year divided by 4 is 0)
               If (Gregorian date is less than 5)
                    Ethiopian month is ነሐሴ (August)
                    Ethiopian date = (Gregorian date + 31) – 5
               Else
                    If (Gregorian date is less than 11)
                         Ethiopian month is ጳጉሜን
                         Ethiopian date = Gregorian date – 4
                    Else
                         Ethiopian month is መስከረም
                         Ethiopian date = Gregorian date – 10
          Else
               If (Gregorian date is less than 5)
                    Ethiopian month is ነሐሴ (August)
                    Ethiopian date = (Gregorian date + 31) – 6
               Else
                    If (Gregorian date is less than 11
                         Ethiopian month is ጳጉሜን
                         Ethiopian date = Gregorian date – 5
                    Else
                         Ethiopian month is መስከረም (September)
                         Ethiopian date = Gregorian date – 10
Else if (Gregorian month is October)
     If (remainder of Ethiopian year divided by 4 is 0)
          If (Gregorian date is less than 12)
               Ethiopian month is መስከረም (September)
```

```
            Ethiopian date = (Gregorian date + 30) – 11
        Else
            Ethiopian month is ጥቅምት (October)
            Ethiopian date = Gregorian date – 11
    Else
        If (Gregorian date is less than 11)
            Ethiopian month is መስከረም (September)
            Ethiopian date = (Gregorian date + 30) – 10
        Else
            Ethiopian month is ጥቅምት
            Ethiopian date = Gregorian date – 10
Else if (Gregorian month is November)
    If (remainder of Ethiopian year divided by 4 is 0)
        If (Gregorian date is less than 11)
            Ethiopian month is ጥቅምት (October)
            Ethiopian date = (Gregorian date + 31) – 11
        Else
            Ethiopian month is ህዳር (November)
            Ethiopian date = Gregorian date – 10
    Else
        If (Gregorian date is less than 10)
            Ethiopian month is ጥቅምት (October)
            Ethiopian date = (Gregorian date + 31) – 10
        Else
            Ethiopian month is ህዳር (November)
            Ethiopian date = Gregorian date – 9
Else if (Gregorian month is December)
    If (remainder of Ethiopian year divided by 4 is 0)
        If (Gregorian date is less than 11)
            Ethiopian month is ህዳር (November)
            Ethiopian date = (Gregorian date + 30) – 10
        Else
            Ethiopian month is ታህሳስ (December)
            Ethiopian date = Gregorian date – 10
    Else
        If (Gregorian date is less than 10)
            Ethiopian month is ህዳር (November)
            Ethiopian date = (Gregorian date + 30) – 9
        Else
            Ethiopian month is ታህሳስ (December)
            Ethiopian date = Gregorian date – 9
```

4.5.1 Ethiopian Holidays

There are different types of holidays in Ethiopia. They categorized into three: Christian holidays, Muslim holidays and National holidays. Table 4.4 presents Christian holidays, Table 4.5 presents Muslim holidays and Table 4.6 presents national holidays.

Table 4.4 Christian Holidays

Christian Holidays	Ethiopian Calendar	Gregorian Calendar
ዘመን መስወጫ-አዲስ አመት (New Year)	**መስከረም 1**	September 11 (September 12 in the leap year)
መስቀል(Finding of the True Cross)	**መስከረም 17**	September 27/28
ልደት (Christmas)	**ታህሳስ 29** **ታህሳስ 28 ሲፕ አመት**	January 7/6
ጥምቀት (Epiphany)	**ጥር 11**	January 19
ስቅለት (Good Friday)	**መጋቢት/ሚያዝያ (ተለዋዋጭ ቀናት)**	March/April (varying date)
ትንሳኤ-ፋሲጋ (Easter)	**መጋቢት/ሚያዝያ (ተለዋዋጭ ቀናት)**	March/April (varying date)

Table 4.5 Muslim Holidays

Islamic Holidays	Ethiopian Calendar
ኢድ-አል-ፈጥር(ረመዳን)- Id Al Fater (Ramadan)	Varying date
ኢድ-አል-አድሃ(ዐረፋ) – Id Al Adaha (Arefa)	Varying date
የነቢዩ መሐመድ ልደት(መዉሊድ)-Birthday of the prophet Mohammed (Maulide)	Varying date

Table 4.6 National, non religious, Holidays

National Holidays (non religious)	Ethiopian Calendar	Gregorian calendar
የዓድዋ ድል መታሰቢያ - Victory of Adwa	**የካቲት 23**	March 2
የኢትዮጲያ አርበኞች መተሰሰቢያ - Patriots Victory Day	**ሚያዝያ 27**	April 6
ዓለም አቀፍ የሳብአደሮች ቀን - Labour Day	**ሚያዝያ 23**	May 1
ኢሕአዴግ ደርግን ያስወገደበት - Downfall of the Dergue	**ግንቦት 20**	May 28

4.5.2 Christian Holidays

Christians in general have two types of holidays, fixed and movable. The fixed ones are observed always at the same time in the year or month, while the movable ones move annually forward or backward. The book **ባህረ ሐሳብ (አቡሻኽር)** is, primarily, a mathematical system for fixing the movable fasts of the Ethiopian Orthodox Church. There are seven officially celebrated Christian holidays:

- **እንቁጣጣሽ-አዲስ አመት** (New Year)
- **መስቀል**(Finding of the True Cross)
- **ልደት** (Christmas)
- **ጥምቀት** (Epiphany)
- **ስቅለት** (Good Friday)
- **ትንሳኤ-ፋሲጋ** (Easter)

Of the seven Christian holidays, four of them **እንቁጣጣሽ (አዲስ አመት)**, **መስቀል**, **ልደት**, **ጥምቀት** are constant; they appear on the same month and day from year to year. But two of them **ስቅለት** and **ትንሳኤ (ፋሲጋ)** are not constant, they fall on different days with in the month of **መጋቢት** or **ሚያዝያ** from year to year. From the constant holidays, Christmas (**ልደት**) can be on **ታህሳስ** 29 or **ታህሳስ** 28 if the year is Ethiopian **ሲኘ** (Leap year). For those movable holidays, Ethiopian Orthodox church uses **ባህረ ሐሳብ** to calculate the days. Based on the calculation of this book, the day of new year **መስከረም** 1, Good Friday, and Easter is calculated as follows.

Calculating the day of መስከረም 1

The Ethiopian calendar is much more similar to the Egyptian Coptic calendar having a year of 13 months, 365 days and 366 days in a leap year (every fourth year) and it is much influenced by the Ethiopian Orthodox Tewahedo Church, which follows its ancient calendar rules and beliefs. The Ethiopian calendar is always seven years and eight months behind the Gregorian (Western) and Eastern Orthodox Church calendars during September to December and eight years and four months behind during January to August. Therefore, the Ethiopians will celebrate the New Year on **መስከረም** 1. Enkutatash (**እንቁጣጣሽ**) is the word for the Ethiopian New Year in Amharic, the working language of Ethiopia, the term preferred by the Ethiopian Orthodox Tewahedo Church. In order to calculate the day of **መስከረም** 1, the Orthodox Tewahdo church uses its own calendar system based on the book "**ባህረ ሐሳብ**". The book provides the formula of calculating the day of **መስከረም** 1 and any other fasting days of the church. Getting the day of **መስከረም** 1 is very important for calculating the movable fasting days. Therefore, based on the book [31] the following formulas are used to calculate the day of **መስከረም** 1 for any given year.

Steps to find the date of መስከረም 1:

1. The following values must be given

- **ዓመተ ምህረት** – is the year for which we want to calculate the day of **መስከረም** 1.
- **ዓመተ ፍዳ** - the years before the birth of Christ, which is equivalent to 5500.

2. Calculate **ዓመተ ዓለም** – is the year of the world.

ዓመተ ዓለም = **ዓመተ ምህረት** + **ዓመተ ፍዳ**

3. Calculate **ዓመተ ወንጌላዊ**

ዓመተ ወንጌላዊ= **ዓመተ ዓለም** / 4

The division is by 4 because there are four **ወንጌላዊ**. The remainder of the division is used to know the **ወንጌላዊ** of the given year. So, if the remainder is:

1 **ዓመተ ምህረት** becomes Zemene Matthew (**ዘመነ ማቴዎስ**)

2 **ዓመተ ምህረት** becomes Zemene Mark (**ዘመነ ማርቆስ**)

3 **ዓመተ ምህረት** becomes Zemene Luke (**ዘመነ ሉቃስ**)

0 **ዓመተ ምህረት** becomes Zemene John (**ዘመነ ዮሐንስ**)

4. Calculate **ጠቅላላ ዓመት**

ጠቅላላ ዓመት = **ዓመተ ዓለም** + **ዓመተ ወንጌላዊ** + 2

5. Calculate when **መስከረም** 1 falls.

Divide **ጠቅላላ ዓመት** by 7 (because there are 7 days) and use the remainder to know the day of **መስከረም** 1. If the remainder is:

1 **መስከረም** 1 is on Sunday **(እሁድ**)

2 **መስከረም** 1 is on Monday (**ሰኞ**)

3 **መስከረም** 1 is on Tuesday (**ማክሰኞ)**

4 **መስከረም** 1 is on Wednesday **(ረቡዕ)**

5 **መስከረም** 1 is on Thursday (**ሐሙስ**)

6 **መስከረም** 1 is on Friday **(ዐርብ**)

0 **መስከረም** 1 is on Saturday (**ቅዳሜ**)

Based on the above calculation, let us calculate when **መስከረም** 1 falls for the current year (2002 **ዓ.ም**).

