



**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**Facultad de Ingeniería en Electricidad y Computación**

“DESARROLLO DE UN SISTEMA UNIFICADO DE  
MONITOREO, NOTIFICACIÓN Y CONTROL DE  
PARQUEOS DISPONIBLES EN ESPOL”

**INFORME DE MATERIA INTEGRADORA**

Previo a la obtención del Título de:

**INGENIERO EN TELEMÁTICA**

GABRIELA ALEXANDRA TRUJILLO CASTILLO

RONALDO YOUSSEF CAMPUZANO POMA

GUAYAQUIL – ECUADOR

AÑO: 2017

## **AGRADECIMIENTO**

Agradezco principalmente a Dios, por protegerme día a día y permitirme disfrutar de mi familia y amigos, quienes han sido de gran apoyo para la realización de este proyecto.

Agradezco también a mis padres, cuyos consejos y advertencias me han ayudado a superar cada reto u obstáculo que he tenido que enfrentar a lo largo de mi vida personal y durante mi formación profesional.

Gracias a todas las personas que forman parte de mi vida, gracias a su apoyo incondicional, gracias por creer en mí y en mis habilidades, ustedes me han dado fuerzas para seguir siempre adelante y nunca flaquear.

**GABRIELA TRUJILLO**

Mis más sinceros agradecimientos a mi familia, por el apoyo brindado durante la realización de este proyecto.

A los profesores y compañeros que aportaron con sus conocimientos para llevar a cabo de manera óptima cada una de las etapas de este proyecto.

**RONALDO CAMPUZANO**

## **DEDICATORIA**

El presente proyecto lo dedico principalmente a mi familia, por brindarme su amor y motivación para seguir adelante en cualquier meta que me proponga.

A todos quienes se sientan atraídos por los conocimientos expuestos en este trabajo para de alguna manera motivarlos a seguir esta hermosa rama de la ingeniería, Telemática.

**GABRIELA TRUJILLO**

El presente proyecto lo dedico a mi familia, por apoyarme en todo momento y brindarme el aliento necesario para alcanzar mis objetivos.

**RONALDO CAMPUZANO**

## TRIBUNAL DE EVALUACIÓN

.....  
**Rebeca Estrada, Ph. D.**

PROFESOR EVALUADOR

.....  
**M. Sc. Wendy Yáñez**

PROFESOR EVALUADOR

## DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y damos nuestro consentimiento para que la ESPOl realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

.....  
Gabriela Trujillo

.....  
Ronaldo Campuzano

## RESUMEN

Este proyecto está orientado a resolver los diversos problemas que se generan debido a la falta de conocimiento de lugares disponibles para parquear dentro de los estacionamientos con mayor afluencia de la ESPOL, siendo estos FIEC, CELEX y FICMP. Dado que los usuarios de los estacionamientos no poseen ningún medio de información, que les permita conocer el número de parqueos disponibles dentro de los estacionamientos, se generan problemas como retraso de actividades, congestión vehicular, gasto innecesario de combustible, y esto a su vez coadyuva a la contaminación ambiental.

Para resolver el problema, este proyecto se plantea la creación de un sistema que permita a los usuarios conocer y proveer información sobre el estado de los lugares destinados a estacionar los vehículos dentro de las facultades mencionadas, debido al alto número de personas que día a día usan los estacionamientos. El sistema utiliza una técnica basada en el concepto de Mobile Crowdsensing oportunista, la cual será usada como pieza fundamental en el desarrollo del mismo. Esta metodología incluye el desarrollo y la aplicación de conceptos como Internet de las cosas, aplicaciones móviles, computación ubicua, entre otros. El sistema implementa una aplicación móvil para dispositivos Android que interactúa con el usuario y es mediante ella, que la información necesaria es recolectada, enviada y recibida en tiempo real para mantener el monitoreo y control de los espacios. Además, el análisis de la información recibida es realizado por un servidor central el cual se encarga de ejecutar los algoritmos necesarios para que los datos recolectados sean aprovechados como fuente principal de información y funcione óptimamente.

Las pruebas realizadas, los resultados, y opiniones proporcionadas por parte de los usuarios que utilizan el sistema, son las que aportan con retroalimentación para futuros proyectos y mejoras sin alterar el concepto principal usado: crowdsensing.

## ÍNDICE GENERAL

AGRADECIMIENTO .....	ii
DEDICATORIA .....	iii
TRIBUNAL DE EVALUACIÓN .....	iv
DECLARACIÓN EXPRESA .....	v
RESUMEN .....	vi
ÍNDICE GENERAL.....	vii
CAPÍTULO 1 .....	1
1 PROPUESTA .....	1
1.1 Antecedentes.....	1
1.2 Descripción del problema .....	2
1.3 Justificación .....	2
1.4 Posibles soluciones .....	3
1.5 Solución propuesta.....	4
1.6 Objetivos.....	5
1.6.1 Objetivo General.....	5
1.6.2 Objetivos Específicos .....	5
1.7 Alcances y Limitaciones .....	6
CAPÍTULO 2.....	7
2 MARCO TEÓRICO DE METODOLOGÍA DE LA SOLUCIÓN PROPUESTA.....	7
2.1 Metodología.....	7
2.2 Tecnologías para el desarrollo .....	7
2.2.1 Crowdsensing.....	7
2.2.2 Geolocalización .....	9

2.3	Sistemas de parqueo existentes.....	10
2.3.1	Smart Parking System.....	10
2.3.2	SmartApp.....	11
2.4	Herramientas de software.....	13
2.4.1	PHP.....	13
2.4.2	MySQL.....	14
2.4.3	Geofencing y Geofence API.....	14
2.4.4	Activity Recognition API.....	15
2.5	Herramientas de hardware.....	15
2.5.1	GPS.....	15
2.5.2	Acelerómetro.....	15
2.5.3	Giroscopio.....	16
CAPÍTULO 3.....		17
3	DESCRIPCIÓN DE LA SOLUCIÓN.....	17
3.1	Descripción General.....	17
3.2	Arquitectura.....	17
3.2.1	Requerimientos mínimos del usuario.....	19
3.3	Datos Preliminares.....	19
3.3.1	Lugares Disponibles.....	19
3.3.2	Delimitación Geográfica de los Estacionamientos.....	20
3.4	Proceso de ejecución.....	22
3.4.1	Adquisición de datos.....	22
3.4.2	Conexión al servidor.....	23
3.4.3	Algoritmo de análisis.....	24
3.4.4	Consulta de estado actual.....	25
CAPÍTULO 4.....		26
4	PRUEBAS Y RESULTADOS.....	26

4.1	Pruebas .....	26
4.1.1	Escenario 1.....	26
4.1.2	Escenario 2.....	27
4.2	Resultados de las Encuestas .....	32
4.2.1	Encuesta FIEC .....	32
4.2.2	Encuesta CELEX.....	32
4.2.3	Encuesta FIMCP .....	33
4.2.4	Resultados Generales .....	33
	CONCLUSIONES Y RECOMENDACIONES .....	35
	BIBLIOGRAFÍA.....	37
	ANEXOS.....	40
	ANEXO A: ENCUESTA REALIZADA A LOS USUARIOS DE LA APLICACIÓN MÓVIL ESPOL PARKING.....	41
	ANEXO B: DOCUMENTACIÓN DE LA APLICACIÓN MÓVIL “ESPOL- PARKING” .....	42
	ANEXO C: DESCRIPCIÓN DE LOS SCRIPTS .....	46
	ANEXO D: REPOSITORIO DIGITAL.....	47

# CAPÍTULO 1

## 1 PROPUESTA

### 1.1 Antecedentes

La tecnología que ha surgido durante los últimos años, ha dado apertura para que proyectos que antes se veían mermados en posibilidades puedan ser implementados. Los conceptos que actualmente son usados para dar soluciones a problemas comunes no son ajenos a lo que previamente se había establecido, por lo que resulta sencillo de comprender la estructura y la lógica de los mismos.

Se estima que aproximadamente tres cuartas partes de la población posee un teléfono celular. En Ecuador por ejemplo, hasta el año 2016, un 89% de la población tiene una línea móvil activada, esto es un estimado de 14,85 millones de líneas, de las cuales 7,30 millones están asociadas a un Smartphone, según la ARCOTEL [1]. Gracias a la evolución de los mismos a lo largo de los años, los usuarios son capaces de acceder a todo tipo de información y servicios, siempre que dicho usuario cuente con acceso a una conexión de Internet.

Los teléfonos celulares modernos traen embebidos un conjunto de dispositivos que recopilan información tanto interna como externa que puede ser útil para tomar decisiones instantáneas de acuerdo a las necesidades de los usuarios[2]. La forma de aprovechar los datos que estos dispositivos embebidos miden es a través de aplicaciones móviles, las cuales son interfaces de interacción entre el usuario y todo el hardware del dispositivo móvil [3].

Las aplicaciones móviles facilitan la portabilidad de software y permiten tener control sobre el dispositivo. Se facilita el uso de los sensores internos del dispositivo móvil, manipulación de los datos que estos sensores proporcionan, y el análisis de dichos datos para poder tomar decisiones en determinadas situaciones, haciendo uso de una aplicación que bajo los permisos correspondientes sea ejecutada por el usuario de manera automática [4].

## **1.2 Descripción del problema**

El número de personas con vehículos que circula dentro de campus Gustavo Galindo de la ESPOL ha incrementado significativamente en los últimos años y con esto la falta de disponibilidad de lugares para estacionar dichos vehículos. Este campus cuenta con lugares de estacionamiento libre que pueden ser usados por estudiantes y visitantes y que se encuentran distribuidos por bloques o por facultad, es decir, existe al menos uno por facultad. Sin embargo, existe malestar por parte de quienes ingresan al campus y al no encontrar algún lugar disponible cerca del lugar donde precisan realizar sus tareas, se retrasa el inicio de sus actividades. Este inconveniente genera otras complicaciones tales como fuerte congestión vehicular en horas de mayor afluencia de vehículos que circulan por el campus, retraso del inicio de actividades, gasto innecesario de combustible al recorrer las diferentes áreas para poder encontrar un lugar.

## **1.3 Justificación**

La búsqueda de lugares libres en los estacionamientos de la universidad es una de las tareas más molestas con la que gran parte de los estudiantes lidian diariamente. Por esta razón, se desea implementar un servicio colaborativo en el cual los usuarios de dicho servicio sean quienes proveen información acerca de una situación en específico a través de dispositivos móviles, esta técnica llamada crowdsensing se encuentra basada en el paradigma “el poder de la multitud” el cual, en conjunto con las capacidades de detección de diversos dispositivos móviles, brinda la posibilidad de adquirir conocimientos que posteriormente pueden ser administrados para brindar un servicio que ayude a la comunidad.

