

**Universidad de las Ciencias Informáticas**

**Facultad 2**



Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas

**Título: “Sistema de Administración de la Plataforma de Servicios  
de Mensajería SMS PLASERV”**

**Autor(es):** Zussel Fernández Jo

Santiago Rodríguez Suarez

**Tutor(es):** Ing. Pedro Alejandro Miralles Lorenzo

**Co-tutor(es):** Ing. Darién Jesús Álvarez de la Cruz

**Ciudad de la Habana, junio 2010**

**Declaración de Autoría**

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas para que haga el uso que estime pertinente con el mismo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año 2010.

**Autores:**

Zussel Fernández Jo.

Santiago Rodríguez Suarez.

\_\_\_\_\_

Firma

\_\_\_\_\_

Firma

**Tutor:**

Ing. Pedro Alejandro Miralles Lorenzo.

\_\_\_\_\_

Firma

*La constancia es la virtud por la que todas las cosas dan su fruto.*

*Arturo Graf*

### Agradecimientos

*Quiero agradecer en primer lugar a mi mamá y mi papá, por su apoyo y comprensión. A mi familia por estar siempre pendientes de mis resultados. A mis primas Channy y Yoa, por ser mis hermanas. A mi novio, compañero de tesis y futuro esposo, por aguantarme en mis momentos difíciles y darme fuerzas para llegar al final. A mis amigos dentro y fuera de la UCI, por su preocupación. A Daro y Yania, mis grandes amigas, por escucharme y estar siempre presentes cuando las necesito.*

*Zussel*

*A mi familia, por haberme educado, y guiado siempre. A mi novia, compañera de tesis y futura esposa y a mis suegros. A mis amigos dentro y fuera de la UCI, a Yoanny y los negros de Santiago, al piquete del grupo de primer año y al 2503.*

*Santiago*

*A los profesores, Darien, Frank, Joan y a Pedro, nuestro tutor, por su ayuda. A Aurelio, Maikel, Miguel, Manfred y a los demás del proyecto COMCEL. A la Revolución y a Fidel.*

*Zussel y Santiago*

Dedicatoria

*A mi familia, mis amigos y mi novio.*

*Zusset*

*A mi familia, mis amigos y mi novia.*

*Santiago*

## Resumen

Los teléfonos celulares han revolucionado el área de las comunicaciones, redefiniendo cómo se perciben las comunicaciones de voz, se han convertido en una herramienta primordial para la gente común y de negocios y aunque su principal función es la comunicación de voz, su rápido desarrollo ha incorporado funciones como juegos, reproducción de música MP3, agenda electrónica, fotografía y video digital, navegación por Internet, SMS (**S**hort **M**essage **S**ervice) entre otras. A la luz del éxito de los SMS, el mundo empresarial lo ha adoptado como un vehículo idóneo para comunicarse con sus usuarios y clientes.

En Cuba se hacen grandes esfuerzos para el progreso de la telefonía móvil, ya que el país no está ajeno a este creciente desarrollo. En la Universidad de las Ciencias Informáticas se obtienen resultados en esta esfera con el desarrollo de proyectos productivos y se está desarrollando una plataforma de servicios de mensajería basada en SMS; la cual soportará múltiples operadoras, permitirá crear servicios específicos basados en mensajes SMS y controlará tanto la gestión de servicios a los clientes, como el envío y recepción de información desde y hacia los terminales móviles.

Este trabajo tiene como objetivo desarrollar un sistema de administración para esta plataforma que permita explotar las funcionalidades brindadas por ella, incorporando una herramienta de configuración visual para la creación de los servicios facilitando este proceso.

### **Palabras claves:**

Plataforma de servicios SMS, servicios de mensajería, SMS, sistema de administración telefonía celular.

## Índice

Introducción .....	1
Capítulo 1 Fundamentación Teórica.....	4
Introducción.....	4
1.1.1    Telefonía móvil.....	4
1.1.2    SMS .....	4
1.1.3    Servicios de Valor Añadido .....	5
1.2    Estado del arte.....	5
1.2.1    En el mundo.....	5
1.2.2    En Cuba.....	7
1.3    Metodología de desarrollo RUP .....	9
1.4    Lenguaje de modelado UML (Unified Modeling Language) .....	9
1.5    Herramienta de modelado Visual Paradigm 6.1 .....	10
1.6    Lenguaje de programación Java .....	10
1.7    Plataforma de desarrollo JEE(Java Enterprise Edition).....	11
1.8    Herramienta de desarrollo Eclipse Galileo 3.5.....	11
1.9    Frameworks .....	11
1.9.1    GWT 1.7.1 (Google Web Toolkit).....	11
1.9.2    Spring 3.0.....	14
1.9.3    Spring Web Services 1.8 (Spring-WS).....	14
1.10    Herramienta de control de versiones SVN.....	15
Conclusiones.....	15
Capítulo 2 Características del Sistema.....	16
Introducción.....	16
2.1    El objeto de estudio .....	16
2.1.1    Objeto de automatización.....	16
2.2    Propuesta del sistema .....	16
2.3    Modelo de Dominio .....	17
2.3.1    Descripción de los conceptos de dominio.....	18
2.3.2    Descripción del modelo de dominio .....	19

2.4	Especificación de los requerimientos funcionales y no funcionales del sistema .	19
2.4.1	Requerimientos funcionales .....	19
2.4.2	Requerimientos no funcionales .....	21
2.5	Modelo de casos de uso del sistema .....	21
2.5.1	Definición de los actores del sistema.....	21
2.5.2	Diagrama de casos de uso del sistema.....	22
2.5.3	Descripción de los casos de uso.....	22
	Conclusiones.....	55
Capítulo 3 Análisis y diseño .....		56
	Introducción.....	56
3.1	Modelo de diseño.....	56
3.1.1	Diagrama de clases del diseño .....	56
3.2	Definiciones de diseño que se aplican .....	58
3.2.1	Arquitectura.....	58
3.2.2	Patrones de Casos de Uso.....	59
3.2.3	Patrones de diseño.....	59
3.3	Diagramas de interacción.....	61
	Conclusiones.....	61
Capítulo 4 Implementación y prueba .....		62
	Introducción.....	62
4.1	Modelo de implementación.....	62
4.1.1	Diagrama de despliegue.....	62
4.1.2	Diagrama de componentes .....	62
	Conclusiones.....	64
Capítulo 5 Estudio de la factibilidad: .....		65
	Introducción.....	65
5.1	Estimación por Puntos de Casos de Uso.....	65
5.1.1	Cálculo de Puntos de Casos de Uso sin ajustar.....	65
5.1.2	Cálculo de Puntos de Casos de Uso ajustados.....	66
5.1.3	Estimación del esfuerzo .....	69
5.1.4	Estimación del costo.....	71
5.2	Análisis de costos y beneficios .....	72
	Conclusiones.....	72



Conclusiones generales .....	73
Recomendaciones .....	74
Referencias Bibliográficas .....	75
Anexos .....	<b>¡Error! Marcador no definido.</b>
Anexo 1. ....	<b>¡Error! Marcador no definido.</b>
Anexo 2. ....	<b>¡Error! Marcador no definido.</b>

## Introducción

Las tecnologías inalámbricas han tenido un gran auge en estos últimos años. Siendo una de ellas la telefonía celular. Desde sus inicios a finales de los 70 ha revolucionado enormemente las actividades que se realizan diariamente. El teléfono móvil es un dispositivo inalámbrico electrónico que permite tener acceso a la red de telefonía celular o móvil. Se denomina celular debido a las antenas repetidoras que conforman la red, cada una de las cuales es una célula. Los teléfonos celulares se han convertido en una herramienta primordial para la gente común y de negocios; las hace sentir más seguras y productivas. Aunque su principal función es la comunicación de voz, como el teléfono convencional, su rápido desarrollo ha incorporado otras funciones como juegos, reproducción de música MP3 y otros formatos, correo electrónico, agenda electrónica PDA<sup>1</sup>, fotografía y video digital, navegación por Internet, SMS<sup>2</sup> entre otras.

El servicio de mensajes cortos o SMS provee un medio para el envío de mensajes con tamaño limitado de 160 caracteres desde y hasta terminales móviles. Hoy en día se han convertido en un canal de comunicación plenamente implantado en la sociedad, con las mismas ventajas que ofrece la red de telefonía móvil pero con un coste muy reducido y acotado.

A la luz del éxito de los SMS, el mundo empresarial también lo ha adoptado como un vehículo idóneo para comunicarse con sus usuarios y clientes. En todos los sectores se han identificado necesidades de comunicación que son cubiertas por el SMS con eficacia y eficiencia. También se han desarrollado servicios de valor añadido que utilizan SMS sobre tarificados (conocidos como Premium) para costear la prestación de los servicios o para suministrar contenidos de pago. Uno de los más conocidos es la participación en sorteos de TV, enviando un SMS a un número determinado de teléfono, lo que te asigna una "papeleta" para poder ganar un premio.

En Cuba se hacen grandes esfuerzos para el progreso de la telefonía móvil, ya que el país no está ajeno a este creciente desarrollo. La Universidad de las Ciencias

---

<sup>1</sup> Del inglés *Personal Digital Assistant* (*Asistente Digital Personal*).

<sup>2</sup> *Short Message Service*.

Informáticas es una universidad productiva cuya misión es producir software y servicios informáticos a partir de la vinculación estudio – trabajo, la producción se basa en la integración de los procesos de formación, investigación y producción en torno a una temática para convertirla en una rama productiva, destacándose resultados en la esfera de la telefonía móvil [1]. Se está desarrollando una plataforma de servicios de mensajería basada en SMS; la cual permitirá crear servicios específicos basados en mensajes SMS y controlará tanto la gestión de servicios a los clientes, como el envío y recepción de información desde y hacia los terminales móviles, siendo necesario que los usuarios puedan explotar las funcionalidades brindadas por esta plataforma.

Teniendo en cuenta la situación antes señalada, se formula el siguiente **problema a resolver**: ¿Cómo lograr que los usuarios de la plataforma de servicios de mensajería basada en SMS exploten las funcionalidades brindadas por esta?

Esta tesis tiene como **objeto de estudio**: Las plataformas de servicios de mensajería basadas en SMS.

Como **campo de acción**: Los sistemas de administración de plataformas de servicios de mensajería basadas en SMS.

Dada la problemática tratada y debido a la necesidad que impera de darle solución a la misma, se ha decidido cumplir con el siguiente **objetivo**: Desarrollar un sistema de administración que permita explotar las funcionalidades brindadas por una plataforma de servicios de mensajería basada en SMS.

Los objetivos específicos que se persiguen son:

- Gestionar los clientes, empleados, operadoras y canales de la plataforma.
- Crear servicios de valor agregado basados en SMS.
- Modelar la estructura y funcionamiento de los servicios.

La **idea a defender** que se plantea es: el desarrollo de un sistema de administración para la plataforma de servicios de mensajería basada en SMS permitirá explotar las funcionalidades brindadas por esta.

Para poder cumplir con el objetivo se proponen las siguientes **tareas de investigación**:

- Seleccionar la metodología, herramientas y lenguaje de programación a utilizar en la solución.

- Identificar e implementar buenas prácticas de diseño para el diseño de sistemas de administración.
- Analizar el estado a nivel mundial de los sistemas de administración de plataformas de servicios SMS.
- Analizar el funcionamiento de los servicios SMS en plataformas propietarias (ej.BlueEye).

El trabajo está estructurado en 5 capítulos que se describen brevemente a continuación:

**Capítulo 1:** Contiene el estado del arte a nivel internacional y nacional del tema tratado, así como las metodologías, lenguajes de programación y herramientas que serán utilizadas en el desarrollo de la aplicación.

**Capítulo 2:** En este capítulo se definen las características del sistema, se propone una solución a través de una descripción detallada de un modelo de dominio y los requisitos funcionales y no funcionales de la aplicación.

**Capítulo 3:** Este capítulo se enfoca en el diseño del sistema. Se modelan los diagramas de clases del diseño para cada caso de uso del sistema, así como los diagramas de interacción, específicamente los diagramas de secuencia. Se especifican y describen los patrones utilizados.

**Capítulo 4:** En este capítulo se desarrolla el flujo de trabajo de implementación y prueba. Se confeccionan los diagramas de componentes y de despliegue, dando una visión de cómo quedará desarrollada y distribuida la aplicación.

**Capítulo 5:** Este capítulo muestra un estudio de la factibilidad del software a desarrollar, se realiza una estimación del esfuerzo, costo y tiempo de desarrollo del mismo mediante el método de Puntos de Casos de Uso.

## Capítulo 1 Fundamentación Teórica.

### Introducción

En el presente capítulo se describen los principales aspectos y conceptos que han sido objeto de análisis a lo largo de la investigación, se definen las herramientas, lenguajes y metodologías que serán utilizados para el desarrollo de la aplicación.

#### 1.1.1 Telefonía móvil

La telefonía móvil o también llamada telefonía celular es un sistema de comunicación telefónica totalmente inalámbrica. Se denomina celular debido a las antenas repetidoras que conforman la red, cada una de las cuales es una célula. La comunicación telefónica es posible gracias a la interconexión entre centrales móviles y públicas. La telefonía móvil consiste en la combinación de una red de estaciones transmisoras-receptoras de radio y una serie de centrales telefónicas de conmutación que posibilita la comunicación entre terminales telefónicos portátiles (teléfonos móviles) o entre terminales portátiles y teléfonos de la red fija tradicional [2].

#### 1.1.2 SMS

El servicio de mensajes cortos o SMS es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos (también conocidos como mensajes de texto) entre teléfonos móviles, con tamaño limitado de 160 caracteres. Los mensajes cortos hacen un uso extremadamente eficaz de la red de radio, y además pueden ser enviados y recibidos en cualquier momento, incluso durante una llamada. Debido a que los mensajes SMS son recibidos prácticamente de inmediato por el destinatario y son un medio de comunicación muy personal, muchos ya los están utilizando como el mejor medio para comunicarse con una comunidad para invitar a eventos, dar avisos, enviar alarmas, coordinar evacuaciones, confirmar transacciones bancarias, enviar confirmaciones de compra entre otras [3].

## 1.1.3 Servicios de Valor Añadido

Valor agregado o valor añadido es una característica o servicio extra que se le da a un producto o servicio, con el fin de darle un mayor valor comercial, generalmente se trata de una característica o servicio poco común, o poco usado por los competidores, y que le da al negocio o empresa, cierta diferenciación. [4].

## 1.2 Estado del arte

### 1.2.1 En el mundo

En el mundo existen numerosas empresas que cuentan con plataformas dedicadas a prestar servicios de valor agregado para la telefonía móvil basados en SMS, siendo estos uno de los más utilizados hoy en día. Se quiere desarrollar un sistema de administración para estas plataformas, de manera que los usuarios puedan explotar fácilmente las funcionalidades que brindan. Para ello se incluirá en el sistema una herramienta de configuración visual para la creación de los servicios de la forma más fácil y cómoda posible para el usuario, sin que este cuente con conocimientos avanzados en informática.

