

CAPITULO I. VISION GENERAL

1.1. INTRODUCCIÓN

Los Sistemas de Información nos permiten organizar, administrar y distribuir, la Información de una organización o entidad, un sistema de información es un conjunto de elementos orientados al tratamiento y administración de datos e información organizada para su uso eficiente, estos medios tecnológicos están orientados a simplificar las tareas rutinarias.

El comercio electrónico permite realizar electrónicamente transacciones comerciales, basados en el tratamiento y transmisión electrónica de datos, incluidos texto, imágenes y vídeo. El comercio electrónico comprende actividades muy diversas, como el comercio electrónico de bienes y servicios, suministro en línea de contenidos digitales, transferencia electrónica de fondos, compraventa electrónica de acciones, diseños y proyectos conjuntos, prestación de servicios en línea, comercialización directa al consumidor y servicios posventa.

La falta de tiempo, el trabajo, las largas filas, son los principales factores que impiden la administración oportuna de nuestros pagos de servicios básicos ya que la mora en el pago de los mismos puede causar el corte de éstos.

Los intermediarios son compañías que facilitan las transacciones entre compradores y vendedores, como contraprestación económica reciben un porcentaje del valor de la transacción. La mayoría de las transacciones se realizan a través de estos intermediarios especializados.

El sistema de gestión de pagos de servicios básicos está orientado a brindar mayor comodidad, mayor seguridad, control y seguimiento de los pagos realizados por el concepto de servicios básicos como ser: Telefonía celular, telefonía fija, internet, Tv Cable, Luz, Gas, Agua, Colegio.

1.2. ANTECEDENTES

<http://www.interbank.com.pe/pago-servicios-publicos>

Interbank del Perú, es una entidad que realiza los pagos de servicios públicos en forma fácil y segura mediante una amplia Red de Tiendas a nivel nacional y canales electrónicos, donde se puede realizar los pagos tan sólo con el código de referencia o tu nombre y apellido, para esto el cliente debe afiliarse a este servicio, pero previamente debe aperturar una cuenta de la cual se realizara el descargo correspondiente a los pagos realizados.

<http://www.tigo.com.bo/tigo-money/pago-facturas-tigo>

Tigo Money permite pagar facturas de: telefonía de la misma empresa, internet, música, Yambal, actualmente permite pagar facturas de las empresas de luz y agua (CRE y SAGUAPAC), para ser beneficiario de estos servicios de pago el cliente debe tener dinero en su cuenta, el monto necesario para hacer efectivo el pago de la factura. Sólo se necesita tener el número de código fijo, no es necesario ser el titular para realizar el pago. Se podrán pagar facturas por montos iguales o menores a Bs.- 1800.- (Mil Ochocientos 00/100 Bolivianos), la factura será enviada al cliente Tigo Money que realizó el pago.

<https://gananet.bg.com.bo/gananet/>

Este servicio de banca virtual del Banco Ganadero permite que usted, pueda realizar operaciones bancarias a través de su computador utilizando la red internet, en una interfaz segura y confiable.

Le permite realizar una amplia gama de servicios como ser: consultas de saldos de Cuentas, Tarjetas de Débito o Tarjetas de Crédito, transferencias entre sus cuentas o a cuentas de terceros, Pagar Servicios Básicos.

1.3. PLANTEAMIENTO DEL PROBLEMA

La no disponibilidad y el demasiado tiempo que las personas deben emplear, para cumplir con el pago mensual de sus servicios básicos.

1.4. OBJETIVOS

1.4.1. OBJETIVO GENERAL

Analizar, diseñar e implantar un sistema plataforma para el pago de servicios básicos realizados por terceros.

1.4.2. OBJETIVOS ESPECÍFICOS

1. Desarrollar un sistema, para el pago y control de los servicios básicos realizados por terceros.
2. Utilizar herramientas para la recolección de requerimientos.
3. Aplicar la metodología orientada a objetos en base al Proceso de desarrollo Unificado Racional RUP para el desarrollo del sistema.
4. Utilizar el UML (Lenguaje de Modelado Unificado) para el análisis y diseño del sistema.
5. Utilizar el Modelo Entidad Relación para modelar la Base de Datos del Sistema.
6. Confeccionar interfaces que permitan el fácil manejo del sistema.
7. Implementar el sistema con herramientas orientadas a la web.

1.5. JUSTIFICACIÓN

1.5.1. JUSTIFICACIÓN ACADÉMICA

Con la realización de este trabajo se pondrá en práctica los conocimientos adquiridos durante la formación académica, aplicando en el transcurso del proyecto la ingeniería de software, tecnología orientada a objetos, y otras herramientas de documentación para el desarrollo de sistemas de información.

1.5.2. JUSTIFICACIÓN TECNOLÓGICA

En la actualidad existen herramientas tanto de software y hardware que facilitarán la realización de este trabajo.

1.5.3. JUSTIFICACIÓN SOCIAL

Con este proyecto se quiere remarcar la importancia del uso de la tecnología y consecuentemente, la mejora en los pagos de servicios básicos realizados por terceros, de esta manera poder administrar, ver el historial, etc. de los mismos, información que se la podrá obtener de forma rápida y oportuna.

1.6. ALCANCES

- Diseñar un módulo para la gestión de personas
 - Altas, modificaciones
- Diseñar un módulo para la gestión de usuarios
 - Altas, modificaciones
 - Roles
 - Opciones de menú
- Diseñar un módulo para la gestión de clientes
 - Altas, modificaciones
- Diseñar un módulo para la gestión de servicios
 - Altas, modificaciones
 - Asignación de servicio a cliente
- Diseñar un módulo para la gestión de comisiones
 - Altas, modificaciones
- Diseñar un módulo para la gestión de cuentas de clientes.
 - Altas, modificaciones

- Diseñar un módulo para la gestión de depósitos a las cuentas del cliente
 - Altas, modificaciones
- Diseñar un módulo para la gestión de transacciones
 - Depósitos
 - Pagos
- Diseñar un módulo para reportes según requerimientos de los usuarios
 - Historial de cuentas, depósitos y pagos
 - Listado de clientes
 - Listado de usuarios
 - Listado de pagos, comisiones. etc.

1.7. LIMITACIONES

No se contemplan las etapas de implantación y mantenimiento para este proyecto.

CAPITULO II. MARCO TEORICO

2.1. DINERO ELECTRÓNICO (eCash): Según la definición de la Unión Europea se trata de sistemas de pago que pueden ser utilizados para pagos no relacionados con la empresa que provee el sistema. Es decir, quedan afuera de la definición sistemas como las tarjetas o los certificados de regalo (gift card) o las tarjetas telefónicas. El cliente puede usarlo prácticamente como dinero en efectivo. La mayoría de los sistemas trabajan con tarjetas o cuentas recargables, y pueden ser utilizados también en el comercio común (no electrónico), mientras que otros están basados en software.

2.2. CUENTAS VIRTUALES: En este caso el cliente adquiere un monto de dinero virtual, que puede utilizar para pagos de distintos servicios, muchas veces limitados a una sola empresa. El ejemplo más común son las tarjetas telefónicas, en las cuales el cliente, a través de un código que se encuentra de forma oculta en la misma tarjeta, accede a una cuenta con un monto fijo que puede utilizar.

2.3. EL COMERCIO ELECTRÓNICO: también conocido como **e-commerce** (electronic commerce en inglés), consiste en la compra y venta de productos o de servicios a través de medios electrónicos, tales como Internet y otras redes informáticas. Originalmente el término se aplicaba a la realización de transacciones mediante medios electrónicos tales como el intercambio electrónico de datos, sin embargo con el advenimiento de la Internet y la World Wide Web a mediados de los años 90 comenzó a referirse principalmente a la venta de bienes y servicios a través de Internet, usando como forma de pago medios electrónicos, tales como las tarjetas de crédito.

La cantidad de comercio llevada a cabo electrónicamente ha crecido de manera extraordinaria debido a Internet y los sistemas informáticos. Una gran variedad de comercio se realiza de esta manera, estimulando la creación y utilización de innovaciones como la transferencia de fondos electrónica, la administración de cadenas de suministro, el marketing en Internet, el procesamiento de transacciones en

línea, el intercambio electrónico de datos, los sistemas de administración del inventario y los sistemas automatizados de recolección de datos.

La mayor parte del comercio electrónico consiste en la compra y venta de productos o servicios entre personas y empresas, sin embargo un porcentaje considerable del comercio electrónico se basa en la adquisición de artículos virtuales (software y derivados en su mayoría), algunas ventajas de estas nuevas formas de comercio electrónico son:

- **Ubicuidad:** la tecnología de internet está disponible en todos lados, en el trabajo, en el hogar y en cualquier otro lado a través de los dispositivos móviles, en cualquier momento. El mercado se extiende más allá de los límites tradicionales y se elimina de una ubicación temporal y geográfica. Se crea el “Marketspace”; se pueden realizar compras en cualquier parte, mejorando la conveniencia para el cliente y la reducción de los costos de compra.
- **Alcance global:** la tecnología se extiende más allá de los límites nacionales. Se habilita el comercio a través de los límites culturales y nacionales sin problemas ni modificaciones. El “Marketspace” incluye potencialmente miles de millones de clientes en todo el mundo.
- **Infraestructura de red:** Internet (VAN, LAN, WAN), Intranet, Extranet, televisión por cable y satelital, dispositivos electrónicos, Internet.
- **Infraestructura de los mensajes y de distribución de información:** intercambio electrónico de datos, correo electrónico, protocolo de transferencia de hipertexto y otros.
- **Infraestructura de interfaces:** está asentado en bases de datos, agenda de clientes, aplicaciones y sus interrelaciones.

2.4. FRAMEWORK: Un framework es una colección organizada de clases que constituyen un diseño reutilizable para un dominio específico de software. Contiene un conjunto de librerías, componentes de software y directrices arquitectónicas que

ofrece al desarrollador un kit de herramientas completo para construir una aplicación de principio a fin, siempre teniendo en cuenta que es necesario adaptarlo a cada aplicación en particular.

2.5. RUP: El Proceso Unificado de Racional (Rational Unified Process en inglés, habitualmente abreviado como RUP) desarrollado por la empresa Rational Software, actualmente propiedad de IBM, es una metodología que se utiliza para el desarrollo del sistema de software, el cual contiene las siguientes fases: inicio, elaboración y construcción.

“RUP” es un framework metodológico, lo que significa que puede (y debe) ser adaptada a las características propias del proyecto, de esta forma, RUP se puede utilizar para proyectos grandes, pequeños y medianos. Junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas.

2.6. LENGUAJE UNIFICADO DE MODELADO – UML: por sus siglas en inglés, Unified Modeling Language, es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del modelo de sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante señalar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos, se utiliza para definir, detallar los artefactos en el sistema, documentar y construir, es decir es un lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

2.7. DIAGRAMA DE CASOS DE USO: Es una técnica para capturar información de cómo un sistema o negocio trabaja actualmente y de cómo se desea que trabaje. No pertenece al enfoque orientado a objetos, más bien es una técnica para el modelado de escenarios en los cuales el sistema debe operar. Los diagramas de casos de uso son importantes para modelar el comportamiento de un sistema, un subsistema o una clase, cada uno muestra un conjunto de casos de uso, actores y sus relaciones.

Los diagramas de caso de uso son uno de los cinco tipos de diagramas en UML para modelar aspectos dinámicos de sistemas como él (diagramas de actividad, de estados, de secuencia y colaboración).

Los diagramas de casos de uso son importantes para visualizar, especificar, y documentar el comportamiento de un elemento. Ellos hacen sistemas, subsistemas, y clases entendibles para presentar una vista exterior de cómo estos elementos pueden ser usados dentro del contexto.

2.8. DIAGRAMA DE CLASES: Los diagramas de clases muestran las diferentes clases que componen un sistema y cómo se relacionan unas con otras, se dice que los diagramas de clases son estáticos porque muestran las clases junto con sus métodos y atributos, así como las relaciones estáticas entre ellas: qué clases “conocen”, a qué otras clases o qué clases “son parte” de otras, pero no muestran los métodos mediante los que se invocan entre ellas.

2.9. DIAGRAMA DE SECUENCIA: Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista del escenario, el diagrama de secuencia contiene detalles de

implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario y mensajes intercambiados entre los objetos.

2.10. DIAGRAMA DE COLABORACIÓN: Muestra objetos y sus relaciones, destacando los objetos que participan en el intercambio de mensajes. Los diagramas de colaboración muestran explícitamente las relaciones de roles. Por otra parte un diagrama de colaboración no muestra el tiempo como una dimensión aparte, por lo que resulta necesario etiquetar con números de secuencia tanto la secuencia de mensajes como los hilos concurrentes.

2.11. DIAGRAMA DE ACTIVIDAD: El diagrama de flujo o diagrama de actividades es la representación gráfica del algoritmo. Esta representa una actividad, un paso en el flujo de trabajo o la ejecución de una operación. Un gráfico de actividades describe grupos de secuencias y concurrentes de actividad. Las actividades se enlazan por transiciones automáticas, cuando una actividad termina se desencadena el paso a la siguiente actividad. Un diagrama de actividad nos ayuda a entender el comportamiento de la ejecución de un sistema, sin profundizar en los detalles internos de los mensajes. Los parámetros de entrada y salida de una acción se pueden mostrar usando las relaciones de flujo que conectan la acción y un estado de flujo de objeto.

2.12. DIAGRAMA DE COMPONENTES: Estos diagramas, muestran los componentes del software incluyendo el código fuente, binario y ejecutable. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, las tablas de una base de datos, etc. Las relaciones de dependencia se utilizan en los diagramas de componentes para indicar que un componente utiliza los servicios ofrecidos por otro componente.

2.13. DIAGRAMA DE DESPLIEGUE: Los diagramas de despliegue, se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes, muestran la relación física entre los distintos nodos que componen un sistema (hardware y software) en el sistema final. Es decir,

la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos). Describen la arquitectura física del sistema durante la ejecución, en términos de: procesos, dispositivos y componentes software.

2.14. MySQL: es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. MySQL es una aplicación que permite gestionar archivos llamados de bases de datos.

Existen muchos tipos de bases de datos, desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL, como base de datos relacional, utiliza múltiples tablas para almacenar y organizar la información. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interactuación con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

La condición de open source de MySQL, hace que su utilización sea gratuita e incluso se pueda modificar con total libertad su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.

- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica, transacciones.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

2.15. INTEGRIDAD DE LOS DATOS: Cuando se trabaja con bases de datos, generalmente los datos se reparten entre varios ficheros, en MySQL la base de datos está disponible para varios usuarios de forma simultánea, deben existir mecanismos que aseguren que las interrelaciones entre registros se mantienen coherentes, que se respetan las dependencias de existencia y que las claves únicas no se repitan.

Por ejemplo, un usuario no debe poder borrar una entidad de una base de datos, si otro usuario está usando los datos de esa entidad. Este tipo de situaciones son potencialmente peligrosas, ya que provocan situaciones con frecuencia imprevistas. Ciertos errores de integridad pueden provocar que una base de datos deje de ser usable.

Los problemas de integridad se suelen producir cuando varios usuarios están editando datos de la misma base de datos de forma simultánea. Por ejemplo, un usuario crea un nuevo registro, mientras otro edita uno de los existentes, y un tercero borra otro. El DBMS debe asegurar que se pueden realizar estas tareas sin que se produzcan errores que afecten a la integridad de la base de datos.

2.16. EL MODELO ENTIDAD-RELACIÓN: El modelo entidad-relación (en adelante E-R), consiste en buscar las entidades que describan los objetos que intervienen en el problema y las relaciones entre esas entidades.

Todo esto se plasma en un esquema gráfico que tiene por objeto, por una parte, ayudar al programador durante la codificación y por otra, al usuario a comprender el problema y el funcionamiento del programa.

Pero lo primero es lo primero, y antes de continuar, necesitamos entendernos. De modo que definiremos algunos conceptos que se usan en el modelo E-R. Estas definiciones nos serán útiles tanto para explicar la teoría, como para entendernos entre nosotros y para comprender otros textos sobre el modelado de bases de datos. Se trata de conceptos usados en libros y artículos sobre bases de datos, de modo que será interesante conocerlos con precisión.

- **Entidad:** Es una representación de un objeto individual concreto del mundo real.
- **Conjunto de entidades:** Es la clase o tipo al que pertenecen entidades con características comunes.
- **Atributo:** Cada una de las características que posee una entidad, y que agrupadas permiten distinguirla de otras entidades del mismo conjunto.
- **Dominio:** Conjunto de valores posibles para un atributo.
- **Interrelación:** Es la asociación o conexión entre conjuntos de entidades.
- **Clave:** Es un conjunto de atributos que identifican de forma única una entidad.
- **Clave candidata:** Es cada una de las claves mínimas existentes en un conjunto de entidades.
- **Clave principal:** (o primaria), es una clave candidata elegida de forma arbitraria, que usaremos siempre para identificar una entidad.

2.17. MODELO RELACIONAL: Entre los modelos lógicos, el modelo relacional está considerado como el más simple y el que más nos conviene. Por una parte, el paso del modelo E-R al relacional es muy simple, y por otra, **MySQL**, como implementación de SQL, está orientado principalmente a bases de datos relacionales.

El doctor Edgar F. Codd, un investigador de IBM, inventó en 1970 el modelo relacional, también desarrolló el sistema de normalización, básicamente el modelo relacional se compone de tres partes:

1. Estructura de datos: básicamente se compone de relaciones.
2. Manipulación de datos: un conjunto de operadores para recuperar, derivar o modificar los datos almacenados.
3. Integridad de datos: una colección de reglas que definen la consistencia de la base de datos.

2.18. NORMALIZACIÓN DE BASE DE DATOS RELACIONALES

La normalización es el proceso de organizar los datos de una base de datos. Se incluye la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos como para hacer que la base de datos sea más flexible al eliminar la redundancia y las dependencias incoherentes.

Los datos redundantes desperdician el espacio de disco y crean problemas de mantenimiento. Si hay que cambiar datos que existen en más de un lugar, se deben cambiar de la misma forma exactamente en todas sus ubicaciones. Un cambio en la dirección de un cliente es mucho más fácil de implementar si los datos sólo se almacenan en la tabla Clientes y no en algún otro lugar de la base de datos.

¿Qué es una "dependencia incoherente"? Aunque es intuitivo para un usuario mirar en la tabla Clientes para buscar la dirección de un cliente en particular, puede no tener sentido mirar allí el salario del empleado que llama a ese cliente. El salario del empleado está relacionado con el empleado, o depende de él, y por lo tanto se debería pasar a la tabla Empleados. Las dependencias incoherentes pueden dificultar el acceso porque la ruta para encontrar los datos puede no estar o estar interrumpida.

Hay algunas reglas en la normalización de una base de datos. Cada regla se denomina una "forma normal". Si se cumple la primera regla, se dice que la base de datos está

en la "primera forma normal". Si se cumplen las tres primeras reglas, la base de datos se considera que está en la "tercera forma normal". Aunque son posibles otros niveles de normalización, la tercera forma normal se considera el máximo nivel necesario para la mayor parte de las aplicaciones.

Al igual que con otras muchas reglas y especificaciones formales, en los escenarios reales no siempre se cumplen los estándares de forma perfecta. En general, la normalización requiere tablas adicionales y algunos clientes consideran éste un trabajo considerable. Si decide infringir una de las tres primeras reglas de la normalización, asegúrese de que su aplicación se anticipa a los problemas que puedan aparecer, como la existencia de datos redundantes y de dependencias incoherentes.

2.18.1. Primera forma normal

- Elimine los grupos repetidos de las tablas individuales.
- Cree una tabla independiente para cada conjunto de datos relacionados.
- Identifique cada conjunto de datos relacionados con una clave principal.

No use varios campos en una sola tabla para almacenar datos similares. Por ejemplo, para realizar el seguimiento de un elemento del inventario que proviene de dos orígenes posibles, un registro del inventario puede contener campos para el Código de proveedor 1 y para el Código de proveedor 2.

¿Qué ocurre cuando se agrega un tercer proveedor? Agregar un campo no es la respuesta, requiere modificaciones en las tablas y el programa, y no admite fácilmente un número variable de proveedores. En su lugar, coloque toda la información de los proveedores en una tabla independiente denominada Proveedores y después vincule el inventario a los proveedores con el número de elemento como clave, o los proveedores al inventario con el código de proveedor como clave.

2.18.2. Segunda forma normal

- Cree tablas independientes para conjuntos de valores que se apliquen a varios registros.
- Relacione estas tablas con una clave externa.

Los registros no deben depender de nada que no sea una clave principal de una tabla, una clave compuesta si es necesario. Por ejemplo, considere la dirección de un cliente en un sistema de contabilidad. La dirección se necesita en la tabla Clientes, pero también en las tablas Pedidos, Envíos, Facturas, Cuentas por cobrar y Colecciones. En lugar de almacenar la dirección de un cliente como una entrada independiente en cada una de estas tablas, almacénela en un lugar, ya sea en la tabla Clientes o en una tabla Direcciones independiente.

2.18.3. Tercera forma normal

- Elimine los campos que no dependan de la clave.

Los valores de un registro que no sean parte de la clave de ese registro no pertenecen a la tabla. En general, siempre que el contenido de un grupo de campos pueda aplicarse a más de un único registro de la tabla, considere colocar estos campos en una tabla independiente. Por ejemplo, en una tabla Contratación de empleados, puede incluirse el nombre de la universidad y la dirección de un candidato. Pero necesita una lista completa de universidades para enviar mensajes de correo electrónico en grupo. Si la información de las universidades se almacena en la tabla Candidatos, no hay forma de enumerar las universidades que no tengan candidatos en ese momento. Cree una tabla Universidades independiente y vincúlela a la tabla Candidatos con el código de universidad como clave.

EXCEPCIÓN: cumplir la tercera forma normal, aunque en teoría es deseable, no siempre es práctico. Si tiene una tabla Clientes y desea eliminar todas las dependencias posibles entre los campos, debe crear tablas independientes para las

ciudades, códigos postales, representantes de venta, clases de clientes y cualquier otro factor que pueda estar duplicado en varios registros. En teoría, la normalización merece el trabajo que supone. Sin embargo, muchas tablas pequeñas pueden degradar el rendimiento o superar la capacidad de memoria o de archivos abiertos.

Puede ser más factible aplicar la tercera forma normal sólo a los datos que cambian con frecuencia. Si quedan algunos campos dependientes, diseñe la aplicación para que pida al usuario que compruebe todos los campos relacionados cuando cambie alguno.

2.18.4. Otras formas de normalización

La cuarta forma normal, también llamada Forma normal de Boyce Codd (BCNF, Boyce Codd Normal Form), y la quinta forma normal existen, pero rara vez se consideran en un diseño real. Si no se aplican estas reglas, el diseño de la base de datos puede ser menos perfecto, pero no debería afectar a la funcionalidad.

2.19. PHP: Es un acrónimo recursivo que significa PHP Hypertext Pre-processor, y se trata de un lenguaje de scripting para la programación de páginas dinámicas de servidor. Es un lenguaje de tipo gratuito, y forma parte del software que se conoce como de código abierto (Open Source). Es decir que se le pueden introducir modificaciones y mejoras y ponerlas a disposición de los demás usuarios del mismo.

Otra característica importante es que se trata de un lenguaje multiplataforma, esto quiere decir que la aplicación web desarrollada en PHP puede funcionar en casi cualquier tipo de plataforma Windows, Unix/Linux (y sus diferentes versiones y distribuciones). También ofrece soporte a los motores de base de datos más populares (SQL Server, MySQL, PosgresSQL, Oracle, etc.), como así también acceso ODBC (open database connectivity).

Una aplicación web basada en PHP necesita dos tipos de software. El primero es un servidor web que va a atender las peticiones de los usuarios y devolverá las páginas

solicitadas. El servidor Apache, tanto su versión Windows como Linux es el más utilizado. El segundo software es el propio PHP, es decir el módulo que se va a encargar de interpretar y ejecutar los scripts que se soliciten al servidor.

2.20. SERVIDOR HTTP APACHE: es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años.

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

2.21. JAVASRIPT: JavaScript es un lenguaje interpretado orientado a las páginas web, con una sintaxis semejante a la del lenguaje Java.

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que fabricó los primeros navegadores de Internet

comerciales. Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Se utiliza en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente. Los autores inicialmente lo llamaron Mocha y más tarde LiveScript pero fue rebautizado como JavaScript en un anuncio conjunto entre Sun Microsystems y Netscape, el 4 de diciembre de 1995.

En 1997 los autores propusieron JavaScript para que fuera adoptado como estándar de la European Computer Manufacturers' Association ECMA, que a pesar de su nombre no es europeo sino internacional, con sede en Ginebra. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de ECMAScript. Poco después también lo fue como un estándar ISO.

Javascript nació con la necesidad de permitir a los autores de sitio web crear páginas que permitan intercambiar con los usuarios, ya que se necesitaba crear webs de mayor complejidad. El HTML sólo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios.

2.22. SMARTY: Es un motor de plantillas para PHP, es decir, separa el código PHP, como lógica de negocios, del código HTML, como lógica de presentación, y genera contenidos web mediante la colocación de etiquetas Smarty en un documento. Se encuentra bajo la Licencia Pública General Reducida de GNU.

Es común que en grandes proyectos el rol de diseñador gráfico y el de programador sean cubiertos por personas distintas, sin embargo la programación en PHP tiene la tendencia de combinar estas dos labores en una persona y dentro del mismo código, lo que trae consigo grandes dificultades a la hora de cambiar alguna parte del diseño de la página, pues se tiene que escarbar entre los scripts para modificar la presentación del contenido, Smarty tiene como objetivo solucionar este problema.

Existen más sistemas de plantillas para PHP, pero éste parece ser la más avanzada y con más frecuencia de desarrollo. También hay detractores de estas técnicas que alegan que las mismas hacen en cierta medida un grado más complejo el desarrollo web, por la necesidad de aprender un (pseudo) lenguaje más.

Los detractores de esta idea se basan en el hecho de que, precisamente, el lenguaje PHP nació como un lenguaje rápido para hacer desarrollos web a pequeña escala. A medida van surgiendo sistemas de separación en capas que intentan disciplinar un poco las metodologías de programación envueltas en el desarrollo con PHP, pero que no hacen otra cosa que acercarse más y más a otras herramientas ya existentes en otros entornos de desarrollo más complejos y pensados desde sus orígenes para proyectos más grandes, como pueden ser J2EE (Java), .NET (C#) o Django (Python).

2.23. HERRAMIENTAS CASE: Las herramientas CASE: (computer aided software engineering, ingeniería asistida por computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

De acuerdo con Kendall y Kendall la ingeniería de sistemas asistida por ordenador es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias de desarrollo, su objetivo es acelerar el proceso para el que han sido diseñadas, en el caso de CASE para automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas.

