Real Time and Past Positional Location Analysis of Friends in a Social Network Using Smart Devices

Dr. Sanjeev Dhawan[1], Nishu Dhundwal[2]
[1]*Faculty of Computer Sc. & Engg,* [2]*M.Tech. Student,*
Department of Computer Science & Engineering,
University Institue of Engineering & technology (U.I.E.T),
Kurukshetra University Kurukshetra (K.U.K), Haryana, India

***Abstract*: Nowadays, mobile applications have been used for certain purposes. The fact that mobile phones are comprehensively used by almost everybody in this world makes mobile applications completely more effective & influential with unmatchable & unthinkable capabilities. The application that we have decided & planned to develop will enable a user to perform significant tasks namely: Self-locate the geographical positional coordinates and Cloud-based user's movement tracking. Our application would fortify the users to use their social networking app. by locating the users on globe in a real time fashion. It would enable the users to track the movement and precise geographical coordinates on Google Maps which would enhance the experience of using the social networking application by remaining in contact with the friends and colleagues in a real time environment.**

Keywords: Android, GPS Tracker, LBS, Location coordinates

I. INTRODUCTION

According to a new report from the research firm Berg Insight; "revenues from mobile location based services (LBS)in the European market will grow by 34 percent annually to reach €622 million in 2010". This number shows how important LBS applications are becoming to mobile users [1].
LBS [1] are the mobile services in which the user location information is used to provide a service. The user location information consists of X-Y coordinates generated by any given positioning technique such as Cell-ID, GPS, etc.
The GPS is the most competent positioning technique. With a purpose of being used in navigation system, it was formulated. Due to integration of GPS with some mobiles and also due to reduction in the size of the GPS receivers; GPS has become one of the most significant service providers in the LBS.
Before developing this application, we considered the tools which are already working on this concept. As we know, most of the GPS applications help the user to determine their own location but they had some inherent problem which we identified as below:

- They cannot be used as a spy or tracker application. In other words, if we need to find or track a person or multiple user of the application, it cannot be done.
- The available GPS applications work on a very low-level accuracy i.e. they obtain a generic location name from the satellites and not the precise location
- The location which is obtained from the satellites is not very detailed. It gives the user only the name of the location and country in which he/she is in. If the user needs to have some more detailed information like the exact positional coordinates, it is not possible.
- The movement location coordinates are always plotted on the map. If the user wishes to see a three dimensional view of his movement coordinates, it is not possible
- The GPS applications do not store the past data of the location coordinates for the user.

We identified these issues and keeping them in mind, attempted to solve them in our application as mentioned below:

- We had divided our application into 2 phases i.e. Mobile Part & Web Part. The objective of this division is to obtain the location coordinates of the mobile user movement on a remote server via GPRS in order to track it.
- We have also implemented the feature of high-level accuracy under which we can get the precise current location of the user with the radius of 500m – 1 k.m. Our application fires the reverse lookup command to give the exact name of the location near to which the user is located.
- If the mobile user wishes to know his/her current location information, our application obtains the detailed information from the satellites using GPS and transmits it on the user's handset via GPRS. This location information contains all the positional coordinates, locality name, country name etc.
- If the user's movement needs to seen from different angles, we have implemented the Google API under which the trail of user's positional coordinates of the movement are plotted on Google Map, Satellite, Earth & Hybrid modes so that we can view his/her current location from various angles.
- In the remote server, all the positional coordinates which are fetched by GPS and sent by the Android device through GPRS on the remote server are collected in a database and kept for future referral. Since our application is built to track multiple users, we store individual data of the user's past movement coordinates and the remote web application displays that data in a tabular form. This data consists of: Timestamp, Latitude, Longitude, Altitude, Exact Location (which is fetched through Google reverse lookup function).

II. SYSTEM ARCHITECTURE

In our application, we have developed 3 tiers namely Android Tier [3, 4], Web Server Tier and Database Tier. All the 3 tiers have been developed keeping the objectives of wireless cloud communication into consideration. Figure 1 shows the main elements that construct the system which are GPS, the android tier, cloud tier and the database tier. The core algorithm or logic being used in our application uses the below steps:

