Using ubiquitous computing for preventing risk behaviors based on smart contexts through mobile devices: an approach for nutrition and sexually transmitted diseases

Felipe Andres Besoain Pino
Faculty of Computer Science, Multimedia and Telecommunication
&
Internet Interdisciplinary Institute (IN3-UOC)
Universitat Oberta de Catalunya

A thesis submitted for the degree of
Doctor of Network and Information Technology

2018

This thesis is dedicated to
Brianna Hill and Lucas Besoaín Hill
for their unconditional support on this journey.

Acknowledgements

First, I would like to thank my supervisor Dr. Antoni Perez Navarro for all the support, encouragement and advice that I received from him during the time I have been doing my research.

Thanks to the Internet Interdisciplinary Institute (IN3) for having me in my international research internship in Barcelona on 2016.

I am very and forever grateful of my wife, Brianna Hill, for her constant support and encouragement to pursue and finish this challenge. I also would like to mention my son Lucas, who was born during this process and is a constant source of delight and inspiration.

Abstract

Several methods have been utilized recently to educate people and communities about topics related to worldwide health issues, among these, nutrition and sexual behaviors.

In the area of nutrition, excessive weight and obesity are problems associated with a variety of health issues. These problems are directly related to the current lifestyle of the population. People have fast paced lives, constantly on the run, with little time to devote to buying and preparing healthy meals. In addition, they receive limited education or information about nutrition.

On the other hand, regarding sexual behaviors, advances in the development of information and communication technologies have facilitated social interrelationships, but also sexual contacts without appropriate preventive measures.

This work explores how to develop educational mechanisms for promoting healthy behaviors, focusing on a target group and their context in two areas: 1) bad nutrition (related to obesity or being overweight), and 2) sexual transmitted infections, which were chosen due to their relevance and importance in the area of public health. The aim of this work is to investigate how ubiquitous computing could be useful in preventive health for these problems.

Two research strategies were used. The first strategy corresponds to the software development, while the second is a methodology to validate results. The first strategy used was the iterative development of software.

The second strategy to validate results was the design and creation strategy. Thus, through a combination of these methodologies, mobile applications and their software artifacts were created and then validated through interviews and quizzes. Four applications (UBIAPP, UBINUT, GEONUT and UBESAFE) were developed to address the issue of using mobile applications with preventive health messages as a part of two stages; the first stage consisted of the test applications (UBIAPP and UBINUT), while the second stage corresponded to the refined applications (GEONUT and UBESAFE). The first application to be developed was UBIAPP, which addressed the issue of STI and HIV prevention, focusing on testing preventive messages in a smart context with a group of sexually active men. The second application, UBINUT was designed for the prevention of risky behavior in relation to nutrition problems, considering the perception of the users when they receive a health message. After the testing of these initial applications, the third application, GEONUT was developed as the refined version of UBINUT and it included the addition of concepts of geofencing for the recognition of hot zones associated with places where users can make poor nutritional decisions, such as food courts, cafeterias, street food, etc. Finally, UBESAFE, the refined version of UBIAPP, was created to focus on testing and trying preventive messages in a smart context with a group of MSM, including a gamification approach for sharing data that was not present in the test application (UBIAPP). This refined mobile application considers the previous experience with the last three mobile applications (UBIAPP, UBINUT, and GEONUT) from an informatics point of view and also the experiences of each test carried out with each respective target group.

The results of this work suggest that ubiquitous computing may be useful for alerting users with preventive and educational messages, especially when they are non-intrusive. The proposed applications are non-intrusive because: 1) the users themselves decide to install them; 2) they send messages that help users think about taking appropriate preventive measures; and 3) they work in the background without interfering with users unless a trigger situation is detected. Thus, this type of application could become an important tool in the complex task of promoting healthy behaviors in both areas: nutrition and sexual transmitted infections.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, the motivation behind this work will be presented, along with a list of publications related to the research that has been done on this topic and, finally, a short overview of the organization of this document.

1.1 Motivation

In the face of various health issues, diverse prevention strategies can be used to respond, with varying degrees of success. Some prevention strategies concern themselves with the identification of at-risk groups, such as: drug users, smokers, obese individuals, carriers of infectious diseases, HIV patients, malnourished people, etc. These interventions are aimed at persuading those within these groups to change behaviors or lifestyles, or at least, to control those behaviors in order to reduce the harmful effects that have been noted. Therefore, the main assumption that underlies many preventive health campaigns is that individuals' *risk behavior or conduct* becomes the primary ***risk*** for their health (Schiller et al. [1]).

In the last few years, two groups have become especially relevant: those individuals with sexually transmitted infections such as HIV infection (Finlayson et al. [2], Song et al. [3]), and those that are overweight and/or obese. With regard to HIV, its spread is directly related to the current globalized world and lifestyle (González et al. [4]). Social networks, just like specialized software for making social contacts, contribute to increase different opportunities for infection (Rietmeijer and McFarlane [5], Levine

[6]) since they raise the number of interactions with groups of people looking for sexual partners in this way. This situation is fostering different risk behaviors (Finlayson et al. [2], Song et al. [3], Margolis et al. [7]). Although sexual contacts are not risky by themselves, risk behaviors could include, for example, unprotected sex with one or multiple partners, drug use, alcohol use, etc.

With regard to nutrition, 65% of the world population lives in countries where diseases associated with being overweight and obesity have a mortality rate greater than diseases related to malnutrition (World Health Organization [8]). Fast-paced lifestyles and lack of nutrition education lead to poorly balanced diets and increase the risk behavior of a population. Several risk behaviors drive to obesity and overweight such as the temptations of fast foods that are high in fat and sugar, social pressure, lack of time for physical activities, among others (bringing as a consequence *risk* for obesity). There are several studies that currently address prevention through Information and Communications Technology (ICT), in order to be able to educate and prevent these behaviors (Burgess-Champoux [9], Hingle et al. [10], Sullivan et al. [11]).

Currently, various strategies exist for health education and prevention. These strategies play a fundamental role in societies' well-being, and consequently, they are a constant concern for governments, health ministries, medical centers, etc. (Herrick [12]).

The number of health campaigns has multiplied along with the amount of locations for preventive action (Petersen [13]). Currently, diverse methods are used in health education and prevention, among which it is important to note: advertising campaigns, group work and the development of healthy habits with at-risk groups, advertising control at a national and regional level of different products that have been characterized as unhealthy (tobacco, alcohol, high-fat foods, etc.), the use of social networks, and school programs, among others (Lohse [14]).

A need exists for implementing new education and prevention systems that are applied and related to people's daily life. ICT is deeply bound to the lives of its users, especially through mobile devices (Mosa [15]), given the connectivity of these and the omnipresence of information (Weiss [16], Bell and Dourish [17], Kaplan [18]). It is through this type of omnipresence that the hope is to find an entry point to be able to educate and promote healthy behavior based on the context of targeted users, considering that mobile devices accompany their users through their daily routine, and therefore they are present at opportune moments to avoid risk behavior and/or educate users about the risks of a specific action.

In this work, the focus of the discussion is based in how to detect these behaviors and promote healthy conduct, and how mobile devices can help to prevent HIV and obesity without restricting decisions or interfering with users' privacy.

1.2 List of publications

The list of publications associated with this work is as follows:

- Felipe Besoain, Antoni Perez-Navarro, Felipe Ojeda, Jose Antonio Reyes-Suarez: Promoting Healthy Nutrition Behavior Using Mobile Devices and Ubiquitous Computing. Lecture Notes in Computer Science 11/2015; 9455(2015). DOI:10.1007/9783319264103_9

- Felipe Besoain, Antoni Perez-Navarro, Joan A Caylà, Constanza Jacques Aviñó, Patricia García de Olalla: Prevention of sexually transmitted infections using mobile devices and ubiquitous computing. International Journal of Health Geographics 05/2015; 14(1). DOI:10.1186/s129420150010z

Figure 1.1 shows an infographic diagram of this work, that includes the hypothesis, methodology and the four developed applications with their relationship. For further details, see Section 4.

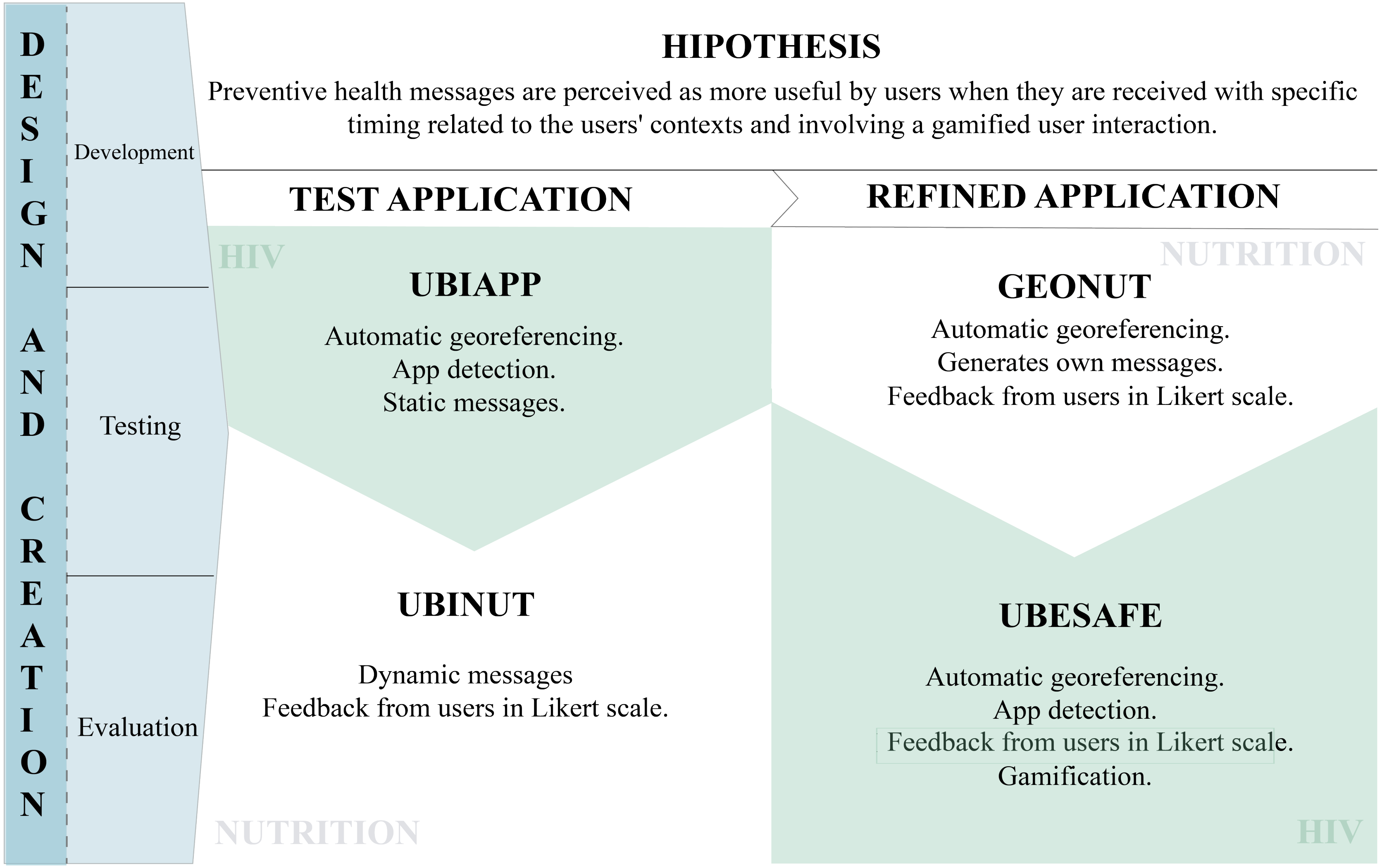


Figure 1.1: Infographic diagram of this work (hypothesis, methodology and IT artifacts)

1.3 Document organization

Chapter 2 provides the state of the art of HIV infection, nutrition and gamification topic that are relevant to understand further parts of this work. Chapter 3 presents the methodology with the design of the research and development of IT artifacts. Chapter 4 exhibits the four IT artifacts tested and developed with their functional and technical description. Chapter 5 presents the conclusions of this work and ideas for future research.

Chapter 2

State of the art

The use of ICT in education, advertising and health education has been much studied in the last few years. Below, the state of the art will be presented in the two lines of work of this project: 1) infectious diseases and in particular HIV infection and 2) bad nutrition.

2.1 Prevention of sexually transmitted infections

Studies from 2005 such as (Rideout et al. [19]), show how the youth of North America interact with ICT, citing relevant statistics such as 93% of young people are in front of a computer a minimum of 44.5 hours a week. However, with the introduction of smartphones to the market and now the successful addition of tablets, information is increasingly ubiquitous and, as a consequence, the access to applications and social networks online is more ubiquitous (Shim [20], Martin [21]). Two premises should be highlighted: 1) today, young people have grown up surrounded by technology; and 2) mobile devices and the internet are an essential part of the users' lives. Finally, in the same context, in (Levine [6]), researchers examine the principal tools that are currently used for preventing the spread of HIV, such as online and mobile applications, games, and social media, among others.

The study (Margolis et al. [7]) showed through an online survey with users of an internet site for men that have sex with men (MSM), that the majority had not been tested for infectious diseases. The results from the 8,040 participants allowed researchers to identify user groups that had never been tested, mainly, young people (between 18 and 24 years old) and residents of rural areas or places with a low population. Their reasons for not having been tested are: they thought that they were not infected (24% of young people) and they did not know where they could get tested (25% of young people).

Using the internet and ICT as a medium, the study (Bull et al. [22]) considers the problems related to social networks for young people. The study was completed with 1,588 participants, who were chosen randomly for a test on the effectiveness of the prevention of HIV on the social network Facebook. In this study, different types of surveys were used to gather information from followers of the movement Just/Us, embracing different aspects of confidentiality and validating the behavior of youth. This is consistent with the results from (Rietmeijer and McFarlane [5]), where it is shown that the internet can currently be seen as a high risk environment, where individuals can find sexual partners, among which there may be those that potentially have a sexually transmitted infections (STI).

This conclusion is particularly relevant since it lines up with the problem of preventing infectious diseases and shows a possible requirement or solution that has not been developed in-depth.

Additionally, and focusing now on mobile devices, in (Muessig et al. [23]), different applications are identified and evaluated that are available on iTunes [1] and Android Google Play [2], for the prevention and care of HIV and other STIs. Each application was downloaded, tested and tried out to evaluate its usefulness in the following areas:

- Knowledge of HIV and STIs.
- Decrease in risk through safe sex.
- Condom promotion.
- Information about HIV and STI tests.

---

[1] `http://store.apple.com/us`

[2] `https://play.google.com/store`

- Availability of resources for those HIV positive.
- Focus on key interest groups.

One interesting result is that in May 2012, 1,937 applications were identified. Of these, 55 met the inclusion criteria (12 for Android, 29 for iPhone, and 14 for both platforms). The inclusion criteria were: the application included content about HIV and STIs, was available in English, and was not designed for medical professionals, among others.

Finally, as is detailed in (Muessig et al. [23]) the majority of the efforts have been directed towards prevention through the use of applications that provide information when facing specific situations and behavior. Nevertheless, as far as we know, no contributions have been shown that develop and use in-depth the abilities and characteristics of mobile devices along with their omnipresence for the education and prevention of risk behaviors in this area. Muessing (Muessig et al. [23]) suggests that researchers in public health should work with application developers in order to incorporate innovative elements, starting with interventions that reduce the risk and the associated behaviors, as well as improve the inclusivity and interactivity of the applications.

2.2 Bad nutrition

Obesity has been categorized as an epidemic by the World Health Organization (WHO) ([24], Caballero [25]). Thus, one goal is being able to control it, and in this context, a range of studies have been carried out in this area. In (Hill et al. [26]), it is suggested that research should focus directly on producing small changes in life styles, diets and physical activity. More specifically, for food, it is necessary to change the composition, portions, etc., whereas for physical activity, it is necessary to focus on creating a favorable environment regarding transportation, physical activity in high schools, university and work, as well as creating areas and awareness about how people spend their free time. These small changes can produce a major change in an individual's life style (Hill et al. [26]).

In order to achieve these small changes, ICT has been used as a medium to reach the target group, specifically: courses online and mechanisms for spreading information, social networks, virtual reality, and mobile devices.

In (Cohen et al. [27]), a study was carried out regarding how online courses can be designed to transfer knowledge and at the same time educate in the same way that face-to-face courses do, delivering the component of social privacy through the internet. The proposal is to use new technologies in order to create healthy marketing and communicate nutritional concepts. In the same line, (Harvey-Berino et al. [28]) explores the feasibility of an online program for controlling behavior in weight management. The study was done at two universities in North America with a total of 336 individuals, showing that weight gain is common in university students. Few resources are available for addressing weight loss or behavioral changes in this group. The majority of the interventions of this type are academic courses associated with credits or group sports. However, these interventions do not focus on changing behavior on transferring techniques for changing behavior in weight gain. The use of programs through the internet helps students to establish healthy eating and exercise behaviors, which contributes to achieving or maintaining a healthy weight. The use of social networks, or what is called online social networking, has been categorized as the third most typical daily activity (Duggan and Brenner [29]). It is shown that the heaviest users of these networks, such as Facebook and LinkedIn, are young adults between 18 and 29 years old (Madden [30]). In (Lohse [14]), it is suggested that the use of social networks is based on the fact that these networks are not limited by money, education or geography. In this study, the use of Facebook was evaluated among low-income individuals, concluding that it has been effective in different target groups.

Virtual Reality has also been an element to consider in promotion and prevention material for nutrition. In (Sullivan et al. [11]), a comparison of weight conservation was carried out between two programs: one in-person and the other with virtual reality (through the internet). One of the benefits of the virtual practice is that many everyday problems are avoided, such as: conflicts because of travel, time between work and home, looking after the children, loss of anonymity and/or confidentiality issues, etc. The loss of weight was greater in the face-to-face program; however, the conservation of weight was greater through the virtual reality program.

Mobile devices have appeared recently as a good mechanism for promoting and spreading information. Adolescents are the main users of mobile phones and text-messaging. In fact, in the United States 75% of young people between the ages of 12 and 27 have a mobile phone, and close to half of them send 50 or more text messages every day (Lenhart et al. [31]).

In (Hingle et al. [10]), the use of mobile phones was studied for prevention in the area of nutrition through text-messaging. The use of text messages is limited to 160 characters, and it was critical for health researchers to understand how a text message can influence adolescents in knowledge, attitudes and behaviors related to diet and physical activity. Two questions were explored:

- What popular technology will be an acceptable means for adolescents for receiving messages that promote a healthy life style (for example, diet and physical activity)?
- Will involving young people in the development of a series of messages result in these messages being more relevant to their life styles and easy to understand?

Finally, it has been concluded that text messages can be successfully sent to users, since the ubiquity of mobile phones make them a tempting and friendly way to promote healthy behavior for adolescents. Not yet defined are the research opportunities for studying the scalability of the notifications to the users and how these influence their behavior, attitude and education.

In the same context, (Rodrigues et al. [32]) presents an application for mobile devices that permits users to register their daily food intake and physical activity, with the ultimate goal of promoting a healthy diet. This application incorporates the usefulness of sharing information with social networks in order to give users the option of sharing their achievements. The users valued the simplicity and usability of this application to be able to enter their information, as well as the interaction with social networks.

In the last year, the number of studies using ICT as a medium for prevention and education in nutrition has increased, as using mobiles has created new opportunities (Hingle et al. [10]). Nevertheless, no contributions have been registered that develop and use in-depth the abilities and technical characteristics of mobile devices, along with their ubiquity, for prevention and education in this area.

2.3 Gamification as a tool

Games are found in various parts of life: in school, work, business, and even in health, an area that has shown significant growth in the use of games as tools during the last few years.

Today, quickly and in continual development, games are popular entertainment tools for all types of people, independent from gender or age. Many games require users to learn and acquire complex abilities, and for this reason promote the development of superior cognitive processes and abilities (Huang and Soman [33]).

Games can also be used outside the field of entertainment. Gamification can be defined as the application of game concepts and mechanics to different contexts, such as crowdsourcing, social networks, loyalty programs, marketing, education, and healthcare, among others. Gamification has become a popular strategy to encourage specific behavior and increase the motivation and dedication of, for example, employees, students, and patients, to carry out tasks or activities considered serious. Thanks to gamification, the use of game design elements in contexts unrelated to games (Deterding et al. [34]), along with recent progress in information technologies, ubiquity, social networks and mobile devices, tools that facilitate daily tasks have been made more available. In summary, gamification is taking things that are not games, and trying to change them so that they feel like games.

Games are designed to be entertaining rather than useful, where desired states of experience are produced and users are motivated to continue participating in an activity intensely and for an unprecedented amount of time. Therefore, through gamification, designers of products, services or applications that are not games can attract users by applying game concepts; this way, users participate on structured platforms with game mechanics and dynamics, thus making the experience more enjoyable, motivating and/or attractive (Deterding et al. [34]), and improving the overall user experience (UX). These game concepts include prizes (Gené [35]) (for instance, points and virtual medals) and seek to satisfy users' needs or desires for recognition, achievement, collaboration, self-expression, and competition. These elements try to motivate users towards a particular objective. Some examples of these prizes can be found in video game console networks, including Xbox and Playstation, among others, but also in many popular games like chess or sports.

To conclude, gamification should be treated as a powerful tool (Escribano [36]), which has evolved thanks to the use of new digital information and communication technologies. It uses gameplay thinking and mechanics in non-game contexts, with the aim of having people adopt specific behavior, thus successfully promoting a particular change in users.

Smartphone use

The spread of information and communication technology (ICT) in the population has increasingly grown, in particular mobile phone technology. Smartphone technology continues to be the principal motor of growth for ICT, especially in developing countries where the spread of mobile media was greater than 50% in 2009. Today, more than 70 economies around the world have exceeded cell phone penetration rates of 100%, and developed countries averaged 113% by the end of 2009 ([37]). However, ubiquity and other aspects of smart mobile technology (smartphone, tablet) have made gamification an appropriate strategy that has already become an important component of many mobile service offers, since, as it is known, companies seek to improve client satisfaction, commitment and retention.

Smart devices, like the smartphone, are useful channels for lending services since they have characteristics that are used for user-game interaction, such as GPS, accelerometers, and external sensors that allow the measurement of movement and distance. In addition, they are generally in the users' pockets, and thus accompany them at all times.

In the area of health, the implementation of gamification strategies in medical procedures and with patients can foster, change and produce healthier behaviors and habits in the population (Gómez et al. [38], Lister et al. [39]).

Concept applications in health

Gamification can also be applied to the area of health. It has been estimated that 60% of health initiatives in the workplace now include gamification elements, a tool that has already become a 2.8 billion dollar industry by 2016 (Lister et al. [39]). In particular, smartphones have recently become an area of interest for carrying out behavioral interventions related to health through gamification strategies (Mendiola et al. [40], Boulos and Yang [41], Giosan et al. [42]), since they have the potential to improve quality of life.

Mobile technology is very promising as a medium for promoting healthy behavior, given that it offers a great number of sensor technologies and data visualization tools that allow detected information ubiquitously to be stored, analyzed and communicated. One form of improving quality of life is through health education, that is, any combination of planned learning experiences designed to facilitate voluntary changes in health behavior (Alias [43]). This is carried out through applications that promote rewards for physical activity, with positive consequences for physical and mental health. The following elements are ways to improve physical and mental health: identification and consumption of healthy food, monitoring the taking of medication, prevention of infection of diseases such as HIV, self-monitoring for diabetes, and health assets that help improve physical and emotional conditions through a desire for self-improvement and optimism, as is the case of the application Superbetter ([44]). Other applications include controlling depression or increasing higher levels of chemotherapy in the blood and higher indexes of the use of antibiotics for patients with cancer, such as in the application Re-Mission ([45]). These examples involve behavior modification which can be addressed through gamification strategies, reducing the risk of illness (Giosan et al. [42], Brauner et al. [46], Giota and Kleftaras [47]), and promoting better or stable life quality.

Games and gamification in nutrition

Mobile games and applications can have a large impact on nutrition and health behavior. A recent example is the game application Pokemon Go; although it was designed to entertain rather than to promote healthy behavior, studies have shown that its use has encouraged physical activity among participants, since it embeds the game in the physical world. In a study in the United States, it was found that over a period of 30 days Pokemon Go increased users' activity on average by 1473 steps a day, an increase of more than 25% in comparison with their previous activity level (Wong [48]).

Other applications have been specifically designed to apply gamification in nutrition. For example, easy access to treatment and support with behavioral self-monitoring can be important for decreasing bad nutrition habits; in order to achieve this end, gamified mobile applications on smart mobile devices have given users control over the consumption of foods and counting calories, as well as control over nutrition profiles, thus adopting healthier life styles. Proof of this is seen in different studies that have analyzed gamified mobile applications, providing information about the effectiveness of these applications' behavioral interventions for users' health with respect to young adult nutrition (Dennison et al. [49], Hebden et al. [50]).

On the other hand, infant nutrition has also been an area of study for the introduction of gamification. Breastfeeding is universally recognized as the ideal way for babies to receive appropriate nutrition, offering many health benefits as much for the mother as for the baby. Nevertheless, the father's influence has been identified as an important fact that influences the mothers breastfeeding behavior. It is in this way that (et al White, B.K. [51]) addresses the father's role in infant nutrition through gamification, developing a mobile application for men about breastfeeding, Milk Man (mil [52]), in order to increase the support that the father can offer.

As has been previously mentioned, bad nutrition causes different types of diseases; however, in addition, diseases found in patients must be treated with special care, through constant physical activity, diet appropriate to the illness, along with diet supplements, pancreatic enzymes, multiple daily vitamins, and daily tracking of carbohydrates, among other factors. This is the case for patients with cystic fibrosis, who must deal with this chronic illness with complex treatment and fragile life quality. Here, (Hilliard et al. [53]) through gamification strategies in mHealth (or mobile health), emphasizes design recommendations for smartphone that encourage self-regulation in the guidelines that must be followed to face a disease, for example, diabetes and HIV.

Gamification in the prevention of HIV

Antiretroviral therapy consists of medications that treat HIV, three antiretroviral drugs that stop the viral replication so that the viral count becomes imperceptible in more than 70% of the cases (GarcÃa O and Olea N [54]), qualitatively and quantitatively recovering the immune response and reducing the associated morbidity, progression and mortality rate of HIV. This therapy must be maintained for life, and it is necessary to frequently make changes in the treatment due to drug toxicity or the appearance of viral resistance. Facing this type of treatment, it is possible to introduce gamification strategies since, as we have already seen, these strategies are becoming even more used to address psychological factors and behaviors associated with the medical treatment regime. Gamified applications, directed towards goals, immersion, challenges and motivation, are used to improve attitudes self-efficacy to change health behaviors. According to (LeGrand et al. [55]), studies and developed applications incorporate new mechanisms based on games and gamification elements. Related to an intervention approach, these include functions for behavior monitoring as well as interactivity through games, points, rewards and social interactions, all designed to improve the adherence to antiretroviral therapy among young HIV positive men that have sexual relationships with men. These men represent 72% of new HIV infections in the United States (LeGrand et al. [55]). That is, given that the treatment is for life, many young people, once diagnosed with the infection, are less likely to keep up the treatment. As a result, tools or interventions are needed to improve their individual health, motivating them to adhere to the treatment. Therefore, smartphones market and gamification can be used to achieve this end, resulting in a highly suitable tool. In this same context, the application BeYou+ can be found, a gamified App designed for HIV patients with the phrase "Be Inspired, Be Healthy, Be Well...BeYou+". This application was created by the Chelsea and Westminster Hospital NHS Foundation Trust to supply real information to users about their body. This information included viral count graphics, a list of medications, the names of health professionals, a protected password, and weekly videos rewarding the achievement of goals and daily notifications.

Another way of contributing in this area is through the prevention of HIV infection. Given that gamification tries to motivate, challenge, teach, raise awareness and impact society to accomplish and influence change in user behavior, it can be used to prevent diseases such as HIV-AIDS. A great deal of information exists so that HIV patients can assimilate and understand that they must monitor their health. However, a large number of different tools also exist for the prevention of this disease and so that the number of infections does not continue to grow. Among these are smartphone applications such as HIV Risk Calculator, an app for people that have had high risk sexual relations and are worried about the possibility of having contracted HIV. Inside the prevention framework, another application can be found that focuses on adolescents and young people to motivate the use of condoms. This application has gamification elements and was launched by Argentina's UNICEF with the aim of generating awareness about the importance of practicing safe sex. The application is an example of gamification that employs game dynamics to promote the acquisition of different habits and the achievement of objectives.

Advantages of gamification in health

Building on the framework of these ideas, as it has been pointed out, gamification spans different areas within the healthcare system thanks to existing technology. Consequently, some advantages can be indentified that contribute to gamification in health; these include the increase in patients commitment and motivation, who are prompted to participate in actions proposed by the gamified tool about their illness, through mechanisms that motivate the completion of these actions, increasing their role in the current situation with regards to the sickness. These game mechanics are suitable for self-representation through avatars, three-dimensional surroundings, competition, feedback, scoring, levels, rewards for achievements, team games, the use of social network elements, and time pressure, among others (Lister et al. [39]). Furthermore, they promote learning and education with respect to the users' health through games that are serious, but that capture the players' attention regardless of their age, and deliver a learning experience about a certain disease, its prevention or necessary behavior to improve life quality.

Gamification strategies reach all types of audiences, given that they have a wide spread, and can reach different people regardless of their age, or the context in which the strategies are used, addressing different aspects of life. Gamification used in applications allow for maximum dissemination when the users share the material through elements such as social networks, creating a viral gamification effect that reaches those most interested (Zichermann and Cunningham [56]).

2.4 Concluding remarks

As has been detailed in the previous section, there are diverse mechanisms and activities for education and prevention in the areas of nutrition and infectious diseases. It is worth noting that each one of these initiatives encompasses the following concepts:

- Adjustment of needs to the target group.

- Need of using the internet and IT devices as a medium for the distribution of information:
    - e - Learning.
    - Virtual Reality.
    - Social networks.
    - Information systems.
    - Etc.

All of these efforts have been made without considering the particular context in which the individuals make their decisions; furthermore, they require that the individuals are aware of the fact that they are working on modifying their behavior. It is for this reason that the proposed principal contribution is to be through ubiquitous computing and the detection of risk behaviors in order to prevent and educate at the exact moment that the users are most likely to be influenced. For example, currently, applications exist that use the identification of a product in the supermarket (through the name, bar code, etc.) to inform the users if the item is healthy or not. This requires that users are **conscious** of wanting to eat healthy in each **instance** in which they select a product. In the future, a ubiquitous interface could be used in tandem with users' sight, so that when users see a product, the application detects the selected food and in real time alerts them with information about the product; this, in turn, would have an impact on the architecture of shopping decisions without requiring that the people are conscious of wanting to identify the foods in order to modify their behavior.

The internet and mobile devices are capable of promoting the transmission or prevention of STIs, HIV, Tuberculosis, etc; this is applicable as much for the users as for the providers. The population's degree of access to ehealth tools is worrying, especially on personal computers, tablets, smartphones, etc. (Swendeman and Rotheram-Borus [57]).

People live in a network of connections: we check e-mails regularly, make phone calls from and to any location without any problem, and use cards to pay for everything from transportation to food. The majority of the actions can be tracked, even incorporating georeferencing technologies, the ability of collecting and analyzing massive amounts of information has given rise to what's known as: computational social science (Cioffi-Revilla [58]). This is the foundation that makes it possible to connect individuals' behavior to the technologies that they use; in other words, it is possible to apply the concept of *computational social science*, for the study and detection of behavior through ICT.

Chapter 3

Methodology

The current chapter introduces the hypothesis and the methodology that has been used to carry out the research. First, the hypothesis and methodology will be outlined in general terms. Then, a closer look will be given to the initial test applications, followed by the refined applications, that is, the modified versions based on initial feedback.

3.1 Hypothesis

The hypothesis of this work is: Preventive health messages are perceived as more useful by users when they are received with specific timing related to the users' contexts and involving a gamified user interaction.

3.2 Objectives

The general objective based on the hypothesis is to study and develop a system that incentivizes health habits through the sending of health messages using mobile devices at the moment that a risk situation could take place.

This general objective is carried out through the following specific objectives:

- Know and define the situations that trigger a risk behavior in STIs and nutrition
- Develop a software for mobile devices with an Android operating system, with the aim to detect situations of risk and notify users about them

- Promote health behaviors through messages when people are more committed to have a risk behavior
- Develop a gamified system to increase user interaction
- Evaluate the IT solution in at least two countries

3.3 Methodology

In this work, two strategies have been used; the first strategy corresponds to the software development, while the second is a methodology to validate results. The first strategy used was the iterative development of software. The second strategy to validate results was the design and creation strategy. The combination of both approaches allows the hypothesis to be researched, taking into consideration the whole development lifecycle of an IT artifact. Thus, through these methodologies, IT artifacts were created and then validated through interviews and quizzes.

The development of the applications has two main phases:

1. Test application: This introduces users to the application and evaluates how to make it useful.
2. Refined application: The end application, where comments from the users are taken into account.

Taking into consideration the research hypothesis, it is important to highlight two main aspects:

1. Preventive health messages must be based on a specific context.
2. Users must have some gamified interaction with the IT artifact for enhancing their UX.

To prove and evaluate the hypothesis, two groups were considered: first, males who have sex with males (MSM) concerned about the HIV contagion; and second, people who are concerned about health and weight issues. These groups were taken into account due to the substantial increment of HIV contagion and obesity in the world. As a consequence of these two issues, not only are people developing health problems, but also the WHO and governments are constantly working and innovating in public preventive campaigns. These campaigns have the main objective of changing habits in order to reduce the number of people that have health problems.

3.3.1 Test applications: UBIAPP and UBINUT

For the design and development of these initial test applications, we considered the previously selected groups: the MSM related to the HIV transmission, and the people who want to lose weight due to nutrition and obesity problems. The application created for the first group was called UBIAPP and the application for the second group was named UBINUT. Therefore, each application (IT artifact) was designed and developed considering the following aspects:

- HIV mobile application (UBIAPP): automatic detection of points of interest (POIs) through a geofencing approach and the detection of risky applications. This application has minimum interaction with the users and prompts static messages from a database.
- Nutrition mobile application (UBINUT): dynamic health messages contextualized through a health professional. The application has light interaction with the users.

Once the test applications were developed following the iterative development of software, they were validated through surveys:

1. Testing with a target group: each mobile application was tested with a target group. The main goal of the testing is to generate data from the experience to be quantified and classified. Thus, the IT artifacts are used as a tool for learning about the impact of the ICT in the health prevention contexts.

2. Evaluation of test results: The results revealed that the main aspects to evaluate are: context, users' interactions and automatization of the process (detection of risk and messages). These aspects were considered in the design of the refined applications.

3.3.2 Refined applications: GEONUT and UBESAFE

Based on the results from the first testing, UBIAPP and UBINUT were both modified and improved to create the refined mobile applications, GEONUT and UBESAFE. So, in the second iteration, two areas were focused on:

1. Informatics optimization.
2. Functionalities for the users related to context, users' interactions and the automatization of the process (detection of risk and messages).

Once GEONUT and UBESAFE had been developed following the iterative development of software, the same validation stages seen previously were used:

1. Testing with a target group: each refined application was tested with a target group. The main goal of the testing is to generate data from the experience to be quantified and classified.
2. Evaluation of test results: The three main aspects that were evaluated are: context, users' interactions and automatization of the process (detection of risk and messages).

In conclusion, through the combination of two previously mentioned methodologies, it was possible to improve the test applications, UBIAPP and UBINUT, and create the refined versions, GEONUT and UBESAFE. Furthermore, through the validation of the refined versions, conclusions could be drawn about the effectiveness of using mobile applications for health prevention purposes and proposals could be made for future applications.

Chapter 4

Development of the software and results

In this chapter, an overview of the developed IT artifacts (test and refined applications) will be presented from a software engineering perspective focusing on the improvements and principal findings of them.

Firstly, a brief overview of the common requirements will be presented, functionalities, and attributes of the system. Then each developed IT artifact will be introduced with a detailed review of functional and technical aspects including the principal functionalities of the system, communications modules, and their interfaces, packages, and design patterns of the modules, among other artifacts of software. Furthermore, for each IT artifact, the testing of the system with a target group, results, and an analysis will be described. Finally, the fundamental issues and conclusions associated with the initial test and refined applications will be presented and reviewed.

4.1 Common requirements

In the stage of development of the methodology Design and Creation, the strategy of research of design and creation will consider a complementary methodology for the development of software. The main strategy of development of software used is the iterative method, through this process, the specifications, attributes, and requirements will be described and developed that are necessary for the system development of functional prototypes in each iteration. The definition of the requirements of the system describes what the system should do.

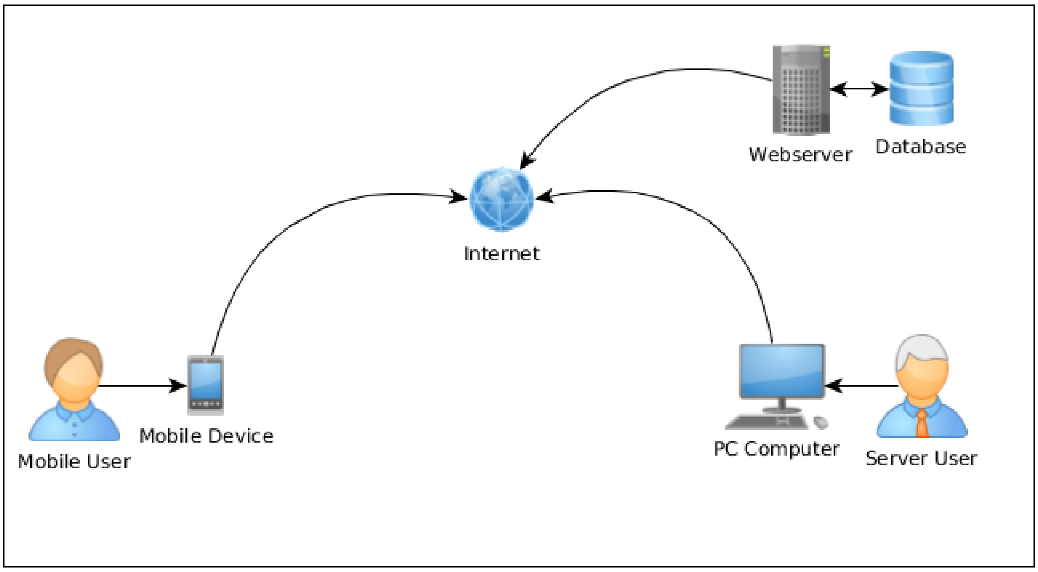


Figure 4.1: General scheme of the developed system

Figure 4.1 represents a general vision of the components that are involved in the developed system. The system consists of the three elements seen in the figure, that is, the webserver, the mobile device and the PC computer. All these components interact through the internet. Furthermore, each element interacts with a specific user. For instance, the application users interact with the mobile devices and are defined as the mobile users, whereas administrators interact with the PC computers and are defined as the server users. Finally, the webserver interacts with the database as an interface for providing stored information to the mobile device and PC computer.

To clarify the denomination of each component in this work, they will be referred to as:

- **System**: the whole technological solution that includes the application that runs on the mobile device and the web system that interacts with the PC computer (front-end and back-end of the web solution).
- **Mobile application**: corresponds to an Android application that is distributed through the Google Play app store and used on a mobile device. This application

is in charge of monitoring users' smart context to notify them when it is required.

- **Web information system**: corresponds to the front-end, back-end and their application programming interface (API) that is stored in the webserver and database. Through the front-end users can set and get information for or from the connected mobile device and PC computer.

4.1.1 Functionalities and attributes of the system

In addition to the common requirements outlined previously, another important aspect of the software development is the system's functionalities and attributes.

The attributes of the system are the characteristics and properties of quality that the system must have. There are several classifications of the requirements, one of them is the FURPS+ (Pressman, 2010), FURPS+ stands for Functional, Usability, Reliability, Performance, and Supportability. The quality factors support the requirements for their development. The plus (+) symbol refers to the restrictions of design and requirement of implementation at different levels: physical, interface, operability, and distribution. These aspects are considered in the development of the applications created in this work.

The Table 4.1 describes the general attributes of the system. These attributes were developed taking into consideration the FURPS+ standard previously described.

4.1.2 Test applications

As mentioned in the methodology, for the design and development of these initial test applications, two groups were considered: 1) MSM concerned about the HIV contagion, and 2) people who are concerned about health and weight issues. Therefore, two mobile applications (IT artifacts) were designed and developed, one for each target group, considering the following aspects:

- **HIV application(UBIAPP)**: known as UBIAPP, this mobile application sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, or their proximity

Attributes of the system		
Att1	Functional	The software should be able to interoperate among other subsystems trough a model client server. It should be capable of keep compatibility and portability to other systems.
Att2	Usability	(Border restriction) each action of the developed software must be at three click of distance (actions of the user and the interface).
Att3	Reliability	The personal information of the users must be available just for the owners and stored in a local database, not allowing access to third parties. Information should be available to the users at any time that it is requested.
Att4	Performance	The software must optimize the available resources of the mobile device. It should maximize the battery among the functionalities. (Border restriction) The information must be loaded in a time rate of maximum 10 seconds.
Att5	Supportability	The software developed must be modular and scalable at any time that the study or context requires.
Att6	Legal	The software must follow the legal requirements of the Free Software Foundation for licensing purposes

Table 4.1: Attributes of the system, these attributes were developed taking into consideration the FURPS+ standard

to areas with a high probability of intercourse (*hot zones*). The main focus was the automatic detection of points of interest (POIs) through a geofencing approach and the detection of risky applications. This application has minimum interaction with the users and prompts static messages from a database.

- **Nutrition application (UBINUT)**: known as UBINUT, this mobile application allows a health professional to send health messages through mobile devices. These messages are related to specific contexts and can be evaluated by the users according to their UX. The main focus was the dynamic health messages contextualized through a health professional. The application has light interaction with the users, thus improving the UX and dynamic feedback.

4.1.3 Refined applications

The refinements of the test applications have been driven by improving the three main aspects (context, users' interactions and automatization of the process). This iteration concentrates on two areas: 1) informatics optimization; and 2) functionalities for users (use cases) related to the three main aspects. The HIV mobile application known as UBIAPP was refined into a new system (web information system and a mobile application) known as UBESAFE. Furthermore, the nutrition mobile application known as UBINUT was refined into a new system known as GEONUT.

- **UBESAFE**: the UBESAFE system has been developed taking into consideration the findings from the UBIAPP test with users. The system sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, or their proximity to areas with a high probability of intercourse (*hot zones*). This application helps users to share messages and hotzones, thus creating a community that has access to updated information. To this end, the main experience was developed with *gamification* concepts.

- **GEONUT**: the GEONUT system has been designed taking into account the findings from the UBINUT test with users. The system sends preventive notifications to users when the mobile application detects their proximity to areas with restaurants and fast food spots that can provoke unhealthy habits in users. In addition, the system allows a health professional to send health messages based on a specific context through mobile devices.

A more specific schematic will be presented in the following sections that focus on each of the individual applications.

4.2 UBIAPP

In this section, the application called UBIAPP will be described. This application corresponds to the first test application designed in this work. The application UBIAPP was the first approach to address the issue of HIV prevention, focusing on testing and trying out preventive messages in a smart context with a group of MSM. The following structure presents the application, its development, testing, results, and an analysis of the results.

Firstly, UBIAPP will be presented and the general aspects with their functionalities will be described. Secondly, the technical aspects of the application will be discussed with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Thirdly, the context of the testing with a target group will be established. Then, the results will be presented.

Finally, the results will be analyzed, with discussion related to the optimization and the functionalities and attributes of the system as well as some conclusions and future work in this area.

4.2.1 Introduction

UBIAPP follows the general diagram of the solution presented in Figure 4.1. In this stage of the research, a single application was developed for Android mobile devices with a version of the operating system 2.2.3 and a simple web information system.

The context taken into consideration was: situations in which people use applications to meet sexual partners nearby, which could increase their chance of exposure to sexually transmitted infections (STI). The main question addressed was: how can users be encouraged to adopt preventive measures without violating their privacy or infringing on the character of the application?

To achieve the goal of preventing STI, two approaches were used: the automatic detection of points of interest (POIs) through a geofencing approach and the detection of risky applications [1].

---

[1] Risky applications are defined as applications that promote and increase the social interactions

UBIAPP sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, or their proximity to areas with a high probability of intercourse (*hot zones*). The underlying idea is the same as that for warning messages on cigarette packets, since users read the message just when they are going to smoke; however, it is important to note that smoking is bad, but a sexual intercourse is not. Therefore, the application detects two main situations:

1. Detecting the proximity to areas with a high probability of intercourse or meetings named as hot zones (see Figure 4.3)
2. Detecting the activation of a particular risky application (see Figure 4.5)

To achieve these goals, the solution proposal follows the scheme presented in Figure 4.1. These components are:

1. Simple web information system (SWIS): through this SWIS, different POIs can be added and stored, known as hot zones with a radius of geofencing on the website from the selection of their position on the online map. Figure 4.2 shows different hot zones added by the user with a yellow radius that represents the area of influence of the POI that is used to geofence the area.
2. Mobile application: this will detect the two previously described situations.

The mobile application for detecting the proximity of the users to a hot zone will continuously compare the position of the users with the stored position of the hot zones. It is important to note that all hot zones are retrieved from the stored data on the SWIS. Figure 4.3 shows different states of the application at different times (screenshots a, b and c). First, on the map, it is possible to see the red areas and a yellow person. The red areas correspond to a hot zone with their area of influence, and the yellow person indicates the current position of the user (see Figure 4.3-a). Second, users can interact with the map through different actions such as moving

---

of a specific target group such as: Grindr, Manhunt, etc.

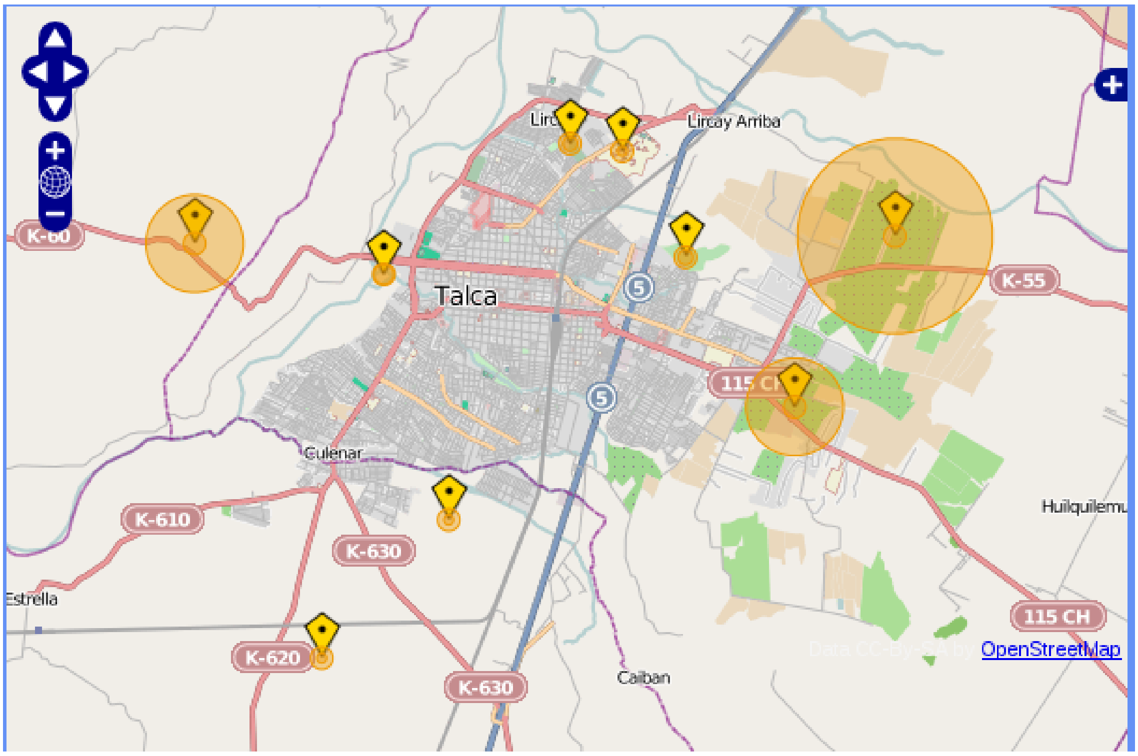
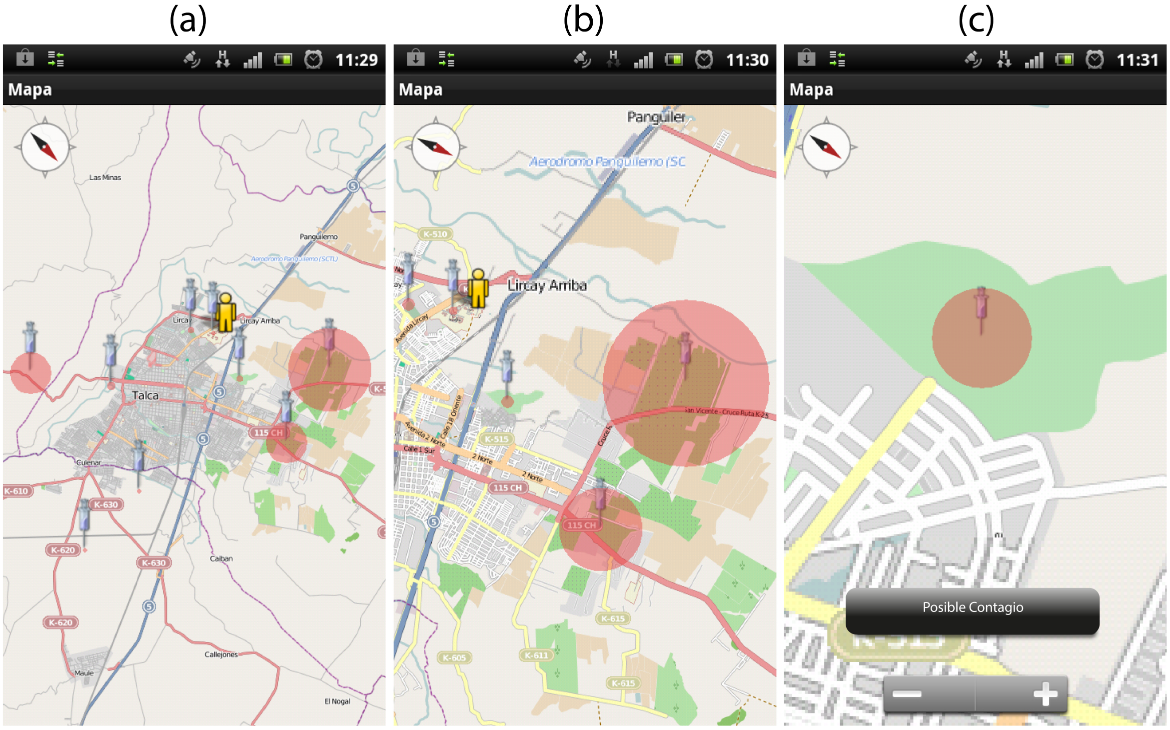


Figure 4.2: UBIAPP - Web client: the user interface of the web server has been constructed by integrating OpenLayers and OpenStreeMap such that users can visualize the various alerts that have been included in the database. New alerts can be created by georeferencing through the map and mouse interaction, and relevant information can be added to the alert.

and zooming in or out, among others (see Figure 4.3-b). Third, if users are in a red zone (which means that users are in a hot zone), the application notifies them with a health message [2] (see Figure 4.3-c).

---

[2] The term health message relates to any preventive message according to the context of the research.

