

**UNIVERSIDAD TÉCNICA DE MACHALA**  
**UNIDAD ACADEMICA DE INGENIERÍA CIVIL**  
**CARRERA DE INGENIERÍA DE SISTEMAS**

**TRABAJO DE TITULACIÓN**

**PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA DE SISTEMAS**

**TEMA:**

**“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN  
MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE  
CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE  
CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL  
CIA.LTDA.)”**

**AUTORA:**

**LOAIZA GONZAGA ANDREA ANABELL**

**TUTOR:**

**ING. FAUSTO REDROVÁN CASTILLO, MG. SC.**

**CO-TUTOR**

**ING. JOFFRE CARTUCHE VARGAS**

**MACHALA - EL ORO - ECUADOR**

**2014**

## **CERTIFICACIÓN DEL TUTOR**

Ing. FAUSTO REDROVÁN CASTILLO, Mg. Sc, Profesor de la Unidad Académica de Ingeniería Civil de la Universidad Técnica de Machala, en calidad de Tutor de la Tesis de Grado titulada: “DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL CIA.LTDA.)”, elaborado por la estudiante ANDREA ANABELL LOAIZA GONZAGA, egresada de la carrera de Ingeniería de Sistemas de la Escuela de Informática, certifico que la mencionada Tesis estuvo bajo mi dirección y supervisión ajustándose a los procedimientos académicos y metodológicos establecidos por la Facultad, razón por la que autorizo su presentación para el trámite legal correspondiente.

---

Ing. Fausto Redrován Castillo, Mg. Sc.

## **CERTIFICACIÓN DEL CO-TUTOR**

Ing. JOFFRE CARTUCHE VARGAS, Profesor de la Unidad Académica de Ingeniería Civil de la Universidad Técnica de Machala, en calidad de Co-Tutor de la Tesis de Grado titulada: “DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL CIA.LTDA.)”, elaborado por la estudiante ANDREA ANABELL LOAIZA GONZAGA, egresada de la carrera de Ingeniería de Sistemas de la Escuela de Informática, certifico que la mencionada Tesis estuvo bajo mi dirección y supervisión ajustándose a los procedimientos académicos y metodológicos establecidos por la Facultad, razón por la que autorizo su presentación para el trámite legal correspondiente.

---

Ing. Joffre Cartuche Vargas

## **DECLARACIÓN DE AUTORÍA**

La información expuesta en este trabajo de investigación denominado: **“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL CIA.LTDA.)”**, es de autoría y responsabilidad de Andrea Anabell Loaiza Gonzaga con C.I.Nº 070569847-0.

---

Andrea Anabell Loaiza Gonzaga

## **AGRADECIMIENTO**

A Dios por bendecir mi camino, dame fuerzas para vencer los obstáculos y permitirme culminar mi carrera.

A mis padres dignos de ejemplo, trabajo y dedicación quienes me han brindado su amor incondicional, han estado en todos los momentos de mi vida apoyándome y alentándome a seguir adelante.

A mi tutor de tesis Ingeniero Fausto Redrován, quien durante todo este tiempo me colaboró en el desarrollo de este trabajo de titulación.

A mis profesores que han sido partícipes en mi formación académica dentro de esta prestigiosa Institución.

Agradecer a todas aquellas personas que en mayor o menor medida han ayudado a que este trabajo se desarrolle.

***Andrea Loaiza***

## **DEDICATORIA**

Quiero dedicar este trabajo en primer lugar a Dios, por ser quien guía mi vida y por darme la fortaleza para poder alcanzar esta meta.

A mis padres Daniel Loaiza y Herminia Gonzaga, quienes han sido el pilar fundamental de mi vida, por su apoyo, consejos, comprensión y por ayudarme con los recursos necesarios para estudiar.

A mis hermanos por estar siempre presentes, acompañándome para poderme realizar y por enseñarme a encarar ñas adversidades sin desfallecer en el intento.

***Andrea Loaiza***

# **“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL CIA.LTDA.)”**

Andrea Anabell Loaiza Gonzaga

## **Resumen**

La Compañía de construcciones e instalaciones eléctricas Coniel Cia.Ltda ha brindado un buen servicio a la comunidad en el área de ingeniería eléctrica, desempeñándose en estos últimos años bajo la adjudicación de contratos de control de pérdida de energía por la CNEL EP.

En la empresa el personal operativo ejecuta procesos manuales para el tratamiento y acceso a la información, por lo que se genera cierto retraso en las actividades que realizan diariamente.

Esto me ha motivado a poner en funcionamiento una propuesta innovadora desarrollada para la gestión de las actividades realizadas por el personal operativo con el fin de desarrollar e implementar una aplicación móvil que permita acceder a la información desde el sitio de trabajo y optimizar el tratamiento de los datos.

Esta aplicación interactiva, rápida y funcional ha sido desarrollada siguiendo los fundamentos de gestión de proyectos de la metodología ágil Mobile-D, junto con tecnologías como el Sistema Operativo Android, el lenguaje de programación Java y con acceso a un servidor con base de datos PostgreSQL.

Por lo que la aplicación se convirtió en gran apoyo para la compañía ya que aceleró la ejecución de sus procesos.

**Palabras Clave:** Aplicación móvil, acceso a la información, procesos manuales, retraso.

## INTRODUCCIÓN

La utilización inadecuada de las nuevas tecnologías de información, genera retraso en los procesos productivos dentro de las empresas u organizaciones, ya que el tratamiento de la información se la realiza mediante tareas manuales.

En la actualidad el avance tecnológico en el ámbito empresarial ha permitido que las organizaciones tengan una mayor rentabilidad, agilidad en la ejecución de sus procesos, rápido y fácil acceso a la información y la transparencia en los servicios o productos que ofertan.

La información es el activo más importante dentro de una organización por lo que debe ser confiable, integra y de fácil acceso.

Grandes empresas como Coca-Cola que cuentan con personal operativo refiriéndose a los trabajadores encargados de tomar pedidos desde los puntos de venta al cliente (tiendas, bares, etc.), cuentan con herramientas que facilitan la transferencia de información en tiempo real, ayudando de esta manera a agilizar los procesos y brindando fácil acceso a la información desde cualquier punto.

Las empresas que se dedican a brindar servicios específicamente de control de pérdida de energía a nivel nacional, no utilizan herramientas tecnológicas para realizar actividades operativas de campo y además el personal encargado no cuenta con la capacitación requerida para usar nuevas tecnologías, por lo que en la compañía de Construcciones e Instalaciones Eléctricas CONIEL CIA.LTDA aún utilizan métodos tradicionales como: el uso de formatos impresos para el control de las actividades operativas, la utilización de cámaras para la captura de fotografías concernientes al trabajo realizado, y la comunicación a través de radios o teléfonos celulares que en muchas veces no cuentan con la claridad necesaria para el intercambio de información eficiente.

Para mejorar el desarrollo de las actividades que realizan las compañías dedicadas a los trabajos de control de pérdida de energía, es necesario implementar nuevas técnicas o mecanismos para optimizar el uso de tecnologías como Aplicaciones móviles que permitan cubrir las falencias que se han venido dado en la ejecución de los trabajos.

Para el desarrollo e implementación de la aplicación móvil se utilizó la plataforma Android aplicando la metodología ágil Mobile-D y con una base de datos POSTGRESQL para el almacenamiento de la información.

## JUSTIFICACIÓN

Los procesos automatizados son factores de vital importancia en cualquier empresa. El desarrollo de las tecnologías de información ha permitido la evolución de los dispositivos móviles capaces de manejar sistemas tecnológicos en distintas áreas.

Las actividades diarias desarrolladas en el campo por el personal operativo de CONIEL CIA.LTDA, requieren que la información sea procesada y almacenada de manera eficiente, hecho que agilizará los procesos de otras actividades.

Con el desarrollo de una aplicación que se ejecute desde un dispositivo móvil se resolverán las necesidades y requerimientos de la empresa, además se logrará un control integral de las actividades y se optimizará el procesamiento de los datos.

La portabilidad de los dispositivos móviles facilitan su traslado al lugar del trabajo, además esto incrementará la confiabilidad de la información, mediante la disminución de pérdida o alteración de los datos por el ejecución manual de este proceso.

Considerando lo señalado se propone denominar este proyecto como **“DESARROLLO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA LA GESTIÓN Y SOPORTE DE LAS ACTIVIDADES DE CAMPO REALIZADAS POR EL PERSONAL OPERATIVO DE CONSTRUCCIONES E INSTALACIONES ELÉCTRICAS (CONIEL CIA.LTDA.)”**

Con esta aplicación se busca obtener una herramienta con opciones de tareas comunes realizadas por el personal operativo que necesitan soporte para un eficiente tratamiento de la información en lo posterior, haciendo uso de las nuevas tecnologías y del auge de las aplicaciones móviles para brindar un mejor desarrollo en las actividades de la compañía CONIEL CIA.LTDA.



# **CAPITULO I**

## **1. MARCO REFERENCIAL**

### **1.1 PLANTEAMIENTO DEL PROBLEMA**

La compañía de Construcciones e Instalaciones Eléctricas CONIEL CIA.LTDA es una empresa dedicada a trabajos de Ingeniería Eléctrica, siendo su punto fuerte los contratos de pérdidas y control de energía adjudicados a través del Sercop ([www.compraspublicas.gob.ec](http://www.compraspublicas.gob.ec)) por la CNEL EP Unidad de Negocios El Oro.

En CONIEL CIA.LTDA, las actividades operativas en el campo son realizadas por cuadrillas de trabajo conformadas por tres personas encargadas de trasladarse a lugares estratégicos o determinados anticipadamente por CONIEL para la realización de trabajos de electrificación correspondientes a control de pérdidas de energía y abastecimiento del servicio a zonas o abonados determinados, cada cuadrilla consta de un anotador, que es el encargado de capturar fotos a razón de justificar cada acción a tomar y llenar plantillas de datos (también llamadas fichas) con información relevante de las operaciones que se realicen, las mismas que pueden ser: servicios nuevos, cambios de medidor o mantenimientos del servicio dependiendo de las necesidades del cliente y lo notificado a realizar.

Dicha información al final de día es trasladada a la oficina central de la compañía para su posterior procesamiento, ingreso al Sistema SICO (Sistema Comercial de la CNEL EP) y tabulación.

#### **1.1.1 PROBLEMA CENTRAL**

La gestión de información actual desde el sitio de trabajo, genera retraso en los procesos productivos y disminuye el porcentaje de confiabilidad e integridad de la información registrada e ingresada en la empresa.

#### **1.1.2 PROBLEMAS PARTICULARES**

- En la empresa CONIEL CIA.LTDA las TIC's son desconocidas y no utilizadas por el personal.
- En ocasiones el personal operativo no cuenta con información necesaria para el registro de la actividad en la ficha (formato para el registro de la actividad realizada) desde el lugar de trabajo y requiere de ayuda del personal administrativo.
- La toma de Información en el sitio muchas veces es inconsistente e incompleta.
- Los procesos realizados por el personal operativo son netamente manuales.
- La empresa posee computadoras potentes que pueden ser utilizadas para la implementación de un servidor, pero únicamente son usadas para la tabulación, búsqueda y registro de la información.
- La empresa no cuenta con la información de ubicación de cada cuadrilla a lo largo del día.

## **1.2 PREGUNTAS CIENTIFICAS**

### **1.2.1 PREGUNTA CENTRAL**

¿La implementación de la aplicación móvil para la optimización de las actividades realizadas por el personal operativo permitirá gestionar la información desde el sitio de trabajo y brindará mejoras en los procesos productivos de la compañía CONIEL CIA.LTDA?

### **1.2.2 PREGUNTAS COMPLEMENTARIAS**

- ¿Con la capacitación al personal de la empresa Coniel Cia. Ltda. se logrará un mejor uso de las TIC's?
- ¿Se logrará mejorar el acceso a la información desde el sitio de trabajo?
- ¿Se mejorará la integridad y confiabilidad de la información?
- ¿La aplicación móvil permitirá llevar una gestión automatizada de los procesos que realiza el personal operativo?
- ¿Se implementará un servidor haciendo uso de la potencia del hardware que posee la empresa?
- ¿A lo largo del día se contará con la información de ubicación donde se encuentra cada cuadrilla?

## **1.3 OBJETIVOS**

### **1.3.1 OBJETIVO PRINCIPAL**

- Implementar una aplicación móvil para la gestión y soporte de las actividades de campo realizadas por el personal operativo de CONIEL Cia.Ltda utilizando la metodología Mobile-D”.

### **1.3.2 OBJETIVOS ESPECIFICOS**

- Capacitar al personal con el fin de lograr la actualización de conocimientos en el campo de nuevas tecnologías mediante cursos impartidos por profesionales en el tema
- Dar soporte necesario de información al personal operativo en los distintos puntos de trabajo mediante un módulo de búsqueda de datos para de esta manera abolir el sistema actual basado en fichas y obtener una mejor confiabilidad de la información registrada.
- Diseñar una aplicación móvil para automatizar los procesos manuales realizados por el personal operativo mediante la herramienta de desarrollo Android Studio.
- Implementar un servidor interno de la empresa con el fin de centralizar la información mediante el uso objetivo de nuevas tecnologías.
- Brindar geolocalización de los grupos de trabajo para llevar un mejor control de los mismos mediante la implementación de un módulo de monitorización que se añadirá al servidor web interno.

## 1.4 ALCANCE

Previo al desarrollo del proyecto de tesis se realizará una investigación en la compañía de Construcciones e Instalaciones Eléctricas Coniel Cia.Ltda., poniendo énfasis a los procesos que realiza el personal operativo durante sus actividades diarias e identificar y analizar los problemas que tienen con el manejo de la información, para de esta manera integrarlos en una aplicación que permitirá un eficiente tratamiento de los datos desde el sitio de trabajo y mejorar la confiabilidad e integridad de la información.

La aplicación será diseñada de tal manera que sea intuitiva y de fácil manejo, para que el personal operativo pueda acceder a las opciones de forma sencilla, pudiendo así obtener mejores resultados en la ejecución de las actividades en tiempo real.

