Location based Alarm using Mobile Device

Fatema Rashid Al Jhawari,
Student, Muscat College,
P.O.Box:2910,
PC 112, Ruwi,
Muscat, Sultanate of Oman,

T.Sheeba
Assistant Professor, Muscat College,
P.O.Box:2910,
PC 112, Ruwi,
Muscat, Sultanate of Oman,

ABSTRACT

Location based services are essential components of many location-based applications. Location based alarm represent an emerging location-aware, just-in-time web services, which can propagate information of location to the right users at the right time and the right place. It reminds the user about the location when the user enters some predefined location of interest in the future. All the user needs to have is the mobile phone with android platform, and then the user can select the destination and find the destination on the Google map. The user can even choose from preselected major locations or recently selected locations.

The main objective of the project is to develop a GPS (Global Positioning System) based application to handle the following requirements: To alert the users through an alarm when the user reaches near a preset location, To retrieve the users current location coordinates (Latitudes and Longitudes), To allows users to set their target location and save that target to the list, Allows user to delete and edit the alarms, To allow user to the put the reminder text along with the alarm.

The final system allow user to easily activate alarm in the mobile device. Based on the saved location on the mobile device, alarm will ring automatically and display remainder message when the user reaches the target location. This mobile alarm service will act as assistance for the frequent travelers to visit new places.

Keywords

Target Location, Alarm, Global Positioning System, Android

1. INTRODUCTION

In country like oman, people find difficult to visit interior places as there no proper hoardings to point out the location. In this situation, user could configure an alarm for the destination location and comfortably search the new locations. It gives the user an alert sound when the users get into the notification area of the device and user can find the place through the use of the mobile phone and alert when approaching the location.

This prototype creates a location based alarm service which enable the frequent travellers to initiate an alarm whenever and wherever it is needed thus improving the quality of life.The alarm can be viewed, deleted and edited by the mobile user without any contradiction in data updating. Google Play services used in the project so that the application can take advantage of the latest, Google-powered features such as Maps, Google+ together with automatic platform updates distributed as an APK through the Google Play store. This makes it faster for the users to receive updates and easier for the user to integrate the newest that Google has to offer.

This application could be helpful for the frequent long distance travelers in the country like tourist, strangers, especially marketing executives, sales executives, and representatives etc who used to travel for new locations frequently. Also, this application could be used to find the nearest place to market, hospitals, restaurant, hotel or other show with the map and track.

2. LITERATURE REVIEW

There exist a number of location based systems that can automatically generate alarm by using mobile phone. Examples of such systems are described in the following links [1]:

Wake App: Wake App is a geo-located alarm that goes off when the user is about to arrive at the selected destination. Once the user select stop on the map, the map closes and the application goes into the background periodically user location using the device’s GPS sensor. When the user bus enters within a pre-configured radius from the stop the user is supposed to get off (default is 1000 meters), the alarm plays a ringtone of user choice. This application is used for train or coach travellers. Pros: Straightforward and simple to use, configure custom radius around destination within which the alarm will go off, automatically increase alarm volume if set low and Cons: Can’t save location for later use.

Bus Snooze: Bus Snooze is a GPS Location based alarm clock which will wake the user up when the user arrive at the desired location. The application allows the user to set location based alarm or time based alarm or both. The combined alarm will sound whichever arrives first – time or destination. The location tracking is done using GPS as well as the users network provider’s location. Pros: Uses both GPS and network location, configure custom radius around destination within which the alarm will go off, save location for later use and Cons: Can only save two location in free version.

WakeMe: WakeMe is another location based alarm. It lets user to choose any location on the map and set up customizable alarms for each location and save them to be used at another time when the user ride the same route. At any time user can open the application to see how much kilometers or miles the user are away from desired stop. Optionally, it can warn the user if the device loses GPS or other location signal. Pros: Uses both GPS and network location, save multiple location for later use, use different alarm configuration for different location, speech alarm on supported phones. Cons: Cannot customize radius around destination within which the alarm will go off, very buggy with occasional crashes.

Bus stop alarm [2]: Bus stop alarm will let users to set a location based alarm on their phone that will go off when they get close to the preselected bus stop. The idea is very innovative in that it is the first android application to incorporate publicly available bus information and GPS in an innovative bus stop alarm system. This application allows users to take advantage of the publicly available bus data and presents it to the user as an intuitive bus stop alarm application. Pros: This application is novel because it really emphasizes the advantage of using public transportation, a user who is taking the bus to an unfamiliar place can go with confidence that they will get off at the correct place even if they don't know exactly when the stop is approaching, a user might use the application to do work or read on the bus, another user might use this application so that they can take a power nap on the bus after a long day at work or school. Cons: No list of major/favourite locations in the application.

Mobile Location Alarm [3]: Mobile Location Alarm allows user to enter new alarm for a particular location and ring the alarm along with the remainder text when the user is near to the location. It allows user to edit, delete, update, enable and disable the alarms. In addition, user can see the locations on map to find out how far he is from the expected location.

Location Based Alarm [4]: The system is developed as 5 modules to handle the following parts.

Display module: In this module the, the Google map is displayed the locations using the GPS and GPRS/3G networks available in android Smart phones. By using this module the user can set their destination and current location based on their needs of travel. And the point to point distance and traffic condition are displayed in this map.

GPS interaction module: In this module the GPS interaction (i.e. the location update is changed based on their user’s time limit. And check whether the GPS and the internet provider is enabled or disabled. Based on that, the alert will shows to the android notification bar in android device.

