

---

**Universidad de las Ciencias Informáticas**  
**Facultad 6**



**Título: Diseño de la capa de servicios para el  
sistema automatizado de  
Reservación del Transporte estudiantil.**

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

**Autoras:** Dunia Sánchez Hernández.

Niurka García Herrera.

**Tutor:** José Fidalgo Hidalgo.

**Cotutor:** Madelayne Martínez González.

Julio del 2007

## **AGRADECIMIENTOS**

¿Agradecer? Sí, a tantas personas que no me alcanzaría esta página para ello. Primeramente agradecerle a mi mamá y a mi abuelita Norma por estar siempre al tanto de mí, y por ayudarme y aconsejarme cuando más las necesité y las necesito; a mi papá que aunque no esté hoy con nosotros es la principal persona que me ha inspirado; a mi hermana por ser una parte de mí; al resto de mi familia por ser tan especiales conmigo, principalmente a mis tíos Luis, Yary y mis abuelitos Ramón, Brígida y Alberto; a William y Dora por querernos tanto y a su familia por ser tan exclusivos.

A mis amigos de estos cinco años en la universidad: Yami, Elennis, Liusmi, Yanet, Haymeé, Haylín, Indira, Yurima, Lisbeth, Anny, Yordan, Joe, gracias por tenerme en cuenta y confiar en mí; a la "Pity", por apoyarme en todo momento y por compartir con ellos tan buenos momentos; a Pepe, Mirialys, Yadainy, Lianet, Naryana, Yirka, Yohander por ser de esas personas que con su cariño, dedicación y amistad, han cambiado mi forma de pensar y me han devuelto mi autoestima que sinceramente, estaba perdida; a mis amistades de la UCI por saberme tratar con mis virtudes y defectos; a mis decanas Yanet y Grisel por ser parte de lo mejor que he conocido en mi vida, por ayudarme y por ser tan excepcionales; a Lázaro por ser tan atento y preocupado ; a todos los que me ayudaron tanto en mi tesis en especial Made, Pablo, y Pacheco; a todo el colectivo de trabajadores y estudiantes de la mini-UCI de Ciego, Dobao, Arael por ser tan sanos, solidarios y mostrar su cariño con detalles agradables; a mis amigos del barrio, Nady, Tita, Yani a ustedes. A las personas que solo con preguntarme ¿Cómo va eso?, me alentaban a seguir adelante. Mis agradecimientos a todos de corazón, no sé que habría sido sin la colaboración de ustedes.

Para todos: Gracias.

**Dunia**

Estas líneas puede que no son tan expresivas como yo quiero, pero con ellas voy a agradecer a todas las personas que me ayudaron y me apoyaron y que son parte de mí. Primeramente a mi hermana Anerley que es mi guía y mi razón de ser, a mis padres Ana y Tony por ser así de especiales, a mis tíos y a María mi única tía, a mis abuelos y en especial a mami china, a mis primos; en fin a toda mi familia. También a Yudelky por ser mi amiga y mi hermana. Alfredo gracias por tu cariño y tu amor, te adoro. Eleidita, Tony, Albe, Ale, Eleida, Leyanis, Luis, Lianet, Luisi gracias por todo y por ser tan buenos como son. A mis compañeros de grupo por compartir estos cinco largos años conmigo, a José Leandro, Indira y Yadira por soportarme sin peros. También a mis compañeros de la Mini-UCI, la decana y los trabajadores de esta por apoyarnos. Dunia gracias por tu paciencia y por ser así de especial.

A todos muchas gracias por existir y ser tan especiales.

**Niurka**

## **DEDICATORIA**

Esta tesis está dedicada especialmente a mi madre que es la luz que me guía día a día, mi sol, mi amiga, mi hermana, mi confidente, mi ángel de la guarda, el aire que respiro; a mi abuelita Norma que no sé que sería sin ella, pues la fuerza que me impulsa para realizar mi quehacer diario es gracias a ella, por ser mi espíritu, mi conciencia, mi alma, mi vida; a mi papá que aunque no esté físicamente, es la persona que siempre ha estado y estará en mis pensamientos y que me ha guiado por el buen camino; a mi hermana por quererla tanto y tratar de enseñarle y darle lo mejor de mí ser; al resto de mis abuelos por mantener esa confianza de que lo podía lograr; a mis tíos y mis primos por ser personas tan maravillosas y tan especiales conmigo, a William por querernos tanto; a Tony que aunque no es mi hermano de sangre lo quiero más de lo que se imagina; y a todas, toditas mis amistades. Para ustedes, es este sueño hecho realidad.

### **Dunia**

Le dedico este trabajo a mi mayor tesoro, mi hermana, que es la me alegra la vida con su cariño y me insita a ser mejor cada día. También a mis padres por ser tan comprensivos y quererme tanto, a mis tíos por ser como son y a María mi única tía, a mis abuelos a ti mami china por apoyarme siempre. A Eleidita y Tony por su cariño infinito y por ser mis otros papás.

Los quiero mucho.

### **Niurka**

## **RESUMEN**

El avance de la informática y con ello el desarrollo de las tecnologías, ha provocado que las organizaciones incorporen en su funcionamiento aplicaciones Web que posibiliten la gestión de la información, tratando de que las mismas sean cada vez más eficientes y que establezcan una comunicación de la forma más sencilla, eficaz y segura posible. En la Universidad de las Ciencias Informáticas existe una gran dependencia entre las aplicaciones por la falta de estándares de comunicación. Es por ello que con este trabajo se persigue viabilizar la interoperabilidad entre aplicaciones informáticas de la institución y el sistema que permite la gestión de la información correspondiente al pase de los estudiantes los fines de semana, mediante el diseño de servicios que utilicen los estándares de los servicios Web. La solución propuesta facilitará el intercambio de información entre aplicaciones informáticas a través de mensajes, ayudará a eliminar la dependencia entre los sistemas y contribuirá con el avance de la tecnología en el centro.

<b>ÍNDICE</b>	
<b>AGRADECIMIENTOS</b> .....	<b>I</b>
<b>DEDICATORIA</b> .....	<b>III</b>
<b>RESUMEN</b> .....	<b>IV</b>
<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	<b>6</b>
1.1 Gestión de la Información. Sistema de Gestión de la Información .....	6
Sistema de Gestión de la información. ....	7
1.2 Servicios Web. ....	7
Arquitectura Orientada a Servicios: SOA. ....	8
1.3 Transferencia Tecnológica. ....	10
1.4 Ingeniería inversa. ....	11
1.5 Metodología a utilizar. ....	11
Extreme Programming: XP .....	12
Proceso Racional Unificado: RUP. ....	12
1.6 Lenguaje Unificado de Modelado: UML .....	13
1.7 Tecnologías actuales. ....	14
1.7.1 Estándares de los Servicios Web. ....	14
1.7.2 Tecnologías utilizadas en el sistema. ....	19
1.7.3 Técnica utilizada en el sistema. ....	22
1.8 Herramientas empleadas. ....	22
Rational Rose. ....	22
1.9 Objeto de estudio. ....	23
1.10 Análisis del flujo de procesos del sistema para la gestión del pase estudiantil en la UCI: Reservación del Transporte. ....	24
1.11 Conclusiones. ....	24
<b>CAPÍTULO 2: INGENIERÍA INVERSA</b> .....	<b>25</b>
2.1 Modelo del negocio. ....	25
Realizar reservación. ....	25

2.2 Actores del Negocio. ....	25
2.3 Trabajadores del Negocio. ....	26
2.4 Diagrama de casos de uso del negocio. ....	27
2.5 Descripción de cada caso de uso del negocio. ....	28
Caso de uso "Realizar reservación" .....	28
Caso de uso "Tramitar pase" .....	29
Caso de uso "Cancelar reservación" .....	30
2.6 Diagrama de actividades. ....	31
2.7 Diagrama de clases del modelo de objeto. ....	32
2.8 Requerimientos del sistema. ....	33
Realizar reservación. ....	34
2.9 Actores del sistema actual. ....	34
2.10 Definición de los requerimientos funcionales y no funcionales. ....	35
Requerimientos funcionales. ....	35
Requerimientos no funcionales. ....	37
2.11 Diagrama de casos de uso del sistema. ....	38
2.12 Descripción de los casos de uso. ....	39
Caso de uso "Gestionar reservación" .....	39
Caso de uso "Administrar usuario" .....	40
Caso de uso "Administrar ruta" .....	41
Caso de uso "Administrar municipios" .....	42
Caso de uso "Administrar horarios de salida y entrada" .....	43
Caso de uso "Administrar puntos de salida" .....	44
Caso de uso "Reporte de reservaciones" .....	45
Caso de uso "Generar boletines" .....	46
Caso de uso "Realizar búsqueda" .....	46
2.13 Conclusiones. ....	47
<b>CAPÍTULO 3: ESTUDIO DE LOS SERVICIOS</b> .....	<b>48</b>
<b>3.1 Levantamiento de Requerimientos.</b> .....	<b>48</b>
Requerimientos funcionales. ....	48
Requerimientos no funcionales. ....	50
3.2 Casos de uso determinados. ....	51
3.3 Diagrama de casos de uso. ....	52

3.4 Descripción de los casos de uso.....	53
Caso de uso “Servicio gestionar reservación”.....	53
Caso de uso “Servicio administrar usuarios del sistema”.....	54
Caso de uso “Servicio administrar ruta de ómnibus”.....	55
Caso de uso “Servicio administrar municipios”.....	56
Caso de uso “Servicio administrar horarios de salida y entrada”.....	58
Caso de uso “Servicio administrar puntos de salida”.....	58
Caso de uso “Servicio de reporte de reservaciones”.....	59
Caso de uso “Servicio generar boletines”.....	61
Caso de uso “Servicio búsqueda”.....	61
3.4 Conclusiones.....	62
<b>CAPÍTULO 4: PROPUESTA DE SOLUCIÓN .....</b>	<b>63</b>
4.1 Diagramas de clases del análisis.....	63
4.2 Modelo del diseño.....	68
Clases del diseño.....	69
Diagrama de clases del diseño.....	70
4.3 Diagrama del modelo de Datos.....	79
4.4 Conclusiones.....	79
<b>CAPÍTULO 5: PROPUESTA DE TRANSFERENCIA TECNOLÓGICA.....</b>	<b>80</b>
5.1 Transferencia tecnológica.....	80
5.2 Características generales del producto a transferir.....	82
5.3 Documentación necesaria para la transferencia.....	83
Manual de usuario.....	83
Manual de instalación y configuración.....	83
5.4 Estrategia de transferencia tecnológica.....	84
5.5 Actividades a desarrollar para el proceso de transferencia.....	84
Procesos para realizar la transferencia.....	85
5.6 Control de calidad.....	86
5.7 Mantenimiento del sistema.....	87
Tipos de mantenimiento.....	87
5.8 Beneficios de la transferencia de tecnología.....	88
5.9 Conclusiones.....	88



<b>CONCLUSIONES .....</b>	<b>89</b>
<b>RECOMENDACIONES.....</b>	<b>90</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>91</b>
<b>BIBLIOGRAFÍA.....</b>	<b>94</b>
<b>ANEXOS.....</b>	<b>97</b>
Anexo 1: Funcionamiento de un Servicio Web.....	97
Anexo2: Arquitectura Orientada a Servicios.....	97
Anexo 3: Partes de la Arquitectura Orientada a Servicios: SOA.....	98
Anexo 4: Estándares de los Servicios Web.....	98
Anexo 5: Comparación entre modelo clásico y el modelo Ajax.....	99
Anexo: 6 Diagrama de secuencia .....	100
Caso de uso "Servicio administrar horario de salida y entrada" .....	100
Caso de uso "Servicio generar boletín" .....	100
Caso de uso "Servicio administrar municipio" .....	101
Caso de uso "Servicio búsqueda" .....	102
Caso de uso "Servicio administrar puntos de salida" .....	102
Caso de uso "Servicio administrar ruta" .....	103
Caso de uso "Servicio reporte de reservaciones".....	104
Caso de uso "Servicio gestionar reservación" .....	106
Caso de uso "Servicio administrar usuario" .....	107
<b>GLOSARIO .....</b>	<b>109</b>

## ÍNDICE DE FIGURAS

Figura 1.1 Dependencia actual de los sistemas de la UCI.....	23
Figura 2.1 Diagrama de caso de uso.....	27
Figura 2.2. Diagrama de actividades del caso de uso “Realizar reservación”.....	31
Figura 2.3. Diagrama de actividades del caso de uso extendido “Cancelar reservación”.....	32
Figura 2.4. Modelo de objeto.....	33
Figura 2.5 Diagrama de casos de uso del sistema.....	38
Figura 3.1 Diagrama de casos de uso de la capa de servicios.....	52
Figura 4.1 Diagrama de clases del caso de uso” Servicio administrar horario”.....	64
Figura 4.2 Diagrama de clases del caso de uso” Servicio administrar municipio”.....	65
Figura 4.3 Diagrama de clases del caso de uso” Servicio administrar puntos de salida”.....	65
Figura 4.4 Diagrama de clases del caso de uso “Servicio administrar ruta”.....	66
Figura 4.5 Diagrama de clases del caso de uso” Servicio administrar usuario”.....	66
Figura 4.6 Diagrama de clases del caso de uso” Servicio generar boletín”.....	67
Figura 4.7 Diagrama de clases del caso de uso” Servicio reportar reservación”.....	67
Figura 4.8 Diagrama de clases del caso de uso” Servicio gestionar reservación”.....	68
Figura 4.9 Diagrama de clases del caso de uso” Servicio búsqueda”.....	69
Figura 4.10 Diagrama de clases del diseño del caso de uso “Servicio administrar horario”.....	71
Figura 4.11 Diagrama de clases del diseño del caso de uso “Servicio generar boletín”.....	72
Figura 4.12 Diagrama de clases del diseño del caso de uso “Servicio administrar municipio”.....	73
Figura 4.13 Diagrama de clases del diseño del caso de uso “Servicio búsqueda”.....	74
Figura 4.14 Diagrama de clases del diseño del caso de uso “Servicio administrar puntos de salida”.....	75
Figura 4.15 Diagrama de clases del diseño del caso de uso “Servicio administrar ruta”.....	76
Figura 4.16 Diagrama de clases del diseño del caso de uso “Servicio reporte reservaciones”.....	77
Figura 4.17 Diagrama de clases del diseño del caso de uso “Servicio gestionar reservación”.....	78
Figura 4.18 Diagrama de clases del diseño del caso de uso “Servicio administrar usuario”.....	79
Figura 4.19 Diagrama del “Modelo de datos”.....	80

## ÍNDICE DE TABLAS

Tabla 2.1 Justificación de actores del negocio.....	26
Tabla 2.2 Justificación de los trabajadores del negocio.....	27
Tabla 2.3 Especificación textual del caso de uso “Realizar reservación”.....	29
Tabla 2.4 Descripción del caso de uso “Tramitar pase”.....	29
Tabla 2.5 Especificación textual del caso de uso “Cancelar reservación”.....	30
Tabla 2.6 Descripción de los actores del sistema.....	35
Tabla 2.7 Descripción del caso de uso “Gestionar reservación”.....	39
Tabla 2.8 Descripción del caso de uso “Administrar usuario”.....	40
Tabla 2.9 Descripción del caso de uso “Administrar ruta”.....	41
Tabla 2.10 Descripción del caso de uso “Administrar municipio”.....	42
Tabla 2.11 Descripción del caso de uso “Administrar horarios de salida y entrada”.....	43
Tabla 2.12 Descripción del caso de uso “Administrar puntos de salida”.....	44
Tabla 2.13 Descripción del caso de uso “Reporte de reservaciones”.....	45
Tabla 2.14 Descripción del caso de uso “Generar boletines”.....	46
Tabla 2.15 Descripción del caso de uso “Realizar búsqueda”.....	47
Tabla 3.1 Descripción del caso de uso “Servicio gestionar reservación”.....	53
Tabla 3.2 Descripción del caso de uso “Servicio administrar usuario del sistema”.....	54
Tabla 3.3 Descripción del caso de uso “Servicio administrar ruta de ómnibus”.....	55
Tabla 3.4 Descripción del caso de uso “Servicio administrar municipios”.....	57
Tabla 3.5 Descripción del caso de uso “Servicio administrar horario de salida y entrada”.....	58
Tabla 3.6 Descripción del caso de uso “Servicio administrar puntos de salida”.....	59
Tabla 3.7 Descripción del caso de uso “Servicio reporte de reservaciones”.....	60
Tabla 3.8 Descripción del caso de uso “Servicio generar boletines”.....	61
Tabla 3.9 Descripción del caso de uso “Servicio búsqueda”.....	62
Tabla 4.1 Descripción de las clases del diseño.....	70

### INTRODUCCIÓN

Nuestro país ha identificado desde muy temprano la necesidad y conveniencia de dominar e introducir los servicios en la red como parte de la práctica social. Esta determinación va encaminada a lograr una cultura digital imprescindible en el hombre actual, logrando acercar a la sociedad a un desarrollo cada vez más sostenible.

Unido al auge de la informática a nivel mundial, en Cuba, la Universidad de las Ciencias Informáticas (UCI) juega un papel cada vez más importante. Esta universidad desde sus inicios desempeña una ardua labor para informatizar varios sectores de la sociedad, contribuyendo de esta manera a su desarrollo. Aparejado al progreso de la UCI, se ha llevado a cabo la informatización de diferentes servicios que se ofrecen en la misma, en dependencia de sus necesidades. Para lograr esto se han implantando aplicaciones como Intranet, Sistema de Opiniones, Akademos, Inter-nos, Guía Telefónica, Directorio, Teleformación, y otros no menos importantes. Dentro de estas aplicaciones se encuentra el sistema que permite la gestión de la información del pase de los estudiantes semanal, el cual facilita, agiliza y humaniza este proceso. Para mantener este sistema se ponen en práctica los conocimientos y la responsabilidad de estudiantes, profesores y trabajadores del centro.

Antes de la existencia del software anteriormente mencionado, la reservación del pase estudiantil se realizaba manualmente. Para efectuarla, los estudiantes hacían la solicitud a la secretaria de su facultad y posteriormente se entregaban los datos de las reservaciones recogidas al encargado de efectuar la gestión del pase. Todos estos trámites hacían el proceso muy engorroso, pudiendo ocasionar pérdida de la información, dificultad para mantenerla actualizada y por consiguiente, que no se tuviera un control eficiente de los ómnibus necesarios para trasladar a los estudiantes hacia los diferentes destinos disponibles, así como su retorno a la institución. Sumado a esto, se evidenciaba la falta de un mecanismo para controlar el servicio del comedor teniendo en cuenta los estudiantes que salían de pase, aspecto que era fundamental.

Precisamente como respuesta a estas problemáticas es que se implanta el sistema que facilita la gestión del pase, a la cual se le fueron agregando funcionalidades paulatinamente, obteniéndose la actual versión conocida como *Reservación del Transporte*. Esta aplicación informática puesta en funcionamiento posibilita que las reservaciones puedan realizarse de forma simultánea, obteniéndose una verificación de

la misma; facilita tener un conciso conocimiento de los ómnibus necesarios para la transportación de los estudiantes que reservaron para la salir el fin de semana; controla la salida de los estudiantes semanalmente y facilita generar los boletines que acreditan a los estudiantes arribar el ómnibus, entre otras funcionalidades. Independientemente de las funcionalidades expuestas anteriormente, este sistema presenta el siguiente inconveniente: no posee una capa de servicios que propicie la comunicación de otras aplicaciones, con el objetivo de intercambiar información y beneficiarse con funcionalidades del mismo.

Actualmente en la UCI coexisten problemas de dependencia entre aplicaciones existentes, provocando la invasión entre las bases de datos de muchos de estos sistemas, donde una de ellas es la base de datos *Ciudadanos*. Igualmente existen dificultades para la sincronización del flujo de datos, y predomina, la poca seguridad y hasta la duplicidad de la información.

Analizando y teniendo en cuenta lo planteado anteriormente, el **Problema Científico** queda expuesto en el siguiente interrogatorio: ¿Como contribuir para lograr la centralización de las funcionalidades que ofrece el sistema de Reservación del Transporte para la utilización de otras aplicaciones?

Es por ello que el **Objeto de Estudio** de este trabajo son las Aplicaciones Web que gestionan información y están basadas en la tecnología de los servicios Web.

De aquí se deriva el **Campo de Acción** que comprende el diseño de los servicios Web para el sistema de Reservación del Transporte Estudiantil de la UCI.

Para solucionar la problemática expuesta, se han planteado un conjunto de objetivos donde el **Objetivo General** de este trabajo es: Realizar el análisis y diseño de los servicios Web del sistema automatizado para la Reservación del Transporte Estudiantil de la UCI, utilizando la tecnología de los servicios Web.

Del objetivo general propuesto se desglosan los **Objetivos Específicos** planteados a continuación:

- Determinar y elaborar requisitos funcionales y no funcionales del sistema actual.
- Diseñar el diagrama de casos de uso del sistema actual.
- Describir los casos de uso del sistema actual.
- Diseñar el diagrama de casos de uso de los servicios propuestos.
- Describir los casos de uso de los servicios.

- Diseñar los diagramas de clases del análisis de los servicios.
- Diseñar los diagramas de clases del diseño de los servicios.
- Proponer la transferencia tecnológica del sistema de Reservación del Transporte hacia la Facultad Regional de Ciego de Ávila.

Para cumplir con los anteriores objetivos se trazaron las siguientes **Tareas** a tener en cuenta:

- Estudio de las características y funcionamiento de la aplicación de gestión del pase estudiantil de la UCI para realizar la ingeniería inversa.
- Investigación sobre las tecnologías utilizadas en el desarrollo e implementación de los servicios.
- Definición de los requisitos funcionales y no funcionales del sistema.
- Diseño del diagrama de casos de uso del sistema.
- Descripción de los casos de uso del sistema.
- Diseño del diagrama de casos de uso de los servicios propuestos.
- Descripción de los casos de uso de los servicios.
- Diseño de los diagramas de clases del análisis de los servicios.
- Diseño de los diagramas de clases del diseño de los servicios.
- Propuesta de la transferencia tecnológica del sistema de Reservación del Transporte hacia la Facultad Regional de Ciego de Ávila.

Los **Aportes Prácticos** que se obtienen al solucionar esta problemática son los siguientes:

- Facilitar la implementación de los servicios Web correspondientes a la capa de servicio del sistema de gestión del pase estudiantil, para que otras aplicaciones puedan invocar funciones de esta aplicación independientemente de:
  - El lenguaje de programación con que hayan sido creadas.
  - La plataforma o sistema operativo en la que se ejecuten.
  - Dispositivos utilizados para acceder a ellas.
- Facilitar que el sistema brinde mejores funcionalidades.

Este trabajo está estructurado por 5 capítulos:

**Capítulo1:** En este capítulo se hace un estudio valorativo sobre conceptos y definiciones vinculadas con la investigación a realizar, temas que son precisos conocer para un mejor entendimiento del trabajo. Además se describen de forma crítica, metodologías, lenguaje de modelado, tecnologías y herramientas propuestas para utilizar en el diseño de los diferentes servicios propuestos para la capa de servicios del sistema automatizado de Reservación del Transporte estudiantil de la UCI, y de tecnologías y técnicas que son utilizadas actualmente en el funcionamiento del sistema. Además, se hace una reflexión de las principales dificultades que fundamentan la propuesta de solución.

**Capítulo2:** En este capítulo se analizan las problemáticas que provocaron la implementación de la aplicación y se identifican las funcionalidades que este ofrece, facilitando comprender la estructura y la dinámica del sistema. Para desarrollar este proceso se realiza una ingeniería inversa utilizando el modelamiento del negocio y análisis del sistema actual. El objetivo del modelamiento del negocio y análisis de requerimientos está basado en definir la visión del flujo de procesos que se lleva a cabo, los objetivos, el alcance del sistema y la descripción de los procesos.

**Capítulo3:** El levantamiento de requisitos está considerado como uno de los procesos más importantes para el desarrollo de cualquier software. Es por ello que en este capítulo se identifican los requisitos tanto funcionales como no funcionales necesarios para determinar las funcionalidades de cada servicio, describiendo cada caso de uso involucrado en el modelado del sistema.

**Capítulo4:** En este capítulo se describe de manera general todo el proceso relacionado con el análisis y diseño de los servicios correspondientes a la capa de servicio propuesta para el sistema de gestión del pase estudiantil de la UCI. Las actividades contempladas en el análisis permiten dar una visión del sistema, analizando los requisitos funcionales, por lo que su objetivo es proyectar que hace el sistema. Sin embargo, el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, por lo que su objetivo es cumplir las expectativas del sistema.

**Capítulo5:** En este capítulo se describe de forma concreta, el proceso relacionado con la transferencia tecnológica del sistema de Reservación del Transporte estudiantil de la UCI, hacia la Facultad Regional de Ciego de Ávila, realizando un estudio de la repercusión de este proceso en el ámbito internacional, nacional, y el motivo de la universidad para llevar a cabo este proceso. Además se hace un análisis crítico de las características del sistema, de la estrategia y las actividades a seguir para efectuar este

proceso, de los tipos de pruebas a realizar en el sistema, del mantenimiento del sistema y de los beneficios que trae consigo realizar este proceso.



## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo se hace un estudio valorativo sobre conceptos y definiciones vinculadas con la investigación a realizar, temas que son precisos conocer para un mejor entendimiento del trabajo. Además se describen de forma crítica, metodologías, lenguaje de modelado, tecnologías y herramientas propuestas para utilizar en el diseño de los diferentes servicios propuestos para la capa de servicio del sistema automatizado de Reservación del Transporte estudiantil de la UCI, y de tecnologías y técnicas que son utilizadas actualmente en el funcionamiento del sistema. Además, se hace una reflexión de las principales dificultades que fundamentan la propuesta de solución.

### 1.1 Gestión de la Información. Sistema de Gestión de la Información.

La evolución hacia la gestión de contenidos, que comprende la gestión de documentos y datos, tanto internos como externos; la necesidad creciente de gestionar electrónicamente información; el reconocimiento de la informática como herramienta y no como base de la gestión de la información, entre otros aspectos de importancia, caracterizan la relevante importancia que adquiere en estos tiempos la gestión de la información. Se entiende por la gestión de la información a todo lo relacionado con la *obtención de información adecuada en la forma apropiada, para la persona que la necesite en el tiempo oportuno, con el fin de tomar la acción correcta*[1]. Este proceso también permite *analizar y utilizar la información que se ha recabado y registrado para permitir a los administradores, de todos los niveles, tomar decisiones documentadas, donde esta información es la necesaria para tomar las decisiones de gestión.* [2]

Es por ello que la gestión de la información implica:

- Determinar la información que se precisa.
- Recoger y analizar dicha información.
- Registrarla y recuperarla cuando sea necesaria.
- Utilizarla y divulgarla. [2]

Por lo tanto, un buen sistema de gestión de la información ayuda a los administradores y usuarios del sistema, estar informados de los datos que necesitan obtener, con el fin de tomar diferentes decisiones en los momentos que le sean de utilidad.

### **Sistema de Gestión de la información.**

El conjunto de políticas y normas relacionadas entre sí que se establecen para el acceso y tratamiento de los recursos de información [1], es a lo que se considera, Sistema de Gestión de la Información. Este sistema incluye diversos registros administrativos, archivos, soporte tecnológico de los recursos y el público a que se destina.

Para que sean más eficientes y de gran utilidad los sistemas de información, se les han incorporado los servicios Web como un componente más en su funcionamiento, proporcionando estos, mecanismos de comunicación estándares entre diferentes aplicaciones y en la misma aplicación.

### **1.2 Servicios Web.**

Existen múltiples definiciones sobre lo que son los servicios Web, lo que muestra cierta complicación a la hora de dar una adecuada definición. Un servicio Web, según [3], no es más que *un componente de software que se comunica con otras aplicaciones codificando los mensajes en XML y enviando estos mensajes a través de protocolos estándares de Internet tal como HTTP*. Otra definición sería hablar de estos servicios como *un conjunto tecnologías con capacidad para interoperar en la Web* [4].

Estas tecnologías intercambian datos entre sí con el objetivo de ofrecer servicios, donde los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web. **(Ver Anexo 1)**

Los servicios Web presentan un grupo de requisitos que se muestran a continuación:

- Interoperabilidad: un servicio remoto debe permitir su utilización por clientes de otras plataformas.

- Posibilidad de aprovechar los estándares de Internet existentes: la implementación del servicio remoto debería aprovechar estándares de Internet existentes tanto como sea posible y evitar reinventar soluciones a problemas que ya se han resuelto.
- Soporte para cualquier lenguaje: la solución no debería ligarse a un lenguaje de programación particular. Un cliente debería ser capaz de acceder a un nuevo servicio Web existente independientemente del lenguaje de programación en el que se halla escrito.
- Soporte para cualquier infraestructura de componente distribuida: la solución no debe estar fuertemente ligada a una infraestructura de componentes en particular. De hecho, no se debería requerir el comprar, instalar o mantener una infraestructura de objetos distribuidos, solo construir un nuevo servicio remoto o utilizar un servicio existente. Los protocolos subyacentes deberían proporcionar un nivel base de comunicación entre infraestructura de objeto distribuidos existentes. [3]

Ventajas de la utilización de los servicios Web:

- Permiten a los programas escritos en diferentes lenguajes sobre diferentes plataformas comunicarse entre sí de un modo basado en estándares.
- Trabajan con protocolos Web estándares: XML, http y TCP/IP.
- Posibilitan a los usuarios construir nuevas y más potentes aplicaciones que utilicen a los servicios Web como bloques de construcción.
- Posibilitan que sistemas externos puedan invocar las funciones de la aplicación a través de Internet (o una intranet corporativa) sin tener que modificar la aplicación.

Para proporcionar interoperabilidad y extensibilidad entre estas aplicaciones, y que al mismo tiempo sea posible su combinación para realizar operaciones complejas, es necesaria una arquitectura de referencia estándar. Es por ello que la arquitectura recomendada cuando se implanta un servicio Web es la arquitectura orientada a servicios: SOA.

### **Arquitectura Orientada a Servicios: SOA.**

La Arquitectura Orientada a Servicios, SOA por sus siglas en inglés, es un diseño del tipo Cliente/Servidor donde una aplicación se conforma de servicios de software y consumidores de esos

servicios conocidos como clientes; pero SOA difiere del modelo general Cliente/Servidor en el acoplamiento débil entre componentes de software y en la separación de las interfases de la implementación [5]. **(Ver Anexo 2)**

De esta arquitectura se puede decir, que está definida para dar soporte a servicios, por lo que, aunque no es lo mismo que los servicios Web, si es cierto que están estrechamente relacionados y que el auge de dichos servicios Web ha supuesto también el de este tipo de arquitectura. Proporciona independencia total de la plataforma y del lenguaje, además de generar servicios reutilizables y compartidos mediante el uso de lenguajes y protocolos totalmente estandarizados como HTTP, SOAP, XML, UDDI o WSDL.