### ALTIRIA

Pone a disposición de las empresas una plataforma tecnológica que permite la utilización de distintas tecnologías para el contacto con los clientes. Tiene incluida una plataforma SMS que integra su propia plataforma SMS Premium que permite crear servicios específicos basados en mensajes SMS, algunos predefinidos y contando con la capacidad de desarrollar servicios a medida para cada cliente [5].

A continuación se listan algunos productos disponibles en esta plataforma:

- Sorteos y concursos: realización de promociones y sorteos empleando mensajes SMS como canal de participación.
- Votación con SMS: La plataforma SMS permite realizar sondeos y votaciones con gestión automatizada de resultados en tiempo real.

### **NudBox**

Nudbox actúa como plataforma tecnológica y medio de pago para el envío de SMS Premium, ofreciendo una serie de servicios con los que se puede percibir una alta rentabilidad. Tiene un sistema de estadísticas, que permite ver las estadísticas por períodos predefinidos (mensuales, diarios y personalizados). Cualquier usuario puede optar a solicitar en sus propios servicios los alias que quieran asignar interactuando en tiempo real e intuitivamente con la plataforma, comunicándole al cliente si está ocupado, si contiene un caracter o palabra inválida o sufre algún otro problema por el cual no es válido ese alias. La plataforma cuenta con algunas herramientas como [6]:

- Sistema de monitorización de servicios, pudiendo comprobar el estado de todos los servicios
- Glosario, en el cual se tratan todas las palabras técnicas del sistema
- Simular mensajes, para sin tener que enviar los mensajes poder probar el servicio
- Sistema de ayuda con las preguntas frecuentes
- Editor de datos personales incluido los de facturación directamente desde el sistema

### **Ekosoft**

Plataforma SMS Premium que da la posibilidad de recibir en tiempo real los SMS Premium que envíen los espectadores u oyentes, posibilidad de responder automáticamente o manual cada mensaje. Ofrece servicios de emisión de contenidos TV conectados en tiempo real a servidores SMS permitiendo un diseño personalizado para cada programa en exclusiva. Brinda servicios de sorteos, posibilitando ver todos los participantes del sorteo y personalizar el número de mensajes a enviar para participar. Brinda también servicios de votaciones viendo los resultados en pantalla en tiempo real, servicio de concursos permitiendo ver los ganadores que hayan respondido bien todas las preguntas, y servicios de campañas SMS [7].

## 1.2.2 En Cuba

Cuba, en su empeño por alcanzar un desarrollo social y económico ha dado los primeros pasos en el desarrollo de este tipo de servicio.

### BlueEye

BlueEye es una plataforma para el desarrollo y despliegue de aplicaciones y servicios de valor agregado que funcionan mediante mensajería corta (SMS).

En general, el escenario más común de uso consiste en que los suscriptores clientes de las aplicaciones BlueEye envían un mensaje texto (SMS) a un número corto de servicio, ofrecido mediante el operador celular local. La primera palabra de este SMS se considera una palabra clave, y en dependencia de la misma, el mensaje recibido por la plataforma es enrutado hacia una de las aplicaciones asociadas a esa palabra. La aplicación en cuestión procesa el pedido del usuario, cuyos detalles vienen en el SMS, y devuelve una respuesta apropiada vía SMS al suscriptor que solicitó el servicio específico inicialmente [8].

Las características principales de la plataforma son las siguientes:

- **Kit de desarrollo:** BlueEye es también un kit de desarrollo de aplicaciones que provee un API<sup>3</sup>, pensado para llevar rápidamente las nuevas ideas a la producción y ejecución sobre un despliegue existente de BlueEye.
- **Ejecución de múltiples aplicaciones:** Existe la habilidad de ejecutar y administrar múltiples aplicaciones sobre la plataforma.
- **Manejo de aplicaciones en caliente:** Es posible agregar o inhabilitar aplicaciones que estén desplegadas en producción sin interrumpir al resto y sin que se necesite tiempo fuera de servicio para ello.
- **Enrutamiento basado en palabras clave y/o números cortos:** Es posible configurar las combinaciones de palabras clave (y sus variantes) con los números cortos para decidir que SMS llegan a cada una de las aplicaciones manejadas en la instancia de BlueEye.

---

<sup>3</sup> *Application Programming Interface*



- **Incluida la lógica para servicios basados en suscripción:** Las aplicaciones que funcionan basadas en suscripción a servicios periódicos no tienen que implementar ellas mismas la lógica de ese modelo, sino que ya está lista como parte de BlueEye, y sólo tienen que hacer uso del convenio.
- **Rutas simultáneas y protocolos múltiples:** BlueEye es capaz de mantener comunicación simultánea con varios centros de mensajería en diversos operadores y otros proveedores, incluso haciendo uso de diversos protocolos de comunicación disponibles en cada uno de ellos. Es posible agregar nuevos protocolos mediante un driver sin afectar al resto del sistema. Utiliza y aprovecha estándares como SAMS (JSR-212) para este propósito.
- **Sistema abierto:** Es posible agregar nuevas funcionalidades que afectan y benefician a todas las aplicaciones, de manera que estas puedan hacer uso de ellas para implementar nuevos y más interesantes servicios.
- **Integrable a sistemas y servicios existentes:** Es posible adaptar y aplicar BlueEye para soluciones específicas o adaptarlo a nuevos modelos de negocio. Las aplicaciones basadas en BlueEye pueden hacer uso de sistemas externos como servicios públicos, bases de datos remotas, interfaces de operación de aplicaciones personalizadas, etc.
- **Fundación para aplicaciones más complejas:** BlueEye permite desarrollar aplicaciones cuyo motivo no sea la recepción y envío de mensajes, sino que esto sea un plus. Es posible desarrollar sitios web, portales de ventas de contenidos, sitios WAP<sup>4</sup>, o incluso aplicaciones J2ME o de escritorio que hagan uso de los servicios de BlueEye.
- **Administrable y auditable:** Es posible personalizar las interfaces de administración y monitorización del sistema, así como el formato de las trazas y reportes que se requiera.
- **Transaccionalidad en las operaciones:** Todas las operaciones dentro de la plataforma son transaccionales y garantizan la integridad y la completitud total o no de las operaciones.
- **Flexibilidad en el despliegue:** Es posible configurar un despliegue distribuido o concentrado. Cada aplicación puede tener su propia base de datos y servidor de

---

<sup>4</sup> *Wireless Application Protocol (Protocolo de aplicaciones inalámbricas)*

aplicaciones independiente.

- **Escalabilidad y tolerancia ante fallos:** Un despliegue de BlueEye permite la tolerancia ante fallos y la escalable tanto horizontal como verticalmente en caso de ser necesario.
- **Desarrollado y ejecutado con tecnologías de código abierto:** Durante el desarrollo y despliegue de BlueEye se utilizan solamente tecnologías de código abierto. El sistema operativo utilizado es Linux.
- **Tecnologías estándares y robustas:** BlueEye está basado en Java, EJB3, Hibernate y otras tecnologías que han probado su madurez en el mundo empresarial [8].

### 1.3 Metodología de desarrollo RUP

Para el modelado de los procesos la metodología a utilizar será la metodología Rational Unified Process (RUP) que es configurable y se adapta a través de los proyectos variados en tamaños y complejidad, siendo esta una característica fundamental ya que el equipo de desarrollo es pequeño y cuenta con poco tiempo. Es un proceso iterativo e incremental que con cada iteración se puede llevar al cliente una versión del producto con el fin de obtener mejoras, permitiendo probar la calidad durante la totalidad del ciclo de vida del proyecto. Contiene artefactos que son diseñados durante las diferentes fases, los cuales describen detalladamente las características del software desde que se realiza el análisis del problema hasta la entrega final del producto, lo que permite reconocer los problemas y fallos de forma temprana y corregirlos. Genera mucha documentación, lo que facilitará la implementación de futuras versiones del producto en caso de que el personal actual del equipo de desarrollo cambie. [9].

### 1.4 Lenguaje de modelado UML (Unified Modeling Language)

El Proceso Unificado RUP utiliza el Lenguaje de Modelado UML, para preparar todos los esquemas de un sistema de software. UML es una parte esencial del Proceso Unificado, está consolidado como el lenguaje estándar en el análisis y diseño de sistemas de cómputo. Posee más características visuales que programáticas facilitando a los integrantes del equipo participar y comunicarse fácilmente. Permite la modelación de sistemas con tecnología orientada a objetos, sin embargo, el análisis en UML se puede realizar independiente del lenguaje en el que finalmente sea implementado el sistema, por lo que será utilizado para la modelación del proyecto [10].

### 1.5 Herramienta de modelado Visual Paradigm 6.1

Es una herramienta CASE<sup>5</sup> que facilita el modelado de artefactos en un Proceso de Desarrollo de Software mediante el Lenguaje de Modelado UML. Soporta ingeniería inversa, generación de código a partir de los diagramas para diferentes plataformas, brinda la posibilidad de documentar todo el trabajo sin necesidad de utilizar herramientas externas en varios formatos como Web o Pdf, genera informes, edita detalles de Casos de Uso. La herramienta es colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto y permite control de versiones [11].

Teniendo en cuenta lo antes expuesto, es seleccionada esta herramienta para el modelado del proyecto, es multiplataforma y su principal ventaja es la licencia comercial que presenta ya que la Universidad de las Ciencias Informáticas cuenta con la misma para su uso. Cabe destacar su robustez, usabilidad y portabilidad.

### 1.6 Lenguaje de programación Java

El lenguaje de programación a utilizar será Java puesto que es un lenguaje de alto nivel, orientado a objetos, multiplataforma, es decir, que permite la ejecución de un mismo programa en múltiples sistemas operativos, es distribuido ya que contiene un conjunto de clases que permiten la comunicación a través de la red facilitando así la creación de aplicaciones distribuidas. Es un lenguaje robusto, pues fue diseñado para crear software altamente fiable. Puesto que Java surgió en la era post-Internet, fue diseñado con la idea de la seguridad y la fiabilidad por lo que se le integraron varias capas de seguridad para evitar que programas maliciosos pudiesen causar daños en los sistemas, garantizando así una mayor seguridad de la aplicación en desarrollo. Es libre y les proporciona a los desarrolladores una mayor flexibilidad, ahorro en los costes de desarrollo, independencia y compatibilidad con otros sistemas [12].

---

<sup>5</sup> *Computer Aided Software Engineering (Ingeniería de software asistida por ordenador)*

## 1.7 Plataforma de desarrollo JEE(Java Enterprise Edition)

Para desarrollar la aplicación fue seleccionada como plataforma **JEE**, ya que provee de grandes ventajas a todos los involucrados en el proceso de desarrollo de software. La Plataforma Java EE ofrece tecnología para el desarrollo de aplicaciones empresariales con distribución multicapa basadas en Web. Además, define una infraestructura común básica para el acceso a bases de datos, gestión de la persistencia, control de seguridad, y gestión de transacciones, ofreciendo una separación clara entre presentación (interfaz), modelo (lógica) de negocio y datos. Incluye varias especificaciones de API, tales como EJB<sup>6</sup>, Servlets (código en Java que se ejecuta en un servidor web), JSP<sup>7</sup> y varias tecnologías de servicios web. Hay muchas herramientas de código abierto, disponibles para extender la plataforma o para simplificar el desarrollo [13].

## 1.8 Herramienta de desarrollo Eclipse Galileo 3.5

Eclipse es un entorno de desarrollo integrado de código abierto, multiplataforma, para integrar herramientas de desarrollo, con una arquitectura abierta y basada en plug-ins, es soportado por los principales sistemas operativos, tiene además una comunidad de desarrolladores, realmente activa. Continuamente se están desarrollando nuevos plug-ins y revisando los anteriores. Permite la instalación de plug-ins destinados a mejorar las funcionalidades del propio IDE<sup>8</sup> y a extenderse en cada vez más tecnologías [14]. Por todas estas características fue seleccionado para el desarrollo del software Eclipse Galileo que es la versión 3.5 de este IDE presentada en junio del año 2009.

## 1.9 Frameworks

### 1.9.1 GWT 1.7.1 (Google Web Toolkit)

Es un framework de desarrollo en Java de código abierto, que permite escapar de la "matriz" de tecnologías usadas actualmente para escribir aplicaciones AJAX, las cuales son difíciles de manejar y propensas a errores. Con GWT, se pueden desarrollar y depurar aplicaciones AJAX usando el lenguaje de programación Java en el entorno de desarrollo de su preferencia. Al acabar la aplicación (escrita en Java),

---

<sup>6</sup> *Enterprise JavaBeans*

<sup>7</sup> *JavaServer Pages*

<sup>8</sup> *Integrated Development Environment(Entorno de desarrollo integrado)*

GWT compila y traduce dicho programa a JavaScript y HTML compatible con cualquier navegador web [15].

Al desarrollar en Java se puede utilizar los entornos de desarrollo más populares para Java como Eclipse, NetBeans. Se pueden realizar abstracciones y utilizar patrones de diseño orientados a objetos basados en Java. Mejora la experiencia del usuario, ya que al ejecutarse la interfaz de usuario en el navegador, la interacción del usuario con la aplicación es más fluida, al no ser necesario llamar al servidor para renderizar las páginas ya no tendremos el efecto de página en blanco. Obtenemos un mayor rendimiento y una mejor escalabilidad que con las aplicaciones webs tradicionales [16].

Fue seleccionado este framework debido a que presenta componentes de la interfaz de usuario dinámicos y reutilizables, tiene compatibilidad con los navegadores, es un proyecto de código abierto y existen un numeroso conjunto de bibliotecas desarrolladas por Google y terceros que amplían las funcionalidades de GWT.

Comunicación cliente – servidor del framework.

