International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 4 Issue 7, July-2015

Smart Phone Powered GPS Positional Navigation and Cloud Based Live Location

Thaair M.Y. Ameen

Assistant Lecturer, University of Mosul Presidency, University of Mosul, Iraq

ABSTRACT

This paper designed to develop an application that enables a mobile handset user to obtain location coordinates like Latitude, Longitude & Altitude from the satellites. In addition, through this application we also aim to track any particular user of this application from a remote server through plotting his positional coordinates at regular time intervals on a map. Plus, it will also give a detailed report of the user's movement. This application will make a combinational use of GPS & GPRS technologies to fulfill the above objectives. The GPS technology use to fetch the location coordinates like Latitude, Longitude & Altitude , and GPRS feature will be used to send those coordinates over a remote server to PC where they will exactly plotted on a Google Map in the form of a trail depicting the user's movement. The trail will consist of milestones which the user visited. It will also send those coordinates in a database along with timestamp so that if any user wishes to see a detailed report of the user's movement, he can do so. The application also will make the use of reverse lookup function to convert & display the exact location of the user from the fetched coordinates. The project it consists of two parts: Web part running at the remote web server and mobile part would be the actual application that will be developed on android platform.

1. INTRODUCTION

Today, mobile applications are used for significant objectives. The fact that mobile devices are extensively used by virtually everyone on this planet makes mobile apps all the more useful and powerful with unique and unimaginable capabilities. The application that we have decided & planned to develop will enable a user to track the location of a mobile device using GPS technology. This application would be developed on Android platform and once installed on a mobile device it will send the location coordinates namely Latitude, Longitude and Altitude on a remote PC. This application will make the use of GPRS feature to send the location coordinates of the mobile handset. We assume that the PC to which it will send the coordinates will be connected to web. We will develop web services/application to which our mobile application will connect and transmit the data. The web application will collect the data and store it in the database. We would then fetch the data and display on a map. In other words, we would be making use of Google API to display the location of mobile handset device on the map by indicating it with a pointer or dot. The page on which map would be displayed will be refreshed every 10 seconds to display the updated location of the user with respect to the coordinates sent by the mobile application. The mobile application will collect the location coordinates from GPS satellites.

Hence, the database could contain location details as below:

- Time
- Latitude
- Longitude
- Altitude
- Location

In addition to plotting the coordinates on Google Maps, we will also display them in a tabular form. Interestingly, there will be a column in that table which will contain a link that will enable the user to see the exact location corresponding to that coordinate on Google Map. In order to determine the precise location name corresponding to Latitude, Longitude & Altitude coordinates; we will use the Google's reverse lookup feature.

2. HTTP FRAMEWORK

HTTP uses TCP as its transport protocol. It uses a client-server model. The Symbian OS supplies a HTTP framework to support all HTTP-defined request methods, including GET and POST. It provides a set of Application programming interfaces (APIs) which helps programmers to develop applications without needing to pay attention to details of the HTTP stack. The framework also supports HTTPS as well, for added security. To develop an HTTP application, a HTTP session needs to be established using the API called Http Session. Within a session, a client sends a request to and waits for a response from an HTTP servers. Since creating a session consumes time and memory, it is recommended that only one session is created in an application. Fortunately, multiple HTTP transactions can take place during one session. Several different session properties can be set, for instance, the network connection could be set for the session so that a user prompt will not be triggered to ask the user to select the network connection to be used. The ownership of this connection belongs to the application, thus after finishing using the connection it is the application's responsibility to close the network connection - rather than this being a responsibility of the framework. The session properties apply to all transactions within the session. However, some properties could be set for a specific transaction overriding the session's settings. The processing of an HTTP transaction is encapsulated in an API called RHTTP Transaction. It creates the request message, including the header and the body, and submits the message to the HTTP server. To customize the request body, an object which implements the interface M Http Data Supplier is needed. This object supports creating the body in parts if the data is large. The HTTP framework provides a way to monitor the transaction so that the developer can concentrate on dealing with processing the data. To monitor a transaction, the interface M Transaction Callback needs to be implemented. Table 1 shows the common events supported by the HTTP framework.

