



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

“Sistema de Información para la Reservación de Espacios del
Instituto y la Torre de Ingeniería”

TESIS

QUE PARA OBTENER EL TITULO DE

Ingeniero en Computación

PRESENTA:

Maribel Guzmán Marcial

DIRECTOR:

Ing. Marco Ambriz Maguey



Noviembre de 2006



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A la Universidad Nacional Autónoma de México, por abrir sus puertas y darme la oportunidad de pertenecer a su comunidad.

A la Facultad de Ingeniería, por permitirme ser una más de sus alumnas y formarme como ingeniero.

Al Instituto de Ingeniería, por creer en mí y darme la oportunidad de formar parte de su equipo de trabajo.

Al Ing. Marco Ambriz por aceptar dirigir este trabajo de tesis.

Gracias a todos por creer en mí.

Dedicatoria:

A mi mami que siempre me ha apoyado, gracias mami por tus sabios consejos, por tu gran esfuerzo y por tu gran corazón, te quiero mucho.

A mis hermanos Gude y Rafa.

A mis amigos y amigas.

A ti amor por apoyarme en todos los sentidos, te amo.

“No importa dónde te encuentres, si eres una estrella, siempre brillarás por luz propia.”

CONTENIDO

INTRODUCCIÓN	V
<u>1 SISTEMA DE INFORMACIÓN PARA LA RESERVACIÓN DE ESPACIOS DEL INSTITUTO Y LA TORRE DE INGENIERÍA</u>	<u>1</u>
1.1 ANTECEDENTES	1
1.1.1 INSTITUTO DE INGENIERÍA	1
1.1.1.1 Organización del II	2
1.1.1.2 Eventos académicos en el II	4
1.1.2 TORRE DE INGENIERÍA	4
1.1.2.1 Eventos académicos en la TI	4
1.1.3 COLABORACIÓN ENTRE AMBAS DEPENDENCIAS	5
1.2 PROBLEMÁTICA	5
1.3 PROPUESTA DE SOLUCIÓN	6
1.4 OBJETIVO	7
1.5 VENTAJAS	7
<u>2 REQUISITOS</u>	<u>9</u>
2.1 ACTORES DEL SISTEMA	9
2.2 ACCIONES A REALIZAR	10
2.3 DESCRIPCIÓN DE CASOS DE USO	11
2.3.1 AUTENTICAR USUARIO	11
2.3.2 ACTUALIZAR INFORMACIÓN EN CATÁLOGOS	12
2.3.2.1 Actualizar información del II	12
2.3.2.2 Actualizar información de la TI	13
2.3.3 MODIFICAR EL ESTADO DE LA RESERVACIÓN	15
2.3.4 CONSULTAR RESERVACIONES POR FECHA Ó NÚMERO DE SOLICITUD Y POR FECHA DE EVENTO	16
2.3.5 ELIMINAR RESERVACIONES	17
2.3.6 GENERAR REPORTES DE RESERVACIONES	18
2.3.7 MODIFICAR INFORMACIÓN DE LA RESERVACIÓN	19
2.3.8 REALIZAR RESERVACIÓN	20
2.3.9 REALIZAR RESERVACIÓN CON SERVICIOS ADICIONALES	21
2.4 DIAGRAMAS DE CASOS DE USO	23
2.5 REQUISITOS ADICIONALES	29
2.5.1 REQUISITOS MÍNIMOS DE PLATAFORMA HARDWARE	29
2.5.2 REQUISITOS DE PLATAFORMA SOFTWARE	29
<u>3 ANÁLISIS</u>	<u>31</u>

3.1 REALIZACIONES DE CASOS DE USO – ANÁLISIS	32
3.1.1 DIAGRAMAS DE COLABORACIÓN	32
3.1.2 CLASES DE INTERFAZ	32
3.1.3 CLASES DE ENTIDAD	33
3.1.4 CLASES DE CONTROL	33
3.1.5 DIAGRAMAS DE COLABORACIÓN DE LAS REALIZACIONES DE LOS CASOS DE USO DEL SREITI	34
3.1.5.1 Autenticar usuario	34
3.1.5.2 Actualizar información en catálogos	35
3.1.5.3 Actualizar información del II	35
3.1.5.4 Actualizar información de la TI	36
3.1.5.5 Activar o desactivar el estado de servicios adicionales	37
3.1.5.6 Confirmar o cancelar la reservación	38
3.1.5.7 Consultar reservaciones por fecha ó número de solicitud ó por fecha de evento	39
3.1.5.8 Eliminar reservaciones	40
3.1.5.9 Generar reportes de reservaciones	41
3.1.5.10 Modificar información de la reservación	42
3.1.5.11 Realizar reservación	42
3.1.5.12 Realizar reservación con servicios adicionales	44
3.2 PAQUETES DEL ANÁLISIS	45
4 DISEÑO	47
4.1 DISEÑO CONCEPTUAL	47
4.2 DISEÑO LÓGICO	47
4.2.1 REALIZACIONES DE CASOS DE USO – DISEÑO	48
4.2.1.1 Diagrama de Clases	48
4.2.1.2 Diagramas de Secuencia	51
4.2.1.3 Diagrama de actividades	59
4.2.2 DISEÑO DE LA BASE DE DATOS	60
4.2.2.1 Diagrama entidad - relación	61
4.2.2.2 Descripción de las entidades del diagrama entidad – relación del SREITI 65	
4.2.3 EL SREITI COMO DOS SUBSISTEMAS	73
4.2.4 PATRÓN DE DISEÑO EN 3 CAPAS	74
4.3 DISEÑO FÍSICO	77
4.3.1 MODELO DE DESPLIEGUE	77
5 IMPLEMENTACION	81
5.1 MODELO DE IMPLEMENTACIÓN	81
5.2 APLICACIÓN WEB ASP .NET PARA EL SREITI	81
5.2.1 COMPONENTES	82
5.2.1.1 Interfaces de usuario	82
5.2.1.2 Servicios Web	83
5.2.1.3 Acceso a datos	83
5.2.1.4 Manejador de Base de Datos	83
5.2.1.5 Configuración	84
5.2.1.6 Diagramas de componentes	84

5.3 ARQUITECTURA DE SEGURIDAD	93
6 PRUEBAS	95
6.1 MODELO DE PRUEBAS	95
6.1.1 CASOS DE PRUEBA	95
6.1.1.1 Pruebas de unidad	95
6.1.1.2 Pruebas de integración	104
6.1.1.3 Interfaces de usuario obtenidas en las pruebas de integración	106
6.1.1.4 Pruebas de sistema	108
CONCLUSIONES	109
GLOSARIO	111
REFERENCIA BIBLIOGRÁFICA	113

INTRODUCCIÓN

El Instituto y la Torre de Ingeniería de la UNAM son centros de investigación y desarrollo tecnológico en distintas áreas de la Ingeniería que contribuyen al crecimiento de nuestro país en estos dos aspectos. En ambos se fomenta la formación de estudiantes a través de su participación en los proyectos de investigación y desarrollo tecnológico; se contribuye a la efectiva capacitación de alto nivel; se genera conocimiento científico y tecnológico y se promueve la colaboración entre grupos académicos para abordar problemas tecnológicos de gran alcance y la difusión de sus resultados.

Una forma de coadyuvar a las actividades que realizan, es permitiendo la realización de eventos de tipo académico dentro de sus instalaciones a personal interno y externo a las mismas a través de la renta y préstamo gratuito de sus espacios.

Actualmente ambas organizaciones no cuentan con un sistema informático que les facilite la administración y reservación de sus espacios, evitando con ello el manejo de muchos documentos y una demora en tiempo como consecuencia del trámite de aprobación de la renta o préstamo gratuito de los mismos. Por su parte los usuarios, tienen la necesidad de desplazarse hasta la dependencia respectiva para realizar su reservación o bien consultar disponibilidad vía telefónica en días y horarios de oficina y posteriormente redactar y mandar un oficio para realizar la reservación.

Ante esta situación, el presente trabajo de Tesis proporciona una solución a través del desarrollo de un sistema de información que está basado en el seguimiento de una metodología capaz de proporcionar un método sistemático de desarrollo de forma que se pueda controlar el mismo para así lograr los objetivos planteados de forma exitosa, sin dejar de lado el empleo de Tecnologías de Información que sirven como soporte en dicha tarea.

A lo largo de cada capítulo que conforma este trabajo, se explica como fue desarrollado el sistema. A continuación se describe brevemente el contenido de cada uno de ellos.

En el Capítulo 1 se describe la problemática actual y la justificación del desarrollo de un sistema informático, así como también el objetivo y las ventajas que éste brinda al Instituto y la Torre de Ingeniería.

En el capítulo 2 está dedicado a la definición de los requerimientos del usuario y del sistema.

En el capítulo 3 se hace un análisis del sistema, partiendo de los requisitos de los usuarios en general.

En el Capítulo 4 se plantea el diseño del sistema.

En el capítulo 5 se explica la implementación del sistema.

Finalmente en el Capítulo 6 se describen las pruebas realizadas al sistema.

1 SISTEMA DE INFORMACIÓN PARA LA RESERVACIÓN DE ESPACIOS DEL INSTITUTO Y LA TORRE DE INGENIERÍA

El Sistema de Reservación de Espacios del Instituto y la Torre de Ingeniería (SREITI) surge como una necesidad de ambas dependencias por realizar a través de Internet la reservación de sus espacios, para así permitir que dicho proceso se lleve a cabo de forma rápida y fácil por todas aquellas personas interesadas en realizar actividades de tipo académico en los espacios del Instituto y la Torre de Ingeniería.

1.1 Antecedentes

1.1.1 Instituto de Ingeniería

El Instituto de Ingeniería de la Universidad Nacional Autónoma de México (IIUNAM) es el centro de investigación en diversas áreas de la ingeniería más productivo del país. Es una comunidad integrada por 93 investigadores, 95 técnicos académicos, 409 becarios que realizan trabajos de tesis de licenciatura, maestría y doctorado y 184 personas del área administrativa. Sus instalaciones ocupan 13 edificios en la zona de Ciudad Universitaria, en la ciudad de México, con una extensión de 20,000 metros cuadrados construidos entre laboratorios, cubículos, áreas comunes y un auditorio.

Desde su fundación, la política del Instituto ha sido realizar investigación orientada a problemas generales de la ingeniería, colaborar con entidades públicas y privadas para mejorar la práctica de la ingeniería en el ámbito nacional, proporcionar servicios de ingeniería a los diversos sectores de la sociedad. Asimismo, ha puesto especial atención en la formación de recursos humanos y en difundir los resultados de sus investigaciones, contribuyendo así al desarrollo del país y al bienestar de la sociedad.

1.1.1.1 Organización del II

El Instituto de Ingeniería se organiza en grupos de investigación distribuidos en 15 coordinaciones, agrupadas a su vez en tres subdirecciones:

- Subdirección de Hidráulica y Ambiental
 - Hidráulica
 - Ingeniería Ambiental
 - Bioprocesos Ambientales
 - Ingeniería de Procesos Industriales y Ambientales

- Subdirección de Estructuras
 - Estructuras y Materiales
 - Geotecnia
 - Ingeniería Sismológica
 - Mecánica Aplicada
 - Sismología e Instrumentación Sísmica
 - Vías Terrestres

- Subdirección de Electromecánica y Cómputo
 - Automatización
 - Ingeniería de Sistemas
 - Ingeniería Mecánica, Térmica y de Fluidos
 - Instrumentación
 - Sistemas de Cómputo

La estructura organizacional del II, también está conformada por otros grupos tal y como se puede apreciar en la figura de la siguiente página.



Figura 1.1 Organigrama del II.

1.1.1.2 Eventos académicos en el II

Dentro de las actividades que realiza el II, está la presentación de eventos académicos, estos son llevados a cabo en espacios que se encuentran dentro de las propias instalaciones del II, mismos que sólo pueden ser utilizados por personal interno del II, razón por la cual no se cobra por su uso, salvo uno de ellos (Salón de seminarios Emilio Rosenblueth) el cual puede ser también utilizado por personal de la UNAM o por instituciones externas. En la siguiente tabla se puede observar cómo están distribuidos estos espacios.

Tabla 1.1 Espacios del II.

Organización	Espacio
Dirección y Secretarías	Sala de juntas de la Dirección
	Sala de juntas Emilio Rosenblueth
	Sala de juntas de ex directores
	Salón de seminarios Emilio Rosenblueth
Subdirección de Electromecánica	Sala de juntas de Electromecánica
	Sala de juntas de Automatización
Subdirección de Estructuras	Sala de juntas de Estructuras
	Sala de juntas de Geotecnia
Subdirección de Hidráulica y Ambiental	Sala de juntas de Ambiental
	Sala de juntas de Ingeniería Ambiental

1.1.2 Torre de Ingeniería

Por su parte, la Torre de Ingeniería (TI) es una unidad Universitaria de nueva creación dirigida a fortalecer la colaboración que realiza la UNAM con los sectores productivo, social y servicios. Actualmente en la TI participan cuatro entidades universitarias: la Facultad de Ingeniería, la Facultad de Química, el Instituto de Ingeniería y el Centro de Ciencias Aplicadas y Desarrollo Tecnológico. Los directores de las cuatro entidades junto con un representante de Rectoría y uno del Patronato Universitario, integran el Consejo Directivo de la Torre de Ingeniería, que es el órgano rector de la administración de la Torre.

1.1.2.1 Eventos académicos en la TI

Con el objetivo de permitir la realización de eventos de tipo académico dentro de sus instalaciones a personal interno y externo a las mismas, la Torre de Ingeniería brinda el servicio de renta de sus espacios. El uso de estos espacios requiere el cobro de una cuota económica, misma que está en función del tipo de dependencia a la que pertenezca el solicitante.

En la siguiente tabla, se puede observar un listado de los mismos en la forma en que son clasificados por la TI.

Tabla 1.2 Espacios de la TI.

Individuales	Salones de usos múltiples	Espacios del nivel de la terraza
Auditorio	Sala Sur 1	Cafetería – Terraza y Cocina
Cafetería	Sala Sur 2	Oficina Ejecutiva
Sala de exposiciones	Sala Sur 3	Sala de Juntas de Consejo
	Sala Sur 4	Sala de Juntas
	Sala Norte 1	Recepción
	Sala Norte 2	

1.1.3 Colaboración entre ambas dependencias

Dentro de los proyectos que se desarrollan en ambas dependencias, se encuentran aquellos que están relacionados con la ingeniería de software, y es aquí en donde de acuerdo a sus necesidades ambas dependencias colaboran entre sí para poder realizarlos.

El Instituto de Ingeniería a través de su Coordinación de Cómputo y más en específico el área de Base de Datos y Sistemas de Información es la encargada de realizar este tipo de proyectos. Es así como la realización, administración y mantenimiento del SREITI está a cargo del II.

1.2 Problemática

Actualmente, la reservación de espacios en la TI, involucra principalmente dos procesos, uno es el que realiza la persona que quiere reservar el espacio y otro el que lleva a cabo la persona encargada de administrar la renta del mismo. En el primero, el usuario debe desplazarse forzosamente hasta la oficina correspondiente para consultar horarios, espacios disponibles en fecha solicitada, equipo adicional que se incluye y costos correspondientes a éste último y al tiempo de uso del espacio; o bien realizar una llamada telefónica en días y horas de oficina solicitando la información correspondiente, de esta forma, el administrador consulta la disponibilidad y si ésta existe, el usuario deberá redactar y mandar un oficio para realizar su reservación. En el segundo, para autorizar una reservación, debe consultar un calendario y un listado de las reservaciones confirmadas anteriormente para verificar que no se traslapen los eventos, también debe consultar un listado con las cuotas económicas correspondientes al espacio y equipo adicional seleccionado por el usuario. Por otro lado para poder acomodar el mobiliario con el que cuenta el espacio en una forma determinada, se debe imprimir toda la información correspondiente para entregarla al personal encargado de realizar esta tarea.

En el II, el proceso cambia un poco, pues como ya se había mencionado antes, sólo un espacio puede ser rentado y todos los demás son utilizados por personal interno del II. Para realizar la reservación de estos últimos, el usuario debe anotar su

nombre, el nombre de evento y horario del mismo en una hoja de papel y pegarla en la entrada principal del espacio que esté solicitando; lo cual lleva a la situación de que si la hoja se llega a despegar y perder o alguien la desprende, las personas que se habían anotado en ella, regresan al mismo lugar para confirmar el estado de su reservación o bien para realizar su evento y en su lugar, encuentran una hoja nueva con otra información u otro evento que ya se está llevando a cabo. Para reservar el espacio que si requiere de una cuota para su uso, el usuario debe acudir a la oficina del personal encargado de administrar este espacio y solicitar la reservación del mismo, al igual que en la TI, el administrador debe consultar la disponibilidad y las cuotas correspondientes. Aquí, la distribución del mobiliario con el que cuentan los espacios, no cambia.

Esta forma, en la cual ambas dependencias llevan a cabo el proceso de reservación de sus espacios involucra el manejo de muchos documentos, una demora en tiempo como consecuencia del trámite de aprobación de la renta o préstamo gratuito de sus espacios, descontento y retraso en las actividades de los usuarios.

1.3 Propuesta de solución

Ante las situaciones descritas en la sección anterior y debido a que ambas dependencias tienen mucha similitud en sus procesos de reservación de espacios y además colaboran entre ellas, se decidió resolver la problemática de cada una con un solo sistema informático que ayude a optimizar a ambas el proceso de reservación y administración de sus espacios. Así mismo, se propone que el sistema, tenga distintas interfaces gráficas para de esta forma conservar la entidad gráfica de cada dependencia.

Para poder llevar a cabo de forma exitosa la solución planteada, se necesita la ayuda de una metodología de desarrollo de software capaz de proporcionar un método sistemático de desarrollo de forma que se pueda controlar el mismo para así lograr los objetivos planteados, sin dejar de lado el empleo de Tecnologías de Información que sirven como soporte en dicha tarea. Hoy en día, se sabe que existen muchas metodologías y Tecnologías de Información y que de acuerdo a las necesidades de nuestro proyecto de software podemos optar por una de ellas. En este caso se optó por utilizar como metodología de desarrollo de software al Proceso Unificado de Rational (Rational Unified Process, RUP) que a su vez utiliza al Lenguaje Unificado de Modelado (Unified Modeling Language, UML) como una herramienta para especificar, construir, visualizar y documentar un sistema software. El Proceso Unificado se caracteriza por estar dirigido por casos de uso, centrado en la arquitectura y ser iterativo e incremental que en contrapartida con un desarrollo secuencial en cascada, en un desarrollo iterativo se gana conocimiento del sistema a través de refinamientos sucesivos.

Hasta ahora, sólo se ha hablado acerca del cómo se va a desarrollar el software, pasemos a la parte del cómo será implementado el mismo, tarea que no es nada fácil, pues conforme la tecnología va avanzando, aparecen nuevas soluciones nuevas formas de programación, nuevos lenguajes y un sin fin de herramientas que intentan hacer la labor del desarrollador un poco más fácil. Así mismo, hoy en día estamos en una época de transición, encaminados hacia un nuevo estilo de programación basada en estándares; por ello, se eligió a la tecnología .NET de Microsoft como herramienta de desarrollo de software. Esta tecnología cuenta con una plataforma comúnmente llamada plataforma .NET la cual sirve para el desarrollo de aplicaciones en general (no necesariamente para Windows, ni de tres capas) cuyo objetivo principal, es ofrecer una red de servicios que se ejecutan bajo

un entorno común (Common Language Runtime, CLR) y que hayan sido diseñados con cualquier lenguaje compatible con el CLR (Lenguajes de este tipo son C#, C++ y Visual Basic) a través de su entorno de desarrollo conocido como Visual Studio .NET. Por otro lado, se utilizará C# como lenguaje de programación. Éste, es un lenguaje de programación orientado a objetos que deriva de C y C++ eliminando características inseguras y complejas de éstos y añadiendo otras que ofrecen mayor comodidad. Finalmente, para almacenar todos los datos que nos ayuden a llevar un control del sistema de información, se usará una base de datos creada con el sistema manejador de base de datos SQL Server 2000.

1.4 Objetivo

Realizar un sistema de información para realizar la reservación de espacios del Instituto y la Torre de Ingeniería que sea capaz de agilizar y hacer eficiente el actual proceso que se lleva a cabo.

1.5 Ventajas

Al contar con un sistema de información como el propuesto en la sección 1.3, podemos listar las siguientes ventajas:

- Administración y reservación de espacios desde cualquier lugar con acceso a Internet en el que se encuentren los usuarios y los administradores de los espacios.
- Optimización de trámites administrativos.
- Fácil administración de espacios, costos y reservaciones.
- Ahorro en tiempo para realizar y autorizar reservaciones.
- Conservación de entidad gráfica de cada dependencia.
- Manejo rápido y cómodo de la información.

2 REQUISITOS

En este capítulo, se identifican las necesidades de los usuarios como requisitos (es decir, las condiciones o capacidades que el sistema debe cumplir) en forma de un modelo de casos de uso. Este modelo se realiza mediante una descripción general del sistema identificando sus actores y las acciones a realizar por los mismos, la descripción detallada de cada caso de uso (requisitos funcionales) y un conjunto de diagramas. Los demás requisitos (no funcionales) se listan aparte.

Con la realización del modelo de casos de uso, se obtiene la vista conceptual de la arquitectura del sistema. Esta vista constituye una de las cinco vistas de las que se compone la arquitectura software en el PU, las cuales son representadas por los siguientes modelos: modelo de casos de uso, modelo de análisis, modelo de diseño, modelo de implementación y modelo de pruebas. Gracias a este conjunto de modelos es como se logra obtener una arquitectura robusta que sea capaz de definir todos aquellos elementos estructurales que compondrán el sistema y sus interfaces, junto con sus comportamientos.

Para dar comienzo a esta tarea, a continuación se definen a los actores del sistema.

2.1 Actores del sistema

Un actor representa un conjunto coherente de roles que los usuarios de los casos de uso juegan al interactuar con éstos. Normalmente, un actor representa un rol que es jugado por una persona, un dispositivo hardware o incluso otro sistema al interactuar con nuestro sistema.

En la definición de los requerimientos del SREITI (Sistema de Reservación de Espacios del Instituto y la Torre de Ingeniería), se identificaron a los siguientes actores clasificándolos por dependencia.

Instituto de Ingeniería, II

- Usuario II (Estudiantes de servicio social, Becarios, y Personal Académico y Administrativo del II).
- Administrador II (Director, Subdirectores, Coordinadores y Secretarías representantes de cada Subdirección y de la Dirección del II)

Torre de Ingeniería, TI

- Usuario TI (Estudiantes, Becarios, Secretarías, Personal Académico de la UNAM y personal externo a la misma).
- Administrador TI (Coordinador Administrativo, Jefe de departamento y Secretaría de Coordinación Administrativa)

2.2 Acciones a realizar

En esta sección, se describen las acciones que en general los actores requieren realizar en el sistema de forma separada, es decir con interfaces de usuario diferentes, políticas de uso variantes y con información correspondiente a cada institución.

- Autenticarse en el sistema.
- Actualizar¹ información en catálogos tales como:
 - Cuotas de espacios, equipo adicional y servicio de cafetería (agua, café, galletas, refresco, otros).
 - Destinatarios de correo electrónico.
 - Distribución de mobiliario con el que cuentan los espacios.
 - Imágenes de espacios.
 - Número de personas por el cual se cobrará el consumo de determinado producto.
 - Productos de servicio de cafetería.
 - Tipo de equipo adicional (hasta el momento sólo se cuenta con equipo audiovisual y mobiliario) y equipos asociados a los tipos existentes.
 - Tipo de dependencia y dependencias asociadas al mismo.
 - Tipo de espacio y espacios asociados al mismo.
- Activar o desactivar el estado de servicios adicionales (equipo adicional y servicio de cafetería).
- Modificar el estado (Cancelada, Confirmada y Pendiente) de la reservación.
- Consultar reservaciones por fecha ó número de solicitud y por fecha de evento.
- Eliminar reservaciones.
- Generar reportes de reservaciones.
- Modificar información de la reservación.
- Realizar reservación.
- Realizar reservación con la opción de agregar servicios adicionales (equipo adicional y servicio de cafetería).

¹ Se utiliza el término "Actualizar" en sentido genérico, para referirse a todas las operaciones de inserción, borrado y modificación.

2.3 Descripción de Casos de Uso

Un caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso.

Los actores y las acciones a realizar descritas anteriormente nos dan una idea acerca del contexto del sistema y son una base para la descripción de los casos de uso del mismo. En la descripción de éstos últimos, se hacen las consideraciones necesarias para poder hacer posible el manejo de la información de forma conveniente para cada institución.

A continuación se describe cada uno de los casos de uso del SREITI.

2.3.1 Autenticar usuario

Actores:	Usuario II/Administrador II/Administrador TI
Casos de uso que incluye:	Ninguno
Propósito:	Autenticar en el sistema a usuarios y administradores de los espacios del II y a administradores de la TI con una cuenta de Windows.
Resumen:	El Usuario II/Administrador II/Administrador TI se autenticará en el sistema con la cuenta de Windows que tiene asignada en el II y con ello para el caso de los Usuarios y Administradores del II se determinará a qué Subdirección pertenece y se obtendrá información personal del mismo. Se enviará a otra página de acuerdo a su categoría (Usuario II/Administrador II/Administrador TI).
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. El Usuario II/Administrador II/Administrador TI accede al sistema a través de una página en Internet.	2. Solicita nombre de usuario y contraseña.
3. El Usuario II/Administrador II/Administrador TI escribe su nombre de usuario y contraseña correspondientes a la cuenta de Windows que tienen asignada en el II.	4. Valida al Usuario II/Administrador II/Administrador TI y para el caso del Usuario II y el Administrador II determina a que subdirección pertenece y obtiene información personal del mismo. Envía a otra página de acuerdo a su categoría (Usuario II/Administrador II/Administrador TI)
Cursos alternos	
Línea 1:	El Usuario II/Administrador II/Administrador TI escribe mal el URL de la página e impide continuar con el proceso.
Línea 3:	El nombre de usuario y/o contraseña son

	incorrectos, el sistema muestra un mensaje de acceso no autorizado e impide que se continúe con el proceso.
--	---

2.3.2 Actualizar información en catálogos

La actualización de la información en catálogos, varía en función de la dependencia, debido a la variante en la que realizan la reservación de sus espacios; mientras que el II sólo requiere realizarla, para la TI además de ello, también le es necesario brindar el servicio de cafetería y la renta de equipo adicional.

2.3.2.1 Actualizar información del II

Actores:	Administrador II.
Casos de uso que incluye:	Autenticar usuario.
Propósito:	<p>Actualizar información de alguna de las siguientes opciones.</p> <ul style="list-style-type: none"> i) <i>Cuota de espacio.</i> El Administrador II modifica la cuota económica del espacio en función de la dependencia a la que pertenezca el usuario. ii) <i>Destinatarios de correos electrónicos.</i> El Administrador II ingresa, elimina o modifica los destinatarios de los correos electrónicos que serán enviados por el sistema cada vez que se realice una reservación ó el administrador modifique el estado de la misma (confirmado ó cancelado). iii) <i>Espacio.</i> El Administrador II ingresa espacios o modifica características de los existentes (nombre, descripción, capacidad máxima, otros). iv) <i>Imagen.</i> El Administrador II asigna, elimina o modifica imágenes asociadas a los espacios.
Resumen:	El Administrador II selecciona del menú principal del sistema la información a modificar. El sistema envía a la interfaz de administrador correspondiente a la opción elegida. El Administrador II actualiza la información. El sistema valida y guarda información e informa al Administrador II de este suceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona del menú	2. Envía a interfaz de administrador

principal del sistema, la información a actualizar.	correspondiente a la opción elegida.
3. Actualiza información (realiza cambios, ingresa información).	4. Valida información. Guarda la actualización realizada por el Administrador II y le informa de este suceso.
5. Lee mensaje. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Línea 4:	La información no es válida e impide continuar con el proceso.

