

**Universidad de las Ciencias Informáticas**

**Facultad 4**



**Título: Diseño del Sistema de solicitud de  
Entrada para Visitantes de Profesores Residentes**

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

**Autor: Adriana Martínez Pérez**

**Tutor: Eutimio Liusbel de la Rosa Armas**

**Asesor: Yilmavis La Rosa Sordo**

**Ciudad de la Habana, junio del 2008.**

**“Año 50 de la Revolución”**

## DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año 2008.

Adriana Martínez Pérez

Eutimio Liusbel de la Rosa Armas

\_\_\_\_\_

Firma del Autor

\_\_\_\_\_

Firma del Tutor

## DATOS DE CONTACTO

**Eutimio Liusbel de la Rosa Armas.** Lic. Ciencias de la Computación.

- ✓ Diplomado en Matemática Aplicada. (UCI)
- ✓ Postgrado en Docencia Universitaria. (UCI)
- ✓ Postgrado en Economía Política. (UCI)
- ✓ Postgrado en Inglés Básico. (UCI)
- ✓ Jefe de las asignaturas Matemática I y II en el dpto. de Ciencias Básicas de la Facultad 4 de la UCI.
- ✓ Jefe de Disciplina (Matemática I, II, III, IV; Matemática Discreta; Álgebra; Probabilidades y Estadísticas; Investigación de Operaciones) en la facultad 4 de la UCI.

**e-mail:** [eutimio@uci.cu](mailto:eutimio@uci.cu)

**Vilmavis La Rosa Sordo.** Ing. Ciencias Informáticas

- ✓ Jefe de Departamento Ingeniería y Gestión de software facultad 6.
- ✓ UCI. Grupos de Bioinformática.

**e-mail:** [vlarosa@uci.cu](mailto:vlarosa@uci.cu)

## AGRADECIMIENTOS

*Realmente el agradecimiento va mucho más allá de decir tan sólo “gracias”, para mí, es estar en deuda eterna con esas personas que te apoyaron, te dieron ánimo y cuando pensaste que sería imposible te dijeron “sí puedes”.*

*Por eso al menos quisiera reflejar aquí mi gratitud para con mis amigos, son la garantía más grande de tener siempre con quien contar, a mis profesores, a los de antes y a los de ahora, por su infinita paciencia, en especial a mi tutor Eutimio y a mi amiga Vilmavis.*

*A Keyly, el amor de mi vida, por hacer hasta lo imposible por hacerme feliz, a mi familia y a los que me quieren como si lo fueran, a Onelia, Félix y Jorgito, la otra familia que la vida me dio, por su preocupación constante. A mis padres que me dieron aliento, apoyo, confianza, y creyeron en mí.*

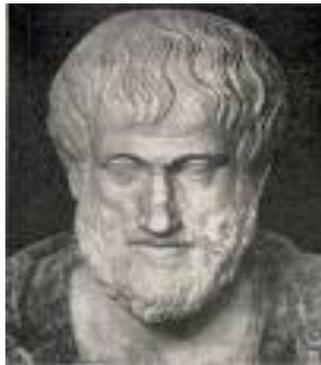
*A la Revolución y a Fidel.*

## DEDICATORIA

*A mis padres que lo son todo para mí, sepan que estarán siempre en mi corazón, su vida he sido yo y ustedes siempre serán la mía.*

*A mi bebé por su forma de amar.*

*“Los sabios tienen sobre los ignorantes las mismas ventajas que los vivos sobre los muertos, que la sabiduría es un adorno en la prosperidad y un refugio en la adversidad”*



*Aristóteles*

## **RESUMEN**

El presente trabajo tiene como tema el "Diseño del Sistema de Solicitud de Entrada para Visitantes de Profesores Residentes" en la Universidad de las Ciencias Informáticas (UCI). El mismo surge a raíz de que en el propio centro se tiene en ejecución un servicio, a través del cual se controlan las solicitudes de entrada (pases) para visitantes, realizadas por los profesores residentes. Sin embargo, el mismo se lleva a cabo de forma manual y tras el aumento considerable del claustro de profesores, y gran parte de ellos, internos en el centro, la información a controlar ha crecido notablemente, lo que trae consigo que se convierta en un proceso poco eficiente.

Por lo anteriormente planteado, el objetivo radica en diseñar una aplicación que proporcione mayor facilidad, control u organización para este proceso. La misma deberá ser una Aplicación Web que permita de una forma más rápida y sencilla, la reservación de entrada y opcionalmente de alimentación, además de brindar información relacionada con las reservaciones de forma general, para lo cual se propone una estructura que servirá de base para cumplir con dichas funcionalidades.

En este documento se refleja todo el desarrollo del trabajo, especificando la aplicación de metodologías, herramientas y tecnologías actuales a considerar, las que garantizarán el diseño de un software con calidad.

## **PALABRAS CLAVE**

Diseño, sistema, solicitud, visitantes, profesores residentes, pase, reservación.

# TABLA DE CONTENIDOS

<b>AGRADECIMIENTOS</b> .....	I
<b>DEDICATORIA</b> .....	II
<b>RESUMEN</b> .....	IV
<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	5
<b>1.1 Introducción</b> .....	5
<b>1.2 El diseño como espacio de información</b> .....	5
<b>1.3 Sistemas diseñados para la realización de reservaciones</b> .....	5
<b>1.4 Tendencias y Tecnologías actuales a considerar</b> .....	7
1.4.1 Aplicación Web.....	7
1.4.2 Orientación a Objetos.....	8
1.4.3 Servidor Web propuesto.....	9
1.4.4 Arquitectura de Aplicación .....	10
1.4.5 Patrón de Arquitectura.....	12
1.4.6 Lenguajes de programación para el diseño del sistema.....	12
1.4.7 Sistema de Gestión de Bases de Datos (SGBD) propuesto.....	14
1.4.8 Metodología de Desarrollo.....	16
1.4.9 Lenguaje de modelado.....	21
1.4.10 Herramienta para el Modelado .....	23
<b>1.5 Conclusiones</b> .....	24
<b>CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA</b> .....	25
<b>2.1 Introducción</b> .....	25
<b>2.2 Objeto de Estudio</b> .....	25
<b>2.3 Situación Problémica</b> .....	25
<b>2.4 Información que se maneja</b> .....	26
<b>2.5 Objeto de Automatización</b> .....	27
<b>2.6 Propuesta de Solución</b> .....	28
<b>2.7 Reglas del Negocio</b> .....	28
<b>2.8 Modelo del Negocio</b> .....	29
2.8.1 Actores del Negocio.....	29
2.8.2 Trabajadores del Negocio .....	30
2.8.3 Diagrama de Casos de uso del Negocio .....	31
2.8.4 Realización de los Casos de Uso.....	32
<b>2.9 Especificación de los Requisitos de Software</b> .....	46
2.9.1 Requerimientos Funcionales .....	46
2.9.2 Requerimientos No Funcionales .....	48
<b>2.10 Modelado del sistema</b> .....	50
2.10.1 Actores del Sistema.....	50

2.10.2 Diagrama de Casos de Uso del Sistema.....	51
2.10.3 Listado de los Casos de uso del Sistema .....	52
2.10.4 Descripción Textual de los Casos de Uso Expandidos .....	57
<b>2.11 Conclusiones</b> .....	94
<b>CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA</b> .....	95
<b>3.1 Introducción</b> .....	95
<b>3.2 Análisis del Sistema</b> .....	95
3.2.1 Modelo de Análisis.....	95
3.2.2 Diagrama de clases del análisis.....	96
<b>3.3 Diseño del Sistema</b> .....	96
3.3.1 Patrones de Diseño .....	96
3.3.2 Diagramas de interacción .....	97
3.3.4 Diagrama de Clases de Diseño Web.....	100
3.3.5 Descripción de las clases utilizadas.....	101
<b>3.4 Diseño de la Base de Datos</b> .....	101
3.4.1 Modelo Entidad Relación de la Base de Datos.....	102
3.4.2 Descripción de las tablas .....	102
<b>3.5 Principios de diseño</b> .....	106
<b>3.6 Interfaz de Usuario</b> .....	106
<b>3.7 Tratamiento de Errores</b> .....	107
<b>3.8 Seguridad</b> .....	107
<b>3.9 Conclusiones</b> .....	108
<b>CONCLUSIONES</b> .....	109
<b>RECOMENDACIONES</b> .....	110
<b>BIBLIOGRAFÍA</b> .....	111
<b>REFERENCIAS BIBLIOGRÁFICAS</b> .....	113
<b>GLOSARIO DE TÉRMINOS</b> .....	114
<b>ANEXOS</b> .....	116

# INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones (TIC), constituyen una vía para desarrollar y distribuir información mediante la utilización de medios informáticos, ampliando de esta forma las posibilidades de desarrollo social.

Su rápida evolución a nivel mundial ha creado una plataforma para el libre flujo de información, ideas y conocimientos en todo el planeta y ha modificado de manera sustancial e irreversible la forma en que éste funciona, influenciando fuertemente la estructura y dinámica de los procesos económicos y sociales.

En Cuba, se han incorporado los diferentes sectores de la sociedad a la revolución de las TIC, lo que llamamos la “informatización de la sociedad”, satisfaciendo así las necesidades de información y conocimiento de todas las personas y esferas en el país. Sus esfuerzos por patrocinar su desarrollo, han conducido a la creación de un núcleo de fuerza de trabajo altamente entrenado y capacitado, único en el mundo subdesarrollado. Actualmente en nuestro país se están convirtiendo en un aspecto fundamental de las estrategias cubanas para el avance socioeconómico.

Como una de las sedes principales que posee nuestro país, para el continuo desarrollo e implantación de las TIC y como institución especializada para el aprendizaje de las mismas, está la Universidad de las Ciencias Informáticas (UCI), donde más que nuevas herramientas constituyen un gran aliado. Cada día se despliegan constantemente estrategias para su uso en todas las direcciones y se diseñan e informatizan cada vez más, disímiles procesos y actividades.

Uno de estos procesos, llevado a cabo actualmente por la Dirección de Residencia de la Universidad, permite que los profesores residentes en la misma, tengan la posibilidad de solicitar la entrada de personal ajeno al centro en rol de visitante con previo autorizo. Este proceso constituye una gran ventaja pues permite controlar y organizar la información relacionada con la solicitud de entrada y con la opcional solicitud de alimentación, además de la estancia transitoria de visitantes.

A pesar de lo anteriormente planteado este proceso llega a ser un trabajo que se puede categorizar de engorroso y algo complejo, debido a diferentes factores, dentro de los cuales podemos destacar:

- ✓ El aumento considerable del claustro de profesores de la Universidad, donde muchos de ellos son internos, por lo que residen a tiempo completo en la misma y constituyen los participantes iniciadores de esta actividad.
- ✓ La gran cantidad de información que se tramita a raíz de las peticiones de este servicio diariamente.
- ✓ La particularidad de que son personas en específico quienes están capacitadas para autorizar tanto la solicitud de pase como de alimentación, por lo que no es muy difícil imaginar que todo esto provoca un cúmulo de información pendiente que requiere determinada atención.
- ✓ Los numerosos pasos que debe realizar el profesor residente para lograr obtener la autorización de entrada y el derecho a la alimentación, los que se desarrollan incluso, de forma independiente y en distintos lugares.
- ✓ La lentitud y las pocas facilidades que tienen lugar, como secuela principalmente de que todo este control se realiza de forma manual, lo que evita una mayor agilidad y rapidez, siendo esta dificultad el núcleo problemático fundamental dentro de este entorno.

Luego de evidenciar la necesidad de encontrar una solución que satisfaga las necesidades y expectativas de los clientes y que responda ante estas dificultades, que de alguna forma entorpecen el buen funcionamiento de los servicios en la Universidad de forma general, todos los esfuerzos estarán encaminados a resolver el **problema** que queda formulado con la siguiente interrogante:

¿Cómo diseñar una aplicación que permita facilitar la gestión y el control de la información del Servicio de Solicitud de Entrada para Visitantes?

Según las condiciones dadas y como elemento imprescindible para planificar la investigación, el **objeto de estudio** radica en el proceso de gestión y control de la información del Servicio de Solicitud de Entrada para Visitantes.

Para precisar el objeto de la investigación se determina como **campo de acción** el Servicio de Solicitud de Entrada para Visitantes presente en la UCI.

Con la finalidad de orientar y servir de guía al proceso de investigación se plantea la siguiente **hipótesis**:

Se logrará realizar el diseño del sistema de solicitud de entrada para visitantes de profesores residentes en la UCI.

Como **objetivo general** proponemos diseñar un sistema informático, que brinde la posibilidad de reservar pase para visitantes automáticamente.

Para resolver el problema planteado en la investigación definimos los siguientes **objetivos específicos**:

- ✓ Definir y modelar los procesos de negocio existentes y analizar la posibilidad de proponer mejoras a los mismos.
- ✓ Llevar a cabo un levantamiento de requisitos con los que el software deberá cumplir.
- ✓ Realizar el análisis y diseño del sistema.
- ✓ Diseñar una Base de Datos que soporte las funcionalidades necesarias a cumplir por el sistema.

Para lograr cumplir con los objetivos específicos anteriormente definidos nos hemos trazados las siguientes **tareas de investigación**:

- ✓ Estudio y análisis profundo de cómo funciona el servicio de solicitud de entrada para visitantes de profesores residentes presente en la UCI.
- ✓ Realizar un análisis de las tecnologías, herramientas y metodologías factibles en la actualidad para llevar a cabo las actividades de diseño.
- ✓ Diseñar un sistema cuya funcionalidad principal sea realizar reservaciones de entrada para visitantes, aplicando la Metodología RUP como marco de trabajo.
- ✓ Crear un informe donde se plasme todo lo tratado durante el desarrollo de la problemática planteada.

Este documento está estructurado de forma general por tres capítulos.

En el **Capítulo 1** se expone la **Fundamentación Teórica** del trabajo al tratar el diseño como espacio de información, caracterizar algunos de los sistemas informáticos mediante los cuales se lleva a cabo una reservación y realizar un estudio de las tecnologías, herramientas y metodologías a utilizar en el diseño del sistema.

En el **Capítulo 2** se reflejan las **Características del Sistema** enfocándose hacia la propuesta de solución principalmente a través de la modelación del negocio y la captura de los requisitos funcionales y no funcionales con que el sistema deberá cumplir, logrando así la modelación del mismo en términos de casos de uso.

En el **Capítulo 3** se realizan el **Análisis y Diseño del Sistema** al crear los diagramas de clases tanto del análisis como del diseño, los diagramas de secuencia, el Diagrama Entidad Relación de la Base de Datos y la Descripción de las tablas, entre otros aspectos importantes para el diseño del sistema.

# CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

## 1.1 Introducción

La idea de este capítulo es lograr transmitir los aspectos teóricos fundamentales con los que se contarán para el desarrollo del trabajo. Se reflexiona sobre el diseño como espacio de información lo que unido a la evolución de los procesos de reservación, permite luego exponer las principales características del diseño de algunos sistemas que dan la posibilidad de reservar automáticamente a nivel nacional e internacional. Además, con la ayuda de evaluaciones comparativas, se seleccionan los lenguajes, herramientas y tecnologías, con los que será posible llevar a cabo el diseño del sistema, guiado en todo momento por la metodología de desarrollo también definida.

## 1.2 El diseño como espacio de información

Los espacios de información se encuentran de forma permanente en nuestro medio de desarrollo, por lo que es sumamente vital conocer cómo es mejor comunicar y acceder a la información que se nos presenta. En este caso, entender la estructura de un espacio de información para un sitio Web pudiera reducirse a las siguientes preguntas:

- ¿Cuál es la estructura de la información?
- ¿Cómo represento visualmente esa estructura?
- ¿Qué relaciones existen entre las páginas Web?
- ¿Cómo se representan estas relaciones?

Las mismas son erradicadas con el diseño del sistema ya que este debe comunicar "la visión general" del mismo a las personas interesadas, prodigando detalles suficientes para que sea de utilidad en el desarrollo. Debe además transmitir visualmente la información estructural y funcional acerca del sitio Web, dígame relación entre las páginas, funcionalidad e interactividad. Su meta es evitar abstracciones innecesarias en los diagramas, ya que el contenido de los mismos debe estar muy cerca de lo que será observado en un futuro.

## 1.3 Sistemas diseñados para la realización de reservaciones

Debido a que la actividad central que se llevará a cabo en el diseño del sistema será realizar una reservación, se ha considerado muy conveniente abordar algunos sistemas en los que se aplica dicho evento, no siendo algo trivial caracterizar antes el proceso de evolución de las reservaciones.

### **Evolución de las reservaciones**

Una reservación siempre ha constituido una acción frecuente en todos los sectores de una sociedad, acarreando generalmente una buena cantidad de pasos para su consolidación. Hace diez años atrás, para poder solicitar una reservación, se debía entrar en contacto personal con el proceso de negocio existente. Con el desarrollo de las TIC, ha cambiado positivamente esta metodología, ya que se han diseñado sistemas que le permiten a los usuarios efectuar reservas online, llegando a ser muy populares y eficientes, dada principalmente su facilidad de uso y comodidad.

A nivel internacional, por ejemplo, se encuentra **SUBIVAL**:

El Sistema Integrado de Bibliotecas de la Universidad de Valparaíso, Chile, es la unidad donde se planifica, coordina y gestiona la adquisición, administración, promoción y distribución de los servicios y recursos de información dentro de la misma. Para ello incluye un sistema cuyo diseño les permite a los usuarios realizar reservaciones de equipos, o sea, computadoras, y salas, especificando lugar, fecha y horario en que desea sea efectiva la reservación. Además expone un método de seguridad donde los usuarios deben ingresar identificadores para acceder al sistema.

Por su parte nuestro país cuenta con **GERONTOGER**:

La Sociedad Cubana de Gerontología y Geriátrica auspicia congresos y seminarios internacionales, para lo cual cuenta con un sistema que se encarga de procesar directamente las reservaciones de alojamiento, traslados de entrada y salida al aeropuerto internacional así como otros servicios asociados. El proceso de reservación es sencillo y rápido. Solo se requiere transitar por 5 pasos para recibir la confirmación final.

1. **Selección del Hotel y Fechas:** Se podrá elegir entre la lista de hoteles asociados al evento, seleccionando de forma personalizada los datos de llegada y salida según las conexiones aéreas. Es posible reservar simultáneamente hasta 3 habitaciones dobles para 6 participantes en el evento.
2. **Revisión y Resumen:** Visualización del precio final a pagar por pasajero, así como el importe total y el detalle del paquete de alojamiento seleccionado.
3. **Información Personal:** Se deberá introducir información personal (Nombres, Apellidos, Dirección electrónica, etc.)

4. **Pago:** Se deberá introducir los datos de la tarjeta de crédito directamente en el centro de cobro. Visualizando automáticamente la notificación del estado de la operación. Se utilizan servicios SSL, como protocolos criptográficos que proporcionan comunicaciones seguras en Internet, que garantiza la total privacidad de sus datos personales.
5. **Confirmación:** El usuario recibirá instantáneamente en su buzón de correo, la confirmación electrónica de la reserva realizada con detalles de cada uno de los participantes y servicios incluidos.

Luego, uno de los centros de nuestro país donde se utilizan sistemas de reservación es la UCI, la misma cuenta con varias aplicaciones de este tipo, entre ellas se encuentra el **Sistema de Reservación de Transporte para estudiantes:**

El Sistema de Reservación de Transporte para estudiantes de la UCI, les brinda la posibilidad a todos los estudiantes de solicitar un pase de fin de semana que garantiza el transporte, de ida y de vuelta. El mismo cuenta con toda la información necesaria para que los usuarios conozcan las características de una solicitud de pase, lo que incluye un Listado de Rutas y un Mapa con los puntos de salida de la UCI, así como un formulario para la autenticación, ya que este procedimiento es fundamental en la reservación. Tras dicho evento se muestra información pertinente sobre el usuario autenticado y el sistema le muestra los datos a llenar, esencialmente el destino, y en caso de no tener residencia en Ciudad de La Habana, el familiar al que visitará.

### 1.4 Tendencias y Tecnologías actuales a considerar

Ante todo, es importante decir que la selección de las tendencias y tecnologías se realiza en base a las necesidades propias de la situación, o sea, que no existe la mejor tecnología de forma general, simplemente, se escoge la que mejor cumpla con los requisitos y necesidades.

Es válido también señalar que en la selección de las mismas se han descrito algunas, que a pesar de no utilizarse directamente, se reconocieron como muy adecuadas para el desarrollo del sistema y en estrecha relación con las que sí se utilizan, dígame Servidor Web y Sistema Gestor de Base de Datos.

#### 1.4.1 Aplicación Web

Hace algunos años el mundo giraba entorno a los sistemas Desktops o locales, ya que estos mantienen un contacto permanente entre los procesos internos del programa y lo que sucede en la interfaz de usuario. Debido a la dificultad de que necesitan que cada aplicación tenga su propio

programa cliente y su interfaz de usuario, lo que significa que tendrían que ser instalados separadamente en cada estación de trabajo, ha surgido una gran tendencia por las tecnologías Web.

La WEB se ha declarado el medio y la forma más potente y moderna de ofrecer información. Permite una manera más organizada de acceder a la disponibilidad de la misma, presentando una interfaz amigable con el usuario. Las aplicaciones web son populares debido a la facilidad para actualizarlas y mantenerlas sin su distribución e instalación en miles de computadoras individuales.

En **TICS Consulting** (Servicios de Consultoría Avanzadas en Tecnologías de la Información, Comunicaciones y Seguridad Informática) sugieren muy fuertemente el desarrollo de aplicaciones Web, utilizando preferentemente Software Libre, ya que permiten realizar todas las tareas necesarias hoy en día en un entorno empresarial.

### 1.4.2 Orientación a Objetos

Un paradigma de programación es un modelo básico de diseño y desarrollo de programas que permite determinar la estructura de los mismos con la ayuda de modelos conceptuales. Existen múltiples formas de construcción de sistemas software, entre ellas la programación estructurada, funcional, lógica, orientada a objetos, entre otras. Sin embargo la tecnología orientada a objetos es el paradigma de programación más utilizado en la actualidad ya que su consistente base teórica y la amplia gama de herramientas que permiten crear código a través de diseños orientados a objetos la convierten en la alternativa más adecuada para el desarrollo de aplicaciones. Además fomenta la reutilización y extensión del código, permite crear sistemas más complejos y facilita el mantenimiento del software. Lo interesante es que proporciona conceptos con los cuales se modela y representa el mundo real tan fielmente como sea posible.

La importancia de hacer esta referencia radica en que para el desarrollo de software orientado a objetos no basta usar un lenguaje que lo implemente, también se necesitará realizar un análisis y diseño orientado a objetos, ya que el modelado visual es fundamental para el propio desarrollo, siendo este argumento esencial para valorar la importancia de este trabajo, ya que un diseño orientado a objetos, maximiza la modularidad y la encapsulación, ya que el sistema se descompone en objetos con responsabilidades claramente especificadas, la extensibilidad pues da la posibilidad de ampliar la funcionalidad de la aplicación de manera sencilla y la reusabilidad ya que permite reutilizar parte del código para el desarrollo de una aplicación similar.

### 1.4.3 Servidor Web propuesto

Un servidor web no es más que un programa que se ejecuta de forma continua en un ordenador (también se utiliza el término para referirse al ordenador que lo ejecuta), manteniéndose a la espera de peticiones por parte de un cliente (un navegador web) y que contesta a estas peticiones de forma adecuada, mediante una página web que será mostrada en el navegador o mostrando el mensaje correspondiente si se detectó algún error. Algunos servidores web importantes son IIS (Internet Information Services) y Apache.

#### **IIS (Internet Information Services)**

Internet Information Services es una serie de servicios para los ordenadores que funcionan con Windows y convierte a un ordenador en un servidor. Su orientación a componentes es completa y bien implementada. Es fácil de instalar, configurar y su mantenimiento es trivial. Uno de los temas en los que Microsoft todavía tiene que avanzar es en la seguridad, el abarcar tantos conceptos, tecnologías y componentes hace que IIS sea más susceptible de tener agujeros de seguridad, lo que lo hace más simple y limitado.

#### **Apache**

Apache es uno de los servidores web más populares del mercado, y el más utilizado actualmente. Su configurabilidad, robustez y estabilidad hacen que este sea confiable y se caracterice por una gran calidad profesional. La popularidad de este software libre grandemente reconocido en muchos ámbitos empresariales y tecnológicos, se debe principalmente a las siguientes razones:

- ✓ Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- ✓ Es una tecnología gratuita de código fuente abierto.
- ✓ Es un servidor altamente configurable. Sus capacidades son muy sencillas de ampliar.
- ✓ Te permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.

Teniendo en cuenta las características antes mencionadas, se propone como servidor web para probar los desarrollos de forma local Apache. Fundamentalmente gracias a su presentación como software Libre y su potente seguridad, ya que las webs que se encuentran en servidores que corren bajo el software de Microsoft Internet Information Services (IIS) tienen el doble de posibilidades de hospedar malware que las que funcionan con Apache.

### 1.4.4 Arquitectura de Aplicación

Una arquitectura es un entramado de componentes funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente. [4]

Debemos señalar que para seleccionar el modelo de una arquitectura, hay que partir del contexto tecnológico y organizativo del momento. Teniendo en cuenta además que la administración de sistemas y la definición de dónde ubicar los datos, deben ser seriamente consideradas y evaluadas, ya que las facilidades de administración de sistemas son un requisito indispensable para una implantación exitosa.

Cuando se habla de aplicaciones Web tenemos que pensar en la mejor arquitectura para el control e intercambio de información a través de la red, siendo la Arquitectura Cliente / Servidor la más significativa.

#### Arquitectura Cliente/Servidor

La arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina **cliente** al proceso que inicia el diálogo o solicita los recursos y **servidor** al proceso que responde a las solicitudes. [2]

Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste le responde proporcionándolo. Normalmente, pero no necesariamente, el cliente y el servidor están ubicados en distintos procesadores. Los clientes se suelen situar en ordenadores personales y/o estaciones de trabajo y los servidores en procesadores departamentales o de grupo.

Esta arquitectura se refiere a la forma en la que es diseñada la aplicación tanto física como lógicamente. En el diseño físico se especifica exactamente donde se encontrarán las piezas de la aplicación (discos, ejecutables, cable de red y computadoras). En el diseño lógico o conceptual se especifica la estructura de la aplicación y sus componentes, sin tomar en cuenta dónde se localizará el software, hardware e infraestructura.

Entre las principales características se pueden destacar las siguientes:

- ✓ El servidor presenta a todos sus clientes una interfaz única y bien definida.
- ✓ El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se

encuentra, ni de su sistema operativo.