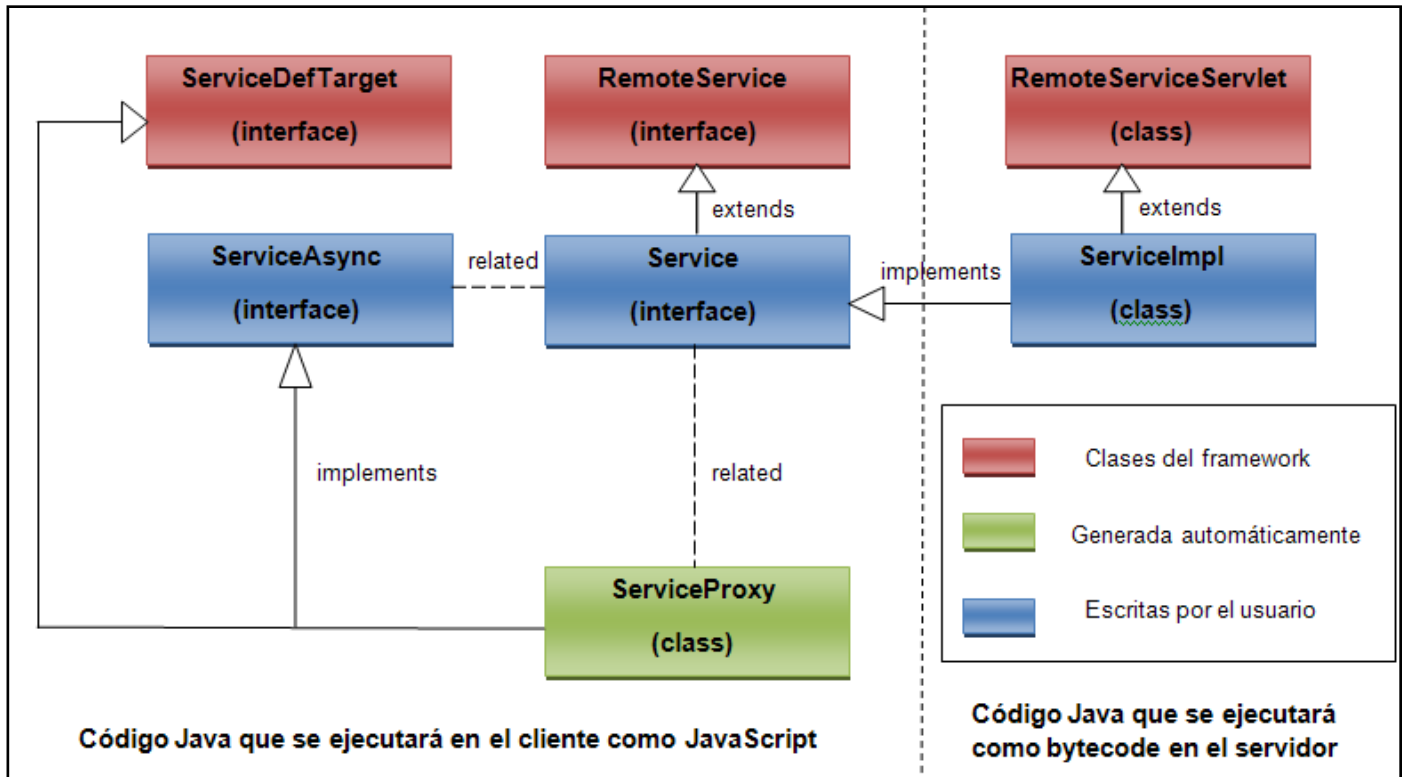


Figura 3.1. Componentes del mecanismo RPC de GWT.

El marco GWT RPC hace que sea fácil para los componentes cliente y servidor de la aplicación web intercambiar objetos a través de http. El código de cliente va a utilizar una clase de proxy automático para realizar llamadas al servicio (se refiere al código del servidor que se invoca desde el cliente). La implementación de un servicio GWT RPC se basa en la arquitectura de Servlet de Java conocida. GWT se encargará de la serialización de los objetos Java que iban y venían, las alegaciones de las llamadas en el método y el valor de retorno [17].

En la creación de un GWT RPC intervienen tres elementos:

- El servicio que se ejecuta en el servidor (el método que está llamando).
- El código de cliente que invoca el servicio.

- Los objetos de datos Java que pasan entre el cliente y el servidor.

Tanto el cliente como el servidor tienen la capacidad para serializar y deserializar datos.

Con el fin de definir la interfaz RPC deben crearse tres componentes:

- Definir una interfaz para su servicio que extiende de RemoteService y las listas de todos sus métodos de RPC.
- Crear una clase que extiende de RemoteServiceServlet e implementa la interfaz que ha creado anteriormente.
- Definir una interfaz asíncrona a su servicio para llamar desde el código de cliente.

RemoteServiceServlet maneja automáticamente la serialización de los datos pasados entre el cliente y el servidor e invocando el método previsto en la implementación del servicio [17].

### 1.9.2 Spring 3.0

Spring es un Framework de código abierto de desarrollo de aplicaciones para la plataforma Java que ha ido revolucionando la manera de programar promoviendo buenas prácticas de diseño e implementación, razones por las cuales fue seleccionado para desarrollar la aplicación. El mismo se encarga de construir clases una vez que se desee, inyectará las dependencias que ellas contengan, construyendo los objetos según sea necesario y de esta forma facilitar el desarrollo. Una de las grandes ventajas que posee Spring es que está diseñado como una serie de módulos que pueden trabajar independientemente uno de otro. Además, intenta mantener un mínimo acoplamiento entre la aplicación y el propio framework de forma que podría ser desvinculada de él sin mucha dificultad [18].

### 1.9.3 Spring Web Services 1.8 (Spring-WS)

Para la creación de servicios web se usa Spring Web Service, que es un producto de Spring Framework para facilitar la creación de servicios web basados en el intercambio de documentos. Spring-WS se basa en servicios en los cuales primero se establece un contrato y luego se implementa, evitando atar al contrato como sucede en los casos en los cuales se genera el mismo a partir de las clases Java, facilitando aplicar las mejores prácticas para la creación de servicios web.

## 1.10 Herramienta de control de versiones SVN

Para el control de versiones será utilizado SVN<sup>9</sup> que es un sistema de control de versiones de código abierto que mantiene los registros de todos los cambios que se han realizado a los archivos de un software, permitiendo la recuperación de versiones anteriores de la información que se trata o examinar el historial de cambios de dicha información. Puede operar a través de la red, lo cual permite su uso desde diferentes puestos de trabajo en un equipo de desarrollo.

Imperan dos razones fundamentales para el uso de esta herramienta en el desarrollo de la aplicación:

- Gestiona las modificaciones durante el desarrollo.
- Permite que varias personas trabajen sobre los mismos ficheros.

### Conclusiones

En este capítulo se realizó una descripción de algunas plataformas que existen en el mundo y en Cuba que se dedican a prestar servicios basados en SMS. Además, se hizo un análisis de las metodologías y herramientas a utilizar. Como resultado se determinó el uso de RUP como metodología de desarrollo y UML como lenguaje de modelado. Java como lenguaje de programación, Eclipse Galileo 3.5 como entorno de desarrollo y Visual Paradigm como herramienta de modelado.

---

<sup>9</sup> *Subversion*



### Capítulo 2 Características del Sistema.

#### **Introducción**

En el presente capítulo se describen las características del sistema a desarrollar, se realiza el proceso de captura de requisitos funcionales y no funcionales a los que se debe dar cumplimiento, así como la identificación de actores y casos de uso del sistema.

#### **2.1 El objeto de estudio**

Debido al avance de las tecnologías de la información y las telecomunicaciones se ha extendido el uso de la telefonía móvil. Hoy en día los teléfonos celulares se han convertido en una herramienta primordial en la vida cotidiana de las personas. Su rápido desarrollo ha incorporado, además de su principal función que es la comunicación de voz otras funciones como juegos, correo electrónico, fotografía y servicio de mensajes cortos (SMS). Existen plataformas que mediante este último brindan servicios de valor agregado, brindándole al cliente o usuario de cierto servicio información extra.

Se hace necesario el desarrollo de una aplicación que permita de forma visual interactuar con estas plataformas facilitando el proceso de creación de servicios.

##### **2.1.1 Objeto de automatización**

Serán objeto de automatización los procesos de crear, modificar y eliminar servicios en una plataforma de servicios de mensajería basada en SMS.

#### **2.2 Propuesta del sistema**

Se propone diseñar un sistema de administración con una interfaz sencilla y amigable que permita interactuar con la plataforma de servicios de mensajería basada en SMS Plaserv a través de servicios web. Este sistema estará dividido en dos módulos, el módulo de administración y el de diseño. Desde el módulo de administración se podrán gestionar los empleados que trabajarán con la plataforma, así como las operadoras y canales de comunicación. Desde el módulo de diseño se podrán gestionar los clientes que soliciten la creación de los servicios, así como la gestión de estos últimos. Para la creación de los servicios se le incorpora a este módulo una herramienta de configuración visual que facilitará este proceso. Esta herramienta brindará la posibilidad de diseñar una gran variedad de servicios, contando la misma con una barra de herramientas y un área de trabajo. La barra de herramientas está formada por diferentes componentes, los cuales serán seleccionados y arrastrados hacia el área de trabajo. Estos

componentes son configurables por el usuario y poseen diferentes funcionalidades dependiendo de su tipo, se enlazan unos con otros definiendo como va a ser la estructura del mensaje que se va a recibir o enviar (dependiendo del servicio) una vez que el usuario haga uso del servicio desde su móvil.

### **2.3 Modelo de Dominio**

En el sistema a desarrollar no se identifican con claridad los actores, trabajadores y procesos de negocio, por lo que se decide realizar un modelo de dominio el cual es un subconjunto del modelo de objeto del negocio donde se representan los conceptos y eventos fundamentales que se expresan como clases con la cardinalidad que existe entre ellas.

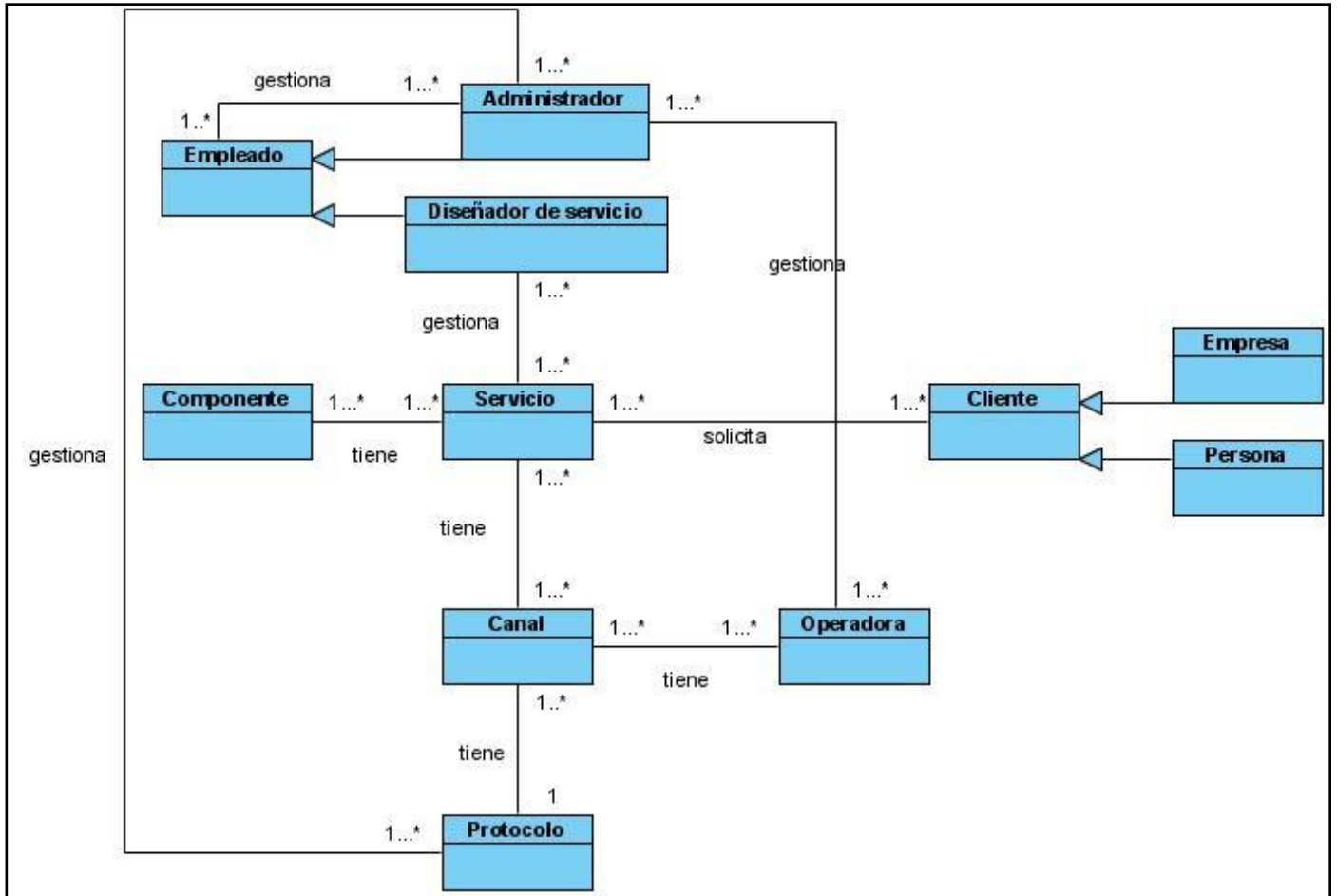


Figura 2.1: Modelo de dominio

### 2.3.1 Descripción de los conceptos de dominio

**Empleado:** Es quien va a realizar acciones sobre el sistema según el rol que tenga.

**Administrador:** Es la persona encargada de gestionar las operadoras y los protocolos.

**Diseñador de servicio:** Es la persona encargada de gestionar los servicios.

**Servicio:** Representa los diferentes servicios que pueden ser creados.

**Componente:** Es la estructura básica para modelar el servicio, es configurable por el diseñador.

**Cliente:** Es una generalización de empresa y persona.

**Empresa:** Representa una empresa cliente que solicita servicios.

**Persona:** Representa una persona cliente que solicita servicios.

**Operadora:** Es quien suministra los canales.

**Canal:** Medio por el cual se envían y reciben los SMS.

**Protocolo:** Se le asigna a diferentes canales para la comunicación.

### 2.3.2 Descripción del modelo de dominio

El empleado es el encargado de realizar acciones sobre el sistema según el rol que tenga, administrador o diseñador de servicios. En caso que el empleado tenga asignado el rol de diseñador de servicios puede entonces crear servicios a petición de un cliente que puede ser una persona o una empresa. Los servicios se modelan mediante componentes, tienen asignado un canal y este a su vez tiene asignado un protocolo de comunicación. Los canales son suministrados por operadoras. En caso que el empleado tenga asignado el rol de administrador puede gestionar las operadoras, los protocolos y a su vez gestionar los empleados.

## 2.4 Especificación de los requerimientos funcionales y no funcionales del sistema

A continuación se relacionan los requerimientos funcionales y no funcionales capturados para el desarrollo del sistema.

### 2.4.1 Requerimientos funcionales

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir y se mantienen invariables sin importar con que cualidad o propiedad se relacionen.

#### R1 Autenticar Usuario

Permite diferenciar el nivel de acceso, requiere de un usuario y una contraseña, esto permite definir los roles que desempeña cada uno en el sistema y a que información tiene derecho acceder y modificar.

#### R2 Gestionar Empleado

R2.1 Insertar empleado

R2.2 Modificar empleado

R2.3 Eliminar empleado

R2.4 Mostrar empleado

### **R3 Gestionar Servicio**

R3.1 Insertar servicio

R3.2 Modificar servicio

R3.3 Eliminar servicio

R3.4 Mostrar servicio

### **R4 Gestionar Cliente**

R4.1 Insertar cliente

R4.2 Modificar cliente

R4.3 Eliminar cliente

R4.4 Mostrar cliente

### **R5 Gestionar Operadora**

R5.1 Insertar operadora

R5.2 Modificar operadora

R5.3 Eliminar operadora

R5.4 Mostrar operadora

### **R6 Gestionar Canal**

R6.1 Insertar canal

R6.2 Modificar canal

R6.3 Eliminar canal

R6.4 Mostrar canal

### **R7 Modelar Servicio**

Modelar servicio mediante componentes visuales móviles y configurables por el usuario.