2.3.2.2 Actualizar información de la TI

Actores:	Administrador TI.
Casos de uso que incluye:	Autenticar usuario.
Propósito:	<p>Actualizar información de alguna de las siguientes opciones.</p> <ul style="list-style-type: none"> i) <i>Cuotas económicas de espacios, equipo adicional y servicio de cafetería (agua, café, galletas, refresco, otros).</i> El Administrador TI modifica cuotas en función de la dependencia a la que pertenezca el usuario y en el caso del servicio de cafetería, también se considera el número de personas que consumen el producto y características propias del mismo. ii) <i>Destinatarios de correos electrónicos.</i> El Administrador TI ingresa, elimina o modifica los destinatarios de los correos electrónicos que serán enviados por el sistema cada vez que se realice una reservación ó el administrador modifique el estado de la misma (confirmado ó cancelado). iii) <i>Distribución del mobiliario de los espacios.</i> El Administrador TI ingresa, modifica, elimina o asigna distribuciones del mobiliario con el que cuenta cada uno de los espacios. iv) <i>Imagen.</i> El Administrador TI asigna, elimina o modifica imágenes asociadas a los espacios. v) <i>Número de personas.</i> El Administrador TI ingresa o modifica un número de personas por el cual se cobrará el

	<p>consumo de un determinado producto (agua, café, galletas, refresco, otros).</p> <p>vi) <i>Productos de servicio de cafetería.</i> El Administrador TI ingresa un producto ó modifica características de los existentes.</p> <p>vii) <i>Servicios adicionales.</i> Activar o desactivar el estado de los servicios adicionales (equipo adicional y servicio de cafetería) para poder ponerlos a disposición de los usuarios en el momento que se requiera.</p> <p>viii) <i>Tipo de equipo (equipo audiovisual ó mobiliario) y equipos adicionales asociados al mismo.</i> El Administrador TI ingresa ó modifica un tipo de equipo ó un equipo asociado al mismo.</p> <p>ix) <i>Tipo de dependencia y dependencias asociadas al mismo.</i> El Administrador TI ingresa ó modifica un tipo de dependencia ó una dependencia asociada al mismo.</p> <p>x) <i>Tipo de espacio y espacios asociados al mismo.</i> El Administrador TI ingresa ó modifica un tipo de espacio ó un espacio asociado al mismo.</p>
Resumen:	El Administrador TI selecciona del menú principal del sistema la información a modificar. El sistema envía a interfaz de administrador correspondiente a la opción elegida. El Administrador II actualiza la información. El sistema valida y guarda información e informa al Administrador II de este suceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona del menú principal del sistema, la información a actualizar.	2. Envía a interfaz de administrador correspondiente a la opción elegida.
3. Actualiza información (realiza cambios, ingresa información).	4. Valida información Guarda la actualización realizada por el Administrador II y le informa de este suceso.
5. Lee mensaje. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Línea 4:	La información no es válida e impide continuar con el proceso.

2.3.3 Modificar el estado de la reservación

Actores:	Administrador TI/Administrador II.
Casos de uso que incluye:	Autenticar usuario y Consultar reservaciones por número de solicitud.
Propósito:	Cambiar el estado de la reservación (Cancelada, Confirmada y Pendiente).
Resumen:	El Administrador en cuestión selecciona la reservación a la cual le quiere modificar el estado (Cancelada, Confirmada ó Pendiente). Indica al sistema que desea continuar con el proceso. El sistema realiza el cambio y le informa al Administrador de este suceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona de la lista de reservaciones, la reservación a la cual le quiere modificar el estado (Cancelada, Confirmada ó Pendiente). Indica al sistema que desea continuar con el proceso.	2. Envía a otra página en donde muestra características más precisas de la reservación y una opción para modificar el estado de la misma.
3. Modifica el estado de la reservación e indica al sistema que desea continuar con el proceso.	4. Valida información. Guarda los cambios realizados de la opción elegida e informa al Administrador de este suceso.
5. Verifica mensaje e información. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Ninguno.	

2.3.4 Consultar reservaciones por fecha ó número de solicitud y por fecha de evento

Actores:	Administrador TI/Administrador II.
Casos de uso que incluye:	Autenticar usuario.
Propósito:	<p>Consultar registros de reservaciones en el sistema por fecha o número de solicitud y por fecha de evento para poder realizar las siguientes acciones:</p> <ul style="list-style-type: none"> i) <i>Revisar el registro de reservaciones que se tienen.</i> ii) <i>Eliminar reservaciones.</i> iii) <i>Modificar el estado de las reservaciones.</i> iv) <i>Modificar información de la reservación.</i> <p>Nota: Al Administrador II, sólo le es necesario realizar la consulta de reservaciones por número de solicitud.</p>
Resumen:	El Administrador en cuestión, selecciona una opción de consulta e indica al sistema que desea continuar con el proceso. El sistema valida información recibida y realiza una búsqueda de las reservaciones existentes según la opción elegida; finalmente muestra información encontrada.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona una opción de consulta. Indica al sistema que desea continuar con el proceso.	2. Valida información y realiza búsqueda de reservaciones. Muestra lista de reservaciones encontradas.
3. Verifica reservaciones encontradas. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Línea 2:	La información no es válida ó no existe el número de solicitud e impide continuar con el proceso.

2.3.5 Eliminar reservaciones

Actores:	Administrador TI/Administrador II.
Casos de uso que incluye:	Autenticar usuario.
Propósito:	Eliminar reservaciones que no tengan utilidad.
Resumen:	El Administrador en cuestión, selecciona de la lista de reservaciones mostrada por el sistema, la reservación a eliminar e indica al sistema que desea continuar con el proceso. El sistema elimina la opción elegida e informa al Administrador de este suceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona de la lista de reservaciones mostrada por el sistema, la reservación a eliminar. Indica al sistema que desea continuar con el proceso.	2. El sistema elimina la opción elegida e informa al Administrador de este suceso.
3. Verifica mensaje e información. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Ninguno.	

2.3.6 Generar reportes de reservaciones

Actores:	Administrador TI/Administrador II
Casos de uso que incluye:	Autenticar usuario.
Propósito:	Generar reportes de las reservaciones realizadas en el sistema.
Resumen:	El Administrador en cuestión, selecciona año, mes, estado de la reservación y costo (opcional) para generar un reporte. El sistema muestra una lista de reservaciones de acuerdo a las opciones elegidas y una opción para generar el reporte en Excel. El Administrador tiene la opción de indicar al sistema que genere un reporte en Excel.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona año, mes, estado de la reservación y costo (opcional) para generar un reporte e indica al sistema que desea continuar con el proceso.	2. Muestra una lista de reservaciones de acuerdo a las opciones elegidas y una opción para generar el reporte en Excel.
3. Verifica información y puede optar por indicarle al sistema que genere un reporte en Excel.	4. Genera reporte en Excel si es que así se le indica.
5. Verifica mensaje e información. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Ninguno.	

2.3.7 Modificar información de la reservación

Actores:	Administrador TI/Administrador II
Casos de uso que incluye:	Autenticar usuario y Consultar reservaciones por número de solicitud.
Propósito:	Modificar características de la reservación tales como: horario, fecha, nombre de evento, espacio, distribución, servicios adicionales, otros.
Resumen:	El Administrador en cuestión selecciona la reservación a la cual le quiere realizar modificaciones tales como: horario, fecha, nombre de evento, espacio, distribución, servicios adicionales, otros. Indica al sistema que desea continuar con el proceso. El sistema realiza el cambio y le informa al Administrador de este suceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona de la lista de reservaciones, la reservación a la cual le quiere realizar cambios e indica al sistema que desea continuar con el proceso.	2. Envía a otra página en donde muestra características más precisas de la reservación y una opción para realizar modificaciones.
3. Selecciona la opción para realizar modificaciones.	4. Permite realizar modificaciones.
5. Realiza modificaciones e indica al sistema que desea continuar con el proceso.	6. Valida información. Guarda los cambios realizados e informa al Administrador de este suceso.
7. Verifica mensaje e información. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Línea 6:	La información no es válida e impide continuar con el proceso.

2.3.8 Realizar reservación

Actores:	Usuario TI/Administrador TI/Usuario II/Administrador II.
Casos de uso que incluye:	Autenticar usuario.
Propósito:	Reservar uno o más espacios correspondientes a la institución a la que pertenezca el usuario en cuestión.
Resumen:	<p>El usuario selecciona características de su reservación tales como: fecha o intervalo de fechas, espacio, horario, número de personas, nombre de evento. El sistema valida la información introducida y verifica que no exista una reservación con las mismas características o bien un traslape de eventos.</p> <p>Si la información es válida y no existe una reservación con las mismas características ó un traslape de eventos, el usuario puede continuar con el proceso. El sistema envía a otra página que muestra una forma en donde el usuario debe llenar con información personal del mismo (nombre, teléfono, correo electrónico, tipo de dependencia y dependencia) e indicar que desea continuar con el proceso. El sistema envía a otra página con la opción de escribir comentarios y si existe una cuota por el uso del espacio, el sistema le da al usuario la opción de cotizar el uso del ó los espacios a utilizar. El usuario escribe o no comentarios e indica al sistema que realice la cotización. El sistema realiza la cotización en función al tipo de dependencia a la que pertenece el usuario, tiempo de uso y al espacio elegido, y la muestra. El usuario realiza la reservación o hace cambios para modificar costos e indica al sistema que desea continuar con el proceso. El sistema genera un número de solicitud, registra la ó las reservaciones de esta solicitud con estado pendiente para el caso de los espacios correspondientes a la TI y con estado confirmado para los espacios correspondientes al II, y envía un correo electrónico al usuario y al administrador de los espacios correspondientes a la TI o al II, notificando que se ha realizado la reservación de uno o más espacios. Envía a otra página presentando un resumen de la información registrada. Este resumen es el comprobante de solicitud para el usuario.</p> <p>De lo contrario deberá elegir otro horario o fecha de evento.</p>
Curso normal de los eventos	

Acción del actor	Acción del sistema
1. Selecciona el tipo de reservación que desea realizar (reservación por día o por intervalo de días).	2. Muestra un calendario.
3. Selecciona una fecha o un intervalo de fechas para su evento.	4. Valida fecha(s) seleccionada(s). Presenta dos listas; una, con eventos si es que existen para la(s) fecha(s) seleccionada(s) y otra con el tipo de espacio a reservar (Sólo para los espacios correspondientes a la TI).
5. Selecciona un tipo de espacio (Sólo el Usuario TI ó el Administrador TI).	6. Muestra una lista de espacios de acuerdo a la opción seleccionada y una forma para introducir información correspondiente al evento (horario, número de personas, nombre de evento).
7. Selecciona espacio, número de personas, horario, nombre de evento. Indica al sistema que ha terminado de seleccionar las características necesarias para su evento.	8. Valida que el tiempo de sesión de usuario sea menor a 20 minutos. Valida capacidad de espacio y horario. Verifica que no exista un evento con las mismas características o un traslape de eventos. Muestra una lista con la información introducida por el usuario.
9. Verifica su información en la lista. Si ésta es correcta, le indica al sistema que desea continuar con el proceso; de lo contrario, puede eliminar la reservación y realizar otra o bien hacer cambios en la lista y continuar con el proceso.	10. Muestra otra página con una forma que el usuario debe llenar con información personal del mismo (nombre, teléfono, correo electrónico, tipo de dependencia y dependencia).
11. Llena la forma e indica al sistema que desea enviar la información.	12. Valida información introducida por el usuario. Presenta otra página con un resumen de la misma y con las opciones de escribir comentarios y cotizar la reservación (sólo si existe una cuota por el uso del espacio).
13. Puede o no escribir comentarios e indica al sistema que cotiche la reservación por el uso del ó los espacios a utilizar.	14. Valida información introducida (si es que existe) por el usuario. Realiza la cotización en función al tipo de dependencia a la que pertenece el usuario, tiempo de uso y espacio, y la muestra.
15. Revisa la cotización e indica al sistema que desea continuar con el proceso.	16. Genera un número de solicitud y registra la ó las reservaciones realizadas. Envía un correo electrónico al usuario y al administrador notificando la realización de la o las reservaciones. Envía a otra página en donde muestra un resumen de toda la información introducida y

	generada. Este resumen es el comprobante de solicitud del usuario.
17. Imprime su comprobante de solicitud. Sale del sistema cerrando la página de Internet.	
Cursos alternos	
Línea 8:	El tiempo de sesión de usuario ha terminado. Aparecerá una leyenda indicando lo que sucede y lo que se debe de hacer ante esta situación; ó la información no es válida e impide continuar con el proceso.

2.3.9 Realizar reservación con servicios adicionales

Actores:	Usuario TI ó Administrador TI
Casos de uso que incluye:	Autenticar usuario, Realizar reservación.
Propósito:	Reservar uno o más espacios correspondientes a la TI con servicios adicionales (Servicio de cafetería, Equipo audiovisual y Mobiliario)
Resumen:	El Usuario TI ó Administrador TI selecciona la opción de servicios adicionales, el sistema muestra una lista de servicios adicionales. El usuario selecciona los servicios que necesita y el sistema realiza una cotización de los mismos. El Usuario TI ó Administrador TI revisa la cotización e indica al sistema que desea continuar con el proceso.
Curso normal de los eventos	
Acción del actor	Acción del sistema
1. Selecciona la opción agregar servicios adicionales. Indica al sistema que desea continuar con el proceso.	2. Muestra una lista de servicios adicionales disponibles.
3. Selecciona el servicio adicional que necesita. Indica al sistema que desea continuar con el proceso.	4. Realiza una cotización del servicio seleccionado en función del tipo de dependencia a la que pertenece el Usuario TI ó Administrador TI y a características propias del servicio.
5. Indica al sistema que desea continuar con el proceso.	6. Valida información. Informa al Usuario TI ó Administrador TI de este suceso.
7. Lee mensaje del sistema y revisa información.	

Indica al sistema que desea continuar con el proceso (Reservación).	
Cursos alternos	
Línea 6:	El tiempo de sesión de usuario ha terminado. Aparecerá una leyenda indicando lo que sucede y lo que se debe de hacer ante esta situación; ó la información no es válida e impide continuar con el proceso.

2.4 Diagramas de Casos de Uso

En general, un diagrama de casos de uso, es una herramienta gráfica que nos sirve para modelar el comportamiento de un sistema, un subsistema o una clase. Dicho diagrama consiste en un conjunto de casos de uso, actores y sus relaciones.

A continuación se pueden apreciar los diagramas de casos de uso por separado para la reservación de espacios del II y la TI de acuerdo a la descripción de casos de uso que se hizo en la sección anterior para cada entidad respectivamente.

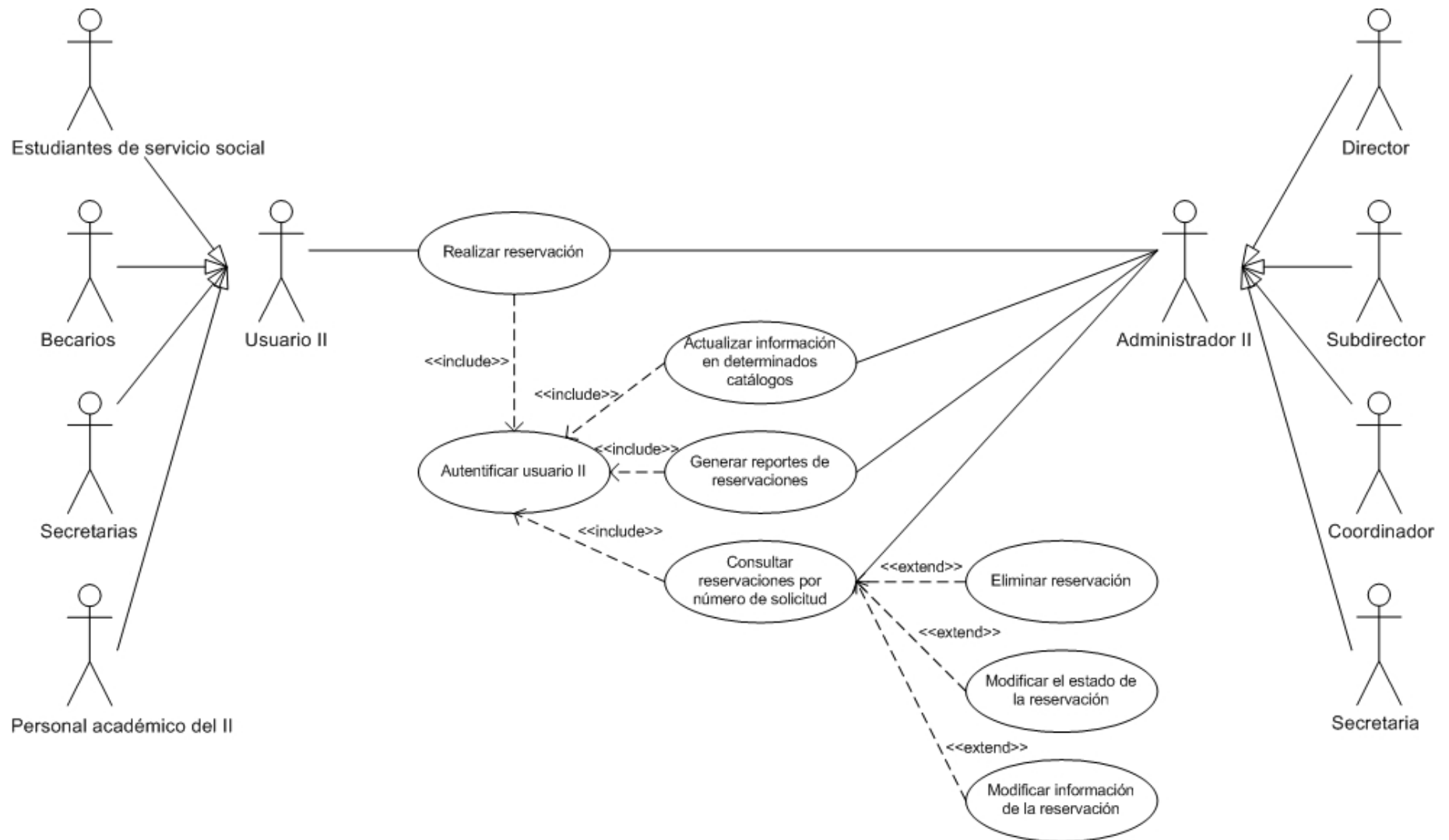


Figura 2.1. Diagrama de Casos de Uso del II.

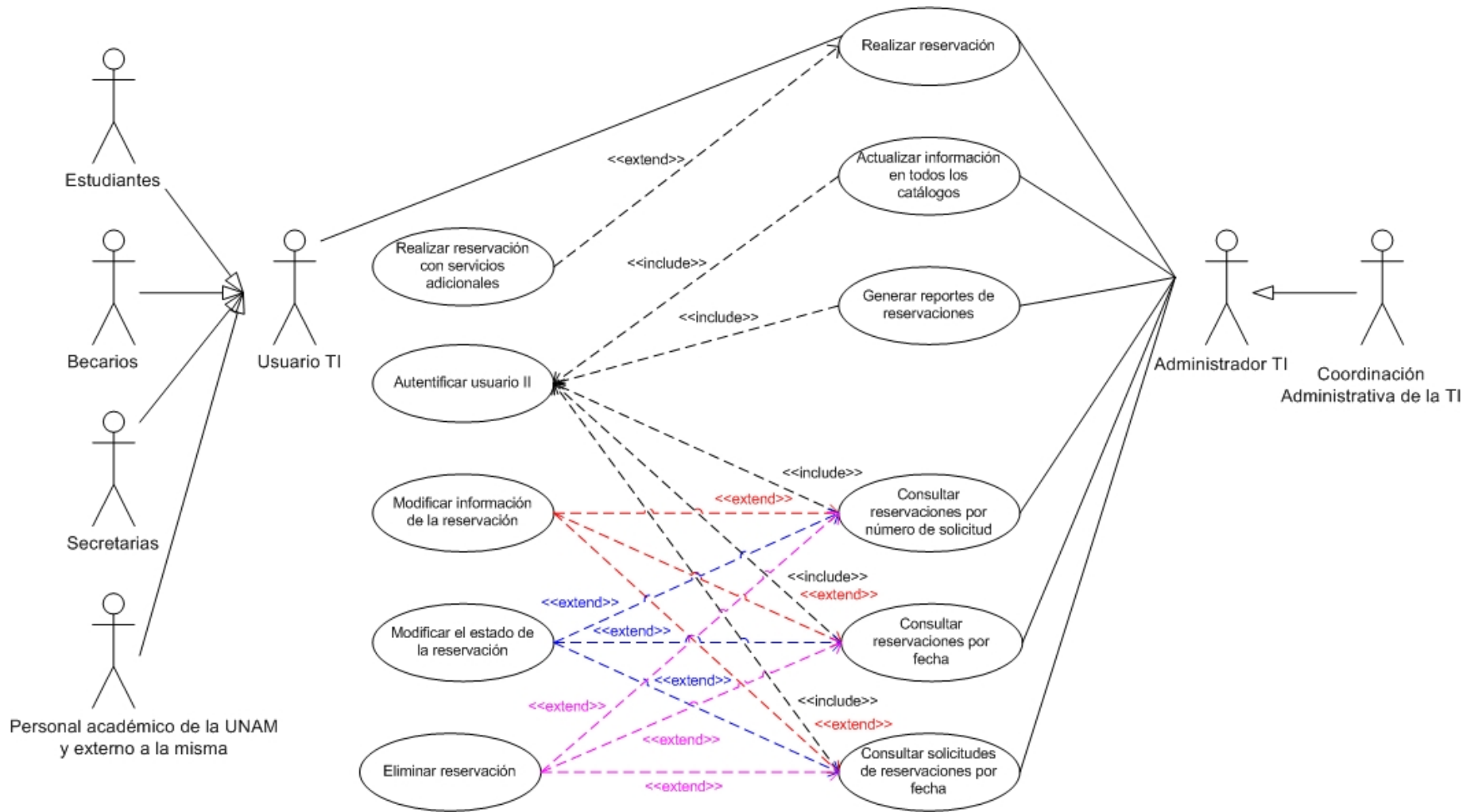


Figura 2.2. Diagrama de Casos de Uso de la TI.

2.5 Requisitos adicionales

Los requisitos adicionales son fundamentalmente requisitos no funcionales que no pueden asociarse a ningún caso de uso en concreto. He aquí los siguientes requisitos adicionales que el SREITI necesita para su funcionamiento.

2.5.1 Requisitos mínimos de plataforma hardware

Servidores (Sólo para la aplicación del SREITI)

- Procesador: Pentium IV, 1 GHz. o equivalente.
- Memoria RAM: 512 MB.
- Espacio disponible en el disco duro: 20GB

Clientes

- Procesador: Pentium II, 450 MHz. o equivalente.
- Memoria RAM: 256 MB
- Conexión a Internet
- Impresora

2.5.2 Requisitos de plataforma software

Servidores

- Sistema Operativo: Windows Server 2003.
- Internet Information Server (IIS) versión 6.0.
- .NET Framework.
- SQL Server 2000.

Clientes

- Sistema Operativo: Windows XP Profesional.
- Explorador de Internet: Internet Explorer 4.0 en adelante o Netscape 5.0 en adelante.

3 ANÁLISIS

En este capítulo, se analizan los requisitos que se describieron en el capítulo anterior, para de esta forma conseguir una comprensión más precisa y una descripción de los mismos que sea fácil de mantener y que nos ayude a estructurar el sistema entero.

El lenguaje que se utiliza en el análisis se basa en un modelo de objetos conceptual cuyo nombre es modelo de análisis. Este modelo nos ayuda a refinar los requisitos y nos permite razonar sobre los aspectos internos del sistema.

En la siguiente tabla, se pueden apreciar las características de este modelo y una comparación de las mismas con el modelo de casos de uso.

Tabla 3.1 Comparación del modelo de casos de uso y el modelo de análisis.

Modelo de casos de uso	Modelo de análisis
Descrito con el lenguaje del cliente.	Descrito con el lenguaje del desarrollador.
Vista externa del sistema.	Vista interna del sistema.
Estructurado por los casos de uso; proporciona la estructura a la vista externa.	Estructurado por clases y paquetes estereotipados; proporciona la vista a la estructura interna.
Utilizado fundamentalmente como contrato entre el cliente y los desarrolladores sobre que debería y qué no debería hacer el sistema.	Utilizado fundamentalmente por los desarrolladores para comprender cómo debería darse forma al sistema, es decir, cómo debería ser diseñado e implementado.
Captura la funcionalidad del sistema, incluida la funcionalidad significativa para la arquitectura.	Esboza como llevar a cabo la funcionalidad dentro del sistema, incluida la funcionalidad significativa para la arquitectura; sirve como una primera aproximación al diseño.
Define casos de uso que se analizarán con más profundidad en el modelo de análisis.	Define realizaciones de casos de uso, y cada una de ellas representa el análisis de un caso de uso del modelo de casos de usos.

3.1 Realizaciones de Casos de Uso – Análisis

Una realización de caso de uso – análisis es una colaboración dentro del modelo de análisis que describe cómo se lleva a cabo y se ejecuta un caso de uso determinado en términos de las clases de análisis y de sus objetos del análisis en interacción.

Para llevar a cabo las realizaciones de casos de uso – análisis se hace uso de los diagramas de colaboración.

3.1.1 Diagramas de colaboración

Un diagrama de colaboración, es un diagrama de interacción que destaca la organización estructural de los objetos que envían y reciben mensajes y cuyo objetivo principal es identificar requisitos y responsabilidades sobre los objetos y no identificar secuencias de interacción detalladas y ordenadas cronológicamente.

El artefacto principal en los diagramas de colaboración son las clases de análisis, las cuales representan una abstracción de una o varias clases y/o subsistemas del diseño del sistema. Son clasificadas en tres estereotipos básicos: de interfaz, de control y de entidad. Estos tres tipos están estandarizados en UML y se utilizan para ayudarnos a distinguir el ámbito de las diferentes clases. A continuación se presenta una descripción de cada una de ellas.

3.1.2 Clases de interfaz

Se utilizan para modelar la interacción entre el sistema y sus actores. Esta interacción a menudo implica recibir (y presentar) información y peticiones de (y hacia) los usuarios y los sistemas externos. A menudo representan abstracciones de ventanas, formularios, paneles, interfaces de comunicaciones, interfaces de impresoras, sensores, terminales y otros.

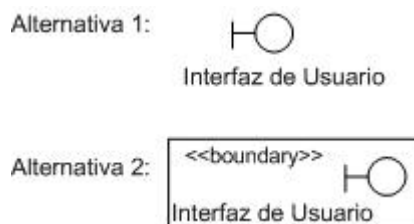


Figura 3.1 Alternativas de representación de las clases de interfaz.

3.1.3 Clases de entidad

Las clases de entidad se utilizan para modelar información que posee una vida larga y que es a menudo persistente. Las clases de entidad modelan la información y el comportamiento asociado de algún fenómeno o concepto, como una persona, un objeto del mundo real, o un suceso del mundo real.



Figura 3.2 Alternativas de representación de las clases de entidad.

3.1.4 Clases de control

Las clases de control representan coordinación, secuencia, transacciones, y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto. Las clases de control también se utilizan para representar derivaciones y cálculos complejos, como la lógica del negocio, que no pueden asociarse con ninguna información concreta, de larga duración, almacenada por el sistema (es decir, una clase de entidad concreta).

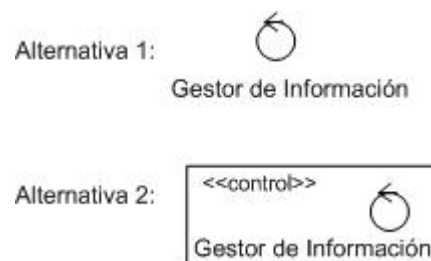


Figura 3.3 Alternativas de representación de las clases de control.

3.1.5 Diagramas de colaboración de las realizaciones de los casos de uso del SREITI

3.1.5.1 Autenticar usuario

En este diagrama, se puede apreciar que el usuario o administrador en cuestión accesa al sistema vía Internet a través de una interfaz de usuario llamada "Inicio"; el sistema solicita que se autentique y dependiendo de la entidad del usuario o administrador, entrará al sistema con una cuenta de Windows o una cuenta anónima.