El funcionamiento de esta arquitectura está distribuida por roles y cualidades que la caracterizan. Estas características se muestran a continuación:

### Características de SOA:

- Es una colección de servicios débilmente acoplados.
- Comunicación con el servicio por protocolos estándares.
- Mecanismo para registrar y localizar los servicios.
- Lenguaje de meta-datos para describir los servicios ofrecidos.

### Roles:

- Proveedor de servicio: ofrece un servicio y publica su definición en un registro junto con una descripción.
- Consumidor del servicio: localiza e invoca el servicio.
- Registro o agencia de descubrimiento: ofrece información sobre la definición y descripción de los servicios disponibles. **(Ver Anexo 3)**

La arquitectura orientada a servicios está compuesta por WSDL que realiza la descripción de los servicios; UDDI que permite el registro y búsqueda de los servicios y además SOAP, http que son los que posibilitan el uso de los servicios. Más adelante se muestran las funciones de cada una de estas tecnologías básicas utilizadas en los servicios Web.

### 1.3 Transferencia Tecnológica.

El término "Transferencia Tecnología" se ha hecho sinónimo de una amplia gama de actividades. Así, la transferencia de tecnología se ha definido como un *proceso para concebir un uso nuevo para una tecnología existente*, o también es definida, como un *proceso para la investigación que se convierte en desarrollo económico* [6]. Pero según [7], la transferencia de tecnología, es el *proceso de transmisión, absorción, adaptación, difusión y reproducción de la tecnología hacia una entidad distinta a la que se originó*. En esta última definición se tiene en cuenta el concepto básico de la existencia de un emisor del conocimiento y de un receptor del mismo.

Cuando se va a realizar una transferencia de tecnología, es válido tener en cuenta, los siguientes aspectos:

- Transmisión de conocimientos tecnológicos para adquirir, instalar y utilizar máquinas, materiales o bienes intermedios.
- Transmisión de conocimientos tecnológicos necesarios para la instalación, operación y funcionamiento de proyectos.
- Materiales destinados a la formación de personal y servicios, tanto de consultoría como de gestión, prestados por personal especializado.[7]

Cuando ya se han determinado e identificado los elementos que serán objeto de transferencia es necesario conocer las modalidades y las fases del proceso. Normalmente son conocidas las modalidades que se muestran a continuación:

- Transferencia horizontal de tecnología: proceso por el cual una tecnología es trasladada de un país a otro.
- Transferencia vertical de tecnología: proceso por el cual dentro de un mismo país, se traslada tecnología de una institución a otra.[7]

La forma de llevar a cabo una transferencia de tecnología es variada, ya que pueden realizarse de diversas formas. Este proceso se puede realizar:

- Por escrito: entiéndase por planos, especificaciones técnicas, etcétera.
- Por medios ópticos y magnéticos: CD- ROM y disquetes.
- Por medios telefónicos: ya sea Módem o cable;
- Por vídeos y sonidos: multimedia.
- Por combinaciones de las anteriores.

### **1.4 Ingeniería inversa.**

La ingeniería inversa es conocida como el proceso que posibilita obtener información técnica a partir de un producto accesible al público, con el fin, de determinar en qué está hecho, qué le permite funcionar y cómo fue implementado. Los productos más comunes que son sometidos a la ingeniería inversa, son los programas de computadoras y los componentes electrónicos.

Este proceso se considera el primer paso a la hora de realizar una reingeniería, por lo que, es reconocido, como el proceso de analizar algún software con el objetivo de recuperar su diseño y especificación. A pesar de que la reingeniería y la ingeniería inversa a menudo tienen mucha relación, no son lo mismo, ya que el objetivo de la ingeniería inversa es obtener el diseño o la especificación de un sistema, mientras que el objetivo de la reingeniería es obtener un nuevo sistema más mantenible. Es aquí la vinculación de ambas, ya que el diseño recuperado en la ingeniería inversa, se utiliza para tener un mejor conocimiento del sistema antes de proporcionarle funcionalidades nuevas.

### **1.5 Metodología a utilizar.**

La realización de sistemas cada vez más eficientes, enfocados siempre a la comodidad del usuario, ha provocado que se desarrollen metodologías que permitan que los sistemas sean desarrollados de una forma más óptima y sencilla.

Los sistemas actuales son implementados para proporcionar mayores comodidades a los usuarios lo cual ha provocado dos cosas, que se realicen sistemas cada vez más complejos y que se desarrollen metodologías buscando la manera más óptima de desarrollarlos. Algunas de estas metodologías son XP y

RUP, metodologías que a continuación se analizarán, para de una forma crítica, elegir la que se utilizará en el diseño de los servicios.

### **Extreme Programming: XP**

Es una de las metodologías de desarrollo de software más exitosa en la actualidad, utilizada para proyectos de corto plazo, equipo pequeño y cuyo plazo de entrega era ayer. Esta metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

Esta metodología se basa en:

- Pruebas unitarias: Pruebas realizadas a los principales procesos, de tal manera que se pueden hacer pruebas de las fallas que pudieran ocurrir.
- Refabricación: Reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: Propone que dos desarrolladores participen en un proyecto en una misma estación de trabajo. [8]

### **Proceso Racional Unificado: RUP.**

Proceso Racional Unificado o RUP (Rational Unified Process), es un proceso de desarrollo de software que junto con el Lenguaje Unificado de Modelado UML, forman la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP es en realidad un refinamiento realizado por Rational Software del más genérico Proceso Unificado.

Sus principales características son:

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en Ingeniería de Software:
  - ✓ Desarrollo iterativo.
  - ✓ Administración de requisitos.
  - ✓ Uso de arquitectura basada en componentes.

- ✓ Control de cambios.
- ✓ Modelado visual del software.
- ✓ Verificación de la calidad del software. [9]

Este proceso se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Es válido mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevado bajo dos disciplinas, las cuales son la disciplina de desarrollo y la de soporte. Además, incluye artefactos y roles, y es acompañado por una herramienta que soporta todos los procesos que se necesitan: Rational Rose Enterprise.

Independientemente que las dos metodologías son de las más usadas en la sociedad, la metodología a utilizar es RUP, ya que es considerada como una de las metodologías más recurridas para alcanzar un grado de certificación en el desarrollo del software, incluyendo que es más adaptable para proyectos de largo plazo, algo que es vital tener en cuenta. Es necesario decir de esta metodología, que está estrechamente relacionada con UML, pues si RUP establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción hasta su nivel más concreto, UML ofrece la notación gráfica necesaria para representar los sucesivos modelos que se obtienen en el proceso de refinamiento.

### **1.6 Lenguaje Unificado de Modelado: UML**

UML o Lenguaje Unificado de Modelado es un lenguaje usado para *especificar, visualizar y documentar los componentes de un sistema en desarrollo orientado a objetos* [9]. Establece la base para un estándar en el dominio del análisis y el diseño orientado a objetos.

A pesar de su status de estándar ampliamente reconocido y utilizado, UML siempre ha sido muy criticado por su carencia de una semántica precisa, lo que ha dado lugar a que la interpretación de un modelo UML no pueda ser objetiva. Otro problema de UML es que no se presta con facilidad al diseño de sistemas distribuidos. En tales sistemas cobran importancia factores como transmisión, persistencia, entre otros. Este lenguaje no cuenta con maneras de describir tales factores. No se puede, por ejemplo, usar UML para señalar que un objeto es persistente, o remoto, o que existe en un servidor que corre continuamente y que es compartido entre varias instancias de ejecución del sistema analizado.



Independientemente de todas las críticas, tiene aspectos que corroboran que sea una herramienta útil para representar los modelos del sistema en desarrollo. UML prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial del significado de estos diagramas y símbolos. Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de la docena de metodologías populares, causando a los revisores tener que aprender las semánticas y notaciones de la metodología empleada antes de intentar entender el diseño como tal. Ahora con UML, diseñadores diferentes, modelando sistemas diferentes, pueden sobradamente entender cada uno los diseños de los otros. Esta notación ha sido desarrollado con el fin de emplearla para modelar diferentes sistemas: de información, técnicos, empotrados en tiempo real; y no sólo es útil para la programación, sino también para modelar negocios, es decir, los procesos y procedimientos que establecen el funcionamiento de una empresa. [9]

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real, ya que ofrece nueve diagramas en los cuales modelar diversos sistemas.

### **1.7 Tecnologías actuales.**

Las diferentes tecnologías que son desarrolladas actualmente por la gran solicitud de los usuarios, y la necesidad y facilidad de aplicarlas en la creación de sistemas o aplicaciones, ya sean complejas o sencillas, son verdaderamente varias. Algunas de ellas son las utilizadas en el desarrollo de servicios Web, lenguajes de programación, servidores, gestores de base de datos y otras tantas que posibilitan que los sistemas sean confiables y óptimos.

#### **1.7.1 Estándares de los Servicios Web.**

En todo el proceso en el que se desenvuelven los Servicios Web, intervienen una serie de tecnologías, partes de la arquitectura orientada a servicios, que hacen posible la circulación y fluidez de la información. Por un lado, está SOAP y XML-RPC los cuales son protocolos basados en XML, estandarizados por el consorcio W3C, que especifican todas las reglas necesarias para ubicar servicios Web XML, integrarlos en aplicaciones y establecer la comunicación entre ellos. Por otro lado, WSDL, que permite que un servicio y un cliente establezcan un acuerdo en lo que se refiere a los detalles de transporte de mensajes

y su contenido. Por último, pero no menos importante, UDDI, el cual permite que los servicios que se registran puedan ser encontrados por los usuarios. A continuación se muestran las mismas. **(Ver Anexo 4)**

### **XML: Extensible Markup Language.**

XML es un conjunto de reglas para definir etiquetas semánticas que permite organizar un documento en diferentes partes. Este metalenguaje, define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados, además de ser un subconjunto de SGML especializado en la gestión de información para la Web.

XML fue creado bajo el amparo del W3C, con el objetivo de definir, validar y compartir formatos de documentos en la Web. En la actualidad este lenguaje presenta varios objetivos donde algunos de ellos son los siguientes:

- Soportar una amplia variedad de aplicaciones.
- Debe ser directamente utilizable sobre Internet.
- Debe ser compatible con SGML.
- Los documentos XML deben ser legibles por humanos y razonablemente claros.

Las principales características que presenta este lenguaje son las que se muestran a continuación:

- Es una tecnología más abierta y extensible.
- Integra los datos de las fuentes más dispares, es decir, se podrá hacer el intercambio de documentos entre las aplicaciones tanto en la propia computadora como en una red local o extensa.
- Los datos son compuestos de múltiples aplicaciones.
- Gestiona y manipula los datos desde el propio cliente Web.[10]

XML consta de cuatro especificaciones donde sienta las bases sintácticas y el alcance de su implementación:

- DTD: Es un archivo(o archivos) que encierra una definición formal de un tipo de documento y, a la vez, especifica la estructura lógica de cada documento. El DTD del XML es opcional.
- XSL: Define o implementa el lenguaje de estilo de los documentos escritos para XML, permitiendo modificar el aspecto de un documento. Puede ser utilizado donde se requiera más potencia del diseño.
- XLL: Define el modo de enlace entre diferentes enlaces, el cual se considera un subconjunto de HyTime, y sigue algunas especificaciones del TEI.
- XUA: Estandarización de navegadores XML. Todavía está en proceso de creación de borradores de trabajo. Se aplicará a los navegadores para que compartan todas las especificaciones XML.[10]

Al igual que el HTML el XML utiliza etiquetas y atributos pero mientras que HTML especifica lo que cada etiqueta y atributo significan y frecuentemente la apariencia que presentará en un navegador el texto que hay entre ellos, XML usa las etiquetas sólo para delimitar piezas de datos, y deja la interpretación de los datos, completamente, a la aplicación que los lee. Eligiendo XML como base para algún proyecto se tiene a disposición una gran y creciente comunidad de herramientas. Optar por XML es parecido a elegir SQL para las bases de datos donde se debe, construir una base de datos propia y procedimientos o programas propios para manipularla.

### **Protocolos de Servicios Web.**

**XML-RPC:** Está diseñado para ser sencillo, por lo que presenta una curva de aprendizaje muy suave, lográndose, en muy poco tiempo y con poco conocimiento resultados satisfactorios. Sin embargo no posibilita elegir el conjunto de caracteres a utilizar en el desarrollo de Servicios Web

**SOAP:** Está creado con idea de dar un soporte completo y minucioso de todo tipo de servicios Web, la curva de aprendizaje es lenta por lo que en poco tiempo y con poco conocimiento no se obtienen resultados satisfactorios. Al contrario de XML-RPC dispone de más potencia y se puede elegir el conjunto de caracteres a utilizar en el desarrollo de Servicios Web como por ejemplo entre US-ASCII, UTF-8 y UTF-16. Es válido agregar que el proyecto Apache, dispone ya de un módulo de trabajo con SOAP, que se puede usar como una librería cliente para invocar servicios SOAP disponibles en cualquier lugar, o como una herramienta del lado del servidor para implementar servicios SOAP.

El protocolo a utilizar por las funcionalidades que presenta, por ser compatible con tecnologías utilizadas en el funcionamiento del sistema y por todo lo expuesto anteriormente es SOAP. A continuación se presenta una breve fundamentación sobre la selección.

### **Protocolo Simple de Acceso a Objetos: SOAP.**

Este protocolo es el que permite la comunicación entre aplicaciones a través de mensajes por medio de Internet. Sus siglas provienen del término Simple Object Access Protocol. Es independiente de la plataforma y del lenguaje; está basado en XML y es la base principal de los Servicios Web. Soporta aplicaciones de tipo documental en las que el mensaje SOAP es simplemente un envoltorio sobre un documento XML. [11]

La característica más convincente de este protocolo es que ha sido implementado en plataformas hardware y software distintos, lo que facilita que pueda utilizarse para enlazar sistemas dispares dentro y fuera de una institución. Es válido agregar, que puede utilizar los parseadores XML y las librerías HTTP para realizar la mayor parte del trabajo duro, en el que la falta de complejidad hace que esté fácilmente disponible.

Debido a la ubicuidad de HTTP y la simplicidad de SOAP los convierten en una base ideal para implementar servicios Web XML que pueden consumirse desde prácticamente cualquier entorno. A continuación se citarán algunas de las principales ventajas de este protocolo.

#### Ventajas de SOAP:

- No está asociado con ningún lenguaje.
- No se encuentra fuertemente asociado a ningún protocolo de transporte.
- No está atado a ninguna infraestructura de objeto distribuido.
- Aprovecha los estándares existentes en la industria.
- Permite la interoperabilidad entre múltiples entornos.[11]

### NuSOAP

NuSOAP es un kit de herramientas para desarrollar Servicios Web bajo el lenguaje PHP, el cual está compuesto por una serie de clases que hacen mucho más fácil su funcionamiento. Esta herramienta provee soporte para el desarrollo de clientes y de servidores. NuSOAP esta basado en SOAP 1.1, WSDL 1.1 y HTTP 1.0/1.1.

A pesar de que existen otras herramientas que se utilizan de soporte para los Servicios Web en PHP, este es uno de los que están en una fase de desarrollo mucho más avanzada.

Ventajas de NuSOAP:

- Está en una fase madura de desarrollo.
- No necesita módulos adicionales.
- Es muy fácil su instalación y uso. [12]

### **Lenguaje de Descripción de Servicios Web: WSDL.**

Es un protocolo basado en XML que describe los accesos al Servicio Web. Sus siglas corresponden a Web Services Description Language. Este protocolo es considerado como el manual de operación del Servicio Web, ya que muestra cuales son las interfaces que provee el Servicio Web y los tipos de datos necesarios para la utilización del mismo.

Se puede decir que un archivo WSDL es un documento XML que describe un conjunto de mensajes SOAP y la realización del intercambio de mensajes; además de definir dónde está disponible el servicio y qué protocolo de comunicaciones utilizar para comunicarse con el servicio.

### **Búsqueda y Registro de Servicios: UDDI.**

Es un modelo de directorios para Servicios Web lo que lo convierte en una especificación para mantener directorios estandarizados de información acerca de los mismos, sus capacidades, ubicación, y requerimientos en un formato reconocido universalmente. Este directorio incluye varias formas de

buscar los servicios que necesitamos para construir las aplicaciones, y utiliza los WSDL para describir las interfaces de estos servicios.

La gran funcionalidad que brindan los servicios Web y la posibilidad de realizar cambios y mejorar nuestras aplicaciones hace que sean un tema apasionante y de constante estudio. Un mismo servicio Web puede ser consumido tanto por aplicaciones que se ejecutan en un servidor y entregan código HTML al cliente, como por aplicaciones que corren directamente en la máquina del cliente.

Luego de conocer el funcionamiento, la importancia y las diferentes características de los servicios Web, se puede resumir, que son uno de los componentes básicos en la distribución y transacción de procesos distribuidos ya sea en Internet o una Intranet. Los estándares abiertos y el foco en la comunicación y colaboración entre las personas y aplicaciones, han creado un entorno donde estos se convierten en la plataforma para la integración de aplicaciones desde diversas fuentes que trabajan conjuntamente, con independencia de dónde residen o su implementación.

### **1.7.2 Tecnologías utilizadas en el sistema.**

#### **PHP: Lenguaje del lado del servidor.**

PHP, cuyas siglas responden a un acrónimo recursivo Hypertext Preprocessor, es un lenguaje sencillo, de sintaxis cómoda y similar a la de otros lenguajes como Perl, C y C++. Este lenguaje es rápido, interpretado, orientado a objetos y multiplataforma. Es ideal tanto para aprender a desarrollar aplicaciones Web como para desarrollar aplicaciones Web complejas, donde se encuentran disponibles múltiples librerías.

El intérprete de PHP, los diversos módulos que presenta y la gran cantidad de librerías desarrolladas para este, son de código libre por lo que el programador de PHP dispone de un impresionante conjunto de herramientas libres para desarrollar aplicaciones, lo que posibilita que sea usado por millones de sitios Web.

La versión de PHP que utiliza el sistema es PHP 4 el cual permite:

- Hacer grandes cosas con pocas líneas de código, lo que hace que merezca la pena aprenderlo.

- Es mucho más legible que PERL.
- Recoger el trabajo del diseñador gráfico e incrustar el código PHP posteriormente de encapsularse dentro del código HTML.
- Funcionar en todas las plataformas que soporten Apache pues es multiplataforma.
- Obtener su código en la Web ya que esta disponible bajo la licencia GPL.[13]

### **Servidores Web.**

Un servidor Web es un *programa que atiende y responde a las diversas peticiones de los navegadores* [8], proporcionándoles los recursos que solicitan mediante el protocolo HTTP o el protocolo HTTPS.

Existen varios servidores de código libre, donde uno de ellos es el servidor Apache.

### **Apache.**

Apache es el servidor Web líder en el mercado. Su coste gratuito, gran fiabilidad y extensibilidad lo convierten en una herramienta potente y muy configurable. Este servidor presenta varias características donde las que más se destacan, y no porque sean las más importantes, son los mensajes de error altamente configurables, las bases de datos de autenticación y el negociado de contenido, aunque es criticado por la falta de una interfaz gráfica que ayude en su configuración. Está diseñado para ser un servidor potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos.

Este servidor es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

- Módulos Base: Módulo con las funciones básicas del Apache.
- Módulos Multiproceso: son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor. [14]

### **Sistema Gestor de Base de Datos.**

Un Sistema Gestor de Base de Datos es un conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad [15]. Es conveniente decir, que este software específico está dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

Existen en la actualidad SGBD de código libre, muchos de ellos del mismo nivel cualitativo que algunos de los SGBD comerciales más conocidos. A continuación se realiza una valoración de Postgre SQL, gestor de base de datos que es utilizado por la aplicación informática.

### **PostgreSQL.**

PostgreSQL es un motor de base de datos, servidor de base de datos relacional libre, liberado bajo la licencia BSD. Está considerado como la base de datos de código abierto más avanzada del mundo. Proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. La siguiente es una breve lista de algunas de esas características, a partir de PostgreSQL 7.1.x.

- DBMS Objeto-Relacional.  
Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multiversión, soporte multiusuario, transacciones, optimización de consultas, herencia, y arreglos.
- Altamente Extensible.  
Soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- Lenguajes Procedurales.  
Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. [16]



### 1.7.3 Técnica utilizada en el sistema.

#### Ajax.

Ajax es el acrónimo para Asynchronous JavaScript + XML, la cual no es en sí una tecnología, sino la unión de varias tecnologías que juntas logran cosas realmente impresionantes. El concepto de esta unión de tecnologías conocida por Ajax, es cargar y redireccionar una página para mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en background, los datos que son usados para actualizar la página solo re-redireccionando la página y mostrando u ocultando porciones de la misma.

Esta tecnología incorpora:

- Presentación basada en estándares usando XHTML y CSS.
- Exhibición e interacción dinámicas usando el Document Object Model.
- Intercambio y manipulación de datos usando XML y XSLT.
- Recuperación de datos asincrónica usando XMLHttpRequest.
- JavaScript vinculando todo el conjunto.[17]

El motor AJAX permite que la interacción del usuario con la aplicación suceda asincrónicamente independientemente de la comunicación con el servidor. **(Ver anexo5)**

### 1.8 Herramientas empleadas.

Para el entendimiento del sistema, el modelamiento de su funcionamiento y la realización de la propuesta del diseño de los servicios Web, se efectuó el estudio de la herramienta existente más apropiada justificando el motivo de su selección.

#### Rational Rose.

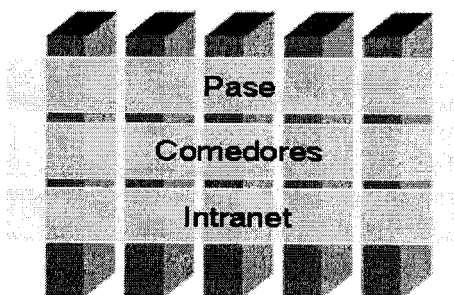
Rose es una herramienta para la *representación gráfica de los casos de uso de negocio y de aplicación, utilizando el Unified Modeling Language (UML)* [18], estándar de la industria. Permite construir los modelos de análisis y diseño especificados en UML, donde provee una edición específica para los lenguajes Java y Visual Basic. Posibilita a los analistas de negocios, la capacidad de modelar y visualizar

sus procesos de negocios y destacar oportunidades para aumentar la eficiencia; y a los analistas de base de datos, crear el modelado de diseño de base de datos, con el fin de mejorar la comunicación entre los clientes y los desarrolladores.

Rational Rose unifica tanto a los analistas de negocios, de sistemas y de datos, al permitirles crear y administrar modelos en una herramienta con un solo lenguaje de modelado. Es por ello que es considerada la herramienta líder en el mundo para el modelado de sistemas complejos y de tiempo real.

### 1.9 Objeto de estudio.

El objeto de estudio de este trabajo de investigación se enfoca en las Aplicaciones Web que posibilitan la gestión de la información y están basadas en los servicios Web para facilitar la comunicación distribuida entre aplicaciones y servicios. La Universidad de las Ciencias Informáticas, UCI, presenta actualmente dificultades que están afectando a muchos sistemas existentes en su funcionamiento. Algunos de estos problemas es la poca independencia existente entre las aplicaciones, lo cual provoca la necesidad de tener aplicaciones con las mismas tecnologías para lograr su comunicación y así puedan intercambiar información que necesiten, la invasión de bases de datos, donde la más usada es la base de datos *Ciudadanos*, la falta de estándares de comunicación y hasta la poca seguridad de la información en algunos casos. Estas problemáticas son causadas por la dinámica que coexiste en la institución y la creación de soluciones que se utilizan solamente para un corto tiempo, por lo que la falta de documentación es otra problemática muy vigente. A continuación se muestra una figura que muestra la dependencia existente entre las aplicaciones de la UCI, las cuales son caracterizadas por ser sistemas horizontales:



**Figura1.1 Dependencia actual de los sistemas de la UCI.**

### **1.10 Análisis del flujo de procesos del sistema para la gestión del pase estudiantil en la UCI: Reservación del Transporte.**

El análisis del flujo de los procesos permite dar a conocer el funcionamiento actual de los procesos que lleva a cabo el sistema, para de esta forma, obtener una solución del problema en cuestión. Actualmente el sistema de Reservación del Transporte estudiantil, imposibilita a las aplicaciones informáticas de la UCI la utilización de funcionalidades que este ofrece. El principal sistema que necesita acceder al sistema del pase estudiantil para obtener información, es la aplicación informática que controla la gestión de acceso del comedor, ya que requiere tener conocimiento y control de los estudiantes que reserven para la salida del pase el fin de semana, con el objetivo de denegarle el servicio del comedor. Esta aplicación obtiene ese informe a través de un documento que se genera, lo que evidencia la problemática ocasionada por la inexistencia de una tecnología que posibilite este tipo de comunicación. Es por ello que se propone darle solución a esta problemática, mediante el diseño de una capa de servicios para el sistema de reservación del pase estudiantil de la UCI utilizando la tecnología y los estándares de los servicios Web, que permitan y faciliten, el intercambio de información entre diferentes aplicaciones informáticas sin tener en cuenta en que está implementado, en la plataforma en que está implantado ni la tecnología que emplea.

### **1.11 Conclusiones.**

En este capítulo se hace una valoración crítica de conceptos, definiciones, tecnologías, metodologías y herramientas que se involucran al tema de investigación y que son útiles para el proceso del diseño de la capa de servicio del sistema de pase de Reservación del Transporte estudiantil en la UCI. Además se esclarecen las razones que evidencian darle solución al objeto de estudio, por lo que se puede llevar a cabo el estudio y documentación del funcionamiento del sistema.

## **CAPÍTULO 2: INGENIERÍA INVERSA**

En este capítulo se analizan las problemáticas que provocaron la implementación de la aplicación y se identifican las funcionalidades que este ofrece, facilitando comprender la estructura y la dinámica del sistema. Para desarrollar este proceso se realiza una ingeniería inversa utilizando el modelamiento del negocio y análisis del sistema actual. El objetivo del modelamiento del negocio y análisis de requerimientos está basado en definir la visión del flujo de procesos que se lleva a cabo, los objetivos, el alcance del sistema y la descripción de los procesos.

### **2.1 Modelo del negocio.**

Algunas de las actividades más importantes a desarrollar en el modelo del negocio son la definición e identificación de los procesos que ocasionaron la implementación del sistema, la identificación de actores y trabajadores del negocio, la delimitación de los casos de uso del negocio y la descripción correspondiente a cada caso de uso. A continuación se realiza un análisis de los procesos que definieron el negocio antes de implementar el sistema. Este proceso es realizar reservación.

#### **Realizar reservación.**

El proceso de realizar reservación tiene como objetivo que cada estudiante reserve su pase y lo cancele si le es necesario. Al estudiante solicitar la reservación de su pase, la secretaria lo agrega al listado anotando el nombre y apellidos, grupo y la ruta que desea. Luego esta le comunica la cantidad de estudiantes que reservaron, para que el administrador conozca la cantidad de estudiantes que solicitaron reservar el pase de la facultad. En caso de que un estudiante después de realizar la reservación la desea cancelar, la secretaria lo elimina de la lista y actualiza la lista de las reservaciones efectuadas.

### **2.2 Actores del Negocio.**

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos, con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.

Actores del negocio	Justificación
Estudiante	El estudiante es el que inicia las acciones que dan cumplimiento al proceso de realizar reservación, siendo así beneficiado con el resultado de dicho proceso.

**Tabla 2.1 Justificación de actores del negocio.**

### 2.3 Trabajadores del Negocio.

Un trabajador del negocio define el comportamiento y responsabilidades de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo, los cuales interactúan con actores del negocio y realizan las actividades.

Trabajadores del negocio	Justificación
Secretaria	Es la encargada de recoger en un listado, todos aquellos estudiantes que reservaron el pase para ese fin de semana, las cancelaciones efectuadas, así como la ruta que utilizarán. No es beneficiado con ninguna de las acciones que realiza, simplemente las lleva a cabo.
Decano	Es el que informa al administrador la cantidad de estudiantes de su facultad que salen de pase ese fin de semana y la ruta que necesitan emplear ese fin de semana. No se beneficia con ninguna de las acciones que realiza.

<p>Administrador</p>	<p>Es el responsable de tramitar la cantidad de ómnibus necesaria para transportar los estudiantes que reservaron pase hacia los distintos destinos disponibles, así como garantizar su retorno a la universidad. No es beneficiado con ninguna de las acciones desarrolladas.</p>
----------------------	--

Tabla 2.2 Justificación de los trabajadores del negocio.

### 2.4 Diagrama de casos de uso del negocio.

Un diagrama de caso de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio. En la siguiente figura se muestra el diagrama de caso de uso del negocio:

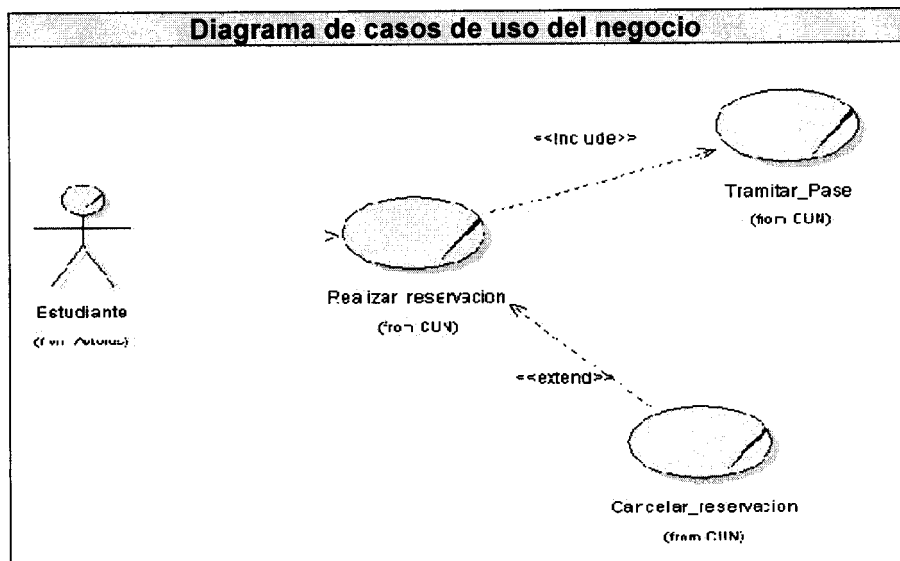


Figura 2.1 Diagrama de caso de uso.

