



Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA DE SEGOVIA

Grado en ingeniería Informática de Servicios y Aplicaciones

Agencia de Viajes Extremos

Alumno: Miguel Extremo Nicolás
Tutor/a/es: Francisco José González Cabrera

RESUMEN

La siguiente documentación incorpora el proceso de obtención de una aplicación software cuyo objetivo es hacer posible y facilitar la gestión del trabajo referente a una agencia de viajes que no está abierta al público general, sino que sólo puede ser utilizada por el director y los empleados de la agencia.

El programa se ha desarrollado para que el director y los empleados gestionen los clientes que reservan hoteles, los hoteles que podrán ser reservados, las reservas como tales y los empleados que pueden utilizar la aplicación dentro de la agencia (esto último corresponde al director o administrador de la aplicación).

En la primera parte se muestra la especificación de requisitos del software en la que se obtienen las demandas del cliente, algo imprescindible para saber qué es lo que el programa ha de cumplir para responder a las expectativas del cliente. Después se elabora un proceso de análisis profundo seguido de la fase de diseño. Así nos encontramos con algo primordial para la elaboración del código software de manera clara y concisa.

Al realizar esta documentación se han elaborado un conjunto de diagramas basados en técnicas de orientación a objetos.

En la obtención de requisitos se sigue la metodología de Durán y Bernárdez, que nos muestra los requisitos a través de un conjunto de tablas y los diagramas de casos de uso que facilita el lenguaje unificado de modelado (UML). Las fases de análisis y diseño se han desarrollado a través del proceso unificado de modelado, a través del lenguaje UML.

Índice

1. Introducción.....	6
1.1. Participantes en el proyecto	6
1.2. Objetivos del sistema	6
1.3 Alcance.....	7
2. Estado del arte.....	9
2.1 Análisis DAFO.....	9
3. Costes del proyecto.....	11
3.1 Costes de personal.....	11
3.2 Costes software.....	12
3.3 Costes hardware.....	13
3.4 Método de puntos de función (Método Albrecht).....	13
4. Análisis del sistema.....	17
4.1. Requisitos de información	17
4.2. Requisitos funcionales.....	22
4.2.1. Diagramas de casos de uso	22
4.2.1.1 Diagrama del sistema.....	22
4.2.1.2 Diagrama de casos de uso de gestión de clientes.....	23
4.2.1.3 Diagrama de casos de uso de gestión de hoteles.....	23
4.2.1.4 Diagrama de casos de uso de gestión de reservas.....	24
4.2.1.5 Diagrama de casos de uso de gestión de empleados.....	24
4.3. Casos de uso del sistema.....	25
4.4. Actores del sistema	40
4.5. Requisitos no funcionales.....	41
5. Fase de análisis	42
5.1 Modelo de Análisis	43
5.2 Gestión de Clientes	43
5.2.1 Diagramas de secuencia de gestión de clientes	44
5.3 Gestión de Hoteles	45

5.3.1 Diagramas de secuencia de gestión de hoteles	46
5.4 Gestión de Reservas	47
5.4.1 Diagramas de secuencia de gestión de reservas	48
5.5 Gestión de Empleados.....	49
5.5.1 Diagramas de secuencia de gestión de empleados	50
5.6 Diagrama de Clases	51
5.5.1 Descripción del sistema actual	51
5.6.2 Diagrama de clases entidad.....	53
6. Fase de diseño.....	53
6.1. Modelo de Diseño	56
6.2. Modelo arquitectónico.....	56
6.3. Diagrama de componentes	57
6.4. Diagrama de despliegue	58
6.5. Diagrama del patrón Modelo-Vista-Controlador	59
7. Prueba de errores	60
7.1 Pruebas de caja blanca.....	60
7.2 Pruebas de caja negra.....	61
8. Documentación.....	62
8.1 Manual de usuario.....	62
9. AMPLIACIONES	65
Referencias.....	65

MEMORIA

1. Introducción

1.1. Participantes en el proyecto

Miguel Extremo Nicolás – Diseño, planificación, requisitos, programación, pruebas.

1.2. Objetivos del sistema

La finalidad primordial del sistema es la de elaborar una aplicación web que haga del proceso de gestión de los datos de una agencia de viajes concreta algo barato, rápido, fácil, eficiente y eficaz.

La aplicación será utilizada en primer lugar por el administrador, que se encargará de dar de alta, eliminar o modificar el conjunto de empleados de la agencia que tengan permiso para utilizar el programa.

Él y el resto de empleados podrán gestionar los clientes, los hoteles y las reservas que se hagan en la agencia.

Tabla 1. OBJ-1 Gestionar los clientes

OBJ-1	Gestionar los clientes
<i>Descripción</i>	El sistema deberá poder ser capaz de gestionar la información referente a los clientes: dar de alta, dar de baja, ser modificados o ser consultados.
<i>Importancia</i>	Alta
<i>Comentarios</i>	Ninguno

Fuente: Elaboración propia.

Tabla 2. OBJ-2 Gestionar los hoteles

OBJ-2	Gestionar los hoteles
<i>Descripción</i>	El sistema tendrá que posibilitar gestionar la información relativa a los hoteles: dar de alta, dar de baja, modificar o ser consultados.
<i>Importancia</i>	Alta
<i>Comentarios</i>	Ninguno

Fuente: Elaboración propia.

Tabla 3. OBJ-3 Gestionar las reservas

OBJ-3	Gestionar las reservas
<i>Descripción</i>	El sistema deberá ser capaz de gestionar la información relacionada con las reservas: dar de alta, dar de baja, modificar o ser consultadas.
<i>Importancia</i>	Alta
<i>Comentarios</i>	Ninguno

Fuente: Elaboración propia.

Tabla 4. OBJ-4 Gestionar los empleados

OBJ-4	Gestionar los empleados
<i>Descripción</i>	El sistema deberá hacer posible tramitar la información relativa a los empleados: dar de alta, dar de baja, modificar y ser consultados.
<i>Importancia</i>	Muy alta
<i>Comentarios</i>	Hay que diferenciar entre el administrador, que se encargará de gestionar esta cuestión, y el resto de empleados, que son registrados, borrados, modificados o consultados por el primero.

Fuente: Elaboración propia.

1.3 Alcance

El proyecto está pensado para que puedan utilizarlo todos los trabajadores de la agencia de viajes Extremos, desde los empleados, hasta el administrador.

Es una propuesta, que en el caso de dar los resultados esperados y tener una buena aceptación por el cliente final de la aplicación podría convertirse en la base para el desarrollo de una aplicación de gestión interna de reservas a nivel provincial y nacional, abarcando así mayor número de hoteles disponibles para su gestión de reservas.

Se desarrollará una aplicación de gestión del clientes, hoteles, reservas y empleados para que los trabajadores de la agencia puedan realizar:

- Gestión de clientes: incluye las características y funcionalidades asociadas a la creación, modificación y visualización de los datos de los clientes.

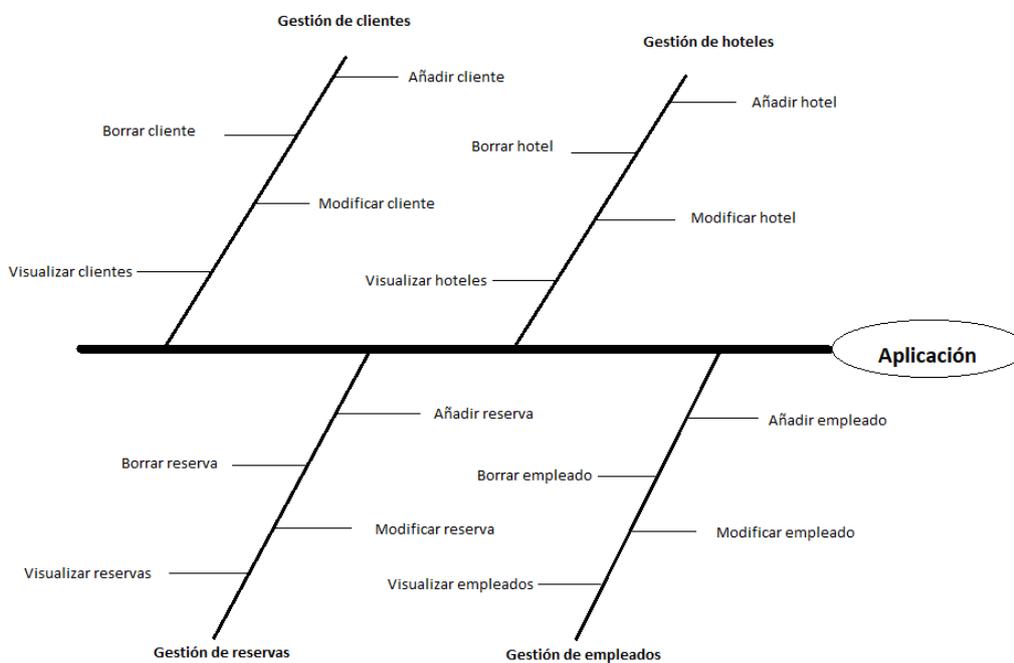
-Gestión de hoteles: incluye las características y funcionalidades asociadas a la creación, modificación y visualización de los hoteles.

-Gestión de reservas: incluye las características y funcionalidades asociadas a la creación, modificación y visualización de las reservas.

-Gestión de empleados: incluye las características y funcionalidades asociadas a la creación, modificación y visualización de los empleados para los administradores.

A continuación, se presenta el árbol de características de la aplicación:

Figura 1 - Árbol de características de la aplicación



2. Estado del arte

2.1 Análisis DAFO

El análisis DAFO es una técnica que nos permite poner la situación actual de un negocio, y que facilita tomar las decisiones estratégicas adecuadas. Esta técnica se suele dar en estudios de mercado, o para establecer planes de negocio por lo que, para la realización de este proyecto, nos es muy útil para secciones que se encuentran más adelante.

Las letras que componen DAFO significan Debilidades, Amenazas, Fortalezas y Oportunidades, y orientadas a este sector del mercado se analizarán en base al perfil de un alojamiento.

■ Debilidades

Forman las características que limitan la capacidad de desarrollo de tu negocio. Dados para el entorno en el que nos encontramos, surgen los siguientes puntos:

- Nos dirigimos a un mercado maduro en el cual hay aplicaciones de gestión con más de 5 años de experiencia.
- Los metabuscadores de reserva están más cerrados a integrarse con nuevas plataformas.
- Es un mercado muy atomizado, que requiere un gran esfuerzo para dar a conocer el producto.

■ Amenazas

Son los factores externos que pueden impedir la ejecución de tu estrategia o poner en peligro la viabilidad de tu negocio. Dados para el entorno en el que nos encontramos, surgen los siguientes puntos:

- El mercado de las aplicaciones SaaS (Software as a Service) cada vez está más maduro y ofrecen en experiencias tanto en PC como móvil.
- Potencial de irrupción de empresas como Amazon y Google que den un servicio integrado de metabuscador más software de gestión.

■ Fortalezas

Forman las características que limitan la capacidad de desarrollo de tu negocio. Dados para el entorno en el que nos encontramos, surgen los siguientes puntos:

- Conocimiento del sector turístico con más de 10 años de experiencia en el sector.
- Motivación personal.
- No requiere una inversión inicial fuerte en sistemas de gestión.
- Es un modelo escalable que permitirá integrar todas las experiencias de los usuarios en una única plataforma.

■ Oportunidades

Reúnen el conjunto de recursos internos, posiciones de poder y cualquier tipo de ventaja competitiva propia del negocio. Dados para el entorno en el que nos encontramos, surgen los siguientes puntos:

- Existe un fuerte nicho de mercado en apartamentos y hoteles rurales que no disponen de sistemas de gestión.
- Posibilidad de acceder a las salidas de financiación de la administración pública para desarrollar la fase II.
- Posibilidad de ir de la mano de los portales turísticos de las comunidades autónomas para dar a conocer el servicio.
- Dado que el modelo de negocio se basa en un software gratuito, será una alternativa real a aquellos servicios de pago que están operando en el mercado.

3. Costes del proyecto

3.1 Costes de personal

Estos gastos están relacionados con todos los recursos humanos que participan en las diferentes etapas de planificación inicial.

Para este proyecto, solo hay un recurso humano que participa activamente en todas las fases del proyecto: el autor. Este autor desempeña diferentes roles según la fase y la tarea que esté realizando para proporcionar una estimación más precisa y realista de este tipo de gastos. Los roles asumidos en el proyecto son los siguientes:

- Director de Proyecto: se encarga de gestionar tanto al equipo de trabajo y sus tareas desempeñadas como el control del cumplimiento de objetivos y características del proyecto. También es el encargado de realizar las tareas de revisión y redacción de la documentación.

- Analista: se encarga de realizar las tareas de análisis y exploración de requisitos que debe cumplir el proyecto. También se encarga de gestionar la arquitectura del sistema.

- Diseñador de UX/ IU: se encarga de analizar y diseñar todos los aspectos relacionados con la experiencia e interfaces de usuario.

- Desarrollador: se encarga de llevar a cabo las labores de implementación del proyecto.

- Tester: se encarga de comprobar que se cumplen todos los objetivos y requisitos establecidos para el producto (tanto el producto completo como los distintos productos que se van generando a lo largo de las fases del proyecto).

