



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA, REDES Y  
COMUNICACIÓN DE DATOS**

**TESIS PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO ELECTRÓNICO**

**TEMA: DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN  
CON ANDROID PARA LA GESTIÓN REMOTA DE UN  
SMARTPHONE**

**AUTOR: TORRES RACINES, TITO DAVID**

**DIRECTOR: ING. ALULEMA, DARWIN**

**CODIRECTOR: ING. ROMERO, CARLOS**

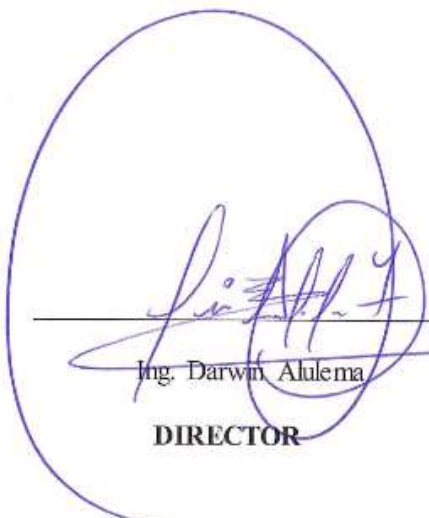
**SALGOLQUÍ**

**2015**

## CERTIFICACIÓN

Certificamos que el presente proyecto de grado titulado: “DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN CON ANDROID PARA LA GESTIÓN REMOTA DE UN SMARTPHONE”, fue desarrollado en su totalidad por el Señor, Tito David Torres Racines con CI: 1721752721, bajo nuestra dirección.

Atentamente,



Ing. Darwin Alulema  
**DIRECTOR**



Ing. Carlos Romero  
**CODIRECTOR**

## **AUTORÍA DE RESPONSABILIDAD**

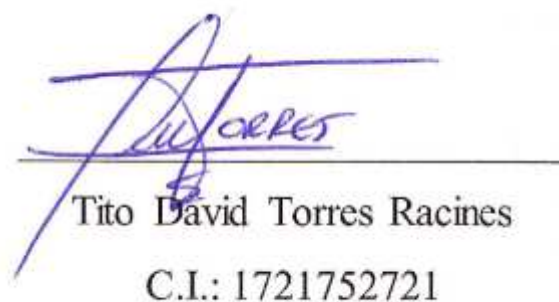
TITO DAVID TORRES RACINES

### **DECLARO QUE:**

El proyecto de grado denominado “DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN CON ANDROID PARA LA GESTIÓN REMOTA DE UN SMARTPHONE”, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan al pie, de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.




Tito David Torres Racines  
C.I.: 1721752721

Sangolquí, 19 de Marzo del 2015

## AUTORIZACIÓN

Yo, Tito David Torres Racines

Autorizo a la Universidad de las Fuerzas Armadas – ESPE, la publicación en la biblioteca virtual de la institución del proyecto de grado titulado: **“DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN CON ANDROID PARA LA GESTIÓN REMOTA DE UN SMARTPHONE”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.



Tito David Torres Racines  
C.I.: 1721752721

Sangolquí, 19 de Marzo del 2015

## **DEDICATORIA**

Con todo mi cariño y mi amor para las personas que hicieron todo en la vida para que yo pudiera lograr mis sueños, por motivarme y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Wilfrido y Anita

Tito David Torres Racines.

## AGRADECIMIENTO

Agradezco en primer lugar a Dios por permitirme disfrutar y vivir estos momentos que nunca se volverán a repetir por ser únicos y maravillosos.

Agradezco a mis padres por su apoyo que nunca termina, que se han sacrificado más que mí, en lograr esta meta, a mi padre Wilfrido Torres que ha sido un ejemplo de honestidad y de trabajo ineludible, a mi madre Anita Racines por darme un ejemplo de gran esmero y tenacidad en la vida.

También agradezco a mis hermanos Daniel Torres Y Sofía Torres los cuales ayudaron dándome su aliento, su ánimo a lo largo de su vida.

Agradezco a los ingenieros que supieron impartirme los conocimientos que hoy poseo pero también por su parte la ejemplificación de que siempre se seguirá estudiando adquiriendo nuevos conocimientos.

Finalmente agradezco pero no menos importante a Hilda Navarrete, Gonzalo Valencia, Marisol Villaruel, Roberto Chiliza, Lorena Pazmiño, Fabricio Ramos, Lady Mariana Torres, Fabiola Torres, Alfredo Chiliza, Hugo Rodríguez que siempre creyeron en mí para lograr mi meta.

## ÍNDICE DE CONTENIDOS

<b>CERTIFICACIÓN .....</b>	<b>ii</b>
<b>AUTORÍA DE RESPONSABILIDAD .....</b>	<b>iii</b>
<b>AUTORIZACIÓN .....</b>	<b>iv</b>
<b>DEDICATORIA .....</b>	<b>v</b>
<b>AGRADECIMIENTO.....</b>	<b>vi</b>
<b>ÍNDICE DE CONTENIDOS .....</b>	<b>vii</b>
<b>ÍNDICE DE FIGURAS .....</b>	<b>xi</b>
<b>ÍNDICE DE TABLAS .....</b>	<b>xiv</b>
<b>RESUMEN .....</b>	<b>xv</b>
<b>ABSTRACT .....</b>	<b>xvi</b>
<b>CAPÍTULO I .....</b>	<b>1</b>
<b>1. INTRODUCCION .....</b>	<b>1</b>
<b>1.2 DEFINICIÓN DEL PROBLEMA .....</b>	<b>3</b>
<b>1.3 JUSTIFICACIÓN E IMPORTANCIA .....</b>	<b>3</b>

1.4	ALCANCE DEL PROYECTO.....	5
1.5	OBJETIVOS .....	6
<b>CAPÍTULO II.....</b>		<b>7</b>
<b>2. ESTUDIO DE LA ARQUITECTURA DE CLIENTE SERVIDOR</b>		<b>7</b>
2.1	TECNOLOGÍAS PARA EL CLIENTE .....	7
2.1.1	Arquitectura de Android .....	8
2.1.2	Versiones de Plataforma Android.....	12
2.1.3	Arquitectura de Aplicaciones Android .....	19
2.1.4	Desarrollador de Aplicaciones Android .....	25
2.2	TECNOLOGÍAS PARA EL SERVIDOR.....	28
2.2.1	Servidor Socket.....	30
2.2.2	Interfaz de Usuario vía Web.....	33
2.2.2.1	Lenguaje de Programación del lado del Servidor (PHP).....	34
2.2.2.2	Lenguaje de Programación del lado del Cliente (JavaScript) .....	34
2.2.2.3	Tecnología Ajax. ....	34
2.2.2.4	Api de Google Maps.....	35
2.2.3	Gestión de Base de Datos.....	35
2.2.3.1	Base de Datos Relacional.....	36
2.2.4	Servidor Vsftpd.....	36



<b>CAPÍTULO III .....</b>	<b>38</b>
<b>3. DISEÑO DEL SISTEMA.....</b>	<b>38</b>
3.1 SISTEMA REQUERIDO .....	38
3.2 DIAGRAMAS .....	39
3.2.1 Diagrama de Clases Cliente.....	39
3.2.2 Diagrama Entidad Relación de Base de Datos Servidor.....	42
3.2.3 Diagrama de Clases Servidor Socket Registro y Gps.....	43
3.2.4 Diagramas de casos de uso .....	44
3.2.5 Diagrama de Flujo de Servidor Socket Registro .....	46
3.2.6 Diagrama de Flujo de Servidor Socket Gps.....	48
3.2.7 Diagrama de Flujo de Formulario Registro .....	49
3.2.8 Diagrama de Flujo de Servicio Ftp Cliente .....	51
3.2.9 Diagrama de Flujo de Servicio Gps Cliente .....	53
3.3 DISEÑO DE LA APLICACIÓN PARA EL CLIENTE.....	55
3.4 DISEÑO DE LA APLICACIÓN PARA EL SERVIDOR .....	57
3.4.1 DISEÑO DE LA INTERFAZ WEB .....	57
<b>CAPÍTULO IV.....</b>	<b>59</b>
<b>4. IMPLEMENTACION Y PRUEBAS .....</b>	<b>59</b>

4.1	ESCENARIO RED DE AREA LOCAL .....	59
4.2	ESCENARIO RED DE AREA EXTENDIDA (WAN) .....	67
4.3	ANÁLISIS DE RESULTADOS .....	73
<b>CAPÍTULO V .....</b>		<b>78</b>
<b>5.</b>	<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>78</b>
5.1	CONCLUSIONES .....	78
5.2	RECOMENDACIONES.....	79
<b>REFERENCIA BIBLIOGRÁFICA.....</b>		<b>81</b>

## ÍNDICE DE FIGURAS

<b>Figura. 1.</b> Capas en Sistema Operativo Android.....	10
<b>Figura. 2.</b> Jerarquia de Actividades .....	20
<b>Figura. 3.</b> Ciclo de Vida de una Aplicación. ....	21
<b>Figura. 4.</b> Ciclo de Vida de un Servicio. ....	23
<b>Figura. 5.</b> Función de Capa de Transporte. ....	30
<b>Figura. 6.</b> Protocolo TCP.....	31
<b>Figura. 7.</b> Comunicación Server Socket.....	32
<b>Figura. 8.</b> Visualización del Mapa Google Maps V3.....	35
<b>Figura. 9.</b> Sistema Requerido .....	38
<b>Figura. 10.</b> Diagrama UML de Aplicación Cliente .....	40
<b>Figura. 11.</b> Diagrama UML de Base de Datos .....	42
<b>Figura. 12.</b> Diagrama UML de Servidor Socket GPS y Registro.....	43
<b>Figura. 13.</b> Diagrama Caso de Uso Registro .....	44
<b>Figura. 14.</b> Diagrama Caso de Uso GPS .....	45
<b>Figura. 15.</b> Diagrama Caso de Uso Envió de Archivos.....	45
<b>Figura. 16.</b> Diagrama Caso de Uso Consulta vía Web.....	46
<b>Figura. 17.</b> Diagrama de Flujo Servidor Socket Registro .....	48
<b>Figura. 18.</b> Diagrama de Flujo Servidor Socket GPS.....	49
<b>Figura. 19.</b> Diagrama de Flujo de Formulario Registro .....	50
<b>Figura. 20.</b> Diagrama de Flujo de Servicio FTP Cliente .....	53

<b>Figura. 21.</b> Diagrama de Flujo de Servicio GPS Cliente.....	54
<b>Figura. 22.</b> Áreas de la Interfaz Android.....	55
<b>Figura. 23.</b> Vista de Presentación de La Aplicación Android.....	56
<b>Figura. 24.</b> Esquema de Interfaz Web .....	57
<b>Figura. 25.</b> Página Web de historial de ubicación.....	58
<b>Figura. 26.</b> Topología de la red LAN .....	59
<b>Figura. 27.</b> Topología de red LAN con Direccionamiento.....	60
<b>Figura. 28.</b> Prueba Servidor Socket Registro LAN .....	63
<b>Figura. 29.</b> Carpeta generada para Usuario .....	64
<b>Figura. 30.</b> Prueba Servidor Socket GPS LAN .....	64
<b>Figura. 31.</b> Prueba Servidor MySQL.....	65
<b>Figura. 32.</b> Prueba Servidor Vsftpd.....	66
<b>Figura. 33.</b> Topología de la red WAN.....	67
<b>Figura. 34.</b> Topología de red WAN con Direccionamiento .....	67
<b>Figura. 35.</b> Prueba Servidor Socket Registro WAN.....	68
<b>Figura. 36.</b> Carpeta generada para Usuario .....	69
<b>Figura. 37.</b> Prueba Servidor Socket GPS WAN.....	69
<b>Figura. 38.</b> Prueba Servidor MySQL en red WAN .....	70
<b>Figura. 39.</b> Prueba Servidor Vsftpd.....	71
<b>Figura. 40.</b> Captura Wireshark Servicio FTP.....	72
<b>Figura. 41.</b> Consumo de Batería de la aplicación.....	73
<b>Figura. 42.</b> Memoria utilizada por el Usuario .....	74
<b>Figura. 43.</b> Uso de datos Smartphone.....	74

**Figura. 44.** Consumo de Batería..... 77

**Figura. 45.** Consumo de la memoria RAM ..... 77

## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Especificaciones de Versiones API.....	15
<b>Tabla 2.</b> Características de las Versiones API.....	16
<b>Tabla 3.</b> Descripción de Vida de una Aplicación.....	22
<b>Tabla 4.</b> Ventajas de Sistema Operativo Centos.....	29
<b>Tabla 5.</b> Tabla de direccionamiento LAN.....	60
<b>Tabla 6.</b> Características de Alcatel OT-995.....	61
<b>Tabla 7.</b> Características Router Cisco Linksys E900.....	62
<b>Tabla 8.</b> Tabla de direccionamiento WAN.....	68
<b>Tabla 9.</b> Características de los Smartphones.....	76
<b>Tabla 10.</b> Rendimiento de Aplicación.....	76

## **RESUMEN**

El presente proyecto analiza en primera instancia la estructura del Sistema Operativo Android relacionada con Linux, posteriormente se analiza la arquitectura de aplicaciones Android para el desarrollo de una aplicación, tomando en cuenta, las normas legales para publicar aplicaciones en Google Play Store como desarrollador, después se analiza una arquitectura servidor cliente, especificando las tecnologías que poseen tanto el servidor como el cliente, con estas tecnologías se realiza el diseño del cliente Android que cuenta con un análisis de casos de uso y diagramas de flujo, los cuales describen el funcionamiento en servicios y actividades en el desarrollado de la aplicación en Android, paralelamente se diseña el servidor en CentOS el cual tiene servicios de FTP, WEB y Server Socket con las que se cuenta para la gestión de información personal del Smartphone así como también para la adquisición de la ubicación del Smartphone en un servidor MySQL, finalmente se realiza pruebas de la aplicación en el cliente verificando que los servicios se desplieguen y análisis de resultados enfocándonos en el rendimiento de la aplicación en tres diferentes Smartphones.

### **PALABRAS CLAVES:**

- **ANDROID**
- **SMARTPHONE**
- **LINUX**
- **JAVA**
- **WEB**

## **ABSTRACT**

This project analyzes the structure at first Android Operating System related to Linux, then the architecture of android applications for the development of an application is analyzed, taking into account the legal rules to publish applications in Google Play Store as a developer, then analyzes a client server architecture, specifying the technologies that have both the server and the client, the design of these technologies Android client that has a use case analysis and flow charts that describe the operation is performed and services activities in the developed application in Android, parallel server in CentOS which has FTP services, Web and Server Socket with that account for personal information management Smartphone as well as to acquire the location is designed Smartphone into a MySQL server, finally testing the application is done on the client verifying that the services are deployed and results analysis focusing on the performance of the application in three different Smartphones.

### **KEYWORDS:**

- **ANDROID**
- **SMARTPHONE**
- **LINUX**
- **JAVA**
- **WEB**



## **CAPÍTULO I**

### **1. INTRODUCCION**

El primer Smartphone o teléfono inteligente que atrajo fue el BlackBerry de la compañía canadiense Research In Motion, pero, como resultado de que en el mercado los celulares su costo y el mantenimiento del servicio ponía esta tecnología lejos del alcance de la gran mayoría de usuarios, no fue sino hasta la introducción del iPhone en el 2007, que el concepto de un Smartphone fácil de usar, estandarizado y de precio accesible en países desarrollados tecnológicamente desató la masificación de los Smartphones.

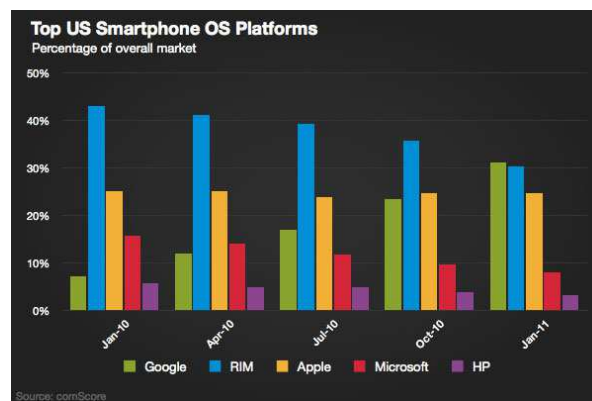
Cuando BlackBerry estaba en la cima del mercado había otros competidores disputándose el mercado entre ellos Treo de PalmInc y Symbian de Nokia. Pero la falta de estándares entre proveedores celulares hacía que fuera necesario personalizar el hardware al tipo de tecnología del carrier, esto fragmentaba el mercado e impedía aprovechar las ventajas de la economía de escala.

Andy Rubín tuvo la idea de desarrollar un sistema operativo para celulares basado en la filosofía de Código abierto y para ello se creó Android. Rubín se había dado cuenta de que la gran fragmentación del mercado hacía imposible que la tecnología evolucionara rápidamente en el sector de los celulares. Por lo tanto decidió plantear la idea de un sistema operativo para celulares que fuera de código abierto, adaptable a cualquier hardware, pero ofreciera un entorno de desarrollo único que permitiera crear

aplicaciones para el sistema operativo además pudieran correr en cualquier hardware que lo soportara; esta idea es la misma que estaba detrás del éxito del IBM PC.

Rubín se aproximó a Google para ofrecer exclusividad en las búsquedas realizadas desde los celulares con Android a cambio de que Google expresara su apoyo público a la plataforma. Luego de que Rubín le hiciera la presentación a Larry Page en el 2005, este recibió una oferta de compra de parte de Google por \$50 millones, y la dirección del departamento de la compañía que se encargaría del desarrollo de la plataforma para celulares.

El G1 que fue primer celular con Android ofrecido por Google apareció a finales del 2008, pero con la llegada del Nexus One y Android 2.1 en enero de 2010, que las ventas de celulares con Android se incrementaron y consiguieron posicionarse en más de la mitad del mercado de smartphones en Estados Unidos. Y así desplazando a BlackBerry y Apple. En la figura 1 se muestra el rápido ascenso de Android en comparación con todos los demás sistemas operativos para dispositivos móviles.



**Figura 1. Porcentaje de Mercado de S.O para Smarthphones.**

**Fuente: (Victor, 2013)**

## **1.2 DEFINICIÓN DEL PROBLEMA**

A su vez con el desarrollo de la tecnología y el ascenso de nuevos productos tecnológicos como Smartphones, notebooks y tablets, la sociedad se encuentra en una crisis a nivel mundial refiriéndose así a la delincuencia, más concretamente al robo, que afectaría al gran porcentaje de la población mundial lo cual se debería buscar soluciones que si se da el robo de uno de estos productos, más específicamente los Smartphone, se lo podría localizar de manera ágil y precisa, pero a su vez en casos diferentes debería haber la capacidad de poder recuperar información que se cuenta en el Smartphone.

En el mercado se han desarrollado muchas aplicaciones sobre android, con el objetivo de encontrar la ubicación y realizar tareas en un dispositivo móvil de cualquier fabricante, es así, que se ha desarrollado una gama de aplicaciones con diferentes características como son la aplicación de seguimiento Position Logic para Blackberry, que se ejecuta en cualquier teléfono Blackberry que cumpla con los siguientes requisitos que van desde que el sistema operativo debe ser Black Berry con una versión específica que cuente a su vez con GPS y plan de datos

En lo que se refiere a teléfonos que se desempeñan sobre Android hay una diversidad de aplicaciones las cuales su común característica o función es de encontrar la ubicación del teléfono así como borrar la información de memoria interna SD de forma remota, y así características como bloqueo del teléfono, grabación con el micrófono y aviso si se introduce un nuevo sim; todo esto desde una aplicación online que permite desde cualquier navegador realizar estas tareas.

## **1.3 JUSTIFICACIÓN E IMPORTANCIA**

La Fiscalía de la Unidad de Flagrancia del cuartel Modelo reporto cincuenta y dos denuncias por robo de teléfonos celulares, suscitados en diferentes circunstancias, asalto, extravío, se receptaron en julio pasado.

La Superintendencia de Telecomunicaciones, en el 2011 reportó 500.000 robos de móviles a nivel nacional, es decir más de 40.000 por mes, cerca de 1.333 aparatos robados al día pero en el 2012, de acuerdo a la entidad estatal, la cifra descendió a 216.000, supuestamente producto de la exigencia de empadronar los celulares y las líneas para bajar el delito. La medida corrió específicamente para los usuarios prepago, que a diferencia de los postpago no se registraban al momento de adquirir un equipo o tarjeta sim, pero, no es hasta el 5 de abril del 2012, según el Ministerio de Telecomunicaciones, que de 13'466.808 líneas prepago se empadronaron 817.714 y en julio de ese año unas 130.000 líneas prepago fueron desactivadas por no cumplir con el registro.

Una de las medidas que tomo la SUPERTEL (Superintendencia de Telecomunicaciones) es la de bloquear permanentemente 165 000 teléfonos móviles que habían sido reportados como robados.

Con la delincuencia el ciudadano o usuario en general de los Smartphones está expuesto a ser víctima de robos, por tal motivo con el diseño de una aplicación que permita localizar la ubicación exacta del Smartphone hace que el delito no se consuma en su totalidad.

Es fundamental entender la gran importancia que es la información de datos personales para cada usuario, más que la pérdida del objeto como tal, la información se convierte en algo fundamental a recuperar para el ultrajado, para esto se toman herramientas tecnológicas que ayudan a hacer que la información pueda ser recuperada de manera remota a su vez clave es el contacto del teléfono con el internet, lo cual permite el intercambio de información en una sinergia servidor cliente que hace de esto una herramienta eficiente.

Ya que la importancia de la información es lo más primordial, también la ubicación del Smartphone pero utilizado como un medio de control parental, es decir que cuando se requiera saber dónde se encuentra una persona se hará uso del GPS del Smartphone y como medio de gestión se podrá realizar actividades en el Smartphone

según las necesidades de cada usuario como llamada de auxilio, mensajes que se enviaran automáticamente, filmación y más actividades.

#### **1.4 ALCANCE DEL PROYECTO**

El presente proyecto consistió en el diseño de una aplicación basada en Android para localizar posibles teléfonos robados y permitir realizar una gestión de los datos guardados en el teléfono para que se los pueda recuperar de manera remota o a su vez puedan ser eliminados posteriormente, para lo cual se hizo un análisis de que información es factible obtener y cual no.

El desarrollo de la aplicación en Android utiliza como herramientas de control el GPS el que permite localizar y monitorear el Smartphone en el caso que este fue víctima de robo o el simple hecho de no saber dónde se encuentra.

