IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 6, Ver. III (Nov - Dec 2017), PP 01-07
www.iosrjournals.org

The impact of using Gregorian calendar dates in systems that adapt localization: In the case of Ethiopia

Getnet Mossie Zeleke[1], Metages Molla Gubena[2]
[1,2]*Department of Information Technology, College of Technology, Debre Markos University, Ethiopia*

Abstract: *Ethiopian calendar has 13 months in which 12 months have 30 days equal and the 13th month has 5 or 6 days length. Date is one of the inputs for web or desktop applications. Java Development Kit(JDK) and Joda-Time Date Time package have been used for date manipulation in java based applications developed for local use in Ethiopia. Besides, Gregorian calendar date time pickers have also been used in web applications. This leads the application developers not to fully adapt localization. To fill this gap we developed JavaScript Date Picker and date manipulator java package in Ethiopian calendar basis. The first product consists of Amharic week day and month names which enable users to pick Ethiopian date as an input in web applications. The second product is used to manipulate Ethiopian date in java desktop and web applications. In the package different methods are defined to perform date related activities such as date calculations, extraction of date element in a given date, presentation of date in different date formats like መስከረም 12, 2010 etc. Unlike other date time java packages, dealing with Ethiopian dates using our date package does not require date conversion. Testing of the products produced the expected result.*

Keywords*: Amharic Month Name, Date Manipulator java package, Date Picker, Ethiopian Calendar, Localization*

Date of Submission: 27-10-2017 Date of acceptance: 16-11-2017

I. INTRODUCTION

The vast growth and improvement of technology has great impact in the day to day activities of human being. Now a day organizations in the world use new technologies and automated systems to simplify their tasks, to increase their productivity, to be competitive in the market and increase their customer satisfaction. International organizations adapt internalization during design and development of their systems. On the other hand systems designed and used in local organizations adapt localization.

Internationalization is the process of designing a software application so that it can be adapted to various languages and regions without engineering changes. [1]

Localization is means of adapting computer software to different languages, regional differences and technical requirements of a target locale. Localization is the process of adapting internationalized software for a specific region or language by adding locale-specific components and translating text. [1]

Ethiopia uses Amharic language as national language. Therefore, organizations in the country use this language for communication in their business process. Ethiopia has also its own calendar and government organizations use this calendar to plan and perform office tasks.

The Ethiopian Calendar has more in common with the Coptic Egyptian Calendar. The Ethiopic and Coptic calendars have 13 months, 12 of 30 days each and an intercalary month at the end of the year of 5 or 6 days depending whether the year is a leap year or not. 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 year starts on 11 September in the Gregorian calendar or on the 12th in (Gregorian) Leap Years. [2] There is a 7 year gap (from September to December) and 8 years gap (from January to August) between the Ethiopian and Gregorian Calendars.

Organizations in Ethiopia are currently moving towards automated systems or business applications to run their businesses effectively and utilize resources efficiently. Most of these systems take date like date of birth, promotion date, recruitment date etc as input. The systems also need to perform different date calculations for instance calculating age, service year of an employee etc. Date is also one of the system outputs that can be used in letters or report of an organization.

II. PROBLEM STATEMENT

Localization is tried to be adapted by system developer in most systems used by local organizations in Ethiopia. But some of the graphical interfaces, inputs given to the system and outputs generated from the system are not localized. For instance Gregorian calendar is used in most system than Ethiopian Calendar. So date inputs to the system, date calculations in the system and date outputs from the system have been dealt using Gregorian calendar. Due to this reason, the systems that are developed for local use have the following limitations:

a) The systems by themselves do not meet one of the requirements of localization because the Month and Day names displayed in the Gregorian calendar are not written in official language(Amharic) of Ethiopia.

b) Organizations are enforced to use the Gregorian calendar dates which is not appropriate for local communication via letters generated from the system.

c) Date inputs like date of birth, due date, date of recruitment, issue date, employee promotion date, etc to a system needs to be converted into Gregorian calendar manually or using third party tools before it is fetched to the system.

d) Date manipulations such as date difference between two Ethiopian dates, number of weekend days between two Ethiopian dates, getting Ethiopian current date, day, month, year and local letter date format is not possible using the built-in Java Development Kit(JDK) date methods, different date methods in JDK are implemented using Gregorian calendar. That means there is no method that accepts Ethiopian date as it is and deal with it.