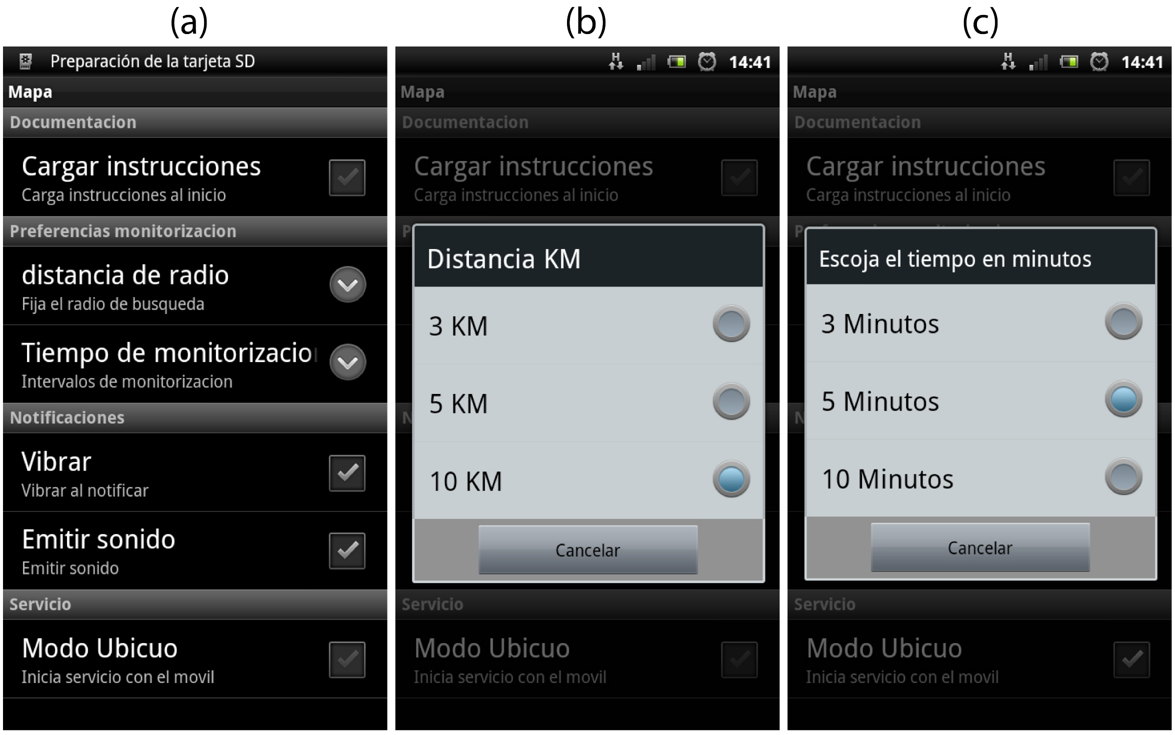


Posible Contagio

Figure 4.3: UBIAPP - Android client: the user interface shows the various nearby *hot zones* (red circles) and users can select these areas to obtain more information.

As an example of how the activation of a particular risky application is detected, Figure 4.5 shows three actions that are carried out by users at different times (screenshots a , b and c).

- First, users will start the surveillance of risky applications. This means that the software is constantly monitoring the mobile phone for the running of a risky application (the risky application can by installed, but it will be detected every time that users open it). In other words, this module works just like an antivirus (see Figure 4.5-a).
- Second, users open a risky application, in this case, Manhunt (see Figure 4.5-b).
- Third, UBIAPP detects the activation of Manhunt and notifies users through the notification system of Android (notification bar) with a health message, as it is shown in the image (see Figure 4.5-c).



(a)

Preparación de la tarjeta SD

Mapa

Documentacion

Cargar instrucciones ☐
Carga instrucciones al inicio

Preferencias monitorizacion

distancia de radio
Fija el radio de busqueda

Tiempo de monitorizacio
Intervalos de monitorizacion

Notificaciones

Vibrar ☑
Vibrar al notificar

Emitir sonido ☑
Emitir sonido

Servicio

Modo Ubicuo ☐
Inicia servicio con el movil

(b)

14:41

Distancia KM
3 KM ○
5 KM ○
10 KM ●
Cancelar

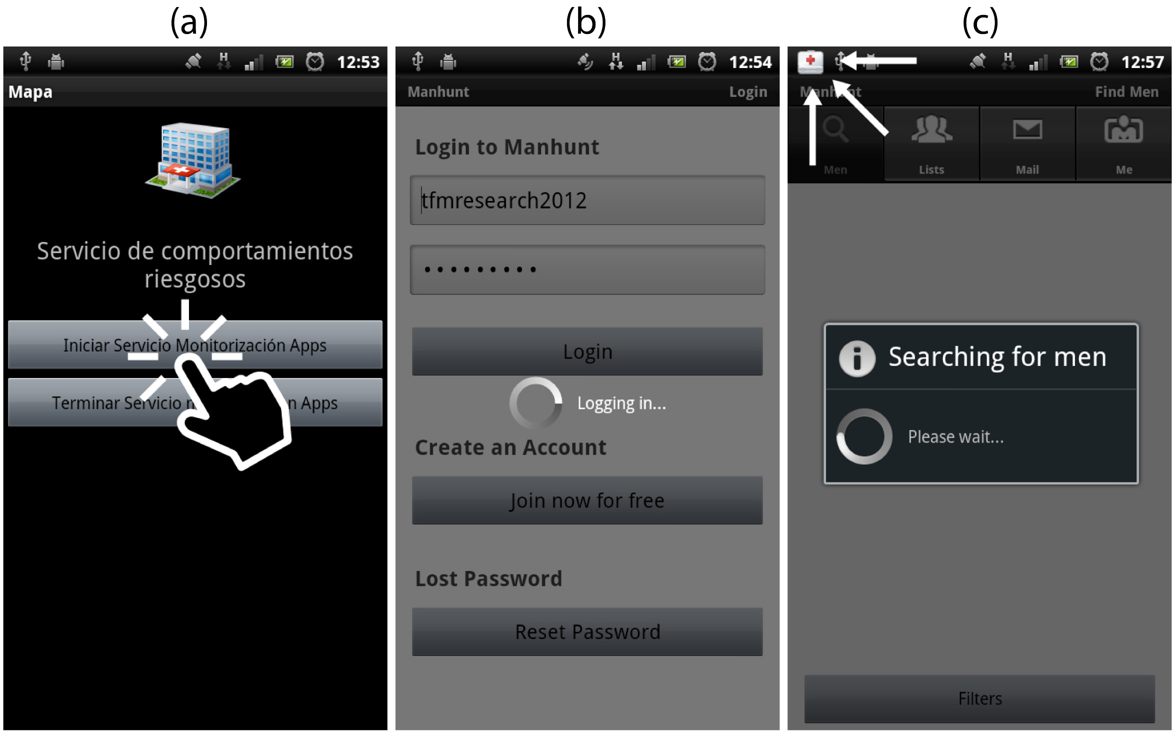
(c)

14:41

Escoja el tiempo en minutos
3 Minutos ○
5 Minutos ●
10 Minutos ○
Cancelar

Figure 4.4: UBIAPP - Android client: the interface shows the configuration section of the application.

UBIAPP allows for customization by users with different parameters for both kinds of detection (Figure 4.4 screenshots a, b and c). Therefore, users can configure the application to their needs. More specifically, for the detection of risk applications users can add or delete their favorite applications from a list of risk applications; and for the detection of hot zones, users can configure the timing of detections, sounds, and vibration for the notifications, among other features. Figure 4.4 shows the configuration section that uses the same standard and graphics of the operating system.



tfmresearch2012

•••••••••

Logging in...

Searching for men

Please wait...

Figure 4.5: UBIAPP - Android client: the software runs in the background and alerts users when certain applications are run (labels translated from Spanish).

4.2.2 Summary

The UBIAPP system has two major components: 1) the simple web information system , and 2) the mobile application developed for Android operating system version 2.2.3. The first component allows the adding and storing of hot zones through an interactive map web interface; the second component sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, or their proximity to areas with a high probability of intercourse (*hot zones*).

The proposed application is designed to give users a sense of control because: 1) the users themselves decide to install it, accepting the conditions; 2) it sends a message that helps users think about taking appropriate preventive measures (a list of messages will be shown in the results section); and 3) it works in the background without interfering with users unless a trigger situation is detected.

In the next subsection, the technical aspects of the application will be presented with its software artifacts, which are: actors, general use case, architecture of the solution and statechart diagrams, among others. Then, the context of the testing will be established with a target group. Furthermore, the results will be presented.

Finally, the results will be analyzed, with discussion related to the optimization and the *functionalities and attributes of the system* as well as some conclusions and future work in this area.

4.2.3 Technical aspects

The following subsections will describe the IT artifacts that were involved in the software development process for UBIAPP.

4.2.3.1 Actors

The system is composed of several entities that interact among themselves. The external entities that interact with the system are called actors. These actors interact with the system with a specific role in the use cases (Larman [59]). The following list represents the main actors that interact with the different components of the software:

- **User**: this actor works with the services of the information system. It is classified in two main roles:
  - **Mobile user**: user that runs on a mobile device the application UBIAPP, sending and getting data from the main server.
  - **Server user**: system administrator that controls the data, messages, positions and information of the mobile users in the information system (website).
- **Database**: this actor represents the database management system (DBMS) for the webserver. The database has a central core to distribute information to all the live clients. The DBMS used for these purposes is MySql.

- **Information server**: this actor represents the web information system with all its components such as: the database, front and back end, among others. It will interact with the actors: server users and the mobile application.

- **Mobile application**: this actor represents the mobile application with all its components. It will interact with the actors: mobile users and the information server.

4.2.3.2 General use cases

To understand how the software behaves, two general use cases are presented. These use cases briefly explain each process and its interaction with the various solution components, which will also be outlined in this section. The first use case triggers a message when the application detects that users are within *hot zones*; the second triggers messages when a contact application is launched on the smartphone, as defined in the summary.

Use case US01:	*General detection of* hot zones
Actor(S):	Users, information server.
Purpose:	Detect a *hot zone* at an appropriate time and notify users of its location.
Summary:	This use case begins when the notification service detects a *hot zone* near the users' current positions. The alert is communicated using a RestFul service to a server containing a database of *hot zones*. Then, the software notifies users about the zone.
Preconditions:	1) The database of *hot zones* should contain information on their positions and radius; 2) users should have launched the notification service within the software (it is assumed that users have previously configured the software).

Actor's actions	**Application's answers**
	1.- This use case starts when the *notification service* obtains users' geopositions.
	2.- The notification service requests information from the server with users' geopositions and the *hot zones* closest to them.
	3.- It determines if users are near a *hot zone*, using distances that have been configured previously.
	4.- It notifies users of the *hot zone* through the notifications bar.
5.- Users select the notification.	
	6.- The system opens a window using *mapView*, in which it shows the various *hot zones* including their radius.
7.- Users can select a risk presented in the mapView and obtain information about it (available when the application is used for infectious diseases prevention).	

Alternative flow: 3.- The *notification service* do not find a *hot zone* near users' geopositions, therefore the system checks again after X amount of time has passed or Y meters have been traveled.

Table 4.2: General detection of a *hot zone*

Use case US02:	General detection of a contact application (such as Manhunt)
Actor(S):	User, information server.
Purpose:	Detect a contact application at an appropriate time and notify users of preventive measures and, if they request it, information on the nearest place to get them.
Summary:	This use case begins when the notification service detects that a risky application has been executed. The software launches a preventive message (see Results section).
Preconditions:	The notification service runs independently of how it was started, for example, by software or by the operating system (it is assumed that users have previously configured the software).

Actor's actions	**Application's answers**
1.- This use case starts when users start risky applications.	
	2.- The *surveillance service* detects the launching of a contact application.
	3.- It notifies users of the launch of the application through the notifications bar.
4. - Users select the notification.	
	5. - The system opens a window with mapView in which the nearest health center or pharmacy to users' positions is shown.
6. - Users can select the health center or pharmacy and obtain information about it.	
7.- Users can select a risk presented in the mapView and obtain information about it.	

Alternative flow: 2.- The *surveillance service* did not find a risk-associated application therefore the system checks again after X amount of time.

Table 4.3: General detection of a monitorized application

More details on use cases *General detection of hot zones* and *General detection of a contact application* are provided in Tables 4.2 and 4.3, respectively.

These use cases suggest that the solution proposed by this work requires the consideration of the following aspects:

- Creation and storage of geodata sources.
- Development of a web service for distributed systems.
- Integration with mobile devices.

4.2.3.3 Sequence diagrams

A sequence diagram shows object interactions arranged in a time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

In this section, two main *sequence diagrams* will be discussed that correspond to the two main general use cases of the system, see Tables 4.3 and 4.2. Besides that, some references to source code will be shown when it is required.

One of the questions related to this use case 4.3 was: how can risky behavior be prevented through a message at the exact time when users are using applications of social media? From an informatics perspective, the issue of social media applications is addressed in four parts:

1. To know which apps are the ones that the application has to detect to make a blacklist.
2. To know what users are doing with their smartphones the whole time.
3. To check and detect if any of the users' active apps are on the blacklist.
4. To notify users of the situation to make them conscious of their actions.

The following sequence diagram represents a black box interaction between the main components:

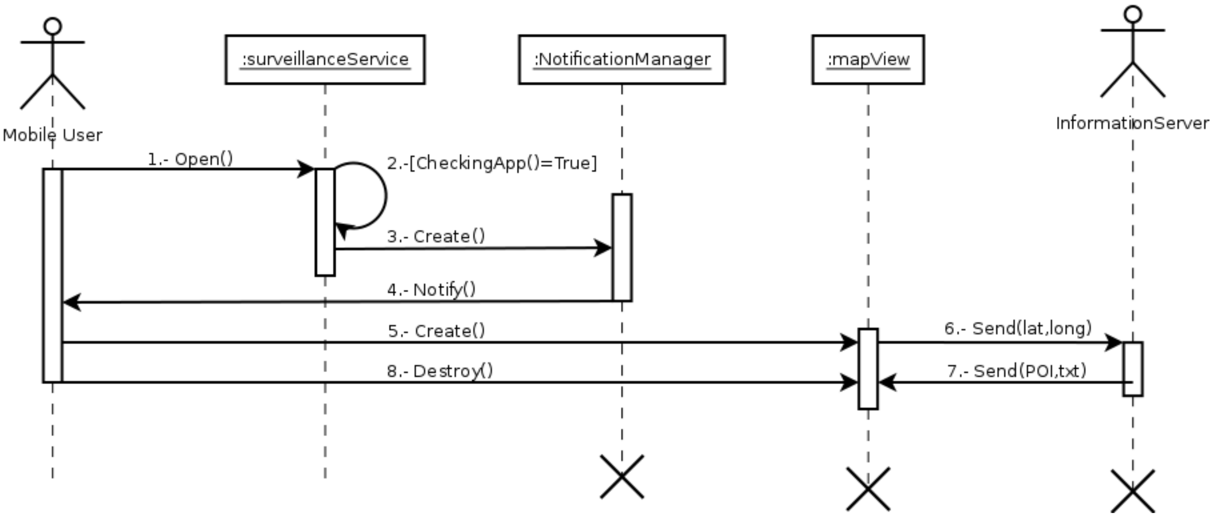


Figure 4.6: Sequence diagram: black box representation of the general use case US02 *General detection of a contact application*, see Table 4.3

As can be seen in the sequence diagram of Figure 4.6, two main actors are involved in this interaction: 1) the mobile user, and 2) the information server. The following description is based on the previously remarked question.

The first part (to know which apps are the ones that the application has to detect to make a blacklist) was addressed taking into consideration the interaction with the mobile users. In this case, the users are asked to select from a list of installed applications which ones they want to add to the blacklist. After that process, the applications are saved on a dynamic object for future comparison with the *surveillance service*. This decision was made thinking about giving the mobile users the opportunity of deciding which applications to add, but more than that, it gives the opportunity to make a decision based on what the mobile users want. Therefore, they are not forced to add any particular applications. This instance is a key factor for preventing behaviors and making the mobile users concious of it.

With regards to the second part (to know what users are doing with their smartphones the whole time), once the mobile users run the *surveillance service*, it will check permanently which applications are running on the operating system and make a dynamic list with their names. With this approach, the second part is resolved (this is represented in the sequence diagram of process 2).

Part three (to check and detect if any of the users' actions are on the blacklist) can be seen in process 3 of the *surveillance service*. The Android service is checking the list of currently running applications with the blacklist until it detects a match.

To resolve the final part (to notify users of the situation to make them conscious of their actions), when the service detects a match, it will send a message to the *notification manager* which has the responsibility of showing it to the mobile users through the notification bar.

Figure 4.5 represents this use case from a graphical user interface (GUI) perspective on the mobile phone. The GUI represents what the mobile users see, but the last description in the sequence diagram shows what the application does (See Figure 4.6).

The previous discussion focused on the question of how to prevent risky behavior when users open social media applications; in contrast, the text below will explore how risky behavior can be prevented when users enter a hot zone, see Table 4.2. From an informatics perspective, the issue of hot zones can be addressed in four parts:

1. To know where the hot zones are in a particular area.
2. To know the current position of the mobile users.
3. To check and detect if the mobile users are nearby a hot zone.
4. To notify users of the situation to make them conscious of their actions.

The following sequence diagram represents a black box interaction between the main components:

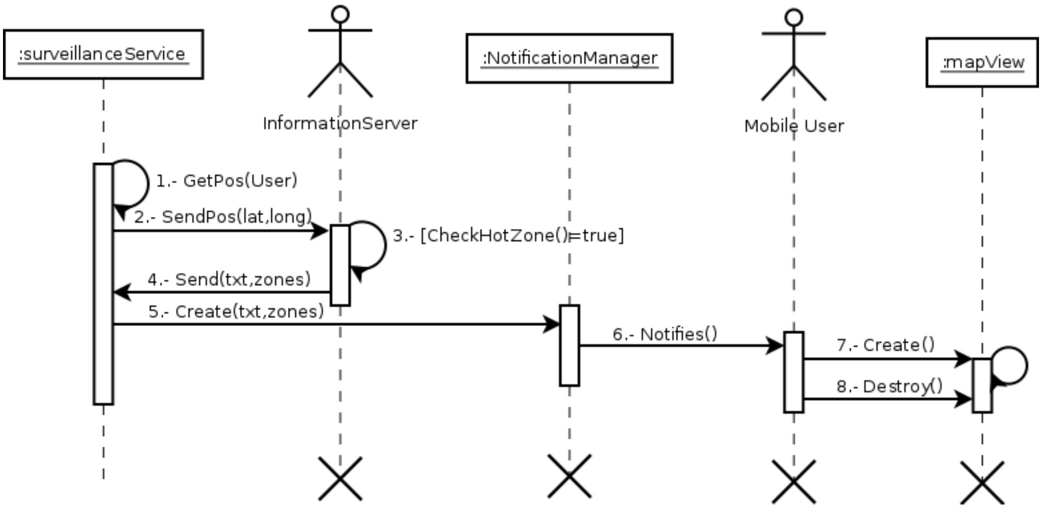


Figure 4.7: Sequence diagram: black box representation of the general use case US01*General detection of a hot zone*, see Table 4.2

As can be seen in the sequence diagram 4.7 two main actors are involved in this interaction: 1) the mobile user, and 2) the information server.

In order to address the first and second parts (to know where the hot zones are in a particular area; and to know the current position of the mobile users), the developed system has two components, the web component and the mobile component. In this context, the defined actor *information server* is part of the web component and the *mobile application* is part of the mobile component in general terms. Therefore, for having a sustainable way to know where the hot zones are in a particular area, the decision was made to use the client - server architecture. In this way, the *information server* is responsible of storing all the geographical points with their information in a database (back end). Moreover, this component should provide an interface for entering the data into the database. For this purpose, a web interface was developed to add the POIs and their information, see Figure 4.2. Finally, every comparison of the *mobile application* services with the *information server*, will use the data stored in the database of the *information server*(this is represented in the process 3 sequence diagram, see Figure 4.7; details of how the communication is established between these components will be described in the section about the architecture of the solution).

With regards to the third part (to check and detect if the mobile users are nearby a hot zone), a background service was created, as can be seen in process 1 of the *surveillance service.* The Android service is checking the current position of the mobile users through the localization services available for the mobile device. When the service detects the mobile users' positions with a predefined level of accuracy (see Figure 4.4), it will send the *information server* its position. The *information server* has the responsibility of comparing the users' positions with stored hot zones. This comparison considers the distance of the mobile users' position with a hot zone searching for all the POIs nearby with a predefined distance. These distances can be configured by users after the installation process, see Figure 4.4).

To resolve the last part (to notify users of the situation to make them conscious of their actions), if the server detects a hot zone in the determined distance, then it sends a signal to the mobile application. This signal will handle a message to the *notification manager*, which has the responsibility of showing it to the mobile users through the notification bar.

Figure 4.3 represents this use case from a graphical user interface (GUI) perspective on the mobile phone. The GUI represents what the mobile users see, but the last description of the sequence diagram shows what the application does.

4.2.3.4 Statechart diagram

The statechart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and changes when some event is triggered. So, the most important purpose of the statechart diagram is to model the lifetime of an object from creation to termination. So, the emphasis is given to the states of the *surveillance service.*

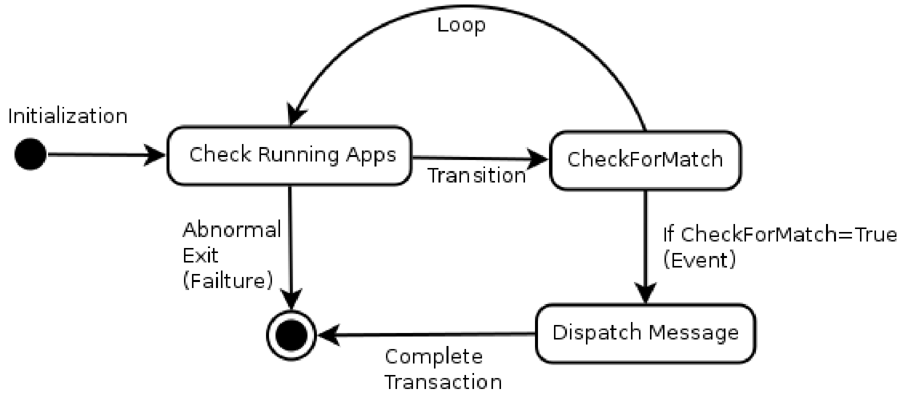


Figure 4.8: Statechart diagram of the *surveillance Service*

In Figure 4.8, the first state is *checking the running applications of the system* (check running apps). This state begins once the process starts. The next states are *checking for a match* (check for match) and *dispatch message* (dispatch message). These events are responsible for state changes of the object. During the lifecycle of a service, it goes through these states and there may be some abnormal exits. These abnormal exits may occur due to some problem in the system or during the comparison process (which are handled by a try/catch block). When the entire lifecycle is complete, it is considered as the complete transaction and the service is ended until the mobile users start the service again.

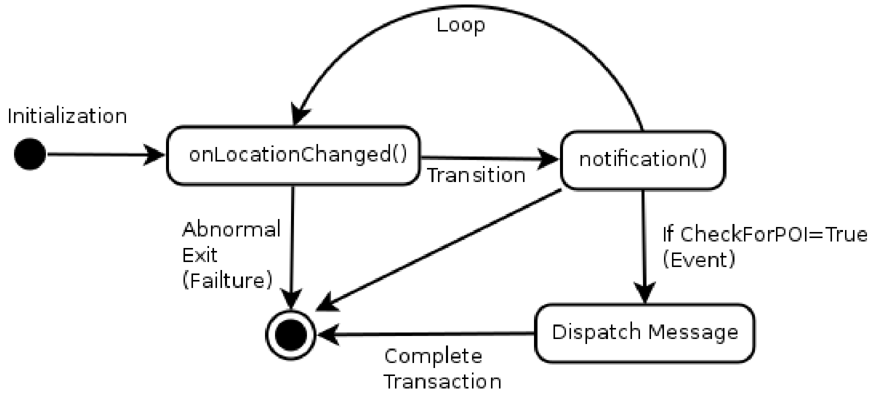


Figure 4.9: Statechart diagram of the *risk service*

On the other hand, the *risk service* will use the internal GPS or Networking of the mobile device for location and the *information server* for comparing the POIs. This state begins once the process starts. The next state corresponds to the location service which is named *onLocationChanged*. Once the location of the *mobile users* changes it will trigger the next states such as: *notification* and *dispatch message*. When the entire lifecycle is completed it is considered as a complete transaction and the service is ended until the mobile users start it again, see Figure 4.9 and Listing 4.1.

Listing 4.1: Risk service: once the service is created, and bound to the state onCreate, the criteria of localization are defined and then the last position of the mobile users is stored in the state onLocationChange

```
public class localizacion_sensor extends Service implements LocationListener{


  @Override
  public void onCreate(){

    locationManager = (LocationManager)getSystemService(Context.LOCATION_SERVICE);

       Criteria criteria = new Criteria();
       criteria.setAccuracy(Criteria.ACCURACY_FINE);
       criteria.setAltitudeRequired(false);
       criteria.setBearingRequired(false);
       criteria.setCostAllowed(true);
       criteria.setPowerRequirement(Criteria.POWER_LOW);

       provider = locationManager.getBestProvider(criteria, true);
       location = locationManager.getLastKnownLocation(provider);
     locationManager.requestLocationUpdates(provider, 0,0, this);

  }


  @Override
  public void onLocationChanged(Location location) {
    // TODO Auto-generated method stub
    setLatitude(location.getLatitude());
    setLongitude(location.getLongitude());
    setAltitude(location.getAltitude());

  }
}
```

Further details of the architecture of software and components will be described in the next section.

4.2.3.5 Architecture of the solution

To implement a distributed system that is scalable and independent, RESTful services principles were used in conjunction with various open source components [60, 61]. REST defines a set of architectural principles focusing on a system's resources. All the various states of the resources are addressed and transferred using the HTTP protocol, which allows the system to interact with any client capable of implementing the HTTP protocol.

Figure 4.10 shows the conceptual architecture that integrates all components and services of the proposed solution, according to the REST principles. This system has two components:

- The web component (web server): this is mainly a web server for managing the application data, and includes a web client.
- The mobile component (Android client): this is mainly an Android client.

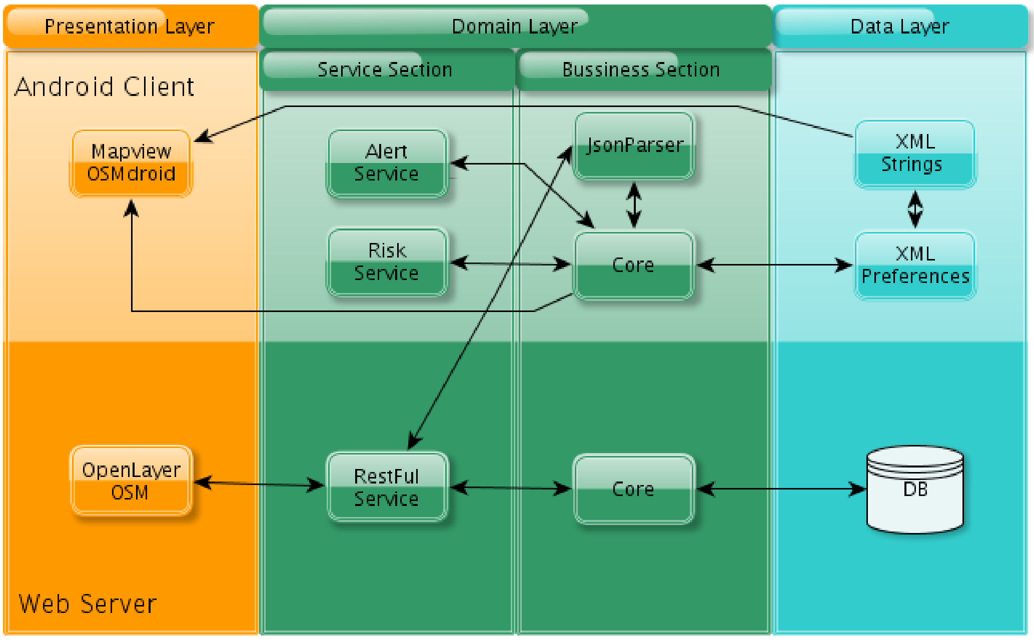


Figure 4.10: Implemented architecture and essential components and services. The architecture consists of three main layers: the presentation, domain and data layers, representing the interaction between the essential components of the solution. This representation is divided between the Android client (top) and the web service (bottom). The re-used open source components in the proposed architecture are also shown.

- **The web component** : The web component allows the saving and managing of geographic information of locations and zones that are relevant for STI prevention: for example, places where users can acquire condoms, or zones where contacts usually take place. Health professionals can create and update this information to optimize preventive actions.

From an architecture point of view, the web component has three layers: the presentation layer, the domain layer and the data layer.

- **The presentation layer of the web component**: The presentation layer of the web component corresponds to the web site that allows the introduction of georeferenced locations and zones (See Figure 4.2). PHP has been used as a control language [62], OpenStreetMap (OSM)[63] as a free map framework, and the OpenLayers libraries [64] to handle the various layers of the maps. Note that while OpenStreetMaps (OSM) can be used as a source for points of interest, these points do not currently contain the specific information required for the proposed preventive application. Therefore, OSM is used as base cartography, and to identify the location of the elements.
- **The domain layer of the web component**: The domain layer is subdivided in two parts: the service section and the business section.
    - **The service section in the web part**: The service section implements a RESTful service that processes requests and communicates with the business section. Note that this is the section that ultimately interacts with clients associated with the service. This architecture provides a solution with greater interoperability and modularity. Therefore, the architecture will not be affected by eventual transformations on the server or client side.

      Usually, client applications request resources from the server using GET methods from the HTTP protocol. These resources are identified using a URI. The HTTP protocol is used in RESTful services to interact with the resources of the server. This is very beneficial for mobile clients because of their limited resources. Using this design, the client load will be lightened by turning it over to the server, which has more resources.

    The RESTful service receives a request from the Android client (*notification service*), which also sends users' geolocations and a search radius. The request is a simple URI.

    Using this information, the RESTful service communicates with the persistent data layer on the server (See Figure 4.10). More specifically, it communicates with the database in order to verify if users are near a relevant area . If this is the case, the service returns the area in order to display it on the mobile device.

    * **The business section of the web domain layer**: The business section corresponds to the core of the application, which executes a geopoints search algorithm that interacts directly with the data layer.

  - **The data layer of the web component**: The data layer (or persistence data layer), is a database that stores geopoint data, as well as their associated characteristics, and information added by health professionals.

- **The mobile component (Android client)**: A key element of the application is the interaction with users through the Android client. With the aim of developing Android software with reusable, modular and independent components, development was based on General Responsibility Assignment Software Patterns (GRASP). These patterns allow the creation of high cohesion and low coupling between objects (Larman [59]).

  The Android client is also divided into the same three layers as the server, namely the presentation, domain and data layers.

  - **The presentation layer of the Android client**: The presentation layer contains two main parts: the interface for accessing data on the configuration, and mode of execution; and the integration of OpenStreetMap with Android, which displays features in a particular area.

    The *cl.tfm.Maps* package was used, which contains the classes required to visualize *hot zones*, using the OSMDroid libraries [65].

The map obtains the stored preferences, such as the search radius for *hot zones*, instantiating a *PreferenceManager* object in the Android API. The minimum search distance was selected so that users would have enough time to respond and take preventive action. Using these data, it is possible to instantiate the class that communicates with the *RESTful* service, *JsonParser*. This class sends a *request* with a URI containing the longitude and latitude of the current position of the mobile device, and the radius of the search. The response is saved in an *ArrayList* of the *OverlayItem* type, which is defined in the OSMdroid[65] API. Inside each list, each OverlayItem object contains the latitude, longitude, affected radius, and the icon to be drawn on the map. Since this functionality can help to prevent other localized diseases, it also can give information about the risk found.

In order to visualize the *hot zones* and health points (see Figure 4.3), an object instance of the *activity* type has been created [66], which follows the lifecycle of an *activity*, defined by the Android operating system. To draw on the map, three layers were constructed:

- Map layer or *mapView*: in the main activity, specifically in its *onCreate()* state, a mapView is loaded, which is defined by an XML file. The layer was made and several features were added as defined by the Android API, such as *setBuiltInZoomControls(); enableCompass(); enableFollowLocation();*.
- Point layer: this obtains the latitude, longitude and icon from the *ArrayList* created by the JsonParser object, and adds OverlayItem objects to the layer.
- Area layer: a new overlay object was instantiated, overriding the *draw()* method. In this method, the *ArrayList of the OverlayItem type* list is reviewed with the geopoints. A point at the same location is also created, but with a circle drawn in the middle using *Canvas*,

    specifically with *drawCircle()*. In order to measure the radius accurately, the projection of the map is obtained and transformed using the geopoint. Similarly, to transform the distance in meters, the API method *metersToEquatorPixels()* from OSMdroid is used. Thus, each time a ZoomIn or ZoomOut is performed, the method will re-draw the radius of the circle according to the map's projection and zoom.

  Each *activity* in Android has its own lifecycle. Therefore, it is important to point out that each state in the cycle interacts with the operating system. The following states were worked with: *onCreate()*, *onStart()*, *onDestroy()*, in which different interactions with the respective components of the operating system were instantiated, run and killed.

- **The domain layer of the Android client**: As for the web component, the domain layer is divided in two sections: the service section and the business section.

    - **The service section of the client domain layer**: The service section implements two internal monitoring services that are responsible for reviewing and comparing certain mobile states related to the location and the applications that are running that time. Simultaneously, it interacts with the business section.

      There are two kind of services: the notification service and the surveillance service.

        - **The notification service in the client domain layer**: The notification service sends the users' geolocations and the search radius to the RESTful service on the server via the business section. With this information, the RESTful service communicates with the server's persistent data layer (See Figure 4.10). More specifically, it communicates with the database in order to verify if the users are near a relevant area. If this is the case, the service returns the area in order to show it in the mobile device. Thus, this service detects *hot zones*. From technological point of view, it is linked with two essential components: *JsonParser* and *NotificationManager*. The algorithm for detecting if users are near an area stored in the web server database is based on the implementation of a *LocationListener*.

Once the service is initialized, in the *onCreate()* state, the following resources are defined:

1. *LocationManager*
2. *NETWORK_PROVIDER*
3. *GPS_PROVIDER*
4. *NotificationManager*

The descriptions of these resources are available in the documentation of Android API [66].

There are two options to geolocate users: *GPS_PROVIDER*, that uses GPS; and *NETWORK_PROVIDER*, that uses the network. *GPS_PROVIDER* is more accurate; however, it uses more battery. In contrast, *NETWORK_PROVIDER* is less accurate, but it uses less battery. In the case of this software, the accuracy of the *NETWORK_PROVIDER* is sufficient because it has a noise which is a short distance by foot. Therefore, ultimately it was decided to use the *NETWORK_PROVIDER* to save the battery.

Once the provider is selected, a *requestLocationUpdates()* is called, defining its parameters: the provider, the quantity of time and the distance that is selected to update.

After the update, if the location has changed the *onLocationChanged()* method detects this change, the current position is gotten, passing it to the notification class. This instantiates JsonParser with the current location of the mobile with the aim of getting a *hot zone* near the radius previously defined by users.

    If there is a message, it is communicated with the operating system's *NotificationManager.* To alert users, sound and vibration notifications have been developed, using the message bar of the mobile device to display a preventive message. In addition, an *Intent* [66] is activated when users select the message in the bar. If there is no message to send, the service will continue running until the users or the operating system kills the service.

  - **The surveillance service in the client domain layer**: The surveillance service monitors the launch of some applications, such as when users run a program to find sexual partners. When this happens, the service sends the users' geolocations, in an URI, to the RESTful service on the server, via the business section. With this information, the RESTful service communicates with the server's persistent data layer (see Figure 4.10). Specifically, it communicates with the database to get all health points within the chosen radius where the users can obtain condoms or other preventive measures, and returns their geopoints and names. Note that in some situations the service might not find any response within the given radius.

    When any of the services detects a situation that requires a notification, this message is sent to users. The *cl.tfm.services* package stores the internal services that are responsible for notifying users when they run a monitored program that facilitates contacts, or when the users are near to a *hot zone.*

- **The business section of the client domain layer**: The business section contains the core of the application that interacts with *JSon-Parser*, and is responsible for sending messages from the notification and surveillance services to the RESTful service of the web component. The *cl.tfm.JsonParser* package has the classes required make different *requests* to the *RESTful* service and process data, thereby facilitating their subsequent representation by other components of the software.

4.2.3.6 Optimization and key points

Optimization is a key factor for mobile devices since the main power source (the battery) is limited. The resources that spend more battery on a mobile device are screen, GPS, long processing time and internet connection. Taking this into consideration, all the algorithms are optimized to provide the maximum efficiency in the use of resources. For instance, all the processing of checking the *mobile users'* positions with the list of hot zones were calculated in the *web information server*. Therefore, the calculus that is done comparing 1 to N points is the responsibility of the web server, minimizing the processing time in the mobile device. A drawback of this method is the dependency of the internet connection because, it is necessary to be connected to send and get the data of the Restful service.

The major issue with saving battery is determining the intervals of time for checking the users' position. The constant use of GPS will drain the battery in a few hours. So, to reduce the battery drain of the *risk service*, it was decided to let users choose a reasonable time interval from the configuration interface, see Figure 4.4. After some laboratory testing, it was decided to offer: three, five and ten minutes, making an analogy to extreme, normal and light monitoring. This timing is accurate for three main reasons:

1. **Distance**: how many meters a person can walk in just three, five or ten minutes. It is necessary to remember that the use case is to detect a hot zone under normal conditions. So, this is taken as a premise for optimization.

2. **Risk Service**: the service has to get the users' positions and send them to the *web server*. Once that process is done, the *risk service* will unregister the location service to save battery and will reestablish it when necessary (avoiding requesting localization constantly).

3. **Detection**: once the *risk service* detects a hot zone, the service will go to sleep for a specified amount of time. In this scenario, the *risk service* is constantly checking the users' positions, sleeping, and waking up in a constant loop that increases the battery life of the device.

Table 4.4 summarizes the main aspects that were considered in this version of the research:

Android Client / Code name	Optimization	Drawback
UBIAPP	Implementation of transitions and different states of the service (sleeping and running)	The long use of the application and service will drain the battery and short the battery life because the use is constant and higher than the normal use
UBIAPP	Timings and use of the services are configured by the users	-
UBIAPP	Processing data is done on the *web server* for optimization of the mobile resources	Internet connection is required for this purpose

Table 4.4: Optimization and drawbacks of the app code name: UBIAPP

4.2.4 Testing and results

In this section, the results from the testing of UBIAPP will be shown. First of all, users' interaction with the software will be shown; then, a analysis will be presented of how useful ubiquitous computing is as a preventive tool.

To test the application and choose the healthy messages, a test was performed with 17 MSM. This testing was carried out by both professionals from health departments (ASPB) and researchers of computer sciences, multimedia and telecommunications departments (UOC).

4.2.4.1 Testing context

Most users of contact applications such as *Grindr* or *Wapa* are aware of the risk of transmission of STIs. However, since those applications allow them to rendezvous shortly after making contact, this can give users little time to remember the recommended precautions. Thus, it would be interesting to be able to remind users of what they already know, by launching a preventive message at precisely the moment when it would be most relevant and would have the greatest impact (Rietmeijer and McFarlane [5], Kaplan [18]).

An example of this idea, as mentioned previously, is the use of shocking images on cigarette packets to reflect the risk of smoking, such as a photo of yellow teeth or a cancer patient. Thus, when smokers reach for a cigarette, they receive a visually powerful preventive message. Several studies (Chanon et al. [67], Ehrman et al. [68], Bradley et al. [69]) have analyzed the impact of this type of messages and conclude that they are important for the complex task of changing behaviors. While this comparison with cigarette packets is interesting, the problem addressed in this work is different because:

1. In this case the action itself is not a risk, but rather the failure to use preventive measures when performing the action.
2. Users are told to do something, rather than not to do something (for example: ways to have safer sex).

It is also important to note that potential users of the application are people that are already concerned with their sexual health, and therefore such messages might be more likely to have a positive effect.

Thus, the situations that can trigger the message are:

- *Execution of some kind of application*, which is the primary function and applies in situations where users open applications designed for contacting sexual partners, such as *Manhunt* (see Figure 4.5).
- *Proximity to a geographical zone where sexual contacts often take place*, which is the secondary function and applies in situations where users enter or are near to a *hot zone*.

It is important to note that the application not only sends a message, but also shows a map with the route to the nearest point where they can obtain preventive measures. Once the notification has been displayed, the software can interact with other installed applications, for example allowing the users to share the notifications through email, text message, social networks, etc. Thus, users have a fully-connected experience that can also help to promote prevention among others. When they receive a notification, users can make an informed decision regarding the possible consequences of their behavior. As a result, the use of this software raises users' awareness of their actions and encourage them to take steps to limit the spread of STIs.

In addition, the proposed software could also offer an effective mean of prevention in some cities, by helping users to avoid certain areas when not carrying appropriate preventive measures. Therefore, as a measure for mitigating transmission of diseases, this software gives users access to information about contact areas. When users open the software, all the *hot zones* near users' geolocation are retrieved and shown on the map screen (see Figure 4.3). Each red circle corresponds to a *hot zone*, and the users can obtain information about each *hot zone* by touching the icon associated with it. Therefore, as a measure for mitigating transmission of disease, this software gives users access to information about contact areas.

The software only helps users to remember information, but it is the user who decides.

It is important to note that it can also act as a finder, but this is not a problem since the goal is to use preventive measures and not eliminate sexual contact.

In the following subsections, three key aspects are introduced for the effectiveness of the application: the levels of functioning (conscious and automatic); the notifications launched; and the interface.

4.2.4.2 Levels of functioning

UBIAPP operates in two modes that reflect its function at the automatic and the conscious level:

- **Automatic level (AL)**: this option must be configured in the software. Users install the application if they decide that they want to receive a preventive message when they are likely to have an *express* date. Once installed, the application runs constantly in the background, so users do not need to run it in order to receive notifications. This gives users maximum liberty in using their mobile, while keeping them aware. On the other hand, note that it must be the users who voluntary decide to install and use the application.

  When users run a service, even if the application is closed, the service will continue to run in the background. This allows users to have a completely different experience, since they will be able to use their device normally: to talk on the phone, navigate the web, or simply leave the device idle.

  Running the surveillance service to monitor a contact application or selecting the automatic mode provides constant monitoring of the list of programs that are running in the operating system (this service checks the program list every three minutes). It also monitors position to launch a warning when entering a hot zone.

- **Conscious level (CL)**: in this mode, users run the application with the purpose of finding zones identified as having a high probability of sexual contacts, the so-called *hot zones*. Note that this function is location-dependent: in some cities there are locations where contacts are more likely, while in large cities like Barcelona, with more than 7 million tourists per year, mobility is very high and it may be difficult to identify specific locations where contacts take place. Even so, in Barcelona there are some specific points where sexual contacts are more likely, like some clubs or gay saunas. Therefore, this functionality could also be useful because it improves the UX. As mentioned previously, it can also act as a finder, but this is not problematic because the objective is using preventive measures and not stopping sexual contact.

The omnipresence and imperceptibility of this preventive notification service is a defining characteristic of ubiquitous computing (Bell and Dourish [17], Bull et al. [22]). The main advantage of this ubiquitous system is that the users are not conscious of the fact that their smartphone is being used as a healthcare device.

4.2.4.3 Preventive messages

The notifications launched play a key role in proposed prevention strategy. Studies in adolescents show that the message itself is important (Cornelius et al. [70], Perry et al. [71], R et al. [72]). The time when it is sent and its format is also important (Cornelius et al. [70]). In all of these studies, however, the messaging service is run by an operator who sends the message. In some cases, teenagers have the opportunity to answer the message(Cornelius et al. [73]) and it is common for these services to have a face-to-face component. In other examples based on Public Service Announcements (R et al. [72]) the messages are longer and more complex than the short message used on a smartphone.

The proposed software is an automated system of unattended messaging, so it is even more important to carefully choose the messages previously. In this sense, these messages are more similar to those used on cigarrete packets, where impact reviews demonstrate the importance of the clarity of the message (Hammond [74]).

To choose the preventive messages, as said before, a test was performed on 17 MSM, since they are at high risk of infection with HIV. Their demographic characteristics are shown in Table 4.5. Nearly 60% are in their thirties and more than 20% are over 50 years old. Most were Spanish (11), and 6 were from other countries.

Table 4.5: Demographic characteristics of volunteers who helped to choose the messages ($n = 17$)

Age		**Origin**	
21–30	17.6%	Spain	64.7%
31–40	47.1%	Andorra	5.9%
41–50	11.8%	Brazil	5.9%
51–60	23.5%	Chile	5.9%
		Honduras	5.9%
		Peru	5.9%
		Dominican Republic	5.9%

This table shows the demographic characteristics of people who helped to choose the messages.

In (Table 4.6), the preferences of volunteers to find a partner are shown. In the table, it can be seen that 78.1% go to specific locations (saunas, discotheques, bars, gymnasiums and parks), 21.9% use geolocators and 28.9% use the internet[3]. Therefore, the software could be useful to most of these volunteers because they find partners in specific locations, or use geolocator applications.

Nonetheless, although all participants volunteered to use the application, only 56.3% considered themselves or people like them as the target of the application (see Table 4.7), while the remaining 43.8% considered that this kind of application targets people different from them. Regarding age, the most popular answer (30.6%) considered that this application targets people aged 21–30 years, which corresponds to the age of most of the volunteers, and only 19.4% considered that it targets people of any age. When asked if they would download the application, 72% answered affirmatively.

Volunteers were asked to choose the messages they found most and least suitable as preventive messages, and were also offered the chance to propose new messages.

---

[3] The sum of the percentages is not 100% because some volunteers used more than one of these channels.

Table 4.6: Places where users usuallyfind partner ($n = 17$)

Place	
Gay sauna	28.1%
Internet	28.1%
Geolocators	21.9%
Discoteques	21.9%
Bars	21.9%
Has partner	3.1%
Gimnasium	3.1%
Beach	3.1%
Park	3.1%
Street	3.1%

Percentages does not sum 100% because every volunteer could mark several options. This table shows where the volunteers in the sample usually make contact.

Table 4.7: Target of the application according to potential users' perception ($n = 17$)

Target profile		Target age	
Anybody	43.8%	15–20	16.7%
Someone like me	12.5%	21–30	30.6%
Someone different from me	43.8%	31–40	19.4%
		41–50	11.1%
		51–65	2.8%
		Any age	19.4%

This table describes the volunteers' perception as the target of the application, from the profile point of view, as well as from the age point of view. Percentages do not sum to 100% because each volunteer could mark several options.

The most popular sentences, which will be displayed randomly by the application, were as follows:[4]

- Enjoy sex and enjoy life. Do not expose yourself to HIV
- Remember: HIV is invisible
- Appearances can be deceiving
- Take the test, avoid AIDS, condoms please, avoid HIV

In contrast, the least popular sentences were:[5]

---

[4] Translated from Catalan.

[5] Translated from Catalan.

- Every year 700 people become infected with HIV in Catalonia
- Every treatment of HIV costs 12,000 euros/year
- 2x1! HIV is free!
- Packaging, where are you going? Where the condoms go?[6]

4.2.4.4 Interface

The goal of the interface was to develop a simple interface where users could access information quickly and configure the software, including minimal functionalities, such as ZoomIn and ZoomOut, to more complex functions, such as seeing different locations on the map.

The monitoring services work according to the users' configuration. Figure 4.4 shows how users can manually turn on the service. The appearance of the message prompts users to touch the screen, which opens map visualization software installed on their mobile, such as OSMdroid or Google Maps, showing their current position and the area of interest.

The software also has some features that can be customized by users via the data configuration screen, under the *Android guidelines*:

- Show instructions (CL and AL): shows a brief summary of how the software works.
- Ubiquitous mode (AL): initializes the notification service whenever the mobile device is switched on, without users needing to start the process every time.
- Vibration (AL): sets the mobile to vibrate when an alert is shown.
- Sound (AL): sets the mobile to make a sound when an alert is shown.
- Radius distance (CL): sets the search radius for *hot zones*.

---

[6] Wordplay in Catalan. Untranslatable.

- Monitoring time (CL): defines how often users want the services to monitor *hot zones*.

The parentheses indicate to which level corresponds every item of configuration, conscious or automatic.

4.2.5 Discussion of the results

The aim of this initial test application was to investigate how ubiquitous computing could be useful in preventive health for HIV contagion in MSM. In particular for generating automatic detection of points of interest (POIs) through a geofencing approach and the detection of risky applications.

There is a need to develop new preventive health methods, and this work focused on georeferenced ubiquitous computing with mobile devices because of the popularity of these devices and their connectivity with the environment.

Although people live in a society where it is easy to access information, there are still many who are not aware of STIs and how they spread as a consequence of not taking appropriate preventive measures. In addition, mobile devices and applications now allow users to meet nearby partners and have *express* dates, giving them little time to think about safety measures. This work has proposed an application that provides users with information that promotes safe sex.

The Android application that has been developed sends a preventive message when the users: 1) run a contact application; or 2) approach a zone where sexual contacts are known to take place regularly (*hot zone*). Moreover, a module of this application helps the users find places where they can acquire preventive measures.

The application works in background and users are unaware that it is running (it runs at an automatic level). However, users have to install the application, i.e. its use is voluntary. The target of the application are, therefore, people who are already conscious about STIs but forget about safety measures when using mobile devices to meet partners. The messages displayed do not try to prevent the behavior itself, but rather to inform the users of preventive measures that should be taken into account. The final decision is then up to the users.

In addition, there is another functioning mode called the *conscious level*, which allows users to detect *hot zones* and avoid them. This functionality could also be used to find and go to these zones, although, since the software targets people who are concerned about their sexual health, this is not expected to be its main use. Nevertheless, in case it is used as a finder, the application can help to take the appropriate measures before going to that zone. Volunteers were asked to choose the messages they found most and least suitable as preventive messages, and were also offered the chance to propose new messages. This is very important since the messages must be contextualized to a special circumstance or target group.

The test results suggest that this type of application could help to reduce the high incidence of STIs, including HIV infection. The main advantages of this application are: 1) currently, it is common for people to use mobile devices for dating, so these devices are the most direct platform from which to launch preventive messages; 2) the preventive messages arrive at the most appropriate time, when the sexual contact is imminent, so that users are alerted at their moment of greatest vulnerability; and 3) users install the application voluntarily, so it is not expected to be perceived as intrusive, which is one of the main complaints of the target group.

Finally, it is important to note that the addition of new functionalities in this prototype is technically straightforward to implement, given that it has a modular architecture with high cohesion and low coupling between classes. This fact will be explained and taken into consideration in the next developments.

4.3 UBINUT

In this section, the second initial test application called UBINUT will be described. The application UBINUT was the second approach to address the issue of preventive messages in a smart context, focusing on testing and trying out the application with a group of people with nutrition problems. This contrasts with the previous section, which focused on UBIAPP and explored the issue of preventive messages in a smart context for the prevention of risky behavior and STIs.

The following structure presents the application, its development, testing, results, and an analysis of the results.

Firstly, UBINUT will be presented and the general aspects with its functionalities will be described. Secondly, the technical aspects of the application will be presented with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Thirdly, the context of the testing will be established with its results. Finally, the results will be analyzed, with discussion related to the optimization and the functionalities and attributes of the system as well as some conclusions and future work in this area.

4.3.1 Introduction

UBINUT follows the general diagram of the solution presented in Figure 4.1. In this stage of the research, a single application was developed for Android mobile devices with a version of the operating system 4.2.3 and a simple web information system.

The context taken into consideration was excessive weight and obesity, since these two are considered worldwide problems that are related to a variety of health issues. Several methods have been utilized recently to educate people and communities about topics related to nutrition behaviors. How can educational mechanisms be developed for healthy behaviors, focusing on a target group and their context?

To achieve the goal of developing educational mechanisms for healthy behaviors in nutrition, the proposed objective was to send health messages through mobile devices based on a smart context. To achieve this aim, UBINUT was developed.

UBINUT sends health messages to users supervised by a nutritionist. The underlying idea is to send messages related to a specific time of day, such as breakfast, lunch or dinner.

Moreover, it is necessary to consider other factors, for instance, the weather conditions or holiday festivities, among others variables. Thus, users won't get a message encouraging them to do sports outside when it is raining, or tips to have a healthy breakfast at lunch time. Therefore, the application works in two main lines:

- Sending health messages to users supervised by a nutritionist.
- Sending and receiving feedback for the preventive messages at the exact time that they are received.

