Contenido

[1. Visión 1](#_Toc166044862)

[1.1. Descripción del entorno de negocio 1](#_Toc166044863)

[1.1.1.Descripción de la empresa cliente 1](#_Toc166044864)

[1.1.2.Descripción del sistema informático actual 1](#_Toc166044865)

[1.2. Descripción de actores involucrados 1](#_Toc166044866)

[1.3. Descripción de la solución 2](#_Toc166044867)

[1.3.1. Características y beneficios 2](#_Toc166044868)

[1.3.2 Diagrama de contexto 2](#_Toc166044869)

# 1. Visión

# 1.1. Descripción del entorno de negocio

## 1.1.1.Descripción de la empresa cliente

<Descripción de la empresa u organización cliente (sedes, empleados y organización, idioma, volumen detallado de datos que maneja, perspectivas de futuro que puedan influir en el sistema a desarrollar)>

## 1.1.2.Descripción del sistema informático actual

<Descripción del sistema informático actual (hardware y software de base, descripción de las aplicaciones actuales relacionadas con el sistema a desarrollar, situación de los datos necesarios para el sistema a desarrollar)>

# 1.2. Descripción de actores involucrados

<Lista de actores relacionados con la aplicación (personas u otros sistemas), indicando para cada una su función en el negocio del cliente y su relación con la aplicación, es decir, si será un usuario del sistema, que funciones desarrollará, que información proporcionará y/o recibirá del sistema.>

* Actor 1
* Actor 2
* Actor n

# 1.3. Descripción de la solución

## 1.3.1. Características y beneficios

<Descripción detallada de la solución completa (arquitectura hardware y software) al problema planteado en el enunciado del proyecto, mediante una explicación no ambigua (que sea fácilmente entendible por el cliente) de esa posible solución (aplicación web, en intranet, cliente/servidor, de escritorio, móvil,…). Beneficios obtenidos por el negocio cliente y los futuros usuarios con la implantación de este sistema (como mejora la forma de trabajar de los actores descritos en 1.2)>

* Descripción de la solución
* Beneficios

## 1.3.2 Diagrama de contexto

<Representa el sistema a desarrollar como una caja negra y muestra las relaciones con el exterior: personas u otros sistemas. Un único diagrama utilizando Use Case Diagram de Modelio (los elementos Actor, Use case e Information Flow), en el que se ve el sistema como un único caso de uso y sus relaciones (información que dan y/o reciben) con los actores (personas o sistemas de información). En el diagrama se describe brevemente el flujo de información de cada relación (puede ser un nombre o una frase).>

Contenido

[2. Requisitos 1](#_Toc166045228)

[2.1. Requisitos no funcionales 1](#_Toc166045229)

[2.1.1. Seguridad 1](#_Toc166045230)

[2.1.2. Hardware y software 1](#_Toc166045231)

[2.1.3. Disponibilidad y rendimiento 1](#_Toc166045232)

[2.1.4. Usabilidad y accesibilidad 1](#_Toc166045233)

[2.1.5. Otros 2](#_Toc166045234)

# 2. Requisitos

<Es posible anidar requisitos identificándolos en secuencia, por ejemplo RF1.2 dependería de RF1 o RNF2.3.1 dependería de RNF2.3. Todos los requisitos deben ser precisos, deben poder probarse.>

# 2.1. Requisitos no funcionales

## 2.1.1. Seguridad

|  |  |
| --- | --- |
| **Identificativo** | **Descripción** |
| RNF1.1 |  |
| RNF1.2 |  |
|  |  |
|  |  |
| RNF1.n |  |

## 2.1.2. Hardware y software

|  |  |
| --- | --- |
| **Identificativo** | **Descripción** |
| RNF2.1 |  |
| RNF2.2 |  |
|  |  |
|  |  |
| RNF2.n |  |

## 2.1.3. Disponibilidad y rendimiento

|  |  |
| --- | --- |
| **Identificativo** | **Descripción** |
| RNF3.1 |  |
| RNF3.2 |  |
|  |  |
|  |  |
| RNF3.n |  |

## 2.1.4. Usabilidad y accesibilidad

|  |  |
| --- | --- |
| **Identificativo** | **Descripción** |
| RNF4.1 |  |
| RNF4.2 |  |
|  |  |
|  |  |
| RNF4.n |  |

## 2.1.5. Otros

<**Apartado opcional**, solo si es necesario incluir algún requisito no funcional que no corresponda con ninguna categoría anterior.>

|  |  |
| --- | --- |
| **Identificativo** | **Descripción** |
| RNF5.1 |  |
| RNF5.2 |  |
|  |  |
|  |  |
| RNF5.n |  |