III. Main Objective

The main objective of this project is to design JavaScript Ethiopian Calendar Date Picker and Ethiopian Date Manipulator Java Package so as to reduce the impact of using Gregorian calendar in Ethiopia local systems development.

IV. Previous Related Work

There are different online JavaScript Ethiopian Calendar Date Pickers. For instance online interactive Ge'ez calendar converter [7] and Ethiopian Online Calendar [8] are some. These calendars are designed to show the Ethiopian Calendar and Holidays in the calendar, allow users to print it and perform conversion to other calendar date such as Gregorian Date. Java Development Kit (JDK) contains interfaces, classes and methods which are used to perform different mathematical operations, String, Array, Character, Date manipulations and other tasks. The existing JDK classes (such as java.util.Date and SimpleDateFormatter) aren’t thread-safe, leading to potential concurrency issues for users—not something the average developer would expect to deal with when writing date-handling code.[4] Joda-Time provides a quality replacement for the Java date and time classes. Users are now asked to migrate to java.time (JSR-310).Joda-Time is the de facto standard date and time library for Java prior to Java SE 8. The design allows for multiple calendar systems, while still providing a simple API. The “default” calendar is the ISO8601 standard which is used by many other standards. The Gregorian, Julian, Buddhist, Coptic, Ethiopic and Islamic calendar systems are also included. Supporting classes include time zone, duration, format and parsing.[5]

EthiopicChronology is a class in Joda-Time package that implements the Ethiopic calendar system.[6] Joda-Time date time library enables programmers to provide Gregorian Date and get equivalent date in others date Calendar system defined. For example if a Gregorian date (midday September 10, 2017 (ISO))is given, it will return the equivalent date in Ethiopian Calendar system date which is 2009-13-05T12:00:00.000+03:00
But programmers who develop systems in Ethiopia for local use require a package or class that can directly take Ethiopian date into their system, extract day or month or year from the given Ethiopian date, present Ethiopian Date in different format and get difference between two Ethiopian dates without converting it to Gregorian calendar system date. Therefore, Joda-Time has not solved the needs of software developers who want to deal with Ethiopian Calendar in local system.

V. Implementation And Testing

In order to solve the problems stated above, using Gregorian date as a baseline we designed

1. JavaScript Ethiopian CalendarDate Picker with weekend days on and off in JavaScript so that it can be easily used in web applications.
2. Ethiopian Date Manipulator Java Package that can be used by Java application developers. This package includes a class called EthiopianDate with different methods which are defined to deal with Ethiopian Dates.

1. JavaScript Ethiopian Calendar Date Picker

It can be embedded to a web application like any other external JavaScript file would. For example to include a calendar with weekend days on

i. *<head><script type=”text/JavaScript” src=”EthiopianCalendarDatePicker.js” charSet=”UTF-8”>*

</head> and to include a calendar with weekend days off.

ii. *<head><script type=”text/JavaScript” src=”EthiopianCalendarDatePickerWeekendDaysOff.js” charSet=”UTF-8”></head>*Here weekend days cannot be picked.

Once the file is included it requires one of the functions to be revoked where the date is required to be used as an input. The calendar generates a calendar that consists of years ranging 1920 to (CurrentYear+5), 13 months (1-12 month include 30 days and the 13th month includes 5/6 days) with Amharic name and 7 week days with short Amharic name.