Cuando se hace la planificación de la base de datos, la primera etapa del ciclo de vida de las aplicaciones de bases de datos, también se puede escoger una herramienta CASE (Computer-Aided sofware Engineering) que permita llevar a cabo el resto de

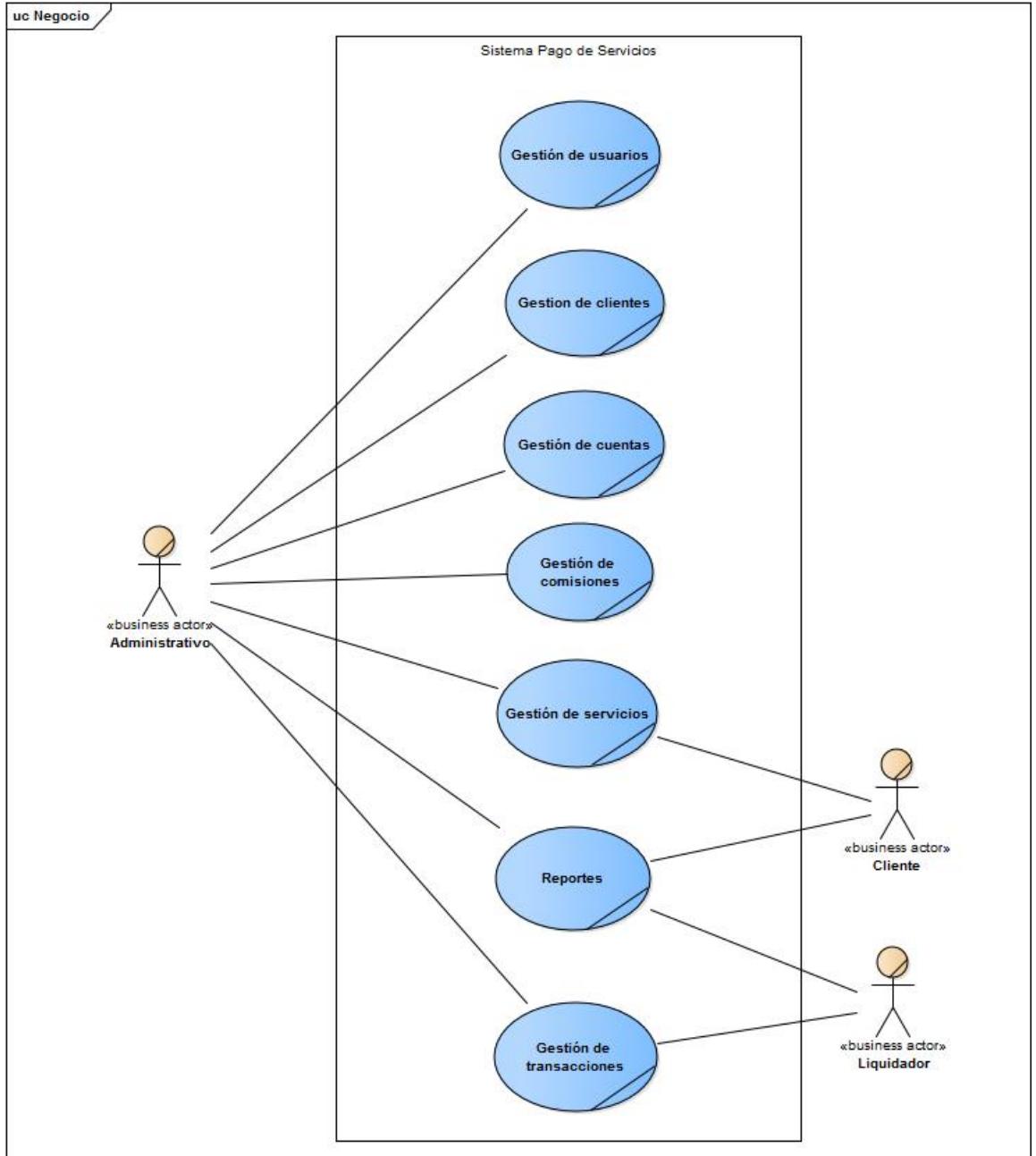
tareas del modo más eficiente y efectivo posible. Una herramienta CASE suele incluir:

- Un diccionario de datos para almacenar información sobre los datos de la aplicación de bases de datos.
- Herramientas de diseño para dar apoyo al análisis de datos.
- Herramientas que permitan desarrollar el modelo de datos corporativo, así como los esquemas conceptual y lógico.
- Herramientas para desarrollar los prototipos de las aplicaciones.

El uso de las herramientas CASE puede mejorar la productividad en el desarrollo de una aplicación de bases de datos.

CAPITULO III. ANALISIS DEL SISTEMA

3.1 DIAGRAMA DE CASO DE USO DE NEGOCIO

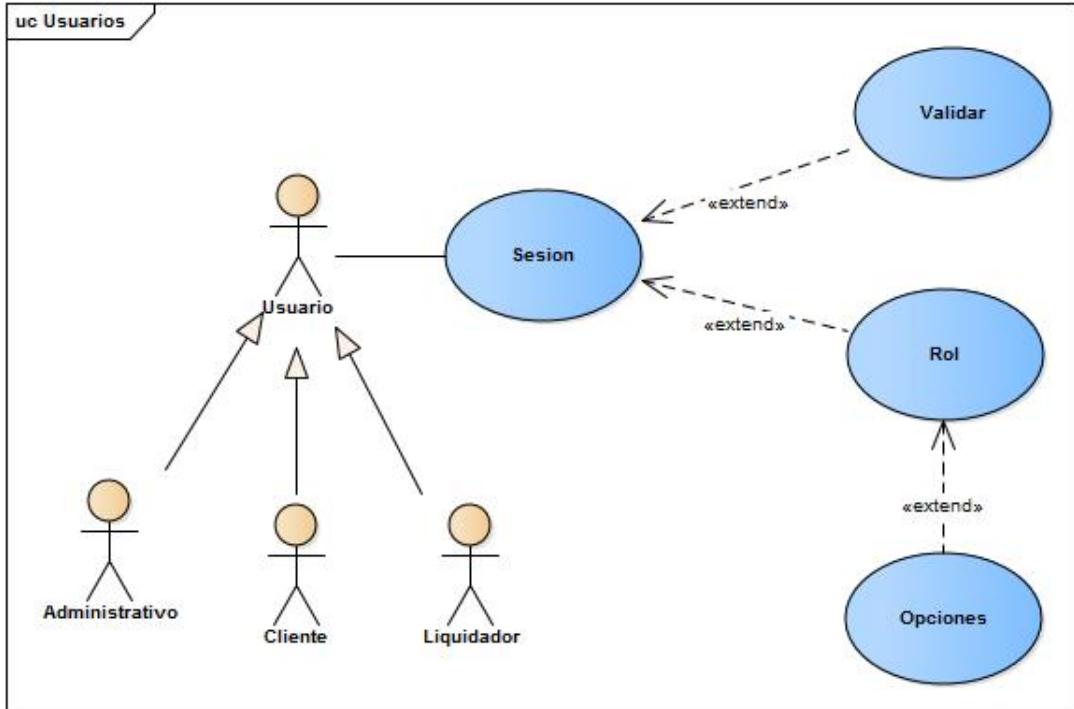


3.2 DESCRIPCIÓN DE LOS ACTORES

ACTOR	DESCRIPCIÓN
ADMINISTRATIVO	Este usuario es el responsable de la administración de: usuarios, clientes, cuentas de los clientes, comisiones, servicios y depósitos.
CLIENTE	Son todas aquellas personas que contratan el servicio de pago de facturas las cuales pueden hacer seguimiento de su cuenta, depósitos, comisiones y pago de los servicios pagados por terceros incluso puede ver la factura y datos para completar su libro de compras de los pagos realizados.
LIQUIDADOR	Es la persona encargada de realizar el pago de los servicios básicos de los clientes y subir la información del pago realizado al sistema.

3.3 DIAGRAMAS DE CASO DE USO

3.3.1 DIAGRAMA DE CASO DE USO USUARIOS



Caso de Uso:	Validar
Descripción:	Este caso de uso se encarga de buscar el registro del usuario y comparar la clave de acceso para el ingreso al sistema
Actores:	Usuario: Administrativo, Cliente, Liquidador
Precondición:	Que el usuario exista en la Base de Datos
Flujo normal	<ol style="list-style-type: none"> El usuario ingresa mediante intranet o internet al sistema Escribe los datos necesarios para ingresar Envía la petición para el acceso
Post condición:	El usuario ingresa al sistema
Flujo Alternativo:	El usuario no existe en la base de datos/Clave de usuario no es válida

Caso de Uso:	Rol
Descripción:	Este caso de uso indica el rol que tiene el usuario dentro del sistema
Actores:	Usuario: Administrativo, Cliente, Liquidador
Precondición:	Que el usuario exista en la Base de Datos y que éste tenga una clave de acceso permitida.
Flujo normal	1. El sistema indica el rol o roles asignados al usuario
Post condición:	
Flujo Alternativo:	El usuario no tiene rol asignado, el administrador debe asignar uno.

Caso de Uso:	Opciones
Descripción:	Este caso de uso muestra las opciones del rol del usuario
Actores:	Usuario: Administrativo, Cliente, Liquidador
Precondición:	Que el usuario exista en la Base de Datos, éste tenga una clave de acceso permitida y un rol asignado.
Flujo normal	1. El sistema presenta una lista de opciones u operaciones en forma de menú para que este pueda interactuar con los diferentes módulos del sistema.
Post condición:	
Flujo Alternativo:	El rol no tiene opciones para mostrar, el administrador es el encargado de asignar las opciones a cada rol.

3.3.2 DIAGRAMA DE CASO DE USO GESTIÓN ADMISNITRATIVA



Caso de Uso:	AM persona (Alta / Modificación Persona)
Descripción:	Adicionar o modificar una persona
Actores:	Usuario: Administrativo
Precondición:	Que la persona no exista en la Base de datos
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos de una persona 2. El administrador graba los datos
Post condición:	Persona adicionada con éxito/Datos modificados con éxito.
Flujo Alternativo:	La persona existe en la Base de Datos

Caso de Uso:	AM Usuario (Alta / Modificación Usuario)
Descripción:	Adicionar o modificar un usuario
Actores:	Usuario: Administrativo
Precondición:	Que exista un super usuario administrador que pueda crear nuevos usuarios en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos de un usuario 2. El administrador graba los datos
Post condición:	Usuario es adicionado con éxito/Datos del usuario modificados con éxito.
Flujo Alternativo:	El usuario existe en la Base de Datos

Caso de Uso:	AM Rol (Alta / Modificación Rol)
Descripción:	Adicionar o modificar rol
Actores:	Usuario: Administrativo
Precondición:	Que exista un súper usuario administrador que pueda crear nuevos roles en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos de un rol 2. El administrador graba los datos
Post condición:	El rol es adicionado con éxito/Datos del rol modificados con éxito.
Flujo Alternativo:	El rol existe en la Base de Datos

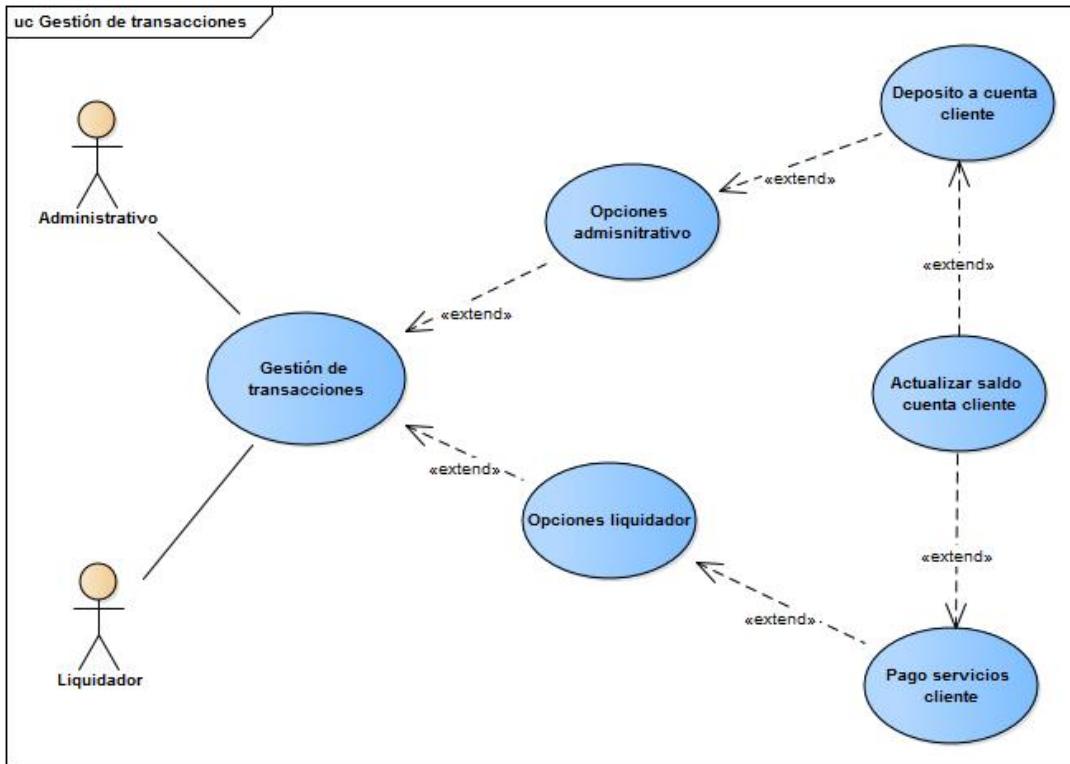
Caso de Uso:	AM Opciones (Alta / Modificación Opciones)
Descripción:	Adicionar o modificar opciones
Actores:	Usuario: Administrativo
Precondición:	Que exista un súper usuario administrador que pueda crear nuevas opciones en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos de una opción 2. El administrador graba los datos
Post condición:	El rol es adicionado con éxito/Datos de la opción modificados con éxito.
Flujo Alternativo:	El rol existe en la Base de Datos

Caso de Uso:	AM Cliente
Descripción:	Adicionar o modificar Cliente
Actores:	Usuario: Administrativo
Precondición:	Que exista registrados los datos de la persona en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos del cliente 2. El administrador graba los datos
Post condición:	El cliente es adicionado con éxito/Datos del cliente son modificados con éxito.
Flujo Alternativo:	El cliente existe en la Base de Datos

Caso de Uso:	AM Cuenta
Descripción:	Adicionar o modificar Cuenta
Actores:	Usuario: Administrativo
Precondición:	Que exista el cliente registrado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador introduce los datos o modifica los datos de la cuenta 2. El administrador graba los datos
Post condición:	La cuenta es adicionada con éxito/Datos de la cuenta son modificados con éxito.
Flujo Alternativo:	

Caso de Uso:	AM Comisión
Descripción:	Adicionar o modificar Comisión
Actores:	Usuario: Administrativo
Precondición:	Que exista el cliente registrado en el sistema
Flujo normal	<ol style="list-style-type: none">1. El administrador introduce los datos o modifica los datos de la comisión2. El administrador graba los datos
Post condición:	La comisión es adicionada con éxito/Datos de la comisión son modificados con éxito.
Flujo Alternativo:	

3.3.3 DIAGRAMA DE CASO DE USO GESTIÓN DE TRANSACCIONES

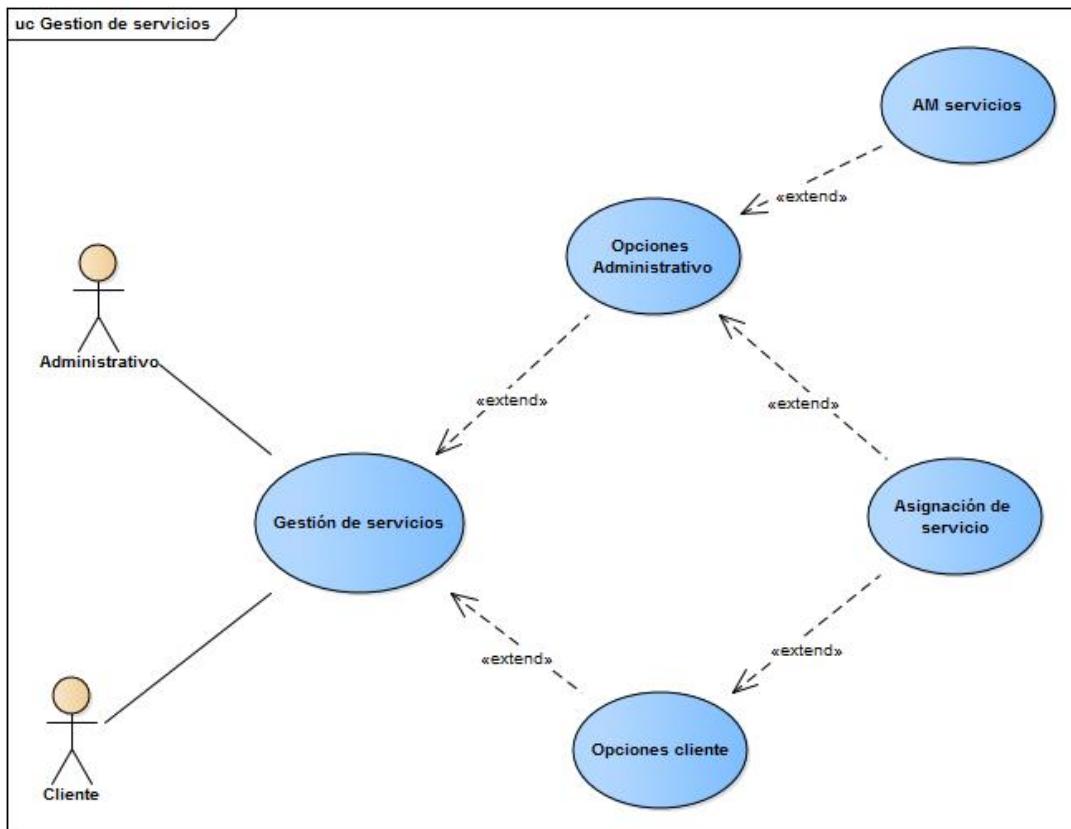


Caso de Uso:	Depósito a cuenta del cliente
Descripción:	Se realiza un ingreso o depósito a la cuenta del cliente la cual se utilizará más adelante para cancelar los servicios.
Actores:	Usuario: Administrativo
Precondición:	Que exista el cliente y la cuenta registrada en la Base de Datos.
Flujo normal	<ol style="list-style-type: none"> El administrador introduce los datos o modifica datos del depósito a la cuenta del cliente. El administrador graba los datos
Post condición:	El depósito se realiza con éxito
Flujo Alternativo:	Crear una cuenta asociada a un cliente

Caso de Uso:	Actualizar saldo cuenta cliente
Descripción:	Se realiza la actualización el saldo en la cuenta del cliente
Actores:	Usuario: Administrativo, Liquidador
Precondición:	Que exista una transacción de depósito o pago.
Flujo normal	<p>1. Al final de cada transacción se actualizan automáticamente el saldo de la cuenta del cliente.</p>
Post condición:	Se actualiza con éxito el saldo de la cuenta del cliente.
Flujo Alternativo:	

Caso de Uso:	Pago servicios cliente
Descripción:	El liquidador realiza el pago de un servicio del cliente
Actores:	Usuario: Liquidador
Precondición:	Que exista una transacción de pago de servicios.
Flujo normal	<p>1. El liquidador genera una transacción de pago de servicios</p> <p>2. Los datos del depósito se completan y guardan para generar los reportes para el cliente</p>
Post condición:	La transacción y el deposito se guardan correctamente
Flujo Alternativo:	

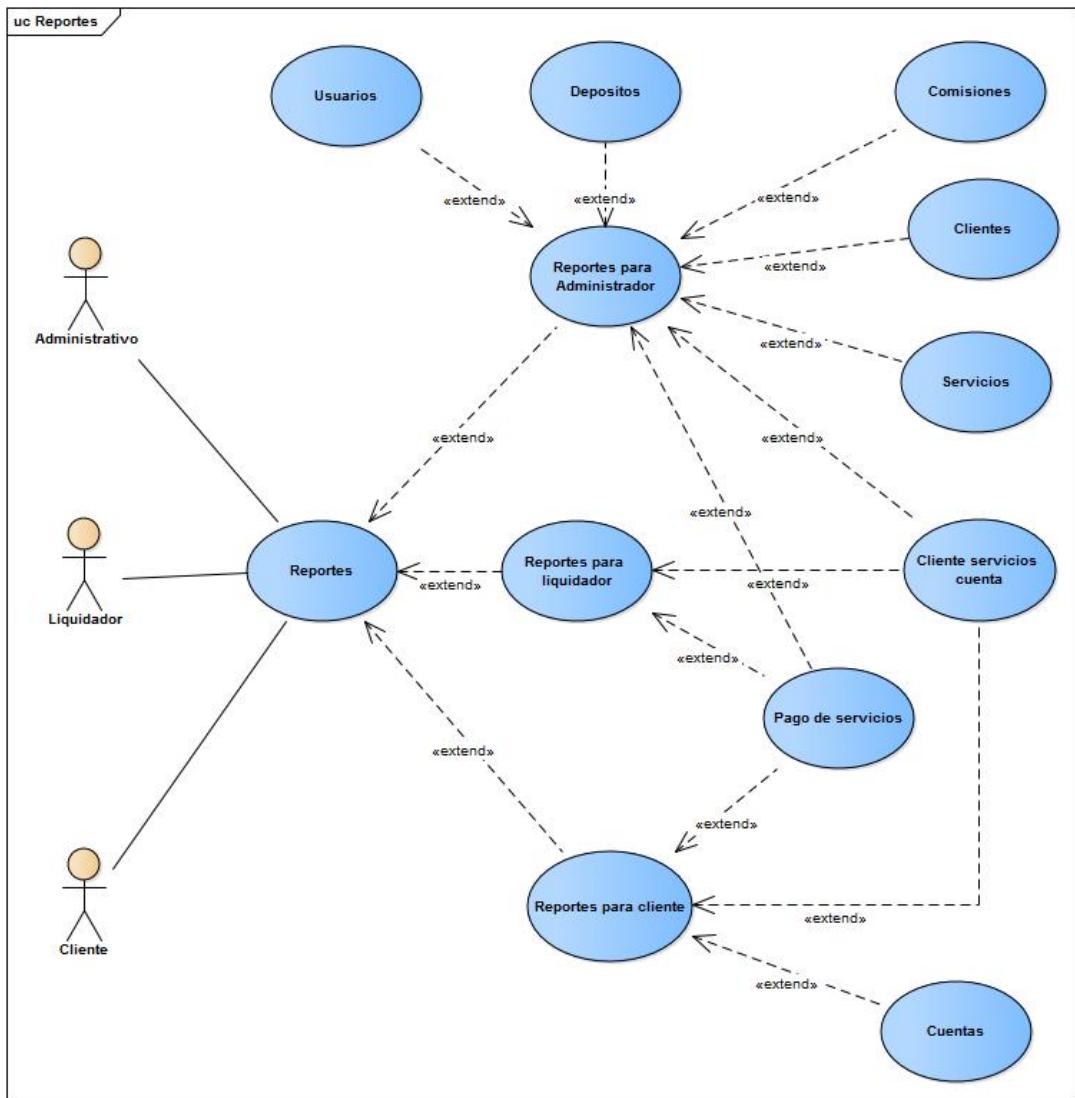
3.3.4 DIAGRAMA DE CASO DE USO GESTIÓN DE SERVICIOS



Caso de Uso:	AM servicios
Descripción:	El administrativo crea o modifica los servicios que pueden cancelar los clientes.
Actores:	Usuario: Administrativo
Precondición:	
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo crea o modifica los datos de los servicios que pueden pagar los clientes 2. El administrativo guarda los datos del servicio.
Post condición:	Los datos del servicio son guardados con éxito
Flujo Alternativo:	

Caso de Uso:	Asignación de servicio
Descripción:	Se asigna al cliente los servicios que éste puede cancelar
Actores:	Usuario: Administrativo, Cliente
Precondición:	Que exista el servicio previamente
Flujo normal	<ol style="list-style-type: none">1. El Administrativo o Cliente asigna uno o varios servicios al cliente.2. El administrativo o cliente guarda los datos de la asignación.
Post condición:	Los datos de la asignación son guardados con éxito
Flujo Alternativo:	Crear el servicio

3.3.5 DIAGRAMA DE CASO DE USO REPORTES



Caso de Uso:	Usuarios (Reporte)
Descripción:	Se realiza un listado de todos usuarios del sistema.
Actores:	Usuario: Administrativo
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo elige la opción de listado de todos los usuarios del sistema. 2. El Administrativo ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

Caso de Uso:	Depósitos (Reporte)
Descripción:	Se realiza un listado de todos depósitos según criterio: por fecha, por cliente.
Actores:	Usuario: Administrativo
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo elige algún criterio para realizar la consulta a la Base de Datos. 2. El Administrativo ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

Caso de Uso:	Comisiones (Reporte)
Descripción:	Se realiza un listado de las comisiones activas e inactivas
Actores:	Usuario: Administrativo
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo elige la opción para hacer la consulta a la Base de Datos. 2. El Administrativo ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

Caso de Uso:	Clientes (Reporte)
Descripción:	Se realiza un listado de todos clientes.
Actores:	Usuario: Administrativo
Precondición:	Que existan datos en la tabla respectiva a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo elige la opción para realizar la consulta a la Base de Datos. 2. El Administrativo ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

Caso de Uso:	Servicios (Reporte)
Descripción:	Se realiza un listado de todos servicios disponibles.
Actores:	Usuario: Administrativo
Precondición:	Que existan datos en la tabla respectiva a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Administrativo elige la opción para realizar la consulta a la Base de Datos. 2. El Administrativo ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