Contenido

[3. Casos de uso 1](#_Toc136166583)

[3.1. Diagramas de caso de uso 1](#_Toc136166584)

[3.2. Lista de casos de uso 1](#_Toc136166585)

[CU1-Nombre de CU1 2](#_Toc136166586)

[CU2-Nombre de CU2 2](#_Toc136166587)

[CUn-Nombre de CUn 2](#_Toc136166588)

# 3. Casos de uso

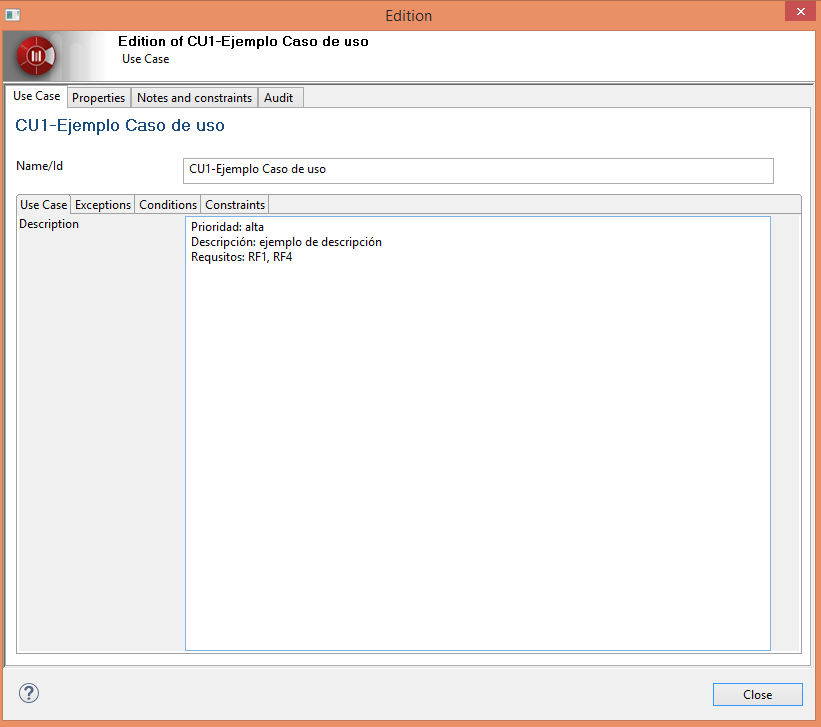
# 3.1. Diagramas de caso de uso

<Copia de las imágenes de los diagrama de casos de uso realizados en Modelio. uno por actor. Si existen casos de uso que se lanzan de forma programada se considerarán lanzados por el actor Sistema y, por tanto el actor Sistema tendrá su propio diagrama de casos de uso.>

# 3.2. Lista de casos de uso

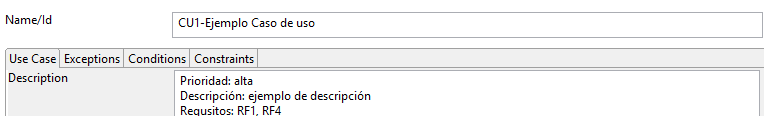
<Lista de los casos de uso que incluya identificativo, nombre, prioridad y, descripción. **Opcionalmente** se puede incluir la lista de requisitos cubiertos. Cada caso de uso se documenta en Modelio:

1. Nombre del caso de uso: ‘secuencial del tipo CU1, CU2,…’ ‘-‘ ‘nombre del caso’ (ejemplo CU1-Genera elemento)
2. Descripción del caso de uso, con el formato de la imagen siguiente (**la línea de Requisitos es opcional**). Copiar la imagen de la descripción del caso de uso de Modelio y pegar debajo del título del caso de uso

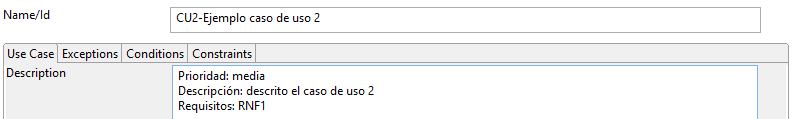


<Ejemplo:

CU1-Ejemplo Caso de uso



CU2-Ejemplo caso de uso 2



>

## CU1-Nombre de CU1

## CU2-Nombre de CU2

## CUn-Nombre de CUn

Contenido

[4. Modelo de dominio 1](#_Toc96080507)