Table 1. Months

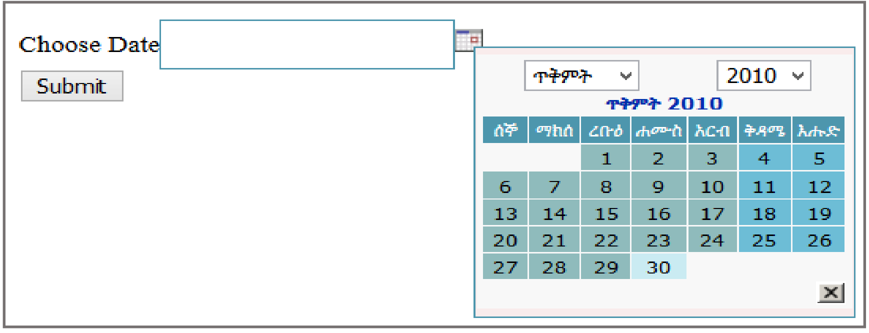
Ge'ez, Amharic, and Tigrinya (with Amharic suffixes in parentheses)	**Julian (Old Calendar) Start Date**	**Gregorian Start Date [From March 1900 to February 2100]**	**Gregorian Start Date in Year after Ethiopian Leap Day**
Mäskäräm (መስከረም)	August 29	September 11	September 12
Ṭəqəmt(i) (ጥቅምት)	September 28	October 11	October 12
Ḫədar (ኅዳር)	October 28	November 10	November 11
Taḫśaś (ታኅሣሥ)	November 27	December 10	December 11
Ṭərr(i) (ጥር)	December 27	January 9	January 10
Yäkatit (Tn. Läkatit) (የካቲት)	January 26	February 8	February 9
Mägabit (መጋቢት)	February 25	March 10	March 10
Miyazya (ሚያዝያ)	March 27	April 9	April 9
Gənbot (ግንቦት)	April 26	May 9	May 9
Säne (ሰኔ)	May 26	June 8	June 8
Ḥamle (ሐምሌ)	June 25	July 8	July 8
Nähase (ነሐሴ)	July 25	August 7	August 7
Ṗagʷəmen/Ṗagume(ጳጐሜን/ጳጉሜ)	August 24	September 6	September 6

These dates are valid only from March 1900 to February 2100. This is because 1900 and 2100 are not leap years in the Gregorian calendar, while they are still leap years in the Ethiopian calendar, meaning dates before 1900 and after 2100 will be offset.[3]

Table2. Week day name in Amharic

Ethiopian Day Names (With Amharic Suffix in parenthesis)	**Gregorian Day Name**
Segno (ሰኞ)	Monday
Maksegno (ማክሰኞ)	Tuesday
Rebu (ረቡዕ)	Wednesday
Hamus (ሐሙስ)	Thursday
Arb (ዓርብ)	Friday
Kidame (ቅዳሜ)	Saturday
Ehud (እሑድ)	Sunday