1. Givens

ዓመተ ምህረት = 2002

ዓመተ ፍዳ = 5500

2. **ዓመተ ዓለም** = **ዓመተ ምህረት** + **ዓመተ ፍዳ**

= 2002 + 5500

= 7502

3. **ዓመተ ወንጌላዊ**= **ዓመተ ዓለም** / 4

= 7502 / 4

= 1875

ወንጌላዊ = 7502 % 4

= 2

Where % refers to the modules operator which returns the remainder of the division of the first number by the second number.

The remainder is 2. Therefore, **ዓመተ ምህረት 2002** (year 2002) is Zemene Mark **(ዘመነ ማርቆስ**).

4. **ጠቅላላ ዓመት** = **ዓመተ ዓለም** + **ዓመተ ወንጌላዊ**+ 2

= 7502 + 1875 + 2

= 9379

5. **መስከረም** 1 day = **ጠቅላላ ዓመት** % 7

= 9379 % 7

= 6

The remainder is 6. Therefore, **መስከረም** 1 was on Friday (**አርብ**).

Algorithm for calculating the day of መስከረም 1

Get Ethiopian year

Get **ዓመተ ፍዳ (5500)**

Calculate **ዓመተ ዓለም**

ዓመተ ዓለም = ዓመተ ምህረት ዓመተ ፍዳ

Calculate **ዓመተ ወንጌላዊ**

ዓመተ ወንጌላዊ = ዓመተ ዓለም / 4

If (the remander of **ዓመተ ዓለም** divided by **4** equals 1)

ወንጌላዊ =ማቴዎስ

Else If (the remander of **ዓመተ ዓለም** divided by **4** equals 2)

ወንጌላዊ =ማርቆስ

If (the remander of **ዓመተ ዓለም** divided by **4** equals 3)

ወንጌላዊ =ሉቃስ

If (the remander of **ዓመተ ዓለም** divided by **4** equals 0)

ወንጌላዊ =ዮሐንስ

Calculate **ጠቅላላ ዓመት**

ጠቅላላ ዓመት = ዓመተ ዓለም + **ዓመተ ምህረት** + 2

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 1)

ቀን = እሁድ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 2)

ቀን = ሰኞ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 3)

ቀን = ማክሰኞ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 4)

ቀን = ረቡዕ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 5)

ቀን = ሐሙስ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 6)

ቀን = አርብ

If (the remander of **ጠቅላላ ዓመት** divided by **7** equals 0)

ቀን = ቅዳሜ

Calculating the day of Good Friday and Easter

Based on the calculation of **ባህረ ሐሳብ** Good Friday should always fall on Friday and Easter should fall on Sunday only within the month **መጋቢት** or **ሚያዝያ**. To find the days of these two holidays there are a number of requirements. The following are the requirements that should be calculated.

- The day of **መስከረም 1**
- **መደብ**
- **ወንበር**
- **አበቅቴ**
- **መጥቅ**
- **መባጃ ሐመር**
- **ነነዌ**
- **ጾም መግቢያ**

In addition to the above values “**የለት ተዉሳክ**” and “**የጾም ተዉሳክ**” constants are used during the calculation of the holidays. Table 4.7 shows the **ተዉሳክ**” of the 7 days and Table 4.8 shows the **ተዉሳክ**” of the **ጾም**.

Table 4.7 **የዕለት ተዉሳክ**

Day	**የዕለት ተዉሳክ**
Sunday (**እሁድ**)	7
Monday (**ሰኞ**)	6
Tuesday (**ማክሰኞ**)	5
Wednesday (**ረቡዕ**)	4
Thursday (**ሐሙስ**)	3
Friday (**አርብ**)	2
Saturday (**ቅዳሜ**)	8

Table 4.8 **የጾም ተዉሳክ**

በዓለ ጾሙ	**የሚዉልበት ዕለት**	**እስከ ነነዌ ያሉት ቀናት**	**ተዉሳክ (በ 30 ሲገደፍ)**
ዐቢይ ጾም	**ሰኞ**	14	14
ደብረ ዘይት	**እሁድ**	41	11
ሆሳዕና	**እሁድ**	62	2
ስቅለት	**አርብ**	67	7
ትንሳኤ	**እሁድ**	69	9

Steps:

1. Calculate **መስከረም** 1 – shown in the above steps

2. Calculate **መደብ**

    መደብ = **ዓመተ አለም** % 19

3. Calculate **ወንበር**

    ወንበር= **መደብ**-1

Exception: If **መደብ** is zero (0), **ወንበር** will be 19

4. Calculate **አበቅቴ**

    አበቅቴ = **መንበር** * 11

If **አበቅቴ** is greater than 30, **አበቅቴ** will be the remainder of (**ወንበር** * 11) divided by 30.

5. Calculate **መጥቅ**

መጥቅ = 30 - **አበቅቴ**

Always the sum of **አበቅቴ** and **መጥቅ** is 30. So, if **አበቅቴ** is zero, **መጥቅ** will be 30.

Two important rules:

- If **መጥቅ** is between 1 and 14, **በዓለ መጥቅ** will be on **ጥቅምት** and **ነነዌ** will be in **የካቲት**.
- If **መጥቅ** is between 15 and 30, **በዓለ መጥቅ** will be in **መስከረም** and **ነነዌ** will be in **ጥር**.

The date of the **መጥቅ** will be used as the date of **በዓለ መጥቅ** on the month it falls. For instance, if **መጥቅ** is 5, **በዓለ መጥቅ** will be on **ጥቅምት** 5. In order to get the day of **ጥምቀት** 5 we have to count starting from the day of **መስከረም** 1. Once we get the date of **በዓለ መጥቅ**, we use **የለት ተዉሳክ** to calculate **መባጃ ሐመር**.

6. Calculate **መባጃ ሐመር** date

መባጃ ሐመር = **መጥቅ** + **የለት ተዉሳክ**

The value of **መባጃ ሐመር** is used as the date of **ነነዌ**.

Exception: If **መጥቅ** is greater than 22 **በዓለ መጥቅ** will be on **መስከረም** and we find the day of **የለት ተዉሳክ** which falls in **መስከረም** but **ነነዌ** will be on the day of **መባጃ ሐመር** within the month of **የካቲት**.

7. Calculate **ጾም መግቢያ** date

ጾም መግቢያ = **ነነዌ** + **አብይ ጾም ተዉሳክ**

- If **ጾም መግቢያ** is greater than 30, **ጾም መግቢያ** will be on the next month with the day of the remainder of **ጾም መግቢያ** divided by 30.

8. Calculate **ደብረ ዘይት** date

ደብረ ዘይት = **መባጃ ሐመር** + **የደብረ ዘይት ተዉሳክ** . If **ደብረ ዘይት** is less than **ጾም መግቢያ**, increment month by 1.

9. Calculate **ሆሳዕና** date

ሆሳዕና = **መባጃ ሐመር** + **የሆሳዕና ተዉሳክ** . If **ሆሳዕና** is less than **ደብረዘይት**, increment month by 1.

10. Calculate **ስቅለት** date

ስቅለት = **መባጃ ሐመር** + **የስቅለት ተዉሳክ**. If **ስቅለት** is less than **ሆሳዕና**, increment month by 1.

11. Calculate **ትንሳኤ** date

ትንሳኤ = **መባጃ ሐመር** + **የትንሳኤ ተዉሳክ** (9). If **ትንሳኤ** is less than **ስቅለት**, increment month by 1.

According to **ባህረ ሐሳብ ስቅለት** always must falls between **መጋቢት** 24 and **ሚያዝያ 28** and **ትንሳኤ** must falls between **መጋቢት 26** and **ሚያዝያ 30**.

Based on the above rules let us calculate the day of **ስቅለት** and **ትንሳኤ** for the current year (2002 **ዓ.ም**).

1. From the previous calculation **መስከረም** 1 was on Friday (**አርብ**).

2. **መደብ** = **ዓመተ ዓለም** %19

= 7502 % 19

= 16

3. **ወንበር** = **መደብ**-1

=16-1

=15

4. **አበቅቴ** = **ወንበር** * 11

= 15 * 11

= 165

Since **አበቅቴ** is greater than 30 we have to divide it by 30 and take the remainder.

አበቅቴ = 165 % 30

= 15

5. **መጥቅ** = 30 - **አበቅቴ**

= 30-15

= 15

Since **መጥቅ** is between 15 and 30, **በአለ መጥቅ** will be on **መስከረም** and **ነነዌ** will be on **ጥር**.

To know the day of **በዓለ መጥቅ**, we count from the day of **መስከርም** 1. **መስከርም** 1 is on Friday **በዓለ መጥቅ** is on Friday **መስከረም** 15. **የዕለት ተዉሳክ** of Friday is 2.

