iCheck-Up: An Android Patient Monitoring Application

Joana Sofia Osório
Graphics and Multimedia Systems
1100594@isep.ipp.pt

Luís Guilherme Castelo
Graphics and Multimedia Systems
1091205@isep.ipp.pt

Mauro Fabiano Rocha
Graphics and Multimedia Systems
1101230@isep.ipp.pt

ABSTRACT

Remote assistance and monitoring helps caregivers to assist patient's daily living activities and it also promotes the functional independence of the patients.

This paper describes a patient monitoring system designated as iCheck-Up that was particularly designed for children and the elderly. iCheck-Up runs on all Android devices with a GPS, accelerometer and Wi-Fi communication capability. Our system allows caregivers and health professionals to remotely monitor the location and movement of patients. The data collected from the device is stored in a database server and used to detect real-time emergencies. If an emergency is suspected an alert is automatically triggered and the user is prompted to validate. If the user is unable to confirm the alert it is automatically validated after a defined period of time.

Additionally, we have also developed a web site that works as a communication channel between health professionals, caregivers and patients. This part of system allows caregivers and health professionals to view the history of patient's status, location and the current status and trigger an alarm in emergency situations.

iCheck-Up will not only play a crucial role in assisting the daily life of patients but also in contributing for the development of mobile health.

Keywords

Mobile, health, patient monitoring. GPS, accelerometer.

1. INTRODUCTION

The use of sensors as a medical tool for health care promotion and research is extremely important because it allows the remote monitoring of patients. Most mobile devices currently in the market are equipped with accelerometers to detect screen orientation and a GPS receptor to determine user location. However these two sensors can be repurposed to provide patient activity rate (accelerometer) and real-time location (GPS). The combined use of these two sensors can therefore be used to detect emergency situations, to provide feedback on mobility and even on daily habits and routines that may be important for health monitoring [1].

To promote the use of mobile devices for health purposes, we have developed a mobile monitoring patient system, called iCheck-Up. This application monitors the movements and position of the patient in real-time. It provides not only a remote monitoring system, but also real-time emergency detection capabilities that can help to accelerate emergency medical intervention. This might have positive outcomes on patient health following sudden accidents and life-threatening situations and it is particularly targeted for children and elderly people. The monitoring and interpretation of patient's movements and behavioural patterns using the data collected by the accelerometer and GPS will play a relevant role in the capacity to detect emergency situations. At the same time our system also has the advantage of providing information to caregivers and health professionals about their patients at home or on the move, using the designed website.

In this paper we present our prototype for an android patient monitoring application. We begin by comparing the most relevant previous related work, and then present system requirements and system design. Finally we summarize the main conclusions of this analysis.

2. RELATED WORK

Patient monitoring is essential in health care to improve the response time in emergency situations [2], therefore we found a large range of applications/systems in this area.

Lv *et al.*[3] developed an elderly monitoring system, called iCare. This system uses GPS technology to provide the location of the user and combines it accelerometer readings to evaluate the levels of activity. Additionally, this system also sends messages via GSM and calls ambulances to the location of the user in emergency situations. However, iCare [3] presents some limitations related to the analysis and presentation of patient information. This application provides the communication of medical knowledge via database server to facilitate medical guidance but does not show to the health professionals the complete medical history of the patient. To bypass this limitation in our system we have implementing a web site with all the information of the patients that safeguards patient privacy while providing complete medical information when needed.

In the area of fall monitoring and response, Sposaro and Tyson [4] developed an Android application called iFall. This application uses Smartphones with an integrated tri-axial accelerometer to track the occurrence of falls. Data received from accelerometer sensors is evaluated with algorithms and position data. To prevent false positives the fall notifications require user confirmation, but if the user does not respond for a long period of time, the system alerts emergency contacts via SMS. This application is similar to the real-time emergency system that we propose but unlike iCheck-Up the information of user motion monitoring is not stored in any database server.

iWander [5] is another example of an android application in the area of patient monitoring. This application is designed for dementia patients and uses GPS to guide them home and to send notifications to the caregiver containing their location. The main limitation of this system is that it does not detect the patients' abnormal activity consequently this system cannot give feedback to health care professionals.

