

ANDROID MAPPING APPLICATION

Abdalwhab Bakheet¹ Ahmed Abd Almahmoud² and Wigdan Ahmed³

Department of Electrical and Electronic Engineering,
University of Khartoum, Khartoum, Sudan

abdalwhab.bakheet@gmail.com
eng.ahmedabdelmahmoud@gmail.com
wigdan.mohamed@gmail.com

ABSTRACT

Location-aware and mapping applications have gone from a desirable feature to an essential part of any smart phone. Whether a user is checking into a social network, looking for a pharmacy in the middle of the night, or located in somewhere and needs help, the key is always the same: location.

In this project, an Android mapping application is developed. The application is able to display the map of the whole world while online or, display a pre-downloaded map while offline, track the user's location, display a compass to determine north, send the user's location to others in case of emergency using SMS, receive and interpret received location from the message, display it on the map, and notify the user by the reception of the location.

The application was developed using agile methodology. It, met its objectives and successfully passed 91% of the final system test, recording that some limitations were discovered, the application needs further testing and can be implemented for particular company or university using their own maps or editing the maps in OSM (open street maps).

KEYWORDS

Android, mapping application, mobile development

1. INTRODUCTION

Mobile phones and their applications have become an essential part of our lives. They are not only connecting us with friends and families, but also they can now tell us where we are, where to go, what to do, and how to do it. The Internet is chock full of applications that can entertain us and make our lives easier.

Location-aware and mapping applications have gone from a desirable feature to an essential part of any smart phone. Companies, universities, airports and organizations of today are now providing maps either as part of their websites or as stand-alone applications to give direction services. Users provided with information about how to reach the specific organization, and how to navigate inside that organization.

Due to this massive need, scores of companies and developers have developed customized mapping applications and mapping APIs (application programming interface). There is no need to provide a detailed rundown of the available mapping applications and their features, because we

believe that the reader is already familiar with those applications. Instead this paper describes the development of such an application, its phases and problems encountered, in the hope that it will help anyone developing a similar application.

The original purpose to develop this application was to provide a custom mapping application for the University of Khartoum middle complex. The goal was to allow visitors, students and staff of the University to navigate in the complex using their smart-phones. However, after starting the research and development of the application, it was found that it is better to provide a general solution that will be interesting and useful for any user (not just for the University of Khartoum) to increase its market value.

2. APPLICATION FINAL SPECIFICATION

The application:

1. Displays a detailed map of the University of Khartoum, since the application was originally developed for the University of Khartoum.
2. Detects and displays the user's location on the map.
3. Enables the user to zoom the map in and out.
4. Allows the user to choose between two modes: an online mode displaying the map of the whole world and an offline mode displaying a pre-downloaded or auto-cached map that does not need an Internet connection.
5. Tracks the user location online and offline (without using the Internet).
6. Provides a compass that shows the direction towards north even in areas where signal is missing.
7. Provides a scale bar showing the relation between distances in the map and real distances.
8. Can send the user location to saved emergency numbers.
9. Receives the location sent by another user and displays it on the map.
10. Alerts the user of the reception of a message that contains location sent by another user.
11. Allows registering and edit of emergency numbers.

3. METHODOLOGY

Being the first time an Android application and maybe the first mobile application ever to be developed at the University of Khartoum it was not possible to foresee and plan for the whole application development life cycle. Therefore an agile software development methodology was adopted, tacking a small piece of requirement (only one or two new features), implementing and testing them separately from the application and then integrate them with the application, testing the application and repeating this sequence for the next feature.

Any mapping application includes two main features: displaying a map and determining the user's location.

A. Display a map

The application should display a map that is detailed enough to allow the user to find a particular place of interest.

The map is displayed from a map server. It can be displayed directly from the server or stored in the handset for offline use. Displaying the map directly from the server has the disadvantage that the loading of the map tiles will be dependent on the availability and speed of the Internet

connection i.e. in some places there is no signal, and in others there is signal but the data rate is very small.

The level of details of the map displayed by the application will depend on the quality of the map on the server. Maps qualities vary greatly from one server to another, and even for the same server the level of detail varies for different regions.