Testing the Ethiopian Calendar Date Picker



Choose Date

Submit

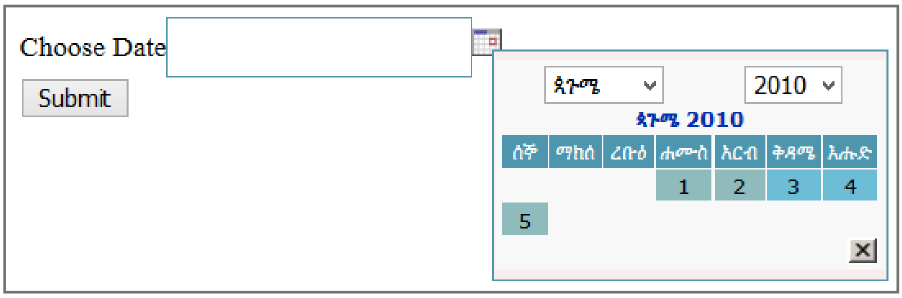
ጥቅምት ⌄ 2010 ⌄

ጥቅምት 2010

ሰኞ	ማክሰ	ረቡዕ	ሐሙስ	አርብ	ቅዳሜ	እሑድ
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

☒

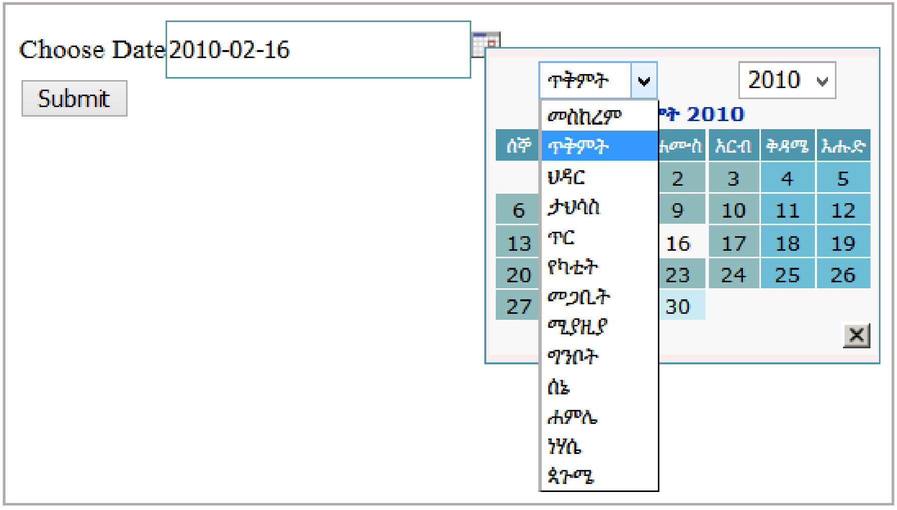
Figure 1. Ethiopian Calendar Date picker that show current date information



Choose Date

ሰኞ	ማክሰ	ረቡዕ	ሐሙስ	አርብ	ቅዳሜ	እሁድ
			1	2	3	4
5						

Figure 2. Ethiopian Calendar Date Picker showing the 13th month of the selected year



Choose Date 2010-02-16

Figure 3. Ethiopian Calendar Date Picker with selected date

II. Ethiopian Date Manipulator Java Package

The Java Package we developed for Ethiopian calendar system date manipulator named by *EthiopianDateGetnetMetages*.jar includes the methods described below. It can be imported to a java desktop or web application whenever required. Example:By importing the jar file into the application as follows

1. *<%@ page import=" EthiopianDateGetnetMetages .EthiopianDate"%>* in java web application or
2. *importEthiopianDateGetnetMetages.EthiopianDate;* in java desktop application developers will be able to access the methods.

a) getEthiopianCurrentDate() method returns the current Ethiopian date in YYYY-MM-DD format.
b) getEthiopianCurrentDay() method returns current day in integer or string format.
c) getEthiopianCurrentDayNameEnglish() method returns current day name in English.
d) getEthiopianCurrentDayNameAmharic() method returns current day name in Amharic.
e) getEthiopianCurrentMonth() method returns current month in integer or string format.
f) getEthiopianCurrentMonthNameEnglish() method returns current month name in English.
g) getEthiopianCurrentMonthNameAmharic() method returns current month name in Amharic.
h) getEthiopianCurrentYear () method returns current Ethiopian year.
i) getEthiopianDay(EthiopianDate) method extracts the day from the given date and returns it.
j) getEthiopianMonth(EthiopianDate) method extracts the month from the given date and returns it.
k) getEthiopianYear(EthiopianDate) method extracts the year from the given date and returns it.
l) getEthiopianDayNameAmharic(EthiopianDate) method extracts the day from the given date, determine Amharic month name and returns it.
m) getEthiopianDayNameEnglish(EthiopianDate) method extracts the day from the given date, determine Amharicmonth name and returns it.
n) getEthiopianMonthNameAmharic(EthiopianDate) method extracts the month from the given date, determine Amharic month name and returns it.
o) getEthiopianMonthNameEnglish(EthiopianDate) method extracts the month from the given date, determine Amharic month name and returns it.
p) getEthiopianDateDifference(StartDate, EndDate) method takes two Ethiopian dates as argument and returns their difference in days without converting the dates into Gregorian Dates. EndDate must be greater than StartDate.

q) getTotalSaturdays(StartDate,EndDate) method returns the total number of available Saturdays between two Ethiopian dates.

r) getTotalSundays(StartDate,EndDate) method returns the total number of available Sundays between two Ethiopian dates.

