



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“IMPLEMENTACIÓN DE UN SISTEMA DE SENSADO A
TRAVÉS DE USUARIOS MÓVILES PARA EL CONTROL
Y NOTIFICACIÓN DE DISPONIBILIDAD DE PARQUEO
LIBRE DE FIEC-ESPOL.”**

INFORME DE MATERIA INTEGRADORA

Previo a la obtención del Título de:

INGENIERA EN TELEMÁTICA

BRAVO MITE ELIZABETH MONSERRAT

TOMALÁ ESPARZA AMIRA CAROLINA

GUAYAQUIL – ECUADOR

AÑO: 2016

AGRADECIMIENTOS

Agradezco primero a Dios quien me ha dado sabiduría e inteligencia para continuar y culminar mis estudios. De la misma forma, agradezco en gran manera a mis padres y hermano quienes me han apoyado en cada etapa académica que me ha tocado vivir, también a mi abuela que siempre estuvo ahí dándome consejos y pues finalmente agradezco a todos mis profesores quienes han sido excelentes instructores para formarme como ingeniera.

AMIRA CAROLINA TOMALÁ ESPARZA

Le agradezco a Dios, ya que sin su presencia en mi vida no habría sido capaz de llevar a cabo esta meta dentro de mi plan de vida. Eterna gratitud a mis padres quienes siempre fueron el motor para continuar y no decaer ante ninguna adversidad que se me presentó en el camino; Emmita, siempre estuvo presente con una palabra de aliento, un consejo, en esas malas noches por mis proyectos o en un festejo; Benito, gracias por mostrarme una dirección a mi formación profesional desde pequeña y por compartir instantes de mi desarrollo académico. A los tutores que estuvieron a lo largo de mi vida académica, quienes compartieron sus conocimientos, y me ayudaron a formar un perfil profesional de calidad. A mis queridos amigos de CN, sin su apoyo y consideraciones no habría logrado avanzar durante este último periodo.

ELIZABETH MONSERRAT BRAVO MITE

DEDICATORIA

Dedico a mi mamá porque ha sido el pilar fundamental en mi vida y me ha brindado su amor, tiempo, paciencia y apoyo incondicional a lo largo de mi vida, por su esfuerzo al guiarme por el buen camino y aconsejarme para poder ser una persona con excelentes valores y virtudes.

A mi abuela; quien me ama mucho, siempre está atenta a mi vida y me tiene en sus oraciones.

Y en general a toda mi familia que es lo mejor y más valioso con lo que Dios me ha bendecido

AMIRA CAROLINA TOMALÁ ESPARZA

Este trabajo está dedicado a mis padres y mi familia, sobre todo a mi Abueli, sé que estarías muy orgullosa.

ELIZABETH MONSERRAT BRAVO MITE

TRIBUNAL DE EVALUACIÓN

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

PROFESOR EVALUADOR

.....
Ing. Vladimir Sánchez

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 ESPOLE 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"

.....
Bravo Mite Elizabeth Monserrat

.....
Tomalá Esparza Amira Carolina

RESUMEN

Hoy en día, la cantidad de vehículos que circulan en el campus Gustavo Galindo de la Escuela Superior Politécnica del Litoral, se ha visto en aumento. Es por ello que los lugares destinados para el estacionamiento de los vehículos resultan insuficientes en algunos casos, dependiendo del horario.

Los usuarios de los estacionamientos no cuentan con algún sistema de información que les permita conocer la disponibilidad de parqueos vacantes y esto se genera en pérdida de tiempo, retrasos al tener que rondar toda la localidad para buscar un espacio libre y poder dejar el vehículo estacionado.

El propósito de este trabajo crear una plataforma que maneje los datos de los usuarios y la cantidad de parqueos que tiene un estacionamiento dentro del campus. Para esta etapa inicial la investigación y desarrollo se ha centrado en el estacionamiento FIEC/FIMCP. Adicionalmente, se desarrollará una aplicación móvil que permita la interfaz con los usuarios para que puedan reportar al sistema y así acceder a la información de parqueos disponibles.

El sistema propuesto se basará en el concepto de Mobile Crowdsensing. Mobile Crowdsensing es una técnica que permite capturar datos a través de los dispositivos móviles de los usuarios por medio de una aplicación que hace uso de los diferentes tipos de sensores que posee el dispositivo, de esta manera es posible detectar los diversos acontecimientos del medio que lo involucra. Mediante este concepto, se permite capturar datos tales como: que usuarios han dejado estacionado su vehículo y el número de parqueos disponibles que otros usuarios móviles que deseen reportar dicha información.

En las pruebas realizadas se pudo estimar que los usuarios demuestran afinidad con la idea y el uso de una aplicación móvil para tener información actualizada de la disponibilidad de estacionamientos y les permita un ahorro de tiempo como incentivo a sus reportes. Adicionalmente, se demuestra que es factible manejar los datos de los usuarios y de existir reportes no reales es posible descartarlos de la media y así crear una estimación de la cantidad de parqueos libres existente.

ÍNDICE GENERAL

AGRADECIMIENTOS.....	ii
DEDICATORIA.....	iii
TRIBUNAL DE EVALUACIÓN.....	iv
DECLARACIÓN EXPRESA.....	v
RESUMEN.....	vi
ÍNDICE GENERAL.....	vii
CAPÍTULO 1.....	1
1. SISTEMA DE PARQUEO.....	1
1.1 Antecedentes.....	1
1.2 Descripción del problema.....	2
1.3 Delimitación del problema.....	4
1.4 Posibles Soluciones.....	4
1.5 Justificación.....	6
1.6 Objetivos.....	7
1.6.1 Objetivo General.....	7
1.6.2 Objetivos Específicos.....	7
CAPÍTULO 2.....	8
2. MOBILE CROWDSENSING.....	8
2.1 Características.....	9
2.1.1 Tipos.....	9
2.1.2 Recompensas o penalidades.....	10
2.2 Arquitectura.....	11
2.2.1 Colaboradores.....	12
2.2.2 Plataforma.....	12
2.2.3 Clientes.....	13
CAPÍTULO 3.....	18
3. DESARROLLO DEL SISTEMA.....	18

3.1	Metodología	18
3.1.1	Parámetros	19
3.1.2	Recompensas	19
3.2	Investigación de campo (encuestas y análisis)	19
3.2.1.	Análisis de la primera encuesta: Usuario Generales	20
3.2.2.	Análisis de la segunda encuesta: Usuarios de Parqueo	23
3.3	Descripción Conceptual / Estructura del proyecto	25
3.4	Descripción del Sistema MCS	26
3.5	Elementos de Programación Utilizados	27
3.6	Funcionalidad	29
3.6.1	Servidor	29
3.6.2	Aplicación Móvil	32
3.7	Resultados	34
	CONCLUSIONES Y RECOMENDACIONES	42
	BIBLIOGRAFÍA	43
	ANEXOS	48
	ANEXO A. Detalle de la primera encuesta realizada	48
	ANEXO B. Detalle de la segunda encuesta realizada.	50
	ANEXO C. Manual de Usuario	51
	ANEXO D. Manual de Instalación	59

CAPÍTULO 1

1. SISTEMA DE PARQUEO

1.1 Antecedentes.

El espectro de análisis del sistema de parqueos dentro de la ciudad de Guayaquil es muy amplio en estos días, ya que muchos presentan una debilidad ante seguridades y control de acceso que tienen las personas, a las instalaciones, a través de un vehículo de uso personal.

Se ha podido constatar que la mayoría de lugares donde un usuario es capaz de dejar estacionado su vehículo, solo cuentan con un sistema de tickets, los cuales, solo disponen de personal humano para hacer entrega y recibo de los mismos. Estos tickets únicamente tienen escrito la placa del vehículo y la hora de ingreso al parqueadero. Más sin embargo no es posible tener conocimiento previo de la disponibilidad de parqueos existentes en el estacionamiento.

Cuatro años atrás se pudo visualizar un gran cambio en lo que respecta al sistema de estacionamiento dentro de la ciudad, sobre todo en los grandes centros comerciales, se instalaron sensores de movimiento en cada uno de los espacios destinados para el parqueo, lo cual permitía al usuario que ingresaba a las instalaciones conocer la disponibilidad y sugerencias del lugar donde existía un espacio libre para dejar el vehículo estacionado [1]. No obstante, el sistema de registro vehicular al parqueadero seguía manteniéndose, el uso de tickets. Aunque algunos ya presentaban una automatización y seguridad con respecto al registro de los vehículos, la entrega del ticket se generaba una vez que el usuario era registrado en la base de datos del centro comercial, estos a su vez contienen un código de barras y la información relevante al vehículo como la placa y el propietario.

La siguiente evolución que presentaron ciertos estacionamientos fue la entrega de tarjetas magnéticas. En estas tarjetas se procede a registrar la placa del vehículo que ingresa al establecimiento, de esta manera se controla el factor de falsificación de los tickets creando una barrera de seguridad ante un intento de robo vehicular, lo que se presenta aún en ciertas instituciones.

Esta modalidad de tener un control de los vehículos que ingresaban a las instalaciones de ciertas instituciones, se hizo muy común, he inclusive se implementaron en las Universidades.

Entre las universidades conocidas con un sistema de control de acceso a un estacionamiento se encuentran la Universidad Católica Santiago de Guayaquil, Universidad Politécnica Salesiana, Campus Guayaquil. Es significativo aclarar que la Universidad Politécnica Salesiana cuenta con diferentes edificaciones para uso de parqueo de vehículos tanto para estudiante y visitantes como para profesores, y en ciertas se visualiza ya implementado un panel de led indicativo de la cantidad de espacios libres.

Dentro de esta era de desarrollo tecnológico es posible contar con diferentes opciones como las antes mencionadas cuales han ido evolucionando eventualmente a las necesidades. Al paso de los años el uso de los dispositivos móviles se ha vuelto parte importante para crear nuevas opciones de solución ante los diferentes problemas que presenta la sociedad, tal como leer un código de barras o QR para acceder a diferentes medios de información, o capturar imágenes para reportar eventualidades de la comunidad. Es por ello que alrededor de mundo se han ideado ciertas aplicaciones para que los dispositivos móviles puedan ser parte de la solución a través de sus diferentes características y sensores para, detectar y valorar las circunstancias que se desenvuelven en un cotidiano vivir y de esta manera resolver las dificultades que tengan ya que es un método en que el gasto por inversión resulta más ligero que implementar una infraestructura con equipos de demandan gasto de mantenimientos eventualmente y que luego tienen que ser reemplazados. Esta técnica en la cual los usuarios pueden enviar datos por medio de una aplicación móvil para mantener informados de los cambios que se puedan dar, es conocida como Mobile Crowdsensing.

1.2 Descripción del problema.

Dentro de las instalaciones de la Escuela Superior Politécnica del Litoral, ESPOL, en el campus Gustavo Galindo, consta con diferentes áreas destinadas para uso

de parqueo vehicular tanto para estudiantes como personal externo o trabajadores de ESPOL.

Cada facultad, unidad académica, edificio administrativo, cuenta con un espacio cercano asignado para facilitar el parqueo de los vehículos de los usuarios que vayan a hacer uso de las instalaciones antes mencionadas.

Con el aumento de vehículos que circulan en el campus ESPOL por parte de los estudiantes y visitantes, en la actualidad, origina la falta de lugares destinados al estacionamiento, todo esto sumado a la ausencia de un plan estratégico para la liberación vehicular y del bloqueo en el que se ha visto sumido el campus, han sido el principal detonante del caos en intersecciones y áreas de parqueadero, esto se ve reflejado en la incomodidad que genera para los usuarios.

Otra de las consecuencias que ha traído este aumento de vehículos es la contaminación ambiental a la que se ve expuesta a los estudiantes. El stress y la tensión son factores importantes a considerar ya que afectan directamente a los conductores en el momento que intentan encontrar un sitio disponible para dejar sus vehículos y proseguir con sus responsabilidades dentro del campus, clases, reuniones de estudio, asistir a las citas académicas, etc.

La oferta de estacionamientos es insuficiente, lo cual genera conflictos de tránsito, y pérdida de tiempo, derivados del exceso de circulación lo cual complica determinadamente el encontrar un sitio de parqueo, sea este uno cercano o distante.

Por el momento, no se ha implementado un sistema apropiado con el cual los estudiantes puedan observar la cantidad de sitios libres para dejar estacionado su vehículo antes de ingresar a una localidad de parqueo.

1.3 Delimitación del problema.

La falta de sitios donde estacionarse en el campus ESPOL, ocasiona un desorden y obstrucción en el flujo vehicular debido a la circulación que deben efectuar los autos en busca de un lugar para estacionarse.

La delimitación del problema se encuentra definida al parqueo FIEC/FIMCP el cual forma parte del Campus Gustavo Galindo de la ESPOL, en donde se desarrollan varias actividades de diferentes índoles, clases, conferencias, reuniones, exposiciones, etc.

Cabe señalar que cada lugar para estacionar dentro del parqueo FIEC/FIMCP no cuenta con una apropiada numeración que lo identifique, a pesar de tener limitado el lote. A esto se adiciona que los usuarios ubican sus vehículos en zonas no asignadas tratando de buscar un lugar para dejar su vehículo sin perjudicar la salida de los que ya están estacionados y así evitar un poco más la congestión existente.

Los estudiantes son los primeros afectados bajo la disponibilidad de lugares libres en un parqueadero específico, ya que al no encontrar parqueo principalmente se retrasan para iniciar sus responsabilidades.

Dentro del área de estudio para el desarrollo de proyecto, se ha propuesto el análisis de la región de parqueadero antes mencionada, sin embargo, es posible extenderse a los demás que dispone el campus.

1.4 Posibles Soluciones.

Dentro de nuestro medio se encuentran implementado, o en vías de desarrollo, varios sistemas en los cuales, brindan de alguna forma una solución al problema existente dentro de las áreas designadas para el parqueo de vehículos.

- Sistema de parqueo con sensores ultrasónicos. [2]

Materiales	Costos
Sensor ultrasónico	\$40
Led Display Parqueo	\$65
TOTAL	\$105

Tabla 1: Costos de implementos requeridos.

- Sensor de espacios de estacionamiento mediante cámara web. [3]

Materiales	Costos
Logitech HD Webcam C310	\$31
Raspberry Pi 3 Model B	\$40
TOTAL	\$71

Tabla 2: Costos de implementos requeridos.

- Sistema de parqueo con sensores inalámbricos. [4]

Materiales	Costos
Sensor Smart Parking	\$450
Meshlium Digimesh PRO BG-AP(dependiendo del rango de acción que abarca)	\$1130.78
Wasmote 802.15.4-PRO SMA 5 dBi 2.4 GHz	\$189
Wasmote Digi Mesh-PRO SMA 5dBi 2.4 GHz	\$189
Expansion Radio Board	\$30.40
Modulos Xbee 802.15.4 PRO SMA 5 dBi	\$49
Modulos Xbee Digimesh-PRO SMA 5 dBi	\$49
Modulo RFID 13.56 MHz/NFC	\$61
Bateria recargable Li-Ion 12V, 6600 mAh	\$37
Cargador para batería recargable	\$5
Regulador de carga 12V	\$25
Tarjetas RFID 13.56 MHz/NFC	\$1
TOTAL	\$2216.18

Tabla 3: Costos de implementos requeridos.

- Sistema de reconocimiento por medio de uso de cámaras.

Materiales	Costos
CCD Cámara *4	\$80
Servidor	\$300
Switch 4 Puertos *2	\$40
TOTAL	\$420

Tabla 4: Costos de implementos requeridos.

Actualmente, el parqueo FIEC/FIMCP cuenta con 160 lugares habilitados para beneficio de los estudiantes y visitantes de las instalaciones.

Los valores antes estipulados, en las tablas 1 a la 4, están referenciados para cada lugar designado para el parqueo, exceptuando el análisis del *Sistema de reconocimiento por medio de uso de cámaras*, ya que aquel solo funciona con cámaras a la entrada y salida del parqueo. El costo de

cableado en caso de ser necesario dependerá de la estructura a usar, aquello no se encuentra dentro de este análisis.

Como se observa en la tabla 5, los costos de implementación para cubrir toda el área del estacionamiento FIEC/FIMCP, representan gastos significativamente altos dependiendo del sistema a implementar.

SISTEMA	VALOR UNITARIO	VALOR TOTAL FIEC/FIMCP
Sistema parqueo con sensores ultrasonico	\$40	\$6465
Sensor de espacios de estacionamiento mediante cámara web	\$31	\$430
Sistema de parqueo con sensores inalámbricos	\$2216.18	\$351200
Sistema de reconocimiento por medio de uso de cámaras	\$420	\$420

Tabla 5: Costos de implementos requeridos.

1.5 Justificación.

La información sobre los parqueaderos en tiempo real permite disminuir el tráfico que es causado por la búsqueda de los mismos, ya que al direccionar al usuario de forma inmediata al lugar que contiene un sitio disponible para aparcar el auto, se elimina el tránsito que hubiese ocasionado al tratar de encontrar un parqueadero.

El sistema ayudará a proporcionar los datos necesarios para que los conductores puedan obtener un sitio en los lugares destinados a parqueos, acción que actualmente es una tarea laboriosa y en ocasiones infructuosa ya que los usuarios no tienen más opción que dejar estacionados los automóviles en sitios no dispuestos para este fin.

En particular, el campus de la ESPOL enfrenta este problema, el aumento de automóviles y la insuficiencia de sitios dispuestos para parqueaderos. Al disponer de la información, muchos de los problemas antes mencionados se reducirán, sobre todo el estudiante podrá tener mejor manejo de su tiempo. A continuación, se detallan algunos de los beneficios.

- Administración eficiente de los parqueaderos.
- Disminución de tráfico por búsqueda de lugares disponibles para aparcar.
- Reducción del tiempo destinado para la búsqueda.
- Satisfacción del usuario.
- Confianza en la información publicada.
- Seguridad en el uso del parqueo.

1.6 Objetivos

1.6.1 Objetivo General

Elaborar una aplicación móvil para la administración y presentación de información acerca de la ubicación, disponibilidad, asignación de espacios en parqueadero FIEC/FIMCP, el cual brindará una solución ágil y tecnológica a una problemática que se vive a diario dentro del Campus ESPOL.

1.6.2 Objetivos Específicos

- Diseñar una aplicación móvil, que permite a los usuarios realizar consultas en tiempo real de los lugares destinados al parqueo de vehículos.
- Mantener informados a los usuarios en cuanto a espacios disponibles dentro del parqueo de la FIEC.
- Llevar un control actualizado de la demanda de estacionamientos dentro de la Facultad.
- Proporcionar Seguridad a los usuarios que estacionan sus vehículos dentro del parqueo de la FIEC.
- Aplicar métodos y mecanismo de monitoreo en tiempo real mediante la ubicación estratégica de sensores, los cuales permiten captar información sobre los parqueos disponibles.