- ✓ Un servidor da servicio a múltiples clientes de forma concurrente.
- ✓ Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los clientes o de los servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.

Como ventajas podemos destacar:

- ✓ Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema.
- ✓ Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado.
- ✓ Facilita la integración entre sistemas diferentes y comparte información, además su mantenimiento y el desarrollo de aplicaciones es rápido.
- ✓ Favorece el uso de interfaces gráficas interactivas. Los sistemas construidos bajo este esquema tienen mayor interacción intuitiva con el usuario.
- ✓ El acceso a la información se realiza de forma más ágil y al estar almacenada en el servidor existe un mejor control de la seguridad.

### **Arquitectura de tres-capas**

Una disposición muy común de la Arquitectura Cliente/Servidor son los sistemas multicapa, siendo su principal objetivo separar la lógica del negocio de la lógica de diseño ya que al usuario final solo le interesa la visualización de la información en el sistema, no la forma en que se gestiona. El diseño más utilizado actualmente es el diseño en tres capas.

Los diseños en tres capas (presentación, negocio y acceso a datos) han probado sus ventajas a lo largo del tiempo, a partir de que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio solo se ataca al nivel requerido. Además todos los datos se almacenan en los servidores, así que tienen una mejor capacidad del control de la seguridad.

A cada capa se le asigna una misión específica que permite que el diseño de la arquitectura sea escalable y que pueda ampliarse con facilidad si es necesario.

- ✓ Capa de presentación o aplicaciones cliente: Utilizada por la aplicación para interactuar con el usuario; el ingreso y egreso de datos se realiza a través de esta capa.

- ✓ Capa de negocios: Encargada de realizar el procesamiento funcional.
- ✓ Capa de acceso a datos: Utilizada para el acceso, ingreso y mantenimiento de datos al sistema.

### 1.4.5 Patrón de Arquitectura

Un patrón de arquitectura de software describe un problema particular y recurrente del diseño, que surge en un contexto específico, y presenta un esquema genérico y probado de su solución. Los patrones de arquitectura expresan el esquema fundamental de organización para sistemas de software. Proveen un conjunto de subsistemas predefinidos; especifican sus responsabilidades e incluyen reglas y guías para organizar las relaciones entre ellos.

#### Patrón Arquitectura en Capas

El Patrón Arquitectura en Capas permite la reutilización de las capas y del código, donde la solución de un proceso puede aplicarse a otros similares. Además define cómo organizar el modelo de diseño en capas, las que pueden estar físicamente distribuidas, lo cual quiere decir que los componentes de una capa sólo pueden hacer referencia a componentes en capas inmediatamente inferiores. La programación por capas es una técnica de ingeniería de software propia de la programación por Objetos, cuya ventaja principal es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido, a esto se le llama modularidad, y al implementarlo se asegura un trabajo de forma ordenada y separada.

### 1.4.6 Lenguajes de programación para el diseño del sistema

Un lenguaje de programación es aquel elemento dentro de la informática que nos permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis, y está a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes.

Los lenguajes de programación para la Web son aquellos lenguajes que permiten que las aplicaciones sean dinámicas, que permitan la interacción con el usuario y la personalización de la información, estos lenguajes pueden ser del lado del cliente o del lado del servidor.

Los lenguajes de lado del servidor, son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Actualmente se desarrollan tecnologías del lado del servidor para coordinar los lenguajes del lado del cliente, los que se utilizan para añadir más funcionalidad a través de elementos dinámicos a la interfaz

de usuario, pueden ser directamente "digeridos" por el navegador y no necesitan un pre-tratamiento. Actualmente existen diferentes lenguajes de programación para desarrollar en la Web, tanto de lado del servidor como de lado del cliente, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas, entre los que destacan: **HTML** (HyperText Markup Language), **PHP** (Personal Home Page), **JavaScript**, **ASP** (Active Server Pages), **JSP** (Java Server Pages), entre otros.

### Lenguajes del lado del cliente

#### **HTML (HyperText Markup Language)**

HTML es un lenguaje de marcado diseñado para estructurar textos de manera agradable y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web. El HTML se ha convertido en uno de los formatos más populares y fáciles de aprender que existen para la elaboración de documentos para la Web, además, su surgimiento constituyó la base o componente más importante para los mismos. Es admitido por todos los exploradores, posee archivos pequeños y despliegue rápido. No es un Lenguaje de Programación, sin embargo, permite incluirle código en lenguajes de programación, bajo ciertos criterios, extendiendo así su capacidad y funcionalidad.

#### **JavaScript**

JavaScript es un lenguaje que no requiere compilación, multiplataforma, su código se incluye directamente en el mismo documento y es usado para el desarrollo de aplicaciones cliente-servidor en páginas HTML. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado ya que permite de forma eficiente validar formularios, además de ser un lenguaje simple, de fácil aprendizaje y dinámico, ya que lo que hace es responder a eventos locales, producidos por el usuario, y reaccionar ante ellos en tiempo real.

### Lenguajes del lado del servidor

#### **PHP (Hypertext Pre-processor)**

PHP es un lenguaje interpretado del lado del servidor, utilizado para la generación de páginas web dinámicas embebido en código HTML, lo que constituye su meta principal. Para su funcionamiento necesita tener instalado un servidor Web, como lo es Apache.

#### **PHP posee ventajas como:**

- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos y es muy fácil

de aprender, estas constituyen sus características más destacadas.

- ✓ Se caracteriza por ser un lenguaje muy rápido.
- ✓ Tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos por lo que se considera un lenguaje multiplataforma.
- ✓ Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad
- ✓ Posee una amplia documentación donde se destaca la explicación de funciones con la ayuda de ejemplos.
- ✓ Permite las técnicas de Programación Orientada a Objetos.
- ✓ Biblioteca nativa de funciones sumamente amplia e incluida

Por la facilidad de uso, aprendizaje y utilización a nivel mundial se escogieron php y JavaScript, contando además con las ventajas anteriormente descritas siendo la más significativa su desarrollo como Software Libre. Utilizando también, HTML principalmente en el diseño ya que posibilita la inclusión de ambos en su declaración de código.

### **1.4.7 Sistema de Gestión de Bases de Datos (SGBD) propuesto.**

Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada ó estructurada. [3]

Un Sistema de gestión de base de datos es un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Está compuesto por un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta. Tiene como propósito general manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información. Algunos de los SGBD comúnmente utilizados a nivel mundial son: Oracle, MySQL, PostgreSQL, entre otros.

#### **Oracle**

Una Base de Datos Relacional es un conjunto de una o más tablas estructuradas en registros (líneas) y campos (columnas), que se vinculan entre sí por un campo en común, lo que permite velocidad y flexibilidad.

Oracle es una herramienta para la gestión de bases de datos. Es un producto vendido a nivel mundial, pero la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales. Es el mayor y más usado Sistema Manejador de Base de Dato Relacional (RDBMS)

en el mundo. La Corporación Oracle lo ofrece como un producto incorporado a la línea de producción. Además incluye cuatro generaciones de desarrollo de aplicación, herramientas de reportes y utilitarios. Garantiza la autenticidad apropiada de los usuarios y la privacidad e integridad de los datos, permite manejar la asignación de privilegios, monitorear las operaciones de la base de datos a lo largo de toda la empresa y su arquitectura ofrece escalabilidad para soportar un gran número de usuarios y cargas de trabajo de alto volumen de transacciones.

### **PostgreSQL**

El término Base de Datos Objeto Relacional (BDOR) se usa para describir una base de datos que ha evolucionado desde el modelo relacional hacia otra más amplia que incorpora conceptos del paradigma orientado a objetos. Por tanto, un Sistema de Gestión Objeto-Relacional (SGBDOR) contiene ambas tecnologías: relacional y de objetos.

PostgreSQL es un avanzado sistema de bases de datos Objeto Relacional de código abierto, estable, de alto rendimiento y con gran flexibilidad. Permite mantener la integridad de los datos y puede manejar múltiples conexiones concurrentes de los clientes, característica muy significativa de su funcionamiento. Cuenta con una arquitectura que se caracteriza por su confiabilidad y escalabilidad y permite una manipulación potente, flexible y eficiente de la información. Posee una interfaz amigable y confiable que guía a los usuarios con menor experiencia a través del complejo proceso de creación haciendo todo más veloz y dinámico. Es muy portable, por lo que se puede ejecutar en la gran mayoría de sistemas operativos existentes en la actualidad y funciona perfectamente con grandes cantidades de datos.

### **MySQL**

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Posee gran velocidad, facilidad de uso e infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación. Cuenta con una gran portabilidad entre sistemas y gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos. Se caracteriza por su bajo consumo lo que lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema, facilidad de configuración e instalación.

Se determinó la utilización de PostgreSQL como SGBD, partiendo de su desarrollo como software libre, luego gracias a que cuenta con grandes facilidades para el acceso concurrente de usuarios,

característica que seguramente se destacará en el proceso de reservación.

### 1.4.8 Metodología de Desarrollo

Una Metodología de desarrollo de software es un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. En ella se va indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben tener. Por otro lado detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

Todo desarrollo de software es riesgoso y difícil de controlar, por lo que debemos llevar una metodología de por medio, para poder obtener la satisfacción con el resultado tanto de los clientes como de los propios desarrolladores. El uso de la misma garantiza determinadas características de gran importancia en los sistemas, dentro de ellas la calidad, es el factor primordial tanto para el cliente como para los desarrolladores.

También está el tiempo, que es un factor crítico que afecta todo producto, el cual incide con mayor fuerza cuando no se hace un análisis profundo o se desconoce en detalles el sistema que se desea realizar, cuando el trabajo en equipo no está bien organizado o se realiza una mala planificación de las tareas a llevar a cabo.

Si importante es utilizar una metodología de desarrollo de software, lo es más, utilizar la adecuada, la más apropiada para nuestro caso y la que más se adapta a nuestro medio. Con una buena elección de la metodología de desarrollo de software, garantizamos conocer con anterioridad cual será el resultado a obtener, detectaremos a tiempo los errores que pudieran surgir en el camino, controlaremos la introducción de nuevas herramientas y los cambios en la organización.

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte tenemos aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software, dando mayor valor al individuo y a la colaboración con el cliente.

Por lo que existen varias metodologías clasificadas y enfocadas en:

- ✓ Metodologías ágiles
- ✓ Metodologías Tradicionales

### **Metodologías ágiles**

Las Metodologías Ágiles basan su fundamento en la adaptabilidad de los procesos de desarrollo, promueve la formalización de procesos adaptables y tienen como principal característica la habilidad de responder al cambio.

El ejemplo más destacado y exitoso de la metodología ágil, es la programación extrema o eXtreme Programming (XP) como enfoque de la ingeniería de software. Como metodología ágil se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Se puede considerar la programación extrema como la adopción de las mejores metodologías de desarrollo de acuerdo a lo que se pretenda llevar a cabo con el proyecto aplicándolo de manera dinámica durante el ciclo de vida del software.

### **Metodologías Tradicionales**

Por su parte las Metodologías Tradicionales se caracterizan por exponer procesos basados en planeación exhaustiva.

Esta planeación se realiza esperando que el resultado de cada proceso sea determinante y predecible. Se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán.

Están caracterizadas por:

- ✓ Ser rígidas y dirigidas por la documentación que se genera en cada una de las actividades desarrolladas.
- ✓ Ofrecer cierta resistencia a los cambios.
- ✓ Constituir un proceso mucho más controlado, con numerosas políticas y normas
- ✓ Más artefactos
- ✓ Más roles
- ✓ La arquitectura del software es esencial y se expresa mediante modelos.

El ejemplo más representativo e importante de este tipo de Metodologías es El Proceso Unificado de Desarrollo Software o Proceso Unificado de Rational (RUP), ya que los dos nombres suelen utilizarse para referirse a un mismo concepto.

### **Proceso Unificado de Rational (RUP)**

RUP constituye un marco de trabajo para el desarrollo de procesos que habilita el desarrollo de software de una manera organizada, con una asignación precisa de responsabilidades entre los miembros de un equipo, y con un enfoque rigurosamente centrado en las necesidades de los usuarios. Además hace el proceso práctico brindando amplias guías, plantillas y ejemplos para todas las actividades críticas del desarrollo.

La metodología RUP, está basada en 5 principios claves:

#### **1. Adaptar el proceso**

El proceso deberá adaptarse a las características propias del proyecto u organización. El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.

#### **2. Balancear prioridades**

Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.

#### **3. Demostrar valor iterativamente**

Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.

#### **4. Elevar el nivel de abstracción**

Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software. Esto previene a los ingenieros de software ir directamente de los requisitos a la codificación de software a la medida del cliente. Un nivel alto de abstracción también permite discusiones sobre diversos niveles arquitectónicos.

## 5. Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción.

RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al concluir cada uno de ellos, cada ciclo se divide en **fases** que finalizan con un hito donde se debe tomar una decisión importante, las mismas son:

- ✓ **Conceptualización (Concepción o Inicio):** Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- ✓ **Elaboración:** Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.
- ✓ **Construcción:** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene 1 o varios release del producto que han pasado las pruebas. Se ponen estos release a consideración de un subconjunto de usuarios.
- ✓ **Transición:** El release ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores.

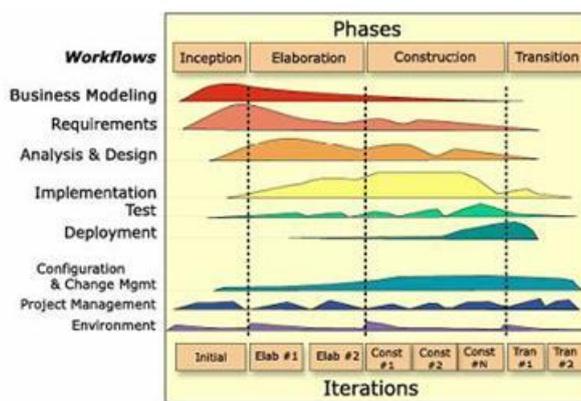


Figura 1.1 RUP en dos dimensiones

Sus principales características son:

- ✓ Dirigido por los casos de uso

En el Proceso Unificado los casos de uso se utilizan para capturar los requisitos funcionales y para

definir los contenidos de las iteraciones. La idea es que cada iteración tome un conjunto de casos de uso o escenarios y desarrolle todo el camino a través de las distintas disciplinas: diseño, implementación, prueba, etc.

- ✓ Centrado en la arquitectura

El Proceso Unificado asume que no existe un modelo único que cubra todos los aspectos del sistema. Por dicho motivo existen múltiples modelos y vistas que definen la arquitectura de software de un sistema.

- ✓ Iterativo e Incremental

RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración. Aunque cada iteración tiene que proponerse un incremento en el proceso de desarrollo, todas deben aportar al principal resultado de la fase en la que se desarrolla.

- ✓ Enfocado en los riesgos

El Proceso Unificado requiere que el equipo del proyecto se centre en identificar los riesgos críticos en una etapa temprana del ciclo de vida. Los resultados de cada iteración, en especial los de la fase de Elaboración, deben ser seleccionados en un orden que asegure que los riesgos principales son considerados primero.

Además plantea una forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo), pretende implementar las mejores prácticas en Ingeniería de Software con el fin de lograr crear equipos de alto rendimiento que producen proyectos más exitosos y que satisfacen las necesidades del usuario, estas son:

- ✓ Administrar requerimientos
- ✓ Usar arquitectura de componentes
- ✓ Modelar visualmente
- ✓ Verificar calidad
- ✓ Desarrollar iterativamente
- ✓ Controlar los cambios

### Los elementos de RUP son:

- ✓ **Actividades:** Son los procesos que se llegan a determinar en cada iteración.
- ✓ **Trabajadores:** Vienen a ser las personas o entes involucrados en cada proceso.
- ✓ **Artefactos:** Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Para planificar y controlar la propuesta de este trabajo se decidió utilizar la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos, Proceso Unificado de Modelado (RUP), por todas las características y facilidades, planteadas anteriormente, que aportan organización a todo el proceso de desarrollo de software, además:

- ✓ Al utilizar RUP se pueden reconocer problemas y fallos previamente a que ocurran y de esta forma prevenirlos y corregirlos, permitiendo así una definición acertada del sistema en un inicio para hacer innecesarias las reconstrucciones parciales posteriores y una mayor utilización de recursos.
- ✓ Evita desviaciones importantes respecto de los plazos de tiempo.
- ✓ Debido que realiza pruebas constantemente se puede asegurar la calidad del software.
- ✓ Actúa como un modelo que puede adaptarse a las necesidades y características de cualquier tipo de proyecto en específico ya sea grande o pequeño.
- ✓ Su desarrollo como método iterativo le permite reducir riesgos y dividir los proyectos en pequeños ciclos o iteraciones a través de cada una de las fases.

### 1.4.9 Lenguaje de modelado.

Un modelo no es más que el sistema representado en un lenguaje definido, conocido como lenguaje de modelado. El lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad es **UML** por sus siglas en inglés Unified Modeling Language (Lenguaje Unificado de Modelado).

Es un lenguaje gráfico para:

- ✓ Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- ✓ Especificar: UML permite especificar cuáles son las características de un sistema antes de su

construcción.

- ✓ Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- ✓ Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

UML es un estándar para describir un modelo, incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Uno de los objetivos principales de la creación de UML era posibilitar el intercambio de modelos entre las distintas herramientas CASE orientadas a objetos del mercado. Para ello era necesario definir una notación y semántica común. Por lo que UML es una consolidación de muchas de las notaciones y conceptos más usados orientados a objetos, con él se fusiona la notación de muchas técnicas para formar una herramienta compartida entre todos los ingenieros software que trabajan en el desarrollo orientado a objetos, ya que consta de todos los elementos y diagramas que permiten modelar los sistemas en base a este paradigma.

Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software, tal como en nuestro caso, al desarrollar el Proceso Unificado de Rational, pero no especifica en sí mismo qué metodología o proceso usar.

Su unificación le permite que sea interpretado por cualquier analista en cualquier parte del mundo y el análisis y diseño que esté notado en UML podrá ser implementado en cualquier lenguaje ya que no depende de una herramienta en específico.

Un modelo UML esta compuesto por tres clases de bloques de construcción:

- ✓ Elementos: Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.)
- ✓ Relaciones: relacionan los elementos entre sí.
- ✓ Diagramas: Son colecciones de elementos con sus relaciones.

Es de esperar que cualquier desarrollador versado en orientación a objetos conozca y use UML, con el los procesos de desarrollo de software han logrado erradicar todos los problemas relacionados con el modelado grafico. Además incluye al cliente en todas las etapas del proyecto y a todos los conceptos

que se consideran necesarios para utilizar un proceso moderno iterativo, basado en construir una sólida arquitectura para resolver requisitos dirigidos por casos de uso, lo que le provee un alto grado de compatibilidad con el proceso Unificado de Desarrollo (RUP).

### 1.4.10 Herramienta para el Modelado

Las herramientas de modelado constituyen el medio donde se modela el sistema que se desea, guiándose por una metodología y utilizando algún lenguaje de modelado. De acuerdo a estas características, las herramientas CASE son un ejemplo de las mismas, siendo estas las más utilizadas.

CASE es la automatización del software. Es una filosofía que se orienta a la mejor comprensión de los modelos de empresa, sus actividades y el desarrollo de los sistemas, ya que combina las herramientas software (aplicaciones) con las metodologías de desarrollo. Estas diversas aplicaciones informáticas están destinadas a aumentar la productividad en el desarrollo de software.

Como herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, documentación o detección de errores entre otras.

Las siguientes son las Herramientas CASE más utilizadas en el modelado soportando lenguaje UML:

#### **Rational Rose Enterprise Edition**

Rational Rose Enterprise Edition cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables.

Permite a los diseñadores modelar sus componentes e interfaces de forma individual y luego unirlos con otros componentes del proyecto. Rational Rose Enterprise es el producto más completo de la familia Rational Rose. Incluye soporte Unified Modeling Language (UML). Proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente.

Como herramienta de desarrollo basada en modelos, se integra con las bases de datos y los IDE de las principales plataformas. Proporciona modelado de datos, incluido el modelo de entidad-relación y mecanismos para la realización de ingeniería inversa partiendo del código de un programa, obteniendo información sobre su diseño.

### **Visual Paradigm**

Visual Paradigm es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software, permitiendo el modelado de procesos de negocios utilizando UML.

Es considerado como muy completo y fácil de usar. Posee soporte multiplataforma, capacidades de ingeniería directa e inversa, licencia gratuita y comercial. Además ofrece diseño centrado en casos de uso, permite la exportación e importación de imágenes, posibilitando así la asignación de nuevos estereotipos.

Ayuda a reconocer los elementos en el espacio de trabajo por el movimiento del mouse y darles la ubicación deseada. Brinda facilidades desde los propios elementos para enlazarlos entre si y sus recursos de trabajo poseen un diseño intuitivo, editados para facilitar la identificación visual, la misma se utilizará en el desarrollo de la ingeniería de Software.

### **1.5 Conclusiones**

En este capítulo se ha expuesto información indispensable que da soporte y conocimientos en pro del diseño de un sistema, que elimine las dificultades del proceso de solicitud de entrada llevado a cabo con toda la información en soporte duro, por lo que resulta difícil a la hora de gestionar la misma. Para ello se han combinado diferentes tecnologías, con la meta de lograr el diseño de una aplicación que cumpla con las necesidades de los usuarios. Php, con las grandes ventajas que presenta como principal lenguaje de programación. Se utilizará el Proceso Unificado de Desarrollo (RUP) para el análisis y diseño, que a su vez hará uso del Lenguaje Unificado de Modelado (UML) utilizando como herramienta Visual Paradigm. Concluye así este capítulo, dándole paso a los siguientes con el afán de posibilitar la búsqueda de una solución a la situación problemática.

# CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

## 2.1 Introducción

La idea de este capítulo es lograr describir los procesos de negocio relacionados con la solicitud de entrada a la universidad, llevada a cabo por los profesores residentes en la misma a favor de sus visitantes, así como definir las funcionalidades que debe permitir el sistema a diseñar, partiendo en primer lugar de la situación problemática existente. Para ello se identifican y analizan las necesidades del usuario definiendo los aspectos que serán objeto de automatización y brindando una propuesta de solución que incluye algunas reglas dentro del negocio. Además se presenta un estudio del modelo del negocio y la especificación de los requisitos funcionales y no funcionales como guía durante la modelación del sistema.

## 2.2 Objeto de Estudio

La Dirección de Residencia de Profesores de la Universidad de las Ciencias Informáticas, brinda un servicio que da la posibilidad a los profesores residentes en el centro de realizar una solicitud de entrada a la residencia de los mismos para sus visitantes.

El proceso de gestión y control de la información de este servicio de solicitud de entrada para visitantes, resulta ser el objeto de estudio de esta investigación.

Este proceso permite controlar y organizar eficientemente la entrada, estancia y salida de personal ajeno a la universidad, así como la información que se genera a raíz de una solicitud de pase, lo que resulta imprescindible y posee gran importancia para mantener registrada de forma actualizada todos los datos persistentes.

Los procesos relacionados con el control de los pases autorizados son:

- ✓ Solicitar Pase
- ✓ Solicitar Alimentos
- ✓ Solicitar Tickets
- ✓ Solicitar Aprobación de Pase Especial
- ✓ Entregar Autorización
- ✓ Solicitar Autorización de Entrada

## 2.3 Situación Problemática

Tras el aumento considerable del claustro de profesores, y gran parte de ellos, residentes en el centro,

la información a controlar durante este proceso ha crecido notablemente. Las actividades que tienen lugar brindan pocas facilidades, sobre todo dada la peculiaridad de que solo trabajadores en específico pueden autorizar los documentos más importantes que se generan en el mismo, los profesores residentes deben llevar a cabo numerosos pasos para hacer efectiva la entrada de sus familiares a la universidad y principalmente, pues todo este control se realiza de forma manual, lo que evita una mayor agilidad y rapidez, y lo convierte en un proceso engorroso y poco eficiente. Por esta razón surge la necesidad de diseñar un sistema basado en las actividades relacionadas con este proceso, siendo la mejor opción una aplicación web, que elimine las dificultades existentes y garantice un rápido procesamiento de la información.

### 2.4 Información que se maneja

- ✓ Modelo de Solicitud de Entrada (Pase): Es el documento más importante dentro del proceso, en él se registran los datos necesarios para reflejar la solicitud de entrada realizada por un profesor residente. Con su autorización por la Directora de Residencia de profesores se le permite la entrada a la universidad a visitantes.
- ✓ Registro de Visitantes: En él se registran los datos necesarios para reflejar las solicitudes de entrada realizadas por todos los profesores residentes.
- ✓ Registro de Control de Solicitud de Alimentos: En él se registran los datos necesarios para reflejar las solicitudes de alimentos realizadas por todos los profesores residentes.
- ✓ Modelo de Solicitud de Alimentos: En él se registran los datos necesarios para reflejar la solicitud de alimentos realizada por un profesor residente. Su autorización por la Asistente de Control de la Dirección de Alimentación le permite a los profesores garantizar que su visitante pueda optar por las opciones (desayuno, almuerzo y comida) que les provee la Universidad.
- ✓ Tickets: Evidencian el pago de la alimentación por parte de los profesores.
- ✓ Notificación: Correo electrónico enviado como confirmación del autorizo de un pase por más de tres días (pase especial).
- ✓ Documento de Pases Especiales: En él se registran los datos de las solicitudes aprobadas por el Vicerrector de Residencia de Profesores

- ✓ Libro de incidencia: En él se registran los datos necesarios para reflejar la entrada de personal ajeno a la Universidad.
- ✓ Carnet de Identidad: Identifica unívocamente, en este caso, a todos los visitantes que realizan una solicitud de entrada.

### 2.5 Objeto de Automatización

El objeto de automatización que se podrá llevar a cabo a partir del diseño del sistema radica en el proceso de solicitud de entrada para visitantes por parte de los profesores residentes en la universidad, de las actividades que incluye el mismo se proponen automatizar:

- ✓ Solicitar Pase

Cuando un profesor residente realiza una solicitud de entrada se le recogen los datos necesarios y se le entrega un modelo autorizado que le permite hacer efectiva dicha solicitud. En caso de ser un pase especial, el mismo debe haber sido antes aprobado por el Vicerrector de Residencia de Profesores.

- ✓ Solicitar Alimentos

Cuando un profesor residente realiza una solicitud de alimentos, debe presentar un modelo de solicitud de entrada autorizado, en caso positivo se le recogen los datos necesarios y se le entrega un modelo autorizado que le permite hacer efectiva dicha solicitud.

- ✓ Solicitar Tickets

Cuando un profesor residente realiza una solicitud de tickets, debe presentar un modelo de solicitud de alimentos autorizado, en caso positivo se archiva dicho modelo y se le entregan los tickets haciéndose efectiva la solicitud.