Tabla 5. - Tabla de costes de personal

Rol	Tiempo (h)	Coste/h	Coste real
Gestor de proyecto	150	20€/h	3.000,00 €
Analista	75	15€/h	1.125,00 €
Desarrollador/ Tester/Diseñador	200	12,50€/h	2.500,00 €
Total			6.625,00 €

3.2 Costes software

Se refieren a los gastos generados por todos los programas y herramientas informáticas empleados para llevar a cabo el proyecto. En el siguiente cuadro se muestra el desenlace de esta evaluación:

Tabla 6. - Tabla de costes de software

Componente	Precio	Vida útil media	% Uso	Coste real
Windows 10 home	120,00 €	4 años	25,00%	30,00 €
Microsoft 365	70,00 €	1 año	100%	70,00 €
Eclipse IDE	0,00 €	-	-	0,00 €
Google chrome	0,00 €	-	-	0,00 €
StarUml	0,00 €	-	-	0,00 €
Total				100,00 €

3.3 Costes hardware

Se refieren a los gastos generados por todos los componentes empleados para llevar a cabo el proyecto. En el siguiente cuadro se muestra el desenlace de esta evaluación:

Tabla 7. - Tabla de costes de hardware

Componente	Precio	Vida útil media	% Uso	Coste real
Ordenador	1400,00 €	4 años	25,00%	350,00 €
Monitor	250,00 €	3 años	33,33%	83,33 €
Ratón y teclado	180,00 €	2 años	50,00%	90,00 €
Conexión a internet	50,00 €	1 mes	1200%	600,00 €
Total				1123,33 €

3.4 Método de puntos de función (Método Albrecht)

En esta sección se realizará el análisis de los costes mediante puntos de función, que evalúa el coste del trabajo de cada una de las funciones que tendrá la aplicación. De esta manera y con todos los puntos de función, podremos estimar el coste del desarrollo. En cuanto a las funciones de la aplicación, se pueden clasificar en diferentes grupos:

- Entradas de usuario: Compone los datos que el usuario introduce en el sistema.
- Salidas de usuario: Compone los datos que la aplicación muestra al usuario.
- Consultas externas: Compone las peticiones que se procesan en un sistema externo al de la aplicación y que interactúa con ella.
- Ficheros lógicos internos: Compone los archivos que la aplicación utiliza en su base de datos.

- Ficheros lógicos externos: Compone los archivos que la aplicación almacena en otro lugar que no es el dispositivo en el que se usa la aplicación.

Una vez establecidos los campos de información, es necesario determinar la complejidad de cada una de las funcionalidades del sistema. Para ello, se utilizan las tablas siguientes:

Entradas de datos

Tabla 8. - Tabla de entrada de datos

Descripción	Complejidad
Acceso a la aplicación	Baja
Formulario de registro de cliente	Baja
Formulario de registro de hotel	Baja
Formulario de registro de reserva	Media
Formulario de registro de empleado	Baja
Modificación de clientes	Media
Modificación de hoteles	Media
Modificación de reservas	Media
Modificación de empleado	Media

Salidas de datos

Tabla 9. - Tabla de salida de datos

Descripción	Complejidad
Listado de clientes	Media
Listado de hoteles	Media
Listado de reservas	Media
Listado de empleados	Media

Ficheros lógicos internos de datos

Tabla 10. - Tabla de ficheros lógicos internos de datos

Descripción	Complejidad
Clientes	Baja
Hoteles	Baja
Reservas	Baja
Empleados	Baja

Una vez establecidos los puntos de función sin ajustar (PFNA), se suman teniendo en cuenta su ponderación o importancia dependiendo del grupo al que pertenece cada funcionalidad y su complejidad. A continuación, se muestran los puntos de función sin ajustar calculados por cada grupo:

	Peso complejidad baja	Peso complejidad media	Peso complejidad alta
Entradas	$3 \times 4 = 12$	$4 \times 5 = 20$	$6 \times 0 = 0$
Salidas	$3 \times 0 = 0$	$4 \times 4 = 16$	$6 \times 0 = 0$
Consultas externas	$4 \times 0 = 0$	$5 \times 0 = 0$	$7 \times 0 = 0$
Ficheros internos	$7 \times 4 = 28$	$10 \times 0 = 0$	$15 \times 0 = 0$
Ficheros externos	$5 \times 0 = 0$	$7 \times 0 = 0$	$10 \times 0 = 0$
Total = 32 + 16 + 28 = 76 PFNA			

Ya definidos los puntos de función sin ajustar (PFNA) es momento de asignar el grado de complejidad de los 14 factores de ajuste que caracterizan la complejidad y la funcionalidad del sistema. Para ello se cuenta con una escala que va desde el 0 al 5, puntuando los factores de complejidad descritos en la siguiente tabla:

Factor de ajuste	Complejidad
Comunicación de datos	3
Funciones distribuidas	2
Prestaciones	2
Gran uso de la configuración	1
Velocidad de las transacciones	4
Entrada online de datos	0
Diseño para la eficiencia del usuario final	3
Actualización de datos online	0

Complejidad de procesos lógicos de la aplicación	3
Reusabilidad de código	3
Fácil instalación	4
Facilidad de operación	3
Localizaciones múltiples	0
Facilidad de cambios	4
Total	32

Para calcular el factor de ajuste (FA) se aplica la siguiente fórmula basada en la suma total de los factores de complejidad (FC) obtenidos previamente:

$$FA = 0,65 + (0,01 * FC) = 0,65 + (0,01 * 32) = \mathbf{0,97}$$

Una vez obtenido el factor de ajuste, se calculan los puntos de función ajustados (PFA):

$$PFA = PFNA * FA = 76 * 0,97 = \mathbf{73,72}$$

Para finalizar la estimación es necesario calcular tanto la duración del proyecto como el esfuerzo personal en horas. Para ello es necesario establecer la equivalencia puntos de horas por puntos de función contando con que 1 mes de esfuerzo (16 días laborables) equivale a 9 puntos de función.

$$\mathbf{Duración de proyecto estimada = 73,72 PFA / 9 = 8,19 meses}$$

4. Análisis del sistema

4.1. Requisitos de información

Tabla 11. IRQ-1 Información sobre clientes

IRQ-1	Información sobre clientes
Objetivos asociados	OBJ-1 Gestionar los clientes OBJ-4 Gestionar los empleados
Requisitos asociados	
Descripción	El sistema deberá almacenar los datos referentes a los clientes que haya en la agencia.
Datos específicos	Código de cliente, nombre, apellidos, DNI.
Importancia	Media
Comentarios	

Tabla 12. IRQ-2 Información sobre hoteles

IRQ-2	Información sobre hoteles
Objetivos asociados	OBJ-1 Gestionar los hoteles OBJ-4 Gestionar los empleados
Requisitos asociados	IRQ1- Información sobre hoteles
Descripción	El sistema deberá almacenar los datos de los hoteles que estén registrados en la agencia.
Datos específicos	Código de hotel, nombre, categoría y localización.
Importancia	Media
Comentarios	

Fuente: Elaboración propia.

Tabla 13. IRQ-3 Información sobre reservas

IRQ-3	Información sobre reservas
Objetivos asociados	OBJ-3 Gestionar las reservas OBJ-4 Gestionar los empleados
Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de los clientes y los hoteles que reservan esos clientes para confeccionar las reservas que se registran en la agencia.
Datos específicos	Código de reserva, código de cliente, código de hotel, fecha de inicio, fecha de fin y precio.
Importancia	Media
Comentarios	

Fuente: Elaboración propia.

Tabla 14. IRQ-4 Información sobre empleados

IRQ-4	Información sobre empleados
Objetivos asociados	OBJ-4 Gestionar los empleados
Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de los empleados que son dados de alta en el programa por medio del administrador.
Datos específicos	Código de empleado, nombre de usuario (nick), password y nombre.
Importancia	Muy alta

Comentarios	
--------------------	--

Fuente: Elaboración propia.

Tabla 15. IRQ-5 Información sobre sesiones

IRQ-5	Información sobre sesiones
Objetivos asociados	OBJ-4 Gestionar los empleados
Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de los usuarios de la aplicación para que al intentar acceder al sistema se les cree una nueva sesión si introducen los datos de acceso (nick y contraseña) correctamente.
Datos específicos	Código de empleado, nombre de usuario (nick), password y nombre.
Importancia	Muy alta
Comentarios	

Fuente: Elaboración propia.

Tabla 16. CRQ-1 Relación entre clientes, hoteles y reservas

CRQ-1	Relación entre clientes, hoteles y reservas
Objetivos asociados	OBJ-3 Gestionar las reservas
Requisitos asociados	IRQ-1 Información sobre los clientes IRQ-2 Información sobre los hoteles IRQ-3 Información sobre las reservas

Descripción	
Importancia	Media
Comentarios	En las reservas se tiene en cuenta el cliente que posee la reserva y el hotel que está reservando ese cliente.

Fuente: Elaboración propia.

Tabla 17. CRQ-2 Relación entre clientes, hoteles, reservas y empleados

CRQ-2	Relación entre clientes, hoteles, reservas y empleados
Objetivos asociados	OBJ-3 Gestionar las reservas OBJ-4 Gestionar los empleados
Requisitos asociados	IRQ-1 Información sobre los clientes IRQ-2 Información sobre los hoteles IRQ-3 Información sobre las reservas IRQ-4 Información sobre los empleados
Descripción	Los empleados se encargan de gestionar toda la información relativa a las reservas: datos de la reserva, así como los de los clientes que reservan y los de los hoteles que son reservados.
Importancia	Alta
Comentarios	Esta relación representa el préstamo del ejemplar de un libro concedido a un socio.

Fuente: Elaboración propia.

Tabla 18. CRQ-3 Relación entre empleados y administrador

CRQ-3	Relación entre empleados y administrador
--------------	---

Objetivos asociados	OBJ-4 Gestionar los empleados
Requisitos asociados	IRQ-5 Información sobre los empleados IRQ-6 Información sobre el administrador
Descripción	El administrador posee autoridad para dar de alta a los empleados, darlos de baja, modificarlos o visualizar sus datos, privilegio que éstos no tienen.
Importancia	Muy alta
Comentarios	Un empleado no puede darse de alta en la aplicación por sí mismo, pero sí podrá registrarse cada vez que vaya a utilizar el programa una vez que haya sido dado de alta por el administrador.

Fuente: Elaboración propia.

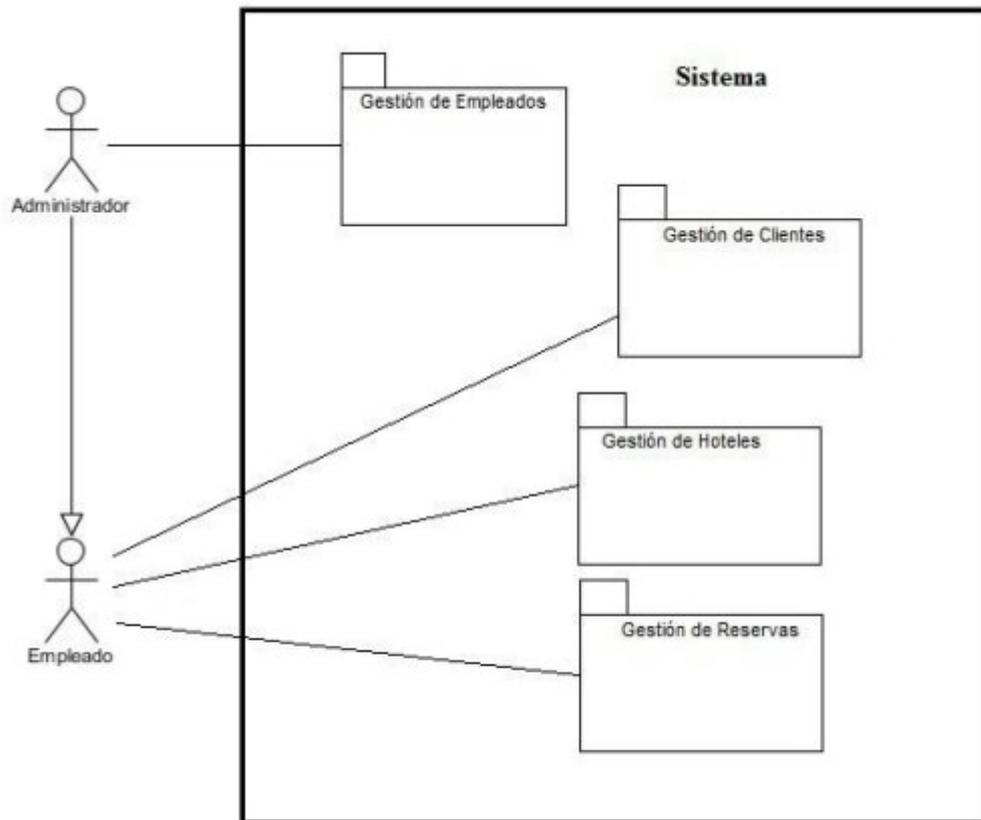
4.2. Requisitos funcionales

4.2.1. Diagramas de casos de uso

4.2.1.1 Diagrama del sistema

Vamos a representar el diagrama de sistema, que nos hará posible representar la relación que existe entre los actores que interaccionan dentro de él y las distintas partes de éste. Luego dividiremos cada paquete en los casos de uso que lo forman.

Figura 2. Diagrama del sistema

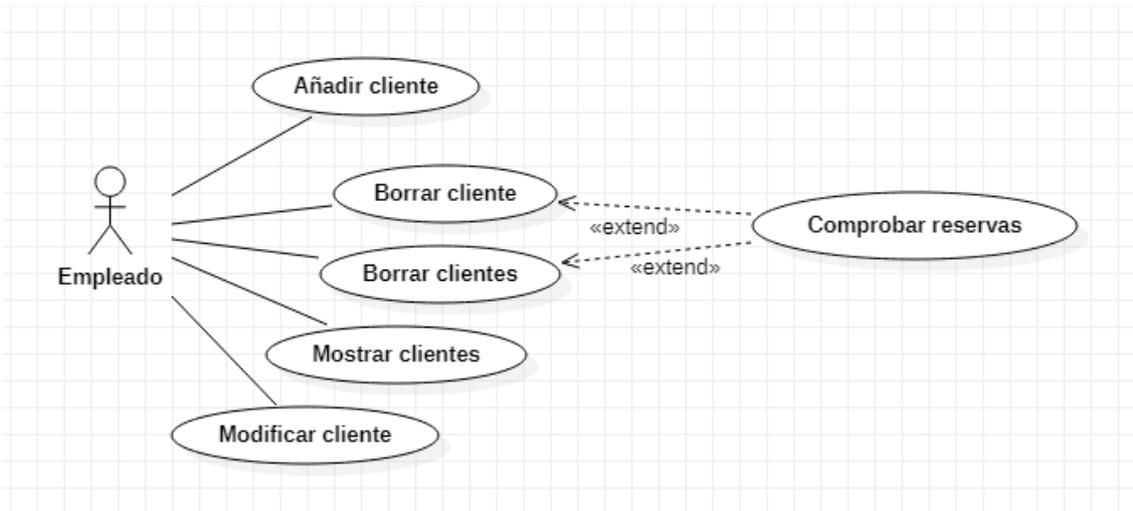


Fuente: Elaboración propia

4.2.1.2 Diagrama de casos de uso de gestión de clientes

Este diagrama representa la funcionalidad relativa a la información necesaria para trabajar con los datos de los clientes que son dados de alta en la aplicación.