CAPÍTULO 2

2. MOBILE CROWDSENSING.

La terminología Crowdsensing, se ha ido popularizando desde hace más de una década. Este concepto implica que se hará uso de los dispositivos pertenecientes a los usuarios que se encuentren en una red específica. El objetivo principal para el uso de los dispositivos, es el de reunir datos que son importantes para la comunidad a que están vinculados, de esta manera los datos que son capturados hacen un reflejo de los principales intereses de los usuarios. Este concepto se aplica a todo dispositivo móvil inteligente, es decir a aquellos que permitan una comunicación entre distintos dispositivos bajo diferentes medios y normas, entre ellas WIFI, NFC, Bluetooth, etc. Actualmente el dispositivo móvil inteligente más usado es el teléfono celular, el cual tiene muchos sensores importantes para el desarrollo de esta idea, pero es conveniente mencionar que hay otros dispositivos que utilizan diferentes sensores que destacan por su uso cotidiano en ciertos casos, a pesar que no interactúan directamente con el usuario final, es posible encontrarlos incluso en vehículos.

La oportunidad de crear una red de alto alcance a un costo bajo es uno de los factores que potencializa esta tendencia, ya que no es necesario invertir un gran capital monetario para suplir la necesidad de equipos que provean la captura de datos en tiempo real y sean capaces de tener una comunicación inalámbrica hacia la plataforma. A medida que aumenta la cantidad de usuarios debido a su difusión, se estima un crecimiento gradual de la plataforma.

Como se mencionó, los dispositivos móviles inteligentes tienen sensores que posibilitan la captura de ciertos datos que son suministrados para un análisis y poder establecer que inclinación tiene la información que ha sido recopilada de todos los usuarios. Los sensores se catalogan en diferentes aspectos como los de posición, radios, ambientales y existen otros que manejan características propias del dispositivo. Por lo tanto, es importante tener identificado cuáles serán los datos y cuáles serán sus métodos de recolección de información.

Smart cities, es el ejemplo fehaciente de la implementación de este modelo de detección de datos, en la cual su propósito es poder generar una comunidad involucrada en diferentes aspectos de interés como lo son, seguridad, economía, emergencias, salud, desastres, tránsito y muchos más, con los que de acuerdo a la participación de los usuarios es posible mantener una gestión de información actualizada. El conjunto de los aspectos mencionados, crean una Smart City, pero para llegar a ese punto ideal es preciso ir desarrollando cada uno bajo este modelo de mobile crowdsensing. Por consiguiente se enfatiza que el propósito de esta tecnología es mantener una comunidad activa, participativa y sostenible para desarrollar un cambio positivo a la calidad de vida actual.

En vista que estos dispositivos tendrán un uso continuo por parte del usuario, no solamente por la colaboración que realice a la plataforma sino por el uso habitual que tiene, el tiempo de vida de la batería es uno de los factores externos que influyen en la captura de datos, así como también los requerimientos de sistema que pueda gestionar el software en caso de que un dispositivo no cumpla con todos los requisitos.

Gracias al uso de software y hardware libre es posible desarrollar aplicaciones que ayudan a solidificar lo que significa Mobile Crowdsensing ya que estas impulsan ciertos sensores que son de utilidad para la captura de datos. Para el manejo de estos sensores y es importante determinar una API (Application Programming Interface) que vaya acorde con el propósito que se quiera lograr por medio de una aplicación móvil ya que provee un conjunto de funciones que habilitan el control de los mismos.

2.1 Características

Mobile Crowdsensing está compuesta por tipos de funcionamiento, operadores, recompensas, tal como se visualiza en la figura 2.1. A continuación se detalla el enfoque de cada una de ellas.

2.1.1 Tipos

Dentro de la tendencia del Crowdsensing se ven implicadas dos tipos de funcionamiento las cuales son participativa y la oportunista en el marco de interacción de las personas y así mismo se divide en dos para la cantidad de personas que se ven involucradas, comunidad o personal.

- Participativa: Es aquella donde el usuario se ve involucrado de manera recurrente en las diferentes actividades que se vea comprometido en la aplicación.
- Oportunista: Este funcionamiento implica un trabajo mínimo por parte del usuario ya que la tarea asignada funciona de manera automática para la captura de datos a ser detectados.
- Comunidad: Los datos que son capturados deben provenir de una gran cantidad de personas que estén envueltas en el ejercicio, en las cuales exista un gran nivel de confianza y una buena interacción para que dichos datos sean de gran aporte.
- Personal: La información que es proporcionada por una sola persona describiendo alguna actividad en particular de cualquier evento en el cual se es participe y del cual se pueda informar a los demás usuarios de la plataforma.

2.1.2 Recompensas o penalidades

Bajo las políticas que maneja el Mobile Crowdsensing, se encuentra estipulado que los usuarios-trabajador reciban ciertas recompensas por el aporte generado en la plataforma.

Las recompensas pueden ser contempladas, en ciertas aplicaciones como lo es el caso:

- Aportes monetarios: Gift Hunter Club. [5]
- Estímulo social: TuMicro, Perú [6]; Bike Cidadao, Brasil. [7]
- Gaming: Contribuye con Google. [8]

Dependiendo de los intereses que existan de los colaboradores. Es importante tener en claro las condiciones de privacidad que se declaran para los usuarios, ya que no toda información proporcionada puede o debe ser expuesta o distribuida para terceros.

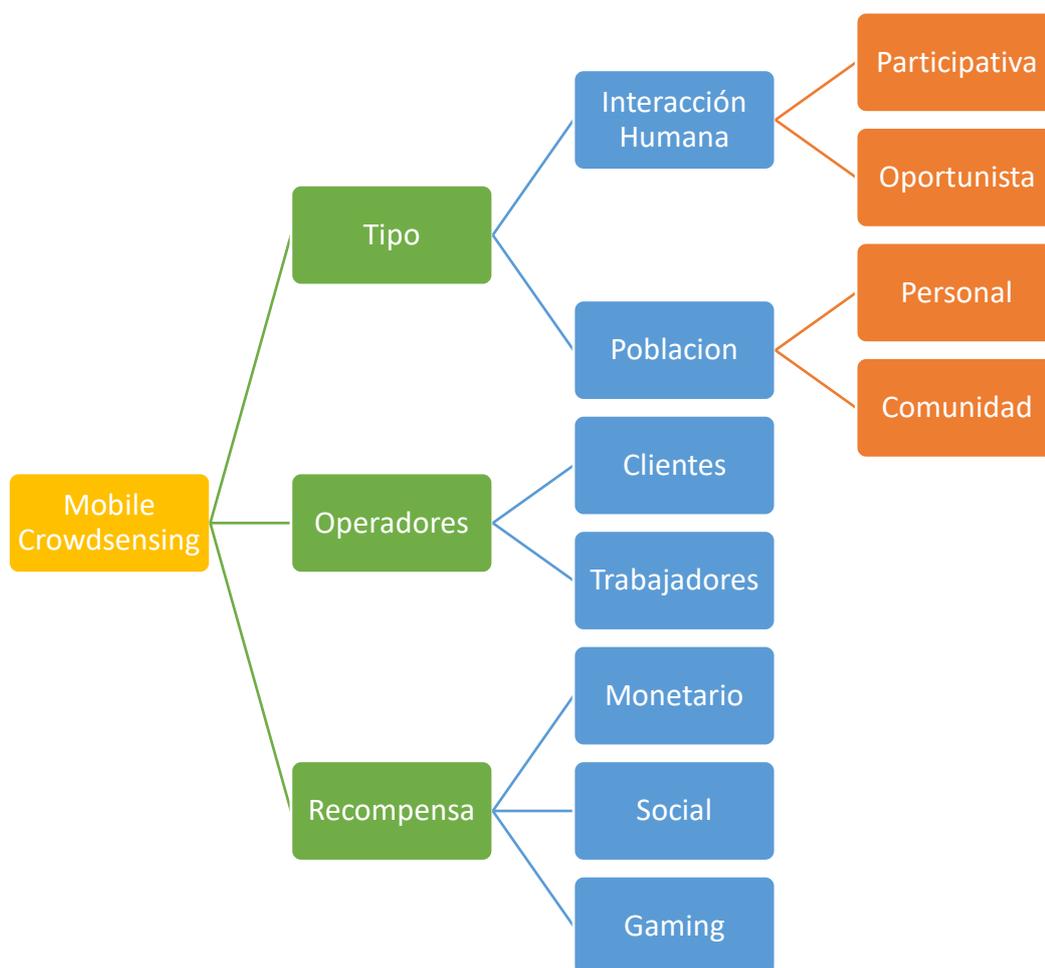


Figura 2.1: Diagrama conceptual de Mobile Crowdsensing.

2.2 Arquitectura

El modelo de Mobile Crowdsensing está compuesto por las características antes mencionadas. A continuación, se explicará la arquitectura que debe tener una aplicación que da cabida a esta tecnología.

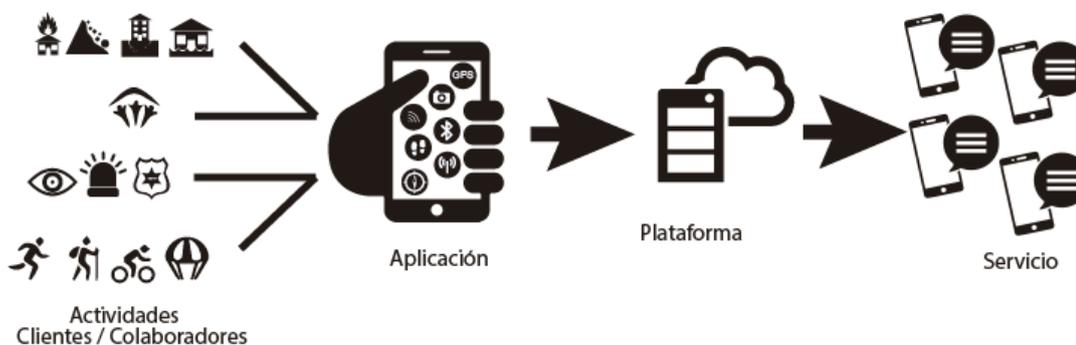


Figura 2.2: Arquitectura del Mobile Crowdsensing.

2.2.1 Colaboradores

Las personas que cumplen este rol, son aquellas que ayudan a capturar datos referentes al entorno donde se desarrolla una actividad en específica, permiten mantener un constante flujo de información, aunque no todas las aportaciones que son realizadas pueden ser reales, su veracidad podría depender de nivel de compromiso demostrado en su historial de aportes. Gracias a sus colaboraciones se podrá compartir a los demás usuarios de la plataforma una información oportuna y relevante.

2.2.2 Plataforma

Las solicitudes y los aportes realizados por los clientes y colaboradores respectivamente convergen a un solo destino el cual todos los datos se gestionan para proveer una información sólida y veraz. En esta parte de la estructura, los datos proporcionados son almacenados, ya que los participantes solo están encargados de emitir información, pero es necesario tener una base para recopilar dichos valores proporcionados para luego ser analizados y, después de ser procesados, emitir un resultado en forma de servicio el cual será presentado al cliente para que haga uso del mismo. Cabe señalar que dentro de la plataforma se realizan varias consultas de ser necesarias, a otros medios externos para

corroborar la veracidad de la información. La presentación de los datos como servicio es la parte fundamental de toda la arquitectura es por ello que se debe gestionar los datos de una forma clara y confiable.

2.2.3 Clientes

Comúnmente como conocido como usuarios finales, estas personas son quienes hacen uso ya de la información procesada de acuerdo a los requerimientos estipulados en el desarrollo de la plataforma, son capaces de dar a conocer los diferentes sucesos que ocurren en su entorno.

Las personas viven diariamente diferentes sucesos, realizan actividades, de las cuales se puede concentrar la atención y tratar de medir la importancia que tiene en cada uno de los seres humanos.

A través de los aportes de las personas, es posible capturar los datos de dichas actividades y por medio del uso de una aplicación, previamente instalada en un dispositivo móvil inteligente y detectar el tipo de actividad involucrada. En esta parte del proceso se ven involucrados los diferentes sensores del dispositivo, GPS, Cámara, Sensor de luz, etc.

Los datos que son capturados son enviados a un servidor, este puede estar establecido localmente en una red o tenerlo en una nube, los datos que necesitan ser enviados pueden hacerlo mediante una conexión a una red Wifi, Bluetooth, red móvil; en caso de que el servidor se encuentre localizado en la nube, el dispositivo necesita tener acceso a Internet. Una vez que los datos son recibidos se inicia el análisis y se procesan de acuerdo a las funciones establecidas, tal como se indica gráficamente en la figura 2.2.

Es importante señalar que el uso de un servidor potencializa a la aplicación ya que no consumen recursos del dispositivo y este no se ve afectado por procesos en cola adicionales.

Una vez que el servidor ha procesado los datos envía una respuesta a los requerimientos del usuario en caso de ser necesario. Recordando que el tipo de usuario puede ser personal o comunidad.

2.3 Ejemplo de Aplicaciones desarrolladas

Como antes se mencionó, en la actualidad se están generando muchas aplicaciones que son orientadas a las notificaciones de tiempo real enfocadas con las diferentes actividades desarrolladas por los diferentes grupos de población.

Es por ello que se aprovechan las características del medio para potencializar una aplicación, puede esta ser del campo ambiental, social, deportivo, etc.

De las aplicaciones destacables se encuentra Samrakshane, figura 2.3, la cual es una aplicación que recopila datos sobre los hábitos de las diferentes especies en estado salvaje, la idea consiste en desarrollar un medio del cual se mantenga informado la posición del avistamiento y si se encuentra en buenas condiciones o existe algún tipo de amenaza contra su estado natural. [9]



Figura 2.3: Interfaz de aplicación Samrakshane.

Una aplicación importante es Pier Trucker, figura 2.4 la cual a través de la captura de fotografías es posible detectar la congestión vehicular que existe por parte de

los tráileres que circulan en la ciudad y en los estacionamientos donde hacen base. [10]

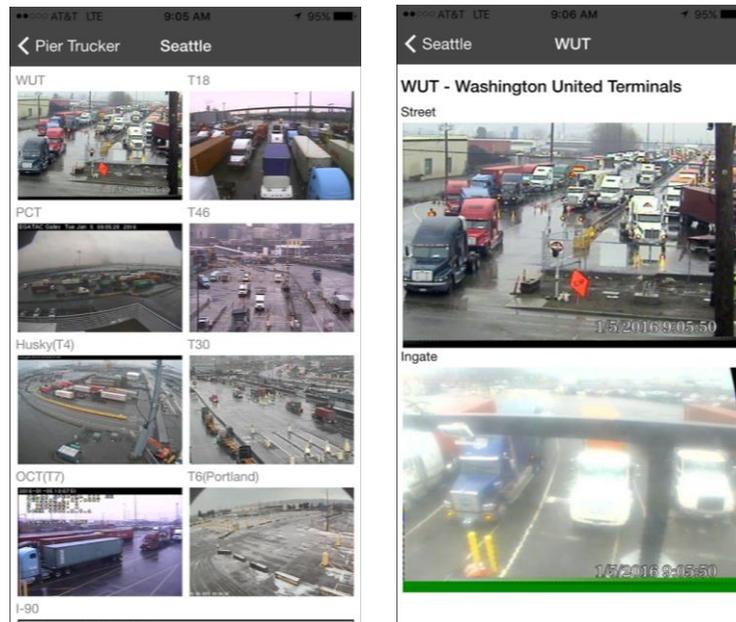


Figura 2.4: Interfaces de visualización de Pier Trucker.

En Latinoamérica se comienza a visualizar un crecimiento ante esta tecnología, Mobile Crowdsensing, y se comienza a dar apertura a los diferentes campos que involucran una comunidad. En Brasil, existe una aplicación de ciclismo, Bike Cidadão, figura 2.5, en la cual da a conocer una ruta para ciclismo así mismo el usuario tiene la facultad de poder compartir información a otros ciclistas como condiciones meteorológicas, alertas de puntos peligrosos en medio de la ruta y todo en tiempo real.

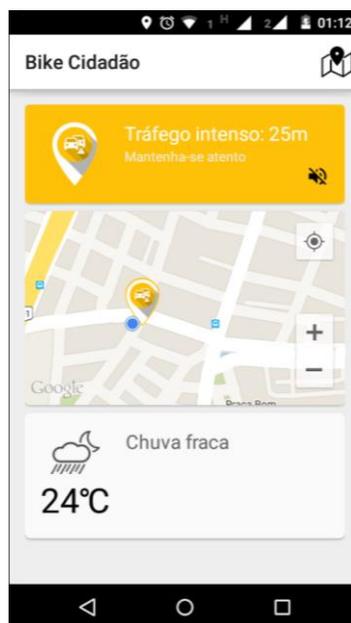


Figura 2.5: Notificación de la aplicación Bike Cidadão.

En Perú, dentro de las ciudades de Lima y Callao se ha implementado una aplicación en la que es posible conocer las diferentes rutas de transporte del medio urbano, de esta manera se crea una integración de todo el sistema de transporte, ver figura 2.6. Las personas que hacen uso del bus denominado Micro, y tienen activa la aplicación permiten conocer la ubicación y el tiempo aproximado de llegada a la próxima parada, y a su vez si existe algún tipo de congestionamiento vial.

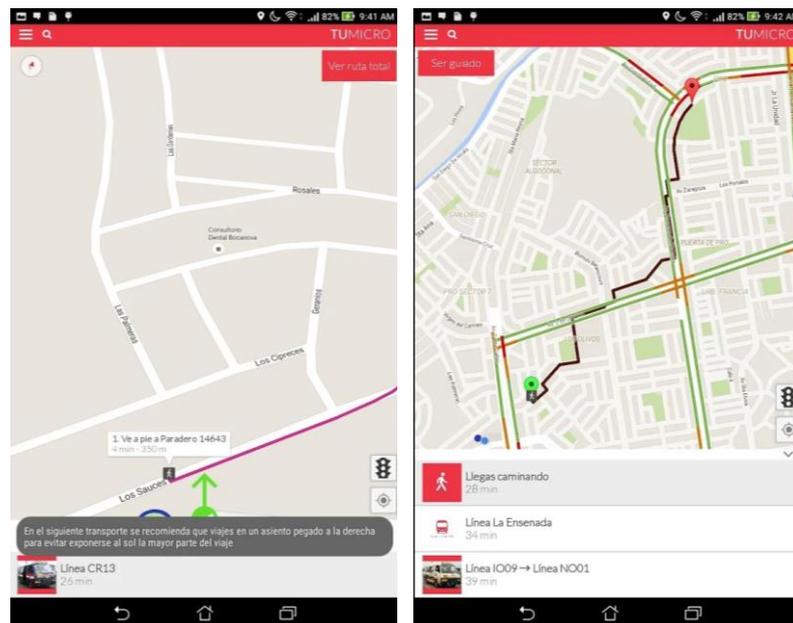


Figura 2.6: Interfaz de aplicación TuMicro

Es así como la tendencia de esta tecnología va creciendo y es notable la participación que los usuarios comienzan a tener ante de las diferentes aplicaciones que se van desarrollando e implementando en el mundo.

CAPÍTULO 3

3. DESARROLLO DEL SISTEMA.

En la actualidad, el uso de un dispositivo móvil es cada vez más frecuente y necesario. Se puede señalar que un 70% de la población hace uso de uno [11] [12].

Nuestro interés se sitúa en el desarrollo de un sistema para brindar una mejora a la problemática detallada por medio de uso del sensor de posicionamiento.