2.5 Descripción de cada caso de uso del negocio.

Caso de uso "Realizar reservación"

<b>Caso de uso.</b>	Realizar reservación.
<b>Caso de uso asociados.</b>	
<b>Actores del negocio.</b>	Estudiante (Inicia).
<b>Trabajadores del negocio:</b>	Secretaria, Decano, Administrador.
<b>Propósito:</b>	Posibilitar al estudiante realizar la reservación de su pase para salir el fin de semana, cancelar en caso de que lo necesite e informarle al administrador de las reservaciones.
<b>Resumen:</b>	
<p>El caso de uso inicia cuando un estudiante llega al decanato donde es atendido por la secretaria y le informa que desea reservar pase de ida y/o vuelta así como la ruta. La secretaria lo agrega al listado de las reservaciones. Luego le comunica a la decana para que informe al administrador de las reservaciones realizadas. En caso de que el estudiante desee cancelar la reservación se lo comunica a la secretaria la cual actualiza el listado de las reservaciones, finalizando así el caso de uso.</p>	
<b>Casos de uso asociados:</b> Cancelar reservación, Tramitar pase.	
<b>Flujo de normal de eventos.</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1. El estudiante se presenta al decanato y solicita reservar el pase.	2. La secretaria le informa las rutas disponibles.
	3. La secretaria le pregunta cual es la ruta a reservar, el nombre, apellidos y el grupo.
4. El estudiante el brinda la información requerida	5. Lo agrega al listado de los estudiantes que solicitaron esa ruta.
	6. Le informa que su reservación ha sido realizada.
7. El estudiante se retira.	8. Luego se invoca al caso de uso <b>Tramitar pase.</b>

<b>Flujos Alternos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio.</b>
1.1 El estudiante se presenta al decanato e informa que desea cancelar su reservación.	2.1 Se invoca al caso de uso <b>Cancelar reservación.</b>
<b>Prioridad</b>	Crítico.

**Tabla 2.3 Especificación textual del caso de uso “Realizar reservación”**

**Caso de uso “Tramitar pase”**

<b>Caso de uso.</b>	Tramitar pase
<b>Caso de uso base</b>	Realizar reservación.
<b>Trabajadores del negocio:</b>	Secretaria, Decano, Administrador.
<b>Propósito:</b>	Informarle al administrador las reservaciones que han sido realizadas por facultad.
<b>Resumen:</b>	
<p>El caso de uso inicia cuando la secretaria le informa a la decana, mediante un listado, la cantidad de estudiantes que reservaron pase de la facultad y la ruta correspondiente. Luego la decana le entrega esa información al administrador, el cual actualiza el listado de reservaciones que posee. Una vez concluido este proceso, se retorna al caso de uso base.</p>	
<b>Flujo de normal de eventos.</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
	1. La secretaria le entrega el listado de la cantidad de estudiantes de la facultad que solicitaron la reservación del pase con las rutas correspondientes.
	2. La decana le hace entrega de ese listado al administrador.



	3. El administrador actualiza el listado general.
<b>Prioridad</b>	Crítico

**Tabla 2.4 Descripción del caso de uso “Tramitar pase”**

**Caso de uso “Cancelar reservación”**

<b>Caso de Uso.</b>	<b>Cancelar reservación.</b>
<b>Caso de uso base</b>	Realizar reservación.
<b>Trabajadores del negocio.</b>	Secretaria, Decano, Administrador.
<b>Resumen:</b>	El caso de uso inicia cuando el estudiante no puede realizar su salida el fin de semana por una determinada situación, por lo que cancela la reservación. Es allí donde le plantea a la secretaria sobre su decisión, la secretaria cancela su reservación, y actualiza el listado de reservaciones, retornando al caso de uso base.
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>
1. El estudiante se presenta al decanato e informa sobre su cancelación de la reservación.	2. La secretaria le pregunta ruta, nombre, apellidos y grupo.
3. El estudiante informa los datos solicitados.	4. La secretaria inicia una búsqueda del estudiante en los documentos correspondientes y cancela su reservación.
	5. Le informa al estudiante que su reservación ha sido cancelado.
6. El estudiante se retira.	7. Se retorna al caso de uso base.
<b>Prioridad</b>	Crítico

**Tabla 2.5 Especificación textual del caso de uso “Cancelar reservación”**



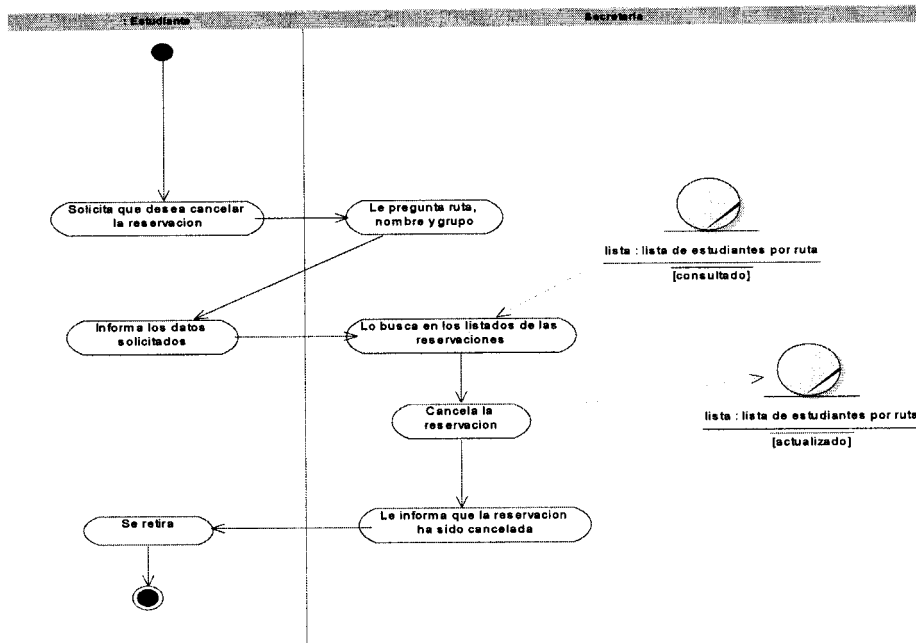
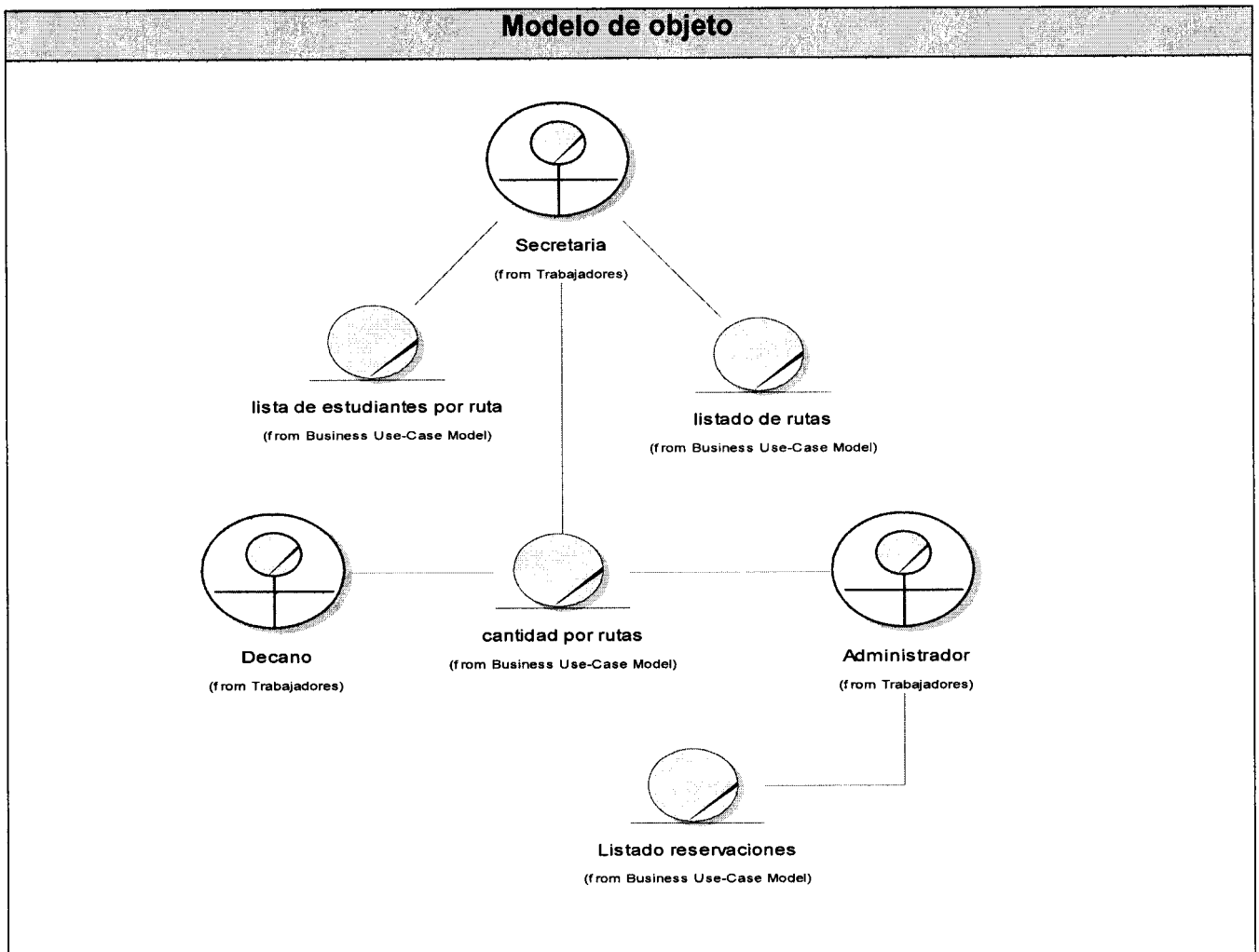


Figura 2.3. Diagrama de actividades del caso de uso extendido “Cancelar reservación”

## 2.7 Diagrama de clases del modelo de objeto.

El diagrama de clases, se construye en el negocio para describir el modelo de objetos del negocio, donde muestra la participación de los trabajadores, de las entidades del negocio y la relación existente entre ellos. A continuación se muestra el diagrama de clase del modelo de objeto del caso de uso del negocio “Realizar reservación.”



**Figura 2.4. Modelo de objeto.**

## 2.8 Requerimientos del sistema.

En el capítulo anterior se realizó un análisis crítico del flujo de proceso del sistema actual considerando la posible propuesta de mejora a este sistema, el cual mantiene su actual funcionamiento pero con otras funcionalidades incluidas. A continuación se realiza una descripción de los procesos que detallan el análisis de la aplicación actual. Este proceso es Realizar Reservación.

**Realizar reservación.**

El proceso de realizar reservación tiene como objetivo que cada estudiante realice la reservación de su pase, sea capaz de modificarlo o cancelarlo si le es necesario. Es válido agregar, que el decano también tiene la posibilidad de realizar esta reservación en caso excepcional.

Cuando el estudiante realiza la reservación del pase, el sistema va registrando todas esas reservaciones para luego, facilitarle al administrador conocer la cifra total de estudiantes que reservaron por ruta, por facultad y por año, con el objetivo de gestionar los ómnibus necesarios para la salida el fin de semana. Si el estudiante se le presenta cualquier necesidad y cree necesario que tiene que cambiar datos de la reservación realizada, tiene la posibilidad de modificarla. Ocurre lo mismo cuando no tiene la posibilidad de salir ese fin de semana de pase, donde el sistema le permite cancelar la reservación realizada. Todas esas actualizaciones el sistema lo tiene en cuenta para actualizar el listado de las reservaciones.

A partir de que se deshabilita el sistema, es decir, que no se pueden realizar más reservaciones, el asistente de impresión obtiene del sistema todas las reservaciones realizadas de su facultad para poder imprimir el boletín que acredita al estudiante arribar al ómnibus.

**2.9 Actores del sistema actual.**

Los actores del sistema pueden representar el rol que juegan una o varias personas, un equipo o un sistema automatizado; no siendo parte de él, y con el cual pueden intercambiar información pudiendo ser un recipiente pasivo de información. Los actores del sistema se muestran a continuación:

Actores	Justificación
Usuario	Este actor generaliza procesos desarrollados en el sistema.
Decano	Es un usuario que es el responsable de reservar o cancelar la reservación a un estudiante en caso de ser necesario, y de revisar y conocer las reservaciones realizadas de su facultad por año y por ruta.

Administrador	Es un usuario que se encarga de administrar los cambios en la aplicación, entiéndase por las rutas, los usuario, los municipios, los horarios, los puntos de salida; y obtiene la cifra de estudiantes por ruta, año y facultad que reservaron ya sea la ida o el regreso, con el objetivo de garantizar la transportación y controlar cualquier cambio.
Impresión	Es un usuario que responde por la impresión de los boletines que acreditan a los estudiantes para arribar al ómnibus en la salida del pase.
Estudiante	Es un usuario que realiza la reservación del pase, lo modifica y lo cancela en caso que sea necesario.

**Tabla 2.6 Descripción de los actores del sistema.**

## **2.10 Definición de los requerimientos funcionales y no funcionales.**

### **Requerimientos funcionales.**

Los requerimientos funcionales son aquellos requerimientos que, representan capacidades que el sistema debe cumplir para satisfacer las necesidades del usuario y que contribuyen a encontrar funcionalidades del sistema implantado. El sistema de Reservación del Transporte, presenta los siguientes requisitos funcionales:

#### **R.1 Gestionar reservación.**

**R.1.1** Insertar información necesaria.

**R.1.2** Modificar información.

**R.1.3** Cancelar reservación.

**R.1.4** Buscar reservación.

**R.2 Administrar usuarios del sistema.**

**R.2.1 Insertar usuario.**

**R.2.2 Mostrar listado de usuarios.**

**R.2.3 Buscar usuario.**

**R.3 Administrar ruta de ómnibus.**

**R.3.1 Agregar una ruta.**

**R.3.2 Mostrar listado de rutas.**

**R.3.3 Buscar ruta.**

**R.4 Administrar municipios.**

**R.4.1 Insertar municipio.**

**R.4.2 Modificar municipio.**

**R.4.3 Buscar municipio.**

**R.5 Administrar horarios de entrada y salida.**

**R.5.1 Modificar horario.**

**R.5.2 Buscar horario.**

**R.6 Administrar puntos de salida.**

**R.6.1 Insertar punto de salida de los ómnibus.**

**R.6.2 Modificar puntos de salida de los ómnibus.**

**R.6.3 Buscar puntos de salida.**

**R.7 Mostrar reporte.**

**R.7.1 Mostrar total de reservaciones por ruta.**

**R.7.2 Mostrar total de reservaciones por año.**

**R.7.3 Mostrar total de reservaciones por facultad.**

**R.7.4 Buscar reservaciones.**

**R.8 Generar boletín.**

**R.8.1 Generar boletín.**

**R.8.2 Buscar estudiante.**

**R.9 Realizar búsqueda.****Requerimientos no funcionales.**

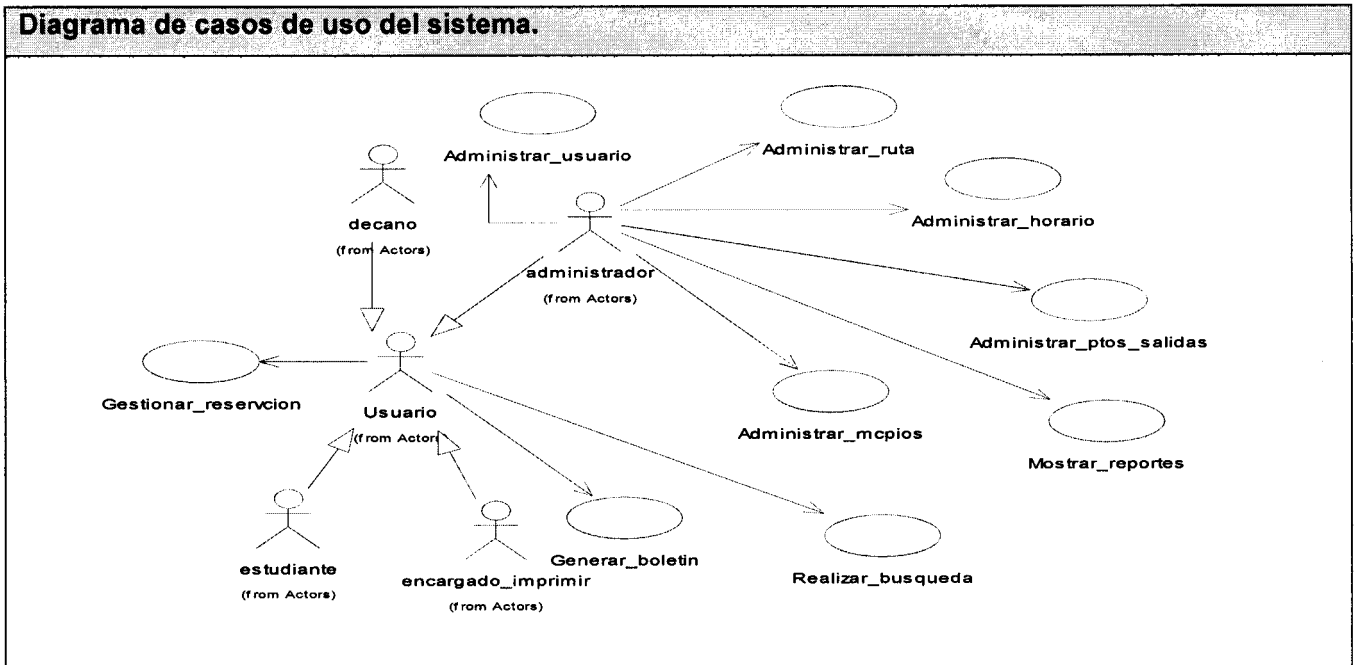
Los requerimientos no funcionales son cualidades que el sistema debe tener y que, además, son consideradas como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requisitos no funcionales son fundamentales en el éxito del sistema ya que están vinculados a requerimientos funcionales.

- **Software.**
  - Requiere de un servidor Web Apache.
  - Requiere el uso de la técnica Ajax en la implementación.
  - Requiere como gestor de base de datos a Postgre SQL.
- **Hardware.**
  - Requiere de una computadora con procesador Pentium II o un procesador superior.
- **Apariencia o interfaz externa.**
  - Diseño sencillo y con gran facilidad de uso.
  - El color predominante es el azul.
  - No presenta muchas imágenes para facilitar la navegación.
- **Seguridad.**
  - Requiere de autenticación.
  - Requiere de que las funcionalidades del sistema se muestren según los privilegios del usuario.
  - Requiere de protección a acciones no autorizadas.
- **Usabilidad.**
  - El sistema es usado por aquellas personas que son autorizadas o que lo utilizan para obtener algún beneficio.



- El sistema brinda información y la posibilidad de recibir inquietudes en caso que sea necesario.
- **Soporte.**
  - Requiere un servidor de datos con grandes volúmenes de información y velocidad de procesamiento.

**2.11 Diagrama de casos de uso del sistema.**



**Figura 2.5 Diagrama de casos de uso del sistema.**

## 2.12 Descripción de los casos de uso.

## Caso de uso "Gestionar reservación"

<b>Nombre del CU</b>	Gestionar reservación
<b>Actores</b>	Usuario (inicia)
<b>Propósito</b>	Posibilitar al usuario realizar la reservación del pase.
<b>Resumen</b>	<p>El caso de uso inicia cuando el usuario selecciona la opción de reservar, modificar o cancelar.</p> <p>a) Si decide reservar, el sistema verifica que la reservación no haya sido realizada. Si no ha sido realizada posibilita que el usuario introduzca los datos del familiar, la dirección, el parentesco, seleccione la ruta, seleccione el tipo de reservación (ida o regreso), seleccione el punto intermedio y el municipio. El sistema verifica los datos y confirma que su reservación se ha realizado. En caso de que los datos hayan sido insertados incorrectamente, el sistema muestra un mensaje de error y el usuario vuelve a introducir los datos finalizando así el caso de uso.</p> <p>b) Si decide modificar el sistema verifica que la reservación exista y posibilita que se realice la modificación. El usuario realiza las modificaciones necesarias, el sistema comprueba que los datos fueron modificados correctamente, actualizándose la reservación. Luego muestra una confirmación de la modificación. Si los datos fueron modificados incorrectamente muestra un mensaje de error, finalizando así el caso de uso.</p> <p>c) Si decide cancelar la reservación el sistema comprueba que la reservación exista y posibilita que se realice la cancelación. El sistema muestra una confirmación de la cancelación realizada, finalizando así el caso de uso.</p>
<b>Referencias</b>	R2.1, R2.2, R2.3, R2.4.
<b>Precondiciones</b>	El usuario debe estar autenticado.

<b>Poscondiciones</b>	Actualizar la base de datos de reservaciones.
<b>Prioridad</b>	Crítico

**Tabla 2.7 Descripción del caso de uso “Gestionar reservación”**

**Caso de uso “Administrar usuario”**

<b>Nombre del CU</b>	Administrar usuario.
<b>Actores</b>	Administrador (inicia).
<b>Propósito</b>	Permitir al administrador controlar los usuarios con privilegios en el sistema.
<b>Resumen</b>	<p>El administrador solicita insertar o visualizar el listado de los usuarios.</p> <p>a) Si decide insertar un nuevo usuario, el sistema busca que el usuario no exista y le permite al administrador insertar los datos del usuario: nombre y apellidos, privilegio y el área. El sistema verifica que los datos fueron insertados correctamente actualizando el listado de los usuarios del sistema. En caso de que los datos no fueron insertados correctamente, el sistema muestra un sistema de error, finalizando así el caso de uso.</p> <p>b) Si decide mostrar el listado de los usuarios, el sistema busca todos los usuarios del sistema y muestra el privilegio, el nombre, el apellido, y la foto, finalizando así el caso de uso.</p> <p>En caso de que no existan usuarios, el servicio envía un mensaje de error, finalizando así el caso de uso.</p>
<b>Referencias</b>	R3.1, R3.2, R3.3.
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	Para las siguientes acciones:

	<p>Insertar usuario: Se adiciona un nuevo usuario con los privilegios correspondientes en el sistema.</p> <p>Mostrar usuarios: Muestra todos los usuarios con privilegios en el sistema.</p>
<b>Prioridad</b>	Crítico

**Tabla 2.8 Descripción del caso de uso “Administrar usuario”**

**Caso de uso “Administrar ruta”**

<b>Nombre del CU</b>	Administrar ruta de ómnibus.
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar las rutas de los ómnibus.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador solicita insertar una ruta nueva o mostrar las rutas disponibles.</p> <p>a) Si decide insertar una nueva ruta el sistema verifica que la ruta no exista y entonces le permite insertar los datos de la ruta: número de ruta, nombre de la ruta, municipio, parada intermedia, punto de salida e itinerario. El sistema verifica que los datos son insertados correctamente, actualizando el listado de las rutas. En caso de que los datos fueron insertados incorrectamente, muestra un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide mostrar rutas el sistema busca todas las rutas disponibles y luego les muestra la ruta, el número de la ruta, el itinerario, el punto de salida y el municipio de la ruta.</p>
<b>Referencias</b>	R4.1, R4.2, R4.3.
<b>Precondiciones</b>	El administrador debe estar autenticado.

<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar ruta: Se adiciona una ruta nueva.</p> <p>Mostrar rutas: Muestra todas las rutas disponibles.</p>
<b>Prioridad</b>	Secundario

**Tabla 2.9 Descripción del caso de uso “Administrar ruta”**

**Caso de uso “Administrar municipios”**

<b>Nombre del CU</b>	Administrar municipios.
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar los municipios hacia los cuales se realizarán las transportaciones los fines de semana.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador solicita insertar o modificar un municipio.</p> <p>a) Si decide insertar el sistema verifica el municipio no existe y posibilita al administrador introducir los datos del municipio: el nombre del municipio. El sistema verifica que los datos hayan sido insertados correctamente y actualiza el listado de los municipios. En caso de que los datos sean insertados incorrectamente el sistema muestra un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide modificar un municipio el sistema verifica que el municipio exista y le posibilita al administrador realizar la modificación. Luego comprueba que los datos fueron modificados correctamente y actualiza el listado de los municipios. En caso de que los datos fueron modificados incorrectamente, muestra un</p>

	mensaje de error, finalizando así el caso de uso.	
<b>Referencias</b>	R5.1, R5.2, R5.3,	
<b>Precondiciones</b>	El administrador debe estar autenticado.	
<b>Poscondiciones</b>	Para las siguientes acciones:  Insertar municipio: Se adiciona un nuevo municipio.  Modificar municipio: Se modifica un municipio existente.	
<b>Prioridad</b>		Secundario

**Tabla 2.10 Descripción del caso de uso “Administrar municipio”**

**Caso de uso “Administrar horarios de salida y entrada”**

<b>Nombre del CU</b>	Administrar horarios de salida y entrada.	
<b>Actores</b>	Administrador (inicia)	
<b>Propósito</b>	Permitir al administrador controlar los horarios de salida del pase y del retorno del mismo.	
<b>Resumen</b>	El caso de uso inicia cuando el administrador solicita modificar el horario existente. El sistema comprueba que el horario existe y posibilita al administrador realizar la modificación. En caso de que la modificación no se realizara correctamente el sistema muestra un mensaje de error finalizando así el caso de uso.	
<b>Referencias</b>	R6.1, R6.2.	
<b>Precondiciones</b>	El administrador debe estar autenticado.	
<b>Poscondiciones</b>	Se modifica el horario existente.	
<b>Prioridad</b>		Secundario

**Tabla 2.11 Descripción del caso de uso “Administrar horarios de salida y entrada”**

**Caso de uso “Administrar puntos de salida”**

<b>Nombre del CU</b>	Administrar puntos de salida.
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar los lugares donde se ubicarán los ómnibus para la salida del pase.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador solicita insertar un nuevo punto de salida o modificar uno existente.</p> <p>a) Si decide insertar un nuevo punto de salida el sistema verifica que el punto no existe y es entonces que permite al administrador insertar el lugar del punto. Luego de insertar el dato el sistema verifica que fue insertado correctamente, actualizando así la lista de los puntos de salida. En caso de que la información haya sido insertada incorrectamente, el sistema muestra un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide modificar algún punto de salida el sistema comprueba que el punto existe y entonces le posibilita realizar la modificación. En caso de que la modificación no sea realizada correctamente, el sistema muestra un mensaje de error.</p>
<b>Referencias</b>	R7.1, R7.2, R7.3.
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar punto: Se adiciona un nuevo punto de salida.</p> <p>Modificar punto: Se modifica algún punto existente.</p>
<b>Prioridad</b>	Secundario

**Tabla 2.12 Descripción del caso de uso “Administrar puntos de salida”**

**Caso de uso “Reporte de reservaciones”**

<b>Nombre del CU</b>	Reporte de reservaciones realizadas.	
<b>Actores</b>	Administrador (inicia)	
<b>Propósito</b>	Permitir al administrador conocer la cifra de las reservaciones realizadas por facultad, por año y por ruta.	
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador solicita conocer las reservaciones realizadas por ruta, por facultad y por año.</p> <p>a) El administrador decide conocer la cifra de reservaciones realizadas por facultad y el sistema busca todas las reservaciones realizadas de esa facultad y lo muestra, finalizando así el caso de uso.</p> <p>b) El administrador decide conocer la cifra de los estudiantes que reservaron por años y el sistema le busca todas las reservaciones realizadas por año y lo muestra, finalizando así el caso de uso.</p> <p>c) El administrador decide conocer la cantidad de estudiantes que reservaron por ruta, el sistema le busca todas las reservaciones realizadas por año y se lo muestra, finalizando así el caso de uso.</p>	
<b>Referencias</b>	R.8.1, R.8.2, R.8.3, R.8.4.	
<b>Precondiciones</b>	El usuario debe estar autenticado.	
<b>Poscondiciones</b>	Mostrar el total de las reservaciones de ida o regreso realizadas correspondientes al fin de semana ya sea por ruta, por año o por facultad.	
<b>Prioridad</b>		Crítico

Tabla 2.13 Descripción del caso de uso “Reporte de reservaciones”



**Caso de uso “Generar boletines”**

<b>Nombre del CU</b>	Generar boletines.	
<b>Actores</b>	Usuario (inicia)	
<b>Propósito</b>	Permitir al usuario imprimir los boletines de los estudiantes.	
<b>Resumen</b>	El caso de uso inicia cuando el usuario (asistente de impresión o decano), solicita generar el boletín de los estudiantes. Para ello selecciona el nombre, apellidos, facultad, grupo y el tipo de reservación (de ida o regreso) del estudiante. El sistema verifica que la reservación exista y genera el boletín. En caso de que la reservación no exista muestra un mensaje de error y no genera el boletín.	
<b>Referencias</b>	R.9.1, R.9.2.	
<b>Precondiciones</b>	El usuario debe estar autenticado.	
<b>Poscondiciones</b>	Generar el boletín de cada estudiante.	
<b>Prioridad</b>		Crítico

**Tabla 2.14 Descripción del caso de uso “Generar boletines”**

**Caso de uso “Realizar búsqueda”**

<b>Nombre del CU</b>	Realizar búsqueda.	
<b>Actores</b>	Usuario (inicia)	
<b>Propósito</b>	Permitir al usuario buscar estudiantes que reservaron.	
<b>Resumen</b>	El caso de uso inicia cuando el usuario (administrador, asistente de impresión, decano) decide realizar una búsqueda avanzada de algún estudiante insertando el nombre, los apellidos, el número de solapín, seleccionando la facultad, el grupo y el tipo de reservación (ida o regreso). El sistema realiza la búsqueda y muestra al estudiante. En caso de que el	

	estudiante no haya realizado la reservación, el sistema muestra un mensaje, finalizando así el caso de uso.
<b>Referencias</b>	<b>R.10.1.</b>
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Poscondiciones</b>	Mostrar el estudiante del cual fueron insertados los datos.
<b>Prioridad</b>	Crítico

**Tabla 2.15 Descripción del caso de uso “Realizar búsqueda”**

### **2.13 Conclusiones.**

En este capítulo se hace un estudio detallado de las causas que provocaron la implementación del sistema y de los requisitos con los que cumple el sistema identificándose los actores de este proceso, los trabajadores y las actividades que efectúan los mismos según sus privilegios. Además se explica como se efectúa el proceso de reservación del pase en el sistema actual. El estudio realizado es expuesto a través del modelo del negocio, el cual incluye los modelos de casos de uso y el modelo de objeto y el diagrama de caso de uso del sistema con las especificaciones correspondientes.

