ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"Pago Electrónico a Través de Teléfonos Móviles"

TESIS DE GRADO

Previo a la obtención del Título de:

INGENIERO EN COMPUTACION ESPECIALIZACIÓN SISTEMAS DE INFORMACIÓN Y SISTEMAS TECNOLÓGICOS

Presentada por:

Fernando Paúl Espinosa Peñaherrera

Angel Fernando Soto Sarango

GUAYAQUIL - ECUADOR

AÑO 2009

A G R A D E C I M I E N T O

Ing. Bismar Salamea, por su guía en la elaboración de la tesis, al Ing. Francisco Novillo por ser nuestro director en la presente tesis.

DEDICATORIA

En primer lugar a Dios y la Virgen del Cisne, por haberme dado salud y ser mi guía en mi vida, a mis Padres, hermanos y sobrinos que con sus consejos, amor, comprensión han sabido ganarse mi cariño, respeto y admiración, a mis amigos lojanos que siempre estuvieron para apoyarme en los buenos y malos momentos.

De manera especial a dos seres queridos que en este momento no están físicamente conmigo, Mi madre †María Sarango y mi hermana †Elsa Maria Soto Sarango, que fueron los pilares fundamentales para que lleve a feliz culminación mi carrera.

Angel Fernando Soto Sarango

Dios por el camino recorrido, a mi familia por su apoyo durante mis años de estudio

Fernando Paúl Espinosa Peñaherrera.

TRIBUNAL DE GRADO

Ing. Holger Cevallos
SUBDECANO DE LA FIEC

Francisco Novillo P.
DIRECTOR DEL TOPICO

Ing. Washington Medina M.
VOCAL PRINCIPAL

Dr. Boris Ramos S.
VOCAL PRINCIPAL

DECLARACION EXPRESA

"La responsabilidad del contenido de esta Tesis de Grado, nos corresponde exclusivamente: y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL."

---------- ----------

Fernando Paúl Espinosa Peñaherrera Angel Fernando Soto Sarango

RESUMEN

El objetivo del presente proyecto es ofrecer un mecanismo de pago electrónico a las personas desde cualquier lugar geográfico con cobertura celular.

El presente proyecto sea dividido en cinco capítulos, en el primer capítulo se habla de los antecedentes del negocio, posteriormente hablamos sobre la situación actual del negocio, la forma de procesar las transacciones, proseguimos con la posible tecnología a utilizar para resolver nuestro problema planteado, continuamos con nuestras limitaciones operacionales que tendremos de nuestro proyecto, las justificaciones para el cambio de la forma de realizar las transacciones, la lógica del negocio, y finalmente planteamos nuestros objetivos para el pretendemos alcanzar con nuestro proyecto.

En el capítulo dos, se describe toda la investigación teórica del proyecto iniciando con las descripciones de J2ME, sus principales ventajas y características para trabajar en dispositivos con poca capacidad de procesamiento, seguidamente se habla de las características de GPRS, mencionado que nuestra comunicación entre el móvil y el servidor de control

será a través de GPRS, continuamos con mencionando características y ventajas de utilizar el protocolo HTTPS, continuamos con el protocolo TCP/IP, para finalizar hablando sobre las tarjetas de crédito, emisores de las mismas, tipos de procesadores en el mercado.

En el capítulo tres, se describe al proyecto como se llegó a la solución del mismo, inicialmente con la solución que planteamos, luego con la descripción del software que desarrollamos para resolver el mismo, describiendo el software desarrollado en el dispositivo móvil o aplicación cliente, posteriormente se describe la aplicación Servidor y la Aplicación Switch, finalmente continuamos con las seguridades de nuestra aplicación, para ello mencionamos los tipos de estándares que se utilizaron en el desarrollo de nuestras aplicaciones, la criptografía, y finalmente el tipo de seguridad que se utiliza en cada una de las aplicaciones que se desarrollo como son la aplicación Cliente, la aplicación Servidor, las seguridades en la comunicación del móvil con el servidor de control, posteriormente la seguridad entre el servidor de control y el Sistema de Switch de Autorizaciones.

En el capitulo cuatro, se describe como se implementó las aplicaciones, iniciando con los Módulos de Aplicación, siguiendo con los Modelos de

Clases, los Modelos de Datos, y finalmente con los resultados obtenidos en el desarrollo de nuestra aplicación.

En el capítulo final realizamos un plan de negocios, para ello iniciamos con el análisis de mercado, en donde realizamos encuestas a nuestros posibles clientes, seguidamente se realizo un análisis FODA (Fortalezas, Oportunidades, Debilidades y Oportunidades), seguidamente se segmento nuestro mercado para obtener un mercado potencial donde será incluido nuestro proyecto o puesto en funcionamiento nuestra aplicación, seguidamente continuamos con la descripción de los costos de diseño e implementación, recursos necesarios para iniciar nuestro proyecto, y finalmente con un análisis financiero a tres años de nuestro proyecto, en el cual a través del VAN podemos ver que es muy beneficio la realización del mismo al obtener 8,63 de VAN por cada dólar invertido.

INDICE GENERAL

INDICE GENERAL_____I

INDICE DE FIGURAS_____ III

INDICE DE TABLAS_____V

INTRODUCCION_____**1**

CAPITULO I

Antecedentes _____3

1.1. Situación Actual _____ 4

1.2. Tecnología a Utilizar _____7

1.3. Limitaciones Operacionales _____8

1.4. Justificación para los Cambios _____ 10

1.5. Lógica del Negocio _____ 11

1.6. Objetivo General _____15

1.7. Objetivos Específicos _____ 16

1.8. Conclusiones _____19

CAPITULO II

Teoría _____20

2.1. Características Principales de Java 2 Micro Edition (J2ME) _ 27

2.2. Características Principales de General Packet Radio Service 29

2.3. Hiper Text Transfer Protocol Secure (HTTPS) _____33

2.4. Protocolo de Comunicación TCP/IP _____34

2.5. Las Tarjetas de Crédito _____37

CAPITULO III

Descripción del Proyecto _____40

3.1. Solución Planteada _____ 41

3.2. Descripción del Software _____ 44

3.2.1. Descripción de la Aplicación J2ME _____ 45

3.2.2. Descripción del Switch_____ 51

3.3. Seguridad

3.3.1 Estándares de Seguridad _____ 58

3.3.2 Criptografía _____ 59

3.3.3 Descripción de la Arquitectura de Seguridad _____ 61

CAPITULO 4

Implementación del Proyecto _____69

4.1. Módulos de Aplicación _____ 70

4.2. Modelo de Clases _____
73

4.3. Modelo de Datos _____ 88

4.4 Resultados Obtenidos _____ 89

CAPITULO 5

Plan de Negocios _____107

5.1. Análisis de Mercado _____108

5.2. Competidores _____110

5.3. Mercado Potencial _____ 113

5.4. Costo de Diseño _____115

5.5. Costos de Implementación _____116

5.6. Recursos Necesarios _____ 118

5.7. Análisis Financiero a 3 años _____119

CONCLUSIONES Y RECOMENDACIONES

ANEXOS

BIBLIOGRAFIA

INDICE DE FIGURAS

Pag.

Figura 1.1. Tecnologías a Utilizar en Nuestro Proyecto ……..8

Figura 1.2. Diagrama de la Lógica del Negocio ……..15

Figura 2.1. Campo de texto para el ingreso de un número de tarjeta de crédito ……23

Figura 2.2. Aquitectura TCP/IP ……..35

Figura 2.3. Encapsulación de Datos ...…….36

Figura 2.4. Estructura de Datos …….36

Figura 3.1. Diagrama de la Arquitectura del Sistema ……42

Figura 3.2. Pantalla de configuración de la aplicación cliente …….46

Figura 3.3. Secuencia de pantallas para realizar una transacción con la aplicación cliente ……..49

Figura 3.4. Diagrama de flujo de autenticación de aplicaciones cliente ….68

Figura 4.1. Mòdulos de Aplicación…….. ….71

Figura 4.2. Diagrama de clases de la aplicación cliente. Parte 1 ……76

Figura 4.3. Diagrama de clases de la aplicación cliente. Parte 2 ……77

Figura 4.4. Diagrama de clases de la aplicación Servidor ……. ……..80

Figura 4.5. Diagrama de Secuencia para el caso de uso 2..…… ……..87

Figura 4.6. Diagrama entidad-relación de la base de datos del servidor de control…….88

Figura 4.7. Formulario de Solicitud de Autorización de Cobro …….95

Figura 4.8. Respuestas de Transacción Exitosa ……95

Figura 4.9. Contenido de la ventana de Log del emulador del Nokia SDK para el caso de una petición de autorización ……..96

Figura 4.10. Datos de autorización exitosa de cobro en la aplicación Local.exe ……...97

Pag.

Figura 4.11. Realización de una autorización de cobro en el sistema Emisor.exe ……...98

Figura 4.12. Mensaje de datos ingresados no válidos ……99

Figura 4.13. Baja de un usuario del servicio en el administrador de la base de datos……100

Figura 4.14. Mensaje indicando que los datos no pasaron el proceso de autenticación en el servidor ……101

Figura 4.15. Aplicación Local.exe. Número de tarjeta para el que se solicita autorización no corresponde al rango de bines del emisor …102

Figura 4.16. Mensaje mostrado al usuario en caso de que el número de tarjeta para el que se solicita autorización no corresponde al rango de bines del emisor ……102

Figura 5.1. Mercado Potencial: Parque de los Jipis (Cuenca)……114

Figura 5.2. Cotización del Servidor Dedicado para el Primer Año……117

Figura 5.3. Cotización de Servidores Dedicado para el segundo y tercer año……118

INDICE DE TABLAS

Pag.

Tabla 2.1. Cuadro comparativo del análisis de J2ME, WAP y SMS como alternativas para la implementación de la solución propuesta ……27

Tabla 2.2. Tarjetas de Crédito y Bancos que la Poseen ……..38

Tabla 2.3. Procesadores de Tarjetas de Crédito ……...39

Tabla 5.1. Fortalezas y Debilidades de Nuestra Empresa ……112

Tabla 5.2. Oportunidades y Amenazas de Agentes Externos ..……112

Tabla 5.3. Costos de Diseño ..……116

Tabla 5.4. Análisis Financiero a tres años del Proyecto ……120

Tabla 5.5. Costos y Beneficios por Transacción ..……121

Tabla 5.6. Calculo del TIR y VAN de Nuestro Proyecto ……122

INTRODUCCIÓN

Muchas cosas han cambiado en el mundo, especialmente con la tecnología de las telecomunicaciones y en las empresas en los últimos años.

El mundo se ha globalizado, la competencia está en todos lados, estos nuevos desafíos han llevado a una transformación profunda de la manera de realizar las transacciones comerciales. La aparición de nuevas tecnologías (nuevas arquitectura y dispositivos) y la consolidación de otras anteriores (por ejemplo las redes IP) ha facilitando la evolución natural desde la voz hacia los datos, hecho que se conoce mejor con el término de multimedia. Esto ha permitido la creación de medios más potentes y novedosos y de nuevos canales de relación entre personas o entre personas y sistemas.

En vista de la evolución de las comunicaciones surgió la idea de crear un sistema de pago a través del teléfono móvil, el cual solicita la autorización para realizar el cobro a la tarjeta de crédito, mediante un software instalado en el teléfono móvil.

Nuestro objetivo es ofrecer un mecanismo de pago electrónico a las empresas y personas desde cualquier lugar geográfico con cobertura celular y así extender el uso de la infraestructura de pagos electrónicos en los negocios de gran y menor tamaño (PYMES), tales como: comercio al menudeo, tiendas de abarrotes, locales de comida, islas, papelerías, farmacias y hasta taxis.

CAPITULO I

1. ANTECEDENTES

Hoy en día los teléfonos celulares constituyen un recurso al alcance de prácticamente toda persona en nuestro país, existen 10.981.455 abonados que corresponde aproximadamente al 84,48% de la población [1]. El uso masivo de este dispositivo ha permitido que las operadoras de telefonía celular expongan sus servicios para tener la posibilidad de ofrecer una variedad de soluciones que permitan satisfacer las necesidades de los usuarios. Es por esto que en el mercado podemos encontrar una gran variedad de modelos cada vez con mejores capacidades de procesamiento.

Toda la tecnología aplicada en estos dispositivos permite ampliar aun más el campo de servicios que ofrece la telefonía celular.

Al igual que los teléfonos celulares, el uso de las tarjetas de crédito se incrementa cada año [2] y la facilidad de obtenerla en muchos países motiva la idea de tener un sistema de pago a través de un teléfono celular.

El objetivo principal de nuestro trabajo es ofrecer un mecanismo de pago electrónico a empresas y personas que desde cualquier lugar geográfico con cobertura celular permita extender el uso de la infraestructura de pagos electrónicos a los negocios de gran y menor tamaño.

1.1 Situación Actual.

Al investigar la situación actual de los pequeños y medianos negocios (PYMES), muchos de estos o la gran mayoría no disponen de la infraestructura de comunicación a través de una red de telefonía convencional, o de un medio para realizar cobros con tarjeta de crédito. Uno de los principales problemas a los que se enfrentan las empresas

para proceder con una transacción de pago es la obtención de la autorización necesaria para el inicio de su operación.

Si consideramos el uso de plataformas de pago electrónico que en la actualidad existen, podemos mencionar los portales de pago electrónico utilizados para transacciones comerciales o mobile commerce (m-commerce)[1][3] a través del Internet, entre los más populares tenemos PayPal [4], ALIGNET [5].

Estas soluciones resultan muy útiles para los modelos de negocios en Internet, si consideramos por el contrario negocios más pequeños en los cuales no se disponen de los medios económicos para sustentar la implementación y mantenimiento de una solución de este tipo, surge la idea de proveer de un mecanismo muy similar pero menos costoso utilizando dispositivos móviles, de esta forma podemos aprovechar la infraestructura que ofrecen las operadoras de telefonía móvil, para

[1] m-commerce (mobile commerce) el comercio electrónico móvil. Esto es, la posibilidad de realizar transacciones comerciales a través de un dispositivo móvil. En este proceso de comercio móvil están o pueden estar incluidos todos los pasos de una transacción comercial. 2001, Havet Interactive S.A., GPRS: La Nueva Generación de telefonía móvil

brindar a los pequeños y medianos negocios un servicio adicional para el cobro por sus ventas.

Si consideramos además que hoy en día la mayoría de las personas disponen de un teléfono celular, gracias a la constante expansión en la cobertura de las redes de telefonía móvil y al marketing que basado en sus mejores capacidades y funcionalidades se le ha dado a estos dispositivos, convierte a estos aparatos en un vehículo muy atractivo como sistema de pago electrónico [6].

Esto permite plantearnos el diseño de un sistema de pago electrónico a través de teléfonos móviles, el cual establecerá una conexión con la red de entidades financieras y solicitará autorización al emisor de la tarjeta de crédito del cliente para realizar el cobro a través de la misma mediante el desarrollo de una aplicación instalada en el teléfono celular.

Nuestro objetivo principal por lo tanto es ofrecer un mecanismo de pago electrónico a las empresas y personas desde cualquier lugar geográfico

con cobertura celular que permita extender el uso de la infraestructura de pago electrónico en los PYMES tales como: comercio al menudeo, tiendas de abarrotes, locales de comida, islas, papelerías, farmacias, taxis, entre otros.

1.2. Tecnología a utilizar

Para la realización de este proyecto de tesis se ha efectuado un análisis de las principales tecnologías y aspectos técnicos involucrados en su desarrollo como son: lenguajes de programación, mecanismos de seguridad, protocolos de comunicación entre otros y sobre todo tomando muy en cuenta las Normas de Seguridad de Datos de la Industria de Tarjetas de Pago (PCI) [7]. Estos aspectos poseen las características que nos permitirán solucionar la problemática planteada en el enunciado anterior.

Las siguientes son las tecnologías disponibles que hemos calificado las más indicadas para ser aprovechadas en el desarrollo de nuestra solución, las mismas que en el capítulo siguiente serán descritas con mayor profundidad y daremos nuestras razones para usarlas, en la figura 1.1 presentamos las tecnologías a utilizar en nuestro proyecto:

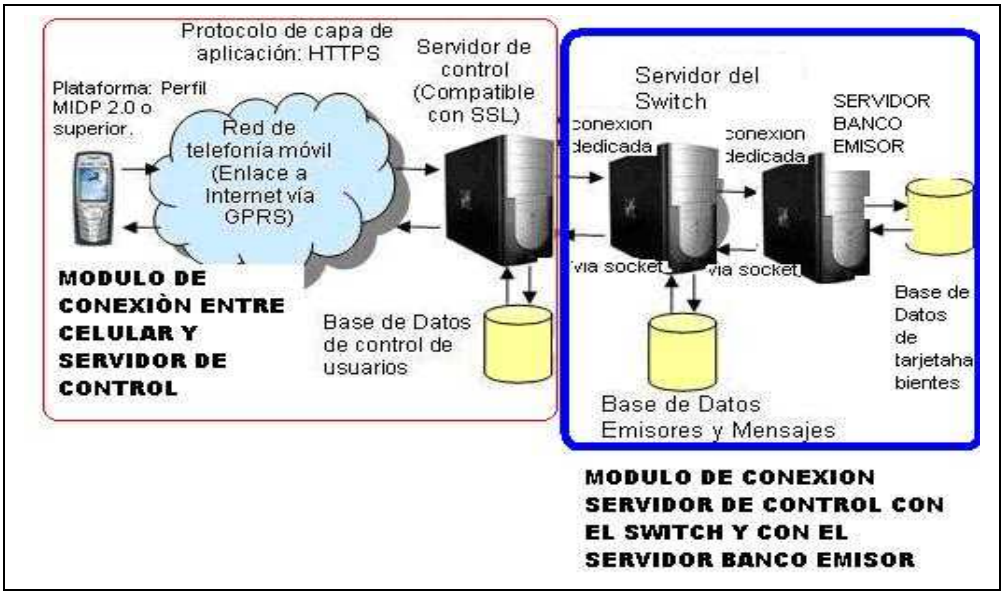


Figura 1.1 Tecnologías a Utilizar en Nuestro Proyecto

1.3. Limitaciones operacionales

En el presente subcapítulo hacemos referencia a las posibles barreras que se podrían presentar previa la implementación de un sistema de esta índole.

Una de las principales es la negativa de las entidades financieras, pasarelas de pago y demás componentes principalmente los switch[2] de autorizaciones [8] o Procesadores de Tarjetas de Crédito[3] para transacciones electrónicas que tienen un dominio del mercado, las mismas que no brindan ninguna información técnica del proceso, información que sería muy útil para el desarrollo de la solución propuesta, siendo este último nuestro principal limitación operacional.

Puede resultar desventajoso para nuestra propuesta que frente a los medios tradicionales de cobro con tarjeta de crédito los usuarios encuentren dificultades al adaptarse al nuevo sistema, que es básicamente la reacción de los usuarios al cambio o al uso de nueva tecnología (el proceso de cobro a través del teléfono móvil) o simplemente puede ser que los clientes de los establecimientos asociados a nuestra aplicación no confíen en la seguridad de las transacciones por este medio.

[2] Software de Autorizaciones de Tarjetas de Crédito en Línea, utilizando la ISO 8583, SuSistema Cia. Ltda., 2002, SWITCH DE AUTORIZACIONES

[3] Los Principales Procesadores o Emisores de Tarjetas de Crédito son: DATAFAST, MEDIANET, UNIBANCO, AUSTRO Y SERVITRANSEL.

1.4. Justificación para los cambios

Una vez realizado un breve análisis de la situación actual de las posibles transacciones electrónicas que pueden realizarse en nuestro país y considerando nuestra alternativa como la solución que permita incorporar a más pequeños y medianos negocios en el Ecuador, el Tópico de Desarrollo de Aplicaciones para Teléfonos Móviles y el Grupo de Investigación GICOM de la Escuela Superior Politécnica del Litoral (ESPOL), a través del presente proyecto “Pago Electrónico a través del Teléfono Móvil”, pretender ofrecer a los PYMES una solución que permita incrementar sus ventas, agregando a su grupo de clientes a un mercado de 1.691.885 millones de tarjeta habientes[9] en nuestro mercado local.

Entre los principales beneficios que podrá obtenerse del desarrollo de la solución podemos mencionar los siguientes:

- Realizar transacciones en cualquier momento y lugar incluso en lugares a donde no lleguen los medios de comunicación tradicionales o no se disponga de la infraestructura necesaria para la aceptación de pagos con tarjeta de crédito. Esto beneficiaría

enormemente a los comerciantes contribuyendo especialmente al desarrollo de la pequeña y mediana empresa.

- El incremento en las ganancias de los comerciantes, al ofrecer otra posibilidad de pago lo que a su vez se refleja en la prestación de un mejor servicio.

- La simplicidad en el proceso de toma de la información sobre la transacción.

1.5. Lógica del negocio

El objetivo de describir la lógica del negocio para nuestro caso, es ofrecer un modelo que permita entender de manera general el funcionamiento de esta solución una vez que esté implementada.

Para el caso de la solución propuesta, el procedimiento a llevar a cabo consistirá en efectuar pagos por el celular mediante una aplicación que permitirá al usuario para establecer un enlace con nuestro servidor Web,

este a su vez se conectará a otro servidor que hará la conexión con el banco emisor.

El usuario, contraseña y número identificador del equipo son los datos que se utilizan para la autenticación en la base de datos a la que se conecta la aplicación servidor.

Para minimizar los riesgos en caso de que un equipo se extravíe o sea robado, cada usuario del sistema poseerá un usuario, contraseña y número de identificador del equipo, en la base de datos a la que se conecta la aplicación servidor. Por razones principalmente de seguridad se ha decidido que estos datos de autenticación no estén al alcance del comerciante que utilizará la aplicación, si no que se mantengan almacenados internamente en el equipo celular. Además, los celulares no guardarán información alguna respecto a las transacciones realizadas, debemos dejar claro que el teléfono celular solo será utilizado como una herramienta que permita realizar una conexión con los procesadores de tarjetas de crédito y entidades bancarias, estos son los que tienen un sinnúmero de seguridades y sistemas anti fraudé, los mismos que detallaremos en el capítulo tres.

El comerciante que desee realizar un cobro, deberá ingresar los datos de la transacción junto con el detalle de la tarjeta en un formulario que le proveerá la aplicación instalada en el teléfono móvil. Los datos de cada transacción junto con los datos de usuario del comerciante deberán viajar cifrados hasta el servidor de control. Esta tarea la realizará la misma aplicación instalada en el teléfono móvil de tal manera que se impida posibles fraudes.

El servidor de control contendrá los detalles de las cuentas de vendedores y por tanto se encargará de realizar su respectiva verificación. De aquí en adelante el proceso a seguir es similar al proceso que se sigue al efectuar transacciones electrónicas con tarjeta de crédito en un sitio Web, es decir el servidor de control envía la información de la transacción a una entidad de procesamiento de pagos vía Internet que es el equivalente a un Terminal de punto de venta con tarjetas de crédito en un almacén, está a su vez envía la información de la transacción, vía conexión segura a la entidad verificadora de tarjetas de crédito del banco del vendedor, la cual reenvía la información de la transacción a una red de procesamiento de transacciones en donde una entidad financiera autorizada encamina los datos al respectivo banco emisor (el banco que le emitió la tarjeta de crédito al cliente) para

autorización. La respuesta será recibida en el servidor de control, el cual enviará al usuario solicitante un mensaje de aceptación o rechazo del petitorio realizado.

Para visualizar mejor la interacción de los procesos se realizó el diagrama de la lógica del negocio como podemos observar en la figura 1.2.