Debido a la gran demanda y popularidad de dispositivos móviles, entre los cuales se destacan los smartphone, representan una nueva oportunidad de acceso para los sistemas informáticos. El uso de la tecnología de Mobile Crowdsensing permite crear una alternativa en la cual, el usuario pueda obtener información de una fuente confiable sin necesidad de involucrar costos elevados de implementación ni mantenimiento de equipos por parte de la empresa que brinda el servicio, como lo hacen otros sistemas ya establecidos dentro del mercado comercial como los sensores de ultrasonido con indicadores leds, sistemas de cámaras, sistemas que demandan un elevado costo de inversión inicial para poder detallar información relevante al usuario del estacionamiento. Mobile Crowdsensing permite extender la información a los usuarios participes sin necesidad que se encuentren a la entrada del parqueadero de esta manera no existiría una pérdida de tiempo por parte del beneficiario al momento de saber dónde dejar estacionado su vehículo, del mismo modo el gasto por equipos es prácticamente nulo, ya que los usuarios hacen uso de su propio dispositivo móvil, siendo un mecanismo de solución ágil y prometedora al momento de suministrar información.

3.1 Metodología

Una de las características del crowdsensing son las recompensas que se entregan a los usuarios que realizan las colaboraciones para mantener actualizada la información por medio de la captura de datos, independientemente si es participativa u oportunista. Por lo tanto, forma parte de la metodología de trabajo en el despliegue del diseño.

Para el desarrollo conceptual del proyecto se ha establecido ciertos parámetros de los cuales dependerá la ejecución de la aplicación.

3.1.1 Parámetros

- El usuario debe haberse registrado en la aplicación previamente con sus datos personales, deberá registrar la placa del vehículo o los vehículos con los que suele ingresar al campus Gustavo Galindo, en caso de poseer.
- La aplicación se actualizará conforme se notifiquen los ingresos de los usuarios al parqueo y a su vez el usuario tendrá la información solicitada de dicho parqueo. Para seguridad del estudiante con referente a su vehículo se debe indicar la placa del vehículo con el que ingresó.
- Se realizará una validación por medio de posicionamiento de GPS al momento de encontrarse dentro del área geográfica del parqueo FIEC/FIMCP para registrar su ingreso al parqueo, esto nos permite estimar la cantidad de espacios libres restantes.
- Los usuarios de la aplicación que reportaran acerca de la disponibilidad de espacios libres.

3.1.2 Recompensas

- El usuario que ha completado la tarea asignada tendrá como recompensa, el poder visualizar la cantidad de parqueos libres de acuerdo a lo registrado en la base de datos. Las recompensas son accesibles al día siguiente del aporte.

3.2 Investigación de campo (encuestas y análisis)

Para poder desarrollar la propuesta, es necesario conocer la tendencia de dispositivos que los usuarios utilizan. Es por ello que se procedió a encuestas en diferentes ocasiones a dos grupos de estudiantes, usuarios generales y usuarios de parqueo.

3.2.1. Análisis de la primera encuesta: Usuario Generales

Bajo esta primera encuesta realizada se analizaron aspectos de simpatía con respecto a una propuesta por el desarrollo de una aplicación como propuesta a la solución del conocimiento de parqueos libres en determinada localidad, y posibles factores que están involucrados en la problemática. Por medio de los diagramas de barras de la figura 3.1 a la figura 3.5 se representan las respuestas de los participantes.

Los usuarios que colaboraron en el desarrollo de esta primera encuesta (Ver Anexo A) fueron aquellos estudiantes que conocen de la problemática que existe al momento de usar un parqueadero dentro del campus ya sea que hagan uso de un vehículo o hayan experimentado la necesidad de buscar un estacionamiento libre.

También se pudo constatar la preferencia del sistema operativo usado.

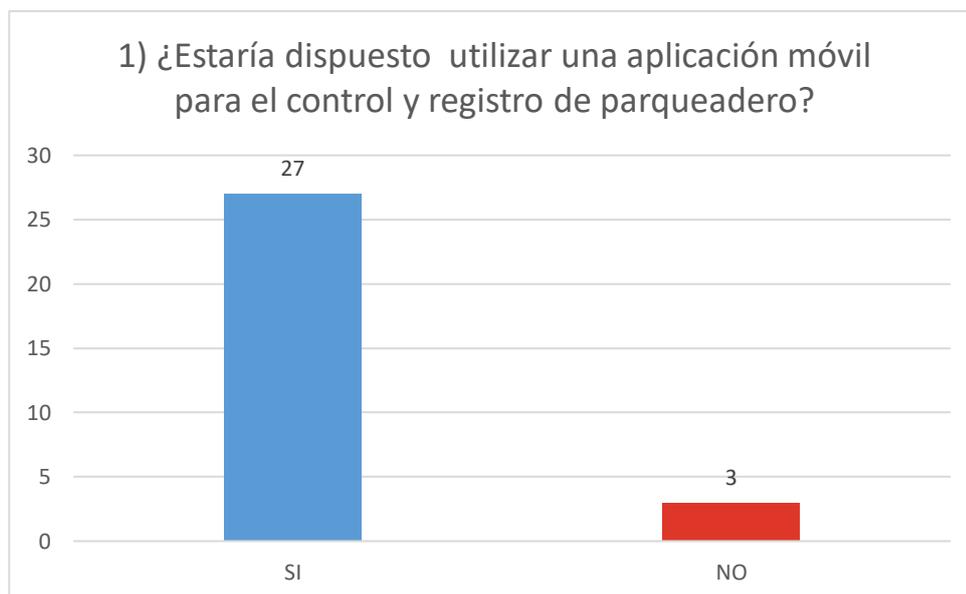


Figura 3.1: Gráfico estadístico.

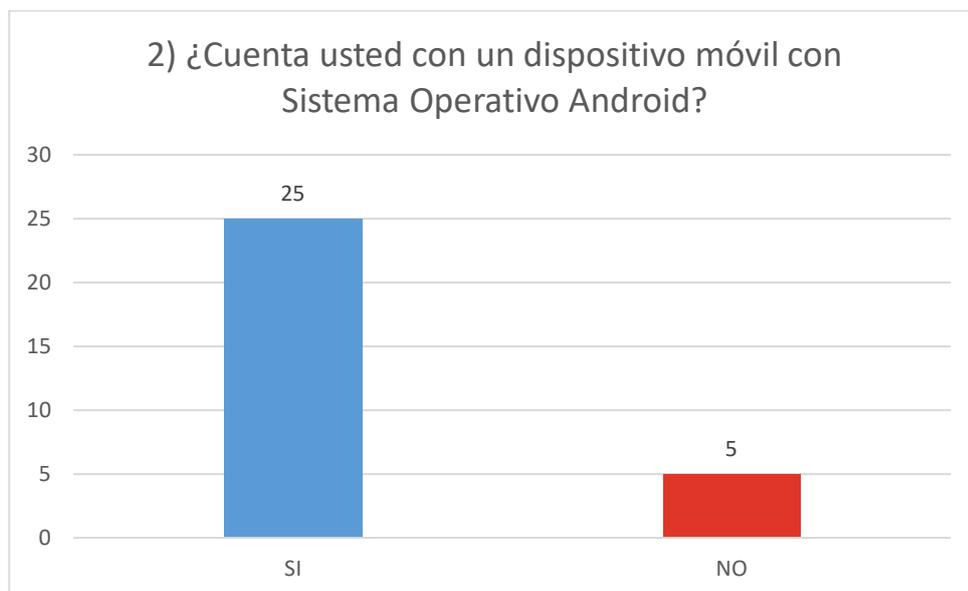


Figura 3.2: Gráfico estadístico.

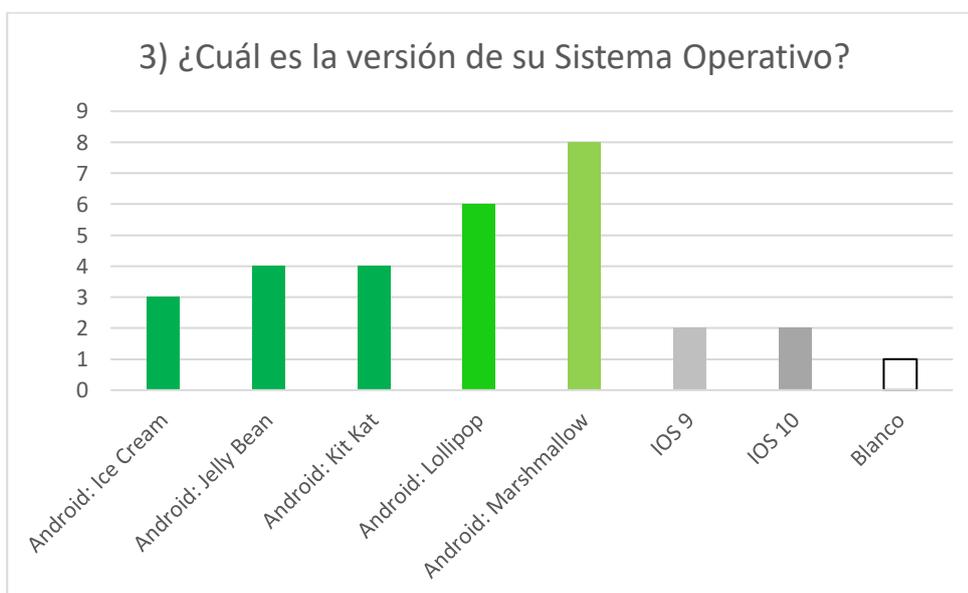


Figura 3.3: Gráfico estadístico.

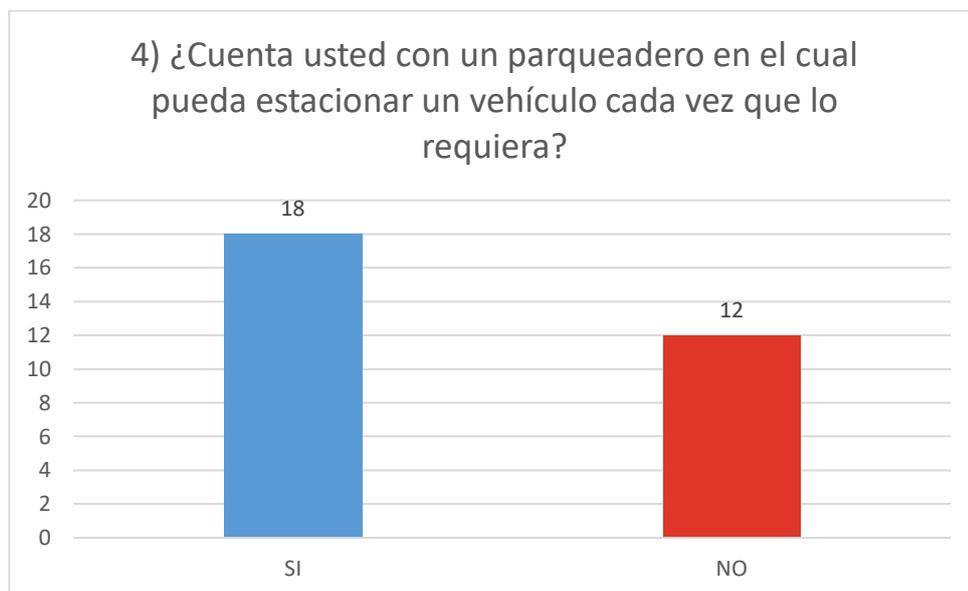


Figura 3.4: Gráfico estadístico.

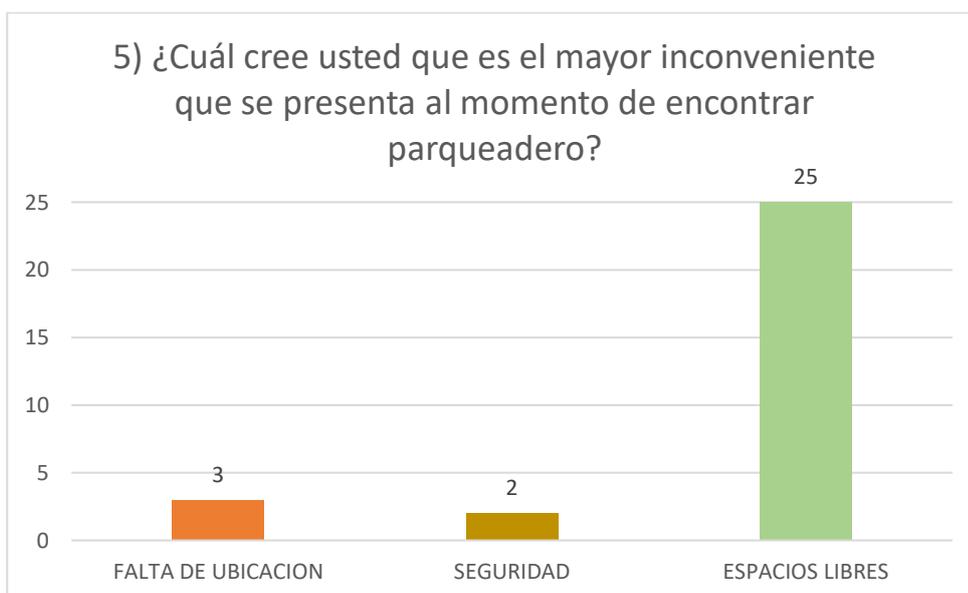


Figura 3.5: Gráfico estadístico.

De esta primera encuesta realizada, se ha analizado que gran cantidad de los participantes tienen buena predisposición al uso de una aplicación móvil en la que se efectúe un control y un registro de los vehículos que se encuentran estacionados dentro de las instalaciones de la universidad, así

mismo que el 83% de los usuarios destacan los espacios libres como el principal problema a la hora de buscar un lugar para dejar el vehículo.

3.2.2. Análisis de la segunda encuesta: Usuarios de Parqueo

La segunda encuesta (Anexo B) fue dirigida al público que hace uso del parqueo determinado para el estudio, FIEC/FIMCP. La finalidad de esta, fue determinar la frecuencia de uso de este espacio y el sistema operativo de los dispositivos móviles que frecuentemente usan.

En las figuras 3.6 a la 3.8 se visualiza el diagrama estadístico correspondiente a las respuestas de los participantes de la encuesta.

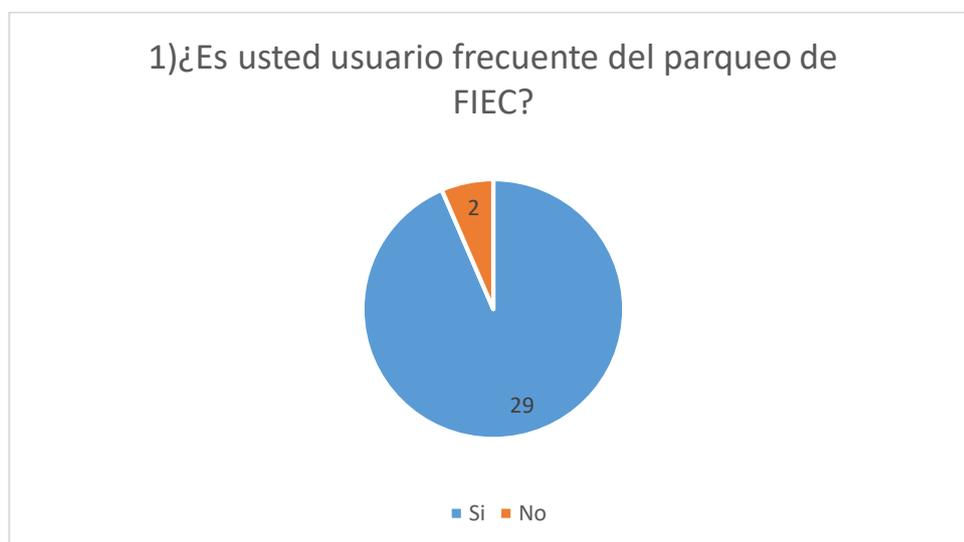


Figura 3.6: Gráfico estadístico.

2)¿Cual es su frecuencia de ingreso al parqueo en la semana?



Figura 3.7: Gráfico estadístico.

3)¿Cuál es marca de su dispositivo móvil?

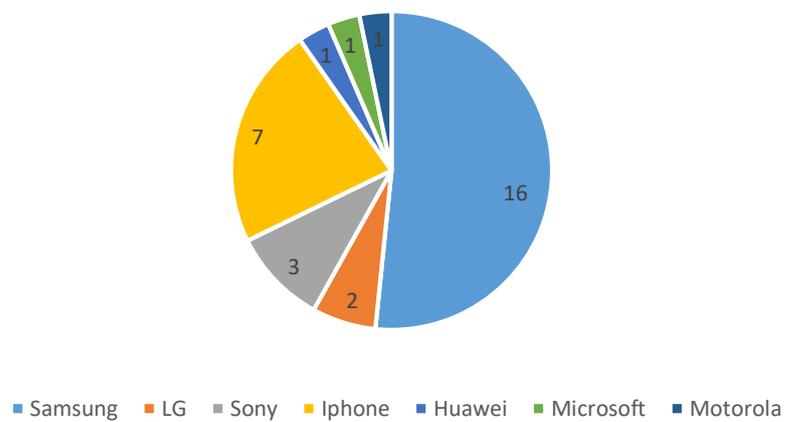


Figura 3.8: Gráfico estadístico.

A partir de estos datos fue posible determinar el sistema operativo usado de acuerdo a la frecuencia, tal como se visualiza en la figura 3.9.

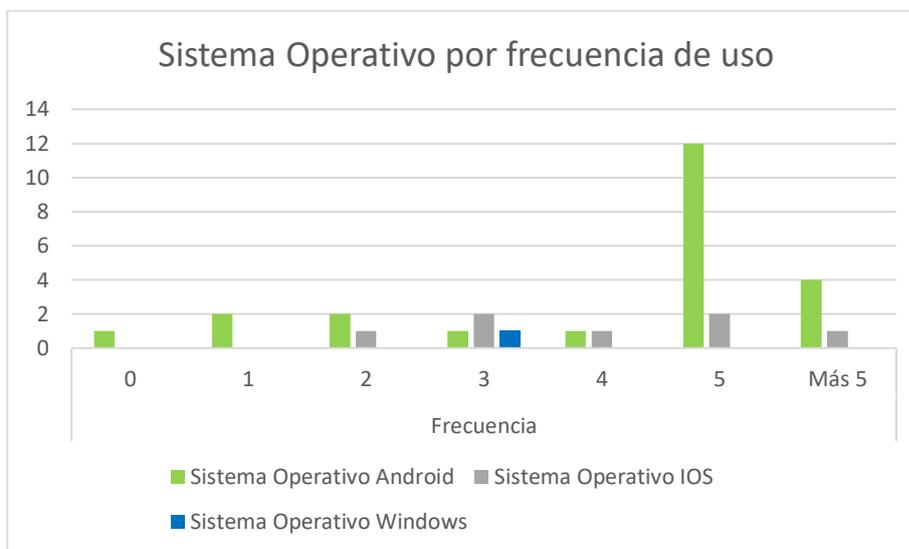


Figura 3.9: Gráfico estadístico.