Actualmente los estacionamientos más utilizados por los estudiantes, debido a su tamaño y ubicación estratégica, son los de FIEC, FIMCP y CELEX; los cuales en conjunto suman alrededor de 320 lugares para estacionar los autos. Estos datos otorgan un punto de partida para implementar un sistema que ayude a monitorear y determinar los lugares libres para parquear en los estacionamientos de mayor concurrencia.

#### 1.4 Posibles soluciones

Para resolver este problema se creará una aplicación móvil en la que los usuarios recibirán una notificación al ingresar al campus, la cual contendrá información acerca de cuantos lugares disponibles existen dentro del campus y donde pueden ser encontrados.

Se deberá validar por medio de posicionamiento GPS el momento en que el usuario se encuentra dentro del campus. Por otro lado, la aplicación desarrollada accederá a una base de datos la cual trabaja con datos en tiempo real sobre el número de parqueos disponibles y su respectiva ubicación. Para la validación de lugares disponibles en cada estacionamiento se han establecido las siguientes alternativas:

- El uso de sensores infrarrojos que monitoreen la entrada y salida de vehículos, dicha información será procesada con la ayuda de un Arduino y enviada hacia la base de datos a través de un módulo WIFI haciendo uso de la red de la ESPOL.

<b>Materiales</b>	<b>Precio</b>
Sensor Infrarrojo x2	\$100
Arduino UNO	\$25
Módulo WIFI ESP8266	\$11

**Tabla 1: Precio de solución por infrarrojos**

- Uso de sensores ultrasónicos en cada lugar disponible, al igual que la propuesta anterior dicha información será procesada con la ayuda de un Arduino y enviada hacia la base de datos a través de un módulo WIFI haciendo uso de la red de la ESPOL.

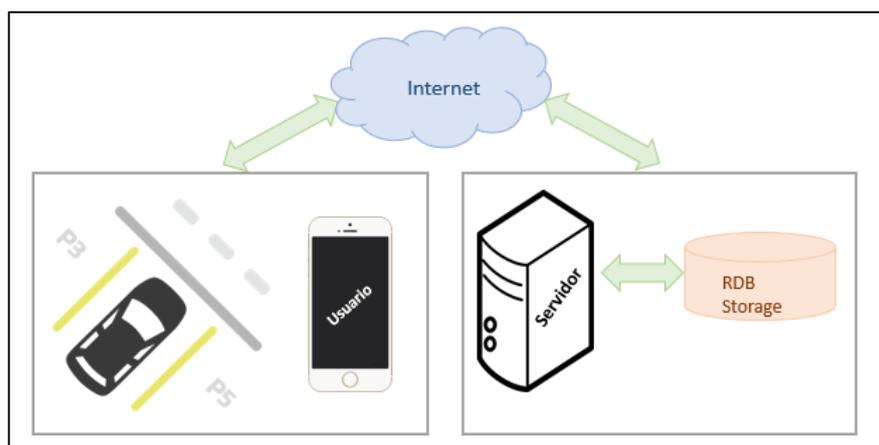
<b>Materiales</b>	<b>Precio</b>
Sensor ultrasónico	\$5
Arduino UNO	\$25
Módulo WIFI ESP8266	\$11

**Tabla 2: Precio de solución por ultrasonido**

- Realizar un “registro de parqueo” por medio de la aplicación para que notifique al servidor que un lugar ha sido ocupado y así mantener actualizada la base de datos.
- Implementación de una red de sensores dentro de cada estacionamiento que envíen el estado de los parqueos para actualizar automáticamente la información en la base de datos.

### 1.5 Solución propuesta

El sistema propuesto permitirá conocer la información requerida por parte de los usuarios y así mismo hacer que los usuarios retroalimenten con información sobre el estado de los parqueos disponibles, mediante la modalidad de crowdsensing para lo cual es necesario que el sistema cuente con los siguientes componentes mostrados en la figura 1.



**Figura 1.1: Esquema de la solución propuesta**

Los datos se obtienen directamente desde los usuarios por medio de eventos que genera la aplicación dependiendo de su localización GPS, para lo cual es necesario que el dispositivo tenga sistema operativo Android y con la opción de ubicación automática activada. La geolocalización por GPS es primordial para la obtención de datos y por ende es muy importante para mantener en óptimas condiciones y con información consistente el sistema.

El servidor obtiene los datos recopilados por los usuarios y actualiza su base de datos de acuerdo a los eventos que ocurren en los estacionamientos y son

informados por los clientes. Este servidor también es el encargado de responder a las solicitudes de los usuarios cuando se realice la búsqueda de parqueos disponibles.

La aplicación es interactiva e informativa sobre la cantidad de parqueos disponibles y su ubicación dentro de los tres principales estacionamientos del campus Gustavo Galindo de la ESPOL, tiene las siguientes características y funciones principales:

- Determinar lugares de parqueo disponibles y no disponibles en los tres estacionamientos principales de la ESPOL, estos son: FIEC, FIMCP y CELEX, a través del uso de GPS y el envío de información al servidor central. Esta característica no corresponde a la modalidad de crowdsensing.
- Notificar a los usuarios sobre los estacionamientos que cuenten con lugares disponibles para que puedan planificar sus actividades con anticipación.

## **1.6 Objetivos**

### **1.6.1 Objetivo General**

Crear un sistema automatizado que permita conocer el número y la ubicación de parqueos disponibles dentro del campus Gustavo Galindo de ESPOL, a través de una aplicación móvil que interactúe con el usuario para mejorar la movilidad vehicular y optimizar la búsqueda de estacionamientos.

### **1.6.2 Objetivos Específicos**

- Desarrollar una aplicación móvil que interactúe con el usuario realizando consultas y actualizaciones de los parqueos disponibles en tiempo real.
- Controlar el estado de los parqueos en cada estacionamiento para notificar a los usuarios sobre los respectivos cambios que se generen.
- Diseñar una red de parqueos inteligentes que permita monitorear los espacios disponibles en cada facultad.

## 1.7 Alcances y Limitaciones

- Limitación del Sistema operativo: La aplicación está desarrollada únicamente para dispositivos Android, debido a la apertura a usar los sensores de estos dispositivos.
- Alcance geográfico: Como parte de una mejora a proyectos pasados, se procederá a implementar este sistema en los tres principales estacionamientos del campus, estos son: CELEX, FIEC y FIMCP.
- Uso de datos móviles: Debido a que aún existen zonas donde el servicio de internet inalámbrico en el campus es deficiente o nulo, se prevé que en algunas ocasiones se necesitará hacer uso de los datos móviles del usuario. Esto es considerado como una gran limitante debido a que solo el 5% de los usuarios de Smartphones tienen paquetes postpago de datos móviles [1].

## CAPÍTULO 2

### 2 MARCO TEÓRICO DE METODOLOGÍA DE LA SOLUCIÓN PROPUESTA.

#### 2.1 Metodología

- La aplicación móvil será desarrollada para dispositivos Android que dispongan de una conexión a Internet activa que permita el intercambio de información entre el servidor central y el dispositivo móvil.
- El estado de los parqueos se lo determinará en base a la información enviada por los usuarios de manera automática, es decir, no se necesita la participación explícita de los usuarios para actualizar el estado de los parqueos.
- Será necesario que el usuario otorgue los permisos correspondientes para que la aplicación pueda obtener información útil que permita mantener la consistencia de los datos.
- Mediante el dispositivo móvil será posible determinar la ubicación del usuario y su estado de movimiento, lo que permitirá determinar si un espacio de parqueo fue liberado u ocupado.
- El servidor central se encargará de realizar las operaciones necesarias para determinar el estado de los parqueaderos. A partir de esto se enviará datos actualizados a la aplicación del usuario sobre cuántos parqueos libres existen en los estacionamientos de las facultades mencionadas.
- La red de parqueos estará compuesta por tres estacionamientos: CELEX, FIEC y FIMCP. Las tres áreas estarán siendo monitoreadas por el servidor central en caso de existir algún cambio dentro de las mismas.

#### 2.2 Tecnologías para el desarrollo

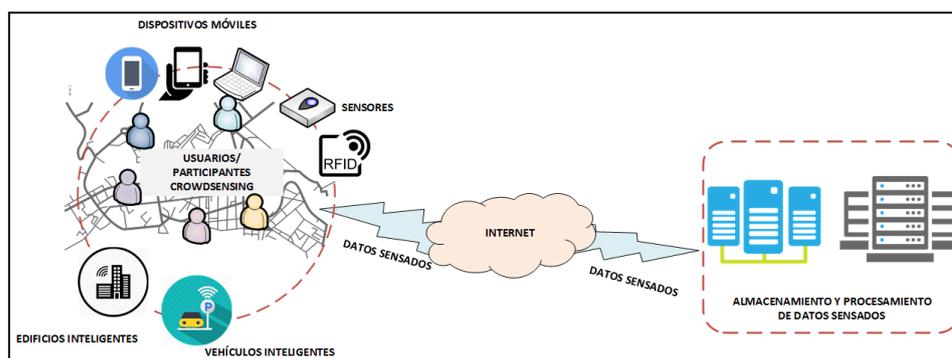
##### 2.2.1 Crowdsensing

*Mobile crowdsensing* surge como un nuevo paradigma basado en el poder de la multitud conjuntamente con la capacidad de los sensores de

varios dispositivos móviles, tales como teléfonos inteligentes, o dispositivos de uso común. Los dispositivos equipados con múltiples sensores, desde GPS hasta cámara y micrófono, junto con la movilidad de los usuarios permiten adquirir información del ambiente que los rodea a través de sus sensores [5]

Esta técnica consiste en tomar la participación humana como base para obtener datos que pueden ser usados por parte de la entidad que los solicita para tomar medidas al respecto. Estos datos pueden ser tan simples como estado del clima, o tan complejos como niveles de contaminación en el ambiente, ruido ambiental o advertencias de tránsito entre usuarios.

Una arquitectura de cómo luce la modalidad de crowdsensing se ilustra en la figura 2.1, donde se puede apreciar que los usuarios participantes son quienes proveen información del medio mediante los dispositivos móviles y demás sensores que pueden ser ligados a Internet de las cosas (IoT, por sus siglas en inglés). En el lado derecho de la imagen es posible observar que, a través de una conexión vía Internet, se encuentran conectados los equipos encargados de procesar, analizar, y almacenar todos los datos que se envíen al sistema.