With iCheck-Up we are intend to monitor patient activity and location. We also want to provide medical and family guidance through a web site that ensures the communication of health professionals and caregivers and patients.

3. REQUIREMENTS SPECIFICATION

The main goal of iCheck-Up is to optimize the process of collecting and sending patients' information in emergency situations. An agile system of diagnostic and reporting is very important in this context because the reaction rate and capacity responsibility are absolutely essential.

Health professionals can adjust treatments and act in accordance with the circumstances given that all patient information is centralized on a web page created for the purpose.

Below, we present the main functions of the system:

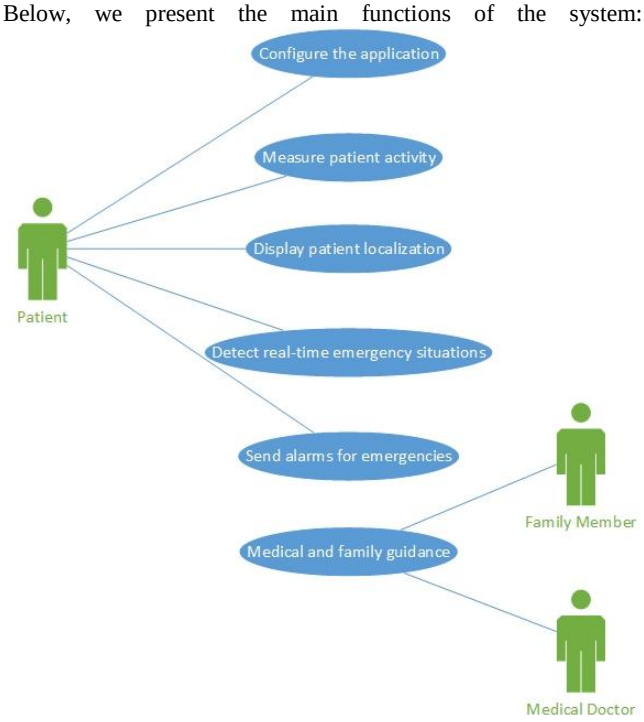


Figure 1 - Use Case Diagram

3.1 Configure the application

During the first run of the application the patient should configure it with his personal data and medical history.

The user should also provide the contact details of his clinical station and family members who will be notified in case of emergency.

3.2 Measure patient activity

The application has the ability to record the patient's physical activity (type of locomotion [walking or running], number of steps).

This way it is possible to know if the patient was engaged in an activity which required intense/sustained physical effort and which may have caused the emergency.

3.3 Display patient localization

When the patient uses the emergency button located on the main panel of the application, his location is automatically stored in the remote database and in the local **SQLite Database** [6] (for information backup in cases of momentary connection failure).

This way, both medical auxiliaries and family members can go to the patient's location and help the user.

This feature is also useful for predicting health and dietary habits (for example, restaurants which the patient visits, type of diet practiced).

3.4 Detect real-time emergency situations

iCheck-Up will detect some real-time emergency situations based on accelerometer data information (e.g., unusual movements). However iCheck-Up will only automatically trigger an emergency alarm upon user confirmation or after a long time without response. In addition of alarm emergency notifications, health professionals can detect these emergencies in real time on the web site.

3.5 Send alarms for emergencies

If the patient is feeling sick, he can call an ambulance by pressing a large button located in the main panel of the application or by executing a voice command (important in situations of reduced mobility).

The call will be automatically made and the loudspeaker activated. When an emergency is triggered the application sends a SMS [7] to the clinical station and the designed family member, besides calling the general emergency telephone number [8]. This way both health professionals and family members will know that they are facing an emergency situation.

