



**HOTEL ZARGO**

# **ESPECIFICACIÓN DE REQUISITOS**

Ignacio Cerdá Sánchez  
Noel Clemente Montero  
Gorka Jimeno Garrachón  
Pablo Olivera Zaldúa  
Álvaro Sáez Hernando  
Luis Valero Martín

14 Enero de 2013

Especificación de Requisitos  
Hotel Zargo





## **1. Introducción**

1.1 Propósito	4
1.2 Alcance	4
1.3 Definiciones, acrónimos y abreviaturas	4
1.4 Referencias	6
1.5 Resumen	6

## **2. Descripción**

2.1 Perspectiva del producto	7
2.2 Funciones del producto	7
2.3 Características del usuario	9
2.4 Restricciones	9
2.5 Supuestos y dependencias	9
2.6 Requisitos futuros	10

## **3. Requisitos específicos**

3.1 Interfaces externos	11
3.2 Funciones	12
3.3 Requisitos de rendimiento	26
3.4 Requisitos lógicos de la base de datos	26
3.5 Restricciones de diseño	27
3.6 Atributos del sistema software	28

## **4. Apéndices**

Anexo I: Gestión de cambios	29
-----------------------------	----



## 1. Introducción

### 1.1 Propósito

Este documento indica los objetivos del proyecto de software para la gestión de un hotel dentro de la asignatura de ingeniería del software, así como la forma de interacción con el usuario, las restricciones, las condiciones del proyecto y los requisitos necesarios.

### 1.2 Alcance

El objetivo del proyecto es ayudar y facilitar la gestión tanto de las habitaciones, como de las reservas, clientes y empleados de un hotel.

El software será accesible para los empleados del hotel cuyas competencias estén relacionadas con la gestión de este. Éstos podrán acceder mediante los ordenadores conectados a la red interna del hotel.

### 1.3 Definiciones, acrónimos y abreviaturas

Usaremos las siguientes abreviaturas:

**Backup:** (Copia de seguridad) El fin de estas copias adicionales es que puedan utilizarse para restaurar el original después de una eventual pérdida de datos.

**BDD o BD:** Base de datos o banco de datos. Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

**HTTPS:** (Hypertext Transfer Protocol Secure) En español, *Protocolo seguro de transferencia de hipertexto*, más conocido por sus siglas **HTTPS**, es un protocolo de aplicación basado en el protocolo HTTP, destinado a la transferencia segura de datos de hipertexto, es decir, es la versión segura de HTTP. Es utilizado principalmente por cualquier tipo de servicio que requiera el envío de datos personales o contraseñas.

**HW:** (Hardware) Conjunto de los componentes que integran la parte material de una computadora.



**IEEE:** (Instituto de Ingenieros Eléctricos y Electrónicos) Es una asociación técnico profesional mundial dedicada a la estandarización. Es la mayor asociación internacional sin ánimo de lucro formada por profesionales de las nuevas tecnologías.

**JAVA EE:** Lenguaje de programación diseñado para la creación de aplicaciones empresariales y web.

**JRE:** Java Runtime Environment o JRE es un conjunto de utilidades que permite la ejecución de programas Java.

**LOPD:** La Ley Orgánica 15/1999 de 13 de diciembre de Protección de Datos de Carácter Personal, (LOPD), es una Ley Orgánica española que tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor, intimidad y privacidad personal y familiar.

**MySQL:** Sistema de gestión de bases de datos basado en software libre.

**MVC:** Modelo Vista Controlador, es un patrón de arquitectura que divide la aplicación interactiva en tres componentes, modelo (contiene la funcionalidad básica y los datos), vistas (muestra/recoge información al/del usuario) y controladores (median entre vistas y modelo)

**SRS:** Especificación de Requisitos de Software (SRS). Define de forma precisa el producto de software que se va a construir.

**SSH:** SSH (Secure Shell, en español: intérprete de órdenes segura) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red

**SW:** (Software) Es el equipamiento lógico o soporte lógico de un sistema informático, requiere de elementos Hardware para su uso.

**TCP/IP:** Protocolo de red que permite la comunicación entre computadoras.



## 1.4 Referencias

- IEEE Std 830 1998(R2009). Especificación de Requisitos según el estándar de IEEE 830.LOPD
- Para elaborar la documentación hemos recurrido a las transparencias de las asignaturas de Ingeniería del Software impartidas por Antonio Navarro Martín.

