VERSION 1.0

February 17, 2016

Remind me

Supervised by: Mr. Md. Abdus Salim Mollah

Associate Professor

Department of Computer Science & Engineering

Khulna University of Engineering & Technology

Presented by: Md. Ashad-Uz-Zaman

(Roll: 1207004)

A. B. M. Sayem Bin Bari

(Roll: 1107040)

Subrata Dutta

(Roll: 1007002)

ABSTRACT

Electronic reminders based on calendars in smart phones are effective and popular. However such reminder is generally activated by time and not by location. In many situations tasks are only meaningful to be performed at a particular location, so it would be useful if reminders for those tasks can be activated when the person to be reminded is physically near or located at that location. In this project, a new application (app) is developed for Android-based smartphones and tablets. With location reminder **Remind Me** app the users can store their tasks and associate them to a particular location and when the user passes close to it, the app reminds them of the task. So, with the help of **Remind Me**, user gets reminder of what to do, when to do and at what location. The app has basic Time Based Reminder as an additional feature.

Any kind of criticisms about the development of the app will be highly appreciated.

Table of Contents

1. Background & Rationale:

In contemporary society, many people are overwhelmed by the number of tasks that need to be accomplished, these tasks are of different types, ranging from every day based meetings at work, and non-daily based such as buying groceries, paying bills etc. To help ourselves to remember these tasks, basic practices are to take notes on the paper based day planners or use post-its or use a personal task management software on computers and/or smartphones. As smartphones are used as an alternative to personal computers, taking notes on smartphones is a more convenient choice.

Similar to programs on a personal computer, applications (app) can be downloaded and installed on a smartphone. Similar to a personal computer there exists a wide selection of proprietary and open source mobile operating system platforms. Out of these smartphone operating systems, the most prominent ones are Apple iOS , Google Android, Symbian from Symbian foundation, RIM blackberry OS and Windows mobile from Microsoft.

This project focuses on developing a mobile application, **Remind Me** for Google's Android operating system. The majority of smartphone manufacturers such as Samsung, Motorola and HTC have adopted Android as the operating system for their products. As indicated by App Brain statistics, there are more than 1.4 million applications accessible as of February 2015, of which approximately 1.3 million are free and 0.2 million are paid [15]. On July 24, 2013 Google declared that Play store had one million applications and had seen more than 50 billion downloads. Consistently more than 1 million new Android devices are activated daily around the world. The number of downloads from Google play store every month is more than 1.5 billion.

This project is developed for Android Based smart phones on Android Studio, and IDE for developing applications on the Android platform. The user interface of this project is prototyped by using XML and prototyping tool. This project utilizes Location Based Services for creating location based reminders. More on Android, Android Studio, XML and Location Based Services is discussed in sections 1.1, 1.2, 1.3 and 1.4.

1.1 Android:

Android is a mobile operating system based on Linux. Android is primarily used for smartphones, tablet computers, watches, smart glasses, home appliances, cars, cameras, game consoles and mirrors. Android is available in 46 languages and powers millions of mobile devices in more than 190 countries around the world [4]. Android provides a world class platform for developing apps and games for Android users everywhere, as well as an open market place for distributing them instantly . The vast majority of Android applications are created in java, which is a standout amongst the most generally used programming languages around the world. **Remind Me** is compatible on different versions of Android, as starting from the minimum SDK version of Android 4.0 (Ice Cream Sandwich) to recent update Android 5.0 (Lollipop). The development environment used for **Remind Me** is Android Studio IDE.

1.2 Android Studio:

Android Studio is an IDE for developing on the Android platform. It is free and open source software. The Intelligent code editor in the Android Studio helps the user to be more productive while developing apps. Android Studio makes it easy to develop apps for any Android device. Android Studio's new view and module support makes it easier to manage app projects and resources.

1.3 XML:

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. It is defined by the W3C's XML 1.0 Specification and by several other related specifications, all of which are free open standards. Android Studio offers an advanced layout editor that allows to drag-and-drop widgets into layout and preview layout while editing the XML. Within the layout editor, one can switch between the Text view, where edit the XML file as text, and the Design view.

1.4 Location Based Services:

The ability to sense location is one aspect where a smart phone separates itself from a conventional personal computer in terms of benefits and utility. A smart phone can be easily carried by the user unlike a personal computer, a smart phone can be used anywhere for accessing the location based applications such as Google maps for looking up driving directions. The importance and usefulness of location sensing has already been well recognized and accepted with the fame of GPS based navigation systems. Most of the today's smart phones have built-in location sensing capabilities. Since most people rely on smart phones when onthe-go and far from home or office, apps that influence location based services can add real value to the user.

The location of the user is determined by Android's network location provider using GPS, cell tower signals and Wi-Fi signals. The Location Manager System services take care of the access to location-based services. The Location Manager can be accessed by using the getsystemservice () method. The current location of the user can be fetched in following ways.

1. GPS (Global Positioning System)
2. Network Service Location

Any of the above providers can be used to fetch the current location of the user (or) user's device. In this project both GPS and Network Service location are used to fetch the current location of the user. The reason for using both is, it may not be possible to obtain the location using GPS provider indoors and location using network location providers when the network connectivity is poor and also the actual location of the user cannot be determined by any of these ways. So the better way is using by comparing the both ways.