- ✓ Solicitar Aprobación de Pase Especial

Cuando un profesor residente necesita realizar una solicitud de entrada por un período mayor que el establecido tiene que solicitar aprobación exponiendo la causa que lo justifica.

- ✓ Entregar Autorización

Cuando un profesor residente ya tiene en sus manos un pase autorizado debe entregarlo en la Entrada

de Servicio donde será archivado hasta el momento de llegada del familiar, teniendo la posibilidad de enviar otra persona en su nombre.

### ✓ Solicitar Autorización de Entrada

Cuando un familiar requiere entrada a la universidad se verifica si está archivada alguna autorización a su nombre, y en caso positivo se hace efectiva su solicitud.

## 2.6 Propuesta de Solución

Para darle solución a la situación problemática se ha decidido diseñar un sistema automatizado en el entorno Web que mejore la calidad y el tratamiento de la información del servicio de solicitud de entrada para visitantes. Una aplicación cliente servidor, a la cual se deberá tener acceso mediante la intranet de la Universidad y para la cual además se deberá establecer una Base de Datos que almacene la información.

Inicialmente cada uno de los usuarios que desee entrar al sistema deberá autenticarse, verificándose así con su nombre de usuario y contraseña, que tenga acceso al mismo, vale señalar que se podrá acceder al sistema solamente por la cuenta del dominio UCI.

Si el usuario es un profesor residente podrá realizar una reservación de pase, con la opción de reservar alimentación, llenando los campos presentados con los datos de su visitante, modificar o cancelar una reservación previamente efectuada, y en cada caso se le comunicará al profesor residente la efectividad de su acción mediante un mensaje de información. Además podrá solicitar aprobación para pases especiales y consultar las reservaciones que ha realizado. El sistema también deberá brindar reportes sobre las reservaciones de forma general que permitirán obtener información, estadísticas y control sobre las mismas.

## 2.7 Reglas del Negocio

Un profesor debe realizar la solicitud de pase y de alimentación con 72 y 24 horas de antelación como máximo respectivamente.

Un profesor solo puede solicitar un pase por un período de tres días como máximo, de lo contrario si existe alguna causa por la que necesita contar con un mayor período de estancia para su visitante deberá plantearla al Vicerrector de Residencia quien autorizará en caso justificado la misma.

Un profesor solo puede solicitar la entrada de visitantes cuya ciudadanía sea “cubana”.

Además algunas restricciones fueron planteadas por la Dirección de la Universidad en la siguiente Nota Informativa:

“A partir del miércoles 21 de noviembre del 2007, los residentes pueden realizar solicitud para entrada de visitantes a su apartamento solo a aquellas personas consideradas dentro de los Vínculos Familiares que a continuación mencionamos:

Madre, Padre, Hermano(a), Tío(a), Primo(a), y Pareja.

Para los 5 primeros vínculos familiares, mencionados anteriormente, debe existir al menos un apellido en común y en caso de ser “Pareja” el sexo tendrá que ser opuesto al del profesor que realice la reservación.”

### 2.8 Modelo del Negocio

El Modelo del Negocio nos permite alcanzar cierto nivel de conocimientos sobre el problema en cuestión y permite obtener una visión de la organización que posibilita definir y comprender los procesos, roles y responsabilidades de la misma.

#### 2.8.1 Actores del Negocio

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

#### Descripción de los actores del negocio

Actor del Negocio	Descripción
Profesor Residente	Es aquel rol que realiza solicitudes de entrada, solicitudes de alimentos, y compra de tickets para garantizar la alimentación de su visitante. Además

	puede realizar la entrega de pases una vez autorizados en la Entrada de Servicio de la Universidad o enviar a alguien en su nombre. También puede solicitar un pase especial por un período mayor al establecido si tiene alguna causa que lo justifique
Persona	Es aquel rol designado por el profesor residente que se encarga ocasionalmente de la entrega de pases una vez autorizados en la Entrada de Servicio de la Universidad
Visitante	Es aquel rol que solicita autorización para entrar a la Universidad

**Tabla 2.1 Descripción de los actores de los actores del negocio**

### 2.8.2 Trabajadores del Negocio

Un trabajador del negocio es una abstracción de una persona (o un grupo de personas), una máquina o un sistema automatizado, que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores y manipulando entidades del negocio, representando estas generalmente, a los objetos que se toman, inspeccionan, producen o utilizan durante la realización de los casos de uso del negocio.

#### Descripción de los trabajadores del negocio

<b>Trabajador del Negocio</b>	<b>Descripción</b>
Asistente de Profesores	Es aquel rol encargado de recoger los datos necesarios para llenar una Solicitud de Entrada a la Residencia de Profesores
Directora de Residencia de Profesores	Es aquel rol encargado de autorizar una Solicitud de Entrada a la Residencia de Profesores
Vicerrector de Residencia de Profesores	Es aquel rol que se encarga de aprobar las solicitudes de pases especiales
Asistente de Control	Es aquel rol encargado de verificar la autorización de entrada, recoger los datos necesarios para llenar una

	Solicitud de Alimentos y autorizar la misma
Económica General	Es aquel rol encargado de verificar la autorización de la Solicitud de Alimentos, archivar la misma y entregar los tickets luego de haber efectuado el cobro
Técnico General	Es aquel rol encargado de archivar los pases autorizados hasta el momento de llegada del familiar, cuando verifica la veracidad de los datos necesarios y los registra para el debido control de su presencia en la Universidad

**Tabla 2.2 Descripción de los trabajadores del negocio**

### **2.8.3 Diagrama de Casos de uso del Negocio**

Un proceso de negocio es un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y manera y que emplean los recursos de la organización para dar resultados en apoyo a sus objetivos.

Un Caso de uso del negocio representa a un proceso de negocio, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores del negocio.

El Diagrama de Casos de uso del Negocio es un modelo que describe los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), es decir, describe las acciones que se realizan en el negocio.

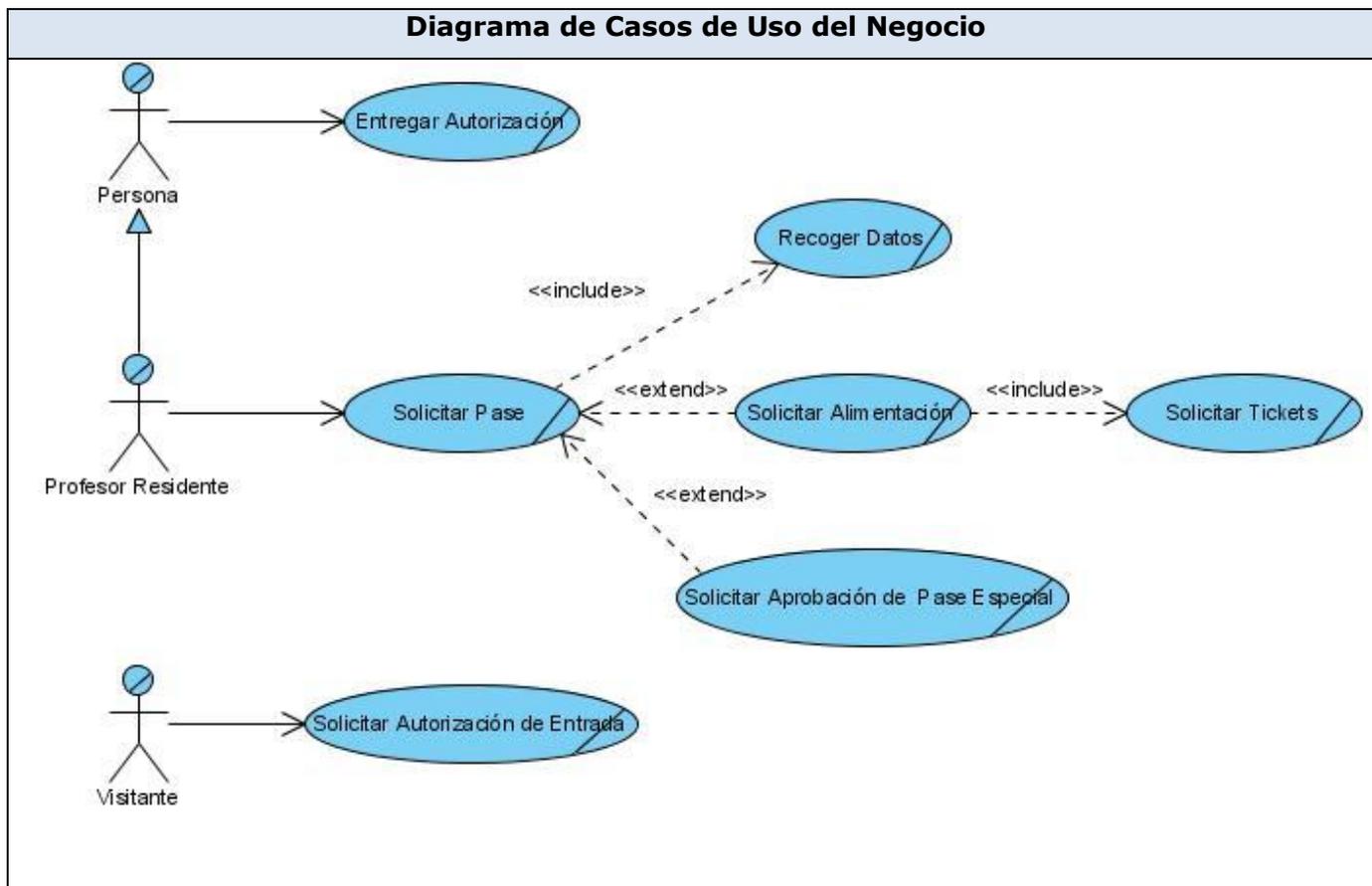


Figura 2.1 Diagrama de Casos de Uso del Negocio

### 2.8.4 Realización de los Casos de Uso

La Realización de un Caso de Uso de negocio muestra como colaboran los trabajadores y entidades del negocio para ejecutar el proceso. Se considera que con una descripción textual y los diagramas de actividad y objetos, es suficiente para describir completamente el proceso de negocio.

#### Descripción Textual de los Casos de Uso del Negocio

La Descripción Textual de los Casos de Uso se formaliza con el siguiente formato:

#### Descripción Textual del Caso de Uso Solicitar Pase

<b>Caso de Uso</b>	Solicitar Pase
<b>Actores</b>	Profesor Residente (Inicia)
<b>Trabajadores</b>	Asistente de Profesores
<b>Propósito</b>	Dar a los profesores residentes la posibilidad de

	realizar una solicitud de entrada a la Universidad para sus visitantes
<b>Resumen</b>	
El Caso de Uso se inicia cuando el profesor residente solicita a la asistente de profesores un pase para su visitante, quien si el período por el cual se necesita la solicitud se encuentra dentro del establecido (3 días) se dispone a recoger datos finalizando así el caso de uso	
<b>Precondiciones</b>	La solicitud se debe realizar con 72 horas de antelación a la fecha de llegada del visitante como máximo
<b>Casos de Uso Asociados</b>	Solicitar Aprobación de Pase Especial <<extend>> Solicitar Alimentación <<extend>> Recoger Datos <<include>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
1.El profesor residente solicita un pase para su visitante  3.El profesor residente comunica la información solicitada  5.Si el profesor residente quiere realizar una solicitud de alimentos Ver Caso de Uso Solicitar Alimentación	2.La asistente de profesores solicita conocer la cantidad de días para los que necesita autorización  4.Si la cantidad de días que solicita el profesor residente para su visitante se encuentra dentro del período establecido Ver Caso de Uso Recoger Datos
<b>Flujos Alternos</b>	
	4.Si la cantidad de días que solicita el profesor residente para su visitante es mayor que la establecida la asistente de profesores solicita conocer si tiene aprobación de pase especial

<p>5.El profesor comunica la información solicitada</p> <p>7.El profesor residente proporciona los datos solicitados</p>	<p>6.Si el profesor informa que tiene aprobación de pase especial la asistente de profesores solicita nombre y número de solapín</p> <p>8.Si la asistente de profesores verifica en el documento de pases especiales que la notificación para este profesor ha sido efectiva Ver Caso de Uso Recoger Datos</p>
	<p>6.Si el profesor comunica que no tiene aprobación la asistente de profesores le informa que la necesita Ver Caso de Uso Solicitar Aprobación de Pase Especial</p>
	<p>8.Si la asistente de profesores verifica en el documento de pases especiales que la notificación para este profesor no ha sido efectiva deniega la solicitud</p>
<p><b>Poscondiciones</b></p>	<p>Se procede al proceso de negocio pertinente</p>
<p><b>Prioridad</b></p>	<p>Crítico</p>
<p><b>Mejoras</b></p>	<p>El diseño de las actividades llevadas a cabo en este proceso señalarán los aspectos a destacar para que con su automatización se verifique el período de las reservaciones automáticamente</p>

**Tabla 2.3 Descripción Textual del Caso de Uso Solicitar Pase**

### Descripción Textual del Caso de Uso Recoger Datos

<p><b>Caso de Uso</b></p>	<p>Recoger Datos</p>
<p><b>Actores</b></p>	<p>Profesor Residente</p>
<p><b>Trabajadores</b></p>	<p>Asistente de Profesores y Directora de</p>

	Residencia de Profesores
<b>Propósito</b>	Recoger todos los datos necesarios para registrar de forma controlada la entrada de personal ajeno a la Universidad
<b>Resumen</b>	El Caso de Uso se inicia cuando la asistente de profesores solicita los datos necesarios para hacer efectiva la solicitud. El caso de uso finaliza cuando el profesor residente recibe el pase autorizado por la directora de residencia de profesores
<b>Precondiciones</b>	El periodo de estancia debe estar dentro del establecido o se debe haber solicitado un pase especial
<b>Casos de Uso Asociados</b>	Solicitar Pase <<include>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
2. El profesor residente proporciona la información solicitada	<ol style="list-style-type: none"> <li>1. La asistente de profesores solicita al profesor residente su nombre y apto, nombre, vínculo familiar y ciudadanía del visitante, además de la fecha de llegada y de salida del mismo</li> <li>3. La asistente de profesores registra la información proporcionada en el Registro de Visitantes</li> <li>4. La asistente de profesores llena con los mismos datos el Modelo de Solicitud de Entrada a la Residencia de Profesores</li> <li>5. Si la Directora de Residencia de Profesores se encuentra en ese momento la asistente de profesores le entrega el modelo de solicitud para su autorización</li> <li>6. La Directora de Residencia de Profesores</li> </ol>

<p>10. El profesor residente recibe el pase</p>	<p>recibe el modelo de solicitud</p> <p>7. La Directora de Residencia de Profesores firma y acuña el pase quedando de esta forma autorizado</p> <p>8. La Directora de Residencia de Profesores entrega el pase autorizado a la asistente de profesores</p> <p>9. La asistente de profesores entrega el pase autorizado al profesor residente</p>
<p><b>Flujos Alternos</b></p>	
	<p>5. Si la Directora de Residencia de Profesores no se encuentra en ese momento se le informa al profesor residente que debe volver luego a recoger su pase autorizado acción sin la cual no podrá solicitar alimentación</p>
<p><b>Poscondiciones</b></p>	<p>Queda registrada una nueva solicitud de pase</p>
<p><b>Prioridad</b></p>	<p>Crítico</p>
<p><b>Mejoras</b></p>	<p>El diseño de las actividades llevadas a cabo en este proceso señalarán los aspectos a destacar para que con su automatización les permita a los profesores residentes la reservación de pase con mayor facilidad y rapidez</p>

**Tabla 2.4 Descripción Textual del Caso de Uso Recoger Datos**

**Descripción Textual del Caso de Uso Solicitar Alimentación**

<p><b>Caso de Uso</b></p>	<p>Solicitar Alimentación</p>
<p><b>Actores</b></p>	<p>Profesor Residente</p>

<b>Trabajadores</b>	Asistente de Control
<b>Propósito</b>	Dar a los profesores residentes la posibilidad de garantizar la alimentación de sus visitantes durante su estancia en la universidad y registrar de forma controlada la información que se deriva
<b>Resumen</b> El caso de uso se inicia cuando el profesor residente pide a la asistente de control una solicitud de alimentos para su visitante, la misma verifica el pase autorizado y recoge los datos necesarios. El caso de uso finaliza cuando el profesor residente recibe la solicitud autorizada por la propia asistente de control	
<b>Precondiciones</b>	Se debe presentar un pase autorizado y la solicitud se debe realizar con 24 horas de antelación a la fecha de llegada del visitante como máximo
<b>Casos de Uso Asociados</b>	Solicitar Pase <<extend>> Solicitar Tickets <<include>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
1. El profesor residente pide una solicitud de alimentos para su visitante  3. El profesor residente presenta el pase	2. La asistente de control solicita al profesor residente el pase  4. La asistente de control recibe el pase  5. La asistente de control verifica que el pase este autorizado  6. Si el pase esta autorizado la asistente de control registra el nombre del profesor residente y la fecha en la que se presentó a realizar la solicitud, nombre del visitante y la fecha de llegada y de salida del mismo en el Registro de

<p>8. El profesor residente proporciona la información solicitada</p> <p>13. El profesor residente recibe la solicitud de alimentos autorizada</p> <p>14. Ver Caso de Uso Solicitar Tickets</p>	<p>Visitantes</p> <p>7. La asistente de control le solicita al profesor residente conocer las opciones (desayuno, almuerzo y comida) de que hará uso el visitante y el complejo comedor al que pertenece</p> <p>9. La asistente de control registra la nueva información en el Registro de Visitantes</p> <p>10. La asistente de Control llena con los mismos datos el modelo de solicitud de alimentos</p> <p>11. La asistente de control firma y acuña la solicitud de alimentos quedando de esta forma autorizada</p> <p>12. La asistente de control entrega la solicitud de alimentos autorizada al profesor residente</p>
<b>Flujos Alternos</b>	
	<p>6. Si el pase no está autorizado la asistente de control le deniega la solicitud al profesor residente</p>
<b>Poscondiciones</b>	<p>Queda registrada una nueva solicitud de alimentos</p>
<b>Prioridad</b>	<p>Secundario</p>
<b>Mejoras</b>	<p>El diseño de las actividades llevadas a cabo en este proceso señalarán los aspectos a destacar</p>

	para que con su automatización les permita a los profesores residentes la reservación de alimentos con mayor facilidad y rapidez
--	--

**Tabla 2.5 Descripción Textual del Caso de uso Solicitar Alimentación**

**Descripción Textual del Caso de Uso Solicitar Tickets**

<b>Caso de Uso</b>	Solicitar Tickets
<b>Actores</b>	Profesor Residente
<b>Trabajadores</b>	Económica General
<b>Propósito</b>	Evidenciar la acción de pago de la alimentación por los profesores residentes y registrar de forma controlada la información diaria que se deriva
<b>Resumen</b>	
El caso de uso se inicia cuando el profesor residente solicita a la económica general los tickets para su visitante, la misma verifica la solicitud de alimentos autorizada, seguidamente la archiva y procede a cobrar los tickets. El caso de uso finaliza cuando el profesor residente recibe los tickets	
<b>Precondiciones</b>	Se debe presentar una Solicitud de Alimentos autorizada
<b>Casos de Uso asociados</b>	Solicitar Alimentación <<include>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
1. El profesor residente solicita los tickets para su visitante.  3. El profesor residente entrega el Modelo de Solicitud de Alimentos	2. La económica general solicita al profesor residente el Modelo de Solicitud de Alimentos  4. La económica general recibe la solicitud de alimentos  5. La económica general verifica que la solicitud esté autorizada

<p>8. El profesor residente paga los tickets</p> <p>11. El profesor residente recibe los Tickets</p>	<p>6. Si la alimentación está autorizada la económica general archiva el Modelo de Solicitud de Alimentos</p> <p>7. La económica general solicita el pago de los tickets solamente para almuerzo y comida al profesor residente</p> <p>9. La económica general recibe el dinero</p> <p>10. La económica general entrega los tickets</p>
<p><b>Flujos Alternos</b></p>	
	<p>6. Si la alimentación no está autorizada la Técnica General de Economía deniega la solicitud al profesor residente</p>
<p><b>Poscondiciones</b></p>	<p>Se realiza diariamente una panilla perteneciente a la Dirección General de Alimentos donde se reflejan algunos datos de los familiares de los profesores que se presentan cada día, como son nombre y apellidos y las opciones de que hará uso</p>
<p><b>Prioridad</b></p>	<p>Secundario</p>
<p><b>Mejoras</b></p>	<p>Este proceso solo incluirá el diseño para actividades de implementación que suplan con la actividad de presentar la Solicitud de Alimentos Autorizada por parte del profesor residente, el resto de las acciones se mantienen</p>

**Tabla 2.6 Descripción Textual del Caso de Uso Solicitar Tickets**

**Descripción Textual del Caso de Uso Entregar Autorización**

<p><b>Caso de Uso</b></p>	<p>Entregar Autorización</p>
<p><b>Actores</b></p>	<p>Persona (Inicia)</p>

<b>Trabajadores</b>	Técnico General
<b>Propósito</b>	Archivar el pase autorizado hasta la llegada del visitante para poder verificar que el mismo puede entrar a la Universidad
<b>Resumen</b>	El caso de uso se inicia cuando el profesor residente u otra persona en su nombre, lleva a la entrada de servicio el modelo de solicitud de entrada a la residencia de profesores. El caso de uso termina cuando el técnico general archiva el pase autorizado hasta el momento de llegada del visitante
<b>Precondiciones</b>	Se debe presentar el Modelo de Solicitud de Entrada a la Residencia de Profesores autorizado
<b>Casos de uso asociados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
1. La persona entrega el Modelo de Solicitud de Entrada a la Residencia de profesores	2. El técnico general recibe el Modelo de Solicitud de Entrada a la Residencia de profesores 3. El técnico general verifica autorización 4. Si la solicitud esta autorizada el técnico general archiva el Modelo de Solicitud de Entrada a la Residencia de profesores
<b>Flujos Alternos</b>	
	4. Si la solicitud no está autorizada el técnico general rechaza la entrega
<b>Poscondiciones</b>	Queda archivado el pase hasta la llegada del visitante
<b>Prioridad</b>	Secundario
<b>Mejoras</b>	El diseño del sistema permitirá suprimir por completo las actividades llevadas a cabo en este proceso

**Tabla 2.7 Descripción Textual del Caso de Uso Entregar Autorización**

### Descripción Textual del Caso de Uso Solicitar Autorización de Entrada

<b>Caso de Uso</b>	Solicitar Autorización de Entrada
<b>Actores</b>	Visitante (Inicia)
<b>Trabajadores</b>	Técnico General
<b>Propósito</b>	Brindarle a los visitantes la posibilidad de entrar a la Universidad y registrar de forma controlada la presencia de personal ajeno a la misma
<b>Resumen</b>	
El caso de uso se inicia cuando el visitante se presenta en la Entrada de Servicio y solicita autorización para entrar a la universidad, el técnico general verifica que el visitante esté autorizado. El caso de uso finaliza con la devolución de su carnet de identidad al visitante	
<b>Precondiciones</b>	Se encontrará archivada la autorización de entrada del visitante
<b>Casos de uso Asociados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
<p>1. El visitante solicita autorización para entrar a la Universidad</p> <p>3. El visitante entrega el Carnet de Identidad</p>	<p>2. El Técnico General solicita el Carnet de Identidad</p> <p>4. El Técnico General recibe el Carnet de Identidad</p> <p>5. El Técnico General busca la autorización que coincida con el nombre del familiar</p> <p>6. Si está archivada la autorización el Técnico General con la ayuda de la misma y del Carnet de identidad, registra en el Libro de Incidencias, el nombre, edificio, y apartamento del profesor, el nombre y número de Carnet de Identidad del familiar, la fecha de llegada y de salida del mismo, así como la fecha y hora de ese</p>

<p>8. El visitante informa el número de teléfono del apartamento del profesor</p> <p>11. El visitante recibe su Carnet de Identidad</p>	<p>momento.</p> <p>7.El Técnico General solicita al visitante el número de teléfono del apartamento del profesor residente</p> <p>9. El Técnico General registra el número telefónico en el Libro de Incidencias</p> <p>10. El Técnico General devuelve el Carnet de Identidad al visitante</p>
<b>Flujos Alternos</b>	
	<p>6. Si no está archivada la autorización de entrada del familiar el Técnico General deniega la solicitud</p>
<b>Poscondiciones</b>	<p>Queda registrada la entrada de un visitante a la Universidad</p>
<b>Prioridad</b>	<p>Crítico</p>
<b>Mejoras</b>	<p>Este proceso solo incluirá el diseño para actividades de implementación que suplan con la actividad de verificar la autorización de la entrada del visitante por parte del técnico general, el resto de las actividades se mantienen</p>

**Tabla 2.8 Descripción Textual del Caso de Uso Solicitar Autorización de Entrada**

**Descripción Textual del Caso de Uso Solicitar Aprobación de Pase Especial**

<b>Caso de Uso</b>	Solicitar Aprobación de Pase Especial
<b>Actores</b>	Profesor Residente
<b>Trabajadores</b>	Vicerrector de Residencia de Profesores y Asistente de Profesores
<b>Propósito</b>	Brindarle la posibilidad a los profesores

	residentes que presenten alguna razón justificada para solicitar un pase especial por más de tres días
<b>Resumen</b> <p>El caso de uso se inicia cuando el profesor residente solicita al vicerrector de residencia de profesores un pase especial para su visitante, quien solicita conocer la causa que lo justifica, en dependencia de la cual aprueba o deniega dicha solicitud, notificando su acción a la asistente de profesores mediante un correo electrónico, quien registra los datos de la confirmación en el documento de pases especiales finalizando así el caso de uso.</p>	
<b>Precondiciones</b>	El profesor residente deberá presentar una causa justificada para que sea aprobada la solicitud
<b>Casos de uso Asociados</b>	Solicitar Pase <<extend>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del negocio</b>
<p>1. El profesor residente solicita aprobación de un pase especial</p> <p>3. El profesor residente le expone la información solicitada al vicerrector de residencia</p> <p>5. El profesor residente proporciona los datos solicitados</p>	<p>2. El vicerrector de residencia solicita conocer la cantidad de días y la causa que lo justifica</p> <p>4. Si la causa expuesta justifica la aprobación del pase especial el vicerrector de residencia solicita nombre y número de solapín</p> <p>6. El vicerrector de residencia registra los datos proporcionados en un correo electrónico</p> <p>7. El vicerrector de residencia envía el correo electrónico con los datos de la solicitud aprobada a la asistente de profesores</p> <p>8. La asistente de profesores recibe la notificación</p>

	9. La asistente de profesores registra los datos del profesor residente en el documento de pases especiales
<b>Flujos Alternos</b>	
	4. Si la causa expuesta no justifica la aprobación del pase especial se deniega la solicitud
<b>Poscondiciones</b>	Queda aprobada una solicitud de pase especial
<b>Prioridad</b>	Crítico
<b>Mejoras</b>	El diseño de las actividades llevadas a cabo en este proceso señalarán los aspectos a destacar para que con su automatización les permita a los profesores residentes la solicitud de un pase especial con mayor facilidad y rapidez

**Tabla 2.9 Descripción Textual del Caso de Uso Solicitar Aprobación de Pase Especial**

### **Diagramas de Actividades**

Un Diagrama de Actividades permite describir gráficamente la estructura de los flujos básicos o alternativos de los casos de uso del negocio. Consiste en un grafo que contiene estados en que puede hallarse una actividad, los mismos representan la ejecución de una sentencia de un procedimiento. Los diagramas de actividad de los Casos de Uso del negocio se podrán encontrar en el [Anexo 1](#).