UBINUT has two main components which are:

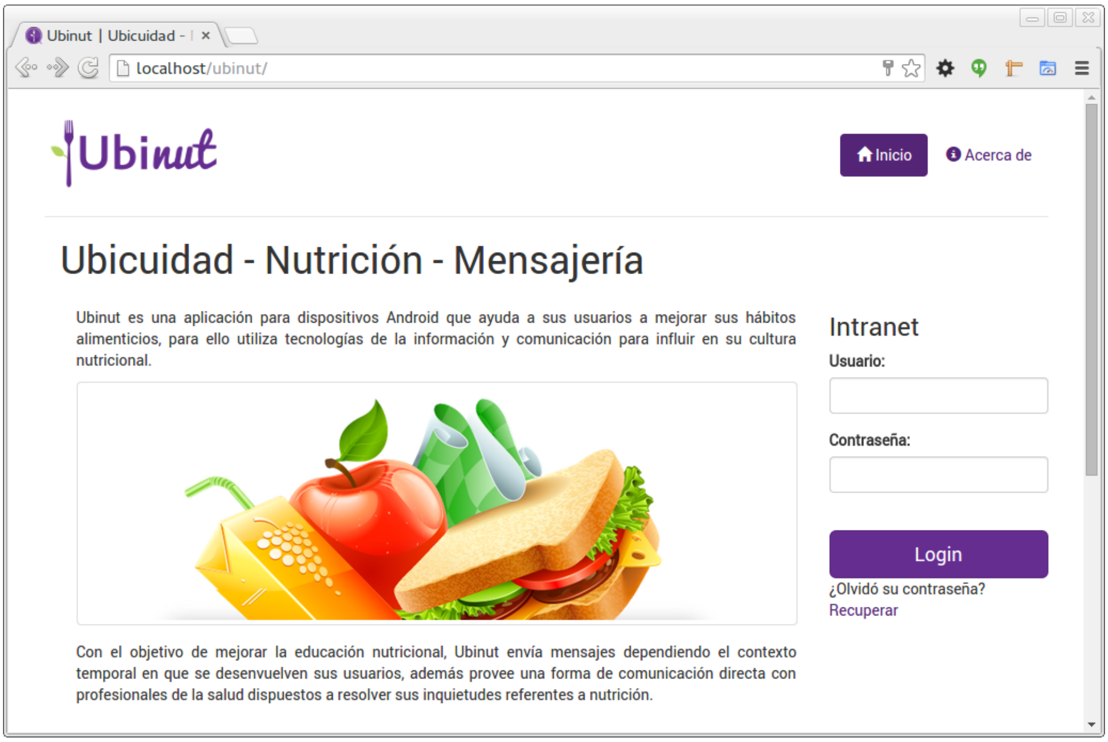
1. Simple web information system (SWIS): through this SWIS, the nutritionist can write and send messages to all the users (mobile clients) at a specific time. Figure 4.11 shows the web interface that the nutritionist logs into to send messages.
2. Mobile application: the mobile application will receive the messages at a specific time. Then, the message will be ranked by users on a scale from 1 to 5 using the Android ranking system.

The mobile application has two main functionalities. First, it notifies users when the nutritionist has sent a new health message, and second, the application allows users to send private questions to the nutritionist in order to get answers from a health professional. It is important to note that all health messages are retrieved from the stored data on the SWIS.

Figure 4.12 shows three states of the application at different times (from left to right):

1. Once the application is opened, the users will see the splash logo. Meanwhile, the application is checking internet connectivity.

2. The users can see the last message received with its score. It is important to note that new messages are received at any time.
3. Users can send private questions to a nutritionist at any moment. These questions can be categorized and replied to through the SWIS.



Ubicuidad - Nutrición - Mensajería

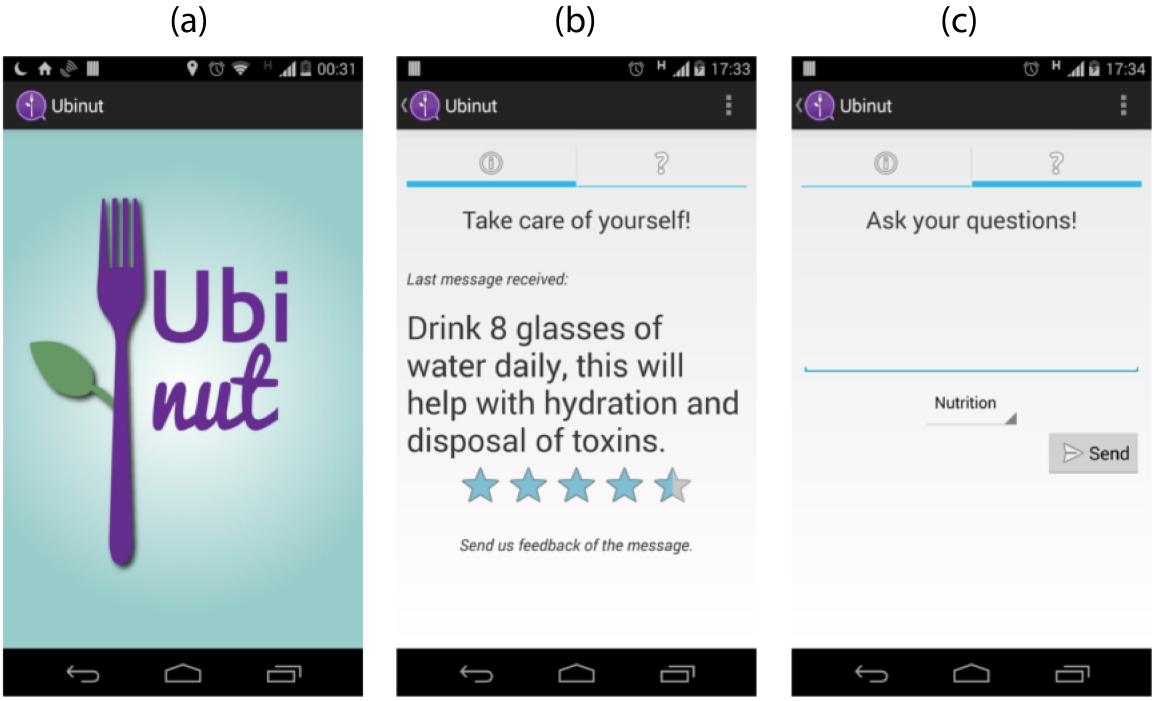
Ubinut es una aplicación para dispositivos Android que ayuda a sus usuarios a mejorar sus hábitos alimenticios, para ello utiliza tecnologías de la información y comunicación para influir en su cultura nutricional.

Con el objetivo de mejorar la educación nutricional, Ubinut envía mensajes dependiendo el contexto temporal en que se desenvuelven sus usuarios, además provee una forma de comunicación directa con profesionales de la salud dispuestos a resolver sus inquietudes referentes a nutrición.

Figure 4.11: UBINUT - Web client: through this interface, the nutritionist can write and send messages to all the users (mobile clients) at a specific time.

As an example of how the application works, a general workflow of the application will be described. To begin, it is necessary to emphasize that context is crucial to encourage awareness in the users. In this case, the context is driven by a nutritionist. Therefore, once the users have installed UBIAPP, the application will be running in the background all the time with the objective of receiving a message from the nutritionist through the SWIS.

Consequently, once the nutritionist sends a message through the SWIS, the application will receive and send a notification to the users (this notification uses the notification manager service of the Android operating system, and is presented on the notification bar, see Figure 4.12 screenshots a, b and c). When the users open the notification, the application will prompt them to rank the message out of five stars. Figure 4.12-b shows a message with the score of 4.5. After the ranking process, the application sends this score to the SWIS where the nutritionist can see the average and the highest scored message.



Take care of yourself!

Last message received:

Drink 8 glasses of water daily, this will help with hydration and disposal of toxins.

Send us feedback of the message.

Ask your questions!

Figure 4.12: UBINUT - Android client: the software gets a new message from the web server and notifies the users allowing them to rate the received message.

This functionality was taken into consideration to have a way to measure and categorize messages. Thus, in future uses of the application, the nutritionist will know how the group of users are reacting to the health messages. The ranking process creates a positive interaction with the users because they will rank the message at the exact time that it is received, collecting the first reactions to the intervention. The expectation was to receive more positive rankings when the message was in line with the context, and lower rankings when the message was out of context.

Another important feature is that the users have the opportunity to ask questions to a nutritionist (see Figure 4.12-c). This is important for two main reasons:

1. It contributes to the interaction and user experience of knowing that the application is not 100% automatized.
2. It creates a databank of issues that allows the nutritionist to identify the major information missing on certain topics that can be used in future health messages.

Finally, it is important to note that according to the methodology and the design of the research, the main focus of this application is to work with dynamic health messages, contextualized through a health professional and considering light interaction with the users.

4.3.2 Summary

The UBINUT system has two major components: 1) the simple web information system , and 2) the mobile application developed for Android operating system version 4.2.3. The first component allows the nutritionist to write and send messages to all the users (mobile clients) with specific timing. The proposed application is non-intrusive because: 1) the users themselves decide to install it; 2) the ranking of messages is anonymous; and 3) it works in the background without interfering with users.

Ubiquitous computing in general, and smartphones in particular, could be useful and contribute to improve nutrition behaviors. Smartphones are considered useful to this end because they are very popular, have an internet connection, and are constantly carried by users and used in multiple contexts. UBINUT sends health messages based on users' context, when the users are more likely to make poor nutritional decisions (specific context).

As mentioned in the introduction to UBIAPP and the overview of the structure for this section, in the next subsection, the technical aspects of the application will be presented. Then, the context will be established of the testing of its results.

Finally, the results will be analyzed, some aspects related to the optimization and the functionalities and attributes of the system will be discussed, and some conclusions and future work in this area will be presented.

4.3.3 Technical aspects

The following subsections will describe the IT artifacts that were involved in the software development process for UBINUT.

4.3.3.1 Actors

The main actors that interact with the UBINUT system are: mobile user, server user, database, information server, and mobile application. It is important to note that the server user will take a relevant role in the system. This role will be described in the following sections. The same actors as those from UBIAPP have been defined because the domain of the problem for both test applications is the same. For further description, see 4.2.3.1.

4.3.3.2 General use cases

To understand how the software behaves, two general use cases will be presented. These use cases briefly explain each process and its interaction with the various solution components, which will also be outlined in this section. The first use case triggers a message when the application detects a new message in the database of the system; the second use case sends a direct message to the nutritionist from the mobile application. More details on use cases *UN01: General use of the system UBINUT* and *UN02: Sending a private message to the nutritionist* are provided in Tables 4.8 and 4.9, respectively.

Use case UN01:	General use of the system UBINUT
Actor(s):	Mobile application, information server.
Purpose:	To detect a message stored in the information server and notify the user with it.
Summary:	This use case begins when the nutritionist sends appropriate health messages through a website. The *notification service* of the Android application detects a new message on the information server. The message is downloaded using a RestFul service. Then the message is communicated to the user, through the *notification manager*. The user receives the message and then has the opportunity to rank it using a system of one to five stars; furthermore, the user can send questions to the nutritionist, and the nutritionist can respond to these questions.
Preconditions:	1) The smartphone should be connected to the internet; 2) a nutritionist should post a new message on the web system taking into consideration the previously defined contexts.

Actor's actions	**Application's answers**
1.- This use case starts when the nutritionist sends a new message through the information system	
	2.- The *update service* detects that a new message has been provided (this requires internet connection from the device to the web server).
	3.- It notifies users of a new message through the notifications bar
4. - Users select the notification.	
	5. - The system opens a window with the message and provides the opportunity to rank it using a system of one to five stars.
6. - Users can rank the message according to the star system.	
7.- Users can send a direct message to a Nutritionist (see use case UN02)	

Alternative flow: 2.- The *update service* requires access to the internet. In addition, it is necessary to point out that the service checks the system again after X amount of time.

Table 4.8: UN01: General use of the system UBINUT in a conversational format, which emphasizes the interaction between the actors and the system.

<table>
<tr><th>Use case UN02:</th><th colspan="2">Sending a private message to the nutritionist</th></tr>
<tr><td>Actor(s):</td><td colspan="2">Mobile application, information server.</td></tr>
<tr><td>Purpose:</td><td colspan="2">To send a message to the information server.</td></tr>
<tr><td>Summary:</td><td colspan="2">This use case begins when the user wants to send a private question to a nutritionist or health professional. The user will write a message on the mobile application. The message is sent using a RestFul service. Then the message is received by the Information server, where it can be reviewed by the nutritionist.</td></tr>
<tr><td>Preconditions:</td><td colspan="2">1) The smartphone should be connected to the internet; 2)The user must write a message on the mobile application.</td></tr>
<tr><td colspan="2">Actor’s actions</td><td>Application’s answers</td></tr>
<tr><td colspan="2">1.- This use case starts when the mobile user sends a new message through the mobile application (internet connection from the mobile device is required)</td><td></td></tr>
<tr><td colspan="2"></td><td>2.- The <i>information server</i> detects that a new message has been provided.</td></tr>
<tr><td colspan="2"></td><td>3.- The message is stored on a local database</td></tr>
<tr><td colspan="2">4. - Users are notified that the message has been sent</td><td></td></tr>
<tr><td colspan="3">Alternative flow: 1.- The <i>mobile application</i> requires access to internet. If the mobile application does not have an internet connection, the mobile application will try to deliver the message later ; 4.- The message cannot be sent, and the user is notified of this situation.</td></tr>
</table>

Table 4.9: UN02: Sending a private message to the nutritionist use case in a conversational format, which emphasizes the interaction between the actors and the system.

Based on both general use cases, the following aspects are taken into consideration:

- Creation and storage of the messages
- Development of a web service for multiple clients
- Generation of a rank system for the messages (Likert scale)

4.3.3.3 Sequence diagrams

In this section, the two main *sequence diagrams* will be discussed that correspond to the two main general use cases of the system, see Tables 4.8 and 4.9. Besides that, some references to source code will be shown when it is required.

In UBINUT, the way the application interacts with the users is refocused with respect to UBIAPP. As mentioned in the introduction to UBIAPP, the goal of this second test application is to focus on dynamic health messages contextualized through a health professional. Hence, considering the experience from UBIAPP, it was concluded that getting instant feedback from a health professional is extremely necessary for generating a strong relationship between the user and the system, which improves the UX. Taking this into consideration, the decision was made to develop a system that allows the administrator of the system (in this case the nutritionist) a way to communicate massively with the users. This could be done with these technologies:

1. Using a Push / Pop service
2. Sending text messages
3. Using a social media platform such as Whatsapp, Facebook or GoogleTalks.

In this research, a system of message exchange based on the RestFul web service has been developed. The reasons behind this decision were: firstly, to avoid the prices of acquiring a PUSH/POP service and text messages; and secondly, to avoid forcing the users to use a social media platform (these platforms are massive, but not all users like all of them). This web service will be explained in the following diagram, and the *update service* from the Android application will be described in the next state chart diagram.

The following sequence diagram represents a black box interaction between the main components

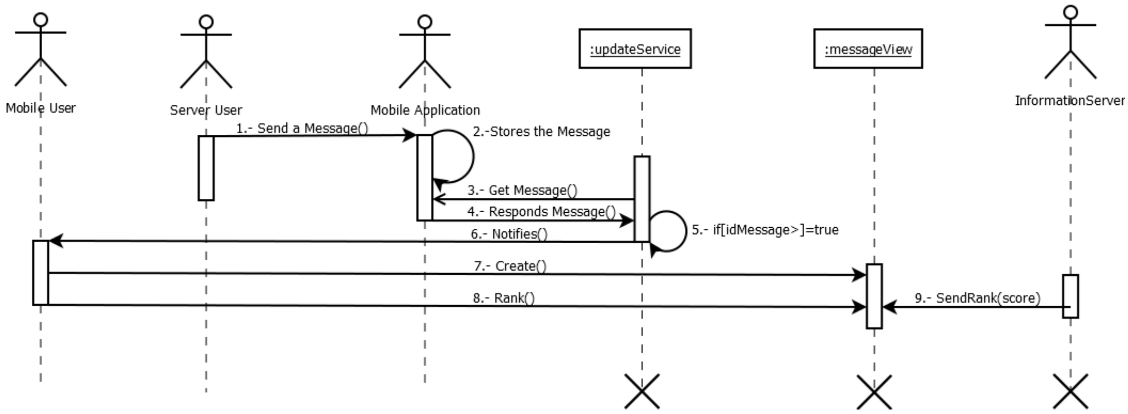


Figure 4.13: Sequence diagram: black box representation of the general use case UN01 *General use of the system UBINUT*, see Table 4.8

As can be seen in the sequence diagram of Figure 4.13, there are four main actors that are involved in this interaction: 1) mobile user, server user, mobile application and the information server. A figure was developed that represents from a graphical user perspective these interactions defined as a process (for complementary information See Figure A.1 in the Appendix).

The following description corresponds to the main process described in the sequence diagram of Figure 4.13. In this context, a RestFul web service was developed to allow the nutritionist, known as the *server user*, to send messages to the users. Therefore, the first process is when the nutritionist writes and stores a message on the *information server*. Once the new message is stored in the DBMS of the *information server*, the restful web service will automatically reply with the last stored message every time that a client (*mobile application*) requests a message. Hence, every time that the *update service* requests a new message, the restful service will reply with the last message id and the message, for complementary information see Figure A.2 from the appendix. The *mobile application* will compare the id of the message stored in the server with the id stored locally. If the id of the message stored on the server is greater than the id stored on the *mobile application*, then it will notify the user. It will deliver a message to the *notification manager*, which has the responsibility of showing it to the mobile users through the notification bar. After that, the users have the opportunity to rank the message on a scale from 1 o 5. Once it is ranked, the *mobile application* will send that data to the *information server*.

The ranking system is a major improvement in the research since initially messages were sent based on a specific context, but rapidly more questions needed to be addressed, for example:

- How can the application know if the message was relevant to the context?
- How can the application select the messages for the users?
- Etc.

Taking these questions into consideration, this ranking system was developed using the Restful principles. It is easy, quick and focuses on the user experience of the application.

The Figure 4.12 represents this use case from a graphical user interface (GUI) perspective on the mobile phone. The GUI represents what the mobile users see, but the last description in the sequence diagram of the Figure 4.13 shows what the application does.

The second main functionality represented by UN02 is sending a private message to the nutritionist (see Table 4.9). This use case will use the same module of communication with the server as UN01 (see Table 4.8). Moreover, once the mobile user has written the message on the mobile application, this will be sent by HTTP/POST protocol to the *information server*. The information server will store the message in the database for later review by the nutritionist. The sequence diagram 4.14 shows this workflow in a black box format.

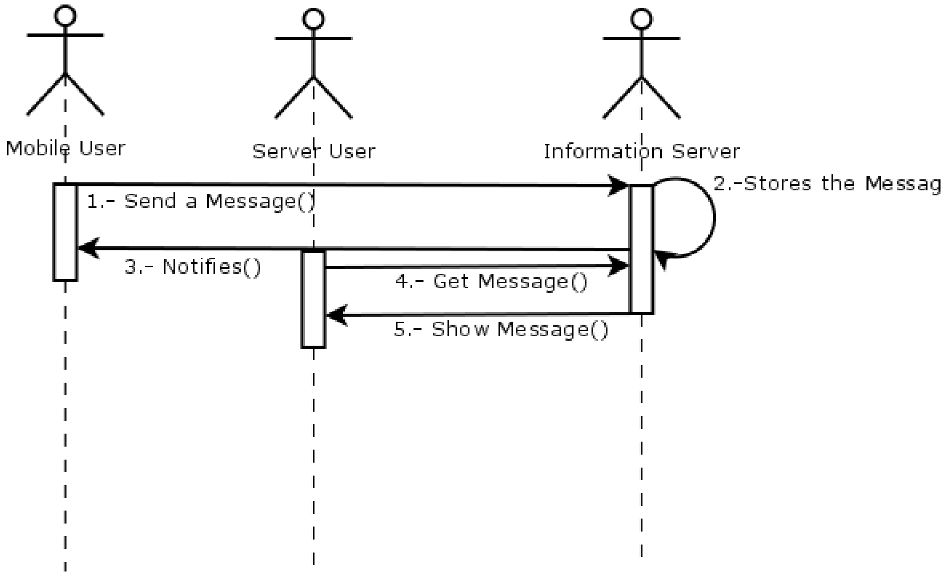


Figure 4.14: Sequence diagram: black box representation of the general use case UN02 *Sending a private message to the nutritionist*, see Table 4.9

The nutritionist will reply to the users through email. At this point of the research, a third module for the application was not implemented in order to receive messages. This is because the decision was made to take advantage of the services that are already provided by the mobile device and Android OS. This approach enhances the UX of the mobile application because the information that is provided can be accessed from different sources such us mobile or PC browsers, a mobile client, etc. On one hand, this strengthens the ubiquity of the information, and on the other hand, lessens the workload for the mobile application, thus optimizing the available resources.

4.3.3.4 Statechart diagram

In the following subsection, a statechart diagram will describe the different states for the update service, which is responsible for getting new messages from the information server on the mobile device. As was explained before, the statechart diagrams describe the flow of control from one state to another state. States are defined as a condition in which an object exists, and it changes when a certain event is triggered.

As was mentioned in the general use case UN01, a key feature of UBINUT is getting health messages provided by a nutritionist on the mobile device. The update service works as Figure 4.15 shows. Once the service is created, it will continuously request the last message from the information server. The information server will reply with an id and the message in a plain text format (all this data is transferred using a RestFul service). The id is compared with the current message id on the device. Therefore if the id from the information server is larger than the id from the mobile device, it is concluded that a new message is on the information server and the message is delivered to the users on the notification bar.

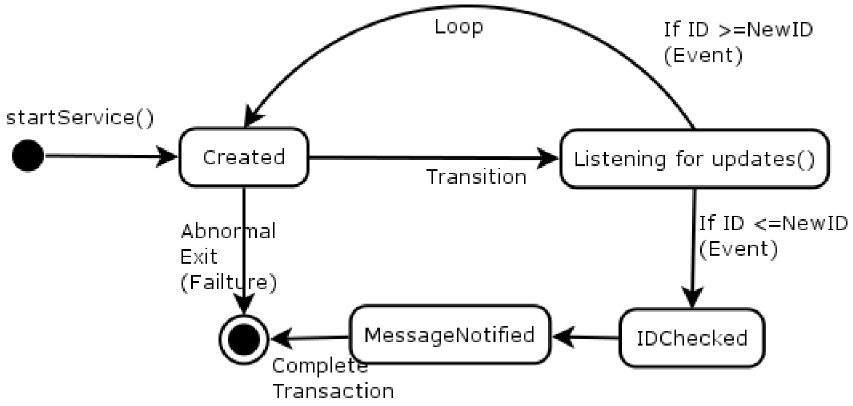


Figure 4.15: StateChart update service: Diagram related to the general use case UN01 *General use of the system UBINUT*, see Table 4.8

Considering this scenario, the statechart diagram has four states that are described as follows:

- Created: in this state, the process has access to the local preferences and the internal database for getting data such the last id message.
- Listening for updates: here the service is running in its active lifetime, getting the available data from the information server.
- IDchecked: the ids from both the mobile application and the information server message are compared.
- MessageNotified: the message is delivered to the users. Once this happens, the last internal id message is updated.

4.3.3.5 Architecture of the solution

UBINUT uses the same RESTful service principles that were explained in the UBI-APP architecture of the solution section (see Section 4.2.3.5).

Figure 4.16 shows the conceptual architecture that integrates all components and services of the proposed solution, according to the REST principles. This system has two components:

- The web component (server): this is mainly a web server for managing the application data, and includes a web client.
- The mobile component (Android): this is mainly an Android client.

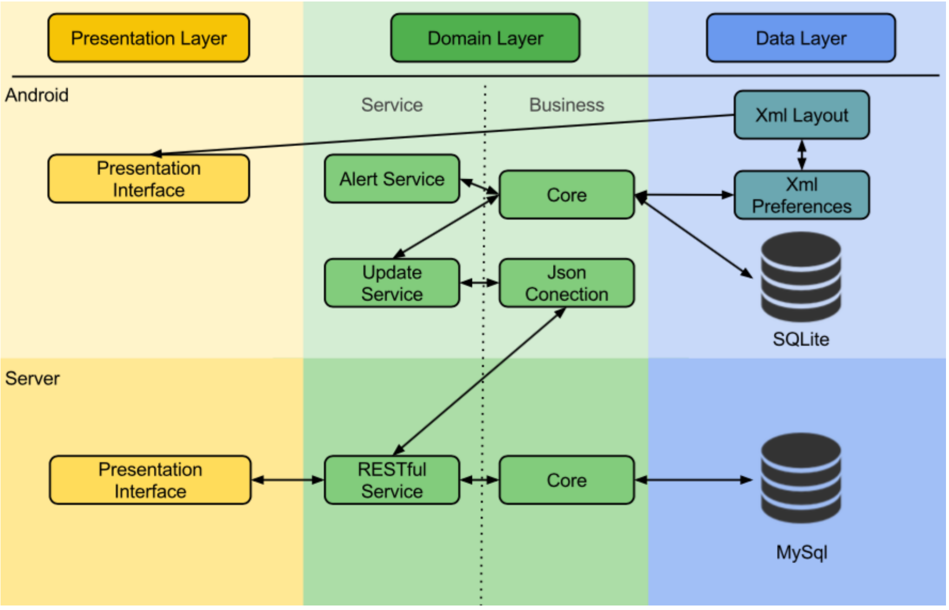


Figure 4.16: Implemented architecture and essential components and services. The architecture consists of three main layers: the presentation, domain and data layers, representing the interaction between the essential components of the solution. This representation is divided between the Android client (top) and the web service (bottom). The re-used open source components in the proposed architecture are also shown.

As described in the general use of the system UN01, users install the application, which then works in the background as a service that is imperceptible to the users. Through a website, the nutritionist sends appropriate health messages taking into account the context in which they will have the greatest effect. The users receive the message and then have the opportunity to rank it using a system of one to five stars; furthermore, users can send questions back to the nutritionist and the nutritionist can respond to these questions. Both, the ranking of the messages and the questions from users are saved on a web server, for complementary information see Figure A.1 from the appendix.

The system has three main components: the UBINUT application, the web interface and the database.

- The first component, the UBINUT application, works in the background as a service on Android smartphones. This means that users do not have to open the application to make it run, see it or get any information from it. Rather, it is running all the time (here the update service is a key component). Thus, the application is able to receive ubiquitous messages from the web interface. New messages are gotten from the server on the smartphones at any time. Another feature of the application is that it allows users to send questions to the nutritionist in charge of the study.
- The second component, the web interface, is a website that allows the nutritionist to post a message in reply to the questions that users ask, see Figure 4.11.
- The third component, the database, stores the data more specifically, when the nutritionist posts a message on the website, all the smartphones will recognize the message as new and they will deliver it to the owner. Furthermore, the database stores users' perceptions of the received messages. This means that once users receive a notification, the application prompts them to score the message on a scale from 1 to 5; and this information is then stored in the database. This allows the nutritionist to get perceptions in real time and analyze the data for future studies. Figure 4.12 shows a message on the mobile device and the web interface of the system.

- **The domain layer of the web component**: The domain layer is subdivided in two parts: the service section and the business section. The domain layer is considered to be the most important layer of the mobile application because the update service from the domain layer and its components establish the communication between the information server and the mobile application.
  - **The service section in the web part**: The service section implements a RESTful service that processes requests and communicates with the business section. Note that this is the section that ultimately interacts with clients associated with the service. This architecture provides a solution with greater interoperability and modularity. Therefore, the architecture will not be affected by eventual transformations on the server or client side.

    Usually, client applications request resources from the server using GET methods from the HTTP protocol. These resources are identified using a URI. The HTTP protocol is used in RESTful services to interact with the resources of the server. This is very beneficial for mobile clients because of their limited resources. Using this design, the client load will be lightened by turning it over to the server, which has more resources.

    The RESTful service receives a request from the Android client (*update service*). The request is a URI.

    Using this information, the RESTful service communicates with the persistent data layer on the server (See Figure 4.16). More specifically, it communicates with the database in order to get the latest message and message id.
  - **The business section of the web domain layer**: The business section from the server has the core of the application, which executes routines that interact directly with the data layer.
- **The domain layer of the Android client**: As for the web component, the domain layer is divided in two sections: the service section and the business

section.

- **The service section of the client domain layer**: The service section implements an internal monitoring service known as update service. This service is responsible for detecting and getting new messages from the information server. Simultaneously, it interacts with the business section.

  The update service request and receive a message through the RESTful service on the information server via the business section. With this information, the RESTful service communicates with the server's persistent data layer (See Figure 4.10). More specifically, it communicates with the database in order to get the last message provided by the nutritionist.

  As was previously described in the statechart diagram of the update service (Figure 4.15), the system controls the messages by an id that is incremented automatically every time that the nutritionist sends a new message. Once the message is detected, the core from the business client side will handle the notification and also the new latest id message in the *XML preferences*. In this case, this component is better for storing the message state because only one datum changes its state. Therefore, it is not necessary to query dynamically a database for obtaining these data (making the process for obtaining the data more efficient).

- **The business section of the client domain layer**: It is important to note that there are two cores in the business section of the client domain layer, one corresponds to the server and the other corresponds to the Android application. Both cores are in the domain layer. Previously the core that corresponds to the server was discussed. In contrast, here, the core that corresponds to the Android application will be discussed. The core of the Android application interacts with *JSonParser*, and is responsible for sending messages from the notification and update service to the RESTful service of the web component.

The *cl.ubinut.JsonParser* package has the classes required to make different *requests* to the *RESTful* service and process data, thereby facilitating their subsequent representation by other components of the software.

4.3.3.6 Optimization and key points

For UBINUT, the most critical process is the HTTP request made by the update service to the RESTful service. This process is made according to certain timing preconfigured by the users. Since the data traffic for each request is minimum, less than 200 characters, and all the processes are carried out in the background, the use of the battery is limited and optimized. It is important to mention that all the processes in the background such as services, tasks, or routines, among others, do not use the screen which is the component that uses the battery the most.

4.3.4 Testing and results

In this section, the results will be presented, showing how users interact with the software and afterwards, in the analysis of the results section, several conclusions will be drawn regarding how ubiquitous computing can be used as a preventive tool.

As a result of the research following the design and creation approach, a mobile Android application has been created with the focus on dynamic health messages contextualized through a health professional: UBINUT. Functional and technical aspects were described in the previous sections. To test the application and the rank message system, a test was performed with university students. This testing was carried out by both, professionals from health departments (University of Talca) and researchers of computer sciences, multimedia and telecommunications departments (UOC).

4.3.4.1 Testing context

The test has been carried out with the UBINUT mobile application. The study was designed according to the recommendations suggested by several reports for medical or therapeutic such us interventions as in (Bowen et al. [75], Leon et al. [76]), specially focused on practicality issues of applying such a ubiquitous system.

Forty volunteer university students were enrolled to participate actively in the feasibility evaluation of the UBINUT system. Ninety-five percent of the participants declared that they utilized social networks through their smartphones frequently (at least twice a day). Their age was between 18 and 28 years. Seventy percent of them were women. The Body Mass Index (BMI) distribution among participants was: 78% between 18.5 and 24.9; and 22% more than 25. Even though most of them had an under-risk BMI level, it is important to consider that the UBINUT system aims to have a preventive role more than a remedial one.

4.3.4.2 Data sources and collection

Data were collected through the smartphones owned by the users. All data obtained from the system was stored in the database as the foundation of the analysis. Moreover, the data were collected through the internet and analyzed concurrently.

Data were collected at three different times: first, through the entry survey; second, through the smartphones; finally, through an exit survey. The objective of the entry survey was to learn more about the target group, regarding knowledge, habits and healthy behavior. The objective of the exit survey was to validate the knowledge of the target group and collect software suggestions.

The application collects three types of data at different times:

1. Profile information: this information is collected the first time that users run the application, including: name, email and student ID number. Additionally, the age, weight, and height are requested. With these data, it was possible to assign students a profile based on the general data and BMI. It was also possible to gather information about their behaviors.

2. Perceptions: when UBINUT detects a new message in the system, it will notify the user in the notification bar of the smartphone. The user will be prompted to read the healthy tip and score it on a scale from 1 to 5. (See Figure A.5)

This process is done each time that UBINUT detects a new message available on the system. A nutritionist controlled the messages from the system.

3. Question to the nutritionist: the users were able to send questions to the nutritionist in charge. The nutritionist could review and answer the questions in the system.

The nutritionist has access to a website that shows the list of messages with their perceptions in real-time. It also allows them to see and respond to the questions sent by the users through the mobile application. Figure 4.11 shows the user interface of the website and the interaction with the list of messages and graphs of perceptions. These features add value to the system since the nutritionist/administrator of the system can see in real-time the positive or negative impact of a message sent to the users. Therefore, the nutritionist can evaluate if the messages are appropriate for the target group and improve them taking into account the users' contexts and needs.

4.3.4.3 Sources of health messages

The health messages used in this test come from two sources: 1) the national health guide "guías alimentarias para la población Chilena" [77] and 2) a group of professional nutritionists from the Universidad de Talca.

The origin of the national health guide is related with the needs to establish a set of public health messages that consider the current epidemiologic profile of Chilean society, with the aim to help people achieve a balanced diet and healthy weight. This action was taken by the Chilean minister in 2013 and the guide was distributed to local health and education centers.

The publication is composed of a set of 11 health education messages that adapt scientific knowledge about food and physical activity for the Chilean population, considering their current situation and socio-economic factors. To illustrate, the lifestyle of Chilean society has changed during the last ten years. Now, there is more purchasing power and access to information from different sources such as: television, internet, and social networks, among others. Although people are more informed about these topics, the information often comes from biased sources. For example, there is strong marketing from companies. In general, there is an increase in the intake of sugar-based beverages, high energy foods and eating at food courts or restaurants.

The methodology used by the government to create the national health guide draws from several national public health institutions (namely, the OMS, USDA, FAO, and CDC). In addition, focus groups were carried out with target groups of different socio-economic statuses, also including representation from the north, center, and south of the country. After the analysis, the government defined 11 messages based on the current epidemiologic and socio-economic profile of the country.

Since the national health guide has only 11 messages, this limited number was complemented by texts written by a group of nutritionists from the Universidad de Talca. They took into consideration the national health guide and its findings and defined three main areas in which to create the new messages: food, sports, and festivities. The first two areas (food and sports) were taken from the national health guide, with the idea of writing new messages that would complement those in the guide. The third area (festivities) was created based on the context in which the application would be tested, including the month (in which Chile celebrates its national holiday) and its typical traditions, the geographic area, and the socio-economic status of the participants. The messages were created based on the expert opinions of the nutritionists, considering their academic and professional experience. In total, 29 messages were written by the nutritionists.

4.3.4.4 Feasibility evaluation

In the feasibility evaluation, a nutritionist participated, writing the messages and responding to users questions. The nutritionist responded to the questions within 24 hours.

The study had a duration of 40 days; a total of 40 messages were delivered through the system (one per day) with different nutrition information tips. The messages were designed according to the specific context of the users, considering for instance the season of the year when the study was applied and public policy guidelines: i) the occurrence of the national independence holidays in Chile was taken advantage of, when people usually increase their food intake significantly and eat a great amount and variety of typical Chilean food, which includes the excessive consumption of fatty meat and alcoholic beverages. On these days special messages were designed and sent to users considering specific nutritional tips about this; and ii), during the 40 day period, a series of 11 messages were delivered that were closely related to a document prepared by the Chilean government as mentioned previously, with healthy living tips, which is the national health guide [77].

During the trial period 699 evaluations were received from users, which rated the different messages on a 1 to 5 Likert scale, obtaining an average of nearly 17 responses a day. The mean of the ratings received during the whole period was 3,89 indicating that the information sent to participants was highly rated in general. It is important to highlight those messages scored with the highest rating by the participants, considering only those with scores above average plus one standard deviation. These messages are described in Table 4.10. It was noted that messages delivered on national independence days (17th and 18th of September) were among the highly scored. This shows that users seem to appreciate messages with contextualized information with the time they are sent. Particularly, in this case the messages were associated with traditional foods and local traditions that take place during this period.

Regarding the sent messages related to the Chilean Government Food Guide, interestingly, it was found that 7 from the 11 tips contained in this guide were among the highly rated messages. This finding allows us to assess the real impact that the public nutritional guidelines have on participants, which seems to provide a positive opinion about them. It is noteworthy that only 50% of the participants declared to know about these guidelines previous to the feasibility study.

To assess the impact of using the UBINUT system, a survey was applied to all participants before and after the feasibility study was developed, see Table 4.11.

This was done in order to identify potential changes in behavior, attitudes and knowledge about nutrition and healthy living. The results of the surveys suggest

Message sent "txt"	Average scoring
1 hour of aerobics burns between 270 and 350 calories (N)	4,6
To stay hydrated, drink six to eight glasses of water a day. Water is essential to maintain body temperature and fluid balance (N)	4,6
Take care of your heart avoiding fried and fatty food such as sausages and mayonnaise. Consumption of saturated fat is the main risk factor for cardiovascular disease (G)	4,5
Barbecues and sausages must be combined with salads and leafy green vegetables, not potatoes or bread (N)	4,5
Traditional dances are also considered as aerobic exercise. If you dance an hour, you are burning the equivalent to a *choripan* (type of fatty hot dog) hot dog or a glass of *mote* con *huesillos* (a high sugar beverage) (N)	4,4
Eat 5 vegetables and fruits of different colors through the day. These are considered a great contribution of fiber, vitamins, minerals and bioactive compounds that are beneficial to your health.	4,4
Read and compare food labels and prefer those with less fat, sugar and salt (sodium).	4,4
To keep your heart healthy, eat fish baked or grilled, twice a week. Fish is an important source of protein, iron, zinc and fatty acids of the Omega 3 type.	4,4

Table 4.10: Messages sent by the system scored with the highest rating by the participants. Messages sources: N = UBINUT nutritionist, G = Government guide.

Topic	Perception before	Perception after
Knowledge about the Chilean population food guide	50%	80%
Quantity of calories in fast food	40%	80%
Relationship between exercise and diet	70%	95%
Relationship between aerobic and anaerobic exercise to lose weight	45%	60%
Healthy food alternatives to fast food	25%	50%
Recipes for foods with natural ingredients	30%	50%

Table 4.11: Impact of the application during the trial experiment with the target group from entry and exit surveys

that participants seem in general to improve their knowledge about healthy food and healthy life habits. The knowledge increased between 15% to 40% after using the UBINUT system.

Based on the feedback from users, two points of interest were identified: 1) the time when messages were received, 2) the location where messages were received.

As proposed earlier, timing is the key for a message to have a positive impact on the users. The UBINUT system delivers the messages through a RESTFul service, and therefore the smartphone needs to be connected to the internet in order to receive the message. If the smartphone is not connected at the time the message is sent, then the users will get the message as soon as it gets connected. However, the issue of being connected was minor since the target group was university students that spent most of their time on campus, which is fully covered with wireless internet connection.

An important improvement that was noted that needed to be made to UBINUT is related to the places where users receive a message. The geographical location, though not included in UBINUT, provides a special context in which users could make decisions about healthy eating. This context is when they are going to eat or pass places such as a cafeteria, supermarket, fast food windows, etc. In all these places, because of the publicity and advertisements, users could make unhealthy decisions. Since this last point is not considered in UBINUT, it is incorporated into the refined application GEONUT.

4.3.5 Discussion of the results

The aim of this research was to investigate how ubiquitous computing in general, and smartphones in particular, could be useful and contribute to improve nutrition behaviors. Smartphones are considered useful for this goal because they are very popular, have internet connection, and are constantly carried by users and used in multiple contexts by them. Therefore, UBINUT has been developed to improve nutrition behaviors. The application works in the background and users are unaware that they are running (ubiquitous context) on their mobile device. They send context aware messages to users.

UBINUT sends health messages considering two special contexts: time and weather. The messages are posted by a nutritionist, who decides which message to send and when. Moreover, users are able to rate each message once it is received and ask direct questions to the nutritionist. The target of the application is, therefore, people who are already conscious about the need to develop healthy habits and nutritionists who want to try a different method to hold the attention and to engage their patients. The focus of the messages is on making users aware of healthy habits in a special context, but it is important to note that the final decision is then up to the users.

From the test developed, it can be seen that the majority of users participated in the ranking of messages, suggesting that they valued participating actively (otherwise, they would have skipped the step of ranking messages). It was also seen that messages must be related to users' contexts and daily routines to achieve the proposed goals. Thus, users feel comfortable and receive the advice in a positive way. Based on the feedback from the exit survey, they also value positively the interaction with the nutritionist. Therefore, the conclusion was it is important that, to be useful, this kind of applications have some active participation from the users: they like to be not only a passive receptor of messages, but also interact with them. However, the main drawback of UBINUT is that the nutritionist cannot send the message when the users are near a restaurant or a cafeteria, where it is important to be aware of healthy food.

The system contributes to efforts to promote healthy lifestyles among young adults, taking into consideration their context. This has a significant impact on knowledge about nutrition topics within a target group. As a consequence, users are able to make better decisions in their daily routine, taking into account information and aspects previously unknown to them.

There is a high rating for messages that took place on the holidays and for messages from the Chilean population food guide. This suggests that the messages' context is very relevant for the target group. Thus, people feel identified and become more aware of their actions. It is important to note that messaging the users more than once per day could be invasive. Users receive different messages all day from companies that offer special deals to a random public and also notifications from social networks.

In conclusion, as previously discussed, there is a challenge for the nutritionist to keep the attention and to engage the users. Based on the results from the surveys and rankings, messages must be related to users' contexts and daily routines to archive the proposed goals. Thus, users feel comfortable and receive the advice in a positive way (ranking the messages highly). Additional research must be done to determine the influence and the impact of ubiquitous technology on the behavior of the users, including new variables related to their position, actions, environment and daily routine. More research in these areas was carried out in the development of the refined version of UBINUT, which is GEONUT.

4.4 GEONUT

In this section, the refined application from UBINUT will be described, which is called GEONUT. The application GEONUT was a part of the second approach to addressing the issue of preventive health messages for nutrition. This application focuses on testing and trying out preventive messages in a smart context with a group of people with nutrition problems. It also includes concepts of geofencing for the recognition of hot zones associated with places where users can make poor nutritional decisions, such as food courts, cafeterias, street food, etc. The refined mobile application considers the previous experiences with UBIAPP and UBINUT from a technological point of view, and the test for GEONUT was carried out based on the conclusions from these experiences.

The following section presents the application, its development, testing, results, and analysis of the results.

Firstly, GEONUT will be presented and the general aspects will be described with its functionalities. Secondly, the technical aspects of the application will be presented with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Thirdly, the context of the testing will be established with its results.

Finally, the results will be analyzed, with discussion related to the optimization, the functionalities and attributes of the system as well as some conclusions and future work in this area.

4.4.1 Introduction

GEONUT follows the general diagram of the solution presented in Figure 4.1. In this stage of the research, a single application was developed for Android mobile devices with a version of the operating system 4.2.3 and a simple web information system.

The context taken into consideration was the results of the test developed for UBINUT. From that test, it can be seen that the strong points of UBINUT that were evaluated favorably by participants in the surveys, are:

- Active participation with the application (rating of the messages)
- Messages related to users' contexts and daily routines
- Interaction with the nutritionist

The weak points of UBINUT are:

- The application cannot recognize the users' location to be able to provide relevant information for that zone
- The nutritionist cannot send messages when users are near a restaurant or a cafeteria, where it is important to be aware of healthy food

These drawbacks can by mitigated by using the previously introduced concept of *hot zones* in UBIAPP. The informatics modules developed were used and refined with this purpose.

To achieve the goal of developing educational mechanisms for healthy behaviors in nutrition, the proposed objective was to send health messages through mobile devices based on a smart context including concepts of geofencing for an automatic notification related to geographic position. To achieve this aim, GEONUT was developed.

GEONUT is an improvement to the main functionalities of UBINUT. Therefore, it sends health messages to users supervised by a nutritionist. In addition, it will automatically recognize certain hot zones related to places where the users can make poor nutritional decisions, notifying them to raise awareness of the situation.

The underlying idea is to send messages related to a given moment of day, such as breakfast, lunch or dinner, also according to certain geographic positions. The application has light interaction with the users, offering dynamic feedback of the received messages.

Therefore, the application works in two main lines:

- Sending health messages to users supervised by a nutritionist

- Sending health messages to users when the application detects proximity to areas with a high probability of making poor nutritional decisions, known as hot zones

GEONUT has two main components, which are:

1. Simple web information system (SWIS): through this SWIS, the nutritionist can write and send messages to all the users (mobile clients) at a given moment (functionality taken from UBINUT). Figure 4.17 shows the web interface that the nutritionist logs into to send messages.
2. Mobile application: the mobile application will receive the messages based on a smart context: 1) messages sent by the nutritionist; or 2) messages sent because of proximity to areas with a high probability of making poor nutritional decisions. After receiving the message, it will be ranked by the users on a Likert scale from 1 to 5 using the Android ranking system.

Ubicuidad - Georeferencia - Nutrición

Geonut es una aplicación para dispositivos Android que ayuda a sus usuarios a mejorar sus hábitos alimenticios, para ello utiliza diversas tecnologías para influir en las desiciones que toman las personas al consumir alimentos.

Para ello se indentifican zonas de riesgo, correspondientes a lugares donde una persona puede caer en conductas poco saludables desde un punto de vista nutricional (locales de comida rápida, heladerias, bares, etc.), Geonut es capaz de detectar la proximidad de dichas zonas y aconcejar a sus usuarios con mensajes de vida saludable.

Hábitos

La comida es nuestra fuente energía, y como tal es necesaria periódicamente, es por esto que las personas desarrollan hábitos alimenticios. Geonut pretende guiar a sus usuarios hacia un estilo de vida saludable.

Ubicuidad

Los teléfonos inteligentes (Android) nos permiten introducirnos en la vida de sus usuarios, Geonut utiliza la omnipresencia de esta tecnología para implementar un producto poco invasivo y que ayuda a generar un cambio de conducta.

Descarga

http://goo.gl/RBxCa2

Intranet

Usuario: admin

Contraseña: •••••••

¿Olvidó su contraseña?

Figure 4.17: GEONUT - Web client: through this interface, the nutritionist can write and send messages to all the users (mobile clients) at a specific time.

The mobile application has two main functionalities:

1. It notifies the users when the nutritionist has sent a new health message.
2. It detects proximity to areas (hot zones) with a high probability of making poor nutritional decisions.

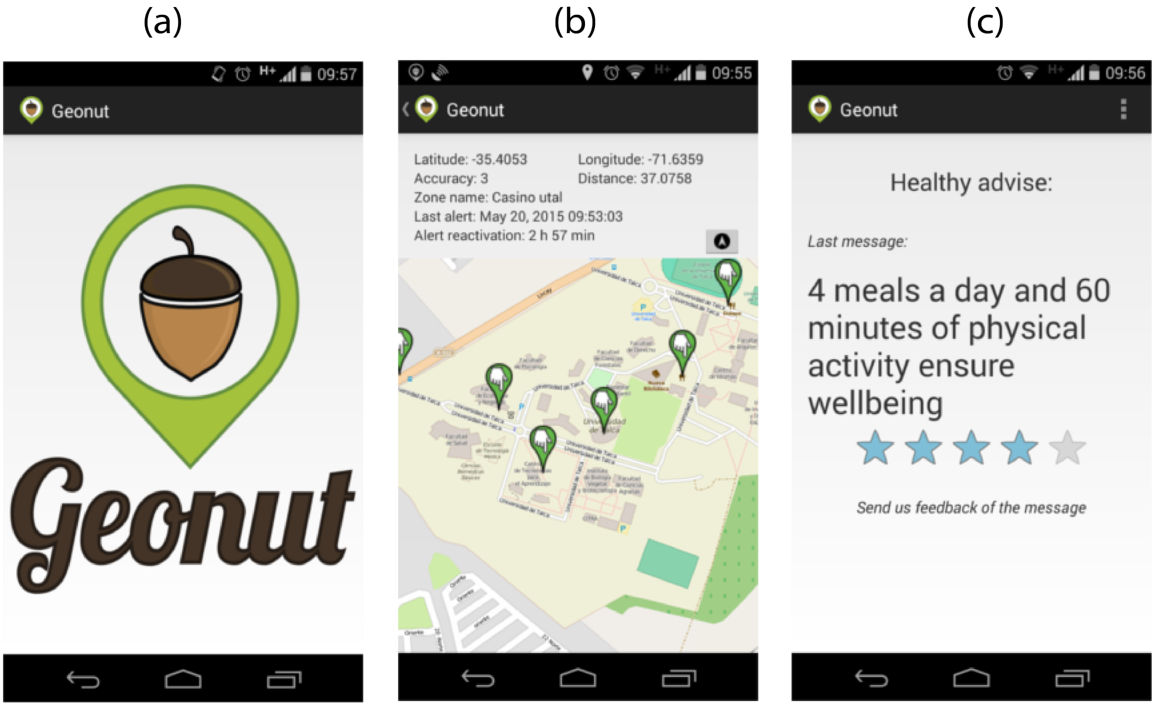
It is important to note that all health messages for the first functionality are retrieved from the stored data on the SWIS, and all the health messages for the second functionality are retrieved from a local database.

Figure 4.18 shows three states of the application at different times (from left to right). First, once the application is opened, the users will see the splash logo. Meanwhile, the application is checking internet connectivity. Second, a mapView is shown with all the hot zones nearby their positions. In each hot zone, the users can get the name and distance from their position to that point (this functionality needs a geolocalization service like a network or GPS). Lastly, the users can see the last message received with its score. It is important to mention that new messages are received at any time.

As an example of how the application works, a general workflow of the application will be described. To begin, it is necessary to emphasize that context is crucial to encourage awareness in the users. In this case, the context is driven by a nutritionist and the automatic recognition of the hot zones. Therefore there are two main workflows related to the contextualized messages to consider:

- Once the nutritionist sends a message through the SWIS, the application will receive and send a notification to the users. This functionality was taken from the previously developed application UBINUT. For further details, see Table 4.8

- Once the users are nearby a hot zone, the application will automatically recognize this situation and send them a notification. This functionality was taken and improved from the previously developed application UBIAPP. For further details, see the section 4.2.3.2.



Latitude: -35.4053
Longitude: -71.6359
Accuracy: 3
Distance: 37.0758
Zone name: Casino utal
Last alert: May 20, 2015 09:53:03
Alert reactivation: 2 h 57 min

Healthy advise:

Last message:

4 meals a day and 60 minutes of physical activity ensure wellbeing

★★★★☆

Send us feedback of the message

Figure 4.18: GEONUT - Android client: the software gets a new message from the web server every time that it detects a hot zone and notifies users allowing them to rate the received message.

For both notification processes, the application uses the notification manager service of the Android operating system and presents the message on the notification bar, see Figure 4.18 screenshots a, b and c. Moreover, since the application is running in the background all the time, there is no action needed by the users to activate these detections and functionalities.

When users open the notification, the application will prompt the users to rank the message out of five stars. Figure 4.18-c shows a message with the score of 4.0. After the ranking process, the application sends this score to the SWIS, where the nutritionist can see the average and the highest scored message.

Although some functionalities come from both test applications (UBIAPP and UBINUT), in GEONUT several methods have been improved regarding the geoposition and optimization of the application. These improvements include:

1. This application will automatically get the hot zones from the Open Street Map (OSM) community.

2. There is no need to add hot zones from a city manually through the SWIS.
3. The hot zones are stored in a local database in the device.
4. There is no need of internet connection for the geofencing detection and notification approach.

These modifications, and other important improvements, will be described in the technical section.

4.4.2 Summary

The GEONUT system has two major components: 1) the simple web information system , and 2) the mobile application developed for Android operating system version 4.2.3. The first component allows the nutritionist to write and send messages to all the users (mobile clients) with specific timing. The second component allows users to receive notifications based on a smart context in two ways: from the nutritionist or the detection of being near hot zones.

Ubiquitous computing in general, and smartphones in particular, could be useful to improve nutrition behaviors. Smartphones are considered useful to this goal because they are very popular, have an internet connection, and are constantly carried by users and used in multiple contexts by them. GEONUT incorporates the geofencing approach in order to take advantage of being carried by users in several contexts.

GEONUT has been developed considering both test applications (UBIAPP and UBINUT) with their modules, architecture, and functionalities.

In the next subsection, the technical aspects of the application will be presented with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Then, the context will be established of the testing of its results.

Finally, the results will be analyzed, with discussion related to the optimization and the functionalities and attributes of the system, as well as some conclusions and future work in this area.

4.4.3 Technical aspects

The following subsections will describe the IT artifacts that were involved in the software development process for GEONUT.

4.4.3.1 Actors

Considering that this work is developing incremental research with IT artifacts, the addition of new functionalities on a base prototype is technically straightforward to implement, given that it has modular architecture. GEONUT has been developed considering the previous test applications UBIAPP and UBINUT. Therefore, the informatics components from each one were reused.

In this case, the main actors that interact with the GEONUT system are the same as those defined for UBINUT. Thus, they are: mobile user, server user, database, information server, and mobile application. For further description, see the actor section of UBINUT in 4.3.3.1.

4.4.3.2 General use cases

In this subsection, the general use cases will be outlined that describe the main functionality of GEONUT. Therefore, to understand how the software behaves the interaction between components will be seen through the use case software artifact.

Previously, it was mentioned that GEONUT works in two main lines:

- Sending health messages to users supervised by a nutritionist
- Sending health messages to users when the application detects proximity to areas with a high probability of making poor nutritional decisions, known as hot zones

The first line has been fully developed and tested in the application UBINUT. More specifically, it is described by the general use case UN01 4.8. The second line is described by the use case GN01 General detection of a hot zone (defined above), which is presented in the Table 4.12.