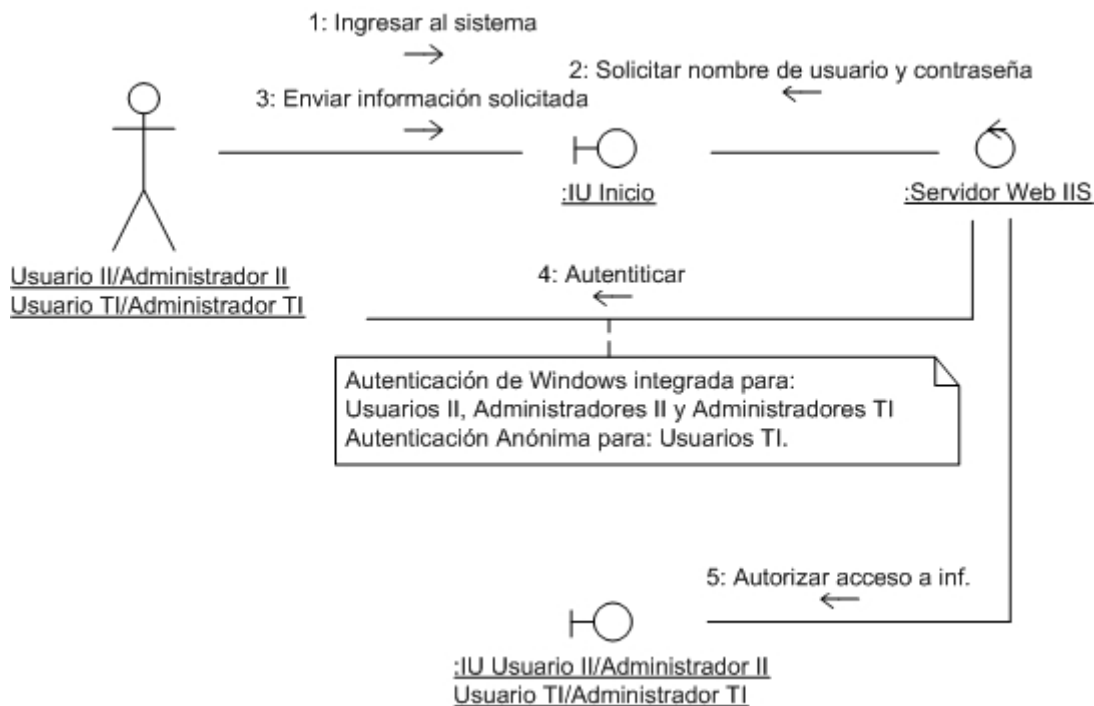


Figura 3.4 Diagrama de colaboración de la realización del caso de uso: Autenticar usuario.

3.1.5.2 Actualizar información en catálogos

Esta sección comprende la actualización de los catálogos, tales como altas bajas y cambios; cabe destacar que no en todos los catálogos se pueden hacer las tres acciones, esto se hace con la finalidad de conservar el histórico de la información, un ejemplo de esto es: Si el administrador del sistema elimina un espacio del catálogo de espacios y existen reservaciones asociadas a ese espacio, el borrado de la información se hará en cascada y al momento de generar su reporte de reservaciones no aparecerán las reservaciones con el espacio eliminado.

3.1.5.3 Actualizar información del II

El administrador del SREII sólo podrá actualizar información de cuotas de espacio, destinatarios de correo electrónico, espacios e imágenes asociadas a los mismos, esto es porque el SREII no ofrece servicios adicionales de equipo adicional y cafetería.

El siguiente diagrama ilustra el proceso que en general el administrador debe seguir para poder actualizar información de cada catálogo, el primer paso es entrar a la interfaz de usuario correspondiente al catálogo que se quiere actualizar, una vez que se está ahí el administrador modifica, elimina o ingresa información, el sistema hace una validación de la misma y si es válida entonces es actualizada.

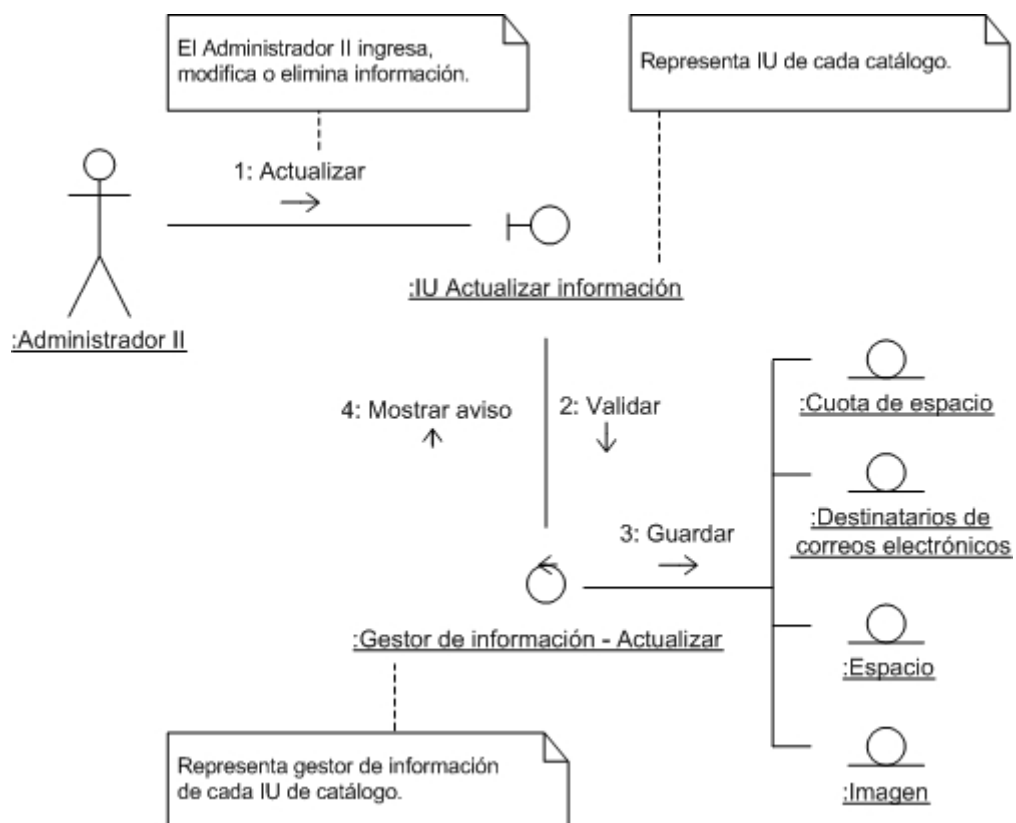


Figura 3.5 Diagrama de colaboración de la realización del caso de uso: Actualizar información en catálogos – Información del II.

3.1.5.4 Actualizar información de la TI

Si hacemos una comparación de este diagrama con el anterior, podemos observar que lo único que cambia es el número de clases de entidad, esto es debido a los requerimientos que tiene la TI en su sistema.

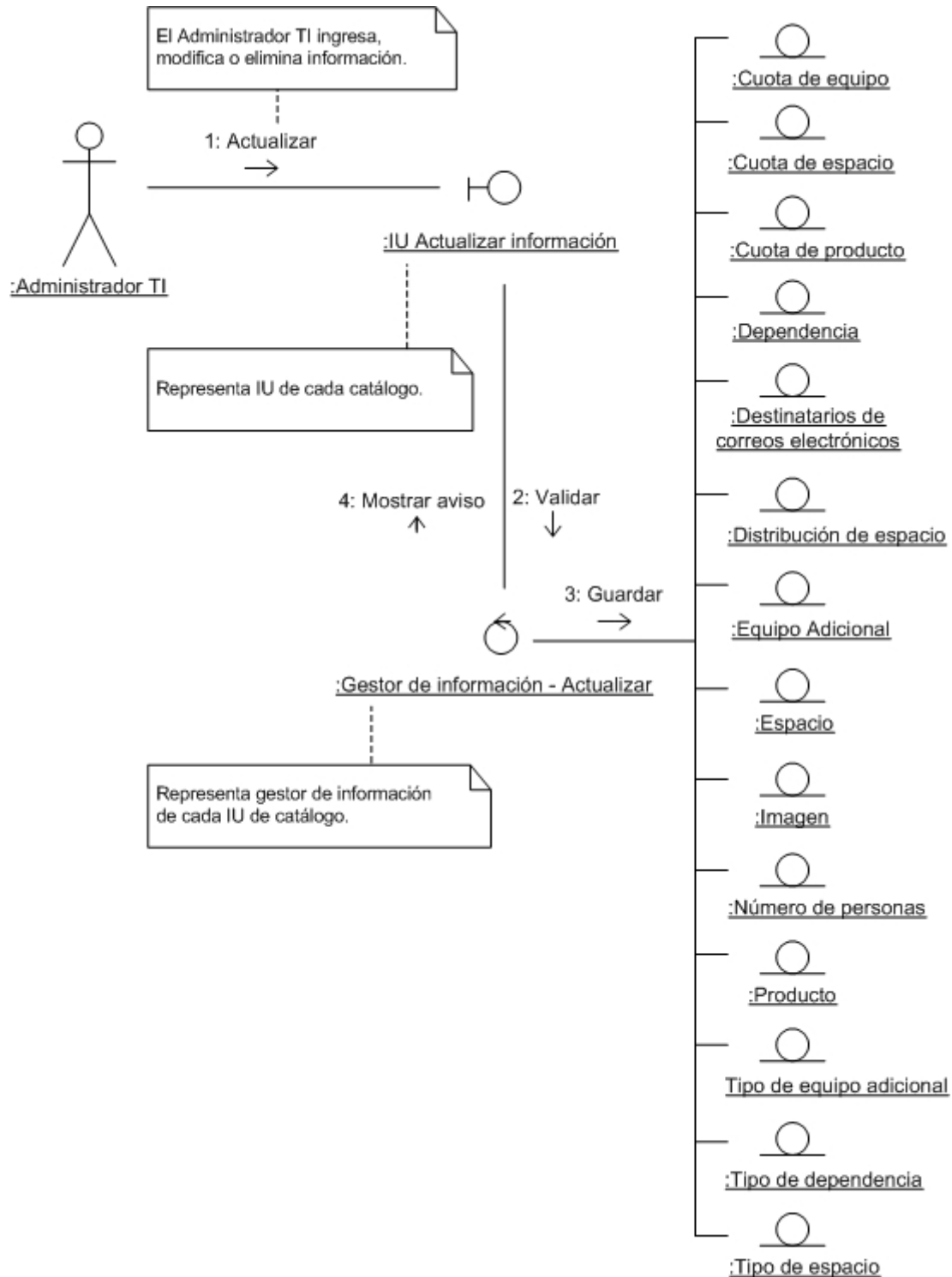


Figura 3.6 Diagrama de colaboración de la realización del caso de uso: Actualizar información en catálogos – Información de la TI.

3.1.5.5 Activar o desactivar el estado de servicios adicionales

Este diagrama nos permite observar cómo es que se activa o desactiva un servicio adicional, iniciando de la siguiente forma: el administrador entra al módulo de realizar modificaciones a través de la interfaz de usuario "Modificar estado de servicios adicionales" y selecciona de un catálogo el servicio adicional al cual desea modificarle el estado, el sistema valida la información y si es válida la actualiza.

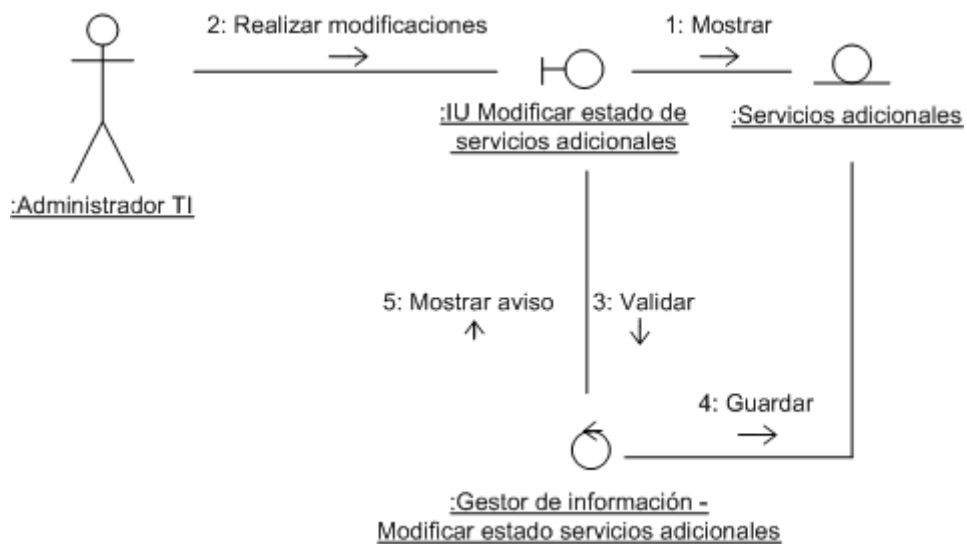


Figura 3.7 Diagrama de colaboración de la realización del caso de uso: Activar ó desactivar el estado de servicios adicionales.

3.1.5.6 Confirmar o cancelar la reservación

Para confirmar o cancelar una reservación, el administrador debe consultar una lista de las reservaciones existentes en el sistema a través de la interfaz de usuario "Consultar reservaciones", posteriormente seleccionar la reservación a la cual desea modificar el estado, esto lo hace en una interfaz de usuario llamada "Modificar características de la reservación"; posteriormente el administrador modifica el estado, al hacerlo el sistema valida la información y si es válida la actualiza.

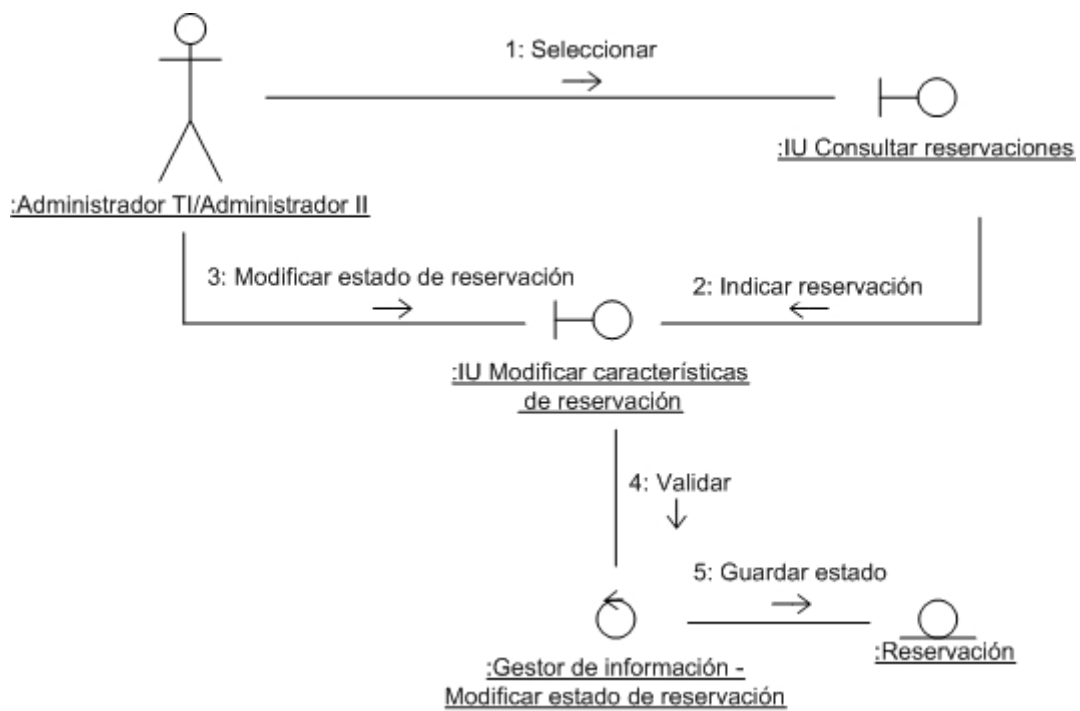


Figura 3.8 Diagrama de colaboración de la realización del caso de uso: Confirmar ó cancelar la reservación.

3.1.5.7 Consultar reservasiones por fecha ó número de solicitud ó por fecha de evento

El administrador puede consultar por fecha ó número de solicitud ó por fecha de evento, las reservasiones que existen registradas en el sistema. El siguiente diagrama ilustra la forma de hacerlo, esto es a través de la interfaz de usuario "Consultar reservasiones" aquí el administrador selecciona la forma en que quiere realizar su búsqueda, una vez seleccionada le aparecerá un icono de un calendario en donde podrá seleccionar la(s) fecha(s) o bien una caja de texto en donde podrá escribir el número de solicitud; una vez proporcionadas las entradas, la salida será una lista de reservasiones si es que existen.

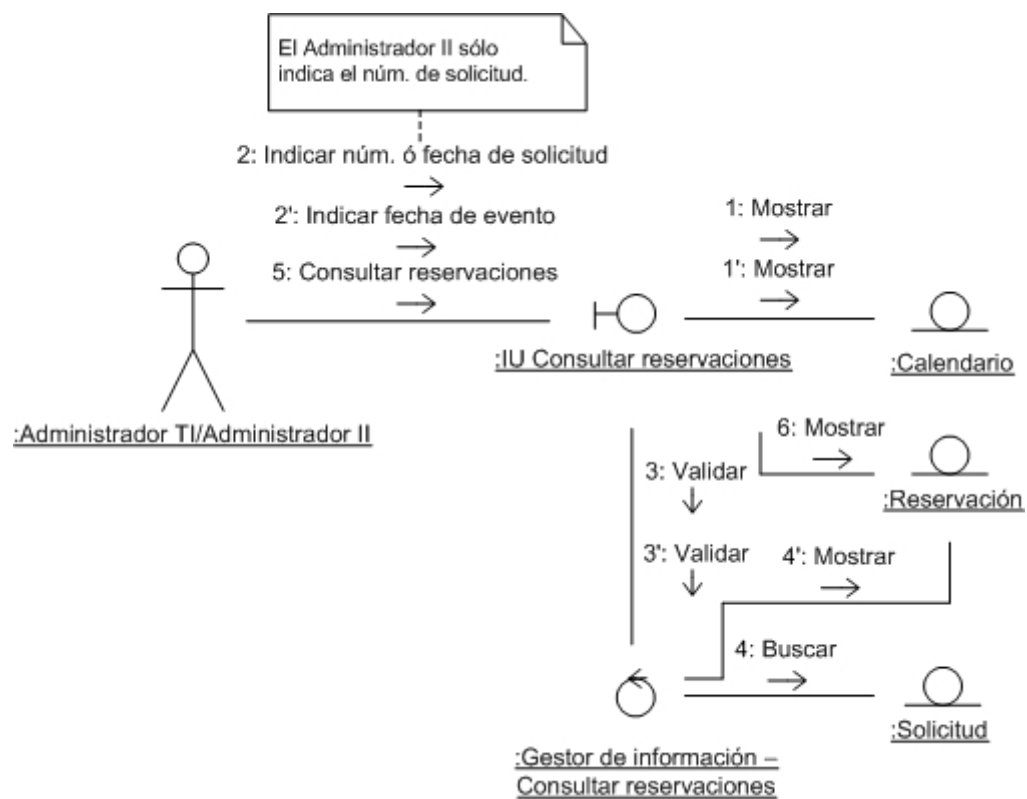


Figura 3.9 Diagrama de colaboración de la realización del caso de uso: Consultar reservasiones por fecha ó número de solicitud ó por fecha de evento.

3.1.5.8 Eliminar reservaciones

Para eliminar reservaciones, el administrador debe obtener una lista de reservaciones existentes en el sistema a través de la interfaz de usuario "Consultar reservaciones" una vez que haya entrado a esta interfaz, le aparecerá una opción para eliminar la reservación y esta última lo llevará a la interfaz "Eliminar reservación", aquí selecciona la opción eliminar y el sistema muestra un mensaje indicando que se ha eliminado la reservación.

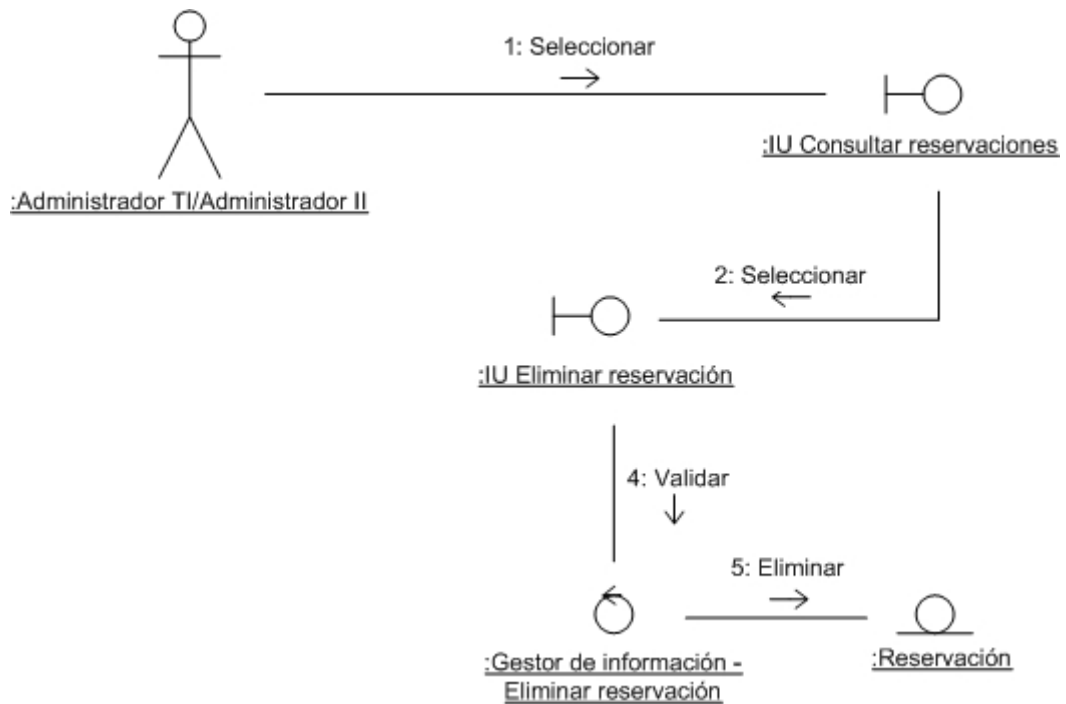


Figura 3.10 Diagrama de colaboración de la realización del caso de uso: Eliminar reservaciones.

3.1.5.9 Generar reportes de reservaciones

Para generar reportes el administrador selecciona el año, mes y el estado de la reservación a través de la interfaz de usuario "Reportes", el sistema valida la información y genera un reporte de las reservaciones con las opciones seleccionadas.

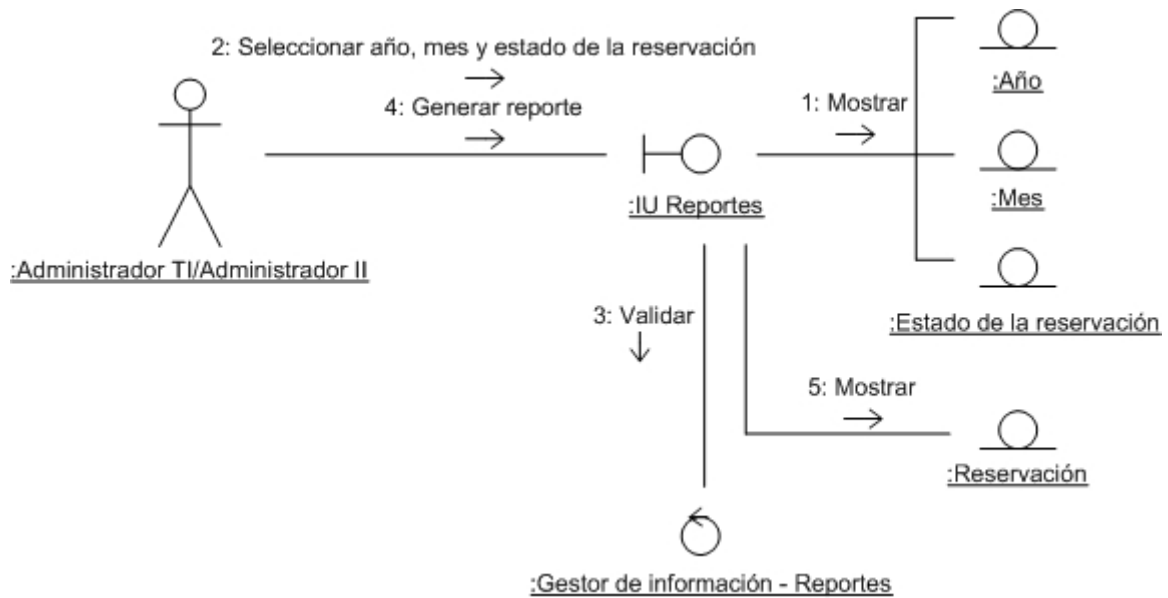


Figura 3.11 Diagrama de colaboración de la realización del caso de uso: Generar reportes de reservaciones.

3.1.5.10 Modificar información de la reservación

El administrador puede modificar la información de las reservaciones al consultar las que existen en el sistema con la interfaz de usuario “Consultar reservaciones” y posteriormente indicando la reservación a modificar, esto último lo llevará a la interfaz de usuario “Modificar características de reservación” aquí se realizan las modificaciones deseadas y si la información es válida, se registra en el sistema.

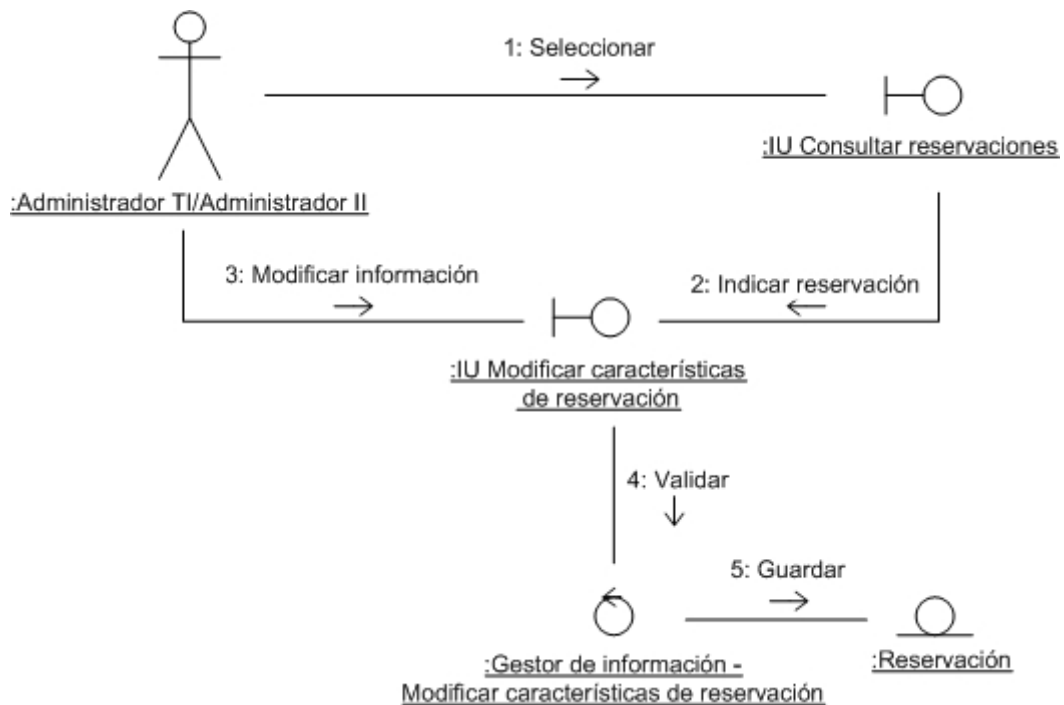


Figura 3.12 Diagrama de colaboración de la realización del caso de uso: Modificar información de la reservación.

3.1.5.11 Realizar reservación

El siguiente diagrama ilustra el proceso que usuarios y administradores deben seguir para realizar una reservación. Este proceso se lleva a cabo a través de las interfaces de usuario: “Realizar reservación”, “Solicitud”, “Cotizar reservación(es)”, y “Resumen”.

En “Realizar reservación” el usuario selecciona las características de su reservación tales como fechas, horarios, espacios, nombre de evento y número de personas que asistirán a el mismo, el sistema valida la información y si existe disponibilidad entonces permite continuar con el proceso llevando a el usuario a la interfaz “Solicitud” aquí el usuario llena un formulario con información personal, para continuar debe seleccionar la opción que lo lleve a “Cotizar reservación(es)” aquí el usuario puede cotizar su reservación y realizarla, eliminarla o bien agregar más reservaciones a su solicitud si el usuario está conforme con el costo de su(s) reservación(es) entonces debe elegir la opción de reservar, al hacerlo, la(s) reservación(es) se registra(n) en el sistema, éste último genera un número de solicitud y envía un correo electrónico a los administradores del sistema para notificar el registro de una solicitud de reservación(es) en el sistema. Finalmente el sistema lleva al usuario a la interfaz “Resumen” aquí se muestra el resumen de la información capturada y en donde se termina el proceso.