Se realizó el diseño de dos aplicaciones complementarias la primera el cliente, va a ser la aplicación para el Smartphone que ha sido desarrollada sobre Android y la segunda aplicación del servidor sobre una PC, la cual es la encargada de rastrear al teléfono por medio del internet, más específicamente con el desarrollo de la aplicación la cual será situada en un servidor y hará el proceso de gestión para poder eliminar o transferir los datos actuales del Smartphone con la única condición que el Smartphone debe que tener un plan de datos activo, el cual ayudara a ser encontrado.

Es fundamental contar un efectivo sistema de protección de datos personales este sea preventivo o una solución posterior lo cual es la garantía de la seguridad de la información, entendida como la implementación a las medidas administrativas, físicas y técnicas eficaces para garantizar y velar por la integridad, confidencialidad y disponibilidad de tus datos personales.

Pero en el caso posterior a la perdida de la información por el robo, el diseño de esta aplicación contribuye a minimizar la pérdida total o parcial, entre otras, que pueden lesionar tus derechos o libertades fundamentales.

## **1.5 OBJETIVOS**

### **1.5.1 Objetivo General**

- Diseñar e implementar una aplicación para la gestión remota de un Smartphone con sistema operativo android.

### **1.5.2 Objetivos Específicos**

- Analizar los lenguajes de programación para desarrollo de aplicaciones en Android
- Diseñar e Implementar la aplicación para el Smartphone Cliente
- Diseñar e Implementar la aplicación para el Servidor
- Evaluar el desempeño de la aplicación en el Smartphone Cliente
- Evaluar el desempeño de la aplicación para el Servidor

## **CAPÍTULO II**

### **2. ESTUDIO DE LA ARQUITECTURA DE CLIENTE SERVIDOR**

#### **2.1 TECNOLOGÍAS PARA EL CLIENTE**

En abril de 1991, Linus Torvalds, empezó a trabajar en unas simples ideas para un núcleo de un sistema operativo mientras era estudiante de ciencias de computación , inmediatamente empezó con un intento por obtener un núcleo de sistema operativo gratuito similar a Unix que funcionara con microprocesadores Intel 80386, al instaurarse Linux consiguió rápidamente desarrolladores y usuarios que adoptaron códigos de otros proyectos de software libre para usarlo con el nuevo sistema operativo lo cual significo contribuciones de miles de programadores de todo el mundo que normalmente Linux se utiliza junto a un empaquetado de software, llamado distribución GNU/Linux y servidores.

El kernel o núcleo de Linux es el componente central de este sistema operativo, el cual cumple la función de que el software y el hardware puedan trabajar juntos, pero más que eso las funciones más importantes del mismo, aunque no las únicas, son:

- Administración de la memoria para todos los programas y procesos en ejecución.

- Administración del tiempo de procesador que los programas y procesos en ejecución utilizan.
- Es el encargado de que podamos acceder a los periféricos y elementos de nuestro ordenador de una manera cómoda.

Hasta que empezó el desarrollo de la serie 2.6 del núcleo, existieron dos tipos de versiones del núcleo:

**Versión de producción:** La versión de producción, era la versión estable hasta el momento, esta era el resultado final de las versiones de desarrollo o experimentales, cuando el equipo de desarrollo del núcleo experimental, decidía que tenía un núcleo estable y con la suficiente calidad, se lanzaba una nueva versión de producción o estable, esta versión era la que se debía utilizar para un uso normal del sistema, ya que eran las versiones consideradas más estables y libres de fallos en el momento de su lanzamiento.

**Versión de desarrollo:** Esta versión era experimental y era la que utilizaban los desarrolladores para programar, comprobar y verificar nuevas características, correcciones, estas versiones eran inestables y no se debían usar sin saber lo que se hacía.

### 2.1.1 Arquitectura de Android

Después de comprender que es el kernel de linux es necesario entender el rol que cumple este mismo kernel en el sistema operativo android y por qué se relacionaron, esto ocurre porque Linux al ser un sistema operativo por esencia claramente libre, ofreciendo una alternativa ante los sistemas cerrados y protegidos por copyright, fue la elección lógica y apropiada cuando Android decidió formarse, al ser tan abierto y libre, el kernel de Linux podía ser moldeado teniendo en cuenta las necesidades que podría tener este nuevo sistema operativo móvil, de todas formas la relación de Android y de Linux es de colaboración ya que Google que por cierto es propietario de



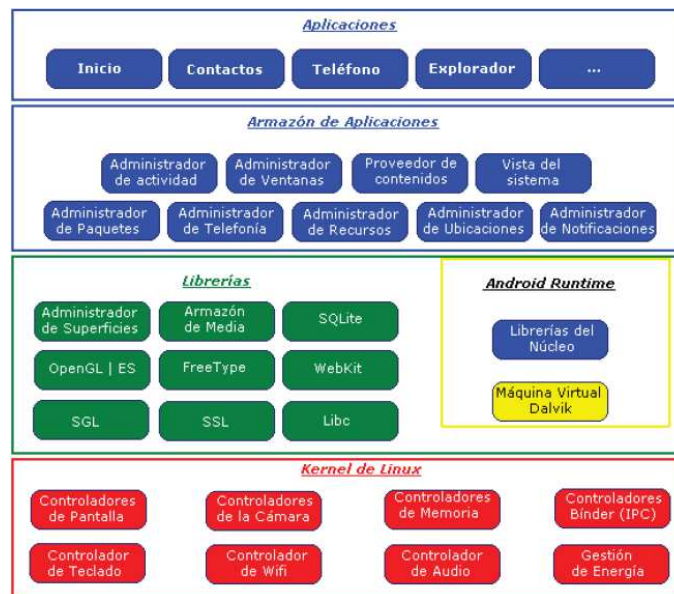
android, está añadiendo muchas cosas a ese kernel, que después ha pasado a ser responsabilidad de Linux.

Google tiene que mantener el código que añade por su cuenta para crear Android, pero no está obligado a mantener el código perteneciente a Linux, sino que de eso se encargan empleados y voluntarios entusiastas de Linux, con esto para Google lo más sencillo sería que casi todo el código de Android estuviese incluido en Linux, y así solo tendría que gastar recursos en mantener una pequeña parte.

En 2010 esta actitud llevó a algunos desarrolladores de Linux a tomar una decisión de eliminar código de Android incluido en Linux, con el pretexto de que no estaba siendo mantenido por nadie, afortunadamente desde entonces estas hostilidades se han enfriado, y aunque Google sigue manteniendo para sí una buena porción de Android, en cada nueva versión del kernel de Linux se están añadiendo nuevas funcionalidades pensadas especialmente para Android.

El núcleo de Android está formado por el sistema operativo Linux versión 2.6, esta capa proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de drivers para dispositivos, esta capa del modelo actúa como capa de abstracción entre el hardware y el resto de la pila, por lo tanto, es la única que es dependiente del hardware

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria externa y más, en la Figura 2.2 se muestran las capas que conforman el sistema operativo Android:



**Figura 1. Capas en Sistema Operativo Android. Fuente: (Báez, 2013)**

En la figura 2.2 se puede analizar que la base de android es el kernel de linux propiamente dicho, donde android puede acceder a todos los controladores que se pudiese según las características del hardware.

Posteriormente la capa de Android Runtime está basado en el concepto de máquina virtual utilizado en Java, dado las limitaciones de los dispositivos donde ha de correr Android esto puede ser poca memoria y procesador limitado no fue posible utilizar una máquina virtual Java estándar, entonces Google tomó la decisión de crear una nueva, la máquina virtual Dalvik, que respondiera mejor a estas limitaciones.

La máquina virtual Dalvik facilita optimización de recursos es decir que ejecuta ficheros Dalvik ejecutables con extensión dex que es un formato optimizado para ahorrar memoria, además, está basada en registros a su vez cada aplicación corre en su propio proceso Linux con su propia instancia de la máquina virtual Dalvik este proceso de Dalvik delega al kernel de Linux algunas funciones como threading y el manejo de la memoria a bajo nivel.

También se incluye en el Runtime de Android el “Núcleo de Librerías” con la mayoría de las librerías disponibles en el lenguaje Java.

- **System C library:** Es una derivación de la librería BSD de C estándar (libc), adaptada para dispositivos embebidos basados en Linux.
- **Media Framework:** Librería basada en PacketVideo's OpenCORE que soporta códec de reproducción y grabación de multitud de formatos de audio vídeo e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.
- **Surface Manager:** Maneja el acceso al subsistema de representación gráfica en 2D y 3D.
- **WebKit:** Soporta un moderno navegador web utilizado en el navegador Android y en la vista webview, se trata de la misma librería que utiliza Google Chrome y Safari de Apple.
- **SGL:** Motor de gráficos 2D.
- **Librerías 3D:** Implementación basada en OpenGL, las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.
- **FreeType:** Fuentes en bitmap y renderizado vectorial.
- **SQLite:** Potente y ligero motor de bases de datos relacionales disponible para todas las aplicaciones.
- **SSL:** Proporciona servicios de encriptación Secure Socket Layer.

Entorno de aplicación proporciona una plataforma de desarrollo libre para aplicaciones con gran riqueza e innovaciones como sensores, localización, servicios, barra de notificaciones, esta capa ha sido diseñada para simplificar la reutilización de componentes, las aplicaciones pueden publicar sus capacidades y otras pueden hacer uso de ellas.

Una de las mayores fortalezas del entorno de aplicación de Android es que se aprovecha el lenguaje de programación Java, el SDK de Android no acaba de ofrecer todo lo disponible para su estándar del entorno de ejecución Java (JRE), pero es compatible con una fracción muy significativa de la misma.

Los servicios más importantes que incluye son:

- **Views:** Conjunto de vistas, es decir parte visual de los componentes
- **Resource Manager:** Proporciona acceso a recursos que no son en código.
- **Activity Manager:** Maneja el ciclo de vida de las aplicaciones y proporciona un sistema de navegación entre ellas.
- **Notification Manager:** Permite a las aplicaciones mostrar alertas personalizadas en la barra de estado.
- **Content Providers:** Mecanismo sencillo para acceder a datos de otras aplicaciones como los contactos.

En la capa de Aplicaciones está formado por el conjunto de aplicaciones instaladas en una máquina Android, todas estas corren en la máquina virtual Dalvik para garantizar la seguridad del sistema, normalmente las aplicaciones Android están escritas en Java.

### 2.1.2 Versiones de Plataforma Android

Android está diseñado para funcionar en una gama diferente de dispositivos, como teléfonos, tabletas o televisores, el hecho es que para poder desarrollar android todo su potencial necesita que todos los dispositivos donde se los quiere utilizar tienen que tolerar cierta variabilidad característica y proporcionar una interfaz de usuario flexible que se adapta a las diferentes configuraciones de pantalla.

Android proporciona un marco de aplicación dinámica en que se proporciona recursos de aplicaciones específicas de la configuración en archivos estáticos por ejemplo como los diferentes diseños de XML para los diferentes tamaños de pantalla.

Puede especificar los requisitos de características de su aplicación y el control de los tipos de dispositivos pueden instalar su aplicación desde Google Play Store, la compatibilidad en el desarrollo de Android es vital en diversas situaciones hay dos

tipos de compatibilidad: compatibilidad de dispositivos y de compatibilidad de aplicaciones.

Debido a que Android es un sistema operativo de código abierto, cualquier fabricante de hardware puede construir un dispositivo que funciona con el sistema operativo Android, pero, un dispositivo es considerada "Android compatible" sólo si puede funcionar correctamente aplicaciones escritas para el entorno de ejecución de Android, los detalles exactos del entorno de ejecución de Android están definidos por el programa de compatibilidad Android y cada dispositivo deben pasar la prueba de compatibilidad Suite (CTS) con el fin de ser considerado compatible.

Se debe proveer si su aplicación es compatible con cada configuración de dispositivo, debido a que Android se ejecuta en una amplia gama de configuraciones de dispositivos, algunas características no están disponibles en todos los dispositivos como por ejemplo, algunos dispositivos pueden no incluir un sensor de brújula o acelerómetro, pero, si la funcionalidad principal de su aplicación requiere el uso de un acelerómetro o sensor de brújula, su aplicación sólo es compatible con dispositivos que incluyen un sensor de brújula.

Interfaz de programación de aplicaciones o API es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción que son usadas generalmente en las "librerías", además es una fuente de código basado en la especificación destinada a ser utilizada como una interfaz de componentes de software para comunicarse entre sí.

Una API puede incluir especificaciones para las rutinas y estructuras de datos, las clases de objetos y variables, a medida que desarrolla su aplicación en Android, que es útil para comprender el enfoque general de la plataforma para la gestión del cambio de la API, también es importante entender el identificador del nivel de la API y el papel que desempeña en la consecución de la compatibilidad de la aplicación con los dispositivos en los que puedan estar instalados, para esto el nivel del API es importante conocer, a su vez el nivel API es un valor entero que identifica la revisión de la API en el marco que ofrece una versión de la plataforma Android.

La plataforma Android ofrece una API de marco en el que las aplicaciones pueden utilizar para interactuar con el sistema Android.

La API de marco consiste en:

- Un conjunto básico de paquetes y clases
- Un conjunto de elementos y atributos XML para la declaración de un archivo de manifiesto
- Un conjunto de elementos y atributos XML para la declaración y el acceso a los recursos
- Un conjunto de Intenciones
- Un conjunto de permisos que pueden solicitar aplicaciones, así como refuerzos de permiso incluido en el sistema

Es muy importante observar que hay clases y métodos que están disponibles a partir de una versión de API, si las vamos a usar hemos de conocer la versión mínima necesaria.

Cuando se ha lanzado una nueva plataforma siempre ha sido compatible con las versiones anteriores, es decir, solo se añaden nuevas funcionalidades y en el caso de modificar alguna funcionalidad no se elimina, se etiquetan como obsoletas pero se pueden continuar utilizando.

La tabla 1 especifica el nivel de API con el apoyo de cada versión de la plataforma Android.

**Tabla 1**  
**Especificaciones de Versiones API**

<b>VERSION DE LA PLATAFORMA</b>	<b>NIVEL API</b>	<b>CODIGO DE VERSION</b>
<b>Android 1.0</b>	1	BASE
<b>Android 1.1</b>	2	BASE_1_1
<b>Android 1.5</b>	3	MAGDALENA
<b>Android 1.6</b>	4	DONUT
<b>Android 2.0.1</b>	5	ECLAIR_0_1
<b>Android 2.0.1</b>	6	ECLAIR_MR1
<b>Android 2.1.x</b>	7	ECLAIR
<b>Android 2.2.x</b>	8	FROYO
<b>Android 2.3.2 - 2.3.1</b>	9	GINGERBREAD
<b>Android 2.3.4 - 2.3.3</b>	10	GINGERBREAD_MR1
<b>Android 3.0.x</b>	11	PANAL
<b>Android 3.1.x</b>	12	HONEYCOMB_MR1
<b>Android 3.2</b>	13	HONEYCOMB_MR2
<b>Android 4.0 - 4.0.1 - 4.0.2</b>	14	ICE_CREAM_SANDWICH
<b>Android 4.0.3</b>	15	ICE_CREAM_SANDWICH_MR
<b>Android 4.1</b>	16	JELLY_BEAN
<b>Android 4.2</b>	17	JELLY_BEAN_GUMMY_BEAN
<b>Android 4.3</b>	18	JELLY_BEAN
<b>Android 4.4</b>	19	KIT_KAT
<b>Android 5</b>	20	LOLLIPOP

Fuente: (Tomás, 2014)

Tabla 2

## Características de las Versiones API

VERSION API	CARACTERISTICAS
<b>Android 1.0</b>	Primera versión de Android nunca se utilizó comercialmente, por lo que no tiene mucho sentido desarrollar para esta plataforma.
<b>Android 1.1</b>	No se añadieron mayores servicios pero simplemente fijaron algunos errores de la versión anterior pero a su favor es la opción si queremos desarrollar una aplicación compatible con todos los dispositivos Android aunque ya su presencia en el mercado sea casi nula.
<b>Android 1.5</b>	Es la primera versión con algún usuario, como novedades, se incorpora el teclado en pantalla con predicción de texto, los terminales ya no tienen que tener un teclado físico, así como la capacidad de grabación avanzada de audio y vídeo, además aparecen los widgets de escritorio y live folders, también incorpora soporte para bluetooth estéreo, por lo que permite conectarse automáticamente a auriculares bluetooth.
<b>Android 1.6</b>	Se incorpora la búsqueda avanzada en todo el dispositivo, el cual se complementa con la posibilidad de la síntesis de texto a voz, también se implementa el poder trabajar con diferentes densidades de pantalla, posteriormente se da soporte para resolución de pantallas WVGA y es aquí donde aparece un nuevo atributo XML y el evento onClick con la actualización de Play Store.
<b>Android 2.0.1</b>	Esta versión de API apenas cuenta con usuarios, esta versión incorpora un API para manejar el bluetooth 2.1 el cual permite sincronizar con cualquier otro dispositivo también se mejora la gestión de contactos y ofrece más ajustes en la cámara, añade nueva interfaz del navegador y soporte para HTML5, también se cuanta con la clase MotionEvent que soporta eventos en pantallas multitáctil.
<b>Android 2.1.x</b>	Destacamos el reconocimiento de voz que permite introducir un campo de texto dictando sin necesidad de utilizar el teclado, se puede obtener información sobre la señal de la red actual que posea el dispositivo y en el paquete WebKit se incluyen nuevos métodos para manipular bases de datos almacenadas en Web.
<b>Android 2.2.x</b>	Se mejora de velocidad de ejecución de las aplicaciones, esto se consigue con la introducción de un nuevo compilador JIT de la máquina Dalvik, a su vez se añaden varias mejoras relacionadas con el navegador Web, como el soporte de Adobe Flash 10.1 y la incorporación del motor JavaScript V8 utilizado en Chrome, se proporciona un servicio para la copia de seguridad de datos que se puede realizar desde la propia aplicación para garantizar al usuario el mantenimiento de sus datos.

CONTINUA →



VERSION API	CARACTERISTICAS
<b>Android 2.3.1 - 2.3.2</b>	<p>Soporte nativo para SIP y telefonía por internet VoIP.</p> <p>Entrada de texto del teclado virtual más rápida e intuitiva, con mejoras en precisión, texto sugerido y entrada por voz.</p> <p>Nuevos efectos de audio tales como reverberación, ecualizador, virtualización de audífonos y aumento de bajos</p> <p>Nuevo gestor de descargas, que da a los usuarios fácil acceso a cualquier archivo descargado del navegador, correo electrónico u otra aplicación.</p> <p>Soporte para múltiples cámaras en el dispositivo, incluyendo cámara frontal facial, si está disponible.</p> <p>Soporte para reproducción de video por WebM/VP8, codificación de audio por AAC</p> <p>Mejoras en la administración de la energía, con un mayor rol activo en aplicaciones de administración que se mantienen activas en el dispositivo por mucho tiempo.</p> <p>Mejorado soporte para el desarrollo de código nativo.</p> <p>Cambio desde YAFFS a ext4 en dispositivos nuevos.</p> <p>Mejoras en audio, gráficos y entrada para desarrolladores de juegos.</p> <p>Recolector basura concurrente para incrementar el rendimiento.</p> <p>Soporte nativo para más sensores tales como giroscopio y barómetro.</p>
<b>Android 2.3.3 - 2.3.7</b>	<p>Rebaja de la seguridad de SSL al usar protocolos de cifrado inseguros</p> <p>Mejoras en el rendimiento por red del Nexus S 4G.</p> <p>Arreglado una falla de Bluetooth en el Samsung Galaxy S.</p> <p>Animación de sombras al deslizar por listas.</p> <p>Mejorada la eficiencia de la batería.</p>
<b>Android 3.0.x</b>	<p>Optimizada para dispositivos con pantallas grandes con los fragments, con los que podemos diseñar diferentes elementos del interfaz de usuario, también se añade la barra de acciones, donde las aplicaciones pueden mostrar un menú siempre visible con las teclas en pantalla y sin ser físicas.</p> <p>Se mejora los gráficos 2D/3D gracias al renderizador OpenGL acelerado por hardware posteriormente aparcera el nuevo motor de gráficos Rederscript, que saca mayor rendimiento al hardware e incorpora su propia API.</p> <p>Primera versión de la plataforma que soporta procesadores multinúcleo aparte de que la máquina virtual Dalvik se optimiza para permitir multiprocesamientos, lo que permite una ejecución más rápida de las aplicaciones, incluso aquellas que son de hilo único.</p> <p>Se incorporan varias mejoras multimedia, como listas de reproducción M3U a través de HTTP Live Sreaming, soporte a la protección de derechos musicales (DRM) y soporte para la transferencia de archivos multimedia a través de USB con los protocolos MTP y PTP.</p> <p>Se añade las nuevas políticas administrativas con encriptación del almacenamiento, caducidad de contraseña y mejoras para administrar los dispositivos de empresa de forma eficaz.</p>

CONTINUA 

VERSION API	CARACTERISTICAS
<b>Android 3.1.x</b>	Se permite manejar dispositivos conectados por USB tanto host como dispositivo además se desarrolla el Protocolo de transferencia de fotos y vídeo (PTP/MTP) y de tiempo real (RTP).
<b>Android 3.2</b>	Optimizaciones para distintos tipos de tableta, Zoom compatible para aplicaciones de tamaño fijo, también, Sincronización multimedia desde SD.
<b>Android 4.0 - 4.0.X</b>	Se unifican las dos versiones anteriores 2.x para teléfonos y 3.x para tabletas en una sola compatible con cualquier tipo de dispositivo, aparece un nuevo gestor de tráfico de datos por Internet, donde podremos ver el consumo de forma gráfica y donde podemos definir los límites a ese consumo para evitar cargos inesperados con la operadora.
<b>Android 4.0.3</b>	Se introducen ligeras mejoras en algunas APIs incluyendo el de redes sociales, calendario, revisor ortográfico, texto a voz y bases de datos.
<b>Android 4.1</b>	En la interfaz de usuario se soluciona su fluidez incorporando varias técnicas, como sincronismo vertical, triple búfer y aumentar la velocidad del procesador al tocar la pantalla, también para mejorar la seguridad las aplicaciones son cifradas.
<b>Android 4.2</b>	Aparece o se incorpora un nuevo teclado predictivo deslizante al estilo Swype, también se incorpora la posibilidad conectar dispositivo y TVHD mediante wifi (Miracast).
<b>Android 4.3</b>	Esta versión los programadores pueden definir restricciones en las apps, que los propietarios puedan activar si quieren, además se da soporte para Bluetooth Low Energy (BLE) y en la seguridad gestiona y oculta las claves privadas y credenciales.
<b>Android 4.4</b>	Todos los componentes principales de Android han sido recortados para reducir sus requerimientos de memoria, y se ha creado una nueva API que permite adaptar el comportamiento de la aplicación en dispositivos con poca memoria, además en consumo de energía se mejoran los sensores para disminuir el consumo y se incorpora un sensor contador de pasos. Se incorpora un administrador de impresión para enviar documentos a través de WiFi a una impresora, también se añade un content provider para gestionar los SMS además, para los desarrolladores se añade una nueva máquina virtual ART, que consigue tiempos de ejecución muy superiores a la máquina Dalvik, sin embargo, todavía está en una etapa experimental.
<b>Android 5.0</b>	Una característica de ahorro de batería que se extiende el uso de dispositivos de hasta 90 minutos. El tiempo estimado de batería restante aparece cuando el dispositivo está enchufado. El tiempo restante de batería antes de tener que cargar el dispositivo de nuevo ahora se puede encontrar en la configuración de la batería.