### **Modelo de Objetos**

Un modelo de objetos muestra la participación de los trabajadores y entidades del negocio y las relaciones entre ellos.

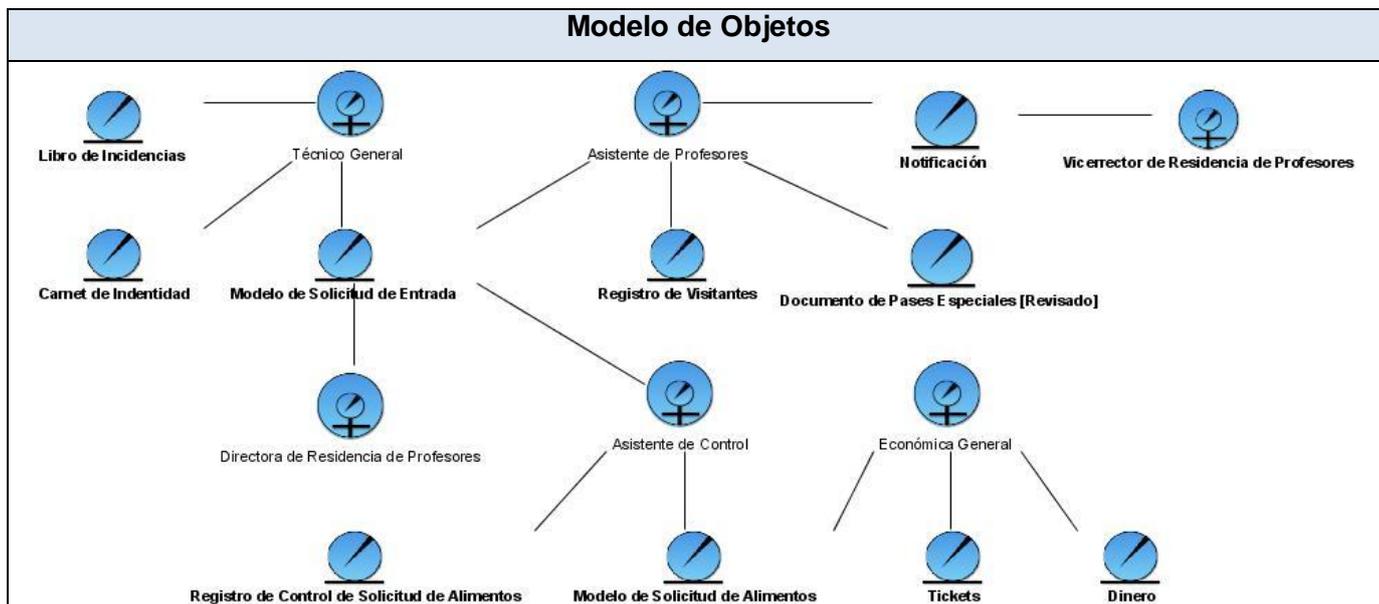


Figura 2.9 Modelo de Objetos

## 2.9 Especificación de los Requisitos de Software

La Especificación de los Requisitos de Software o Captura de Requisitos, es el proceso de indagar sobre lo que se debe construir, y de esta forma guiar el desarrollo hacia el sistema correcto. Con una descripción suficientemente buena de los mismos podrá llegarse a un acuerdo entre el cliente (incluyendo los usuarios) y los desarrolladores sobre que debe y que no debe hacer el sistema.

### 2.9.1 Requerimientos Funcionales

Los Requerimientos Funcionales son capacidades o condiciones que el sistema debe cumplir, o sea, indican que es lo que el sistema debe hacer. Se mantienen invariables, sin importarle con que propiedades o cualidades se relacionen.

#### R1 Autenticar Usuario

R1.1 Verificar Usuario

R1.2 Verificar Contraseña

R1.3 Definir permisos según el rol del usuario autenticado

#### R2 Gestionar Usuario

R2.1 Insertar Usuario

R2.1.1 Asignar Rol

R2.2 Modificar Usuario

R2.3 Eliminar Usuario

### **R3 Gestionar Rol**

R2.1 Insertar Rol

R2.1.1 Asignarle permisos

R2.2 Modificar Rol

R2.3 Eliminar Rol

### **R4 Gestionar Reservación**

R4.1 Realizar Reservación

R4.1.1 Mostrar datos de Profesor

R4.1.2 Registrar datos del visitante

R4.3 Modificar Reservación

R4.4 Cancelar Reservación

R4.5 Mostrar Reservaciones de Profesor

### **R5 Realizar Reservación de Alimentos**

R5.1 Mostrar Complejo Comedor de Profesor

R5.2 Registrar Opciones

### **R6 Solicitar Pase Especial**

R6.1 Mostrar datos de Profesor

R6.2 Registrar datos del visitante

### **R7 Aprobar Pase Especial**

R7.1 Mostrar Listado de solicitudes de pases especiales

R7.2 Aprobar Solicitud

### **R8 Mostrar Reportes**

R8.1 Mostrar Listado de las reservaciones de pase realizadas por un profesor dado su solapín

R8.2 Mostrar Listado de las reservaciones de pase realizadas por un profesor en un mes, dado su solapín y el mes

R8.3 Mostrar Listado de las reservaciones de pase realizadas en un mes, dado el mes

R8.4 Mostrar Listado de las reservaciones de alimentos realizadas por un profesor dado su solapín

R8.5 Mostrar Listado de las reservaciones de alimentos realizadas por un profesor en un mes, dado su solapín y el mes

R8.6 Mostrar Listado de las reservaciones de alimentos realizadas en un mes, dado el mes

R8.7 Mostrar Listado de las reservaciones de pase realizadas dada la fecha de llegada y la fecha de salida

R8.8 Mostrar Listado de las reservaciones de alimentos realizadas dada la fecha de llegada y la fecha de salida

R8.9 Mostrar Listado de los familiares que se encuentran en la residencia de profesores dado el edificio

R8.10 Mostrar Listado de las reservaciones de pase realizadas por un profesor dado su login

### 2.9.2 Requerimientos No Funcionales

Los Requerimientos No Funcionales son propiedades o cualidades que el producto debe tener. Son las características que hacen al producto atractivo, usable, rápido o confiable, los mismos son fundamentales para el éxito del producto e indican como ha de comportarse el sistema.

- ✓ Requerimientos de apariencia o interfaz externa

La aplicación deberá desarrollarse en un ambiente Web. Debiendo presentar una interfaz sencilla, amigable y de fácil uso, para que pueda ser utilizada sin mucho entrenamiento por los usuarios, teniendo en cuenta que algunos de éstos pueden no ser especialistas en informática, sin dejar de poseer un ambiente profesional.

- ✓ Requerimientos de Usabilidad

El sistema deberá poder ser usado por todos los profesores residentes en la UCI, quienes podrán realizar una reservación de pase y opcionalmente de alimentación y modificar o cancelar una previamente hecha. También podrá solicitar aprobación para un pase especial y conocer las reservaciones que ha realizado tras su autenticación.

Además por algunos trabajadores autorizados a los que el sistema brindará reportes sobre las reservaciones, estos permitirán obtener información, estadísticas y control sobre las mismas y por el administrador quien podrá principalmente agregar nuevos usuarios y roles al sistema. Definiéndose así

niveles de usuario que se diferenciarán en las opciones que el sitio les posibilite. Basta con que los mismos cuenten con conocimientos básicos en el manejo de la computadora y el ambiente Web a grandes rasgos.

- ✓ Requerimientos de Rendimiento

Para garantizar una aplicación con óptimas funcionalidades, la misma debe garantizar la rapidez de respuesta ante las solicitudes de los usuarios, debe llevar a cabo con gran velocidad el procesamiento de la información y el rápido acceso a sus páginas. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas a la Base de Datos.

- ✓ Requerimientos de Seguridad

Confidencialidad: Al sistema deberán tener acceso todos los profesores residentes de la universidad y algunos usuarios, todos con el los permisos correspondientes, por lo que será necesaria una autenticación previa, estando la información a la que estos acceden protegida del acceso no autorizado y la divulgación.

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

Disponibilidad: Al sistema deberá ser posible acceder las 24 horas del día para todos los usuarios autorizados

- ✓ Requerimientos de Portabilidad

La herramienta desarrollada deberá ser multiplataforma (Linux o Windows)

- ✓ Restricciones en el diseño y la implementación

Utilizar los estándares establecidos de codificación, diseño, entre otros. Emplear como servidores Web y de bases de datos Apache y PostgreSQL respectivamente. Utilizar como lenguaje del lado del

servidor PHP y del lado del cliente JavaScript y HTML. Para garantizar una mejor documentación del sistema, así como el uso de última tecnología, se utiliza para realizar el análisis y el diseño del sistema UML, y su extensión para el desarrollo de proyectos Web. Como herramienta de apoyo a este lenguaje se utiliza Visual Paradigm.

### 2.10 Modelado del sistema

El Modelado del sistema se centra fundamentalmente en estructurar los requisitos funcionales y no funcionales mediante casos de uso.

#### 2.10.1 Actores del Sistema

Los actores del sistema representan los distintos tipos de usuarios y terceros fuera del sistema que colaboran o interactúan con el mismo.

#### Descripción de los Actores del Sistema

Actor	Descripción
Usuario	Es aquel rol autorizado a entrar al sistema y según sus permisos podrá llevar a cabo acciones dentro del mismo
Administrador	Es aquel rol usuario del sistema que podrá insertar, modificar o eliminar usuarios en el mismo y asignarles un rol en caso de inserción. Además podrá insertar, modificar o eliminar roles en el sistema y a la vez asignarle permisos en caso de inserción o modificación
Profesor Residente	Es aquel rol usuario del sistema que podrá realizar reservaciones de pases y opcionalmente de alimentación y modificar o cancelar las previamente hechas. Además podrá solicitar aprobación de un pase especial y conocer las reservaciones que ha realizado a través de su login
Asistente de Profesores	Es aquel rol usuario del sistema que podrá ver la información general relacionada con las

	reservaciones de pase
Asistente de Control	Es aquel rol usuario del sistema que podrá ver la información general relacionada con las reservaciones de alimentos
Técnico General	Es aquel rol usuario del sistema que podrá ver la información relacionada con las reservaciones de pase para el período de fechas especificado
Económica General	Es aquel rol usuario del sistema que podrá ver la información relacionada con las reservaciones de alimentos para el período de fechas especificado
Vicerrector de Residencia de Profesores	Es aquel rol usuario del sistema que podrá aprobar o denegar una solicitud de pase especial y hacerla efectiva, además podrá ver la información relacionada con las reservaciones de pase
Instructora	Es aquel rol usuario del sistema que podrá ver la información relacionada con la estancia de visitantes en la residencia de profesores

**Tabla 2.10 Descripción de los Actores del Sistema**

### 2.10.2 Diagrama de Casos de Uso del Sistema

Un Diagrama de Casos de Uso del Sistema es un modelo que contiene actores, casos de uso y sus relaciones. Este permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre las condiciones y posibilidades que debe cumplir el sistema para cada tipo de usuario.

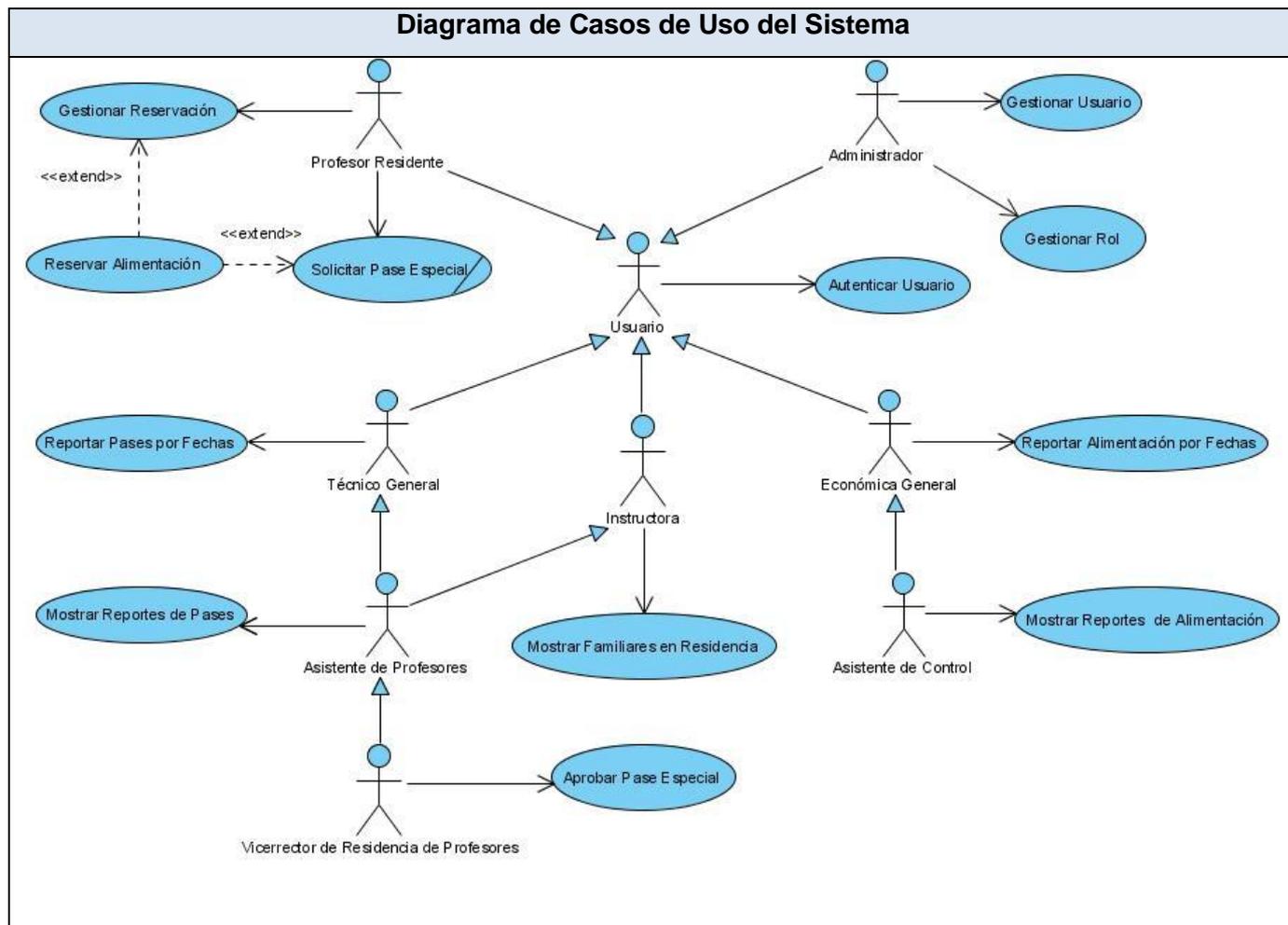


Figura 2.10 Diagrama de Casos de Uso del Sistema

### 2.10.3 Listado de los Casos de uso del Sistema

<b>CU -1</b>	<b>Autenticar Usuario</b>
<b>Actor</b>	Usuario (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el usuario introduce su nombre de usuario y contraseña del dominio en el sistema. Este verifica que los datos son correctos y da acceso a las opciones correspondientes finalizando así el caso de uso
<b>Referencia</b>	R1

Tabla 2.11 Autenticar Usuario

<b>CU -2</b>	<b>Gestionar Usuario</b>
<b>Actor</b>	Administrador (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el administrador accede a la página donde se le muestran las opciones que le corresponden según sus permisos (Insertar, Modificar o Eliminar un Usuario) y selecciona la que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Referencia</b>	R2

**Tabla 2.12 Gestionar Usuario**

<b>CU -3</b>	<b>Gestionar Rol</b>
<b>Actor</b>	Administrador (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el administrador accede a la página donde se le muestran las opciones que le corresponden según sus permisos (Modificar, Eliminar o Insertar un Rol que incluye asignarle permisos) y selecciona la que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Referencia</b>	R3

**Tabla 2.13 Gestionar Rol**

<b>CU -4</b>	<b>Gestionar Reservación</b>
<b>Actor</b>	Profesor Residente (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el profesor residente accede a la página donde se le muestran las opciones que le corresponden según sus permisos (Realizar, Cancelar o Modificar una

	Reservación y Mostrar Reservaciones realizadas) y selecciona la opción que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Referencia</b>	R4

**Tabla 2.14 Gestionar Reservación**

<b>CU -5</b>	<b>Reservar Alimentación</b>
<b>Actor</b>	Profesor Residente
<b>Descripción</b>	El caso de uso se inicia cuando el sistema muestra el Complejo Comedor al que pertenece el profesor residente, quien selecciona las opciones de alimentación que utilizará el visitante finalizando así el caso de uso
<b>Referencia</b>	R5

**Tabla 2.15 Reservar Alimentación**

<b>CU -6</b>	<b>Solicitar Pase Especial</b>
<b>Actor</b>	Profesor Residente (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el profesor residente accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Solicitar un Pase Especial e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Referencia</b>	R6

**Tabla 2.16 Solicitar Pase Especial**

<b>CU -6</b>	<b>Aprobar Pase Especial</b>
<b>Actor</b>	Vicerrector de Residencia de Profesores (Inicia)

<b>Descripción</b>	El caso de uso se inicia cuando el profesor Vicerrector de Residencia de Profesores accede a la página donde selecciona la opción de Mostrar Pases Especiales, luego selecciona la solicitud que desea analizar y entonces aprueba o deniega la misma. El Caso de uso finaliza con el registro de la nueva reservación en el sistema
<b>Referencia</b>	R7

**Tabla 2.17 Aprobar Pase Especial**

<b>CU -8</b>	<b>Mostrar Visitantes en Residencia</b>
<b>Actor</b>	Instructora (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando la Instructora accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Mostrar Visitantes en Residencia e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Referencia</b>	R8

**Tabla 2.18 Mostrar Visitantes en Residencia**

<b>CU -9</b>	<b>Reportar Pases por Fechas</b>
<b>Actor</b>	Técnico General (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando el Técnico General accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Mostrar reservaciones en un Período e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Referencia</b>	R8

**Tabla 2.19 Reportar Pases por Fechas**

<b>CU -10</b>	<b>Mostrar Reportes de Pases</b>
<b>Actor</b>	Asistente de Profesores (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando la Asistente de Profesores accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona el reporte que desea obtener e introduce los datos necesarios para hacerlo efectivo. El caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Referencia</b>	R8

**Tabla 2.20 Mostrar Reportes de Alimentación**

<b>CU -11</b>	<b>Reportar Alimentación por Fechas</b>
<b>Actor</b>	Económica General (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando la Económica General accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Mostrar Alimentación en un Período e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Referencia</b>	R8

**Tabla 2.21 Reportar Alimentación por Fechas**

<b>CU -12</b>	<b>Mostrar Reportes de Alimentación</b>
<b>Actor</b>	Asistente General (Inicia)
<b>Descripción</b>	El caso de uso se inicia cuando la Asistente de Control accede a la página donde se le muestran las opciones que le corresponden según sus

	permisos y selecciona el reporte que desea obtener e introduce los datos necesarios para hacerlo efectivo. El caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Referencia</b>	R8

**Tabla 2.22 Mostrar Reportes de Alimentación**

### 2.10.4 Descripción Textual de los Casos de Uso Expandidos

<b>Caso de Uso</b>	<b>Autenticar Usuario</b>
<b>Actor</b>	Usuario (Inicia)
<b>Propósito</b>	Verificar que el usuario está autorizado a entrar al sistema
<b>Resumen</b>	El caso de uso se inicia cuando el usuario introduce su nombre de usuario y contraseña del dominio en el sistema. Este verifica que los datos son correctos y da acceso a las opciones correspondientes finalizando así el caso de uso
<b>Precondiciones</b>	El usuario debe haber seleccionado la opción de acceder al sitio Web
<b>Poscondiciones</b>	El usuario queda autenticado
<b>Referencia</b>	R1
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El usuario introduce su nombre de usuario y contraseña del dominio 2. El usuario selecciona la opción "Aceptar"	2.1 El sistema verifica que ningún campo obligatorio este vacío 2.2 El sistema verifica que el usuario y la contraseña introducidos tengan acceso al sistema 2.3 El sistema le permite la entrada al usuario

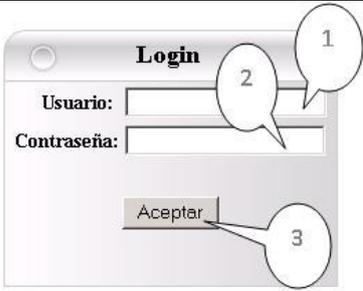
	mostrando las opciones que le corresponden según sus permisos
<b>Prototipo de Interfaz de Usuario</b>	
	
<b>Descripción de las Entradas de Datos</b>	
<b>[1]</b> Campo de Texto para introducir el usuario	
<b>[2]</b> Campo de Texto para introducir la contraseña	
<b>[3]</b> Botón que inicia las acciones para autenticar	
<b>Flujos Alternos</b>	
	2.1 Si el sistema verifica que algún campo obligatorio está vacío emite un mensaje de error
	2.2 Si el sistema verifica que el usuario y la contraseña introducidos no tienen acceso al sistema emite un mensaje de error
<b>Prioridad</b>	Crítico

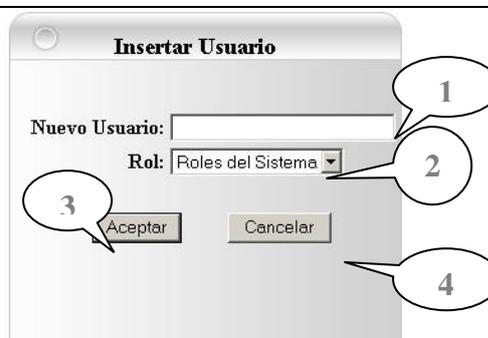
Tabla 2.23 Descripción Textual CU Autenticar Usuario

Caso de Uso	Gestionar Usuario
<b>Actor</b>	Administrador (Inicia)
<b>Propósito</b>	Permitir en el sistema: Insertar Usuario Modificar Usuario Eliminar Usuario
<b>Resumen</b>	El caso de uso se inicia cuando el administrador accede a la página donde se le muestran las opciones que le corresponden según sus

	permisos (Modificar, Eliminar e Insertar un Usuario lo que incluye asignarle un rol) y selecciona la que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Precondiciones</b>	El usuario autenticado debe ser administrador del sistema
<b>Poscondiciones</b>	Queda actualizada la información sobre los usuarios que tienen acceso al sistema
<b>Referencia</b>	R2
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El administrador accede al sistema  2.El administrador selecciona la acción que desea realizar	1.1 El sistema le muestra las opciones que le corresponden según sus permisos  2.1 Si el administrador del sistema selecciona: a) "Insertar Usuario" ir a la Sección con el mismo nombre b) "Modificar Usuario" ir a la Sección con el mismo nombre c) "Eliminar Usuario" ir a la Sección con el mismo nombre
<b>Sección "Insertar Usuario"</b>	
2. El administrador del sistema llena los campos  3. El administrador del sistema selecciona la opción "Aceptar"	1.1 El sistema le muestra los campos a llenar ✓ Nuevo usuario ✓ Rol (mostrando los existentes en el sistema)  3.1 El sistema verifica que ningún campo obligatorio este vacío

	<p>3.2 El sistema verifica que el usuario introducido pertenezca al dominio UCI</p> <p>3.3 El sistema verifica que el usuario introducido no se encuentre ya autorizado</p> <p>3.4 El sistema almacena la información relacionada con el registro</p> <p>3.5 El sistema muestra un mensaje notificando la efectividad de su acción</p>
--	--

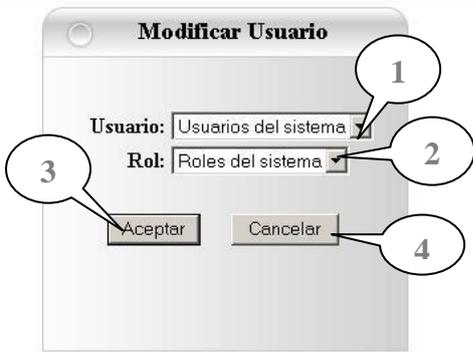
### Prototipo de Interfaz de Usuario



[1] Campo de Texto para introducir el usuario
[2] Campo de Texto para seleccionar el rol
[3] Botón que inicia las acciones para insertar un usuario
[4] Botón que regresa a la página anterior

### Flujos Alternos

3.Si el administrador selecciona la opción "Cancelar" el sistema regresa a la página anterior	
	3.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
	3.2 Si el sistema verifica que el usuario introducido no pertenece al dominio UCI muestra un mensaje de error
	3.3 Si el sistema verifica que el usuario introducido ya se encuentra autorizado muestra un mensaje de error

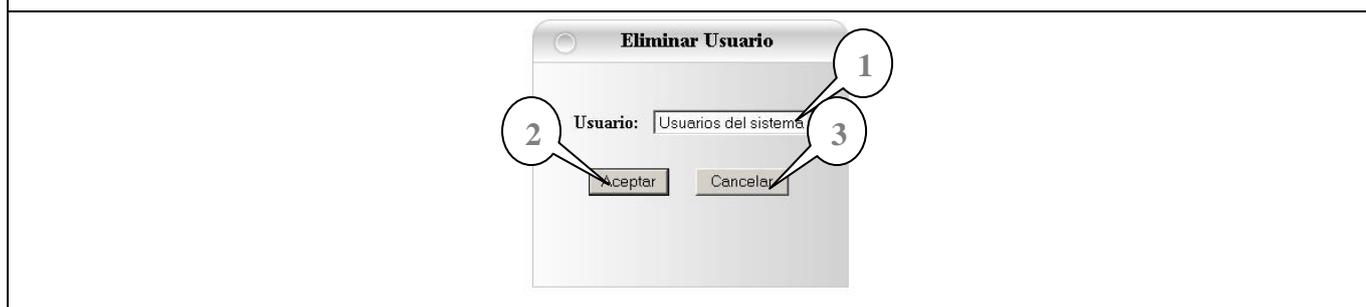
<b>Sección “Modificar Usuario”</b>	
<p>2. El administrador del sistema llena los campos</p> <p>3. El administrador del sistema selecciona la opción “Aceptar”</p>	<p>1.1 El sistema le muestra los campos a llenar</p> <ul style="list-style-type: none"> <li>✓ Usuario (mostrando los existentes en el sistema)</li> <li>✓ Rol (mostrando los existentes en el sistema)</li> </ul> <p>3.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>3.2 El sistema verifica que el rol del usuario introducido no sea el que tiene asignado</p> <p>3.3 El sistema almacena la información relacionada con la modificación</p> <p>3.4 El sistema muestra un mensaje notificando la efectividad de su acción</p>
<b>Prototipo de Interfaz de Usuario</b>	
	
