

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
INGENIERO DE SISTEMAS**

**TEMA:
DESARROLLO DE UNA APLICACIÓN MÓVIL QUE ENVÍE UNA
SOLICITUD DE REQUERIMIENTO DE GAS DOMÉSTICO A LOS
DISTRIBUIDORES O A LOS VEHÍCULOS DE DISTRIBUCIÓN MÁS
CERCANOS A UNA UBICACIÓN EN LA CIUDAD DE QUITO MEDIANTE
GEOREFERENCIACIÓN**

**AUTOR:
CHRISTIAN JAVIER ACURIO ACURIO**

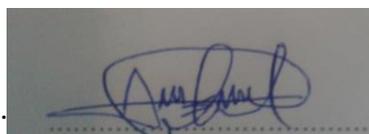
**TUTOR:
FERNANDO JACINTO RODAS ORELLANA**

Quito, Agosto 2017

CESIÓN DE DERECHOS DE AUTOR

Yo, Christian Javier Acurio Acurio, con documento de identificación N° 1714712666, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación intitulado: Desarrollo de una aplicación móvil que envíe una solicitud de requerimiento de gas doméstico a los distribuidores o a los vehículos de distribución más cercanos a una ubicación en la ciudad de Quito mediante Georeferenciación, mismo que ha sido desarrollado para optar por el título de: Ingeniero de Sistemas en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



Christian Javier Acurio Acurio

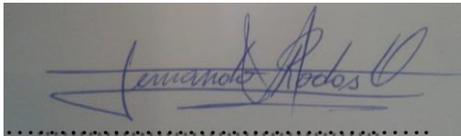
CI: 1714712666

Quito, agosto del 2017

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación:
Desarrollo de una aplicación móvil que envíe una solicitud de requerimiento de gas doméstico a los distribuidores o a los vehículos de distribución más cercanos a una ubicación en la ciudad de Quito mediante Georeferenciación, realizado por Christian Javier Acurio Acurio, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, agosto del 2017

A rectangular box containing a handwritten signature in blue ink. The signature is cursive and appears to read 'Fernando Rodas Orellana'. Below the signature, there is a horizontal line and a dotted line.

Fernando Jacinto Rodas Orellana

CI: 1708514821

Dedicatoria

Dedico este proyecto a mis padres que con su esfuerzo, trabajo y dedicación han logrado que logre a culminar una meta más que me he propuesto en el transcurso de mi vida, sabiendo corregir mis errores y guiándome por el camino del bien para que llegue a ser una persona exitosa.

A mis hermanos que a pesar de los obstáculos que se nos cruzaron en la vida siempre me enseñaron a seguir adelante y me demostraron que cuando uno se propone algo lo puede conseguir así sea lo más difícil que a uno se le pueda cruzar en la vida.

ÍNDICE

INTRODUCCIÓN	1
Antecedentes	2
Justificación.....	5
Objetivo general	5
Objetivos específicos	5
Marco metodológico	6
Capítulo 1	7
Marco Teórico.....	7
1.1 Lenguaje Unificado de Modelado	7
1.1.1 Diagrama de Casos de uso.	7
1.1.2 Diagrama de clases.....	8
1.1.3 Diagrama de navegación.	8
1.1.4 Diagrama de flujo de procesos.	8
1.2 Metodología XP	8
1.3 Aplicación Móvil o APP	10
1.4 Sistema Operativo	10
1.4.1 Android.	11
1.5 Entorno de desarrollo Android Studio	13
1.6 Georreferenciación.....	13
1.7 Google Maps	13
1.8 Api de Google Android Maps	14
1.9 Gestor De Base De Datos.....	14
1.9.1 MySQL.....	15
1.10 Web services	15
1.10.1 Windows Azure.....	16
Capítulo 2	17
Análisis y Diseño	17
3.1 Análisis de requerimientos	17
3.1.1 Identificación de usuarios.	17
3.1.2 Historia de usuarios.....	18
3.1.3 Requerimiento usuario Distribuidor de gas doméstico.	18
3.1.4 Requerimiento cliente.	20
3.2 Diseño del aplicativo.....	22
3.2.1 Diagrama de casos de uso.	22
3.2.2 Diagrama físico de la base de datos MySQL.	24
3.2.3 Diccionario de datos.....	25
3.2.4 Diagrama de clases.....	26
3.2.5 Diagrama de navegación.	31
3.2.6 Diagrama de flujo.....	32
3.2.7 Diagrama de distribución.	34
3.2.8 Prototipo de la interfaz gráfica.	34
Capítulo 3	43
Construcción y pruebas	43
4.1 Tecnologías y arquitectura de la aplicación móvil.....	43
4.1.1 Arquitectura de la aplicación móvil.	43
4.1.2 Entorno de desarrollo Android Studio.	46

4.1.3 Windows Azure.....	46
4.2 Codificación	46
4.2.1 Clases y métodos de la aplicación.....	47
4.3 Pruebas de la aplicación móvil.....	55
4.3.1 Pruebas funcionamiento aplicación móvil.	55
4.3.2 Pruebas de rendimiento aplicación móvil.	63
4.3.3 Pruebas de Estrés Aplicación móvil.....	65
CONCLUSIONES	69
RECOMENDACIONES	71
REFERENCIAS.....	72

ÍNDICE TABLAS

Tabla 1. Historia de usuario registro de distribuidores de gas domestico.....	19
Tabla 2. Historia de usuario control de acceso distribuidor.....	19
Tabla 3. Historia de usuario compartir su ubicación	20
Tabla 4. Historia de usuario cambio de contraseña distribuidor.....	20
Tabla 5. Historia de usuario buscar distribuidores cercanos a su ubicación actual ...	21
Tabla 6. Historia de usuario visualizar información de distribuidores	21
Tabla 7. Historia de usuario realizar pedidos al distribuidor más cercano	22
Tabla 8. Descripción de las tablas de la base de datos.....	25
Tabla 9. Diccionario de la tabla datos_distribuidor	26
Tabla 10. Diccionario de datos tabla coordenadas_distri.....	26
Tabla 11. Interfaz activity_registro_distribuidor	48
Tabla 12. Insertar de datos distribuidor	48
Tabla 13. Ingreso de distribuidores a la base de datos	49
Tabla 14. Activity_login.xml	49
Tabla 15. Inicio sesión usuario distribuidor.....	49
Tabla 16. Consulta de usuario distribuidor para inicio de sesión	50
Tabla 17. Login_Secun.xml	50
Tabla 18. Consulta para inicio sesión distribuidor.....	51
Tabla 19. Actualizar coordenadas	52
Tabla 20. activity_actualizar_passw.xml	52
Tabla 21. Cambiar contraseña distribuidores.....	52
Tabla 22. Actualizar contraseña distribuidor	53
Tabla 23. activity_maps.xml.....	54
Tabla 24. Buscar distribuidores en el mapa	54
Tabla 25. Consultar coordenadas	55

ÍNDICE DE FIGURAS

Figura 1. Sistema de capas de Android.....	12
Figura 2. Esquema de un web service.....	16
Figura 3. Caso de uso distribuidor.....	23
Figura 4. Caso de uso para el cliente de la aplicación móvil.....	24
Figura 5. Diagrama físico de la base de datos Mysql.....	25
Figura 6. Diagrama de clases correspondiente a la aplicación móvil.....	30
Figura 7. Diagrama navegación distribuidor.....	31
Figura 8. Diagrama de navegación Cliente.....	32
Figura 9. Diagrama de flujo de la aplicación móvil.....	33
Figura 10. Diagrama de despliegue para la aplicación móvil.....	34
Figura 11. Interfaz pantalla principal rol distribuidor.....	35
Figura 12. Interfaz para el registro de un nuevo distribuidor.....	36
Figura 13. Interfaz para ingresar al sistema como distribuidor.....	37
Figura 14. Interfaz para actualizar contraseña distribuidor.....	38
Figura 15. Interfaz pantalla principal rol cliente.....	39
Figura 16. Interfaz para visualizar a los distribuidores en google maps.....	40
Figura 17. Interfaz para visualizar la información del distribuidor.....	41
Figura 18. Interfaz para realizar pedidos.....	42
Figura 19. Arquitectura de la aplicación móvil.....	44
Figura 20. Ejemplo de interfaz grafica.....	44
Figura 21. Arquitectura de túnel PHP y JSON.....	45
Figura 22. Prueba para ingresar de un nuevo distribuidor.....	57
Figura 23. Prueba para iniciar sesión distribuidor.....	58
Figura 24. Prueba para cambiar la contraseña de un usuario distribuidor.....	59
Figura 25. Prueba para buscar distribuidores de gas doméstico.....	60
Figura 26. Prueba para visualizar la información del distribuidor seleccionado.....	61
Figura 27. Prueba para enviar un pedido a un distribuidor.....	62
Figura 28. Prueba de rendimiento para el inicio sesión distribuidor.....	63
Figura 29. Prueba de rendimiento para el modulo envío recurrente de coordenadas.....	64
Figura 30. Prueba de rendimiento para la consulta recurrente de coordenadas.....	65
Figura 31. Prueba de estrés para el inicio sesión del distribuidor.....	66
Figura 32. Prueba de estrés para él envío recurrente de coordenadas.....	67
Figura 33. Prueba de estrés para la consulta recurrente de coordenadas.....	68

Resumen

En la actualidad los dispositivos móviles han ido evolucionando al punto que permiten integrar dentro de su sistema operativo aplicaciones que funcionan en conjunto con otras tecnologías como web services, base de datos remotas, máquinas virtuales, formando aplicaciones móviles con recursos ilimitados. La presente aplicación móvil, se centra en ubicar dentro de la ciudad de Quito a los diferentes distribuidores de gas doméstico, ya sean estos fijos o móviles, con el objetivo de que si un cliente se encuentre en la necesidad de conseguir el producto mencionado pueda obtenerlo de manera rápida y sin complicaciones.

El cliente al hacer uso de la aplicación obtiene información de los distribuidores y vehículos con el producto más cercanos y puede solicitar el mismo por medio de un pedido mediante un mensaje de texto, logrando agilizar y automatizar el proceso que conlleva conseguir este tipo de producto en la actualidad.

La información de los distribuidores es almacenada dentro de una base de datos en la nube, lo que permitirá a los usuarios obtener información exacta de cada uno de ellos.

Los distribuidores móviles tienen un módulo en el que pueden registrarse, iniciar sesión y compartir cada determinado tiempo sus coordenadas automáticamente, permitiendo hacer uso de estas para reflejar su posición dentro del mapa que será visualizado por los clientes.

La aplicación se desarrolló haciendo uso de la plataforma Android Studio y la metodología XP, para la base de datos remota se usó MySQL junto con Windows Azure.

Abstract

Currently, mobile devices have been developing to the point of integrating apps within the operating system (OS), which run jointly with other technologies, such as web services, remote databases, virtual machines that become part of mobile apps with unlimited resources. Existing mobile app, to be located in Quito City, is focused to diverse distributors of domestic gas, even they are regular or stay driving, so as to the customer in need of such product, get it quickly and without complications.

When the customer uses the app, he/she can get information of near gas distributors and vehicles, and request the product by a text message, so the process to get such product is currently effective and automated.

Distributors' information is kept within a cloud database, which shall allow users to get their precise information.

Driving distributors have a module to sign in, log in and automatically share their coordinates every certain time, with the purpose of knowing their location within the map, so it is viewed by the customers.

The app was developed by using Android Studio platform and XP methodology. MySQL together with Windows azure were used for the remote database.

INTRODUCCIÓN

En la actualidad la mayoría de personas disponen de un limitado tiempo para ocuparse de las diferentes actividades del hogar, una de estas actividades es la adquisición de un nuevo cilindro de gas doméstico para su consumo personal, muchas veces esa actividad se torna un poco tedioso por la falta de información para ubicar un punto de distribución cercano que tenga en su inventario cilindros de gas doméstico para su venta, es por esto que junto con la tecnología se desarrolló una aplicación móvil que permite obtener información del distribuidor junto con su ubicación y también se puede realizar un pedido de dicho producto por medio de la aplicación a los diferentes puntos de distribución de gas doméstico tanto local como móvil logrando solucionar de manera rápida la necesidad del consumidor. Cabe mencionar que los diferentes distribuidores de gas doméstico no cuentan con los recursos económicos que les permita implementar este tipo de tecnología y de esta manera lograr compartir su ubicación para que pueda ser visualizada por sus distintos clientes que requieran de su servicio de forma rápida y efectiva. Es ahí donde intervienen las tecnologías que nos permiten desarrollar software libre, las cuales ayudaran a resolver el problema percibido dentro de la ciudad de Quito. La aplicación móvil de georreferenciación, se desarrolló utilizando metodologías que nos permiten aplicar buenas prácticas junto con la investigación de diferentes técnicas de desarrollo a nivel nacional e internacional permitiéndonos obtener un producto final de calidad.

El documento está dividido en tres capítulos, los mismos que se describen a continuación.

El capítulo uno contiene la justificación y los objetivos que se plantearon para el desarrollo de la aplicación, como también la metodología y los temas de investigación que sirvieron para el desarrollo de la aplicación.

El segundo capítulo detalla los usuarios y los requerimientos de cada uno, los diagramas que el lenguaje UML detalla y los prototipos de las interfaces que la aplicación móvil contiene en su estructura.

El tercer capítulo muestra las tecnologías que se usó para la creación de la aplicación, su estructura, codificación y las diferentes pruebas que se realizaron para validar el funcionamiento de la aplicación móvil, finalizando con las conclusiones y recomendaciones del trabajo realizado.

Antecedentes

Las aplicaciones móviles con el transcurso de los años han ido evolucionando cada vez son más utilizadas por diferentes personas, muchas de estas aplicaciones móviles son desarrolladas para satisfacer necesidades del diario vivir de los distintos usuarios que las utilizan a nivel mundial. Las aplicaciones encontradas en la actualidad son muy diversas, las cuales constan de innumerables servicios y características, como la localización de negocios (Georreferenciación), juegos, chats, etc.

Las aplicaciones móviles que adoptan la georreferenciación (ámbito comercial o el desarrollo de negocios) dentro de su arquitectura, son desarrolladas para automatizar la localización de diferentes objetos por ejemplo vehículos, personas, locales comerciales que distribuyen productos de primera necesidad, incluso muchas aplicaciones permiten realizar compras o pedidos en base a un punto de referencia, evitando que los usuarios dependan de elementos primitivos como los mapas de papel,

periódicos, guías telefónicas, etc. las mismas que muchas veces tienen información inexacta y generan pérdidas de tiempo.

Muchos desarrolladores de software libre en el Ecuador tienen como objetivo crear APPS que cubran dichas necesidades y que estén disponibles sin costo dentro de una plataforma donde los usuarios puedan obtenerlas y puedan hacer uso de estas. Existen entidades públicas y privadas que crean aplicaciones móviles para cubrir necesidades de gestión internas de la empresa.

Una tesis que sirve de ejemplo es la de Francisco Andrés Gallegos Riera de la Universidad San Francisco de Quito en el 2012, con el título “Mapa Virtual USFQ 3D: Community Aplicación Nativa”¹. Dicha tesis utiliza componentes y características de georreferenciación.

El proyecto está enfocado para que los estudiantes nuevos y antiguos, como también para las personas externas a la universidad, se encuentren informados. Es un software de georreferenciación (Mapa interactivo), que permite e informa acerca de los eventos a realizarse en el campus tales como: conferencias, concursos y a su vez es un localizador de aulas de clases, oficinas y edificios. Siendo una de las fortalezas de la información este proyecto busca la integración de esta gran comunidad universitaria. (Gallegos Riera, 2012, pág. 6)

Otra tesis desarrollada en base a la georreferenciación es la de Iván Cáceres y Gregori Larco, publicada en el 2016 con el Título “Análisis, Diseño, Desarrollo e implementación de un sistema de georreferenciación de operadores económicos para

¹ Tesis elaborada por (Gallegos Riera, 2012).

la Superintendencia de Control del Poder de Mercado”². La Superintendencia de Control de Poder de Mercado (SCPM) es una entidad pública, encarga de regular el comercio en el Ecuador.

Dicha institución no disponía de un número exacto y la ubicación geográfica de farmacias en la ciudad de Quito, como también se encontraba en la necesidad de proveer información a la ciudadanía las 24 horas del día y que se encuentre dicha aplicación al alcance gratuito de los consumidores, ya que no todas las farmacias poseen los recursos para adquirir una aplicación móvil para la carga de información de su negocio, logrando facilitar a los consumidores la tomar de decisiones para acudir a la farmacia más cercana. (Cáceres Díaz & Larco Tapia, 2016, pág. 6)

Para las personas que utilizan un cilindro de gas doméstico en su vida diaria, sería de mucha ayuda contar con una aplicación de georreferenciación que les permita tener a la mano la localización de los diferentes distribuidores que se encuentran a su alrededor dentro de la ciudad de Quito, permitiéndoles realizar pedidos de dicho producto en tiempo real, logrando con esto ahorrarles tiempo y dinero. De igual manera los diferentes distribuidores de cilindros de gas doméstico tendrán la opción de interactuar con el usuario y responder su pedido, logrando confirmar la existencia del producto y su disponibilidad para entregarlo.

La aplicación estará disponible para todas las personas que dispongan de un dispositivo móvil que maneje el sistema operativo Android, tal dispositivo permitirá utilizar la aplicación con todas las características mencionadas al cien por ciento.

² Tesis elaborada por (Cáceres Díaz & Larco Tapia, 2016)

Justificación

Se plantea elaborar un sistema móvil automatizado que permita solucionar la necesidad de adquirir un cilindro de gas doméstico mediante el envío del requerimiento a los centros de distribución y/o al vehículo distribuidor más cercano para que ellos al recibir el pedido se contacten con el cliente o si es el vehículo quien recibe el mensaje, se dirija de manera inmediata a la dirección del cliente que tiene la necesidad.

Este software proveerá al cliente con información de la ubicación de los diferentes puntos de distribución de gas tanto móviles como locales con opción para realizar pedidos de forma rápida y sencilla, los distribuidores podrán recibir el pedido y poder atender al cliente en su hogar, esto servirá de mucha ayuda a los diferentes consumidores de gas doméstico y permitirá que los distribuidores de gas, ofrezcan un mejor servicio.

Objetivo general

Desarrollo de una aplicación móvil que envíe una solicitud de requerimiento de gas doméstico a los distribuidores o a los vehículos de distribución más cercanos a una ubicación en la ciudad de Quito mediante georreferenciación.