### 2.4.2 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, son las características que hacen al producto atractivo, usable, rápido y confiable.

#### Apariencia o Interfaz Externa

**R1** La aplicación propuesta poseerá una interfaz sencilla, amigable y fácil de utilizar, dirigida directamente al usuario del sistema.

#### Usabilidad

**R2** La aplicación podrá ser utilizada por usuarios con conocimientos básicos en computación.

#### Portabilidad

**R3** El sistema deberá ser multiplataforma.

#### Hardware

**R4** Se necesitará como requerimientos mínimos una PC con velocidad de procesamiento de 2.00 GHz y una memoria RAM de 1GB.

#### Seguridad

**R5** El sistema controlará los diferentes niveles de acceso, prioriza la identificación del usuario antes de que sea capaz de realizar cualquier acción sobre el sistema.

#### Soporte

**R6** El sistema contará con un manual para los usuarios donde se expone como debe ser el trabajo con la aplicación.

### 2.5 Modelo de casos de uso del sistema

El diagrama de casos de uso del sistema representa gráficamente los procesos y su interacción con los actores.

#### 2.5.1 Definición de los actores del sistema

Actor	Justificación
Empleado	Representa una generalización de los actores del sistema y es el encargado de autenticarse.

Administrador	Es el encargado de gestionar empleados, operadoras y canales.
Diseñador de servicios	Es el encargado de gestionar servicios y clientes.

Tabla 2.1: Descripción de los actores del sistema.

### 2.5.2 Diagrama de casos de uso del sistema

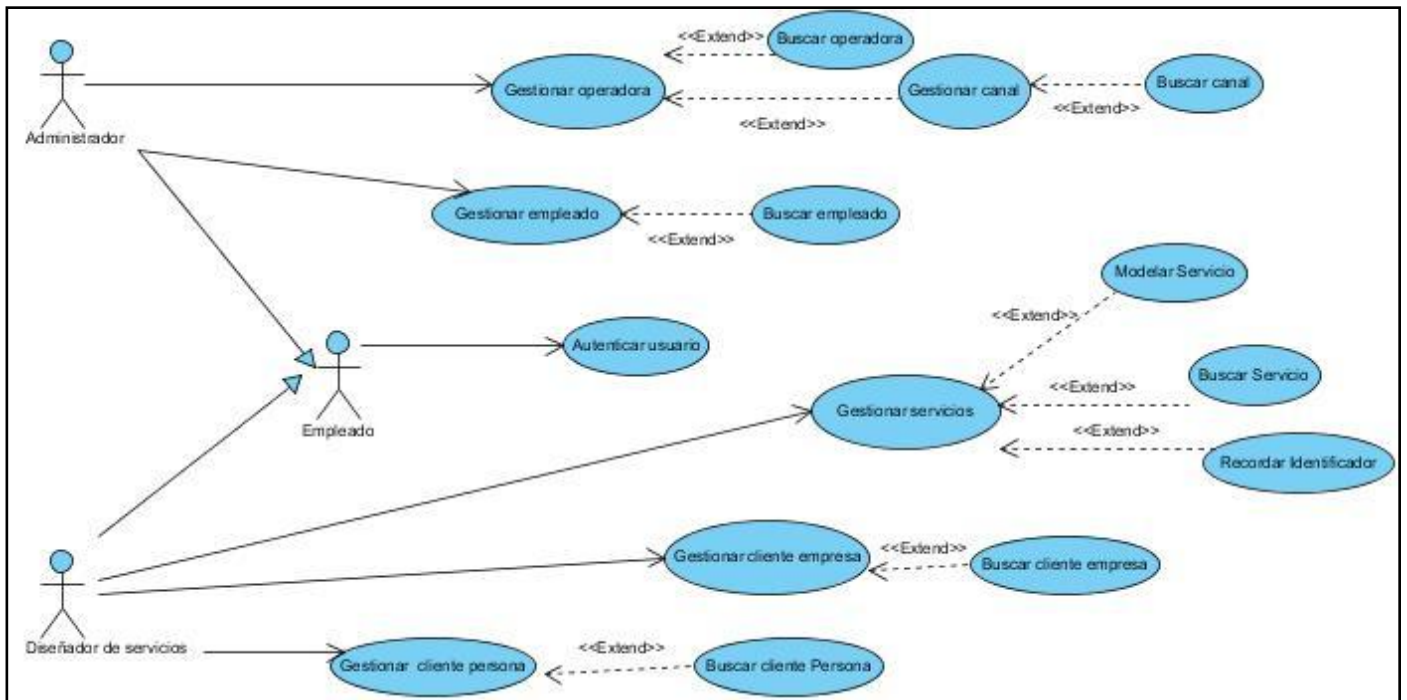


Figura 2.2: Diagrama de casos de uso del sistema

### 2.5.3 Descripción de los casos de uso

<b>Nombre del Caso de Uso</b>	<b>Autenticar usuario</b>	
<b>Actores</b>	Empleado	
<b>Propósito</b>	Permitir que el empleado se autentique.	
<b>Resumen</b>	Es aquí donde el empleado se autentica.	
<b>Referencias</b>		
<b>Precondiciones</b>		
<b>Pos condiciones</b>	El empleado ya autenticado.	
<b>Sección "Principal"</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El empleado, inserta en los campos correspondientes el usuario y la contraseña y a continuación presiona el botón Enviar.	<p>1.1 El sistema verifica que los campos no estén vacíos.</p> <p>1.2 El sistema comprueba que los datos tanto del usuario como la contraseña existan en la base de datos.</p> <p>1.3 El sistema muestra un mensaje indicando que la autenticación se ha efectuado satisfactoriamente.</p>	
<b>"Curso alternativo"</b>		
	<p>1.1 Se emite un mensaje indicando que el campo no debe estar vacío y regresa a la acción 1 de esta SECCION.</p> <p>1.2 Se emite un mensaje indicando que el usuario o la contraseña no son validos, que no se encuentran en la base de datos y regresa a la acción 1 de esta SECCION.</p>	




Prototipo de Interfaz	
	

Tabla 2.2: Descripción del caso de uso Autenticar usuario.

<b>Nombre del Caso de Uso</b>	<b>Gestionar empleado</b>
<b>Actores</b>	Administrador
<b>Propósito</b>	Permitir que el administrador del sistema gestione los empleados, dígase insertar, modificar, eliminar cualquiera de sus datos, así como mostrar cualquier información del empleado.
<b>Resumen</b>	Es aquí donde se puede insertar, modificar, eliminar y mostrar un empleado.
<b>Referencias</b>	
<b>Precondiciones</b>	El administrador ya autenticado, los roles buscados y cargados.
<b>Pos condiciones</b>	
<b>Sección "Principal"</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>

<p>1. El administrador necesita insertar, modificar, eliminar o mostrar datos de un empleado.</p>	<p>El sistema ejecuta alguna de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>a. Si se selecciona Insertar empleado el sistema muestra la Interfaz Insertar empleado ir a SECCION Insertar empleado.</li> <li>b. Si se selecciona mostrar empleado ir a SECCION Mostrar empleado.</li> <li>c. b. Si se selecciona Modificar empleado ir a SECCION Modificar empleado.</li> <li>d. b. Si se selecciona Eliminar empleado ir a SECCION Eliminar empleado.</li> </ul>
<p><b>“SECCION Insertar empleado”</b></p>	
<p>1.1 El administrador llena los campos usuario, contraseña, carne de identidad, primer nombre, segundo nombre, primer apellido, segundo apellido, sexo, correspondiente al empleado, así como datos de la dirección tradicional: (país, ciudad, calle, entre calles (calle1 calle2), número, distrito), así como los datos de dirección no convencional (tipo y valor) y selecciona el rol correspondiente para ese empleado.</p>	<p>1.2 El sistema verifica que los campos usuario, contraseña, carne de identidad, primer nombre, primer apellido, segundo apellido, sexo no estén vacíos.</p> <p>1.3 El sistema busca que el usuario del empleado, la contraseña y carne de identidad no estén asignados a ningún otro empleado.</p> <p>1.4 El sistema verifica que los datos hayan sido insertados correctamente.</p> <p>1.5 El sistema almacena los datos llenados.</p>
<p><b>Curso alternativo</b></p>	
<p><b>“SECCION Mostrar empleado.”</b></p>	

1 El administrador desea ver los datos de algún empleado según un criterio dado. 2 El administrador selecciona el empleado de la interfaz Buscar Empleado.	1.1 El sistema muestra la invocación del Caso de uso Buscar Empleado. 2.1 El sistema muestra todos los datos del empleado seleccionado en la Interfaz Mostrar Empleado.
<b>Curso alternativo</b>	
<b>Sección “Eliminar empleado”</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1.1 El administrador selecciona el empleado que desea eliminar de la interfaz Buscar empleado. 2.1 El administrador presiona el botón aceptar.	1 El sistema invoca el Caso de uso Buscar Empleado. 2 El sistema muestra un mensaje de verificación, para asegurarse de que en realidad el administrador desea eliminar. 3 El sistema elimina el empleado con sus datos correspondientes.
<b>“Curso alternativo”</b>	
	1 La interfaz Buscar Empleado no necesita ser invocada debido a que la ventana ya está abierta y regresa a la acción 1.1 del actor.
<b>Sección “Modificar empleado”</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>

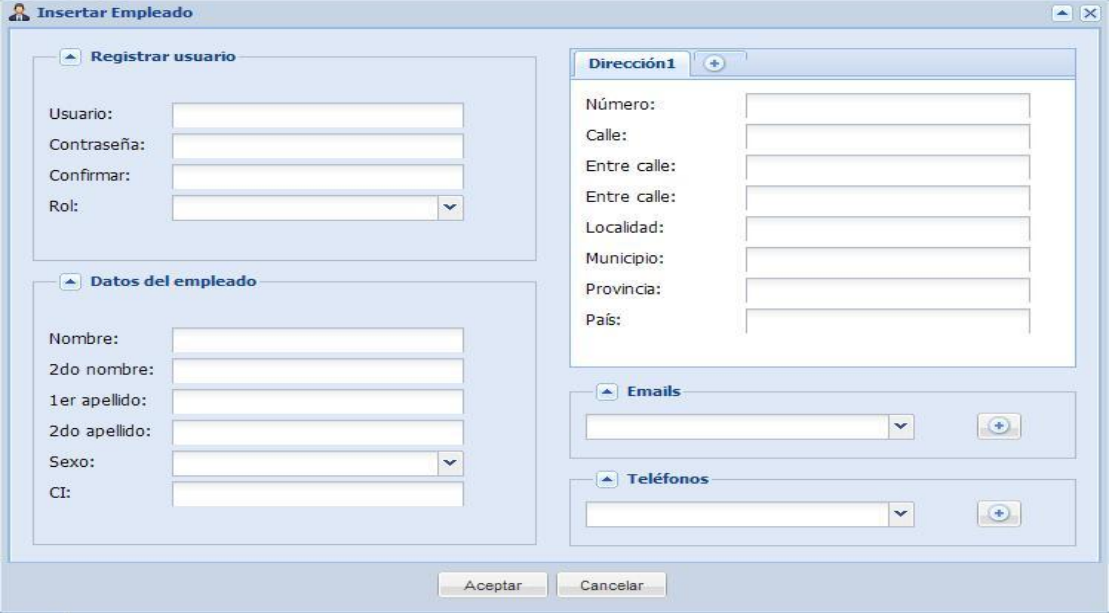
<p>1 El administrador desea modificar datos del empleado.</p> <p>2 El administrador selecciona el empleado de la interfaz Buscar Empleado.</p> <p>3 El administrador modifica los datos del empleado y presiona el botón Modificar.</p>	<p>1.1 El sistema invoca la interfaz Buscar empleado ver caso de uso Buscar Empleado.</p> <p>2.1 El sistema muestra todos los datos del empleado seleccionado en la Interfaz Modificar Empleado.</p> <p>3.1 El sistema almacena los datos modificados.</p>
<p><b>Curso alternativo</b></p>	
	<p>1 La interfaz Buscar Empleado no necesita ser invocada debido a que la ventana ya está abierta y regresa a la acción 1 del actor.</p>
<p><b>Prototipo de Interfaz</b></p>	
	

Tabla 2.3: Descripción del caso de uso Gestionar empleado.

<b>Nombre del Caso de Uso</b>	<b>Buscar empleado.</b>	
<b>Actores</b>	Administrador	
<b>Propósito</b>		
<b>Resumen</b>		
<b>Referencias</b>		
<b>Precondiciones</b>	El administrador ya autenticado.	
<b>Pos condiciones</b>		
<b>Sección “Principal”</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1 El administrador selecciona por algún criterio (carne de identidad, primer nombre, segundo nombre, primer apellido, segundo apellido, y sexo, país, distrito) por el cual desea buscar los empleados.	1.1 El sistema verifica que al menos haya seleccionado un criterio. 1.2 El sistema muestra los campos a llenar por el administrador según el criterio seleccionado.	
2 El administrador llena el o los campos según el criterio escogido y presiona un botón Mostrar.	2.1 El sistema busca empleados y muestra los datos de los criterios que seleccionó. 3.1 El sistema cierra la ventana que representa la Interfaz Buscar.	
3 El administrador decide si quiere cerrar la ventana en el botón cerrar.		
<b>“Curso alternativo”</b>		
	3.1 El sistema mantiene abierta la Interfaz Buscar.	

Tabla 2.4: Descripción del caso de uso Buscar empleado.