Figura 3. Diagrama de casos de uso de Gestión de Clientes

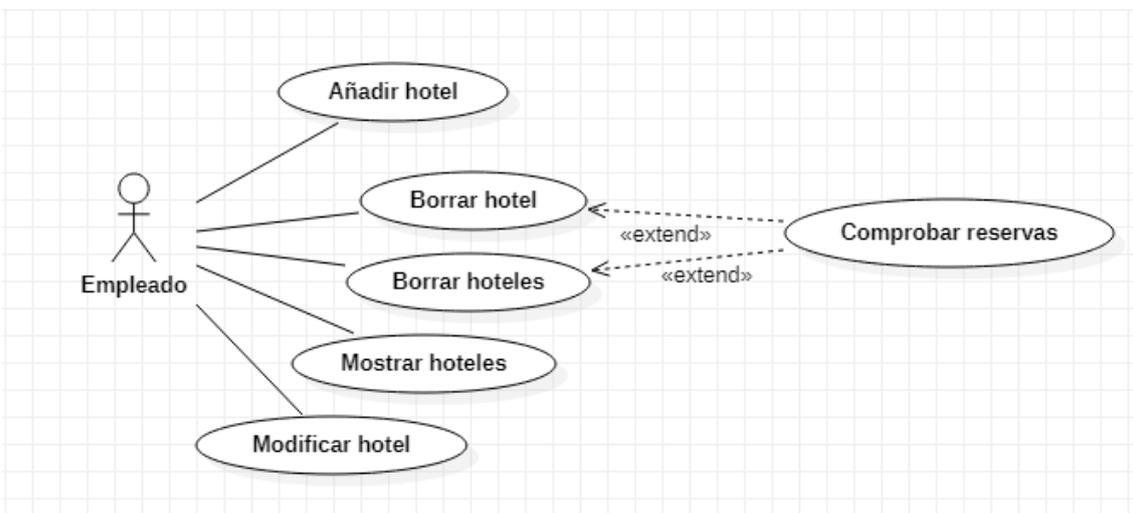


Fuente: Elaboración propia

4.2.1.3 Diagrama de casos de uso de gestión de hoteles

Esta parte es la encargada de gestionar todas actividades relacionadas con los hoteles de la aplicación.

Figura 4. Diagrama de casos de uso de Gestión de Hoteles

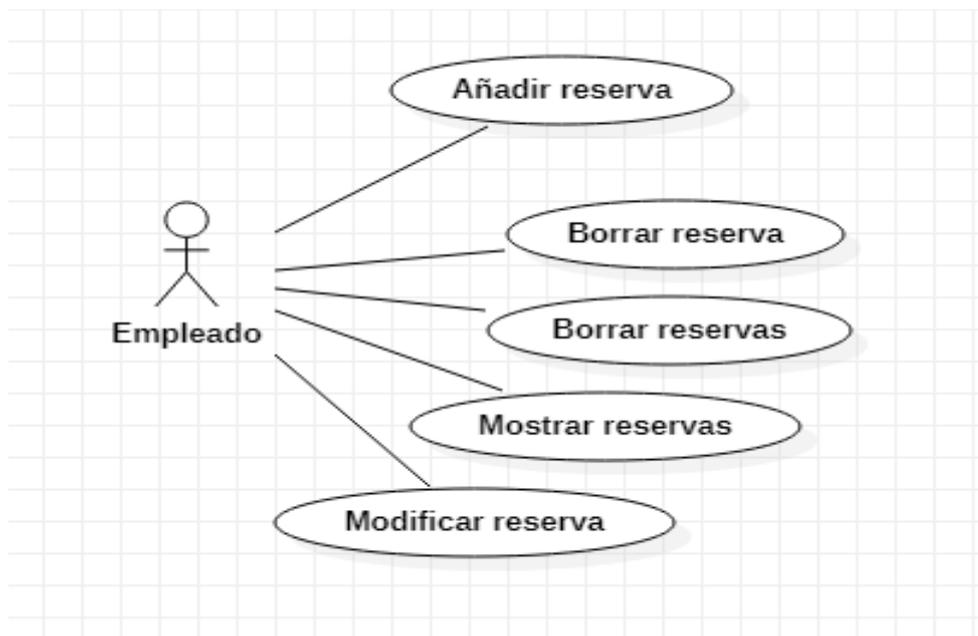


Fuente: Elaboración propia

4.2.1.4 Diagrama de casos de uso de gestión de reservas

Este diagrama muestra las acciones posibles a realizar sobre la información relativa a las reservas que se realicen en la aplicación para los clientes de los hoteles.

Figura 5. Diagrama de casos de uso de Gestión de Reservas

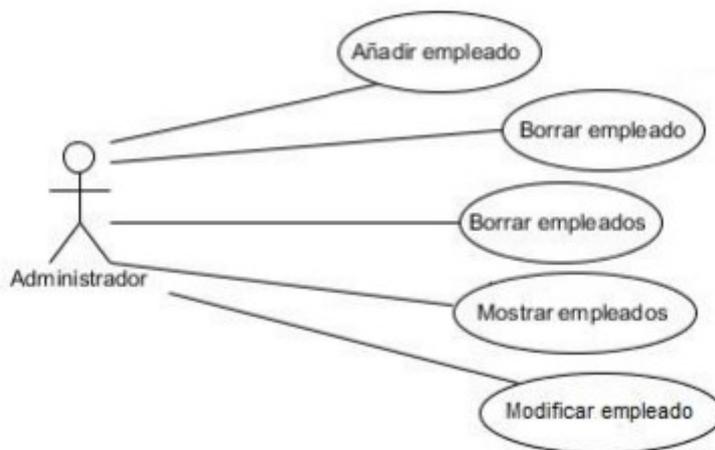


Fuente: Elaboración propia

4.2.1.5 Diagrama de casos de uso de gestión de empleados

Este apartado de gestión le otorga al administrador la función de dar de añadir, dar de baja o consultar los datos de los empleados de la biblioteca que van a trabajar con la aplicación.

Figura 6. Diagrama de casos de uso de Gestión de Empleados



Fuente: Elaboración propia

4.3. Casos de uso del sistema

Tabla 19. UC-01 Añadir cliente

UC-01	Añadir cliente	
Objetivos asociados	OBJ-1 Gestionar los clientes	
Requisitos asociados	IRQ-1 Información sobre los clientes	
Descripción	Este caso de uso nos permite dar de alta un cliente en el programa.	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita añadir un cliente
	2	El programa recibe los datos del cliente para integrarlo en la base de datos
	3	Se añade el cliente
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 20. UC-02 Borrar cliente

UC-02	Borrar cliente
Objetivos asociados	OBJ-1 Gestionar los clientes

Requisitos asociados	IRQ-1 Información sobre los clientes IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite borrar un cliente del programa por medio de su código de cliente.	
Precondición	El cliente no ha de tener reservas asociadas.	
Secuencia normal	Paso	Acción
	1	El empleado solicita borrar un cliente
	2	El sistema recoge los datos del cliente
	3	Se borra el cliente con éxito
Postcondición		
Excepciones	2	El cliente tiene reservas asociadas. Mostrar error.
Comentarios		

Fuente: Elaboración propia.

Tabla 21. UC-03 Borrar clientes

UC-03	Borrar clientes	
Objetivos asociados	OBJ-1 Gestionar los clientes	
Requisitos asociados	IRQ-1 Información sobre los clientes IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite dar de baja todos los clientes que están registrados en la base de datos.	
Precondición	Los clientes no deben tener ninguna reserva asociada	
Secuencia normal	Paso	Acción

	1	El empleado solicita borrar clientes
	2	El sistema busca en la base de datos si algún cliente tiene reservas asociadas

	3	El sistema borra todos los clientes
Postcondición		
Excepciones	2	Uno o más clientes posee/n reserva/s asociada/s. Mostrar error.
Comentarios		

Fuente: Elaboración propia.

Tabla 22. UC-04 Mostrar clientes

UC-04	Mostrar clientes	
Objetivos asociados	OBJ-1 Gestionar los clientes	
Requisitos asociados	IRQ-1 Información sobre los clientes	
Descripción	Este caso de uso permite consultar todos los clientes de la aplicación.	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita mostrar todos los clientes
	2	El sistema muestra todos los clientes
Postcondición		

<i>Excepciones</i>	
<i>Comentarios</i>	

Fuente: Elaboración propia.

Tabla 23. UC-05 Modificar cliente

UC-05	Modificar cliente	
<i>Objetivos asociados</i>	OBJ-1 Gestionar los clientes	
<i>Requisitos asociados</i>	IRQ-1 Información sobre los clientes	
<i>Descripción</i>	Este caso de uso permite modificar un cliente de la aplicación.	
<i>Precondición</i>		
<i>Secuencia normal</i>	Paso	Acción
	1	El empleado solicita modificar un cliente
	2	El sistema da la opción de modificar cualquiera de los datos del cliente
	3	El empleado modifica los datos del cliente que desee
<i>Postcondición</i>		
<i>Excepciones</i>		
<i>Comentarios</i>		

Fuente: Elaboración propia.

Tabla 24. UC-06 Añadir hotel

UC-06	Añadir hotel
Objetivos asociados	OBJ-2 Gestionar los hoteles
Requisitos asociados	IRQ-1 Información sobre los hoteles
Descripción	Este caso de uso permite dar de alta un hotel.
Precondición	

Secuencia normal	Paso	Acción
	1	El empleado solicita dar de alta un hotel
	2	El sistema recibe los datos del hotel a añadir
	3	Se añade el hotel
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 25. UC-07 Borrar hotel

UC-07	Borrar hotel
Objetivos asociados	OBJ-2 Gestionar los hoteles
Requisitos asociados	IRQ-2 Información sobre los hoteles

	IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite dar de baja un hotel teniendo en cuenta si tiene o no reservas asociadas.	
Precondición	El hotel que se desea borrar no ha de tener reservas asociadas.	
Secuencia normal	Paso	Acción
	1	El empleado solicita borrar un ejemplar
	2	El sistema recibe los datos del ejemplar y comprueba si tiene reservas asociadas
	3	Se borra el ejemplar
Postcondición		
Excepciones	2	El hotel tiene reservas asociadas. Mostrar error.
Comentarios		

Fuente: Elaboración propia.

Tabla 26. UC-08 Borrar hoteles

UC-08	Borrar hoteles
Objetivos asociados	OBJ-2 Gestionar los hoteles
Requisitos asociados	IRQ-2 Información sobre los hoteles IRQ-3 Información sobre las reservas
Descripción	Este caso de uso permite eliminar todos los hoteles de la aplicación.
Precondición	Ningún hotel debe tener alguna reserva asociada

Secuencia normal	Paso	Acción
	1	El empleado solicita borrar los hoteles de la aplicación
	2	El sistema comprueba si algún hotel posee alguna reserva asociada
	3	Se borran todos los hoteles de la aplicación
Postcondición		
Excepciones	2	Algún hotel posee una o más reservas asociadas. Mostrar error.
Comentarios		

Fuente: Elaboración propia.

Tabla 27. UC-09 Mostrar hoteles

UC-09	Mostrar hoteles
Objetivos asociados	OBJ-2 Gestionar los hoteles

Requisitos asociados	IRQ-2 Información sobre los hoteles	
Descripción	Este apartado permite mostrar todos los hoteles del programa.	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita mostrar los hoteles del programa
	2	El sistema muestra por pantalla todos los hoteles
Postcondición		

<i>Excepciones</i>	
<i>Comentarios</i>	

Fuente: Elaboración propia.

Tabla 28. UC-10 Modificar hotel

UC-10	Modificar hotel	
<i>Objetivos asociados</i>	OBJ-2 Gestionar los hoteles	
<i>Requisitos asociados</i>	IRQ-2 Información sobre los hoteles	
<i>Descripción</i>	Este caso de uso permite modificar un hotel cualquiera.	
<i>Precondición</i>		
<i>Secuencia normal</i>	Paso	Acción
	1	El empleado solicita modificar un hotel
	2	El sistema busca en la base de datos la información del hotel a modificar y le da la opción al empleado de modificar cualquier dato relativo al hotel en cuestión
	3	Se modifica el hotel

<i>Postcondición</i>	
<i>Excepciones</i>	
<i>Comentarios</i>	

Fuente: Elaboración propia.