Event Name	Description
EGotResponseHeaders	Indicates that the response has been received and the status line and header field information can be retrieved
EGotResponseBodyData	Indicates that body data is ready for access
EResponseComplete	Indicates it is the end of the body
ESucceeded	Transaction completed OK, the session needs to be closed
EFailed	The cause of the failure needs to be investigated

Table (1): HTTP Transaction Event Code

String Poll is an important concept for the HTTP framework. It efficiently deals with standard strings. This is particularly useful for HTTP, because HTTP uses a lot of well-know and standard strings such as the header field name. There is member data in the RHttp Session class called string pool into which the commonly used HTTP strings will be loaded. They can be conveniently passed to an HTTP API. For instance, when the request method needs to be defined as POST, the value string Pool. String F (HTTP::EPOST, RHTTP Session:: Get Table ( )) can be directly assigned to.

3. DATABASE STRUCTURE

This application contains one database called Tracker and 2 tables in it as mentioned below:

a. Location: This table stores the location coordinates of the user's movement which has logged in the mobile application and has a corresponding account in the remote web server also. The location coordinates are fetched by the mobile application with the help of GPS & GPRS. Below is the structure of the table:

Table (2): Location details character

FIELD	TYPE
Latitude	**Double**
Longitude	**Double**
Altitude	**Double**
Accuracy	**Float**
Speed	**Float**
Sent Time	**Timestamp**
User id	**Integer**

b. Users: This table stores the registration details of the user who sign up in the web application.

Table (3): User details character

FIELD	TYPE
ID	**Integer**
Name	**Varchar**
Email	**Varchar**
Phone	**Varchar**
Password	**Varchar**

4. TESTING RESULTS

In the testing of this project, we created different test cases as below in order to perform module-wise testing. The test cases were formulated keeping the overall objectives of the applications into consideration. In other words, we tested a certain module to ensure that it should perform its own function in addition to some other related functionality with other modules, if necessary. We recorded our testing results by giving different inputs to the modules and observing the actual result as against the expected one. Wherever, the test result failed, we incorporated the essential modifications to correct it.

S.NO.	MODULE	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
1)	Login (Mobile Part)	Incorrect / Blank Username & Password	The user should be prompted through an error message	The error message is appearing	**ok**
2)	Login (Mobile Part)	Correct Username & Password	The GPS service should be activated which must start sending the location coordinates on the remote server via GPRS	The application is sending the coordinates on remote server database	**ok**
3)	My Location (Mobile Part)	--	The current location data must be fetched using GPS & GPRS	The data is being fetched	**ok**
4)	Login (Web Part)	Correct Username & Password	The location coordinates should be correctly fetched from the database & plotted on Google Earth, Map & Satellite along with a tabular display	The coordinates are getting plotted correctly on Google Earth & Map and are also displayed in a table	**ok**
5)	**Sign Up (Web Part)**	**Registration Details**	**The details should be successfully inserted in the database & the user account should be created**	**User account is getting created with details being successfully inserted in the database**	**ok**

Table (4): Module Testing

5. VEHICLE POSITIONING

This application integrates satellite-positioning systems that tell people where they are with non-voice mobile services that let people tell others where they are. The Global Positioning System (GPS) is a free-to-use global network of 24

satellites run by the US Department of Defense. Anyone with a GPS receiver can receive their satellite position and thereby find out where they are. Vehicle positioning applications can be used to deliver several services including remote vehicle diagnostics, ad-hoc stolen vehicle tracking and new rental car fleet tariffs. The Short Message Service is ideal for sending Global Positioning System (GPS) position information such as longitude, latitude, bearing and altitude. GPS coordinates are typically about 60 characters in length. GPRS could alternatively be used.

6. SOFTWARE

At the heart of the tracking system is a collection of software libraries developed as part a software support system for location. The provision of a Android user interface to common location tasks allows the implementation details of complex tasks to be hidden, thereby offering the systems designer a cleaner workflow. Software abstractions of tracking and communication tasks have been created, allowing the user to enter functionality into the application. An Android user interface to location has been implemented to allow access to data from the GPS module in mobile. Using an abstraction model for tracking interfaces, the process of gathering data is simplified. This in turn allows a modular approach to application development.