## **CAPÍTULO 3: ESTUDIO DE LOS SERVICIOS**

El levantamiento de requisitos está considerado como uno de los procesos más importantes para el desarrollo de cualquier software. Es por ello que en este capítulo se identifican los requisitos tanto funcionales como no funcionales necesarios para determinar las funcionalidades de cada servicio, describiendo cada caso de uso involucrado en el modelado del sistema.

### **3.1 Levantamiento de Requerimientos.**

#### **Requerimientos funcionales.**

Los requerimientos funcionales son aquellos requerimientos que, representan capacidades que el sistema debe cumplir para satisfacer las necesidades del usuario y que contribuyen a encontrar funcionalidades del sistema implantado.

La capa de servicios del sistema de Reservación del Transporte, presenta los siguientes requisitos funcionales:

#### **R.1 Servicio de gestionar reservación.**

- R.1.1** Insertar información necesaria.
- R.1.2** Modificar información.
- R.1.3** Cancelar reservación.
- R.1.4** Buscar reservación.
- R.1.5** Verificar reservación.

#### **R.2 Servicio de administrar usuarios del sistema.**

- R.2.1** Insertar usuario.
- R.2.2** Obtener listado de usuarios.
- R.2.3** Buscar usuario.
- R.2.4** Verificar usuario.

#### **R.3 Servicio de administrar ruta de ómnibus.**

**R.3.1** Agregar una ruta.

**R.3.2** Obtener ruta.

**R.3.3** Buscar ruta.

**R.3.4** Verificar ruta.

**R.4** Servicio de administrar municipios.

**R.4.1** Insertar municipio.

**R.4.2** Modificar municipio.

**R.4.3** Buscar municipio.

**R.4.4** Verificar municipio.

**R.5** Servicio de administrar horarios de entrada y salida.

**R.5.2** Modificar horario.

**R.5.4** Buscar horario.

**R.6** Servicio de administrar puntos de salida.

**R.6.1** Insertar punto de salida de los ómnibus.

**R.6.2** Modificar puntos de salida de los ómnibus.

**R.6.3** Buscar puntos de salida.

**R.7** Servicio de obtener total de reservaciones.

**R.7.1** Obtener total de reservaciones por ruta.

**R.7.2** Obtener total de reservaciones por año.

**R.7.3** Obtener total de reservaciones por facultad.

**R.7.4** Buscar reservaciones.

**R.8** Servicio de gestionar boletín.

**R.8.1** Generar boletín.

**R.8.2** Buscar estudiante.

**R.9** Servicio de búsqueda.

### Requerimientos no funcionales.

Los requerimientos no funcionales son las propiedades que debe presentar el sistema, por lo que, son consideradas características del sistema. A continuación se muestran los requerimientos no funcionales del sistema con la utilización de los servicios Web.

- **Software**
  - Requiere de un servidor Web Apache.
  - Requiere el uso de la técnica Ajax en la implementación.
  - Requiere como gestor de base de datos a Postgre SQL.
  - Requiere de las tecnologías de los servicios Web.
- **Hardware.**
  - Requiere de una computadora con procesador Pentium II o un procesador superior.
- **Apariencia o interfaz externa.**
  - Diseño sencillo y con gran facilidad de uso.
  - El color predominante es el azul.
  - No presenta muchas imágenes para facilitar la navegación.
- **Seguridad.**
  - Requiere de autenticación.
  - Requiere de que las funcionalidades del sistema se muestren según el privilegio del usuario.
  - Requiere de protección a acciones no autorizadas.
- **Usabilidad.**
  - El sistema es usado por aquellas personas o sistema que son autorizadas a su uso o que lo utilizan para beneficio.
  - El sistema brinda información y la posibilidad de recibir inquietudes en caso que sea necesario.

- **Soporte.**

- Requiere un servidor de datos con grandes volúmenes de información y velocidad de procesamiento.

### **3.2 Casos de uso determinados.**

Los casos de uso no son más que funcionalidades de los sistemas para aportar un resultado a los actores. Los casos de uso son los que se muestran a continuación:

- Servicio administrar horarios de entrada y salida.
- Servicio administrar puntos de salida.
- Servicio administrar municipio.
- Servicio administrar ruta.
- Servicio administrar usuario.
- Servicio generar boletín.
- Servicio búsqueda.
- Servicio reportar reservaciones.
- Servicio gestionar reservación.

3.3 Diagrama de casos de uso.

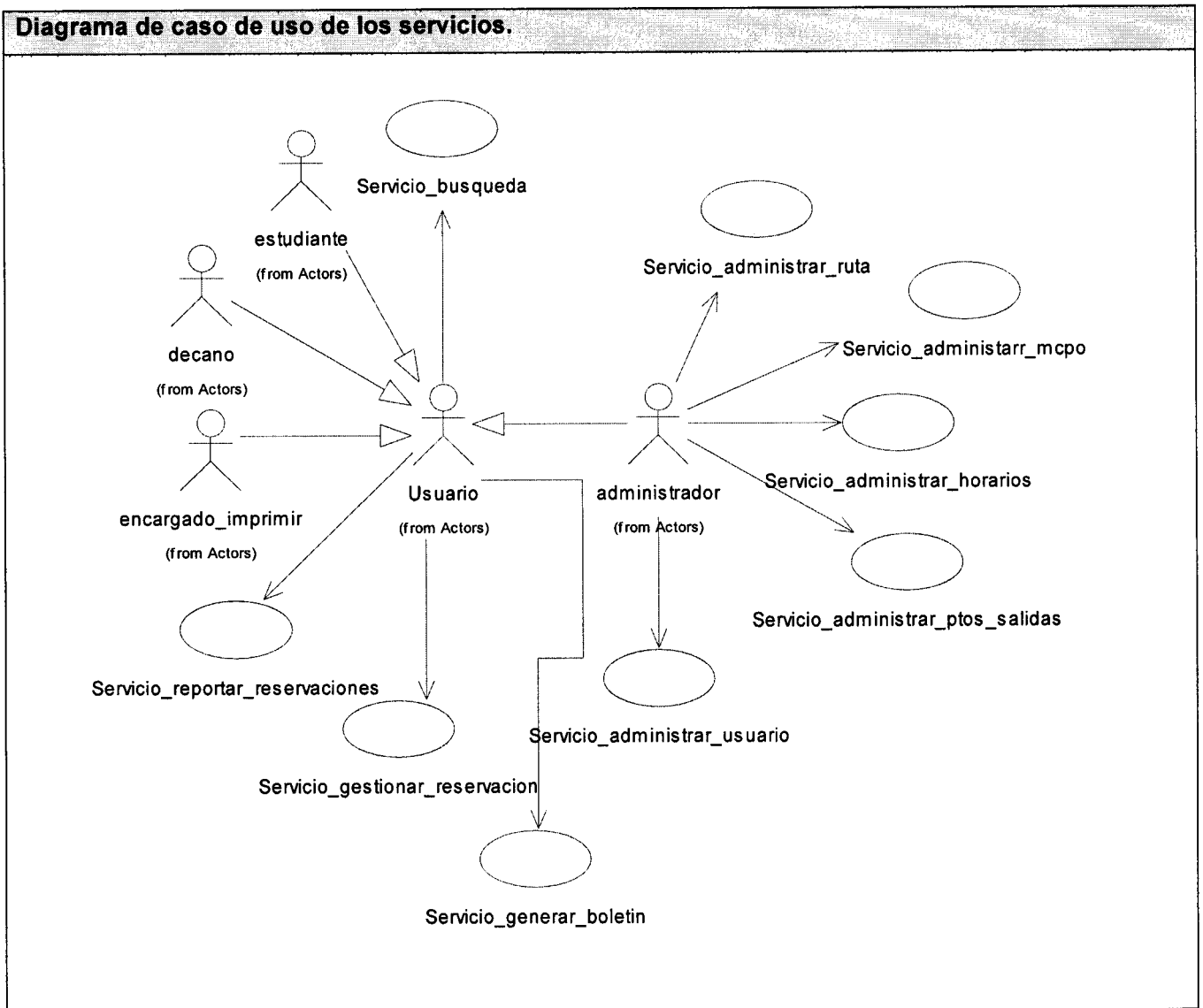


Figura 3.1 Diagrama de casos de uso de la capa de servicios.

3.4 Descripción de los casos de uso.

Caso de uso “Servicio gestionar reservación”

<b>Nombre del CU</b>	Servicio gestionar reservación
<b>Actores</b>	Usuario (inicia)
<b>Propósito</b>	Posibilitar al usuario realizar la reservación del pase.
<b>Resumen</b>	<p>El caso de uso inicia cuando el usuario (estudiante o decano) selecciona la opción de reservar, modificar o cancelar.</p> <p>a) Si decide reservar, el usuario invoca al servicio pasándole la dirección del familiar (calle, número de apartamento, número de la casa, entrecalles, localidad), el tipo de familiar, el tipo de reservación (ida o regreso), la ruta, el punto intermedio y el municipio. El servicio verifica que la reservación no ha sido realizada e inserta los datos en la base de datos correspondiente. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que la reservación existiera, el servicio envía un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide modificar, el usuario invoca al servicio y en la llamada le pasa los datos a modificar y los nuevos datos. El servicio verifica que la reservación existe y entonces modifica los datos, actualizando las bases de datos correspondientes. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que la reservación no exista, el servicio envía un mensaje de error, finalizando así el caso de uso.</p> <p>c) Si decide cancelar la reservación, el usuario invoca al servicio y el este comprueba que la reservación exista y posibilita que se realice la cancelación. En respuesta le envía un mensaje de confirmación, finalizando así el caso de</p>



	<p>uso.</p> <p>En caso de que la reservación no exista el servicio envía un mensaje de error, finalizando así el caso de uso.</p>
<b>Referencias</b>	<b>R2.1, R2.2, R2.3, R2.4.</b>
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Reservar: Se realiza una nueva reservación.</p> <p>Modificar: Se modifica la reservación.</p> <p>Cancelar: Se cancela la reservación.</p>
<b>Prioridad</b>	Crítico

**Tabla 3.1 Descripción del caso de uso “Servicio gestionar reservación”**

**Caso de uso “Servicio administrar usuarios del sistema”**

<b>Nombre del CU</b>	Servicio administrar usuario
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar los usuarios con privilegios en el sistema.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador solicita al servicio para insertar o mostrar listado de usuarios.</p> <p>a) Si decide insertar un nuevo usuario, el administrador invoca el servicio y con la llamada viaja el nombre y apellidos del usuario, el privilegio. El servicio verifica que el usuario no exista en la base de datos y lo inserta. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p>

	<p>En caso de que el usuario exista el servicio envía un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide mostrar el listado de los usuarios, el administrador invoca el servicio, y este realiza una búsqueda de los usuarios en la base de datos correspondiente. En respuesta envía un arreglo con el nombre, los apellidos y el privilegio de cada usuario, finalizando así el caso de uso.</p> <p>En caso de que no existan usuarios el servicio muestra un mensaje de error, finalizando así el caso de uso.</p>
<b>Referencias</b>	<b>R3.1, R3.2, R3.3.</b>
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar usuario: Se adiciona un nuevo usuario con privilegios en el sistema.</p> <p>Mostrar usuarios: Obtiene todos los usuarios con privilegios existentes.</p>
<b>Prioridad</b>	Crítico

**Tabla 3.2 Descripción del caso de uso “Servicio administrar usuario del sistema”**

**Caso de uso “Servicio administrar ruta de ómnibus”**

<b>Nombre del CU</b>	Servicio administrar ruta de ómnibus.
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar las rutas de los ómnibus.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador invoca al servicio para insertar una ruta nueva o mostrar las rutas disponibles.</p> <p>a) Si decide insertar una nueva ruta el administrador invoca al servicio, y le pasa</p>

	<p>la nueva ruta, el municipio, el número de la ruta, el punto de salida y el itinerario de la ruta. El servicio verifica en la base de datos que la ruta no exista e inserta los datos de la ruta correspondiente en la base de datos. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que la ruta exista, envía un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide mostrar rutas, el administrador invoca al servicio, y este busca en la base de datos todas las rutas disponibles. En respuesta le envía un arreglo con el nombre de cada ruta, el municipio, el número, el itinerario y los puntos de salida correspondientes a cada ruta, finalizando así el caso de uso.</p> <p>En caso de que no existan rutas el sistema envía un mensaje de error, finalizando así el caso de uso.</p>
<b>Referencias</b>	<b>R4.1, R4.2, R4.3.</b>
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar ruta: Se adiciona una ruta nueva.</p> <p>Obtener rutas: Muestra todas las rutas existentes.</p>
<b>Prioridad</b>	Secundario

**Tabla 3.3 Descripción del caso de uso “Servicio administrar ruta de ómnibus”**

**Caso de uso “Servicio administrar municipios”**

<b>Nombre del CU</b>	Servicio administrar municipios.
<b>Actores</b>	Administrador (inicia)
<b>Propósito</b>	Permitir al administrador controlar los municipios hacia los cuales se realizarán

	las transportaciones los fines de semana.
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador invoca al servicio para insertar o modificar un municipio.</p> <p>a) Si decide insertar un municipio, el administrador invoca el servicio, y en la llamada viaja el nuevo municipio a insertar. El servicio busca en la base de datos que el municipio no exista e inserta el nuevo municipio, actualizando así la base de datos de correspondiente a municipios. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que el municipio ya exista, el servicio envía un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide modificar municipio, el administrador invoca el servicio pasando en la llamada el municipio a modificar y el nuevo municipio. El servicio verifica que el municipio exista y realiza la modificación correspondiente, actualizando así la base de datos correspondiente a municipios. En respuesta envía un mensaje de confirmación.</p> <p>En caso de que el municipio no existiera, el servicio envía un mensaje de error, finalizando así el caso de uso.</p>
<b>Referencias</b>	R5.1, R5.2, R5.3.
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar municipio: Se adiciona un nuevo municipio.</p> <p>Modificar municipio: Se modifica un municipio existente.</p>
<b>Prioridad</b>	Secundario.

Tabla 3.4 Descripción del caso de uso “Servicio administrar municipios”

**Caso de uso “Servicio administrar horarios de salida y entrada”**

<b>Nombre del CU</b>	Servicio administrar horarios de salida y entrada.	
<b>Actores</b>	Administrador (inicia)	
<b>Propósito</b>	Permitir al administrador controlar los horarios de salida del pase y del retorno del mismo.	
<b>Resumen</b>	<p>El caso de uso inicia cuando el administrador invoca al servicio, y con la llamada viaja el horario existente y el nuevo horario. El servicio comprueba que el horario existe y modifica en la base de datos, insertando el nuevo horario. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que el horario no exista, el servicio envía un mensaje de error finalizando así el caso de uso.</p>	
<b>Referencias</b>	R6.1, R6.2.	
<b>Precondiciones</b>	El administrador debe estar autenticado.	
<b>Poscondiciones</b>	Se modifica el horario existente.	
<b>Prioridad</b>		Secundario

Tabla 3.5 Descripción del caso de uso “Servicio administrar horario de salida y entrada”

**Caso de uso “Servicio administrar puntos de salida”**

<b>Nombre del CU</b>	Servicio administrar puntos de salida.	
<b>Actores</b>	Administrador (inicia)	
<b>Propósito</b>	Permitir al administrador controlar los lugares donde se ubicarán los ómnibus para la salida del pase.	
<b>Resumen</b>	El caso de uso inicia cuando el administrador solicita insertar un nuevo punto de salida o modificar uno existente.	

	<p>a) Si decide insertar un nuevo punto de salida el administrador invoca al servicio y con la llamada viajan los datos del punto de salida a insertar. El servicio verifica que el punto no existe e inserta en la base de datos el punto correspondiente, actualizando así la lista de los puntos de salida. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que el punto exista en la base de datos, el servicio envía un mensaje de error, finalizando así el caso de uso.</p> <p>b) Si decide modificar algún punto el administrador llama al servicio y envía el municipio que va a modificar y el nuevo municipio. El servicio comprueba que el punto existe y entonces realiza la modificación pertinente, actualizando la base de datos. En respuesta envía un mensaje de confirmación, finalizando así el caso de uso.</p> <p>En caso de que el punto no exista no sea realizada la modificación y en respuesta envía un mensaje de error, el</p>
<b>Referencias</b>	R7.1, R7.2, R7.3.
<b>Precondiciones</b>	El administrador debe estar autenticado.
<b>Poscondiciones</b>	<p>Para las siguientes acciones:</p> <p>Insertar punto: Se adiciona un nuevo punto de salida.</p> <p>Modificar punto: Se modifica algún punto existente.</p>
<b>Prioridad</b>	Secundario

**Tabla 3.6 Descripción del caso de uso “Servicio administrar puntos de salida”**

**Caso de uso “Servicio de reporte de reservaciones”**

<b>Nombre del CU</b>	Servicio de reporte de reservaciones realizadas.
<b>Actores</b>	Administrador (inicia)

<b>Propósito</b>	Permitir al administrador conocer la cifra de las reservaciones realizadas ya sea por facultad, por ruta o por año.
<b>Resumen</b>	<p>El administrador solicita conocer la cifra de los estudiantes que reservaron ya sea por facultad, por año o por ruta.</p> <p>a) El administrador invoca al servicio para conocer la cantidad de reservaciones por facultad, el servicio realiza una búsqueda en la base de datos de las reservaciones hechas, y selecciona las reservaciones por facultad. En respuesta, las envía mediante un arreglo que contiene el número de cada facultad con la cantidad de reservaciones realizadas, finalizando así el caso de uso. En caso de que no existan reservaciones el servicio envía un mensaje de error.</p> <p>b) El administrador invoca al servicio para conocer la cifra de los estudiantes que reservaron por años y el servicio realiza una búsqueda en la base de datos de todas las reservaciones realizadas, seleccionando las reservaciones por año. En respuesta envía, mediante un arreglo, la cantidad de reservaciones realizadas por año, finalizando así el caso de uso. En caso de que no existan reservaciones el sistema envía un mensaje de error.</p> <p>c) El administrador invoca al servicio para conocer la cantidad de estudiantes que reservaron por ruta, el servicio realiza una búsqueda en la base de datos de todas las reservaciones realizadas por año y en respuesta, lo envía mediante un arreglo que contiene las rutas con la cantidad de reservaciones realizadas, finalizando así el caso de uso. En caso de que no existan reservaciones el servicio envía un mensaje de error</p>
<b>Referencias</b>	R.8.1, R.8.2, R.8.3, R.8.4.
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Poscondiciones</b>	Para mostrar la cifra de las reservaciones se obtiene el total de las reservaciones realizadas por facultad, por año y por ruta.
<b>Prioridad</b>	Crítico

Tabla 3.7 Descripción del caso de uso “Servicio reporte de reservaciones”

## Caso de uso “Servicio generar boletines”

<b>Nombre del CU</b>	Servicio generar boletines.
<b>Actores</b>	Usuario (inicia)
<b>Propósito</b>	Permitir al usuario imprimir los boletines de los estudiantes.
<b>Resumen</b>	El caso de uso inicia cuando el usuario (encargado de impresión, decano), invoca el servicio para generar el boletín de los estudiantes. Para ello el usuario selecciona el nombre, apellidos, facultad, grupo y el tipo de reservación (de ida o regreso) del estudiante, datos que serán enviados con la llamada al servicio. El servicio verifica que la reservación exista y posibilita a que se genere el boletín. En caso de que la reservación no exista envía un mensaje de error y no posibilita que se genere el boletín, finalizando así el caso de uso.
<b>Referencias</b>	R.9.1, R.9.2.
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Poscondiciones</b>	Genera un boletín para cada estudiante que reservó.
<b>Prioridad</b>	Crítico

Tabla 3.8 Descripción del caso de uso “Servicio generar boletines”

## Caso de uso “Servicio búsqueda”

<b>Nombre del CU</b>	Realizar búsqueda.
<b>Actores</b>	Usuario (inicia)
<b>Propósito</b>	Permitir al usuario buscar estudiantes que reservaron.
<b>Resumen</b>	El caso de uso inicia cuando el usuario invoca al servicio seleccionando la



	opción de buscar la reservación de un estudiante, donde en esa llamada viaja la facultad seleccionada, el grupo, el nombre, los apellidos y el número de solapín del estudiante. El servicio verifica en la base de datos que el estudiante haya reservado y obtiene los datos del estudiante, mostrándolos mediante un arreglo que contiene el nombre, los apellidos y el tipo de reservación que solicitó. En caso de que el estudiante no haya reservado, envía un mensaje de error, finalizando así el caso de uso.
<b>Referencias</b>	R.10.1.
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Poscondiciones</b>	Busca el estudiante del cual fueron insertados los datos.
<b>Prioridad</b>	Crítico

**Tabla 3.9 Descripción del caso de uso “Servicio búsqueda”**

**3.4 Conclusiones.**

En este capítulo se realiza un análisis y estudio detallado de los aspectos que son necesarios para poder diseñar la capa de servicios. Todo el análisis y estudio realizado, ha permitido conocer y recopilar los elementos que posibilitan a pasar a la próxima fase que propone RUP, para de esa forma, diseñar la capa de servicios propuesta para la solución del problema.

## CAPÍTULO 4: PROPUESTA DE SOLUCIÓN

En este capítulo se describe de manera general todo el proceso relacionado con el análisis y diseño de los servicios correspondientes a la capa de servicio propuesta para el sistema de gestión del pase estudiantil de la UCI. Las actividades contempladas en el análisis permiten dar una visión del sistema, analizando los requisitos funcionales, por lo que su objetivo es proyectar que hace el sistema. Sin embargo, el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, por lo que su objetivo es cumplir las expectativas del sistema.

### 4.1 Diagramas de clases del análisis.

Un diagrama de clases del análisis representa las cosas del mundo real, no de la implementación automatizada de las mismas. Es conocido como un artefacto en el que se representan los conceptos en un dominio del problema.

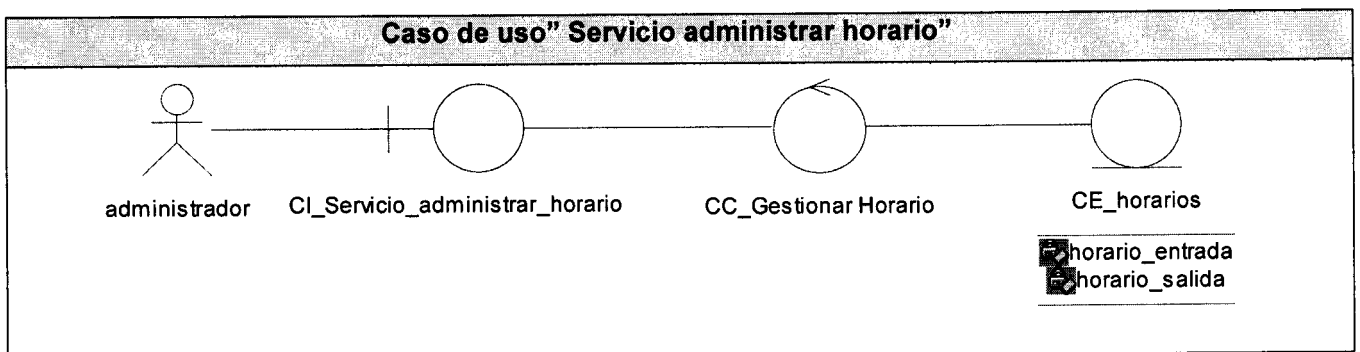


Figura 4.1 Diagrama de clases del caso de uso" Servicio administrar horario"

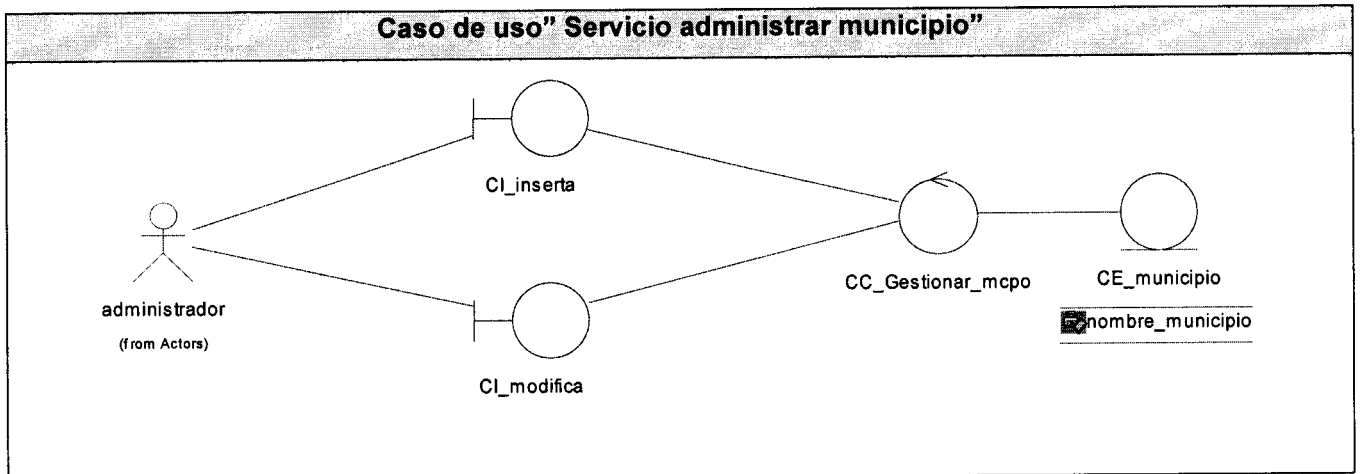


Figura 4.2 Diagrama de clases del caso de uso" Servicio administrar municipio"

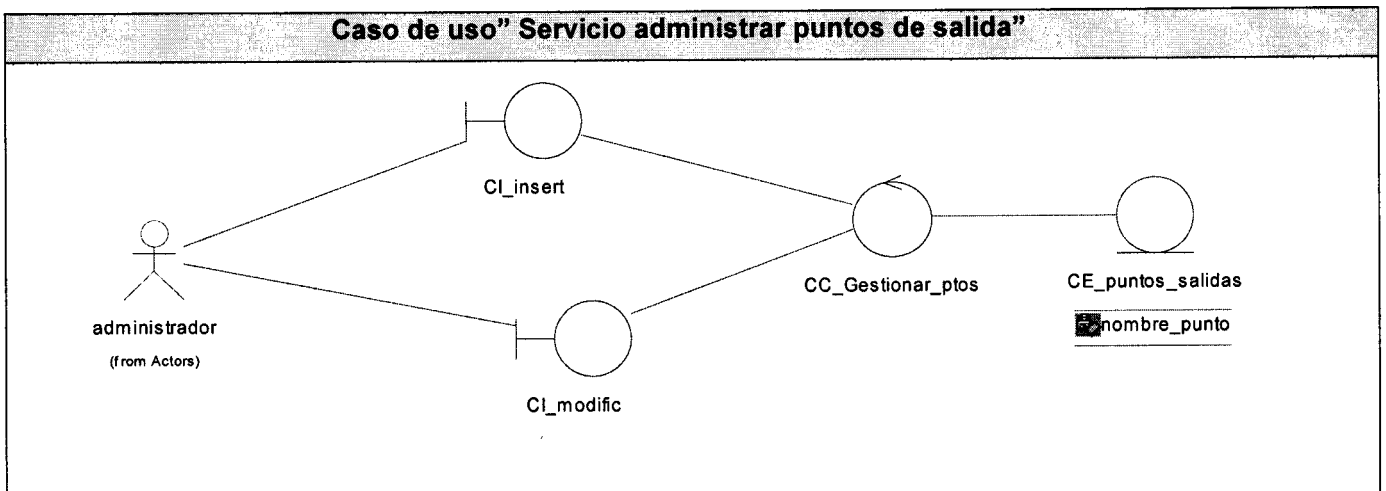


Figura 4.3 Diagrama de clases del caso de uso" Servicio administrar puntos de salida"

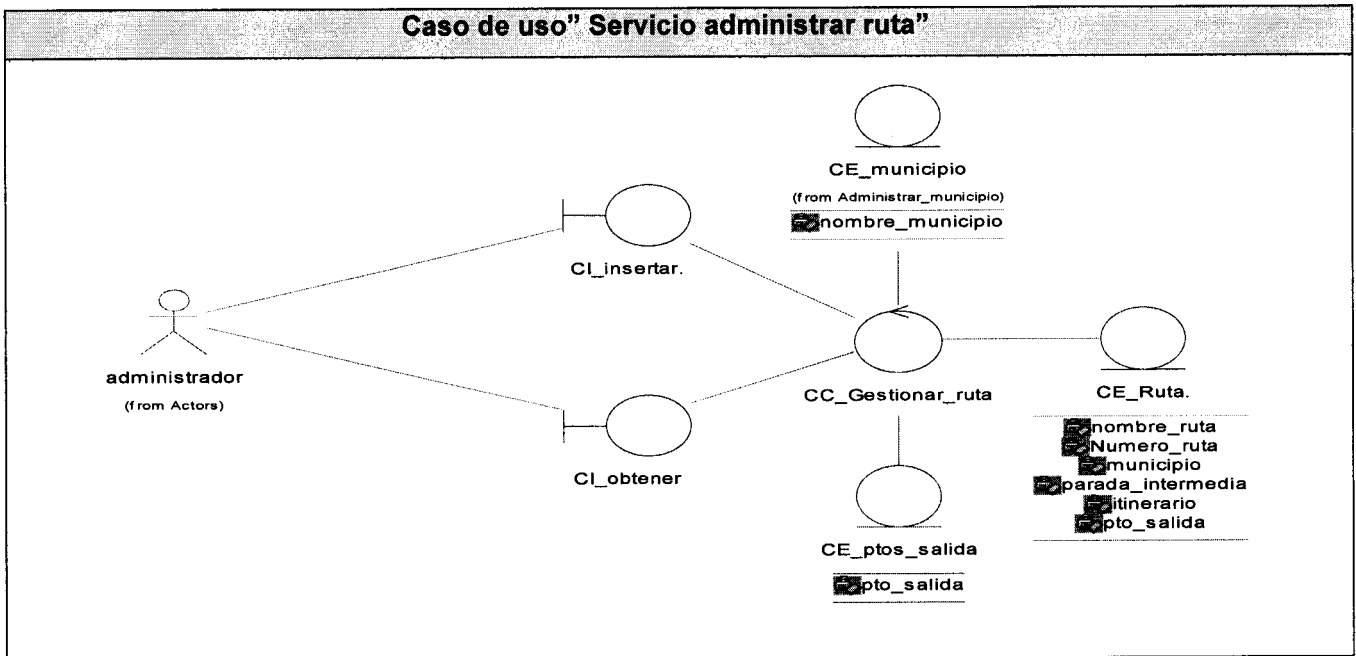


Figura 4.4 Diagrama de clases del caso de uso "Servicio administrar ruta"

