

Universidad de las Ciencias Informáticas
Facultad 4



**Título: Concepción y Desarrollo del
Módulo de Servicio de Información
del Proyecto Telebanca.**

Trabajo de Diploma para optar por el título de

Ingeniero en Ciencias Informáticas

Autor(es): Yarisell Ramirez Morejón

Dagmar Vázquez Padrón

Johan Morgado Estela

Tutor(es): Ing. Yuliesky Torres Iglesias

Junio 2007

DECLARACIÓN DE AUTORÍA

Por este medio declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) a hacer uso del mismo en su beneficio.

Para que así conste firmamos el presente documento ____ a los días del mes de junio del año 2007.

Firma del Autor
Yarisell Ramirez Morejón

Firma del Autor
Dagmar Vázquez Padrón

Firma del Autor
Johan Morgado Estela

Firma del Tutor
Ing. Yuliesky Torres Iglesias

DATOS DE CONTACTO

Ing. Yuliesky Torres Iglesias

Graduado de Ingeniero Informático en el 2005 y profesor instructor con dos años de experiencia docente en la Universidad de las Ciencias Informáticas (UCI). Actual jefe del Proyecto Telebanca.

AGRADECIMIENTOS

A cada uno de nuestros profesores que a lo largo de nuestra carrera nos dieron lo mejor de sí, en conocimiento y talento personal, combinando brillantemente maestría y sencillez.

A nuestros padres y hermanos: Que sacrificaron momentos de sus vidas para que pudiéramos seguir adelante en nuestros estudios.

A nuestros amigos y hermanos de estudios que nos han ayudado para salir adelante y que siempre han confiado en nosotros para culminar esta tarea.

A todas aquellas personas que en mayor o menor medida brindaron su colaboración desinteresada.

Quisiéramos expresar nuestro cordial agradecimiento a todas aquellas personas que de una forma u otra nos han orientado y guiado en el presente trabajo, y en especial a nuestro tutor, por su dedicación y ayuda constante.

DEDICATORIA

A Nuestro Comandante en Jefe Fidel Castro, por hacer germinar la semilla de la Informática y cultivarla cada día con amor, ahínco y dedicación. Por darnos la posibilidad de ser profesionales competentes, encargados de desarrollar esta ciencia en casa y en otros países del mundo. Por haber cultivado y llevado a la práctica la maravillosa idea del nacimiento de esta escuela, donde hemos hecho realidad nuestra verdadera y necesaria vocación.

A nuestros padres y hermanos: Que sacrificaron momentos de sus vidas para que pudiéramos seguir adelante en nuestros estudios.

A nuestros amigos y hermanos de estudios que nos han ayudado para salir adelante y que siempre han confiado en nosotros para culminar esta tarea.

A todas aquellas personas que en mayor o menor medida brindaron su colaboración desinteresada.

RESUMEN

Debido a la situación que existe hoy en nuestro país con respecto al pago de los servicios surge la necesidad de crear un sistema para la gestión de este proceso, el software Telebanca, el cual tendrá un alto valor social, orientado a elevar la eficiencia del pago de los servicios tanto para la población, las empresas e instituciones que participan en el servicio. Para lograr una mayor agilidad de este proceso se requiere de la implementación del Módulo de Servicio de Información el cual facilitará ese primer contacto entre las operadoras y el cliente. Su objetivo principal es brindar la información más actualizada con tal de resolver los intereses más comunes de los clientes, ya sean preguntas del servicio como información precisa de las empresas asociadas a la Banca Telefónica.

Si se desarrolla dicho sistema automatizado para la gestión de la información de los procesos que realiza el Banco permitirá una mayor rapidez y certeza para la toma de decisiones.

El Módulo de Servicio de Información brindará al cliente información de la de agenda electrónica, que reúne todos los datos de las entidades pertenecientes al Sistema Bancario Nacional o que se relacione con este. Así como información de los procesos solicitados por los clientes a través de la operadora que lo atiende, lo cual será registrado por el sistema. Existirá un Administrador de Información que gestionará la información que se brindará y hará reportes de acuerdo a las necesidades.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA	II
RESUMEN	III
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1 Estado del arte	5
1.1.1 ¿Qué es un sistema de información?	5
1.1.2 ¿Qué es un Call Center?	6
1.2 Ámbito nacional e internacional	7
1.3 Herramientas y Tecnologías utilizadas	9
1.3.1 .NET Framework	9
1.3.2 Lenguaje C#	11
1.3.3 Aplicación Web	12
1.3.4 Tecnología ASP .NET	13
1.3.5 Aplicación Web ASP.NET	14
1.4 Programación por capas	15
1.5 Metodologías de desarrollo del software	16
1.5.1 Proceso Unificado Rational (RUP)	16
1.5.2 Lenguaje unificado de Modelado (UML)	17
1.6 Selección de la metodología a utilizar	17
1.7 Herramientas utilizadas	18
1.7.1 Visual Studio .Net	18
1.7.2 Microsoft Visual SourceSafe	19
1.7.3 Crystal Reports	21
1.7.4 Rational Rose	22
1.8 Conclusiones	23

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	24
2.1 Introducción	24
2.2 Objeto de estudio	24
2.2.1 <i>Problema y situación problemática</i>	24
2.2.2 <i>Objeto de automatización</i>	24
2.2.3 <i>Información que se maneja</i>	25
2.2.4 <i>Propuesta de sistema</i>	25
2.3 Modelo del dominio	25
2.3.1 <i>Diagrama de clases del dominio</i>	26
2.4 Especificación de los requisitos de software	26
2.4.1 <i>Requerimientos Funcionales</i>	26
2.4.2 <i>Requisitos No Funcionales</i>	27
2.5 Definición de los casos de uso	29
2.5.1 <i>Definición de los actores</i>	29
2.5.2 <i>Listado de los casos de uso. (Ver Anexo 2)</i>	29
2.5.3 <i>Diagrama de casos de uso</i>	29
2.5.4 <i>Casos de uso expandidos</i>	30
2.6 Conclusiones	63
 CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	 64
3.1 Introducción	64
3.2 Análisis	64
3.2.1 <i>Diagrama de clases del análisis</i>	64
3.3 Diseño	65
3.3.1 <i>Diagramas de interacción</i>	65
3.3.2 <i>Diagrama de clases del diseño</i>	70
3.3.3 <i>Descripción de las Clases</i>	71
3.3.4 <i>Diagrama de clases persistentes</i>	89
3.3.5 <i>Tratamiento de errores</i>	91
3.4 Conclusiones	91

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA	92
4.1 Introducción	92
4.2 Implementación	92
4.2.1 <i>Diagrama de despliegue</i>	92
4.2.2 <i>Diagrama de componentes</i>	93
4.3 Prueba	97
4.3.1 <i>Modelo de prueba</i>	98
4.1 Conclusiones	117
CONCLUSIONES	118
RECOMENDACIONES	119
BIBLIOGRAFÍA	120
GLOSARIO	123

INTRODUCCIÓN

Nuestro país está consciente de que una sociedad para ser más eficaz, eficiente y competitiva debe aplicar la informatización en todas sus esferas y procesos. En este sentido, Cuba ha identificado desde muy temprano la conveniencia y necesidad de dominar e introducir en la práctica social las Tecnologías de la Información y las Comunicaciones y lograr una cultura digital como una de las características imprescindibles del hombre nuevo.

La informatización de la sociedad se define en Cuba como el proceso de utilización ordenada y masiva de las TIC para satisfacer las necesidades de información y conocimiento de todas las personas y esferas de la sociedad. Este proceso busca lograr más eficacia y eficiencia, que permitan una mayor generación de riquezas y hagan sustentable el aumento sistemático de la calidad de vida de los cubanos. En su actual situación de limitaciones económicas, tecnológicas y de comunicación, Cuba ha decidido adoptar como opción de desarrollo inicial el uso social intensivo de sus escasos recursos de conectividad y medios técnicos.

La estrategia de informatización, como expresión del proceso revolucionario cubano, tiene al ciudadano en el centro de sus objetivos, buscando elevar su calidad de vida en su desempeño familiar, laboral, educacional, cultural, social y político.

La Universidad de las Ciencias Informáticas tiene como misión el desarrollo de aplicaciones para la informatización de la sociedad, para esto uno de los proyectos de la facultad 4 está encargado de brindar una propuesta de la aplicación Telebanca, cuyo propósito es darle soporte al trabajo de las operadoras del Call Center y se basa fundamentalmente en gestionar el pago de servicio a tenedores de Cuentas de Ahorro asociados a Tarjetas Débito RED del Banco Metropolitano (BM) a través de las operadoras de la Banca Telefónica.

En la actualidad en nuestro país la población posee grandes problemas para acceder a la información que necesita sobre los procesos bancarios, no existe la cantidad de

personal suficiente para responder las dudas e inquietudes referentes a cualquier tipo de servicio bancario. Además se necesita tener toda la información centralizada para facilitar que el cliente obtenga rápidamente la respuesta.

Para que Telebanca ofrezca sus servicios es necesaria la realización del Módulo de Servicio de Información, con este módulo quedará habilitada una vía para brindar información a la población de los servicios que realiza el banco y se logrará la gestión de información con mayor rapidez y certeza para la toma de decisiones de los clientes.

Este trabajo surge como necesidad de resolver el siguiente **problema**: ¿Cómo facilitar la gestión de la información de los procesos del Banco al cliente mediante un sistema automatizado?

Con vista a la solución del problema antes planteado se decide como **objeto de estudio** la gestión de la información de los procesos bancarios.

El **objetivo general** del presente trabajo es desarrollar el Módulo de Servicio de Información del proyecto Telebanca el cual facilitará ese primer contacto entre las operadoras del Call Center y el cliente.

Dentro de los **objetivos específicos** se encuentran:

- Analizar aspectos teóricos, conceptuales de la información que brinda el Banco a sus clientes.
- Realizar el análisis y diseño del Módulo de Servicio de Información.
- Implementar el Módulo de Servicio de Información.
- Probar el Módulo de Servicio de Información.

Para cumplir con dichos objetivos se proponen las siguientes **tareas de la investigación**:

1. Estudiar el proceso de información por parte del Banco al cliente.
2. Estudiar otros sistemas existentes que se encargan de brindar información al cliente mediante un Call Center.
3. Aplicar la metodología RUP al proceso de desarrollo de la aplicación.
4. Desarrollar la implementación dicho módulo.
5. Aplicar las pruebas pertinentes para el correcto funcionamiento del módulo.

Definiendo así el **campo de acción** como la gestión de la información de los procesos bancarios cubanos.

La investigación se sustenta en la siguiente **hipótesis** si se desarrolla un sistema automatizado para la gestión de la información de los procesos que realiza el Banco permitirá una mayor rapidez y certeza para la toma de decisiones.

Con el desarrollo de este sistema se espera que los involucrados en este proceso cuenten con una herramienta de trabajo que sea capaz de gestionar de una forma rápida y eficiente la información que se maneja con respecto a los procesos bancarios.

El presente documento consta de cuatro capítulos:

En el capítulo 1 se recoge el análisis de la información existente acerca del tema a tratar y las tendencias actuales que existen en el mundo. También se incluye una descripción del lenguaje de programación a utilizar tanto para la implementación.

En el capítulo 2 se describen aspectos del sistema, tales como: la especificación de los detalles de la construcción de la herramienta, una propuesta de solución, requerimientos funcionales y no funcionales y los casos de uso descritos y desarrollados.

En el capítulo 3 se plantean detalles relacionados con el diseño que se propone, los diagramas de interacción y diagramas de clases, es decir, análisis y diseño del sistema.

En el capítulo 4 se realiza el diagrama de despliegue, el diagrama de componentes y la descripción de los casos de prueba de integración, para cada caso de uso.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Estado del arte

Cada día las empresas enfrentan nuevos desafíos: deben alcanzar niveles máximos de calidad y satisfacción de sus clientes, lanzar nuevos productos, posicionar y consolidar los ya existentes, buscar mayor participación de mercado, etc.

Enfrentar cada uno de estos retos implica cubrir necesidades de información mayores día a día y, aunque la estadística, la computación y la ingeniería están muy desarrolladas, casi nunca se posee toda la información que se necesita para tomar decisiones, esto es debido en gran parte a que la economía y las personas son altamente variables.

El desarrollo de la atención al cliente se ha convertido en una de las estrategias fundamentales para las empresas. En este ámbito el Call Center ha pasado a ser una de las piezas claves en el conjunto de las gestiones de la calidad de los servicios y la satisfacción de los clientes.

Esta dinámica, ligada a la necesidad de obtener utilidades, obliga a las empresas a requerir más y mejor información y a diseñar sistemas que les permiten conseguirla y gestionarla de la mejor manera.

1.1.1 ¿Qué es un sistema de información?

Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio.(PERALTA)

Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información.

Entrada de Información: Es el proceso mediante el cual el sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos. Esto último se denomina interfases automáticas.(PERALTA)

Almacenamiento de información: El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos.

Procesamiento de Información: Es la capacidad del sistema de información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

Salida de Información: La salida es la capacidad de un sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Es importante aclarar que la salida de un sistema de información puede constituir la entrada a otro sistema de información o módulo. En este caso, también existe una interfase automática de salida. Por ejemplo, el sistema de control de clientes tiene una interfase automática de salida con el sistema de Contabilidad, ya que genera las pólizas contables de los movimientos procesales de los clientes.(PERALTA)

1.1.2 ¿Qué es un Call Center?

Un Call Center es un centro de atención telefónica que se utiliza como vínculo entre la empresa y el cliente y está enmarcado en el departamento de atención al cliente. Es una plataforma telefónica su función principal es facilitar la comunicación entre las personas, a través del teléfono. Permite que sus clientes se contacten con su empresa o sean contactados para solucionar sus problemas e inquietudes.(AYUDA)

Los Call Centers nacieron de la oportunidad de prestar un servicio inmediato al cliente a través del teléfono. Al principio éstos tenían un carácter más informativo y de servicio adicional a los productos. Después, debido a la fuerte competencia, el Call Center se convirtió en un servicio más habitual y necesario que de lujo, como había sido en su origen.

Importancia de los Call Center

La implantación de un Call Center trae muchas ventajas reduciendo los costos operativos de duración de llamadas, tiempos de espera, la complejidad de gestión de cada llamada, incrementa la calidad del servicio de información así como la satisfacción del Cliente. La operación en un Call Center es completamente automatizada para garantizar agilidad, calidad, precisión y bajos costos en la atención de sus clientes.

Como en cualquier otro sector, la innovación tecnológica ha sido un factor determinante en la evolución de los Call Centers, pues ha ayudado a reducir el tiempo de respuesta, a mejorar los procesos y los servicios al cliente llegando a abastecer todo el ciclo de relación.

En las últimas décadas, la demanda de servicios de los Call Centers ha experimentado un importante crecimiento. Se debe tener en cuenta que se trata de un contexto donde hay distintos objetivos públicos que están cambiando. Entre ellos se encuentra el consumidor que es más exigente y pretende una interacción eficiente con su empresa proveedora de cualquier producto o servicio. Y el Call Center parece ser la herramienta idónea para proponer y administrar esa interacción.

Debido a la importancia que han adquirido hoy en día los Call Centers como plataformas telefónicas se puede resaltar que han permitido facilitar la comunicación entre las personas a través del teléfono y han logrado que sus clientes se contacten con su empresa o sean contactados para solucionar sus problemas e inquietudes, han tenido una magnífica aceptación por parte de los clientes porque con una simple llamada telefónica resuelven sus dudas, han provocado que en el mundo hayan surgido un sin número de estos sistemas automatizados dirigidos al control eficiente y veloz de dichos procesos. A nivel nacional e internacional se han desarrollado un gran número de aplicaciones para dar información de los servicios de determinada empresa mediante un Call Center a los clientes, con similitudes y diferencias desde el punto de vista de implementación y servicios brindados.

1.2 Ámbito nacional e internacional

A continuación se hace un estudio de algunos de los sistemas existentes en el ámbito nacional e internacional para la gestión de la información.

En nuestro país se encuentra el Call Center ETECSA, este es fundamental para el negocio porque ayuda a las Empresas a garantizar servicios de atención y gestión a sus clientes sin necesidad de invertir en infraestructura propia para lograr esto cuenta con un equipo de ejecutivas calificadas, variedad de servicios y la más avanzada tecnología, ofreciendo diversos servicios entre los que se encuentra el servicio de información al cliente, con este Call Center las empresas se hacen más competitivas y establecen relaciones más eficientes con sus clientes.

En el ámbito internacional existen variedades de Call Center ya implementados como lo es: ENTEL Call Center, por su parte asegura la absoluta confidencialidad de la información a través de elementos técnicos que permiten asegurar los compromisos de confianza contraídos con sus clientes. Los servicios prestados por ENTEL Call Center pretenden ajustarse a las necesidades particulares de cada empresa. Una buena alternativa son los llamados servicios de atención de clientes, que permiten recibir las consultas y mantener una comunicación fluida entre la empresa que contrata a ENTEL Call Center y sus clientes potenciales y reales. Entre otros servicios se destaca el apoyo al cliente. Para cumplir efectivamente con los diferentes servicios, ENTEL Call Center cuenta con ejecutivas telefónicas especialmente capacitadas para ofrecer las mejores condiciones de comunicación con cada uno de sus clientes. Además, para asegurar el más alto nivel de atención, disponen de profesionales especializados en la supervisión y control de calidad del servicio, además de ofrecer otros servicios.

Call Center Central de Reservas de Sierra Nevada: desde su implantación ha tenido una magnífica aceptación por parte de los clientes, ya que con una simple llamada telefónica resuelven sus dudas, compra un paquete de esquí, realiza cualquier tipo de operación del Club Sierra Nevada (registro, información del club, solicitud de tarjeta, condiciones de uso, entrega de la tarjeta, obtienen ayuda, etc.) y son atendidos en todo momento por un personal amable y servicial. El Call Center (Central de Llamadas) se implantó para atender el creciente número de clientes que cada vez más optan por este destino turístico. De esta forma, el desarrollo de nuevas soluciones tecnológicas ha permitido que los visitantes puedan disfrutar de servicios, como la reserva de plazas desde sus hogares o una mayor rapidez en el acceso a las pistas.

Entre muchas de sus aplicaciones, permite la atención centralizada de todas las llamadas telefónicas de los usuarios por operadores especializados en atención al cliente. Uno de los objetivos de este centro de

llamadas es la rapidez en la respuesta, dar una solución a cualquier problema y obtener buenos resultados de satisfacción de los clientes, son los objetivos primordiales de todos los agentes que conforman el Call Center.

De acuerdo a todas estas características y prestaciones se ha propuesto realizar el software Telebanca, de modo que para el total funcionamiento de este sistema se plantea realizar un Módulo de Servicio de Información para gestionar la información que se maneja en el mismo brindando un servicio con mayor rapidez y certeza para la toma de decisiones.

1.3 Herramientas y Tecnologías utilizadas

1.3.1 .NET Framework

Microsoft .NET Framework representa un nuevo paradigma en el desarrollo de software, y los profesionales de tecnologías de la información tienen que afrontar la tarea de administrar e implementar estas nuevas aplicaciones y componentes en sus infraestructuras ya existentes.

.NET Framework es un entorno multilenguaje que permite generar, implantar y ejecutar Servicios Web y aplicaciones XML. Es una nueva plataforma diseñada para simplificar el desarrollo de aplicaciones en el entorno distribuido de Internet. Además permite el desarrollo de aplicaciones a través del uso de un conjunto de herramientas y servicios que proporciona, su diseño está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Ofrecer un entorno de ejecución de código que fomente la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.
- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan secuencias de comandos o intérpretes de comandos.

- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.

.NET Framework contiene dos componentes principales:

Common Language Runtime (CLR): es el fundamento de la tecnología. El motor de tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la interacción remota, al tiempo que aplica una seguridad estricta a los tipos y otras formas de especificación del código que fomentan su seguridad y solidez.

Además, el entorno administrado del motor de tiempo de ejecución elimina muchos problemas de software comunes.

Biblioteca de clases: el otro componente principal de .NET Framework, es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios Web Forms y los servicios Web XML.

Además son tremendamente importantes para proporcionar interoperabilidad al lenguaje, porque permiten a los desarrolladores usar una sola interfaz de programación para toda la funcionalidad expuesta por el CLR.

.NET Framework proporciona a los desarrolladores una plataforma robusta y ampliable para crear soluciones escalables y seguras.

1.3.2 Lenguaje C#

C# (leído en inglés “C Sharp” y en español “C Almohadilla”) es el nuevo lenguaje de propósito general diseñado por Microsoft para su plataforma.

Es posible programar en la plataforma .NET en prácticamente cualquier lenguaje, pero Microsoft ha decidido crear uno nuevo porque ha visto conveniente poder disponer de un lenguaje diseñado desde 0 con vistas a ser utilizado en .NET, un lenguaje que no cuente con elementos heredados de versiones anteriores e innecesarios en esta plataforma y que por tanto sea lo más sencillo posible para programarla aprovechando toda su potencia y versatilidad. Programar usando C# es mucho más fácil e intuitivo que hacerlo en cualquiera de los otros lenguajes ya que C# carece de elementos heredados innecesarios en .NET. Por esta razón, se suele decir que C# es el lenguaje nativo de .NET.

Es el primer lenguaje orientado a componentes en la familia de lenguajes C y C++. Es un lenguaje de programación simple, moderno, orientado a objetos y con un Sistema de tipos seguro derivado de C y C++, además combina la alta productividad de Microsoft Visual Basic y la eficacia bruta de C++.

C# combina los mejores elementos de múltiples lenguajes de amplia difusión como C++, Java, Visual Basic o Delphi. La idea principal detrás del lenguaje es combinar la potencia de lenguajes como C++ con la sencillez de lenguajes como Visual Basic, y que además la migración a este lenguaje por los programadores de C/C++/Java sea lo más inmediata posible.

C# es un lenguaje de programación que toma las mejores características de lenguajes preexistentes, a continuación se recoge de manera resumida alguna de las mismas.

Es un lenguaje sencillo, eliminando muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET como por ejemplo: El código escrito en C# es autocontenido, lo que significa que no necesita de ficheros adicionales al propio fuente, el tamaño de los tipos de datos básicos es fijo e independiente del compilador, Sistema operativo o máquina para quienes se compile (no como en C++), lo que facilita la portabilidad del código además no se incluyen elementos poco útiles de lenguajes como C++ tales como macros, herencia múltiple o la necesidad de un operador diferente del punto (.).

C# es un lenguaje moderno incorporando en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones.

Como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos el cual soporta todas las características propias del paradigma de programación orientada a objetos: encapsulación, herencia y polimorfismo.

Su propia sintaxis incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas. Es decir, la sintaxis de C# permite definir cómodamente propiedades (similares a campos de acceso controlado), eventos (asociación controlada de funciones de respuesta a notificaciones) o atributos (información sobre un tipo o sus miembros).

C# incluye mecanismos que permiten asegurar que los accesos a tipos de datos siempre se realicen correctamente, lo que permite evita que se produzcan errores difíciles de detectar por acceso a memoria no perteneciente a ningún objeto y es especialmente necesario en un entorno gestionado por un recolector de basura.

C# permite definir, a través de estructuras, tipos de datos para los que se apliquen las mismas optimizaciones que para los tipos de datos básicos, Para facilitar la legibilidad del código y conseguir que los nuevos tipos de datos básicos que se definan a través de las estructuras estén al mismo nivel que los básicos predefinidos en el lenguaje C# permite redefinir el significado de la mayoría de los operadores.