Objetivos específicos

Analizar e investigar el desarrollo de la aplicación móvil mediante el uso de georreferenciación, para lograr una aplicación de localización automática.

Construir una aplicación móvil para el envío de solicitudes para el suministro de cilindros de gas doméstico.

Obtener información detallada de los diferentes distribuidores de gas, para ingresar los puntos de distribución de gas doméstico en el mapa de la aplicación.

Diseñar una aplicación móvil haciendo uso de la aplicación de desarrollo Android Studio.

Generar pruebas de funcionamiento y rendimiento con las personas involucradas en los dos puntos tanto cliente y distribuidores de cilindros de gas doméstico, con esto lograr una aplicación estable y sencilla de usar.

Marco metodológico

El desarrollo de aplicaciones, están sujetas a un grupo de requerimientos recolectados en base a una necesidad de la sociedad la cual se tomó como objetivo. La aplicación ayudara a automatizar estas necesidad logrando la interacción entre las tecnologías de la información y los usuarios, esto provoca que dichas aplicaciones tengan que ser creadas siguiendo unos lineamientos o guías que nos permitan hacer uso de buenas prácticas para obtener un software de excelente calidad.

En la actualidad existen un gran número de metodologías que proporcionan buenas características para el desarrollo de una aplicación móvil, las cuales permiten descubrir posibles errores o problemas que se presenten en el transcurso del desarrollo de la aplicación.

Para el presente proyecto orientado a resolver la situación actual que viven las partes involucradas al adquirir un cilindro de gas en la ciudad de Quito, es conveniente buscar una metodología que permita trabajar con una aplicación móvil, para lograr cumplir todos los requerimientos de la aplicación.

Capítulo 1

Marco Teórico

1.1 Lenguaje Unificado de Modelado

El lenguaje unificado de modelo (UML), es un lenguaje gráfico que proporciona los elementos para lograr un análisis funcional, un adecuado control de cambios y un modelo visual de las diferentes partes que involucran el desarrollo de una aplicación. El resultado final obtenido, serán diferentes diagramas, cada uno de estos representaran diferentes aspectos como la funcionalidad, definición de clases, definición de actividades, etc. Los modelos de los diagramas se crearan con la ayuda de símbolos y reglas que fueron definidas para facilitar la lectura de estos.

UML es una notación al que utiliza diagramas para mostrar distintos aspectos de un software. Es una herramienta de comunicación formal, con una serie de construcciones, una sintaxis, y una semántica definida. Así los elementos constructivos los diagramas y sus partes, la sintaxis es la descripción de cómo deben realizarse esos diagramas y la semántica define el significado de cada diagrama y elemento de los mismos. (Fontela, 2012, pág. 12)

Este lenguaje de modelamiento cuenta con diagramas de casos de uso, diagrama de clases, diagrama de navegación, diagrama de flujos de procesos, que debemos seguir para tener una documentación disciplinada.

1.1.1 Diagrama de Casos de uso.

El diagrama de caso de uso se enfoca en describir de manera general las diferentes acciones que se encuentran dentro de la funcionalidad del programa en base al punto de vista y las necesidades descritas por el usuario.

1.1.2 Diagrama de clases.

Este tipo de diagramas permiten representar de manera gráfica la estructura de las clases que generan el funcionamiento de la aplicación, es decir podemos representar todos los métodos y atributos que están dentro de una clase.

1.1.3 Diagrama de navegación.

El diagrama de navegación sirve para crear una estructura organizada de las secciones, contenidos o actividades visuales que se puede apreciar dentro de una estructura jerárquica, de la aplicación, la cual tendrá que ser creada para que pueda ser apreciada por el usuario con facilidad.

1.1.4 Diagrama de flujo de procesos.

El diagrama de flujo de procesos o también conocido como diagrama de actividades, muestra el flujo de trabajo, en el cual se debe incluir actores junto con las actividades que cada uno va a cumplir, desde el punto de inicio hasta el punto final, junto con esto se debe hacer uso de un tipo de simbología que nos permitirá tener un diagrama ordenado y entendible.

1.2 Metodología XP

La Programación Extrema o XP es una metodología de desarrollo ágil, orientada a equipos pequeños, conformados por pocos programadores. Esta metodología permite analizar, diseñar, programar e implementar aplicaciones muy volátiles en tiempos muy cortos, con una alta capacidad de respuesta, ante imprevistos cambios y errores que pueden aparecer en el transcurso del desarrollo de la aplicación, además de ofrecer una programación organizada, reduce significativamente la cantidad de documentación del proyecto respecto a otras metodologías existentes, lo cual permite que aumente la productividad y que disminuya los tiempos en el desarrollo de la aplicación.

En XP todos los requerimientos se expresan como escenarios, los cuales se implementan como una serie de tareas. Los programadores trabajan en parejas y desarrollan pruebas para cada tarea antes de escribir el código. Todas las pruebas se deben ejecutar satisfactoriamente cuando el código nuevo se ingresa al sistema. Existe un pequeño espacio de tiempo entre las entregas de los módulos del sistema. (Sommerville, 2005, pág. 364)

Algo muy interesante de esta metodología es que permite incluir al usuario para el desarrollo de la aplicación, provocando un desarrollo interactivo con la participación del cliente en niveles extremos.

El ciclo de desarrollo es una guía, la cual se ira cumpliendo conforme se va avanzando con el desarrollo de los módulos de la aplicación.

Ciclo de desarrollo.

El cliente define el valor de negocio a implementar.

El programador estima el esfuerzo necesario para su implementación.

El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.

El programador construye ese valor de negocio.

Vuelve al paso 1.

El desarrollo de una aplicación móvil de georreferenciación, conlleva un estudio previo de diferentes temas, que serán integrados para obtener una aplicación que cubrirá la necesidad prevista de la sociedad.

1.3 Aplicación Móvil o APP

Una aplicación móvil es un conjunto de procesos los cuales cumplen diferentes actividades basada en la interacción del usuario con ellas. En la actualidad esta demanda, ha hecho que el desarrollo de aplicaciones móviles vaya creciendo en base a las necesidades de diferentes empresas, negocios y muchas veces necesidades cotidianas de personas, las aplicaciones que ahora podemos encontrar son muy diversas y creadas en diferentes lenguajes. Existen aplicaciones para ventas, compras, juegos, aplicaciones para aprendizaje, etc. Estas aplicaciones son desarrolladas para que funcionen en diferentes dispositivos (celulares, tabletas, PDA).

1.4 Sistema Operativo

En el mundo la mayoría de personas cuentan con un dispositivo electrónico, que está estructurado por un sistema operativo, el cual permite que se integren dentro de su interfaz diferentes componentes, los cuales servirán como intermediario entre el sistema y el usuario. Hasta la fecha existen sistemas operativos que manejan interfaces muy cómodas y eficientes para cualquier tipo de trabajo y usuario.

En los últimos años los sistemas operativos han ido evolucionando hasta llegar a los Sistemas Operativos móviles, los cuales se encuentran de moda por las diversas características que proveen, una de las más importantes es su portabilidad, ya que puede coexistir dentro de un dispositivo móvil (celular, Tablet), permitiéndonos llevar dentro de este, un escritorio de aplicaciones que se encuentran disponibles en cualquier momento. Ahora no es necesario tener un computador portátil a la mano para realizar una actividad, ya que existen versiones de sistemas operativos para dispositivos móviles que cumplen con muchas de las funciones de un computador.

Los tipos de software o sistemas operativos móviles que existen en la actualidad.

Android

IOS

Symbian

1.4.1 Android.

La aplicación se desarrollara utilizando el sistema operativo Android, el cual se eligió por todas las características que provee y por ser unos de los sistemas operativos más acogidos en el Ecuador. Es un sistema operativo gratuito, multiplataforma y muy ordenado en su estructura de programación, su lenguaje está basado en Linux y su programación es muy similar al lenguaje de Java. “Este sistema operativo permite programar aplicaciones empleando una variación de Java llamada Dalvik, y proporciona todas las interfaces necesarias para desarrollar fácilmente aplicaciones que acceden a todas las funciones del teléfono”³, por ejemplo.

Interacción con el GPS del dispositivo.

Manejo de imágenes en 2D y 3D

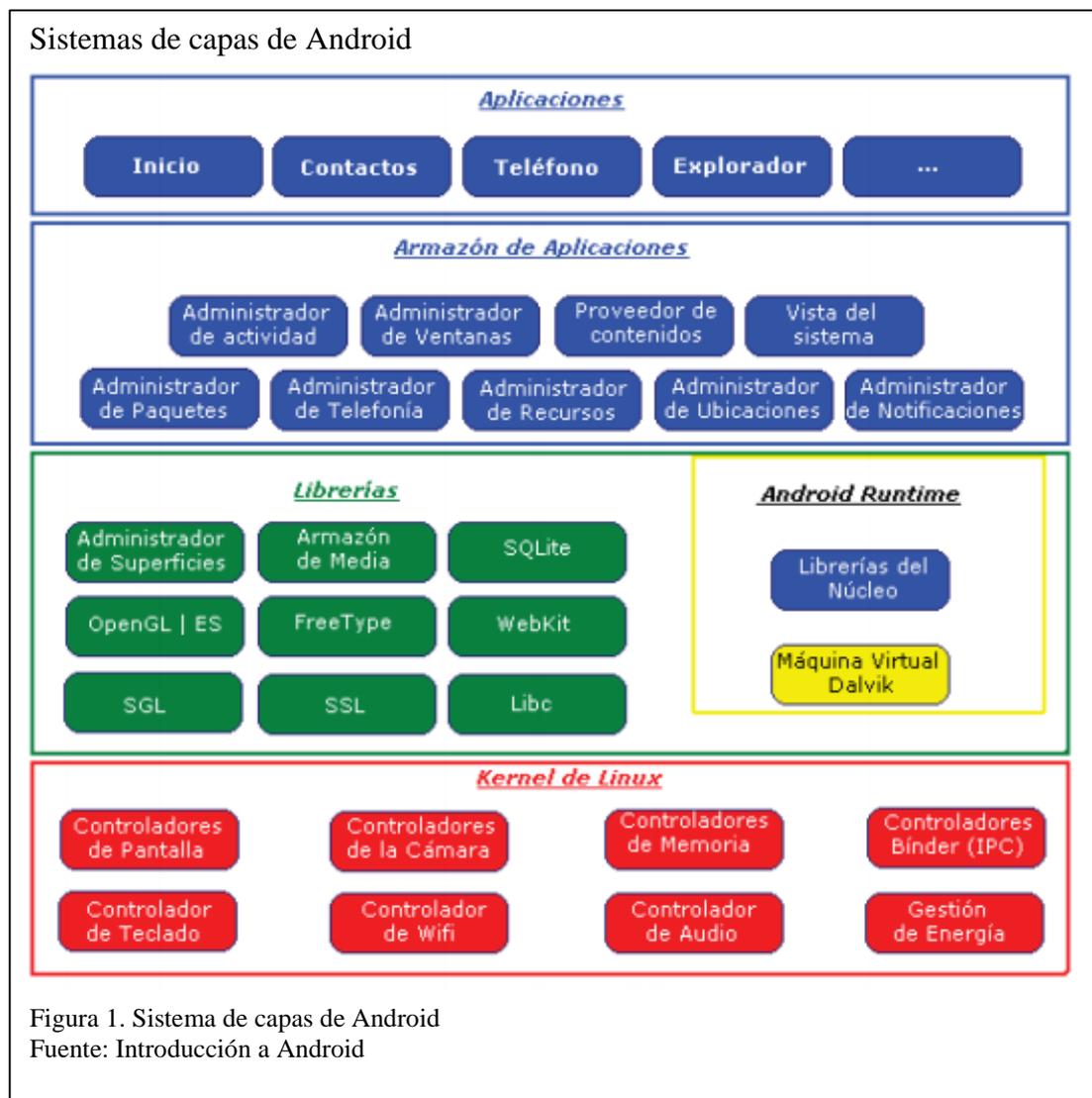
Conexión a múltiples base de datos.

Interacción con programas internos (Mensajes, Llamadas, Hotmail).

Formatos multimedia.

La estructura de Android está dividida en diferentes capas, las cuales se van a presentar a continuación.

³ Elaborado por: (Robledo Sacristán & Robledo Fernández, 2012)



En la Figura 1 se muestra la arquitectura que maneja el Kernel de Linux, la cual está distribuida por diferentes capas, esta estructura permite que Android ingrese de manera jerárquica a los diferentes segmentos que se encuentran en la estructura (Capa de aplicaciones, Librerías y Kernel). En la primera capa puede acceder a los controladores y a las diferentes librerías creadas para el desarrollo de aplicaciones, la segunda capa proporciona la organización para los administradores de recursos y la última capa proporciona las aplicaciones a las que tiene acceso.

1.5 Entorno de desarrollo Android Studio

Android Studio es un entorno de desarrollo integrado (IDE), basado en IntelliJ IDEA de la compañía JetBrains, que proporciona varias mejoras con respecto al plugin ADT (Android Developer Tools) para eclipse. Android Studio utiliza una licencia de software libre Apache 2.0. (Deza Castillo, 2017)

Es una plataforma exclusiva para el desarrollo de aplicaciones, maneja como lenguaje nativo Java, presenta una estructura ordenada de programación, lo que resulta de mucha ayuda para el diseño de aplicaciones.

1.6 Georreferenciación

En la actualidad la georreferenciación es un método muy utilizado por diferentes aplicaciones web y aplicaciones móviles, permitiendo referenciar diferentes puntos (personas, negocios, vehículos) dentro en un mapa (Google Maps). La ubicación del objeto se realiza por medio de la comunicación de un servidor en conjunto con una aplicación móvil y un dispositivo que contenga GPS para después presentar las coordenadas obtenidas del servidor dentro de un mapa que se encontrara en el sistema desarrollado

1.7 Google Maps

Es una aplicación web que se encuentra dentro de un servidor la misma que provee de mapas desplegable y fotografías del mundo entero tomadas por satélites en tiempo real, permite trazar rutas entre diferentes puntos, logrando conocer la distancia, el tiempo de llegada y la existencia de tráfico en una ruta determinada, etc. Los usuarios tienen diversas posibilidades para interactuar con esta aplicación, ya sea con el puntero de mouse o el teclado para ingresar una ubicación requerida. Google Maps tiene

librerías que se pueden integrar a una aplicación móvil, estas librerías contienen diferentes características que sirven para hacer uso del GPS del dispositivo.

1.8 Api de Google Android Maps

Api de Google Android Maps es una librería creada por Google para facilitar el uso de la Georreferenciación o Geolocalización dentro de la aplicación móvil, su estructura está compuesta por atributos, métodos y componentes que sirven para interactuar con el usuario, estos componentes permiten modificar los mapas tradicionales obtenidos de Google Maps a las necesidades del usuario, hay que tener en cuenta el dispositivo que va a servir de huésped debe contener GPS, ya que la librería interactuara directamente con el hardware del dispositivo para hacer uso de todas las características que esta librería proporciona.

1.9 Gestor De Base De Datos

Un gestor de base de datos (SGBD) está conformado por un conjunto de programas que sirven para crear, modificar, extraer y gestionar información de una base de datos por medio de líneas de comandos y una interface amigable.

En la actualidad existen diversos gestores de base de datos, cada uno de estos fueron creados para diferentes ambientes de trabajo por ejemplo SQL, ORACLE funcionan muy bien en aplicaciones de escritorio dentro de una red LAN, SQLite es un gestor de base de datos exclusivo para aplicaciones móviles, Mysql es uno de los gestores más utilizados en la actualidad por los desarrolladores de aplicaciones web y aplicaciones móviles.

1.9.1 MySQL.

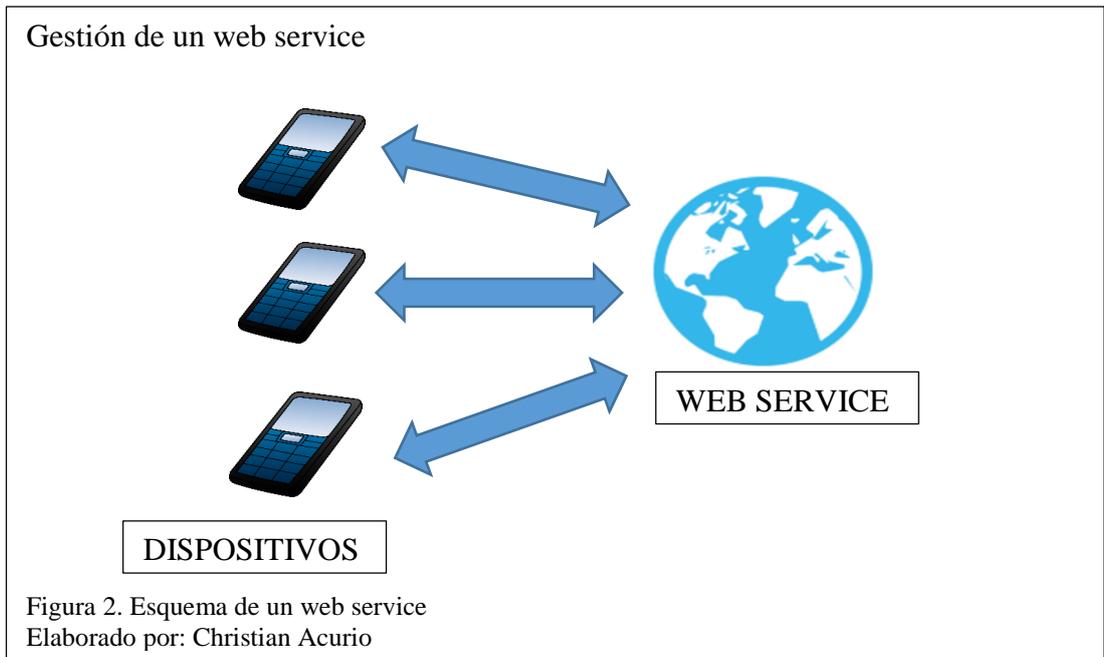
Es un sistema de administración de base de datos relacionales rápido, sólido y flexible, permite crear base de datos con acceso desde páginas web dinámicas, para la creación de sistemas de transacción on-line o para cualquier otra solución profesional que indique almacenar datos, teniendo la posibilidad de realizar múltiples y rápidas consultas. (Cobo, Gómez, Pérez, & Rocha, 2005, pág. 339)

Gracias a sus características se puede usar en conjunto con un web services, en el que se integrara la base de datos que permita almacenar información para después poder usarla, teniendo la posibilidad de acceder a los datos almacenados en esta, desde cualquier lugar del mundo, ya que estará disponible las 24 horas del día.

