A SYSTEM FOR TRACKING AND RECORDING LOCATIONS OF ANDROID DEVICES

Milan Lukić[1], Goran Sladić[2], Stevan Gostojić[2], Branko Milosavljević[2], Zora Konjović[2]
m.lukic@4expand.com, {sladicg, gostojic, mbranko, ftn_zora}@uns.ac.rs
[1]*4Expand, Novi Sad*
[2]*Faculty of Technical Sciences, University of Novi Sad*

Abstract – *A location-based service (LBS) is a platform that provides information services based on current or known location, supported by the digital map platform. Most smartphones obtain location using Global Positioning System (GPS), cellular network or wireless network. In this paper we present the system for locating and tracking Android devices. It enables one person to track another and to receive notification when the tracked user leaves defined area.*

Keywords – *tracking, location, location-based services, smartphones, Android*

1. INTRODUCTION

Initially mobile phones were used for voice communication only, but nowadays the scenario has changed. Voice communication is just one aspect of a mobile phone. There are other aspects which are major focus of interest such as web browsers and location-based services [1].

A location-based service (LBS) can be defined as a service that depend on and is enhanced by positional information of mobile device [2]. A LBS is a mobile information service that extends spatial and temporal information processing capability to end users via Internet and wireless communications [3, 4].

In [5] authors defined LBS as a service where:

- the user is able to determine its location,
- the information provided is spatially related to user's location, and
- the user is offered dynamic or two-way interaction with the location information or content.

Location-based services have attracted considerable attention due to their potential to transform mobile communications and the potential for a range of highly personalized and context-aware services [6]. The potential for location-based services is evident from powerful and ubiquitous wireless devices that are growing in popularity [7]. They are the key enabler for a plethora of applications across different domains ranging from tracking and navigation systems, trough directory, entertainment and emergency services, to various mobile commerce applications [4, 8, 9]. LBS provide the user with contents customized by her current location, such as the nearest restaurants, hotels or clinics, which are retrieved from a spatial database stored remotely on the LBS server. LBS don't serve individual users only, but also play an important role in public safety, transportation, emergency response, and disaster management.

With an increasing number of mobile devices featuring built-in Global Positioning System (GPS) technology, LBS have experienced rapid growth in the past few years [1, 8].

Android platform is a new generation of smartphone platform launched by Google [10, 11, 12, 13]. Android supports location-based and mapping services, which is of concern to vast numbers of software developers. Until now, development of mobile LBS and mapping applications was complex and difficult, and often required paying high copyright fees to map makers [11, 13, 14]. Android is free and open source, providing an easy-to-use development kit containing flexible location-based services including map display.
It supports three different methods to locate device position; Global Positioning System (GPS), cellular network and Wi-Fi network using different implementations of the location provider.

In this paper we present a tracking system which uses smartphone with a built-in GPS module. The system enables collection and storage of a user's locations and monitoring of user's movement. GPS was used as the location provider due to highest precision, although the proposed system can be adjusted to use other location providers with the minimal effort.

The application periodically reads the GPS position of the client device, and then sends coordinates to the server, which processes and stores them in a database. Afterwards, the user who monitors the client can see the movements of the tracked user through web-based or Android application. The user's route is drawn on the Google map [15].

The proposed solution also provides means to define each user's monitoring area. In this case, if the user leaves defined area, the monitoring user will receive a notification informing her that the monitored user has left the area. This is especially useful for parents who want to track the movement of their children.

The rest of this paper is structured as follows. The next section describes the global system architecture. The third section gives a brief overview of the client module for sending user's location. The Android application for

displaying route is given in the fourth section. The fifth section describes the server side module of the proposed solution. In the conclusion, strengths and weaknesses of this solution are elaborated and directions of further research are given.

2. THE SYSTEM ARCHITECTURE

The proposed system consists of four modules:

- Android module (service) that periodically sends GPS coordinate to server,
- server module that collect and process data and provides administration of the system,
- Android module for displaying user's route, and
- web module for displaying user's route.

Figure 1 shows modules of the system. The Android service that periodically send information to the server is based on the *AlarmManager* Android component [11] which periodically "awakes" the device, reads the GPS coordinates and sends them to the server.

The web service that receives and processes information is a central part of the system. It receives data from Android devices and provides data to web and Android client applications.

The Android client application performs data visualization. This module also receives notifications form the server when monitored user leaves selected area.

