

Android Based Vehicle Tracking System

Omer Ali Abubakr Abd Elrhman, Asim Mohammed Ahmed, Tarig Hyder Mekki, Ghassan Mohammed Taha
Department of Electrical and Electronic Engineering, University of Khartoum
Khartoum, Sudan

Abstract

Android, as an operating system, has provided users with great opportunity to innovate and get things done in a mobile device. This paper presents how to use the GPS technology in Android devices to complete an interactive application which can be used to monitor a fleet of vehicles and display their positions on Google Maps. By using SMS messages, this information can be transmitted to the server. It provides a telemonitoring system for distribution or transportation vehicles owned by a specific company. The whole system is made of two key parts. The first one is the client, which represents an Android application that is installed in the vehicle. During a vehicle's motion, its location can be reported by SMS messages. The second is the server, which is a computer programme representing a map using Google Maps to show the last known locations of all tracked vehicles.

The current system is able to provide the monitoring process from anywhere. The purpose of this system is to use the Android platform to provide the following features: i) Location information (longitude, latitude). ii) Real time tracking using SMS. iii) Map View of all vehicles' locations.

This system is needed by many companies to monitor illegal and unethical use of their vehicles. It also provides assurance that the location of the vehicle is known in the case of robbery.

I. INTRODUCTION

Smart phones have become an essential part of human life. They are integrated with multiple and different features that allow us to communicate with the world, organize our lives and document events. One of the most important features is location-based services.

Smart phones use different features to get the location of the phone. One of these features is the GPS. The GPS uses satellites to get the exact location of the phone in terms of longitude and latitude. The smart phone utilizes this coordination and uses them to show the phone's location in a map application. In addition, special mobile applications have been developed with various abilities of navigating [1,2].

Tracking vehicles has always been a problem for big companies dealing with transport vehicles or cargo vehicles. A GPS tracking device is complicated, expensive and the technologies used in it are monopolized by their vendors. An open source technology with location-based services and free map API is needed to develop a tracking system for multiple vehicles all at once with low expenses.

The purpose of this project is to develop a system with a client Android application that acquires the phone location and sends this location to a server programme. The server programme receives the location of multiple

vehicles and displays them on a map, with the ability to track one vehicle or multiple vehicles on the same map.

The objective of this project can be summarized as follows: Develop a vehicle tracking system that in general has the ability to: i) get the location of the device in longitude-latitude format. ii) store the location history of all vehicles. iii) display the last known location of the vehicle. iv) display the location of multiple vehicles on one map. v) display the information of the vehicle and the time on the map. vi) track the location history of one vehicle.

II. CLIENT "ANDROID" APPLICATION

This application represents the client of the system where the location of the vehicle is acquired and sent to the server. It is basically an Android application [3].

After installing the application on the targeted smartphone running on one of the versions of Android OS, the icon appears on the screen along with the other applications' icons. Clicking the icon will open the application [4]. Figure 1 shows the main flow of the client application whereby opening the application leads to a splash screen and after two seconds the login screen is opened.

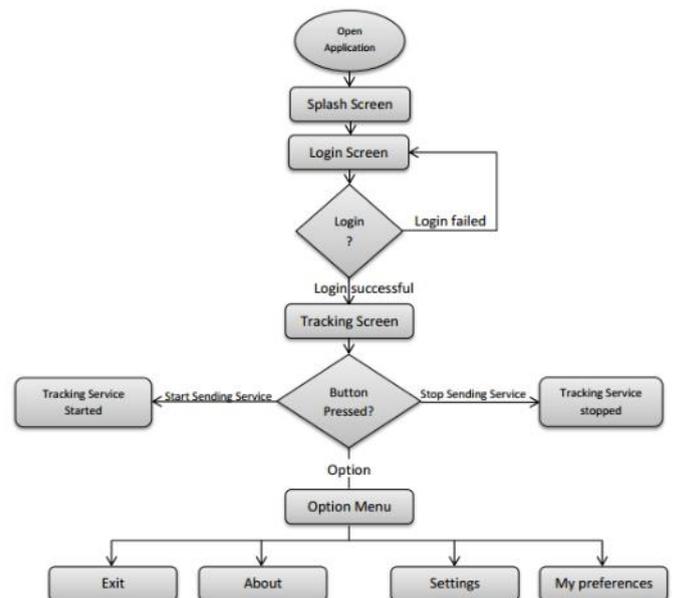


Fig 1. The main flow of the client application

GPS technology [5] is used to track the users' current location. It is the major important function of the application.