<b>Nombre del Caso de Uso</b>	<b>Gestionar operadora</b>	
<b>Actores</b>	Administrador	
<b>Propósito</b>	Permitir que el administrador del sistema gestione las operadoras, dígase insertar, eliminar, modificar cualquiera de sus datos, así como mostrar cualquier información de las operadoras.	
<b>Resumen</b>	Es aquí donde se puede insertar, modificar, eliminar y mostrar una operadora.	
<b>Referencias</b>		
<b>Precondiciones</b>	El administrador ya autenticado, los roles buscados y cargados.	
<b>Pos condiciones</b>		
<b>Sección "Principal"</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El administrador necesita insertar, modificar, eliminar o mostrar datos de una operadora.	<p>El sistema ejecuta alguna de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>a. Si se selecciona Insertar operadora el sistema muestra la Interfaz Insertar operadora ir a SECCION Insertar operadora.</li> <li>b. Si se selecciona mostrar operadora ir a SECCION Mostrar operadora.</li> <li>c. Si se selecciona Modificar operadora ir a SECCION Modificar operadora.</li> <li>d. Si se selecciona Eliminar operadora ir a SECCION Eliminar operadora.</li> </ul>	
<b>"SECCION Insertar Operadora"</b>		

<p>1 El administrador inserta el nombre de la operadora</p> <p>2 El administrador presiona en el botón Agregar Canal.</p> <p>3 El administrador presiona el botón aceptar y los datos se guardan en la base de datos.</p>	<p>1 .1 El sistema verifica que no exista una operadora con el mismo nombre.</p> <p>2.1 El sistema muestra la Interfaz Insertar Canal ver caso de uso Gestionar Canal sección Insertar Canal.</p>
<b>Curso alternativo</b>	
<b>“SECCION Mostrar operadora.”</b>	
<p>1 El administrador desea ver los datos de alguna operadora según un criterio dado.</p> <p>2 El administrador selecciona la operadora de la interfaz Buscar Operadora.</p> <p>3 El administrador presiona el botón Ver Canal.</p>	<p>1.2 El sistema invoca el Caso de uso Buscar Operadora.</p> <p>2.1 El sistema muestra todos los datos de la Operadora seleccionada en la Interfaz Mostrar Operadora con 2 botones Cerrar y Ver Canal.</p> <p>3.2 El sistema invoca el caso de uso Gestionar Canal sección Mostrar Canal</p>
<b>Curso alternativo</b>	
<p>3 El administrador presiona el botón Cerrar.</p>	<p>3.1 El sistema cierra la interfaz y regresa a la interfaz Buscar Operadora ver caso de uso Buscar Operadora.</p>
<b>“SECCION Eliminar operadora.”</b>	
<p>1.1 El administrador selecciona la operadora.</p>	<p>1 El sistema invoca el caso de uso Buscar Operadora.</p> <p>2 El sistema elimina la operadora con todos sus datos correspondientes.</p>
<b>Curso alternativo</b>	

	1 La interfaz ya estaba abierta y se regresa a la acción 1.1 del actor.
<b>“SECCION Modificar operadora.”</b>	
<p>1 El administrador desea modificar datos de una operadora.</p> <p>2 El administrador selecciona la operadora de la interfaz Buscar operadora.</p> <p>3 El administrador modifica los datos de la Operadora y presiona el botón Aceptar.</p> <p>4 El administrador presiona el botón Modificar Canal.</p>	<p>1.1 El sistema invoca la interfaz Buscar Operadora ver caso de uso Buscar Operadora.</p> <p>2.1 El sistema muestra todos los datos de la operadora seleccionado en la Interfaz Modificar Operadora.</p> <p>3.1 El sistema almacena los datos modificados.</p> <p>4.1 El sistema muestra la interfaz Modificar Canal Ver caso de uso Gestionar Canal sección Modificar Canal.</p>
<b>Curso alternativo</b>	
4 El administrador presiona el botón Eliminar Canal Ver caso de uso Gestionar Canal sección Eliminar Canal.	1.1 La interfaz ya estaba abierta y regresa a la acción 2 del actor.



**Prototipo de Interfaz**

The screenshot shows a window titled "Insertar operadora" with a light blue header. The window is divided into two main sections: "Datos de la operadora" on the left and "Dirección" on the right. The "Datos de la operadora" section contains four text input fields labeled "Nombre:", "Slogan:", "Tipo:", and "Industria:". Below these are two sections, "Emails" and "Teléfonos", each featuring a dropdown menu and a "+" button. The "Dirección" section contains seven text input fields labeled "Número:", "Calle:", "Entre calle:", "Entre calle:", "Localidad:", "Municipio:", "Provincia:", and "País:". At the bottom of the window, there are four buttons: "Aceptar", "Agregar canal", "Ver canales", and "Cancelar".

Tabla 2.5: Descripción del caso de uso Gestionar operadora.

<b>Nombre del Caso de Uso</b>	<b>Buscar operadora.</b>	
<b>Actores</b>	Administrador	
<b>Propósito</b>		
<b>Resumen</b>		
<b>Referencias</b>		
<b>Precondiciones</b>	El administrador ya autenticado.	
<b>Pos condiciones</b>		
<b>Sección "Principal"</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1 El administrador selecciona el criterio por el cual desea buscar la operadora.	1.1 El sistema verifica que al menos haya seleccionado un criterio.	
2 El administrador llena el o los campos según el criterio escogido y presiona un botón Mostrar.	1.2 El sistema muestra los campos a llenar por el administrador según el criterio seleccionado.	
3 El administrador presiona el botón cerrar.	2.1 El sistema busca la operadora y muestra los datos de los criterios que seleccionó. 3.1 El sistema cierra la ventana que representa la Interfaz Buscar.	
<b>"Curso alternativo"</b>		
	3.1 El sistema mantiene abierta la Interfaz Buscar.	

Tabla 2.6: Descripción del caso de uso Buscar operadora.

<b>Nombre del Caso de Uso</b>	<b>Gestionar canal</b>	
<b>Actores</b>	Administrador	
<b>Propósito</b>	Permitir que el administrador del sistema gestione los canales, dígase insertar, eliminar, modificar cualquiera de sus datos, así como mostrar cualquier información del canal.	
<b>Resumen</b>	Es aquí donde se puede insertar, modificar, mostrar y eliminar un canal.	
<b>Referencias</b>		
<b>Precondiciones</b>	El administrador ya autenticado, los roles buscados y cargados.	
<b>Pos condiciones</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
<b>“SECCION Insertar canal”</b>		
1 El administrador inserta los datos del canal. 2 El administrador presiona el botón aceptar.	1 .1 El sistema verifica que el nombre del canal no exista en la base de datos. 2.1 El sistema cierra la ventana y regresa a la interfaz Insertar operadora ver caso de uso Gestionar Operadora sección Insertar Operadora.	
<b>Curso alternativo</b>		
<b>“SECCION Mostrar canal.”</b>		

<p>1. El administrador desea ver los datos de algún canal según un criterio dado.</p> <p>2. El administrador selecciona el canal de la interfaz Buscar canal.</p> <p>3. El administrador presiona el botón Cerrar.</p>	<p>1.1 El sistema invoca Caso de uso Buscar canal.</p> <p>2.1 El sistema muestra todos los datos del canal seleccionado en la Interfaz Mostrar canal.</p> <p>3.1 El sistema cierra la ventana y regresa a la Interfaz Buscar Canal ver caso de uso Mostrar Canal.</p>
<b>Curso alternativo</b>	
	<p>1.1 La ventana Buscar canal ya estaba abierta y regresa a la acción 2 del actor.</p>
<b>“SECCION Modificar canal.”</b>	
<p>1 El administrador desea modificar datos de un canal.</p> <p>2 El administrador selecciona el canal de la interfaz Buscar canal.</p> <p>3 El administrador modifica los datos del canal y presiona el botón Modificar.</p>	<p>1.1 El sistema invoca la interfaz Buscar canal ver caso de uso Buscar canal.</p> <p>2.1 El sistema muestra todos los datos del canal seleccionado en la Interfaz Modificar canal.</p> <p>3.1 El sistema almacena los datos modificados.</p>
<b>“Curso alternativo”</b>	
	<p>1.1 La ventana Buscar canal ya estaba abierta y regresa a la acción 2 del actor.</p>
<b>“SECCION Eliminar canal.”</b>	
<p>1.1 El administrador selecciona el canal.</p>	<p>1 El sistema invoca el caso de uso Buscar canal.</p> <p>2 El sistema elimina el canal con todos sus datos correspondientes.</p>

Tabla 2.7: Descripción del caso de uso Gestionar canal.

<b>Nombre del Caso de Uso</b>	<b>Buscar canal.</b>
<b>Actores</b>	Administrador
<b>Propósito</b>	
<b>Resumen</b>	
<b>Referencias</b>	
<b>Precondiciones</b>	El administrador ya autenticado.
<b>Pos condiciones</b>	
<b>Sección "Principal"</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>

<p>1 El administrador selecciona el criterio por el cual desea buscar el canal.</p> <p>2 El administrador llena el o los campos según el criterio escogido y presiona un botón Mostrar.</p> <p>3 El administrador presiona el botón cerrar.</p>	<p>1.1 El sistema verifica que al menos haya seleccionado un criterio.</p> <p>1.2 El sistema muestra los campos a llenar por el administrador según el criterio seleccionado.</p> <p>2.1 El sistema busca el canal y muestra los datos de los criterios que seleccionó.</p> <p>3.1 El sistema cierra la ventana que representa la Interfaz Buscar.</p>
<b>“Curso alternativo”</b>	
	<p>3.1 El sistema mantiene abierta la Interfaz Buscar Canal.</p>

Tabla 2.8: Descripción del caso de uso: Buscar canal.

<b>Nombre del Caso de Uso</b>	<b>Gestionar cliente persona.</b>	
<b>Actores</b>	Diseñador de servicio.	
<b>Propósito</b>	Permitir que el diseñador de servicios gestione los clientes persona, dígame insertar, eliminar, modificar cualquiera de sus datos así como mostrar cualquier información del cliente persona.	
<b>Resumen</b>	Es aquí donde se puede insertar, modificar, mostrar y eliminar un cliente persona.	
<b>Referencias</b>		
<b>Precondiciones</b>	El diseñador de servicios ya autenticado, los roles buscados y cargados.	
<b>Pos condiciones</b>		
<b>Sección “Principal”</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El Diseñador de servicios necesita insertar, eliminar, modificar o buscar datos de un cliente persona.	<p>El sistema ejecuta alguna de las siguientes acciones:</p> <p>a. Si se selecciona Insertar cliente persona el sistema muestra la Interfaz Insertar cliente persona ir a SECCION Insertar cliente persona.</p> <p>b. Si se selecciona Mostrar cliente persona ir a SECCION Mostrar cliente persona.</p> <p>c. b. Si se selecciona Modificar cliente persona ir a SECCION Modificar cliente persona.</p> <p>d. b. Si se selecciona Eliminar cliente persona ir a SECCION Eliminar cliente persona.</p>	
<b>“SECCION Insertar cliente persona”</b>		

<p>1.1 El Diseñador de servicios llena los campos: identificador, primer nombre, segundo nombre, primer apellido, segundo apellido, sexo, carne de identidad, correspondiente al cliente persona, así como datos de la dirección tradicional: (país, ciudad, calle, entre calles(calle1 calle2), número, distrito), y dirección no convencional(tipo y valor) .</p>	<p>1 El sistema verifica que los campos, carne de identidad, primer nombre, primer apellido, segundo apellido y sexo no estén vacíos. 1.3 El sistema verifica que los datos hayan sido insertados correctamente. 1.4 El sistema almacena los datos.</p>
<b>Curso alternativo</b>	
<b>“SECCION Mostrar cliente persona.”</b>	
<p>1.1 El Diseñador de servicios selecciona el cliente de la interfaz Buscar cliente persona. 2.1 El Diseñador de servicios presiona el botón Cerrar.</p>	<p>1 El sistema invoca Caso de uso Buscar cliente persona. 2 El sistema muestra todos los datos del cliente persona seleccionando en la Interfaz Mostrar cliente persona. 3 El sistema cierra la ventana y regresa a la Interfaz Buscar cliente persona.</p>
<b>Curso alternativo</b>	
<b>“SECCION Eliminar cliente persona.”</b>	
<p>1.1 El Diseñador de servicios selecciona el cliente persona.</p>	<p>1 El sistema invoca el caso de uso Buscar cliente persona. 2 El sistema elimina el cliente persona con todos sus datos correspondientes.</p>
<b>“SECCION Modificar cliente persona.”</b>	



<p>1 El Diseñador de servicios desea modificar datos de un cliente persona.</p> <p>2 El Diseñador de servicios selecciona el cliente persona de la interfaz Buscar cliente persona.</p> <p>3 El Diseñador de servicios modifica los datos del cliente persona y presiona el botón Modificar.</p>	<p>1.1 El sistema invoca la interfaz Buscar cliente persona ver caso de uso Buscar cliente persona.</p> <p>2.1 El sistema muestra todos los datos del cliente persona seleccionado en la Interfaz Modificar cliente persona.</p> <p>3.1 El sistema almacena los datos modificados.</p>
--	--

**Prototipo de Interfaz**

The screenshot shows a window titled "Inserte datos de persona" with a light blue background. It contains several input sections:

- Datos de persona:** A group box containing text labels and input fields for "Identificador:", "Nombre:", "2do nombre:", "1er Apellido:", "2do Apellido:", "CI:", and "Sexo:" (with a dropdown arrow).
- Dirección1:** A group box with a "+" icon, containing text labels and input fields for "Número:", "Calle:", "Entre calle:", "Entre calle:", "Localidad:", "Municipio:", "Provincia:", and "País:".
- Emails:** A group box with a dropdown menu and a "+" button.
- Teléfonos:** A group box with a dropdown menu and a "+" button.

At the bottom of the window, there are two buttons: "Aceptar" and "Cancelar".

Tabla 2.9: Descripción del caso de uso Gestionar cliente persona

<b>Nombre del Caso de Uso</b>	<b>Buscar cliente persona.</b>
<b>Actores</b>	Diseñador de servicios
<b>Propósito</b>	
<b>Resumen</b>	
<b>Referencias</b>	
<b>Precondiciones</b>	El Diseñador de servicios ya autenticado.
<b>Pos condiciones</b>	
<b>Sección “Principal”</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1 El Diseñador de servicios selecciona algún criterio por el cual desea buscar el cliente persona. 2 El Diseñador de servicios llena el o los campos según el criterio escogido y presiona un botón Mostrar. 3 El Diseñador de servicios presiona el botón cerrar.	1.1 El sistema verifica que al menos haya seleccionado un criterio. 1.2 El sistema muestra los campos a llenar por el diseñador de servicios según el criterio seleccionado. 2.1 El sistema busca el cliente persona y muestra los datos de los criterios que seleccionó. 3.1 El sistema cierra la ventana que representa la Interfaz Buscar cliente persona.
<b>“Curso alternativo”</b>	
	3.1 El sistema mantiene abierta la Interfaz Buscar cliente persona.

Tabla 2.10: Descripción del caso de uso Buscar cliente persona.