Tabla 29. UC-11 Añadir reserva

UC-11	Añadir reserva	
Objetivos asociados	OBJ-3 Gestionar las reservas	
Requisitos asociados	IRQ-1 Información sobre los clientes IRQ-2 Información sobre los hoteles IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite dar de alta una reserva en el programa.	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita añadir una reserva
	2	El sistema recibe los datos de la reserva, así como los códigos del cliente para el que se realiza la reserva y el del hotel que se va a reservar
	3	Se crea la reserva
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 30. UC-12Borrarreserva

UC-12	Borrar reserva
Objetivos asociados	OBJ-3 Gestionar las reservas

Requisitos asociados	IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite dar de baja una reserva existente en la aplicación.	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita borrar una reserva
	2	El sistema busca en la base de datos la información referente a dicha reserva
	3	Se borra la reserva
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 31. UC-13 Borrar reservas

UC-13	Borrar reservas
Objetivos asociados	OBJ-3 Gestionar las reservas
Requisitos asociados	IRQ-3 Información sobre las reservas
Descripción	Este caso de uso permite dar de baja todas las reservas del programa
Precondición	

<i>Secuencia normal</i>	Paso	Acción
	1	El empleado solicita borrar todas las reservas
	2	Se borran todas las reservas
<i>Postcondición</i>		
<i>Excepciones</i>		
<i>Comentarios</i>		

Fuente: Elaboración propia.

Tabla 32. UC-14 Mostrar reservas

UC-14	Mostrar reservas	
<i>Objetivos asociados</i>	OBJ-3 Gestionar las reservas	
<i>Requisitos asociados</i>	IRQ-3 Información sobre las reservas	
<i>Descripción</i>	Este caso de uso permite consultar todas las reservas de la aplicación	
<i>Precondición</i>		
<i>Secuencia normal</i>	Paso	Acción
	1	El empleado solicita mostrar todas las reservas
	2	Se muestran todas las reservas
<i>Postcondición</i>		
<i>Excepciones</i>		
<i>Comentarios</i>		

Fuente:Elaboración propia.

Tabla 33. UC-15 Modificar reserva

UC-15	Modificar reserva	
Objetivos asociados	OBJ-3 Gestionar las reservas	
Requisitos asociados	IRQ-3 Información sobre las reservas	
Descripción	Este caso de uso permite modificar una reserva concreta existente en el programa	
Precondición		
Secuencia normal	Paso	Acción
	1	El empleado solicita modificar una reserva
	2	El sistema le da la opción al empleado de modificar cualquier dato referente a la reserva en cuestión, siempre y cuando el código del cliente y el del hotel para los que se realiza la reserva existan en la base de datos
	3	Se modifica la reserva
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 34. UC-16 Añadir empleado

UC-16	Añadir empleado
--------------	------------------------

Objetivos asociados	OBJ-4 Gestionar los empleados
----------------------------	-------------------------------

Requisitos asociados	IRQ-4 Información sobre los empleados	
Descripción	Este caso de uso le permite al administrador dar de alta un empleado en la aplicación	
Precondición	Hay que tener permiso de administrador	
Secuencia normal	Paso	Acción
	1	El administrador solicita dar de alta un empleado
	2	El sistema recibe la información del empleado
	3	Se añade el empleado
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 35. UC-17 Borrar empleado

UC-17	Borrar empleado
Objetivos asociados	OBJ-4 Gestionar los empleados
Requisitos asociados	IRQ-4 Información sobre los empleados
Descripción	Este caso de uso permite dar de baja un empleado existente en el programa

Precondición	Hay que tener permiso de administrador	
Secuencia normal	Paso	Acción
	1	El administrador solicita borrar un empleado
	2	El sistema busca en la base de datos la información relativa al empleado a borrar
	3	Se borra el empleado

Postcondición	
Excepciones	
Comentarios	

Fuente: Elaboración propia.

Tabla 36. UC-18 Borrar empleados

UC-18	Borrar empleados	
Objetivos asociados	OBJ-4 Gestionar los empleados	
Requisitos asociados	IRQ-4 Información sobre empleados	
Descripción	Este caso de uso permite borrar todos los empleados de la aplicación	
Precondición	Hay que tener permiso de administrador	
Secuencia normal	Paso	Acción
	1	El administrador solicita borrar todos los empleados

	2	Se borran todos los empleados de la aplicación
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 37. UC-19 Mostrar empleados

UC-19	Mostrar empleados	
Objetivos asociados	OBJ-4 Gestionar los empleados	
Requisitos asociados	IRQ-4 Información sobre empleados	
Descripción	Este caso de uso permite mostrar todos los empleados de la aplicación	
Precondición	Hay que tener permiso de administrador	
Secuencia normal	Paso	Acción
	1	El administrador solicita mostrar los datos de todos los empleados
	2	Se muestran los datos de todos los empleados
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

Tabla 38. UC-20 Modificar empleado

UC-20	Modificar empleado	
Objetivos asociados	OBJ-4 Gestionar los empleados	
Requisitos asociados	IRQ-4 Información sobre empleados	
Descripción	Este caso de uso permite modificar un empleado de la aplicación	
Precondición	Hay que tener permiso de administrador	
Secuencia normal	Paso	Acción
	1	El administrador solicita modificar un empleado del programa
	2	El sistema le da la posibilidad al administrador de modificar cualquier dato del empleado en cuestión

	3	Se modifica el empleado
Postcondición		
Excepciones		
Comentarios		

Fuente: Elaboración propia.

4.4. Actores del sistema

Tabla 39. ACT-1 Administrador

ACT-1	Administrador
--------------	----------------------

Descripción	El administrador es un empleado con el privilegio especial de acceder al menú de gestión de empleados, en el que es posible dar de alta, dar de baja un empleado o todos, modificar un empleado o mostrar la lista de empleados registrados en el sistema.
Comentarios	Posee un nombre de usuario y una contraseña diferente al del resto de empleados.

Fuente: Elaboración propia.

Tabla 40. ACT-2 Empleado

ACT-2	Empleado
Descripción	El empleado puede acceder al menú de gestión de clientes, de hoteles o de reservas, pudiendo dentro de cualquiera de éstos dar de alta, dar de baja uno o varios de los elementos mencionados, mostrar la lista de elementos registrados en la aplicación o modificar uno de ellos.
Comentarios	

Fuente: Elaboración propia.

4.5. Requisitos no funcionales

Tabla 41. Tabla de requisitos no funcionales

ID	Categoría	Descripción
RNF-01	Usabilidad	El tiempo requerido de aprendizaje por parte de un usuario del funcionamiento del sistema (una vez revisado el manual de usuario) no será superior a las 2 horas
RNF-02	Usabilidad	La tasa de errores cometidos por el usuario final debe de ser inferior al 1% del total
RNF-03	Usabilidad	La aplicación debe caracterizarse por contar con un diseño Responsive para su correcta visualización y ejecución en ordenadores y

		tablets
RNF-04	Usabilidad	El sistema debe presentar al usuario final mensajes de éxito y error
RNF-05	Eficiencia	El sistema debe soportar una tasa de sesiones activas concurrentes de 100
RNF-06	Eficiencia	Toda respuesta a cualquier transacción realizada por el usuario final o por el sistema no debe exceder un tiempo de respuesta de 20 segundos
RNF-07	Eficiencia	La probabilidad de fallo de la aplicación debe ser inferior al 1%
RNF-08	Seguridad	Las transacciones realizadas con servidor son siempre cifradas y realizadas bajo el protocolo seguro HTTPS
RNF-09	Seguridad	Las contraseñas se hashean mediante el algoritmo MD5
RNF-10	Seguridad	La autenticación o inicio de sesión en la aplicación por parte de los usuarios se realiza a través de las credenciales de login y contraseña
RNF-11	Disponibilidad	El sistema debe tener una disponibilidad 24/7 (24 horas al día los 7 días de la semana)
RNF-12	Disponibilidad	La disponibilidad del sistema cada vez que el usuario final intente acceder al mismo será del 99%
RNF-13	Implementación	El diseño de la interfaz de usuario será desarrollado con HTML y CSS
RNF-14	Integridad	Cumplirá con la ley de protección de datos

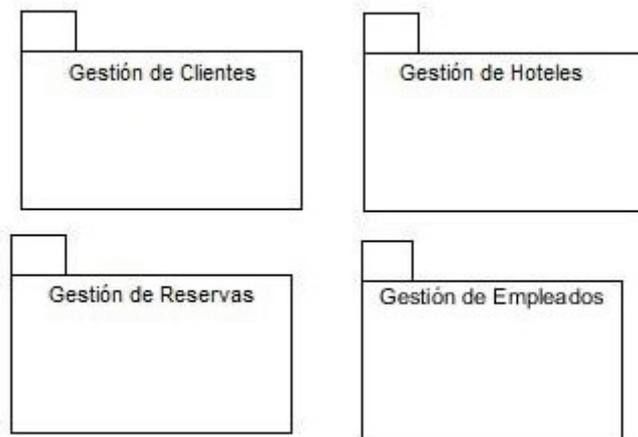
5. Fase de análisis

5.1 Modelo de Análisis

En esta etapa estudiaremos los requisitos de la fase de obtención de requisitos.

Se ha dividido el programa en el siguiente conjunto de paquetes:

Figura 7. Paquetes del Modelo de Análisis

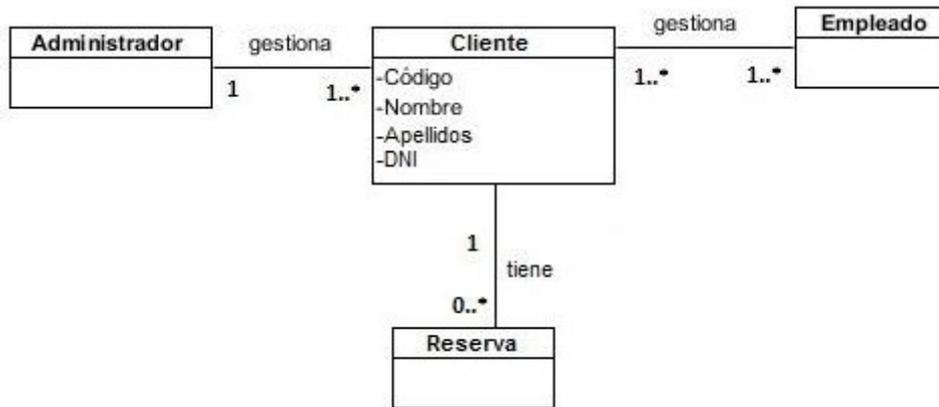


Fuente: Elaboración propia

No se ha creado un paquete para la Gestión de Sesiones por no considerarse muy significativo de cara a la documentación del sistema.

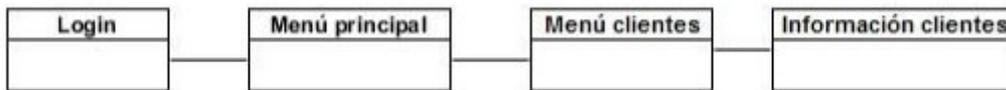
5.2 Gestión de Clientes

Figura 8. Diagrama de clases de Entidad del apartado de Gestión de Clientes



Fuente: Elaboración propia

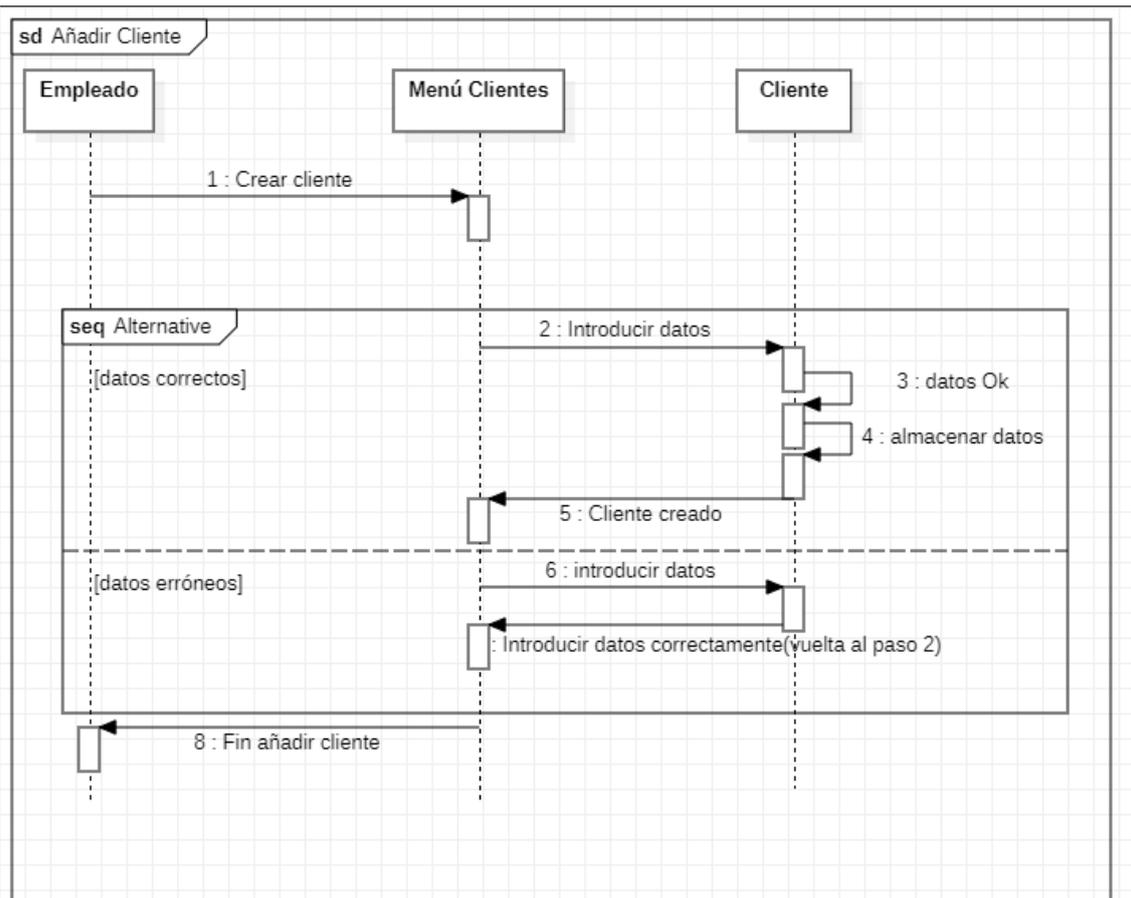
Figura 9. Diagrama de clases de Interfaz del apartado de Gestión de Clientes