Para la realización del proyecto de titulación se utilizará un lenguaje de programación Java, el IDE Android Studio, base de datos PostgreSQL y dispositivos móviles con Sistema Operativo Android, además para un mejor control y gestión del proyecto se usará la metodología ágil Mobile-D que contemplará las siguientes acciones:

MÓDULO	FUNCIÓN	SUBFUNCIÓN	DESCRIPCIÓN
<b>Login de la Aplicación</b>	Ingreso de usuario y contraseña	Validación y Verificación de los datos	Valida y verifica los datos ingresados, la disponibilidad de la sesión, las sesiones activas en el servidor y finalmente da apertura a una sesión.  También comprueba si el usuario que se encuentra realizando la petición de ingreso a la aplicación tiene acceso al contrato que ha seleccionado.  La persona encargada de recolectar la información de la actividad realizada ingresa a la aplicación mediante un usuario y una contraseña y accede al menú principal.
		Comprobación de la identidad del usuario ingresado	
		Ingreso a la aplicación	
		Comprobación de Sesiones del servidor	
		Comprobación de sesiones disponibles	
		Apertura de Sesión	
		Comprobación de acceso a contrato	

<b>Ingreso de Actividades</b>	Ingreso de información de las actividades realizadas	Búsqueda, Validación y Verificación de datos de abonado y de información de medidores	El personal operativo puede ingresar información necesaria de la actividad realizada, así mismo puede buscar datos del abonado y de los medidores asignados a él en el sistema comercial SICO, además puede capturar las imágenes para constatar la ejecución de su trabajo. Puede buscar las actividades realizadas para verificar si ya han sido ingresadas.
		Búsqueda de actividades realizadas	
		Ingreso de detalle de instalación del servicio	
		Captura de fotografías de la actividad realizada	
<b>Búsqueda</b>	Búsqueda de información de abonados por criterio en el sistema comercial SICO.	Búsqueda de datos por cuenta (código único de abonado)	Este módulo realiza búsquedas por criterios al sistema comercial SICO, y provee información necesaria para el personal operativo.
		Búsqueda de datos por medidor	
		Búsqueda de datos por nombre de abonado	
		Búsqueda de datos por geocódigo	
<b>Fotos</b>	Capturar fotografías de las actividades realizadas	Ingreso de cuenta para almacenar las fotografías	Este módulo permite capturar las fotografías de la actividad realizada, así mismo seleccionar la fecha para crear una nueva cuenta en la que almacenará las imágenes o seleccionar una cuenta de la lista a la que desee añadir más fotos o eliminar alguna(s) de la galería.  También puede seleccionar una imagen de la galería de la cuenta seleccionada para visualizar en tamaño real.
		Capturar imágenes	
		Selección de fecha y de cuenta para captura de imágenes	
		Añadir más imágenes a una cuenta seleccionada	
		Eliminar imágenes seleccionadas	
		Visualización de imágenes	

<b>Geolocalización</b>	Detectar ubicación del sitio en el que se encuentran las cuadrillas	Detectar ubicación actual de las cuadrillas	Esta sección de la aplicación permite tener un mejor control de la ubicación actual de las cuadrillas, además de conocer su propia ubicación y obtener la ruta más corta hacia una cuadrilla específica.
		Detectar mi ubicación	
		Ruta más cercana a una cuadrilla específica	

**Tabla 1.** Detalle de alcance del proyecto

## CÁPITULO II

### 2. MARCO TEÓRICO

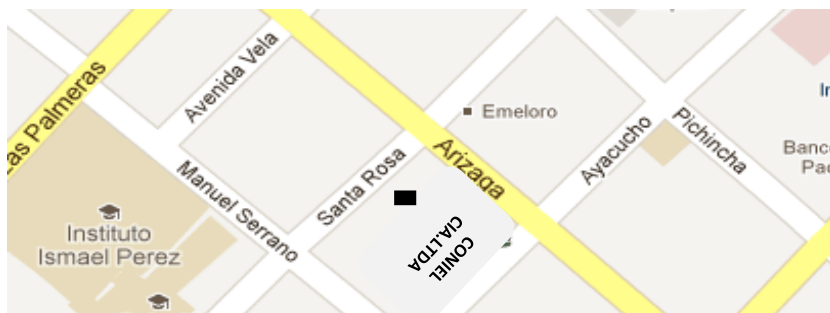
#### 2.1 ANTECEDENTES HISTÓRICOS DE LA EMPRESA

##### 2.1.1 RESEÑA HISTÓRICA DE CONIEL CIA.LTDA

La compañía de Construcciones e Instalaciones Eléctricas CONIEL CIA.LTDA, tiene sus inicios debido a la gran demanda de servicios eléctricos en nuestra Provincia. Fue constituida el 02 de julio del 2002, resultado de la asociación entre el Tnlg. Julio Loaiza y el Sr. Luis Pérez, para de esta manera dar paso al crecimiento de la empresa Privada cuyo único objetivo era ganar experiencia en el ámbito profesional y brindar servicios de calidad a la ciudadanía. Con el pasar de los años, la compañía es fuertemente constituida y cuenta con un amplio número de contratos firmados con la CNEL EP, los mismos que conforman su experiencia profesional. La empresa ha cumplido a cabalidad con las actividades de la prestación de los servicios adjudicado con la CNEL EP. La Compañía ha ido creciendo y ganando terreno en la Provincia y haciéndose acreedora a muy buenas referencias de trabajo por parte de la CNEL EP.

Actualmente es una compañía fuertemente establecida que brinda servicios a la comunidad trabajando en conjunto con la CNEL EP. (Coniel Cia.Ltda, 2013)

##### 2.1.2 UBICACIÓN

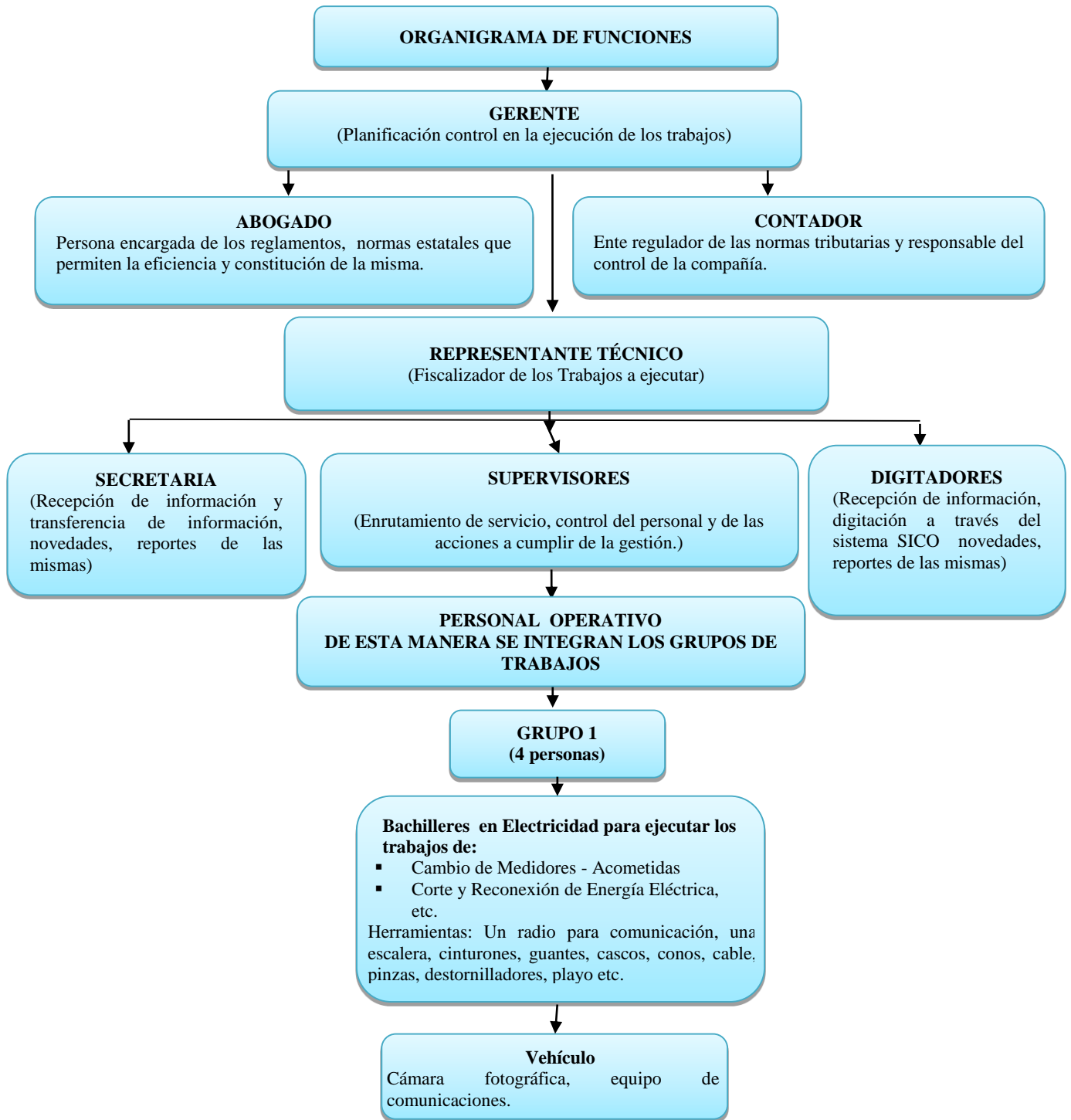


*Figura 1.* Croquis de ubicación de la Empresa Coniel Cía. Ltda.

*Fuente:* Coniel Cia.Ltda

**Dirección:** Machala, Santa Rosa e/ Arízaga y Gral. Manuel Serrano.  
**Teléfono:** 2938-581

### 2.1.3 ORGANIGRAMA ESTRUCTURAL



*Figura 2.* Estructura Organización de CONIEL Cia.Ltda.

*Fuente:* Coniel Cia.Ltda

## **2.2 ANTECEDENTES CONCEPTUALES**

### **2.2.1 TELEFONÍA MOVIL**

#### **2.2.1.1 Concepto**

“La telefonía fija o móvil, es aquella que hace referencia a las líneas y equipos que se encargan de la comunicación entre terminales telefónicos, generalmente enlazados entre ellos con la central por medio de conductores metálicos.” (Procoop, 2010)

“La telefonía celular es un sistema de comunicación telefónica totalmente inalámbrica, en este caso los sonidos se convierten en señales electromagnéticas, que viajan a través del aire, siendo recibidas y transformadas nuevamente en mensaje a través de antenas repetidoras o vía satélite.” (Botero, 2012)

La telefonía móvil es un sistema que permite el acceso a un nuevo medio de comunicación con el objeto de proveer el contacto entre usuarios.

#### **2.2.1.2 Historia**

Martínez (2001) afirma que en los Estados Unidos al introducir el primer radioteléfono en 1973 se lo consideró a Martin Cooper quien trabajaba en Motorola como el “padre de la telefonía celular” ya que fue el pionero en esta tecnología, pero fue en Tokio Japón en 1979 donde aparece el primer sistema comercial lanzado por la compañía NTT (Nippon Telegraph & Telephone Corp). En el año de 1983 la entidad reguladora de Estados Unidos adopta reglas para la creación del servicio de telefonía y pone en marcha el primer sistema comercial de telefonía celular en la ciudad de Chicago. Desde entonces la tecnología inalámbrica tuvo gran aceptación en varios países ya que la vieron como una alternativa a la telefonía convencional alámbrica, por lo que a los pocos años de utilizarse se empezó a saturar el servicio, y esto llevo a la necesidad de transformar los sistemas analógicos a digital con el fin de dar acceso a más usuarios, además de implementar técnicas que permitan el acceso múltiple al canal. (pág. 2)

#### **2.2.1.3 Dispositivos Móviles**

“Los dispositivos móviles son aparatos de pequeño tamaño, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada, diseñados específicamente para una función, pero que pueden llevar a cabo otras funciones más generales.” (Álvarez J. , 2008)

“Son aquellos suficientemente pequeños para ser transportados y empleados durante su transporte. Normalmente se sincronizan con un sistema de sobremesa para actualizar aplicaciones y datos, es aquel que es capaz de comunicarse o acceder a una red sin cables.” (Fernández, 2006)

Son dispositivos que permiten acceder a los servicios que brinda la telefonía móvil o celular a través de conexiones inalámbricas, son de fácil de transportar y permiten el manejo de información desde cualquier sitio.



#### **a) Características**

Para Guevara (2010), los dispositivos móviles se caracterizan por ser aparatos pequeños de memoria limitada, fáciles de movilizar, con diseños específicos destinados a una función principal, capacidades especiales para el procesamiento, conexión permanente a una red, y además de versatilidad para el desarrollo de otras funciones.

#### **b) Ventajas**

Las ventajas de un dispositivo móvil son la variedad de funcionalidades que posee, la fácil portabilidad, y la posibilidad de mantener una conexión a una red permanente. (Rabajoli, 2007)

#### **2.2.1.4 Aplicaciones móviles**

Según Quintanilla (2013), las aplicaciones móviles son programas que brindan una variedad de servicios como de entretenimiento, de información, que además son pensadas para facilitar la consecución de una tarea determinada o asistir en operaciones y gestiones del día a día.

De acuerdo con Reynaga (2013), una aplicación móvil es un software diseñado para ser ejecutada en dispositivos móviles inteligentes, y que además para el desarrollo de las mismas se deben tomar en cuenta una gran variedad de características como el tamaño de la pantalla, datos específicos de software y configuraciones.

### **2.2.2 SISTEMAS OPERATIVOS PARA DISPOSITIVOS MOVILES**

Rivera (2012) menciona que el sistema operativo móvil se diferencia con el de una pc tradicional en que no se pueden ejecutar aplicaciones robustas sobre él, pero en general se desempeña con rapidez. Es una plataforma que interpreta lo que el usuario desea que la terminal realice y lo ejecuta con inteligencia, esto permite la interacción real con lo que se puede hacer a través de las capacidades del hardware de un equipo.

Además el sistema operativo permite la gestión de los recursos entre las aplicaciones que los conforman, son más simples, orientados hacia la conectividad inalámbrica y a necesidades específicas.

#### **2.2.2.1 Sistemas Operativos para móvil**

Montoya (2012) explica los siguientes sistemas operativos para móvil:

<b>Sistema Operativo</b>	<b>Descripción</b>
<b>Android</b>	Desarrollado por Android Inc, firma que en el 2005 fue comprada por Google. Es un sistema operativo para teléfonos inteligentes.
<b>Symbian</b>	Este sistema operativo para terminales móviles se creó con el objetivo de competir con otros sistemas operativos del mismo propósito como el de Palm o el Windows Phone de Microsoft y ahora Android de Google Inc. Y iOS de Apple Inc.

<b>Ios</b>	IOS es un sistema operativo móvil basado en la manipulación directa, es decir incluye gestos como deslices, toques, pellizcos, lo que hace posible la interacción a través del contexto de la interfaz.
<b>BlackBerry OS</b>	Da soporte para varios métodos de entrada como touchpad y pantallas táctiles, además permite la ejecución de varias tareas a la vez.
<b>Windows Phone</b>	Fue desarrollado por Microsoft, es un sistema operativo móvil compacto con interfaz de usuario natural destinado para +ser utilizado en teléfonos inteligentes.

**Tabla 2.** Sistemas Operativos para móvil

*Fuente:* Montoya (2012)

## **2.2.3 SISTEMA OPERATIVO ANDROID**

### **2.2.3.1 Conceptos**

Ribas (2013) afirma que:

Android es una plataforma de desarrollo libre, y de código abierto: El núcleo del sistema está basado en un Linux (versión 2.6 para versiones 3.0 del kernel para posteriores) al que se le han hecho ciertas modificaciones para que pueda ejecutarse en teléfonos y terminales móviles. Android es el nombre esencial para un sistema operativo enfocado al uso del mismo en dispositivos móviles, tomando en cuenta que al inicio se lo creó para ser usado solamente en teléfonos celulares. En la actualidad se puede encontrar a Android en todo tipo de dispositivos como tablets, Smartphone, netbooks, entre otros. (pág. 18)

Paredes Velasco, Santacruz Valencia y Domínguez Mateos (2012) consideran que:

Android es una plataforma formada por un conjunto de software en estructura de pila (software stack) que incluye un sistema operativo, software para conectar aplicaciones (middleware) y aplicaciones base. El SDK (Software Development Kit, Kit de Desarrollo de Software) de Android proporciona varias herramientas y API (Applications Programming Interface, Interfaz de Programación de Aplicaciones) que son necesarias para desarrollar aplicaciones Android. (pág. 15)

Android es un sistema operativo para teléfonos móvil basado en el Kernel de Linux, de código libre, posee una amplia gama de aplicaciones disponibles. Permite el acceso a aplicaciones propias del sistema operativo como a (GPS, Cámara, Llamadas, Mensajes, etc), además de que da acceso a escritura y lectura a memoria tanto interna como externa.

### 2.2.3.2 Historia

Según Lara Cancela y Sara Ostos (2012), en el 2005 Google compra la firma Android Inc., y en ese entonces promocio a fabricantes de dispositivos y operadoras una plataforma para dispositivos móviles basada en el kernel de Linux que provee un sistema flexible y actualizable. La Open Handset Alliance, es una alianza dedicada al desarrollo de estándares abiertos para dispositivos móviles, que está conformada por un sin número de empresas entre fabricantes, desarrolladores y operadores de servicios, quienes adoptaron como principal producto el buscador y las aplicaciones para el uso en móviles de google.

