

A Proposal for a Technological Solution to Improve user Experience in a Shopping Center based on Indoor Geolocation Services

Luinel Andrade, Johan Quintero
School of Computing
Central University of Venezuela
Caracas, Venezuela

Eric Gamess
Mathematical, Computing, &
Information Sciences
Jacksonville State University
Jacksonville, Alabama, USA

Antonio Russoniello
School of Computing
Central University of Venezuela
Caracas, Venezuela

Abstract—For shopping centers, mobile devices and their associated technologies represent great business opportunities and a way to improve the user experience within their facilities. These types of constructions are usually quite large, multi-story, and with a significant number of shops, services, where the visitors may find themselves having difficulties to have a complete and up-to-date list of the stores, determine which stores and services are those that meet the characteristics or specifications they seek, know the location of the shops or how to reach them. This research studies and contemplates different technologies, tools, and approaches for the development of a technological solution for shopping centers that offers in its functionality a geolocation system in the interior spaces of the buildings. Our technological solution includes a mobile application for the Android operating system implemented by using the native development approach, and a web application for managing data, where the contents and settings of the mobile application will be obtained following the client/server model through a private API. It is worth to mention that the already mentioned system of geolocation in interiors is implemented using WiFi technology and the different Access Points installed in the shopping center, through which users can obtain their position, locate the stores or services of their interest, and receive indications on how to reach them.

Keywords—*Mobile application; web application; geolocation; shopping center; WiFi; access points*

I. INTRODUCTION

As time goes by, technology is evolving rapidly, thus increasing the solutions that seek to facilitate the day-to-day life of people. It is a fact that technologies such as mobile devices and web applications are part of the daily occupations of people, in both the personal and work activities, and influence numerous aspects of their life such as personal relationships, entertainment, education, and economic activities, among others.

Mobile devices evolve and produce changes in society through the different technological features that they include and the diverse applications that are available to users, generating new trends and behavior models that are increasingly becoming more relevant for organizations and companies from different sectors of activity, creating great business opportunities when taking advantage to them. For

these reasons, shopping centers should not only focus on increasing their sales and the number of visitors, but also improve the experience of visitors within their facilities, by providing a service that allows the interaction to be more personal and immediate, generating good memories into the people, and getting closer to their consumers, by incorporating the daily technological resources that they carry and use.

Shopping centers are usually very large constructions, with lot of space, several stories and a large number of shops, services and parking spaces, where visitors may have difficulties to know which stores are in the place and their location, which stores and services are those that meet the characteristics or specifications they seek, and much more. The first objective of this research is to contemplate and analyze various technologies, before developing a mobile application, through which users can see information of a shopping center, browse directories of stores and services, consult maps, have a geolocation system in interiors, and see the various promotions, sales and events offered. Additionally, we also develop a web application that allows the administrative staff of the commercial center to manage the data, content and settings that will be displayed in the mobile application. It is worth to mention that the research was done for a mall in Venezuela that we will not identify for privacy reasons.

The rest of this paper is structured as follows. In Section II, we review the previous work. An introduction to mobile devices and the Android operating system is done in Section III and Section IV, respectively. The different implementation approaches for a mobile application are presented in Section V, while Section VI discusses the development tools. The architecture of our solution is presented in Section VII, with more details of the web application in Section VIII, and the mobile application in Section IX. Section X is dedicated to the evaluation and validation of our work. Finally, Section XI concludes the paper and gives directions for future development.

II. RELATED WORK

Larcomaps is a mobile application developed for the Larcomar shopping center, located in the district of Miraflores in Lima, Peru. It has both Android and iOS versions. This application provides the map of the facilities of the shopping

center, with all its stores, parking, etc. In addition, it offers a routing service that shows users the path they must follow to reach a certain destination, such as a store, a bathroom, a food court, or any other internal location of the shopping center. Another important feature of this application is to allow users to add a reminder of the place where they parked their car.

Plaza Las Americas, the biggest shopping center in San Juan, Puerto Rico, has its own mobile application developed for the Android and iOS operating systems. Through the Plaza-Mall App, users are able to access the stores' directory, information about the mall's services, maps, promotions, and offers of the stores, among other functions. Users can purchase gift cards from the mall directly from the application, and also have the possibility to log-in through Facebook or register to receive offers and information through emails. To facilitate the location of their vehicle, users can save the information of where they parked, by taking a photo, recording a voice message, or marking the location with a pin.

Indoor Location [1] is an application developed by a group of students from the Complutense University of Madrid, Madrid, Spain, which allows indoor geolocation of mobile devices using WiFi networks. It is developed with Java and uses Swing and AWT (Abstract Window Toolkit) as graphic libraries, which makes it a completely cross-platform application.

ProMotion App [2] is an application that presents all the stores and shops affiliated with the system to users, where they can see promotions of the stores through notifications when physically approaching them. The promotions notifications appear in the upper part of the screen of the mobile devices when they are within a range of 0-30 meters from the associated stores, using Bluetooth Low Energy technology and Beacons devices as a location method.

III. MOBILE DEVICES

A large number of electronic devices are currently classified as mobile devices, for example: cell phones, portable game consoles, tablets, digital agendas, calculators, digital cameras, among others. With this diversity of devices, it is difficult to determine which characteristics must be possessed to be considered as mobile devices. Some essential aspects that they fulfill are: (1) they are small devices that can usually be transported in the pocket of the owner or in a small bag, (2) they have processing capacity, permanent or intermittent connection to a network, and memory, and (3) they are usually associated with personal use and have a high capacity for interactions through the screen or the keyboard.

This research focuses on two types of mobile devices that are mobile phones (specifically smartphones) and tablets.

A. Operating Systems for Mobile Devices

In a few words, it could be said that an operating system (OS) is a program that manages the hardware of a computer. It also provides basic application programs and acts as an intermediary between users and the computer's hardware [3].

A mobile device also has its own OS, which is loaded and executed when the device is turned on, and in charge of managing the resources of the system, both software and

hardware, thus allowing communications between the equipment and users. Mobile operating systems are focused on mobility, wireless connectivity, multimedia formats and the optimal management of processing, storage, and energy consumption [4].