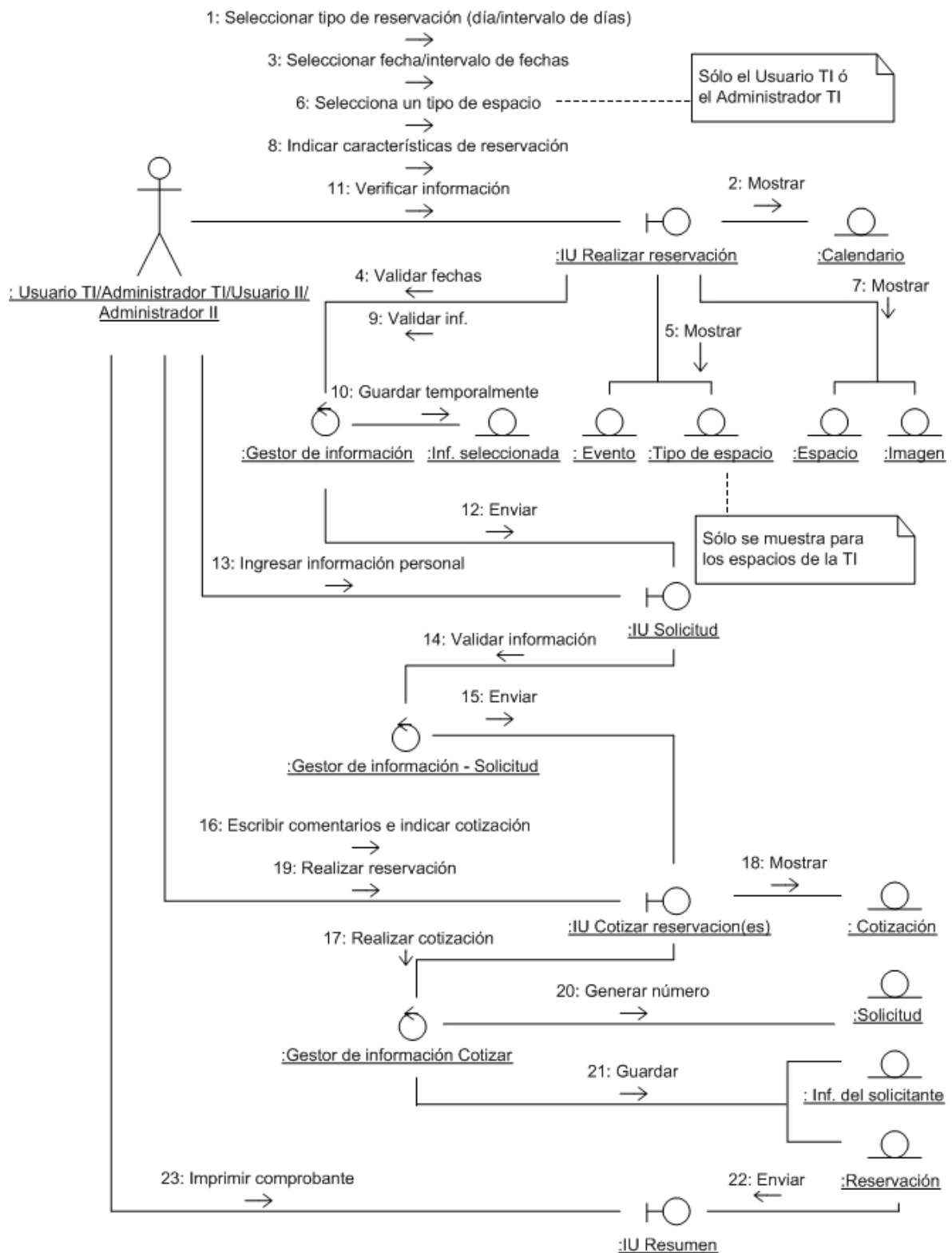


Figura 3.13 Diagrama de colaboración de la realización del caso de uso: Realizar reservación.

3.1.5.12 Realizar reservación con servicios adicionales

En este diagrama se puede observar como se realiza una reservación con servicios adicionales, el proceso de reservación (diagrama anterior) sólo se ve alterado cuando el usuario selecciona la opción de servicios adicionales en la interfaz "Cotizar reservación", la cual lo lleva a otra interfaz llamada "Servicios adicionales", aquí se selecciona el tipo de servicio que se desea tener, los servicios pueden ser de cafetería y/o de equipo adicional, una vez seleccionado el servicio, el sistema realiza una cotización del mismo y el usuario decide si lo registra o bien lo puede eliminar.

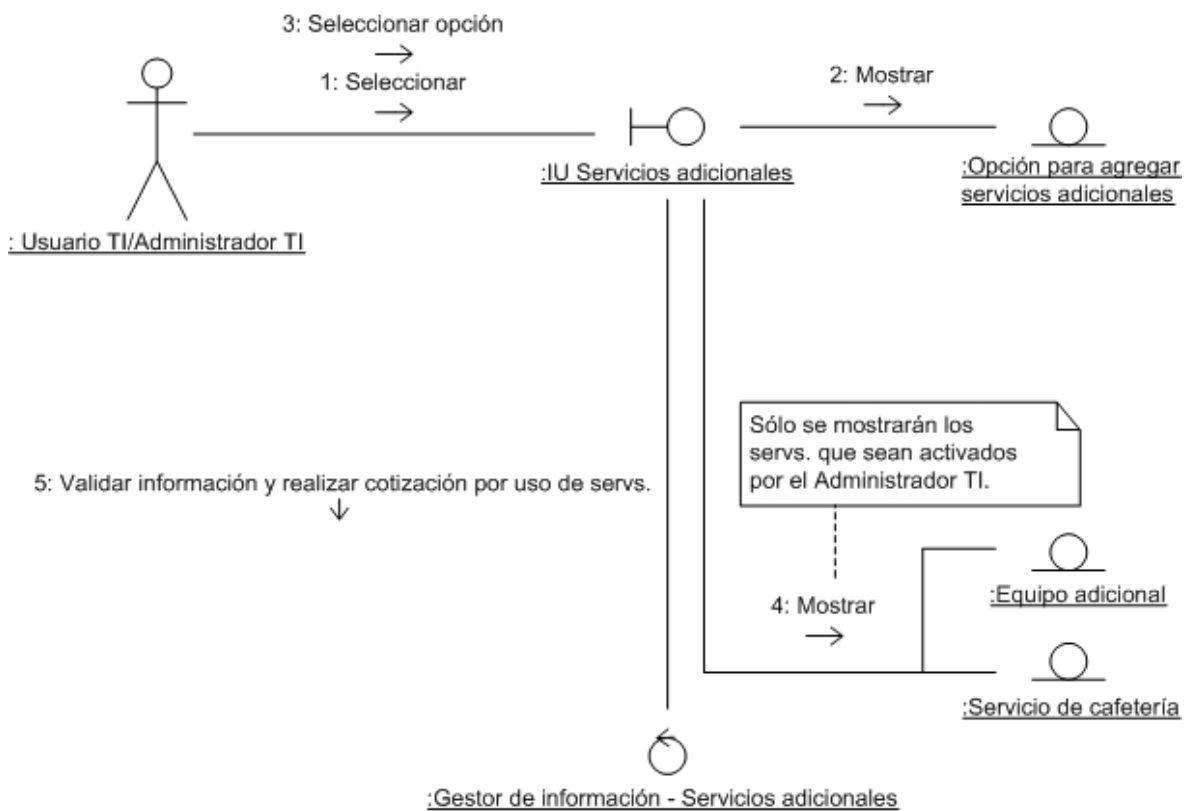


Figura 3.14 Diagrama de colaboración de la realización del caso de uso: Realizar reservación con servicios adicionales.

Estos fueron todos los diagramas que se consideraron necesarios para la realización de casos de uso – análisis del SREITI.

Para ver información más detallada de cada caso de uso, consultar el capítulo 2.

3.2 Paquetes del análisis

Los paquetes del análisis proporcionan un medio para organizar los artefactos del modelo de análisis en piezas manejables. Un paquete de análisis puede constar de clases de análisis, de realizaciones de casos de uso, y de otros paquetes del análisis (recursivamente), además pueden representar una separación de intereses de análisis. Esto último es precisamente lo que se pretende hacer con el SREITI, agrupar las funcionalidades para organizar y tener un mejor manejo de la información.

De acuerdo a las clases de análisis obtenidas en los diagramas de colaboración, éstas pueden ser agrupadas en los siguientes paquetes:

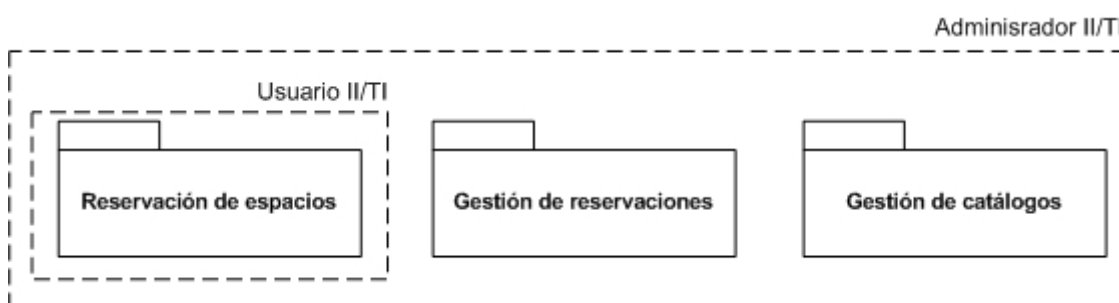


Figura 3.15 Paquetes del SREITI.

Agrupación de clases de interfaz en los paquetes obtenidos:

Reservación de espacios: Realizar reservación, Solicitud, Cotizar reservación, Resumen y todas las clases de entidad y control con las que se relacionan.

Gestión de reservas: Modificar estado de servicios adicionales, Modificar características de la reservación, Consultar reservas, Eliminar reservas, Reportes y todas las clases de entidad y control con las que se relacionan.

Gestión de catálogos: Actualizar información y todas las clases de entidad y control con las que se relaciona.

4 DISEÑO

El propósito fundamental del diseño, es adecuar los requisitos funcionales y los no funcionales con las actuales tecnologías de información con las que cuenta el Instituto y la Torre de Ingeniería. Para lograr esta tarea, el diseño será dividido en tres etapas: diseño conceptual, diseño lógico y diseño físico.

4.1 Diseño Conceptual

En el diseño conceptual se modelan los requerimientos del usuario, a través de los casos de uso; si bien recordamos, este diseño ya fue realizado implícitamente en el Capítulo 2, sin embargo se hace referencia al mismo para poder distinguir esta etapa del diseño.

4.2 Diseño Lógico

En esta etapa del diseño se trabaja con un modelo de objetos llamado modelo de diseño, el cual describe la realización física de los casos de uso. En el modelo de diseño, los casos de uso son realizados por las clases de diseño y sus objetos. Esto se representa por colaboraciones en el modelo de diseño y denota realización de casos de uso – diseño. Adviértase que realización de caso de uso – diseño es diferente de la realización de caso de uso – análisis. Lo anterior describe cómo se realiza un caso de uso en términos de interacción entre objetos del diseño, mientras que lo último describe cómo se realiza un caso de uso en términos de interacción entre objetos del análisis.

En la siguiente tabla se pueden apreciar las características de este modelo y una comparación de las mismas con el modelo de análisis.

Tabla 4.1 Comparación del modelo de análisis y el modelo de diseño.

Modelo de análisis	Modelo de diseño
Modelo conceptual, porque es una abstracción del sistema y permite aspectos de la implementación.	Modelo físico porque es un plano de la implementación.
Tres estereotipos conceptuales sobre las clases: Control, Entidad e Interfaz.	Cualquier número de estereotipos (físicos) sobre las clases, dependiendo del lenguaje de implementación.
No muy centrado en la secuencia.	Muy centrado en la secuencia.
Bosquejo del diseño del sistema.	Manifiesto del diseño del sistema.

4.2.1 Realizaciones de casos de uso – diseño

Una realización de caso de uso – diseño tiene una descripción de flujo de eventos textual, diagramas de clases que muestran sus clases de diseño participantes y diagramas de interacción que muestran la realización de un flujo o escenario concreto de un caso de uso en términos de interacción entre objetos del diseño.

La primer realización de caso de uso - diseño que se abordará es el diagrama de clases.

4.2.1.1 Diagrama de Clases

El diagrama de clases describe los tipos de objetos que hay en el sistema y las relaciones estáticas que existen entre ellos. Hay dos tipos de relaciones estáticas:

- Asociaciones (Una solicitud tiene una o varias reservaciones).
- Subtipos (Torre de Ingeniería es una organización).

Los diagramas de clases también muestran los atributos y operaciones de una clase y las restricciones a que se ven sujetos, según la forma en que se concentren los objetos.

A continuación, se puede observar el diagrama de clases del SREITI, en él se incluyen las principales clases y las relaciones que existen entre las mismas.

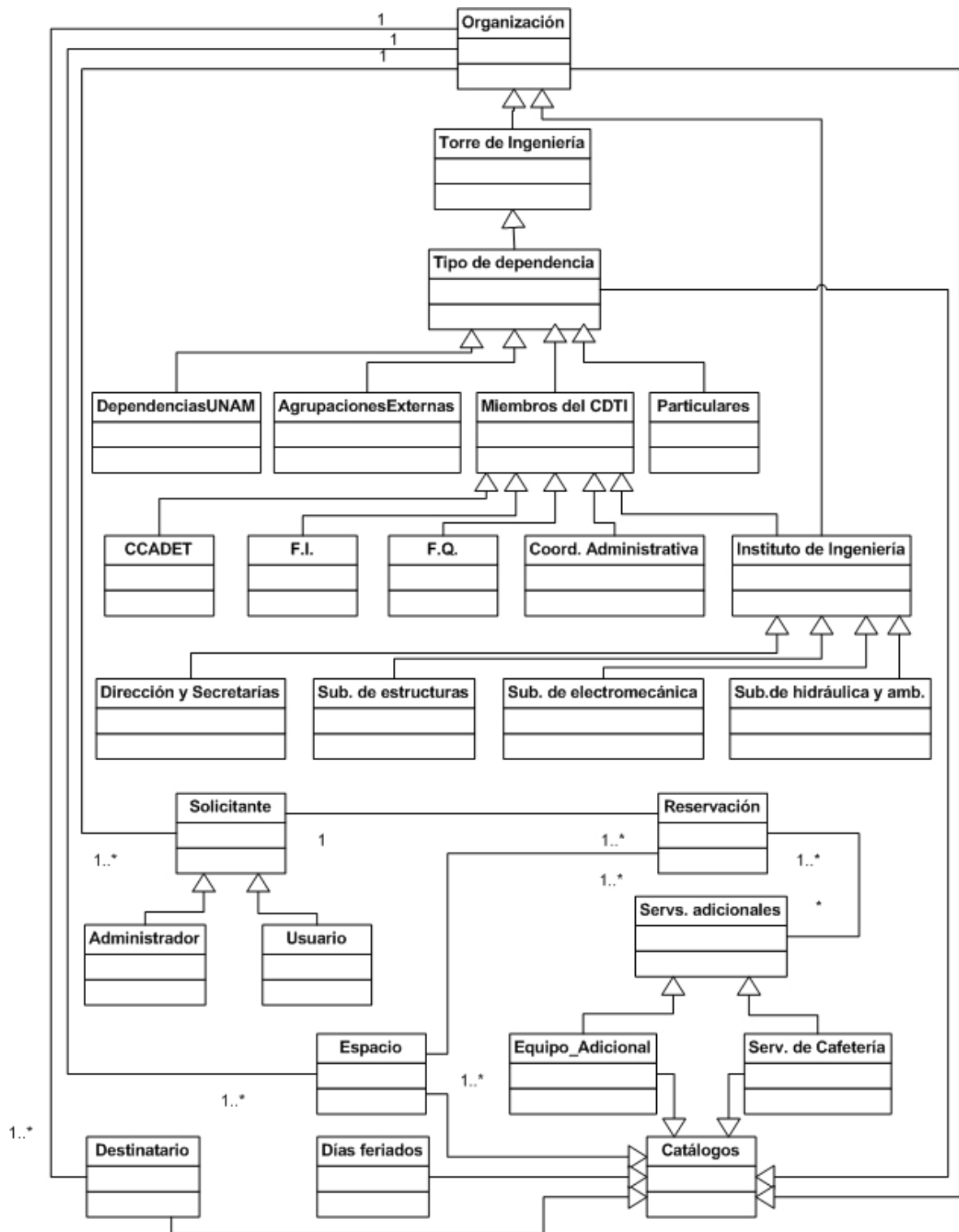


Figura 4.1 Diagrama de clases del SREITI.

4.2.1.1.1 Atributos y métodos de las principales clases entidad

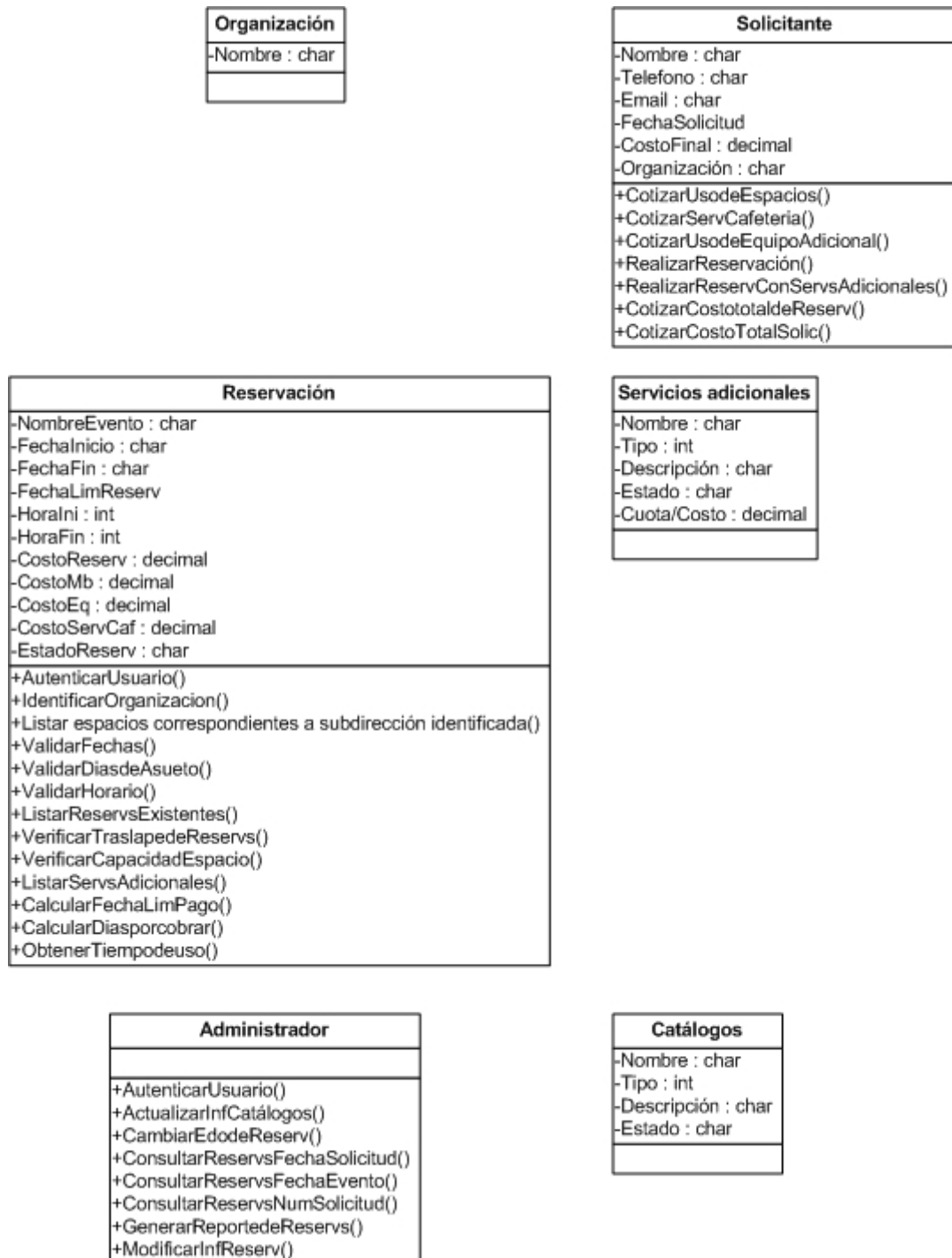


Figura 4.2 Atributos y métodos de las principales clases entidad del SREITI.

4.2.1.2 Diagramas de Secuencia

Un diagrama de secuencia es un diagrama que muestra las interacciones entre los objetos organizadas en una secuencia temporal. En particular muestra los objetos participantes en la interacción y la secuencia de mensajes intercambiados. A diferencia de los diagramas de colaboración, los diagramas de secuencia incluyen secuencias temporales pero no incluyen las relaciones estructurales entre objetos.

A continuación se muestran cada uno de los diagramas de secuencia que conforman el SREITI.

4.2.1.2.1 Caso de uso: Autenticar usuario

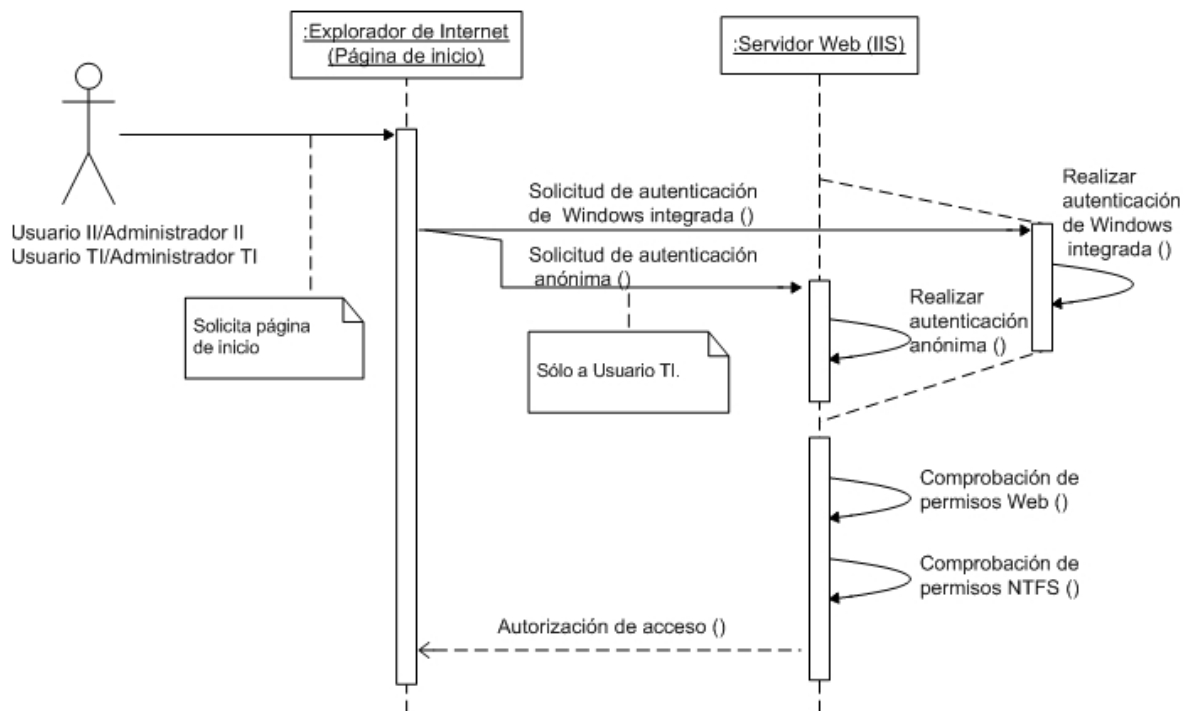


Figura 4.3 Diagrama de secuencias del caso de uso: Autenticar usuario

4.2.1.2.2 Caso de uso: Realizar reservación

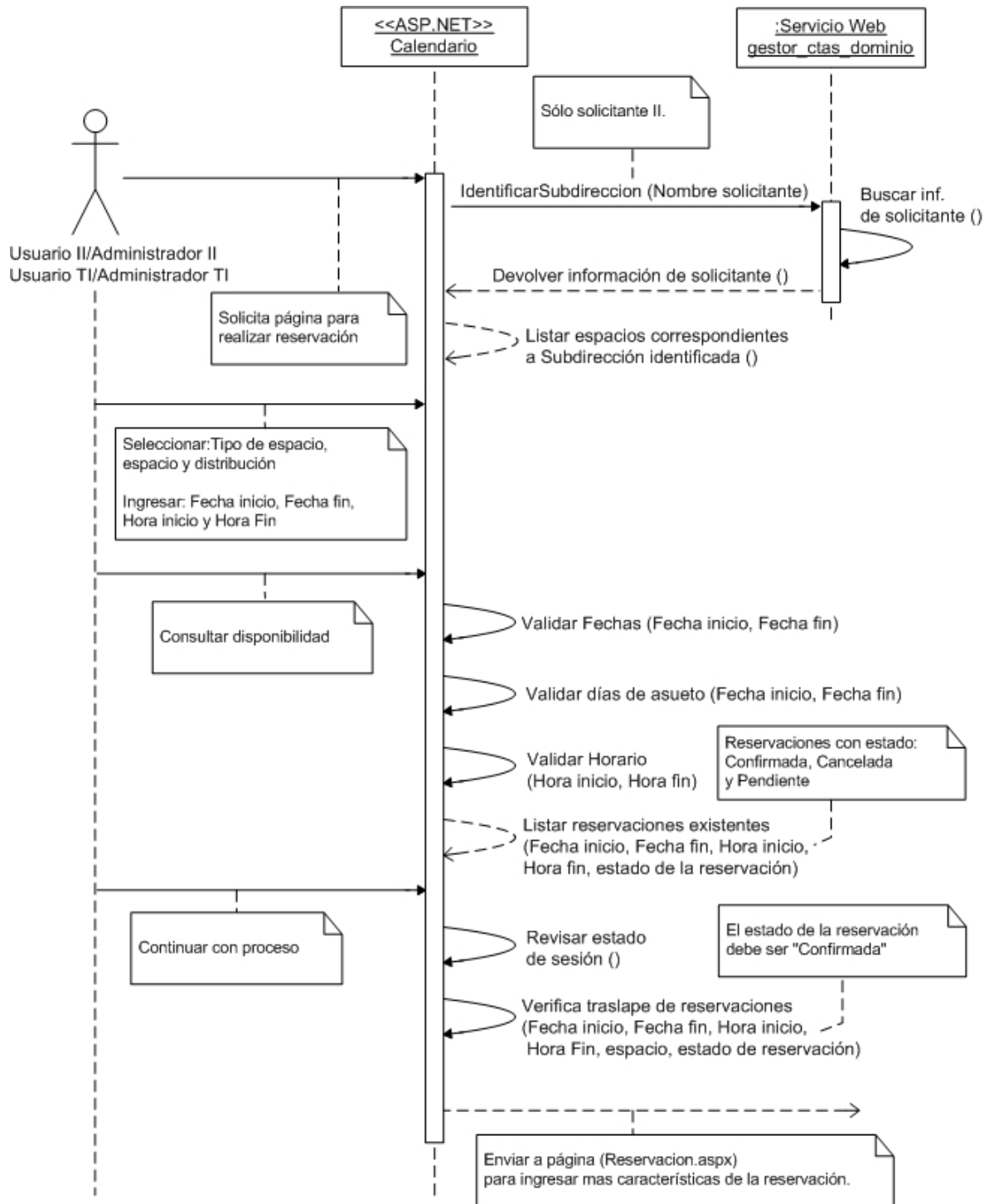


Figura 4.4 Diagrama de secuencias del caso de uso: Realizar reservación

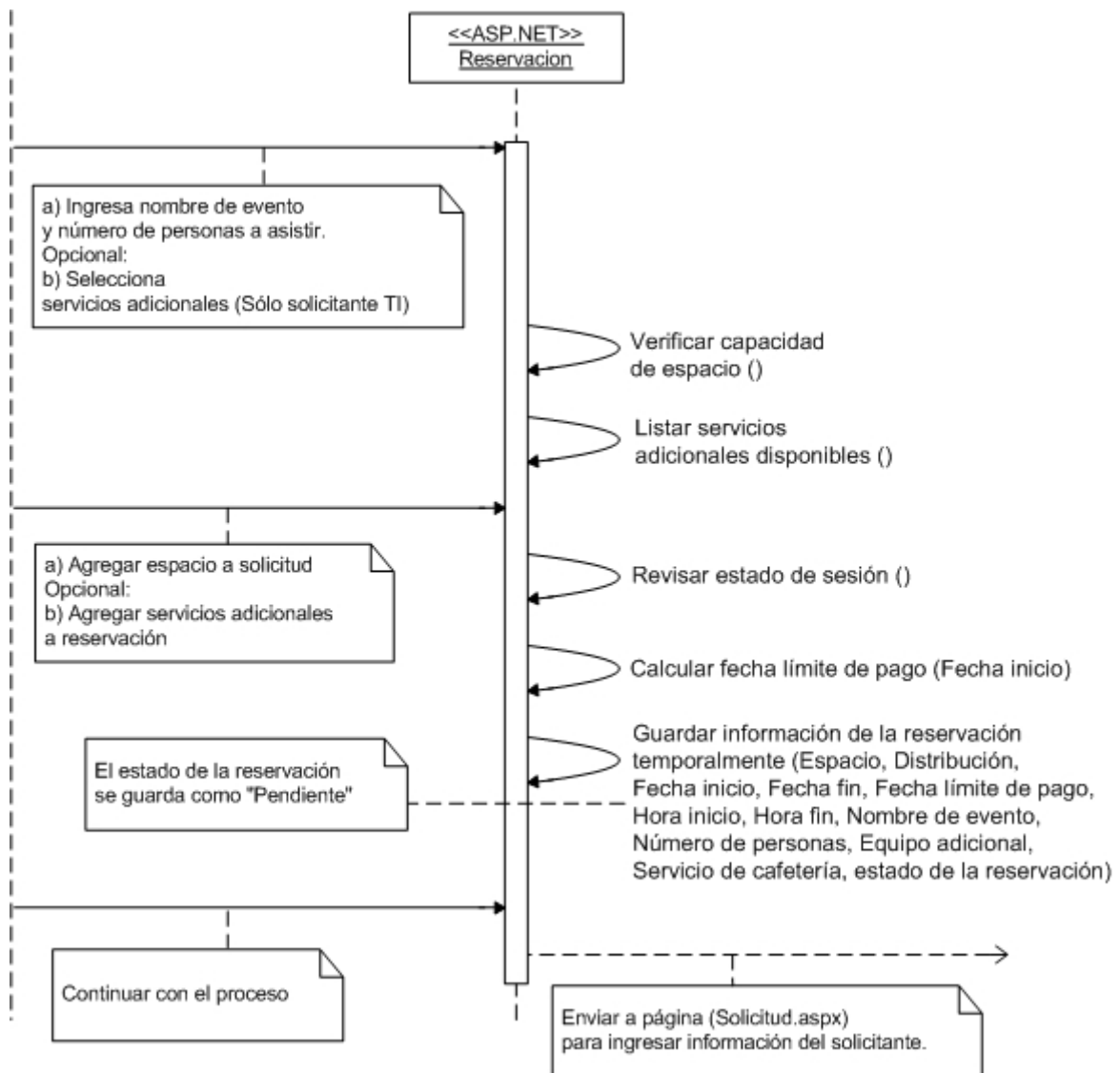


Figura 4.5 Continuación del Diagrama de secuencias del caso de uso: Realizar reservación