De acuerdo con Castillo (2012), la compañía Android Inc. quien estaba desarrollando el sistema operativo para móvil Android, fue adquirida en el 2005 por Google. Por lo que en el 2008, el dispositivo T-Mobile fabricado por HTC y desarrollado conjuntamente con Google, fue el primero en poseer el sistema operativo en su versión 1.0 de Android.

### 2.2.3.3 Características

Para Ribas (2013), android posee grandes características como el soporte para audio, video y formato de imágenes, servicio de telefonía, permite la reutilización y la sustitución de los componentes, usa la máquina virtual Dalvik optimizada para dispositivos móviles, tiene un navegador integrado, permite el almacenamiento de datos estructurados a través de SQLite, y provee de servicios como Bluetooth, Wi-fi, Cámara, GPS. (pág. 19)

### 2.2.3.4 Arquitectura de Android

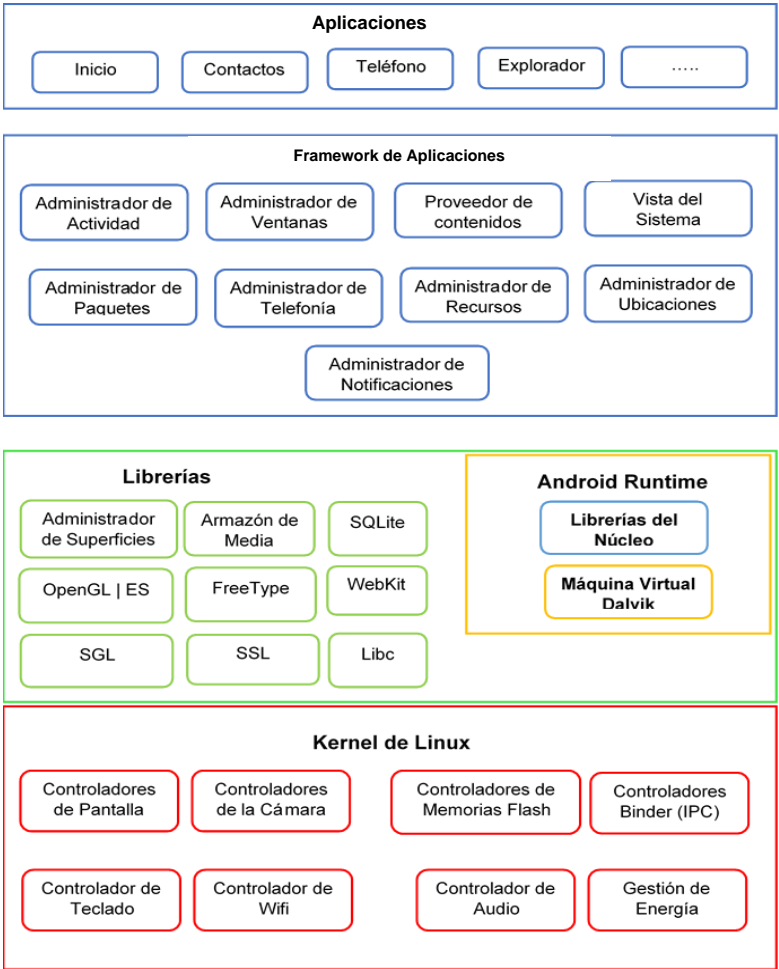
El sistema operativo Android según Ribas (2013, pág. 42) posee una arquitectura conformada por capas de software, en la que en la parte inferior hay un conjunto de drivers que se basan en Linux, y también un conjunto de librerías que pueden ser accedidas a través del nivel superior denominada Framework de aplicaciones. Además posee una pila de software en la que se encuentran el sistema operativo, middleware y aplicaciones básicas para el usuario.

Capa	Descripción
<b>Aplicaciones</b>	En esta capa se incluyen las aplicaciones básicas del dispositivo
<b>Framework de Aplicaciones</b>	Esta capa permite a las librerías javas el acceso a los recursos de niveles anteriores a través de la máquina virtual de Dalvik. Además está formada por clases y servicios que usan las aplicaciones para la ejecución de un trabajo.
<b>Las librerías nativas</b>	Están escritas en C o C++, se compilaron para la arquitectura de hardware usadas por el dispositivo, además fueron instaladas en el por el proveedor del mismo. Son contenidas en la capa inmediata superior al kernel de Linux.
<b>Runtime</b>	En un dispositivo móvil se pueden correr múltiples máquinas virtuales de manera eficiente, ya que cada proceso de una aplicación obtiene una instancia a la máquina virtual.

<b>Kernel Linux</b>	El manejo de la memoria, el control de los procesos u otros servicios relativos al sistema operativo están basados en Linux, ya que provee una capa de abstracción de hardware para Android.
---------------------	--

**Tabla 3.** Descripción de las capas de la Arquitectura de Android

**Fuente:** Ribas (2013), págs. 42-43



**Figura 3.** Arquitectura de Android


**Fuente:** Adaptado de Ribas (2013, pág. 44)

### 2.2.3.5 Ventajas y Desventajas

Según Santa María (2014), Android es un sistema operativo de código libre, que cuenta con un sin número de aplicaciones disponibles para descargar, además de que es un sistema compacto que permite la ejecución de varias tareas a la vez, aunque cabe destacar que así como posee grandes cualidades, una de sus desventajas es la poca duración de la batería.

### 2.2.3.6 Versiones

Martínez (2013) detalla las versiones del sistema operativo Android:

Nombre	Versión	Fecha de Lanzamiento	Descripción	Logo
<b>Apple Pie</b>	1.0	23 Septiembre del 2008	Fue lanzado el 23 de septiembre de 2008, el primer smartphone en el que fue instalado es el HTC Dream. Tenía soporte para aplicaciones básicas como SMS y cámara.	
<b>Banana Bread</b>	1.1	9 Febrero del 2009	En esta versión se dio soporte para diseños de sistemas básicos, además de la posibilidad de adjuntar archivos en mensajes.	
<b>Cupcake</b>	1.5	30 Abril del 2009	Contaba con un rediseño completo en su interfaz, además de transiciones animadas, mejoras en la velocidad de la cámara, teclado en la pantalla y soporte de bluetooth stereo.	
<b>Donut</b>	1.6	15 Septiembre de 2009	Fue una pequeña actualización, pero vino empaquetada con un cuadro de búsqueda mejorado, cámara y aplicación de galería, y una renovada Android Market.	
<b>Eclair</b>	2.0–2.1	25 Octubre de 2009	Se rediseño la interfaz del navegador, se da soporte nativo de flash para la cámara, zoom digital en las fotos.	
<b>Froyo</b>	2.2	Mayo de 2010	Incorpora el motor de Java V8 y ofrece a los usuarios un aumento de velocidad gracias al compilador JIT que permite iniciar las solicitudes más rápido y además de	

			mejorar el rendimiento general del sistema.	
<b>Gingerbread</b>	2.3	6 Diciembre de 2010	Se vio un considerable incremento en la velocidad de ejecución de tareas, además se renovó el diseño de la interfaz.	
<b>HoneyComb</b>	3.0–3.1	22 Febrero de 2011	Nueva interfaz de usuario con una barra de sistema en la parte inferior de la pantalla que permitía el acceso rápido a notificaciones, estados y botones de navegación suavizados y el Action Bar.	
<b>Icecream Sandwich</b>	4.0	19 Octubre 2011	Se implementó el diseño Holo en las interfaces de usuario e íntegro el sistema operativo en sus versiones para Tablets y Smartphones.	
<b>Jelly Bean</b>	4.1	Junio 2012	Mejoro la fluidez de las interfaces de usuario, brindando estabilidad y mejor rendimiento de la misma.	
<b>KitKat</b>	4.4	Diciembre 2013	Es la versión vigente del sistema operativo para móviles, con una interfaz impecable.	
<b>Lollipop</b>	5.0	Noviembre 2014	Este posee soporte para 64 bits y funciona en teléfonos, tablets y tvs.	

**Tabla 4.** Versiones del Sistema Operativo Android

*Fuente:* Adaptado de Martínez (2013)

### 2.2.3.7 Componentes de una Aplicación Android

Ribas Lequerica (2013) considera que:

Para diseñar una aplicación en Android, es necesario tener claros los elementos que la componen y la funcionalidad de cada uno de ellos. Android trabaja en Linux, y cada

aplicación utiliza un proceso propio. Se distinguen por el ID, un identificador para que solo ella tenga acceso a sus archivos. Los componentes son los elementos básicos con los que se construyen el proyecto. Habrá tantas actividades como ventanas distintas tenga la aplicación. (pág. 41)



**Figura 4.** Componentes de una Aplicación Android

**Fuente:** Ribas (2013), pág. 41

Componente	Descripción
<b>Activity</b>	Una Activity es una clase que permite representar la parte gráfica de la aplicación es decir es donde se mostraran las vistas (Views).
<b>Broadcast Intent Receivers</b>	Es un receptor de mensajes emitidos por el sistema.
<b>Service</b>	Es un componente que se ejecuta en background, y que periódicamente se conecta a un servidor para buscar cambios en la información.
<b>Content providers</b>	Provee una capa de abstracción para acceder a los datos almacenados de tal manera que sean accesibles entre aplicaciones.
<b>Fragment</b>	Su cometido principal es la reutilización tanto de código de lógica de trabajo como de las interfaces de esos códigos.
<b>Intents</b>	Los intentos son objetos de la clase que contiene datos del mensaje a transmitir.

**Tabla 5.** Descripción de los componentes de una Aplicación Android

**Fuente:** Adaptado de (Ribas Lequerica, 2013, pág. 49)

### 2.2.3.8 Estado de los procesos

De acuerdo con Ribas (2013), los procesos son situados en orden jerárquico basándose en sus estados. Cada aplicación de Android corre su propio proceso, el cual se crea cuando se ejecuta la aplicación y permanece ahí hasta que deja de trabajar o hasta cuando el sistema requiere de memoria para ejecutar otras aplicaciones, la jerarquía en la que se basa Android es la siguiente:

- ✓ **Actividad de primer plano (Foreground Activity):** Es el Activity con la que el usuario interactúa.
- ✓ **Actividad Visible:** Es la Activity que se ejecuta algo así como un cuadro de dialogo, que aunque no se puede interactuar con ella si es visible al usuario.
- ✓ **Actividad de Fondo (Background Activity):** Son las actividades que se están ejecutando en segundo plano es decir que el usuario no ve y que han sido pausadas.
- ✓ **Proceso vacío:** Son procesos que por lo general no depende de la ejecución de ningún otro componente, es decir no tienen asociada ninguna actividad, por lo que son los primeros que pueden ser descartados si es que el sistema necesita recursos. (págs. 54 - 55)

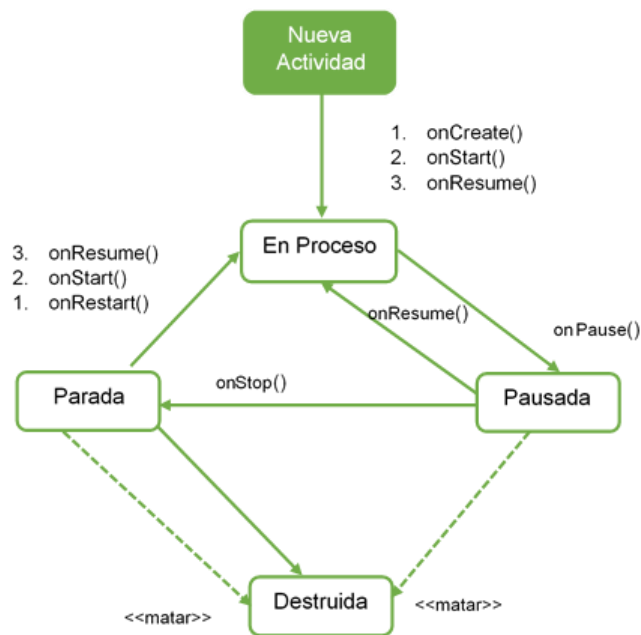
### 2.2.3.9 Ciclo de vida de una actividad

Una aplicación Android durante su ciclo de vida adopta varios estados que dependerán de la situación en la que se encuentre. (Ribas Lequerica, 2013, pág. 50)

De acuerdo con Ribas (2013), durante la ejecución normal de una aplicación, sus Activity pueden cambiar a alguno de los siguientes estados:

- ✓ **Activada:** Sucede cuando se encuentra la primera pila de ejecución, el usuario puede interactuar con ella, es decir esta visible para él.
- ✓ **Pausada:** Una actividad pasa a este estado cuando ha sido pausada, es decir ha perdido el foco y pasa a segundo plano.
- ✓ **Parada: Adopta** este estado cuando una actividad es cubierta en su totalidad por otra, además pasa a segundo plano.
- ✓ **Destruída:** Una actividad es destruida cuando el sistema ha liberado todos sus recursos, además si desea ejecutarse nuevamente deberá iniciar un nuevo ciclo de vida (págs. 50 - 51)





**Figura 5.** Estados de una Activity

**Fuente:** Adaptado de Ribas (2013, pág. 52)

### 2.2.3.10 Acceso a Servicios web SOAP en Android

Para acceder a servicios web es necesario hacer uso de una librería externa. Se debe utilizar una librería que permita de forma cómoda y fácil acceder a servicios web de estándar SOAP, tomando como referencia se menciona a la librería ksoap2-android que es adaptada especialmente a Android. (Gómez Oliver, 2012)

Según Gómez Oliver (2012), en la implementación del evento onClick del botón, que será el encargado de comunicarse con el servicio web y procesar el resultado, se definirán cuatro constantes que servirán en varias ocasiones durante el código:

- **NAMESPACE.** Espacio de nombres utilizado en nuestro servicio web.
- **URL.** Dirección URL para realizar la conexión con el servicio web, se debe tener en cuenta el puerto que se está ejecutando en la máquina.
- **METHOD\_NAME.** Nombre del método web concreto que vamos a ejecutar.
- **SOAP\_ACTION.** Equivalente al anterior, pero en la notación definida por SOAP.