<b>Nombre del Caso de Uso</b>	<b>Gestionar cliente empresa.</b>	
<b>Actores</b>	Diseñador de servicio.	
<b>Propósito</b>	Permitir que el Diseñador de servicios gestione los clientes empresa, dígase insertar, eliminar, modificar cualquiera de sus datos así como mostrar cualquier información del cliente empresa.	
<b>Resumen</b>	Es aquí donde se puede insertar, modificar, mostrar y eliminar un cliente empresa.	
<b>Referencias</b>		
<b>Precondiciones</b>	El Diseñador de servicios ya autenticado, los roles buscados y cargados.	
<b>Pos condiciones</b>		
<b>Sección “Principal”</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El Diseñador de servicios necesita insertar, eliminar, modificar o buscar datos de un cliente empresa.	<p>El sistema ejecuta alguna de las siguientes acciones:</p> <p>a. Si se selecciona Insertar cliente empresa el sistema muestra la Interfaz Insertar cliente empresa ir a SECCION Insertar cliente empresa.</p> <p>b. Si se selecciona mostrar un cliente empresa ir a SECCION Mostrar cliente empresa.</p> <p>c. b. Si se selecciona Modificar cliente empresa ir a SECCION Modificar cliente empresa.</p> <p>d. b. Si se selecciona Eliminar un cliente empresa persona ir a SECCION Eliminar cliente empresa.</p>	
<b>“SECCION Insertar cliente empresa”</b>		

<p>1.1 El Diseñador de servicios llena los campos del identificador, el nombre, industria, slogan, tipo, así como datos de la dirección tradicional: (país, ciudad, calle, entre calles (calle1 calle2), número, distrito), y dirección no convencional (tipo y valor).</p>	<p>1 El sistema verifica que los campos no estén vacíos.                  2 El sistema verifica que los datos hayan sido insertados correctamente.                  3 El sistema almacena los datos.</p>
<b>Curso alternativo</b>	
<b>“SECCION Mostrar cliente empresa</b>	
<p>1.1 El Diseñador de servicios selecciona el cliente empresa de la interfaz Buscar cliente empresa.                  2.1 El Diseñador de servicios presiona el botón Cerrar.</p>	<p>1 El sistema invoca Caso de uso Buscar cliente empresa.                  2 El sistema muestra todos los datos del cliente empresa seleccionado en la Interfaz Mostrar cliente empresa.                  3 El sistema cierra la ventana y regresa a la Interfaz Buscar cliente empresa.</p>
<b>Curso alternativo</b>	
<b>“SECCION Eliminar cliente empresa.”</b>	
<p>1.1 El Diseñador de servicios selecciona el cliente empresa.</p>	<p>1 El sistema invoca el caso de uso Buscar cliente empresa.                  2 El sistema elimina el cliente empresa con todos sus datos correspondientes.</p>
<b>“SECCION Modificar cliente empresa.”</b>	

<p>1 El Diseñador de servicios desea modificar datos de un cliente empresa.</p> <p>2 El Diseñador de servicios selecciona el cliente empresa de la interfaz Buscar cliente empresa.</p> <p>3 El Diseñador de servicios modifica los datos del cliente empresa y presiona el botón Modificar.</p>	<p>1.1 El sistema invoca la interfaz Buscar cliente empresa ver caso de uso Buscar cliente empresa.</p> <p>2.1 El sistema muestra todos los datos del cliente empresa seleccionado en la Interfaz Modificar cliente empresa.</p> <p>3.1 El sistema almacena los datos modificados.</p>
--	--

**Prototipo de Interfaz**

The screenshot shows a window titled "Inserte datos de empresa" with a standard Windows-style title bar. The window is organized into four main sections, each with a small expandable arrow icon at the top left:

- Datos de la empresa:** Contains five input fields labeled "Nombre:", "Identificador:", "Slogan:", "Tipo:", and "Industria:".
- Dirección:** Contains seven input fields labeled "Número:", "Calle:", "Entre calle:", "Entre calle:", "Localidad:", "Municipio:", "Provincia:", and "País:".
- Emails:** Contains a dropdown menu and a plus button (+).
- Teléfonos:** Contains a dropdown menu and a plus button (+).

At the bottom of the window, there are two buttons: "Aceptar" and "Cancelar".

Tabla 2.11: Descripción del caso de uso Gestionar cliente empresa.

<b>Nombre del Caso de Uso</b>	<b>Buscar cliente empresa.</b>
<b>Actores</b>	Diseñador de servicios
<b>Propósito</b>	
<b>Resumen</b>	
<b>Referencias</b>	
<b>Precondiciones</b>	El Diseñador de servicios ya autenticado.
<b>Pos condiciones</b>	
<b>Sección “Principal”</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1 El Diseñador de servicios selecciona algún criterio por el cual desea buscar el cliente empresa. 2 El Diseñador de servicios llena el o los campos según el criterio escogido y presiona un botón Mostrar. 3 El Diseñador de servicios presiona el botón cerrar.	1.1 El sistema verifica que al menos haya seleccionado un criterio. 1.2 El sistema muestra los campos a llenar por el Diseñador de servicios según el criterio seleccionado. 2.1 El sistema busca el cliente empresa y muestra los datos de los criterios que seleccionó. 3.1 El sistema cierra la ventana que representa la Interfaz Buscar cliente empresa.
<b>“Curso alternativo”</b>	
	3.1 El sistema mantiene abierta la Interfaz Buscar cliente empresa.

Tabla 2.12: Descripción del caso de uso Buscar cliente empresa.

<b>Nombre del Caso de Uso</b>	<b>Gestionar servicios.</b>	
<b>Actores</b>	Diseñador de servicios	
<b>Propósito</b>	Permitir que se inserte, elimine o modifique un servicio deseado.	
<b>Resumen</b>	Es aquí donde el diseñador de servicios inserta un servicio, con el canal que le provee el sistema, modifica o elimina uno ya existente.	
<b>Referencias</b>		
<b>Precondiciones</b>	Autenticar usuario, canales disponibles cargados en la interfaz.	
<b>Pos condiciones</b>		
<b>Sección “Principal”</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El actor desea insertar, eliminar o modificar un servicio.	<p>El sistema brinda la posibilidad de insertar, eliminar, modificar y mostrar un servicio.</p> <ul style="list-style-type: none"> <li>a. El sistema muestra la interfaz insertar servicio en caso de que el usuario lo desee, ir a SECCION Insertar servicio.</li> <li>b. El sistema muestra la interfaz modificar servicio en caso de que el usuario lo desee, ir a SECCION Modificar servicio.</li> <li>c. El sistema muestra la interfaz eliminar servicio en caso de que el usuario lo desee, ir a SECCION Eliminar servicio.</li> <li>d. El sistema muestra la interfaz mostrar servicio en caso de que el usuario lo desee, ir a SECCION Mostrar servicio.</li> </ul>	
<b>“Sección Insertar servicio”</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	

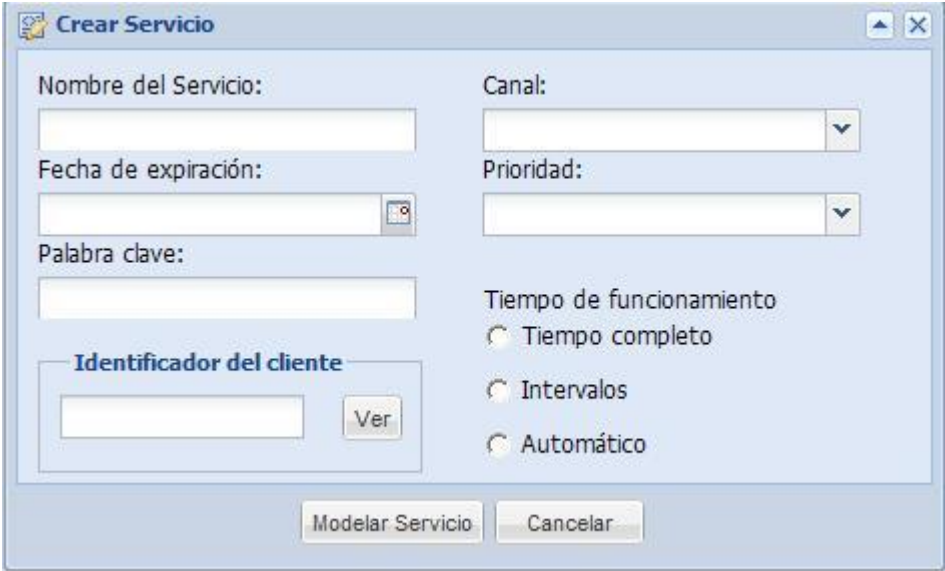
<p>1 El diseñador de servicios inserta el nombre del servicio, fecha de expiración, palabra clave, identificador del cliente al que se le va a asignar el servicio y selecciona el canal, tiempo de funcionamiento y la prioridad.</p>	<p>1.1 El sistema comprueba que los campos no estén vacíos.                  1.2 El sistema comprueba que el identificador exista.                  1.3 El sistema almacena los datos.</p>
<b>Flujo alterno</b>	
<b>Acciones del actor</b>	<b>Respuestas del sistema</b>
<p>1 El diseñador selecciona la opción Recordar identificador Ver caso de uso Recordar Identificador y regresa a la acción 1 del actor.</p>	<p>1.2 El sistema emite un mensaje indicando que el identificador no existe y regresa a la acción 1 del actor.</p>
<b>Prototipo de Interfaz</b>	
	

Tabla 2.13: Descripción del caso de uso Gestionar servicio.



<b>Nombre del Caso de Uso</b>	<b>Buscar servicio.</b>	
<b>Actores</b>	Diseñador	
<b>Propósito</b>		
<b>Resumen</b>		
<b>Referencias</b>		
<b>Precondiciones</b>	El diseñador ya autenticado.	
<b>Pos condiciones</b>		
<b>Sección "Principal"</b>		
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1El diseñador selecciona algún criterio por el cual desea buscar los servicios.	1.1El sistema verifica que al menos haya seleccionado un criterio.	
2El diseñador llena el ó los campos según el criterio escogido y presiona un botón Mostrar.	1.2 El sistema muestra los campos a llenar por el diseñador según el criterio seleccionado.	
3 El diseñador decide si quiere cerrar la ventana en el botón cerrar.	2.1 El sistema busca los servicios y muestra los datos de los criterios que selecciono.	
	3.1 El sistema cierra la ventana que representa la Interfaz Buscar.	
<b>"Curso alternativo"</b>		
	3.1 El sistema mantiene abierta la Interfaz Buscar.	

Tabla 2.14: Descripción del caso de uso Buscar servicio.

<b>Nombre del Caso de Uso</b>	<b>Modelar servicio</b>
<b>Actores</b>	Diseñador de servicios
<b>Propósito</b>	Permitir que el diseñador de servicio, diseñe como va a quedar el servicio solicitado por el cliente.
<b>Resumen</b>	
<b>Referencias</b>	
<b>Precondiciones</b>	El diseñador ya autenticado.
<b>Pos condiciones</b>	
<b>Sección "Principal"</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>

<p>1. El Diseñador de servicios necesita diseñar un servicio para un cliente</p>	<p>El sistema ejecuta alguna de las siguientes acciones:</p> <ul style="list-style-type: none"><li>a. Si selecciona el componente Inicio Servicio ver Sección Insertar componente Inicio Servicio.</li><li>b. Si selecciona el componente Editor Texto ver Sección Insertar componente Editor Texto.</li><li>c. Si selecciona el componente Lista Aceptación ver Sección Insertar componente Lista Aceptación.</li><li>d. Si selecciona el componente Votación ver Sección Insertar componente Votación.</li><li>e. Si selecciona el componente SMS ver Sección Insertar componente SMS.</li><li>f. Si selecciona el componente Fin Servicio ver Sección Insertar componente Fin Servicio.</li></ul> <p>Si selecciona el componente Eliminar ver Sección Insertar componente Eliminar.</p> <p>Si selecciona el componente Nota ver Sección Insertar componente Nota.</p> <p>Si selecciona el componente Arista ver Sección Insertar componente Arista.</p>
<p><b>“SECCION</b> Insertar componente Inicio Servicio”</p>	

<p>1. El Diseñador de servicios selecciona el componente Inicio Servicio de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente Editor Texto”</b>	
<p>1. El Diseñador de servicios selecciona el componente Editor Texto de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente Lista Aceptación”</b>	
<p>1. El Diseñador de servicios selecciona el componente Lista Aceptación de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	

<b>“SECCION Insertar componente Votación”</b>	
<p>1. El Diseñador de servicios selecciona el componente Votación de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente SMS”</b>	
<p>1. El Diseñador de servicios selecciona el componente SMS de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente Fin Servicio”</b>	

<p>1. El Diseñador de servicios selecciona el componente Fin Servicio de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente Nota”</b>	
<p>1. El Diseñador de servicios selecciona el componente Nota de la barra de herramientas.</p> <p>2 El Diseñador de servicios arrastra el componente hacia el área de trabajo donde desea colocar el componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>1.2 El sistema muestra el componente en el lugar solicitado por el diseñador de servicio.</p>
<b>Curso alternativo</b>	
<b>“SECCION Insertar componente Arista”</b>	
<p>1. El Diseñador de servicios selecciona el componente Arista de la barra de herramientas.</p> <p>2 El Diseñador de servicios selecciona el componente en el área de trabajo que desea unir con otro componente seleccionado por él.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>2.1 El sistema muestra la unión de estos componentes a través de una arista.</p>

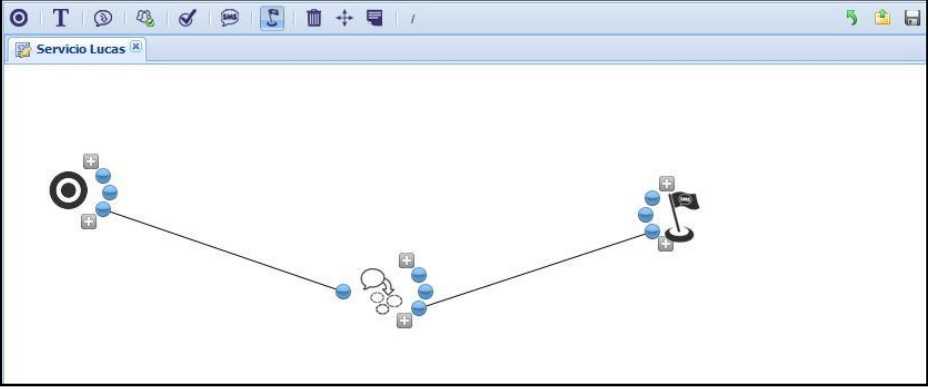
Curso alternativo	
<p>1. El Diseñador de servicios selecciona el componente en el área de trabajo que desea unir con otro componente.</p> <p>2 El Diseñador de servicios selecciona el otro componente.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>2.1 El sistema muestra la unión de estos componentes a través de una arista.</p>
“SECCION Insertar componente Eliminar”	
<p>1. El Diseñador de servicios selecciona el componente Eliminar en la barra de herramientas.</p> <p>2. El Diseñador de servicios selecciona el componente que desea eliminar.</p> <p>3 El diseñador presiona la tecla ESC para desactivar el componente Eliminar.</p>	<p>1.1 El sistema marca el componente para indicar cuál de todos ha sido el seleccionado.</p> <p>2.1 El sistema elimina el componente con todas las relaciones posibles que tenga.</p> <p>3.1 El sistema desmarca el componente para indicar que el componente ha sido desactivado.</p>
Curso alternativo	
<p>3 El diseñador presiona da doble clic encima del componente Eliminar para desactivar el componente.</p>	<p>3.1 El sistema desmarca el componente para indicar que el componente ha sido desactivado.</p>
Prototipo de Interfaz	
 <p>The screenshot shows a software window titled "Servicio Lucas". The window contains a diagram with several nodes and connections. On the left, there is a target icon (a bullseye) surrounded by four blue circular nodes, each with a small square icon containing a plus sign. A line connects this group to a central node, which is a cluster of blue circular nodes with plus signs. Another line connects this central node to a group of four blue circular nodes on the right, each with a small square icon containing a plus sign. The interface also shows a standard Windows-style toolbar at the top with various icons for text, undo, redo, and other functions.</p>	

Tabla 2.15: Descripción del caso de uso Modelar servicio.