## 1.5 Resumen

Se desea crear una aplicación software para facilitar la gestión de un hotel, de tal forma que se pueda tener un registro de los datos de los clientes, de los datos de los empleados y de las habitaciones del hotel. Contando con éstos registros la aplicación también podrá efectuar reservas de una o varias habitaciones por parte de un cliente, en las que se indicarán las condiciones de la reserva.

Para explicar la especificación de requisitos del proyecto, este documento se divide en tres secciones:

- Sección 1: Introducción. Proporciona una visión general del objetivo de la aplicación.
- Sección 2: Descripción de los factores generales que afectan al producto y los requisitos principales que conforman su desarrollo.
- Sección 3: Definición específica de los requisitos generales que satisfacen al sistema, así como las funciones y restricciones que llevará a cabo.



## 2. Descripción general

### 2.1 Perspectiva del producto

Pretendemos crear un sistema SW para gestionar un hotel. A través de esta aplicación podremos saber el estado de cada habitación y de los empleados del hotel.

Los empleados autorizados podrán gestionar las reservas y los clientes.

Habrà dos tipos de empleados:

- Servicios.
- Administración.

Habrà dos tipos de clientes:

- Empresas.
- Particulares.

Como administrador podrá gestionar las habitaciones, los empleados y los turnos.

### 2.2 Funciones del producto:

El software consta de siete módulos principales, cuyas funciones se describen a continuación:

Módulos:

**2.2.1 Clientes:** Gestiona la información de los clientes del hotel.

- 1) Dar de alta un cliente.
- 2) Dar de baja un cliente.
- 3) Modificar datos de un cliente.
- 4) Listar clientes.

**2.2.2 Habitaciones:** Gestiona la información de las habitaciones del hotel, lo que consta en cada habitación y su estado (si están ocupadas o no).

- 1) Dar de alta una habitación
- 2) Dar de baja una habitación
- 3) Modificar datos de una habitación.



#### 4) Listar habitaciones

**2.2.3 Empleados:** Gestiona la información de los empleados que trabajan en el hotel. Y especifica, mediante las opciones que se dan en la lista, el turno en el que trabaja cada empleado.

- 1) Dar de alta un empleado.
- 2) Dar de baja un empleado.
- 3) Modificar datos de un empleado.
- 4) Listar Empleados.

**2.2.4 Reservas:** Gestiona la información de las reservas de las habitaciones que los clientes del hotel han realizado. Y especifica, mediante las opciones que se dan en la lista, las condiciones que se ha contratado en la reserva.

- 1) Dar de alta una reserva.
- 2) Dar de baja una reserva.
- 3) Modificar datos de una reserva.
- 4) Listar reservas.
- 5) Hacer efectiva reserva.
- 6) Buscar disponibilidad.

**2.2.5 Turnos:** Gestiona la información de los turnos de los empleados. Y especifica las condiciones del turno.

- 1) Dar de alta un turno.
- 2) Dar de baja un turno.
- 3) Modificar datos de un turno.
- 4) Listar turnos.

**2.2.6 Servicios del hotel:** Gestiona la información de los turnos de los empleados. Y especifica las condiciones del turno.

- 1) Dar de alta un servicio.
- 2) Dar de baja un servicio.





- 3) Modificar datos de un servicio.
- 4) Listar servicio.

## 2.3 Características del usuario

El servicio está pensado para algunos empleados del hotel. Se precisan unos conocimientos básicos para usar la aplicación. Hay dos tipos de usuarios de la aplicación:

- Servicio: podrá visualizar su información, la de los clientes y las reservas.
- Administrador: podrá dar de alta, modificar y dar de baja las reservas, los clientes, las condiciones, servicios, habitaciones, empleados y turnos.

## 2.4 Restricciones

**Limitaciones hardware:** El sistema operativo deberá ser Microsoft Windows (XP o posterior) o Linux para los PC's.

**Operaciones en paralelo:** La base de datos podrá ser accedido simultáneamente por usuarios y administradores.

**Requisitos de lenguaje de alto nivel:** El desarrollo se realizará en Java usando Eclipse como IDE.

**Requisitos de fiabilidad:** La aplicación se ejecutará durante toda la jornada laboral para poder gestionar todos los servicios y el correcto funcionamiento del negocio, la tolerancia a un error en la aplicación es baja.