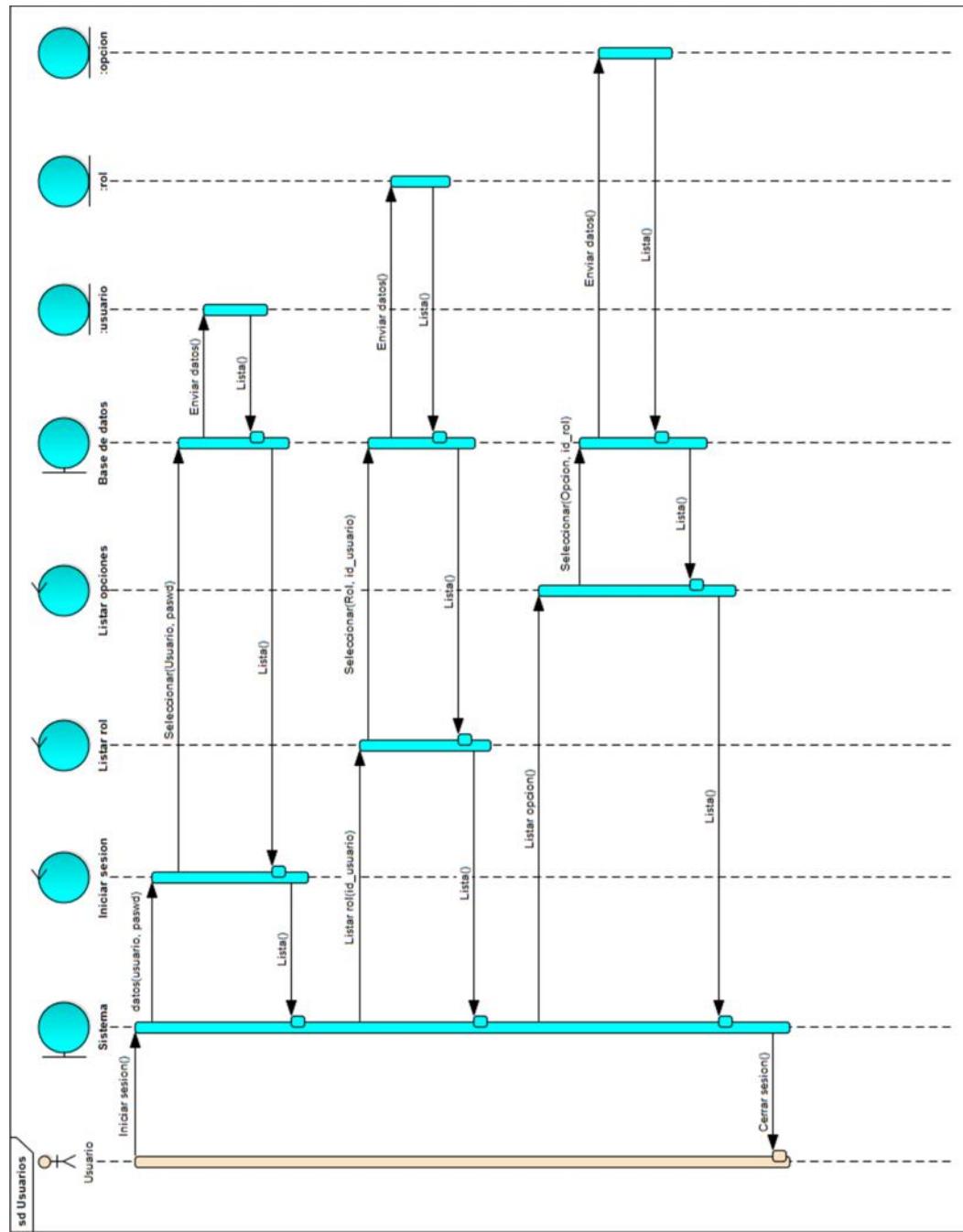
Caso de Uso:	Clientes servicios cuentas (Reporte)
Descripción:	Se realiza una listado de todos Clientes, servicios y cuentas.
Actores:	Usuario: Administrativo, Liquidador, Cliente
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. Se elige algún criterio para realizar la consulta a la Base de Datos. 2. Se ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

Caso de Uso:	Pago de servicios (Reporte)
Descripción:	Se realiza un listado de los pagos de los servicios
Actores:	Usuario: Administrativo, Liquidador, Cliente
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. Se elige algún criterio para realizar la consulta a la Base de Datos. 2. Se ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

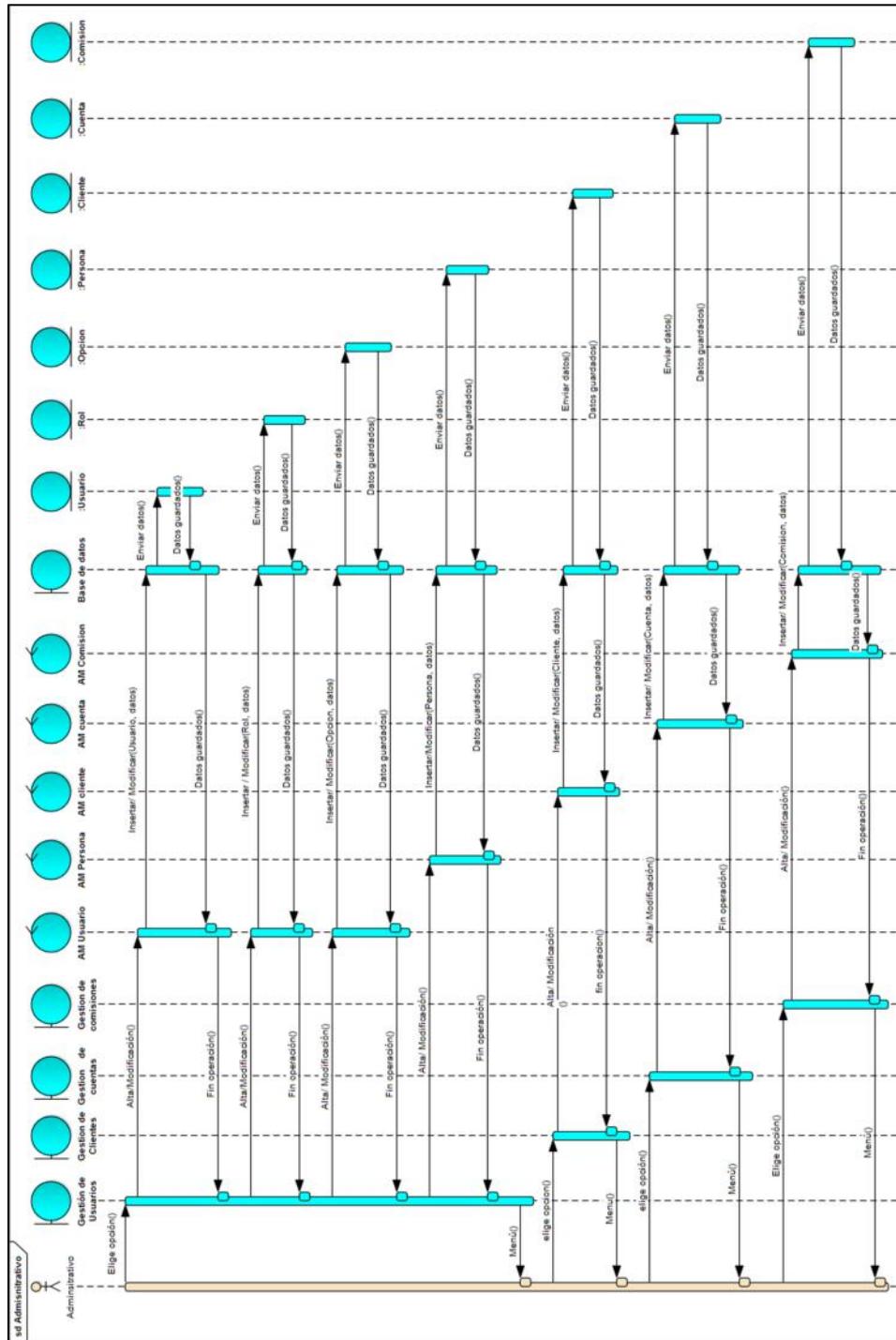
Caso de Uso:	Cuentas (Reporte)
Descripción:	Se realiza un listado del detalle de moviendo de las cuentas del cliente.
Actores:	Usuario: Cliente
Precondición:	Que existan datos en las tablas respectivas a la consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario elije un criterio para realizar la consulta a la Base de Datos. 2. Se ejecuta la consulta a la Base de Datos 3. El sistema le devuelve el listado solicitado.
Post condición:	
Flujo Alternativo:	