La compatibilidad de este lenguaje es muy importante manteniendo una sintaxis muy similar a C, C++ o Java.

1.3.3 Aplicación Web

Una aplicación web es una aplicación informática que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet. Las aplicaciones web son populares debido a la práctica del navegador web como cliente ligero. La habilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

Las aplicaciones web le brindan la comodidad de incorporar aplicaciones que pueden ser accesibles desde cualquier punto de conexión a Internet, permitiendo a sus empleados y/o clientes operar con ellas con los mínimos requerimientos: un navegador de Internet y conocimientos básicos sobre éste.

Una aplicación web está comúnmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web es la primera capa, un motor usando alguna tecnología web dinámica (ejemplo: PHP, Java Servlets o ASP) es la capa de en medio, y una base de datos como última capa. El navegador web manda peticiones a la capa media, que la entrega valiéndose de consultas y actualizaciones a la base de datos generando una interfaz de usuario.

Mediante una Aplicación Web la empresa podrá ofrecer servicios como teletrabajo, teleformación, consultoría personalizada y comercio electrónico, entre otros.

1.3.4 Tecnología ASP .NET

Es un conjunto de tecnologías de desarrollo de aplicaciones web comercializado por Microsoft.

ASP.NET es más que una nueva versión de las páginas Active Server (ASP); proporciona un modelo de desarrollo Web unificado que incluye los servicios necesarios para que los programadores creen aplicaciones Web para la empresa. Si bien ASP.NET es en gran medida compatible con la sintaxis de ASP, proporciona también un modelo de programación y una estructura nuevos para crear aplicaciones más escalables y estables que ayuden a proporcionar mayor protección. Las aplicaciones ASP se pueden ampliar agregándoles funcionalidad de ASP.NET. ASP.NET ha cambiado de manera significativa con respecto a ASP, siendo más estructurado y orientado a objetos.*(Información básica sobre tecnología ASP.NET)*

ASP.NET es un entorno compilado basado en .NET. Se pueden crear aplicaciones en cualquier lenguaje compatible con .NET, como Visual Basic .NET, C# y JScript .NET. Además, .NET Framework está disponible en su totalidad para cualquier aplicación ASP.NET.

El marco de trabajo de páginas ASP.NET crea una abstracción de la interacción cliente-servidor Web tradicional de forma que se puedan programar aplicaciones mediante métodos y herramientas tradicionales que admitan el diseño rápido de aplicaciones y la programación orientada a objetos.

El marco de trabajo de páginas ASP.NET es un entorno de programación que se ejecuta en un servidor Web para producir y administrar de forma dinámica páginas de formularios Web Forms. En Visual Studio,

Web Forms proporciona un diseñador de formularios, un editor, controles y depuración, lo que en conjunto permite generar rápidamente interfaces de usuario programables basadas en el servidor para exploradores y dispositivos cliente Web. *(Introducción a ASP.NET)*

Las páginas de formularios Web Forms se ejecutan en cualquier explorador o dispositivo cliente. Sin embargo, este tipo de páginas se pueden diseñar para ejecutarse en un explorador determinado, como Microsoft® Internet Explorer 5 y aprovechar así todas las funciones de un explorador o dispositivo cliente determinado.

ASP.NET trae diversas mejoras entre las cuales se destacan:

- **Rendimiento:** la aplicación se compila en una sola vez al lenguaje nativo, y luego, en cada petición tiene una compilación. Es decir se compila desde el código nativo, lo que permite mucho mejor rendimiento. También permite el almacenamiento del caché en el servidor.
- **Rapidez en programación:** mediante diversos controles, podemos con unas pocas líneas y en menos de 5 minutos mostrar toda una base de datos y hacer rutinas complejas.
- **Servicios Web:** trae herramientas para compartir datos e información entre distintos sitios.
- **Seguridad:** tiene diversas herramientas que garantizan la seguridad de nuestras aplicaciones.

1.3.5 Aplicación Web ASP.NET

ASP.NET facilita el desarrollo de aplicaciones si lo comparamos con el modelo ASP clásico, por lo que la productividad de los programadores mejorará considerablemente. Esta plataforma permite dotar de funciones adicionales a una Aplicación Web por escribir una menor cantidad de código, entre otras características.

Las partes de una Aplicación Web ASP.NET incluyen:

- Formularios Web Forms o páginas .aspx.

Los formularios Web Forms y las páginas .aspx proporcionan la interfaz de usuario para la aplicación Web.

- Páginas con código trasero.

Las páginas con código trasero están asociadas a los formularios Web Forms y contienen el código del servidor para el formulario Web Form.

- Archivos de configuración.

Los archivos de configuración son archivos XML que definen la configuración predeterminada para la Aplicación Web y para el servidor Web. Cada Aplicación Web tiene un archivo de configuración Web.config. Además, cada servidor Web tiene un archivo machine.config.

- Archivo Global.asax.

Los archivos Global.asax contienen el código necesario para responder a los eventos a nivel de aplicación provocados por ASP.NET.

1.4 Programación por capas

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, simplemente es necesario conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El diseño más utilizado actualmente es el diseño en tres capas. (Ver Anexo 1).

Capas o niveles

1.- **Capa de presentación:** es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

Esta capa contiene todas las clases de interfaz de usuarios que representan las pantallas de la aplicación que el usuario ve. Esta capa depende de la capa de Negocio y de la capa de datos.

2.- **Capa de negocio:** es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.

En esta capa se encuentran todas las clases controladoras que representan el comportamiento de la aplicación, según los casos de uso. Esta capa representa la frontera del cliente con la capa Intermedia. La capa de Negocio depende de la capa de datos.

3.- **Capa de datos:** es donde residen los datos. Está formada por uno o más gestor de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

1.5 Metodologías de desarrollo del software

1.5.1 Proceso Unificado Rational (RUP)

El Proceso Unificado es un proceso de desarrollo de software configurable que se adapta a proyectos que varían en tamaño y complejidad. Se basa en muchos años de experiencia en el uso de la tecnología de objetos en el desarrollo de software de misión crítica en una variedad de industrias.

Es un marco genérico de trabajo que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud.(UML - RUP)

RUP describe cómo obtener los requerimientos, cómo organizarlos, cómo documentar los requerimientos de funcionalidad y restricciones, cómo rastrear y documentar las decisiones, y cómo captar y comunicar los requerimientos del negocio. Los casos de uso y los escenarios indicados por el proceso, han probado ser una buena forma de captar requerimientos y guiar el diseño, la implementación y las pruebas.(GUERRERO)

El proceso se basa en diseñar tempranamente una arquitectura base ejecutable apoyando el desarrollo basado en componentes. Para esto RUP propone un modelamiento visual de la estructura y el comportamiento de la arquitectura y las componentes.

No sólo la funcionalidad es esencial, también el rendimiento y la confiabilidad por lo tanto RUP ayuda a planificar, diseñar, implementar, ejecutar y evaluar pruebas que verifiquen estas cualidades.

Los cambios son inevitables en cada proyecto, pero es necesario evaluar si éstos son necesarios y también es necesario rastrear su impacto. RUP indica como controlar, rastrear y monitorear los cambios dentro del proceso iterativo de desarrollo.

RUP divide el proceso de desarrollo en ciclos, donde se obtiene un producto al final de cada ciclo.

1.5.2 Lenguaje unificado de Modelado (UML)

UML es un lenguaje estándar para escribir planos de software. UML puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra gran cantidad de software.

Es un lenguaje muy expresivo, que cubre todas las vistas necesarias para desarrollar y luego desplegar tales sistemas. Aunque sea expresivo, UML no es difícil de aprender ni de utilizar. El vocabulario y las reglas de un lenguaje como UML indican como crear y leer modelos bien formados.(UML, que es?)

1.6 Selección de la metodología a utilizar

Para el desarrollo de este sistema se propone utilizar la metodología Rational Unified Process (RUP). Es importante destacar que en la documentación del trabajo utilizaremos la terminología propia de la metodología a utilizar y como notación utilizaremos el Lenguaje Unificado de Modelado (UML).

1.7 Herramientas utilizadas

Para el desarrollo de este sistema se realizó un análisis sobre las posibles herramientas a utilizar para su confeccionamiento. Teniendo en cuenta este análisis se decidió hacer la siguiente propuesta.

1.7.1 Visual Studio .Net

Visual Studio .NET es un IDE (entorno de programación) desarrollado por Microsoft a partir de 2002. Es para el sistema operativo Microsoft Windows. Visual Studio .NET es la herramienta de desarrollo multilenguaje más completa para construir e integrar rápidamente aplicaciones y servicios Web XML. Aumenta de un modo extraordinario la productividad de los desarrolladores y crea nuevas oportunidades de negocio. En su diseño se han integrado a fondo los estándares y protocolos de Internet, como XML y SOAP, por lo que Visual Studio .NET simplifica considerablemente el ciclo de vida del desarrollo de aplicaciones.

Visual Studio .NET permite a los programadores generar con rapidez aplicaciones para Internet de próxima generación orientadas a cualquier dispositivo y que se integran en cualquier plataforma. Con el más moderno y completo entorno de programación, Visual Studio .NET ofrece a los programadores las herramientas para integrar soluciones en diferentes sistemas operativos y lenguajes. Con Visual Studio .NET, los programadores pueden convertir fácilmente lógica empresarial existente en servicios Web XML reutilizables, encapsulando procesos y dejándolos disponibles para las aplicaciones en cualquier plataforma. Los programadores pueden incorporar fácilmente cualquier número de servicios Web XML que estén catalogados y disponibles en cualquier directorio UDDI (integración, descubrimiento y descripción universal) independiente, proporcionando una base fuerte de servicios y lógica empresarial para sus aplicaciones. (*Visual Studio .NET Professional*)

Visual Studio .NET permite a los programadores crear e implementar lógica empresarial importante basada en un servidor. Históricamente, la programación basada en servidor ha sido una tediosa tarea de codificación, propensa a generar errores y difícil de probar. Con Visual Studio .NET, los programadores pueden componer visualmente componentes de nivel medio utilizando el Diseñador de componentes visual, que permite a los programadores arrastrar objetos no visuales —como colas de mensajes, temporizadores y registros de eventos— hasta una superficie de diseño desde el Explorador de

servidores, una nueva ventana de herramientas que descubre automáticamente todos los recursos necesarios basados en servidor. Además posee un eficaz diseñador para páginas Web que sirve de ayuda a los programadores a sentirse cómodos durante el diseño de soluciones complejas basadas en el Web.

Visual Studio .NET constituye el núcleo del desarrollo .NET. Es un completo entorno de desarrollo en el que podemos diseñar, desarrollar, depurar e implementar nuestras aplicaciones .NET y nuestros Servicios Web XML. Visual Studio .NET, como herramienta de desarrollo, proporciona las características siguientes:

- Soporta varios lenguajes de desarrollo.
- Herramientas para generar aplicaciones Web, aplicaciones Windows y servicios Web XML.
- Herramientas de acceso a datos.
- Procesamiento completo de errores, incluyendo depuración local, depuración remota y uso de trazas.

Visual Studio .NET 2005 es la versión definitiva en inglés vio la luz en Noviembre del 2005.

- Incorpora .NET Framework 2.0.
- Hay más ediciones diferenciadas por el precio y las características.
- Ayuda con refactorización, reestructuran automáticamente el código fuente, por ejemplo, promocionando variables locales a parámetros o convirtiendo un bloque de código en un método.
- El desarrollo de páginas con ASP.NET ha cambiado.
- Soporte para el nuevo software servidor Team System.
- Añadido soporte de tests para todo tipo de aplicaciones.

1.7.2 Microsoft Visual SourceSafe

Microsoft Visual SourceSafe es un sistema de control de versiones en el nivel de archivos, que permite a muchos tipos de organizaciones trabajar en distintas versiones de un proyecto al mismo tiempo. Esta funcionalidad es especialmente ventajosa en un entorno de desarrollo de software, donde se usa para mantener versiones de código paralelas. Sin embargo, el producto también se puede utilizar para mantener archivos en cualquier otro tipo de equipo. Visual SourceSafe admite el desarrollo multiplataforma al permitir la edición y el uso compartido de los datos. Se ha diseñado para controlar los

problemas de seguimiento y portabilidad que implica mantener una base de control de código fuente, como una base de código de software, en varios sistemas operativos. Para los desarrolladores, Visual SourceSafe aloja código reutilizable u orientado a objetos. Asimismo, facilita el seguimiento de las aplicaciones que utilizan módulos de código concretos. (*Introducción a Visual SourceSafe*)

Visual SourceSafe incluye, como mínimo, las siguientes funciones:

- Ayuda al equipo a evitar la pérdida accidental de archivos.
- Permite realizar un seguimiento de las versiones anteriores de un archivo.
- Admite la bifurcación, el uso compartido, la combinación y la administración de versiones de archivos.
- Realiza el seguimiento de las versiones de proyectos completos.
- Realiza el seguimiento del código modular (un archivo que se reutiliza, o se comparte, en varios proyectos).

Visual SourceSafe permite compartir archivos entre proyectos de forma rápida y eficaz. La organización de los archivos en proyectos hace que la coordinación de los equipos sea un proceso intuitivo. Cuando se agrega un archivo a Visual SourceSafe, este archivo se almacena en la base de datos y queda a disposición de otros usuarios. Los cambios realizados en él se guardan para que cualquier usuario pueda recuperar una versión anterior en todo momento. Los miembros de su equipo podrán ver la última versión de un archivo, realizar cambios en sus copias locales y guardar nuevas versiones en la base de datos. Cuando un conjunto de archivos está listo para entregarse, Visual SourceSafe permite compartir y obtener las distintas versiones del conjunto con facilidad.

Cada vez más, los desarrolladores obtienen acceso a las funciones de Visual SourceSafe desde sus entornos de desarrollo en programas de terceros. Visual SourceSafe se puede integrar con toda facilidad en Visual Studio y otras herramientas de desarrollo como Microsoft Access.

Visual SourceSafe 6.0

Visual SourceSafe 6.0 es el sistema de administración de código fuente y control de versiones más fácil y productivo para equipos de desarrollo que utilizan Microsoft Visual Studio® .NET. Los desarrolladores pueden administrar el código fuente, contenido Web y cualquier otro tipo de archivo de manera fácil y segura, todo desde Visual Studio .NET.

Ofrece las siguientes características:

- Mayor productividad del desarrollador.
- Mejor calidad del software.
- Administración de software en equipo.

1.7.3 Crystal Reports

Crystal Reports es un mecanismo "oficial" del Visual Studio de creación de reportes. Crystal Reports es un producto muy desarrollado y complejo para la creación de reportes, y cuenta con una infinidad de opciones. Crystal es un producto creado en su esencia orientado al usuario final; es decir que un ejecutivo pueda crear sus propios informes sin necesidad de asistencia de un desarrollador. (*Crystal Reports 10 .NET*)

Es la herramienta de elaboración de informes estándar para Visual Studio .NET. Esta herramienta permite crear contenido interactivo con calidad de presentación, punto fuerte de Crystal Reports durante años, en la plataforma .NET. Se puede almacenar informes en plataformas Windows y Web, así como publicar informes de Crystal como servicios Web de informes en un servidor Web.

Con esta herramienta se puede crear una aplicación web que permita a los usuarios profundizar en un gráfico y filtrar la información en función de sus necesidades. Realmente, el gráfico es un informe de Crystal que interactúa con otros controles de la aplicación.

Crystal Reports ofrece una manera fácil y única, sea cual sea la fuente, para hacer informes, al contrario de lo que sucede con los pertenecientes a los propios entornos de bases de datos. Para el programador, Crystal Reports ofrece la posibilidad de integrar informes y gráficos de calidad profesional, desde cualquier lenguaje anfitrión.

Crystal Reports está dotado de un gestor de diccionarios que permite almacenar todos los datos necesarios para realizar los informes. Se trata de una útil aplicación independiente del generador de informes, que nos permite utilizar la base de datos según las necesidades específicas de nuestro informe, de manera independiente al motor de la base de datos. Además ésta información queda registrada para usos posteriores.(*Crystal Reports 10 .NET*)

Crystal Reports soporta tres tipos principales de informes: de columnas (RPT), de etiquetas (RPL) y multidimensionales (cross-tab, con extensión RPC). De esta manera los ficheros RPT, RPL y RPC son compilados creando un fichero EXE. Posteriormente, el informe compilado, mediante un asistente especial, se puede almacenar en un disquete con todos los archivos necesarios para su distribución.

Es una herramienta para realizar exclusivamente informes, con lo que de alguna manera, el usuario ya debe saber anticipadamente los datos que desea.

Crystal Reports 10

Con Crystal Reports 10 puede crear rápidamente informes complejos de aspecto profesional. En lugar de utilizar codificación, la interfaz de Crystal Report Designer le permite crear y dar formato al informe que desee. El eficaz motor de informes procesa los criterios de formato, agrupamiento y creación de gráficos que especifique.

1.7.4 Rational Rose

Rational Rose es la herramienta CASE desarrollada por los creadores de UML (Booch, Rumbaugh y Jacobson), que cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables. Facilita el desarrollo de un proceso cooperativo en el que todos los agentes tienen sus propias vistas de información (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue), pero utilizan un lenguaje común para comprender y comunicar la estructura y la funcionalidad del sistema en construcción.

Es la herramienta líder en el mundo de modelación visual para el proceso de modelación del negocio, análisis de requerimientos y diseño de arquitectura de componentes.

1.8 Conclusiones

Con las particularidades antes expuestas de las aplicaciones web y de los servicios que se pueden brindar a través de las mismas se puede decir que su uso se está haciendo cada vez más frecuentes. Debido a que este tipo de aplicación debe acceder a los datos de la empresa para ofrecer información actualizada y que permite a sus empleados y/o clientes operar con ellas con los mínimos requerimientos: un navegador de Internet y conocimientos básicos sobre éste; será utilizada una Aplicación Web para la implementación de este módulo. En nuestra propuesta de sistema se utilizará la programación en tres capas.

Después de haber realizado un análisis de la información recopilada sobre los diferentes Call Center se determinó que estos poseen características que se asimilan al sistema propuesto, por lo que nos basamos en su funcionamiento y desarrollo. Para el sistema se decidió utilizar el lenguaje de programación para aplicaciones web ASP.NET porque nos ofrece diversas funcionalidades ventajosas para nuestra aplicación.

Finalmente para el desarrollo del sistema se utilizarán las siguientes metodologías, tecnologías y herramientas por las características fundamentadas con anterioridad: la plataforma .NET de Microsoft (C#), como metodología RUP, como notación el Lenguaje Unificado de Modelado (UML) y como herramienta el Rational Rose, para la generación de los reportes se utilizará la herramienta Crystal Reports 10 y por último para el control de versiones en el nivel de archivos Microsoft Visual SourceSafe 6.0.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

A través de este capítulo se describe el objeto de estudio, el entorno de trabajo en que se desarrolla la aplicación, se realiza la propuesta del sistema y se analizan los requerimientos funcionales y no funcionales. Se realiza el modelado del dominio y se realiza la definición de los casos de uso de los actores que intervienen en ellos y se muestra el diagrama resultante de casos de uso.

2.2 Objeto de estudio

2.2.1 Problema y situación problemática

Debido al aumento en el desarrollo económico de nuestra población aumentan sus dudas e inquietudes lo cual requiere más personal por parte de las empresas bancarias para dar respuestas inmediatas y brindar la información necesaria a la población, pues la misma para recibir esta información tiene que trasladarse hacia el Banco y en este realizar largas colas para poder ser atendidos o en el caso que la persona llame por teléfono demora en ser atendido por la falta de personal o porque el personal existente tiene otras tareas asignadas por lo cual esa persona no puede ser atendida como debe ser, además existe la necesidad de saber cuáles son las mayores dudas de la población respecto a las prestaciones del Banco y en que dirección hay que centrar los esfuerzos para satisfacer las necesidades de las personas. También existe la problemática de centralizar toda la información referente a los procesos bancarios.

2.2.2 Objeto de automatización

Se desea automatizar los procesos antes mencionados como lo es la gestión de la información donde esta es almacenada para conocer posteriormente cuáles son las mayores dudas que tiene la población respecto a las prestaciones del Banco. Además se dará respuesta a la información solicitada por los clientes respecto a procesos referentes a temas bancarios e informaciones que conciernen a otras entidades de la banca cubana.

Teniendo en cuenta lo antes planteado en relación a la gestión de la información se propone el presente sistema como vía para brindar información a la población de los servicios que realiza la Banca.

2.2.3 Información que se maneja

La información que se maneja son los datos referente a los procesos como el tema, el texto correspondiente al proceso, el subtema, el tema padre, entre otros. En el caso de la agenda electrónica se manipula los datos de una entidad de la agenda electrónica los cuales son: nombre de la entidad, dirección, teléfono, fax, sitios Web relacionados, código anterior, código actual, correos electrónicos.

2.2.4 Propuesta de sistema

El sistema que se propone presenta funcionalidades que ayudarán al trabajo de los involucrados en esta labor de la institución, pues el mismo mostrará una interfaz fácil y amigable donde estos tendrán las herramientas de trabajo necesarias para la gestión de la información de los procesos y la agenda electrónica. Mostrará información importante para los involucrados, pues estos podrán conocer en todo momento determinada información y los temas más solicitados por los clientes y así conocer sus mayores dudas respecto a las prestaciones de la Banca.

2.3 Modelo del dominio

Considerando las descripciones de los procesos anteriormente comentados, se llega a la conclusión de que el negocio que se está estudiando tiene muy bajo nivel de estructuración por lo que se ve en la necesidad de utilizar un modelo del dominio porque permite mostrar los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema.

Se realizó un modelo del dominio debido a las características de nuestro negocio, además de no presentar procesos de negocios bien definidos.

Esto ayuda a los usuarios, clientes y desarrolladores e interesados a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema. Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

2.3.1 Diagrama de clases del dominio

El modelo del dominio se describe mediante diagramas UML, específicamente con un diagrama de clases conceptuales significativas en el dominio del problema.

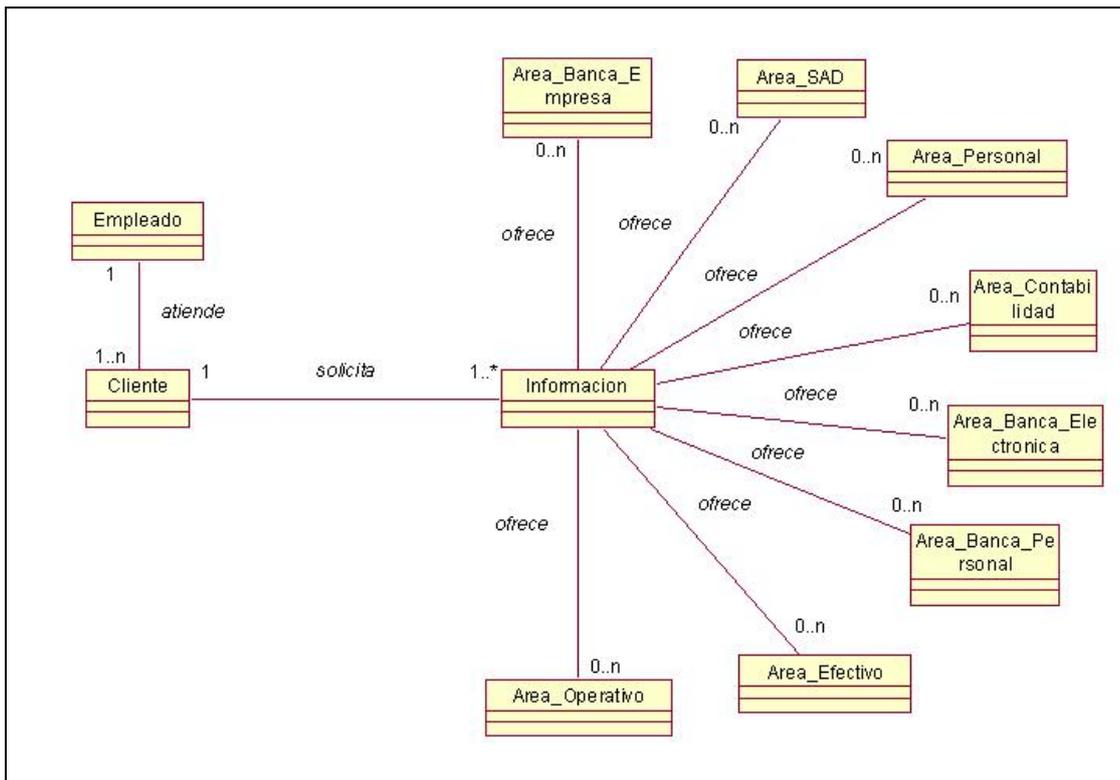


Figura 1. Diagrama del modelo del dominio.