1.4.1 GPS:

The GPS uses a star grouping of satellites and ground stations to compute position and time almost anywhere on earth. A GPS receiver calculates its position by using a satellite ranging technique. To find the current location of the user the smartphones must have an inbuilt GPS receiver. In Android the statement LocationManager.GPS_PROVIDER determines the location using satellites.

1.4.2 Network Service Location:

Android Network location provider determines user location based on availability of cell tower and Wi-Fi access points. In Android the access to network provider is requested using LocationManager.NETWORK_PROVIDER.

2. Narrative:

2.1 Project Objectives:

The main objective of the proposed system is to enhance the way in which people use reminders. This is achieved by helping the people to create both time and location based reminders. In this application, user can set reminder under some category list in which different location is already specified. By filling the desired categorized item the user can set alarms for that desired categorized location. The application also allows the people to create reminder for unknown locations.

In summary, this application helps people organize their tasks effectively by taking the burden of remembering things and reminding them at a specific location or in a time interval at a location.

2.2 Project Scope:

Technology has improved a lot over the last few decades. One of the best and biggest technological advancements is the invention of smart phone. A smart phone is a device which offers more advanced computing and connectivity than regular mobile phones. In the last one or two years the smart phone users have rapidly increased and the count is still on. People using smart phones demand for better applications and updates for existing ones, which in turn creates a huge scope of Android mobile application development. Android is a fully open-source platform which was created completely for smart phones and similar devices like tablets. This project is developed for Android based smart phones. User can only avail the services of the **Remind Me** app when his/her smart phone's GPS is switched on or when they are connected to the internet. The app should keep an eye on the GPS information collected by user's smart phone to trigger the alert when the user approaches a destination. According to data collected during a 7day period ending on February 2, 2015 by Google there are 90.4% of Android users who use Android versions 4.0 to 5.0. **Remind Me** app is compatible with different versions of Android, starting from SDK version of Android 4.0 (Ice Cream Sandwich) to the recent update Android 5.0 (Lollipop).

2.3 Functionalities of the Project:

This project presents **Remind Me**, a mobile app targeted for Google's Android platform. The following are the functionalities provided by this app.

1. A reminder system that allows users to create location based reminders and be alerted when they are approaching the location. Based on the distance selected, the user will be reminded when he gets into the distance range.

2. It also has a feature to select a range of dates in which he/she should be alerted when approaching a location.

3. Ability to create time based reminders and repeating reminders.

4. It also has additional feature like ability to list most frequent tasks.

3. System Design:

The schematic diagram of the system **Remind Me** application is shown in Figure 3.1. Most smart phones are built-in with both a GPS receiver and a Wi-Fi network interface card, which can get radio signals from GPS satellites and Wi-Fi Access Points (APs), individually. Taking into account the GPS readings and the data from the Wi-Fi APs, the application can perform to estimate the current location of the user. The database is designed to store location based tasks of the user, which are stored in different tables. If a location-based task exists in the database, then the application will compare the current sensed location with the location associated with the task. At the point when the user is physically near to the predefined location, the reminder then will be activated to remind the client of his/her task. If a time-based task exists in the database, then the application will compare the current date and time in the system with the date and time associated with the task. When both the date and times are matched, a reminder is triggered to remind the user of his/her task.