- We have developed the Android Tier using Java. The Android module is the application which we have developed using Advanced Java. We make this tier point to the web server or cloud tier using web services.
- The cloud or web server tier is connected to the Android tier wirelessly using Java web services. The cloud tier is developed using Model, View and Controller (MVC) architecture [6, 7] of Java web development.
- We make the Android tier connected to the web tier using a public IP address. In order to create a private cloud, we use our own USB internet connection. After the connection is established, we turn off the firewall and antivirus programs within our application so that any request coming from the remote mobile client running on Android is not blocked and allowed.
- The third tier in web server is the database tier which has been implemented using Hibernates framework.
- When the Android mobile tier is activated, it starts collecting the latest GPS location coordinates i.e. Latitude, Longitude and Altitude of the user. As the fresh coordinates are received from the geosynchronous satellites via GPS using Google Maps Reverse Lookup API, they are delivered to the web component running on MVC architecture through Java web services via HTTP.
- As our application is meant of real time tracking of multiple users individually, it performs so through a unique identifier of every user i.e. user id. In other words, every user has an account created on the web server after which a username and password is returned through a sign up.
- This username and password is also stored in the database tier as well. The database consists of 2 tables namely, Users and Location. The table named Users stores the registration details of the users whereas the Location table stores the GPS location coordinates of the mobile users. In the location table, apart from Latitude, Longitude and Altitude, other significant parameters like GPS Accuracy, Timestamp are also stored.
- Since the coordinates are received from the Android mobile device via HTTP, they are delivered through Java web services to the web server. The web server receives the coordinates and inserts them in the database using Hibernates. Not to mention that the connectivity of the Android and web server tier is through a public IP address.
- The application that executes on the Android device assumes the role of a spy application so that the Android user never knows that he/she is being tracked remotely on a web server. This spy application is launched once the user logs in with the same username and password which was created on the web server during sign up. After successful sign up in the device, we create a background web service that collects the latest GPS location coordinates of the user and transmits them to the web server or cloud tier on a specified public IP address through Java web services.
- After the coordinates are stored in the database at cloud tier, they shall also be plotted on the Google Map, Earth, Satellite and Hybrid variants using Google markers. The Google markers will then be sequenced to make the Administrator user aware of the movement trail of the Android user. Additionally, all the Google markers shall be connected with each other using a RED line in order to display that movement trail.

III. IMPLEMENTATION

We have implemented the application in such a way that it works efficiently and without any error and with the maximum turnaround time. The codes have been written in such a way such that they produce maximum turnaround and response times.

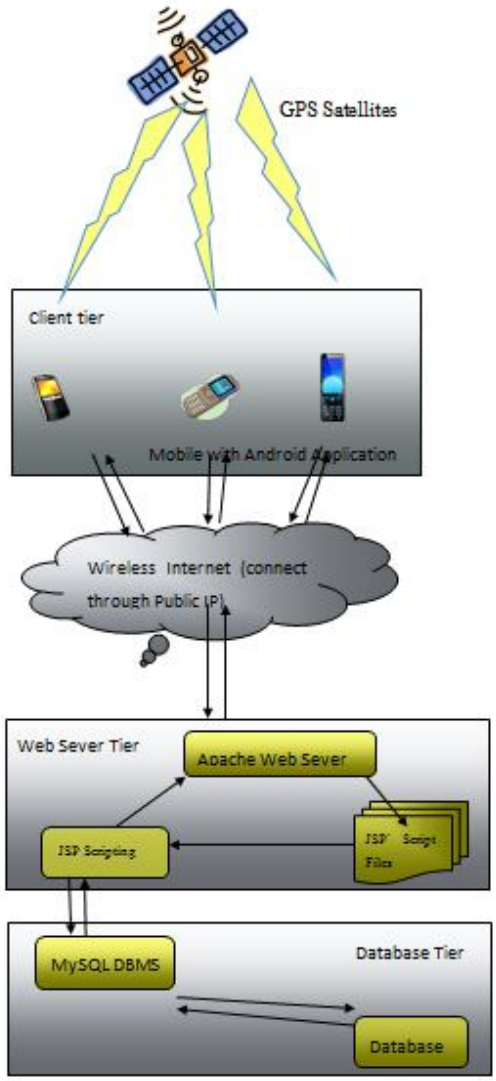

Figure 1: System Architecture

The application implements following tasks listed below in several tiers. Every tier has been coded using Java in such a manner that it communicates with each other using web services with the help a public IP address. The Android app. or Android tier is coded using Java with the help of Android libraries [5].

It is connected with the cloud or server application using web services. In other words, using the Android application, the user will be to know his/her current GPS location coordinates with the help of Google's reverse look up function. Once the coordinates are received, they will be transmitted to the server application running on Apache Tomcat web server using an IP address.