Asignación de valores a las constantes:

```
String NAMESPACE = "http://sgoliver.net/";
String URL="http://10.0.2.2:1473/ServicioClientes.asmx";
String METHOD_NAME = "NuevoClienteSimple";
String SOAP_ACTION = "http://sgoliver.net/NuevoClienteSimple";
```

**Figura 6.** Ejemplo de asignación de valores a las constantes

**Fuente:** Gómez Oliver (2012)

Gómez Oliver (2012) menciona que los pasos del proceso de acceso a un servicio web serán crear la petición SOAP al servicio web, enviarla al servidor y recibir la respuesta por lo que a continuación se explica cada uno:

Para crear la petición (request) al método, se debe crear un nuevo objeto SoapObject pasándole el namespace y el nombre del método web. A esta petición se debe asociar los parámetros de entrada mediante el método addProperty() al que se pasarán los nombres y valores de los parámetros.

```
SoapObject request = new SoapObject(NAMESPACE, METHOD_NAME);
request.addProperty("nombre", txtNombre.getText().toString());
request.addProperty("telefono", txtTelefono.getText().toString());
```

**Figura 7.** Ejemplo petición (request)

**Fuente:** Gómez Oliver (2012)

Se debe crear el contenedor SOAP (envelope) y asociarle la petición (request). Para ello se debe crear un nuevo objeto SoapSerializationEnvelope indicando la versión de SOAP que se va a usar. Además se debe indicar que se trata de un servicio web. Finalmente se debe asociar la petición antes creada al contenedor llamando al método setOutputSoapObject ().

```
SoapSerializationEnvelope envelope =
    new SoapSerializationEnvelope(SoapEnvelope.VER11);

envelope.dotNet = true;

envelope.setOutputSoapObject(request);
```

**Figura 8.** Ejemplo de creación del contenedor SOAP (envelope)

**Fuente:** Gómez Oliver (2012)

Para pasar la URL de conexión al servicio web, se debe crear un objeto de tipo HttpTransportSE, que se encargará de establecer comunicación HTTP con el servidor. Se utiliza el método Call() para realizar la llamada al servicio web.

```
HttpTransportSE transporte = new HttpTransportSE(URL);

try
{
    transporte.call(SOAP_ACTION, envelope);

    //Se procesa el resultado devuelto
    //...
}
catch (Exception e)
{
    txtResultado.setText("Error!");
}
```

**Figura 9.** Ejemplo de objeto para comunicación HTTP

**Fuente:** Gómez Oliver (2012)

Para obtener el resultado devuelto tras realizar la llamada al servidor, se utiliza el método `getResponse()`. Se convertirá a objeto `SoapPrimitive` si el resultado que se espera es un valor simple (número entero) o se puede realizar la conversión a una cadena de caracteres. Esta conversión depende del resultado que se espera.

```
SoapPrimitive resultado_xml =(SoapPrimitive)envelope.getResponse();
String res = resultado_xml.toString();

if(res.equals("1"))
    txtResultado.setText("Insertado OK");
```

*Figura 10.* Ejemplo de método `getResponse`

*Fuente:* Gómez Oliver (2012)

## **2.2.4 PROGRAMACIÓN ORIENTADA A OBJETOS**

### **2.2.4.1 Concepto**

Según Llobet Azpitarte et al. (2008), un lenguaje de programación no es orientado a objeto solo porque está basado en objetos sino que también debe tener clases y relaciones de herencia entre ellas, además se basa en la idea natural de la existencia de un mundo lleno de objetos. La programación orientada a objetos ha dado mejoras de alcance alto en el diseño, desarrollo y mantenimiento del software. (pág. 12)

#### **2.2.4.2 Propiedades de la programación orientada a objetos**

##### **a) Objetos**

De acuerdo con Llobet Azpitarte et al. (2008), un objeto es un conjunto complejo de datos y programas, poseen estructura de datos y forman parte de una organización jerárquica. Contiene en su interior un número de componentes estructurados por lo que no es un dato simple. (pág. 13)

##### **b) Las clases**

Una clase es la descripción de una familia de objetos que tienen la misma estructura (atributos) y el mismo comportamiento (métodos). (Llobet Azpitarte et al., 2008, pág. 15)

Terrero y Paredes (2011) consideran que una clase es una plantilla que define atributos u objetos con propiedades, y métodos. (pág. 37)

##### **c) Atributos**

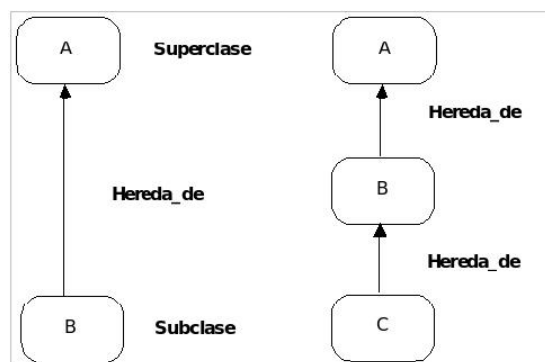
De acuerdo con Carballo (2010), un atributo es una característica de un objeto. Además un atributo sirve para definir información que se encontrará oculta dentro de un objeto.

#### d) Métodos

Terrero y Paredes (2011) afirman que, los métodos son las acciones u operaciones que puede realizar la clase, son declarados con paréntesis en los que se pueden ubicar valores comúnmente conocidos como parámetros. (pág. 39)

#### e) Herencia

Llobet Azpitarte et al. (2008) explican que el comportamiento y los datos de las clases hijas son siempre una extensión de las propiedades asociadas a las clases padres. Es una propiedad que permite a los ejemplares de una subclase tener acceso a métodos y datos de una superclase. (pág. 19)



*Figura 11.* Esquema de Herencia

*Fuente:* Llobet Azpitarte et al. (2008), pág. 19

#### f) Encapsulamiento

Es un mecanismo que consiste en organizar datos y métodos de una estructura, es decir, evita el acceso a datos por cualquier medio distinto a los especificados, por lo que garantiza la integridad de los datos que contiene un objeto. (Terrero & Paredes, 2011, pág. 35)

#### g) Polimorfismo

Para Terrero y Paredes (2011), es la capacidad de un objeto para adquirir varias formas. El uso del polimorfismo se da cuando se utiliza la referencia de una clase padre, para referirse al objeto de la clase hijo. (pág. 36)

### 2.2.5 ENTORNO DE DESARROLLO

#### 2.2.5.1 IDE Android Studio

Google Inc. (2009) explica que:

Android Studio es un nuevo entorno de desarrollo de Android basado en IntelliJ IDEA. Ofrece nuevas características y mejoras con respecto a Eclipse ADT. En la parte superior de las capacidades que usted espera de IntelliJ, Android Studio ofrece:

- ✓ Sistema de construcción a base de Gradle flexible.
- ✓ Construir variantes y generación de múltiple APK.
- ✓ Extensas plantillas soportadas por los servicios de Google y varios tipos de dispositivos.
- ✓ Editor de diseño Rich con soporte para la edición de tema.
- ✓ Herramientas para capturar rendimiento, usabilidad, compatibilidad de versiones, y otros problemas
- ✓ ProGuard y aplicación de firma de capacidades.

El IDE Android Studio es el entorno de desarrollo Android creado por Google, posee características que facilitan el desarrollo, la refactorización es más potente, incorpora una herramienta para el análisis del código, la edición es más fluida, además de que permite la previsualización de los recursos.

### **2.2.5.2 Android SDK**

De acuerdo con Ramírez Hernández (2011), el android sdk se trata de un conjunto de herramientas de desarrollo que sirven para construir, probar y depurar aplicaciones para Android. Se trata de una interfaz de programación de aplicaciones que utiliza el lenguaje de programación java.

El kit de desarrollo de software de Android (Android SDK) contiene las herramientas necesarias para crear, compilar y empaquetar aplicaciones de Android. La mayoría de estas herramientas son basadas en línea de comandos. La principal manera de desarrollar aplicaciones para Android se basa en el lenguaje de programación Java. (Vogel, 2014)

### **2.2.5.3 Xml**

Sagástegui Lescano (2008) menciona que XML es un lenguaje de etiquetas, en la que cada paquete está delimitado por dos etiquetas como se hace en otros lenguajes como html, con la diferencia de que en xml se separa el contenido de la presentación. Nació como una forma de reducir SGML usando la gramática de lenguajes específicos. Sus características más importantes son que permite separar el código de la presentación y que es completamente extensible mediante el uso de nuevas etiquetas creadas por el desarrollador.

De acuerdo con Lamarca Lapuente (2013), XML son las siglas del Lenguaje de Etiquetado Extensible. Con la palabra "Extensible" se alude a la no limitación en el número de etiquetas, ya que permite crear aquellas que sean necesarias. XML es un lenguaje que permite jerarquizar y estructurar la información y describir los contenidos dentro del propio documento, así como la reutilización de partes del mismo

#### 2.2.5.4 Lenguaje de programación Java

Según Terrero y Paredes (2011):

La compañía Sun describe el lenguaje Java como: “simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico. (pág. 3)

“Java es un lenguaje de programación de alto nivel con el que se pueden escribir tanto programas convencionales como para internet.” (Ceballos, 2011, pág. 22)

Java es un lenguaje de programación multiplataforma, orientado a objetos y basado en clases.

#### 2.2.5.5 PostgreSQL

Gibert Ginestà y Pérez Mora (2012) consideran que para relacionar la versión con capacidad de programación SQL y Postgres se cambió el nombre a PostgreSQL. Con PostgreSQL, el énfasis ha pasado a aumentar características y capacidades, aunque el trabajo continúa en todas las áreas. Es un sistema de gestión de base de datos de código abierto, objeto relacional, es distribuido y trabaja bajo la licencia BSD. Utiliza multiprocesos en vez de multihilos para brindar estabilidad al sistema, además de que usa el modelo cliente/servidor.

De acuerdo con Denzer (2002), PostgreSQL es un sistema gestor de base de datos relacionales avanzado basado en código abierto, lo que quiere decir que el código fuente de este programa está disponible a cualquier persona, está bajo licencia BSD. Permite la manipulación bajo las reglas del álgebra relacional, las tablas se pueden relacionar unas a otras haciendo uso de llaves, además los datos se almacenan en tablas conformadas por columnas y renglones.

##### a) Las principales mejoras en PostgreSQL incluyen:

Para Gibert Ginestà y Pérez Mora (2012) las mejoras de postgresQL son:

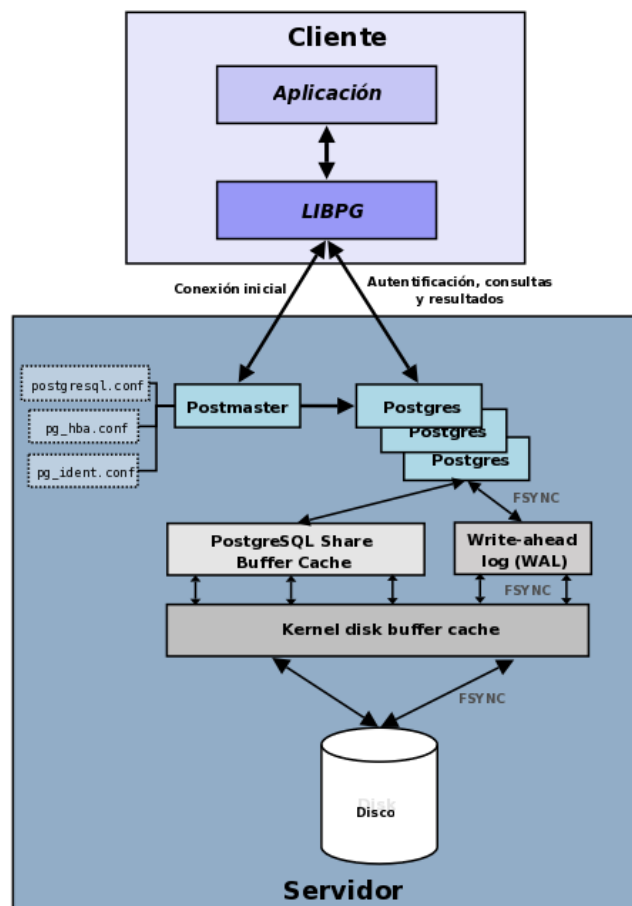
- ✓ Los bloqueos de tabla han sido sustituidos por el control de concurrencia multi-versión.
- ✓ Se han implementado importantes características del motor de datos, incluyendo subconsultas, valores por defecto, restricciones a valores en los campos (constraints) y disparadores (triggers).
- ✓ Se han añadido funcionalidades en línea con el estándar SQL92
- ✓ Los tipos internos han sido mejorados
- ✓ La velocidad del código del motor de datos ha sido incrementada aproximadamente en un 20-40%, y su tiempo de arranque ha bajado el 80% desde que la versión 6.0 fue lanzada.

##### b) Componentes de un sistema PostgreSQL

Martínez (2010) explica los componentes:

- ✓ **Aplicación cliente:** Esta es la aplicación cliente que utiliza PostgreSQL como administrador de bases de datos.

- ✓ **Demonio postmaster:** Escucha por un puerto/socket por conexiones entrantes de clientes.
- ✓ **Ficheros de configuración:** Los utilizados por PostgreSQL, postgresql.conf, pg\_hba.conf y pg\_ident.conf
- ✓ **Procesos hijos postgres:** Se encargan de autentificar a los clientes, de gestionar las consultas y mandar los resultados a las 0 clientes.
- ✓ **PostgreSQL share buffer cache:** Memoria compartida usada por PostgreSQL para almacenar datos en caché.
- ✓ **Write-Ahead Log (WAL):** Componente del sistema encargado de asegurar la integridad de los datos (recuperación de tipo REDO).
- ✓ **Kernel disk buffer cache:** Caché de disco del sistema operativo
- ✓ **Disco:** Disco físico donde se almacenan los datos y toda la información necesaria para que PostgreSQL funcione.



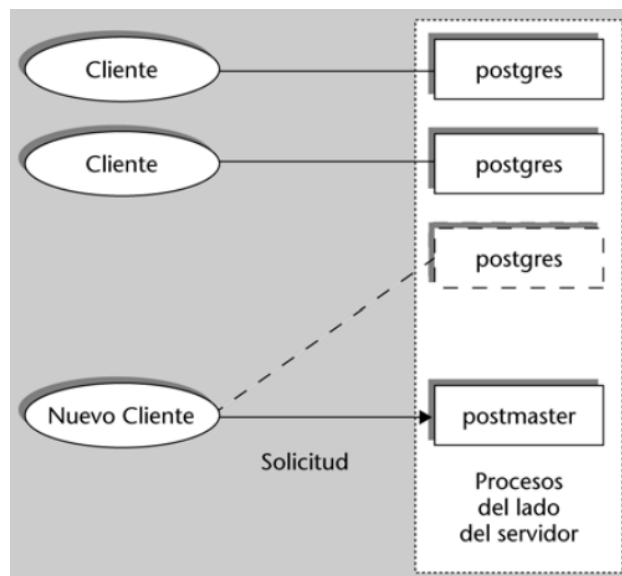
**Figura 12.** Componentes de un Sistema PostgreSQL

*Fuente:* Martínez (2010)

### c) Arquitectura de PostgreSQL

Según Gibert Ginestà y Pérez Mora (2012), PostgreSQL está basado en una arquitectura cliente-servidor. El programa servidor se llama **Postgres** y entre los muchos programas cliente tenemos, por ejemplo, **pgaccess** (un cliente gráfico) y **psql** (un cliente en modo texto). Un proceso servidor *postgres* puede atender exclusivamente a un solo cliente; es decir, hacen falta tantos procesos servidor *postgres* como clientes haya. Cuando los clientes realicen una petición de conexión el encargado de ejecutar un servidor para cada uno es el proceso **postmaster**. El proceso postmaster se ejecuta en un sitio (*host*), pero solo se puede ejecutar un proceso postmaster y varios procesos *postgres*. (pág. 65)

El siguiente gráfico muestra de forma esquemática las entidades involucradas en el funcionamiento normal del gestor de bases de datos:



**Figura 13.** Arquitectura de PostgreSQL

**Fuente:** Gibert Ginestà & Pérez Mora (2012), pág. 65

### 2.2.5.6 Python

González Duque (2010) afirma que principio de los años 90, Guido Van Rossum creó un lenguaje de programación inspirado en el grupo de cómicos ingleses “Monty Python”. Python es un lenguaje de programación orientado a objetos, multiparadigma, multiplataforma, con estructuras de datos de alto nivel a través de un tipado dinámico. Brinda una sintaxis muy simple y limpia que ayuda a crear un código legible. (pág. 7)

De acuerdo con Álvarez M. (2003), Python ofrece rapidez en el desarrollo de aplicaciones, es un lenguaje de programación scripting, que trabaja independiente de la plataforma, no se necesita compilar el código para que se ejecute por lo que se lo conoce como lenguaje interpretado.



### **a) Ventajas de Python**

Para González Duque (2010) las ventajas son las siguientes:

- ✓ Lenguaje expresivo
- ✓ Python es muy legible. Permite la escritura de programas cuya lectura resulta más fácil.
- ✓ Ofrece un entorno interactivo que facilita la realización de pruebas.
- ✓ El entorno de ejecución de Python detecta muchos de los errores de programación que escapan al control de los compiladores.
- ✓ No puede usarse como lenguaje imperativo procedimental o como lenguaje orientado a objetos.
- ✓ Posee estructuras de datos que se pueden manipular de modo Sencillo. (págs. 7-8)

### **2.2.5.7 Servicios Web**

De acuerdo con Ramos Muñoz (2013), los servicios web son tecnologías que permiten en intercambio de datos entre sí. Además tienen capacidad de interoperar en la red.

Para Pastorini (2006), un servicio web especifica un conjunto de operaciones, por medio de la URL, desde la cual un cliente remota lo puede consumir. Un servicio web proporciona una serie de servicios listos para ser consumidos a través de la red.