1.10 Web services

Un web service es la parte lógica del negocio, capaz de procesar y accesible desde cualquier lugar, para personas que tengan permisos para ello, a través de cualquier medio. Mas explícitamente, un web service es una interface hacia una aplicación o proceso accesible vía red informática mediante cualquier tipo de tecnología orientada a internet, tales como FTP, HTTP, SMTP. (Ribas Lequerica, 2003, pág. 10)

Un web service es un espacio en la nube en el cual se puede implementar una máquina virtual con cualquier tipo de sistema operativo ya sea Windows o Linux, lo que permitirá instalar los componentes necesarios para lograr la interacción de las diferentes aplicaciones, ya sean páginas web o aplicaciones móviles. La comunicación se realiza mediante una IP pública que el proveedor del servicio se encarga en asignar.



1.10.1 Windows Azure.

Windows Azure es una plataforma en la nube, en la que se puede integrar un sin número de recursos para administrar diferentes tipos de aplicaciones ya sean estas web o móviles, desde cualquier parte del planeta.

Capítulo 2

Análisis y Diseño

La aplicación móvil a se desarrolló aplicando la estructura que la metodología XP nos proporciona, partiendo desde el análisis de requerimientos, los diferentes diagramas del lenguaje UML, la estructura de la base de datos, hasta el final que se detallan la interfaces que se implementaran en la aplicación móvil.

3.1 Análisis de requerimientos

En el análisis de requerimientos se detallan las funciones de la aplicación móvil (las interfaces, su funcionalidad y los roles de los usuarios), estos deben ser detallados metódicamente, ya que serán de mucha ayuda para el desarrollo de la aplicación.

3.1.1 Identificación de usuarios.

Antes de describir las historias de usuarios, se definirán los involucrados y los roles que cada uno cumplirá en el funcionamiento de la aplicación móvil.

Distribuidores de gas doméstico: estos usuarios son los encargados de compartir su ubicación una vez que hayan iniciado sesión por medio de la aplicación móvil logrando que los diferentes clientes pueden ubicarlos por medio de un mapa que se encontrara dentro de un módulo de la aplicación.

Iniciar la aplicación

Ingreso al módulo de distribuidor

Registro de usuario

Inicio sesión

Envío de coordenadas

Actualización de contraseña

Cliente (Usuario general): son encargados de ubicar a los diferentes distribuidores que se encuentran registrados en la base de datos, los cuales serán presentados en un mapa que tendrá una opción que permitirá realizar pedidos con un mensaje de texto por medio de la aplicación.

Iniciar la aplicación

Ingreso al módulo cliente

Ubicar los distribuidores de gas doméstico más cercanos

Realizar pedidos al distribuidor seleccionado

Interactuar con el distribuidor por medio de mensajes de texto

3.1.2 Historia de usuarios.

La metodología XP permite definir los requerimientos de los usuarios que están involucrados en el desarrollo de la aplicación móvil, estos deberán ser detallados de manera clara y concisa, junto con los involucrados. Para representarlos se utilizaran tablas, las cuales contendrán cada uno de los requerimientos mencionados anteriormente.

3.1.3 Requerimiento usuario Distribuidor de gas doméstico.

Se detallarán las diferentes actividades que serán realizadas por el usuario que tendrá como rol de distribuidor de gas doméstico, las cuales fueron planteadas por el desarrollador y el usuario.

Tabla 1 detalla el requerimiento del distribuidor para ingresar la información de cada uno que conforme este grupo.

Tabla 1. Historia de usuario registro de distribuidores de gas doméstico

Numero: 1		Usuario: Distribuidor de gas	
Nombre historia: registro de distribuidores de gas doméstico			
Prioridad del negocio: Alta		Riesgo de desarrollo: medio	
Puntos estimados: 2		Iteración asignada: 1	
Programador responsable: Acurio Christian			
Descripción: Los distribuidores serán capaces de ingresar su información para posteriormente poder iniciar sesión.			
Observaciones: Verificado por el cliente			

Notas. Historias de usuario para registrar su información dentro de la aplicación móvil

Elaborado por: Christian Acurio

Tabla 2 detalla la necesidad del distribuidor para que cada usuario que se encuentre registrado en la aplicación móvil pueda iniciar sesión.

Tabla 2. Historia de usuario control de acceso distribuidor

Numero: 2		Usuario: Distribuidor de gas	
Nombre historia: Control de acceso distribuidores			
Prioridad del negocio: Alta		Riesgo de desarrollo: Alto	
Puntos estimados: 2		Iteración asignada: 1	
Programador responsable: Acurio Christian			
Descripción: Para ingresar a la función distribuidor, la aplicación debe solicitar credenciales para así poder compartir sus coordenadas.			
Observaciones: Verificado por el cliente			

Nota. Historia de usuario para controlar el ingreso a los distribuidores registrados.

Elaborado por: Christian Acurio }

Tabla 3 se refleja la necesidad de compartir la ubicación de los diferentes distribuidores que usen la aplicación móvil.

Tabla 3. Historia de usuario compartir su ubicación

Numero: 3		Usuario: Distribuidor de gas	
Nombre historia: Compartir ubicación			
Prioridad del negocio: Alta		Riesgo de desarrollo: Alto	
Puntos estimados: 2		Iteración asignada: 1	
Programador responsable: Acurio Christian			
Descripción: Los distribuidores al iniciar sesión tendrán la opción de compartir sus coordenadas con los diferentes clientes.			
Observaciones: Verificado por el cliente			

Nota. Historia de usuario para compartir sus coordenadas o ubicación con sus diferentes clientes.
Elaborado por: Christian Acurio

Tabla 4 se expresa el requerimiento de los distribuidores, por tener la opción de cada uno poder cambiar su contraseña de acceso.

Tabla 4. Historia de usuario cambio de contraseña distribuidor

Numero: 4		Usuario: Distribuidor de gas	
Nombre historia: Cambio de contraseña distribuidor			
Prioridad del negocio: Alta		Riesgo de desarrollo: Medio	
Puntos estimados: 2		Iteración asignada: 1	
Programador responsable: Acurio Christian			
Descripción: Los distribuidores registrados individualmente podrán cambiar su contraseña			
Observaciones: Verificado por el cliente			

Nota. Cada uno de los usuarios podrá modificar su contraseña de ingreso.
Elaborado por: Christian Acurio

3.1.4 Requerimiento cliente.

Se detallará las diferentes actividades que serán realizadas por el usuario que tendrá como rol de cliente (consumidor de cilindros de gas doméstico), las cuales fueron planteadas por el desarrollador y el usuario.

Tabla 5 detalla los requerimientos del usuario para poder buscar al distribuidor más cercano respecto a su ubicación actual dentro de la ciudad de Quito.

Tabla 5. Historia de usuario buscar distribuidores cercanos a su ubicación actual

Numero: 5	Usuario: Cliente
Nombre historia: Buscar distribuidores cercanos a su ubicación actual	
Prioridad del negocio: Alta	Riesgo de desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Acurio Christian	
Descripción: El cliente será capaz de visualizar dentro de un mapa los distribuidores que se encuentren cerca de su ubicación actual dentro de la ciudad de Quito.	
Observaciones: Verificado por el cliente	

Nota. Historia de usuario para buscar distribuidores cercanos a su ubicación actual

Elaborado por: Christian Acurio

Tabla 6 detalla los requerimientos del cliente de la aplicación para visualizar la información del distribuidor seleccionado.

Tabla 6. Historia de usuario visualizar información de distribuidores

Numero: 6	Usuario: Cliente
Nombre historia: Visualizar información de distribuidores	
Prioridad del negocio: Alta	Riesgo de desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Acurio Christian	
Descripción: El cliente será capaz de visualizar la información del distribuidor al que se realizará el pedido de dicho producto.	
Observaciones: Verificado por el cliente	

Nota. Historia de usuario para visualizar la información del distribuidor que se seleccionó para realizar un pedido

Elaborado por: Christian Acurio

Tabla 7 refleja la necesidad del cliente para poder realizar pedidos de cilindros de gas doméstico a un distribuidor que se encuentre más cerca a su posición actual.

Tabla 7. Historia de usuario realizar pedidos al distribuidor más cercano

Numero: 7		Usuario: Cliente	
Nombre historia: Realizar pedidos al distribuidor más cercano			
Prioridad del negocio: Alta		Riesgo de desarrollo: Alta	
Puntos estimados: 2		Iteración asignada: 1	
Programador responsable: Acurio Christian			
Descripción: El cliente será capaz de realizar pedidos por medio de un mensaje a los distribuidores que se encuentren cerca de su ubicación actual, dentro de la ciudad de Quito.			
Observaciones: Verificado por el cliente			

Nota. Historia de usuario para realizar pedidos al distribuidor más cercano a su ubicación actual
 Elaborado por: Christian Acurio

3.2 Diseño del aplicativo

El diseño está constituido por un grupo de diagramas, esquemas que el lenguaje UML permite describir. Estos diagramas demuestran diferentes funcionalidades, los cuales serán de ayuda para los encargados de desarrollar la aplicación móvil. Se detallará el diagrama de casos de uso, los diagramas de flujo, diagrama de clases, diagrama de navegación, también se detallará el diagrama físico de la base de datos junto con el diccionario de datos, el cual explicara cada una de las tablas junto con sus campos de datos, culminando con la descripción de las distintas interfaces gráficas de la aplicación móvil.

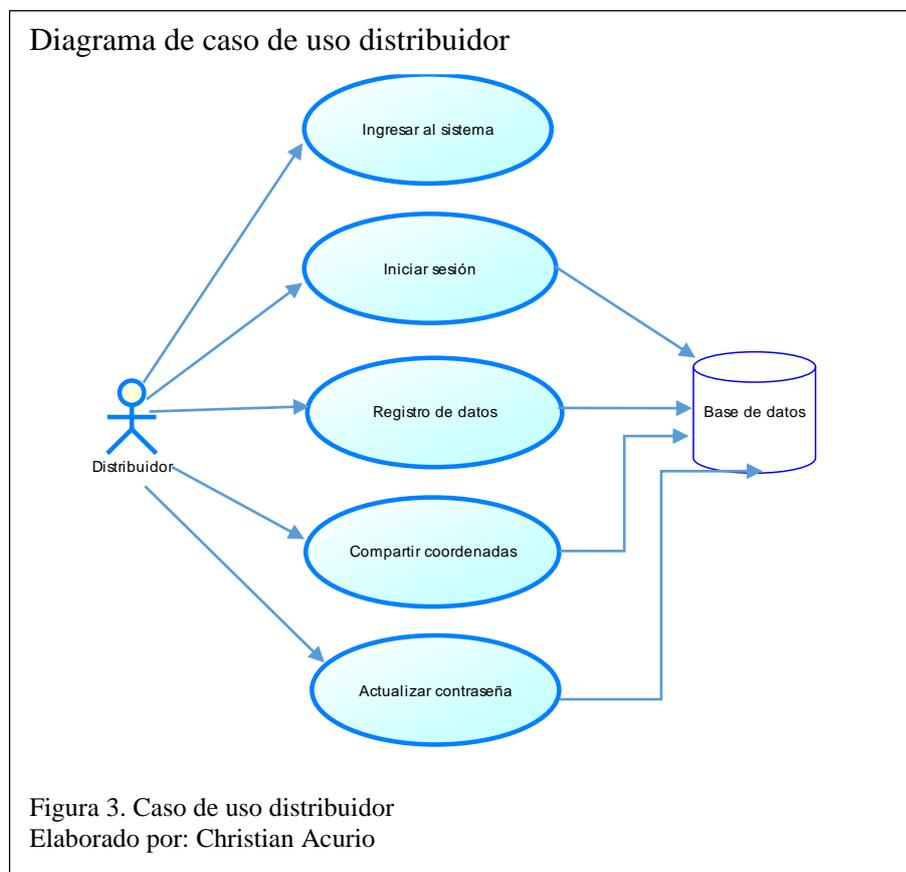
3.2.1 Diagrama de casos de uso.

El diagrama de casos de uso permite crear una representación gráfica de los diferentes módulos de la aplicación móvil, que serán usados por los usuarios definidos anteriormente, logrando describir en estas la interacción de cada uno de los involucrados.

3.2.1.1 Diagrama de caso de uso distribuidor de gas doméstico.

Las actividades que el usuario distribuidor de gas doméstico realizará se detallarán en el siguiente diagrama

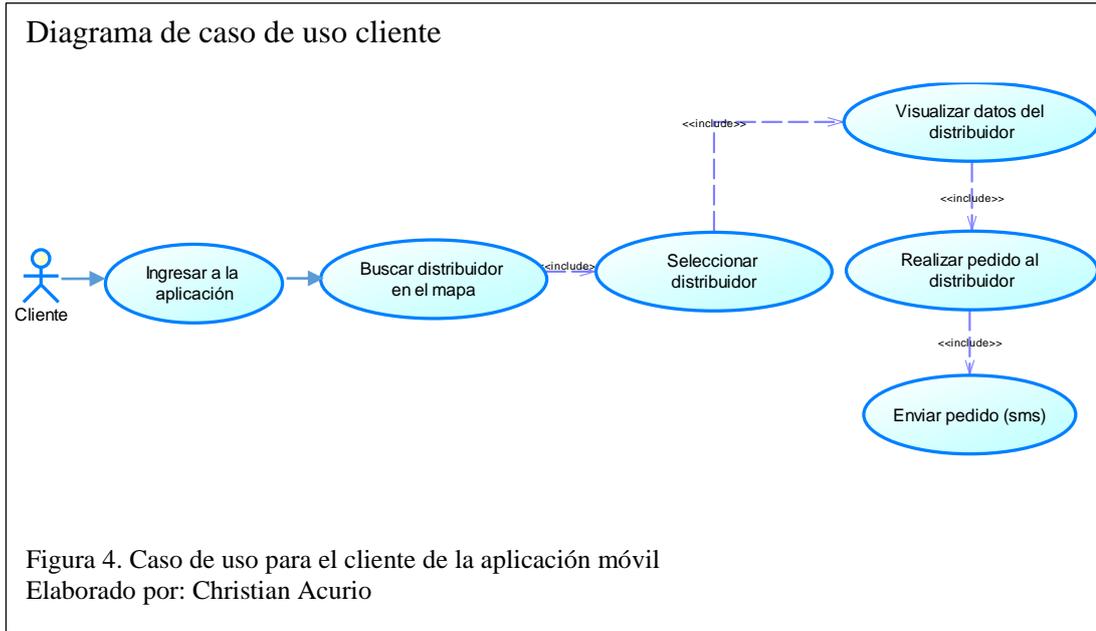
Figura 3 describe las actividades principales como los sub actividades que el usuario distribuidor puede realizar dentro de la aplicación móvil. Estas actividades son ingresar al sistema, inicio sesión, registro distribuidores, compartir coordenadas.



3.2.1.2 Diagrama de caso de uso cliente.

Este diagrama describe las funcionalidades que realiza un usuario general.

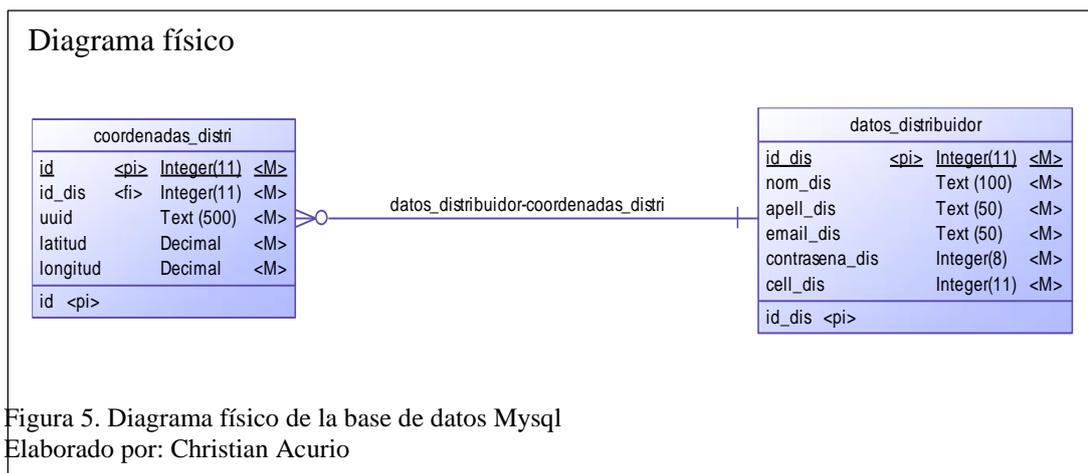
Figura 4 describe las actividades principales como las sub actividades que el cliente (usuario general) puede realizar dentro de la aplicación móvil.



3.2.2 Diagrama físico de la base de datos MySQL.

La base de datos ha sido creada, pensando en la funcionalidad que debe tener la aplicación móvil, para lograr almacenar la información necesaria para la interacción de la base de datos y los diferentes módulos de la aplicación.

Figura 5 detalla las características de cada una de las tablas que conforman la base de datos de la aplicación móvil. Se especifican los campos, las llaves primarias, las llaves secundarias y sus relaciones.



A continuación se presenta una tabla basada en el diseño conceptual de la base de datos, la cual contendrá la descripción de los diferentes campos que contienen las tablas.

Tabla 8. Descripción de las tablas de la base de datos

Tablas	Descripción
datos_distribuidor	Almacena los datos de los distribuidores ingresados por medio de la aplicación.
coordenadas_dis	Almacena la información necesaria para recolectar las coordenadas de los distribuidores.

Nota. Descripción de las tablas.
Elaborado por: Christian Acurio

3.2.3 Diccionario de datos.

El diccionario de datos detalla las características de los diferentes campos (tipos de datos, su descripción y tamaño del campo) de las tablas que constituyen la base de datos que sirve para almacenar la información necesaria para el funcionamiento de la aplicación móvil.

Tabla 9 muestra el diccionario de datos de la tabla datos_distribuidor, la cual almacena la información de los distribuidores registrados en la aplicación móvil.

Tabla 9. Diccionario de la tabla datos_distribuidor

Nombre de la tabla	datos_distribuidor		
Nombre del campo	Tipo	Tamaño	Descripción
id_dis	INTEGER	11	Identificador de registro
nom_dis	TEXT	100	Nombre del distribuidor
apell_dis	TEXT	50	Apellido del distribuidor
email_dis	TEXT	50	Correo electrónico del distribuidor
contraseña_dis	INTEGER	8	Contraseña ingresado del distribuidor
cell_dis	INTEGER	11	Numero celular ingresado por el distribuidor

Nota. Descripción de los campos de la tabla datos_distribuidor

Elaborado por: Christian Acurio

La tabla 10 muestra el diccionario de datos de la tabla coordenadas_distri, esta tabla almacena las coordenadas que se utilizan para ubicar a los diferentes distribuidores registrados.