2.4 Especificación de los requisitos de software

2.4.1 Requerimientos Funcionales

RF.1 Gestionar Información.

RF.1.1 Gestionar Agenda Electrónica.

RF.1.1.1 Permitir al Administrador de Información insertar nueva entidad.

RF.1.1.2 Permitir al Administrador de Información modificar datos de la entidad.

RF.1.1.3 Permitir al Administrador de Información eliminar entidad, al ser eliminada se almacena en un histórico.

RF.1.2 Gestionar Información de Procesos.

RF.1.2.1 Permitir al Administrador de Información insertar nueva información de procesos.

RF.1.2.2 Permitir al Administrador de Información modificar datos de la información de procesos.

RF.1.2.3 Permitir al Administrador de Información eliminar información de procesos.

RF.2 Buscar información.

RF.2.1 Buscar en la Agenda Electrónica.

RF.2.2 Buscar Información de Procesos.

RF.2.3 Registrar información consultada.

RF.3 Realizar reportes de las solicitudes de información de los clientes.

2.4.2 Requisitos No Funcionales

1. Requerimientos de apariencia o interfaz externa

La herramienta propuesta será usada por personas que no necesariamente tienen conocimientos profundos de informática, por lo que la interfaz debe ser amigable y fácil de usar, de manera que no sea una dificultad para los usuarios el uso de esta.

2. Requerimientos de Usabilidad

La herramienta será utilizada por operadores que no necesariamente tendrán experiencia en el uso de la computadora, a estos se les dará un adiestramiento básico en el uso de la computadora y fundamentalmente en el uso de la aplicación. Estas personas tendrán un nivel de acceso en la aplicación, en la cual realizarán solamente las acciones que tienen asignadas.

3. Requerimientos de Rendimiento

Para un funcionamiento óptimo de la aplicación se seguirán las diferentes técnicas de elaboración en la Web, que faciliten el rápido acceso a sus páginas. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas en la Base de Datos.

La herramienta propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición. El tiempo de respuesta de la operadora al cliente debe estar alrededor de los 2.5 segundos.

4. Requerimientos de Soporte

Para garantizar el soporte de esta herramienta, se documentará la aplicación con un manual de ayuda para los usuarios, así como la posibilidad de emitir sus quejas y sugerencias a los desarrolladores de la herramienta, por correo o por teléfono, realizar mantenimiento al sistema y darle solución a cualquier problema que surja con la aplicación.

5. Requerimientos de Portabilidad

La herramienta propuesta podrá ser usada bajo cualquier Sistema operativo; el Servidor Web bajo sistema operativo Windows 2000 en adelante, y la Base de Datos bajo Windows Server 2000 en adelante. Los servidores Web y de Base de datos deben estar en PC distintas. Para su implementación se usará como herramienta de desarrollo Visual Studio .Net y Gestor de Base de Datos SQL Server 2000.

6. Requerimientos de Seguridad

Confidencialidad: la información manejada por el sistema debe estar protegida de acceso no autorizado y divulgación. Deberán estar encriptadas las matrices que se asociarán a las distintas tarjetas de autenticación de los clientes.

Integridad: la información manejada por el sistema debe ser objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

Disponibilidad: La aplicación deberá estar disponible en todo momento para aquellas personas con acceso a la información y los mecanismos utilizados para lograr la seguridad no deben ser un obstáculo a los usuarios para obtener los datos deseados en un momento dado.

7. Requerimientos de Software

En la computadora de los usuarios sólo se requiere de navegador para Internet o Intranet, bajo cualquier sistema operativo; en el Servidor Web, sistema operativo Windows 2000 en adelante, y en la Base de Datos, Windows Server 2000 en adelante. Para su implementación se usará como herramienta de desarrollo Visual Studio .Net y Gestor de Base de Datos SQL Server 2000.

8. Requerimientos de Hardware

En el cliente se requiere de una máquina de 128 MB de RAM, dos servidores Web con 1 Giga de RAM como mínimo, y el servidor de Base de Datos con 2 Giga de RAM y 80 GB de capacidad del disco duro, todas las computadoras implicadas en los servidores, la administración como las de los usuarios, deben estar conectadas a una red y tener al menos 100 MB de velocidad.

9. Restricciones en el diseño y la implementación

Se debe realizar una herramienta que permita dar respuesta en el menor tiempo posible, garantizando la calidad del sistema y la conexión con el Web Service, la base de dato debe ser desarrollada en SQL Server. Para garantizar el desarrollo de la aplicación se utilizará como guía el Proceso Unificado de Desarrollo, obteniendo como documentación los distintos artefactos propuestos. Se utilizará para realizar los modelos del sistema UML (Unified Modelling Language) y como herramienta de apoyo a este Lenguaje de Modelación se utiliza Rational Rose. Para la gestión de configuración se utilizará Microsoft Visual SourceSafe.

2.5 Definición de los casos de uso

2.5.1 Definición de los actores

Actores	Justificación
Operadora de Información	Es responsable de responder a las preguntas que realicen los clientes a través del Call Center.
Administrador de Información	Es responsable de actualizar la información que debe brindar las Operadoras de Información tanto de la agenda electrónica como de los procesos.

2.5.2 Listado de los casos de uso. (Ver Anexo 2)

2.5.3 Diagrama de casos de uso

Basándose en que un diagrama de casos de uso representa un conjunto de casos de uso para un sistema, los actores y la relación entre casos de uso y actores; se realizó el siguiente diagrama de casos de uso.

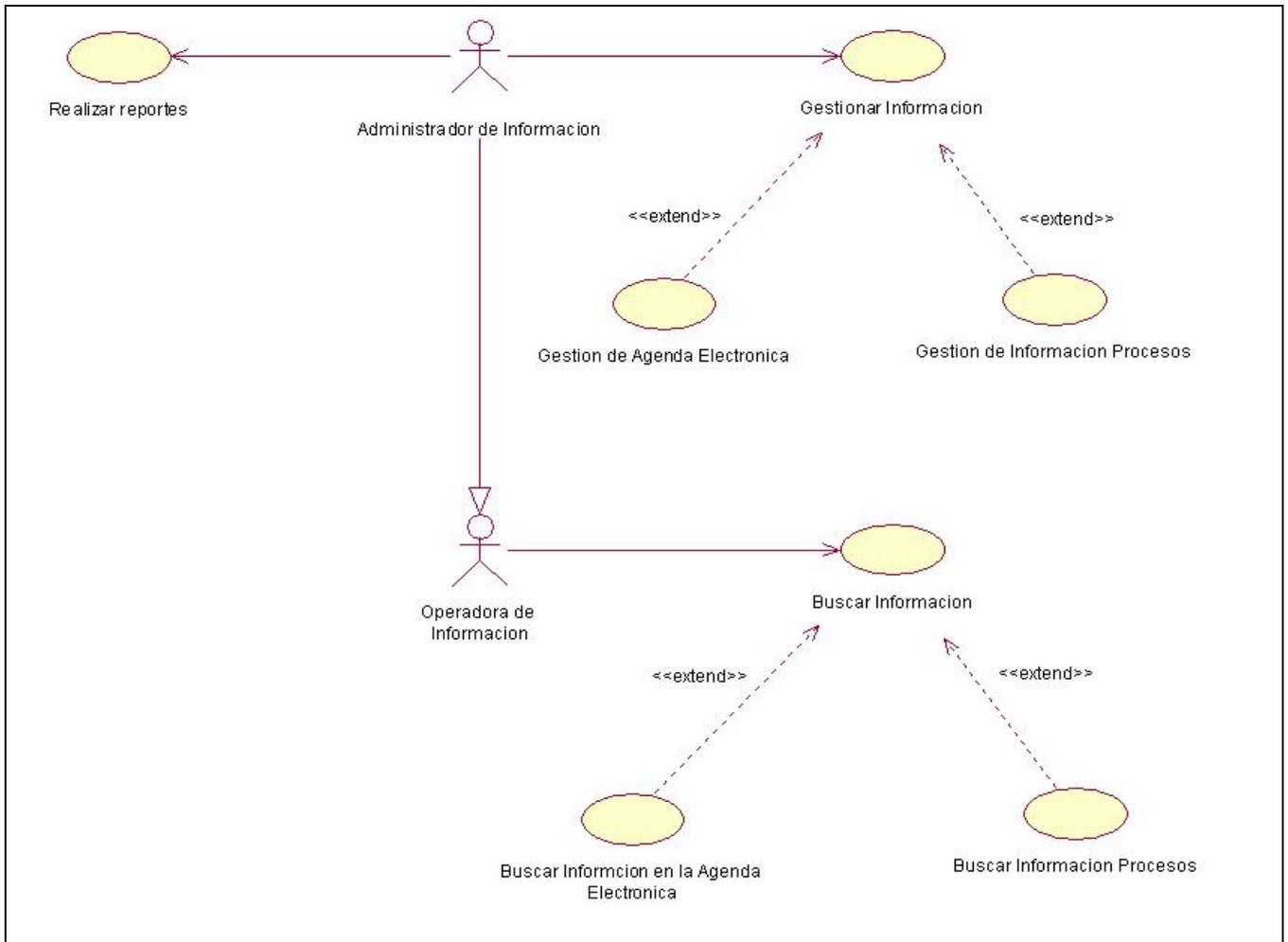


Figura 2. Diagrama de casos de usos del Sistema.

2.5.4 Casos de uso expandidos.

A través de la expansión de los casos de uso se describe paso a paso la secuencia eventos que los actores utilizan para completar un proceso a través del sistema.

2.5.4.1 CU_Gestionar Información

Caso de Uso:	CU_Gestionar Información
Actores:	Administrador de Información
Resumen:	El caso de uso inicia cuando el Administrador de Información necesita Gestionar información.

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el Administrador de Información selecciona la opción de Gestionar Información.	2. El Sistema muestra las opciones de Gestión de Agenda Electrónica o Gestión de Información de Procesos.
3. El Administrador de Información escoge el tipo de Gestión de Información que desea hacer. a) Si escoge Gestión de Agenda Electrónica, Ver Especificación del caso de uso Agenda Electrónica. b) Si escoge Gestión de Información de Procesos, Ver Especificación del caso de uso Información de Procesos.	

Flujos alternativos

Acción 3: Si el Administrador de Información decide no Gestionar Información.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 3	3.1 Se termina el Caso de Uso.

2.5.4.2 CU_Gestión de Agenda Electrónica

Caso de Uso:	CU_ Gestión de Agenda Electrónica.
Actores:	Administrador de Información.
Resumen:	El caso de uso se inicia cuando el Administrador de Información desea realizar algunas de las acciones. Ellas son eliminar, insertar o modificar en la agenda electrónica.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el Administrador	2. El Sistema muestra la lista de entidades de la

de Información selecciona la opción de Gestionar Información en la Agenda Electrónica.	agenda electrónica y las opciones para gestionar la agenda electrónica: insertar, modificar y eliminar información.
<p>3. El Administrador de Información selecciona una de las opciones relacionadas con la gestión de la información:</p> <p>c) Solicita insertar información en la agenda electrónica. Ver sección Insertar Información.</p> <p>d) Selecciona una entidad y solicita modificarle la información. Ver sección Modificar Información.</p> <p>e) Selecciona una entidad y solicita eliminarla. Ver sección Eliminar Información.</p>	
Sección Insertar Información	
	<p>4. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica, las cuales son:</p> <p>a) Nombre de la entidad (obligatorio).</p> <p>b) Dirección (obligatorio)</p> <p>c) Teléfono(s) (obligatorio)</p> <p>d) Fax (no obligatorio)</p> <p>e) Sitios Web relacionados (no obligatorio)</p> <p>f) Código anterior (no obligatorio).</p> <p>g) Código actual (obligatorio).</p> <p>Correos electrónicos (no obligatorio)</p>
5. El Administrador de Información llena los campos de la Agenda Electrónica.	
6. El Administrador de Información solicita guardar los campos de la Agenda Electrónica.	

	7. El Sistema valida que la información entrada fue la necesaria para la operación.
	8. El Sistema valida que la entidad a insertar sea única en la base de datos.
	9. El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"
10. El Administrador de Información acepta la operación de guardar la información en la Agenda Electrónica.	11. El Sistema guarda la nueva información.
	12. El Sistema muestra el mensaje "Se ha insertado satisfactoriamente la Entidad".
13. El Administrador de Información acepta el mensaje	14. Pasar a la acción 2.
Sección Modificar Información	
	15. El Sistema valida que se haya marcado una entidad.
	16. El Sistema le muestra la información de la entidad escogida. Los campos que pueden ser modificados son: a) Dirección (no obligatorio). b) Teléfono(s) (obligatorio) c) Fax (no obligatorio) d) Sitios Web relacionados (no obligatorio) e) Código anterior (no obligatorio) f) Código actual (obligatorio) g) Correos electrónicos (no obligatorio)
17.El Administrador de Información modifica los	

campos que desee de la entidad seleccionada.	
18.El Administrador de Información solicita guardar los cambios en la agenda electrónica.	19. El Sistema valida que la información entrada fue la necesaria para la operación.
	20. El Sistema valida que la entidad modificada sea única en la base de datos.
	21. El Sistema muestra el mensaje “¿Está seguro que desea modificar la Entidad?”
22. El Administrador de Información acepta guardar las modificaciones.	23. El Sistema guarda la nueva información.
	24. El Sistema muestra el mensaje”Se ha modificado la entidad satisfactoriamente”.
25. El Administrador de Información acepta el mensaje.	26. Pasar a la acción 2.
Sección Eliminar Información	
	27. El Sistema valida que se haya marcado una entidad.
	28. El Sistema muestra el mensaje “¿Esta seguro que desea eliminar la entidad seleccionada?”.
29. El Administrador de Información acepta eliminar la entidad seleccionada.	30. El Sistema guarda la entidad en un histórico.
	31. El Sistema elimina la entidad de la Agenda Electrónica.
	32. El Sistema muestra el mensaje”Se a eliminado satisfactoriamente la entidad”.
33. El Administrador de Información acepta el	

mensaje.	
----------	--

Flujos alternativos

Acción 2 F1: Si no existen entidades en la agenda electrónica.

Acción 2 F2: Si no se puede mostrar la lista de entidades porque la conexión con la base de datos está deshabilitada.

Acción 3: Si el Administrador de Información cancela la opción de Gestionar Agenda Electrónica.

Acción 6: Si el Administrador de Información cancela insertar la nueva entidad.

Acción 7: Si no es válida la información de la nueva entidad.

Acción 8 F1: Si la entidad a insertar ya existe.

Acción 8 F2: Si no se puede verificar si la entidad a insertar ya existe porque la conexión con la base de datos está deshabilitada.

Acción 10: Si el Administrador de Información cancela guardar la nueva entidad en la agenda electrónica.

Acción 11: Si el Sistema no puede guardar la nueva entidad porque la conexión con la Base de datos está deshabilitada.

Acción 15: Si el Administrador de Información no marca una entidad para modificarle la información.

Acción 16: Si no se pueden mostrar las informaciones de la entidad seleccionada porque la conexión a la base de datos está deshabilitada.

Acción 18: Si el Administrador de Información cancela hacer las modificaciones de las entidades que había escogido.

Acción 19: Si la información entrada no es válida.

Acción 20 F1: Si la entidad modificada es igual a una que ya existe.

Acción 20 F2: Si no se puede verificar si la entidad modificada es igual a una que ya existe porque la conexión con la base de datos está deshabilitada.

Acción 22: Si el Administrador de Información cancela la opción de confirmación de guardar las modificaciones.

Acción 23: Si el Sistema no puede guardar las nuevas modificaciones porque la conexión con la base de datos está deshabilitada.

Acción 27: Si el Administrador de Información no marca una entidad para eliminar.

Acción 29: Si el Administrador de Información cancela la opción de eliminar la entidad seleccionada.

Acción 30: Si el Sistema no puede guardar en el histórico porque la conexión con la base de datos está deshabilitada.

Acción 31: Si el Sistema no puede eliminar porque la conexión con la base de datos está deshabilitada.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 2 F1	2 F1.1 El Sistema muestra la lista de entidades de la Agenda Electrónica vacía y la opción de "Insertar" habilitada.
Acción 2 F2	2 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
2 F2.2 El Administrador de Información acepta	2 F2.3 Se termina el Caso de Uso.
Acción 3	3.1 Se termina el Caso de Uso.
Acción 6	6.1. Pasar a la Acción 2.
Acción 7	7.1 El Sistema marca con un asterisco los campos no válidos.
	7.2 El Sistema muestra el mensaje "Entrada de datos no válida".
7.3 El Administrador de Información acepta el mensaje.	7.4 Pasar a la Acción 4.
Acción 8 F1	8 F1.1 El Sistema muestra el mensaje "La entidad que se intenta insertar ya existe."
8 F1.2 El Administrador de Información acepta el	8 F1.3 Pasar a la Acción 4.

mensaje.	
Acción 8 F2	8 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
8 F2.2 El Administrador de Información acepta el mensaje.	8 F2.3 Pasar a la Acción 4.
Acción 10	10.1 Pasar a la Acción 4.
Acción 11	11.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
11.2 El Administrador de Información acepta el mensaje.	11.3 Pasar a la Acción 4.
Acción 15	15.1 El Sistema muestra deshabilitada la opción de modificar la entidad.
	15.2 Pasar a la Acción 2
Acción 16	16.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
16.2 El Administrador de Información.	16.2 Pasar a la Acción 2.
Acción 18	18.1 Pasar a la Acción 2.
Acción 19	19.1 El Sistema marca con un asterisco los campos no válidos.
	19.2 El Sistema muestra el mensaje "Entrada de datos no válida".
19.3 El Administrador de Información acepta el mensaje.	19.4 Pasar a la Acción 16.
Acción 20 F1	20 F1.1 El Sistema muestra el mensaje "La entidad a guardar ya existe."
20 F1.2 El Administrador de Información acepta el mensaje.	20 F1.3 Pasar a la Acción 16.
Acción 20 F2	20 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."

20 F2.2 El Administrador de Información acepta el mensaje.	20 F2.3 Pasar a la Acción 16.
Acción 22	22.1 Pasar a la Acción 16.
Acción 23	23.1 El Sistema muestra el mensaje “Error de conexión con la base de datos.”
23.2 El Administrador de Información acepta el mensaje.	23.3 Pasar a la Acción 16.
Acción 27	27.1 El Sistema muestra deshabilitada la opción de eliminar la entidad.
	27.2 Pasar a la Acción 2.
Acción 29	29.1 Pasar a la Acción 2.
Acción 30	30.1 El Sistema muestra el mensaje “Error de conexión con la base de datos.”
30.2 El Administrador de Información acepta el mensaje.	30.3 Pasar a la Acción 2.
Acción 31	31.3 El Sistema espera 1 min.
	31.4 Pasar a la Acción 31.

2.5.4.3 CU_Gestión de Información de Procesos

Caso de Uso:	CU_Gestión de Información de Procesos
Actores:	Administrador de Información
Resumen:	El caso de uso se inicia cuando el Administrador de Información desea realizar algunas de las acciones para la gestión de la Información de los Procesos. Ellas son eliminar, insertar o modificar Información de Procesos.
Flujo Normal de Eventos	
Sección Gestión de Información de Procesos	
Acción del Actor	Respuesta del Sistema
1. El Administrador de Información selecciona la opción de Gestionar Información de Procesos.	2. El Sistema muestra la lista de temas de la información de procesos y las opciones para

	gestionar la información: insertar, modificar y eliminar.
<p>3. El Administrador de Información selecciona una de las opciones relacionadas con la gestión de la información:</p> <p>a) Solicita insertar información de procesos. Ver sección Insertar Información.</p> <p>b) Selecciona un tema y solicita modificarle la información. Ver sección Modificar Información.</p> <p>c) Selecciona un tema y solicita eliminarlo. Ver sección Eliminar Información.</p>	
Sección Insertar Información de Procesos	
	<p>4. El Sistema muestra el formulario para entrar la información:</p> <p>a) Tema</p> <p>b) Texto correspondiente al proceso</p> <p>c) Lista de temas para seleccionar de cual será subtema.</p> <p>d) Palabra clave correspondiente.</p>
5. El Administrador de Información inserta los datos deseados en la información de procesos.	
6. El Administrador de Información solicita guardar la información de procesos.	7. El Sistema valida que la información a insertar es la necesaria para la operación.
	8. El Sistema valida que el tema a insertar sea único en la base de datos.
	9. El Sistema muestra el mensaje “¿Estás seguro que desea insertar esta Información de Procesos?”.
10.El Administrador de Información acepta la	11. El Sistema guarda la nueva información.

operación de guardar los datos en la información de procesos.	
	12. El Sistema muestra el mensaje “Se ha insertado satisfactoriamente la Información de Proceso.”
13. El Administrador de Información acepta el mensaje.	14. Se pasa a la acción 4
Sección Modificar Información de Procesos	
	15. El Sistema valida que se marque un tema a modificar.
	16. El Sistema muestra los datos del tema seleccionado.
17. El Administrador de Información modifica los datos que desea del tema seleccionado.	
18. El Administrador de Información solicita guardar la información de procesos.	19. El Sistema valida que la información modificada es la necesaria para la operación
	20. El Sistema valida que el tema modificado sea único en la base de datos.
	21. El Sistema muestra el mensaje “¿Está seguro que desea Modificar la Información de Proceso?”
22. El Administrador de Información acepta guardar las modificaciones.	23. El Sistema guarda la nueva información.
	24. El Sistema muestra el mensaje “Se ha modificado satisfactoriamente la Información de Proceso.”
25. El Administrador de Información acepta el	26. Se pasa a la acción 2

mensaje.	
Sección Eliminar Información de Procesos	
	27. El Sistema valida que se haya marcado un tema.
	28. El Sistema verifica si el tema seleccionado no tiene subtemas asociados.
	29. El Sistema muestra el mensaje "¿Estás seguro que desea eliminar la Información de Proceso?"
30. El Administrador de Información acepta eliminar.	31. El Sistema guarda la información eliminada en un histórico.
	32. El Sistema elimina el tema de la lista de temas.
	33. El Sistema muestra el mensaje: "Se ha eliminado satisfactoriamente la Información de Proceso."
34. El Administrador de Información acepta el mensaje.	