[1] Campo de Texto para seleccionar el usuario	
[2] Campo de Texto para seleccionar el rol	
[3] Botón que inicia las acciones para modificar un usuario	
[4] Botón que regresa a la página anterior	
<b>Flujos Alternos</b>	
<p>3.Si el administrador selecciona la opción “Cancelar”, el sistema regresa a la página anterior</p>	

	3.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
	3.2 Si el sistema verifica que el rol del usuario introducido es el que tiene asignado emite un mensaje de error

**Sección “Eliminar Usuario”**

<p>2. El administrador del sistema llena los campos</p> <p>3. El administrador del sistema selecciona la opción “Aceptar”</p>	<p>1.1 El sistema le muestra los campos a llenar</p> <ul style="list-style-type: none"> <li>✓ Usuario a eliminar (mostrando los existentes en el sistema)</li> </ul> <p>3.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>3.2 El sistema almacena la información relacionada con la acción</p> <p>3.4 El sistema muestra un mensaje notificando la efectividad de su acción</p>
---	--

**Prototipo de Interfaz de Usuario**



- [1]** Campo de Texto para introducir el usuario
- [2]** Botón que inicia las acciones para eliminar un usuario
- [3]** Botón que regresa a la página anterior

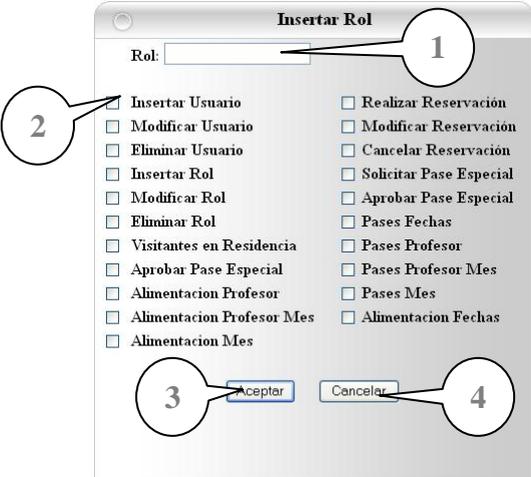
**Flujos Alternos**

3.Si el administrador selecciona la opción “Cancelar”, el sistema regresa a la página	
---	--

anterior	
	3.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
<b>Prioridad</b>	Crítico

**Tabla 2.24 Descripción Textual CU Gestionar Usuario**

<b>Caso de Uso</b>	<b>Gestionar Rol</b>
<b>Actor</b>	Administrador (Inicia)
<b>Propósito</b>	Permitir en el sistema: Insertar Rol Modificar Rol Eliminar Rol
<b>Resumen</b>	El caso de uso se inicia cuando el administrador accede a la página donde se le muestran las opciones que le corresponden según sus permisos (Modificar, Eliminar o Insertar un Rol que incluye asignarle permisos) y selecciona la que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Precondiciones</b>	El usuario autenticado debe ser administrador
<b>Poscondiciones</b>	Queda actualizada la información sobre los roles de los usuarios que tienen acceso al sistema
<b>Referencia</b>	R3
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1.El administrador accede al sistema	1.1 El sistema le muestra las opciones que le corresponden según sus permisos
2.El administrador selecciona la acción que desea realizar	2.1 Si el administrador selecciona: a) "Insertar Rol" ir a la Sección con el mismo

	<p>nombre</p> <p>b) “Modificar Rol” ir a la Sección con el mismo nombre</p> <p>c) “Eliminar Rol” ir a la Sección con el mismo nombre</p>
<b>Sección “Insertar Rol”</b>	
<p>2. El administrador del sistema llena los campos</p> <p>3. El administrador del sistema selecciona la opción “Aceptar”</p>	<p>1.1 El sistema le muestra los campos a llenar</p> <ul style="list-style-type: none"> <li>✓ Nuevo Rol</li> <li>✓ Permisos (todas las funcionalidades que posee el sistema)</li> </ul> <p>3.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>3.2 El administrador verifica que el rol introducido no se encuentre ya registrado</p> <p>3.3 El sistema almacena la información relacionada con el registro</p> <p>3.4 El sistema muestra un mensaje notificando la efectividad de su acción</p>
	

[1] Campo de Texto para insertar el rol	
[2] Campo de Texto para seleccionar el permiso	
[3] Botón que inicia las acciones para insertar un rol	
[4] Botón que regresa a la página anterior	
<b>Flujos Alternos</b>	
3.Si el administrador selecciona la opción “Cancelar”, el sistema regresa a la página anterior	
	3.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
	3.2 Si el administrador verifica que el rol introducido se encuentra ya registrado emite un mensaje error
<b>Sección “Modificar Rol”</b>	
<p>2.El administrador del sistema selecciona el rol que desea modificar</p> <p>3. El administrador del sistema selecciona la opción “Modificar”</p> <p>4. El administrador del sistema modifica los permisos</p> <p>5. El administrador del sistema selecciona la opción “Aceptar”</p>	<p>1.1 El sistema le muestra los campos a llenar</p> <ul style="list-style-type: none"> <li>✓ Rol (mostrando los existentes en el sistema)</li> </ul> <p>3.1 El sistema verifica que haya seleccionado algún rol</p> <p>3.2 El sistema le muestra los permisos que posee ese rol actualmente en el sistema</p> <p>5.1 El sistema verifica que al menos un permiso este seleccionado</p> <p>5.2 El sistema verifica que los permisos introducidos no sean los que ya tenía asignado</p> <p>5.3 El sistema almacena la información relacionada con la acción</p> <p>5.4 El sistema muestra un mensaje notificando la</p>

	efectividad de su acción
<b>Flujos Alternos</b>	
	3.1 Si el sistema verifica que no ha seleccionado ningún rol emite un mensaje de error
5. Si el administrador selecciona la opción “Cancelar”, el sistema regresa a la página anterior	
	5.1 Si el sistema verifica que no existen permisos seleccionados emite un mensaje de error
	5.2 Si el sistema verifica que los permisos introducidos son los que ya tenía asignado emite un mensaje de error
<b>Sección “Eliminar Rol”</b>	
2. El administrador del sistema llena los campos 3. El administrador del sistema selecciona la opción “Aceptar”	1.1 El sistema le muestra los campos a llenar <ul style="list-style-type: none"> <li>✓ Rol a eliminar (mostrando los existentes en el sistema)</li> </ul> 3.1 El sistema verifica que haya seleccionado algún rol 3.2 El sistema actualiza la información relacionada con la acción 3.4 El sistema muestra un mensaje notificando la efectividad de su acción
<b>Flujos Alternos</b>	
3. Si el administrador del sistema selecciona la opción “Cancelar”, el sistema regresa a la página anterior	
	3.1 Si el sistema verifica que no ha seleccionado ningún rol emite un mensaje de error

<b>Prioridad</b>	Crítico
------------------	---------

**Tabla 2.25 Descripción Textual CU Gestionar Rol**

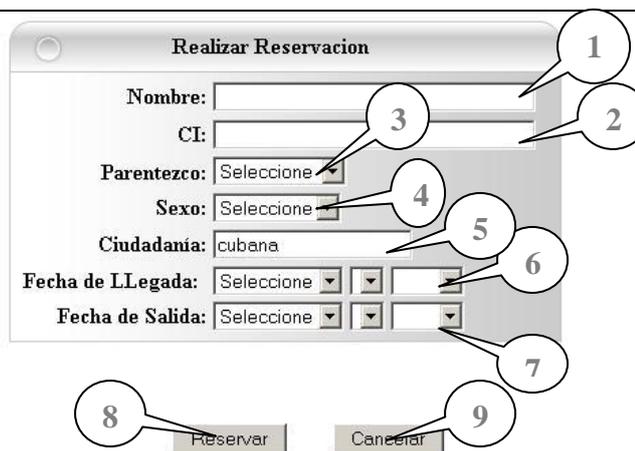
<b>Caso de Uso</b>	<b>Gestionar Reservación</b>
<b>Actor</b>	Profesor Residente (Inicia)
<b>Propósito</b>	Permitir que los profesores puedan realizar reservaciones de pases para sus visitantes y modificar o cancelar las previamente hechas. Además podrán conocer todas sus reservaciones realizadas
<b>Resumen</b>	El caso de uso se inicia cuando el profesor residente accede a la página donde se le muestran las opciones que le corresponden según sus permisos (Realizar, Cancelar o Modificar una Reservación y Mostrar Reservaciones realizadas) y selecciona la opción que desea realizar e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Precondiciones</b>	El usuario autenticado debe ser un profesor residente
<b>Poscondiciones</b>	Queda actualizada la información relacionada con las reservaciones de pase
<b>Referencia</b>	R4
<b>Casos de Uso Relacionados</b>	Reservar Alimentación <<extend>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1.El profesor residente accede al sistema	1.1 El sistema le muestra las opciones que le corresponden según sus permisos
2.El profesor residente selecciona la acción que desea realizar	2.1 Si el profesor residente selecciona: a) “Realizar Reservación” ir a la Sección con el

	<p>mismo nombre</p> <p>b) “Modificar Reservación” ir a la Sección con el mismo nombre</p> <p>c) “Cancelar Reservación” ir a la Sección con el mismo nombre</p> <p>c) “Mostrar Reservaciones realizadas” ir a la Sección con el mismo nombre</p>
<b>Sección “Realizar Reservación”</b>	
<p>2. El profesor residente llena los campos</p> <p>3. Si el profesor residente desea realizar una reservación de alimentos Ver Caso de Uso Reservar Alimentación</p> <p>4. El profesor residente selecciona la opción “Aceptar”</p>	<p>1.1 El sistema muestra los datos del profesor que tomó de la autenticación</p> <p>1.2 El sistema muestra los campos a llenar con datos del visitante:</p> <ul style="list-style-type: none"> <li>✓ nombre</li> <li>✓ carnet de identidad</li> <li>✓ vínculo familiar</li> <li>✓ sexo</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> </ul> <p>4.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>4.2 El sistema verifica que los datos a entrar sean del tipo correcto</p> <p>4.3 El sistema verifica que la reservación cumpla con las restricciones en cuanto a parentesco, sexo y periodo de estancia</p> <p>4.4 El sistema verifica que ese profesor residente no tenga otra reservación con la que tenga algún día en común</p> <p>4.5 El sistema almacena la información</p>

relacionada con la reservación

4.6 El sistema muestra un mensaje notificando la efectividad de su reservación

**Prototipo de Interfaz de Usuario**



[1] Campo de Texto para introducir el nombre del visitante

[2] Campo de Texto para introducir el CI del visitante

[3] Campo de Texto para seleccionar el parentesco del visitante

[4] Campo de Texto para seleccionar el sexo del visitante

[5] Campo de Texto no modificable

[6] Campos de Textos para seleccionar la fecha de llegada del visitante

[7] Campos de Textos para seleccionar la fecha de salida del visitante

[8] Botón que inicia las acciones para reservar

[9] Botón que regresa a la página anterior

**Flujos Alternos**

<p>4. Si el profesor residente selecciona la opción "Cancelar", el sistema regresa a la página anterior</p>	
	<p>4.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error</p>
	<p>4.2 Si el sistema verifica que los datos a entrar son de tipo incorrecto emite un mensaje de error</p>
	<p>4.3 Si el sistema verifica que la reservación no cumple con alguna de las restricciones en cuanto</p>

	a parentesco, sexo y período de estancia emite un mensaje de error
	4.4 Si el sistema verifica que ese profesor residente ya tiene otra reservación con la que tenga algún día en común emite un mensaje de error
<b>Sección “Modificar Reservación”</b>	
<p>2. El profesor residente selecciona la opción “Modificar” de la reservación a la que desee realizarle algún cambio</p> <p>3. El profesor residente modifica los campos</p> <p>4. El profesor residente selecciona la opción “Aceptar”</p>	<p>1.1 El sistema muestra las reservaciones posibles a modificar (aquellas cuyo estado sea “pendiente”), identificándose con los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del visitante</li> <li>✓ carnet de identidad del visitante</li> <li>✓ vínculo familiar</li> <li>✓ sexo</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> </ul> <p>2.1 El sistema da la posibilidad de modificar los datos mostrados</p> <p>4.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>4.2 El sistema verifica que los datos a entrar sean del tipo correcto</p> <p>4.3 El sistema verifica que la reservación cumpla con las restricciones en cuanto a parentesco, sexo y período de estancia</p> <p>4.4 El sistema verifica que ese profesor residente no tenga otra reservación pendiente con la que tenga algún día en común</p>

	<p>4.5 El sistema verifica que los datos introducidos no coincidan con los existentes</p> <p>4.6 El sistema almacena la información relacionada con la reservación</p> <p>4.7 El sistema muestra un mensaje notificando la efectividad de su acción</p>
<b>Flujos Alternos</b>	
	1.1 Si el profesor no tiene reservaciones posibles a modificar el sistema muestra un mensaje de información
4. Si el profesor residente selecciona la opción “Cancelar”, el sistema regresa a la página anterior	
	4.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
	4.2 Si el sistema verifica que los datos a entrar son de tipo incorrecto emite un mensaje de error
	4.3 Si el sistema verifica que la reservación no cumple con alguna de las restricciones en cuanto a parentesco, sexo y período de estancia emite un mensaje de error
	4.4 Si el sistema verifica que ese profesor residente ya tiene otra reservación con la que tenga algún día en común emite un mensaje de error
	4.5 Si el sistema verifica que los datos introducidos coinciden con los existentes emite un mensaje de error
<b>Sección “Cancelar Reservación”</b>	
	1.1 El sistema muestra las reservaciones posibles a cancelar (aquellas cuyo estado sea “pendiente”), identificándose con los siguientes datos:

<p>2. El profesor residente selecciona la reservación que desea cancelar</p> <p>3. El profesor residente selecciona la opción "Cancelar"</p>	<ul style="list-style-type: none"> <li>✓ nombre del visitante</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> </ul> <p>3.1 El sistema muestra un mensaje notificando la efectividad de su acción</p>
<b>Flujos Alternos</b>	
	<p>1.1 Si el profesor no tiene reservaciones posibles a cancelar el sistema muestra un mensaje de información</p>
<p>3.Si el profesor residente selecciona la opción "Salir", el sistema regresa a la página anterior</p>	
<b>Sección "Mostrar Reservaciones"</b>	
	<p>1.1 El sistema muestra todas las reservaciones realizadas por el profesor residente a través de su login, los datos que se deberán mostrar son los siguientes:</p> <ul style="list-style-type: none"> <li>✓ fecha de la reservación</li> <li>✓ nombre del visitante</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> </ul>
<b>Flujos Alternos</b>	
	<p>1.1 Si el profesor no ha realizado reservaciones el sistema muestra un mensaje de información</p>
<b>Prioridad</b>	Crítico

Tabla 2.26 Descripción Textual CU Gestionar Reservación

<b>Caso de Uso</b>	<b>Reservar Alimentación</b>
<b>Actor</b>	Profesor Residente

<b>Propósito</b>	Permitir que los profesores residentes puedan realizar reservaciones de alimentos para sus visitantes
<b>Resumen</b>	El caso de uso se inicia cuando el sistema muestra el Complejo Comedor al que pertenece el profesor residente, quien selecciona las opciones de alimentación que utilizará el visitante finalizando así el caso de uso
<b>Precondiciones</b>	El sistema deberá encontrarse en la página que permita realizar la selección
<b>Poscondiciones</b>	Quedan seleccionadas las opciones de alimentación de que hará uso el visitante
<b>Referencia</b>	R5
<b>Casos de Uso Relacionados</b>	Solicitar Pase <<extend>> Solicitar Pase Especial <<extend>>
<b>Descripción de las entradas de datos</b>	
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra el Complejo Comedor al que pertenece el profesor autenticado 2. El sistema muestra los campos a llenar <ul style="list-style-type: none"> <li>✓ desayuno</li> <li>✓ almuerzo</li> <li>✓ comida</li> </ul>
3. El profesor residente llena los campos	
<b>Prototipo de Interfaz de Usuario</b>	

[1] Campo de Texto para seleccionar las opciones de que hará uso el visitante	
[2] Campo de Texto para mostrar el Complejo Comedor al que pertenece el profesor	
<b>Prioridad</b>	Secundario

Tabla 2.27 Descripción Textual CU Reservar Alimentación

<b>Caso de Uso</b>	<b>Solicitar Pase Especial</b>
<b>Actor</b>	Profesor Residente (Inicia)
<b>Propósito</b>	Permitir que los profesores residentes puedan realizar solicitudes de pases especiales por más de tres días para sus visitantes
<b>Resumen</b>	El caso de uso se inicia cuando el profesor residente accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Solicitar un Pase Especial e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Precondiciones</b>	El usuario autenticado debe ser un profesor residente
<b>Poscondiciones</b>	Queda registrada la información relacionada con las reservaciones de pases especiales
<b>Referencia</b>	R6
<b>Casos de Uso Relacionados</b>	Reservar Alimentación <<extend>>
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1.El profesor residente accede al sistema	1.1 El sistema le muestra las opciones que le corresponden según sus permisos
2.El profesor residente selecciona la acción "Solicitar Pase Especial"	2.1 El sistema muestra los datos del profesor que tomó de la autenticación 2.2 El sistema muestra los campos a llenar sobre

<p>3. El profesor residente llena los campos</p> <p>4. Si el profesor residente desea realizar una reservación de alimentos Ver Caso de Uso Reservar Alimentación</p> <p>5. El profesor residente selecciona la opción “Aceptar”</p>	<p>el visitante:</p> <ul style="list-style-type: none"> <li>✓ nombre</li> <li>✓ carnet de identidad</li> <li>✓ vínculo familiar</li> <li>✓ sexo</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ Causa</li> </ul> <p>5.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>5.2 El sistema verifica que los datos a entrar sean del tipo correcto</p> <p>5.3 El sistema verifica que la solicitud cumpla con las restricciones en cuanto a parentesco y sexo</p> <p>5.4 El sistema verifica que ese profesor residente no tenga otra reservación con la que tenga algún día en común</p> <p>5.5 El sistema almacena la información relacionada con la solicitud</p> <p>5.6 El sistema muestra un mensaje notificando la efectividad de su solicitud</p>
<p><b>Prototipo de Interfaz de Usuario</b></p>	

The screenshot shows a web form titled "Solicitar Pase Especial" with the following fields and callouts:

- 1**: Nombre (Text field)
- 2**: CI (Text field)
- 3**: Parentesco (Dropdown menu)
- 4**: Sexo (Dropdown menu)
- 5**: Ciudadanía (Text field, value: cubana)
- 6**: Fecha de Llegada (Date picker)
- 7**: Fecha de Salida (Date picker)
- 8**: Causa (Text area)
- 9**: Reservar (Submit button)
- 10**: Cancelar (Cancel button)

[1] Campo de Texto para introducir el nombre del visitante
[2] Campo de Texto para introducir el CI del visitante
[3] Campo de Texto para seleccionar el parentesco del visitante
[4] Campo de Texto para seleccionar el sexo del visitante
[5] Campo de Texto no modificable
[6] Campos de Textos para seleccionar la fecha de llegada del visitante
[7] Campos de Textos para seleccionar la fecha de salida del visitante
[8] Campo de Texto para exponer la causa
[9] Botón que inicia las acciones para solicitar
[10] Botón que regresa a la página anterior

### Flujos Alternos

5. Si el profesor residente selecciona la opción "Cancelar", el sistema lo envía a la página anterior	
	5.1 Si el sistema verifica que algún campo obligatorio esta vacío emite un mensaje de error
	5.2 Si el sistema verifica que los datos a entrar son de tipo incorrecto emite un mensaje de error
	5.3 Si el sistema verifica que la reservación no cumple con alguna de las restricciones en cuanto a parentesco o sexo emite un mensaje de error
	5.4 Si el sistema verifica que ese profesor

	residente tiene otra reservación con la que coincida algún día emite un mensaje de error
<b>Prioridad</b>	Crítico

**Tabla 2.28 Descripción Textual CU Solicitar Pase Especial**

<b>Caso de Uso</b>	<b>Aprobar Pase Especial</b>
<b>Actor</b>	Vicerrector de Residencia de Profesores (Inicia)
<b>Propósito</b>	Permitir que los profesores residentes obtengan la aprobación para que su visitante pueda permanecer en la Universidad por un período de tiempo mayor que el establecido (3 días)
<b>Resumen</b>	El caso de uso se inicia cuando el Vicerrector de Residencia de Profesores accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de “Mostrar Solicitudes de Pases”, luego de analizar las causas expuestas puede aprobar o denegar dichas solicitudes. El Caso de uso finaliza con un mensaje que notifica la realización de la acción
<b>Precondiciones</b>	El usuario autenticado debe ser el Vicerrector de Residencia de Profesores
<b>Poscondiciones</b>	Queda actualizada la información relacionada con las reservaciones de pases
<b>Referencia</b>	R7
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1.El Vicerrector de Residencia accede al sistema	1.1 El sistema le muestra las opciones que le corresponden según sus permisos
2.El Vicerrector de Residencia selecciona la opción “Mostrar Solicitudes de Pases”	2.1 El sistema muestra los datos de las

<p>3. El Vicerrector de Residencia aprueba la solicitud cuya causa sea justificada y de esta forma queda reservada la misma</p>	<p>reservaciones cuyo estado es “solicitada”</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ solapín</li> <li>✓ Nombre del visitante</li> <li>✓ carnet de identidad</li> <li>✓ vínculo familiar</li> <li>✓ sexo</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ Causa</li> </ul> <p>3.1 El sistema muestra un mensaje notificando la efectividad de su acción</p>
<p><b>Flujos Alternos</b></p>	
	<p>1.1 Si no existen solicitudes de pases especiales el sistema muestra un mensaje de información</p>
<p>3. Si el Vicerrector de Residencia no considera justificada la causa expuesta deniega la solicitud</p>	<p>3.1 El sistema muestra un mensaje notificando la efectividad de su acción</p>
<p>3. Si el Vicerrector de Residencia selecciona la opción “Cancelar”, el sistema lo envía a la página anterior</p>	
<p><b>Prioridad</b></p>	<p>Crítico</p>

Tabla 2.29 Descripción Textual CU Aprobar Pase Especial

<p><b>Caso de Uso</b></p>	<p><b>Mostrar Visitantes en Residencia</b></p>
<p><b>Actor</b></p>	<p>Instructora (Inicia)</p>
<p><b>Propósito</b></p>	<p>Permitir un mayor control sobre la estancia de personal ajeno a la Universidad dentro de la misma</p>
<p><b>Resumen</b></p>	<p>El caso de uso se inicia cuando la Instructora</p>

	accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Mostrar Visitantes en Residencia e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Precondiciones</b>	El usuario autenticado debe ser Instructora
<b>Poscondiciones</b>	Se dispone de la información relacionada con la estancia de visitantes en la residencia de profesores
<b>Referencia</b>	R8
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. La Instructora accede al sistema</p> <p>2. La Instructora selecciona la opción "Mostrar Visitantes en Residencia"</p> <p>3. La Instructora llena los campos</p> <p>4. La instructora selecciona la opción "Aceptar"</p>	<p>1.1 El sistema le muestra las opciones que le corresponden según sus permisos</p> <p>2.1 El sistema muestra los campos a llenar</p> <ul style="list-style-type: none"> <li>✓ número del edificio (mostrando los existentes en el sistema)</li> </ul> <p>4.1 El sistema verifica que ningún campo obligatorio este vacío</p> <p>4.2 El sistema verifica que exista alguna reservación en ese edificio cuyo estado sea "en curso"</p> <p>4.3 El sistema muestra todas las reservaciones que cumplan con dicha condición, los datos que se deberán mostrar son los siguientes:</p> <ul style="list-style-type: none"> <li>✓ número de solapín</li> <li>✓ nombre del profesor</li> </ul>

	<ul style="list-style-type: none"> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> </ul>
<b>Flujos Alternos</b>	
	4.1 Si el sistema verifica que algún campo obligatorio este vacío emite un mensaje de error
	4.2 Si el sistema verifica que no existe ninguna reservación que se esté llevando a cabo en ese edificio emite un mensaje de información
<b>Prioridad</b>	Secundario

**Tabla 2.30 Descripción Textual CU Mostrar Visitantes en Residencia**

<b>Caso de Uso</b>	<b>Mostrar Reportes de Pases</b>
<b>Actor</b>	Asistente de Profesores (Inicia)
<b>Propósito</b>	Permitir reportar reservaciones de pase: <ul style="list-style-type: none"> <li>✓ de un Profesor</li> <li>✓ de un Profesor en un mes</li> <li>✓ en un mes</li> </ul>
<b>Resumen</b>	El caso de uso se inicia cuando la Asistente de Profesores accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona el reporte que desea obtener e introduce los datos necesarios para hacerlo efectivo. El caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Precondiciones</b>	El usuario autenticado debe ser Asistente de Profesores
<b>Poscondiciones</b>	Se dispone de la información relacionada con las reservaciones de pase

<b>Referencia</b>	R8
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. La Asistente de Profesores accede al sistema</p> <p>2. La Asistente de Profesores selecciona el reporte que desea obtener</p>	<p>1.1 El sistema le muestra las opciones que le corresponden según sus permisos</p> <p>2.1 Si La Asistente de Profesores selecciona:</p> <p>a) "Reservaciones de Pase de un Profesor" ir a la Sección con el mismo nombre</p> <p>b) "Reservaciones de Pase de un Profesor en un mes" ir a la Sección con el mismo nombre</p> <p>c) "Reservaciones de Pase en un mes" ir a la Sección con el mismo nombre</p>
<b>Sección "Reservaciones de Pase de un Profesor"</b>	
<p>3. La Asistente de Profesores registra el número del solapín del profesor</p> <p>4. La Asistente de Profesores selecciona la opción "Aceptar"</p>	<p>2.1 El sistema muestra el campo a llenar con el número del solapín del profesor</p> <p>4.1 El sistema verifica que el campo no esté vacío</p> <p>4.2 El sistema verifica que el solapín registrado pertenezca a algún profesor que haya realizado al menos una reservación de pase</p> <p>4.3 El sistema muestra todas las reservaciones realizadas por el profesor, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ edificio</li> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> </ul>