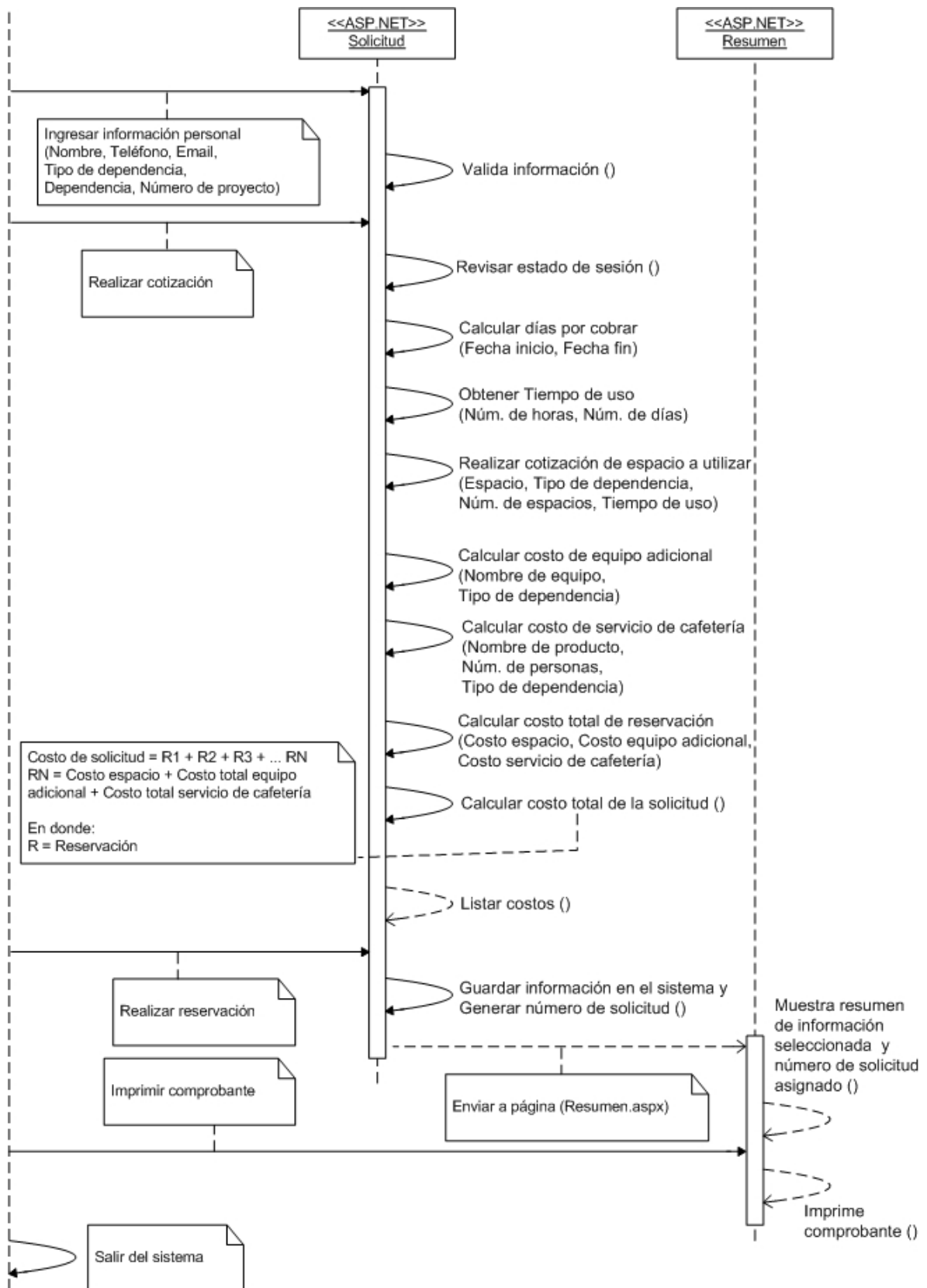


Figura 4.6 Continuación del Diagrama de secuencias del caso de uso: Realizar reservación

4.2.1.2.3 Caso de uso: Consultar reservas por fecha o número de solicitud ó por fecha de evento

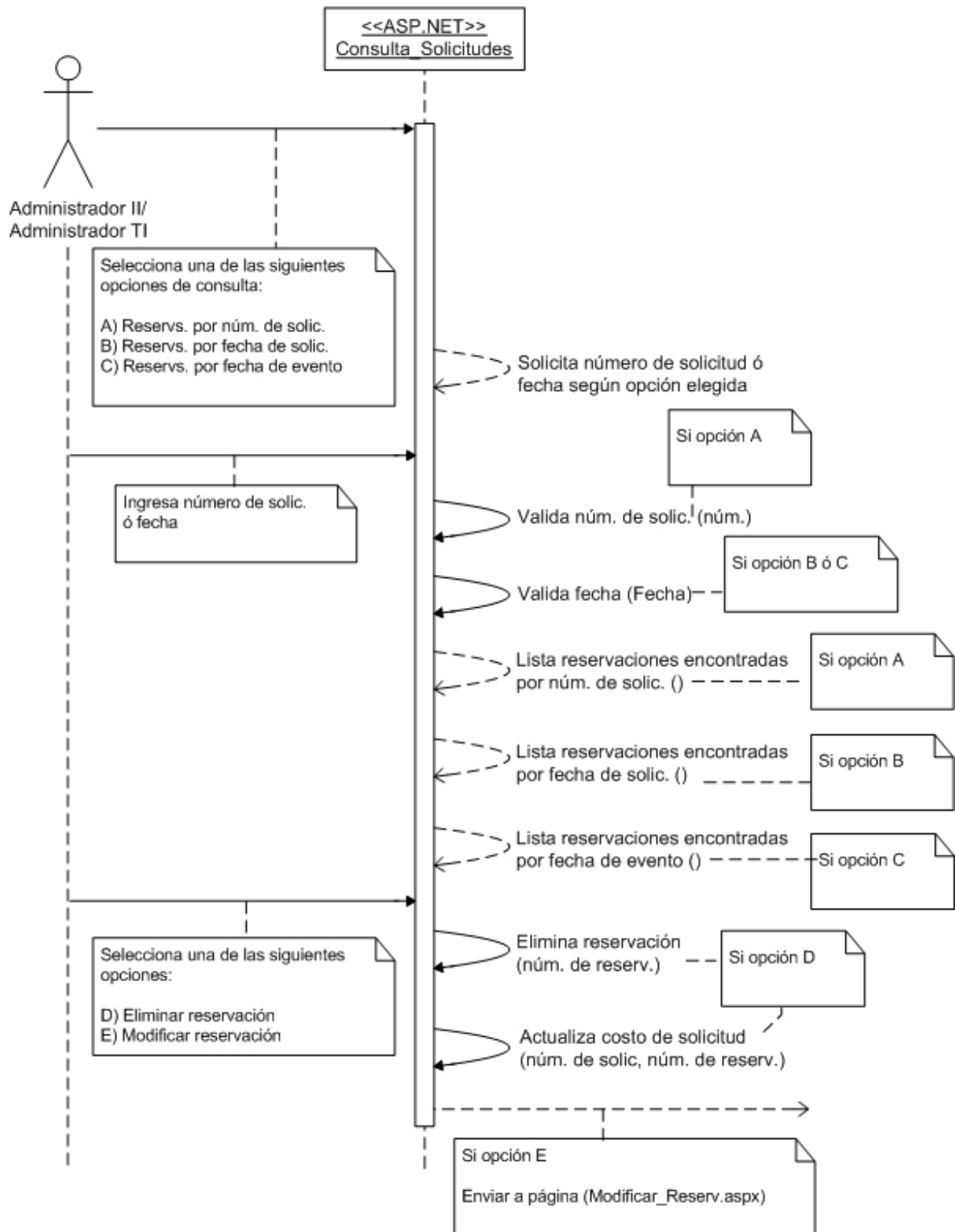


Figura 4.7 Diagrama de secuencias del caso de uso: Consultar reservas por fecha o número de solicitud ó por fecha de evento



Figura 4.8 Continuación del Diagrama de secuencias del caso de uso: Consultar reservaciones por fecha o número de solicitud ó por fecha de evento

4.2.1.2.4 Caso de uso: Generar reportes de reservaciones

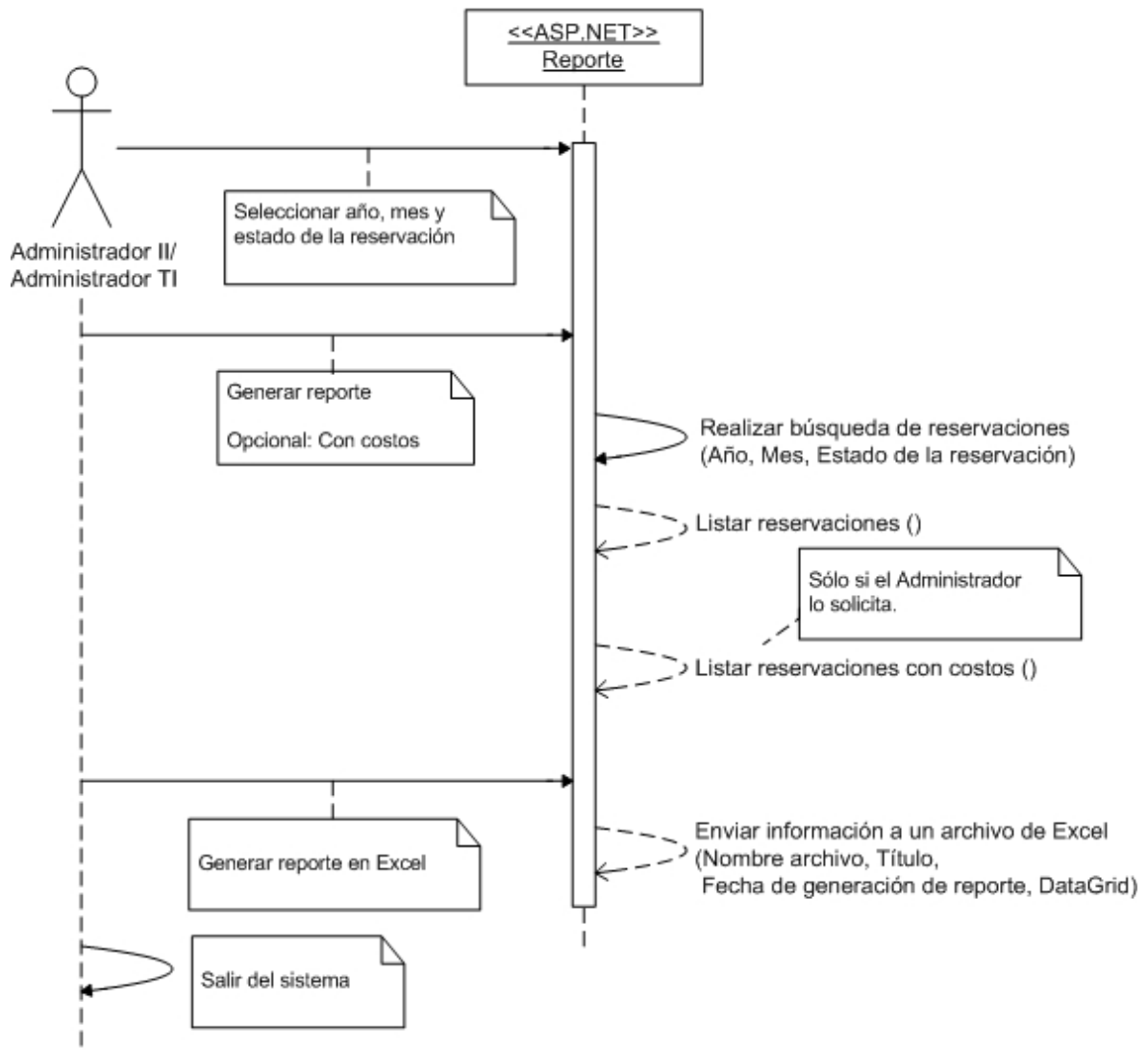


Figura 4.9 Diagrama de secuencias del caso de uso: Generar reportes de reservaciones.

4.2.1.2.5 Caso de uso: Actualizar información en catálogos

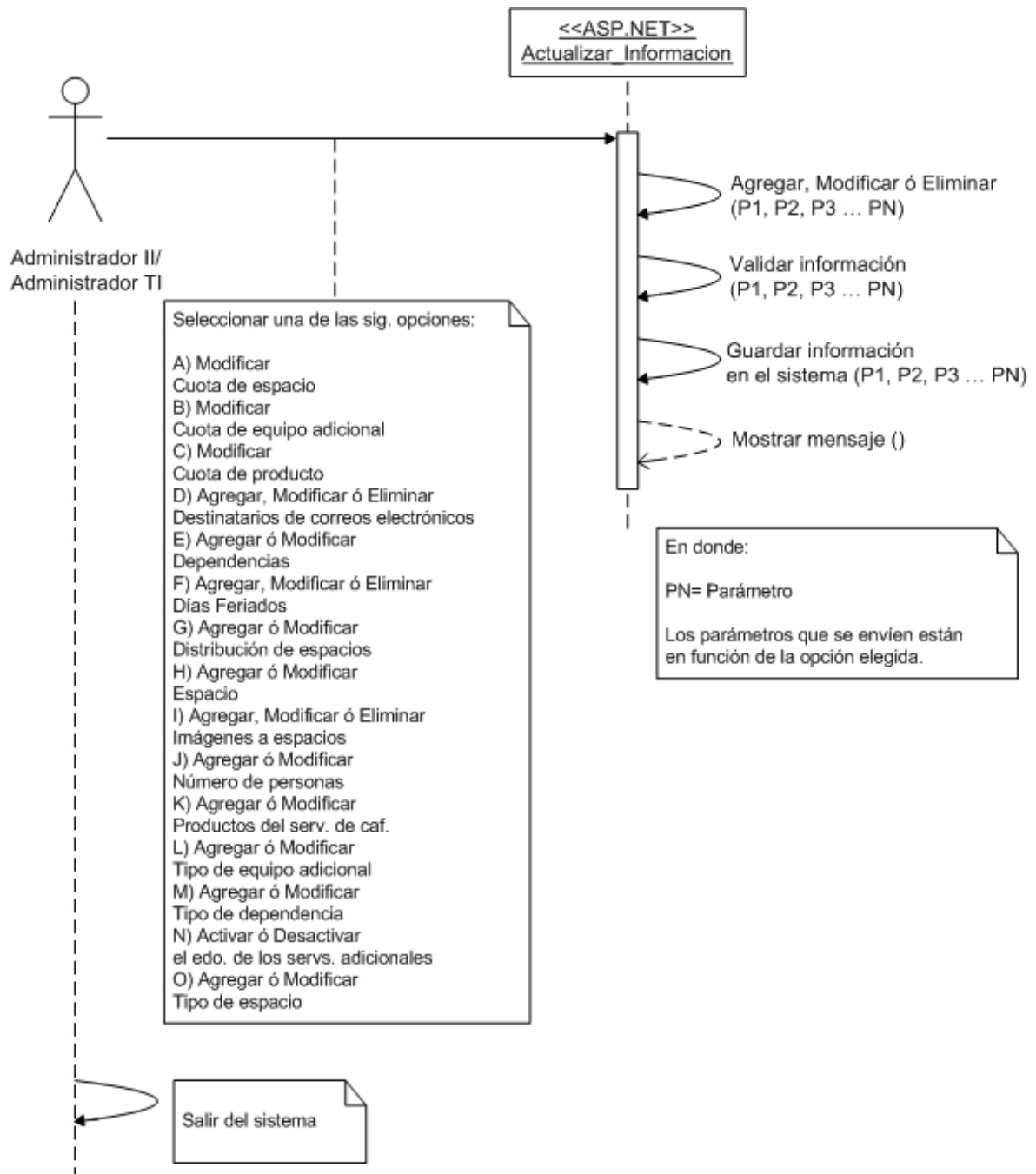


Figura 4.10 Diagrama de secuencias del caso de uso: Actualizar información en catálogos.

4.2.1.3 Diagrama de actividades

Un diagrama de actividades es fundamentalmente un diagrama de flujo que muestra el flujo de control entre actividades, estas últimas representan un paso en el flujo de trabajo o la ejecución de una operación. Puede contener bifurcaciones, así como divisiones de control en hilos concurrentes. Los hilos concurrentes representan actividades que se pueden realizar concurrentemente por los diversos objetos o personas. Las actividades concurrentes se pueden realizar simultáneamente o en cualquier orden. Un diagrama de actividades es como un organigrama tradicional, excepto que permite el control de concurrencia además del control secuencial.

Este tipo de diagramas sirven para entender el comportamiento de alto nivel de la ejecución de un sistema, sin profundizar en los detalles internos de los mensajes. Los parámetros de entrada y salida de una acción se pueden mostrar usando las relaciones de flujo que conectan la acción y un estado de flujo de objeto.

La diferencia que existe entre un diagrama de actividades y un diagrama de interacción es que este último muestra objetos que pasan mensajes, en cambio un diagrama de actividades muestra las operaciones que se pasan entre los objetos.

A continuación se puede observar el diagrama de actividades del SREITI, en él, las actividades que son realizadas por el usuario y administrador son separadas por una línea punteada para poder distinguirlas.

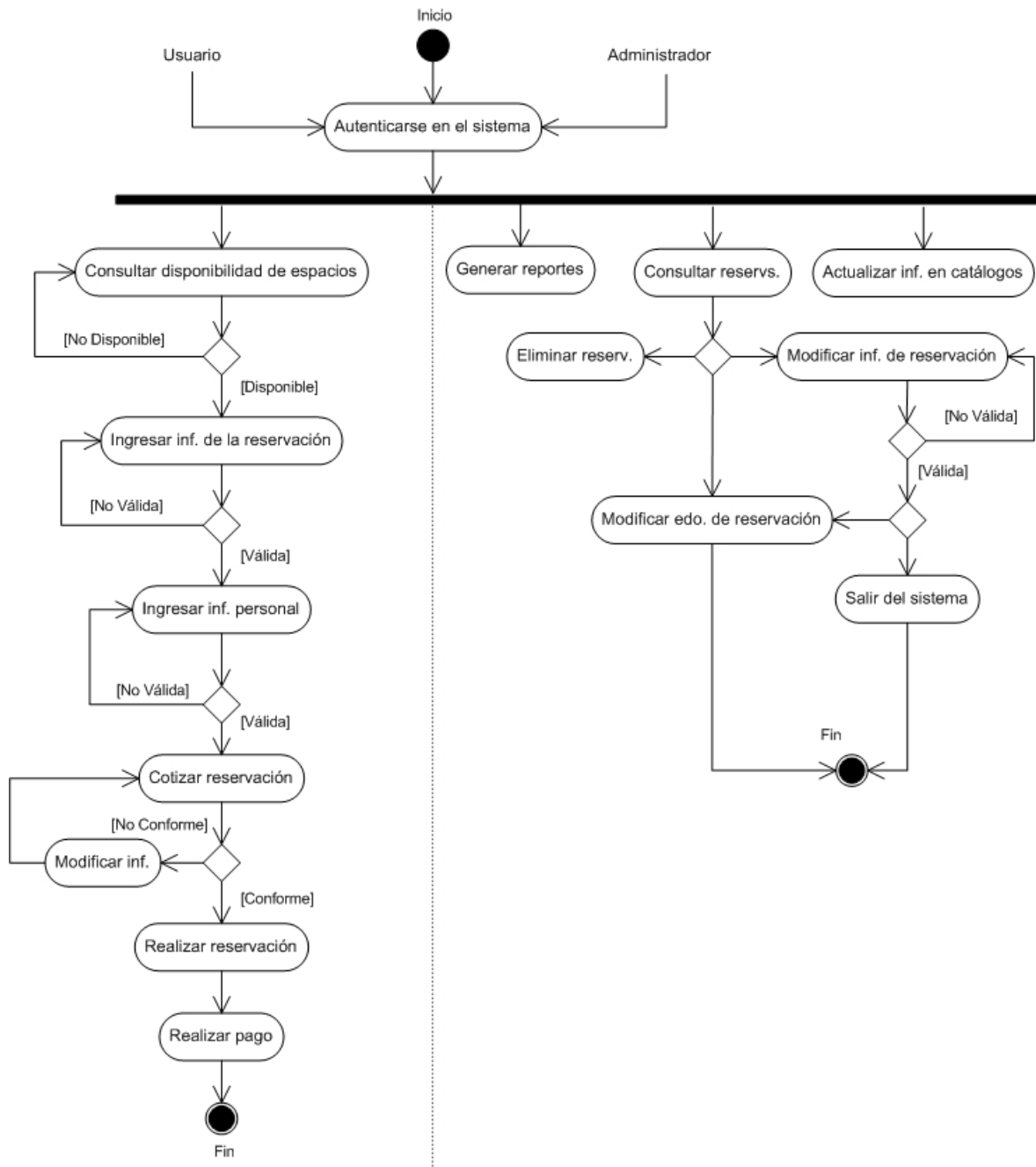


Figura 4.11 Diagrama de Actividades del SREITI.

4.2.2 Diseño de la Base de Datos

Para poder almacenar la información y hacer que esté disponible a múltiples usuarios (personas y otras aplicaciones) cuando se solicite, se considera necesario utilizar una base de datos cuyo diseño esté basado en un enfoque relacional pues éste, representa la tendencia dominante en el mercado actual además, constituye por si solo el avance más importante en toda la historia del campo de las bases de datos, características que permitirán su fácil mantenimiento a un costo bajo.

Además de cumplir con un enfoque relacional, el diseño, de la base de datos del SREITI debe de cumplir con una de las políticas que actualmente el II toma en cuenta a la hora de diseñar sus sistemas; dicha política consiste en rehusar información que ya existe en determinados sistemas y que es útil para otros, para de esta forma evitar redundancia de información entre sistemas y al mismo tiempo interconectarlos entre sí.

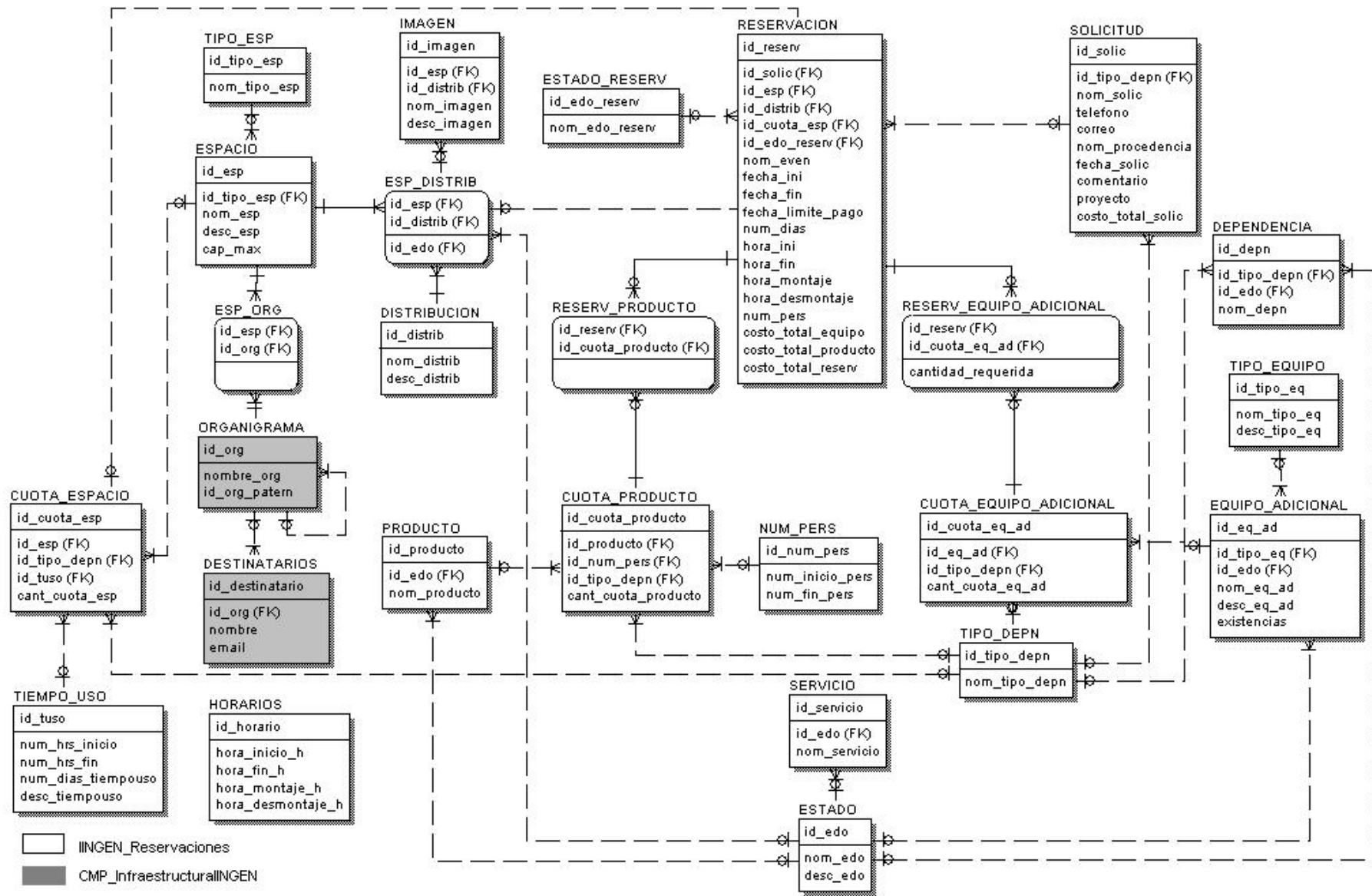
La base de datos que contiene información que le es útil al SREITI se llama CMP_InfraestructuralINGEN, de aquí se toma información de la estructura organizacional del II para realizar una relación de los espacios que le corresponden a cada unidad organizativa del II; también contiene información de los días feriados que se tomarán en cuenta para la realización de reservaciones.