Place management module: In this module the location details are stored in SQLite data storage within the android device. i.e. The visited location details are get from the location updates and stored in SQLite data storage for the user future reference.

Ring tone module: Thee default five ringtones are stored within the application and also the ring tone chooser based on their user need from their audio gallery. And also it has the volume control and vibrates mode control settings in this application module.

Alarm module: This is main module of this project; in this the alarm service and location updates are done using the android background services. The location updates is done by using the GPS and Internet providers and the alarm is set using the android device alarm services.

3. METHODOLOGY

The methodology adopted for the design and implementation of the project includes: Design the project requirements, Selection of appropriate technology and Implementation of modules (set alarm, generate alarm, delete and edit alarm).

3.1 Design the Project Requirements

3.1.1 System Requirements

3.1.1.1 Hardware Requirements

System	: Pentium IV 2.4 GHz.
Hard Disk	: 40 GB.
Floppy Drive	: 1.44 Mb.
Monitor	: 15 VGA Color.
Mouse	: Logitech.
RAM	: 1 GB.

3.1.1.2 Software Requirements

Coding Language	: Java 1.6
Tool Kit	: Android 2.2
IDE	: Eclipse 3.6.2 (Helios) or greater
Back End	: SQL

3.1.1.3 Operating Systems

Windows XP (32-bit)

Vista (32- or 64-bit)

Windows 7 (32- or 64-bit)

3.1.2 Overall System Architecture

Figure 1 shows the overall architecture of location based alarm system. It shows the actions that a user could reasonably expect to be able to perform from the mobile device user software. Home screen provides four options: Saved Alarms, Map, Activate alarm and Deactivate Alarm. “Saved Alarms” displays the alarm name along with its longitude and latitude values. This screen provides options to “Delete” and “Edit” alarms. Delete and edit as it name implies used to delete and edit the alarm name and message. Map button is used to display the map where the user can select the target location which retrieves latitude and longitude values from location provider (GPS/ Network) and prompt user to enter the alarm name and message. These details are then saved in the database. Activate Alarms is used to activate the alarm which enables the alarm and displays the provider name (GPS/Network) in the screen. Deactivate Alarms are used to disable the alarms.