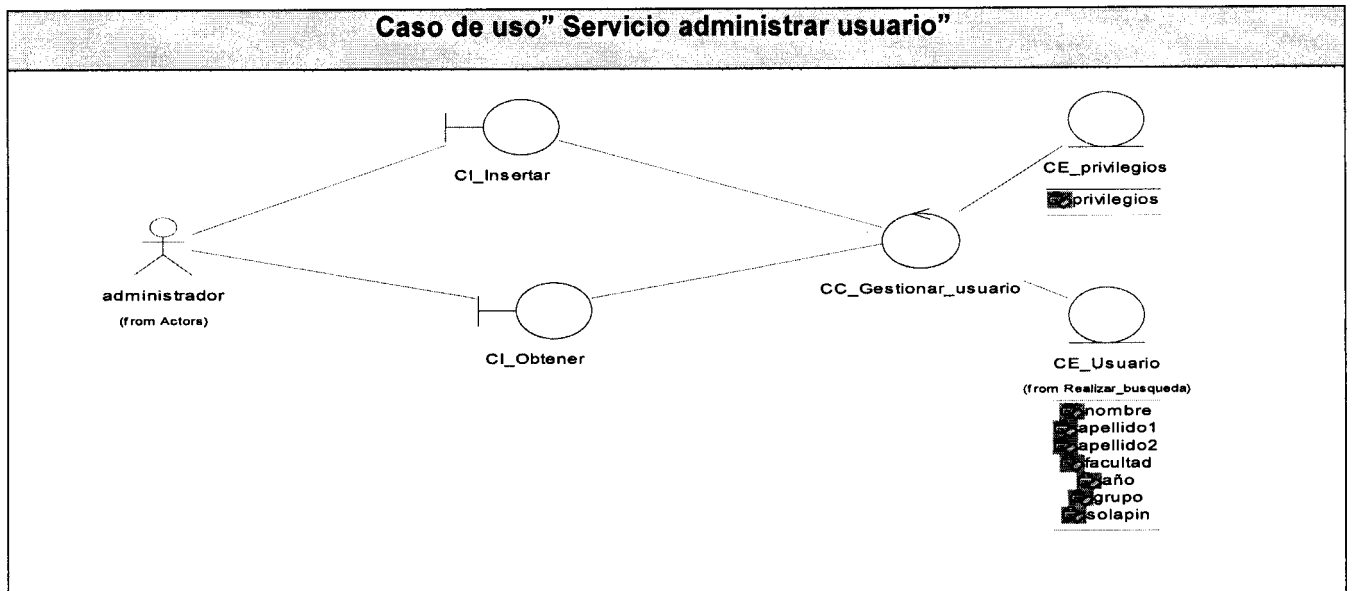


Figura 4.5 Diagrama de clases del caso de uso" Servicio administrar usuario"

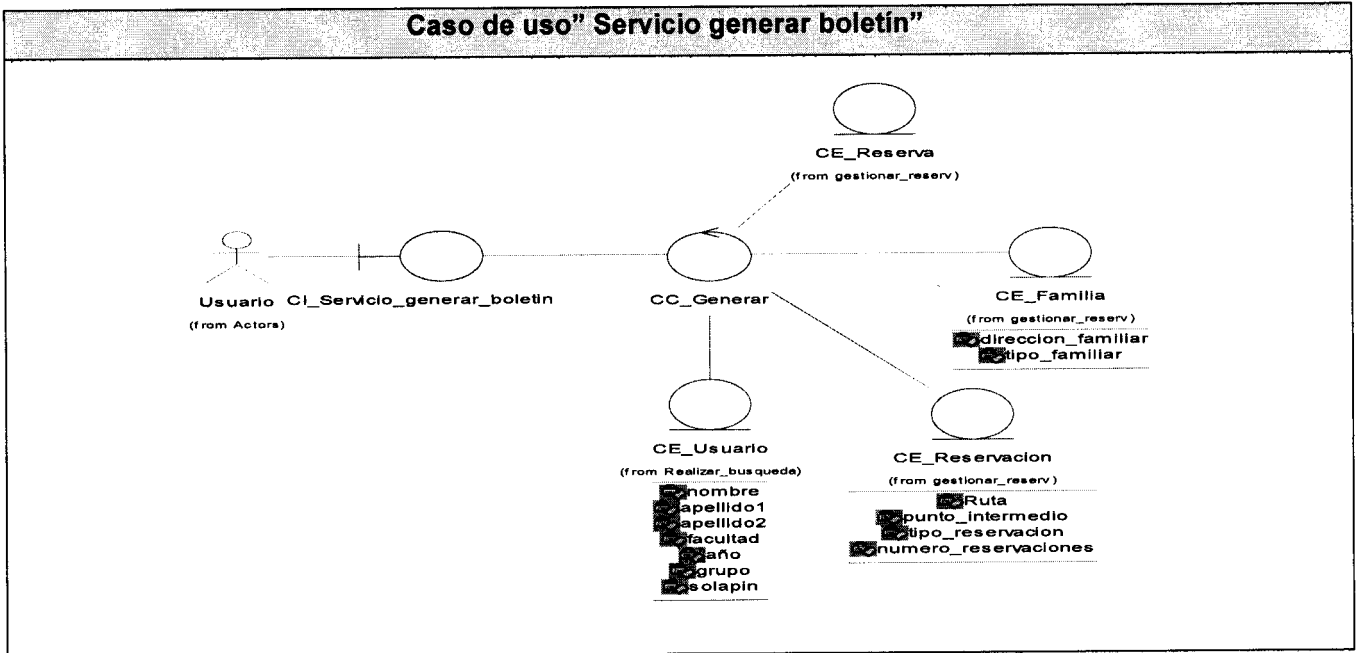


Figura 4.6 Diagrama de clases del caso de uso" Servicio generar boletín"

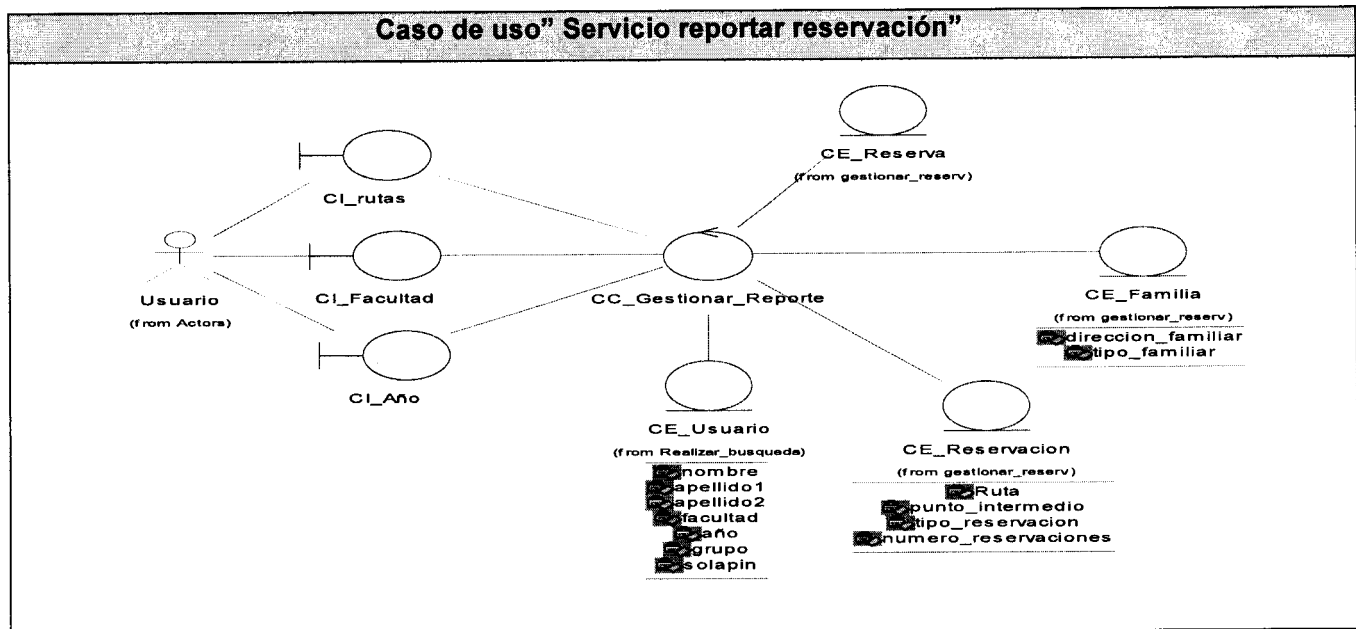


Figura 4.7 Diagrama de clases del caso de uso" Servicio reportar reservación"

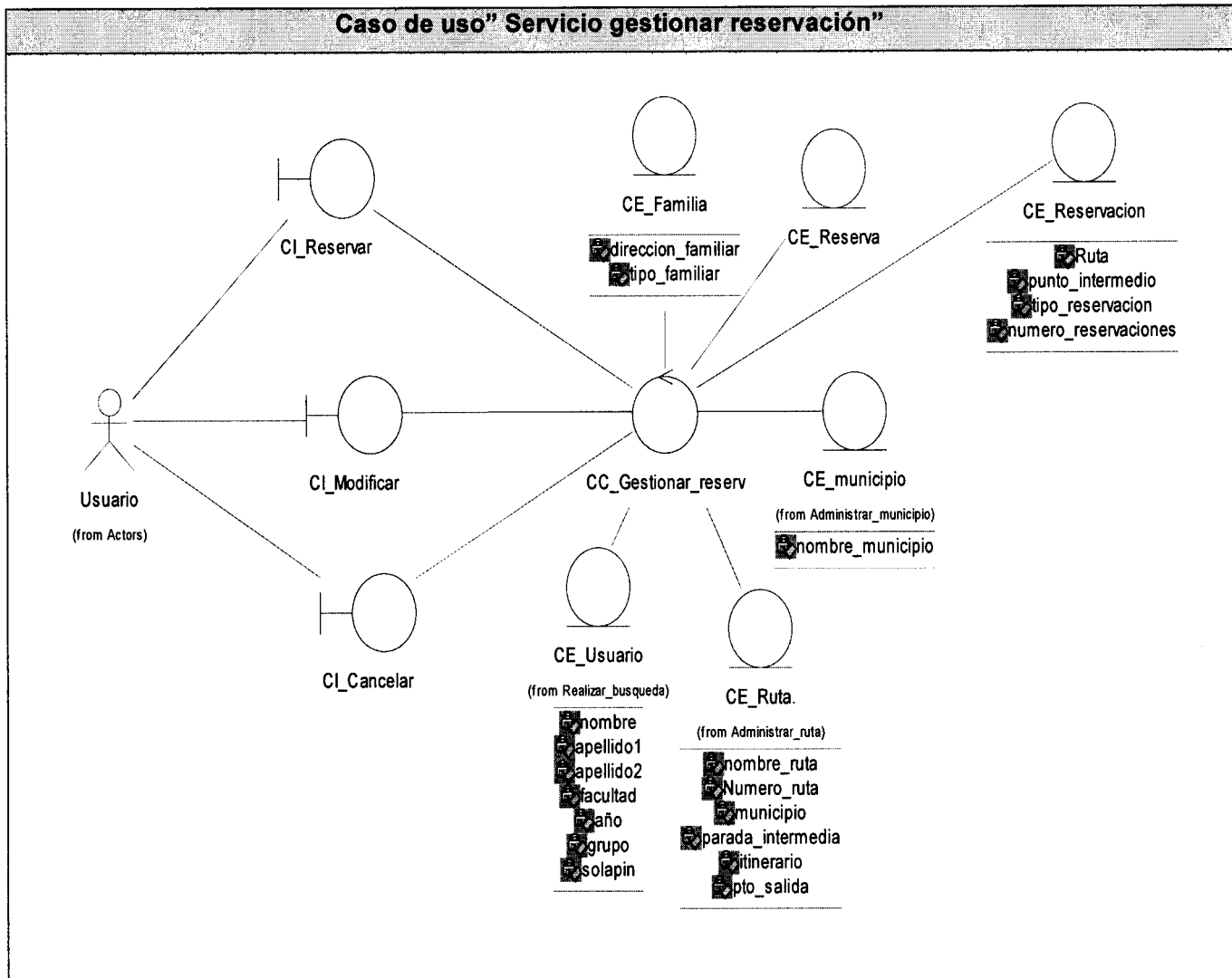


Figura 4.8 Diagrama de clases del caso de uso " Servicio gestionar reservación "

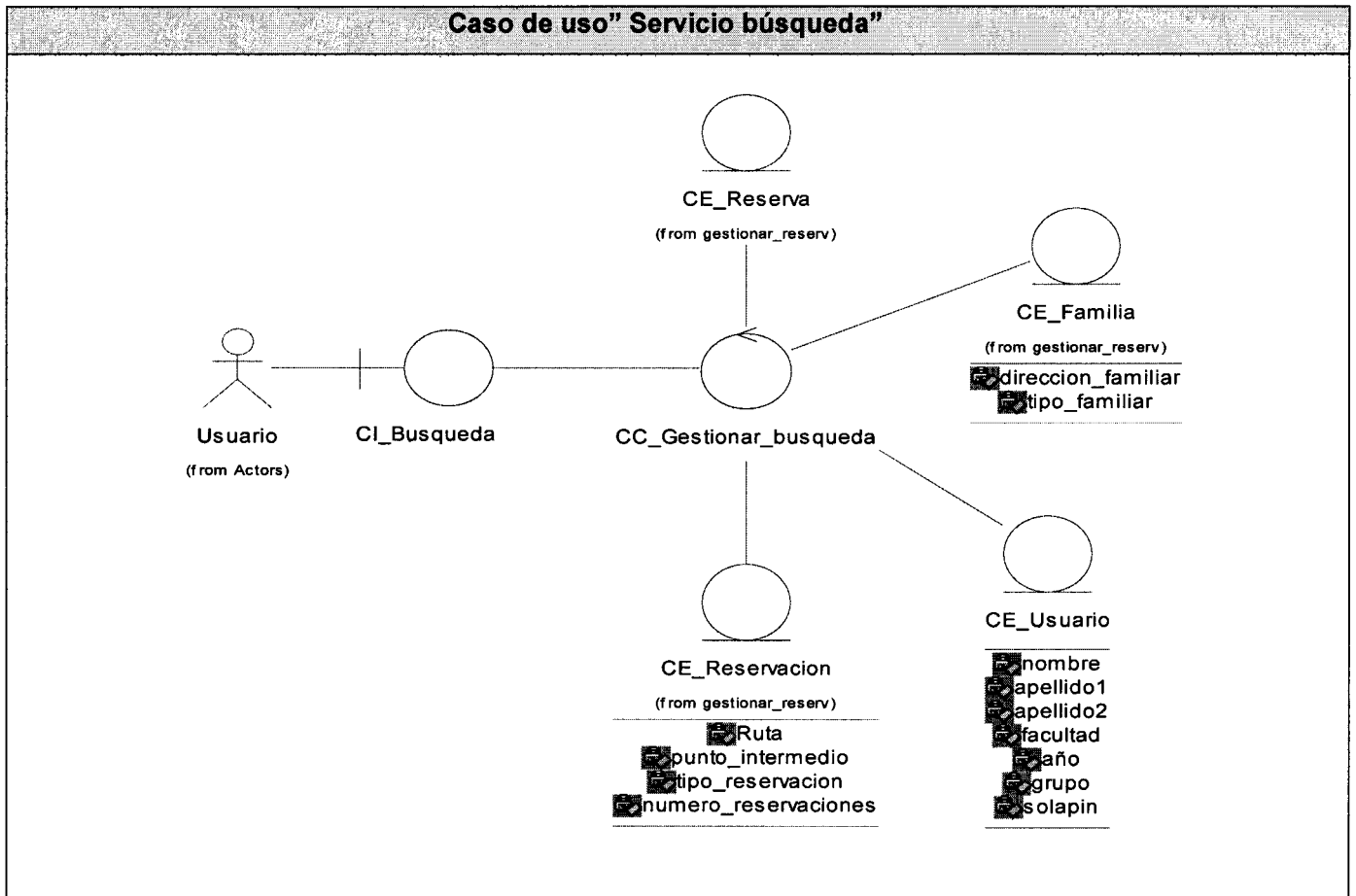


Figura 4.9 Diagrama de clases del caso de uso" Servicio búsqueda"

#### 4.2 Modelo del diseño.

En el diseño se modela el sistema y se diseña, la arquitectura que puede soportar todos los requisitos funcionales y no funcionales, y las restricciones que se le asignan. La entrada esencial de este flujo es el resultado del análisis, o sea el modelo del análisis, que proporciona una comprensión detallada de los requisitos.

En capítulos anteriores se analizó la arquitectura que emplean los servicios Web, la cual proporcionan una estructura clara y comprensible a la aplicación. La arquitectura orientada a servicios está definida por niveles, los cuales son: nivel de presentación, nivel de aplicación y nivel de persistencia. En el nivel de

presentación están las páginas que interactúan con el usuario y muestran la peticiones que se solicitaron al nivel de aplicación. En el nivel de aplicación se encuentra la lógica del negocio en forma de servicios Web, la cual se divide en dos niveles, la fachada del negocio y la lógica del negocio. Por último se encuentra el nivel de persistencia el cual controla el acceso a los datos y se encuentra constituido por dos niveles, acceso a datos y los datos como tal. A continuación se muestra las clases del modelo de diseño y la arquitectura de estos servicios mediante el diagrama de clases del diseño.

### Clases del diseño.

Las clases de diseño son clases con un mayor nivel de detalle, que se conciben para satisfacer los requisitos funcionales y los no funcionales, teniendo en cuenta la tecnología en la cual se implementará. A continuación se describen las clases del diseño propuesto:

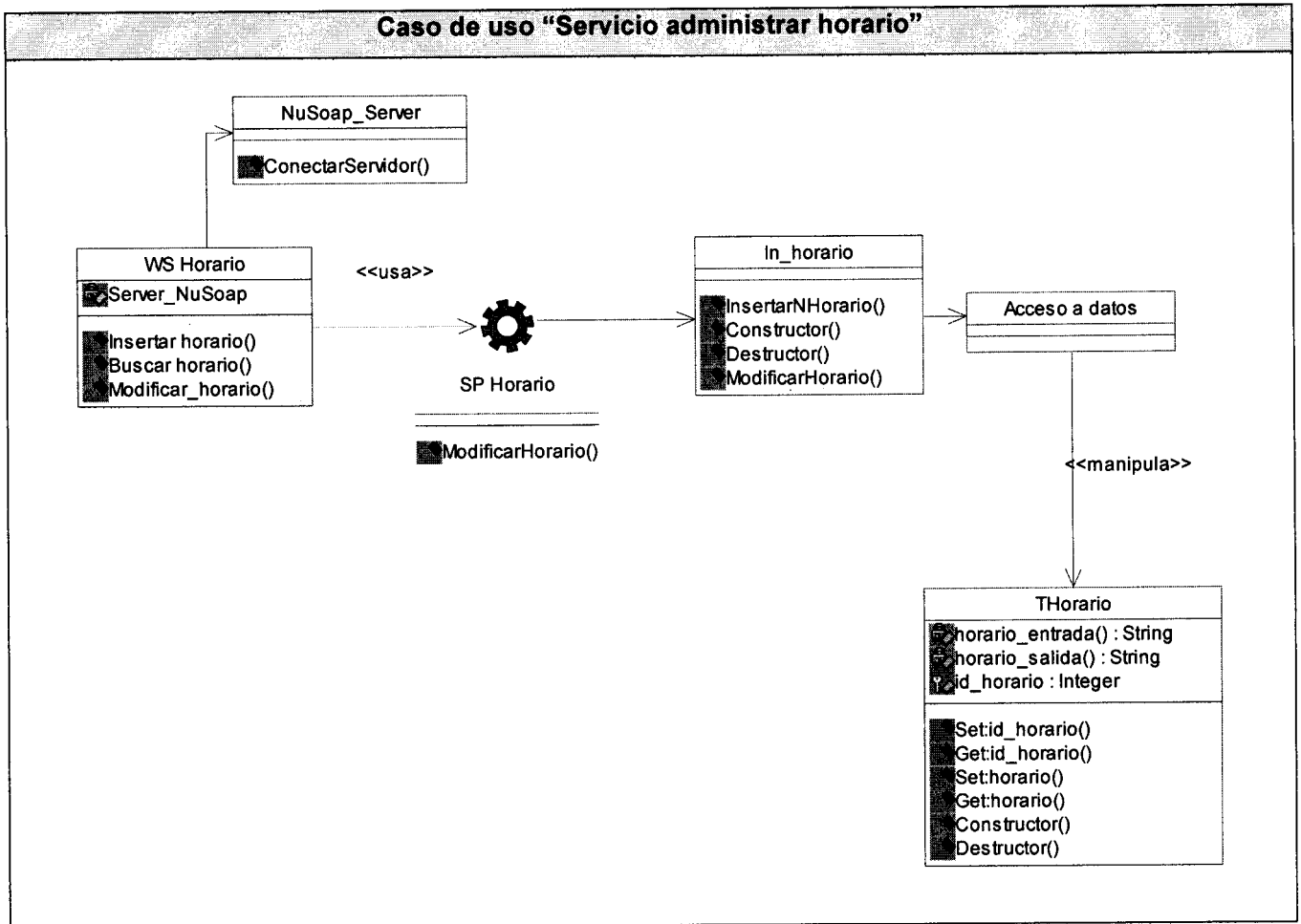
Clases	Descripción
Servicios Web	Son las clases con prefijos WS, y poseen en su estructura todos los métodos que invoca el usuario.
Páginas servidoras	Son las clases con prefijos SP que dan soporte y controlan el acceso a las clases lógicas.
Lógica de negocio	Son las clases con prefijo In y presentan en sus métodos todas las operaciones específicas para cada caso de uso.
Acceso a datos	Es la encargada de acceder a la base de datos para leer, escribir o verificar el estado de los objetos, liberando a la aplicación del acceso a la base de datos. Contiene todas las operaciones que se van a realizar en el sistema.
Tablas	Son las clases con prefijo T, y poseen en su estructura todos los datos que van a ser utilizados por la clase de acceso a datos.

Tabla 4.1 Descripción de las clases del diseño”



**Diagrama de clases del diseño.**

Los diagramas de clases del diseño, se utilizan para modelar la vista de diseño estática de un sistema. A continuación se muestran los diagramas de clases del diseño correspondiente a cada caso de uso y los diagramas de interacción correspondientes. (Ver Anexo 6)



**Figura 4.10 Diagrama de clases del diseño del caso de uso "Servicio administrar horario"**

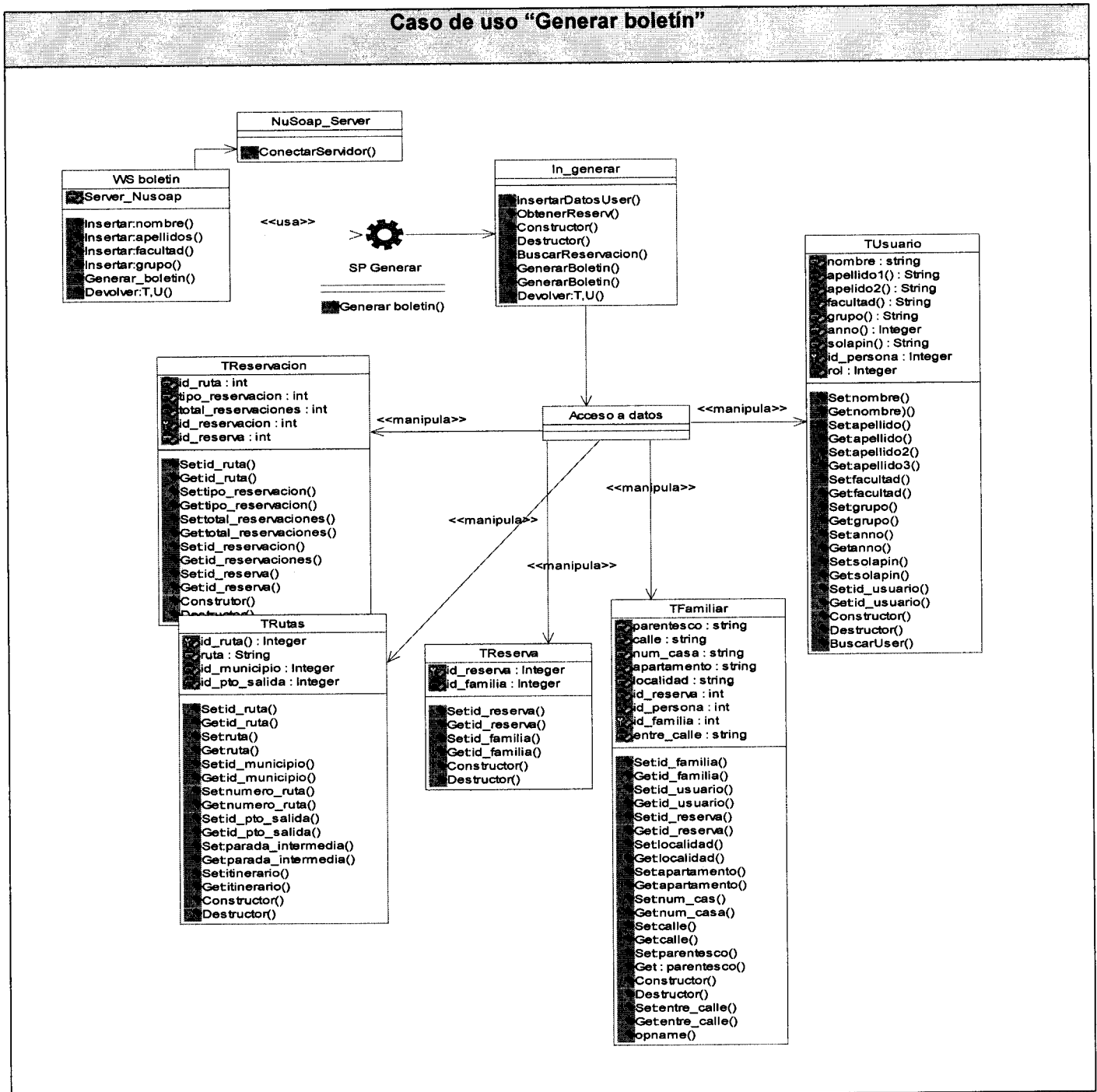


Figura 4.11 Diagrama de clases del diseño del caso de uso "Servicio generar boletín"

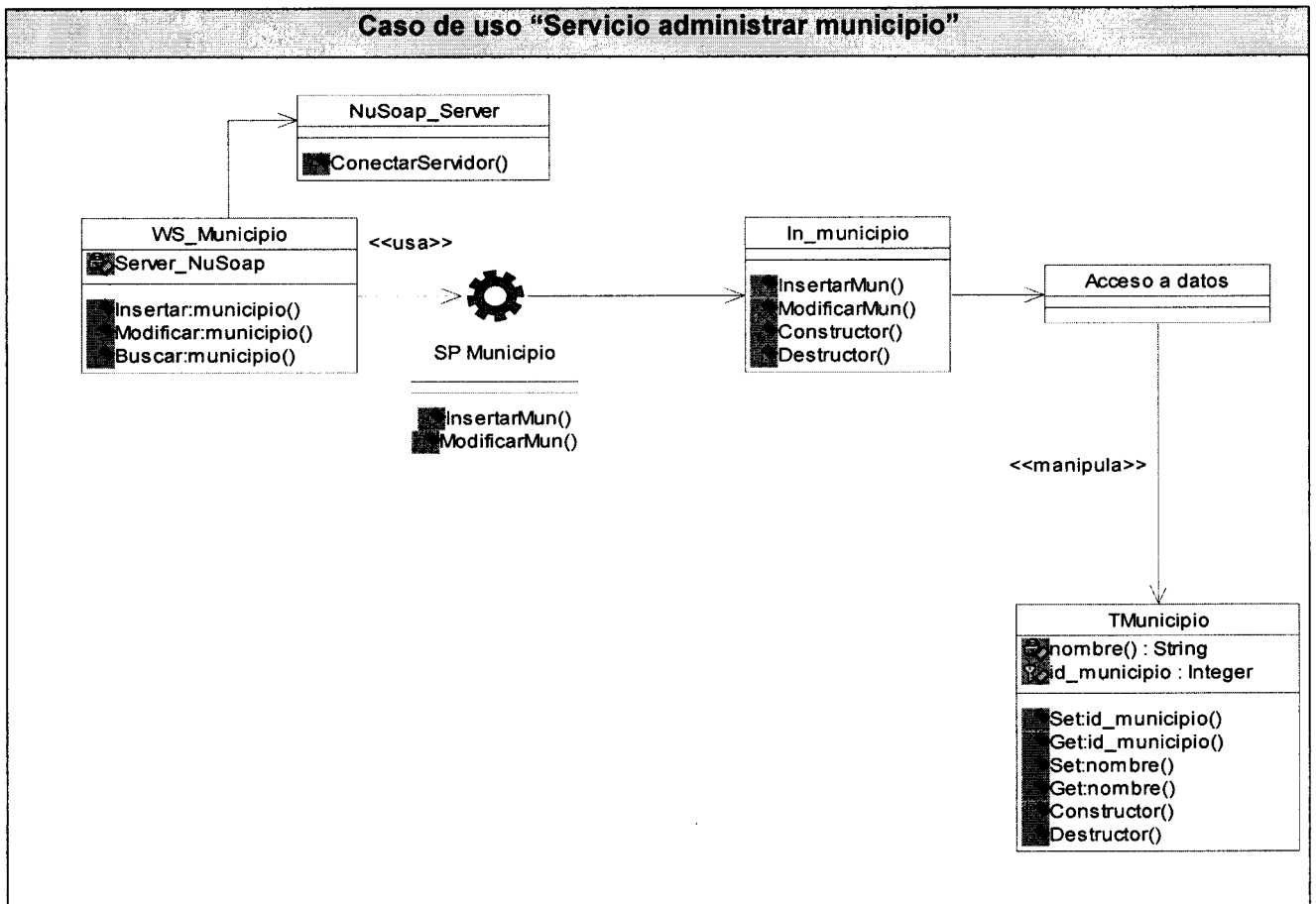


Figura 4.12 Diagrama de clases del diseño del caso de uso "Servicio administrar municipio"