3.4 DIAGRAMAS DE SECUENCIA

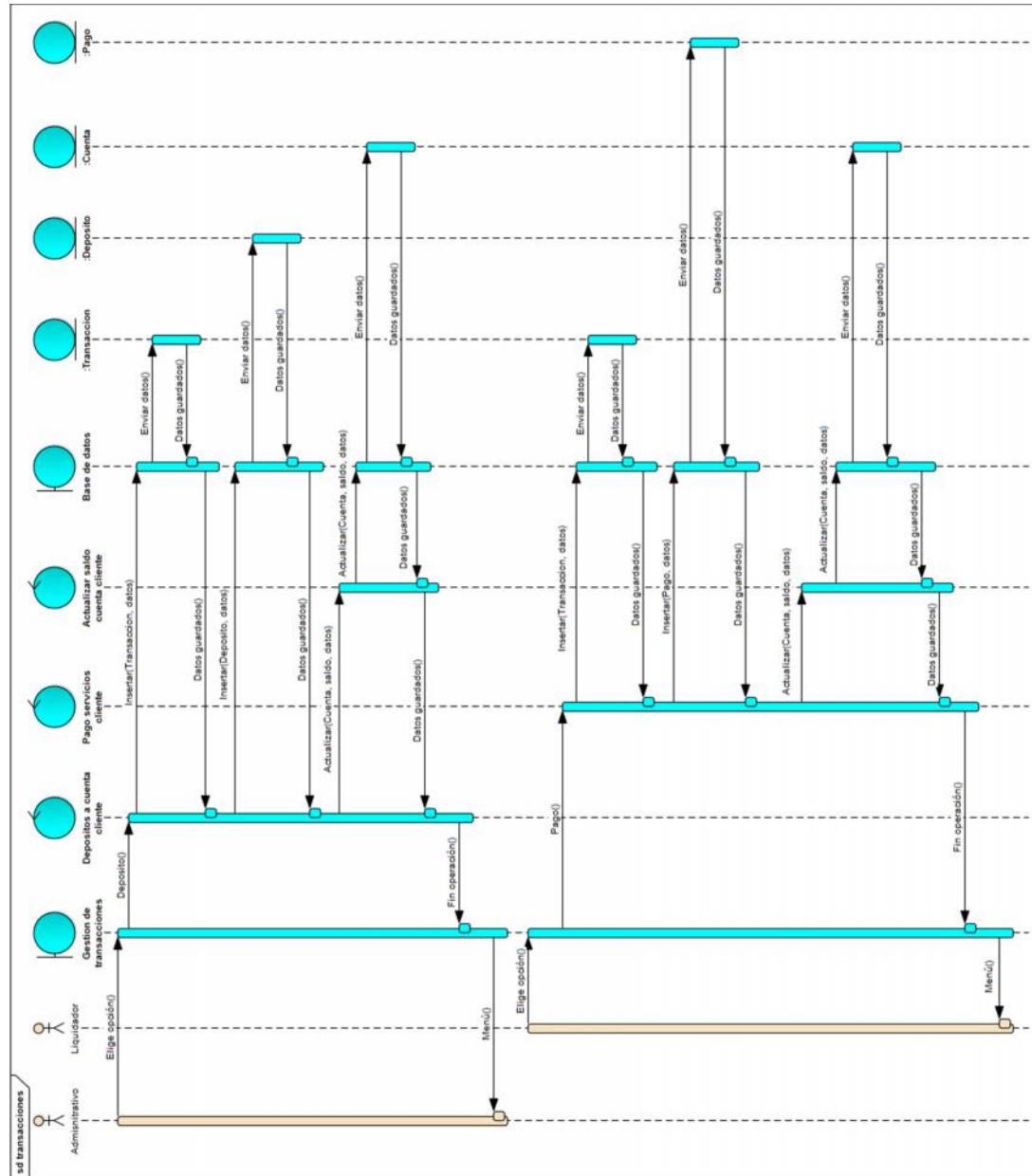
3.4.1 DIAGRAMAS DE SECUENCIA USUARIOS



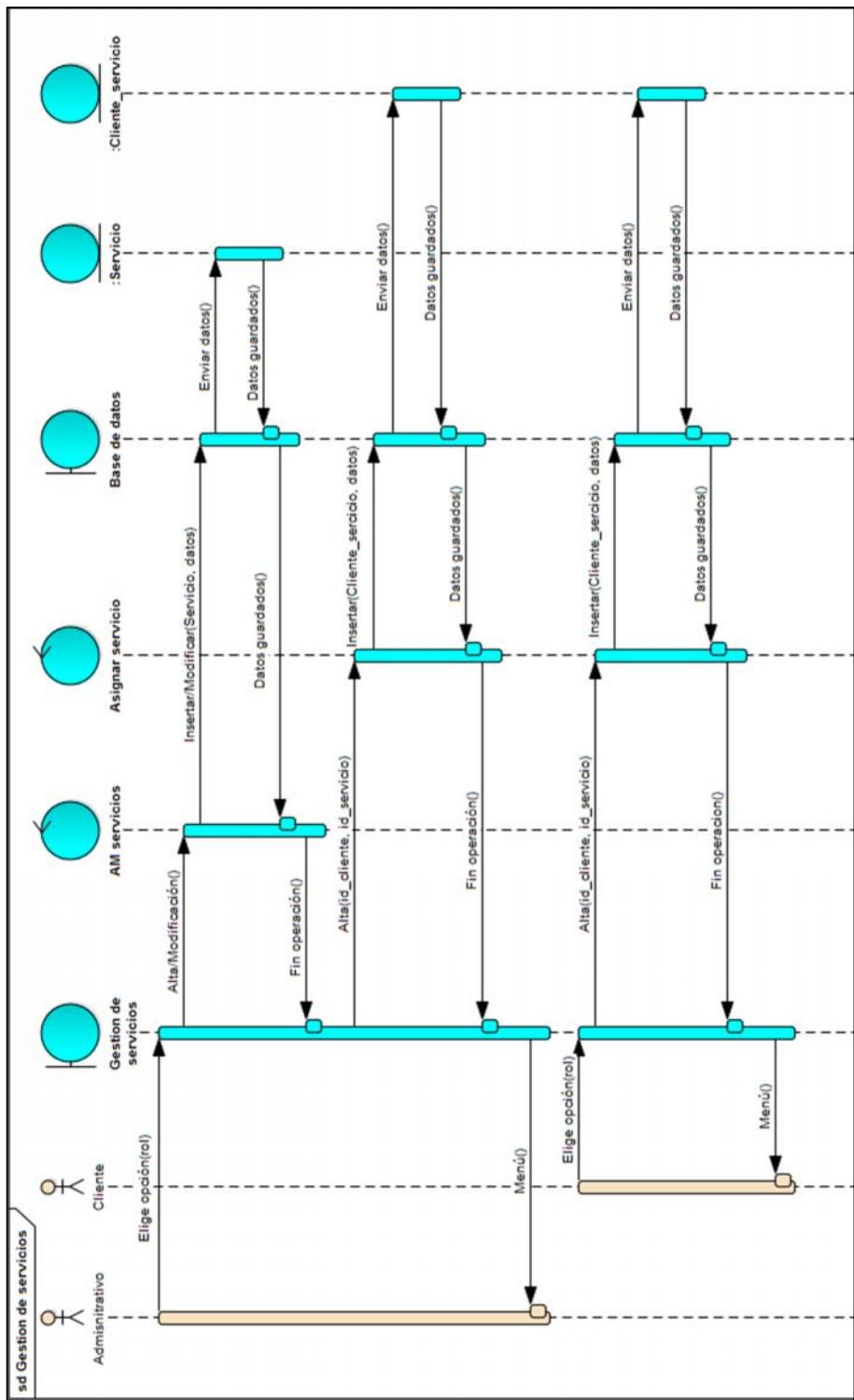
3.4.2 DIAGRAMAS DE SECUENCIA GESTIÓN ADMISNITRATIVA



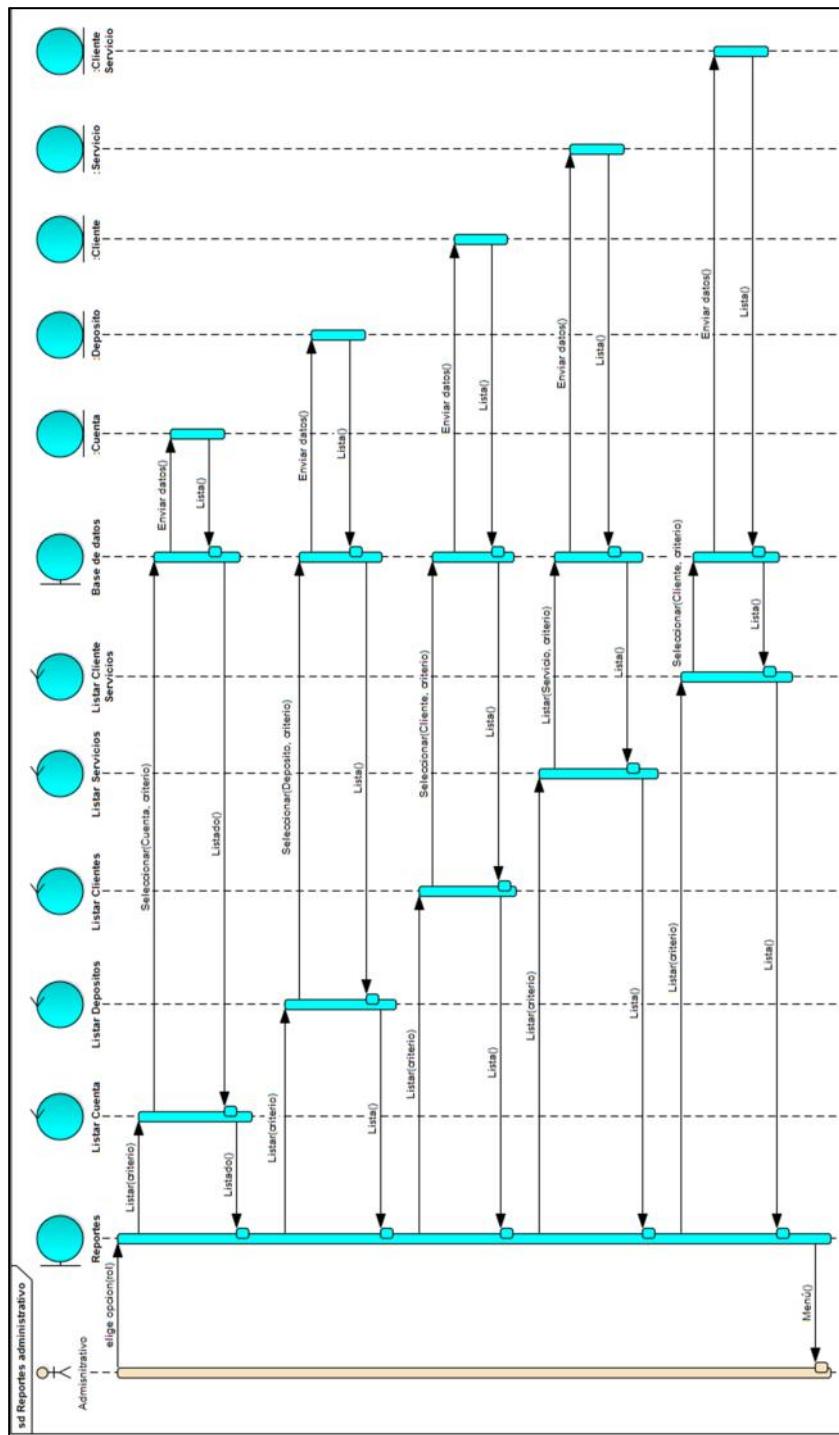
3.4.3 DIAGRAMAS DE SECUENCIA GESTIÓN DE TRANSACCIONES



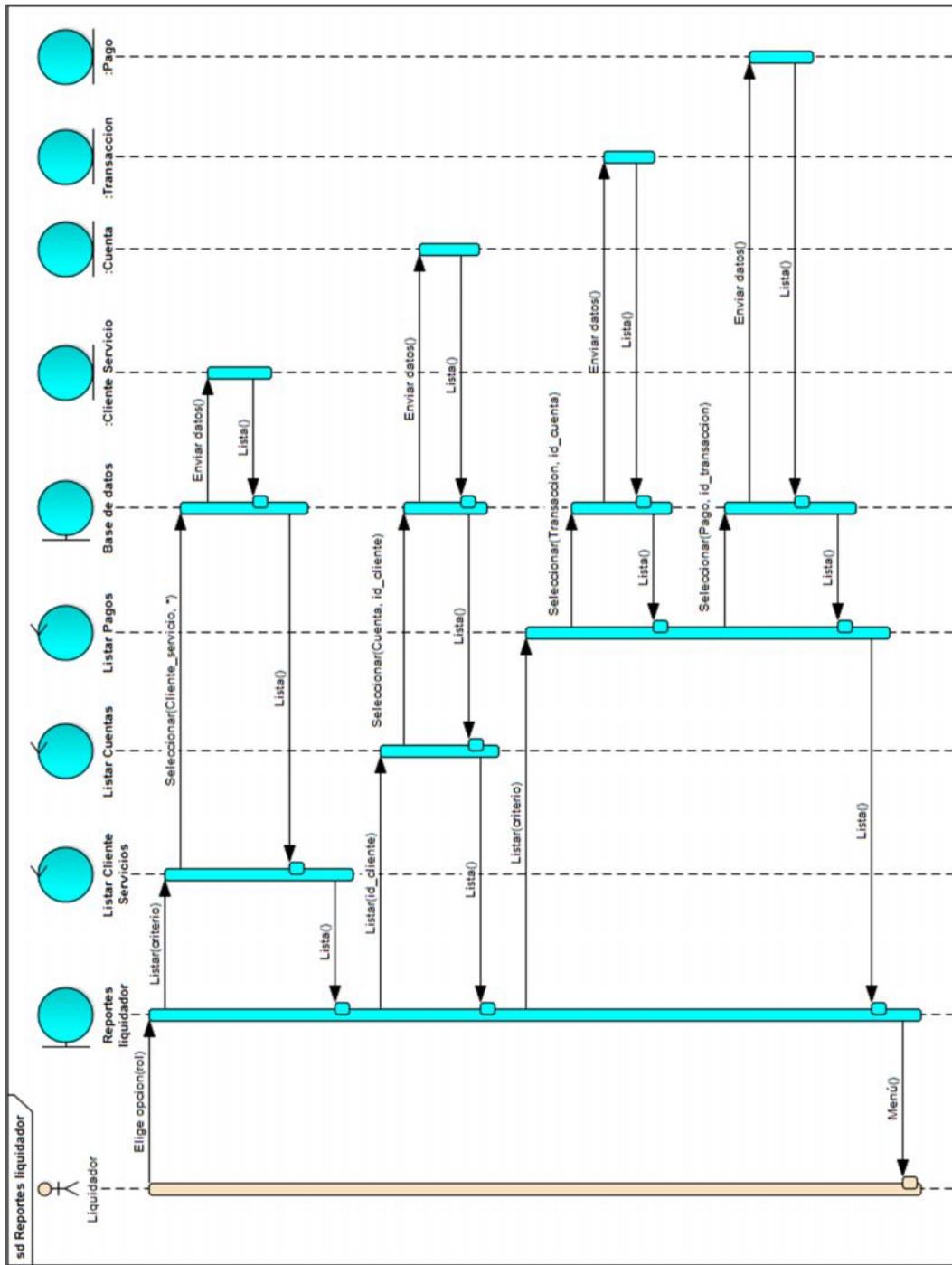
3.4.4 DIAGRAMAS DE SECUENCIA GESTIÓN SERVICIOS



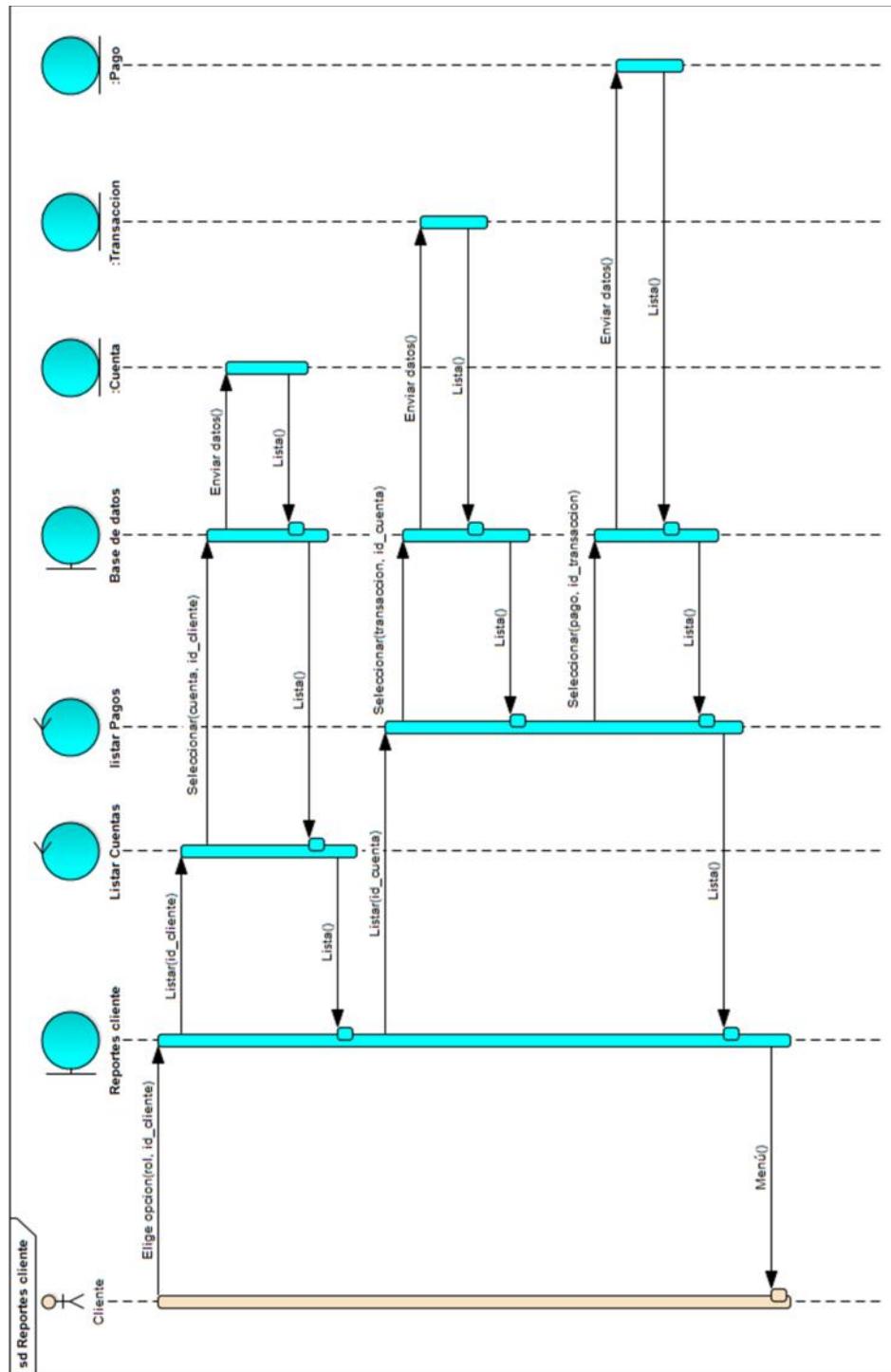
3.4.5 DIAGRAMA DE SECUENCIA REPORTES ADMISNITRATIVO



3.4.6 DIAGRAMAS DE SECUENCIA REPORTES LIQUIDADOR

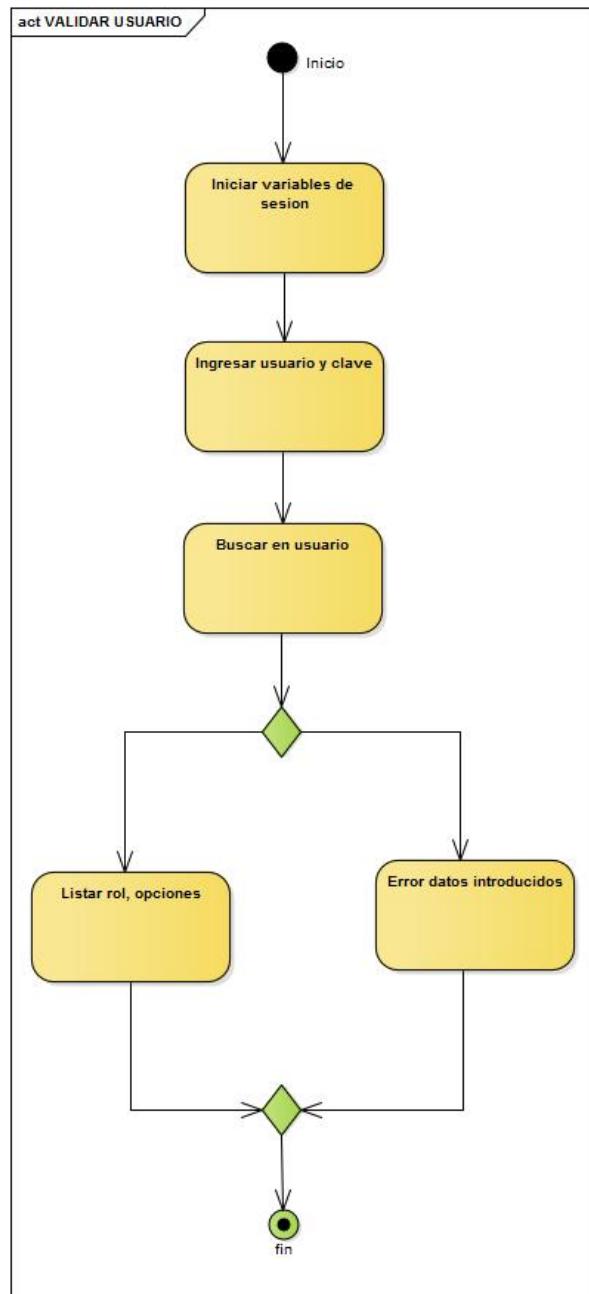


3.4.7 DIAGRAMA DE SECUENCIA REPORTES CLIENTE

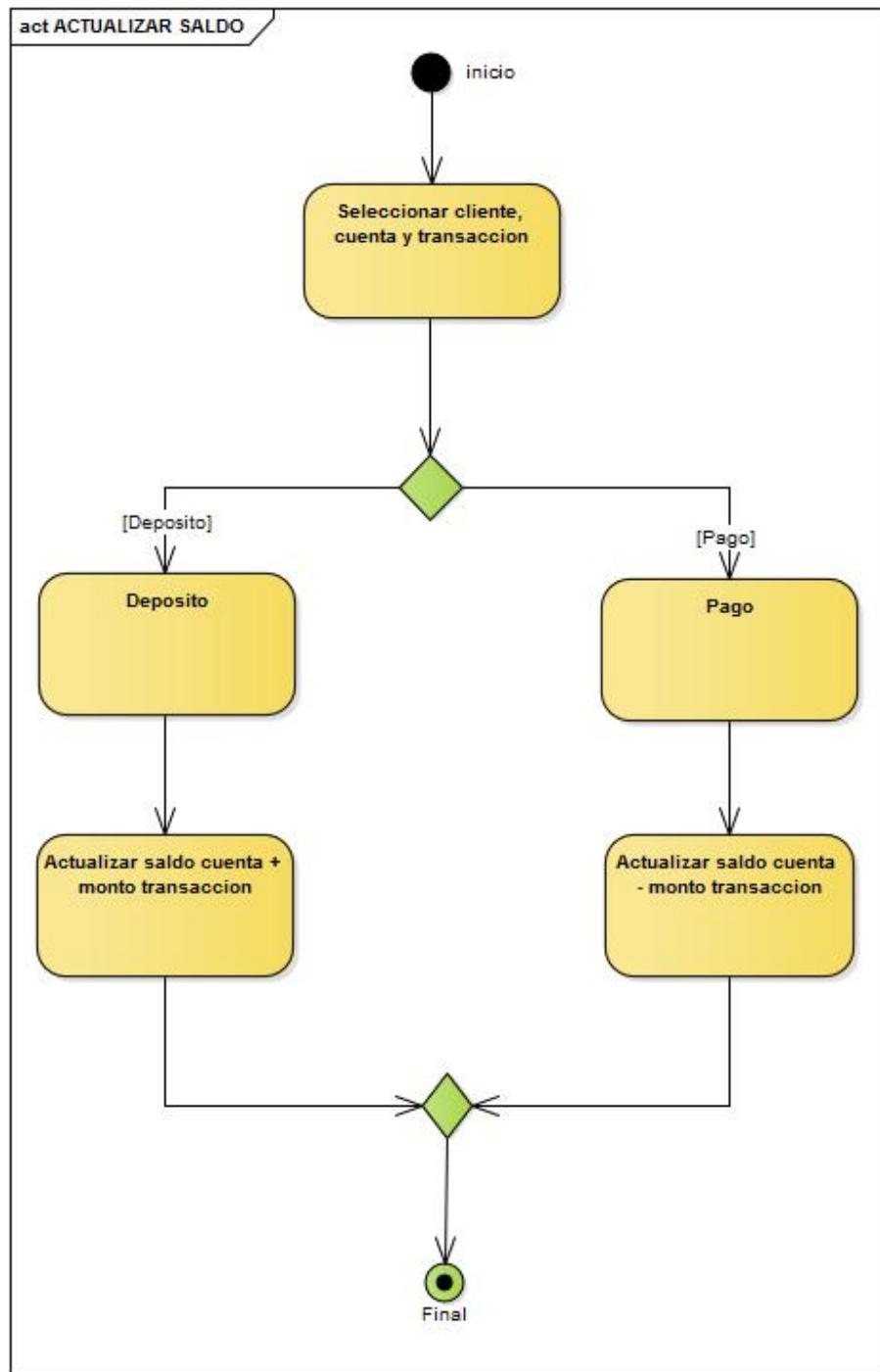


3.5 DIAGRAMA DE ACTIVIDADES

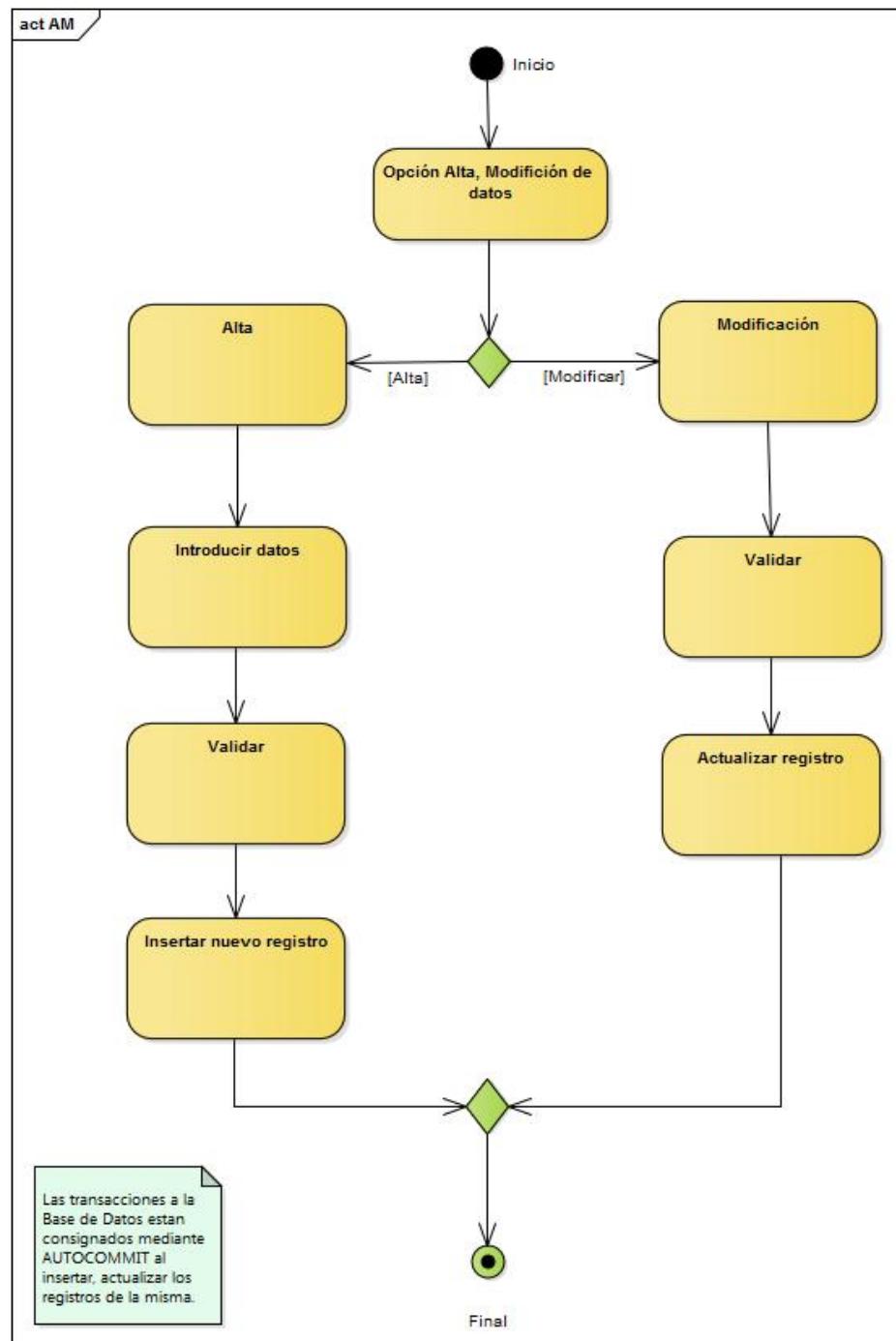
3.5.1 DIAGRAMA DE ACTIVIDADES VALIDAR USUARIO



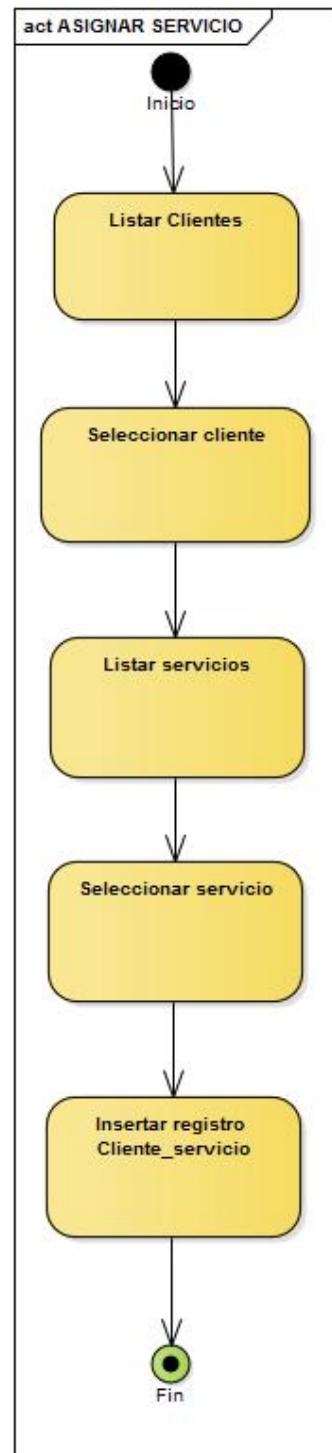
3.5.2 DIAGRAMA DE ACTIVIDADES ACTUALIZAR SALDO



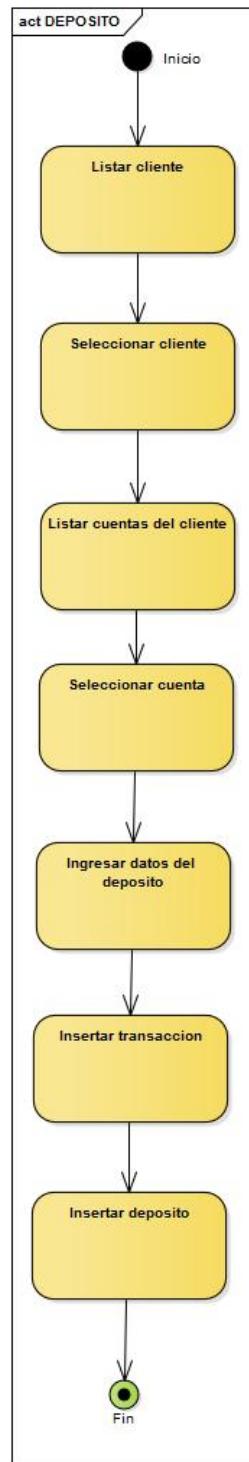
3.5.3 DIAGRAMA DE ACTIVIDADES GENÉRICO PARA ALTA/MODIFICACION DE DATOS



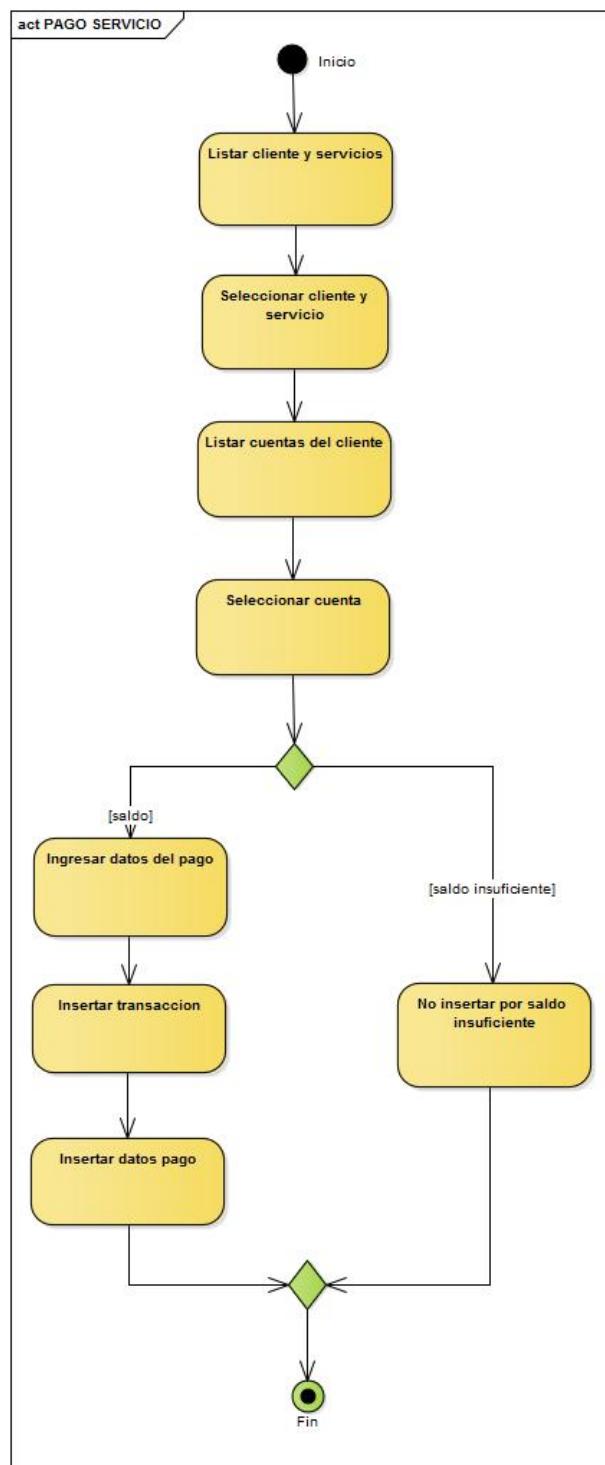
3.5.4 DIAGRAMA DE ACTIVIDADES ASIGNAR SERVICIO



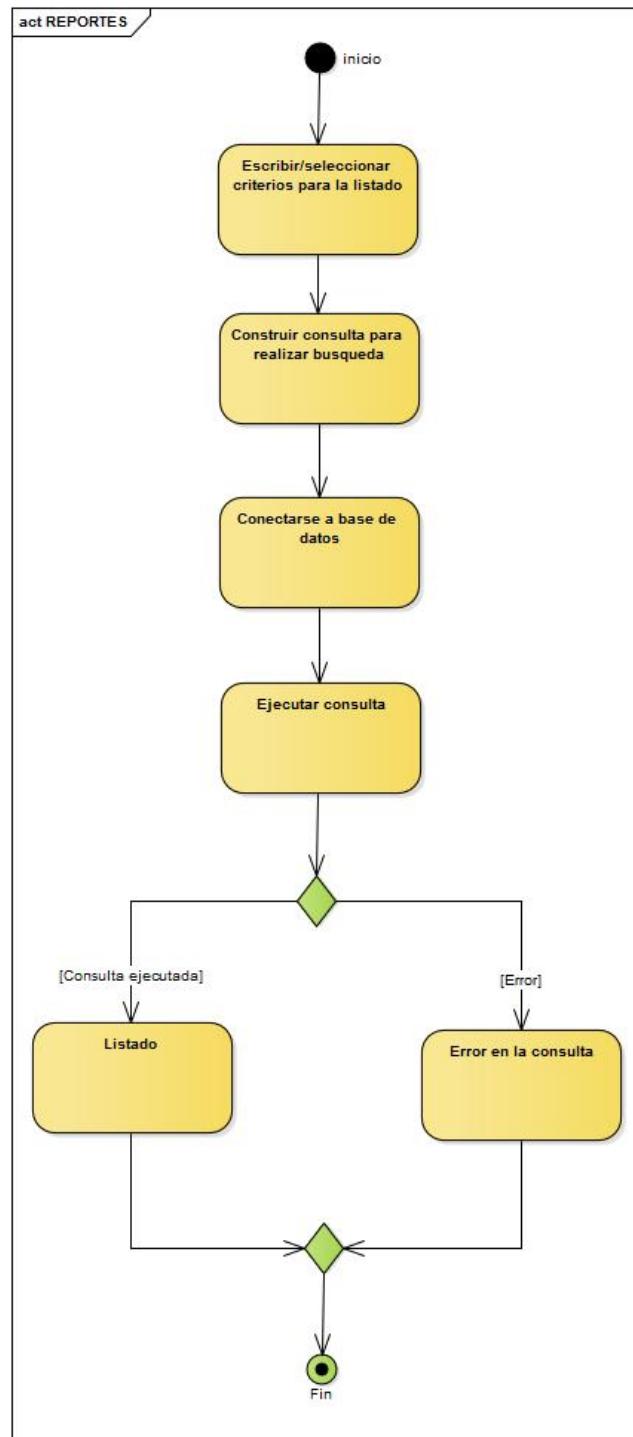
3.5.5 DIAGRAMA DE ACTIVIDADES REALIZAR DEPÓSITO



3.5.6 DIAGRAMA DE ACTIVIDADES PAGO SERVICIO



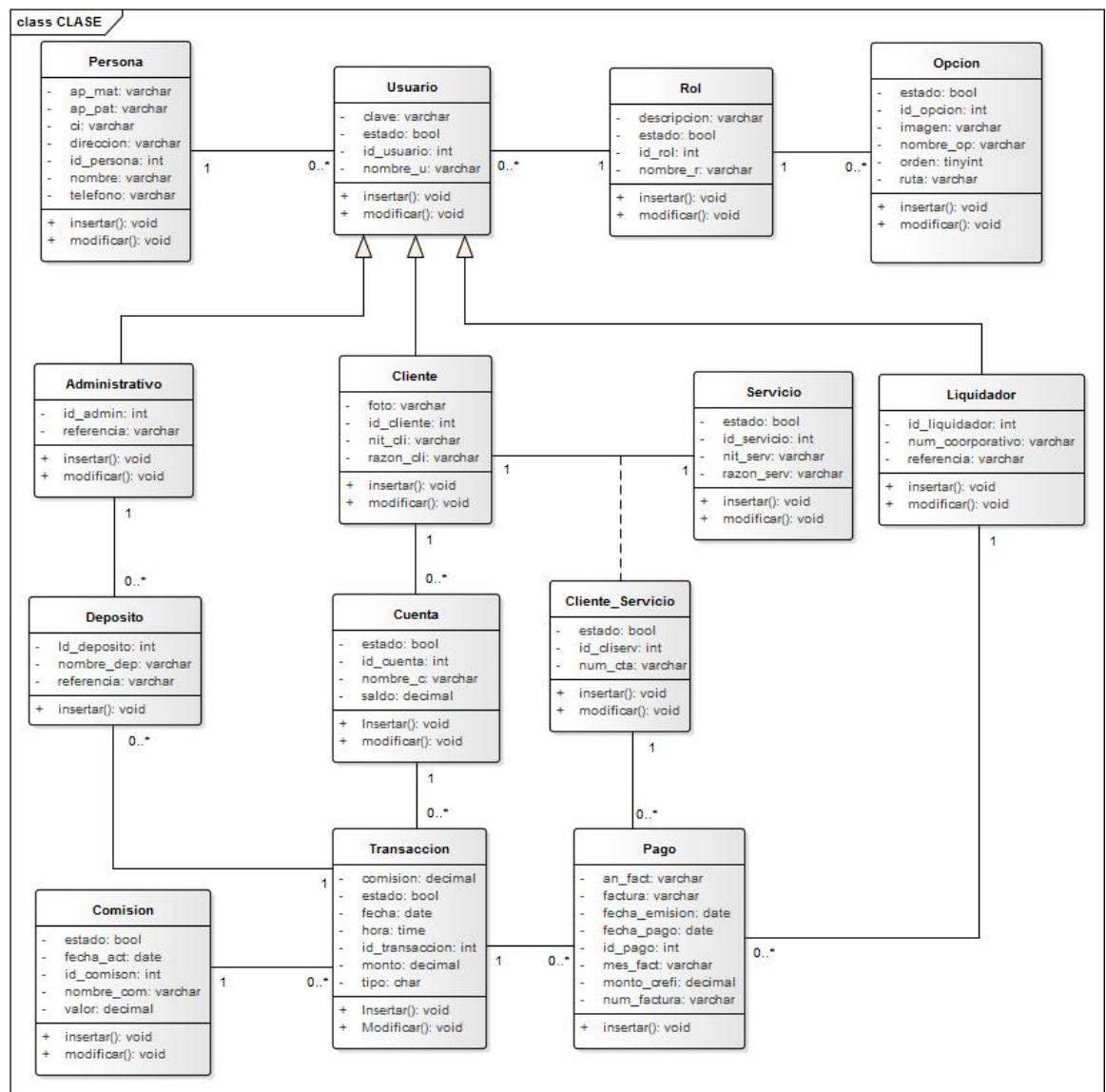
3.5.7 DIAGRAMA DE ACTIVIDADES GENÉRICO PARA REPORTES



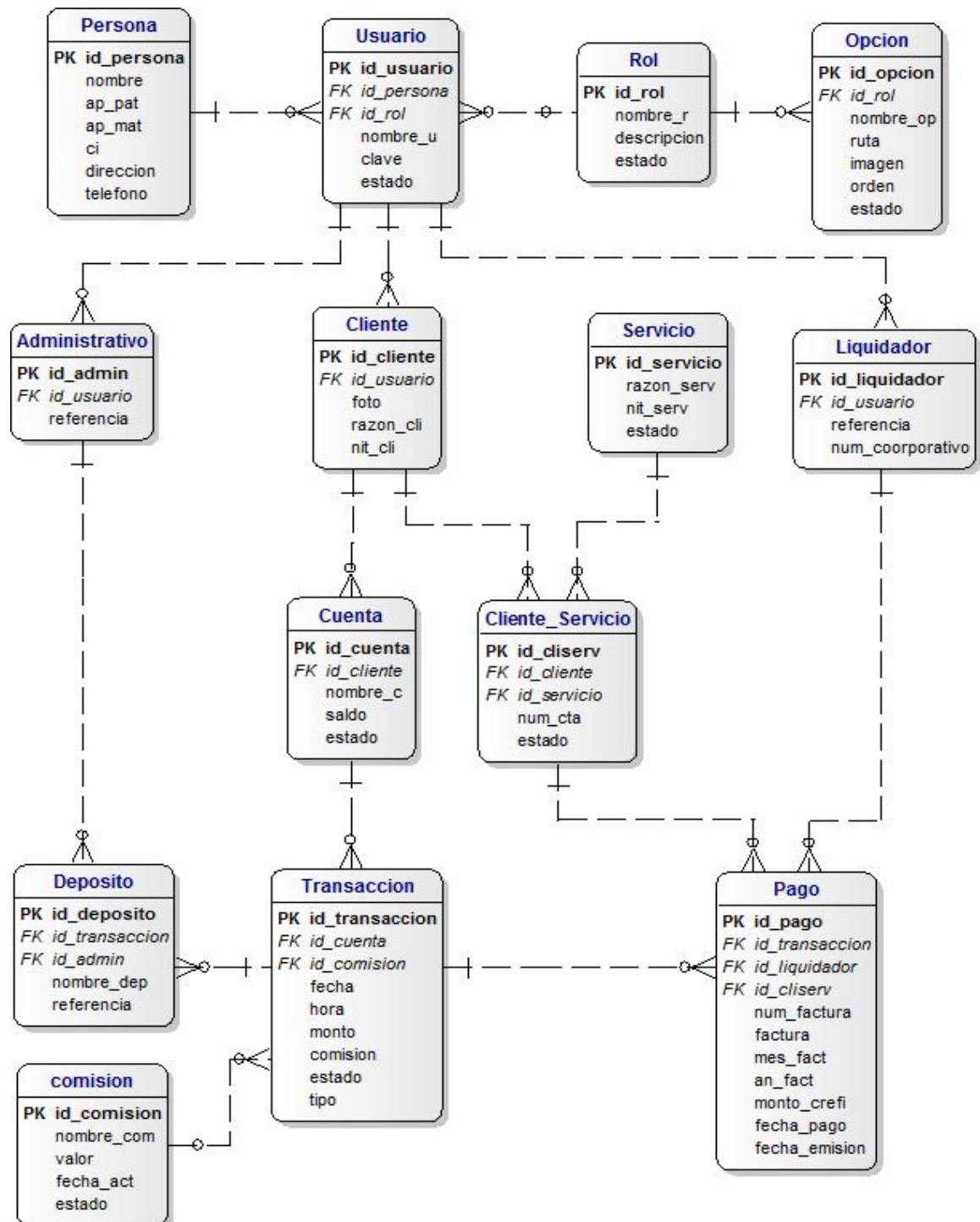
CAPITULO IV. DISEÑO DEL SISTEMA

4.1 DISEÑO DEL SISTEMA

4.1.1 DIAGRAMA DE CLASE DEL SISTEMA



4.1.2 DIAGRAMA DE BASE DE DATOS MODELADO ENTIDAD-RELACIÓN



4.1.3 DESCRIPCIÓN DE LAS TABLAS QUE COMPONEN LA BASE DE DATOS DEL SISTEMA

Lista de Tablas que componen el sistema

Nombre	Llave primaria	Columnas	Comentario
Administrativo	PK_Id_admisnitrativo	3	
Cliente	PK_Id_cliente	5	
Cliente_Servicio	PK_Cliente_Servicio	5	
Comision	PK_Comision	5	
Cuenta	PK_Cuenta	5	
Deposito	PK_Deposito	5	
Liquidador	PK_Liquidador	4	
Opcion	PK_Opcion	7	
Pago	PK_Pago	11	
Persona	PK_Persona	7	
Rol	PK_Rol	4	
Servicio	PK_Servicio	4	
Transaccion	PK_Transaccion	9	
Usuario	PK_Usuario	6	

 TABLA: Administrativo				
DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo				
Campo	Llave	Tipo dato	No Nulo	Descripción
Id_admin	PK	INTEGER	Si	Identificador del usuario administrativo (Llave primaria)
Id_usuario	FK	INTEGER	Si	Identificador del usuario (Llave foránea)
Referencia		VARCHAR (20)		Indica alguna referencia del personal administrativo

 **TABLA:** Cliente

DESCRIPCIÓN:

Campo	Llave	Tipo dato	No Nulo	Descripción
id_cliente	PK	INTEGER	Si	Identificador del cliente (Llave primaria)
id_usuario	FK	INTEGER	Si	Identificador del usuario (Llave foránea)
foto		VARCHAR (40)		Fotografía del cliente
razon_cli		VARCHAR (40)		Razón Social del cliente
nit_cli		VARCHAR (15)		Nit del cliente

 **TABLA:** Cliente Servicio

DESCRIPCION:

Campo	Llave	Tipo dato	No Nulo	Descripción
id_cliserv	PK	INTEGER	Si	Identificador del cliente-servicio (Llave primaria)
id_cliente	FK	INTEGER	Si	Identificador del cliente (Llave foránea)
id_servicio	FK	INTEGER	Si	Identificador del servicio (Llave foránea)
num_cta		VARCHAR(20)		Número de cuenta al que se debe pagar el servicio
estado		BOOLEAN		Estado de la asignación del servicio al cliente.