The web module provides administration interface, also it is possible to see user's route using this application.

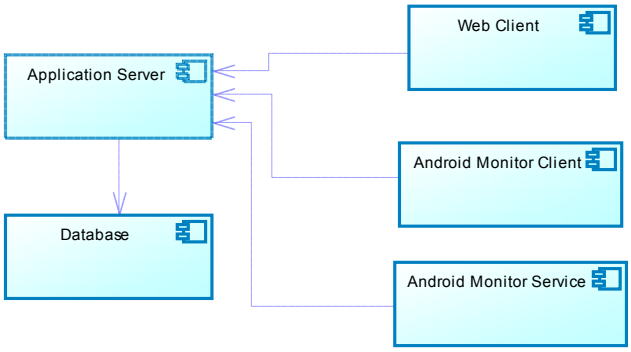

Figure 1. Modules of the system

Figure 2 shows the use case diagram of the system. *Administrator* performs administration of users, including their registration and administration of the system. Before starting to use the system, the user must be logged in. *Monitoring user* can create a monitoring request for the selected user. This user can track route of the monitored user and define zone for this user. Before *Monitoring user* can start monitoring the user, *Monitored user* must approve the request.

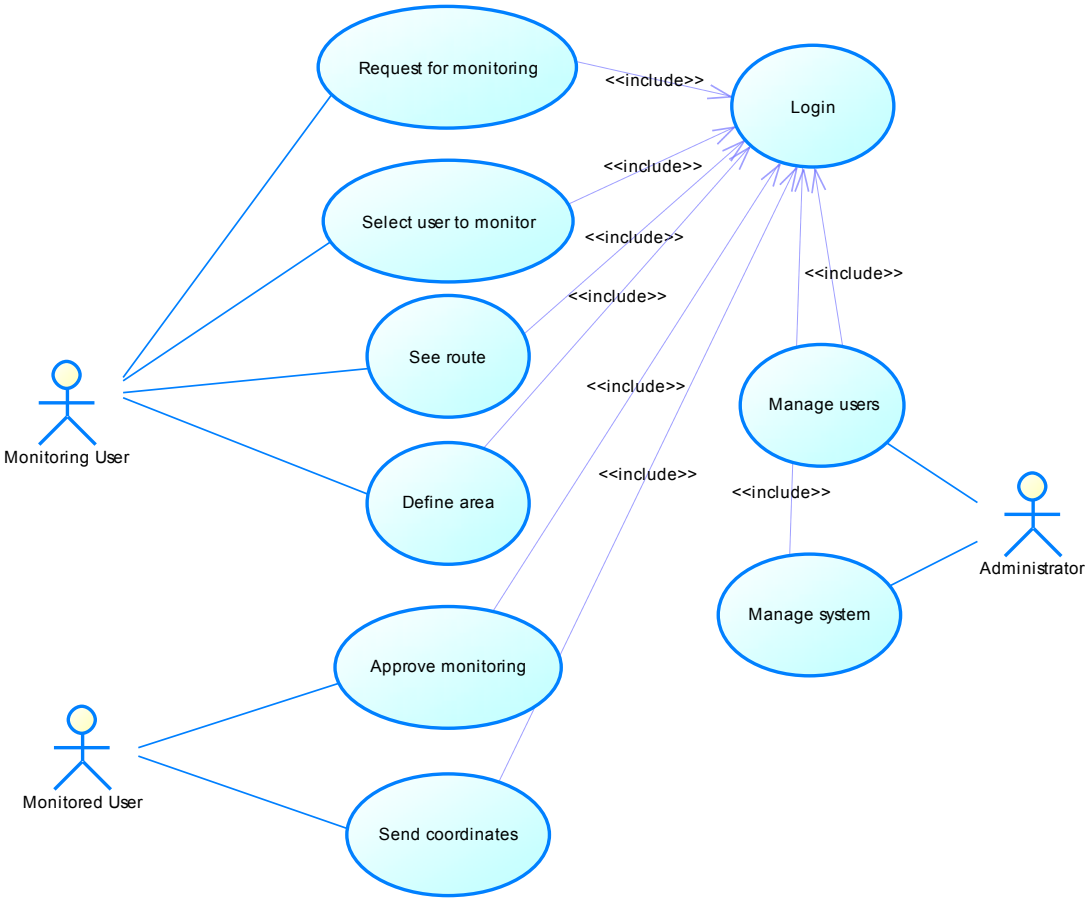

Figure 2. Use-case diagram

3. SENDING CLIENT POSTION

Figure 3 shows the class diagram for Android service that periodically sends location of the user to the web service. The *LocationPoolerInit* class performs necessary configuration and initialization of the service. *LocationPoller* inherits *BroadcastReceiver* [11] in order to broadcast device's coordinates [11]. This component is launched by *AlarmManager* [11]. It passes the work over to *LocationPollerService*. The *LocationPollerService* class reads device's location; it also wakes the device and ensures that the processor stays awake until the job is done. *LocationPollerService* uses the *PollerThread* class to look up the current location and to handle timeout scenarios. *The LocationReceiver* class accepts broadcasted messages with location data and through R*equestTask* sends them to the server. *LocationDelegate* defines methods which will be executed before and after a message is sent to the server.