	<ul style="list-style-type: none"> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ estado</li> </ul>
<b>Flujos Alternos</b>	
4. Si el Asistente de Profesores selecciona la opción "Cancelar", el sistema lo envía a la página anterior	
	4.1 Si el sistema verifica que el campo esta vacío emite un mensaje de error
	4.2 Si el sistema verifica que el solapín registrado no pertenece a algún profesor que haya realizado al menos una reservación de pase se emite un mensaje de información
<b>Sección</b> "Reservaciones de Pase de un Profesor en un mes"	
<p>3. La Asistente de Profesores llena los campos</p> <p>4. La Asistente de Profesores selecciona la opción "Aceptar"</p>	<p>2.1 El sistema muestra los campos a llenar:</p> <ul style="list-style-type: none"> <li>✓ número del solapín del profesor</li> <li>✓ mes que interesa</li> </ul> <p>4.1 El sistema verifica que ningún campo esté vacío</p> <p>4.2 El sistema verifica que el solapín registrado pertenezca a algún profesor que haya realizado al menos una reservación de pase en el mes en cuestión</p> <p>4.3 El sistema muestra todas las reservaciones realizadas por el profesor en el mes especificado, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ edificio</li> </ul>

	<ul style="list-style-type: none"> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ estado</li> </ul>
--	---

### Prototipo de Interfaz de Usuario

Solapín:  Mes:

NombreP	Edif	Apto	Telef	NombreV	CIdentidadV	f_Reserv	f_Legada	f_Salida	Estado

Aceptar Cancelar

[1] Campo de texto para insertar el número del solapín
[2] Campo de texto para seleccionar el mes
[3] Datos a mostrar en cada reporte
[4] Botón que inicia las actividades de Reportar
[5] Botón que regresa a la página anterior

### Flujos Alternos

4. Si el Asistente de Profesores selecciona la opción "Cancelar", el sistema lo envía a la página anterior	
	4.1 Si el sistema verifica que algún campo esta vacío emite un mensaje de error
	4.2 Si el sistema verifica que el solapín registrado no pertenece a algún profesor que haya realizado al menos una reservación de pase en el mes

	especificado se emite un mensaje de información
<b>Sección</b> “Reservaciones de Pase en un mes”	
<p>3. La Asistente de Profesores llena el campo</p> <p>4. La Asistente de Profesores selecciona la opción “Aceptar”</p>	<p>2.1 El sistema muestra el campo a llenar con el mes que interesa</p> <p>4.1 El sistema verifica que el campo no este vacío</p> <p>4.2 El sistema verifica que en el mes en cuestión se hayan realizado reservaciones de pase</p> <p>4.3 El sistema muestra todas las reservaciones realizadas en el mes en cuestión, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ edificio</li> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ estado</li> </ul>
<b>Flujos Alternos</b>	
4. Si el Asistente de Profesores selecciona la opción “Cancelar”, el sistema lo envía a la página anterior	
	4.1 Si el sistema verifica que algún campo está vacío emite un mensaje de error
	4.2 Si el sistema verifica que en el mes en cuestión no se realizaron reservaciones se emite un mensaje de información
<b>Prioridad</b>	Secundario

**Tabla 2.31 Descripción Textual CU Mostrar Reportes de Pases**

Caso de Uso	Mostrar Reportes de Alimentación
<b>Actor</b>	Asistente de Control (Inicia)
<b>Propósito</b>	Permitir reportar reservaciones de alimentos: <ul style="list-style-type: none"> <li>✓ de un Profesor</li> <li>✓ de un Profesor en un mes</li> <li>✓ en un mes</li> </ul>
<b>Resumen</b>	El caso de uso se inicia cuando la Asistente de Control accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona el reporte que desea obtener e introduce los datos necesarios para hacerlo efectivo. El caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Precondiciones</b>	El usuario autenticado debe ser la Asistente de Control
<b>Poscondiciones</b>	Se dispone de las estadísticas relacionadas con las reservaciones de alimentos
<b>Referencia</b>	R8
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. La Asistente de Control accede al sistema	1.1 El sistema le muestra las opciones que le corresponden según sus permisos
2. La Asistente de Control selecciona el reporte que desea obtener	2.1 Si La Asistente de Control selecciona: <ul style="list-style-type: none"> <li>a) “Reportar Reservaciones de Alimentos de un Profesor” ir a la Sección con el mismo nombre</li> <li>b) “Reservaciones de Alimentos de un Profesor en un mes” ir a la Sección con el mismo nombre</li> </ul>

	c) “Reservaciones de Alimentos en un mes” ir a la Sección con el mismo nombre
<b>Sección</b> “Reservaciones de Alimentos de un Profesor”	
3. La Asistente de Control registra el número del solapín del profesor	<p>2.1 El sistema muestra el campo a llenar con el número del solapín del profesor</p> <p>3.1 El sistema verifica que ningún campo esté vacío</p> <p>3.2 El sistema verifica que el solapín registrado pertenezca a algún profesor que haya realizado al menos una reservación de pase</p> <p>3.3 El sistema muestra todas las reservaciones de alimentos realizadas por el profesor, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ desayuno</li> <li>✓ almuerzo</li> <li>✓ comida</li> <li>✓ complejo comedor</li> <li>✓ estado</li> </ul>
<b>Flujos Alternos</b>	
	<p>3.1 Si el sistema verifica que el campo esta vacío emite un mensaje de error</p> <p>3.2 Si el sistema verifica que el solapín registrado no pertenece a algún profesor que haya realizado</p>

	al menos una reservación de alimentos se emite un mensaje de información
<b>Sección</b> “Reservaciones de Alimentos de un Profesor en un mes”	
3. La Asistente de Control llena los campos	<p>2.1 El sistema muestra los campos a llenar:</p> <ul style="list-style-type: none"> <li>✓ número del solapín del profesor</li> <li>✓ mes que interesa</li> </ul> <p>3.1 El sistema verifica que ningún campo esté vacío</p> <p>3.2 El sistema verifica que el solapín registrado pertenezca a algún profesor que haya realizado al menos una reservación de alimentos en el mes en cuestión</p> <p>3.3 El sistema muestra todas las reservaciones de alimentos realizadas por el profesor en el mes en cuestión, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ desayuno</li> <li>✓ almuerzo</li> <li>✓ comida</li> <li>✓ complejo comedor</li> <li>✓ estado</li> </ul>
<b>Prototipo de Interfaz de Usuario</b>	

Solapin:  Mes:

NombreP	Edif	Apto	Telef	NombreV	CI	IdentidadV	f_Reserv	f_Legada	f_Salida	D	A	C	CCom	Estado

Aceptar Cancelar

[1] Campo de texto para insertar el número del solapín

[2] Campo de texto para seleccionar el mes

[3] Datos a mostrar en cada reporte

[4] Botón que inicia las actividades de Reportar

[5] Botón que regresa a la página anterior

### Flujos Alternos

- 3.1 Si el sistema verifica que algún campo esta vacío emite un mensaje de error
- 3.2 Si el sistema verifica que el solapín registrado no pertenece a algún profesor que haya realizado al menos una reservación de alimentos en el mes en cuestión emite un mensaje de información

### Sección "Reservaciones de Alimentos en un mes"

3. La Asistente de Control llena el campo

- 2.1 El sistema muestra el campo a llenar con el mes interesa
- 3.1 El sistema verifica que ningún campo este vacío
- 3.2 El sistema verifica que en el mes en cuestión se hayan realizado reservaciones de alimentos
- 3.3 El sistema muestra todas las reservaciones de alimentos realizadas en el mes en cuestión, en las que se presentan los siguientes datos:
  - ✓ nombre del profesor
  - ✓ apartamento

	<ul style="list-style-type: none"> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ desayuno</li> <li>✓ almuerzo</li> <li>✓ comida</li> <li>✓ complejo comedor</li> <li>✓ estado</li> </ul>
<b>Flujos Alternos</b>	
	<p>3.1 Si el sistema verifica que algún campo esta vacío emite un mensaje de error</p> <p>3.2 Si el sistema verifica que en el mes en cuestión no se realizó ninguna reservación de alimentos emite un mensaje de información</p>
<b>Prioridad</b>	Secundario

**Tabla 2.32 Descripción Textual CU Mostrar reportes de Alimentación**

<b>Caso de Uso</b>	<b>Reportar Pases por Fechas</b>
<b>Actor</b>	Técnico General (Inicia)
<b>Propósito</b>	Permitir conocer las reservaciones de pase realizadas dentro del rango de fechas especificado
<b>Resumen</b>	El caso de uso se inicia cuando el Técnico General accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Reportar Pases en un Período e introduce los datos

	necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Precondiciones</b>	El usuario autenticado debe ser el Técnico General
<b>Poscondiciones</b>	Se dispone de la información para permitir la entrada a la Universidad
<b>Referencia</b>	R8
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El Técnico General accede al sistema</p> <p>2. El Técnico General selecciona la opción de “Reportar Pases en un Período”</p> <p>3. El Técnico General llena los campos</p>	<p>1.1 El sistema le muestra las opciones que le corresponden según sus permisos</p> <p>2.1 El sistema muestra los campos a llenar con las fechas solicitadas</p> <p>3.1 El sistema verifica que ningún campo este vacío</p> <p>3.2 El sistema verifica que exista alguna reservación de pase dentro de ese período</p> <p>3.3 El sistema muestra todas las reservaciones de pases realizadas dentro de ese período, en las que se presentan los siguientes datos:</p> <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ edificio</li> <li>✓ apartamento</li> <li>✓ teléfono</li> <li>✓ nombre del visitante</li> <li>✓ vínculo familiar</li> <li>✓ carnet de identidad del visitante</li> <li>✓ fecha de reservación</li> </ul>

	<ul style="list-style-type: none"> <li>✓ fecha de llegada</li> <li>✓ fecha de salida</li> <li>✓ estado</li> </ul>
<b>Prototipo de Interfaz de Usuario</b>	
<b>[1]</b>	Botón para seleccionar la fecha de llegada
<b>[2]</b>	Botón para seleccionar la fecha de salida
<b>[3]</b>	Datos a mostrar en cada reporte
<b>[4]</b>	Botón que inicia las actividades de Reportar
<b>[5]</b>	Botón que regresa a la página anterior
<b>Flujos Alternos</b>	
	3.1 Si el sistema verifica que algún campo está vacío emite un mensaje de error
	3.2 Si el sistema verifica que no existe ninguna reservación de pase dentro de ese periodo emite un mensaje de información
<b>Prioridad</b>	Secundario

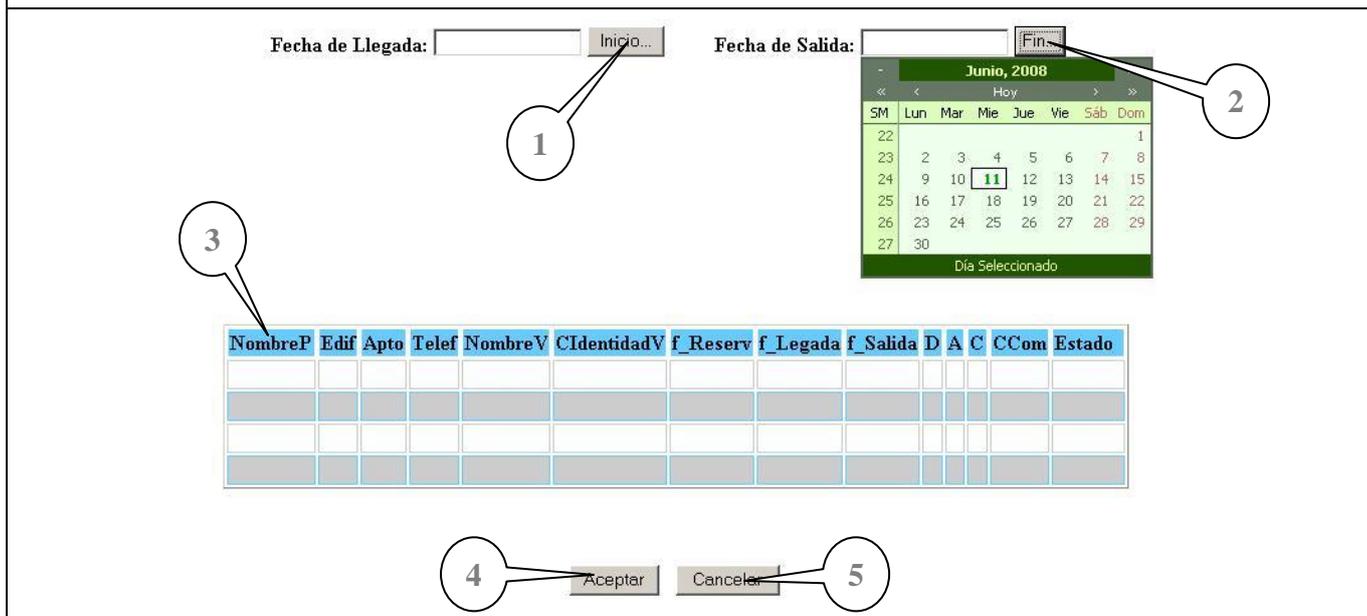
Tabla 2.33 Descripción Textual CU Reportar Pases por Fechas

<b>Caso de Uso</b>	<b>Reportar Alimentación por Fechas</b>
<b>Actor</b>	Económica General (Inicia)
<b>Propósito</b>	Permitir conocer las reservaciones de

	alimentación realizadas en el rango de fechas especificado
<b>Resumen</b>	El caso de uso se inicia cuando la Económica General accede a la página donde se le muestran las opciones que le corresponden según sus permisos y selecciona la opción de Mostrar Alimentación en un Período e introduce los datos necesarios para hacerla efectiva. El Caso de uso finaliza cuando el sistema muestra los resultados de la búsqueda
<b>Precondiciones</b>	El usuario autenticado debe ser la Económica General
<b>Poscondiciones</b>	Se dispone de la información para entregarle los tickets a los profesores
<b>Referencia</b>	R8
<b>Casos de Uso Relacionados</b>	--
<b>Flujo Normal de Eventos</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. La Económica General accede al sistema  2. La Económica General selecciona la opción de "Reportar Alimentación en un periodo"  3. La Económica General llena los campos	1.1 El sistema le muestra las opciones que le corresponden según sus permisos  2.1 El sistema muestra los campos a llenar con las fechas solicitadas  3.1 El sistema verifica que ningún campo este vacío  3.2 El sistema verifica que exista alguna reservación de de alimentos dentro de ese período  3.3 El sistema muestra todas las reservaciones realizadas en ese rango de fechas, en las que se presentan los siguientes datos: <ul style="list-style-type: none"> <li>✓ nombre del profesor</li> <li>✓ apartamento</li> </ul>

- ✓ teléfono
- ✓ nombre del visitante
- ✓ vínculo familiar
- ✓ carnet de identidad del visitante
- ✓ fecha de reservación
- ✓ fecha de llegada
- ✓ fecha de salida
- ✓ desayuno
- ✓ almuerzo
- ✓ comida
- ✓ complejo comedor
- ✓ estado

**Prototipo de Interfaz de Usuario**



[1] Botón para seleccionar la fecha de llegada

[2] Botón para seleccionar la fecha de salida

[3] Datos a mostrar en cada reporte

[4] Botón que inicia las actividades de Reportar

[5] Botón que regresa a la página anterior

**Flujos Alternos**

3.1 Si el sistema verifica que algún campo está

	vacío emite un mensaje de error
	3.2 Si el sistema verifica que no existe ninguna reservación de alimentos dentro de ese periodo emite un mensaje de información
<b>Prioridad</b>	Secundario

**Tabla 2.34 Descripción Textual CU Reportar Alimentación por Fechas**

### 2.11 Conclusiones

En este capítulo se han modelado en términos de casos de uso los procesos que se llevan a cabo en el negocio existente, identificando además los actores y trabajadores que intervienen en los mismos y describiendo las actividades que los componen, lo que sirvió de base para determinar las funcionalidades necesarias con que la solución debe cumplir y modelar una propuesta en términos de casos de uso. Concluye así este capítulo, dando paso al diseño del sistema.

# CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

## 3.1 Introducción

En el presente capítulo analizamos los requisitos que se describieron en la captura de requisitos, refinándolos y estructurándolos, a través del Modelo de Análisis al realizar los diagramas de clases del análisis y luego comenzar a dar forma al sistema en su totalidad mediante el Modelo de Diseño al elaborar los diagramas de clases de diseño Web y los diagramas de secuencia. Además se realiza el diseño de la base de datos mediante los diagramas lógico y físico de los datos y se abordan otros aspectos relacionados con el diseño, la seguridad y el tratamiento de errores en la aplicación.

## 3.2 Análisis del Sistema

El análisis consiste en obtener una visión del sistema que se preocupa de ver que hace, o sea, su objetivo es comprender perfectamente los requisitos de software y no precisar como se implementará la solución, interesándose únicamente por los requisitos funcionales y utilizando como lenguaje el modelo de análisis.

### 3.2.1 Modelo de Análisis

Un Modelo de Análisis proporciona una vista interna del sistema, ofrece una especificación mas precisa de los requisitos, estructurándolos de modo que se facilite su comprensión, preparación, modificación y en general, su mantenimiento. El mismo es descrito utilizando el lenguaje de los desarrolladores y se considera como una primera aproximación al modelo de diseño.

#### Ventajas del Modelo de Análisis:

- ✓ Ayuda en la transición al diseño
- ✓ Apoya el cambio de tecnología de programación
- ✓ Sirve para mantener una visión general del sistema
- ✓ Sirve para planificar y dividir el diseño e implementación en pequeños módulos
- ✓ Apoya la aplicación de reingeniería a aplicaciones existentes

#### Clases del Modelo de Análisis

El Modelo de Análisis contiene clases del análisis las que siempre encajan en tres estereotipos básicos:

- ✓ Entidad: Modelan información que posee larga vida y que es a menudo persistente
- ✓ Interfaz: Modelan la interacción entre el sistema y sus actores
- ✓ Control: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso

### 3.2.2 Diagrama de clases del análisis

Se realizó un diagrama de clases del análisis por caso de uso del sistema descrito en el capítulo anterior, donde se refleja la interacción entre todas las clases que intervienen en cada diagrama. A continuación se presenta el más significativo, los restantes se podrán encontrar en el [Anexo2](#)

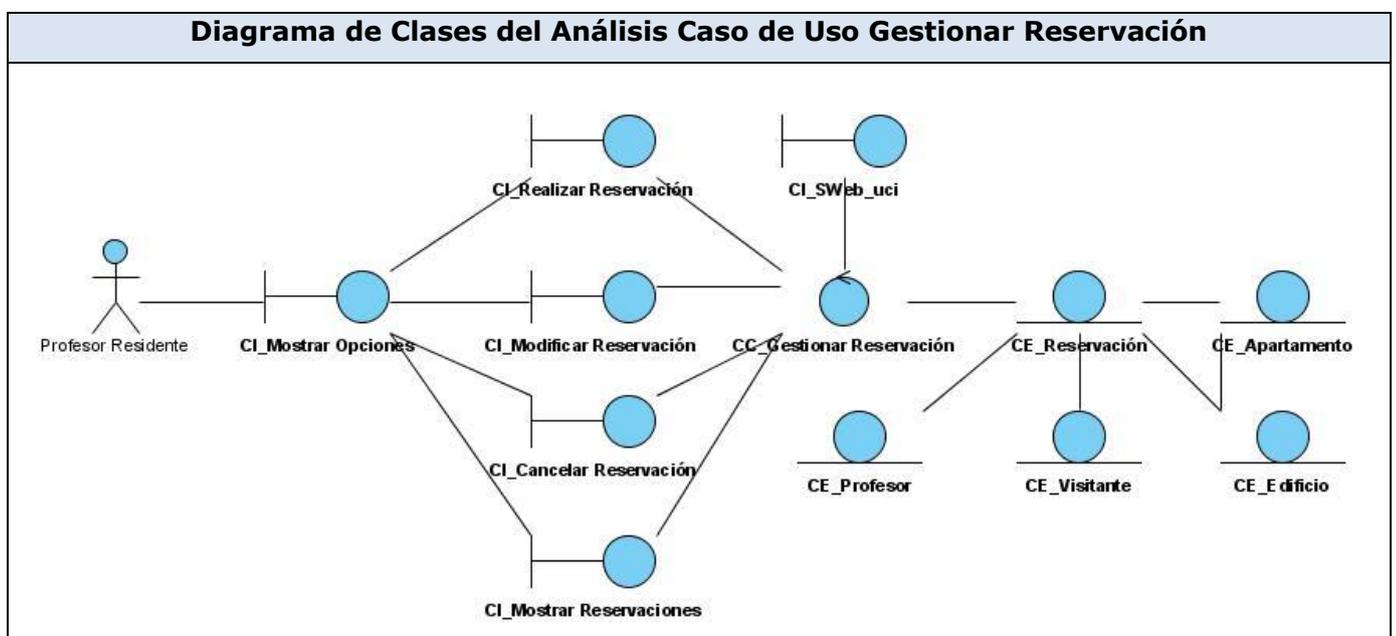


Figura 3.4 Diagrama de Clases del Análisis Caso de Uso Gestionar Reservación

### 3.3 Diseño del Sistema

El Diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales y restricciones relacionadas principalmente con el lenguaje de programación, además debe crear una entrada apropiada y un punto de partida para las actividades de implementación y utilizando como manifiesto el modelo de diseño.

#### 3.3.1 Patrones de Diseño

Para realizar las actividades de diseño es de suma importancia tener en cuenta a los patrones de diseño. Los mismos son soluciones simples y elegantes a problemas específicos y comunes del diseño

orientado a objetos. Con ellos la modelación es mucho más flexible y reutilizable, además facilitan la comunicación pues establecen un marco de referencia, dígase terminología, justificación, y la especificación para las actividades de implementación incluyendo la reutilización de código. En el diseño del sistema se utilizarán los patrones generales de software para asignación de responsabilidades GRASP aunque estos más que patrones, son una serie de buenas prácticas de aplicación para el diseño de software, los que se describen a continuación:

**Experto:** Se encarga de asignar una responsabilidad a la clase que tiene la información necesaria para cumplirla, siendo este el principio fundamental en virtud del cual se le asignan las responsabilidades a los objetos.

**Creador:** Se encarga de identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases.

**Controlador:** Sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado.

**Alta cohesión:** Nos dice que la información que almacena una clase debe de ser coherente y está en la mayor medida de lo posible relacionada con la clase.

**Bajo acoplamiento:** Se basa en lograr tener las clases lo menos ligadas entre sí posible. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

Siendo estos dos últimos de suma importancia para la programación en capas para lograr que la aplicación sea fácilmente desarmable y sea más sencillo realizar mejoras y actualizaciones al sistema.

### 3.3.2 Diagramas de interacción

Los diagramas de interacción muestran cómo se comunican los objetos en una interacción. Existen dos tipos de diagramas de interacción: el Diagrama de Colaboración y el Diagrama de Secuencia.

### **Diagramas de Secuencia**

En el diseño, es preferible realizar los diagramas de secuencia ya que el centro de atención principal es encontrar secuencias de interacciones detalladas y ordenadas en el tiempo. Los Diagramas de secuencia muestran las interacciones entre objetos mediante transferencia de mensajes entre los mismos. A continuación se presenta el más significativo, los restantes se podrán encontrar en el Anexo3.