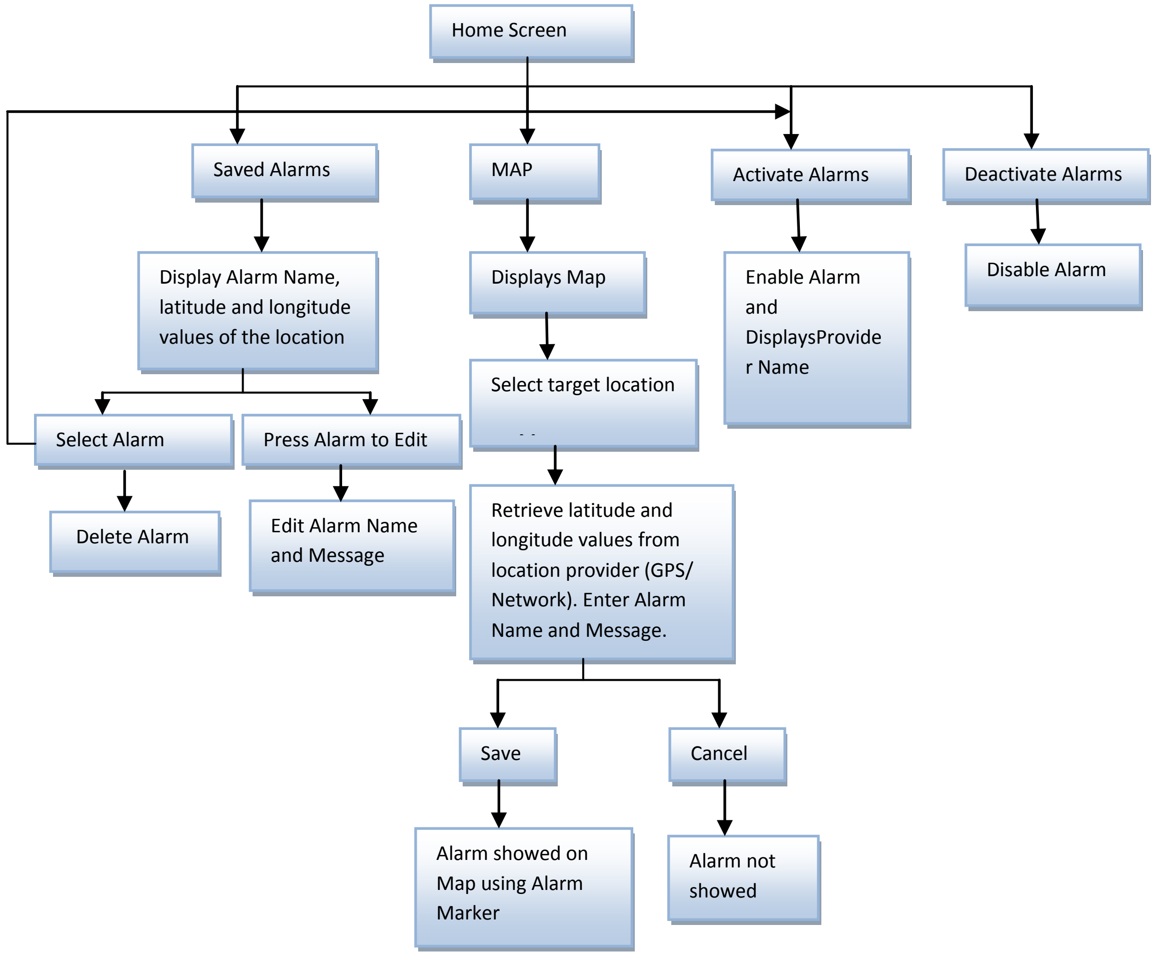

Home Screen
Saved Alarms
MAP
Activate Alarms
Deactivate Alarms
Display Alarm Name, latitude and longitude values of the location
Displays Map
Enable Alarm and DisplaysProvider Name
Disable Alarm
Select target location
Select Alarm
Press Alarm to Edit
Delete Alarm
Edit Alarm Name and Message
Retrieve latitude and longitude values from location provider (GPS/ Network). Enter Alarm Name and Message.
Save
Cancel
Alarm showed on Map using Alarm Marker
Alarm not showed

Figure 1. Overall Architecture

3.2 Selection of Appropriate Technology

The system has used Android Google API, Android Development Tool plugin, Eclipse, Sun JDK to develop the application. Android platforms give a world-class platform for creating applications for Android users. It also gives tools for creating applications that look great and take advantage of the hardware capabilities available on each device.

3.2.1 Android

Android [4] is a Linux-based operating system designed primarily for touch screen mobile devices such as smart phones and tablet computers. It is an Open Handset Alliance's (OHA) [5] mobile operating system. This application platform is very similar to Java SE. Android uses Apache Harmony's class library where only a few of the original Java SE packages have been removed. These have been replaced by GUI packages that are more suited for the reduced screen sizes used by mobile devices.

The Android SDK is available for Windows, Linux and Mac OS X, free of charge. Developers can use popular Java development tools like Eclipse and Ant. Existing Java SE based code can also be ported to Android with relative ease, as long as it does not interface with any of the packages that have been removed.

With the Android platform recently becoming very popular, this application will reach a lot of users who are using vehicle for transportation. The user interactive design is simple and intuitive so that most users can easily use it for the first time. The combination of GSM mobile and satellite-based GPS in one innovative unit gives users the ability to initiate an alarm calls whenever and wherever they need or want to do so.

The advantages of using Android[6] is as follows:

- Multitasking: Android phones can run many applications, i.e., the user can browse facebook while listening to the song.
- Ease of Notification: Any SMS, email, or missed call there will always be a notification on the home screen android phone, so the user will not miss a single SMS, Email or even Misscall.
- Easy access to thousands of applications via the Google Android Android App Market: Thousands of applications and games are ready to be downloaded on Android phones.
- Can install a modified ROM: There are many custom ROM that can be used in mobile phones Android.
- Widget: With the widgets on the home screen, the user can easily access a variety of settings quickly and easily.
- Google Maniac: Android phone has integrated with Google services, so the user can quickly check e-mail from Gmail.

3.2.2 SQLite

Android provides several ways to store user and application data. SQLite is one way of storing user data. SQLite is a very light weight database which comes with Android OS. The android.database and android.database.sqlite packages offer a higher-performance alternative where source compatibility is not an issue.

3.2.3 XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.XML is used for the creation of UI layouts in Android. Android provides a straightforward XML vocabulary that corresponds to the View classes and subclasses, such as those for widgets and layouts.

The advantage to declaring UI in XML is that it enables the user to separate the presentation of the application from the code that controls its behavior. UI descriptions are external to the application code, which means that the user can modify or adapt it without having to modify the source code and recompile. For example, XML layouts can be created for different screen orientations, different device screen sizes, and different languages. Additionally, declaring the layout in XML makes it easier to visualize the structure of the UI, so it's easier to debug problems [8] [9].

3.2.4 GPS

Global Positioning System (GPS) [5] is a satellite based, medium earth orbit (MEO), navigation technology. GPS relies on a constellation of at least 24 satellites to provide location, speed and direction information to its users. It works by using a technique called trilateration combined with atomic clocks in the satellites in order to accurately determine the correct location.GPS finds the user position by calculating differences in the times the signals, from different satellites, take to reach the receiver. GPS signals are decoded, so the smart phone must have in-built GPS receiver.

The accuracy of GPS is relatively high compared to most other techniques, but it requires line of sight to satellites which severely limits its use indoors. In big cities with lots of high buildings and narrow streets GPS will often have very low accuracy because the number of satellites it can see is limited.

The Android emulator allows a file with pre-recorded track points to be installed so that it can emulate a real GPS and make applications believes that it is actually moving. Using this technique, a track was recorded with a real GPS and uploaded to the emulator. To be able to find out if a user is inside or outside the area one must first find a suitable method to define this area. When a GPS is used the area is defined as a circle with radius r and center coordinate (xc; yc). A disadvantage of GPS is that the users will most likely fail if the user wants to specify the center of the circular area in the middle of the user area.

3.3 Implementation

My prototype creates a location based alarm service which enable the thus frequent travellers to initiate an alarm whenever and wherever it is needed improving the quality of life. The alarm can be viewed, deleted and edited by the mobile user without any contradiction in data updating. Google Play services [10] used in the project so that the application can take advantage of the latest, Google-powered features such as Maps, Google+, and more, with automatic platform updates distributed as an APK through the Google Play store. This makes it faster for the users to receive updates and easier for the user to integrate the newest that Google has to offer. Below figure 2 shows the application “Map Alarm” in the mobile phone.