**TABLA:** Comisión

DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_comision	PK	INTEGER	Si	Identificador de la comisión (Llave primaria)
nombre_com	FK	VARCHAR(20)		Nombre de la comisión
valor		DECIMAL(10,2)		Porcentaje de comisión para pagos, para depósitos por defecto está reservado la primer fila con valor 0% de
fecha_act		DATE		Fecha de activación de la comisión
estado		BOOLEAN		Estado de la comisión

**TABLA:** Cuenta

DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_cuenta	PK	INTEGER	Si	Identificador de la cuenta (Llave primaria)
id_cliente	FK	INTEGER	Si	Identificador del cliente dueño de la cuenta (Llave foránea)
nombre_c		VARCHAR (20)		Nombre de la cuenta o propósito de la misma
saldo		DECIMAL(10,2)		Saldo actual de la cuenta en Bolivianos
estado		BOOLEAN		Estado de la cuenta 0=activa 1=inactiva

 TABLA: Depósito				
DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo				
Campo	Llave	Tipo dato	No Nulo	Descripción
id_deposito	PK	INTEGER	Si	Identificador del depósito (Llave primaria)
id_transaccion	FK	INTEGER	Si	Identificador de la transacción (Llave foránea)
id_admin		INTEGER	Si	Identificador del usuario administrador que realiza el depósito transacción (Llave foránea)
nombre_dep		VARCHAR(40)		Nombre o notas de la persona que realiza el depósito
referencia		VARCHAR(40)		Referencia de la persona que realiza el depósito

 TABLA: Liquidador				
DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo				
Campo	Llave	Tipo dato	No Nulo	Descripción
id_liquidador	PK	INTEGER	Si	Identificador del usuario tipo liquidador (Llave primaria)
id_usuario	FK	INTEGER	Si	Identificador del usuario (Llave foránea)
referencia		VARCHAR (40)		Indica alguna referencia del liquidador
num_coorporativo		VARCHAR(15)		Número telefónico corporativo

 **TABLA:** Opción

DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_opcion	PK	INTEGER	Si	Identificador de la opción del menú (Llave primaria)
id_rol	FK	INTEGER	Si	Identificador del rol al que pertenece la opción (Llave foránea)
nombre_op		VARCHAR (20)		Nombre de la opción del menú
ruta		VARCHAR(40)		Ruta donde se ejecutara el modulo que corresponde a la opción
imagen		VARCHAR(40)		Imagen del icono para la opción.
orden		TINYINT		Orden en el que aparecen las opciones del menú
estado		BOOLEAN		Estado de la opción 0=inactivo 1=activo

 **TABLA:** Pago

DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_pago	PK	INTEGER	Si	Identificador del pago (Llave primaria)
id_transaccion	FK	INTEGER	Si	Identificador de la transacción (Llave foránea)
id_liquidador		INTEGER	Si	Identificador del usuario liquidador que realiza el pago del servicio (Llave foránea)
id_cliserv		INTEGER	Si	Identificador del servicio asignado a un cliente que se paga (Llave foránea)
num_factura		VARCHAR(20)		Número de factura

factura		VARCHAR(40)		Dirección de la imagen de la factura
mes_fact		VARCHAR(2)		Mes de factura
an_fact		VARCHAR(4)		Año de la factura
monto_crefí		DECIMAL(10,2)		Monto válido para crédito fiscal
fecha_pago		DATE		Fecha de pago de la factura
fecha_emision		DATE		Fecha de emisión de la factura

 **TABLA:** Persona

DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_persona	PK	INTEGER	Si	Identificador de la persona (Llave primaria)
nombre		VARCHAR(20)		Nombre de la persona
ap_pat		VARCHAR (20)		Apellido paterno de la persona
ap_mat		VARCHAR(20)		Apellido materno de la persona
ci		VARCHAR(15)		Cédula de identidad de la persona
direccion		VARCHAR(40)		Dirección
telefono		VARCHAR(15)		Número de telefónico

 **TABLA:** Rol

DESCRIPCION: Tabla donde se almacena todos los usuarios de la categoría administrativo

Campo	Llave	Tipo dato	No Nulo	Descripción
id_rol	PK	INTEGER	Si	Identificador del rol del usuario(Llave primaria)
nombre_r		VARCHAR(20)		Nombre del rol
descripcion		VARCHAR(40)		Descripción del rol
estado		BOOLEAN		Estado del rol

 TABLA: Servicio				
DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo				
Campo	Llave	Tipo dato	No Nulo	Descripción
id_servicio	PK	INTEGER	Si	Identificador del servicio (Llave primaria)
razon_serv		VARCHAR (40)		Razón social del servicio
nit_serv		VARCHAR (15)		Nit de la entidad que ofrece el servicio
estado		BOOLEAN		Estado del servicio 0=inactivo 1=activo

 TABLA: Transacción				
DESCRIPCIÓN: Tabla donde se almacena todos los usuarios de la categoría administrativo				
Campo	Llave	Tipo dato	No Nulo	Descripción
id_transaccion	PK	INTEGER	Si	Identificador de la transacción (Llave primaria)
id_cuenta	FK	INTEGER	Si	Identificador de la cuenta(Llave foránea)
id_comision	FK	INTEGER	Si	Identificador de la comisión
fecha		DATE		Fecha actual del servidor de la transacción
hora		TIME		Hora actual del servidor
monto		DECIMAL(10,2)		Monto de la transacción
comision		DECIMAL(10,2)		Monto de la comisión por la transacción si es deposito a cuanta del cliente esta es 0 por defecto.
estado		BOOLEAN		Estado de la transacción 0=anulada 1=activa
tipo		CHAR(1)		Tipo de la transacción para identificar si es un Pago o Deposito

 **TABLA:** Usuario

DESCRIPCIÓN: Tabla donde se almacena los datos del usuario mismos que sirven para el ingreso al sistema.

Campo	Llave	Tipo dato	No Nulo	Descripción
id_usuario	PK	INTEGER	Si	Identificador del usuario (Llave primaria)
id_persona	FK	INTEGER	Si	Identificador de la persona (Llave foránea)
id_rol	FK	INTEGER	Si	Identificador del Rol dentro del sistema: Administrativo, Liquidador, Cliente (Llave foránea)
nombre_u		VARCHAR(15)		Nombre del usuario usado para ingreso al sistema
clave		VARCHAR(40)		Se almacena la clave del usuario en codificación MD5
estado		BOOLEAN		Indica el estado del usuario 0=inactivo 1=Activo

4.1.4 GENERACIÓN DE TABLAS EN SQL: MOTOR DE BASE DE DATOS MYSQL 5

```

CREATE TABLE `Usuario` (
    `id_usuario` INTEGER NOT NULL AUTO_INCREMENT,
    `id_persona` INTEGER NOT NULL,
    `id_rol` INTEGER,
    `nombre_u` VARCHAR(15),
    `clave` VARCHAR(40),
    `estado` BOOL,
    CONSTRAINT `PK_Usuario` PRIMARY KEY (`id_usuario`)
);

CREATE TABLE `Persona` (
    `id_persona` INTEGER NOT NULL AUTO_INCREMENT,
    `nombre` VARCHAR(20) NOT NULL,
    `ap_pat` VARCHAR(20),
    `ap_mat` VARCHAR(20),
    `ci` VARCHAR(15),
    `direccion` VARCHAR(40),
    `telefono` VARCHAR(15),
    CONSTRAINT `PK_Persona` PRIMARY KEY (`id_persona`)
);

CREATE TABLE `Rol` (
    `id_rol` INTEGER NOT NULL AUTO_INCREMENT,
    `nombre_r` VARCHAR(20),
    `descripcion` VARCHAR(40),
    `estado` BOOL,
    CONSTRAINT `PK_Rol` PRIMARY KEY (`id_rol`)
);

CREATE TABLE `Opcion` (
    `id_opcion` INTEGER NOT NULL AUTO_INCREMENT,
    `id_rol` INTEGER NOT NULL,
    `nombre_op` VARCHAR(20),
    `ruta` VARCHAR(40),
    `imagen` VARCHAR(40),
    `orden` TINYINT,
    `estado` BOOL,
    CONSTRAINT `PK_Opcion` PRIMARY KEY (`id_opcion`)
);

```

```

CREATE TABLE `Liquidador` (
  `id_liquidador` INTEGER NOT NULL AUTO_INCREMENT,
  `id_usuario` INTEGER NOT NULL,
  `referencia` VARCHAR(40),
  `num_corporativo` VARCHAR(15),
  CONSTRAINT `PK_Liquidador` PRIMARY KEY
  (`id_liquidador`)
);

CREATE TABLE `Cliente` (
  `id_cliente` INTEGER NOT NULL AUTO_INCREMENT,
  `id_usuario` INTEGER NOT NULL,
  `foto` VARCHAR(40),
  `razon_cli` VARCHAR(40),
  `nit_cli` VARCHAR(15),
  PRIMARY KEY (`id_cliente`)
);

CREATE TABLE `Administrativo` (
  `id_admin` INTEGER NOT NULL AUTO_INCREMENT,
  `id_usuario` INTEGER NOT NULL,
  `referencia` VARCHAR(20),
  PRIMARY KEY (`id_admin`)
);

CREATE TABLE `Servicio` (
  `id_servicio` INTEGER NOT NULL AUTO_INCREMENT,
  `razon_serv` VARCHAR(40),
  `nit_serv` VARCHAR(15),
  `estado` BOOL,
  CONSTRAINT `PK_Servicio` PRIMARY KEY (`id_servicio`)
);

CREATE TABLE `Cuenta` (
  `id_cuenta` INTEGER NOT NULL AUTO_INCREMENT,
  `id_cliente` INTEGER NOT NULL,
  `nombre_c` VARCHAR(40),
  `saldo` decimal(10,2),
  `estado` BOOL,
  CONSTRAINT `PK_Cuenta` PRIMARY KEY (`id_cuenta`)
);

```

```

CREATE TABLE `Transaccion` (
    `id_transaccion` INTEGER NOT NULL AUTO_INCREMENT,
    `id_cuenta` INTEGER NOT NULL,
    `id_comision` INTEGER,
    `fecha` DATE,
    `hora` TIME,
    `monto` DECIMAL(10,2),
    `comision` DECIMAL(10,2),
    `estado` BOOL,
    `tipo` CHAR(1),
    CONSTRAINT `PK_Transaccion` PRIMARY KEY (`id_transaccion`)
);

```

```

CREATE TABLE `Deposito` (
    `id_deposito` INTEGER NOT NULL AUTO_INCREMENT,
    `id_transaccion` INTEGER NOT NULL,
    `id_admin` INTEGER NOT NULL,
    `nombre_dep` VARCHAR(40),
    `referencia` VARCHAR(40),
    CONSTRAINT `PK_Deposito` PRIMARY KEY (`id_deposito`)
);

```

```

CREATE TABLE `Pago` (
    `id_pago` INTEGER NOT NULL AUTO_INCREMENT,
    `id_transaccion` INTEGER NOT NULL,
    `id_liquidador` INTEGER NOT NULL,
    `id_cliserv` INTEGER NOT NULL,
    `num_factura` VARCHAR(20),
    `factura` VARCHAR(40),
    `mes_fact` VARCHAR(2),
    `an_fact` VARCHAR(4),
    `monto_crefi` decimal(10,2),
    `fecha_pago` DATE,
    `fecha_emision` DATE,
    CONSTRAINT `PK_Pago` PRIMARY KEY (`id_pago`)
);

```

```

CREATE TABLE `Cliente_Servicio` (
    `id_cliserv` INTEGER NOT NULL AUTO_INCREMENT,
    `id_cliente` INTEGER NOT NULL,
    `id_servicio` INTEGER NOT NULL,
    `num_cta` VARCHAR(20),
    `estado` BOOL,
    CONSTRAINT `PK_Cliente_Servicio` PRIMARY KEY
    (`id_cliserv`)
);

CREATE TABLE `comision` (
    `id_comision` INTEGER NOT NULL AUTO_INCREMENT,
    `nombre_com` VARCHAR(20),
    `valor` decimal(10,2),
    `fecha_act` DATE,
    `estado` BOOL,
    CONSTRAINT `PK_comision` PRIMARY KEY (`id_comision`)
);

```

RESTRICCIONES DE CLAVES FORÁNEAS

```

ALTER TABLE `Usuario` ADD CONSTRAINT `Persona_Usuario`
    FOREIGN KEY (`id_persona`) REFERENCES `Persona`
    (`id_persona`);

ALTER TABLE `Usuario` ADD CONSTRAINT `Rol_Usuario`
    FOREIGN KEY (`id_rol`) REFERENCES `Rol` (`id_rol`);

ALTER TABLE `Opcion` ADD CONSTRAINT `Rol_Opcion`
    FOREIGN KEY (`id_rol`) REFERENCES `Rol` (`id_rol`);

ALTER TABLE `Liquidador` ADD CONSTRAINT
    `Usuario_Liquidador`
    FOREIGN KEY (`id_usuario`) REFERENCES `Usuario`
    (`id_usuario`);

ALTER TABLE `Cliente` ADD CONSTRAINT `Usuario_Cliente`
    FOREIGN KEY (`id_usuario`) REFERENCES `Usuario`
    (`id_usuario`);

```

```

ALTER TABLE `Administrativo` ADD CONSTRAINT
`Usuario_Administrativo`
FOREIGN KEY (`id_usuario`) REFERENCES `Usuario`(`id_usuario`);

ALTER TABLE `Cuenta` ADD CONSTRAINT `Cliente_Cuenta`
FOREIGN KEY (`id_cliente`) REFERENCES `Cliente`(`id_cliente`);

ALTER TABLE `Transaccion` ADD CONSTRAINT
`Cuenta_Transaccion`
FOREIGN KEY (`id_cuenta`) REFERENCES `Cuenta`(`id_cuenta`);

ALTER TABLE `Transaccion` ADD CONSTRAINT
`comision_Transaccion`
FOREIGN KEY (`id_comision`) REFERENCES `comision`(`id_comision`);

ALTER TABLE `Deposito` ADD CONSTRAINT
`Transaccion_Deposito`
FOREIGN KEY (`id_transaccion`) REFERENCES `Transaccion`(`id_transaccion`);

ALTER TABLE `Deposito` ADD CONSTRAINT
`Administrativo_Deposito`
FOREIGN KEY (`id_admin`) REFERENCES `Administrativo`(`id_admin`);

ALTER TABLE `Pago` ADD CONSTRAINT `Transaccion_Pago`
FOREIGN KEY (`id_transaccion`) REFERENCES `Transaccion`(`id_transaccion`);

ALTER TABLE `Pago` ADD CONSTRAINT `Cliente_Servicio_Pago`
FOREIGN KEY (`id_cliserv`) REFERENCES `Cliente_Servicio`(`id_cliserv`);

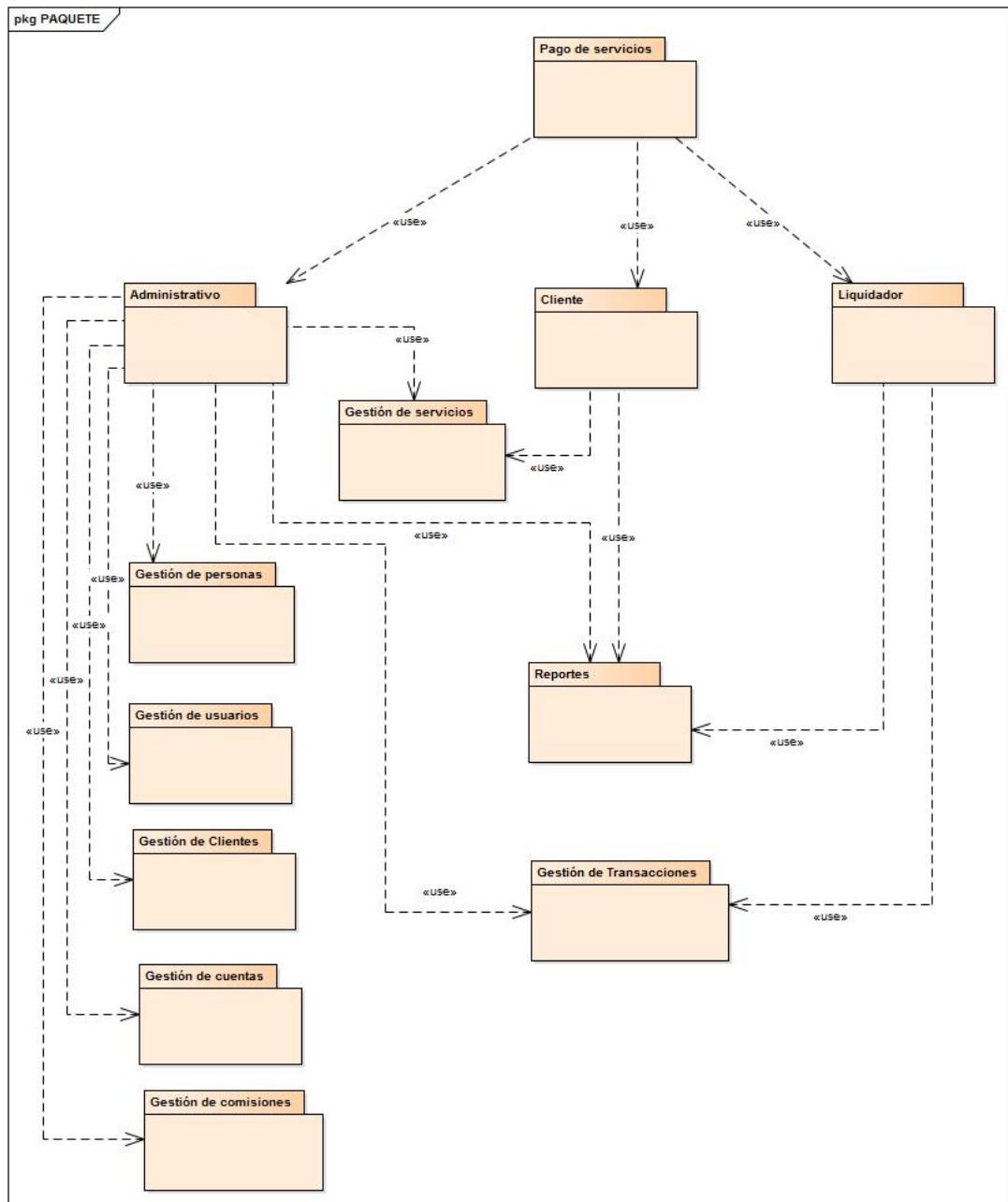
ALTER TABLE `Pago` ADD CONSTRAINT `Liquidador_Pago`
FOREIGN KEY (`id_liquidador`) REFERENCES `Liquidador`(`id_liquidador`);

ALTER TABLE `Cliente_Servicio` ADD CONSTRAINT
`Cliente_Cliente_Servicio`
FOREIGN KEY (`id_cliente`) REFERENCES `Cliente`(`id_cliente`);

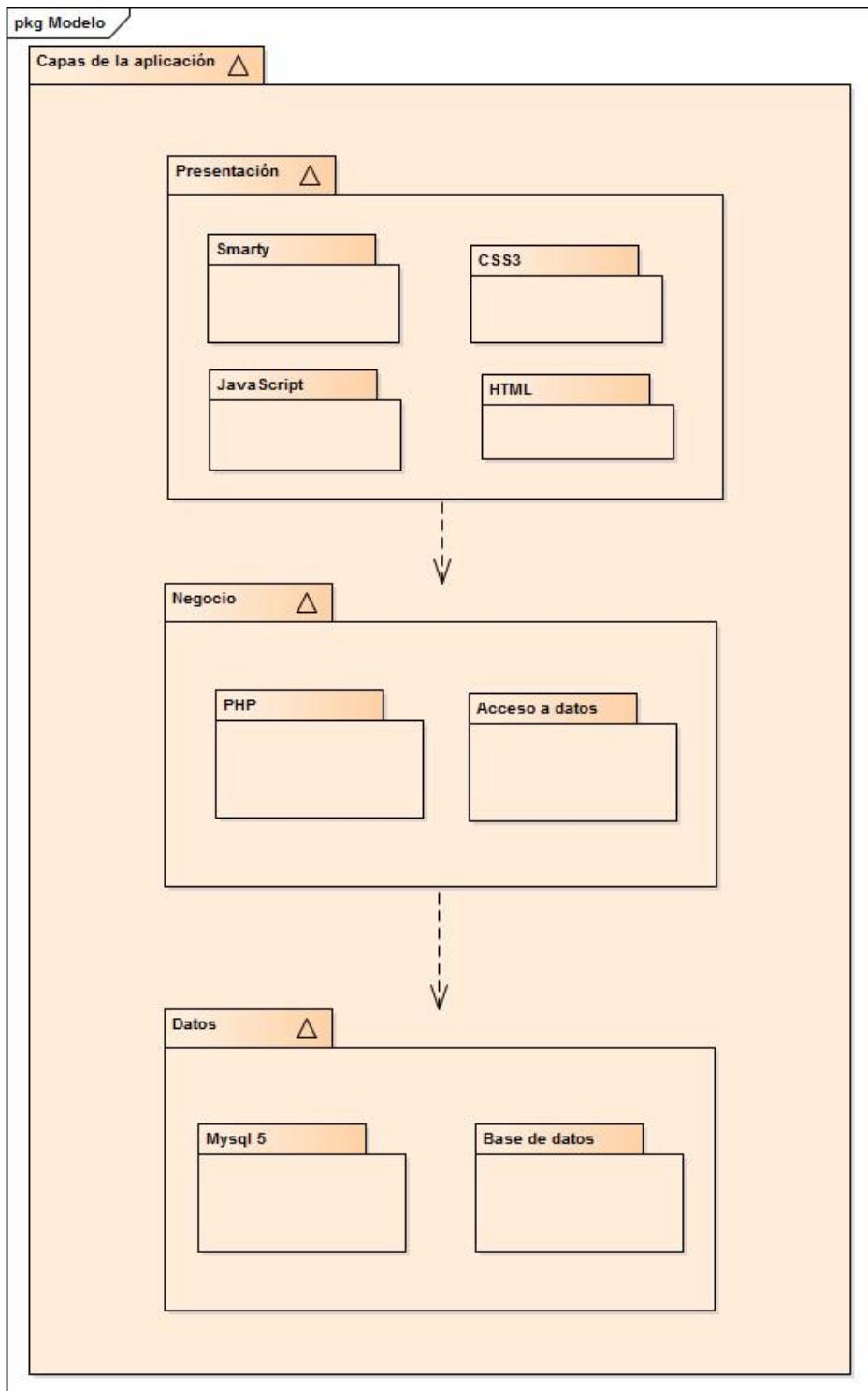
ALTER TABLE `Cliente_Servicio` ADD CONSTRAINT
`Servicio_Cliente_Servicio`
FOREIGN KEY (`id_servicio`) REFERENCES `Servicio`(`id_servicio`);