Use case GN01:	General detection of a hot zone
Actor(s):	Mobile application, information server.
Purpose:	To detect a hot zone at an appropriate time and notify the users with a message.
Summary:	This use case begins when the notification service detects a POI near the users' current position. From a local database, a message is communicated to the users through the *notification manager*.
Preconditions:	1) The application has been downloaded and has stored messages for the users from the information server; 2) the users have configured the smartphone's GPS, allowing the application to read data from it; 3) the application has downloaded and stored the risk zones from the information server.

Actor's actions	**Application's answers**
	1.- This use case starts when the *alert service* gets new users' positions.
	2.- The *alert service* compares the users' positions with the stored POI in the local database (this does not require internet connection from the device to the web server).
	3.- If the *alert service* finds a match, it notifies users of a new message through the notifications bar
4. - Users select the notification.	
	5.- The system opens a window with mapView with the users' positions and the risk zones that are nearby.
6. - Users can select a POI and obtain information about it.	
Alternative flow: 2.- The system must have a local database with the stored POI. This process happens at the first initialization of the application through a configuration and downloading process.	

Table 4.12: GN01: General detection of a hot zone in a conversational format, which emphasizes the interaction between the actors and the system.

As described in the use case GN01, a module called *alert service* was used. This module corresponds to the module *alert service* from the UBIAPP application. The main responsibility of *alert service* was to implement a localization service with geofencing purposes. Details about the first implementation can be found in the UBIAPP section, specifically in the state chart diagram 4.9

In the next subsection, the major changes developed in the *alert service* will be explained. It is important to remark that, for the first version, the processing was done in the *information server* for optimization purposes. In this version, the processing was done locally on the mobile device. Therefore, it was not necessary to be connected to the internet.

Further analysis shows us that this was a good decision because:

- It saves battery for the mobile device since it is not making a request to a web service in a short period.
- It saves data traffic to the mobile device, thus optimizing the internet plan.

4.4.3.3 Sequence diagrams

In this section, the two main *sequence diagrams* will be discussed. They correspond to the two main general use cases of the system. The general use case of the system UBINUT is described in the Table 4.9 and the general detection of a hot zone is described in the Table 4.12. Additionally, some references to source code will be shown when it is required.

The sequence diagram (see Figure 4.19) for the use case described in the general use case of the system UBINUT can be found in Figure 4.13. This block is fully described in the UBINUT application section. The UBINUT module is reused because the main functionality has been complemented by the second use case of GEONUT: General detection of a hot zone.

The following sequence diagram represents the interaction between the main components as a black box:

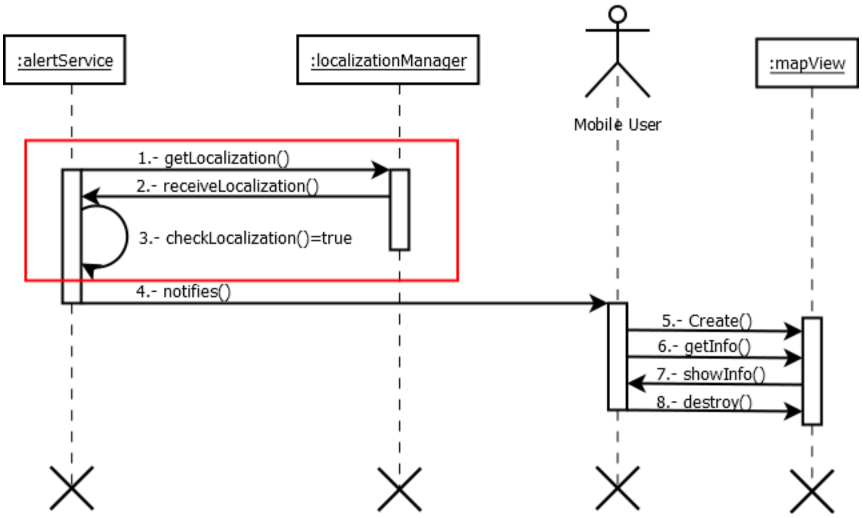


Figure 4.19: Sequence diagram: black box representation of the general use case GN01 *General detection of a hot zone*(The red rectangle represents a bucle on the process), see Table 4.12

As can be seen in the sequence diagram of Figure 4.19, the main actor that is involved in this interaction is the mobile user.

The red rectangle represents a bucle. This bucle drives the process of getting localization with the *localization manager* of the Android API and checking internally if the localization is near a risk zone. If the result of this comparison is true, then it notifies the *mobile users* through the notification bar. The *mobile users* will be able to get information about the risk zones that are nearby their position and also get a health message at that time, see Figure 4.19

Figure 4.18 represents this use case: General detection of a risk zone, from a graphical user interface (GUI) perspective on the mobile phone. The GUI represents what the mobile user sees, and the sequence diagram shows what the application does.

4.4.3.4 Statechart diagram

In the following subsection, a statechart diagram will be used to describe the different states for the update and alert service, which are responsible for getting new messages from the information server on the mobile device and detecting hot zones nearby users' positions. As was explained before, the statechart diagrams describe the flow of control from one state to another state. States are defined as a condition in which an object exists, and it changes when a certain event is triggered.

It is important to note that in the time that passed between the development of the test applications, UBIAPP and UBINUT, and the refined applications, GEONUT and UBESAFE, Android came out with new API versions to improve the Android OS performance. Therefore, whereas UBIAPP and UBINUT were developed with the version 2.x of the Android API, by the time GEONUT and UBESAFE were created, Android had introduced versions 4.x and 6.x. GEONUT uses API version 4.x and UBESAFE uses version 6.x. Each new version required that the background services work in a different way. In order to understand these changes in GEONUT, it is necessary to describe the lifecycles of an Android application and services, which correspond to all the applications that are developed in Android, including the test applications and the refined applications. After presenting these lifecycles from a general point of view, then the changes will be highlighted that were necessary to accommodate the new API version for GEONUT.

Lifecycle of an Android application

To understand the changes made for GEONUT, first it is necessary to present the lifecycle of an Android application. This corresponds to all Android applications, with any version of the API, including the test and the refined applications presented in this work:

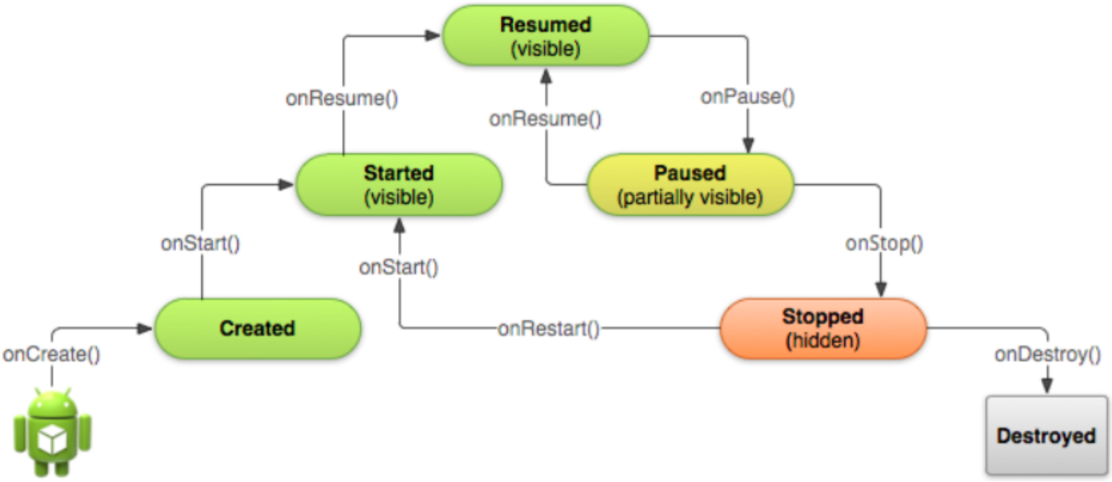


Figure 4.20: The activity lifecycle on an Android application

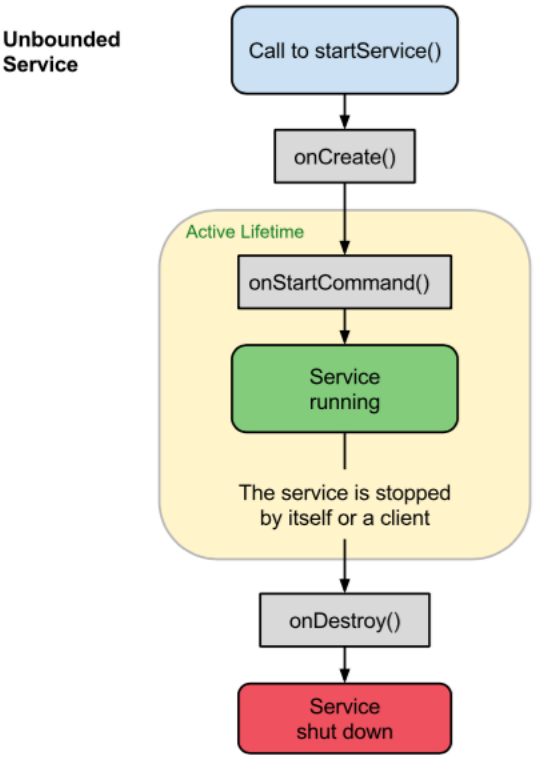


Figure 4.21: The service lifecycle of an Android application

Figure 4.20 shows the lifecycle of all Android applications. Here, it is possible to see how every state has its own role in an application. Normally, when an application is open, this will pass through the state *onCreate*. This is where the objects and their characteristics are defined. Then, the lifecycle will continue according to the application's needs. A mobile device is a concurrent device; this means that while one application is being used, the device can detect an incoming call and give the users that information at that precise time. When this happens, if an Android activity is running in the foreground, then that activity will be put in the states *paused* and *resumed*.

The lifecycle of a service in Android is a little different (See Figure 4.21). In GEONUT (which includes a new version of the *alert service* for the 4.x API that is different from the alert service used in UBIAPP with the 2.x API) uses unbounded services. This means that it is not attached to the application that created it, allowing the service to run indefinitely.

Statecharts

The GEONUT application has two main services. The first service, the update service, is described in the statechart section of the UBINUT applicaton. More specifically, details can be found in Figure 4.15. The second service, the alert service, is described in Figure 4.22, which shows a simplified version of its states:

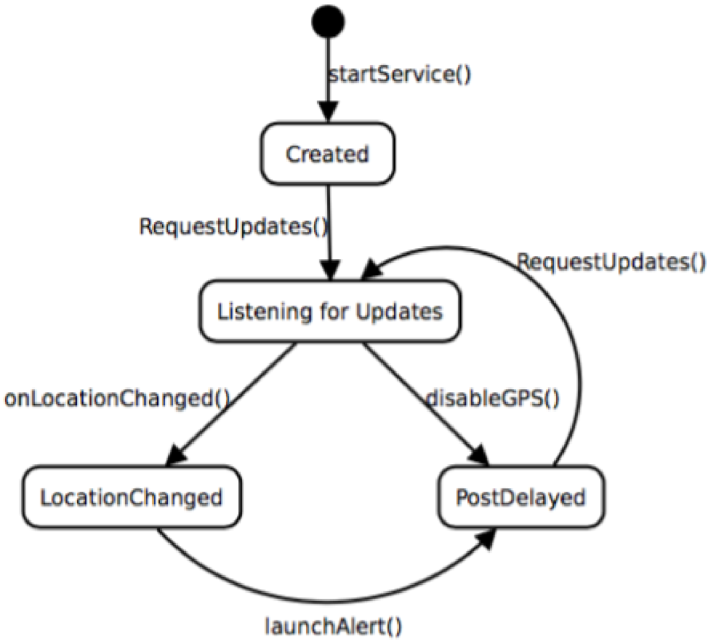


Figure 4.22: Statechart alert service: Diagram related to the general use case GN01*General detection of a hot zone*, see Table 4.12

There are four states that are described as follows:

- **Created**: this state has access to the local preferences and the database.
- **Listening for updates**: this state looks for a new user position.
- **onLocationChanged**: this transition responds to every change in the position of the device.
- **PostDelayed**: once the location change finds a match, then it will unbind the alert server and create a *post delayed task*, which will turn on the *alert service* again according to the users' preferences.

It is important to remark that the end of the statechart is not shown because the *alert service* runs indefinitely while the device is on.

The transitions from one state to another in the service have been developed thinking about ubiquitous implementation of the application. More specifically:

- **startService()**: an unbounded service is initialized with the objective to run a process constantly and independently of the main activity of the application.

- **RequestUpdates()**: when the listening for the location is initialized, a combination of GPS and mobile networking (wireless and cell phone towers) is used for obtaining the location of the users. There is also a preference that allows the use of only the GPS.

- **onLocationChanged()**: when a new location of the users is detected, the service will:

    1. if the new position is a position that is outside of the last area, then it will update the last position to the new position. Also, it will check if the current position has a match with the risk zones.
    2. if the new position is a position with low accuracy, that means that the mobile device is not getting data from the GPS. In this case, it is highly likely that the *mobile user* is in an indoor area. Therefore, the GPS is disabled for a while in order to save battery and also give the *mobile user* some time to move to an exterior zone.
    3. if the conditions of a match are satisfied, then the service will notify the users through an alert. Some of the conditions are: minimum accuracy of a location, distance from the users' positions to a risk zone and the last time a notification was sent.

- **disableGPS()**: in the case that it is not possible to get data from the GPS with good accuracy, this provider is disabled in order to work with the network provider.

- **launchAlert()**: sends the alerts on the mobile device, besides setting a reactivation timing for new alerts according to the users' preferences.

4.4.3.5 Architecture of the solution

GEONUT uses the same RESTful service principles as UBINUT and UBIAPP that were explained in the UBIAPP architecture of the solution section (see Section 4.2.3.5).

Figure 4.16 shows the conceptual architecture that integrates all components and services of the proposed solution (incorporating the alert service module). A summary will be presented of the **domain and data** layer because of their relevance to the proposed solution, according to the REST principles. This system has two components:

1. **The web component**: this is mainly a web server for managing the application data, and includes a web client.
    - **The service section in the web part**: the service section implements a RESTful service that processes requests and communicates with the business section. Note that this is the section that ultimately interacts with clients associated with the service. This architecture provides a solution with greater interoperability and modularity. Therefore, the architecture will not be affected by eventual transformations on the server or client side. The RESTful service receives a request from the Android client (*Json connection*).
    - **The business section of the web domain layer**: the business section corresponds to the core of the application, which executes the messages, hot zones and SQL queries that interact directly with the data layer.
    - **The data layer of the web component**: the data layer (or persistence data layer), is a database that stores messages, users and scores data.
2. **The Android client**: this is an application that runs on the Android OS.
    - **The service section of the client domain layer**: the service section implements two internal monitoring services that are responsible for reviewing and comparing certain mobile states related to the location and the applications that are running at that time. Simultaneously, it interacts with the business section. There are two kind of services: the update service and the alert service.

- **The business section of the client domain layer**: contains the core of the application that interacts with *JSonParser*, and is responsible for sending messages from the notification and surveillance services to the RESTful service of the web component. The *cl.geonut.JsonParser* package has the classes required to make different *requests* to the *RESTful* service and process data, thereby facilitating their subsequent representation by other components of the software.

The GEONUT system has been developed taking into consideration the findings from the UBINUT test with users. This application is a solution for the general use cases: *General use case of the system UBINUT* (see Section 4.3.3.2) and *General detection of a hot zone (geofencing)* (see Section 4.2.3.2).

The same architecture is used as in UBINUT, but add the functionality of launching messages automatically when the users get close to some kind of hot zone.

The hot zones come from two different sources:

1. Private source: the data layer on the server side, where private POIs are stored corresponding to places such as restaurants, cafeterias, and ice cream stores (usually a small set of POIs)
2. Public source: from a RESTFul service of OpenStreetMap, where all public POIs are downloaded related to food, food courts, pubs, cafeterias, and restaurants (usually a large set of POIs).

In summary, the main difference between these two sources is that the first is private whereas the second is public, taking information from OpenStreetMap.[7]

Working with geolocation has two main drawbacks: 1) GPS must be active, reducing battery duration; and 2) internet access must be available in order to download

---

[7] The users can set this source in the preferences section of the application. They can choose either private or public sources, but not both.

and update maps, and a huge amount of information must be downloaded. With respect to the first drawback, two strategies were applied to save battery life:

1. First, the users can configure how often the GPS sensor is polled, for example, every three, five or twelve hours. item Second, to maximize the efficiency of the GPS use, if when the GPS sensor is polled, no location can be established with high accuracy, then the GPS is put on standby for one hour (since it is assumed that the users are in an indoor area).

To solve the second drawback, GEONUT downloads the information related to the POIs nearby the users once the application is installed and the smartphone is connected to the internet. All this information is stored in a local database. Thus, future queries of the POIs do not need internet access. For both kinds of sources (public and private), the service downloads data from a defined perimeter around the users' positions. The POIs are received through a RESTFul service on the client side, and all data is saved in a local SQLlite Database.

GEONUT adds a service to the UBINUT system that also runs in the background. This new service obtains the geopositions of the users and compares them with the local database in order to detect if the users are inside a hot zone (a hot zone consists of a POI and a surrounding circle, with a defined radius).

If the users are within a hot zone, the users are notified through the *notification manager* with a previously stored message. The users will also have the chance to score the received message for future analysis.

In a general context, users will go about their normal routine and the GEONUT application will automatically detect (through the alert service) when they are at risk of falling into temptation of eating unhealthy food by being inside of a risk zone.

The timing of the notification is very important because users are in a moment in which they could make a poor nutrition decision. Moreover, users do not expect that the notification will arrive at that moment, since they are unaware of the application running in the background. Therefore, the information is presented ubiquitously based on that context. Figure. 4.18 shows a message on the mobile device, and a mapView with the POI or hot zones of the system.

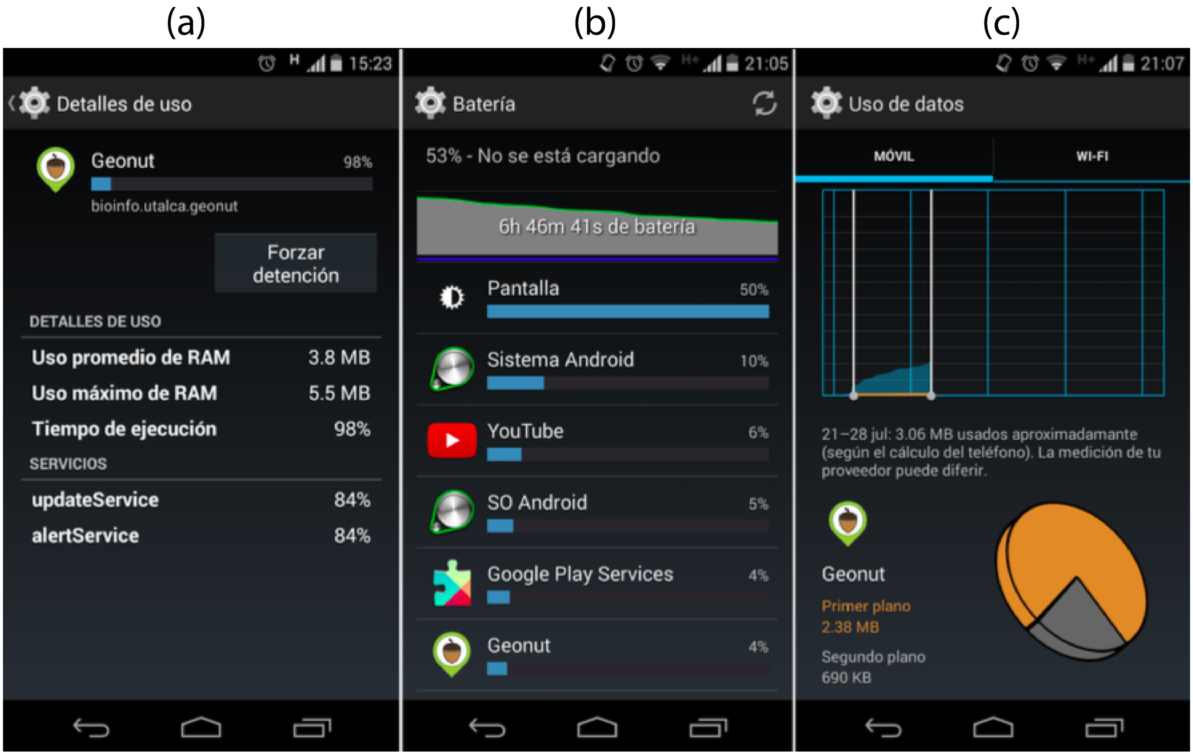
4.4.3.6 Optimization and key points

For this application, the most critical process is the HTTP request made by the update service to the RESTful service. This process is carried out according to certain timing preconfigured by the users. Since the data traffic for each request is minimum, less than 200 characters, and all the processes are carried out in the background, the use of the battery is limited and optimized. Resources that spend more battery on a mobile device are the screen, GPS, long processing times and an internet connection. Taking this into consideration, all the algorithms are optimized to provide the maximum efficiency in the use of these resources.

It is important to mention that all the processes in the background, such as services, tasks, and routines, among others, do not use the screen, which is the component that uses more battery.

During the development of GEONUT, a few conditions of the *alert service* were changed, thinking about the major changes from the Android operating system version 2.3.3 to the version 4.2.3. The methodology of the device localization was also modified, changing drastically the main algorithm that was responsible for localization of the *mobile user* based on timing, to an algorithm based on timing and states with post delayed actions.

At the same time, comparison of the *mobile user* localization with the risk zones is carried out locally, reducing the data traffic and the internet connection process of the mobile device.



Detalles de uso

Geonut 98%
bioinfo.utalca.geonut

Forzar detención

DETALLES DE USO

Uso promedio de RAM	3.8 MB
Uso máximo de RAM	5.5 MB
Tiempo de ejecución	98%

SERVICIOS

updateService	84%
alertService	84%

Batería

53% - No se está cargando

6h 46m 41s de batería

Pantalla	50%
Sistema Android	10%
YouTube	6%
SO Android	5%
Google Play Services	4%
Geonut	4%

Uso de datos

21–28 jul: 3.06 MB usados aproximadamante (según el cálculo del teléfono). La medición de tu proveedor puede diferir.

Geonut
Primer plano
2.38 MB
Segundo plano
690 KB

Figure 4.23: Details of the observed performance of the mobile application GEONUT

Finally, Figure 4.23 (screenshots a, b and c) shows the performance of the developed application on a Samsung S6 device with Android operating system version 4.2.3. As mentioned before, it is very important that the mobile application must have a good performance in order to work continuously and therefore be a ubiquitous system. In Figure 4.23-a, the amount of used RAM and the running time of the services (*update and alert services*) can be seen. Then, in Figure 4.23-b, the approximate use of the battery of the application is shown with the list of the other resources that use the battery constantly. Finally, in Figure 4.23-c, it is possible to see how much data traffic the application has had.

4.4.4 Testing and results

In this section, the results will be presented, showing how users interact with the software with the testing context. Afterwards, in the analysis of the results section, several conclusions will be drawn regarding how ubiquitous computing can be used as a preventive tool.

GEONUT has been developed as a result of the research following the design and creation approach, as well as considering the two initial test applications. This application focuses on dynamic health messages contextualized through a health professional, with automatic detection of hot zones where users are more susceptible to make poor nutritional decisions.

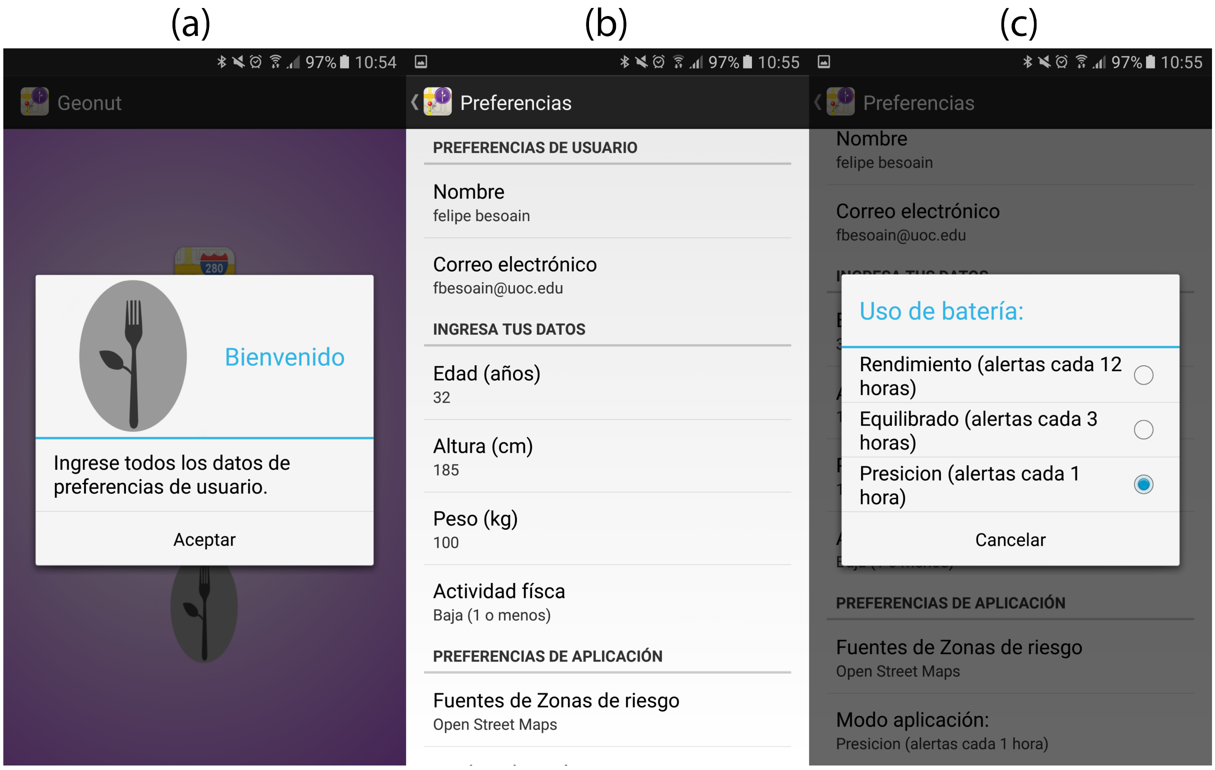
The test of the application was planned in Barcelona, since the first experience, with UBINUT, was in Chile. Nevertheless, the test was not be carried out because of two reasons.

1. Number of hot zones: in major cities like Barcelona, the number of hot zones georeferenced in OSM are massive. Therefore, the number of detections is too big within a short period of time: users will always be in a hot zone.
2. Work or house position: in some users, it was detected that the distance between hot zones and their home or work was within the geofence zone. Thus, users were constantly living or working in a hot zone, making detection meaningless. This can be explained by how, in a big metropolis, the population lives in apartment buildings with food businesses nearby. In contrast, in small towns, the population lives in houses at a distance from the food markets that are concentrated downtown, in a certain geographic area. In big cities like Barcelona, however, food vendors are distributed throughout various sectors.

Considering this situation, the early test of the application will be presented, which was done in Talca, Chile (a small town), followed by an international test in San Francisco, California. These early tests are developed from a technological point of view and known as functional tests. That is, these tests were designed to be a precursor to a full test, ensuring that the technology was adequate before undertaking a complete study. Therefore, once these initial functional tests revealed that a full test would not be feasible, it was decided that no test would be carried in Barcelona. Since no full tests were completed, these first functional tests are now presented, which are the only results available. In contrast to full tests, since the functional tests were created simply to test the functionality of the software, they are characterized by a small number of participants.

Figure 4.24 presents the first run of the application (screenshots a, b and c). This process is related to the data collection process of the application. All data obtained from the system was stored in the database as the foundation for further analysis. The data were collected through the internet and presented on the SWIS.

Data is collected at two different times: first, through the mobile device where users enter their personal information, and second, every time that users rank a health message.



(a)

Bienvenido

Ingrese todos los datos de preferencias de usuario.

Aceptar

(b)

PREFERENCIAS DE USUARIO

Nombre
felipe besoain

Correo electrónico
fbesoain@uoc.edu

INGRESA TUS DATOS

Edad (años)
32

Altura (cm)
185

Peso (kg)
100

Actividad físca
Baja (1 o menos)

PREFERENCIAS DE APLICACIÓN

Fuentes de Zonas de riesgo
Open Street Maps

(c)

Nombre
felipe besoain

Correo electrónico
fbesoain@uoc.edu

Uso de batería:

- Rendimiento (alertas cada 12 horas) ○
- Equilibrado (alertas cada 3 horas) ○
- Presicion (alertas cada 1 hora) ●

Cancelar

PREFERENCIAS DE APLICACIÓN

Fuentes de Zonas de riesgo
Open Street Maps

Modo aplicación:
Presicion (alertas cada 1 hora)

Figure 4.24: GEONUT - Android client: first run of the application, the users must complete their profile and set the applications preference

Figure 4.24-b shows how the users can fill in their profile information. This information is collected the first time that the users run the application. At that moment,

the users' contact information is requested, which is: name, email, age, height, weight, physical activity and source of risk zone. Additionally, the age, weight, height and how often they do sports is requested. With these data, it is possible to classify the users based on the general data, BMI and their behaviors.

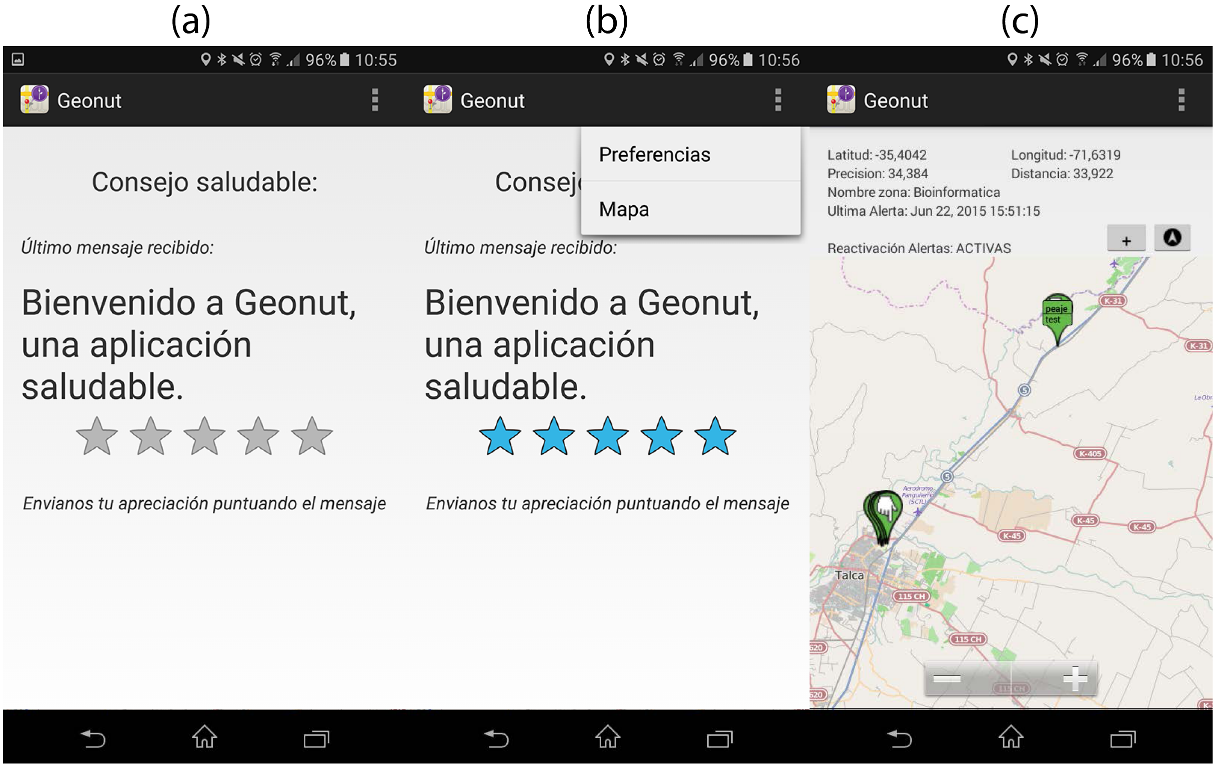
Figure 4.24-c shows how the users run the background service (alert service) in three different modes:

1. Precision: users will receive an alert every hour if a hot zone is detected. Then the service is down for that period of time.
2. Balance: users will receive an alert every three hours if a hot zone is detected. Then the service is down for that period of time.
3. Performance: users will receive an alert every twelve hours if a hot zone is detected. Then the service is down for that period of time.

Once the users have finished setting the application for the first time, the mobile application sends these data to the information server. The information server creates a users' registry in a local database and then sends a welcome message to the mobile devices (the screen on the left in Figure 4.24 shows this message).

Testing in a small town

A small geographical zone was established to test and try the GEONUT system. This test was done in the University of Talca, and five hot zones on campus were considered. Six students from the University tried the application for a week in order to see how the software behaved on their mobile devices. After the testing period, the students completed a survey. This survey asked for information about their experience, number of notifications, and problems detected, among other variables. Figure 4.25 shows the application working after the first installation (screenshots a, b and c). Figure 4.25-a shows a welcome message. Figure 4.25-b shows the welcome message rated with five stars and the access to the configuration menu. Lastly, Figure 4.25-c shows the hot zones nearby the user.



Consejo saludable:

Último mensaje recibido:

Bienvenido a Geonut, una aplicación saludable.

☆☆☆☆☆

Envianos tu apreciación puntuando el mensaje

Consej

Último mensaje recibido:

Bienvenido a Geonut, una aplicación saludable.

★★★★★

Envianos tu apreciación puntuando el mensaje

Latitud: -35,4042
Longitud: -71,6319
Precision: 34,384
Distancia: 33,922
Nombre zona: Bioinformatica
Ultima Alerta: Jun 22, 2015 15:51:15
Reactivación Alertas: ACTIVAS

Figure 4.25: GEONUT - Android client: scoring a message and checking the mapView on the application

During the seven days of testing, 39 ranked messages (an average of six messages per student) were received . The analysis shows that from the seven days of the test, the system received transactions from the users on only five days. This situation can be explained because the students do not go to the university during the weekend.

The prediction was that students would pass by a hot zone at least once a day, and they would not spend an extended period of time in any hot zone; the results were as expected because the number of hot zones on campus was five and the campus has 64 hectares. In a normal routine, students move on campus during classes through the hot zones (cafeteria and restaurant), and they go to the hot zones at certain times for lunch or dinner, including just for talking or studying.

In Table 4.4.4 resumes some important aspects from this experience:

Question	Feedback
Did you use GPS on your device during all the tests?	The response was positive and 4 students used GPS.
Did you detect abnormalities or problems on your device?	6 students did not find problems.
Did you use the map? Could you visualize the downloaded risk zones on the device?	3 students used the map and saw the hot zones on it. 3 students did not check the map.
Was the function of the application clear?	5 students considered the application useful and clear. 1 student did not understand why messages could help him.
Were you able to identify the application by its logo or name?	3 students liked the logo and 3 students did not like the logo.

Testing in a big city

The application was tested in San Francisco, USA. The test was done by a single user during a week. Figure 4.26 (screenshots a, b and c) shows the number of hot zones (points of interest) detected by OSM. As can be seen, the number of hot zones is greater than that in a small town.

Figure 4.26-b and Figure 4.26-c show the exact moment when the user approached a hot zone and got information about the nearest point.

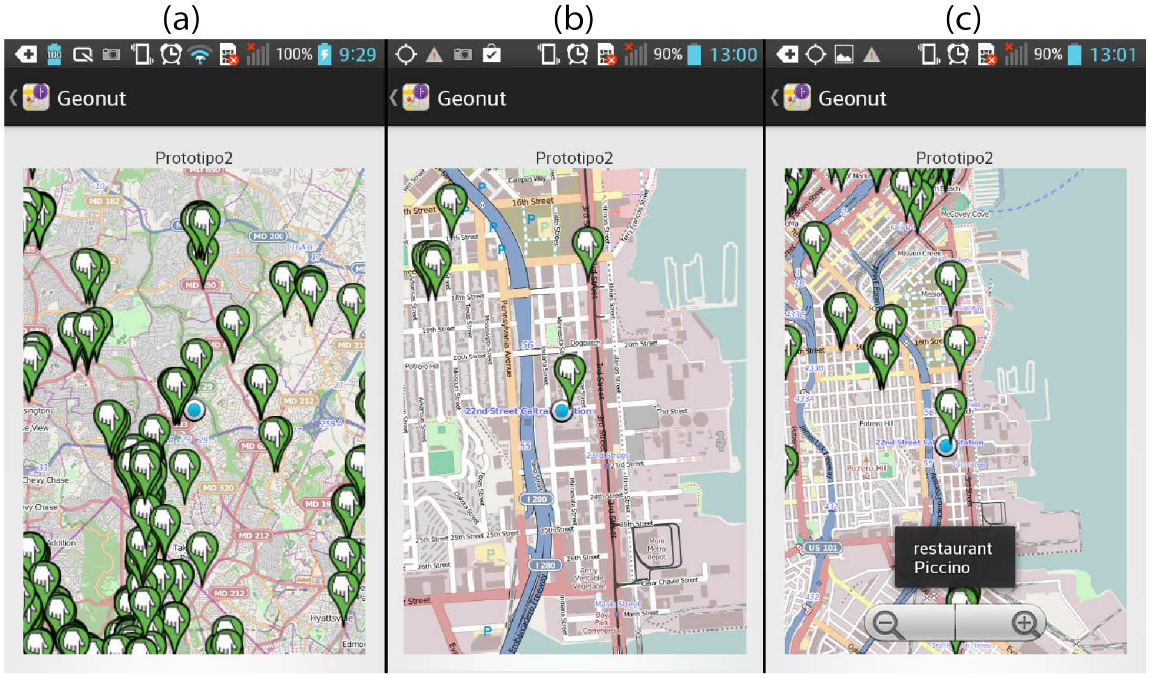


Figure 4.26: GEONUT - Android client: mapView with the POIs on San Francisco / USA

Figure 4.27 (screenshots a, b and c) shows the process of how users get the health message when in or nearby the hot zone, in this case, the restaurant Piccino. The users open the notification and evaluate the message according to the context.

It is important to point out that Figures 4.26 and 4.27 show the testing process in different conditions. For getting all the hot zones, the mobile devices must have internet access. Once this information is downloaded from the internet, an internet connection is not required for the detection and notification of messages because the data is located in an internal database. After being disconnected from the internet, once the device has internet connection again, then the application will send the scores to the information server.

Hola!
¡Tienes un consejo disponible!

Consejo de salud:

Último mensaje recibido:

Prefiere las mantequillas bajas en grasa en vez de las margarinas

★★★★☆

Envíanos tu apreciación puntuando el mensaje

Figure 4.27: GEONUT - Android client: receiving a health message at the exact time when the user is passing through a hot zone

In the GEONUT test in San Francisco, it is possible to see the difference of a small controlled experiences with GEONUT, like in Talca, versus the experience in a big city, like in San Francisco. In San Francisco, the configuration for receiving a message was every three hours. Nevertheless, the number of notifications per day was excessive.

Notifications were received at night in the hotel in San Francisco because of the surrounding food markets, which caused the principal objective of the application to fail (detecting hot zones in a smart context, on the go, walking or passing through).

In this context, it is important to note that the geographical place where users are is crucial for the success of the purpose of the application. Big cities have a larger number of restaurants and food markets, in some cases more than three per street. Meanwhile, small towns have the markets concentrated in a limited area. Therefore, GEONUT must be improved to consider the new use cases and provide a solution for the major cities where the detection of hot zones requires a refined algorithm and new functionalities.

4.4.5 Discussion of the results

GEONUT adds new modules and functionalities to the UBINUT Android application which has modular architecture. The principal focus of GEONUT is to extend the possible context where the application can prevent a risk behavior in a defined geographical location (geofencing). Another difference between GEONUT and UBINUT is that the sending of messages by GEONUT when the users are near POIs is automatic, instead of sending messages mediated by a nutritionist. It also solves the need for connectivity with a local database of messages and POIs. Therefore, there is no need to be connected to the internet in order to get a notification in a ubiquitous geographic location context. In summary, the main differences that GEONUT has in comparison to UBINUT are:

- Use of geofencing
- Automatic sending of health messages from a local database
- Internet connection is not necessary to send health messages to users when a risk zone is detected

After the technical testing process, it was concluded that the number of hot zones georeferenced in OSM is massive. Therefore, the number of detections is greater in a short period of time. It was also concluded that the distance between hot zones and the users' home or work is within the geofence zone. These issues are related to a big metropolis because the population is concentrated in apartment buildings with commerce and markets nearby. In contrast, in small towns, the population is concentrated in houses, and the food markets are spread downtown, which is related to a certain geographic area. In a big metropolis, this is distributed throughout various sectors.

New functionalities must be implemented in order to solve these problems.

- First, users need to add and delete hot zones according to their needs. This is necessary to avoid problems of detections when the users are at home or work. Also, users move through the city following a routine and known paths. Therefore, it is clear that they prefer having certain hot zones rather than having a database of all of them.

- Second, timing between detections should be configured by the users according to their experience and expectations.
- Third, the user experience must be improved in some sustainable way to prevent the application from being repetitive and artificial.

Finally, further work will address three directions: first, working in the mentioned functionalities; second, implementing a more accurate ontology algorithm in order to improve the automatic system that sends messages in GEONUT, making the selected messages closer to those that a nutritionist would choose; and third, measuring the impact of GEONUT as a ubiquitous system for promoting healthy habits through an evaluation methodology, such as the four group test.

4.5 UBESAFE

In this section, the refined application called UBESAFE will be described. The application UBESAFE was the fourth and last approach to addressing the issue of using mobile applications with preventive health messages. This application focuses on testing and trying preventive messages in a smart context with a group of MSM including a gamification approach for sharing data. This refined mobile application considers the previous experience with the last three mobile applications (UBIAPP, UBINUT, and GEONUT) from an informatics point of view and also the experiences of each test carried out with each respective target group.

The structure for the next section will be: firstly, UBESAFE will be presented and the general aspects will be described with its functionalities. Secondly, the technical aspects of the application will be presented with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Thirdly, the context of the testing will be established with its results.

Finally, the results will be discussed, with the functionalities and attributes of the system as well as some conclusions and future work in this area.

4.5.1 Introduction

UBESAFE follows the general diagram of the solution presented in Figure 4.1. In this stage of the research, a single application was developed for Android mobile devices with a version of the operating system 6.0.1 and a web information system to manage the data for and from the users.

The context taken into consideration was the results of the three previously tested mobile applications. From each test, the best-evaluated functionalities were taken. It can be seen from the results from the testing of UBINUT and GEONUT that users like their active participation, thanks to the rating of the messages, but it is necessary to increase their UX with more instances where they can configure the application or contribute to the main purpose of preventing risky behavior. It was seen that messages must be related to users' contexts considering their daily routine, considering location and other actions that they carried out with the mobile device.

Moreover, in order to make a sustainable system focused on prevention of health issues, community is an important concept that should be considered and addressed with a gamification approach in this system.

UBESAFE is an improvement of the last three mobile applications developed, UBIAPP, UBINUT and GEONUT. UBESAFE sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, the access to a specific URL on the internet, or their proximity to areas with a high probability of intercourse (*hot zones*). It also develops a community for the users, considering their ideas and knowledge of the hot zones. In order to provide a sustainable way of getting new data, the main experience was developed with *gamification* concepts. It is important to note that UBESAFE wants to cause awareness of each detected situation through health messages, taking into account privacy and users' preferences.

Therefore, the application works in four main lines:

- Sending health messages to users when the application detects a hot URL [8].
- Sending health messages to users when the application detects the use of a contact application (risk applications) [9]
- Sending health messages to users when the application detects the proximity to areas with a high probability of intercourse (hot zone).
- Allowing the users to make community sharing messages and POI (hot zones) through the system, enhancing the experience with a gamified scoreboard.

UBESAFE has two main components which are:

---

[8] Website where users can meet or chat with unknown people

[9] Risk application refers to any contact applications such as: Manhunt, Tinder, Badoo, and Brenda among others.

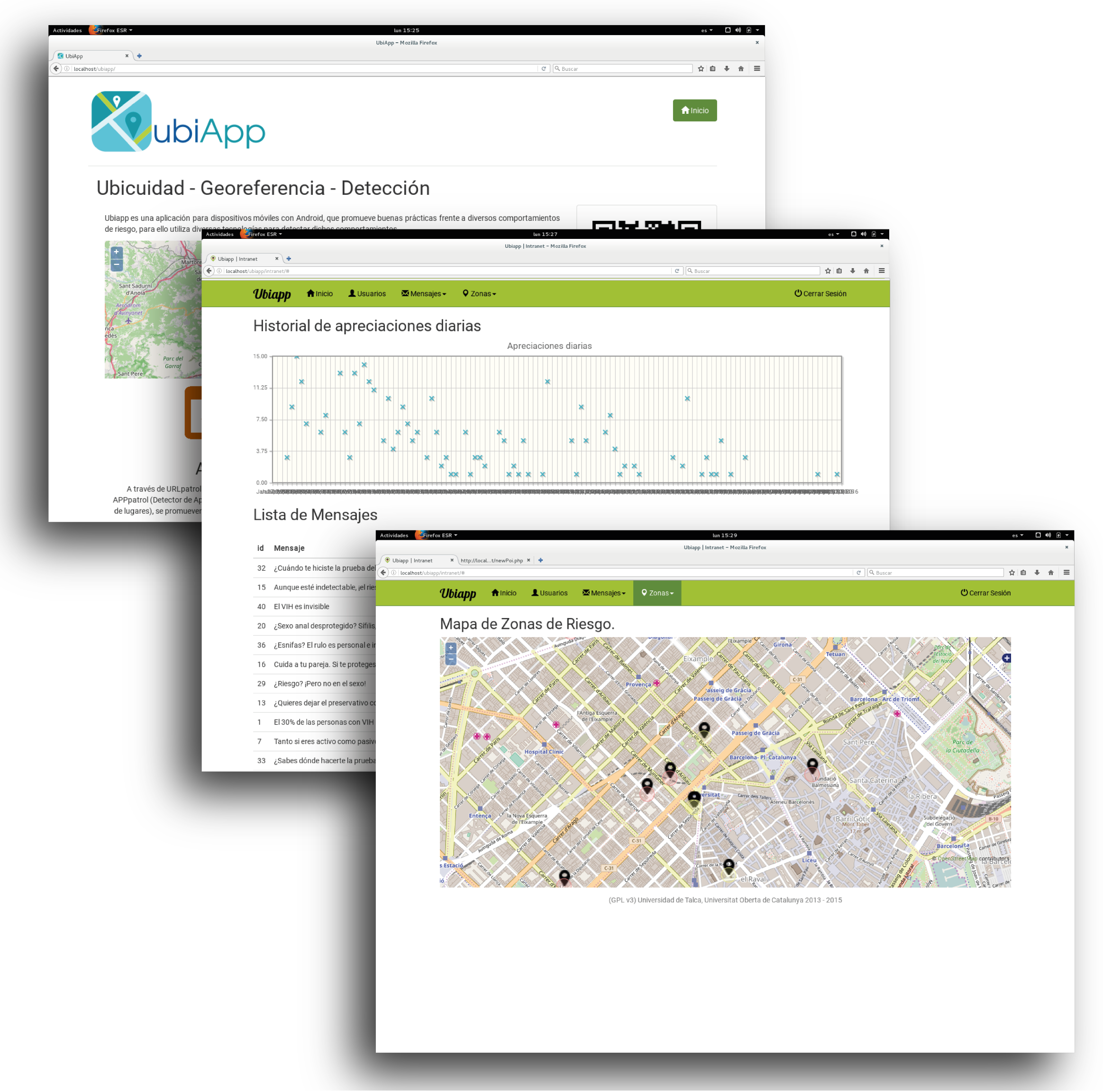
1. SWIS: through this SWIS, the health administrator is able to approve, modify or delete the message and POIs shared by the users. The system also shows statistics of users' scoring and most valued messages, number of users, etc. Figure 4.28 shows the web interface from the SWIS.

2. Mobile application: the mobile application has two main activities:

    - URLpatrol: this activity has all the functionalities of a web browser but with the preventive system incorporated.
    - UBESAFE: this activity will work detecting the different situations that can make the users aware of their actions through a health message. Thus, the message will be ranked by the users on a Likert scale from 1 to 5 using the Android ranking system.

Simple web information system (SWIS)

The SWIS is the interface that allows a health professional to review and check the messages and POI shared by the mobile users. This interface is important because it is part of a workflow that is controlled by a health professional. The workflow secures the information and validates the messages that will be sent to the mobile users. Figure 4.28 represents the SWIS with their modules and functionalities.



Ubicuidad - Georeferencia - Detección

Ubiapp es una aplicación para dispositivos móviles con Android, que promueve buenas prácticas frente a diversos comportamientos de riesgo, para ello utiliza div

A través de URLpatrol
APPpatrol (Detector de Ap
de lugares), se promuever

Historial de apreciaciones diarias

Lista de Mensajes

id	Mensaje
32	¿Cuándo te hiciste la prueba del
15	Aunque esté indetectable, ¡el ries
40	El VIH es invisible
20	¿Sexo anal desprotegido? Sífilis,
36	¿Esnifas? El rulo es personal e ir
16	Cuida a tu pareja. Si te proteges
29	¿Riesgo? ¡Pero no en el sexo!
13	¿Quieres dejar el preservativo cc
1	El 30% de las personas con VIH
7	Tanto si eres activo como pasiv
33	¿Sabes dónde hacerte la prueba

Mapa de Zonas de Riesgo.

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

Figure 4.28: UBESAFE - Web client: web interface where the health professional can access and manipulate all the data related to the UBESAFE system

Once the health administrator gets into the SWIS, the following modules will be presented:

- Users: information of the users (name, nationality, age, and UID [10]). For complementary information see Figure A.6 on the appendix.
- Messages: here the health administrator will be able to create, manage and delete messages from the system. This will also include the messages shared by the mobile devices. For complementary information see Figure A.8 on the appendix.
  - Create a message: the administrator can add a health message to the system. The message must be in 3 languages (English, Spanish, and Catalan) due to internationalization purposes. If the administrator does not include the message in one of the available languages, the message will not appear in that language. For complementary information see Figure A.9 on the appendix.
  - Manage: the administrator can update or delete a specific message. This option is used to modify mobile users' contributions or refine a proposed health message. For complementary information see Figure A.10 on the appendix.
  - Evaluation: in this section, the administrator can see the frequency of scored messages by the mobile users per day (graph). Moreover, a list order by the most value message of the system with the average scoring.For complementary information see Figure A.11 on the appendix.
  - Contribution: a list with the mobile users contribution is shown. The administrator can update the message, approving or deleting it. Once the message is approved, it is considered for the next update of the database. For complementary information see Figure A.12 on the appendix.

---

[10] Internal code that identifies the mobile devices where the UBESAFE app was installed.

- Zones: here the health administrator will be able to create, manage and delete hot zones from the system. This will also include the hot zones and POI shared by the mobile devices. For complementary information see Figure A.13 on the appendix.
  - Create: the administrator, through positioning a POI into the map interface, is able to add a hot zone to the database.
  - Mapview: here the hot zones are shown into a mapView. Thus, the administrator can have a global and geographic perspective of the data that is stored into the system, For complementary information see Figure A.14 on the appendix.
  - Manage: the administrator can see, approve or delete POIs in the system. The interface provides the latitude and longitude information, and also can show the point on a map, by using Google Maps interface.For complementary information see Figure A.15 on the appendix.
  - Contribution: a list with the mobile users contribution is shown. The administrator can see, approve or delete every single POI. For complementary information see Figure A.16 on the appendix.