**Criticidad de la aplicación:** Las tareas de alta y baja de clientes tendrán una importancia baja, mientras que las peticiones de servicios y actualizaciones de los servicios del hotel deberán ser completadas con éxito a menos que haya un fallo muy grave.

**Consideraciones de robustez y seguridad:** Se deberán encriptar las contraseñas de los usuarios, además de garantizar la privacidad de los datos.

## 2.5 Supuestos y dependencias

**Supuestos:** El sistema deberá funcionar sobre Windows XP (o superior) y sobre Ubuntu 11.04 (o superior), pero las pruebas se realizarán sobre Windows 7.



No se requieren conocimientos de programación ni el uso de otros programas externos.

Para ejecutar nuestra aplicación será necesario disponer de un PC básico cuyos requisitos serán:

- Procesador Intel Pentium IV, AMD equivalente, o superior.
- Tarjeta gráfica
- Disco duro con un espacio libre de 200 MB
- Entorno operativo Windows XP (o superior) o Ubuntu 11.04 (o superior)

Para almacenar los datos generados se necesitará un PC más potente que actúe como servidor.

Por último suponemos que se dispone de conexión a una LAN en todo momento.

**Dependencias:** Para la ejecución de nuestro programa será necesario tener instalado Java JRE 1.6. A su vez se necesitará tener instalada una versión MySQL en el servidor para almacenar toda la información generada.

## 2.6 Requisitos futuros

De momento no se han descrito requisitos futuros.



### 3. Requisitos específicos

#### 3.1 Interfaces externos

Número de requisito	R1
Nombre de requisito	Requisito de Autenticación
Tipo	X Requisito                      Restricción
Fuente del requisito	Es imprescindible que el usuario se identifique para acceder al sistema. Deberá hacerlo mediante el login y password elegido en el registro.
Prioridad del requisito	X Alta/Esencial                      Media/Deseado                      Baja/ Opcional

Número de requisito	R2
Nombre de requisito	Requisito de Modificación
Tipo	X Requisito                      Restricción
Fuente del requisito	El administrador podrá modificar o cancelar las reservas de habitaciones, mientras que el usuario solamente las podrá consultar.
Prioridad del requisito	Alta/Esencial                      X Media/Deseado                      Baja/ Opcional

**Requisitos comunes de los interfaces:** La interfaz de autenticación, necesita un login y password válidos para acceder a la siguiente interfaz.

#### Interfaces de usuario:

La interfaz en uso, deberá mostrar al usuario solamente los datos necesarios para realizar cualquier operación de reserva, consulta, modificación o cancelación.

La interfaz en uso, deberá mostrar al administrador sólo la información necesaria para realizar cualquier modificación o cancelación.

Aquí se reflejan características lógicas de cada interfaz entre el producto del software y sus usuarios.

Características de la configuración:

- Formatos de la pantalla → Deberá contar con una resolución de 1280 x 1024 píxeles y orientación horizontal.
- Control por pestañas y esquemas de ventana intuitivos.
- Sistema de reportes opcional.
- Mensajes de error cortos, con información útil.



**Interfaces de hardware:** La pantalla deberá mostrar las interfaces así como la información necesaria para que el usuario pueda trabajar adecuadamente con el sistema.

El ratón: el sistema requerirá del ratón para que el usuario pueda realizar selecciones y oprimir botones.

El teclado: el sistema permitirá al usuario introducir datos mediante el teclado.

**Interfaces de software:** El sistema operativo deberá ser Microsoft Windows (XP o posterior) o Linux para los PC's.

Necesario tener instalado Java JRE 1.6.

Necesitará tener instalada una versión MySQL en el servidor para almacenar la información.

**Interfaces de comunicación:** El sistema se comunica con las bases de datos mediante MySQL. Existirá una base de datos de clientes y otra de habitaciones. También será necesario llevar el control de turnos.

Por último suponemos que se dispone de conexión a una LAN en todo momento.

### 3.2 Funciones

Los siguientes requisitos funcionales definen las acciones fundamentales que deben tener lugar en el software, aceptando y procesando las entradas y procesando y generando las salidas.