Fuente: (Tomás, 2014)

### 2.1.3 Arquitectura de Aplicaciones Android

En primer lugar al hablar de herramientas de desarrollo de aplicaciones nos referimos a los componentes u objetos con los que tenemos que contar y entender su función y manejo para desarrollar una aplicación en android, a continuación las herramientas necesarias.

**VistaView:** Las vistas son los elementos que componen la interfaz de usuario de una aplicación por ejemplo un botón, una caja de texto, todos los objetos son descendientes de la clase View, y por tanto, pueden ser definidos utilizando código Java.

**Layout:** Es un grupo de vistas apiladas de una determinada forma es decir diferentes tipos de formas ya sea esta lineal, en cuadrícula o indicando la posición constante de cada vista, como no podría ser de otra manera los Layouts también son objetos descendientes de la clase View por tal motivo los Layouts pueden ser definidos en código, aunque la forma habitual de definirlos es utilizando código XML.

**Actividades:** El concepto de actividad en Android representa una unidad de interacción con el usuario, lo que llamamos una pantalla de la aplicación, cada actividad tiene asociada una vista o view esto constituye el conjunto de elementos gráficos que se presentan al usuario en pantalla para la interacción con el sistema es decir como una caja de edición o un botón.

Para crear actividades personalizadas es necesario definir una clase que herede de la clase base Activity a su vez una aplicación suele estar formada por una serie de actividades, de esta forma el usuario puede navegar entre actividades.



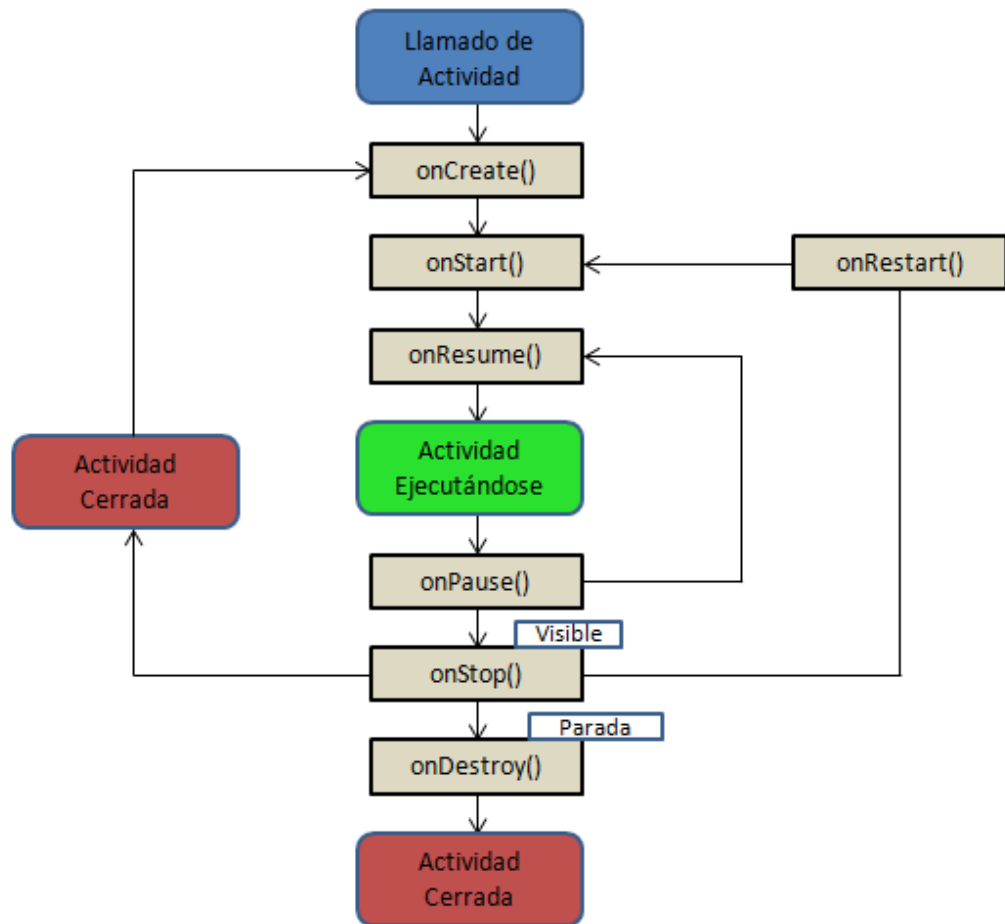
**Figura. 2. Jerarquía de Actividades**

Los elementos como botones o cajas de texto que se muestran en las actividades deben ser definidos en un fichero xml que tiene un nombre NombreActividad.xml el cual se encuentra en la carpeta ./res/layout, al formarse este fichero xml se formara automáticamente una clase en java (NameActivity.class) que podrá operar o contar con los objetos declarados en el fichero xml.

El ciclo de vida de una aplicación es controlado principalmente por el sistema, en lugar de ser controlado directamente por el usuario, básicamente las actividades son las que controlan el ciclo de vida de las aplicaciones, dado que el usuario no cambia de aplicación, sino de actividad por eso el sistema va a mantener una pila con las actividades previamente visualizadas, de forma que el usuario va a poder regresar a la actividad anterior pulsando la tecla de retorno.

Como vemos, Android es sensible al ciclo de vida de una actividad, por lo tanto necesitas comprender y manejar los eventos relacionados con el ciclo de vida si quieres crear aplicaciones estables.

A continuación en la figura 4 se muestra el ciclo de vida de una actividad.



**Figura. 3. Ciclo de Vida de una Aplicación.**

**Fuente: (Pavón, 2013)**

En la tabla 3 se puede observar cada una de las etapas del ciclo de vida de una aplicación en Android

Tabla 3

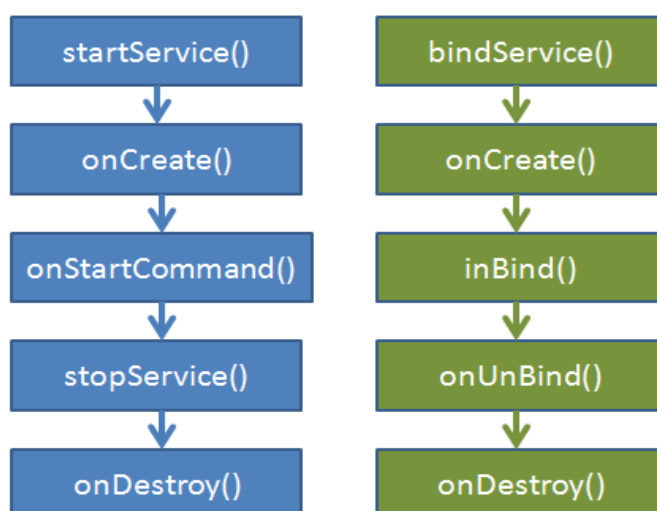
## Descripción de Vida de una Aplicación

METODO	DESCRIPCIÓN
<b>Actividad Ejecutándose (Running)</b>	La actividad está encima de la pila, lo que quiere decir que esta ejecución y es la tarea principal.
<b>Visible (Paused)</b>	La actividad se encuentra semi-suspendida, es decir, aún se está ejecutando y esta visible, pero no es la tarea principal, se guarda la información en este estado para prevenir una posible pérdida de datos en caso de que el sistema decida prescindir de ella para liberar memoria.
<b>Parada (Stopped)</b>	Cuando la actividad no es visible, el programador debe guardar el estado de la interfaz de usuario, preferencias y otros.
<b>Actividad cerrada (Destroyed)</b>	Cuando la actividad termina al invocarse el método <i>finish()</i> , o es destruida por el sistema.
<b>onCreate (Bundle)</b>	Se llama en la creación de la actividad, se utiliza para realizar todo tipo de inicializaciones, como la creación de la interfaz de usuario o la inicialización de estructuras de datos, también puede recibir información de estado de la actividad (en una instancia de la clase Bundle), en el caso que se reanuda desde una actividad que ha sido destruida y vuelta a crear.
<b>onStart()</b>	Nos indica que la actividad está a punto de ser mostrada al usuario.
<b>onResume()</b>	Se llama cuando la actividad va a comenzar a interactuar con el usuario.
<b>onPause()</b>	Indica que la actividad está a punto de ser lanzada a segundo plano, normalmente porque otra actividad es lanzada.
<b>onStop()</b>	La actividad ya no va a ser visible para el usuario.
<b>onRestart()</b>	Indica que la actividad va a volver a ser representada después de haber pasado por onStop().
<b>onDestroy()</b>	Se llama antes de que la actividad sea totalmente destruida.

Fuente: (Pavón, 2013)

**Servicios:** Los servicios son tareas no visibles que se ejecutan siempre por debajo, incluso cuando la actividad asociada no se encuentra en primer plano, es un buen método para realizar tareas paralelas ya que una de sus características es que tiene un hilo propio pero siempre tiene que ser lanzado o llamado por una aplicación en primer plano es decir no se puede ejecutar solo, también otra característica es que no necesita interfaz, a no ser que se pida explícitamente, en cuyo caso la clase Service la exportaría.

A continuación en la figura 5 se muestra el ciclo de vida de un servicio.



**Figura. 4. Ciclo de Vida de un Servicio.**

**Fuente: (Pavón, 2013)**

El ciclo de vida de un servicio como se muestra en la figura. 5 se inicia con el método `onCreate(Bundle)`, y se libera con el método `onDestroy()`, sin embargo, el desarrollo puede llevarse a cabo de dos maneras, dependiendo de cómo se lance:

Si se llama al método `startService()`, esto implicará que el servicio ejecutará todo su ciclo vital, el siguiente método tras `onCreate(Bundle)` será `onStartComand(Intent, int)`, para terminar el servicio externamente, se usa `stopService()`, e internamente, `stopSelf()` o `stopSelfResult()`,

En otro caso, si el servicio se llama con `bindService()`, el usuario podrá interactuar mediante la interfaz que exporta el servicio, y tras `onCreate(Bundle)` se ejecutará el método `onBind(Intent)`, en este caso, el servicio se termina llamando al método `onUnbind(Intent)`, también es posible reiniciarlo con el método `onRebind(Intent)`.

**Receptores de mensajes de distribución:** También llamados broadcast receiver o notificaciones, son los encargados de reaccionar ante los eventos ocurridos en el dispositivo, ya sean generados por el sistema o por una aplicación externa, estos no tienen interfaz, pero pueden lanzar una activity por medio de un evento, la clase que defina estos componentes heredará de la clase `BroadCastReceiver`.

Su ciclo de vida es muy corto, ya que solo están activos mientras se ejecuta el método `onReceive (Context, Intent)`, que es equivalente al `onCreate(Bundle)` de otros componentes, el objeto `Context` nos pasa el estado actual, y el `intent`, nos permitirá lanzar el evento.

**Proveedores de contenidos:** Estos proveedores se encargan de que la aplicación pueda acceder a la información que necesita, siempre que se haya declarado el correspondiente provider en el `AndroidManifest`, compartiendo información sin revelar estructura u orden interno.

Implementan una interfaz, pero se comunica con ella a través de la clase `ContentResolver`, cada vez que se usa un `ContentResolver`, se activa un `ContentProvider`, para obtener los datos necesarios, es necesario conocer la URI (identificador) del dato, los campos que tiene, y los tipos de esos campos, con esto ya podemos llamar al método `ContentResolver.query()`.



#### 2.1.4 Desarrollador de Aplicaciones Android

Para publicar aplicaciones de Android en Google Play, es necesario tener una de desarrollador de Google Play, con esa cuenta de Google, se registra para obtener una cuenta de desarrollador, cuando se obtiene la cuenta, se utiliza la consola de Google Play para desarrolladores para publicar y administrar aplicaciones.

Durante el proceso de configuración, se revisa se acepta el Acuerdo de distribución para desarrolladores de Google Play el cual especifica entre lo más importante.

- Aceptación del Acuerdo
- Precios y Pagos
- Uso de Google Play Store
- Concesiones de Licencias
- Publicidad y Características de Marca
- Solicitudes de eliminación del producto.
- Credenciales de Programador
- Privacidad e información
- Recisión del Acuerdo.
- Renuncia de Garantías.
- Limitación de Responsabilidades.
- Indemnización.
- Cambios en el Acuerdo.
- Términos Legales Generales.

**Aceptación del Acuerdo:** “Estar de acuerdo constituye un contrato jurídicamente vinculante entre usted y Google en relación con su uso de Google Play Store para la distribución de Productos, acepta que Google, únicamente en su nombre, y no en nombre de Google, muestre los Productos y los ponga a disposición de los usuarios para su descarga y compra” obtenido de (Google Play, 2014).

**Precios y Pagos:** “Si quiere cobrar una tarifa por sus Productos, también deberá obtener y mantener una Cuenta de pago válida proporcionada por un Procesador de pago autorizado” obtenido de (Google Play, 2014).

**Uso de Google Play Store:** “Acepta que no se involucrará en ninguna actividad con Google Play Store, incluidas la programación o distribución de Productos, que interfiera, afecte, dañe o permita el acceso no autorizado a los dispositivos, servidores, redes, u otras propiedades o servicios de terceros que incluyan, entre otros, a usuarios de Android, a Google o a cualquier proveedor de red móvil. Usted no puede utilizar la información del cliente obtenida de a través de Google Play Store para vender o distribuir Productos fuera de Google Play Store” obtenido de (Google Play, 2014).

Acepta que si utiliza Google Play Store para distribuir Productos, protegerá la privacidad y los derechos legales de los usuarios. Si los usuarios le proporcionan los nombres de usuario, las contraseñas u otra información de acceso o personal, o si su Producto accede a estos o los utiliza, deberá advertir a los usuarios que la información estará disponible en su Producto, así como proporcionar un aviso de privacidad y protección legalmente adecuada para esos usuarios” obtenido de (Google Play, 2014).

Después de aceptar las Acuerdo de distribución para programadores de Google Play se procede a aceptar Políticas del Programa para Desarrolladores de Google Play que especifica entre lo más importante.

### **Políticas de Contenido**

**“Contenido sexual explícito:** están prohibidas las aplicaciones que promuevan la pornografía o incluyan material pornográfico” obtenido de (Google Play, 2014)

**“Violencia y acoso:** no se permite la publicación de imágenes que muestren violencia” obtenido de (Google Play, 2014)

**“Promoción del odio:** no se permite la publicación de contenido que fomente el odio hacia los miembros de un grupo por su raza u origen étnico, religión,

discapacidad, sexo, edad, condición de veterano militar o identidad u orientación sexual”, obtenido de (Google Play, 2014).

**“Suplantación de identidad o comportamiento engañoso:** no debes sustituir la identidad de otras personas ni declarar que tu aplicación está autorizada o producida por otra empresa u organización si esa información no es cierta”, obtenido de (Google Play, 2014).

**“Propiedad intelectual:** no se deben infringir los derechos de propiedad intelectual de terceros (incluidos derechos de patentes, marcas comerciales, secretos comerciales y derechos de autor y otros derechos de propiedad) ni incitar o inducir a que se infrinjan los derechos de propiedad intelectual. Google adoptará las medidas pertinentes si recibe notificaciones precisas de presuntas infracciones de derechos de autor. Para obtener más información al respecto o presentar una solicitud basada en la DMCA, consulta los procedimientos relacionados con los derechos de autor”, obtenido de (Google Play, 2014)

**“Información personal y confidencial:** no se permite la divulgación o la publicación de información de carácter privado y confidencial de los usuarios como, por ejemplo, números de tarjetas de crédito, números de identificación nacional, números de permisos de conducir o de otro tipo de licencias, contactos no públicos o cualquier otra información que no sea de acceso público”, obtenido de (Google Play, 2014).

**“Actividades ilegales:** no se permiten las actividades ilícitas, No debes participar en actividades ilegales al utilizar este producto como, por ejemplo, la venta no autorizada de medicamentos que necesitan prescripción médica”, obtenido de (Google Play, 2014).

**“Juegos de azar:** no se permite ningún tipo de contenido o servicio que promueva los juegos de azar online, entre los que se incluyen los casinos online, las loterías y las apuestas deportivas, así como los juegos de habilidades que ofrecen premios en metálico o en otro valor”, obtenido de (Google Play, 2014).

**“Productos peligrosos:** no se permite la publicación de contenido que sea dañino para redes, servidores u otras infraestructuras, que interfiera en su funcionamiento o que acceda a estos sistemas de forma no autorizada”, obtenido de (Google Play, 2014).

## 2.2 TECNOLOGÍAS PARA EL SERVIDOR

Una distribución Linux es un conjunto de software acompañado del núcleo Linux que se enfoca a satisfacer las necesidades de un grupo específico de usuarios, de este modo hay distribuciones para hogares, empresas y servidores.

Las distribuciones son ensambladas por individuos, empresas u otros organismos, en cada distribución puede incluir cualquier número de software adicional, incluyendo software que facilite la instalación del sistema, tal distribución incluye el núcleo Linux, en la mayoría de los casos las herramientas GNU, al que suelen añadirse también multitud de paquetes de software.

Las herramientas que suelen incluirse en la distribución de este sistema operativo se obtienen de diversas fuentes, y en especial de proyectos de software libre, como GNU, GNOME y KDE, también se incluyen utilidades de otros proyectos como Mozilla, Perl, Ruby, Python, PostgreSQL, MySQL, Xorg, casi todas con licencia GPL.

CentOS "Sistema Comunitario de funcionamiento empresarial" es una ramificación a nivel binario de la distribución "Linux Red Hat Enterprise Linux RHEL", compilado por voluntarios a partir del código fuente liberado por "Red Hat".

Es un sistema operativo de código abierto, basado en la distribución Red Hat Enterprise Linux, operándose de manera similar, y cuyo objetivo es ofrecer al usuario un software de "clase empresarial" gratuito entre sus ventajas es robusto, estable y fácil de instalar y utilizar, desde la versión 5, cada lanzamiento recibe soporte durante diez

años, por lo que la actual versión 7 recibirá actualizaciones de seguridad hasta el 30 de junio de 2024.

Centos es un sistema operativo basado en el marco Linux, que fue desarrollado para el uso libre por la organización Proyecto Centos, como tal, se basa en gran medida en la comunidad de usuarios que son capaces de programar y aplicaciones que se ejecutarán en el sistema de codificación, pero sabiendo que Centos no es el único sistema operativo derivado de Linux, hay varias diferencias que hacen de Centos único entre otros sistemas operativos basados en Linux.

**Tabla 4**  
**Ventajas de Sistema Operativo Centos**

<b>SISTEMA OPERATIVO CENTOS</b>	
<b>CARACTERISTICA</b>	<b>DESCRIPCIÓN</b>
<b>Estabilidad</b>	Ofrece mucha estabilidad, además de ser más operativo para los usuarios que otros sistemas Linux de libre distribución, también CentOS sólo ejecuta las versiones más básicas, estables de programas de software, lo que reduce el riesgo de fallos del sistema, a su vez como desventaja, esto da lugar a un menor grado de funcionalidad en comparación con las versiones de software avanzadas compatibles con otros sistemas Linux.
<b>Velocidad</b>	CentOS puede funcionar mucho más rápido que sistemas operativos basados en Linux similares, ya que sólo ejecuta las versiones básicas de software, de esta forma el procesador que ejecuta el sistema Centos no se queda bloqueado al intentar ejecutar muchas aplicaciones diferentes, también una de las ventajas en su velocidad es que programas Centos también son menos propensos a tener problemas de seguridad o bugs bloqueadores, los cuales pueden reducir la velocidad de los procesos del pc.
<b>Confiabilidad</b>	CentOS puede funcionar un ordenador durante bastante tiempo sin necesidad de actualizaciones de sistemas adicionales esto se debe a que las actualizaciones de hardware para Centos se han desarrollado para ser concurrentes con las actualizaciones del sistema Hat Enterprise Linux Centos, el ciclo de la compatibilidad de actualizaciones para CentOS es de unos cinco años, otros sistemas basados en Linux tienen ciclos más cortos de apoyo, a partir de tres años después de unos 18 meses.

### 2.2.1 Servidor Socket

Un socket, es un método para la comunicación entre un programa del cliente y un programa del servidor en una red, el cual tiene una dirección ip y un puerto, si hablamos de la dirección ip este nos permite enrutar los paquetes y el puerto identifica los procesos o aplicaciones que se ejecutan en ese puerto.

Los protocolos TCP y UDP trabajan en la capa de transporte del Modelo TCP/IP, de manera que datos de la capa de aplicación son preparados por esta capa para el procesamiento de estos y ser enviado por la red.