3.6 Medical and family guidance

The application periodically sends the patient's data to the Database Server with the aim of updating his clinical status on the web page.

This way, both the family members and the healthcare professionals are aware of the patient's health status and can change the medicine if necessary, for example, ensuring dynamic monitoring.

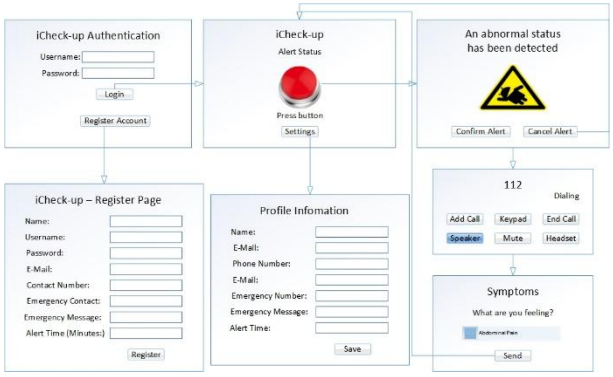


Figure 2 - Navigation Scheme

The main panel of the application has a large button for reporting emergency situations and a Settings button.

The Settings activity allows the user to update his personal information and emergency contacts.

In case of emergency the user is redirected to a screen designed to report his symptoms so that healthcare professionals can perform a pre-diagnostic. In addition our system sends a SMS and makes a call to the emergency number.

4. SYSTEM DESIGN

This section describes the design and architecture of the system iCheck-Up and is based on the requirements that were proposed.

Figure 3 shows our system's architecture.

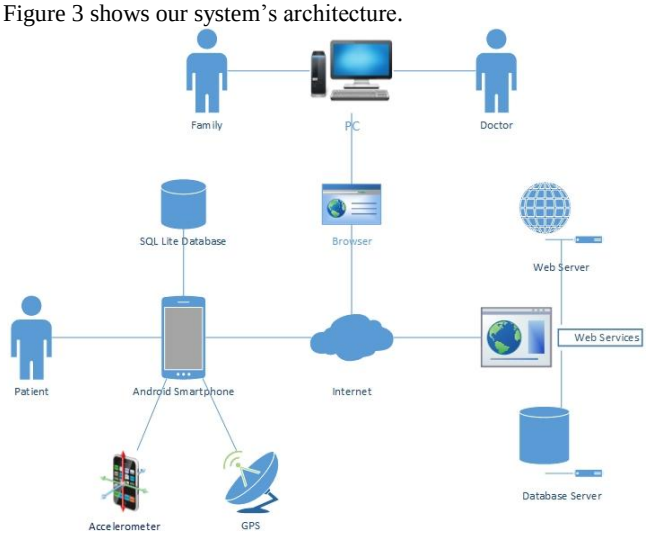


Figure 3 - System architecture

iCheck-Up can be divided into four components. The first one is the Smartphone which includes two integrated sensors [9]: accelerometer [10] and GPS [11]. The second component includes the web services that receive and send the information to the Smartphone. The third component is the database server. This component is responsible for storing all the information related to the patient. Lastly, the fourth component of our architecture is the web server which contains all the necessary files to represent patient information through web pages. Smartphone

4.1 Smartphone

The Smartphone must communicate with the GPS [11] and accelerometer [10] sensors in order to monitor the movements and position of the patient. The Smartphone will monitor the data acquired from sensors and automatically alert the emergency centre and the patient's family when an emergency is detected.

In case of failure of the GPS signal we will use a triangulation system utilizing GSM antennas [12] to deal with situations in which the patient is between buildings or in places where it is not possible to receive the GPS signal.

The Smartphone must submit the monitored information to the Web Services, if any wireless transmission is available the information persistence is achieved by using a remote **MySQL database** [6].

The local **SQL Lite database** [6] can be used if there is an internet connection failure to ensure that patient information is not lost, this guarantees some minimal services in case the web site does not respond.