Con estas estadísticas se ha procedido a establecer que el sistema operativo predominante es ANDROID. Cabe mencionar que los participantes de estas encuestas, han mencionado cual es el modelo de su dispositivo, de esta manera se ha podido determinar la versión del sistema operativo que usan.

Después de analizar las estadísticas, se destaca el sistema operativo ANDROID: JELLY BEAN como base para desarrollar la aplicación móvil.

Con ello también se pudo evidenciar los diferentes sensores que poseen estos dispositivos móviles, por lo tanto, se ha optado por la detección de datos a través del sensor de posicionamiento, GPS.

3.3 Descripción Conceptual / Estructura del proyecto

Basados en las implicaciones de Mobile Crowdsensing, la estructura del desarrollo del sistema de Mobile Crowdsensing involucra las siguientes fases:

- Desarrollo de una Aplicación Móvil: Proveer esta herramienta a los usuarios ayuda a suministrar la información que se detecta por parte de los usuarios considerando que es el vínculo para poder manejar los diferentes sensores del dispositivo.

- Diseño de una base de datos: Nos permite estructurar los datos recopilados de las diferentes entidades que se involucran en el desarrollo del sistema, así como sus relaciones.
- Implementación de un Servidor Web: Facilita el almacenamiento y las consultas que se generen ya que el servicio no es solo a nivel local, por lo tanto, se pretende brindar acceso a la plataforma desde cualquier localización.
- Aplicación del sistema GPS: Por medio del uso del sensor GPS del dispositivo se pretende generar un sistema de actualización de posicionamiento automático el cual ayude a verificar si se encuentra dentro del perímetro limitado para el estacionamiento.
- Validación de Placa del automotor: Atributo especial de la aplicación el cual permite optimizar el sistema al contar con una verificación previa al registro de vehículos con placas reales.

La finalidad de la aplicación es brindar servicio sin restricción de área de cobertura, es por ello que se tomó la decisión de tener un servidor remoto en una nube. Basado en Mobile Crowdsensing, la aplicación es de carácter participativa y comunidad.

3.4 Descripción del Sistema MCS

Siguiendo el curso de Mobile Crowdsensing, es muy significativo los roles que son desempeñados por parte de la comunidad, siendo estos clientes o trabajadores, porque gracias a la información que es receptada es posible estimar una realidad del evento que está aconteciendo y proporcionar una información mucho más certera. Es por ello que se necesita identificar al usuario que contribuye dentro de la plataforma y los datos con los que colabora.

La aplicación cuenta con diferentes interfaces que van vinculada con la actividad de este usuario dentro de la plataforma establecida.

- Información de perfil.
- Historial de registros en el parqueadero.

- Reporte.
- Consultas.

La aplicación debe establecer mediante localizaciones temporales si el usuario se encuentra dentro del perímetro establecido del estacionamiento.

Una vez que es detectado su ubicación, permite que el usuario haga el registro pertinente en caso de hacer uso del parqueo, de la misma manera al momento de realizar la salida del lugar donde estaba estacionado su vehículo. Así mismo para realizar un reporte es necesario encontrarse en el parqueadero o cerca del mismo.

Los aportes que realicen los usuarios en el transcurso de la jornada, establecida en el periodo de 7H00 – 18H00, serán almacenados para ser analizados posteriormente y estimar el promedio de uso del estacionamiento en el instante de las contribuciones. Las personas que hayan colaborado, podrán conocer la cantidad disponible de parqueos al día siguiente, pero con los valores estimados del día en curso, esta consulta es posible como método de recompensa por sus aportaciones.

3.5 Elementos de Programación Utilizados

Al tener en consideración todos los requerimientos que se van a desarrollar en la plataforma de nuestra aplicación, la información que se maneja dentro de nuestro sistema es posible que sea transmitida de manera clara y precisa sobre todo en la actualidad, ya que la vinculación de diversas técnicas tecnológicas se hace más necesarias. Hay que tener en cuenta que la captura de los datos que serán transmitidos dependerán de la participación del usuario final, por ello es conveniente generar una aplicación que sea comprensible y de fácil manejo.

Los componentes que se involucran al momento de crear la aplicación móvil como medio de interacción con el usuario son:

- **Java:** Lenguaje multiplataforma creado a partir de 1991 por la compañía Sun Microsystems. Este lenguaje es orientado a objetos, además permite crear una amplia gama de programas que pueden ser

dispuestos en diferentes dispositivos y su principal virtud es funcionar en los diferentes sistemas operativos existentes sin que haya una distinción una vez que este ha sido compilado. [13] [14]

- **API de Google para Android:** Es un conjunto de reglas, funciones y procedimientos que nos permiten trabajar con una estructura ya creada y hacer uso de sus herramientas dentro de otro software para facilitar la manipulación de los datos. Son versátiles y multiplataforma. [15] [16]

Por otra parte, se encuentran aquellos elementos que permiten la manipulación y debido proceso de los datos que son recopilados para dar un servicio adecuado de respuesta ante el requerimiento:

- **Django:** Es un web framework, gratuito y de código abierto, en el cual se encuentran establecidas ciertas características que son comunes al momento de desarrollar una web, adicionalmente este framework maneja lenguaje Python, y es de alto nivel. Los diferentes métodos disponibles dentro de esta herramienta permiten crear o modificar sitios web ya que no funciona como una plantilla para creación de websites y de esta manera permite crear un código fluido y con menos cantidad de líneas. [17]
- **Python:** Es un lenguaje de programación orientado a objetos, el cual es diseñado para tener una sintaxis fácil de manejar y de entender. Es multiplataforma y no necesita ser compilado, este lenguaje es empleado no tan solo al manejo de sitios web sino también para el desarrollo de diversas aplicaciones como los video juegos o sistemas de control, esto es debido a su fácil manipulación de datos ya que consta con muchas librerías que permiten acceder a los diferentes servicios posibles para ofrecer. [18] [19]
- **Sqlite:** Es un manejador de base de datos, su software es de código abierto, usado principalmente en dispositivos en los cuales requieran una base de datos embebida. Se genera un archivo en el cual contiene toda la estructura de la base de datos. La información que se almacena está comprendida desde objetos hasta las composiciones que

establece el usuario. Su cualidad a remarcar es que soporta sintaxis de las funciones SQL. [20] [21]

- **Json:** Abreviación proveniente de, JavaScript Object Notation, es un formato en el que los datos son presentados para un intercambio de información. Su estructura de presentación de datos consta de los nombres de las variables que se relacionan con la consulta y los valores que contienen cada una de ellas, haciéndolo sencillo y ligero de comprender a la hora de manipular la información necesaria. Este formato es usado popularmente en los requerimientos de consultas a servidores web y es una alternativa mucho más práctica que alguna otra en el mercado actualmente ya que cualquier lenguaje de programación consta con librerías que facultan el manejo de este formato sin ninguna restricción para la parte que hace la solicitud y la que envía una respuesta. [22] [23]

3.6 Funcionalidad

El desarrollo del proyecto consta de dos partes importantes los cuales detallan procesos pertenecientes al desarrollo de la aplicación y por otra parte los que se encuentran implementados en el servidor. Entre ambos se destacan los procesos que se señalan a continuación.

3.6.1 Servidor

Es un servicio de alojamiento de los datos, en el cual consta la base de datos y los procesos a realizar con los mismos.

Se ha establecido el framework de Django para levantar los servicios de comunicación entre la aplicación móvil y los datos que se encuentran hospedados en la base de datos tal como se observa en la figura 3.12 y figura 3.13.

Uno de los datos más importante es la placa del vehículo ya que no es posible poder gestionar una plataforma que manejará información sensible de los usuarios proporcionando datos inválidos.

- **Base de datos:** La gestión de los datos a recopilarse deben ser almacenados dentro de un sistema de archivos de una manera estructurada. Es por ello que se procedió a evaluar las diferentes entidades que se involucran, así como las respectivas relaciones que permiten asociar los atributos de unas entidades con otras. En la figura 3.10 se visualiza el esquema usado para el desarrollo del proyecto.

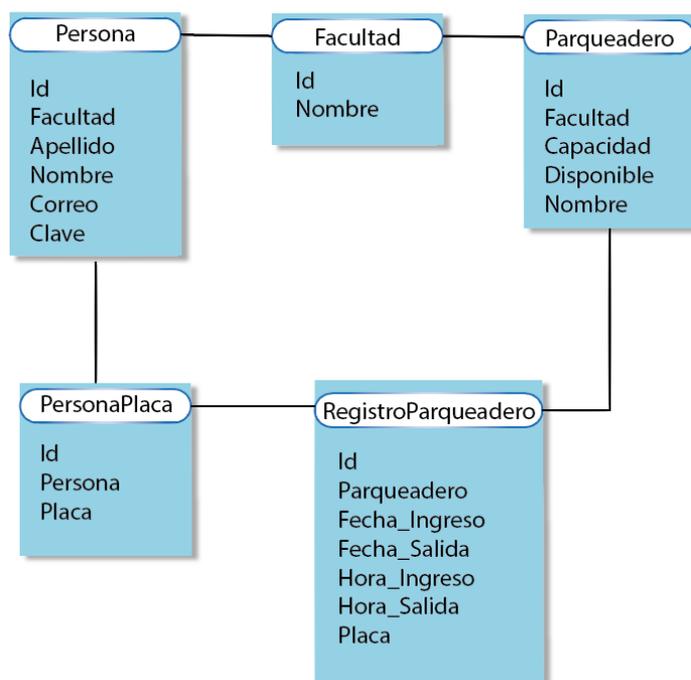


Figura 3.10: Tablas de la base de datos del sistema.

- **Validación de placa:** Al momento de registrar una placa vehicular se realiza un procedimiento de validación a la Agencia Nacional de Transito (ANT), a través de su sitio web de consultas.

El link de enlace es el siguiente:

https://sistemaunico.ant.gob.ec:5038/PortalWEB/paginas/clientes/clp_criterio_consulta.jsp, como se observa en la figura 3.11.

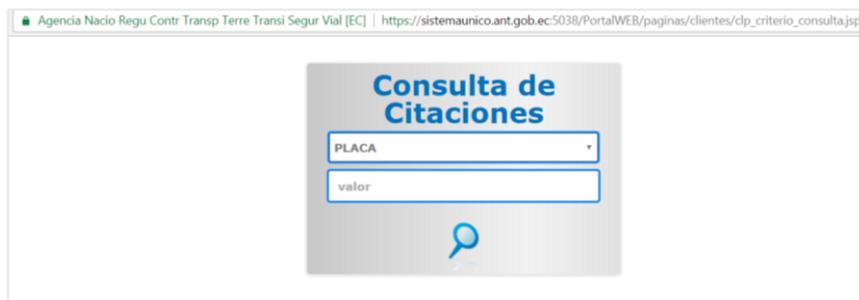


Figura 3.11: Captura de interfaz de la página de consulta de ANT.

- **Validación de cantidad máxima de parqueo:** El sistema cuenta con un número determinado para estacionamiento vehicular, el mismo que siempre se verifica al realizar un registro ya que no es posible promover un sistema con una debilidad en la que se rebase el límite.

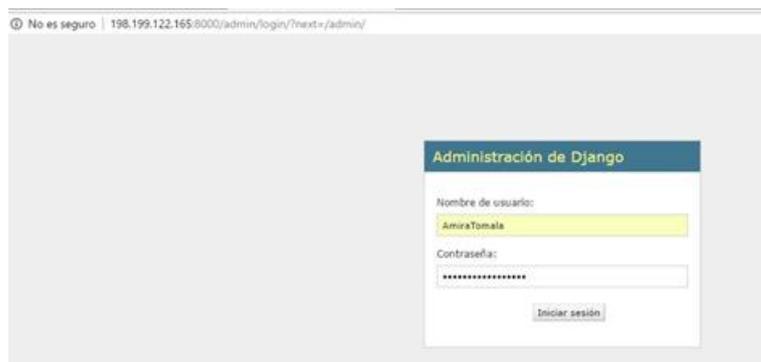


Figura 3.12: Captura pantalla ingreso al servidor con credenciales autorizadas.



Figura 3.13: Pantalla de administración del framework Django.

3.6.2 Aplicación Móvil

Android Studio, es el entorno de desarrollo integrado oficial en el cual se pueden crear aplicaciones para el sistema Android. Brinda herramientas y frameworks de prueba, soporta varias versiones al momento de desarrollar una aplicación. Entorno utilizado para el desarrollo de la aplicación el cual maneja dos partes, el diseño de interface y la programación de los eventos que se desarrollan en ella. Entre los procesos propios de la aplicación, se tiene la detección de la ubicación ya que depende totalmente del sensor de GPS perteneciente al dispositivo.

- **Determinación de geolocalización**

- **Algoritmo del polinomio:** Este algoritmo permite determinar un área a través de varios puntos de localización, de esta forma nos permite trabajar bajo una región irregular, como lo es esta locación. El algoritmo Ray Casting permite determinar si un punto se encuentra dentro del polígono dependiendo del número de intersecciones con los lados del mismo, si al finalizar es un número impar, quiere decir que se encuentra dentro del polígono [24].

Seudocódigo del algoritmo empleado: [25]

```
{
para(i = 0 hasta i < N)
{ /* Se elimina del vértice "i" el eje correspondiente a la coordenada
dominante. Se obtiene (ui, vi) */
    proyectar_vertice(i)
/* Se traslada el vértice hacia el origen de UV:(ui, vi) - ri */
    trasladar_origen(i) }
/* Se inicializa el contador de cruces */
cruces = 0
para(cada Pa, Pb) /* Para cada arista del polígono */
```

```
{ /* Se comparan los signos de v'a y v'b para averiguar si la arista
cruza el eje U. Si son opuestos cruza el eje U */
```

```
    si(signo(v'a) ≠ signo(v'b))
```

```
/* Se comprueba si el cruce está en U+ */
```

```
    si((u'a > 0) y (u'b > 0))
```

```
        cruces++
```

```
    sino
```

```
        si((u'a > 0) o (u'b > 0))
```

```
            si(u'a - v'a·(u'b - u'a)/(v'b - v'a) > 0)
```

```
                cruces++
```

```
}
```

```
si(cruces es impar)
```

```
    la intersección está dentro del polígono
```

```
sino
```

```
    la intersección está fuera del polígono
```

```
}
```

- o **Selección de puntos de localización:** El área que se selecciona o solo involucra el perímetro del parqueadero, como se puede observar en la figura 3.14, es importante expandir el perímetro unos cuantos metros, en el caso de la aplicación se dispuso de una expansión de 20 metros, de esta forma se puede verificar que los usuarios que hagan un reporte estén cerca de dicha zona.

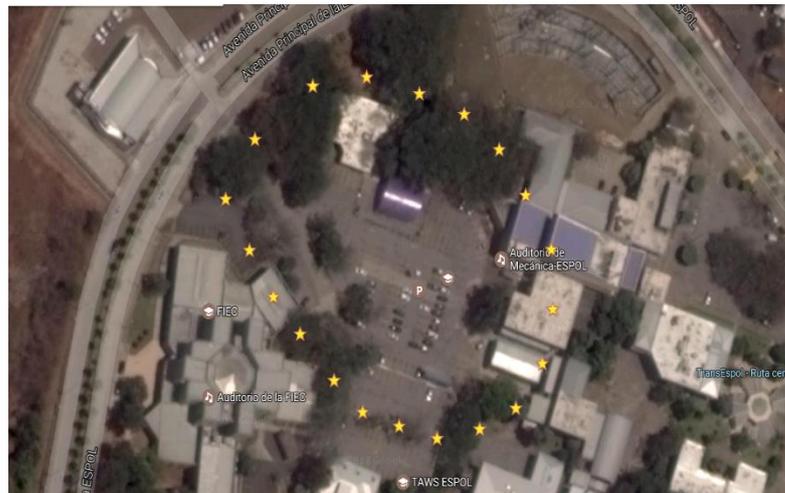


Figura 3.14: Vista aérea del parqueadero FIEC/FIMCP con los puntos de longitud y latitud correspondientes.

3.7 Resultados

3.7.1 Prueba de registro de usuarios

Se procedió a realizar registro de varios usuarios al mismo tiempo y en intervalos, no se visualizaron problemas al comunicarse con el servidor y fueron registros exitosos como se observa en las figuras 3.15 y 3.16.

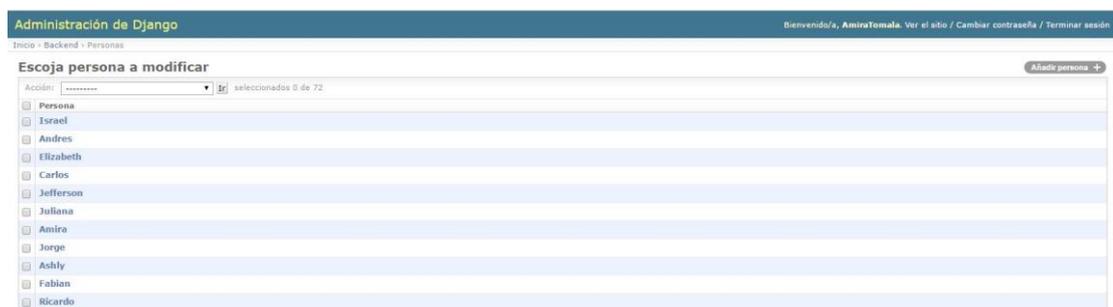


Figura 3.15: Pantalla con los usuarios registrados en la base de datos.

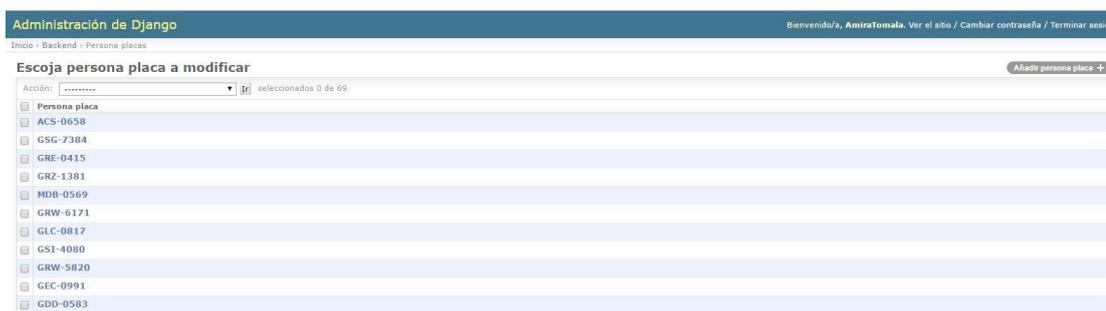


Figura 3.16: Consulta del registro de las placas de los usuarios registrados.

3.7.2 Prueba de localización

Para llevar a cabo esta prueba, se procedió a enviar mensajes definidos como *TOAST*, los cuales nos permitían verificar la ubicación enviada a través de actualizaciones constantes y validar si la ubicación, latitud y longitud, pertenecen al perímetro determinado inicialmente. Como se puede observar en la figura 3.17 se ha comprobado que la aplicación sitúa correctamente al usuario dentro del parqueadero y el perímetro extendido permitiendo hacer el correcto registro en la base de datos y así como el acceso a la sección de reportes, ver figura 3.18.