Many map servers allow the community of users to edit maps and contribute additional details, and then they review the edits and display them in the server. After that the developers can access them by the application or download them.

To display maps from a particular server, developers use the application programming interface (API) provided by the server that provides classes needed to access the maps located in the server.

B. Determine user location

To help users find a destination, or pinpoint their locations, mapping applications include the ability to determine the user's location. The required accuracy depends on the purpose of the application, but for most applications the degree of accuracy is sufficient if the application can determine the location accurate enough for the user to be able to know what street or building he is in, this degree of accuracy will be fine.

At the same time the accuracy of the location provided by the application depends on the method used to determine the location, and the accuracy and the availability of that method in the particular region where the user is.

Android supports determining the user's location using global position system (GPS) that is built in the handset, or using information from a nearby transmission tower. Moreover, if the user is connected to a Wi-Fi network, it can be used to determine the location. It must be mentioned that all these location providers are unreliable to some extent. For example, GPS signals do not reach inside buildings [1].

3.1. Application development iterations

3.1.1. Determine how to do it

Like with any project, the first phase was information gathering and analysis, trying to understand how the project can be done, what the various methods are, together with their respective merits. Any mapping application involves two parts; a client-side part and a server-side part. The client-side part provides a user interface and accesses a map server. On the server-side part developers prepare the maps in a well-defined format and provide an API to access the map located on the server, then developers can use the API to develop applications that access the maps and use additional features provided by the API.

Some servers publish their API for public access and use for free, others require an API key, and some raise a fee for the use of the API.

Therefore the first phase of the project was to study the various APIs available on the Internet, their functionality, and their merits.

There are so many mapping APIs available, each of which belonging to and accessing a particular server. Some APIs target indoor mapping, while others target outdoor mapping.

The most famous API is Google's mapping API, but there are many other mapping APIs like MapQuest, Nutiteq, ArcGIS, Decarta, Guidbee, Mapsfore, OpenLayers, and OsmDroid.

We studied the merits of each one, and even tested some of them by accessing and displaying maps using them, finally settle for outdoor mapping, because outdoor mapping is more general and more suitable for the University of Khartoum complex.

In this paper we will not discuss all APIs available, because there are many APIs, and they are always changing, new APIs are introduced and the available ones are constantly improved. The first step for anyone trying to develop a similar application is to look and examine the available APIs, their features and license agreements.

One interesting API is Nutiteq mapping API besides supporting the basic features: showing interactive online map, and map overlays, it also supports many base maps options OpenStreetMap, Bing, MapQuest, MapBox, AND, CloudMade, where Google supports only Google Maps, and aerials.

It also supports online routing, geocoding, and unlike Google's API it supports offline maps and offline routing [2].

Google's mapping API, has the advantages of wide spread, it is well documented, and easy to use. We didn't use it because for free license the application must be free and publicly available, also the number of requests per day is limited, it only access Google maps server, and we couldn't grantee that if we edit the maps on their server that they will approve it (because of the political problem with Sudan).

OsmDroid was selected because of its flexible license agreement, and its features: OsmDroid mapping API is free, and does not even require a mapping API key. It accesses Open street maps which is a collaborative project to create a free editable map of the world. For these two reasons we had chosen it. In addition OsmDroid allows developer to change the server of the maps; it can be used with other APIs to provide routing service, and several other features that we used part of in the application.

3.1.2. Adding features

The features of the application's final specification were developed one at a time, with each development iteration adding a new feature: displaying the map directly from open street maps server, determining the user's location, supporting offline mapping, editing open street maps to add more details about the University of Khartoum middle complex, tracking user location both online and offline, and arguably the most interesting feature of all: sending the user location in case of emergency. This last feature is the one we will discuss in a bit more details.

Emergency service:

A user may save up to five numbers to send his or her location to using SMS in case of emergency by two clicks. When the message is received by the recipient's mobile the application receives the message, and extracts the location information from it, then opens displaying the received location and notifies the user by the reception of the location.

Some screen-shots for the application are displayed in the results section.