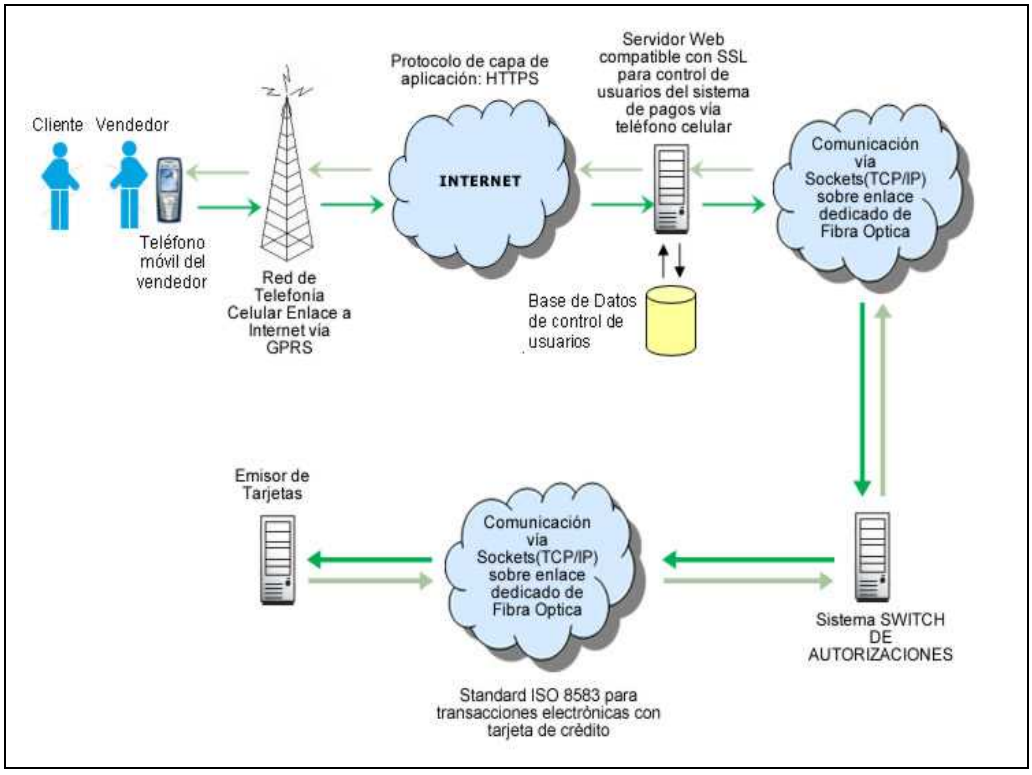


Figura 1.2 Diagrama de la lógica del negocio

1.6. Objetivo General

El objetivo principal que aspiramos en el desarrollo de la presente tesis es ofrecer un mecanismo alternativo de pago electrónico con débito a la tarjetas de crédito de un cliente a través de teléfonos móviles (teléfonos móviles que se encuentren dentro del área de cobertura de las operadoras de telefonía), enfocando la solución especialmente en los pequeños y medianos negocios.

1.7 Objetivos Específicos

Reducción de Costos

Por el lado de los comerciantes, este proyecto pretende minimizar significativamente los costos de transacciones aprovechando el hecho de que las comisiones por utilizar el teléfono celular como un Terminal de Punto de Venta serán mucho más reducidas. Sumado a esto está el hecho de que con esta alternativa los comerciantes no deberán disponer de equipos o de ningún infraestructura para efectuar cobros con tarjeta de crédito si no que solamente bastará que dispongan de un teléfono móvil. Esto resultará un gran atractivo para los negocios interesados en aceptar

pago electrónico en transacciones, sin que les represente un gasto exagerado.

Aprovechamiento del uso masivo de los teléfonos celulares

Hoy en día la telefonía móvil se ha vuelto ampliamente popular entre la población, principalmente debido a que los costos cada vez más reducidos de los equipos, los cuales presentan cada vez mejores características y funcionalidades hacen a estos alcanzables para un número cada vez mayor de usuarios. Sumado a esto tenemos el hecho de la constante expansión de las redes de telefonía móvil, las cuales llegan cada vez a más lugares y por lo tanto a una mayor parte de la población. Estos motivos le representan una gran ventaja a los teléfonos celulares como medio de comercio electrónico con respecto a los medios tradicionales de cobro con tarjetas de crédito. Por lo tanto aprovechar esta ventaja es uno de los objetivos a alcanzar con el desarrollo de esta solución.

Fomentar el crecimiento de los pequeños y medianos negocios

Otro de los importantes objetivos que se pretende alcanzar con este proyecto y que representará grandes beneficios para la sociedad y el

sistema financiero en general es el de ayudar al crecimiento de la pequeña y mediana empresa, puesto que al agregar a la arraigada cultura del uso de dinero en efectivo para el comercio un medio alternativo de comercio electrónico, los pequeños y medianos empresarios podrían ver incrementado su número de potenciales clientes, lo cual se traduciría en mejores utilidades para los mismos.

Facilitar las transacciones a compradores y vendedores

Este es uno de los principales objetivos a alcanzar, debido a que pretendemos que esta herramienta facilite la realización de transacciones con tarjeta de crédito tanto para consumidores como para comerciantes, ya que se trata de un medio de fácil uso y que además puede ser empleado en cualquier lugar donde exista cobertura de telefonía celular.

Incentivar la inversión en el desarrollo de este tipo de soluciones

Lograr que las instituciones involucradas con pagos inviertan en investigación y desarrollo de este tipo de soluciones es otro de los objetivos que se pretende alcanzar, ya que hoy en día se debe aprovechar al máximo el crecimiento de usuarios de telefonía móvil como se expuso en uno de los anteriores objetivos y de este modo lograr que

en nuestro país se brinden servicios cada vez más variados y de mejor calidad para estar a la par con los países desarrollados, en un mundo globalizado.

1.8 Conclusiones

La presente solución otorga a los teléfonos celulares la funcionalidad de potenciales terminales de punto de venta, con lo cual estos ya no solo serán usados para comunicación y entretenimiento solamente si no que se convertirán también en poderosas herramientas de comercio electrónico, extendiendo el uso de la infraestructura de pagos con tarjeta de crédito en negocios tales como: comercio al menudeo, tiendas de abarrotes, locales de comida, islas, papelerías, farmacias, taxis entre otros.

CAPITULO II

2. TEORIA

En el presente capítulo describiremos de manera detallada las principales características referentes a los aspectos tecnológicos que consideramos primordiales para cumplir con los objetivos propuestos en nuestro proyecto, los mismos que hemos seleccionado para la implementación de nuestra solución y que fueron mencionados en el capítulo anterior, como se muestra en la figura 1.1 del Capítulo 1.

Inicialmente detallaremos las principales características de Java 2 Micro Edition (J2ME), la herramienta de desarrollo seleccionada para la creación de la interfaz de usuario, la cual se instalará en el teléfono móvil y guiará al usuario para la introducción de los datos de una transacción de cobro con tarjeta de crédito y establecimiento de una conexión con un servidor de control.

Posteriormente detallaremos las principales características de la tecnología General Packet Radio Service (GPRS), características que a nuestro criterio hacen de esta tecnología de comunicación la más adecuada para nuestros fines y que por tanto ha sido seleccionada para la realización del intercambio de datos de transacciones y mensajes entre el teléfono móvil y el servidor de control.

Como plataforma de desarrollo utilizaremos J2ME por las prestaciones que presenta a la hora de programar aplicaciones para dispositivos móviles.

A continuación vamos a explicar porque J2ME fue considerada la primera opción para el desarrollo de nuestra solución, habiendo estudiado además las alternativas de utilizar Short Message System (SMS) [10] o Wireless Application Protocol (WAP) [10].

- Java se ha convertido en una de las tecnologías más reconocidas y utilizadas para desarrollo de software de redes, desde su surgimiento en el año de 1995, además la configuración Connected Limited Device Configuration (CLDC) 1.0 [11] ha tenido una masiva acogida desde su lanzamiento en el año 2000, lo que ha resultado en que J2ME se convierta en una atractiva tecnología para el desarrollo de aplicaciones móviles.

- J2ME brinda la posibilidad de crear interfaces gráficas que permitirán una mejor experiencia del usuario en el uso de la aplicación, característica de la que no se dispone por ejemplo en el caso de usar un SMS para el envío de la información de una transacción

- En contraste con el uso de SMS o WAP, mediante el uso de un Midlet [11] se puede ser mucho más específico con el usuario en cuanto al tipo de información que de este se requiere. Por

ejemplo, mediante la propiedad Textfield.NUMERIC de un control Textfield [11] se puede lograr la restricción de que solamente dígitos puedan ser ingresados en el control. De este modo no será necesario cambiar el modo de escritura en el teléfono celular o presionar varias veces una tecla para obtener un determinado carácter. Esto resulta muy útil por ejemplo en el caso del ingreso de un número de tarjeta de crédito por parte del usuario, como se indica en la Figura 2.1.

Figura 2.1 Campo de texto para el ingreso de un número de tarjeta de crédito.

- Además de contar con un entorno de ejecución propio, J2ME mediante su sistema Record Management System (RMS) [11], nos provee de un medio para almacenamiento de información en el equipo, característica que veremos más adelante es un requisito

indispensable para el caso de nuestra solución y de la cual no se dispone en el caso de una aplicación WAP en donde en ocasiones solo se dispone de una capacidad limitada de almacenamiento a través de cookies.

- Existen ciertos aspectos de dependencia en cuanto al Hardware, los cuales varían de acuerdo a los modelos y fabricantes de equipos, como por ejemplo: los diferentes tamaños de las pantallas. Esto podría afectar al desarrollo de ciertos tipos de aplicaciones como en el caso de juegos, pero no resulta muy relevante en el caso del desarrollo de aplicaciones de pagos móviles ya que para el desarrollo de este tipo de aplicaciones el requerimiento mínimo es disponer de los componentes estándar para interacción con el usuario. En J2ME la API [10] de interface de usuario de alto nivel a través de la clase Screen [11] provee de las interfaces comunes o estándar para la interacción con el usuario, como son: formularios, cuadros de texto, mensajes de alerta, listas, etc.

- A diferencia de WAP en donde los contenidos se despliegan en un micro navegador que se encargará de interpretarlos, en las aplicaciones desarrolladas con J2ME no se dispone de esta

capacidad, sin embargo, a través de las diferentes clases heredadas de Java se tiene la facilidad de analizar e interpretar cualquier tipo de contenido como puede ser el caso de documentos en formato Hyper Text Markup Language (HTML) o Extensible Markup Language (XML) [12].

- Debido a las capacidades de trabajar en un entorno multihilo que posee Java, en un Midlet se puede realizar la interacción con el usuario sin inconvenientes inclusive mientras algunos otros procesos importantes estén siendo ejecutados.

La tabla 2.1 resume los aspectos del análisis comparativo de las 3 tecnologías como alternativas para el desarrollo del proyecto que resultaron los justificativos principales para la elección de J2ME como plataforma de desarrollo en el caso de nuestro proyecto

	J2ME	WAP/WML	SMS

Costos de uso de la red	Solo se utiliza la red cuando es necesario lo cual se traduce en bajos costos.	Al residir por completo la lógica del negocio en el servidor, se incrementa el uso de la red.	Costo de envío de mensajes de texto son mayores al costo de envío de datos vía HTTP.
Capacidad de almacenamiento	Provee de un sistema de almacenamiento persistente especialmente destinado para este fin.	Muy limitada, mediante el uso ocasional de cookies en algunos navegadores.	No se dispone de esta característica.

interface gráfica de usuario	Se dispone de la API de interface de usuario de alto nivel que provee de los controles comunes o estándar para la interacción con el usuario	Limitada en cuanto se refiere a validaciones y manejo de eventos	No se cuenta con la posibilidad de brindar esta característica

Tabla 2.1 Cuadro comparativo del análisis de J2ME, WAP y SMS como alternativas para la implementación de la solución propuesta

2.1.Características principales de Java 2 Micro Edition (J2ME)

La plataforma de Desarrollo Java 2, Micro Edition (J2ME), fue pensado para trabajar sobre equipos con ciertas limitaciones: en procesamiento, pantalla, energía basada en baterías y de alguna forma poseen servicios de red para comunicarse con el exterior.

Java 2 Plataform, Micro Edition (J2ME): Esta versión del lenguaje Java está enfocada a la aplicación de la tecnología Java en dispositivos electrónicos con capacidades computacionales y gráficas muy reducidas, tales como teléfonos móviles, Personal Digital Asistente (PDAs) o electrodomésticos inteligentes. Esta edición tiene unos componentes básicos que la diferencian de las otras versiones, como el uso de una máquina virtual denominada Kilo Virtual Machina (KVM), debido a que requiere sólo unos pocos Kilobytes de memoria para funcionar en vez del uso de la Java Virtual Machina clásica[11].

Dentro de las principales características tenemos:

- Es un lenguaje totalmente orientado a objetos.

- La portabilidad y compatibilidad entre plataformas.

- Una aplicación desarrollada en J2ME podrá ser ejecutada en cualquier equipo celular o PDA que tenga una Máquina Virtual Java instalada y, en esta categoría encontramos la mayoría de los dispositivos móviles que se ofrecen en el mercado actualmente

- Es la integración transparente con otras Tecnologías JAVA.

- La posibilidad de ejecutar aplicaciones altamente dinámicas en el dispositivo inalámbrico, en este sentido, es posible ejecutar, guardar programas altamente gráficos, video a través de la conexión en Internet, caso no posible con Wireless Application Protocol (WAP)/ Wireless Markup Language (WML) [10].

- La interface Gráfica en general se ve ampliamente superada a diferencia de aplicaciones WAP/WML.

Para poder tener un entorno de ejecución JAVA para J2ME que cumpla los requisitos de un rango amplio de dispositivos y mercados objetos es necesario que se disponga de Configuración, Perfiles y Paquetes opcionales. J2ME se basa en los conceptos de configuración y perfil.

2.2. **Características principales de General Packet Radio Service(GSM)**

La tecnología GPRS, o generación 2.5, representa un paso hacia los sistemas inalámbricos de Tercera Generación o Universal Mobile Telecommunications System (UMTS). Su principal características

radica en la posibilidad de disponer de un terminal permanentemente conectado, tarifando únicamente por el volumen de datos transferidos (enviados y recibidos) y no por el tiempo de conexión. Proporciona altas velocidades de transferencias de datos (especialmente útil para conectar a Internet) y se utiliza en las redes GSM y Code Division Múltiple Access (CDMA)[13] .

La tecnología GPRS permite proporcionar servicios de datos de una forma más eficiente a como se venía haciendo hasta el momento especialmente consiste en modificar la forma de transmitir datos, pasando de la conmutación de circuitos en GSM (donde el circuito está permanentemente reservado mientras dure la comunicación aunque no se envíe información en un momento dado) a la conmutación de paquetes [14].

Algunas de las características principales de GPRS son [13]:

- Velocidad de transferencia hasta 144kbps (170kps teórico máximo).

- Pago por volumen de datos transmitidos y no por tiempo de conexión.

- Mejora sustancialmente el sistema de mensajería, permitiendo Multimedia Messaging System (MMS) con mensajes de voz, texto, imágenes y video.

- Acceso GPRS a aplicaciones Wireless Application Protocol (WAP)

- GPRS es básicamente una comunicación basada en paquetes de datos.

- Conexión permanente.

- Cada elemento de la red sabe como encaminar cada paquete.

- Cuatro niveles de codificación radio.

Los Teléfonos celulares existentes en el mercado con soporte para GPRS presentan las siguientes características comunes [14]:

- **Capacidad Dual:** Están adaptados para aprovechar la cobertura existente tanto en GSM para la voz como en GPRS para los datos.

- **Velocidad de transferencia:** Se utiliza varios canales simultáneos. El número de canales depende de cada equipo, variando de 1 a 4 para la recepción de datos y de 1 a 2 para el envío.

- **Tarjeta Subscriber Identity Module (SIM):** Utilizan la tarjeta SIM GSM.

El enlace que utilizamos entre el móvil y el servidor de control será a través de GPRS que por lo antes expuesto resulta ser la mejor opción. Dentro de los servicios soportados tenemos: World Wide Web (WWW), File Transfer Protocol (FTP), Telnet, Chat, E-mail, Imagen, Audio, Video[14].

Esta tecnología consiste en otra forma de transmisión de datos en donde se pasa de la conmutación de circuitos comúnmente utilizada en GSM (En donde el circuito está permanentemente reservado mientras dure la comunicación) a la conmutación de paquetes.

2.3.Hyper Text Transfer Protocol Secure (HTTPS).

El protocolo HTTPS la versión segura del protocolo HTTP. HTTPS utiliza un cifrado basado en el protocolo Secure Socket Layers (SSL). Este es un protocolo que se sitúa entre el protocolo de capa de red (Ej.: TCP) y el protocolo de la capa de aplicación (Ej.: HTTP). SSL proporciona mecanismos para establecer una comunicación fiable entre un cliente y un servidor, por medio de mecanismos como: autenticación, uso de firmas digitales para validar integridad, certificados y uso de encriptación para privacidad [16].

La encriptación consiste en “transformar un mensaje inteligible” en otro que no lo sea en absoluto, para después devolverlo a su forma original, sin que nadie que vea el mensaje cifrado sea capaz de entenderlo [17].

SSL utiliza el esquema de encriptación de clave publica/privada, el cual utiliza 2 claves, si una clave se utiliza para encriptar el mensaje, la otra servirá para desencriptarlo, como el nombre lo indica, la clave publica es una clave cuyo acceso es permitido a cualquier persona, de esta manera es posible recibir mensajes encriptados de personas que tengan acceso a nuestra clave pública, siendo nosotros los únicos que podemos leerla con nuestra clave privada [16].

El mecanismo en SSL que permite garantizar la integridad del mensaje se llama “de mensaje abreviado” y consiste en crear un pequeño resumen de el mensaje usando una función de hash, el receptor creará su propio mensaje abreviado y lo comparará con el recibido, si ambos coinciden significa que el mensaje fue recibido intacto.

Si cada parte tiene un certificado que valide la identidad del otro, confirme la llave pública y esté firmado por una agencia creíble (trusted agency), entonces ellos estarán seguros de que realmente se están comunicando con quienes ellos creen, este protocolo fue escogido por seguir con los PCI mencionados en nuestro capitulo anterior.

2.4. Protocolo de Comunicación TCP/IP.

Proviene del nombre de los dos protocolos que lo forman: Transmisión Control Protocol (TCP) y el Internet Protocol (IP). Es totalmente independiente del medio de transmisión físico, tiene un esquema de direccionamiento amplio y común [18].

Arquitectura de TCP/IP

La figura 2.2 muestra la arquitectura:

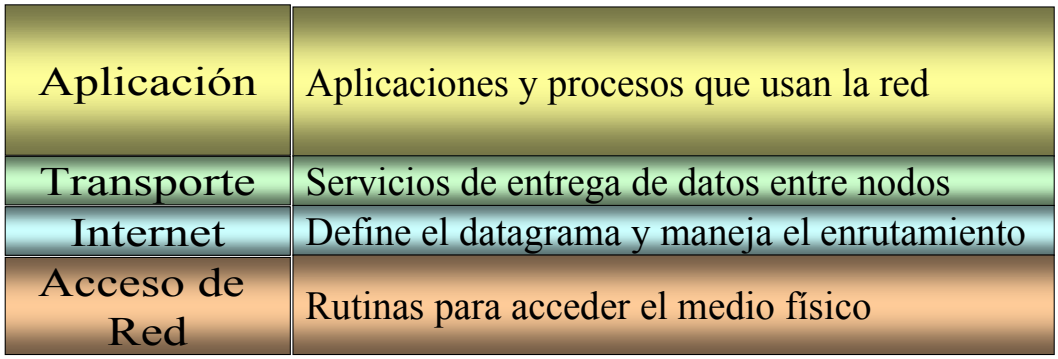


Figura 2.2 Arquitectura de TCP/IP [19]

Encapsulación de Datos

- Cada capa de la pila TCP/IP adiciona información de control para asegurar la entrega correcta de los datos.
- Cuando se recibe, la información de control se retira.
- En la figura 2.3 se puede observar la forma de cómo están encapsulados los datos.

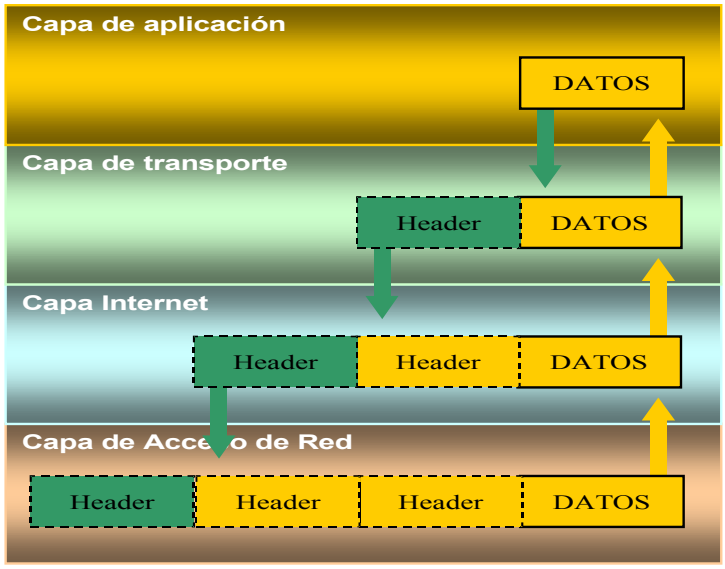


Figura 2.3 Encapsulación de Datos [19].

Estructura de Datos

- En la siguiente figura 2.4 podemos ver la descripción de la estructura de datos y las capas donde se desarrolla cada una de ellas.

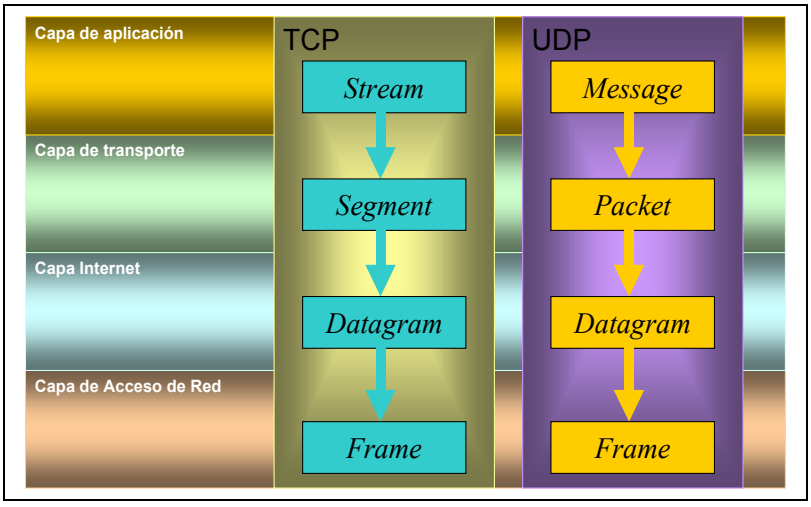


Figura 2.4 Estructura de Datos [19].

Protocolo Internet (IP)

El protocolo IP tiene a su cargo las siguientes funciones [18]:

- Define el datagrama, que es la unidad básica de transmisión en Internet.
- Define el esquema de direccionamiento de Internet. Mueve datos entre capa de acceso de red y la capa de transporte host to host

Características:

- Es un protocolo connectionless (no intercambia información de control – handshake, para establecer una conexión nodo a nodo antes de transmitir).
- No corrige ni detecta errores en la información.
- Otros protocolos hacen estas tareas.

2.5. Las Tarjetas de Crédito.

Primeramente vamos a hablar un poco de los emisores o procesadores de las tarjetas de crédito actualmente en nuestro país, existen diversas marcas de tarjetas de crédito con las que cuentan los diferentes bancos

para ello vamos a detallar en la presente tabla 2.2 la marca de tarjeta con su respectiva institución que la emiten [2]:

TARJETA DE CREDITO	INSTITUCION EMISORA
MASTERCARD	Banco Bolivariano, Banco Guayaquil, Banco del Austro, Banco Pacífico, Banco Pichincha, Banco Internacional, Banco Produbanco, Mutualista Pichincha y Pacificard
DINERS	Sociedad Financiera Diners Club
VISA	Banco Amazonas, Banco Bolivariano, Banco Comercial de Manabí, Banco Guayaquil, Banco de Loja, Banco de Machala, Banco del Austro, Banco Pacífico, Banco Pichincha, Banco Rumiñahui, Banco Guayaquil Bank Trust, Banco Internacional, Banco MM Jaramillo Arteaga, Banco Produbanco y Mutualista Azuay
AMERICAN EXPRESS	Banco Guayaquil
CUOTA FACIL	Banco Unibanco
TARJETA CREDITO SI	Banco Territorial
MI SOCIA	Banco Solidario
CREDITO SI	Banco Territorial
ROSE	Banco Internacional

Tabla 2.2 Tarjetas de Crédito y Bancos que la Poseen.