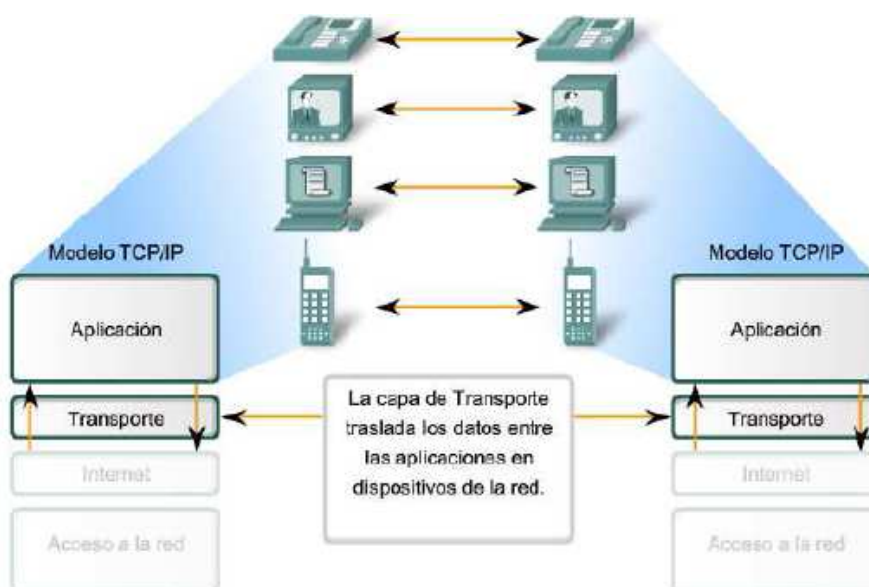


Figura. 5. Función de Capa de Transporte.

Fuente: (Astudillo & Delgado, 2012)

**Protocolo de Control de Transmisión (TCP):** Es un protocolo orientado a la conexión el cual cuenta con características de confiabilidad, secuenciación, control de flujo.

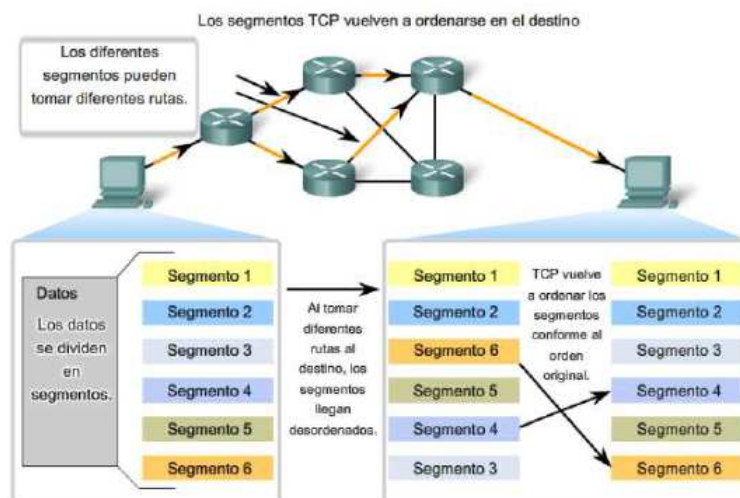
**Confiabilidad:** El protocolo TCP tiene un sistema de acuse de recibo que permite al cliente y al servidor garantizar la recepción mutua de datos, cuando se emite un segmento, se lo vincula a un número de secuencia, con la recepción de un segmento

de datos, la máquina receptora devolverá un segmento de datos donde el indicador ACK esté fijado en 1 que quiere decir un acuse de recibo, acompañado por un número de acuse de recibo que equivale al número de secuencia anterior.

**Secuenciación:** En este caso los paquetes se dividen y enumeran de forma que el receptor pueda ordenarlos y rearmarlos.

**Control de Flujo:** En muchos casos, es posible limitar la cantidad de acuses de recibo con el fin de aliviar el tráfico en la red, esto se logra fijando un número de secuencia después del cual se requiera un acuse de recibo, este número en realidad se guarda en el campo ventana del encabezado TCP/IP.

Este método se llama efectivamente el "el método de la ventana corrediza" porque, en cierta medida, se define una serie de secuencias que no necesitan acuses de recibo y que se desplaza a medida que se reciben los acuses de recibo.



**Figura. 6. Protocolo TCP.**

Fuente: (Astudillo & Delgado, 2012)

**Protocolo de datagramas de Usuario (UDP):** El protocolo de transporte UDP ofrece a las aplicaciones un mecanismo para enviar datagramas IP en bruto encapsulados sin tener que establecer una conexión.

Entre las características de este Protocolo tenemos:

- **Retransmisión:** UDP no admite numeración de los datagramas, factor que, sumado a que tampoco utiliza señales de confirmación de entrega, hace que la garantía de que un paquete llegue a su destino sea mucho menor y no exista una solicitud de retransmisión.
- **Control de Flujo:** Cuando la transferencia de datagramas se establece a través de una red WAN haciendo uso de un protocolo UDP, los enrutadores pueden experimentar momentos de congestión, ya que los mismos interactúan con un servicio de conexión no orientado y el ancho de banda de un enlace WAN en la mayoría de los casos es menor que el ancho de banda de una red LAN.

Un server socket es un sistema de comunicaciones entre procesos en una red la cual tiene como funcionalidad en modo de conexión es declarar un objeto socket estableciendo el puerto que va a escuchar así como aceptar al cliente que requiera su servicio, en la figura 7 se explica el proceso que realiza el servidor socket con un cliente.

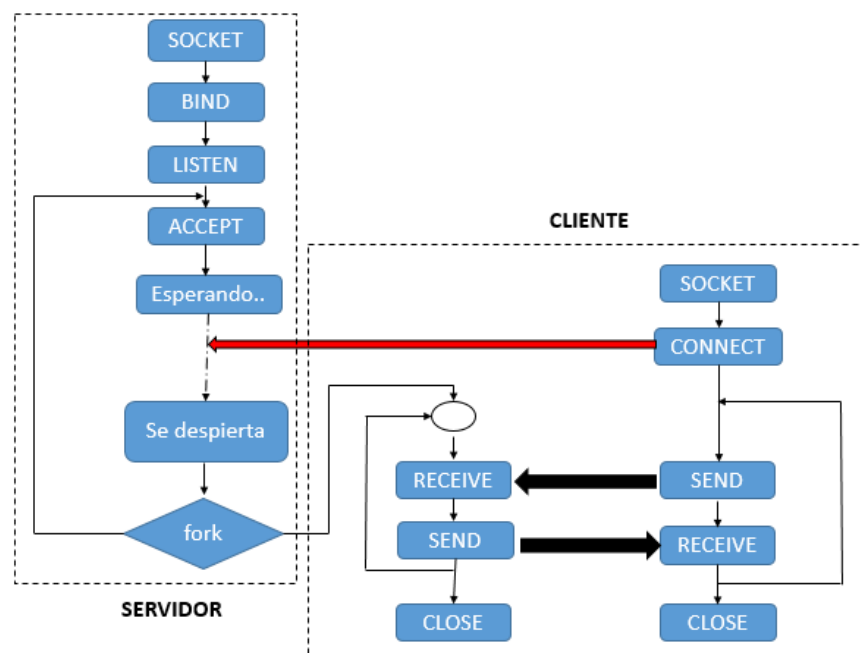


Figura. 7. Comunicación Server Socket.



En la figura 7 los sockets utilizan una serie de primitivas para establecer el punto de comunicación, para conectarse a una maquina remota con determinado puerto, que va estar escuchando hasta que un cliente lo solicite y se pueda hacer una interacción de envío y recepción de información, pero este cuando desee se podrá desconectar de dicho socket.

Para escoger el puerto que va a escuchar el servidor hay que tener en cuenta que no puede utilizar cualquier puerto, ya que IANA<sup>1</sup> asigna puertos para tal o cual proceso o servicio.

- Puertos bien conocidos: Estos son del 0 al 1023 los cuales están reservados para servicios como HTTP, FTP y más.
- Puertos registrados: Estos son del 1024 al 49151 los cuales están reservados para procesos y aplicaciones de usuarios.
- Puertos dinámicos: Estos son del 49152 al 65535 los cuales se asigna de manera dinámica a clientes solicitando una conexión.

### 2.2.2 Interfaz de Usuario vía Web

La interfaz de usuario vía web es la que permite la interacción entre el usuario y los archivos subidos al servidor así como la posición GPS del dispositivo Smartphone, el cual se lo mostrara la posición en la que se encuentre o se encuentra.

Los componentes de la interfaz web se describen a continuación:

- Lenguaje de Programación del lado del servidor (PHP)
- Lenguaje de programación del lado del cliente (JavaScript).
- Tecnología Ajax.

---

<sup>1</sup> Internet Assigned Numbers Authority es la entidad que supervisa la asignación global de direcciones IP y sistemas autónomos. <http://www.internetassignednumbersauthority.org/>

### **2.2.2.1 Lenguaje de Programación del lado del Servidor (PHP)**

Es un lenguaje de programación diseñado para la web que se ejecuta del lado del servidor y se lo muestra al cliente, más concretamente el lenguaje PHP es el procesador de hipertexto que es de código abierto y brinda las siguientes ventajas.

1. Lenguaje totalmente libre y abierto.
2. Curva de aprendizaje muy baja.
3. Los entornos de desarrollo son de rápida y fácil configuración.
4. Fácil despliegue: paquetes totalmente autoinstalables que integran PHP.
5. Fácil acceso a bases de datos.
6. Comunidad muy grande.

### **2.2.2.2 Lenguaje de Programación del lado del Cliente (JavaScript)**

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas, técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos, en otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java, en el plano legal JavaScript es una marca registrada de la empresa “Sun Microsystems”.

### **2.2.2.3 Tecnología Ajax.**

Ajax es Asynchronous JavaScript and XML, está basado en JavaScript y en solicitudes HTTP, es una nueva manera de utilizar los estándares ya existentes, ya que permite realizar acciones en la página sin la necesidad de recargarla.

Con esto Ajax nos permite la comunicación directamente con el servidor, utilizando el objeto XMLHttpRequest, por lo cual intercambian dinámicamente datos por ejemplo en una sección pequeña de la página web.

#### 2.2.2.4 Api de Google Maps

Existen una variedad de APIs geográficas en la web, en este caso utilizaremos google maps por su facilidad al implementarlo, porque cuenta con la información suficiente para poderlo realizar, aparte de sus bondades en la programación por poseer eventos y servicios.

Para montar en una página web el mapa de Google, es necesario contar con una clave única la que te servirá cuando utilices un dominio, los navegadores en cualquier parte lo puedan visualizar.

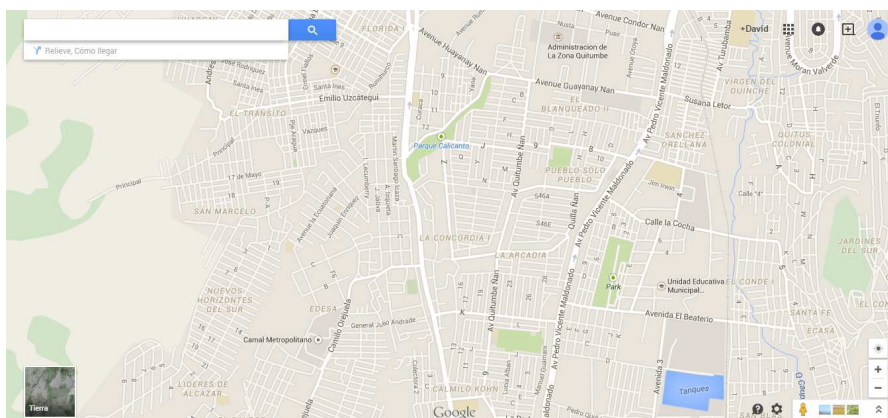


Figura. 8. Visualización del Mapa Google Maps V3

#### 2.2.3 Gestión de Base de Datos

El propósito general de los sistemas de gestión de base de datos es el de manejar la información obtenida de manera clara, sencilla y ordenada tales datos se convertirán en algo muy relevante.

### 2.2.3.1 Base de Datos Relacional

Una base de datos es una entidad en la cual se pueden almacenar datos de manera estructurada, con la menor redundancia posible, diferentes usuarios deben poder utilizar estos datos.

Los datos serán organizados en tablas que constituyen una de las más importantes estructuras enlazadas entre sí de manera lógica, cada tabla incluirá campos o columnas, también las filas o registros.

Cada columna identifica de manera única una fila de una tabla, por ejemplo una columna llamada ciudad hace referencia a que en ese campo se almacenaran nombres de ciudades.

En las bases de datos hay dos tipos de columnas que destacan de otras en la tabla.

- **Clave Primaria:** Es el campo más importante de la tabla, debido a que puede por sí solo identificar a la tabla.
- **Claves Externas:** Es una columna que hace referencia a un campo o una clave primaria de otra tabla.

Mi aplicación maneja una base de datos MySQL, esta base de datos es el sistema de administración de bases de datos relacionales de código abierto más extendida del mundo.

### 2.2.4 Servidor Vsftpd

Linux el demonio FTP o Very Secure FTP Daemon está diseñado desde la base para ser rápido, estable y lo más importante, seguro es por eso su habilidad para manejar grandes números de conexiones de forma eficiente y segura es lo que hace que vsftpd sea el único FTP independiente distribuido.

El modelo de seguridad utilizado por vsftpd tiene tres aspectos principales:

- Clara separación de procesos privilegiados y sin privilegios.
- Procesos separados manejan tareas diferentes y cada uno de estos procesos se ejecuta con los privilegios mínimos requeridos para la tarea
- Las tareas que requieren altos privilegios son manejadas por procesos con los mínimos privilegios necesarios. Influenciando las compatibilidades encontradas en la biblioteca libcap, las tareas que usualmente requieren privilegios de súper usuario se pueden ejecutar de forma más segura desde un proceso menos privilegiado.

La mayoría de los procesos se ejecutan enjaulados en un ambiente “Chroot”<sup>2</sup> siempre que sea posible, se cambia la raíz de los procesos al directorio compartido, este directorio se considera luego como la jaula “Chroot”.

Esto previene actividades maliciosas de cualquier hacker potencial en algún directorio que no están por debajo del nuevo directorio root, es por eso que el uso de estas prácticas de seguridad tiene el efecto siguiente en como vsftpd trata con las peticiones:

La mayor parte de la interacción con clientes FTP la manejan procesos hijo sin privilegios en una jaula chroot, debido a que estos procesos hijo no tienen privilegios y solamente tienen acceso al directorio que está siendo compartido, cualquier proceso fallido solamente permitir al atacante acceder a los archivos compartidos.

---

<sup>2</sup> Es una operación que invoca un proceso, cambiando para este y sus hijos el directorio raíz del sistema consultado de <https://www.jcea.es/artic/chroot.htm>

## CAPÍTULO III

### 3. DISEÑO DEL SISTEMA

#### 3.1 SISTEMA REQUERIDO

Los proyecto se divide en dos partes la primera es el cliente en un Smartphone y la otra el servidor como se puede ver en la figura 9.

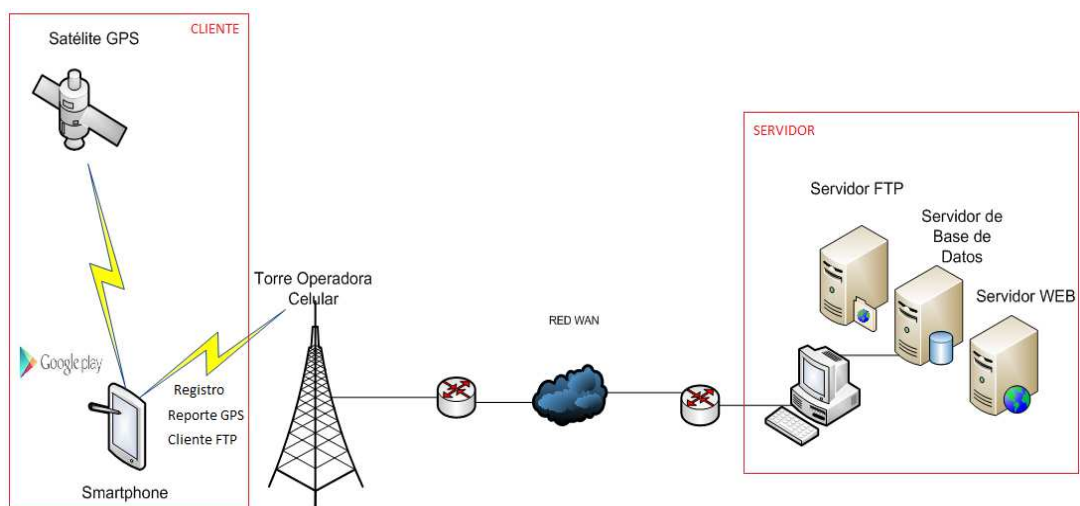


Figura. 9. Sistema Requerido

El cliente instala la aplicación en primera instancia y procede a registrarse llenando un formulario que le proporciona la aplicación, posteriormente el cliente escoge un

Nick y Contraseña para acceder vía web para observar sus datos en el celular y su ubicación.

El cliente Smartphone después de realizar la instalación, reporta su posición cada periodo de tiempo establecido por el mismo cliente en el momento de instalación de la aplicación, así como el periodo de tiempo para su servicio “FTP”<sup>3</sup> el cual va a garantizar su información refiriéndose así a sus contactos con extensión “Vcf”<sup>4</sup>, fotos y una carpeta personal que el cliente escoja.

El servidor tiene la capacidad de estar escuchando de manera continua alguna petición de los clientes ya sea para registro o reporte de posición del Smartphone así también el “Servidor web Http Apache” ayuda con la consulta de los clientes a su información personal via web , el cual consiste en ingresar a un sitio web y hacer un “Login”<sup>5</sup> con el ingreso de usuario y contraseña única, de esta manera se accede a una interfaz dinámica de historial de ubicaciones en el mapa, así también el servicio ftp el cual garantiza seguridad y confidencialidad de sus datos los cuales se pueden descargar por el cliente.

## **3.2 DIAGRAMAS**

### **3.2.1 Diagrama de Clases Cliente**

Un diagrama de actividades no es más que un caso especial de un diagrama de estados, en el que todos los estados o en su gran mayoría son actividades, en este caso este diagrama de actividades muestra el flujo de control entre una serie de actividades, es por eso que en la figura 10 se observa cada una de las clases implementadas en el presente proyecto.

---

<sup>3</sup> File Transfer Protocol es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor.

<sup>4</sup> vCard es un formato estándar para el intercambio de información personal, específicamente tarjetas personales electrónicas

<sup>5</sup> En el ámbito de seguridad informática, login o logon es el proceso mediante el cual se controla el acceso individual a un sistema informático mediante la identificación del usuario utilizando credenciales.

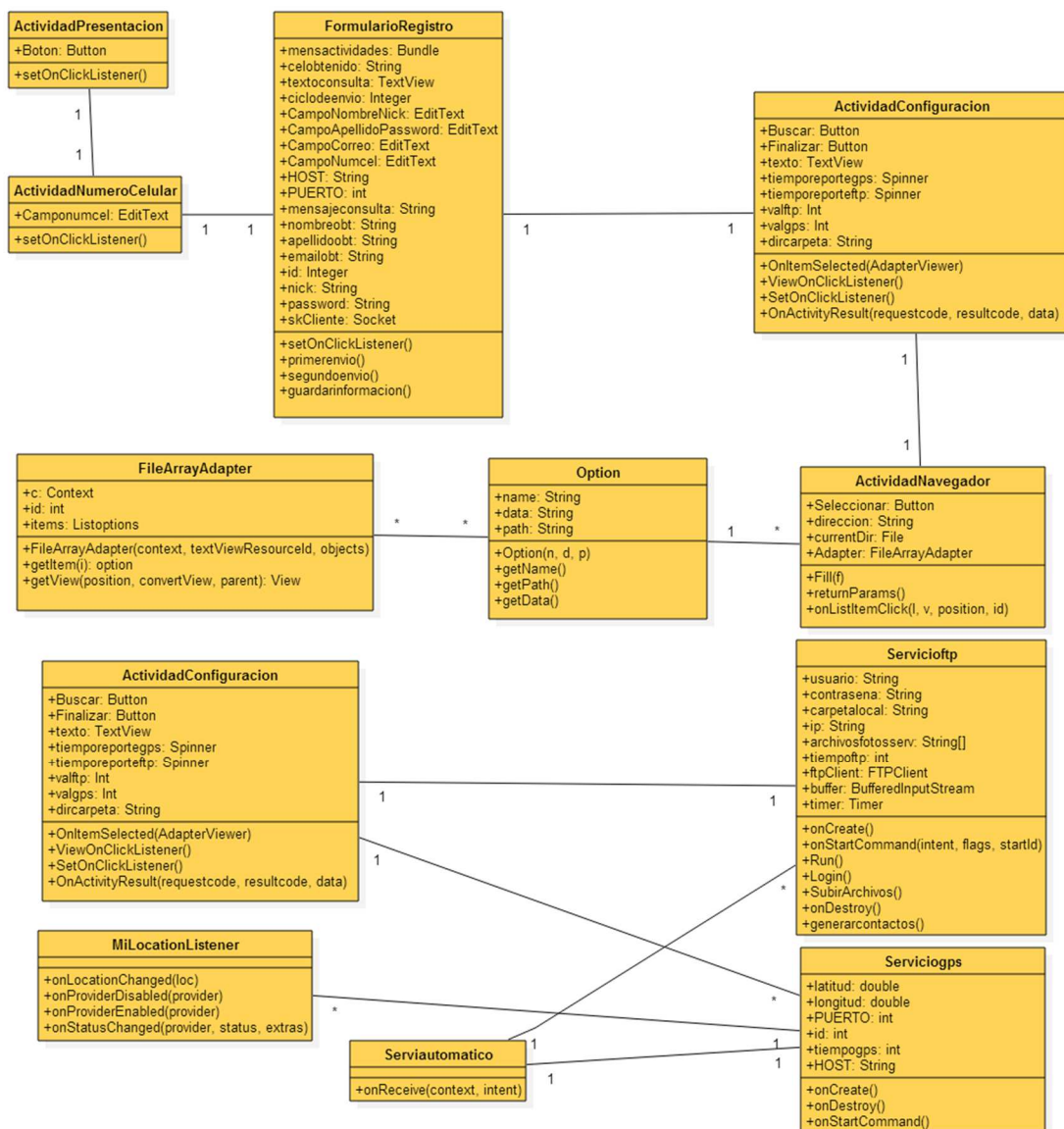


Figura. 10. Diagrama UML de Aplicación Cliente

En la figura 10 se observa el orden de las clases cliente que se van creando a partir de la interacción del cliente y el servidor, en su primera instancia con el registro del cliente y posteriormente con el proceso de lanzamiento de servicios automáticos.

En primer lugar la “ActividadPresentación”, es la primera clase que se ocupa cuando se inicia la aplicación, esta clase tiene como atributo una imagen de fondo y un botón el cual confirma si usted desea continuar con el registro.



La clase “ActividadNumeroCelular”, se crea después para recibir el número de celular en un campo de texto y llevar este dato en adelante.