### **a) ¿Para qué sirven los Servicios Web?**

Permite al usuario visualizar información dinámica mediante la interacción de aplicaciones entre sí que se comunican a través de mecanismos estándares. Además hace posible la combinación entre aplicaciones para poder realizar operaciones complejas, y de esta manera ofrecer interoperabilidad y extensibilidad entre ellas. (Ramos Muñoz, 2013)

### **2.2.5.8 Soap (Simple Object Access Protocol)**

#### **a) Introducción**

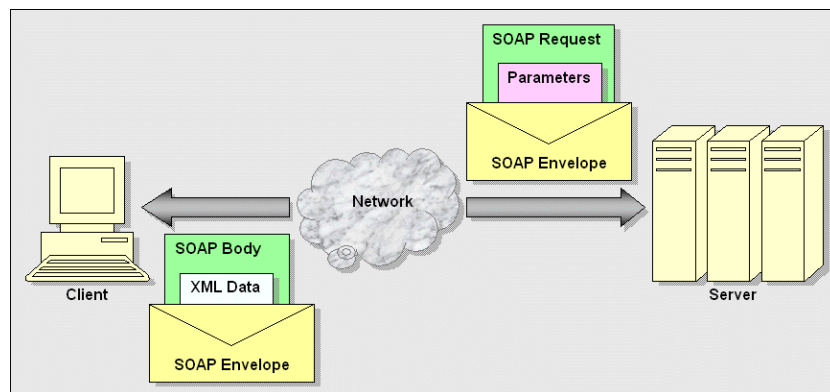
Moscatelli (2008) considera que a finales del año de 1998, a través de una especificación realizada por Dave Winer, de la empresa UserLand Software surge SOAP (Simple Object Access Protocol) un mecanismo basado en XML para realizar llamadas RPC. Aproximadamente en septiembre de 1999 publican la versión 0.9 de SOAP, tomando como base la especificación realizada y la unión de varias empresas. Meses más tarde fue lanzada la versión 1.0, la que fue reconocida como versión oficial por el organismo de IETF (Internet Engineering Task Force). Luego en mayo del año 2000 fue enviada a la W3C (World Wide Web Consortium) la versión 1.1 con el fin de que forme parte del grupo de protocolos basados en xml, para que meses más tarde la organización edite la versión 1.2 de SOAP el W3C Working Draft. Esta especificación describe básicamente un formato de

comunicación entre aplicaciones, para lograr esto combina tecnologías existentes y aceptadas en la industria del software. En particular, combina XML para la codificación de los mensajes y HTTP como protocolo de transporte (aunque no se excluye el uso de otros protocolos de transporte). (pág. 3)

**b) ¿Qué es SOAP?**

De acuerdo con Moscatelli (2008), SOAP es un mecanismo simple y liviano para la comunicación, en un entorno distribuido o descentralizado, entre componentes de software o aplicaciones. SOAP no establece el uso de un protocolo para el transporte, por lo que la comunicación se realiza a través de mensajes codificados en XML. Por esta razón se define una técnica para el intercambio de información, tipeada y estructurada, entre dos aplicaciones dentro de un entorno distribuido, queriendo lograr la simplicidad y la extensibilidad en el diseño. SOAP expresa la semántica de las aplicaciones, a través de un mecanismo simple que le permite crear un modelo modular empaquetado de mensajes y la definición de como codificar los datos de las aplicaciones de dichos módulos. (pág. 4)

**c) Arquitectura básica de SOAP**



**Figura 14.** Arquitectura básica de SOAP

**Fuente:** <http://www.fing.edu.uy/>

Moscatelli (2008) explica que:

En la figura se observa la arquitectura básica de un sistema, construida sobre SOAP y los mensajes que definen la interacción entre la aplicación cliente y la aplicación servidor. Generalmente la aplicación cliente envía un mensaje (REQUEST vía HTTP), el cual al ser recibido por la aplicación servidor genera una respuesta (RESPONSE) que es enviada a la aplicación cliente vía HTTP. Se observa además que en el caso de usar SOAP para realizar RPC, la invocación RPC se mapea naturalmente al REQUEST de HTTP y la respuesta RPC se mapea al RESPONSE de HTTP. (pág. 5)

#### d) Protocolo SOAP

Según Moscatelli (2008) la especificación del protocolo SOAP indica que el mismo consiste de 3 partes:

- ✓ El constructor SOAP ENVELOPE que define un framework para expresar qué hay en un mensaje, a quién está dirigido el mensaje y cuando es opcional o mandatorio.
- ✓ Las reglas de codificación que definen un mecanismo de serialización para ser usado para intercambiar instancias de tipos de datos.
- ✓ La representación SOAP RPC que define una metodología que puede ser usada para representar invocaciones a procedimientos remotos y sus respuestas (pág. 6).

#### 2.2.5.9 Wsdl

WSDL es sinónimo de Web Services Description Language. WSDL es un lenguaje para describir servicios Web y cómo acceder a ellos, está escrito en XML. (Esquivia Rodríguez, 2013)

Es una gramática XML que nos indica cuáles son las interfaces que provee el Servicio web y los tipos de datos necesarios para su utilización. (Tajes Martínez, 2008)

##### a) Documento WSDL

Un documento WSDL es un documento XML simple. Contiene un conjunto de definiciones para describir un servicio web utilizando estos elementos principales:

Elemento	Descripción
<types>	Un contenedor para las definiciones de tipos de datos que utiliza el servicio web
<message>	Una definición con tipo de los datos que se comunica
<portType>	Un conjunto de operaciones con el apoyo de uno o más puntos finales
<binding>	Una especificación de protocolo y datos de formato para un tipo de puerto en particular

**Tabla 6.** Elementos para describir un Servicio Web

**Fuente:** Refsnes Data Inc (2012)

Un documento WSDL puede contener también otros elementos, como elementos de extensión, y un elemento de servicio que permite agrupar las definiciones de varios servicios web en un documento único WSDL. (Esquivia Rodríguez, 2013)

De acuerdo con Acedo (2012), el documento de WSDL define los servicios como colecciones de puntos finales de red o puertos.

#### 2.2.5.10 ¿Qué es JSON?

De acuerdo con Esquivia (2013) Json es:

Es un formato para el intercambio de datos, que describe los datos con una sintaxis dedicada y se usa para identificar y gestionar los datos. JSON nació como una alternativa a XML, el fácil uso en javascript, una de las mayores ventajas que tiene el uso de JSON es que puede ser leído por cualquier lenguaje de programación. Por lo tanto, puede ser usado para el intercambio de información entre distintas tecnologías.

Para Sánchez (2008), JSON es un subconjunto del lenguaje javascript que se basa en la construcción de una lista ordenada de valores, listas de objetos, que pueden incluir a su vez tablas hash, objetos con una colección de pares nombre/valor.

Esta sintaxis JSON define un objeto empleados, con una serie de 3 registros de empleados (objetos):

```
JSON Example
{
  "employees": [
    { "firstName": "John", "lastName": "Doe" },
    { "firstName": "Anna", "lastName": "Smith" },
    { "firstName": "Peter", "lastName": "Jones" }
  ]
}
```

*Figura 15.* Ejemplo definición de objeto Json

*Fuente:* Refsnes Data Inc (2012)

### 2.2.6 METODOLOGÍA MOBILE-D

VTT Technology for business (2007) afirma que:

Es un metodología para el desarrollo ágil de software, que no solamente está orientada al desarrollo de aplicaciones móviles, también se puede usar en aplicaciones de seguridad, financieras, de logísticas, y de simulación.

Mobile-D se basa en la programación extrema (XP) para la implementación, Cristal methodologies para la escalabilidad y en el Proceso Unificado de Desarrollo (RUP) para la cobertura del ciclo de vida.

- La fase de exploración, siendo ligeramente diferente del resto del proceso de producción, se dedica al establecimiento de un plan de proyecto y los conceptos básicos, por lo tanto, se puede separar del ciclo principal de desarrollo (aunque

no debería obviarse). Los autores de la metodología ponen además especial atención a la participación de los clientes en esta fase.

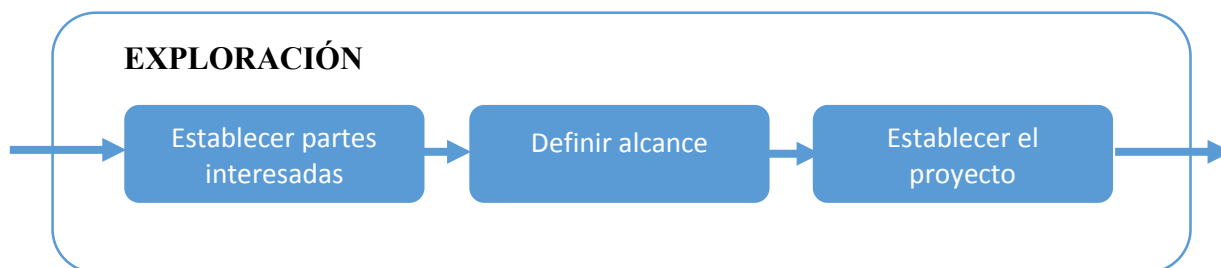
- Durante la fase de inicialización, los desarrolladores preparan e identifican todos los recursos necesarios. Se preparan los planes para las siguientes fases y se establece el entorno técnico (incluyendo el entrenamiento del equipo de desarrollo). Los autores de Mobile-D afirman que su contribución al desarrollo ágil se centra fundamentalmente en esta fase, en la investigación de la línea arquitectónica. Esta acción se lleva a cabo durante el día de planificación.
- En la fase de "producción" se repite la programación de tres días (planificación, trabajo, liberación) se repite iterativamente hasta implementar todas las funcionalidades. Primero se planifica la iteración de trabajo en términos de requisitos y tareas a realizar. Se preparan las pruebas de la iteración de antemano (de ahí el nombre de esta técnica de TestDriven Development, TDD). Las tareas se llevarán a cabo durante el día de trabajo, desarrollando e integrando el código con los repositorios existentes. Durante el último día se lleva a cabo la integración del sistema (en caso de que estuvieran trabajando varios equipos de forma independiente) seguida de las pruebas de aceptación.
- En la fase de estabilización, se llevan a cabo las últimas acciones de integración para asegurar que el sistema completo funciona correctamente. Esta será la fase más importante en los proyecto multi-equipo con diferentes subsistemas desarrollados por equipos distintos. En esta fase, los desarrolladores realizarán tareas similares a las que debían desarrollar en la fase de "productización o producción", aunque en este caso todo el esfuerzo se dirige a la integración del sistema. Adicionalmente se puede considerar en esta fase la producción de documentación.
- La última fase (prueba y reparación del sistema) tiene como meta la disponibilidad de una versión estable y plenamente funcional del sistema. El producto terminado e integrado se prueba con los requisitos de cliente y se eliminan todos los defectos encontrados.

## CÁPITULO III

### 3. DESARROLLO DE LA PROPUESTA

#### 3.1 EXPLORACIÓN E INICIALIZACIÓN

El objetivo de esta fase es la planificación y el establecimiento de las bases del proyecto, en la cual se obtendrá como resultado la recolección de requisitos funcionales como no funcionales, que servirán para conceptualizar el modelamiento y lógica del sistema.



*Figura 16. Etapas de la Fase de Exploración*

#### 3.1.1 MODELAMIENTO DE NEGOCIO

##### 3.1.1.1 Visión General del Proyecto

El objetivo del proyecto es desarrollar una aplicación móvil para automatizar los procesos de la Gestión de Actividades Operativas de Coniel Cia.Ltda, en el que se implementen Web Services para la transferencia de información.

##### 3.1.1.2 Establecimiento de las partes interesadas (Stakeholder Establishment)

El propósito de esta etapa es identificar y establecer los grupos de interés o personas involucradas, las mismas que tienen un rol en alguna tarea durante el desarrollo del software, excluyendo al equipo de desarrollo en sí.

<b>ROL</b>	Jefe de Proyecto
<b>Nombres</b>	Ing. Fausto Fabián
<b>Apellidos</b>	Redrován Castillo.
<b>Responsabilidades</b>	Es el encargado de planificar, ejecutar y controlar el desarrollo del proyecto. Ayuda al equipo en la toma de decisiones dentro del ámbito y objetivo del proyecto.

*Tabla 7. Rol de Jefe de Proyecto*

*Elaborado por:* Andrea Loaiza

### 3.1.2 MODELAMIENTO DE REQUERIMIENTOS DEL SISTEMA

El objetivo de esta etapa es definir las metas del proyecto, se realizará la definición inicial de los requisitos funcionales y no funcionales del sistema.

#### 3.1.2.1 Definición de Requisitos iniciales

Se generará una visión inicial de la funcionalidad del software, además los requisitos documentados serán la base para empezar con el desarrollo del sistema ya que son tomados en cuenta desde el punto de vista del usuario. El propósito será identificar las funcionalidades y los requerimientos no funcionales para el desarrollo del software.

- **Requisitos Funcionales**

Se identifican las funcionalidades del software:

FUNCIÓN	# REQ	REQUISITO	RESTRICCIONES
<b>Login de la Aplicación</b>	R1	Iniciar Sesión en la aplicación	Validar datos
	R2	Ingreso a la aplicación	Previo R1
<b>Ingreso de Actividades</b>	R3	Presentación de listado de pasos a realizar para ingresar la actividad	Previo R2
	R4	Ingresar información de la actividad que se realiza	R4
	R5	Almacenar la información ingresada en la base de datos	Previo R4
<b>Búsqueda</b>	R6	Consultar información de abonados según criterios	Previo R2
	R7	Visualizar información del abonado y de los medidores asignados a él.	Previo R6
<b>Gestión de Fotos</b>	R8	Visualizar lista de fechas en las que se han capturado fotos	Previo R2
	R9	Visualizar cuentas según fecha seleccionada	Previo R8
	R10	Capturar fotos dentro de la lista de cuentas según la fecha seleccionada.	Previo R9
	R11	Visualizar galería de imágenes según la cuenta seleccionada de la lista	Previo R9
	R12	Agregar más fotos dentro de una cuenta cuando se muestre la galería de la misma.	Previo R11

	R13	Eliminar fotos de la galería	Previo R11
	R14	Visualizar en tamaño original las imágenes	Previo R11
<b>Geolocalización</b>	R15	Visualizar posición actual	Previo R2
	R16	Trazar ruta entre dos cuadrillas seleccionadas	Previo R2
	R17	Visualizar distancia entre dos puntos	Previo R16
	R18	Visualizar tiempo entre dos puntos	Previo R16
<b>Salir</b>	R19	Salir de la aplicación (Desloguear)	Previo R2

**Tabla 8.** Matriz de Requisitos Funcionales

*Elaborado por:* Andrea Loaiza

#### ▪ **Requisitos No Funcionales**

Se describirán ciertas características que limitan el sistema.

# REQ	REQUISITO
R20	El sistema debe contar con un Login, para mayor seguridad.
R21	Interfaz amigable
R22	Fácil usabilidad
R23	Ejecutarse en Tablet con sistema operativo Android.
R24	Mensajes de error
R25	Debe proveer de tiempos de respuestas rápidos.