Ahora vamos a mencionar a los procesadores de las tarjetas de crédito y con qué bancos trabajan, en la tabla 2.3 siguiente los detallamos:

PROCESADOR DE TARJETAS	INSTITUCIONES
DATAFAST	Banco Guayaquil, Banco Pichincha, Banco Pacífico, Banco de Loja, Pacificard, Sociedad Financiera Diners Club, Banco Guayaquil Bank

	Trust, Banco Amazonas, Banco MM Jaramillo Arteaga
MEDIANET	Banco Bolivariano, Banco Produbanco, Banco Internacional
UNIBANCO	Banco Unibanco
AUSTRO	Banco del Austro
SERVITRANSEL	Cooperativas

Tabla 2.3 Procesadores de Tarjetas de Crédito.

La cobertura provincial del número de tarjetas de crédito lo podemos ver en el anexo 1, y también presentamos en el anexo 2 la población que poseen tarjetas de crédito por clase.

CAPÍTULO III

3. DESCRIPCION DEL PROYECTO

En el presente capítulo el objetivo es explicar de manera detallada el desarrollo de las aplicaciones que nos ayudaron a resolver la problemática planteada en el capítulo I, con las diversas tecnologías utilizadas para resolver el mismo.

Finalmente y por ser un factor crítico en el desarrollo de sistemas de este tipo, se le dedicará especial atención al tema de las seguridades las cuales serán mencionadas en la sección 3.3 de una manera detallada.

3.1. Solución Planteada.

Nuestra solución consiste en un sistema en el cual los comerciantes dispondrán de una aplicación instalada en un teléfono celular, que les permitirá realizar cobros con tarjeta de crédito a sus clientes mediante un enlace directo a un servidor Web el cual previa autenticación del comerciante en una base de datos, se enlazará a su vez con una entidad de procesamiento de transacciones con tarjeta de crédito (switch)[4][8] para solicitar la autorización del cobro.

En la figura 3.1 se muestra un diagrama que esquematiza la arquitectura del sistema en general.

[4] Software de Autorizaciones de Tarjetas de Crédito en Línea, utilizando la ISO 8583, SuSistema Cia. Ltda., 2002, SWITCH DE AUTORIZACIONES.

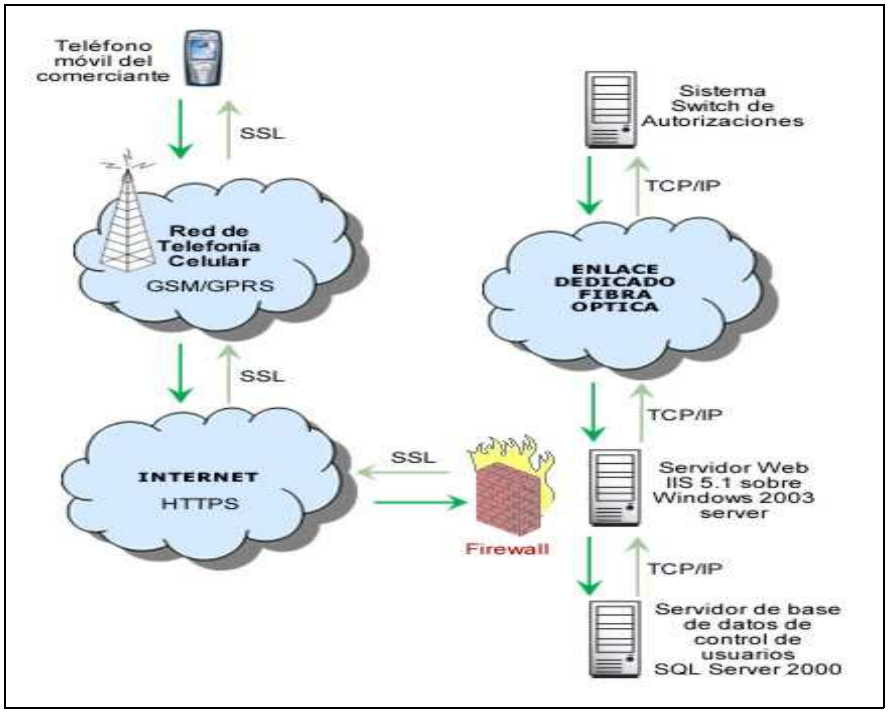


Figura 3.1 Diagrama de la arquitectura del sistema.

La arquitectura del sistema utiliza dos servidores: un servidor Web y un servidor de base de datos. En el servidor Web, se alojará un sitio Web desarrollado en ASP.NET que se ejecutará sobre Internet Information Server. Este sitio Web se comunicará con el servidor de base de datos a través de TCP/IP, este servidor almacenará en una base de datos SQL Server los datos de los comerciantes que harán uso del sistema y las transacciones que estos realicen.

Acogiéndonos a las normas PCI, estos servidores estarán protegidos por un firewall [20] adecuadamente configurado.

Para el acceso al sistema a través de un teléfono móvil, éste debe tener correctamente configurados los siguientes parámetros: La dirección Uniform Resource Locator (URL) [21] del servidor y los datos para autenticación del comerciante y equipo. Para la realización de la conexión con el servidor, el teléfono móvil debe iniciar el establecimiento de una conexión segura, para lo cual recibirá el certificado del servidor Web el cual estará firmado por una reconocida entidad emisora de certificados electrónicos [22] y una vez reconocido, validado dicho certificado se establecerá una conexión segura SSL entre el teléfono móvil y el servidor Web. Una vez realizado esto, los datos de autenticación son recibidos por un script en ASP.NET. Este script permitirá la comunicación con el servidor de base de datos para la comprobación de los datos de autenticación, y posterior realización de la transacción solicitada por medio de la conexión vía TCP/IP a un sistema Switch de procesamiento de transacciones con tarjeta de crédito de una reconocida entidad en nuestro medio.

Para realizar la conexión del teléfono móvil al servidor Web se utilizan dos redes, en primer lugar se debe conectar a la red del operador de telefonía móvil que provee este servicio el cual a su vez le permitirá enviar los datos a través de la red de Internet.

A continuación describiremos con lujo de detalle todos los aspectos técnicos relacionados tanto al flujo de información como a su procesamiento en las diferentes partes del sistema.

3.2 Descripción del Software.

En esta sección realizaremos un análisis de las aplicaciones requeridas para el desarrollo del proyecto que son las siguientes:

- La aplicación a instalar en el teléfono celular del comerciante desarrollado en J2ME, a la cual de ahora en adelante llamaremos Aplicación Cliente.

- El servidor de control de usuarios del sistema al cual llamaremos Aplicación Servidor.

3.2.1 Descripción de la Aplicación Cliente en J2ME

Para la implementación de esta aplicación se ha elegido la plataforma J2ME.

Funcionamiento de la aplicación cliente

Una vez instalada la aplicación cliente en el teléfono celular, al ser ejecutada, lo primero que se le muestra al usuario es la pantalla de configuración del sistema (Figura 3.2). Esta será de uso exclusivo de un técnico o ejecutivo encargado de instalar la aplicación y permitirá ingresar los siguientes datos: el nombre de usuario del comerciante que utilizará la aplicación, la contraseña para el usuario, número identificador para equipo en el cual se instala la aplicación y la dirección URL de la aplicación servidor. El manejo de estos datos se lo realiza por medio de dos niveles de abstracción:

- En el nivel más alto mediante una instancia de la clase que hemos denominado ConfigInfo [11], que permite el almacenamiento y posterior recuperación de cualquiera de los parámetros mencionados.

- La clase ConfigInfo la cual a su vez hace uso de otra clase denominada RecordManager [11] que cumple la función de un administrador generalizado del RMS del sistema.

Figura 3.2 Pantalla de configuración de la aplicación cliente.

El usuario, contraseña y número identificador del equipo son los datos que se utilizan para la autenticación en la base de datos a la que se conecta la aplicación servidor. Por razones

principalmente de seguridad se ha decidido que estos datos de autenticación no estén al alcance del comerciante que utilizará la aplicación, si no que se mantengan almacenados internamente en el equipo. Es por esto que la pantalla de configuración solo se mostrará la primera vez que se ejecute la aplicación, esto significa que si hubo algún error en el ingreso de estos datos, la única forma de corregirlo será volviendo a instalar la aplicación.

Una vez instalada y configurada la aplicación cliente, está lista para permitir al comerciante efectuar cobros con tarjeta de crédito a sus clientes. Para realizar una transacción el comerciante deberá ingresar a la aplicación cliente instalada en el teléfono celular y seleccionar la opción Transacción, una vez hecho esto, será creada una instancia de la clase Transacción. Esta clase tiene como atributos todos los parámetros necesarios a ser ingresados por el comerciante para efectuar una transacción y su función es ir registrando todos estos datos a lo largo de la secuencia de pantallas en donde éstos se le solicitan al comerciante (Figura 3.3). Estos parámetros son los siguientes:

- Tarjeta de crédito

- Tipo de diferido.

- Número de la tarjeta de crédito.

- Número de verificación de la tarjeta (CVV2/CVC2/CID)

- Fecha de caducidad de la tarjeta de crédito

- Número de celular del cliente

- Monto de la transacción.

- Número de meses en el caso de diferido.

Figura 3.3 Secuencia de pantallas para realizar una transacción con la aplicación cliente.

En la última pantalla y una vez verificado que todos los datos hayan sido ingresados correctamente, se crea una instancia de la clase DataTranSender, la cual es la encargada de realizar la conexión con la aplicación servidor. El constructor de esta clase recibe como parámetros los parámetros recogidos por la clase Transacción y hace uso de la clase ConfigInfo para recuperar la información de configuración de la aplicación. El siguiente fragmento de código muestra la instanciación de la clase DataTranSender. Una vez instanciada esta clase, se

realiza la comunicación en un nuevo hilo de ejecución mediante el método start () de la clase.

DataTranSender tr = new DataTranSender(midlet,this,username,password, midlet.tran.getTarjeta(),midlet.tran.getNumeroTarjeta(),midlet.tran.getNumeroCid(), midlet.tran.getFechaCaducidad(),midlet.tran.getFonoCliente(),midlet.tran.getValor(), midlet.tran.getTipoDiferido(), midlet.tran.getNumeroMeses()); tr.start();.

Posteriormente todos estos datos se pasan como parámetros al constructor de la clase Trama, que es la encargada a su vez de generar como su nombre lo indica, una trama de longitud variable conteniendo los datos de la transacción y que será enviada al servidor. La composición de esta trama se muestra en el anexo 3.

3.2.2 Descripción de la aplicación Switch y servidor

Funcionamiento de la aplicación servidor

La clase Default es la clase principal de la aplicación y de acuerdo al modelo code-behind[23] de ASP.NET contiene el código de script detrás del archivo Default.aspx que es la página a la cual las aplicaciones cliente realizarán las peticiones.

El primer proceso que se realiza en esta clase es el de autenticación, para esto es necesario obtener los tres datos de autenticación recibidos en la trama enviada desde la aplicación cliente, y esto se obtiene mediante el método GetLogin(). El proceso de autenticación consta de tres niveles: El primero en el que se realizará la autenticación de la aplicación cliente propiamente dicha a través del agente de usuario, El segundo en donde se autenticarán los datos del comerciante y finalmente la autenticación mediante el número identificador del equipo. Estos niveles de autenticación serán descritos en más detalle en la sección siguiente donde nos referimos al tema de seguridades.

Una vez generada la trama, el siguiente paso es realizar el envío de ésta hacia la aplicación servidor, esta será la primera parte de la comunicación entre componentes en la arquitectura del sistema. Para el envío de datos, la clase DataTranSender a través de su método sendData() hace uso de una instancia de la clase HttpsConnection la cual se encuentra contenida en el paquete javax.microedition.io de J2ME que define los métodos y propiedades necesarios para establecer una conexión de red segura. El establecimiento de una conexión por medio de este método en J2ME es necesario realizarlo en un hilo independiente de ejecución, esto se logra por medio de los métodos start() y run() también definidos en la clase DataTranSender. Estos métodos permiten crear y ejecutar el método sendData() en un nuevo hilo.

Antes de hacer uso del objeto de la clase HttpsConnection para el envío de los datos, es necesario especificar otros dos parámetros. Primero debemos definir el método envío de información al servidor Web. En nuestro caso utilizaremos el método GET. Este método solicita información al servidor Web, en nuestro caso a un script desarrollado en ASP.NET que se

ejecuta en la aplicación servidor. Luego, es muy importante en el caso de nuestra aplicación, definir antes del envío la propiedad User-Agent ya que como veremos más adelante, está nos provee de un nivel adicional de autenticación además de los anteriormente descritos.

En el siguiente fragmento de código se muestran los pasos principales que se realizan en el método sendData() para el establecimiento de la conexión con el servidor:

```
http = (HttpsConnection) Connector.open(URL);

// fijar el método de envío de datos como GET

http.setRequestMethod(HttpConnection.GET);

    //Especificar el User Agent

http.setRequestProperty("User-Agent","Profile/MIDP-2.0
Configuration/CLDC-1.1 (Cobro Electronico Celular
197997492)");

if (http.getResponseCode() == HttpsConnection.HTTP_OK){

// Si la respuesta del servidor es la esperada, recibir los datos
```

```
sb = new StringBuffer();

int ch;

recibir = http.openInputStream();

while ((ch = recibir.read()) != -1)

sb.append((char) ch);}

        else{

        // Si la respuesta del servidor no es la esperada, se
        obtiene el código y mensaje de error

respuesta_error = "Error, Codigo respuesta http: " +
http.getResponseCode()                              + " " +
http.getResponseMessage();              }
```

Una vez que se ha pasado por el proceso de autenticación anteriormente descrito, la aplicación está lista para procesar la parte de la trama recibida que contiene los datos de la transacción. Para esto se hace uso de la clase Trama. Esta clase al igual que la clase del mismo nombre implementada en la aplicación cliente, se encargará de generar una nueva trama, ahora con el formato requerido por un nuevo receptor que en

este caso será el sistema Switch de autorizaciones que nos proveerá del servicio de procesamiento de las transacciones. En el Anexo 4, detallamos el formato que debe tener esta nueva trama.

Una vez que se obtiene la trama en el formato descrito, lo cual se logra mediante el método GetTrama() de la clase Trama, se puede proceder a enviar la misma hacia el sistema Switch de Autorizaciones, el cual responderá con la respectiva autorización o negación del cobro. La clase encargada de implementar la comunicación con el Switch la hemos denominado Gateway.

Básicamente, esta clase provee los métodos necesarios para lectura y escritura en dos archivos Entraxxx.dat y Rxxxyyy.dat que a su vez serán utilizados por una aplicación provista para conexión con el Switch de autorizaciones y de esta manera solicitar la autorización de los pagos.

Cabe mencionar que previo al envío de una trama al Switch de autorizaciones, se realiza una validación del número de tarjeta de crédito para comprobar si este cumple con la especificación ISO 2894, el anexo 5 tenemos una gráfica detallada del funcionamiento del mismo con el formato ISO 8583 el mismo que es utilizado por el SWICTH. Esta especificación provee de un algoritmo que permite comprobar la validez de la estructura del número de tarjeta de crédito, el siguiente paso es grabar en la tabla locales de donde provino el mensaje de que celular y local, en la base de datos Mensajes.MDB graba y ve a que procesador pertenece la tarjeta para poder establecer conexión, y crear el archivo de acuerdo a lo solicitado por dicho procesador, este a su vez vuelve a establecer una conexión de respuesta con el programa Switch, pasa a través de la tabla Local para ver a que local corresponde la respuesta, va al subprograma SWITCH2 Distribuidor y pasa a grabar en la base Mensajes.MDB si respuesta de la petición, pasa Tabla Locales para ver a que celular debe devolver la respuesta, graba en el archivo RXXXYYY.DAT si no tiene asterisco en la primera posición, sino tiene que esperar a que el Servidor Web a través de su clase gateway coloque dicho asterisco, porque si no lo

tiene entonces esta listo para que el Servidor Web lea la respuesta y coloque el asterisco en la primera posición.

Debemos tomar en cuenta que el programa Switch, considera un tiempo de Time-Out (de 40 a 60 segundos), si en ese tiempo no recibe respuesta, enviar una transacción 3 (Reversa Automática) y luego solicitar nuevamente la Autorización, este paso suele hacerse de 2 a tres veces.

Esta aplicación SWITCH está desarrollada en Visual Basic, es una aplicación distribuida, la misma que esta en funcionamiento en algunos establecimientos como son: Lloyds Bank, Banco Pichincha, Marathon Sports, Fybeca, etc., trabaja en algunas plataformas como son: Windows (95,98, Milenium, Xp, Nt, Server 2.00x), 4690, Novell y Linux. En el anexo 6 presentamos información adicional a este SWITCH.

3.3 Seguridades

3.3.1 Estándares de Seguridad.

Dentro de los estándares de seguridad que seguimos en nuestro proyecto tenemos: ISO 2894, ISO 8583 que utiliza el Switch, y los estándares PCI [7].

Cabe mencionar que previo al envío de una trama al Switch de autorizaciones, se realiza una validación del número de tarjeta de crédito para comprobar si este cumple con la especificación ISO 2894. Esta especificación provee de un algoritmo que permite comprobar la validez de la estructura del número de tarjeta de crédito.

El algoritmo ISO 2894 es el siguiente:

1. Calcular el peso para el primer dígito: si el número de dígitos es par el primer peso es 2 de lo contrario es 1. Después los pesos alternan entre 1, 2, 1, 2, 1 …
2. Multiplicar cada dígito por su peso.

3. Si el resultado del 2º paso es mayor que 9, restar 9.
4. Sumar todos los dígitos.
5. Comprobar que el resultado es divisible por 10.

3.3.2 Criptografía.

La técnica en donde se utiliza la misma llave tanto para encriptar como para desencriptar la información enviada se denomina criptografía simétrica, mientras que la técnica en donde se utiliza una clave pública para encriptación y una privada para desencriptación se denomina criptografía asimétrica. En nuestro proyecto se trabaja con el protocolo de seguridad 128-bit Secure Socket Layer (SSL) 3.0. [27]

Este nos asegura una conexión encriptada a través de un esquema mixto. Este usa tanto el sistema simétrico como el asimétrico de la siguiente forma: La clave simétrica es cifrada con la clave pública, y el mensaje saliente es cifrado con la clave simétrica, todo combinado automáticamente en un sólo paquete. El destinatario usa su clave privada para descifrar la clave simétrica y acto seguido

usa la clave simétrica para descifrar el mensaje. Además mediante el uso de un certificado digital proveniente de una autoridad certificadora se garantiza que la clave pública que estará en los teléfonos celulares corresponde a la clave privada del servidor. Los algoritmos comúnmente usados en este esquema son: RSA o DSA para cifrado asimétrico y RC4, IDEA o 3DES para cifrado simétrico [28].

Como ya se había mencionado el perfil MIDP 2.0 que es el perfil que utilizamos para el desarrollo de la aplicación cliente, brinda mediante la interface javax.microedition.io.HttpsConnection, soporte tanto para trabajar con el protocolo SSL 3.0 o con su versión actualizada denominada TLS 1.0 y entre las funcionalidades más interesantes que ofrece está la de proveer información detallada sobre la conexión segura que se está utilizando. Para esto se cuenta con el método getSecurityInfo () en cual retorna una instancia de otra interface denominada SecurityInfo. Es a través del método getCypherSuite () de esta interface que se pudo obtener la información sobre el paquete de cifrado utilizado por nuestra conexión segura realizada mediante un certificado digital instalado

en el servidor y proveniente de VeriSign [22]. La ejecución de este método arrojó en nuestro caso el resultado siguiente:

TLS_RSA_WITH_RC4_128_MD5

Esto significa que se está usando el algoritmo asimétrico RSA [24] para el intercambio de claves, mientras que para la encriptación de la información a enviar se está usando el algoritmo simétrico RC4 con una clave de 128 bits, además del uso de MD5 como función de hash[28].

3.3.3 Descripción de la Arquitectura de Seguridad

Seguridades en el teléfono celular

En lo que se refiere a la aplicación instalada en el teléfono celular uno de los principales desafíos existentes es el hecho de que en el dispositivo estarán guardados datos confidenciales del comerciante como son su nombre de usuario, contraseña e identificador de equipo.

Como ya lo habíamos mencionado anteriormente la especificación original MIDP define un método de almacenamiento a través del

Sistema de almacenamiento de registros o Record Management Store (RMS)[11]. Al ser este el método usado para guardar de los datos de autenticación en el dispositivo, fue necesario investigar el nivel de seguridad que ofrece esta forma de almacenamiento. Sobre esto se pudo averiguar que un Midlet puede compartir su almacén de registros con otros, pero es posible declararlo como privado al momento de su creación. De este modo se elimina cualquier posibilidad de acceso no autorizado al RMS de la aplicación cliente.

Lo anteriormente descrito no garantiza la seguridad de la aplicación cliente, debido a que en caso de caer el paquete generado para instalación en malas manos, este fácilmente puede ser descompilado por un usuario experto a través de diversas herramientas que permitirían obtener su código fuente. Esto puede ser usado para procesos de ingeniería reversa y acceso a los datos de autenticación guardados en el RMS de la aplicación. La manera más extendida entre los desarrolladores para enfrentar este problema es la denominada ofuscación de código. Este proceso consiste en una modificación deliberada del código de forma tal que resulte prácticamente imposible descompilarlo e incluso interpretarlo correctamente aun si se obtiene el código fuente por el

procedimiento que fuera. Esta ofuscación se consigue en la práctica por medio de la inclusión de bucles irrelevantes, cálculos innecesarios, comprobaciones absurdas, nombres de funciones y de variables que no tienen nada que ver con su cometido, funciones extensas que no sirven para nada, interacciones inverosímiles entre variables y funciones, etc. Existen varias aplicaciones gratuitas que permiten realizar este proceso denominadas ofuscadores. En nuestro caso, el IDE Netbeans 5.0 [26], incluye la posibilidad de ser configurado para realizar este proceso de forma automática al momento de empaquetar la aplicación para distribución.

Seguridades en la comunicación con la aplicación servidor.

Como ya se había mencionado para la transmisión de los datos hacia la aplicación servidor se utilizará una conexión a Internet a través de la red GPRS del operador de telefonía móvil. Una vez que la información se encuentra en Internet, esta viaja a través de una variedad de equipos de red como servidores, routers, entre otros, esto significa que en cualquier punto de la red la información es susceptible de ser interceptada usando los conocimientos y herramientas adecuadas.

Al estar utilizando una red pública para la transmisión de datos confidenciales, es necesario de acuerdo a las normas PCI, usar una tecnología especializada de comunicación segura. En nuestro caso hemos decidido usar el medio más estandarizado para la transmisión de datos sobre Internet sobre todo en lo referente a aplicaciones o sitios de comercio electrónico. Se trata de SSL, en donde se reemplaza una conexión vía HTTP por otra HTTPS o HTTP seguro.

Para la conexión de las aplicaciones cliente y servidor se utiliza dos tipos de autenticación:

- La autenticación de servidor se refiere al proceso por medio del cual la aplicación servidor se identifica ante la aplicación cliente.
- La autenticación del cliente se refiere al proceso mediante el cual la aplicación cliente se identifica ante la aplicación servidor.