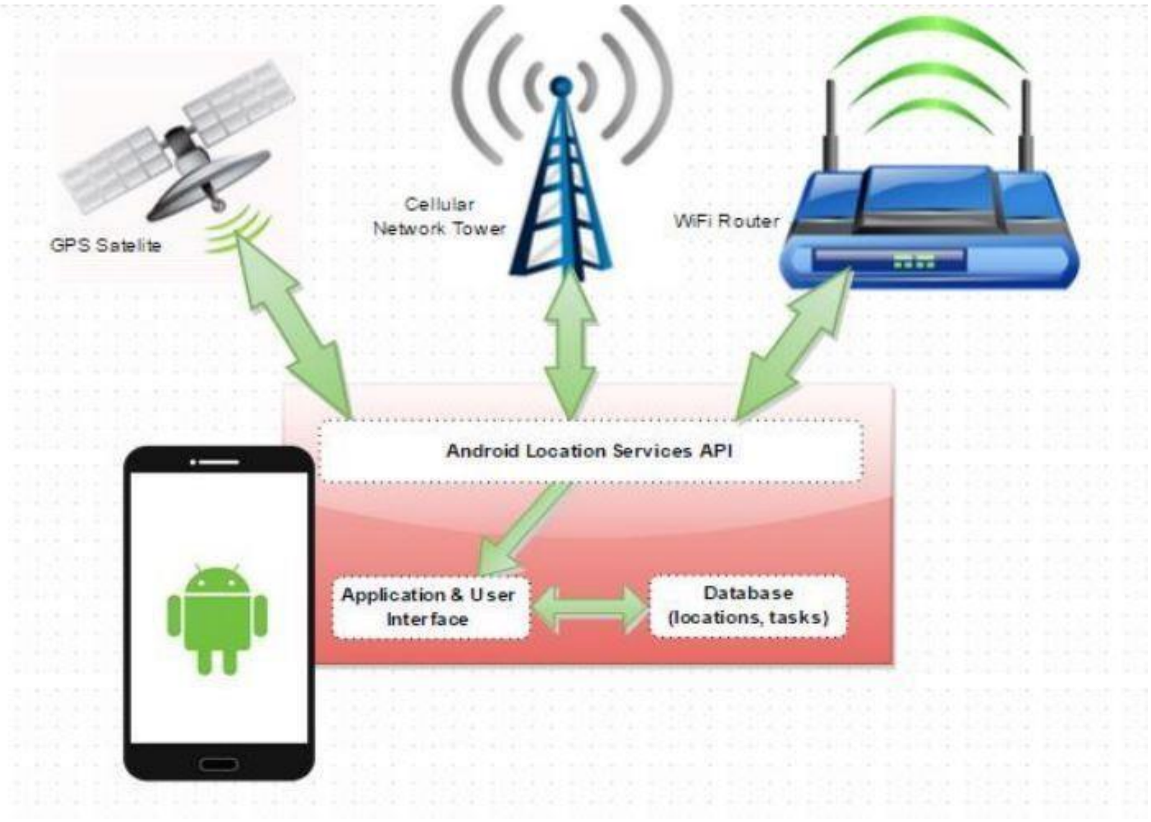

GPS Satelite
Cellular Network Tower
WiFi Router
Android Location Services API
Application & User Interface
Database (locations, tasks)

Figure. 3.1 Schematic diagram of the **Remind Me** application.

4. Flowchart:

Flowchart for developing **Remind Me** app is given below:

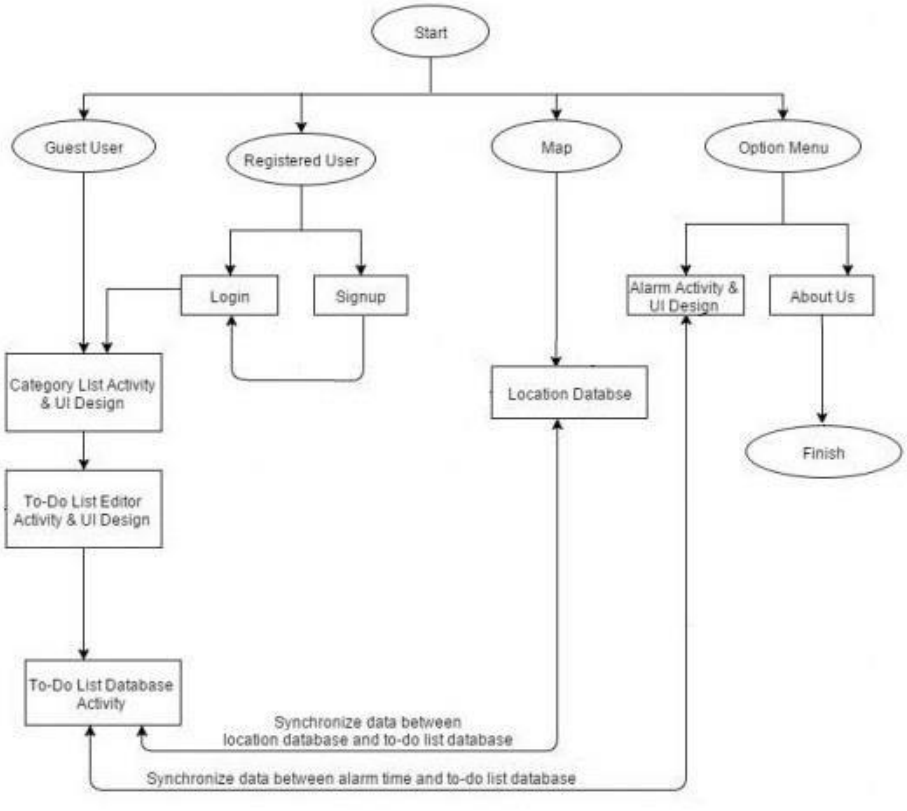

Start
Guest User
Registered User
Map
Option Menu
Login
Signup
Alarm Activity & UI Design
About Us
Category List Activity & UI Design
Location Databse
Finish
To-Do List Editor Activity & UI Design
To-Do List Database Activity
Synchronize data between location database and to-do list database
Synchronize data between alarm time and to-do list database

Fig 4.1: **Remind Me**

5. System Description:

Remind Me is based for two types of users.

I. Guest User
II. Registered User

Guest user can use the app for dynamic. Such as, guest can only set reminder and be notified when notification arrives. On the Other hand, for registered user there is a server for saving his/her to-do list and can be updated from anywhere by logged into his/her account.

There is a category list afterwards. In this category list different types of item category is placed under which user can add his/her to-do list and location. In option menu user can reset the whole category list. User can select the time and alarm type under this window. In category list window, there also a pop up menu where category item can be edited. In every category item, there is a window dialogue where added items are showed.

6. User Interface:

The main activities in the application are User Activity, Login Activity, Category List Activity, Item Editor Activity and User Location Activity. All these modules implementation is discussed in detail in this chapter.

Fig 6.1 shows the home screen of the **Remind Me** app. From the home screen the user can interact with User Activity screen where user can select the user type.