- **Android:** It is an operating system and a software platform, based on Linux for mobile phones, although it is also used for other devices such as tablets, wearable devices, music players, and even netbooks. Older versions of Android allow users to program applications in a Java environment, on a Dalvik virtual machine that is a variation of the Java virtual machine with compilation at runtime [5]. Newer versions of Android are based on Android Runtime (ART), which uses ahead-of-time (AOT) compilation to entirely compile the application bytecode into machine code upon the installation of an application. It was developed by Android Inc., a company that was later bought by Google in 2005, but it was not until 2008 when it began to become popular, thanks to the Open Handset Alliance consortium, which decided to join the project, promote free software, and develop open standards for mobile devices.
- **iOS:** It is an operating system developed by Apple Inc. for its devices such as iPhone, iPod Touch, and iPad. Since it is a proprietary operating system, it cannot be installed on devices of other brands. The first version of iOS was presented in 2007 along with the first iPhone. It is derived from MacOS, which was also developed by Apple Inc. for its range of Macintosh computers since 2002.
- **Windows Phone:** Windows Phone is a mobile operating system developed by Microsoft for its smartphones that was first released in October 2010. It was introduced as the successor of Windows Mobile 5, 6, and Zune, and has a number of changes that make it primarily aimed at the consumer market rather than the business market. Microsoft dropped it in the first quarter of 2015, to focus on the mobile operating system called Windows 10 Mobile, which primary focus is the unification with Windows 10, its PC counterpart.
- **Blackberry OS:** It is a proprietary operating system embedded in the wide range of mobile phones of the Canadian company BlackBerry Limited, formerly called Research In Motion (RIM). BlackBerry OS has a core based on the Java Virtual Machine (JVM). It is written in Java and C++.
- **Firefox OS:** It is an open-source mobile operating system, based on HTML5, with a Linux kernel, for several platforms. It is composed of 3 layers, which are: Gonk (the core), Gecko (the web rendering engine), and Gaia (the user interface). At the end of 2015, the Mozilla Foundation concluded the development of the Firefox OS system for mobile phones and announced the end of its development. The main obstacles encountered in the development of the system were

commercial. The system was not sold and the costs exceeded the benefits.

IV. ANDROID OPERATING SYSTEM

In this section, we focus on the Android OS because we selected it for the development of the mobile application. Android is an operating system and a software platform based on the Linux kernel [6]. It is open-source, and can also be freely extended to incorporate new cutting-edge technologies that are emerging. It has been written for mobile devices and one of the principles of its design is to give developers the ability to create mobile applications that take full advantage of all the tools that modern mobile devices can offer.

A. Architecture of Android

Android is a stack of open-source software based on Linux created for a wide variety of devices. Fig. 1 shows the main components of the Android architecture [7].

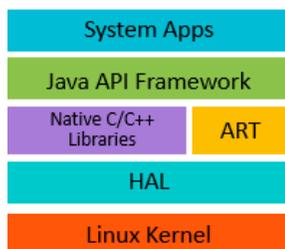


Fig. 1. Architecture of android.

- **Linux Kernel:** The kernel is the basis of an operating system. It provides fundamental system services such as security, memory management, process management, network functions, and driver management [7].
- **Hardware Abstraction Layer (HAL):** Provides standard hardware interfaces of the device to the Java API Framework. The HAL consists of several library modules and each of these implements an interface for a specific type of hardware component, such as the camera or Bluetooth modules [7].
- **Android Runtime:** Android runtime (ART) is the managed runtime used by applications and some system services on Android. ART replaced the Dalvik Virtual Machine (DVM) since version 5.0 of Android. Both were specifically created for this platform and are compatible with each other.
- **Native C/C++ Libraries:** Many components and central services of the Android system, such as ART and HAL, are based on native code that requires native libraries written in C and C++. The Android platform provides the Java API Framework layer to expose the functionality of some of these native libraries to applications [7].
- **Java API Framework:** The Java API Framework layer is a set of APIs written in Java where the full set of Android OS functions is available. These APIs are the foundations needed to create Android applications, simplifying the reuse of system components and services, such as the following [7]: View System,

Resource Manager, Notifications Manager, Activities Manager, and Content Providers.

- **System Applications:** Android includes a set of applications for camera, email, SMS messaging, calendar, contacts, Internet browsing, among other elements. These system applications provide key functionalities that developers can access from their own applications.

B. Android Applications and their Components

A mobile application, also known as an app, can be defined as a computer application designed to be executed on mobile devices to perform tasks, allowing users to satisfy needs of different types such as educational, professional, entertainment, health, etc. Mobile applications are found and run at the last level of the Android architecture and use services, APIs, and libraries of the lower levels. The file format used in Android to install applications is the .apk (Android Package). This .apk format is a variant of the Java .jar format.

Next, we present the different components related to an Android application.

- **Activity:** It serves as an entry point for the interaction of a user with an application, since it is the window in which the application displays its user interface. Also, it is fundamental for the navigation within an application or between applications.
- **Intent:** It is an object that can be used to request an action from another application component. Intents facilitate communication between components, and there are three fundamental use cases that are: (1) to start an activity, (2) to start a service, and (3) to deliver a broadcast.
- **Service:** It is an application component that can perform long-running operations in the background, and that does not provide a user interface. It can be started by another application component, and be run in the background, even if the user switches to another application.
- **Content Providers:** They manage the access to a structured set of data. They encapsulate the data and provide mechanisms to define data security. Content providers are the standard interface that connects data in one process with code that is executed in another process [8].

V. DEVELOPMENT APPROACHES FOR MOBILE APPLICATIONS

Choosing a development approach for a mobile application is one of the decisions that most influence the final result of the product, since it involves taking into account many aspects such as budget, project deadlines, application functionality, type of users to which the app is directed, among others. In the following subsections, the different development approaches are described, to give an idea of the advantages and disadvantages of each one.

A. Native Applications

It is a mobile application that is targeted to be installed and executed in a specific operating system. This type of applications can freely access all the APIs offered by the OS and, in many cases, has unique functions and features that are typical of the platform on which it is run.

The development process is usually similar for different operating systems, but the SDK (Software Development Kit) is specific to each platform, and each mobile OS has its own tools, languages, formats, and distribution channels [9]. The main disadvantage of this approach is that the mobile application developed for a specific operating system cannot be used in another platform, so that the same application would carry a different development and maintenance process in each OS.

In turn, the fact that an application is developed for a specific operating system is an advantage, since it allows to take better advantage of its specific hardware such as GPS, accelerometer, camera, compass, among others, because their implementations are developed in native programming languages for each OS and accessed through proprietary APIs of each system, hence obtaining a better performance compared to other development approaches.

B. Web Applications

These are applications that run in a web browser, and can be adapted and optimized for any mobile device. Modern mobile devices have powerful browsers that support the new functionalities of the HyperText Markup Language version 5 (HTML5), Cascading Style Sheets (CSSs), and the JavaScript (JS) language.

The main advantage of using web-based mobile applications is that they are multiplatform, so once developed, they can be distributed to different operating systems, which helps to minimize development costs when comparing to the development of a native application. Another advantage is that there is no need to control the version of the application that is published or the subsequent updates in the clients, since new versions or updates automatically reach all users, regardless of the OS or device they have.

Among the disadvantages of this development approach [10] is that an important part of the functionality of the device is being lost by not developing natively with the SDKs and APIs available for each platform. That is, in web applications, many of the features of the device cannot be used or only partially.

C. Hybrid Applications

A hybrid application is one that combines native development with web technology, where a large part of the application is developed with web technologies such as HTML5, CSS, and JavaScript, for multiple platforms, maintaining direct access to native APIs if needed; then it is wrapped into a native container using a packager, which acts as an intermediary and translates all the instructions so that the operating system of the device can understand them when executing the application. The native portion of the application uses operating system APIs to create a built-in HTML search

engine that functions as a bridge between the browser and the device's APIs. This bridge allows the hybrid application to take advantage of all the features offered by modern devices [9].