```

1.7.1. DIAGRAMA DE PAQUETE

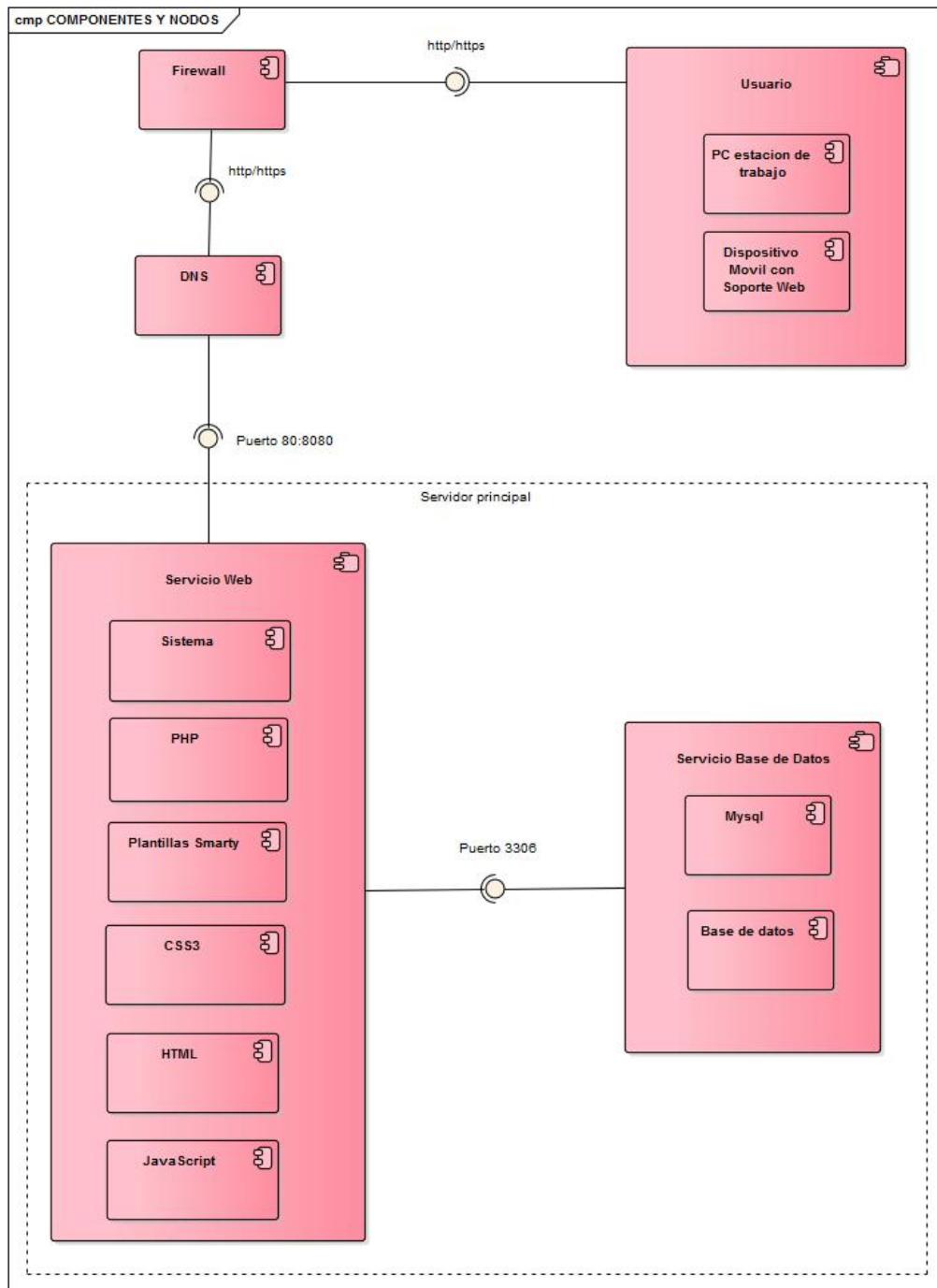


4.1.5 DIAGRAMA DE MODELO

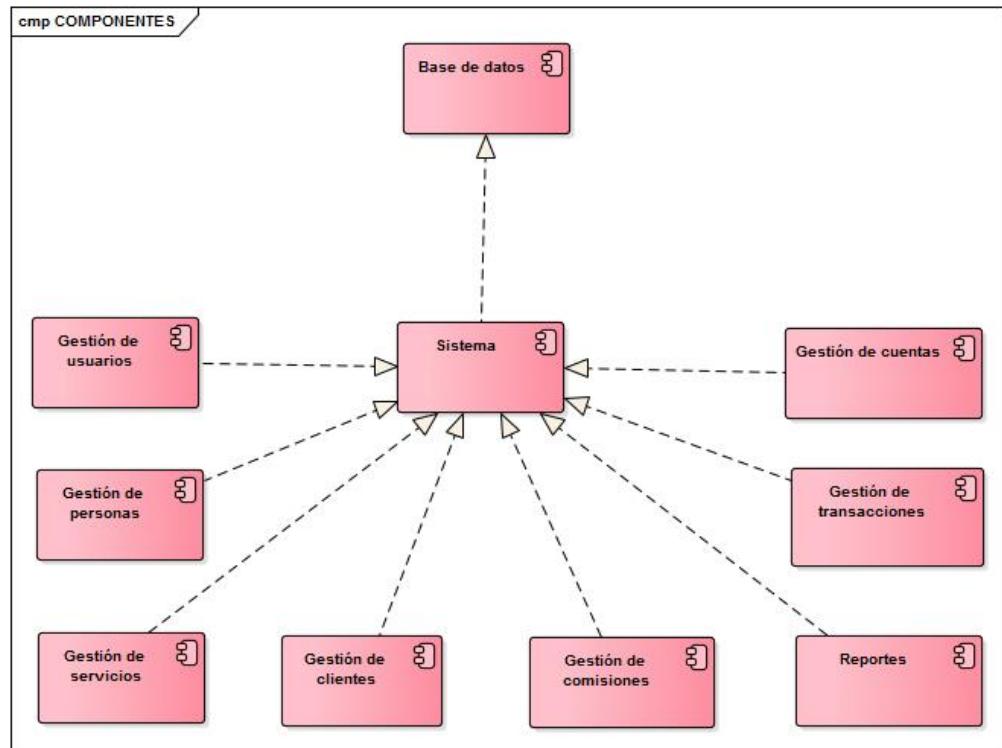


4.1.6 DIAGRAMA DE COMPONENTES

4.1.6.1 DIAGRAMA DE COMPONENTES Y NODOS

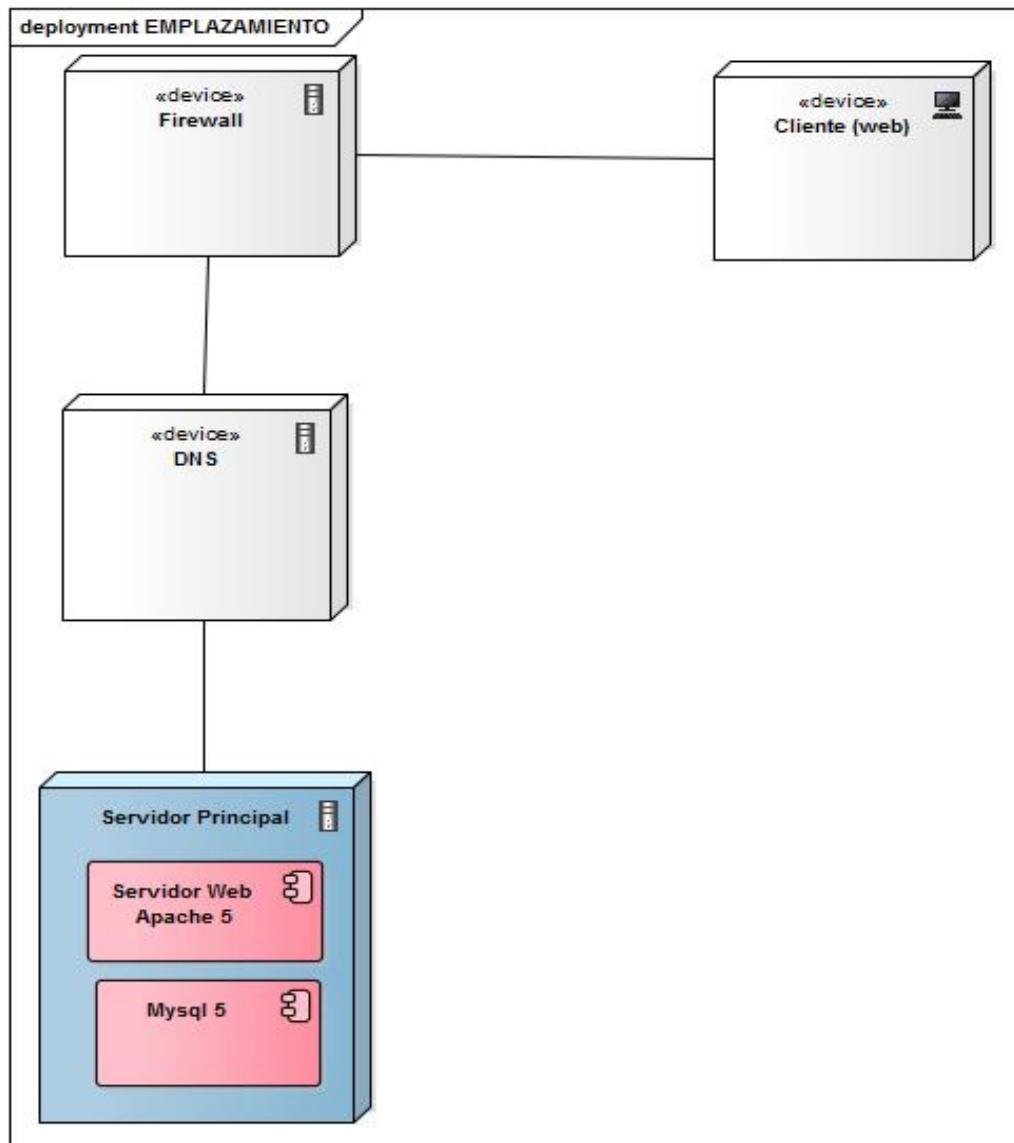


4.1.6.2 DIAGRAMA DE COMPONENTES DEL SISTEMA

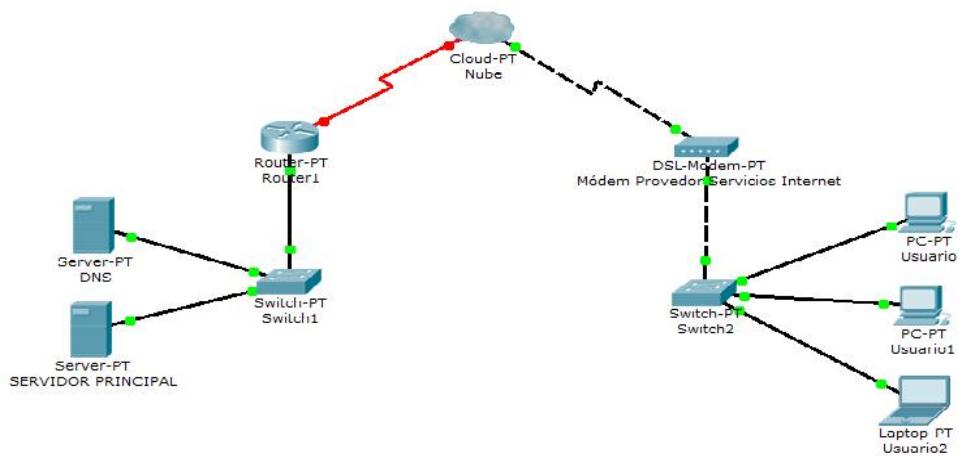


4.1.7 DIAGRAMA DE EMPLAZAMIENTO

4.1.7.1 DIAGRAMA DE DESPLIEGUE

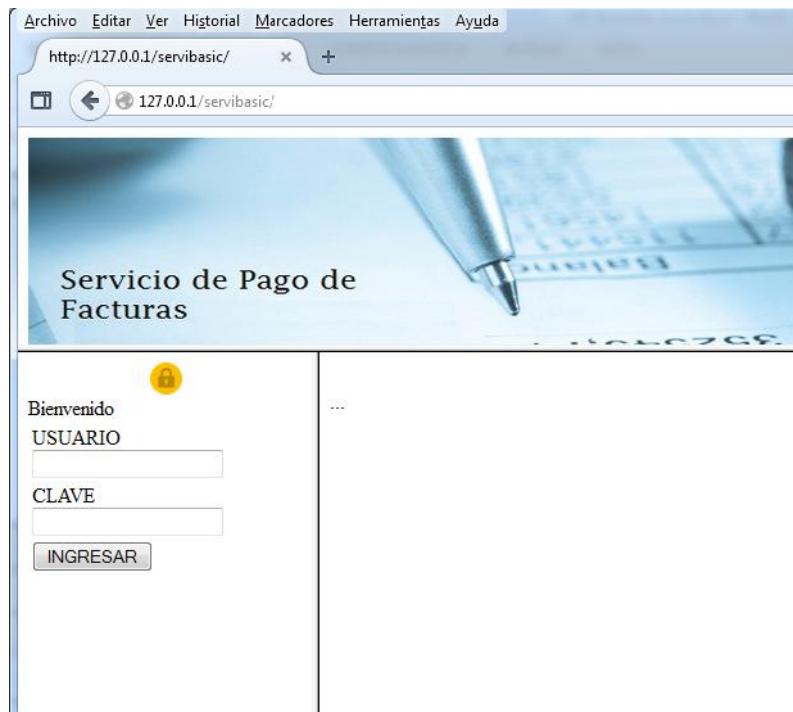


4.1.7.2 DIAGRAMA DE ARQUITECTURA DE RED



4.1.8 DISEÑO Y DESCRIPCIÓN DE PANTALLAS DEL SISTEMA

4.1.8.1 PANTALLA DE ACCESO AL SISTEMA



Nombre: Pantalla principal de Ingreso al Sistema

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario de ingreso al Sistema

Campos:

- Usuario: Nombre del usuario
- Clave: Clave del usuario

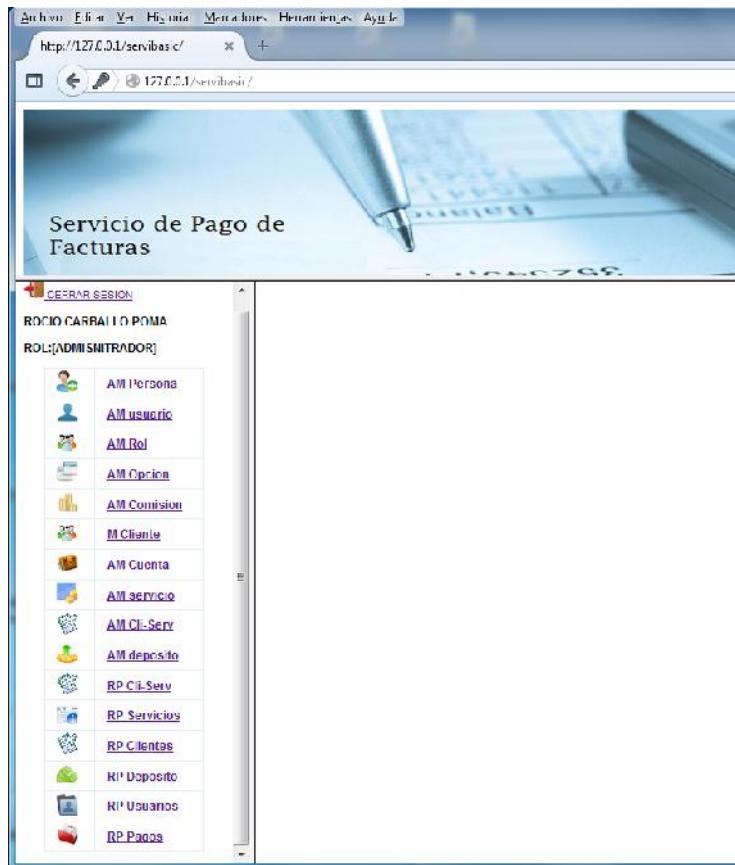
Botones:

- Ingresar: Permite comprobar si el usuario existe y validar su contraseña

Descripción: Cuando la validación del usuario es correcta el sistema busca el rol asignado y sus respectivas opciones del menú.

Comentario: Si el usuario no existe muestra una pantalla de error.

4.1.8.2 PANTALLA LISTA DE OPCIONES DEL MENÚ PARA EL USUARIO ADMINISTRADOR



Nombre Entrada: Pantalla principal con el listado de opciones de menú para el usuario administrador

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario principal con lista de opciones de menú

Enlaces:

Cerrar Sesión: Borra las variables de sesión necesarias para interactuar con el sistema y cierra las ventanas del sistema.

Gestión de usuarios:

- AM persona: Permite Adicionar/ Modificar datos de personas
- AM Usuario: Permite
- AM Rol
- AM Opción

Gestión de Clientes:

- M Cliente: permite modificar los datos del cliente

Gestión de Cuentas:

- AM Cuenta

Gestión de Comisiones:

- AM Comisión

Gestión de Servicios:

- AM Servicio
- AM Cliente_servicio

Gestión de Transacciones:

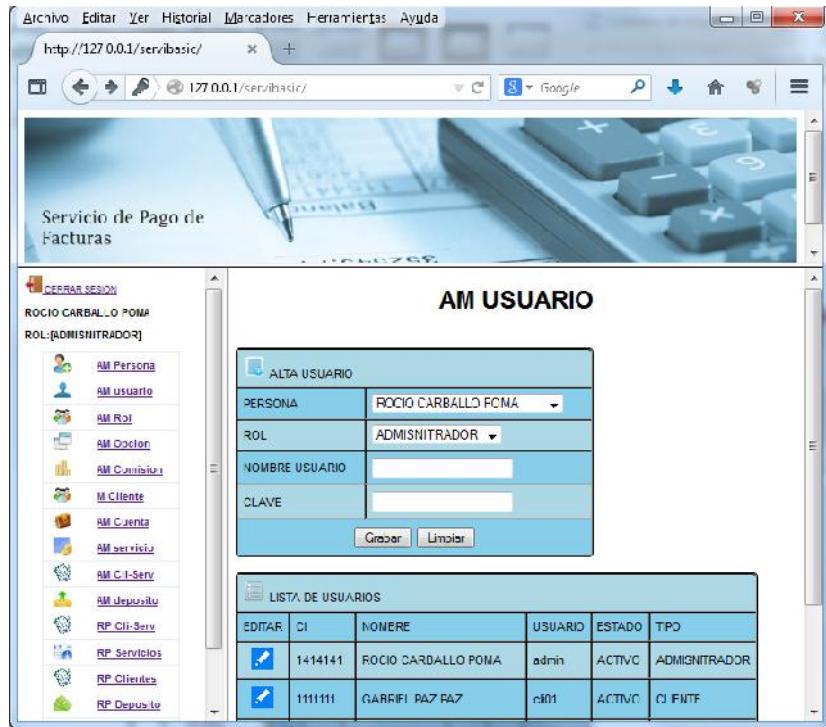
- AM Deposito:

Reportes:

- RPT_Servicios:
- RPT_Cliente_servicios
- RPT_Clientes
- PRT_Depositos
- RPT_Usuarios
- RPT_Pagos

Descripción:**Comentario:**

4.1.8.3 PANTALLA AM (ALTA/MODIFICACIÓN) USUARIO



Nombre: Pantalla AM Usuario

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario de alta para usuarios

Campos:

- PERSONA: Nombre de la persona registrada previamente en el sistema
- ROL: Rol definido en sistema (Administrativo, Liquidador o Cliente)
- NOMBRE USUARIO: Nombre del usuario para ingresar al sistema
- CLAVE: Clave para acceso al sistema

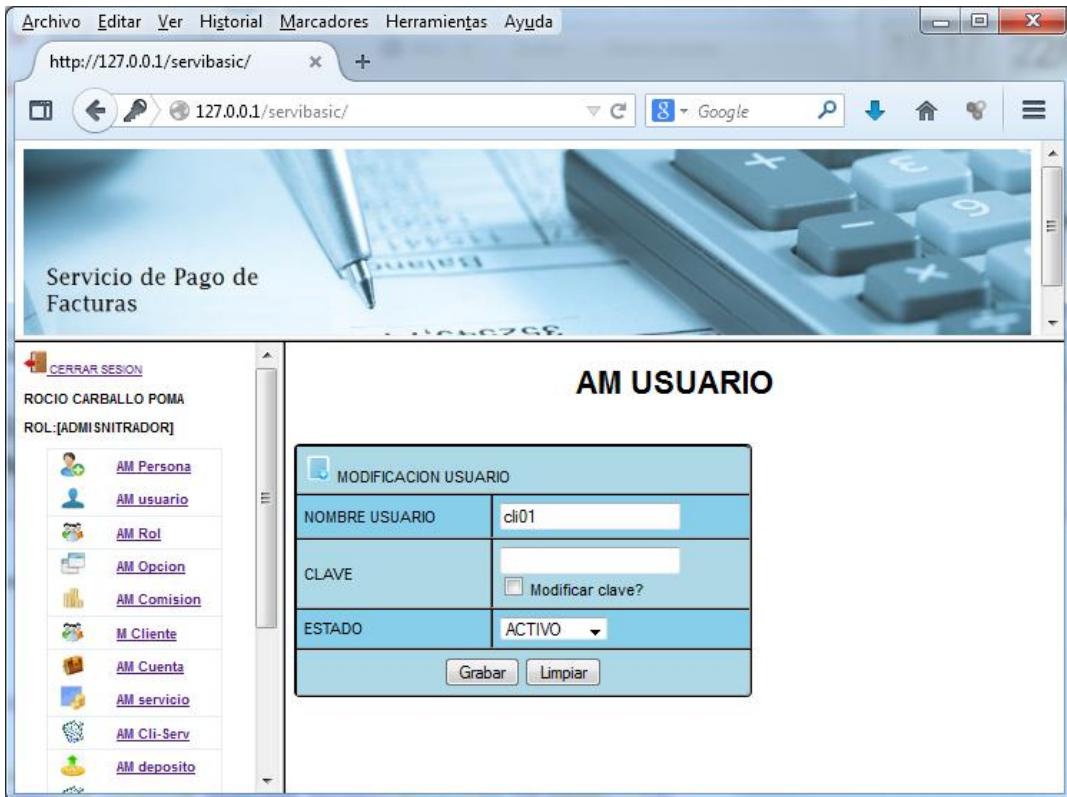
Botones:

- Grabar: Permite grabar los datos del formulario en la base de datos.
- Limpiar: Limpia los datos del formulario
- Modificar Usuario

Descripción: Alta, modificación del usuario del sistema.

Comentario: Previamente se debe registrar a la persona con todos sus datos para luego registrar al usuario en el sistema.

4.1.8.4 PANTALLA MODIFICAR USUARIO



Nombre: Pantalla para modificar usuario

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario de modificación de usuario

Campos:

- NOMBRE USUARIO: Nombre del usuario
- CLAVE: Clave
- ESTADO: Activo/Inactivo
- MODIFICAR CLAVE?: marcando este cuadro podemos modificar la clave.

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario

Descripción: Esta pantalla permite modificar los datos del usuario.

Comentario: Para cambiar de rol al usuario se debe colocar en estado inactivo y crear otro usuario para guardar un histórico de todas sus transacciones realizadas.

4.1.8.5 PANTALLA AM OPCIÓN

EDITAR	ROL	NOMBRE OPCION	ORDEN	ESTADO	RUTA	MARFN	IMAGEN
<input type="checkbox"/>	ADVISOR	AM persona	1	ACTIVO	/am_persona/	/Imagenes/persona.jpg	
<input type="checkbox"/>	ADVISOR	AM usuario	2	ACTIVO	/am_usuario/	/Imagenes/usuario.jpg	
<input type="checkbox"/>	ADVISOR	AM Rol	3	ACTIVO	/am_rol/	/Imagenes/rol.jpg	
<input type="checkbox"/>	ADVISOR	AM opcion	4	ACTIVO	/am_opcion/	/Imagenes/opcion.jpg	

Nombre: Pantalla Alta / modificación de opciones para menú

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario de alta/ modificación de opciones del menú

Campos:

- ROL: Rol del usuario (Administrativo, Cliente, Liquidador)
- NOMBRE DE LA OPCIÓN: Nombre para el enlace de la opción.
- RUTA: Ruta donde se ejecuta el módulo o subprograma
- IMAGEN: Imagen del ícono para acceder a la opción del menú
- ORDEN: Orden en el que aparece la opción en el menú del usuario

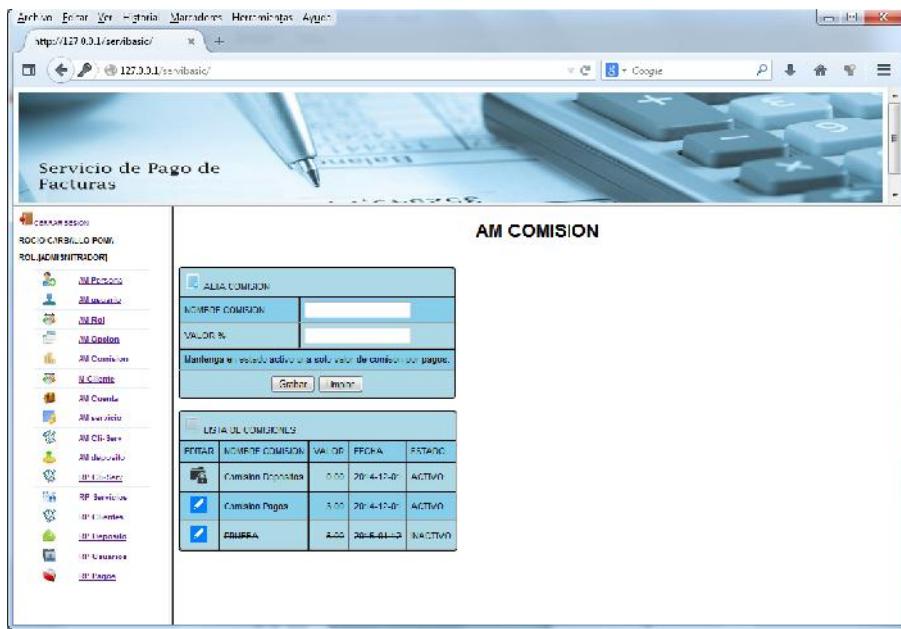
Botones/enlaces:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción:

Comentario:

4.1.8.6 PANTALLA AM (ALTA/MODIFICACIÓN) COMISIÓN



Nombre: Pantalla Alta/modificación de comisión

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario de alta de comisiones que se cobran por los pagos realizados.

Campos:

- NOMBRE COMISIÓN: Nombre descriptivo de la comisión
- VALOR EN %: Valor numérico que indica el porcentaje que se cobra al cliente por cada transacción de pago por concepto de servicios.

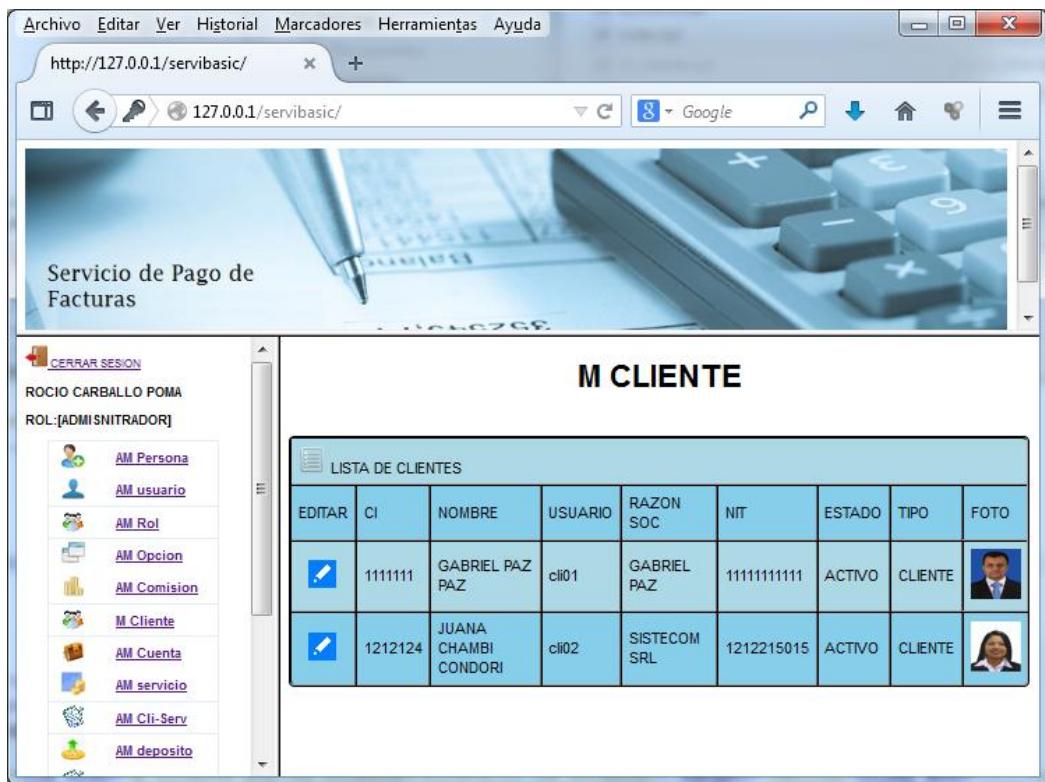
Botones/enlaces:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Registro Protegido por el sistema valor por defecto.
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción: Esta pantalla permite dar de alta las comisiones de cobro por los servicios que deben cancelar los clientes.