### **Conclusiones**

En este capítulo se realizó una descripción de la solución propuesta, se presentó el modelo de dominio, se analizaron y describieron los requisitos funcionales y no funcionales. Se identificaron y describieron los actores y casos de uso, estableciéndose las relaciones entre ellos en el diagrama de casos de uso.





Figura 3.2: Diagrama de clases del diseño del CU Autenticar usuario.

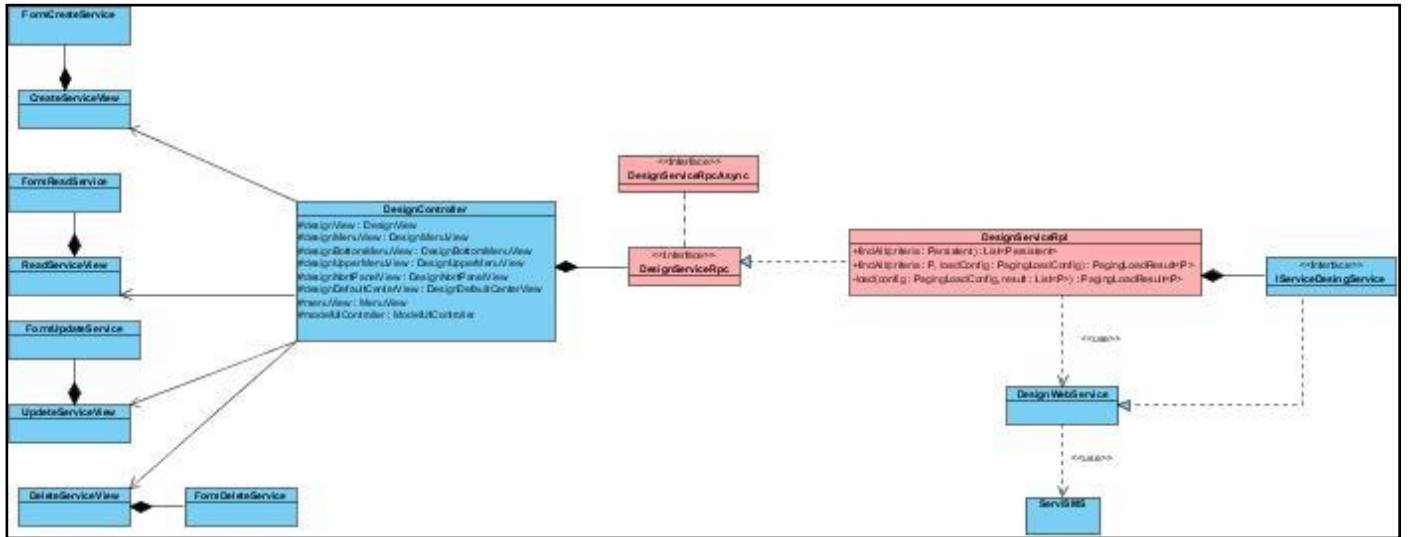


Figura 3.3: Diagrama de clases del diseño del CU Gestionar servicio.

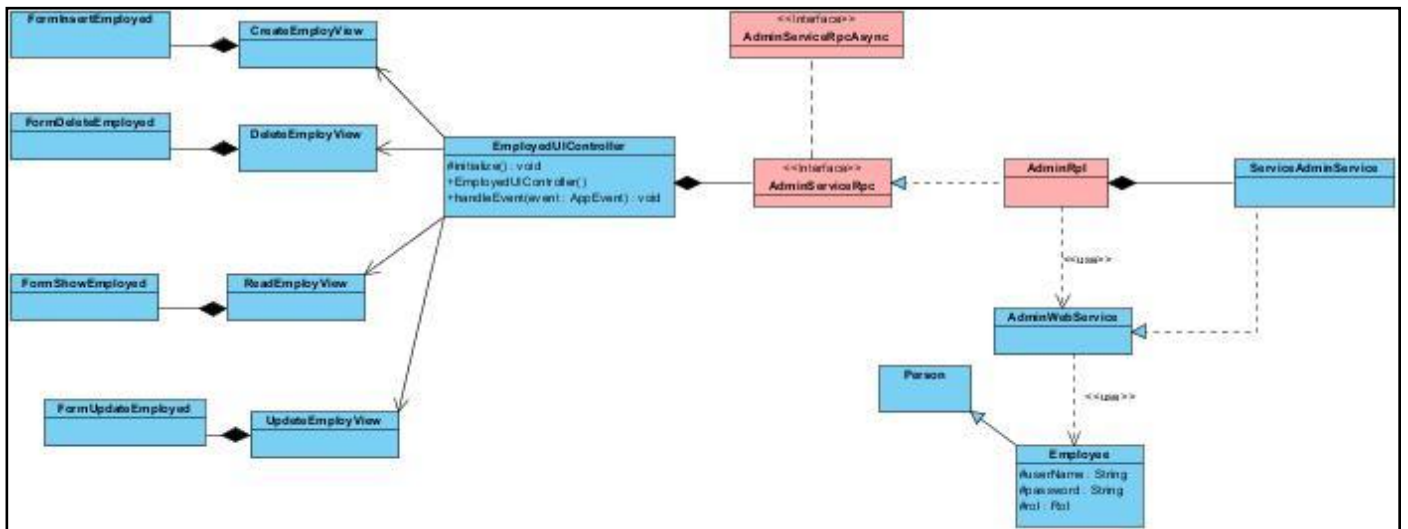


Figura 3.4: Diagrama de clases del diseño del CU Gestionar Empleado.

### 3.2 Definiciones de diseño que se aplican

#### 3.2.1 Arquitectura

##### Patrón Modelo-Vista-Controlador (MVC)

En la mayoría de las aplicaciones web la lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio.

Con el modelo vista controlador se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad.

De esta forma, las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos [19].

Elementos del patrón:

Modelo: Datos y reglas de negocio

Vista: Muestra la información del modelo al usuario

Controlador: Gestiona las entradas del usuario

Un modelo puede tener diversas vistas, cada una con su correspondiente controlador [18].

El modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Llevar un registro de las vistas y controladores del sistema [19].

El controlador es responsable de:

- Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contener reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada a actualizar [19].

Las vistas son responsables de:

- Recibir datos del modelo y mostrarlos al usuario.
- Tener un registro de su controlador asociado [19].

Este patrón se aplica en las clases, Controller que constituyen los controladores, en las clases View que constiuyen las vistas y el modelo lo forman las clases WebServices y las clases del dominio.

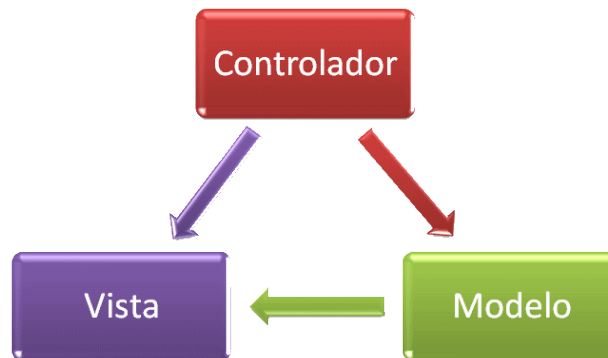


Figura 3.5. Modelo-Vista-Controlador

### 3.2.2 Patrones de Casos de Uso

Los patrones de casos de uso capturan mejores prácticas para modelar casos de uso.

- **Patrón CRUD (Creating, Reading, Updating, Deleting)**

Este patrón se utiliza cuando todos los flujos contribuyen al mismo valor del negocio, y estos a su vez son cortos y simples, modela todas las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación [20]. Se aplica este patrón en las descripciones de los casos de uso, así como en el diagrama de casos de uso, agrupando los flujos que son cortos y simples en casos de uso llamados gestionar.

### 3.2.3 Patrones de diseño

Son directrices y principios estructurados que describen un problema común y entregan una buena solución ya probada a la que le dan un nombre. Estos ayudan a diseñar correctamente en menos tiempo, ayudan a construir problemas reutilizables y facilitan la documentación [21].

- **Bajo acoplamiento**

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. Una clase con bajo (o débil) acoplamiento no depende de muchas otras.

El Bajo Acoplamiento soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, que aumentan la oportunidad de una mayor productividad.

Este patrón se aplica en la inyección de dependencias, posibilitando así que las clases dependan lo menos posible unas de otras.

- **Alta Cohesión**

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Este patrón mejora la claridad y la facilidad con que se entiende el diseño, se simplifican el mantenimiento y las mejoras en funcionalidad, a menudo se genera un bajo acoplamiento. En el sistema se tiene bien definido las responsabilidades de cada clase, por ejemplo, se definen clases por cada caso de uso, para evitar la sobrecarga de funcionalidades en una clase.

- **Controlador**

Un Controlador es un objeto de interfaz no destinada al usuario que se encarga de manejar un evento del sistema. Define además el método de su operación. Un evento del sistema es un evento de alto nivel generado por un actor externo; es un evento de entrada externa. En la aplicación este patrón se ve reflejado en las clases controladoras, las cuales son las encargadas de manejar las peticiones del cliente.

- **Experto**

Expresa la "intuición" de que los objetos hacen cosas relacionadas con la información que poseen. Existen muchos expertos parciales que colaboraron en la tarea puesto que muchas veces la información se encuentra esparcida en varias clases de objetos.

Este patrón permite conservar el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto soporta un bajo acoplamiento, lo que favorece al hecho de tener sistemas más robustos y de fácil mantenimiento. El comportamiento se distribuye entre las clases que

cuentan con la información requerida, alentando con ello definiciones de clases "sencillas" y más cohesivas que son más fáciles de comprender y de mantener. Así se brinda soporte a una alta cohesión. Este patrón se aplica en las clases RPC y RPL, donde cada una realiza funciones específicas, siendo las encargadas de establecer la conexión con el servidor.

- **Creador**

La creación de objetos es una de las actividades más frecuentes en un sistema orientado a objetos, este patrón guía la asignación de responsabilidades con la creación de objetos y define que una instancia de un objeto la tiene que crear el objeto que tiene la información para ello. Este patrón se aplica en las clases View donde se crean los objetos de interfaz de usuario.

### 3.3 Diagramas de interacción

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema, pueden utilizarse para visualizar, especificar, construir y documentar la dinámica de una sociedad particular de objetos, o se pueden utilizar para modelar un flujo de control particular de un caso de uso. En UML los diagramas de interacción pueden representarse a través de los diagramas de colaboración y/o de los diagramas de secuencia. Los diagramas de secuencia muestran interacciones entre objetos basadas en el tiempo. Para ver diagramas de secuencia ver Anexo 2.

### Conclusiones

En este capítulo se realizaron los diagramas de clases del diseño, así como los diagramas de secuencia por cada caso de uso identificado. Se presentan también los patrones utilizados.

### Capítulo 4 Implementación y prueba

#### Introducción

En este capítulo se desarrolla el flujo de trabajo de implementación y prueba. Para realizar el modelo de implementación se confeccionan los diagramas de componentes y de despliegue, dando una visión de cómo quedara desarrollada y distribuida la aplicación.

#### 4.1 Modelo de implementación

El modelo de implementación especifica como los elementos del modelo de diseño se implementan en términos de componentes como ficheros de código fuente, ejecutables, librerías entre otros.

##### 4.1.1 Diagrama de despliegue

Un diagrama de despliegue muestra la distribución física del sistema, las relaciones entre los componentes hardware y software en el sistema final.

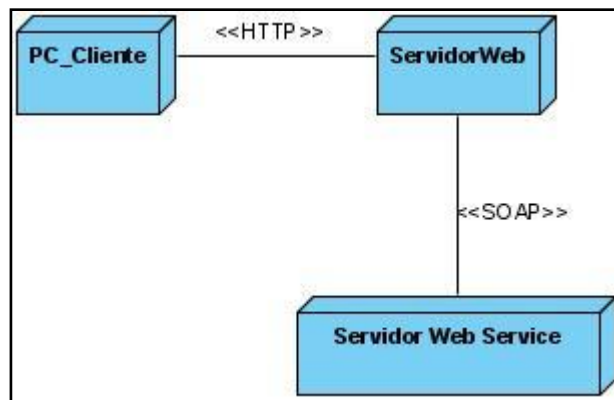


Figura 4.1: Diagrama de despliegue.

##### 4.1.2 Diagrama de componentes

Los diagramas de componentes muestran las organizaciones y dependencias lógicas que existen entre los componentes software, sean estos componentes de código fuente, binarios o ejecutables. Estos

diagramas son utilizados para modelar la vista estática del sistema implantado y cada uno de estos diagramas describe un apartado del sistema.

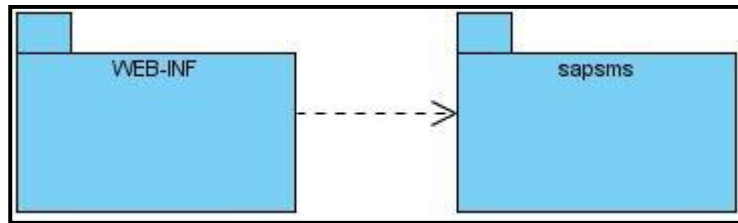


Figura 4.2: Diagrama de componentes general.