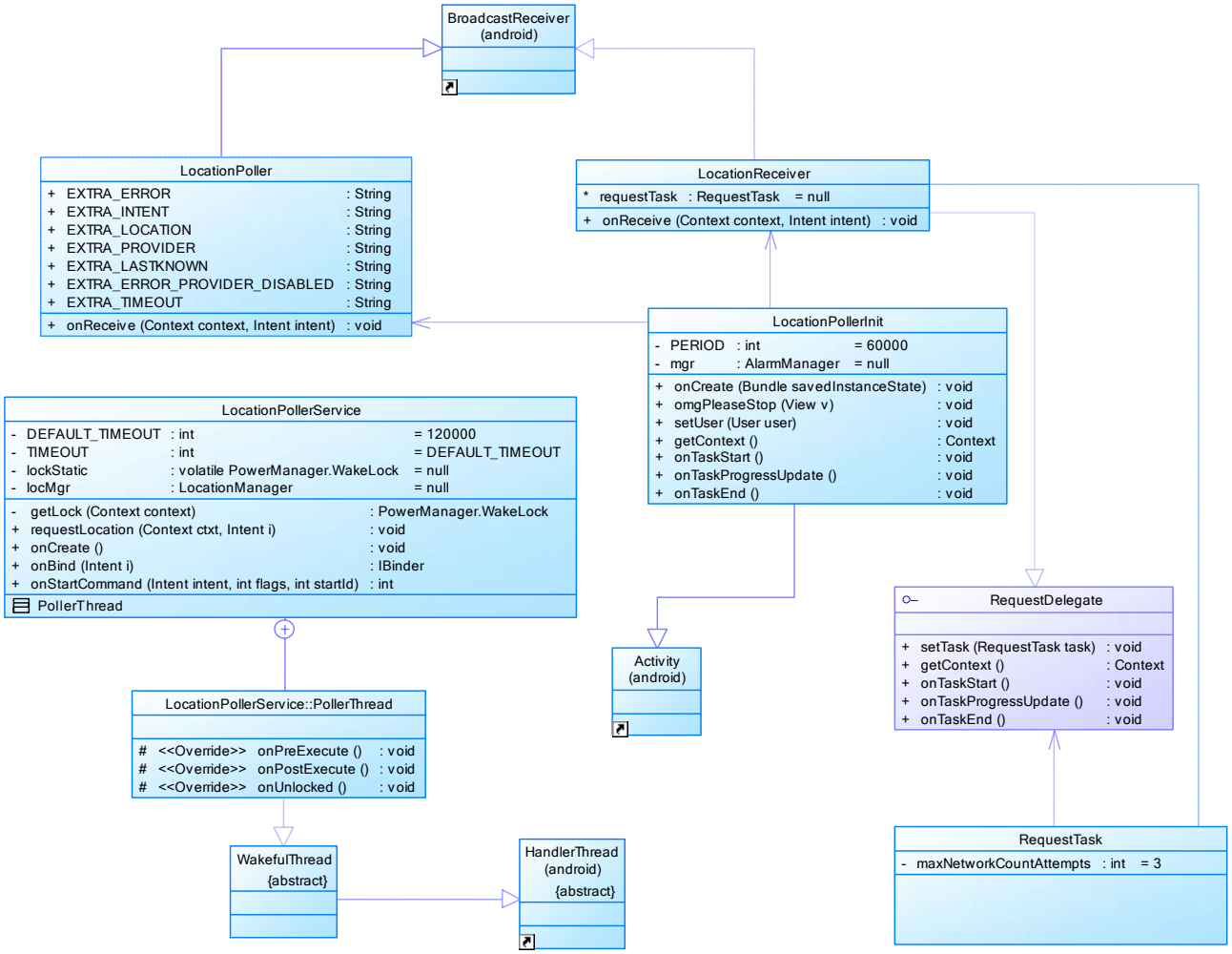

Figure 3. Android service for sending client location

4. DISPLAYING ROUTE ON ANDROID DEVICE

Classes presented in Figure 4 are used to display user's location and route on the Android device. The *MonitorActivity* class is the main class in the diagram. It inherits the Android *Activity* class and implements the *NetworkDelegate, MonitoredUsersDelegate* and *RoutesDelegate* interfaces. By extending the *Activity* class, *MonitorActivity* can interact with the user. The *NetworkDelegate* class defines methods which will be executed when the device opens and closes Internet connection. *MonitoredUsersDelegate* class defines methods which will be executed when the application receives a message about monitored user form the server, and *RoutesDelegate* class defines similar methods for processing routes received from the server. The *MonitoredUserRequestTask* class requests the list of monitored users (by the current user) to the server. The *RouteRequestTask* class requests the routes of the selected user to the server.

Figure 5 shows an example of the route drawn on the Android smartphone and Figure 6 shows an example of the defined monitoring area.