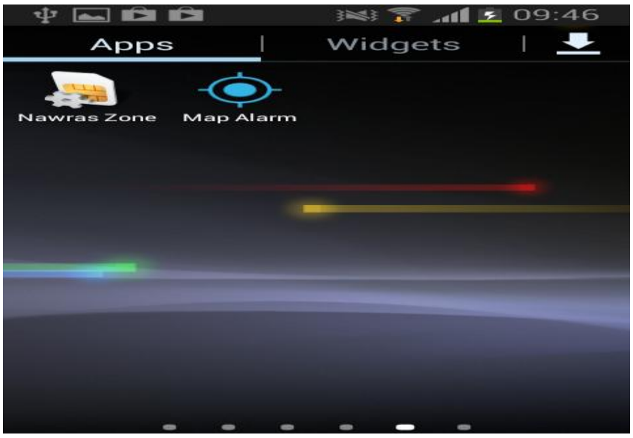


Figure 2. “Map Alarm” Application

3.3.1 Modules Developed

The system is developed as three modules which are described in the following sections:

3.3.1.1 Set Alarm

To set the alarm, user has to click the button “Map” on the home screen as shown in Fig 3. An intent is started to start the activity specified in the class “LocationMapActivity.class” by calling the method StartActivity().

Intent is an abstract description of an operation to be performed. It can be used with, startActivity to launch an Activity, broadcastIntent to send it to any interested BroadcastReceiver components, and startService(Intent) or bindService(Intent, ServiceConnection, int) to communicate with a background Service.

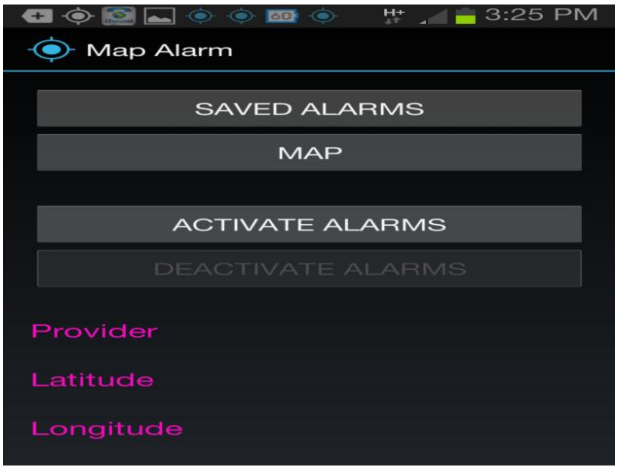


Figure 3. Home Screen

To display the Map: Map (Figure 4) is used to show the current location of the user. The Google Maps Android API allows the user to display a Google map in the Android application. Two main classes “GoogleMap” and “MapFragment” are used to represent maps in the API. The key class used is the “GoogleMap”. MapFragment, a subclass of the Android Fragment class, allows the user to place a map in an Android Fragment. MapFragment objects act as containers for the map, and provide access to the GoogleMap object.

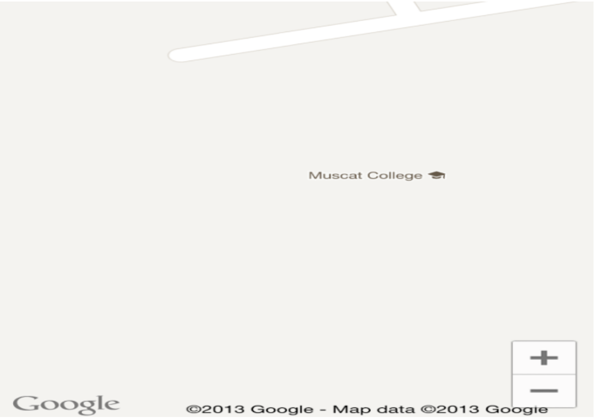


Figure 4.Map Display

To add Map Code: To work with the map inside the code, first the layout files are set as content view for the activity. For example, the layout file has the name activity_location_map.xml and SetContentView() is used to set the activity content from a layout resource.