Posteriormente se crea la clase “FormularioRegistro”, en la cual se definen constantes y atributos, en lo que se refiere a constantes se especifica la ip del host, así como el puerto al que se enviara la información, en cambio en los atributos se contó con cuatro campos de texto, los cuales llevaran información como nombre, apellido, correo electrónico, número de celular, Nick y contraseña.

La Clase “ActividadConfiguración”, su misión es la de mostrar dos listados de tiempo en cual el usuario escogerá el lapso de tiempo que desee para repostes GPS, así como cada qué periodo de tiempo para subir sus archivos nuevos al servidor ftp.

La “clase ActividadNavegador”, se complementa con la Clase “Actividad configuración” porque es llamada por el usuario para escoger una dirección de la memoria externa sd, esta será una carpeta que su contenido será enviado cada tanto al servidor ftp

La Clase “FileArrayAdapter”, necesita el listado o “ListView” que le pasemos un “Array” de tipo texto donde iremos guardando los nombres de los ficheros, estableceremos un ciclo donde separaremos los directorios de los ficheros, a la vez que extraeremos el nombre del directorio.

La clase “Serviciogps”, es el encargado de reportar cada cierto tiempo, suposición enviando esta información a través de un socket, esta clase por ser de tipo servicio, tiene la particularidad que hasta que no sea destruida seguirá en funcionamiento en su propio hilo, lo que quiere decir que si la aplicación ya finaliza el registro esta seguirá ejecutándose.

La clase “Servicioftp” es la encargada de enviar archivos que consten en las carpetas de fotos, también una carpeta personal la cual el usuario escogió en el momento de su registro, esta clase tiene acceso a la información de un Id único que es guardado en memoria interna en el momento del registro, por tal motivo puede

conectarse con sus credenciales al servidor FTP y tiene la posibilidad de enviar los archivos de bases de datos de WhatsApp que se generan, con la condición que el Smartphone este ruteado.

La clase “ServiAutomatico” es la encargada de lanzar los servicios GPS y FTP si se reinicia el Smartphone.

### 3.2.2 Diagrama Entidad Relación de Base de Datos Servidor

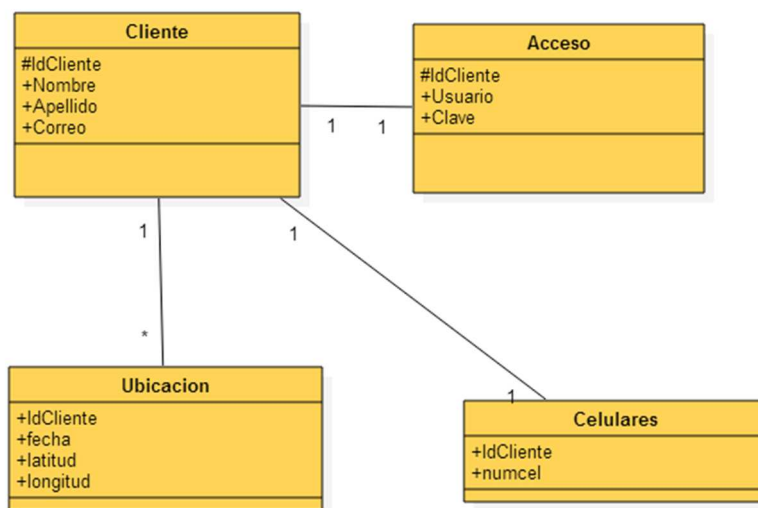


Figura. 11. Diagrama UML de Base de Datos

En la figura 11 se observa el diagrama entidad relación de la base de datos del presente proyecto, dicho esto la base de datos se llama baseregistro la cual cuenta con cuatro tablas, estas se constituyen de la siguiente manera:

**Cliente:** Esta tabla cuenta con un idCliente el cual lleva una clave primaria ya que es única para cada cliente, también se cuenta con el Nombre, Apellido, y Correo.

**Acceso:** Esta tabla fue generada para identificaciones para el ingreso por vía web, ya que cuando el usuario realice el “Login” vía web, el servidor revise los datos en la tabla acceso.

**Celulares:** Esta tabla cuenta con idCliente y el número el cual nos ayudara a verificar en un principio si es o no cliente de esta aplicación.

**Ubicación:** Esta tabla comparte con todos los clientes con su id único, fecha en la que se reportó su posición de latitud y longitud, pero no hay que alarmarse porque cuando se realiza consulta del usuario via web, el mecanismo de consulta se la hace con php con su id y contraseña y solo mostrara datos propios.

### 3.2.3 Diagrama de Clases Servidor Socket Registro y Gps

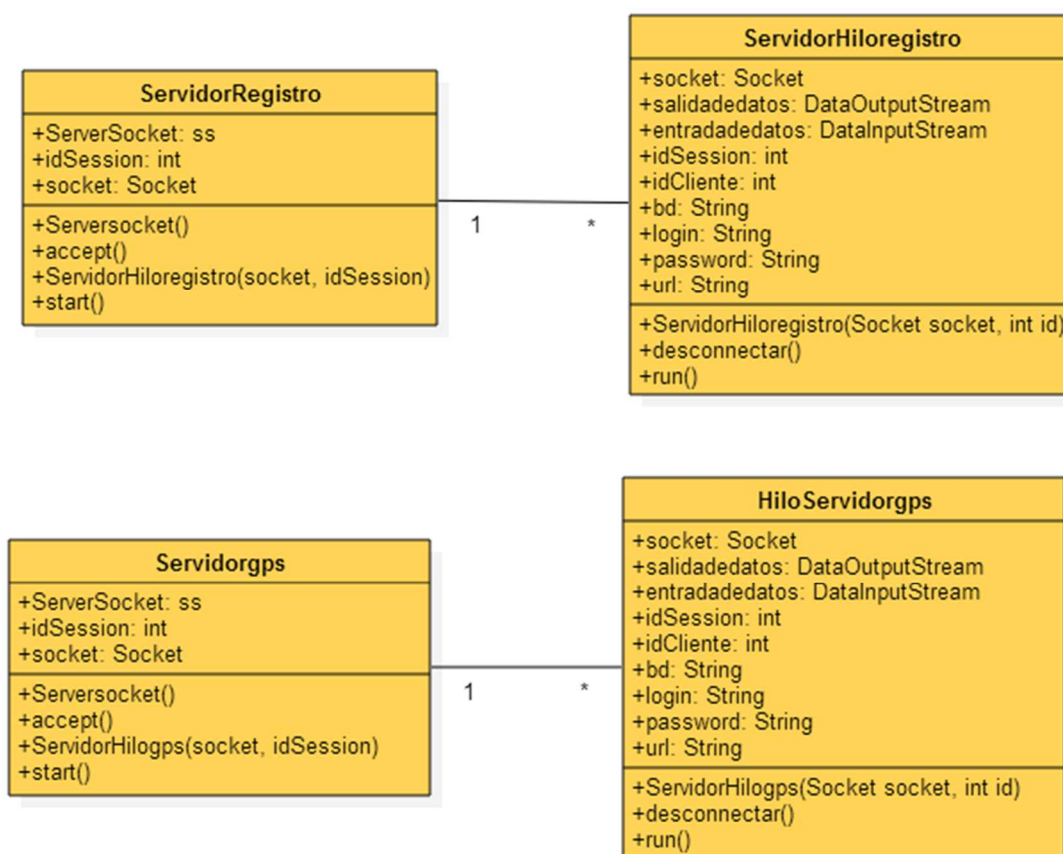


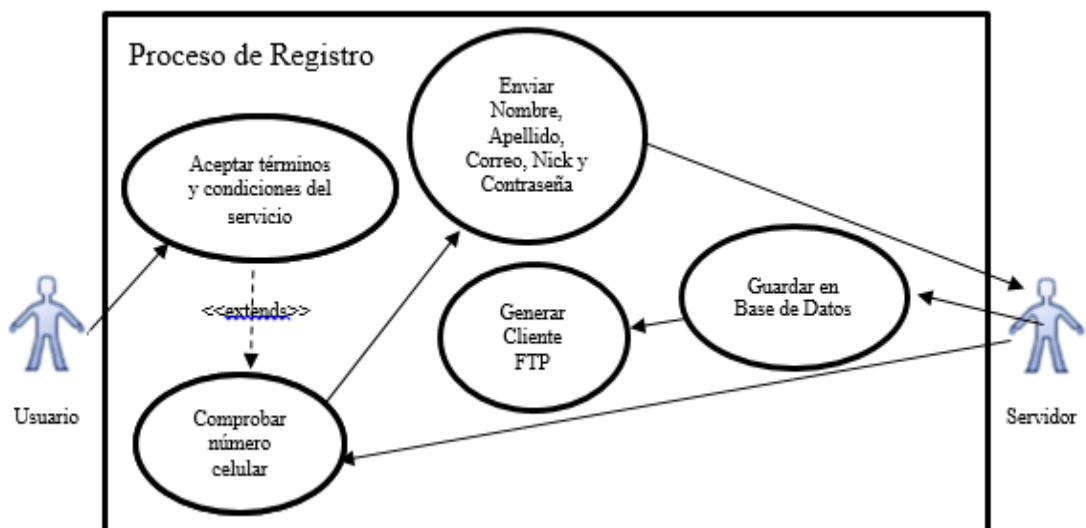
Figura. 12. Diagrama UML de Servidor Socket GPS y Registro.

En la figura 12 se observa la clase “ServidorRegistro” y “Servidorgps”, estas clases son creadas como un servidor socket que es un sistema de comunicaciones entre procesos en una red, la cual tiene como funcionalidad en modo de conexión es declarar un objeto socket estableciendo el puerto que va a escuchar así como aceptar al cliente

que requiera su servicio, pero también generan dos clases paralelas llamadas `HiloServidorgps` y `ServidorHiloregistro` lo cual permite ejecutar varios procesos a la vez, esto sería idóneo si muchos clientes realizan una conexión al mismo tiempo, es decir, de forma concurrente y por tanto eso logra que el servicio se ejecute de manera más eficiente.

### 3.2.4 Diagramas de casos de uso

La interacción del usuario con los procesos del sistema se muestra a continuación, donde se indica todas las acciones que realizara el usuario y la respuesta a dicha acción, para este proyecto se ha fijado cuatro casos de uso.



**Figura. 13. Diagrama Caso de Uso Registro**

En la figura 13 se observa el caso de uso en la etapa de registro de un usuario enviando información necesaria y a su vez el servidor después de recibir esa información, tiene la capacidad de generar una carpeta única para cada cliente.

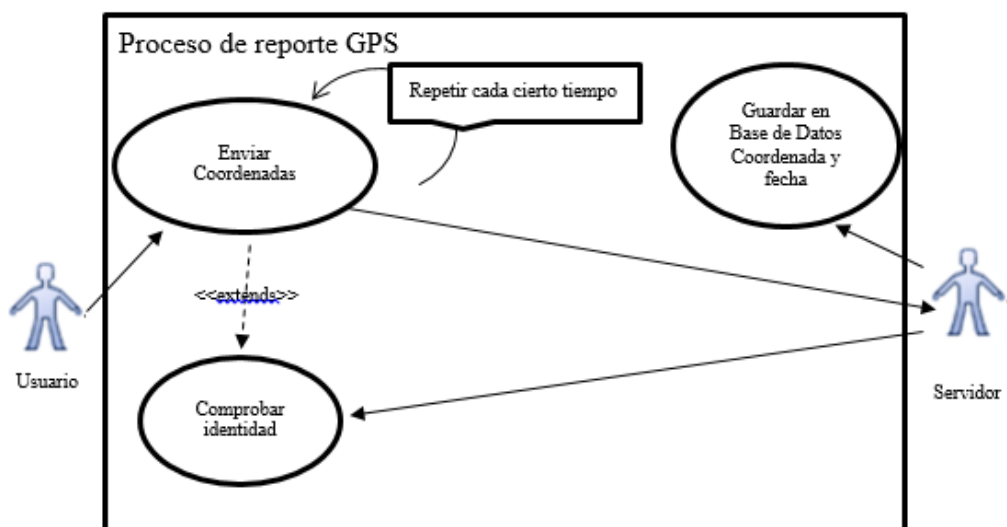


Figura. 14. Diagrama Caso de Uso GPS

En la figura 14 se observa el caso de uso en la etapa de reporte GPS de un usuario enviando información de ubicación al servidor después de un determinado tiempo especificado por el usuario.

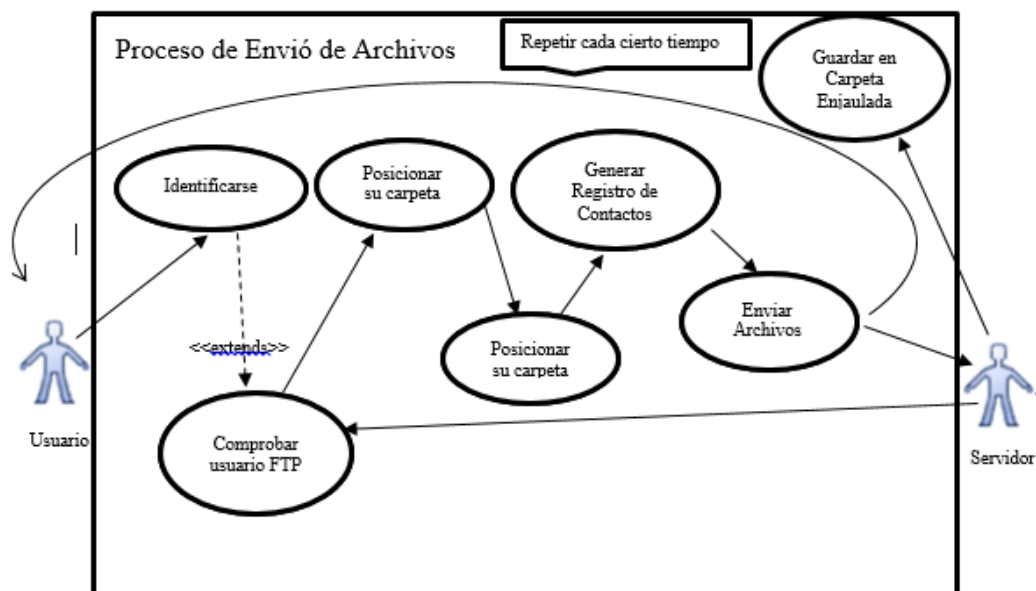
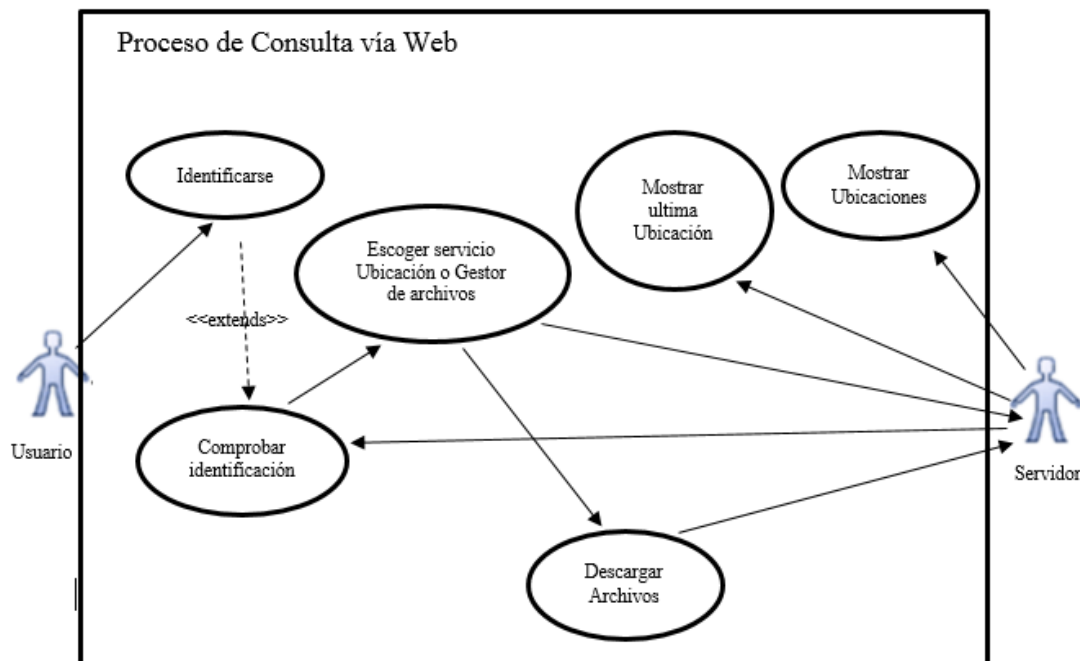


Figura. 15. Diagrama Caso de Uso Envío de Archivos

En la figura 15 se observa el caso de uso cuando el usuario está haciendo uso del servicio ftp para subir sus archivos al servidor, este caso de uso interactúa el usuario Smartphone con el servidor de manera automática, en primera instancia se identifica

si se logra loguear podrá tener acceso a su carpeta y enviar información refiriéndome a los ficheros.



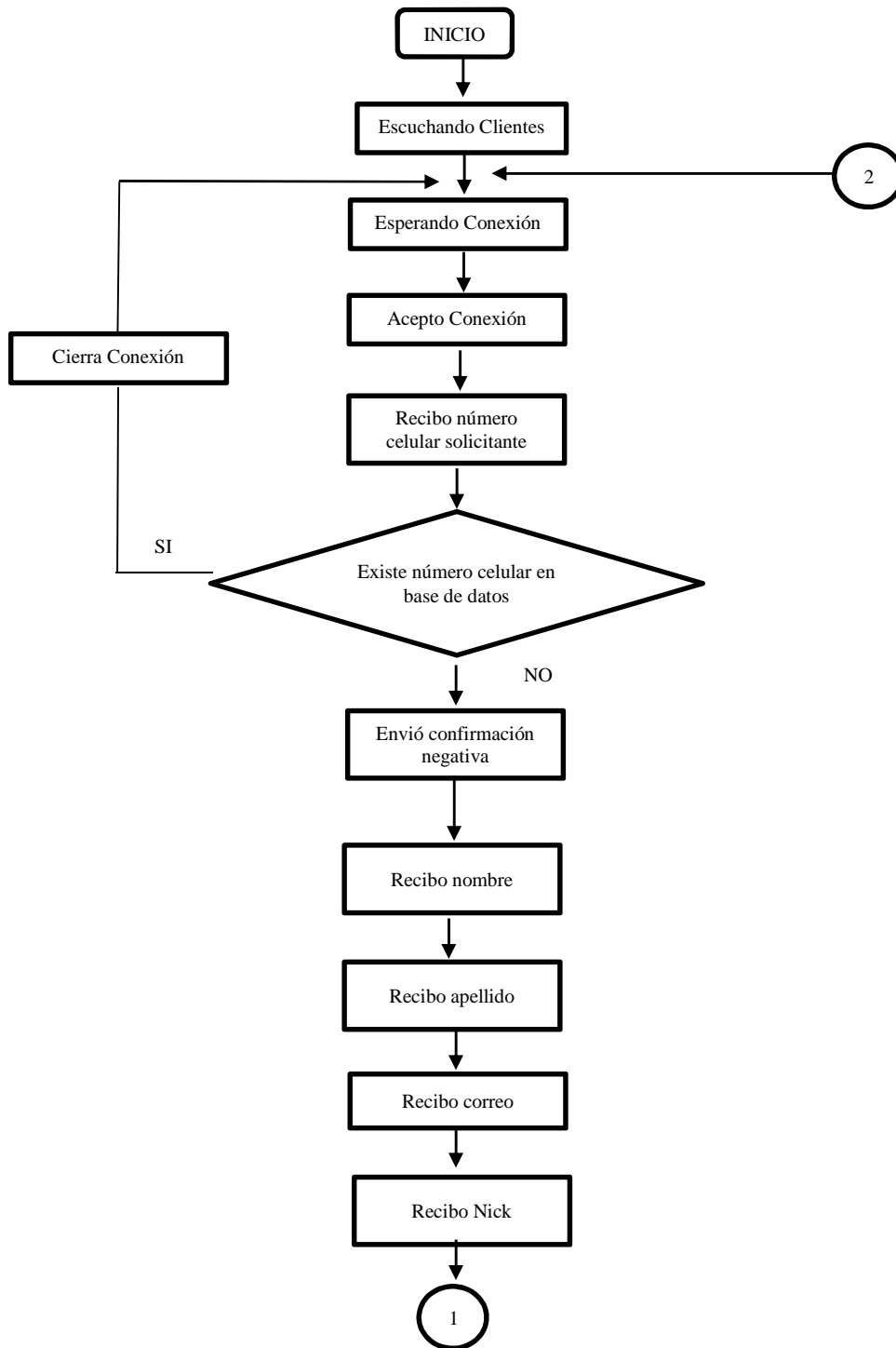
**Figura. 16. Diagrama Caso de Uso Consultas vía Web**

En la figura 16 se observa el caso de uso cuando el usuario está haciendo uso del servicio web para revisar su última posición así como un historial pormenorizado de sus instancias en tal o cual punto con hora y fecha exacta, también cuando se realiza la consulta de ficheros subidos al servidor.

### 3.2.5 Diagrama de Flujo de Servidor Socket Registro

En la figura 17 se observa el diagrama de flujo del servidor socket para el registro, el cual genera un socket que escucha por un puerto de manera ininterrumpida, las solicitudes de registro, después de que el cliente hace la solicitud del servicio el servidor lo acepta esperando en primera instancia el número de celular, si este no consta en la base de datos del servidor envía un mensaje de respuesta diciendo que ese número de celular ya cuenta con dicho servicio o no, entonces en el caso que el número de celular no sea parte de la base de datos enviara un mensaje de que se puede registrar

y espera el resto de datos como son nombre, apellido, correo electrónico, nick y contraseña, posteriormente guarda la información en la base de datos y envía una identificación única llamada idCliente y llama a un archivo bash de linux para generar un cliente con las credenciales de Nick y contraseña.