Tabla 10. Diccionario de datos tabla coordenadas_distri

Nombre de la tabla	coordenadas_distri		
Nombre del campo	Tipo	Tamaño	Descripción
id	INTEGER	11	Identificador de registro
uuid	TEXT	500	Identificador único del dispositivo móvil
latitud	DOUBLÉ	0	Coordenada de ubicación
longitud	DOUBLÉ	0	Coordenada de ubicación

Nota. Descripción de los campos de la tabla coordenadas_distri

Elaborado por: Christian Acurio

3.2.4 Diagrama de clases.

El diagrama de clases permite describir las clases utilizadas en el sistema para la gestión lógica y funcional de la aplicación móvil, es decir permite detallar los métodos y atributos utilizados en cada uno de las clases.

Diagrama de clases

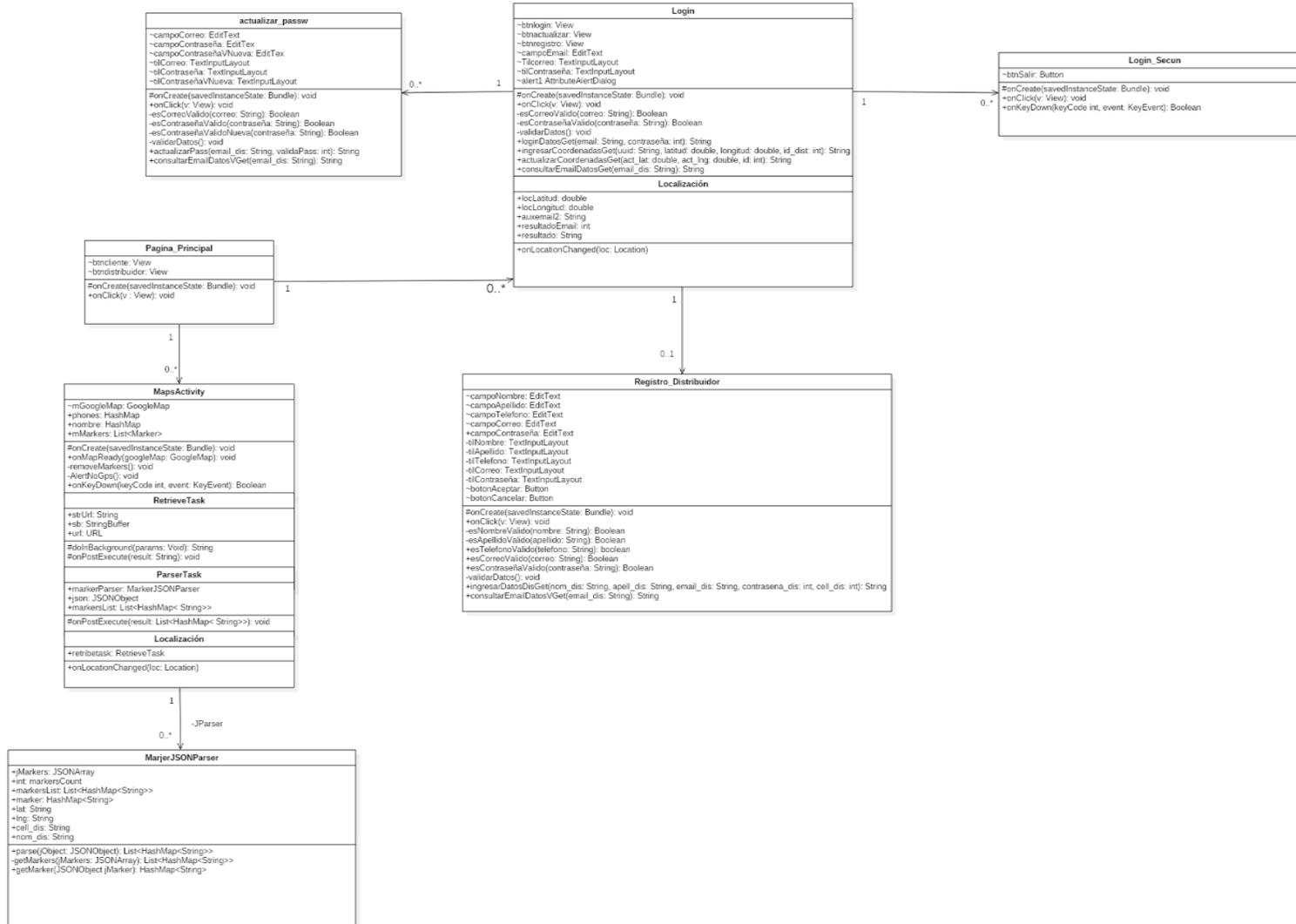


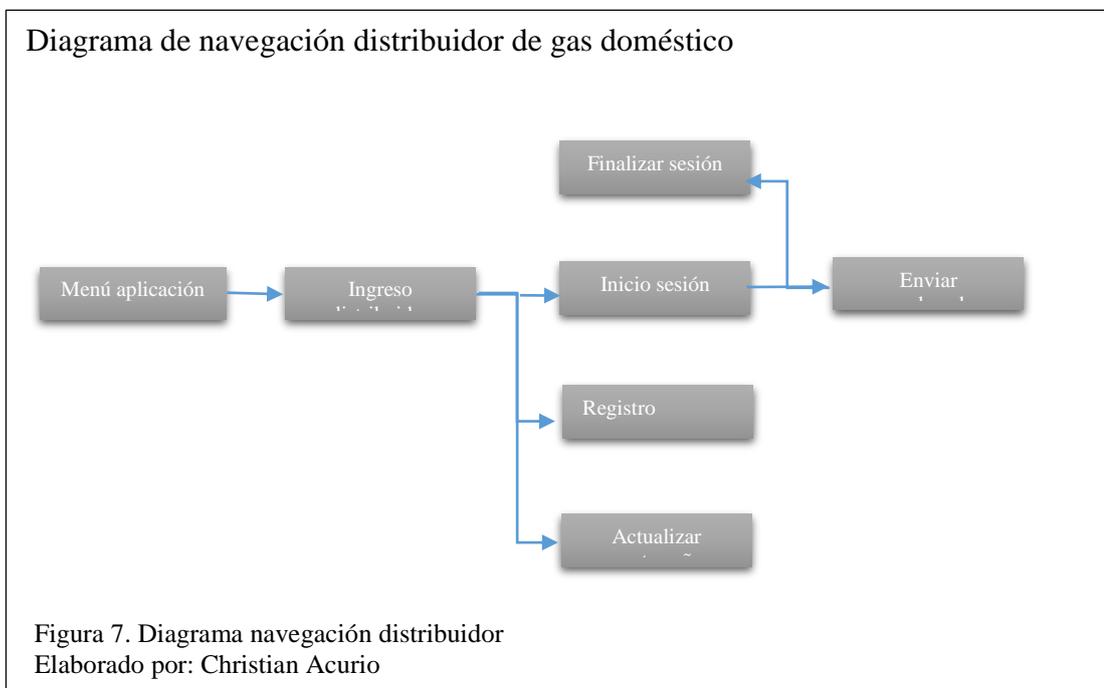
Figura 6. Diagrama de clases correspondiente a la aplicación móvil
Elaborado por: Christian Acurio

3.2.5 Diagrama de navegación.

Estos diagramas representan gráficamente las acciones que los usuarios pueden realizar al interactuar con los diferentes módulos que la aplicación móvil provee, las acciones serán descritas para cada uno de los roles de usuarios descritos anteriormente en este documento (Distribuidor y Cliente).

3.2.5.1 Diagrama de navegación distribuidor de gas doméstico.

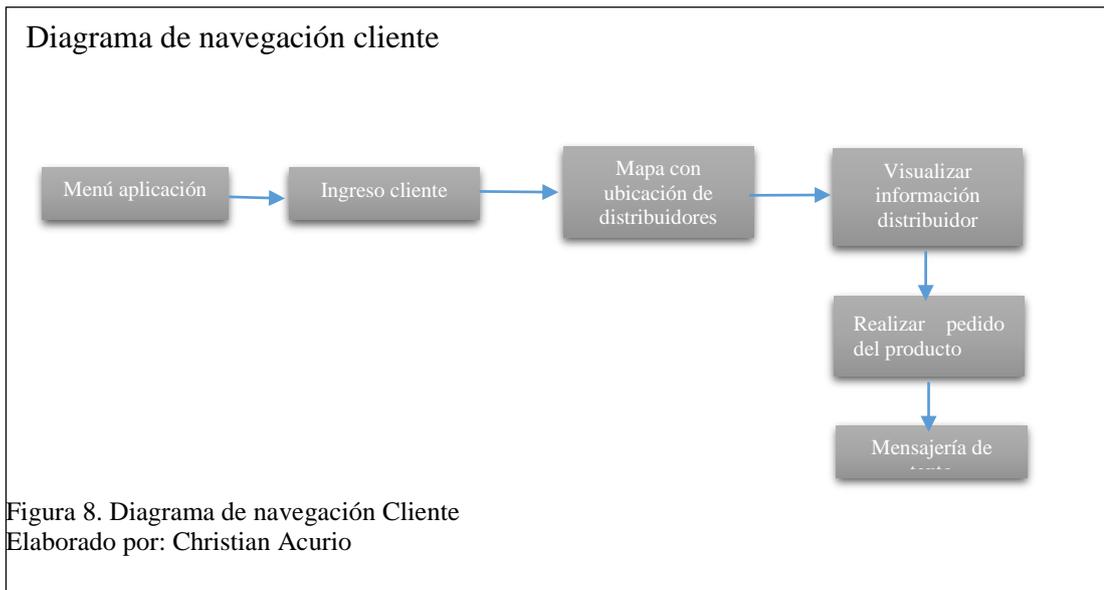
Este diagrama describirá todas las acciones posibles que el usuario (Distribuidor de gas doméstico) podrá realizar al navegar por la aplicación.



3.2.5.2 Diagrama de navegación cliente.

Este diagrama describe todas las acciones posibles que el usuario (Cliente) podrá realizar al navegar por la aplicación.

En la Figura 8 podemos apreciar que el usuario que ingrese con el rol de cliente, podrá hacer uso de las diferentes acciones de la aplicación.



3.2.6 Diagrama de flujo.

El diagrama de flujo describe los diferentes procesos que abarca cada uno de las acciones que los usuarios pueden realizar en la aplicación móvil, esta guía nos facilita la visualización del flujo de los procesos que abarcan las acciones que contiene dicha aplicación, siendo de mucha ayuda para interpretar las características que se deben plasmar.

En la figura 9 se puede apreciar una guía que contiene los procesos orientados a los distintos usuarios, los cuales harán uso de la misma.

Diagrama de flujo aplicación móvil

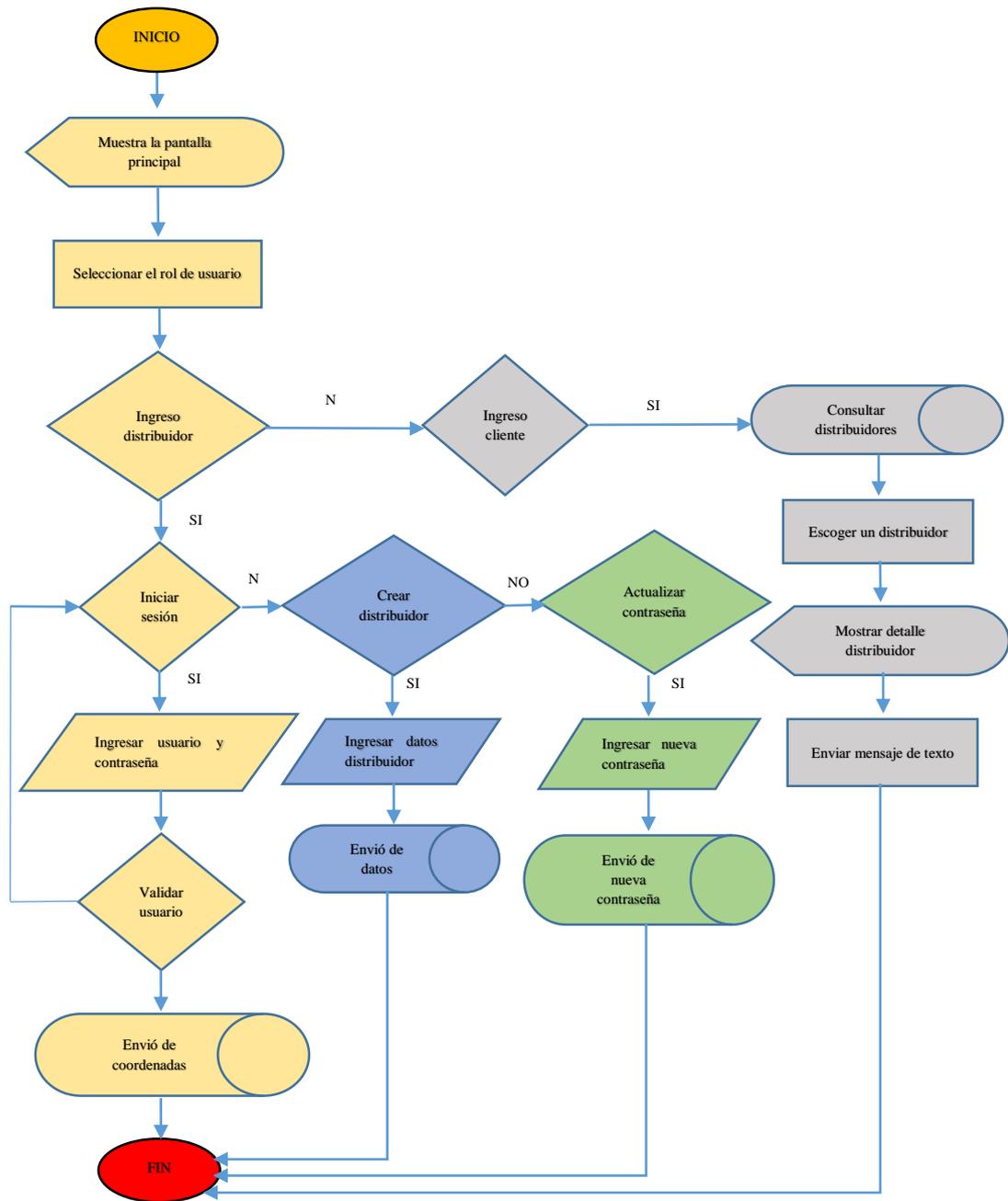
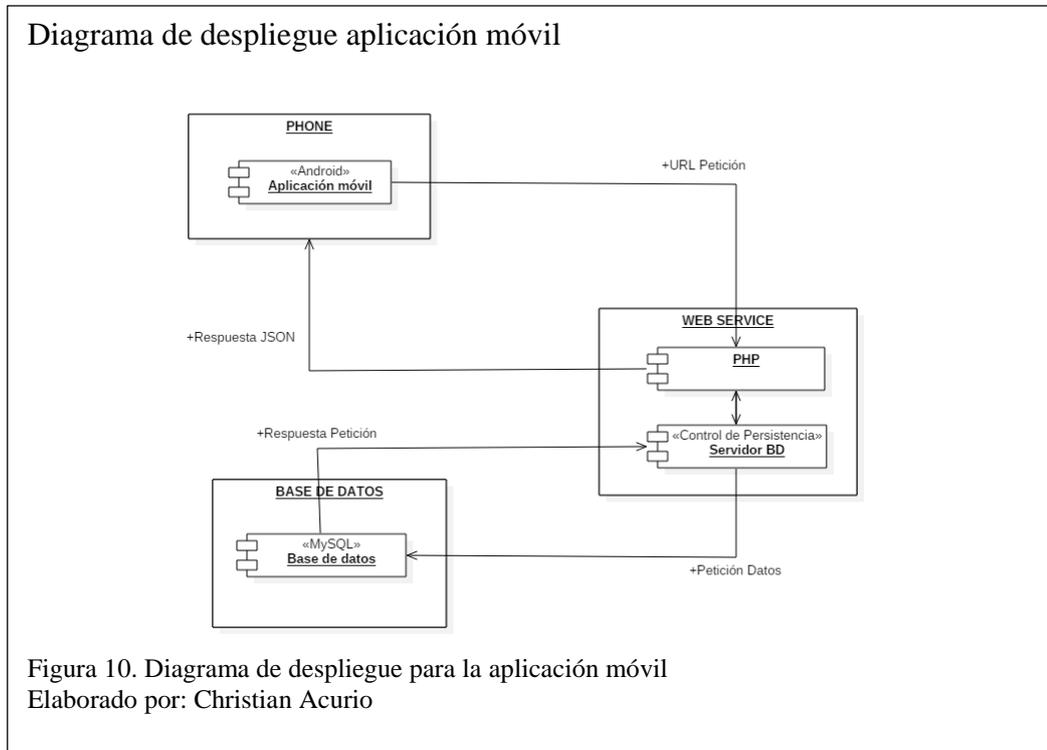


Figura 9. Diagrama de flujo de la aplicación móvil
Elaborado por: Christian Acurio

3.2.7 Diagrama de distribución.

El diagrama de distribución permite modelar la estructura que conforma los elementos del software y hardware, los cuales permiten el funcionamiento de la aplicación móvil.



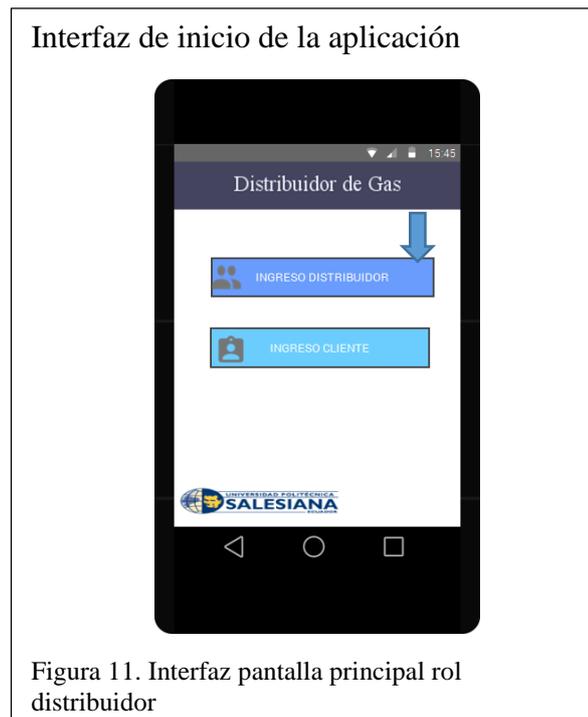
3.2.8 Prototipo de la interfaz gráfica.