The framework supports Google's cloud to device messaging service for communication between the application server and Android user. It allows two-party application Android to send currently location to their server's applications. The Android application is not designed for sending a lot of user content via the GPRS. Rather, it should be used to tell the server that there is new data on the Android application, so that the application can fetch it. Communication module contains C2DM Bridge part for sending location information and GPS part for on/off operation. It is implemented using Net Beans 6.9. The Android user interface module used the Eclipse Android developer's tools for development of user interface. This interface is used for knowing location areas current. JDK is integrated in Eclipse IDE, coding is done by using android application fundamental concepts. This interface runs as android application on Android smart phone.

The problem with using JDBC in a high-load environment is that it can become overwhelmed with requests, and begin dropping them or delaying them, causing a bottleneck in the architecture that slows down the entire application. To avoid this situation, Tomcat uses a technique called connection thread pooling, implemented through a technology called Commons DBCP. Rather than creating new connections every time a new request comes in, the requests are queued, and matched with a pool of pre-generated connections as they become available. Using JDBC also allows developers to assume that each request will receive its own JDBC connection, which significantly simplifies the transaction code. Tomcat makes it easy to use this technology by allowing it to be defined as a JNDI Resource, and handling the actual transaction of the connections itself. If this still seems confusing, don't worry - we'll go over all of the steps required to get these technologies working for you in the next section.

7. RESULTS

The connecting mobile with a computer through C2DM needed to registration in C2DM is recorded with computer technology C2DM using HTTP POST and return token .As well as in case of connecting C2DM with mobile, Mobile request registration ID and return a special type of registration ID to mobile. After my server is registered with Google and has at last one device it can send message , C2DM is ready to be used and real data is fetched from my server.

