FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Localization technology in special contexts: a collaborative anti-lost application

Ivo Rocha

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Prof. José Nuno Teixeira de Almeida

July 30, 2015

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES **2014/2015**

A Dissertação intitulada

"Localization Technology in Special Contexts: a collaborative anti-lost application"

foi aprovada em provas realizadas em 23-07-2015

o júri

Presidente Professor Doutor Eurico Manuel Elias Morais Carrapatoso
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Professor Doutor Salviano Filipe Silva Pinto Soares
Professor Auxiliar do Departamento de Engenharias da Escola de Ciências e Tecnologias da Universidade de Trás-os-Montes e Alto Douro

Professor Doutor José Nuno Teixeira de Almeida
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.

Autor - Ivo Jorge Magalhães Rocha

Resumo

Nesta dissertação é descrito o trabalho desenvolvido no último semestre do curso "Mestrado Integrado de Engenharia Electrotécnica e de Computadores", na FEUP. O principal objetivo deste trabalho é a conceção e desenvolvimento de um sistema que permita o seguimento, em tempo real, de crianças que se encontrem num centro comercial. Neste sistema, as crianças possuem uma pulseira que comunica com o telemóvel (que deverá ser Android e suportar *Bluetooth Smart Technology*) do seu pai, via bluetooth, o que permite averiguar se ela se encontra perto ou não. Possui, ainda, um sistema de alarme que será acionado, caso o telemóvel não encontre a pulseira pertencente ao seu filho.

A aplicação está programada para comunicar com o Posto de Informação do centro comercial, via Wi-Fi ou dados móveis. Este possuirá uma aplicação Java a ser executada, trocando mensagens periodicamente e transmitindo informação sempre atualizada.

O telemóvel utiliza a infraestrutura de pontos de acesso do centro comercial para transmitir a sua localização, bem como da lista de crianças que o rodeiam.

Quando o telemóvel efetua a busca e não encontra a criança que está "ligada" ao telemóvel, o alarme é ativado. Trata de comunicar com o "InfoPoint", que irá enviar a última localização conhecida e o intervalo de tempo desde que a informação relativa a esta pulseira foi atualizada.

Resumidamente, esta rede colaborativa, centralizada no Posto de Informação, é composta pelos seguintes elementos:

- Os telemóveis dos adultos;
- As pulseiras das crianças;
- O computador do Posto de Informação;
- A infraestrutura de pontos de acesso do centro comercial.

Esta dissertação descreve, não só como o sistema foi concebido e implementado, mas também os testes que foram desenvolvidos e os resultados que foram obtidos. Possíveis modificações e funcionalidades que poderiam ser acrescentadas são, também, descritas.

Palavras-chave: Android, *Bluetooth Smart Technology*, localização de crianças, Wi-Fi, Pontos de Acesso, BLE, Localização colaborativa.

Abstract

In this master thesis, it is described the work developed in the final semester of the course "Master in Electrical and Computers Engineering", at FEUP. The main objective of this work is the conception and the development of a system, enabling the real time tracking of children, inside a shopping centre. In this system, children have a bracelet that communicates with his/her parent's smartphone (Android-based and supporting Bluetooth Smart Technology) over bluetooth, which tests if the child is close or not. It has an alarm system, which is activated if the mobile phone does not find its owner's child's bracelet.

The application is programmed to communicate with shopping centre's Information Point over Wi-Fi or mobile data. It will be running a Java application, which periodically exchanges messages and always transmits updated information.

The smartphone uses the shopping centre's access point infrastructure to transmit its own location, as well as the list of children that are close to it.

When the mobile phone does not track its own child bracelet, the alarm is activated. It will communicate with the "InfoPoint", which will send this specific child's last known location, as well as the stretch of time since this bracelet information was updated.

In summary, this collaborative application/network, centralized in the Information Point, it is composed by the following elements:

- Adults' smartphones;
- Children's bracelets;
- Information Point's computer;
- Shopping centre's access point infrastructure.

This master thesis describes not only the system that was conceived and implemented, but also which tests were developed and which results were obtained. Possible modifications and new functionalities, that could be added, are also described.

Keywords: Android, Bluetooth Smart Technology, Children tracking, Wi-Fi, Access Points, BLE, Collaborative localization.

Acknowledgments

I have reserved this space to thank all people directly and indirectly involved, not only in this dissertation, but also in my five-year academic pathway, as well. It is important to highlight those who made their contribution to my success and who I would sincerely like to thank:

- My master thesis' supervisor, professor José Nuno Teixeira de Almeida, for all the ideas and corrections that he suggested during this semester. His expertise and motivational speeches were crucial to this dissertation;
- My brother, André, for all his patience and time that I took from him. He helped on testing the application and read this document several times, correcting linguistic mistakes;
- My girlfriend, Vânia, for suggesting layout modifications, not only in the application, but also on this document as well. Her ideas culminated in a more intuitive Android application;
- My parents, for all their comprehension and patience during this semester and for always motivating me whenever I needed;
- My friends and colleagues, mainly António and Pedro, for always helping me during these five years.

Ivo Rocha

“Learn from yesterday, live for today, hope for tomorrow.
The important thing is not to stop questioning.”

Albert Einstein

Table of Contents

List of Figures

List of Tables

Glossary

ADB	Android Debug Bridge
AES	Advanced Encryption Standard
AFH	Adaptive Frequency Hopping
AP	Access Point
BLE	Bluetooth Low Energy
BSSID	Basic Service Set Identifier
BST	Bluetooth Smart Technology
CSMA-CA	Carrier Sense Multiple Access - Collision Avoidance
GFSK	Gaussian Frequency Shift Keying
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
ISM	Industrial Scientific and Medical
JDBC	Java Database Connectivity
MAC	Media Access Control
PCB	Print Circuit Board
RFID	Radio Frequency Identification
RMIT	Royal Melbourne Institute of Technology
RSSI	Received Signal Strength Indication
SHA	Secure Hash Algorithm
SQL	Structured Query Language
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
URL	Uniform Resource Locator
WHMS	Wearable Sensor-Based Systems
WLAN	Wireless Local Area Network
WWW	World Wide Web

Chapter 1

Introduction

1.1 Context

This document, entitled "Dissertação", reflects the work undertaken in the final curricular unit of MIEEC (Integrated Masters in Electrical and Computers Engineering) course in the current year of 2014/2015. Inherent to the document, there is the characterization of the theme explored in the previous semester, addressing, now, not only the system design, but also its software implementation. Some tests were developed and its scenarios and results will be shown later on this master thesis.

An Android application was developed, and while it is running in an adult's smartphone, it gives the possibility to keep track of his/her child when he/she is in a space with a great number of people. This application is envisaged to work in closed but wide spaces, like a shopping center for example, where there are Wi-Fi access points which helps keeping track of children. The child will have a device in his/her wrist (named "bracelet") which will communicate with his/her parents' smartphone over Bluetooth.

1.2 Motivation

Children, due to their lack of experience and natural curiosity in exploring surroundings, tend to separate from their parents, and if there is an enormous amount of people in the evolving space, it will represent a major problem and concern to them.

The application that was designed represents a possibility to help parents doing their duties with less concern and it is also a way of quickly finding his/her child in the case of he/she leaves his/her vision sight. As security is extremely important, mainly if there are children involved, this system is important to overcome a problem that existed for a long period of time.

The technology that was used Bluetooth Low Energy (BLE) started to emerge in the marketplace, which represents a major motivation in approaching the subject. The amount of time that the devices' battery lasts, combined with the low amount of power that is consumed by

Bluetooth Smart Technology 's [1] modules, allows longer activity times, up to one year, compared to other Bluetooth devices.

Relatively to the application itself, Android programming is quite challenging, because the marketplace requires this kind of applications, inserted in the context of the Internet of Things [2] and, moreover, in the *Smart Cities*, where the sensor networks will be created. Furthermore, devices that communicate over BLE are quickly emerging in the marketplace too. Citing [3]: "*Bluetooth Smart is the leading candidate to connect the Internet of Things*". Concerning also the devices, Android as an operating system is increasingly being used by all of us and developing an android-based application was a motivating exercise.

1.3 Objectives

It is supposed that there are some people in the shopping center, whose smartphones are running this application, so that their mobile phones can create a sensor network, indispensable to the well-function of the system. They will not communicate with each other directly, but they all exchange information with an Information Point, localized at the shopping center, which corresponds to the network centralization.

The project's idea is to develop a network centralized in the Information Point (Infopoint), which helps a parent keeping track of his/her child and to enable other parents keeping track of their own children, by periodically sending status messages to the Infopoint. This last will receive those messages and it will proceed accordingly to the schematic mentioned later in this document.

Creating a real time system which receives information from each parent, keeping an updated list of each child's position and answering every alarm situation, is this project's main goal.

We would like to point out that the technology used in the communication between children and their parents is Bluetooth Low Energy. It was chosen not only due to its low power consumption characteristic, which represents a big advantage, but also because it is a very important emerging technology, like it was emphasized above.

1.4 Structure

Besides introduction, this dissertation contains 5 more chapters. In chapter 2 different localization technologies are described, and it will focus on the Bluetooth technology. Related work will also be referenced in this chapter. In the next one (chapter 3) is described the system to address the problem presented in this dissertation. In chapter 4 there is a general description and explanation of the software implementation, both Android and Java (Infopoint) applications. Then, in chapter 5 the different test scenarios are described and explained. The results will also be an important part of this chapter. Final, but not least, there is the conclusion and the future work around the application (chapter 6).

Chapter 2

Localization Technologies - Related Work

2.1 Literature Analysis

2.1.1 Systems using Bluetooth

In the article Ramlee et al. (2012) [4] Bluetooth Piconet Technology [5] is used to control several devices, available at home, and it is meant to be used by people with physical disabilities. The idea is based on the concept of a smarthouse and giving the possibility to control several devices such as the television and the cellphone. It is possible to control switches, which is very helpful to people who have a wheelchair, because they usually are 1.5 meters high, according to this article.

This technology supports up to 7 devices being used simultaneously. The author of this article used a Microchip microcontroler that enables controlling a major number of devices. In order to monitor which lamps are turned on, the application used Wi-Fi technology to connect to an IP Camera, which transmitted an updated information of the room.

A research to discover other devices on sale, that have a similar functionality (Bluetooth Smart Technology to track people and objects) was made. Bluetrack [6] is a good example. It is placed on a person's wrist or shoe and makes it possible to determine how far he/she is from the adult.

Linquet [7] is an application directed to locate objects. After placing a tag on the object we need to track, the smartphone runs an application that informs how far that object is. It can even be placed on pet's leash, to constantly track its position. Others examples, which use the bluetooth technology are XY findit [8] and Find'em Tracking [9].

2.1.2 Systems using other technologies

In the cientific article of Alghamdi et al. (2013) [10] it is presented a method to help deficient people moving in indoor spaces. The technology chosen in this project is RFID (Radio Frequency Identification). Using active tags located inside a building (in this article it was one of the floors at RMIT University - Australia) at strategic spots, allowed blind people moving securely across it. Each person would use an RFID reader on his belt, which would scan and read the nearest tags, according to the methodology presented at Alghamdi and van Schyndel (2012) [11]. The major advantage over GPS is that it can be successfully used inside buildings, since GPS signal, is quite ineffective at closed spaces.

It is also mentioned in this article the use of QR-Codes (Quick-Response Codes), which highlights some important rooms (like bathrooms for example). This technology needs a reader, which is represented by an application running at a smartphone that, with its camera, enables the reading of this codes. This mechanism could only be used by non-blind people. In addition, they had the chance to download the floor's map, through a QR-Code tag reading. They could choose the destination and the application would calculate the shortest path, through Dijsktra algorithm.

We would like to mention a set of simulations scenarios that were tested, at his master thesis work, by the student Jorge Neiva (2012) [12]. Resorting to the results of the simulation, he had the chance to conclude which spots were the most effective to place the tag's (passive) that aimed blind people moving through the "B" building, at Faculdade de Engenharia da Universidade do Porto (FEUP). Using Bluetooth communication between smartphone and the person's headset, a tag reader placed at the strategic part of a shoe would guide him through the building, following the algorithm results also.

There were some methods detailed at the article presented by Almeida and Pinheiro (2014) [13] that aim blind athletes to practise stadium athletics races, without leaving their tracks. The main goal of this idea was to dismiss the need for a guide and prevent athletes from running out of their tracks and colliding with other athletes. A study about the distance between active tags was formulated, in which there were 5 zones highlighted: Security zone, alarm zone 1 (right and left) and alarm zone 2 (left and right). The target of the idea was that the runner should always be in the safe zone. If he is out of it, he is alerted to correct his trajectory through a sonorous signal. In alarm zone 2 there was a stronger signal than when he is the alarm zone 1 because the risk of leaving the track was imminent.

An analysis and comparison between several WHMS (Wearable Sensor-Based Systems), which allow some health related scannings (electrodes, oximeters and acelerometers) is presented described at Pantelopoulos and Bourbakis (2010) [14]. Several technologies were explored to transmit the information retrieved by the sensors (Infrared and ZigBee). Emphasizing Zigbee, a low-power technology which operates in the same band as Bluetooth and Wi-Fi (2,45GHz, ISM-Industrial, Scientific and Medical band). An analysis of the article Suryadevara e Mukhopadhyay (2012) [15] was made to understand how Zigbee worked and how it can be applied. The target of this project is to monitor elder people's health. The sensors were placed in strategic spots, like beds or bathrooms, and kept information about which rooms each person has been to. Both articles were important to analyse if Zigbee was more indicated to this project than BLE.

2.1.3 Signal Degradation

Since Bluetooth and Wi-Fi work in the same frequency band, some collisions could occur between the two modules. After reading the paper from Lansford et al. (2001) [16], where this problematic is analysed, it is verified that the interference is real. Wi-Fi, specifically the norm 802.11b, needs about 22MHz of the frequency spectrum to exchange information with other Wi-Fi modules. Besides Bluetooth's 4.0 band being, instantaneously, only 1MHz wide, if it coincides with 22MHz wide band, an interference will occur and some packets might be lost. So Wi-Fi connection signal is degraded. Furthermore, it is possible to verify in the figure 2.1 that Bluetooth maximum transmission speed is reduced by 43%, approximately. The analysis to check if this fact would become a problem to the proposed solution, will be done later.

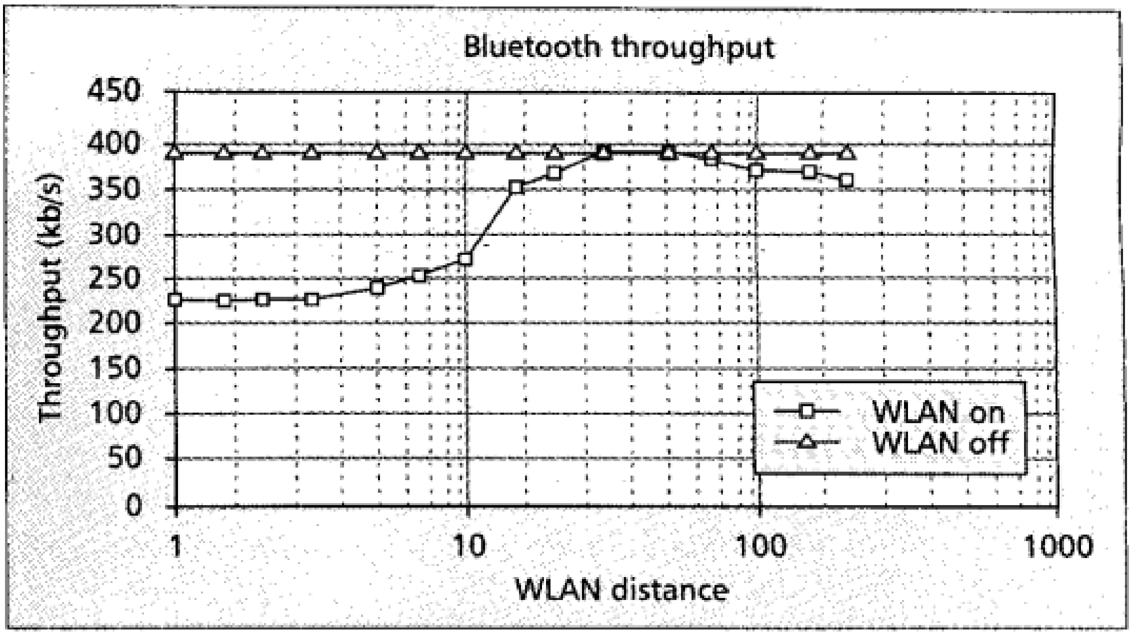


Figure 2.1: Bluetooth signal degradation in presence of WLAN, as distance (meters) rises

2.2 Bluetooth Technology

2.2.1 General Description

Bluetooth technology was used in this project. Moreover, Wi-Fi technology, described at the norm IEEE 802.11, was indispensable on the formation of the sensor networks. Developing an Android based-application, capable of combining both technologies and which is permanently tracking children was the biggest challenge of this dissertation.

The Bluetooth technology works at the ISM (Industrial Scientific and Medical) band, between 2.4 and 2.485 GHz [1], and it is a full-duplex communication, with a hop rate of 1600 jumps/s. This characteristic reduces interference with other devices. AFH (Adaptive Frequency Hopping)

[17] is a technique which analyses the spectrum and finds out which frequencies are free, reducing the number of collisions between hops.

This asynchronous mean of communication has a maximum information exchange rate of 1 MB/s and uses GFSK (*Gaussian Frequency Shift Keying*) modulation. Moreover, data can be encrypted using 128 AES (Advanced Encryption Standard) technique.

To measure how strong the signal is, one can use the RSSI (Received Signal Strength Indication) value. It is a numeric value, defined by the IEEE 802.11 standard [18], ranging from 0 to minus 255 and indicating the measured RF (Radio Frequency) energy.