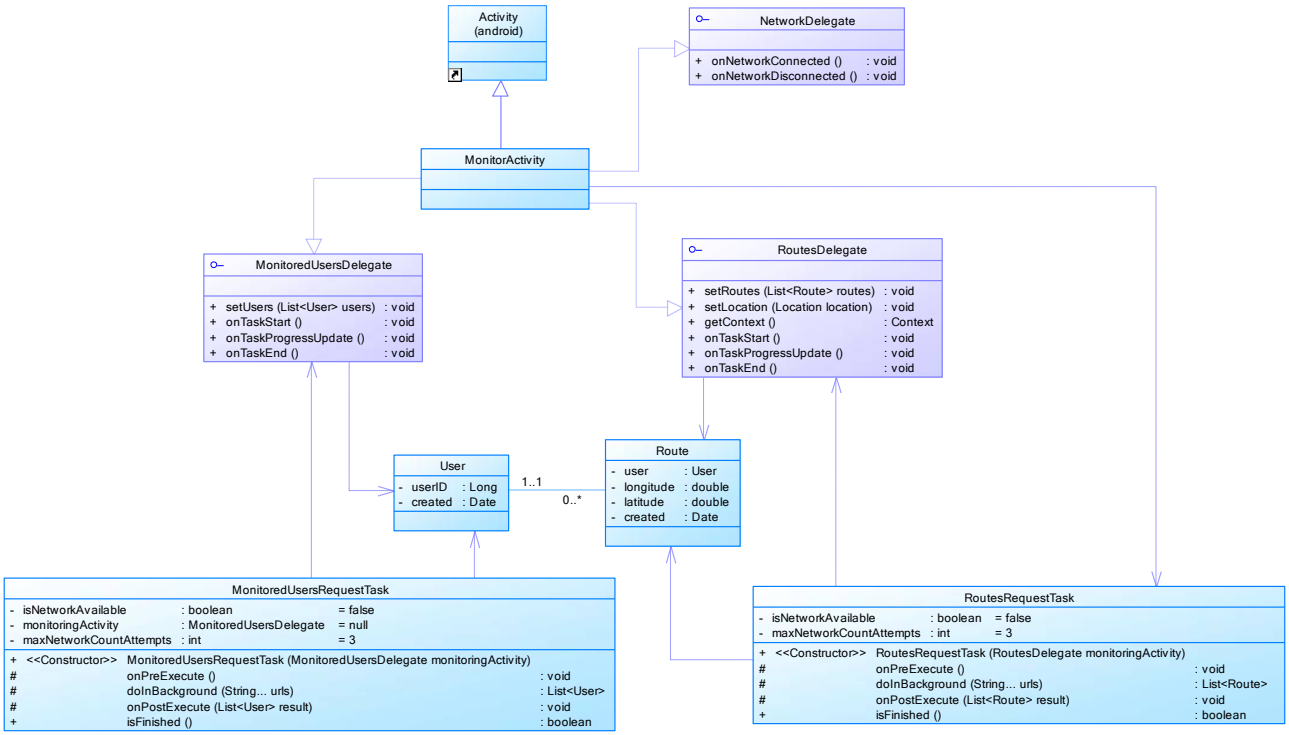

Figure 4. Android application for displaying user's location

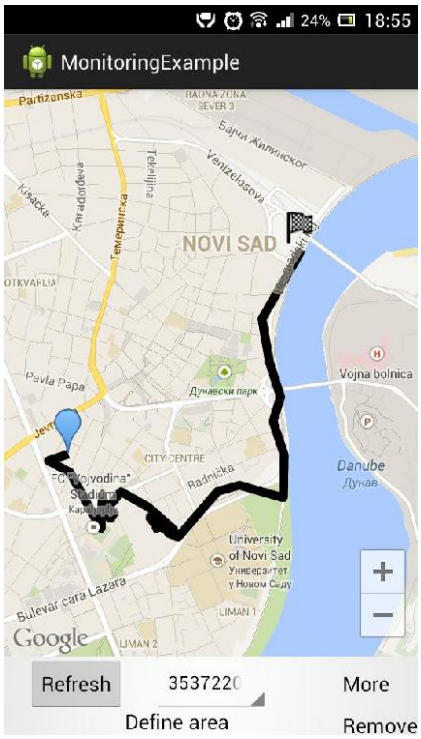

Figure 5. User's route

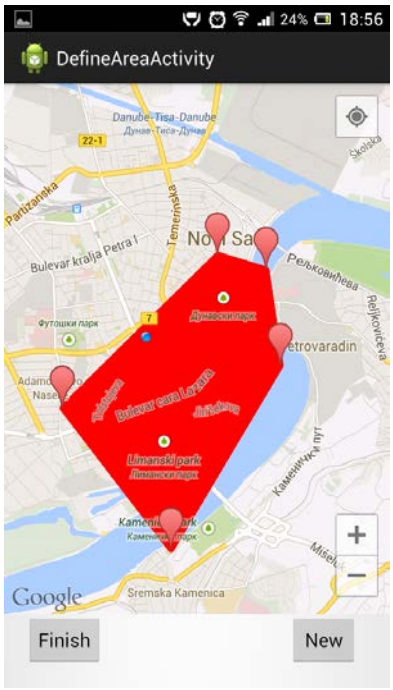

Figure 6. Monitoring area

5. SERVER MODULE

5.1. MONITORING SUBSYSTEM

The part of the server application used to define monitoring and monitored zones is presented in Figure 7. The *MonitorController* class is used for communication with client applications (Android and web) using REST protocol. It uses *MonitorService* implementation (*MonitorServiceImpl*) to manage and read users' monitoring areas. The *MonitorServiceImpl* class uses *MonitorRepositoryImpl* to access database. The *Monitor* class represents defined user's monitoring area since monitoring area can be defined for each class instance (the *area* attribute).