Figura 3.17: Captura de Pantalla del dispositivo móvil al estar situado dentro del perímetro.

CNT 10:44 AM 87%

Reportar

Facultad
Fiec

Parqueo
Parqueo Fiec 1

Parqueos Disponibles 89

GUARDAR

Figura 3.18: Captura de pantalla del dispositivo. Reporte de sitios libres.

Dentro de este procedimiento los valores medidos teóricamente y los datos experimentales recopilados en las tablas 6 y 7, su representación cartesiana en la figura 3.19. es posible verificar el margen de error entre las mismas como se detalla en la figura 3.20 Se procede a realizar la conversión de la posición del GPS a unidades UTM, ya que dicho proceso permite trabajar con el sistema métrico y esto facilita el uso de coordenadas en plano cartesiano.

Dentro de las coordenadas UTM, el Ecuador se encuentra en la Zona 17, hemisferio Sur.

COORDENADAS TEORICAS

	X(metros)	Y (metros)
1	614907,240	9762930,729
2	614877,7659	9762944,081
3	614882,1818	9762944,94
4	614885,9653	9762948,20
5	614892,5135	9762944,70
6	614926,1743	9762954,871
7	614955,307	9762913,251
8	614880,8068	9762951,088

9	614878,6578	9762947,352
10	614876,1666	9762947,321
11	614875,1759	9762946,106
12	614875,1866	9762945,387
13	614875,7324	9762930,20
14	614873,3405	9762929,115
15	614870,0609	9762930,919
16	614863,5327	9762931,233
17	614851,2628	9762943,491
18	614859,5145	9762959,216
19	614838,6354	9762887,484
20	614858,0915	9762960,024
21	614857,948	9762961,70
22	614855,7193	9762971,678
23	614860,1535	9762983,194
24	614849,909	9762980,968
25	614822,6205	9762986,691
26	614800,8844	9762963,512
27	614817,5363	9762967,713
28	614822,6048	9762963,365
29	614830,8034	9762949,916

Tabla 6: Datos teóricos en coordenadas UTM.

COORDENADAS EXPERIMENTALES		
	X(metros)	Y(metros)
1	614907,240	9762930,729
2	614877,7659	9762944,081
3	614882,1818	9762944,94
4	614885,9653	9762948,20
5	614892,5135	9762944,70
6	614926,1743	9762954,871
7	614955,307	9762913,251
8	614880,8068	9762951,088
9	614878,6578	9762947,352
10	614876,1666	9762947,321
11	614875,1759	9762946,106
12	614875,1866	9762945,387
13	614875,7324	9762930,20
14	614873,3405	9762929,115
15	614870,0609	9762930,919
16	614863,5327	9762931,233

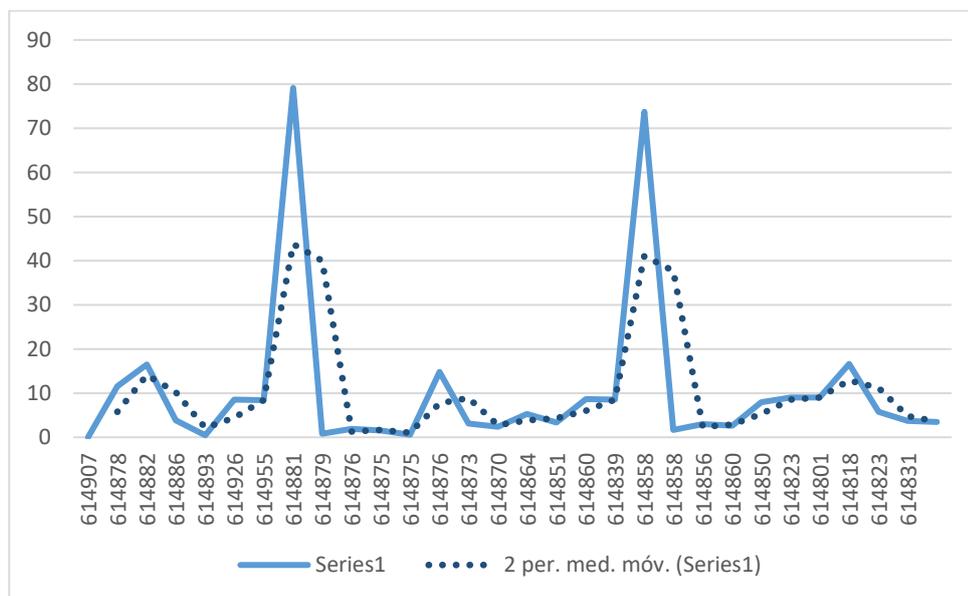


Figura 3.20: Variación del error entre valores teóricos y experimentales con el promedio entre sus valores.

3.7.3 Prueba de registro en el parqueadero

Una vez que el usuario se ha registrado en el parqueadero se procede a ejecutar un decremento de la capacidad del estacionamiento FIEC/FIMCP y el respectivo registro en la base de datos, ver figura 3.21.

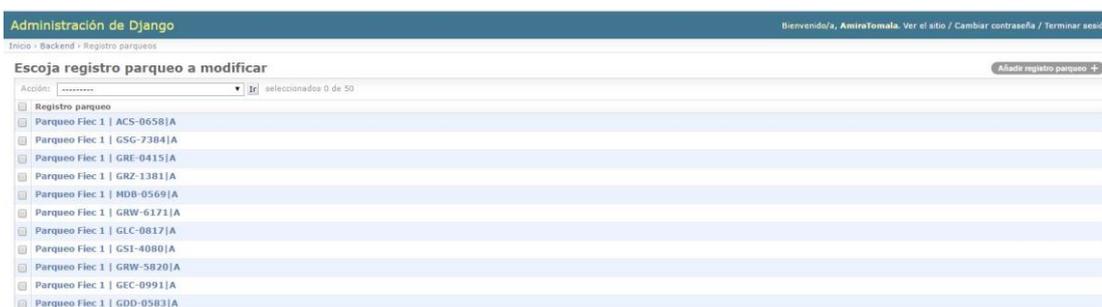


Figura 3.21: Consulta a la base de datos con los usuarios registrados en el parqueadero FIEC/FIMCP.

3.7.4 Prueba de reportes recibidos

Se verifica que los datos proporcionados por los usuarios pueden salir de la tendencia, por ende, estos valores atípicos fueron extraídos de los cálculos para promediar la disponibilidad y obtener un valor estimado cercano a la realidad, ver figura 3.22. Las pruebas se realizaron en tres grupos, tal como se puede observar en la tabla 8. El primer grupo hizo pruebas en el horario de 9H30 – 10H30, el segundo grupo realizó las pruebas de 11H20 – 12H10 y las pruebas realizadas por el tercer grupo en el horario de 13H10 – 14H15.

Cantidad disponible de espacio en el parqueo										
Valores Reportados	10	20	10	9	25	7	12	9	11	5
Grupo	1	1	1	2	2	2	3	3	3	3

Tabla 8: Valores reportados por usuarios.

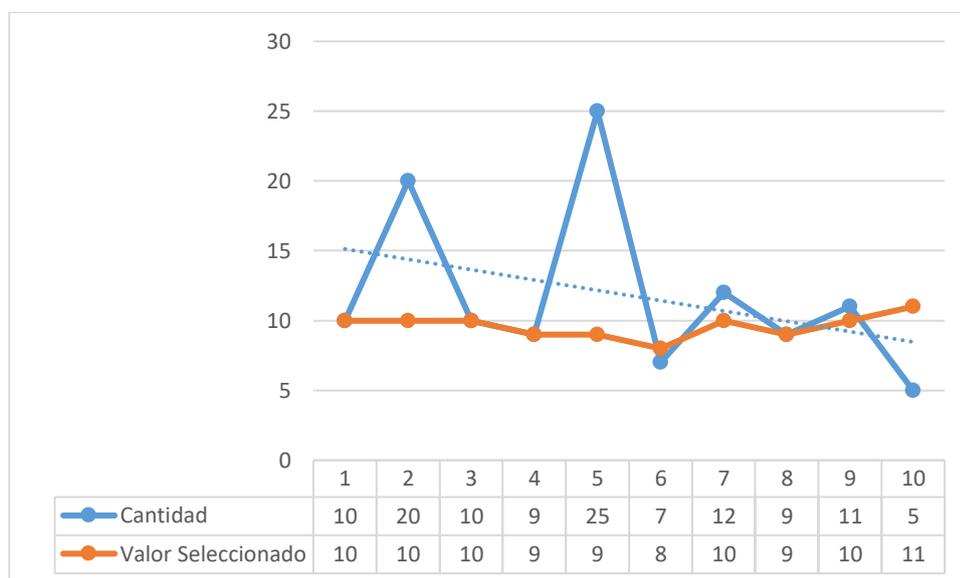


Figura 3.22: Gráfico de reportes.

3.7.5 Resultados de la encuesta de satisfacción

En la figura 3.23, se muestra la valoración recibida del uso de la aplicación a través de la encuesta en línea. De la misma se puede detallar que la aplicación tiene un alto grado de aceptación de los usuarios por su práctica solución y forma de reportar.

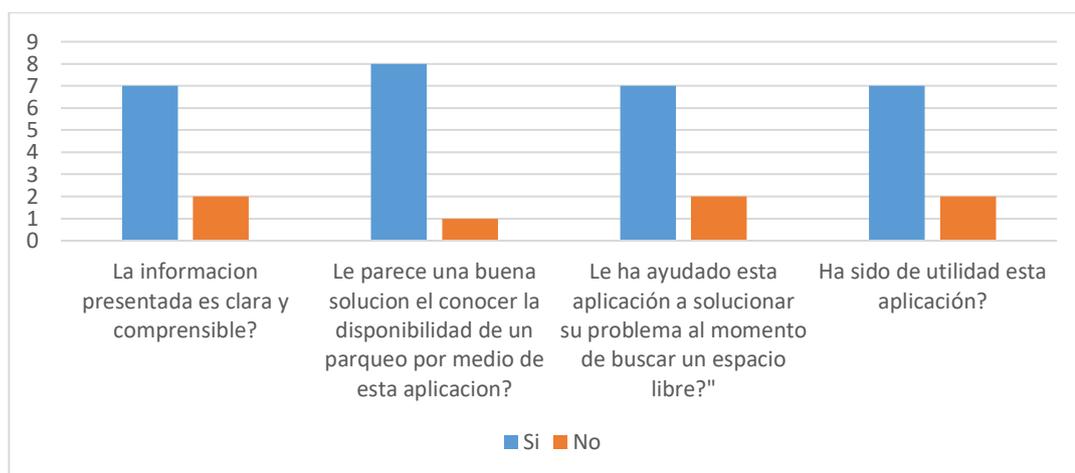


Figura 3.23: Gráfico estadístico correspondiente a las respuestas de los usuarios de la encuesta de satisfacción.

CONCLUSIONES Y RECOMENDACIONES

Es posible crear y fomentar un ambiente de participación activa por parte de la comunidad politécnica, y de esta manera contribuir con el desarrollo de las diferentes tecnologías que se desenvuelven en nuestro diario vivir.

Eventualmente la aplicación, será parte de un mecanismo de información y solución ante la problemática existente acerca de la disponibilidad de parqueaderos cercanos a una unidad académica.

La funcionalidad y la veracidad de la información mostrada dependerá de cuan activa e involucrada se encuentre los usuarios del campus Gustavo Galindo ya que la plataforma requiere una interacción humana.

Existe el caso que hay usuarios que no tienen un carro pero son participes en la aplicación, reportando la disponibilidad de parqueos en el estacionamiento por lo tanto, en caso de estar entre los primeros automáticamente tienen la recompensa de poder consultar la disponibilidad del parqueadero, debido a esto se sugiere a la unidad académica o a la institución tenerlos en consideración para que reciban algún tipo de incentivo ya que están fomentando la integración de la comunidad a pesar de que no reciben algún beneficio para sí.

Para futuros análisis y desarrollos se sugiere que los lugares destinados para el estacionamiento vehicular cuenten con una numeración adecuada, ya que sería posible desarrollar un mecanismo más preciso en el conteo de los lugares e identificación de los mismos a través de un sistema de detección de imagen, lector de código QR o detección NFC.

BIBLIOGRAFÍA

- [1] E. Universo, «El Universo,» 7 Enero 2013. [En línea]. Available: <http://www.eluniverso.com/2013/01/07/1/1445/nuevos-sistemas-parqueos-centros-comerciales.html>. [Último acceso: 20 Enero 2017].
- [2] TENET, «Sistema de dirección ultrasónico inteligente del estacionamiento del sensor DC 12V para el distrito comercial,» [En línea]. Available: <http://spanish.vehicleparkingsystem.com/sale-4009654-intelligent-ultrasonic-sensor-parking-guidance-system-dc-12v-for-commercial-district.html>. [Último acceso: 19 Diciembre 2016].
- [3] D. Torres, «Sensor de espacios de estacionamiento mediante camara web,» [En línea]. Available: <https://www.youtube.com/watch?v=qW9tPijAbls>. [Último acceso: 19 Diciembre 2016].
- [4] M. D. Márquez Rosero, «Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE,» Febrero 2015. [En línea]. Available: La información sobre los parqueaderos en tiempo real permite disminuir el tráfico que es causado por la búsqueda de los mismos, ya que al direccionar al usuario de forma inmediata al lugar que contiene un sitio disponible para aparcar el auto, se elimina. [Último acceso: 06 Diciembre 2016].
- [5] I. Hall, «Google Play,» 7 Octubre 2016. [En línea]. Available: <https://play.google.com/store/apps/details?id=com.innovativehall.ghc>. [Último acceso: 16 Enero 2017].
- [6] TuRuta, «TuRuta,» [En línea]. Available: <http://tumicro.pe/>. [Último acceso: 21 Enero 2017].
- [7] P. d. ESCIn, «Google Play,» 13 Agosto 2016. [En línea]. Available: <https://play.google.com/store/apps/details?id=br.ufpe.cin.contexto.bikecidao>. [Último acceso: 16 Enero 2017].

- [8] G. Inc., «Google Play,» 23 Enero 2017. [En línea]. Available: <https://play.google.com/store/apps/details?id=com.google.android.apps.village.boond>. [Último acceso: 27 Enero 2017].
- [9] Pakshi, «Google Play,» 26 Noviembre 2013. [En línea]. Available: <https://play.google.com/store/apps/details?id=com.rakshane.samrakhshane>. [Último acceso: 22 Enero 2017].
- [10] V. Services, «Google Play,» 11 Septiembre 2016. [En línea]. Available: <https://play.google.com/store/apps/details?id=com.vladservices.piertrucker>. [Último acceso: 16 Enero 2017].
- [11] Ericsson, «Ericsson,» 03 Junio 2015. [En línea]. Available: <https://www.ericsson.com/news/1925907>. [Último acceso: 10 Enero 2017].
- [12] C. Sturm, «Wayerless,» 24 Febrero 2010. [En línea]. Available: <https://www.wayerless.com/2010/02/onu-dos-tercios-de-la-humanidad-usan-telefonos-moviles/>. [Último acceso: 03 Enero 2017].
- [13] M. A. Alvarez, «Qué es Java,» 18 Julio 2001. [En línea]. Available: <https://desarrolloweb.com/articulos/497.php>. [Último acceso: 19 Diciembre 2016].
- [14] «Preguntas Generales,» Java, [En línea]. Available: https://www.java.com/es/download/faq/whatis_java.xml. [Último acceso: 20 Diciembre 2016].
- [15] M. Merino, «¿Qué es una API y para qué sirve?,» TICbeat, 12 Julio 2014. [En línea]. Available: <http://www.ticbeat.com/tecnologias/que-es-una-api-para-que-sirve/>. [Último acceso: 19 Diciembre 2016].
- [16] andrearrs, «¿Qué es una API?,» hipertextual, 15 Mayo 2014. [En línea]. Available: <https://hipertextual.com/archivo/2014/05/que-es-api/>. [Último acceso: 19 Diciembre 2016].
- [17] D. Girls, «¿Qué es Django?,» [En línea]. Available: <https://tutorial.djangogirls.org/es/django/>. [Último acceso: 26 Diciembre 2016].

- [18 M. A. Alvarez, «Qué es Python,» 19 Noviembre 2003. [En línea]. Available:
] <https://desarrolloweb.com/articulos/1325.php>. [Último acceso: 19
Diciembre 2016].
- [19 C. S. Roldán, «¿Qué es Python?,» CodeJobs, 02 Marzo 2013. [En línea].
] Available: <https://www.codejobs.biz/es/blog/2013/03/02/que-es-python>.
[Último acceso: 20 Diciembre 2016].
- [20 F. Rómmel, «SQLite: La Base de Datos Embebida,» [En línea]. Available:
] [https://sg.com.mx/revista/17/sqlite-la-base-datos-
embebida#.WKTaZ_nhDb1](https://sg.com.mx/revista/17/sqlite-la-base-datos-embebida#.WKTaZ_nhDb1). [Último acceso: 21 Diciembre 2016].
- [21 J. Revelo, «Tutorial De Bases De Datos SQLite En Aplicaciones Android,»
] 20 Octubre 2014. [En línea]. Available:
[http://www.hermosaprogramacion.com/2014/10/android-sqlite-bases-de-
datos/](http://www.hermosaprogramacion.com/2014/10/android-sqlite-bases-de-datos/). [Último acceso: 20 Diciembre 2016].
- [22 «Introducción a JSON,» [En línea]. Available: [http://www.json.org/json-
es.html](http://www.json.org/json-es.html). [Último acceso: 20 diciembre 2016].
- [23 M. P. ESTESO, «JSON, PROGRAMACIÓN WEB, SERIE JSON,
] TUTORIALES,» [En línea]. Available: [https://geekytheory.com/json-i-que-
es-y-para-que-sirve-json/](https://geekytheory.com/json-i-que-es-y-para-que-sirve-json/). [Último acceso: 10 Enero 2017].
- [24 Sakseiw, «Punto dentro de polígono,» 4 Octubre 2013. [En línea].
] Available: [https://sakseiw.wordpress.com/2013/10/04/punto-dentro-de-
poligono/](https://sakseiw.wordpress.com/2013/10/04/punto-dentro-de-poligono/). [Último acceso: 17 Enero 2017].
- [25 R. Ramos, «Ray casting avanzado,» [En línea]. Available:
] <http://di002.edv.uniovi.es/~rr/Tema4.pdf>. [Último acceso: 28 Diciembre
2016].
- [26 C. Borcea, M. Talasia y R. Curtmola, Mobile Crowdsensing, NJ: CRC
] Press, 2016.
- [27 C. Borcea, M. Talasila y R. Curtmola, «Mobile Crowd Sensing,» NJ.
]

- [28 D. D. Kliazovich y P. P. Bouvry, «Mobile Crowdsensing».
]
- [29 G. Cardone, L. Foschini, P. Bellavista y A. Corrad, «Fostering
] ParticipAction in Smart Cities:A Geo-Social Crowdsensing Platform,»
IEEE Communications Magazine , pp. 112-119, 2013.
- [30 S. B. Jimenez, «Crowdsensing en Ciudades Inteligentes,» Ciudad Real -
] España, 2015.
- [31 R. K. Ganti, F. Ye y H. Lei, «Mobile Crowdsensing: Current State and,»
] Hawthorne, NY.
- [32 B. GUO, Z. WANG, Z. YU, Y. WANG, N. Y. YEN, R. HUANG y X. ZHOU,
] «Mobile Crowd Sensing and Computing: The Review of an Emerging
Human-Powered Sensing Paradigm,» *ACM Computing Surveys*, vol. V, nº
N, 2015.
- [33 Y. Wang, X. Jia, Q. Jin y J. Ma, «Mobile crowdsourcing: framework,
] challenges, and solutions,» 2016.
- [34 P. P. Jayaraman, C. Perera, D. Georgakopoulos y A. Zaslavsky, «Efficient
] Opportunistic Sensing using Mobile Collaborative Platform MOSDEN,»
Canberra, Australia, 2013.
- [35 «Dispositivos Inteligentes Requieren Sistemas Más Inteligentes de
] Pruebas Automatizadas,» National Instruments, 14 Diciembre 2016. [En
línea]. Available: <http://www.ni.com/white-paper/53075/es/#toc1>. [Último
acceso: 25 Enero 2017].
- [36 eleconomista.com.mx, «DISPOSITIVOS INTELIGENTES SIGUEN
] MARCANDO TENDENCIA A NIVEL MUNDIAL,» 21 Agosto 2015. [En
línea]. Available:
[http://techo.americaeconomia.com/articulos/dispositivos-inteligentes-
siguen-marcando-tendencia-nivel-mundial](http://techo.americaeconomia.com/articulos/dispositivos-inteligentes-siguen-marcando-tendencia-nivel-mundial). [Último acceso: 20 Enero
2017].
- [37 R. E. (RES), «¿Qué servicios ofrece una Smart City a sus ciudadanos?
] (1),» 15 Julio 2013. [En línea]. Available:

<http://www.ecointeligencia.com/2013/07/que-servicios-ofrece-una-smart-city-a-sus-ciudadanos-1/>. [Último acceso: 10 Diciembre 2016].