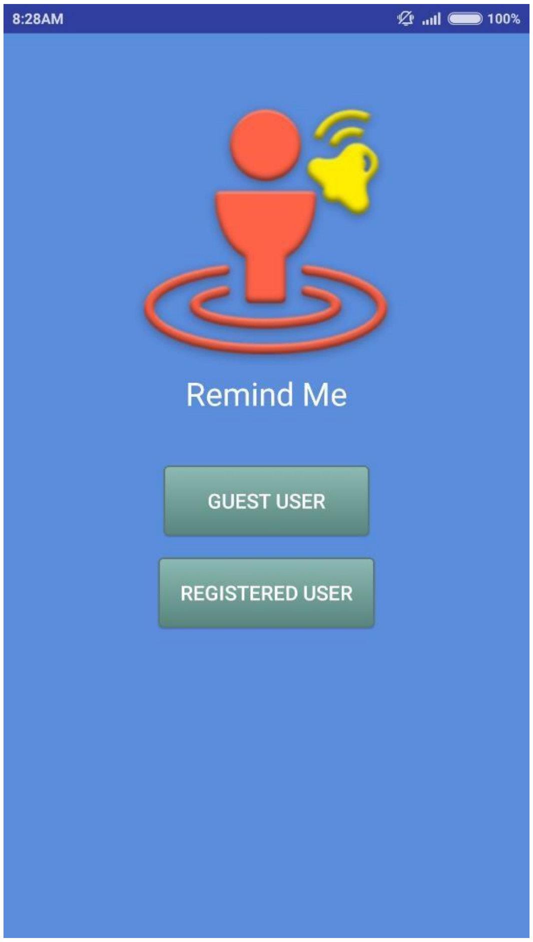


Fig 6.1: User Activity

Fig 6.2 shows the log in window for the registered user where user id and password will be provided. If user is not registered, he/she can sign up by clicking Signup button. Otherwise by clicking Login button, user can enter into the app.

If the user is not registered yet, then he/she can go back and use **Remind Me** as a guest user.

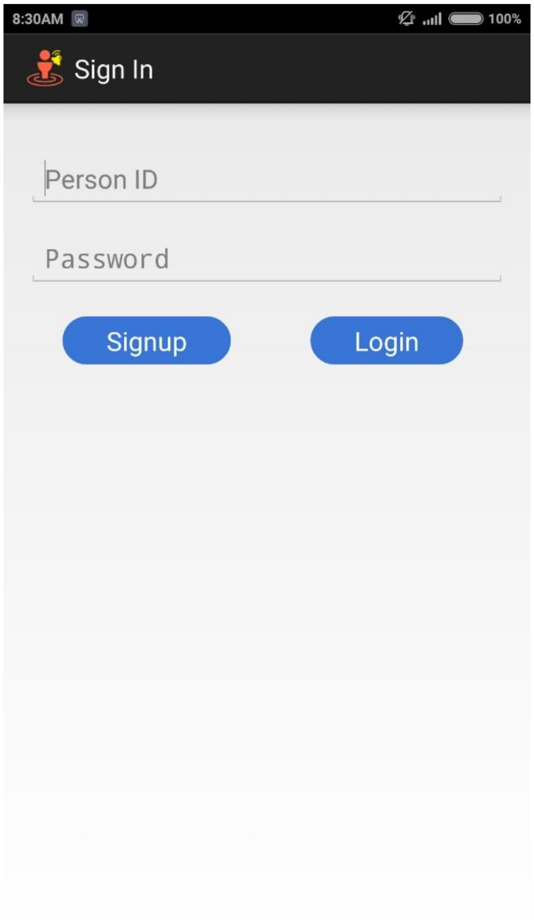


Fig 6.2: Login Activity

Fig 6.3 shows that the category list of different places from where user can buy or do his/her desired work. In this category activity, there is 7 types of categorized place. In each of this place, one must need to go to do daily work. User can make choice in which places he/she wants to go and set the place location with time table and **Remind Me** will remind him/her about his/her work by time basis.

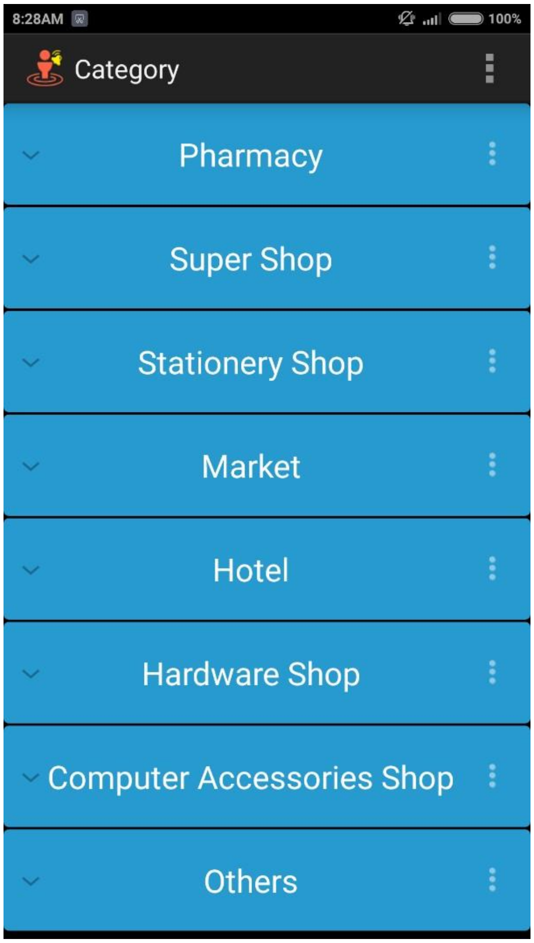


Fig 6.3: Category Activity

Fig 6.4 shows the item add menu by category basis. In every category menu, there is a three dot option menu in where item can be added and removed simultaneously. For adding item, edit menu should be pressed and the additional item add menu will be popped up. Here is a blank space to insert the item name. After clicking add item button, the item will be saved in the additional category table.