4.2.2.1 Diagrama entidad - relación

El diagrama entidad – relación es un tipo de diagrama que nos ayuda a especificar la información de nuestro de sistema.

Para poder representar gráficamente el modelo de la base de datos del SREITI se utilizará un diagrama entidad – relación.

En la siguiente página se muestra dicho diagrama.



4.2.2.2 Descripción de las entidades del diagrama entidad – relación del SREITI

CUOTA_EQUIPO_ADICIONAL: Almacena información de la cuota que se debe pagar por el uso de algún equipo adicional en la reservación de un espacio. La cantidad de la cuota, está en función del equipo adicional seleccionado por el usuario y el tipo de dependencia a la que pertenece este último.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_cuota_eq_ad	PK	int (4)	NN, U	Identificador de la cuota del equipo adicional a utilizar.
id_eq_ad	FK1	int (4)	NN	Identificador del equipo adicional a utilizar.
id_tipo_depn	FK2	int (4)	NN	Identificador del tipo de dependencia a la que pertenece el usuario.
cantidad_cuota_eq_ad		decimal (9)	NN	Cantidad de dinero que se cobra por el uso de un equipo adicional.

CUOTA_ESPACIO: Almacena información de la cuota que se debe pagar por el uso de un determinado espacio. La cantidad de la cuota está en función del espacio seleccionado por el usuario, el tipo de dependencia a la que pertenece éste último, el número de espacios que sean seleccionados y el tiempo de uso.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_cuota_esp	PK	int (4)	NN, U	Identificador de la cuota del espacio a utilizar.
id_esp	FK1	int (4)	NN	Identificador del espacio a utilizar.
id_tipo_depn	FK2	int (4)	NN	Identificador del tipo de dependencia a la que pertenece el usuario.
id_tuso	FK4	int (4)	NN	Identificador del tiempo de uso del espacio.
cantidad_cuota_esp		decimal (9)	NN	Cantidad de dinero que se cobra por el uso de un espacio.

CUOTA_PRODUCTO: Almacena información de la cuota que se debe pagar por el consumo de un determinado producto. La cantidad de la cuota está en función de características propias del producto, del número de personas que lo consuman y el tipo de dependencia a la que pertenezca el usuario.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_cuota_producto	PK	int (4)	NN, U	Identificador de la cuota del producto.
id_producto	FK1	int (4)	NN	Identificador del producto a consumir.
id_num_pers	FK2	int (4)	NN	Identificador del número de personas que consumen un determinado producto.
id_tipo_depn	FK3	int (4)	NN	Identificador del tipo de dependencia a la que pertenece el usuario.
cantidad_cuota_producto		decimal (9)	NN	Cantidad de dinero que se cobra por el consumo de un producto.

DEPENDENCIA: Almacena información de las dependencias asociadas al tipo de dependencia a la que pertenece el usuario.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_depn	PK	int (4)	NN, U	Identificador de la dependencia.
id_tipo_depn	FK1	int (4)	NN	Identificador del tipo de dependencia.
id_edo	FK2	int (4)	NN	Identificador del estado de la dependencia.
nom_depn		varchar (100)	NN	Nombre de la dependencia.

DESTINATARIOS: Almacena información de las personas a quien les llegará un correo electrónico cuando una reservación sea realizada.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_destinatario	PK	int (4)	NN, U	Identificador del destinatario.
id_org	FK	int (4)	NN	Identificador de la organización a la que pertenece el destinatario.
nombre		varchar (150)	NN	Nombre del destinatario.
email		varchar (100)	NN	Correo electrónico del destinatario.

DISTRIBUCION: Almacena información de las distribuciones que un espacio puede llegar a tener.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_distrib	PK	int (4)	NN, U	Identificador de la distribución.
nom_distrib		varchar (50)	NN	Nombre de la distribución.
desc_distrib		varchar (100)	NN	Descripción de la distribución.

EQUIPO_ADICIONAL: Almacena información del equipo adicional que se puede solicitar al hacer la reservación de un espacio.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_eq_ad	PK	int (4)	NN, U	Identificador del equipo adicional.
id_tipo_eq	FK1	int (4)	NN	Identificador del tipo de equipo adicional.
id_edo	FK2	int (4)	NN	Descripción de la distribución.
nom_eq_ad		varchar (100)	NN	Nombre del equipo adicional.
desc_aq_ad		varchar (100)	NN	Descripción del equipo adicional.
existencias		int (4)	NN	Número de existencias del equipo.

ESP_DISTRIB: Almacena información de las distribuciones asociadas a cada uno de los espacios existentes.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_esp	PK, FK1	int (4)	NN, U1	Identificador del espacio.
id_distrib	PK, FK2	int (4)	NN, U2	Identificador de la distribución.
id_edo	FK3	int (4)	NN	Identificador del estado.

ESP_ORG: Almacena información de los espacios asociados a cada una de las organizaciones existentes.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_esp	PK, FK1	int (4)	NN, U1	Identificador del espacio.
id_org	PK, FK2	int (4)	NN, U2	Identificador de la organización.

ESPACIO: Almacena información de los espacios que se ponen a disposición de los usuarios para realizar su reservación.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_esp	PK	int (4)	NN, U	Identificador del espacio.
id_tipo_esp	FK	int (4)	NN	Identificador del tipo de espacio.
nom_esp		varchar (100)	NN	Nombre del espacio.
desc_esp		varchar (100)	NN	Descripción del espacio.
cap_max		int (4)	NN	Número máximo de personas que pueden ocupar el espacio.

ESTADO: Almacena información del estado que puede ser asignado a los espacios, distribuciones, productos, equipos adicionales y dependencias. Hasta el momento el estado puede ser activo o cancelado.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_edo	PK	int (4)	NN, U	Identificador del estado.
nom_edo		varchar (100)	NN	Nombre del estado.
desc_edo		varchar (150)	NN	Descripción del estado.

ESTADO_RESERV: Almacena información del estado que puede ser asignado a la reservación. Hasta el momento el estado de la reservación puede ser confirmada, cancelada o pendiente.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_edo_reserv	PK	int (4)	NN,U	Identificador del estado de la reservación.
nom_edo_reserv		varchar (50)	NN	Nombre del estado.

HORARIOS: Almacena información de los horarios en los que se realizan las reservaciones.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_horario	PK	int (4)	NN	Identificador del horario.
hora_inicio		int (4)	NN	Hora inicio del horario.
hora_fin		int (4)	NN	Hora fin del horario.
hora_montaje		int (4)	NN	Hora montaje del horario.
hora_desmontaje		int (4)	NN	Hora desmontaje del horario.

IMAGEN: Almacena información de las imágenes asociadas a los espacios y distribuciones.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_imagen	PK	int (4)	NN, U	Identificador de la imagen.
id_esp	FK1	int (4)	NN	Identificador del espacio.
id_distrib	FK2	int (4)	NN	Identificador de la distribución.
nom_imagen		varchar (250)	NN	Ruta física de la imagen (incluye nombre).
desc_imagen		varchar (150)	NN	Descripción de la imagen.

NUM_PERS: Almacena información del número de personas que consumen un determinado producto.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_num_pers	PK	int (4)	NN, U	Identificador del número de personas.
num_inicio_pers		int (4)	NN	Número inicio de personas.
num_fin_pers		int (4)	NN	Número fin de personas.

ORGANIGRAMA: Almacena información del organigrama del II.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_org	PK	int (4)	NN, U	Identificador de la organización.
nom_org		varchar ()	NN	Nombre de la organización.
id_org_patern		int (4)	NN	Identificador de la organización padre.

PRODUCTO: Almacena información del producto que puede ser solicitado dentro del servicio de cafetería.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_producto	PK	int (4)	NN, U	Identificador del producto.
id_edo	FK	int (4)	NN	Identificador del estado del producto.
nom_producto		varchar (150)	NN	Nombre del producto.

RESERV_EQUIPO_ADICIONAL: Almacena información del equipo adicional solicitado en una determinada reservación.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_reserva	PK, FK1	int (4)	NN, U1	Identificador de la reservación.
id_cuota_eq_ad	PK, FK2	int (4)	NN, U2	Identificador de la cuota del equipo adicional a utilizar.
cantidad_requerida		smallint (2)	NN	Cantidad de equipo requerida.

RESERV_PRODUCTO: Almacena información de los productos que solicita una determinada reservación dentro del servicio de cafetería.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_reserva	PK, FK1	int (4)	NN, U1	Identificador de la reservación.
id_cuota_producto	PK, FK2	int (4)	NN, U2	Identificador de la cuota del producto a consumir.

RESERVACION: Almacena información de la reservación.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_reserv	PK	int (4)	NN, U	Identificador de la reservación.
id_solic	FK1	int (4)	NN	Identificador de la solicitud a la que pertenece la reservación.
id_esp	FK2	int (4)	NN	Identificador del espacio.
id_distrib	FK3	int (4)	NN	Identificador de la distribución del espacio seleccionado.
id_cuota_esp	FK4	int (4)	NN	Identificador de la cuota del espacio seleccionado.
id_edo_reserv	FK5	Int (4)	NN	Identificador del estado de la reservación.
nom_even		varchar (100)	NN	Nombre del evento.
fecha_ini		smalldate time (4)	NN	Fecha inicio del evento.
fecha_fin		smalldate time (4)	NN	Fecha fin del evento.
fecha_limite_pago		smalldate time (4)	NN	Fecha límite de pago de la reservación.
num_dias		smallint (2)	NN	Número de días de la reservación.
hora_ini		smallint (2)	NN	Hora inicio del evento.
hora_fin		smallint (2)	NN	Hora fin del evento.
hora_montaje		smallint (2)	NN	Hora de montaje del evento.
hora_desmonaje		smallint (2)	NN	Hora de desmontaje del evento.
num_pers		smallint (2)	NN	Número de personas que utilizarán el espacio.
costo_total_equipo		decimal (9)	NN	Costo total del equipo adicional a utilizar.
costo_total_producto		decimal (9)	NN	Costo total de los productos a consumir dentro del servicio de cafetería.
costo_total_reserv		decimal (9)	NN	Costo total de la reservación. El total es la suma de la cuota del espacio, el costo total del equipo adicional y el costo total de los productos.

SERVICIO: Almacena información de los servicios adicionales que se proporcionan en la reservación de los espacios.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_servicio	PK	int (4)	NN, U	Identificador del servicio.
id_edo	FK	int (4)	NN	Identificador del estado del servicio.
nom_servicio		varchar (150)	NN	Nombre del servicio.

SOLICITUD: Almacena información del usuario que solicita la reservación de uno o más espacios.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_solic	PK	int (4)	NN, U	Identificador de la solicitud.
id_tipo_depn	FK	int (4)	NN	Identificador del tipo de dependencia.
nom_solic		varchar (100)	NN	Nombre del usuario (solicitante)
telefono		varchar (20)	NN	Número telefónico del usuario.
correo		varchar (60)	NN	Correo electrónico del usuario.
nom_procedencia		varchar (60)	NN	Nombre de procedencia.
fecha_solic		smalldate time (4)	NN	Fecha de solicitud.
comentario		varchar (250)	NN	Comentarios que el usuario envía al administrador del sistema.
proyecto		varchar (250)	NN	Descripción del proyecto al que se atribuirá el costo de la solicitud.
costo_total_solic		decimal (9)	NN	Costo total de la solicitud. El total es la suma del costo total de cada reservación.

TIEMPO_USO: Almacena información del tiempo de uso de los espacios.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_tuso	PK	int (4)	NN, U	Identificador del tiempo de uso.
num_hrs_inicio		int (4)	NN	Número de horas inicio.
num_hrs_fin		int (4)	NN	Número de horas fin.
num_dias_tiemposuso		int (4)	NN	Número de días del tiempo de uso.
desc_tiempo_uso		varchar (200)	NN	Descripción del tiempo de uso.

TIPO_DEPN: Almacena información del tipo de dependencia a la que pertenece el usuario.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_tipo_depn	PK	int (4)	NN, U	Identificador del tipo de dependencia.
nom_tipo_depn		varchar (50)	NN	Nombre del tipo de dependencia.

TIPO_EQUIPO: Almacena información del tipo de equipo adicional que se ofrece en la reservación de espacios.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_tipo_eq	PK	int (4)	NN, U	Identificador del tipo de equipo.
nom_tipo_eq		varchar (50)	NN	Nombre del tipo de equipo.
desc_tipo_eq		varchar (200)	NN	Descripción del tipo de equipo.

TIPO_ESP: Almacena información del tipo de espacios que existen.

Columna	Tipo de llave	Tipo de dato	Nulos/Únicos	Descripción
id_tipo_esp	PK	int (4)	NN, U	Identificador del tipo de espacio.
nom_tipo_esp		int (4)	NN	Nombre del tipo de espacio.

4.2.3 El SREITI como dos subsistemas

Para poder representar una separación de aspectos de diseño y unas variantes de funcionalidad, se consideró necesario dividir al SREITI en dos subsistemas. De esta forma, cada subsistema proporcionará las interfaces y funcionalidades requeridas para cada dependencia.

La figura de abajo ilustra este concepto. En esta misma figura se puede observar que también se incluyen los paquetes obtenidos en el análisis, esto es debido a que serán utilizados por cada uno de los subsistemas.

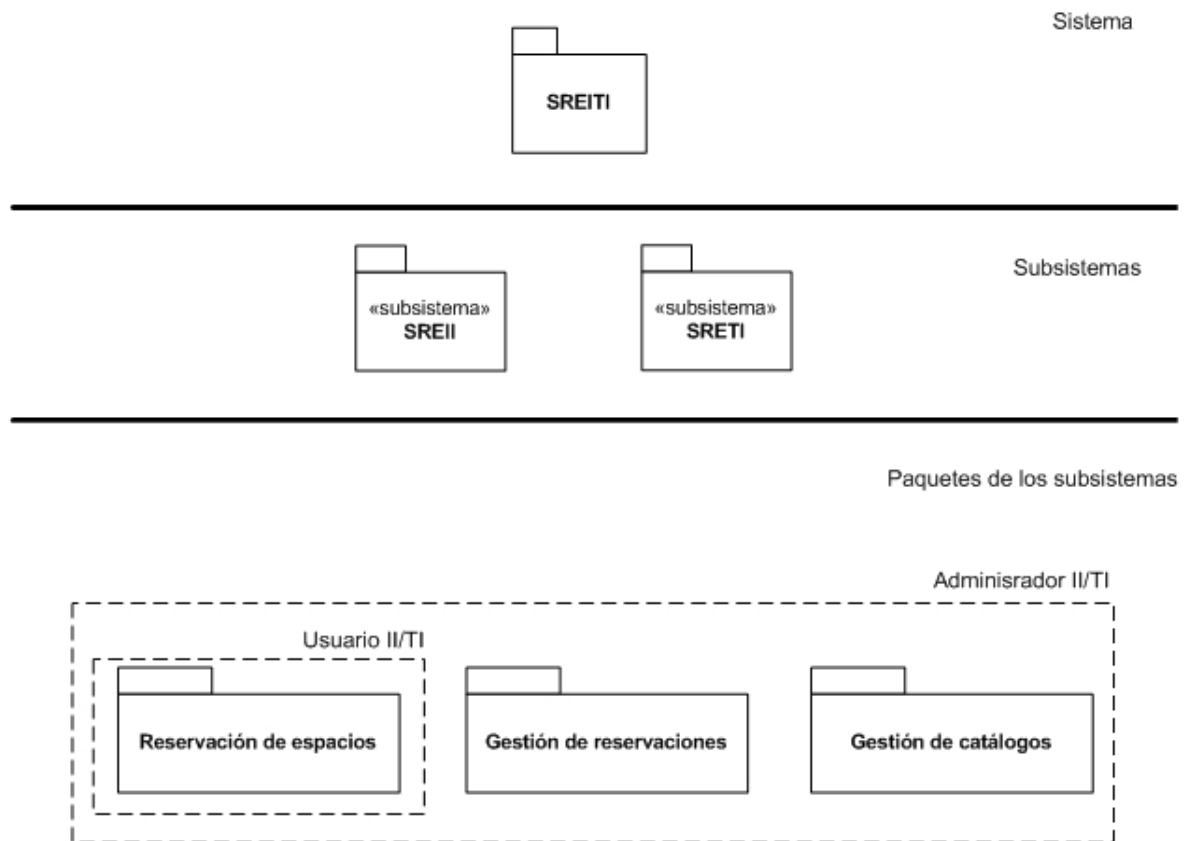


Figura 4.12 Subsistemas del SREITI.

4.2.4 Patrón de diseño en 3 capas

Las aplicaciones han pasado por un proceso evolutivo enorme. Desde su inicio con las aplicaciones monolíticas donde en una aplicación la interfaces de usuario, la lógica de cómo funcionaba la empresa y el manejo de la información almacenada y recuperada estaban juntas.

Luego la industria implementó un nuevo modelo de aplicaciones, las aplicaciones distribuidas cliente/servidor, que se convirtió en el estándar por un tiempo. Pero con la llegada de las aplicaciones Web se hizo necesario un nuevo estándar para las operaciones de los sistemas, y es así como surgió el modelo de las aplicaciones en n-capas.

Este modelo por lo general esta basado en un esquema de tres capas: acceso a datos, lógica de negocios e interfaz de usuario (presentación); no obstante, es posible continuar dividiendo este modelo en sub capas para una mayor flexibilidad en la distribución en el equipo de desarrollo y durante el mantenimiento.

A continuación se da una explicación más detallada de cada una estas capas:

- **Presentación:** La capa de presentación representa la parte del sistema con la que interactúa el usuario. En una aplicación Web, un navegador puede utilizarse como cliente del sistema, pero esta no es la única posibilidad, también puede generarse una aplicación que cumpla las funciones de un cliente "ligero" para interactuar con el usuario.
- **Lógica de negocio:** El comportamiento de la aplicación es definido por los componentes que modelan la lógica de negocio. Estos componentes reciben las acciones a realizar a través de la capa de presentación, y llevan a cabo las tareas necesarias utilizando la capa de datos para manipular la información del sistema. Tener la lógica de negocio separada del resto del sistema también permite una integración más sencilla y eficaz con sistemas externos, ya que la misma lógica utilizada por la capa de presentación puede ser accedida desde procesos automáticos que intercambian información con los mismos.
- **Acceso a datos:** Sus funciones incluyen el almacenamiento, la actualización y la consulta de todos los datos contenidos en el sistema. En la práctica, esta capa es esencialmente un servidor de bases de datos aunque podría ser cualquier otra fuente de información. Gracias a esta división, es posible agregar soporte para una nueva base de datos en un período de tiempo relativamente corto. La capa de datos puede estar en el mismo servidor que las de lógica de negocio y presentación, en un servidor independiente, o incluso estar distribuida entre un conjunto de servidores.

En el SREITI se adoptó este tipo de diseño ya que ofrece las siguientes ventajas:

- **Permite la reutilización:** la aplicación está formada por una serie de componentes que se comunican entre sí a través de interfaces y que cooperan para lograr el comportamiento deseado. Esto permite no solamente que estos componentes puedan ser fácilmente reemplazados por otros, por ejemplo porque se necesita mayor funcionalidad sino también que los mismos puedan ser utilizados para otras aplicaciones.
- **Acompaña el crecimiento:** cada uno de los componentes de la aplicación pueden colocarse en el mismo equipo o distribuirse a través de una red. De esta manera, proyectos de gran tamaño pueden dividirse en pequeños proyectos más simples y manejables, que se pueden implementar en forma progresiva, agregando nuevos servicios según la medida de crecimiento de la organización.
- **Uso eficiente del hardware:** debido a que los componentes pueden ser distribuidos a través de toda la red, se puede hacer un uso más eficiente de los recursos de hardware. En vez de necesitarse grandes servidores que contengan la lógica de negocios y los datos, es posible distribuirlos en varias máquinas más pequeñas, económicas y fáciles de ser reemplazadas.
- **Mínima inversión inicial:** generalmente, un cambio en el sistema de gestión traía asociado una inversión importante en actualización de hardware en los clientes debido a nuevas necesidades de cómputo de las aplicaciones "pesadas". Los clientes "ligeros" de esta nueva modalidad permite mantener el equipamiento actual o adquirir uno de muy bajo costo y actualizar, sólo en caso de ser necesario, la tecnología del servidor o servidores.

- **Distintas presentaciones:** debido a que separa la presentación de la lógica de negocios, es mucho más sencillo realizar tantas presentaciones diferentes como dispositivos con capacidades e interfaces se tenga (PC, PDA, celulares, etc.).
- **Encapsula los datos:** debido a que las aplicaciones cliente se comunican con los datos a través de peticiones que los servidores responden ocultando y encapsulando los detalles de la lógica de la aplicación, obtenemos un nivel de abstracción que permite un acceso a los datos consistente y seguro. Con esto se pretende que si hay cambios en la capa de datos, la capa de negocios se haga cargo de administrar tales cambios y el cliente, en la mayor parte de los casos ni se entere.
- **Ahorra tiempo y costos:** en el desarrollo de nuevas aplicaciones y la integración en el resto de los procesos de gestión de la empresa.
- **Mejor calidad en las aplicaciones:** como las aplicaciones son construidas en unidades separadas, estas pueden ser probadas independientemente y con mucho más detalle, esto conduce a obtener un producto mucho más sólido.

La siguiente figura muestra este patrón de diseño.

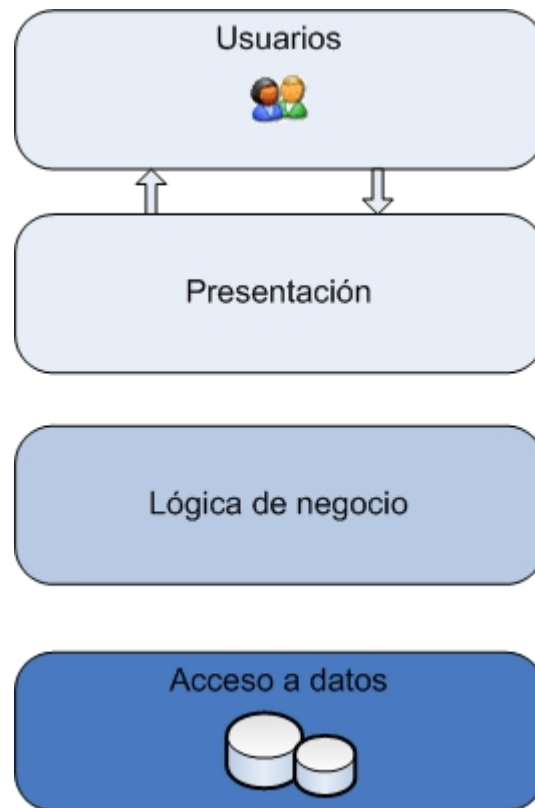


Figura 4.12 Patrón de diseño en tres capas.

4.3 Diseño físico

En esta etapa del diseño, se define la forma en que se realizará el procesamiento entre los distintos equipos que conforman la solución, incluyendo los servicios y procesos de base. Aquí, los elementos definidos en el diseño lógico se “mapean” a componentes de software (servicios, procesos, otros) o de hardware que definen de una forma más precisa como se ejecutará la solución. Para realizar esta tarea, haremos uso del modelo de despliegue.

4.3.1 Modelo de despliegue

En el modelo de despliegue se describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. En este modelo, se tomará como base el patrón de tres capas visto en la sección anterior.

En la siguiente figura se puede observar una solución Web compuesta por 3 por nodos procesadores: Clientes, Servidor Web y Servidor de bases de datos. Dentro de los nodos existen componentes que ejecutan procesos y/o servicios y sus relaciones de dependencia, por ejemplo, el Internet Explorer muestra la página ASPX o HTML que corresponde a la presentación o interfaz de usuario de la aplicación.

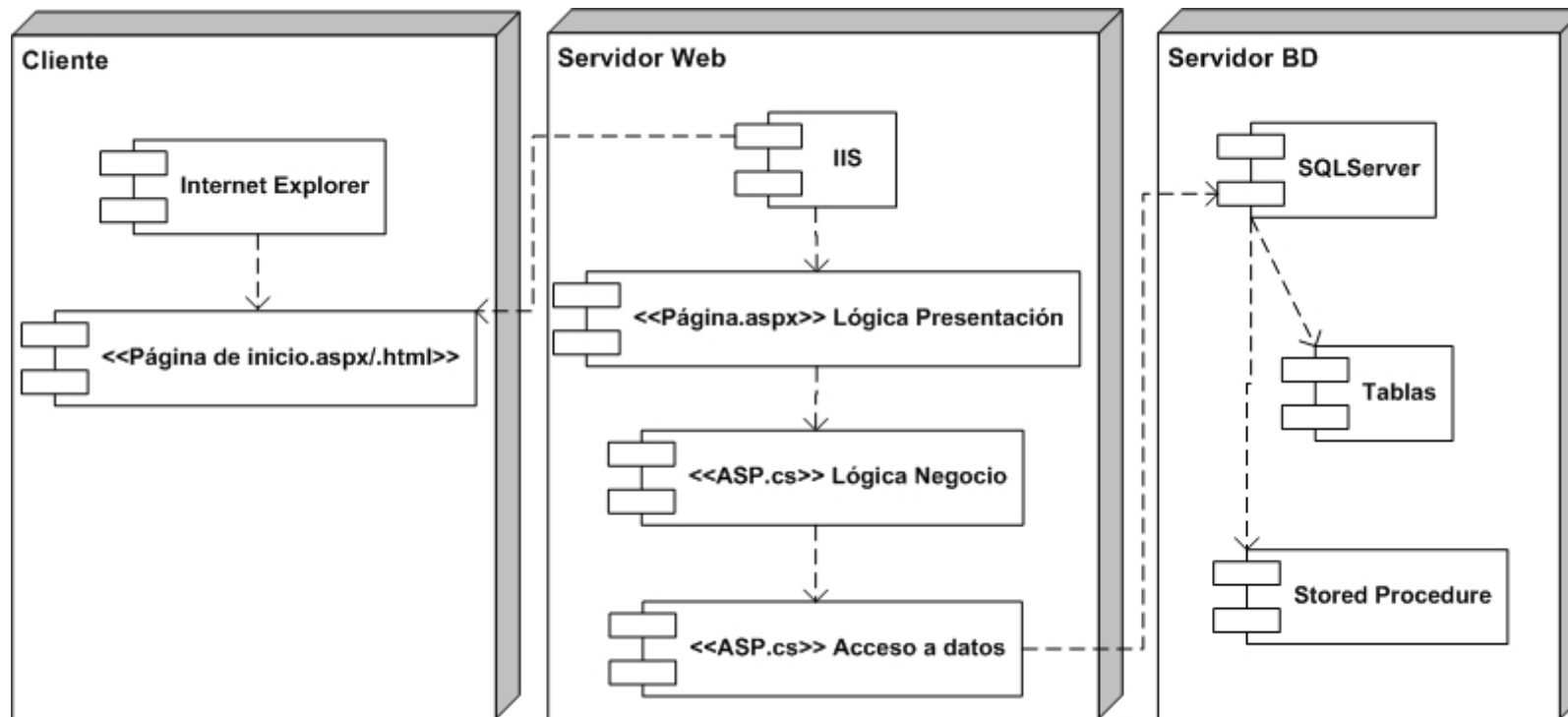


Figura 4.13 Modelo de despliegue del SREITI.