The main advantage of hybrid applications is that, similarly to web-application, they are cross-platform applications so that they can be executed in different operating systems. Also, due to the aforementioned, they allow the reuse of code, so there is no need to write a different application for each system. Another important advantage is the "partial" direct access to native APIs and resources of the operating system.

However, one of the main disadvantages is that it does not have full access to all hardware resources and APIs available due to the limitation of the framework used to develop the hybrid app. Additionally, its performance is lower than the one of a native application since the latter run at a lowest level than hybrid ones, because they are developed directly for a specific operating system. Instead, a hybrid application has to be packaged, and each instruction has to go through a translation process before being executed.

D. Comparing the Development Approaches

Native applications stand out for their better performance and for having better access to the APIs and mobile device functionalities. However, they are not multiplatform and tend to have more expensive development processes in effort, time, and resources. Web-based applications are cross-platform, have a simpler and less expensive development process, and their updates are faster since the application resides on a server and reaches all users when the resources are requested. However, their biggest disadvantage is that they do not take advantage of all the functionalities of the devices. Hybrid applications are halfway between the two approaches mentioned above, offering some of the advantages that each one can have, especially if the developers want the application to be used in various operating systems, even though, most likely, the level of performance is lower.

The approach chosen by us to develop our application to improve user experience in a shopping center is the native one. Among some of the reasons for this choice is to have a user interface closely linked to the design trends of Android, and that we need to use native APIs to take advantage of certain functionalities of mobile devices related to WiFi.

VI. DEVELOPMENT TOOLS

In this section, we present a series of tools and technologies that are important and necessary to develop the technological solution in which the study is focused. To understand a little more why some of the technologies and development tools that follow are considered, it is worth to remember that the mobile application to be developed makes use of an indoor geolocation system, and the web application that supports the creation and administration of the contents uses a Content Management System (CMS).

A. Web Application

Regarding the development of web applications, a type of abstraction is made to differentiate and separate two parts of the system that are: Front-End and Back-End.

- **Front-End:** This represents the processes, functionalities, and technologies of the web application that run on the client's side, that is, the processes and technologies that run or are on the side of the web browser. It is usually associated with the visual part of the application, such as: the structure, styles, colors, animations, and effects.

The technologies used for their development are: HTML5, CSS, and JavaScript; where in a summarized way, HTML5 is used to define the structure, CSS to assign the styles, and JavaScript to bring dynamism to the web application.

- **Back-End:** This part of the application represents the processes that are executed on the server side, which are not visible to users, such as the processing of data, requests, etc. In addition, it allows the handling and manipulation of the processed data which implement the business logic, that is, the implementation of the functions, connections, and queries to the database, the management of sessions, among others.

Some of the programming languages that allow the development of this part of the application are PHP (PHP Hypertext Preprocessor), Python, .NET, Java, Node.js, etc. Developers can also use frameworks, which allow not only the development of the application's Back-End, but also the development of the Front-End.

B. Frameworks

They are software structures made up of customizable and interchangeable components for the development of an application. These frameworks have a set of tools, libraries, conventions, and good practices that seek to encapsulate repetitive tasks in generic reusable modules.

Currently, frameworks are used mostly in the development of web and mobile applications. Some examples of frameworks for web development are: Laravel, CakePHP, Symfony, Zend, CodeIgniter, Yii2, Ruby on Rails, among others. On the other hand, examples of frameworks for mobile development are: Ionic, jQuery Mobile, PhoneGap, and React-Native.

For this project, we decided to build the web application with Laravel. It is an open-source framework that was officially launched in 2011, multiplatform, and that allows the development of web applications and services with the PHP programming language.

Some of the advantages of this framework that have motivated our choice are that it has abundant documentation on the web, a large community of users, forums, it is modular and with a wide variety of package systems and drivers to extend its functionality. Also, it has different ways for accessing relational databases, and it provides utilities that benefit the application's deployment and maintenance. For example, it supplies a command line interface tool called Artisan that allows performing tasks such as data migrations, it has an ORM (Object-Relational Mapping) called Eloquent that makes the interaction with databases totally oriented to objects, and

finally, it is adaptable to the MVC (Model-View-Controller) pattern.

C. Database Management System (DBMS)

A DBMS [11] (Database Management System) is a coordinated set of programs, procedures, languages, etc., which provides the necessary means to describe and manipulate data stored in a database, guaranteeing their security. For our project, we selected MySQL that was originally developed by MySQL AB. This company was purchased by SUN Microsystems in 2008 and, in turn, SUN Microsystems was acquired by Oracle Corporation in 2010. We used the community version under the GNU General Public License.

Some of the reasons why we decided to work with this DBMS are: (1) it is an open-source database, (2) it is supported and compatible with multiple programming languages, and (3) it is very stable in applications where there is low concurrency in the modification of data but a high level of data reading, which fit to our needs of data management in both the web application and the mobile application.

D. REST Application Programming Interface (API)

The REST API [12] is an architecture for the exchange and manipulation of data. Currently, it is the most frequently used because it is the most logical and efficient architectural style in the creation of APIs for Internet services. REST allows the creation of applications and services that can be used by any client that understands HTTP to obtain data or generate operations on that data in various formats such as XML and JSON (JavaScript Object Notation).

A reason that motivates the utilization of REST is that it makes use of the global addressing scheme based on Uniform Resource Identifiers (URIs) which identify resources or conceptual objects. The representation of such objects is distributed through messages in the web.

E. Java

We used Java as a programming language to develop our mobile application, because it is the official language used to develop native mobile applications in the Android operating system. Java is a language with the following characteristics [13]: simple, object-oriented, distributed, robust, secure, multi-threaded, portable, neutral architecture, and high performance. The Java Virtual Machine (JVM) is a virtual environment that is used to interpret the Java bytecode, since Java was made to run on any platform without recompiling the binaries. That is, the bytecode can be obtained in any architecture and operating system, and be executed in any other system with a JVM.

F. Integrated Development Environment (IDE)

It is an application that contains a set of programming tools to facilitate the development of applications, that is, it is made up of a source code editor, a compiler, a debugger and tools to simplify the construction of Graphical User Interfaces (GUIs). We selected Android Studio, the official IDE for the development of Android applications, for the implementation of our mobile application. It is based on the IntelliJ IDEA software from JetBrains. We chose Android Studio since it provides good tools for the fast creation of applications for all types of Android devices. In addition, it has a powerful code

editor, the developer tools of IntelliJ, and even more features that increase productivity during the development of Android applications.

G. Geolocation with WiFi

Geolocation can be defined as the ability to calculate and obtain the geographical location of an object, such as a smartphone. Some of the technologies that can be used to carry out the geolocation are: GPS (Global Positioning System), RFID (Radio Frequency Identification), and WPS (WiFi-based Positioning System).