[4.1. Diagrama Entidad/Relación 1](#_Toc96080508)

[4.2. Descripción de entidades y atributos 1](#_Toc96080509)

[4.3. Descripción de relaciones y cardinalidades 1](#_Toc96080510)

# 4. Modelo de dominio

<En el modelo de dominio no se está diseñando una BD relacional, se está definiendo el dominio de nuestro problema que se expresa mediante un diagrama E/R (modelo lógico) no con un diagrama de tablas (modelo físico)>

# 4.1. Diagrama Entidad/Relación

<Copia de la imagen del diagrama E/R realizado en Oracle Data Modeler. El modelo incluirá entidades, atributos de entidades y de relaciones n:m, tipo de dato y obligatorio o no de cada atributo, marca de identificativos únicos (claves) y relaciones con su cardinalidad máxima y mínima. Las relaciones podrán ser identificadas, no identificadas o de herencia. En el caso de las relaciones de herencia hay que indicar si son exclusivas o no y completas o no. La cardinalidad mínima se representa como una línea continua (1) o discontinua (0). Es importante que se distinga bien en la imagen si la línea es continua o discontinua. Se debe seguir el documento ORACLEDMUso >

# 4.2. Descripción de entidades y atributos

<Lista de entidades y atributos y sus descripciones. La descripción es necesaria cuando el nombre de la entidad o del atributo no explica de forma clara y sin ninguna duda su significado. Es obligatoria la descripción de los atributos que sean calculados o que tengan una lista de posibles valores, indicando, en la descripción, su cálculo y/o la lista de posibles valores y el significado de cada uno.

Se obtendrá de la herramienta Oracle Data Modeler un fichero pdf, de nombre 4\_Entidades, siguiendo las indicaciones del documento ORACLEDMUso. El fichero se considerará parte del artefacto 4 y siempre se entregarán juntos. En este apartado del documento no hay que escribir nada.>

# 4.3. Descripción de relaciones y cardinalidades

<Lista de relaciones, sus cardinalidades máximas y mínimas, su descripción y los atributos de la relación con su descripción, si los tuviera. La descripción es necesaria cuando el nombre de la relación no explica de forma clara y sin ninguna duda su significado. Es obligatoria la explicación de las cardinalidades máximas y/o mínimas cuando impliquen una decisión propia o una interpretación del enunciado porque en este no estén suficientemente definidas.

Se obtendrá de la herramienta Oracle Data Modeler un fichero html y se genera un fichero pdf de nombre 4\_Relaciones, siguiendo las indicaciones del documento ORACLEDMUso. El fichero se considerará parte del artefacto 4 y siempre se entregarán juntos. En este apartado del documento no hay que escribir nada.>

Contenido

[5. Modelado de casos de uso 1](#_Toc138932713)

[5.1. CU1-*Nombre CU1* 1](#_Toc138932714)

[Descripción y Frecuencia 1](#_Toc138932715)

[Precondiciones y Postcondiciones 2](#_Toc138932716)

[Escenario principal 2](#_Toc138932717)

[Escenarios excepcionales o alternativos 3](#_Toc138932718)

[Diagrama de actividad 3](#_Toc138932719)

[Descripción de la interfaz 3](#_Toc138932720)

[5.2. CU2-*Nombre CU2* 3](#_Toc138932721)

[Descripción y Frecuencia 3](#_Toc138932722)

[Precondiciones y Postcondiciones 4](#_Toc138932723)

[Escenario principal 4](#_Toc138932724)

[Escenarios excepcionales o alternativos 4](#_Toc138932725)

[Diagrama de actividad 4](#_Toc138932726)

[Descripción de la interfaz 4](#_Toc138932727)

[5.n. CUn-*Nombre CUn* 4](#_Toc138932728)

[Descripción y Frecuencia 4](#_Toc138932729)

[Precondiciones y Postcondiciones 4](#_Toc138932730)

[Escenario principal 4](#_Toc138932731)

[Escenarios excepcionales o alternativos 4](#_Toc138932732)

[Diagrama de actividad 4](#_Toc138932733)

[Descripción de la interfaz 4](#_Toc138932734)