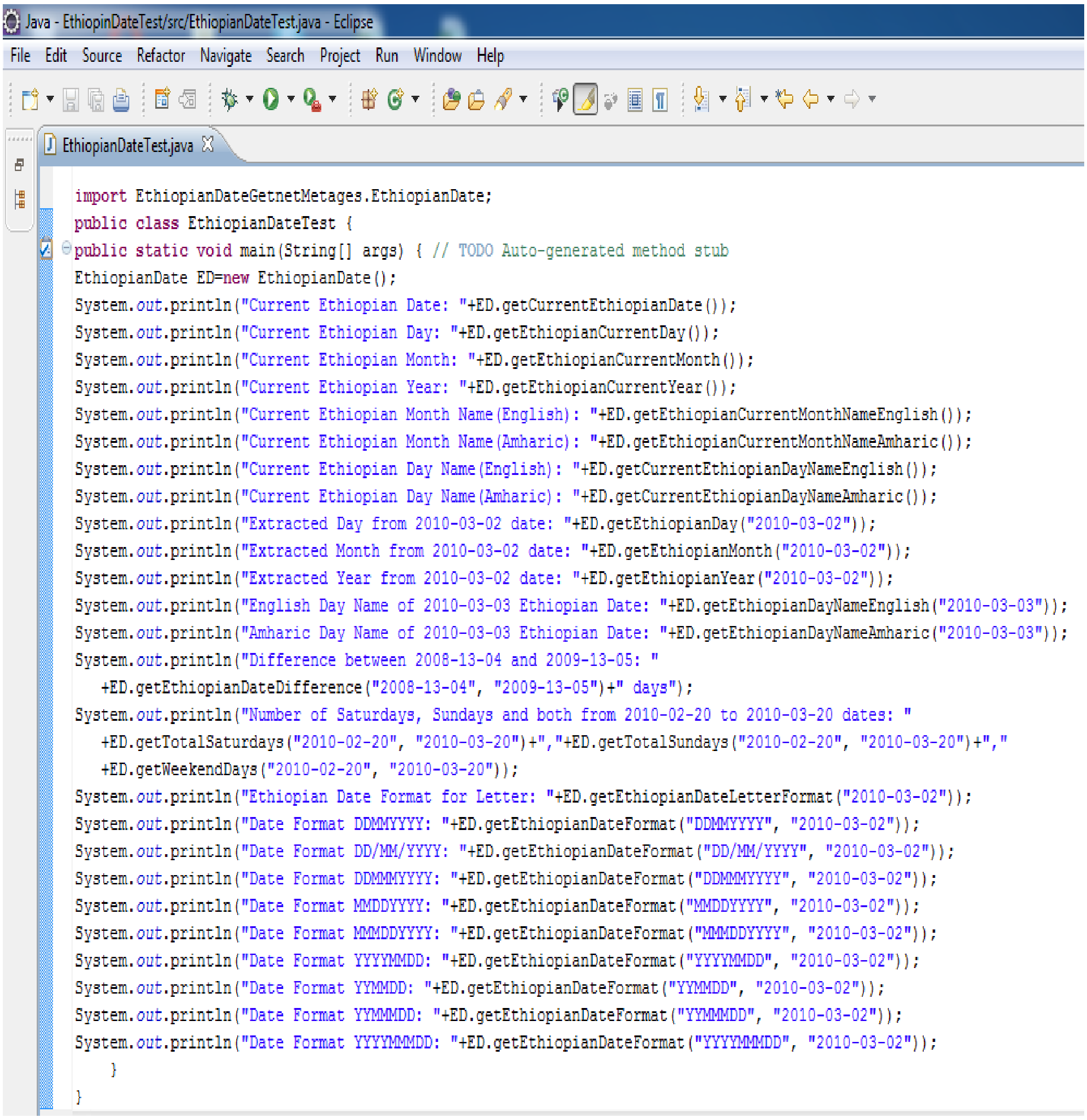
s) getTotalWeekendDays(StartDate,EndDate) method returns the total number of available weekend (Saturday and Sunday) days between two Ethiopian dates.

t) getEthiopianDateLetterFormat (EthiopianDate) method returns a date in DD-MM-YYYY format.

u) getEthiopianDateFormat(SelectedFormat,EthiopianDate) method returns a date in the given date format. The date formats include DDMMYYYY, DD/MM/YYYY, DDMMMYYYY, MMDDYYYY, MMMDDYYYY, YYYYMMDD, YYMMDD, YYMMMDD and YYYYMMMDD.