The app is also able to perform an emergency request through a button click.

- Perform a call to emergency number;
- Send a notification of the emergency situation.

Figure 4 represents an outline deployment view of Android iCheck-Up application.

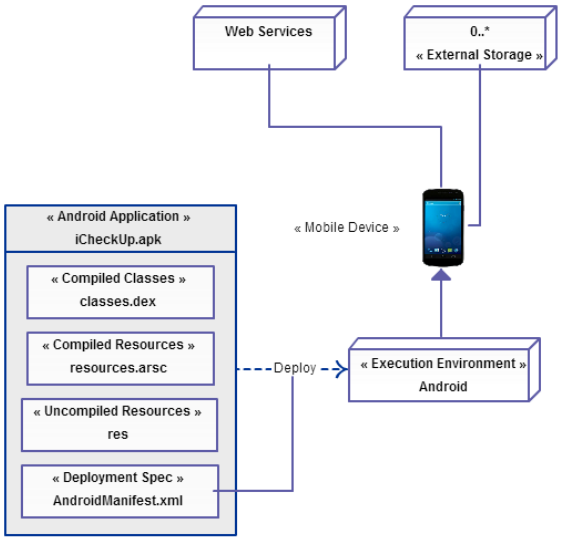


Figure 4 - Deployment View

Android applications are composed of several components (e.g., activities, services, content providers, and information persistence). Each of these components performs a different responsibility in the overall application behaviour.

The Android applications are developed in Java, so they use the Android SDK tools to compile and package code with resource files. The result of this compilation resultsin *.apk* files. iCheck-Up.apk is the Android application that need to be deployed to Android-enable mobile devices. Our Android application is composed of compiled classes, compiled resources, uncompiled resources and deployment specifications.

Particularly in what regards to deployment specifications, we highlight the manifest file (AndroidManifest.xml) that describes all application requirements (e.g., minimum version of Android required, hardware configurations, declaration of all components in the application).

With Android API Level 8 or later, some applications can be installed in external storage (e.g., SD cards). This external storage can also store external **SQL Lite databases** [6]. However, by default applications are installed in the internal storage and cannot be moved to external storage. In our particular case, we aim at using the external storage for the database.

Finally, we will use **RESTful Web services** [13] to receive and sent patient information. This component will be explained in the next section.

4.2 Web Services

The Web Services are responsible for ensuring the communication between the Smartphone and the servers. They must guarantee the connection to interact with the database in order to run applications for storage and release of patient information. Web services can be perceived as an intermediary, because they receive patient data, process and transmit it to the database server.

This component uses **RESTful** technology. In addition to making the communication with Smartphones easier, the **RESTful Web Services** [13] also guarantee the integrity and confidentiality of information. This type of Web Services allows the use of data in **XML** and **JSON** formats. **JSON** is a native format for this type of

technology. It will be used to make information processing easier. This type of data is very compact which makes its transmission faster which is useful because the 3G connections are very limited.

4.3 Database Server

The Database Server is responsible for performing the persistence and availability of information about the patient. The external database is also used as an information repository for the entire patient's information. Since some Smartphones have limitations in terms of memory and processor, it is important to store information in several ways.

This component interacts directly with the **Web Services** component.

4.4 Web Server

The Web Server contains all the files required to preview information through web pages. This information is intended to the family of the patient and doctors.