**Controles de validez para las entradas:** El sistema debe controlar la validez de cualquier dato introducido en el sistema.

- Usuario y clave
- Formato de fecha y hora
- Datos de habitaciones
- Identificador de personal

**Secuencia exacta de operaciones:** Respuesta ante situaciones anormales:

- Ante cualquier entrada en el sistema considerada no válida, se capturará el error y se informará convenientemente al usuario.
- Relaciones de salidas a las entradas:

Tanto los tipos de entrada como los tipos de salida serán caracteres alfanuméricos.



## CLIENTES

<b>Función</b>	<b>Dar de alta un cliente</b>
Prioridad	Alta
Estabilidad	Alta
Descripción	Se registra un cliente en el sistema.
Entrada	Nombre, Apellidos, DNI, Teléfono, T.Crédito, Tipo, Dirección.
Salida	ID del cliente.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema.
Acción	Registrar un nuevo cliente.
Precondición	El DNI no puede coincidir con ningún otro existente en la base de datos.
Postcondición	Usuario registrado en el sistema
Efectos laterales	N/A

<b>Función</b>	<b>Baja de un cliente</b>
Prioridad	Media
Estabilidad	Alta
Descripción	Se elimina un cliente del sistema
Entrada	ID del cliente.
Salida	Mensaje confirmando que el cliente ha sido eliminado.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Baja de cliente
Precondición	El cliente no debe tener pendiente ninguna reserva. El ID del cliente debe existir en la BD.
Postcondición	El cliente es eliminado del sistema
Efectos laterales	N/A



<b>Función</b>	<b>Modificación de datos de un cliente</b>
Prioridad	Baja
Estabilidad	Alta
Descripción	Se modifican los datos de un cliente registrado en el sistema
Entrada	Nombre, Apellidos, DNI, Teléfono, T.Crédito, Tipo, Dirección.
Salida	Cliente con los datos modificados
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Modificación de datos del cliente
Precondición	El ID del cliente debe existir en la base de datos.
Postcondición	Datos de clientes son modificados en el sistema
Efectos laterales	N/A

<b>Función</b>	<b>Listar clientes</b>
Prioridad	Muy baja
Estabilidad	Alta
Descripción	Se presentan los datos de los de los clientes dados de alta en el sistema
Entrada	N/A
Salida	Listado de clientes
Origen	Empleado
Destino	Empleado
Necesita	Acceso a base de datos del sistema
Acción	Consulta de historial de clientes.
Precondición	Ser empleado autorizado
Postcondición	N/A
Efectos laterales	N/A



## HABITACIONES

<b>Función</b>	<b>Dar de alta una habitación</b>
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Dar de alta una habitación en el hotel.
Entrada	Número de habitación, Precio, Número de camas.
Salida	ID de la habitación.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Añade habitación a la base de datos.
Precondición	El número de habitación no este en la base de datos.
Postcondición	Habitación dada de alta.
Efectos laterales	N/A

<b>Función</b>	<b>Dar de baja una habitación</b>
Prioridad	Media.
Estabilidad	Alta.
Descripción	Dar de baja una habitación en el hotel.
Entrada	ID de la habitación.
Salida	Mensaje confirmando que la habitación ha sido eliminada.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Borra habitación a la base de datos.
Precondición	El ID de la habitación este en la base de datos.
Postcondición	Habitación dada de baja
Efectos laterales	N/A



<b>Función</b>	<b>Modificación de la habitación</b>
Prioridad	Baja.
Estabilidad	Alta.
Descripción	Se modifican datos de una habitación del sistema.
Entrada	Número de habitación, Precio, Número de camas.
Salida	Habitación con los datos modificados.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Modificación de habitación.
Precondición	La habitación debe existir en la base de datos.
Postcondición	Datos de la habitación modificados en el sistema.
Efectos laterales	N/A

<b>Función</b>	<b>Listar habitaciones</b>
Prioridad	Media.
Estabilidad	Alta.
Descripción	Se presentan los datos de los de las habitaciones dadas de alta en el sistema.
Entrada	N/A
Salida	Listado habitaciones.
Origen	Empleado.
Destino	Empleado.
Necesita	Acceso a base de datos del sistema.
Acción	Consulta de historial.
Precondición	N/A
Postcondición	Servicios solicitados presentados por pantalla.
Efectos laterales	N/A