1) WiFi

We decided to choose the WiFi technology to implement the functions related to geolocation. WiFi is one of the communication technologies with the greatest presence in the world. It is based on the 802.11 standard of the Institute of Electrical and Electronics Engineers (IEEE), which has conquered the market since the first approval of the 802.11b specification in 1999 [14].

One of the main elements in WiFi networks in infrastructure mode are the Access Points (APs), which allow communications between the wired network and wireless devices. An AP receives and sends data, both through Ethernet cables and wirelessly. Other functions associated with these equipments are the access control, security, and isolation of the ambient noise or improvement of the signal/noise ratio [15]. The range of WiFi devices will depend on many factors, such as walls, ceilings, obstacles, etc.; and there may be other factors that, in turn, increase the range of the signal, such as the installation of WiFi repeaters.

2) Motivation to use WiFi Technologies for Geolocation

WiFi has advantages over the GPS geolocation system, because the latter does not work in interior spaces since GPS receivers cannot receive the signal from the satellites.

Another advantage of WiFi is that it is a widespread and daily use technology presents in many places such as homes, offices, schools, libraries, shopping centers, squares, etc. In the case of shopping centers, it is very common that there is a large number of APs inside the building, whether they are owned by the stores or they are part of the network infrastructure of the building, which makes it viable to have access to this technology. This high presence of APs within the facilities of those constructions allows data to be collected from the WiFi networks, and once a relationship between a group of WiFi networks and a certain geographical point is established, the location of users can be determined to support the indoor positioning system required by our mobile application.

3) Method of the k-Nearest Neighbors

Most methods of geolocation using WiFi technology are based on the signal strength with which the APs are detected by mobile devices. Depending on the algorithm used, the precision and complexity of the approach vary. Some of the methods to carry out the geolocation using WiFi networks are: power vector, power triangulation, proximity heuristics [6], method of the k-nearest neighbors [16], [17], probabilistic methods [15], and location using RTT through ping messages [1].

We selected the method of the k-nearest neighbors to develop the geolocation functionalities of the mobile application since, according to our research, it is the most appropriate for factors such as: (1) implementation, (2) calculation cost, and (3) precision. Below, we present more information about the method in details.

The k-nearest neighbor method is based on the definition of a metric in the space of cells covered electro-magnetically by the APs. Each cell or area of the space considered to perform the geolocation is assigned a value for every AP that represents the power of its signal received in that place. When the power is measured at an arbitrary point, the distance is defined as the difference between the initially recorded power during the fingerprinting process and the measured power. The absolute value of this difference will be considered as a distance [15].

Once a measurement is made in a specific place, the method looks for a point that minimizes the sum of the squares of the distance. If there were no measurement errors, there should be a point for which the quadratic sum of distances is zero. However, since errors always exist, the point for which the quadratic sum of the distance is minimum is considered to be the best approximation that can be obtained.

The fingerprinting process should be done as a preliminary phase to the implementation of the k-nearest neighbor method. To do so, a map of the place is required, and it must be divided into sectors or areas that have uniform measurements. The locations of the APs and the locations where the measurements will be made must be recorded on the map. Then the building must be traversed, and in each measurement location, the values of the strength of the signal received from each accessible AP must be kept. It is recommended to take different measurements for each AP from the most possible centered position of each of the sectors, and each geographical orientation, keeping the height of the device constant.

Depending on the distance between the measuring points that are taken during the fingerprinting phase, the precision of the algorithm will vary. This process is performed once and serves to calculate the position of all users. When there are structural changes in the building or considerable changes in the infrastructure of the wireless network (e.g., new positions for some APs, new APs, eliminated APs), there is a need to repeat the fingerprinting process. Once the signals have been scanned, each sector must be assigned a characteristic measurement that represents the signal of each AP in that location.

To get the position of a device, the measurement of the power of the signal must be made in the device for each AP, obtaining as a result a vector x_1, x_2, \dots, x_N , where N is the number of APs. It is assumed then that $P_{A_j,k}$ is the power that was measured from AP k in region A_j .

Consider the sum (1), the estimation of the point where the signal measurement was made will be the region for which S_{A_j} is minimum.

$$S_{A_j} = \sqrt{\sum_{k=1}^N (P_{A_j,k} - x_k)^2} \quad (1)$$

VII. ARCHITECTURE FOR THE PROPOSED SOLUTION

Fig. 2 depicts the general architecture of the technological solution, where the position of each element to be integrated is specified.

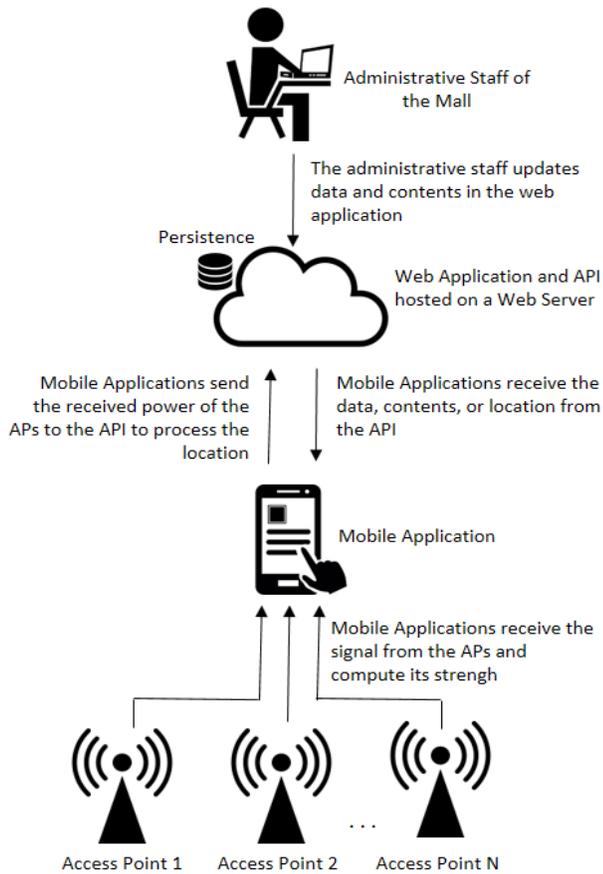


Fig. 2. General architecture of the solution.

VIII. INTRODUCTION AND ANALYSIS OF THE WEB APPLICATION

The web application for the management of the data, contents, and configurations that the mobile application consults was developed using the Laravel framework and MySQL DBMS for the persistence of data. In general, the functionalities offered by this application are the management of the data to be consumed by the mobile application through the API. These data include: global information of the shopping center, information about shops, services, categories, promotions, events, and data related to the geolocation system such as maps, sectors, access points, and measurements done during the fingerprinting process. In addition, the web application allows the management of users and users' profiles, inherent of its administration.

A. Development of the Web Application

To understand how the development process was made using the Laravel framework, it is worth to mention that all the modules had a similar development. Hence, the general process to carry out the coding is explained below.