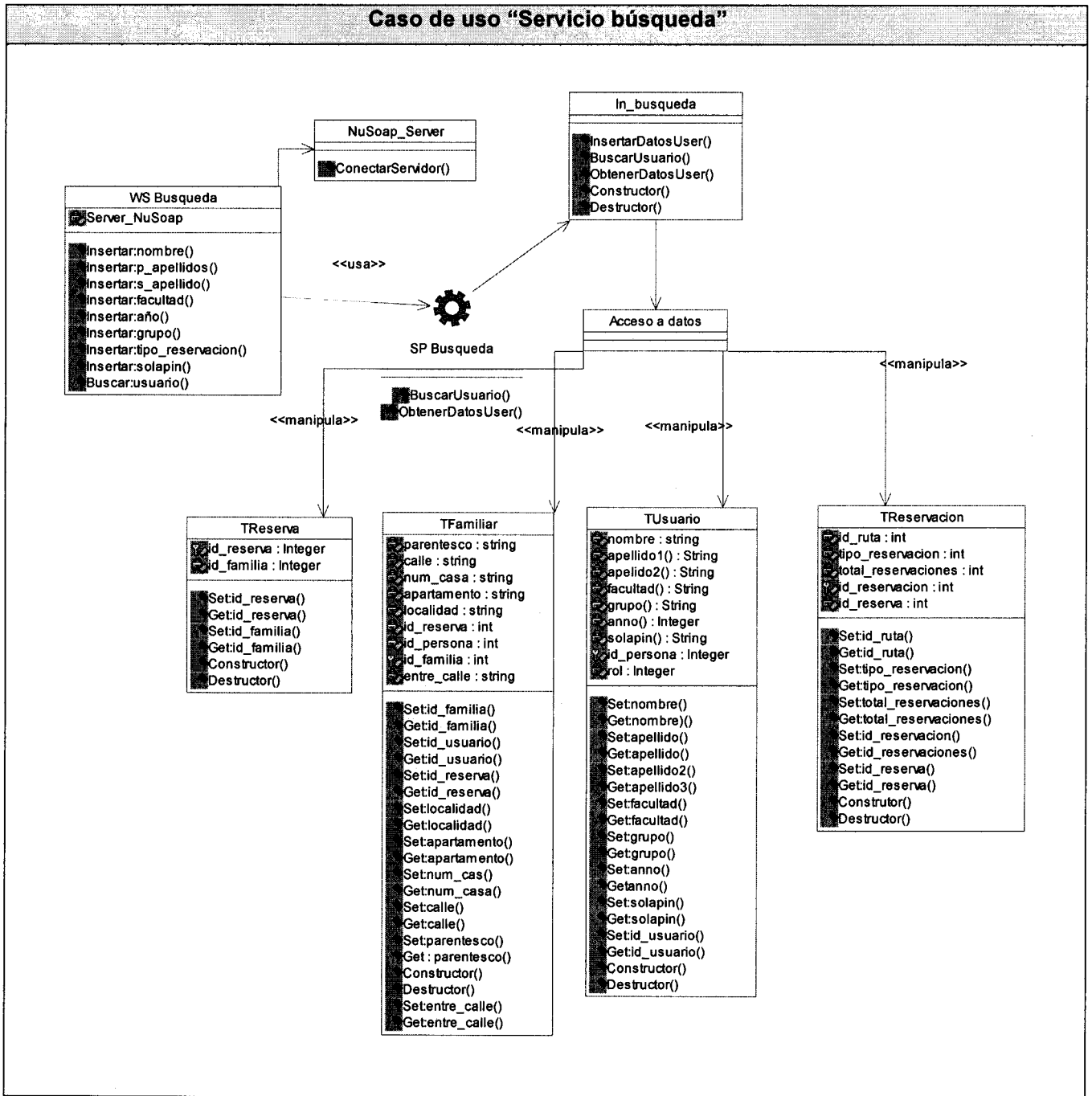


Figura 4.13 Diagrama de clases del diseño del caso de uso "Servicio búsqueda"

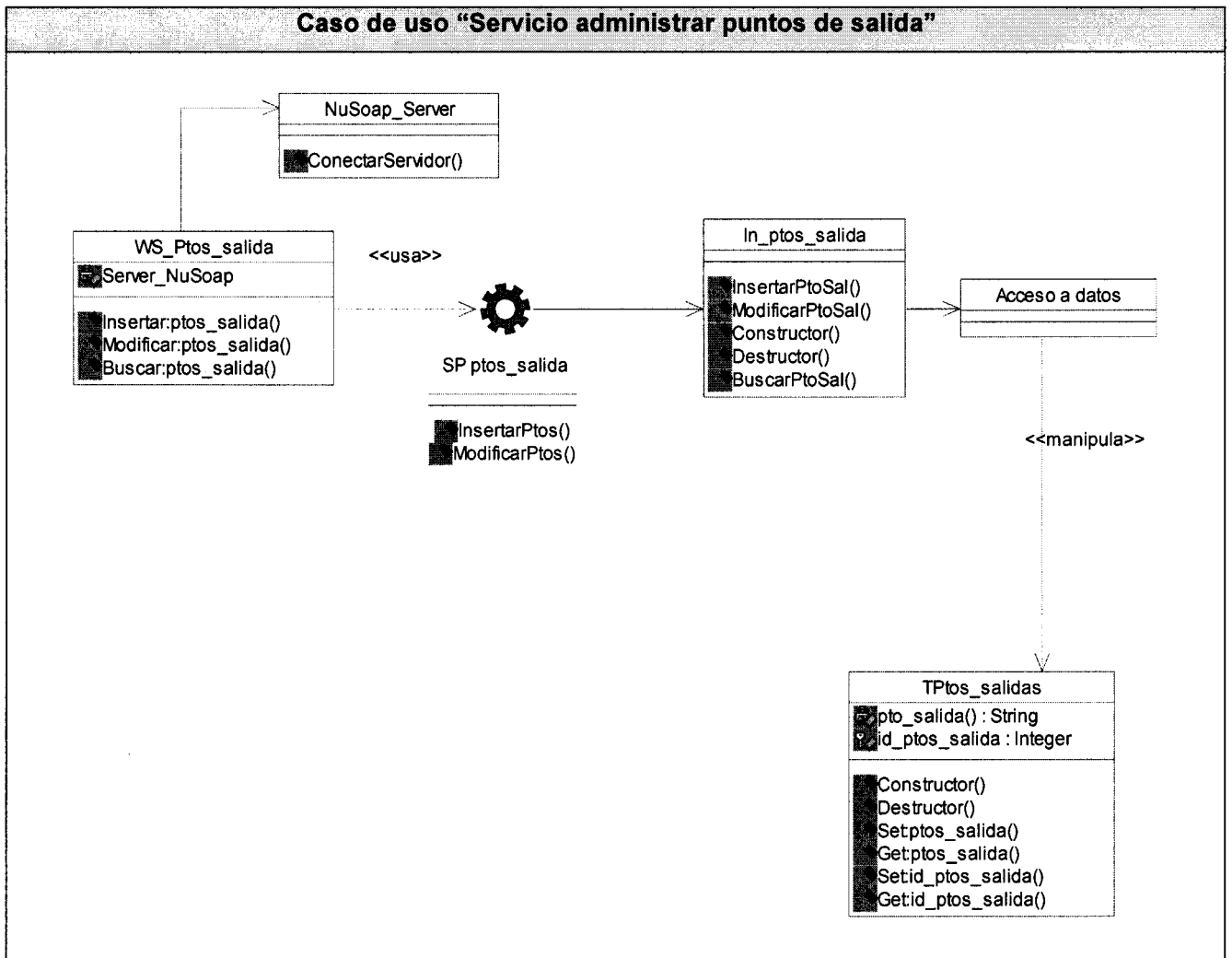


Figura 4.14 Diagrama de clases del diseño del caso de uso "Servicio administrar puntos de salida"

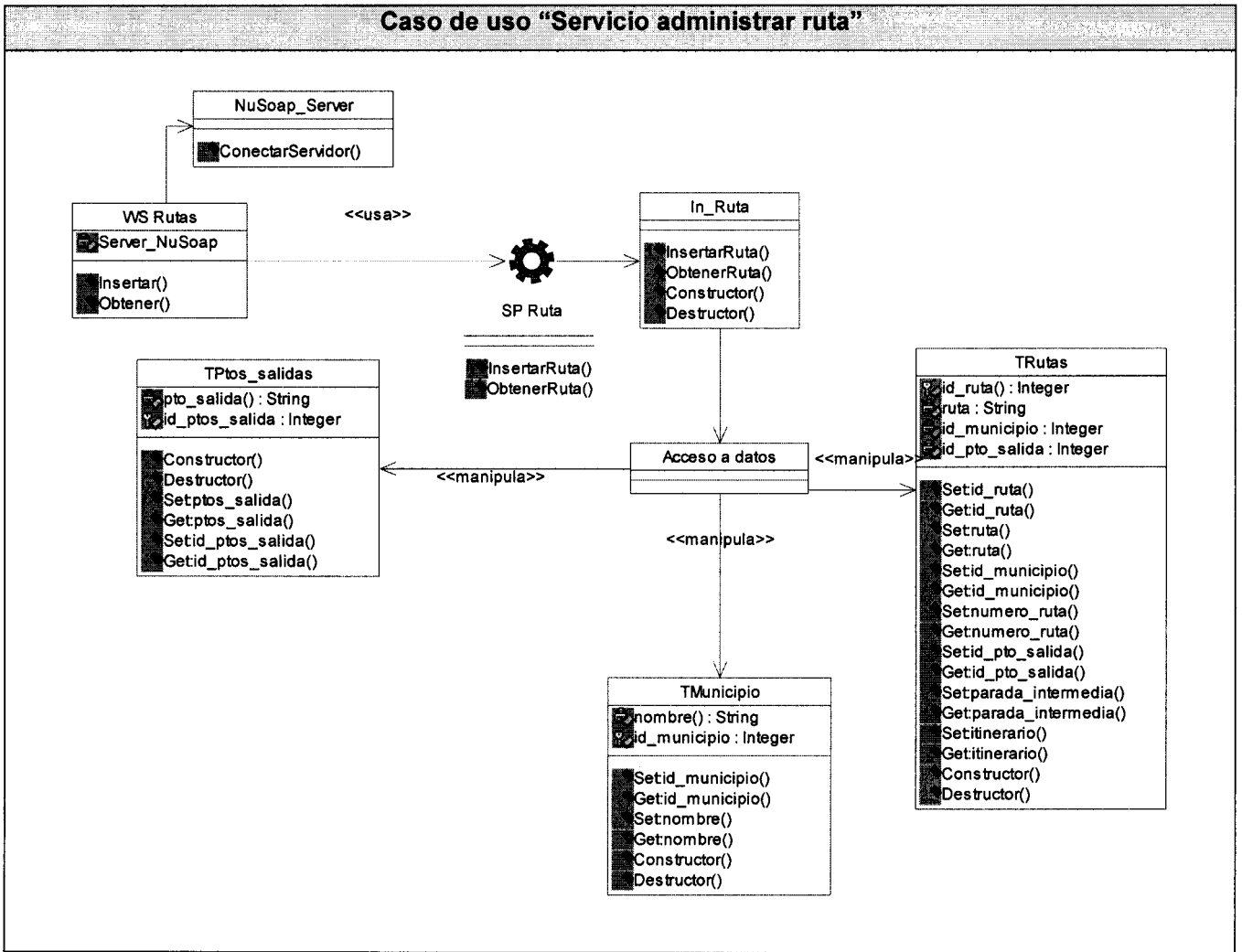


Figura 4.15 Diagrama de clases del diseño del caso de uso "Servicio administrar ruta"

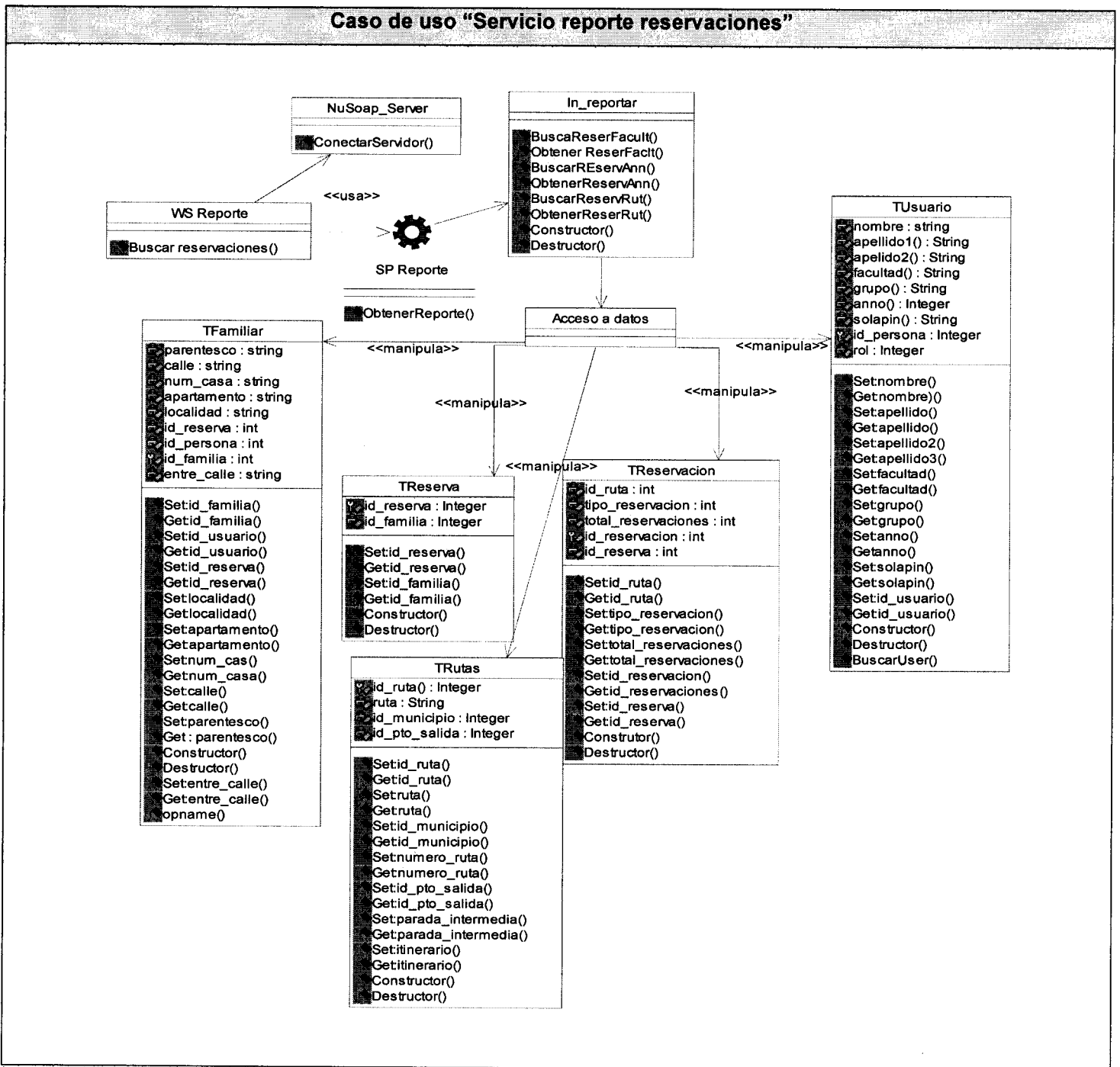


Figura 4.16 Diagrama de clases del diseño del caso de uso "Servicio reporte reservaciones"

Caso de uso "Servicio gestionar reservación"

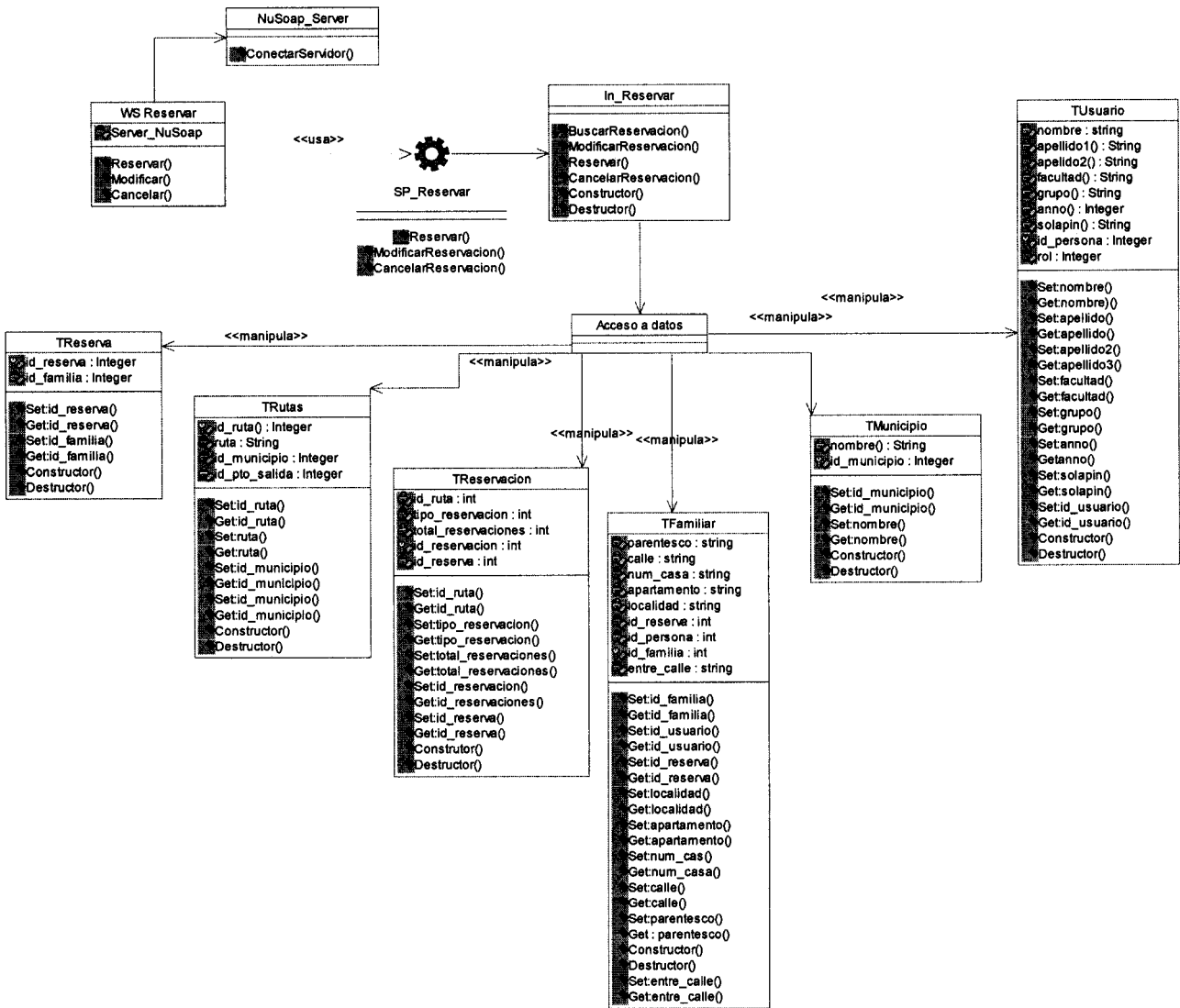


Figura 4.17 Diagrama de clases del diseño del caso de uso "Servicio gestionar reservación"



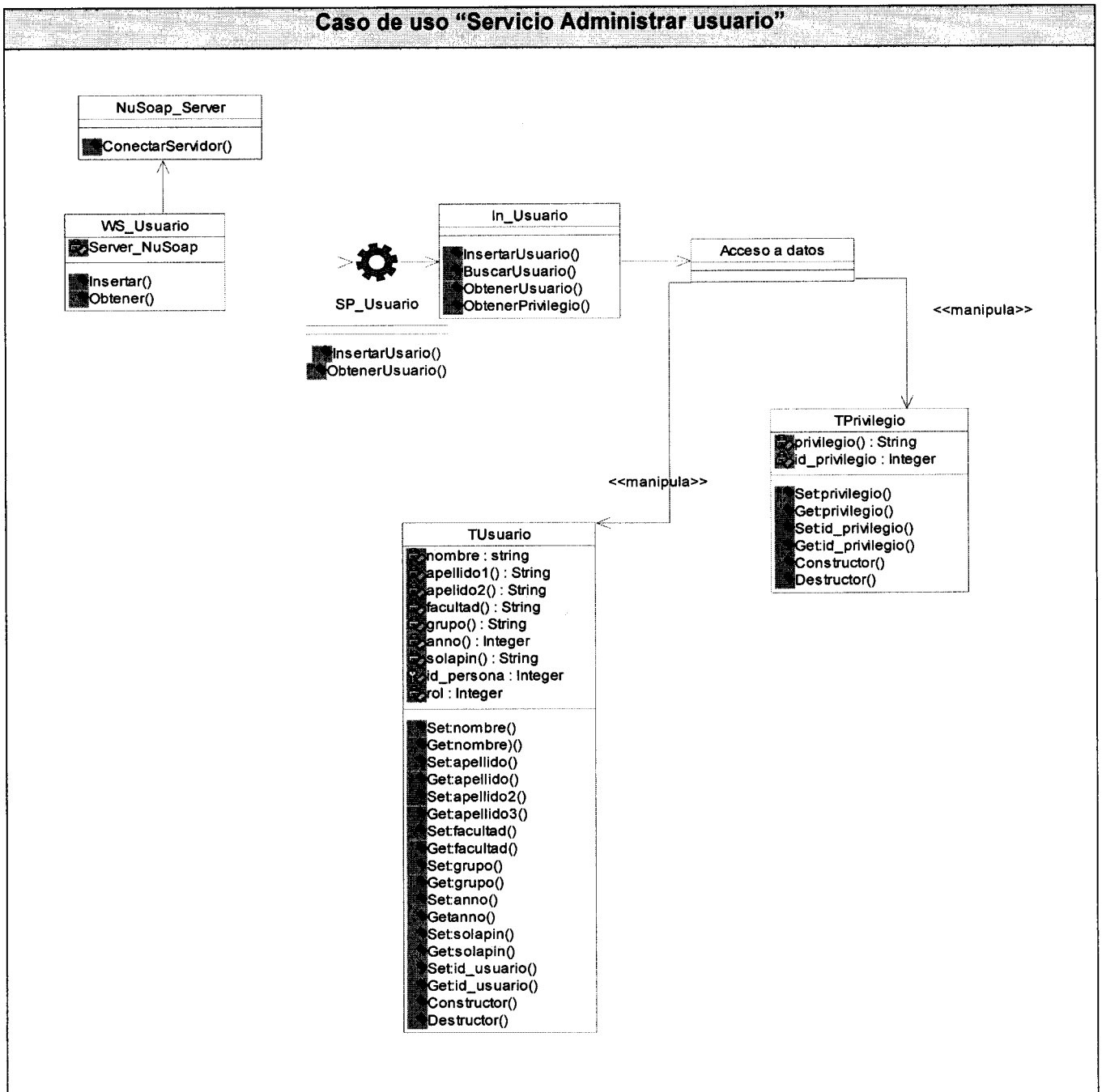


Figura 4.18 Diagrama de clases del diseño del caso de uso "Servicio administrar usuario"



### **CAPÍTULO 5: PROPUESTA DE TRANSFERENCIA TECNOLÓGICA.**

En este capítulo se describe de forma concreta, el proceso relacionado con la transferencia tecnológica del sistema de Reservación del Transporte estudiantil de la UCI, hacia la Facultad Regional de Ciego de Ávila, realizando un estudio de la repercusión de este proceso en el ámbito internacional, nacional, y el motivo de la universidad para llevar a cabo este proceso. Además se hace un análisis crítico de las características del sistema, de la estrategia y las actividades a seguir para efectuar este proceso, de los tipos de pruebas a realizar en el sistema, del mantenimiento del sistema y de los beneficios que trae consigo realizar este proceso.

#### **5.1 Transferencia tecnológica.**

Con el avance de las tecnologías de la información, y en especial Internet, la informática se ha convertido en un aspecto imprescindible, constituyendo de vital importancia su uso en nuestros días. En nuestro país, se está desarrollando un levantamiento que tiene como principal objetivo, lograr la informatización de la sociedad. A raíz de este proceso, surge la idea de transmitir la trascendencia de la UCI hacia los diferentes territorios del país, con la apertura de facultades regionales, las cuales conservan el mismo estilo de la fuente rectora.

Una de las facultades en cuestión, es la Facultad Regional de Ciego de Ávila, para la cual se propone, realizar la transferencia tecnológica del sistema de Reservación del Transporte estudiantil de la UCI, con el fin de contribuir con la informatización de la misma y viabilizar su desarrollo. A continuación se realiza un estudio del arte del tema a nivel internacional y nacional.

#### **Estudio del tema.**

En capítulos anteriores se definió la transferencia tecnológica como el traspaso de un paquete tecnológico o parte de él, desde una unidad u organización hacia otra entidad o país, con el objetivo de que produzcan y pueda ofrecer mayores servicios.

Este proceso en la actualidad tiene muchos matices, ya que los países que se encuentran en vísperas del desarrollo, no poseen la capacidad suficiente para respaldar un avance tecnológico, por lo que la

alternativa es, la transferencia de tecnología. Sin embargo, dicha adquisición se basa en las necesidades específicas de cada empresa y de la posibilidad que presenten las mismas para realizar la negociación pertinente. El cambio de la estructura tecnológica en la economía de países en vías de desarrollo y el aprovechamiento de los resultados de la revolución científico-técnica, va acompañado por nuevas exigencias para la transferencia de tecnologías desde países industrializados.

La idea de la transferencia de tecnología surge del análisis de que ningún país, ni ninguna empresa, pueden ser totalmente autosuficientes en lo que a tecnología se refiere. En Cuba, los contratos referentes a la transferencia de tecnología se asocian fundamentalmente con la inversión extranjera regulada por la Ley No. 77 y la creación de empresas mixtas o contratos económicos de carácter internacional, en los cuales interviene el capital extranjero.

Los contratos que se establecen para la transferencia de tecnología no son en partes iguales, ya que estos se adaptan a las circunstancias de cada uno de los casos para los que pueden ser utilizados. Entre ellos se pueden mencionar algunos de los más frecuentes:

- El contrato de transferencia de tecnología para la fabricación de un producto.
- El contrato de transferencia de tecnología para la implementación de procesos y el suministro de licencias.
- El contrato de transferencia de tecnología para el suministro de equipos.
- El contrato tecnológico de servicio.

En nuestro país, las tareas encaminadas al progreso científico-técnico, se desarrollan de la manera más integral posible, no sólo creando instituciones de investigación, sino desarrollando actividades como la información científica, la normalización y el control de calidad, la organización científicas del trabajo, las patentes y licencias, la proyección industrial y las tareas vinculadas con la transferencia de tecnología y su asimilación.

Este proceso también es muy tratado en las universidades, las cuales, son un ejemplo esencial en este campo. Existen universidades que han experimentado el proceso de transferencia para dar conocimiento a graduados o para ponerla en práctica con la ayuda de los mismos.

En la Universidad de las Ciencias Informáticas (UCI) se encamina este proceso de transferencia de tecnología hacia las distintas facultades de los territorios, el cual posibilitará, la mejora de las condiciones existentes en cada facultad, lo cual sería de gran utilidad para la Facultad Regional de Ciego de Ávila. La transferencia de tecnología que se está llevando a cabo desde la UCI tiene como objetivo fomentar la existencia de sistemas, incrementar el uso de las tecnologías y de los recursos informáticos en las facultades regionales.

### **5.2 Características generales del producto a transferir.**

El sistema que permite la gestión del pase de los estudiantes en la UCI, Reservación del Transporte, es un sistema que se caracteriza por posibilitar la reservación del pase del fin de semana a través de la red. Esta aplicación informática, es capaz, dada una reservación existente, de modificarla y hasta cancelarla en caso necesario, comunicándole al estudiante mediante un correo de confirmación, la acción que realizó. Esta actividad se puede efectuar desde cualquier computadora de la institución que pueda acceder a la aplicación. Permite además: la simultaneidad de las reservaciones; la obtención de reportes de las reservaciones realizadas ya sea por facultad, por ruta o por año; generar los boletines que acreditan a los estudiantes arribar el ómnibus; realizar una búsqueda avanzada de estudiantes que un administrador o decano necesite conocer su reservación; agregar las rutas, los municipios, los puntos de salida que están disponibles y conocer los puntos de salida a través del mapa del sistema. Son estos los aspectos relevantes que ofrece el mismo.

En la implementación de la aplicación se utilizó el lenguaje de programación del lado del servidor PHP 4, uno de los gestores de base de datos más utilizados en la actualidad como Postgre SQL, Ajax como la técnica que permite la comunicación asincrónica del usuario con el sistema y el servidor Web Apache. La interfaz de la aplicación es muy amigable y sencilla, por lo que es agradable y de fácil uso. Con respecto a la seguridad es válido decir, que es uno de los aspectos que más se tuvo en cuenta, ya que todos los usuarios del sistema acceden a las páginas que se les habilitan teniendo en cuenta el privilegio que se les asigne.

### **5.3 Documentación necesaria para la transferencia.**

Esta aplicación contiene manuales que posibilita a los distintos usuarios del sistema, conocer el correcto funcionamiento de la aplicación, con el objetivo de agilizar el trabajo con el mismo. En estos manuales se explica detalladamente como un usuario puede hacer uso del sistema según el rol que desempeñe. También se han elaborado manuales de instalación y configuración, donde explican como hacer las instalaciones de las tecnologías utilizadas por la aplicación. Para lograr un correcto funcionamiento del Sistema de Reservación del Transporte que será implantado en la Facultad Regional de Ciego de Ávila, se propone que al personal que trabajará con la parte interna del sistema, es decir administradores, se les oferten cursos básicos de capacitación de las tecnologías que soportan el sistema: PHP, PostgreSql y Servidor Web Apache.

#### **Manual de usuario.**

El manual del usuario es un documento cuyo objetivo principal es dar una guía al usuario de cómo utilizar las opciones que presenta la aplicación, que ayudaría a la orientación del personal en caso de que se presente alguna dificultad cuando se esté trabajando en la misma.

Este manual de usuario que se propone contiene la información necesaria para que los usuarios utilicen correctamente la aplicación y esclarezcan sus dudas. Aparece explicado el funcionamiento de la aplicación de forma clara y sin muchos tecnicismos para evitar que todo aquel que no sea informático pueda utilizarlo de igual manera.