Whenever the GPS of the phone is enabled, GPS locates the current location point, and the application saves this location to be sent later using other functions. Whenever the location of the phone changes, GPS will track it and updates the current location of the phone. This service starts automatically when logged into the application unless the GPS is disabled whereby a “GPS is disabled” message is popped on the screen.

Sending the location of the phone to the server is the second function of the application. This service uses SMS messages to send the location from the application.

The application acquires the location from the location service and the message sending service creates an SMS message with the server phone number and sends it automatically. The time interval between the sending of the SMS messages is determined in the settings as well as the number of messages to notify the user. The sending service sends a message at every time interval as specified in the settings. After a number of messages are sent, the service notifies the user with the total number of messages sent. This service is started by clicking the “start sending” button in the tracking activity described earlier. When the service is started a notification pops up in the notification bar. This notification will be visible as long as the service is running.

The sending service starts automatically along with the location service if the auto boot option is selected from the settings menu. Figure 2 shows the main interface of the client application.

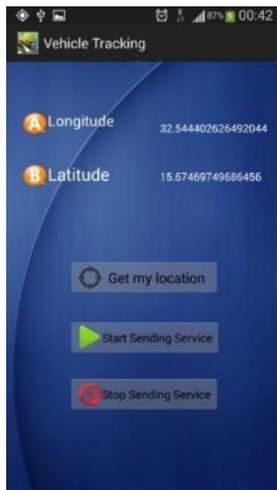


Fig 2. The main interface of the client application

III. SERVER “COMPUTER” PROGRAMME

This application represents the server of the system whereby the vehicles’ location is presented on a map. It is basically a computer programme. Figure 3 shows the main flow of the server programme whereby opening the application leads to a splash screen and after two seconds the main screen is opened.

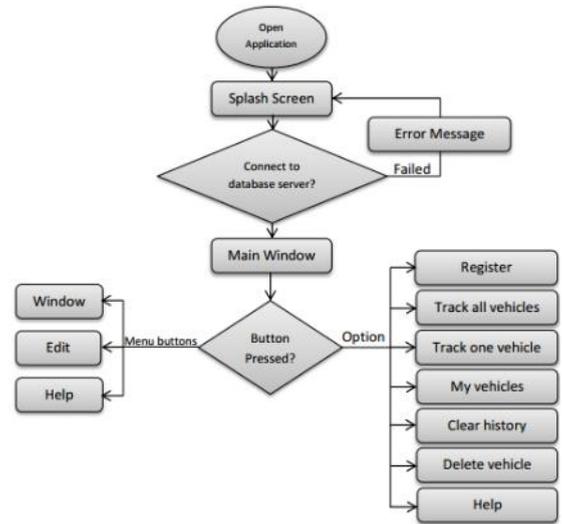


Fig 3. The main flow of the server programme

The SMS is received from the client through a modem and then saved into a database using the SMS enabler programme. The programme then collects these messages from the database and saves them into a file in a certain format. The JavaScript receives this file and uses it in showing the current location of the vehicles on the map. Figure 4 shows the flowchart in handling the SMS messages.