The coordinates received by the server application will be plotted on Google Map, Earth and Satellite with the help of a RED marker which will signify the current trail of movement of the user. All the significant GPS parameters will be fetched through consulting the geosynchronous satellites orbiting the earth by making the use GPS technology.

The coordinates being fetched from the GPS satellites will be inserted in the database on the basis of the user ID who has delivered those coordinates. As we can track the movement of multiple Android users at the same time on Google Map, Earth and Satellite, their coordinates shall also be stored in the database with their user ID as the primary key**.** Additionally, when the user needs to set a location notification alert or a handset profile transition alert, the location will be stored in the device in the form of coordinates so that when those coordinates are received, they are checked from the database and the required event could be triggered appropriately. For ex: A notification event or a handset profile transition event

IV. RESULT AND DISCUSSION

Devising an objective to harness the power of smart phone technology, we have developed an application which will empower the user to use GPS feature for not just knowing the location but also utilizing that information for more meaningful tasks such as:

A. Self-Locate The Geographical Positional Coordinates

Through this functionality, the Android device will send the current global positional information on the handset consisting of important details like *Latitude, Longitude, Altitude, Location, Timestamp etc*. Apart from them, there would be other significant parameters which will be fetched through consulting the geosynchronous satellites orbiting the earth by making the use GPS technology.

GPS technology works on the concept of accuracy i.e. if we turn ON the GPS module in the handset, it will contact the geosynchronous satellites in order to obtain the current positional information of the handset and once that information is obtained, it will be transmitted to the device using web technology which is essentially accessed using GPRS protocol. There is a time gap of short interval in which the mobile device uses GPS and consults the satellites to ask for the current location information of the device.

That is to say, the device does not uses GPS continuously; it refreshes the position after every 1 or 2 minutes and then, it is dependent upon the GPRS bandwidth speed how soon those coordinates are transferred to the device. GPRS modes on 3G networks are fast due to which the user can quickly see his/her current positional coordinates on the handset as soon as they are obtained from the satellites via GPS.

Another important aspect of GPS is, it does not work effectively in underground or basement locations because of the interference. Also, in some sensitive locations on the globe, GPS signals are barred. There also, it will not work. This is also one of the prime reasons why on Google Earth, locations within USA are seen very clear whereas in India, only prime locations are clearly visible. This technology works effectively in an open environment.

Using this application, the Android user would be able to able to make the use of above aspects in order to determine the current positional information which will be delivered on his/her handset.

B. Cloud-Based User's Movement Tracking

Through this functionality, the obtained GPS geographical positional coordinates namely *Latitude, Longitude and Altitude* will be sent by the application over the web on a remote server which is located on the cloud. The significance of sending these coordinates to the server is to track the movement of Android device *live*, remotely, may be even without the user knowing that he/she is being tracked or spied. The application of such concepts are in real world scenarios like armed vehicles tracking during war situations, missile tracking, school bus tracking, fuel tankers or cash carrying vehicles tracking etc. Most importantly, this feature will work as a background process and the user of this application will never know that he/she is being tracked unless informed so. This application will collect the data and store it in the database. It would then fetch that data and display it in the form of movement trail. In other words, it would take the help of Google API to display the location of mobile device on *Google Map, Google Earth and Google Satellite*. These locations will be plotted sequentially in the form of a trail wherever the user moves on the globe and if user wish to view the exact coordinates of that location it can be done so by pointing to that location movement of the user. All the significant GPS parameters will be fetched through consulting the geosynchronous satellites orbiting the earth by making the use GPS technology. The coordinates being fetched from the GPS satellites will be inserted in the database on the basis of the user ID who has delivered those coordinates. As we can track the movement of multiple Android users at the same time on Google Map, Earth and Satellite, their coordinates shall also be stored in the database with their user ID as the primary key.

After successful sign up in the device which we are going to track, a background web service is created that collects the latest GPS location coordinates of the user and transmits them to the web server on a specified public IP address through Java web services. These coordinates that are also stored in the database, they shall also be plotted on the Google Map, Earth, Satellite and Hybrid variants using Google markers. The Google markers will then be sequenced to make the Administrator user aware of the movement trail of the Android user. Additionally, all the Google markers shall be connected with each other using a RED line in order to display that movement trail. The screenshots below shows how a user's location coordinates transmitted by android device user is being collected and plotted on google maps.