Fuente: Elaboración propia

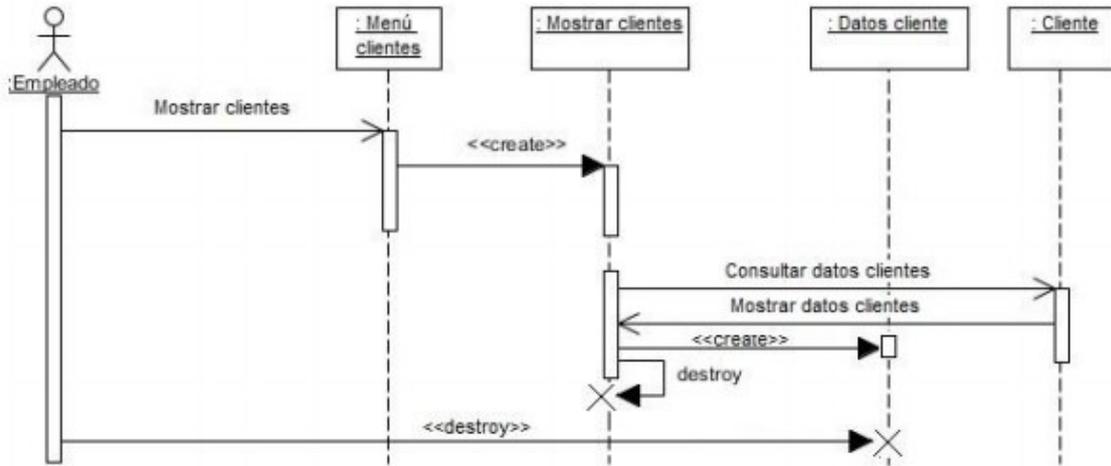
5.2.1 Diagramas de secuencia de gestión de clientes

Figura 10. Diagrama de secuencia de "Añadir cliente"



Fuente: Elaboración propia

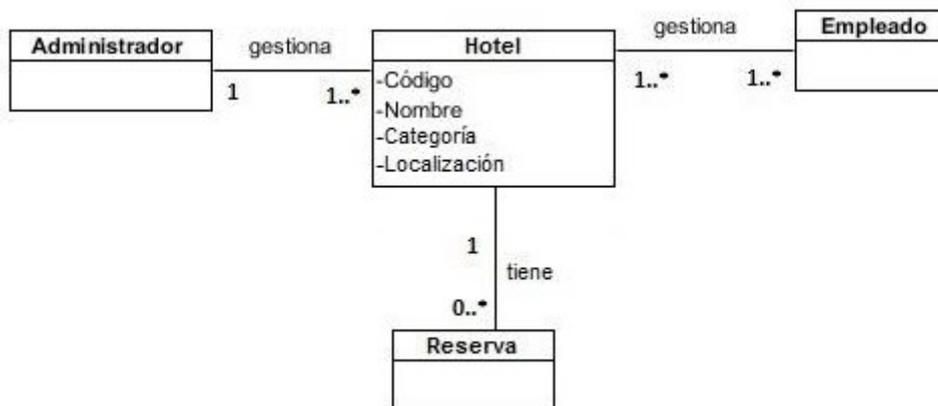
Figura 11. Diagrama de secuencia de "Mostrar clientes"



Fuente: Elaboración propia

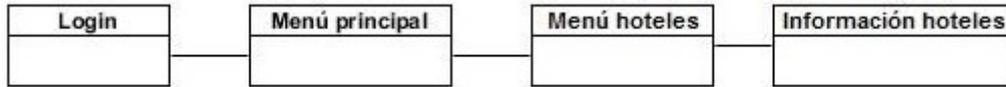
5.3 Gestión de Hoteles

Figura 12. Diagrama de clases de Entidad del apartado de Gestión de Hoteles



Fuente: Elaboración propia

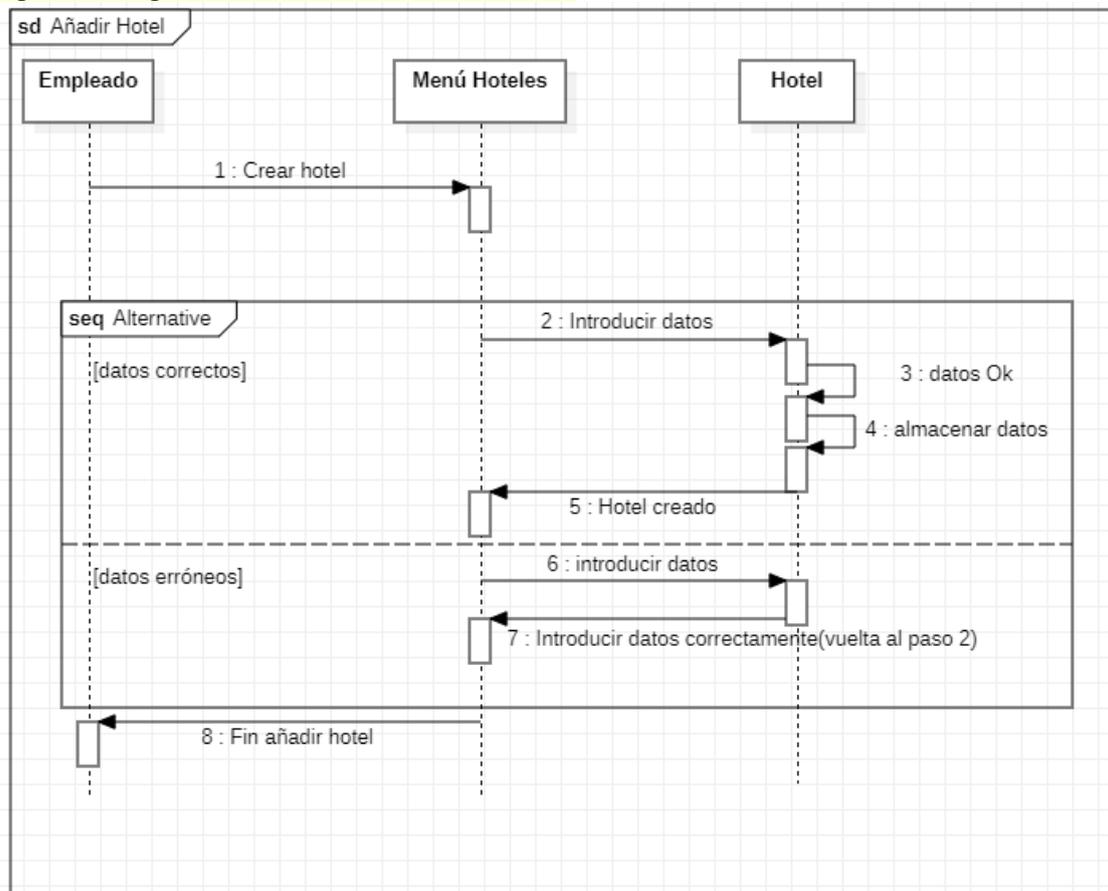
Figura 13. Diagrama de clases de Interfaz del apartado de Gestión de Hoteles



Fuente: Elaboración propia

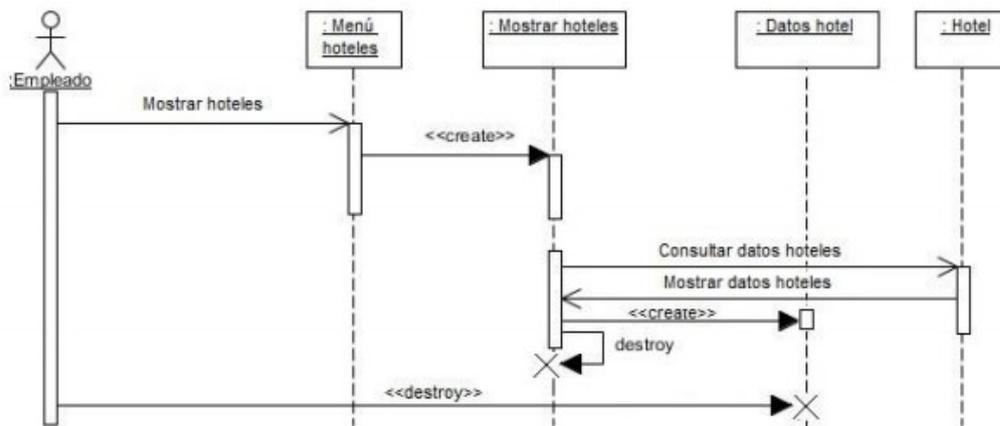
5.3.1 Diagramas de secuencia de gestión de hoteles

Figura 14. Diagrama de secuencia de "Añadir hotel"



Fuente: Elaboración propia

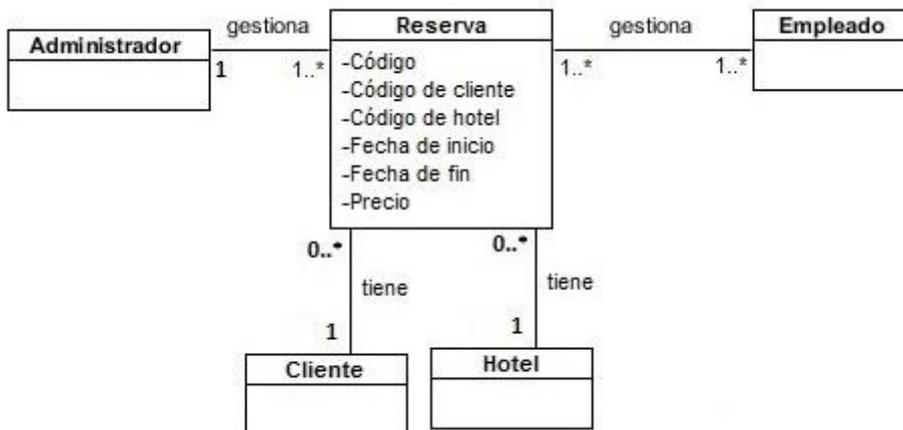
Figura 15. Diagrama de secuencia de Mostrar hoteles”



Fuente: Elaboración propia

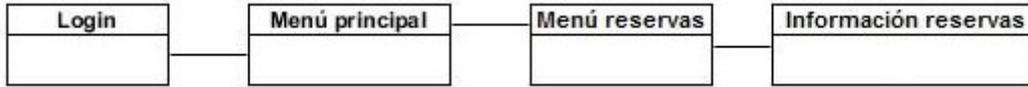
5.4 Gestión de Reservas

Figura 16. Diagrama de clases de Entidad del apartado de Gestión de Reservas



Fuente: Elaboración propia

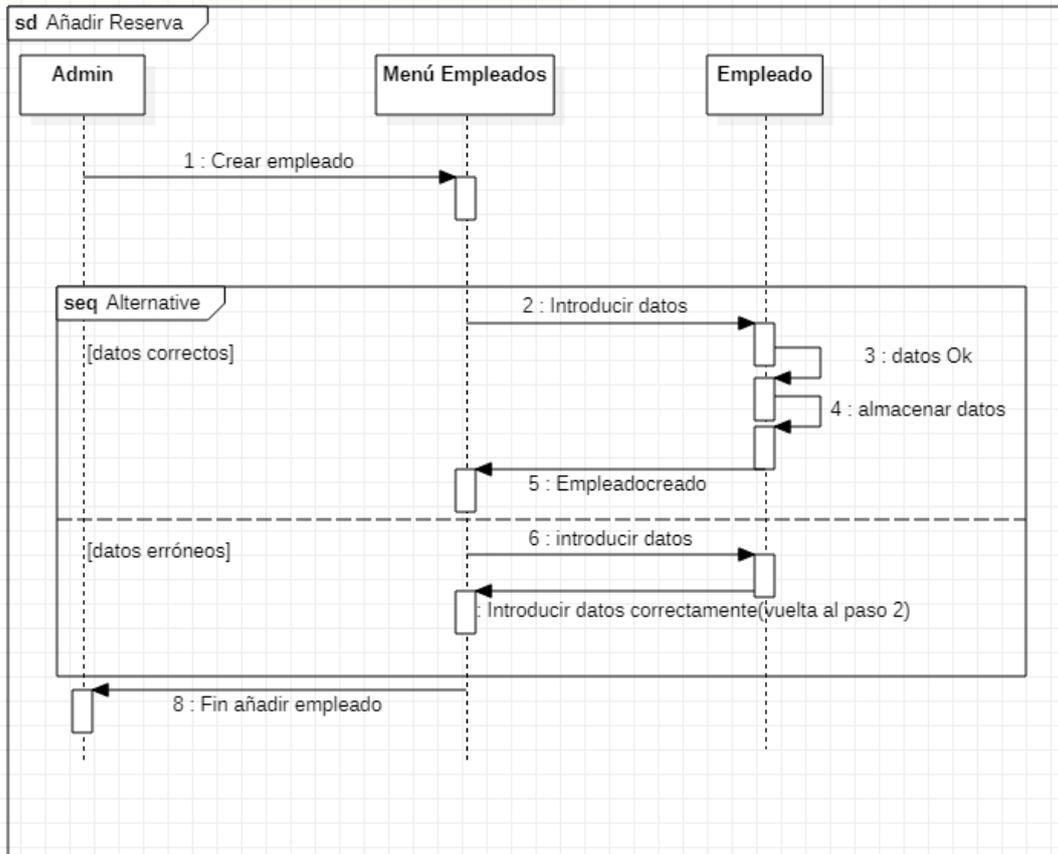
Figura 17. Diagrama de clases de Interfaz del apartado de Gestión de Reservas