4. DESIGN

As shown in (Figure 1) there is six main activities (classes that extend Android activity class which corresponds and manage one user interface).

“OurMapping” class is responsible for managing the application main window.

“OsmMap” class was written because many activities need to display map, so they all need common classes like “MapView” which displays maps, “MapController” to control the map. Also they need “MyLocationOverlay” to display a location overlay on the map, “ScaleBarOverlay” to display scale bar.

Therefore any other class that needs to display a map will just extends “OsmMap”, no need to rewrite code, and reduced maintenance overhead.

“SetNumbers” class is responsible for setting the emergency numbers; the class uses a “SharedPreferences” and “SharedPreferences.Editor” to store the numbers.

“SendMyLocation” is responsible for sending the user location; it uses “LocationManager” to get the user location, and “SharedPreferences” to get the stored telephone numbers.

Help class displays help information about the application, where “About” class displays information about the version of the application.

(Figure 2) shows the five activities that extend “OsmMap”, where each one of them is responsible for managing one user interface.

(Figure 2) also shows “SMSReceiver” class which is the only class in the application that is not an activity, it does not need a user interface, it runs in the background that is why it extends Android “BroadcastReceiver”, which allows it to listen to the reception of new messages and activate “MessageReceived” class.

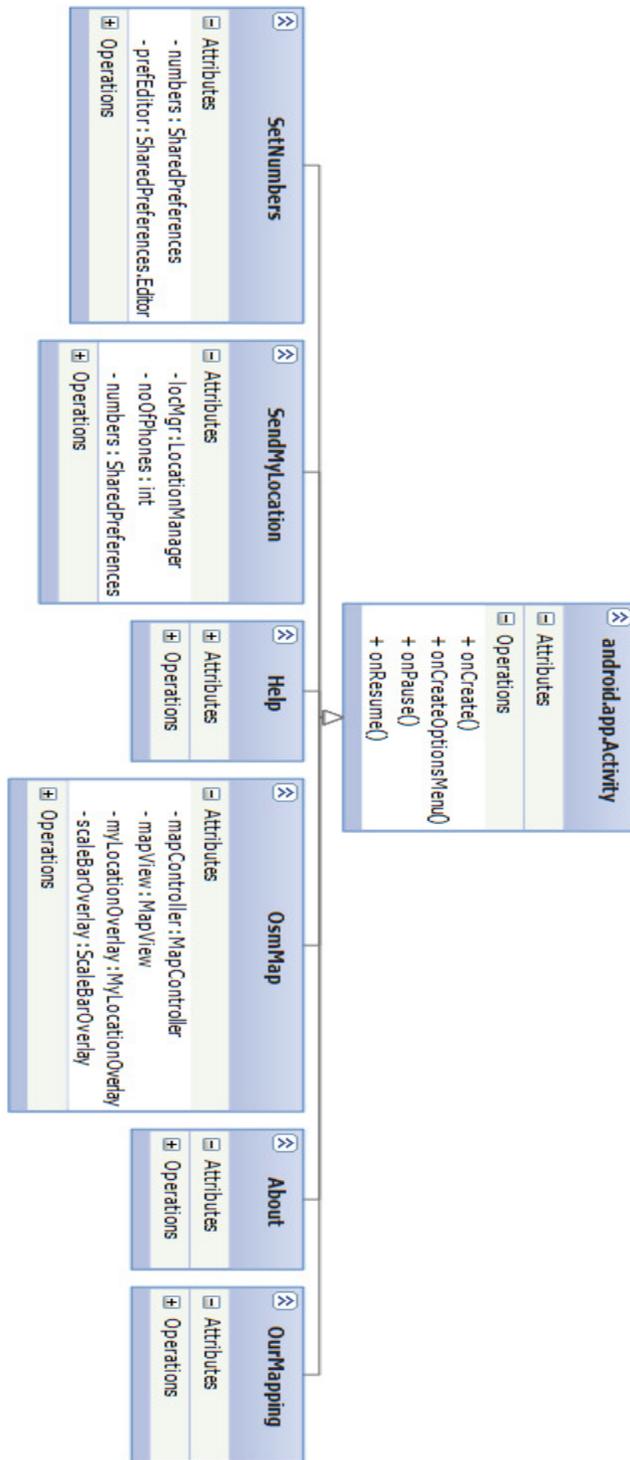


Figure 1. Program basic structure part1 using unified modeling language

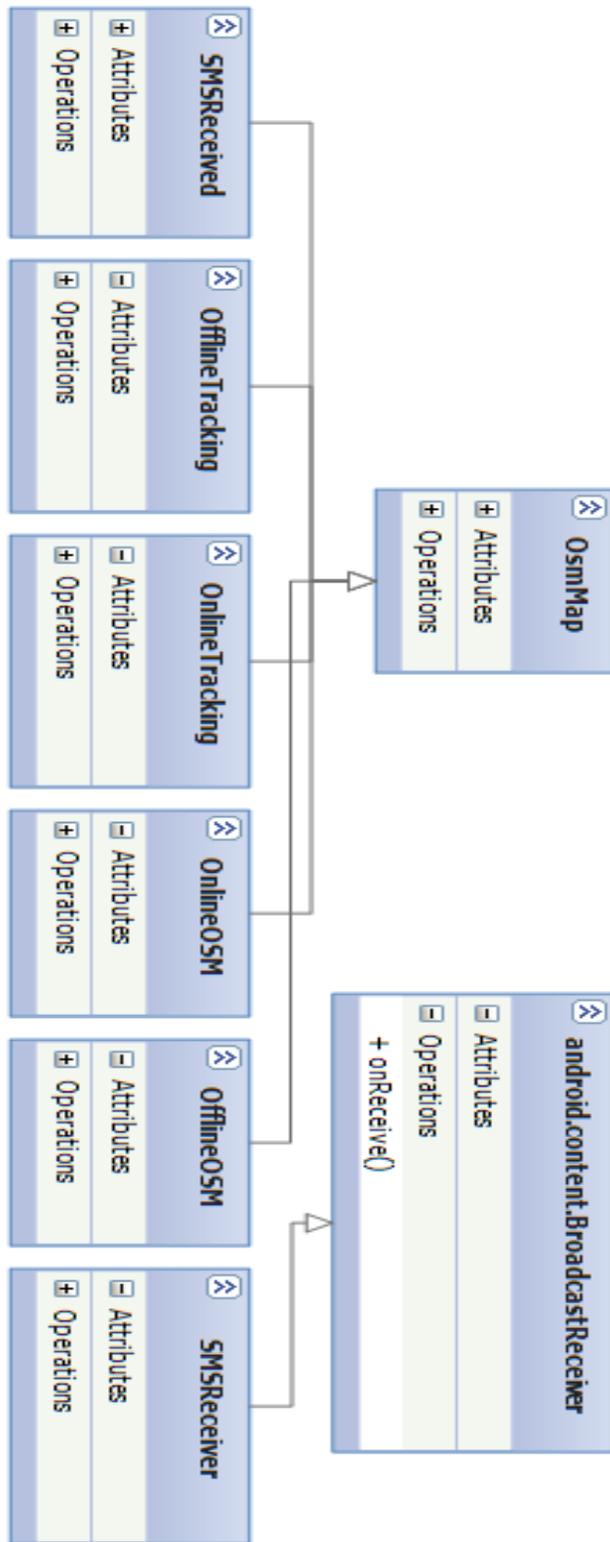


Figure 2. Program basic structure part2 using unified modeling language

5. RESULTS

Here are some screen shots for the application:

Like shown in (Figure 3) when the application starts it displays the main menu, showing four options: online, offline, help and about, allowing the user to choose any one of them.



Figure 3. Main menu

(Figure 4) shows the application online mode, where the application downloads maps from the server. The application shows a scale bar, a compass, zoom controls, and online icon is highlighted in green indicate that this mode is active, while offline and location icons are inactive. The location icon is highlighted in green if tracking the user location is enabled.