Diagrama de Secuencia Caso de Uso Gestionar Reservación

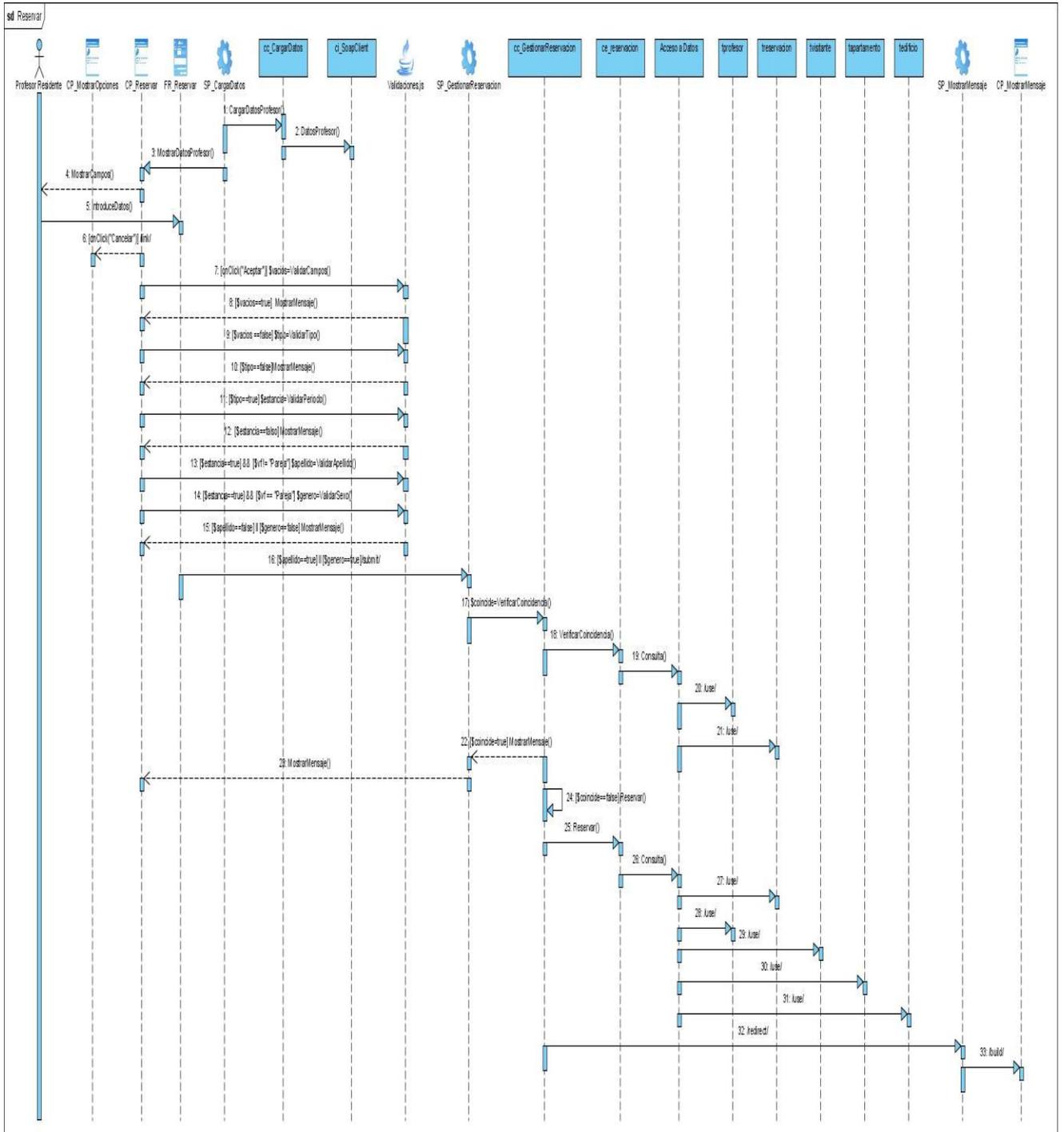


Figura 3.9 Diagrama de Secuencia Escenario “Realizar Reservación” CU Gestionar Reservación

3.3.4 Diagrama de Clases de Diseño Web

Se realizó un diagrama de clases de diseño Web por caso de uso del sistema descrito en el capítulo anterior, donde se refleja la interacción entre todas las clases que intervienen en cada diagrama, incluyendo además operaciones. Se utilizan los estereotipos UML extendiendo la notación para describir los elementos que pueden aparecer en una aplicación Web. A continuación se presenta el más significativo, los restantes se podrán encontrar en el Anexo4

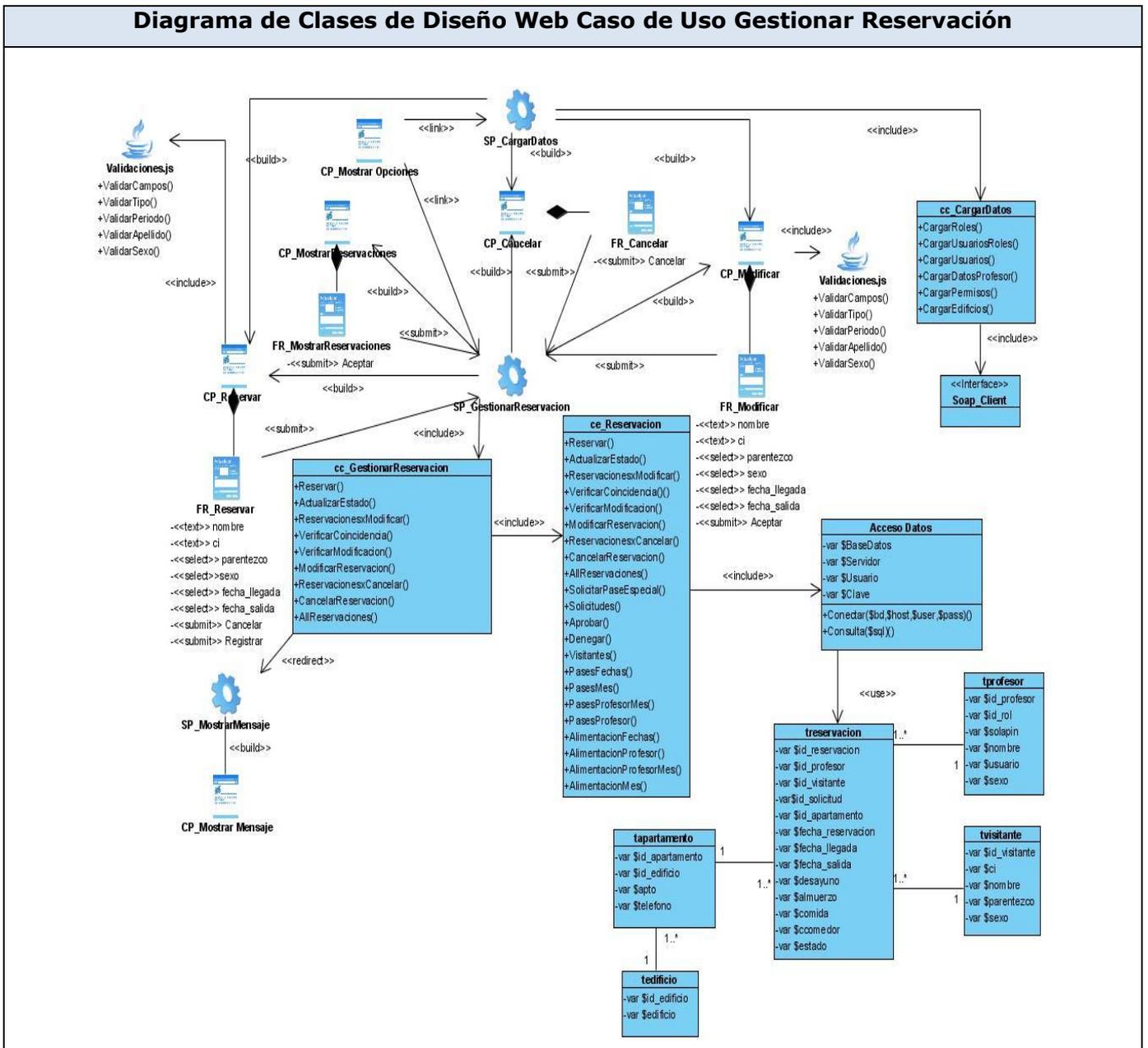


Figura 3.29 Diagrama de Clases de Diseño Web Caso de Uso Gestionar Reservación

### 3.3.5 Descripción de las clases utilizadas

La descripción de las clases utilizadas en el diseño se pueden encontrar en el [Anexo 5](#)

### 3.4 Diseño de la Base de Datos

El Diseño de la Base de datos es el proceso por el que se determina la organización de una base de datos, incluidos su estructura y contenido. [5] El mismo tiene como objetivos:

- Representar los datos que requieren las principales áreas de aplicación y los grupos de usuarios, y representar las relaciones entre dichos datos.
- Proporcionar un modelo de datos que soporte las transacciones que se vayan a realizar sobre los datos.
- Especificar un esquema que alcance las prestaciones requeridas para el sistema.

E incluye el diseño conceptual de la Base de Datos a través del modelo entidad-relación, el mismo está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas.

3.4.1 Modelo Entidad Relación de la Base de Datos

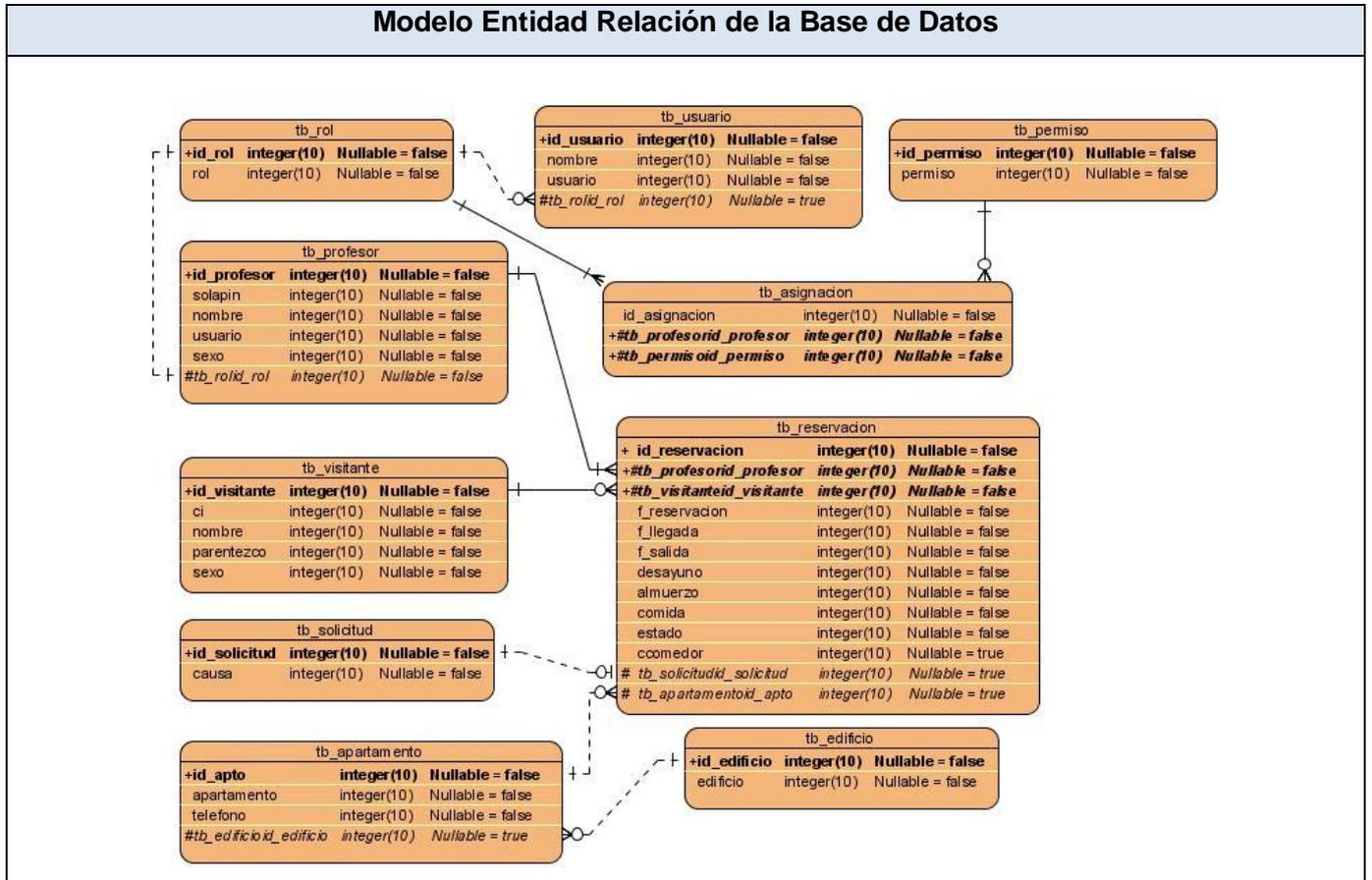


Figura 3.38 Modelo Entidad Relación de la Base de Datos

3.4.2 Descripción de las tablas

<b>Nombre:</b> tb_usuario		
<b>Descripción:</b> En esta tabla se almacenan los datos de los usuarios específicos que tendrán acceso al sistema		
Atributo	Tipo	Descripción
id_usuario	int(11)	Identificador del usuario
nombre	varchar(30)	Nombre propio del usuario
usuario	varchar(15)	Nombre del usuario
id_rol	int(11)	Identificador del rol que ocupa el usuario (llave extranjera)

Tabla 3.17 Descripción de la tabla tb\_usuario

<b>Nombre:</b> tb_rol		
<b>Descripción:</b> En esta tabla se almacenan los datos de los roles posibles a ocupar por los usuarios para acceder al sistema		
Atributo	Tipo	Descripción
id_rol	int(11)	Identificador del rol
rol	varchar(25)	Nombre del rol

**Tabla 3.18 Descripción de la tabla tb\_rol**

<b>Nombre:</b> tb_permiso		
<b>Descripción:</b> En esta tabla se almacenan los permisos posibles para un rol		
Atributo	Tipo	Descripción
id_permiso	int(15)	Identificador del permiso
permiso	varchar(50)	Nombre del permiso

**Tabla 3.19 Descripción de la tabla tb\_permiso**

<b>Nombre:</b> tb_solicitud		
<b>Descripción:</b> En esta tabla se almacena la información sobre las solicitudes de pases especiales		
Atributo	Tipo	Descripción
id_solicitud	int(11)	Identificador de la solicitud
causa	varchar(420)	Explicación de la causa de la solicitud

**Tabla 3.20 Descripción de la tabla tb\_solicitud**

<b>Nombre:</b> tb_asignacion		
<b>Descripción:</b> En esta tabla se almacenan las asignaciones entre roles y permisos		
Atributo	Tipo	Descripción
id_asignacion	int(11)	Identificador de la asignación
id_rol	int(11)	Identificador del rol
id_permiso	int(11)	Identificador del permiso

**Tabla 3.21 Descripción de la tabla tb\_asignacion**

<b>Nombre:</b> tb_profesor		
<b>Descripción:</b> En esta tabla se almacenan los datos sobre los profesores		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_profesor	int(11)	Identificador del profesor
solapin	int(11)	Solapin del profesor
nombre	varchar(30)	Nombre del profesor
usuario	varchar(15)	Usuario del profesor
sexo	varchar(10)	Sexo del profesor
id_rol	int(11)	Identificador del rol que ocupa el profesor (llave extranjera)

**Tabla 3.22 Descripción de la tabla tb\_profesor**

<b>Nombre:</b> tb_visitante		
<b>Descripción:</b> En esta tabla se almacena información relacionada con los visitantes de los profesores que realizaron la reservación		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_visitante	int(11)	Identificador del familiar
ci	int(11)	Carnet de Identidad del familiar
nombre	varchar(30)	Nombre del familiar
parentesco	varchar(10)	Parentesco del familiar con el profesor
sexo	varchar(10)	Sexo del familiar

**Tabla 3.23 Descripción de la tabla tb\_visitante**

<b>Nombre:</b> tb_reservacion		
<b>Descripción:</b> En esta tabla se almacena información relacionada con las reservaciones realizadas por los profesores residentes		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_reservacion	int(11)	Identificador de la reservación

id_profesor	int(11)	Identificador del profesor
id_visitante	int(11)	Identificador del familiar
id_solicitud	int(11)	Identificador de la solicitud
id_apto	int(11)	Identificador del apartamento
f_reservación	date	Fecha en que se realizó la reservación
f_llegada	varchar(10)	Fecha de llegada del familiar
f_salida	varchar(10)	Fecha de salida del familiar
desayuno	int(1)	Opción a solicitar
almuerzo	int(1)	Opción a solicitar
comida	int(1)	Opción a solicitar
ccomedor	int(1)	Complejo comedor al que pertenece el profesor residente que realiza la reservación
estado	varchar(12)	Estado de la reservación, puede ser: <ul style="list-style-type: none"> <li>✓ Efectiva</li> <li>✓ En curso</li> <li>✓ Pendiente</li> <li>✓ Solicitada</li> <li>✓ Denegada</li> </ul>

**Tabla 3.24 Descripción de la tabla tb\_reservacion**

<b>Nombre:</b> tb_apartamento		
<b>Descripción:</b> En esta tabla se almacena información relacionada con los apartamentos		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_apto	int(11)	Identificador del apartamento
apartamento	int(11)	Numero del apartamento
telefono	varchar(30)	Numero de telefono del apartamento
id_edificio	varchar(10)	Identificador del edificio en que

		esta el apartamento
--	--	---------------------

**Tabla 3.25 Descripción de la tabla tb\_apartamento**

<b>Nombre:</b> tb_edificio		
<b>Descripción:</b> En esta tabla se almacena información relacionada con los edificios		
Atributo	Tipo	Descripción
id_edificio	int(11)	Identificador del edificio
edificio	int(11)	Numero del edificio

**Tabla 3.26 Descripción de la tabla tb\_edificio**

### 3.5 Principios de diseño

El diseño de una aplicación requiere de una atención especial ya que éste será el que modele la interacción entre la aplicación y el usuario, y por tanto posibilitará la consecución de los objetivos perseguidos por el mismo, por lo que se debe trabajar en pro de su facilidad de uso y a la vez de una apariencia agradable, siendo conveniente además que sea técnica y visualmente coherente, bien estructurado y de apariencia profesional. Los usuarios que utilizarán el sistema deberán diferenciarse en cuanto al uso de las nuevas tecnologías por lo que se deberá desarrollar la misma en base al nivel medio que en algunos prevalece. Por todo lo anteriormente expuesto se deberán considerar en el diseño de la aplicación las siguientes características:

- ✓ Los elementos que se presenten en cada una de las pantallas no deberán ser numerosos y aquellos que se repitan deberán ser mostrados en la misma posición, tomándose siempre en cuenta el color y contraste de los mismos.
- ✓ Las páginas que muestren información, lo harán en el mismo orden y de manera similar.
- ✓ El sistema solo deberá mostrar al usuario aquellas opciones a las que dado su rol tenga permiso

### 3.6 Interfaz de Usuario

En pro de facilitar la utilización de la aplicación por los usuarios, la misma deberá presentar una interfaz sencilla y de fácil uso, estableciéndose un mismo patrón para el diseño de todas las páginas. Teniéndose especial interés en no sobrecargar las mismas con banners y controles innecesarios, se

deberá contar solamente con un banner en la cabecera que identifique la aplicación. Además una barra de menú con diferentes opciones se deberá encontrar en la parte superior de la aplicación de forma horizontal, donde pudieran incluirse otros servicios a los que el usuario tendría acceso, se recomienda trabajar con las familias de fuentes "Times New Roman, Times, serif "las que no deberán diferir mucho de 16 px y se deberá contar con un área de trabajo. En todo momento el sistema deberá dar información sobre el usuario que se ha autenticado (nombre completo). Se deberá utilizar un formulario para la autenticación de todos los usuarios mostrándose en la parte superior izquierda de la página. Los colores, fundamentalmente blanco y azul en varias tonalidades y el color rojo propuesto para los mensajes de error emitidos, con el objetivo de que sean observados sin dificultad. Poniendo en practica los aspectos anteriormente descritos se garantizara una rápida y fácil adaptación por parte de los usuarios para con el despliegue del sistema.

### 3.7 Tratamiento de Errores

Identificar y controlar los posibles errores que se pueden presentar a la hora de interactuar con el software es de vital importancia para así garantizar un correcto funcionamiento del sistema. El sistema se deberá proponer prevenir al máximo los posibles errores, sobre todo los que ocurren por parte del usuario, para ello solo se le brindaran las opciones a las que tiene acceso. Se deberá insistir en que el usuario realice la menor cantidad posible de entrada de datos aprovechando al máximo los componentes visuales de selección. Al introducir información en un formulario se deberán verificar los campos obligatorios, la veracidad y tipo de los datos. En caso de error se deberá emitir un mensaje de alerta en color rojo para que resalte fácilmente a la vista, especificando claramente en que consiste para su fácil comprensión por el usuario. Se deberán tratar estos errores de forma tal que las interacciones con la base de datos se realicen de forma correcta. Mediante la validación en el lado del cliente, se garantizara que los datos suministrados por los usuarios, se almacenen íntegros y no existan inconsistencias. Se propone utilizar el lenguaje Java Script para la implementación de las funciones encargadas del control y validación de datos. Se mostraran mensajes de confirmación luego de llevadas a cabo las distintas acciones dentro del sistema.

### 3.8 Seguridad

Se propone implementar la seguridad de forma efectiva con la utilización de servicios web. Los que se pueden encontrar en el Registro de Identidad del Directorio de Servicios Web de la Universidad, posibilitando el acceso y la utilización de información necesaria, principalmente ya que a partir de su

combinación se pueden realizar operaciones más complejas como mecanismos de comunicación e interoperabilidad.

Además de LDAP (Lightweight Directory Access Protocol) como protocolo de red que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP puede considerarse una base de datos (aunque su sistema de almacenamiento puede ser diferente) al que pueden realizarse consultas. Almacena la información de login (usuario y contraseña) y es utilizado para autenticarse aunque es posible almacenar otra información (datos de contacto del usuario, ubicación de diversos recursos de la red, permisos, certificados...). En conclusión, LDAP es un protocolo de acceso unificado a un conjunto de información sobre una red.

### **3.9 Conclusiones**

Tras obtener el Modelo de Análisis como entrada fundamental para las actividades de diseño, conservar en las mismas todo lo posible la estructura que este define, estructurar la base de datos que soportará al sistema, y profundizar en características de la interfaz de usuario y la aplicación de forma general, se ha creado una entrada apropiada y un punto de partida para actividades de implementación y prueba.

### CONCLUSIONES

Con el estudio realizado sobre las metodologías, herramientas y tecnologías en vigor actualmente para el desarrollo de sistemas, se obtuvo el diseño del mismo, información vital para que en las actividades de construcción se brinde la información necesaria sobre las reservaciones de una forma flexible y de fácil reconocimiento por todos los usuarios, teniendo como punto de partida el servicio de solicitud de entrada llevado a cabo por la Dirección de Residencia de la Universidad como proceso de negocio.

Se realizó además un análisis de las principales características de la estructura de algunos sistemas que dan la posibilidad de reservar automáticamente a nivel nacional e internacional, lo que contribuyó en la determinación de los requisitos funcionales de la aplicación a diseñar.

Con el diseño realizado se logró suplir las expectativas básicas para cada uno de los usuarios del sistema ya que a través de la metodología RUP se lograron obtener los artefactos indispensables que posibilitarán llevar a cabo las siguientes fases de desarrollo del mismo. Se considera que su utilización prodigará una mejora considerable en la calidad y eficiencia de los procesos que propone.

Se concluye con todos los objetivos específicos cumplidos para satisfacer el diseño del sistema de solicitud de entrada como objetivo general de este trabajo, solucionando así la situación problemática que dio lugar al tema de la investigación.

### **RECOMENDACIONES**

Todo desarrollo intelectual esta sujeto al cambio en pro de mejoras, por lo que se recomienda:

- ✓ Continuar con el estudio del sistema con el objetivo de añadir nuevas funcionalidades que contribuyan con el perfeccionamiento del servicio de solicitud de entrada.
- ✓ Tener en cuenta dentro de las mismas, la posibilidad de que el administrador del sistema pueda consultar el estado de los distintos usuarios y roles dentro del mismo y de que el sistema lleve a cabo el envío de avisos mediante correos electrónicos a los profesores para las actividades relacionadas con las reservaciones.
- ✓ Llevar a cabo la implementación del sistema a partir del diseño propuesto en la investigación utilizando las tecnologías seleccionadas y propuestas.
- ✓ Enlazar el sistema propuesto una vez realizado su desarrollo con el resto de los servicios de la intranet universitaria.

### BIBLIOGRAFÍA

**Graells, Dr. Pere Marquès. 2000.** LAS TIC Y SUS APORTACIONES A LA SOCIEDAD. [Online] 2000. <http://dewey.uab.es/PMARQUES/tic.htm>.

**Estadísticas, Oficina Nacional de.** USO SOCIAL DE LAS TICS. EXPERIENCIACUBANA. [Online] [http://www.itu.int/ITU-D/ict/conferences/panama06/material/29\\_Cuba\\_Presentation\\_s.pdf](http://www.itu.int/ITU-D/ict/conferences/panama06/material/29_Cuba_Presentation_s.pdf).

**Rolando Alfredo Hernández León, Sayda Coello González. 2002.** EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA. *Universidad de las Ciencias Informáticas*. [Online] Noviembre 2002. <http://revistas.mes.edu.cu/eduniv/02-Libros-por-ISBN/959-16-0400/0343-Paradigma-Cuantitativo-Investigacion-Cientifica.pdf>.

**WITHROW, JASON. 2001.** Diagramas de sitios Web (I). Planificando un espacio de información. *ProyectoWeb - Sobre Diseño de Interacción, Usabilidad y AI*. [Online] octubre 2001. <http://www.proyectoweb.cubaweb.cu/boletin/diagramas-de-sitios-web-1.html>. Boletín Nro 71.

**Permuy, Fernando Bellas.** Introducción a la Orientación a Objetos. *Departamento de Tecnologías de la Información y las Comunicaciones (TIC). Universidad de A Coruña*. [Online] <http://www.tic.udc.es/~fbellas/teaching/iao/IOO.pdf>.

**Gimeno, Alberto. 2006.** Aplicaciones web vs aplicaciones escritorio. [Online] sep 05, 2006. [http://weblogs.javahispano.org/gimenete/entry/aplicaciones\\_web\\_vs\\_aplicaciones\\_escritorio](http://weblogs.javahispano.org/gimenete/entry/aplicaciones_web_vs_aplicaciones_escritorio).

**2007.** Desarrollo de aplicaciones web sobre plataformas libres. *TICS Consulting (Servicios de Consultoría Avanzadas en Tecnologías de la Información, Comunicaciones y Seguridad Informática)*. [Online] octubre 14, 2007. <http://www.ticsconsulting.es/aplicaciones.php>.

**Ricardo J. Vargas Del Valle, Juan P. Maltés Granados.** Programación en Capas. *Universidad de Costa Rica, Ciencias de Computación e Informática, San José, Costa Rica*. . [Online] <http://www.dimare.com/adolfo/cursos/2007-2/pp-3capas.pdf>.

**Denzer, Patricio. 2002 .** PostgreSQL. [Online] octubre 23 , 2002 . <http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/denzer/informe.pdf>.

**Herrera, Álvaro. 2004.** Introducción a PostgreSQL. [Online] noviembre 13 , 2004. <http://alvherre.atentus.cl/charlas/10-Introduccion.pdf>.

**Sanchez, Ing. Informático . María A. Mendoza. 2004.** Metodologías De Desarrollo De Software. [Online] Junio 7 , 2004. [Cited: Noviembre Jueves, 29 , 2007.] [http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html).

**Cid, Miguel Angel. 2007.** Como hacemos del software un producto de ingeniería. *Programa de Ingeniería de software. Universidad Pontificia de Salamanca. Facultad de Informática.* [Online] mayo 17, 2007. <http://www.unibe.edu.do/tic/ingenieria.pdf>.

**José H. Canós, Patricio Letelier y M<sup>a</sup> Carmen Penadés.** Metodologías Ágiles en el Desarrollo de Software. *Universidad Politécnica de Valencia.* [Online] <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>.

**Jiménez, Hugo F. Arboleda. 2005 .** Modelos de ciclo de vida en desarrollo de software. [Online] October 5 , 2005 . <http://www.acis.org.co/index.php?id=551>.

**Gil, Robin Alberto Castro. 2004.** Estructura básica del proceso unificado de desarrollo de software. *Universidad Icesi.* [Online] 04 14, 2004. [http://dspace.icesi.edu.co/dspace/bitstream/item/399/1/rcastro\\_estructura-bas-puds.pdf](http://dspace.icesi.edu.co/dspace/bitstream/item/399/1/rcastro_estructura-bas-puds.pdf).

**Larman, Craig. La Habana, 2004..** *UML Y PATRONES. Volumen I. Introducción al análisis y diseño Orientado a Objetos.* La Habana, 2004.

**Grady Booch, James Rumbaugh, Ivar Jacobson.** *El Proceso Unificado de Desarrollo de Software .* 2004.

### REFERENCIAS BIBLIOGRÁFICAS

1. Arquitectura Cliente/Servidor. *Guías Técnicas Aplicables a la Contratación de Bienes y Servicios de Tecnologías de la Información y las Comunicaciones*. Ministerio de Administraciones Públicas. Consejo Superior de Informática. [Online] <http://www.csi.map.es/csi/silice/Global71.html>.
2. Bases de Datos. *FUNDAMENTOS INFORMATICOS*. [Online] Febrero 2, 2008. . <http://fundamentosinformaticosjl.wordpress.com/category/base-de-datos/>.
3. **SOLÉ, RAMON**. Una estrategia informática para la Administración Local a mediados de los años noventa. *Papers de Formació Municipal*. [Online] 1997. <http://www.pv.ccoo.es/fsap/dipuvalencia/varis/Gestio/Informatiuca%20y%20local.pdf>.
4. **Chávez, Carlos Alberto García**. Planificación, diseño y administración de bases de datos. *Curso: Diseño de base de datos relacionales*. [Online] <http://www.mailxmail.com/curso/informatica/disenobasesdatosrelacionales/capitulo6.htm>.
5. **Oktaba, Hanna**. Introducción a Patrones. *Facultad de Ciencias, UNAM*. [Online] <http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/patrones.html>.