Autenticación del servidor

El procedimiento consiste de los siguientes pasos

- Importar el Test CA Root proveniente de la entidad de donde se obtuvo el certificado de prueba en un keystore o almacén de claves de Java, lo cual se lo hizo mediante el comando keytool de Java. A continuación mostramos un ejemplo del uso de este comando:

  Keytool -import -keystore keystore -storepass password -file ca_cert.txt -alias httpsca

  En donde keystore es el nombre del almacén de claves al cual se va a agregar, password la contraseña del almacén de claves, ca_cert.txt el archivo que contiene el Test CA Root y httpsca el nombre que se le asignará a dicha clave pública en ese almacén de claves.

- Posteriormente una vez incluido el certificado en el keystore, esté adquiere el formato adecuado y queda listo para ser importado al almacén de claves del emulador. Esta importación se la hizo usando la herramienta MEKeytool que es el

equivalente en el perfil MIDP del comando keytool. Presentamos a continuación un ejemplo de su uso:

Java -jar bin/MEKeyTool.jar -import -keystore keystore -alias httpsca -storepass password

- Opcionalmente se puede chequear la lista de claves públicas contenidas en el almacén de claves mediante la siguiente sentencia:

Java -jar bin/MEKeyTool.jar -list

Una vez realizado este proceso de manera correcta el dispositivo que en nuestro caso fue el emulador, puede autenticar al servidor. A continuación mostramos un ejemplo de una conexión segura en J2ME:

String URL = “https://unhost/unarchivo.aspx”;

HttpsConnection c = (HttpsConnection)Conector.open(URL);

Autenticación del cliente

Para incrementar las seguridades en el caso de la autenticación del cliente en el servidor, se ha implementado una arquitectura de autenticación de clientes de tres niveles en la aplicación servidor.

- En el primer nivel se obtiene el Agente de usuario mediante la propiedad UserAgent de la clase Request que es la que en una aplicación Web de ASP.NET se encarga del acceso a los detalles de las peticiones del cliente. Este nivel permite validar el acceso, de manera que solo puedan acceder los tipos de aplicaciones autorizadas al siguiente nivel.

- Una vez que se ha comprobado que el tipo de aplicación que intenta acceder es válido, se pasa al siguiente nivel de autenticación, en donde se realizará la consulta a la base de datos para comprobar mediante el usuario y contraseña que efectivamente se trata de un cliente autorizado.

- Una vez comprobados los datos del cliente, el tercer nivel consiste en autenticar la aplicación instalada en el equipo y determinar si esta pertenece al usuario que se autenticó en el nivel anterior por medio del número identificador de equipo

En la figura 3.4 se muestra un diagrama de la arquitectura implementada para la autenticación de aplicaciones cliente:

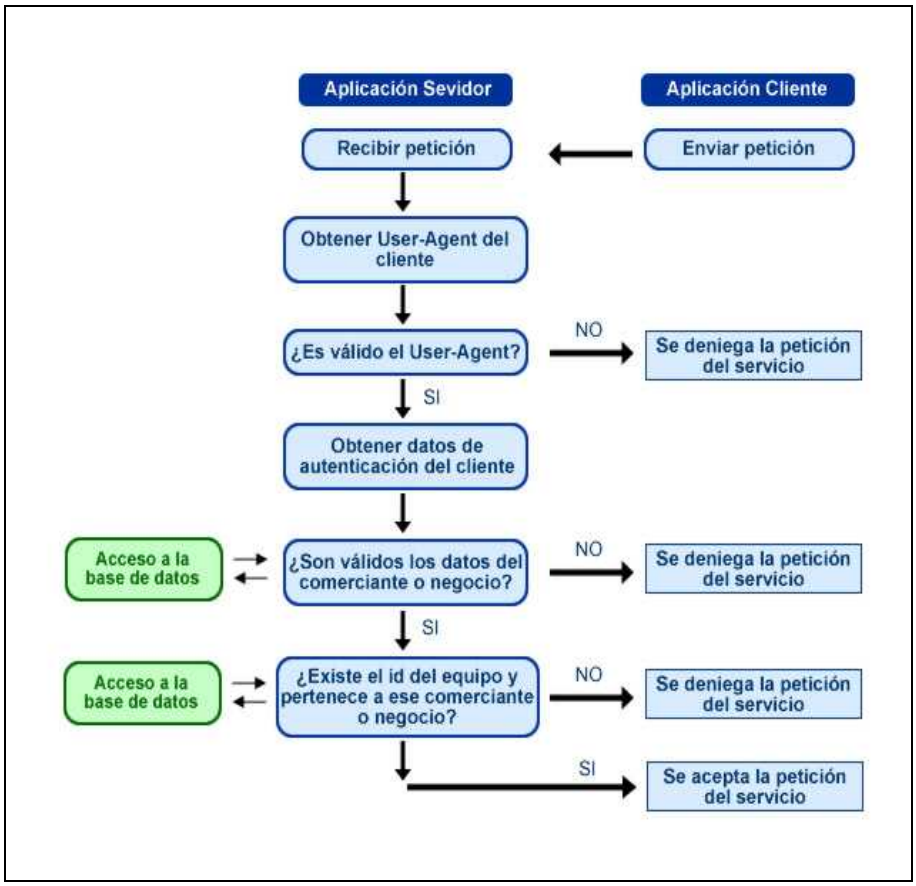


Figura 3.4 Diagrama de flujo de autenticación de aplicaciones cliente.

CAPÍTULO IV

4. IMPLEMENTACION DEL PROYECTO.

En el presente capítulo explicaremos de manera detallada la implementación del proyecto a través de sus modelos de clase, de datos, y los diferentes módulos que conforman nuestra aplicación.

4.1. Módulos de Aplicación.

En nuestra aplicación hemos considerado el desarrollo de algunos módulos que forman parte del modelo principal propuesto en la Figura 1.1, estos son:

- Módulo de Aplicación Cliente (MAC).

- Módulo de Aplicación Servidor (MAS).

- Módulo del sistema Switch de Autorizaciones (MSSA).

En la Figura 4.1 se puede observar el modelo de nuestro proyecto separado en los módulos de aplicación que se enunciaron anteriormente.

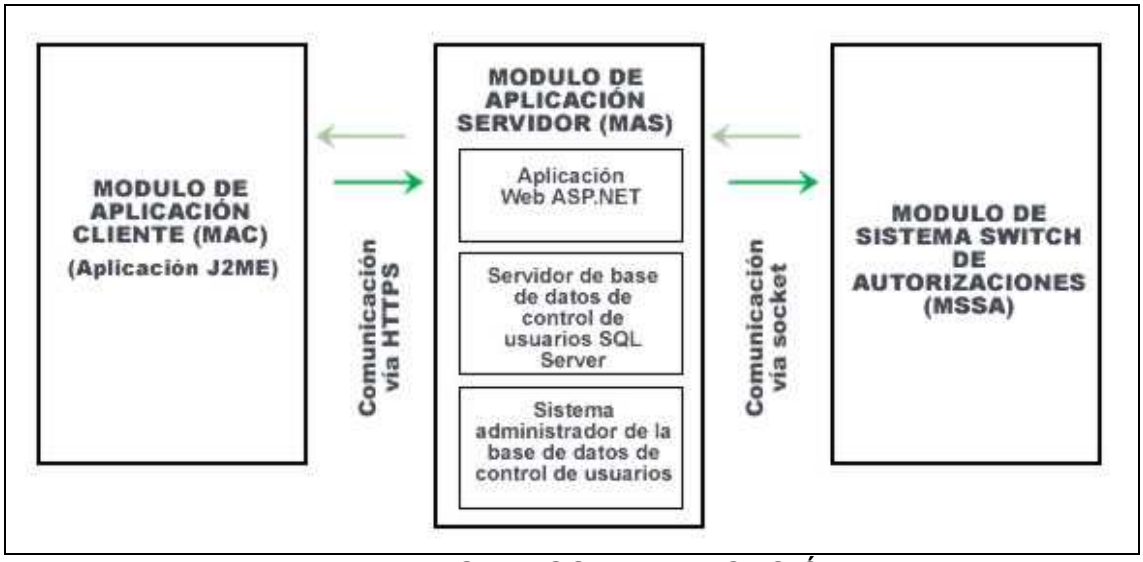


Figura 4.1 MODULOS DE APLICACIÓN

El primer módulo hace referencia a la aplicación cliente mencionada en el capítulo anterior y consiste del software que debe instalarse en el teléfono celular, que cumple la función de brindar la interface de usuario que permita establecer de una manera sencilla y segura la comunicación con la aplicación servidor para la realización de una transacción.

El segundo módulo se refiere a la aplicación servidor descrita en el capítulo anterior y está compuesta por 3 subcomponentes principales que son:

- La aplicación Web desarrollada en el lenguaje ASP.NET que cumple la función de implementar la lógica del negocio.

- El servidor de bases de datos que se ejecuta sobre el sistema de gestión de bases de datos SQL Server y que almacena los datos de los usuarios del servicio así como los registros de todas las transacciones realizadas por estos.

- El sistema administrador de la base de datos de control de usuarios desarrollado en lenguaje ASP.Net que permite administrar de manera sencilla la base de datos de usuarios.

Finalmente tenemos el MSSA. Lo más lógico es considerar a este sistema como un módulo a parte ya que se trata de una entidad externa que cumple con la función de brindar a nuestro proyecto de una manera transparente y segura el servicio de conexión con las diferentes entidades emisoras de tarjetas de crédito para la realización de las transacciones.

Cabe indicar que en el capítulo 3 se realizó una descripción detallada del funcionamiento de los principales componentes de cada uno de estos módulos, así como de los aspectos técnicos involucrados en los mismos.

4.2. Modelo de Clases.

A continuación describiremos la estructura de las clases principales que se utilizan en el proyecto. Se utiliza un diagrama de clases que es una notación del Lenguaje de Modelamiento Unificado (UML) [29] que se realiza en la etapa de diseño.

A continuación se muestra el diagrama de clases para la aplicación en J2ME que se ejecuta en el teléfono celular, la cual correspondiente al primer módulo. En la figura 4.2 mostramos la primera parte del diagrama de clases de esta aplicación. Esta primera parte consta de las clases que implementan la funcionalidad principal de la aplicación, la cual se mencionó previamente en este capítulo en la descripción del MAC. Estas clases son las siguientes:

- MidletPagoElectronico: Es la clase principal del midlet y es en la implementación de esta clase donde se instancian y se inicializan todos los objetos necesarios para el funcionamiento de la aplicación en J2ME que se ejecuta en el teléfono móvil.

- Transacción: Esta clase tiene la función de almacenar y permitir el acceso y modificación de todos los datos de una transacción que son ingresados por el usuario mediante la interface gráfica de la aplicación.

- DataTranSender: Es una de las clases más importantes de la implementación debido a que permite el establecimiento de la comunicación con el MAS. El funcionamiento de esta clase fue descrito de forma detallada en el capítulo 3 sección 3.2.1 pagina 43.

- Trama: Se encarga de convertir los datos de la transacción al formato de trama requerido para su envío al MAS. En el anexo 3 se muestra en detalle este formato.

En la figura 4.3 se muestra la segunda parte del diagrama de clases de la aplicación cliente. El objetivo principal en esta segunda parte del diagrama de clases es mostrar la relación de las clases principales con otras clases adicionales que sirven de apoyo para lograr el correcto funcionamiento de la aplicación. Estas clases son las siguientes:

- FormDataTran: Es una de las clases que tienen la función de implementar la interface gráfica que permite el ingreso de los datos de una transacción a ser enviados al servidor por medio del teléfono celular. Hemos seleccionado a esta clase para ser mostrada en este diagrama debido a que es la que implementa la interface para el ingreso de los datos más relevantes en una transacción de solicitud de autorización de cobro.

- ConfigInfo: Esta es una clase importante ya que cumple con la función de hacer transparente a la aplicación el acceso al sistema administrador de registros o RMS del midlet para el ingreso, modificación y consulta de los datos de configuración de un terminal en el mismo.

- TranParser: Esta clase interviene al momento de recibir las respuestas de la aplicación servidor a las peticiones realizadas y se encarga de interpretar el código HTML recibido para mostrar al usuario los mensajes de respuesta de una manera clara.

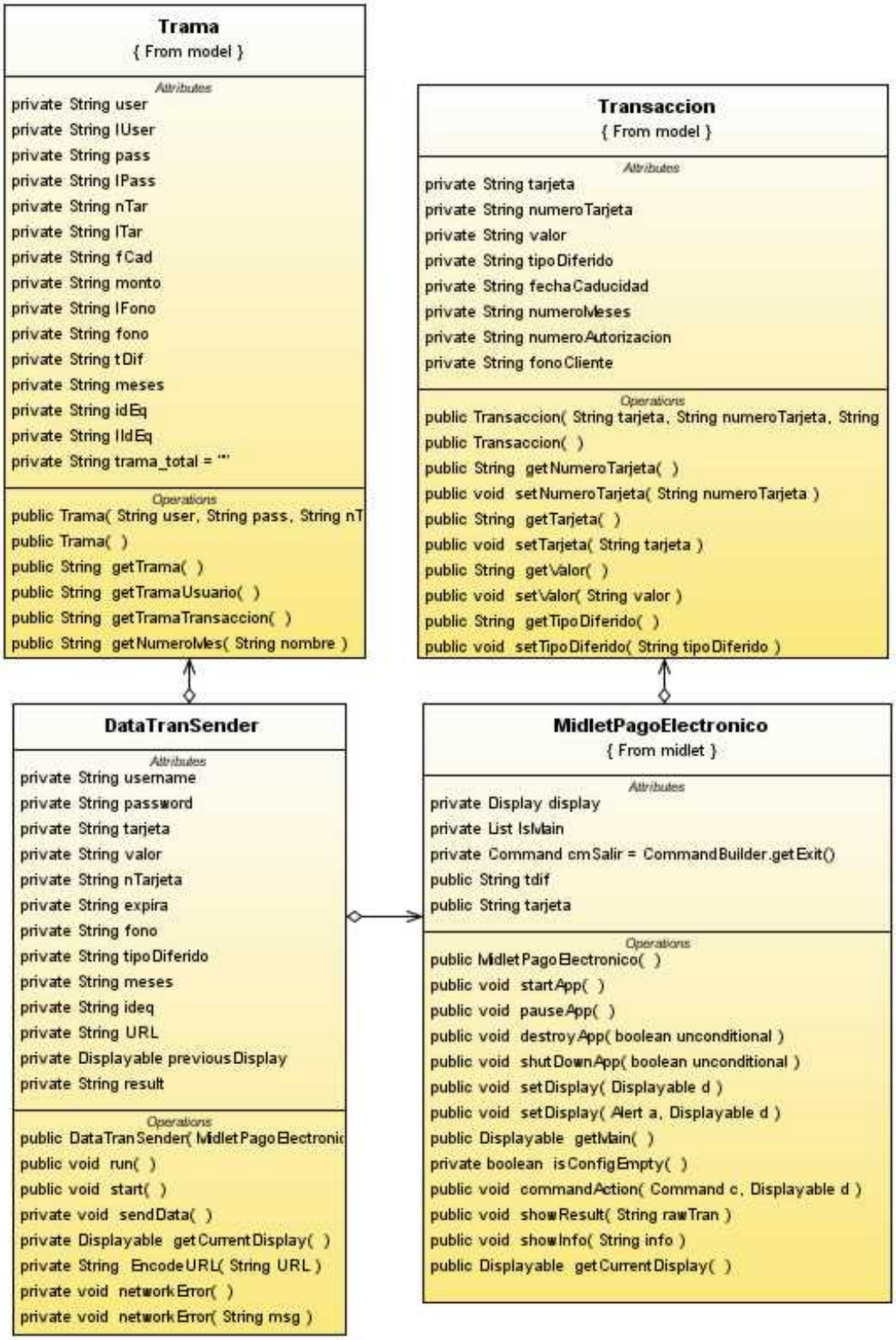


Figura 4.2 Diagrama de clases de la aplicación cliente. Parte 1.

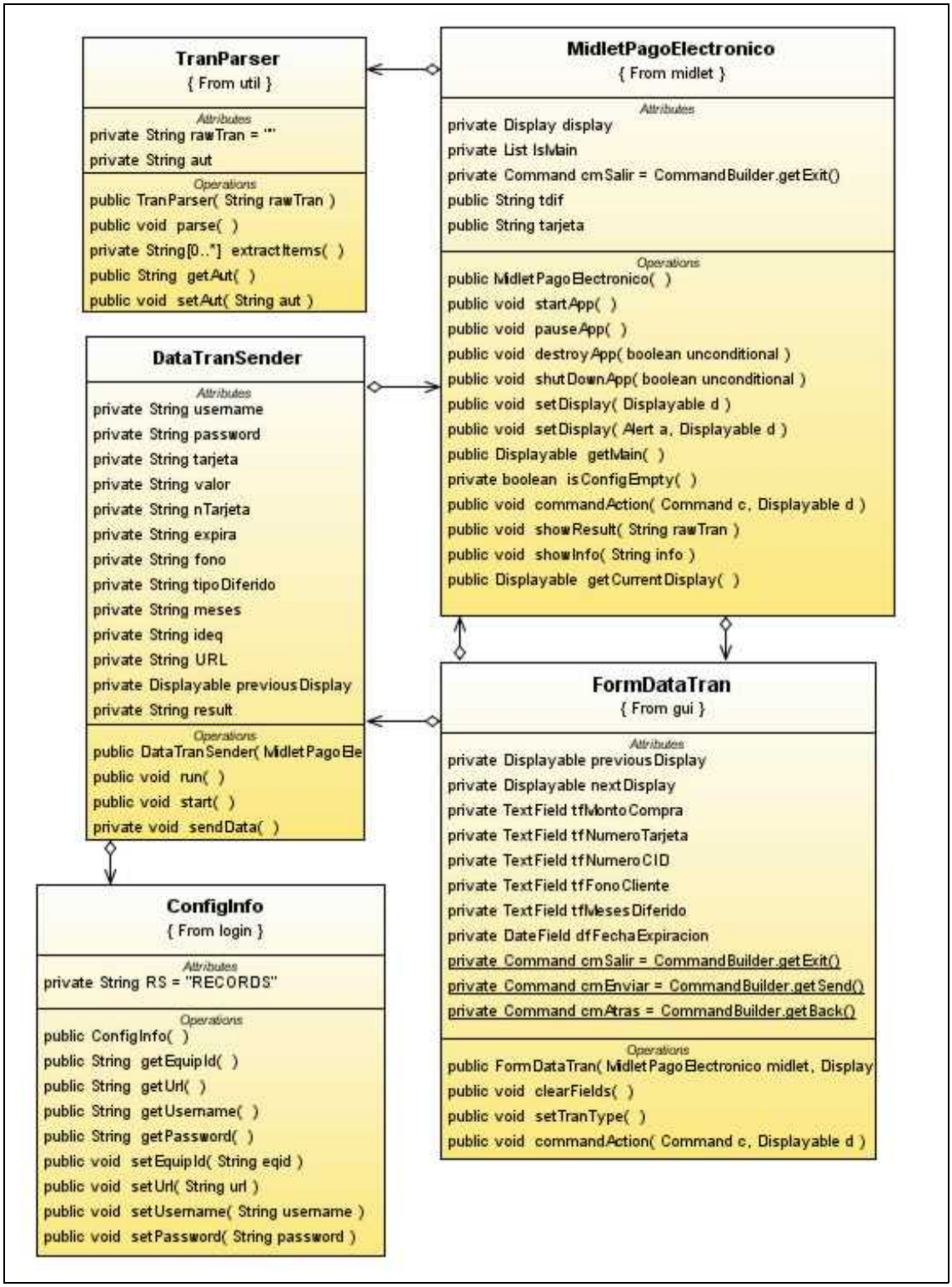


Figura 4.3 Diagrama de clases de la aplicación cliente. Parte 2.

En la figura 4.4 se muestra el diagrama de clases de la aplicación Web en ASP.NET que se encuentra en el modulo de aplicación servidor. Se detalla las principales clases de la implementación de la misma que son las siguientes:

- Default: Esta clase contiene el código del script detrás del archivo Default.aspx que es la página a la cual las aplicaciones cliente realizarán las peticiones. Es la clase principal de la aplicación y desde esta se instancian todas las demás clases necesarias para el funcionamiento de la misma.

- Trama: Esta clase al igual que la clase del mismo nombre implementada en el MAC, se encargará de generar una nueva trama, ahora con el formato requerido por el MSSA que nos proveerá del servicio de procesamiento de las transacciones. En el Anexo 4, se muestra en detalle el formato de esta trama.

- Transacción: Esta clase tiene la función de realizar el acceso a la base de datos de control de usuarios para la realización de inserción y consulta de datos de las transacciones realizadas

por los usuarios del servicio a través de las aplicaciones cliente instaladas en los teléfonos celulares.

- Gateway: Esta clase provee los métodos necesarios para la comunicación con el MSSA, dicha comunicación consiste en realizar lecturas y escrituras de las tramas de datos en los archivos Entraxxx.dat y Rxxxyyy.dat que a su vez son utilizados por una aplicación provista por SuSistema Ltda. para conexión con el MSSA denominada Local.exe, la cual tiene la función de realizar el envío y recepción de tramas de datos al MSSA.

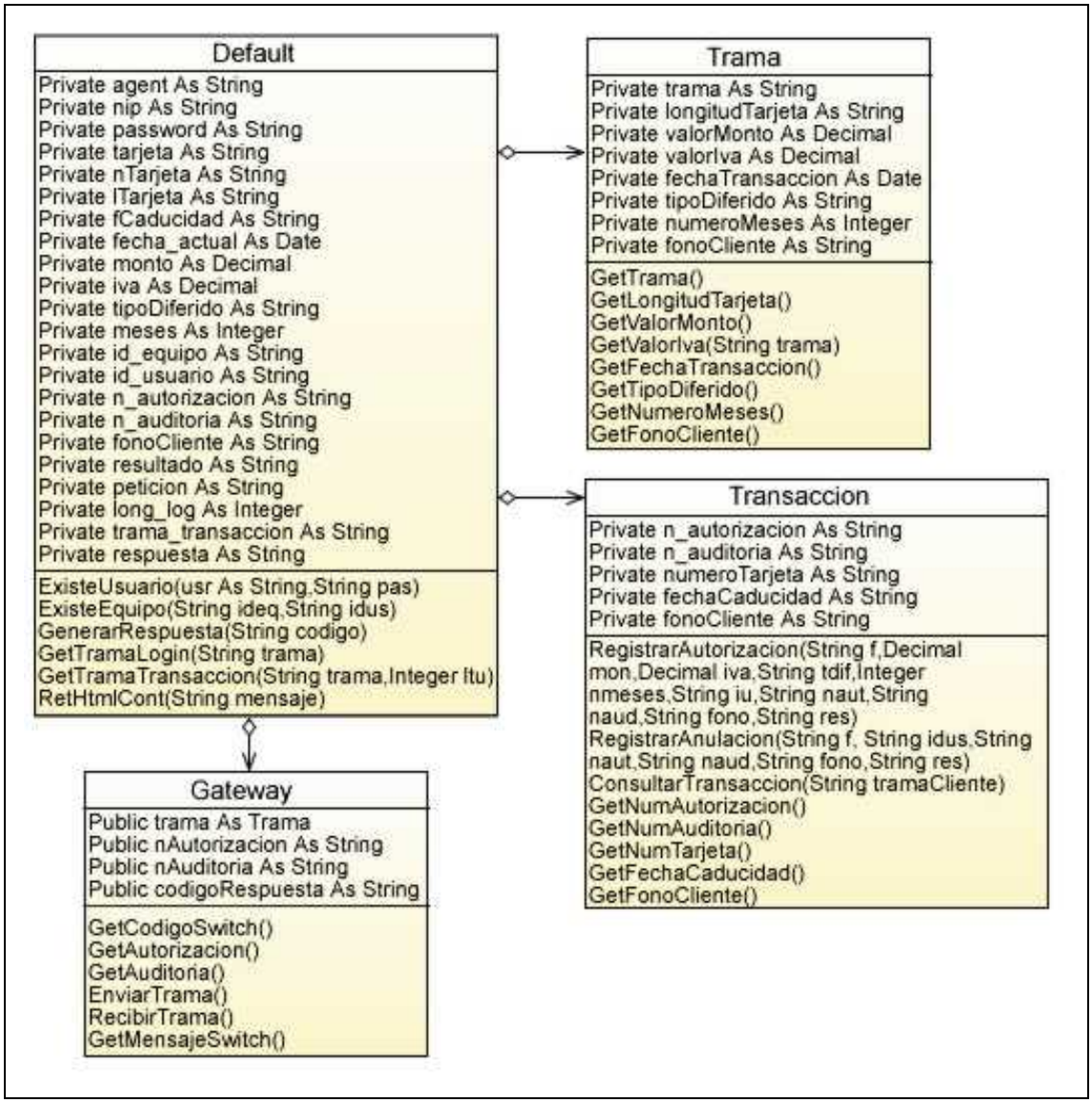


Figura4.4 Diagrama de clases de la aplicación servidor.