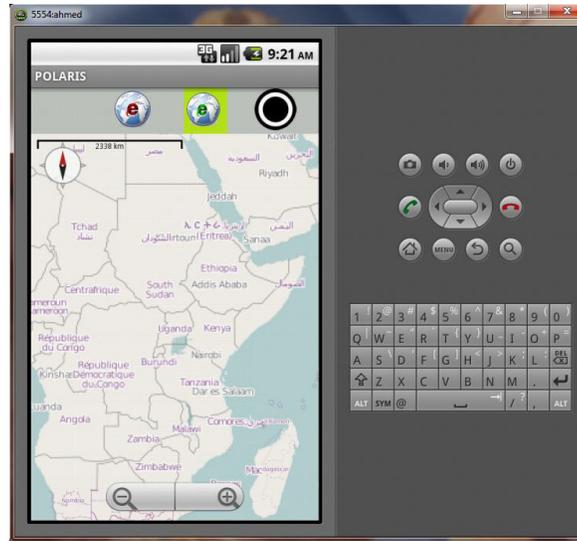


Figure 4. Application on the online mode

(Figure 5) shows the application offline mode, where maps are not loaded from the server and only the maps that were cached or pre-downloaded before are displayed, that why it is noticeable that when the map is zoomed in no details appeared, just like an image, also it is noticeable that the offline icon is highlighted in green while the online and location icons are not.

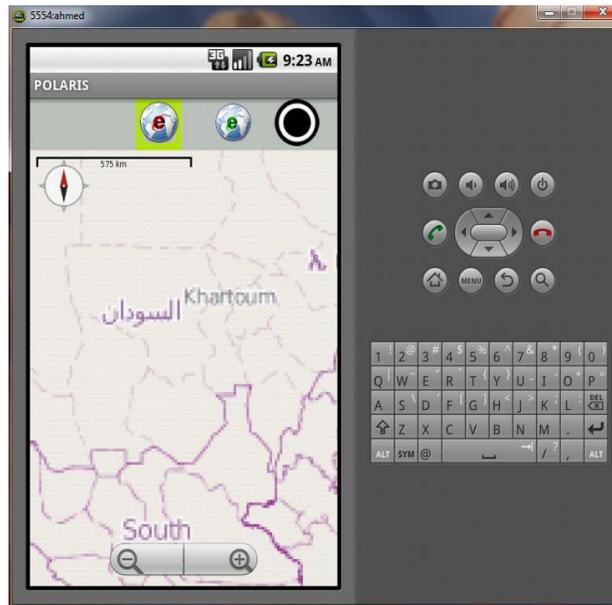


Figure 5. Application on the offline mode

(Figure 6) shows the tracking and displaying of the user's location on the map, also the location button is displayed in green which means it is active.

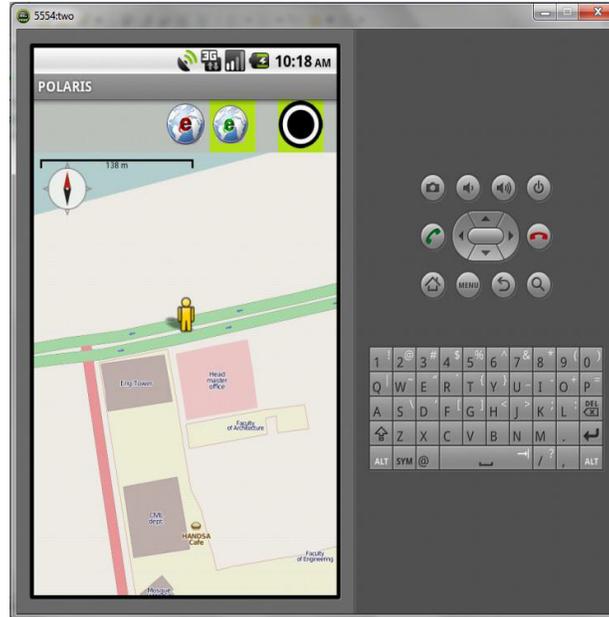


Figure 6. Tracking user location

(Figure 7) Shows the options to register the emergency numbers, to send the user's location and to return to the main menu.