The screenshot below shows movement trail of user being tracked. Also, the location of user being tracked is being stored in tabular form as well as shown in fig 2.

When we click on the hyperlink "*show on map*", the location of mobile user is being plotted on Google Map using Google markers as shown in Fig2.

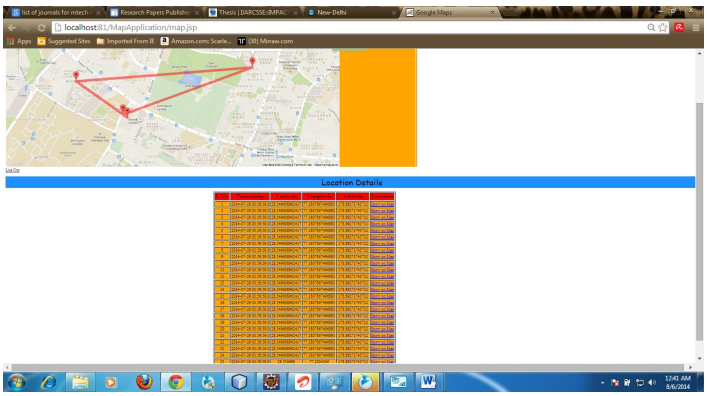

Fig. 2 Screenshot 1

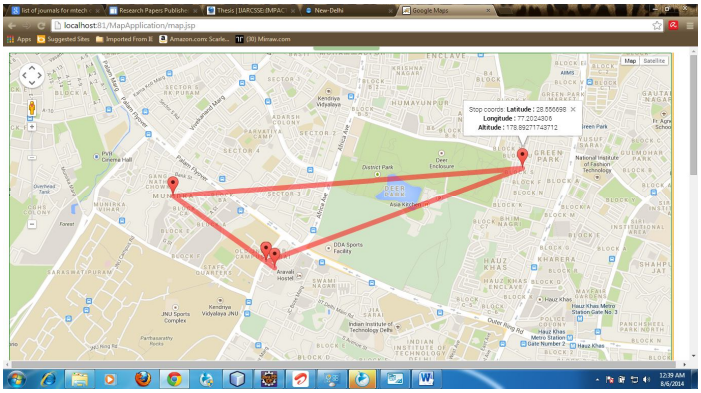

Fig. 3 Screenshot 2

V. CONCLUSION AND FUTURE SCOPE

We can hereby conclude that this application will help the user to see his/her location data including coordinates and other information. It will also help other user to track the actually location of the Android mobile handset at the remote location on the server. The unique feature of this application is, it plots the location trail of the user's movement on a Google Map & Earth thereby giving a clearer real time view. The user's location information is also stored in the database for future referral. Through this application, different users would be able to sign-up with different accounts and can be tracked. This application provides additional mechanism for automatic changing the handset profile according to the handset location on the map. This feature is classified under the artificial intelligent part of computer programming science.

Although this application has been developed to introduce uniqueness and remove some of the shortcomings in already developed applications, it still has got scope of improvements:

- We can develop this application so that it tracks multiple users at the same time. We would also plan to develop this application in order to track multiple mobile handsets, simultaneously i.e. once this application is installed on multiple devices, the location coordinates for all of them will be sent on the remote PC where the web services would be running which would enable the user to track the location of all the users at same time.
- We can further improve this application in such a way that if the user is rapidly moving, his/her location data should be correctly captured.
- We can further improve this application to check the GPRS connection failure in which case it can still collect the location coordinated through GPS and can keep buffering so that when GPRS comes back, it should send them all to the remote server.

REFERENCES

[1] Axel Küpper , Location-based services, fundamentals and operation, WILEY, 2nd edition, 2005.
[2] Rick Broida, *How to Do Everything with Your GPS*, McGraw-Hill/Osborne, 2nd edition, 2004.
[3] Donn Felker, *Android Application Development For Dummies*, (Dec 7, 2010).
[4] Ed Burnette ,*Hello, Android: Introducing Google's Mobile Development Platform (Pragmatic Programmers)* , (Aug 4, 2010).
[5] Reto *Meier* ,*Professional Android 2 Application Development (Wrox Programmer to Programmer)*, (Mar 1, 2010).
[6] Ray Rischpater ,*Beginning Java ME Platform (Expert's Voice in Open Source)* , (Oct 23, 2008).
[7] James Edward Keogh, *J2ME: The Complete Reference* ,(Feb 27, 2009).