- [38 Endesa, «Smart Cities,» [En línea]. Available:] http://www.endesaeduca.com/Endesa_educar/recursos-interactivos/smart-city/. [Último acceso: 07 Enero 2017].
- [39 Jaime_RGP, «Descubre qué es Django, el framework web de moda,»] ComputerHoy.com, 25 Enero 2014. [En línea]. Available: <http://computerhoy.com/noticias/internet/descubre-que-es-django-framework-web-moda-8641>. [Último acceso: 20 Diciembre 2016].
- [40 Logitek, «Smart City,» 2013. [En línea]. Available:] <http://www.creatingSMARTcities.es/>. [Último acceso: 21 Enero 2017].
- [41 J. P. Porto y A. Gardey, «DEFINICIÓN DE JAVA,» 2013. [En línea].] Available: <http://definicion.de/java/>. [Último acceso: 19 Diciembre 2016].
- [42 A. Studio, «Android Studio,» [En línea]. Available:] <https://developer.android.com/studio/intro/index.html?hl=es-419>. [Último acceso: 23 Enero 2017].
- [43 Techopedia, «Smart Device,» [En línea]. Available:] <https://www.techopedia.com/definition/31463/smart-device>. [Último acceso: 03 Enero 2017].
- [44 «JSON,» Curso librerías Web 2.0. , Noviembre 2008. [En línea]. Available:] <https://si.ua.es/es/documentacion/mootools/documentos/pdf/json.pdf>. [Último acceso: 16 Diciembre 2016].
- [45 J. C. M. Urquiaga, «¿Por qué usar Django?,» Devcode, [En línea].] Available: <https://devcode.la/blog/por-que-usar-django/>. [Último acceso: 05 Enero 2017].

ANEXOS

ANEXO A. Detalle de la primera encuesta realizada

	MODELO DE CELULAR	1) ¿Estaría dispuesto a utilizar una aplicación móvil para el control y registro de <i>parqueadero</i> ?	2) ¿Cuenta usted con un dispositivo móvil con Sistema Operativo Android?	3) ¿Cuál es la versión de su Sistema Operativo?	4) ¿Cuenta usted con un <i>parqueadero</i> en el cual pueda estacionar un vehículo cada vez que lo requiera?	5) ¿Cuál cree usted que es el mayor inconveniente que se presenta al momento de encontrar <i>parqueadero</i> ?
1	Samsung S3	SI	SI	Android 4.0	SI	Espacios Libres
2	Samsung S5 mini	SI	SI	Android 4.0	SI	Espacios Libres
3	Samsung S5 mini	SI	SI	Android 4.0	NO	Espacios Libres
4	Samsung S3 mini	SI	SI	Android 4.1.2	SI	Seguridad
5	Samsung S3 mini	SI	SI	Android 4.1.2	NO	Espacios Libres
6	Samsung S4 mini	SI	SI	Android 4.2.2	SI	Espacios Libres
7	Samsung S4 mini	SI	SI	Android 4.2.2	NO	Falta de Ubicación
8	Samsung S4	NO	SI	Android 4.4	NO	Espacios Libres
9	Sony Ericsson M2	SI	SI	Android 4.4	SI	Espacios Libres
10	Samsung S5	SI	SI	Android 4.4.2	SI	Espacios Libres
11	Samsung S4	SI	SI	Android 4.4.4	NO	Espacios Libres
12	Samsung S5	SI	SI	Android 5.0	SI	Falta de Ubicación
13	Samsung A3	SI	SI	Android 5.0.2	SI	Espacios Libres
14	Samsung A3	SI	SI	Android 5.0.2	NO	Espacios Libres
15	Samsung J2	SI	SI	Android 5.1.1	NO	Espacios Libres
16	Samsung J3	SI	SI	Android 5.1.1	SI	Falta de Ubicación
17	Samsung Note 3	SI	SI	Android 5.5.1	NO	Espacios Libres
18	HTC M8	SI	SI	Android 6.0	SI	Espacios Libres

19	Samsung S6	SI	SI	Android 6.0	SI	Espacios Libres
20	Samsung J7	SI	SI	Android 6.0.1	NO	Espacios Libres
21	Samsung S7 Edge	NO	SI	Android 6.0.1	SI	Seguridad
22	Sony Xperia Z3	SI	SI	Android 6.0.1	SI	Espacios Libres
23	Sony Xperia Z5 Compact	SI	SI	Android 6.0.1	SI	Espacios Libres
24	Samsung J7	NO	SI	Android 6.0.1	NO	Espacios Libres
25	Samsung Note 5	SI	SI	Android 6.0.1	SI	Espacios Libres
26	Iphone 6	SI	NO	IOS 10.1.1	SI	Espacios Libres
27	Iphone 6	SI	NO	IOS 10.1.1	NO	Espacios Libres
28	Iphone 6 S	SI	NO	IOS 9.3.2	SI	Espacios Libres
29	Iphone 4 S	SI	NO	IOS 9.3.5	NO	Espacios Libres
30	Nokia Xpress Music	SI	NO		SI	Espacios Libres

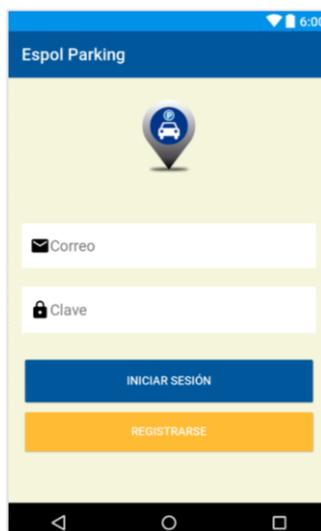
ANEXO B. Detalle de la segunda encuesta realizada.

	<i>1) ¿Es usted usuario frecuente del parqueo de FIEC?</i>	<i>2) ¿Cuál es su frecuencia de ingreso al parqueo en la semana?</i>	<i>3) ¿Cuál es marca de su dispositivo móvil?</i>	<i>4) ¿Cuál es el modelo de su dispositivo móvil?</i>
1	SI	5	Sony	Z3
2	SI	4	Samsung	J7
3	SI	5	Iphone	6
4	SI	3	Samsung	S5
5	SI	2	Lg	G2
6	SI	5	Samsung	Note3
7	SI	Más de 5	Samsung	S3 Mini
8	SI	3	Iphone	6
9	SI	5	Samsung	S5
10	SI	2	Iphone	5
11	SI	5	Iphone	6
12	SI	5	Sony	Xperia M5
13	SI	5	Lg	Nexus5
14	SI	5	Samsung	S4 Mini
15	SI	2	Samsung	Galaxy Note 4
16	SI	1	Samsung	A5
17	NO	0	Samsung	Note 4
18	SI	Más de 5	Iphone	5S
19	SI	5	Sony	Z1
20	SI	Más de 5	Samsung	Note 4
21	SI	5	Samsung	S7 Edge
22	SI	5	Samsung	J5
23	SI	5	Samsung	J7
24	SI	5	Samsung	A3
25	SI	Más de 5	Huawei	P8
26	SI	3	Iphone	6
27	SI	3	Microsoft	L950
28	SI	Más de 5	Samsung	S5
29	NO	1	Samsung	S7
30	SI	5	Motorola	Moto X
31	SI	4	Iphone	7

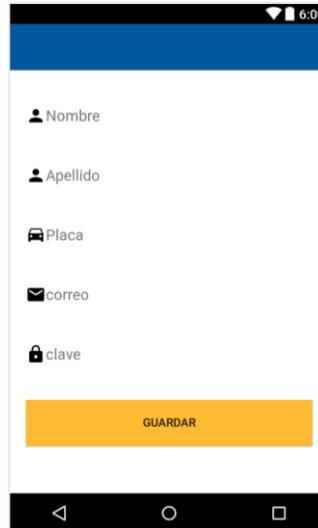
ANEXO C. Manual de Usuario**MANUAL DE USUARIO****ESPOL PARKING****REGISTRO NUEVO USUARIO**

Lo primero que debe hacer es registrar una cuenta de usuario por medio de su correo electrónico.

1. En la primera pantalla al inicia la aplicación, dar clic en “Registrarse”



2. Completar los campos de datos personales.

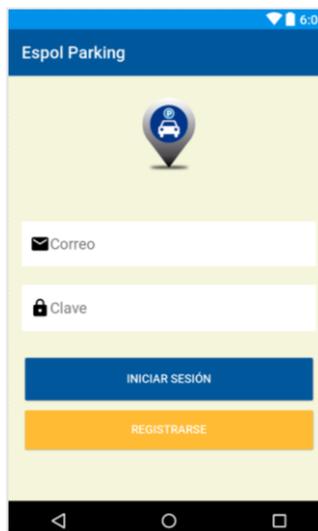


A screenshot of a mobile application registration form. The form is displayed on a white background with a blue header bar at the top. The form contains five input fields, each with a corresponding icon: a person icon for 'Nombre', another person icon for 'Apellido', a license plate icon for 'Placa', an envelope icon for 'correo', and a lock icon for 'clave'. Below the input fields is a prominent yellow button labeled 'GUARDAR'. The status bar at the top right shows the time as 6:00. The bottom navigation bar is visible at the bottom of the screen.

3. Dar clic en guardar
4. A continuación, ingresará a la interfaz de usuario con un mensaje de "Registro exitoso de usuario"

INICIO DE SESION

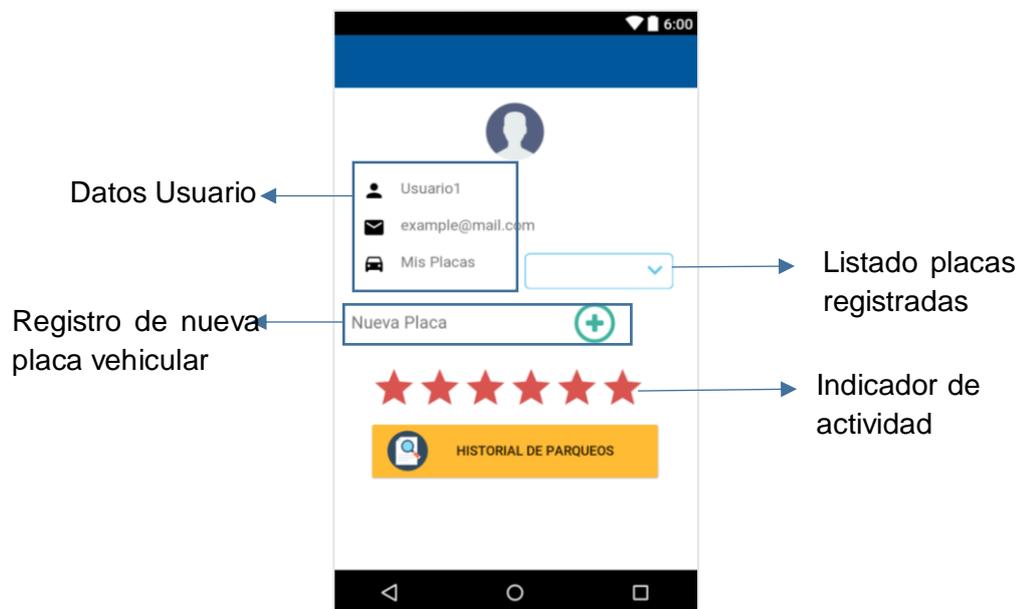
1. Ingresar el correo electrónico y la clave con el que se registró en la aplicación.
2. Dar clic en "Iniciar Sesión"



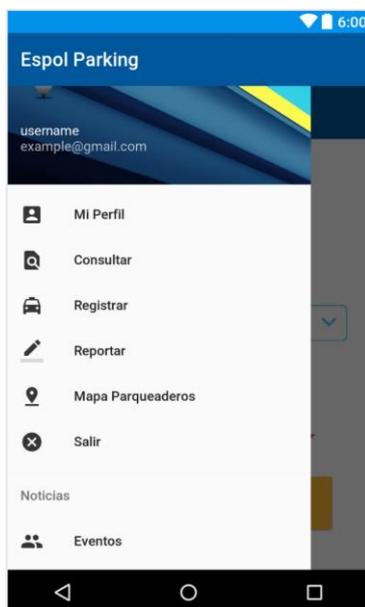
A screenshot of the login screen of the 'Espol Parking' application. The screen has a blue header bar with the text 'Espol Parking'. Below the header is a yellow background with a blue location pin icon containing a car and a person. There are two input fields: 'Correo' with an envelope icon and 'Clave' with a lock icon. Below the input fields are two buttons: a blue button labeled 'INICIAR SESIÓN' and a yellow button labeled 'REGISTRARSE'. The status bar at the top right shows the time as 6:00. The bottom navigation bar is visible at the bottom of the screen.

INTERFAZ DE USUARIO

Interfaz principal

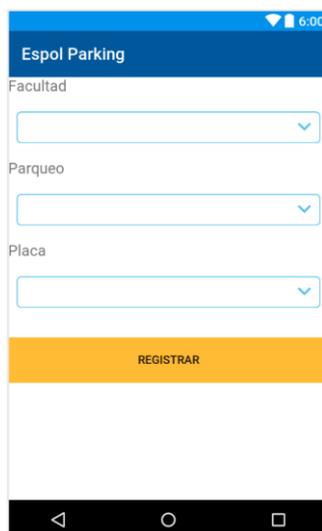


Interfaz de Menú



REGISTRAR PARQUEO

1. Acceder al menú, seleccionar “Registrar”
2. Escoger la Facultad e indicar el parqueo y la placa del vehículo con el que va hacer uso del estacionamiento.



3. Dar clic en “Registrar”
4. Regresará a la interfaz de usuario y vera un mensaje indicando que su registro ha sido exitoso, en caso de estar dentro del perímetro del parqueadero.

SALIR PARQUEO

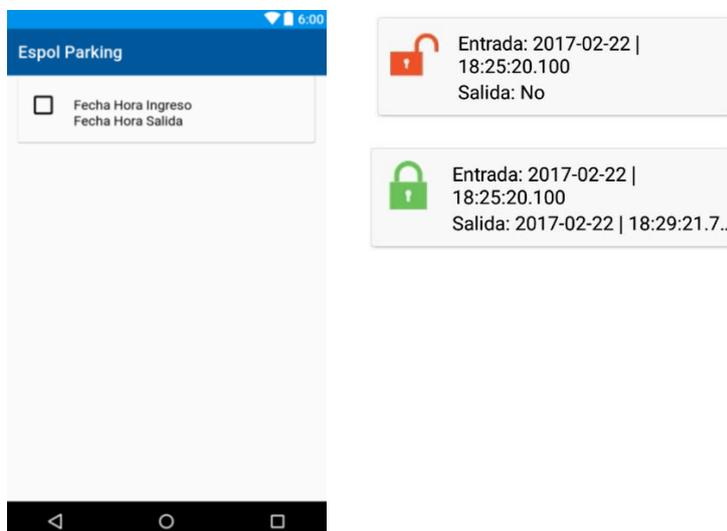
1. Acceder al menú, seleccionar “Salir”.
2. La aplicación procede a finalizar la ejecución de la aplicación y registra la salida del parqueadero.

HISTORIAL DE PARQUEOS

Se cuenta con un historial el que permite verificar los registros realizados en el parqueadero. El cual nos brinda información tal como la fecha y hora de entrada y salida del estacionamiento.

1. Dentro de la interfaz de usuario, dar clic en el boton de “Historial de Parques”

2. Se abre una interfaz en la que permite corroborar el estado, si se encuentra aún dentro del parqueadero o si ya ha registrado su salida.



REPORTAR

Para poder realizar colaboraciones dentro de la plataforma es importante que el usuario se encuentre dentro del estacionamiento o sus alrededores. Esto debe realizarse dentro del horario establecido el cual es 7H00 – 18H00

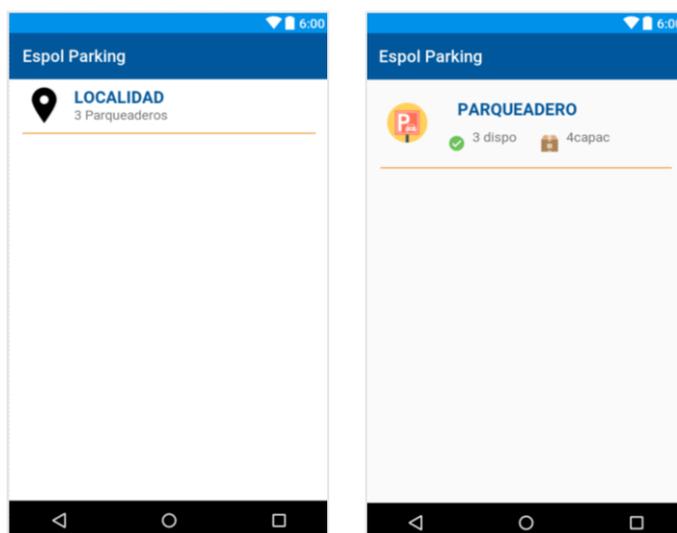
1. Acceder al menú, seleccionar "Reportar"
2. La siguiente interfaz, deberá seleccionar la facultad y el parqueadero del cual se quiere reportar la disponibilidad que evidencia.
3. Indicar la cantidad de parqueos disponibles que visualiza, y proceder a dar clic en "Guardar"



CONSULTAR

Esta ventana estará disponible cada vez que el usuario haya alcanzado un aporte máximo o se haya registrado en un parqueadero. Podrá consultar al día siguiente después de verificar su actividad.