Mobile application UBESAFE

The mobile application UBESAFE has two main activities with their own funcionalities:

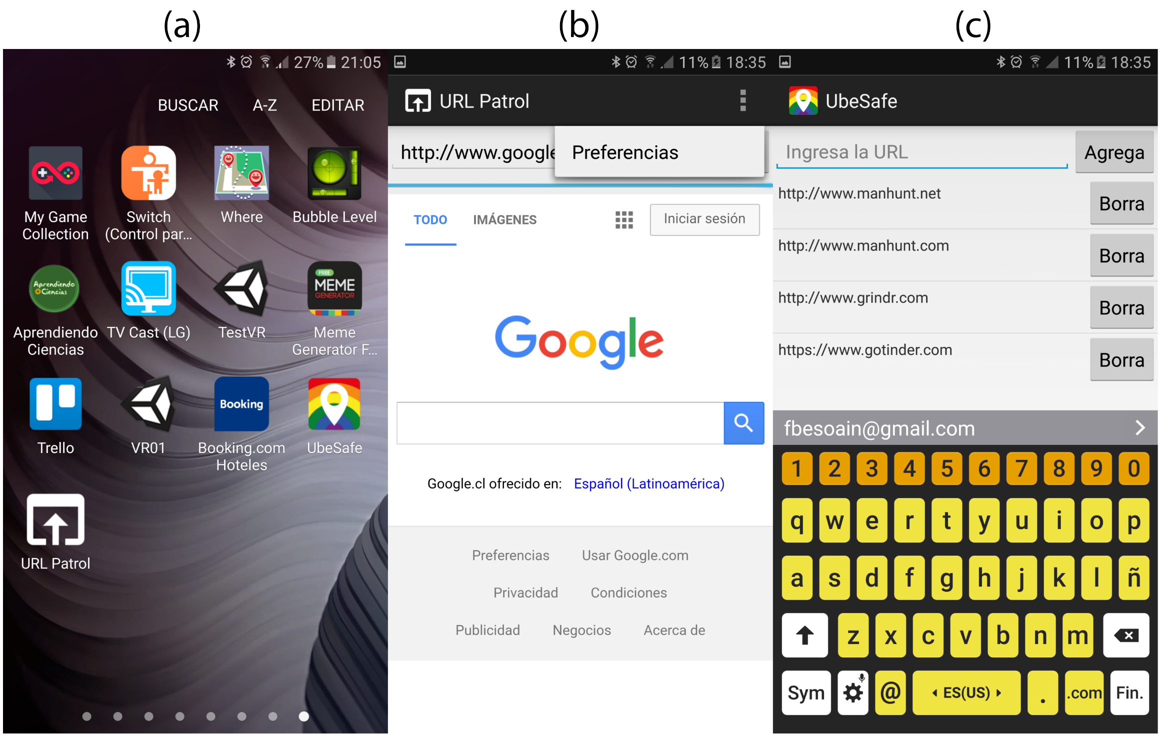
- First, URL patrol notifies the users when it detects a hot URL.
- Second, UBESAFE includes the detection of the proximity to areas (hot zones) or the use of any risk applications.

It is important to note that all health messages are retrieved from a local database. This database is controlled from the SWIS and updated every time that a new message or POI is detected and the administrator releases a new version of the database.

UBESAFE will be presented considering all its functionalities.

URL patrol UBESAFE

Figure 4.29 shows three states of the application at different times (screenshots a, b and c). First, once the application is opened by touching the URL patrol icon, users will be able to navigate on the internet (see Figure 4.29-a). Second, the users can see a web page, in this case, Google search engine. They can navigate on the internet just like any browser client (see Figure 4.29-b). Lastly, users can open the URL patrol preferences, where they can add or delete any website that is related to use contact applications or contact website. The application comes with a preloaded database of websites such us: Grindr, Manhunt, Gotinder, etc (see Figure 4.29-c).



http://www.google

Google.cl ofrecido en: Español (Latinoamérica)

http://www.manhunt.net

http://www.manhunt.com

http://www.grindr.com

https://www.gotinder.com

fbesoain@gmail.com

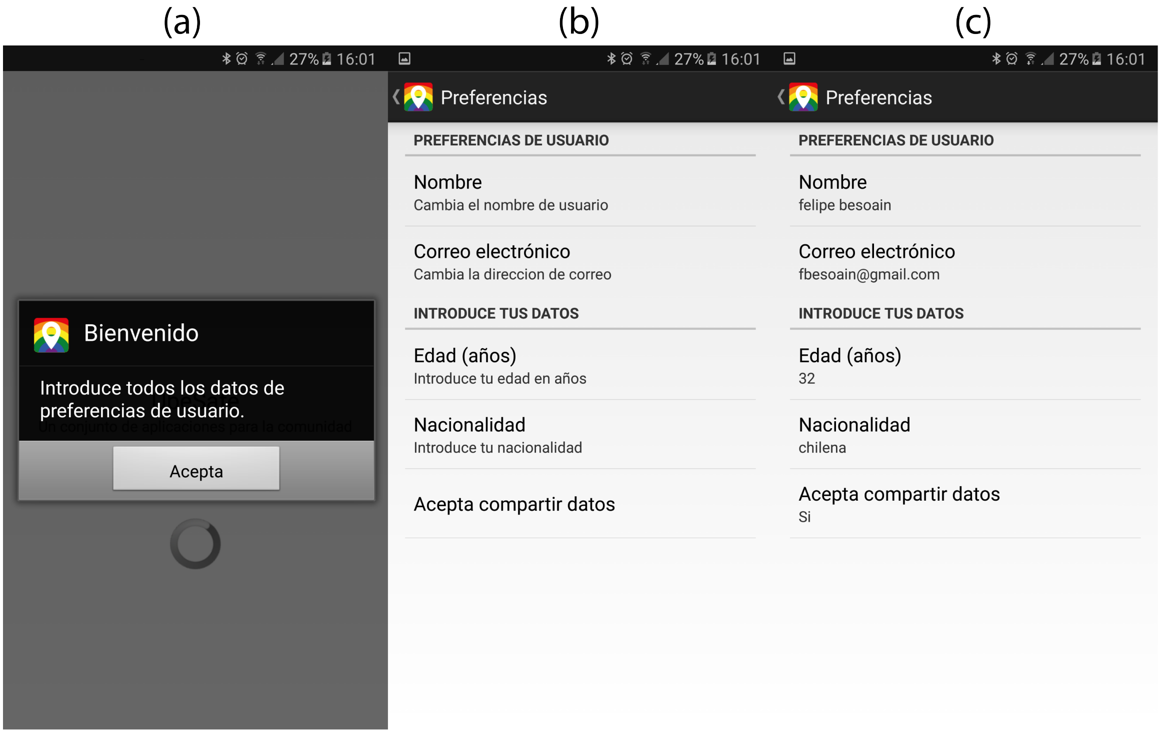
Figure 4.29: UBESAFE - Android client: running the URL patrol activity from the UBESAFE application

When the users navigate on a website that is on the list, the system will detect that action notifying the mobile users with a health message. It is important to mention that URL patrol is not another mobile application, but part of UBESAFE and can be run independently for UX purposes. Mobile users can configure URL patrol as their default web client and use it to navigate on the internet as part of the detection and prevention system.

First run UBESAFE

The first time that UBESAFE is run on the mobile device, the mobile users will have to fill in their data and configure the application. The application allows the users to participate in the community sharing data for research purposes, or they can choose to be anonymous.

Figure 4.30 shows three states of the application when introducing users' data at different times (screenshots a, b and c). First, once the application is opened by touching the UBESAFE icon, the application will check the user data (see Figure 4.30-a). Second, a preference list will be shown to fill with their information (see Figure 4.30-b). Lastly, once the users have entered all the information, the application will process in background to sign in the users to the SWIS database and download the messages and POI available from the SWIS, in order to query them locally (see Figure 4.30-c).



(a)

Bienvenido

Introduce todos los datos de preferencias de usuario.

UbeSafe

Un conjunto de aplicaciones para la comunidad

(b)

Preferencias

PREFERENCIAS DE USUARIO

Nombre
Cambia el nombre de usuario

Correo electrónico
Cambia la direccion de correo

INTRODUCE TUS DATOS

Edad (años)
Introduce tu edad en años

Nacionalidad
Introduce tu nacionalidad

Acepta compartir datos

(c)

Preferencias

PREFERENCIAS DE USUARIO

Nombre
felipe besoain

Correo electrónico
fbesoain@gmail.com

INTRODUCE TUS DATOS

Edad (años)
32

Nacionalidad
chilena

Acepta compartir datos
Si

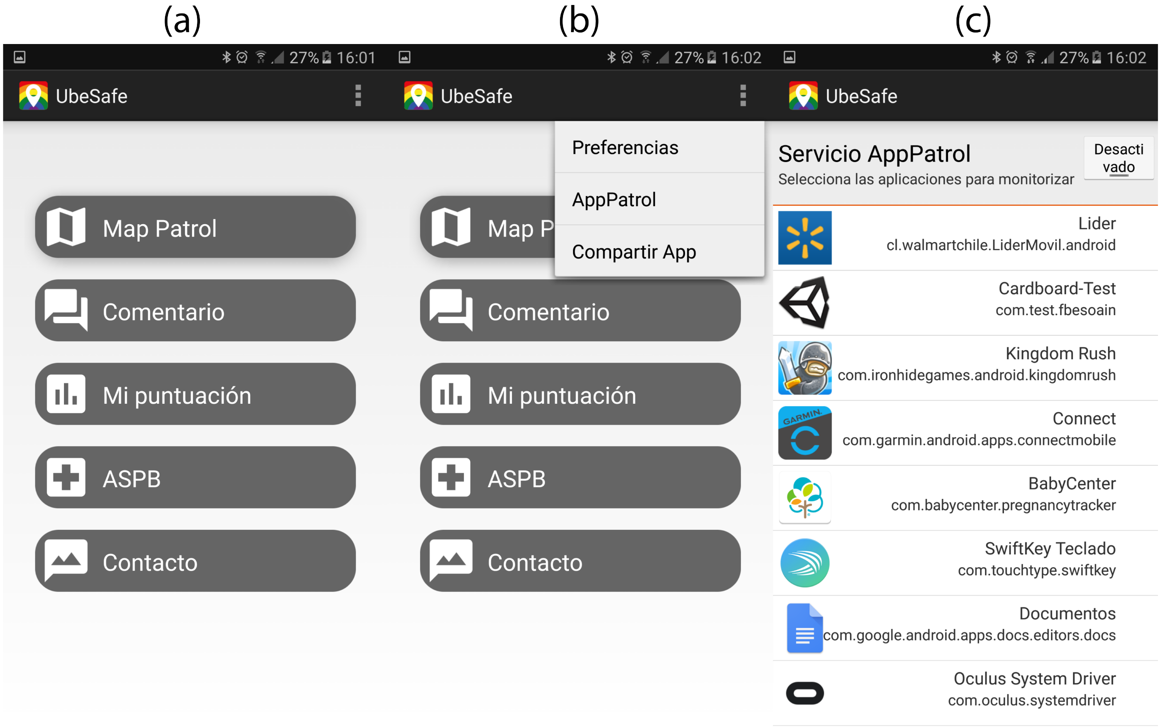
Figure 4.30: UBESAFE - Android client: first run of the application UBESAFE, the users must set their profile information

Once the mobile users have completed the information requested by the application, the health administrator will be able to see the users' data in the users section of the SWIS system.

Detecting risk applications

For detecting risk applications, the users need to configure the services known as AppPatrol. This service will show the mobile users all the applications installed on their devices. The mobile users will select the apps for monitoring and then activate the service. This service, like all the services of UBSAFE, works in background, and the mobile users do not need to start it again. It will continuously be monitoring the devices until the service is deactivated.

Figure 4.31 shows three states of the application at different times (screenshots a, b and c). First, once the application is opened by touching the UBESAFE icon (see Figure 4.31-a). Second, the users select in the preference section the AppPatrol settings (see Figure 4.31-b). Lastly, a list with all the icons and name of the installed applications will be shown (this service is off by default ,see Figure 4.31-c).



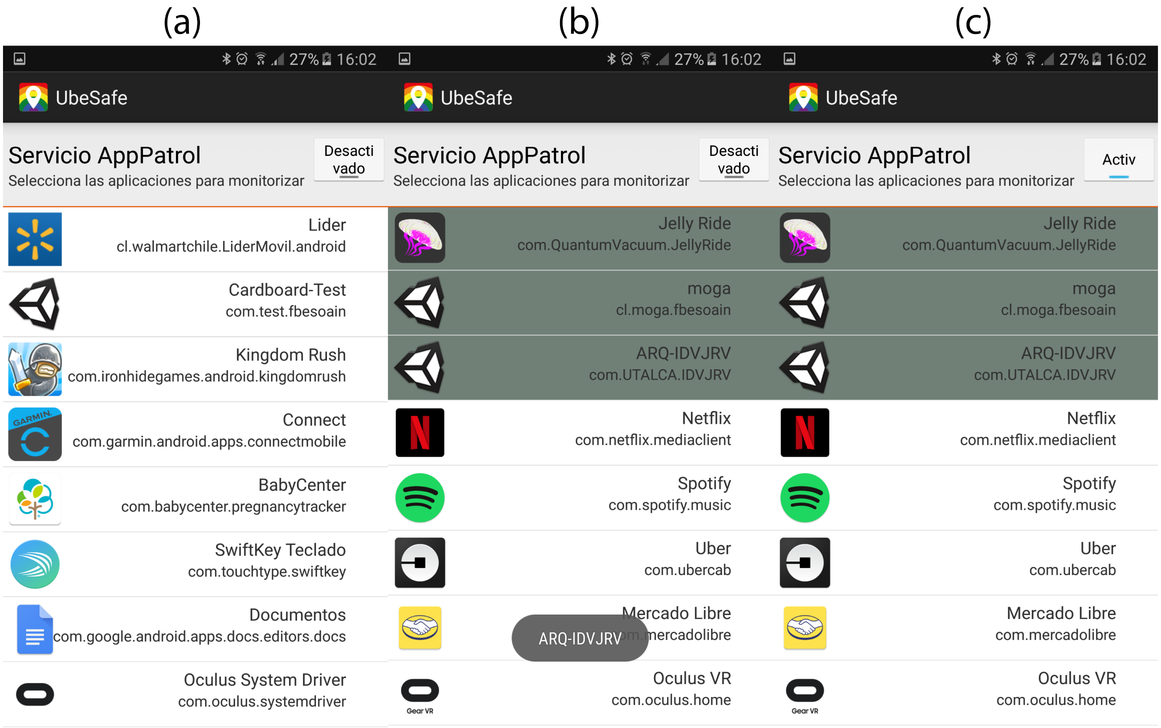
Servicio AppPatrol
Selecciona las aplicaciones para monitorizar

Lider
cl.walmartchile.LiderMovil.android
Cardboard-Test
com.test.fbesoain
Kingdom Rush
com.ironhidegames.android.kingdomrush
Connect
com.garmin.android.apps.connectmobile
BabyCenter
com.babycenter.pregnancytracker
SwiftKey Teclado
com.touchtype.swiftkey
Documentos
com.google.android.apps.docs.editors.docs
Oculus System Driver
com.oculus.systemdriver

Figure 4.31: UBESAFE - Android client: setting the AppPatrol service on the application

Once the users activate the service, they can choose the apps to be warned about. Figure 4.32 shows the process of selecting them.

Figure 4.32 shows three states of UBESAFE at different times (screenshots a, b and c). When configuring the AppPatrol service. First, the users are in AppPatrol settings (see Figure 4.32-a). Second, the users select applications from the list to monitor. This action is carried out by doing a long press on the list (according to the mobile standards a long action present selection on a list, see Figure 4.32-b). Lastly, the users start the service. This service will work always (algorithms to optimize this service are shown in the future sections, see Figure 4.32-c).



Servicio AppPatrol
Selecciona las aplicaciones para monitorizar

Lider
cl.walmartchile.LiderMovil.android
Cardboard-Test
com.test.fbesoain
Kingdom Rush
com.ironhidegames.android.kingdomrush
Connect
com.garmin.android.apps.connectmobile
BabyCenter
com.babycenter.pregnancytracker
SwiftKey Teclado
com.touchtype.swiftkey
Documentos
com.google.android.apps.docs.editors.docs
Oculus System Driver
com.oculus.systemdriver

Servicio AppPatrol
Selecciona las aplicaciones para monitorizar

Jelly Ride
com.QuantumVacuum.JellyRide
moga
cl.moga.fbesoain
ARQ-IDVJRV
com.UTALCA.IDVJRV
Netflix
com.netflix.mediaclient
Spotify
com.spotify.music
Uber
com.ubercab
Mercado Libre
com.mercadolibre
Oculus VR
com.oculus.home

Figure 4.32: UBESAFE - Android client: risk application has been selected and the service is activated

Detecting hot zones

This module is responsible for sending health messages to users when the application detects the proximity to areas with a high probability of intercourse (hot zones). In addition, the UX has been enhanced with several functionalities:

- To add private POI to the database of the mobile devices.
- To share the POI with the community sending the information to the Information server.
- To delete any POI from the database, allowing the mobile users to choose which hot zones to detect.

Figure 4.33 shows three states of the application at different times (screenshots a, b and c). When configuring the hot zone. First, once the application is opened by touching the UBESAFE icon (see Figure 4.33-a). Second, the users have selected the

Map Patrol option. The application opens a mapView with users' current positions and POI or hot zones nearby (see Figure 4.33-b). Lastly, in the configuration section, the mobile users can add POI and manage a single POI (share with the community or delete, see Figure 4.33-c).

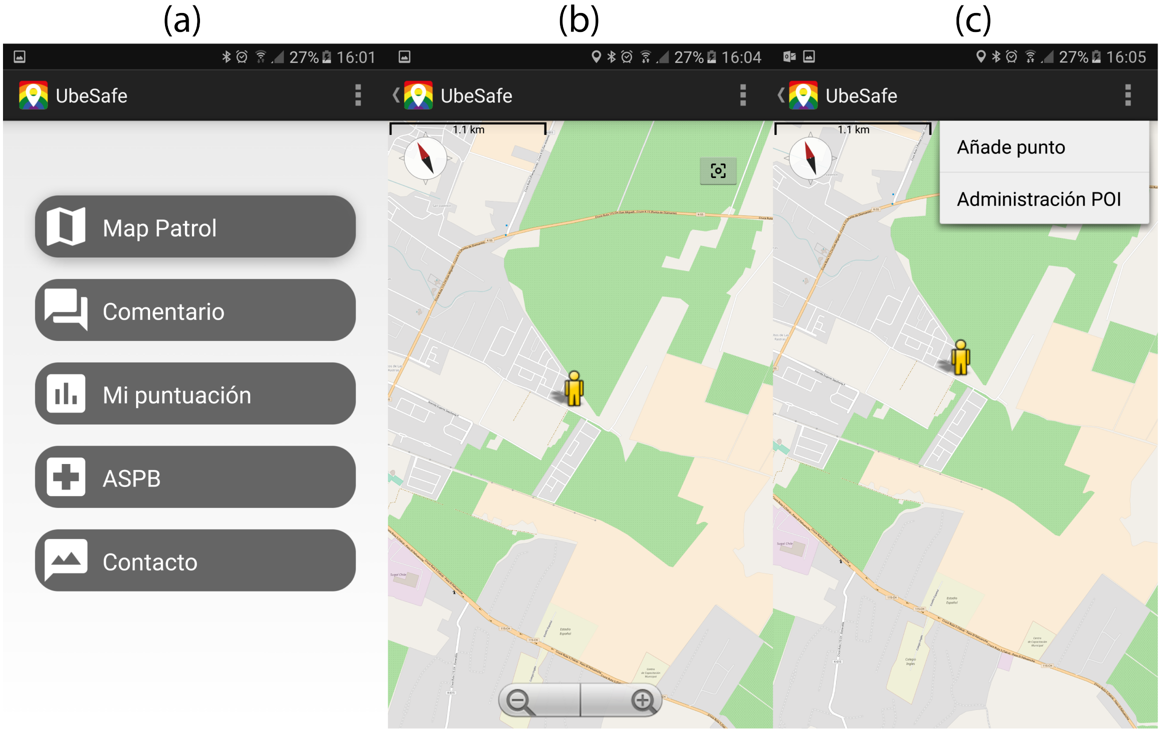
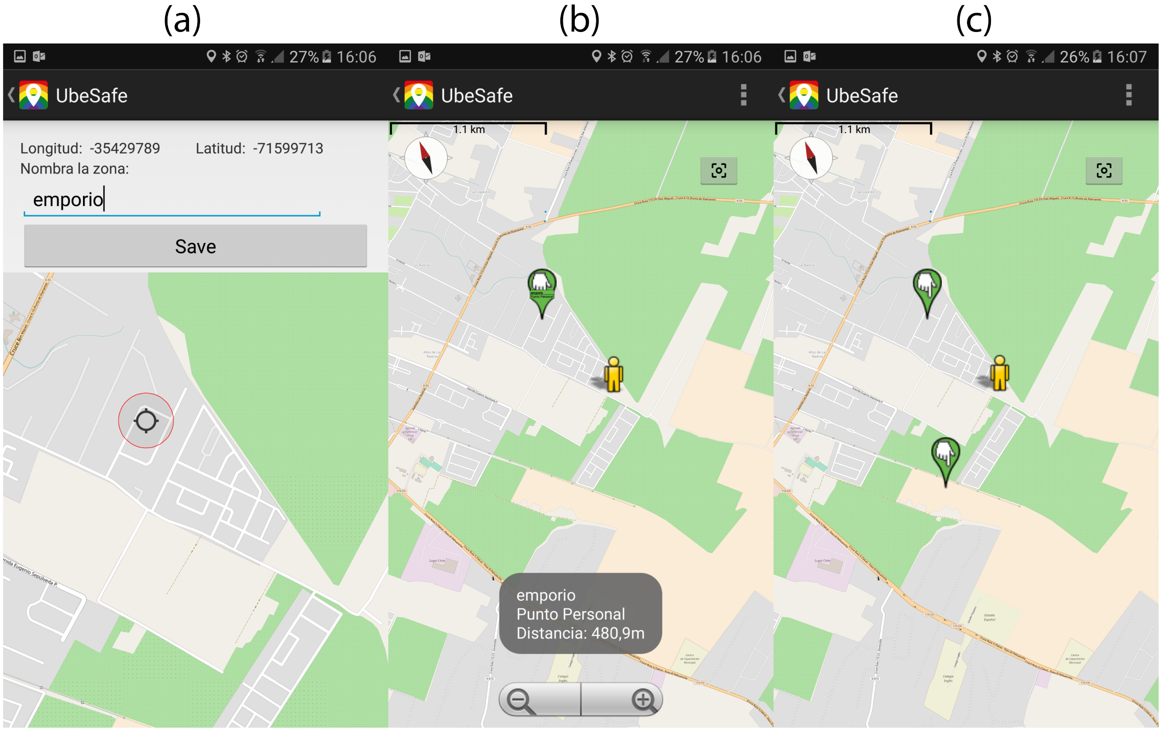


Figure 4.33: UBESAFE - Android client: mapView with the current positions and POI or hot zones nearby

UBESAFE allows mobile users to add their own POI. This is an important feature because it gives control of the application and detection to the end users. The main purpose is not to force any action, more than that, it is to promote changes and awareness. Figure 4.34 shows three states of the application at different times (screenshots a, b and c). First, the users have selected adding POI in the map patrol settings. In the mapView interface, the users select from a mapView a point by doing a long press on the map. The application automatically will get the latitude and longitude; then, the users must write a name of the POI (see Figure 4.34-a). Second, the application shows the mapView with the recently added POI. If the users touch the POI, then UBESAFE shows information and distance from the users' current position to the POI (see Figure 4.34-b). Lastly, on the right, two POI can be seen nearby the users' current position (see Figure 4.34-c).



Longitud: -35429789 Latitud: -71599713
Nombra la zona:
emporio
Save

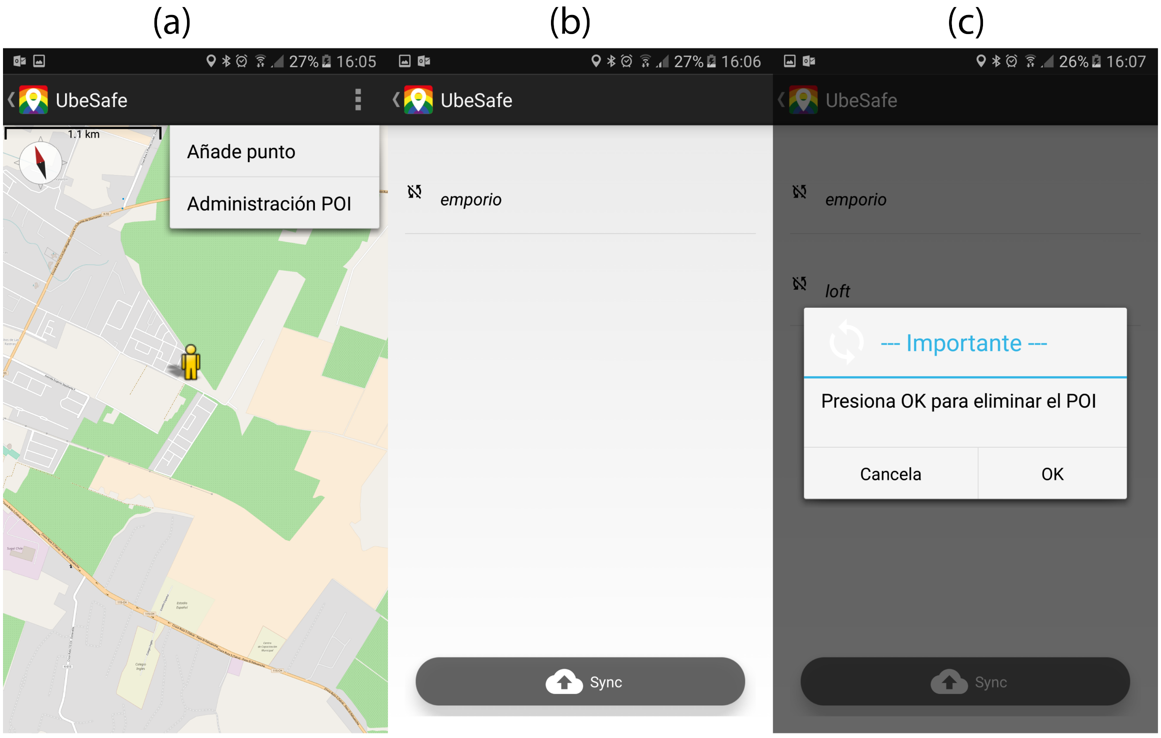
emporio
Punto Personal
Distancia: 480,9m

Figure 4.34: UBESAFE - Android client: adding users favorite hot zones for future monitoring of the alert service

Besides adding their own POI to the application, mobile users also have the option of sharing the POI with the community by sending the information to the information server, and deleting any POI from the database, allowing the mobile users to choose which hot zones to detect. Figure 4.35 shows three states of the application at different times (screenshots a, b and c). First, the users have selected to manage POI in the map patrol settings (see Figure 4.35-a). Second, the application shows a list with the POIs [11] (see Figure 4.35-b). Lastly, by doing a long press on the target POI the mobile users can delete the POI from the local database. The users also, by doing a simple press, can select the POI to share it with the community (see Figure 4.35-c).

---

[11] It is important to note that each POI has an icon to the left that shows the current status of the POI: shared or local



Presiona OK para eliminar el POI

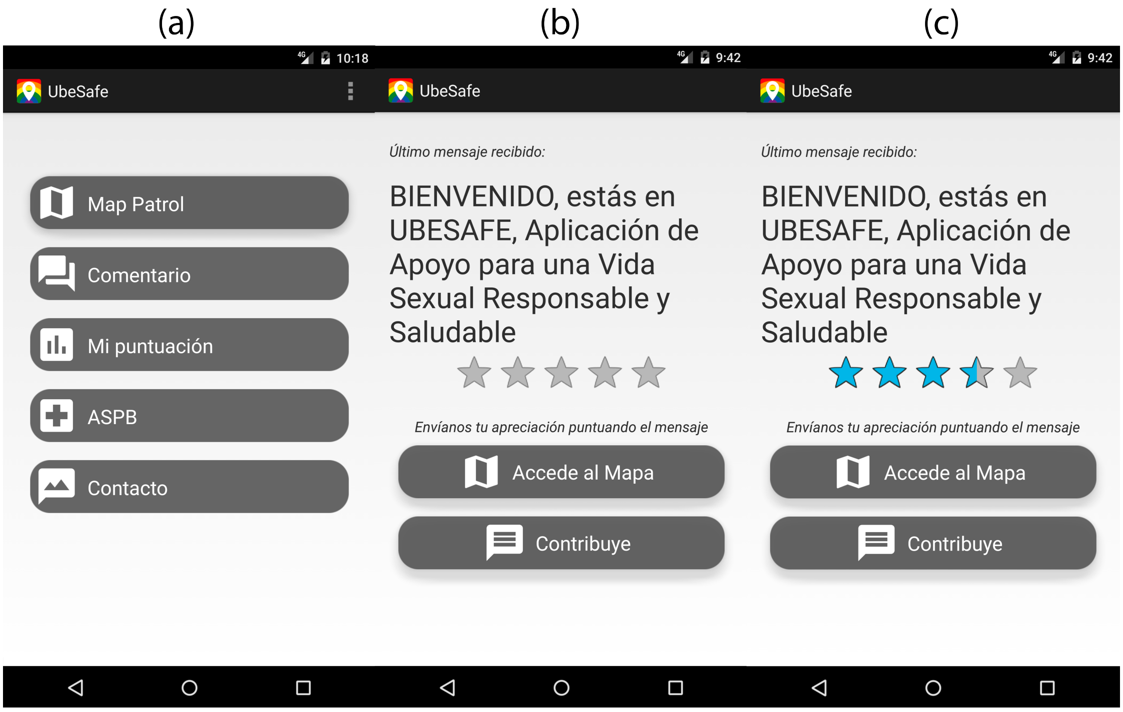
Figure 4.35: UBESAFE - Android client: deleting a hot zone from the list. Therefore, this zone its excluded for future monitoring of the alert service.

Managing health messages

To improve the functionalities and UX related to the health messages, UBESAFE allows the mobile users to score the health message for the notification process. The application uses the notification manager service of the Android operating system and presents the message on the notification bar. Moreover, since the application is running in the background all the time, there is no action needed by the users to activate this detection.

This interface includes two shortcuts to the map patrol, so the mobile users can see their position and all the POIs nearby, and also allowing the mobile users to add, delete and upload their own health messages. The mobile users can access this interface from the main menu in comments, see Figure 4.36-a or at any time that they get a new notification.

When the users open the notification, the application will prompt the users to rank the received message out of five stars. Figure 4.36-b and Figure 4.36-c shows a message with the score of 5.0. After the ranking process, the application sends this score to the SWIS where the health administrator can see the average and the highest scored message.



Último mensaje recibido:

BIENVENIDO, estás en UBESAFE, Aplicación de Apoyo para una Vida Sexual Responsable y Saludable

Envíanos tu apreciación puntuando el mensaje

Último mensaje recibido:

BIENVENIDO, estás en UBESAFE, Aplicación de Apoyo para una Vida Sexual Responsable y Saludable

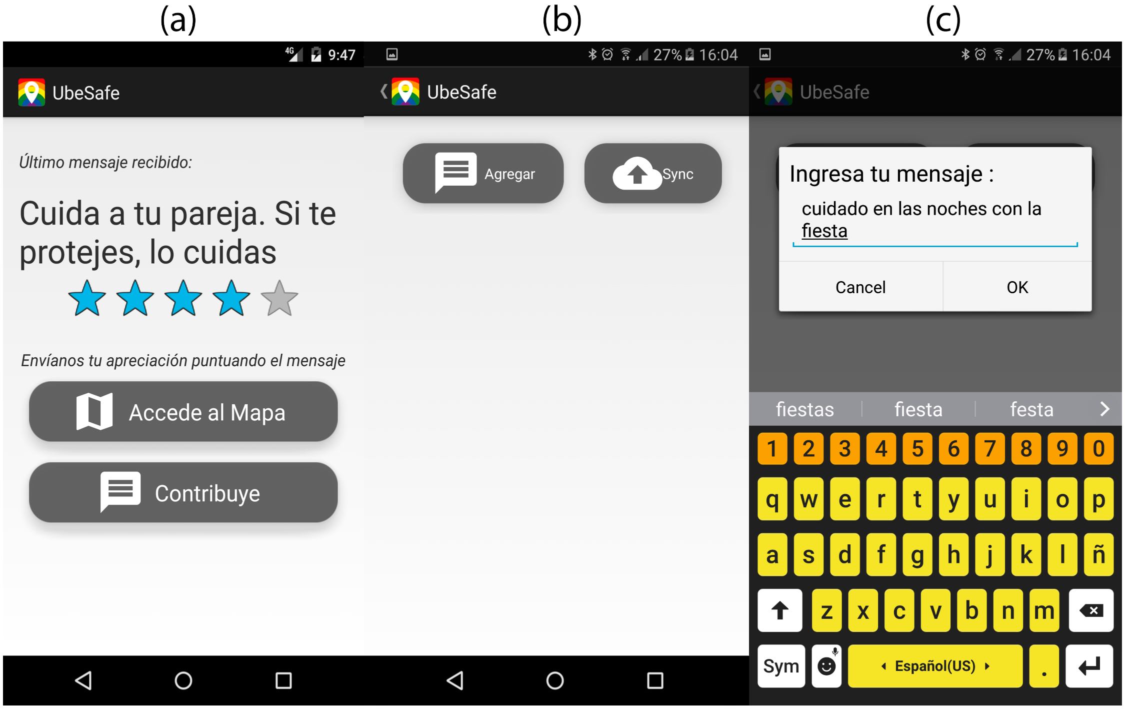
Envíanos tu apreciación puntuando el mensaje

Figure 4.36: UBESAFE - Android client: primary interface for scoring messages with shortcuts for contribution and mapView.

Mobile users can add their own messages, because they can have private messages for their consideration. Figure 4.37 shows three states of the application at different times (screenshots a, b and c). First, interface that is open everytime that mobile users gets a new notification (independent from the source, see Figure 4.37-a). Second, the application shows a list with the messages [12], here the mobile users can add or upload a message to the information server (see Figure 4.37-b). Lastly, the users can add their own private message that will be included into the local database (see Figure 4.37-c).

---

[12] It is important to note that each message has an icon to the left that shows the current status of the message: shared or local



Último mensaje recibido:

Cuida a tu pareja. Si te protejes, lo cuidas

Envíanos tu apreciación puntuando el mensaje

Ingresa tu mensaje :

cuidado en las noches con la fiesta

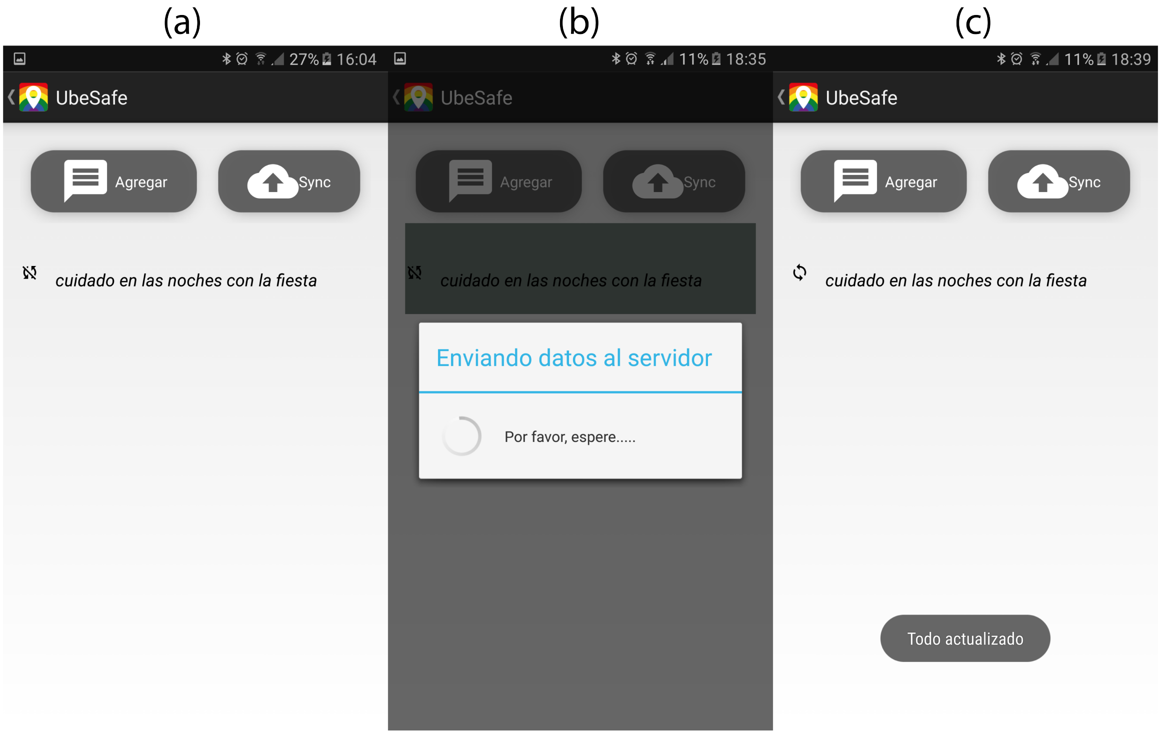
Figure 4.37: UBESAFE - Android client: adding a personal message to the list of health messages

Lastly, mobile users can upload and share their private messages with the community. When the users want to share a message, it will be uploaded to the information server where the health administrator through the SWIS can review it. This revision could modify the original message if it is needed. Then the message will be added to the main database of the system.

Figure 4.38 shows three states of the application at different times (screenshots a, b and c). First, the application shows a list with the messages, [13] here the mobile users can add or upload a message to the information server (see Figure 4.38-a). Second, the users, by doing a simple press, select the message and then select the sync symbol (see Figure 4.38-b). Lastly, the message has changed the icon from the left, showing that the private message is updated to the main servers (information server, see Figure 4.38-c).

---

[13] It is important to note that each message has an icon to the left that shows the current status of the message: shared or local



cuidado en las noches con la fiesta

Enviando datos al servidor

Por favor, espere.....

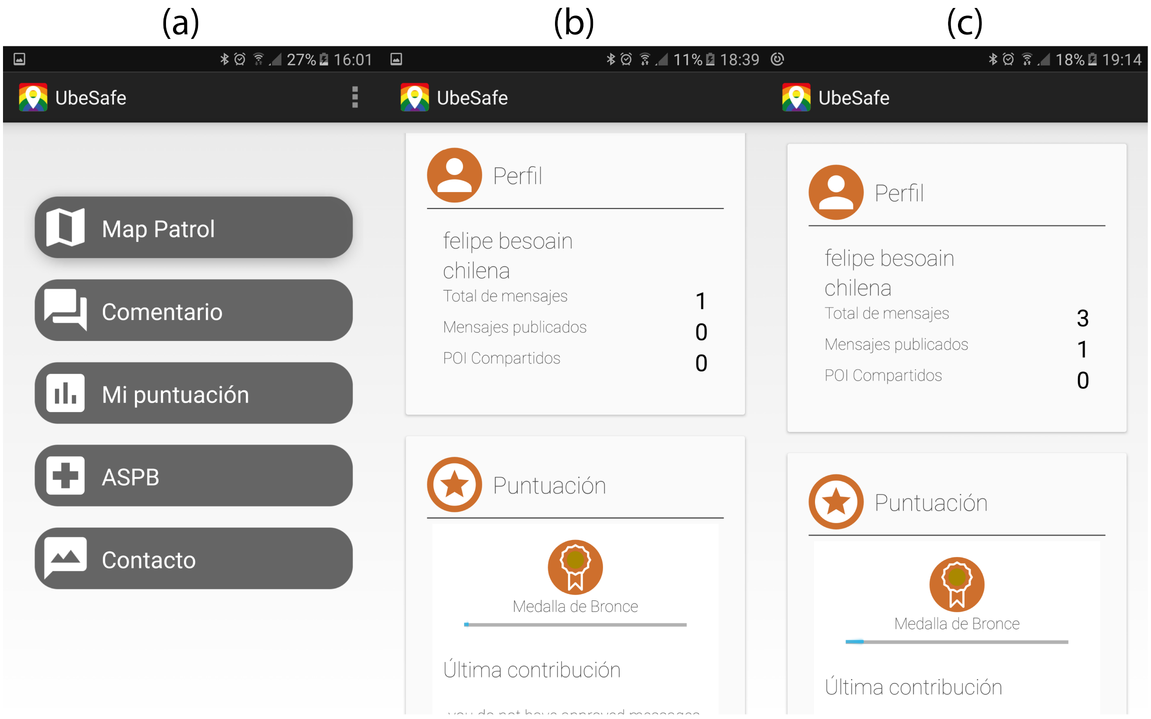
Todo actualizado

Figure 4.38: UBESAFE - Android client: sharing a selected message with the community of UBESAFE

Gamification scoreboard

UBESAFE has a scoreboard with the most valued health messages and users' ranking. Every time that mobile users share a POI or health message, they get an amount of experience (points) in exchange. The amount of experience awarded is related to the number of actions that the users have carried out in the system. The more messages or POIS that they share with the community, the more experience they get. Depending on the amount of experience, mobile users will get a medal that reflects their rank in the system. The contribution will also be presented on a scoreboard, enhancing the experience with the system and promoting the sharing and contributing of POI and messages to the UBESAFE system.

Figure 4.39 shows three states of the application at different times (screenshots a, b and c). First, the application shows the main menu, where the users select the punctuation (see Figure 4.39-a). Second, the users can see the number of shared contributions (messages and POI). They can also see a bar of experience and the current medal (see Figure 4.39-b). Lastly, the image from the right shows how the users have increased their experience in the bar. This happens because the mobile users have shared more messages and POI (see Figure 4.39-c).

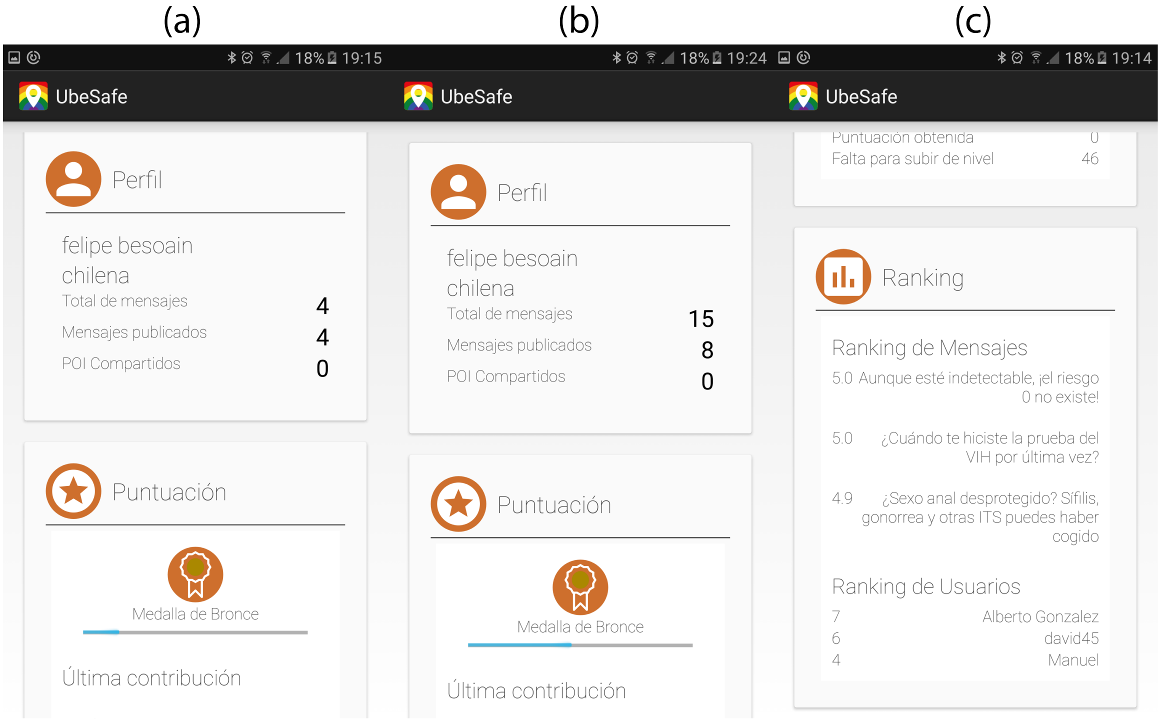


felipe besoain chilena
Total de mensajes 1
Mensajes publicados 0
POI Compartidos 0

felipe besoain chilena
Total de mensajes 3
Mensajes publicados 1
POI Compartidos 0

Figure 4.39: UBESAFE - Android client: accessing the personal profile that is public for the community with a scoreboard and current experience and medals gained in the system

The next Figure 4.40 shows how the users get more experience by sharing messages (screenshots a, b and c). First, the user has shared four messages and gotten some experience in exchange (see Figure 4.40-a). Second, the user has shared eight from fifteen private messages and get some experience in return (see Figure 4.40-b). Lastly, there is a ranking of the three most valued health messages with their average scoring. Also, a ranking of users is shown. These data is changing all the time, depending on the users' actions (see Figure 4.40-c).


(a)

UbeSafe

Perfil

felipe besoain
chilena

Total de mensajes	4
Mensajes publicados	4
POI Compartidos	0

Puntuación

Medalla de Bronce

Última contribución

(b)

UbeSafe

Perfil

felipe besoain
chilena

Total de mensajes	15
Mensajes publicados	8
POI Compartidos	0

Puntuación

Medalla de Bronce

Última contribución

(c)

UbeSafe

Puntuación obtenida	0
Falta para subir de nivel	46

Ranking

Ranking de Mensajes

5.0	Aunque esté indetectable, ¡el riesgo 0 no existe!
5.0	¿Cuándo te hiciste la prueba del VIH por última vez?
4.9	¿Sexo anal desprotegido? Sífilis, gonorrea y otras ITS puedes haber cogido

Ranking de Usuarios

7	Alberto Gonzalez
6	david45
4	Manuel

Figure 4.40: UBESAFE - Android client: gamified scoreboard and top three health messages scored by the community

The use of experience points, medals and ranking in UBESAFE is an example of gamification because it uses these elements to motivate users to participate more actively in the community. It promotes a sense of competition between the users and allows them to see their accomplishments.

4.5.2 Summary

The UBESAFE system has two major components:

1. The simple web information system: allows the health administrator to add, modify and delete messages and POI
2. The mobile application developed for Android operating system version 6.0.1.: allows the mobile users to receive notifications based on a smart context in three ways:
    - Browsing a hot link

- Using a risk application
- Being nearby a hot zone

The system develops a community for the users, considering their ideas and knowledge of hot zones. In order to provide a sustainable way of getting new data, the main experience was developed with *gamification* concepts.

This prototype follows the RESTful services principles and has two parts: an Android OS application with emphasis on ubiquitous computing and designed according to General Responsibility Assignment Software Patterns (GRASP), and a server with a web information system. The main components are reused from UBI-APP, UBINUT, and GEONUT and refined for optimization purposes. New modules and characteristics have been developed from the conclusions of focus groups.

Ubiquitous computing may be useful for alerting users with preventive and educational messages. The proposed application is non-intrusive because: 1) the users themselves decide to install it, 2) it sends a message that helps users to think about taking appropriate preventive measures, 3) it works in the background without interfering with users unless a trigger situation is detected, and 4) it develops a community through the gamification system developed. Thus, this type of application could become a valuable tool in the complex task of STI prevention between users and public or private health centers.

In the next subsection, the technical aspects of the application will be presented with its software artifacts like actors, general use case, architecture of the solution, and statechart diagrams, among others. Then, the context will be established for the testing of its results.

Finally, the results will be analyzed, with discussion related to the optimization and the functionalities and attributes of the system, as well as some conclusions and future work in this area.

4.5.3 Technical aspects

The following subsections will describe the IT artifacts that were involved in the software development process for UBESAFE.

4.5.3.1 Actors

Considering that incremental research is being developed with IT artifacts, the addition of new functionalities on a base prototype is technically straightforward to implement, given that it has modular architecture. UBESAFE has been developed considering the previous research applications UBIAPP, UBINUT and GEONUT. Therefore, the informatics component from each one were reused.

In this case, the main actors that interact with the UBESAFE system are the same as those defined for GEONUT. Thus, they are: mobile user, server user, database, information server, and mobile application. Moreover, in UBESAFE a *mobile database* actor was defined. The mobile database is an actor that stores data from the app UBESAFE locally. The main data stored is personal data from the mobile users and private information that could be shared only if the users want. The DBMS used for this purpose is SQLite.

For further description, see the actor list of GEONUT in Section4.4.3.1.

4.5.3.2 General use cases

To understand how the *UBESAFE* software behaves, four general use cases are presented. These use cases include the new functionalities and the previously described implementations of the *alert, update service* and the *AppPatrol*.

As part of the methodology and design of the research, all the new functionalities and improvements come from the conclusions of the previously developed applications (UBIAPP, UBINUT, and GEONUT). The results from each test with the target group were also taken into account as input for the design and development process of UBESAFE. The following section explains the most relevant use cases in an extended format divided into two areas: 1) client side (Android device); and 2) server side (webserver).

Client side (mobile application)

The following functionalities related to adding messages and POIs in the mobile application are crucial for the UX, see Tables 4.13, 4.14. The idea was that when users have the chance to customize UBESAFE for their purposes, they will feel a personal connection to the application, thereby enhancing the UX.

The following tables explain the use cases in a conversational and extended format, which emphasizes the interaction between the actors and the application.

The next two functionalities are related to make a sustainable system that gets messages from health professionals but also from the users. Thus, the functionality is included to share information with the community. This community is controlled by the health administrator that manages the information when it is necessary (approves, updates, refines or deletes proposed messages and POI). This functionality is described in the Table 4.13.

Use case US01:	Adding a personal POI from the mobile application
Actor(S):	Android application, information server.
Purpose:	To store a POI in the local database of the mobile device.
Summary:	This use case begins when the mobile users want to store a private POI. This POI must be stored in the private database of the mobile device and be part of the main source of POIs that the *alert service* uses to compare the users' localization versus risk zones.
Preconditions:	Main application must be running (front end).

Actor's actions	**Application's answers**
1.- This use case starts when the mobile users want to store a private POI into the device. The users select the tab add POI	
	2.- The application opens a mapView where the users can drag and drop a POI indicator.
3.- The users add the name of the POI in the GUI and drag and drop the POI indicator in the place of interest.	
4.- The mapView puts the POI indicator where the users have dropped it.	
5.- The users select to save the POI	
	6.- The application saves the POI into a local database.

Alternative flow: 5.- The users save the POI without entering a name. In this case, the application will notify the users of this requirement.

Table 4.13: This table represents the use case US01 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Use case US02:	Adding a personal message from the mobile application
Actor(S):	Android application, information server.
Purpose:	To store a personal message on the local database of the mobile device.
Summary:	This use case begins when the mobile users want to store a private message to the main list of preventive messages.
Preconditions:	Main application must be running (front end).
Actor's actions	**Application's answers**
1.- This use case starts when the mobile users want to store a private message into the device. The users select the tab add messages.	
	2.- The application opens an interface where the users can write a message.
3.- The users write a message.	
4.- The users save the message.	
	5.- The application saves the message into a local database.
Alternative flow: 4.- The users save the message empty. In this case, the application will notify the users of this requirement (message must exist).	

Table 4.14: This table represents the use case US02 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

In order to promote and motivate mobile users to participate of the community, a gamification (see Section 2.3) based scoreboard was implemented with points, medals and scoreboard recognition for each contribution that the mobile users made. Therefore, there is a external motivation to engage the mobile users providing information for the main system keeping it updated with new information. These use cases are described in a conversational and extended format in Tables 4.15 4.16.

Use case US04:	Generate a gamification board
Actor(S):	Android application, information server.
Purpose:	To generate a gamification system with a scoreboard in the mobile application.
Summary:	This use case begins when the system quantifies the amount of contribution by users. Users will get points and badges on a simple gamification scoreboard to enhance the contribution between them.
Preconditions:	The mobile users have stored and shared a private message or POI in the mobile device.

Actor's actions	**Application's answers**
1.- This use case begins when the users want to know their position on the contribution board.	
	2.- The application shows a scoreboard with trophies and medals that represents a rank in the community.
3.- The users can contribute more to the community to improve their position on the board.	

Alternative flow: 2.- The mobile device does not have an internet connection. Therefore, data cannot be transferred from the information server (the application will notify the users).

Table 4.15: This table represents the use case US04 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Use case US03:	Sharing data with the web server
Actor(S):	Android application, information server.
Purpose:	To share private data such as: POI and preventive messages with the web server.
Summary:	This use case begins when the mobile users want to share information with the main server (community). This information could be a message, a POI or all of them.
Preconditions:	1) The application has been downloaded and stored messages for the users from the information server; 2) The application has downloaded and stored the hot zones from the information server; 3) The mobile user has stored a private message or POI in the mobile device.

Actor's actions	**Application's answers**
1.- The mobile users want to share information with the main server (community). This information could be a message or a POI.	
2.- The users select the message or POI to be shared.	
3.- The users send the message or POI to the information server.	
	4.- The information server receives the data.
	5.- The information server stores the data and notifies the client of a successful transaction.
6.- The application notifies the users that the transaction is complete.	
Alternative flow: 3.- The mobile device does not have an internet connection, and 5.- The information server does not complete the transaction (in both cases, the application will notify the users).	

Table 4.16: This table represents the use case US03 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Server side (information server)

The health administrator has a major task controlling and managing each contribution from the mobile users. Therefore it is necessary to have a SWIS that checks and handles the management of the information. The following functionalities are related to the creation of a message and a POI. Since the health administration is adding new data from a health perspective, these data are automatically included in the main database for the mobile clients. It also considers internationalization allowing the administrator to add the messages in three different languages. Tables 4.17, 4.18 describe the process in detail.