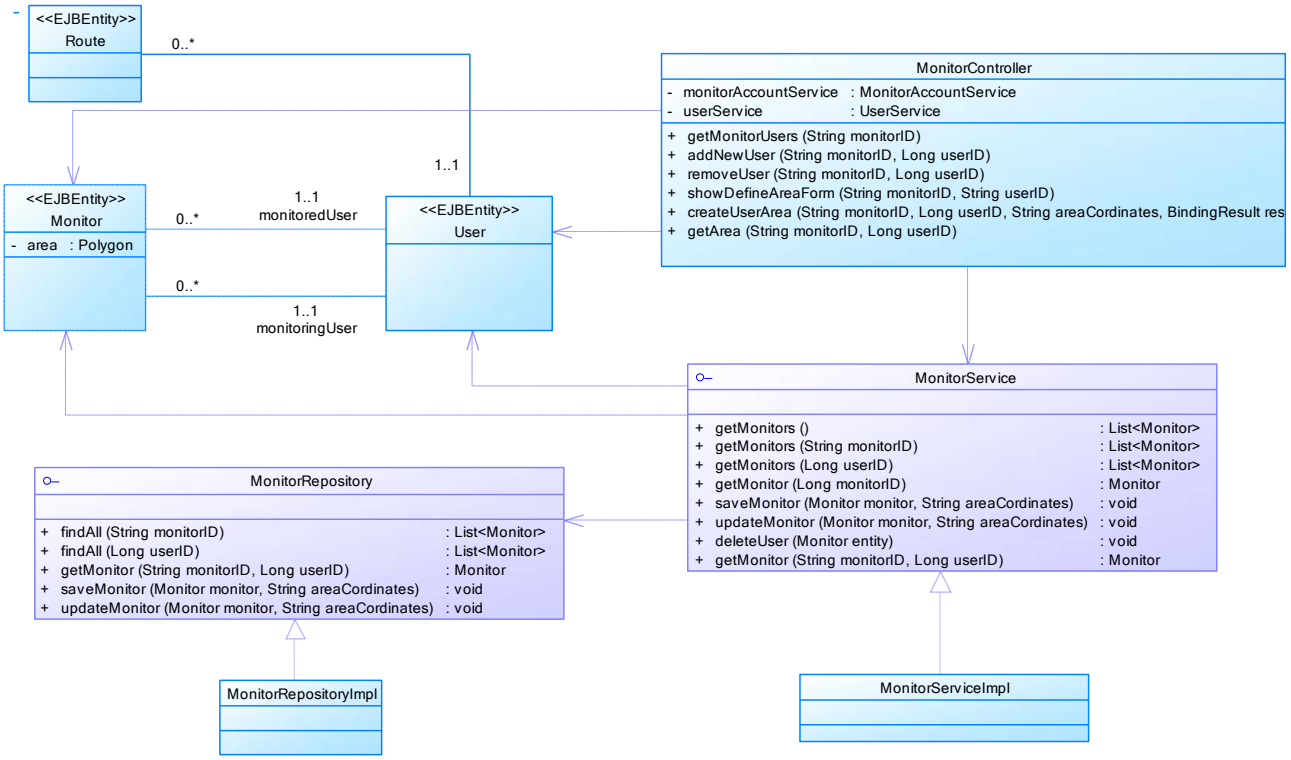

Figure 7. Monitoring part of the server module

5.1. ROUTE MANAGEMENT SUBSYSTEM

Figure 8 shows server classes used for managing routes. The *RouteController* class implements REST service that communicates with client applications (Android and web). The *RouteService class* defines methods for storing and retrieving users' routes. The *RouteServiceImpl* implements the *RouteService* interface. It relies on the appropriate *RouteRepository* implementation to store and retrieve data from the database. The *Route* class models user's route and the *User* class models system's user.