Especificación de casos de usos

A continuación mostramos una lista de los casos de usos que se han considerado como los más importantes en el sistema:

1. Ejecutivo de servicio técnico configura un terminal.
2. Usuario solicita autorización de cobro.
3. Administrador accede al sistema de administración de la base de datos de control de usuarios del MAS.
4. Administrador crea un nuevo usuario.
5. Administrador modifica datos de un usuario existente.
6. Administrador agrega un nuevo terminal a un usuario.
7. Administrador modifica datos de un terminal existente.

A continuación se detallará los casos de uso mencionados:

CASO 1: Ejecutivo de servicio técnico configura un terminal.

Descripción: Un ejecutivo encargado de soporte ingresa los datos de autenticación asignados al comerciante dueño del celular y la dirección URL del servidor de control.

Notas:

- La contraseña es asignada por el administrador en el registro del usuario en el servidor de control.
- Una vez instalado el sistema cliente en el teléfono celular de un usuario, la primera pantalla que se muestra es la de configuración.

Valor medible: El acceso al menú principal es otorgado o no.

Escenarios:

1.1.1 Los datos fueron ingresados correctamente y se muestra el menú principal del sistema.

1.1.2 Faltan datos por ingresar, en cuyo caso se muestra un mensaje de notificación y posteriormente se muestra nuevamente la pantalla de configuración.

1.1.3 Alguno o algunos de los datos no son válidos en cuyo caso se muestra un mensaje de notificación y posteriormente se muestra nuevamente la pantalla de configuración.

CASO 2: Usuario solicita autorización de cobro.

Descripción: Un usuario solicita autorización para realizar un cobro a un cliente.

Notas:

- Los datos que se solicitan para realizar la petición de autorización son: Nombre de tarjeta, tipo de diferido, número de tarjeta de crédito del cliente, código de verificación de la tarjeta, fecha de caducidad de la tarjeta, monto de transacción y teléfono del cliente.

Valor medible: El cobro es autorizado o no.

Escenarios:

2.1 Todos los datos están correctos en cuyo caso se muestra la pantalla que indica que la transacción se realizó de manera exitosa y

además muestra el número de autorización asignado por el emisor para la transacción.

2.2 Faltan datos por ingresar, en cuyo caso se muestra un mensaje de notificación y posteriormente se muestra nuevamente la pantalla de ingreso de datos de la transacción.

2.3 Alguno o algunos de los datos de autenticación del usuario del terminal no son válidos en cuyo caso no se realiza la petición de autorización y se muestra un mensaje de notificación.

2.4 Se niega la petición de autorización debido a que el sistema switch de autorizaciones no autorizó el cobro por algún motivo como por ejemplo: errores en los datos del tarjetahabiente o insuficiencia de fondos.

2.5 No se puede realizar la petición de autorización debido a fallas técnicas.

CASO 3: Administrador accede al sistema de administración de la base de datos de control de usuarios del MAS.

Descripción: Un usuario administrador accede al módulo de administración de la base de datos del servidor de control proporcionando su identificador de usuario y contraseña.

Notas:

- Existen dos tipos de usuarios: administradores del servidor de control y los comerciantes que hacen uso de los terminales.

Valor medible: El acceso al módulo de administración es concedido o no.

Escenarios:

4.1 El acceso es concedido debido a que el identificador de usuario y contraseña son correctos.

4.2 El acceso no es concedido debido a que el identificador de usuario, contraseña o ambos están incorrectos.

CASO 4: Administrador crea un nuevo usuario.

Descripción: El administrador del sistema ingresa un nuevo usuario que tendrá acceso al servicio a través del teléfono móvil.

Notas:

- Él módulo de administración asignará de manera predeterminada el número de cédula del comerciante como identificador de usuario.
- La contraseña se generará de manera aleatoria y tendrá una longitud de 12 caracteres.

Valor medible: El nuevo usuario es creado o no.

Escenarios:

5.1 El nuevo usuario es creado de manera exitosa.

5.2 El nuevo usuario no se ingresa por que ya existe.

5.3 El nuevo usuario no se ingresa por que faltan datos.

CASO 5: Administrador modifica datos de un usuario existente.

Descripción: El administrador del sistema modifica los datos de un usuario del servicio.

Notas:

- La búsqueda del usuario se puede realizar en base a su nombre o apellido.
- Se tendrá la opción de generar una nueva contraseña.

Valor medible: Los datos del usuario son modificados o no.

Escenarios:

6.1 Los datos del usuario son modificados de manera exitosa.

6.2 El usuario no se modifica porque no se llenaron todos los campos.

CASO 6: Administrador agrega un nuevo terminal a un usuario.

Descripción: El administrador del sistema crea un nuevo terminal para un usuario del servicio.

Notas:

- Los datos solicitados son: Marca del equipo, modelo y numero identificador del equipo o IMEI.

- El identificador del terminal será numérico y se generará de manera incremental.

Valor medible: El nuevo terminal es agregado o no.

Escenarios:

7.1 El nuevo terminal es creado de manera exitosa.

7.2 El nuevo terminal no se crea porque no se llenaron todos los campos.

CASO 7: Administrador modifica datos de un terminal existente.

Descripción: El administrador del sistema modifica los datos del terminal de un usuario del servicio.

Notas:

- La búsqueda del terminal a modificar se puede realizar en base al nombre o apellido del usuario propietario del terminal.

Valor medible: Los datos del terminal son modificados o no.

Escenarios:

7.1 Los datos del terminal son modificados de manera exitosa.

7.2 Los datos del terminal no se modifican porque no se llenaron todos los campos.

Diagrama de secuencia del sistema

Los diagramas de secuencia constituyen una de las herramientas más efectivas para modelar la interacción entre los objetos de un sistema. Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo.

En nuestro caso hemos decidido mostrar el diagrama de secuencia para el caso de uso 2 en el cual un usuario del sistema efectúa una transacción de cobro mediante la solicitud de autorización con la tarjeta de crédito de un cliente, debido a que es en este caso en el que interactúan todos los componentes del sistema. En la figura 4.5 se muestra el diagrama de secuencia para el caso de uso 2.

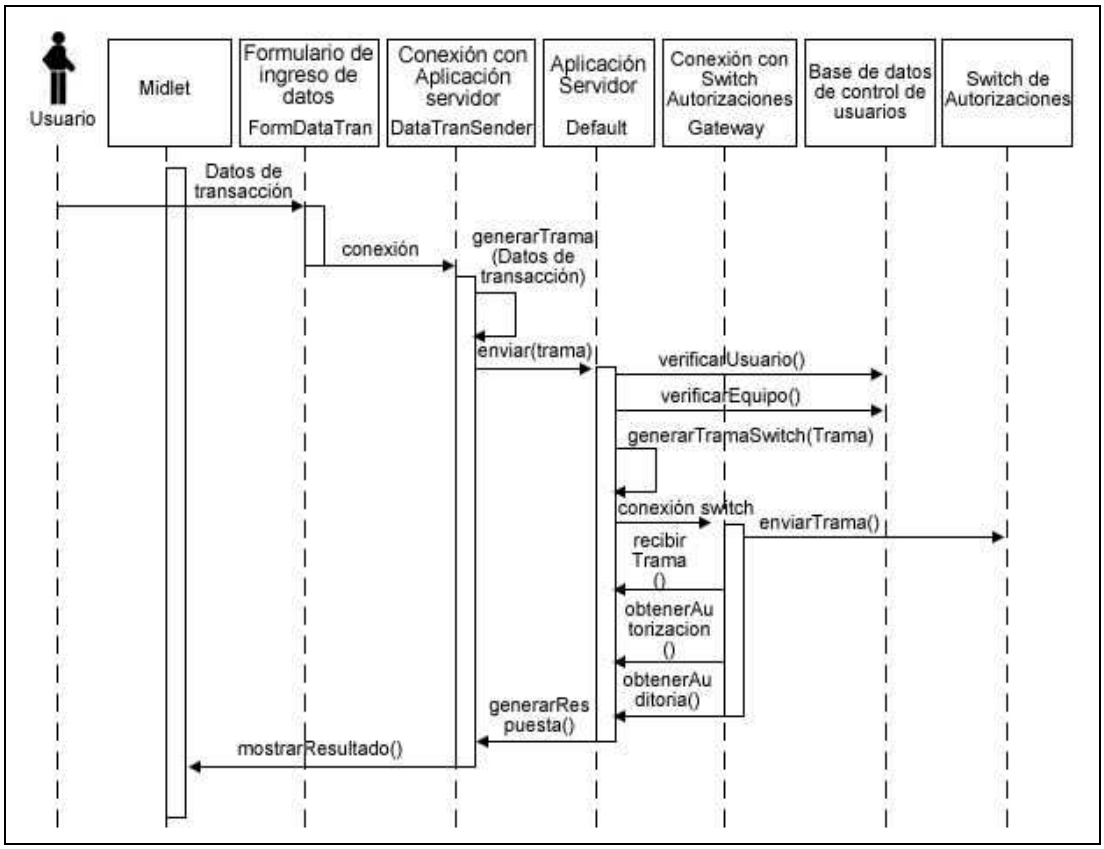


Figura 4.5 Diagrama de secuencia para el caso de uso 2.

4.3. Modelos de Datos.

Como ya se había indicado anteriormente, en el MAS se requiere de un medio que permita realizar el almacenamiento de los datos de los usuarios del servicio y de todos los movimientos que estos realicen. Para esto se diseñó una base de datos de la cual mostramos en la figura 4.6 el respectivo diagrama entidad relación en donde se incluyen las tablas principales de la misma:

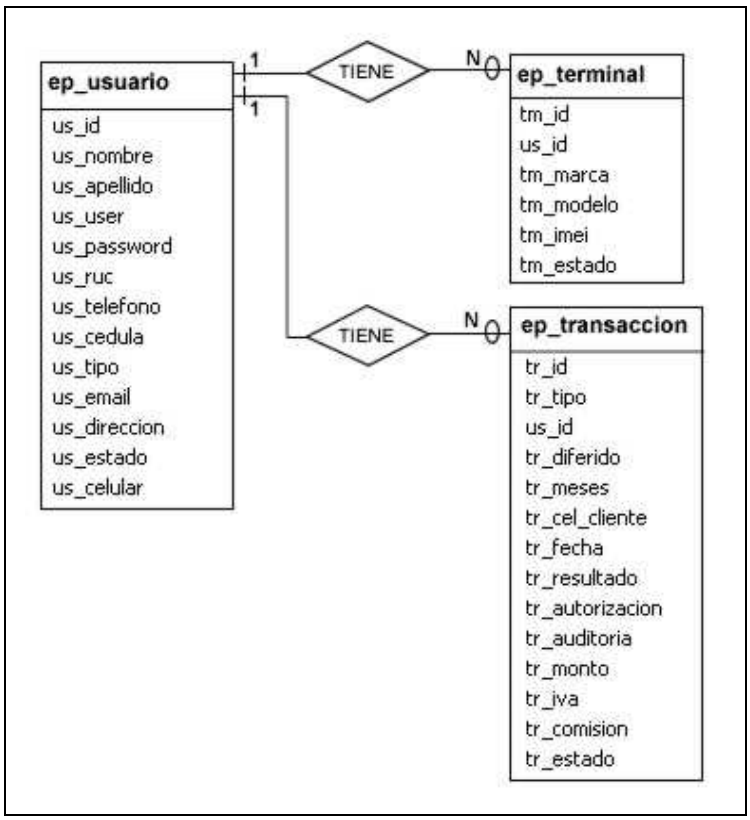


Figura 4.6 Diagrama entidad-relación de la base de datos del servidor de control.

La base de datos del servidor tiene las siguientes tablas principales:

- ep_usuario
- ep_terminal
- ep_transaccion

Para una mejor descripción de cada una de las tablas principales podemos verlo en el Anexo 9.

4.4. Resultados Obtenidos.

A continuación se describen las pruebas que se realizaron al sistema con el fin de obtener los resultados que validen a este proyecto como una solución para el problema planteado en el capítulo 1.

Las primeras pruebas realizadas estuvieron orientadas a evaluar el correcto funcionamiento de cada uno de los módulos del sistema. A continuación se detallan los aspectos principales en cuanto a requerimientos funcionales que fueron evaluados:

- Funcionamiento de la aplicación ASP.NET que implementa la lógica del negocio en el MAS.
- Comunicación entre los módulos MAC y MAS.
- Comunicación de la aplicación ASP.NET con la base de datos en SQL Server 2000 dentro del MAS.
- Funcionamiento del MSSA.
- Comunicación entre el MAS y el MSSA.

Para evaluar el correcto funcionamiento de la aplicación Web ASP.NET que forma parte del MAS, se desarrolló un cliente de prueba vía navegador Web, el cual consistió de una página con un formulario que permitía el envío de una trama a la aplicación y mostraba las respuestas generadas por la misma.

Las pruebas de comunicación entre los módulos de aplicación cliente y servidor consistieron básicamente en pruebas de conexión vía HTTP. En un principio se realizaron con ambos módulos funcionando en una misma máquina y usando dos diferentes emuladores de teléfonos celulares: El proporcionado por el J2ME Wireless Toolkit y el emulador del Nokia Series 40 Developer Platform 2.0 SDK. Una vez logrado el resultado esperado mediante las pruebas desde los emuladores, en los cuales se obtuvo y se interpretó de manera correcta las respuestas generadas por el MAS para las peticiones de tipo GET, se pasó a realizar las mismas pruebas mediante conexión entre el MAC ya funcionando en un dispositivo real y el MAS funcionando en una máquina con plataforma Windows 2003 Server y con una IP pública proporcionada por el GICOM para la realización de dichas pruebas.

Las pruebas de las seguridades del sistema en lo referente a conexión segura sobre HTTPS y autenticación del servidor Web mediante certificado digital pudieron ser realizadas únicamente mediante el emulador y con un certificado de prueba de 14 días de validez obtenido de VeriSign. Se comprobó el correcto funcionamiento en estas pruebas, al verificar que en el caso de los clientes con el certificado instalado estos pudieron realizar la conexión exitosamente al servidor, mientras que a los que no disponían del correspondiente certificado en sus almacenes de claves se les negó la conexión al servidor.

Para realizar las pruebas de conexión con la base de datos de manera rápida se utilizó el cliente Web mencionado anteriormente. Estas pruebas consistieron básicamente en realizar consultas e inserciones en la base.

Una vez obtenidos los resultados esperados en las pruebas anteriores, se procedió a evaluar el funcionamiento del MSSA, para esto se realizó la instalación de la versión del sistema Switch de Autorizaciones que nos fue facilitada por la empresa SuSistema. Se

instaló en un mismo equipo todos los componentes necesarios para el funcionamiento de este sistema, esto permitió realizar una simulación de la operación de este servicio. Estos componentes son 3:

- La aplicación Local.exe la cual está destinada a ejecutarse en cada uno de los locales comerciales que contratan este servicio y que en nuestro caso debe estar instalado en la misma máquina en donde funciona el MAS, ya que nuestro sistema hace las veces de un cliente del servicio proporcionado por SuSistema Cia Ltda. La aplicación Local.exe cumple con la función de envío y recepción de tramas de datos al componente principal del sistema switch de autorizaciones que es la aplicación Host.exe.

- La aplicación Host.exe (El sistema switch de autorizaciones propiamente dicho) es el componente más importante, y constituye la central en donde se procesan las transacciones de los todos los clientes afiliados al servicio, los cuales disponen cada uno de su respectiva aplicación Local.exe. La aplicación Host.exe dispone de los medios necesarios para la comunicación con cada una de las entidades emisoras de tarjetas de crédito del mercado.

- La aplicación Emisor.exe es la contraparte del sistema switch de autorizaciones que funciona del lado de cada uno de los emisores o procesadores de tarjetas de crédito y que les permite a estos la comunicación con la aplicación Host.exe.

Ya con estos 3 componentes instalados de forma correcta se procedió a probar su funcionamiento realizando lecturas y escrituras en los archivos de texto EntraXXX.dat y RXXXYYY.dat utilizados por la aplicación Local.exe para el envío y recepción de datos desde y hacia la aplicación Host.exe y de este modo dejar listo el MSSA.

Una vez culminadas las pruebas básicas de funcionamiento de cada módulo por separado, se procedió a realizar las pruebas integrando los 3 módulos que componen el proyecto. A continuación hemos decidido mostrar los resultados arrojados de las pruebas para el caso de la realización de una solicitud de autorización de cobro puesto que para el caso de una anulación o reversa de autorización el proceso que se siguió fue similar. Estas pruebas fueron realizadas tanto con el

MAC funcionando en un emulador como con el MAC funcionando en un equipo Nokia 6230i.

En la figura 4.7 se muestra el formulario que se presenta al usuario para el ingreso de datos de una petición de autorización de cobro con tarjeta de crédito. Los campos que se deben llenar son: Número de tarjeta, código de verificación de la tarjeta, fecha de caducidad de la tarjeta, monto de la transacción, teléfono del cliente y número de meses en caso que sea diferido.

Figura 4.7 Formulario de solicitud de autorización de cobro

Una vez llenados todos los campos de manera correcta, se seleccionó la opción enviar para realizar el envío de datos al servidor. Se realizó primero la prueba en el caso de que todos los datos sean llenados de

manera correcta y nos aseguramos que todos los módulos funcionen bien. El resultado obtenido se muestra en la figura 4.8

Figura 4.8 Respuesta de autorización exitosa

En la figura 4.9 se puede observar el contenido mostrado en la ventana de Log del emulador de Nokia. Entre los datos más importantes se muestran: la tarjeta seleccionada por el usuario para la transacción, el tipo de diferido seleccionado, la trama generada en el MAC previo envío al MAS, y la respuesta del mismo, en este caso el correspondiente número de autorización.

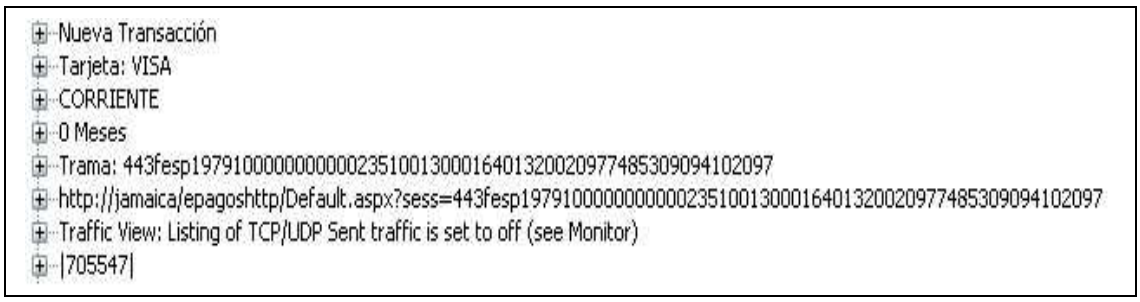


Figura 4.9 Contenido de la ventana de Log del emulador del Nokia SDK para el caso de una petición de autorización

En la figura 4.10 podemos observar el resultado arrojado por esta transacción en el caso de la aplicación Local.exe la cual ya fue descrita anteriormente. En la parte superior de la interface de esta aplicación se listan las tramas de datos enviadas y podemos observar la trama que en el caso de esta prueba se generó para esta transacción y fue enviada al MSSA, mientras que en la parte inferior se listan las tramas de respuesta recibidas. Podemos observar que en la trama retornada en el caso de esta prueba viene incluido el mismo número de autorización que se recibió en la aplicación cliente, lo cual significa que el cobro fue autorizado correctamente.

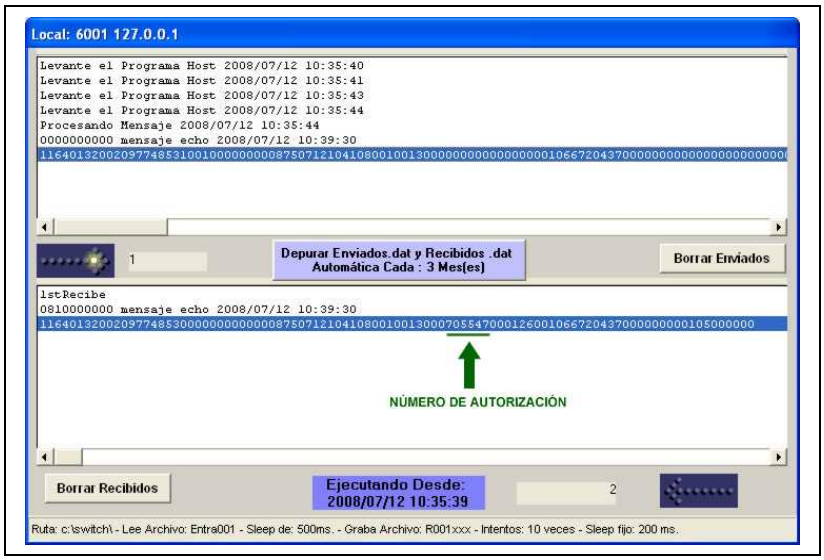


Figura 4.10 Datos de autorización exitosa de cobro en la aplicación Local.exe

En el anexo 7 podemos observar el resultado arrojado por esta transacción en el caso de la interface gráfica de la aplicación Host.exe,

parte del MSSA y la cual en nuestro caso se encuentra instalada en el mismo equipo en donde funciona el MAC. En la parte izquierda donde se encuentra el texto con fondo de color verde se indica los nombres de los locales conectados, mientras que a la derecha se puede ver los nombres de los emisores conectados al servicio, en ambos casos junto con el número de puerto que usan para la conexión y su correspondiente dirección IP. En la parte superior izquierda tenemos una lista de las transacciones requeridas por los clientes del servicio, mientras que en la parte superior derecha podemos ver una lista de las respuestas recibidas desde los emisores para cada una de estas transacciones. Podemos observar entre otros datos: El nombre del emisor que autorizó el cobro, el número de auditoría asignado por el sistema switch y el número de autorización que al igual que en el caso anterior es el mismo que se recibió en el MAC.

En la figura 4.11 se muestra el resultado de esta transacción en la aplicación Emisor.exe, la interface de ésta es similar a la de la aplicación Local.exe, con la diferencia de que en este caso la comunicación entre ésta y la aplicación Host.exe se realiza usando el estándar ISO 8583, es por esta razón que los datos que se presentan en sistema Hexadecimal. Esto es un requerimiento de este estándar.