Once configured the connection to the MySQL DBMS, the models can be created from the command line using the commands from Artisan. The models were created in the folder "/app;" which are PHP files with information of the tables of the database to be used for the respective model. Then, the migration file must be created. Migration files are found in the path "/database/migrations" and allow the creation of the structure of a table of the database. This type of files facilitates a version control on the tables of the database.

The controllers can be created using commands from Artisan, in order to handle the events generated by users and invoke the requests to the models when they are made on the information. These controllers are PHP files that are generated in the path "/app/Http/Controllers." In the header of those files, it is necessary to import the model of the represented entity and any other models that are required to be used in that controller.

Once having the controllers, it must be ensured that the application has the necessary routes configured. The routes of the application are defined in the file "routes.php" located in the path "/app/Http." When accessing a route from a browser, an HTTP request is sent to the aforementioned file. An error is generated if the route does not exist. Otherwise, it redirects to the controller that contains the logic of the application. Instead of completely defining the logic of the requests in the file "routes.php," it is also possible that this behavior is organized through classes of type "Controller". Controllers work with the GET, POST, PUT, and DELETE requests, which can be grouped and manipulated logically in a class. The methods of the controllers are associated with the HTTP requests in the following way: GET (index, create, show, edit), POST (store), PUT (update), and DELETE (destroy). When using a type of route called "resource" in the file "routes.php," Laravel internally creates the application routes related to a specific resource and assigns them to the controller's methods.

The logic of the application is coded in the different methods of the controller. Some of the different actions that are carried out in these methods are: process the requests generated by the users, interact with the model for the manipulation of the data, validate data in the forms, validate the users' permissions on the different modules, send data received from the model to the views, etc.

The views can be understood as the public part of the system that users will be able to see. They are implemented using HTML to define the structure, CSS to assign the styles, and JavaScript to apply the dynamism. In addition, in the case of Laravel, a templating engine called Blade is also used. The views are located in the path: "/resources/views".

In the framework, template files can also be used and are usually the files of the main views, which have the segments of code that are repeated in more than one view, such as the header, the main menu, and the footer. The idea is to keep common code in a specific place, instead of repeating it in several views.

As an example, Fig. 3 depicts the view shown to users when running method "index()" in the Commerce module (i.e. the directory of stores that are stored in the database).

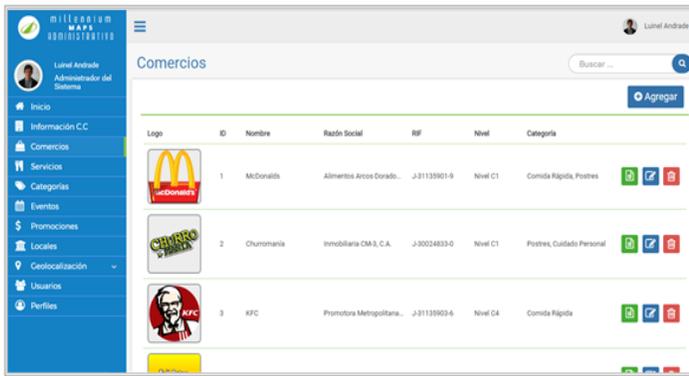


Fig. 3. Directory of the stores.

Fig. 4 depicts the view shown to users when running method “show()” in the Commerce module (i.e. the specific details associated with a store).



Fig. 4. Detailed view of a specific store.

B. Development of the API

Once developed all the modules of the web application, it was straightforward to write the API since it was only necessary to develop new controllers that will validate a key called API Key, through a middleware. In the API of this project, it was only necessary to implement query methods on the database. The queries are made through GET requests of the HTTP protocol.

In the API for routes, one of the most important functionality is done by the method to get the geolocation in the interior space of the shopping center. This method returns the sector of the shopping center in which a user is located and is available in the path “api/med/locate”. In this function, the method of k-neighbors is implemented to compute the user’s location, according to the strength of the signal received from the APs.

To get the position of a user, the mobile application sends the information (MAC Address, signal strength, etc.) of the reachable access points to the API. The API receives the request, analyzes and processes the received data, before returning the <Map-ID> and <Sector-ID>, of the actual location.

IX. GENERAL ANALYSIS OF THE MOBILE APPLICATION

The mobile application was developed using Android Studio as an IDE and Java as the programming language, since Java is the official language to develop applications for Android. For the development of the application, we used Git for version control, and the GitLab platform which is based on Git and allows private repositories for free.

At a glance, the application has a set of functionalities that allow users to see the list of shops of the mall, the promotions of these shops, the events and services offered by the shopping center, as well as the maps of each floor. In addition, it has a function to record the parking ID where the user’s vehicle is parked, and a geolocation system so that the user can be located inside the shopping center.

A. Development of the Mobile Application

For the development process, different types of files were used. Java classes allowed to manage the behavior and functionalities of the application. The layout files, with “XML” extension, represent the user interface of the application. It is worth to mention that the layouts are associated with the Java classes so that, from the code, it is simple to get access to the elements of each view.

In addition to the aforementioned files, other files are used to represent certain resources that the application needs such as: images, strings, colors. To represent the images, a directory called “drawable” is used. For the usage of certain text strings defined in the project, an XML file called “strings.xml” is used. The “colors.xml” file allows the establishment of the colors.

The file “AndroidManifest.xml” was also used. In this file, the Android Manifest provides essential information about the application to the operating system so it can execute the code of the application. Finally, the gradle files represent the settings for the compilation of the application, the minimum SDK on which the application can run, the version of the SDK with which the application was compiled, its dependencies, among others.

To obtain all the information that is shown in each module of the application, requests are made to the API, using Retrofit, which is a REST client for Android and Java. These requests are made in a file called “APIServices.java”, where the type of request is specified (GET, POST, PUT, or DELETE), the URI, the name of the function to which the call will be made from the code, the object to receive the response, and the parameters that will be passed to the request. Upon receiving the response of these requests, they are stored in an object according to the models defined in the applications which are made up of the properties and their respective getters and setters.

As for the modules of the application, some of them list elements such as: stores, categories, promotions, events, and services; in each of these lists, when selecting a specific element, it is presented/shown in details.

Fig. 5 depicts the screens of the lists of shops, promotions, and events. For each of the previous category, the elements are shown, with certain “filtering” options, for a better selection.



Fig. 5. Screenshot of the lists of shops, promotions and events.

Fig. 6 shows the information associated with a store: Store Info, Promotions, and Events. In “Store Info”, a general description of the store is presented, and includes: (1) the floor where the store is situated, (2) a brief description of the store, (3) its electronic mail, (4) the URL of its web page, (5) its opening hours, (6) its telephone number, (7) its social networks, and (8) the categories to which it belongs. At the level of a promotion, the information includes: (1) a brief description of the promotion, (2) the range of dates of the promotion, (3) the range of hours of the promotion, and (4) the categories to which it belongs. At the level of an event, the information includes: (1) a brief description of the event, (2) the range of dates of the event, (3) the range of hours for the event, and the place where the event will be celebrated. In addition, the details of the shop and the event have an additional option. In the case of the shop, users can view the location of the shop in the maps of the shopping center. In the case of the event, users can add this event to the calendar of their mobile device.