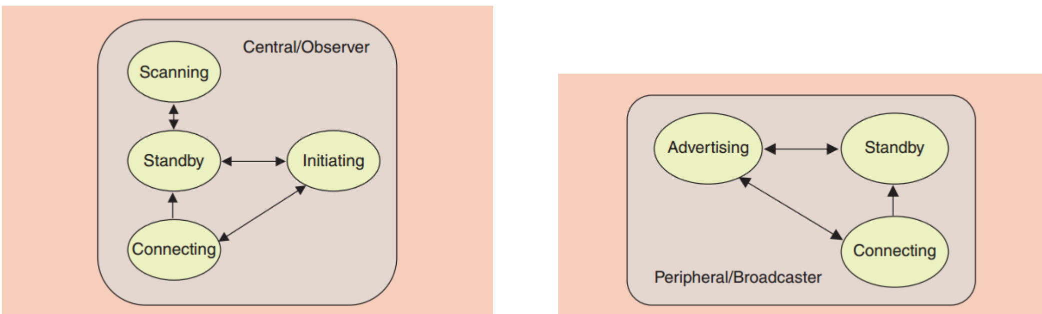
Like it was mentioned on the article [19], the network "nucleous", which is represented by the smartphone, can be in one of the four following states:

- Scanning - Constant searching for bluetooth devices;
- Standby - Amount of time, at low power mode, when there is no information exchanged between devices (stop Scanning);
- Initiating - Application initialization mode;
- Connecting - Represents a successful scan, (e.g., when the bracelet is found).

On the other hand, the sensor (the bracelet in this project) can only be in one of these three states:

- Standby - Amount of time, at low power mode, when there is no information exchanged with the smartphone;
- Advertising - Active mode, when smartphone can locate the bracelet;
- Connecting - Represents a successful scan, when the bracelet is found.

The figure 2.2 shows how this project's state machine works:



(a) Smartphone (b) Bracelet

Figure 2.2: System devices states

BLE technological advance has a major impact in medicine, with a great number of devices that communicate over this technology [20]. There is an increasing number of applications being developed in Sports and Fitness areas [21] too. The number of Bluetooth Smart Technology devices and its applications are somehow specific, but this technology allows, contrarily to Bluetooth specified in IEEE 802.15 norm, accessing more devices (like it was discussed before, the latter only allows up to 7 devices) [22].

To measure how far devices are, each one has a Bluetooth system which will pair with the other and the distance between them will be measured by the power emitted by the bluetooth antennas. Depending on system's bluetooth class, the pairing which enables communication, can occur.

There are three different classes at Bluetooth Smart Technology , in which "range" varies in both of them. At this specific project, class 2 was used (since Bluetooth devices that were tested can not be more than 10 meters apart from the smartphone) [23].

Comparing Bluetooth Smart Technology to other wireless technologies, like ZigBee for example, the first was preferred because almost all mobile phones which are on sale support this technology (Smart Ready). Furthermore, the power consumption is lower [3]. It is even lower than ANT technology, another wireless mean of communication [24], which it is estimated that, in standby mode, consumes $60\mu A$, 60 times bigger than BLE current consumption. Both ZigBee and ANT operate in the Industrial Scientific and Medical band as well. As possible advantage, ZigBee can reach 100m in some applications [25], without using an external antenna. As disadvantage, this technology is not native on the great majority of smartphones.

To create a low power sensor network, the developer must pay attention to several parameters, like node's hardware and network protocols [26]. BLE allows a low power communication, which saves devices' batteries and, together with Wi-Fi, will aid in the establishment a network capable of helping parents keeping track of their children.

2.2.2 History

Bluetooth appeared in the marketplace near two decades ago and it is estimated that it was available in almost 90% of the mobile phones, at the year of 2014 [27]. In the year of 2001 it arise with the first printers, hence enabling the communication with computers through this technology (only laptops had a bluetooth antenna). Headsets and hand-free devices, which are connected to the car lighter, also caused a big development on this technology.

BLE specifically, also named Bluetooth Smart Technology , is available on Android operating system, since it was updated to version 4.3 (Jelly-Bean) [28] (these devices also need an antenna which supports BLE) and it is available on iOS devices since iPhone 4S, which was the first cellphone capable of using this technology, in 2011 [27]. About Windows Phone devices, they can communicate with BLE devices since the 8th version of this operating system [29].

Since Bluetooth technology was developed, it has a great number of applications. For example headsets [30], which use Bluetooth to pair to the cellphone, and hand-free kits [31] which are placed in car's dashboard, give drivers the opportunity to answer and to make phone calls without placing the mobile phone next to the ear (figure 2.3). It was also useful to exchange data between cellphones, like pictures and videos for example, widely used by people in the last decade.

(a) Motorola headset (b) Parrot car kit

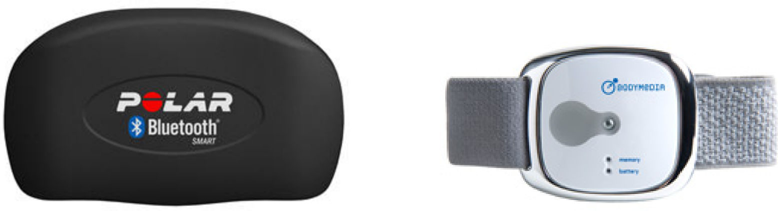
Figure 2.3: Bluetooth ready devices

The fields and applications envisaged by Bluetooth Smart Technology are quite different. At the medical area for example, there is a great number of devices that communicate over this technology (figure 2.4).

Even in sports and fitness exercises this technology is constantly being used. Heart Monitors, which track heart rate while people are practising exercise and sends the information to a mobile phone, or body monitors, which track daily habits and exercises, giving the chance to change some routines, and then workouts can be adjusted next time people do exercise (figure 2.5).

(a) Stethoscope (b) Glucose Meter

Figure 2.4: Medical BLE devices



(a) Heart Monitor (b) Body Monitor

Figure 2.5: Sports ready BLE devices

Chapter 3

System Design

3.1 Problem Characterization

This project consists in applying Bluetooth and WLAN technologies to progressively track children position, using an Android based smartphone. The use of this system is confined to closed spaces and it needs an access point infrastructure, as well as a computer running a specific software, placed at the shopping center Information Point (for example), which will communicate directly with the mobile phones. The latter will retrieve information from the access points to indicate their localization.

The system was developed so that it can be used in wide but closed spaces, like a shopping center for example. Nowadays, they possess internet connection, with strong Wi-Fi signal in almost every floor. This infrastructure will enable a successful exchange of information, and the centralization of this network corresponds to the computer at the Information Point. It is also foreseen that an adult, which has a child in charge, has a 4.3 or over Android smartphone, with specific hardware that supports BLE, since this update accepts Bluetooth Smart Technology devices (with the old norm, these devices are not scanned by this application). All Android phones have a WLAN antenna, so it is not a concern in the project. The child will have a device, with a bracelet-like format, which will emit a signal that can be captured by his/her parent smartphone that indicates if the child is next to him/her or not.

3.2 System Components

In the chaper 2 there were presented some devices which could track people and objects using Bluetooth technology. Combining this methodology with Wi-Fi communication, which will facilitate the establishment of a network using parents cellphones, will be important to track a child inside the shopping center.

In the figure 3.1 it is outlined the architecture of the solution. The access points drawn are part of shopping center infrastructure and they help the Information Point (i) to track the parents'

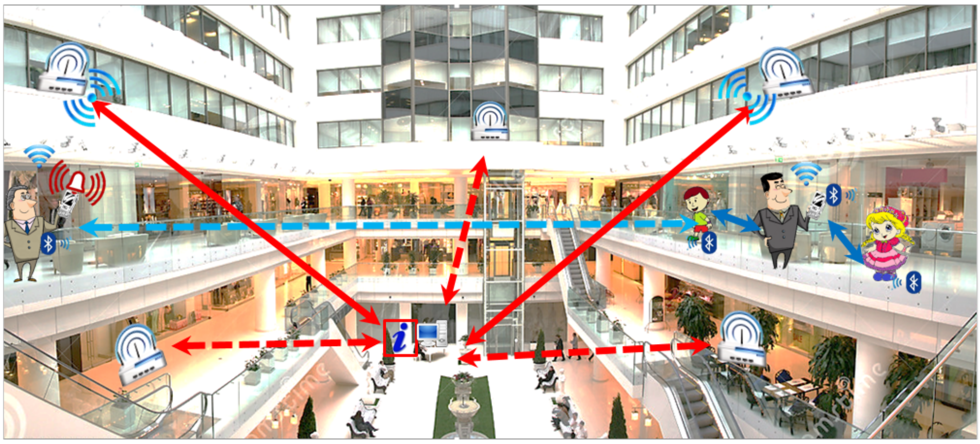


Figure 3.1: System Architecture

cellphones. The bracelets on children wrists emit a signal which helps parents understand if they are under their vision sight.

As shown in the figure, the girl, on the right side, is relatively close to her father, so her bracelet can communicate with her parent's smartphone (blue line). Oppositely, the boy's bracelet signal could not be captured by his father mobile phone, because he is too far away (blue dashed line). The mobile phone will sound an alarm that will advise him.

Over the access point infrastructure, the girl's father (which also captures the signal emitted by the boy's bracelet) will send a message to the Information Point, containing a list of all Bluetooth Smart Technology devices found by his smartphone (in this specific the bracelet identification of his daughter and the other boy). On the other hand, the boy's father will inform the Information Point that he can not track his son's position. In response, he will receive a message containing the localization of the access point nearest to the girl's father (since it was this smartphone which detected the boy's bracelet).

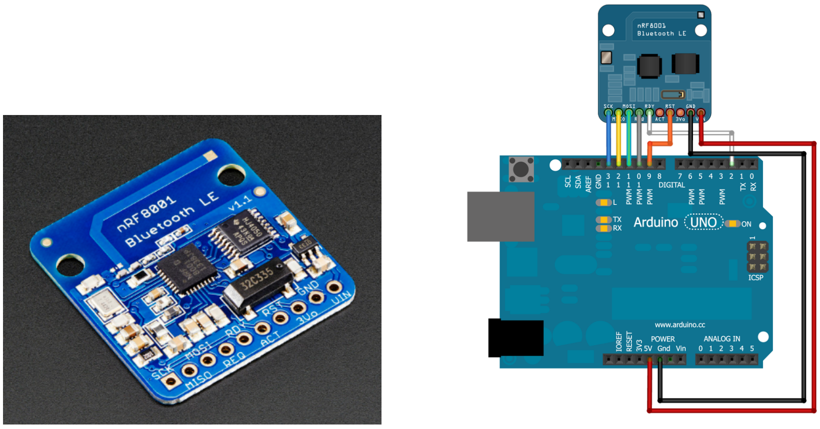
The devices that form the network and its specifications will be presented next.

3.2.1 Child Bracelet

The child will have a bracelet on his/her wrist, containing a Bluetooth Smart Technology module that directly communicates with his/her father's cellphone. The maximum distance that separates father and son is the maximum range of this technology (approximately 10 meters on class 2 devices). Each bracelet has a unique ID that works as an identification card, which associates the kid to his/her father exclusively.

To guarantee that the child does not remove it, a mechanical security system was thought (but not implemented in the prototype) similar to a lock, which will hamper its removal, unless the proper key is used. The bracelet format would be adjustable, in a way that it could be standardized.

The power supply will be a watch-like battery, so that the BLE module could work properly. It would have an user-friendly shape, so it is appealing to children and not obstacle to its



(a) nRF8001 Adafruit module (b) Module to Arduino connection

Figure 3.2: Bracelet Prototype - nRF8001 module from Adafruit

function. Since there is no bracelet on sale with these characteristics, the system was tested with two different devices. As shown in figure 3.2, firstly using a nRF8001 module [32] from Adafruit [33], connected to an Arduino Uno microcontroller [34], which only supplies voltage to this module. Secondly, as shown in figure 3.3, the system was tested using a CC2540DK Texas Instruments development kit which does not need to be connected to any microcontroller, since it is self-powered.

Figure 3.3: Bracelet Prototype - CC2540 module from Texas Instruments

3.2.2 Parent's Mobile Phone

The smartphone must have an antenna that supports BLE. The system was developed for Android mobile phones, which must have version 4.3 or later. The main routine of this application scans for Bluetooth Smart Technology devices and verifies if his/her child's bracelet ID was found. If it does not find this bracelet ID, an alarm will sound to warn the father.

Besides this verification, it will keep an updated list of all BLE devices found by it. This list is sent, then, over Wi-Fi, to the Information Point.

Besides this method, a parents' virtual network is created. If the Information Point knows the localization of the access point and which parent sent the information, it can easily find out which children are next to a specific parent.

The application is capable of running in background mode, so that the owner of the cellphone can use it normally, to make a phone call, for example. One of the main challenges is to find out how to keep battery from lasting only a couple of hours, since using Bluetooth scan routine spends a great amount of battery.

If a specific father loses track of his/her child, his/her mobile phone will inform the Information Point. Then, after some procedures and verifications (detailed later in this document), the InfoPoint will tell this person where his child was last tracked.

3.2.3 Network Centralization

As it was referred before, the use of access points is indispensable. At the Information Point, a Java-based program, that receives messages from the mobile phones and answers accordingly, is running. If this application has a complete knowledge about every access point, it can inform parents, in real time, about where his/her child was last located, using access point's BSSID (Basic Service Set Identifier). This value is a 48-bit field, of the same format as an IEEE 802 MAC address, and it uniquely identifies each Basic Service Set [35].

Besides these background procedures, this application is ready to display which parents are currently running the Android application, at the shopping center.

3.2.4 Security

Anti-theft system used in the bracelet was already mentioned before. Moreover, it is important to guarantee children's privacy. So, his/her bracelet ID would be encrypted and then send to his/her parent and to the other members of the network. Only this child's parent and the Information Point know how to decrypt each ID. The other members of the network would send each bracelet ID encrypted. It is important to highlight that this encryption methodology was not implemented in the prototype.

How is it possible to guarantee security and privacy? To be part of this network, each parent needs to be registered in the shopping center database, so that the Java application can associate each bracelet to the correct parent. The information about each profile is only transparent to the Information Point.

This registration is a secure way to prevent third people to use the application in personal benefit. The application can only be used after login, so that each time a person sends a message to the Information Point, his identification is included. On an extreme scenario, if someone tries to kidnap a child, exploring this the characteristics of this system, his personal identification and the child bracelet ID will be interconnected for a long period of time and it is possible to identify the kidnapper very quickly, since the bracelet can not be easily removed.

Chapter 4

Software Implementation

4.1 General Description

After the conception of the system, designed to aid parents in keeping track of their children, two main applications were developed, one to be run on smartphones (client side) and the other on a central computer (server side). A brief explanation of this system characteristics, as well as its implementation, is described through this chapter.

The system is formed by a group of parents, their children and the Information Point, as it is described on figure 3.1. The access points located in the shopping center will be a major help on finding a child, if missing.

Each parent is supposed to have an Android phone, which will be running an application that is constantly scanning for Bluetooth Low Energy devices. This application, after having the list of found devices, will send status messages to the Information Point periodically. These status messages, besides the list of found BLE devices/bracelets, include the ID of the nearest access point of the shopping center. Hence, from this ID, it is possible to derive an approximate localization of respective BLE devices.

In figure 4.1 it is represented how the system operates. Each client is connected to his bracelet, over bluetooth, and reports his state to the server, including the information about the nearest access point.

Relatively to the transmission of the status messages, this can be independent of the shopping center wireless network. In fact, a parent can turn mobile internet data on and it will communicate with the Information Point over it. This application will be described in the Client Side section.

Other considerations are included in the last section. It is mainly focused on the database, where each parent's information is stored. A login process is required to access the application, in order to guarantee a transparent identification of every parent, who communicate with the Information Point.

The computer at the shopping center's Information Point will be running a Java-based program, which will receive every message from each parent and will answer accordingly. Besides these background procedures, it will show, in real time, which parents are online in the

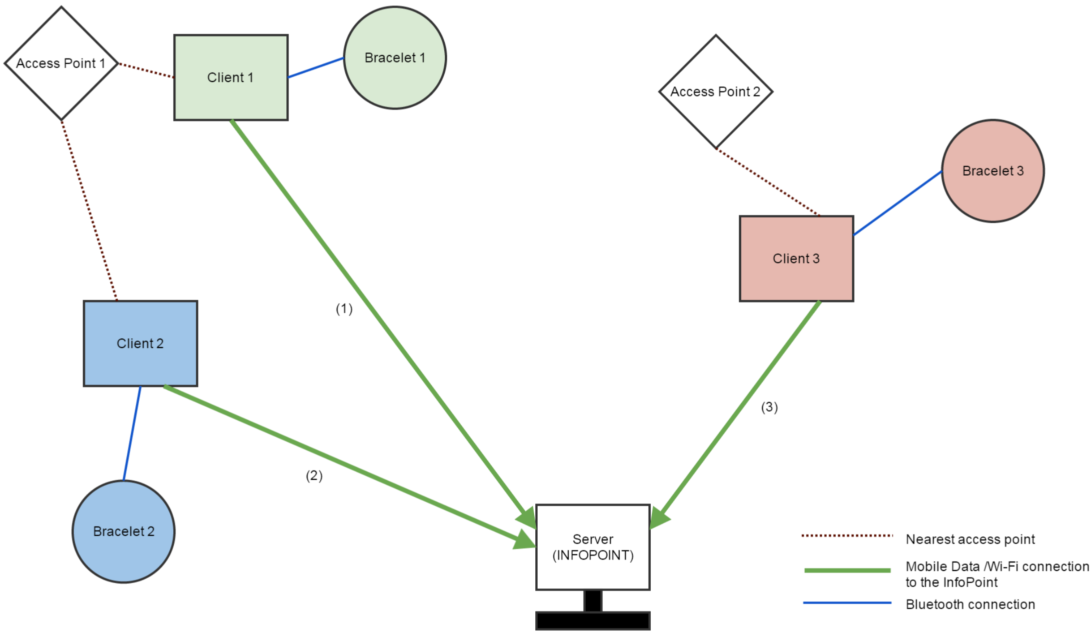


Figure 4.1: System Operation Diagram

network. Since all information crosses this application, it is considered the network centralization. This program has a complete knowledge of every access point's position, so that it can inform each parent correctly.

If this system is implemented at a real shopping center, the Java program would save the localization of each access point and which store is nearby. It is important to highlight that the GPS antenna is never used in this project. The Server Side section will illustrate this program's functionalities.