9. **መባጃ ሐመር** = **መጥቅ** + **የለት ተዉሳክ**

= 15 + 2

= 17

- If **መባጃ ሐመር** is greater than 30, **መባጃ ሐመር** will be the remainder of **መባጃ ሐመር** divided by 30.

Therefore, **ነነዌ** will be on **ጥር** 17.

10. **ጾም መግቢያ** = **መባጃ ሐመር** + **የዓብይ ጾም ተዉሳክ**

= 17+14

= 31

Since **ጾም መግቢያ** is greater than 30, we divide it by 30 and take the remainder and increment the month by 1.

ጾም መግቢያ = 31%30

= 1

Therefore, the day of **ጾም መግቢያ** is **የካቲት** 1.

11. **ደብረ ዘይት** = **መባጃ ሐመር** + 11

= 17+11

= 28

Therefore, **ደብረ ዘይት** will be on **የካቲት** 12

12. **ሆሳዕና** = **መባጃ ሐመር** + 2

= 17 + 2

= 19

Therefore, **ሆሳዕና** will be on **መጋቢት**19

13. **ስቅለት** = **መባጃ ሐመር** + 7

= 17+7

= 24

Therefore, **ስቅለት** will be on **መጋቢት** 24

12. **ትንሳኤ** = **መባጃ ሐመር** + 9

= 17 + 9

= 26

Therefore, **ትንሳኤ** will be on Megabit 26

Algorithm for calculating the day of Good Friday and Easter

Get the day of September (**መስከረም) 1**

Calculate **መደብ**

መደብ = remainder of **ዓመተ ዓለም** divided by 19

Calculte **ወንበር**

ወንበር = መደብ - 1

If (**መደብ** is equal to zero)

ወንበር = 19

Calculte **አበቅቴ**

አበቅቴ = ወንበር * **11**

If (**አበቅቴ** is greater than 30)

አበቅቴ = the remainder of **አበቅቴ** divided by 30

Calculate **መጥቅ**

መጥቅ = 30 - አበቅቴ

If (**መጥቅ** is between 1 and 14)

በዓለ መጥቅ is in october (**ጥቅምት)**

ነነዌ is in February (**የካቲት)**

If (**መጥቅ** is between 15 and 30)

በዓለ መጥቅ is in September (**መስከረም)**

ነነዌ is in January (**ጥር)**

Calculate **መባጃ ሐመር**

መባጃ ሐመር = መጥቅ + **የእለት ተዉሳክ**

If (**መጥቅ** is greater than 22)

በዓለ መጥቅ is in September (**መስከረም)**

Find the day when **በዓለ መጥቅ** falls and read **የእለት ተዉሳክ** of the day

Calculate **መባጃ ሐመር** and use it as the date of **ነነዌ**

Make **ነነዌ** to **የካቲት**

Calculate **ጾም መግቢያ**

ጾም መግቢያ = ነነዌ + **አብይ ጾም ተዉሳክ**

If (**ጾም መግቢያ** is greater than 30)

= the remainder of **ጾም መግቢያ** divided by 30

If (**ጾም መግቢያ** is less than **መባጃ ሐመር**)

Increment month by one

Calculate **ደብረ ዘይት**

ደብረ ዘይት = **መባጃ ሐመር** + **የደብረ ዘይት ተዉሳክ**

If (**ደብረ ዘይት** is greater than 30)

ደብረ ዘይት= the remainder of **ደብረ ዘይት** divided by 30

If (**ደብረ ዘይት** is less than **ጾም መግቢያ)**

Increment month by one

Calculate **ሆሳዕና**

ሆሳዕና = **መባጃ ሐመር** + **የሆሳዕና ተዉሳክ**

If (**ሆሳዕና** is greater than 30)

ሆሳዕና = the remainder of **ሆሳዕና** divided by 30

If (**ሆሳዕና** is less than **ደብረ ዘይት)**

Increment month by one

Calculate **ስቅለት**

ስቅለት= **መባጃ ሐመር** + **የስቅለት ተዉሳክ**

If (**ስቅለት** is greater than 30)

ስቅለት= the remainder of **ስቅለት** divided by 30

If (**ስቅለት** is less than **ሆሳዕና)**

Increment month by one

Calculate **ትንሳኤ**

ትንሳኤ = **መባጃ ሐመር** + **የ ትንሳኤ ተዉሳክ**

If (**ትንሳኤ** is greater than 30)

ትንሳኤ = the remainder of **ትንሳኤ** divided by 30

If (**ትንሳኤ** is less than **ስቅለት)**

Increment month by one

4.5.3 Muslim Holidays

There are three Muslim holidays that celebrated by Ethiopian Muslims. The date of the holidays depends on the calendar of the Islamic calendar but celebrated on the Ethiopian calendar which is ahead of the Islamic Calendar by 11 days [34, 36]. This is because of the Islamic Calendar has 12 month with 29 or 30 days. Due to this, the Muslim holidays in Ethiopian calendar appear in various days and months from year to year. In order to calculate the month and the day of the holiday of the next year, we subtract 11 from the current holiday date. To calculate the holiday of the next year the current year holiday date must be used as a reference. For instance if Id Al Fater (Ramadan) is on **መስከረም** 26, 2002, on the next year Id Al Fater will be on **መስከረም** 15 (26-11), 2003. So, the formula is just to keep the holiday of one year as a constant and subtract 11 from the year used as a constant to get the holiday of the next year. To find for the year 2004 subtract 22 which is (2*11). This way, we can continue to calculate the holidays of each year.

The algorithm to calculate the date of Muslim Holidays

Get the date, month and year of the Holiday of one year as constant value
To calculate the holiday of the next year
    Get date, month of the year
    Subtract 11 from the previous date
        Date = constant date – 11
    If (the result of subtraction is negative)
        Date =30 + Date
        Decrement month by 1
The above algorithm is true for all the three holidays celebrated by Ethiopian Muslims.

CHAPTER FIVE: IMPLEMENTATION

5.1 System Development Environment

This chapter deals with the implementation of the system that will run on the top of Android platform. Android platform is designed for mobile devices maintained by Google and supported by the Open Handset Alliance, a group of companies who have come together to accelerate innovation in mobile and offer consumers a richer, less expensive, and better mobile experience. Together they have developed Android, the first complete, open, and free mobile platform. To develop Android applications in Eclipse environment Google has provided an Eclipse-plug-in, Android Development Toolkit (ADT). Android programming is done using Java syntax, plus a class library that resembles a subset of the Java SE library (plus Android-specific extensions) and they run within a virtual machine (VM). It's important to note that the VM is not a JVM but it is the Dalvik Virtual Machine, an open source technology. Each Android application runs within an instance of the Dalvik VM, which in turn resides within a Linux-kernel managed process. As shown in Figure 5.1, the user application is at the top layer and the core Linux Kernel is at the bottom.

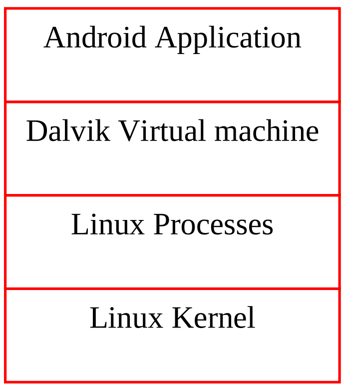


Figure 5.1 Dalvik Virtual Machine running the Android applications [36]

5.2 Android platform - Architecture

Android uses Linux for its device drivers, memory management, process management, and networking. Android native libraries are written in C/C++ internally, but it is possible to call them through Java interfaces. Android Dalvik Virtual Machine runs dex files, which are converted at compile time from standard class and jar files. Dex files are more compact and efficient than class files, an important consideration for the limited memory and battery powered devices that Android targets. Most applications of the users are running on the application layer of the Android platform.

Android architecture is basically divided into several components: Application Framework, Libraries, Android Runtime and Linux Kernel. As mentioned before, Android has virtual machine called Dalvik and every Android application will run on its own process, that is, using its own instance of Dalvik JVM. The Dalvik JVM runs files called Dalvik executables (.dex files), a kind of Java bytecode optimized for the Android platform [21]. Figure 5.2 shows the architecture of the Android platform.