Use case WS01:	Add a POI from the web interface
Actor(S):	Information server.
Purpose:	To add a POI from the web interface and stored on the local database.
Summary:	This use case begins when the administrator wants to add a POI to the system. The frontend will provide to the administrator a mapView to facilitate the localization of the POI.
Preconditions:	The administrator must be logged in the web interface.

Actor's actions	**Application's answers**
1.- This use case begins when the administrator wants to add a POI to the system.	
2.- The administrator adds a POI into a mapView interface.	
	3.- The system will get the latitude and longitude of the POI.
4.- The administrator adds a name for the POI.	
5.- The administrator saves the information in the system.	
	6.- The system notifies the users that the transaction is complete.

Alternative flow: 5.- The information server does not complete the transaction (the application must notify the user).

Table 4.17: This table represents the use case WS01 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Use case WS02:	Add a message from the web interface
Actor(S):	Information server.
Purpose:	To add a message from the web interface and store it on the local database.
Summary:	This use case begins when the administrator wants to add a message to the system. The system will allow the administrator to enter the information in three languages: English, Spanish and Catalan.
Preconditions:	The administrator must be logged into the web interface.

Actor's actions	**Application's answers**
1.- This use case begins when the administrator wants to add a message to the system.	
2.- The administrator adds a 200 character message in three languages: Spanish, English and Catalan (the messages were limited to 200 characters to optimize how the message appears on different sized screens and to avoid lengthy messages that lose users' attention).	
	3.- The system will check the length of the messages and notify the users when they exceed the maximum number of characters allowed (200 characters).
4.- The administrator saves the information in the system.	
	6.- The system notifies the users that the transaction is complete.

Alternative flow: 4.- The information server does not complete the transaction (the application must notify the users) and 2.- The administrator adds the message in one or two languages. In this case, the system must process the transaction, but the client must handle this exception (depending on the language of the device).

Table 4.18: This table represents the use case WS02 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Since the SWIS has the responsibility of managing all data of the UBESAFE system, it is necessary to consider the functionalities associated with the communications of the data between the information server and the mobile clients. It is important to note that each information that is included into the main database of the UBESAFE system (database that is transferred to the mobile clients) requires the approval of the health administration. This decision was made to keep the integrity of the data and avoid messages or POI that could be off topic or inappropriate.

Use case WS03:	Share data with the community
Actor(S):	Information server.
Purpose:	To share the selected POI and messages with the community (mobile clients).
Summary:	This use case begins when the administrator wants to share or activate messages and POI for the alert service.
Preconditions:	1) The administrator must be logged in the web interface; 2) The administrator must previously store POIs or messages on the server database.

Actor's actions	**Application's answers**
1.- This use case begins when the administrator wants to share or activate messages and POIs for the alert service.	
2.- The administrator has two ways to follow the purpose of this use case: 1) following the use case WS02; 2) approving and sharing the information (messages and POI) provided by the mobile users.	
3.- The administrator checks the information provided by the mobile users.	
	4.- The system must allow three options: 1) approve the information (it means that the contribution meets the guidelines of the administration); 2) edit the information (it means that the contribution does not meet the guidelines of the administration but can be modified); and 3) delete the information (it means that the contribution does not meet the guidelines of the administration).
5.- The administrator approves or deletes the POI or message.	
	6.- The system notifies the users that the transaction is complete.

Alternative flow: 5.- The information server does not complete the transaction (the application must notify the user).

Table 4.19: This table represents the use case WS03 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

Use case WS04:	Generate a gamification board system
Actor(S):	Mobile device, information server.
Purpose:	Generate the controller that rules the system of the scoreboard.
Summary:	This use case begins when clients request information from the web server to generate the gamificated scoreboard of the community .
Preconditions:	1) Mobile users have shared information with the community (POIs or messages); 2) The administrator has validated the information shared by the mobile users.

Actor's actions	**Application's answers**
1.- This use case begins when clients (mobile devices) request information from the web server.	
	2.- The system processes the requested information.
	3.- The system sends the information through a standard communication interface.
4.- The client processes this information.	
5.- The client generates a scoreboard with medals and distinctive colors.	

Alternative flow: 3.- The client does not complete the transaction (The application must notify the users). Therefore, it is not possible to generate a scoreboard with medals and distinctive colors.

Table 4.20: This table represents the use case WS04 in a conversational and extended format, which emphasizes the interaction between the actors and the application.

4.5.3.3 Sequence diagrams

In this section, four *sequence diagrams* will be discussed related to the previously presented use cases of the system. Besides that, some references to source code will be shown when it is required.

Client side (mobile application)

From the four use cases described in the Tables 4.13 4.14 4.15 and 4.16. The use cases *Sharing data to the web server*; and *Generating a gamification board* are considered for describing the interaction process among components.

Figure 4.41 shows the sequence diagram of the use case *Sharing data to the web server*. The method for sharing data from the mobile device to the information server is the same for both kind of data, (messages and POIs). In this scenario, two actors are involved in the process:

- Mobile users: the mobile users initialize the main activity that allows users to enter the data into the mobile device, then through the implemented restFul service the data is sent to the information server.
- Information server: in the information server, the data is handled and stored in a database for future queries.

It is necessary to mention that this process is carried out from a module that interacts with the actor information server. This module has the responsibility of communicating the data through the given interface from the mobile device to the information server.

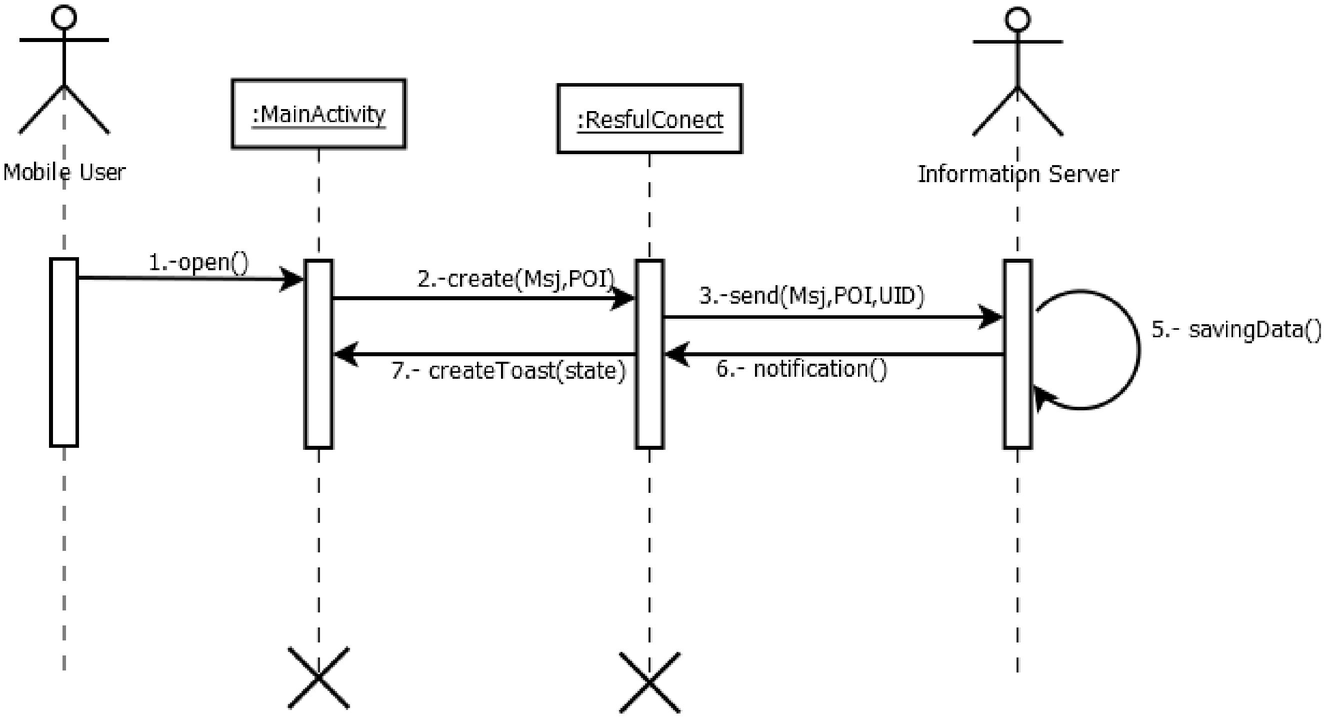


Figure 4.41: Sequence diagram: black box representation of the general use case *Sharing data to the web server*, see Table 4.13

Figure 4.42, represents the use case that generates a gamification board. In this case, there are two actors involved in the process: mobile users and information server. The process begins when the mobile users want to see their stats on the scoreboard. The scoreboardview object will create an instance of a Restfulconnect with the aim to transfer data from the information server. Here the component Resfulconnect is getting the information related to the users and sending the users UID to the information server. This process must happen every time that the mobile users access the scoreboardview because the stats are constantly changing as more data (messages and POIs) are registered in the information server (Create, read, update and delete (CRUD) to a message or POI).

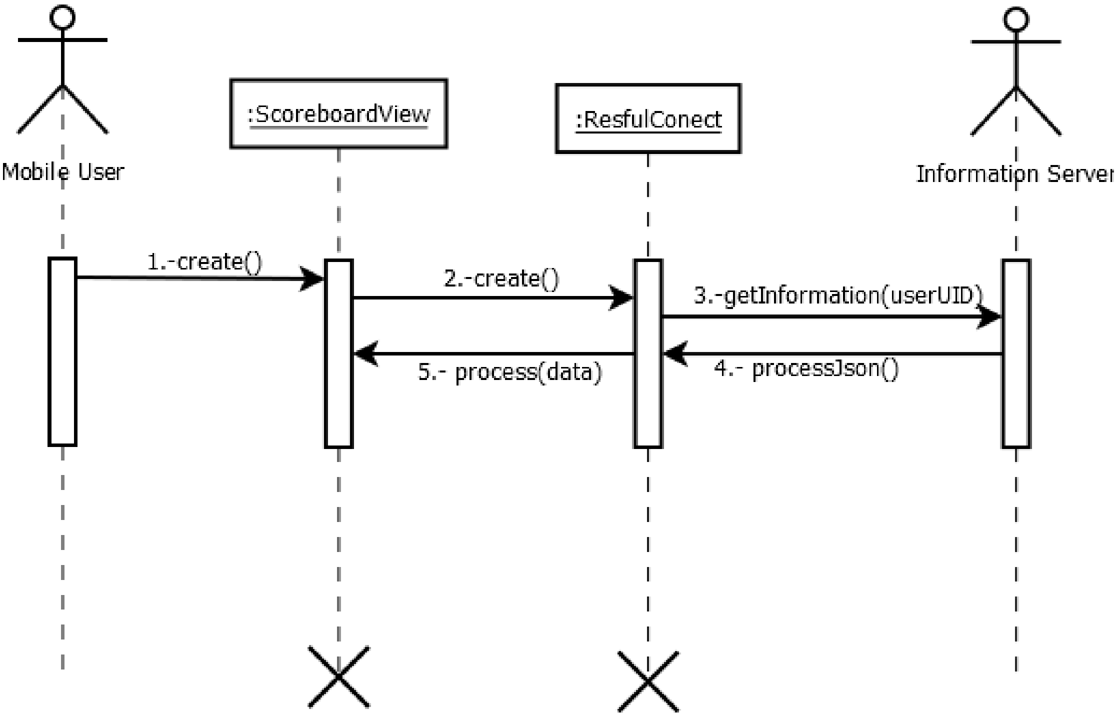


Figure 4.42: Sequence diagram: black box representation of the general use case *Generate a gamification board*, see Table 4.14

For complementary information about the final product of these processes, see in the Appendix Figure 4.36 and 4.39, respectively.

Server side (information server)

From the four use cases described in the Tables 4.17 4.18 4.19 and 4.20, the uses cases *Share data with the community*; and *Generate a gamification board system* are considered for describing the interaction process among components.

These processes are considered as the most important processes from the server side because they allow the health administrator to manage and control all the data from the system.

Figure 4.43 presents the CRUD for the messages and POIs. Two main actors are involved in the process:

- Server user: the health administrator, the process begins when the health administrator modifies a message or POI through the SWIS.
- Information server: the information server handles and processes the administrator's actions and is responsible for managing the CRUD of the data in the local database.

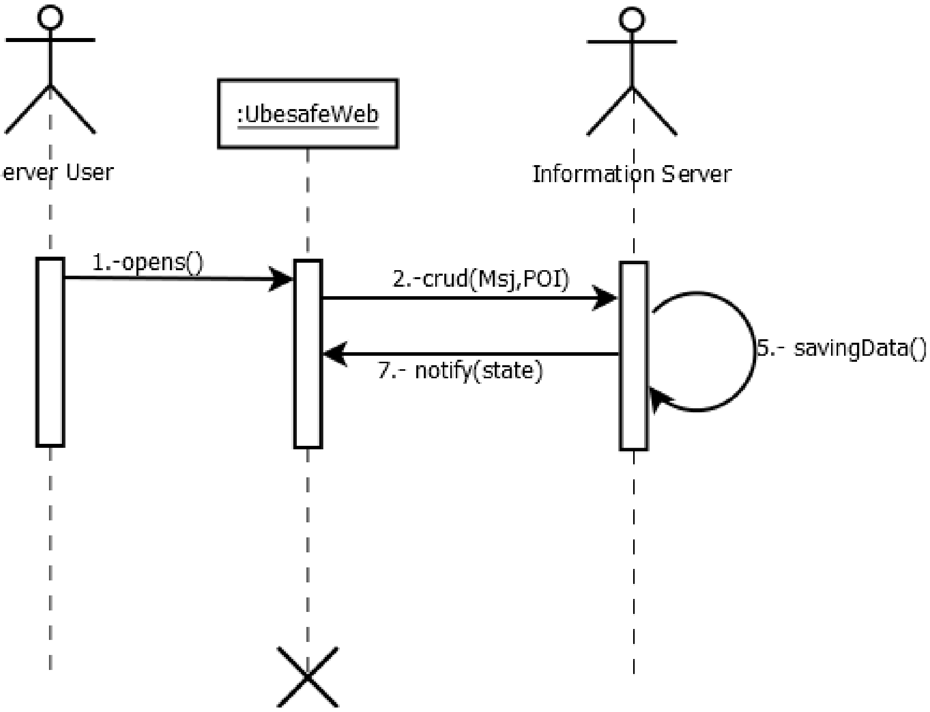


Figure 4.43: Sequence diagram: black box representation of the general use case *Share data with the community*, see Table 4.17

Figure 4.44 represents the process related to *generate a gamification board system.* In this process, the mobile applications, information server, and database are the three principal actors. Here, any mobile application can request the statistics of the mobile users from the information server. The information server will query the database and process the information for giving the information through a restful interface. The red square represents the interface of communications that allows the mobile applications to query information without being responsible for knowing the implementation of the internal process of the information server.

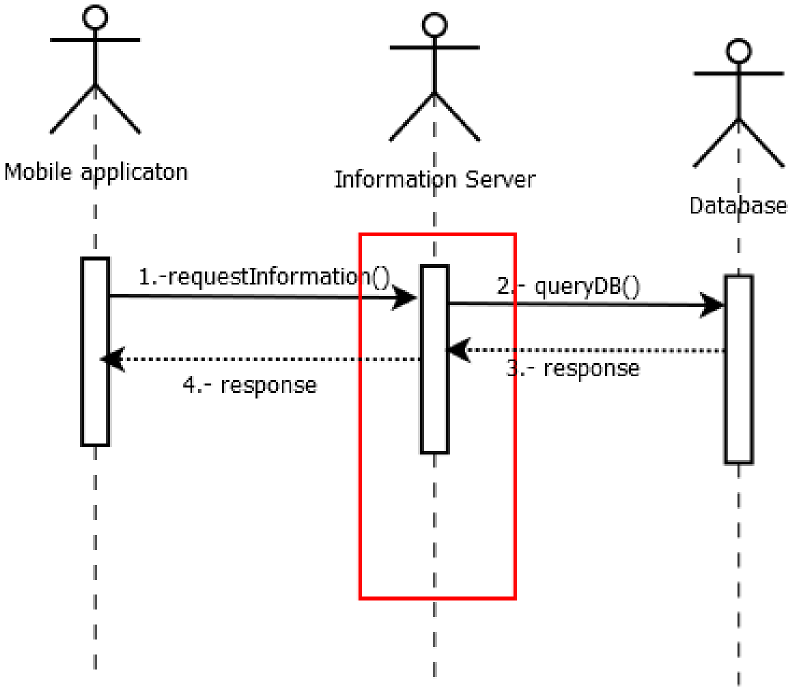


Figure 4.44: Sequence diagram: black box representation of the general use case *Generate a gamification board system*, see Table 4.18

The final product of these processes can be seen in Figures A.10 and A.2, respectively.

4.5.3.4 Statechart diagram

In the following subsection, a statechart diagram will describe the different states for all the components of UBESAFE.

Reutilization of components of software

Following the general diagram of the solution presented in Figure 4.1 and including modules from the previously described applications (*UBIAPP, UBINUT and GEONUT*), a new application called *UBESAFE* was developed. The best components from each application were taken, such as:

- **Alert service**: module responsible for getting the users' locations and finding a match with a risk zone.
- **Update service**: module responsible for getting messages provided by the *server user* to the *mobile users* and also to send the rank of them to the main server.
- **Risk service**: module responsible for detecting an application that is running on the operating system to notify the users.

Although the principal functionalities and requirements of these modules are kept, major changes were made in their structure. These changes were driven considering that the development started on Android 2.2.3 with *UBIAPP* where the first version of the *alert service* was developed. Then in the second iteration of the research, the development of *UBINUT* and *GEONUT* was on Android 4.2.3, with the new versions of the *alert service* and the *update service.*

Finally, UBESAFE has been developed in Android 6.0.1. This version comes with new features and almost all the first methods and functions are deprecated. The most important change for this work was that the architecture of the background services was migrated to the component *alert manager* for optimization purposes.

Migrating from Android background service to AlarmManager

When the first versions of the *alert service* were tried, these versions were able to run continuously from the mobile device. But in Android 4.0.3, a change was made in the maximum use of memory that an application and its services can use. Therefore, some tests show that after about two hours (depending on the device) the *background service* crashed. The main problem was that, at that moment, it was not possible to detect when the service crashed and therefore create a new instance of it.

To solve this issue, the new *AlarmManager* module was used (available from Android 4.x), that allows the scheduling of all the intents of an application. The main problem with the background services is that the OS can remove them either if they crash or if enough memory is not available. However, if an alarm is scheduled, a specific service can be invoked at the planned time of the alarm (AlarmManager). Therefore, for UBESAFE, using the AlarmManager is better than running services in the background.

An alarm (based on the AlarmManager class) provides a way to perform time-based operations outside the lifetime of the application (the same purpose of the *AlarmManager* described in the Figure 4.20 with the unbinded service).

The alarms have the following characteristics:

1. They allow firing intents at set times and/or intervals.
2. Can be used in conjunction with broadcast receivers to start services and perform other operations.
3. They operate outside of the application, so can be used to trigger events or actions even when the app is not running, and even if the device itself is asleep.
4. They help to minimize the app's resource requirements. Operations can be scheduled without relying on timers or continuously running background services.

All theses characteristic are required for improving the background services development. Considering all the technical aspects, the major changes of the *alert*,*update* and *risk* services will be described.

4.5.3.5 Alert service

As described above, repeating alarms are a good choice for scheduling regular events or data lookups. A repeating alarm has the following characteristics:

- Alarm type:
  - ELAPSED_REALTIME: fires the pending intent based on the amount of time since the device was booted, but does not wake up the device. The elapsed time includes any time during which the device was asleep.
  - ELAPSED_REALTIME_WAKEUP: wakes up the device and fires the pending intent after the specified length of time has elapsed since device boot.
  - RTC_FIRES: the pending intent at the specified time but does not wake up the device.
  - RTC_WAKEUP: wakes up the device to fire the pending intent at the specified time.
- A trigger time.
- The alarm's interval.
- A pending intent that fires when the alarm is triggered.

It is important to note that there are two general clock types for alarms: *elapsed real time* and *real time clock* (RTC). Elapsed real time uses the *time since system boot* as a reference, and real time clock uses UTC (wall clock) time. This means that elapsed real time is suited to set an alarm based on the passage of time since it is not affected by time zone/locale. The real time clock type is better suited for alarms that are dependent on current locale (more information can be found in the Android official documentation [14]).

---

[14] https://developer.android.com/index.html

The Listing 4.2 shows how an alarm based on the described characteristics is implemented. In this case, an RTC_WAKEUP alarm was chosen, that will trigger every five minutes with the *AlarmnManagerAlertService.class.*

Listing 4.2: Alert service: setting the alarm for X intervals

```
public void SetAlarm(Context context)
{
AlarmManager am=(AlarmManager)context.getSystemService(Context.ALARM_SERVICE);
Intent intent = new Intent(context, AlarmManagerAlertService.class);
intent.putExtra(ONE_TIME, Boolean.FALSE);
PendingIntent pi = PendingIntent.getBroadcast(context, 0, intent, 0);
am.setRepeating(AlarmManager.RTC_WAKEUP, System.currentTimeMillis(), 60000 * 5, pi);

}
```

Once the alarm has been set, the operating system will trigger it at the time that was configured. When it is triggered, the first private method that will call is the *requestUpdate()* which is responsible for requesting updates on the location manager. The requestUpdate() method runs independently threads for getting the location of the *mobile devices* with the provider available, for complementary information see the Listing B.1 on the appendix section.

Since the alarm controls the alert service, which involves users' locations in comparison with hot zones, it is important to mention that the algorithm of location has been changed. The new method gives a location more quickly and accurately than the old one. These are the steps to follow the algorithm:

1. Gets the system service of *locationManager*
2. If there is a provider enabled, then it will create a thread with a timer for getting data from the manager (there are two possible providers: network and GPS).
    - when the locationManager gets a new position this will be handled with the private method *isBetterLocation()*
3. The providers network and GPS are requested. The network will be polled as fast as the OS allows, and the GPS will be polled every 100 milliseconds.

4. A timeout is established.

5. The best location is compared with the *risk zones*.

The main core of the algorithm is based on requests from both providers of location of the *mobile users*, taking into consideration that the network provider is faster with results, but they are less accurate than the GPS provider. Although, the following cases should be considered based on the testing:

1. **Big cities**: knowing that big metropolises such as Santiago, Barcelona, Madrid and New York, among others, have a lot of signal towers, the network provider is shown to be faster and more accurate. However, it is important to mention that it depends on the geographical location and the conditions of the city. Therefore, a mix between network and GPS providers is the best option for getting faster and accurate location.
    - **Indoor**: the network provider is faster and more accurate than the GPS provider (also, it is known that GPS provider in an indoor area will just drain the battery without getting accurate data and at some point not even connecting to the providers).
    - **Outdoor**: the network provider is faster and very accurate, sometimes being as good as the GPS provider (also, it is known that GPS provider in an outdoor area will get the best location with this provider).

2. **Rural areas**: knowing that small cities have less signal towers than metropolises, the network provider is the worst provider for these conditions. However, it is important to mention that it depends on the geographical location and the conditions of the city. Therefore, a mix between Network and GPS providers is the best option for getting faster and accurate location.
    - **Indoor**: the network provider is faster and less accurate than the GPS provider (also, it is known that GPS provider in an indoor area will just

drain the battery without getting accurate data and at some point not even connecting to the providers).

- **Outdoor**: the network provider is faster and less accurate than the GPS provider, the GPS provider is the best provider for these conditions (also, it is known that GPS provider in an outdoor area will get the best location with this provider).

Considering these main aspects, the algorithm will request data from both providers available. All the locations from the different providers are stored in a list for a certain amount of time (defined by the alarm). Then, the method isBetterLocation() will pick the best location with the best accuracy from this list.

Consequently, the best location can be gotten in a short period of time, maximizing the resources of the mobile device. This approach gives the application better control over system resources.

It is also important to address what happens when the device shuts down. The process of shutting down happens on the smartphones for many reasons (resting, optimization, and battery, among others). In the version of the *alert service* in UBIAPP, the users must turn on manually the service for the first time. To generate a ubiquitous system, it is necessary for that task to happen automatically, so the users are not aware of running the service. By default, all alarms are cancelled when a device shuts down. To prevent this from happening in UBESAFE and for the main purpose, the application will automatically restart a repeating alarm if the users reboot or shut down the device. This ensures that the AlarmManager will continue doing its task without the users needing to manually restart the alarm.

For doing this process, it is necessary to work with a *broadcaster receiver* by using the RECEIVE_BOOT_COMPLETED permission in the application's manifest. This allows the receiving of the ACTION_BOOT_COMPLETED that is broadcast after the system finishes booting the device. Once the signal has been sent, it will be handled by the *Bootmanager* class that implements the method *onReceive*, see Listing 4.3.

Listing 4.3: Alert service: setting the alarm for X intervals

```
public void onReceive(Context pContext, Intent intent) {

AlarmManagerUpdateDBService amUDB = new AlarmManagerUpdateDBService();
AlarmManagerAlertService amAS = new AlarmManagerAlertService();
AlarmManagerAppPAtrol alarm = new AlarmManagerAppPAtrol();

amAS.SetAlarm(pContext);
amUDB.SetAlarm(pContext);

if(getBooleanFromSP("tb",pContext)) {
alarm.SetAlarm(pContext);
}
}
```

This algorithm was implemented for the three services, considering the *mobile user* consideration. So the *onReceive* method will instanciate an object of all three services and call the internal method *setAlarm* that has a different implementation in each one, basically due to the periodicity and timing of the intervals of actions.

Figure 4.45 represents the different states of the new version of the *alert service* that uses the alarm manager system. This process starts at two points:

1. When the broadcaster receiver detects that the mobile device has started (the previous state was shut down).
2. When the alarm manager system starts different threads. These threads are in charge of requesting updates from the location manager for checking internally the best location. This is marked in the red square that can be seen in this bucle.

This bucle and the lifecycle of the thread are interesting because they share a common variable which is the list of best location. Therefore, in the bucle the threads also are running in concurrency. Finally, this finishes once the timer task completes the process by killing the threads and checking if the best location has a match. This ends the lifecycle of the *alert service.*

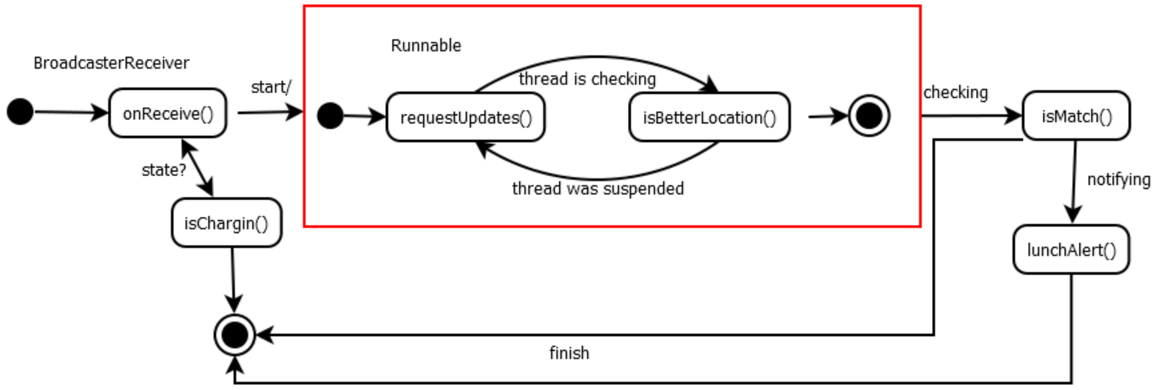


Figure 4.45: Statechart of the last version of the *alert service*

The state *isChargin* is made through a private method that has been made for the optimization process. Further details of this state will be described in the section Optimizations and key points.

4.5.3.6 Update service

The update service changes the main functionalities. In *UBINUT*, the responsibility was to get messages every time that the *server user* posts a new message on the web system. Therefore, the *update service* was a background service that was asking through a web service every minute for a new message.

The new version of the update service is responsible for getting all the data related to the application, which means messages and riskzones that are stored in a local database. This decision was made for the following optimization reasons:

- Internet connection: since the localization services do not need internet connection, some cases have been detected in which the application was able to locate the *mobile users*, but not able to show them the message because the mobile devices did not have an internet connection (signal or data).
- Optimization of resources: the background services use the mobile resources all the time. On the other hand, the alarms allow the application to control the periodicity of the recurrent task. In this case, getting new data from the database stored on the webserver is a long periodicity task.
- Version of the database: the new version of the database stored on the webserver. Every time the server users manage the POIs and messages, they can decide to release a new version.

Listing 4.4 shows how an alarm is implemented based on the described characteristics. In this case, an RTC_WAKEUP alarm is chosen that will be triggered in two different cases:

- First installation: the first time that the application is installed will call the method *SetAlarmNow* of the AlarmManagerUpdateDBService.class. This method sets the alarm and the process for getting the data 2 seconds after set has been done.

- Normal use: the normal use of the application does not need to be requesting updates constantly. Therefore, after the first installation, the alarm will be set through the method *SetAlarm*. Here the interval chosen is INTERVAL_DAY, which means the alarm will fire every day at the set time.

Listing 4.4: Update service: setting the alarm for different intervals, ref: AlarmManagerUpdateDBService.class

```
public void SetAlarmNow(Context context){

AlarmManager am = (AlarmManager) context.getSystemService(Context.ALARM_SERVICE);
Intent intent = new Intent(context, AlarmManagerUpdateDBService.class);
PendingIntent alarmIntent = PendingIntent.getBroadcast(context, 0, intent, 0);
am.set(AlarmManager.ELAPSED_REALTIME_WAKEUP, SystemClock.elapsedRealtime()+1000*2, alarmIntent);

}

public void SetAlarm(Context context){

AlarmManager am = (AlarmManager) context.getSystemService(Context.ALARM_SERVICE);
Calendar calendar = Calendar.getInstance();
calendar.setTimeInMillis(System.currentTimeMillis());
calendar.set(Calendar.HOUR_OF_DAY, 16);
calendar.set(Calendar.MINUTE, 00);

Intent intent = new Intent(context, AlarmManagerUpdateDBService.class);
PendingIntent alarmIntent = PendingIntent.getBroadcast(context, 0, intent, 0);

am.setInexactRepeating(AlarmManager.RTC_WAKEUP, calendar.getTimeInMillis(),
AlarmManager.INTERVAL_DAY, alarmIntent);

}
```

Once the alarm has been set, the operating system will trigger it at the time that was configured. When it is trigged, the first private method that will be used is the *checkCloudData()*, which is responsible for requesting updates on the *web server*. The updates are gotten depending on the users' configuration, including: provider of data and the language of the mobile device. Therefore the method *checkCloudData()* will follow these steps:

1. Getting users' configuration: the users have a preference view on the application. These preferences allow the users to configure: timing, dataproviders, and localization providers.

2. Getting data: the method will get two kinds of data

    - Getting risk zones: these zones could be gotten from an OpenStreetmap database or a private server, in this case the *web server*

- Get messages: these messages are available in three different languages: English, Spanish and Catalan. Therefore, the method will get messages in the mobile device set language.

3. Storing the data: the data is stored in a local database (sqlite).

When data is gotten from Open Street Maps, it is done using their OSM API (*OsmXapi*), which allows the collection of data from different tags such as: fast_food, food_court, and restaurant, among others. In the test with the nutrition users, this source of data was very useful because of the amount of data that was provided by the OSM community.

However, a drawback of getting information from the OSM community is that there are no tags for other elements such as: gay_bar, gay_restaurant, etc., which are locations appropriate for the use of UBESAFE. More details of those findings can be found in the Section 4.4.5.

The private server allows UBESAFE to control and manage the data entered and provided by the users. A private database of the system has been developed to store and provide this information to the mobile device, following the architecture client /-server based on the restful protocols.

It is also necessary to explore what happens when the device shuts down. By default, all alarms are cancelled when a device shuts down, following the method implemented in the *alert service*. To prevent this from happening in UBESAFE, the application will automatically restart a repeating alarm from the AlarmManager if the users reboot or shut down the device. This ensures that the AlarmManager will continue doing its task without the users needing to manually restart the alarm.

To work with AlarmManager, it is necessary to use a *broadcaster receiver* by using the RECEIVE_BOOT_COMPLETED permission in UBESAFE's manifest. This allows UBESAFE to receive the ACTION_BOOT_COMPLETED that is broadcast after the system finishes booting the device. Once the signal has been sent, it will be handled by the *Bootmanager* class that implements the method *onReceive*, see the Listing 4.3. Here, the *amUDB* instance an *AlarmManagerUpdateDBService* object will set the alarm again.

Figure 4.46 represents the different states of the new version of the *update service*. This process starts at two points: 1) when the broadcaster receiver detects that the mobile device has been started (the previous state was shut down) and 2) when the *AlarmManager* starts the process to request updates and getting data from web server. This is marked in the red square seen in this bucle.

This bucle represents the recurrent task that is done in the set time of the alarm. First, it will check for new risk zones that could come from different providers (OSM or private) and then it will check for new messages. This process could end in two ways: first, with an abnormal exit or failure; or second, once the process of getting data is done.

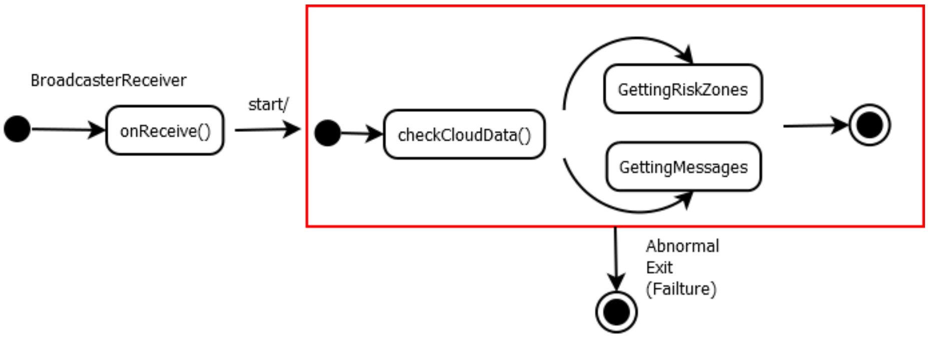


Figure 4.46: statechart of the last version of the *update service*

4.5.3.7 Risk service

When the first versions of the risk service were tried, they were able to run continuously on the mobile device. But in Android 4.0.3, a change was made with the maximum use of memory that an application and its services could use. Therefore, some tests show that after about two hours (depending on the device), the background service crashed. The main problem is that, at that moment, it was not possible to detect when the service crashed and therefore to create a new instance of the service. As explained in the subsection *Migrating from Android background service to AlarmManager*, the service was migrated to an alarm process to handle and schedule a recurring task.

The first version of the service was called *risk services*. Because it was a service that will detect any risk application on the mobile device. According to the development of the research, the name was changed to **AppPatrol** to keep watching over the applications that the users are running on their devices.

The first version of the service used the Android API for getting the list of applications that the users are running on their devices. Once the list of running applications was gotten, then a comparison was made with a list of applications previously configured by the users, known as the black list. Therefore, if any of the running applications was on the black list, the services will notify the users.

As was explained in the *UBIAPP system* (see Section 4.2), the application was developed on the Android operating system version 2.2.3, using the methods *getRunningTasks* and *getRunningAppProcesses*. But since Android 5.0+ these method has been deprecated due to security reasons (security of the operating system). Both of those methods are now deprecated and only return the own application process. Hence, if the applications run more than one process, it will be necessary to get information from those processes, but not the information of all the running processes on the device.

However, there are other internal ways to get relative information about the processes, but it has been reported that some OEMs have removed this preference. Due to the knowledge about these issues, *Android processes library*[15] was used. The Android process library is an open source library that gets information about the processes that are running on the devices using the */proc* filesystem standard by Unix. This implementation was made taking into consideration that Android OS runs with a Linux Kernel.

The directory */proc* on any Linux based operating system is very special. It is referred to as a process information pseudo-file system. It does not contain real files but runtime system information (e.g. system memory, devices mounted, hardware configuration, etc). For this reason, it can be regarded as a control and information center for the kernel. In fact, many system utilities are simply calls to files in this directory. For example, lsmod, lspci and ps are commands that process the information of the files in the directory */proc* to retrieve the parsed and processed information for each command.

Therefore, the risk service from UBIAPP was migrated to an alarm and to the external Android process library. Google has significantly restricted access to the /proc directory in Android 7.0, hence none of these methodologies will work on the latest version of Android OS due to: first, the restriction of access to the directories such as /proc (only with root access); and second, the deprecated methods to get information relative to the running processes on the device.

---

[15] More details: https://github.com/jaredrummler/AndroidProcesses

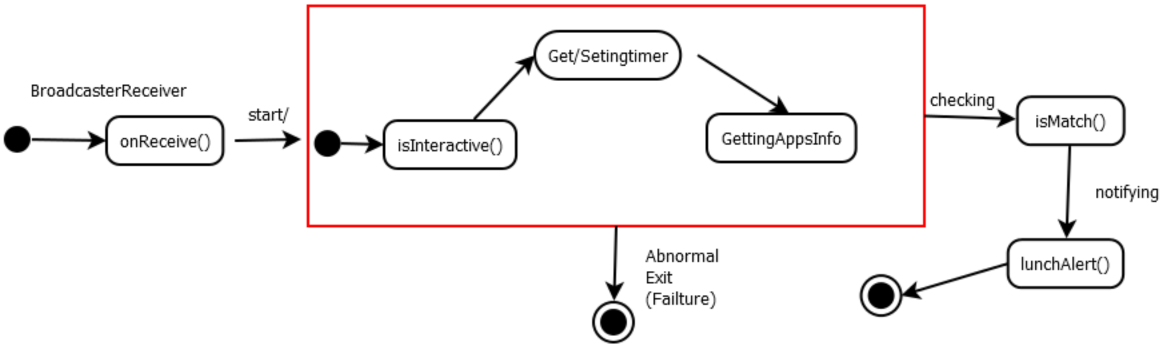


Figure 4.47: statechart of the last version of the *AppPatrol*

Figure 4.47 represents the different states of the new version of the *AppPatrol service*. This process starts at two points: 1) when the broadcaster receiver detects that the mobile device has started (its previous state was shut down) and 2) when the AlarmManager starts the process to monitor the running applications on the device. This is marked in the red square that can be seen in this bucle. This bucle represents the recurrent task that is done at the set time of the alarm.

The following are the key states of the algorithm for detection of running applications (AppPatrol):

- isInteractive: here the current state of the device is checked, using the *PowerManager* of Android API. Thus, it is possible to infer what the users are doing with the device. For instance, if the device is in a state of standby, it is possible to infer that the users are not doing anything with the device at that time. Therefore, UBESAFE detects when the users are using the device, inferring that at that time they are running a target application. This state works with the following methods:
  - *isInteractive()* for Android 4.x+ and *isScreenOn()* for Android 3.x-. When this method returns true, the device is awake and ready to interact with the users. The main screen is usually turned on while in this state. When this method returns false, the device is dozing or asleep and must be woken

before it will become ready to interact with the users again. The main screen is usually turned off while in this state. Hence, if the device is interactive, it continues to the next state *getting and setting timer*, if not, nothing is done.

- Get/SettingTimer: this state is responsible for getting the last time that the *AppPatrol* found a match and comparing that time with the current one. The difference must be the time gap previously configured by the users. This allows the AppPatrol to know when the last detection was found and to not notify the users more than once. For example, if the users are running an X app that is also previously configured on the black list, the AppPatrol will detect it and notify the users; but if the users are still using it, they don't want to keep receiving notifications because the notifications disturb the user experience (more details of this situation can be found in the results sections).

  Thus, if the difference of the last detection versus the previously configured time is positive, it continues to the next state *GettingAppsInfo*, if not, nothing is done.

- GettingAppsInfo: depending on the version of the Android OS, *getRunningTasks* and *getRunningAppProcesses* or the *Android Processes Library* are used to get the current list of running applications. Depending on the method, it is possible to obtain the Foregrounds applications or all the applications that are running on the device.

After getting the list of applications, it will be compared with the black list to find a match. If there is a match, the process will notify the users through the notification bar. This process could end in two ways: first, through an abnormal exit or failure (during the bucle); or second, once the process of getting data is done.

4.5.3.8 Architecture of the solution

UBESAFE uses the same RESTful service principles as GEONUT, UBINUT and UBIAPP that were explained in the UBIAPP architecture of the solution section (see Section 4.2.3.5).

UBESAFE utilizes all the modules from the previous mobile applications developed. Changes were made in the implementation of the modules but not in their structure of communications. The changes were driven considering that the development started on Android 2.2.3 with *UBIAPP* where the first version of the *alert service* was developed. Then, in the second iteration of the research, UBESAFE was developed on Android 4.2.3 *UBINUT* and *GEONUT* with the new releases of the *alert service* and the *update service.*

Therefore, UBESAFE has the same architecture of the solution previously described for GEONUT. Figure 4.16 shows and describes the conceptual architecture that integrates all components and services of UBESAFE.

4.5.3.9 Optimization and key points

Optimization is a key factor for mobile devices since the main power source (the battery) is limited. The resources that spend more battery on a mobile device are the screen, GPS, long processing times and an internet connection. Taking this into consideration, all the algorithms were optimized to provide the maximum efficiency in the use of resources.

Optimization and refining has been done during the whole process of software development, considering all the applications developed. Here, in UBESAFE, *AlarmManager* and *broadcaster receiver* allow UBESAFE to control the different states of the mobile device during its uptime.

For the alert service, the following states were considered to save battery.

- First, it was detected every time that the mobile device is connected to a USB cable. The connection could happen with two purposes:

   1. Charge the mobile device
   2. Connect to a computer

  In both cases, the mobile device is not being used outside of a normal routine.

- Second, the algorithm was enhanced for localization, polling the GPS less than the alert service of the other mobile applications with high accuracy.
- Third, the update service is not running all the time. It is responsible for retrieving data from the information server every other day, because the versions of data are expected to change in a period of days.

For the risk service, the following premise was considered to save battery. Every time that the users are using a contact application on their mobile devices, they are using the device. Therefore, the application with the risk services will only be detected when the screen is on. Otherwise, it is assumed that the mobile device is off or in standby mode. Hence, the current state of the device is checked, using the *PowerManager* of Android API. Thus, it is possible to infer what the users are doing with the device, starting the risk service when it is necessary rather than all the time.

Finally, the AlarmManager provides access to system-level alarm services. Using the AlarmManager allows an application to schedule tasks that may need to run or repeat beyond the scope of its lifecycle. The Android system tries to batch alarms with similar intervals or times together in order to preserve battery life. By batching alarms from multiple applications, the system can avoid frequent device waking and networking. Therefore, it saves battery and resources.

4.5.4 Testing and results

In this section, the results will be presented describing how users interact with the UBESAFE system. Afterwards, in the in the analysis of the results section, several points and conclusions will be drawn regarding how ubiquitous computing can be used as a preventive tool.

As a consequence of the research following the design and creation approach, a georeferenced application has been created that uses ubiquitous computing concepts for health prevention purposes. An Android application has been developed named UBESAFE. Functional and technical aspects of UBESAFE have been shown in the previous sections. To test the application and choose the health or preventive messages, a test was performed in two steps.

1. A functional test was done with 5 MSM for two weeks with the aim of receiving feedback on the UX through a focus group.
2. A feasibility test was then carried out with 4 MSM for a month to try the UBESAFE system with all of its functionalities.

In both experiments, the testing was carried out by both professionals from health departments (ASPB) and researchers of computer sciences, multimedia and telecommunications departments (UOC).

4.5.4.1 Functional testing context

Two experiences were considered for the testing of the UBESAFE system. The first experience was with 5 MSM, since they are at high risk of infection with HIV. Their demographic characteristics are shown in Table 4.21. In the group, 3 are in their thirties or late twenties and 2 are over 40 years of age. Most of them were Spanish (4), and 1 was from another country.

This table shows the demographic characteristics of people who helped to perform the first test of UBESAFE.

Age	Ocupation	Level of studies	Nationality
27	Nurse	Universitaries	Spain
42	Interiorista	Universitaries	Spain
30	Medic	Universitaries	Chile
37	Receptionist	Profesional	Spain
45	Actor	Licensed	Spain

Table 4.21: **Demographic characteristics of volunteers who helped to choose the messages** ($n = 5$)

The purpose of this initial test was to receive feedback on the UX through a focus group with MSM. The previous experiences with UBIAPP, UBINUT, and GEONUT were taken into account in order to test the system.

First, it was considered important to get feedback from UBESAFE just like any application on the users' mobile devices. This version of the application was released with just three functions:

1. Detection of hot zones
2. Detection of risk applications
3. Detection of a URL of contact

This version did not have the functionalities described in the use cases 4.13 4.14 4.15 4.16.

To understand the target group and their impressions, two surveys were taken:

1. Entry survey: with the objective to categorize the demographic characteristics of volunteers, typical use of their mobile devices, requirements, and behaviors in these contexts.
2. Exit survey: with the objective to know their first impressions of the applications, experience with the notification system, and abnormal behavior of the application, among other variables.

Both surveys were made by the ASPB and carried out with a focus group with the volunteers.

As can be seen in the Table 4.22, all the volunteers declare using the mobile device as their primary device for accessing to the internet. Moreover, all of them have an internet plan on their devices. Therefore, they are fully connected the whole time. In this context, they also declare to be constantly aware if the devices have some notification or message. This effect is increased by the ubiquity of the information. Today a notification on the mobile device is information from a message, text, email, or game notification, among others.

Additionally, all of the volunteers mention that they use the mobile device to chat with friends. Meanwhile, 2 of them also declare that they use the mobile device to chat with unknown people. In this context, the mobile device is also used to search for information related to health topics such as STIs, HIV, health centers and sports.

Device	**Number of volunteers**
Pc computer	2
Smarthphone	5
Tablets	3
Others	

Table 4.22: **Devices preferred to access the internet**

The volunteers used UBESAFE for two weeks with the following detections:

- *Execution of some kind of application*, applies in situations where users open applications designed for contacting sexual partners, such as *Manhunt* (see Figure 4.5).
- *Proximity to a geographical zone where sexual contacts often take place*, applies in situations where users enter or are near to a *hot zone.*
- *Detection of a target URL*, this feature applies in situations where users open a target URL.

It is important to note that this version of the application only sends a health message. Once the notification has been displayed, the software can interact with other installed applications, for example allowing the users to share the notification through email, text message, social networks, etc. Thus, users have a fully-connected experience that can also help to promote prevention among others. When they receive a notification, the users can make an informed decision regarding the possible consequences of their behavior. As a result, the use of this software raises users' awareness of their actions and encourages them to take steps to limit the spread of STIs.

All of the volunteers declare that they were able to install and configure the application without problems and the application did not compromise the standard functioning of the devices.

Regarding the three main functionalities of UBESAFE in this version (detection of hot zones, risk applications and URL of contact), 2 of the volunteers mentioned that they received health notification when they were near a hot zone or using a risk application, and 3 received a health message through the URL detection when they were navigating on the internet.

As part of the discussion and conclusions of the experience of the focus group, the Table 4.5.4.1 describes highlights to be considered:

Volunteers comments	analysis
I got a lot of messages for the use of one application	This feature was developed on purpose in the first version, to see how the mobile users will react to the notifications on demand. Next versions allowed a configuration of the timing of the notifications.
Battery consumption when I use the map	Battery consumption is an issue in all applications that require the constant processing of data. In this case, the use of the mapView consumes energy from two principal sources: localization and screen.
Repetition of the health messages in the different detections	This happens because the experiment had 20 messages for testing purposes. The messages will be presented randomly. A big database of health message is required to avoid repetition.
It is necessary to have more hot zones	Some zones are provided by the health administrator, but there is knowledge that only MSM know and could be beneficial for the community and future interventions of the public health service.

With this first initial testing it was possible to test the three most important functionalities of detection of the applications and know and classify the users with the aim of enhancing the UX of the system. This test also had a technical tracking of bugs and issues through the Google play platform for developers. Finally, there is a continuous refinement of the modules of the application as part of the iterative development methodology. All these experiences have increased the value of the product with more emphasis on the users than on the process.

4.5.4.2 Feasibility testing context

The test has been carried out with the UBESAFE mobile application. The study was designed according to the recommendations suggested by several reports for medical or therapeutic like interventions as in (Bowen et al. [75], Leon et al. [76]), especially focused on practicality issues of applying such a ubiquitous system. Four MSM volunteers were enrolled to participate actively in the feasibility evaluation of the UBESAFE system. All the participants declared that they utilized social networks through their smartphones frequently (at least twice a day). Their age was between 27 and 45 years old. It is important to consider that the UBESAFE system aims to have a preventive role making users more conscious of their actions.

4.5.4.3 Data sources and collection

The data were collected through the smartphones owned by the users. All the data obtained from the system were stored in the database as the foundation of the analysis. Moreover, the data were collected through the SWIS and analyzed concurrently. It is relevant to mention that all the users agreed to share the information for further analysis; setting this option was a requirement to run the application. Data were collected at three different times: first, through the entry survey; second, through the smartphones; finally, through an exit focus group. The objective of the entry survey was to learn more about the target group, regarding knowledge, habits, and behavior. The aim of the exit focus group was to learn about the UX experience with UBESAFE and collect software suggestions.

The application collects three types of data at different times:

1. Profile information: this information is collected the first time that the users run the application. At that moment, the users' contact information is requested such as name, email, age and nationality, see Figure 4.30. Additionally, the users must set if they want to share this information with the system or use the system anonymously. With these data, users are created on the SWIS that later are related with their scores on the health messages.

2. Perceptions: when UBESAFE detects any of the three detection systems, it will notify the users in the notification bar of the smartphone. The users will be prompted to read the health message and score it on a scale from 1 to 5, see Figure 4.37.

3. Question for the health administrator: the users were able to send questions to the health administrator in charge (professional of the ASPB).

The health administrator through the SWIS can access to see and review all the data from the system. Several actions can be done related to the CRUD of messages and POIs on the system. Moreover, the health administrator has a key role in the revision and approval of health data provided by the mobile users. In addition to the management options, the SWIS shows the list of messages and graphs of perceptions of the users, for complementary information see Figure A.11 on the Appendix section. These features add value to the system since the administrator of the system can see in real-time the positive or negative impact of a health message.

4.5.4.4 Feasibility evaluation

In the feasibility evaluation, one health professional from the ASPB participated, writing the initial database with 30 messages and responding to users questions. The initial messages were taken from the study presented in the UBIAPP testing experiment, and the questions were responded to within 24 hours. The study had a duration of 30 days; 357 evaluation of health messages were registered in total in the system (2.9 evaluation average per volunteer per day). The messages come from two sources:

1. Health administrator: the test was begun with 30 messages related to prevention of risky behaviors in MSM. Ten messages were taken from the UBIAPP study. In that study, volunteers were asked to choose the messages they found most and least suitable as preventive messages and were also offered the chance to propose new messages. In addition to this initial list of messages, the ASPB provided twenty more messages that were reviewed by health professionals of the institution. The messages were displayed randomly by the application.
2. Mobile users: as was described in the use cases 4.13, 4.14, mobile users are able to share messages and POI with the system. Therefore, a sustainable way was created to maintain the system with new information and encourage users through a gamification system. In order to keep the primary objectives of the messages, POI and system, the health administrator must review these data.

During the trial period, 357 evaluations from users were received, which rated the different messages on a 1 to 5 scale, obtaining an average of nearly 2.9 responses a day per user. The mean of the ratings received during the whole period was 4.60, indicating that the information sent to participants was highly rated in general. Sixty four messages were registered in the SWIS. From those, 30 messages are from the initial load of data from the health administrator and 34 correspond to messages shared by the users. From the 34 messages, only 10 were not considered appropriate to the system (by the administrator). Four out of ten of the highest rated messages were shared by the users rather than the administrator. These messages are shown in Table 4.5.4.4.

Message	Average scoring	Frequency
When was the last time that you got tested for HIV? (N)	5,0	10
Even if nothing has been detected, 0 risk doesn't exist!	4.95	10
HIV is invisible	4.91	11
Unprotected anal sex? You can get syphilis, gonorrhoea and other STD	4.86	11
Do you snort? The tube is personal and non-transferable	4.80	10
Take care of your partner. If you protect yourself, you protect him	4.79	12
Risk? But not in sex!	4.70	10
Do you want to stop using a condom with your guy? Let's get tested together	4.70	15
Oral sex has also some risks	4.68	10
Let us be serious against HIV.	4.6	9

Table 4.23: Messages sent by the system scored with the highest rating by the participants.

Now that the information about the health messages has been seen, it is necessary to turn to the POIs and hot zones. The initial database started with seven hot zones entered by the health administrator. After the test, the database ended up with sixteen hot zones. Therefore 56.25% of the hot zones were contributions of the mobile users.