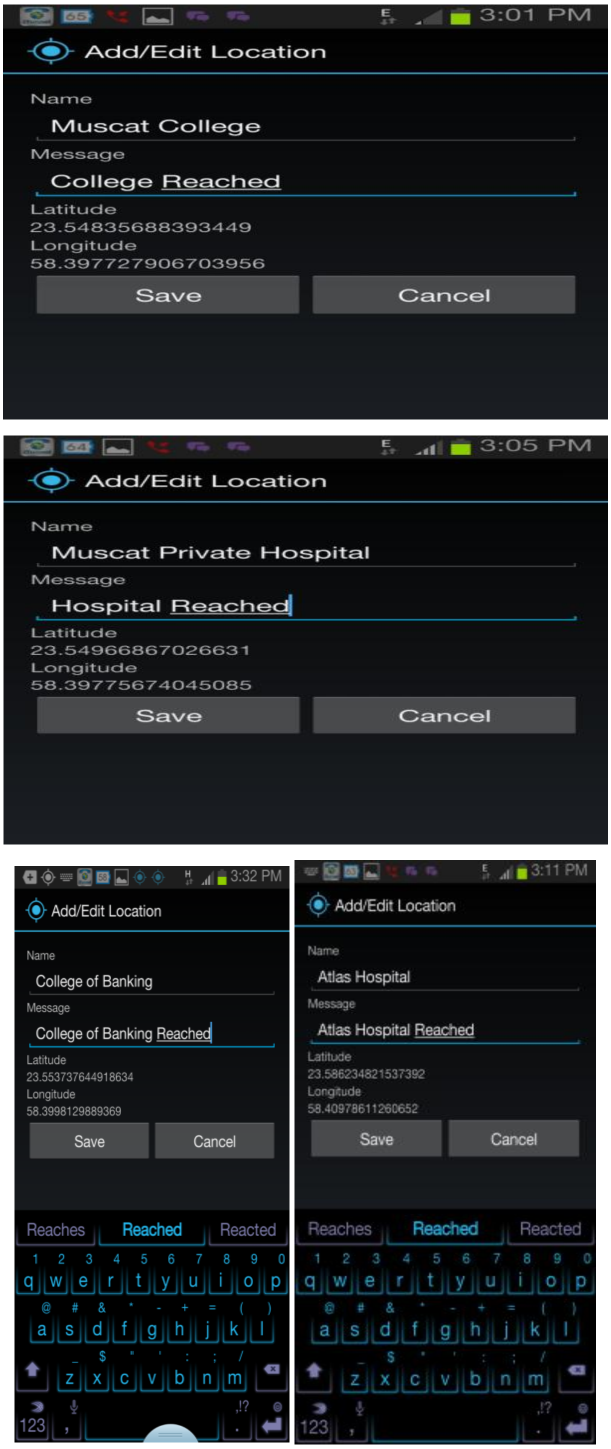
To obtain handle to the Map: Initially a reference to the class LocationManager is obtained by calling the getSystemService() method. Then a null check of GoogleMap is done to confirm that the map is not already instantiated. Map is obtained by getting a handle to the map by calling the method getFragmentManager().findFragmentById(), by passing resource ID as the parameter and the return value is passed to the GoogleMapvariable. Before interacting with the GoogleMap object, it is necessary to confirm that the object can be instantiated, and the Google Play Services components are correctly installed on the target device. To do that, MapFragement.getMap() method return value is checked. If the return value is not null, the next step is to set up the map.

In the next step, user has to select the target location on the map. For example in the below Fig 5, Muscat college, Muscat Private Hospital, College of Banking and Finance and Atlas Hospital are selected as target location.

Next a location provider is selected which may differ in their performance characteristics such as time-to-fix, accuracy, monetary cost, power consumption, and so on. Two types of location providers: GPS and Network based location providers are enabled in this project by calling the method isProviderEnabled().GPS provider has greater accuracy but requires a longer fix time and it available in outer area whereas Network based location providers are less accurate compared to GPS and it is available inside the building. GPRS is accessed when the user is outside the building and Network is accessed inside the building. To check whether these location providers are enabled, method isProviderEnabled() is used which return either true or false. True value indicates that the specified location provider is enabled.

After enabling one of the location providers, next step is to update the location periodically by the named provider using the method requestLocationUpdates().In this method time (100ms) and distance (10 meters) are set as the parameter which indicates that change in location of every 10 meters and passing of 100ms, an update is send to the application. In case the provider is disabled by the user, updates will stop, and a provider non availability update will be sent. As soon as the provider is enabled again, location updates will immediately resume and a provider availability update is sent.

The location update retrieves the latitude and longitude value of the selected target current location in a separate window. In the same window options are provided to enter the alarm name and message. The user then enters the alarm name and the message as shown in the below Figure 5. The entire alarm details are then added to the database by calling an intent which enables the updated location to store in the database.



3:01 PM
Add/Edit Location
Name
Muscat College
Message
College Reached
Latitude
23.54835688393449
Longitude
58.397727906703956

3:05 PM
Add/Edit Location
Name
Muscat Private Hospital
Message
Hospital Reached
Latitude
23.54966867026631
Longitude
58.39775674045085

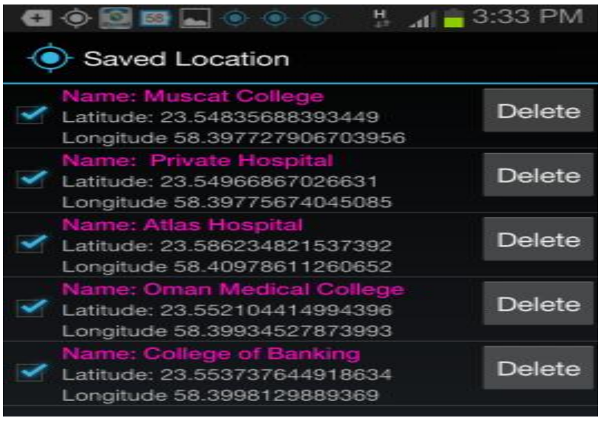
3:32 PM
Add/Edit Location
Name
College of Banking
Message
College of Banking Reached
Latitude
23.553737644918634
Longitude
58.3998129889369

3:11 PM
Add/Edit Location
Name
Atlas Hospital
Message
Atlas Hospital Reached
Latitude
23.586234821537392
Longitude
58.40978611260652

Figure 5. Retrieved latitude and longitude values of target location

Once the alarm is set, alarms are automatically saved allowing user to use it on future rides as shown in Fig 6. Description of the locations is saved on the mobile phone using android SQLite database. The saved alarm details are displayed (Figure

6) by clicking the button “Saved Alarms” in the home screen. The tick mark shown in the left side of the below figure shows that the alarms are selected for ringing.



3:33 PM

Saved Location

✓ Name: Muscat College
Latitude: 23.54835688393449
Longitude 58.397727906703956
Delete

✓ Name: Private Hospital
Latitude: 23.54966867026631
Longitude 58.39775674045085
Delete

✓ Name: Atlas Hospital
Latitude: 23.586234821537392
Longitude 58.40978611260652
Delete

✓ Name: Oman Medical College
Latitude: 23.552104414994396
Longitude 58.39934527873993
Delete

✓ Name: College of Banking
Latitude: 23.553737644918634
Longitude 58.3998129889369
Delete

Figure 6. Saved Alarms

To Create Alarm Marker: Markers (Figure 7) indicate single locations on the map. Markers can be customized by changing the default color, or replacing the marker icon with a custom image.