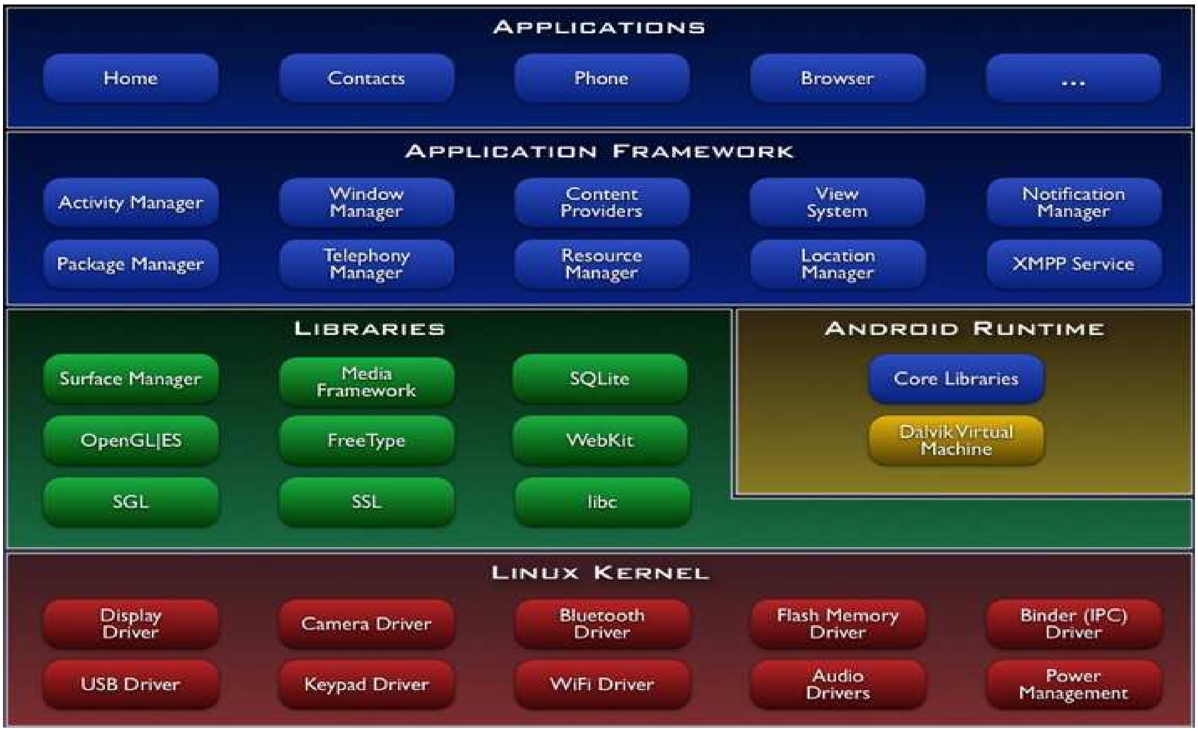


Figure 5.2 Architecture of Android platform [21]

Applications

Android ships a set of core applications such as an email client, SMS program, calendar, maps, browser, contacts, and others. All applications are written using the Java programming language. Any application developed for Android platform will run on the top of this layer [37].

Application Framework

Developers have full access to the same framework APIs used by the core applications. The application architecture is designed to simplify the reuse of components; any application can publish its capabilities and any other application may then make use of those components. This mechanism allows components to be replaced by the user [37].

Libraries

Android includes a set of C/C++ libraries used by various components of the Android system. These capabilities are exposed to developers through the Android application framework. Android includes a set of core libraries that provides most of the functionality available in the core libraries of the Java programming language. The libraries provide the reusable and sharable low-level code for basic functions such as codecs - software for coding and decoding digital sound and video - functions for the presentation of rich graphics on a small displays, secure shell support for encrypted TCP/IP traffic into the cloud, as well as component support for Web browsing (WebKit), SQL database functionality (SQLite), and standard C library functionality you would expect in a Linux system [37].

Linux Kernel

Android relies on Linux version 2.6 for core system services such as security, memory management, process management, network stack, and driver model. The kernel also acts as an abstraction layer between the hardware and the rest of the software stack. The Acorn RISC Machine (ARM) Linux core forms the solid base upon which all the other layers stand. Linux is a proven technology that is highly reliable, and the ARM processor family is known for high performance on very low power requirements. ARM is one of the most popular platforms for embedded devices. With this architecture the Android platform provides a number of advantages such as: Open, All applications are created equal, Fast & easy application development and breaking down application boundaries [37].

Open

Android was built from the ground-up to enable developers to create compelling mobile applications that take full advantage of all a handset has to offer. It was built to be truly open. Android is open source; it can be liberally extended to incorporate new cutting edge technologies as they emerge. The platform will continue to evolve as the developer community works together to build innovative mobile applications [37].

All applications are created equal

Android does not differentiate between the phone's core applications and third-party applications. They can all be built to have equal access to a phone's capabilities providing users with a broad spectrum of applications and services. With devices built on the Android Platform, users are able to fully tailor the phone to their interests [37].

Breaking down application boundaries

Android breaks down the barriers to building new and innovative applications. For example, a developer can combine information from the web with data on an individual's mobile phone such as the user's contacts, calendar, or geographic location to provide a more relevant user experience. With Android, a developer can build an application that enables users to view the location of their friends and be alerted when they are in the vicinity giving them a chance to connect [37].

Fast & easy application development

Android provides access to a wide range of useful libraries and tools that can be used to build rich applications. For example, Android enables developers to obtain the location of the device, and allows devices to communicate with one another enabling rich peer-to-peer social applications. In addition, Android includes a full set of tools that have been built from the ground up alongside the platform providing developers with high productivity and deep insight into their applications [37].

Android SDK comes with a bunch of tools that relieve the creation of an Android application. The following tools are the most important once which are available in the SDK.

- aapt (Android Asset Packaging Tool) – used to create *.apk-files which contain all the resources as well as the program itself. Those ZIP-format based files can be transferred to and installed on an Android phone or the emulator.
- adb (Android Debug Bridge) - used to set up connections to a real Android device or an Android emulator instance in order to transfer and install (apk)-files on it. With adb the developer also has the possibility to remote control the devices shell.
- dx (Dalvik Cross-Assembler) - used for merging and converting Java-Standard-ByteCode Classes (*.class) into one single binary file (*.dex) that can be executed by the Dalvik VM. These *.dex-files are subject to be put into an *.apk-file together with resource files.
- ddms (Dalvik Debug Monitor Service) – used to provide port-forwarding services, screen capture on the device, thread and heap information on the device, logcat, process and radio state information, incoming call and SMS spoofing, location data spoofing, and more.

Using the different tools, provided by SDK, the program written by the user is compiled to create the executable file which is called the Android Package (.apk) that can run o the emulator of real device. Figure 5.3 shows the Android application compilation process.

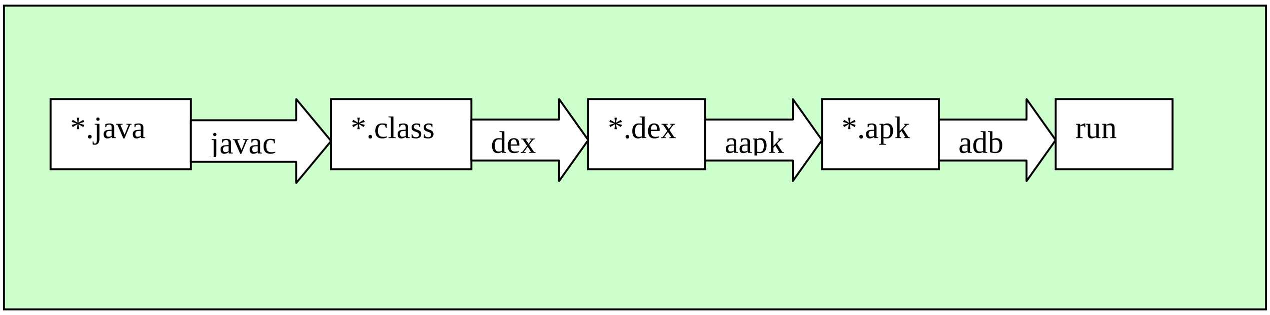


Figure 5.3 Compilation process of Android application

5.3 Development Environment and Interfaces

For the development of the application Eclipse integrated environment that supports Android Development Tools (ADT) plug-in has been used. The ADT plug-in gives enhancement for developing Android applications and handles all packaging and installation of the application for using the Android Emulator on Eclipse IDE. Figure 5.4 shows the Eclipse IDE on which the code of the application is written.