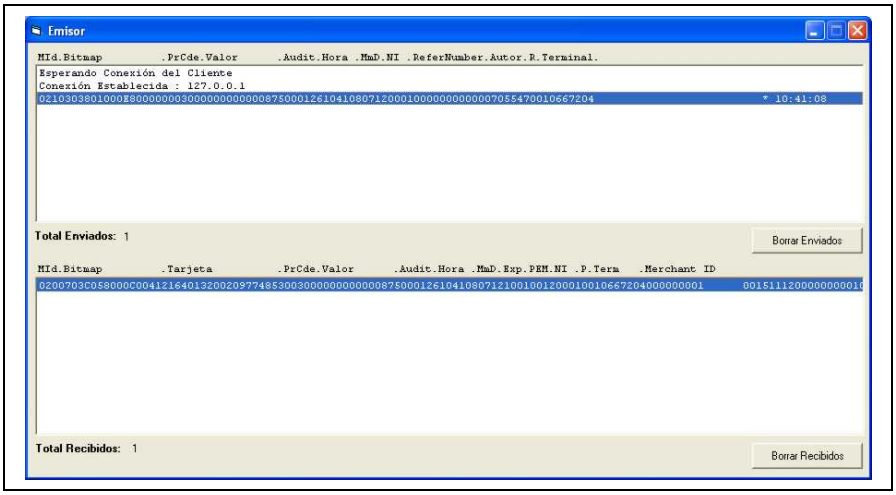


Figura 4.11 Realización de una autorización de cobro en el sistema Emisor.exe

A continuación veremos los resultados de las pruebas en el caso de los escenarios en los que por diversas razones no se obtuvo los resultados esperados.

En primer lugar se probó la validación de datos previo al envío de los mismos al MAS ingresando datos incorrectos como por ejemplo un número de tarjeta de crédito de menos de 14 dígitos o dejando algún campo sin llenar y se intentó enviar estos datos al MAS en cuyo caso el sistema no realizó el envío de los datos y se obtuvo el resultado que se muestra en la figura 4.12 Con esto se comprobó que esta validación funciona correctamente.

Figura 4.12 Mensaje de datos ingresados no válidos

A continuación se procedió a dar de baja a un usuario haciendo uso del administrador de la base de datos de control de usuarios que forma parte del MAS (figura 4.13) para posteriormente intentar realizar una petición de autorización, con lo cual se obtuvo del MAS el mensaje que se muestra en la figura 4.14 con lo cual se comprobó el correcto funcionamiento del proceso de autenticación.

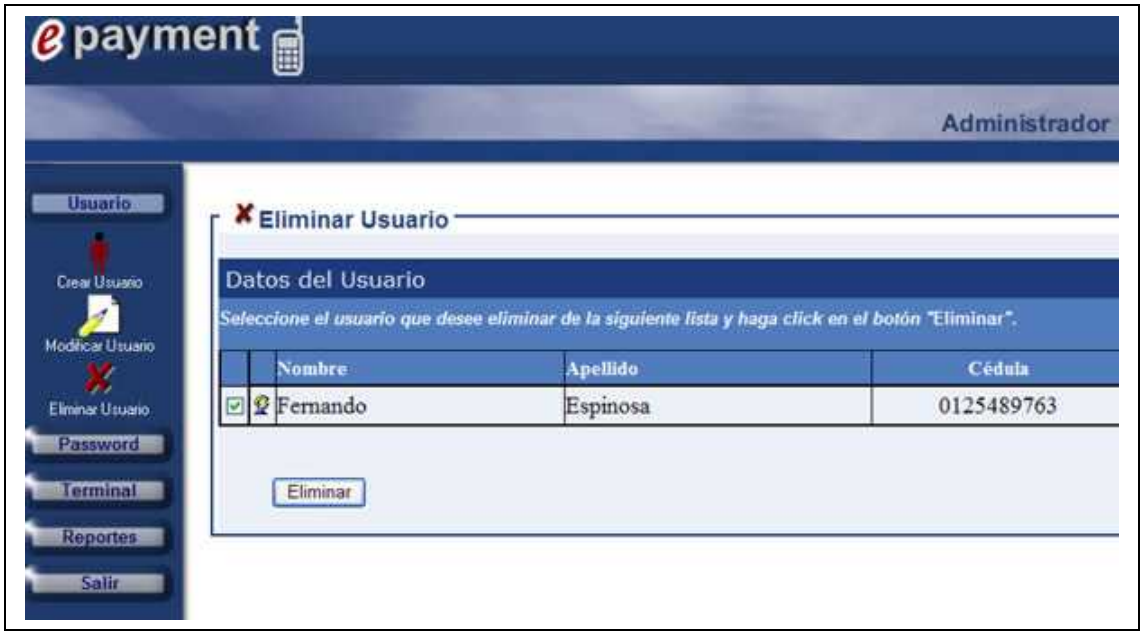


Figura 4.13 Baja de un usuario del servicio en el administrador de la base de datos.

Figura 4.14 Mensaje indicando que los datos no pasaron el proceso de autenticación en el servidor.

La siguiente prueba a realizar consistió en ingresar un número de tarjeta cuyo rango de bines se encuentra fuera del rango del emisor al

cual se solicita la autorización de cobro, en este ejemplo el banco Bolivariano. En la figura del anexo 8 podemos observar que al intentar realizar una autorización de cobro con este número de tarjeta, en la aplicación Host.exe del MSSA se notifica lo sucedido en la lista de transacciones enviadas. Para el caso de la aplicación Local.exe (figura 4.15) se recibe la trama pero esta no es grabada en el archivo de texto RXXXYYYdat. Esto se detecta en el MAS mediante la lectura de este archivo desde la aplicación ASP.NET, la cual responde al MAC con el mensaje que se muestra en la figura 4.16.

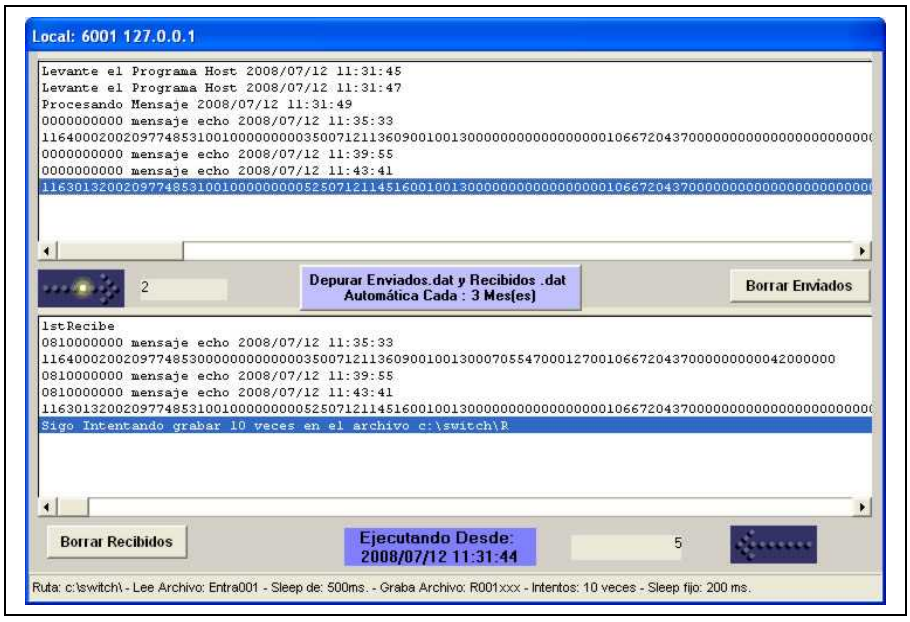


Figura 4.15 Aplicación Local.exe. Número de tarjeta para el que se solicita autorización no corresponde al rango de bines del emisor.

Figura 4.16 Mensaje mostrado al usuario en caso de que el número de tarjeta para el que se solicita autorización no corresponde al rango de bines del emisor.

Códigos de control de errores.

En todo tipo de aplicación, los códigos de control de errores constituyen una parte muy importante. Es necesario realizar una planificación adecuada de los mismos. Estos códigos son muy útiles no solamente para garantizar el buen funcionamiento de la aplicación una vez implementada si no también como lo pudimos comprobar en el caso de nuestro proyecto, en la fase de desarrollo del mismo en lo referente a aspectos como son las depuraciones y pruebas.

Cabe mencionar que al utilizar nuestro proyecto un servidor Web para atender a las solicitudes de los usuarios, este tiene como una de las funciones principales detectar solicitudes incorrectas y tomar las acciones que sean necesarias en estos casos. En una aplicación de este tipo en la que se espera un gran volumen de solicitudes, el número de solicitudes incorrectas puede ser muy alto. Se debe asegurar que el impacto de dichas solicitudes en el rendimiento del servidor sea mínimo y la mejor manera de hacerlo es analizando y rechazando las solicitudes erróneas lo más pronto posible. En el caso de nuestro proyecto se hizo un análisis de cada una de las fases de la arquitectura del mismo, el cual consistió en una comprobación de cada una de las fases para detectar las posibles solicitudes erróneas que puedan surgir y rechazarlas antes de pasar a la fase siguiente. Sin embargo, como se había mencionado anteriormente las limitaciones en cuanto a capacidad de procesamiento de los teléfonos móviles hicieron que se optara por realizar la mayor parte del análisis y rechazo de solicitudes erróneas en el MAS. A pesar de esto, ciertas validaciones sencillas como es por ejemplo el caso de la comprobación de la longitud de los números de tarjeta de crédito ingresados fueron realizadas en el MAC.

Todas las comprobaciones que requieren de acceso al servidor de base de datos que se encuentra en el MAS, requieren inevitablemente de acceso a la red del operador de telefonía celular y por lo tanto representan un impacto tanto en lo referente a rendimiento como en el caso de los costos. Como ejemplo mencionamos el resultado de una prueba realizada en el caso de la solicitud de una reversa o anulación de autorización. En las condiciones en las que se realizaron las pruebas, las cuales fueron descritas anteriormente, se procedió a realizar primero una solicitud de reversa con todos los datos correctos, el resultado fue, que la anulación se realizó correctamente, el tiempo de respuesta fue de aproximadamente 11 segundos y el costo de uso de la red del operador celular, en este caso Porta fue de 0,01 USD. Luego se procedió a realizar una operación de anulación enviando un número de autorización incorrecto, en este caso, debido a que el número de autorización no pasó la comprobación realizada en el servidor de base de datos del MAS, esta solicitud fue rechazada antes de ser enviada a la fase del MSSA y el tiempo de respuesta fue de aproximadamente 7 segundos y el costo al igual que en el caso anterior fue de 0,01 USD. De la realización de estas pruebas pudimos concluir que en el caso de los tiempos de respuesta el realizar el control de la solicitud incorrecta en el MAS ahorró aproximadamente 4 segundos, con esto nos dimos cuenta de la importancia de los códigos

de control de errores, aunque de igual forma influyó de manera clara en el rendimiento, mientras que para el caso de los costos en ambos casos se obtuvo el mismo resultado. Por lo tanto lo más adecuado en estos casos y debido a que en el caso de estas comprobaciones el código de control de solicitudes incorrectas se encuentra en el MAS, es necesario recomendar a los usuarios que traten de minimizar el número de errores cometidos.

El MAS representa la segunda fase en la arquitectura del proyecto y es en esta fase en donde se realizó el análisis más cuidadoso de las posibles solicitudes erróneas para tratar de detectar la mayor cantidad posible de éstas y no dejarlas pasar a la fase del MSSA, ya que al requerir el paso a esta fase de comunicación remota, influye notablemente en los tiempos de respuesta. Como ejemplo tenemos el caso de las solicitudes en las que se detecta un número de tarjeta con formato incorrecto, en cuyo caso se usa en el MAS el algoritmo que valida que el número de tarjeta cumpla con la especificación ISO 2894, de este modo se evita que números de tarjeta que no cumplan con esta especificación sean enviados al MSSA.

CAPÍTULO V

5. PLAN DE NEGOCIOS.

En este capítulo final de nuestra tesis desarrollaremos un mini plan de negocios que permita verificar la rentabilidad del mismo, para ello realizaremos un análisis de los ingresos y egresos durante 3 años. Iniciaremos con un análisis del mercado, posteriormente analizaremos a nuestros competidores, el nicho de mercado al cual aplicaremos,

definiremos además los costos de diseño e implementación, luego unificaremos todo esto para realizar un análisis financiero a tres años.

5.1. Análisis de Mercado.

El incremento de las redes celulares en los últimos años en especial con la llegada de la tecnología GSM/GPRS[14] ha permitido que una gran cantidad de aplicaciones que hasta hace mucho tiempo eran imposible desarrollarse. Conocemos además del amplio grupo de usuarios de tarjetas de crédito (Como se mencionó en el capítulo 1 sección 1.4) lo cual permite complementar mejor la implementación de nuestro proyecto para permitir pagos por medio de teléfonos móviles en zonas con cobertura celular.

Las transacciones de pago también han tenido una evolución, desde la cancelación a través del dinero efectivo, la utilización del cheque, hasta llegar a las tarjetas de crédito [30]. Este último muchas de las ocasiones necesita de una red telefónica convencional para su autorización.

Actualmente tenemos también la posibilidad de comprar vía Internet a través de sitios de comercio electrónico, pero este no es nuestro objetivo (esto lo analizamos en el Capitulo1, sección 1.1). Nuestro enfoque realmente apunta al mercado de las PYMES las cuales no disponen de una infraestructura de telefonía convencional, pero que si posean cobertura celular, con este requisito sustentaremos la infraestructura que hemos propuesto. Para ello pretendemos que las transacciones de pago con tarjetas de crédito utilicen nuestra plataforma. A través de este pequeño análisis pretendemos distinguir nuestro mercado o principales consumidores de nuestro producto.

Nuestra idea es entonces ofrecer un servicio de Pago Electrónico Móvil dirigido principalmente a las PYMES, para ello es importante mencionar cuales son los tipos de negocios que ofrecerán nuestro servicio y quienes harán uso de este servicio respectivamente:

- Tiendas de abarrotes, islas, kioscos de comida, restaurant de las carreteras, locales de parques, cooperativas de taxis, y cualquier otro tipo de negocio que se encuentre registrado en nuestra plataforma y en zona con cobertura celular.

- Toda persona que posea una tarjeta de crédito podrá hacer uso de nuestro servicio.

5.2. Competidores.

En nuestro mercado local no existe un servicio de característica similar como el que planteamos en este proyecto, pero podemos considerar como nuestro principal competidor el pago en efectivo, consideramos que el pago en efectivo es uno de los mas fuertes en los tipos de negocios mencionados en la sección anterior. Sin embargo al existir hoy en día un alto porcentaje de inseguridad en todo el país, nos permite convertir toda esta amenaza en una oportunidad de negocio, ya que bastaría que el cliente presente su tarjeta de crédito e identificación personal para poder procesar su transacción de pago.

Existen también los servicios tradicionales de cobro con tarjeta de crédito a través de empresas procesadoras de este tipo de transacciones como son Datafast, Medianet, que utilizan un dispositivo de Punto de Venta (POS) [31], el local necesita para esto contar con una línea telefónica convencional y estar afiliados a un sistema que le permita hacer o recibir el pago [32]. Nuestra principal ventaja sobre esta solución es que no necesitamos de línea telefónica convencional, solamente cobertura celular y hoy en día este servicio se ha

incrementado a nivel nacional[1]. Esto constituye una ventaja respecto a la plataforma POS.

Ahora vamos a realizar un análisis de nuestra plataforma en cuanto a fortalezas y debilidades, proceso realizado mediante una interacción razonada y crítica de las cualidades de conformación y organización de nuestra empresa. En la siguiente tabla 5.1 podremos el análisis:

FORTALEZAS	**DEBILIDADES**
• Sistema único en el mercado • No requerimos de elementos adicionales de hardware por parte del cliente, excepto del dispositivo móvil • Las transacciones son realizadas en línea. • Nuestros clientes no requieren de un nivel de experiencia en desarrollo de sistemas para realizar las transacciones. • Equipo competitivo y actitud pro-positiva • Conocimiento de la realidad de las transacciones en nuestro país. • Capacidad de organización • Conocimiento de los mercados potenciales.	• Incertidumbre de cómo van a recibir nuestros clientes o posible mercados potenciales nuestro sistema. • Falta de optimización de los canales de comunicación. • Nuevos en el mundo de los negocios, especialmente en este campo.

[1] Cobertura Celular, http://www.movistar.com.ec; http://www.porta.net/110.3576.php; http://www.alegro.com.ec/empresas/CoberturaVentas/tabid/293/Default.aspx

• Disponibilidad de las herramientas necesarias para el desarrollo de actividades. • Apoyo en todos los aspectos de la Empresa SuSistema Cia. Ltda.	

Tabla 5.1 Fortalezas y Debilidades de Nuestra Empresa

Para terminar esta sección analizaremos los posibles agentes que influyen positiva o negativamente en nuestro trabajo y que permiten determinar las posibles oportunidades y amenazas:

OPORTUNIDADES	**AMENAZAS**
• Actualmente no existe aplicación en nuestro medio enfocado a este tipo de transacciones. • Si la solución tiene éxito se pueden crear una gran variedad de servicios adicionales como: transferencias de dinero, pagos en línea, etc. • Incursionar de una manera segura y firme en el mercado de las transacciones de las tarjetas de crédito de este tipo.	• Intervención de otras empresas en nuestro nicho de mercado. • Desinterés de las Pymes para realizar las transacciones por temor al cambio.

Tabla 5.2 Oportunidades y Amenazas de Agentes Externos

5.3 Mercado Potencial.

Para el presente análisis partiremos por conocer nuestros posibles mercados, para ello hemos realizado una pequeña encuesta en lugares en los que pretendemos entrar con fuerza con nuestra infraestructura. El formato de la encuesta la adjuntamos en el Anexo 10.

Después de haber recopilado la información podemos mencionar que podemos considerar como potenciales mercados aquellos lugares alejados de las ciudades, poblaciones pequeñas en las que no existen instituciones bancarias, pero si cobertura celular.

Las islas de los centros comerciales, lugares tradicionales de venta de flores, bisutería, objetos típicos de cada región o población del país, incluso en las cooperativas de taxis, como mercados deben considerarse como nuestro objetivo.

En la figura 5.1 podemos ver un mercado potencial, en el anexo 11 mostramos algunos de nuestros mercados potenciales.

Figura 5.1 Mercado Potencial: Parque de los Jipis (Cuenca)

Para segmentar nuestro mercado, también hemos realizado breve análisis del número actual de tarjetahabientes al que podemos abordar. En primer lugar debemos mencionar que nuestro servicio esta dirigido a cualquier tipo de tarjetas de crédito pero por lo que explicaremos más adelante tenemos un mercado aún más interesante.

En el capítulo 1.4 habíamos mencionado el gran universo de personas que poseen tarjetas de crédito, de esos casi dos millones de tarjeta habientes reduciremos nuestro objetivo a la tarjeta de Crédito Cuotafácil, que es la de mayor frecuencia entre los concurrentes a nuestros mercados con un mercado que actualmente abarca los

298.275 tarjeta habientes y con consumos entre los 25 y 180 dólares mensuales por cliente.

Con esto podemos concluir que nuestro mercado potencial está en las Pymes principalmente por que serán nuestra entrada a los clientes y es a través de ellos que pretendemos llegar al mercado con mayor segmento de tarjetas de crédito del país

5.4. Costos de Diseño.

Durante la fase de análisis de diseño de nuestra aplicación se realizaron varias tareas con el fin de recopilar la suficiente información que permita el desarrollo de nuestra aplicación. Esta etapa resultó ser una de las más críticas de nuestro proyecto, especialmente por la necesidad de conocer y buscar información acerca del intercambio de mensajes entre la aplicación y el banco.

Una vez conocida una de las posibles formas de realizar transacciones de pago con tarjeta de crédito se nos ofreció la posibilidad realizar una pasantía en la ciudad de Quito en la cual se adquirió un poco mas de conocimiento en este tipo de plataformas de pago.

Luego del análisis de nuestra plataforma, proseguimos con el diseño de la misma, la cual fue enfocada en cuatro etapas: diseño de datos, arquitectura de la aplicación, diseño de interfaz y el diseño de procedimientos. A continuación vamos a detallar los costos que involucraron la fase de análisis y diseño, ver tabla 5.3:

COSTOS DE ANALISIS		
DESCRIPCION	**TIEMPO**	**VALOR**
ASESORIA	3 DIAS	$ 300
2 ANALISTAS	1 MES	$ 700
INTERNET	2 MESES	$ 34
COSTOS DE DISEÑO		
2 DISEÑADORES	1 MES	$ 700
TOTAL		$ 1.734

Tabla 5.3 Costos de Diseño

Los rubros descritos anteriormente son los necesarios para el inicio de nuestro proyecto, claro está que aun no se especifican lo referente a la

oficina, equipos de computación y servicios básicos, estos serán descritos en el enunciado 5.6 de este capítulo.

5.5. Costos de Implementación.

Estos costos serán descritos a continuación, para ello nos referiremos a cada uno de ellos.

- Programadores: tendremos 2 programadores, los cuales según los costos del mercado tienen un valor de $350 mensuales, el mismo que tendrá una duración de 3 meses, adicional un mes de pruebas.

- Contratación del servidor dedicado, las razones por las que utilizamos este tipo de servicio está explicado en el capítulo 3 sección 3.1, según la cotización de precios tenemos la siguiente oferta que creemos ser la mejor, la cual la podemos ver en la siguiente figura 5.2

Figura 5.2 Cotización del Servidor Dedicado para el Primer año [33]

- Por motivos de incremento de los usuarios y las transacciones debemos ir mejorando nuestro servidor, por esa razón para el segundo año y tercero se alquilara mejores servidores para ello realizamos las cotizaciones respectivas, las mismas que podemos ver en las grafica 5.3 para el segundo y tercer año.

Grafica 5.3 Cotizaciones de Servidores para el Segundo y Tercer año [33]

5.6. Recursos Necesarios

Los recursos necesarios para poder dar inicio a nuestro proyecto, con el análisis, diseño e implementación son:

← * Oficina.

← * Servicios Básicos (Agua, Luz y Teléfono).

← * Dos Computadoras.

← * Internet.

←

Los valores de algunos de estos, por ser fijos y adquiridos una sola vez los haremos contar en nuestro análisis financiero en el año cero o inicio.

El personal a contratar será de acuerdo a las necesidades en cada período o año, inicialmente contaremos con un técnico – programador, el mismo que será el encargado de llevar a cabo las instalaciones de la aplicación en el celular y dar asesoramiento o servicio al cliente, una secretaria- contadora y un vendedor del servicio.

Para nuestro segundo año, incrementaremos el personal con un técnico-programador adicional, un vendedor y contadora.

5.7. Análisis financiero a 3 años.

En esta parte de nuestro proyecto mencionaremos todo lo referente a los gastos, ingresos y la utilidad de nuestro proyecto, el cual se ha proyectado que en tres años tenga la suficiente autonomía y que las perspectivas de crecimiento sean buenas al final del mismo considerando además las respectivas tasas de crecimiento, depreciación de los equipos, cambios en los equipos de servidores, etc.

Para un mejor entendimiento vamos a ver el anexo 12, o en la tabla 5.4 en donde están descritos cada uno de los ingresos, egresos y la utilidad para el periodo de 3 años.

PAGO ELECTRONICO A TRAVES DE DISPOSITIVOS MOVILES

ANALISIS FINANCIERO A TRES AÑOS

RUBRO	AÑO 0	AÑO 1	AÑO2	AÑO 3
EGRESOS				
Gastos Iníciales				
Adquisición de dos Computadores	1500			
Instalación de Internet	35			
Costos de Análisis y Diseño	1734			
Costo de Desarrollo e Implementación	3005			
Oficina	720			
Servicios Básicos	120			
TOTAL GASTOS INICIALES	7114			

Gastos Anuales				
Oficina		1440	1584	2400
Servicios Básicos		240	360	960
Pago por Costos de Servidor Dedicado		2460	3612	6000
Pago a SuSistema		20099	43854	87555
Internet		408	516	912
Técnico Programador		4200	12000	19200
Secretaria			3600	6000
Vendedor		2400	8400	7200
Contador			7200	9600
TOTAL		**31247**	**81126**	139827
INGRESOS				
Rubro por cobro de Transacciones a $0,25 c/u		33498	98902	218888
TOTAL INGRESOS - EGRESOS		2251	17776	79061

TABLA 5.4 ANALISIS FINANCIERO A 3 AÑOS DEL PROYECTO

Cabe mencionar que se ha llegado a un acuerdo con la empresa SuSistema Cia. Ltda., para lo cual se nos cobrará por cada transacción $0.15 si el numero de transacciones están por debajo de las 25000 mensuales, al sobrepasar este número de transacciones mensuales tendremos que cancelar $0.10 por transacción a SuSistema Cia. Ltda., por esta razón se ha considerado cobrar el valor de $0.25 por transacción a nuestros clientes, en la siguiente tabla 5.5 podremos ver los costos y beneficios por transacción.