En esta sección se diseñaron prototipos de todas las interfaces gráficas, junto con cada una de las funcionalidades que contendrá la aplicación móvil, las cuales permitirán a los desarrolladores tener una guía para que el diseño de las interfaces pueda ser de manera clara y amigable y así cada uno de los usuarios hagan uso de las mismas, dependiendo de los roles que ocupe cada uno de estos dentro de las funciones de la aplicación.

3.2.8.1 *Interfaz aplicación móvil para el usuario distribuidor de gas doméstico.*

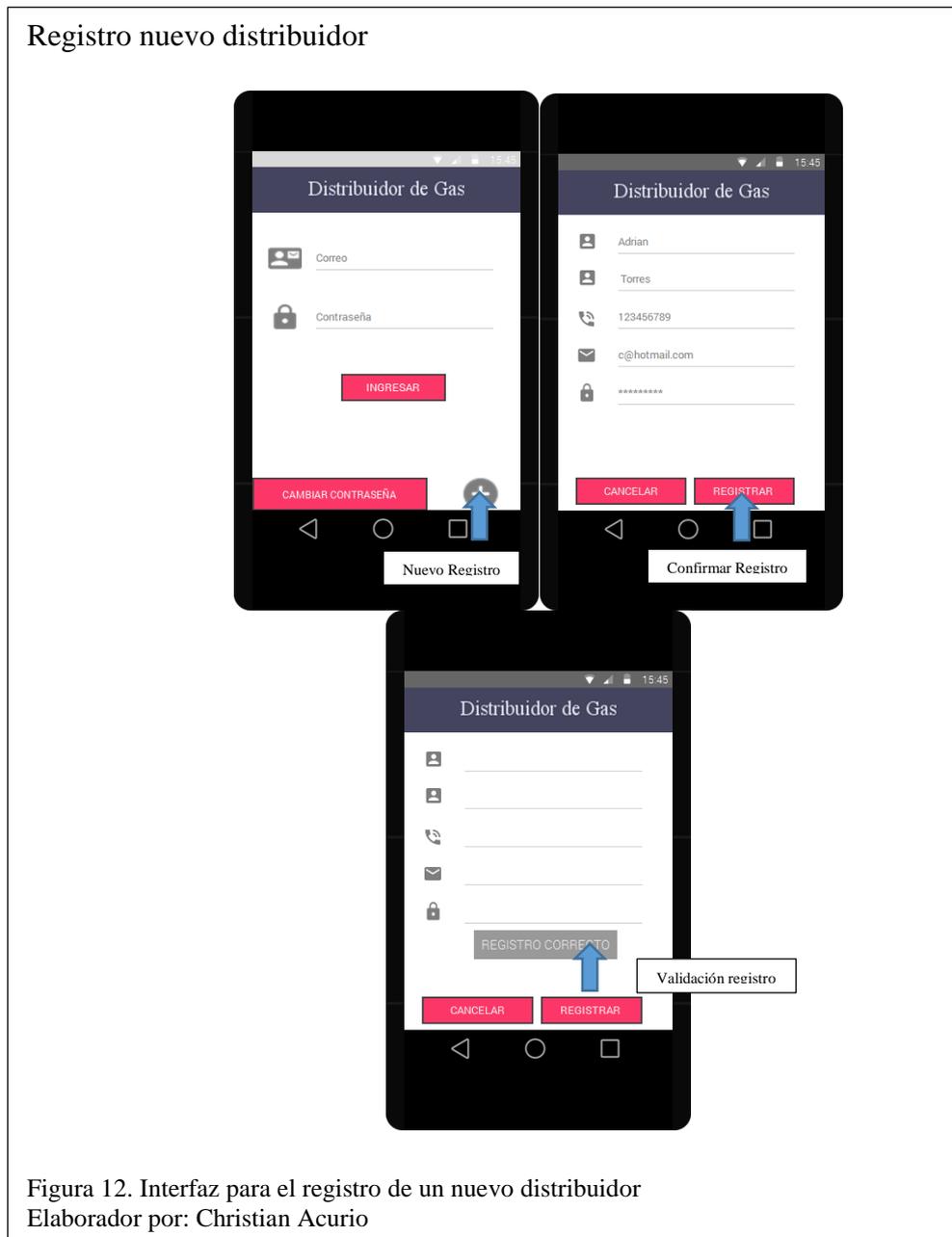
A continuación se presentara las pantallas de la aplicación y cada una de las funcionalidades que podrá realizar el usuario distribuidor.

3.2.8.1.1 *Interface de inicio de la aplicación móvil.*



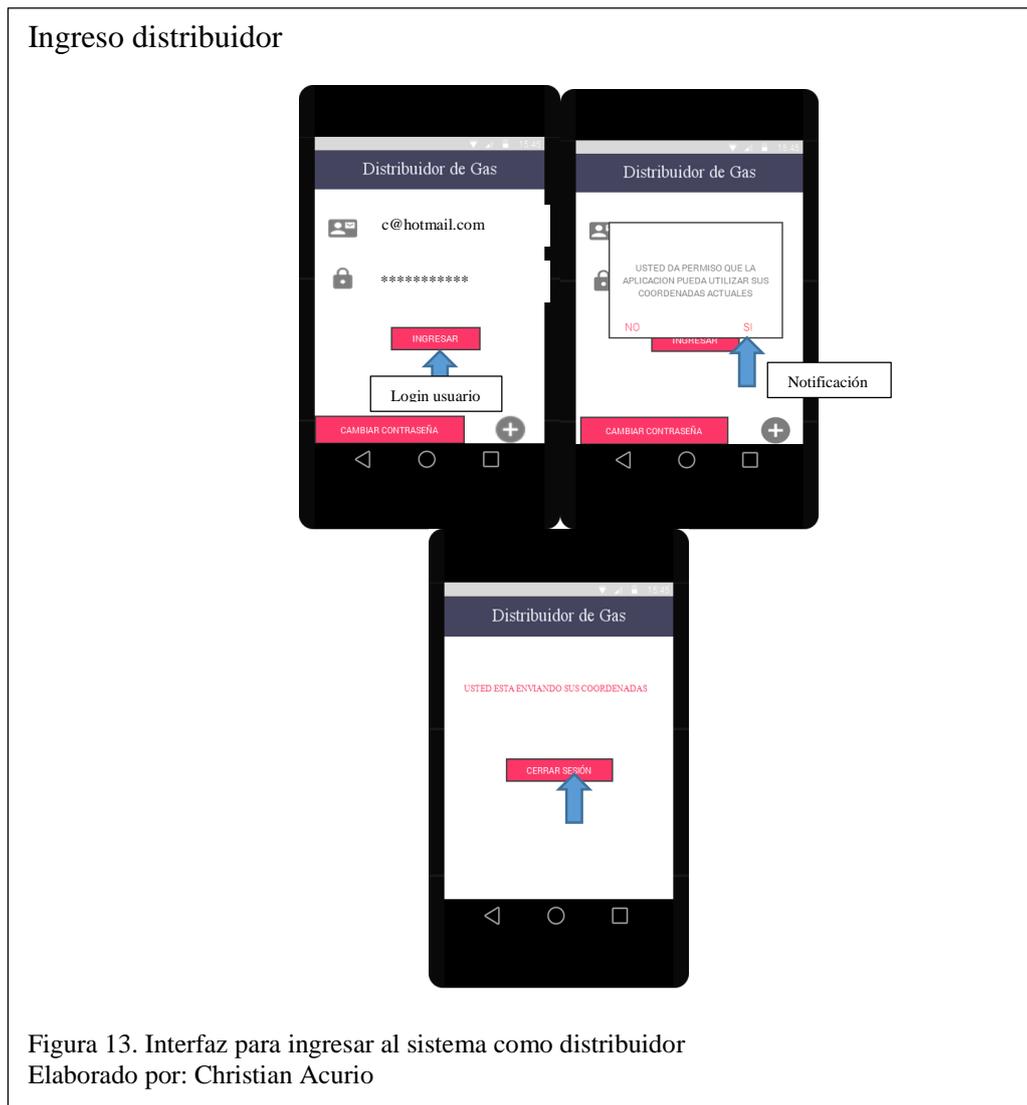
En la Figura 11 se ilustra la pantalla principal que será visible para todos los usuarios, como se puede observar está estructurada por dos botones, cada uno tiene diferente función, el botón ingreso distribuidor tiene la función de permitir que el usuario con el rol de distribuidor ingrese a sus funciones, tal como se explica con las siguientes ilustraciones.

3.2.8.1.2 Interface para el registro de distribuidor.



En la Figura 12 se puede apreciar las interfaces involucradas en el registro de un nuevo distribuidor, para esto el usuario tendrá que dar clic en el botón nuevo registro, a continuación tendrá que ingresar correctamente los datos solicitados, para finalizar el usuario tiene que dar clic en el botón registrar, el resultado será una notificación que confirma el registro.

3.2.8.1.3 Interface para el ingreso a la función distribuidor.



La Figura 13 ilustra el proceso que realizara en usuario distribuidor para compartir las coordenadas respecto a su ubicación, primero tiene que ingresar su correo electrónico y la contraseña, los cuales fueron ingresados posteriormente en el registro, al dar clic en el botón ingresar aparecerá una notificación que pregunta al usuario si desea que la aplicación puede enviar las coordenadas con su ubicación actual, si el usuario acepta dará como resultado una pantalla que informa al usuario que se está enviado sus coordenadas, el usuario tiene la opción de cerrar su sesión cuando desee.

3.2.8.1.4 Interface para actualizar contraseña distribuidor.

En la siguiente figura se detallará la interfaces que involucran el cambio de contraseña de un usuario distribuidor.



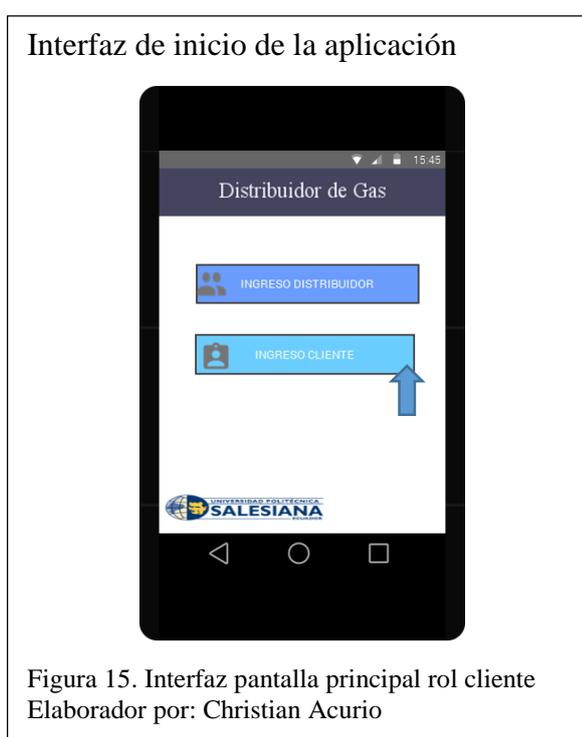
La Figura 14 detalla el proceso que tiene que seguir el usuario con el rol distribuidor para realizar el cambio de contraseña, en caso de que este haya olvidado la ingresada, para lograr esto el usuario tiene que ir al botón cambiar contraseña, se desplegará una nueva pantalla en la que se solicitará el correo electrónico y la nueva contraseña junto con otro campo que servirá para validar la nueva contraseña, finalmente el usuario

tendrá que presionar el botón actualizar para confirmar el cambio si el proceso fue exitoso el programa desplegará un mensaje de conformación.

3.2.8.2 *Interfaz aplicación móvil para el usuario Cliente.*

En esta sección se detallará las interfaces junto a las funciones que involucran al usuario con el rol de Cliente.

3.2.8.2.1 *Interface de inicio de la aplicación móvil.*



En la Figura 15 se describe la pantalla principal de la aplicación móvil para el ingreso de un cliente (Usuario general).

3.2.8.2.2 Interfaz visualización de distribuidores en el mapa de google.

Visualización de distribuidores en google maps

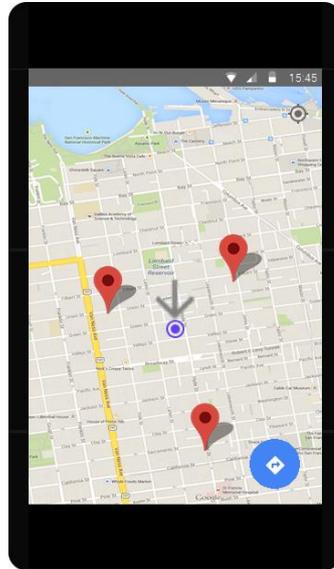
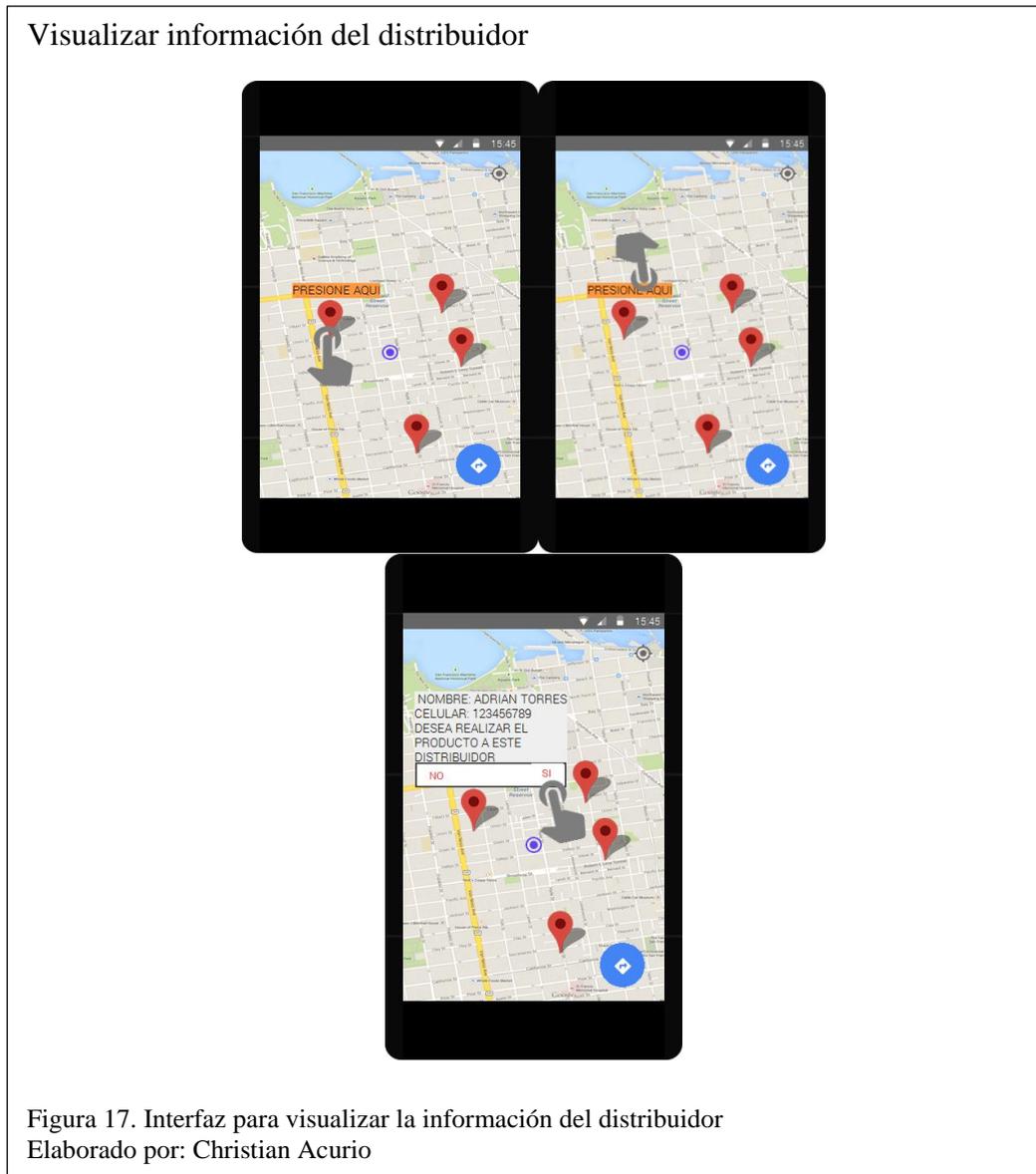


Figura 16. Interfaz para visualizar a los distribuidores en google maps
Elaborado por: Christian Acurio

La Figura 16 muestra un mapa de la ciudad de Quito, en la cual se muestra automáticamente la ubicación actual del cliente y también todos los distribuidores disponibles respecto a su ubicación.

3.2.8.2.3 Interface visualizar información del distribuidor.



Dentro de la Figura 17 se muestra las interfaces y las funciones que el usuario tiene que realizar para visualizar la información del distribuidor, para lograr esto el cliente tiene que presionar en la marca seleccionada, la cual visualizará un mensaje indicando que tiene que ser presionado, al realizar esta acción desplegará una pantalla que describirá el nombre y número de teléfono del distribuidor junto con un mensaje que informa si desea realizar el pedido.

3.2.8.2.4 Interfaz pedidos distribuidor.



La Figura 18 se observa la función del botón SI, el cual despliega la mensajería de texto con un mensaje predeterminado, el cual solicitará información (nombre, dirección) perteneciente al usuario, esta información permitirá al distribuidor ubicar el punto de entrega del producto.

Capítulo 3

Construcción y pruebas

En el capítulo 3 se explica todas las tecnologías, procesos y arquitecturas que estuvieron presentes para la creación de la aplicación móvil, también se detalla las pruebas de funcionamiento de la aplicación, las cuales permiten apreciar todos los requerimientos y funcionalidades detalladas por los usuarios.