#### **Manual de instalación y configuración.**

Es el documento que explica los pasos a realizar en cada momento para realizar una instalación adecuada del sistema en cuestión y también explica como configurarlo de la forma más óptima posible.

Es por ello que se hace la propuesta de realizar la transferencia del sistema de Reservación del Transporte hacia la Facultad de Ciego de Ávila, por la facilidad para la gestión del pase en la facultad y la gran funcionalidad que presenta en su funcionamiento.

### **5.4 Estrategia de transferencia tecnológica.**

Para llevar a cabo la transferencia de tecnología, se necesita crear un plan que oriente la estrategia a seguir para que se efectúe de forma factible.

Cuando se va realizar una transferencia de tecnología, existe un oferente y un receptor. El oferente puede ser una fuente de innovación tecnológica, un producto o un proceso, entiéndase con lo expuesto anteriormente una empresa privada, una universidad, un organismo de investigación o un centro tecnológico. Sin embargo el receptor, puede ser una persona o una empresa que es la que adquiere la innovación tecnológica, el producto, el proceso o la experiencia.

Este proceso está compuesto por cuatro partes:

- Definición del problema.
- Oferta tecnológica inicial.
- Determinación de la oferta tecnológica intermedia.
- Evaluación permanente y retroalimentación.

### **5.5 Actividades a desarrollar para el proceso de transferencia.**

A continuación se hace referencia a las principales actividades a tener en cuenta para la transferencia.

- Autorización del personal que se encarga del mantenimiento y soporte del sistema de Reservación del Transporte estudiantil en la UCI.
- Estudio y búsqueda de los programas que se necesitan para que la aplicación pueda ser instalada de forma correcta como: PHP 4, PostgreSql, Servidor Web Apache.
- Elaborar el manual para la instalación y la configuración del sistema, con el objetivo de guiar al personal encargado de la instalación del software y las configuraciones necesarias para que el sistema funcione perfectamente.
- Instalación del software necesario para que pueda correr la aplicación en la computadora destinada.
- Curso de capacitación al personal encargado del sistema.
- Instalación del sistema en la computadora que se utilizará de servidor para probar su funcionamiento.

- Analizar el funcionamiento del sistema de Reservación del Transporte.

### **Procesos para realizar la transferencia.**

- Obtener la información referente a lo que se necesita para que el producto funcione perfectamente, es decir los programas de instalación como PHP 4, PostgreSql, Servidor Web Apache, Ajax y el producto en cuestión, teniendo en cuenta los manuales de instalación, de configuración y del usuario. Es recomendable trasladar esta información en soporte digital y magnético para garantizar su seguridad.
- Realizar el traslado de la aplicación con toda la información requerida hacia la Facultad Regional.
- Crear las condiciones necesarias e imprescindibles en la facultad para la ubicación del sistema, como por ejemplo las computadoras que se van a emplear para montar la aplicación y la base de datos, las cuales deben tener un gran nivel de seguridad; garantizar que la máquina cuente con un buen antivirus; prever la conectividad y garantizar que las redes se encuentren en óptimas condiciones.
- Instalar y configurar el software necesario para que la aplicación funcione adecuadamente.
- Publicar la dirección para que se pueda acceder al sistema de Reservación del Transporte.
- Hacer uso del sistema en la Facultad Regional de acuerdo a las condiciones que se requiera en la misma y, según los inconvenientes, realizar los cambios necesarios con respecto a sus funcionalidades.
- Analizar la instalación anterior y observar detenidamente los resultados obtenidos del sistema en funcionamiento.
- Realizar la capacitación del personal mediante un curso para todas aquellas personas que interactúen con dicho sistema para que conozcan su funcionamiento y las respuestas necesarias ante determinadas situaciones que se presenten.
- Mostrar y explicar al personal que trabajará con el sistema los distintos manuales para que se guíen. Estos manuales serían los de instalación, configuración y el de usuario para futuras consultas.
- Culminar con la entrega de los manuales y la documentación técnica, la cual contiene toda la ingeniería del sistema.



Se recomienda la estrategia antes analizada para realizar la transferencia de esta tecnológica ya que no requiere de gran cantidad de requisitos, ni de una documentación muy tediosa.

### 5.6 Control de calidad.

Después de haber instalado el sistema en la Facultad Regional de Ciego de Ávila se realizarán las pruebas correspondientes para comprobar su adecuado funcionamiento.

Es recomendable hacer una prueba después de que esté instalado el sistema, con el objetivo de conocer la aceptación del nuevo producto. Estas pruebas son básicamente funcionales, ya que se realizan sobre el sistema completo. Se emplean dos técnicas para las pruebas de aceptación:

- **La prueba alfa:** Se realiza por el cliente, en el lugar de desarrollo. Se utiliza el sistema de forma natural con el desarrollador o programador como espectador del usuario. Para que tenga validez, se debe crear un ambiente con las mismas condiciones con las cuales interactuará en la nueva instalación. Después de haber logrado el objetivo, se documentan los resultados la misma.
- **La prueba beta:** Es realizada por lo usuarios finales en sus lugares o puestos de trabajo. Se diferencia de la prueba alfa por que el desarrollador no se encuentra presente. De esta manera la prueba se hace con la aplicación en directo en un marco no controlado por el desarrollador o programador. El nuevo cliente anota todos los problemas ya sean reales o imaginarios que se presenten durante el desarrollo de la prueba, para luego informarlo al desarrollador o programador.

Con los resultados encontrados durante la etapa de prueba, el desarrollador ya está en condiciones de llevar a cabo, las modificaciones que sean necesarias y de esta manera, obtener una versión mejorada.

Los objetivos principales de efectuar las pruebas son:

- Detectar errores.
- Obtener un buen caso de prueba.
- Arreglar un error que fue descubierto en la etapa de pruebas.

## 5.7 Mantenimiento del sistema.

Las computadoras a utilizar para la instalación del sistema y de los programas necesarios para el funcionamiento del mismo, necesitan de un antivirus actualizado y parches de seguridad con el objetivo de evitar consecuencias que podrían afectar en gran medida, tanto al sistema como a la computadora.

Dar mantenimiento al sistema constituye una de las fases en el Ciclo de Vida de Desarrollo de Sistemas, ya que es una de las tareas más comunes que se realizan diariamente por los ingenieros. Este proceso tiene como objetivo perfeccionar el funcionamiento del sistema y mejorar su rendimiento. Esta fase de mantenimiento involucra cambios en el sistema, cambios que son para corregir defectos o encontrar problemas en el mismo.

### Tipos de mantenimiento.

Existen varios tipos de mantenimiento para un sistema. Dentro de ellos se encuentran los siguientes:

**Perfectivo:** Es el conjunto de actividades para mejorar o añadir nuevas funcionalidades requeridas por el usuario.

**Adaptativo:** Es el conjunto de actividades para adaptar el sistema a los cambios, ya sea de hardware o de software.

**Correctivo:** Es el conjunto de actividades dedicadas a corregir defectos en el hardware o en el software, defectos que son detectados por los usuarios durante la explotación del sistema.

**Preventivo:** Es el conjunto de actividades para prevenir el funcionamiento del sistema y facilitar el mantenimiento futuro.

Para efectuar el mantenimiento del sistema es necesario:

- Revisar la integridad del sistema operativo.
- Verificar la inexistencia de virus u otras amenazas.
- Optimizar el arranque de Windows.
- Hacer actualizaciones en el mismo.

### **5.8 Beneficios de la transferencia de tecnología.**

Todo país, institución u organización puede obtener nuevas ventajas y beneficios con transferencia de tecnología, ya que esta puede aumentar la productividad y el desarrollo de la entidad o país. El objetivo primordial de la Transferencia de Tecnológica es la obtención de nuevas funcionalidades que facilite, el uso de la tecnología que se utiliza en otro lugar con más desarrollo que el analizado.

### **5.9 Conclusiones**

En este capítulo se explica de forma concreta todo el proceso para la realización de la propuesta de transferencia del sistema de Reservación del Transporte hacia la Facultad Regional de Ciego de Ávila. Todo este proceso comprende las características del sistema, la documentación que se necesita para el entendimiento del mismo, la estrategia a seguir como guía para la transferencia, las actividades a desarrollar, las pruebas de control y el mantenimiento del sistema.

### CONCLUSIONES

- ✓ Se realizó un estudio crítico mediante la ingeniería inversa para entender el funcionamiento del sistema con el objetivo de realizar el diseño de los servicios Web.
- ✓ Se desarrolló el diseño de los servicios Web para que la aplicación pueda utilizar esta tecnología una vez que sean implementados. Se utilizó como proceso de desarrollo el Proceso Racional Unificado, combinado con el lenguaje de modelado UML y auxiliado por la herramienta de modelado Rational Rose.
- ✓ Se propuso la transferencia tecnológica del sistema de Reservación del Transporte estudiantil de la UCI para la Facultad Regional de Ciego de Ávila, por las funcionalidades que presenta y para facilitar el proceso de gestión del pase en esa facultad.

## RECOMENDACIONES

Se recomienda:

- ✓ La implementación de los servicios Web diseñados para el sistema, por lo que se propone el uso del lenguaje PHP, basado en el análisis realizado en el trabajo.
- ✓ Desarrollar una nueva versión del sistema Reservación del Transporte estudiantil de la UCI que contenga los servicios Web.
- ✓ Llevar a la práctica la propuesta que se plantea de transferencia tecnológica hacia la Facultad Regional de Ciego de Ávila.
- ✓ Extender el uso de los servicios Web para otras aplicaciones informáticas de la UCI.

## REFERENCIAS BIBLIOGRÁFICAS

[1] Espinosa Luisa Aparicio; Rodríguez, Róger Martín. Glosario de términos bibliotecológicos y de Ciencias de la Información.

Disponible en:

[http://www.uh.cu/facultades/fcom/portal/interes\\_glosa\\_terminos.htm](http://www.uh.cu/facultades/fcom/portal/interes_glosa_terminos.htm)

[2] Bartle, Phil. Información para la gestión y gestión de la información, 2007.

Disponible en:

<http://www.scn.org/mpfc/modules/mon-miss.htm>

[3] González, Benjamín C. XML Web Service.

Disponible en

<http://www.desarrolloweb.com/articulos/1545.php>

[4] W3C: World Wide Web Consortium. Guía breve de Servicios Web, 2006.

Disponible en:

<http://www.w3c.es/Divulgacion/Guiasbreves/ServiciosWeb>

[5] Fagas, Francisco Xavier. Aplicaciones para aplicaciones corporativas, 2007.

Disponible en:

[http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ\\_2086.asp#M1](http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_2086.asp#M1)

[6] Centro de Automatización de Procesos, 2006.

Disponible en:

<http://www.puj.edu.co/centro/cap/5891.htm>

[7] Tápanes, Daysaríh; Rodríguez, Armando. La transferencia de tecnología asociada al proceso inversionista en Cuba en el cuatrienio 2002-2005, 2006.

Disponible en:

<http://www.congreso-info.cu/UserFiles/File/Info/Info2006/Ponencias/218.pdf>

[8] Mateu, Carles. Desarrollo de Aplicaciones Web, 2004

Disponible en:

<http://www.uoc.edu/masters/esp/img/873.pdf>

[9] Loyola, Karol. Ingeniería del software RUP y UML, 2006.

Disponible en:

<http://www.mmug.cl/articulos.php?id=287&tod=1>

[10] García, Arenas María Isabel Curso XML

Disponible en:

<http://geneura.ugr.es/~maribel/xml/introduccion/index.shtml#12>

[11] Wolter, Roger. Fundamentos de los servicios Web XML, 2003.

Disponible en:

[http://www.microsoft.com/spanish/msdn/articulos/archivo/151102/voices/fundamentos\\_xml.asp#11](http://www.microsoft.com/spanish/msdn/articulos/archivo/151102/voices/fundamentos_xml.asp#11)

[12] Servicios Web con PHP, 2004.

Disponible en:

<http://www.malditainternet.com/node/1169?PHPSESSID=b911fbc452ee1fb230f97274208bb425>

[13] Web dinámicas con PHP

Disponible en:

<http://ascii.eii.us.es/docs/2002-03/php/php4.html>

[14] Cuenca, Carlos Luis. Arquitectura del servidor Apache,

Disponible en:

<http://www.desarrolloweb.com/articulos/1112.php>

[15] Garbage Collector, 2004

Disponible en:

[http://www.error500.net/garbagecollector/bases\\_de\\_datos/sistema\\_gestor\\_de\\_base\\_de\\_dato.html](http://www.error500.net/garbagecollector/bases_de_datos/sistema_gestor_de_base_de_dato.html)

[16] González, Carlos. Base de Datos PostgreSQL, SQL avanzado y PHP, 2007

Disponible en:

<http://www.usabilidadweb.com.ar/postgre.php>

[17] Garrett, Jesse James. Ajax un acercamiento a las aplicaciones Web, 2005.

Disponible en:

<http://www.uberbin.net/archivos/internet/ajax-un-nuevo-acercamiento-a-aplicaciones-web.php>

[18] Herramientas de análisis y requerimiento.

Disponible en:

<http://www.isca.com.ve/productos.htm>



## BIBLIOGRAFÍA.

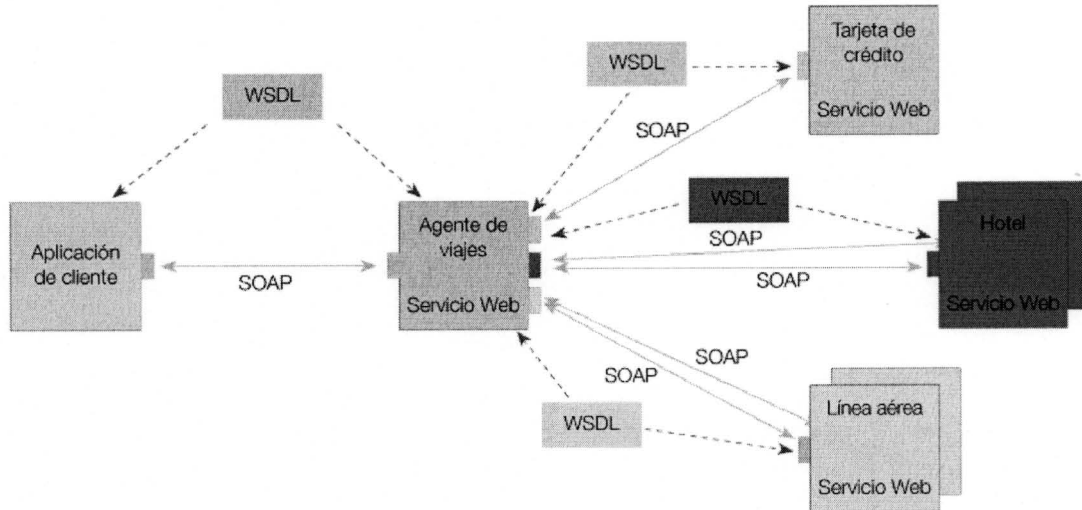
- America West Airlines, 2005  
Disponible en:  
<http://mx.attachmate.com/es-MX/CustomerStories/America+West.htm>
  
- Wolter, Roger. Fundamentos de los servicios Web XML, 2003.  
Disponible en:  
[http://www.microsoft.com/spanish/msdn/articulos/archivo/151102/voices/fundamentos\\_xml.asp#1](http://www.microsoft.com/spanish/msdn/articulos/archivo/151102/voices/fundamentos_xml.asp#1)
  
- Disponible en:  
<http://ascii.eii.us.es/docs/2002-03/php/php4.html>
  
- James, Jesse. Ajax: un nuevo acercamiento a las aplicaciones Web,2005  
Disponible en:  
<http://www.maestrosdelweb.com/editorial/ajax/>
  
- Disponible en:  
<http://es.wikipedia.org/wiki/AJAX>
  
- Rodríguez, Félix; Sepúlveda, Dolores. Sistemas de Información y las telecomunicaciones.  
Disponible en:  
<http://www.csi.map.es/csi/silice/Global75.html>
  
- 2007. Disponible en:  
<http://es.wikipedia.org/wiki/Cliente-servidor>
  
- The Apache Software Foundation. Servidor Apache, 2007  
Disponible en:  
<http://www.abcdatos.com/webmasters/programa/z2820.html>

- Sistema Gestor de Base de Datos, 2004  
Disponible en:  
[http://www.error500.net/garbagecollector/bases\\_de\\_datos/sistema\\_gestor\\_de\\_base\\_de\\_dato.html](http://www.error500.net/garbagecollector/bases_de_datos/sistema_gestor_de_base_de_dato.html)
  
- González, Carlos. Base de Datos PostgreSQL, SQL avanzado y PHP, 2007  
Disponible en:  
<http://www.usabilidadweb.com.ar/postgre.php>
  
- Randrade. ¿Qué es PostgreSQL? 2005.  
Disponible en:  
<http://www.postgresql.org.mx/?q=node/6>
  
- Elita, Recursos de Información en UNMSN,  
Disponible en  
[http://recursos-informacion.blogspot.com/2006/06/resumen-exposicin-6-transferencia-de\\_12.html](http://recursos-informacion.blogspot.com/2006/06/resumen-exposicin-6-transferencia-de_12.html)
  
- Fagas, Francisco. Arquitectura para aplicaciones corporativas.  
Disponible en:  
[http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ\\_2086.asp#M1](http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_2086.asp#M1)
  
- Servicios Web con PHP, 2007  
Disponible en:  
<http://www.malditainternet.com/node/1169?PHPSESSID=b911fbc452ee1fb230f97274208bb425>
  
- Gutiérrez, Emmanuelle. XML en 10 puntos. 2006  
Disponible en:  
<http://www.sidar.org/recur/desdi/traduc/es/xml/xml10p/xml10p.htm>
  
- Gil, Manuel Alejandro. Arquitectura Orientada a Servicios, Universidad de las Ciencias Informáticas.

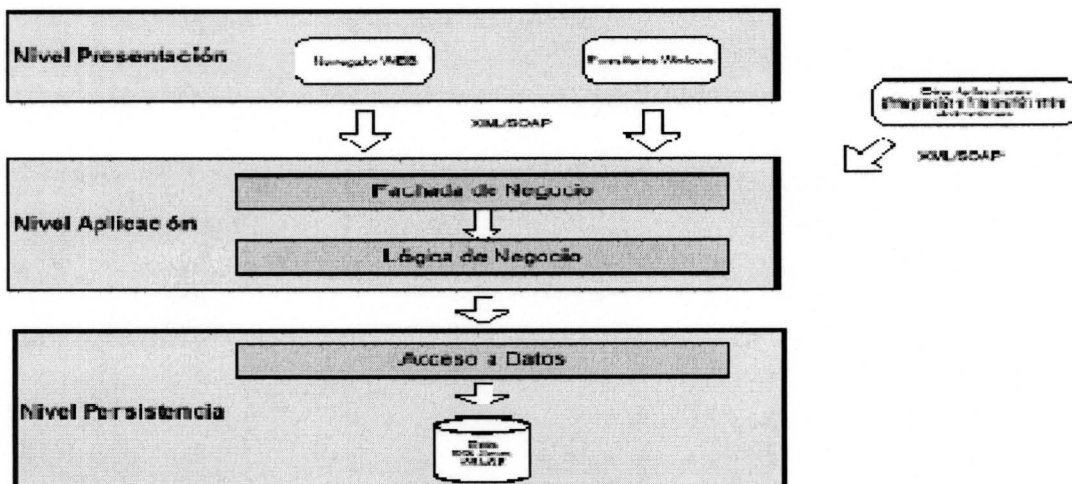
- Hernández, Enrique Orallo. El Lenguaje de Modelado (UML)
- Eito, Ricardo Brun. XML y Servicios Web, 2002
- Álvarez, Juan Carlos; Mateos, Montserrat; Moreno, María N. Metodología de Reingeniería del Software para la Remodelación de Aplicaciones Científicas Heredadas.2005
- Pelechado, Vicente; Ruiz Marta; Fons, Joan J.; Valderas, Pedro. Desarrollo de aplicaciones Web basadas en servicios Web.
- Conferencias y clase práctica Ingeniería del Software I.2005-2006.UCI  
Ingeniería del Software II.2005-2006.UCI

ANEXOS

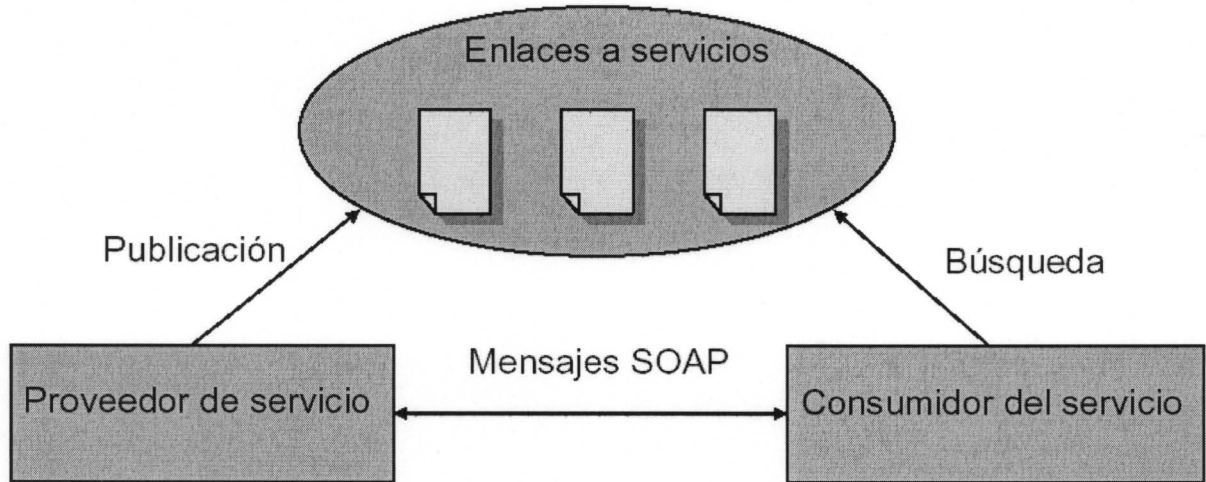
Anexo 1: Funcionamiento de un Servicio Web.



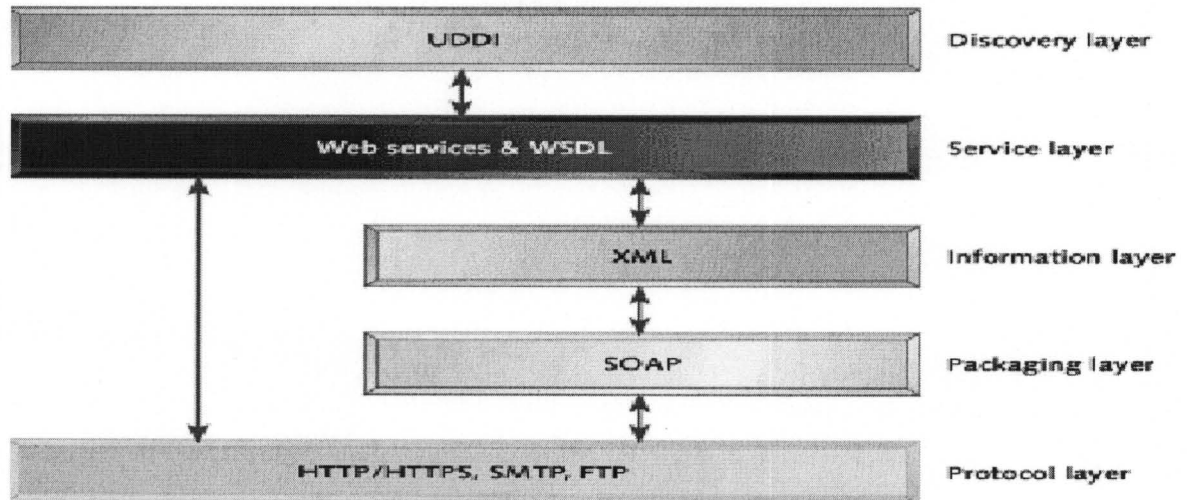
Anexo2: Arquitectura Orientada a Servicios.



**Anexo 3: Partes de la Arquitectura Orientada a Servicios: SOA.  
Registro UDDI**

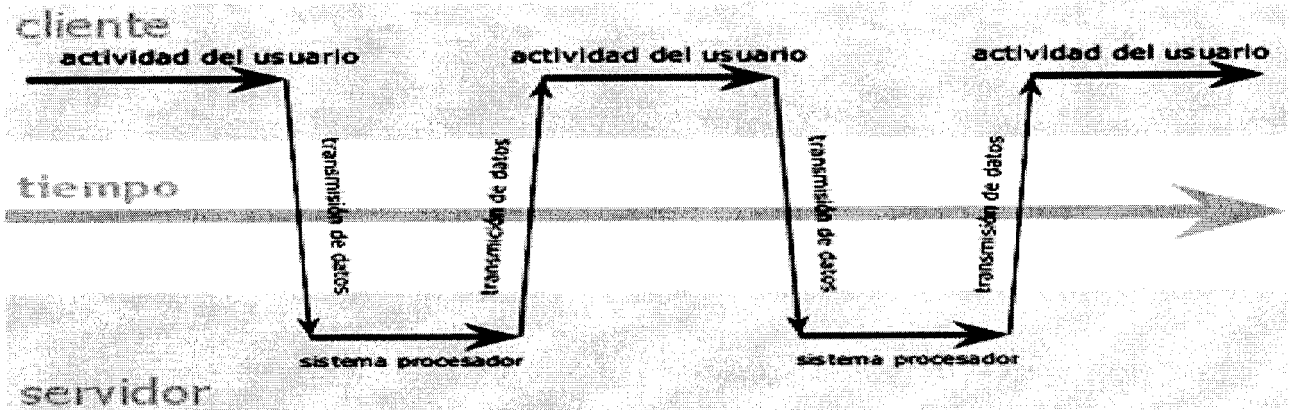


**Anexo 4: Estándares de los Servicios Web.**

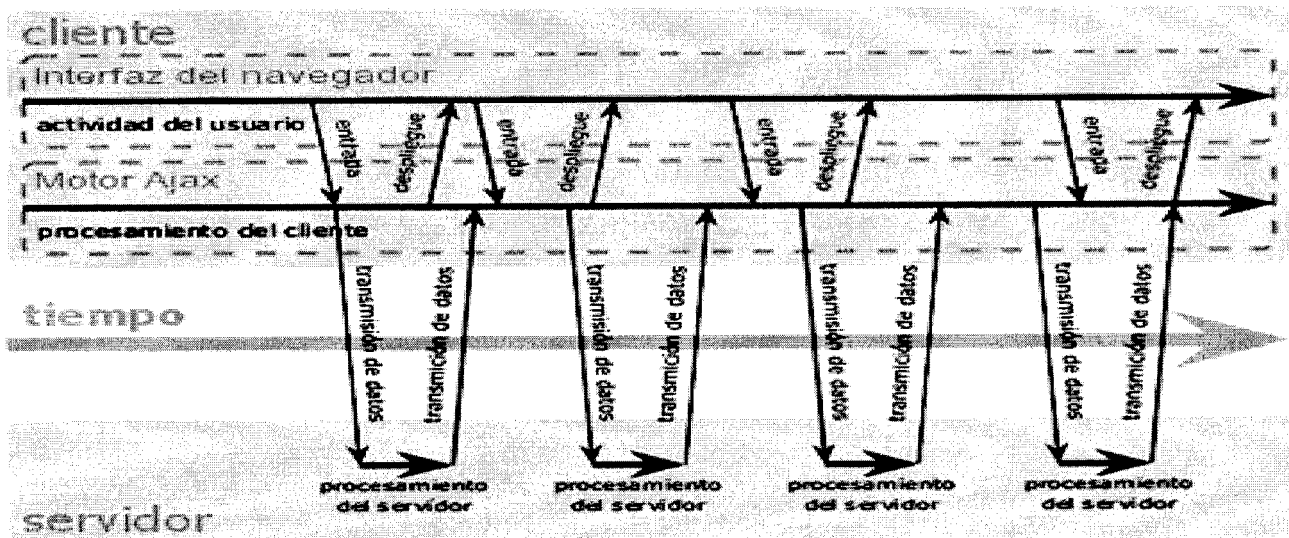


Anexo 5: Comparación entre modelo clásico y el modelo Ajax

**modelo clásico de aplicaciones web (síncrono)**

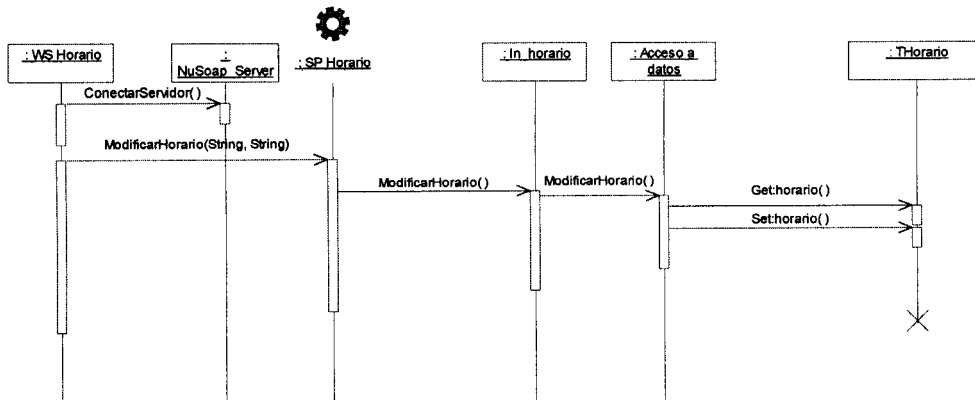


**modelo Ajax de aplicaciones web (asíncrono)**

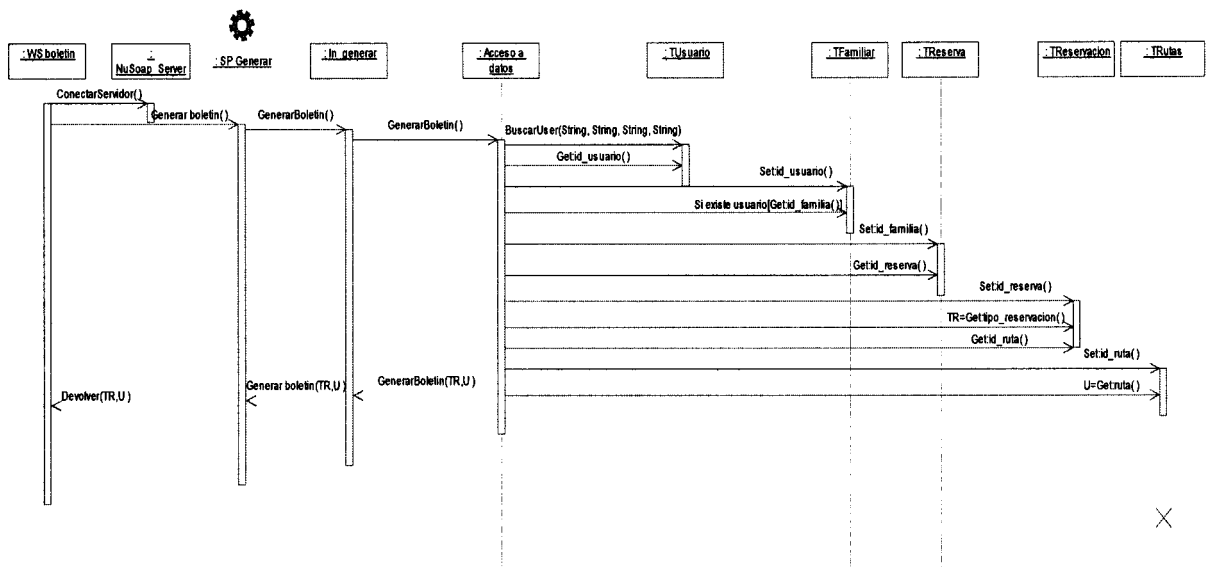


Anexo: 6 Diagrama de secuencia

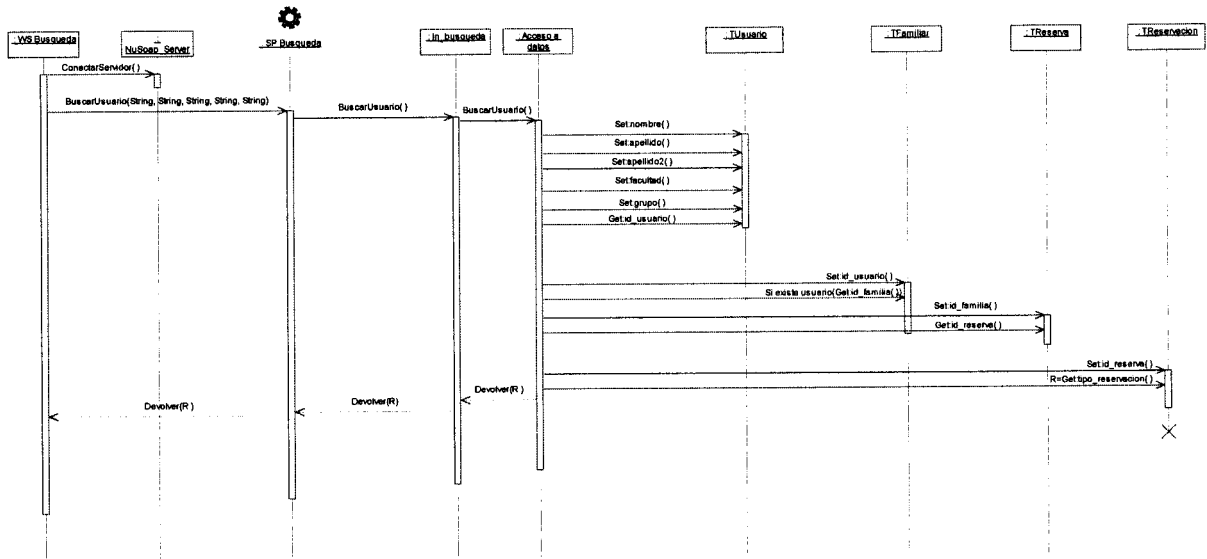
Caso de uso "Servicio administrar horario de salida y entrada"