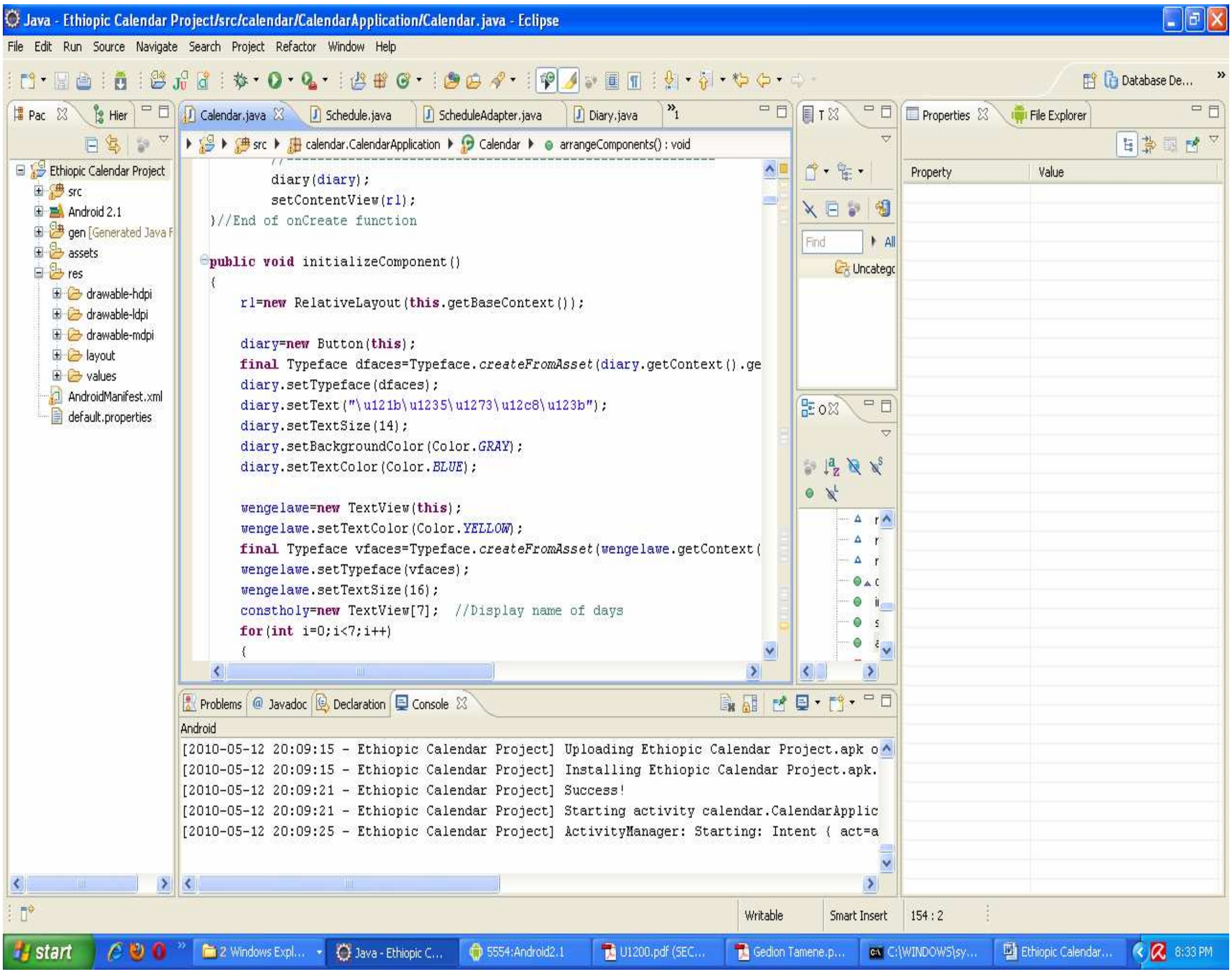


Figure 5.4 The Eclipse Integrated Development Environment

Once the code is written, it will be compiled and run on the Android emulator. The Android SDK includes a mobile device emulator - a virtual mobile device that runs on computer. The code written on Eclipse IDE will be compiled and run on the Emulator. The emulator lets to prototype, develop, and test Android applications without using a physical device. To model and test the application more easily, the emulator supports Android Virtual Device (AVD) which lets the user to specify the Android platform that the user wants to run on the emulator. For running this application Android 2.1 emulator is used. Figure 5.5 shows the calendar application on the Android 2.1 emulator.

Figure 5.5 Snapshot of the calendar system

As shown in figure 5.5 by default the calendar displays the current Ethiopian date, month and year. The month and the year are displayed on the respective box. The current date for the current month and year is shaded with red color. The calendar also displays any holiday that exists for the given month and year. As shown in figure 5.5 there is one public holiday on the month **ግንቦት** in the year 2002. For any given year it is possible to look the holidays by selecting the month on which this holiday falls. In order to see the dates and holidays of any month for any year, the user must select the month from the list. Figure 5.6 shows how the user selects the month from the list.

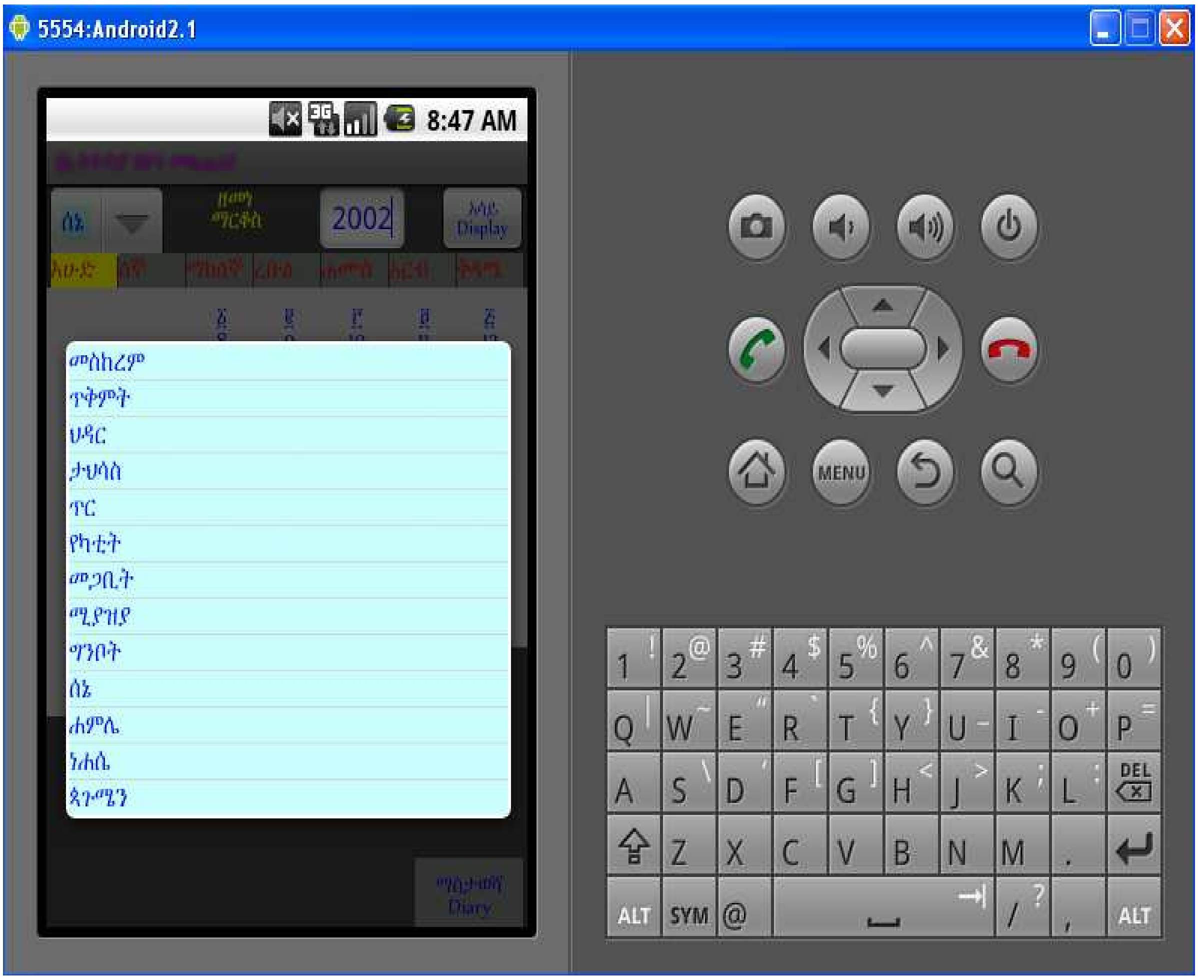
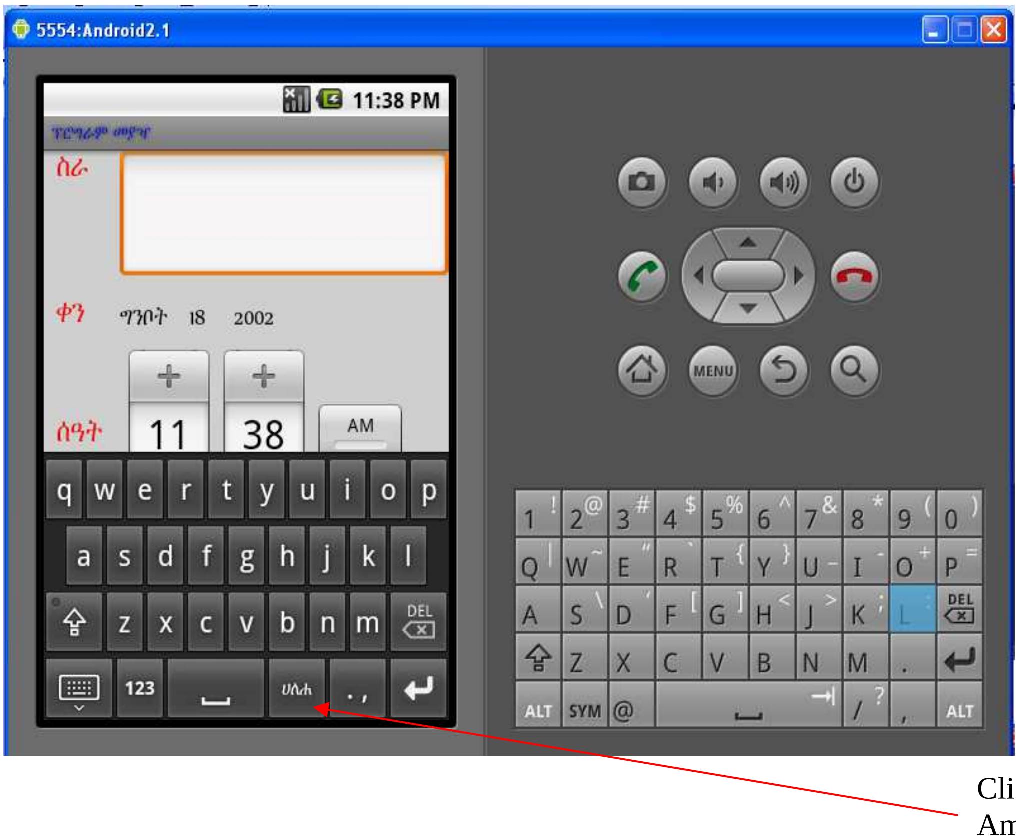


Figure 5.6 Snapshot of how to select month from the list

For any selected month again it is possible to display the dates and holidays by entering the year in the text box. Figure 5.7 shows how the user selects the year from the list.