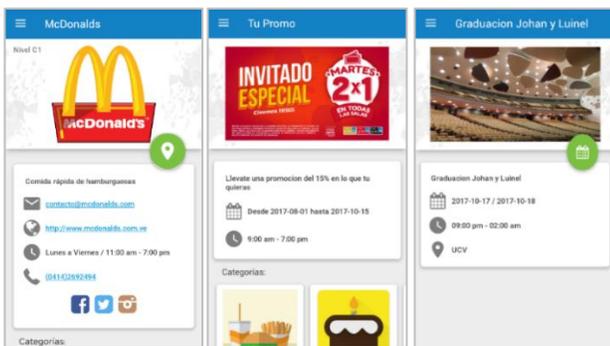


Fig. 6. Detailed screenshot of a store info, promotions, and events.

Furthermore, the application has modules such as: “Contact”, “About the Application”, and “Your Parking Position”. The “Contact” module has general information about the shopping center, including contact information. The “About the Application” module provides useful information about the developed application (the authors). The “Your Parking Position” module allows users to view, modify, and eliminate the parking position where their vehicle is stationed.

Fig. 7 shows the “Contact” module, where the general information of the shopping center is given, which includes: (1) a brief description, (2) the address, (3) email, (4) web page, (5) opening hours, (6) telephone numbers, and (7) social

networks. In addition, users have the option to locate the shopping center on Google Maps.



Fig. 7. Screenshot of the contact module.

Finally, the “Maps” module was developed, which represents the most complex part of the application. This module has two operation modes: (1) the user is inside the facilities of the shopping center, and (2) the user is outside the shopping center. When the user of the mobile device is outside the shopping center, the “Maps” module displays a message to inform the user that he/she is outside the shopping center facilities, and therefore he/she cannot be located within it. However, the user can still make use of the other functionalities of this module such as seeing the maps of each floor of the shopping center, etc.

The function of “How to Get to a Store” consists in showing the list of shops of the shopping center, and when the user selects a specific store, an emergent message is shown with the instructions on how to get to the required shop. The function “List of Services” consists in showing the list of services of the shopping center. When the user selects a specific service, he/she is shown a map per floor that has only the requested service.

Fig. 8 depicts the “Maps” module, which shows the directions on how to get to two different shops (Adelmed and Mercantil). Additionally, users have the option to see the list of shops, the available services, and there are six buttons to change the current floor.

When the user is inside the facilities of the shopping center, the “Maps” module works practically in the same way as the previous case, with the difference that in this case, when the map module is started, the map of the floor where the user is located is shown, and the actual sector of the user is marked in green on the map.

Once the Maps module is started, a capture of the neighboring access points is made every eight seconds, to get the MAC address, the SSID, and the strength of the signal of each access point. Subsequently, these data are sent to the API, and the response of the request is the sector where the user is located, at the time of each capture.

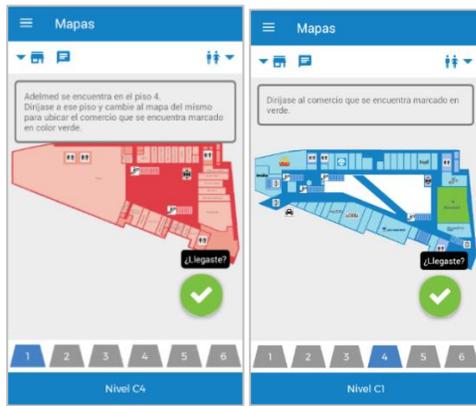


Fig. 8. Screenshot of the maps module with indications on how to reach a specific store.

In each capture, the sector where the user is located is obtained, and the picture of the corresponding map is shown, with this sector in green. In the next capture, if the new sector that is obtained as a result of the API is in a different floor, a popup message is shown to inform the user of the new floor, and the picture of the map of the new floor is loaded, with the new sector in green. Fig. 9 depicts the map of the 4th floor (Level C1) with the actual sector of the user shown in green.



Fig. 9. Screenshot of the map of the fourth floor (level c1) where the green sector specifies the actual position of the user.

B. Development of Tools for the Process of Fingerprinting

In addition to the development of the web application and the mobile application in which the present research work is focused, we also developed a second mobile application for Android OS and a program in C++ to support the fingerprinting process that must be done to feed the database, and be able to process the algorithm of the nearest k-neighbors. The objective of this second mobile application is to facilitate and speedup the recollection of data from the APs in the different sectors of the shopping center.

Its usage is straightforward and consists in identifying the sector to be captured, before selecting if the data is collected for insertions in the database or for testing the geolocation functionality in the API. For each reachable AP, the data

captured are: SSID, MAC, and signal strength. Fig. 10 shows a screenshot of the application.

Once the fingerprinting process is over, the results of the capture of the mobile device must be transferred to a PC. In the PC, the C++ program reads the capture, groups the results by AP, and computes the average of the signal for each AP. Then, the C++ program stores the results in a file with a specific format with four data: (1) the MAC of the AP, (2) the SSID of the AP, (3) the average of the strength of the signal, and (4) a recommendation of whether or not the information associated with the AP must be stored in the DB. This decision is based on the stability of the signal captured from the AP. That is, if the standard derivation is low, the AP is considered stable, and the C++ program will recommend the insertion of its data in the DB.



Fig. 10. Screenshot of the mobile application for the process of fingerprinting.

X. TESTS AND ANALYSIS OF RESULTS

In this section, we present the tests made to the mobile application and the results obtained, in order to evaluate the level of usability, the level of quality, and the compliance with the specific objectives and functions for which it was created.

A. Acceptance Tests

A technical survey was applied to a group of 50 end-users of different genders, social and educational level, in order to assess the usability and proper functioning of the app, and determines if it meets the requirements established in the project. The survey consisted of twenty questions that are listed below:

- 1) How do you rate the combination of colors used for the application?
- 2) How do you evaluate the distribution of the elements in the application?
- 3) How do you assess the application design?
- 4) How do you rate the navigation in the application?
- 5) How do you evaluate the fluency when running the application?
- 6) Do you consider that the application is easy and intuitive to use?
- 7) How do you evaluate the splash screen shown when launching the application?

- 8) Does the application's menu displays its options clearly and legibly?
- 9) Do you consider that the elements shown in the Home of the application are the indicated?
- 10) Do you consider that "Your Parking Position" is an important functionality for the application? That is, the functionality of letting the user write down his/her parking slot, and receiving directions on how to get back to the car.
- 11) Do you consider that it is easy and intuitive to handle the functionality of "Your Parking Position"?
- 12) Do you consider that the list of shops, promotions, events, and services clearly shows each element and what they refer to?
- 13) Is the filtering option easy to manage in the list of shops, promotions, events, and services?
- 14) Do you consider that the information shown for the elements (stores, promotions, events, services) is adequate?
- 15) Is the option of "location" can be clearly understood in the Trade Detail, Service Detail and Contact?
- 16) Is the option "Add to the Calendar" is clearly understood in the information of events?
- 17) Is it clear that the texts that represent emails, web pages, and telephone numbers are clickable?
- 18) Do you consider that it is easy and intuitive to use the Maps module in the application?
- 19) Is it easy and intuitive to understand the option of "services" and "shops" of the Maps functionality?
- 20) Do you consider that the indications about how to reach a certain shop of the "Maps" functionality are clear?