Fuente: Elaboración propia

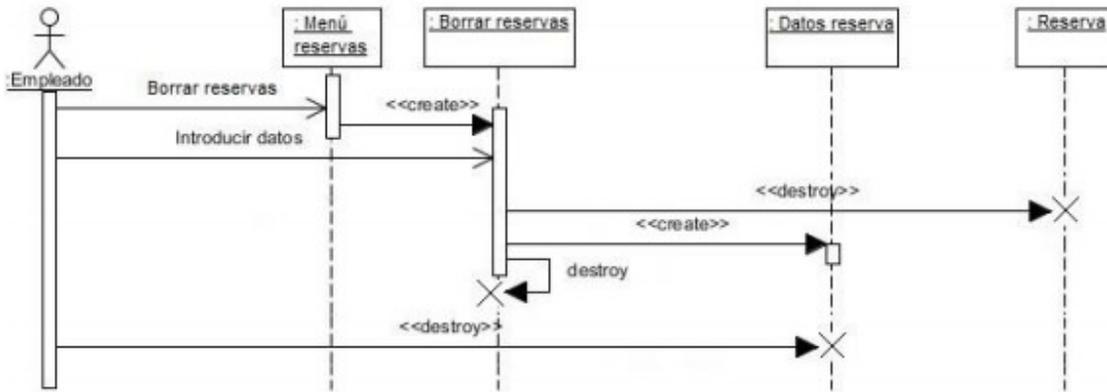
5.4.1 Diagramas de secuencia de gestión de reservas

Figura 18. Diagrama de secuencia de "Añadir reserva"



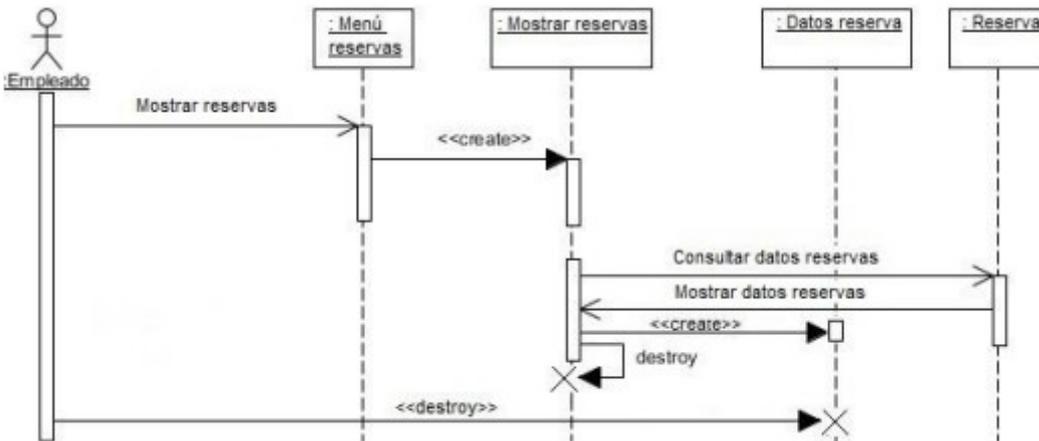
Fuente: Elaboración propia

Figura 19. Diagrama de secuencia de "Borrar reservas"



Fuente: Elaboración propia

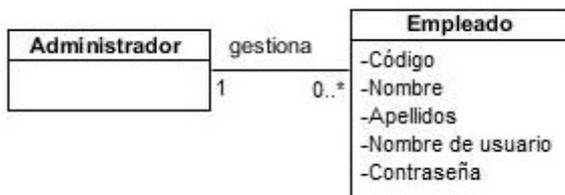
Figura 20. Diagrama de secuencia de "Mostrar reservas"



Fuente: Elaboración propia

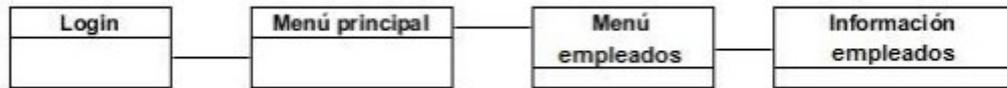
5.5 Gestión de Empleados

Figura 21. Diagrama de clases de Entidad del apartado de Gestión de Empleados



Fuente: Elaboración propia

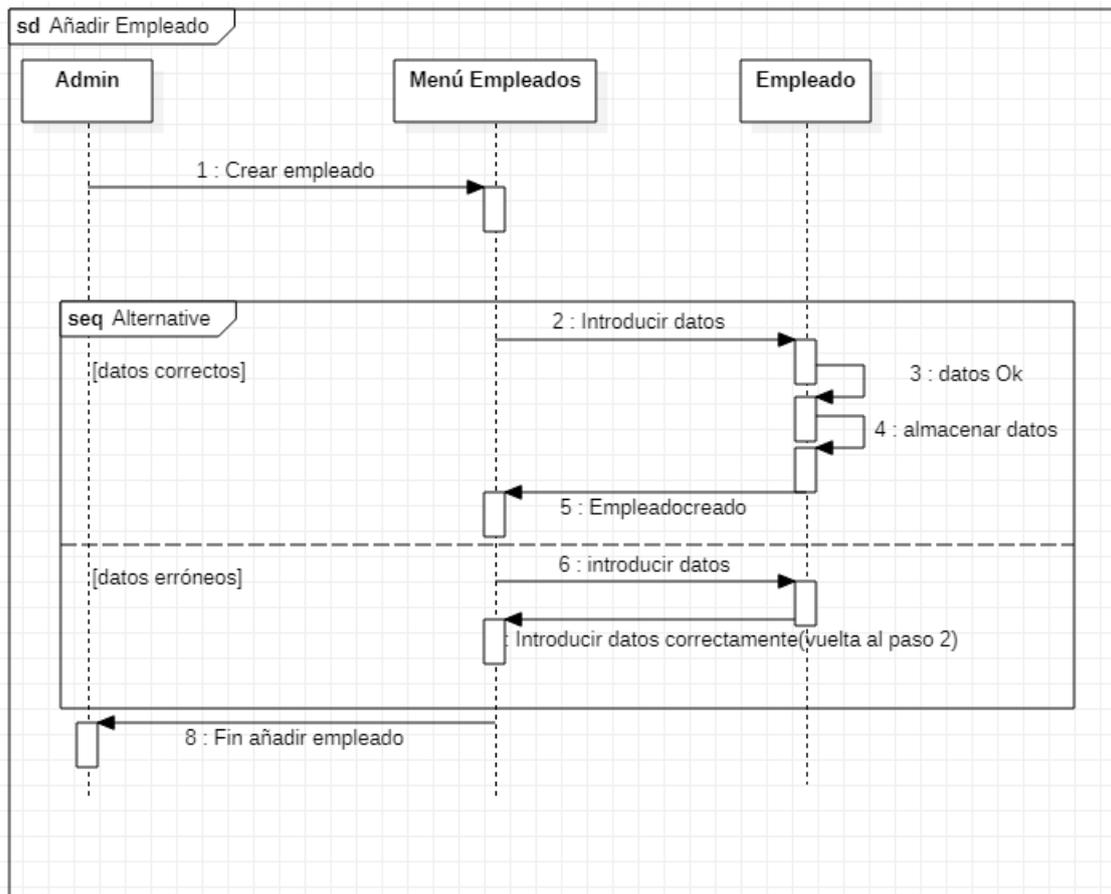
Figura 22. Diagrama de clases de Interfaz del apartado de Gestión de Empleados



Fuente: Elaboración propia

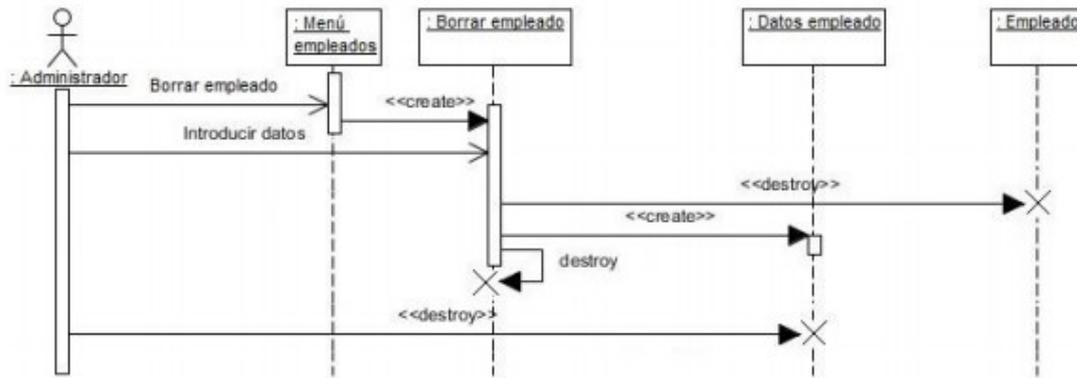
5.5.1 Diagramas de secuencia de gestión de empleados

Figura 23. Diagrama de secuencia de “Añadir empleado”



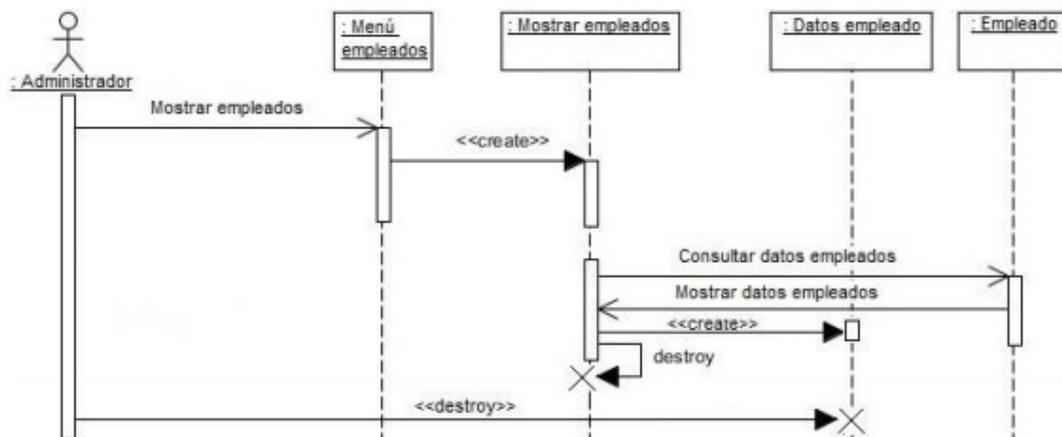
Fuente: Elaboración propia

Figura 24. Diagrama de secuencia de “Borrar empleado”



Fuente: Elaboración propia

Figura 25. Diagrama de secuencia de "Mostrar empleados"



Fuente: Elaboración propia

5.6 Diagrama de Clases

5.5.1 Descripción del sistema actual

Cliente – El cliente será una agencia de viajes ficticia.

Usuarios – Habrá 2 tipos de usuarios:

Administrador: El administrador se encarga de la gestión de la información relativa a los empleados de la agencia de viajes y podrá ejercer cualquier actividad que puedan desempeñar éstos.

Empleados: Son, junto con el administrador, los encargados de la gestión de la información relativa a los clientes, los hoteles, y las reservas de la agencia.

El administrador que se encarga de dirigir la gestión de la agencia de viajes utilizará una base de datos en la que se almacenarán todos los datos referentes a los empleados, los clientes, los hoteles y las reservas que se realicen en la agencia.

ADMINISTRADOR

- Nombre de usuario (nick), contraseña de acceso, código de usuario, nombre de usuario.

EMPLEADOS

- Nombre de usuario (nick), contraseña de acceso, código de empleado, nombre de usuario.

CLIENTES

- Código de cliente, nombre, apellidos, reservas asociadas y DNI.

HOTELES

- Código de hotel, nombre, categoría, localización.

RESERVAS

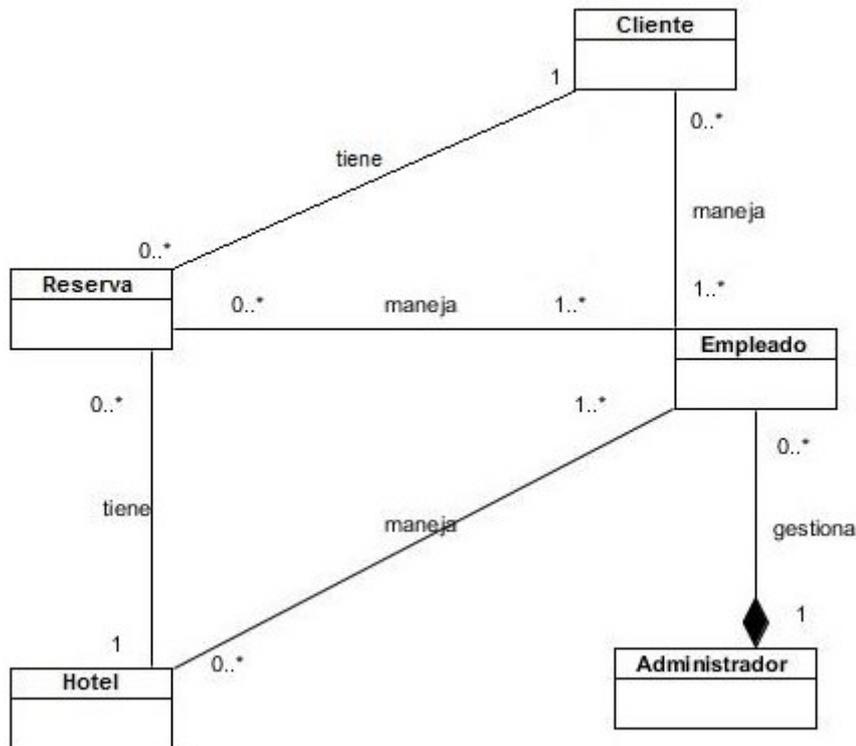
- Código de reserva, código de cliente, código de hotel, fecha de inicio, fecha de fin, precio.

Toda esta información se guardará en una base de datos, obtenida a través de consultas en lenguaje SQL.

El programa se manipulará a través de una página web en lenguaje HTML (HyperText Markup Language).