Flujos alternativos

Acción 2 F1: Si no existen temas en la información de procesos.

Acción 2 F2: Si no se puede mostrar la lista de temas porque la conexión con la base de datos está deshabilitada.

Acción 3: Si el Administrador de Información decide no seleccionar ninguna de las opciones de Gestión de la Información.

Acción 6: Si el Administrador de Información selecciona cancelar la opción de insertar temas en la Información de Procesos.

Acción 7: Si la información insertada no es válida o no es la necesaria para la operación.

Acción 8 F1: Si el tema a insertar ya existe.

Acción 8 F2: Si no se puede verificar si el tema a insertar ya existe porque la conexión con la base de datos está deshabilitada.

Acción 10: Si el Administrador de Información escoge la opción cancelar al ser solicitada la confirmación de guardar los cambios.

Acción 11: Si la información no pudo ser guardada porque la conexión con la base de datos está deshabilitada.

Acción 15: Si el Administrador de Información no marca un tema para modificarle la información.

Acción 16: Si la información del tema no se puede mostrar porque la conexión con la base de datos está deshabilitada.

Acción 18: Si el Administrador de Información selecciona cancelar la opción de modificar el tema seleccionado en la Información de Procesos.

Acción 19: Si la información modificada no es válida.

Acción 20 F1: Si el tema modificado es igual a uno que ya existe.

Acción 20 F2: Si no se puede verificar si el tema modificado es igual a una que ya existe porque la conexión con la base de datos está deshabilitada.

Acción 22: Si el Administrador de Información escoge la opción cancelar al ser solicitada la confirmación de guardar los cambios.

Acción 23: Si la información no pudo ser guardada porque la conexión con la base de datos está deshabilitada.

Acción 27: Si el Administrador de Información no marca tema para Eliminar.

Acción 28 F1: Si el Sistema no puede eliminar el tema ya que este contiene subtemas.

Acción 28 F2: Si el Sistema no puede verificar que el tema tiene subtemas porque la conexión con la base de datos está deshabilitada.

Acción 30: Si el Administrador de Información escoge la opción cancelar al ser solicitada la confirmación de eliminar el tema.

Acción 31: Si la información no pudo ser guardada en el histórico porque la conexión con la Base de Datos está deshabilitada.

Acción 32: Si la información no pudo ser eliminada porque la conexión con la base de datos está deshabilitada.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 2 F1	2 F1.1 El Sistema muestra la lista de entidades de la Agenda Electrónica vacía y la opción de "Insertar" habilitada.
Acción 2 F2	2 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
2 F2.2 El Administrador de Información acepta.	2 F2.3 Terminar el Caso de Uso.
Acción 3	3.1 Terminar el Caso de Uso.
Acción 6	6.1 Pasar a la Acción 2.
Acción 7	7.1 El Sistema marca con un asterisco los campos no válidos.
	7.2 El Sistema muestra el mensaje "Entrada de datos no válida".
7.2 EL Administrador de Información acepta el mensaje.	7.3 Pasar a la Acción 4.
Acción 8 F1	8 F1.1 El Sistema muestra el mensaje "La Información de Procesos que se intenta insertar ya existe".
8 F1.2 El Administrador de Información acepta.	8 F1.3 Pasar a la Acción 4.
Acción 8 F2	8 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
8 F2.2 El Administrador de Información acepta el mensaje.	8 F2.3 Pasar a la Acción 4.

Acción 10	10.1. Pasar a la Acción 4.
Acción 11	11.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
11.2 El Administrador de Información acepta el mensaje.	11.3 Pasar a la Acción 4.
Acción 15	15.1 El Sistema muestra deshabilitada la opción de modificar la Información de Procesos.
15.2 El Administrador de Información acepta el mensaje.	15.3 Pasar a la Acción 2.
Acción 16	16.1 El Sistema muestra el mensaje "Error de conexión con la base de datos."
16.2 El Administrador de Información acepta el mensaje.	16.3 Pasar a la Acción 2.
Acción 18	18.1. Pasar a la Acción 2.
Acción 19	19.1 El Sistema marca con un asterisco los campos no válidos.
	19.2 El Sistema muestra el mensaje "Entrada de datos no válida".
19.3 El Administrador de Información acepta el mensaje.	19.4 Pasar a la Acción 16
Acción 20 F1	20 F1.1. El Sistema muestra el mensaje "El tema a guardar ya existe".
20 F1.2 El Administrador de Información acepta el mensaje.	20 F1.3 Pasar a la Acción 16
Acción 20 F2	20 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos".
20 F2.3 El Administrador de Información acepta el mensaje.	20 F2.3 Pasar a la Acción 16
Acción 22	22.1. Pasar a la Acción 16

Acción 23	23.1. El Sistema muestra el mensaje "Error de conexión con la base de datos".
23.2 El Administrador de Información acepta el mensaje.	23.3 Pasar a la Acción 16
Acción 27	27.1 El Sistema muestra deshabilitada la opción de eliminar la Información de Procesos.
27.2 El Administrador de Información acepta el mensaje.	27.3 Pasar a la Acción 2.
Acción 28 F1	28 F1.1 El Sistema muestra el mensaje "No se puede eliminar temas con subtemas asociados".
28 F1.2 El Administrador de Información acepta el mensaje.	28 F1.3 Pasar a la Acción 2.
Acción 28 F2	28 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos".
28 F2.2 El Administrador de Información acepta el mensaje.	28 F2.3 Pasar a la Acción 2.
Acción 30	30.1. Pasar a la Acción 2.
Acción 31	31.1 El Sistema muestra el mensaje "Error de conexión con la base de datos".
31.2 El Administrador de Información acepta el mensaje.	31.3 Pasar a la Acción 2.
Acción 32	32.1. El Sistema espera 1 min.
	32.3. Pasar a la acción 32

2.5.4.4 CU_Buscar Información

Caso de Uso:	CU_Buscar Información.
Actores:	Operadora de Información.
Resumen:	El caso de uso inicia cuando la Operadora de Información desea buscar información que puede ser en la Agenda Electrónica o en la Información de

	Procesos.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando la Operadora de Información selecciona la opción de Buscar la Información.	2. El Sistema muestra las opciones de Buscar en la Agenda Electrónica o Buscar Información de Procesos.
3. La Operadora de Información escoge el tipo de búsqueda de la Información que desea hacer: a) Si escoge Buscar en la Agenda Electrónica, Ver Especificación del caso de uso Buscar en Agenda Electrónica. b) Si escoge Buscar Información de Procesos, Ver Especificación del caso de uso Buscar Información de Procesos.	

Flujos alternativos

Acción 3: Si la Operadora de Información decide no buscar información.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 3	3.1 Se termina el Caso de Uso.

2.5.4.5 CU_Buscar en la Agenda Electrónica

Caso de Uso:	CU_Buscar en la Agenda Electrónica.
Actores:	Operadora de Información.
Resumen:	El caso de uso se inicia cuando la Operadora de Información desea hacer búsquedas de Información en la Agenda Electrónica.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

<p>1. El caso de uso comienza cuando la Operadora de Información selecciona la opción de Buscar Información en la Agenda Electrónica.</p>	<p>2. El Sistema muestra la forma de buscar las entidades en la Agenda Electrónica, puede ser por:</p> <ul style="list-style-type: none"> a) Nombre de la entidad. b) Dirección. c) Teléfono(s). d) Fax. e) Sitios Web relacionados. f) Código anterior. g) Código actual. h) Correos electrónicos.
<p>3. La Operadora de Información puede introducir: Nombre de la entidad, dirección, teléfono(s), fax, Sitios Web relacionados, código anterior de la sucursal o el código actual y los correos electrónicos que serán las condiciones con que quiere que cumplan las entidades que busca en la Agenda Electrónica.</p>	
<p>4. La Operadora de Información solicita que se muestre la entidad con las características deseadas.</p>	<p>5. El Sistema muestra la entidad encontrada con sus datos según los criterios seleccionados.</p>
<p>6. La Operadora de Información consulta los datos y marca la información específica solicitada por el cliente.</p>	
<p>7. La Operadora de Información acepta la información específica solicitada por el cliente.</p>	<p>8. El Sistema registra automáticamente los datos consultados para realizar nuevos reportes.</p>

Flujos alternativos

Acción 3: Si la Operadora de Información no introduce ningún criterio de búsqueda.

Acción 4: Si la Operadora de Información cancela la opción de buscar las entidades.

Acción 5 F1: Si el Sistema no encuentra entidades que cumplan con el criterio.

Acción 5 F2: Si no se puede mostrar la entidad con sus datos según los criterios seleccionados porque la conexión con la base de datos está deshabilitada.

Acción 6: Si la Operadora de Información cancela la opción de no marcar ninguna información específica solicitada por el cliente.

Acción 7: Si la Operadora de Información cancela la información específica solicitada por el cliente.

Acción 8: Si la información no ha sido registrada correctamente porque la conexión con la base de datos está deshabilitada.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 3	3.1 El Sistema muestra la lista de entidades de la Agenda Electrónica.
Acción 4	4.1 Se termina el Caso de Uso.
Acción 5 F1	5 F1.1 El Sistema muestra el mensaje "No se ha encontrado ninguna información asociada a los criterios seleccionados".
5 F1.2 La Operadora de Información acepta el mensaje.	5 F1. 3 Pasar a la Acción 2.
Acción 5 F2	5 F2.1 El Sistema muestra el mensaje "Error de conexión con la base de datos".
5 F2.2 La Operadora de Información acepta el mensaje.	5 F2.3 Se termina el Caso de Uso.
Acción 6	6.3 Pasar a la Acción 5.
Acción 7	7.1 Pasar a la Acción 2.
Acción 8	8.1 Se espera 1 minuto.
	8.2 Pasar a la Acción 7.

2.5.4.6 CU_Buscar Información de Procesos

Caso de Uso:	CU_Buscar Información de Procesos.	
Actores:	Operadora de Información.	
Resumen:	El caso de uso inicia cuando la Operadora de Información desea buscar Información de Procesos.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando la Operadora de Información selecciona la opción de Buscar la Información de Procesos.	2. El sistema le muestra las formas de buscar los temas en la Información de Procesos: por temas o por palabra clave.	
3. La Operadora de Información escoge el tipo de búsqueda. a) Escoge búsqueda por temas. Ver sección Buscar por temas. b) Escoge búsqueda por palabras clave. Ver sección Buscar por palabra clave.		
Sección Buscar por temas		
	4. El sistema le muestra la lista de temas existentes.	
5. La Operadora de Información escoge de la lista el tema que desea.		
6. La Operadora de Información acepta ver la información del tema escogido.	7. El sistema muestra los datos que contiene el tema escogido.	

8. La Operadora de Información consulta la información del tema seleccionado.	9. El sistema registra automáticamente los datos consultados para realizar futuros reportes.
Sección Buscar por palabra clave	
	10. El sistema muestra la opción de insertar la palabra clave.
11 La Operadora de Información inserta la palabra clave.	
12. La Operadora de Información acepta mostrar las informaciones asociadas a la palabra clave.	13. El sistema muestra la lista de temas.
14. La Operadora de Información escoge uno de los temas que nos muestra el Sistema acorde a sus necesidades.	
15. La Operadora de Información acepta ver la información del tema escogido.	16. El sistema muestra la información del tema.
17. La Operadora de Información consulta la información del tema seleccionado.	18. El sistema registra automáticamente los datos consultados para realizar futuros reportes.

Flujos alternativos.

Acción 3: Si la Operadora de Información cancela la opción de buscar Información de Procesos.

Acción 4 F1: Si no existe ningún tema.

Acción 4 F2: Si la información no pudo ser mostrada porque la conexión con la base de datos está deshabilitada.

Acción 6: La Operadora de Información cancela ver la información que contiene el tema escogido.

Acción 7 F1: Si el tema escogido contiene subtemas.

Acción 7 F2: Si la información no pudo ser mostrada porque la conexión con la base de datos está deshabilitada.

Acción 9: Si la información no pudo ser registrada porque la conexión con la base de datos está deshabilitada.

Acción 12: Si la Operadora de Información cancela ver los temas asociados a la palabra clave.

Acción 13 F1: Si no existe información que corresponda con la palabra clave insertada.

Acción 13 F2: Si la lista no pudo ser mostrada porque la conexión con la base de datos está deshabilitada.

Acción 15: Si la Operadora de Información cancela ver la información del tema seleccionado.

Acción 16: Si la información no pudo ser mostrada porque la conexión con la base de datos está deshabilitada.

Acción 18: Si la información no pudo ser registrada porque la conexión con la base de datos está deshabilitada.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 3	3.1 Se termina el caso de uso
Acción 4 F1	4 F1.1 El Sistema muestra el mensaje “No existen temas en la Información de Procesos”.
4 F1.2 La Operadora de Información acepta el mensaje.	4 F1.3 Se termina el caso de uso.
Acción 4 F2	4 F2.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
4 F2.2 La Operadora de Información acepta el mensaje.	4 F2.3 Se termina el caso de uso.
Acción 6	6.1 Se pasa para la Acción 2.
Acción 7 F1	7 F1.1 El Sistema le muestra la información del tema y la lista de subtemas existentes.
	7 F1.2 Se pasa a la Acción 5.
Acción 7 F2	7 F2.1 El Sistema muestra el mensaje “Error de

	conexión con la base de datos”.
7 F2.2 La Operadora de Información acepta el mensaje.	7 F2.3 Se termina el caso de uso.
Acción 9	9.1 Se espera 1 minuto.
	9.2 Se pasa a la Acción 9.
Acción 12	12.1 Se pasa a la Acción 2.
Acción 13 F1	13 F1.1 El Sistema muestra el mensaje “No hay datos que cumplan con los criterios de búsqueda”.
13 F1.2 La Operadora de Información acepta el mensaje.	13 F1.3 Se pasa a la Acción 10
Acción 13 F2	13 F2.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
13 F2.2 La Operadora de Información acepta el mensaje.	13 F2.3 Se termina el caso de uso.
Acción 15	15.1 Se pasa a la Acción 10
Acción 16	16.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
16.2 La Operadora de Información acepta el mensaje.	16.3 Se termina el caso de uso.
Acción 18	18.1 Se espera 1 minuto.
	18.2 Se pasa a la Acción 18.

2.5.4.7 CU_Realizar Reportes

Caso de Uso:	CU_Realizar Reportes
Actores:	Administrador de Información
Resumen:	El caso de uso se inicia cuando el Administrador de Información desea realizar reportes de la información consultada a petición de los clientes.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

1. El caso de uso comienza cuando el Administrador de Información selecciona la opción de Realizar Reportes.	2. El Sistema muestra las opciones para realizar reportes: Trabajo de las operadoras e Informe de las consultas realizadas a solicitud del cliente.
3. El Administrador de Información selecciona: a) Trabajo de las Operadoras, Ver Sección Trabajo de las Operadoras. b) Sección Informes de las Consultas, Ver Sección Informes de las Consultas.	
Sección Trabajo de las Operadoras	
	4. El Sistema muestra un formulario para seleccionar un rango de fecha y las opciones de los tipos de reportes. a) Personal Resumen (Lista de operadora y cantidad de veces que accedió por cada tipo de información) b) Personal Detallado (cantidad de veces que se accede por cada tipo de información para una operadora, especificando en el caso de la agenda electrónica el dato consultado). c) General Resumen (cantidad de información accedida por cada tipo de información) d) General Detallado (cantidad de Información accedida por cada tipo de información, en el caso de la agenda electrónica se ponen las cantidades por cada dato de la entidad).
5. El Administrador de Información selecciona el rango de fecha para el cual quiere hacer el reporte	

<p>y el Tipo de Reporte para realizar el reporte:</p> <p>a) Personal Detallado, Ver Sección Personal Detallado.</p> <p>b) Personal Resumen, Ver Sección Personal Resumen.</p> <p>c) General Detallado, Ver Sección General Detallado.</p> <p>d) General Resumen, Ver Sección General Resumen.</p>	
Sección Personal Detallado	
	<p>6. El Sistema muestra un formulario con el listado de las Operadoras y Tipo de Información.</p> <p>a) Agenda Electrónica</p> <p>b) Información de Procesos</p>
<p>7. El Administrador de Información selecciona la Operadora y el tipo de Información.</p>	
<p>8. El Administrador de Información selecciona la opción "Aceptar".</p>	<p>9. El Sistema busca la cantidad de consultas realizadas en la agenda electrónica por cada tipo de dato y por día.</p>
	<p>10. El Sistema muestra el reporte brindando la opción de imprimirlo.</p>
<p>11. El Administrador de Información acepta la opción "Imprimir reporte".</p>	<p>12. El Sistema imprime el reporte.</p>
Sección Personal Resumen	
	<p>13. El Sistema muestra un formulario con el Listado de las Operadoras.</p>
<p>14. El Administrador de Información selecciona la Operadora.</p>	

15. El Administrador de Información selecciona la opción "Aceptar".	16. El Sistema busca la cantidad de consultas realizadas por cada tipo de información y por día.
	17. El Sistema muestra el reporte brindando la opción de imprimirlo.
18. El Administrador de Información acepta la opción "Imprimir reporte".	19. El Sistema imprime el reporte.
Sección General Resumen	
	20. El Sistema busca la cantidad de consultas realizadas por cada tipo de información y por día.
	21. El Sistema muestra el reporte brindando la opción de imprimirlo.
22. El Administrador de Información acepta la opción "Imprimir reporte".	23. El Sistema imprime el reporte.
Sección General Detallado	
	24. El Sistema muestra un formulario donde escoge el Tipo de Información. a) Agenda Electrónica b) Información de Procesos
25. El Administrador de Información selecciona el tipo de información.	
26. El Administrador de Información selecciona la opción "Aceptar".	27. El Sistema busca la cantidad de consultas realizadas en la agenda electrónica por cada tipo de dato y por día.
	28. El Sistema muestra el reporte brindando la opción de imprimirlo.
29. El Administrador de Información acepta la opción "Imprimir reporte".	30. El Sistema imprime el reporte.
Sección Informe de las Consultas	

	31. El Sistema muestra un formulario para seleccionar el rango de fecha, el orden ascendente o descendente y las opciones de los tipos de reportes: a) Consultas a Agenda Electrónica. b) Consultas a Información de los procesos.
32. El Administrador de Información selecciona el rango de fecha, el orden y el tipo de información.	
33. El Administrador de Información selecciona la opción "Aceptar".	34. El Sistema busca la cantidad de consultas realizadas en la agenda electrónica por cada tipo de dato y por día y lo organiza según el orden seleccionado.
	35. El Sistema muestra el reporte brindando la opción de imprimirlo.
36. El Administrador de Información acepta la opción "Imprimir reporte".	37. El Sistema imprime el reporte.

Flujos alternativos

Acción 3: Si el Administrador de Información cancela la acción de realizar reportes.

Acción 5: Si el Administrador de Información cancela la acción de realizar reporte del trabajo de las operadoras.

Acción 6: Si no se puede mostrar el listado de las operadoras por estar la conexión con la Base de datos deshabilitada.

Acción 8: Si el Administrador de Información cancela la acción de realizar reporte personal detallado del trabajo de las operadoras.

Acción 9 F1: Si Administrador de Información selecciona Información de procesos.

Acción 9 F2: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 9 F3: Si el Sistema no puede buscar la información porque la conexión con la base de datos está deshabilitada.

Acción 11: Si el Administrador de Información cancela la acción de imprimir el reporte.

Acción 12: Si ocurre un error al imprimir la información.

Acción 13: Si no se puede mostrar el listado de las operadoras por estar la conexión con la base de datos deshabilitada.

Acción 15: Si el Administrador de Información cancela la acción de realizar el reporte personal resumido del trabajo de las operadoras.

Acción 16 F1: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 16 F2: Si el Sistema no puede buscar información porque la conexión con la base de datos está deshabilitada.

Acción 18: Si el Administrador de Información cancela la acción de imprimir el reporte.

Acción 19: Si ocurre un error al imprimir la información.

Acción 20 F1: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 20 F2: Si el Sistema no puede buscar información porque la conexión con la base de datos está deshabilitada.

Acción 22: Si el Administrador de Información cancela la acción de imprimir el reporte.

Acción 23: Si ocurre un error al imprimir la información.

Acción 26: Si el Administrador de Información cancela la acción de realizar el reporte general detallado del trabajo de las operadoras.

Acción 27 F1: Si el Administrador de Información selecciona Información de procesos.

Acción 27 F2: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 27 F3: Si el Sistema no puede buscar la información porque la conexión con la base de datos está deshabilitada.

Acción 29: Si el Administrador de Información cancela la acción de imprimir el reporte.

Acción 30: Si ocurre un error al imprimir la información.

Acción 33: Si el Administrador de Información cancela la acción de realizar el reporte de informe de las consultas.

Acción 34 F1: Si el Administrador de Información selecciona Información de procesos.

Acción 34 F2: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 34 F3: Si el Sistema no puede buscar la información porque la conexión con la base de datos está deshabilitada.

Acción 36: Si el Administrador de Información cancela la acción de imprimir el reporte.

Acción 37: Si ocurre un error al imprimir la información.

Flujos alternativos de Flujos alternativos

Acción 9 F1.1 F1: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 9 F1.1 F2: Si el Sistema no puede buscar información porque la conexión con la base de datos está deshabilitada.

Acción 27 F1.1 F1: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 27 F1.1 F2: Si el Sistema no puede buscar información porque la conexión con la base de datos está deshabilitada.

Acción 34 F1.1 F1: Si el Sistema no encuentra información que cumpla con esos criterios.

Acción 34 F1.1 F2: Si el Sistema no puede buscar información porque la conexión con la base de datos está deshabilitada.

Flujos Alternativos	
Acción del Actor	Respuesta del Sistema
Acción 3	3.1 Terminar el caso de uso.
Acción 5	5.1 Pasar Acción 2.