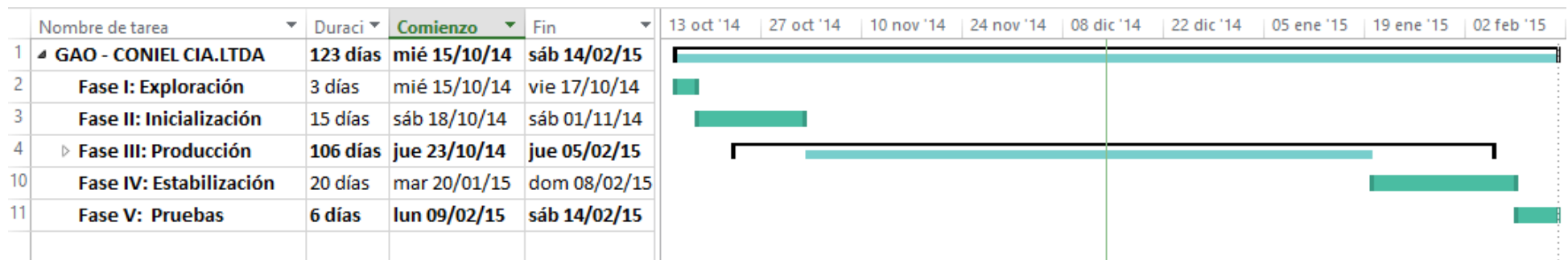
**Tabla 9.** Matriz de Requisitos No Funcionales

*Elaborado por:* Andrea Loaiza



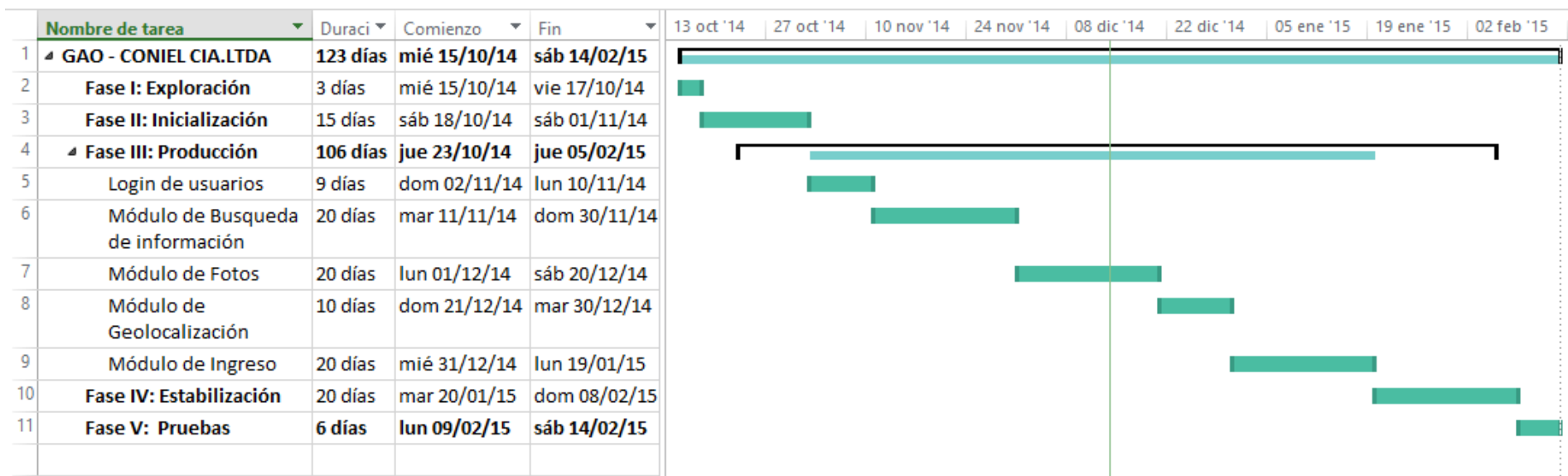
### 3.1.2.2 Definición de la aplicación del proyecto

- Cronograma del Proyecto



*Figura 17.* Cronograma de Trabajo por etapas metodológicas.

*Elaborado por:* Andrea Loaiza



**Figura 18.** Cronograma de la etapa de Producción

**Elaborado por:** Andrea Loaiza

- **Análisis de Costo**

Para el análisis de costo se toman en cuenta varios factores como la los equipos, recursos humanos, software, etc. El costo de este proyecto incluido la implementación se realiza en base a un tiempo estimado de 4 meses. Además para la estimación del costo se tomaran también en cuenta el software ya sea libre o no propietario que se utilizara para el desarrollo de la aplicación.

**a) Análisis de costo para el equipo de trabajo:**

ROL	MESES	PERSONAS	COSTO MENSUAL	COSTO TOTAL
Jefe de Proyecto	4	1	\$ 1200	\$ 4.800,00
Programador	4	1	\$ 900	\$ 3.600,00
Testing	4	1	\$ 500	\$ 2.000,00
<b>TOTAL</b>				\$10.400,00

*Tabla 10.* Análisis de Costo de Equipo de Trabajo

*Elaborado por:* Andrea Loaiza

**b) Análisis de costo por dispositivos móviles**

Dentro del análisis de costo se tomarán en cuenta los valores de los dispositivos en los que se ejecutará la aplicación.

ITEM	CANTIDAD	VALOR UNITARIO	COSTO TOTAL
Smartphones	2	\$ 215,00	\$ 430,00
Tablets	2	\$ 380,00	\$ 760,00
<b>TOTAL</b>			\$ 1.190,00

*Tabla 11.* Análisis de Costos por dispositivos móviles

*Elaborado por:* Andrea Loaiza

**c) Costo Total del proyecto**

Para calcular el costo total de producción del sistema se deben sumar los costos calculados anteriormente:

ITEM	COSTO TOTAL
Costo Equipo de Trabajo	\$10.400,00
Costo de dispositivos móviles	\$ 1.190,00
Paquete de datos	\$ 140,00
<b>TOTAL</b>	<b>\$ 11.730,00</b>

*Tabla 12.* Costo total del proyecto

*Elaborado por:* Andrea Loaiza

Luego de haber realizado el análisis de costos se ha demostrado que el Costo Total del desarrollo de la aplicación es de \$ 11.730,00.

▪ **Recursos Humanos**

Se asignarán roles, responsabilidades y funciones a las personas involucradas en desarrollo del software.

ROL	RESPONSABILIDAD	RECURSO HUMANO
Jefe de Proyecto	Es el encargado de planificar, ejecutar y controlar el desarrollo del proyecto. Ayuda al equipo en la toma de decisiones dentro del ámbito y objetivo del proyecto.	Ing. Fausto Fabián Redrován Castillo
Programadora	Codifica las funcionalidades del software.	Srta. Andrea Anabell Loaiza Gonzaga
Testing	Realiza pruebas escritas por el desarrollador y además brinda soporte para las pruebas.	Srta. Andrea Anabell Loaiza Gonzaga
Cliente	Brinda la información necesaria para establecer los requisitos funcionales y no funcionales del sistema.	Representante Legal de CONIEL Cia.Ltda

*Tabla 13.* Definición de Roles y Responsabilidades

*Elaborado por:* Andrea Loaiza

## 3.2 PRODUCCIÓN, ESTABILIZACIÓN Y PRUEBAS

### 3.2.1 ANÁLISIS DEL SISTEMA

#### 3.2.1.1 Tarjetas de Historias de Usuarios (Story Cards)

Las historias de usuario (Story Cards), son expresiones en lenguaje cotidiano que permiten capturar los procesos que realiza un usuario, que sirven para definir las funciones de un sistema.

##### a) Story Card del Módulo de Login de Usuarios

Número / ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Empleado		
01	Nuevo	Fácil	Fácil	4	5	Baja	
	Fijo	Mejorado	Mejorado			Media	
	Mejora	Duro	Duro			Alta	
Descripción							
Para el inicio de sesión en la aplicación móvil, el trabajador deberá ingresar el usuario y contraseña que se le haya asignado. Cuando se intente iniciar sesión se verificarán los datos ingresados, por lo que si los datos son correctos se le redireccionará al menú principal de la aplicación, caso contrario visualizará un mensaje de error en el que se pide la comprobación de los datos.							
Fecha	Estado			Comentario			
02/11/2014	Definido						
05/11/2014	Implementado						
06/11/2014	Hecho						
10/11/2014	Verificado						
-	Propuesto / Cancelado / Comparado						

*Story Card 1.* Login de Usuarios

*Elaborado por:* Andrea Loaiza

**b) Story Card del Módulo de Búsqueda de Información**

Número / ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Empleado		
02	Nuevo	Fácil	Fácil	3	5	Baja	
	Fijo	Mejorado	Mejorado			Media	
	Mejora	Duro	Duro			Alta	
Descripción							
Para acceder a la búsqueda de información el usuario debe estar logueado. Podrá realizar consultas por criterio de abonados, mediante la transferencia de información por Web Services y conexión al sistema comercial SICO. El usuario tendrá opciones de realizar consultas por cuenta, nombre, geocódigo y medidor.							
Fecha		Estado		Comentario			
11/11/2014		Definido					
25/11/2014		Implementado					
26/11/2014		Hecho					
30/11/2014		Verificado					
-		Propuesto / Cancelado / Comparado					

*Story Card 2. Búsqueda de Información*

*Elaborado por:* Andrea Loaiza

c) **Módulo de Fotos**

Número / ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Empleado		
03	Nuevo	Fácil	Fácil	4	5	Baja	
	Fijo	Mejorado	Mejorado			Media	
	Mejora	Duro	Duro			Alta	
Descripción							
Los usuarios podrán visualizar una lista con las fechas en las que se han registrado imágenes, así mismo dependiendo de la fecha que seleccionen visualizarán una lista de cuentas de las actividades realizadas en el día, por lo que el sistema debe permitir que al seleccionar una cuenta de la lista se carguen en una galería las imágenes que han sido capturadas dentro de esa carpeta. Además se podrá visualizar las imágenes en tamaño original, permitirá agregar, eliminar fotos y almacenar la ubicación en la que está realizando la actividad.							
Fecha		Estado		Comentario			
01/12/2014		Definido					
12/12/2014		Implementado					
15/12/2014		Hecho					
20/12/2014		Verificado					
-		Pospuesto / Cancelado / Comparado					

*Story Card 3. Fotos*

*Elaborado por:* Andrea Loaiza

**d) Módulo de Geolocalización**

Número / ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Empleado		
04	Nuevo	Fácil	Fácil	3	5	Baja	
	Fijo	Mejorado	Mejorado			Media	
	Mejora	Duro	Duro			Alta	
Descripción							
El usuario visualizará el mapa en el que podrá ubicar su posición actual, además de que podrá seleccionar cuadrillas y trazar rutas entre ellas. También obtendrá la distancia y el tiempo que hay de un punto a otro.							
Fecha	Estado			Comentario			
21/12/2014	Definido						
25/12/2014	Implementado						
27/12/2014	Hecho						
30/12/2014	Verificado						
-	Pospuesto / Cancelado / Comparado						

*Story Card 4.* Geolocalización

*Elaborado por:* Andrea Loaiza



**e) Módulo de Ingreso de Actividad**

Número / ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Empleado		
05	Nuevo	Fácil	Fácil	4	5	Baja	
	Fijo	Mejorado	Mejorado			Media	
	Mejora	Duro	Duro			Alta	
Descripción							
El usuario visualizará una lista de pasos a realizar para el ingreso de la actividad. Además también podrá realizar consultas de datos de abonados para de esta manera corroborar la información, obtendrá listas de los materiales que utiliza en la actividad lo que permitirá tener un mejor control de los mismos. El ingreso de actividades debe contener información integra por lo que de esta manera se evitará la pérdida de los datos.							
Fecha		Estado		Comentario			
31/12/2014		Definido					
10/12/2014		Implementado					
13/12/2014		Hecho					
19/12/2014		Verificado					
-		Pospuesto / Cancelado / Comparado					

*Story Card 5. Ingreso de Actividades*

*Elaborado por:* Andrea Loaiza

### 3.2.1.2 Tarjetas de Tareas (Task Cards)

Las tarjetas de tareas son definidas por cada historia de usuario y sirven para llevar a cabo ordenes de trabajo destinadas al equipo de desarrollo, lo que permitirá obtener una estimación del calendario con la programación de las actividades o tareas a realizar.

#### a) Crear e Ingresar información a Base de Datos

Núm ./ ID		Dificultad		Confianza	Duración		
	Tipo	Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	Notas
01	Nuevo	Fácil	Fácil	4	1 día	3 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
<p>La compañía CONIEL Cia.Ltda ha proporcionado un archivo de Microsoft Excel con la información de inventarios actual en la que se puede encontrar el stock de materiales para un contrato en específico, por lo que es necesario almacenar esta información en una base de datos de tal manera que el usuario tenga acceso a la misma . Las tareas que se realizarán son las siguientes:</p> <ul style="list-style-type: none"><li>▪ Diseñar base de datos relacional</li><li>▪ Almacenar información proporcionada por la compañía en la base de datos</li></ul>							
Fecha		Estado		Comentario			
16/10/2014		Definido					
18/10/2014		Implementado					
18/10/2014		Hecho					
18/10/2014		Verificado					
-		Pospuesto / Cancelado / Comprobado					

*Task Card 1.* Crear e Ingresar información a Base de Datos

*Elaborado por:* Andrea Loaiza

**b) Creación de Clases y Diseño de Interfaces del Login de Usuario**

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
02	Nuevo	Fácil	Fácil	4	2 días	3 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
<p>Luego de tener la base de datos creada y con datos almacenados, se puede empezar a realizar el análisis y diseño de las interfaces y clases, con el fin de interrelacionarla a cada entidad con el objetivo de optimizar la ejecución de los procesos. Las tareas consistirán en:</p> <ul style="list-style-type: none"><li>✓ Analizar y diseñar un prototipo y relacionarlo con las entidades para construir clases e interfaces.</li><li>✓ Implementar el análisis y diseño realizado previamente, tomando en cuenta los atributos, datos, nomenclaturas, etc.</li></ul>							
Fecha		Estado		Comentario			
02/11/2014		Definido					
05/11/2014		Implementado					
06/11/2014		Hecho					
10/11/2014		Verificado					
		Pospuesto / Cancelado / Comprobado					

**Task Card 2.** Creación de Clases y Diseño de Interface del Login de Usuario

**Elaborado por:** Andrea Loaiza

c) Creación e Implementación de Web Services para la aplicación en el servidor

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
03	Nuevo	Fácil	Fácil	4	8 días	15 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
<p>Se crearán e implementarán Web Services para los procesos que lo requieran sobre el servidor, además se planificarán los mismos para su correcta interpretación. Por lo que esta tarea consistirá de las siguientes actividades:</p> <ul style="list-style-type: none"><li>✓ Analizar y diseñar Web Services para la aplicación</li><li>✓ Implementar el web Service tomando en cuenta atributos, tipos de datos y nomenclaturas, etc.</li></ul>							
Fecha	Estado			Comentario			
	Definido						
	Implementado						
	Hecho						
	Verificado						
	Pospuesto / Cancelado / Comprobado						

*Task Card 3.* Creación e Implementación de Web Services de la aplicación en el servidor

*Elaborado por:* Andrea Loaiza

**d) Creación de Clases e Interfaces para consumo de Web Services para los módulos de la aplicación**

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
04	Nuevo	Fácil	Fácil	4	4 días	8 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
Descripción							
<p>En esta tarea se elaborará la estructura para el consumo de Web Services que se tendrá en el diseño de la aplicación móvil, para que de este modo obtener una estructura lógica , que permita implementarlos de forma rápida mediante el análisis y diseño de las clases e interfaces para su posterior consumo. Las actividades que se deben realizar en esta tarea son:</p> <ul style="list-style-type: none"><li>✓ Realizar un análisis que entidades se utilizarán para el diseño de las clases e interfaces que son necesarias para el consumo de los Web Services.</li><li>✓ Implementar las clases para el consumo de Web Services en la aplicación tomando en consideración atributos, tipos de datos y nomenclaturas, etc.</li></ul>							
Fecha	Estado			Comentario			
	Definido						
	Implementado						
	Hecho						
	Verificado						
	Pospuesto / Cancelado / Comprobado						