Figure 7. Options to set the emergency numbers and send the user's location

(Figure 8) shows the window to register the emergency numbers, where the user can add and save up to five numbers.

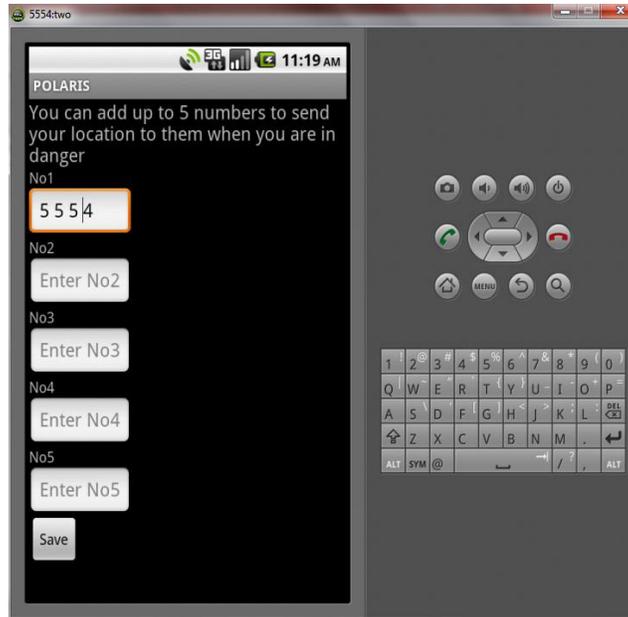


Figure 8. Window to set the emergency number

(Table 1) shows a comparison between the application main features and couple of major android application that use open street maps.

Table 1. Comparison between our application and other OSM mapping applications [3]

Name	Display map	Navigate	Make track	Price
AFTrack GPS-Tracking	Yes	Yes	Yes	3.90€;4.95\$;free
AlpineQuest GPS Hiking	Yes	No	Yes	3€
AndNav2	Yes	Yes	Yes	Free
AndRoad	Yes	Yes	Yes	Free
Ape@map	Yes	No	Yes	Free
ARnav	Yes	Yes	No	Free
Our application	Yes	Yes	No	-----

The table shows a couple examples with comparison based on main features only, because this is not the right place to do a complete, deep comparison between available applications. This paper is more focused on how similar applications can be developed. However from this comparison we can see that there are many available applications that can differ in the features they provide.

6. CONCLUSION

The application was successfully developed and achieved its requirements, but it requires more testing to increase stability, especially the user's tracking feature. The application needs to be

tested on real physical devices. All the testing of the application was done using emulators due to the lack of a real handset for testing.

The application can be implemented to any particular company or university by editing the maps at open street maps. Further research can be done to change the map server to a server specified by the customer.

ACKNOWLEDGMENT

Thanks to Allah the Almighty without His help we would not be able to get this project done. We wish to thanks Wigdan Ahmed lecturer at the University of Khartoum, faculty of engineering for being our supervisor in this project.

Thanks to everyone who helped us in the development of the project specially Alla Ahmed, Ahmed Abdallah, Habab Siddig.

Thanks to Elamin Altayeb and Abdallah Ulber for helping us revising the paper.

REFERENCES

- [1] (2012) Android developer website. [Online]. Available: <http://developer.android.com/guide/topics/location/index.html>.
- [2] (2013) Nutiteq website. [Online]. Available: <http://www.nutiteq.com/nutiteq-sdk/comparison/>
- [3] (2013) Open Street Map website. [Online] Available: <http://wiki.openstreetmap.org/wiki/Android>

Authors

1. Abdalwhab Bakheet Mohamed

B.Sc. Electrical and electronic engineering (Software Engineering major) Currently: Working as Software test engineer at Banan IT, Sudan

2. Ahmed Abd Elmahmoud Elhaj Salih.

B.Sc. Electrical and electronic engineering (Telecommunication Engineering major) Currently: Technical manager at Furgan FM, Sudan

3. Wigdan Ahmed

M.Sc. Computer Science and Entrepreneurship CCNP, CCDP, MCSE, ITIL V3 Foundation, CDRE Currently: Senior Infrastructure Engineer at EBS, Sudan