Acción 6	6.1 El Sistema muestra el mensaje:” Error de conexión con la base de datos”.
6.2 El Administrador de Información acepta el mensaje.	6.3 Pasar a la acción 4.
Acción 8	8.1 Pasar Acción 4.
Acción 9 F1	9 F1.1 El Sistema busca la cantidad de consultas realizadas en la información de procesos por cada tema.
	9 F1.2 Pasar a la acción 10.
Acción 9 F2	9 F2.1 El Sistema muestra el mensaje “No existe información que cumpla con esos criterios”.
9 F2.2 El Administrador de Información acepta el mensaje.	9 F2.3 Pasar a la Acción 6.
Acción 9 F3	9 F3.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
9 F3.2 El Administrador de Información acepta el mensaje.	9 F3.3 Pasar a la Acción 6.
Acción 11	11.1 Pasar a la Acción 6.
Acción 12	12.1 El Sistema muestra el mensaje “Ha ocurrido un error al imprimir”.
12.2 El Administrador de Información acepta el mensaje.	12.3 Pasar a la acción 10.
Acción 13	13.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
13.2 El Administrador de Información acepta el mensaje.	13.3 Pasar a la acción 4.
Acción 15	15.1 Pasar Acción 4.
Acción 16 F1	16 F1.1 El Sistema muestra el mensaje “No

	existe información que cumpla con esos criterios”.
F1.2 El Administrador de Información acepta el mensaje.	16 F1.3 Pasar a la Acción 13.
Acción 16 F2	16 F2.1 El Sistema muestra el mensaje: “Error de conexión con la base de datos”.
16 F2.2 El Administrador de Información acepta el mensaje.	16 F2.3 Pasar a la Acción 13.
Acción 18	18.1 Pasar a la Acción 13.
Acción 19	19.1 El Sistema muestra el mensaje “Ha ocurrido un error al imprimir”.
19.2 El Administrador de Información acepta el mensaje.	19.3 Pasar a la acción 17.
Acción 20 F1	20 F1.1 El Sistema muestra el mensaje “No existe información que cumpla con esos criterios”.
20 F1.2 El Administrador de Información acepta el mensaje.	20 F1.3 Pasar a la Acción 4.
Acción 20 F2	20 F2.1 El Sistema muestra el mensaje “Error de conexión con la base de datos”.
20 F2.2 El Administrador de Información acepta el mensaje.	20 F2.3 Pasar a la Acción 4.
Acción 22	22.1 Pasar a la Acción 4.
Acción 23	23.1 El Sistema muestra el mensaje “Ha ocurrido un error al imprimir”.
23.2 El Administrador de Información acepta el mensaje.	23.3 Pasar a la acción 21.
Acción 26	26.1 Pasar Acción 4.
Acción 27 F1	27 F1.1 El Sistema busca la cantidad de

	consultas realizadas en la información de procesos por cada tema.
	27 F1.2 Pasar a la acción 28.
Acción 27 F2	27 F2.1 El Sistema muestra el mensaje: "No existe información que cumpla con esos criterios".
27 F2.2 El Administrador de Información acepta el mensaje.	27 F2.3 Pasar a la Acción 24.
Acción 27 F3	27 F3.1 El Sistema muestra el mensaje: "Error de conexión con la base de datos".
27 F3.2 El Administrador de Información acepta el mensaje.	27 F3.3 Pasar a la Acción 24.
27 F1.1 F2.2 El Administrador de Información acepta el mensaje.	27 F1.1 F2.3 Pasar a la Acción 24.
Acción 29	29.1 Pasar a la Acción 24.
Acción 30	30.1 El Sistema muestra el mensaje "Ha ocurrido un error al imprimir".
30.2 El Administrador de Información acepta el mensaje.	30.3 Pasar a la acción 28.
Acción 33	26.1 Pasar Acción 2.
Acción 34 F1	34 F1.1 El Sistema busca la cantidad de consultas realizadas en la información de procesos por cada tema.
	34 F1.2 Pasar a la acción 35.
Acción 34 F2	34 F2.1 El Sistema muestra el mensaje: "No existe información que cumpla con esos criterios".
34 F2.2 El Administrador de Información acepta el mensaje.	34 F2.3 Pasar a la Acción 31.

Acción 34 F3	34 F3.1 El Sistema muestra el mensaje: "Error de conexión con la base de datos".
34 F3.2 El Administrador de Información acepta el mensaje.	34 F3.3 Pasar a la Acción 31.
Acción 36	36.1 Pasar a la Acción 31.
Acción 37	37.1 El Sistema muestra el mensaje "Ha ocurrido un error al imprimir".
37.2 El Administrador de Información acepta el mensaje.	37.3 Pasar a la acción 35.
Flujos alternativos de flujos alternativos	
Acción 9 F1.1 F1	9 F1.1 F1.1 El Sistema muestra el mensaje "No existe información que cumpla con esos criterios".
9 F1.1 F1.2 El Administrador de Información acepta el mensaje.	9 F1.1 F1.3 Pasar a la Acción 6.
Acción 9 F1.1 F2	9 F1.1 F2.1 El Sistema muestra el mensaje "La conexión con la base de datos está deshabilitada".
9 F1.1 F2.2 El Administrador de Información acepta el mensaje.	9 F1.1 F2.3 Pasar a la Acción 6.
Acción 27 F1.1 F1	27 F1.1 F1.1 El Sistema muestra el mensaje: "No existe información que cumpla con esos criterios".
27 F1.1 F1.2 El Administrador de Información acepta el mensaje.	27 F1.1 F1.3 Pasar a la Acción 24.
Acción 27 F1.1 F2	27 F1.1 F2.1 El Sistema muestra el mensaje: "La conexión con la base de datos está deshabilitada".
Acción 34 F1.1 F1	34 F1.1 F1.1 El Sistema muestra el mensaje: "No

	existe información que cumpla con esos criterios”.
34 F1.1 F1.2 El Administrador de Información acepta el mensaje.	34 F1.1 F1.3 Pasar a la Acción 31.
Acción 34 F1.1 F2	34 F1.1 F2.1 El Sistema muestra el mensaje: “La conexión con la base de datos está deshabilitada”.
34 F1.1 F2.2 El Administrador de Información acepta el mensaje.	34 F1.1 F2.3 Pasar a la Acción 31.

2.6 Conclusiones

Al plantear determinados problemas prácticos existentes en la gestión de la información de los servicios bancarios, se pone de manifiesto la necesidad de crear elementos que garanticen la gestión de información con mayor rapidez y certeza para la toma de decisiones. De acuerdo al problema planteado se realizó un modelo del dominio donde se capturaron los tipos más importantes de objetos que existen y los eventos que suceden en el entorno donde está enmarcado el problema del sistema.

En este capítulo se comenzó a desarrollar la propuesta de solución, obteniéndose a partir del análisis de los procesos del dominio, un listado con los requerimientos funcionales y no funcionales para el desarrollo de nuestro sistema, además quedaron representados gráficamente los procesos y su interacción con los actores mediante los diagramas de casos de uso, y finalmente se describieron todas las acciones de los actores del sistema con los casos de uso. Con lo ya establecido se puede empezar la construcción del sistema, cumpliendo todos los requerimientos y las funciones que han sido consideradas necesarias en este capítulo.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

En este capítulo se determinan las clases necesarias para llevar a cabo las funcionalidades del sistema. Precisamente se abordará el flujo de trabajo de análisis y diseño.

En este capítulo se pretende abordar el análisis y diseño de este sistema de modo que no abordaremos la capa de acceso a datos, sino que trabajaremos con la lógica de presentación y la lógica de negocio.

3.2 Análisis

En el modelo de análisis se estructuran los requisitos de manera que nos facilite su comprensión, su preparación, su modificación y en general su mantenimiento. Consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales.

3.2.1 Diagrama de clases del análisis

Los diagramas de clases de análisis son diagramas de estructura estática que muestran las clases del sistema y sus interrelaciones.

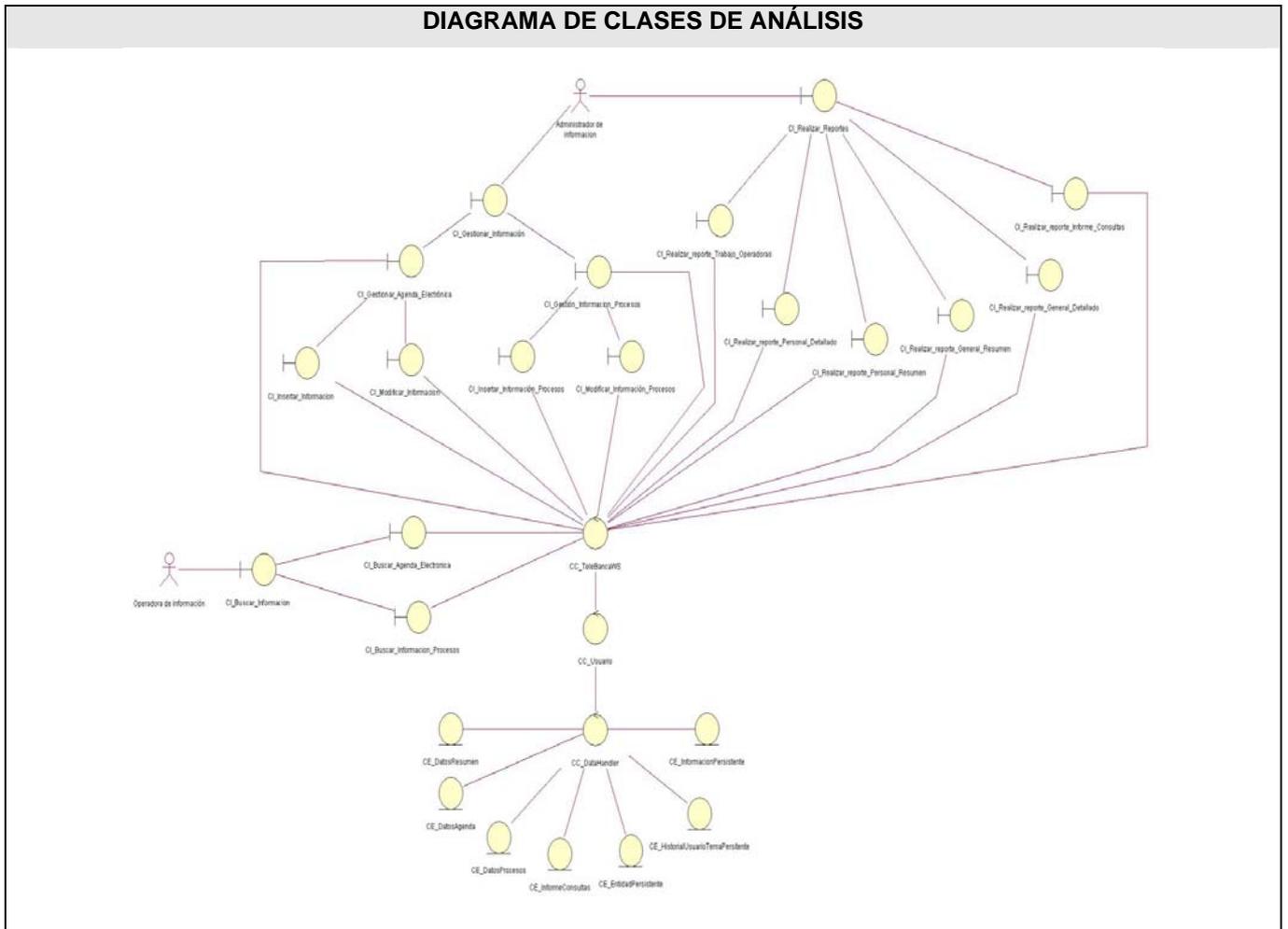


Figura 3. Diagrama de clases del análisis.

3.3 Diseño

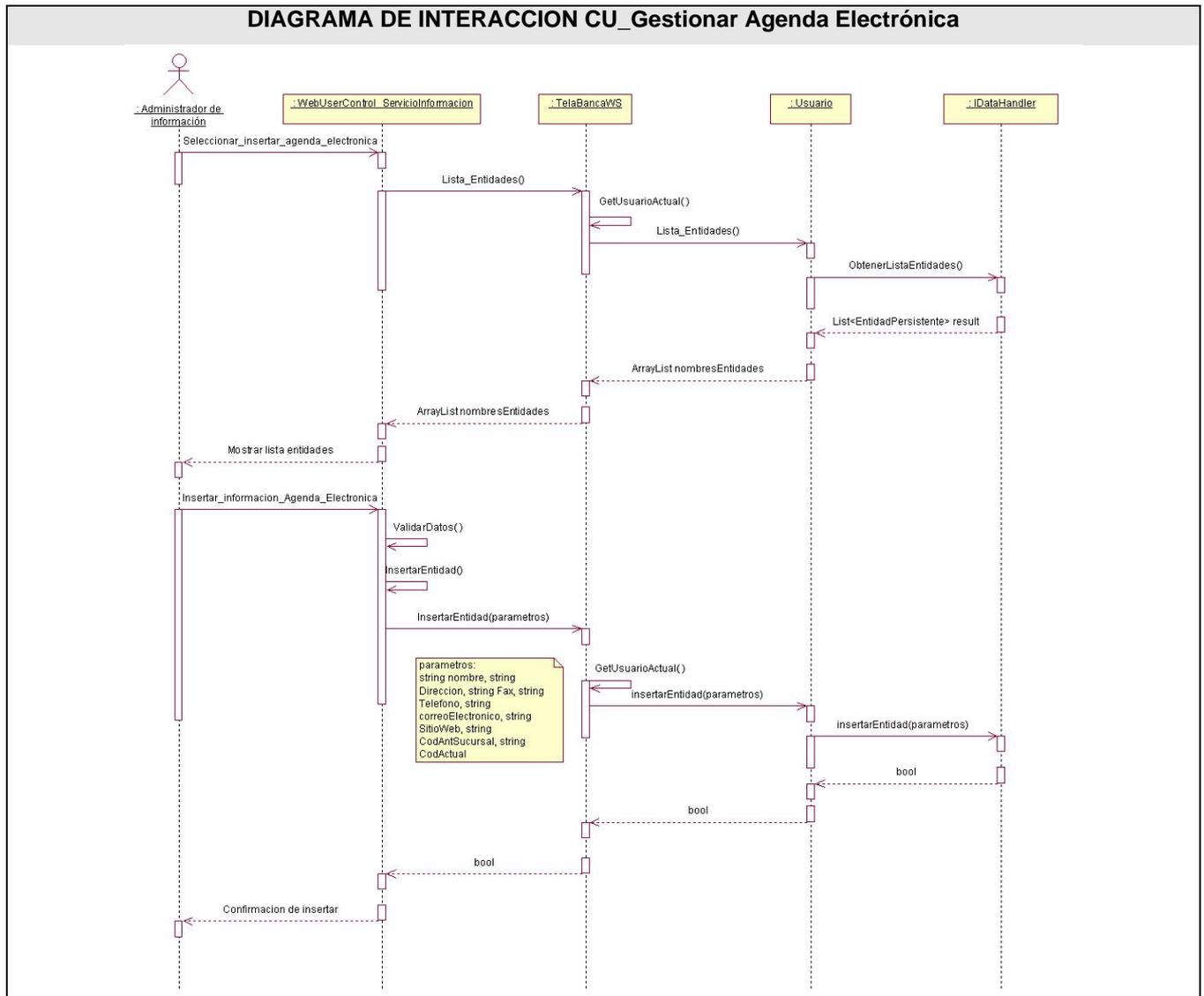
El diseño tiene el propósito de formular los modelos que se centran en los requisitos no funcionales y en el dominio de la solución y que prepara para la implementación y prueba del sistema. Pretende crear un plano del modelo de implementación.

3.3.1 Diagramas de interacción

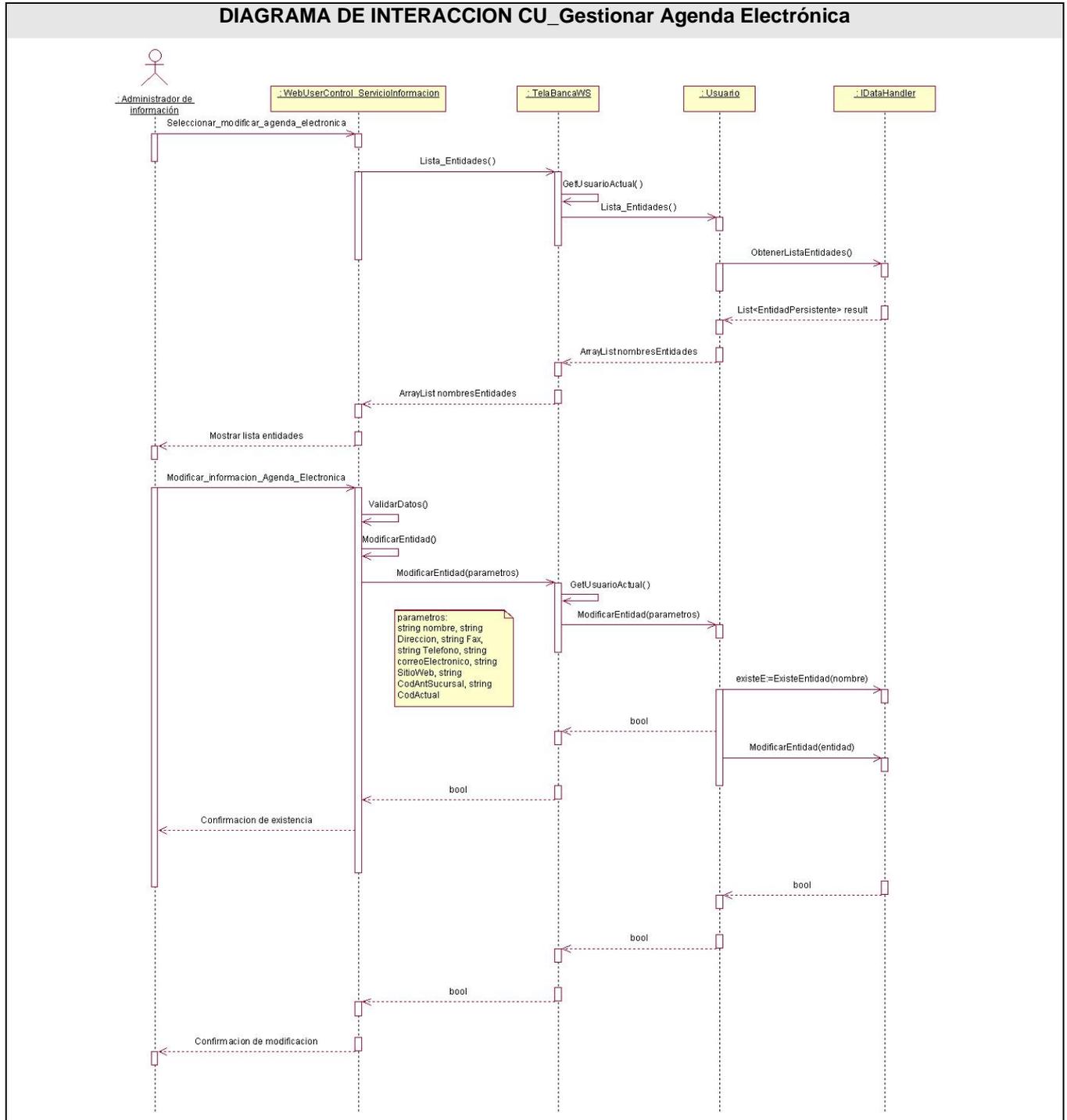
Los diagramas de interacción, representan la forma por la cual un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde

se obtienen las responsabilidades claramente. En este caso se mostrarán algunos de los diagramas de uno de los casos de usos críticos, el resto de los diagramas de interacción se encuentran en el (Anexo 3).

3.3.3.1 Diagrama de interacción CU_Gestionar Agenda Electrónica, sección insertar

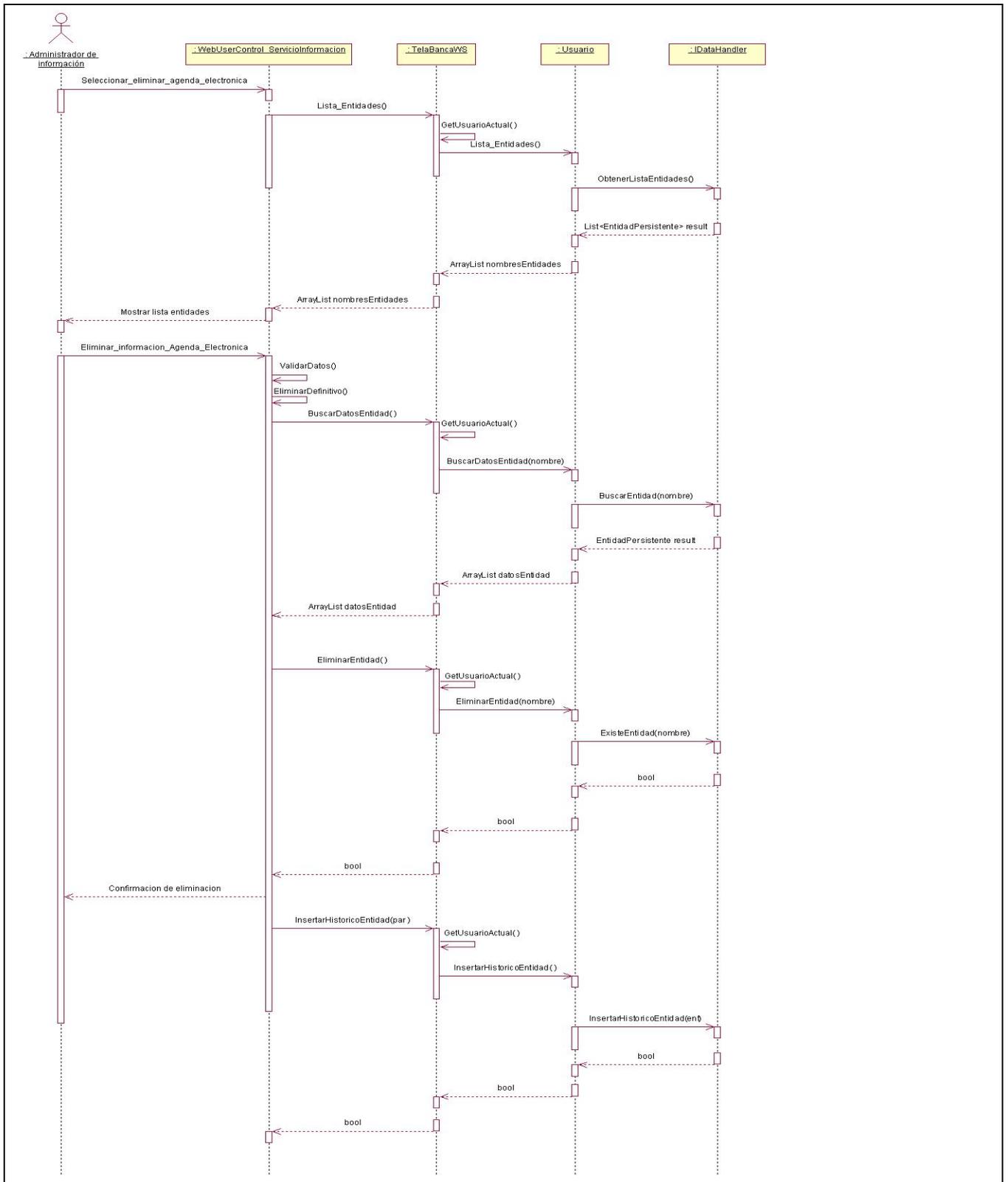


3.3.3.2 Diagrama de interacción CU_Gestionar Agenda Electrónica, sección modificar



3.3.3.3 Diagrama de interacción CU_Gestionar Agenda Electrónica, sección eliminar

DIAGRAMA DE INTERACCION CU_Gestionar Agenda Electrónica



3.3.2 Diagrama de clases del diseño

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema.