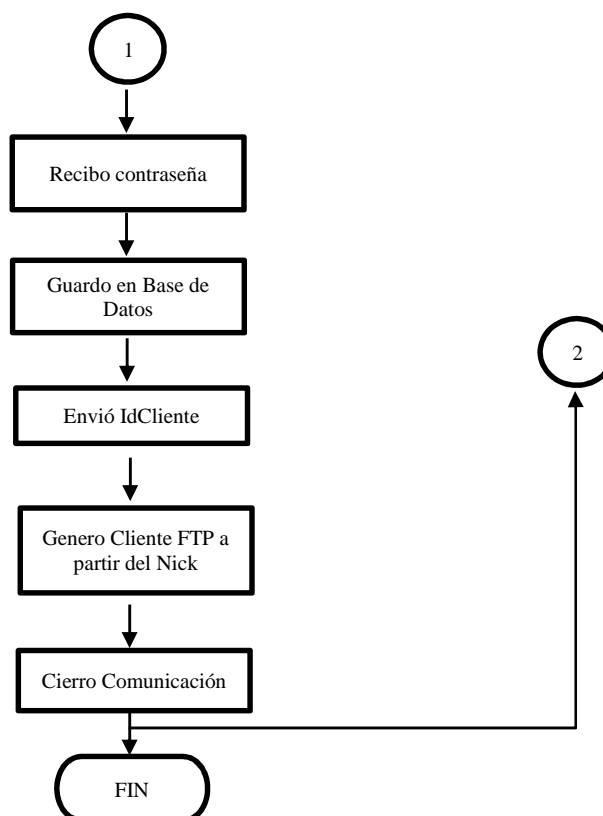
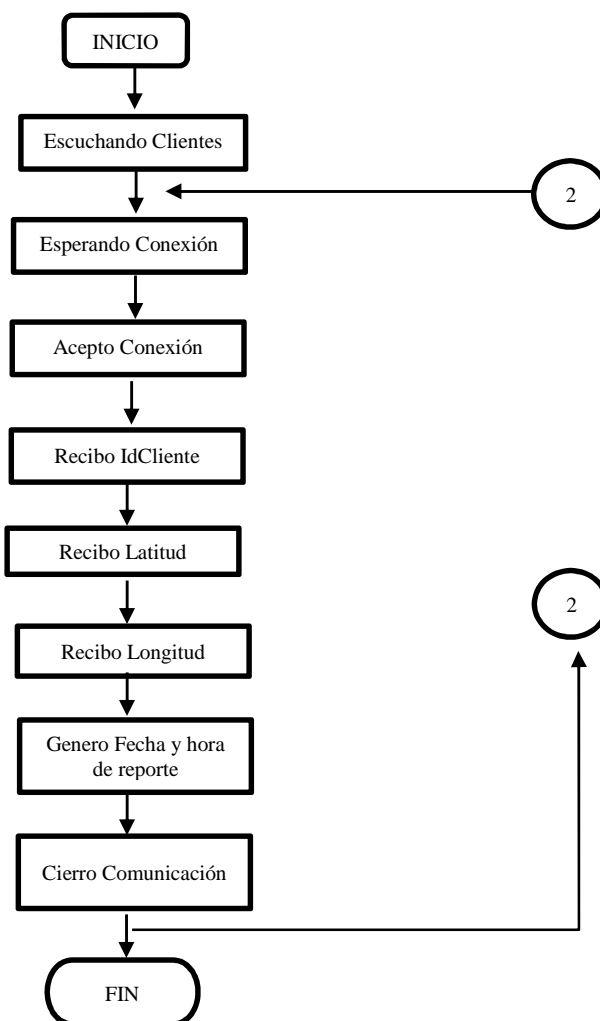


Figura. 17. Diagrama de Flujo Servidor Socket Registro

### 3.2.6 Diagrama de Flujo de Servidor Socket Gps

En la figura 18 se observa el diagrama de flujo del servidor socket gps, el cual genera un socket que escucha por un puerto de manera ininterrumpida, las solicitudes de reporte gps, después de que el cliente hace la solicitud del servicio el servidor lo acepta esperando en primera instancia la identificación única que tiene cada cliente, si este consta en la base de datos del servidor espera la posición de longitud y latitud del cliente y genera la fecha de llegada del reporte.





**Figura. 18. Diagrama de Flujo Servidor Socket GPS**

### 3.2.7 Diagrama de Flujo de Formulario Registro

En la figura 19 se observa el diagrama de flujo del formulario registro en el cliente, el cual genera varias actividades las cuales envían información del cliente través de una conexión socket apuntado hacia el servidor socket, la información que se proporciona es nombre, apellido, correo electrónico, nick y contraseña, después se espera una identificación única y la guarda en la memoria interna del teléfono que es de carácter privado la cual solo podrá ser leída por la propia aplicación, finalmente se escoge un tiempo para el reporte gps y otro tiempo para la actualización de los archivos que constan dentro del teléfono, también se escoge la dirección de una carpeta en específico que se sube al servidor FTP .

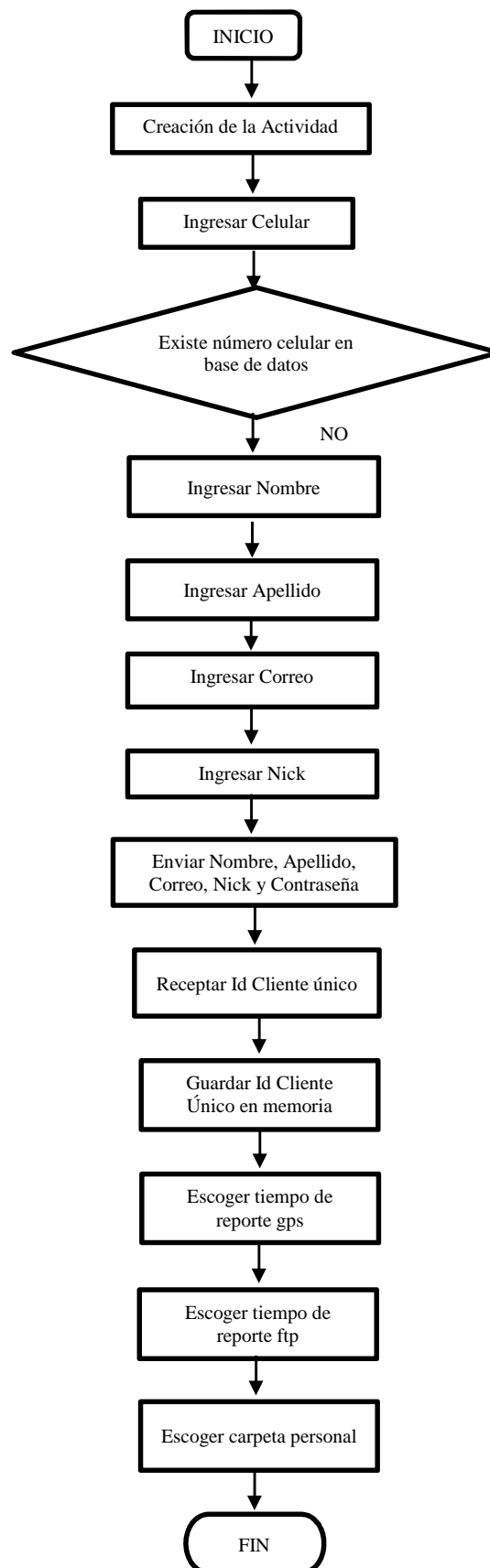
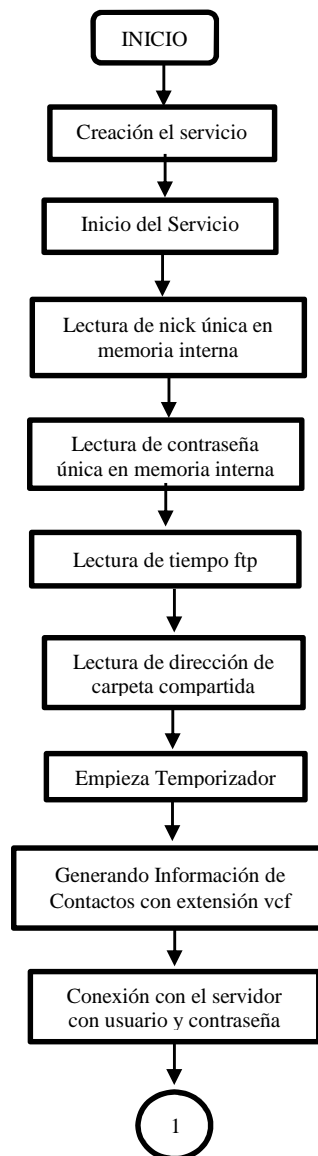
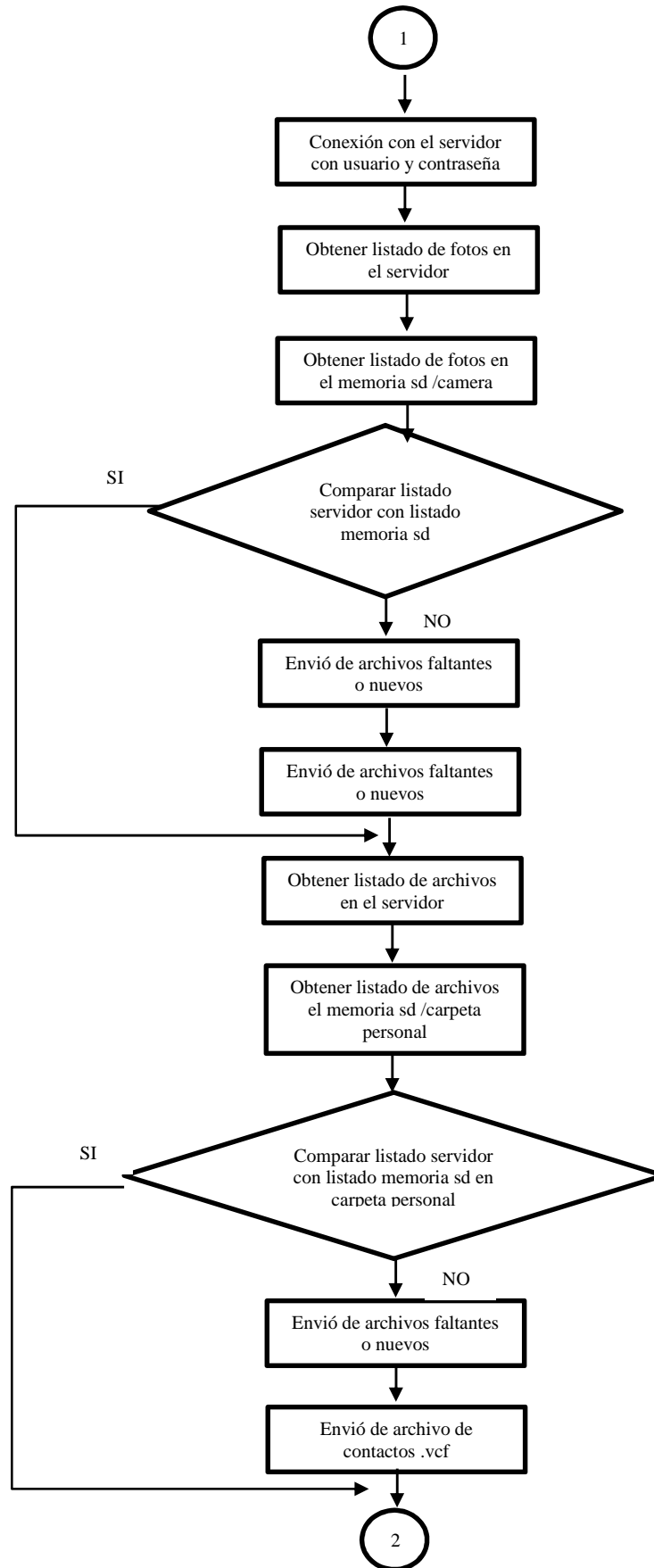


Figura. 19. Diagrama de Flujo de Formulario Registro

### 3.2.8 Diagrama de Flujo de Servicio Ftp Cliente

En la figura 20 se observa el diagrama de flujo del servicio FTP cliente, el cual inicia el servicio leyendo en memoria interna del Smartphone los parámetros Nick, contraseña y tiempo ftp que se proporciona en la instalación de la aplicación, después entra en funcionamiento un reloj el cual hace que este servicio se repita cada tanto tiempo como lo dice el tiempo ftp, a partir de ahí el Smartphone se conecta al servidor ftp facilitando las credenciales de Nick y contraseña enviando los archivos de contactos, base de datos WhatsApp, fotos y carpeta personal.





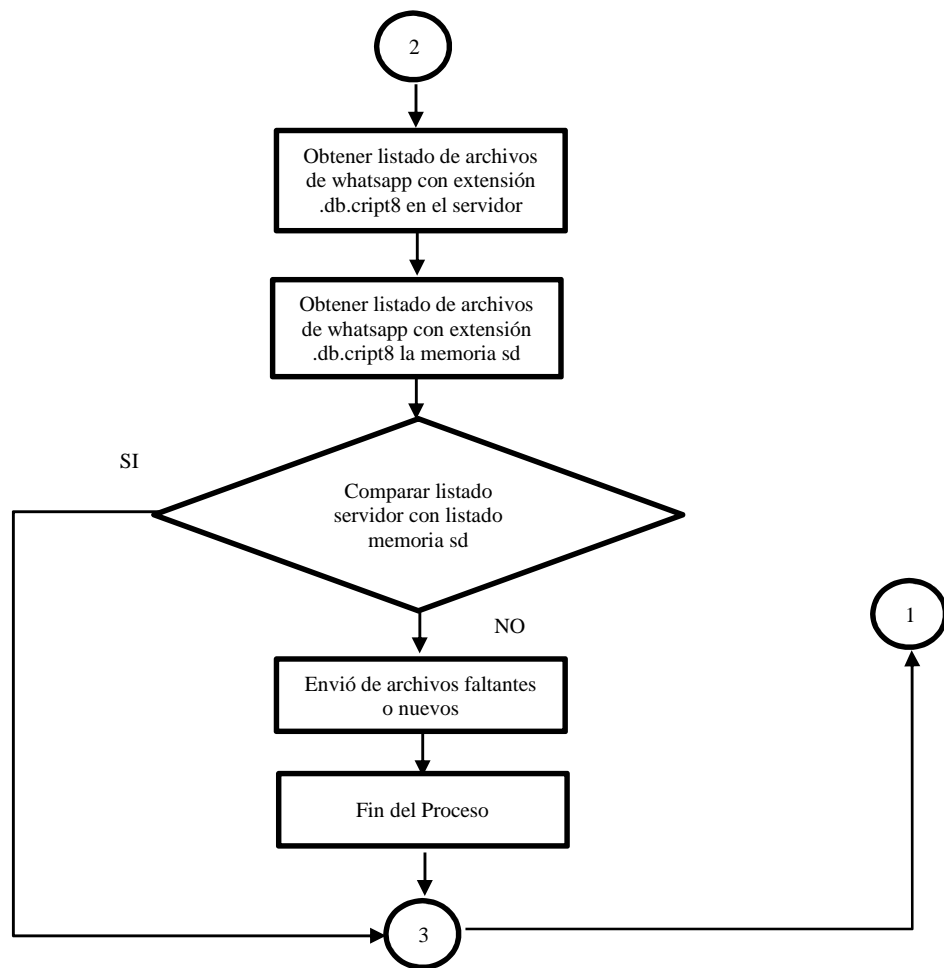


Figura. 20. Diagrama de Flujo de Servicio FTP Cliente

### 3.2.9 Diagrama de Flujo de Servicio Gps Cliente

En la figura 21 se observa el diagrama de flujo del servicio GPS cliente, el cual inicia el servicio leyendo en memoria interna del Smartphone los parámetros identificación única y tiempo gps que se proporciona en la instalación de la aplicación, después entra en funcionamiento el evento “LocationListener” el cual actualiza la posición gps cada tiempo gps y envía la posición de longitud y latitud a través de un socket.

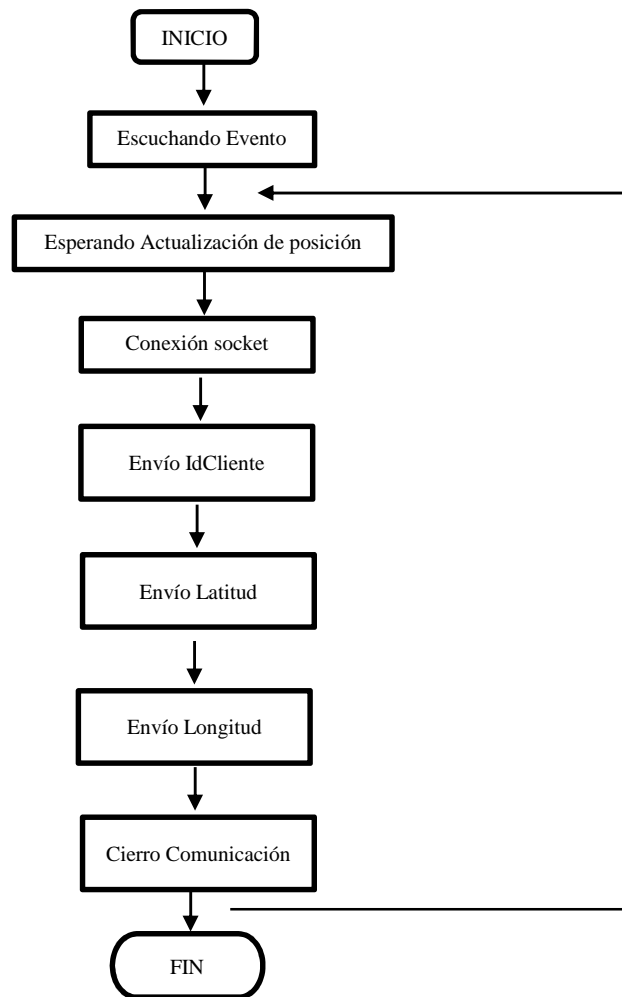


Figura. 21. Diagrama de Flujo de Servicio GPS Cliente

### 3.3 DISEÑO DE LA APLICACIÓN PARA EL CLIENTE

El diseño de la interfaz de usuario para el registro del Smartphone se esquematiza utilizando fragmentos ordenados, el cual se lo ha delimitado de la siguiente manera:



**Figura. 22. Áreas de la Interfaz Android**

El área denominada Título, es donde se define el nombre de la actividad que se está realizando, el nombre de la actividad se la puede editar en el archivo android “Manifest xml”<sup>6</sup> de la aplicación, esta capa se la puede definir dependiendo la actividad.

Esta área denominada Cabecera, se puede especificar el nombre de la actividad así como una imagen de cabecera.

---

<sup>6</sup> Es un archivo de configuración donde podemos aplicar las configuraciones básicas de nuestra aplicación.

El área denominada cuerpo, consta del cuerpo de las actividades donde se especificara mensajes de confirmación así como campos de texto para ingreso de formularios o envió de mensajes del servidor.

El área denominada Pie, está delimitada para ubicar los botones que permitirá la interacción cliente servidor en la etapa de registro.

Después de especificar las áreas para el diseño de la interfaz en android, como se puede observar en la figura 23.



**Figura. 23. Vista de Presentación de La Aplicación Android**



### 3.4 DISEÑO DE LA APLICACIÓN PARA EL SERVIDOR

#### 3.4.1 DISEÑO DE LA INTERFAZ WEB

Una página web está estructurada a partir de etiquetas HTML, pero se lo hacía con tablas pero por su complejidad y errores que se generaba se ha optado por contenedores “Div”<sup>7</sup> flotantes trabajando a la par con hojas de estilo CSS.

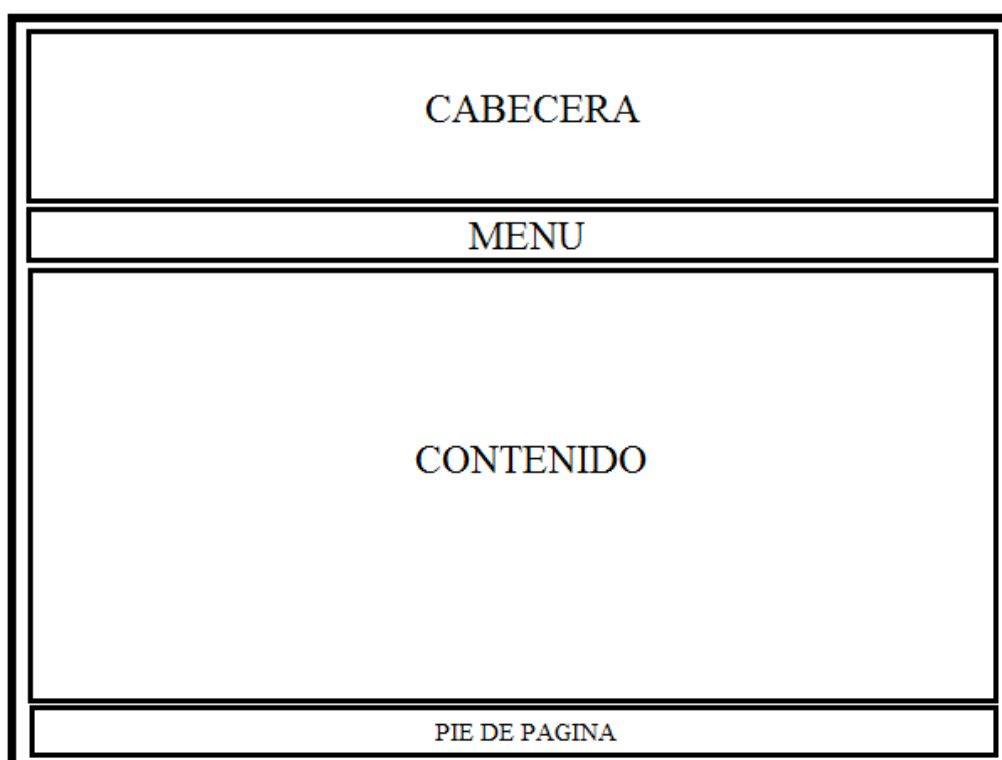


Figura. 24. Esquema de Interfaz Web

La estructura de cada una de las páginas web está dividida en seis capas:

- **Cabecera** : En esta capa se especifica información del Usuario el cual este utilizando este servicio.
- **Menú** : En esta capa se especifica las diferentes opciones o botones.

---

<sup>7</sup> La etiqueta Div se emplea para definir un bloque de contenido o sección de la página para aplicarle diferentes estilos.

- **Body** : En esta capa se encuentra las siguientes capas.
- **Contenido** : En esta capa se encuentra la información inherente al cliente como su ubicación en el mapa así como sus archivos personales.
- **Pie de Página** : En esta capa se especifica quien realizo el trabajo

Después de haber delimitado las capas se obtiene una vista de la página web como muestra la figura 25.