Fig. 11 and 12 depict the results of the survey.

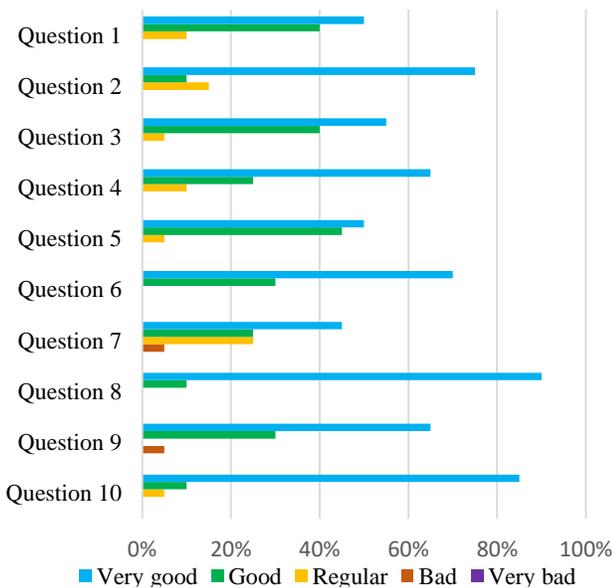


Fig. 11. Results of Questions 1 to 10 of the survey.

In general, the results of our survey show that the answers given by the people who assessed the application were positive; hence, we can say that it complies with the established functional requirements and with the scope defined

in the beginning. However, it is worth to mention that in some of the questions asked, some regular responses were obtained. Although it does not represent the majority of the opinions, it is important to consider correcting these aspects of the application as future work, or before the application passes to a production phase, since they are an indication that the application can be improved.

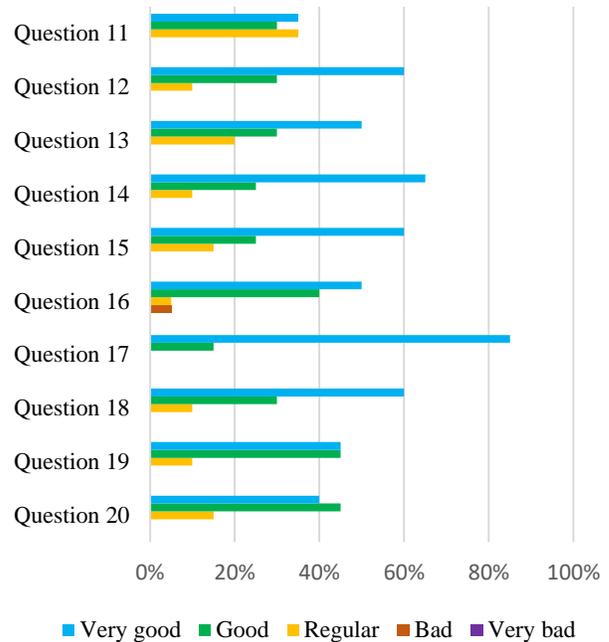


Fig. 12. Results of Questions 11 to 20 of the survey.

B. Geolocation Tests

To test the k-neighbor algorithm as a WiFi geolocation method and for purposes of the present investigation, we decided to use only two levels of maps of the selected shopping center (Level C1 and Level C2), where Level C1 was divided into eight sectors (sector 1 to sector 8), and Level C2 was divided into seven sectors (sector 9 to sector 15). As stated previously, the work was done in a mall of the Venezuela that we will not identify for privacy reasons.

To generate the database used by the algorithm in the fingerprinting process, thirty samples of the signal of each AP were taken in the centroid position of each one of the sectors. Through the C++ program that we developed, we analyzed the collected data and selected the APs that were more stable in the captures for processing and insertion in the database through the web application.

Once the fingerprinting process was completed, several tests were carried out within the shopping center facilities to assess its correct operation. The tests were conducted in two different scenarios. In the first scenario, the shopping center was closed, that is, the corridors were completely free, and there were no people in it. In the second scenario, the shopping center was opened to the public, with many visitors in the corridors. The idea of the second scenario was to evaluate the alteration made by the mobile devices of the numerous visitors over the geolocation service.

For each sector, we did twenty geolocation tests. Five of these tests were performed statically in the approximate position of the centroid of the sector. The remaining fifteen tests were made with a mobile device in movement within the sector. In the testing process, a result was considered correct if the API returned the sector where the user was exactly positioned. In the border of two sections, we considered that the result was correct if the algorithm returned any one of the two adjacent sectors. On the other hand, a result was considered as incorrect if the API returned a sector of another level with respect to the current level of the user, or when the returned sector was in the current level of the user, but did not correspond to its actual sector.

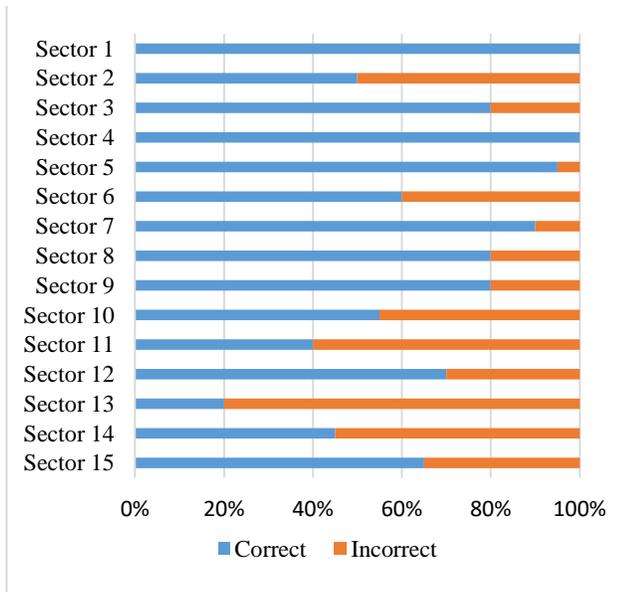


Fig. 13. Results obtained by the geolocation system when the shopping center was closed.