Markers identify locations on the map. Markers use a standard icon, common to the Google Maps look and feel. It's possible to change the icon's color, image or anchor point via the API. Markers are objects of type Marker, and are added to the map with the GoogleMap.addMarker(markerOptions) method. Marker is added by getting the latitude, longitude and alarm name using the position and title properties.

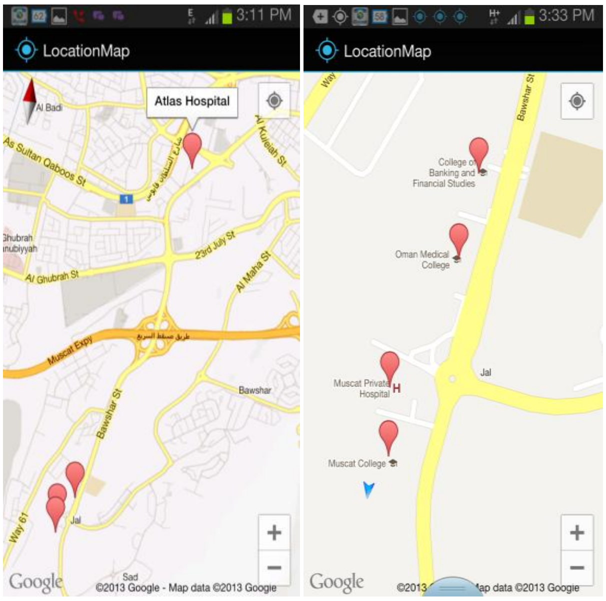


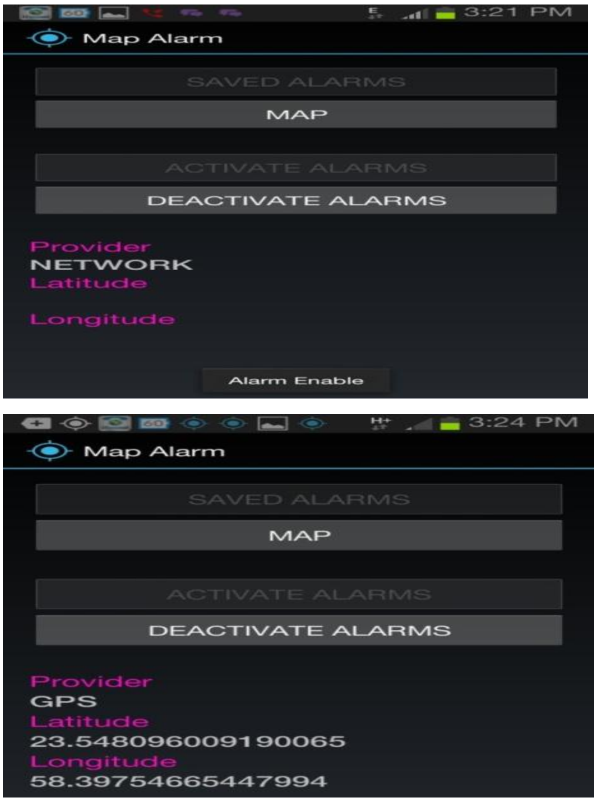
Figure 7 . Alarm Marker

Notification of Location Change: Method lastknownlocation() is used to obtain the location from the given provider. To receive notification from the LocationManager when the location has changed LocationListener class is used. These methods are called if the LocationListener has been registered with the location manager service using the requestLocationUpdates(String, long, float, LocationListener) method. The retrieved location is then passed as parameter to the method onLocationChanged() to notify that the location has been changed.

3.3.1.2 Generate Alarm

Generate the alarm is done by ringing the alarm if user is near to the location for which the alarm is set. This is directly calculated with the user’s geographic area.

Activate Alarm: To activate the alarm first the button “Activate Alarms” in the home screen is pressed which shows the name of the location provider (Network and GPS) as shown in the below figure 8. First a location provider Network and GPS based location providers are enabled by calling the method isProviderEnabled(). After enabling one of the location providers, next step is to update the location periodically every 100msand 10 meters by the named provider using the method requestLocationUpdates().