Figure 5.7 Snapshot of how to enter the year in the textbox

Based on the calendars users can schedule their tasks and also keep diaries for any specific date, month and year. To schedule a task at some specific date, month and year, the user selects the month and enter year then click on specific date to open the form that allows the user to schedule task. Figure 5.8 shows the schedule form that is displayed when the user click on the specific date from the calendar.



Click to change to Amharic font

Figure 5.8 Snapshot of task scheduling form of English keyboard layout

After the task scheduling form is displayed, click on the task (**ስራ**) text box and change the language to Ethiopic by clicking on "**ሀሰሐ**" as shown in Figure 5.8. Once the font layout changed to Amharic, the user can write the task to be organized on the text box as shown in Figure 5.9.

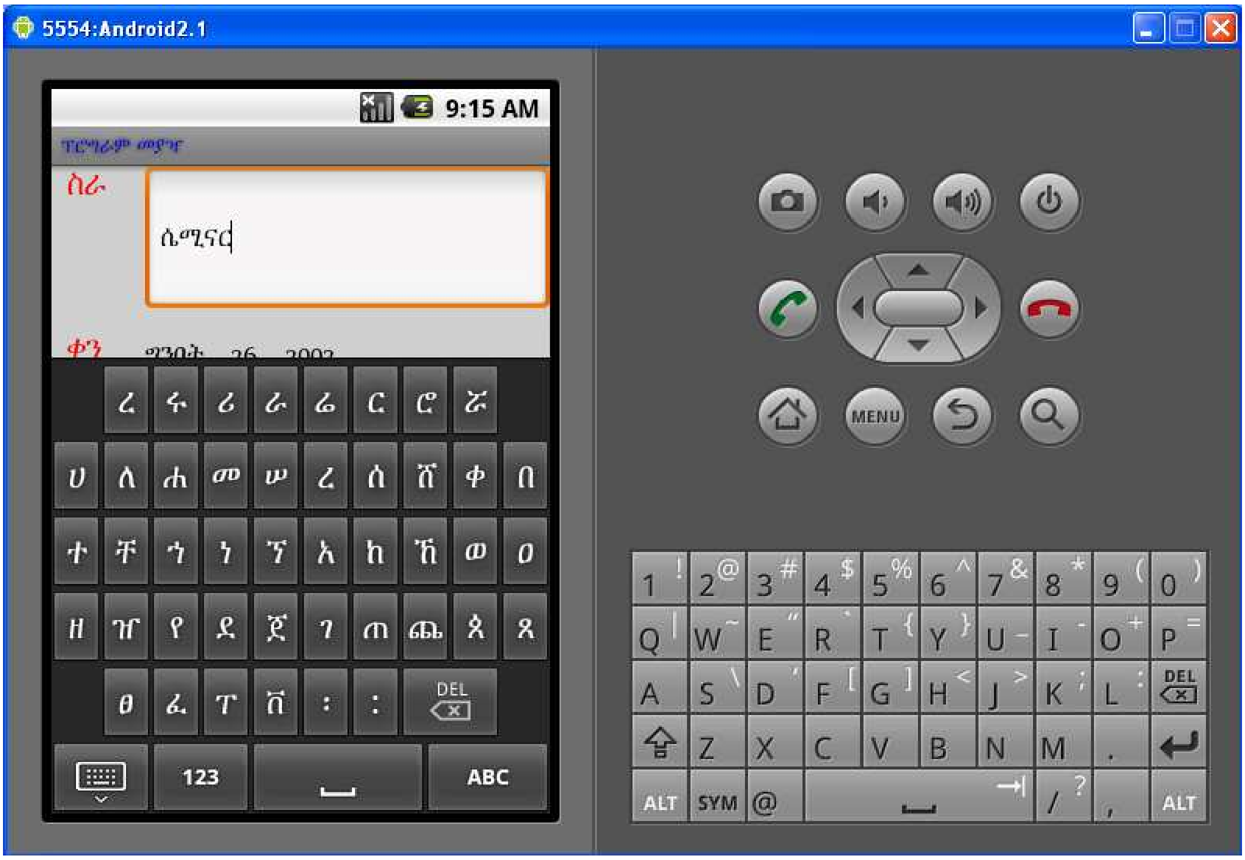
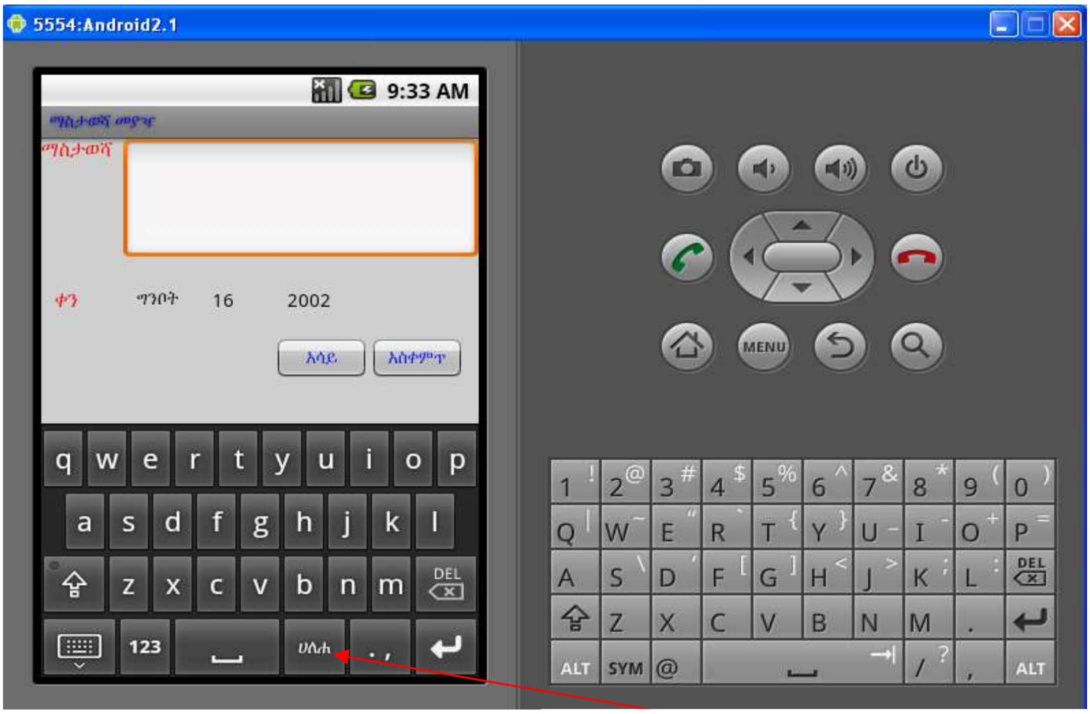


Figure 5.9 Snapshot of task scheduling form with Amharic keyboard layout

As shown in Figure 5.9, the user can write his task using Amharic keys and set the time for the schedule of the task. Based on the time arranged, the system alarms the user when the time of the schedule reaches.

The other functionality of a calendar provided on this application is to allow users to keep diary using Ethiopic script. To keep diary on the current date, the user clicks on "**ማስታወሻ**" button from the calendar. Then the diary form will be displayed with the current date, month and year. Figure 5.10 shows the diary form. Then the user changes the keyboard layout to Amharic as shown in Figure 5.10.



Click to change to Amharic keyboard layout

Figure 5.10 Snapshot of how to change to Amharic keyboard layout

As shown in Figure 5.11, the user writes the diary using Amharic texts.