**Figura 2.1: Funcionamiento general de un sistema crowdsensing**

Existen dos tipos de obtención de datos vía crowdsensing que dependen exclusivamente del grado de participación que el usuario tiene al

momento de enviar la información que se solicita, estos son: participativo y oportunista.

Participativo: El usuario es quien voluntariamente emite la información requerida.

Oportunista: los datos son obtenidos y compartidos automáticamente sin la intervención del usuario y en muchos casos sin el conocimiento explícito del mismo.

Mobile Crowdsensing ocurre en tres etapas: adquisición, almacenamiento y envío de datos, siendo la más importante la adquisición de datos debido a que aprovecha los sensores disponibles en el dispositivo a través del paradigma de Internet de las Cosas. Esta adquisición puede ser de tipo manual, a través de la cual el usuario toma fotos o usa aplicaciones del teléfono; puede ser controlada por el usuario, es decir, el usuario decide cuando recoger la información, pero no cómo; y puede ser automática cuando una aplicación se encarga de la recopilación de información bajo alguna circunstancia predefinida, por ejemplo, recopilar información ambiental cuando la temperatura supere un valor [6].

### **2.2.2 Geolocalización**

La geolocalización es el posicionamiento que define la localización de un dispositivo, en un sistema de coordenadas dentro de nuestro planeta. Este proceso es empleado por los sistemas de información geográfica, que se encuentran diseñados especialmente para capturar, almacenar, manipular y analizar la información geográfica referenciada [7].

Para la geolocalización existen varias alternativas que permiten conocer la ubicación, entre ella el uso de celulares debido a que los usuarios portan sus dispositivos todo el tiempo, permitiendo fácilmente conocer su ubicación y actualizarla a medida que se movilizan [8].

El servicio de GPS de los dispositivos móviles es de gran importancia a la hora de permitirle a una persona saber cómo llegar o ubicar un lugar determinado del territorio. El GPS fue un pionero y abrió paso luego a nuevas propuestas la geolocalización.

Entre el amplio abanico de opciones que permiten descubrir la geolocalización de un dispositivo se destacan los teléfonos móviles de alta gama. En particular aquellos que traen integrados receptores de GPS, los cuales y gracias a la red de satélites que rodea al planeta podrán localizar a su portador donde sea que se encuentre. [7]

### **2.3 Sistemas de parqueo existentes**

Actualmente alrededor del mundo, especialmente en países desarrollados, existen sistemas y aplicativos que, bajo el concepto de *Smart Cities*, implementan el control de los lugares de estacionamientos en la calle (denominados *on-street*) y en lotes o garajes destinados a brindar el servicio de parqueo de vehículos (llamado también parqueo *off-street*).

Dentro de toda la gama de sistemas que brindan este servicio de parqueo inteligente, existen dos que acaparan gran mercado debido a la variedad de soluciones que ofrecen a los clientes: Smart Parking System y SmartApp.

#### **2.3.1 Smart Parking System**

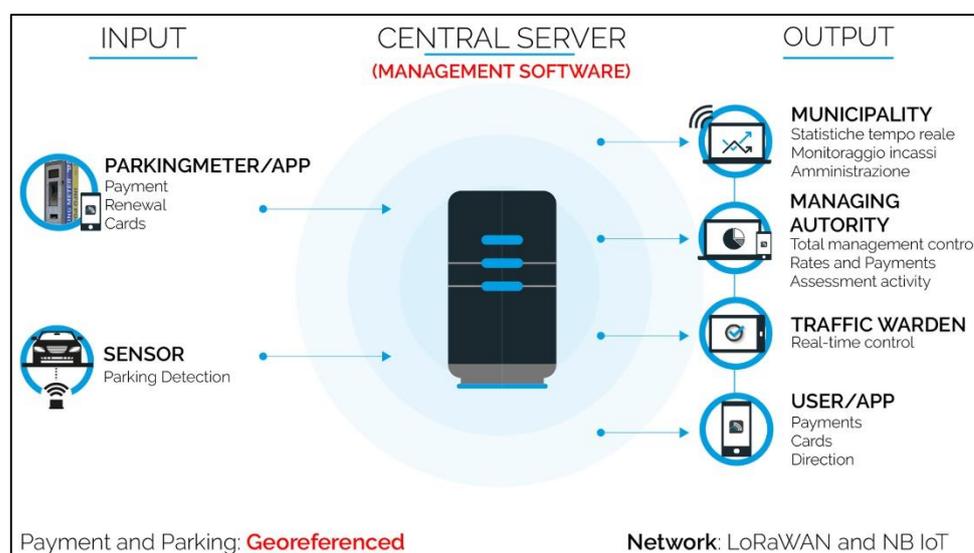
Este sistema fue desarrollado por la empresa italiana INTERCOM S.P.A. y brinda una solución completa, fácil de administrar, y agradable para el usuario, que permita controlar y notificacar sobre parqueos *on-street* dentro de un área en específico o de una ciudad completa [9].

El sistema consta de algunos elementos clave, tal como se ilustra en la figura 2.2:

- Sensores ultrasonido ubicados dentro debajo del suelo en cada lugar de manera individual.
- Parquímetro o Aplicación móvil 4Park encargados de realizar el cobro por el tiempo que el usuario deja su vehículo estacionado. La

aplicación además muestra información en tiempo real y permite administrar pagos y además es capaz de guiar al usuario al estacionamiento que él elija.

- Servidor central, el cual toma la información de los sensores, del parquímetro o de la aplicación móvil, a través de red de enlaces radiales
- Red de comunicación implementada bajo protocolos de comunicación conocidos, ya sea LoRaWAN, LoRa INTERCOMP, o NarrowBand-IoT.
- Medios de salida que permiten el monitoreo y administración de la información.



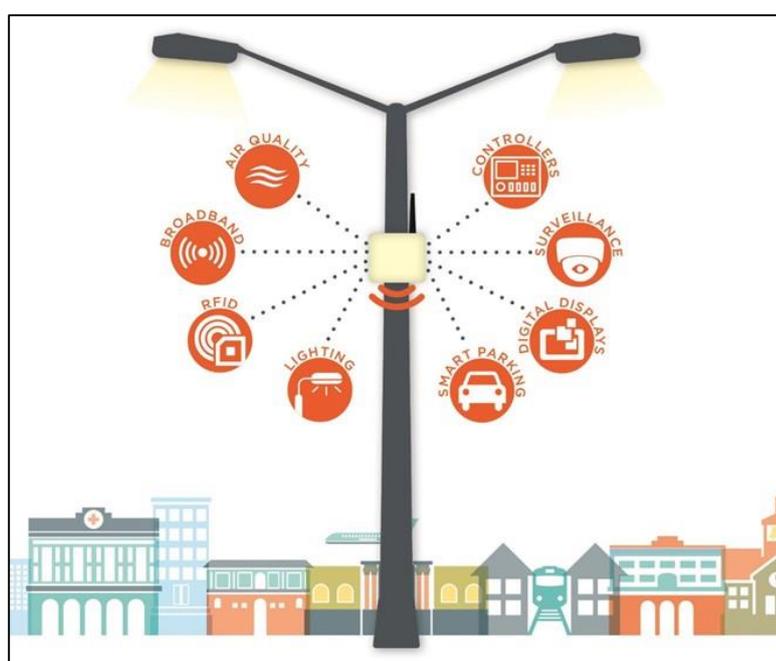
**Figura 2.2: Arquitectura de Smart Parking System [9].**

### 2.3.2 SmartApp

Es una aplicación móvil desarrollada por la empresa australiana Smart Parking Limited. Esta aplicación forma parte de un sistema que la empresa vende a las ciudades de Australia y Nueva Zelanda como solución para la modernización del control de estacionamientos *on-street* y *off-street*. Las ventajas que ofrece esta compañía con su sistema y su aplicación móvil es la variedad de métodos de toma de información, es decir, el cliente es quien decide el tipo de red a usar de acuerdo a sus necesidades.

Dentro de las soluciones que ofrece, se encuentran sensores ultrasonidos en el suelo, por lo alto, etiquetas RFID dentro de cada vehículo, y parquímetros inteligentes.

Para enlazar los sensores se utiliza una puerta de enlace denominada SmartSpot Gateway tal como se esquematiza en la figura 2.3. Este gateway además es compatible con los estándares más conocidos de comunicación inalámbrica lo cual facilita la integración de otros servicios a través de él [10].



**Figura 2.3: Uso del SmartSpot Gateway como parte de una implementación para *Smart Cities* [11].**

La infraestructura de red que usa se basa en estándares como Zigbee, LoRaWAN, 3G, 4G, y 5G; mientras que el monitoreo y manejo de recursos se lo hace por medio de los servicios privados de la compañía llamados SmartCloud y SmartRep. De parte del usuario, la información llega a través de la aplicación móvil SmartApp.

En la figura 2.4 se muestra un ejemplo real en la ciudad de Wellington, Nueva Zelanda donde la compañía instaló alrededor de 3800 sensores en el suelo o llamados *in-ground* a lo largo y ancho de la ciudad [12]



**Figura 2.4: Implementación del sistema de parqueaderos por medio de sensores *in-ground* en Wellington, Nueva Zelanda [12].**

## 2.4 Herramientas de software

### 2.4.1 PHP

PHP es un lenguaje de código abierto adecuado especialmente para el desarrollo web. Lo que distingue a PHP de algo del lado del cliente es que el código es ejecutado en el servidor, generando objetos HTML y enviándolo al cliente únicamente el resultado de ejecutar el script. El

servidor web puede ser configurado también para que procese todos los ficheros HTML con PHP, sin la necesidad que los usuarios tengan conocimiento [13].

PHP es utilizado en el proyecto para ejecutar el algoritmo principal que valida los lugares disponibles y modifica los datos almacenados que son presentados al usuario.

#### **2.4.2 MySQL**

MySQL es la base de datos de código abierto más popular del mundo. Con su desempeño comprobado, confiabilidad, y facilidad de uso, MySQL se ha convertido en la base de datos que encabeza la lista de opciones para el desarrollo de aplicaciones web [14].

Se usa MySQL en el sistema con la finalidad de mantener guardados los datos de ingreso y salida de vehículos a los parqueaderos, así como los valores actualizados de los lugares disponibles que son consultados por la aplicación móvil y mostrados al usuario.

#### **2.4.3 Geofencing y Geofence API**

*Geofencing* es una característica dentro de un programa de software que usa el sistema de posicionamiento global (GPS) o el identificador de radio frecuencia (RFID) para definir límites geográficos.