**Task Card 4.** Creación de Clases e Interfaces para consumo de Web Services para los módulos de la aplicación

*Elaborado por:* Andrea Loaiza

**e) Creación de Clases y Diseño de Interfaces del módulo de Búsqueda de Información**

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
05	Nuevo	Fácil	Fácil	4	10 días	20 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
<p>Para el módulo de búsqueda de información a través de Web Services en el Sistema Comercial SICO, se deberán realizar las siguientes actividades:</p> <ul style="list-style-type: none"><li>✓ Analizar y Diseñar las clases e interfaces de usuarios que cumplan con los requerimientos tanto como para la interpretación de los Web Service como para el usuario.</li><li>✓ Implementar las clases y la interfaz considerando los atributos, tipos de datos y nomenclaturas de las entidades relacionadas con el proceso.</li></ul>							
Fecha	Estado			Comentario			
11/11/2014	Definido						
25/11/2014	Implementado						
26/11/2014	Hecho						
30/11/2014	Verificado						
-	Pospuesto / Cancelado / Comprobado						

*Task Card 5.* Creación de clases y diseño de interfaces para el módulo de Búsqueda de información

*Elaborado por:* Andrea Loaiza

**f) Creación de Clases y Diseño de Interfaces del módulo de Fotos**

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
07	Nuevo	Fácil	Fácil	4	10 días	20 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
En esta tarea se creará los prototipos de interfaz de usuario para definir las características, apariencia y funcionamiento de la misma. El diseño y funcionalidad que se desarrollarán durante esta tarea serán en base a los requerimientos del usuario. Por lo que esta tarea consiste en:  ✓ Analizar los procesos que realizan los usuarios y su entorno de trabajo. ✓ Prototipar la interfaz de usuario en base a los requisitos de usuarios. ✓ Implementar las clases y la interfaz considerando los atributos, tipos de datos y nomenclaturas de las entidades relacionadas con el proceso.							
Fecha		Estado		Comentario			
01/12/2014		Definido					
12/12/2014		Implementado					
15/12/2014		Hecho					
20/12/2014		Verificado					
		Pospuesto / Cancelado / Comprobado					

**Task Card 6.** Creación de Clases y Diseño de Interfaz del Módulo de Fotos

**Elaborado por:** Andrea Loaiza

**g) Creación de Clases y Diseño de Interfaces del módulo de Geolocalización**

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 .- Poca 4 .- Mucha	Estimado	Empleado	
07	Nuevo	Fácil	Fácil	4	4 días	10 días	
	Fijo	Mejorado o	Mejorado				
	Mejora	Duro	Duro				
	Descripción						
En esta tarea se creará los prototipos de interfaz de usuario del módulo de geolocalización para definir las características, apariencia y funcionamiento de la misma. El diseño y funcionalidad que se desarrollarán durante esta tarea serán en base a los requerimientos del usuario. Por lo que esta tarea consiste en:  ✓ Analizar los requerimientos de usuarios para identificar las necesidades. ✓ Prototipar la interfaz de usuario en base al análisis anterior. ✓ Implementar las clases y la interfaz considerando los atributos, tipos de datos y nomenclaturas de las entidades relacionadas con el proceso.							
Fecha		Estado		Comentario			
21/12/2014		Definido					
25/12/2014		Implementado					
27/12/2014		Hecho					
30/12/2014		Verificado					
		Pospuesto / Cancelado / Comprobado					

**Task Card 7.** Creación de Clases y Diseño de Interfaz del módulo de Geolocalización

**Elaborado por:** Andrea Loaiza



## h) Creación de Clases y Diseño de Interfaces del módulo de Ingreso de Actividades

Núm. / ID	Tipo	Dificultad		Confianza	Duración		Notas
		Antes	Después	1 (poca confianza) 4 (muchacha confianza)	Estimado	Empleado	
08	Nuevo	Fácil	Fácil	4	8 días	20 días	
	Fijo	Mejorado	Mejorado				
	Mejora	Duro	Duro				
Descripción							
<p>En esta tarea se creará los prototipos de interfaz de usuario del módulo de ingreso de actividades, para definir las características, apariencia y funcionamiento de la misma. El diseño y funcionalidad que se desarrollarán durante esta tarea serán en base a los requerimientos del usuario. Por lo que esta tarea consiste en:</p> <ul style="list-style-type: none"><li>✓ Identificar como ejecutan el proceso de ingreso de información el personal operativo y analizar el entorno de trabajo.</li><li>✓ Diseñar la interfaz de usuario dependiendo del análisis realizado anteriormente.</li><li>✓ Implementar las clases y la interfaz considerando los atributos, tipos de datos y nomenclaturas de las entidades relacionadas con el proceso.</li></ul>							
Fecha		Estado		Comentario			
31/12/2014		Definido					
10/12/2014		Implementado					
13/12/2014		Hecho					
19/12/2014		Verificado					
-		Pospuesto / Cancelado / Comprobado					

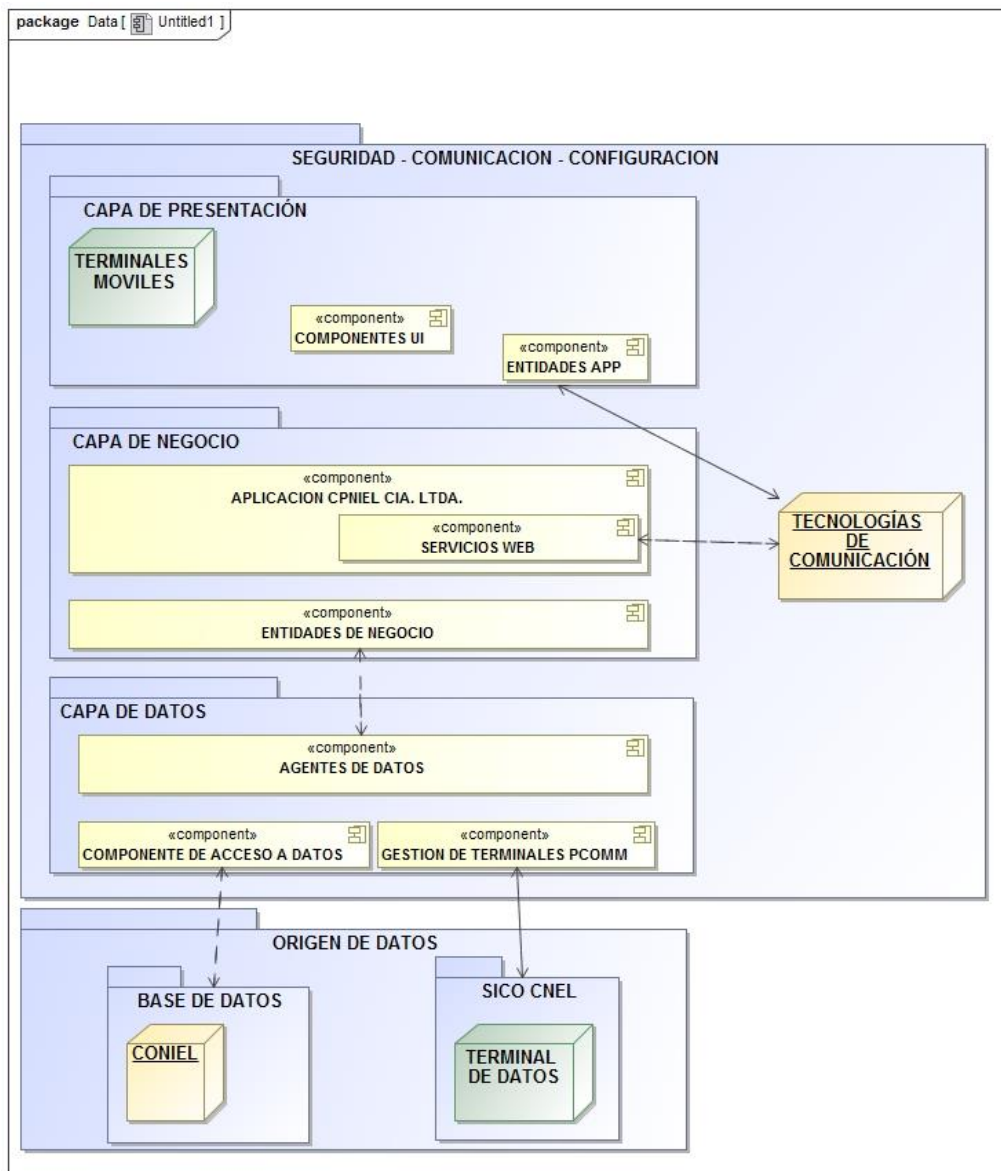
**Task Card 8.** Creación de Clases y Diseño de Interfaces de usuario del módulo de Ingreso de Actividades

**Elaborado por:** Andrea Loaiza

### 3.2.2 DISEÑO DEL SISTEMA

En esta tarea tiene como objetivo prepara la capa de arquitectura para que estén en buena disposición, y de esta manera lograr tener un crecimiento arquitectónico y sistemático de los requerimientos identificados por el usuario en las fases anteriores.

#### 3.2.2.1 Arquitectura del Proyecto



**Figura 19.** Diseño de la Arquitectura de la Aplicación

*Elaborado por:* Andrea Loaiza

#### **3.2.2.2 Primera Iteración**

En la primera interacción se realizó la definición de las personas involucradas, Roles, etc. Es decir el objetivo de esta interacción fue de instituir la arquitectura de la aplicación.

#### **3.2.2.3 Segunda Iteración**

Esta interacción consistió en la captura de datos, por lo que se logró capturar, transformar y cargar datos reales de la compañía CONIEL Cia.Ltda, en la base de datos relacional de la aplicación.

#### **3.2.2.4 Tercera Iteración**

En esta interacción se gestionó las clases, interfaces y entidades, además de los Web Services que se utilizarán y consumirán en la aplicación móvil. Se implementó el servidor para que permita el ingreso y salida de petición a la base de datos, además de su conectividad desde la web, se hicieron pruebas de consumo de los web Services.

#### **3.2.2.5 Cuarta Iteración**

Se diseñaron interfaces de usuarios, y se terminaron de desarrollar las funcionalidades de clases y web Services faltantes.

#### **3.2.2.6 Quinta Iteración**

Se concluyó con el desarrollo e implementación de la aplicación, validando y verificando posibles errores.

#### **3.2.2.7 Sexta Iteración**

Se realizaron pruebas de funcionamiento, para ultimar detalles para la entrega final de la aplicación móvil.

#### **3.2.2.8 Modelo Entidad Relación**

### 3.2.3 IMPLEMENTACIÓN Y PRUEBAS

#### 3.2.3.1 Implementación del Sistema

- **Estándar de Codificación**

La metodología Mobile-D con la que ha sido desarrollado este proyecto define el uso de estándares de codificación, lo que permite disminuir las posibilidades de cometer errores de programación ayudando a optimizar la forma en la que se da mantenimiento a la aplicación por lo que se cuenta con código ordenado.

- a) **Variables**

Para las variables con nombres compuestos la primera palabra va toda en minúscula mientras que la segunda palabra de la que se compone su primera letra va en mayúscula. **Ejemplo:** nombreCliente

- b) **Métodos**

Los nombres de los métodos que son compuestos de varias palabras se pondrá en mayúscula la primera letra de cada palabra con excepción de la primera palabra que compone el nombre en donde todas son minúsculas. **Ejemplo:** eliminarFoto();

- c) **Clases**

Los nombres de las clases se escribirán en mayúscula solo la primera letra de cada palabra. **Ejemplo:** IngresarActividad

- **Seguimiento de iteraciones**

El proyecto está definido en seis iteraciones. Desde la segunda iteración se tiene asignada Story Cards que definen los requisitos de usuarios, cada una de estas Tarjetas de Historias de usuarios se compone de tarjetas de tareas (Task Cards) que son tareas individuales que se deben realizar para completar el requisito definido en los Story Cards.

- a) **Día de Planeación**

En la metodología Mobile-D, la fase de “Producción o Productización” describe que debe haber un día planeación, de programación y uno de liberación para cada iteración. A continuación se detallan actividades de cada etapa de esta fase para el diseño de la aplicación móvil.

- **Análisis de Requerimientos**

Se definen requisitos funcionales y no funcionales, descritos en la matriz de requisitos de este documento para el desarrollo de las Historias de Usuarios y de las tarjetas de Tareas.

- **Planeamiento de Iteración de Historias de Usuarios (Story Cards)**

Después de la descripción de los requisitos funcionales del proyecto se realizó la definición de las Historias de Usuarios que se detallan a continuación:

Nº	HISTORIA DE USUARIO (STORY CARDS)
1	Módulo de Login de Usuario
2	Módulo de Búsqueda de Información
3	Módulo de Fotos
4	Módulo de Geolocalización
5	Módulo de Ingreso de Actividad

**Tabla 14.** Resumen de Historias de Usuarios planificadas

*Elaborado por:* Andrea Loaiza

- **Planeamiento de Iteración de Tarjetas de Tareas (Task Cards)**

Las tarjetas de tareas son un conjunto de actividades que se deben realizar para cada una de las historias de usuarios, por lo que la metodología plantea un día para cada una. A continuación se detallan las tareas que se desarrollaron para cada historia de usuario:

Nº	TARJETAS DE TAREAS (TASK CARDS)
1	Crear e Ingresar información a Base de Datos
2	Crear clases e interfaces para las historias de usuarios en el Servidor
3	Crear e implementar Web Services para las historias de usuarios
4	Crear y diseñar interfaces de usuario para Historia de usuarios
5	Crear clases e interfaces de usuarios para consumo de Web Services por Historia de usuario.

**Tabla 15.** Resumen de Tarjetas de Tareas planificadas

*Elaborado por:* Andrea Loaiza

- **Generación de Pruebas de Aceptación**

Estas pruebas de aceptación permiten valorar la calidad de los resultados obtenidos del desarrollo del software, verificando que se dé el cumplimiento pleno de lo descrito en las historias de usuarios.

**b) Día de Programación**

- **Desarrollo de pruebas**

Las pruebas que se realizan son de forma unitaria, es decir se las hace durante el desarrollo. Como ejemplo se mostrará las pruebas que se realizaron al método Login de usuarios que es controlada por el Web Service SW\_usuarios, que contiene el método de conexión Login (); hará la conexión con el servidor para validar y comprobar que los datos ingresados para el inicio de sesión usuario y contraseña, existan en la base de datos. A continuación se puede observar la clase SW\_usuarios que contiene el método Login (); de lado del servidor:

```
class SW_Usuarios(DefinitionBase):
```

```
    @rpc(_returns=Array(Array(primitive.String)))
```

```
    def getContratos(self, ):

```

```
        m=[]

```

```
        r = contrato.objects.filter(finalVigencia__gte=datetime.date.today())

```

```
        for cont in r:

```

```
            m.append([cont.num, cont.zonas])

```

```
        return m

```

```
    @rpc(primitive.String, primitive.String, primitive.String,
        _returns=Array(primitive.String))

```

```
    def login(self, u, p, c):

```

```
        error = []

```

```
        user = authenticate(username=u, password=p)

```

```
        if user is not None:

```

```
            if user.is_active:

```

```
                if user.sesion_sico:

```

```
                    error = ["El Usuario especificado ya esta en uso."]