Localización Segura

Salir BIENVENIDO A ESTA SESION me

Fecha	Latitud	Longitud
2015-01-29	-0.184407	-78.484354
2014-08-2	-0.224472	-78.521995
2014-08-2	-0.283265	-78.549031
2014-08-2	-0.299377	-78.550818
2014-08-2	-0.304978	-78.553242
2014-08-2	-0.308827	-78.553667
2014-08-2	-0.315537	-78.443717
2014-08-2	-0.315537	-78.443717
2014-08-2	-0.197951	-78.503798
2014-08-2	-0.224472	-78.521995
2014-08-2	-0.283265	-78.549031
2014-08-2	-0.299377	-78.550818

Desde: 2009-02-08

Hasta: 2015-02-10

consultar

Localización Segura - Hecha por Tito Torres Racines - Proyecto de Tesis ESPE

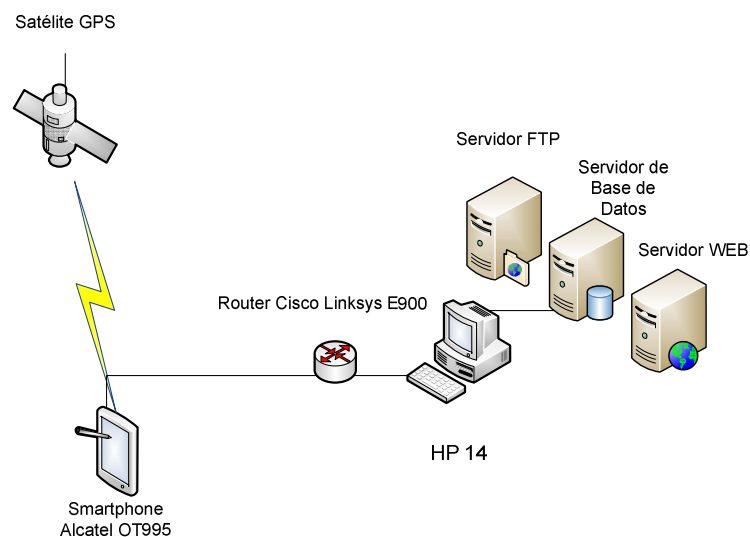
Figura. 25. Página Web de historial de ubicación

## CAPÍTULO IV

### 4. IMPLEMENTACION Y PRUEBAS

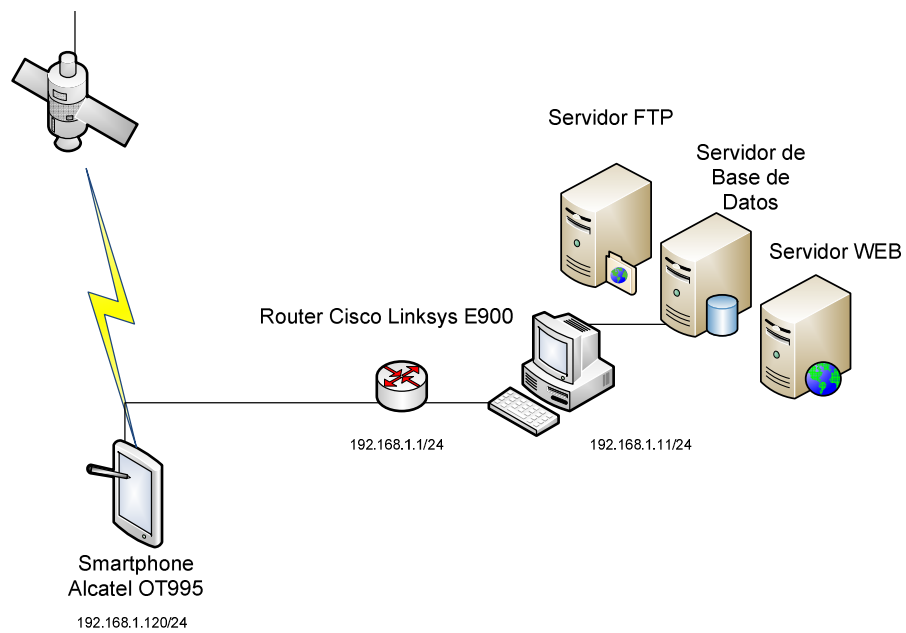
#### 4.1 ESCENARIO RED DE AREA LOCAL

El escenario planteado en la figura 26 muestra una red LAN la cual cuenta en su dominio con tres servidores los cuales están bajo el sistema operativo de Centos 6.5 en la modalidad de máquina virtual, también el Smartphone está conectado a la red LAN de dicho diagrama.



**Figura. 26** Topología de la red LAN

A continuación se realizó la etapa de direccionamiento la cual se puede ver en la figura 27 y tabla 5.



**Figura. 27. Topología de red LAN con Direccionamiento**

**Tabla 5**

**Tabla de direccionamiento LAN**

Dispositivo	Interfaz	Dirección IP	Máscara	Tipo de Red
<b>Router Cisco Linksys E900</b>	Ethe1	186.4.217.25		WAN
	Ethe2	192.168.1.1	255.255.255.0	LAN
<b>Smartphone Alcatel Touch 995</b>	Ethe1	192.168.1.20	255.255.255.0	
<b>HP14</b>	Ethe1	192.168.1.11	255.255.255.0	LAN

Se realizó la implementación de la aplicación Android en un Smartphone, el cual tiene las siguientes características como se puede ver en la tabla 6

**Tabla 6**  
**Características de Alcatel OT-995**

<b>GENERAL</b>	<b>Red</b>	<b>GSM 850 / 900 / 1800 / 1900 - HSDPA 900 / 2100</b>
<b>TAMAÑO</b>	Dimensiones	128 x 68 x 9.8 mm
	Peso	124 g
<b>DISPLAY</b>	Tipo	TFT touchscreen capacitivo, 16M colores
	Tamaño	480 x 800 pixels, 4.3 pulgadas
<b>MEMORIA</b>		- Sensor acelerómetro para auto rotación - Controles sensibles al tacto - Soporte Multitouch - Sensor de proximidad para auto apagado
	Agenda telefónica	Entradas y campos prácticamente ilimitados, Foto de llamada
	Registro de llamadas	Prácticamente ilimitado
	Slot de tarjeta	microSD hasta 32GB
		- 2GB ROM, 512MB RAM - Procesador Qualcomm MSM8255T Snapdragon 1.4GHz, GPU Adreno 205
<b>CARACTERÍSTICAS</b>	GPRS	Clase 12 (4+1/3+2/2+3/1+4 slots)
	Velocidad de datos	32 - 48 kbps
	OS	Android OS, v2.3 Gingerbread
	Mensajería	SMS, MMS, Email, Push Email, IM
	Navegador	HTML
<b>BATERÍA</b>		Standard, Li-Ion 1500 mAh
	Stand-by	Hasta 360 h (2G) / Hasta 288 h (3G)
	Tiempo de conservación	Hasta 14 h (2G) / Hasta 7 h (3G)

Las características del router Cisco Linksys E900 con el cual se utilizó se detalla en tabla 7.

**Tabla 7**  
**Características Router Cisco Linksys E900**

<b>Modelo:</b>	<b>Linksys E900</b>
<b>Tecnología:</b>	Inalámbrica N
<b>Bandas:</b>	2,4 GHz
<b>Transmisión/recepción:</b>	2 x 2
<b>Antenas:</b>	2 (internas)
<b>Puerto USB:</b>	No
<b>Puertos x velocidad:</b>	4 x Ethernet
<b>Compatibilidad con sistemas operativos:</b>	Windows, Mac
<b>Requisitos mínimos del sistema:</b>	<p><b>PC:</b> equipo con Wi-Fi y unidad de CD o DVD, con Windows XP SP3, Windows Vista SP1 o versión posterior, Windows 7, o Windows 8</p> <p><b>Mac:</b> equipo con Wi-Fi y unidad de CD o DVD, con OS X Leopard 10.5 o Snow Leopard 10.6</p>
<b>Requisitos del navegador de Internet:</b>	Internet Explorer 7, Safari 4, Firefox 3 o versión posterior para la configuración opcional en navegador

Después de comprobar la conectividad de la topología mostrada en la figura 27, se procedió a ejecutar la aplicación android la cual se llama “Localización Segura”<sup>8</sup>, para esto la aplicación Localización Segura fue desarrollada en el entorno de desarrollo Android Studio con la compilación USB en tiempo real, como se puede ver en el Anexo del presente proyecto se indica la correcta configuración para dicho propósito.

<sup>8</sup> Nombre de la Aplicación en Android para la gestión remota y Geolocalización.

### 4.1.1. Pruebas de Software en el Servidor

Como se observa en la figura 28, el proceso de registro de cada uno de los clientes llega al servidor, en un principio con la conexión socket como se ve en la línea número 1, después el servidor recibe los datos personales como es el celular, nombre, apellido correo electrónico y en la parte final recibe el Nick y contraseña como en la línea número 2, posteriormente con dichos datos se realiza el proceso de java el cual crea el usuario ftp llamando a un punto “bash”<sup>9</sup> el cual genera un usuario con esos datos como se ve en la línea 3.

```

run:
Iniciando servidor para Registro... [OK]
Nueva conexión entrante para registro: Socket[addr=/192.168.1.113,port=49232,localport=1024]
El cliente con idSesion 0 saluda
Su celular dice que es : 0993577749
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
No esta Registrado este numero de celular en nuestra base de datos
Su celular dice que es : 0993577749
Su nombre dice que es : David
Su apellido dice que es : Torres
Su correo dice que es : davidtr_2007@hotmail.com
Su nick dice que es : davidtr
Su contraseña dice que es : clavepiloto
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
1
USAGE: create_ftp_user.sh [username] [password]
Creating user: davidtr
With password: clavepiloto
Apagando vsftpd: [ OK ]
Iniciando vsftpd para vsftpd: [ OK ]
Nueva conexión entrante para registro: Socket[addr=/192.168.1.113,port=53892,localport=1024]

```

1. Conexión Establecida

2. Datos para Acceso Web

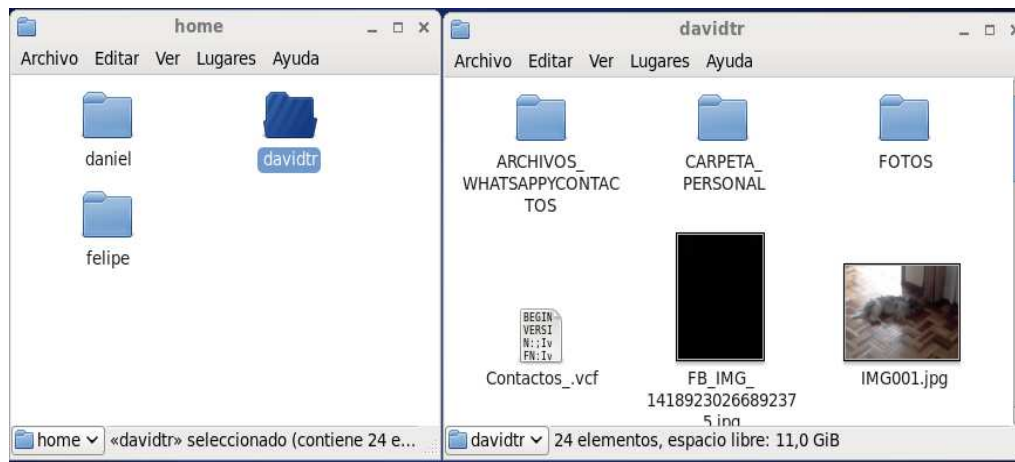
3. Creación de Usuario ftp

**Figura. 28. Prueba Servidor Socket Registro LAN**

Posteriormente del Registro se cerciora que los archivos de su celular sean subidos al servidor al instante y cada cierto periodo de tiempo ya estipulado, como se observa en la figura 29, también se añade a los anteriores archivos de la carpeta personal y fotos es la del archivo de “Vcard<sup>10</sup>” para el respaldo de contactos.

<sup>9</sup> Bash es un programa informático cuya función consiste en interpretar órdenes que está basado en Está basado en la Shell de Unix.

<sup>10</sup> vCard es un formato estándar para el intercambio de información personal, específicamente tarjetas personales electrónicas se puede obtener más información de <https://tools.ietf.org/html/rfc6350>



**Figura. 29. Carpeta generada para Usuario**

Paralelamente, el servicio de reporte GPS se realiza automáticamente cada periodo de tiempo establecido en el momento de registro y configuración, obteniendo así las coordenadas del Smartphone.

En la figura 30 se observa la identificación única de cada Smartphone así como la fecha y hora del reporte GPS.

```

run:
Iniciando servidor para Gps...
[OK]
Nueva conexión entrante para registro: Socket[addr=/192.168.1.109,port=44768,localport=1025]
El cliente con idSesion 0 saluda
Identificación : 3
Fecha : 2015-01-31 22:09:01
Latitud: -0.30858175
Longitud : -78.55375676
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... 0k
Nueva conexión entrante para registro: Socket[addr=/192.168.1.109,port=44769,localport=1025]
El cliente con idSesion 1 saluda
Identificación : 3
Fecha : 2015-01-31 22:16:43
Latitud: -0.30834595
Longitud : -78.5536489
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... 0k
Nueva conexión entrante para registro: Socket[addr=/192.168.1.109,port=44770,localport=1025]
El cliente con idSesion 2 saluda
Identificación : 3
Fecha : 2015-01-31 22:19:27
Latitud: -0.30895125
Longitud : -78.5535807
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... 0k

```

1. Reporte  
GPS

**Figura. 30. Prueba Servidor Socket GPS LAN**



Es importante revisar el correcto funcionamiento de la base de datos que es consultada muchas veces, es así como se muestra en la figura 31 la información que tiene la base de datos.

```
mysql> select * from acceso;
+-----+-----+-----+
| IdCliente | Usuario | Clave      |
+-----+-----+-----+
|          1 | davidtr | clavepiloto |
+-----+-----+-----+
1 row in set (0.00 sec)

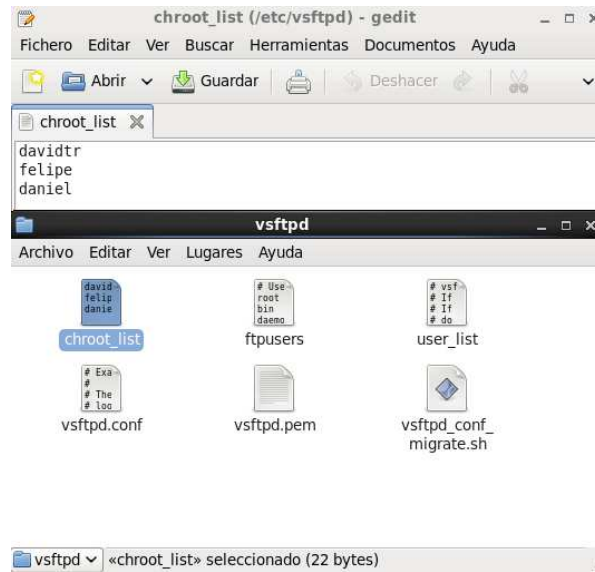
mysql> select * from celulares;
+-----+-----+
| IdCliente | numcel      |
+-----+-----+
|          1 | 0993577749 |
+-----+-----+
1 row in set (0.04 sec)

mysql> select * from cliente;
+-----+-----+-----+-----+
| IdCliente | Nombre | Apellido | Correo      |
+-----+-----+-----+-----+
|          1 | David  | Torres   | davidtr_2007@hotmail.com |
+-----+-----+-----+-----+
1 row in set (0.04 sec)

mysql> select * from ubicacion;
+-----+-----+-----+-----+
| IdCliente | fecha                | latitud    | longitud    |
+-----+-----+-----+-----+
|          1 | 2015-02-11 00:07:12 | -0.30871034 | -78.55334768 |
|          1 | 2015-02-11 00:07:39 | -0.30870456 | -78.55337290 |
|          1 | 2015-02-11 00:10:35 | -0.30884315 | -78.55354850 |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

**Figura. 31. Prueba Servidor MySQL**

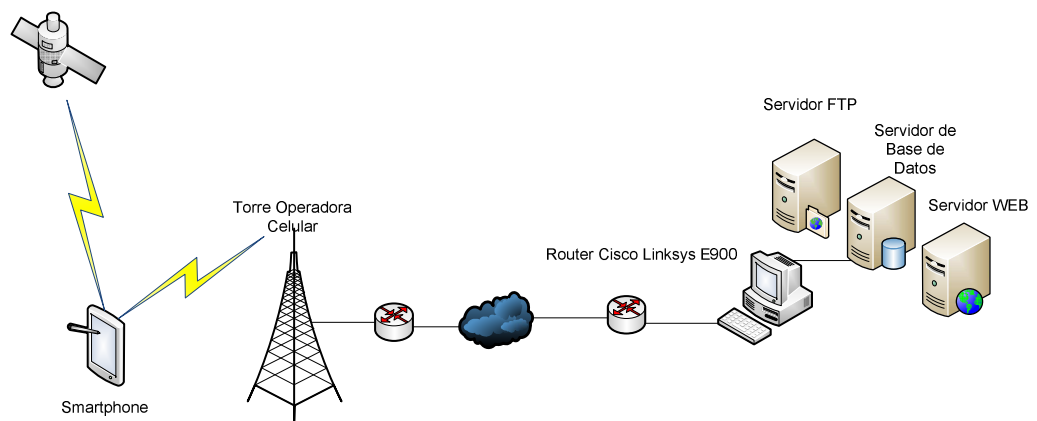
En la figura 32, se observa el correcto funcionamiento del servidor Vsftpd con sus usuarios generados en la lista de “Chroot\_list” el cual posee la lista de usuarios virtuales, estos usuarios pueden navegar por toda su carpeta personal.



**Figura. 32. Prueba Servidor Vsftpd**

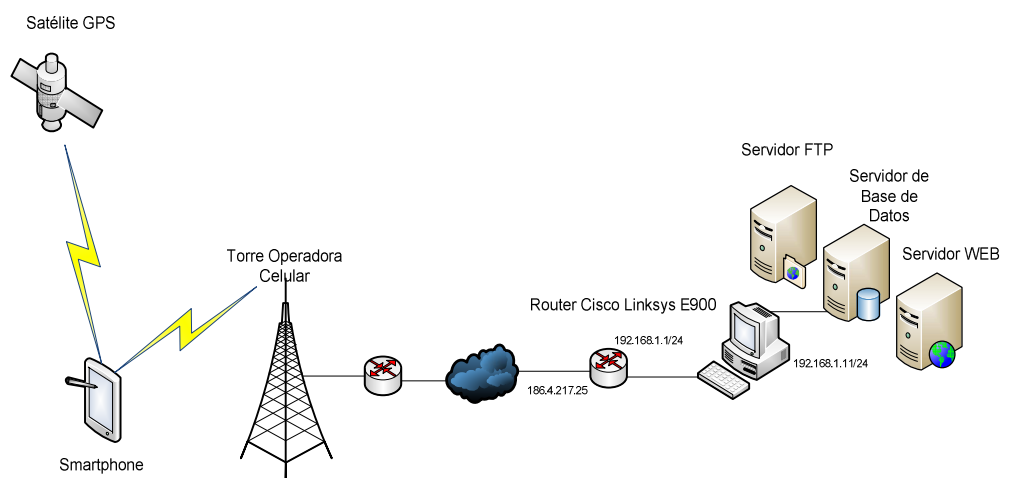
## 4.2 ESCENARIO RED DE AREA EXTENDIDA (WAN)

El escenario planteado en la figura 33 muestra una red WAN la cual cuenta en su dominio con tres servidores los cuales están bajo el sistema operativo de Centos 6.5 en la modalidad de máquina virtual, también el Smartphone está conectado a la red WAN mediante un plan de datos.



**Figura. 33. Topología de la red WAN**

A continuación se realizó la etapa de direccionamiento la cual se puede ver en la figura 34 y tabla 8.



**Figura. 34. Topología de red WAN con Direccionamiento**

Tabla 8

Tabla de direccionamiento WAN

Dispositivo	Interfaz	Dirección IP	Máscara	Tipo de Red
Router Cisco Linksys E900	Ethe1	186.4.217.25		WAN
	Ethe2	192.168.1.1	255.255.255.0	LAN
Smartphone Alcatel Touch 995	Ethe1	Dirección Ip obtenida por red celular		
HP14	Ethe1	192.168.1.11	255.255.255.0	LAN

Se realizó la implementación de la aplicación Android como en la red WAN escogiendo a Android Studio como entorno de desarrollo, también el Smartphone Alcatel OT995

#### 4.2.1. Pruebas de Software en el Servidor

```

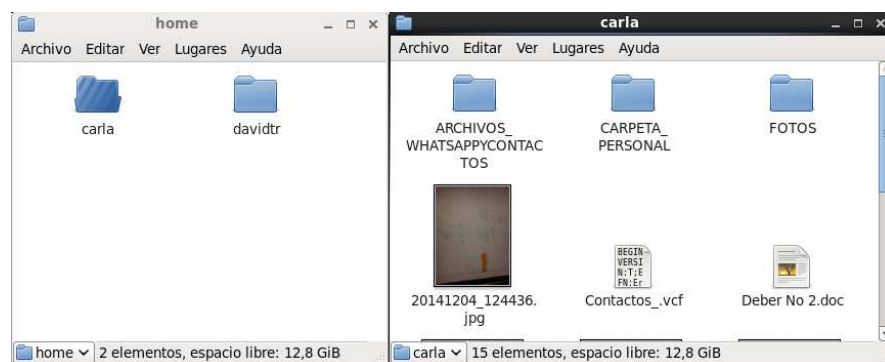
Iniciando servidor para Registro... [OK]
Nueva conexión entrante para registro: Socket[addr=/200.85.83.14,port=13747,localport=1024]
El cliente con idSesion 0 saluda
Su celular dice que es : 0992530641
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
No esta Registrado este numero de celular en nuestra base de datos
Su celular dice que es : 0992530641
Su nombre dice que es : Carla
Su apellido dice que es : Salgado
Su correo dice que es : carlasalgado12@hotmail.com
Su nick dice que es : carla
Su contraseña dice que es : clavepiloto2
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
2
USAGE: create_ftp_user.sh [username] [password]
Creating user: carla
With password: clavepiloto2
Apagando vsftpd: [ OK ]
Iniciando vsftpd para vsftpd: [ OK ]

```

Figura. 35. Prueba Servidor Socket Registro WAN

Como se puede observar en la figura 35, el proceso de registro de cada uno de los clientes llega al servidor recibiendo la solicitud de conexión hasta que se lo acepta posteriormente recibe un Nick y contraseña lo cual ayuda al proceso de Java de crear el usuario ftp llamando a un punto bash el cual generara dicho usuario.

Posteriormente del Registro se cerciora que los archivos del celular sean subidos al servidor al instante, como se observa en la figura 36.



**Figura. 36. Carpeta generada para Usuario**

Paralelamente como se muestra en la figura 37 el reporte GPS se realiza automáticamente cada un tiempo establecido en el momento de registro y configuración, obteniendo así las coordenadas del Smartphone.