## EMPLEADOS

Función	Dar de alta empleado
Prioridad	Alta
Estabilidad	Alta
Descripción	Se registra un empleado nuevo en el sistema
Entrada	Nombre, Apellidos, DNI, Tlfn, Salario, Turno
Salida	ID del empleado.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Registrar empleado
Precondición	El DNI no puede coincidir con ningún otro existente en la base de datos.
Postcondición	Empleado registrado en el sistema
Efectos laterales	N/A

Función	Dar de baja un empleado
Prioridad	Media
Estabilidad	Alta
Descripción	Se elimina un empleado registrado en el sistema
Entrada	ID del empleado.
Salida	Mensaje confirmando la eliminación del empleado.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Baja de empleado
Precondición	El ID del empleado debe existir en la base de datos.
Postcondición	Empleado eliminado del sistema
Efectos laterales	Eliminar todos los datos almacenados en el sistema relacionados con este empleado



<b>Función</b>	<b>Modificación de datos del empleado</b>
Prioridad	Baja
Estabilidad	Alta
Descripción	Se modifican los datos de un empleado registrado en el sistema
Entrada	Nombre, Apellidos, DNI, Tlfn, Salario, Turno
Salida	Empleado con los datos modificados
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Modificación del cargo de un empleado
Precondición	El ID del empleado debe existir en la base de datos.
Postcondición	Datos del empleado modificados en el sistema
Efectos laterales	N/A

<b>Función</b>	<b>Listar empleados</b>
Prioridad	Muy baja
Estabilidad	Alta
Descripción	Lista los empleados existentes en la base de datos
Entrada	N/A
Salida	Empleados
Origen	Empleado
Destino	Empleado
Necesita	Acceso a la base de datos
Acción	Listar los empleados
Precondición	N/A
Postcondición	Lista de empleados
Efectos laterales	N/A



## RESERVAS

Función	Dar de alta una reserva
Prioridad	Alta
Estabilidad	Alta
Descripción	Se registra una nueva reserva en el sistema.
Entrada	ID de habitaciones, ID del cliente, Fecha de entrada, Fecha de salida, Fianza, Número de personas y servicios del hotel.
Salida	ID de la reserva.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Añadir reserva
Precondición	La reserva se puede realizar si se encuentra libre la habitación en las fechas indicadas.
Postcondición	Habitación reservada
Efectos laterales	N/A

Función	Dar de baja una reserva
Prioridad	Media
Estabilidad	Alta
Descripción	Se da de baja una reserva ya realizada del sistema.
Entrada	ID de la reserva.
Salida	Mensaje confirmando reserva dada de baja
Origen	Empleado
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Baja de reserva
Precondición	La reserva debe existir en la base de datos.
Postcondición	Reserva eliminada del sistema,dejando libre la habitación en la fecha indicada.
Efectos laterales	Eliminar todos los datos almacenados en el sistema relacionados con esta reserva.



<b>Función</b>	<b>Modificación de una reserva.</b>
Prioridad	Baja
Estabilidad	Alta
Descripción	Se modifican las condiciones contratadas en cierta reserva.
Entrada	ID de habitaciones, ID del cliente, Fecha de entrada, Fecha de salida, Fianza, Número de personas y servicios del hotel.
Salida	Reserva con condiciones modificadas
Origen	Empleado.
Destino	Sistema
Necesita	Acceso a base de datos del sistema
Acción	Modificación de las condiciones de la reserva.
Precondición	La reserva debe estar a nombre del cliente que hace el cambio de condiciones. No exista conflicto de fechas en la nueva reserva.
Postcondición	Condiciones modificadas en el sistema.
Efectos laterales	N/A

<b>Función</b>	<b>Listar reservas</b>
Prioridad	Muy baja
Estabilidad	Alta
Descripción	Lista las reservas existentes en la base de datos
Entrada	N/A
Salida	Reservas
Origen	Empleado
Destino	Empleado
Necesita	Acceso a la base de datos
Acción	Listar las reservas
Precondición	N/A
Postcondición	Lista de reservas
Efectos laterales	N/A