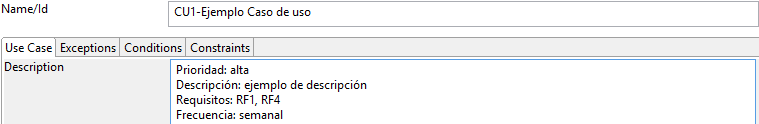
# 5. Modelado de casos de uso

<Un apartado para cada caso de uso a modelar.>

# 5.1. CU1-*Nombre CU1*

## Descripción y Frecuencia

<Añadir la frecuencia en la descripción del caso de uso de Modelio. La frecuencia es opcional, escribir solo si el caso de uso tiene una frecuencia definida, si no dejar en blanco. Copiar la imagen de la descripción del caso de uso de Modelio. Ejemplo:

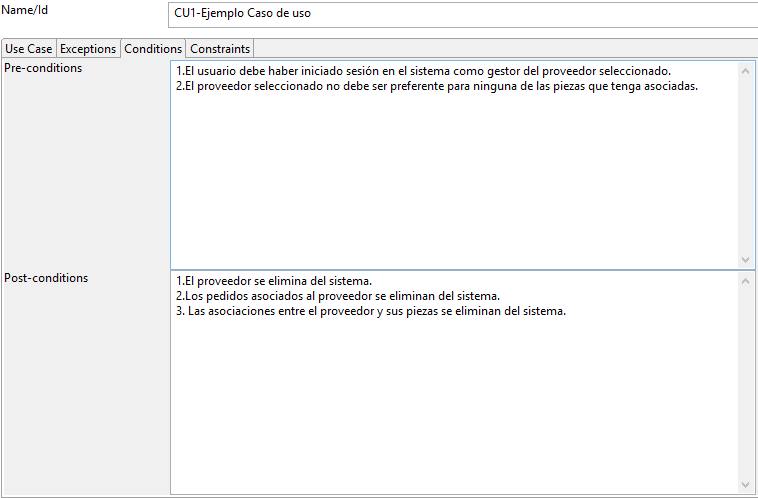
 >

## Precondiciones y Postcondiciones

<Las precondiciones deben estar relacionadas con el sistema. Son condiciones necesarias para poder realizar el caso de uso y que se refieren al rol y permisos del usuario conectado o que se han probado en un caso de uso que después llama a este con include o extend. Por ejemplo, como precondición del CU Realizar pedido manual, es incorrecto: ’hace falta comprar material’ ya que es algo que el sistema no comprueba, lo decide el usuario. Hay que distinguir entre precondiciones y escenarios alternativos. Por ejemplo, el CU Lista de piezas obsoletas llama al CU Eliminar pieza con extend cuando pulso sobre una pieza de la lista. Si incluyo en el CU Eliminar pieza la precondición ‘la pieza es obsoleta’ es correcto, pero si llamo directamente al CU Eliminar pieza sin pasar por el CU Lista de piezas obsoletas, sería incorrecto porque no se ha probado antes que la pieza es obsoleta, sería un escenario alternativo del CU Eliminar pieza.

En postcondiciones hay que indicar entidades, relaciones y atributos que se crean, borran o modifican, por ejemplo, es incorrecto ‘Los cambios realizados por el coordinador deben quedar registrados en el sistema’, ¿qué cambios? Otro ejemplo: es incorrecto ‘Se realiza un pedido básico de manera automática por parte del sistema’, sería correcto incluir tres postcondiciones: ‘Pedido creado con estado no enviado; Relación del pedido con el proveedor creada; Relaciones del pedido con las piezas seleccionadas creadas’. También se incluyen acciones que no se registran en la base de datos, por ejemplo: ‘Pedido impreso’ o ‘Mensaje de correo electrónico enviado al suministrador asociado al pedido’. Si un escenario alternativo produce postcondiciones se deben mostrar claramente en el escenario (no en postcondiciones), por ejemplo si un pedido no se puede borrar porque tenía ya alguna recepción asociada, se controla en un escenario alternativo y se indica en el escenario que el atributo inactivo del pedido se modifica con el valor ‘sí’.

Las precondiciones y postcondiciones se documentan en Modelio y se pega la imagen en este apartado. Ejemplo:

 >

## Escenario principal

<Los pasos de los escenarios de los casos de uso deben ser acciones concretas y precisas y las alternativas deben ir en escenarios alternativos. Por ejemplo, es incorrecto ‘2. Cuando el número de stock sea inferior a 3, se debe avisar al usuario para realizar pedidos extra.’, ¿a qué usuario?, ¿cómo se avisa?, sería correcto ‘2. Obtiene el valor de stock del material leído. 3. Envía un mensaje de correo electrónico al responsable del material. 4. Termina el proceso; Escenario alternativo: En el paso 2 si el stock es 3 o superior va al paso 4’. En este ejemplo stock y responsable son atributos de la entidad material. Otro ejemplo, es incorrecto ‘Escenario alternativo: La información proporcionada en el paso 3 está vacía si el técnico ya no trabaja para la empresa, pero aún no se ha dado de baja o si este no ha ido a trabajar’ ¿cómo sabe el sistema todo esto?, para que pudiera ser correcto en el modelo de dominio debería haber entidades/relaciones o atributos relacionados con la baja de técnicos y su fichaje cada día y el escenario se referiría a ellos.