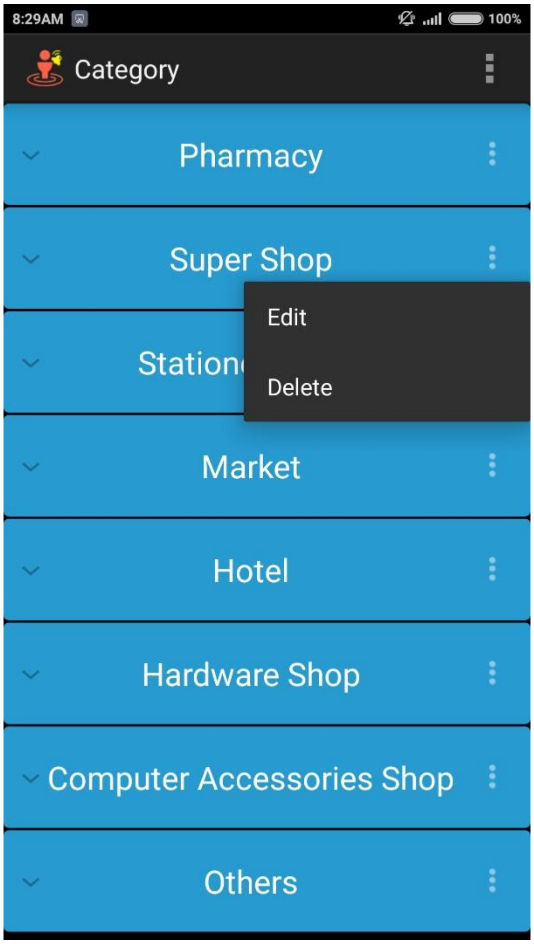


Fig 6.4: Category Edit menu

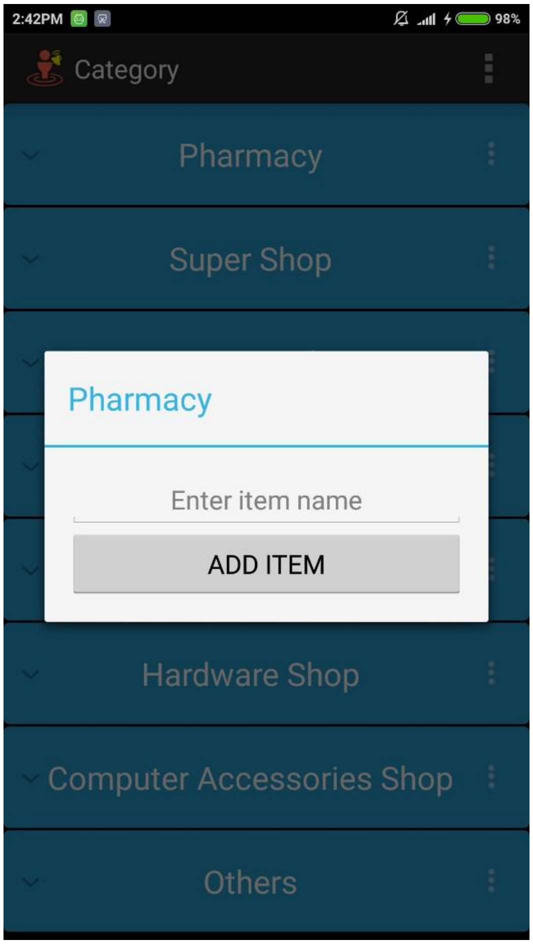


Fig 6.4: Item add menu

One can add as many item he/she wants to add in a category list. In fig 6.5 it is shown that, under “Pharmacy” category list, there are several item lists.

If user wants to remove the listed item, he/she can easily delete the list by clicking delete option. Otherwise for single item delete he/she can long press on the desired list item and easily can delete it.