Caso de uso "Servicio generar boletín"

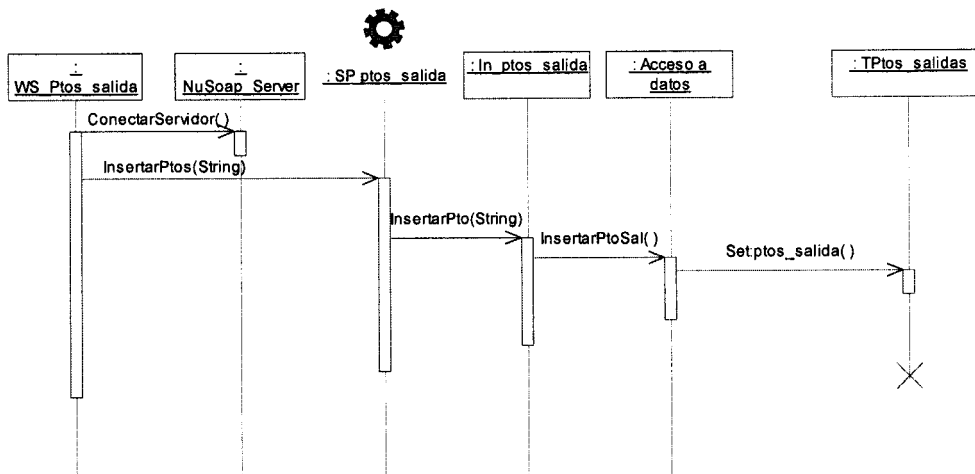


Caso de uso "Servicio búsqueda"



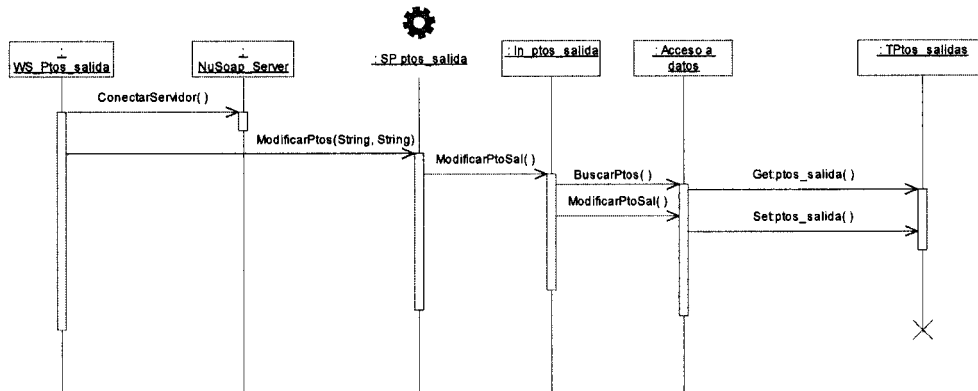
Caso de uso "Servicio administrar puntos de salida"

Escenario: Insertar punto de salida.



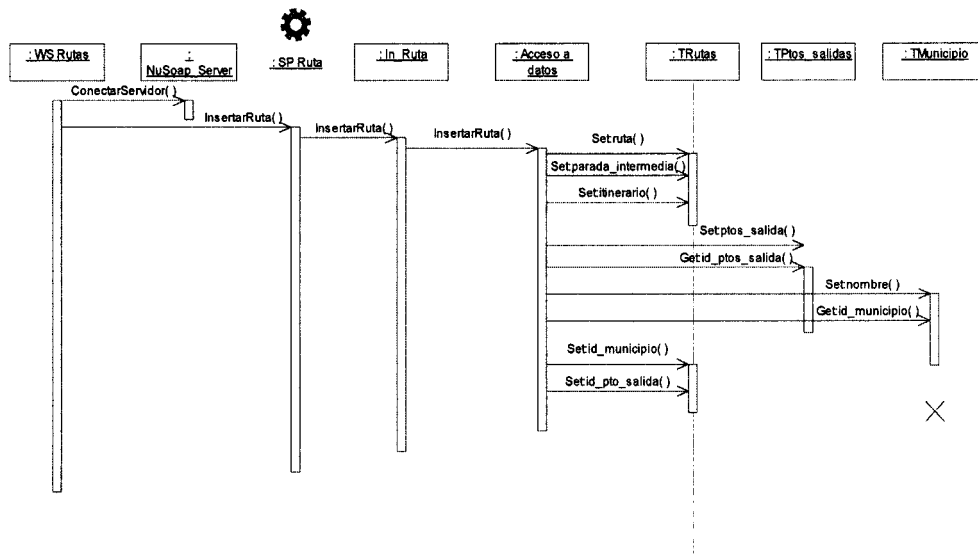


**Escenario: Modificar punto de salida.**

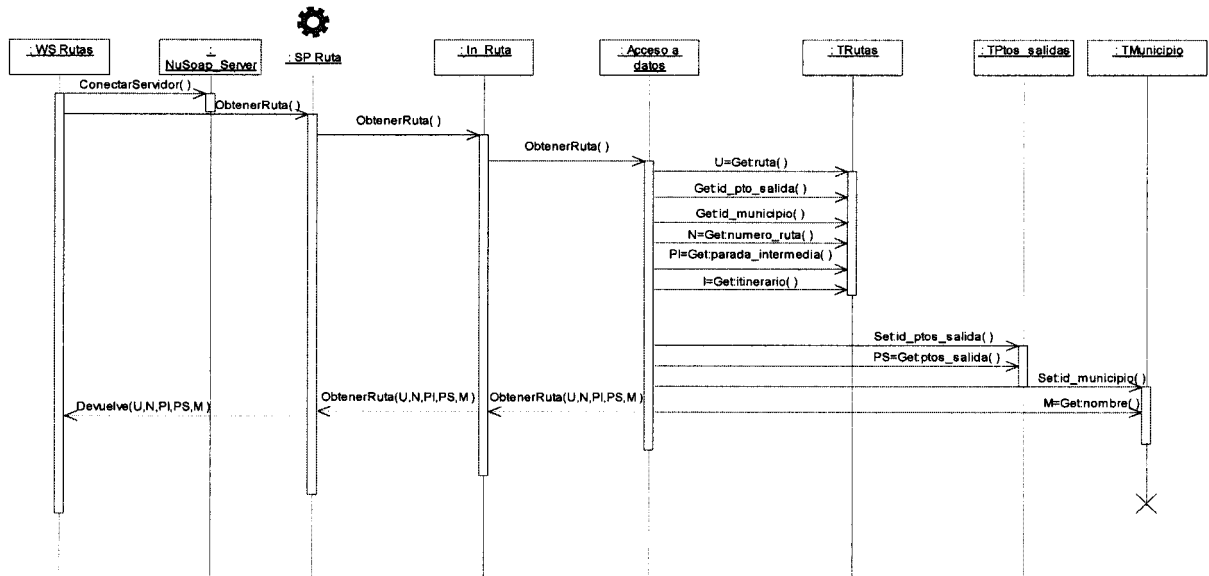


**Caso de uso "Servicio administrar ruta"**

**Escenario: Insertar ruta.**

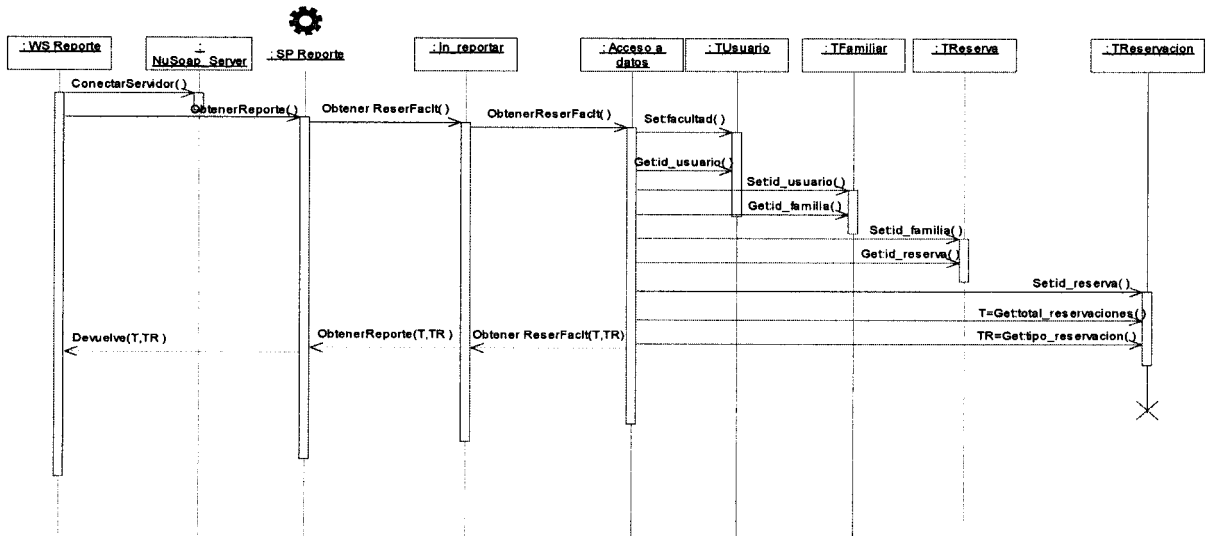


Escenario: Obtener ruta.

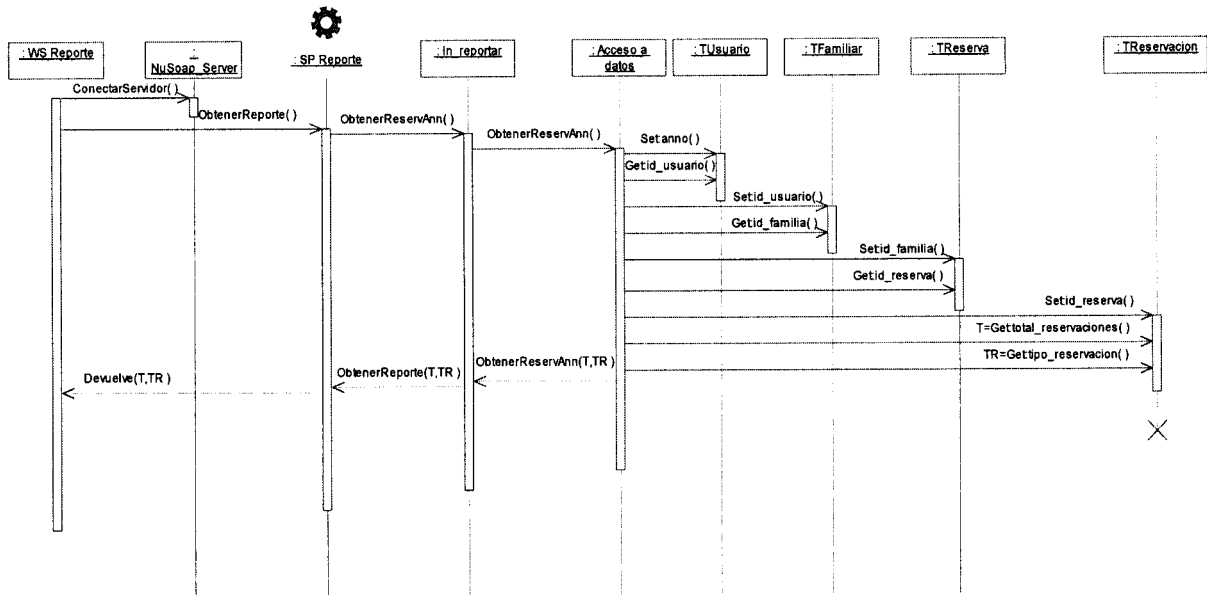


Caso de uso "Servicio reporte de reservaciones"

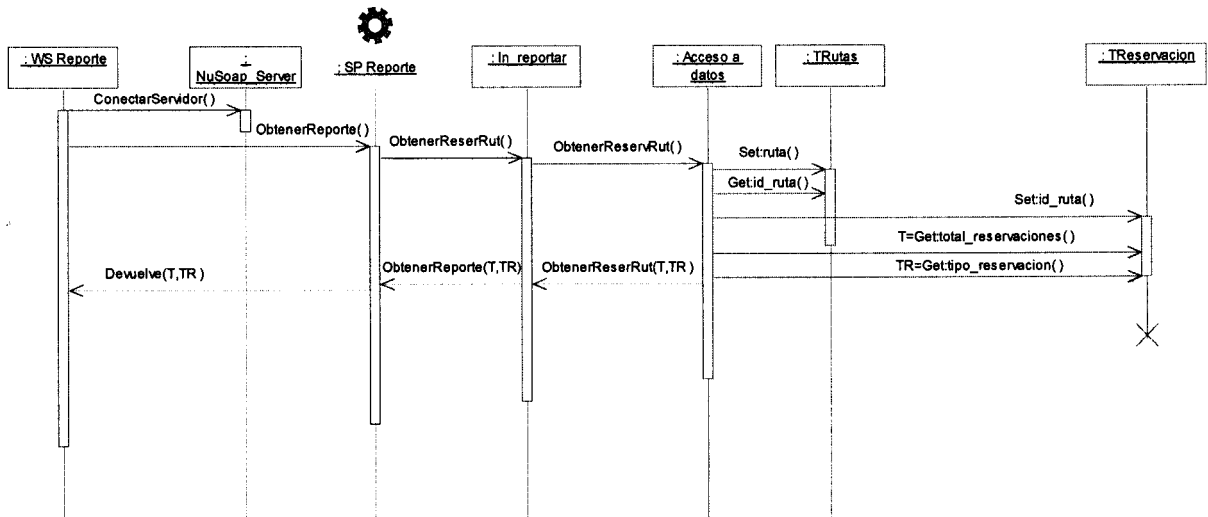
Escenario: Reporte por facultad.



Escenario: Reporte por año.

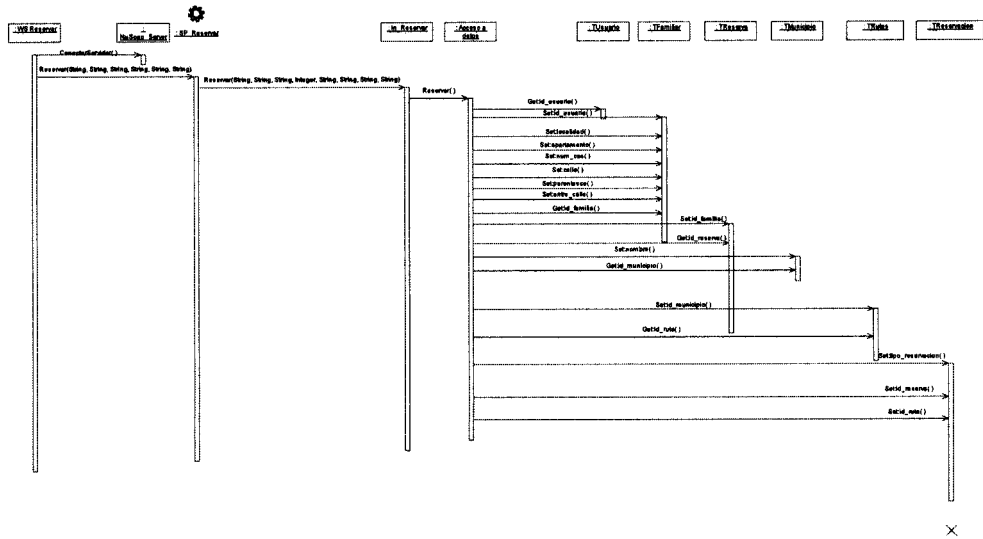


Escenario: Reporte por ruta.

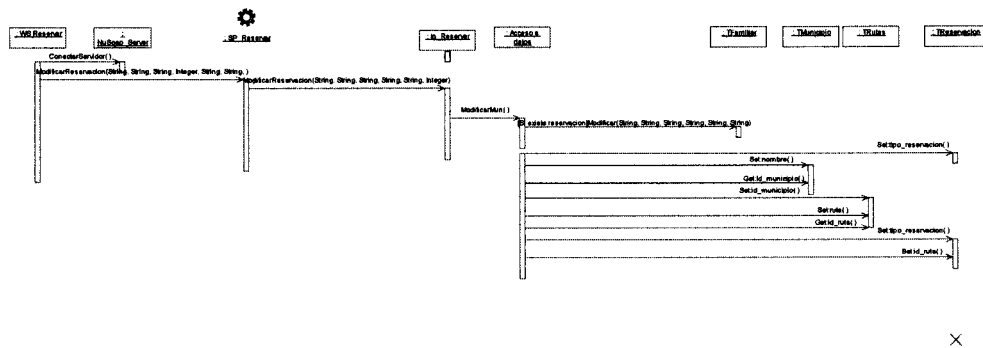


### Caso de uso "Servicio gestionar reservación"

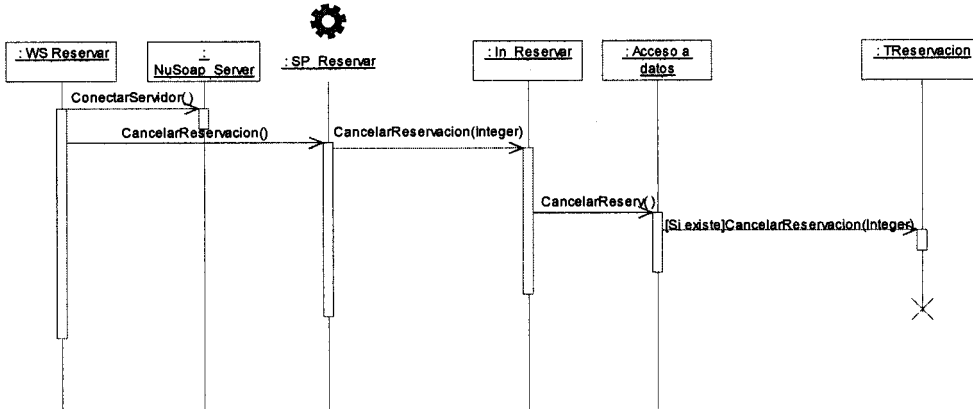
#### Escenario: Reservar.



#### Escenario: Modificar reservación.

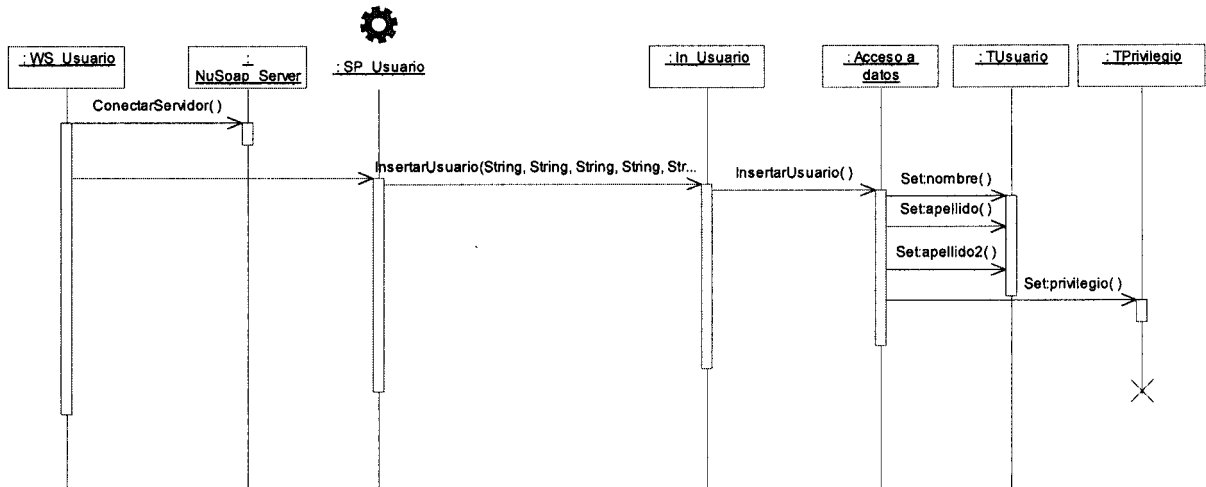


**Escenario: Cancelar reservación.**

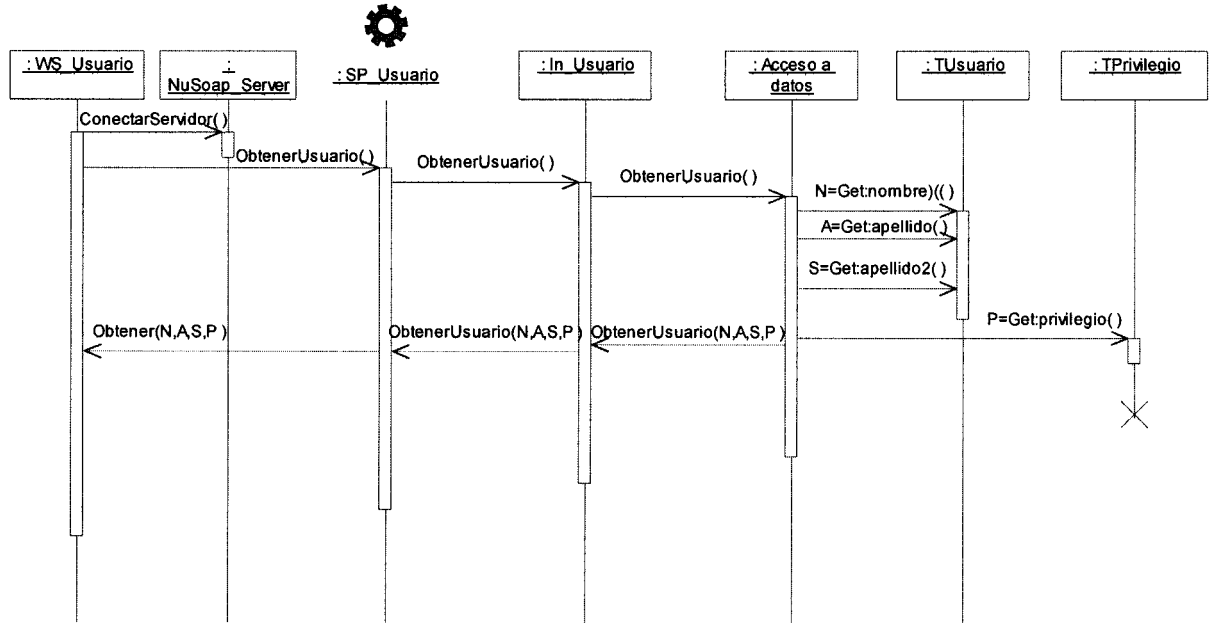


**Caso de uso "Servicio administrar usuario"**

**Escenario: Insertar usuario.**



Escenario: Obtener usuario.



1 **GLOSARIO**

- 2 **Akademios:** Aplicación Web desarrollada en la UCI para archivar, reportar y gestionar el desempeño  
3 académico de los estudiantes.
- 4 **ALTEC:** Asociación latino-iberoamericana de gestión tecnológica.
- 5 **BSD:** Licencia de software otorgada a los sistemas Berkeley Software Distribution. Pertenece al grupo de  
6 licencias de software libre.
- 7 **C/C++:** Lenguaje de programación, diseñado como extensión del lenguaje de programación C.
- 8 **CSS:** *Cascading Style Sheets*. Lenguaje formal usado para definir la presentación de un documento  
9 estructurado escrito en HTML o XML (y por extensión en XHTML).
- 10 **DB2:** Marca comercial, propiedad de IBM, bajo la cual se comercializa el sistema de gestión de base  
11 datos.
- 12 **DBMS:** Sistemas de gestión de base de datos.
- 13 **DTD:** Document Type Definition. Es una descripción de estructura y sintaxis de un documento XML.
- 14 **GPL:** *General Public License* o licencia pública general. Es una licencia creada por la Free Software  
15 Foundation, y está orientada principalmente a proteger la libre distribución, modificación y uso de  
16 software.
- 17 **HTML:** HyperText Markup Language. Es un lenguaje diseñado para estructurar textos y presentarlos en  
18 forma de hipertexto.
- 19 **HTTP:** HyperText Transfer Protocol. Protocolo usado en cada transacción de la Web.
- 20 **HTTPS:** Versión segura del protocolo HTTP.
- 21 **Hytime:** Lenguaje de estructuración hipermedia basado en el tiempo, ISO 10744

- 1 **Inter-nos:** Aplicación Web desarrollada en la UCI para brindar servicios de multimedia.
- 2 **Java:** Lenguaje de programación de alto nivel, orientado a objetos.
- 3 **JavaScript:** Lenguaje que no requiere compilación, utilizado principalmente en páginas Web.
- 4 **Ley No77:** Ley de la Inversión Extranjera en Cuba.
- 5 **Módem:** Equipo electrónico que sirve para transmitir y recibir información digital a distancia.
- 6 **Oracle:** Sistema de gestión de base datos relacional o RDBMS por el acrónimo en inglés de Relational  
7 Data Base Management System).
- 8 **Perl:** Lenguaje Práctico para la extracción. Es un lenguaje que toma características del C.
- 9 **PL/SQL:** Lenguaje de programación embebido en Oracle y PostgreSQL.
- 10 **PL/pgSQL:** Lenguaje imperativo provisto por el gestor de base de datos PostgreSQL.
- 11 **SQL:** Lenguaje de Consulta Estructurado. Un lenguaje declarativo de acceso a bases de datos  
12 relacionales.
- 13 **TCP/IP:** Conjunto de protocolos. Los protocolos más importantes que la componen: Protocolo de Control  
14 de Transmisión (TCP) y Protocolo de Internet (IP).
- 15 **TEI:** Text Encoding Initiative. Iniciativa de codificación de texto.
- 16 **US-ASCII:** Códigos de control.
- 17 **UTF-8:** Norma de transmisión de longitud variable para caracteres codificados utilizando Unicode.
- 18 **UTF-16:** Código de caracteres que proporciona una forma de representar caracteres Unicode.
- 19 **Visual Basic:** Lenguaje de programación. Es un dialecto de BASIC, con importantes añadidos con la  
20 intención de simplificar la programación.



- 1 **Web:** Sistema de documentos de hipertexto enlazados y accesibles a través de Internet.
- 2 **Windows:** Sistema operativo con interfaz gráfica para computadoras personales cuyo propietario es la  
3 empresa Microsoft.
- 4 **W3C:** World Wide Web Consortium. Es un consorcio internacional que produce estándares para la Word  
5 Wide Web.
- 6 **XHTML:** Lenguaje extensible de marcado de hipertexto. Es el lenguaje de marcado pensado para sustituir  
7 a HTML como estándar para las páginas Web.
- 8 **XMLHttpRequest:** Interfaz empleada para realizar peticiones HTTP y HTTPS a servidores WEB.
- 9 **XML-RPC:** Protocolo de llamada a procedimiento remoto que usa XML para codificar las llamadas.
- 10 **XLL:** EXtensible Linking Language. Define el modo de enlace entre diferentes enlaces
- 11 **XSL:** Extensible Stylesheet Language. Permite describir la información contenida en un documento XML
- 12 **XSLT:** XSLT o XSL Transformaciones. Estándar de la organización W3C que presenta una forma de  
13 transformar documentos XML en otros e incluso a formatos que no son XML.
- 14 **XUA:** XML User Agent. Estandarización de navegadores XML.
- 15