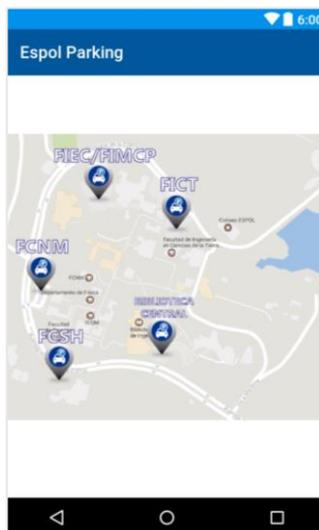
1. Acceder al menú, seleccionar "Consultar"
2. Seleccionar la localidad a consultar.
3. Se visualiza la cantidad de parqueos disponibles y el total de su capacidad por cada estacionamiento declarado.



MAPA

La aplicación cuenta con un mapa indicativo de los parqueaderos asignados dentro del campus Gustavo Galindo, ESPOL.

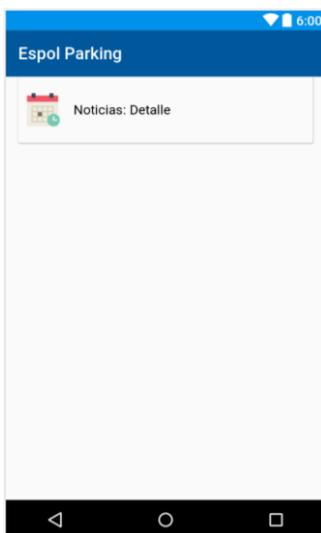
1. Ingresar al menú, escoger la opción “Mapa Parqueaderos”
2. Se puede visualizar el mapa con los sitios de referencia.



EVENTOS

Dentro del campus Gustavo Galindo, ESPOL en la facultad FIEC o FIMCP, se realizan varias actividades en las cuales se notificaran al momento de planificarse de esta manera el usuario conocerá con anticipación si se permitirá el ingreso a dicho parqueadero o estará mucho más congestionado de lo normal y buscar uno alternativo.

1. Ingresar al menú, en la parte final del mismo existe la sección “Noticias” escoger la opción “Eventos”
2. Se visualiza el conjunto de eventos planificados indicando fecha y hora a desarrollarse y la facultad.



ANEXO D. Manual de Instalación

MANUAL DE INSTALACIÓN



ESPOL PARKING

MANUAL DE INSTALACIÓN

El presente manual se determina los pasos del proceso de instalación para la plataforma de operación de la aplicación móvil. Esta plataforma está desarrollada bajo características del framework DJANGO y código en Python

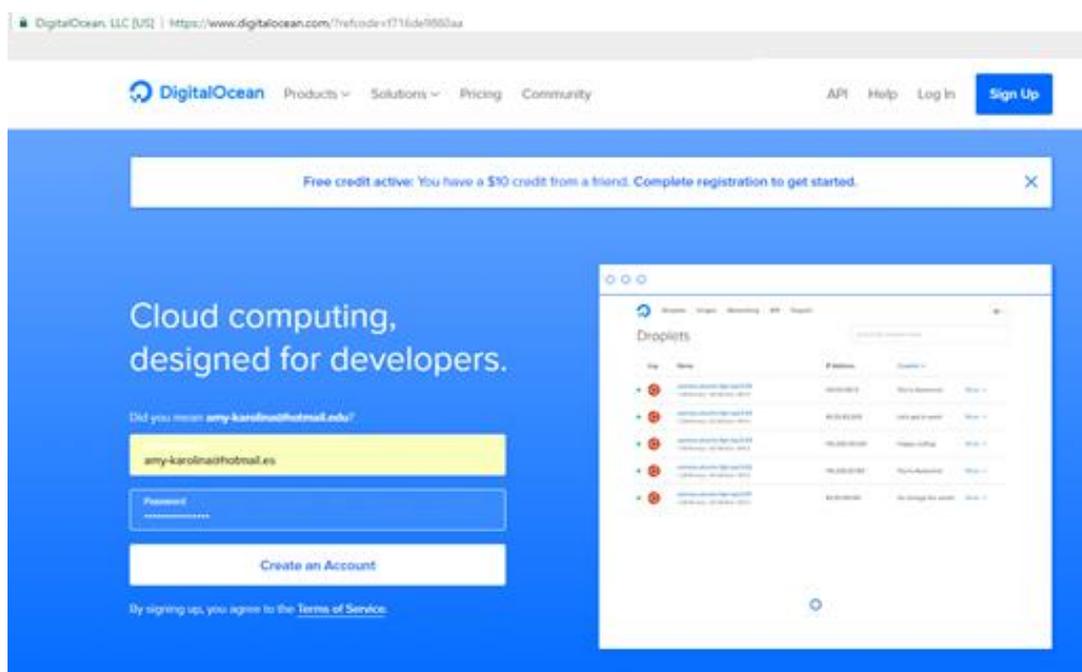
INSTALACIÓN DEL SERVIDOR

Ya que la característica principal fue desarrollar un servidor del cual se pueda acceder independiente de la red, se realizó una suscripción a un servicio web de nube. Cabe mencionar que los servicios del servidor pueden ser establecidos para un servidor local.

SOLICITUD SERVIDOR NUBE DIGITALOCEAN

1. Acceder a la web del proveedor de DigitalOcean:
<https://www.digitalocean.com/?refcode=f716de9860aa>

2. Proceder a realizar el registro del usuario por medio de un correo electrónico y una contraseña. Una vez llenado estos campos dar clic en “Crear Cuenta”
3. A continuación, se debe llenar un formulario con los datos personales del usuario.
4. Una vez finalizado este proceso, se debe esperar un correo de confirmación en la dirección de correo electrónico antes registrado.



5. Realizada la confirmación acceder a la web del proveedor para escoger uno de los planes que tiene a disposición. Para efectos del desarrollo se ha seleccionado aquel que tenía características como
 - 512 MB de RAM/1CPU
 - 20 GB de SSD
 - 1 TB de transferencia mensual

Standard High memory

\$5/mo \$0.007/hour	\$10/mo \$0.015/hour	\$20/mo \$0.030/hour	\$40/mo \$0.060/hour	\$80/mo \$0.119/hour	\$160/mo \$0.238/hour
512 MB / 1 CPU 20 GB SSD disk 1000 GB transfer	1 GB / 1 CPU 30 GB SSD disk 2 TB transfer	2 GB / 2 CPUs 40 GB SSD disk 3 TB transfer	4 GB / 2 CPUs 60 GB SSD disk 4 TB transfer	8 GB / 4 CPUs 80 GB SSD disk 5 TB transfer	16 GB / 8 CPUs 160 GB SSD disk 6 TB transfer
\$320/mo \$0.476/hour	\$480/mo \$0.714/hour	\$640/mo \$0.952/hour			
32 GB / 12 CPUs 320 GB SSD disk 7 TB transfer	48 GB / 16 CPUs 480 GB SSD disk 8 TB transfer	64 GB / 20 CPUs 640 GB SSD disk 9 TB transfer			

- Después de haber realizado la selección, se accede al panel de control en el cual se configura el sistema operativo a instalar. Para ello se eligió una distribución de Linux, Ubuntu 16.04.1

Standard High memory

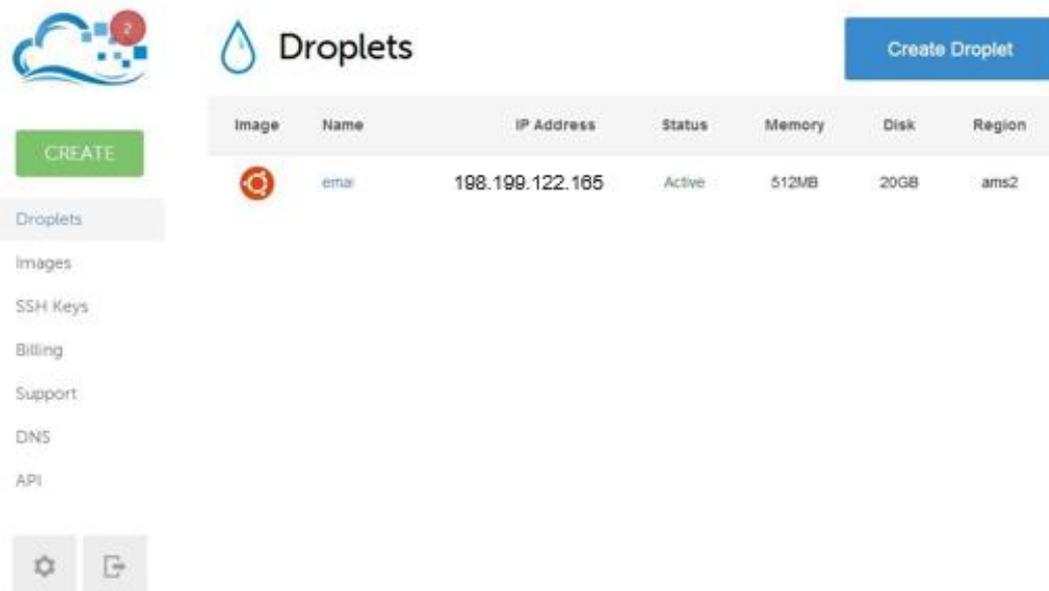
\$5/mo \$0.007/hour	\$10/mo \$0.015/hour	\$20/mo \$0.030/hour	\$40/mo \$0.060/hour	\$80/mo \$0.119/hour	\$160/mo \$0.238/hour
512 MB / 1 CPU 20 GB SSD disk 1000 GB transfer	1 GB / 1 CPU 30 GB SSD disk 2 TB transfer	2 GB / 2 CPUs 40 GB SSD disk 3 TB transfer	4 GB / 2 CPUs 60 GB SSD disk 4 TB transfer	8 GB / 4 CPUs 80 GB SSD disk 5 TB transfer	16 GB / 8 CPUs 160 GB SSD disk 6 TB transfer
\$320/mo \$0.476/hour	\$480/mo \$0.714/hour	\$640/mo \$0.952/hour			
32 GB / 12 CPUs 320 GB SSD disk 7 TB transfer	48 GB / 16 CPUs 480 GB SSD disk 8 TB transfer	64 GB / 20 CPUs 640 GB SSD disk 9 TB transfer			

- Con ello el proceso de solicitud y pre configuración ha finalizado.

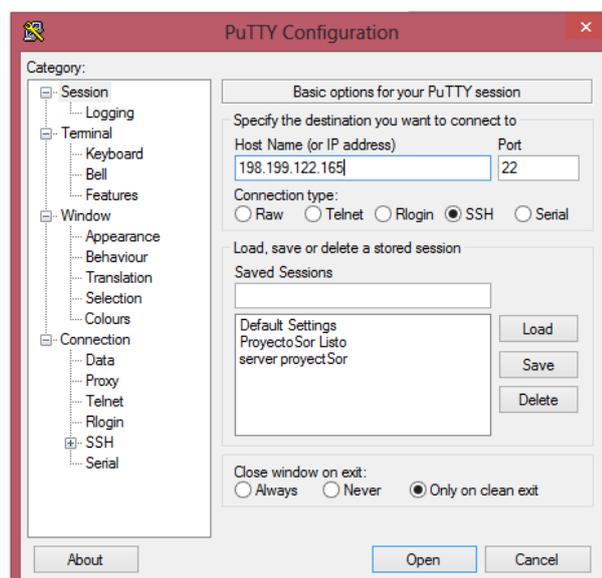
CONFIGURACIÓN DEL SERVIDOR

Para realizar la configuración del servidor, se accede vía SSH por terminal

- Obtener la dirección IP del servidor, ingresando a DROPLETS. La ip asignada al servidor es 198.199.122.165.



2. La conexión se realiza a través de PuTTY, las credenciales para acceder a la configuración, se han enviado previamente al correo electrónico.



```

root@ubuntu-512mb-nyc1-01: ~
login as: root
root@198.199.122.165's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-53-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

76 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Sun Mar  5 21:23:04 2017 from 181.198.246.14
root@ubuntu-512mb-nyc1-01:~#

```

INSTALACIÓN DJANGO EN EL SERVIDOR NUBE DIGITAL

Una de las características principales de DJANGO es la herramienta de entorno virtual, la cual permite que los cambios que se realicen no tenga un efecto colateral al resto del sistema, esto a su vez facilita que los paquetes que se instalen para un proyecto no interfieran en los requisitos de los otros proyectos que se puedan ejecutar.

1. Iniciar la conexión con el terminal e iniciar sesión
2. Realizar una actualización del sistema por medio del comando

```
sudo apt-get update
```

3. Para la instalación de Python 2:

```
sudo apt-get install python-pip
```

4. Instalar el entorno virtual

```
sudo pip install virtualenv
```

5. Una vez instalado el paquete del entorno virtual, se procede a buscar en el directorio la ruta en la cual el entorno va a ser creado y el nombre que recibirá, se ejecuta el siguiente comando

```
cd /home/
```

```
sudo virtualenv env_parkea
```

6. Se requiere instalar los paquetes de entorno aislado, para ello es necesario acceder al directorio del entorno virtual y activarlos

```
cd env_parkea
```

```
source bin/activate
```

7. Se procede a instalar la versión de DJANGO seleccionada.

```
sudo pip install django==1.8
```

8. Instalar el paquete de mysql-python

```
pip install mysql-python
```

Para salir del entorno virtual se debe escribir el comando *deactivate* desde cualquier parte del sistema

CREACIÓN DE UN PROYECTO EN DJANGO

Para iniciar el desarrollo de un proyecto en DJANGO es necesario conocer las bases de su creación.

1. Crear un nuevo proyecto usando el comando

```
django-admin startproject parkea
```

Este comando crea un nuevo directorio dentro de la ruta donde se ejecute.

```
bin include lib local parkea
```

2. Luego situarnos en el directorio del proyecto

```
cd parkea
```

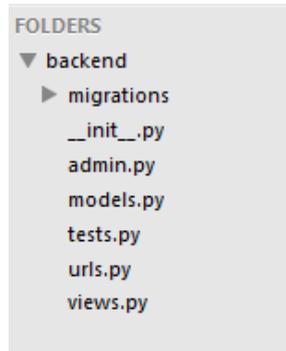
3. Para iniciar la base de datos, usar el siguiente comando. Cabe mencionar que DJANGO usa de manera predeterminada SQLite en sus versiones más recientes.

```
python manage.py migrate
```

4. Para crear la aplicación se debe escribir el siguiente comando

python manage.py startapp backend

Este comando crea un directorio que contiene 4 archivos estructurados de la siguiente manera



5. Se procede a crear un superusuario, el cual tendrá acceso al servidor en DJANGO, para ello se solicitará un nombre de usuario, correo electrónico y contraseña

python manage.py createsuperuser

6. Una vez que el usuario ha sido creado, puede iniciar el servidor de desarrollo de DJANGO

python manage.py runserver 198.198.122.165:8000

Si está trabajando con un servidor local la IP es: 127.0.0.1:8000

7. Para comprobar que el servidor se encuentra operativo, escribir en la barra de dirección de un navegador: 198.199.122.165:8000. Obteniendo el siguiente contenido.

It worked!
Congratulations on your first Django-powered page.

Of course, you haven't actually done any work yet. Here's what to do next:

- If you plan to use a database, edit the `DATABASES` setting in `myproject/settings.py`.
- Start your first app by running `python myproject/manage.py startapp [appname]`.

You're seeing this message because you have `DEBUG = True` in your Django settings file and you haven't configured any URLs. Get to work!

8. Para acceder a la página de administración del servidor vía web, digitar en la barra de dirección del navegador `198.199.122.165:8000/admin`

198.199.122.165:8000/admin/login/?next=/admin/

Administración de Django

Nombre de usuario:
AmiraTomala

Contraseña:
*****|

Iniciar sesión

9. Iniciar sesión con las credenciales creadas como superusuario. En caso de ser válidas, será llevado a la página de administración.

The screenshot shows the Django administration interface. At the top, it says "Django administration" and "WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT". Below this, there's a "Site administration" section. Underneath, there's a sub-section for "AUTHENTICATION AND AUTHORIZATION" with two rows: "Groups" and "Users". Each row has a "+ Add" button and a "Change" button with a pencil icon. To the right, there's a "Recent Actions" box with "My Actions" and "None available".

ACTUALIZAR LOS ARCHIVOS DE UN PROYECTO

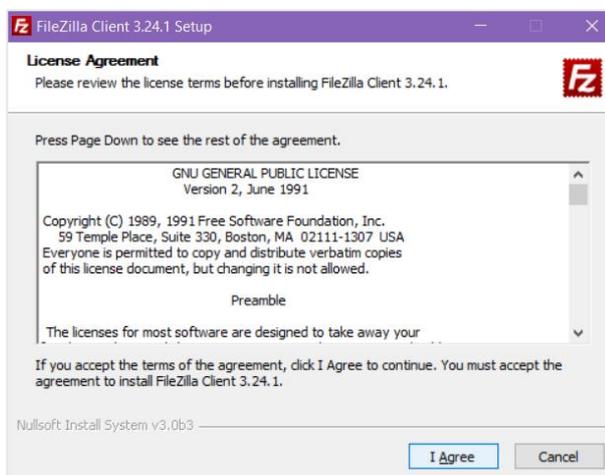
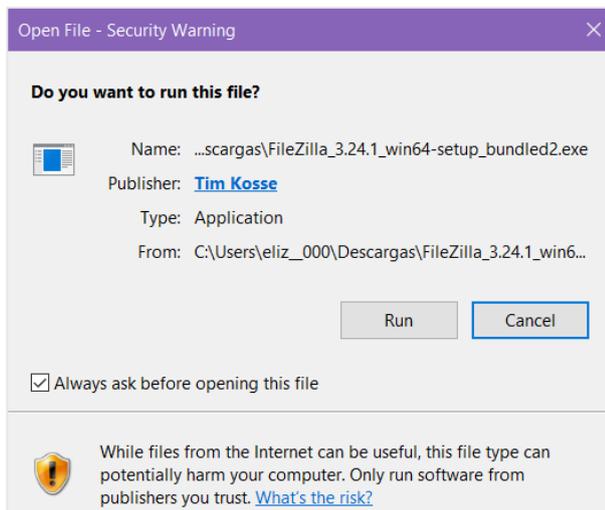
Una vez que cuenta con un directorio al cual asignarle los respectivos archivos que contienen la configuración y funciones del proyecto puede proceder de la siguiente manera.