Se incluirá una tabla del tipo (una columna Sistema, que representa toda la aplicación como una caja negra y una columna por cada actor que envía información al caso de uso):

|  |  |
| --- | --- |
| Actor | Sistema |
| 1.- Paso 1. | 2.-Paso 2. |
| 3.- Paso 3. | 4.- Paso 4. |
| ... | ... |

>

## Escenarios excepcionales o alternativos

<Los diagramas de caso de uso y el modelado de casos de uso deben ser coherentes, por ejemplo, si en el modelado del caso de uso Alta de petición dice ‘Paso 3: si el cliente no está registrado en el sistema se abrirá la interfaz de alta de clientes para poder realizar la petición.’ en el diagrama de casos de uso Alta de petición deberá estar unido con extend a Alta de cliente y en el modelo no se pondrá esta llamada en el paso 3, sino en un escenario alternativo: ‘Paso 3: si el cliente no está registrado <extend> (Alta de cliente) y vuelve al paso 3’. Ejemplos:

En paso N: si yyyyy, entonces xxxxx

En cualquier momento: si zzzzz, entonces wwwww

Se deben mostrar claramente las postcondiciones que produzcan.>

## Diagrama de actividad

<Copia de la imagen del diagrama hecho en Modelio. Debe incluir el escenario principal y los alternativos. En Modelio el diagrama debe crearse desde el caso de uso de forma que queden vinculados.>

## Descripción de la interfaz

<Lista e imagen de las pantallas que aparecen en el desarrollo del caso. Se puede utilizar cualquier herramienta para diseñar las pantallas, incluso un dibujo sobre papel escaneado. Describir pautas generales de diseño de la interfaz (teclas de función, menú, iconos, ayuda on-line, mensajes de error)>

# 5.2. CU2-*Nombre CU2*

## Descripción y Frecuencia

## Precondiciones y Postcondiciones

## Escenario principal

## Escenarios excepcionales o alternativos

## Diagrama de actividad

## Descripción de la interfaz

# 5.n. CUn-*Nombre CUn*

## Descripción y Frecuencia

## Precondiciones y Postcondiciones

## Escenario principal

## Escenarios excepcionales o alternativos

## Diagrama de actividad

## Descripción de la interfaz

Contenido

[6. Modelo de datos 1](#_Toc96082827)

[6.1. Diagramas de tablas 1](#_Toc96082828)

[6.1.1. Explicación del diagrama 1](#_Toc96082829)

[6.1.2. Script BD 1](#_Toc96082830)

[6.2. Otros elementos de almacenamiento 1](#_Toc96082831)

[6.2.1. Diseño y explicación elemento 1 2](#_Toc96082832)

[6.2.2. Diseño y explicación elemento 2 2](#_Toc96082833)

[6.2.n. Diseño y explicación elemento n 2](#_Toc96082834)

# 6. Modelo de datos

# 6.1. Diagramas de tablas

<Copia de la imagen del diagrama de tablas realizado en Oracle data Modeler. El diagrama se generará automáticamente a partir del diagrama E/R. Se toman decisiones no automáticas sobre: relaciones de herencia y relaciones 1:1. Para decidir la conversión de las relaciones de herencia, seguir las indicaciones del documento ORACLEDMUso. Para decidir la conversión de las relaciones 1:1 Oracle Data Modeler tiene en cuenta las cardinalidades mínimas, por lo que, si la conversión automática no es la deseada, eliminar la clave ajena directamente en el diagrama de tablas. Para generar el modelo físico (diagrama de tablas): se abre el modelo lógico (diagrama E/R) y se pulsa la opción Realizar Ingeniería a Modelo Relacional, dejando las opciones por defecto. Finalmente es posible desnormalizar por necesidades de diseño: rendimiento, seguridad,…>

## 6.1.1. Explicación del diagrama

<Explicación de los motivos que han llevado a tomar cada una de las decisiones de la conversión: relaciones de herencia , 1:1 y desnormalización.>