4.2 Client Side

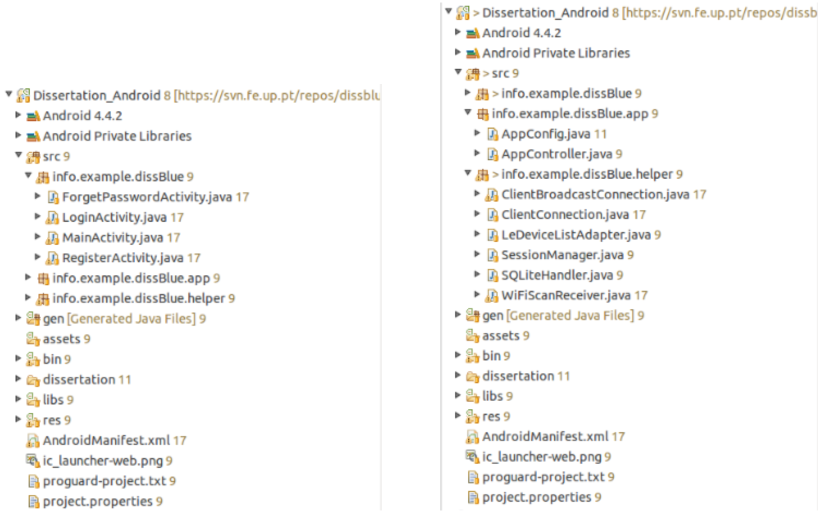
The main routines of the application to run in the client side (entitled "DissBlue" in this work), are the following:

- Scanning for Bluetooth Smart Technology devices;
- Sending status messages to the Information Point.

The sources files, which are compiled and executed in this application, are summarized in the figure 4.2. This project's application is based in other Android application, which is described on the "Android Hive" website [36].

There are four different activities in this application (Project Main Package):

- Login Activity - User can login in the application;



(a) Project Main Package (b) Project Auxiliar Packages

Figure 4.2: Android Project

- Register Activity - User can store his data in the database;
- Forget Password Activity - Restore user's password;
- Main Activity - All routines are processed.

As Project Auxiliar Packages, there are other two packages ("app" and "helper") which contain auxiliary files used by the activities mentioned above. The "app" package includes two files, which allows to connect to the file in FEUP's server [37], which is responsible to connect to the database. The other one - "helper" package - has some functions that allow the exchange of information with the Information Point, Bluetooth Smart Technology list file, store user data locally in the smartphone, save the user's session and detail access point's characteristics.

The remaining folders include, not only the activities layouts, but also some files which are automatically generated by the compiler. This application is defined so that it can run in the 4.4.2 Android version (since it is the smartphone's tester Android version) but it can run on an older version - Android 4.3 at least (API 18) and a newer version - Android 5.0 (Lollipop - API 23). The Android Manifest file, in the project's root explorer, is extremely important and it will detailed later.

The flowchart presented next illustrates the correct path to access the main activity - main process.

Each activity's procedures described in the figure 4.3 will be detailed in the following subsections.

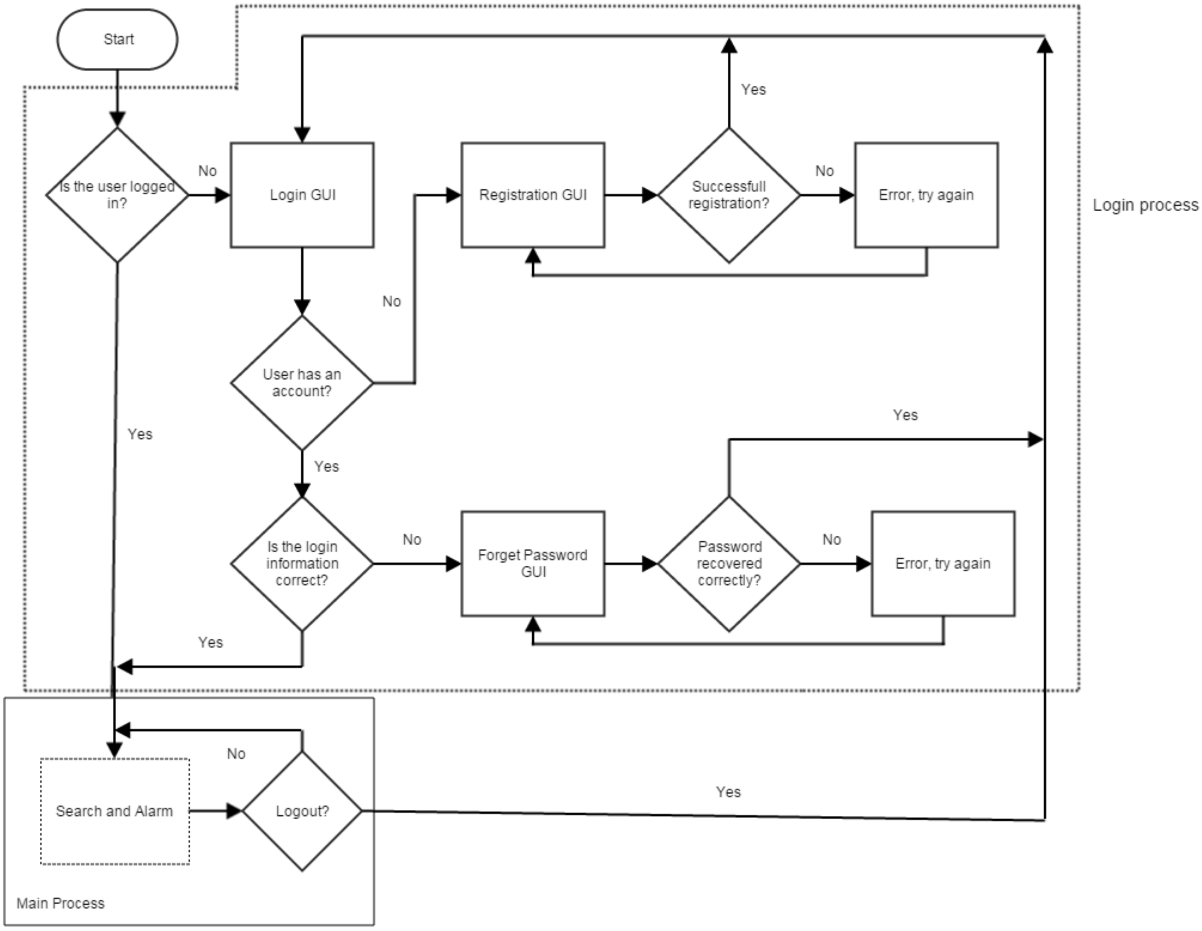


Figure 4.3: Login flowchart

4.2.1 Installation Guide

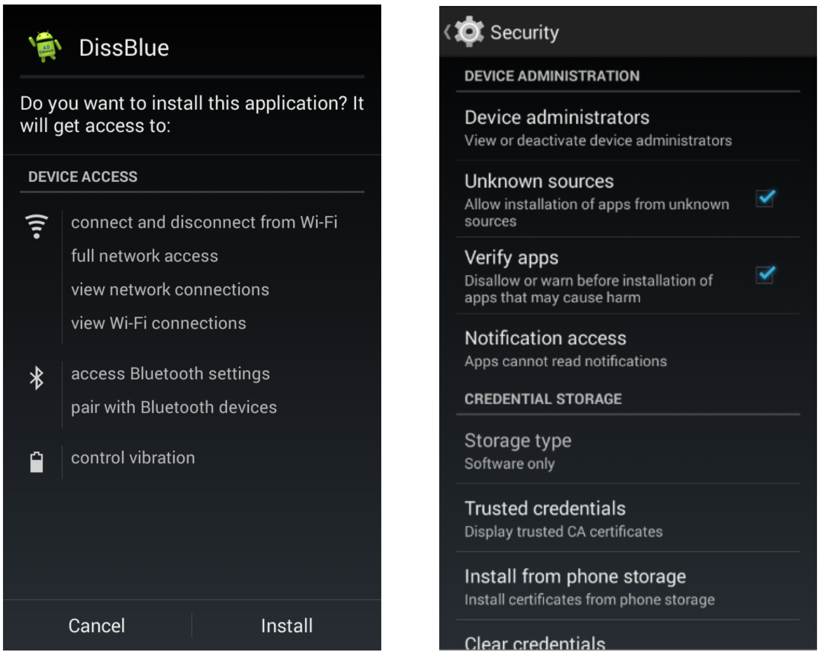
Before detailing the activity itself, it is presented how the application can be installed. Firstly, people need to download the application from the website [38] and then open it. An image, as shown in the figure 4.4 (a), should prompt in the mobile phone. The permissions, which this application asks for, are detailed later in this section.

If the message "Installation blocked: For security, your phone is set to block installation of apps obtained from unknown sources." appears, a permission to install unknown sources must be enabled. Click on *Definitions Menu*, than *Security* and enable the option "Unknown sources" (figure 4.4 (b)). Thus, try to reinstall the application.

After a successful installation, login activity is prompted, when we start the application.

4.2.2 Login Activity

This activity is prompted when the application starts running (figure 4.5). It requests permission to turn Bluetooth on (indispensable on main activity) and turns Wi-Fi on



(a) Install Application (b) Enable unknown sources

Figure 4.4: Installation Process

automatically: apart from turning it on, it does not connect to an access point right away. It is useful to scan for access points and determine which one is closer.

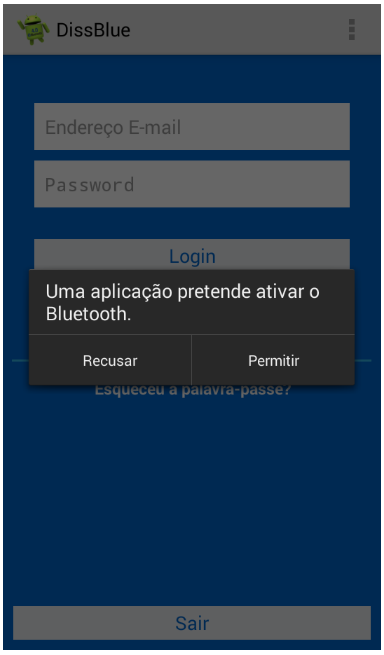
If the user has created an account before, he only needs to enter his credentials and if they are correct, the main activity is prompted. The validation of the set "email address, password" is done by an external library.

The "volley" library [39] communicates with a file that is stored at the server (in this project the file was created on Faculdade de Engenharia da Universidade do Porto 's server at [37]). This file receives the request from the application and connects to the database. For this specific activity, it has two procedures:

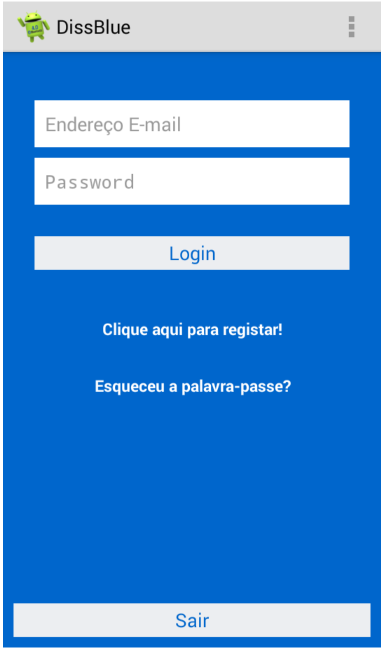
- Compares the tag sent by the application, and if it is "login", it calls the function "getUserByEmail";

- It executes a query which selects the user by his email (which will return an error message if the email does not exist), encrypts the password sent by the application and compares with the password stored in the database.

The password is encrypted with the SHA256 cipher, which needs a salt code that is randomly generated and stored at the database, as well. This guarantees that the password is not stored as plain text and makes it very difficult to guess, since the attacker needs not only the password's ciphertext but also the salt code.



(a) Permissions Requested



(b) Login Layout

Figure 4.5: Login Menu

This functionality, besides doing these verifications, returns some information about the user - his username, his child's bracelet ID and the date he has registered. If the password is correct, the login activity is finished and the main activity is prompted. After this, each time an user runs the application, he does not need to login, because his session is stored.

One person can login in different mobile phones. In addition, the same smartphone can accept different users to login (one at a time), because the profiles are stored in the database and they can be accessed by any smartphone.

Furthermore, the user can register in the application (if he does not have an account), recover his password and exit the application - where a dialogue box is shown to confirm if he really wants to close the application (figure 4.6).

Register and forget password buttons, if pressed, finish the login activity and start its respective activity.

4.2.3 Register Activity

One must press the register button, from login menu, to prompt the register activity.

A person should, then, enter his data: name, email address, password and then confirm password (see figure 4.7). The bracelet ID field can be filled in two different ways. One can write his child bracelet ID by hand or press the bracelet identification button. This routine scans for Bluetooth Smart Technology devices and fills that field. It is important to underline that the registration process should be done before going to the shopping center, so that the scan routine

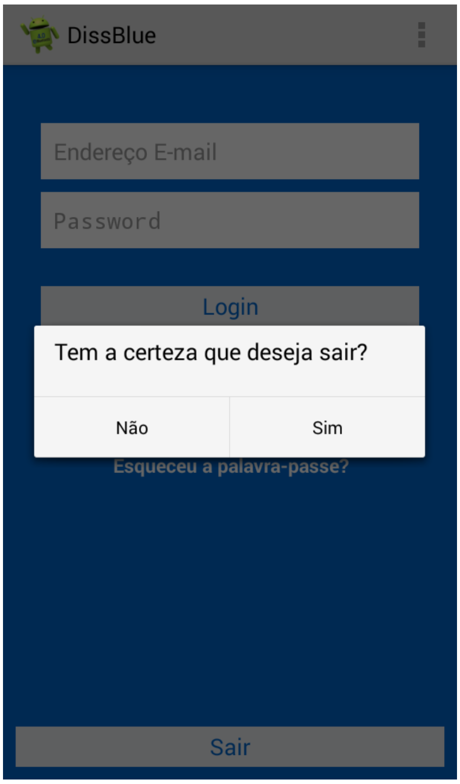


Figure 4.6: Exit application

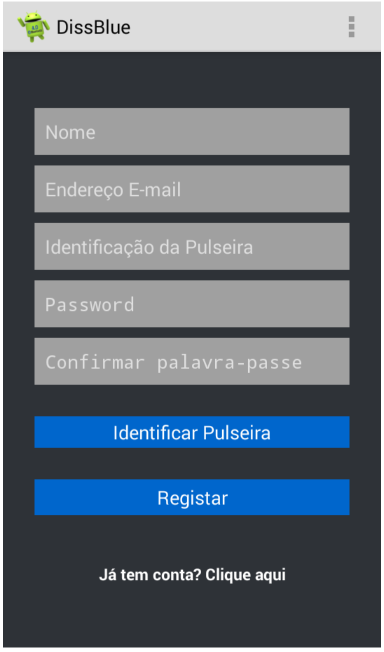
finds his child's bracelet only. It is also crucial to mention that, for now, the application is ready to associate one parent to one child only (1:1 correspondence). In future work, it should be possible to associate one parent to two or more children (1:N correspondence).

The routine that identifies the bracelet ID after a scan is important, not only to the registration process, but also to acknowledge if the mobile phone, where a person is trying to register his profile, has BLE antenna. In the case where this feature is unavailable, the routine will not return any ID.

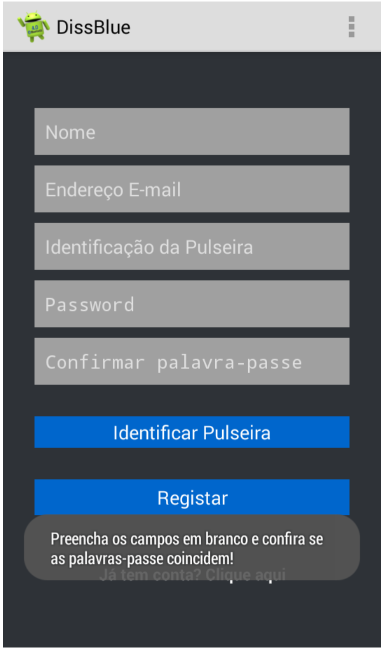
Email address field, as well as the bracelet ID, are unique. In other words, if the application detects that the email address or the bracelet ID are already registered in the database, one can not use that information. The uniqueness of the bracelet ID field prevents that someone registers as this child's father (figure 4.7 (b)).

This process is similar to login. It also connects to the server [37] and sends the "register" tag. This file, when receives this tag, verifies if the email address and bracelet ID exist in the database. If so, it informs the user by sending an error message to the application. If not, user data is stored. The password is encrypted using the cipher mentioned above and the salt is stored too, as well as the time (date and hour) when the user has registered.

After a successful registration, the login activity is prompted, where the user should enter the details he has just created. If a user enters the register's menu by mistake, he has also the opportunity to go back to login activity, by pressing the login activity button.



(a) Register Layout



(b) Registration Error Example

Figure 4.7: Register Activity

4.2.4 Forget Password Activity

Since a user cannot create more than one profile for the same bracelet, it is crucial that he can restore the password when he does not remember which one he chose. The aim of this activity is to provide the opportunity to change his password (figure 4.8).

The user must enter the email address that was used when he registered and provide a new password, which must be confirmed. If the passwords do not match, an error will occur.

To update his profile information, the application connects to the server [37] again. At this time, the tag is "forgetPassword". The file, on the website mentioned before, verifies if the email address provided exists in the database. If the answer is positive, his profile is updated and his new password is encrypted and stored, as well as the salt that was generated.

An user can leave the application whenever he wants, pressing the exit button. If this activity is prompted by mistake, he can always prompt the login activity again by choosing "Já tem conta? Clique aqui" option.

This process has, for now, an important hiatus, which compromises user's security. If a person discovers the email that someone used up in the registration procedure, he can easily change that user's password, because there is no security process that guarantees that the person who changes the password is the owner of that specific profile.

Hence, in future work, an extra security step should be added in the forget password activity. In addition to the changing procedure, an email to confirm that the user actually wants to change the password, should be sent to the email address provided in the initial registration phase. Then,

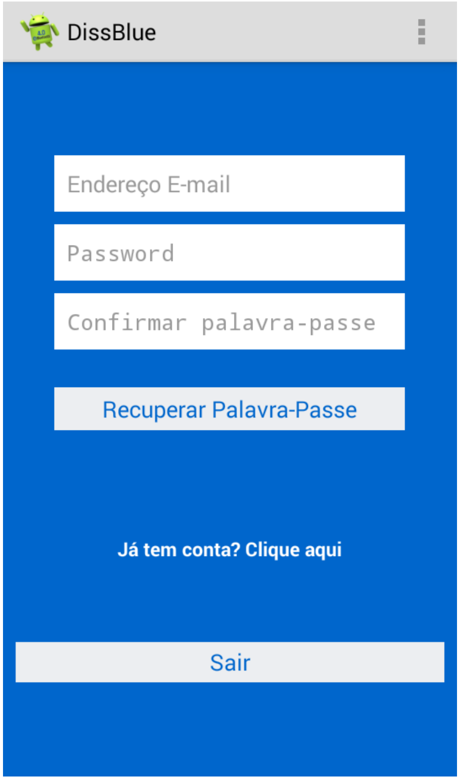


Figure 4.8: Forget Password Activity

the person has to click in a specific URL that guarantees that he really wants a new password. This verification step was not developed in the project, due to time restrictions.

4.2.5 Main Activity

The Main activity contains all the procedures and functions to search for children bracelets and to communicate with the Information Point (InfoPoint). It can be accessed if the user has already logged in the application. The username and email address are shown in the activity's layout, as well as the logout button (see figure 4.9).

The application requests to turn Bluetooth on, again, in case it was turned off since the login activity. If the user does not allow this request, the application will finish. The Wi-Fi is also turned on, and the vibrator service [40] is also initialized.

There is a method to enumerate all access points that the antenna has found, which will help the InfoPoint knowing where a specific person is. This function uses the "WifiScanReceiver" file to instantiate a broadcast receiver. This procedure was studied on Android Developer's website [41] and some auxiliar functions were added to this file. Firstly, the application searches for Wi-Fi access points and stores them in a list format variable. Then, it records the received power of each AP and respective BSSID (which is the identification provided by the AP) and returns the BSSID

Figure 4.9: Main Activity - Initial display indicating user details

of the access point with the maximum power. This process is constantly being summoned, thus the information is always being updated.

This activity has access to the user's profile information, so that it can search for his child's bracelet ID. The Bluetooth Smart Technology search routine [42] [43], which is periodically being summoned, has a timer associated that refreshes the search each 15 seconds (approximate period of time to scan and list the BLE devices). This process is executed in a thread, so that parallel processes can co-exist . The file "LeDeviceListAdapter" adapted from [44], available on the "helper" package, is crucial to store a list of BLE devices' address (which are unique and will be part of the message send to the Information Point). The main application verifies, then, if his child bracelet ID is included on that list.

The flowchart, available on the figure 4.10, represents how this application behaves, supposing that the user has already logged in the application.

If the address of his bracelet's child is found, the application only needs to send the list mentioned above and the boolean variable "child_ok" will have a "true" value. If the ID is not present in the scan list, an alarm situation will occur. The application uses the vibration service, which will activate the smartphone's vibration each 3 seconds (another thread starts at this moment and is executed periodically)."Child_ok" has, then, a "false" value and the Information Point will react accordingly. It is important to underline that, even if user's child bracelet ID does not make part of the list, the message send to the InfoPoint includes the other Bluetooth Smart Technology devices scanned, to aid other parents finding their children, if missing. Besides being

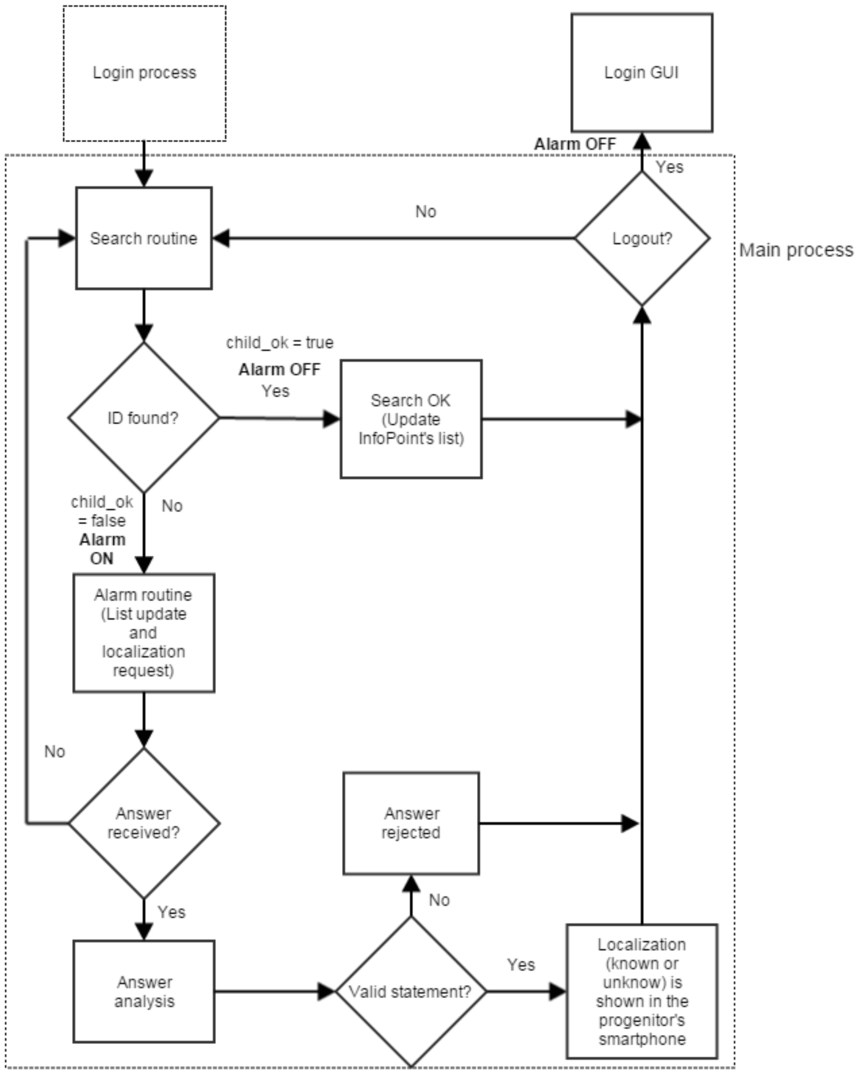


Figure 4.10: Main activity's flowchart

in the alarm mode, the BLE scan routine does not stop.

In order to leave the alarm mode, the Information Point sends the last known localization, which the user will follow. When he is close enough to his child, the scan routine will find this child's bracelet, the vibration will stop and the field "child_ok" will be true again.

The message that is periodically sent to the Information Point has the following structure:

- ID - Logged in user's ID;
- child_ok - False if child is missing;
- accessPoint_ID - BSSID of the AP closer to the user;
- currentTime - Timestamp of the sent message;

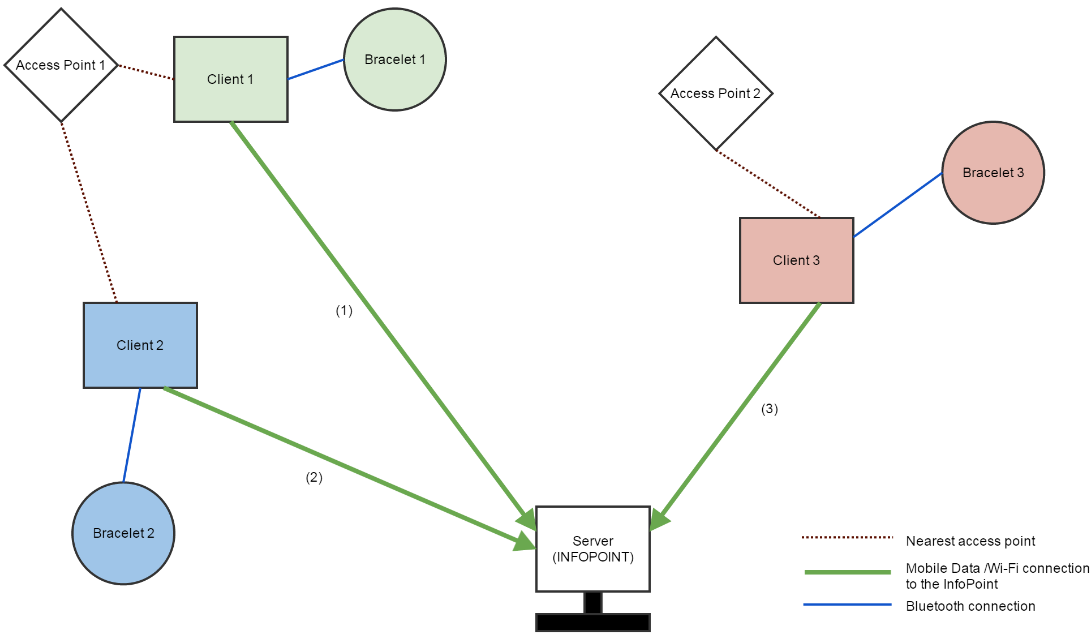
- braceletID_List - List of scanned BLE devices.

```
private String refreshStatus(String ID, boolean child_ok, String accessPoint_ID,
        String currentTime, String [] braceletID_List){
        .
        .
        .
        return MsgToSend;
}
```

Figure 4.11: Structure of the messages sent to the Information Point

The "refreshStatus" function returns a message that will be sent to the Information Point.

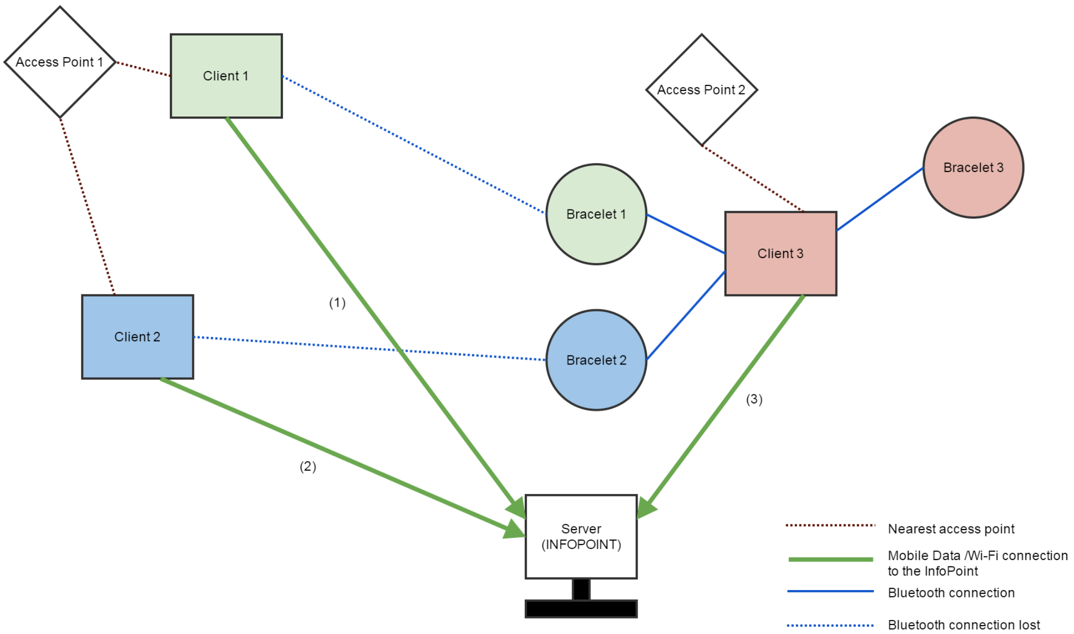
In figure 4.12 is shown a situation where every client can successfully find his child's bracelet. Each client's list, which is sent to the InfoPoint, contains not only the list of detected bracelets, but also the information about the nearest access point (AP), the timestamp and the value of the "child_ok" variable.



	Client ID	Child_Ok	Nearest AP	Timestamp	List of detected bracelets
(1)	Client 1	True	AP 1	10:30h	Bracelet 1
(2)	Client 2	True	AP 1	10:30h	Bracelet 2
(3)	Client 3	True	AP 2	10:30h	Bracelet 3

Figure 4.12: System operation and clients' messages - Situation 1

Now the situation depicted in figure 4.13, clients 1 and 2 lose track of their children. Simultaneously, they could not track any other bracelet, so their bracelets' list is empty. In the other hand, client 3 found the three bracelets, thus the list sent by him, to the InfoPoint, will aid the other two parents finding their children.



	Client ID	Child_Ok	Nearest AP	Timestamp	List of detected bracelets
(1)	Client 1	False	AP 1	10:35h	
(2)	Client 2	False	AP 1	10:35h	
(3)	Client 3	True	AP 2	10:35h	Bracelet 1, Bracelet 2, Bracelet 3

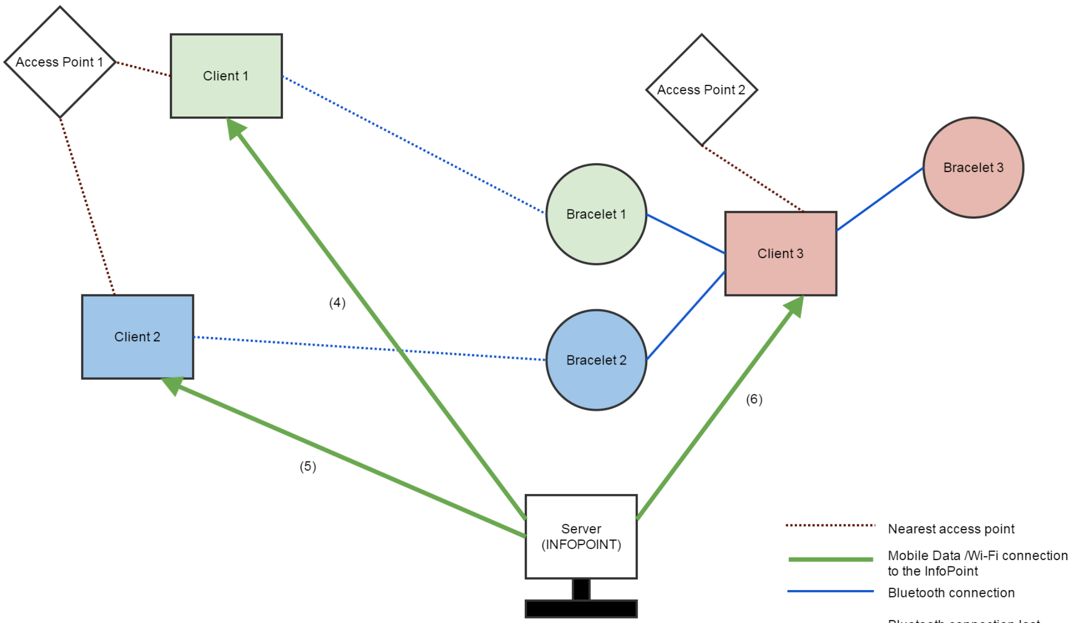
Figure 4.13: System operation and clients' messages - Situation 2

There is a thread running at the Information Point, which is waiting for the messages sent by users. TCP (*Transmission Control Protocol*) sockets transport the packet and enable the exchange of information.

The auxiliar file "ClientConnection" starts a socket and connects to "gnomo.fe.up.pt", at the port 5555 where the InfoPoint application is running. It converts the message to a specific structure in order to the socket understands it and then sends it to this hostname, to this specific port. TCP sockets were chosen over UDP (*User Datagram Protocol*), because it is important to acknowledge that the message has really reached its destination.

In figure 4.14 there are represented the messages sent by the server (InfoPoint) to the clients. Since it was client 3 who found bracelets 1 and 2, the nearest access point, regarding these bracelets, is AP 2. The stretch of time included in the messages received corresponds to the difference between the current time and the time each bracelet's last know localization was updated.

The user can logout at any time. It will stop the scanning and the vibration service, as well as the refresh status, even if his child is missing.



	Client ID	Message received from the server
(4)	Client 1	Your child was found next to AP 2, 1 second ago.
(5)	Client 2	Your child was found next to AP 2, 1 second ago.
(6)	Client 3	Your child is safe.

Figure 4.14: System operation and clients' messages - Situation 2 - Replies from the server

4.2.6 Other Considerations

Since the Java application is running on the Faculdade de Engenharia da Universidade do Porto's server, and if the application is tested outside its network, there is the need to use a VPN, on the smartphone, to connect to this server. Without the VPN, there are no message exchanged between the smartphone and the application.

The "AndroidManifest" file contains some important information about the application. The permissions, described on the figure 4.4 (a), are declared on this file (figure 4.15 (a)).

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.VIBRATE" />
```

Figure 4.15: Android Manifest File

It specifies the minimum API in which this application can run (18), because it was the first

API that supports the Bluetooth Smart Technology routines. The version of the application and its package is also represented on this file.

Some applications were important to test the hardware. Firstly, it was crucial to realize if the mobile phone, which was used to test the application, had a BLE antenna. The Android version is 4.4.2 (it is over 4.3), so in order to test the antenna, BLE Checker was downloaded [45]. Since it gave a positive answer, the phone BQ Aquaris 5 could be used. Before using this mobile phone, an Android phone was emulated [46], which could run in a computer and it would save BQ's battery. It was useful to test initial routines (like the registration, login, connection to database) but it can not be used to test the Bluetooth Smart Technology scan routines, because the emulator could not simulate a bluetooth antenna.

Secondly, the bracelet prototype needed to be tested too. An application developed by Nordic Semiconductors, which manufactered the module, was installed on the smartphone [47] and the other application [48] was installed on the Arduino, using Arduino IDE. It first looks for BLE devices and shows them on a list (figure 4.16 (a)). The desktop application connects to the module, over the Arduino, and shows a "connected" message on Arduino IDE Serial Monitor (figure 4.16 (c)).

The smartphone sends a message ("Hello laptop"), which is captured by the nRF8001 module and displayed on the serial monitor. Then the desktop application sends a message to the smartphone ("Hello smartphone"), which has received it correctly (figure 4.16 (b)). The UART protocol used on this test requires that the baudrate is the same on the smartphone and desktop application, so that the information is correctly decoded.

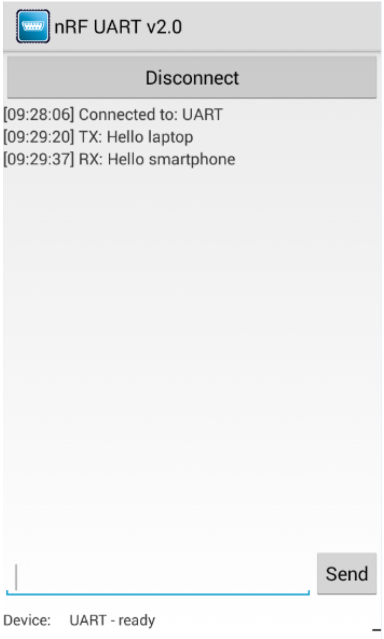
In the application developed in this dissertation, there are no messages exchanged between the smartphone and this module and the Arduino only provides power supply to nRF8001. This test was conceived to realize if the module is working properly and to measure how far the smartphone can be from the module.

An important and somehow difficult task of the development was debugging the application. The "Log" function was used [49] so that we could print out some successful (or unsuccessful) messages, as well as the errors that occur on the development itself. There is the need to enable the "Programmer Options" and USB debugging on the smartphone, in order to the application, that is being tested, to send the log over USB to the Eclipse IDE, while it is executing. This technique is crucial, because if the application has an error, it just stops running and does not mention what caused the problem.

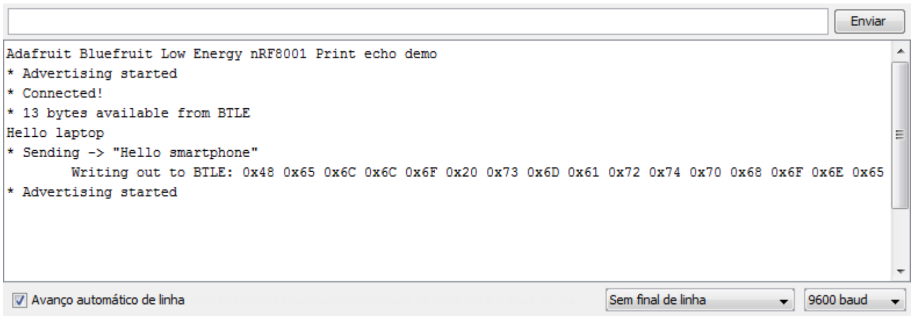
Before starting developing the application, there were some packages installed on the Eclipse IDE. Android Studio's package allows, not only to create a singleton Android project, but also to help debugging the application. An ADB server is executed when the application starts running and it captures the information sent by the mobile phone, from the errors to the state messages. This package was responsible to aid on creating the Android emulator too, which simulated the mobile phone's behaviour.

The database (a sample user is available on the figure 4.17) was developed in SQL and executed on Faculdade de Engenharia da Universidade do Porto's database service [50] (this

(a) nRF Application - Connection



(b) nRF Application - Messages exchanged



(c) Desktop Application

Figure 4.16: nRF 80001 test application

website is only available on FEUP's network), which stores every user's profile. It is accessed indirectly by the Android application (using the volley library [39]) and by the Java application, using JDBC (explained in the next section).

To keep the application's source code secure, as well as the Java application, all the information is stored at a FEUP repository (redmine). It prevents some fatality to happen and allowed to work in more than one computer, using updated information always.

The android application was tested in more than one smartphone, to verify if its layout would fit in different screen sized mobile phones and if the main routines were executed correctly.

uid	unique_id	name	email	braceletID	encrypted_password	salt	created_at
1	5551d89ce46c07.86557588	Ivo	ivo-rocha@live.com.pt	E4:B4:2A:DA:D5:17	FFrWE7Ss3iqYpINK3l+w5DQSC7RhZmJINzY5YTI5	afbe769a29	2015-05-12 11:40:28

Figure 4.17: Application Database

4.3 Server Side

A Java-based application is executed on a computer at the Information Point. It receives the status messages from users and updates each bracelet's last known localization, storing the access point AP closer to the user. The following flowchart explains how the routine works.

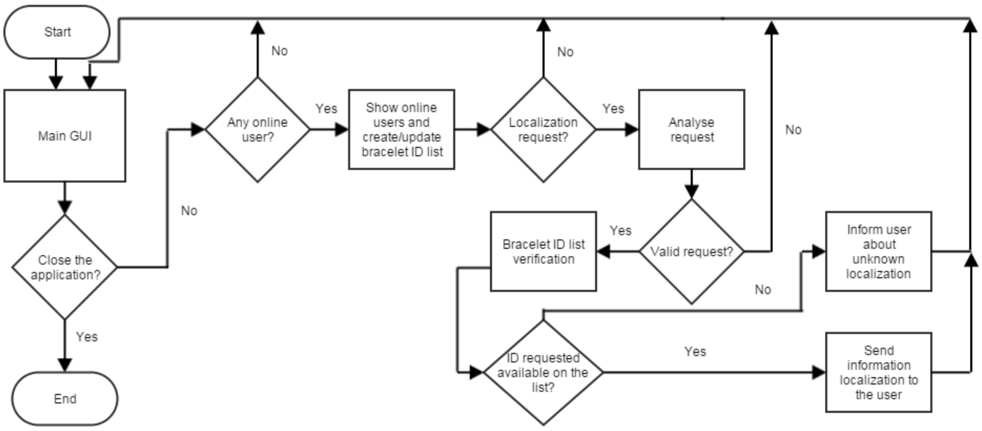


Figure 4.18: Java application flowchart (server side)

When the application starts running it has a label indicating that this is the Information Point application. It has a TCP socket executing on a thread which waits for the messages sent by the users. Like it was mentioned before, these messages include the identification of the user (not his name). The application connects directly to the database, using JDBC functionality [51], and selects the name of the user, using his identification (unique_id). In order to connect to the database, the application must be running on FEUP's server or use a VPN service. This library needs to be compiled with the application.

Then, the application uses a JList field to dynamically show which users are online. This list is updated in background mode and the visible field is only updated if a new user joins the network. To avoid a user from being always in the list (if a user does logout on the application, he is offline) a timer starts running each time a user sends a message (N users: N timers). After a period of 30 seconds without sending a refresh status messages, it is removed from the list, available in figure 4.19, where there are three online users.

Data treatment is invisible on the Java application frame. Each time a message is received, the field "child_ok" is analysed. If it has a "true" value, then the list of detected bracelets is further analysed. A routine goes through this list and for each bracelet ID:

- If this is the first time this identification appears, it stores a new bracelet ID field, among with the access_point BSSID (last known localization);
- If this field already exists, the access_point BSSID is updated.

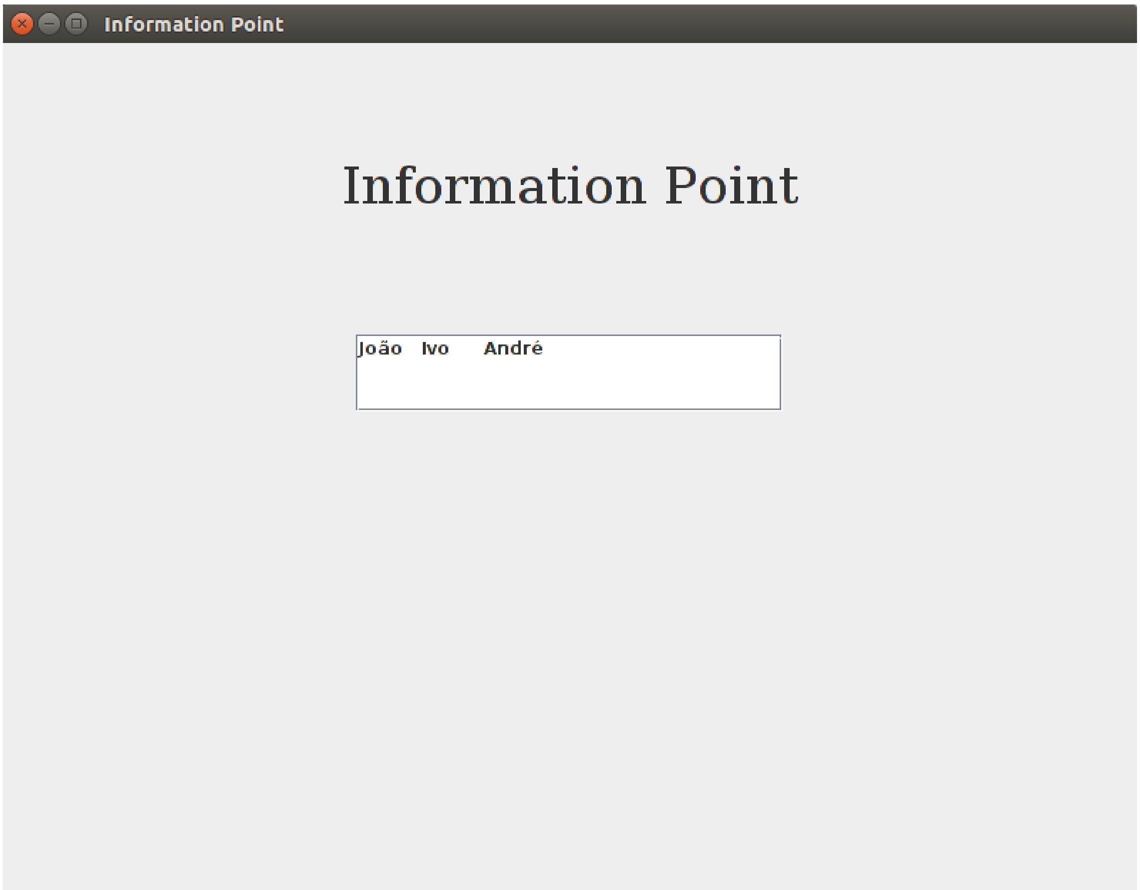


Figure 4.19: Java application on Information Point server (list of online users)

The specified routines guarantees that each bracelet ID's information kept is always updated and ready to be helpful to aid parents, whose children are missing. According to the flowchart (figure 4.18), it is important to highlight that, even if there is no localization messages, the lists are updated.

On the other hand, if the field "child_ok" is false it means that a child is missing. In this case, the first thing that the application does is to connect to the database and confirms which bracelet ID corresponds to this user. Then, it goes through the bracelet ID list sent by the users and verify if his child identification does not make part of that list. If it does, then a "false-negative" situation occurs and the user is advised that he does not need to concern. If it does not, it looks for the user's bracelet ID and builds a message that contains its last known localization.

Since the application keeps the time relative to every message is received, it calculates the difference between a user's localization request and the time that his child bracelet ID was last updated. The application uses a TCP socket to send the BSSID of the access point and the time difference mentioned above, so that the user knows how long ago his child was last seen.

The information relative to the bracelets is stored on a "TreeMap" variable [52]. It stores a pair of bracelet ID (key) and access point (value) and if a child is missing, the AP closer to the user, who sent the message, can be obtained using the key.

JDBC procedure requests an open-source library, which can be downloaded from MySQL website [53] and it has to be compiled at the same moment as application.

Chapter 5

System Testing and Results

5.1 Test Scenarios

In this chapter, there are described three different tests scenarios applied to the system. The results and messages exchanged between the smartphone and the Information Point are summarized and explained here as well.

These are the three developed tests:

- First Test Scenario : 1 parent - 1 child;
- Second Test Scenario : 1 parent - N children;
- Third Test Scenario: N parents - N children;

The first test scenario consists on testing the application standalone. In other words, there is no connectivity between the application and the Information Point: the aim of the test is the verification of the basic procedure - the cellphone scans for the child's bracelet associated to the user and, if it does not find it, the vibration will be activated. When the child's bracelet is next to the user again, the vibration will stop.

Concerning the second test scenario this will include, not only the first test scenario procedures (with some modifications explained, later, in this document), but it will also test the connectivity with the Information Point. Another bracelet will be added to the network to confirm that the cellphone doesn't vibrate, even if it finds a Bluetooth Smart Technology device which does not correspond to his child's bracelet.

On the final test, a second parent is added to the network. The search routines are the same as the second scenario, although the user will be forced to lose track of his child. The second user must keep track of both children's position and the information that he provides to the InfoPoint will aid the first parent in locating his child.

Since the number of BLE (Bluetooth Low Energy) available devices, in the tests, is confined to two, there is a maximum of two parents and, consequently, two bracelets.

In the following subsections, each test procedure will be detailed.

5.1.1 First Test Scenario : 1 parent - 1 child

In the first test scenario, the principal aid of the Information Point is dismissed (nevertheless, the internet connection is indispensable, because the application needs to connect to the database to validate the user's credentials). The user logins on the application and the scan routine starts. Furthermore, the application is designed so that a message is sent to the Information Point each 30 seconds. On this test this feature was disabled.

The messages that inform the user about his child's position were modified, so that the application could work locally without the Information Point. In the other two test scenarios, it is the InfoPoint that communicates the user about his child status (under the security zone or, if he/she is missing, it informs about his/her last known localization).

Besides testing the communication itself, this test will evaluate how far the bracelet can be from the user (both bracelets mentioned in chapter 3 will be tested) and how much battery is used up during these procedures (two mobile phones were tested).

The test will consist on:

1. Turning the bracelet's power off;
2. Scanning for the bracelet;
3. Turning the bracelet's power on;
4. Scanning for the bracelet;
5. Turning the bracelet's power off;
6. Scanning for the bracelet.

This test will take place in two different mobile phones, although not simultaneously (the network is not created). This "standalone" test is useful to acknowledge when the child is at the user's sight.

The results are available on section 5.2.1.

5.1.2 Second Test Scenario : 1 parent - N children

The second testing scenario will include the communication with the Information Point. The user's cellphone will scan for Bluetooth Smart Technology devices and it will send a message to the Information Point, which will answer accordingly.

The results will include, not only the messages exchanged between the mobile phone and the Information Point, but also some comments about these messages (for verification purposes). Two different cellphones will be used on this scenario, each one with a different user logged in, testing a different bracelet.

This test will consist on:

1. Turning both bracelets' power off;
2. Scanning for the bracelets;
3. Turning the user's child's bracelet power on;
4. Scanning for the bracelets;
5. Turning the other bracelet's power on;
6. Scanning for the bracelets;
7. Turning the user's child's bracelet power off;
8. Scanning for the bracelets.

In the last step, the mobile phone is supposed to alert the user that his child's bracelet was not found in the scan routine. The message sent by the Information Point will have a timestamp that was generated when the 6th step took place. The process to keep this variable always updated is explained on its test scenario results.

5.1.3 Third Test Scenario : N parents - N children

In this test scenario, the parent's network will be created. The aim of this last test is to confirm if the information about each bracelet's localization is constantly being updated, so that the Information Point always sends the last known localization of each child.

It will simulate a user which loses track of his child and has to communicate that situation to the InfoPoint. A second user will keep track of both children and the message, which he will be constantly sending, will contain both bracelets on its list. Afterwards, the first user will receive his child localization on his mobile phone, being this message sent by the InfoPoint, but based on the information transmitted by second user's smartphone).

This test will consist on:

1. Turning both bracelets' power on;
2. User number 1 and user number 2 scanning for the bracelets;
3. User number 1 losing track of his child. User number 2 keeping track of both children;
4. User number 1 and user number 2 scanning for the bracelets.

The results on this test scenario will contemplate not only the smarphone's log messages, but also the information which is constantly being received by the InfoPoint. Contrarily to the previous test scenario, the Java application will have to accept messages from two users, simultaneously, and it must not mix their information.

5.2 Test Results

In this section all test results will be available. In addition, it includes details and comments about each result to evaluate its precision and correctness.

5.2.1 First Scenario Results

This standalone test evaluates only the communication between each smartphone and its owner child's bracelet. The test, that was purposed for this scenario, took place twice and the results, for the first smartphone, are available in figure 5.1.

It is important to highlight that this smartphone (BQ Aquaris 5) is running Android 4.4.2 and it has a Bluetooth Smart Technology antenna.

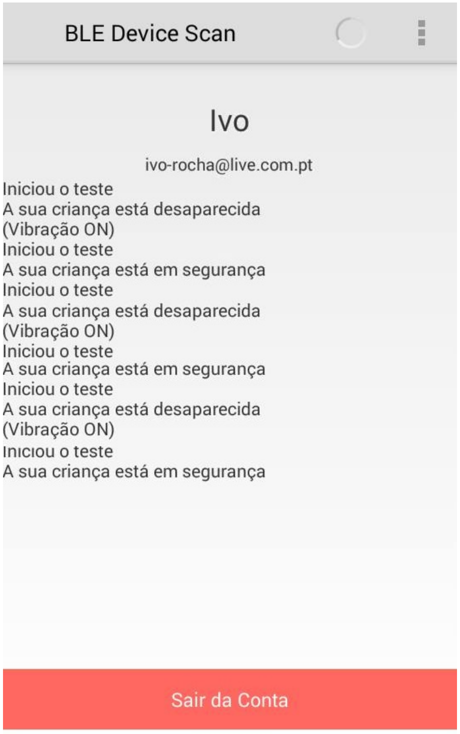


Figure 5.1: First Test Scenario - Application screen messages (smartphone with Android 4.4.2)

When this test initiated, the bracelet power was off so the antenna could not find the device. Therefore, the vibration was activated. When the power was on and the main routine scanned for the device, it was able to find the bracelet and the smartphone advised that the child was on the security zone. The third step consisted on turning the power off again, to simulate that the device was unreachable. The vibration was activated, again, to warn the user about this issue.

This test was repeated, to evaluate its sustainability, once again. The results demonstrate that the application worked as accurate as the first time.

It is crucial to underline three details. Firstly, it is not specified in this figure when the vibration is off - it is off when the child is at the security zone. Secondly, the messages presented in this picture ("A sua criança está em segurança" and "A sua criança está desaparecida") were developed to make this test scenario possible, because it is the Information Point that communicates if the child is on the security zone or not. Since there is no connectivity with the InfoPoint, these messages were "attached" to the application to advise the user about his child's situation. Thirdly, the child's bracelet that corresponds to the user logged in this application ("Ivo") is represented by the nRF8001 module, which has the BSSID **E4:B4:2A:DA:D5:17**.

As it is mentioned in the 4, the scan routine is called every fifteen (15) seconds. The application needs this amount of time to search for the devices and to display their status in the smartphone's screen. When this test started, the application took this stretch of time since the test initiates ("Iniciou o teste") until the status's message is communicated to the user.

The next figure (5.2) represents the same proceeding testing. The steps took to test the application were exactly the same. The smartphone used to test the application was different (Motorola Moto G), which is running Android 5.0: it allowed to prove that the application could run not only on the API it was developed for, but also on newer Android versions.

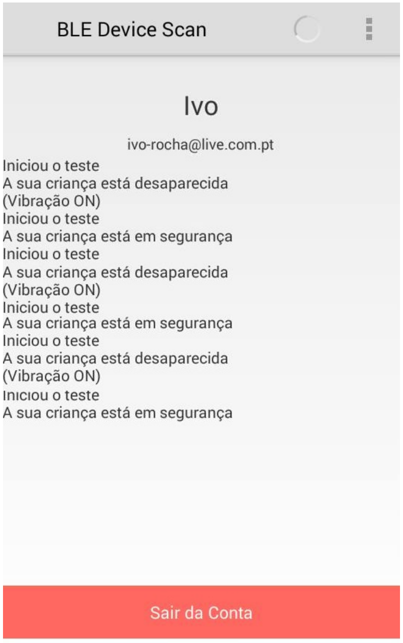


Figure 5.2: First Test Scenario - Application screen messages (smartphone with Android 5.0)

The user that tested the application on this smartphone is different ("Andre") and the bracelet ID that corresponds to his account is different too (Texas CC2540K module which has the BSSID **D0:39:72:A7:D0:C3**).

Regarding the picture above, it is possible to conclude that the results were the same and the premise that the application could run on newer Android version was verified, either.

The three details that were underline regarding figure 5.1 apply also in this test scenario. The stretch of time that the application lasted, since the test is initialized until the message appears, is the same (fifteen seconds).

Besides testing the communication itself, it was tested how the signal degrades as the distance between the smartphone and the bracelet rises. Seven different distances were tested (0.1m, 0.5m, 1.0m, 2.5m, 5.0m, 7.5m and 10.0m). The results that regard the last distance (10.0m) were not described in the following tables, as it corresponds to the maximum range of these devices bluetooth's antennas and the signal was hardly captured by both smartphones.

Furthermore, it was tested if these values were deteriorated when the Wi-Fi antenna was on, since Bluetooth Smart Technology and Wi-Fi operate in the same band (Industrial Scientific and Medical). Both smartphones and bracelets were tested and the summary of the results is available on tables 5.1 and 5.2. Each entry on these tables represent the average of ten samples that were tested, by each smartphone, and they test the **RSSI** (Received Signal Strength Indication) value measured by each smartphone. All samples are available on appendixes A and B.

	Moto G		BQ	
	WI-FI OFF	**WI-FI ON**	**WI-FI OFF**	**WI-FI ON**
0.1m	-44.3dB	-44.4dB	-50.7dB	-50.9dB
0.5m	-66.3dB	-68.4dB	-69.1dB	-79.2dB
1.0m	-68.6dB	-72.6dB	-75.1dB	-81.1dB
2.5m	69.9dB	-74.8dB	-75.7dB	-89.6dB
5.0m	-72.2dB	-76.3dB	-87.5dB	-89.8dB
7.5m	-89.6dB	-90.8dB	-90.5dB	-97.2dB

nRF8001 module - RSSI values

Table 5.1: Tests regarding Nordic Semiconductor module (bracelet 1)

After analysing the table 5.1, which summarizes the results on testing the application on the nRF8001 module, it is possible to confirm that the signal degrades as the distance rises. In other words, the closer the child is to his parent, the better the signal, captured by the smartphone's antenna, is. Moreover, turning Wi-Fi on, which is indispensable to enable messages exchange with the Information Point in the following test scenarios, worsens the Bluetooth Low Energy signal. This test validates the figure 2.1, at chapter 2.

It is not possible to conclude if the Moto G results are better than the BQ ones because of the Android version (since the Moto G is running Android 5.0) or because the hardware is better(which is the best presumption, since this smartphone was manufactured by Google, as well as the operating system running on it, which guarantees better performance).

The conditions on which these tests took place did neither include a great number of people (since it was tested on an empty wide room, at the university) nor "obstacles" between the bracelet and the smartphones.

The table 5.2 summarizes the tests applied to the Texas module. Comparing this module to nRF8001, it is possible to verify that the signal emitted by this antenna does not degrade as much as the Nordic Semiconductors' module, mainly when the Wi-Fi is on (which represents a big plus, since Wi-Fi indispensable in this system).

Before exploring the last test, it is fundamental to highlight how these measures took place in the smartphones.

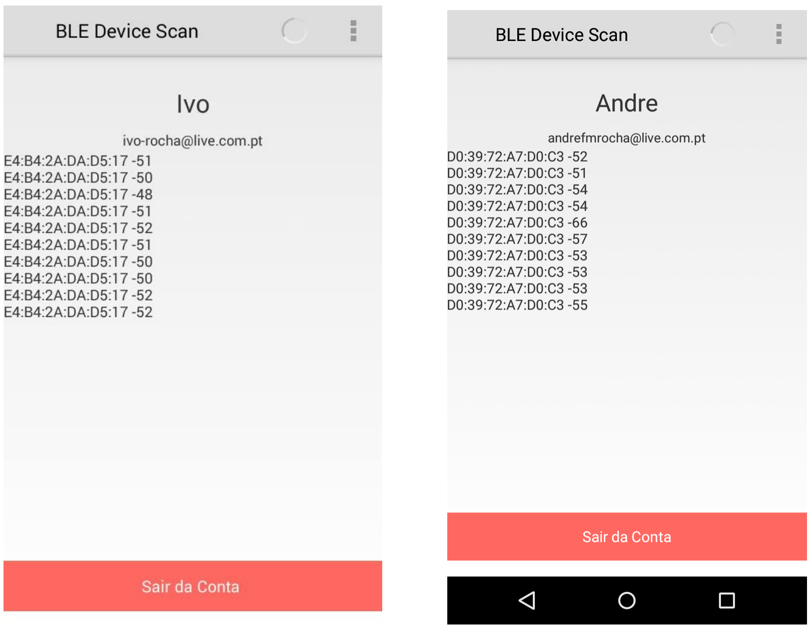
	Moto G		BQ	
	WI-FI OFF	**WI-FI ON**	**WI-FI OFF**	**WI-FI ON**
0.1m	-43.3dB	-46.3dB	-53.3dB	-54.9dB
0.5m	-54.8dB	-54.8dB	-54.0dB	-56.7dB
1.0m	-66.7dB	-67.5dB	-59.6dB	-61.3dB
2.5m	67.7dB	-70.1dB	-70.7dB	-75.2dB
5.0m	-71.1dB	-75.8dB	-86.8dB	-87.4dB
7.5m	-80.1dB	-80.8dB	-90.5dB	-90.7dB

D0:39:72:A7:D0:C3 (Texas module)

Table 5.2: Tests regarding Texas module (bracelet 2)

The figure 5.3 shows the RSSI values (dB) determined by each smartphone's antenna.

The BLE scan routine, which was developed in this application, has the ability to command the hardware to measure the power (in dB) emitted by each Bluetooth Smart Technology device.



(a) BQ device (0.1m distance, Wi-Fi On)

(b) Motorola device (0.5m distance, Wi-Fi Off)

Figure 5.3: RSSI values determined by the application BLE scan routine

To complete this test scenario, both smartphones did a stress test: It consisted on running this application for three hours, to evaluate how much battery is used up. It is important to underline that the Wi-Fi was on during this process. The results are the following:

- BQ device : During this test, 39% of the battery was used up;
- Motorola device: During this test, 43% of the battery was used up.

5.2.2 Second Scenario Results

The following test scenario includes not only searching for two bracelets at the same time, but also the connectivity with the Information Point.

Once again, both smartphones and bracelets were tested, and the messages exchanged are available in this subsection.

Since the smartphones need to communicate with the Information Point, the Wi-Fi must be always on. The access point, which is crucial to aid in locating the children, is the same on every tests and it has the BSSID **00:24:17:9f:a7:7c**. To calculate which access point the smartphone should include on its message, it reads every access point that was found y its antenna and chooses the one with higher power.

The Information Point application runs on Faculdade de Engenharia da Universidade do Porto's server and the mobile phones need to connect to this server to send (and receive) the information messages.

Next, the figure 5.4 is analysed.

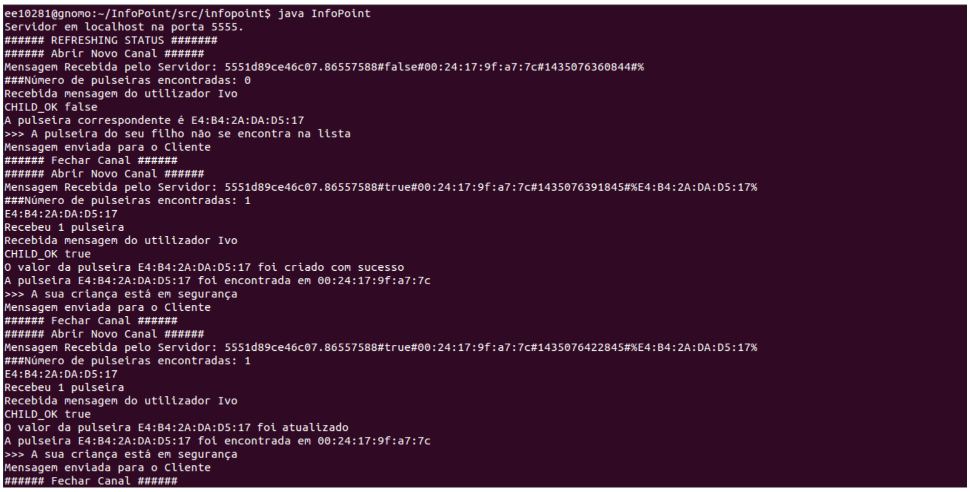


Figure 5.4: Second Test Scenario - Messages sent by user one (Part 1)

After the application is executed, the smartphone communicates to the Information Point that he is online, by sending a status message. The character "#" separates every field that composes the message:

1. User Identification;
2. Child_Ok (which is true if the user's bracelet is on the scanning list);
3. Access Point BSSID;
4. Timestamp indicating when the message is sent;
5. Bracelet ID List (which is empty in the first message);

After the message, it is mentioned how many bracelets were found and who has sent the message (to acknowledge the username, the Information Point has to make a connection to the user's database and search which username corresponds to this user's identification). Then, it highlights that no child was found, because the "child_ok" variable is false. The next procedure is to verify which bracelet corresponds to this user's identification. Again, the application needs to connect to the database to retrieve the correct value.

Finally, the Java application looks for this user's bracelet ID on the "last known localization" treeMap, which is empty at this moment, so the localization of this child is unknown. The message, that the this user will receive on his smartphone is, "A pulseira do seu filho não se encontra na lista" only.

In the next message, this user's bracelet ID was found. It is available on the message's bracelet's list, so the variable "child_ok" has the "true" value. Since the "last known localization" treeMap was empty, this bracelet and the correspondent access point BSSID were added to it. Moreover, a new timestamp was calculated to acknowledge that a new value was added.

The Information Point reads the bracelet's list and as the "child_ok" has a true value, it will compare each bracelet to the one retrieved from the database. In this case they match, so the message sent by the application is "A sua criança está em segurança".

Then, a new message is sent which it is exactly the same as the previous one. This specific message has only one goal: verify if the bracelet's list is constantly being updated, which is possible to prove with the message. A new timestamp value is generated, which is subsequent to the previous one. The child is still on the safe zone.

The next message, available on figure 5.5, includes not only this user's bracelet, but also another Bluetooth Smart Technology device. The scan routine captured two ID's and included both on the message sent to the Information Point. The other ID was included in the Information Point bracelet list and the previous value was updated. A new timestamp was generated.

Afterwards, this user's bracelet was powered off. The message includes the other bracelet ID and advises the InfoPoint that the child is missing. Besides not finding this user's child, the other bracelet last known localization is updated to aid other parents, if needed. The application goes through its bracelet ID list and sends the last access point ID located next to this child. The timestamp available on this message is given by the difference between the time this message was created and the last time this specific bracelet ID information is added to the InfoPoint's list.

To sum up, the other bracelet was also powered off. The message received by the Information Point does not include any bracelet and the timestamp sent in the message corresponds to the last generated and has to be bigger than the one included in the previous message.

The messages displayed on the smartphone are described on the figure 5.6.

It is fundamental to analyse if these messages coincide with the ones received in the Information Point. Before sending a message, which occurs every 30 seconds, the smartphones complete two scan routines. The concurrency issues, inherent to the use of threads, was not working completely at this time, however it does not interfere on the messages sent to the InfoPoint, neither puts the children safety at risk.

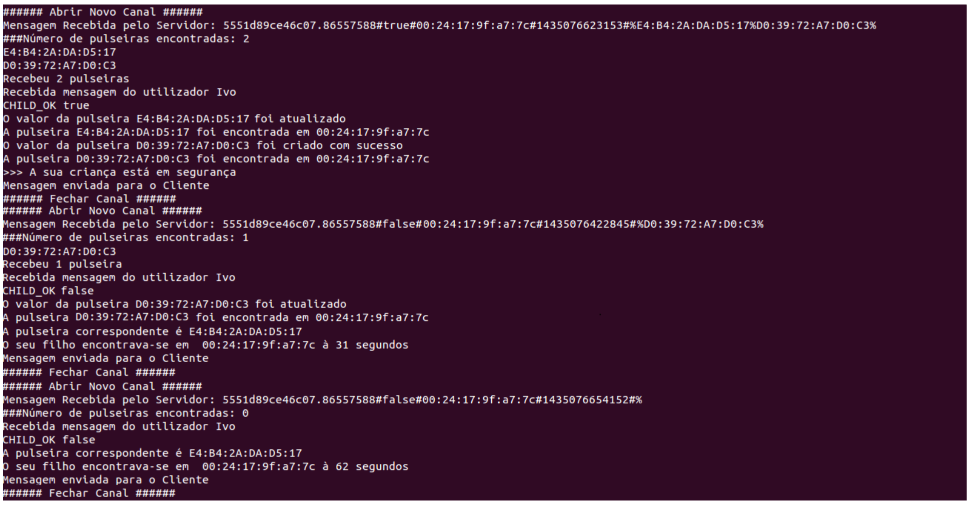
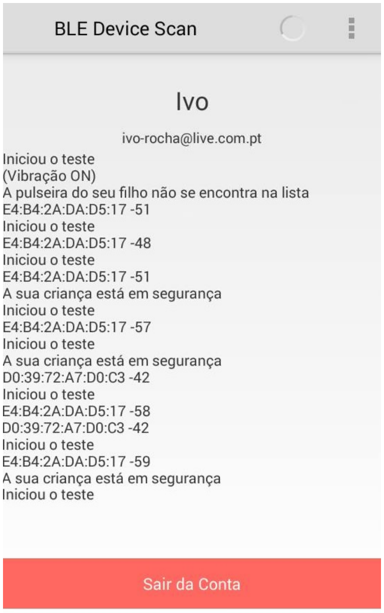
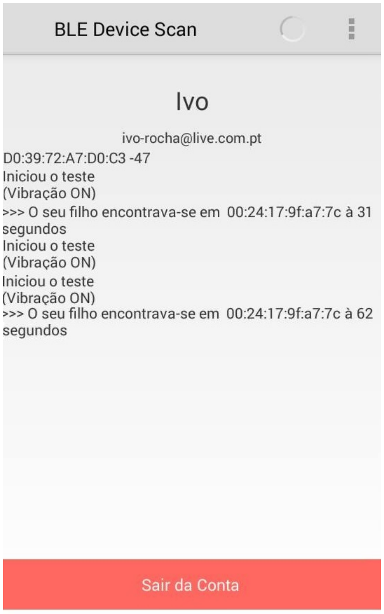


Figure 5.5: Second Test Scenario - Messages sent by user one (Part 2)



(a) Messages - Part 1



(b) Messages - Part 2

Figure 5.6: Second Test Scenario - Messages displayed on BQ's screen

In the figure 5.6 (a), three steps were analysed. The messages displayed are correct, according to the InfoPoint information. In the figure (b), the other bracelet is added to the network and both are included on the bracelet list sent to the Information Point. Then, this user's bracelet is powered off and his child's localization is sent by the InfoPoint. Finally, the other bracelet is powered off too, and the localization received by the smartphone is the same as the previous one, but with a delay.

After testing this smartphone (BQ), the other one (Moto G) was tested too. The logic and testing procedures are the same and the correctness of these tests is verified on the following pictures. Pictures 5.7 and 5.8 correspond to the messages received at the Information Point.

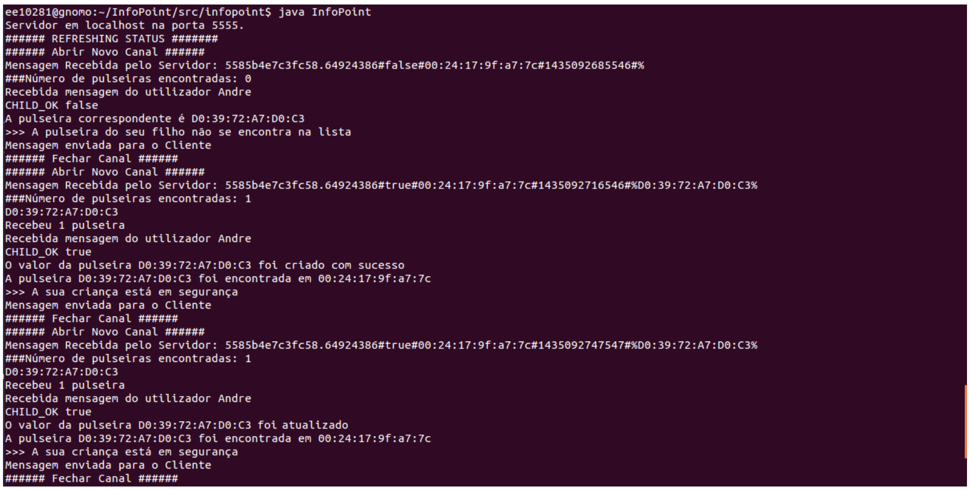


Figure 5.7: Second Test Scenario - Messages sent by user two (Part 1)

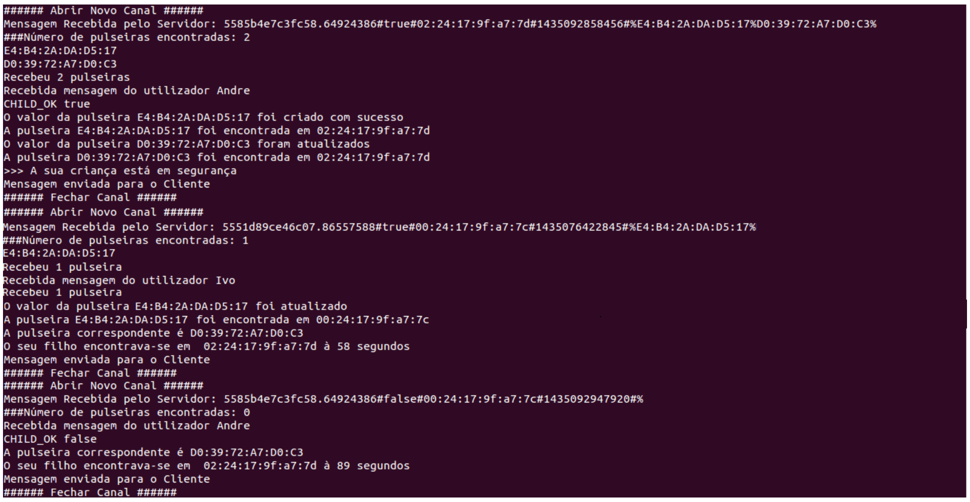


Figure 5.8: Second Test Scenario - Messages sent by user two (Part 2)

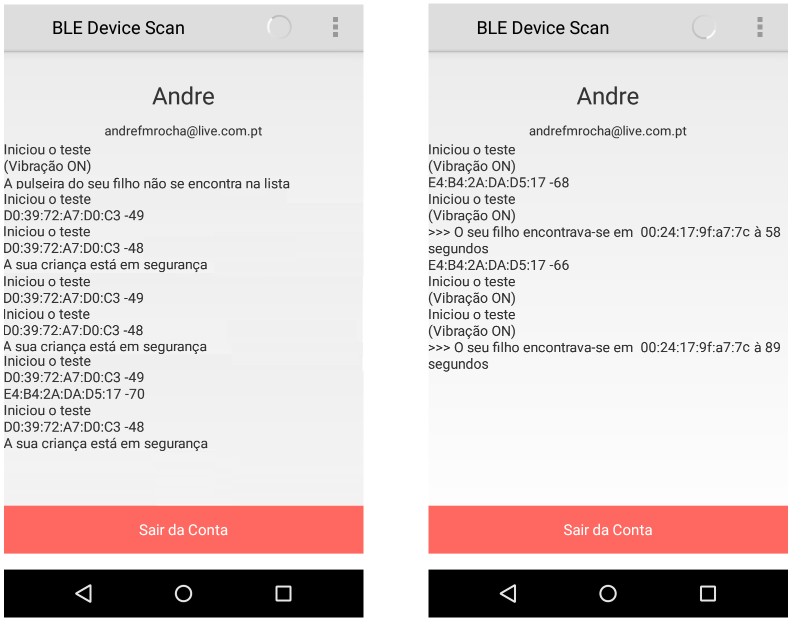
The messages received by this user were displayed on his smartphone and they are available on figure 5.9.

5.2.3 Third Scenario Results

The third and last test scenario involves, not only the two bracelets and the Information Point, but also another parent which, together with the first, will form the adult's network. The Information Point application will be waiting for the messages send by both parents and will update its bracelet's list, similarly to the second test scenario.

The figure 5.10 represents the messages sent by each parent, which will be described below.

The first message describes that user's number one ("Ivo") mobile phone could not find any bracelet, so the list is empty. The same has happened on the second message, but this time



(a) User number one messages - Part 1 (b) User number one messages - Part 2

Figure 5.9: Third Test Scenario - Messages displayed on Motorola's screen

regarding user number two ("Andre"). Both users are connected to the same access point (AP), whose BSSID is **00:24:17:9f:a7:7c**.

Afterwards, user's number two message reached the destination in first place. He is still connected to the same AP and he found two bracelets. Since the Information Point bracelets' list is empty, these new entries will be added to this list and the timestamp will be generated.

In the figure 5.11, it is possible to verify that the user number one is connected to the same access point and he found the same bracelets as user number two. The Information Point bracelets' list is updated, so as the timestamp.

To simulate user number one losing track of his child's bracelet (whose BSSID is **E4:B4:2A:DA :D5:17**), he moved to another room, where there is a different access point, and where his mobile phone could not track any bracelet. As he moves to another room, user number two stayed still and sent his status message to the Information Point, as usual. It is important to highlight that users send a new message every 30 seconds.

The last message, which was sent by user number one, demonstrates that, besides being connected to a different AP (whose BSSID is **e8:94:f6:3d:a7:86**), his bracelet's list was empty. The Information Point, after analysing the "child_ok" variable, searches on its database to determine which bracelet ID corresponds to this user. After this searching procedure, it checks if its bracelet's list has this specific bracelet ID. Once it has, it will send the last know localization to this user as well as the stretch of time since it was last found.

A real implementation of this system on a shopping center would be different in an important detail. The Information Point's application, before starting running, had to keep a list of access

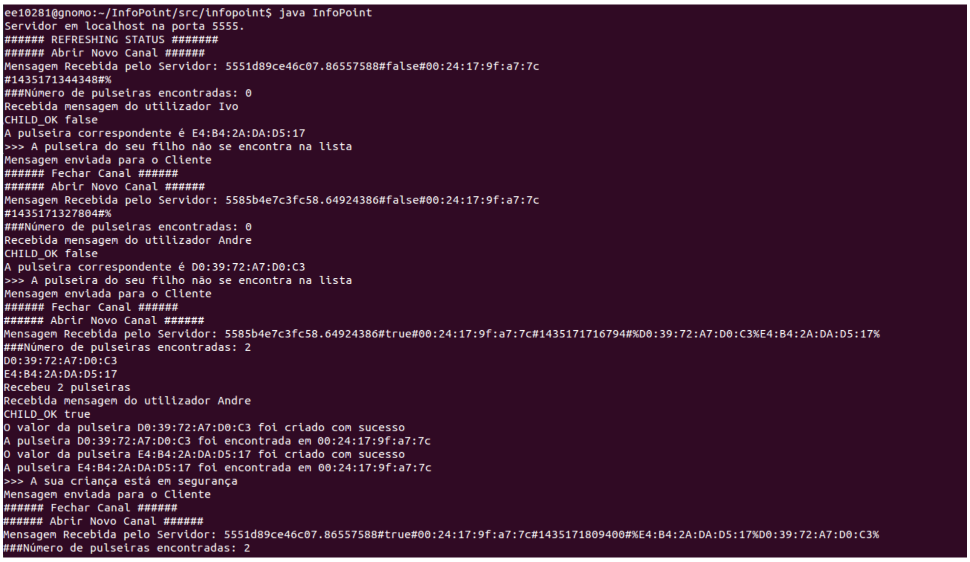


Figure 5.10: Third Test Scenario - Messages received by Information Point (Part 1)

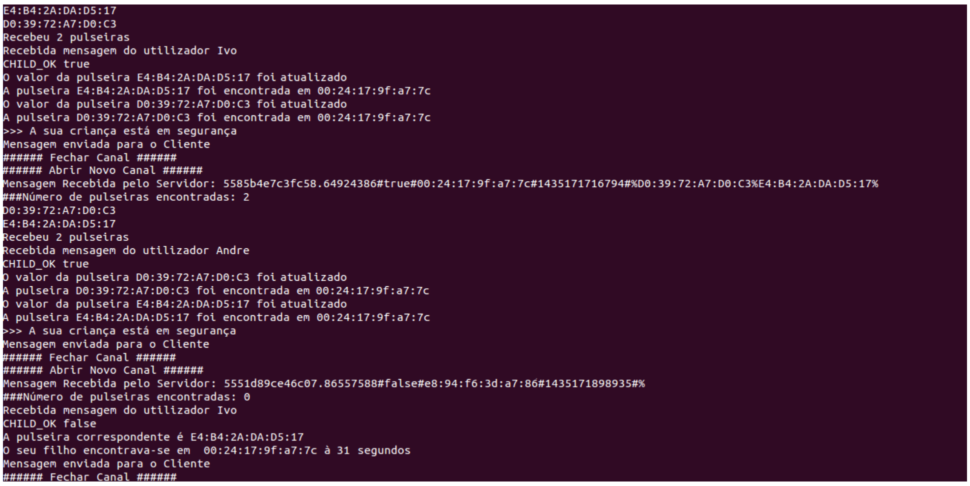
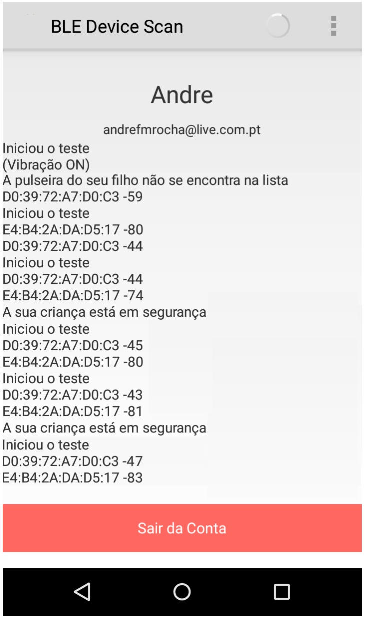


Figure 5.11: Third Test Scenario - Messages received by Information Point (Part 2)

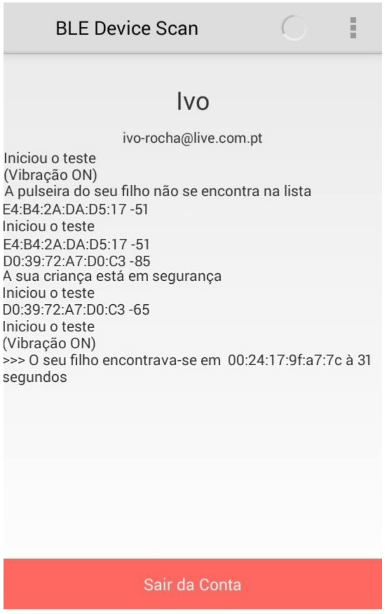
points' IDs and associate them with a specific store. When a user loses track of his child, the Java application would send the name of store closer to his child, instead of sending the access point BSSID (since the user would need to know where every access point is located). For demonstration purposes, and once this system was not implemented on a shopping center, the application sends the access point's BSSID.

Analysing each smartphone's "log", available on figure 5.12, it is possible to check that user number two message agrees with the messages received by the Information Point. Since he can always keep track of his child (apart from the first scanning routine), he does not need to concern, because his child is on the safe zone.

On the other hand, user number one loses track of his child in the last scan routine, so the Information Point sends him his child's the last known location, similarly to what happened on the second test scenario.



(a) User number two log



(b) User number one log

Figure 5.12: Third Test Scenario - Smartphones' log messages

Chapter 6

Conclusion and Future Work

This chapter exposes the conclusion and the final considerations about this project. In addition, there are presented some possible improvements relative to a future work on the system.

6.1 Conclusion

In this dissertation, it is described all the work which was developed regarding the theme "Localization technology in special contexts: a collaborative anti-lost application". The main goal of this system was to develop an Android application, that could track children positions, which use a Bluetooth Smart Technology device on their wrist at a shopping centre, in a real time basis. Technically, for this purpose, each child should use a Bluetooth Smart Technology device on his/her wrist. Furthermore, a Java-based application, running in shopping centre Information Point, must receive messages from every parent in the network and answer accordingly.

This closed, but wide, space, where the system is designed to work on, must have an access point's (AP) infrastructure which will aid in the children localization function. This is achieved, since the identification of the AP closer to the smartphone is included in the message sent to the Information Point. When an user's smartphone loses track of his child, it communicates with the Java application, which will send the last known location and the elapsed time since this child was last tracked.

As every parent scan for all bluetooth low energy devices nearby, including them all in a list of found devices, and sending, then, this list to the Java application, the system is considered collaborative. This is an important and crucial aspect of the system which greatly helps to keep children safe.

Developing an Android application, its routines, and working with Bluetooth Smart 4.0, a technology that has a very low power consumption and different scan routines, when compared to previous Bluetooth norms, was very challenging.

This project included three main phases:

1. System planning and conception;

2. System implementation;

3. System testing.

In the first phase, it was discussed how parents could keep track of their children in a normal infra-structured space, as a shopping centre, without needing additional components for the infrastructure and for the portable devices. Moreover, the device, which should be used by each parent, must be relatively cheap and easy to buy - a smartphone. Thus, taking advantage of the access points, available on almost every store, and the smartphones' bluetooth and wireless antennas, the system was, then, conceived.

The network centralization, represented by the Information Point Java application, can have any operating system, as Java is a portable language, hence, executed in almost any computer.

The second phase was responsible to transform the ideas from the first one into a real application. After deciding which devices could represent children bracelets and testing them, the Android application started being developed. The scanning routine, and the way how the smartphone communicates with the InfoPoint were, then, developed, as well as the Information Point application. This development lasted the most, comparing with the other two, and after an initial functional version, important improvements still occurred, as the tests took place.

Finally, the application was fully tested, and its performance results obtained in the final phase. This was fundamental to prove if the system could be implemented in the future and to understand if the conception and the implementation were correct. Relatively to the bracelets, the maximum range, inherent to BLE technology, enables tracking of children 7.5m far from his/her parent. Moreover, the battery used up is not critical, as it could run for a long period of time properly.

After completing successfully these three phases, the main goal of this project was fulfilled. Bluetooth Smart Technology is available on the newer smartphones and it is the best technology to deal with these periodic scanning procedures. The Android application, running on the smartphones, as well as the Java application, were completed on time and its routines' results demonstrate that the system was correctly implemented.

As the final conclusion, it is fundamental to underline that Bluetooth Low Energy is continuously growing, and used on the new products that arrive in the marketplace, presenting higher maximum ranges, which could improve this system's performance.

6.2 Future Work

After studying this technology's specifications and conceiving this system, we believe that this is a project with good perspectives. It could be implemented on a shopping centre, if an entity is interested and a partnership is celebrated.

Besides doing what it is purposed, both applications could still be improved and new features could be added.

Since the prototype is working properly, the first improvement should be creating the PCB (Print Circuit Board) and producing a real bracelet with the specifications which were already mentioned (apart from communicating through Bluetooth Smart Technology, it should have a smart lock to avoid an easy removal).

The forgot password procedure should also be improved. Instead of changing the password in the application itself, an email should be sent to the user, containing an URL that would aid the user on safely changing his password.

Each user, which makes his registration in the application, can only associate one child to his account. In the future, it could be possible to register a major number of children by each account. Afterwards, the logic of application must be changed regarding these modifications (as the application associates only one bracelet to each user, in the present version). Moreover, this project could be adapted to track objects or even elder people, for example. The bracelet could also include other sensors which would evaluate children (or elder people) health, such as a heart monitor or a thermometer, for instance.

If higher maximum ranges, inherent to superior Bluetooth Smart Technology classes, could be achieved, different "zones" could be created. The safe zone could be chosen by each user and the application would only advise the user if his/her child leaves that zone.

Using this system in open spaces, could also be a new possibility. Instead of using an access point's infrastructure, it would use the GPS antenna to aid locating the children. The precision is lower, comparing to the application developed in this project, nevertheless we think it can work as well.

The smartphones, which support this application, are only Android-based. It would be more versatile if it could run in any smartphone (iOs, Blackberry and Windows Phone). The goal is to maintain the same procedures and routines, notwithstanding running in different operating systems.

Appendix A

Tables - Moto G

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
0.1m	-45dB
0.1m	-45dB
0.1m	-44dB
0.1m	-42dB
0.1m	-43dB
0.1m	-45dB
0.1m	-46dB
0.1m	-45dB
0.1m	-45dB
0.1m	-43dB
Average = -44.3dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
0.1m	-44dB
0.1m	-44dB
0.1m	-44dB
0.1m	-46dB
0.1m	-43dB
0.1m	-45dB
0.1m	-46dB
0.1m	-43dB
0.1m	-44dB
0.1m	-45dB
Average = -44.4dB	

Table A.1: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 0.1m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
0.5m	-66dB
0.5m	-68dB
0.5m	-70dB
0.5m	-63dB
0.5m	-66dB
0.5m	-63dB
0.5m	-63dB
0.5m	-71dB
0.5m	-68dB
0.5m	-65dB
Average = -66.3dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
0.5m	-67dB
0.5m	-62dB
0.5m	-65dB
0.5m	-71dB
0.5m	-74dB
0.5m	-73dB
0.5m	-73dB
0.5m	-71dB
0.5m	-66dB
0.5m	-62dB
Average = -68.4dB	

Table A.2: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 0.5m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
1.0m	-68dB
1.0m	-66dB
1.0m	-76dB
1.0m	-68dB
1.0m	-67dB
1.0m	-66dB
1.0m	-69dB
1.0m	-69dB
1.0m	-67dB
1.0m	-70dB
Average = -68.6dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
1.0m	-76dB
1.0m	-74dB
1.0m	-77dB
1.0m	-71dB
1.0m	-76dB
1.0m	-73dB
1.0m	-70dB
1.0m	-67dB
1.0m	-72dB
1.0m	-70dB
Average = -72.6dB	

Table A.3: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 1.0m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
2.5m	-70dB
2.5m	-66dB
2.5m	-70dB
2.5m	-71dB
2.5m	-70dB
2.5m	-69dB
2.5m	-71dB
2.5m	-70dB
2.5m	-70dB
2.5m	-72dB
Average = -69.9dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
2.5m	-74dB
2.5m	-66dB
2.5m	-90dB
2.5m	-72dB
2.5m	-72dB
2.5m	-72dB
2.5m	-75dB
2.5m	-70dB
2.5m	-75dB
2.5m	-82dB
Average = -74.8dB	

Table A.4: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 2.5m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
5.0m	-70dB
5.0m	-77dB
5.0m	-73dB
5.0m	-71dB
5.0m	-74dB
5.0m	-75dB
5.0m	-69dB
5.0m	-72dB
5.0m	-70dB
5.0m	-71dB
Average = -72.2dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
5.0m	-78dB
5.0m	-76dB
5.0m	-75dB
5.0m	-74dB
5.0m	-77dB
5.0m	-76dB
5.0m	-75dB
5.0m	-76dB
5.0m	-79dB
5.0m	-77dB
Average = -76.3dB	

Table A.5: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 5.0m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
7.5m	-86dB
7.5m	-83dB
7.5m	-90dB
7.5m	-93dB
7.5m	-90dB
7.5m	-89dB
7.5m	-92dB
7.5m	-94dB
7.5m	-87dB
7.5m	-92dB
Average = -89.6dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
7.5m	-94dB
7.5m	-87dB
7.5m	-91dB
7.5m	-91dB
7.5m	-93dB
7.5m	-90dB
7.5m	-98dB
7.5m	-85dB
7.5m	-94dB
7.5m	-85dB
Average = -90.8dB	

Table A.6: User number one Bracelet (E4:B4:2A:DA:D5:17), Moto G, 7.5m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
0.1m	-44dB
0.1m	-42dB
0.1m	-42dB
0.1m	-44dB
0.1m	-42dB
0.1m	-44dB
0.1m	-45dB
0.1m	-42dB
0.1m	-44dB
0.1m	-44dB
Average = -43.3dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
0.1m	-46dB
0.1m	-46dB
0.1m	-47dB
0.1m	-47dB
0.1m	-47dB
0.1m	-47dB
0.1m	-47dB
0.1m	-46dB
0.1m	-46dB
0.1m	-44dB
Average = -46.3dB	

Table A.7: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 0.1m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
0.5m	-53dB
0.5m	-57dB
0.5m	-56dB
0.5m	-54dB
0.5m	-57dB
0.5m	-57dB
0.5m	-56dB
0.5m	-54dB
0.5m	-52dB
0.5m	-52dB
Average = -54.8dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
0.5m	-52dB
0.5m	-51dB
0.5m	-54dB
0.5m	-54dB
0.5m	-66dB
0.5m	-57dB
0.5m	-53dB
0.5m	-53dB
0.5m	-53dB
0.5m	-55dB
Average = -54.8dB	

Table A.8: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 0.5m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
1.0m	-72dB
1.0m	-84dB
1.0m	-64dB
1.0m	-60dB
1.0m	-66dB
1.0m	-66dB
1.0m	-63dB
1.0m	-60dB
1.0m	-66dB
1.0m	-66dB
Average = -66.7dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
1.0m	-71dB
1.0m	-76dB
1.0m	-68dB
1.0m	-60dB
1.0m	-72dB
1.0m	-66dB
1.0m	-68dB
1.0m	-65dB
1.0m	-69dB
1.0m	-60dB
Average = -67.5dB	

Table A.9: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 1.0m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
2.5m	-72dB
2.5m	-72dB
2.5m	-70dB
2.5m	-64dB
2.5m	-64dB
2.5m	-70dB
2.5m	-66dB
2.5m	-60dB
2.5m	-69dB
2.5m	-70dB
Average = -67.7dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
2.5m	-65dB
2.5m	-70dB
2.5m	-74dB
2.5m	-70dB
2.5m	-68dB
2.5m	-71dB
2.5m	-71dB
2.5m	-70dB
2.5m	-69dB
2.5m	-73dB
Average = -70.1dB	

Table A.10: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 2.5m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
5.0m	-67dB
5.0m	-68dB
5.0m	-77dB
5.0m	-81dB
5.0m	-68dB
5.0m	-73dB
5.0m	-65dB
5.0m	-70dB
5.0m	-70dB
5.0m	-72dB
Average = -71.1dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
5.0m	-71dB
5.0m	-73dB
5.0m	-69dB
5.0m	-75dB
5.0m	-69dB
5.0m	-75dB
5.0m	-80dB
5.0m	-83dB
5.0m	-82dB
5.0m	-81dB
Average = -75.8dB	

Table A.11: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 5.0m

(a) Wi-Fi Off

Moto G - WI-FI OFF	
Distance	**RSSI**
7.5m	-78dB
7.5m	-83dB
7.5m	-83dB
7.5m	-82dB
7.5m	-79dB
7.5m	-81dB
7.5m	-77dB
7.5m	-79dB
7.5m	-81dB
7.5m	-78dB
Average = -80.1dB	

(b) Wi-Fi On

Moto G - WI-FI ON	
Distance	**RSSI**
7.5m	-82dB
7.5m	-78dB
7.5m	-80dB
7.5m	-79dB
7.5m	-81dB
7.5m	-78dB
7.5m	-80dB
7.5m	-83dB
7.5m	-81dB
7.5m	-86dB
Average = -80.8dB	

Table A.12: User number two bracelet (D0:39:72:A7:D0:C3), Moto G, 7.5m

Appendix B

Tables - BQ Aquaris

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
0.1m	-51dB
0.1m	-50dB
0.1m	-48dB
0.1m	-51dB
0.1m	-52dB
0.1m	-51dB
0.1m	-50dB
0.1m	-50dB
0.1m	-52dB
0.1m	-52dB
Average = -50.7dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
0.1m	-52dB
0.1m	-50dB
0.1m	-49dB
0.1m	-53dB
0.1m	-51dB
0.1m	-51dB
0.1m	-51dB
0.1m	-51dB
0.1m	-51dB
0.1m	-50dB
Average = -51.0dB	

Table B.1: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 0.1m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
0.5m	-76dB
0.5m	-71dB
0.5m	-63dB
0.5m	-66dB
0.5m	-67dB
0.5m	-67dB
0.5m	-76dB
0.5m	-74dB
0.5m	-63dB
0.5m	-68dB
Average = -69.1dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
0.5m	-75dB
0.5m	-80dB
0.5m	-78dB
0.5m	-74dB
0.5m	-75dB
0.5m	-71dB
0.5m	-79dB
0.5m	-86dB
0.5m	-86dB
0.5m	-88dB
Average = -79.2dB	

Table B.2: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 0.5m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
1.0m	-71dB
1.0m	-72dB
1.0m	-76dB
1.0m	-70dB
1.0m	-71dB
1.0m	-79dB
1.0m	-71dB
1.0m	-77dB
1.0m	-83dB
1.0m	-81dB
Average = -75.1dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
1.0m	-91dB
1.0m	-89dB
1.0m	-95dB
1.0m	-86dB
1.0m	-85dB
1.0m	-85dB
1.0m	-84dB
1.0m	-70dB
1.0m	-74dB
1.0m	-52dB
Average = -81.1dB	

Table B.3: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 1.0m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
2.5m	-79dB
2.5m	-70dB
2.5m	-73dB
2.5m	-73dB
2.5m	-74dB
2.5m	-75dB
2.5m	-81dB
2.5m	-79dB
2.5m	-73dB
2.5m	-80dB
Average = -75.7dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
2.5m	-88dB
2.5m	-86dB
2.5m	-92dB
2.5m	-95dB
2.5m	-89dB
2.5m	-95dB
2.5m	-86dB
2.5m	-85dB
2.5m	-84dB
2.5m	-96dB
Average = -89.6dB	

Table B.4: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 2.5m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
5.0m	-85dB
5.0m	-95dB
5.0m	-85dB
5.0m	-85dB
5.0m	-88dB
5.0m	-86dB
5.0m	-95dB
5.0m	-89dB
5.0m	-87dB
5.0m	-80dB
Average = -87.5dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
5.0m	-85dB
5.0m	-88dB
5.0m	-88dB
5.0m	-88dB
5.0m	-88dB
5.0m	-90dB
5.0m	-95dB
5.0m	-92dB
5.0m	-89dB
5.0m	-95dB
Average = -89.8dB	

Table B.5: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 5.0m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
7.5m	-91dB
7.5m	-89dB
7.5m	-91dB
7.5m	-94dB
7.5m	-89dB
7.5m	-88dB
7.5m	-94dB
7.5m	-88dB
7.5m	-94dB
7.5m	-87dB
Average = -90.5dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
7.5m	-100dB
7.5m	-97dB
7.5m	-98dB
7.5m	-95dB
7.5m	-97dB
7.5m	-101dB
7.5m	-97dB
7.5m	-93dB
7.5m	-97dB
7.5m	-97dB
Average = -97.2dB	

Table B.6: User number one bracelet (E4:B4:2A:DA:D5:17), BQ, 7.5m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
0.1m	-58dB
0.1m	-54dB
0.1m	-55dB
0.1m	-54dB
0.1m	-51dB
0.1m	-54dB
0.1m	-53dB
0.1m	-51dB
0.1m	-53dB
0.1m	-50dB
Average = -53.3dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
0.1m	-57dB
0.1m	-54dB
0.1m	-56dB
0.1m	-54dB
0.1m	-51dB
0.1m	-56dB
0.1m	-54dB
0.1m	-58dB
0.1m	-55dB
0.1m	-53dB
Average = -54.9dB	

Table B.7: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 0.1m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
0.5m	-51dB
0.5m	-54dB
0.5m	-58dB
0.5m	-54dB
0.5m	-51dB
0.5m	-51dB
0.5m	-56dB
0.5m	-54dB
0.5m	-54dB
0.5m	-57dB
Average = -54.0dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
0.5m	-55dB
0.5m	-56dB
0.5m	-54dB
0.5m	-56dB
0.5m	-55dB
0.5m	-57dB
0.5m	-59dB
0.5m	-61dB
0.5m	-58dB
0.5m	-56dB
Average = -56.7dB	

Table B.8: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 0.5m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
1.0m	-64dB
1.0m	-56dB
1.0m	-60dB
1.0m	-64dB
1.0m	-57dB
1.0m	-62dB
1.0m	-60dB
1.0m	-55dB
1.0m	-56dB
1.0m	-62dB
Average = -59.6dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
1.0m	-59dB
1.0m	-60dB
1.0m	-72dB
1.0m	-71dB
1.0m	-55dB
1.0m	-59dB
1.0m	-59dB
1.0m	-64dB
1.0m	-56dB
1.0m	-58dB
Average = -61.3dB	

Table B.9: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 1.0m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
2.5m	-76dB
2.5m	-73dB
2.5m	-73dB
2.5m	-62dB
2.5m	-71dB
2.5m	-67dB
2.5m	-70dB
2.5m	-72dB
2.5m	-67dB
2.5m	-76dB
Average = -70.7dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
2.5m	-81dB
2.5m	-80dB
2.5m	-68dB
2.5m	-72dB
2.5m	-70dB
2.5m	-80dB
2.5m	-69dB
2.5m	-88dB
2.5m	-76dB
2.5m	-68dB
Average = -75.2dB	

Table B.10: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 2.5m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
5.0m	-74dB
5.0m	-81dB
5.0m	-80dB
5.0m	-81dB
5.0m	-94dB
5.0m	-88dB
5.0m	-97dB
5.0m	-90dB
5.0m	-87dB
5.0m	-96dB
Average = -86.8dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
5.0m	-84dB
5.0m	-85dB
5.0m	-86dB
5.0m	-96dB
5.0m	-97dB
5.0m	-73dB
5.0m	-86dB
5.0m	-88dB
5.0m	-88dB
5.0m	-91dB
Average = -87.4dB	

Table B.11: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 5.0m

(a) Wi-Fi Off

BQ - WI-FI OFF	
Distance	**RSSI**
7.5m	-72dB
7.5m	-93dB
7.5m	-88dB
7.5m	-93dB
7.5m	-88dB
7.5m	-95dB
7.5m	-92dB
7.5m	-93dB
7.5m	-97dB
7.5m	-94dB
Average = -90.5dB	

(b) Wi-Fi On

BQ - WI-FI ON	
Distance	**RSSI**
7.5m	-89dB
7.5m	-87dB
7.5m	-87dB
7.5m	-94dB
7.5m	-94dB
7.5m	-89dB
7.5m	-92dB
7.5m	-94dB
7.5m	-88dB
7.5m	-93dB
Average = -90.7dB	

Table B.12: User number two bracelet (D0:39:72:A7:D0:C3), BQ, 7.5m

References

[1] Bluetooth SIG. Bluetooth fast facts, December 2014. http://www.bluetooth.com/Pages/Fast-Facts.aspx.

[2] Bluetooth SIG. Bluetooth history, December 2014. http://www.bluetooth.com/Pages/Bluetooth-Smart.aspx.

[3] Joseph Decuir. Bluetooth 4.0: Low energy, February 2015. http://www.californiaconsultants.org/download.cfm/attachment/cnsv-1205-decuir.pdf.

[4] R.A. Ramlee, D.H.Z. Tang, and M.M. Ismail. Smart home system for Disabled People via Wireless Bluetooth. In *2012 International Conference on System Engineering and Technology (ICSET)*, pages 1–4, September 2012. doi:10.1109/ICSEngT.2012.6339347.

[5] J. Mander and D. Picopoulos. Bluetooth piconet applications. pages 1–25, 2002. http://www.ee.ucl.ac.uk/~afernand/Example1.pdf.

[6] StickNFind. Ultra-small bluetooth gps locator, February 2015. https://www.sticknfind.com/BluTracker/.

[7] linquet. linquet - prevent the loss of your phone and valuables, February 2015. https://linquet.com/.

[8] xyfindit. Xy find it - keep track of everything that matters, February 2015. http://www.xyfindit.com/.

[9] findemtracking. Findemtracking - never lose your belongings again, February 2015. http://findemtracking.com/.

[10] S. Alghamdi, R. Van Schyndel, and A. Alahmadi. Indoor navigational aid using active RFID and QR-code for sighted and blind people. In *2013 IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 18–22, April 2013. doi:10.1109/ISSNIP.2013.6529756.

[11] S. Alghamdi and R. van Schyndel. Accurate positioning based on a combination of power control and signal strength indicator using active RFID system. pages 1–5. School of Surveying and Geospatial Engineering, 2012. https://researchbank.rmit.edu.au/view/rmit:19466.

[12] Jorge Neiva. "localização e orientação "indoor" com recurso à tecnologia rfid". Master's thesis, Universidade do Porto, Julho 2012.

[13] Nuno T. Almeida and Vitor Pinheiro. Alarm System and Trajectory Correction in Paralympic Athletics Races. *Procedia Technology*, 17:170–177, 2014. http://www.sciencedirect.com/science/article/pii/S2212017314004617. doi:10.1016/j.protcy.2014.10.225.

[14] A. Pantelopoulos and N.G. Bourbakis. A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 40(1):1–12, January 2010. doi:10.1109/TSMCC.2009.2032660.

[15] N.K. Suryadevara and S.C. Mukhopadhyay. Wireless Sensor Network Based Home Monitoring System for Wellness Determination of Elderly. 12(6):1965–1972, June 2012. doi:10.1109/JSEN.2011.2182341.

[16] Jim Lansford, A. Stephens, and R. Nevo. Wi-Fi (802.11b) and Bluetooth: enabling coexistence. *IEEE Network*, 15(5):20–27, September 2001. doi:10.1109/65.953230.

[17] N. Golmie, O. Rebala, and N. Chevrollier. Bluetooth adaptive frequency hopping and scheduling. In *2003 IEEE Military Communications Conference, 2003. MILCOM '03*, volume 2, pages 1138–1142 Vol.2, October 2003. doi:10.1109/MILCOM.2003.1290352.

[18] Salviano Soares Sérgio Silva. Anechoic Chamber Interference Tests on a Bluetooth Low Energy Front-End Towards a Subcutaneous Glucose Sensor. 2014.

[19] J. Decuir. Introducing Bluetooth Smart: Part 1: A look at both classic and new technologies. *IEEE Consumer Electronics Magazine*, 3(1):12–18, January 2014. doi:10.1109/MCE.2013.2284932.

[20] Bluetooth SIG. Bluetooth medical, December 2014. http://www.bluetooth.com/Pages/Medical.aspx.

[21] Bluetooth SIG. Bluetooth sports, December 2014. http://www.bluetooth.com/Pages/Sports.aspx.

[22] Hao Yan. An open source inertial sensor network with bluetooth smart. Master's thesis, University of Toronto, May 2014.

[23] Bluetooth SIG. Bluetooth basics, December 2014. http://www.bluetooth.com/Pages/Basics.aspx.

[24] P. Premprayoon and P. Wardkein. Topological communication network design using ant colony optimization. In *The 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005*, volume 2, pages 1147–1151, 2005. doi:10.1109/ICACT.2005.246163.

[25] Jin-Shyan Lee, Yu-Wei Su, and Chung-Chou Shen. A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. In *33rd Annual Conference of the IEEE Industrial Electronics Society, 2007. IECON 2007*, pages 46–51, November 2007. doi:10.1109/IECON.2007.4460126.

[26] Bahareh Gholamzadeh and Hooman Nabovati. *Concepts for Designing Low Power Wireless Sensor Network*. 2008.

[27] Bluetooth SIG. Bluetooth history, December 2014. http://www.bluetooth.com/Pages/History-of-Bluetooth.aspx.

[28] Android. Android jelly bean, January 2015. http://www.android.com/versions/jelly-bean-4-3/.

[29] Microsoft Community. Windows phone ble, January 2015. http://developer.nokia.com/community/wiki/Introduction_to_Bluetooth_support_on_Windows_Phone_8.

[30] Priya Viswanathan. How to choose a good cheap bluetooth headset, 2015. http://www.newgadget.org/mobile-phones/how-to-choose-a-good-cheap-bluetooth-headset/.

[31] Connecting Business. Parrot car kit, February 2015. http://www.connectingbusiness.com/products/accessories/parrot/ck3100.html.

[32] Adafruit. Adafruit nrf80001, February 2015. https://learn.adafruit.com/getting-started-with-the-nrf8001-bluefruit-le-breakout/introduction.

[33] Adafruit. Adafruit website, February 2015. https://www.adafruit.com/.

[34] Arduino. Arduino uno, February 2015. http://arduino.cc/en/Main/arduinoBoardUno.

[35] IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pages 1–1076, June 2007. doi:10.1109/IEEESTD.2007.373646.

[36] Android Hive. Android login and registration with php, mysql and sqlite, March 2015. http://www.androidhive.info/2012/01/android-login-and-registration-with-php-mysql-and-sqlite/.

[37] FEUP. Gnomo feup server, April 2015. http://paginas.fe.up.pt/~ee10281/dissertation/.

[38] Ivo Rocha. Android application, April 2015. https://dl.dropboxusercontent.com/u/18469248/Dissertation_Android.apk.

[39] Android Developers. Transmitting network data using volley, April 2015. https://developer.android.com/training/volley/index.html.

[40] Android Developers. Vibrator, May 2015. http://developer.android.com/reference/android/os/Vibrator.html.

[41] Android Developers. Wifiinfo, April 2015. http://developer.android.com/reference/android/net/wifi/WifiInfo.html/.

[42] Feup. Bluetooth low energy, March 2015. http://developer.android.com/guide/topics/connectivity/bluetooth-le.html.

[43] Android Developers. Bluetoothadapter.lescancallback, March 2015. https://developer.android.com/reference/android/bluetooth/BluetoothAdapter.LeScanCallback.html.

[44] The Android Open-Source Project. Device scan actitivity, April 2015. https://android.googlesource.com/platform/development/+/cefd49aae65dc85161d08419494071d74ffb982f/samples/BluetoothLeGatt/src/com/example/bluetooth/le/DeviceScanActivity.java.

[45] Donly. Ble checker, March 2015. https://play.google.com/store/apps/details?id=com.magicalboy.btd&hl=pt-PT.

[46] Android Developers. Android emulator, March 2015. http://developer.android.com/tools/help/emulator.html.

[47] Nordic Semiconductor ASA. nrf uart 2.0, March 2015. https://play.google.com/store/apps/details?id=com.nordicsemi.nrfUARTv2.

[48] Adafruit. Adafruit nrf80001, March 2015. https://github.com/adafruit/Adafruit_nRF8001.

[49] Android Developers. Log, April 2015. http://developer.android.com/reference/android/util/Log.html.

[50] Feup. Gnomo phpadmin, April 2015. http://db.fe.up.pt/phpmyadmin.

[51] Tutorials Point. Jdbc - sample, example code, May 2015. http://www.tutorialspoint.com/jdbc/jdbc-sample-code.htm.

[52] Oracle. Class treemap<k,v>, May 2015. https://docs.oracle.com/javase/7/docs/api/java/util/TreeMap.html.

[53] MySQL. Download connector/j, May 2015. http://dev.mysql.com/downloads/connector/j/.