4.1 Tecnologías y arquitectura de la aplicación móvil.

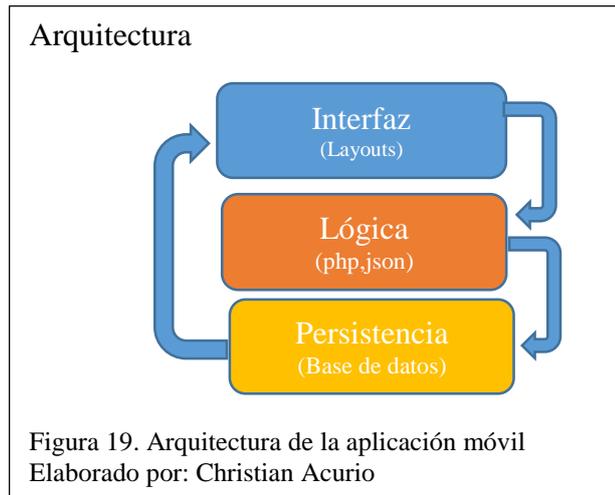
Se explica las tecnologías y arquitecturas que fueron utilizadas para estructurar la aplicación móvil.

La aplicación móvil funciona a través de un web services, el cual se encuentra dentro de una máquina virtual en la plataforma de Windows Azure. Esta aloja el servidor XAMPP que contiene la base de datos Mysql y los archivos PHP, los cuales permiten a la aplicación móvil interactuar con la base de datos de forma remota.

4.1.1 Arquitectura de la aplicación móvil.

La arquitectura de la aplicación móvil está compuesta por tres capas, cada una de estas cumplen una función diferente para lograr el funcionamiento correcto de los distintos procesos que involucra a la interfaz, el proceso lógico y la persistencia de la base de datos.

La capa interfaz contiene un fichero llamada layouts, el cual contiene todas las interfaces de la aplicación móvil. La capa lógica se encarga de todos los procesos lógicos (funciones y clases). La capa persistencia es el proceso que se encarga de la base de datos.



4.1.1.1 *Interfaz.*

La capa interfaz es la encargada de mostrar las interfaces de la aplicación, las cuales permiten a los usuario interactuar con los diferentes diseños y procesos que cada uno de estos contiene.

En la Figura 20 se muestra un ejemplo de una interfaz gráfica con diferentes funcionalidades.



4.1.1.2 Lógica.

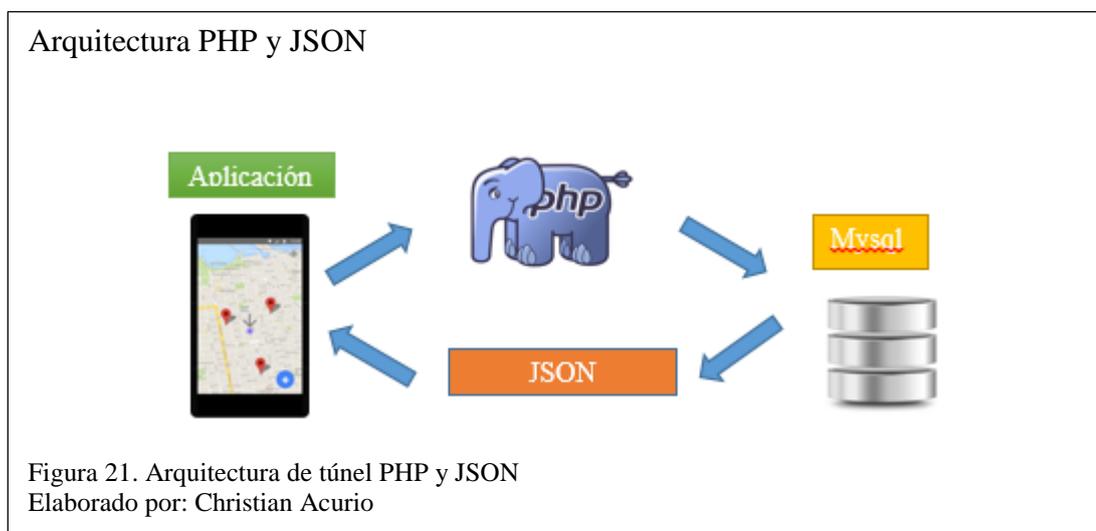
Esta capa contiene todo el funcionamiento lógico de la aplicación móvil, la parte lógica abarca cada una de las clases, las funciones y métodos, los cuales se encargan de procesar los datos ingresados en los diferentes formularios o consultas de datos, esto depende de las acciones de los usuarios.

4.1.1.3 Persistencia.

La capa persistencia se encarga de la gestión que realiza la base de datos, a una solicitud de información desde una aplicación móvil, estas solicitudes pueden involucrar actualizar, eliminar, consultar e ingresar información a las base de datos.

En la actualidad no existe un método que permita realizar una conexión de forma directa desde una aplicación móvil a una base de datos, por lo que se ha tomado las tecnologías PHP y JSON para realizarla, ya que son las más utilizadas por desarrolladores.

PHP y JSON sirve de conducto para que la aplicación móvil pueda interactuar junto con la base de datos, la cual fue levantada en Mysql.



En la Figura 21 se puede apreciar el funcionamiento de la estructura de PHP y JSON junto con la aplicación móvil y la base de datos Mysql para lograr el funcionamiento de esta combinación, la aplicación debe tener en su parte lógica el método HttpURLConnection, el cual permite que la aplicación pueda tener comunicación con un archivo PHP, dicho archivo debe contener en su código una línea de comandos SQL, que dará como resultado una consulta, inserción, actualización o eliminación de información en la base de datos, en caso de que la aplicación móvil este en espera de información solicitada, esta es entregada en formato JSON, este es interpretado por la aplicación según las necesidades del desarrollador.

4.1.2 Entorno de desarrollo Android Studio.

Para desarrollar la parte lógica de la aplicación se utilizó la versión gratuita 2.3.2 de Android Studio. Esta versión gratuita contiene todas las herramientas y características necesarias para el desarrollo de la aplicación por lo que no hizo falta usar la versión paga.

4.1.3 Windows Azure.

En la plataforma de Windows azure se integró una máquina virtual con sistema operativo Windows server 2012, en donde se instaló el servidor apache XAMPP y se integró la base de datos que maneja la información de los distribuidores que están registrados y los archivos PHP que gestionan las solicitudes de la aplicación.

4.2 Codificación

El desarrollo de la aplicación móvil se realizó siguiendo la metodología XP, desde el descubrimiento de las necesidades de los usuarios, hasta establecer las interfaces finales de esta, esto se realizó utilizando el método recomendado por la metodología

en la que explica que se debe realizar pruebas en cada proceso de desarrollo de la aplicación móvil.

A continuación se explica el orden que se siguió para el desarrollo de los módulos de la aplicación móvil.

Modulo distribuidor

Registro de un nuevo distribuidor

Inicio sesión distribuidor

Envío de recurrente de coordenadas

Actualizar contraseña distribuidor

Modulo Cliente

Buscar distribuidores en el mapa

4.2.1 Clases y métodos de la aplicación.

En este punto se explica las interfaces, clases y métodos que manejan en común la arquitectura de tres capas (interfaz, lógica y persistencia) dentro de la aplicación móvil.

4.2.1.1 Registro de un nuevo distribuidor.

El funcionamiento lógico del módulo de registro de un nuevo distribuidor empieza con el archivo Registro_Distribuidor.java, este archivo contiene el método ingresarDatosDisGet(), su funcionamiento es recoger los datos ingresados por el usuario en los campos de texto de la interfaz, para después enviar por medio de una petición GET al archivo ingresaDatosDis.php, el cual recibe los datos para interpretarlos mediante una línea de código que al ejecutar realiza un insert SQL en la base de datos.

Capa interfaz

Tabla 11. Interfaz activity_registro_distribuidor

Archivo	Método	Descripción
activity_registro_distribuidor.xml	Ingreso de un nuevo registro de un distribuidor	Interfaz que permite registrar los datos de un nuevo distribuidor en la base de datos.

Nota: Esta tabla contiene la actividad con su respectiva descripción.

Elaborado por: Christian Acurio

Capa lógica

Tabla 12. Insertar de datos distribuidor

Archivo	Método	Descripción
Registro_distribuidor.java	onCreate()	Es el primer método que inicializa al ejecutar la aplicación móvil.
	ingresarDatosDisGet()	Este método maneja los datos ingresados en los campos de texto que se encuentran en la interfaz de la aplicación, para lograr la inserción a la base de datos, por medio de una petición URL, la cual direcciona al archivo ingresaDatosDis.php.
	consultarEmailDatosVGet()	Método que consulta la existencia del correo ingresado por el usuario.
	validarDatos()	Método que ejecuta la validación de los campos y el método ingresarDatosDisGet() para finalizar la inserción.

Nota: Esta tabla contiene la clase con su respectiva descripción.

Elaborado por: Christian Acurio

Capa persistencia

Tabla 13. Ingreso de distribuidores a la base de datos

Archivo	Método	Descripción
ingresaDatosDis.php	mysqli_query()	Permite ejecutar el insert SQL con la información recibida desde la interfaz ,a la base de datos

Nota: Esta tabla contiene el archivo PHP con su respectiva descripción.

Elaborado por: Christian Acurio

4.2.1.2 Inicio sesión distribuidor.

El inicio de sesión de un distribuidor registrado en la base de datos, tiene como objetivo la autenticación de un distribuidor, este proceso utiliza el archivo login.php; el cual ejecuta una consulta SQL a la base de datos, el usuario logra interactuar por medio de la interfaz activity_login.xml.

Capa Interfaz

Tabla 14. Activity_login.xml

Archivo	Método	Descripción
activity_login.xml	Inicio de sesión distribuidor	Interfaz de inicio de sesión de un usuario distribuidor

Nota: Esta tabla contiene la actividad con su respectiva descripción.

Elaborado por: Christian Acurio

Capa lógica

Tabla 15. Inicio sesión usuario distribuidor

Archivo	Método	Descripción
login.java	onCreate()	Es el primer método que se inicializa al ejecutar la aplicación móvil.
	loginDatosGet()	Es el método que gestiona la petición de datos por medio de una URL, los cuales son recibidos en formato JSON desde el archivo login.php.
	consultarEmailDatosGet()	Método que permite validar la existencia del

		correo electrónico ingresado
	validarDatos()	Método que ejecuta la validación de los campos junto con todos métodos que permiten la funcionalidad de la aplicación.

Nota: Esta tabla contiene la clase con su respectiva descripción.

Elaborado por: Christian Acurio

Capa persistencia

Tabla 16. Consulta de usuario distribuidor para inicio de sesión

Archivo	Método	Descripción
login.php	json_encode()	Método que envía a la aplicación el resultado de la consulta en formato JSON.

Nota: Esta tabla contiene el archivo PHP con su respectiva descripción.

Elaborado por: Christian Acurio

4.2.1.3 *Envió recurrente de coordenadas.*

El envío recurrente de las coordenadas del dispositivo anfitrión a una base de datos remota, entra en funcionamiento cuando el usuario distribuidor inicia sesión con éxito dentro del módulo de la aplicación móvil.

Capa interfaz

Tabla 17. Login_Secun.xml

Archivo	Método	Descripción
Login_Secun.xml	Envió recurrente de coordenadas	Pantalla que indica que se está enviando las coordenadas del dispositivo anfitrión a una base de datos remota

Nota: Esta tabla contiene la actividad con su respectiva descripción.

Elaborado por: Christian Acurio

Capa lógica

Tabla 18. Consulta para inicio sesión distribuidor

Archivo	Método	Descripción
login.java	onCreate()	Es el primer método que se inicializa al ejecutar la aplicación móvil.
	ingresarCoordenadasGet()	Método que gestiona la inserción de las coordenadas recibidas por la clase secundaria locationListerner hasta una URL, junto con el archivo ingresar.php.
	actualizarCoordenadasGet()	Método que gestiona la actualización de las coordenadas por medio de una URL, junto con el archivo Actualizar_Coordenadas.php.
	Localización	Clase secundaria que permite hacer uso de diferentes métodos para la obtención de las coordenadas que el GPS anfitrión produce constantemente.
	validarDatos()	Método que ejecuta la validación de los campos junto con todos métodos que permiten la funcionalidad de la aplicación.
	onLocationChanged()	Este método se ejecuta cada vez que el GPS recibe nuevas coordenadas, debido a la detección de un cambio de ubicación.

Nota: Esta tabla contiene la clase con su respectiva descripción.

Elaborado por: Christian Acurio

Capa persistencia

Tabla 19. Actualizar coordenadas

Archivo	Método	Descripción
Actualizar_Coordenadas.php	mysqli_query()	Permite ejecutar el insert SQL con la información recibida desde la interfaz ,a la base de datos

Nota: Esta tabla contiene el archivo PHP con su respectiva descripción.

Elaborado por: Christian Acurio

4.2.1.4 Actualizar contraseña distribuidor.

El módulo actualizar contraseña distribuidor, utiliza el archivo actualizar_password.php. Este archivo ejecuta una actualización SQL a la base de datos, el usuario podrá cambiar la contraseña en caso de olvido de esta por medio de la interfaz activity_actualizar_passw.xml.

Capa interfaz

Tabla 20. activity_actualizar_passw.xml

Archivo	Método	Descripción
activity_actualizar_passw.xml	Actualizar la contraseña de inicio sesión distribuidor	Interfaz que permite al usuario cambiar la contraseña, cuando este lo requiere

Nota: Esta tabla contiene la actividad con su respectiva descripción.

Elaborado por: Christian Acurio

Capa lógica

Tabla 21. Cambiar contraseña distribuidores

Archivo	Método	Descripción
actualizar_passw.java	onCreate()	Es el primer método que se inicializa al ejecutar la aplicación móvil.
	actualizarPass()	Método que gestiona la actualización de la contraseña del usuario distribuidor por medio de una URL, junto con el

		archivo actualizar_password.php.
	consultarEmailDatosVGet()	Método que permite obtener el correo electrónico de un distribuidor por medio de una consulta SQL, lo que permitirá validar si el correo electrónico ingresado existe, desde la interfaz activity_actualizar_passw.xml.
	validarDatos()	Método que ejecuta la validación de los campos junto con todos métodos que permiten la funcionalidad de la aplicación.

Nota: Esta tabla contiene la clase con su respectiva descripción.

Elaborado por: Christian Acurio

Capa persistencia

Tabla 22. Actualizar contraseña distribuidor

Archivo	Método	Descripción
actualizar_password.php	mysqli_query()	Permite ejecutar el insert SQL con la información recibida desde la interfaz ,a la base de datos

Nota: Esta tabla contiene el archivo PHP con su respectiva descripción.

Elaborado por: Christian Acurio

4.2.1.5 *Buscar distribuidores en el mapa.*

Para realizar la búsqueda de distribuidores dentro del mapa de la ciudad de Quito, utilizamos el archivo obtener_coordenadas.php, el cual genera la consulta SQL a la base de datos, esto permite que se presenten los distribuidores dentro del mapa de la aplicación móvil, esta información es enviada a la aplicación en formato JSON y es interpretada por la interfaz activity_maps.xml.

Capa interfaz

Tabla 23. activity_maps.xml

Archivo	Método	Descripción
activity_maps.xml	Búsqueda de distribuidores móviles y fijos	Interfaz que muestra al cliente los distribuidores de gas doméstico, dentro de un mapa

Nota: Esta tabla contiene la actividad con su respectiva descripción.

Elaborado por: Christian Acurio

Capa lógica

Tabla 24. Buscar distribuidores en el mapa

Archivo	Método	Descripción
MapsActivity.java	onCreate()	Es el primer método que se inicializa al ejecutar la aplicación móvil.
	onMapReady()	Método que ejecuta todos los componentes que el mapa va a contener.
	RetrieveTask	Clase secundaria que contiene el proceso del método doInBackground().
	doInBackground()	Método que permite recuperar las ubicaciones almacenadas en la base de datos remota por medio de una URL
	ParserTask	Clase secundaria que maneja un hilo en el que almacena los datos JSON recuperados del servidor MySQL
	MarkerJSONParser	Clase que separa los datos recibidos en formato JSON, para después poder utilizarlos
	onPostExecute()	Este método almacena las coordenadas que son parseadas dentro de un array para poder mostrarlos en el mapa en conjunto.

	Localización	Clase secundaria que permite hacer uso de diferentes métodos para la obtención de las coordenadas que el GPS anfitrión produce constantemente.
	onLocationChanged()	Este método tiene como función actualizar cada determinado tiempo el proceso que se encuentre dentro de la clase secundaria Localización.

Nota: Esta tabla contiene la clase con su respectiva descripción.

Elaborado por: Christian Acurio

Capa persistencia

Tabla 25. Consultar coordenadas

Archivo	Método	Descripción
obtener_coordenadas.php	json_encode()	Método que envía a la aplicación el resultado de la consulta en formato JSON.

Nota: Esta tabla contiene el archivo PHP con su respectiva descripción.

Elaborado por: Christian Acurio

4.3 Pruebas de la aplicación móvil

Se realizó las pruebas de la aplicación móvil pensando en el rendimiento y el funcionamiento de los diferentes módulos creados con el objetivo de dar una buena experiencia a los diferentes usuarios que van hacer uso de la aplicación, también se presentó con detalles el funcionamiento de la aplicación en distintos tipos de dispositivos móviles, logrando con esto probar que la aplicación funciona en tecnologías que se encuentran en auge en la actualidad.

4.3.1 Pruebas funcionamiento aplicación móvil.

El desempeño de la aplicación móvil, se realizó haciendo uso de dos dispositivos móviles, cada uno de estos, con diferentes características, tanto el hardware como en

el software, esto permitió comprobar el funcionamiento de la aplicación móvil. Las características de los dispositivos se detallan a continuación.

SAMSUNG J1

Sistema operativo: Android 5.1.1

Memoria total del dispositivo: 8 GB

Procesador: 1.2Ghz Quad-Core

Memoria RAM: 1 GB

SAMSUNG J7

Sistema operativo: Android 6.0.1

Memoria total del dispositivo: 16 GB

Procesador: 1.6 Ghz Quad-Core

Memoria RAM: 2 GB

Como se puede apreciar en las características el SAMSUNG J1 usa la API 22, que alberga internamente la versión del sistema operativo Android 5.1.1 y el SAMSUNG J7 usa la API 23, alberga internamente la versión del sistema operativo Android 6.0.1, se marca esta característica porque en la nueva versión la API 23 que Android hizo pública hasta la fecha, se diferencia de las anteriores, en la forma de validar los permisos que los desarrolladores tienen que integrar en la construcción de las aplicaciones que hacen uso de las funciones internas de los dispositivos móviles (GPS, mensajería de texto).

La aplicación móvil funciona plenamente en dispositivos antiguos como en los actuales.