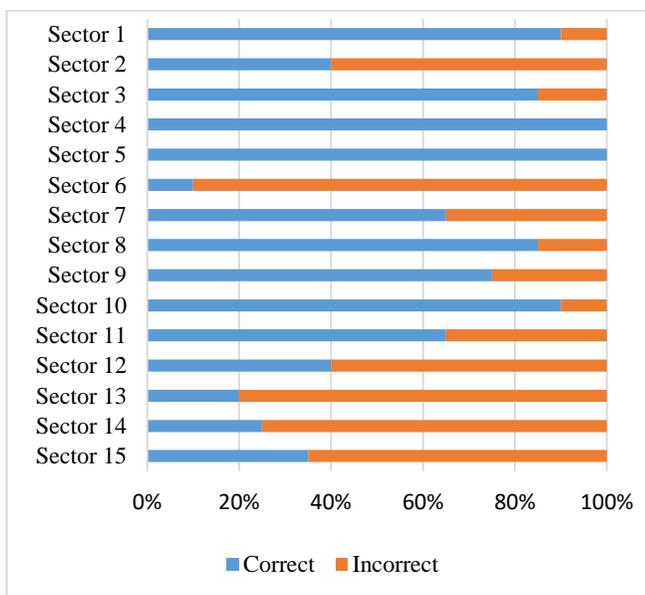


Fig. 14. Results obtained by the geolocation system when the shopping center was open.

Fig. 13 and Fig. 14 show the results of the geolocation tests in the interior spaces of the shopping center when it was closed and opened to the public, respectively.

For the geolocation tests carried out, the general percentage of success was 69.0% when the shopping center was closed, and 61.7% when the shopping center was open. It is important to mention that the selected shopping center did not have its own WiFi network. We worked with the APs installed by the shops, which had different characteristics (different brands, standards, antennas, signal power, etc.). Better results could be achieved if all the considered APs had the same characteristics, and if we knew their exact position. In some APs, it is even possible to change the firmware, and thus modify the power with which the signal is sent according to the needs that are presented by the structure of the building.

XI. CONCLUSIONS AND FUTURE WORK

For companies and organizations, mobile applications represent an added value and a high advance in technologies. In the case of shopping centers, they offer an increase in productivity and greater business opportunities, representing a very important support in the relationship between the mall and its visitors, expanding the network of benefits for both parties.

In the present research, a series of concepts, tools and technologies were presented that served as a solid base for the successful development of a technological solution for shopping centers that allows the diffusion and management of data, such as contact information, directory of shops, promotions, events, services, maps, and a geolocation system based on WiFi. We developed a solution for a shopping center of Venezuela that includes a mobile application for mobile devices with Android OS, a web application implemented using the Laravel framework for data and content management to be consulted by the app, and a private API as a communication mechanism between both applications. It is worth mentioning that the APs that were used in the location service, that belong to the shops within the facilities of the shopping center, are also part of the architecture of the solution. Thanks to these APs, the functionality of geolocation in interior spaces were implemented, by using the algorithm of the nearest k-neighbors. Even if the research was done for a specific mall in Venezuela, it could be reproduced in any shopping center of the world.

In this work, we could verify how certain factors are vital to guarantee a precise geolocation system. Based on this acquired experience, some recommendations can be given to improve the results: (1) pre-establish the location of the APs avoiding obstacles that could interfere their signal, (2) use APs of similar characteristics in terms of power of transmission and subsequently modify these parameters to fine-tune the coverage footprint of each device, (3) perform a monitoring and control of the APs to take action against possible changes in their transmission power, (4) evaluate the use of wireless repeaters in specific cases where the signal suffers attenuation.

As future work, we propose to extend our work so we can have WiFi coverage in the internal parking spaces of the shopping center. Also, we plan to consider other network technologies such as Bluetooth and Beacon devices, and make

a comparative study regarding the accuracy of the geolocation system according to the selected technology.

REFERENCES

- [1] P. Mulas, A. González and R. Rivera, "Localización de dispositivos móviles en interiores usando redes wireless", Faculty of Computing, Complutense University of Madrid, Madrid, Spain, 2007.
- [2] P. Díaz and E. Alvarado, "Desarrollo de soluciones web y móvil para la integración de marketing de proximidad y gestión publicitaria con dispositivos Beacon como tecnología base", School of Computing, Central University of Venezuela, Caracas, Venezuela, 2016.
- [3] A. Silberschatz, P. Baer Galvin, and G. Gagne, "Fundamentos de sistemas operativos". Seventh edition, McGraw-Hill, 2005, p. 3.
- [4] G. M. Ramírez Villegas, "Seguridad en aplicaciones móviles", Unidad Educativa Nacional Abierta y a Distancia, April 2013, http://datateca.unad.edu.co/contenidos/233016/EXE_SAM/leccin_1_sistemas_operativos_moviles.html.
- [5] M. Báez, Á. Borrego, J. Cordero, L. Cruz, M. González, et al. "Introducción a Android", Complutense University of Madrid, E.M.E. Editorial, Madrid, Spain.
- [6] A. Chico, "Diseño y desarrollo de un sistema de posicionamiento en interiores basado en WiFi con tecnología Android", University Carlos III of Madrid, Madrid, Spain, December 2009, pp. 59-62.
- [7] Developer Android, "Platform architecture", <https://developer.android.com/guide/platform/index.html>.
- [8] Developer Android, "Content providers", <https://developer.android.com/guide/topics/providers/content-providers.html>.
- [9] IBM Corporation, "El desarrollo de aplicaciones móviles nativas, Web o Híbridas", New York, United States of America, April 2012.
- [10] P. Rincón, "Aplicaciones móviles nativas con consumo de APIs online, estudio comparado con aplicaciones web móviles en iOS y Android y caso práctico 'Native Client' para Wordpress", University Carlos III of Madrid, Madrid, Spain, July 2012.
- [11] M. Piattini and A. Miguel, "Fundamentos y modelos de base de datos". Segunda Edición, Editorial RA-MA, Madrid, Spain, 1999.
- [12] A. Richardson, "Automating and testing a REST API: a case study in API testing using: Java, REST assured, Postman, tracks, URL and HTTP proxies, compendium developments Ltd", 1 edition, August 2017.
- [13] G. Torres, "Espacios virtuales de experimentación cooperativa. Caso de estudio: Laboratorio virtual de cinemática", Center for Research in Information Technology and Systems, Autonomous University of the State of Hidalgo, Pachuca, Mexico, 2001.
- [14] M. El Yaagoubi, "Acceso a internet vía WiFi-WiMax", Department of Electronic Technology, University Carlos III of Madrid, Madrid, Spain, 2012, p.31.
- [15] D. Cohen, L. Cohen, G. Faillace, M. Gianatelli, V. Haber et al. "La localización utilizando WiFi (802.11b-g), diferentes algoritmos", Faculty of Engineering, University of Palermo, Buenos Aires, Argentina, 2005, pp. 5-13.
- [16] H. Nascimento, E. Rodrigues, F. Cavalcanti, and A. Paiva, "An algorithm based on Bayes inference and k-nearest neighbor for 3D WLAN indoor positioning", XXXIV Brazilian Symposium on Telecommunication, September 2016, Santarem, Brazil.
- [17] P. Torteeka and X. Chundi, "Indoor positioning based on WiFi fingerprint technique using Fuzzy k-nearest neighbor", in Proceedings of the 2014 11th International Bhurban Conference on Applied Sciences and Technology (IBCAST 2014), January 2014, Islamabad, Pakistan.