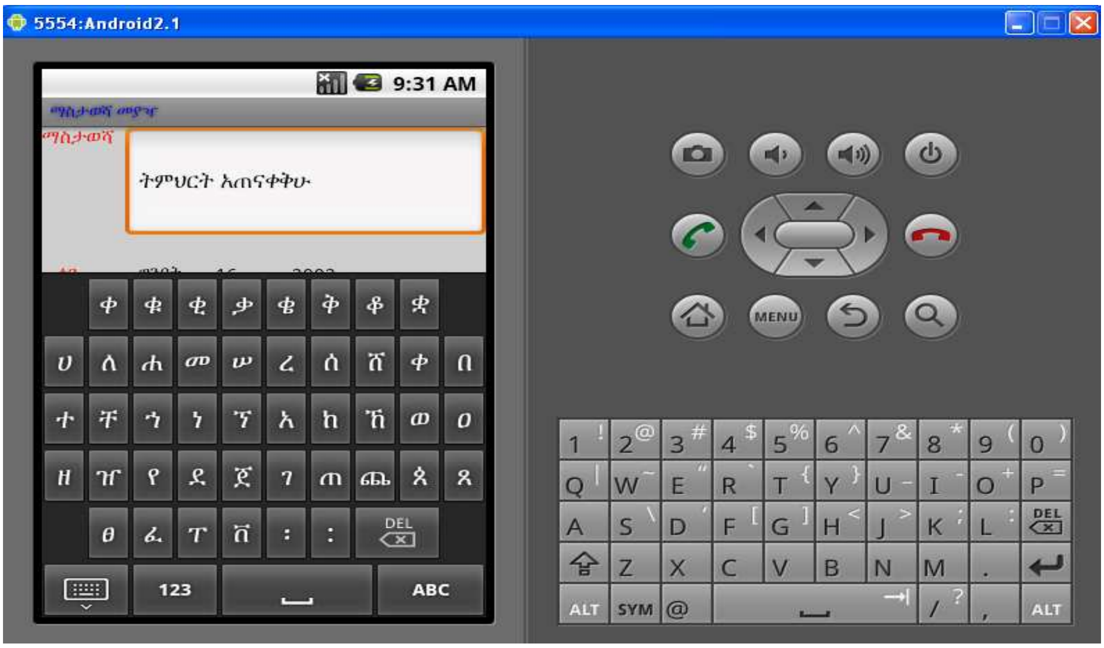


Figure 5.11 Snapshot of diary form with Amharic keyboard layout

Figure 5.12 shows the snapshot of how the calendar system is working in the real android Samsung mobile phone.

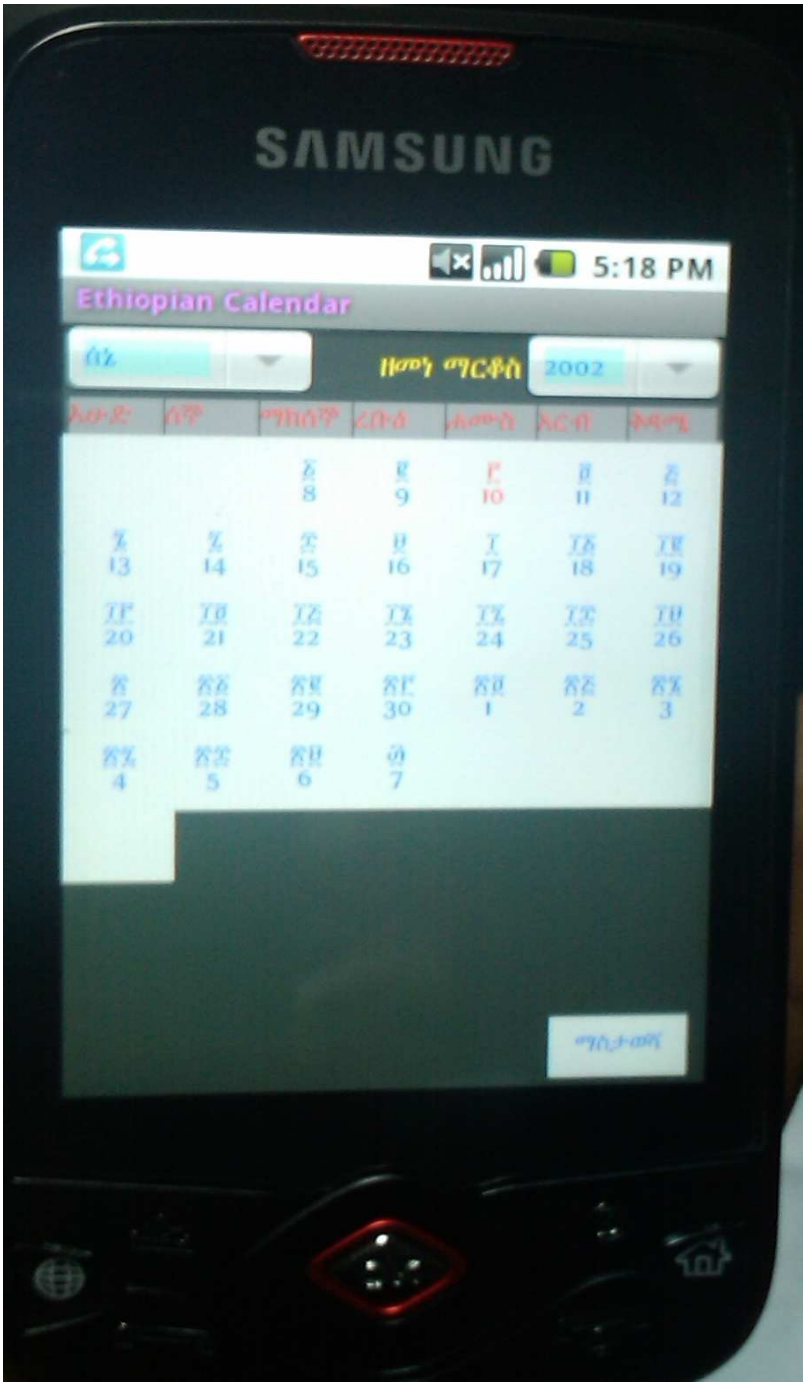


Figure 5.12 Snapshot of the calendar taken from the real mobile device

5.4 Testing of the application

In order to make sure that the system is working correctly, unit and system testing were applied. The first testing is testing on the calendar. As stated in the previous part, the Ethiopian calendar is based on the Ethiopian Orthodox church. The church uses its own calculation to create a calendar for the country. The correctness of the calendar is tested by the calculation stated in **ባህረ ሐሳብ (አቡሻኸር**). In addition to the calculation, the calendar is checked with other Ethiopian calendar system developed like Power Ge’ez calendar and Ethiopica calendar. According to this test, this calendar is working correctly. But, to check the corresponding Gregorian calendar for the Ethiopian calendar, the calendar of the computers operating system is used. To check the correctness of the Orthodox Christian holidays, the books [31, 32] have been used. Specially **ባህረ ሐሳብ (አቡሻኸር**) [31], calculates the holidays for 518 years (1962-2480), of this we have checked for 50 years.

Since Ethiopian Muslim holidays are based on the Islamic calendar, to test the Muslim holidays, we have used different Islamic calendar such as Saudi Arabia calendar [38], Salat Prayer calendar and paper based calendars that provide the holiday date. According to the test, the result shows that the system works correctly 100%.

CHAPTER SIX: CONCLUSION AND RECOMMENDATION

6.1 Conclusion

Android phones have only recently become a reality, and they are rapidly gaining popularity, to the point where approximately 60,000 of them are being shipped every day [33]. That means Android phones represent a significant and ever-growing portion of the overall market. As the market of Android continues to grow, more and more developers are being attracted to develop different application for it. The applications range from social games to more business-orientated options and this range will continue to increase in the coming years. As the distribution of this mobile phone increases from time to time, many users will have the access to use these mobile phones. Thus, developing and integrating applications like Ethiopian calendar system and organizer for Android phones has great importance for its users.

Ethiopia has a very ancient calendar system which is different from the Gregorian calendar system. Since there is a difference between the Gregorian and Ethiopian calendar system in terms of dates, months and years there should be a calendar system that provide both calendar system together. This project provided a perpetual Ethiopian calendar with the corresponding Gregorian calendar for any years. No mobile phone has integrated the Ethiopian calendar system so far. The integration of this application to Android phones provides the service for its users without the need of conversion from one system to another manually.

Most Ethiopian users use Ethiopian calendar system in their day to day activities. For instance in order to organize their tasks and keeping diary for some specific Ethiopian date, users uses this calendar system. So, this application supports users to consult their mobile calendar to organize and keep diary in Ethiopian calendar using Ethiopic scripts. Lastly the application was tested and the result shows that the system works perfectly and correctly.

6.2 Contribution

Though there are some Ethiopian calendar system, their algorithm is not accessible because of they are business oriented application. Therefore, we have developed our algorithms. Thus, the contribution of this project is summarized as follows:

- Designed algorithms to implement a perpetual Ethiopian calendar.
- Designed algorithms to compute Ethiopian holidays (religious and non-religious holidays).
- Designed algorithms to convert Ethiopic calendar to European or Gregorian Calendar
- Developed perpetual Ethiopic and Gregorian calendar based on the algorithms
- Designed and implemented a mobile organizer/scheduler system for Android phone that support Ethiopic script
- Designed and implemented a mobile diary system for Android phone that support Ethiopic script text entry.
- Tested the correctness and the validity of the developed system using several years of tested paper based and digital calendars.
- Demonstrated the usability of developed system on the real Android phones.

6.3 Future Work

This application is developed for Android based smart phones. So, as a future work it is good if the application will be generic inorder to run it in any mobile phone even personal computer.

References

[1]. Ethiopian calendar: http://www.selamta.net/Ethiopian%20Calendar.htm, Selamta, last accessed on January 27, 2010.

[2]. Abreham Tenaw, “Abushaher – Ye Ethiopia Yezemen Kemer”, Gondar printing press, Gondar, Ethiopia, December 2008.