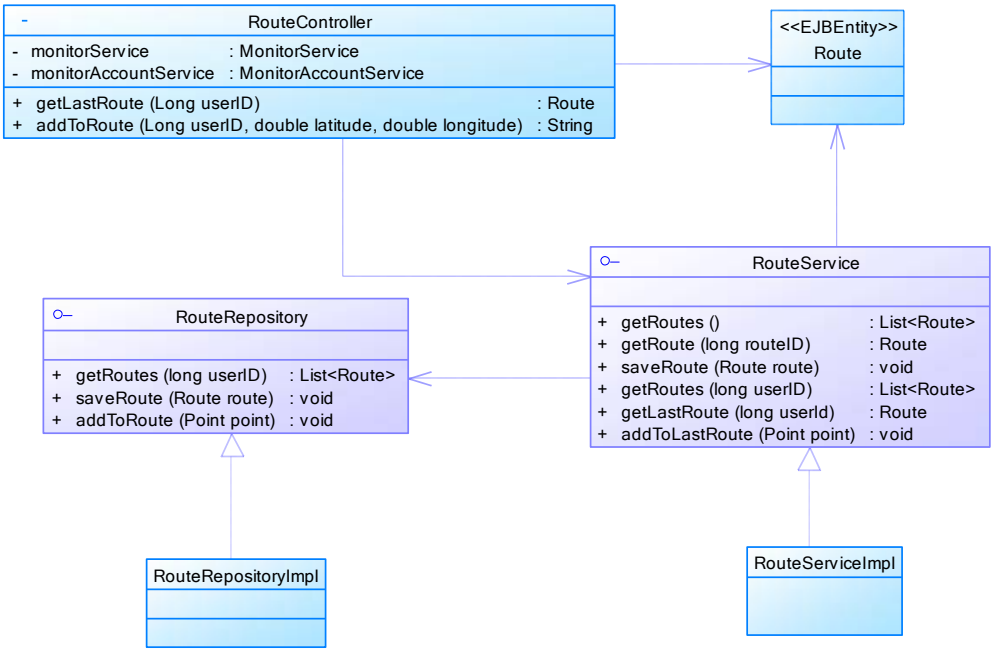

Figure 8. Route management part of the server module

5.1. NOTIFICATION SUBSYSTEM

As noted above, if the monitored user exits the defined zone, the monitoring user should be notified. Each monitoring user can define a different monitoring zone for each monitored users. Notifications are implemented using the GCM (Google Cloud Messaging) service [16]. GCM is a service which provides messaging from the server to Android applications.