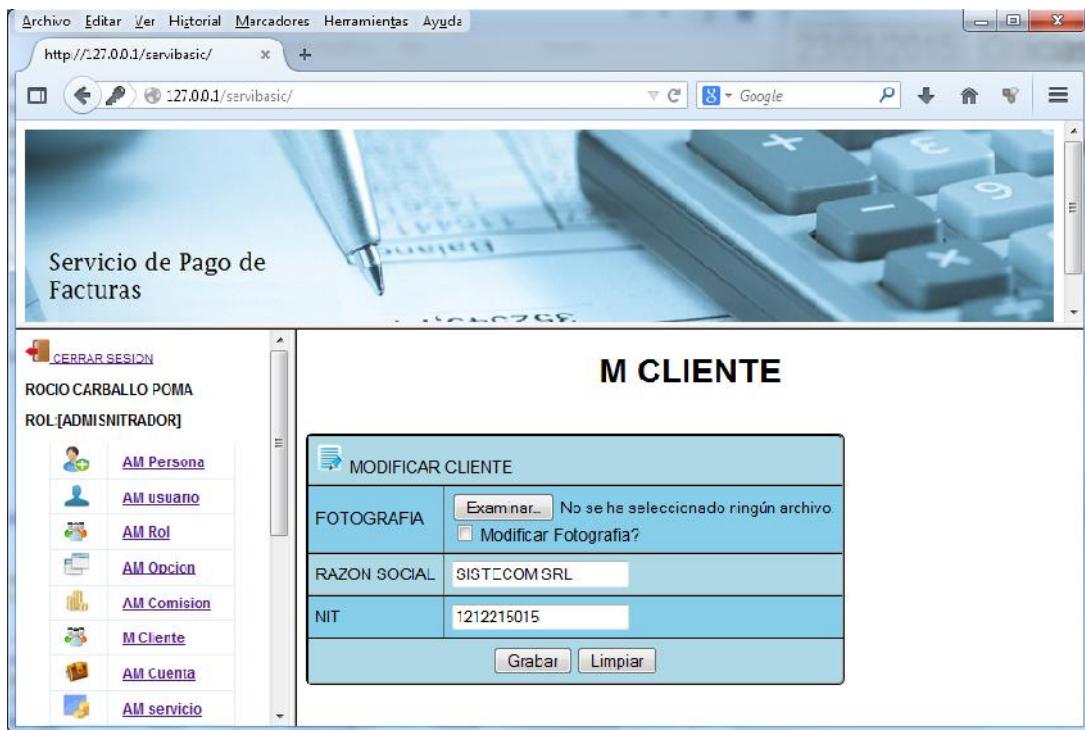
Comentario: Debe dejar activa una sola comisión por los pagos.

4.1.8.7 PANTALLA MODIFICAR DATOS CLIENTE (LISTA)



Nombre: Pantalla modificar datos del Cliente (Lista)	
Tipo de Forma:	<input checked="" type="checkbox"/> Entrada <input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte <input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno <input checked="" type="checkbox"/> Externo
Contenido: <ul style="list-style-type: none"> Lista de los clientes Botones/enlaces: <ul style="list-style-type: none"> Modificar registro 	
Descripción: Esta pantalla presenta una lista con todos los clientes por orden de registro y permite acceder a la pantalla de edición mediante el botón respectivo.	
Comentario:	

4.1.8.8 PANTALLA MODIFICAR CLIENTE



Nombre: Pantalla Modificar datos del Cliente

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario para modificar datos del cliente

Campos:

- FOTOGRAFÍA: Archivo digital que contiene la fotografía del cliente.
- MODIFICAR FOTOGRAFÍA: Si este cuadro está marcado permite realizar la modificación de la fotografía.
- RAZÓN SOCIAL: Nombre o razón social del cliente
- NIT: Número de identificación tributaria del cliente

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario

Descripción: Esta pantalla permite la modificación de los datos del cliente.

Comentario:

4.1.8.9 PANTALLA ALTA/MODIFICACIÓN CUENTA

EDITAR	N CUENTA	NOMBRE CLIENTE	RAZON SOCIAL	NIT	NOMBRE CUENTA	SALDO	ESTADO
	3	GABRIEL PAZ PAZ	GABRIEL PAZ	1111111111	PAGO COLEGIO	0.00	ACTIVO
	1	GABRIEL PAZ PAZ	GABRIEL PAZ	1111111111	PAGO SERVICIOS	897.00	ACTIVO

Nombre: Pantalla Alta modificación de cuenta

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario para alta/ modificación de cuentas de clientes

Campos:

- **CLIENTE:** Cliente registrado en el sistema.
- **NOMBRE CUENTA:** Nombre de la cuenta que usara el cliente para descargos de pagos a los servicios asignados.

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción:

Comentario:

4.1.8.10 PANTALLA ALTA MODIFICACIÓN SERVICIO

The screenshot shows a web browser window with the URL <http://127.0.0.1/servibasic/>. The title bar says "AM SERVICIO". The left sidebar has a "CERRAR SESIÓN" button and a "ROL [ADMINISTRADOR]" section. Below that is a list of icons with corresponding module names: AM Persona, AM Usuario, AM Rol, AM Oficina, AM Comisión, M Cliente, AM Cuenta, AM Servicio, AM Cl-Serv, AM deposito, RP Cli-Serv, RP Servicios, and RP Clientes. The main content area has two forms. The top form is titled "ALTA SERVICIO" and contains fields for "RAZON SOCIAL" and "NIT", with "Grabar" and "Limpiar" buttons. The bottom form is titled "LISTA DE SERVICIOS" and shows a table with three rows of data:

EDITAR	RAZON SERVICIO	NIT	ESTADO
	SERVICIOS ELECTRICOS TARIJA	1021717022	ACTIVO
	COSAALT LTDA TJA	1024715025	ACTIVO
	ENTEL SA	10245466412	ACTIVO

Nombre: Pantalla alta /modificación de Servicio

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario para alta/ modificación de Servicio

- RAZÓN SOCIAL: Razón Social o nombre de la empresa proveedora del servicio
- NIT: Número de identificación tributaria de la empresa

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción:

Comentario:

4.1.8.11 PANTALLA ALTA/MODIFICACIÓN SERVICIO CLIENTE

Nombre: Pantalla Alta modificación de Servicio Cliente

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario para alta/ modificación de Servicio

- CLIENTE: Cliente registrado en la base de datos.
- SERVICIO registrado en la base de datos

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción:

Comentario:

4.1.8.12 PANTALLA ALTA MODIFICACION DEPÓSITO

The screenshot shows a web browser window with the URL <http://127.0.0.1/servbasic/>. The title bar says "Servicio de Pago de Facturas". The main content area has a header "AM DEPOSITO". Below it is a form titled "ALTA DEPOSITO" with fields: CUENTA - CIUD-NIT (with value "GAERIEL PAZ PAZ [GAERIEL PAZ] [PAGO SERVICIOS] 897.00 Bs."), MONTO (empty), NOMBRE DEP. (empty), and REFERENCIA (empty). At the bottom of the form are "Grabar" and "Limpiar" buttons. Below the form is a table titled "LISTA DE DEPOSITOS" with columns: EDITAR, IMPRIMIR, NOVILLI CLIENTE, RAZON SOCIAL, NIT, CUENTA, SALDO, FECHA, MONTO TRANS, and ESTADO. One row is visible with values: IMPRIMIR (pencil icon), NOVILLI PAZ PAZ, GAURILL PAZ, 1111111111, HAGO SERVICIOS, 897.00, 2015-01-12, 800.00, and ACTIVO.

Nombre: Pantalla Alta / Modificación Deposito

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido: Formulario para alta/ modificación de Servicio

- CUENTA CLIENTE: Cuenta del cliente
- MONTO: Monto a depositar
- NOMBRE DEP.: Nombre de la persona que deposita no necesariamente debe existir en la base de datos
- REFERENCIA: Datos de referencia de la persona que deposita, cédula de identidad, teléfono u otro dato complementario.

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)
- Imprimir Registro.

Descripción:

Comentario:

4.1.8.13 REPORTE REGISTRO COMPROBANTE DE DEPÓSITO.



Nombre:	Reporte impreso del deposito		
Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida	
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla	
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo	
Contenido:			
<ul style="list-style-type: none"> • CLIENTE: Nombre del cliente • RAZÓN SOCIAL: Razón Social del cliente • NIT CLIENTE: Número de identificación tributaria del cliente • NOMBRE DE LA CUENTA: Nombre de la cuenta del cliente • SALDO: Saldo actual de la cuenta del cliente • FECHA DE DEPÓSITO: Fecha de la transacción depósito • MONTO: Monto del deposito • NOMBRE DEL DEPOSITANTE: datos para control • REFERENCIA: Datos para control • ESTADO DEL DEPÓSITO: Estado del deposito 			
Botones/enlaces:			
<ul style="list-style-type: none"> ▪ Imprimir esta página: Envía el reporte al dispositivo de impresión ▪ Cerrar esta página: Cierra la ventana emergente. 			
Descripción:			
Comentario: Este mismo reporte se muestra para el cliente.			

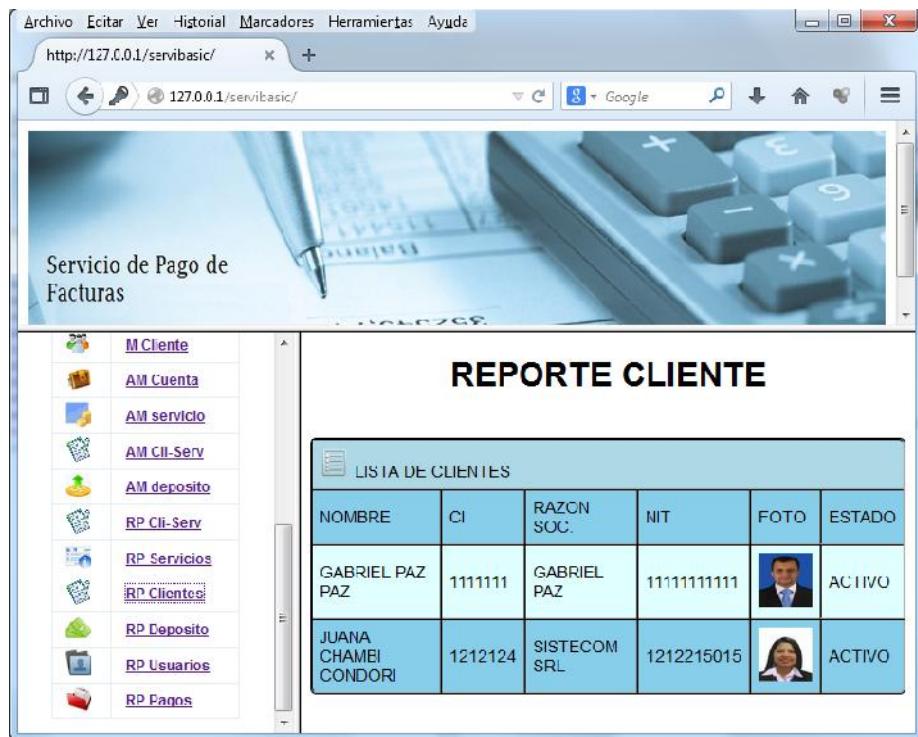
4.1.8.14 REPORTE DE SERVICIOS

The screenshot shows a web application window with a title bar containing 'Archivo', 'Editar', 'Ver', 'Historial', 'Marcadores', 'Herramientas', and 'Ayuda'. The address bar shows 'http://127.0.0.1/servibasic/'. The main content area has a header 'Servicio de Pago de Facturas' with a background image of a keyboard and pen. To the left is a sidebar with icons and links: M Cliente, AM Cuenta, AM servicio, AM Cli-Serv, AM deposito, RP Cli-Serv, RP Servicios, RP Clientes, RP Deposito, RP Usuarios, and RP Pagos. The main content area is titled 'REPORTE SERVICIOS' and contains a table titled 'LISTA DE SERVICIOS' with the following data:

Nº	RAZON SERVICIO	NIT	ESTADO
1	SERVICIOS ELECTRICOS TARIJA	1024717022	ACTIVO
2	COSAALT LTDA TJA	1024715025	ACTIVO
3	ENTEL SA	10245465412	ACTIVO

Nombre: Reporte Servicios	
Tipo de Forma:	<input type="checkbox"/> Entrada <input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte <input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno <input type="checkbox"/> Externo
Contenido:	
<ul style="list-style-type: none"> • Nº: Número de registro • RAZÓN SOCIAL: Razón social de la empresa proveedora del servicio • NIT: Número de identificación tributaria de la proveedora de servicios • ESTADO: Estado del servicio (Activo/Inactivo) 	
Descripción: Lista de todos los servicios activos o inactivos presentes en el sistema.	
Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.	

4.1.8.15 REPORTE DE CLIENTES



Nombre: Reporte clientes	
Tipo de Forma:	<input type="checkbox"/> Entrada <input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte <input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno <input type="checkbox"/> Externo
Contenido:	
<ul style="list-style-type: none"> • NOMBRE: Nombre completo del cliente • CI: Número de cedula de identidad • NIT: Número de identificación tributaria • FOTO: Fotografía en formato mapa de bits del cliente • ESTADO: Estado del cliente activo o inactivo 	
Descripción: Lista de todos los clientes.	
Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.	

4.1.8.16 REPORTE DEPÓSITOS

REPORTE DEPOSITO

LISTA DE DEPOSITOS							
Nº	FECHA	RUT/IMP.	NOMBRE CLIENTE	RAZON SOCIAL	NIT	CUENTA	MONTO
1	2015-01-12		GABRIEL PAZ PAZ	GABRIEL PAZ	111111111111	PAGO SERVICIOS	€00 00
2	2015-01-12		GABRIEL PAZ PAZ	GABRIEL PAZ	111111111111	PAGO SERVICIOS	€00 00
TOTAL Bs.							1000

Nombre: Reporte Depósitos

Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- CLIENTE: Filtro por cliente
- FECHA: Filtro por fecha de inicio - fecha final
- N°: Número de registro
- FECHA: Fecha de la transacción de depósito.
- NOMBRE CLIENTE: Nombre completo del cliente
- RAZON SOCIAL: Razón social
- NIT: Número de identificación tributaria del cliente.
- CUENTA: Cuenta del cliente
- MONTO: Monto de la transacción del depósito.
- ESTADO: Estado de la transacción del depósito (Activa/Anulada)

Botones/enlaces:

- Imprimir el registro

Descripción: Lista de todos los depósitos por criterios de cliente o fecha

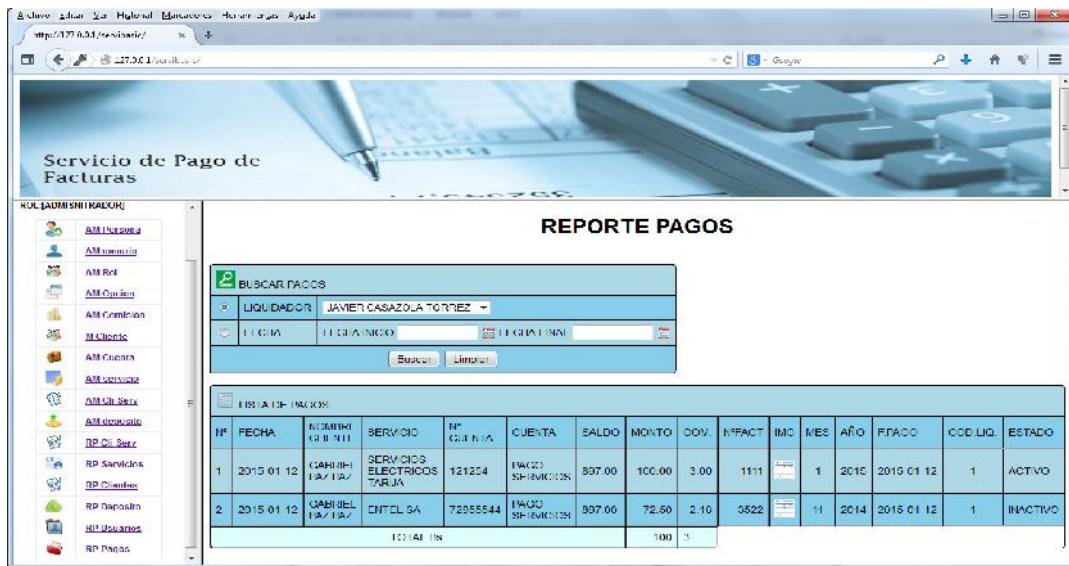
Comentario:

4.1.8.17 REPORTE USUARIOS

LISTA DE USUARIOS				
NOMBRE	CI	NOMB.USUARIO	ROL	ESTADO
ROCIO CARBALLO POMA	1414141	admin	ADMISNITRADOR	ACTIVO
GABRIEL PAZ PAZ	1111111	cli01	CLIENTE	ACTIVO
JAVIER CASAZOLA TORREZ	14141478	liq01	LIQUIDADOR	ACTIVO
JUANA CHAMBI CONDORI	1212124	cli02	CLIENTE	ACTIVO
MARCELO HUANCA PORTAL	52525252	liq02	LIQUIDADOR	ACTIVO

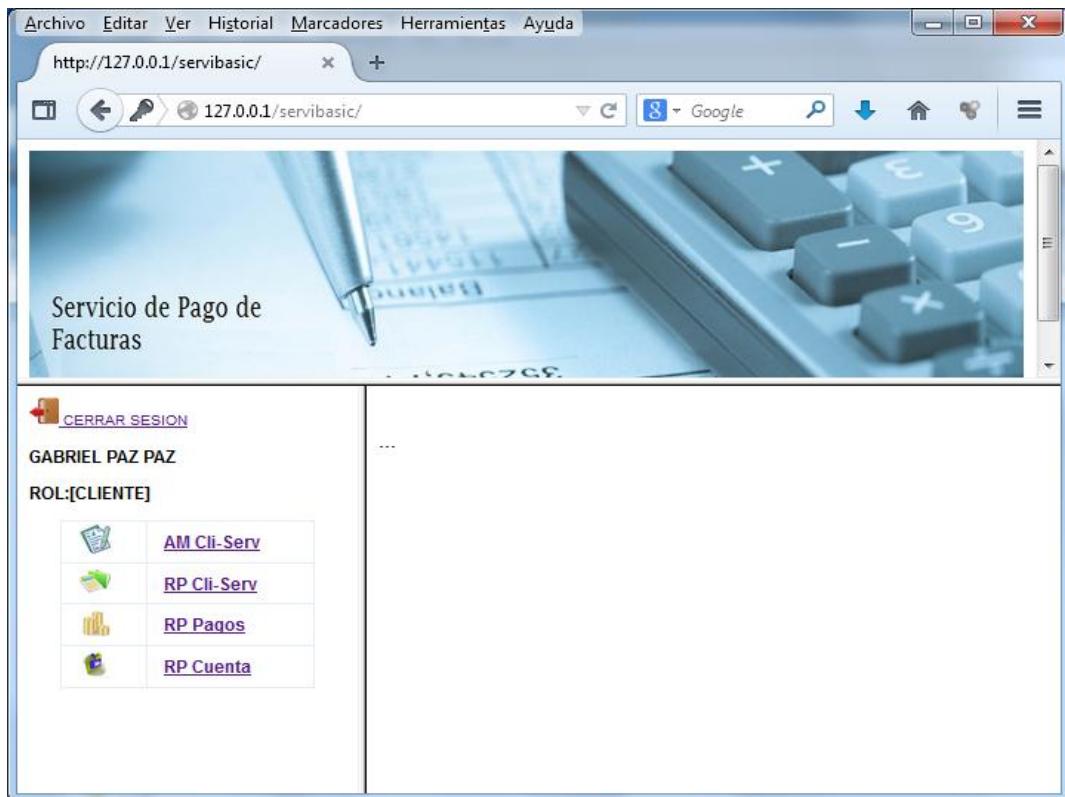
Nombre: Reporte usuarios		
Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo
Contenido:		
<ul style="list-style-type: none"> • NOMBRE: Nombre completo del usuario • CI: Cedula de identidad de la persona • NOMB. USUARIO: Nombre del usuario para ingresar al sistema • ROL: Rol del usuario dentro del sistema • ESTADO: Estado del usuario 		
Descripción: Lista de todos los usuarios del sistema		
Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.		

4.1.8.18 REPORTE DE PAGOS



Nombre: Reporte usuarios	<input type="checkbox"/> Entrada		<input checked="" type="checkbox"/> Salida		
Tipo de Forma:	<input checked="" type="checkbox"/> Reporte		<input type="checkbox"/> Pantalla		
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Interno		<input type="checkbox"/> Externo		
Contenido: Lista de todos los usuarios del sistema					
<ul style="list-style-type: none"> • N°: Número de registro • FECHA: Fecha de la transacción. • NOMBRE CLIENTE: Nombre completo del cliente • SERVICIO: Nombre del servicio que se paga. • N° CUENTA: Número de cuenta del servicio. • CUENTA: Cuenta que se usa para descargo del pago • SALDO: Saldo de la cuenta utilizada para el pago • MONTO: Monto que se paga por el servicio • COM: Monto de la comisión • N.FACT: Número de factura • IMG: Imagen que corresponde a la factura digitalizada • MES/AÑO: mes y año del servicio facturado • F.PAGO: Fecha de pago • COD. LIQ: Código del liquidador • ESTADO: Estado del usuario 					
Descripción: Lista de pagos realizados por el liquidador o fecha inicio – fecha final					
Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.					

4.1.8.19 PANTALLA MENÚ DE OPCIONES PARA EL CLIENTE



Nombre Entrada: Pantalla principal con el listado de opciones de menú para el usuario cliente

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input checked="" type="checkbox"/> Externo

Contenido:

Enlaces:

Cerrar Sesión: Borra las variables de sesión necesarias para interactuar con el sistema y cierra las ventanas del sistema.

Gestión de Servicios:

- AM Cliente_servicio:

Reportes:

- RPT_Servicios:
- RPT_Cliente_servicios
- RPT_Pagos
- RPT_Cuenta

Descripción: Pantalla principal con lista de opciones de menú

Comentario:

4.1.8.20 PANTALLA ALTA MODIFICACIÓN CLIENTE-SERVICIO



Servicio de Pago de Facturas

 CERRAR SESIÓN GABRIEL PAZ PAZ ROL [CLIENTE] <ul style="list-style-type: none"> AM Cli-Serv RP Cli-Serv RP Pagos RP Cuenta 	<h3 style="text-align: center;">AM SERVICIO CLIENTE</h3> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="padding: 5px;">ALTA SERVICIOS CI IFNTF</td> </tr> <tr> <td style="width: 30%;">SERVICIO</td> <td style="width: 70%; background-color: #ADD8E6;">SERVICIOS ELECTRICOS TARIJA ▾</td> </tr> <tr> <td>NUMERO DE CUENTA</td> <td>Este es el numero de cuenta asignado al servicio basico que desea pagar impreso en su factura</td> </tr> <tr> <td colspan="2" style="text-align: center; padding: 5px;"> <input type="button" value="Grabar"/> <input type="button" value="Limpiar"/> </td> </tr> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="7" style="padding: 5px;">LISTA SERVICIOS ASIGNADOS A CLIENTES</td> </tr> <tr> <th style="width: 10%;">EDITAR</th> <th style="width: 15%;">NOMBRE CLIENTE</th> <th style="width: 15%;">RAZON SOCIAL</th> <th style="width: 10%;">NIT</th> <th style="width: 10%;">NUM. CUENTA</th> <th style="width: 15%;">SERVICIO</th> <th style="width: 10%;">ESTADOC</th> </tr> <tr> <td></td> <td>GABRIEL PAZ PAZ</td> <td>GABRIEL PAZ</td> <td>11111111111</td> <td>121251</td> <td>SERVICIOS ELECTRICOS TARIJA</td> <td>ACTIVO</td> </tr> </table>	ALTA SERVICIOS CI IFNTF		SERVICIO	SERVICIOS ELECTRICOS TARIJA ▾	NUMERO DE CUENTA	Este es el numero de cuenta asignado al servicio basico que desea pagar impreso en su factura	<input type="button" value="Grabar"/> <input type="button" value="Limpiar"/>		LISTA SERVICIOS ASIGNADOS A CLIENTES							EDITAR	NOMBRE CLIENTE	RAZON SOCIAL	NIT	NUM. CUENTA	SERVICIO	ESTADOC		GABRIEL PAZ PAZ	GABRIEL PAZ	11111111111	121251	SERVICIOS ELECTRICOS TARIJA	ACTIVO
ALTA SERVICIOS CI IFNTF																														
SERVICIO	SERVICIOS ELECTRICOS TARIJA ▾																													
NUMERO DE CUENTA	Este es el numero de cuenta asignado al servicio basico que desea pagar impreso en su factura																													
<input type="button" value="Grabar"/> <input type="button" value="Limpiar"/>																														
LISTA SERVICIOS ASIGNADOS A CLIENTES																														
EDITAR	NOMBRE CLIENTE	RAZON SOCIAL	NIT	NUM. CUENTA	SERVICIO	ESTADOC																								
	GABRIEL PAZ PAZ	GABRIEL PAZ	11111111111	121251	SERVICIOS ELECTRICOS TARIJA	ACTIVO																								

Nombre: Pantalla Alta / Modificación Cliente Servicio

Tipo de Forma: Entrada Salida

Formato de Entrada/Salida: Reporte Pantalla

Tipos de Reporte: Interno Externo

Contenido:

- SERVICIO: Servicio
- NÚMERO DE CUENTA: Número de cuenta asignada al cliente dentro del sistema del servicio

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro: (El proceso es similar a los anteriores casos)

Descripción: Formulario para alta/ modificación Servicios asignados al cliente

Comentario:

4.1.8.21 REPORTE SERVICIOS DEL CLIENTE

Servicio de Pago de Facturas

DATOS DEL CLIENTE	
CLIENTE	GABRIEL PAZ PAZ
DIRECCION	INGAVI 1651
TELEFONO	106654022
USUARIO	cli01
ESTADO	ACTIVO
RAZON SOCIAL	GABRIFI PAZ
NT	1111111111

DATOS CUENTAS CLIENTE			
Nº	CUENTA	SALDO	ESTADO
1	PAGO SERVICIOS	897.00	ACTIVO
2	PAGO COLEGIO	0.00	ACTIVO

DATOS SERVICIOS CLIENTE			
Nº	SERVICIO	NUM. CTA.	ESTADO
1	SERVICIOS ELECTRICOS TARJ.A	121254	ACTIVO
2	COSAALT LTDA TJA	15151515151	ACTIVO
3	ENTEL SA	70055511	ACTIVO

Nombre: Reporte Cliente-Servicios

Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida
-----------------------	----------------------------------	--

Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla
-----------------------------------	---	-----------------------------------

Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo
--------------------------	---	----------------------------------

Contenido:

- DATOS DEL CLIENTE:
- DATOS DE LAS CUENTAS DEL CLIENTE: Lista de las cuentas del cliente.
- DATOS DE LOS SERVICIOS ASIGNADOS AL CLIENTE: Lista de los servicios asignados al cliente.

Descripción:

Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.