Provider
NETWORK
Latitude

Longitude

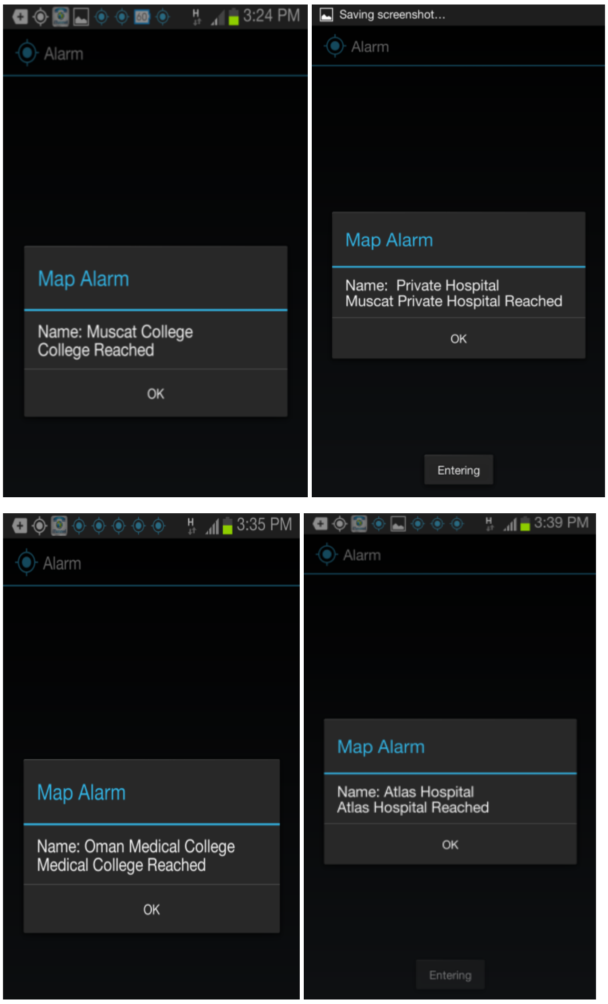
Provider
GPS
Latitude
23.548096009190065
Longitude
58.39754665447994

Figure 8. Activate Alarms

Next a new intent for broadcast is created with a custom action “proximityalert". Proximity key value is set to LocationManager.KEY_PROXIMITY_ENTERINGKey. It is used for the bundle extra holding a boolean indicating whether a proximity alert is entering (true) or exiting (false). If entering value is true, then alarm location id, alarm name, latitude and longitude values are retrieved and intent is started for the alert dialog by calling the class ProximityAlarmActivity.class.

This pending intent is retrieved using the method getBroadcast() if all the given parameters in the method are matched in which intent is specified as one of the parameter. If the described PendingIntent already exists, the current one is cancelled before generating a new one using the parameter FLAG_CANCEL_CURRENT. Finally, addProximityAlert method is called which set a proximity alert for the location given by the position (latitude, longitude), radius (in meters), expiration time (in milliseconds) and Pending Intent. When the device detects that it has entered or exited the area surrounding the location, the given PendingIntent will be used to create an intent to be fired. The fired Intent will have a boolean extra added with key KEY_PROXIMITY_ENTERING. If the value is true, the device is entering the proximity region; if false, it is exiting. Due to the approximate nature of position estimation, if the device passes through the given area briefly, it is possible that no Intent will be fired. Similarly, Intent could be fired if the device passes very close to the given area but does not actually enter it. After the number of milliseconds given by the expiration parameter, the location manager will delete this proximity alert and no longer monitor it. A value of -1 indicates that there should be no expiration time. Internally, this method uses both NETWORK_PROVIDER and GPS_PROVIDER.

Below Figure 9 shows the screen shots of alarm generated when the user enters the target location.



Name: Muscat College
College Reached

Name: Private Hospital
Muscat Private Hospital Reached

Name: Oman Medical College
Medical College Reached

Name: Atlas Hospital
Atlas Hospital Reached

Figure 9. Alarms

After generating the alert message, a notification is shown using getActivity().getSystemService (Context.NOTIFICATION_SERVICE). After that the notification is hided using the flag Notification. FLAG_AUTO_CANCEL.Builder class is used for creating the convenient dialog construction using method like setTitle(), setMessage(),setPositiveButton() etc.

Deactivate Alarm: To deactivate alarm, first the intent is removed by calling the method UnregisterPendingIntent(). Then by using the method FindViewById(int) the widgets are retrieved in that UI, that the user need to interact with programmatically. Widgets such as location provider, latitude, longitude and registerAlarm Button values are retrieved using this method and finally the alarm button is disabled (Figure 10)by setting the function "setEnabled" to false.

Next the pending intent has to be unregistered. To do that, initially all location updates are removed for the specified LocationListener and then the proximity alert is removed using the method removeProximityAlert() for the given PendingIntent specified as the parameter.

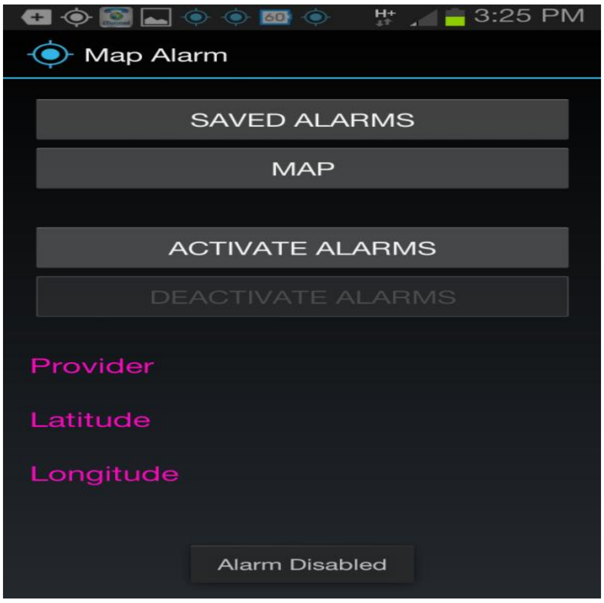
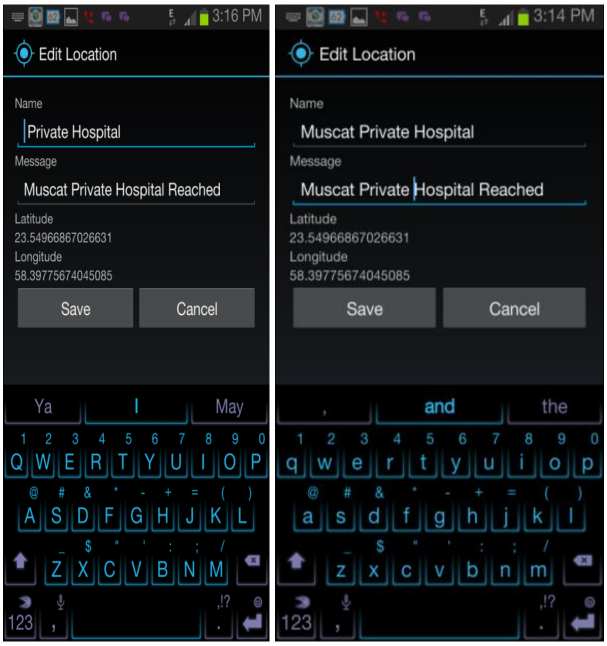