3.3.2.1 Diagrama de clases del diseño de la capa de lógica de la presentación

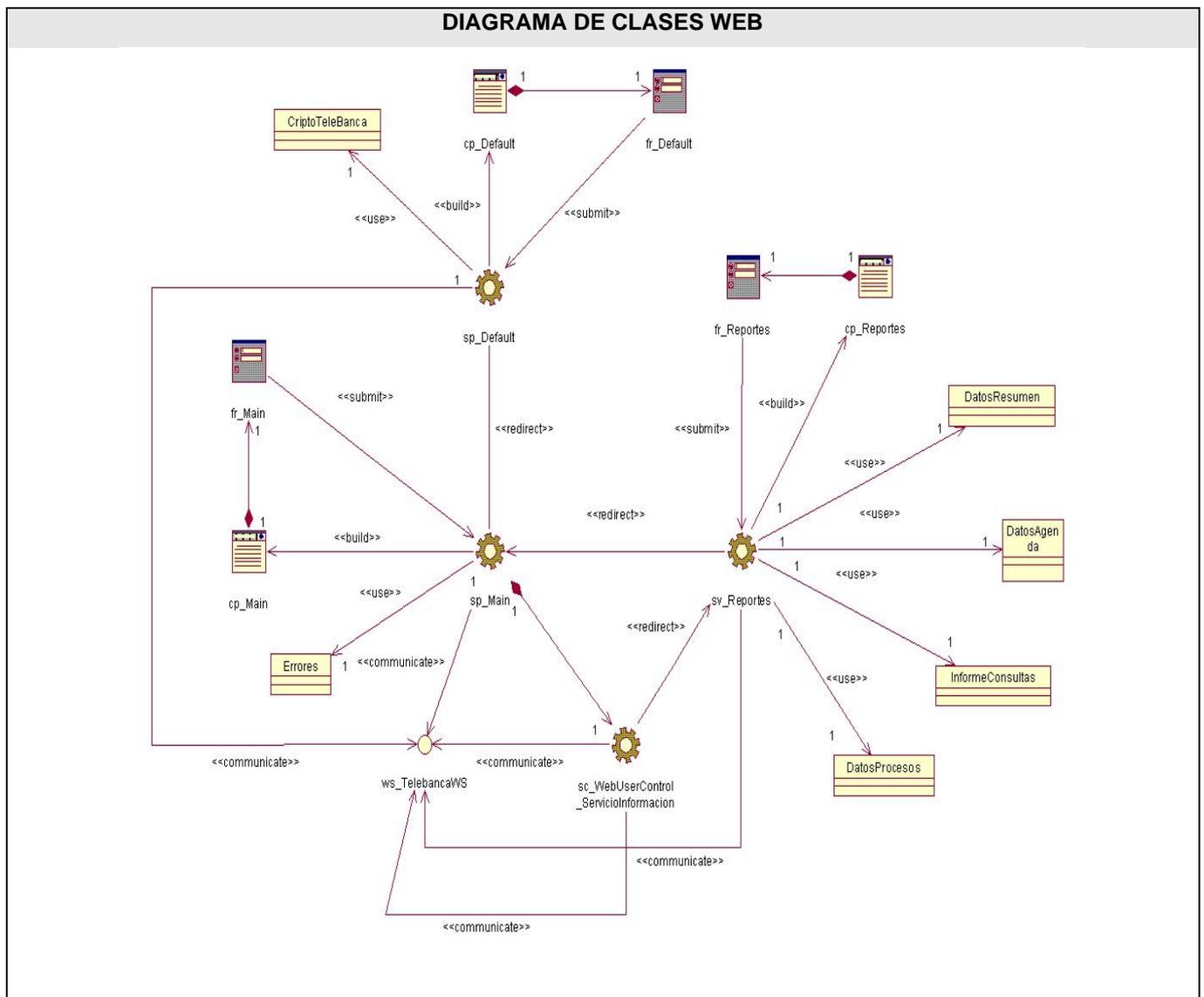


Figura 4. Diagrama de clases del diseño de la capa de lógica de la presentación.

3.3.2.2 Diagrama de clases del diseño de la capa de negocio

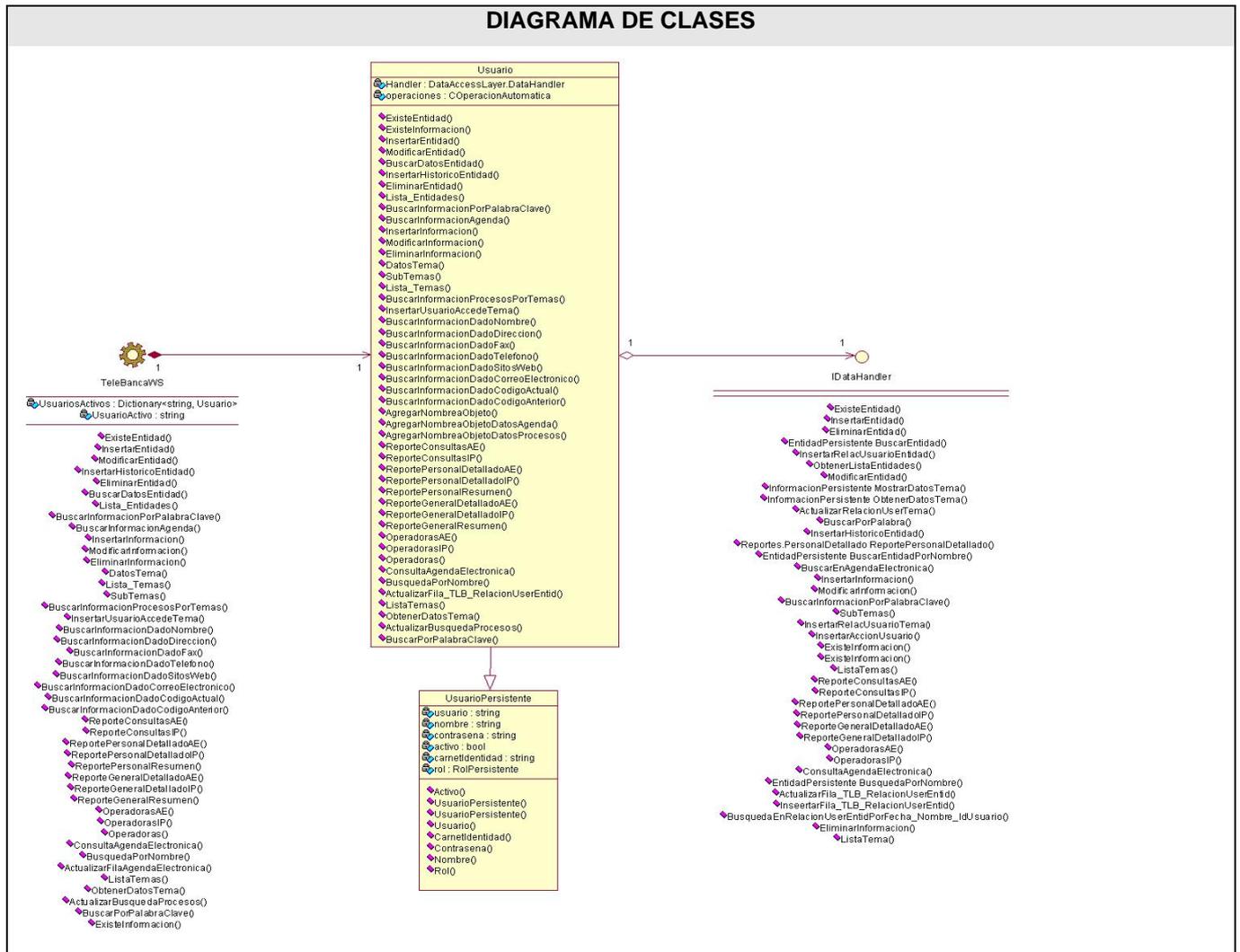


Figura 5. Diagrama de clases del diseño de la capa de negocio.

3.3.3 Descripción de las Clases

Nombre: WebUserControl_ServicioInformacion

Tipo de clase: Interfaz

Atributo	Tipo
Servicio	TeleBancaWS.TeleBancaWS
Rest	Thread
Salv	Thread
Para cada responsabilidad:	
Nombre:	Button1_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para gestionar la Agenda Electrónica.
Nombre:	Button2_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista de Gestionar Información de Procesos.
Nombre:	Button3_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para buscar en la Agenda Electrónica.
Nombre:	Button4_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para buscar información de procesos.
Nombre:	Button5_Click1(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para insertar información en la Agenda Electrónica.
Nombre:	Button6_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para modificar la Entidad.
Nombre:	Button7_Click(object sender, EventArgs e)
Descripción:	Se encarga de enviar la solicitud para eliminar una Entidad.
Nombre:	Button8_Click(object sender, EventArgs e)
Descripción:	Se encarga de verificar la validez de los datos proporcionados al insertar una entidad.
Nombre:	Button9_Click1(object sender, EventArgs e)
Descripción:	Se encarga cancelar la creación de una nueva Entidad y muestra la vista de gestionar la Agenda Electrónica.
Nombre:	Button10_Click(object sender, EventArgs e)
Descripción:	Se encarga de enviar los datos y la solicitud para el reporte del trabajo de las operadoras.
Nombre:	Button11_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar el reporte del trabajo de las operadoras.

Nombre:	Button12_Click1(object sender, EventArgs e)
Descripción:	Se encarga de enviar los datos y la solicitud para modificar una entidad.
Nombre:	Button13_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la modificación de una entidad y muestra la vista de gestionar la Agenda Electrónica.
Nombre:	Button14_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la búsqueda de información de procesos por tema y muestra la vista de buscar información de procesos.
Nombre:	Button15_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar toda la lista de las agendas electrónicas mostrada y muestra la vista para buscar en la agenda electrónica.
Nombre:	Button16_Click(object sender, EventArgs e)
Descripción:	Se encarga de enviar la solicitud para hacer los reportes ya sea de Agenda Electrónica o Información de procesos.
Nombre:	Button17_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar el reporte general detallado de agenda electrónica y muestra la vista de seleccionar los reportes de trabajo de la operadora.
Nombre:	Button18_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para insertar información de procesos.
Nombre:	Button19_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para modificar información de procesos.
Nombre:	Button20_Click(object sender, EventArgs e)
Descripción:	Se encarga de enviar los datos y la solicitud para eliminar un tema.
Nombre:	Button21_Click(object sender, EventArgs e)
Descripción:	Se encarga de verificar la validez de los datos proporcionados al insertar una información de proceso.
Nombre:	Button22_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la creación de un tema y muestra la vista de gestionar información de procesos.

Nombre:	Button23_Click(object sender, EventArgs e)
Descripción:	Se encarga de aceptar que se realice el reporte personal resumen.
Nombre:	Button24_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar el reporte personal resumen y muestra la vista de seleccionar los reportes de trabajo de la operadora.
Nombre:	Button25_Click1(object sender, EventArgs e)
Descripción:	Se encarga de enviar los datos y la solicitud para modificar una información de proceso.
Nombre:	Button26_Click1(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la modificación de un tema y muestra la vista de gestión de información de proceso.
Nombre:	Button27_Click(object sender, EventArgs e)
Descripción:	Se encarga de enviar los datos y la solicitud para hacer los reportes de Informe de Consultas ya sea en la agenda electrónica o la información de proceso.
Nombre:	Button28_Click1(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para hacer los reportes por el Trabajo de las Operadoras.
Nombre:	Button29_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para el reporte del Informes de las Consultas.
Nombre:	Button30_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la selección de los reportes de trabajo de las operadoras y muestra para seleccionar el tipo de reporte ya sea trabajo de las operadoras o informe de consulta.
Nombre:	Button31_Click(object sender, EventArgs e)
Descripción:	Se encarga de buscar las entidades según el criterio de búsqueda.
Nombre:	Button32_Click(object sender, EventArgs e)
Descripción:	Se encarga de buscar y mostrar los datos de la entidad seleccionada.
Nombre:	Button33_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar los datos de la entidad buscada y después muestra la vista de inicio del módulo de información.
Nombre:	Button34_Click(object sender, EventArgs e)

Descripción:	Se encarga de mostrar la vista para la búsqueda de información de procesos por tema.
Nombre:	Button35_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar la vista para la búsqueda de información de procesos por palabra clave.
Nombre:	Button36_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar los datos del tema seleccionado de la búsqueda de información de proceso
Nombre:	Button37_Click(object sender, EventArgs e)
Descripción:	Se encarga de guardar los datos de la información de proceso buscada en un histórico y muestra la vista de buscar información de procesos.
Nombre:	Button38_Click(object sender, EventArgs e)
Descripción:	Se encarga de cerrar el tema consultado y muestra la vista de inicio del módulo de información.
Nombre:	Button39_Click(object sender, EventArgs e)
Descripción:	Se encarga de buscar la información de proceso según el criterio de búsqueda.
Nombre:	Button40_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar datos de tema seleccionado de la búsqueda por palabra clave.
Nombre:	Button41_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar mostrar datos de tema seleccionado de la búsqueda por palabra clave.
Nombre:	Button42_Click(object sender, EventArgs e)
Descripción:	Se encarga de mostrar el tema buscado de la búsqueda por palabra clave.
Nombre:	Button43_Click(object sender, EventArgs e)
Descripción:	Se encarga de cerrar el tema consultado y muestra la vista de inicio del módulo de información.
Nombre:	Button44_Click1(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la selección de los reportes de informes de consultas y muestra para seleccionar el tipo de reporte ya sea trabajo de las operadoras o informe de consulta.

Nombre:	Button45_Click2(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el teléfono que esta selccionado en el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button46_Click(object sender, EventArgs e)
Descripción:	Se encarga de aceptar para la realización de los reportes de trabajo de la operadora con la selección de los distintos tipos de reportes y el criterio.
Nombre:	Button47_Click1(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un teléfono para el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button48_Click(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el sitio web que esta selccionado en el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button49_Click(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un sitio web para el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button50_Click1(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el correo electrónico que esta selccionado en el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button51_Click1(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un correo electrónico para el DropDownList9 en Insertar información agenda electrónica.
Nombre:	Button52_Click(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el teléfono que esta selccionado en el DropDownList12 en modificar información en la agenda electrónica.
Nombre:	Button53_Click(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un teléfono para el DropDownList12 en modificar información agenda electrónica.
Nombre:	Button54_Click(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el sitio web que esta selccionado en el DropDownList12 en

	modificar información en la agenda electrónica.
Nombre:	Button55_Click(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un sitio web para el DropDownList12 en modificar información agenda electrónica.
Nombre:	Button56_Click(object sender, EventArgs e)
Descripción:	Se encarga de eliminar el correo electrónico que esta selccionado en el DropDownList12 en modificar información en la agenda electrónica.
Nombre:	Button57_Click(object sender, EventArgs e)
Descripción:	Se encarga de adicionar un correo electrónico para el DropDownList12 en modificar información agenda electrónica.
Nombre:	Button58_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la búsqueda de la agenda electrónica hecha y muestra la vista de inicio de la búsqueda de agenda electrónica.
Nombre:	Button59_Click(object sender, EventArgs e)
Descripción:	Se encarga de cancelar la búsqueda de agenda electrónica y muestra la vista de las diferentes búsquedas.
Nombre:	Page_Load(object sender, EventArgs e)
Descripción:	Se encarga de cargar la página y todas las funcionalidades programadas.
Nombre:	Menu1_MenuItemClick(object sender, MenuEventArgs e)
Descripción:	Se encarga de acceder a las distaintas funcionalidades del menú del modulo de información.
Nombre:	InsertarEntidad()
Descripción:	Se encarga de insertar una entidad.
Nombre:	QuitarEspacios(string cadena)
Descripción:	Se encarga de quitarle los espacios a una cadena de caracteres.
Nombre:	Insertar()
Descripción:	Se encarga de mostrar el mensaje de que Se ha insertado satisfactoriamente la Información de Proceso.
Nombre:	EliminarDefinitivo()

Descripción:	Se encarga de enviar la solicitud para eliminar una entidad o si no ha podido ser eliminada.
Nombre:	TieneSubTemas(int idTema)
Descripción:	Se encarga de verificar si un tema tiene subtemas asociados.
Nombre:	Eliminar()
Descripción:	Se encarga de enviar los datos y la solicitud para eliminar información.
Nombre:	ListBox2_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de si hay algun elemento seleccionado en el ListBox2 activa los botones de modificar y eliminar la información de proceso.
Nombre:	ModificarEntidad()
Descripción:	Se encarga de enviar los datos y la solicitud para modificar información.
Nombre:	Calendar3_Init(object sender, EventArgs e)
Descripción:	Se encarga de recoger la fecha de inicio en el intervalo para un posterior reporte, por defecto tiene la fecha actual.
Nombre:	Calendar1_Init(object sender, EventArgs e)
Descripción:	Se encarga de recoger la fecha de inicio en el intervalo para un posterior reporte, por defecto tiene la fecha actual.
Nombre:	Calendar2_Init(object sender, EventArgs e)
Descripción:	Se encarga de recoger la fecha de fin en el intervalo para un posterior reporte, por defecto tiene la fecha actual.
Nombre:	Calendar4_Init(object sender, EventArgs e)
Descripción:	Se encarga de recoger la fecha de fin en el intervalo para un posterior reporte, por defecto tiene la fecha actual.
Nombre:	Calendar4_SelectionChanged(object sender, EventArgs e)
Descripción:	Se encarga de validar el rango de fecha para los reportes de informe de consulta.
Nombre:	DropDownList5_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de seleccionar el tipo de información del reporte general detallado, ya sea Agenda electrónica o Información de proceso.
Nombre:	View22_Activate(object sender, EventArgs e)

Descripción:	Se encarga de mostrar el listado de todas la operadoras del Sistema.
Nombre:	View21_Activate(object sender, EventArgs e)
Descripción:	Se encarga de validar tener seleccionado una tipo de información para mostrar las operadoras.
Nombre:	Calendar3_SelectionChanged(object sender, EventArgs e)
Descripción:	Se encarga de validar el rango de fecha para los reportes de informe de consulta.
Nombre:	Calendar1_SelectionChanged(object sender, EventArgs e)
Descripción:	Se encarga de validar el rango de fecha para los reportes de trabajo de operadoras.
Nombre:	Calendar2_SelectionChanged(object sender, EventArgs e)
Descripción:	Se encarga de validar el rango de fecha para los reportes de informe de consulta.
Nombre:	ListBox8_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el boton aceptar para mostrar el reporte trabajo de las operadoras cuando se selecciona un tipo de informacion y una operadora.
Nombre:	ListBox10_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el boton aceptar para hacer el reporte personal resumen.
Nombre:	Modificar()
Descripción:	Se encarga de enviar los datos y la solicitud para modificar información.
Nombre:	View7_Activate(object sender, EventArgs e)
Descripción:	Se encarga de mostrar en el ListBox2 la lista de información de proceso que existe y desabilitar los botones modificar y eliminar del gestionar informacion de proceso.
Nombre:	HabilitarBoton(CheckBox Ch, Button Bu)
Descripción:	Se encarga de habilitar botonescuando se marca algun CheckBox.
Nombre:	View12_Activate(object sender, EventArgs e)
Descripción:	Se encarga de desabilitar el botón aceptar en el buscar agenda electrónica.
Nombre:	CheckBox1_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox2_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox3_CheckedChanged(object sender, EventArgs e)

Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox4_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox5_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox6_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox7_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	CheckBox8_CheckedChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	ListBox4_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón Aceptar de Buscar Información de proceso por tema.
Nombre:	View4_Activate(object sender, EventArgs e)
Descripción:	Se encarga de cargar la lista de todas las entidades que hay y deshabilitar los botones modificar y eliminar.
Nombre:	View6_Activate(object sender, EventArgs e)
Descripción:	Se encarga de cargar los datos de la entidad.
Nombre:	ListBox1_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar los botones modificar y eliminar cuando se selecciona una entidad en el ListBox1 de la agenda electrónica.
Nombre:	ListBox3_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar lo botón aceptar cuando se selecciona una entidad en el ListBox3 del buscar en la agenda electrónica.
Nombre:	View11_Activate(object sender, EventArgs e)
Descripción:	Se encarga de cuando se cargue la vista deshabilitar el botón aceptar del buscar en la agenda electrónica.
Nombre:	ListBox6_SelectedIndexChanged(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar de buscar por palabra clave.

Nombre:	View17_Activate(object sender, EventArgs e)
Descripción:	Se encarga de habilitar el botón aceptar de buscar por palabra clave.

Nombre: TeleBancaWS	
Tipo de clase: Controladora	
Atributo	Tipo
UsuariosActivos	Dictionary<string, Usuario>
UsuarioActivo	string
Para cada responsabilidad:	
Nombre:	ExisteEntidad(string nombre)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	InsertarEntidad(string nombre, string direccion, string telefono, string fax, string codigo, string codigoAnterior, string sitiosWeb, string correosElectronicos)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ModificarEntidad(string nombre, string direccion, string telefono, string fax, string codigo, string codigoAnterior, string sitiosWeb, string correosElectronicos)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	InsertarHistoricoEntidad(string nombre, string direccion, string telefono, string fax, string codigoactual, string codigoanterior, string correo, string sitioweb)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	EliminarEntidad(string nombre) EliminarEntidad(string nombre)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarDatosEntidad(string nombre)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	Lista_Entidades()
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionPorPalabraClave(string palabraClave)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionAgenda(string consulta)

Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	InsertarInformacion(string tema, string[] palabrasClaves, int temaPadre, string texto)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ModificarInformacion(string tema, List<string> palabrasClaves, int temaPadre, string texto, int idTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	EliminarInformacion(int idTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	DatosTema(int idTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	Lista_Temas()
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	SubTemas(int idTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionProcesosPorTemas(string tema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	InsertarUsuarioAccedeTema(int idTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoNombre(string nombre)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoDireccion(string direccion)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoFax(string fax)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoTelefono(string telefono)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoSitosWeb(string sitiosWeb)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoCorreoElectronico(string correoElectronico)

Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoCodigoActual(string codigoActual)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarInformacionDadoCodigoAnterior(string codigoAnterior)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReporteConsultasAE(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReporteConsultasIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReportePersonalDetalladoAE(DateTime pFechaI, DateTime pFechaF, string pId_Operadora)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReportePersonalDetalladoIP(DateTime pFechaI, DateTime pFechaF, string pId_Operadora)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReportePersonalResumen(string pUsuario, DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReporteGeneralDetalladoAE(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReporteGeneralDetalladoIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ReporteGeneralResumen(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	OperadorasAE(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	OperadorasIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	Operadoras(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.

Nombre:	ConsultaAgendaElectronica(string nombre, string Direccion, string Fax, string Telefono, string correoElectronico, string SitioWeb, string CodAntSucursal, string CodActual)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BusquedaPorNombre(string Nombre)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ActualizarFilaAgendaElectronica(int Cod_Entidad, int CodAntSucursal, int Direccion, int Fax, int Telefono, int CorreosElectronicos, int SitiosWeb, int NombreC, DateTime FechaOriginal, string NombreOriginal, string Idusuario)
Descripción:	Se encarga de invocar al método ActualizarFila_TLB_RelacionUserEntid de la clase Usuario.
Nombre:	ListaTemas()
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ObtenerDatosTema(string Tema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ActualizarBusquedaProcesos(DateTime Fecha,string Idusuario,string IdTema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	BuscarPorPalabraClave(string palabra)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.
Nombre:	ExisteInformacion(string Tema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase Usuario.