COSTOS DE TRANSACCION MENSUALES

TRANSACCIONES POR MES	PAGO SuSistema Cia. Ltda.	BENEFICIO
MENOR A 25000 transacciones	$0.15 por transacción	$0.10 por transacción

MAYOR A 25000 transacciones	$0.10 por transacción	$0.15 por transacción

TABLA 5.5 COSTOS Y BENEFICIOS POR TRANSACCION

Nuestro objetivo inicial es colocar nuestro sistema en unos 50 usuarios como mínimo para lo cual pretendemos contar con 100 a 110 transacciones mensuales por cada usuario, para el siguiente mes pensamos se pueden incrementar en un 20% la cantidad de usuarios, llegando a transacciones de 46 mensuales por usuario, obteniendo un total de transacciones mensuales de 5500 para el segundo mes, al finalizar el año, tenemos pensado tener unos 151 clientes realizando unas 120 transacciones cada uno mensualmente finalizando el último mes del año con un total de 18146 transacciones mensuales. Con más detalles podemos observar en el anexo 13 las transacciones realizadas durante el primer año de vida de nuestro proyecto.

En el anexo 14 y 15 constan las transacciones realizadas durante el segundo y tercer año respectivamente.

Ahora con los datos obtenidos podemos realizar un análisis económico que permita tener una referencia económica mediante el cálculo de la Tasa de Rendimiento Interno (TIR) y nuestro valor actual neto (VAN). En la siguiente tabla 5.6 podremos ver los resultados, para ello vamos

a tomar la tasa de interés activa comercial del 14,76 %, según el Banco Central del Ecuador, los flujos de fondos son los ingresos obtenidos en cada año, eso lo podemos ver en el Anexo 12:

AÑO	FLUJO DE FONDOS
0	-7114
1	2251
2	17776
3	79061

TIR	**173%**
VAN	**$ 60.656**

Tabla 5.6: Calculo del TIR y VAN de nuestro proyecto.

Como nuestro VAN es positivo podemos decir que nuestra inversión es muy aceptable realizarla, y al tener un TIR del 173% mucho mayor que el 14,76% de la tasa efectiva, es realmente muy bueno realizarlo, es decir $8,53 de VAN, por cada dólar invertido.

CONCLUSIONES

A través del presente proyecto y considerando los resultados obtenidos consideramos que la implementación del proyecto es viable, lo cual es demostrado a través de los resultados obtenidos tanto en la parte económica como a la solución de la problemática planteada, se concluye entonces que:

- Se cumplió con el objetivo planteado, de brindar de servicio de pago electrónico a través del móvil o celular desde cualquier lugar geográfico del país que posea cobertura celular, dirigido especialmente a los PYMES.
- El VAN del proyecto resulto ser positivo de USD 61545 y una tasa de retorno TIR del 179%, por lo que se demuestra que nuestro proyecto es rentable desde el punto de vista financiero.
- Se mejora el nivel de competitividad de los PYMES, con el resto de negocios involucrando a los estos en la modernidad de realizar las transacciones, brindándoles un servicio acorde a los avances tecnológicos.
- Se brinda un sistema que esta de acuerdo a los avances tecnológicos, demostrando que la Escuela Superior Politécnica del Litoral se

preocupa por las necesidades de su comunidad en brindar soluciones a problemáticas de nuestro entorno.

RECOMENDACIONES

Por lo anteriormente expuesto y para mejorar la realización de este tipo de trabajos investigativos, es menester nuestro recomendar algunos puntos que mejorarían la realización de los mismos:

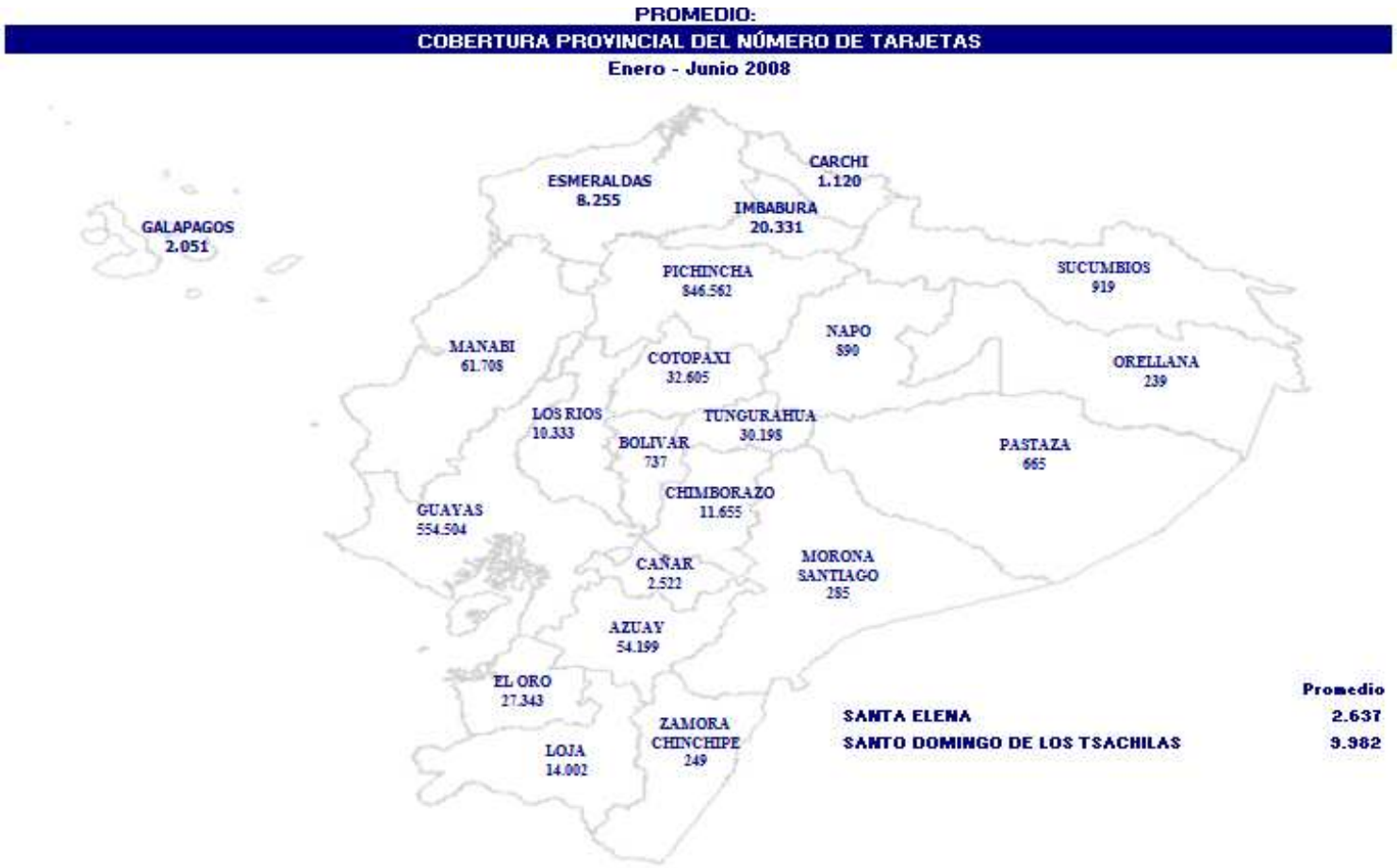
- Fortalecer los lazos de cooperación entre la Universidad y las empresas del sector productivo, para que faciliten la información requerida por los estudiantes para la realización de sus proyectos.
- Permitir que los estudiantes en sus tópicos de graduación, especialmente en la segunda parte del mismo, dedicarse por completo a la realización del proyecto asignado, para que puedan graduarse dentro de lo previsto.
- Exponer a los PYMES, las ventajas de sus negocios en la implementación de este proyecto, especialmente los beneficios que ellos obtendrían.
- Continuar incentivando a la investigación a través del dictado de tópicos y la formación de grupos de investigación como el Grupo de

Investigación GICOM de la Escuela Superior Politécnica del Litoral (ESPOL).

- Permitir que los estudiantes continúen con el dictado de su tópico, aun sin que algunos estudiantes no cancelen los haberes en la segunda parte del mismo, para ayudarlos en la finalización del mismo dentro de los tiempos previstos.

A N E X O S

ANEXO 1: COBERTURA PROVINCIAL DE NÙMERO DE TARJETAS



Fuente: Estructuras integradas de datos / Subgerencia de servicios informáticos / Central de Riesgos (SCR), pagina Web https://www.superban.gov.ec/pages/c_tarjetas_habientes.htm
Elaboración: Dirección General de Estudios y Estadísticas/ Dirección de Estadísticas
Fecha de actualización y/o reproceso: 6 de mayo de 2008

ANEXO 2: COBERTURA DE TARJETAS POR CLASE

NUMERO DE TARJETAS POR CLASE
Del 1 al 30 de junio de 2008

INSTITUCIÓN	TARJETA	INTERNACIONAL		NACIONAL		TOTAL
		CORPORATIVA	INDIVIDUAL	CORPORATIVA	INDIVIDUAL	
BANCO DE GUAYAQUIL	AMERICAN EXPRESS	555	155.632			156.187
	Total AMERICAN EXPRESS	**555**	**155.632**	**-**	**-**	**156.187**
SOCIEDAD FINANCIERA DINERS CLUB	DINERS	2.357	91.465	1.069	138.526	233.417
	Total DINERS	**2.357**	**91.465**	**1.069**	**138.526**	**233.417**
BANCO BOLIVARIANO	MASTERCARD		5.248			5.248
BANCO DE GUAYAQUIL	MASTERCARD	1	19.571			19.572
BANCO DEL AUSTRO	MASTERCARD		9.467			9.467
BANCO DEL PACIFICO	MASTERCARD		65.715		383	66.098
BANCO DEL PICHINCHA	MASTERCARD	103	37.391		6.323	43.817
BANCO INTERNACIONAL	MASTERCARD	36	7.048			7.084
BANCO PRODUBANCO	MASTERCARD	463	45.051		232	45.746
MUTUALISTA PICHINCHA	MASTERCARD		13.323	1	55	13.379
PACIFICARD	MASTERCARD	523	94.469		11.273	106.265
	Total MASTERCARD					

		1.126	**297.283**	**1**	**18.266**	**316.676**
BANCO AMAZONAS	VISA	31	1.815		1	1.847
BANCO BOLIVARIANO	VISA	407	36.800	1	9.143	46.351
BANCO COMERCIAL DE MANABI	VISA		535		495	1.030
BANCO DE GUAYAQUIL	VISA	5.307	42.307			47.614
BANCO DE LOJA	VISA		3.404			3.404
BANCO DE MACHALA	VISA	32	20.083		6.860	26.975
BANCO DEL AUSTRO	VISA		27.512		11.166	38.678
BANCO DEL PACIFICO	VISA	128	83.615	20	18.949	102.712
BANCO DEL PICHINCHA	VISA	586	75.390		157.343	233.319
BANCO GENERAL RUMIÑAHUI	VISA		3.224		13.342	16.566
BANCO GUAYAQUIL BANK TRUST	VISA	1	75			76
BANCO INTERNACIONAL	VISA	308	23.898			24.206
BANCO MM JARAMILLO ARTEAGA	VISA		4.487			4.487
BANCO PRODUBANCO	VISA	71	1.938		10.869	12.878
BANCO TERRITORIAL S.A.	VISA		2.731			2.731
BANCO UNIBANCO	VISA		743			743
MUTUALISTA AZUAY	VISA		767		128	895
	Total VISA	**6.871**	**329.324**	**21**	**228.296**	**564.512**
BANCO DE LOS ANDES C.A., EN	CREDIANDES					

LIQUIDACION					3.685	3.685
	Total CREDIANDES	**-**	**-**	**-**	**3.685**	**3.685**
BANCO SOLIDARIO	MI SOCIA				204	204
	Total MI SOCIA	**-**	**-**	**-**	**204**	**204**
BANCO TERRITORIAL	CREDITO SI				118.780	118.780
	Total CREDITO SI	**-**	**-**	**-**	**118.780**	**118.780**
BANCO UNIBANCO	CUOTAFACIL				298.275	298.275
	Total CUOTAFACIL	**-**	**-**	**-**	**298.275**	**298.275**
BANCO INTERNACIONAL	ROSE				6.826	6.826
	Total ROSE	**-**	**-**	**-**	**6.826**	**6.826**
	TOTAL GENERAL	**10.909**	**873.704**	**1.091**	**812.858**	**1.698.562**

Fuente: Estructuras integradas de datos / Subgerencia de servicios informaticos / Central de Riesgos (SCR)
Elaboración: Dirección Nacional de Estudios y Estadísticas/ Dirección de Estadísticas/ CGV
Fecha de actualización y/o reproceso: 6 de agosto de 2008

ANEXO 3 Formato de trama utilizada para el envío de los datos de una transacción a la aplicación servidor.

DESCRIPCION	TIPO	BYTES	DATOS
Longitud del nombre de usuario	Numérico	1	Longitud del nombre de usuario
Longitud de la contraseña	Numérico	1	Longitud de la contraseña del usuario
Longitud del número Id del equipo	Numérico	1	Longitud del número Id del equipo
Nombre de usuario	Alfanumérico	6.. 8	El nombre de usuario con el que se autentica al comerciante en la base de datos de la aplicación servidor.
Contraseña de usuario	Alfanumérico	6.. 8	La contraseña con la que se autentica al comerciante en la base de datos de la aplicación servidor.
Id del equipo	Numérico	6.. 8	El número con el que se autentica al equipo del comerciante en la base de datos de la aplicación servidor.
Valor	Numérico	12	Valor total de la transacción, 10 bytes para enteros y 2 para decimales
Fecha de Caducidad de la tarjeta	Numérico	4	Formato AAMM año y mes
Tipo de Diferido	Numérico	2	Tipo de Diferido (Corriente, Con Interés o Sin Intereses), si es corriente va 30, de lo contrario va el 3 y el tipo de diferido asignado por el emisor.
Número de Meses	Numérico	2	Si es corriente va ceros, caso contrario el número de meses pactado con el emisor
Longitud del número de tarjeta	Numérico	2	Longitud del número de tarjeta
Número de Tarjeta	Numérico	14..15..16	Diners tiene 14, Amex tiene 15 y Visa 16.

ANEXO 4 Formato de trama utilizada desde el servidor para la comunicación con el switch.

DESCRIPCION	TIPO	BYTES	DATOS
Transacción	Numérico	1	1 Requerimiento de Autorización 2 Aviso de Autorización 3 Reversa de Autorización
Longitud de la Tarjeta	Numérico	2	Longitud de la Tarjeta
Número de Tarjeta	Numérico	14..15..16	Diners tiene 14, Amex tiene 15 y debe venir ajustado con 0 a la derecha completando los 16 y Visa 16. Cuando la tarjeta no fue leída la banda magnética, caso contrario va ceros
Fecha de Caducidad	Numérico	4	Formato AAMM año y mes Cuando la tarjeta no fue leída la banda magnética, caso contrario va 4 ceros
Valor	Numérico	12	Valor total de la transacción, 10 enteros y dos decimales
Fecha de transacción	Numérico	4	Formato MMDD mes y día
Hora de transacción	Numérico	6	Formato HHMMSS hora, minutos y segundos
Código de Establecimiento	Numérico	3	Código asignado a ese establecimiento
Número de Caja	Numérico	3	Número de caja de ese establecimiento
Tipo de Diferido	Numérico	2	Tipo de Diferido (Corriente, Con Interés o Sin Intereses), si es corriente va 30, de lo contrario va el 3 y el tipo de diferido asignado por el emisor.
Número de Meses	Numérico	2	Si es corriente va ceros, caso contrario el número de meses pactado con el

			emisor
Número de Autorización	Numérico	6	Número de autorización enviado por el emisor para la transacción 1, en la transacción 2 el número dado por el local y para la transacción 3 la autorización enviada en la transacción inicial 1
Número de Auditoría	Numérico	6	Número asignado por el Sistema SWITCH en caso de transacción 1-2, para la transacción 3 el número asignado en la transacción inicial 1
Código de Respuesta	Numérico	2	Código de respuesta enviado por el emisor, si es 0 viene el número de autorización y significa aprobada, caso contrario es negada.
Terminal Id	Numérico	8	Código de Terminal asignado por el Emisor
Longitud del TrackII	Numérico	2	Longitud del trackII, cuando la tarjeta ha sido leída la banda magnética
TrackII	Numérico	..37	El dato del trackII leído de la banda
IVA	Numérico	12	El valor del IVA, 10 enteros y 2 decimales
Lote	Numérico	6	El número del lote asignado por el emisor

Formato de Archivo EntraXXX.dat. El Mensaje de respuesta tendrá el mismo formato, más los datos adicionales de fecha y hora de respuesta del Emisor y será grabado en el Servidor en el archivo RXXXYYY.DAT que también tendrá un solo registro, donde XXX significa el código del local y YYY la caja que proceso el mensaje.

ANEXO 5 ESQUEMA DE COMUNICACIÓN VIA SOCKET CON ENLACE DEDICADO

Requerimiento de Mensaje desde la caja

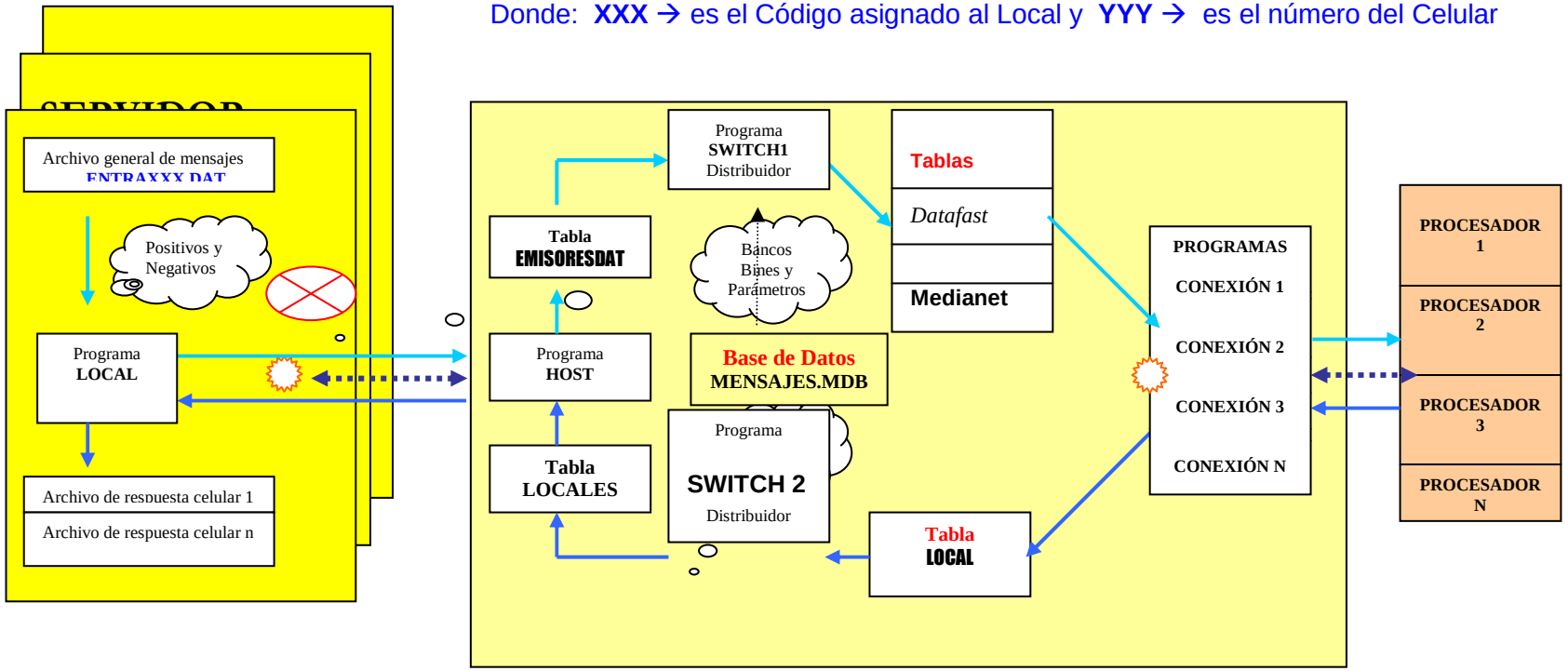
Respuesta de mensaje

Por Pérdida de Comunicación

Socket de Envío y de Recepción

Auditoria de Sockets

Donde: **XXX** → es el Código asignado al Local y **YYY** → es el número del Celular



ANEXO 6 SWITCH DE AUTORIZACIONES

SWITCH DE AUTORIZACIONES

El proceso de envío y recepción de mensajes se lo realiza mediante sockets usando el protocolo de comunicaciones TCP/IP.