### GLOSARIO DE TÉRMINOS

**Análisis:** Investigación de un dominio, la cual da origen a modelos que describen sus características estáticas y dinámicas. Se centra en cuestiones de “que” más que de “como”.

**Análisis Orientados a Objetos:** Investigación de un dominio o sistema de problemas, a partir de los conceptos de dominio, como tipos de objetos, asociaciones y cambios de estado.

**Aplicación:** Sistema que ofrece a un usuario final un conjunto coherente de casos de uso.

**Artefacto:** Los artefactos son productos de trabajo intermedios o finales que se producen y se utilizan durante un proyecto. Los artefactos se usan para capturar y difundir la información del proyecto.

**Asociación:** Descripción de un conjunto relacionado de enlaces o vínculos entre objetos de dos tipos.

**Caso de Uso:** Descripción narrativa textual de la secuencia de eventos y acciones que ocurren cuando un usuario parte o divide en un diálogo con un sistema durante un proceso significativo.

**Diseño:** Proceso que se sirve de los productos del análisis para generar una especificación destinada a implementar un sistema. Descripción lógica de cómo funcionará un sistema.

**Diseños Orientados a Objetos:** Especificación de una solución lógica de software a partir de objetos de software: clases, atributos, métodos y colaboraciones.

**Fase:** Período de tiempo entre dos hitos principales de un proceso de desarrollo.

**Mensaje:** Mecanismo en virtud del cual los objetos se comunican entre sí. Generalmente una respuesta para ejecutar un método.

**Metodología:** Un sistema de principios y normas generales de organización y estructuración teórico-práctica de actividades.

**Modelo:** Descripción de las características estáticas, dinámicas o ambas de un tema, presentada en

varias vistas (generalmente diagramáticas o textuales).

**Persistencia:** Almacenamiento duradero del estado de un objeto.

**Reglas del Negocio:** Describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.

**Requerimiento:** Cualquier necesidad de un área usuaria que debe cubrirse mediante una solución de tipo informática.

**Rol:** Papel, cometido o función que tiene o desempeña un actor

**Software:** Es la suma total de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de cómputo.

**Tecnología:** Aplicación de los conocimientos científicos para facilitar la realización de las actividades humanas. Supone la creación de productos, instrumentos, lenguajes y métodos al servicio de las personas.

**Usuario Autenticado:** Es aquel usuario que ha proporcionado información mediante la cual el mecanismo de seguridad garantiza su identificación al intentar acceder a los componentes del sistema.

# ANEXOS

## Anexo1. Diagramas de Actividades

Diagrama de Actividades Caso de Uso Solicitar Pase

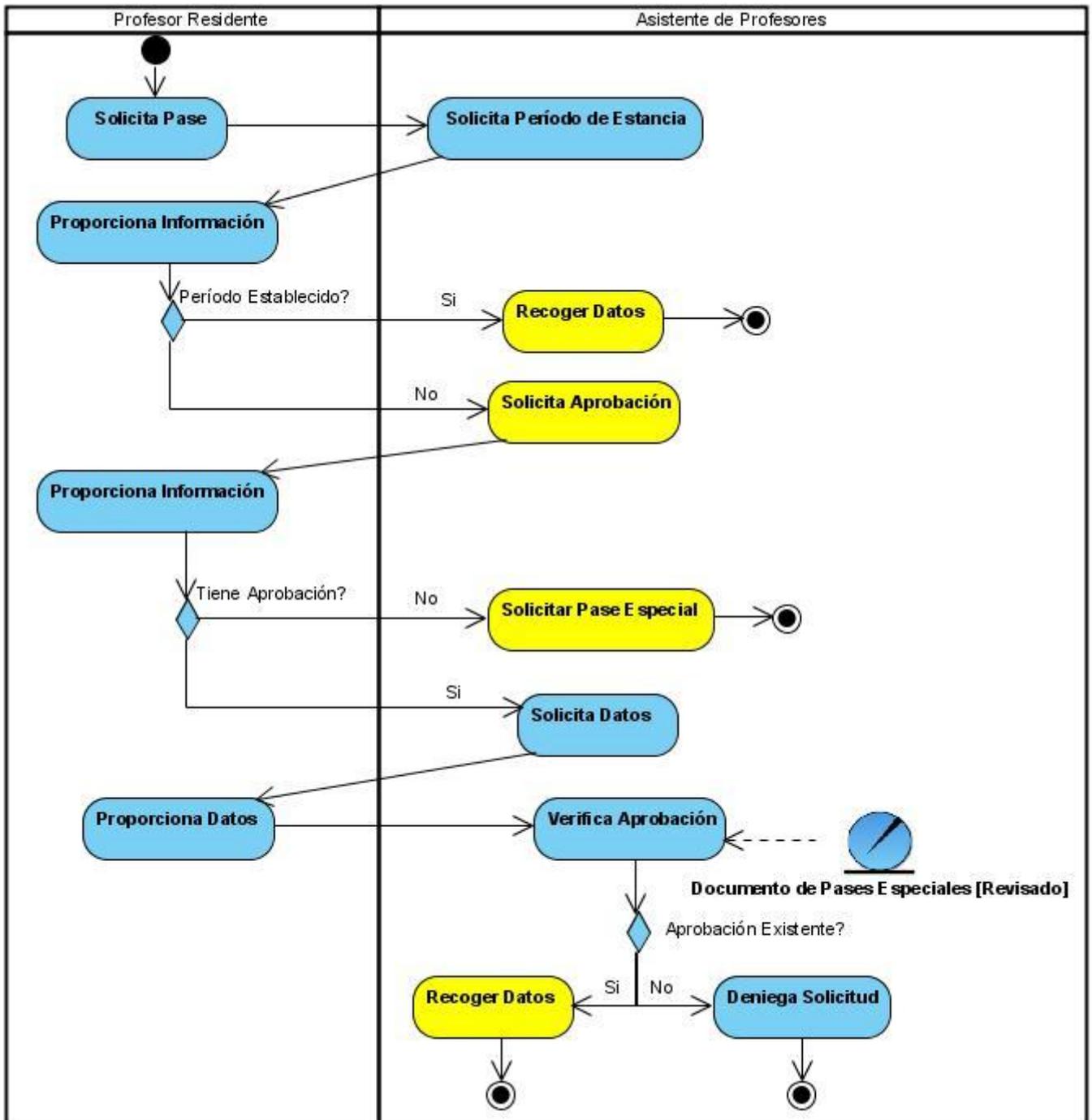


Figura 2.2 Diagrama de Actividades Caso de Uso Solicitar Pase

Anexo1. Diagramas de Actividades

Diagrama de Actividades Caso de Uso Recoger Datos

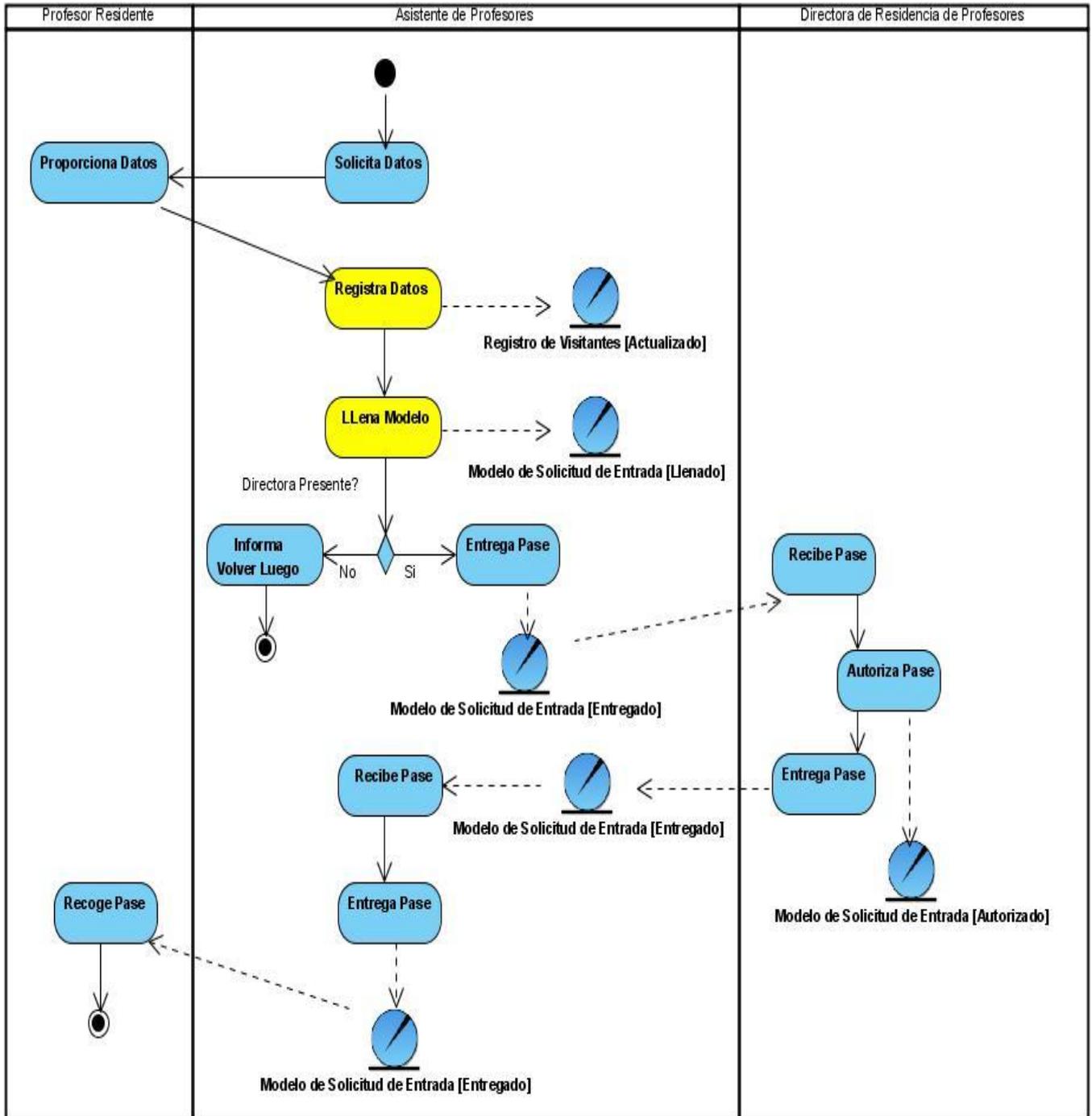


Figura 2.3 Diagrama de Actividades Caso de Uso Recoger Datos

Anexo2. Diagramas de Clases del Análisis

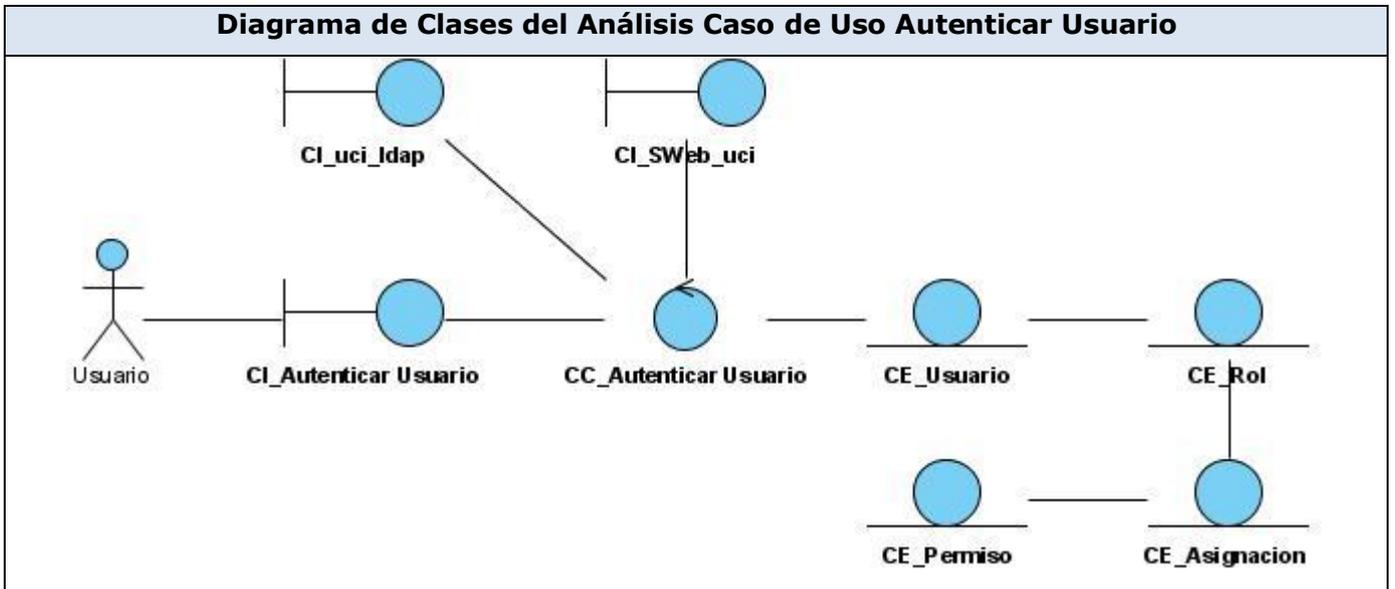


Figura 3.1 Diagrama de Clases del Análisis ases del análisis Caso de Uso Autenticar Usuario

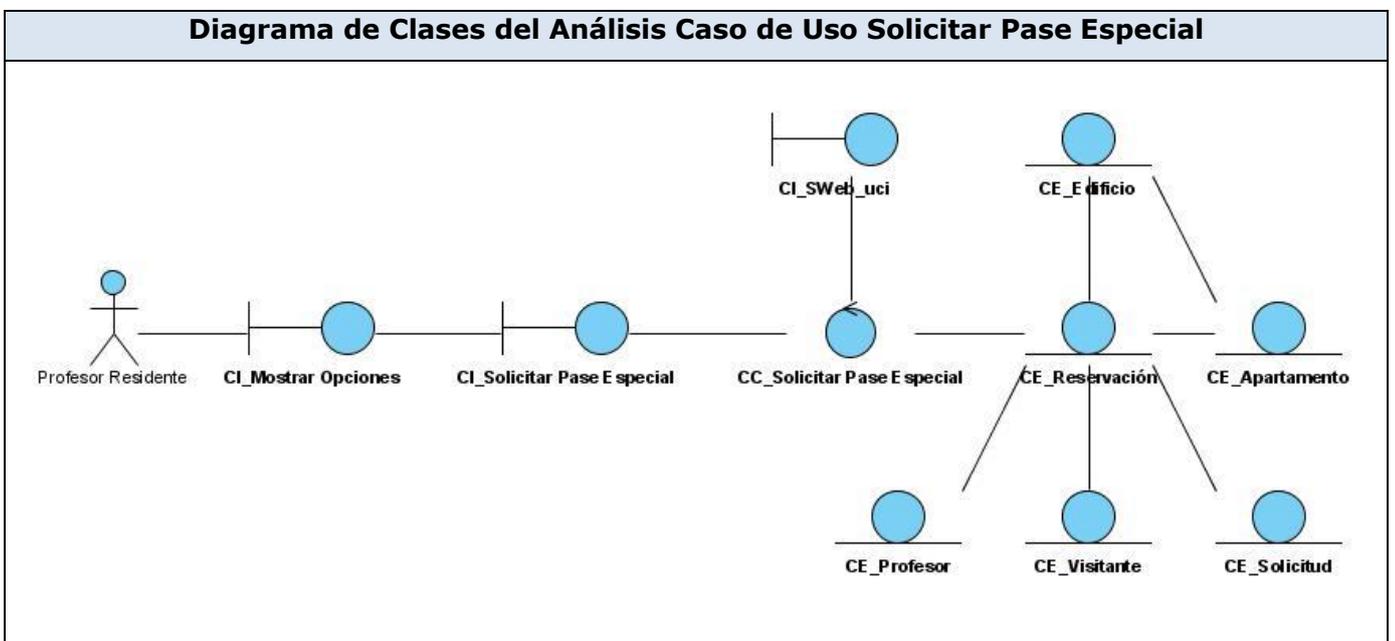


Figura 3.6 Diagrama de Clases del Análisis Caso de Uso Solicitar Pase Especial

Anexo3. Diagramas de Secuencia

Escenario "Modificar Reservación"

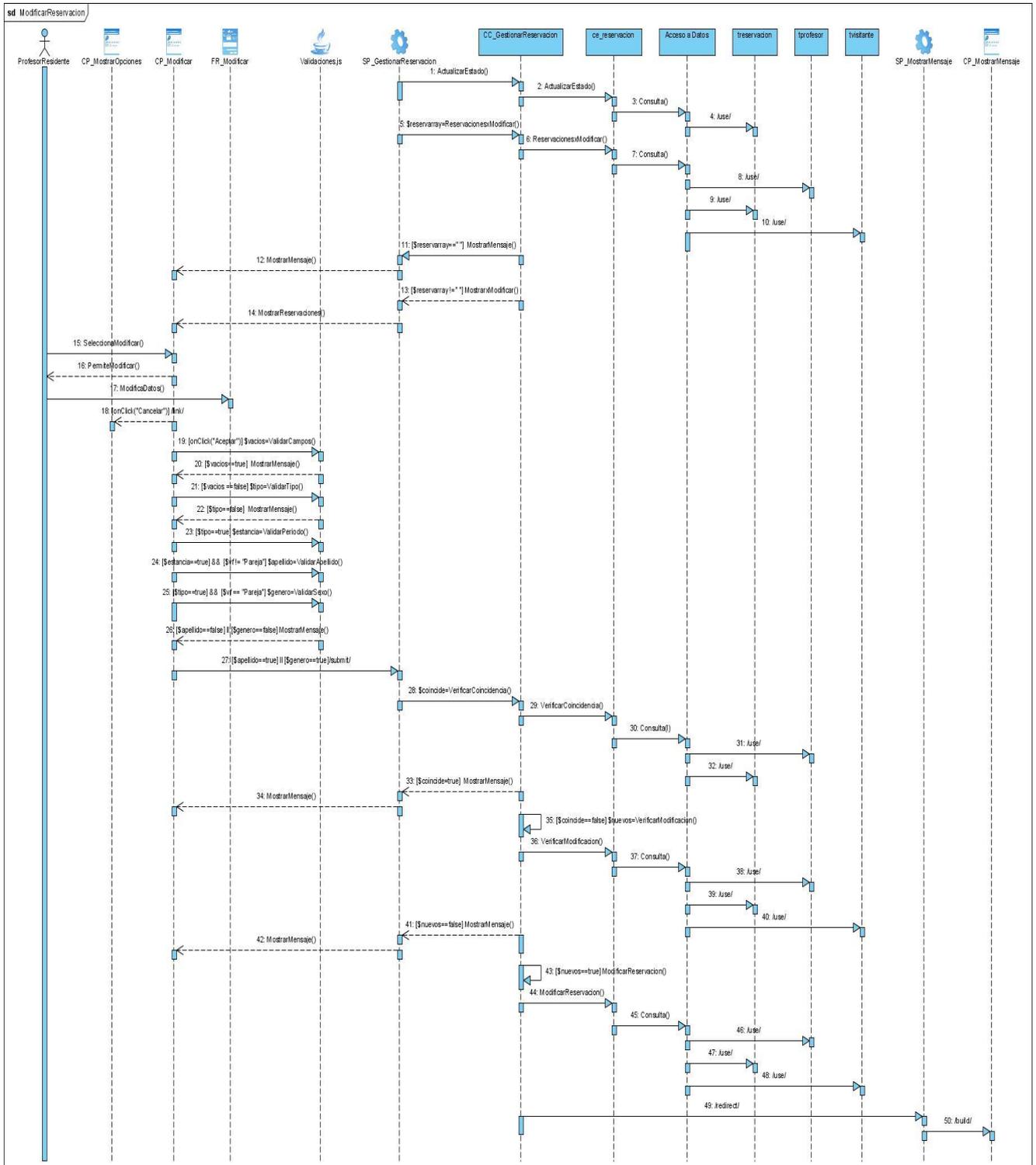


Figura 3.11 Diagrama de Secuencia Escenario "Modificar Reservación" CU Gestionar Reservación

Anexo4. Diagramas de Clases de Diseño Web

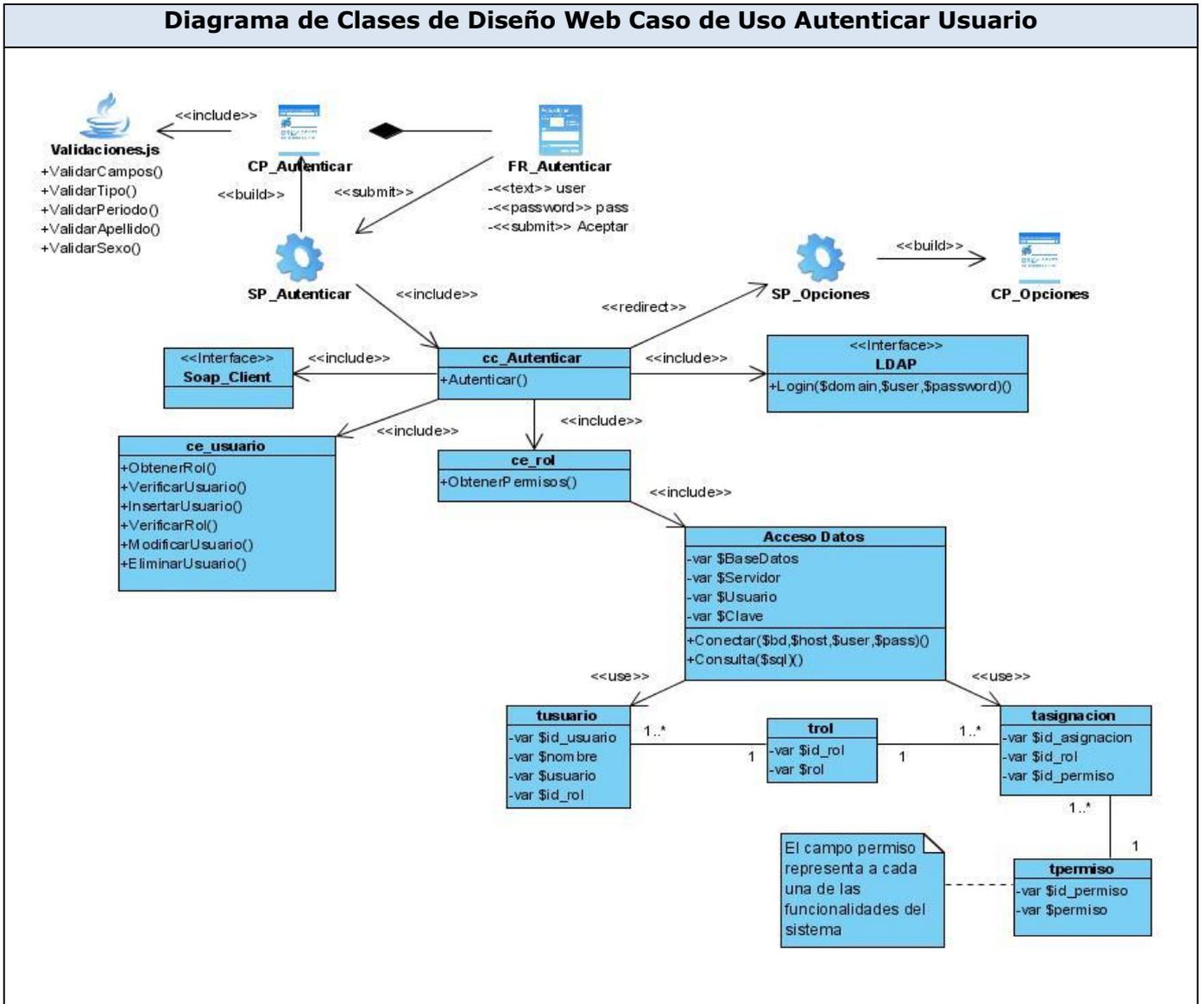


Figura 3.26 Diagrama de Clases de Diseño Web Caso de Uso Autenticar Usuario

Anexo5. Descripción de las clases del Diseño

Descripción de la clase CC\_GestionarReservacion

<b>Nombre:</b> CC_GestionarReservacion	
<b>Tipo de Clase:</b> Controladora	
<b>Para cada responsabilidad:</b>	
<b>Nombre:</b>	VerificarCoincidencia
<b>Descripción:</b>	Permite antes de reservar verificar que el período de fechas introducidas no coincidan en ningún momento con alguna otra reservación que tenga el profesor dado el usuario, la fecha de llegada y la fecha de salida
<b>Nombre:</b>	Reservar
<b>Descripción:</b>	Permite realizar una reservación registrando dados los datos del profesor, del familiar y de la propia reservación, dígase nombre del profesor, solapin, nombre del familiar, fecha de llegada, fecha de salida, entre otros datos.
<b>Nombre:</b>	ActualizarEstado
<b>Descripción:</b>	Permite actualizar el estado de las reservaciones, las que están “pendientes” pasan a “en curso” y las en curso pasan a ser efectivas dependiendo de la fecha que sea en el momento
<b>Nombre:</b>	ReservacionesxModificar
<b>Descripción:</b>	Permite Mostrar todas las reservaciones cuyo estado sea “pendiente” dado el usuario del profesor
<b>Nombre:</b>	VerificarModificacion
<b>Descripción:</b>	Permite verificar que los datos de la reservación hayan sido modificados dado el id de la reservación y los nuevos datos
<b>Nombre:</b>	ModificarReservacion
<b>Descripción:</b>	Permite modificar los datos de una reservación dado el id de la reservación y los nuevos datos
<b>Nombre:</b>	ReservacionesxCancelar
<b>Descripción:</b>	Permite Mostrar todas las reservaciones cuyo estado sea “pendiente” dado el usuario del profesor
<b>Nombre:</b>	CancelarReservacion
<b>Descripción:</b>	Permite Cancelar una Reservación dado el id de la reservación
<b>Nombre:</b>	AllReservaciones
<b>Descripción:</b>	Permite conocer todas las reservaciones de un profesor dado su usuario

Tabla 3.8 Descripción de la clase CC\_GestionarReservacion