Esta característica permite a un administrador configurar triggers de tal manera que se activen cuando un dispositivo entre o salga de los límites geográficos delimitados por el administrador.

Las barreras virtuales pueden ser activas o pasivas. Las activas requieren un usuario final para habilitar los servicios de localización y una aplicación móvil debe estar abierta. Las pasivas siempre están habilitadas, es decir, aprovechan la conexión Wi-Fi o datos móviles mientras trabajan en segundo plano [15].

El uso de Geofence API dentro del proyecto es fundamental ya que delimita las zonas sobre las cuales se realizará el análisis basado en la geolocalización y el reconocimiento de la actividad.

#### **2.4.4 Activity Recognition API**

El término Activity Recognition abarca todas las áreas de investigación sobre cómo reconocer las acciones y objetivos de uno o más agentes desde una serie de observaciones de las acciones de los agentes y las condiciones del medio en el que se encuentran [16].

Google plantea el Activity Recognition API el cual es una interfaz que utiliza los sensores acelerómetro y giroscopio para determina el tipo de actividad en la que el agente (en este caso el dispositivo móvil) se encuentra participando sin la necesidad de tomar información individual de cada sensor y analizarlos para llegar a una conclusión. El API retorna la actividad que fue detectada junto con un factor de confiabilidad de dicha información [17].

### **2.5 Herramientas de hardware**

#### **2.5.1 GPS**

El GPS es un sistema que permite determinar la localización de los dispositivos terminales dentro del planeta tierra gracias a los 24 satélites que rodean la atmósfera.

El uso y masificación del GPS está particularmente extendido en los teléfonos móviles inteligentes, lo que ha hecho surgir todo un ecosistema de software para este tipo de dispositivos, así como nuevos modelos de negocios que van desde el uso del terminal móvil para la navegación tradicional punto-a-punto hasta la prestación de los llamados Servicios Basados en la Localización [18].

#### **2.5.2 Acelerómetro**

Es un sensor responsable de detectar cambios de orientación con la finalidad que los dispositivos puedan rotar la pantalla hacia posiciones

vertical u horizontal. Además, es usado también para calcular distancias y velocidades complementado con el giroscopio.

Entre los más comunes, se tiene los capacitivos, los cuales están compuestos por dos placas metálicas, una fija y otra móvil, y un dieléctrico entre ellas. Cuando se produce una aceleración en el dispositivo, el teléfono la detecta mediante el cambio de capacidad del capacitor al cambiar de posición la placa móvil. La energía también cambia y es cuantificada y procesada por un chip que determina la aceleración que el dispositivo ha adquirido [19].

### **2.5.3 Giroscopio**

El giroscopio es un sensor que mide las acciones y movimientos de un dispositivo con un brazo de accionamiento sobre el cual se produce una vibración vertical cuando el dispositivo rota. La parte fija llamada estator se dobla, lo que genera movimiento sobre el brazo y haciendo posible que se calcule la velocidad angular para que finalmente el dispositivo reconozca la señal eléctrica de dicha velocidad y reconozca los cambios de posición en el espacio de sí mismo.

El giroscopio junto con el acelerómetro se encarga de determinar los cambios de posición y velocidad del dispositivo que los incluyen [19].

## CAPÍTULO 3

### 3 DESCRIPCIÓN DE LA SOLUCIÓN.

#### 3.1 Descripción General

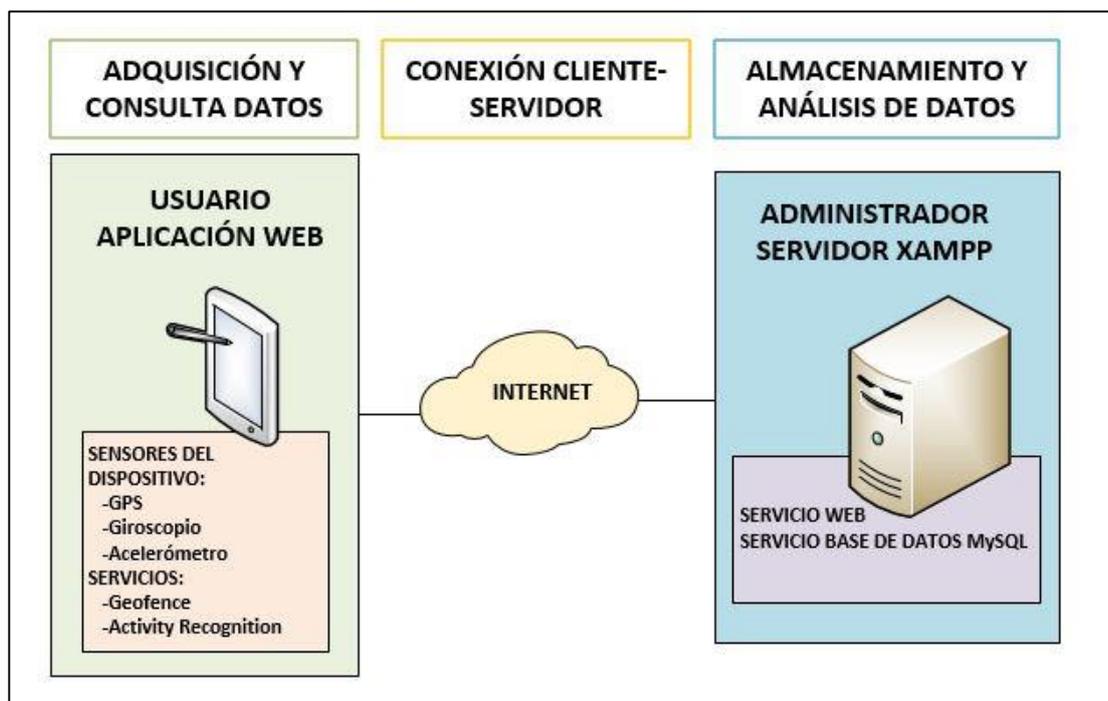
El sistema encargado de monitorear y notificar sobre los lugares disponibles en los tres parqueaderos mencionados en el capítulo 1, ha sido diseñado a través del uso de tecnologías que promueven la participación del usuario de manera oportuna, es decir, el sistema se encarga de tomar los datos que necesita por parte del usuario para funcionar correctamente.

Una aplicación móvil en Android es la encargada de tener contacto directo con el usuario. La aplicación tiene doble funcionalidad: 1) notificar al usuario y permitirle realizar consultas en tiempo real, 2) aportar al sistema con información del estado actual de los parqueaderos haciendo uso de los sensores del dispositivo para mantener actualizados los datos.

La información generada por los sensores es enviada al servidor de base de datos con el uso del lenguaje PHP para que posteriormente un algoritmo actualice y notifique en tiempo real el estado actual de los parqueaderos.

#### 3.2 Arquitectura

El sistema está dividido en tres partes que se comunican entre sí, transmitiendo datos necesarios para que funcione correctamente, tal como se muestra en la figura 3.1.



**Figura 3.1: Arquitectura del Sistema**

La etapa de adquisición y consulta de datos engloba el contacto directo con el usuario y la interacción que este tiene con el sistema, ya sea proporcionando datos o consultándolos. En esta parte se utilizan las tecnologías descritas en el capítulo 2 y es la parte más crítica del sistema ya que se requiere una toma de datos precisa para que el sistema funcione en condiciones óptimas.

Del otro lado se encuentra el servidor y la infraestructura brindada por quienes ofrecen el servicio. Esta etapa se encarga del almacenamiento de datos y del análisis y procesamiento de los mismos con la finalidad de que el algoritmo, que se lo presentará posteriormente, sea ejecutado con éxito y así mantener consistencia en la base de datos del sistema.

La conexión entre los clientes y el servidor central a través de Internet es lo que completa el sistema dando así lugar a que la transferencia de información sea exitosa, rápida y oportuna para cumplir con el objetivo de tener datos correctamente procesados en tiempo real.

### 3.2.1 Requerimientos mínimos del usuario

La parte que tiene contacto con el usuario es la aplicación móvil desarrollada en Android, por lo tanto, existen requisitos mínimos que los dispositivos deben cumplir para el correcto funcionamiento:

- API de Android tiene que ser superior a la versión 9, es decir, los dispositivos deben contar como mínimo con el sistema operativo Android 2.3 – Gingerbread.
- Android Build Tools debe ser superior a la versión 21, correspondiente al mismo Android 2.3
- Los dispositivos deben contar con los sensores GPS, Acelerómetro, y Giroscopio. El 98% de los dispositivos actuales cuentan con dichos sensores según el centro de soporte de Android.
- Para establecer conexión el dispositivo debe estar en red mediante Internet ya sea por datos móviles o Wi-Fi.

## 3.3 Datos Preliminares

### 3.3.1 Lugares Disponibles

Actualmente los parqueaderos de FIEC, FIMCP y CELEX cuentan con alrededor de 320 lugares disponibles para que los estudiantes y usuarios en general estacionen sus vehículos. Estos lugares están distribuidos de la siguiente manera:

- CELEX: 153 lugares disponibles
- FIEC: 122 lugares disponibles
- FIMCP: 45 lugares disponibles

Estos datos son los que se toman en cuenta al momento de poner en marcha el sistema. Los parqueos disponibles son los que aparecen en la pantalla principal de la aplicación móvil y los números aumentan o disminuyen dependiendo si se liberan o se ocupan respectivamente.