## 6.1.2. Script BD

<Se obtendrá tras la generación automática en Oracle Data Modeler del script de BD, un fichero pdf, de nombre 6\_ScriptBD. Seguid las indicaciones del documento ORACLEDBUso. El fichero se considerará parte del artefacto 6 y siempre se entregarán juntos. En este apartado del documento hay que escribir un comentario sobre el fichero generado, por ejemplo, sobre si existen warnings o errores o sobre los posibles valores de los campos discriminantes en las relaciones de herencia o sobre posible triggers generados.>

# 6.2. Otros elementos de almacenamiento

<Otros elementos archivados necesarios para la aplicación, por ejemplo ficheros de intercambio de información con otros sistemas o plantillas de correo electrónico.>

## 6.2.1. Diseño y explicación elemento 1

<Diseño del elemento, es decir formato detallado de cabeceras, líneas, etc. y una explicación textual de su uso.>

## 6.2.2. Diseño y explicación elemento 2

## 6.2.n. Diseño y explicación elemento n

Contenido

[7. Modelo de diseño 1](#_Toc166057203)

[7.1. Diagrama de clases de diseño 1](#_Toc166057204)

[7.2. Modelo de objetos de software y sus colaboraciones 1](#_Toc166057205)

[7.2.1. CU1-*Nombre CU1* 1](#_Toc166057206)

[**Diagrama de secuencia del sistema** 1](#_Toc166057207)

[**Diagrama de colaboración de diseño** 1](#_Toc166057208)

[7.2.2. CU2-Nombre CU2 2](#_Toc166057209)

[7.2.n. CUn-Nombre CUn 2](#_Toc166057210)

# 7. Modelo de diseño

# 7.1. Diagrama de clases de diseño

<Imagen del diagrama de clases realizado en Modelio. Se genera un diagrama de clases en Modelio basado en el diagrama de tablas de Toad, **solamente con las clases y sus atributos, métodos y relaciones necesarios para realizar los diagramas de secuencia del sistema (solo las clases de estereotipo interfaz) y de colaboración de diseño (resto de elementos del diagrama de clases)**. Se añaden la clase Fachada\_BD y los controladores e interfaces necesarios para el diseño de los casos de uso seleccionados. Según se van necesitando clases, atributos, métodos y relaciones en los diagramas de colaboración de diseño se van añadiendo en el diagrama de clases.>

# 7.2. Modelo de objetos de software y sus colaboraciones

<Se realizará un apartado para cada uno de los casos de uso.>

## 7.2.1. CU1-*Nombre CU1*

### **Diagrama de secuencia del sistema**

<Imagen del diagrama de secuencia en Modelio, que muestra un objeto Sistema, que representa toda la aplicación como una caja negra, un objeto por cada actor que envía información al caso de uso y los mensajes que envían al sistema para realizar el escenario principal del caso de uso. Los mensajes representan los pasos que realizan los actores en el escenario principal. En el diagrama se incluye un objeto Sistema. Los actores (que se han incluido en el diagrama de clases como clases de interfaz) se arrastran del repositorio de Modelio al diagrama como objetos. La información enviada por los actores se añade como mensajes desde el objeto actor al Sistema. El formato del mensaje será el de un método (u operación) que llevará, si es el caso, parámetros de entrada y de salida, y que se escribirá en el nombre del mensaje.>

### **Diagrama de colaboración de diseño**

<Imagen del diagrama de colaboración en Modelio, que representa el detalle de diseño del escenario principal del caso de uso. Se van arrastrando como objetos las clases de interfaz (actores), la clase controlador, la clase Fachada\_BD y las clases de dominio necesarias (si no existen ya, se crean antes en el diagrama de clases). Se incluyen los mensajes del diagrama de secuencia anterior entre la interfaz (o interfaces) y el controlador y los necesarios entre las clases controlador, de dominio y Fachada\_BD, para realizar la funcionalidad detallada del escenario, incluyendo los números de secuencia. Los mensajes necesarios se añaden primero como métodos (u operaciones) de la clase destino en el diagrama de clases. Después en el diagrama de colaboración se añaden como mensajes.>

* **Nombre del método 1**

<Solo para aquellos métodos en que sea necesaria una explicación. Los métodos de las clases suelen ser procesos básicos. Si el método no es obvio, es decir, selecciona información utilizando criterios, genera información combinando o elaborando otra o realiza cálculos, debe ser explicado mediante un diagrama de actividad y/o pseudocódigo, no solo textualmente.>