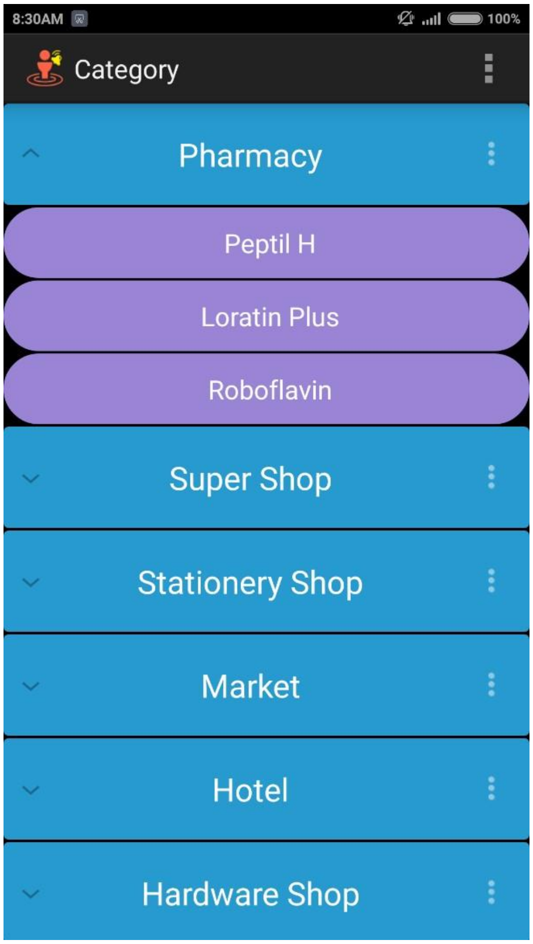


Fig 6.5: Window Dialogue

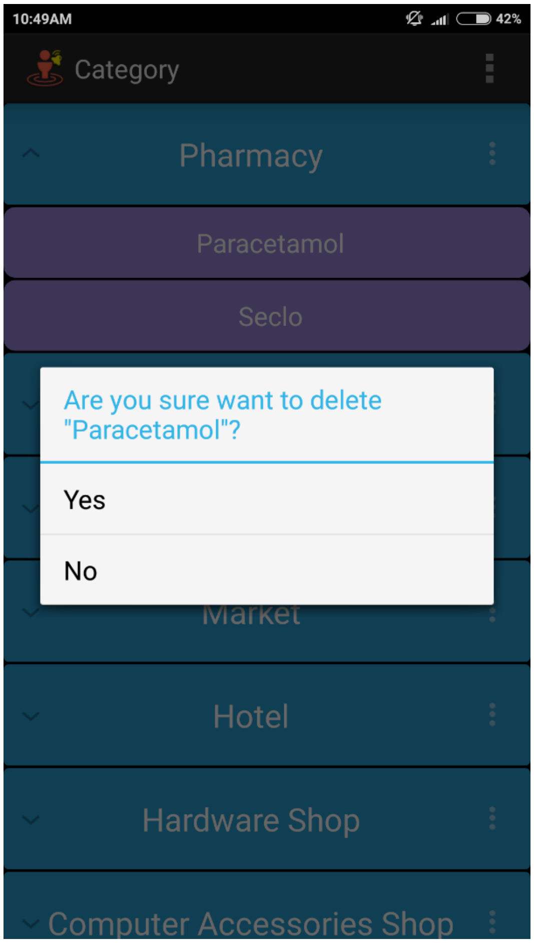

Are you sure want to delete "Paracetamol"?

Fig 6.6: Item Delete

In category option menu, user can select his/her alarm time settings. User can set alarm in schedule based as well as non-schedule based. User can select the alarm tone from Set alarm tone settings. User can set both from system alarm and custom alarm tone.

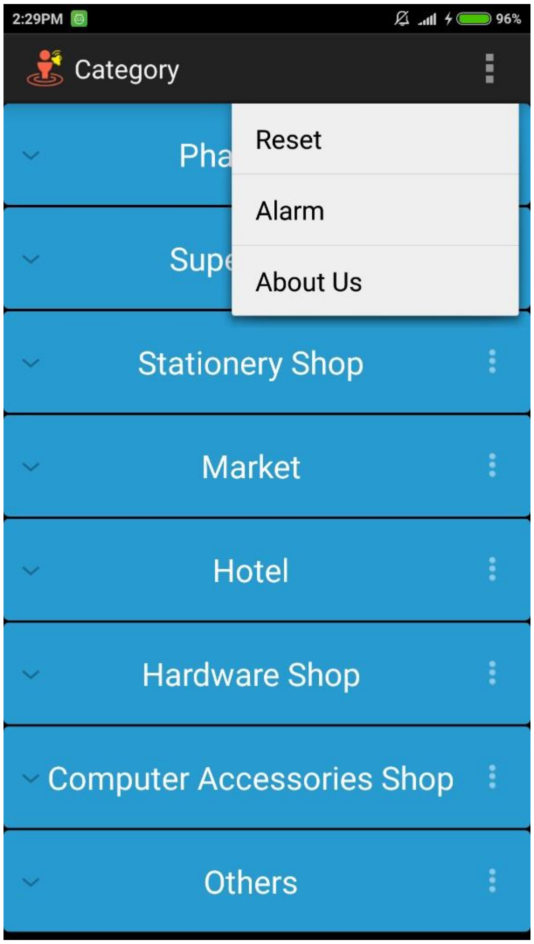
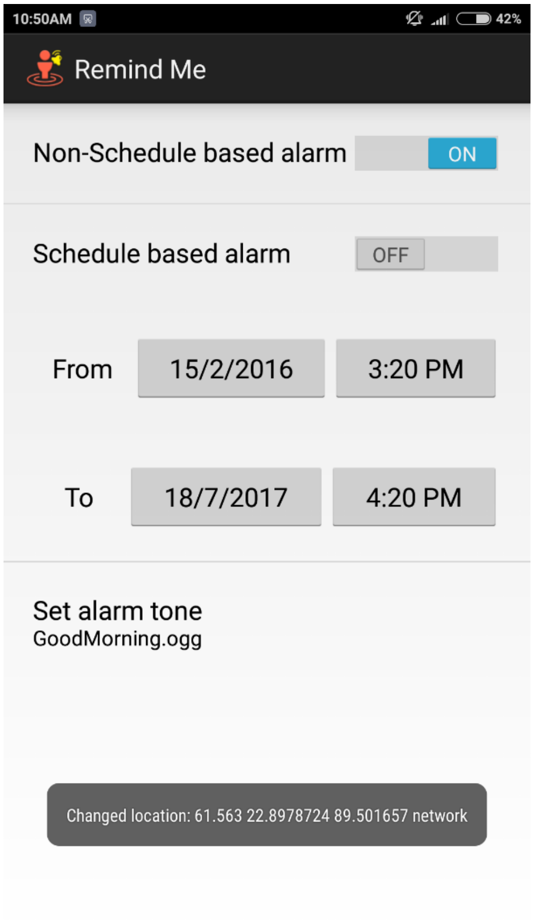