The website will allow certified users to query the following information:

- History of patient alert, status, location and symptoms;

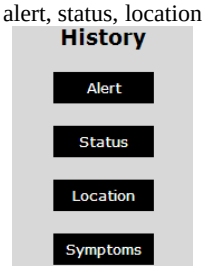


Figure 5 - History of patient

- Ability to monitor the current status of the patient;
- Location-tracking;

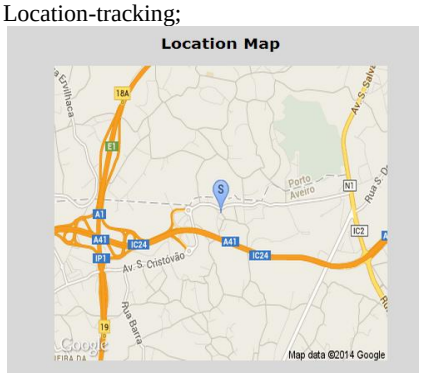
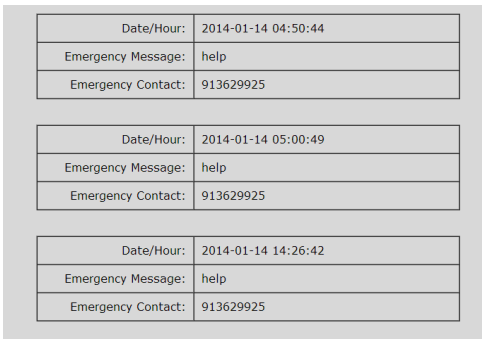


Figure 6 - Location Map of a patient in web site

- Alarm (detection of emergency situations in real time).



Date/Hour:	2014-01-14 04:50:44
Emergency Message:	help
Emergency Contact:	913629925

Date/Hour:	2014-01-14 05:00:49
Emergency Message:	help
Emergency Contact:	913629925

Date/Hour:	2014-01-14 14:26:42
Emergency Message:	help
Emergency Contact:	913629925

Figure 7 - Alert history of patient

5. PROTOTYPE IMPLEMENTATION

Our system iCheckUp acts like a mobile health monitoring system. We divided iCheck has been divided into five distinct parts: (1) login authentication; (2) main interface; (3) real-time monitoring; (4) configurations. In addition we also provide some services to monitor patient localization.

The application can only be execute on **Android 4.1.2** [14, p. 1] or higher and was developed in **Java** with **Eclipse IDE** and **Android Development Tools (ADT)**[15]. In what regards to design patterns we use **Active Record** [16] to facilitate the creation of business objects that require persistent storage to **SQLite database**[6].

Our web application on the server is developed in **PHP** [17] and use **REST PHP web services** [17] to communicate with iCheckUp android application.

How iCheckUp is a prototype in a first stage, we mainly focus on the main user interface and real-time monitoring.

5.1 Login authentication

Our application supports login authentication. Patients can perform the registration in our website by clicking "Register". Once registered, patients should authenticate with username and password. Patients can also view and edit their information in the profile option of our main interface.

Figure 8, represents our login interface on smartphone.

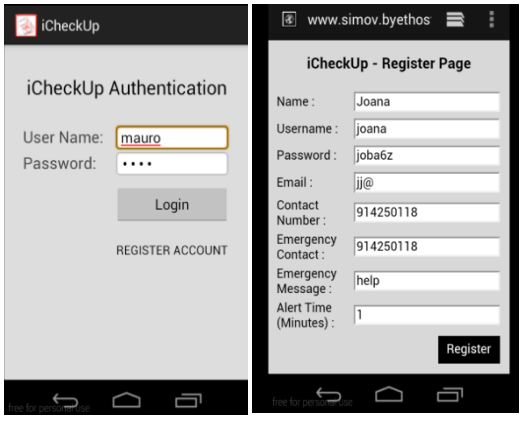


Figure 8 - Login interface on smartphone

5.2 Main interface

iCheckUp is designed as a real-time health monitor application but also provides other unique functions through our menu like "Personal info" with patient information, "Clinic state" with patient's current state, "Patient localization" with patient's

location and “Send symptoms” for the patients to submit their disease symptoms to our server system.

Figure 9, show main interface on an Android smartphone.