```

Iniciando servidor para Gps...
[OK]
Nueva conexión entrante para registro: Socket[addr=/200.85.83.14,port=38654,localport=1025]
El cliente con idSesion 0 saluda
Identificación : 2
Fecha : 2015-02-24 21:56:30
Latitud: -0.3087629
Longitud : -78.5536803
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
Nueva conexión entrante para registro: Socket[addr=/200.85.83.14,port=42103,localport=1025]
El cliente con idSesion 1 saluda
Identificación : 2
Fecha : 2015-02-24 21:58:16
Latitud: -0.30877123
Longitud : -78.55363654
Conexión a base de datos jdbc:mysql://localhost/baseregistro ... Ok
Nueva conexión entrante para registro: Socket[addr=/200.85.83.14,port=40416,localport=1025]
El cliente con idSesion 2 saluda
Identificación : 2
Fecha : 2015-02-24 22:02:03
Latitud: -0.30888443
Longitud : -78.55363531

```

**Figura. 37. Prueba Servidor Socket GPS WAN**

Es importante revisar el correcto funcionamiento de la base de datos que será consultada muchas veces, es así como se muestra en la figura 38 la información que tiene la base de datos.

```
mysql> select * from acceso;
+-----+-----+-----+
| IdCliente | Usuario | Clave      |
+-----+-----+-----+
|          1 | davidtr | clavepiloto |
|          2 | carla   | clavepiloto2 |
+-----+-----+-----+
2 rows in set (0.00 sec)

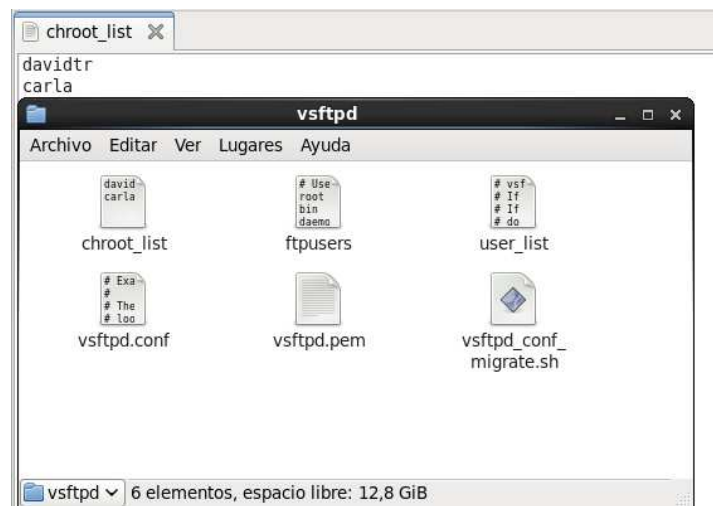
mysql> select * from celulares;
+-----+-----+
| IdCliente | numcel  |
+-----+-----+
|          1 | 0993577749 |
|          2 | 0992530641 |
+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from cliente;
+-----+-----+-----+-----+
| IdCliente | Nombre | Apellido | Correo      |
+-----+-----+-----+-----+
|          1 | David  | Torres   | davidtr_2007@hotmail.com |
|          2 | Carla  | Salgado  | carlasalgado12@hotmail.com |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from ubicacion;
+-----+-----+-----+-----+
| IdCliente | fecha                | latitud    | longitud    |
+-----+-----+-----+-----+
|          1 | 2015-02-11 00:07:12 | -0.30871034 | -78.55334768 |
|          1 | 2015-02-11 00:07:39 | -0.30870456 | -78.55337290 |
|          1 | 2015-02-11 00:10:35 | -0.30884315 | -78.55354850 |
|          2 | 2015-02-24 21:56:30 | -0.30876290 | -78.55368030 |
|          2 | 2015-02-24 21:58:16 | -0.30877123 | -78.55363654 |
+-----+-----+-----+-----+
```

Figura. 38. Prueba Servidor MySQL en red WAN

En la figura 39, se observa el correcto funcionamiento del servidor Vsftpd con sus usuarios generados en la lista de “Chroot\_list” el cual posee la lista de usuarios virtuales, estos usuarios pueden navegar por toda su carpeta personal.



**Figura. 39. Prueba Servidor Vsftpd**

Como se observa en la figura 40 se realiza el proceso de logueo con éxito en el bloque 1, después en el bloque 2 se realiza el proceso de cambio de raíz es decir el cliente se posiciona en la carpeta raíz que la ha hecho asignada por el servidor, como última etapa el bloque 3 se observa la recepción de los archivos hospedados en el Smartphone.

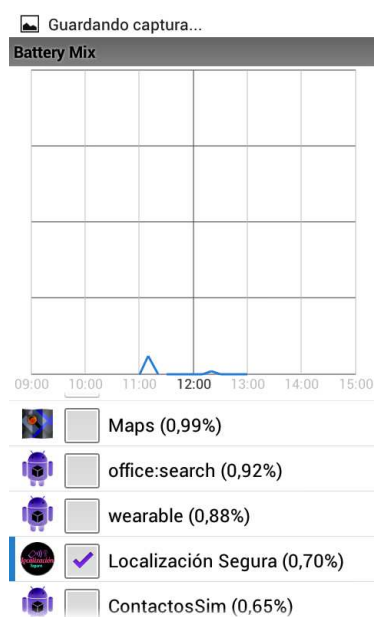
No.	Time	Source	Destination	Protocol	Info
1	161281	222.8839820(192.168.1.11)	192.168.1.1	FTP	Response: 220 Welcome to blah FTP service.
	161298	222.8911970(192.168.1.1)	192.168.1.1	FTP	Request: USER davidtr
	161303	222.8915920(192.168.1.11)	192.168.1.1	FTP	Response: 331 Please specify the password.
	161307	222.8944320(192.168.1.1)	192.168.1.1	FTP	Request: PASS clavepiloto
	161377	222.9564140(192.168.1.11)	192.168.1.1	FTP	Response: 230 Login successful.
161380	222.9730710(192.168.1.1)	192.168.1.1	FTP	Request: TYPE I	
161382	222.9734330(192.168.1.11)	192.168.1.1	FTP	Response: 200 Switching to Binary mode.	
2	161383	222.9772200(192.168.1.1)	192.168.1.1	FTP	Request: CWD /
	161384	222.9773630(192.168.1.11)	192.168.1.1	FTP	Response: 250 Directory successfully changed
	161394	222.9873800(192.168.1.1)	192.168.1.1	FTP	Request: SYST
	161397	222.9874870(192.168.1.11)	192.168.1.1	FTP	Response: 215 UNIX Type: L8
	161422	223.0349950(192.168.1.1)	192.168.1.1	FTP	Request: PASV
161423	223.0359700(192.168.1.11)	192.168.1.1	FTP	Response: 227 Entering Passive Mode (192,168,	
161437	223.0443440(192.168.1.1)	192.168.1.1	FTP	Request: LIST /	
161438	223.0445480(192.168.1.11)	192.168.1.1	FTP	Response: 150 Here comes the directory listing	
161485	223.0514090(192.168.1.11)	192.168.1.1	FTP	Response: 226 Directory send OK.	
161510	223.0650970(192.168.1.1)	192.168.1.1	FTP	Request: PASV	
3	161511	223.0653000(192.168.1.11)	192.168.1.1	FTP	Response: 227 Entering Passive Mode (192,168,
	161534	223.0740770(192.168.1.1)	192.168.1.1	FTP	Request: STOR IMG_20150205_172041.jpg
	161535	223.0747110(192.168.1.11)	192.168.1.1	FTP	Response: 150 Ok to send data.
	162718	223.5899230(192.168.1.11)	192.168.1.1	FTP	Response: 226 Transfer complete.
	162720	223.5949150(192.168.1.1)	192.168.1.1	FTP	Request: PASV
162721	223.5959500(192.168.1.11)	192.168.1.1	FTP	Response: 227 Entering Passive Mode (192,168,	
162732	223.6049450(192.168.1.1)	192.168.1.1	FTP	Request: STOR IMG_20150205_173039.jpg	
162738	223.6053380(192.168.1.11)	192.168.1.1	FTP	Response: 150 Ok to send data.	
163897	223.9969770(192.168.1.11)	192.168.1.1	FTP	Response: 226 Transfer complete.	

Figura. 40. Captura Wireshark Servicio FTP



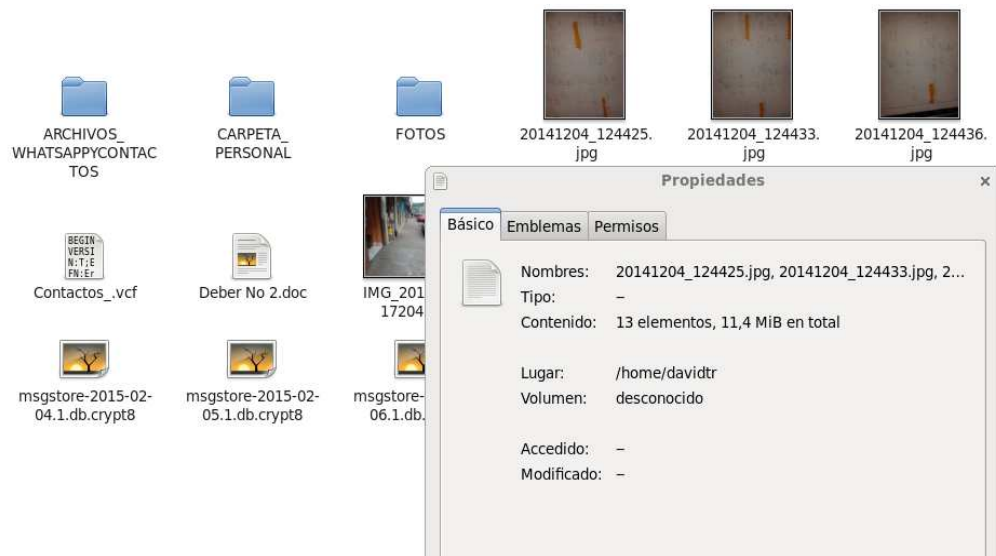
### 4.3 ANÁLISIS DE RESULTADOS

La aplicación móvil denominada Localización Segura pesa 251Kb lo cual significa que no representa una aplicación muy pesada además, se avaluó el consumo de batería utilizando una aplicación gratuita llamada “Battery mix” la cual proporciona información sobre el proceso en ejecución de la batería, cuando la aplicación móvil está al máximo de sus servicios es decir ejecutándose el sensor GPS y envío de datos ftp representa el 0,70 % como se observa en la figura 41.



**Figura. 41. Consumo de Batería de la aplicación.**

Como se puede ver en la figura 42 la cantidad de memoria subida del Smartphone al servidor es de 11,4 Megas, pero al revisar los datos utilizados en el Smartphone por parte de la aplicación en la figura 43 revela 14,45 Megas lo cual define que el registro consume 3,05 M.



**Figura. 42. Memoria utilizada por el Usuario**



**Figura. 43. Uso de datos Smartphone**

El análisis del consumo de datos de la aplicación se tomó un número determinado de reportes y se constató el consumo de datos proporcionados por el Smartphone varias veces.

Cálculos:

$$\text{Numero de reportes gps} = 25$$

$$\text{Tiempo de muestra gps} = 25 \text{ min}$$

$$\text{Datos consumidos} = 0,01\text{MB}$$

Entonces para definir el valor de cada reporte GPS:

$$\text{Valor reporte GPS} = \frac{\text{Datos Consumidos}}{\text{Numero de reportes}} \quad (1)$$

$$\text{Valor reporte GPS} = \frac{0,01\text{MB}}{25}$$

$$\text{Valor reporte GPS} = 0,4\text{KB}$$

Lo que quiere decir, que al mes si se cuenta con un plan de datos el consumo de envío de la posición tendrá el siguiente valor él envío.

En un día reportando cada 1 minuto se obtiene 1440 reportes y al mes se obtiene 44640 reportes al mes

Por lo tanto al mes los reportes GPS consumen:

$$44640 \times 0,4\text{KB} = 17,856 \text{ MB} \quad (2)$$

La Comparación del rendimiento de la Aplicación Móvil en tres Smartphones diferentes se observa en la figura 44, los datos fueron obtenidos de una aplicación gratuita llamada “Assistant for Android” la cual brinda monitoreo del CPU, RAM, ROM, Almacenamiento interno de cada aplicación o proceso en ejecución.

**Tabla 9**  
**Características de los Smartphones**

<b>Dispositivo</b>	<b>Duración de Batería</b>	<b>Memoria RAM</b>	<b>Tipo de Batería</b>
<b>LG-P500h</b>	7 horas 30 min	512M	Standard, Li-Ion 1500 mAh
<b>Alcatel OT995A</b>	7 horas	512M	Standard, Li-Ion 1500 mAh
<b>Samsung mini S4</b>	14 horas	1G	Standard, Li-Ion 1900 mAh

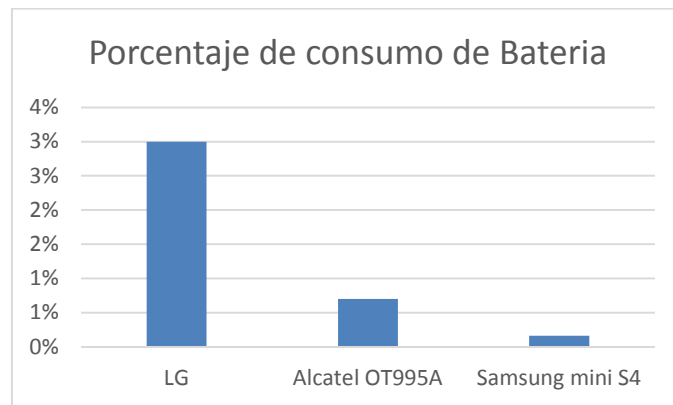
En la tabla 9 se observa las características de los Smartphones en duración de su batería y de su memoria RAM que serán comparados.

**Tabla 10**  
**Rendimiento de Aplicación**

<b>Dispositivo</b>	<b>% Batería</b>	<b>Memoria RAM</b>	<b>%RTCPU</b>	<b>Versión Android</b>
<b>LG-P500h</b>	3%	3,38M	0,97	2.2
<b>Alcatel OT995A</b>	0,70%	3,09M	0,11	4.0.4
<b>Samsung mini S4</b>	0,16%	5,91M	1,84	4.4.2

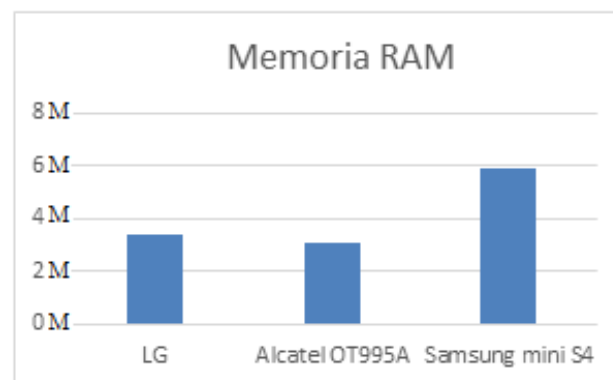
En la tabla 10 se el rendimiento de la aplicación móvil en cada uno de los tres Smartphones utilizados.

Como se observa en la figura 44, demuestra que el uso de la batería por parte de la aplicación móvil en el Samsung Galaxy Mini S4 consume menos batería, ya que cuenta con una duración de batería de 14 horas, tomando en cuenta en uso de llamadas es decir cuando el Smartphone está utilizando al máximo sus procesos, además las otros dos Smartphones tienen una duración de tan solo 7 horas aproximadamente.



**Figura. 44. Consumo de Bateria**

Como se observa en la figura 45, demuestra que el uso de la memoria RAM por parte de la aplicación móvil en el Samsung Galaxy Mini S4 la memoria consumida es mayor ya que cuenta con una mayor capacidad comparada con las demás Smartphones como se comprueba en la tabla 9.



**Figura. 45. Consumo de la memoria RAM**

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 5.1 CONCLUSIONES

- El consumo del plan de datos con la aplicación móvil confirma que al mes consumirá en reportes de ubicación aproximadamente 20MB, el cual se obtuvo con varias pruebas sobre un escenario WAN como se observa la sección de consumo de datos de la aplicación, de ahí se concluye que el consumo de datos al mes es de muy poco valor.
- La aplicación móvil en el consumo de batería mejora si el Smartphone en el cual se instala la aplicación cuenta con una batería Standard Li-Ion de 1900 mAh o superior, como lo corroboran las pruebas que se han realizado en diferentes tipos de Smartphones, que demuestran que mientras la capacidad de carga en mAh sea superior el uso de la batería por la aplicación será cada vez menor.
- La aplicación móvil en el consumo de memoria RAM mejora ostensiblemente si el Smartphone en el cual se instala la aplicación cuenta con una mayor capacidad de memoria, ya que ayuda a que los procesos de gestión al momento de enviar la posición o enviar archivos sea más rápida.
- El diseño y contenido de una aplicación Android tiene que cumplir con acuerdos de distribución de Google play así también sus políticas de

contenidos, en el caso que se quiera publicar dicha aplicación en Google Play Store.

- Con la creación e inicio de los servicios FTP y GPS en la aplicación móvil no garantiza que tales procesos se realicen de manera ininterrumpida ya que el sistema operativo puede detener los servicios en condiciones de memoria extremadamente bajas, cuando se cuenta con un Smartphone con capacidades de almacenamiento y de memoria RAM bajas.
- El proceso de la gestión de ubicación depende de los distintos sensores GPS que poseen cada tipo de Smartphones, por lo que el rendimiento varía de uno a otro y también influyen otros factores como la propia, la configuración y el software del sistema e incluso la ROM que se utiliza, todo puede ser motivo de variación de la calidad de la señal GPS.
- La implementación del servidor Vsftpd fue la mejor opción para generar usuarios ftp de manera automática, porque la aplicación móvil después de generar el registro, inicia el servicio FTP que sube los archivos del Smartphone de manera automática al servidor y para ese momento tiene que haber un usuario en el servidor con el mismo nombre y contraseña como se registró el usuario.

## 5.2 RECOMENDACIONES

- Es recomendable que la primera vez que se realiza la descarga y registro de usuario sea en un escenario de red LAN porque el consumo de datos después del registro inicia con el servicio ftp el cual envía los archivos del Smartphone al servidor y esto dependerá de cuánta información posee el Smartphone.
- Es recomendable para que esta aplicación funcione correctamente, el GPS tiene que estar activo, porque el servicio se crea e inicia por

primera y en ese instante adquiere datos como el tiempo en el cual reportara su posición.

- En trabajos futuros el uso de la seguridad SSL/TLS para la transmisión cifrada de información a través de un certificado es un método para mantener los datos de los usuarios para el envío de archivos sería primordial para la confidencialidad del usuario.
- En trabajos futuros se puede ampliar el sistema con un servidor GSM el cual confirme que el número ingresado para el registro sea realmente quien dice ser.



## REFERENCIA BIBLIOGRÁFICA

- Arguello Rivera, D. D. (Septiembre de 2012). *Desarrollo de una Aplicación que permita la captura, almacenamiento, reproducción, administración y envío de archivos de video y audio e imágenes utilizando tecnología bluetooth, para dispositivos móviles basados en la arquitectura del sistema operativo a.* Quito.
- Astudillo, J., & Delgado, E. (2012). *Sistema de Localizacion Monitoreo y Control Vehicular basado en los protocolos GPS/GSM/GPRS.* Cuenca.
- Báez, M. (15 de Junio de 2013). *tecnologiaUCM.* (V. López, Ed.) Recuperado el 12 de Agosto de 2014, de [www.tecnologiaUCM.es](http://www.tecnologiaUCM.es)
- Google. (31 de Enero de 2013). *Google Developers.* Recuperado el 2 de Noviembre de 2014, de [https://developers.google.com/maps/documentation/javascript/tutorial?hl=es#Loading\\_the\\_Maps\\_API](https://developers.google.com/maps/documentation/javascript/tutorial?hl=es#Loading_the_Maps_API)
- Google Play. (15 de Septiembre de 2014). *Google Play.* Recuperado el 10 de Enero de 2015, de [https://play.google.com/intl/ALL\\_mx/about/developer-distribution-agreement.html](https://play.google.com/intl/ALL_mx/about/developer-distribution-agreement.html)
- Pavón, N. (23 de Enero de 2013). *Ahu.* Recuperado el 12 de Septiembre de 2014, de <http://www.uhu.es/nieves.pavon/documentos/android/basicoandroid.pdf>
- Play, G. (25 de Septiembre de 2014). *Google Play.* Obtenido de [https://play.google.com/intl/ALL\\_mx/about/developer-distribution-agreement.html](https://play.google.com/intl/ALL_mx/about/developer-distribution-agreement.html)
- Tomás, J. (15 de Junio de 2014). *Android Curso.* Obtenido de <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1->

vision-general-y-entorno-de-desarrollo/146-las-versiones-de-android-y-niveles-de-api

Victor, H. (29 de Julio de 2013). *PhoneArena*. Obtenido de <http://www.phonearena.com>


VpsLink. (15 de 01 de 2015). *Vpslink*. Recuperado el 20 de 01 de 2015, de [http://wiki.vpslink.com/Configuring\\_vsftpd\\_for\\_secure\\_connections\\_%28TLS/SSL/SFTP%29](http://wiki.vpslink.com/Configuring_vsftpd_for_secure_connections_%28TLS/SSL/SFTP%29)

## ACTA DE ENTREGA

El proyecto fue entregado al Departamento de Eléctrica y Electrónica y reposa en la Universidad de las Fuerzas Armadas – ESPE desde:

Sangolquí, 23 de Marzo de 2015

ELABORADO POR:



Tito David Torres Racines

172175272-1

AUTORIDAD:



Ph.D. Ing. Nikolaj Sabinosa

DIRECTOR DE LA CARRERA DE ELÉCTRICA Y ELECTRÓNICA,  
REDES Y COMUNICACIÓN DE DATOS