<b>Función</b>	<b>Buscar disponibilidad</b>
Prioridad	Muy baja
Estabilidad	Alta
Descripción	Busca habitaciones disponibles.
Entrada	Fecha_Entrada, Fecha_Salida.
Salida	Habitaciones disponibles.
Origen	Empleado
Destino	Empleado
Necesita	Acceso a la base de datos
Acción	Listar las habitaciones disponible.
Precondición	N/A
Postcondición	Lista de habitaciones.
Efectos laterales	N/A

<b>Función</b>	<b>Hacer efectiva reserva</b>
Prioridad	Muy baja
Estabilidad	Alta
Descripción	Confirma que el cliente ha ocupado la habitación.
Entrada	ID de la reserva.
Salida	Mensaje de confirmación.
Origen	Empleado
Destino	Sistema
Necesita	Acceso a la base de datos
Acción	Confirmar reserva cliente.
Precondición	N/A
Postcondición	Lista de reservas
Efectos laterales	N/A



## SERVICIOS DEL HOTEL

<b>Función</b>	<b>Dar de alta servicio del hotel.</b>
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Se da de alta un nuevo servicio en el hotel.
Entrada	Nombre del servicio.
Salida	ID del servicio creado.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Dar de alta un servicio del hotel.
Precondición	Que no haya un servicio igual.
Postcondición	El servicio se ha dado de alta.
Efectos laterales	N/A

<b>Función</b>	<b>Dar de baja un servicio del hotel.</b>
Prioridad	Media.
Estabilidad	Alta.
Descripción	Se da de baja un servicio existente del hotel en la base de datos.
Entrada	ID del servicio del hotel.
Salida	Mensaje confirmando servicio dado de baja.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Elimina servicio del hotel.
Precondición	Que no haya reservas con ese servicio contratado.
Postcondición	Servicio eliminado.
Efectos laterales	N/A



<b>Función</b>	<b>Modificar servicios del hotel.</b>
Prioridad	Baja.
Estabilidad	Alta.
Descripción	Se modifica el servicio.
Entrada	ID del servicio del hotel.
Salida	Servicio del hotel modificado.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a la base de datos del sistema.
Acción	Modifica el servicio.
Precondición	Que exista el servicio y que no haya reservas con ese servicio contratado.
Postcondición	Servicio modificado.
Efectos laterales	Que las reservas que tengan ese servicio cambien a dicho servicio.

<b>Función</b>	<b>Listar servicios del hotel.</b>
Prioridad	Muy baja.
Estabilidad	Alta.
Descripción	Lista los servicios del hotel existentes en la base de datos.
Entrada	N/A
Salida	Servicios del hotel.
Origen	Empleado.
Destino	Empleado.
Necesita	Acceso a base de datos del sistema.
Acción	Listar los servicios del hotel.
Precondición	N/A
Postcondición	Lista de servicios.
Efectos laterales	N/A



## TURNOS

<b>Función</b>	<b>Dar de alta turno.</b>
Prioridad	Alta.
Estabilidad	Alta.
Descripción	Se da de alta un nuevo turno.
Entrada	Nombre del turno, hora de entrada, hora de salida
Salida	ID del turno creado.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Dar de alta un turno.
Precondición	Que no haya un turno igual.
Postcondición	El turno se ha dado de alta.
Efectos laterales	N/A

<b>Función</b>	<b>Dar de baja un turno.</b>
Prioridad	Media.
Estabilidad	Alta.
Descripción	Se da de baja un turno existente en la base de datos.
Entrada	ID del turno del hotel.
Salida	Mensaje confirmando turno dado de baja.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a base de datos del sistema.
Acción	Elimina turno.
Precondición	Que no haya empleados con ese turno.
Postcondición	Turno eliminado.
Efectos laterales	N/A





<b>Función</b>	<b>Modificar turno.</b>
Prioridad	Baja.
Estabilidad	Alta.
Descripción	Se modifica el turno.
Entrada	ID del turno.
Salida	Turno modificado.
Origen	Empleado.
Destino	Sistema.
Necesita	Acceso a la base de datos del sistema.
Acción	Modifica el turno.
Precondición	Que exista el turno y que no haya empleados con ese turno.
Postcondición	Turno modificado.
Efectos laterales	N/A

<b>Función</b>	<b>Listar turnos.</b>
Prioridad	Muy baja.
Estabilidad	Alta.
Descripción	Lista los turnos existentes en la base de datos.
Entrada	N/A
Salida	Turnos.
Origen	Empleado.
Destino	Empleado.
Necesita	Acceso a base de datos del sistema.
Acción	Listar los turnos.
Precondición	N/A
Postcondición	Lista de turnos.
Efectos laterales	N/A



### 3.3 Requisitos de rendimiento

Siendo éste un servicio en el que el cliente debe recibir información en el momento acerca del estado de la habitación o habitaciones que ha solicitado, las transacciones de información en dichos casos debería ser prácticamente inmediata. Para ello podrían emplear técnicas de programación que supongan un aprovechamiento del tiempo de proceso.