Figure 9 - Main interface on Smartphone

Our main interface has only one button that can be pressed in case of emergency to trigger emergency alarms. However when the patients enter this interface their status is continuously being monitored and in case of emergency it will also trigger an alarm. Our system considers a state of emergency when the patient is not walking and is inactive for a period of time that can be set in their profile.

5.3 Real-time monitoring

iCheckUp is mainly a mobile monitoring system, therefore it combines GPS and accelerometers sensors to provide and calculate the user current status.

In case of emergency our system triggers an alarm that requires user confirmation, but if the user does not respond for a long period of time it triggers the alarm automatically. This time period can be chosen in the configuration settings of the user's profile.

Figure 10 shows our emergency alert interface.

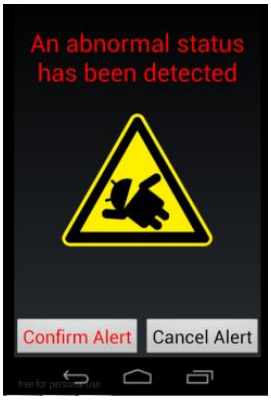


Figure 10 - Emergency alert

After the confirmation of emergency alert, our system will send a SMS and make a call to the patient emergency number. In addition it also shows to the patient a list of common symptoms that can be submitted to our server system to help health professionals in the monitoring of patients.

Figure 11, represents our symptoms list interface.

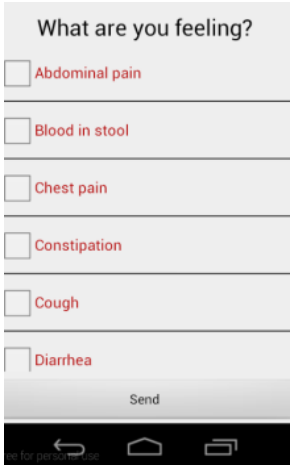


Figure 11 - Symptoms

Patient is also able to see their location when he clicks in the "location of the patient" button of menu. This option also synchronizes their location in the website.

Figure 12 shows the interface of the patient current location in their Smartphone.

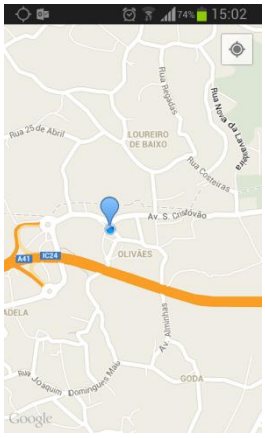


Figure 12 - Patient current location

5.4 Configurations

In the profile configurations patients can edit some of their profile information like: name, email, phone and some alarm configurations. This information is loaded from our web site with the user registration information.

In what regards to alarm configurations the user can choose an emergency number, an emergency message and an alarm time. The Emergency number and emergency message are used in the messages and calls, on the other hand the alarm time is used to detect patient inactivity.

Figure 13, shows our patient profile interface.

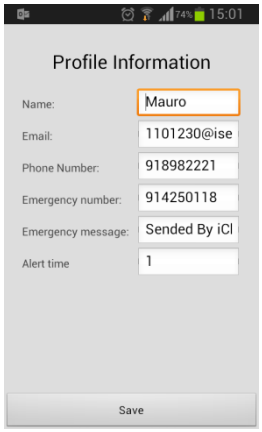


Figure 13 - Patient Profile Information

6. FUTURE WORK

This application is only in its first stage, therefore significant development is required to improve and implement it, particularly in what regards to user interface, our web application and the obtainment of the status and location of the patient.

As our targets are patients of different age groups, we have to take special care in developing the user interface which needs to be user-friendly and predictable.

We also need to guarantee the integrity and confidentiality of patient information, as well as creating an offline version of iCheck-up.

7. CONCLUSIONS

We designed a mobile patient monitoring system, called iCheck-Up, which pretends to be not only a remote monitoring system but also a real-time emergency system.

This system will basically control patient activity/location and trigger alarms for emergency situations. At the same time, this application will also be used as a personal medical information system which allows the family and health professionals to view the current status and condition of the patients by using a website.