4.3.1.1 Modulo distribuidor.

4.3.1.1.1 Prueba de registro de un nuevo distribuidor.

El usuario que ocupe el rol de distribuidor y desee hacer uso de la aplicación podrá realizar su registro con el modulo que fue creado con este objetivo, para esto el usuario tendrá que registrar sus nombres, apellidos, teléfono, correo electrónico y contraseña. A continuación se presenta el funcionamiento de dicho modulo.



4.3.1.1.2 Prueba de inicio sesión distribuidor.

Los distribuidores que se encuentran registrados en la aplicación pueden iniciar sesión, con el objetivo compartir sus coordenadas actuales cada determinado tiempo de manera automática, en las pruebas se pudo apreciar que este proceso depende mucho de la calidad de conexión a internet que el distribuidor dispone, ya que si existe una señal de mala calidad, las coordenadas serán enviadas con el intervalo de tiempo que tome mejorar la calidad de la señal del dispositivo.

Prueba inicio sesión distribuidor

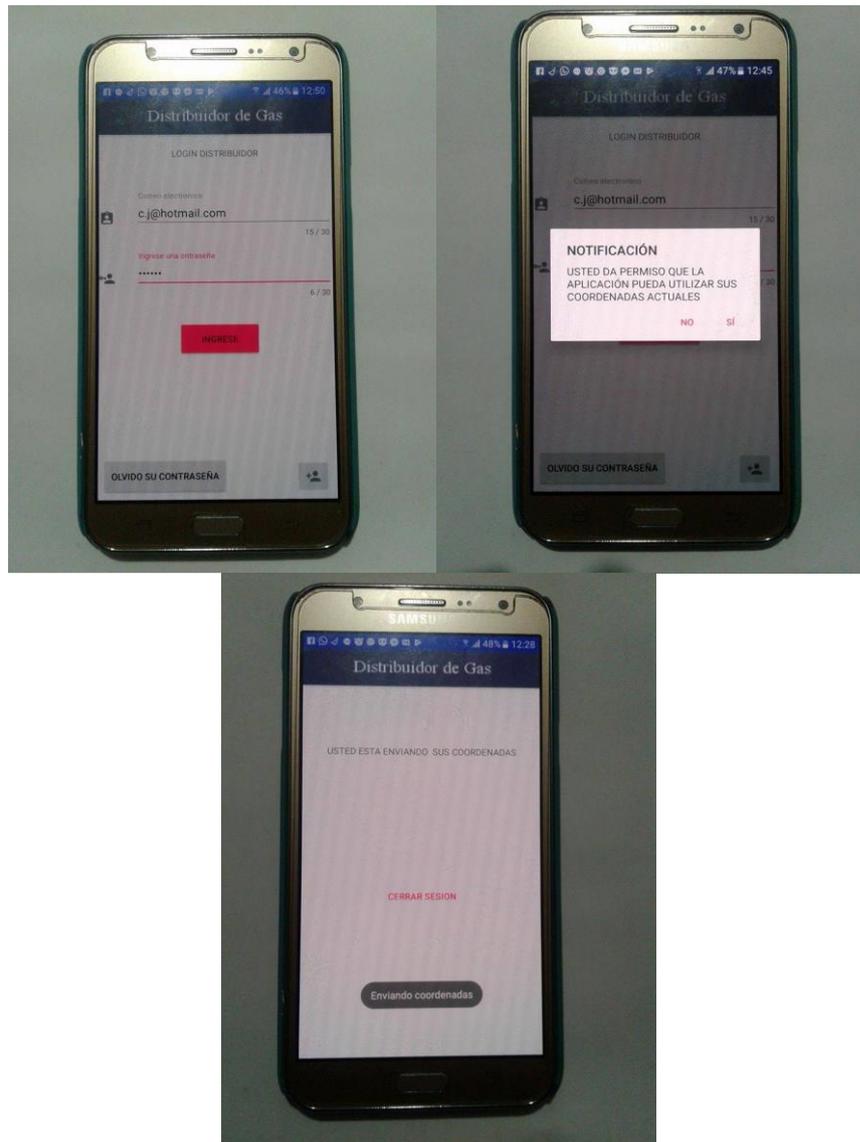


Figura 23. Prueba para iniciar sesión distribuidor
Elaborado por: Christian Acurio

4.3.1.1.3 Prueba actualizar contraseña distribuidor.

Los distribuidores que se encuentren registrados en la aplicación tienen la opción de cambiar la contraseña ingresada en el registro, esto es de mucha ayuda en el caso de olvido de la misma. Para realizar este proceso el usuario tiene que ingresar el correo electrónico que se encuentra registrado, si el correo existe el usuario podrá ingresar la

nueva contraseña junto con su verificación, si los datos son correctos el cambio se realizara automáticamente, como se muestra en la siguiente imagen.



4.3.1.2 *Modulo cliente.*

4.3.1.2.1 *Prueba de búsqueda de distribuidores en el mapa de Quito.*

La aplicación permite a los clientes tener la opción de buscar a los distribuidores de gas doméstico móviles y locales que se encuentren dentro de la ciudad de Quito más cercanos a su ubicación, el tiempo de respuesta del teléfono SAMSUNG J1 es de 1500 ms y el tiempo de respuesta del SAMSUNG J7 es de 1200 ms en esta prueba se puede apreciar que el tiempo de respuesta del SAMSUNG J7 es más corto por motivo a que es un teléfono de gama alta, también se tiene que tomar en cuenta que el tiempo de respuesta depende mucho de la calidad de conexión de internet que se maneja.

Las siguientes figuras muestran el funcionamiento de la aplicación en los dispositivos mencionados.

Buscar en el mapa distribuidores

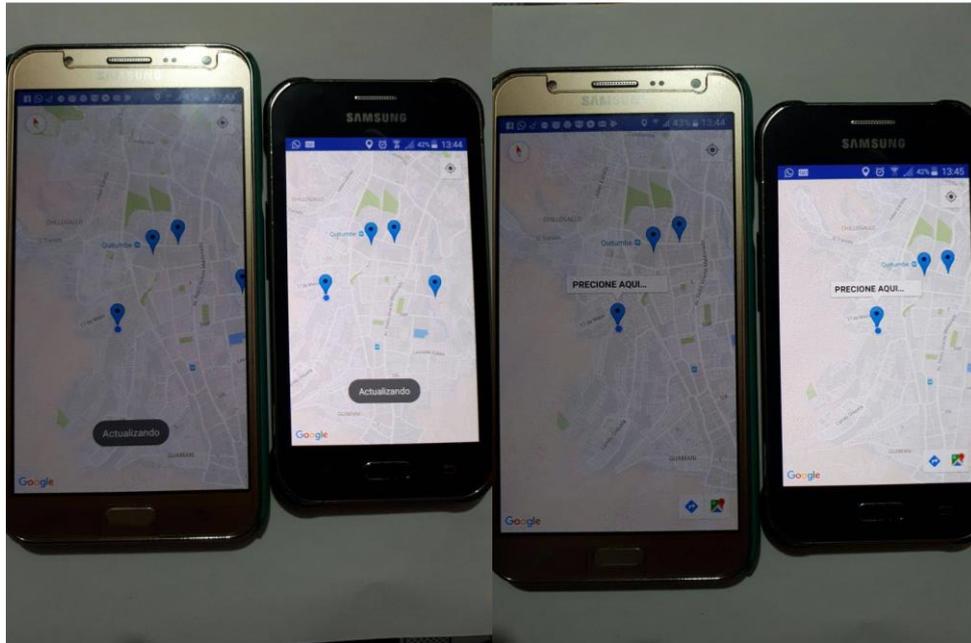


Figura 25. Prueba para buscar distribuidores de gas doméstico
Elaborado por: Christian Acurio

A continuación se muestra las pantallas con la información del distribuidor seleccionado por el cliente, en esta interfaz se puede apreciar el nombre y teléfono del distribuidor seleccionado. Como se puede apreciar en la Figura 25 el funcionamiento de este módulo es correcto en los dos dispositivos.

Prueba visualizar información distribuidor

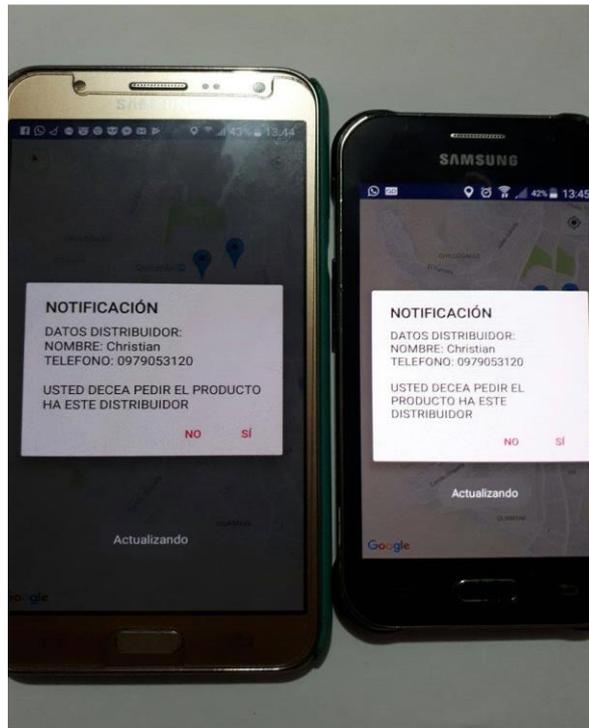


Figura 26. Prueba para visualizar la información del distribuidor seleccionado
Elaborado por: Christian Acurio

En la Figura 26 se muestra el funcionamiento del módulo que permite realizar pedidos al distribuidor seleccionado. El proceso para realizar el pedido es muy intuitivo y fácil de usar para los usuarios, ya que no se percibió ninguna dificultad por parte de las personas que hicieron uso de la aplicación. El envío de la solicitud del producto se realiza por medio de un mensaje de texto, lo cual se realizara automáticamente, siempre y cuando el usuario decida enviarlo, el envío de la solicitud se ejecuta desde la mensajería de texto.

Prueba envío de pedido de gas doméstico

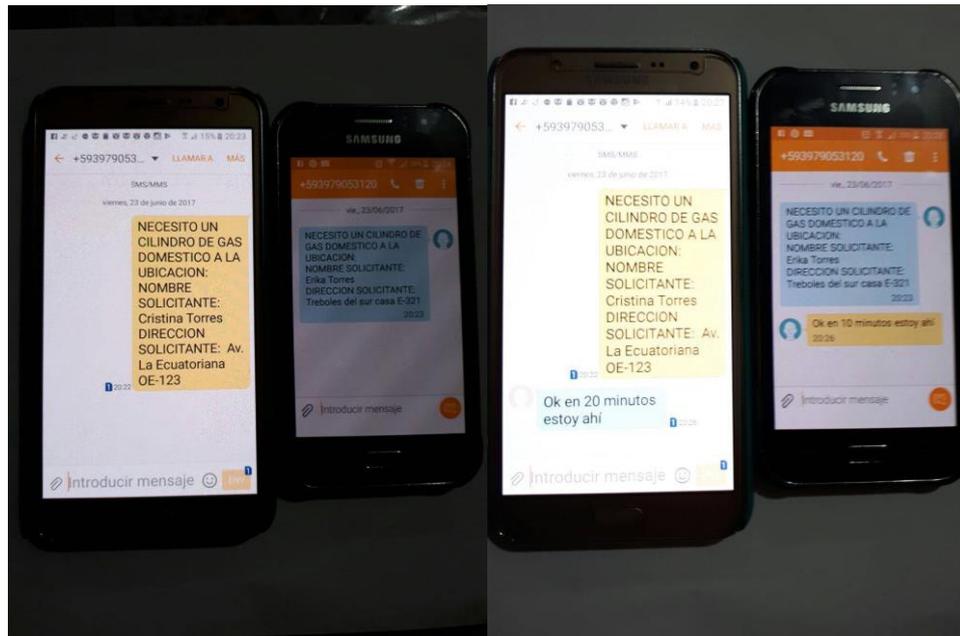


Figura 27. Prueba para enviar un pedido a un distribuidor
Elaborado por: Christian Acurio

Para el proceso anterior el usuario debe disponer de un paquete de mensajes de texto que permita enviar y recibir mensajes de texto a todas las operadoras disponibles en Quito.

Cabe recalcar que para el uso de la mayoría de las funciones de la aplicación es necesario disponer de un plan de datos o conexión WIFI, ya que sin estas no podrá hacer uso de los módulos de la aplicación móvil.

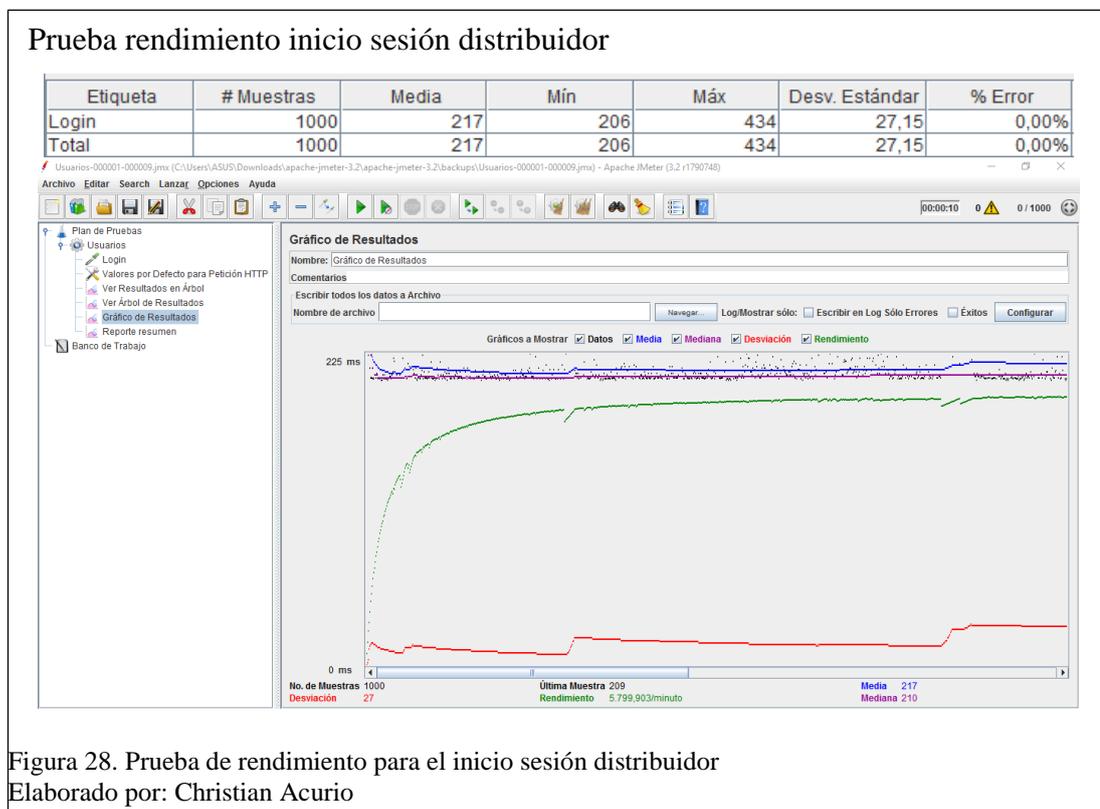
En las pruebas realizadas para la aplicación móvil, se pudo observar todas las características y funcionalidades presentes en la aplicación, Estas pruebas fueron realizadas en tiempo real, con dispositivos diferentes, cada uno con características distintas, lo que permitió observar la adaptación a diferentes ambientes de trabajo.

4.3.2 Pruebas de rendimiento aplicación móvil.

Las pruebas de rendimiento se realizaron utilizando la aplicación JMeter, se escogió esta aplicación por su fácil uso y por todas sus funciones que provee para realizar dichas pruebas.

4.3.2.1 Prueba 1.

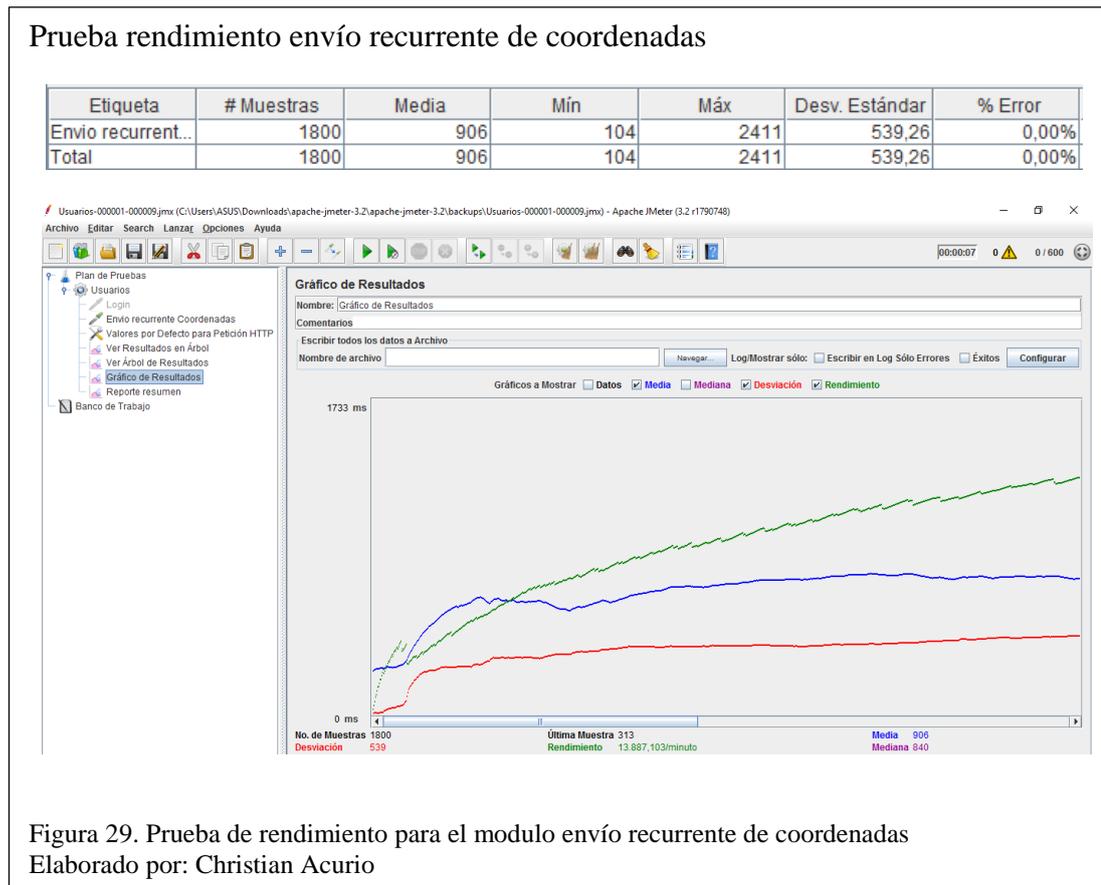
La prueba de rendimiento se enfocara en el módulo de inicio sesión de los distribuidores. Esta prueba se desarrolló con los siguientes parámetros, 1000 usuarios en un periodo de 10 segundos, realizarán una petición.