Figure 10. Disable Alarms

3.3.2 *Delete and Edit Alarms*

Alarm name and the message can be edited by pressing the option alarm in the saved alarm screen. Below Figure.11 shows an example where the alarm name"Muscat Private Hospital" is changed to “Private Hospital”.

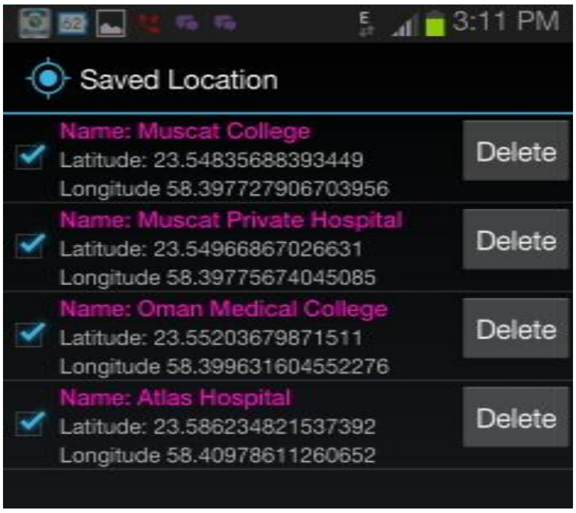


Name
Private Hospital
Message
Muscat Private Hospital Reached
Latitude
23.54966867026631
Longitude
58.39775674045085

Name
Muscat Private Hospital
Message
Muscat Private Hospital Reached
Latitude
23.54966867026631
Longitude
58.39775674045085

Figure 11. Edit Alarm

Similarly, deletion of alarm can be done using the Delete button provided in the saved alarms screen. Below Figure 12 shows an example of deletion of the alarm name “College of Banking and Financial Studies”.



☑ Name: Muscat College
Latitude: 23.54835688393449
Longitude 58.397727906703956

☑ Name: Muscat Private Hospital
Latitude: 23.54966867026631
Longitude 58.39775674045085

☑ Name: Oman Medical College
Latitude: 23.55203679871511
Longitude 58.399631604552276

☑ Name: Atlas Hospital
Latitude: 23.586234821537392
Longitude 58.40978611260652

Figure 12. Delete Alarm

3.4 Conclusions and Future Improvement

The final system allow user to easily activate alarm in the mobile device. Based on the saved location on the mobile device, alarm will ring automatically and display remainder message when the user reaches the target location. The system will also integrate additional settings into the system. This mobile alarm service will act as assistance for the frequent travellers to visit new places.

3.4.1 Future Work

3.4.1.1 Scope of Future Application

The future application of this system is to include voice message. Voice message enhances the usability of the application. Currently, system ringtone is used as the default ringtone in the application. However, choice of ring tones could be provided from the audio gallery, since it has volume control and vibrates mode control settings.

3.4.1.2 Scope of Improvement

The possibility of improvement of the system includes: improvement of the precision of the GPS system positioning, activation of alarm within a certain date and time, determining the distance from the point at which the application is to alarm us, sharing of alarm with other users (sending/receiving) etc.

3.4.1.3 Contributions to Theoretical Knowledge and Professional Practice

The final system ensures user friendly, attractive, efficient and it lessen the manual effort and it ensures data integrity for demonstration purpose. The reliability and accuracy depend primarily on the GPS system which gives us a large dose of security. The user interface design and easy installation makes the application available at any moment.

4. REFERENCES

[1] http://www.instantfundas.com/2011/11/3-location-based-alarms-that-wake-you.html, accessed February 2013.

[2] http://code.google.com/p/bus-stop-alarm/, accessed February 2013.

[3] http://www.projecttopics.info/Java/Mobile_Location_Alarm.php, accessed February 2013.

[4] http://en.wikipedia.org/wiki/Android_ (operating_system), accessed February 2013.

[5] http://brage.bibsys.no/uis/bitstram/URN:NBN:no-ibsys_brage_75971/1/Tjensvold,%20 Jan%20Magne.pdf, accessed February 2013.

[6] http://mobilecon.info/advantages-and-disadvantages-android-mobile-phone.html, accessed February 2013.

[7] http://en.wikipedia.org/wiki/JavaScript, accessed February 2013.

[8] http://en.wikipedia.org/wiki/XML, accessed February 2013.

[9] http://developer.android.com/guide/topics/ui/declaring-layout.html, accessed February 2013.

[10] http://developer.android.com/google/play-services/index.html, accessed February 2013.

[11] http://developer.android.com/reference/android/location/LocationManager.html, accessed February 2013.

[12] http://developer.android.com/training/basics/location/locationmanager.html#TaskPickLocationProvider, accessed February 2013.