Fig 6.7: Category Option Menu



Non-Schedule based alarm ON

Schedule based alarm OFF

From 15/2/2016 3:20 PM

To 18/7/2017 4:20 PM

Set alarm tone
GoodMorning.ogg

Changed location: 61.563 22.8978724 89.501657 network

Fig 6.8: Alarm Settings

7. Sample of Code:

Method for taking the better location:

```
private Location getBetterLocation(Location gpsLocation, Location
networkLocation) {

    // GPS location is always better than Network Location
    Location betterLocation = null;
if (networkLocation == null){
betterLocation = gpsLocation;
    }
    // Check whether the GPS location fix is newer or older
long timeDelta = gpsLocation.getTime() -
networkLocation.getTime();
    int accuracyDelta = (int) gpsLocation.getAccuracy() - (int)
networkLocation.getAccuracy();

    // If it's been more than two minutes since the network
location, use the gps location     // because the user has
likely moved     if (timeDelta > 60000 || accuracyDelta <
0){          betterLocation = gpsLocation;
    }

    // If it's been less than two minutes since the network
location, use the network location     // and it must be
worse
    else if (timeDelta < 60000 || accuracyDelta > 0){
betterLocation = networkLocation;
    }
    else if (timeDelta > 0 && accuracyDelta < 0){
betterLocation = gpsLocation;
    }
    return betterLocation;
}
```

Method for preparing To Date in AlarmActivity:

```
private void prepareToDate(){
    FromToDateDatabase fromToDateDatabase = new
FromToDateDatabase(getApplicationContext());
    int[][] dataToDate = fromToDateDatabase.readToDateTable();
int[] dataToDateTemp = new int[3];     for (int i = 0; i <
dataToDate.length; i++){          for (int j = 0; j <
dataToDate[i].length; j++){             dataToDateTemp[j] =
dataToDate[i][j];
        }
}
    toDate = dataToDateTemp[0] + "/" + (dataToDateTemp[1] + 1) +
"/" + dataToDateTemp[2] + "";
toDateButton.setText(toDate);
}
```

Method for preparing To Time in AlarmActivity:

```
private void prepareToTime(){
    FromToDateDatabase fromToDateDatabase = new
FromToDateDatabase(getApplicationContext());
    int[][] dataToTime = fromToDateDatabase.readToTimeTable();
    int[] dataToTimeTemp = new int[2];
String[] dataToTimeAmPm =
fromToDateDatabase.readToTimeAmPmTable();
String temp = new String();
    for (int i = 0; i < dataToTime.length; i++){
for (int j = 0; j < dataToTime[i].length; j++){
dataToTimeTemp[j] = dataToTime[i][j];
        }
}
    for (int i = 0; i < dataToTimeAmPm.length; i++){
        // nothing
        temp = dataToTimeAmPm[i];
    }      if
(temp.equals("PM")){
dataToTimeTemp[0] -= 12;
    }

    String hourString = (dataToTimeTemp[0] ==
0)?"12":dataToTimeTemp[0] + "";
    String minuteString = (dataToTimeTemp[1] < 10)?"0" +
dataToTimeTemp[1]:dataToTimeTemp[1] + "";
    toTime = hourString + ":" + minuteString + " " + temp + "";
toTimeButton.setText(toTime);
}
```

Method for preparing Alarm Tone in AlarmActivity:

```
private void prepareAlarmTone(){
    AlarmStateDatabase alarmStateDatabase = new
AlarmStateDatabase(getApplicationContext());
String[] alarmToneUri = new String[]{};
    alarmToneUri = alarmStateDatabase.readAlarmToneTable();
String alarmToneNameTemp = new String();     for (int i =
0; i < alarmToneUri.length; i++){         alarmToneNameTemp
= alarmToneUri[i];
    }
    if (alarmToneUri.length > 0) {
        Uri uri = Uri.parse(alarmToneNameTemp);
        Ringtone ringtone =
RingtoneManager.getRingtone(getApplicationContext(), uri);
String alarmToneName =
ringtone.getTitle(getApplicationContext());
        setAlarmToneTextView.setText(Html.fromHtml("<medium>" +
"Set alarm tone" + "</medium>" + "<br />"                 +
"<small>" + alarmToneName + "</small>"));
    } else{
        setAlarmToneTextView.setText(Html.fromHtml("<medium>" +
"Set alarm tone" + "</medium>"));
    }

}
```

Method for measuring the distance:

```
/* Calculate the distance between two locations in Miles unit */
private double distance(double userLatitude, double userLongitude,
double givenLatitude, double givenLongitude){
    double distance = 0;
    double earthRadius = 3958.75; // In miles

    double dLatitude = Math.toRadians(userLatitude -
givenLatitude);
    double dLongitude = Math.toRadians(userLongitude -
givenLongitude);

    double sinOfdLatitude = Math.sin(dLatitude / 2);
    double sinOfdLongitude = Math.sin(dLongitude / 2);

    double a = Math.pow(sinOfdLatitude, 2) +
Math.pow(sinOfdLongitude, 2)
            * Math.cos(Math.toDegrees(givenLatitude)) *
Math.cos(Math.toRadians(userLatitude));

    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
    distance = earthRadius * c;
    return distance; // output distance in Miles unit
}
```