Upon receiving user's coordinates, the server verifies if monitoring zones are defined for that user and checks if received coordinates are out of each zone. In the monitored user is out of the zone, a notification is sent to the monitoring user who defined that zone.

When the Android application receives a notification, it displays it to the user. It is possible to adjust parameters such as the sound of the notification, its icon, etc. Clicking on the notification opens the application and shows user's location.

6. CONCLUSION

Location Based Services provide personalized services to the subscribers based on their current location. LBS offer the tools for efficient management and continuous control. More and more people use LBS in their day to day life which helps then to more efficiently achieve their goals.

Massive usage of smartphones caused proliferation of LBS. First and foremost, it is used as a navigation device since most smartphones have built-in GPS receiver and other location methods. This paper presents a system for tracking users using their smartphones.

The system enables collection and storage of user's locations and monitoring of their movement by other users. The Android client periodically sends device's location to the server, which processes and stores it in the database. The monitoring user can see the route of the monitored user and receive a notification when the monitored user leaves monitoring area. Both Android and web clients are available to monitoring users. One of the useful applications of this system is locating and tracking children by their parents.

In the current version of the system we did not thoroughly analyze security and privacy issues. We plan to solve those problems in future versions. Also, we plan to investigate how other positioning methods can be used to provide indoor coverage.

REFERENCES

[1] Kumar, S., Qadeer, M. A., and Gupta, A. (2009), "Location Based Services Using Android", In *IMSAA'09: Proceedings of the 3rd IEEE International Conference on Internet Multimedia Services Architecture and Applications*, IEEE, pp. 335-339.

[2] Hirsch, F., Kemp, J. and Ilkka, J. (2006), "Mobile Web Services: Architecture and Implementation", *John Wiley & Sons*, ISBN 978-0-470-01596-4.

[3] Hazas, M., Scott, J., and Krumm, J. (2004), "Location-aware Computing Comes of Age". *Computer*, *37*(2), pp. 95-97.

[4] Rao, B., nad Minakakis, L. (2003), "Evolution of Mobile Location-based Services", *Communications of the ACM*, *46*(12), pp. 61-65.

[5] Ferraro, R., and Aktihanoglu, M., (2011), "Location-Aware Applications", *Manning,* ISBN 978-1-935182-33-7.

[6] Dhar, S., and Varshney, U. (2011), "Challenges and Business Models for Mobile Location-based Services and Advertising. *Communications of the ACM*, 54(5), 121-128.

[7] Junglas, I. A. and Watson, R. T. (2008), "Location-based Services", *Communication of the ACM,* 51(3), pp. 65-69.

[8] Rani, C. R., Kumar, A. P., Adarsh, D., Mohan, K. K., and Kiran, K. V. (2012), "Location Based Services in Android". *International Journal of Advances in Engineering & Technology,* 3(1), pp. 209-220.

[9] Mohapatra, D., and Suma, S. B. (2005), "Survey of Location based Wireless Services", In *Proceedings of the IEEE International Conference on Personal Wireless Communications, 2005, ICPWC 2005,* IEEE, pp. 358-362.

[10] Android, http://www.android.com/, accessed January, 05. 2014.

[11] Android developer, http://developer.android.com, accessed January, 05. 2014.

[12] Meier, R. (2012), "Professional Android 4 Application Development", *John Wiley & Sons*, ISBN 978-1-118-10227-5.

[13] Annuzzi, J., Darcey, L., and Conder, S. (2013), "Introduction to Android Application Development: Android Essentials (4th Edition) (Developer's Library)", *Addison Wesley*, ISBN 978-0-321-94026-1.

[14] Shu, X., Du, Z., & Chen, R. (2009), "Research on Mobile Location Service Design Based on Android", *Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09*, IEEE, pp. 1-4.

[15] Google Maps API, https://developers.google.com/maps/, accessed January, 05. 2014.

[16] Google Cloud Messaging, http://developer.android.com/google/gcm/index.html, accessed January, 05. 2014.