### 3.3.2 Delimitación Geográfica de los Estacionamientos

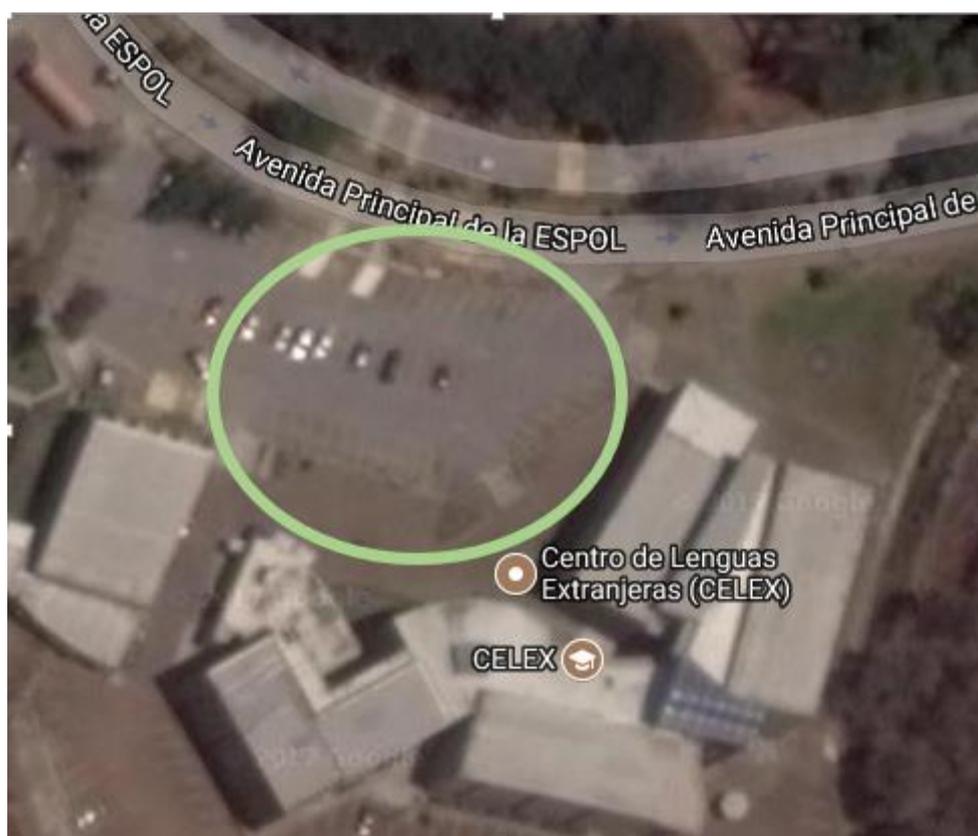
La delimitación geográfica de los parqueaderos se realiza con herramientas como Google Maps y Google Earth. Esta delimitación se basa en conocer las coordenadas del centro de cada estacionamiento, y el radio necesario para que toda el área que contenga lugares de estacionamiento sea cubierta.

Para parqueadero de CELEX fueron tomadas las siguientes mediciones

Latitud: -2.148693.

Longitud: -79.967702.

Radio: 70 metros.



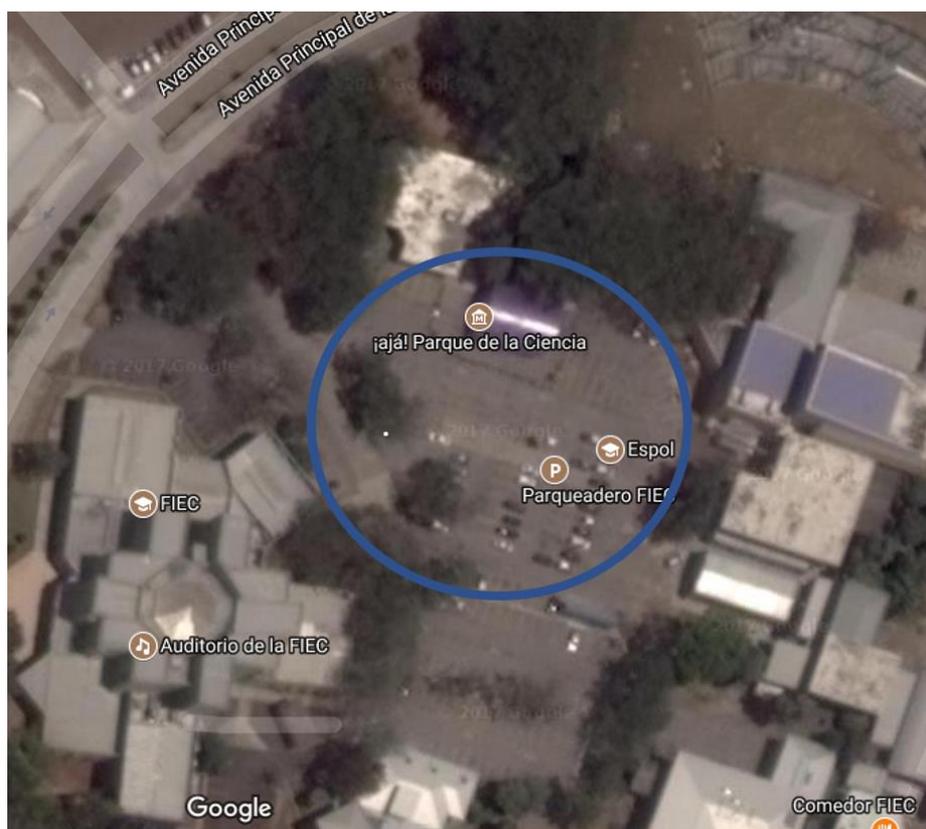
**Figura 3.2: Toma satelital del área delimitada para parqueadero de CELEX**

Para parqueadero de FIEC fueron tomadas las siguientes mediciones

Latitud: -2.144336.

Longitud: -79.967167.

Radio: 75 metros.



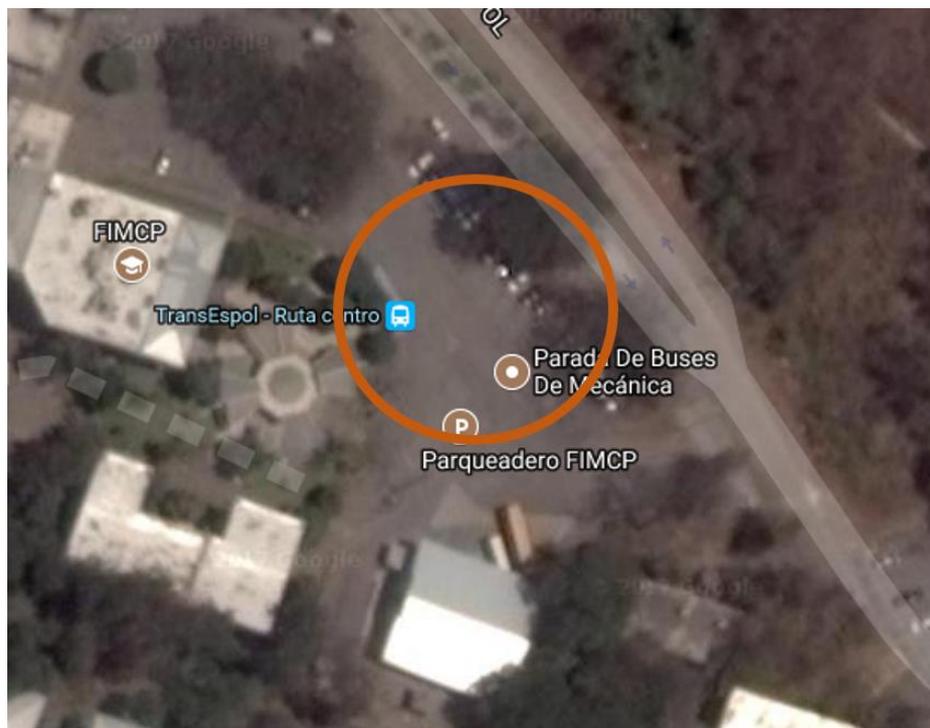
**Figura 3.3: Toma satelital del área delimitada para parqueadero de FIEC**

Para parqueadero de FIMCP fueron tomadas las siguientes mediciones

Latitud: -2.144636.

Longitud: -79.965323.

Radio: 35 metros



**Figura 3.4: Toma satelital del área delimitada para parqueadero de FIMCP**

### **3.4 Proceso de ejecución**

#### **3.4.1 Adquisición de datos**

La recolección de información es fundamental para que cualquier sistema sea eficaz y eficiente. En este caso los datos serán proporcionado por los sensores GPS, Acelerómetro y Giroscopio del dispositivo móvil. Dichos sensores son gestionados por dos APIs dentro de la aplicación: Geofence y Activity Recognition. Los datos que se envían al servidor son generados por estos dos APIs.

Geofence se encarga de generar eventos cuando el dispositivo ingresa o sale de una de las zonas de interés, es decir, al ingreso o salida del campus o de uno de los parqueaderos. Las zonas tienen un identificador asociado para que el servidor pueda reconocer a qué zona en específico

el usuario ha ingresado o salido y pueda mantener consistencia en los datos [20].

Reconocimiento de la actividad (Activity Recognition) contribuye a detectar la actividad que el usuario se encuentra realizando. Esta actividad es básicamente el estado de movimiento que tiene el usuario y los eventos que genera este API pueden estar entre estos tres casos: el usuario está en reposo, el usuario está moviéndose a pie (correr o caminar), el usuario está moviéndose en un vehículo. Este API se ejecuta como un servicio en segundo plano por lo que siempre está monitoreando la actividad que realiza el usuario [21].

Cuando se ingresa o sale de una zona delimitada por Geofence, se realiza la captura del estado de movimiento actual que proporciona el reconocimiento de la actividad, y ese conjunto de datos (de ambos APIs) son enviados al servidor que se encarga de analizar los estados de entrada y salida para actualizar la información en tiempo real.

La adquisición de datos, delimitación de áreas y todo lo referente al desarrollo de la aplicación se encuentran detallados en el Anexo B.

#### **3.4.2 Conexión al servidor**

El servidor comprende un servicio de base de datos es usado para almacenar la información del estado actual de los parqueaderos y de los estados de movimiento que se generan al ingreso de cada área.

La base de datos es MySQL la cual es administrada a través de instrucciones PHP que se ejecutan en la parte de servicio web servidor. Estas dos funcionalidades son logradas a través del software XAMPP que provee el soporte necesario para trabajar en conjunto ambos servicios.

La conexión al servidor es remota, es decir, el servidor no es local para cada dispositivo siguiendo la arquitectura centralizada mostrada anteriormente; contiene una IP pública y los datos pueden ser

consultados desde cualquier dispositivo móvil con la aplicación instalada y conexión a Internet.

### 3.4.3 Algoritmo de análisis

El análisis que realiza el servidor es mediante un algoritmo que se basa en la información que fue enviada por los dos APIs mencionados anteriormente. El servidor espera que lleguen los datos de ingreso a la zona y salida de la zona para analizar los estados de movimiento de cada usuario y generar eventos que representan los hechos que ocurrieron dentro de cada parqueadero.

La lógica de los eventos que el servidor genera está descrita en la tabla 3, la cual cubre todas las posibles combinaciones entre estados de movimiento al momento de entrar o salir de un parqueadero y el evento que el servidor genera de acuerdo a la información previa.