5.6.2 Diagrama de clases entidad

Figura 26. Diagrama de clases de Entidad



Fuente: Elaboración propia

6. Fase de diseño

Este programa está escrito en Java, probablemente el lenguaje de programación más usado en el mundo que ofrece grandes funcionalidades en cuanto a programación orientada a objetos. Para su correcto funcionamiento es preciso instalar en el sistema operativo el JDK (Java Development Kit), que contiene las librerías de Java precisas para el desarrollo de la funcionalidad que se desea acometer. Para el desarrollo de ésta aplicación se ha utilizado el JDK 8.

El programa estará orientado a objetos y trabajará con bases de datos relacionales y conexiones en una red local. Utilizaremos el sistema gestor de bases de datos MySQL para establecer las consultas a la base de datos, ya que es eficaz y garantiza la concurrencia, la integridad de los datos y la seguridad. Debemos tener en cuenta que MySQL es multihilo y multiplataforma y ofrece una gran portabilidad.

Se creará una página web en lenguaje HTML, lo que nos posibilitará utilizar la aplicación desde cualquier ordenador en red de la agencia de viajes. Se usa HTML porque es un lenguaje de páginas web muy intuitivo y fácil de manejar y en el que están escritas la inmensa mayoría de páginas web de la red actualmente.

El programa se ha desarrollado utilizando el patrón Modelo-Vista-Controlador (MVC), que es un modelo según el cual se divide una aplicación web en tres capas independientes e interconectadas para facilitar su gestión. Está diseñado para programas con grandes cantidades de datos, o que realizan complejas operaciones. Es una de las mejores alternativas para desarrollar una aplicación de alto nivel, por ello se incluye en la mayoría de frameworks (entornos de desarrollo) existentes en el mercado.

La capa de “modelo” sirve para hacer efectiva la interacción del programa con los datos, así recibe peticiones de datos y desarrolla operaciones sobre la fuente de datos a la que esté conectada. La parte de “vista” tiene por misión establecer comunicación con el usuario, recogiendo las acciones desea ejecutar y devuelve los resultados que ha devuelto el programa, todo esto en un entorno que cumpla unos requisitos de diseño y desarrollo. El apartado de “controlador” gestiona la lógica de la aplicación y la de negocio, y controla parámetros como la seguridad u otras funciones específicas.

Se ha utilizado para la programación de una parte significativa de la funcionalidad de la aplicación un framework o entorno de desarrollo llamado Struts2. Los frameworks permiten trabajar mediante una serie de elementos de control y gestión en el código de los programas que aumentan su grado de calidad de forma fácil y rápida. Los frameworks permiten la reutilización de código, la modularidad (que permite un desarrollo estructurado de la aplicación), seguridad (esto es posible gracias a las funciones del framework en buena medida) y fiabilidad (suelen estar soportados por comunidades de desarrollo y empresas que defienden su evolución). Como rasgos generales, señalar que éstas herramientas permiten el control de acceso de los usuarios en la aplicación, el acceso a las bases de datos a través de librerías e interfaces, poseen mecanismos de control y validación de eventos y posibilitan el control de sesiones y URL.

Concretamente, el framework Struts2 es un entorno de desarrollo para el desarrollo web que permite que la implementación de las aplicaciones sea más fácil y más rápido. Permite también que éstas aplicaciones sean más flexibles y robustas.

Los datos se almacenan en una base de datos, que es generada mediante consultas en lenguaje SQL (Lenguaje Estructurado de Consulta) y gestionada a través de la presente aplicación en Java mediante el sistema gestor de bases de datos MySQL y un archivo con extensión “.jar” llamado “MySQL Connector Java” que permite el acceso

entre el lenguaje Java y la base de datos (que se almacenará en un archivo con extensión “.sql”).

El programa ofrecerá la posibilidad de visualizar los datos mostrados en cualquier lista de la aplicación en formato CSV, XLS (Excel), XML, PDF y RTF.

Para manejar la administración de MySQL a través de páginas web, se ha utilizado la herramienta “Mysql Workbench”, que permite gestionar y trabajar con bases de datos y los elementos que las conforman (tales como tablas y registros, así como el diseño del modelo relacional), ejecutar sentencias en lenguaje SQL, gestionar privilegios, así como exportar datos en varios formatos.

La aplicación también trabajará mediante servlets (sobre todo a la hora de controlar las sesiones), que consisten en una serie de programas que se encargan de permitir el establecimiento de varias conexiones simultáneas a una misma página web.

Para la gestión de los servlets y de las páginas JSP se ha empleado una aplicación llamada Tomcat, que es un contenedor de servlets y páginas JSP. Este programa posibilita recibir peticiones, dirigirlas a los servlets o páginas JSP concretas, ejecutarlos y devolver la respuesta al cliente. Su instalación no resulta demasiado difícil, teniendo en cuenta que viene explicada con detalle en el manual de la asignatura de Desarrollo web I.

Para acceder a phpMyAdmin, así como al programa Tomcat, se ha utilizado la consola de Xampp, concretamente la versión Xampp Control Panel v.3.2.1. Esta herramienta consiste en un servidor independiente de plataforma y es software libre. Este programa da acceso al sistema gestor de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. Xampp está desarrollado como un servidor web libre, sencillo de usar y capaz de interpretar páginas web dinámicas.

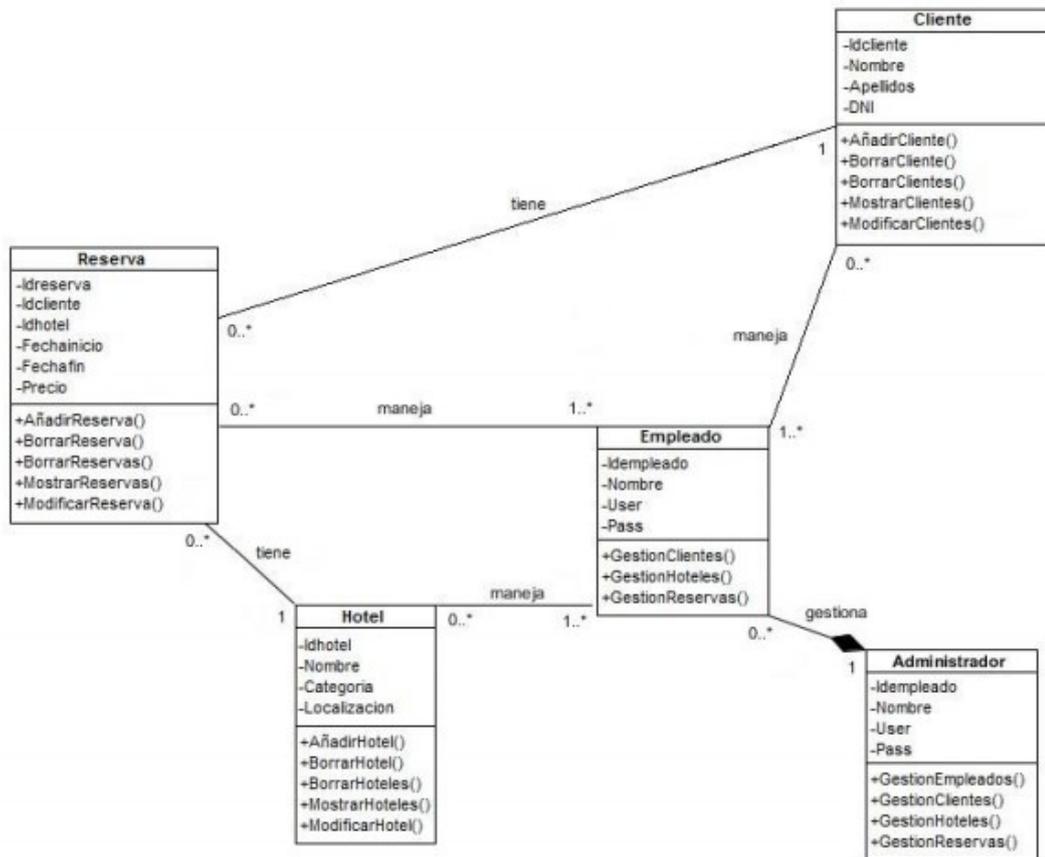
La aplicación se manejará mediante una página web a la que se accede mediante una red local. Para ello se creará dicha web en lenguaje HTML y se modelará mediante un CSS que le dé un buen aspecto a la web.

Como editor del programa se ha usado Eclipse, que es un entorno de desarrollo integrado libre, creado fundamentalmente para el lenguaje de programación Java. Existe un gran número de módulos para ampliarlo. Eclipse IDE es un producto gratuito y libre y sin restricción de uso alguna. Esta plataforma hace posible que las aplicaciones sean creadas a partir de un conjunto de componentes de software llamados módulos (son archivos Java que poseen clases de Java escritas para interactuar con las APIs de Eclipse y un archivo especial que lo identifica como módulo). Concretamente se ha usado la versión Eclipse IDE 2022-12, una de las más novedosas, si no la que más.

Todos los programas utilizados y anteriormente mencionados son de libre distribución y gratuitos.

6.1. Modelo de Diseño

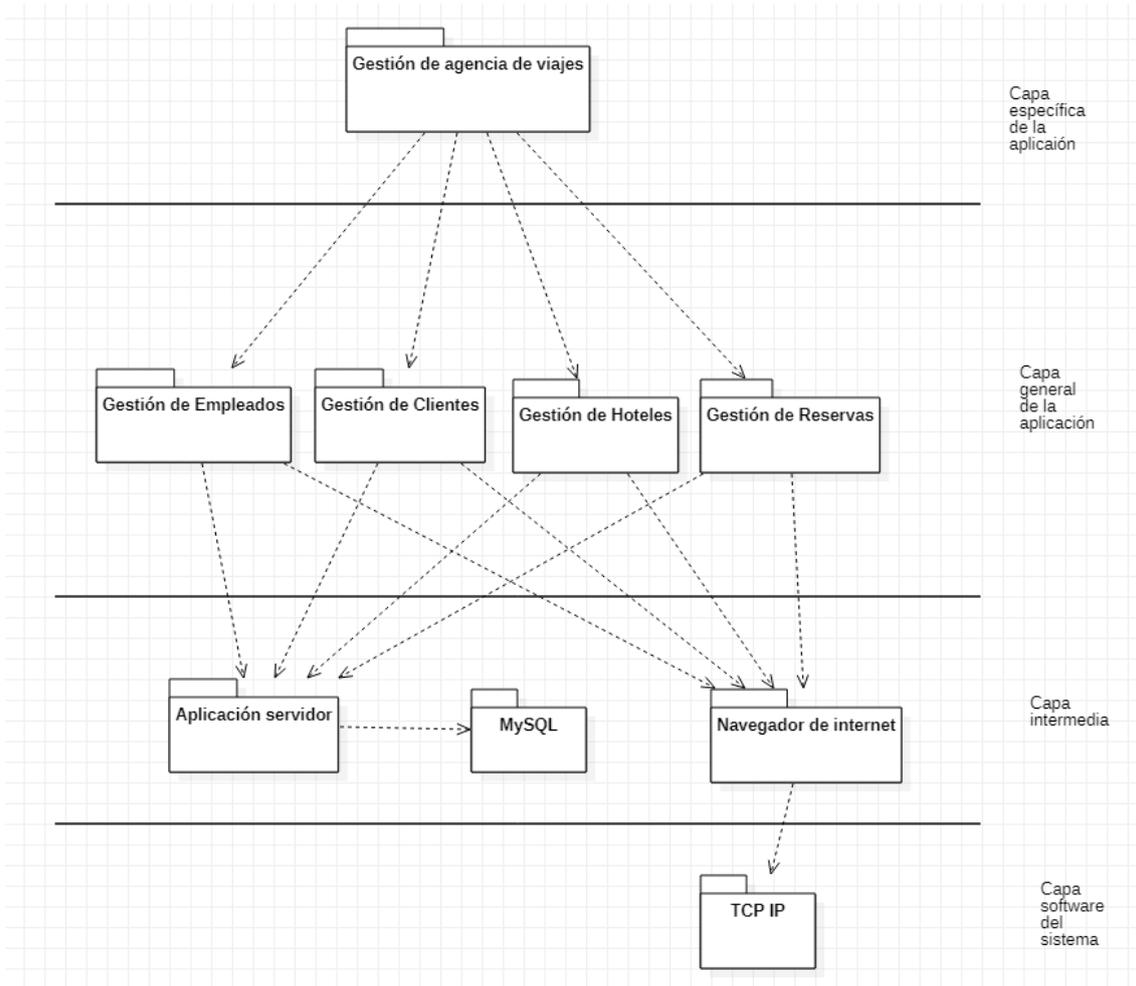
Figura 27. Diagrama de clases



Fuente: Elaboración propia

6.2. Modelo arquitectónico

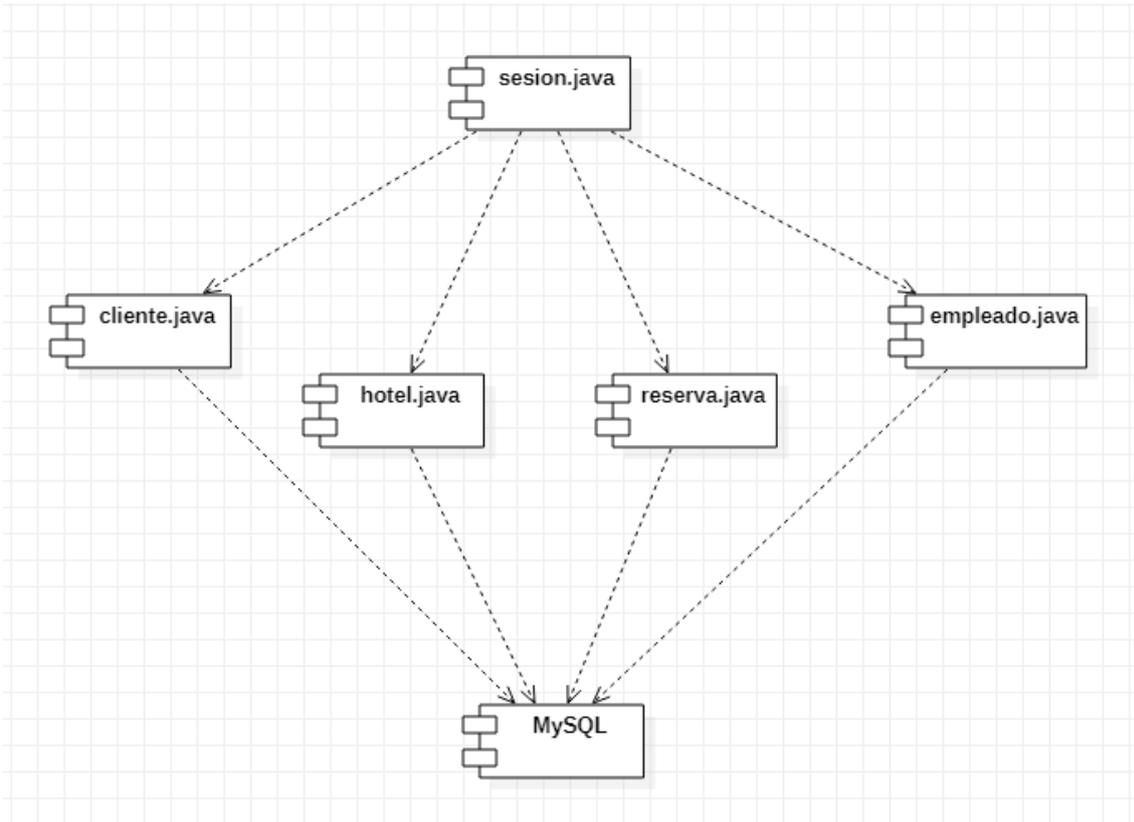
Figura 28. Gráfico Identificación de subsistemas