5 IMPLEMENTACION

En la implementación, las clases y subsistemas encontrados durante el diseño son implementados como componentes de archivos que contienen código fuente., los cuales posteriormente se prueban individualmente y se integran compilándolos y enlazándolos en uno o más ejecutables.

Para realizar esta tarea, haremos uso del Modelo de Implementación.

5.1 Modelo de implementación

El modelo de implementación describe como se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y como dependen los componentes unos de otros.

El modelo de implementación del SREITI que a continuación se presenta, está basado en la Plataforma .Net de Microsoft, a través de una aplicación Web ASP.NET construida en el entorno de desarrollo de Visual Studio .Net, utilizando como lenguaje de programación Visual C#.

5.2 Aplicación Web ASP .NET para el SREITI

Para implementar el SREITI se creo una aplicación Web por cada subsistema (SRETI y SREII) basada en tecnologías: ASP .NET, Servicios Web, ADO.NET y Microsoft SQL Server 2000 las cuales forman parte de la Plataforma .NET.

Una aplicación Web está compuesta por un conjunto de archivos, páginas, módulos y código ejecutable que se invocan o ejecutan dentro del ámbito de un directorio virtual (y sus subdirectorios) en el servidor Web de aplicaciones IIS (Internet Information Server).

Las aplicaciones Web creadas para el SRETI y SREII fueron nombradas como: ReservacionesTI_II y ReservacionesIINGEN_II respectivamente, este nombre coincide con el nombre de los directorios virtuales de cada una de las mismas, sin embargo el nombre de las aplicaciones puede modificarse y el acceso a las mismas no se ve afectado, pues está determinado por el nombre del directorio virtual.

Posteriormente para poder tener acceso remoto a las mismas, se copiaron al servidor de aplicaciones del II dentro del subdominio *aplicaciones*. Quedando así, las siguientes direcciones URL para el SRETI y el SREII, módulo administrador y usuario respectivamente:

- http://aplicaciones.iingen.unam.mx/ReservacionesTI_II/Admin/Default.apx
- http://aplicaciones.iingen.unam.mx/ReservacionesTI_II/User/Default.aspx
- http://aplicaciones.iingen.unam.mx/ReservacionesIINGEN_II/Admin/Default.aspx
- http://aplicaciones.iingen.unam.mx/ReservacionesIINGEN_II/User/Default.aspx

En donde:

<http://aplicaciones.iingen.unam.mx>: Es el dominio del servidor de aplicaciones del II.

ReservacionesTI_II: Nombre del directorio virtual para el SRETI.

Admin: Directorio en donde se encuentran todos los componentes que serán utilizados por el administrador.

User: Directorio en donde se encuentran todos los componentes que serán utilizados por el usuario.

Default.aspx : Página de inicio de cada aplicación.

5.2.1 Componentes

Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases en el modelo del diseño.

Los estereotipos de componentes que se utilizan para implementar el SREITI son los siguientes:

- <<executable>> es un programa que puede ser ejecutado en un nodo.
- <<file>> es un archivo que contiene código fuente o datos
- <<document>> es un documento que puede tener extensión .doc, .pdf, otros.

De acuerdo al tipo de archivos que se manejan en ASP .NET y su clasificación en los estereotipos mencionados, se obtuvieron los siguientes componentes:

5.2.1.1 Interfaces de usuario

Las interfaces de usuario son los componentes de tipo *executable* que el usuario final (usuario y administrador) ve en su explorador de Internet como páginas con extensión *.aspx*.

Éstas, están divididas en los subsistemas que se obtuvieron en el modelo de diseño y a su vez también están divididas en funcionalidades que son utilizadas por administrador y usuario.

5.2.1.2 Servicios Web

Un servicio Web es una aplicación que expone su lógica a clientes de cualquier plataforma mediante una interfaz accesible a través de la red utilizando protocolos estándar de Internet.

Para la implementación del SREITI se hizo uso de dos servicios Web, ***gestor_ctas_dominio*** y ***Wsmail***.

El primero, se conecta al Active Directory para obtener información de los objetos de tipo usuario. El tipo de información obtenida y el formato de salida de la misma, dependen del método seleccionado de entre todos los métodos de los que se compone este servicio Web. Para los fines del SREITI sólo se hace uso del método *BuscarPersonalxNombre_DS* el cual recibe como parámetro una variable de tipo cadena que contiene el nombre completo del usuario o su login de dominio y regresa una objeto *DataSet* el cual contiene información detallada del usuario; en caso de que el usuario no exista o se presente un error, el DataSet regresa sin información.

El segundo, se encarga de mandar correo electrónico autenticando el proceso de envío por medio de un servidor Exchange de correo. En el caso de este servicio Web, se hizo uso de su método *Enviar_Correo_Simple* el cual recibe como parámetros: una cadena compuesta por el remitente, destinatario, destinatarios con copia oculta, tema, cuerpo del mensaje y el formato HTML en el cual será enviado el mensaje; y regresa un valor de tipo booleano, verdadero si el correo fue enviado satisfactoriamente y falso si ocurrió algún problema.

Este tipo de componentes tienen la extensión *.asmx*.

5.2.1.3 Acceso a datos

La interacción de la aplicación con la información contenida en la base de datos se hizo utilizando la tecnología ADO.NET a través de una clase llamada *SQLHelper.cs*. Esta clase forma parte del componente *Data Access Application Block*, el cual es proporcionado por la ayuda de Microsoft.

La clase *SQLHelper.cs*, proporciona un conjunto de métodos estáticos (*ExecuteNonQuery*, *ExecuteDataSet*, *ExecuteReader*, *ExecuteScalar* y *ExecuteXMLReader*) a través de los cuales devuelve los objetos *SqlDataReader*, *DataSet* y *XmlReader* que se pueden utilizar para ejecutar distintos tipos de comandos en una base de datos de SQL Server. Los comandos para selección, inserción, borrado y modificación de datos (*SELECT*, *INSERT*, *DELETE* y *UPDATE*) que se utilizaron en la aplicación fueron llamados a través de *Procedimientos Almacenados*.

5.2.1.4 Manejador de Base de Datos

SQL Server 2000 fue seleccionado como manejador de base de datos por ser una herramienta robusta que también forma parte de la Plataforma .Net y que proporciona seguridad integrada utilizando los mecanismos de autenticación del sistema operativo (Kerberos o NTLM). La autorización la proporcionan los inicios de sesión y los permisos granulares que se pueden aplicar a cada uno de los objetos de base de datos.

5.2.1.5 Configuración

La información de configuración de los recursos de ASP.NET se encuentra en un archivo de configuración llamado *Web.config*, el cual está constituido por una jerarquía de etiquetas y subetiquetas XML anidadas con atributos que especifican las opciones de configuración que lee el Common Language Runtime (como la directiva de enlace de ensamblados, objetos de interacción remota, otros.) y valores que puede leer la aplicación. Todos los componentes que se encuentran por debajo de la aplicación, heredan las opciones de configuración de este archivo.

Las principales modificaciones que se hicieron a los archivos de configuración del SRETI y del SREII son la adición de las siguientes líneas, las cuales representan una especie de valores constantes que se manejan desde este nivel para no definirlo en cada archivo *.cs* (código de atrás en C# de las páginas *.aspx*) y de esta forma sea más fácil su manejo.

- **`<add key="nombrekey" value="127.0.0.1"; database= nombredb; uid=usuariodb; pwd=passwordusuariodb"></add>`**: **nombrekey** toma el valor de la cuenta con la que se hace la conexión a la base de datos, el nombre de la base y el servidor donde se encuentra la misma.
- **`<add key="nombrekey2" value="127.0.0.1"; database= nombredb; uid= admindb; pwd=passwordadmindb"></add>`**: Hace lo mismo que la llave de arriba pero con otro nombre de usuario y contraseña, esto es porque cada usuario tiene diferentes permisos definidos en la base de datos.
- **`<add key="ServidorCorreos" value="nombreservidor"></add>`**: Define el nombre del servidor de correos que se utiliza para el envío de los mismos.

Sólo en el *Web.config* del SREII fue necesario incluir la siguiente línea

- **`<identity impersonate="true"></ identity>`**: Permite controlar la identidad de la aplicación Web. El valor en *"true"* especifica que se utiliza la suplantación del cliente.

5.2.1.6 Diagramas de componentes

Las siguientes figuras ilustran los componentes que se obtuvieron para la reservación de espacios y la gestión de reservaciones y catálogos.

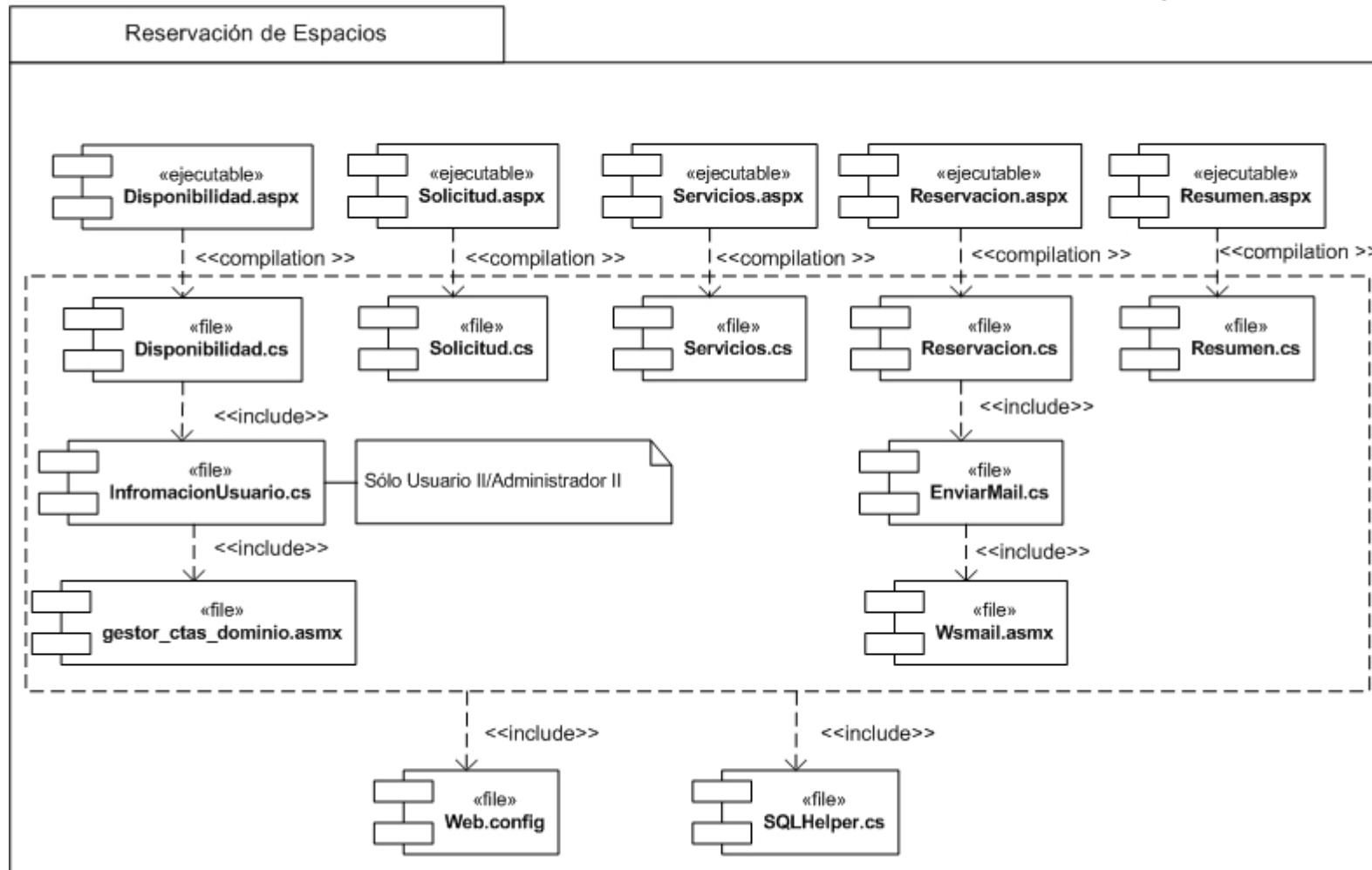


Figura 5.1. Diagrama de Componentes para realizar una reservación en el II o en la TI

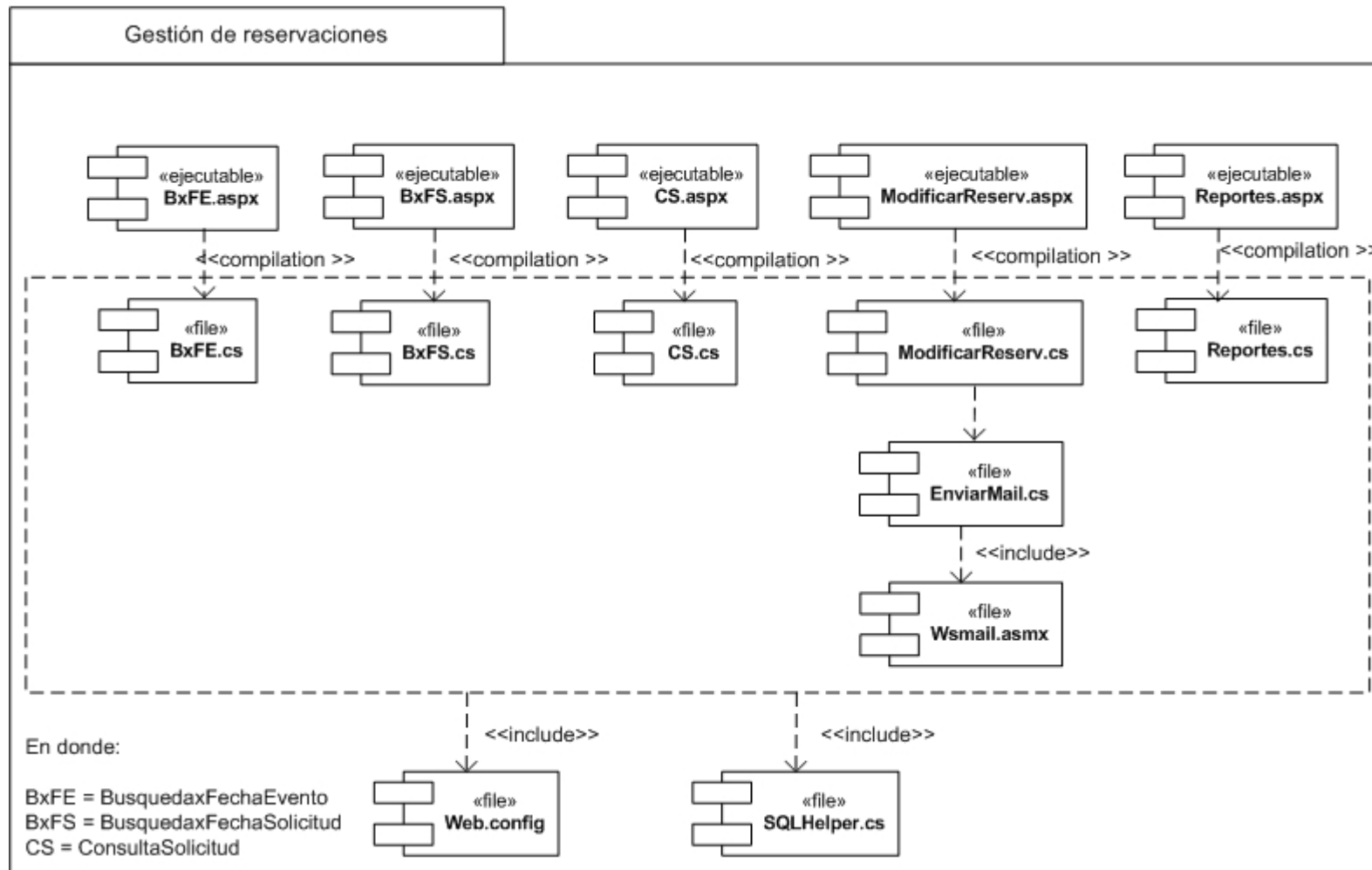


Figura 5.2. Diagrama de Componentes para realizar la gestión de reservaciones en el II o en la TI.

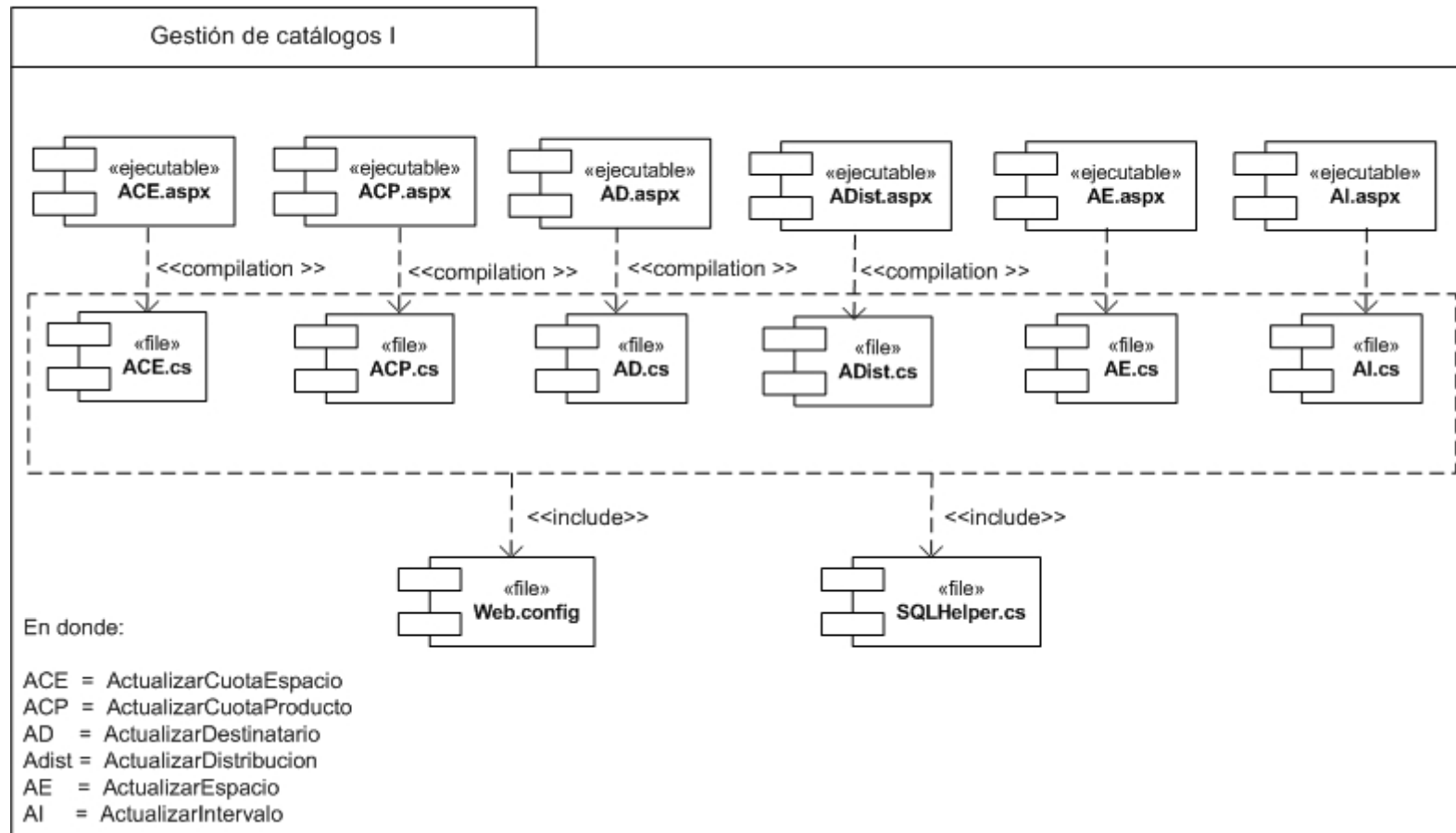


Figura 5.3. Primera parte del Diagrama de Componentes para realizar la gestión de catálogos en el II o en la TI.

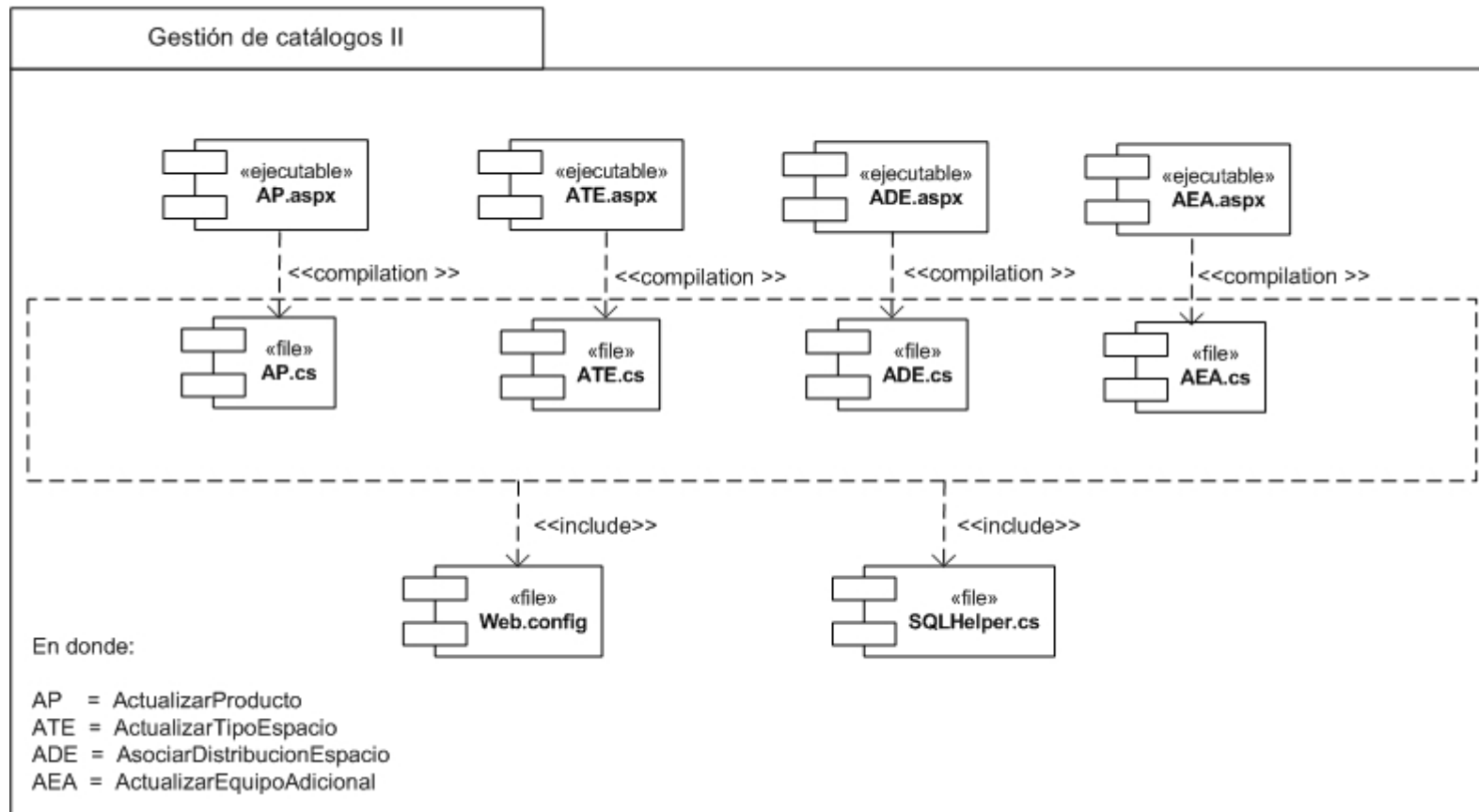


Figura 5.4. Segunda parte del Diagrama de Componentes para realizar la gestión de catálogos en el II o en la TI.

5.3 Arquitectura de seguridad

La figura 5.2 muestra el conjunto de servicios de seguridad proporcionados por las tecnologías utilizadas en la implementación del SREITI como dos subsistemas.

La autenticación y la autorización tienen lugar en muchos puntos independientes de todos los niveles. Estos servicios los proporcionan principalmente los Servicios de Internet Information Server (IIS), ASP.NET, y SQL Server. La autenticación integrada de Windows nos permite obtener las credenciales (nombre de usuario y contraseña) de identificación de un usuario y validar dichas credenciales contra una autoridad que así lo solicite, en este caso la aplicación Web. Este tipo de autenticación se hace posible gracias a que los usuarios que hay que autenticar de esta forma tienen asignada una cuenta de dominio de Windows. En la autorización, se confirma que un usuario que ha sido autenticado tiene el acceso a uno o más recursos.

Es importante tener en cuenta que la autenticación y la autorización van unidas, ya que cualquier directiva de autorización requiere previamente que el usuario haya sido autenticado.

Para el acceso al módulo de Administrador del SRETI y del SRETI, se creó un grupo de administradores respectivamente, a los cuales se les dio permisos de NTFS (New Technology File System) para tener acceso a los componentes correspondientes al mismo módulo.

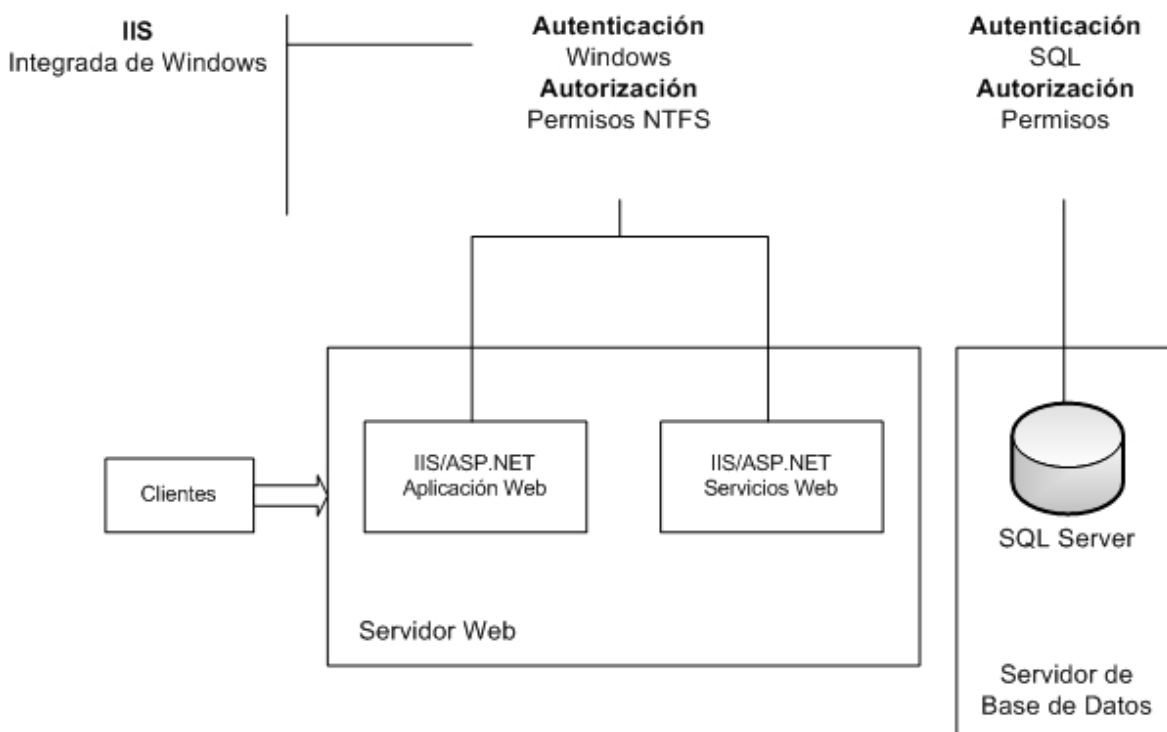


Figura 5.5. Arquitectura de seguridad para el SREII y el SRETI.

6 PRUEBAS

En las pruebas, se encuentran y corrigen los defectos en la calidad del software. Se valida que el producto de software funcione como se diseñó, y que los requisitos fueron implementados apropiadamente.

6.1 Modelo de pruebas

El modelo de pruebas describe cómo se prueban los componentes obtenidos en el modelo de implementación a través de los casos de prueba de unidad, integración y sistema.

6.1.1 Casos de prueba

Un caso de prueba especifica una forma de probar el sistema, incluye las entradas, los resultados y las condiciones bajo las cuales ha de evaluarse el sistema. Un caso de prueba se deriva de un caso de uso. Esto quiere decir que por cada caso de uso, existe una prueba.