INGRESO	SALIDA	EVENTO CREADO
A PIE	A PIE	MANTIENE
A PIE	EN VEHÍCULO	DESCOUPAR PARQUEO
EN VEHÍCULO	A PIE	OCUPAR PARQUEO
EN VEHÍCULO	EN VEHÍCULO	MANTIENE

**Tabla 3: Lógica de comparación de estados de ingreso y salida.**

Estos eventos son utilizados para aumentar o disminuir el número de lugares disponibles en cada parqueadero. Así:

- Si el evento es **MANTIENE**, los lugares disponibles **NO CAMBIAN**.
- Si el evento es **DESCOUPAR PARQUEO**, los lugares disponibles **AUMENTAN**.
- Si el evento es **OCUPAR PARQUEO**, los lugares disponibles **DISMINUYEN**.

De esta forma la información de lugares disponibles se actualiza cada vez que se genera un evento de entrada y salida.

El algoritmo se encuentra plasmado en un script que se ejecuta en el servidor cada vez que sea necesario para mantener la información actualizada en tiempo real. La descripción del script se encuentra desarrollada en el Anexo C.

#### 3.4.4 Consulta de estado actual

La consulta del estado actual de los parqueaderos y sus lugares libres se realiza mediante consultas al servidor y son mostradas a través de la aplicación en la interfaz gráfica de la figura 3.5.



**Figura 3.5: Interfaz gráfica de la aplicación para consulta.**

El estado actual de los lugares libres se muestra en números grandes debajo del nombre de cada parqueadero al que le pertenece.

La actualización de los lugares disponibles es en tiempo real, es decir, apenas se detecta un cambio de acuerdo a los eventos previamente descritos, se procede a actualizar la base de datos y ese cambio es reflejado de inmediato en la interfaz de la aplicación.

## CAPÍTULO 4

### 4 PRUEBAS Y RESULTADOS

#### 4.1 Pruebas

##### 4.1.1 Escenario 1

El primer escenario contó con un solo usuario que realiza las dos acciones principales que detecta el sistema, es decir, estacionar su vehículo (ocupar un espacio), recoger su vehículo y salir del parqueadero (desocupar un espacio). La prueba fue hecha en el parqueadero de CELEX y el parqueo exitoso se ilustra en la figura 4.1 resaltando que el número de parqueos disponibles que se muestra en la aplicación disminuye en uno.

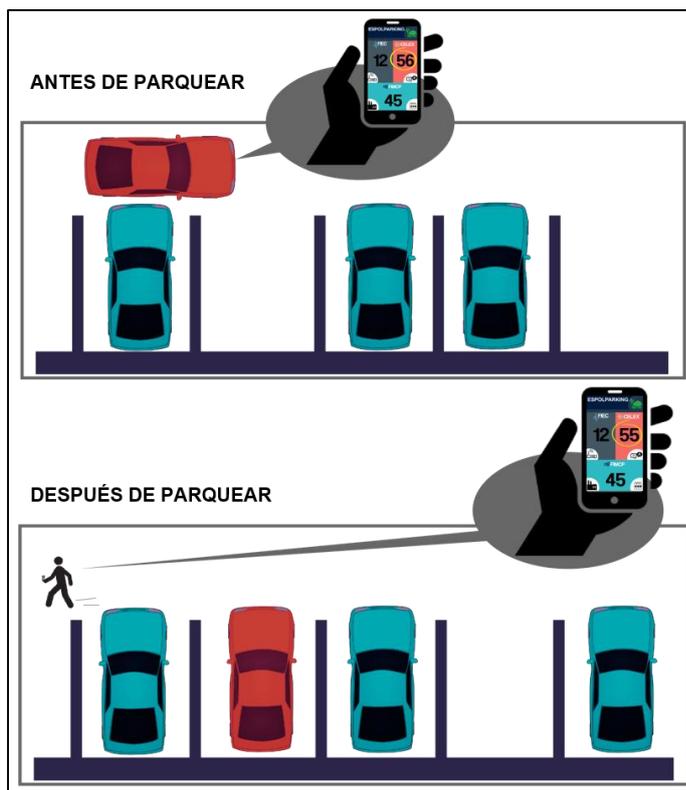
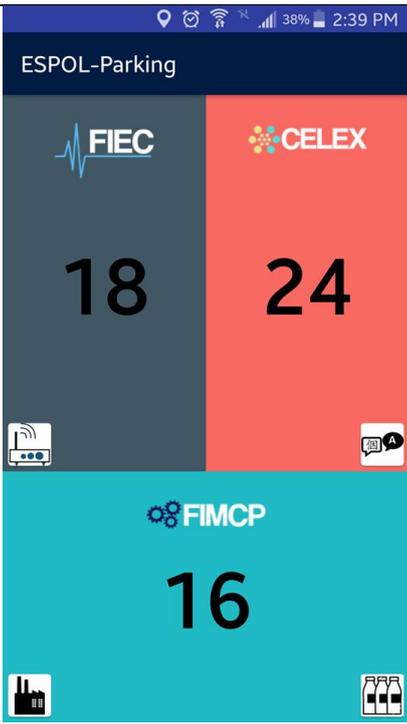


Figura 4.1: Esquema de un parqueo exitoso y disminución del número de parqueos disponibles

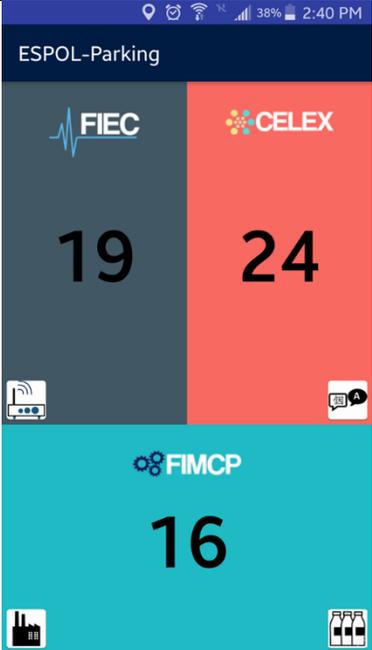
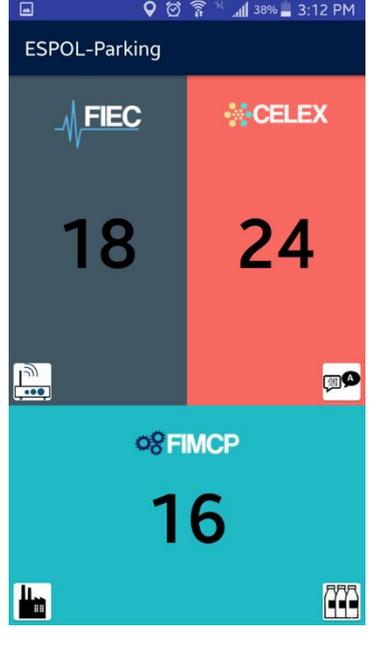
Los resultados de esta primera prueba fueron satisfactorios respecto a los esperados. Se demostró la factibilidad del sistema en tiempo real por lo que cumplió con los objetivos planteados.

#### 4.1.2 Escenario 2

La segunda prueba involucró diez usuarios que realizaron una de las dos acciones en el parqueadero de FIEC de la siguiente forma: del total de usuarios, 4 se estacionaron, 4 recogieron sus vehículos, y 2 estuvieron de paso. En la tabla 4 se muestran los estados de la aplicación, y por ende del sistema, en los diferentes instantes mientras duró la prueba. Son ocho acciones realizadas, puesto que dos vehículos no aportaron con cambios de los datos.

Hora	Acción Realizada	Estado del sistema
14:39	Estado inicial	 <p>The screenshot shows the 'ESPOL-Parking' application interface. At the top, there is a status bar with location, Wi-Fi, signal strength, 38% battery, and 2:39 PM. Below the title bar, there are three main data panels: a dark grey panel for 'FIEC' with the number '18', a red panel for 'CELEX' with the number '24', and a teal panel for 'FIMCP' with the number '16'. Each panel includes a small icon representing its category (a heart for FIEC, a gear for CELEX, and a factory for FIMCP). There are also small icons for Wi-Fi and a notification bubble at the bottom of the panels.</p>

**Tabla 4: Estado del sistema en cada momento de la prueba realizada (parte 1).**

Hora	Acción Realizada	Estado del sistema
14:40	Desocupar	
15:12	Estacionar	

**Tabla 4: Estado del sistema en cada momento de la prueba realizada (parte 2).**

Hora	Acción Realizada	Estado del sistema
15:13	Estacionar	
15:15	Estacionar	

Tabla 4: Estado del sistema en cada momento de la prueba realizada (parte 3).

Hora	Acción Realizada	Estado del sistema
15:24	Desocupar	
15:27	Estacionar	

Tabla 4: Estado del sistema en cada momento de la prueba realizada (parte 4).

Hora	Acción Realizada	Estado del sistema
15:58	Desocupar	
16:02	Desocupar/ Estado final	

**Tabla 4: Estado del sistema en cada momento de la prueba realizada (parte 5).**

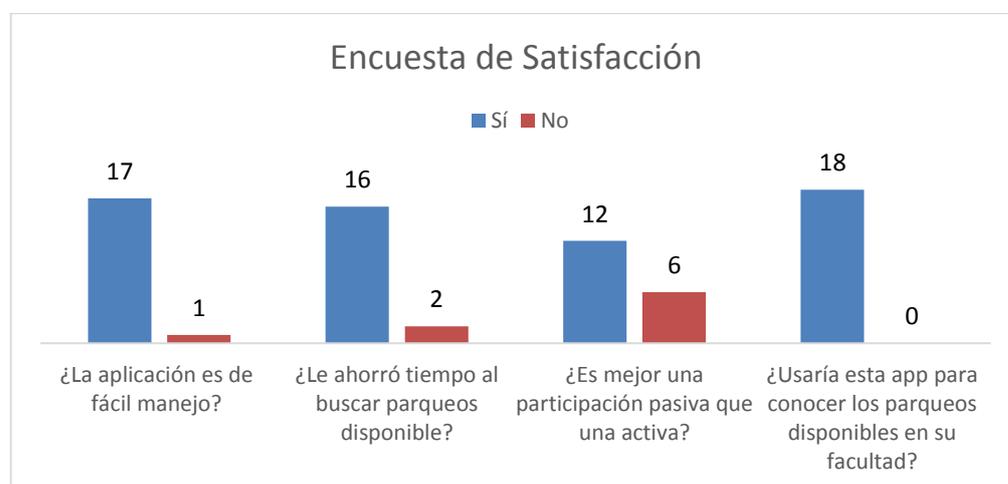
El objetivo de la segunda prueba fue comprobar la consistencia de los datos, al tener múltiples usuarios que interactúen con el sistema proporcionando diferente información simultánea, forzando al sistema a realizar varios análisis en tiempo real. Estos resultados permitieron observar el comportamiento del sistema a medida que el tiempo continuaba y fueron satisfactorios debido a que al final de la prueba los datos fueron 100% consistentes en base a lo previamente calculado.

## 4.2 Resultados de las Encuestas

Para medir el nivel de aceptación y satisfacción de la aplicación ESPOL PARKING y por consiguiente del sistema de control y monitoreo de parqueos, se tomó una muestra de 45 personas con vehículos y que día a día utilizan al menos uno de los tres estacionamientos que este proyecto analiza. El formato de la encuesta se detalla en el Anexo A.

### 4.2.1 Encuesta FIEC

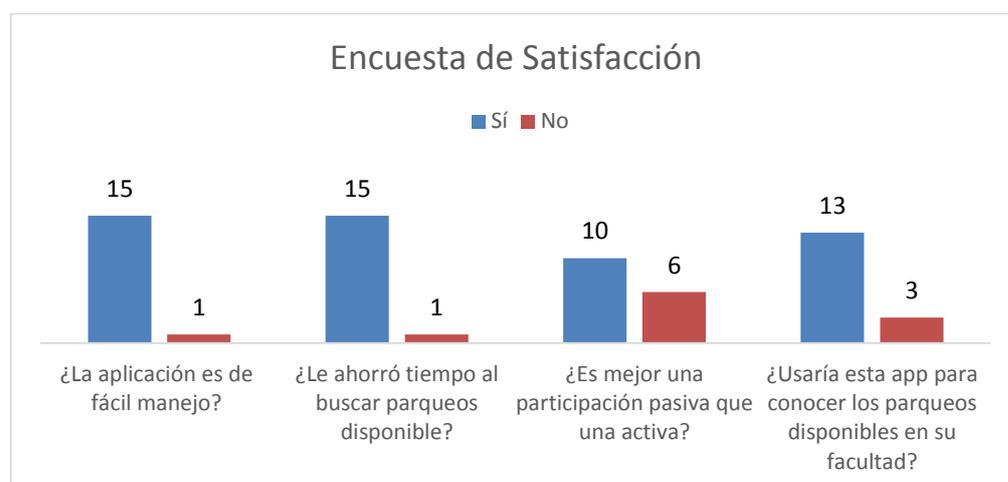
En FIEC las encuestas fueron para 18 personas que usaron la aplicación antes, durante y después de estacionar su vehículo.



**Figura 4.2: Resultados de las encuestas a usuarios de parqueadero FIEC**

### 4.2.2 Encuesta CELEX

En CELEX las encuestas fueron para 16 personas que usaron nuestra aplicación antes, durante y después de estacionar su vehículo.



**Figura 4.3: Resultados de las encuestas a usuarios de parqueadero CELEX**

#### 4.2.3 Encuesta FIMCP

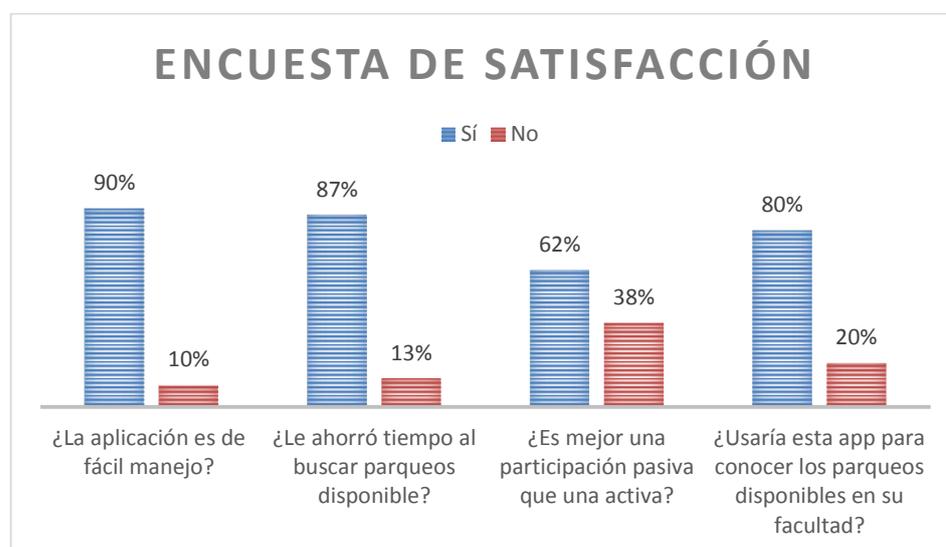
En FIMCP las encuestas fueron para 11 personas que usaron nuestra aplicación antes, durante y después de estacionar su vehículo.



**Figura 4.4: Resultados de la encuesta a usuarios de parqueadero FIMCP**

#### 4.2.4 Resultados Generales

De manera general se tiene una muestra de 45 usuarios cuyas respuestas están representadas en el diagrama de porcentajes.



**Figura 4.5: Resultados de la encuesta realizada a 45 usuarios.**

Esta encuesta permite identificar los puntos clave del uso de la aplicación y su futuro éxito cuando se empiece a distribuir masivamente. La utilidad de la aplicación está demostrada con el 87% de los usuarios que reconocieron que la aplicación y el sistema les ahorraron tiempo al momento de buscar un parqueo, y de igual manera el 80% de los encuestados usaría esta aplicación móvil a futuro para conocer los parqueos disponibles al ingresar al campus.

La pregunta clave fue sobre la modalidad de proporcionar información al sistema sobre el estado de los parqueos, obteniendo como resultado que el 62% de los encuestados sí prefieren una participación pasiva ante una activa. Esta decisión tomada por los usuarios responde al tiempo que les tomaría reportar al sistema que un parqueo ha sido ocupado o desocupado, por lo que preferirían que al estacionar su vehículo el sistema automáticamente conozca que dicha acción fue realizada, y es justamente eso lo que este sistema reconoce al momento que el usuario forma parte del mismo.

## CONCLUSIONES Y RECOMENDACIONES

El sistema notifica y controla del estado de los parqueos en tiempo real utilizando los datos proporcionados al momento de ingresar o salir de los estacionamientos.

La aplicación del concepto de crowdsensing en este proyecto permite asistir y ayudar a los conductores con información en tiempo real de los lugares de parqueos disponibles en los estacionamientos con mayor demanda, usando únicamente los dispositivos móviles de los usuarios.

Las encuestas realizadas revelan que la metodología crowdsensing oportunista cuenta con gran acogida por parte de los usuarios debido a su simplicidad de uso y liberación de responsabilidad activa de ellos para intercambiar información que al ser obtenida del medio automáticamente puede ser ingresada al sistema y realizar los análisis y procedimientos pertinentes.

Los sensores interconectados en teléfonos inteligentes y la gran cantidad de dichos dispositivos en el mercado, permiten que la información proporcionada por los sensores sea más precisa dando paso a la creación de aplicativos que utilicen la información del medio para un fin en común.

El uso de nuevas herramientas de software que utilizan sensores de los dispositivos móviles contribuye al desarrollo de aplicaciones más complejas y con mejor funcionalidad, sin la necesidad de implementar infraestructura adicional a la que es utilizada día a día.

La integración de metodologías de crowdsensing, APIs, y sensores de los dispositivos usados en este proyecto permiten obtener resultados más precisos, reales y confiables de acuerdo a las condiciones bajo las cuales el sistema trabaja.

Es recomendable que, en las etapas iniciales del uso de la aplicación no solo se haga uso del método oportunista cuando se trabaja bajo la modalidad crowdsensing, sino también implementar la toma de datos de manera participativa ya que de esta manera los datos podrán ser balanceados, hasta que se consiga que al menos el

75% de los usuarios de los parqueaderos y poder trabajar únicamente con el método oportunista.

Para que el sistema sea escalable, se recomienda hacerlo distribuido, es decir, no depender únicamente de un servidor central, sino que varios servidores sean capaces de ejecutar localmente los algoritmos y luego transferir los resultados de los mismos.

Una mejora al sistema es la implementación de mecanismos externos que se encarguen de controlar la veracidad de los estados actuales de los parqueaderos. El control estaría a cargo de una persona que periódicamente actualice el estado de los parqueos a través de un dispositivo también conectado al sistema.

Si se desea expandir el alcance del sistema e implementar medidas de seguridad y control personalizado, se podría solicitar un registro de los usuarios con su vehículo (placa, modelo, color de auto, identificación del usuario) previo al ingreso al sistema como participantes.

Mejorar el sistema para que brinde mayor información sobre los parqueos disponibles, es decir, sea capaz de mostrar cuáles de los lugares está disponible y no solo cuántos. Esto se logra numerando los lugares dentro de cada estacionamiento.

## BIBLIOGRAFÍA

- [1] ARCOTEL, “Infografía SMA.” ARCOTEL, 2017.
- [2] BBC News, “Ocho maneras de darle usos insospechados a tu celular,” *BBC Tecnología*, 2015. [Online]. Disponible: [http://www.bbc.com/mundo/noticias/2015/01/150102\\_tecnologia\\_uso\\_distinto\\_de\\_sensores\\_celular\\_ig](http://www.bbc.com/mundo/noticias/2015/01/150102_tecnologia_uso_distinto_de_sensores_celular_ig).
- [3] I. Luzardo, “10 características de los actuales súper celulares,” *Enter Co*, 2016. [Online]. Disponible: <http://www.enter.co/chips-bits/smartphones/10-caracteristicas-de-los-actuales-super-celulares/>.
- [4] W. Miao, “Desarrollo de aplicaciones de sensores en teléfonos y tabletas Android,” *Intel Software*, 2013. [Online]. Disponible: <https://software.intel.com/es-es/android/articles/developing-sensor-applications-on-intel-atom-processor-based-android-phones-and-tablets>.
- [5] Universidad Autónoma de Barcelona, “Mobile Crowd Sensing,” *SeNDA*, 2014. [Online]. Disponible: <http://senda.uab.es/node/15>.
- [6] C. Tanas and J. Herrera, “Users as Smart Sensors: A Mobile Platform for Sensing Public Transport,” *Citiz. Sens. Netw.*, vol. 7685, pp. 81–93, 2012.
- [7] F. Ucha, “Definición de Geolocalización,” *Definición ABC*, 2011. [Online]. Disponible: <https://www.definicionabc.com/geografia/geolocalizacion.php>.
- [8] R. Suarez, “¿Qué es Geolocalización?,” *IEMD*, 2016. [Online]. Disponible: <https://iiemd.com/geolocalizacion/que-es-geolocalizacion>.
- [9] INTERCOMP, “A Smart Parking Systems COMPLETE, EASY to MANAGE and PLEASANT to USE.” [Online]. Disponible: <http://smartparkingsystems.com/en/smart-parking-system/>.
- [10] Smart Parking Limited, “What can SmartApp do for me?” [Online]. Disponible: <http://smartparkingsystems.com/en/smart-parking-system/>.
- [11] Smart Parking Limited, “SmartSpot Gateway: Helping Cities to Operate Efficiently.” [Online]. Disponible: <https://www.smartparking.com/technologies/smartspot-gateway>.