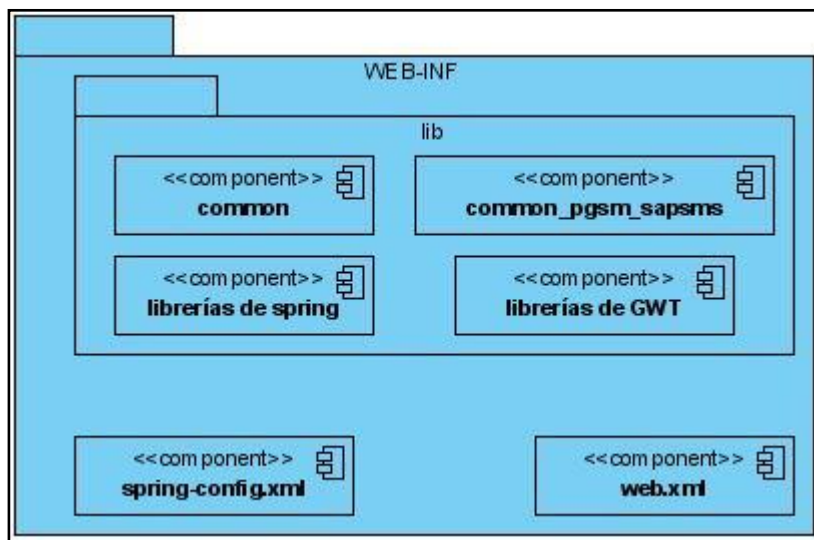


Figura 4.3: Diagrama de componentes del paquete WEB-INF



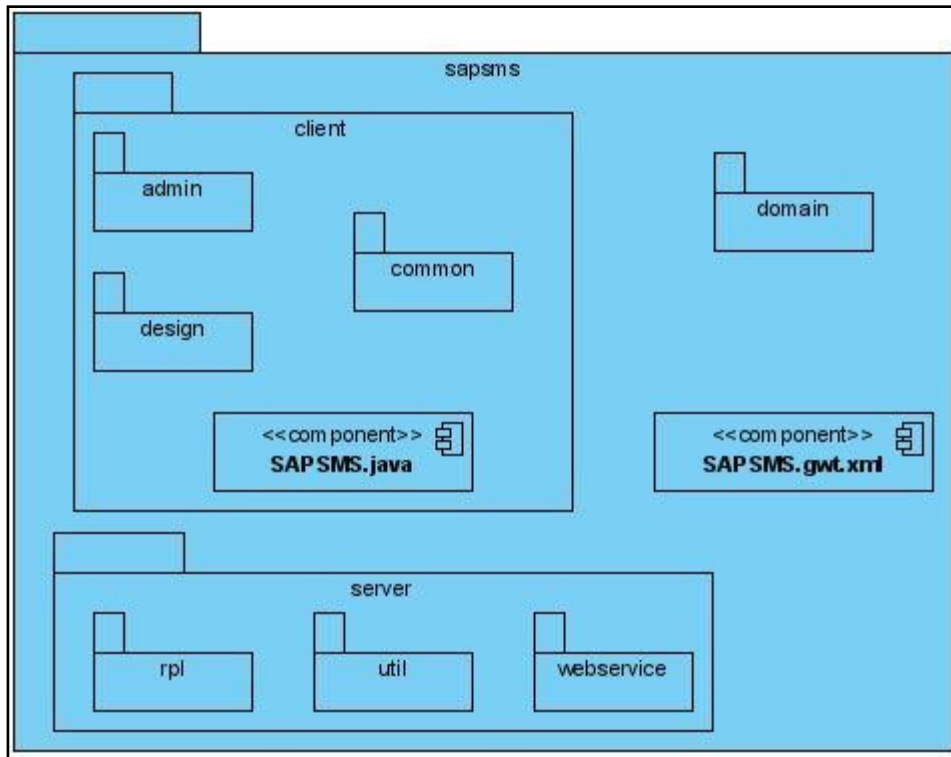


Figura 4.4: Diagrama de componentes del paquete sapsms.

### Conclusiones

En este capítulo se describió la distribución del sistema en nodos especificados en el diagrama de despliegue y se elaboró el diagrama de componentes.

### Capítulo 5 Estudio de la factibilidad:

#### Introducción

El estudio de la factibilidad es una etapa del desarrollo del software que no debe faltar, pues es donde se decide si será factible o no seguir con el desarrollo de este. En este capítulo se realizará mediante el método de Puntos de Casos de Uso una estimación del esfuerzo, costo y tiempo de desarrollo del proyecto.

#### 5.1 Estimación por Puntos de Casos de Uso

Es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de “pesos” a un cierto número de factores que lo afectan, para finalmente contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

##### 5.1.1 Cálculo de Puntos de Casos de Uso sin ajustar

Se calcula a partir de la siguiente ecuación:  $UUCP = UAW + UUCW$

**UUCP:** Puntos de Casos de Uso sin ajustar.

**UAW:** Factor de Peso de los Actores sin ajustar.

**UUCW:** Factor de Peso de los Casos de Uso sin ajustar.

Tipo de actor	Descripción	Factor de peso	actores * peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API).	1	

<b>Medio</b>	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2	
<b>Complejo</b>	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3	2*3
<b>UAW = <math>\sum(\text{actores} * \text{Peso})</math></b>			<b>6</b>

Tabla 5.1: Para calcular el factor de peso de los actores sin ajustar (UAW)

<b>Tipo de CU</b>	<b>Descripción</b>	<b>Peso</b>	<b>CU * Peso</b>
<b>Simple</b>	El CU contiene de 1 a 3 transacciones.	<b>5</b>	7*5
<b>Medio</b>	El CU contiene de 4 a 7 transacciones.	<b>10</b>	7*10
<b>Complejo</b>	El CU contiene más de 8 transacciones.	<b>15</b>	
<b>UUCW = <math>\sum \text{CU} * \text{Peso}</math></b>			<b>105</b>

Tabla 5.2: Para calcular el Factor de Peso de los Caso de Uso sin ajustar (UUCW).

<b>UUCP = 6 + 105 = 111 (Puntos de Casos de Uso sin ajustar)</b>
--

### 5.1.2 Cálculo de Puntos de Casos de Uso ajustados

Se calcula a partir de la siguiente ecuación: **UCP = UUCP x TCF x EF**

**UCP:** Puntos de Casos de Uso ajustados.

**UUCP:** Puntos de Casos de Uso sin ajustar.

**TCF:** Factor de complejidad técnica.

**EF:** Factor de ambiente.

Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

<b>Factor</b>	<b>Descripción</b>	<b>Peso</b>	<b>Valor</b>	<b><math>\Sigma(\text{Peso}_i \times \text{Valor } i)</math></b>
<b>T1</b>	Sistema distribuido.	2	5	10
<b>T2</b>	Objetivos de performance o tiempo de respuesta.	1	3	3
<b>T3</b>	Eficiencia del usuario final.	1	1	1
<b>T4</b>	Procesamiento interno complejo.	1	0	0
<b>T5</b>	El código debe ser reutilizable.	1	3	3
<b>T6</b>	Facilidad de instalación.	0.5	0	0
<b>T7</b>	Facilidad de uso.	0.5	5	2.5
<b>T8</b>	Portabilidad.	2	3	6
<b>T9</b>	Facilidad de cambio.	1	2	2
<b>T10</b>	Concurrencia.	1	0	0
<b>T11</b>	Incluye objetivos especiales de seguridad.	1	2	2
<b>T12</b>	Provee acceso directo a terceras partes.	1	0	0

<b>T13</b>	Se requieren facilidades especiales de entrenamiento a usuarios.	1	0	0
<b>Total</b>				<b>29.5</b>

Tabla 5.3: Para calcular Factor de complejidad técnica (TCF).

Para calcular TCF:

$$\text{TCF} = 0.6 + 0.01 \times \Sigma (\text{Peso} \times \text{Valor asignado})$$

$$\text{TCF} = 0.6 + 0.01 * 29.5$$

<b>TCF = 0.89</b>
-------------------

- Para los factores E1 al E4, un valor asignado de 0 significa sin experiencia, 3 experiencia media y 5 amplia experiencia (experto).
- Para el factor E5, 0 significa sin motivación para el proyecto, 3 motivación media y 5 alta motivación.
- Para el factor E6, 0 significa requerimientos extremadamente inestables, 3 estabilidad media y 5 requerimientos estables sin posibilidad de cambios.
- Para el factor E7, 0 significa que no hay personal part-time (es decir todos son full-time), 3 significa mitad y mitad, y 5 significa que todo el personal es part-time (nadie es full-time).
- Para el factor E8, 0 significa que el lenguaje de programación es fácil de usar, 3 medio y 5 que el lenguaje es extremadamente difícil.

Factor	Descripcion	Peso	Valor	$\Sigma(\text{Peso} \times \text{Valor } i)$
<b>E1</b>	Familiaridad con el modelo de proyecto utilizado	1.5	3	4.5

<b>E2</b>	Experiencia en la aplicación	0.5	3	1.5
<b>E3</b>	Experiencia en orientación a objetos	1	5	5
<b>E4</b>	Capacidad del analista líder	0.5	5	2.5
<b>E5</b>	Motivación	1	5	5
<b>E6</b>	Estabilidad de los requerimientos	2	5	10
<b>E7</b>	Personal part-time	-1	0	0
<b>E8</b>	Dificultad del lenguaje de programación	-1	3	-3
<b>Total</b>				<b>25.5</b>

Tabla 5.4: Para calcular el Factor Ambiente (EF).

Para calcular EF:

$$EF = 1.4 - 0.03 \times \Sigma (\text{Peso}_i \times \text{Valor asignado}_i)$$

$$EF = 1.4 - 0.03 * 25.5$$

$$EF = 0.635$$

Luego UCP:

$$UCP = UUCP \times TCF \times EF$$

$$UCP = 111 * 0.89 * 0.635$$

$$UCP = 62.731 \text{ (Puntos de Casos de Uso ajustados)}$$

### 5.1.3 Estimación del esfuerzo

$$E = UCP \times CF$$

E: Esfuerzo estimado en horas-hombre.

UCP: Puntos de Casos de Uso ajustados.

**CF:** Factor de conversión.

**Para estimar el factor de conversión:**

**CF = 20 horas – hombre** (si total de EF  $\leq$  2)

**CF = 28 horas – hombre** (si total EF = 3 o total EF = 4)

**CF = abandonar o cambiar el proyecto** (si total de EF  $\geq$  5)

**Total EF = Cant EF < 3 (entre E1 - E6) + Cant EF > 3 (entre E7, E8)**

**Total EF = 0**

**CF = 20 horas – hombre porque EF = 0**

**Calculando E**

**E = 62.731 \* 20**

**E = 1254.62 (esfuerzo estimado en horas-hombre)**

**Cálculo del esfuerzo total del proyecto:**

Actividad	% Esfuerzo	Valor esfuerzo
Análisis	10	313.655
Diseño	20	627.31
Implementación	40	1254.62
Prueba	15	470.48
Sobrecarga	15	470.48
<b>Total</b>	<b>100</b>	<b>3136.55</b>

Tabla 5.5: Estimación del esfuerzo del proyecto.

ET = 3136.55 horas-hombre

ET = 16.33 mes- hombre

ET = 16.33 / CH

CH = 2 (Cantidad de hombres)

ET = 8.17

Evaluando la utilización de aproximadamente 8 horas de trabajo diarias y de 24 días laborales al mes, se estima para la elaboración de la aplicación de 8 meses y 17 días, para un equipo de 2 personas.

#### 5.1.4 Estimación del costo

**Costo = CHM \* ET / CH**

Donde:



**CHM:** Costo mes-hombre

**ET:** Esfuerzo total

**CH:** Cantidad de hombres

Si la cantidad de hombres es 2 y se tiene un Salario Promedio de \$100.00

Entonces:

**CHM = CH \* Salario Promedio**

**CHM = 2 \* 100**

**CHM = 200**

Luego:

**Costo = 200 \* 10.7 / 2**

**Costo = \$1070**

### 5.2 Análisis de costos y beneficios

Puesto que la aplicación va dirigida a la plataforma de servicios de mensajería SMS, PLASERV permitiendo que los usuarios puedan explotar de una manera más fácil las funcionalidades que esta brinda, facilitando el proceso de creación de los servicios y las tecnologías que se emplearon en la implementación del sistema son libres, permitiendo que el mismo tenga un costo no muy elevado, se considera que es factible su implementación.

### Conclusiones

En este capítulo se realizó el estudio de la factibilidad mediante el método de Puntos de Casos de Uso, obteniéndose los estimados de horas por hombre y el tiempo que se requiere para desarrollar la solución. También se obtuvo el costo aproximado teniendo en cuenta el salario que reciben los miembros del equipo de trabajo. Se realizó un análisis de distintos indicadores y se llegó a la conclusión de que era factible realizar el Sistema de administración para la plataforma de servicios SMS, PLASERV.

### Conclusiones generales

Después de seleccionar la metodología, herramientas y lenguaje de programación a utilizar en la solución, definir buenas prácticas para el diseño de sistemas de administración y analizar el estado a nivel mundial de los sistemas de administración de plataformas de servicios basados en SMS así como el análisis del funcionamiento de los servicios en estas plataformas, se obtuvo como resultado un sistema de administración para la plataforma de servicios de mensajería basada en SMS, PLASERV, que permite explotar las funcionalidades que brinda esta plataforma, contando el mismo con una herramienta de configuración visual para la creación de los servicios, facilitando este proceso. Además, se gestionan los empleados, clientes, operadoras y canales de la plataforma.

### Recomendaciones

A continuación se exponen algunas recomendaciones a tener en cuenta para futuras versiones del sistema:

- Continuar en el estudio de este tipo de aplicaciones con el fin de añadir mejoras funcionales enfocadas al beneficio del usuario.
- Incluir al sistema seguridad a nivel de transporte para garantizar una mayor seguridad en el envío de datos.
- Se recomienda la implementación de un módulo de reportes para mostrar mediante gráficas el flujo de envío y recepción de mensajes.

### Referencias Bibliográficas

1. La producción en la UCI <http://www.uci.cu/?q=node/46>
2. Teléfonos Móviles [En línea] mayo 20 del 2005 <http://www.telefonos-moviles.com/articles/item.asp?ID=6>
3. **Puneet Gupta**. Short Message Service: What, How and Where? [En línea] 2009. <http://www.wirelessdevnet.com/channels/sms/features/sms.html>
4. Valor Agregado.(febrero 2009)[En Línea] <http://www.crecenegocios.com/valor-agregado/comment-page-1/>
5. Altiria. [En línea] [2008] [http://www.altiria.com/web/sms\\_mms/home.html](http://www.altiria.com/web/sms_mms/home.html).
6. Nudbox [En línea] [2007] <http://www.nudbox.com/smspremium>.
7. Ekosoft [En Línea] <http://www.ekosoft.es/soluciones>.
8. **Alba, Abel** Subgerente Desarrollo Procyonsoluciones. [abel@entumovil.cu](mailto:abel@entumovil.cu)
9. **Booch Grady, Jacobson Ivar, Rumbaugh James**. *El Proceso Unificado de Desarrollo de Software*.
10. **Craig, Larman**. *UML y Patrones*. 1999.
11. PARADIGM, C. V. [En línea] [Citado el: 12 de Febrero del 2009.] <http://www.visual-paradigm.com/product/vpuml/>.
12. Java. *Breve historia de la tecnología Java*. [En línea] [Citado el: 15 de 11 de 2008.] <http://www.java.com/es/about/>.
13. Adictos al Trabajo. *adictosaltrabajo*. [Online] [Cited: abril 23, 2009.] [www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=report](http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=report).
14. **Colectivo de Autores**. Help - Eclipse SDK. *eclipse*. [En línea] 2008. <http://help.eclipse.org/ganymede/index.jsp>.
15. Google Web Toolkit – Documentación en español. [En línea]. <http://esgooglewebtoolkit.blogspot.com/2007/09/qu-es-google-web-toolkit.html>
16. Bravo García, Emilio. El framework Google Web Toolkit. [En línea ] [abril 2010 ] <http://www.i2e.com.es/blog/?cat=3>

17. Communicating with a Server <http://code.google.com/intl/es-ES/webtoolkit/doc/1.6/DevGuideServerCommunication.html>
18. Seam City. [En línea] [Citado el: 21 de enero de 2009.] <http://seamcity.madeinxpain.com/archives/comparativa-spring>.
19. **Burbeck, Steve**. Application programming in Smalltalk-80: How to use Model-View-Controller (MVC). 2007.
20. **Software Guru Conocimiento en práctica** [En línea]. No. 06 <http://www.softwareguru.com.mx>
21. **German Lescano. Patrones** fundamentales en el Diseño Orientado a Objetos. [En línea] 2010. <http://germanlescano.wordpress.com/2010/03/10/patrones-software>