4.5.5 Discussion of the results

UBESAFE adds new modules and functionalities to the previously developed Android applications through its modular architecture.

This software sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, the access to a specific URL on the internet, or their proximity to areas with a high probability of intercourse (*hot zones*). It also develops community for the users, considering their ideas and knowledge of the hot zones. In order to provide a sustainable way of getting new data, the main experience was developed with *gamification* concepts. It is important to note that UBESAFE wants to create awareness of each detected situation through health messages, considering privacy and users' preferences.

Therefore, the application works in four main lines:

- Sending health messages to users when the application detects a hot URL [16].
- Sending health messages to users when the application detects the use of a contact application (risk applications) [17].
- Sending health messages to users when the application detects the proximity to areas with a high probability of intercourse (hot zone).
- Allowing the users to make a community sharing messages and POI (hot zones) through the system, enhancing the experience with a gamified scoreboard.

The application was tested during 30 days; 357 evaluation of health messages were registered in total in the system (2.9 evaluation average per volunteer per day). During the trial period, 357 evaluations were received from users, which rated the different messages on a Likert 1 to 5 scale, obtaining an average of nearly 2.9 responses a day per user. The mean of the ratings received during the whole period was 4.60, indicating that the information sent to participants was highly rated in general.

---

[16] Website where users can meet or chat with unknown people.

[17] Risk application refers to any contact applications such as: Manhunt, Tinder, Badoo, and Brenda among others.

The volunteers highly valued the functionalities related to sharing information and seeing how their peers valued it. The feeling that they are part of an informed community that helps to improve the knowledge on this matter, enhance the UX and purpose behind installing and being part of this application.

Finally, further work will address three directions:

1. Improving the graphic design and UX feedback of the gamified board
2. Implementing a more accurate ontology algorithm in order to improve the automatic system that sends messages in UBESAFE, making the selected messages closer to those that a health professional would choose in each situation
3. Measuring the impact of UBESAFE as a ubiquitous system for promoting healthy habits through an evaluation methodology, such as the four group test (funding required)

Chapter 5

Conclusions and future work

There is a need to develop new preventive health methods, and this work focused on georeferenced ubiquitous computing with mobile devices because of the popularity of these devices and their connectivity with the environment. The hypothesis addressed was that preventive health messages are perceived as more useful by users when they are received with specific timing related to the users' contexts and involving a gamified user interaction. To this end, two research strategies were used.

The first strategy corresponds to the software development, while the second is a methodology to validate results.

- first, the iterative development of software.
- second, the design and creation strategy.

The combination of both approaches allowed the hypothesis to be researched, taking into consideration the whole development lifecycle of an IT artifact.

Thus, through these methodologies, IT artifacts were created and then validated through interviews and quizzes. Five specific objectives were defined with the aim to develop the research of the hypothesis:

1. **Know and define the situations that trigger a risk behavior in STIs and nutrition**: The situations that trigger a risk behavior in STIs were defined in this work as the following: One identified situation was the use of applications

for finding sexual partners. Although sexual contacts are not risky by themselves, this situation is fostering different risk behaviors such as unprotected sex with one or multiple partners, drug use, alcohol use, etc.

Another situation defined to trigger a risk behavior is when people pass by zones where sexual contacts often take place, which are defined as hot zones. It is important to note that in these cases the action itself (sexual contact) is not a risk, but rather the failure to use preventive measures when performing the action becomes a risk behavior.

The situations that trigger a risk behavior in nutrition were defined in this work as the following: One general identified situation is related to how fast-paced lifestyles and lack of nutrition education lead to poorly balanced diets and increase the risk behavior of a population. More specific situations include social pressure, lack of time for physical activities and the proximity to temptations of fast foods that are high in fat and sugar. This last situation is presented when people pass by zones where food and snacks are available such as: cafeterias, fast food courts, and restaurants, among others. These geographical zones are defined as hot zones. In this scenario, the risk behavior is defined as food consumption. Once again, in this case the action itself (eating) is not a risk, but rather the failure to make healthy choices when performing the action becomes a risk behavior.

Mobile devices accompany their users through their daily routine, and therefore they are present at opportune moments to avoid risk behavior and/or educate users about the risks of a specific action.

2. **Develop a software for mobile devices with an Android operating system, with the aim to detect situations of risk and notify users about them**: Four applications (UBIAPP, UBINUT, GEONUT and UBESAFE) were developed as part of two stages; the first stage consisted of the test applications (UBIAPP and UBINUT), while the second stage corresponded to the refined

applications (GEONUT and UBESAFE). In more detail, the test applications are:

(a) UBIAPP: This application corresponds to the first test application designed in this work for the use of preventive messages in a smart context. The application UBIAPP was the first approach to address the issue of STI and HIV prevention, focusing on testing and trying out preventive messages in a smart context with a group of MSM.

(b) UBINUT: This application was the second approach to address the issue of preventive messages in a smart context. However, in contrast with the previous application (UBIAPP), rather than focusing on STIs and HIV, UBINUT was designed for the prevention of risky behavior in relation to nutrition problems.

The refined applications are:

(a) GEONUT: This application was a part of the third approach to address the issue of preventive health messages and is the refined version of UBINUT, focusing on testing and trying out preventive messages in a smart context with a group of people with nutrition problems. It also includes concepts of geofencing for the recognition of hot zones associated with places where users can make poor nutritional decisions, such as food courts, cafeterias, street food, etc., which was not a part of the test application (UBINUT). GEONUT, as a refined application, considers the previous experiences with UBIAPP and UBINUT from a technological point of view, and the testing of GEONUT was carried out based on the conclusions from these experiences.

(b) UBESAFE: This application was the fourth and last approach to address the issue of using mobile applications with preventive health messages. It is the refined version of UBIAPP, focusing on testing and trying preventive messages in a smart context with a group of MSM, including a gamification approach for sharing data that was not present in the test application (UBIAPP). This refined mobile application considers the previous experience with the last three mobile applications (UBIAPP, UBINUT, and GEONUT) from an informatics point of view and also the experiences of each test carried out with each respective target group.

3. **Promote health behaviors through messages when people are more committed to have a risk behavior**: All messages were distributed through the users mobile devices, taking into consideration the possible risk behavior detected.

   Different tests were established related to 4 distinct ways of interacting with messages. These different types of messages were: static, dynamic, evaluated and contributed by the community (explained in the application UBESAFE and specific objective 4). Each application developed explores a different approach of handling messages and constructing an evolutionary solution through past experience.

   With regards to the messages for STIs, users are told to do something, rather than not to do something (for example: ways to have safe sex). It is also important to note that potential users of the applications are people that are already concerned with their sexual health, and therefore such messages might be more likely to have a positive effect. For UBIAPP, the situations that are defined as more committed to have a risk behavior, and thus trigger the message, are:

   - Execution of a contact application, which applies to situations in which users open applications designed for contacting sexual partners, such as Manhunt.
   - Proximity to a geographical zone where sexual contacts often take place, which applies to situations where users enter or are near to a hot zone. For

UBESAFE, the situations that can trigger the message are:

- Execution of a hot URL.
- Execution of a contact application (risk applications).
- Proximity to areas with a high probability of intercourse (hot zone).

It is important to consider that UBIAPP was the first approach to the problem and the messages were delivered through an internal database. Messages were constructed and evaluated with a target group through the ASPB.

UBESAFE was the last approach to the problem, and thus took into account the functionalities and experiences from the previously developed applications (in both the nutrition and STI fields). Messages were constructed and evaluated with a target group through the ASPB. It is important to note that UBESAFE includes the functionality to let users propose their own messages and complement the database of health messages.

With respect to the messages for nutrition, all messages were distributed through the users mobile devices, taking into consideration the possible risk behavior.

Different tests were established related to distinct types of messages; more specifically, dynamic messages with evaluation by the users (UBINUT), and a mix of static and dynamic messages (GEONUT). Each developed application explores a different approach of handling messages in a way that is evolutionary, based on past experience.

UBINUT sends health messages to users supervised by a nutritionist. The underlying idea is to send messages related to a specific time of day, such as breakfast, lunch or dinner. Moreover, it is necessary to consider other factors, for instance, the weather conditions or holiday festivities, among others variables. Thus, users won't get a message encouraging them to do sports outside when it is raining, or tips to have a healthy breakfast at lunch time. For UBINUT, the situations that are defined as more committed to have a risk behavior, and thus trigger the message, are:

- Mealtimes (breakfast, lunch and dinner).
- Holidays.
- Specific weather situations.

It is important to clarify that for UBINUT, the health messages were sent to users by a nutritionist. Also, feedback was sent and received for the preventive messages at the exact time that the messages were received.

GEONUT is an improvement to the main functionalities of UBINUT. Therefore, it sends health messages to users supervised by a nutritionist, but in addition, it will automatically recognize certain hot zones related to places where the users can make poor nutritional decisions, notifying them to raise awareness of the situation. The underlying idea is to send messages related to a given moment of day, such as breakfast, lunch or dinner, also according to certain geographic positions. Similar to UBINUT, the application offers dynamic feedback for the received messages.

For GEONUT, the situations that can trigger the message are:

- Proximity to areas with a high probability of making poor nutritional decisions, known as hot zones.
- Mealtimes (breakfast, lunch and dinner).
- Holidays.
- Specific weather situations.

For both applications, messages come from two sources: 1) the national health guide and 2) a group of nutritionists. The origin of the national health guide is related with the needs to establish a set of public health messages that consider the current epidemiologic profile of Chilean society, with the aim to help people achieve a balanced diet and healthy weight. The nutritionists created additional messages to complement those from the guide, drawing on their academic and professional experience.

4. **Develop a gamified system to increase user interaction**: Considering this aim and the experiences developed in UBIAPP, UBINUT and GEONUT, new functionalities were included in UBESAFE. The system develops a community for users, considering their ideas and knowledge of hot zones. In order to provide a sustainable way of getting new data, the main experience was developed with gamification concepts.

   Four specific use cases were designed and developed. In order to promote and motivate mobile users to participate in the community, a gamification based scoreboard was implemented with points, medals and scoreboard recognition for each contribution that the mobile users made. Therefore, there is an external motivation to engage the mobile users. Another benefit to this interaction is that users provide data to the main system and keep it updated with new information. The UBESAFE system has two major components:

   (a) The simple web information system: allows the health administrator to add, modify and delete messages and POI (created by the administrator or mobile users).

   (b) The mobile application: allows the mobile users to receive notifications based on a smart context in three ways:

      - Browsing a hot link.
      - Using a risk application.
      - Being nearby a hot zone.

   This application was tested with four MSM that rated the sent messages and contributed their own messages to the system. The messages in general were highly rated, and the user written messages were considered appropriate by the administrator and highly rated by other users. The gamification module implemented makes the system more dynamic by making it more personal. Users receive points, medals and there is a public scoreboard that incites them to contribute to the system. In addition to being motivational, the gamification also provides a sustainable way to add more relevant information for the hot zones and health messages.

5. **Evaluate the IT solution in at least two countries**: Although not all 4 applications were tested in both countries, tests were carried out both in Chile and in Spain, as detailed below.

    - **UBIAPP**: To test the application and choose the healthy messages, a test was performed with 17 MSM in Barcelona, Spain. This testing was carried out by both professionals from health departments (ASPB) and researchers of computer sciences, multimedia and telecommunications departments (UOC).
    - **UBINUT**: Forty volunteer university students were enrolled to participate actively in the evaluation of the UBINUT system in Talca, Chile. These volunteers actively used social networks on their smartphones and were college aged. In general, they had an under-risk BMI level, however this was considered appropriate because the system aims to have a preventive role more than a remedial one.
    - **GEONUT**: The test of the application was planned in Barcelona, since the first experience, with UBINUT, was in Chile. Nevertheless, the test was not be carried out because a functionality test revealed that the application functions related to geofencing would not work in big cities like Barcelona. Therefore, no full test was completed for GEONUT.

      The decision to cancel the test in Barcelona was based on two reasons, both related to the inappropriateness of GEONUT for large, metropolitan areas. First, there was an excessive number of hot zones in major cities; thus users are constantly in a hot zone. Second, the location of users work or homes were within hot zones. This makes meaningless the detection of situations related to risk behavior.

Since the originally planned test in Barcelona could not be carried out, the decision was made to include the results of the smaller-scale functionality test, in order to present some sort of results within this work. This functionality test was done in Talca, Chile (a small town), followed by an international functionality test in San Francisco, California to determine the appropriateness for a later Barcelona test (since both San Francisco and Barcelona are big cities). In contrast to full tests, since the functional tests were created simply to test the functionality of the software, they are characterized by a small number of participants.

- **UBESAFE**: Four MSM volunteers were enrolled to participate actively in the feasibility evaluation of the UBESAFE system in Barcelona, Spain. All the participants declared that they utilized social networks through their smartphones frequently. Their age was between 27 and 45 years old. It is important to consider that the UBESAFE system aims to have a preventive role making users more conscious of their actions.

5.1 HIV and other STIs

Although people live in a society where it is easy to access information, there are still many who are not aware of STIs and how they spread as a consequence of not taking appropriate preventive measures. In addition, mobile devices and applications now allow users to meet nearby partners and have *express* dates, giving them little time to think about safety measures. In this work, two applications have been proposed (a test application and a refined application) that provide users with information that promotes safe sex: UBIAPP and UBESAFE.

The aim of UBIAPP, the test application, was to investigate how ubiquitous computing could be useful in preventive health for HIV contagion in MSM, and in particular, for generating automatic detection of POIs through a geofencing approach and the detection of risky applications. The refined application UBESAFE also focuses on testing and trying preventive messages in a smart context with a group of MSM, but includes an additional gamification approach for sharing data. Both applications work in the background and users are unaware that they are running. However, users have to install the application, i.e. its use is voluntary. The target of the applications are, therefore, people who are already conscious about STIs but forget about safety measures when using mobile devices to meet partners. The messages displayed do not try to prevent the behavior itself, but rather to inform the users of preventive measures that should be taken into account. The final decision is then up to the users.

The main functionalities of the applications are:

- Sending health messages to users when the application detects the use of a contact application (risk applications) [1].
- Sending health messages to users when the application detects the proximity to areas with a high probability of intercourse (hot zone).

---

[1] Risk application refers to any contact applications such as: Manhunt, Tinder, Badoo, and Brenda, among others.

- Allowing the users to make a community, sharing messages and POI (hot zones) through the system, enhancing the experience with a gamified scoreboard.

- Sending health messages to users when the application detects a hot URL [2].

It is important to note that only the first two functionalities corresponds to UBIAPP, the first and test application developed in this research, whereas all the listed functionalities correspond to UBESAFE. Therefore, the UBIAPP functionalities are common points for both applications. A group of MSM volunteers participated actively in the testing of both applications. In the first experience, they were asked to choose the messages most and least suitable as preventive messages and were also offered the chance to propose new messages; this process was done through focus groups. In the second experience, this process was automatized, including it as a functionality of the application to make it sustainable in a long lifecycle. This experience was enhanced with gamification topics giving the users points, medals, and a scoreboard in order to promote the sharing of information. Considering all theses aspects, the results of the tests suggest that this type of application could help to reduce the high incidence of STIs, including HIV infection. The main advantages of this application are:

1. Currently, it is common for people to use mobile devices for dating, so these devices are the most direct platform from which to launch preventive messages.

2. The preventive messages arrive at the most appropriate time, when the sexual contact is imminent, so that users are alerted at their moment of greatest vulnerability.

3. Users install the application voluntarily, to decrease the possibilities that it is perceived as intrusive, which is one of the main factors to be considered suggested by the target group.

---

[2] Website where users can meet or chat with unknown people.

5.2 Bad nutrition

Regarding nutrition, the aim of this research was to investigate how ubiquitous computing in general, and smartphones in particular, could be useful and contribute to improve nutrition behaviors. Smartphones are considered useful for this goal because they are very popular, have internet connection, and are constantly carried by users and used in multiple contexts. In this work, two applications have been proposed that provide users with information that promotes healthy nutritional behavior. Two mobile applications, UBINUT and GEONUT, were developed. The aim of UBINUT was to improve nutrition behaviors. Moreover, the principal focus of GEONUT was to extend the possible context where the application can prevent a risk behavior in a defined geographical location (geofencing). Both applications work in the background and users are unaware that they are running (ubiquitous context) on their mobile devices. They send context aware messages to users.

UBINUT sends health messages considering two special contexts: time and weather. The messages are posted by a nutritionist, who decides which message to send and when. Moreover, users are able to rate each message once it is received and ask direct questions to the nutritionist.

The target of the application is, therefore, people who are already conscious about the need to develop healthy habits and nutritionists who want to try a different method to hold the attention and to engage their patients. The focus of the messages is on making users aware of healthy habits in a special context, but it is important to note that the final decision is then up to the users.

GEONUT adds new modules and functionalities to the UBINUT Android application which has modular architecture. The principal focus of GEONUT is to extend the possible context where the application can prevent a risk behavior in a defined geographical location (geofencing). Another difference between GEONUT and UBINUT is that the sending of messages by GEONUT when the users are near POIs is automatic, instead of sending messages mediated by a nutritionist. It also solves the need for connectivity with a local database of messages and POIs. Therefore, there is no need to be connected to the internet in order to get a notification in a ubiquitous geographic location context.

From the test developed, it can be seen that users like their active participation, thanks to the rating of the messages. It has been seen that messages must be related to users' contexts and daily routines to achieve the proposed goals. In this way, users feel comfortable and receive the advice in a positive way. They also value positively the interaction with the nutritionist. Therefore, it can be concluded that it is important that, to be useful, this kind of application needs some active participation from the users: they like to be not only a passive receptor of messages, but also interact with them. Additional research must be done to determine the influence and the impact of ubiquitous technology on the behavior of the users, including new variables related to their position, actions, environment and daily routine.

The system contributes to efforts to promote healthy lifestyles among young adults, taking into consideration their context. This has a significant impact on knowledge about nutrition topics within a target group. As a consequence, students are able to make better decisions in their daily routine, taking into account information and aspects previously unknown to them.

5.3 Comparison between the refined applications

The refined applications of this work correspond to GEONUT and UBESAFE.

GEONUT's test application was UBINUT. On the other hand, UBESAFE's test application was UBIAPP. All these applications were developed with the aim of studying how ubiquitous computing could be useful in preventive health for nutrition and in preventive health for STI and HIV infection in MSM, reducing risky behaviors in both areas.

The following list will describe the common functionalities that GEONUT and UBESAFE share:

- Sends health messages to the users with the aim of making them more aware of their actions.
- Allows a health professional to send additional messages, so that messages are dynamic (in the test applications no additional messages could be added and messages were static).
- Updates an internal database from a webserver, automatically adding the new information and messages.
- Allows users to rate the health messages through a Likert scale.

On the other hand, some differences can be found since GEONUT and UBESAFE have different approaches to the use of geofencing for detecting POIs. In GEONUT, the test showed that geofencing for nutrition could be used in a small town, but not in a big city, due the excessive nutrition POIs that can be found in big cities. However, UBESAFE can be used in a city of any size, since the POIs for STIs and HIV are less concentrated. It is important to note that the main approach in GEONUT for getting information was through OSM services, from an external database. In contrast, in UBESAFE the main approach for getting information was through a private database with a limited number of POIs. Though the number of POIs is more limited, UBESAFE allows users to personalize POIs and messages, while GEONUT does not.

Another important difference is that UBESAFE includes functionalities associated with the detection of risk applications and hot URL. GEONUT does not have these functionalities because, though they are relevant for STI prevention with UBESAFE, they are out of context for nutrition prevention.

The main functionalities of UBESAFE, which was the final and thus most advanced application, are:

1. Sending health messages to users when the application detects a hot URL [3].
2. Sending health messages to users when the application detects the use of a contact application (risk applications).
3. Sending health messages to users when the application detects the proximity to areas with a high probability of intercourse (hot zone).
4. Allowing the users to make a community sharing messages and POI (hot zones) through the system, enhancing the experience with a gamified scoreboard.

The first two functionalities are very particular for prevention of STIs and HIV in MSM due to the number of websites and risk applications related to the problem. However, the last two functionalities could apply for both contexts (nutrition and STIs).

Further work must be done to test these functionalities of UBESAFE with users but for a nutrition context, to study the feasibility of using the same system for both contexts.

From an informatics point of view, all the applications are made with the same approach of communications using the RESTful services principles in conjunction with various open source components. Every new functionality took into consideration the previous modules developed for each application, evolving every product of software from a initial base prototype (test application) to a more refined application.

---

[3] Website where users can meet or chat with unknown people.

5.4 About the experiments and development

Two test applications were developed in order to do a preliminary study and get the first findings from the research on prevention of STIs and in the nutrition field. These applications were UBIAPP and UBINUT.

The main features of UBIAPP were the use of automatic georeferencing of POIs used in a first approach to the geofencing topic related with hot zones and users locations. The detection of risk applications was also included on the smartphone as part of the recognition algorithm to detect when users are more likely to establish contact with MSM. In addition to these two main features (georeferencing POIs and detection of risk applications), static preventive messages were also included, provided by the ASPB and then rated and discussed with an active group of volunteers, in this case, MSM.

These activities were carried out at the ASPB and directed by health professionals in charge of the program for the prevention of STIs. It is important to note that the motivation of the MSM volunteers was their interest in the work with the aim to contribute and be part of this new strategy, and they did not receive any other type of compensation or incentive to participate. Also, the participation of the volunteers was only possible due to previous interventions and relations with the ASPB.

It is necessary to highlight that all the research was done with a small group of people; nevertheless, for this kind of research with this particular group, even though the test group was small, the results are qualitative and representative, which makes them valuable to take into consideration for future trends. On the other hand, a drawback of working with such a small group is that the use of more sophisticated statistics to get additional quantitative data is not possible due the limited number of participants. Regarding the development of the application, the work was done with services available in the API of Android OS. The focus was on developing a functional prototype that could be scalable for the next steps of the research. A drawback was the use of battery of the application since the service is checking the users' position between intervals. The constant use of GPS will drain the battery in a few hours. So, to reduce the battery drain of the *risk service*, it was decided to let users choose a reasonable time interval from the configuration interface. After some laboratory testing, it was decided to offer: three, five and ten minutes, making an analogy to extreme, normal and light monitoring in the preferences.

This work was published and can be reviewed at (Besoain et al. [78]), and described in depth in section 4.5.

The second test application was UBINUT, which was designed for the nutrition field. The main features to test were how to obtain feedback from the users on a Likert scale in real time and send dynamic messages through a system according to the indications of a health professional. The preventive messages were provided by a nutritionist from UTalca and then rated by the users through their mobile devices. The group of volunteers, in this case, were university students who were aware of bad nutrition problems.

The experiment carried out was a feasibility test to try contextualized messages managed by a nutritionist and the messages scores (scored by users on a Likert scale). After the test, a survey was carried out to get more information about the user experience with the application and establish if messages were received correctly. Even though the test began with a group of 53 volunteers, at the end of the experiment the group had diminished to only 16, which resulted in a lack of representative data to analyze with associated statistics.

From the developed test, it can be seen that the majority of users participated in the ranking of messages, suggesting that they valued participating actively (otherwise, they would have skipped the step of ranking messages). It was also seen that messages must be related to users' contexts and daily routines to achieve the proposed goals. Thus, users feel comfortable and receive the advice in a positive way. Based on the feedback from the exit survey, they also value positively the interaction with the nutritionist. Therefore, it was concluded that it is important that, to be useful, these kinds of applications have some active participation from the users: they like to be not only a passive receptor of messages, but also interact with them. However, the main drawback of UBINUT is that the nutritionist cannot send the message when the users are near a restaurant or a cafeteria, where it is important to be aware of healthy food.

This work was published and can be reviewed at (Besoain et al. [79]), and described in depth in section 4.3.

With the results obtained by the test carried out of the test applications, it was possible to establish a base for the refined applications that took concepts and the results from the test applications.

From an informatics point of view, all the applications were made with the same approach of communications using the RESTful services principles in conjunction with various open source components. Every new functionality took into consideration the previous modules developed for each application, evolving every product of software from an initial base prototype (test application) to a more refined application. The main drawbacks of the development were: 1) the version of the Android API: at the start of the research Android 2.X was used, while the last application was finished with 6.X; therefore, major changes were made in the Android OS. 2) The Android security system does not allow access to information about the applications running on the smartphone. Therefore, for detecting risk applications it was necessary to work with low level libraries and processes.

With regards to the first refined application, GEONUT, new modules and functionalities were added to the UBINUT Android application, which has modular architecture. This functionalities are automatic georeferencing, automatic notification of messages and the use of a Likert scale to get users perceptions. In this way, the principal focus of GEONUT is to extend the possible context where the application can prevent a risk behavior in a defined geographical location (geofencing). Another difference between GEONUT and UBINUT is that the sending of messages by GEONUT when the users are near POIs is automatic, instead of sending messages mediated by a nutritionist. It also solves the need for connectivity with a local database of messages and POIs. Therefore, there is no need to be connected to the internet in order to get a notification in a ubiquitous geographic location context.

The main drawback found in the test was that the number of hot zones georeferenced in OSM is massive. Therefore, the number of detections is greater in a short period of time. It was also concluded that the distance between hot zones and the users' homes or work is within the geofence zone. These issues are related to a big metropolis because the population is concentrated in apartment buildings with commerce and markets nearby. In contrast, in small towns, the population is concentrated in houses, and the food markets are spread downtown, which is related to a certain geographic area. In a big metropolis, this is distributed throughout various sectors. For this reason GEONUT was found to work efficiently in small towns, but was less successful in big cities.

With regards to UBESAFE, the second refined application, the context taken into consideration was the results of the three previously tested mobile applications. From each test, the best-evaluated functionalities were taken. The refined versions also took into account the volunteer feedback to improve upon the test applications.

It can be seen from the results from the testing of UBINUT and GEONUT that users like their active participation, thanks to the rating of the messages, but it is necessary to increase their UX with more instances where they can configure the application or contribute to the main purpose of preventing risky behavior. It was seen that messages must be related to users' contexts considering their daily routine, considering location and other actions that they carried out with the mobile device.

Moreover, in order to make a sustainable system focused on prevention of health issues, community is an important concept that should be considered and addressed with a gamification approach in this system.

UBESAFE is an improvement of the last three mobile applications developed, UBIAPP, UBINUT and GEONUT. UBESAFE sends preventive notifications to users when it detects situations such as the activation of particular applications on their smartphones, the access to a specific URL on the internet, or their proximity to areas with a high probability of intercourse (*hot zones*). It also develops a community for the users, considering their ideas and knowledge of the hot zones. In order to provide a sustainable way of getting new data, the main experience was developed with *gamification* concepts. It is important to note that the goal of UBESAFE was to cause awareness of each detected situation through health messages, taking into account privacy and users' preferences. Volunteers remarked that privacy and the fact that their actions or positions were not logged is important for this application.

The application was tested during 30 days; 357 evaluation of health messages were registered in total in the system (2.9 evaluation average per volunteer per day). During the trial period, 357 evaluations were received from users, which rated the different messages on a Likert 1 to 5 scale, obtaining an average of nearly 2.9 responses a day per user. The mean of the ratings received during the whole period was 4.60, indicating that the information sent to participants was highly rated in general. The volunteers highly valued the functionalities related to sharing information and seeing how their peers valued it. Volunteers reported feeling that they are part of an informed community that helps to improve the knowledge on this matter, enhance the UX and purpose behind installing and being part of this application.

This work is been prepared for submission process, and is described in depth in section 4.5.

5.5 Under the point of view of Public Health

With regards to health education and prevention, today there are many different strategies available. Public institutions continually devote resources to these strategies since they are vital for a healthy community.

In the past years, the number of health campaigns, alongside the number of locations where this type of activity is implemented, has grown substantially. In the present, many different types of methods can be found to spread health information and prevention. Some of the main methods include publicity projects, outreach work with groups of individuals that may be at risk, the monitoring and control and a national level of items recognized as having a negative impact on health (whether they be foods, drinks or drugs), programs at educational institutions and the use of social media, to name a few.

In todays society, there is a need for creating new education and prevention systems in the context of everyday life. Moreover, ICT has become an integral part of the lives of those that use it. This is particularly true for mobile devices since they have a high level of connectivity and data is omnipresent. The omnipresence of this information is an opportunity to connect with users and introduce education to encourage positive health decisions in the context of their lives; this can be seen in the way that mobile devices commonly travel with their users through their everyday activities, and thus can be used to reach users at the precise moment when they may be likely to engage in risk behavior or need information about the risks of a current decision.

There are topics that are only recently emerging for discussion in Chilean society, such as: HIV, GAY,SAM, and MSM among others. In this context, the UN announced the world report ”Ending AIDS 2017”, where while it showed positive figures in the international scenario regarding HIV control, it revealed worrying figures for the local reality: Chile is the country in Latin America with the largest increase in the number of new cases between 2010 and 2016.

The UN reported that in Chile the number of people infected increased between 2010 and 2015 by 34%. Currently, according to the Institute of Public Health, an increase has been noted of 66%, which is mainly concentrated among young people between 15 and 29 years old.

The key populations most affected by HIV in Chile are:

- Prisoners, with an HIV prevalence of 0.4
- Sex workers, with an HIV prevalence of 1.1
- Gay men and other men who have sex with men, with an HIV prevalence of 20.3

This report shows the importance of public policies and preventive measures to control and inform the society. In the current work, all the studies concerned with STI and HIV prevention were concentrated in Barcelona, but there is an opportunity to contribute to the knowledge and public health preventive methods in Chile for this topic.

5.6 Future work

The use of ubiquitous computing through mobile devices for prevention in the health field is still an undergoing research topic. Many important questions and drawbacks related with these tasks remain as open challenges. In this sense, some future directions can be visualized to guide our work, as four lines described as follows:

- In applications for prevention in the health field, the incorporation of other information could be investigated. More specifically, the internet of things could be used in relation to the context of a POI in order to improve the algorithm for the detection of risky behavior as well as the notification system of preventive messages.

- It could also be of interest to study a way to incorporate more data with the aim of making profile classification through specialized algorithms. Specifically, it would be interesting to develop an in depth analysis of the message rating system.

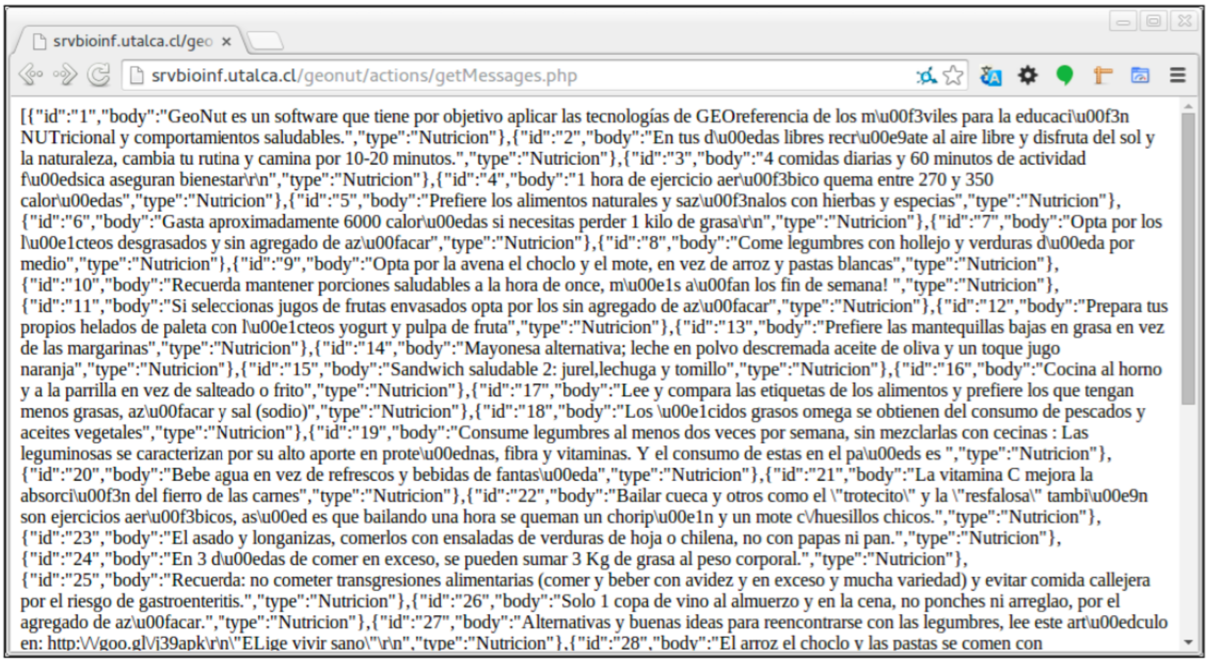
- An important area for future work is to continue with the validation of the refined applications (GEONUT and UBESAFE) since both were made with the support of the ASPB and with only a small group of volunteers to explore this novel way to carry out prevention in health. Its necessary to explore and do an in depth test with experiments, such as designing a pre-test/post-test control group or Solomon four-group test.

- The UBESAFE system can be adapted and tested for a nutrition context, since this system contains features that were only tested in relation to STIs, namely: application detection and gamification features. Thus, a feasibility study could be carried out for using the UBESAFE system for not only STIs but also bad nutrition.

Appendix A

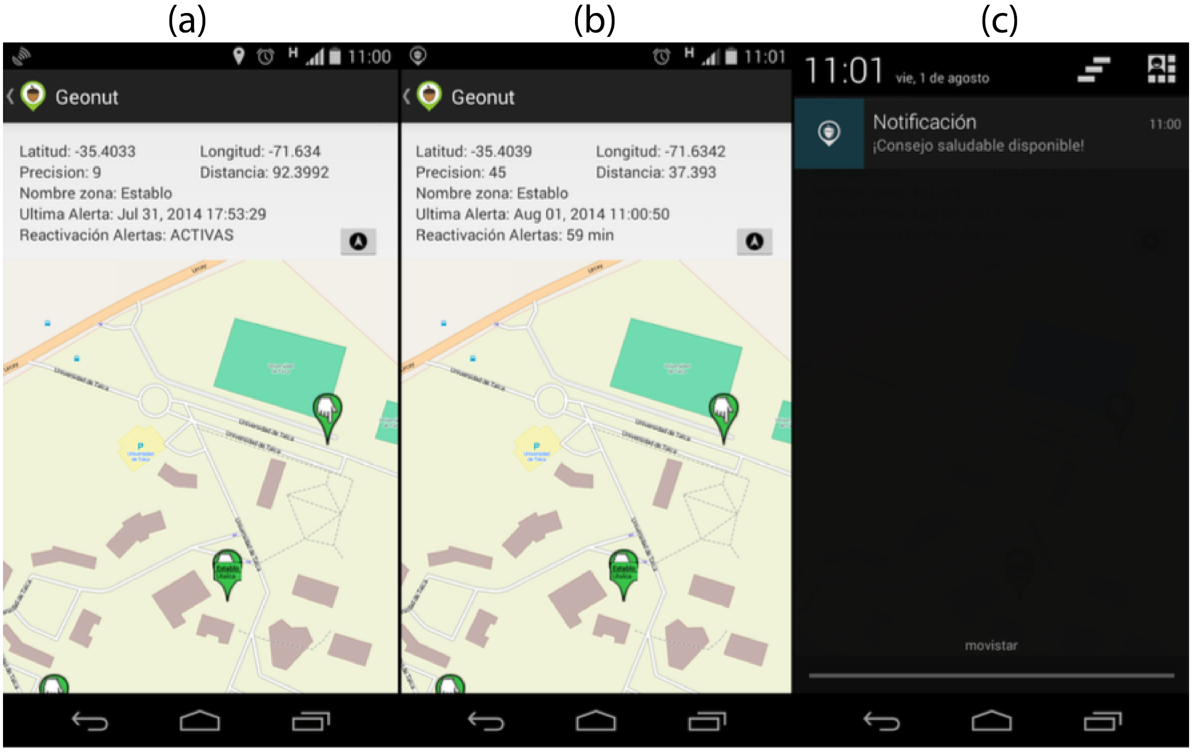
Appendix

Figure A.1: Sequence Diagram: Black box representation of the general Use Case ⟨⟨General use of the system⟩⟩, see Table 4.8



```
[{"id":"1","body":"GeoNut es un software que tiene por objetivo aplicar las tecnologías de GEOreferencia de los m\u00f3viles para la educaci\u00f3n
NUTricional y comportamientos saludables.","type":"Nutricion"},{"id":"2","body":"En tus d\u00edas libres recr\u00e9ate al aire libre y disfruta del sol y
la naturaleza, cambia tu rutina y camina por 10-20 minutos.","type":"Nutricion"},{"id":"3","body":"4 comidas diarias y 60 minutos de actividad
f\u00edsica aseguran bienestar\r\n","type":"Nutricion"},{"id":"4","body":"1 hora de ejercicio aer\u00f3bico quema entre 270 y 350
calor\u00edas","type":"Nutricion"},{"id":"5","body":"Prefiere los alimentos naturales y saz\u00f3nalos con hierbas y especias","type":"Nutricion"},
{"id":"6","body":"Gasta aproximadamente 6000 calor\u00edas si necesitas perder 1 kilo de grasa\r\n","type":"Nutricion"},{"id":"7","body":"Opta por los
l\u00e1cteos desgrasados y sin agregado de az\u00facar","type":"Nutricion"},{"id":"8","body":"Come legumbres con hollejo y verduras d\u00eda por
medio","type":"Nutricion"},{"id":"9","body":"Opta por la avena el choclo y el mote, en vez de arroz y pastas blancas","type":"Nutricion"},
{"id":"10","body":"Recuerda mantener porciones saludables a la hora de once, m\u00e1s a\u00fan los fin de semana! ","type":"Nutricion"},
{"id":"11","body":"Si seleccionas jugos de frutas envasados opta por los sin agregado de az\u00facar","type":"Nutricion"},{"id":"12","body":"Prepara tus
propios helados de paleta con l\u00e1cteos yogurt y pulpa de fruta","type":"Nutricion"},{"id":"13","body":"Prefiere las mantequillas bajas en grasa en vez
de las margarinas","type":"Nutricion"},{"id":"14","body":"Mayonesa alternativa; leche en polvo descremada aceite de oliva y un toque jugo
naranja","type":"Nutricion"},{"id":"15","body":"Sandwich saludable 2: jurel,lechuga y tomillo","type":"Nutricion"},{"id":"16","body":"Cocina al horno
y a la parrilla en vez de salteado o frito","type":"Nutricion"},{"id":"17","body":"Lee y compara las etiquetas de los alimentos y prefiere los que tengan
menos grasas, az\u00facar y sal (sodio)","type":"Nutricion"},{"id":"18","body":"Los \u00e1cidos grasos omega se obtienen del consumo de pescados y
aceites vegetales","type":"Nutricion"},{"id":"19","body":"Consume legumbres al menos dos veces por semana, sin mezclarlas con cecinas : Las
leguminosas se caracterizan por su alto aporte en prote\u00ednas, fibra y vitaminas. Y el consumo de estas en el pa\u00eds es ","type":"Nutricion"},
{"id":"20","body":"Bebe agua en vez de refrescos y bebidas de fantas\u00eda","type":"Nutricion"},{"id":"21","body":"La vitamina C mejora la
absorci\u00f3n del fierro de las carnes","type":"Nutricion"},{"id":"22","body":"Bailar cueca y otros como el \"trotecito\" y la \"resfalosa\" tambi\u00e9n
son ejercicios aer\u00f3bicos, as\u00ed es que bailando una hora se queman un chorip\u00e1n y un mote c\/huesillos chicos.","type":"Nutricion"},
{"id":"23","body":"El asado y longanizas, comerlos con ensaladas de verduras de hoja o chilena, no con papas ni pan.","type":"Nutricion"},
{"id":"24","body":"En 3 d\u00edas de comer en exceso, se pueden sumar 3 Kg de grasa al peso corporal.","type":"Nutricion"},
{"id":"25","body":"Recuerda: no cometer transgresiones alimentarias (comer y beber con avidez y en exceso y mucha variedad) y evitar comida callejera
por el riesgo de gastroenteritis.","type":"Nutricion"},{"id":"26","body":"Solo 1 copa de vino al almuerzo y en la cena, no ponches ni arreglao, por el
agregado de az\u00facar.","type":"Nutricion"},{"id":"27","body":"Alternativas y buenas ideas para reencontrarse con las legumbres, lee este art\u00edculo
en: http:\/\/goo.gl\/j39apk\r\n\"ELige vivir sano\"\r\n","type":"Nutricion"},{"id":"28","body":"El arroz el choclo y las pastas se comen con
```

Figure A.2: User Interface, Android client. The software gets a new message from the web server and notifies the User allowing him to rate the received message.



(a)

Latitud: -35.4033
Longitud: -71.634
Precision: 9
Distancia: 92.3992
Nombre zona: Establo
Ultima Alerta: Jul 31, 2014 17:53:29
Reactivación Alertas: ACTIVAS

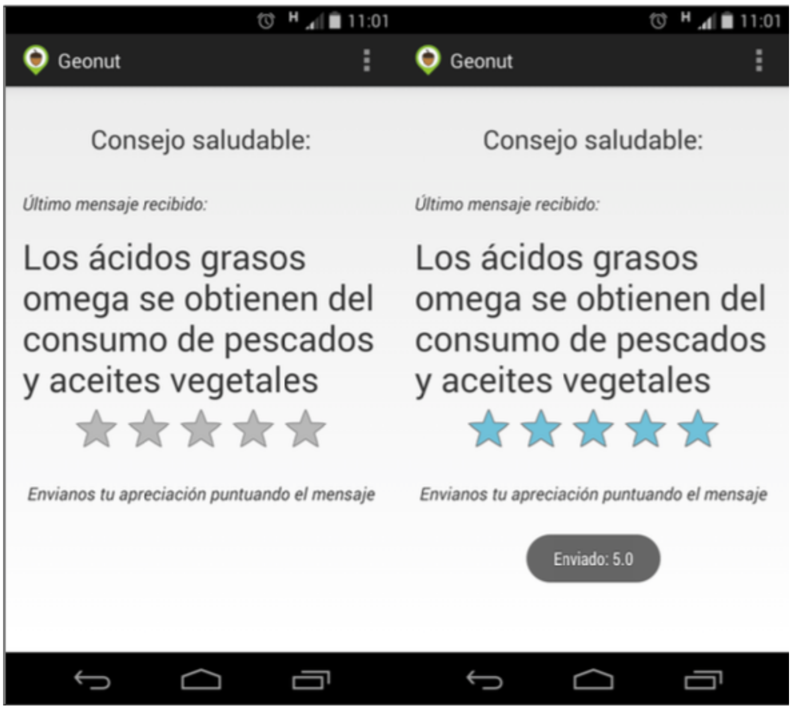
(b)

Latitud: -35.4039
Longitud: -71.6342
Precision: 45
Distancia: 37.393
Nombre zona: Establo
Ultima Alerta: Aug 01, 2014 11:00:50
Reactivación Alertas: 59 min

(c)

Notificación
¡Consejo saludable disponible!

Figure A.3: User Interface, Android client.



Consejo saludable:

Último mensaje recibido:

Los ácidos grasos omega se obtienen del consumo de pescados y aceites vegetales

☆☆☆☆☆

Envianos tu apreciación puntuando el mensaje

Consejo saludable:

Último mensaje recibido:

Los ácidos grasos omega se obtienen del consumo de pescados y aceites vegetales

★★★★★

Envianos tu apreciación puntuando el mensaje

Enviado: 5.0

Figure A.4: User Interface, Android client..

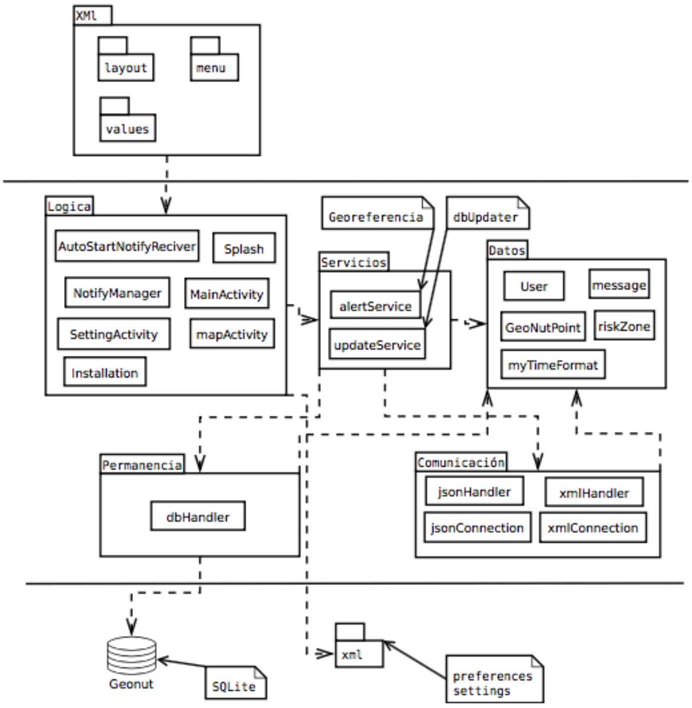


Figure A.5: architecture of the Geonut application

Ubicuidad - Georeferencia - Detección

Ubiapp es una aplicación para dispositivos móviles con Android, que promueve buenas prácticas frente a diversos comportamientos de riesgo, para ello utiliza diversas tecnologías para detectar dichos comportamientos

Descarga

http://goo.gl/RBxCa2

Intranet

Usuario:

Contraseña:

Apps

A través de URLpatrol (Browser y Detector de URL), APPpatrol (Detector de Aplicaciones) y GeoPatrol (Detector de lugares), se promueven buenas prácticas acorde a cada

Ubicuidad

Los teléfonos inteligentes (Android) nos permiten introducirnos en la vida de sus usuarios, Geonut utiliza la omnipresencia de esta tecnología para implementar un

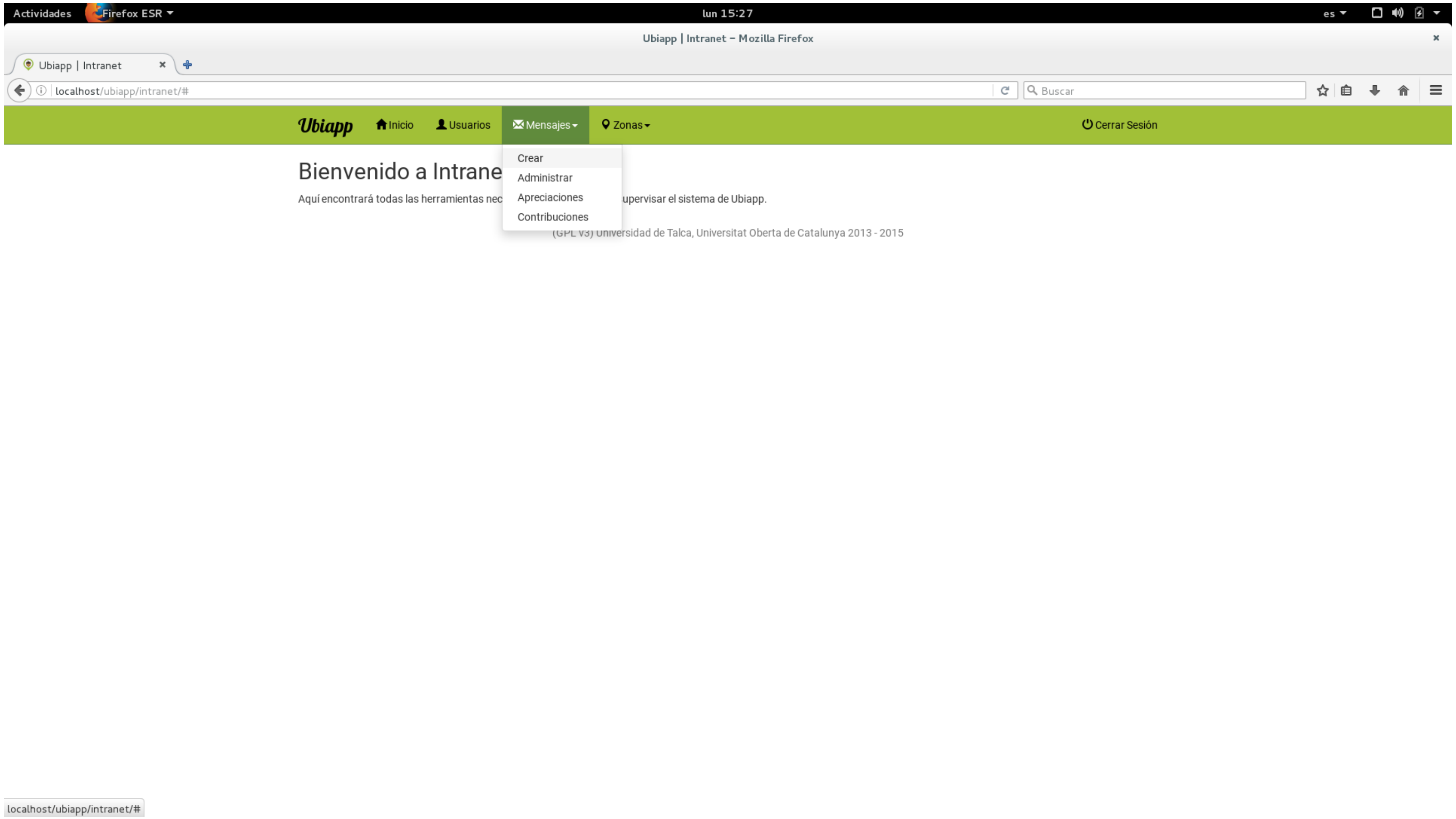
Figure A.6: Landing web interface of the UBESAFE system

Bienvenido a Intranet

Aqui encontrará todas las herramientas necesarias para gestionar y supervisar el sistema de Ubiapp.

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

Figure A.7: initial web interface of the UBESAFE system

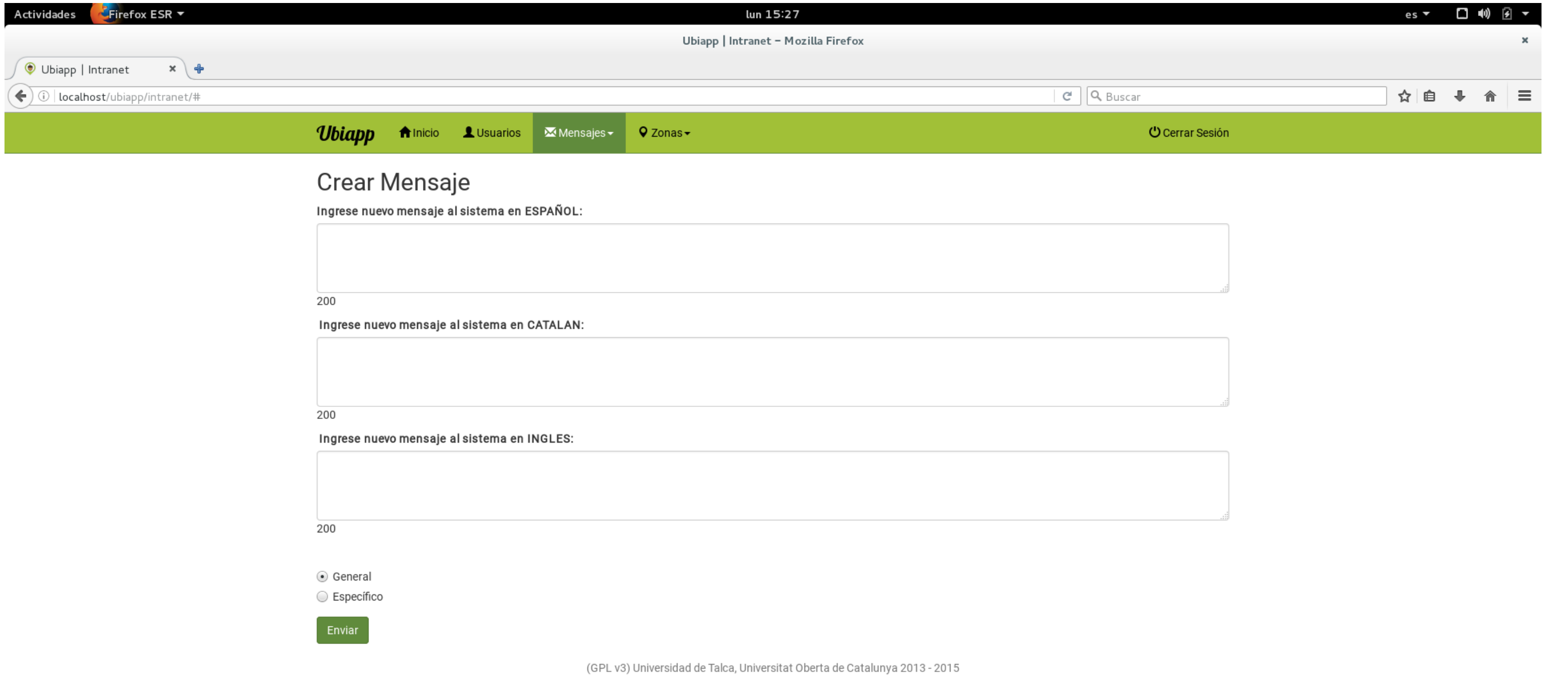


Bienvenido a Intrane

Aquí encontrará todas las herramientas nec upervisar el sistema de Ubiapp.

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

Figure A.8: Message menu web interface of the UBESAFE system



Crear Mensaje

Ingrese nuevo mensaje al sistema en ESPAÑOL:

200

Ingrese nuevo mensaje al sistema en CATALAN:

200

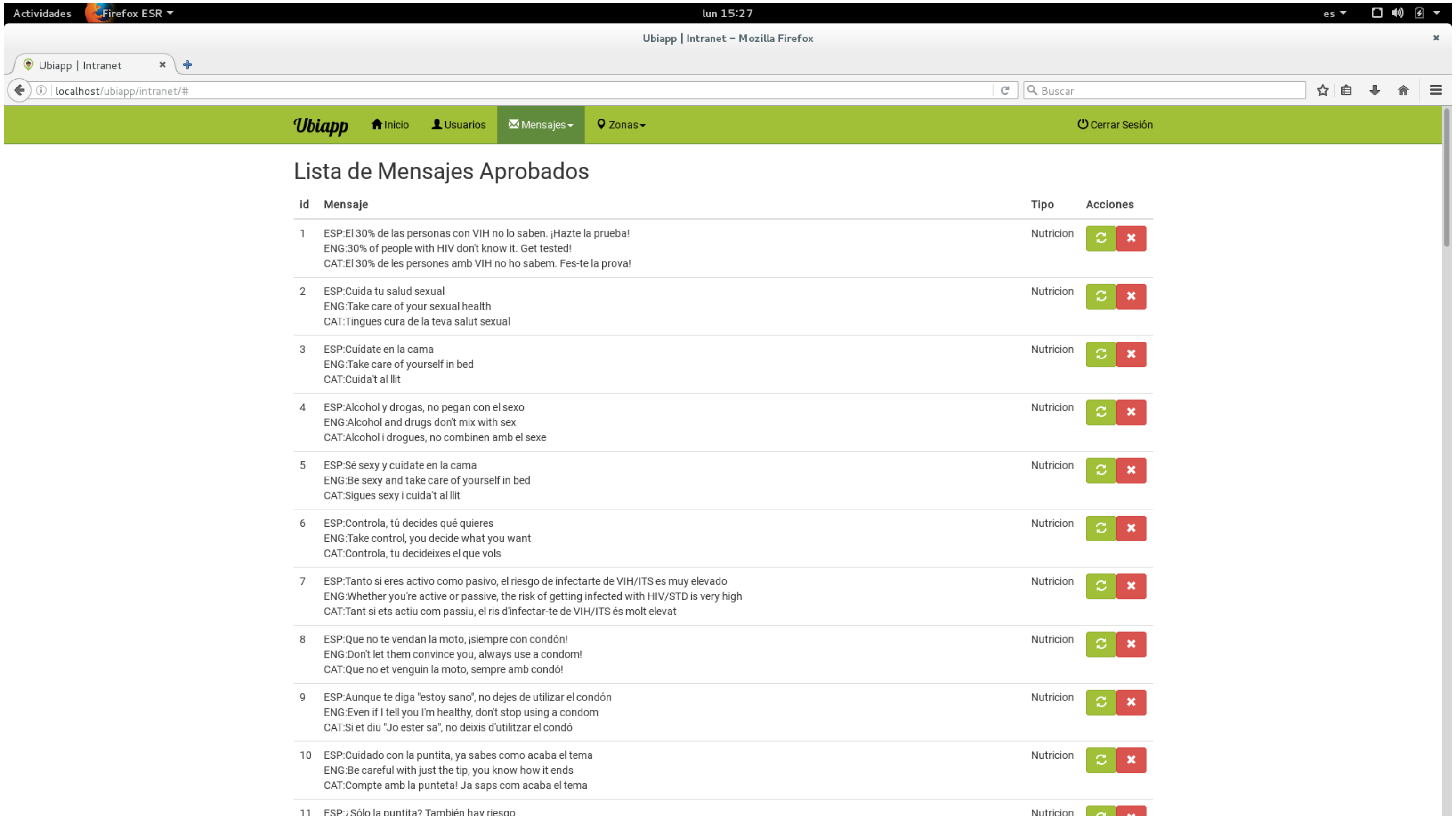
Ingrese nuevo mensaje al sistema en INGLES:

200

● General
○ Específico

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

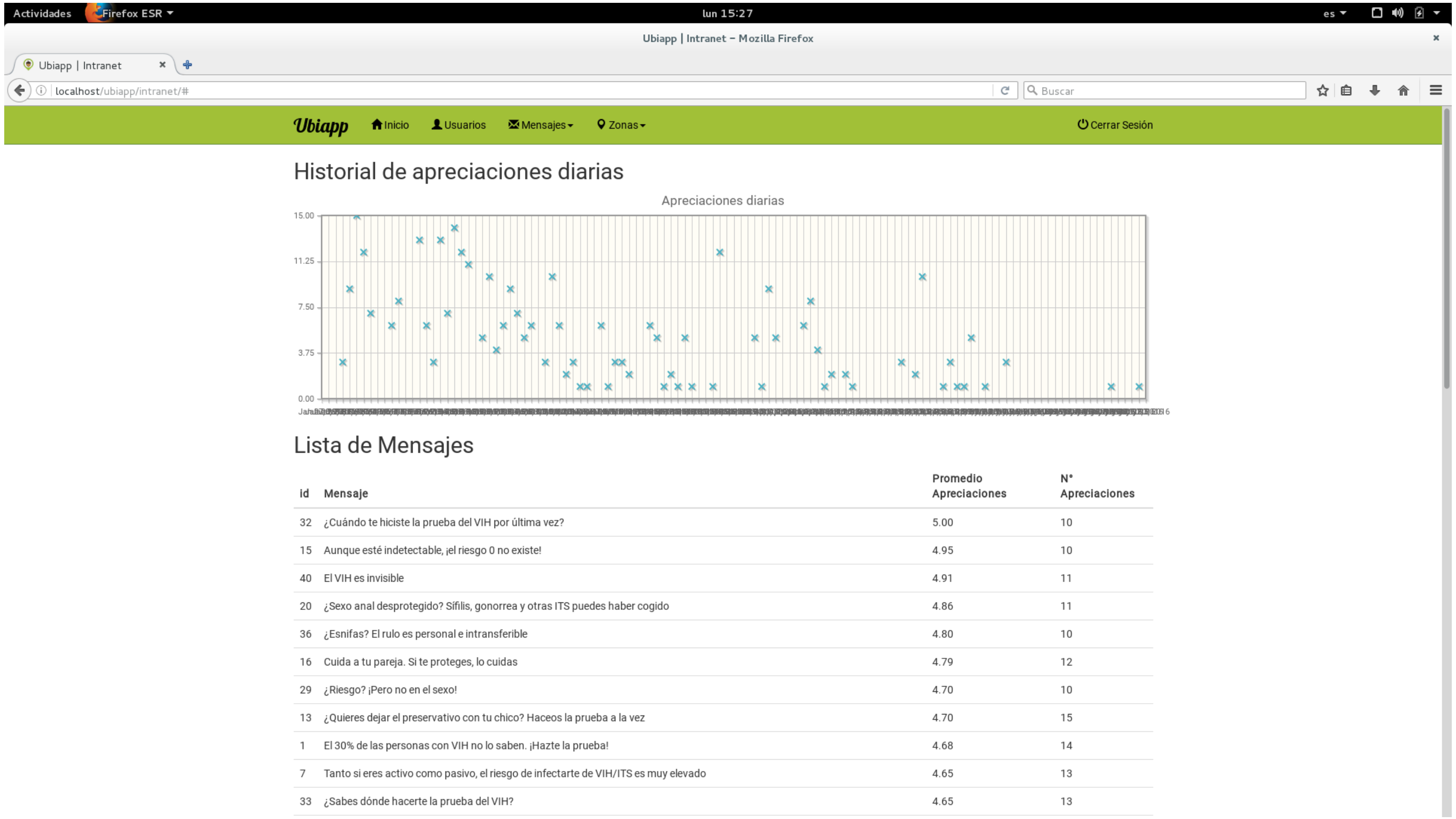
Figure A.9: Messages CRUD web interface of the UBESAFE system



Lista de Mensajes Aprobados

id	Mensaje	Tipo	Acciones
1	ESP:El 30% de las personas con VIH no lo saben. ¡Hazte la prueba! ENG:30% of people with HIV don't know it. Get tested! CAT:El 30% de les persones amb VIH no ho sabem. Fes-te la prova!	Nutricion	
2	ESP:Cuida tu salud sexual ENG:Take care of your sexual health CAT:Tingues cura de la teva salut sexual	Nutricion	
3	ESP:Cuídate en la cama ENG:Take care of yourself in bed CAT:Cuida't al llit	Nutricion	
4	ESP:Alcohol y drogas, no pegan con el sexo ENG:Alcohol and drugs don't mix with sex CAT:Alcohol i drogues, no combinen amb el sexe	Nutricion	
5	ESP:Sé sexy y cuídate en la cama ENG:Be sexy and take care of yourself in bed CAT:Sigues sexy i cuida't al llit	Nutricion	
6	ESP:Controla, tú decides qué quieres ENG:Take control, you decide what you want CAT:Controla, tu decideixes el que vols	Nutricion	
7	ESP:Tanto si eres activo como pasivo, el riesgo de infectarte de VIH/ITS es muy elevado ENG:Whether you're active or passive, the risk of getting infected with HIV/STD is very high CAT:Tant si ets actiu com passiu, el ris d'infectar-te de VIH/ITS és molt elevat	Nutricion	
8	ESP:Que no te vendan la moto, ¡siempre con condón! ENG:Don't let them convince you, always use a condom! CAT:Que no et venguin la moto, sempre amb condó!	Nutricion	
9	ESP:Aunque te diga "estoy sano", no dejes de utilizar el condón ENG:Even if I tell you I'm healthy, don't stop using a condom CAT:Si et diu "Jo ester sa", no deixis d'utilitzar el condó	Nutricion	
10	ESP:Cuidado con la puntita, ya sabes como acaba el tema ENG:Be careful with just the tip, you know how it ends CAT:Compte amb la punteta! Ja saps com acaba el tema	Nutricion	
11	ESP:¿Sólo la puntita? También hay riesgo	Nutricion	

Figure A.10: Message managment web interface of the UBESAFE system

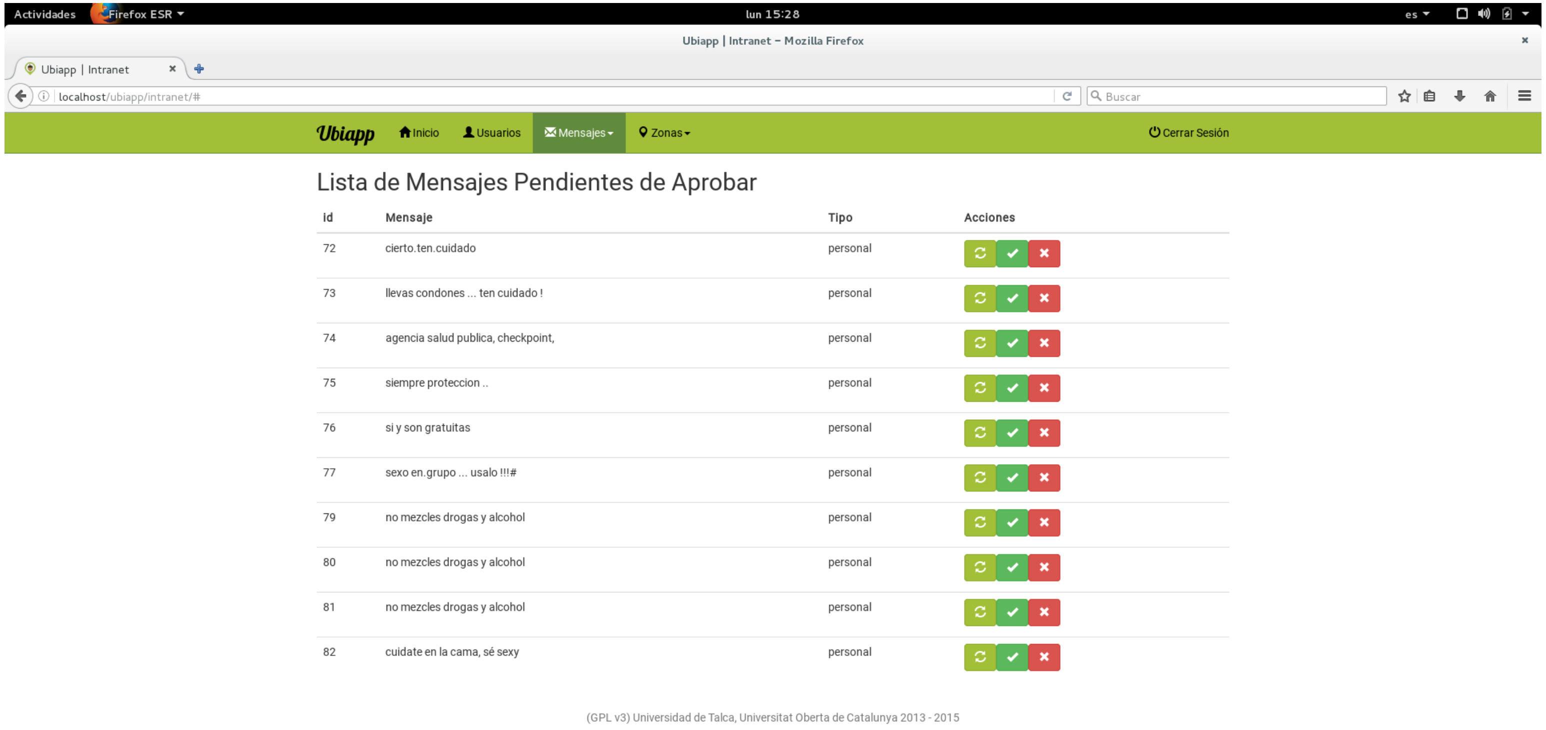


Historial de apreciaciones diarias

Lista de Mensajes

id	Mensaje	Promedio Apreciaciones	N° Apreciaciones
32	¿Cuándo te hiciste la prueba del VIH por última vez?	5.00	10
15	Aunque esté indetectable, ¡el riesgo 0 no existe!	4.95	10
40	El VIH es invisible	4.91	11
20	¿Sexo anal desprotegido? Sífilis, gonorrea y otras ITS puedes haber cogido	4.86	11
36	¿Esnifas? El rulo es personal e intransferible	4.80	10
16	Cuida a tu pareja. Si te proteges, lo cuidas	4.79	12
29	¿Riesgo? ¡Pero no en el sexo!	4.70	10
13	¿Quieres dejar el preservativo con tu chico? Haceos la prueba a la vez	4.70	15
1	El 30% de las personas con VIH no lo saben. ¡Hazte la prueba!	4.68	14
7	Tanto si eres activo como pasivo, el riesgo de infectarte de VIH/ITS es muy elevado	4.65	13
33	¿Sabes dónde hacerte la prueba del VIH?	4.65	13

Figure A.11: Message scoring web interface of the UBESAFE system



Lista de Mensajes Pendientes de Aprobar

id	Mensaje	Tipo	Acciones
72	cierto.ten.cuidado	personal	
73	llevas condones ... ten cuidado !	personal	
74	agencia salud publica, checkpoint,	personal	
75	siempre proteccion ..	personal	
76	si y son gratuitas	personal	
77	sexo en.grupo ... usalo !!!#	personal	
79	no mezcles drogas y alcohol	personal	
80	no mezcles drogas y alcohol	personal	
81	no mezcles drogas y alcohol	personal	
82	cuidate en la cama, sé sexy	personal	

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

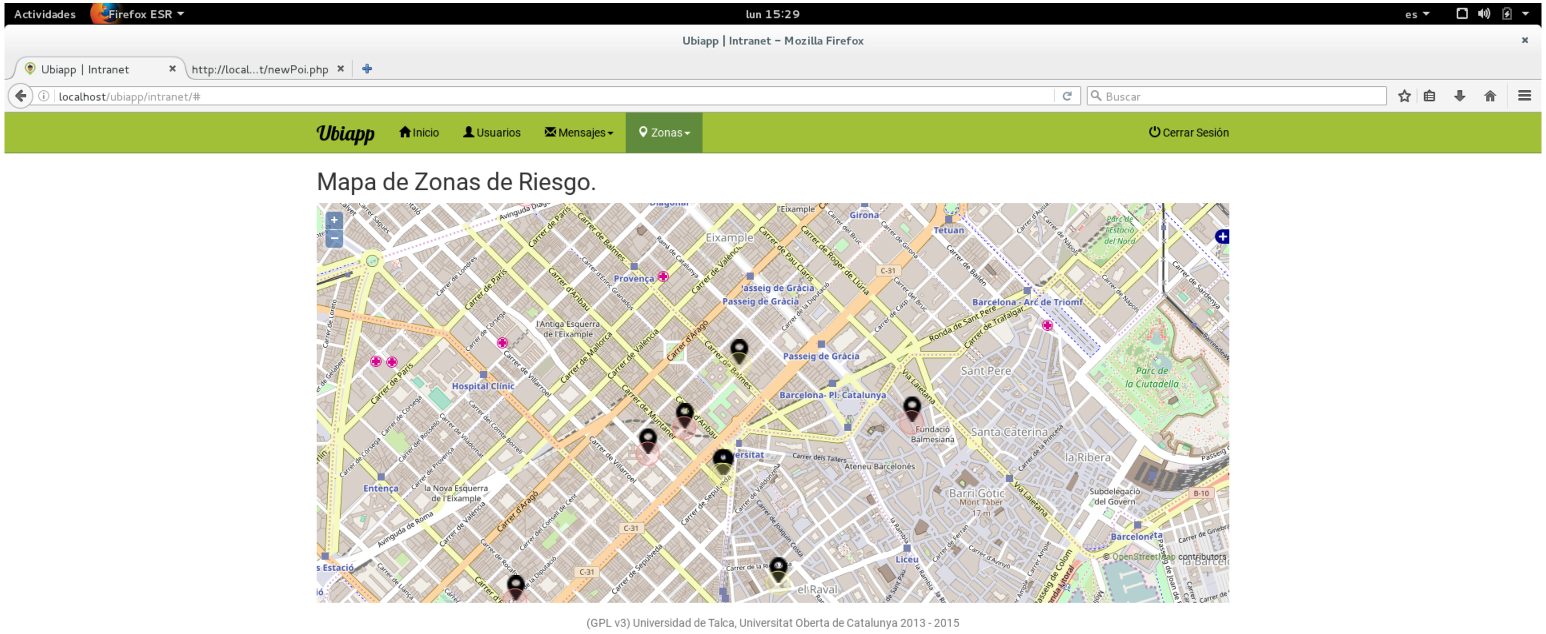
Figure A.12: Message pending to approval web interface of the UBESAFE system

Bienvenido a Intranet

Aquí encontrará todas las herramientas necesarias para gestic … a de Ubiapp.

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

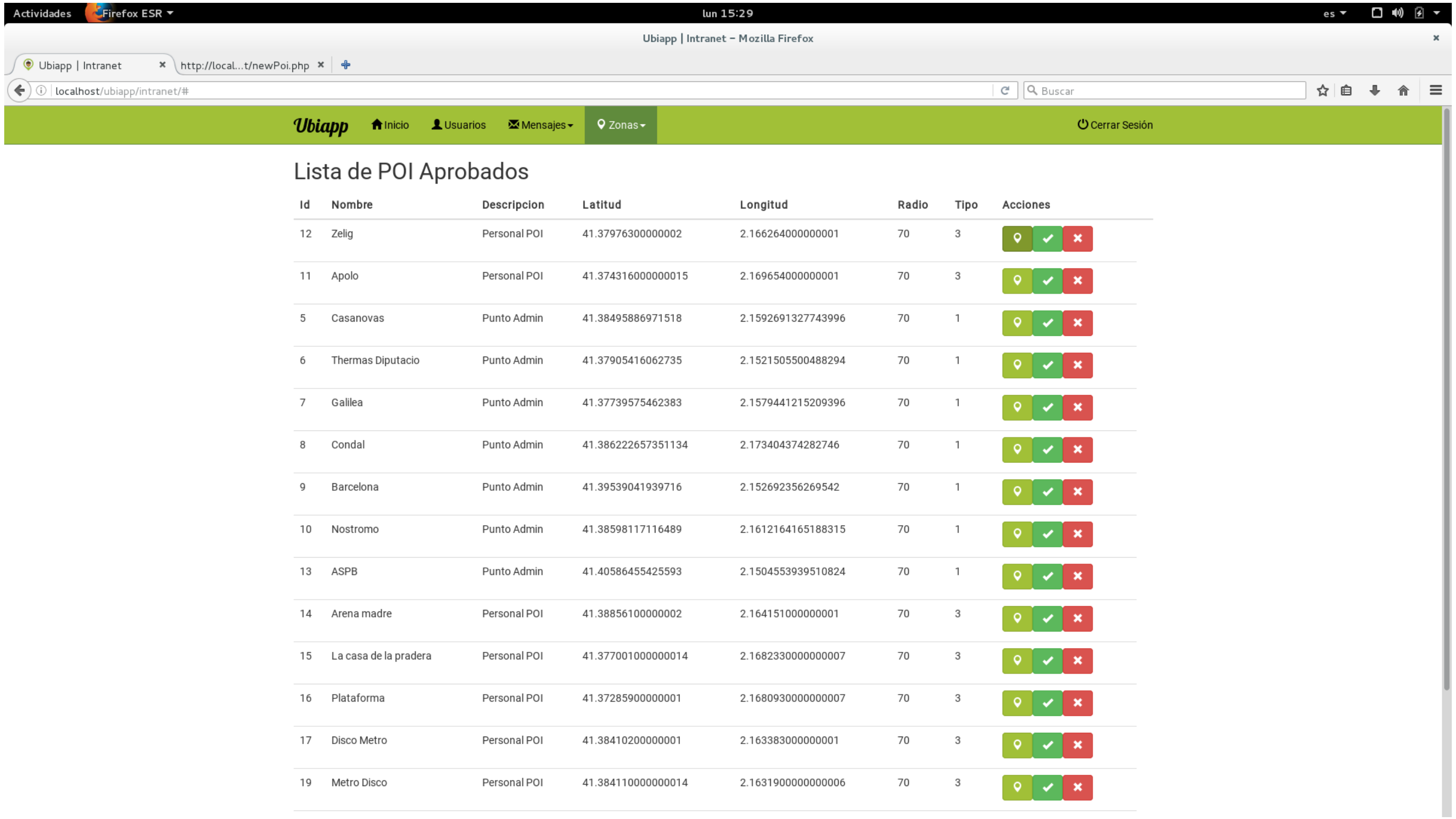
Figure A.13: Menu POI web interface of the UBESAFE system



Mapa de Zonas de Riesgo.

(GPL v3) Universidad de Talca, Universitat Oberta de Catalunya 2013 - 2015

Figure A.14: Pois mapview web interface of the UBESAFE system



Lista de POI Aprobados

Id	Nombre	Descripcion	Latitud	Longitud	Radio	Tipo	Acciones
12	Zelig	Personal POI	41.37976300000002	2.166264000000001	70	3	📍 ✓ ✗
11	Apolo	Personal POI	41.374316000000015	2.169654000000001	70	3	📍 ✓ ✗
5	Casanovas	Punto Admin	41.38495886971518	2.1592691327743996	70	1	📍 ✓ ✗
6	Thermas Diputacio	Punto Admin	41.37905416062735	2.1521505500488294	70	1	📍 ✓ ✗
7	Galilea	Punto Admin	41.37739575462383	2.1579441215209396	70	1	📍 ✓ ✗
8	Condal	Punto Admin	41.386222657351134	2.173404374282746	70	1	📍 ✓ ✗
9	Barcelona	Punto Admin	41.39539041939716	2.152692356269542	70	1	📍 ✓ ✗
10	Nostromo	Punto Admin	41.38598117116489	2.1612164165188315	70	1	📍 ✓ ✗
13	ASPB	Punto Admin	41.40586455425593	2.1504553939510824	70	1	📍 ✓ ✗
14	Arena madre	Personal POI	41.38856100000002	2.164151000000001	70	3	📍 ✓ ✗
15	La casa de la pradera	Personal POI	41.377001000000014	2.1682330000000007	70	3	📍 ✓ ✗
16	Plataforma	Personal POI	41.37285900000001	2.1680930000000007	70	3	📍 ✓ ✗
17	Disco Metro	Personal POI	41.38410200000001	2.163383000000001	70	3	📍 ✓ ✗
19	Metro Disco	Personal POI	41.384110000000014	2.1631900000000006	70	3	📍 ✓ ✗

Figure A.15: POI approval web interface of the UBESAFE system

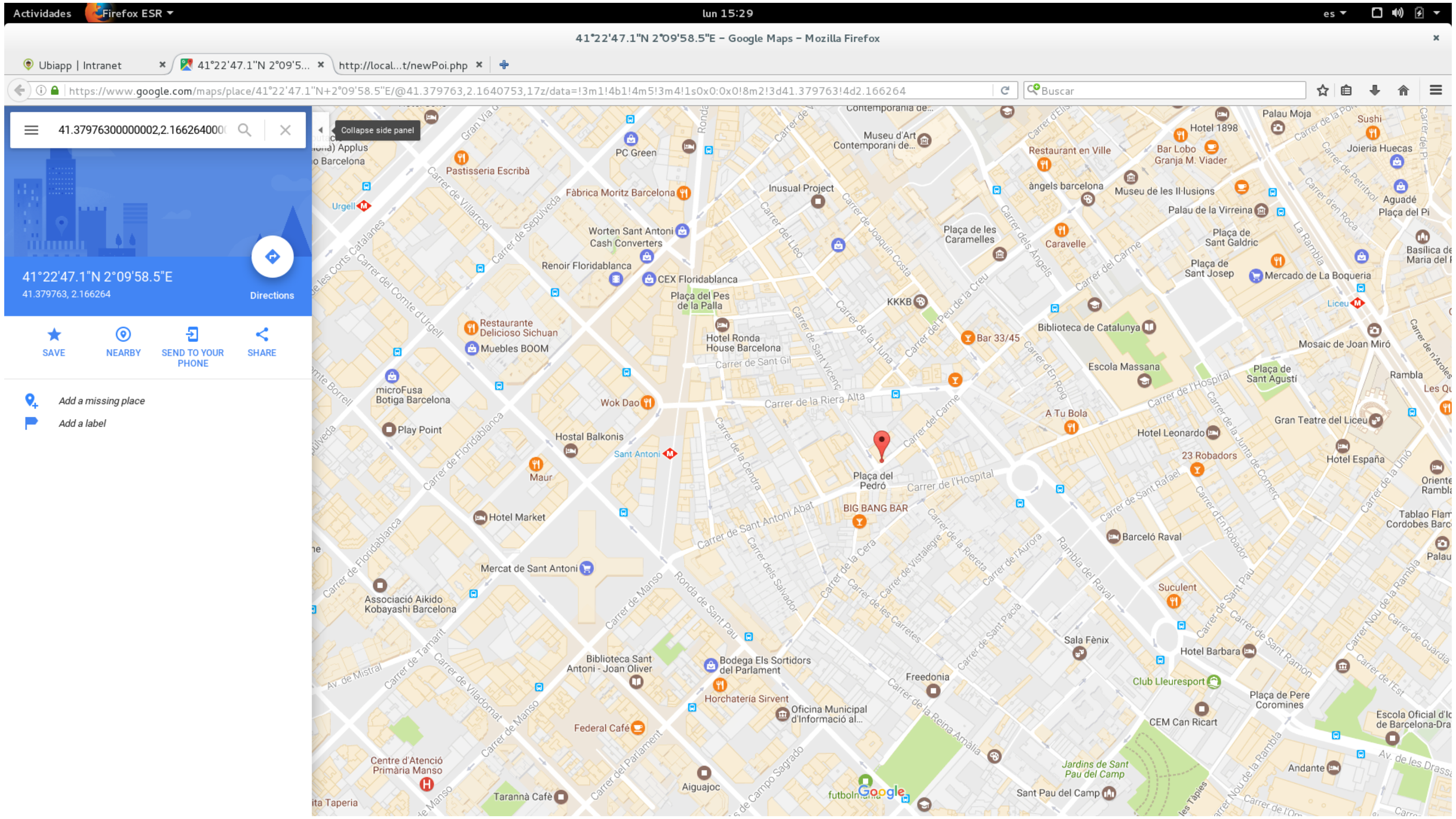


Figure A.16: Mapview of POI web interface of the UBESAFE system

Appendix B

Snippets

Listing B.1: Alert service: setting the alarm for X intervals

```
private void requestUpdates() {

 locationManager = (LocationManager) ctx.getSystemService(Context.LOCATION_SERVICE);

 if (locationManager.isProviderEnabled(LocationManager.NETWORK_PROVIDER) ||
     locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
   new Thread(new Runnable() {
     public void run() {
       try {
         Looper.prepare();
         final Timer t = new Timer();
         final LocationListener locationListener = new LocationListener() {
           private  Location loc = null;
           public void onLocationChanged(Location location) {

             if(isBetterLocation(location,loc)){

               Log.i(MainActivity.TAGAPP,"␣MEJOR␣LECTURA!!!!");
               loc = location;
               bestLocation = location;

              }

              Log.i("alertService","Provider:" + location.getProvider());
              Log.i("alertService","Latitude:" + location.getLatitude());
              Log.i("alertService","Longitude:"+ location.getLongitude());
              Log.i("alertService", "Accuracy:" + location.getAccuracy());

              increaseAccuracyIntent();
            }
          };

         //Realizamos request para los dos providers
         if(locationManager.isProviderEnabled(LocationManager.NETWORK_PROVIDER))
           locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER,
           0, 0, locationListener, Looper.myLooper());
         if(locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER))
           locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,100,
           0,locationListener,Looper.myLooper());

          timeoutLocationManager(locationManager, locationListener,
          Looper.myLooper(), t);
          Looper.loop();

        } catch (Exception e) {
            e.printStackTrace();
        }
      }
    }
     ).start();
           }
       }
```

Bibliography

[1] N G Schiller, S Crystal, and D Lewellen. Risky business: the cultural construction of AIDS risk groups. *Social Science and Medicine*, 38:1337–1346, 1994. URL `http://www.ncbi.nlm.nih.gov/pubmed/8023185`.

[2] Teresa J Finlayson, Binh Le, Amanda Smith, Kristina Bowles, Melissa Cribbin, Isa Miles, Alexandra M Oster, Tricia Martin, Alicia Edwards, and Elizabeth Dinenno. HIV risk, prevention, and testing behaviors among men who have sex with men–National HIV Behavioral Surveillance System, 21 U.S. cities, United States, 2008. *Morbidity and mortality weekly report. Surveillance summaries (Washington, D.C. : 2002)*, 60:1–34, 2011. ISSN 1545-8636. doi: ss6014a1[pii]. URL `http://www.ncbi.nlm.nih.gov/pubmed/22031280`.

[3] Yan Song, Xiaoming Li, Liying Zhang, Xiaoyi Fang, Xiuyun Lin, Yinjie Liu, and Bonita Stanton. HIV-testing behavior among young migrant men who have sex with men (MSM) in Beijing, China. *AIDS care*, 23:179–186, 2011. ISSN 0954-0121. doi: 10.1080/09540121.2010.487088.

[4] Marta C González, César A Hidalgo, and Albert-László Barabási. Understanding individual human mobility patterns. *Nature*, 453:779–782, 2008. ISSN 0028-0836. doi: 10.1038/nature06958.

[5] Cornelis a Rietmeijer and Mary McFarlane. Web 2.0 and beyond: risks for sexually transmitted infections and opportunities for prevention. *Current opinion in infectious diseases*, 22(1):67–71, February 2009. ISSN 1473-6527. doi: 10.1097/QCO.0b013e328320a871. URL `http://www.ncbi.nlm.nih.gov/pubmed/19532082`.

[6] Deb Levine. Using technology, new media, and mobile for sexual and reproductive health. *Sexuality Research and Social Policy*, 8:18–26, 2011. ISSN 18689884. doi: 10.1007/s13178-011-0040-7.

[7] Andrew D Margolis, Heather Joseph, Lisa Belcher, Sabina Hirshfield, and Mary Ann Chiasson. 'Never testing for HIV' among men who have sex with men recruited from a sexual networking website, United States. *AIDS and behavior*, 16(1):23–9, January 2012. ISSN 1573-3254. doi: 10.1007/s10461-011-9883-4. URL `http://www.ncbi.nlm.nih.gov/pubmed/21279431`.

[8] World Health Organization. *Obesity and overweight. Who fact sheet N311.* actualizado en marzo 2013. Web 27 abril 2013 http://www.who.int/mediacentre/factsheets/fs311/en/, 2013.

[9] Teri L Burgess-Champoux. Innovative use of technology in nutrition education research. *Journal of nutrition education and behavior*, 45(1):1, 2013. ISSN 1878-2620. doi: 10.1016/j.jneb.2012.11.003. URL `http://www.ncbi.nlm.nih.gov/pubmed/23305802`.

[10] Melanie Hingle, Mimi Nichter, Melanie Medeiros, and Samantha Grace. Texting for Health: The Use of Participatory Methods to Develop Healthy Lifestyle Messages for Teens. *Journal of Nutrition Education and Behavior*, 45:12–19, 2013. ISSN 14994046. doi: 10.1016/j.jneb.2012.05.001.

[11] Debra K. Sullivan, Jeannine R. Goetz, Cheryl A. Gibson, Richard A. Washburn, Bryan K. Smith, Jaehoon Lee, Stephanie Gerald, Tennille Fincham, and Joseph E. Donnelly. Improving weight maintenance using virtual reality (second life). *Journal of Nutrition Education and Behavior*, 45:264–268, 2013. ISSN 14994046. doi: 10.1016/j.jneb.2012.10.007.

[12] Clare Herrick. Risky bodies: Public health, social marketing and the governance of obesity. *Geoforum*, 38:90–102, 2007. ISSN 00167185. doi: 10.1016/j.geoforum.2006.06.003.

[13] A Petersen. Risk, governance and the new public health. *Foucault*, 1997. URL `http://books.google.com/books?hl=en&lr=&id=XFwHH8zaIx0C&oi=fnd&pg=PA189&dq=governance+health+system&ots=BIA-NsroMC&sig=f6hS-LS5yGkKnvw7PTxzNgb_0FQ`.

[14] Barbara Lohse. Facebook is an effective strategy to recruit low-income women to online nutrition education. *Journal of nutrition education and behavior*, 45 (1):69–76, 2013. ISSN 1878-2620. doi: 10.1016/j.jneb.2012.06.006. URL `http://www.ncbi.nlm.nih.gov/pubmed/23305805`.

[15] Mosa. *A systematic Review of healthcare Applications for smartphones.* BMC Medical Informatics and Decision Making, 12:67, 2012.

[16] Craiger Weiss. *Ubiquitius Computing.* The industrial-Organizational Psycologist. 39 Number 4, 2002.

[17] Genevieve Bell and Paul Dourish. Yesterdays tomorrows: notes on ubiquitous computings dominant vision. *Personal and Ubiquitous Computing*, 11(2):133–143, November 2006. ISSN 1617-4909. doi: 10.1007/s00779-006-0071-x. URL `http://link.springer.com/10.1007/s00779-006-0071-x`.

[18] Warren a Kaplan. Can the ubiquitous power of mobile phones be used to improve health outcomes in developing countries? *Globalization and health*, 2:9, January 2006. ISSN 1744-8603. doi: 10.1186/1744-8603-2-9. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1524730&tool=pmcentrez&rendertype=abstract`.

[19] Victoria J Rideout, Donald F. Roberts, and Ulla G. Foehr. Generation M: Media in the lives of 8-18 year olds. Technical report, 2005. URL `http://www.kff.org/entmedia/upload/8010.pdf$\delimiter"026E30F$nhttp://www.kff.org/entmedia/upload/Generation-M-Media-in-the-Lives-of-8-18-Year-olds-Report.pdf$\delimiter"026E30F$nL:$\delimiter"026E30F$_Synch_`

folder$\delimiter"026E30F$ReferenceManagerData$\delimiter"026E30F$Readings$\delimiter"026E30F$rideout_roberts_foehr2005.pdf.

[20] Richard Shim. Tablets Impact the Notebook Market: Enter the Ultrabook. *Information Display*, 28:12–14, 2012.

[21] Chris Martin. 73 million tablets were shifted in 2011, 2012. URL http://www.theinquirer.net/inquirer/news/2135275/million-tablets-shifted-2011.

[22] Sheana S Bull, Lindsey T Breslin, Erin E Wright, Sandra R Black, Deborah Levine, and John S Santelli. Case study: An ethics case study of HIV prevention research on Facebook: the Just/Us study. *Journal of pediatric psychology*, 36(10):1082–92, 2011. ISSN 1465-735X. doi: 10.1093/jpepsy/jsq126. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3199441&tool=pmcentrez&rendertype=abstract.

[23] Kathryn E Muessig, Emily C Pike, Sara Legrand, and Lisa B Hightow-Weidman. Mobile phone applications for the care and prevention of HIV and other sexually transmitted diseases: a review. *Journal of medical Internet research*, 15:e1, 2013. ISSN 1438-8871. doi: 10.2196/jmir.2301. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3636069&tool=pmcentrez&rendertype=abstract.

[24] World Health Organization. Obesity: preventing and managing the global epidemic, Report of a WHO Consultation . http://apps.who.int/iris/bitstream/10665/42330/1/WHO_TRS_894.pdf?ua=1&ua=1, 2000 Geneva, Switzerland. World Health Organization (WHO technical report series 894).

[25] Benjamin Caballero. The global epidemic of obesity: An overview. *Epidemiologic Reviews*, 29(1):1–5, 2007. doi: 10.1093/epirev/mxm012. URL http://dx.doi.org/10.1093/epirev/mxm012.

[26] James O. Hill, Helen Thompson, and Holly Wyatt. Weight maintenance: What's missing?, 2005. ISSN 00028223.

[27] Nancy L. Cohen, Elena T. Carbone, and Patricia A. Beffa-Negrini. The Design, Implementation, and Evaluation of Online Credit Nutrition Courses: A Systematic Review. *Journal of Nutrition Education and Behavior*, 43:76–86, 2011. ISSN 14994046. doi: 10.1016/j.jneb.2010.04.001.

[28] Jean Harvey-Berino, Lizzy Pope, Beth Casey Gold, Heather Leonard, and Cynthia Belliveau. Undergrad and Overweight: An Online Behavioral Weight Management Program for College Students. *Journal of Nutrition Education and Behavior*, 44:604–608, 2012. ISSN 14994046. doi: 10.1016/j.jneb.2012.04.016.

[29] Maeve Duggan and Joanna Brenner. The Demographics of Social Media Users-2012. Technical report, 2013. URL `http://pewinternet.org/Reports/2013/Social-media-users.aspxFOR`.

[30] Mary Madden. Older Adults and Social Media. Technical report, 2010.

[31] Amanda Lenhart, Richard Ling, Scott Campbell, and Kristen Purcell. Teens and Mobile Phones. Technical report, 2010.

[32] Joel J P C Rodrigues, Ivo M C Lopes, Bruno M C Silva, and Isabel de La Torre. A new mobile ubiquitous computing application to control obesity: SapoFit. *Informatics for Health and Social Care*, 38(1):37–53, 2013. doi: 10.3109/17538157.2012.674586. URL `http://dx.doi.org/10.3109/17538157.2012.674586`.

[33] Wendy Hsin-yuan Huang and Dilip Soman. Gamification Of Education. *University of Toronto - Rotman School of Management*, pages 1–29, 2013. ISSN 09594752. doi: 10.1111/j.1467-8535.2011.01259.x.

[34] Sebastian Deterding, Rilla Khaled, Lennart Nacke, and Dan Dixon. Gamification: Toward a definition. In *CHI 2011 Gamification Workshop Proceedings*, Vancouver, BC, Canada, 2011.

[35] Oriol Borrás Gené. Fundamentos de gamificación. Mayo 2015. URL `http://oa.upm.es/35517/`.

[36] Flavio Escribano. Gamificación versus Ludictadura. *Obra digital*, pages 58–72, 2013. ISSN 2014-5039. URL `http://revistesdigitals.uvic.cat/index.php/obradigital/article/view/22`.

[37] UIT. *Medicin de la Sociedad de la Informacin*. U. I. de Telecomunicaciones, 2010.

[38] Cristina Gallego Gómez, Carmen De Pablos Heredero, Cristina Gallego Gomez, and Carmen de Pablos Heredero. The gamification and the enrichment of innovation practices in the firm: an analysis of experiences. *Intangible Capital*, 9(3):800–822, 2013. ISSN 1697-9818. doi: 10.3926/ic.377. URL `http://www.intangiblecapital.org/index.php/ic/article/view/377`.

[39] Cameron Lister, Joshua H West, Ben Cannon, Tyler Sax, David Brodegard, and Cameron Lister. Just a Fad ? Gamification in Health and Fitness Apps Corresponding Author :. 2:1–12, 2014. doi: 10.2196/games.3413.

[40] F. Martin Mendiola, Miriam Kalnicki, and Sarah Lindenauer. Valuable features in mobile health apps for patients and consumers: Content analysis of apps and user ratings. *JMIR mHealth uHealth*, 3(2):e40, May 2015. doi: 10.2196/mhealth.4283. URL `http://mhealth.jmir.org/2015/2/e40/`.

[41] Maged N Kamel Boulos and Stephen P Yang. Exergames for health and fitness: the roles of GPS and geosocial apps. *International Journal of Health Geographics 2013 12:1*, 12(1):295–305, 2013. ISSN 10730516 (ISSN). doi: 10.1186/1476-072X-12-18.

[42] Cezar Giosan, Cristina Mogoae, Oana Cobeanu, Aurora Szentágotai Ttar, Vlad Murean, and Rare Boian. Using a smartphone app to reduce cognitive vulnerability and mild depressive symptoms: Study protocol of an exploratory randomized controlled trial. *Trials*, 17(1):609, 2016. ISSN 1745-6215. doi: 10.

1186/s13063-016-1740-3. URL `http://trialsjournal.biomedcentral.com/articles/10.1186/s13063-016-1740-3`.

[43] Margarita Serra Alias. *Educacion para la salud y Educacion nutricional*. Universitat Oberta de Catalunya, PID_00185221.

[44] Superbetter App. `https://play.google.com/store/apps/details?id=com.superbetter.paid&hl=en`, 2017. [Online; accessed 1-November-2017].

[45] Inc. HopeLab. Re mission App. `http://www.re-mission2.org`, 2017. [Online; accessed 1-November-2017].

[46] Philipp Brauner, André Calero Valdez, Ulrik Schroeder, and Martina Ziefle. Increase physical fitness and create health awareness through exergames and gamification: The role of individual factors, motivation and acceptance. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7946 LNCS:349–362, 2013. ISSN 03029743. doi: 10.1007/978-3-642-39062-3_22.

[47] Kyriaki G Giota and George Kleftaras. Mental Health Apps : Innovations , Risks and Ethical Considerations. (September):19–23, 2014.

[48] Fiona Y. Wong. Influence of pokémon go on physical activity levels of university players: a cross-sectional study. *Int J Health Geogr*, 16:8, Feb 2017. ISSN 1476-072X. doi: 10.1186/s12942-017-0080-1. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC5322678/`. 80[PII].

[49] Laura Dennison, Leanne Morrison, Gemma Conway, and Lucy Yardley. Opportunities and challenges for smartphone applications in supporting health behavior change: Qualitative study. *J Med Internet Res*, 15(4):e86, Apr 2013. doi: 10.2196/jmir.2583. URL `http://www.jmir.org/2013/4/e86/`.

[50] Lana Hebden, Amelia Cook, Hidde P van der Ploeg, and Margaret Allman-Farinelli. Development of smartphone applications for nutrition and physical

activity behavior change. *JMIR Research Protocols*, 1(2):e9, Jul-Dec 2012. doi: 10.2196/resprot.2205. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3626164/`.

[51] et al White, B.K. Theory-Based Design and Development of a Socially Connected, Gamified Mobile App for Men About Breastfeeding (Milk Man). *Journal of Medical Internet Research*, Vol. 4(No. 2):1–36, 2016. ISSN 2291-5222. doi: 10.2196/mhealth.5652.

[52] Milk Man App. `http://pifistudy.net.au/milkman/`, 2017. [Online; accessed 1-November-2017].

[53] E. Marisa Hilliard, Amy Hahn, K. Alana Ridge, N. Michelle Eakin, and A. Kristin Riekert. User preferences and design recommendations for an mhealth app to promote cystic fibrosis self-management. *JMIR mHealth uHealth*, 2(4):e44, Oct 2014. doi: 10.2196/mhealth.3599. URL `http://mhealth.jmir.org/2014/4/e44/`.

[54] Maritza GarcÃa O and Andrea Olea N. Evolución y situación epidemiológica de la infección por virus de inmunodeficiencia humana y síndrome de inmunodeficiencia adquirida en Chile. *Revista chilena de infectologÃa*, 25:162 – 170, 06 2008. ISSN 0716-1018. URL `http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0716-10182008000300003&nrm=iso`.

[55] Sara LeGrand, Elizabeth Kathryn Muessig, Tobias McNulty, Karina Soni, Kelly Knudtson, Alex Lemann, Nkechinyere Nwoko, and B. Lisa Hightow-Weidman. Epic allies: Development of a gaming app to improve antiretroviral therapy adherence among young hiv-positive men who have sex with men. *JMIR Serious Games*, 4(1):e6, May 2016. doi: 10.2196/games.5687. URL `http://games.jmir.org/2016/1/e6/`.

[56] Gabe Zichermann and Christopher Cunningham. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps.* O'Reilly Media, Inc., 1st edition, 2011. ISBN 1449397670, 9781449397678.

[57] Dallas Swendeman and Mary Jane Rotheram-Borus. Innovation in sexually transmitted disease and HIV prevention: internet and mobile phone delivery vehicles for global diffusion. *Current opinion in psychiatry*, 23:139–144, 2010. ISSN 0951-7367. doi: 10.1097/YCO.0b013e328336656a.

[58] Claudio Cioffi-Revilla. Computational social science, 2010. ISSN 19395108.

[59] Craig Larman. *UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. Segunda edición.* Prentice Hall, 2003.

[60] Open Source Initative. `https://opensource.org/`, 2017. [Online; accessed 1-June-2017].

[61] Free Software Foundation. `http://www.fsf.org/`, 2017. [Online; accessed 1-June-2017].

[62] PHP general-purpose scripting language for web development. `http://php.net/`, 2017. [Online; accessed 1-June-2017].

[63] Open Street Maps. `https://www.openstreetmap.org/`, 2017. [Online; accessed 1-June-2017].

[64] OpenLayers. `https://openlayers.org/`, 2017. [Online; accessed 1-June-2017].

[65] Open Street Maps Android. `https://osmdroid.net/`, 2017. [Online; accessed 1-June-2017].

[66] Android OS. `https://developer.android.com/`, 2017. [Online; accessed 1-June-2017].

[67] Vicki W. Chanon, Chandler R. Sours, and Charlotte A. Boettiger. Attentional bias toward cigarette cues in active smokers. *Psychopharmacology (Berl)*, 212(3):309–320, Oct 2010. ISSN 0033-3158. doi: 10.1007/s00213-010-1953-1. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2967198/`. 20668841[pmid].

[68] Ronald N. Ehrman, Steven J. Robbins, Melissa A. Bromwell, Megan E. Lankford, John R. Monterosso, and Charles P. O'Brien. Comparing attentional bias to smoking cues in current smokers, former smokers, and non-smokers using a dot-probe task. *Drug & Alcohol Dependence*, 67(2):185–191, 2017/09/01 XXXX. ISSN 0376-8716. doi: 10.1016/S0376-8716(02)00065-0. URL `http://dx.doi.org/10.1016/S0376-8716(02)00065-0`.

[69] BP Bradley, M. Field, H. Healy, and K. Mogg. Do the affective properties of smoking-related cues influence attentional and approach biases in cigarette smokers? *Journal of Psychopharmacology*, 22(7):737–745, 2008. doi: 10.1177/0269881107083844. URL `http://dx.doi.org/10.1177/0269881107083844`. PMID: 18208922.

[70] Judith B. Cornelius, Michael Cato, Janet St Lawrence, Cherrie B. Boyer, and Marguerita Lightfoot. Development and pretesting multimedia hiv-prevention text messages for mobile cell phone delivery. *J Assoc Nurses AIDS Care*, 22(5):407–413, Jan 2011. ISSN 1055-3290. doi: 10.1016/j.jana.2010.11.007. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3614064/`. 21256053[pmid].

[71] Raymond C. W. Perry, Karen C. Kayekjian, Rebecca A. Braun, Michelle Cantu, Bhupendra Sheoran, and Paul J. Chung. Adolescents' perspectives on the use of a text messaging service for preventive sexual health promotion. *Jour—nal of Adolescent Health*, 51(3):220–225, 2017/09/01 XXXX. ISSN 1054-139X. doi: 10.1016/j.jadohealth.2011.11.012. URL `http://dx.doi.org/10.1016/j.jadohealth.2011.11.012`.

[72] Solorio R, Norton-Shelpuk P, Forehand M, Martinez M, and Aguirre J. Hiv prevention messages targeting young latino immigrant msm. *AIDS Res Treat*, 2014. URL `http://dx.doi.org/10.1155/2014/353092`.

[73] Judith B. Cornelius, Jacek Dmochowski, Cherrie Boyer, Janet St. Lawrence, Marguerita Lightfoot, and Michael Moore. Text-messaging-enhanced hiv intervention for african american adolescents: A feasibility study. *Journal of the Association of Nurses in AIDS Care*, 24(3):256–267, 2017/09/01 XXXX. ISSN 1055-3290. doi: 10.1016/j.jana.2012.06.005. URL `http://dx.doi.org/10.1016/j.jana.2012.06.005`.

[74] David Hammond. Health warning messages on tobacco products: a review. *Tobacco Control*, 2011. ISSN 0964-4563. doi: 10.1136/tc.2010.037630. URL `http://tobaccocontrol.bmj.com/content/early/2011/05/23/tc.2010.037630`.

[75] Deborah J. Bowen, Matthew Kreuter, Bonnie Spring, Ludmila Cofta-Woerpel, Laura Linnan, Diane Weiner, Suzanne Bakken, Cecilia Patrick Kaplan, Linda Squiers, Cecilia Fabrizio, and Maria Fernandez. How we design feasibility studies. *Am J Prev Med*, 36(5):452–457, May 2009. ISSN 0749-3797. doi: 10.1016/j.amepre.2009.02.002. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2859314/`. 19362699[pmid].

[76] Andrew C. Leon, Lori L. Davis, and Helena C. Kraemer. The role and interpretation of pilot studies in clinical research. *Journal of Psychiatric Research*, 45(5):626–629, 2017/09/01 XXXX. ISSN 0022-3956. doi: 10.1016/j.jpsychires.2010.10.008. URL `http://dx.doi.org/10.1016/j.jpsychires.2010.10.008`.

[77] Ministerio de Salud. Guia alimentaria Chile N148. `http://codemachile.cl/issues/norma-general-tecnica-no148-sobre-guias-alimentarias-para-la-poblacion` 2017. [Online; accessed 1-June-2017].

[78] Felipe Besoain, Antoni Perez-Navarro, Joan A. Caylà, Constanza Jacques Aviñó, and Patricia García de Olalla. Prevention of sexually transmitted infections using

mobile devices and ubiquitous computing. *International Journal of Health Geographics*, 14(1):18, May 2015. ISSN 1476-072X. doi: 10.1186/s12942-015-0010-z. URL `https://doi.org/10.1186/s12942-015-0010-z`.

[79] Felipe Besoain, Antoni Perez-Navarro, Felipe Ojeda, and Jose Antonio Reyes-Suarez. Promoting healthy nutrition behavior using mobile devices and ubiquitous computing. In *Proceedings of the 7th International Work-Conference on Ambient Assisted Living. ICT-based Solutions in Real Life Situations - Volume 9455*, IWAAL 2015, pages 89–100, New York, NY, USA, 2015. Springer-Verlag New York, Inc. ISBN 978-3-319-26409-7. doi: 10.1007/978-3-319-26410-3_9. URL `http://dx.doi.org/10.1007/978-3-319-26410-3_9`.