Nombre: Usuario	
Tipo de clase: Controladora	
Atributo	Tipo
Handler	DataAccessLayer.DataHandler
operaciones	COperacionAutomatica
Para cada responsabilidad:	
Nombre:	ExisteEntidad(string nombre)

Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ExisteInformacion(string Tema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	InsertarEntidad(string nombre, string direccion, string telefono, string fax, string codigo, string codigoAnterior, string sitiosWeb, string correosElectronicos)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ModificarEntidad(string nombre, string direccion, string telefono, string fax, string codigo, string codigoAnterior, string sitiosWeb, string correosElectronicos)
Descripción:	Se encarga de invocar a los métodos ExisteEntidad y ModificarEntidad de la clase DataHandler.
Nombre:	BuscarDatosEntidad(string nombre)
Descripción:	Se crea una variable de tipo EntidadPersistente llamada entidad he invoca al metodo BuscarEntidad de la clase DataHandler.
Nombre:	InsertarHistoricoEntidad(string nombre, string direccion, string telefono, string fax, string codigoactual, string codigoanterior, string correo, string sitioweb)
Descripción:	Se crea un objeto de tipo EntidadPersistente y se invoca al método InsertarHistoricoEntidad de la clase DataHandler.
Nombre:	EliminarEntidad(string nombre)
Descripción:	Se encarga de invocar al método ExisteEntidad y al método EliminarEntidad de la clase DataHandler.
Nombre:	Lista_Entidades()
Descripción:	Se encarga de invocar al método ObtenerListaEntidades de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionPorPalabraClave(string palabraClave)
Descripción:	Se encarga de invocar al método BuscarInformacionPorPalabraClave de la clase DataHandler y crea y retorna un arreglo con la lista de las entidades.
Nombre:	BuscarInformacionAgenda(string consulta)
Descripción:	Se encarga de invocar al método BuscarEnAgendaElectronicade la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.

Nombre:	InsertarInformacion(string tema, string[] palabrasClaves, int temaPadre, string texto)
Descripción:	Crea un objeto de tipo List<string> y invoca a los metodos ExisteInformacion y InsertarInformacion de la clase DataHandler.
Nombre:	ModificarInformacion(string tema, List<string> palabrasClaves, int temaPadre, string texto, int idTema)
Descripción:	Se encarga de invocar a los métodos ExisteInformacion y ModificarInformacion de la clase DataHandler.
Nombre:	EliminarInformacion(int idTema)
Descripción:	Se encarga de invocar al método ExisteInformacion y al método EliminarInformacion de la clase DataHandler.
Nombre:	DatosTema(int idTema)
Descripción:	Se encarga de crear y retornar un arreglo con los temas e invoca al método ObtenerDatosTema de la clase DataHandler.
Nombre:	SubTemas(int idTema)
Descripción:	Se encarga de crear y retornar un arreglo con los nombres de los subtemas e invoca al método SubTemas de la clase DataHandler.
Nombre:	Lista_Temas()
Descripción:	Se encarga de invocar al método ListaTemas de la clase DataHandler y crea y retorna un arreglo con la lista de temas.
Nombre:	BuscarInformacionProcesosPorTemas(string tema)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	InsertarUsuarioAccedeTema(int idTema)
Descripción:	Se encarga de invocar al método InsertarRelacUsuarioTema de la clase DataHandler.
Nombre:	BuscarInformacionDadoNombre(string nombre)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoNombre de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoDireccion(string direccion)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoDireccion de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.

Nombre:	BuscarInformacionDadoFax(string fax)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoFax de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoTelefono(string telefono)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoTelefono de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoSitosWeb(string sitiosWeb)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoSitosWeb de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoCorreoElectronico(string correoElectronico)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoCorreoElectronico de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoCodigoActual(string codigoActual)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoCodigoActual de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	BuscarInformacionDadoCodigoAnterior(string codigoAnterior)
Descripción:	Se encarga de invocar al método BuscarInformacionDadoCodigoAnterior de la clase DataHandler y crea y retorna un arreglo con los nombres de las entidades.
Nombre:	AgregarNombreaObjeto(DataAccessLayer.InformeConsultas aInformeConsulta)
Descripción:	Se encarga de obtener el nombre del usuario que esta pidiendo el reporte y lo muestra en el reporte de informe de consulta.
Nombre:	AgregarNombreaObjetoDatosAgenda(DataAccessLayer.DatosAgenda aDatosAgenda)
Descripción:	Se encarga de obtener el nombre del usuario que esta pidiendo el reporte y lo muestra en el reporte de DatosAgenda.
Nombre:	AgregarNombreaObjetoDatosProcesos(DataAccessLayer.DatosProcesos aDatosProcesos)
Descripción:	Se encarga de obtener el nombre del usuario que esta pidiendo el reporte y lo muestra en el reporte de DatosProcesos.
Nombre:	ReporteConsultasAE(DateTime pFechaI, DateTime pFechaF)

Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReporteConsultasIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReportePersonalDetalladoAE(DateTime pFechaI, DateTime pFechaF, string pId_Operadora)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReportePersonalDetalladoIP(DateTime pFechaI, DateTime pFechaF, string pId_Operadora)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReportePersonalResumen(string pUsuario, DateTime pFechaI, DateTime pFechaF)
Descripción:	Invoca a los métodos ReportePersonalDetalladoAE , ReportePersonalDetalladoIP y BuscarUsuario de la clase DataHandler, además crea un arreglo de tipo DatosResumen y otro de tipo DatosResumen con sus atributos.
Nombre:	ReporteGeneralDetalladoAE(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReporteGeneralDetalladoIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ReporteGeneralResumen(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de crear un arreglo de tipo Dictionary<DateTime, DatosResumen> e invocar a los métodos ReporteGeneralDetalladoAE y ReporteGeneralDetalladoIP, crea un arreglo de tipo DatosResumen y crea otro de tipo DatosResumen y lo retorna con sus atributos.
Nombre:	OperadorasAE(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	OperadorasIP(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	Operadoras(DateTime pFechaI, DateTime pFechaF)
Descripción:	Se encarga de invocar a los métodos OperadorasIP y OperadorasAE de la clase DataHandler y crea arreglos de tipo Dictionary<string, UsuarioPersistente> y

	UsuarioPersistente.
Nombre:	ConsultaAgendaElectronica(string nombre, string Direccion, string Fax, string Telefono, string correoElectronico, string SitioWeb, string CodAntSucursal, string CodActual)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	BusquedaPorNombre(string Nombre)
Descripción:	Se encarga de invocar al método BuscarEntidadPorNombre de la clase DataHandler.
Nombre:	ActualizarFila_TLB_RelacionUserEntid(int Cod_Entidad, int CodAntSucursal, int Direccion, int Fax, int Telefono, int CorreosElectronicos, int SitiosWeb, int NombreC, DateTime FechaOriginal, string NombreOriginal, string Idusuario)
Descripción:	Se encarga de invocar a los métodos BusquedaEnRelacionUserEntidPorFecha_Nombre_IdUsuario, InseertarFila_TLB_RelacionUserEntid y ActualizarFila_TLB_RelacionUserEntid de la clase DataHandler.
Nombre:	ListaTemas()
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.
Nombre:	ObtenerDatosTema(string Tema)
Descripción:	Se encarga de invocar al método MostrarDatosTema de la clase DataHandler.
Nombre:	ActualizarBusquedaProcesos(DateTime Fecha, string Idusuario, string IdTema)
Descripción:	Se encarga de invocar al método ActualizarRelacionUserTema de la clase DataHandler.
Nombre:	BuscarPorPalabraClave(string palabra)
Descripción:	Se encarga de invocar al método de ese mismo nombre de la clase DataHandler.

3.3.4 Diagrama de clases persistentes

DIAGRAMA DE CLASES PERSISTENTES



Figura 6. Diagrama de clases persistentes.

3.3.5 Tratamiento de errores

Identificar y controlar los posibles errores que se pueden presentar a la hora de interactuar con el software es importante para así garantizar un correcto funcionamiento del sistema.

En el sistema la validación para el tratamiento de errores serán referidos principalmente a errores de la interacción del usuario con el sistema, evidentemente los errores ocurrirán incluso con los usuarios de más habilidad y experiencia; de manera general para la manipulación de los errores se utilizó la clase Errores que se encarga lanzamiento de mensajes de error y alertas y lanzamiento de confirmaciones así que todos los datos de entrada serán validados. Se mostrarán mensajes de confirmación en el caso de las inserciones, modificaciones y eliminaciones de manera que no se realice una operación no deseada. En el caso de los datos que sean adicionados por un usuario del sistema, se hace una validación de estos mediante funciones que garantizan que sean válidos, en caso que ocurra un error, el sistema marca con un asterisco los campos no válidos y muestra el mensaje “Entrada de datos no válida”. Al obtener la confirmación de lectura del mensaje de error por parte del usuario oprimiendo el botón “Aceptar”, la caja del diálogo desaparecerá y continuará la ejecución de la aplicación.

Tratamos de que el usuario introduzca la menor cantidad de datos ayudándolo mediante la utilización de componentes para evitar incoherencias e incorrecciones en los mismos.

3.4 Conclusiones

En este capítulo quedan finalizadas las etapas de análisis y diseño del sistema graficándose los diagramas de interacción mediante los diagramas de secuencia de manera tal que se desarrollo un diagrama de secuencia para cada escenario de un caso de uso. Con estos resultados del análisis y diseño representamos las clases del análisis en un diagrama de clases del análisis, además quedaron divididos por paquetes los diagramas de clases del diseño, en un diagrama de clases web quedó representada la capa de presentación y en un diagrama de clases del diseño la capa de negocio. Con todos estos elementos, se tiene una idea más precisa sobre los elementos constitutivos del Sistema que se propone.

Con la culminación de este capítulo se obtuvo una visión mas específica del Sistema logrando crear una entrada apropiada y un punto de partida para actividades de implementación. Además se adquirió una comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, interfaz de usuario, etc.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

4.1 Introducción

En este capítulo se muestran detalles de la construcción de la solución propuesta, basándose en la implementación y prueba. En la fase de implementación se muestra el diagrama de despliegue correspondiente y los diagramas de componentes divididos en capas del Módulo de Servicio de Información concluyendo así este capítulo con el modelo de prueba y sus respectivos casos de prueba.

4.2 Implementación

En el flujo de trabajo de implementación se define la organización del sistema en términos de subsistemas de Implementación organizados en capas describiendo cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue.

4.2.1 Diagrama de despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final.

En el diagrama de despliegue se indica la situación física de los componentes lógicos desarrollados. Es decir se sitúa el software en el hardware que lo contiene.

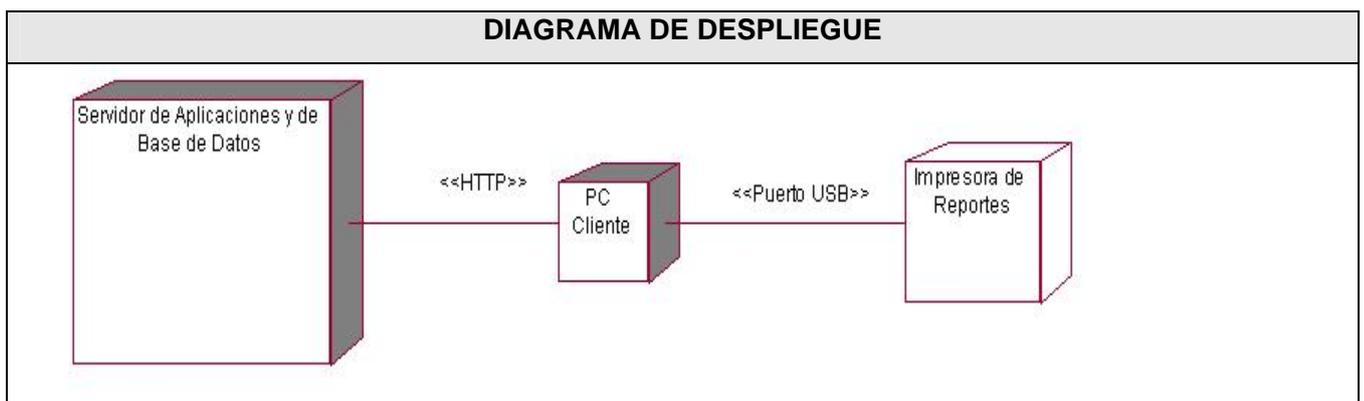


Figura 7. Diagrama de despliegue.

4.2.2 Diagrama de componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables.

A continuación se muestra una representación de las relaciones que existen entre los componentes agrupados en paquetes de la capa de presentación y la capa de negocio, que representan las capas del Módulo de Servicio de Información del proyecto Telebanca.

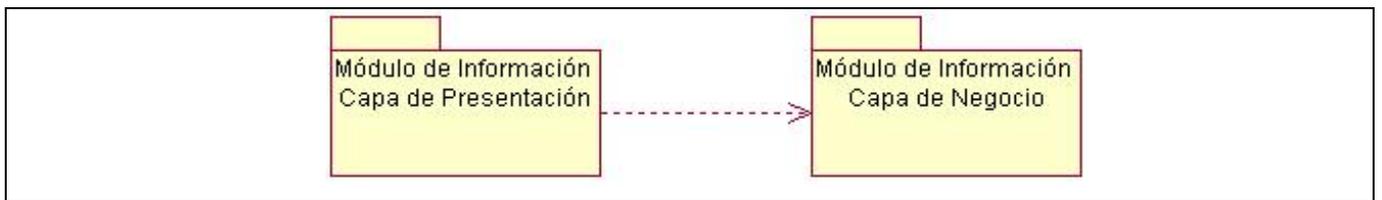


Figura 8. Diagrama general de componentes del Módulo de Servicio de Información.

4.2.2.1 Diagrama de componentes de la capa de presentación

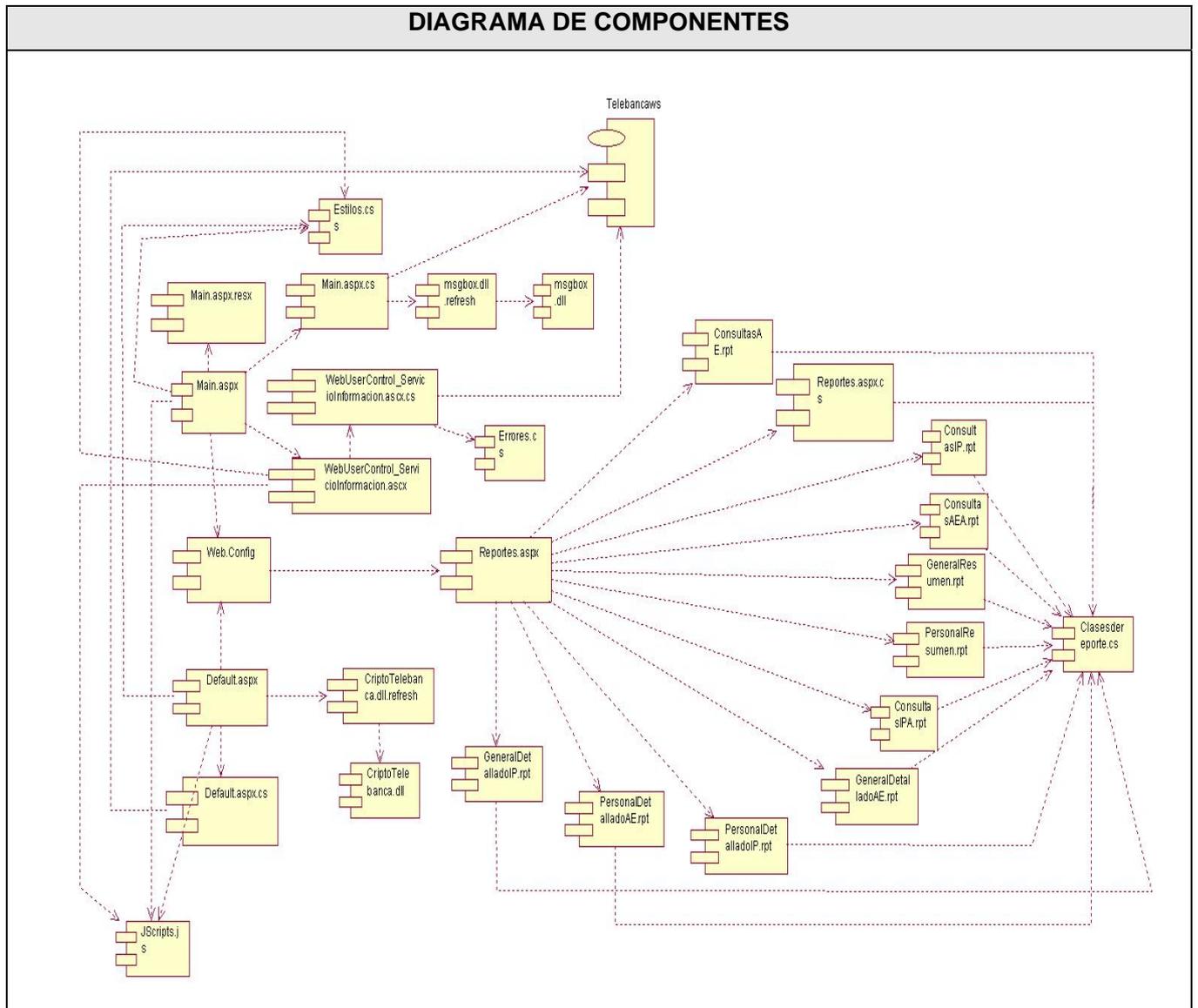


Figura 9. Diagrama de componentes de la capa de presentación.

4.2.2.2 Diagrama de componentes de la capa de negocio

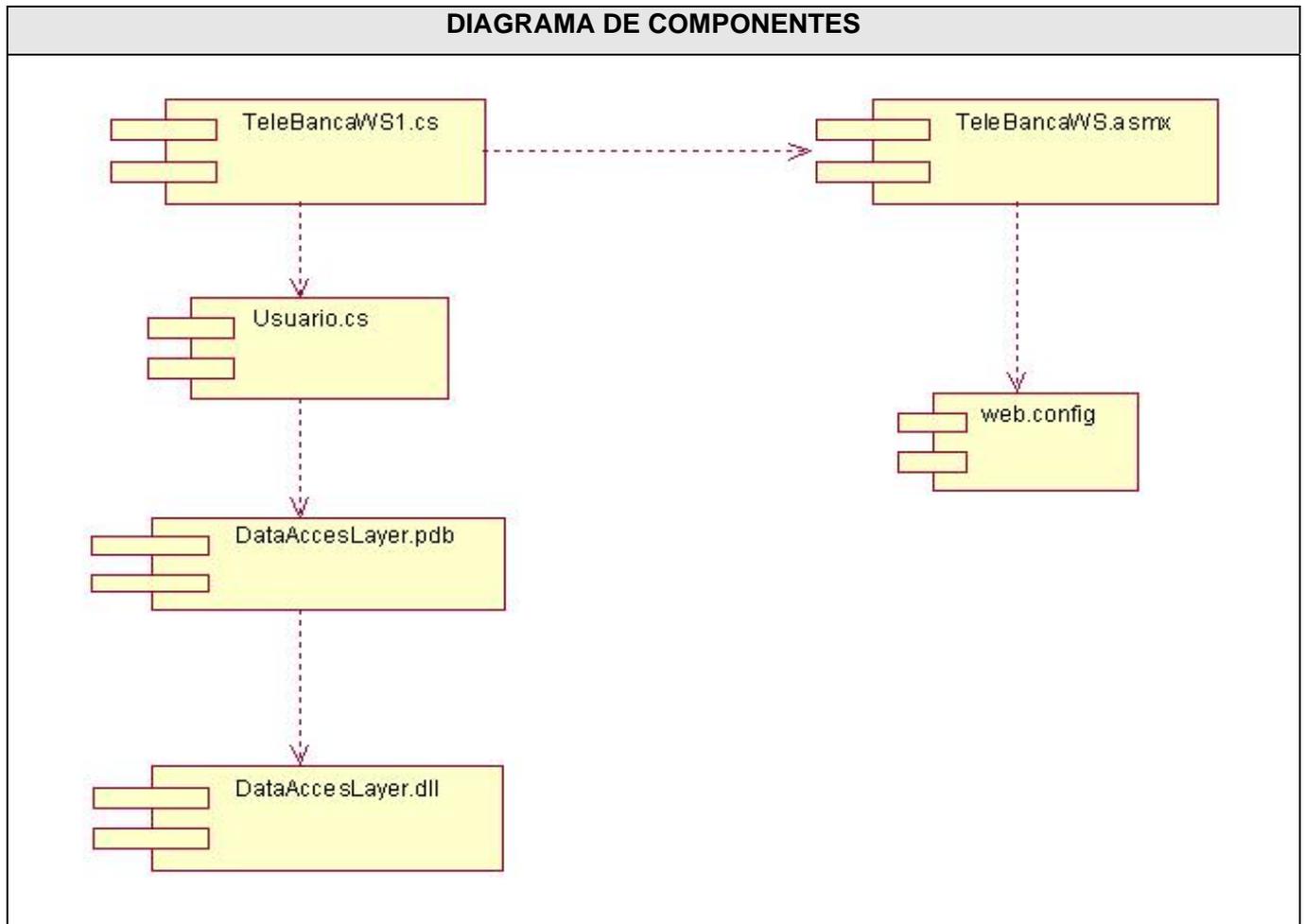


Figura 10. Diagrama de componentes de la capa de lógica del negocio.

La siguiente tabla justifica la presencia de cada componente dando una breve explicación de cada uno de ellos.

Componente	Propósito
Default.aspx	Página Web de autenticación de la aplicación.
Default.aspx.cs	Componente de código de la Página Web de autenticación de la aplicación.
Main.aspx	Página Web de inicio de la aplicación.

Main.aspx.cs	Fichero de código contenedor de la Página Web de inicio de la aplicación.
Main.aspx.resx	Componente auxiliar para la Página Web de inicio de la aplicación.
Reportes.aspx	Página Web para realizar los reportes.
Reportes.aspx.cs	Fichero de código contenedor de la Página Web de reportes.
Web.Config	Contiene los parámetros del Sitio Web.
WebUserControl_ServicioInformacion.ascx	Componente de la página WebUserControl_ServicioInformacion.
WebUserControl_ServicioInformacion.ascx.cs	Fichero de código la página WebUserControl_ServicioInformacion.
CriptoTelebanca.dll	Librería para la criptografía de autenticación de los usuarios.
msgbox.dll	Librería para los mensajes de error.
CriptoTelebanca.dll.refresh	Interfaz entre la librería CriptoTelebanca y la aplicación.
msgbox.dll.refresh	Interfaz entre la librería de los mensajes de error y la aplicación.
ConsultasAE.rpt	Plantilla del reporte de las consultas en la Agenda Electrónica.
ConsultasAEA.rpt	Plantilla del reporte en orden ascendente de las consultas en la Agenda Electrónica.
ConsultasIP.rpt	Plantilla del reporte de las consultas Información de Procesos.
ConsultasIPA.rpt	Plantilla del reporte en orden ascendente de las consultas en la Información de Procesos.
GeneralDetalladoAE.rpt	Plantilla del reporte General Detallado en la Agenda Electrónica.
GeneralDetalladoIP.rpt	Plantilla del reporte General Detallado en la Información

	de Procesos.
GeneralResumen.rpt	Plantilla del reporte General Resumen.
PersonalDetalladoAE.rpt	Plantilla del reporte Personal Detallado en la Agenda Electrónica.
PersonalDetalladoIP.rpt	Plantilla del reporte Personal Detallado en la Información de Procesos.
PersonalResumen.rpt	Plantilla del reporte Personal Resumen.
Clasesdereporte.cs	Clases que contienen los datos que saldrán en los reportes
Errores.cs	Fichero de código para el tratamiento de errores.
Paquete TeleBancaWS	Describe las funcionalidades del Web Service de la lógica del negocio.
Estilos.css	Describe el estilo de las páginas web.
TeleBancaWS1.cs	Fichero contenedor de código del web service.
Usuario.cs	Fichero contenedor de código de la clase Usuario.
TeleBancaWS.asmx	Fichero que soporta el Web Service.
web.config	Configuración de parámetros del sitio web
DataAccesLayer.dll	Librería de comunicación con la Capa de acceso a datos.
DataAccesLayer.pdb	Interfaz de comunicación entre el Sistema y la capa de acceso a datos

4.3 Prueba

Durante la prueba se pretende encontrar y documentar los defectos que puedan afectar la calidad del software, validar que el software trabaje como fue diseñado, validar y probar en forma de demostración los requisitos que debe cumplir el software, validar que los requisitos fueron implementados correctamente. La prueba no puede asegurar la ausencia de defectos; solo pueden demostrar que existen defectos en el software. En la prueba lo fundamental es encontrar y exponer las debilidades en el software.