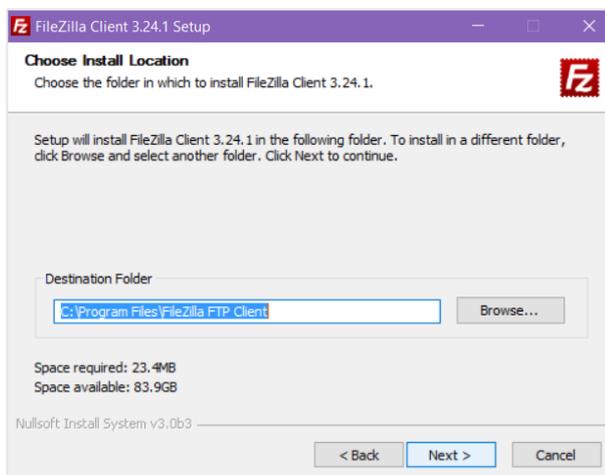
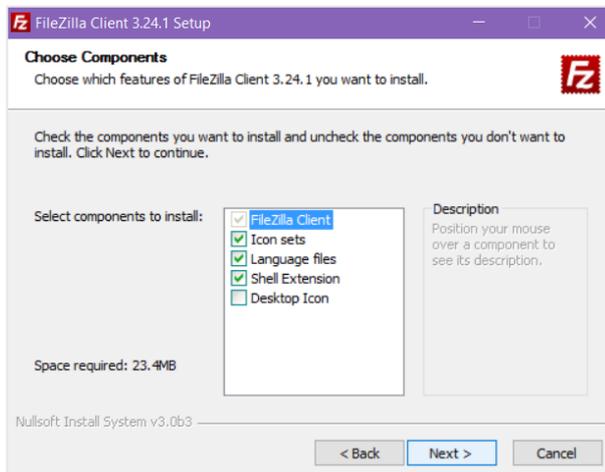
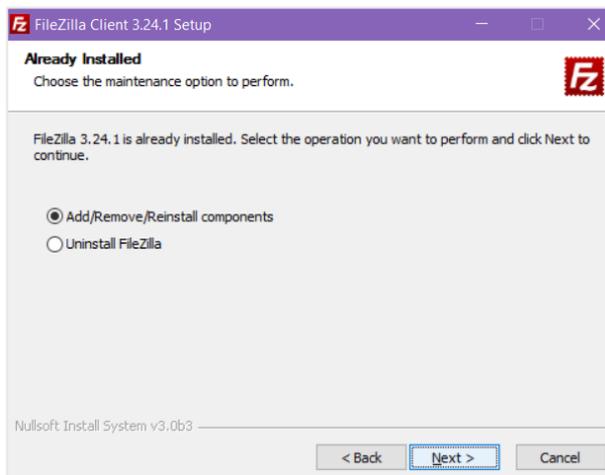
1. Descargar Filezilla, un gestor cliente para la transferencia de archivos al servidor con una interfaz gráfica simple.

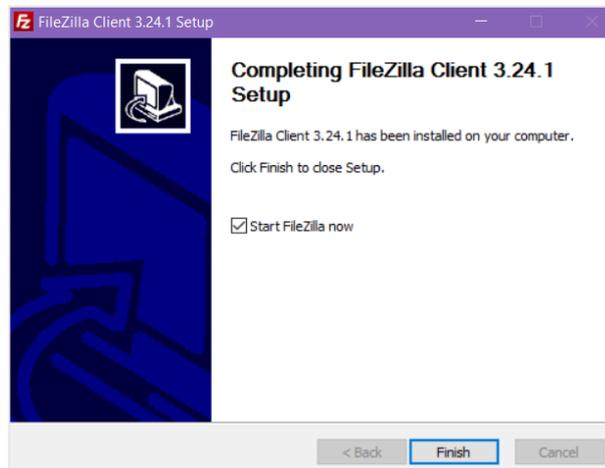
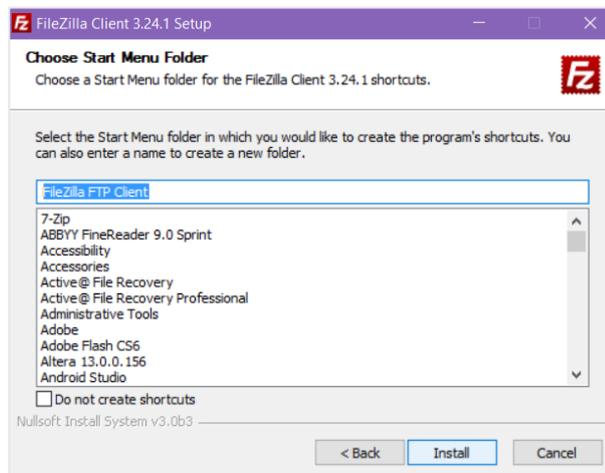
<https://filezilla-project.org/>

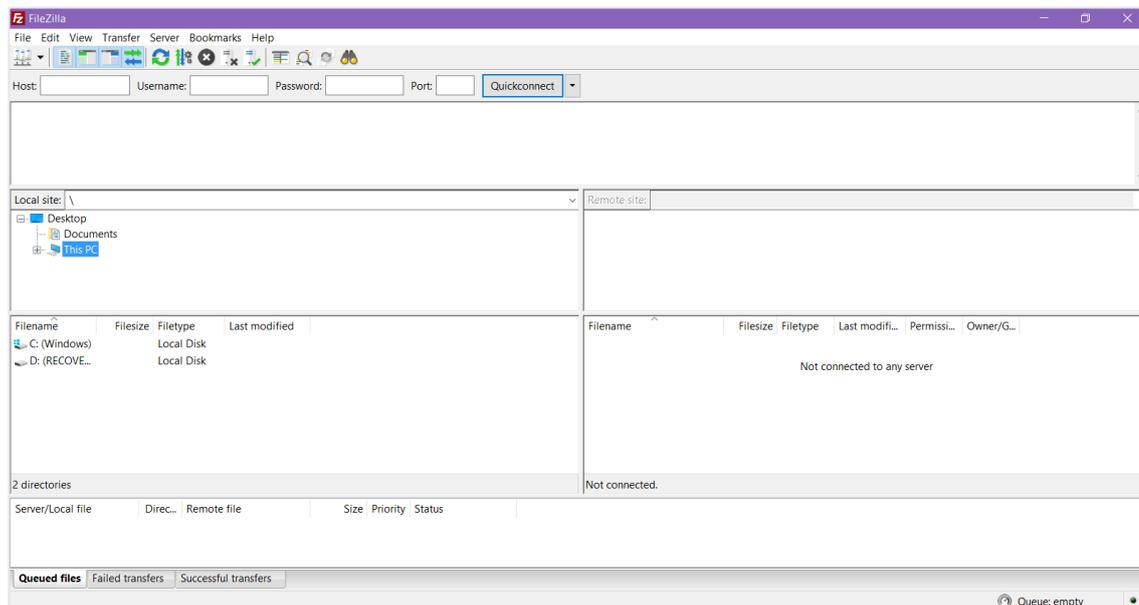
The screenshot shows the FileZilla website homepage. The browser address bar shows "https://filezilla-project.org". The page features the FileZilla logo and the tagline "The free FTP solution". A navigation menu on the left includes links for Home, FileZilla (Features, Screenshots, Download, Documentation), FileZilla Server (Download), Community (Forum, Project page, Wiki), General (FAQ, Contact, License, Privacy Policy), and Development (Source code, Nightly builds, Translations, Version history, Changelog, Issue tracker). The main content area is titled "Overview" and contains a welcome message, support information, and "Quick download links" for the FileZilla Client (All platforms) and FileZilla Server (Windows only). Below this, there's a "News" section with two entries: "2017-03-04 - FileZilla Client 3.25.0-rc1 released" and "2017-02-24 - FileZilla Client 3.25.0-beta1 released". A "Bugfixes and minor changes" section lists fixes for encryption selection and error message texts.

2. Una vez que el programa se haya descargado, se procede a iniciar el proceso de instalación.

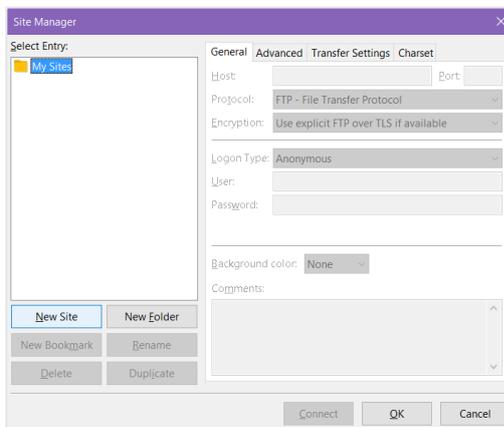




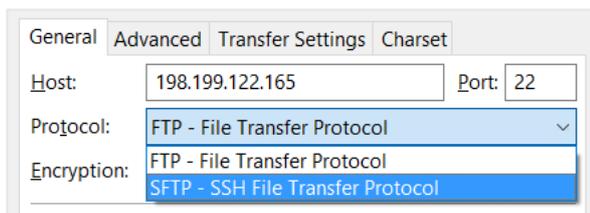




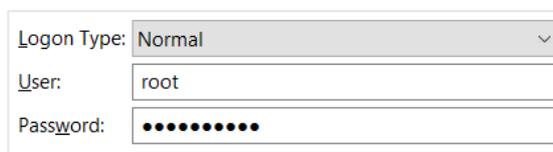
3. Dar clic en “Archivo” y seleccionar “Administracion de Sitio” para configurar el sitio el cual se va a manejar. En la ventana que se presenta dar clic en “Sitio Nuevo”



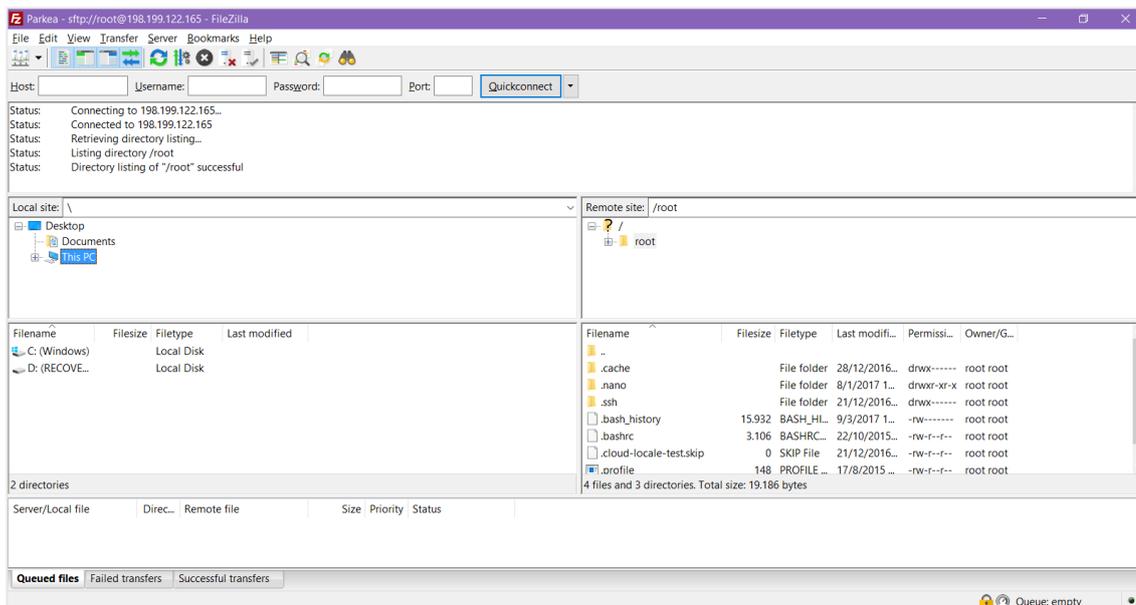
4. Establecer la dirección IP del servidor con el puerto de acceso, adicionalmente seleccionar el protocolo SFTP, el cual trabaja con el protocolo SSH.



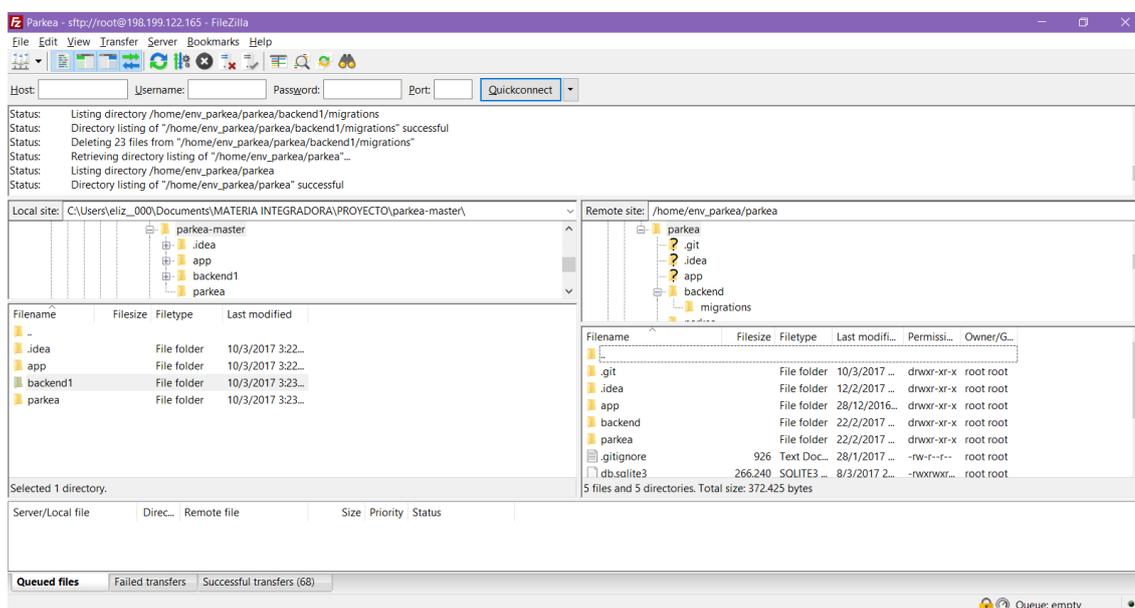
5. Se procede a seleccionar el tipo de inicio de sesión a “Normal” e ingresar las credenciales.



6. Dar clic en “Conectar”. El cual inicia el proceso de conectividad con el servidor. Se puede visualizar el proceso en la división de estatus. Así mismo, una vez finalizado el proceso se puede verificar los archivos contenidos en el servidor.



7. Dentro del directorio del sitio local, se debe buscar la ruta donde se encuentran los archivos que pertenecen a la aplicación que se deben establecer en el servidor.
8. Estos archivos deben ser seleccionados y enviados a la ruta donde se encuentra creado el proyecto dentro del entorno virtual. Para ello es necesario seleccionarlos y arrastrarlos al lado del sitio remoto situado al derecho de la ventana.



9. Se accede a la administración web se puede constatar que los archivos se encuentran operativos dentro de la plataforma



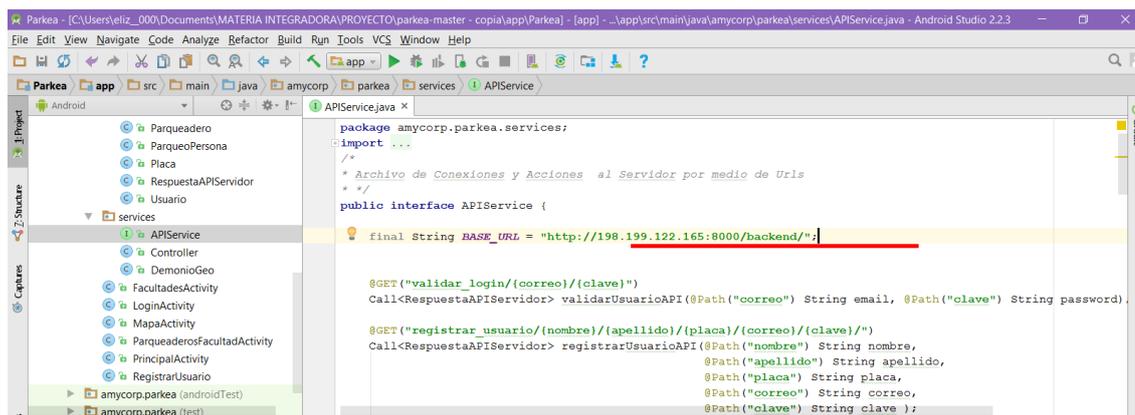
En esta página podrá administrar cada uno de los componentes declarados perteneciente a la base de datos, así como sus registros

INSTALACIÓN APLICACIÓN MÓVIL

Antes de instalar la aplicación en el dispositivo móvil, se requiere hacer un cambio en caso de ser necesario.

CAMBIAR IP SOURCE

Dentro del programa, Android Studio, se debe acceder al archivo ApiService que se encuentra en la carpeta “services” dentro del conjunto del proyecto.



```
package amycorp.parkea.services;
import ...
/*
 * Archivo de Conexiones y Acciones al Servidor por medio de Urls
 */
public interface ApiService {
    final String BASE_URL = "http://198.122.165:8000/backend/";

    @GET("validar_login/{correo}/{clave}")
    Call<RespuestaAPIServidor> validarUsuarioAPI(@Path("correo") String email, @Path("clave") String password);

    @GET("registrar_usuario/{nombre}/{apellido}/{placa}/{correo}/{clave}")
    Call<RespuestaAPIServidor> registrarUsuarioAPI(@Path("nombre") String nombre,
                                                    @Path("apellido") String apellido,
                                                    @Path("placa") String placa,
                                                    @Path("correo") String correo,
                                                    @Path("clave") String clave );
}
```

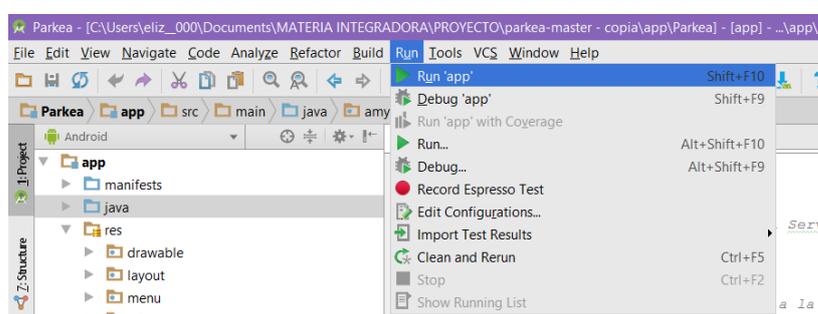
INSTALACIÓN DE LA APLICACIÓN

Existen dos formas que se pueden instalar la aplicación dentro del dispositivo móvil la primera forma es cargarla en tiempo real y la segunda es a través de un archivo .apk

Primera Forma.

En esta forma se necesita tener un teléfono con “Opciones de Desarrollador”, aquello se activa una sola vez accediendo Ajustes > Información del dispositivo y se debe pulsar 7 veces sobre el número de compilación

1. Ir a “Ejecutar” y seleccionar “Ejecutar app”



2. Se inicializa el ADB el cual nos permite seleccionar el dispositivo en el cual se quiere ejecutar la app.
3. Seleccionar el dispositivo y dar clic en “Ok”
4. Esperar que la aplicación haya completado la transferencia para que la instalación sea terminada.

Segunda Forma.

Esta forma es mucho más accesible para la instalación de la aplicación desde aquellos dispositivos que no cuentan con Android Studio, e inclusive desde dispositivos móviles ya que inicialmente se crea un archivo el cual se puede ejecutar en el dispositivo final.

1. Ir a “Crear” y seleccionar “Crear APK”
2. Se procede a crearse un archivo dentro del directorio del proyecto