* **Nombre del método 2**
* **Nombre del método n**

## 7.2.2. CU2-Nombre CU2

## 7.2.n. CUn-Nombre CUn

Contenido

[8. Modelo de la arquitectura 1](#_Toc136251665)

[8.1. Diagrama de despliegue 1](#_Toc136251666)

[8.2. Diagrama de componentes 1](#_Toc136251667)

# 8. Modelo de la arquitectura

# 8.1. Diagrama de despliegue

<Imagen del diagrama de despliegue realizado en Modelio. Se muestra el hardware del sistema (servidores, clientes, impresoras, etc.) y sus comunicaciones (internet, red local, etc.) . Debajo de la imagen un párrafo que explique el diagrama.>

# 8.2. Diagrama de componentes

<Imagen del diagrama de componentes realizado en Modelio. Se mostrarán las clases representadas a través de paquetes, y sus fachadas si es el caso, las bases de datos, otros ficheros y las relaciones entre ellos. En el diagrama, para cada componente, se mostrará, mediante una nota, el nodo en el que se encuentra. Debajo de la imagen un párrafo que explique el diagrama.>

Contenido

[9. Plan de implantación 1](#_Toc24610225)

[9.1. Migración y/o carga inicial de datos 1](#_Toc24610226)

[9.2. Manual de explotación 1](#_Toc24610227)

[9.3. Manual de usuario 2](#_Toc24610228)

# 9. Plan de implantación

# 9.1. Migración y/o carga inicial de datos

<Lista de tablas de la base de datos y otros ficheros que hay que rellenar con datos, indicando el origen de los datos y si necesitan alguna transformación previa a la carga (lo que implicaría desarrollo de software específico para esta transformación). Los datos pueden proceder del sistema actual del cliente o ser datos fijos, como códigos postales o plantillas para mensajes preestablecidos de correo electrónico. La lista debe estar ordenada, por orden temporal de carga utilizando la integridad referencial de la base de datos.>

# 9.2. Manual de explotación

<El objetivo de este documento es recoger el conjunto de tareas que se deberán realizar para la correcta explotación del sistema. Se deberán indicar, entre otros aspectos, las tareas programadas (gestión de backups, mantenimiento de logs, etc.), procedimiento de paradas programadas, monitorización y gestión de la capacidad. La tabla muestra la información que puede ir en un manual de explotación **Se debe rellenar la columna Descripción en los Elementos que no están sombreados.**>

|  |  |
| --- | --- |
| **ELEMENTO** | **DESCRIPCIÓN** |
| **1. CALENDARIO DE OPERACIONES A REALIZAR.** | <Calendario resumen en el que se especifica en qué momento debe realizar cada una de las actividades que se describen a continuación.> |
| **2. DISPOSITIVOS DE ALMACENAMIENTO SECUNDARIO A UTILIZAR.** | <En este punto se le da información sobre los dispositivos de almacenamiento secundario que se deberán utilizar para hacer copias de seguridad o salvados de la base de datos u otra información del Sistema.> |
| **3. REALIZACIÓN DE COPIAS DE SEGURIDAD.** | <En este punto se le informa de cómo realizar, paso a paso, las Copias de Seguridad. También se le indican los momentos más adecuados para realizarlas y los requisitos que se deben cumplir para que éstas puedan llevarse a cabo y resulten útiles.> |
| **4. CLASIFICACIÓN Y ACCESO DE COPIAS DE SEGURIDAD.** | <Información para administrar las Copias de Seguridad de forma que si es necesario acudir a ellas sigan un orden cronológico y estén debidamente etiquetadas e identificadas.> |
| **5. MONITARIZACIÓN Y GESTIÓN DE LA CAPACIDAD.** | <Descripción detallada de aquellos recursos del Sistema que deben ser monitorizados. Se indicarán las necesidades de monitorización existentes para cada recurso, los umbrales esperados y las previsiones estimadas de crecimiento del Sistema en términos de almacenamiento, capacidad de procesamiento, tráfico de red y cualquier otro recurso.> |
| **6. EMISIÓN DE INFORMES A PETICIÓN.** | <Listado de posibles informes a petición y usuarios autorizados.> |
| **7. ESTABLECIMIENTO DE PUNTOS DE RESTAURACIÓN DEL SISTEMA.** | <Información para que el operario conozca los pasos a seguir para establecer puntos de restauración del sistema que permitan deshacer los cambios realizados en el Sistema desde la última vez que el equipo funcionaba correctamente.> |
| **8. SALVADO DE LA BASE DE DATOS.** | <Enumeración de las tareas a realizar y las normas que se deben cumplir para realizar el salvado de la base de datos.> |
| **9. TAREAS DE MANTENIMIENTO DE LOS EQUIPOS.** | <Enumeración de las tareas de mantenimiento y revisiones periódicas que los equipos precisan.> |
| **10. RESPONSABLES DIRECTOS DE CADA FUNCIONALIDAD DEL SISTEMA.** | <Listado de los responsables (y modos de contactar con ellos) de cada proceso que se debe lanzar y que le ha podido dar problemas. Aparece un responsable funcional (usuario de la propia empresa) y un responsable técnico (responsable de mantenimiento de la propia empresa o de la empresa con la que lo tenemos contratado)> |
| **11. ACTUACIONES ANTE SITUACIONES DE RIESGO.** | <Incluye aquellas actuaciones destinadas a restaurar el servicio ante situaciones anormales como caídas del Sistema, corte de suministro eléctrico, pérdida de datos, fallos de seguridad, etc. (planes de contingencia).  Todas aquellas personas relacionadas con el Sistema que se puedan ver afectadas por un arranque, rearranque o parada del Sistema, deben ser avisadas a través de una lista de correo.> |
| **12. ROTACIÓN DE LOGS. PERIODICIDAD.** | <Entre las tareas programadas del Sistema encontramos la rotación de Logs. En este apartado se describirá la política de rotación de logs y la periodicidad de esta rotación. Quedarán definidas las condiciones de rotación (tamaño de logs, intervalo de tiempo, etc.), el procedimiento de rotación y la política de almacenamiento de logs antiguos, así como cualquier otra información que pueda resultar de interés.> |