Para otros casos de uso, como dar un listado de reservas, o cambios de turno entre empleados, no es totalmente necesario que la operación sea inmediata, pero sí es conveniente.

En caso de que la aplicación se quedara bloqueada, debería haber un backup o un servidor de repuesto para volverla a poner en funcionamiento cuanto antes.

### 3.4 Requisitos lógicos de la base de datos

**Tipos de información utilizada por diversas funciones:** Toda la información contenida en la base de datos serán números o caracteres.

**Frecuencia de uso:** El uso de la base de datos será constante ya que se accederá para cualquier consulta, alta, baja o modificación de un cliente, un empleado del hotel, como de los datos de una habitación o de una reserva.

**Capacidades de acceso:** Será posible que varias personas del personal del hotel autorizado soliciten un servicio al mismo tiempo por lo que el acceso simultáneo debe estar contemplado

**Requisitos de retención de datos:** Los datos se almacenarán en una base de datos (MySQL) en un disco duro.

#### Entidades de datos:

Clientes: Id\_cliente (Clave primaria), Nombre, Apellidos, DNI, Teléfono, TCrédito, Reservas, Dirección

Habitaciones: ID (Clave primaria), Precio, Numero de habitación, Numero de camas

Empleados: Id\_empleado (Clave primaria), Nombre, Apellidos, DNI, Teléfono, Turno, Salario



Reservas: ID (Clave primaria), ID\_Habitaciones, ID\_Cliente, Fecha\_entrada, Fecha\_Salida, Fianza, Servicios\_Hotel, Num\_Personas, Efectiva

Servicios\_Hotel: ID (Clave primaria), Servicio\_Hotel

Turnos: ID (Clave primaria), Hora\_Entrada, Hora\_Salida, Turno

#### Restricciones de integridad:

	Cliente	Habitacion	Empleado	Reserva	Turno	Servicio
Cliente	X	X	X	0..N	X	X
Habitacion	X	X	X	0..N	X	X
Empleado	X	X	X	X	1	X
Reserva	1	1..N	X	X	X	0..N
Turno	X	X	1..N	X	X	X
Servicio	X	X	X	0..N	X	X

### 3.5 Restricciones de diseño

Los patrones de diseño que usaremos serán:

- MVC ( Modelo - Vista - Controlador)
- Singleton
- Factory
- DAO
- Fachada

Una de las técnicas de programación que también usaremos será el polimorfismo y la herencia.

El proyecto se codificará en lenguaje Java utilizando la versión 1.6 de la máquina virtual con un soporte de base de datos en MySQL versión 5.5. El sistema operativo de desarrollo será Windows XP o superior, tanto de 32 como 64 bits. También se desarrollará en Linux Ubuntu.



### 3.6 Atributos del sistema software

**Fiabilidad:** Nuestro sistema va a tener unas características de software, que proporcionan una cierta calidad al producto.

**Disponibilidad:** Nuestro software será altamente fiable y será sometido a rigurosos test, siendo un sistema robusto.

**Seguridad:** El software encriptará toda la información confidencial en el protocolo SSL para que se encuentra a salvo de cualquier robo de datos. La conexión entre el cliente y el servidor será mediante una conexión cifrada y segura usando tecnología SSH.

**Mantenibilidad:** El software será desarrollado para que sea altamente ampliable y fácil de mantener.

**Portabilidad:** El programa será desarrollado en java de tal forma que es totalmente portable a cualquier sistema operativo.



## 4. Apéndices

**Anexo I:** Gestión de cambios.

Para consultar la gestión de cambios relacionada con la Especificación de requisitos consultar el fichero adjunto Gestión\_de\_cambios\_SRS.pdf