With this analysis we conclude that iCheck-Up will not only play an essential role in assisting patients' everyday life but it will also contribute to the development of mobile health.

8. ACKNOWLEDGMENTS

Our thanks go to Professor Luís Lino Ferreira for all the help and dedication in the construction of this paper. This work was supported by Instituto Superior de Engenharia do Porto (ISEP).

9. REFERENCES

[1] K. G. Stanley and N. D. Osgood, "The Potential of Sensor-Based Monitoring as a Tool for Health Care, Health Promotion, and Research," *Ann. Fam. Med.*, vol. 9, no. 4, pp. 296–298, Jul. 2011.

[2] "Patient Monitoring." [Online]. Available: http://www.smiths-medical.com/products/patient-monitoring/. [Accessed: 28-Nov-2013].

[3] Z. Lv, F. Xia, G. Wu, L. Yao, and Z. Chen, "iCare: A Mobile Health Monitoring System for the Elderly," in *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, Washington, DC, USA, 2010, pp. 699–705.

[4] F. Sposaro and G. Tyson, "iFall: An android application for fall monitoring and response," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2009. EMBC 2009*, 2009, pp. 6119–6122.

[5] F. Sposaro, J. Danielson, and G. Tyson, "iWander: An Android application for dementia patients," in *2010 Annual nternational Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2010, pp. 3875–3878.

[6] "SQLite Home Page." [Online]. Available: http://www.sqlite.org/. [Accessed: 27-Nov-2013].

[7] "Sending a SMS Message from an Android Application - CodeProject." [Online]. Available: http://www.codeproject.com/Articles/463338/Sending-a-SMS-Message-from-an-Android-Application. [Accessed: 28-Nov-2013].

[8] "calls - How to prepare touchscreen-based Android device for [emergency] calling using only hardware buttons? - Android Enthusiasts Stack Exchange." [Online]. Available: http://android.stackexchange.com/questions/29924/how-to-prepare-touchscreen-based-android-device-for-emergency-calling-using-on. [Accessed: 28-Nov-2013].

[9] "SensorManager | Android Developers." [Online]. Available: http://developer.android.com/reference/android/hardware/SensorManager.html. [Accessed: 27-Nov-2013].

[10] "anddev.org • View topic - Accessing the Accelerometer." [Online]. Available: http://www.anddev.org/accessing_the_accelerometer-t499.html. [Accessed: 27-Nov-2013].

[11] "Integrating Google Maps in Android App | Tutorial | | ChupaMobile." [Online]. Available: http://www.chupamobile.com/tutorial/details/53/Integrating+Google+Maps+in+Android+App/. [Accessed: 27-Nov-2013].

[12] "GSM locator without GPS to a reasonable accuracy by triangulation." [Online]. Available: http://www.fact4ward.com/blog/not-done-yet/gsm-locator/. [Accessed: 30-Nov-2013].

[13] "RESTful Web services: The basics." [Online]. Available: http://www.ibm.com/developerworks/webservices/library/ws-restful/. [Accessed: 27-Nov-2013].

[14] "Android 4.1 APIs | Android Developers." [Online]. Available: http://developer.android.com/about/versions/android-4.1.html. [Accessed: 13-Jan-2014].

[15] "ADT Plugin | Android Developers." [Online]. Available: http://developer.android.com/tools/sdk/eclipse-adt.html. [Accessed: 13-Jan-2014].

[16] "ZuniSoft: ActiveRecord Data Model for Android." [Online]. Available: http://www.zunisoft.com/2009/09/activerecord-data-model-for-android.html. [Accessed: 09-Dec-2013].

[17] "[GUIDE] Android Client-Server Communication (PHP-MYSQL REST API) - xda-developers." [Online]. Available: http://forum.xda-developers.com/showthread.php?t=2325799. [Accessed: 13-Jan-2014].