***a- Mobile* handset** **b- Web Browser**

Figure (1) User Registration

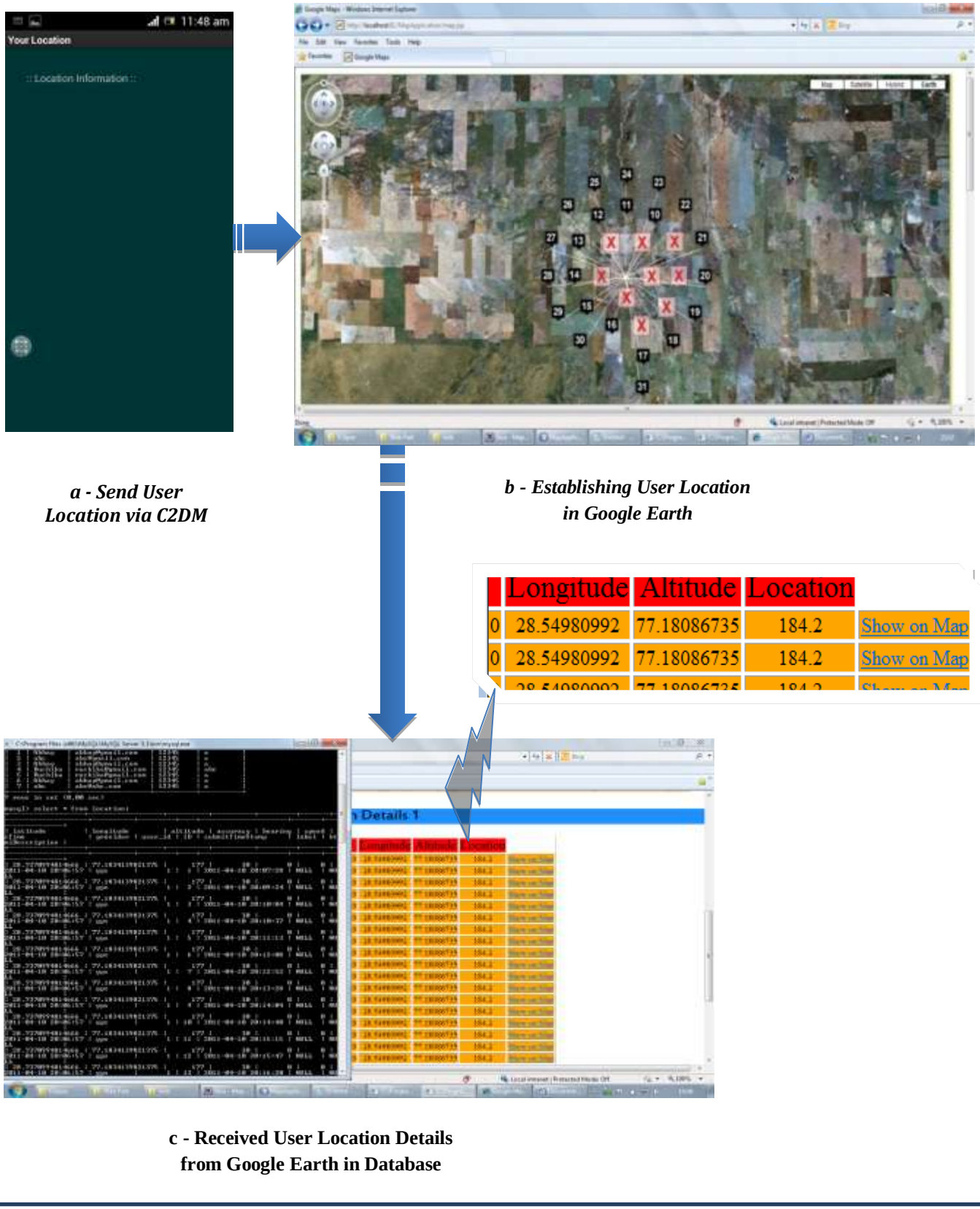

Figure (2): Architecture of Track Location Application

8. DATA FLOW DIAGRAM

A **data flow diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design). A DFD shows what kinds of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