En la Figura 28 se puede apreciar que el rendimiento del módulo de inicio sesión distribuidor fue optima, ya que la aplicación no presento ningún porcentaje de error, el tiempo de entrega de cada petición se realizó en 217 milisegundos y el número total de peticiones fueron 1000.

4.3.2.2 Prueba 2.

Se pondrá a prueba la función de envío recurrente de coordenadas, la cual es realizada por los distribuidores que inician sesión en la aplicación móvil. La prueba se realizó con los siguientes datos, 600 usuarios en un periodo de 5 segundos, realizarán tres peticiones.

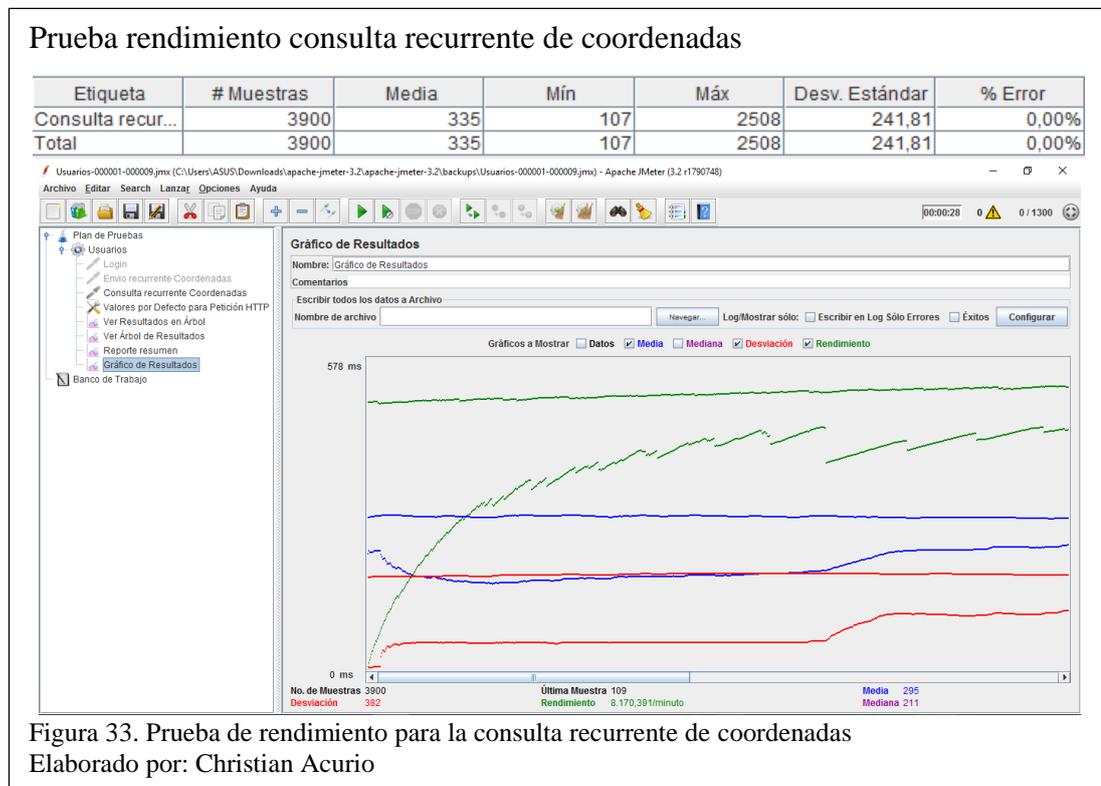


En la Figura 29 se puede apreciar que la aplicación no presento ningún error con la carga de peticiones con la que fue puesta a prueba, el tiempo entre cada petición se realizó en 906 milisegundos y el número total de peticiones fueron 1800.

4.3.2.3 Prueba 3.

Se pondrá a prueba el modulo que consulta recurrentemente de coordenadas al servidor, la cual es realizada por los clientes que hacen uso de la aplicación móvil. La

prueba se realizó con los siguientes datos, 1300 usuarios en un periodo de 10 segundos, realizarán tres peticiones.



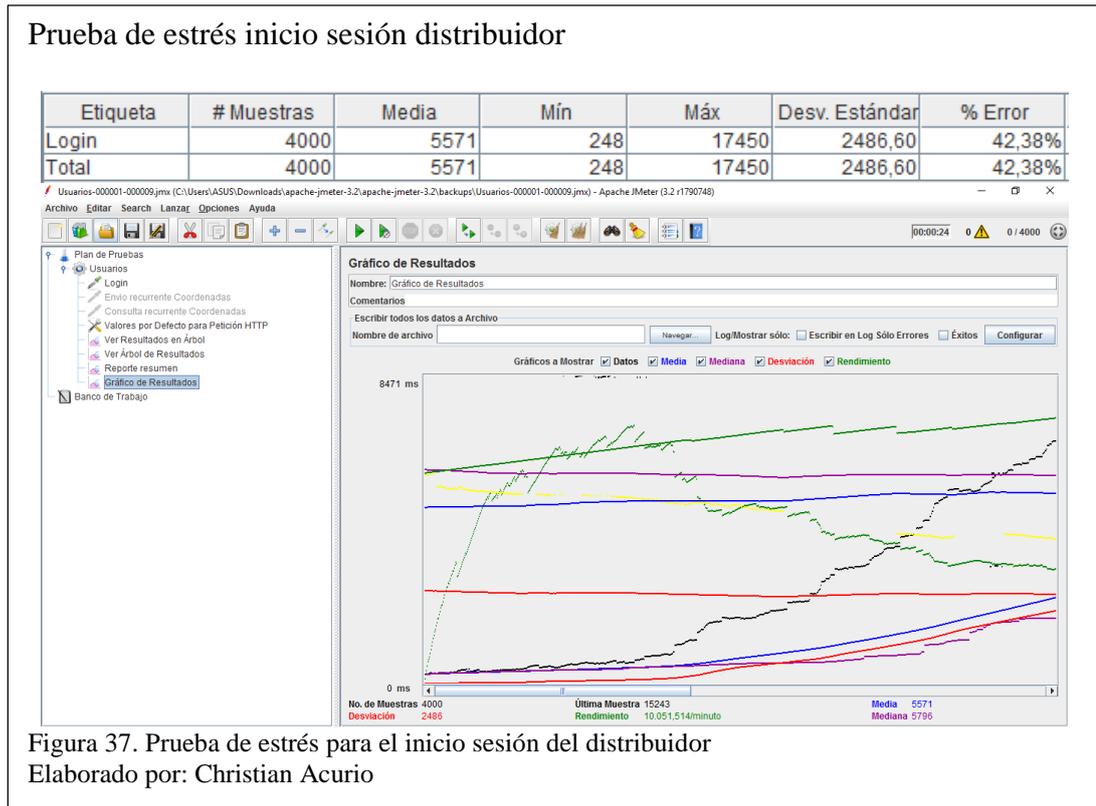
En la Figura 30 se muestra el resultado de la prueba de rendimiento al que fue sometida el módulo de consulta recurrente de coordenadas, en la que se pudo visualizar que el modulo no produjo ningún tipo de error, el tiempo de respuesta de cada petición se realizó en 335 milisegundos, el número total de peticiones fueron 3900.

4.3.3 Pruebas de Estrés Aplicación móvil.

Las pruebas de estrés se realizaron con el objetivo de probar la aplicación móvil con una carga de peticiones he usuarios que esta no pueda soportar y empiece a presentar errores

4.3.3.1 Prueba 1.

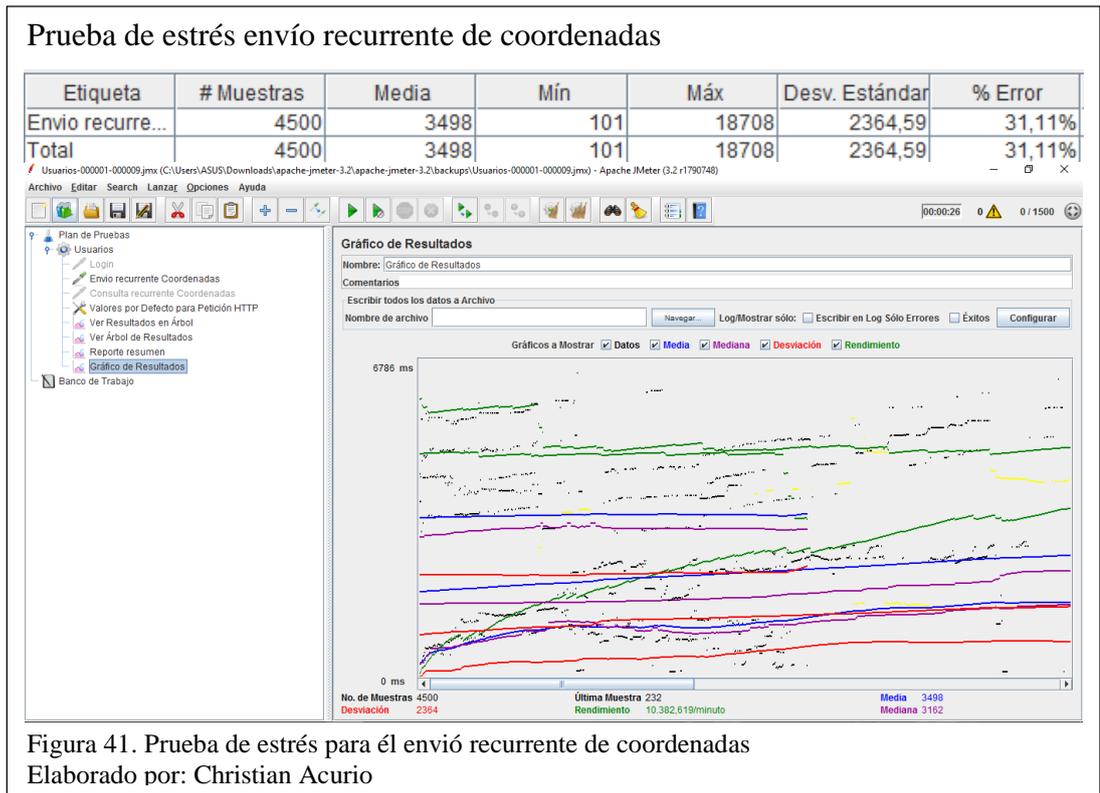
La prueba se realizó en el módulo de inicio sesión distribuidor, ya que este módulo es una de las partes de la aplicación en la que se presenta mayor número de peticiones. Los datos utilizados para la prueba son 4000 usuarios en un periodo de 10 segundos, realizarán 1 petición.



En la Figura 31 se puede observar que el modulo presenta el 42,38 por ciento de error con la carga con la que fue puesta a prueba. Esto nos permite conocer el límite de peticiones que dicho modulo no puede soportar.

4.3.3.2 Prueba 2.

La prueba se realizó en el módulo envío recurrente de coordenadas, por toda la carga de peticiones que soporta. Los datos utilizados para la prueba son 1500 usuarios en un periodo de 5 segundos, realizarán 3 peticiones.



En la Figura 32 se puede apreciar que el módulo de envío recurrente de coordenadas, la prueba se utilizó con un total de 4500 peticiones para dicho módulo, logrando presentar el 31,11 por ciento de error.

4.3.3.3 Prueba 3.

La prueba se realizó al módulo de consulta recurrente de coordenadas, por toda la carga de peticiones que soporta por los diferentes usuarios que hacen uso de la aplicación móvil. Los datos utilizados para la prueba son 3000 usuarios en un periodo de 10 segundos, realizarán 3 peticiones.

Prueba de estrés consulta recurrente de coordenadas

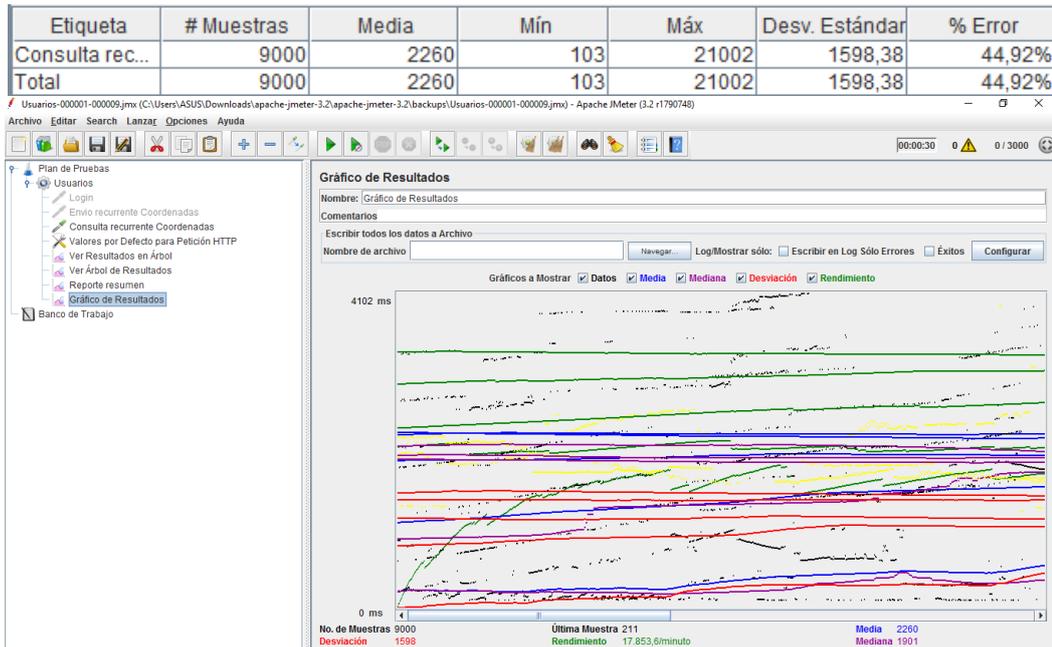


Figura 45. Prueba de estrés para la consulta recurrente de coordenadas
Elaborado por: Christian Acurio

En la Figura 33 se puede apreciar que para la prueba se utilizó con un total de 9000 peticiones, esta cantidad de peticiones logro que la aplicación presentara un 42,92 por ciento de error.

CONCLUSIONES

- El desarrollo de aplicaciones móviles o APP que funcionen en sistemas operativos Android, impuso un gran reto por todo el trabajo de investigación que se tuvo que realizar para poder integrar los requerimientos encontrados en base a la necesidad que viven las dos partes que involucran el proceso de solicitar y distribuir cilindros de gas doméstico en la ciudad de Quito, tomando en cuenta el tiempo que se dispuso para el desarrollo de la aplicación se optó por utilizar la estructura de la metodología XP, la cual permitió agilizar el desarrollo y la documentación de la misma.
- La construcción del módulo de localización fue la parte más complicada en desarrollar la aplicación móvil, ya que el objetivo de esta, no es solo poder visualizar los distribuidores de gas doméstico con locales fijos dentro de un mapa de la ciudad de Quito, sino también se debe visualizar la ubicación actual de los distribuidores que tienen un vehículo que les permite distribuir puerta a puerta el producto. Esto se logró resolver con el método `requestLocationUpdates`, el cual tiene como función actualizar cada determinado tiempo el proceso que se encuentre dentro de la clase secundaria `LocationListener`.
- El desarrollo del aplicativo con el motor base de datos Mysql junto con software Android Studio, resulto muy amigable, considerando que Android no dispone de librerías que permita una conexión directa con Mysql, por lo que se optó por realizar una conexión tipo de túnel que involucra las tecnologías PHP+JSON.
- Al finalizar el proyecto se pudo constatar que se cumplieron con todos los objetivos propuestos y gracias a la elaboración de la aplicación móvil la

ciudadanía podrá tener acceso a los servicios que la misma le brinda pudiendo conseguir un tanque de gas de manera fácil, rápida y precisa.

RECOMENDACIONES

- Se recomienda realizar un estudio para modificar la aplicación móvil y poder incluir a otras ciudades del Ecuador, ya que deben presentar el mismo problema que se apercibe en la ciudad de Quito.
- Es importante que la aplicación mantenga una conexión estable y permanente a internet, ya que si no se cumple con esto, no se podrá hacer uso de los diferentes módulos de la aplicación móvil que dependen de esto.
- Es necesario ejecutar la aplicación en dispositivos que tengan las versiones de API 22 y API 23, es decir dispositivos con sistema operativo Android versión 5.0 a la nueva versión 6.0.1, ya que los dispositivos móviles que vienen con estas versiones traen consigo todas la características necesarias para que la aplicación móvil funcione al cien por ciento.
- Se recomienda construir una aplicación web, con la cual se pueda administrar la base de datos, con el objetivo que los propios distribuidores fijos puedan registrarse o puedan ser registrados por una persona encargada (administrador), para posteriormente poder ser visualizados en la aplicación móvil por todos sus clientes.

REFERENCIAS

- Cáceres Díaz, I. F., & Larco Tapia, G. J. (2016). *Análisis, Diseño, Desarrollo e implementación de un sistema de georreferenciación de operadores económicos para la superintendencia de control del poder de mercado*. Quito.
- Cobo, Á., Gómez, P., Pérez, D., & Rocha, R. (2005). *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. España: Díaz de Santos.
- Deza Castillo, C. (23 de Mayo de 2017). *SlideShare*. Obtenido de SlideShare: <https://www.slideshare.net/CristhianDezaCastill/tema-android-studio>
- Fontela, C. (2012). *UML. Modelo de software para profesionales*. Barcelona: MARCOMBO, S.A.
- Gallegos Riera, F. A. (2012). *Mapa Virtual USFQ 3D: Community Aplicación Nativa*. Quito.
- Ribas Lequerica, J. (2003). *Web Services*. s.f.: Anaya Multimedia.
- Robledo Sacristán, C., & Robledo Fernández, D. (2012). *Programación en Android*. España.
- Sommerville, I. (2005). *Ingeniería de Software*. Madrid: Person Education S.A.