Fuente: Elaboración propia

6.3. Diagrama de componentes

Figura 29. Diagrama de componentes

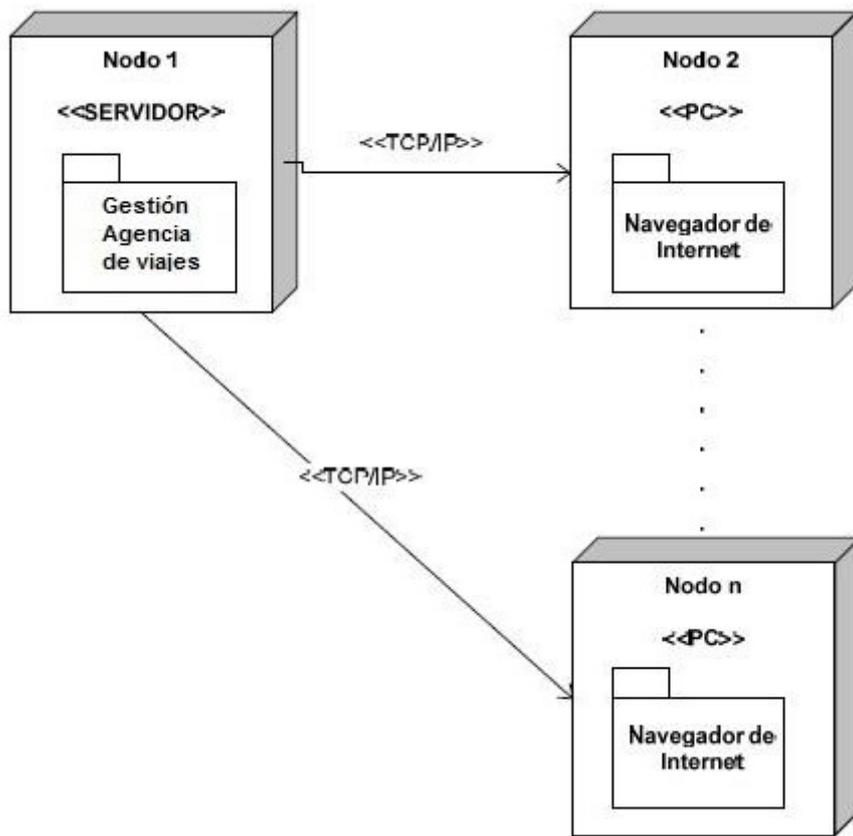


Fuente: Elaboración propia

6.4. Diagrama de despliegue

Este diagrama representa la distribución real de la aplicación de la agencia de viajes en los equipos informáticos de ésta.

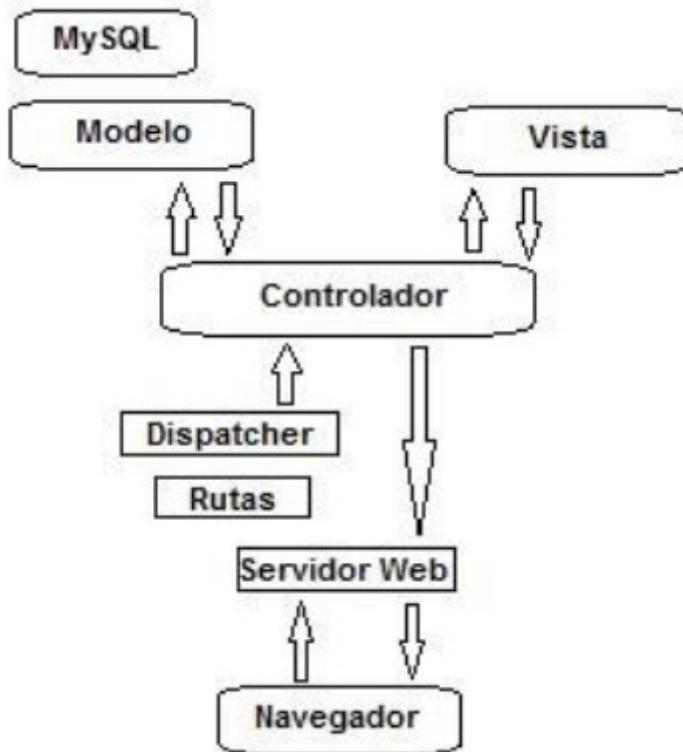
Figura 30. Diagrama de despliegue



Fuente: Elaboración propia

6.5. Diagrama del patrón Modelo-Vista-Controlador

Figura 31. Esquema de interacción del patrón Modelo-Vista-Controlador (MVC)



Fuente: Elaboración propia

7. Prueba de errores

En este capítulo se muestran las pruebas que se han llevado a cabo en el sistema para poder asegurar un funcionamiento adecuado en el caso de que se produzca un comportamiento, situación o respuesta no esperada. Para ello, se realizarán dos tipos de pruebas.

7.1 Pruebas de caja blanca

El objetivo principal de este tipo de pruebas es de validar las funciones internas de los módulos que se van a probar. Se basan en los detalles procedimentales del sistema, y por tanto se encuentran muy relacionadas con el código desarrollado.

El funcionamiento de estas pruebas consiste en coger secciones de código y dividirlos en bloques, de manera que se traten como procesos independientes y que se procesan de manera secuencial. De esta manera, se

consigue rastrear los flujos de actuación que puede adoptar un programa para asegurarnos de que realiza las acciones de la forma esperada.

Se han realizado diversas pruebas de errores relativas a localizar y eliminar los posibles errores que en el transcurso del desarrollo de la aplicación pueden haberse producido.

Al introducir en un campo numérico un valor de tipo texto, el sistema muestra un mensaje por pantalla para que el usuario de la aplicación introduzca un valor numérico. Si por el contrario se introduce un valor numérico en un campo de tipo texto, el sistema también mostrará un error por pantalla.

Si se deja algún campo en blanco, el sistema muestra también un error por pantalla, por estar todos los campos establecidos como requeridos y por tanto obligatorios de ser rellenados.

Al intentar borrar un cliente u hotel que tenga reservas asociadas, no se borrará el registro en cuestión y se mostrará un error por pantalla diciendo que el cliente u hotel a borrar posee reservas asociadas, y que por tanto hay que borrar las reservas asociadas de ese hotel o cliente.

7.2 Pruebas de caja negra

El objetivo de las pruebas de caja negra es de validar las respuestas o salidas que el sistema produce cuando el usuario introduce unas entradas determinadas. No se analiza el funcionamiento interno de sistema, ya que se observa el sistema desde el punto de vista del usuario, que desconoce de los procesos que la aplicación realiza de manera interna.

Las pruebas que se han realizado en este proyecto son las siguientes:

Tabla 42. – PCN-01 Añadir cliente

PCN-01	Añadir cliente
Objetivo	Comprobar que se puede crear un cliente en el sistema
Precondiciones	N/A
Datos de entrada	Nombre: Miguel Apellidos: Extremo Nicolás DNI: 70835493R

Acción esperada	Al pulsar en el botón “Guardar cambios”, se mostrará los datos del cliente creado y se almacenará el cliente creado en el sistema.
Resultado	Correcto

Tabla 43. – PCN-02 Borrar reserva

PCN-02	Borrar Reserva
Objetivo	Comprobar que se puede eliminar una reserva que se encuentra en el sistema.
Precondiciones	La reserva que se desea eliminar debe existir en el sistema.
Datos de entrada	N/A
Acción esperada	Al pulsar en el botón “Eliminar reserva”, se borrará la reserva del sistema y se mostrará el listado de reservas al usuario sin la reserva que ha sido eliminada.
Resultado	Correcto

8. Documentación

8.1 Manual de usuario

En esta sección se detalla un manual para el uso de la aplicación web. Para ello, se debe abrir un navegador web e introducir la url correspondiente a la aplicación.

Dentro de la aplicación lo primero que se tiene que hacer es acceder mediante un usuario y contraseña, ya seas un empleado de la agencia o el administrador de la misma.



Universidad de Valladolid

VIAJES EXTREMOS

GRADO EN INGENIERÍA DE SERVICIOS Y APLICACIONES

PÁGINA JSP DE ENTRADA A LA PARTE PRIVADA DE LA APLICACIÓN

Usuario:
Contraseña:

Viajes Extremos © UVA 2023

Una vez que ya hemos accedido con las credenciales, nos aparece en pantalla un menú horizontal y debajo las instrucciones de cómo pasar a la parte de gestión.

Cuando ya estamos en la parte de gestión en el menú horizontal podemos acceder a las distintas partes de la web como puede ser la gestión de los clientes, de los hoteles o de las reservas.

Si le damos a la opción de gestión de clientes, pasaremos a ver el listado de clientes y las distintas funcionalidades que tenemos como crear nuevos clientes, borrar clientes existentes o modificar datos de cualquiera de los clientes ya existentes.



Universidad de Valladolid

VIAJES EXTREMOS

GRADO EN INGENIERÍA DE SERVICIOS Y APLICACIONES

¡HOLA!

GESTIÓN DE CLIENTES

GESTIÓN DE HOTELES

GESTIÓN DE RESERVAS

SALIR

Esta es la parte privada de la aplicación.
Bienvenido Miguel Extremo
con usuario miguelx

PÁGINA JSP QUE MUESTRA UN LISTADO PRIVADO

Se puede crear un cliente nuevo [aquí](#).

[Borrar todos los clientes](#)

[First/Prev] 1, 2, 3 [Next/Last]

Código	Nombre	Apellidos	DNI		
1	Diego	Perez	02438014A		
2	Jesús	Sánchez	72589672Y		
5	Miguel	Extremo	70835493R		
7	Pepe	Perez	70264737G		
10	Manolo	García	38596472F		

Export options: CSV | Excel | XML

VIAJES EXTREMOS © UVA 2023

Para moverse en la web se debe usar los distintos botones del menú hasta que se quiera salirse de la aplicación. Para hacer logout se debe pulsar la opción “salir” y así volver a la página de login.

¡HOLA!

GESTIÓN DE CLIENTES

GESTIÓN DE HOTELES

GESTIÓN DE RESERVAS

SALIR

9. AMPLIACIONES

A esta aplicación se le podrían agregar diversas funcionalidades adicionales, tales como la elaboración de facturas, así como los libros de contabilidad de la agencia de viajes. También sería posible ampliar el rango de tipos de reservas que pueden efectuarse para un cliente, tales como cruceros y circuitos que elaboran gran cantidad de tour operadores presentes en el mercado.

No se ha incluido en la aplicación ninguna función para comprobar si los hoteles tienen habitaciones disponibles, ya que éste trabajo no es competencia de la agencia de viajes, sino de la empresa mayorista, que son los proveedores con los que cuentan las agencias de viaje a la hora de suministrar gran cantidad de servicios al cliente (la mayorista es quien reserva la habitación del hotel o la plaza de crucero para la agencia en cuestión).

Además, se pretende añadir un método de cobros dentro de la aplicación a la hora de confirmar las reservas, donde existirá la posibilidad de introducir métodos de pago a través de tarjeta de crédito. Mientras tanto la aplicación será sostenida por anuncios publicitarios de terceros para tener una inyección de beneficios para el costo del mantenimiento.

Referencias

Página oficial de Eclipse: utilizada para descargar e instalar Eclipse IDE.

-Eclipse IDE: Disponible en <https://www.eclipse.org/downloads/>

Página oficial de MySQL: utilizada para descargar e instalar MySQL server y MySQL Workbench.

-MySQL: Disponible en <https://dev.mysql.com/downloads/mysql/>

Página oficial de StarUML: utilizada para descargar e instalar StarUML.

-StarUML: Disponible en <https://staruml.io/download/>

Página de StackOverflow: utilizada para solucionar dudas o errores en el código.

-StackOverflow. Disponible en <https://stackoverflow.com/>

Página oficial de Struts2 de Apache: utilizada para aprender y consultar dudas sobre el framework Struts2.

-Struts2. Disponible en <https://struts.apache.org/index.html>

Página de W3School: utilizada para consultar dudas sobre diferentes lenguajes de programación.

-W3School. Disponible en <https://www.w3schools.com/>