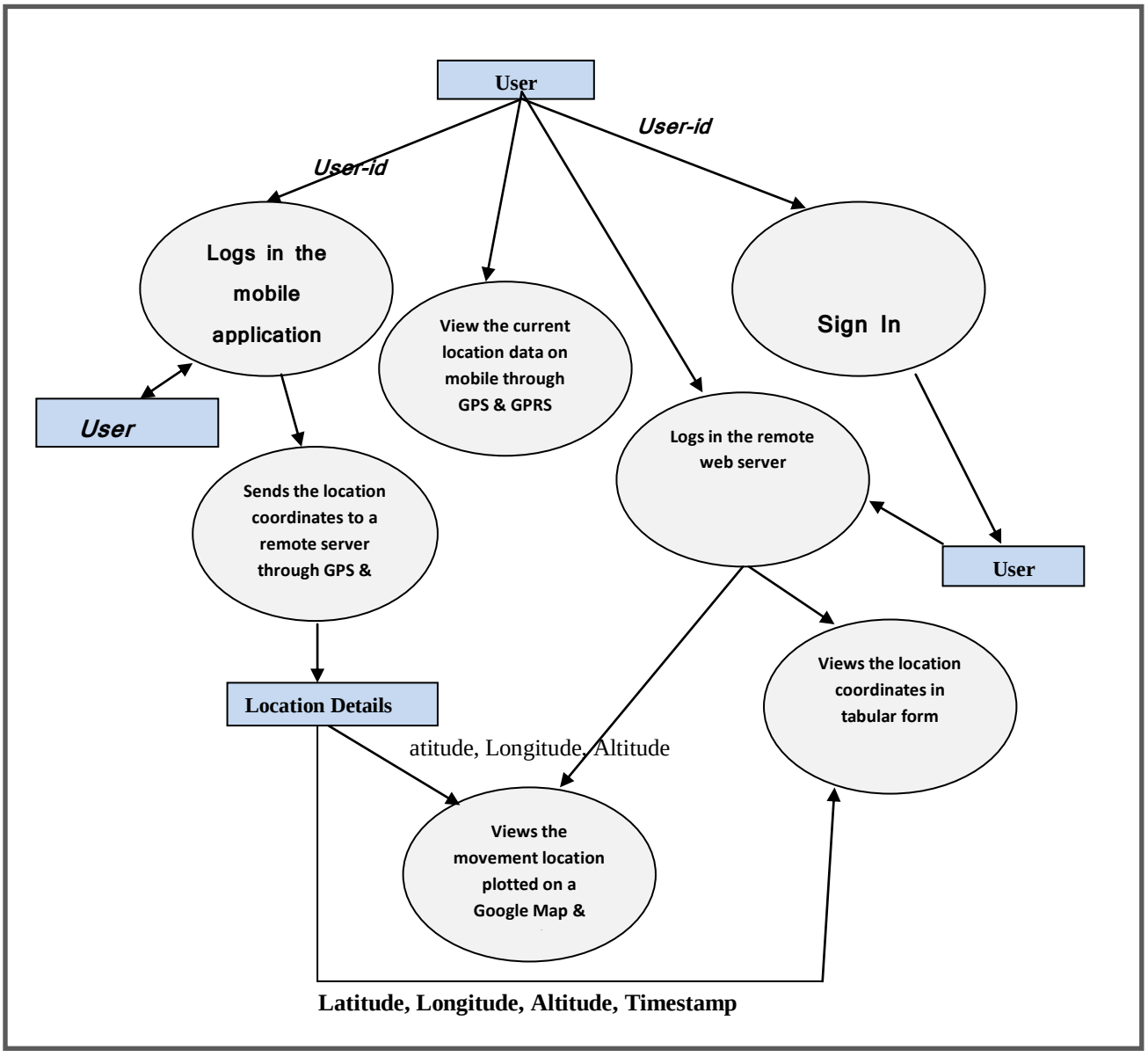

Figure (3) Data flow diagram

CONCLUSIONS

- Using this application, the Android user will be able to receive his/her current location coordinates through GPS
- Using the cloud part, the user shall also be able to deliver those coordinates to a remote server using GPRS where he could be tracked on Google Map

We can hereby conclude that:

1. This application will help the user to see his/her location data including coordinates and other information.

2. It will also help other user to track the actual location of the Android mobile handset at the remote location on the server.
3. 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
4. The user's location information is also stored in the database for future referral.
5. Through this application, different users would be able to sign-up with different accounts and can be tracked.

FUTURE WORK

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 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]. Manav S., Anupom S., (2012), "Implementation of Location Based Services in Android Using GPS and Web Services",International Journal of Computer Science Issues, Vol.9, No.2, pp.237-242.
[2]. Ashokumar R., Prabha D., and Saranya R., (2012)," Personal Safety Triggering System on Android Mobile Platform ",International Journal of Network Security & Its Applications, Vol.4, No.4, pp.179-197.
[3]. Sandeep K., (2013),"Database Management by Using Java Database Connectivity Architecture of Java", Vol.2,pp.204-206.
[4]. Suhas H., Mahima M., (2012),"Android Based Mobile Application Development and its Security", International Journal of Computer Trends and Technology, Vol.3, pp. 486-490.
[5]. A. Schmidt and S. Albayrak, "Malicious software for smartphones," Technische Universit\ät Berlin, DAILabor, Tech. Rep. TUB-DAI, vol. 2, 2008, pp. 08–01.
[6]. E.Y. Chen and M. Ito, "Using end-to-middle security to protect against evil twin access points," 2009 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops, Kos, Greece: 2009, pp. 1-6.
[7]. A. Bose, X. Hu, K.G. Shin, and T. Park, "Behavioral detection of malware on mobile handsets," Proceeding of the 6th international conference on Mobile systems, applications, and services, Breckenridge, CO, USA: ACM, 2008, pp. 225-238.
[8]. Taylor, G., and Blewitt, G., (2000)"Road Reduction Filtering using GPS", Proceedings of 3rd AGILE Conference on Geographic Information Science, Vol.3, pp. 114-120.
[9]. Taylor, G.**, (**1999)," Improved real-time GPS positioning through the reduction of state space", Proceedings of GIS Research UK Conference.
[10]. Christoffer A., (2009), " GPRS and 3G Wireless Applications: Professional Developer's Guide".
[11]. Bob T.,(2008),"GPRS 100 Success Secrets", Emereo Publishing.
[12]. Michael J.,(2008), Enterprise J2ME: Developing Mobile Java Application.
[13]. Addison Wesley, Database Access with JDBC, July 2001, available a http://www.informit.com/articles/article.aspx?p=167843
[14]. Sun Microsystems, Java SE Technologies - Database, September 2009, available athttp://java.sun.com/javase/technologies/database/
[15]. Apache Tomcat, January2013, available at http://tomcat.apache.org/
[16]. The Eclipse Foundation, Eclipse homepage, March 2012, available at https://www.eclipse.org/
[17]. The NetBeans IDE, available at https://netbeans.org/community/releases/69/
[18]. The Android SDK, Android developers homepage, available at http://developer.android.com/sdk/index.html?hl=sk
[19]. The GSM, GPRS, EDGE Training ,April 2005, available at http://www.eogogics.com/wireless-technologies/gsm-gprs-edge