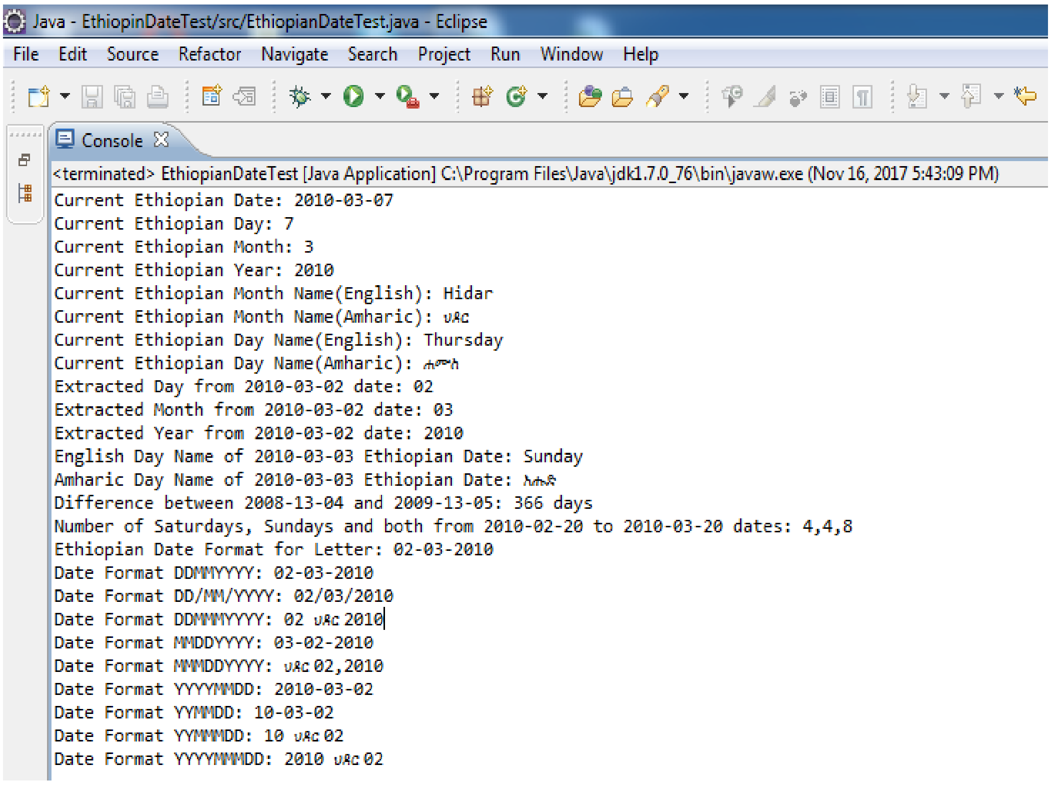
Testing Ethiopian Date Manipulator Java Package

As shown on Fig. 4. the methods mentioned above can be accessed via the object of the class created in the jar file. Example: EthiopianDate ED=new EthiopianDate();