A continuación se hace una descripción de los principales casos de prueba que se realizaron para el SREITI:

6.1.1.1 Pruebas de unidad

Las pruebas de unidad fueron realizadas en la implementación de cada componente del SRETI y del SREII con el propósito de probarlos como unidades individuales. Se llevaron a cabo los siguientes tipos de pruebas de unidad:

6.1.1.1.1 Prueba de especificación o de caja negra

Este tipo de prueba se realizó para verificar el comportamiento del componente sin tener en cuenta cómo se implementa dicho comportamiento del mismo. Es decir, sólo se tomó en cuenta la salida que el componente devuelve cuando se le da una determinada entrada.

Caso práctico: Consultar disponibilidad (Caso de uso: Realizar reservación).

Objetivo: Obtener una lista de horarios disponibles dada la fecha, horario y el espacio que se desea reservar.

Planeación de la prueba

Actividades a realizar:

- Seleccionar la fecha inicio, fecha fin, hora inicio, hora fin, tipo de espacio y espacio que se desea reservar.
- Seleccionar el botón "Consultar".
- Validar la información capturada.
- Mostrar una lista con los horarios disponibles.

Diseño de la prueba

Caso de prueba	Procedimiento de prueba
Consulta de disponibilidad con fechas.	Usuario: <ul style="list-style-type: none"> - Seleccionar fecha inicio y fecha fin de la reservación. - Seleccionar el botón "Consultar". Sistema: <ul style="list-style-type: none"> - Validar campos llenos y permitir continuar con el proceso.
Consulta de disponibilidad sin fechas.	Usuario: <ul style="list-style-type: none"> - Seleccionar el botón "Consultar". Sistema: <ul style="list-style-type: none"> - Validar campos llenos y mostrar un mensaje de error e impedir continuar con el proceso.
Selección de un intervalo de fechas correcto.	Usuario: <ul style="list-style-type: none"> - Seleccionar fecha inicio y fecha fin de la reservación. - Seleccionar el botón "Consultar". Sistema: <ul style="list-style-type: none"> - Validar intervalo de fechas y permitir continuar con el proceso.

<p>Selección de un intervalo de fechas incorrecto (Fecha inicio mayor a fecha fin, Fechas correspondientes a días sábados y/o domingos).</p>	<p>Usuario:</p> <ul style="list-style-type: none"> - Seleccionar fecha inicio y fecha fin de la reservación. - Seleccionar el botón "Consultar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar intervalo de fechas, mostrar mensaje de error e impedir continuar con el proceso.
<p>Selección de un horario correcto.</p>	<p>Usuario:</p> <ul style="list-style-type: none"> - Seleccionar hora inicio y hora fin de la reservación. - Seleccionar el botón "Consultar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar horario y permitir continuar con el proceso.
<p>Selección de un horario incorrecto (Hora inicio mayor o igual a hora fin).</p>	<p>Usuario:</p> <ul style="list-style-type: none"> - Seleccionar hora inicio y hora fin de la reservación. - Seleccionar el botón "Consultar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar horario, mostrar mensaje de error e impedir continuar con el proceso.
<p>Consultar disponibilidad de un espacio que no ha sido reservado (Ver Figura 6.1).</p>	<p>Usuario:</p> <ul style="list-style-type: none"> - Capturar la fecha inicio, fecha fin, hora inicio, hora fin, tipo de espacio y espacio que se desea reservar. - Seleccionar el botón "Consultar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar la información introducida. - Realizar búsqueda de horarios disponibles, mostrar el resultado en una lista y permitir continuar con el proceso.

<p>Consultar disponibilidad de un espacio que ya ha sido reservado (Ver Figura 6.2).</p>	<p>Usuario:</p> <ul style="list-style-type: none"> - Capturar la fecha inicio, fecha fin, hora inicio, hora fin, tipo de espacio y espacio que se desea reservar. - Seleccionar el botón "Consultar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar la información introducida. Realizar búsqueda de horarios disponibles, mostrar el resultado en una lista, mostrar un mensaje de alerta e impedir continuar con el proceso.
--	--

Ejecución de la prueba

Resultados de la prueba para los distintos casos de prueba	
Casos de prueba	Éxito
Consulta de disponibilidad con fechas.	✓
Consulta de disponibilidad sin fechas (Ver Figura 6.1).	✓
Selección de un intervalo de fechas correcto.	✓
Selección de un intervalo de fechas incorrecto. (Ver Figura 6.2)	✓
Selección de un horario correcto	✓
Selección de un horario incorrecto (Ver Figura 6.3)	✓
Consultar disponibilidad de un espacio que no ha sido reservado. (Ver Figura 6.4)	✓
Consultar disponibilidad de un espacio que ya ha sido reservado. (Ver Figura 6.5)	✓

6.1.1.1.2 Interfaces de usuario obtenidas en las pruebas de caja negra

Paso 1 de 5 **Consultar disponibilidad**

Nota: La fecha mínima para realizar una reservación es a partir de: 29/08/2006.

Fecha inicio:  * (dd/mm/yyyy)

Fecha fin:  * (dd/mm/yyyy)

Hora inicio: 8  Hrs.

Hora fin: 9  Hrs.

Tipo de espacio que desea reservar: Espacios individuales 

Espacio: Auditorio  **Capacidad máxima 127 personas.**

- La fecha es requerida
- La fecha es requerida

Imágenes del espacio



Auditorio - Estrado

1 2 3

Figura 6.1. Interfaz de usuario para: Consulta de disponibilidad sin fechas.

Paso 1 de 5

Consultar disponibilidad

Nota: La fecha mínima para realizar una reservación es a partir de: 29/08/2006.

Fecha inicio:	<input type="text" value="31/08/2006"/>  (dd/mm/yyyy)
Fecha fin:	<input type="text" value="29/08/2006"/>  (dd/mm/yyyy)
Hora inicio:	<input type="text" value="8"/>  Hrs.
Hora fin:	<input type="text" value="9"/>  Hrs.
Tipo de espacio que desea reservar:	<input type="text" value="Espacios individuales"/> 
Espacio:	<input type="text" value="Auditorio"/>  Capacidad máxima 127 personas.



La fecha inicio no puede ser mayor a la fecha fin

Consultar

Figura 6.2. Interfaz de usuario para: Selección de un intervalo de fechas incorrecto.

Paso 1 de 5

Consultar disponibilidad

Nota: La fecha mínima para realizar una reservación es a partir de: 29/08/2006.

Fecha inicio:	<input type="text" value="31/08/2006"/>		(dd/mm/yyyy)
Fecha fin:	<input type="text" value="31/08/2006"/>		(dd/mm/yyyy)
Hora inicio:	<input type="text" value="10"/>		Hrs.
Hora fin:	<input type="text" value="10"/>		Hrs.
Tipo de espacio que desea reservar:	<input type="text" value="Espacios individuales"/>		
Espacio:	<input type="text" value="Auditorio"/>		Capacidad máxima 127 personas.



¡La hora inicio no puede ser igual a la hora fin!

Consultar

Figura 6.3. Interfaz de usuario para: Selección de un horario incorrecto

Paso 1 de 5

Consultar disponibilidad

Nota: La fecha mínima para realizar una reservación es a partir de:31/08/2006.

Fecha inicio: (dd/mm/yyyy)

Fecha fin: (dd/mm/yyyy)

Hora inicio: Hrs.

Hora fin: Hrs.

Tipo de espacio que desea reservar:

Espacio: Capacidad máxima 127 personas.

Las imágenes mostradas son ilustrativas, por lo que no se incluye servicio de mantelería

Imágenes del espacio



Auditorio - Estrado

1 2 3

Horarios disponibles para las fechas seleccionadas

Hora inicio	Hora fin
8:00	9:00
9:00	10:00
10:00	11:00
11:00	12:00
12:00	13:00
13:00	14:00
14:00	15:00
15:00	16:00
16:00	17:00
17:00	18:00
18:00	19:00
19:00	20:00
20:00	21:00

Figura 6.4. Interfaz de usuario para: Consultar disponibilidad de un espacio que no ha sido reservado.

Paso 1 de 5

Consultar disponibilidad

Nota: La fecha mínima para realizar una reservación es a partir de:29/08/2006.

Fecha inicio: (dd/mm/yyyy)

Fecha fin: (dd/mm/yyyy)

Hora inicio: Hrs.

Hora fin: Hrs.

Tipo de espacio que desea reservar:

Espacio: Capacidad máxima 127 personas.

Imágenes del espacio



Auditorio - Estrado

1 2 3

Horarios disponibles para las fechas seleccionadas

Hora inicio	Hora fin
8:00	9:00
9:00	10:00
10:00	11:00
11:00	12:00
12:00	13:00
13:00	14:00
14:00	15:00
15:00	16:00
16:00	17:00
17:00	18:00

Ya existe una reservación con el horario que Usted seleccionó. Ver lista de Horarios disponibles

Figura 6.5. Interfaz de usuario para: Consultar disponibilidad de un espacio que ya ha sido reservado.

6.1.1.1.3 Prueba de estructura o de caja blanca

En este tipo de pruebas, se verificó la implementación interna de cada componente para verificar que cada uno de ellos funcionara internamente como se quería.

Caso práctico: Validar fechas en Consultar disponibilidad (Caso de uso: Realizar reservación).

La siguiente figura ilustra el método que se utiliza en el componente *Disponibilidad.cs* para validar las fechas introducidas por el usuario. El método deberá regresar la variable *statusAccion* con valor igual a *false* en caso de entrar en cualquier instrucción *if* y al ser llamado por otro método, al regresar un *false*, esto impedirá la continuación del flujo.

```
public bool ValidaFechas()
1      {
2          statusAccion = false;
3          DateTime FechaInicio = Convert.ToDateTime(DateFechaInicio.Text);
4          DateTime FechaFin = Convert.ToDateTime(DateFechaFin.Text);
5          if(FechaInicio > FechaFin){
6              MsgEventos = "La fecha inicio no puede ser mayor a la fecha fin";
7              return statusAccion;
8          }
9          DateTime dtFechaMinReser = DTminReserv();
10         if(FechaInicio < dtFechaMinReser){
11             MsgEventos = "La fecha inicio no puede ser menor a la fecha minima
12                 de reservación " + dtFechaMinReser.ToString("dd/MM/yyyy");
13             return statusAccion;
14         }
15         //Restricción de sábados y domingos
16         if(FechaInicio.DayOfWeek.ToString().Equals("Saturday")
17             || FechaFin.DayOfWeek.ToString().Equals("Saturday")){
18             MsgEventos = "No se hacen reservaciones en día Sábado";
19             return statusAccion;
20         }
21         if(FechaInicio.DayOfWeek.ToString().Equals("Sunday")
22             || FechaFin.DayOfWeek.ToString().Equals("Sunday")){
23             MsgEventos = "No se hacen reservaciones en día Domingo";
24             return statusAccion;
25         }
26         else{
27             statusAccion = true;
28             return statusAccion;
29         }
30     }
```

Figura 6.6. Código fuente del método *ValidaFechas* del componente *Disponibilidad.cs*.

6.1.1.2 Pruebas de integración

Las pruebas de integración se utilizaron para verificar que los componentes interaccionaran entre si de la forma apropiada después de haber sido integrados en una construcción.

Caso práctico: Caso de uso: Realizar reservación con servicios adicionales.

Objetivo: Realizar la reservación de espacios y solicitar servicios adicionales.

Planeación de la prueba

Actividades a realizar:

- Consultar la disponibilidad del espacio.
- Llenar solicitud.
- Solicitar servicios adicionales.
- Realizar reservación.
- Imprimir comprobante de reservación.

Diseño de la prueba

Caso de prueba	Procedimiento de prueba
Consultar disponibilidad de un espacio (Ver Figura 6.1).	Usuario: <ul style="list-style-type: none"> - Capturar la fecha inicio, fecha fin, hora inicio, hora fin, tipo de espacio y espacio que se desea reservar. - Seleccionar el botón "Consultar". Sistema: <ul style="list-style-type: none"> - Validar la información introducida. - Realizar búsqueda de horarios disponibles, mostrar el resultado en una lista y permitir continuar con el proceso.
Llenar solicitud.	Usuario: <ul style="list-style-type: none"> - Capturar información de usuario y reservación. - Seleccionar el botón "Continuar". Sistema: <ul style="list-style-type: none"> - Validar la información introducida. y permitir continuar con el proceso.

Solicitar servicios adicionales.	<p>Usuario:</p> <ul style="list-style-type: none"> - Seleccionar servicios adicionales de cafetería y equipo audiovisual. - Seleccionar el botón "Continuar". <p>Sistema:</p> <ul style="list-style-type: none"> - Validar la información introducida. y permitir continuar con el proceso.
Realizar reservación del espacio.	<p>Usuario:</p> <ul style="list-style-type: none"> - Seleccionar el botón "Reservar". <p>Sistema:</p> <ul style="list-style-type: none"> - Generar un número de solicitud, asociar el número generado a la reservación y a los servicios adicionales, guardar la información en la base de datos y permitir continuar con el proceso.
Obtener comprobante de reservación.	<p>Sistema:</p> <ul style="list-style-type: none"> - Mostrar Comprobante de reservación..

Ejecución de la prueba

Resultados de la prueba para los distintos casos de prueba	
Casos de prueba	Éxito
Consultar disponibilidad de un espacio. (Ver Figura 6.7)	✓
Llenar solicitud. (Ver Figura 6.8)	✓
Solicitar servicios adicionales. (Ver Figura 6.9)	✓
Realizar reservación del espacio. (Ver Figura 6.10)	✓
Obtener comprobante de reservación. (Ver Figura 6.11)	✓

6.1.1.3 Interfaces de usuario obtenidas en las pruebas de integración

Paso 1 de 5

Consultar disponibilidad

Nota: La fecha mínima para realizar una reservación es a partir de: 31/08/2006.

Horarios disponibles para las fechas seleccionadas

Hora inicio	Hora fin
8:00	9:00
9:00	10:00
10:00	11:00
11:00	12:00
12:00	13:00
13:00	14:00
14:00	15:00
15:00	16:00
16:00	17:00
17:00	18:00
18:00	19:00
19:00	20:00
20:00	21:00

Fecha inicio: (dd/mm/yyyy)
Fecha fin: (dd/mm/yyyy)
Hora inicio: Hrs.
Hora fin: Hrs.
Tipo de espacio que desea reservar:
Espacio: Capacidad máxima 127 personas.
 Las imágenes mostradas son ilustrativas, por lo que no se incluye servicio de mantelería.

Imágenes del espacio



Auditorio - Estrado
1 2 3

Figura 6.7. Interfaz de usuario para: Consultar disponibilidad de un espacio.

Paso 2 de 5

Solicitar reservación

Información del evento

Espacio: Auditorio
Nombre del evento:
Número de personas a asistir:

Información del solicitante

Nombre del solicitante:
Teléfono:
Correo:
Dependencia:

Cargo al proyecto número:

Comentarios:

Nota: Máximo 250 caracteres.

Figura 6.8. Interfaz de usuario para: Llenar solicitud.

Paso 3 de 5

Solicitar servicios adicionales

Información de la reservación									
Espacio: Auditorio									
Fecha de evento: del 31/08/2006 al 31/08/2006 Fecha límite de pago: 28/08/2006									
Horario del evento: de 8:00 a 9:00 Hora de montaje: de 7:00 a 8:00 Hora de desmontaje: de 9:00 a 10:00									
Servicios Adicionales <input checked="" type="checkbox"/> Cafetería <input checked="" type="checkbox"/> Equipo adicional									
Equipo adicional	Descripción	Tipo	Existencias	Cantidad		Equipo solicitado	Cantidad	Cuota	
Pantalla de proyección fija	Grandes dimensiones	Equipo audiovisual	1	1	Agregar	Pantalla de proyección fija	1	0	Eliminar
Total del equipo adicional \$ 0									
Servicio de cafetería			Número de personas			Producto solicitado			
Agua	De 1 a 20	Agregar	Agua	De 1 a 20	Agregar	Agua	De 1 a 20	15.00	Eliminar
Café	De 1 a 20	Agregar							
Refresco	De 1 a 20	Agregar							
Té	De 1 a 20	Agregar							
Galletas	De 1 a 20	Agregar							
Total de servicio de cafetería \$ 15.00									
Costo por el uso del espacio \$ 300.00									
Total de la reservación \$ 315.00									
Regresar					Continuar				

Figura 6.9. Interfaz de usuario para: Solicitar servicios adicionales.

Paso 4 de 5 Realizar reservación

Espacios a reservar														
#Reserv	Fecha inicio	Fecha fin	Fecha de pago	Duración del evento	Hora inicio	Hora fin	Espacio	Distribución	#Personas	Evento	CostoxServEq	CostoxEsp	CostoxReserv	
1	31/08/2006	31/08/2006	28/08/2006	1 día(s)	8:00	9:00	Auditorio	Sin distribución	1	Prueba	0	300.00	315.00	Eliminar
Equipo adicional por espacio							Servicio de cafetería por espacio							
#Reserv	Equipo solicitado	Cantidad Requerida	Cuota	#Reserv	Producto solicitado	Número de personas	Cuota							
1	Pantalla de proyección fija	1	0	1	Agua	De 1 a 20	15.00							
Total del costo de la solicitud \$ 315.00														
Agregar otro espacio							Reservar							

Figura 6.10. Interfaz de usuario para: Realizar reservación del espacio.

Paso 5 de 5 Resumen e impresión de solicitud

Información del solicitante													
Nombre:	Jonathán V.			Número de solicitud: 94									
Teléfono:	56233600			Total \$ 315.00									
Correo:	jvelazquezg@iingen.unam.mx												
Dependencia:	Instituto de Ingeniería												
Cargo al proyecto número:	Sin cargo a proyecto												
Comentarios:													
<p>Nota: a) Su solicitud será activada una vez realizado el pago. Éste se deberá realizar en efectivo o con cheque a nombre de UHAM Torre de Ingeniería en la Coordinación Administrativa de la TI ubicada en el 3er piso con la señorita Laura Hernández Ascencio, de 10:00 a 14:00 horas, tres días antes de la fecha de su evento, de no ser así, su reservación se cancelará. b) Todas las reservaciones tienen una hora de montaje y una hora de desmontaje con respecto al horario de su evento. c) Si la duración de su evento excede de 5 días, la aprobación de su reservación queda a reserva de la Coordinación Administrativa.</p>													
Información del evento													
Espacios reservados													
# Reserv	Fecha inicio	Fecha fin	Fecha de pago	Duración del evento	Hora inicio	Hora fin	Espacio	Distribución	# Personas	Evento	CostoxServEq	CostoxEsp	CostoxReserv
1	31/08/2006	31/08/2006	28/08/2006	1 día(s)	8:00	9:00	Auditorio	Sin distribución	1	Prueba	0	300.00	315.00
Equipo adicional por espacio							Servicio de cafetería por espacio						
#Reserv	Equipo solicitado	Cantidad Requerida	Cuota	#Reserv	Producto solicitado	Número de personas	Cuota						
1	Pantalla de proyección fija	1	0	1	Agua	De 1 a 20	15.00						

Figura 6.11. Interfaz de usuario para: Obtener comprobante de reservación.

6.1.1.4 Pruebas de sistema

Las pruebas de sistema tienen el propósito de realizar las pruebas necesarias en cada iteración y recopilar los resultados de las pruebas de integración.

Las pruebas de sistema para el SREITI se realizaron cuando las pruebas de integración indicaron que el sistema satisfacía con los requisitos planteados en el capítulo 2.

Caso práctico: Pruebas de sistema realizadas a los subsistemas SRETI y SRETI.

CONCLUSIONES

A pesar de existir similitud en funcionalidades fue necesario dividir al SREITI en dos subsistemas (SRETI y SREII), esto con la finalidad de que ambos tuviesen una interfaz de usuario diferente y así permitir conservar su identidad gráfica, además de permitir anexar en un futuro otros requisitos por separado.

Ambos subsistemas, facilitan y agilizan el proceso de reservación y administración de los espacios reduciendo el tiempo que en un principio los usuarios y administradores tenían que dedicar para realizar dichas tareas.

El utilizar al PU como metodología de desarrollo de software, permitió planificar de forma iterativa e incremental la forma en que se llevaría a cabo el desarrollo del SREITI. Cabe destacar, que a pesar de ser una metodología de desarrollo robusta, fue fácil adaptarla a las necesidades del SREITI, tomando sólo aquellas partes que eran de utilidad para el desarrollo del mismo. Con respecto a los roles (analistas, especificadores de casos de uso, ingenieros de casos de uso, diseñadores de interfaz de usuario, arquitectos, ingenieros de componentes, integradores del sistema e ingenieros de prueba) que el PU propone utilizar para el desarrollo de software, puedo decir que al desarrollar el sistema yo sola, tuve que representar a cada uno de los mismos, situación que no fue problema, salvo el hecho de representar al ingeniero de componentes y el ingeniero de pruebas, pues me era difícil detectar mis propios errores de programación. Por ello, es recomendable que en el desarrollo de software participen mínimo dos personas.

En cuanto a tecnologías de implementación de software se refiere, el hecho de utilizar componentes de una misma plataforma en este caso .NET, permite integrar y coordinar los componentes de la misma para comunicar los resultados de la aplicación Web entre sí, sin invertir tiempo en este tipo de tareas; a diferencia de hacerlo combinando componentes de otras plataformas o tecnologías.

Es importante destacar que no sólo la metodología de desarrollo y la tecnología de implementación influyeron en el desarrollo de este sistema; el usuario final, en particular el de la TI, fue quien en mayor parte influyó en el desarrollo del mismo, ya que por ser una dependencia nueva en la UNAM, comenzaba a definir su forma de trabajar, circunstancia que daba como resultado requerimientos cambiantes muy a menudo; aunado a esto, la persona que tenía designado el papel de administrador del sistema, no estaba familiarizada con el uso de sistemas informáticos de este tipo. Esta situación sumada a los errores de funcionamiento detectados por los usuarios finales en cada entrega, dio como resultado un tiempo de desarrollo de 8 meses, superando así el tiempo estimado de entrega, el cual en un inicio fue de cinco meses.

Finalmente, como experiencia profesional puedo decir que al momento de desarrollar software, es importante contar con un documento en el que se registren los cambios realizados y los que a futuro se solicite realizar y que permita tener un control de los mismos, evitando así extender los tiempos de entrega.

Sin embargo debemos acostumbrarnos a que en el desarrollo de software es muy común que los requisitos cambien, y que ésta situación debe ser tomada en cuenta por las autoridades responsables del proyecto y los usuarios finales a la fecha de entrega, así mismo, debe hacerseles saber que los cambios en sus requisitos tienen un costo en tiempo, dinero y esfuerzo.

Por otro lado, soy de la idea de que debe existir una cultura en las personas de generaciones anteriores a la nuestra que no están relacionadas con el área de informática por el uso de sistemas de este tipo, pues lo que en la actualidad se busca es realizar sistemas informáticos para agilizar el procesamiento de la información y con esto, desarrollar herramientas que les ayude a resolver los problemas de una forma más rápida, permitiéndoles así realizar otras tareas que dependiendo de las mismas les pueden ayudar a su desarrollo profesional.

GLOSARIO

Active Directory: Es el servicio de directorio incluido con Windows 2000 Server, que almacena de forma centralizada toda la información de un dominio de administración. Mediante este mecanismo es posible crear una serie de objetos que estén agrupados en categorías lógicas de una red: usuarios, grupos, recursos, permisos, políticas.

Actor: Un conjunto coherente de roles que los usuarios de casos de uso desempeñan cuando interactúan con estos casos de uso.

Aplicación (sistema): Sistema que ofrece a un usuario final un conjunto coherente de casos de uso.

Aplicación distribuida: Es un nuevo modelo para resolver problemas de computación masiva utilizando un gran número de computadoras organizadas que comparten recursos de software y hardware, situadas en distintos lugares y pertenecientes a diferentes dominios de administración.

Arquitectura software: Conjunto de decisiones significativas acerca de la organización de un sistema software, la selección de los elementos estructurales a partir de los cuales se compone el sistema, y la interfaces entre ellos, junto con su comportamiento. La arquitectura del software se interesa no sólo por la estructura y el comportamiento, sino también por las restricciones y compromisos de uso, funcionalidad, flexibilidad al cambio, reutilización, economía y tecnología.

Artefacto: Pieza de información tangible que es creada, modificada y usada por los trabajadores al realizar las actividades. Un artefacto puede ser un modelo, un elemento de un modelo, o un documento.

Centrado en la arquitectura: En el contexto del ciclo de vida del software, significa que la arquitectura de un sistema se usa como un artefacto primordial para la conceptualización, construcción y transición.

Componente: Una parte física y reemplazable de un sistema que se ajusta a, y proporciona la realización de, un conjunto de interfaces.

Dirigido por los casos de uso: En el contexto del ciclo de vida del software, indica que los casos de uso se utilizan como artefacto principal para definir el comportamiento deseado para el sistema, y para comunicar este comportamiento entre las personas involucradas en el sistema. También indica que los casos de uso son la entrada principal para el análisis, diseño, implementación y pruebas del sistema, incluyendo la creación, verificación y validación de la arquitectura del sistema.

Estereotipo: Una extensión del vocabulario de UML, que permite la creación de nuevos tipos de bloques de construcción que se derivan de otros existentes pero que son específicos a un problema particular.

Kerberos: Es un protocolo de seguridad que usa una criptografía de claves simétricas para validar usuarios con los servicios de red.

Método: La implementación de una operación.

Modelo: Abstracción de un sistema cerrada semánticamente.

NTFS: (New Technology File System) Sistema de archivos diseñado específicamente para Windows NT.

NTLM: Protocolo de autenticación Desafío/Respuesta de Windows NT. Este protocolo utiliza cifrado para la transmisión segura de contraseñas.

Patrón: Solución común a un problema común de un determinado contexto.

Plataforma .NET: Es un conjunto de métodos y tecnologías de Microsoft que habilita una plataforma de desarrollo para la construcción, despliegue y operación de aplicaciones tradicionales y Servicios Web.

Sistema informático Es un sistema de información, que utiliza computadoras para almacenar los datos de la empresa, transformarlos y/o transmitirlos, dejándolos a disposición de su personal.

TI: Tecnologías de Información. Se habla de TI para referirse a la informática, las tecnologías de comunicación y cualquier otra técnica que permita manejar, comunicar, procesar la información en cualquiera de los formatos en los que pueda presentarse. También se usan las siglas TIC (Tecnologías de la Información y de las comunicaciones) o NTIC (Nuevas TIC).

UML: Lenguaje estándar para el modelado de software –lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software.

URL: *Uniform Resource Locator*, es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

XML: **Extensible Markup Language**, lenguaje de bajo nivel (a nivel de aplicación, no de programación) para intercambio de información estructurada entre diferentes plataformas, similar a HTML pero su función principal es describir datos y no mostrarlos.

REFERENCIA BIBLIOGRÁFICA

Libros

Jacobson, Ivar. [et al.]. *El proceso Unificado de Desarrollo de Software*. Ed. España: Addison Wesley, 2000.

Booch, Grady. [et al.]. *El Lenguaje Unificado de Modelado*. Ed. España: Addison Wesley, 1999.

Larman, Craig. *UML Y PATRONES: Introducción al análisis y diseño orientado a objetos*. Ed. México: Prentice Hall, 1999.

Rodríguez, Besteiro. *Desarrollo de aplicaciones .NET con Visual C#*. Ed. España: McGraw – Hill, 2002.

Páginas en Internet

MSDN. *Arquitectura de software* [En línea]
<<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art110.asp>>
[Consulta: 03 enero 2005]

Qué es UML? [En línea] <<http://www.creangel.com/uml/>> [Consulta: 10 marzo 2005]

Clikear. *Desarrollo Orientado a Objetos con UML* [En Línea]
<<http://www.clikear.com/manuales/uml/>> [Consulta: 15 marzo 2005]

Microsoft Corporation. *Información general acerca del .NET Framework* [En línea]
<<http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/vsintro7/html/vxtbsVisualStudioNETSoftwareRequirements.asp>> [Consulta: 25 septiembre 2005]

MSDN *Introducción a servicios Web XML en Visual Studio .NET* [En línea]
<<http://www.microsoft.com/spanish/msdn/articulos/archivo/120402/voices/vbtchqettingstartedwithxmlwebservicesinvisualstudionet.asp>> [Consulta: 10 enero 2006]