- [12] Smart Parking Limited, "PayMyPark with Smart Parking technology." [Online]. Disponible: <https://www.smartparking.com/keep-up-to-date/case-studies/wellington-city-council-new-zealand>.
- [13] PHP.Net, "¿Qué se puede hacer en PHP?," *Manual de PHP*. [Online]. Disponible: <http://php.net/manual/es/intro-whatcando.php>.
- [14] Oracle, "World's most popular open source Database," *Oracle MySQL Overview*. [Online]. Disponible: <https://www.oracle.com/mysql/index.html>.
- [15] M. Rouse, "Geofencing," *Techtarget*, 2013. [Online]. Disponible: <http://whatis.techtarget.com/definition/geofencing>.
- [16] T. Choudhury and G. Borrielo, "The Mobile Sensing Platform: An Embedded System for Activity Recognition.," *IEEE Pervasive Magazine-Special Issue on Activity-Based Computing*, 2008.
- [17] M. Kumar, "Google's Activity Recognition API is awesome but where are the apps?," *Geoawesomeness*, 2015. [Online]. Disponible: <http://geoawesomeness.com/googles-activity-recognition-api-is-awesome-but-where-are-the-apps/>.
- [18] U. S. trade comission S. International Trade Comission, "GPS Receiver," in *Certain GPS Chips, Associated Software and Systems, and Products Containing Same*, U.S. International Trade Comission, 2010, p. 6.
- [19] A. Sabán, "Cómo funcionan el acelerómetro y el giroscopio de los móviles," *Hipertextual*, 2016. [Online]. Disponible: <https://hipertextual.com/2016/08/acelerometro-giroscopio>.
- [20] Google, "Geofence API," *Google APIs for Android2*, 2017. [Online]. Disponible: <https://developers.google.com/android/reference/com/google/android/gms/location/Geofence>.
- [21] Google, "ActivityRecognitionApi," *Google APIs for Android*, 2017. [Online]. Disponible: <https://developers.google.com/android/reference/com/google/android/gms/loc>

ation/ActivityRecognitionApi.

## ANEXOS

**ANEXO A: ENCUESTA REALIZADA A LOS USUARIOS DE LA  
APLICACIÓN MÓVIL ESPOL PARKING**

ENCUESTA

Elija el parqueadero que usa con mayor frecuencia:

CELEX

FIEC

FIMCP

Responda las siguientes preguntas

1. ¿La aplicación es de fácil manejo?

Sí

NO

2. ¿Le ahorró tiempo al buscar parqueos disponibles?

Sí

NO

3. ¿Cree usted que una participación activa es mejor que una pasiva?

Sí

NO

4. ¿Usaría esta app para conocer los parqueos disponibles de su facultad?

Sí

NO

## **ANEXO B: DOCUMENTACIÓN DE LA APLICACIÓN MÓVIL “ESPOL-PARKING”**

El ícono de la aplicación móvil es el que se detalla en la figura A.1.



**Figura A.1: Ícono representativo de la aplicación ESPOL-PARKING**

La programación de la aplicación móvil se la realizó a través del IDE Android Studio y el lenguaje de programación es JAVA.

ESPOL-PARKING está dividida en 4 clases que se detallan a continuación.

### **1. Clase MainActivity.java:**

Esta clase es la principal del desarrollo de la aplicación. Integra y enlaza todos los objetos, las clases, los métodos y demás elementos necesarios para el funcionamiento correcto de la aplicación.

- a. Creación de clientes: El método *createGoogleApiClient* permite hacer uso de los servicios que ofrece las APIs de Geofence y ActivityRecognition mediante dichas variables.
- b. Delimitación de las zonas: Las zonas geográficas, que corresponden a los tres parqueaderos, son delimitadas por coordenadas de latitud y longitud, y el radio del área. El método *createGeofences* es el encargado de crear e instanciar las zonas y anexarlas a una lista de zonas de geofence, lo que facilita la escalabilidad del código.
- c. Monitoreo de las zonas: Una vez creadas las zonas a ser controladas el método *startLocationMonitoring* realiza consultas acerca de la posición actual del usuario mediante el uso de GPS.

## 2. Clase `ActivityRecognizedService.java`:

Esta clase abstrae el uso del API Activity Recognition. Registra el tipo de actividad que el dispositivo se encuentra realizando en cada instante. Las operaciones que esta clase realiza son las siguientes:

- a. Detección de Actividad más probable: Existen 8 tipos de actividades que soporta el API de acuerdo a los datos que se registran en los sensores de acelerómetro y giroscopio de los dispositivos. Estos tipos son: ON\_FOOT (caminando o corriendo), RUNNING (corriendo), WALKING (caminando), IN\_VEHICLE (en carro), ON\_BICYCLE (en bicicleta), STILL (sin movimiento), TILTING (cuando el dispositivo detecta algún tipo de inclinación) y UNKNOWN (actividad desconocida). Para el desarrollo de la aplicación fueron necesarios los siguientes tipos de actividad: IN\_VEHICLE para detectar cuando un usuario entre o salga del

estacionamiento en carro, y para detectar que el usuario haya entrado o salido de una zona a pie, se hace uso de los tipos ON\_FOOT, WALKING, y RUNNING. Cuando el API reconoce una actividad, también lo hace con el porcentaje de confianza o seguridad que el API tiene respecto a la decisión que tomó, es decir, para cierto instante de tiempo el API le asigna un porcentaje de confianza a cada uno de los tipos; pero a través del método *getMostProbableActivity* no solo reconoce el tipo de actividad y sus porcentajes de confianza, sino también elige el tipo de actividad con el mayor índice de confianza, es decir, el del cual el API está más seguro que es el valor real.

- b. Manejo de actividades: Una vez que una actividad ha sido detectada se hace uso del método *handleDetectedActivities*, el cual establece el valor de la variable “estado” a “pie” o “carro” dependiendo de la actividad que haya se manifestado.

### 3. Clase `GeofenceService.java` :

Esta clase se encarga de gestionar todos los recursos provistos por Google’s Geofence API. Entre las acciones que permite realizar se encuentran las siguientes:

- a. Detección de entrada o salida de una zona: Cuando un dispositivo, basado en la ubicación geográfica, ingresa o sale de una de las zonas previamente configuradas, el método *getGeofenceTransition* responde con la activación de un *trigger* que acciona el proceso de análisis del estado actual del dispositivo. Existen 3 tipos de estados que el API soporta, siendo estas ENTER (entrando a la zona), DWELL (cuando se encuentra en la zona por un periodo de tiempo) y EXIT (saliendo de la zona). Para el desarrollo de nuestra

aplicación fueron necesarios únicamente los estados de ENTER y EXIT.

- b. Detección de la zona: Seguido de la ejecución del método *getGeofenceTransition*, el método *getTriggeringGeofences* entrega una lista de las zonas que han sido activadas por el *trigger*, ya sea que se active por entrada o salida de la misma.
- c. Registro de actividad al entrar o salir a una zona: Por medio del método *triggeringGeofencesIdsString* es posible obtener el Id (nombre) de la zona en la que se encuentra el usuario. Luego que eso ocurre, se llama al método *handleDetectedState* que contiene el valor instantáneo de la actividad en la que el dispositivo se encuentra. Finalmente, cuando dicho valor es conocido, y al saber en qué parqueadero se está realizando dicho procedimiento, se pasan dichos valores para ser ejecutados por el script `ingresoSalida.php` descrito en el anexo C y escribir en la base de datos.

#### **4. Clase `ApiConnector.java`:**

Dentro de esta clase se puede encontrar el método y las variables utilizadas para presentar los datos contenidos en el script `selectdata.php` descrito en el anexo C, el cual incluye la información alojada en la base de datos acerca de los lugares disponibles en los parqueaderos establecidos.

Para más detalles, el código fuente de la aplicación se encuentra detallado en el anexo D.

### **ANEXO C: DESCRIPCIÓN DE LOS SCRIPTS**

El script denominado `ingresoSalida.php` está desarrollado en PHP para ser ejecutado por el servidor web cuando los datos proporcionados por el usuario le lleguen al mismo por el puerto 80. Este script ejecuta el algoritmo descrito en la sección 3.4.3 y además proporciona un enlace a la base de datos para leer y escribir la información requerida por la aplicación móvil.

El algoritmo funciona de la siguiente forma:

1. Validación del origen de los datos, es decir, de qué facultad se están enviando la información.
2. Lectura de información actual desde la base de datos.
3. Ejecución de las comparaciones descritas en la tabla 3 y generación de eventos.
4. Modificación de la información actual de los parqueos de acuerdo al evento generado.
5. Sobreescritura de datos en el servidor. El número de parqueos de la facultad analizada se modifica por el nuevo valor de acuerdo a los resultados de la ejecución del algoritmo.

El código fuente del script `ingresoSalida.php` se encuentra en el Anexo D.

El script denominado `selectdata.php` de la misma manera que el script anterior se encuentra desarrollado en PHP para ser ejecutado por el servidor web cuando los datos proporcionados por el usuario le lleguen al mismo por el puerto 80. Este script proporciona un enlace a la base de datos para leer únicamente la información relacionada con el número de lugares disponibles en cada uno de los estacionamientos previamente establecidos.

### **ANEXO D: REPOSITORIO DIGITAL**

El repositorio digital se encuentra en el CD adjunto a este documento, dentro del cual se encuentran los siguientes archivos y carpetas:

- Carpeta ESPOL-PARKING: Contiene el código fuente de la aplicación móvil.
- Archivo `ingresoSalida.php`: Código fuente del script del mismo nombre descrito en el anexo C.
- Archivo `selectdata.php`: Código fuente del script del mismo nombre descrito en el anexo C.
- Archivo: Es el instalador de la aplicación móvil. Debe ser abierto desde el dispositivo donde se lo va a instalar o desde un emulador