```

```
                else:

```

```
                    try:

```

```
                        u = usuarioSico.objects.get(user=user, contrato=c)

```

```
                    except:

```

```
                        u = False

```

```

        if isinstance(u, usuarioSico):
            if integracion(u.nombre, u.clave, user):
                error = [
                    'True',
                    str(user.id),
                    str(user.username),
                    str(user.sesion_sico),
                    ('%s %s' % (user.first_name, user.last_name)).encode('utf8')
                ]
            else:
                error = ['El Sistema Comercial(Sico Cnel) no esta disponible por
                el momento, '
                    'intentelo nuevamente mas tarde...']
                user.sesion_sico=""
                user.save()
            else:
                error = ['El Usuario Especificado no cuenta con permisos
                necesarios para acceder al contarto']
        elif u and p:
            error = ["Su Usuario o Contraseña no son correctos, Intentelo
            nuevamente."]
        return error

    @rpc(primitive.Integer, primitive.String, primitive.String,
    _returns=primitive.Boolean)
    def logout(self, id, u, s):
        user = User.objects.get(id=id, username=u, sesion_sico=s)
        if user:
            cerrarSico(s)
            user.sesion_sico = ""
            user.save()
            return True
        return False


sw_usuarios = DjangoSoapApp([SW_Usuarios], __name__)

```

**Figura 20.** Clase SW\_usuarios del servidor

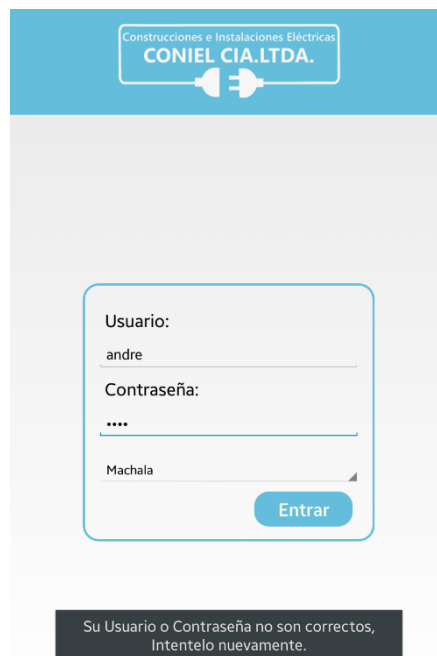
**Fuente:** Andrea Loaiza

Al ejecutar la aplicación se visualizará el Login de usuarios como podemos observar en la siguiente figura:



**Figura 21.** Interfaz de Login de Usuario de la Aplicación móvil

**Fuente:** Andrea Loaiza



**Figura 22.** Caso 1: Prueba Unitaria de Login de Usuario

**Fuente:** Andrea Loaiza



Construcciones e Instalaciones Eléctricas  
**CONIEL CIA.LTDA.**

Usuario:  
andrea

Contraseña:  
...

Machala

Entrar

Su Usuario o Contraseña no son correctos, Intentelo nuevamente.

**Figura 23.** Caso2: Prueba unitaria de Login de Usuario

**Fuente:** Andrea Loaiza

Construcciones e Instalaciones Eléctricas  
**CONIEL CIA.LTDA.**

Usuario:  
andrea

Contraseña:  
\*\*\*\*

Sta. Rosa

Entrar

**Figura 24.** Caso 3: Prueba Unitaria de Login de Usuario

**Fuente:** Andrea Loaiza



**Figura 25.** Mensaje de Bienvenida

**Fuente:** Andrea Loaiza

A continuación detallaremos los resultados obtenidos de la prueba realizada:

<b>Story Card: Login de Usuarios</b>		
<b>Datos</b>		<b>Resultados Medidos</b>
<b>Caso 1</b>		
<b>Usuario Ingresado</b>	andre	<b>Tiempo de ejecución:</b> 3 seg
<b>Contraseña Ingresada</b>	1234	<b>Resultado:</b> Mensaje de error: Su usuario o contraseña no son correctos. Inténtelo de nuevo
<b>Caso 2</b>		
<b>Usuario Ingresado</b>	andrea	<b>Tiempo de ejecución:</b> 3 seg
<b>Contraseña Ingresada</b>	123	<b>Resultado:</b> Mensaje de error: Su usuario o contraseña no son correctos. Inténtelo de nuevo
<b>Caso 3</b>		
<b>Usuario Ingresado</b>	andrea	<b>Tiempo de ejecución:</b> 2 seg
<b>Contraseña Ingresada</b>	1234	<b>Resultado:</b> Bienvenido (a) Andrea Loaiza

**Tabla 16.** Resultados de la prueba unitaria del Login de Usuarios

**Fuente:** Andrea Loaiza

## - Ejecución de iteración

En la ejecución se detalla el desarrollo que corresponde al Story Card de la funcionalidad del Módulo del Login de Usuarios, se toma como ejemplo este proceso tomando en cuenta que los demás son de igual desarrollo. En el apartado anterior se analizó el desarrollo de lado de la vista del servidor en el cual se describían las clases y métodos web que implementa el Servicio Web para la ejecución del Login de usuarios.

En el lado del cliente es decir en la aplicación móvil, la clase java Login.java es la que accederá al servicio web y al método al que ingresará será el doInBackground ().

```
@Override
protected Integer doInBackground(String... params) {
    SW acc = new SW("usuarios.wsdl", "login");
    acc.asignarPropiedades(
        new Tuple[]{
            new Tuple<String, Object>("u", params[0]),
            new Tuple<String, Object>("p", params[1]),
            new Tuple<String, Object>("c", params[2])
        }
    );
    Object r = acc.ajecutar();
    try{
        SoapObject data = (SoapObject)r;
        datos_de_Sesion = new String[data.getPropertyCount()];
        for (int i=0 ; i<data.getPropertyCount() ; i++){
            datos_de_Sesion[i]=""+data.getProperty(i);
        }
    }catch (Exception e){
        toast = "Error, No se ha podido Iniciar Sesión, Verifique su conexión a internet y vuelva a intentarlo";
        this.cancel(true);
    }

    return null;
}
```

**Figura 26.** Implementación del método doInBackground

**Fuente:** Andrea Loaiza

Para obtener mejores resultados del método en la aplicación móvil se creará una clase java que ayudará a entablar comunicación con el servicio web SOAP y también con la base de datos: A continuación se observa la clase SW.java que contiene las cabeceras de comunicación SOAP:

```
public class SW {

    private String namespace = "serviciosWeb.views";
    private String url="http://coniel.servehttp.com:8000/sw/";
    private String soapAction = "";
    SoapObject request = null;

    public SW(String url, String methodName ){
        this.url+=url;
        this.soapAction=this.namespace+"/"+methodName;
        request = new SoapObject(namespace, methodName);
    }
}
```

**Figura 27.** Conexión al Servicio Web desde el cliente

**Fuente:** Andrea Loaiza

Cabe recalcar que para el acceso de los servicios web en Android se utiliza la librería externa ksoap2-android, que es un fork, adaptado a Android. Este framework brinda acceso a los web Services que sigan los estándares de SOAP.

Se puede observar que se crean variables para la clase SW.java como:

**NAMESPACE:** Nombres utilizados el nuestro servicio web

**URL:** Dirección URL para la conexión con el Web Services.

**SOAP\_ACTION:** Es equivalente a los datos anteriores pero haciendo uso de las notaciones de SOAP.

## BIBLIOGRAFÍA

- Acedo, J. (01 de Agosto de 2012). *Programación: Web Service*. Obtenido de <http://programacion.jias.es/2012/01/web-service-definicion-utilizacion-estructura-del-wsdl/>
- Álvarez, J. (10 de Abril de 2008). <http://es.slideshare.net/>. Obtenido de <http://es.slideshare.net/Jmaquino/dispositivos-moviles>
- Álvarez, M. (19 de Noviembre de 2003). *Desarrollo Web*. Obtenido de <http://www.desarrolloweb.com/articulos/1325.php>
- Botero, F. (06 de Mayo de 2012). *Biblioteca Luis Ángel Arango Banco de la República*. Obtenido de <http://www.banrepcultural.org/blaavirtual/ayudadetareas/objetos/objetos39.htm>
- Cancela, L., & Ostos, S. (2012). <https://sites.google.com/>. Obtenido de <https://sites.google.com/site/swcuc3m/home/android/>
- Carballo, P. Y. (21 de Febrero de 2010). *Universidad de los Andes*. Recuperado el 4 de Noviembre de 2013, de Web del profesor: [http://webdelprofesor.ula.ve/ingenieria/hyelitza/materias/programacion2/oxo/ProfaYusneyi\\_Tema8\\_POOClasesyObjetos.pdf](http://webdelprofesor.ula.ve/ingenieria/hyelitza/materias/programacion2/oxo/ProfaYusneyi_Tema8_POOClasesyObjetos.pdf)
- Ceballos, J. (2011). *Programación Java 2*. Mexico: Alfaomega.
- Coniel Cia.Ltda. (12 de Abril de 2013). Datos Informativos de Coniel Cia.Ltda. Machala, El Oro, Ecuador.
- Denzer, P. (23 de Octubre de 2002). *Profesores Uths*. Obtenido de <http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/denzer/informe.pdf>
- Esquivia Rodríguez, A. (13 de Octubre de 2013). *Geek y Theory*. Obtenido de <http://geekytheory.com/json-i-que-es-y-para-que-sirve-json/>
- Fernández, J. (10 de Septiembre de 2006). *Decsai*. Obtenido de [http://leo.ugr.es/J2ME/INTRO/intro\\_4.htm](http://leo.ugr.es/J2ME/INTRO/intro_4.htm)
- Gibert Ginestà, M., & Pérez Mora, O. (2012). <http://ocw.uoc.edu/>. Obtenido de [http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06\\_M2109\\_02152.pdf](http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02152.pdf)
- Gómez Oliver, S. (27 de Febrero de 2012). *sgoliver.net blog*. Obtenido de <http://www.sgoliver.net/blog/?p=2594>
- González Duque, R. (2010). *Python para todos*. España: Autoedición.
- Google Inc. (2009). [developer.android.com/](http://developer.android.com/). Obtenido de <https://developer.android.com/sdk/installing/studio.html>
- Guevara Soriano, A. (06 de Agosto de 2010). *Revista Seguridad*. Obtenido de <http://revista.seguridad.unam.mx/numero-07/dispositivos-m%C3%B3viles>
- Lamarca Lapuente, M. (08 de Diciembre de 2013). *Hipertexto*. Obtenido de <http://www.hipertexto.info/documentos/xml.htm>

- Llobet Azpitarte, R., Alonso Jordá, P., Miedes De Elías, E., Ruiz Fuertes, M., & Torres Goterris, F. (2008). *Introducción a la programación orientada a objetos con java*. Valencia: Universidad Politécnica de Valencia.
- Marco, C. (2010). Sistema Operativo Android Todo lo que querías saber y tenías miedo a preguntar. *Tecnología con estilo Gadgets*, 62.
- Martínez, E. (Mayo de 2001). <http://www.adecom.biz/>. Obtenido de [http://www.adecom.biz/pdf/pdf\\_agosto2005/La%20evolucion%20de%20la%20telefon%C3%ADa%20movil.pdf](http://www.adecom.biz/pdf/pdf_agosto2005/La%20evolucion%20de%20la%20telefon%C3%ADa%20movil.pdf)
- Martínez, J. (21 de Mayo de 2013). *AndroidZone.org*. Obtenido de <http://androidzone.org/2013/05/historia-de-android-la-evolucion-a-lo-largo-de-sus-versiones/>
- Martínez, R. (2 de 10 de 2010). *Postgresql.org.es*. Obtenido de [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql)
- Montoya, J. (2012). *Sistemas Operativos para Mviles*. Obtenido de [https://docs.google.com/presentation/d/177BvKpFn3-B07mICmpHmsPsg9m35Sr\\_K6qBOFwVju4/edit#slide=id.p](https://docs.google.com/presentation/d/177BvKpFn3-B07mICmpHmsPsg9m35Sr_K6qBOFwVju4/edit#slide=id.p)
- Moscatelli, S. (21 de Agosto de 2008). *Fing.edu.uy*. Obtenido de [www.fing.edu.uy/inco/grupos/lins/docsgen/soap/soap.doc](http://www.fing.edu.uy/inco/grupos/lins/docsgen/soap/soap.doc)
- Paredes Velasco, M., Santacruz Valencia, L., & Domínguez Mateos, F. (2012). *Programación Multimedia y Dispositivos Móviles*. Cfgs. Ra-ma.
- Pastorini, A. (11 de Octubre de 2006). *Fing Edu*. Obtenido de <http://www.fing.edu.uy/tecnoinf/mvd/cursos/ria/material/teorico/ria-06-ServiciosWeb.pdf>
- Procoop. (23 de Marzo de 2010). *InfoProcoop*. Recuperado el 30 de Octubre de 2013, de [http://www.infoprocoop.com.ar/index.php?option=com\\_content&view=article&id=86:introduccion-a-la-telefon%C3%ADa-fija&catid=37:telefon%C3%ADa&Itemid=62](http://www.infoprocoop.com.ar/index.php?option=com_content&view=article&id=86:introduccion-a-la-telefon%C3%ADa-fija&catid=37:telefon%C3%ADa&Itemid=62)
- Quintanilla, M. (3 de Julio de 2013). *2Bytes*. Obtenido de <http://blog.dosbytes.com.mx/2013/07/03/que-son-y-para-que-sirven-las-aplicaciones-moviles/>
- Rabajoli, G. (27 de Septiembre de 2007). *Pido Ayuda*. Recuperado el 30 de Octubre de 2013, de <http://pidoayuda.blogspot.com/2007/09/ventajas-y-desventajas-de-los.html>
- Ramírez Hernández, E. (14 de Marzo de 2011). *Universitat Politècnica de València*. Recuperado el 5 de Noviembre de 2013, de Desarrollo de aplicaciones para dispositivos con sistema operativo Android: <http://riunet.upv.es/bitstream/handle/10251/10299/Memoria.pdf>
- Ramos Muñoz, J. (12 de Mayo de 2013). <http://www.w3c.es/>. Obtenido de <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>
- Reynaga Avila, J. (24 de Octubre de 2013). *Slideshare.net*. Obtenido de <http://es.slideshare.net/cardenasnetworks/aplicaciones-moviles-27615261?related=1>
- Ribas Lequerica, J. (2013). *Desarrollo de Aplicaciones para Android*. España: Anaya Multimedia.
- Rivera, A. (2012). *Sistemas Operativos Móviles: Comunicación en tiempo real*. PCWorld.

- Sagástegui Lescano, W. (2008). *Aprenderaprogramar*. Obtenido de [http://aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=102:ique-es-y-para-que-sirve-el-lenguaje-de-etiquetas-xml-extensible-markup-language&catid=46:lenguajes-y-entornos&Itemid=163](http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=102:ique-es-y-para-que-sirve-el-lenguaje-de-etiquetas-xml-extensible-markup-language&catid=46:lenguajes-y-entornos&Itemid=163)
- Sánchez, J. (22 de Julio de 2008). *Autentia Soporte de Desarrollo Informático*. Obtenido de <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=prototypejsAjaxJSON>
- Santa Maria, F. (26 de Febrero de 2014). *Staffcreativa*. Obtenido de <http://blog.staffcreativa.pe/android-ventajas-desventajas/>
- Sébastien, P. (2012). Android Guia de desarrollo de aplicaciones Smartphones y Tablet. En P. Sébastien, *Android Guia de desarrollo de aplicaciones Smartphones y Tablet* (pág. 20). Barcelona: Ediciones ENI.
- Sébastien, P. (2012). Android Guia de desarrollo de aplicaciones para Smartphones y Tablet. En P. Sébastien, *Android Guia de desarrollo de aplicaciones para Smartphones y Tablet* (pág. 11). Barcelona: Ediciones ENI.
- Tajes Martínez, L. (03 de Julio de 2008). *EDV UNIOVI*. Obtenido de WSDL (Web Services Description Language): <http://di002.edv.uniovi.es/~falvarez/WSDL.pdf>
- Terrero, H., & Paredes, J. (2011). *Desarrollo de Aplicaciones Java*. Fundación Código Libre.
- Tomás, G. (15 de Octubre de 2012). *Computer Hoy*. Recuperado el 5 de Noviembre de 2013, de <http://computerhoy.com/listas/moviles/5-mejores-versiones-android-os-1706>
- Tomás, G. J. (2013). El gran libro de Android 3ra Edición. En G. J. Tomás, *El gran libro de Android 3era Edición*. Barcelona: Ediciones Marcombo.
- Vogel, L. (12 de Mayo de 2014). *Vogella*. Obtenido de <http://www.vogella.com/tutorials/Android/article.html>
- VTT Technology for business. (22 de Noviembre de 2007). *Agile.vtt.fi*. Obtenido de <http://agile.vtt.fi/mobile.html>