4.1.8.22 PANTALLA PAGOS REALIZADOS

Nombre: Pantalla para listar los pagos realizados

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- FECHA DE INICIO: Fecha de inicio dd/mm/aaaa
- FECHA FINAL: Fecha final dd/mm/aaaa

Botones:

- Grabar: Permite grabar los datos del formulario
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción: Formulario filtrar los datos de los pagos realizados por fecha

Comentario:

4.1.8.23 PANTALLA REPORTE PAGOS REALIZADOS



Nombre: Reporte de pagos realizados

Tipo de Forma: Entrada Salida

Formato de Entrada/Salida: Reporte Pantalla

Tipos de Reporte: Interno Externo

Contenido:

- Nº: Número de orden del registro
- SERVICIO: Servicio que se paga
- CUENTA: Cuenta de la que se debita el pago al servicio
- SALDO: Saldo actual de la cuenta
- FECHA: Fecha de la transacción de pago
- MONTO: Monto que se paga
- COMISIÓN: % que se paga por comisión del pago
- N° FACTURA: Número impreso en la factura
- IMAGEN: imagen digitalizada de la factura
- MES: Mes que se paga el servicio
- AÑO: año que se paga el servicio
- COD.LIQ: Código del liquidador que realizó la transacción
- ESTADO: Estado de la transacción.

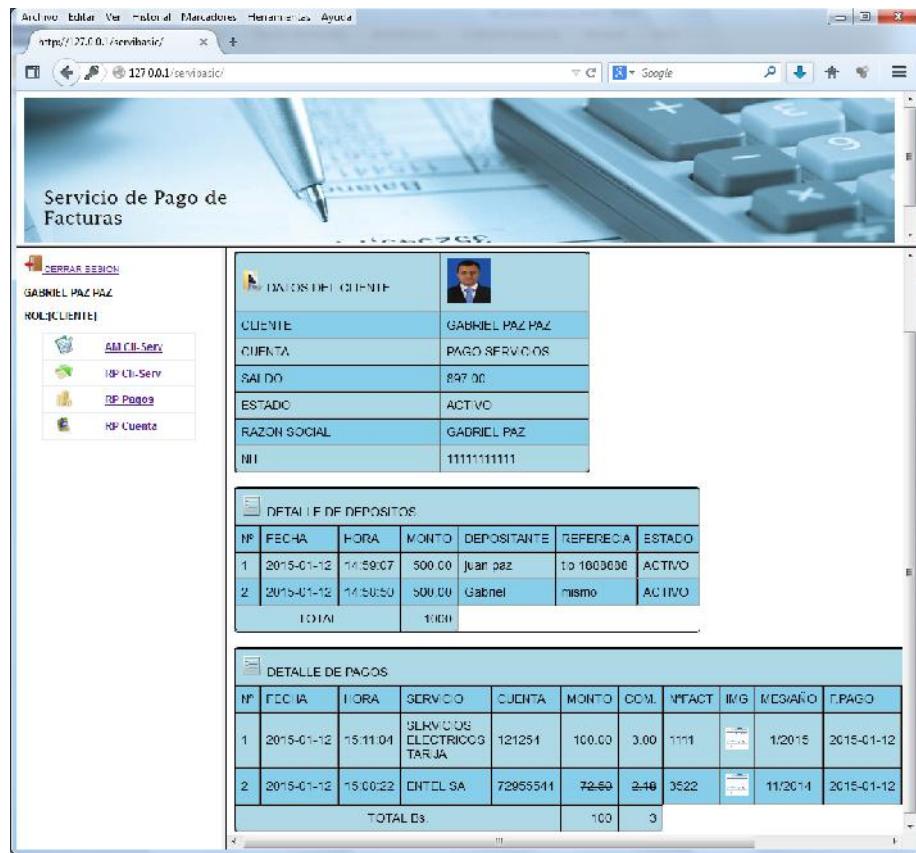
Descripción: Lista de la información del cliente

Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.

4.1.8.24 PANTALLA DETALLE DE CUENTAS

Nombre: Listar detalle de la cuenta
Tipo de Forma: <input checked="" type="checkbox"/> Entrada <input type="checkbox"/> Salida
Formato de Entrada/Salida: <input type="checkbox"/> Reporte <input checked="" type="checkbox"/> Pantalla
Tipos de Reporte: <input type="checkbox"/> Interno <input checked="" type="checkbox"/> Externo
Contenido:
<ul style="list-style-type: none"> • CUENTA • FECHA DE INICIO: Fecha de inicio dd/mm/aaaa • FECHA FINAL: Fecha final dd/mm/aaaa
Botones:
<ul style="list-style-type: none"> ▪ Listar: Permite listar los datos de la cuenta elegida ▪ Limpiar: Limpia los datos del formulario
Descripción: Formulario para filtrar los datos de movimientos de las cuentas del cliente
Comentario:

4.1.8.25 REPORTE DETALLE DE CUENTAS



Nombre: Reporte detalle de cuentas del cliente

Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida
-----------------------	----------------------------------	--

Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla
-----------------------------------	---	-----------------------------------

Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo
--------------------------	---	----------------------------------

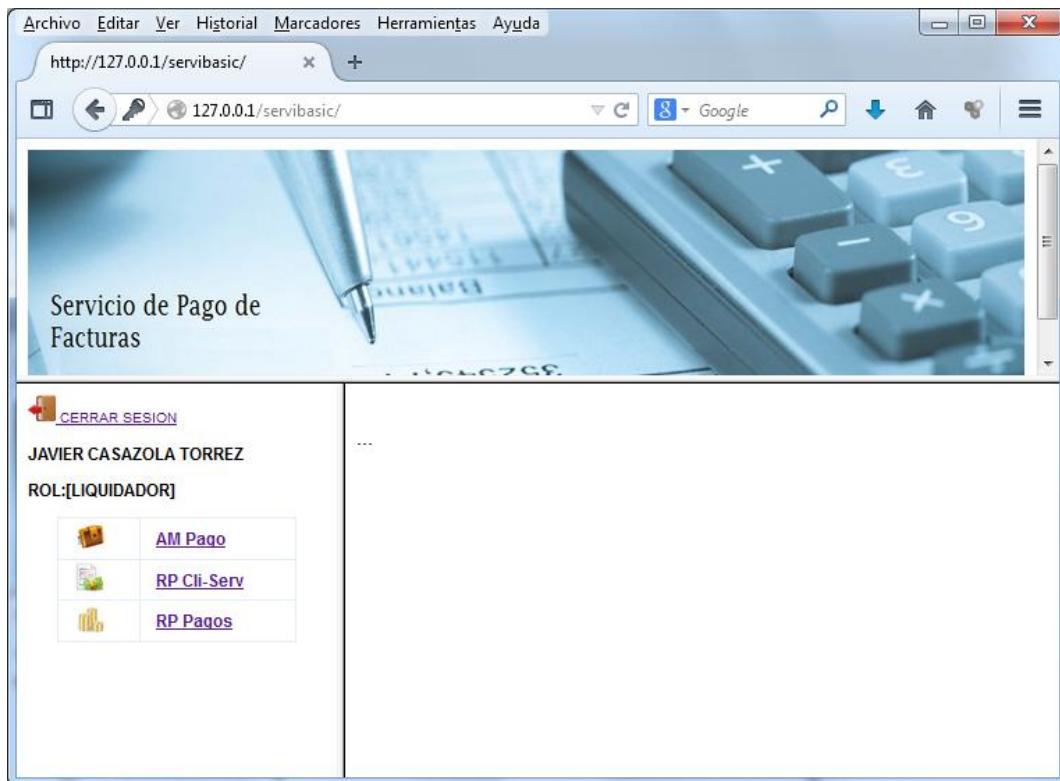
Contenido:

- DATOS DEL CLIENTE:
- DATOS DE LOS DEPOSITOS REALIZADOS: Lista de los depósitos realizados
- DATOS DE LOS PAGOS: Lista de los pagos realizados

Descripción: Lista de la información de las cuentas cliente y las transacciones realizadas

Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.

4.1.8.26 PANTALLA PARA EL USUARIO LIQUIDADOR



Nombre Entrada: Pantalla principal con el listado de opciones de menú para el usuario liquidador

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

Campos:

Enlaces:

- **Cerrar Sesión:** Borra las variables de sesión necesarias para interactuar con el sistema y cierra las ventanas del sistema.

Gestión de transacciones:

- AM PAGO:

Reportes:

- RPT_Cliente_Servicios
- RPT_Pagos

Descripción: Formulario principal con lista de opciones de menú

Comentario:

4.1.8.27 PANTALLA AM PAGO SELECCIÓN DEL CLIENTE Y CUENTA

The screenshot shows a web-based application window titled "AM PAGO". The main area displays a table titled "LISTA DE PAGOS" (List of Payments) with two rows of data. The columns are labeled: LÍNEA, NOMBRE CLIENTE, SERVICIO, N° CUENTA, CULNIA, SALDO, FECHA, MONTO, COM., N° ACT, IMG, MES/AÑO, I.PAGO, and ESTADO. The first row shows a payment from "GARRETAZ PAZ PAZ" for "SERVICIOS ELECTRICOS LAJUA" with account number 121251, balance 397.00, date 2015-01-12, amount 100.00, commission 3.00, and status ACTIVO. The second row shows a payment from "GARRETAZ PAZ PAZ" for "ENTEL SA" with account number 72366011, balance 397.00, date 2015-01-12, amount 72.50, commission 2.10, and status ANULADO. On the left side of the main area, there is a sidebar with a menu containing "AM Pago", "Reporte", and "Historico". The top of the window has a title bar with the URL "http://127.0.0.1/servicebasic/" and a toolbar with various icons.

LÍNEA	NOMBRE CLIENTE	SERVICIO	N° CUENTA	CULNIA	SALDO	FECHA	MONTO	COM.	N° ACT	IMG	MES/AÑO	I.PAGO	ESTADO
<input checked="" type="checkbox"/>	GARRETAZ PAZ PAZ	SERVICIOS ELECTRICOS LAJUA	121251	PAGO SERVICIOS	397.00	2015-01-12	100.00	3.00	1111		1/2015	2015-01-12	ACTIVO
<input type="checkbox"/>	GARRETAZ PAZ PAZ	ENTEL SA	72366011	PAGO SERVICIOS	397.00	2015-01-12	72.50	2.10	3022		11/2014	2015-01-12	ANULADO

Nombre: Pantalla Alta / Modificación Pago selección del cliente y cuenta

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- CUENTA DEL CLIENTE: Servicio

Botones:

- Siguiente: Permite abrir otro formulario donde se completarán los datos del pago
- Limpiar: Limpia los datos del formulario
- Modificar registro (El proceso es similar a los anteriores casos)

Descripción: Formulario para alta/ modificación Pago

Comentario:

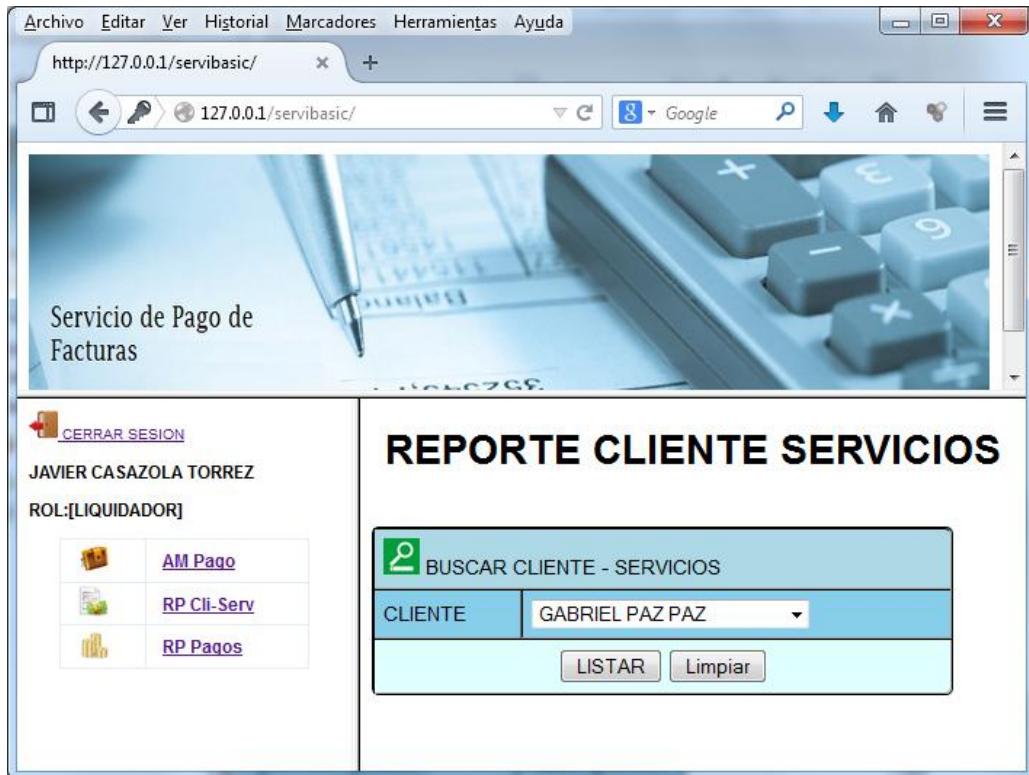
4.1.8.28 PANTALLA AM PAGO

The screenshot shows a software interface titled "AM PAGO". On the left, there's a sidebar with icons for "CERRAR SESIÓN", "JUAN CRISTÓBAL TORREZ", and "ROL [LIQUIDADOR]". Below these are three buttons: "AM Pago", "BPC-iServ", and "BP Pagos". The main area is titled "ALTA PAGO" and contains the following fields:

CUENTA CLIENTE	GABRIEL PAZ PAZ [GABRIEL PAZ] [PAGO SERVICIOS]
SERVICIO	SERVICIOS ELECTRICOS TARIFA 1024717022-121204
MONTO	Baldo: 897.00 Ls.
COMISIÓN	Comisión Pago [3.00 %]
FACTURA	Examinar... No se ha seleccionado ningún archivo.
NÚM.FACTURA	
MES.FACTURA	Jan
AÑO.FACTURA	2015
MONTO CREDITO FISCAL	
FECHA DE PAGO	26/01/2015
FECHA EMISIÓN	26/01/2015
<input type="button" value="Grabar"/> <input type="button" value="Limpiar"/>	

Nombre: Pantalla Alta / Modificación Pago	
Tipo de Forma: <input checked="" type="checkbox"/> Entrada <input type="checkbox"/> Salida	
Formato de Entrada/Salida: <input type="checkbox"/> Reporte <input checked="" type="checkbox"/> Pantalla	
Tipos de Reporte: <input type="checkbox"/> Interno <input checked="" type="checkbox"/> Externo	
Contenido:	
<ul style="list-style-type: none"> CUENTA DEL CLIENTE: Cuenta del cliente selecciona con anterioridad. SERVICIO: Seleccionar el servicio y la cuenta asociada al cliente MONTO: Monto de la transacción del pago. COMISIÓN: Porcentaje de comisión que se cobra del pago realizado por defecto aparece el habilitado por el administrador. FACTURA: Imagen de la factura. NÚM.FACTURA: Número de la factura MES FACTURA: Mes de la factura AÑO FACTURA: Año de la factura MONTO CREDITO FISCAL: Monto válido para crédito fiscal FECHA DE PAGO: Fecha de pago de la factura FECHA DE EMISIÓN: Fecha de emisión de la factura 	
Botones:	
<ul style="list-style-type: none"> Examinar: Permite elegir la imagen digitalizada de la factura. Grabar: Permite grabar los datos del formulario Limpiar: Limpia los datos del formulario Modificar registro (El proceso es similar a los anteriores casos) 	
Descripción: Formulario para alta del pago	
Comentario:	

4.1.8.29 PANTALLA REPORTE CLIENTE-SERVICIO ELEGIR CLIENTE



Nombre: Listar detalle de los servicios asignado al cliente

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- CLIENTE: Cuadro para seleccionar el nombre del cliente

Botones:

- Listar: Lista los datos de servicios del cliente seleccionado
- Limpiar: Limpia los datos del formulario

Descripción: Formulario filtrar los datos de movimientos de las cuentas del cliente

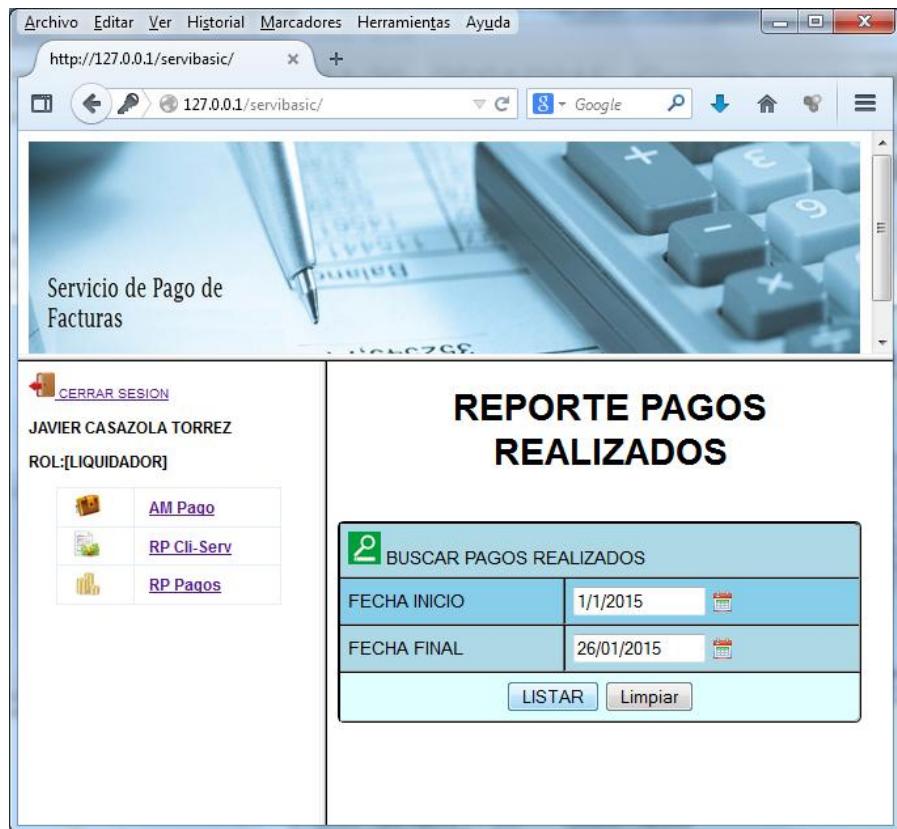
Comentario:

4.1.8.30 PANTALLA REPORTE CLIENTE-SERVICIO DETALLE

 CERRAR SESIÓN JAVIER CASAZOLA TORREZ ROL:[LIQUIDADOR] <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; padding: 2px;"></td> <td style="width: 80%; padding: 2px; text-align: center;">AM Pago</td> </tr> <tr> <td style="width: 20%; padding: 2px;"></td> <td style="width: 80%; padding: 2px; text-align: center;">RP Cli-Serv</td> </tr> <tr> <td style="width: 20%; padding: 2px;"></td> <td style="width: 80%; padding: 2px; text-align: center;">RP Pagos</td> </tr> </table>		AM Pago		RP Cli-Serv		RP Pagos	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="background-color: #ADD8E6;">DATOS DEL CLIENTE</th> </tr> </thead> <tbody> <tr> <td style="width: 5%;"></td> <td style="width: 95%;">GABRIEL PAZ PAZ</td> </tr> <tr> <td>CLIENTE</td> <td>GABRIEL PAZ PAZ</td> </tr> <tr> <td>DIRECCION</td> <td>INGAVI 1651</td> </tr> <tr> <td>TELEFONO</td> <td>16654022</td> </tr> <tr> <td>USUARIO</td> <td>cli01</td> </tr> <tr> <td>ESTADO</td> <td>ACTIVO</td> </tr> <tr> <td>RAZON SOCIAL</td> <td>GABRIEL PAZ</td> </tr> <tr> <td>NIT</td> <td>111111111111</td> </tr> </tbody> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="5" style="background-color: #ADD8E6;">DATOS CUENTAS CLIENTE</th> </tr> <tr> <th style="width: 10%;">Nº</th> <th style="width: 15%;">DETALLE</th> <th style="width: 25%;">CUENTA</th> <th style="width: 15%;">SALDO</th> <th style="width: 25%;">ESTADO</th> </tr> </thead> <tbody> <tr> <td>1</td> <td></td> <td>PAGO SERVICIOS</td> <td>897.00</td> <td>ACTIVO</td> </tr> <tr> <td>2</td> <td></td> <td>PAGO COLEGIO</td> <td>0.00</td> <td>ACTIVO</td> </tr> </tbody> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="4" style="background-color: #ADD8E6;">DATOS SERVICIOS CLIENTE</th> </tr> <tr> <th style="width: 10%;">Nº</th> <th style="width: 40%;">SERVICIO</th> <th style="width: 20%;">NUM. CTA.</th> <th style="width: 30%;">ESTADO</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>SERVICIOS ELECTRICOS TARIJA</td> <td>121254</td> <td>ACTIVO</td> </tr> <tr> <td>2</td> <td>COSAALT LTDA TJA</td> <td>15151515151</td> <td>ACTIVO</td> </tr> <tr> <td>3</td> <td>ENTEL SA</td> <td>72955544</td> <td>ACTIVO</td> </tr> </tbody> </table>	DATOS DEL CLIENTE			GABRIEL PAZ PAZ	CLIENTE	GABRIEL PAZ PAZ	DIRECCION	INGAVI 1651	TELEFONO	16654022	USUARIO	cli01	ESTADO	ACTIVO	RAZON SOCIAL	GABRIEL PAZ	NIT	111111111111	DATOS CUENTAS CLIENTE					Nº	DETALLE	CUENTA	SALDO	ESTADO	1		PAGO SERVICIOS	897.00	ACTIVO	2		PAGO COLEGIO	0.00	ACTIVO	DATOS SERVICIOS CLIENTE				Nº	SERVICIO	NUM. CTA.	ESTADO	1	SERVICIOS ELECTRICOS TARIJA	121254	ACTIVO	2	COSAALT LTDA TJA	15151515151	ACTIVO	3	ENTEL SA	72955544	ACTIVO
	AM Pago																																																																
	RP Cli-Serv																																																																
	RP Pagos																																																																
DATOS DEL CLIENTE																																																																	
	GABRIEL PAZ PAZ																																																																
CLIENTE	GABRIEL PAZ PAZ																																																																
DIRECCION	INGAVI 1651																																																																
TELEFONO	16654022																																																																
USUARIO	cli01																																																																
ESTADO	ACTIVO																																																																
RAZON SOCIAL	GABRIEL PAZ																																																																
NIT	111111111111																																																																
DATOS CUENTAS CLIENTE																																																																	
Nº	DETALLE	CUENTA	SALDO	ESTADO																																																													
1		PAGO SERVICIOS	897.00	ACTIVO																																																													
2		PAGO COLEGIO	0.00	ACTIVO																																																													
DATOS SERVICIOS CLIENTE																																																																	
Nº	SERVICIO	NUM. CTA.	ESTADO																																																														
1	SERVICIOS ELECTRICOS TARIJA	121254	ACTIVO																																																														
2	COSAALT LTDA TJA	15151515151	ACTIVO																																																														
3	ENTEL SA	72955544	ACTIVO																																																														

Nombre: Reporte detalle de los servicios del cliente	
Tipo de Forma:	<input type="checkbox"/> Entrada <input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte <input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno <input type="checkbox"/> Externo
Contenido:	
<ul style="list-style-type: none"> • DATOS DEL CLIENTE: • DATOS DE LAS CUENTAS DEL CLIENTE: Lista de las cuentas del cliente • DATOS DE LOS SERVICIOS: Lista de los servicios asignados al cliente 	
Descripción: Lista de la información de los servicios y cuentas del cliente	
Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.	

4.1.8.31 PANTALLA REPORTE PAGOS ELEGIR FECHAS



Nombre: Listar detalle de los pagos por fecha de transacción

Tipo de Forma:	<input checked="" type="checkbox"/> Entrada	<input type="checkbox"/> Salida
Formato de Entrada/Salida:	<input type="checkbox"/> Reporte	<input checked="" type="checkbox"/> Pantalla
Tipos de Reporte:	<input type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- FECHA DE INICIO: Fecha de inicio dd/mm/aaaa
- FECHA FINAL: Fecha final dd/mm/aaaa

Botones:

- Listar: Permite listar los datos de los pagos realizados por el usuario liquidador con el criterio elegido.
- Limpiar: Limpia los datos del formulario

Descripción: Formulario filtrar los datos de los pagos por fechas

Comentario:

4.1.8.32 PANTALLA REPORTE PAGOS DETALLE



Nombre: Reporte detalle de los pagos realizados

Tipo de Forma:	<input type="checkbox"/> Entrada	<input checked="" type="checkbox"/> Salida
Formato de Entrada/Salida:	<input checked="" type="checkbox"/> Reporte	<input type="checkbox"/> Pantalla
Tipos de Reporte:	<input checked="" type="checkbox"/> Interno	<input type="checkbox"/> Externo

Contenido:

- N°: Número de registro
- NOMBRE CLIENTE: Nombre completo del cliente
- SERVICIO: Nombre del servicio que se paga.
- N° CUENTA: Número de cuenta del servicio.
- CUENTA: Cuenta que se usa para descargo del pago
- SALDO: Saldo de la cuenta utilizada para el pago
- FECHA: Fecha de la transacción
- MONTO: Monto que se paga por el servicio
- COM: Monto de la comisión
- N.FACT: Número de factura
- IMG: Imagen que corresponde a la factura digitalizada
- MES/AÑO: mes y año del servicio facturado
- F.PAGO: Fecha de pago
- ESTADO: Estado del pago

Descripción: Lista de los pagos realizados por el usuario liquidador

Comentario: Se puede usar la opción “imprimir” nativa del navegador para imprimir este reporte.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Se concluyó satisfactoriamente el presente proyecto dando respuesta a los objetivos específicos planteados:

- Se utilizó los diagramas de Casos de Usos para modelar el comportamiento del sistema y requerimientos de los usuarios.
- Se aplicó la metodología RUP para el desarrollo del proyecto ya que la naturaleza de dicha metodología se adecúa perfectamente a las diferentes etapas que conllevan el análisis y diseño del proyecto, además porque es una metodología iterativa e incremental.
- Se utilizaron los diagramas de comportamiento y de estructura de UML para la documentación del proyecto en todas sus etapas.
- Se utilizó el modelo Entidad Relación para modelar la base de datos del sistema así como una herramienta CASE para el apoyo en el diseño.
- Se logró confeccionar las pantallas de entrada como de salida para orientar a la etapa de implementación del sistema
- Se implementó el sistema con tecnologías orientadas a la web es decir: servidor apache, base de datos Mysql y lenguajes de programación PHP, HTML, plantillas y JavaScript.

5.2 RECOMENDACIONES

Se recomienda utilizar como base el presente trabajo para integrarlo con otros subsistemas como: contable, control de personal, seguridad, etc. debiendo ser a requerimiento del usuario final.