Class for playing ringtone service:

```
public class RingtonePlayingService extends Service {
    private static Ringtone ringtone;

    public RingtonePlayingService(){

    }

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int
startId) {
        super.onStartCommand(intent, flags, startId);
        Uri ringtoneUri =
Uri.parse(intent.getExtras().getString("ringtone_uri"));
        ringtone = RingtoneManager.getRingtone(getBaseContext(),
ringtoneUri);
        if (!ringtone.isPlaying()){
            ringtone.play();
        }
        return START_STICKY;
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        if (ringtone.isPlaying()){
            ringtone.stop();
        } else {
            ringtone.stop();
        }

    }
}
```

8. Future Work:

Remind Me can be improved in the future by adding the following functionalities:

- User data can be stored in cloud.
- Ability to integrate with calendar events.
- Ability to share reminders with friends using the same application.
- Ability to navigate to destination location by taking user voice commands.
- Ability to display a list of suggested locations to the user based on users voice.
- Ability to add normal time based scheduling. ▪ Make supported for iOS/windows.

9. Conclusion:

This project presents a new application for android based smart phones and tablets that allows the users to create reminders based on location. **Remind Me** allows users to create location reminders and be alerted when they enter or exit the locality of a given location. This is a project by which tension of work and difficulty of reminding them will decrease. This app also help users to use his/her valuable times best use.

10. References:

http://logomakr.com/

https://romannurik.github.io/AndroidAssetStudio/icons-launcher.html - foreground.type=image&foreground.space.trim=1&foreground.space.pad=0&foreColor=607d8b%2C0&crop=0&backgroundShape=square&backColor=607d8b%2C100&effects=none

http://stackoverflow.com/questions/22441561/expandablelistview-onclick-change-activity

http://www.androidhive.info/2013/09/android-sqlite-database-with-multiple-tables/

http://www.java2s.com/Code/Android/Database/Createtableinsertrecorddeleterecordsquerytableremovetable.htm

http://www.mikeplate.com/2010/01/21/show-a-context-menu-for-long-clicks-in-an-androidlistview/

http://stackoverflow.com/questions/920306/sending-data-back-to-the-main-activity-in-android

http://stackoverflow.com/questions/8892916/how-to-pass-a-string-array-value-from-oneactivity-to-another-activity-in-androi http://stackoverflow.com/questions/5683840/how-to-delete-all-items-from-sqlite-in-android

http://stackoverflow.com/questions/28561551/how-can-i-set-longclick-on-group-inexpandablelistview-android

http://stackoverflow.com/questions/17207366/creating-a-menu-after-a-long-click-event-on-alist-view

http://stackoverflow.com/questions/19069256/getting-expandablelistviews-selected-itemfrom-contextmenu?rq=1

http://stackoverflow.com/questions/9409429/intent-ondestroy-always-being-called-with-backbutton?lq=1 http://stackoverflow.com/questions/6054562/how-to-make-the-corners-of-a-button-round

http://stackoverflow.com/questions/30684613/android-studio-xml-editor-autocomplete-notworking-with-support-libraries

http://stackoverflow.com/questions/34599035/if-cursor-result-set-is-empty-returnsomething/34599241 - 34599241 http://stackoverflow.com/questions/2115758/how-do-i-display-an-alert-dialog-on-android http://stackoverflow.com/questions/6805756/is-it-possible-to-display-icons-in-a-popupmenu http://stackoverflow.com/questions/32491960/android-check-permission-for-locationmanager

http://stackoverflow.com/questions/20809401/contextmenu-for-children-inexpandablelistadapter

http://stackoverflow.com/questions/3145089/what-is-the-simplest-and-most-robust-way-toget-the-users-current-location-in-a?rq=1
http://stackoverflow.com/questions/15454995/popupmenu-with-icons

http://androidexample.com/Time_Picker_Basics_Part_1__Android_Example/index.php?view=article_discription&aid=85 http://www.mysamplecode.com/2013/04/android-switch-button-example.html

http://stackoverflow.com/questions/7573630/how-to-pass-data-from-context-menu-to-alertdialog-in-android?rq=1http://stackoverflow.com/questions/6404275/android-save-androidneturi-object-to-database http://stackoverflow.com/questions/6094315/single-textview-with-two-different-colored-text http://stackoverflow.com/questions/26878707/playing-ringtones-from-a-string-sqlite-database

http://stackoverflow.com/questions/22077795/android-default-tone-picker-issue-default-tonewith-notifications-and-alarm?rq=1

http://stackoverflow.com/questions/7671637/how-to-set-ringtone-with-ringtonemanageraction-ringtone-picker

http://stackoverflow.com/questions/25781177/android-ringtone-picker-list-with-defaultoption-selected

http://stackoverflow.com/questions/11650405/how-to-get-uri-current-default-ringtone-forincoming-sms http://stackoverflow.com/questions/18959685/sqlite-database-updating-a-row-android http://stackoverflow.com/questions/2659954/timepickerdialog-and-am-or-pm