[3]. Create a Simple Mobile Calendar System: http://www.wildtyping.com/how-to-create-a-simple-mobile-calendar-system, Wild typing, last accessed on February 4, 2010.

[4]. Calendar: http://www.pearsoninformation.co.uk/pdf/Calendar.pdf, Pearson Information Ltd, last accessed on February 4, 2010.

[5]. What is a diary? http://www.wisegeek.com/what-is-a-diary.htm, by Malcolm Tatum, last accessed on February 4, 2010.

[6]. Ethiopian Calendar year: http://ethiopica.sourceforge.net, last accessed on February 4, 2010.

[7]. Concepts Data Systems: http://conceptsdatasystems.com, Concepts data systems, last accessed on February 4, 2010.

[8]. Calendar systems: http://whitefiles.org/b1_s/1_free_guides/fg2cd/pgs/c01_clndrs.htm,last accessed on February 6, 2010.

[9]. The Ethiopian calendar-insight: Ethiopian millennium 2000 committee http://libraryy.nalis.gov.tt//Default.aspx?PageContentID=768&tabid=282, last accessed on February 6, 2010.

[10]. Working with the Android calendar, http://www.developer.com/ws/article.php/3850276/working-with-the-Android-calendar.htm, Developer.com , last accessed on February 14, 2010.

[11]. Ethiopian calendar, http://www.softpedia.com/get/Office-tools/Diary-Organizers-Calendar/Ethiopian-Calendar.html, Softpedia, last accessed on February 14, 2010.

[12]. Scott W. Ambler, The object primer: Agile model-driven development with UML 2.0, Cambridge: Cambridge University press, 2004.

[13]. Some other calendars: http://www.webexhibits.org/calendars/calendar-other.html, webexhibits.org, last accessed on March 24, 2010.

[14]. A Hebrew calendar application for a Java TM mobile device: http://kosherdev.com/wp-content/uploads/2009/11/MoladManual.pdf, last accessed on March 14, 2010.

[15]. Wikipedia, the free encyclopedia, http://en.wikipedia.org/wiki/Calendar, last accessed on February 14, 2010.

[16]. The Ethiopic calendar: http://www.ethiopic.com/calendar/ethiopic.htm, by Dr. Aberra Molla, last accessed on February 14, 2010.

[17]. An Ethiopian Journal, Ethiopian Astronomy/Zodiac: http://tseday.worldpress.com/2008/09/ 14/ethiopian-astronomyzodiac/, last accessed on February 14, 2010.

[18]. Ethiopian computers and software: http://www.ethiopic.com/calendar/ethiopic.htm, last accessed on April 4, 2010.

[19]. Ethiopianism: http://www.ethiopianism.net/?p=2357, www.geghna.org, last accessed on May 23, 2010.

[20]. The Ultimate Guide to Every Major Android Phone by Patrick Martin on March 22, 2010: http://www.Androidmobilephone.com/the-ultimate-guide-to-every-major-Android-phone #nexusone, last accessed on April 4, 2010.

[21]. http://www.mad-ip.eu/files/reports/Android.pdf, Martin Jantscher, by last accessed on April 14, 2010.

[22]. The Android operating system for cell phones and wireless mobile devices: http://www.Androidoperatingsystem.com/, last accessed on May 3, 2010.

[23]. Ethiopian and Coptic Calendar Systems for 4.4: http://www.layt.net/john/blog/odysseus/ethiopean_and_coptic_calendar_systems_for_44?page=1, John.layt.netlast accessed on May 7, 2010.

[24]. The Ethiopian calendar - an insight: http://library2.nails.gov.tt/default.aspx?pageconentid=768& tabid=282, last accessed on March 14, 2010.

[25]. Supported Calendar Systems: http://java.sun.com/javase/6/docs/technotes/guides/intl/calendar.doc.html, last accessed March 23, 2010.

[26]. What's up with the Ethiopian calendar and clock?: http://www.visit2ethiopia.com/calendar.htm, last accessed on February 14, 2010.

[27]. Gregorian calendar: http://www.answers.com/topic/gregorian-calendar, last accessed on May 1, 2010.

[28]. **የኢትዮጵያ መጽሐፍ ቅዱስ ማኅበር፡ መጽሐፍ ቅዱስ፡ የብሉይና የሐዲስ ኪዳን መጻሕፍት በብረሃንና ሰላም ማተሚያ ቤት ታተመ 1980 ዓ.ም**

[29]. Allen H. Dutoit & Bernd Bruegge, Object Oriented Software Engineering Conquering Complexity and Changing Systems, Prentice Hall, 2000.

[30]. **ዓስራት ገብረማሪያም እና ገብረሕይወት መሐሪ: የዘመን አቆጣጠር አዲስ አበባ 1987 ዓ.ም**

[31]. **አቡሻኽር (የቀን መቁጠሪያ) አዲስ አበባ 1962 ዓ.ም**

[32]. **ተስፋ ገብረ ሥላሴ: መዝገበ ጸሎት ባለ ሃምሳ ሶስት አርእስት (አይነት)፡ በተስፋ ማተሚያ ቤት ታተመ 1991 ዓ.ም**

[33]. Android Is Shipping 60,000 Phones Daily – Still Lagging Behind iPhone: http://thenextweb.com/mobile/2010/02/16/Android-shipping-60000-phones-single-day/, last accessed on May 15, 2010.

[34]. The Islamic Calendar, http://www.religionfacts.com/islam/holidays/calendar.htm, Religion Facts.com, last accessed on May 15, 2010.

[35]. Determination of the proper date to celebrate Eid al Adha, http://muslim-canada.org/eid_al_adha.html, last accesses on May 17, 2010.

[36]. Introduction to Android development, http://www.ibm.com/developerworks/opensource/library/os-android-devel/index.html, by Frank Ableson, last accesses on May 17, 2010.

[37]. Android, http://www.openhandsetalliance.com/android_overview.html, last accesses on May 17, 2010.

[38]. Saudi Arabia calendar, http://mycalendar.org/Holiday/Saudi%20Arabia/2010/, last accesses on May 17, 2010.

Appendix

List of words with their definition or description

ዘመን መለወጫ-አዲስ አመት	Ethiopian New Year
መስቀል	Finding of the True Cross
ልደት	Christmas
ጥምቀት	Epiphany
ስቅለት	Ethiopian Good Friday
ትንሳኤ-ፋሲጋ	Ethiopian Easter
ኢድ-አል-ፈጥር(ረመዳን)	Id Al Fater (Ramadan)
ኢድ-አል-አድሃ(ዐረፋ)	Id Al Adaha (Arefa)
የነቢዩ መሐመድ ልደት(መዉሊድ)	Birthday of the prophet Mohammed (Maulide)
የዓድዋ ድል መታሰቢያ	Victory of Adwa
የኢትዮጲያ አርበኞች መተሰሰቢያ	Patriots Victory Day
ዓለም አቀፍ የላብአደሮች ቀን	Labour Day
ኢሕአዴግ ደርግን ያስወገደበት	Downfall of the Dergue
ማስታወሻ	Diary
መስከረም	September
ጥቅምት	October
ህዳር	November
ታህሳስ	December
ጥር	January
የካቲት	February
መጋቢት	March
ሚያዝያ	April
ግንቦት	May
ሰኔ	June
ሐምሌ	July
ነሐሴ	August
ጳጉሜን	The thirteenth month of Ethiopia

ዓመተ ምህረት	The year after the birth of Christ
ዓመተ ዓለም	the years of the world (5500 + year of mercy)
ዓመተ ፍዳ	The year before the birth of Christ
ወንጌላዊ	Gospels
ዓመተ ወንጌላዊ	The year of Gospels
ዘመነ ማቴዎስ	Year of Matthew
ዘመነ ማርቆስ	Year of Mark
ዘመነ ሉቃስ	Year of Luke
ዘመነ ዮሐንስ	Year of John
ጠቅላላ ዓመት	Total year (the years of the world + the year of Gospels + 2).
እሁድ	Sunday
ሰኞ	Monday
ማክሰኞ	Tuesday
ረቡዕ	Wednesday
ሐሙስ	Thursday
አርብ	Friday
ቅዳሜ	Saturday
መደብ	The remainder of the year of the world divided by 19
ወንበር	The result of subtraction of one from **መደብ** (**መደብ**-1)
አበቅቴ	The result of the multiplication of by 11 (**ወንበር** * 11)
መጥቅ	The result of subtraction of **አበቅቴ** from thirty **(30-አበቅቴ)**
በዓለ መጥቅ	The celebration date of **መጥቅ**
የዕለት ተዉሳክ	Constant value given for each day
የጾም ተዉሳክ	Constant value given for each Christian holiday
መባጃ ሐመር	**መጥቅ** + **የዕለት ተዉሳክ**
ነነዌ	Value of **መባጃ ሐመር**
ጾም መግቢያ	Starting date of fasting
ደብረ ዘይት	Half of the fasting period
ሆሳዕና	The week before the end of the fasting period
ስራ	Task

Declaration

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

Declared by:

Name: ____

Signature: ____

Date: ____

Confirmed by advisor:

Name: ____

Signature: ____

Date: ____

Place and date of submission: Addis Ababa, June 2010.