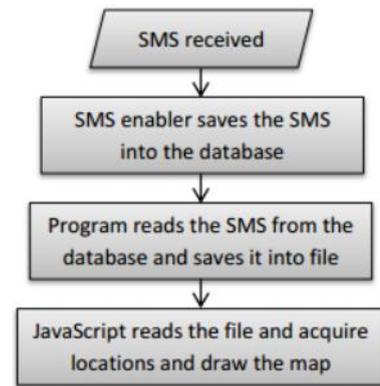


Fig 4. SMS handling

Drawing and updating the map are the functions of the Google Maps API [6]. The map is drawn using information from the file created in SMS handling. The file contains the location of the vehicles as well as the information about the registered vehicles. After the map is drawn it is updated every 10 seconds to check if new location updates is received.

The database design for the programme consists of two unrelated tables. The first table contains the messages received from the clients. The second table contains the information of the registered vehicles to be used on the map. Figure 5 shows the design of the database.

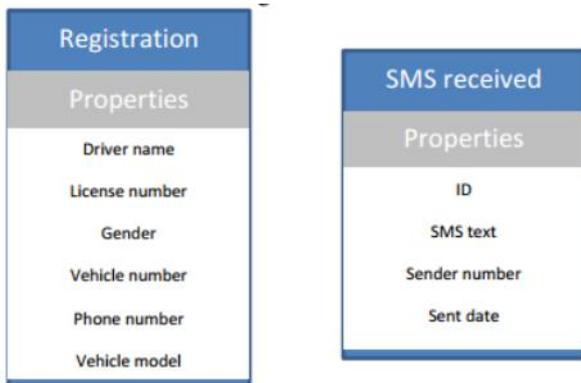


Fig 5. The programme database structure

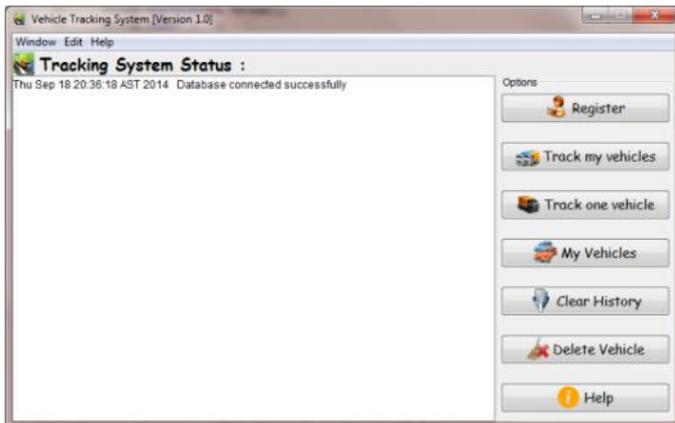


Fig 6. The main interface of the server programme

IV. IMPLEMENTATION

The client application is an Android based application where the authenticated customer interacts with a graphical interface on his smart phone to start or edit the tracking service.

The client application is developed using Java based Android programming and XML. The code for the location consists of location change alerts as well as location provider status “GPS or network”. Figure 7 shows the code for the location request in the application.

```

//***** My Location Listener *****
public class MyLocationListener implements LocationListener{
    SharedPreferences pref = getSharedPreferences("Registration",0);
    @Override
    public void onLocationChanged(Location loc){
        final Editor editor = pref.edit();
        editor.putString("Longitude", Double.toString(loc.getLongitude()));
        editor.putString("Latitude", Double.toString(loc.getLatitude()));
        editor.commit();
    }
    @Override
    public void onProviderDisabled(String provider){
        Toast.makeText( getApplicationContext(),provider + " Disabled",Toast.LENGTH_SHORT ).show();
    }
    @Override
    public void onProviderEnabled(String provider){
        Toast.makeText( getApplicationContext(),provider + " Enabled",Toast.LENGTH_SHORT).show();
    }
    @Override
    public void onStatusChanged(String provider, int status, Bundle extras){
    }
}/* End of Class MyLocationListener */
}/* End of UseGps Activity */

```

Fig 7. Location request code

The server programme is a computer programme where the system owner interacts with a graphical interface on his computer to track the vehicle on Google Map.