4.3.1 Modelo de prueba

A continuación se muestran los casos de prueba los cuales serán descritos tanto para condiciones de entradas válidas, como para condiciones de entradas inválidas así como para los resultados esperados y las condiciones que deben cumplirse mientras se ejecuta el caso de uso. En este caso se mostrarán algunos de los casos de prueba de los casos de usos más importantes, el resto de los casos de prueba se encuentran en el Anexo 4.

4.3.1.1 Caso de prueba de integración CU _Gestionar Información

CU _Gestionar Información			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
Clic en el botón (Gestión de Agenda Electrónica)		El Sistema muestra el formulario con las diferentes opciones de la Agenda.	Que exista conexión con la base de datos.
Clic en el botón (Gestión de Información de Procesos.)		El Sistema muestra el formulario con las diferentes opciones de Gestión de la información.	Que exista conexión con la base de datos.

4.3.1.2 Caso de prueba de integración CU _Gestión de Agenda Electrónica: sección Insertar Información

CU _Gestión de Agenda Electrónica			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
Cerámica Blanca (Nombre)		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
	Cerámica Blanca	El Sistema muestra el mensaje "La entidad que se	Que exista conexión con la

	(Entrada repetida)	intenta insertar ya existe.” El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	base de datos.
	Cerámica _ blanca	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
La Habana, Cuba (Dirección)		El Sistema muestra el mensaje “¿Está seguro que desea insertar esta Entidad?”	Que exista conexión con la base de datos.
	La Habana, Cuba (Entrada repetida)	El Sistema muestra el mensaje “La entidad que se intenta insertar ya existe.”	Que exista conexión con la base de datos.

		El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
(153)-125356 (Teléfono)		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
	(dgh)-24525	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	#(125)-24536	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los	Que exista conexión con la base de datos.

		campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
FS-34XSJ (Fax)		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
	Hhsb-23\$#%	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	,jfgjdfgjnfgjnf	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema	Que exista conexión con la base de datos.

		marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
http:\\ cubasi.cu (Sitios Web Relacionados)		El Sistema muestra el mensaje “¿Está seguro que desea insertar esta Entidad?”	Que exista conexión con la base de datos.
	http:\\ ISLA.cu	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
4 (Código Anterior)		El Sistema muestra el mensaje “¿Está seguro que desea insertar esta Entidad?”	Que exista conexión con la base de datos.
9812546369		El Sistema muestra el mensaje “¿Está seguro que desea insertar esta Entidad?”	Que exista conexión con la base de datos.
	2547896254566	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema	Que exista conexión con la base de datos.

		marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
	87fghgjhjj	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
4563 (Código Actual)		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
9812546369		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
	5445455555126	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los	Que exista conexión con la base de datos.

		datos de una entidad de la agenda electrónica.	
	87fghghjj	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
isac@uh.cu (Correos electrónicos)		El Sistema muestra el mensaje "¿Está seguro que desea insertar esta Entidad?"	Que exista conexión con la base de datos.
	Ytsd@uh.cu	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el	Que exista conexión con la base de datos.

		formulario para entrar los datos de una entidad de la agenda electrónica.	
	per#uh.cu	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.

4.3.1.3 Caso de prueba de integración CU _Gestión de Agenda Electrónica: sección Modificar Información

CU _Gestión de Agenda Electrónica			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
La Habana, Cuba (Dirección)		El Sistema muestra el mensaje "¿Está seguro que desea modificar la	Que exista conexión con la base de datos.

		Entidad?”.	
	La Habana, Cuba (Entrada repetida)	El Sistema muestra el mensaje “La entidad a guardar ya existe.”	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
(153)-125356 (Teléfono)		El Sistema muestra el mensaje “¿Está seguro que desea modificar la Entidad?”.	Que exista conexión con la base de datos.
	(dgh)-24525	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	#(125)-24536	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los	Que exista conexión con la base de datos.

		campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
FS-34XSJ (Fax)		El Sistema muestra el mensaje "¿Está seguro que desea modificar la Entidad?".	Que exista conexión con la base de datos.
	Hhsb-23\$#%	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	,jfgjdfgjnfgjnf	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema	Que exista conexión con la base de datos.

		marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
http:\\ cubasi.cu (Sitios Web Relacionados)		El Sistema muestra el mensaje “¿Está seguro que desea modificar la Entidad?”.	Que exista conexión con la base de datos.
	http:\\ ISLA.cu	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
4 (Código Anterior)		El Sistema muestra el mensaje “¿Está seguro que desea modificar la Entidad?”.	Que exista conexión con la base de datos.
9812546369		El Sistema muestra el mensaje “¿Está seguro que desea modificar la Entidad?”.	Que exista conexión con la base de datos.
	2547896254566	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema	Que exista conexión con la base de datos.

		marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	
	87fghgjhjj	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
4563 (Código Actual)		El Sistema muestra el mensaje "¿Está seguro que desea modificar la Entidad?".	Que exista conexión con la base de datos.
9812546369		El Sistema muestra el mensaje "¿Está seguro que desea modificar la Entidad?".	Que exista conexión con la base de datos.
	5445455555126	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los	Que exista conexión con la base de datos.

		datos de una entidad de la agenda electrónica.	
	87fghghjj	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
isac@uh.cu (Correos electrónicos)		El Sistema muestra el mensaje "¿Está seguro que desea modificar la Entidad?".	Que exista conexión con la base de datos.
	Ytsd@uh.cu	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el	Que exista conexión con la base de datos.

		formulario para entrar los datos de una entidad de la agenda electrónica.	
	per#uh.cu	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	87fghghjj	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar los datos de una entidad de la agenda electrónica.	Que exista conexión con la base de datos.

Se selecciona una entidad. (Selección de entidades)		Se activa el botón Modificar	Que exista conexión con la base de datos.
	No se selecciona ninguna entidad	El Sistema no habilita la opción Modificar .	Que exista conexión con la base de datos.

4.3.1.4 Caso de prueba de integración CU _Gestión de Agenda Electrónica: sección Eliminar Información

CU _Gestión de Agenda Electrónica			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
Clic en el botón (Eliminar)		El Sistema muestra el mensaje “¿Esta seguro que desea eliminar la entidad seleccionada?”.	Que exista conexión con la base de datos.
Se selecciona una entidad. (Selección de entidades)		El Sistema habilita la opción Eliminar .	Que exista conexión con la base de datos.
	No se selecciona ninguna entidad	El Sistema no habilita la opción Eliminar .	Que exista conexión con la base de datos.

4.3.1.5 Caso de prueba de integración CU _Gestión de Información de Procesos: sección Insertar Información de Procesos

CU _Gestión de Información de Procesos			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
Drama		El Sistema muestra el	Que exista

(Tema)		mensaje “¿Estás seguro que desea insertar esta información de Proceso?”	conexión con la base de datos.
	Drama (Entrada repetida)	El Sistema muestra el mensaje “La Información de Proceso que se intenta insertar ya existe.”. El Sistema muestra el formulario para entrar la información.	Que exista conexión con la base de datos.
	#dfddgh	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar la información.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar la información.	Que exista conexión con la base de datos.
Fdgfghghgfg,hgfdhgf (Texto correspondiente al proceso)		El Sistema muestra el mensaje “¿Estás seguro que desea insertar esta información de Proceso?”	Que exista conexión con la base de datos.

	En blanco	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar la información.	Que exista conexión con la base de datos.
SRHDJF (Palabra Clave)		El Sistema muestra el mensaje "¿Estás seguro que desea insertar esta información de Proceso?"	Que exista conexión con la base de datos.
Ddhfhsjfdhj		El Sistema muestra el mensaje "¿Estás seguro que desea insertar esta información de Proceso?"	Que exista conexión con la base de datos.
	Hhsb-23\$#%	El Sistema muestra el mensaje "Entrada de datos no válida" y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra el formulario para entrar la información.	Que exista conexión con la base de datos.

4.3.1.6 Caso de prueba de integración CU _Gestión de Información de Procesos: sección Modificar Información de Procesos

CU _Gestión de Información de Procesos			
Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones

Drama (Tema)		El Sistema muestra el mensaje “¿Está seguro que desea Modificar la Información de Proceso?”	Que exista conexión con la base de datos.
	Drama (Entrada repetida)	El Sistema muestra el mensaje “El tema a guardar ya existe”. El Sistema muestra los datos del tema seleccionado.	Que exista conexión con la base de datos.
	#dfddgfh	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra los datos del tema seleccionado.	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra los datos del tema seleccionado.	Que exista conexión con la base de datos.
Fdgfghfghfg,hgdfhgf (Texto correspondiente al proceso)		El Sistema muestra el mensaje “¿Está seguro que desea Modificar la Información de Proceso?”	Que exista conexión con la base de datos.
	En blanco	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema	Que exista conexión con la base de datos.

		marca con un asterisco los campos no válidos. El Sistema muestra los datos del tema seleccionado.	
SRHDJF (Palabra Clave)		El Sistema muestra el mensaje “¿Está seguro que desea Modificar la Información de Proceso?”	Que exista conexión con la base de datos.
Ddhfhsjfdhj		El Sistema muestra el mensaje “¿Está seguro que desea Modificar la Información de Proceso?”	Que exista conexión con la base de datos.
	Hhsb-23\$#%	El Sistema muestra el mensaje “Entrada de datos no válida” y el Sistema marca con un asterisco los campos no válidos. El Sistema muestra los datos del tema seleccionado.	Que exista conexión con la base de datos.
Se selecciona una entidad. (Selección de entidades)		El Sistema habilita la opción Modificar .	Que exista conexión con la base de datos.
	No se selecciona ninguna entidad	El Sistema no habilita la opción Modificar .	Que exista conexión con la base de datos.

4.3.1.7 Caso de prueba de integración CU _Gestión de Información de Procesos: sección Eliminar Información de Procesos

CU _Gestión de Información de Procesos

Entradas Válidas	Entradas Inválidas	Resultado Esperado	Condiciones
Clic en el botón Eliminar		El Sistema muestra el mensaje “¿Estás seguro que desea eliminar esta información de Proceso?”.	Que exista conexión con la base de datos.
Se selecciona una entidad		El Sistema habilita la opción Eliminar .	
	No se selecciona ninguna entidad	El Sistema no habilita la opción Eliminar .	Que exista conexión con la base de datos.

4.1 Conclusiones

Con la conclusión de este capítulo logramos definir la organización en términos de los componentes organizados en capas, además se organizaron los nodos específicos en un modelo de despliegue logrando integrar los resultados en un Sistema ejecutable. Dichos componentes fueron sometidos a pruebas de integración documentando los defectos que puedan afectar la calidad del software y llegando a la conclusión que los requisitos fueron implementados correctamente.

CONCLUSIONES

Con el desarrollo del Módulo de Servicio de Información se logró habilitar una vía para brindar información a la población de los servicios que realiza el banco, la cual permitirá una mayor rapidez y certeza para la toma de decisiones

Al finalizar el presente trabajo se dan por cumplidos los objetivos planteados en sus inicios; pues hemos logrado una realización eficiente de los procesos involucrados obteniendo el diseño de un producto informático en el que se aplican la metodología RUP, UML para el modelado, C# como un lenguaje de programación.

El desarrollo de esta aplicación constituye un aporte práctico muy importante, debido a que es una novedad tecnológica que marca un proceso de avance en el desarrollo de la informatización de la sociedad que favorece el aumento en la eficiencia de la gestión de la información de los procesos del Banco con tiempo de respuesta mínimo ante una solicitud, permitiendo a su vez tener toda la información centralizada.

RECOMENDACIONES

Hechas las conclusiones del trabajo, se recomienda:

- Implantar el sistema en el resto del país.
- Dar continuidad al ciclo de desarrollo número dos para que puedan ser pagados a través de la modalidad de Telebanca otros servicios.
- Continuar con la investigación para garantizar nuevas y buenas mejoras en futuras versiones del sistema.

BIBLIOGRAFÍA

1. AYUDA. Disponible en: <http://www.entelcallcenter.cl/ayuda/#1>
2. Call Center. Disponible en:
<http://www.cetursa.es/Te%20ofrecemos/Central%20de%20Reservas/Call%20Center>
3. Call Center ETECSA. 2003. [Disponible en: <http://www.etecsa.cu/servicios.asp?codigo=89&padre=89>
4. Crystal Reports 10 .NET Disponible en: <http://www.willydev.net/CrystalDesde0/>
5. GUERRERO, L. A. *CC51H* - Programación Orientada al Objeto, Clase 23 - Proceso Unificado de Desarrollo de Software. Disponible en: <http://www.dcc.uchile.cl/~luguerre/cc51h/clase23.html>
6. Información básica sobre tecnología ASP.NET. Disponible en:
<http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/vbcon/html/vbconASPTechnologyBackground.asp>
7. Introducción a ASP.NET. Disponible en:
<http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/cpguide/html/cpconIntroductionToASP.asp>
8. Introducción a Visual SourceSafe Disponible en: [http://msdn2.microsoft.com/es-es/library/3h0544kx\(VS.80\).aspx](http://msdn2.microsoft.com/es-es/library/3h0544kx(VS.80).aspx) (24/01/2007)
9. PERALTA, M. Sistema de Información. Disponible en:
<http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>
10. UML - RUP. Disponible en: <http://www.apexnet.com.ar/index.php/html/main/up.html>
11. UML, que es? Disponible en: <http://www.3wstudio.com.ar/index.php?mes=2&anio=2007>

12. Visual Studio .NET Professional. Disponible en: <http://www.abox.com/productos.asp?pid=314>
13. Aplicación web. Disponible en: http://es.wikipedia.org/wiki/Aplicacion_web
14. EL TELETRABAJO. Disponible en: <http://www.fortunecity.com/business/moo/955/teletrabajo.htm>
15. ¿QUÉ ES HTML? Disponible en: http://www.espaweb.com/respuestas_online/HTML.html
16. ¿Qué es ODBC? Disponible en: http://www.teklynx.com/mx/support/faq/faq_183.html
17. Recomendaciones para formularios Windows Forms y Web Forms. Disponible en: <http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/vbcon/html/vbtskchoosingbetweenwinformswebforms.asp>
18. Situación actual y perpestivas del Comercio Electrónico en Cuba. [Seminario]. 2002. [Disponible en: Teleformación. Disponible en: <http://prometeo.us.es/teleformacion/caracteristicas/marcotelefor.htm>
19. FERNANDEZ, P. G. ¿Que es son el lenguaje MSIL, el CLR y el compilador JIT? Disponible en: http://www.devjoker.com/asp/ver_contenidos.aspx?co_contenido=74
20. GUERRERO, F. G. Guía de implementación de .NET. Disponible en: http://manual.emagister.com/frame.cfm?id_user=36931042392123177021228415950659&id_centro=27422070033149505256526748524548&id_curso=52894020050265506657696750554555&url_frame=http://www.microsoft.com/latam/technet/articulos/200303/art02/
21. GUERRERO, L. A. CC51H - Programación Orientada al Objeto, Clase 23 - Proceso Unificado de Desarrollo de Software. Disponible en: <http://www.dcc.uchile.cl/~luguerre/cc51h/clase23.html>
22. LÓPEZ, C. La importancia de la información. Disponible en:

<http://www.gestiopolis.com/canales/demarketing/articulos/no8/info1.htm>

23. PERALTA, M. Sistema de Información. Disponible en:

<http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>

24. PURSNANI, V. Introducción a la Programación de Java Servlets. Disponible en:

<http://www.acm.org/crossroads/espanol/xrds8-2/servletsProgramming.html>

25. QUIJADO, J. VBScript. Disponible en: <http://www.programacion.net/asp/tutorial/vbscript/1/>

26. RISCO, E. G.-L. Qué es JAVA Script ?, 1998. [Disponible en:

<http://www.pablin.com.ar/computer/cursos/cursojs/js1.htm>

27. SECO, J. A. G. El lenguaje de programación C#, 2001. [Disponible en:

<http://programacion.com/tutorial/csharp/3/>

28. TOBAL, J. Java. Disponible en: <http://www.arrakis.es/~tobal/java.htm>

29. WOLTER, R. Fundamentos de los servicios Web XML, 2001. [Disponible en:

http://www.microsoft.com/spanish/msdn/articulos/archivo/151102/voices/fundamentos_xml.asp

GLOSARIO

Actor del Sistema: no son parte de él, pueden intercambiar información con él. Pueden ser un recipiente pasivo de información. Pueden representar el rol que juega una o varias personas, un equipo o un sistema automatizado.

Agenda Electrónica: como su nombre lo indica, no es más que una agenda que contiene todos los datos necesarios de las entidades.

Byte: es la unidad básica de almacenamiento de información, generalmente equivalente a ocho bits.

C: es un lenguaje de programación creado en 1969 por Ken Thompson y Dennis M. Ritchie en los Laboratorios Bell. C es apreciado por la eficiencia del código que produce y es el lenguaje de programación más popular para crear software de Sistemas, aunque también se utiliza para crear aplicaciones.

C++: es un lenguaje de programación, diseñado a mediados de los años 1980, por Bjarne Stroustrup, como extensión del lenguaje de programación C.

Caso de Uso: los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del Sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema.

Cuentas de Ahorro asociadas a Tarjetas Débito RED del Banco Metropolitano: Cuentas de Ahorro abiertas por el Cliente en bancos emisores de Tarjetas Débito RED.

Cliente: persona que utiliza el servicio brindado por las Empresas (En este caso es el que accederá al pago del mismo a través de la Banca Telefónica).

Delphi: es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión moderna de Pascal. Soporte para la programación orientada a objetos.

Entidad: empresa y centros asociados al Sistema Bancario Nacional.

Formularios Web Forms: los formularios Web Forms ASP.NET se utilizan para crear aplicaciones en las cuales la interfaz primaria de usuario es un explorador. Naturalmente, entre ellas se incluyen las aplicaciones que se ponen a disposición del público como por ejemplo, las aplicaciones de comercio

electrónico. Una característica importante es que no hay costos de distribución, puesto que los usuarios tienen ya instalada la única parte de la aplicación que necesitan: el explorador.

Los formularios Web Forms ofrecen algunas características que resultan útiles incluso en contextos que no son Web. Dado que están basados en HTML, resultan adecuados para aplicaciones que hacen un uso intensivo de recursos de texto, especialmente aquéllas en las que el formato del texto es importante.

HTML: es el acrónimo de HyperText Markup Language. Es el lenguaje más utilizado para la presentación de textos estructurados en formato de hipertexto, estándar de las páginas web.

Interbase: es un Sistema de Administración de Base de Datos Relacionales desarrollada y comercializada por la compañía Borland Software Corporation y actualmente desarrollada por su filial.

Internet Information Server (IIS): es una serie de servicios para los ordenadores que funcionan con Windows.

Interfaz grafica de usuario: en informática, tipo de visualización que permite al usuario elegir comandos, iniciar programas y ver listas de archivos y otras opciones utilizando las representaciones visuales (iconos) y las listas de elementos del menú. Las selecciones pueden activarse bien a través del teclado o con el ratón.

Java Servlets: los Servlets son componentes del servidor. Estos componentes pueden ser ejecutados en cualquier plataforma o en cualquier servidor debido a la tecnología Java que se usa para implementarlos. Los Servlets incrementan la funcionalidad de una aplicación web. Se cargan de forma dinámica por el entorno de ejecución Java del servidor cuando se necesitan. Cuando se recibe una petición del cliente el Servlet procesa la petición del cliente y envía la respuesta de vuelta que es enrutada al cliente.

Java: es un lenguaje de propósito general orientado a objetos que tiene una sintáxis similar (más sencilla) al C++.

Java Script: es un lenguaje interpretado, multiplataforma, orientado a eventos con manejo de objetos, cuyo código se incluye directamente en el mismo documento, usado para el desarrollo de aplicaciones cliente-servidor en páginas HTML.

MSIL (Microsoft Intermediate Language): es un language intermedio común a todos los Sistemas operativos que soporten .net framework. Es el único código que es capaz de interpretar el CLR, y por tanto cuando se dice que un compilador genera código para la plataforma .NET lo que se está diciendo es que genera MSIL.

Multiplataforma: es un término usado para referirse a los programas, Sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

Open Source: significa que es posible para cualquiera usar y modificar el software. Cualquiera puede bajar el software MySQL desde Internet y usarlo sin pagar nada. Si lo desea, puede estudiar el código fuente y cambiarlo para adaptarlo a sus necesidades.

ODBC (Open DataBase Connectivity): Conexión Abierta de Base de Datos, es un método estándar para compartir datos entre bases de datos y otros programas. Los Controladores ODBC usan el Lenguaje Estructurado de Consultas estándar (SQL) para acceder a datos de fuentes externas a la aplicación.

Plataforma: es una combinación de hardware y software usada para ejecutar aplicaciones.

Pago de Servicios: pago de un cliente a una Empresa por el servicio recibido o deuda contraída con la misma.

Requisito Funcional: Los requerimientos funcionales son capacidades o condiciones que el Sistema debe cumplir.

Requisito No Funcional: Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener.

Servicios Web Xml: son los bloques de construcción básicos en la transición al proceso distribuido en Internet. Proporcionan un modo de describir sus interfaces con suficiente detalle para permitir a un usuario construir una aplicación cliente para hablar con ellos.

Sistema Bancario Nacional: bancos que prestan servicios a la población y a las entidades que brindan servicios.

SQLite: es un Sistema de gestión de bases de datos relacional. Es un proyecto de dominio público creado por D. Richard Hipp. Es muy rápido y la ventaja fundamental es que permite utilizar un amplio subconjunto del lenguaje estándar SQL. SQLite se destaca también por su versatilidad.

Teleformación: tipología de Educación a Distancia que toma auge con la irrupción de Internet en nuestras vidas, al aumentar las posibilidades de acceso a la formación en la medida en que la Red nos va permitiendo llegar a más personas, y -sobre todo- ofrecer ambientes de aprendizaje más complejos y elaborados.

Teletrabajo: es una forma flexible de organización del trabajo que consiste en el desempeño de la actividad profesional sin la presencia física del trabajador de la empresa durante una parte importante de su horario laboral.

TeraByte: es una unidad de medida informática cuyo símbolo es el TB, y es equivalente a 1099511627776 bytes.

VBScript: es un lenguaje de script, directamente derivado de Visual Basic y el JavaScript. Los lenguajes de script son versiones recortadas de otros lenguajes. Estas versiones se usan para su integración en páginas web. Un código escrito en un lenguaje de script se incorpora directamente dentro de un código HTML.

Visual Basic: Desarrollado por Alan Cooper para Microsoft, Su primera versión fue presentada en 1991. Es un lenguaje de programación que se ha diseñado para facilitar el desarrollo de aplicaciones en un entorno grafico (GUI-GRAPHICAL USER INTERFACE) Como Windows 98, Windows NT o superior.

Xml (Extensible Markup Language): es un Lenguaje de Etiquetado Extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Su función principal es describir datos y no mostrarlos. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones.