**FACULTAD DE INGENIERÍA**

**ESCUELA INGENIERÍA DE TELECOMUNICACIONES**

# DESARROLLO DE TECNOLOGÍA NFC PARA MEDIOS DE PAGO A TRAVÉS DE DISPOSITIVOS MÓVILES ALCATEL ONE TOUCH

**TRABAJO ESPECIAL DE GRADO**

Presentado ante la

**UNIVERSIDAD CATÓLICA ANDRÉS BELLO**

Como parte de los requisitos para optar al título de

**INGENIERO DE TELECOMUNICACIONES**


**REALIZADO POR** *Br. Freddy Fernando Vásquez Rodríguez*

**PROFESOR GUÍA** *Diego Pérez*


**FECHA** *Febrero de 2013*


**FACULTAD DE INGENIERÍA**

**ESCUELA INGENIERÍA DE TELECOMUNICACIONES**

# DESARROLLO DE TECNOLOGÍA NFC PARA MEDIOS DE PAGO A TRAVÉS DE DISPOSITIVOS MÓVILES ALCATEL ONE TOUCH

**TRABAJO ESPECIAL DE GRADO**

Presentado ante la

**UNIVERSIDAD CATÓLICA ANDRÉS BELLO**

Como parte de los requisitos para optar al título de

**INGENIERO DE TELECOMUNICACIONES**

**REALIZADO POR** *Br. Freddy Fernando Vásquez Rodríguez*

**PROFESOR GUÍA** *Diego Pérez*

**FECHA** *Febrero de 2013*

**FACULTAD DE INGENIERÍA**

**ESCUELA INGENIERÍA DE TELECOMUNICACIONES**

# DESARROLLO DE TECNOLOGÍA NFC PARA MEDIOS DE PAGO A TRAVÉS DE DISPOSITIVOS MÓVILES ALCATEL ONE TOUCH

**Este jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: _____**

J U R A D O  E X A M I N A D O R

Firma: Firma: Firma:
**Nombre: _____ Nombre: _____ Nombre: _____**

**REALIZADO POR** *Br. Freddy Fernando Vásquez Rodríguez*

**PROFESOR GUÍA** *Diego Pérez*

**FECHA** *Febrero de 2013*

**UNIVERSIDAD CATÓLICA ANDRÉS BELLO**
**FACULTAD DE INGENIERÍA**
**ESCUELA DE INGENIERÍA EN TELECOMUNICACIONES**

# DESARROLLO DE TECNOLOGÍA NFC PARA MEDIOS DE PAGO A TRAVÉS DE DISPOSITIVOS MÓVILES ALCATEL ONE TOUCH.

## RESUMEN.

Desde hace algún tiempo, se ha venido manejando la posibilidad de realizar pagos a través de dispositivos móviles y en ese sentido se han planteado nuevas iniciativas. La mayoría de estas iniciativas se enfocan en utilizar la tecnología NFC (*Near Field Communications*), un protocolo de comunicación estándar que permite comunicaciones de corto alcance entre dispositivos a una distancia de solo centímetros.

Con la intención de facilitar las transacciones de pago, se desarrolló un sistema basado en la aplicación de la tecnología NFC para mejorar los medios de pagos existentes utilizando dispositivos móviles ALCATEL ONE TOUCH. Dicho sistema consiste en acercar un dispositivo móvil que corriendo una aplicación diseñada para ANDROID permita autenticar a un usuario y luego compartir una etiqueta NFC con otro dispositivo para realizar una conexión a una base de datos donde se realizará la consulta y se modificará el saldo de la cuenta del usuario de forma de realizar el cobro y finalizar la transacción.

Palabras claves: NFC, medios de pago, ANDROID.

# DEDICATORIA.

Quiero dedicar este logro primeramente a Dios, por haberme guiado durante todo el camino recorrido hasta llegar a este momento tan especial.

A mis padres Aura Rodríguez y Freddy Vásquez, por su apoyo incondicional y por sus incontables sacrificios para ayudarme durante toda mi vida, en especial durante esta etapa de preparación universitaria. No sé que habría sido de mí sin ustedes, los amo.

A mi hermano Gabriel Vásquez, por siempre estar pendiente de mí y demostrarme su apoyo en los momentos más difíciles.

A mis abuelos José Rodríguez, Aura Gutiérrez, a mis tíos, tías y primos quienes de igual manera me apoyaron en todo momento, han estado siempre pendientes de mi desarrollo y me han acompañado en los buenos y malos momentos, se les quiere mucho.

Como homenaje póstumo para mi abuela Ramona Vásquez y mi tío Alexis Vásquez con quienes me hubiese gustado muchísimo haber compartido este momento tan especial para mí, se que desde donde quiera que están estarán celebrando este momento conmigo y se sentirán orgullosos de mi.

A la persona que ha sido una parte fundamental de mi vida en los últimos años y que ante todo es mi amiga, mi novia María Alejandra Gómez, que me ha acompañado durante diversos momentos de mi vida y me ha apoyado y comprendido como pocas personas lo han podido hacer TE AMO!

A quienes fueron mis compañeros durante estos años de estudios y que se han convertido en más que amigos, a mis futuros colegas Lourdes Terán, Lisbeth Uruburu, Marlyn Zapata, Guillermo Hernández, Gabriel Guzmán, Guillermo Carrillo, animo muchachos que SI SE PUEDE.

Finalmente a las personas que me abrieron las puertas de sus hogares y me hicieron sentir parte de su familia, la Sra. Teresa de Troade, Jesús y María de Lourdes Terán, Hugo y Raiza Uruburu, Gisela González, Manuel Gómez, la

familia Grajal Parejo y a Marta Grajal, gracias por todo el apoyo que me brindaron, nunca los olvidare y siempre estaré profundamente agradecido con uds.

Se les quiere mucho a todos, y de nuevo mil gracias por todo.

*Freddy Vásquez.*

# AGRADECIMIENTOS.

Un agradecimiento especial para todas aquellas personas que brindaron su apoyo para poder hacer posible la realización de este trabajo especial de grado.

Al Lic. Diego Pérez, por guiarme durante todo este largo camino del desarrollo del proyecto y porqué más que un tutor se convirtió en un amigo.

A Iván Garcerant, por brindarme su apoyo y sus conocimientos para alcanzar los resultados del proyecto.

A mi amigo y hermano, Ing. Guillermo Carrillo, quien con su buena voluntad sirvió de guía durante la última parte del desarrollo del proyecto y me brindó sus conocimientos para lograrlo.

A mi hermano y compañero de estudios Gabriel Guzmán quien a pesar de no tener nada que ver con este proyecto se involucró de gran manera y me brindó su apoyo y conocimientos en los momentos más necesitados.

A mis padres y familiares por estar siempre pendientes del desarrollo de este proyecto.

Y finalmente a Dios por guiar el desarrollo de este proyecto por el buen camino.

Gracias a todos!

# INDICE GENERAL.

# INDICE DE FIGURAS

# INDICE DE TABLAS

# INTRODUCCIÓN.

En los últimos años, la tecnología ha ido evolucionando vertiginosamente, provocando que la sociedad cada día la acepte y se adapte a ella, incluso hasta depender de la misma. Esto ha ayudado a que se desarrollen nuevas formas de hacer las actividades tradicionales del hombre en cualquier área y los medios de pago no son la excepción.

En el tema de medios de pago, muchos de los avances tecnológicos apuntan hacia el uso de la tecnología NFC. Debido a que esta tecnología se encuentra en los dispositivos móviles, los usuarios la han ido aceptando amigablemente y en muchos países del mundo ya se están implementando esta forma de realizar pagos.

El crecimiento de estos medios de pago continuará y tendrán un impacto significativo en los pagos futuros, debido a la simplicidad del uso de estos nuevos medios de pago y lo cómodo que resulta hacer una transferencia de dinero con tan solo acercar el dispositivo móvil a un terminal.

El objetivo principal del presente trabajo especial de grado es desarrollar un sistema de medios de pago inalámbrico aplicando la tecnología NFC utilizando los dispositivos móviles de ALCATEL ONE TOUCH con Sistema Operativo ANDROID para su implementación.

Para lograr cumplir estos objetivos, en primer lugar se realizará una investigación del tema planteado, con lo que se construirá el marco teórico, el cuál funcionará como base del proyecto. Posteriormente se procederá a diseñar un sistema que permita realizar este tipo de transacciones, a partir del cual se desarrollaran todos los objetivos específicos del trabajo especial de grado. Se seleccionarán los dispositivos que formaran parte del sistema de forma que funcione y se logre completar la transacción.

Luego de tener el sistema con todos sus elementos, se harán las configuraciones a cada uno de los dispositivos lo que involucrará el desarrollo de aplicaciones en ANDROID, posteriormente, se realizarán pruebas de este sistema

con el fin de, en caso de ser necesario, reconfigurar los dispositivos y arreglar cualquier falla que se presente con el fin de que su implementación sea efectiva.

En este informe se presentan los resultados que se han obtenido luego de realizar el desarrollo de este proyecto con la finalidad de darle cumplimiento a los objetivos planteados.

# I.-PLANTEAMIENTO DEL PROYECTO.

En el presente capítulo, se mostrará cuales fueron las bases iniciales que llevaron a plantear el presente trabajo especial de grado incluyendo los objetivos de la investigación.

## I.1.- Planteamiento del Problema

Desde hace algún tiempo, se ha venido manejando la posibilidad de realizar pagos a través de dispositivos móviles. En poco tiempo se han visto varias iniciativas para incentivar este medio de pago. La mayoría de estas nuevas iniciativas se enfocan en utilizar la tecnología NFC (*Near Field Communications*), un protocolo de comunicación estándar que permite comunicaciones de corto alcance entre dispositivos a una distancia de solo centímetros.

Los usuarios de dispositivos móviles inteligentes, están acostumbrados a tener acceso a muchas cosas durante todo el tiempo y sobre la marcha, cambiando así la forma en que la gente ve los servicios, y la compra y los medios de pago no son la excepción.

La aparición de estos medios de pago, obliga claramente a replantearse muchas de las cuestiones del comercio tradicional, surgiendo nuevos problemas, e incluso agudizando algunos de los ya existentes. En ese catálogo de problemas, se plantean cuestiones que van, desde la validez legal de las transacciones y contratos sin papel, la necesidad de acuerdos internacionales que armonicen las legislaciones sobre comercio, incluido el cobro de impuestos; la protección de los derechos de propiedad intelectual, la protección de los consumidores en cuanto a publicidad engañosa o no deseada, fraude, contenidos ilegales y uso abusivo de datos personales, comparar ofertas y evaluar la fiabilidad del vendedor y del comprador en una relación electrónica, la falta de seguridad de las transacciones y medios de pago electrónicos, la falta de estándares consolidados, la proliferación de aplicaciones y protocolos de comercio electrónico incompatibles y la congestión de Internet.

Sin embargo, y a pesar de todos estos inconvenientes mencionados, los medios de pago a través de dispositivos móviles siguen creciendo debido a la simplicidad de estos servicios, lo fácil que es utilizarlos y lo cómodo que resulta hacer una transferencia de dinero o pago, con tan solo acercar el dispositivo móvil a un terminal, es inevitable que estos servicios tengan un significativo impacto en el mercado de transferencia de dinero.

En vista de todo lo antes mencionado, y con la intención de facilitar las transacciones de pago, se planteó desarrollar un sistema que se base en la aplicación de la tecnología NFC para mejorar los medios de pagos existentes utilizando dispositivos móviles ALCATEL ONE TOUCH.

Tentativamente, se planteó que sea un sistema prepago donde un terminal reconozca a un dispositivo móvil, el cual estará corriendo una aplicación en ANDROID que ayudará y guiará al usuario a realizar la transferencia entre su dispositivo y el terminal para así cumplir con el pago requerido. El sistema incluirá una base de datos donde, al establecer conexión, se buscara al usuario que desea realizar el pago para conocer si tiene el monto requerido y luego proceder a debitar el mismo de su cuenta.

## I.2.- Objetivos

### I.2.1.- Objetivo General

Desarrollar un sistema de medios de pago inalámbrico aplicando la tecnología NFC utilizando los dispositivos móviles de ALCATEL ONE TOUCH con Sistema Operativo ANDROID para su implementación.

### I.2.2.- Objetivos Específicos

- Realizar un estudio sobre las características que debe cumplir un medio de pago para su correcto funcionamiento.
- Diseñar un sistema utilizando NFC para implementarlo en medios de pago.

- Realizar una aplicación en ANDROID que permita el intercambio de datos a través de NFC.
- Crear un prototipo de la experiencia del sistema para comprobar su funcionamiento.

## I.3.- Justificación

La justificación de este proyecto se encuentra en el hecho de desarrollar un sistema de medio de pago que permita la implementación de los dispositivos móviles, los cuáles ahora juegan un papel importante en los usuarios ya que les brinda la facilidad de realizar una gran variedad de tareas al momento que el usuario lo desee. De esta manera, se busca que, mediante la utilización de dispositivos móviles, se implementen los avances tecnológicos que se han obtenido en el área de los medios de pago.

En base a esto, se planteó desarrollar un sistema de medios de pago que aplicando la tecnología NFC, permitan que un usuario que este registrado en el sistema pueda realizar pagos a través de su dispositivo móvil con sistema operativo ANDROID.

## I.4.- Alcances y Limitaciones

### I.4.1.- Alcances

El desarrollo de este trabajo especial de grado incluyó el diseño de un sistema que permita el pago de servicios utilizando dispositivos móviles de ALCATEL ONE TOUCH.

El diseño del sistema incluyó el desarrollo de una aplicación en ANDROID para realizar el pago desde el dispositivo, igualmente fue necesaria la realización de una base de datos que contenga los datos de los clientes que quieran utilizar este servicio, dicha base de datos se alojó en un servidor WEB.

La realización de este trabajo incluyó la implementación de un prototipo del sistema diseñado para comprobar su correcto funcionamiento.

### I.4.2.- Limitaciones

Aunque se realizó un prototipo de la experiencia para comprobar el correcto funcionamiento del sistema, no se incluyó la implementación en transacciones reales de compra.

Al ser un sistema con uso de la tecnología NFC, su uso está limitado a dispositivos móviles ANDROID que posean soporte de dicha tecnología. Adicionalmente, para realizar una transacción es necesaria la conexión del dispositivo móvil que estará implementado este medio de pago con un servidor WEB, por lo que la conexión se ve limitada por la cobertura de la red de las diferentes operadoras en el lugar donde se pretenda realizar la conexión.

# II.- Marco Teórico.

En el presente capítulo, se mostrarán los conceptos y conocimientos necesarios para la realización del trabajo especial de grado.

## II.1.- Near Field Communications (NFC)

NFC es una tecnología de radiofrecuencia que permite la comunicación inalámbrica entre dispositivos móviles siempre que estén muy cerca. NFC es una extensión de la norma ISO 14443, relacionada con tarjetas sin contacto, y se comunica mediante el uso de un campo magnético entre los dispositivos que se desean conectar. Trabaja en la banda de frecuencia HF, específicamente en 13,56 MHz, la cual es una banda libre para la que no se necesita ningún permiso especial y transfiere datos a velocidades de hasta 424 Kbps.

Por ser una tecnología de proximidad, es decir, funciona a distancias cortas, es menos vulnerable al momento de interferir la información ya que es necesario que el dispositivo que pretende interceptar la comunicación este al menos a unos 20 cm de los dispositivos que están en la negociación por lo que es fácil detectarlo.

La comunicación a través de dos dispositivos por medio de NFC, se rige principalmente mediante las normas ISO/IEC 18092 o *Near Field Communication Interface and Protocol 1* (NFCIP-1) y la norma ISO/IEC 21481 o *Near Field Communication Interface and Protocol 2* (NFCIP2), las cuales establecen, entre otros parámetros, las velocidades de transmisión y los tipos de funcionamiento.

Adicionalmente existe un consorcio denominado NFC Forum, que se formó para promover el uso de la tecnología NFC en el desarrollo de especificaciones, garantizando la interoperabilidad entre dispositivos y servicios, y educar al mercado sobre la tecnología NFC.

Según la norma ISO/IEC 18092 (2004) existen dos tipos de funcionamiento de NFC:

– Activo: cuando ambos dispositivos generan un campo electromagnético propio.

  Las etiquetas participantes en este modo pueden establecer una comunicación dinámica bidireccional con el lector NFC. Las etiquetas activas pueden funcionar tanto como transmisor o como receptor de información. Las etiquetas activas son, esencialmente, ordenadores inalámbricos en un chip. Su sofisticación exige que deban tener una fuente de energía externa para alimentar el chip. Debido a esto y la necesidad de alimentación externa, las etiquetas activas se diseñan típicamente para funcionar como un componente integrado de un sistema mayor.

– Pasivo: En este modo de operación, es solamente uno de los dispositivos el que genera el campo electromagnético al que el otro se conectara y se establecerá la comunicación, este dispositivo que genera el campo generalmente es el lector.

  Las etiquetas que intervienen en este modo de operación no requieren de una alimentación propia ya que crean su energía mediante la onda electromagnética recibida desde el lector a través de la antena. (ISO/IEC 18092, 2004)

En cuanto a las velocidades de transmisión, se establecen tres diferentes velocidades, 106 Kbps, 212 Kbps y 424 Kbps. En todas ellas se utiliza la modulación ASK con distintos índices de modulación según sea la tasa de transmisión. (ISO/IEC 18092, 2004)

Para poder establecer una comunicación entre dos dispositivos a través de NFC, es necesario contar con dos elementos importantes los cuáles son una etiqueta y un lector. (Nokia, 2011)

- Etiqueta: La etiqueta es un dispositivo simple, delgado que contiene una antena y una pequeña cantidad de memoria. Se trata de un dispositivo accionado por un campo magnético el cuál dependiendo del tipo de la aplicación tendrá memoria que puede ser de sólo lectura, regrabable, o puede escribir una vez.

- Lector: El lector es un dispositivo activo que genera señales de radio para comunicarse con las etiquetas. Si en la aplicación se hace uso de etiquetas pasivas, entonces es el lector quien a través del campo magnético emitido le da energía a la etiqueta para comenzar la comunicación.

El envío de información se realiza mediante el formato de tramas, a dicha trama se le agrega una cabecera con un byte de inicio, un byte con la longitud de la información y dos bytes con un comando según lo que se quiera hacer. La trama finaliza con el resultado del cálculo del Código de Redundancia Cíclica. (ISO/IEC 18092, 2004)

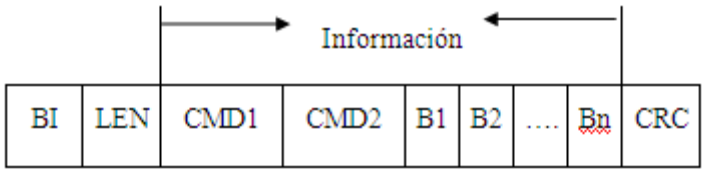


Figura 1 Trama de NFC

Fuente: Elaboración Propia

Como lo indica Nokia (2011), la tecnología NFC tiene tres diferentes maneras de operar que están definidos por las normas ISO 18092 y 14443:

- Lectura – Escritura: en este modo, el dispositivo con NFC está en la capacidad de leer y escribir información de cualquier otro terminal con el que se conecte.
- Par a Par: en este modo, dos dispositivos con NFC pueden intercambiar información como se haría con Bluetooth o Wi-Fi.
- Emulación de tarjetas: en este modo, un dispositivo con NFC se comporta como una tarjeta sin contacto que intercambia la información con un terminal.

Al igual que todas las tecnologías, NFC necesita de protocolos que se encarguen de estandarizar el funcionamiento de la misma. Los encargados de crear estos estándares para NFC son el NFC Forum y la ISO.

NFC Forum establece los siguientes tres estándares:

- **NFC Data Exchange (NDEF):**

NDEF es un formato binario compacto y ligero que se puede llevar a direcciones de páginas web, tarjetas virtuales y tipos de datos específicos de NFC. NDEF permite utilizar fácilmente cualquier tipo de etiqueta compatible para transferir datos, debido a que oculta todas las especificaciones del tipo de etiqueta con los detalles de la aplicación. NDEF intercambia mensajes que consisten en una secuencia de registros donde cada registro lleva una carga útil y una cabecera. El contenido de la carga útil puede ser de tipo URL o un tipo de datos específico de la aplicación NFC. En la cabecera se especifican el tipo de datos y el tamaño del registro. La longitud de carga útil indica el número de octetos de la misma. (Nokia, 2011)

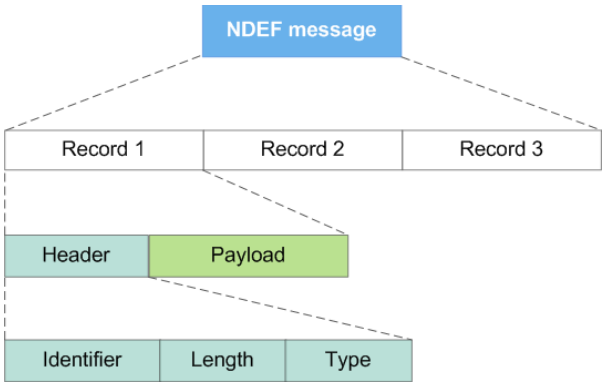


Figura 2 Estructura de mensaje NDEF

Fuente: (Nokia, 2011)

- **Record Type Definition (RTD):**

Son registros que se le agregan a la trama NDEF para especificar el contenido de la carga útil. En el caso de tipos especiales de datos, el NFC Forum especifica los siguientes:

  - NFC Texto RTD
  - NFC URL RTD
  - NFC Posters inteligentes RTD
  - NFC de control genérico RDT

- NFC Firma RTD

El más simple es un tipo de registro de texto, que puede llevar a una cadena única de código. Un registro de texto se puede incluir en un mensaje de NDEF como un texto descriptivo para otro registro. El tipo de registro URL se puede utilizar para almacenar una dirección web, un correo electrónico o un número de teléfono en un formato binario optimizado. El RTD para Posters inteligentes define cómo poner las URL, SMS o números de teléfono en una etiqueta NFC Forum y la forma de transporte entre los dispositivos. (Nokia, 2011)

- **Logical Link Control Protocol (LLCP):**

Es un protocolo de nivel de enlace creado para mejorar la operación en el modo par a par, introduce una conexión bidireccional, permitiendo que ambos dispositivos puedan enviar y recibir datos mediante las siguientes opciones de intercambio de datos: (Nokia, 2011)

- Orientado a la conexión de transferencia: donde los intercambios de datos se reconocen.
- Transferencia sin conexión: donde los intercambios de datos no son reconocidas.

Por su parte la ISO ofrece los siguientes estándares:

- **ISO 14443:**

Es un estándar desarrollado para la comunicación de tarjetas sin contacto en un radio de 13,56 MHz.

Hay dos versiones de la norma ISO 14443-2, capa de radio con modulación diferente y los métodos de codificación de bits. Estas versiones son conocidas como las versiones A y B de la norma ISO 14443. Igualmente, ISO 14443 especifica dos versiones de la elaboración de paquetes y de bajo nivel parte de protocolo (ISO 14443-3). La capa superior de la pila de protocolo ISO define una interfaz de comando (ISO 14443-4) para transferir información. (Nokia, 2011)


Figura 3 Protocolo ISO 14443.

Fuente: (Nokia, 2011)

- **NFC Interface and Protocol Specification (NFCIP-1):**

Es una norma definida por la ISO en la especificación 18092 para establecer cómo será la comunicación entre dos dispositivos utilizando NFC.

La pila de protocolos en NFCIP-1 se basa en la norma ISO 14443. La diferencia principal es un nuevo comando, que sustituye a la parte superior de la pila.

NFCIP-1 incluye dos modos de comunicación que permiten a un dispositivo NFC para comunicarse con otros dispositivos NFC de forma par a par, así como los métodos de modulación aplicados en la señal, las características de seguridad, las velocidades de transmisión y la estructura de la trama de información.


Figura 4 Protocolo NFCIP-1.

Fuente: (Nokia, 2011)

Actualmente, la tecnología NFC se ha venido utilizando para algunas actividades específicas, la más común es utilizarlo como una cartera electrónica. Sin embargo existen otras aplicaciones típicas como lo son:

- Pago: un dispositivo con NFC puede utilizar una aplicación para realizar pagos como lo haría una tarjeta de crédito común. Para pagar sólo basta con acercar el dispositivo a un punto de venta y el dispositivo estaría operando como un emulador de tarjetas sin contacto donde tendría la información necesaria para la realización del pago.
- Compartir archivos: dos dispositivos con NFC pueden establecer una conexión para realizar un intercambio de archivos. Cuando el usuario seleccione el archivo o información a transferir, el dispositivo envía una señal de radio al otro y allí es donde se inicia la comunicación par a par.
- Identificación: el dispositivo móvil podría tener información que permita el acceso a lugares donde se necesite una identificación la cuál será intercambiada a un terminal NFC que reconocerá dicha información y permitirá o no el acceso.

## II.2.- Seguridad:

Una de las principales dificultades que presenta el sector de las telecomunicaciones es el tema de seguridad. Los cambios en las tecnologías de

información y en las comunicaciones han traído una serie de vulnerabilidades y amenazas que deben ser prevenidas con medidas de seguridad adecuadas que garanticen la protección de la información que se está manejando.

Según lo define la Unión Internacional de las Telecomunicaciones el término de seguridad abarca cuatro grandes rasgos, los cuales son:

- **Autenticación:** garantizar la identidad de los que se comunican en la red.
- **Integridad:** garantizar que la información que se transmite no ha sido alterada y que llega íntegra al receptor tal y como ha sido enviada.
- **Confidencialidad:** asegurar que la comunicación sea confidencial, es decir, que nadie distinto del emisor y del receptor tienen acceso a la información.
- **No repudio:** que las partes que intervienen en una transacción o en una comunicación no puedan negar haberlo hecho.

Las soluciones a los problemas que abracan el término seguridad no tienen un lugar específico en el modelo de capas de protocolos, en cambio, cada una de ellas tiene algo que brindar para dar protección al sistema. En la capa física se pueden proteger las líneas de transmisión; en la capa de enlace, los paquetes pueden encriptarse cuando se envíen y desencriptarse al ser recibidos; en la capa de red se pueden utilizar firewalls; en la capa de transporte se pueden encriptar conexiones de punto a punto y finalmente en la capa de aplicación se puede manejar los temas de autenticación y no repudio. (Tanenbaum, 2003)

A continuación se definen algunas de las técnicas que se implementan en los sistemas actuales para proveer de seguridad a las comunicaciones.

### II.2.1.- Criptografía:

La criptografía es un conjunto de técnicas empleadas para conservar segura la información.

En la criptografía existen dos procesos básicos que son:

- **Encriptación:** Proceso mediante el cual el mensaje llano se transforma en un mensaje cifrado mediante una función compleja y una llave de codificación especial.
- **Desencriptación:** Proceso inverso, en el cual el texto cifrado se convierte nuevamente en el texto llano original mediante una función compleja y una llave de desencriptación.

Tanenbaum (2003) define que para realizar estas tareas de encriptación y desencriptación se cuentan con varios algoritmos y funciones como lo son las siguientes:

- **Algoritmos de llave simétrica:**

En este tipo de algoritmos se utiliza la misma llave para encriptar y desencriptar el mensaje. El problema con este tipo de sistemas es el número de llaves que es necesario administrar.

Entre los algoritmos de llaves simétricas más comunes tenemos:

- DES (Estándar de Encriptación de Datos)
- TripleDES
- AES

- **Algoritmos de llave pública:**

En este tipo de algoritmos se utiliza una llave para encriptar el mensaje y otra para desencriptarlo.

La llave de encriptación se conoce como llave pública, mientras que la llave de desencriptación se conoce como llave privada o secreta. Entre los más comunes tenemos:

- RSA. (Rivest Shamir Adleman)
- EL GAMAL.
- DSS. (Digital Signature Estándar)

- **Funciones de compendio de mensaje:**

Una función de compendio de mensajes (message digest) genera un patrón de bits único para una entrada específica.

Las funciones de compendio de mensajes transforman la información contenida en un archivo pequeño o grande, a un sólo número, típicamente de entre 128 y 256 bits.

Entre las funciones más importantes tenemos:

- HMAC (Hashed Message Autenthication Code).
- MD2 (Message Digest #2).
- MD4
- MD5
- SHA (Secure Hash Algorithm).
- SHA1

### II.2.2.- Red Privada Virtual (VPN):

Las redes VPN son aquellas que permiten simular una red privada sobre una red pública, o que permite conectar un equipo a una red LAN a través de la nube, es decir, las redes VPN brindan la posibilidad de que usuarios móviles se conecten a una red privada tal como si estuvieran dentro de la LAN del lugar donde se implementa la VPN. (Joskowicz, 2008)

Esto sería muy favorable en caso de hacer una implementación del sistema donde se requiera que el dispositivo de cobro este en movimiento, como podría ser un ejemplo del pago de un pasaje de autobús.

El establecimiento de este tipo de red es a través de, entre otros por protocolos, IPsec y L2TP.

- **IPsec (Internet Protocol Security)**, es un grupo de protocolos de la capa 3 y 4 del modelo OSI, que se encargan de brindar seguridad a las

comunicaciones autenticando y cifrando los paquetes IP en un flujo de datos.

Este grupo de protocolos es implementado principalmente para asegurar el flujo de paquetes, garantizar la autenticación mutua y establecer parámetros criptográficos.

IPsec brinda esta confidencialidad, integridad, autenticidad y protección a repeticiones utilizando otros dos protocolos que son Authentication Protocol (AH) y Encapsulated Security Payload (ESP). (Dattatreya y Kumar, 2002)

El protocolo AH, es implementado para autenticación, integridad y protección a repeticiones, mientras que el protocolo ESP es implementado tanto para esas funciones de AH como para la confidencialidad de la data enviada.

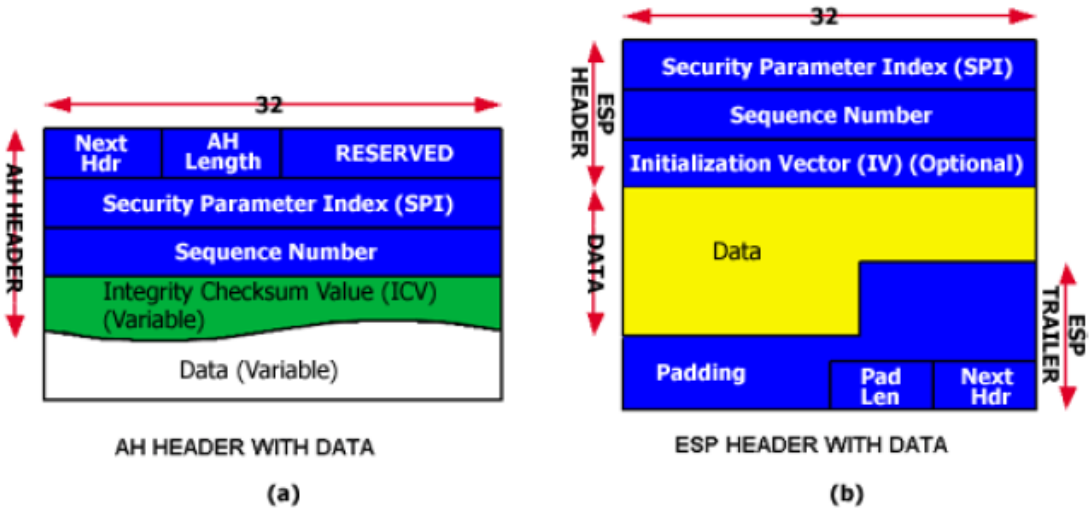


Figura 5 Paquetes AH y ESP.

Fuente: (Dattatreya y Kumar, 2002).

– **L2TP (Layer 2 Tunneling Protocol)**, es un protocolo de túnel utilizado para conexiones de redes VPN. Como lo indica CISCO (s.f), L2TP surge de la unión del protocolo L2F (Layer 2 Forwarding) de CISCO y el protocolo PPTP (Point-to-Point Tunneleing Protocol) de MICROSOFT.

L2TP, no proporciona ningún tipo de cifrado o confidencialidad por sí mismo, sino que se basa en un protocolo de encriptación que pasa dentro del túnel para proporcionar privacidad.

Es por estas debilidades mencionadas que IPsec se ha convertido en el protocolo más utilizado en conexiones de túnel VPN para proveer los paquetes de confidencialidad, integridad y autenticación.

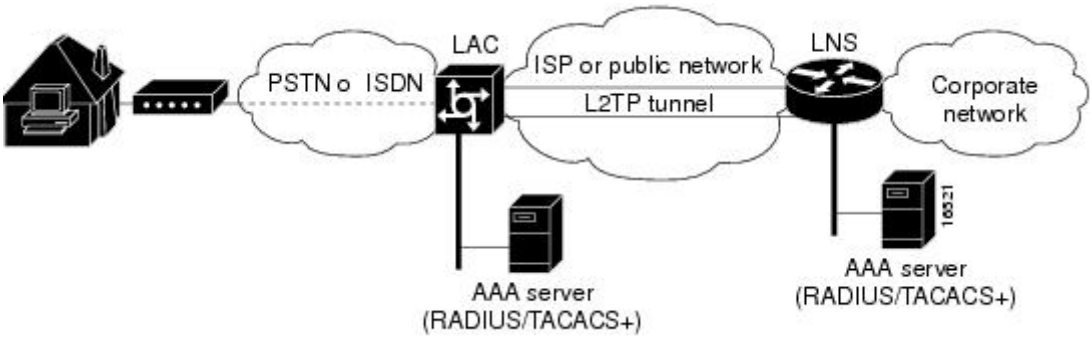


Figura 6 Arquitectura de L2TP.

Fuente: CISCO.

### II.2.3.- Firewall:

Los *firewall* son servidores de seguridad que se encargan de mantener fuera de una red a todo aquel que no esté identificado y cuyo origen sea desconocido. Estos servidores de seguridad están compuestos por enrutadores que se encargan de hacer el filtrado de paquetes según los parámetros establecidos y de una puerta de enlace a la aplicación. (Tanenbaum, 2003)

En general, son sistemas encargados de controlar el tráfico entre una red segura, red LAN, y una red no segura como la red pública. (Joskowicz, 2008)

Para implementar las políticas de seguridad, los servidores hacen uso de las siguientes funciones:

- Bloqueo de tráfico no deseado: filtrado de paquetes, bloqueo de servicios y bloqueo en el acceso a algunas direcciones Web.
- Monitorizar y detectar actividades sospechosas.

- Esconder la red interna mediante traduciendo direcciones públicas a privadas (NAT).

### II.2.4.- Nombre de Punto de Acceso (APN):

Adicionalmente y para brindar mayor soporte de seguridad a la red en un caso de uso como el planteado anteriormente de realizar el pago de un pasaje en una unidad de transporte público, se podría hacer una implementación de un APN privado ya que se requeriría de un dispositivo que se encuentre instalado en la unidad con conexión móvil, a través de GPRS (*Servicios de Radiotransmisión de Paquetes de Datos Generales*) que brinde conexión en cualquier lugar por el que se desplace la unidad según la cobertura de la operadora.

Un APN, es el nombre con el cual se identifica a una red externa a la que se desea conectar mediante algún dispositivo. Un APN, identifica los paquetes de datos de red con el que un usuario móvil desea comunicarse. El APN es utilizado en redes de acceso como GPRS. (España, 2003)

Un APN se utiliza para acceder a un servicio asociado a un GGSN (Gateway GPRS Support Nodes- Nodo de Soporte de la Compuerta GPRS). El nombre se traduce por SGSN (Serving GPRS Support Nodes – Nodo de Soporte del Servicio GPRS) usando los DNS para obtener la dirección IP que identifica al GGSN que puede proporcionar el servicio solicitado. Un APN consiste en un identificador de red y un identificador operacional de APN. El identificador de red APN es un nombre de dominio completo, este identificador se utiliza para identificar la red externa a la que el GGSN está conectado. (Kasera y Narang, 2004)

Un APN generalmente está compuesto por dos partes, el identificador de red y el identificador del operador. El identificador de red indica la red externa a la que el GGSN se encuentra conectado. El identificador del operador define los paquetes de la red del operador en que se encuentra el GGSN.

Sobre este APN privado, se haría la implementación de los protocolos de túneles VPN brindando así total seguridad a la red evitando entonces que intrusos

logren evadir las medidas de seguridad impidiendo así que puedan manipular la red y los datos que por ella se transmitan.

## II.3.- Medios de Pago

Cuando queremos realizar alguna transacción financiera o queremos adquirir algún bien o servicio hacemos uso de los medios de pago. Los medios de pago son la herramienta que se utiliza para transferir valor monetario en una transacción económica a fin de pagar por algún bien o servicio

Durante siglos la humanidad consiguió alimentos, herramientas y otros bienes a través del trueque directo de una cosa por otra. Muchas veces la misma transacción podía darse por montos y productos muy dispares entre lo que estaba determinado por las necesidades de las personas que realizaban dicho trueque. Sin embargo, con el pasar del tiempo y con la aparición de nuevas formas de mercado, se fue creando un sistema más formal para regular los intercambios. Se fijó un valor de cambio para cada cosa y se determinó un precio fijo equivalente a estos bienes más valorados por lo que nació lo que hoy existe como el dinero.

Para cualquier economía es fundamental contar con un sistema de pagos eficiente y seguro. En la búsqueda de esa eficiencia la gran mayoría de las economías del mundo han orientado sus esfuerzos hacia la rápida adopción de sistemas de pago electrónicos.

En ese mismo sentido muchas empresas e individuos han adoptado medios de pago electrónicos por su flexibilidad, confiabilidad y conveniencia, lo cual se ha visto reforzado con el desarrollo de Internet, dándole una nueva dimensión al uso de algunos medios de pago tradicionales y ha abierto nuevos mecanismos de pago electrónico.

## II.4.- Dispositivo Móvil

Se dice que un dispositivo móvil es un dispositivo electrónico de comunicación, normalmente de diseño reducido y sugerente y basado en la

tecnología de ondas de radio (transmite por radiofrecuencia), que tiene la misma funcionalidad que cualquier teléfono de línea fija y que están programados para realizar algunas tareas, muchas de las cuales requieren de procesamiento. (Basterretche, 2007)

Además de ser capaz de realizar llamadas como cualquier otro teléfono convencional, un dispositivo móvil incorpora un conjunto de funciones adicionales, tales como mensajería instantánea, agenda, juegos, almacenamiento de archivos, entre otros, que aumentan la potencialidad de utilización de estos dispositivos. Es más, su desarrollo y exigencia ha llegado a tal punto, que ya se puede hablar incluso de términos tales como memoria RAM y ligarlos al uso de móviles, dentro de los que se pueden manejar información de todo tipo (audio, video, texto, etc.), lo que hace de ellos un complemento perfecto tanto para el hombre de a pie como para el de negocios.

Ese desarrollo ha llevado que los dispositivos móviles ya no sean utilizados sólo para realizar llamadas sino que también se utilizan para manejar cuentas de correo electrónico, reproducción de MP3, entre otras muchas cosas.

Con todos estos avances antes mencionados surge un nuevo concepto, el de teléfonos inteligentes es un dispositivo electrónico que funciona como un teléfono móvil con características similares a las de un ordenador personal. Es un elemento que permite hacer llamadas y enviar mensajes de texto como un móvil convencional pero además incluye características cercanas a las de un ordenador personal esto gracias al desarrollo de los distintos sistemas operativos existentes en el mercado, que le van agregando funciones a los dispositivos, como la tecnología NFC que ya muchos de los últimos equipos que están saliendo al mercado las traen como parte de ellos. (Baz, Ferreira, Rodríguez, García, s.f)

De todos estos dispositivos móviles presentes en el mercado, el presente trabajo de grado se enfocará en los dispositivos de ALCATEL ONE TOUCH que trabajan con el sistema operativo ANDROID.

## II.5.- Sistema Operativo ANDROID

ANDROID es un sistema operativo basado en el núcleo LINUX diseñado originalmente para dispositivos móviles, pero que posteriormente expandió su desarrollo para soportar otros dispositivos tales como tabletas, reproductores MP3, netbook, PC, televisores, lectores electrónicos e incluso se han llegado a ver en electrodomésticos y televisores.

La estructura del sistema operativo ANDROID se compone de aplicaciones que se ejecutan en un entorno de trabajo de JAVA de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de JAVA en una máquina virtual Dalvik con compilación en tiempo de ejecución. El sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2,8 millones de líneas de lenguaje C, 2,1 millones de líneas de Java y 1,75 millones de líneas de C++.

Como todo sistema operativo, los dispositivos deben cumplir con algunos requerimientos básicos y características, algunos de ellos son:

- Pantallas con resoluciones que permitan gráficos optimizados impulsado por una biblioteca de gráficos 2D y gráficos 3D.
- Base de datos SQLite para almacenamiento de datos estructurados.
- Soporte multimedia para medios de audio, video y formatos de imagen (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- Conectividad GSM (*Global System for Mobile communication*) de telefonía (dependiente del *hardware*).
- Soporte de Bluetooth, EDGE, 3G y WiFi (dependiente del *hardware*) Cámara, GPS, brújula y acelerómetro (dependiente del *hardware*).

Desde su lanzamiento inicial, el sistema operativo ANDROID ha sufrido varias actualizaciones, básicamente arreglan errores de *software* de la versión anterior y agregan nuevas funciones.

Las versiones de ANDROID reciben nombre de postres en inglés. Las versiones que oficialmente se conocen y se comercializan son:

- Versión 1.5 Cupcake.

- Versión 1.6 Donut.
- Versión 2.0/2.1 Éclair.
- Versión 2.2 Froyo.
- Versión 2.3 Gingerbread.
- Versión 3.0/3.1/3.2 Honeycomb.
- Versión 4.0 Ice Cream Sandwich.
- Versión 4.1/4.2 Jelly Bean.

## II.6.- JAVA

Como se mencionó anteriormente, las aplicaciones de ANDROID se desarrollan en el marco de trabajo de JAVA por lo que se deben desarrollar en dicho entorno.

JAVA es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++. (Schildt, 2007)

Según Alvarellos D y Pastor G (2012), lo que llevo a Sun Microsystems a diseñar este lenguaje de programación fueron tres aspectos fundamentales:

- La creciente necesidad de interfaces más cómodas e intuitivas que los sistemas de ventanas que existían hasta el momento.
- Fiabilidad del código y facilidad de desarrollo, ya que al utilizar algunas de las características que ofrecían C o C++ se aumentaban el costo de pruebas y depuración.
- Enorme diversidad de controladores electrónicos. Este nuevo lenguaje permitiría escribir un código común para todos los dispositivos electrónicos que se controlan mediante la implementación de microprocesadores que utilizan diversos conjuntos de instrucciones.

A partir del momento de su llegada, JAVA comenzó a formar parte importante en todo lo relacionado a programación ya que se encuentra tanto en la WEB como en casi cualquier dispositivo electrónico existente en el mercado. Alvarellos D y Pastor G (2012) dicen que la principal razón de esto son las características del lenguaje como:

- Java es un lenguaje orientado a objetos por lo que facilita abordar la solución de cualquier tipo de problema.
- Es un lenguaje sencillo pero muy potente.
- La ejecución del código Java es segura y fiable ya que los programas no acceden directamente a la memoria del computador, sino que la ejecución se realiza en un entorno virtual conocido como JVM (JAVA Virtual Machine).
- Es multiplataforma, por lo que el entorno necesario para su ejecución es de pequeño tamaño y puede adaptarse incluso al interior de un navegador.

## II.7.- ECLIPSE

Para desarrollar cualquier lenguaje de programación, es necesario contar con un Software que permita hacer el código. Este tipo de software es conocido como IDE (Interfaz de Desarrollo Integrado) que no es más que una herramienta que permite escribir, guardar y compilar un programa.

En el caso de JAVA existe un IDE llamado ECLIPSE. Como lo dice Gutiérrez J. (2004), Eclipse es una plataforma de IBM de desarrollo de código abierto basada en Java. En si mismo Eclipse es un marco y un conjunto de servicios para construir un entorno de desarrollo a partir de componentes conectados (plug-in). En el caso de desarrollos en JAVA es necesario instalar el JDK (JAVA Development Kit).

El SDK de Eclipse incluye las herramientas de desarrollo de Java, ofreciendo un entorno de desarrollo con un compilador de Java interno y un modelo completo de los archivos fuente de Java.

ECLIPSE está constituido por varias plataformas de trabajo:

1. Plataforma Principal: es donde se inicia ECLIPSE, allí se ejecutan los plug-in.
2. Plataforma para Integrar Distribuciones: en esta parte se encuentran los API's (Aplication Programming Interface) básicos para el desarrollo de cada aplicación según lo que ésta vaya a ofrecer.
3. SWT (Standard Widget Toolkit): son los componentes que constituyen las interfaces gráficas de JAVA.
4. JFace: en esta parte se manejan los archivos, textos y editores de texto.
5. Workbench: se encarga de las vistas, editores, perspectivas y asistentes.

## II.8.- ANDROID VIRTUAL DEVICE (AVD)

Debido a que la aplicación que se desarrollara es para ser utilizada en un dispositivo móvil con ANDROID O.S, se necesita un emulador donde poder probar el correcto funcionamiento de dicha aplicación. Es por ello que utilizamos un AVD.

AVD, no es más que un software que imita el funcionamiento de un dispositivo móvil con ANDROID de acuerdo a como sea configurado.

Un AVD está compuesto por varias partes. La primera es el perfil de hardware, en el se definen las características de hardware del dispositivo que se pretende emular. Se puede definir, entre otras cosas, si el dispositivo tiene cámara, la cantidad de memoria y otras características. Luego tenemos una asignación del símbolo de sistema, en esta parte se escoge la versión que tendrá nuestro dispositivo virtual, esta configuración la logramos a través del ANDROID SDK. Luego tenemos un área de opciones varias donde podemos programar las dimensiones de la pantalla, la apariencia, si el dispositivo virtual tendrá un emulador de tarjeta SD. Finalmente tenemos el área de almacenamiento del equipo en desarrollo, en esta se tendrán los datos del dispositivo virtual, es decir, las aplicaciones instaladas, las configuraciones básicas y la emulación de una tarjeta SD, si fuera el caso.

# III.- Metodología.

En el presente capítulo, se indicarán cuáles fueron las distintas actividades realizadas durante el transcurso de este Trabajo Especial de Grado. Las etapas que conformaron este trabajo se muestran a continuación:

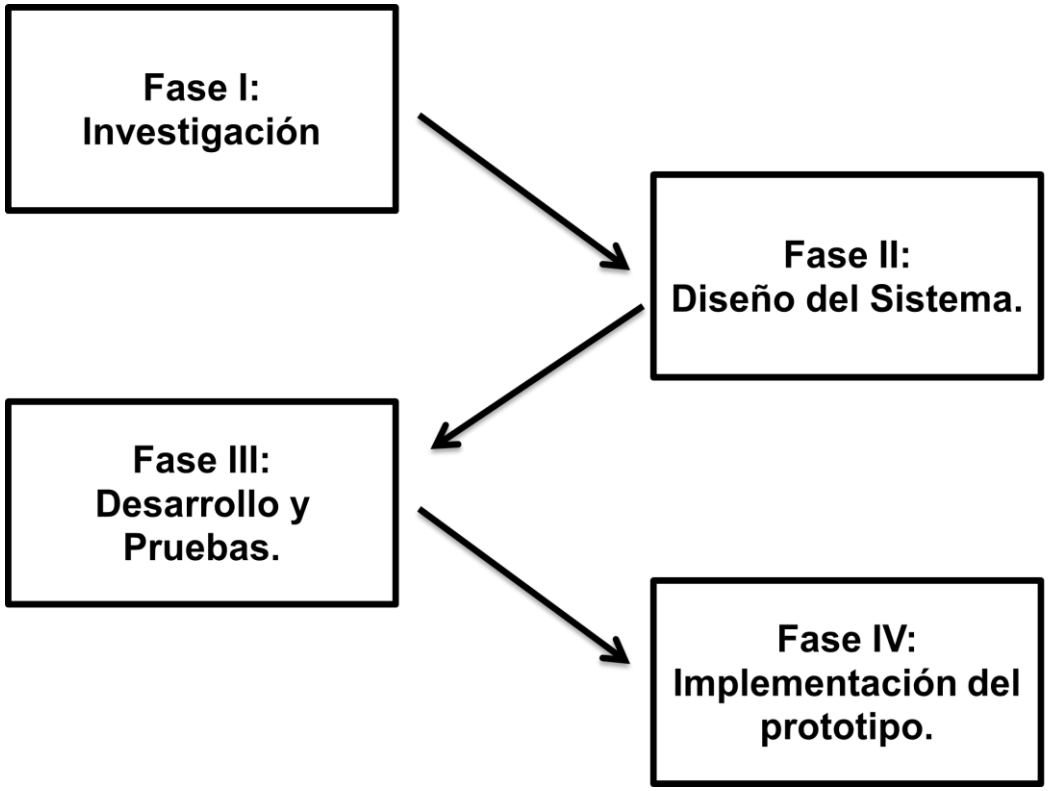


Figura 7 Etapas de la Investigación

Fuente: Elaboración Propia.

A continuación, se muestra una tabla donde se establecen cuáles serán las actividades que se realizarán en cada una de las fases:

| FASE | ACTIVIDAD |
|---|---|
| I: Investigación. | a.- Investigación de las características de un medio de pago.<br>b.- Investigación a cerca de la tecnología NFC.<br>c.- Investigación sobre el sistema operativo ANDROID.<br>d.- Investigación sobre seguridad en redes y telecomunicaciones. |
| II: Diseño del Sistema. | a.- Diseño un sistema que permita el pago con NFC.<br>b.- Evaluación de cuáles son las características de los dispositivos.<br>c.- Selección los dispositivos según las características obtenidas. |
| III: Desarrollo y Pruebas | a.- Desarrollo la aplicación en ANDROID que permita intercambio de datos con la tecnología NFC.<br>b.- Desarrollo de una base de datos.<br>c.- Programación del punto de venta.<br>d.- Realización de pruebas para verificar el funcionamiento de cada parte del sistema. |
| IV: Implementación. | a.- Interconexión de todos los elementos del sistema para demostrar el correcto funcionamiento del mismo. |

Tabla 1 Etapas de Elaboración del Trabajo Especial de Grado.

Fuente: Elaboración Propia.

## III.1.- Investigación

Realización de la investigación y documentación de aquellos términos teóricos necesarios para desarrollar el trabajo especial de grado, cubriendo principalmente los conceptos que estén abarcados en la tecnología NFC, el sistema operativo ANDROID y seguridad. Adicionalmente se estudiaron cuáles serían las características que debe cumplir un medio de pago para que funcione de la mejor manera posible.

## III.2.- Diseño del Sistema

Esta fase del proyecto se dividirá en tres etapas diferentes, Diseño del Sistema, Investigación de características de dispositivos y Selección de dispositivos. Es en esta fase donde comienza el grueso del trabajo especial de grado, ya que se realizará el diseño del sistema que permitirá solucionar el problema planteado de desarrollar la tecnología NFC como medio de pago.

En la primera parte se realizará el diseño de un sistema que permita realizar un pago a través de la tecnología NFC presente en los dispositivos móviles con sistema operativo ANDROID.

Adicionalmente, se establecerán cuáles serán las especificaciones básicas con las que deben cumplir los equipos que conformarán el sistema.

Finalmente, con base en esas características obtenidas, se buscarán algunos dispositivos que cumplan con ellas para la posterior selección de los que formaran parte del sistema y harán posible su funcionamiento.

## III.3.- Desarrollo y Pruebas

En esta fase del proyecto se realizará el desarrollo de los diferentes elementos que formarán el sistema. Esto incluye:

1. Realizar una aplicación en ANDROID que permita transacciones con NFC.
2. Realizar una base de datos para tener el control de los usuarios registrados.
3. Programar el punto de venta para que acepte la transacción.
4. Realización de pruebas en cada una de las actividades.

## III.4.- Implementación del Sistema

Una vez cumplidas las actividades de la fase anterior, se procederá a realizar una implementación de un prototipo del sistema diseñado, de forma tal que se logre demostrar el correcto funcionamiento de todos los elementos que la integran.

# IV.- Desarrollo.

En el presente capítulo se describen como transcurrieron las diferentes fases y etapas en las que estuvo dividió este trabajo especial de grado. Primero se describe el proceso de investigación, posteriormente se indican detalladamente los aspectos que se tomaron en cuenta para el diseño del sistema y finalmente se expondrá como se desarrollo el período de desarrollo y pruebas.

## IV.1.- Investigación

En esta etapa se realizaron las siguientes actividades:

1. Investigación de la tecnología NFC y sus protocolos.
2. Estudio de medios de pago y sus características.
3. Investigación sobre el sistema operativo ANDROID y el desarrollo de aplicaciones en el mismo.
4. Investigación sobre seguridad en sistemas de tecnología.

La investigación sobre NFC, se inició estudiando como estaba conformado un sistema que aplicara esta tecnología, investigando de esa manera sobre los protocolos que la rigen y la estandarización existente en ese tema.

En cuanto al estudio sobre los medios de pago, se basó en investigar cuáles son los componentes de un sistema de medios de pago viendo los que ya existen en el mercado cotidiano, a partir de estos se obtuvieron las características básicas que debe cumplir cualquier sistema de pago que pudiera ser implementado.

Igualmente se realizó una investigación de lo que abarca el sistema operativo ANDROID, durante esta investigación se obtuvieron datos acerca de cómo desarrollar aplicaciones para dicho sistema operativo, esto implica cuál es el lenguaje de programación (JAVA) y cuál es el compilador que se utilizará (ECLIPSE).

Finalmente se investigó sobre el tema de seguridad y las posibles técnicas y protocolos utilizables para brindar al sistema mayor respaldo en términos de autenticación, integridad, confidencialidad y no repudio.

### IV.1.1.- Estudio de Medios de Pago:

Al momento de realizar la investigación a cerca de los medios de pago para determinar las características a través de un estudio de los más comunes en el mercado venezolano.

Además de esta observación, se estableció comunicación con un miembro de una empresa especialista en brindar soluciones de medios de pago, Iván Garcerant, quien a través de comunicaciones personales y reuniones brindó su ayuda al momento de analizar las formas de pago estudiadas.

Para este estudio se observaron los medios más utilizados en el mercado venezolano, como lo son el pago en efectivo, pago con tarjeta de débito o crédito y pago con cheques, con el fin de determinar cuál es su funcionamiento, las características y los elementos que los conforman.

- **Pago en efectivo:**

En este tipo de medio de pago, el cliente realiza el pago en el momento de recibir el producto y en efectivo tal como se indica.

Este pago normalmente es realizado con la moneda de curso legal del país en el que se realice, en nuestro caso la moneda de uso oficial establecida por las leyes es el Bolívar Fuerte, es por esto que ningún establecimiento comercial puede oponerse al pago en efectivo, y de la misma manera pueda exigir que el pago sea realizado solamente de esta manera.

La ventaja principal de utilizar este tipo de pago es que siempre es aceptado y en algunos casos es hasta obligatorio realizar el pago a través de este medio. Adicionalmente es más rápido que cancelar con una tarjeta, de débito o crédito, o con un cheque ya que no es necesario presentar ninguna identificación.

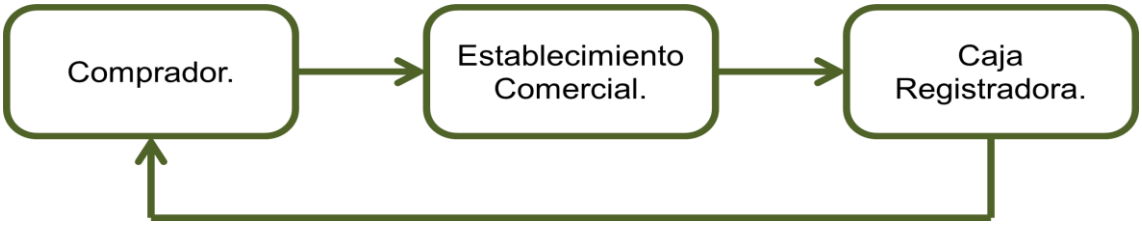


Figura 8 Estructura de Pago en Efectivo.

Fuente: Elaboración Propia.

En la figura anterior se muestra como se conforma la estructura del medio de pago al momento de cancelar el monto con efectivo, en ella se observa claramente la presencia de tres factores fundamentales, el comprador, el establecimiento comercial o vendedor y la caja registradora.

El comprador será la persona que realiza el pago, en este caso en efectivo, quien, tal como lo exige la legislación venezolana a través del SENIAT (Servicio Nacional Integrado de Administración Aduanera y Tributaria), debe proporcionarle sus datos de identificación al vendedor del establecimiento comercial para poder registrar la venta realizada. El establecimiento comercial juega el papel de realizar la venta y de recibir el dinero, en cuanto a la caja registradora será el lugar donde se almacene, a través de los datos obtenidos del comprador, la venta realizada y se imprimirá un comprobante o factura para dársela al comprador.

Sin embargo, a pesar de esta exigencia del SENIAT de brindarle factura al comprador, se observa que en muchos casos el establecimiento no registra en su sistema la compra y, por ende, no se emite la factura.

- **Pago con tarjetas:**

Al momento de realizar un pago por medio de tarjetas, estas pueden ser de débito o crédito. La principal diferencia entre ellas radica en la forma de realizar el cobro al cliente por el monto cancelado. En las tarjetas de débito, el monto es deducido automáticamente de la cuenta del cliente luego de finalizada la transacción, por su parte en las tarjetas de crédito almacenan el monto de la

transacción en la cuenta del cliente afiliada a la tarjeta de crédito para que sea cancelada según lo convenido en el contrato con el banco que la proporciona.

En general al momento de realizar un pago por medio de una tarjeta, la estructura de comunicación que se realiza es la siguiente.

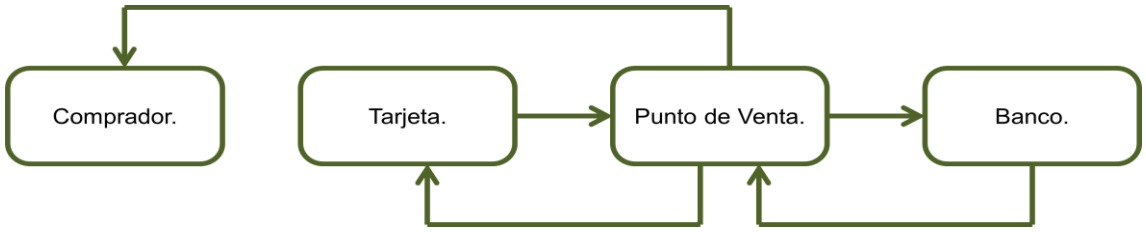


Figura 9 Estructura de Pago con Tarjetas.

Fuente: Elaboración Propia.

El comprador proporciona al vendedor la tarjeta que desee utilizar y en muchos casos exigen un documento de identificación. Actualmente se están utilizando tarjetas con chip integrado que ayudan a evitar fraudes y plagios manteniendo una comunicación constante entre el punto de venta y la tarjeta durante el tiempo que dure la transacción.

El vendedor introduce en el punto de venta que esté en su establecimiento comercial la tarjeta y la cédula de identidad del cliente. Este punto de venta es proporcionado por un banco, donde generalmente el establecimiento debe tener una cuenta. Luego de esto, comienza la comunicación entre el punto de venta y la información codificada que se encuentra dentro del chip de la tarjeta y se debe introducir el monto. Si es una tarjeta de débito se solicita al cliente que introduzca la contraseña, posteriormente se establece la conexión entre el punto de venta y los servidores del banco a través de la línea telefónica.

Finalmente, el banco le envía el resultado de la transacción al punto de venta donde se imprimirá un comprobante de la operación realizada.

- **Pago con cheque:**

Un cheque es un documento de valor, donde una persona, el titular, autoriza a otra a realizar el cobro por un monto específico de la cuenta del banco emisor de la chequera.

La estructura general de este medio de pago es la que se muestra a continuación.

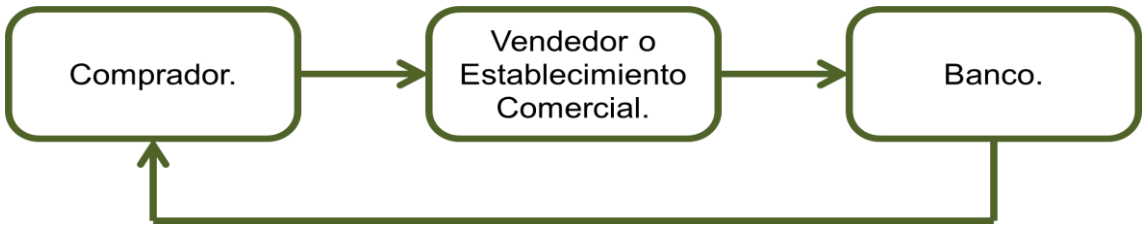


Figura 10 Estructura de Pago con Cheque.

Fuente: Elaboración Propia.

El comprador emite un cheque de su cuenta, autorizando al vendedor o a un establecimiento comercial a realizar el cobro de un monto establecido de la cuenta desde la que se emite el cheque. En este cheque debe ir la firma del cliente la cuál será el comprobante ante el banco de que es un documento legítimo y de que puede ser cobrado solamente por la persona a la que le fue emitido.

Una vez recibido el cheque, el portador (vendedor o establecimiento comercial) se dispondrá a realizar la conformación del cheque para determinar su validez, ya sea realizando una llamada al banco emisor de la chequera o dirigiéndose a las oficinas del banco para cobrarlo o depositarlo en una cuenta. En este último proceso el banco deberá consultar en su base de datos que los datos del emisor coincidan y lo más importantes que las firmas sean idénticas a las de su sistema para autorizar el pago del mismo.

Al autorizar el pago del cheque, el monto será deducido en ese mismo momento de la cuenta de la cual el cliente realizó la emisión y de esta manera quedaría efectuado el pago.

Luego de finalizar este proceso de observación de los medios de pago, se procedió a realizar la segunda etapa del trabajo especial de grado que fue llamada Diseño del Sistema.

## IV.2.- Diseño del Sistema

Se realizó el diseño de un sistema que integrará todas las características de un medio de pago y que adicionalmente implementará la tecnología NFC para establecer una comunicación entre dos dispositivos.

### IV.2.1.- Diseño del Sistema de Medio de Pago

Al momento de realizar el diseño del sistema, se debieron tomar en cuenta todas las características que habían sido obtenidas durante el proceso de investigación, de forma tal que al momento de implementarlo, éste funcionara de manera óptima y acorde a los requerimientos de un sistema de este tipo.

Al finalizar el diseño, se realizó una revisión del sistema diseñado, de forma de evaluar cuáles serían los elementos que cumplirían con las funciones de cada uno de los pasos que formarían dicho sistema.

De esta manera se obtuvo que para que este sistema pudiera funcionar se necesitaban tres elementos fundamentales:

1. Dispositivo Móvil.
2. Punto de Venta.
3. Servidor de Base de Datos.

### IV.2.2.- Investigación de las características de los dispositivos

Durante esta etapa, se realizó una investigación de las características que debían cumplir los equipos o dispositivos que conformarían al sistema que fue diseñado en la etapa anterior.

Luego de haber evaluado cada etapa del sistema, se observó que el requerimiento principal de este sistema, es que los dispositivos posean tecnología NFC, ya que es en esa tecnología en la que se basa el funcionamiento del mismo.

Los dispositivos que deben contar con NFC son el dispositivo móvil ANDROID y el punto de venta.

Adicionalmente, se vio la necesidad de realizar una conexión entre el punto de venta y el servidor donde estuviera alojada la base de datos, por lo que se determinó que la manera de interconectar ambos dispositivos era mediante el uso de la red de internet, ya que el punto de venta debe acceder a la base de datos sin encontrarse conectado localmente a él.

Luego de haber determinado estos dos elementos importantes de comunicación, se estableció que el punto de venta a seleccionar debería entonces cumplir con la condición de tener soporte de NFC y adicionalmente de tener la capacidad de acceder a internet para la conexión antes mencionada.

### IV.2.3.- Selección de los dispositivos

Luego de haber obtenido claramente cuáles serían los requerimientos básicos que debían cumplir los dispositivos, se procedió a realizar una investigación de cuáles eran los posibles equipos en el mercado que podrían cumplir con dichas características.

Al momento de seleccionar el dispositivo móvil, no se presentaron varias opciones ya que el presente trabajo especial de grado fue desarrollado en la empresa ALCATEL ONE TOUCH y por ende el dispositivo móvil debía pertenecer a dicha marca.

En la gama de teléfonos inteligentes con la que cuenta ALCATEL ONE TOUCH sólo se dispone de un dispositivo móvil con soporte de la tecnología NFC, la cuál es fundamental para el cumplimiento de los objetivos planteados. Por lo antes mencionado, el dispositivo que seleccionado fue el ALCATEL ONE TOUCH 996. Las características del dispositivo se muestran en el Apéndice A.

Figura 11 ALCATEL OT 996

Fuente: www.alcatelonetouch.com

Por su parte, al escoger el punto de venta si se presentaron varias opciones. Debido a que la tendencia actual en el mundo, en términos de medios de pago, apunta hacia tecnologías de pago sin contacto, especialmente NFC, muchos fabricantes han ido desarrollando sus terminales pensando en dicha tecnología. Entre esos fabricantes se encuentran Verifone e Ingenico que son los principales fabricantes de puntos de ventas en el mercado actual.

Tal como lo indica Verifon en su sitio web, la empresa cuenta con diferentes series y tipos de punto de venta con soportes NFC. Unos con grandes pantallas a colores como los de la serie MX800 hasta los más sencillos como el PINpad 1000SE que es un punto de venta común con soporte NFC. Entre los dispositivos de Verifon con esta tecnología se encuentran:

- Serie MX800.
- VX Evolution.
- Vx810 y Vx810 Duet.
- QX1000.
- PINpad 1000SE.
- R3210.

Todos estos dispositivos de punto de venta cuentan con soporte para formas de pago sin contacto a través de NFC y brindan conexión hacia la red de manera de poder acceder al servidor.

Por su parte, Ingenico, en su sitio web refleja que como parte de sus avances en esta tecnología de medios de pago sin contacto, también cuenta con algunos terminales con soporte de tecnología NFC. Entre estos dispositivos se encuentra el iCT250.

Al momento de la selección del dispositivo de punto de venta con soporte de NFC, se conversó de nuevo con la empresa especializada en soluciones de medios de pago, quienes brindaron su apoyo y asesoría para dicha selección.

En conversaciones sostenidas con la empresa comunicaron que poseían un modelo de punto de venta con soporte de tecnología NFC, el cual ofrecieron facilitar para la realización de esta investigación. Este punto de venta fue el Ingenico iCT250.

Figura 12 POS Ingenico iCT250

Fuente: http://www.ingenico.com/

Finalmente se seleccionó el equipo que funcionaría para el servidor de Base de Datos, para esto se tomó una laptop ACER Aspire 3100 con la que se contaba y fue adaptada para el uso en este sistema. Se le colocó una tarjeta de memoria RAM de 1 GB y se le instaló el sistema operativo UBUNTU 12.10, el cual ofrece mayor facilidad en el manejo de este tipo de servidores.

## IV.3.- Desarrollo y Pruebas

En esta fase del proyecto, se trabajó en el desarrollo de la aplicación de ANDROID, la base de datos y la aplicación para el dispositivo que funcionaría como punto de venta.

### IV.3.1.- Desarrollo de la Aplicación ANDROID

Esta aplicación fue desarrollada utilizando la herramienta de ECLIPSE, que es una interfaz de desarrollo de JAVA donde se trabaja con clases y rutinas de este lenguaje.

Esta aplicación tendría como función realizar la autenticación del usuario en la base de datos del sistema, de forma tal de garantizar que este fuera el único que pudiera utilizar el monto disponible de su cuenta.

La aplicación se iniciaría con una ventana donde se le solicita al usuario que introduzca su nombre de usuario y su contraseña en los campos determinados para dicho fin, para que posteriormente seleccione la opción de iniciar sesión y realizar la autenticación.

Posteriormente se configuró otra pantalla, a la que se tendría acceso una vez iniciada la sesión. Esta segunda pantalla se estructuró con las siguientes partes, una en la parte superior donde se da la bienvenida al usuario con su nombre, otra donde se da la opción de realizar una consulta de saldo, otra donde se da la opción de realizar el pago y finalmente se muestra la opción de salir del sistema.

Finalmente, se tendrá la última pantalla del sistema a la que se accederá luego de presionar la opción de consultar saldo y en donde se mostrará el resultado de dicha consulta.

Al tener configuradas estas pantallas se procedió a trabajar en la configuración de las acciones que estas debían llevar a cabo según la opción que seleccionara el usuario.

En el paquete ***com.example.nfcpagos***, se encuentran establecidas las clases en las que se dividió dicha aplicación, estas clases fueron ***Pantalla_Login.java***, ***Pantalla_Opciones.java,*** ***Pantalla_Saldo*** y ***HttpPostAux.java***.

En la tabla que se muestra a continuación, se muestra, en resumen, la función de cada clase de esta aplicación.

| CLASE | DESCRIPCIÓN |
|---|---|
| Pantalla_Login.java | En esta clase se configura la pantalla para obtener los datos de autenticación del usuario y establecer la conexión al servidor para realizar la consulta. |
| Pantalla_Opciones.java | En esta clase se configura la pantalla para brindarle al usuario las opciones de Consultar Saldo y de Realizar Pago. |
| Pantalla_Saldo.java | En esta clase se configura la pantalla para obtener realizar una nueva conexión al servidor, en este caso para realizar la consulta del saldo del usuario. |
| HttpPostAux.java | Esta es una clase auxiliar que se utiliza para establecer la conexión http al servidor de la base de datos. |

Tabla 2 Clases de la Aplicación NFCPagos.

Fuente: Elaboración Propia.

La clase ***Pantalla_Login.java***, es donde se configuraron todos los métodos que establecerían las acciones de la pantalla inicial de la aplicación llamada ***pantalla_login_lo***.

La función fundamental de esta clase es obtener los datos de entrada de los campos de textos habilitados para que el usuario ingrese su nombre de usuario y contraseña y que al momento de presionar el botón de entrar, envíe los datos ingresados a la base de datos de forma tal que se logre corroborar o no la existencia de dicho usuario en el sistema.

La conexión con la base de datos se realizó haciendo uso de un método definido en la clase ***Pantalla_Login.java*** llamado ***Login***. Este método sería el encargado de tomar los datos de usuario y contraseña y asignarlos a un arreglo definido del tipo ***HttpPostAux.java*** para que se realice la conexión con la base de datos y realizar la consulta de los datos requeridos.

En este arreglo, que es enviado al servidor de base de datos a través de un post http, está conformado por dos campos, uno que contiene el nombre de usuario y otro que contiene la contraseña cifrada a través de un hash SHA-1 para brindarle beneficios de seguridad a la información que se está transmitiendo durante la conexión.

Para que la conexión con la base de datos se realizara en segundo plano, se hizo uso de la clase ***AsyncTask*** que está definida en el API de ANDROID. Esta clase es la encargada precisamente de eso, de realizar algún tipo de proceso en segundo plano. Para ello cuenta con tres métodos definidos en dicho API que se muestran en la siguiente tabla.

| Método | Función |
|---|---|
| onPreExecute() | Configura una acción que se realizará justo en el momento anterior de iniciar el proceso que se desea realizar en segundo plano. En esta aplicación mostrara por pantalla un mensaje que indique que se está realizando la consulta a la base de datos para realizar la autenticación del usuario. |
| doInBackground() | Configura la acción que se desea se realice en segundo plano. En este caso lo que se desea realizar es la consulta con la base de datos, para realizar la consulta de si existe o no en el registro la persona que está tratando de acceder. |
| onPostExecute() | Configura la acción a realizar luego de que se finalizó el proceso que fue realizado en segundo plano. En este caso se configuró que al finalizar la consulta ocurrieran dos cosas, si existe el usuario se pasa a la segunda pantalla y si no se muestra un mensaje de error. |

Tabla 3 Métodos de la clase AsyncTask

Fuente: Elaboración Propia.

A continuación se presenta una tabla resumen de los métodos que están contenidos en la clase ***Pantalla_Login.java.***

| MÉTODO | DESCRIPCIÓN |
|---|---|
| onCreate(Bundle savedInstanceState) | Este método es creado automáticamente al momento de ser iniciada una actividad con un layout, de forma de que alli se declaren los componentes de esta pantalla. |
| onEntrar (View boton) | En este método se configura la acción que se llevará a cabo una vez que el usuario haya introducido su nombre y contraseña y presione el botón entrar. |
| onSalir (View boton) | Este método configura la acción que se realizará al momento de presionar el botón salir de la pantalla inicial de la aplicación. |
| loginstatus(String username, String password) | En este método se realizó la configuración del proceso de conexión a la base de datos para realizar la autenticación del usuario. |
| Clase Login extends AsyncTask | Esta clase es la encargada de realizar la conexión y autenticación en segundo plano. |
| error1() | Este método ese utilizó para configurar un mensaje de error en caso de que se presione el boton entrar sin haber introducido el nombre de usuario y contraseña. |
| error2() | Este método muestra un mensaje de error en caso de que ocurriese alguno durante la conexión a la base de datos para la autenticación del usuario. |
| hasha | Este método se encarga de realizar la función de compendio de mensaje obteniendo como parámetro de entrada la contraseña a la que le crea un hash con SHA-1 |

Tabla 4 Métodos de la clase Pantalla_Login.java

Fuente: Elaboración Propia.

Por su parte la clase ***Pantalla_Opciones.java***, es la encargada de la configuración de la segunda pantalla de la aplicación denominada ***pantalla_opciones_lo***. En dicha pantalla se le dan al usuario las opciones de realizar una consulta de su saldo disponible en la cuenta y de realizar un pago.

Para la opción de consultar el saldo, se programó un método cuya función sería llamar a la clase ***Pantalla_Saldo.java*** del paquete, dicha clase será explicada un poco más adelante.

Si por el contrario, el usuario selecciona la opción de realizar un pago, la acción llevada a cabo será la de compartir una etiqueta NFC que esté en el dispositivo móvil que este corriendo la aplicación. Para configurar esta acción se realizó un método llamado ***onPagar*** que hará uso de las herramientas del sistema del dispositivo móvil para acceder al manejador de etiquetas que tenga instalada en el equipo y poder compartir la etiqueta con el sistema de pagos.

Para que esto pudiera realizarse, fue necesario modificar algunos permisos contenidos en el ***Android Manifest*** que es un archivo contenido en la raíz del proyecto donde estará configurada toda la información necesaria sobre la aplicación. En este archivo se definen el nombre del paquete JAVA, todas las clases contenidas en el proyecto, los permisos que necesita la aplicación, entre otras informaciones.

En la siguiente tabla se presentan los métodos implementados en esta clase

| MÉTODO | DESCRIPCIÓN |
|---|---|
| onCreate(Bundle savedInstanceState) | Este método es creado automáticamente al momento de ser iniciada una actividad con un layout, de forma de que alli se declaren los componentes de esta pantalla. |
| onSalir2 (View boton) | Este método configura la acción que se realizará al momento de presionar el botón salir de la pantalla opciones de la aplicación. |
| onConsulta() | En este método configura la acción del botón Consultar Saldo, es decir, pasa a la siguiente clase que mostrara su propia pantalla y realizará la consulta. |
| onPagar() | Este método se encarga de utilizar las herramientas del sistema para acceder al manejador de servicios de NFC y poder compartir la etiqueta. |

Tabla 5 Métodos de la clase Pantalla_Opciones.java

Fuente: Elaboración Propia.

Como se menciono anteriormente, la clase ***Pantalla_Saldo.java*** es el resultado de la selección de la opción consultar saldo disponible. En esta clase se hará uso de nuevo de la clase AsynTask del API de ANDROID para realizar en

segundo plano una consulta a la base de datos al igual que se hizo al momento de autenticar al usuario. Posteriormente a la consulta, se mostrará en la pantalla del dispositivo el monto obtenido de la consulta.

En la tabla que se muestra a continuación, se presentan los métodos que conforman esta clase.

| MÉTODO | DESCRIPCIÓN |
|---|---|
| onCreate(Bundle savedInstanceState) | Este método es creado automáticamente al momento de ser iniciada una actividad con un layout, de forma de que alli se declaren los componentes de esta pantalla. |
| onRegresar (View boton) | Este método configura la acción que se realizará al momento de presionar el botón regresar de la pantalla saldo de la aplicación. Se realiza el cambio a la pantalla anterior y se finaliza ésta. |
| Clase Consulta extends AsyncTask | Esta clase es la encargada de conectarse en un segundo plano con la base de datos para realizar la consulta del saldo del usuario. |
| error3() | Este método configura un mensaje de error en caso de ocurrir alguno durante el proceso de consulta del saldo. |

Tabla 6 Métodos de la clase Pantalla_Saldo.java

Fuente: Elaboración Propia.

Finalmente, la clase ***HttpPostAux.java***, es una clase auxiliar utilizada para establecer una conexión http con el servidor WEB donde está alojada la base de datos. Dicha conexión se establece mediante el envío de los parámetros que se necesiten para la consulta que se desea realizar a través de un post con una conexión http. La respuesta de esta consulta será recibida en un arreglo de tipo JSON, el cuál en esta misma clase será descifrado a un arreglo de tipo String para poder utilizar los datos obtenidos.

### IV.3.2.- Desarrollo de la Base de Datos

Al momento de comenzar a diseñar la base de datos se debieron tomar en cuenta algunos aspectos que eran necesarios para lograr la realización de este punto. Primero había que seleccionar a través de cuál herramienta se realizaría el

manejo de la base de datos, posteriormente hubo que realizar el diseño del modelo de entidad relación y finalmente se debió realizar una serie de funciones que permitieran la conexión desde un agente externo hacia el servidor y realizar las consultas.

La herramienta seleccionada para gestionar la base de datos fue MySQL, que es un software libre que permite almacenar datos de forma fácil con alto nivel de seguridad para evitar el plagio de datos que estén contenidos en ella. Adicionalmente, al ser un software de código abierto, permite que cualquier persona que desee utilizarlo pueda modificar el código fuente de manera que logre cumplir con sus necesidades.

Luego de haber seleccionado cuál sería la herramienta para manejar la base de datos, se procedió a diseñar el modelo de entidad relación donde se establecen las diferentes tablas o registros que conformaran nuestra base de datos y los atributos que la componen.

Según los requerimientos de este sistema de medios de pago diseñado, el modelo de entidad relación quedo definido como:

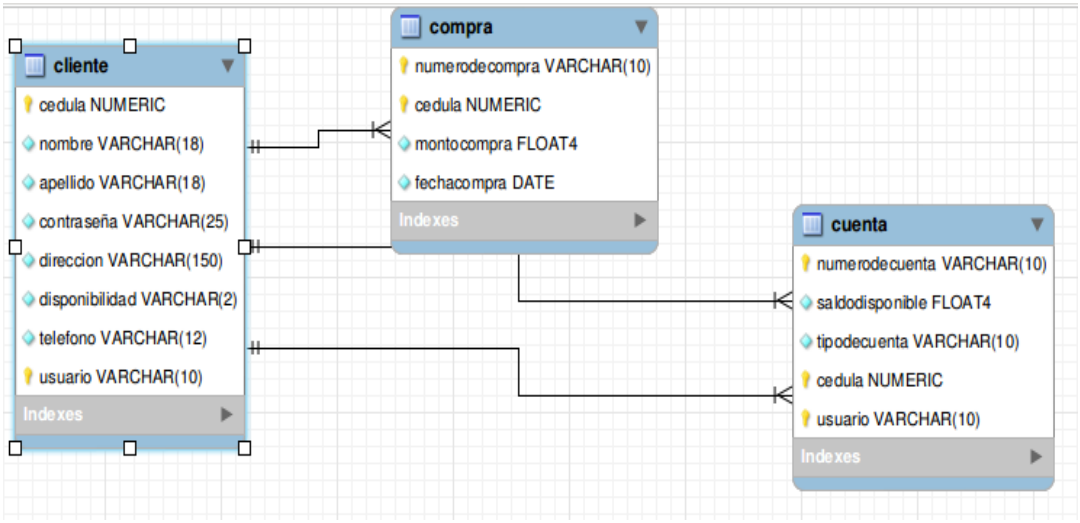


Figura 13 Modelo de Entidad Relación

Fuente: Elaboración Propia.

Como se puede observar, el modelo de entidad relación está conformado por tres tablas, las cuales se resumen a continuación.

| TABLA | COLUMNA | DESCRIPCIÓN |
|---|---|---|
| Cliente | Cédula | Es la identificación única para cada cliente. |
| | Nombre | Nombre del cliente. |
| | Apellido | Apellido del cliente. |
| | Usuario | Nombre de usuario del cliente, al igual que la cédula será una clave primaria, de forma que no podrá repetirse en varios usuarios. |
| | Contraseña | Contraseña del cliente, encriptada con un hash SHA-1, para ingresar al sistema. |
| | Dirección | Dirección del cliente. |
| | Teléfono | Teléfono del cliente |
| | Disponibilidad | Disponibilidad de saldo del cliente. |
| Cuenta | Número de Cuenta | Es una clave primaria que indica el número de la cuenta del cliente. |
| | Saldo Disponible | Indica cuanto es el monto disponible en la cuenta del cliente. |
| | Tipo de Cuenta | Indica el tipo de cuenta del cliente. |
| | Cédula | Es la identificación única para cada cliente. |
| | Usuario | Nombre de usuario del cliente. |
| Compra | Número de Compra | Es la identificación única para cada compra o pago que realice un usuario. |
| | Cédula | Es la identificación única para cada cliente. |
| | Monto de Compra | Monto del pago o compra realizado. |
| | Fecha de Compra | Fecha en la que se realizó el pago o compra. |

Tabla 7 Componentes de las tablas de la base de datos

Fuente: Elaboración Propia.

Finalmente, se procedió a realizar las funciones que permitieran la conexión a la base de datos mediante una serie de procedimientos en php que se alojarían en el servidor WEB. Estas funciones debían permitir tres actividades específicas que son autenticar el usuario, realizar la consulta de saldo y realizar el cobro de la transacción.

Para la autenticación del usuario se hace uso de la función ***Login***, en esta función se configura que los datos recibidos desde la aplicación a través de una conexión http (usuario y contraseña) sean enviados a la base de datos para realizar

una consulta en la tabla ***Cliente*** y comparar si existe algún cliente con los mismos datos enviados, en caso de que se encuentre dicho cliente, se devolverá a la aplicación un arreglo de datos de tipo JSON que serán procesados por la aplicación para que se realicen las acciones ya establecidas en la misma.

Para la consulta de saldo se realizó la función ***Consulta***, en esta función se configura la acción que se realizará cuando el usuario presione el botón de Consultar Saldo en la aplicación. Esta vez se realizará una consulta a la base de datos enviando el nombre del usuario y se realizará la búsqueda en la tabla ***Cuenta*** de la columna ***saldodisponible*** del usuario que esté realizando dicha consulta.

Finalmente, para realizar el cobro, se realizó la función ***Pago***, esta función será la encargada de registrar en la tabla ***Compra*** la nueva transacción que se está realizando y para que en la tabla ***Cliente*** realice la modificación de la columna ***saldodisponible*** a la que se le restará el monto por el que se hizo la transacción.

Como se ha mencionado anteriormente, la base de datos está alojada en un servidor WEB donde estarán igualmente todos estas funciones php que fueron diseñadas. Para el momento en que se deseen realizar dichas conexiones deberán acceder a través de la dirección IP del Servidor.

Para lograr conectarse con el servidor WEB era necesario obtener una dirección DNS, debido a que para contar con esto es necesario comprar la dirección del dominio, se hizo uso de una herramienta libre obtenida desde el sitio web http://no-ip.com. Esta herramienta permite que un equipo donde se instale la misma tenga un nombre para su dirección IP que se actualizará en un período de tiempo configurable.

### IV.3.3.- Programación del Punto de Ventas

Al momento de realizar la programación del punto de ventas ocurrió un gran inconveniente, la empresa con la que se había establecido contacto en la primera parte del proyecto y que había ofrecido el equipo iCT250 para que formara parte del sistema de medio de pago diseñado, por razones ajenas a nuestra

voluntad no pudo realizar facilitar el equipo de manera que no tendríamos este dispositivo fundamental para el sistema.

A pesar de los esfuerzos realizados por el tutor, Lic. Diego Pérez, y por el realizador de este trabajo especial de grado, fue imposible ubicar de nuevo a una empresa u organización que se interesara en prestar su dispositivo. En vista de esto, se decidió que una solución para cubrir este dispositivo, sería implementar el sistema con dos dispositivos móviles, es decir, uno que funcione como el dispositivo móvil del usuario, y otro que haga las funciones de punto de venta leyendo entonces el mensaje que envíe el usuario al hacer contacto a través de NFC.

Para esto fue necesario entonces realizar otra aplicación en ANDROID, donde se diseñaran una serie de clases y métodos que harían las veces de las funciones que desempeñaría el punto de venta. Esta funciones serían la de realizar la lectura de la etiqueta compartida por el dispositivo móvil del usuario, tomar la información obtenida en ella y enviarla a través de una conexión http al servidor WEB donde ya están alojadas las funciones php a que se debe conectar cada actividad para realizar la consulta adecuada y obtener los resultados esperados.

Para realizar todo esto antes mencionado, se hizo uso de nuevo de la herramienta ECLIPSE para realizar la aplicación. En el paquete ***com.example.cobronfc***, se establecieron las clases ***Pantalla_Cobro.java, Pantalla_Final.java, NFCIntent.java*** y ***HttpPostAux.java***.

A continuación se muestra una tabla resumen de la función de cada una de las clases de la aplicación CobroNFC.

| CLASE | DESCRIPCIÓN |
|---|---|
| Pantalla_Cobro.java | En esta clase se configura la pantalla para obtener los datos de la etiqueta compartida por el dispositivo y se pide que se indique el monto a cobrar. |
| Pantalla_Final.java | En esta clase se realiza la conexión al servidor para realizar el cobro del monto establecido. Al finalizar la conexión se mostrará por pantalla el resultado de la consulta. |
| NFCIntent.java | Es una clase auxiliar que se utiliza para procesar la data que está en la etiqueta compartida por el dispositivo móvil. |
| HttpPostAux.java | Esta es una clase auxiliar que se utiliza para establecer la conexión http al servidor de la base de datos. |

Tabla 8 Clases de la Aplicación CobroNFC.

Fuente: Realización Propia.

En la clase ***Pantalla_Cobro.java*** se configuran los métodos para que esta aplicación sea capaz de, implementando las herramientas del sistema, leer el contenido de la etiqueta compartida por el usuario para obtener la información contenida en ella y posteriormente enviarla al servidor de la base de datos para realizar las modificaciones necesarias.

Para poder realizar esta función, se hizo uso de los paquetes, clases y métodos definidos en la versión 10 del API de ANDROID que permiten y facilitan la realización de actividades con el uso de la tecnología NFC y los recursos de *hardware* del sistema.

Durante la investigación se observó que los mensajes intercambiados en una comunicación a través de NFC se realizan mediante el intercambio de mensajes NDEF, definidos anteriormente en el capítulo II. Para ello fue necesario entonces, realizar una búsqueda de los paquetes que ofrece el API de ANDROID para proceder a realizar la lectura de estos mensajes.

Se implemento el paquete "***android.nfc***" que proporciona diferentes clases y métodos para hacer uso de la tecnología NFC y, entre otras funciones, permitir realizar la lectura del mensaje NDEF de las etiquetas NFC.

A continuación se presenta una tabla con las clases de dicho paquete.

| CLASE | DESCRIPCIÓN |
|---|---|
| NdefMessage | Representa el mensaje NDEF recibido que no puede ser modificado. |
| NdefRecord | Representa un registro de NDEF que no puede ser alterado |
| NfcAdapter | Representa el adaptador local NFC del sistema. |
| NfcEvent | Encapsula la información relacionada a cualquier evento NFC. |
| NfcManager | Maneja a un alto nivel los eventos del NfcAdapter. |
| Tag | Representa una etiqueta NFC que ha sido descubierta o detectada. |

Tabla 9 Clases del paquete android.nfc del API.

Fuente: Elaboración Propia.

Esta información obtenida de la etiqueta, que es el nombre del usuario, será cifrada a través de un hash SHA-1 mediante el mismo método que se implemento en la aplicación anterior. Esto se realizó con la finalidad de proteger la información del usuario y evitar que se utilice para fines ilícitos.

Otra función de esta nueva clase será la de solicitar a la persona que esté realizando el cobro que introduzca el monto o que se configure previamente en la aplicación en caso de que sea un monto fijo.

A continuación, se presentan los métodos que conforman la clase ***Pantalla_Cobro.java***.

| MÉTODO | DESCRIPCIÓN |
|---|---|
| onCreate(Bundle savedInstanceState) | Este método es creado automáticamente al momento de ser iniciada una actividad con un layout, de forma de que alli se declaren los componentes de esta pantalla. Aca también se declarará que se utilicen las herramientas del sistema para realizar la lectura de la etiqueta NFC. |
| onPause () | En este método se configura que el adaptador NFC se coloque en *stand by* deshabilitandolo. |
| onResume() | En este método se evalua la condición del adaptador NFC, si esta habilitado o no. Si esta habilitado se activa para recibir una etiqueta, y si esta desabilitado se comunica a través de un mensaje que se active. |
| onNewIntetn(Intent intent) | En este método se configura que se utilice la clase NFCIntent para realizar la lectura del mensaje NDEF y determinar el contenido del mismo. |
| onSalir (View boton) | Este método se encarga de finalizar la actividad una vez se haya presionado el botón Finalizar al haberse realizado la transacción. |
| onCobrar(View boton) | Este método se encarga de pasar a la clase Pantalla_Final donde se realizará el cobro por el monto introducido por el usuario. |
| hasha | Este método se encarga de realizar la función de compendio de mensaje obteniendo como parámetro de entrada la contraseña a la que le crea un hash con SHA-1 |

Tabla 10 Métodos de la clase Pantalla_Cobro.java

Fuente: Elaboración Propia.

Adicionalmente se implemento la clase ***NFCIntent.java*** para realizar la lectura del mensaje NDEF que se encuentre en la etiqueta detectada, en esta clase se implementaron las clases descritas del paquete ***androif.nfc***.

Los métodos implementados en esta clase fueron los siguientes.

| MÉTODO | DESCRIPCIÓN |
|---|---|
| read() | Si se detecto algún mensaje NDEF, este método realiza un llamado al método readNdefTag() |
| readNdefTag() | En este método se realiza la lectura del mensaje NDEF y del registro. |
| String getText (final byte[] payload) | Este método interpreta el mensaje recibido y lo convierte en un objeto de tipo String. |
| NdefRecord newTextRecord | Este método interpreta el registro del NDEF recibido. |

Tabla 11 Métodos de la clase NFCIntent.

Fuente: Elaboración Propia.

Finalmente está la clase ***Pantalla_Final.java*** que es la encargada de realizar el cobro al usuario por el monto establecido, para esto se utilizará de nuevo la clase ***HttpPostAux.java*** que fue definida anteriormente durante el desarrollo de la aplicación del dispositivo móvil.

| MÉTODO | DESCRIPCIÓN |
|---|---|
| onCreate(Bundle savedInstanceState) | Este método es creado automáticamente al momento de ser iniciada una actividad con un layout, de forma de que alli se declaren los componentes de esta pantalla. |
| cobrostatus(String username, String montocobrar) | En este método se realizó la configuración del proceso de conexión a la base de datos para realizar el cobro en la base de datos. |
| Clase Login extends AsyncTask | Esta clase es la encargada de realizar la conexión con el servidor de base de datos para la realización del cobro, todo esto en segundo plano. |
| onFin() | Este método se utiliza para que, una vez finalizada la transacción, vuelva a la pantalla inicial (Pantalla_Cobro) donde podra iniciar un nuevo cobro |

Tabla 12 Métodos de la clase Pantalla_Final.

Fuente: Elaboración Propia.

Igualmente, se implementara la clase del API de ANDROID AsynTask para que el proceso de la conexión a la base de datos y la modificación que se debe realizar se ejecuten en segundo plano, de forma que al finalizar la consulta,

se envíe de nuevo, desde el servidor hacia la aplicación, un arreglo JSON que contendrá un 1 o 0 según se haya realizado con éxito o no la conexión con el servidor.

Adicionalmente, para brindarle algún grado de protección a los datos enviados al servidor fueron igualmente encriptados con la implementación del método ***"hasha"***.

## IV.4.- Implementación del prototipo:

La implementación de un prototipo se realizó con la finalidad de demostrar el funcionamiento de todos los componentes que en conjunto conformaran el sistema. Las principales actividades que se determinaron claves para establecer que el sistema estaba funcional fueron:

- Correcto funcionamiento de la aplicación del dispositivo móvil del usuario.
- Verificar la conexión del dispositivo móvil con la base de datos para la autenticación del usuario y posteriormente para la consulta de saldo.
- Verificar que la etiqueta compartida fuera leída por el dispositivo de cobro.
- Conexión del dispositivo de cobro con la base de datos para la realización del cobro.

Para la realización de esta demostración se realizó la implementación de una red de conexión muy simple con pocos beneficios de seguridad, esto por el hecho de que no se está haciendo uso de dinero ni datos reales. Sin embargo al momento de realizar la implementación en una aplicación real de este sistema, será sumamente importante la implementación de técnicas y tecnologías que brinden un gran soporte de seguridad a los datos que se están transfiriendo en la red de forma de garantizar que nadie, que no esté autorizado, pueda hacer uso de ellos.

Sin embargo, se tomo en cuenta el uso de encriptación de los datos a través de un hash SHA-1 para la información enviada al servidor a través del post http.

# V.- RESULTADOS.

En el presente capítulo se mostrarán en detalle los resultados obtenidos, que dan cumplimiento a los objetivos planteados en el presente trabajo especial de grado, los cuáles en conjunto dan por finalizado el desarrollo de la tecnología NFC para medios de pago a través de dispositivos móviles ALCATEL ONE TOUCH.

## V.1.- Características de un Medio de Pago

Luego de haber obtenido todos los datos de la observación de los diferentes medios de pago, se procedió a hacer una comparación entre ellos con el fin de determinar cuáles son las características básicas que deben cumplir y cuáles son los elementos o factores que siempre deben estar presentes en un sistema de medios de pago.

Primeramente se observa que en todos los medios de pago estudiados, pago en efectivo, pago con tarjetas y pago con cheque, existen tres elementos importantes los cuáles siempre están bien definidos y siempre están presentes para finalizar la transacción tal como se muestra en la siguiente tabla.

| MEDIO DE PAGO | ELEMENTOS | | | CARACTERÍSTICAS | | |
|---|---|---|---|---|---|---|
| | Comprador | Vendedor | Banco o Registro | Identificación | Seguridad | Integridad |
| Efectivo | SI | SI | No necesario | Necesario | No en todos los casos | No en todos los casos |
| Tarjeta | SI | SI | SI | Necesario | Necesario | Necesario |
| Cheque | SI | SI | SI | Necesario | Necesario | Necesario |

Tabla 13 Observación de los Medios de Pago.

Fuente: Elaboración Propia.

En la tabla se destaca que en todos los medios de pago observados, es necesaria la presencia del comprador ya que es la persona dispuesta a realizar un pago a través de cualquiera de los métodos existentes, adicionalmente la presencia de un establecimiento o persona que realice la venta del bien o servicio por el que

el comprador esté dispuesto a pagar. En cuanto a la comunicación con el banco y al almacenamiento de los datos de la transacción no se aplica en todos los casos, ya que durante el estudio se observó que al momento de realizar un pago en efectivo, a pesar de las exigencias de la ley, no en todos los casos los vendedores registran la compra y no dan la factura al comprador.

Por otra parte, otro de los factores que fueron centro de atención de este estudio fueron las características que cumplen y siguen los distintos medios de pago. Estas características observadas fueron la identificación o autenticación de las partes involucradas (comprador y vendedor), la seguridad que existe en la transacción y la integridad de los datos que resulten de la misma.

En las tres formas de pago estudiadas, se notó que en todos los casos es necesario tener plenamente identificado quien es el comprador y quien es el vendedor. En cuanto a los otros dos puntos se observó que no era necesario en todos los casos, sin embargo por ser el sistema que diseñado un sistema de pago electrónico, se debía tener en cuenta estas otras dos características llegando así entonces a que las características necesarias son las siguientes:

- **Autenticación:** en toda transacción para realizar el pago de algún bien y/o servicio, es necesario tener muy claro y muy bien identificada a la persona que está realizando el pago y del que está recibiendo el dinero, de forma tal que no exista usurpación de identidad. La forma de autenticación más común es aquella que implica el uso de claves como por ejemplo al momento de pagar con una tarjeta de débito.

- **Seguridad:** va ligado con la autenticación, implica que el peligro de usurpación de identidad y el uso de la cantidad de dinero que tenga el usuario en su cuenta no sea usado por otra persona sino por el mismo. En la actualidad también existe el problema de sufrir algún robo y que se pueda disponer de su dinero sin autorización, el caso más común es con el pago en efectivo.

– **Integridad de los datos:** implica que al momento de realizar una transacción donde estén involucrados los datos que estén ya sea, en una tarjeta con chip, en un punto de venta, en una etiqueta o en una base de datos, puedan ser intercambiados de manera correcta sin que exista perdida de la misma.

Es decir, a pesar de lo diferente que son las negociaciones de los distintos medios de pago, todos ellos poseen puntos en común como lo son el manejo de información y manejo de dinero, es por esto que cualquier medio de pago debe cumplir con un mínimo de requisitos de seguridad. Está seguridad estará dividida en las etapas establecidas en el capítulo II como lo son autenticación, integridad, confidencialidad y no repudio.

## V.2.- Diseño del Sistema de Medio de Pago con NFC

En base a los sistemas de medio de pago ya existentes, como el uso de tarjetas de débito o crédito, y cumpliendo con las características y elementos determinados luego del estudio sobre los medios de pago, se procedió a realizar un diseño de un sistema que cumpliera con las mismas características y que permitiera realizar una transacción de pago a través de NFC.

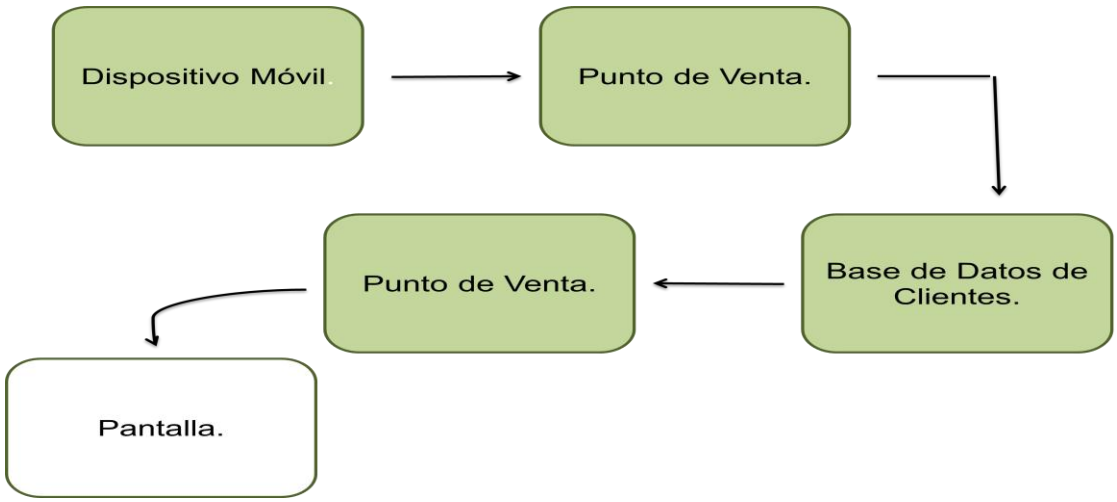


Figura 14 Diagrama de Bloques del Sistema

Fuente: Elaboración Propia.

Se planteó un sistema donde se permita que un usuario, que este registrado en la base de datos de un establecimiento específico o de un banco, logre realizar el pago por un monto determinado a través de su dispositivo móvil compartiendo una etiqueta NFC con el punto de venta.

En la base de datos estará información de los usuarios que se hayan registrado. Dicha información contendrá datos que permitan la identificación del cliente así como datos que identifiquen la cuenta que éste tenga asociada.

Para la implementación del prototipo, se utilizó dinero ficticio, es decir, no hace falta para las pruebas del sistema el uso de moneda real. Además de eso se planteó que sea un sistema prepago en el que cada usuario tenga alguna cantidad de ese dinero abonada en su cuenta de la base de datos.

Para recargar el saldo, el usuario deberá dirigirse a las oficinas del establecimiento donde abonará la cantidad de dinero que desee a su cuenta, de esta manera podrá disponer de dicho saldo para realizar pagos.

Cada usuario será identificado al momento de realizar un pago a través del nombre de usuario que será único para cada cliente y todas las consultas a la base de datos serán realizadas a raíz de este dato.

Tal como lo indica la figura, para su funcionamiento, el sistema cumplirá con los siguientes pasos para lograr completar una transacción:

- El dispositivo móvil hará contacto con el punto de venta. Para que comience la negociación NFC, será necesario correr una aplicación en el dispositivo móvil con sistema operativo ANDROID, que pida una autenticación de usuario para asegurar que sea este solamente quien pueda disponer del monto de su cuenta.
- Al momento de que el punto de venta reconoce al usuario, a través de una identificación única, consultara en un servidor donde esté la base de datos con la información de todos los usuarios, la existencia del mismo y si dispone de monto suficiente para el pago que desea realizar.

- En el servidor se ejecutará un programa que consiste en realizar el cobro al cliente, es decir, en modificar la cantidad de saldo disponible disminuyéndola en el monto que desea pagar.
- Al finalizar este proceso, se devolverá la respuesta al punto de venta. En el se mostrará por pantalla el nombre del usuario y el resultado de la transacción, si fue exitosa o si falló.
- Al realizar esta impresión por pantalla, quedará finalizada la transacción.

## V.3.- Aplicación en ANDROID para el dispositivo móvil

Luego de tener el diseño del sistema, se realizó la aplicación para el dispositivo móvil, a continuación se mostrará los resultados de la misma.

### V.3.1.- Pantalla de Inicio

Esta es la pantalla inicial de la aplicación para el dispositivo móvil ANDROID, en ella se muestran dos cuadros de texto para que el usuario introduzca su nombre de usuario y contraseña para la autenticación. Adicionalmente la pantalla contiene dos botones, una para salir del sistema y otro para ingresar al sistema realizando la consulta a la base de datos.

En la siguiente figura se puede observar la pantalla de inicio de la aplicación:


Figura 15 Pantalla de Inicio de la Aplicación

Fuente: Elaboración Propia.

Durante la autenticación y la conexión a la base de datos, se mostrara una ventana en la pantalla del dispositivo que indicará que se está realizando la acción como se muestra en la figura.

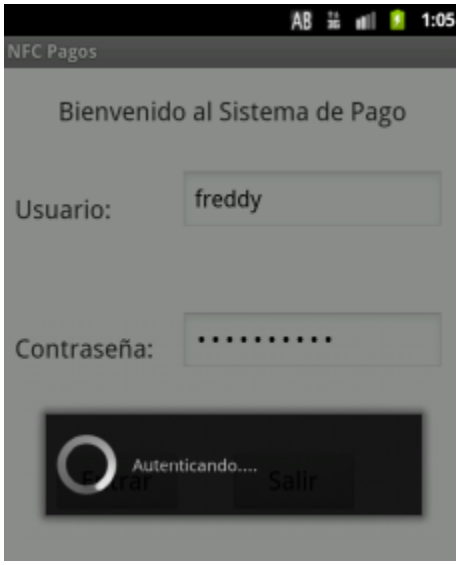


Figura 16 Mensaje de Autenticación en la Pantalla Inicio

Fuente: Elaboración Propia

### V.3.2.- Pantalla de Opciones

En caso de que el usuario exista y la autenticación se realizó correctamente, se pasará a mostrar la segunda pantalla llamada Pantalla Opciones. En ella se muestra un mensaje dándole la bienvenida al usuario con su nombre y se tienen tres botones, uno para la opción consultar saldo, otro para la opción de realizar pago y finalmente el botón para salir.

En la figura que se muestra a continuación, se observa cómo está conformada la pantalla de opciones de la aplicación.

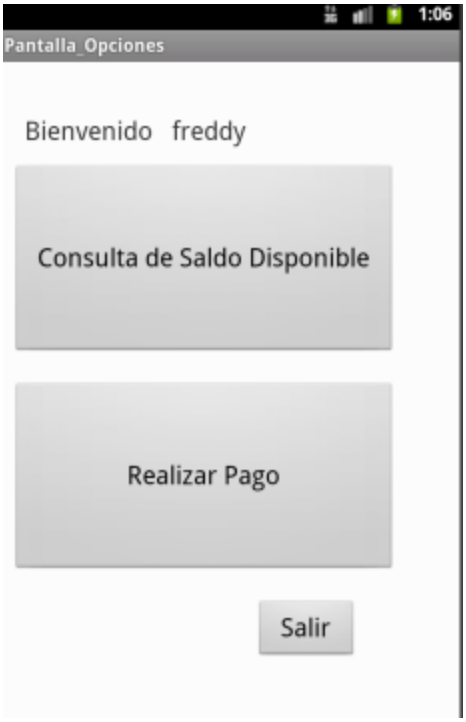


Figura 17 Pantalla de Opciones de la Aplicación

Fuente: Elaboración Propia.

Al presionar el botón de consultar saldo se pasará a la Pantalla Saldo que se describirá un poco más adelante.

Si se presiona el botón realizar pago, se pasará entonces a las herramientas del sistema operativo donde se manejan las etiquetas NFC para proceder a realizar la transacción.

### V.3.3.- Pantalla de Saldo

Como se mencionó anteriormente, al presionar el botón de consultar saldo, se pasará a esta pantalla donde se mostrará al usuario cuál es su saldo disponible.

Al momento de iniciar esta pantalla, aparecerá un cuadro de dialogo que le indicará al usuario que se está realizando la consulta con el servidor de base de datos. Al finalizar la consulta, se obtendrá el resultado y se mostrará en la pantalla de la siguiente manera.


Figura 18 Resultado de la Pantalla Saldo de la Aplicación

Fuente: Elaboración Propia.

## V.4.- Implementación del Prototipo

Finalmente, al tener todos los elementos relacionados con los productos de los objetivos específicos y al desarrollo del trabajo especial de grado, se procedió a realizar la prueba final de todos los elementos en conjunto para demostrar el funcionamiento de este sistema diseñado para realizar un pago a través de NFC.

El primer paso es que el usuario realice la autenticación, ingresando a la aplicación ***NFCPagos*** donde deberá ingresar su usuario y contraseña para entrar al sistema. Una vez en el sistema, y tal como se menciono anteriormente, el usuario podrá realizar una consulta de saldo o un pago.

Al seleccionar la opción de Realizar Pago se mostrará el Manejador de etiquetas NFC del sistema operativo y posteriormente deberá acercar el dispositivo móvil al terminal que funcionará como punto de venta. Al acercarse, el

terminal leerá la información de la etiqueta y procederá a realizar la conexión a la base de datos para realizar el cobro como se muestra a continuación.

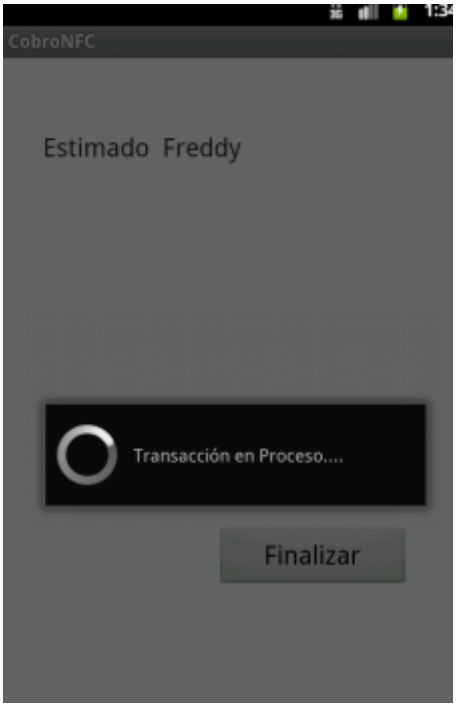


Figura 19 Realización del Cobro

Fuente: Elaboración Propia.

Luego de que se realice la conexión a la base de datos y se finalice el proceso de agregar la compra a la tabla de la base de datos y de disminuir el saldo del usuario, se mostrará en la pantalla del terminal que funciona como punto de venta el siguiente mensaje.

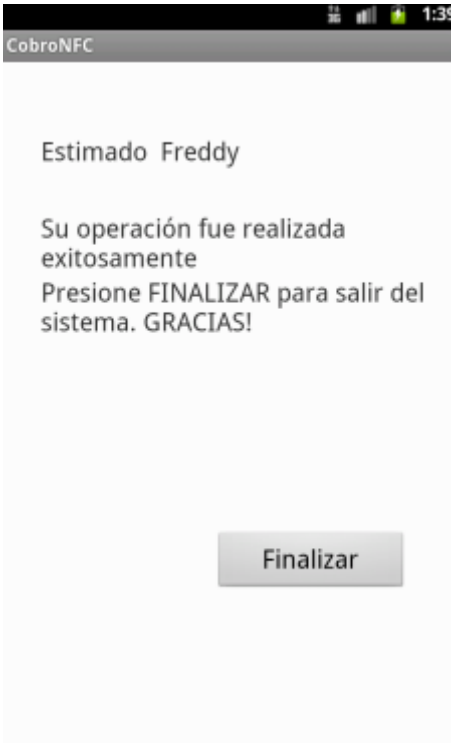


Figura 20 Mensaje por pantalla de Cobro Exitoso

Fuente: Elaboración Propia.

La implementación del sistema se puede realizar tanto a través de una red interna utilizando Wi-Fi o mediante la conexión a la red móvil; sin embargo, debido a problemas presentes en el dispositivo utilizado para las pruebas con respecto a la conexión Wi-Fi, la conexión del dispositivo móvil al servidor se realizó a través de la red móvil.

Habiendo mencionado esta situación, al realizar la conexión a la base de datos desde un dispositivo móvil, ya sea del dispositivo del usuario o desde el dispositivo que cumpla las funciones del punto de venta, la conexión se establece a través de la red móvil del proveedor de servicios de cada uno de los dispositivos, tal como se muestra a continuación.

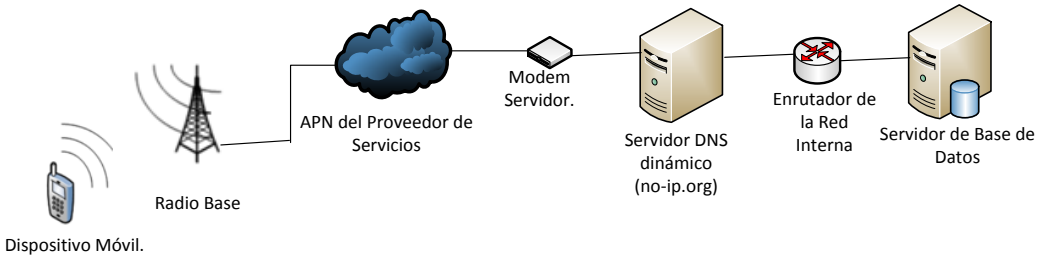


Figura 21 Arquitectura de la Red para la Conexión del Servidor

Fuente: Elaboración Propia.

Como se mencionó anteriormente, la comunicación entre el dispositivo móvil y el servidor se hará mediante la red pública del proveedor de servicios, es decir el dispositivo se conectará a internet a través de la radio base a la que se encuentre conectado en ese momento, posteriormente irá a través de la nube hasta el servidor DNS de nuestro servidor que es proporcionado mediante la herramienta de no-ip.org y luego de esto ya se encontrará en la red de nuestro servidor.

Adicionalmente es necesario configurar en el enrutador de la red interna la tabla NAT para traducir la dirección pública de donde proviene la solicitud hacia el puerto 80 que es el puerto por donde se manejan las solicitudes de HTTP, por lo que es necesario que este abierto para estas direcciones ya que se requiere la conexión.

La comunicación entre el servidor WEB con el servidor de base de datos, se hace mediante el uso de la configuración por defecto de MySQL, es decir que la comunicación se establece a través del puerto TCP 3306 en la dirección del localhost 127.0.0.1 para conexiones externas.

Tal como se mencionó en el capítulo anterior, para aplicaciones reales de este sistema es necesario el uso de diferentes protocolos que brinden seguridad a la red que se utiliza, en la investigación realizada se obtuvo que algunas de las estrategias que se implementan en una red son el uso de servidores de seguridad, el uso de redes privadas virtuales con túneles directos entre dos puntos y utilizando protocolos de seguridad como IPSec.

En vista de esto, se diseño una topología de la posible red que podría ser aplicada donde se ofrece gran seguridad a los datos, lo que es sumamente necesario ya que se habla de dinero y datos del usuario.

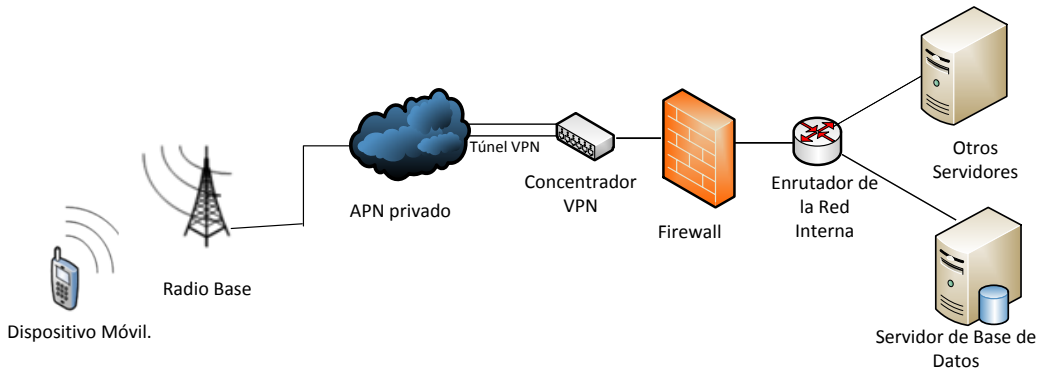


Figura 22 Topología de red con Seguridad.

Fuente: Elaboración Propia.

En la topología mostrada en la figura anterior, se puede observar cómo se implementarían alguno de los protocolos de seguridad de redes para garantizar que no haya intromisiones en la red de nuestra aplicación.

Para entender un poco más el diseño, se hará un ejemplo con un caso de una posible aplicación del sistema.

Supongamos que se desea implementar el sistema en una línea de transporte específica, para esto será necesario que cada una de las unidades cuente con un dispositivo de cobro que tenga posibilidad de conexión instantánea en cualquier lugar que se encuentre, se propone el uso de GPRS ya que permite la conexión a la red de dispositivos móviles. Adicionalmente, la organización deberá contar con el servidor de base de datos donde se registren los usuarios de este sistema, tal como se planteó en el diseño original.

Una vez teniendo estos elementos se procederá a establecer un proceso de negociación con las operadoras de telefonía móvil a fin de determinar con cuál operadora se trabajará. Se propone el uso de un APN privado, que será diseñado y brindado por la operadora, adicionalmente sobre este APN se implementará un túnel VPN que conecte de forma directa al dispositivo móvil que se encuentre instalado en la unidad con la red de la organización.

Sin embargo, a pesar de la topología de red propuesta, ésta no es la única forma de implementar el sistema con una red segura. La implementación del sistema, puede realizarse sobre la red que ya disponga la organización o empresa que desee darle aplicación al sistema, es decir, no es esencial que se aplique esta topología planteada, se pueden adaptar los elementos necesarios de nuestro sistema a la red con la red que dispongan en la organización de manera de reducir costos de implementación y darle uso a sus equipos.

# VI.- CONCLUSIONES Y RECOMENDACIONES.

A continuación se presentarán las conclusiones y recomendaciones a las que se llegó luego de la realización del trabajo especial de grado.

## VI.1.- Conclusiones:

Una vez finalizado el presente trabajo especial de grado, se pueden llegar a las siguientes conclusiones.

Los medios de pago han cambiado a través de los tiempos adaptándose a los avances que va teniendo la sociedad, sin embargo, y a pesar de que la forma de realizar la transacción sea mediante un trueque o mediante un pago electrónico los medios de pago siempre han de cumplir con las tres características principales que son tener claramente definidas quienes son las personas involucradas en el pago, asegurarse de que solo ellas pueden disponer del resultado de la transacción ya sea del dinero o de los productos y que debe existir una integridad en el resultado de la transacción al momento de finalizarla.

Un sistema de medio de pago, ya sea inalámbrico o no, implementando la tecnología NFC o cualquier otra siempre necesitará de unos elementos básicos como lo son el cliente que será la persona dispuesta a pagar, la persona que ofrece el producto a la venta y un lugar donde almacenar o registrar esa transacción realizada. En este caso estará conformado por un dispositivo móvil que será del cliente, el dispositivo punto de venta que será del vendedor y el servidor de base de datos donde se almacenará la transacción realizada en la cuenta del cliente.

Al momento de desarrollar una aplicación para cualquier lenguaje de programación, siempre es importante tener claras cuáles son las tareas que se desea lleve a cabo la aplicación objeto del diseño de manera de poder solucionar el problema. En este caso se pudo observar que para el desarrollo de aplicaciones en ANDROID, es necesario tener una base de conocimiento de la lógica de la programación basada en objetos, luego de esto el desarrollo mediante la interfaz de desarrollo integrado ECLIPSE es muy amigable ya que cuenta con gran serie

de opciones para facilitar la programación. Igualmente el uso de los API de cada entorno de desarrollo facilita mucho el desarrollo ya que brinda todas las clases y herramientas disponibles para realizar alguna tarea específica.

A pesar de que al momento de la implementación del prototipo se presentó la gran dificultad de que no se pudo contar con el dispositivo de punto de venta que se esperaba al momento del inicio del proyecto, se logró realizar la implementación solucionando el problema presentado mediante el uso de otro dispositivo móvil ANDROID que permitiera que se realizara la conexión NFC entre él y el dispositivo del usuario. La transacción se realizó con éxito, sin embargo, al momento de implementar el sistema es necesario estar en una zona donde haya suficiente cobertura de la red de telefonía móvil para poder realizar la conexión al servidor WEB donde está la base de datos, en caso de no haber cobertura la transacción no podrá ser realizada.

Como punto final se pudo concluir que el tema de la seguridad es de suma importancia al momento de diseñar cualquier sistema de comunicación, más aún si se trata de un sistema de medio de pago. El término seguridad abarca entonces puntos clave como son la autenticación, integridad, confidencialidad y no repudio; donde todos y cada uno de estos puntos se encarga de garantizar que una transacción fue realizada por la persona autorizada, que los datos enviados son los correctos, que nadie ajeno a la negociación conozca detalles de información y finalmente que las partes no puedan negar haberla realizado.

## VI.2.- Recomendaciones:

Para aplicaciones reales del sistema, es necesaria la implementación de elementos que brinden seguridad a la red como una infraestructura de red más robusta, pudiendo hacer uso de APN privados, túneles VPN, entre otras tecnologías que brinden mayor protección al sistema.

Se recomienda para futuras investigaciones en el campo de las telecomunicaciones, hacer gran énfasis en el tema de seguridad ya que este es un punto importantísimo en un sistema de cualquier tipo, sobre todo los que manejen

datos importantes de personas u organizaciones. Además, es una tarea fundamental del Ingeniero en Telecomunicaciones velar porque la información que está siendo transmitida logre llegar al otro extremo de la comunicación de forma fidedigna sin ningún tipo de alteración o manipulación por agentes extraños al sistema.

Durante la investigación para el desarrollo del proyecto, se presentaron varias opciones de realizar un sistema de medio de pago. Con el fin de abrir el campo de aplicación de este sistema y no limitarlo solamente a dispositivos ANDROID con NFC, se recomienda que para futuras investigaciones se desarrolle el sistema diseñado desarrollando las aplicaciones en otros ambientes de desarrollo, o mediante la utilización de otras formas de pago como el uso de HTML para realizar el proceso.

Finalmente se recomienda estudiar la posible implementación de esta investigación en las siguientes aplicaciones:

- Pago del pasaje de transporte público en alguna línea de prueba por ejemplo de autobuses o metrobuses.
- Uso del sistema en restaurantes para realizar pedidos y/o pagos a través de etiquetas en el menú o en un punto de fácil utilización para el usuario.
- Para realizar el pago y control de acceso a estacionamientos o a establecimientos.

# VII.- BIBLIOGRAFIA.

Adarve Corporación Jurídica, (sf). "Medios de Pago". FC Editorial.

Alvarellos D, Pastor G (2012). "Guia JAVA para docentes. Nivel 1"

Banco Central de Venezuela (2007). "Innovación en los Sistemas de Pago Minoristas en Venezuela 2006". Recuperado de http://www.bcv.org.ve/snp/Innovacionesdef.pdf.

Basterretche, J.(2007) "Dispositivos Móviles". Recuperado en marzo de 2012 de http://exa.unne.edu.ar/depar/areas/informaticaSistemasOperativos/tfbasterretche.pdf

Baz A., Ferreira I., Rodríguez M., García R. (s.f) "Dispositivos móviles"

CISCO SYSTEMS (s.f.) Layer 2 Tunnel Protocol. Recuperado de http://www.cisco.com/en/US/docs/ios/12_0t/12_0t1/feature/guide/l2tpT.html.

Cornel, G., Hortsmann C. (2001) "Core JAVA 2: Fundamentals" (Quinta Edición) Sun Microsystems Press.

Dattatreya, M y Kumar P (2002). IPsec VPN Fundamentals. Tech Online. Recuperado de: http://www.eetimes.com/design/communications-design/4017938/IPSec-VPN-Fundamentals.

España M (2003). Servicios Avanzados de Telecomunicaciones. Ediciones Díaz De Santos. Recuperado de: http://books.google.co.ve

Garcerant I (2012) Comunicaciones Personales.

García J (2007). "Medios de Pago". Recuperado de http://di002.edv.uniovi.es/~fanjul/ce/descargas/CE-Transparencias-Tema3-v2007.pdf

Google Inc. (s.f) API for Android Developers. Recuperado de http://developer.android.com/reference/packages.html

Gutiérrez J. (2004), "Entorno de Desarrollo Eclipse". Universidad de Valencia.

Ingenico – Sitio Oficial. Recuperado en junio de 2012 de http://www.ingenico.com/

International Organization for Standardization (2004) "ISO/IEC 18092 – Near Field Communication Interface and Protocol 1 (NFCIP-1)".

Joskowicz J (2008). Redes de Datos. Universidad de Montevideo. Recuperado de:http://iie.fing.edu.uy/ense/asign/redcorp/material/2008/Redes%20de 20 Datos%202008.pdf

Kasera S, Narang N (2004). 3G Networks. Architecture, Protocols and Procedures. Tata McGraw-Hill Professional. Recuperado de: http://books.google.co.ve

NFC Forum. Recuperado en marzo de 2012, de http://www.nfc-forum.org/home/

Nokia Developer (2011). Introduction to NFC. Versión 1.1

Nosowitz, D (2011). "Everything You Need to Know About Near Field Communication". Popular Science Magazine. Popular Science.

Ortiz, C. (2006). "An Introduction to Near-Field Communication and the Contactless Communication API"

Schildt, H. (2007). "Fundamentos de JAVA". (Tercera Edición) Mc Graw Hill.

Tanenbaum, A (2003). Redes de Computadoras. (Cuarta Edición). Prentice Hall.

The New Boston (s.f) "List of Videos for Android Application Development"

Recuperado en junio de 2012 de http://thenewboston.org/list.php?cat=6

Vedat C, Ok K, Ozdenizci B. (2012) "Near Field Communication From Theory to Practice". Wiley.

Verifone - Sitio Oficial. Recuperado en junio de 2012 de http://www.verifone.com/

# APÉNDICES.

# Apéndice A: Características del ALCATEL OT996.

| | | |
|---|---|---|
| **GENERAL** | **Red** | GSM 850 / 900 / 1800 / 1900 - HSDPA 900 / 2100 |
| | **Anunciado** | 2012 |
| | **Status** | Disponible |
| **TAMAÑO** | **Dimensiones** | 127 x 68 x 9.75 mm |
| | **Peso** | 124 g |
| **DISPLAY** | **Tipo** | TFT touchscreen capacitivo, 16M colores |
| | **Tamaño** | 480 x 800 pixeles, 4.3 pulgadas |
| | | - Sensor acelerómetro para auto rotación<br>- Controles sensibles al tacto<br>- Soporte Multitouch<br>- Sensor de proximidad para auto apagado |
| **RINGTONES** | **Tipo** | Polifónico, MP3 |
| | **Customización** | Descargas |
| | **Vibración** | Si |
| | | - Conector de audio 3.5 mm |
| **MEMORIA** | **Agenda telefónica** | Entradas y campos prácticamente ilimitados, Foto de llamada |
| | **Registro de llamadas** | Prácticamente ilimitado |
| | **Slot de tarjeta** | microSD hasta 32GB |
| | | - 2GB memoria interna (1GB disponible al usuario), 512MB RAM<br>- Procesador Qualcomm MSM8255 1.4GHz |
| **CARACTERÍSTICAS** | **GPRS** | Si |
| | **Velocidad de datos** | |
| | **OS** | Android OS, v2.3.5 Gingerbread |
| | **Mensajería** | SMS, MMS, Email, Push Email, IM |
| | **Navegador** | HTML |
| | **Reloj** | Si |
| | **Alarma** | Si |
| | **Puerto infrarrojo** | No |
| | **Juegos** | Si + descargables |

| | | |
|---|---|---|
| | **Colores** | Blanco Puro, Negro, Rojo, Fucsia, Negro Blanquecino |
| | **Cámara** | 5 MP, 2592x1944 pixels, autofocus, flash LED, geo-tagging, video 720p@30fps, cámara frontal VGA<br><br>- GPS con soporte A-GPS<br>- Brújula digital<br>- EDGE<br>- 3G HSDPA 7.2 Mbps / HSUPA 5.76Mbps<br>- Wi-Fi 802.11 b/g/n; Wi-Fi Direct<br>- Bluetooth v3.0 A2DP<br>- microUSB 2.0<br>- NFC<br>- Integración con redes sociales<br>- Cancelación activa de ruido con micrófono dedicado<br>- Voz HD<br>- Integración Google Search, Maps, Gmail, YouTube, Google Talk, Picasa<br>- Reproductor de audio MP3/AAC+/WAV/WMA<br>- Reproductor de video MP4/H.264<br>- Radio FM Stereo con RDS<br>- Organizador<br>- Visor de documentos<br>- Memo de voz<br>- Manoslibres incorporado<br>- Ingreso predictivo de texto |
| **BATERÍA** | | Standard, Li-Ion 1500 mAh |
| | **Stand-by** | Hasta 360 h (2G) / Hasta 288 h (3G) |
| | **Tiempo de conversación** | Hasta 14 h (2G) / Hasta 7 h (3G) |

# Apéndice B: Código de la clase Pantalla_Login.java

```
package com.example.nfcpagos;

import java.util.ArrayList;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import com.example.nfcpagos.Pantalla_Login;
import android.os.AsyncTask;
import android.os.Bundle;
import android.app.Activity;
import android.app.ProgressDialog;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.content.Intent;
import android.widget.Toast;
import com.example.nfcpagos.HttpPostAux;
import com.example.nfcpagos.Pantalla_Opciones;
import com.example.nfcpagos.R;


/*
 * En esta clase se configuraran todas las acciones de la pantalla principal de la
* app. En ella se le solicitaran al usuario los datos de Autenticación al sistema y
* con ellos se realizara la consulta a la base de datos
 */

public class Pantalla_Login extends Activity {

	Button entrar,salir;
	EditText user,psw;
	TextView display;
	HttpPostAux post = new HttpPostAux();
	String URL_server = ""; //DirecciÃ³n del servidor de base de datos
	private ProgressDialog pDialog;

	/*
	 * En este mÃ©todo se configura la pantalla inicial y se declaran sus
elementos
```

```
 */

@Override
protected void onCreate(Bundle savedInstanceState) {

	super.onCreate(savedInstanceState);
	setContentView(R.layout.pantalla_login_lo);

	entrar = (Button) findViewById(R.id.entrarButton);
	salir = (Button) findViewById(R.id.salirButton);
	user = (EditText) findViewById(R.id.userEditText);
	psw = (EditText) findViewById(R.id.pswEditText);

}

/*
 * En este método se configuran las acciones del botón ENTRAR, el
* cuál tomara los datos de entrada  y realizara la consulta a la BD
 */
public void onEntrar(View boton){

	String usuario,clave;
	usuario = user.getText().toString();
	clave = psw.getText().toString();

	if (usuario.equals("")||clave.equals("")){
		error1();
	}
	else{
		new Login().execute(usuario,clave);
	}
}

/*
 * En este método se configura el botón SALIR
 */
public void onSalir(View boton){
	System.exit(0);
}

/*
 * En este mÃ©todo se configura un mensaje de ERROR si existe alguno
* en cuanto al nombre de usuario o contraseña
 */
public void error1(){
	Toast error1 = Toast.makeText(getApplicationContext(),"Error:
Nombre de usuario o contraseÃ±a en blanco", Toast.LENGTH_SHORT);
```

```
                error1.show();
        }

        /*
         * Este método muestra un mensaje de ERROR si ocurrió alguno
        * durante la conexión con la BD para la autenticación.
         */
        public void error2(){
                Toast error2 = Toast.makeText(getApplicationContext(),"Error en
la autenticaciÃ³n. Intente de nuevo", Toast.LENGTH_SHORT);
                error2.show();
        }

        /*
         * Valida el estado de la autenticaciÃ³n solamente necesita como
        * parámetros el usuario y la clave
        */
   public boolean loginstatus(String username ,String password ) {

        int logstatus=-1;

        /*
         * Se crea un ArrayList nombre-valor, que contendrá los parámetros de
        * entrada (usuario, clave) para enviarlo como un POST a la BD
         */
        ArrayList<NameValuePair> parametros= new ArrayList<NameValue
Pair>();

                parametros.add(new BasicNameValuePair("usuario",username));
                parametros.add(new BasicNameValuePair("password",password));

        /*
         * Se realiza la consulta a la BD haciendo una conexión al servidor WEB
        * donde está alojada con  los parámetros obtenidos y se obtiene como
        *  respuesta un arreglo JSON
        */
      JSONArray jdata=post.getserverdata(parametros, URL_server);

                if (jdata!=null && jdata.length() > 0){
                        JSONObject json_data;
                        try {
                                json_data = jdata.getJSONObject(0);
                                logstatus=json_data.getInt("logstatus");
                                Log.e("loginstatus","logstatus= "+logstatus);
                        }
                        catch (JSONException e) {
                                e.printStackTrace();
```

```
                }

                /*
                 * Se valida el resultado de la consulta, si fue válido o no
                *  para ingresar a la App
                 */
                if (logstatus==0){
                        Log.e("loginstatus ", "invalido");
                        return false;
                }
                else{
                        Log.e("loginstatus ", "valido");
                        return true;
                }

        }
        else{
                Log.e("JSON  ", "ERROR");
                return false;
        }

}

    /*
     * CLASE ASYNCTASK: clase que se implementara para mostrar el
    * mensaje de Auntenticando y realizar dicho proceso en Background.
     */
     class Login extends AsyncTask< String, String, String > {

             String usr,pass;

             /*
            *Método para configurar que salga el mensaje Autenticando
            *durante el proceso de consultar  en la base de datos
            */
             protected void onPreExecute() {
                     //para el progress dialog
                     pDialog = new ProgressDialog(Pantalla_Login.this);
                     pDialog.setMessage("Autenticando....");
                     pDialog.setIndeterminate(false);
                     pDialog.setCancelable(false);
                     pDialog.show();
             }

             /*
              * Para realizar el proceso de conectarse a la BD y hacer la
              * autenticación en segundo plano
```

```
             */
            protected String doInBackground(String... params) {
                  //se extraen los parametros de entrada a la clase asynclogin
                  usr=params[0];
                  pass=params[1];

                  if (loginstatus(usr,pass)==true){
                        return "ok"; //login valido
                  }
                  else{
                        return "err"; //login invalido
                  }

            }

            /*
             * Luego de haber realizado el doInBackground pasamos a la
             * siguiente pantalla o se  muestra un error.
             */
            protected void onPostExecute(String result) {

                  pDialog.dismiss();//ocultamos progess dialog.
                  Log.e("onPostExecute=",""+result);

                  if (result.equals("ok")){

                        Intent i=new Intent(Pantalla_Login.this,
Pantalla_Opciones.class);
                        i.putExtra("Usuario",user.getText().toString());

                        startActivity(i);

                  }
                  else{
                        error2();
                  }

            }

      }

      @Override
      public boolean onCreateOptionsMenu(Menu menu) {
            // Inflate the menu; this adds items to the action bar if it is present.
            getMenuInflater().inflate(R.menu.pantalla_login_lo, menu);
            return true;
      }
```

## Apéndice C: Código de la clase Pantalla_Opciones.java

```
package com.example.nfcpagos;

import com.example.nfcpagos.Pantalla_Opciones;
import com.example.nfcpagos.R;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class Pantalla_Opciones extends Activity {

	Button salir2,pagar,consulta;
	TextView encabezado;

	Bundle bundle= getIntent().getExtras();
	String usuario = bundle.getString("Usuario");

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.pantalla_opciones_lo);

		salir2 = (Button) findViewById(R.id.salir2Button);
		encabezado = (TextView)
findViewById(R.id.encabezadoTextView);
		consulta = (Button) findViewById(R.id.consultaButton);

		encabezado.setText(encabezado.getText()+" "+usuario);
	}

	/*
	 * En este método se configura el botón SALIR que finaliza la sesión y
	* sale del sistema.
	 */
	public void onSalir2(View boton){
		System.exit(0);
	}

	public void onConsulta(View boton){

		Intent i=new Intent(Pantalla_Opciones.this, Pantalla_Saldo.class);
```

```
            i.putExtra("Usuario",usuario);
            startActivity(i);
        }

        public void onPagar(View boton){
            getSystemService(NFC_SERVICE);
        }


        @Override
        public boolean onCreateOptionsMenu(Menu menu) {
            // Inflate the menu; this adds items to the action bar if it is present.
            getMenuInflater().inflate(R.menu.pantalla_opciones_lo, menu);
            return true;
        }

}
```

## Apéndice D: Código de la clase Pantalla_Saldo.java

```
package com.example.nfcpagos;

import java.util.ArrayList;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import com.example.nfcpagos.Pantalla_Saldo;
import com.example.nfcpagos.R;
import android.os.AsyncTask;
import android.os.Bundle;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class Pantalla_Saldo extends Activity {

	Button atras;
	TextView encabezado2,saldo;
	HttpPostAux post = new HttpPostAux();
	String URL_server = ""; //Dirección del servidor de base de datos
	private ProgressDialog pDialog;

	Bundle bundle= getIntent().getExtras();
	String usuario = bundle.getString("Usuario");

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.pantalla_saldo_lo);

		encabezado2 = (TextView) findViewById(R.id.encabezado2Text
View);
		saldo = (TextView) findViewById(R.id.saldoTextView);
		atras = (Button) findViewById(R.id.atrasButton);

		encabezado2.setText(encabezado2.getText()+" "+usuario);
```

```
            new Consulta().execute(usuario);
        }


        public void onRegresar(View boton){
            Intent i=new Intent(Pantalla_Saldo.this, Pantalla_Opciones.class);
            i.putExtra("Usuario",usuario);
            startActivity(i);
            finish();
        }

        public void error3(){
            Toast error3 = Toast.makeText(getApplicationContext(),"Ocurrió
un error durante la consulta. Intente de nuevo", Toast.LENGTH_SHORT);
            error3.show();
        }

        class Consulta extends AsyncTask< String, String, String > {

            String usr;

            protected void onPreExecute() {
                pDialog = new ProgressDialog(Pantalla_Saldo.this);
                pDialog.setMessage("Consultando Saldo....");
                pDialog.setIndeterminate(false);
                pDialog.setCancelable(false);
                pDialog.show();
            }

            protected String doInBackground(String... params) {

                usr=params[0];
                float saldo=0;

            ArrayList<NameValuePair> parametros= new ArrayList<Name
ValuePair>();

            parametros.add(new BasicNameValuePair("usuario",usr));

             JSONArray jdata=post.getserverdata(parametros, URL_server);

           |if (jdata!=null && jdata.length() > 0){
            JSONObject json_data;
            try {
                json_data = jdata.getJSONObject(0);
                        saldo=json_data.getInt("saldo");
```

```
                        }
                        catch (JSONException e) {
                                e.printStackTrace();
                        }
                String disponible = Float.toString(saldo);
                return disponible;
                        }
                        else{
                                return "error";
                        }


                }

                protected void onPostExecute(String result) {

                        pDialog.dismiss();
                        Log.e("onPostExecute=",""+result);

                        if (result.equals("error")){
                                error3();
                        }
                        else{
                                saldo.setText(saldo.getText()+" "+result);
                        }
                }

        }

        @Override
        public boolean onCreateOptionsMenu(Menu menu) {
                // Inflate the menu; this adds items to the action bar if it is present.
                getMenuInflater().inflate(R.menu.pantalla_saldo_lo, menu);
                return true;
        }

}
```

## Apéndice E: Código php para la autenticación en la Base de Datos.

```
<?php

class login {

        private $db;

        // constructor

        function __construct() {
                require_once 'connectbd.php';
       // connecting to database

                $this->db = new DB_Connect();
                $this->db->connect();

        }

        // destructor
        function __destruct() {

        }


        $usuario = $_POST['usuario'];
        $passw = $_POST['password'];

        $result=mysql_query("SELECT COUNT(*) FROM cliente WHERE
usuario='$usuario' AND contraseÃ±a='$passw' ");
        $count = mysql_fetch_row($result);


        if($count[0]==0){
            $resultado[]=array("logstatus"=>"1");
        }else{
            $resultado[]=array("logstatus"=>"0");
        }

        echo json_encode($resultado);

}

?>
```

## Apéndice F: Código php para la consulta de saldo en la Base de Datos.

```
<?php

class consulta {

	private $db;

	// constructor

	function __construct() {
		require_once 'connectbd.php';
	// connecting to database

		$this->db = new DB_Connect();
		$this->db->connect();

	}

	// destructor
	function __destruct() {

	}


	$usuario = $_POST['usuario'];

	$result=mysql_query("SELECT saldodisponible FROM cuenta WHERE usuario='$usuario'");

	$resultado[]=array($result);

	echo json_encode($resultado);

}

?>
```