The server programme is developed using Java, HTML 5, JavaScript [7] and MySQL database [8]. The code for the map API view consists of the Google Maps API [9] as well as the functions to draw the vehicles' locations on the canvas. Figure 8 shows the code for the map developed in the programme.

```

var map;
function initialize(){
    map = new google.maps.Map(document.getElementById('map'), {
        zoom: 2,
        center: new google.maps.LatLng(15.6, 20),
        mapTypeId: google.maps.MapTypeId.ROADMAP,
        panControl: true,
        zoomControl: true,
        mapTypeControl: true,
        scaleControl: true,
        streetViewControl: true,
        overviewMapControl: true
    });
}

```

Fig 8. Map drawing code

The programme uses text files extracted from the database [10] to draw vehicles' locations on the map as well as each vehicle's information. Figure 9 shows a snapshot of the interface for vehicles' locations.

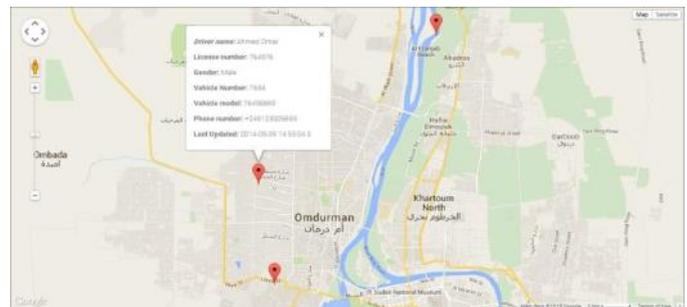


Fig 9. Snapshot of the vehicles tracking

Vehicle tracking client is tested for different smart phones with different versions and it is proven to be fast and reliable. The server programme is tested on different browsers and it works as designed for all the browsers.

V. CONCLUSION

The vehicle tracking system is presented in this paper. The project objective is to develop a system to track vehicles using the Android operating system to get the location of vehicles and send it to the server. The server is a computer programme that receives multiple vehicles' locations through a modem and displays these locations on a map. This objective has been met successfully. The project scope is enhanced to allow users to track one vehicle rather than all vehicles at once.

VI. ACKNOWLEDGMENT

The authors would like to thank Husam Elmutasim, a Graduate Student of the Department of Electrical and Electronic Engineering University of Khartoum, for reviewing the manuscript. The authors also would like to thank Dr. Ghassan Mohammed Taha Ali from the Department of Electrical and Electronic Engineering, University of Khartoum for useful discussions and assistance during the project.

REFERENCES

- [1] Android (operating system) [Online]. Available: [http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system)) [Accessed 22 12 2013]
- [2] Android Software development [online] Available: http://en.wikipedia.org/wiki/Android_software_development [Accessed 22 12 2013]
- [3] Application Fundamentals [online] Available: <http://developer.android.com/guide/components/fundamentals.html> [Accessed 3 1 2014]
- [4] Starting an Activity [online] Available: <http://developer.android.com/training/basics/activitylifecycle/starting.html> [Accessed 5 1 2014]
- [5] Global Positioning System [online] Available: http://en.wikipedia.org/wiki/Global_Positioning_System [Accessed 3 8 2014]
- [6] Google Maps Overview [online] Available: <http://www.ariadne.ac.uk/taxonomy/term/13922?articletype=&term=&organisation=&project=&author=henry%20s.%20thompson> [Accessed 5 8 2014]
- [7] JavaScript [online] Available: <http://en.wikipedia.org/wiki/JavaScript> [Accessed 5 8 2014]
- [8] MySQL Enterprise Edition [online] Available: <http://www.oracle.com/us/products/mysql/mysql-ee-prod-guide-487992.pdf> [Accessed 7 8 2014]
- [9] Google Maps JavaScript API V3 [online] Available: <https://developers.google.com/maps/documentation/javascript/tutorial> [Accessed 5 8 2014]
- [10] Database management system [online] Available: <http://en.wikipedia.org/wiki/Database> [Accessed 7 8 2014]