VENTAJAS

- Tiempos de Respuesta Inmediatos (2 a 8 segundos)
- Eliminación de Fraudes
- Se elimina el uso de teléfonos y por ende de los puntos de venta (POS)
- Garantía en el cobro de Vouchers a los Emisores de Tarjetas de Crédito
- Se evita el uso de Boletines
- Imagen en el Servicio
- Contará con información apropiada en forma ágil y oportuna
- Los costos son de ahora, pero los beneficios son actuales y futuros

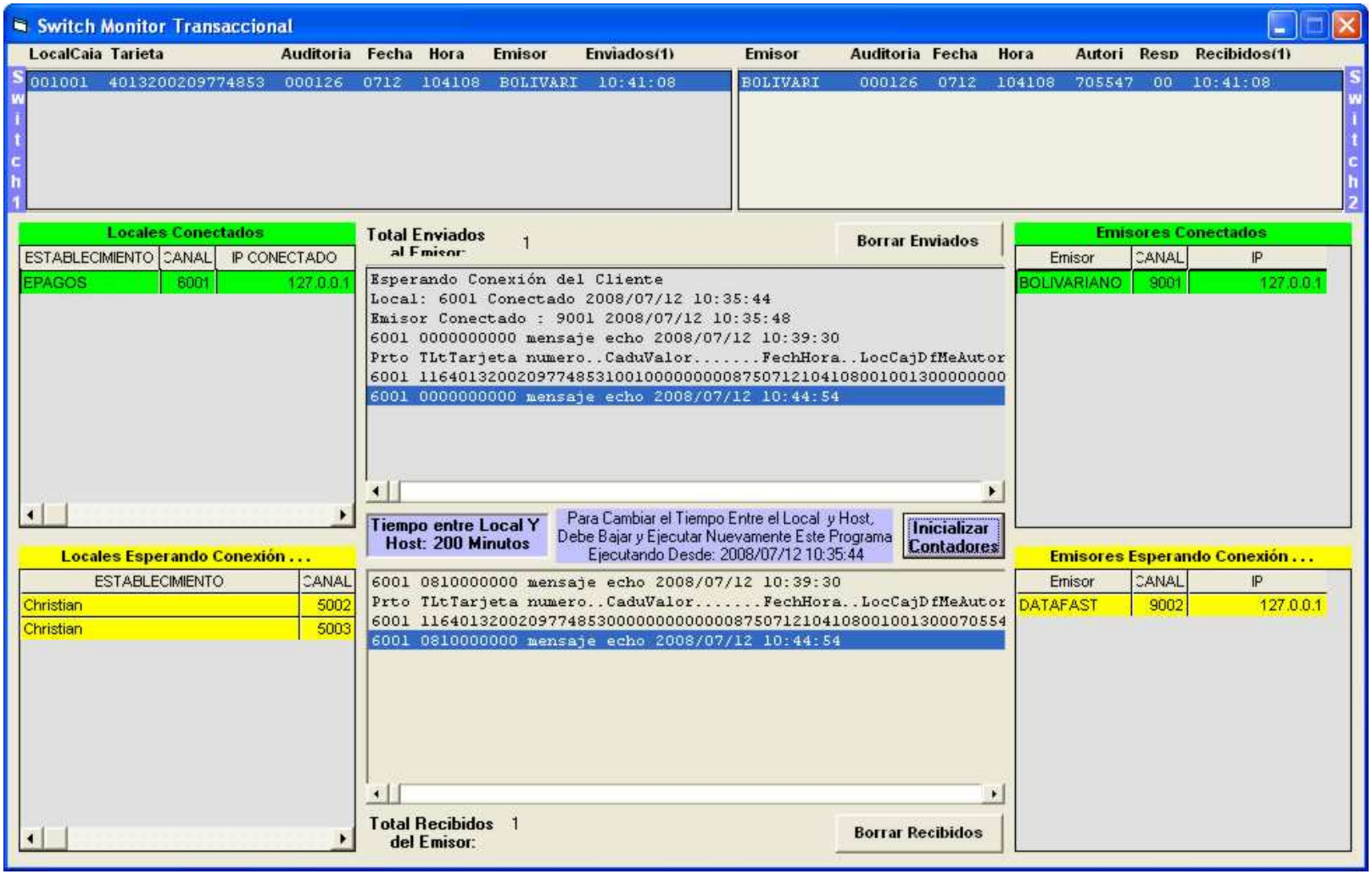
Permite trabajar en varias plataformas como:

- ✓ Windows (95, 98, Milenium, Xp, Nt, Server 2.00x)
- ✓ 4690
- ✓ Novell
- ✓ Linux

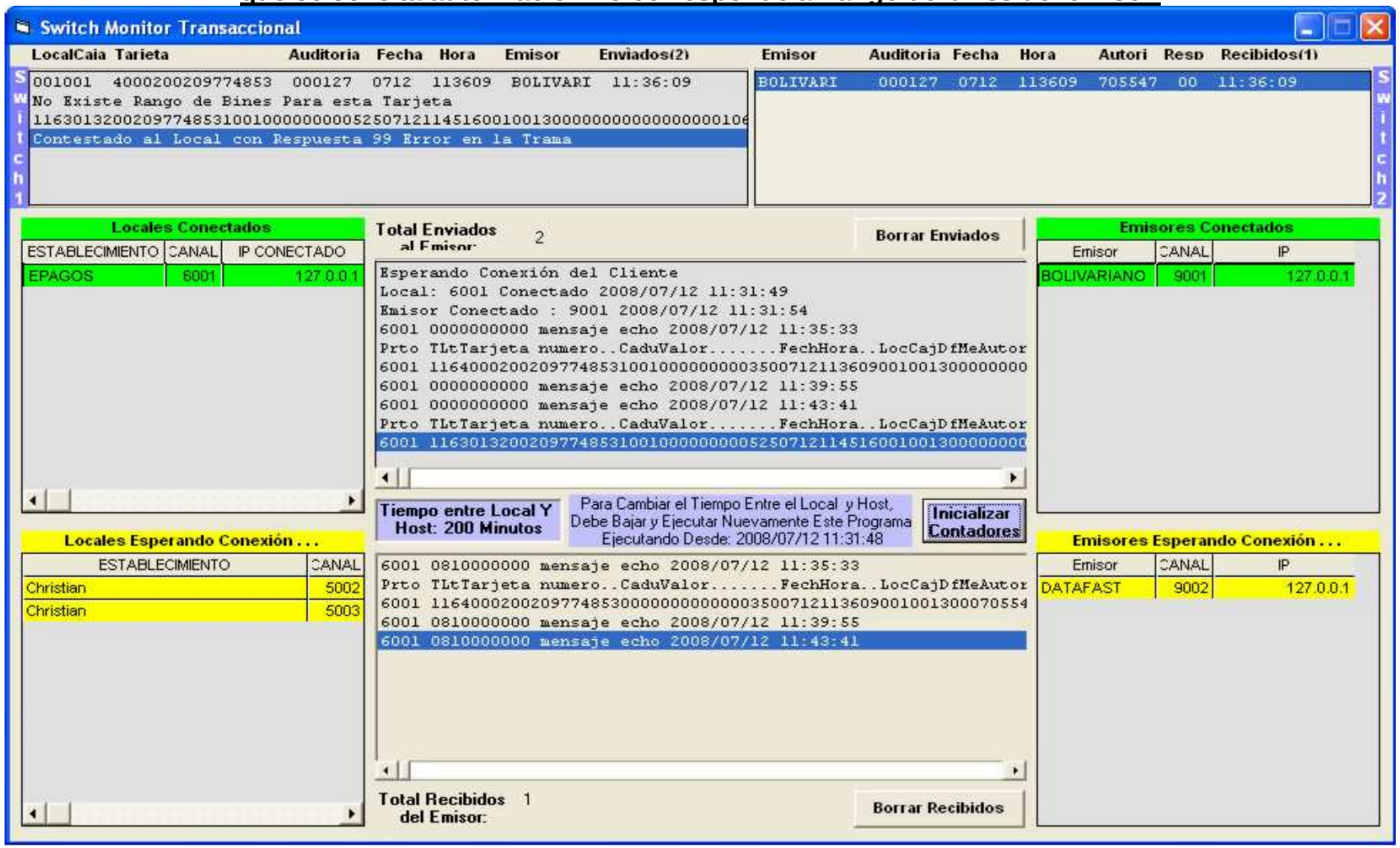
El Sistema contempla lo siguiente:

- ♦ Sockets Cliente y Servidor por cada Local en el Servidor del Local (Parametrizado)
- ♦ Administrador de la Aplicación en el Host
- ♦ Socket Cliente /Servidor para cada Emisor en el Host

ANEXO 7 Realización de una autorización de cobro exitosa en el monitor transaccional del sistema Switch de autorizaciones



ANEXO 8 Mensaje mostrado en el monitor transaccional del sistema switch en caso de que el número de tarjeta para el que se solicita autorización no corresponde al rango de bines del emisor.



ANEXO 9 Descripcion de cada una de las Principales tablas de la base de datos.

La base de datos del servidor tiene las siguientes tablas principales:

- ep_usuario
- ep_terminal
- ep_transaccion

La tabla ep_usuario almacenará los datos de los comerciantes que adquieran el servicio. Estos son los usuarios de las aplicaciones cliente o terminales instalados en los teléfonos móviles. Esta tabla consta de los siguientes campos:

- **us_id:** Es de tipo INT y se usa para almacenar un identificador numérico del usuario que es manejado de forma interna por el servidor de control.
- **us_nombre:** Es de tipo VARCHAR, tiene una longitud máxima de 30 caracteres y se usa para almacenar los nombres de un usuario del servicio.
- **us_apellido:** Es de tipo VARCHAR, tiene una longitud máxima de 30 caracteres y se usa para almacenar los apellidos de un usuario del servicio.
- **us_user:** Es de tipo VARCHAR, tiene una longitud de 12 caracteres y se utiliza para almacenar el identificador del usuario requerido para la autenticación en el servidor.

- **us_password:** Es de tipo VARCHAR, tiene una longitud de 16 caracteres y se usa para almacenar la contraseña del usuario requerida para la autenticación en el servidor.

- **us_ruc:** Es de tipo VARCHAR, tiene longitud de 12 caracteres y se usa para almacenar el número de registro único de contribuyente del usuario del servicio.

- **us_telefono:** Es de tipo VARCHAR y tiene longitud de 9 caracteres, se usa para almacenar el número telefónico de contacto del usuario.

- **us_cedula:** Es de tipo VARCHAR y tiene longitud de 10 caracteres, se usa para almacenar el número de cédula del usuario.

- **us_tipo:** Es de tipo CHAR y tiene longitud de 1 carácter, se usa para almacenar el tipo de usuario: se utiliza ‘u’ para usuarios del servicio y ‘a’ para el caso de usuarios administradores del servidor.

- **us_email:** Es de tipo VARCHAR y tiene longitud de 50 caracteres, se usa para almacenar la dirección de correo electrónico del usuario.

- **us_direccion:** Es de tipo VARCHAR, tiene una longitud de 50 caracteres y se usa para almacenar la dirección del usuario que adquiere el servicio.

- **us_estado:** Es de tipo CHAR, tiene una longitud de 1 carácter y se usa para almacenar el estado del usuario. Para el caso de usuarios activos del sistema se registra el carácter ‘a’ mientras que para usuarios inactivos se registra el carácter ‘i’.

La tabla ep_terminal almacenará los datos de los equipos celulares de los usuarios del servicio en los que se encuentre instalada la aplicación cliente. La cual permite al equipo funcionar como un terminal del sistema. Está compuesta por los siguientes campos:

- **tm_id:** Es de tipo INT y almacena el número identificador del terminal en el servidor.

- **us_id:** Es la clave foránea que almacena el número identificador del usuario al que pertenece el terminal.

- **tm_marca:** Es de tipo VARCHAR, tiene una longitud de 20 caracteres y almacena la marca del equipo celular en donde se encuentra instalada la aplicación.

- **tm_modelo:** Es de tipo VARCHAR, tiene una longitud de 20 caracteres y almacena el modelo del equipo celular en donde se encuentra instalada la aplicación.

- **tm_imei:** Es de tipo VARCHAR, tiene una longitud de 20 caracteres y almacena el código identificador del equipo celular en donde se encuentra instalada la aplicación.

- **tm_estado:** Es de tipo CHAR, tiene una longitud de 1 carácter y se usa para almacenar el estado del terminal. Para el caso de terminales activos del sistema se registra el carácter ‘a’ mientras que para el caso de inactivos se registra el carácter ‘i’.

La tabla ep_transacciones registrará las transacciones realizadas por los usuarios. Estas transacciones podrán ser de dos tipos: petición de autorización de un cobro o anulación de una autorización otorgada previamente. Cabe mencionar que se registrarán también en esta tabla los intentos de transacciones que por diferentes motivos fueron fallidos. Está compuesta por los siguientes campos:

- **tr_id:** Es de tipo INT y almacena un número identificador de la transacción que será utilizado de manera interna por el servidor.
- **tr_tipo:** Es de tipo CHAR, tiene una longitud de 1 carácter y determina el tipo de transacción. Este puede tomar dos valores: ‘1’ si se trata de un requerimiento de autorización y ‘3’ en el caso de una anulación de autorización’.
- **us_id:** Es la clave foránea que almacena el número identificador del usuario que realizó la transacción.
- **tr_diferido:** Es de tipo CHAR y tiene una longitud de 2 caracteres. Se usa para almacenar el tipo de Diferido en el caso de una petición de autorización, si es corriente va ‘30’, de lo contrario va el ‘3’ y el tipo de diferido asignado por el emisor.
- **tr_meses:** Es de tipo INT y tiene una longitud de 2 caracteres. En el caso de tipo corriente se almacena ‘00’ caso contrario el número de meses pactado con el emisor.

- **tr_cel_cliente:** Es de tipo VARCHAR y tiene una longitud de 12 caracteres. Se usa para almacenar un número telefónico de contacto del cliente al cual el usuario realizó el cobro.
- **tr_fecha:** Es de tipo DATETIME y se usa para almacenar la fecha y hora en que se realizó la transacción.
- **tr_resultado:** Es de tipo CHAR y tiene una longitud de 1 carácter. Se usa para almacenar un número que determina si la transacción se completó o no de manera correcta. En el caso de una transacción exitosa se almacena el valor '1', en el caso de que el sistema Switch negó la solicitud por algún motivo se almacena '2' y en el caso de que por algún motivo no hubo respuesta del sistema Switch se almacena '3'.
- **tr_autorizacion:** Es de tipo VARCHAR y tiene una longitud de 6 caracteres. Se usa para almacenar el código de autorización enviado por el emisor al sistema Switch de Autorizaciones en respuesta a una petición de autorización que se realizó de manera exitosa.
- **tr_auditoria:** Es de tipo VARCHAR y tiene una longitud de 6 caracteres. Se usa para almacenar un número identificador de transacción asignado por el sistema Switch de Autorizaciones.
- **tr_monto:** Es de tipo FLOAT. Se usa para almacenar el monto de la transacción.

- **tr_iva:** Es de tipo FLOAT. Se usa para almacenar el valor del IVA para una transacción.
- **tr_comision:** Es de tipo FLOAT. Se usa para almacenar el valor de comisión cobrado por transacción a los usuarios del servicio.
- **tr_estado:** Es de tipo CHAR, tiene una longitud de 1 carácter y se usa para borrado lógico en la base de datos. Para el caso de transacciones activas se registra el carácter ‘a’ mientras que para el caso de eliminadas se registra el carácter ‘i’.

ANEXO 10 Formato de Encuesta realizada a Mercados Potenciales

E N C U E S T A

NOMBRE:_____
SEXO: MASCULINO _____ FEMENINO _____
EDAD:_____

1. **¿Las ventas diarias en promedio entre que valores fluctúan?**

   _____
   _

2. **Les piden cobrar por medio de tarjeta de crédito.**

   _____
   _

3. **¿Estaría usted dispuesto(a) a realizar el cobro por medio de tarjeta de cédito?**

   SI _____ NO_____

4. **¿Cuando vienen sus clientes le han indicado que tarjeta de crédito utilizan.?**

   SI _____ NO _____

   Si su respuesta es afirmativa por favor indicar

   VISA_____ MASTERCARD_____ CUOTAFACIL_____

5. **¿Cuál es el promedio que compran cada cliente.?**

   _____
   __

6. **Por qué no ofrecen el servicio de cobro por tarjeta de crédito.**

   Falta de conocimiento del servicio ____

   No tienen la infraestructura necesaria _____

   Dispositivos caros _____

   Otros:_____

7. **¿Si le ofrecíamos este servicio a través de su celular usted estaría dispuesto a realizarlo?**

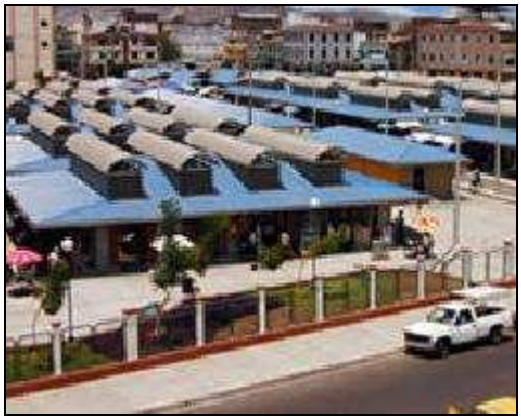
   SI _____ NO _____

8. **¿Cuánto estaría dispuesto(a) a pagar por transacción por este servicio.?**

   _____

ANEXO 11 Graficas de Mercados Potenciales

Parque de la Flores Cuenca



Mercado Artesanal Guayaquil

PLAZA DEL PONCHO OTAVALO

ANEXO 12 ANALISIS FINANCIERO A TRES AÑOS

PAGO ELECTRONICO A TRAVES DE DISPOSITIVOS MOVILES

ANALISIS FINANCIERO A TRES AÑOS

RUBRO	AÑO 0	AÑO 1	AÑO2	AÑO 3
EGRESOS				
Gastos Iniciales				
Adquisicion de dos Computadores	1500			
Instalación de Internet	35			
Costos de Analisis y Diseño	1734			
Costo de Desarrollo e Implementacion	3005			
Oficina	720			
Servicios Básicos	120			
TOTAL GASTOS INICIALES	7114			
Gastos Anuales				
Oficina		1440	1584	2400
Servicios Básicos		240	360	960
Pago por Costos de Servidor Dedicado		2460	3612	6000
Pago a SuSistema		20099	43854	87555
Internet		408	516	912
Técnico Programador		4200	12000	19200
Secretaria			3600	6000
Vendedor		2400	8400	7200
Contador			7200	9600
TOTAL		**31247**	**81126**	139827
INGRESOS				
Rubro por cobro de Transacciones a $0,25 c/u		33498	98902	218888
TOTAL INGRESOS - EGRESOS		2251	17776	79061

ANEXO 13 ANALISIS FINANCIERO A MENSUAL DE LAS TRANSACCIONES DEL PRIMER AÑO

PAGO ELECTRONICO A TRAVES DE DISPOSITIVOS MOVILES

ANALISIS FINANCIERO MENSUAL DEL PRIMER AÑO DE LAS TRANSACCIONES

TRANSACCIONES	NUMERO	COSTO	PAGO A SuSistema
1 MES	5000	1250	750
2 MES	5500	1375	825
3 MES	6600	1650	990
4 MES	7920	1980	1188
5 MES	9504	2376	1426
6 MES	10454	2614	1568
7 MES	11500	2875	1725
8 MES	12650	3162	1897
9 MES	13915	3479	2087
10 MES	16002	4001	2400
11 MES	16802	4201	2520
12 MES	18146	4537	2722
TOTAL	133993	33498	20099
TOTAL NETO ANUAL		**13399**	

ANEXO 14 ANALISIS FINANCIERO A MENSUAL DE LAS TRANSACCIONES DEL SEGUNDO AÑO

PAGO ELECTRONICO A TRAVES DE DISPOSITIVOS MOVILES

ANALISIS FINANCIERO MENSUAL DEL SEGUNDO AÑO DE LAS TRANSACCIONES

TRANSACCIONES	NUMERO	COSTO	PAGO A SuSistema
1 MES	18500	4625	2775
2 MES	20350	5088	3053
3 MES	22385	5596	3358
4 MES	24624	6156	3694
5 MES	27086	6772	2709
6 MES	29794	7449	2979
7 MES	32774	8194	3277
8 MES	36051	9013	3605
9 MES	39656	9914	3966
10 MES	43622	10906	4362
11 MES	47984	11996	4798
12 MES	52782	13196	5278
TOTAL	395608	98902	43854
TOTAL NETO		**55048**	

ANEXO 15 ANALISIS FINANCIERO A MENSUAL DE LAS TRANSACCIONES DEL TERCER AÑO

PAGO ELECTRONICO A TRAVES DE DISPOSITIVOS MOVILES

ANALISIS FINANCIERO MENSUAL DEL TERCER AÑO DE LAS TRANSACCIONES

TRANSACCIONES	NUMERO	COSTO	PAGO A SuSistema
1 MES	55422	13856	5542
2 MES	58193	14548	5819
3 MES	61103	15276	6110
4 MES	64158	16040	6416
5 MES	67366	16842	6737
6 MES	70734	17684	7073
7 MES	74271	18568	7427
8 MES	77984	19496	7798
9 MES	81884	20471	8188
10 MES	85978	21495	8598
11 MES	88127	22032	8813
12 MES	90330	22583	9033
TOTAL	875550	218888	87555
TOTAL NETO		**131332,5**	

BIBLIOGRAFIA

1. NUMERO DE USUARIOS OPERADORAS MOVIL, www.conatel.gov.ec/site_conatel/index.php?option=com_docman&task=doc_download&gid=1138&Itemid=, SUPERINTENDENCIA DE TELECOMUNICACIONES DEL ECUADOR, Agosto de 2008.

2. NUMERO DE TARJETAS POR CLASE, http://www.superban.gov.ec/pages/c_tarjetas_habientes.htm, SUPERINTENDENCIA DE BANCOS Y SEGUROS, Junio de 2008

3. HAVET INTERACTIVE S.A, GPRS: la nueva generación de telefonía móvil, HAVET, pp56 , Noviembre de 2001.

4. PAGUE DE FORMA RAPIDA Y SEGURA EN INTERNET, http://www.paypal.es/es, PayPal, Septiembre de 2008.

5. LA SOLUCION INTEGRAL DE PAGOS PARA SU NEGOCIO ONLINE, http://www.alignet.com, ALIGNET, septiembre de 2008.

6. PREGUNTAS MAS FRECUENTES PAGO –E Y TEF, http://www.davara.com/preguntas/pago.html#3, Davara & Davara Asesores Jurídicos, Julio de 2008.

7. PCI SECURITY STANDARDS COUNCIL TM, Normas de Seguridad de Datos de la Industria de Tarjetas de Pago (PCI), pp 1-9, PCI, Septiembre de 2006.

8. SuSistema Cia. Ltda., Switch de Autorizaciones, pp1-4, Septiembre 2002.

9. Tarjetas de Crédito - Cobertura de Tarjetas, https://www.superban.gov.ec/pages/c_tarjetas_cobertura.htm, SUPERINTENDENCIA DE BANCOS Y SEGUROS, Junio de 2008.

10. FIRTMAN, R. MAXIMILIANO, Programación para Celulares con Java, MP Ediciones, pp16-91, Agosto de 2004.

11. GALVEZ, S.; ORTEGA, L., Java a Tope: J2ME, Dpto. de Lenguajes y Ciencias de la Computación, E.T.S. de Ingeniería Informática, Universidad de Málaga, pp4-155, año 2003.

12. REVISTA DIGITAL LIDER EN INFORMATICA, http://www.mastermagazine.info/termino/7292.php, Master Magazine, 2004.

13. SANCHEZ, J., Análisis y Estudio de Redes GPRS; Escuela de Electricidad y Electrónica Universidad Austral de Chile, pp26-40, 2004.

14. SERVICIO GENERAL DE PAQUETES VIA RADIO, http://es.wikipedia.org/wiki/General_Packet_Radio_Service, WIKIPEDIA, Julio de 2006.

15. Hypertext Transfer Protocol Secure, http://es.wikipedia.org/wiki/HTTPS, WIKIPEDIA, noviembre de 2008.

16. Transport Layer Security, http://es.wikipedia.org/wiki/Transport_Layer_Security, WIKIPEDIA, noviembre de 2008

17. ZAFRILLA, M., Mensaje Cifrado, SM colección Gran Angular, pp153, 2007.

18. CHAVEZ, J., Protocolos de Red: Protocolo TCP/IP, Monografía, pp1-5, 2006.

19. STALLINGS, W., Comunicaciones y Redes de Computadores, 6ª edición Prentice Hall, pp50-65, año 2000.

20. ALORREAGA D., Firewall y Seguridades en Internet, Universidad Nacional Autónoma de México, pp6-20, año 2004.

21. Posicionamiento en Buscadores, http://www.ilatina.es/urlque-url/2-12-7-12.htm, iLatina Software S.L., 2008.

22. Free Trial SSL Certifcate, Test Root CA Instructions, Verisign, Inc. http://www.verisign.com/ssl/buy-ssl-certificates/free-ssl-certificate-trial/test-root-ca/trialcainstall.html, 2008.

23. CERVERA A., Analysis of J2ME For Developing Mobile Payment Systems, IT University of Copenhagen, pp25, año 2002.

24. Criptosistemas de Clave pública. El Cifrado RSA, http://www.dma.fi.upm.es/java/matematicadiscreta/aritmeticamodular/rsa.html, Universidad Politécnica de Madrid, año 2004.

25. KNUDSEN JONATHAN, Midp Application Security 1: Design Concerns and Cryptography, http://developers.sun.com, Septiembre 2002

26. Documentation, Training & Support, http://www.netbeans.org/kb/index.html, NetBeans.

27. Criptografía – El protocol SET, http://es.kioskea.net/contents/crypto/set.php3, Kioskera.net, Octubre 2008.

28. STALLINGS, W., Fundamentos de Seguridad en Redes, 2ª edición Prentice Hall, pp10-47, año 2007.

29. LETELIER, P., Desarrollo de Software Orientado a Objeto usando UML, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia España, Mayo de 2005.

30. ROBAYO, M.; ANDRADE H.; GARCES, R., El Sistema de Pagos Ecuatoriano, Dirección General Bancaria, Banco Central del Ecuador, pp 168-173, Diciembre de 2003.

31. Aplicaciones de Punto de Venta, http://technet.microsoft.com/es-es/library/ms151330.aspx, Microsoft TechNet, Agosto de 2008.

32. Procese las transacciones electrónicamente, https://iata.bankguay.com/amex/smallbusiness/procesar.asp, American Express Cards Welcome, Noviembre de 2008.

33. Servidores Dedicados, http://dattatec.com/site/sp/estados-unidos/servidores-dedicados, dattatec.com Argentina, año 2006.