Java - EthiopinDateTest/src/EthiopianDateTest.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
EthiopianDateTest.java

```
import EthiopianDateGetnetMetages.EthiopianDate;
public class EthiopianDateTest {
public static void main(String[] args) { // TODO Auto-generated method stub
EthiopianDate ED=new EthiopianDate();
System.out.println("Current Ethiopian Date: "+ED.getCurrentEthiopianDate());
System.out.println("Current Ethiopian Day: "+ED.getEthiopianCurrentDay());
System.out.println("Current Ethiopian Month: "+ED.getEthiopianCurrentMonth());
System.out.println("Current Ethiopian Year: "+ED.getEthiopianCurrentYear());
System.out.println("Current Ethiopian Month Name(English): "+ED.getEthiopianCurrentMonthNameEnglish());
System.out.println("Current Ethiopian Month Name(Amharic): "+ED.getEthiopianCurrentMonthNameAmharic());
System.out.println("Current Ethiopian Day Name(English): "+ED.getCurrentEthiopianDayNameEnglish());
System.out.println("Current Ethiopian Day Name(Amharic): "+ED.getCurrentEthiopianDayNameAmharic());
System.out.println("Extracted Day from 2010-03-02 date: "+ED.getEthiopianDay("2010-03-02"));
System.out.println("Extracted Month from 2010-03-02 date: "+ED.getEthiopianMonth("2010-03-02"));
System.out.println("Extracted Year from 2010-03-02 date: "+ED.getEthiopianYear("2010-03-02"));
System.out.println("English Day Name of 2010-03-03 Ethiopian Date: "+ED.getEthiopianDayNameEnglish("2010-03-03"));
System.out.println("Amharic Day Name of 2010-03-03 Ethiopian Date: "+ED.getEthiopianDayNameAmharic("2010-03-03"));
System.out.println("Difference between 2008-13-04 and 2009-13-05: "
   +ED.getEthiopianDateDifference("2008-13-04", "2009-13-05")+" days");
System.out.println("Number of Saturdays, Sundays and both from 2010-02-20 to 2010-03-20 dates: "
   +ED.getTotalSaturdays("2010-02-20", "2010-03-20")+","+ED.getTotalSundays("2010-02-20", "2010-03-20")+","
   +ED.getWeekendDays("2010-02-20", "2010-03-20"));
System.out.println("Ethiopian Date Format for Letter: "+ED.getEthiopianDateLetterFormat("2010-03-02"));
System.out.println("Date Format DDMMYYYY: "+ED.getEthiopianDateFormat("DDMMYYYY", "2010-03-02"));
System.out.println("Date Format DD/MM/YYYY: "+ED.getEthiopianDateFormat("DD/MM/YYYY", "2010-03-02"));
System.out.println("Date Format DDMMMYYYY: "+ED.getEthiopianDateFormat("DDMMMYYYY", "2010-03-02"));
System.out.println("Date Format MMDDYYYY: "+ED.getEthiopianDateFormat("MMDDYYYY", "2010-03-02"));
System.out.println("Date Format MMMDDYYYY: "+ED.getEthiopianDateFormat("MMMDDYYYY", "2010-03-02"));
System.out.println("Date Format YYYYMMDD: "+ED.getEthiopianDateFormat("YYYYMMDD", "2010-03-02"));
System.out.println("Date Format YYMMDD: "+ED.getEthiopianDateFormat("YYMMDD", "2010-03-02"));
System.out.println("Date Format YYMMMDD: "+ED.getEthiopianDateFormat("YYMMMDD", "2010-03-02"));
System.out.println("Date Format YYYYMMMDD: "+ED.getEthiopianDateFormat("YYYYMMMDD", "2010-03-02"));
    }
}
```

Figure 4. Sample code to show how to use the methods defined in the jar file



Java - EthiopinDateTest/src/EthiopianDateTest.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Console

```
<terminated> EthiopianDateTest [Java Application] C:\Program Files\Java\jdk1.7.0_76\bin\javaw.exe (Nov 16, 2017 5:43:09 PM)
Current Ethiopian Date: 2010-03-07
Current Ethiopian Day: 7
Current Ethiopian Month: 3
Current Ethiopian Year: 2010
Current Ethiopian Month Name(English): Hidar
Current Ethiopian Month Name(Amharic): ህዳር
Current Ethiopian Day Name(English): Thursday
Current Ethiopian Day Name(Amharic): ሐሙስ
Extracted Day from 2010-03-02 date: 02
Extracted Month from 2010-03-02 date: 03
Extracted Year from 2010-03-02 date: 2010
English Day Name of 2010-03-03 Ethiopian Date: Sunday
Amharic Day Name of 2010-03-03 Ethiopian Date: እሑድ
Difference between 2008-13-04 and 2009-13-05: 366 days
Number of Saturdays, Sundays and both from 2010-02-20 to 2010-03-20 dates: 4,4,8
Ethiopian Date Format for Letter: 02-03-2010
Date Format DDMMYYYY: 02-03-2010
Date Format DD/MM/YYYY: 02/03/2010
Date Format DDMMMYYYY: 02 ህዳር 2010
Date Format MMDDYYYY: 03-02-2010
Date Format MMMDDYYYY: ህዳር 02,2010
Date Format YYYYMMDD: 2010-03-02
Date Format YYMMDD: 10-03-02
Date Format YYMMMDD: 10 ህዳር 02
Date Format YYYYMMMDD: 2010 ህዳር 02
```

Figure 5. Output for the sample program shown on Fig. 4

VI. Benefits Of The Products

For System Developers:

Making readymade JavaScript Ethiopian Date Picker and Ethiopian date manipulator java package available will encourage developers to design applications in Amharic language which in turn helps the developer to adapt localization. Using these products developers will be able to easily accomplish date related activities such as date calculations, extracting date components, presenting date in different date formats by calling the appropriate methods defined the package.

For end users:

Our products will increase the chance to fully adapt localization in ethiopia. Thus local sytems can be more convnient and interactive for end user. In particular,end users will not be enforced to manually change date from Ethiopian Calendar to Gregorian calendar in order to fetch date input to the system. End users can enter Ethiopian date as it is to the system and the system can generate letters, reports that consist of Ethiopian date from the system.

VII. Conclusion

In the world most countries have their own language and some have unique calendar systems. Ethiopia is one of the country which has its own national language called Amharic and has unique calander system.In systems designed in countries like Ethiopia for local use, adapting localization is recommended in order to make the system easy to understand and interact with. Developers have tried to apply localization concept in most systems used in Ethiopia. But in these systems Gregorian calendar,Java Development Kits(JDK) and Joda-Time date manipulator methods have been used to perform date related activies instead of Ethiopian Calendar. We believe that this is not appropraite specially for system that fully adapt localization because it will have impact on the usabilityof the system. So in this project taking Gregorian calendar as a standard and baseline we developed Ethiopian calendar Date Time Picker using JavaScript that can be used by web application developers. We also developed Ethiopian date manipulator java package which can be imported in both desktop and web based java applications. The products allow business application developers to easily accomplish date related activities in Ethiopian Calendar basis, help end users to fetch Ethiopian date to the system without converting it to Gregorian date and enable the organization to use the formal Ethiopian Calendar dates in letters, reports etc that could be generated from their systems. The project also motivate busines application developers who are from countries having their own calendar and language to meet localization requirment in systems they design for local use.

Reference

[1]. Bert Esselink,A Practical Guide to Localization,,(John Benjamins Publishing, [2000])

[2]. Ethiopian Calendar url:http://www.ethiopian-online.com/Ethiopian%20Calendar.htm,Accessed:November 07,2017

[3]. Ethiopian Calendar, url:https://en.wikipedia.org/wiki/Ethiopian_calendar, Accessed: November 09,2017

[4]. Ben Evans and Richard Warburto, Java SE 8 Date and Time, url:http://www.oracle.com/technetwork/articles/java/jf14-date-time-2125367.html, Accessed: September 12,2017

[5]. Stephen Colebourne, Joda Time, url:http://www.joda.org/joda-time/, Accessed: September 12, 2017

[6]. Brian S O'Neill & Stephen Colebourne, Class EthiopicChronology, url:https://docs.jboss.org/jbossas/javadoc/ 7.1.2.Final /org/joda/ time /chrono/EthiopicChronology.html, Accessed: November 10, 2017

[7]. Ethiopian Calendar, url:http://www.ethiopiancalendar.net/, Accessed: November 10, 2017.

[8]. Ethiopian Online Calendar, url:http://www.selamta.net/Ethiopian%20Online%20Calendar.htm, Accesses: November 10, 2017.

IOSR Journal of Computer Engineering (IOSR-JCE) is UGC approved Journal with Sl. No. 5019, Journal no. 49102.

Getnet Mossie Zeleke & Metages Molla Gubena The impact of using Gregorian calendar dates in systems that adapt localization: In the case of Ethiopia. "IOSR Journal of Computer Engineering (IOSR-JCE), vol. 19, no. 6, 2017, pp. 01-07