# 9.3. Manual de usuario

<Se incluirá el índice detallado completo y el desarrollo de los apartados necesarios para incluir las funciones indicadas en el documento Documento\_del\_Proyecto (El manual no será completo). El manual no debe solo mostrar las pantallas, debe indicar como usar la aplicación para realizar procedimientos completos, explicar los mensajes de error, resolver dudas sobre posibles valores o formato de campos de entrada, etc. En la documentación está disponible un ejemplo real de manual de usuario. Algunos errores comunes que hay que evitar:

-Incoherencia con el resto de artefactos, por ejemplo, decir en el manual que al iniciar sesión hay una opción de recuperar contraseña y que no exista un caso de uso ‘Recuperar contraseña’ que sea llamado con extend desde el caso de uso Iniciar sesión.

-No explicar las consecuencias de acciones importantes, como el borrado. Por ejemplo, en una gestión de empleados que tienen tareas asignadas, si se tiene que dar de baja un empleado: que pasa con las tareas asignadas o con aquellas tareas que ya ha terminado pero que siguen en la base de datos asociadas a ese empleado.

-Explicar las pantallas, sin explicar el proceso completo. Por ejemplo, en una pantalla en la que el empleado cierra una tarea: si un empleado debe cerrar una tarea como máximo 24 horas después de terminarla, que pasa si no lo hace. Otro ejemplo en esa misma pantalla: el texto “Si se usa el botón de ‘Cerrar tarea’, y el procedimiento falla, aparece este mensaje de error: ’La tarea no puede ser cerrada’” no es correcto ya que, el mensaje debería decir al usuario por qué no puede ser cerrada y el manual explicar en todos los posibles casos que debe hacer para resolver el problema que le impide cerrar la tarea.>

Contenido

[10. Aplicación de patrones 1](#_Toc24610238)

[10.1. Patrón de arquitectura Modelo-Vista-Controlador 1](#_Toc24610239)

[10.2. Patrones de diseño 1](#_Toc24610240)

[10.2.1. Nombre del patrón 1 1](#_Toc24610241)

[10.2.2. Nombre del patrón 2 1](#_Toc24610242)

[10.2.3. Nombre del patrón 3 1](#_Toc24610243)

# 10. Aplicación de patrones

# 10.1. Patrón de arquitectura Modelo-Vista-Controlador

<Explicación textual de como se ha utilizado el patrón e imagen del diagrama (clases y/o componentes) en el que se vea su aplicación.>

# 10.2. Patrones de diseño

<Para cada uno de los patrones de diseño elegidos, al menos tres, explicación textual de como se ha utilizado el patrón e imagen del diagrama (clases y/o colaboración) en el que se vea su aplicación. Se utilizarán patrones vistos en teoría y no se puede utilizar el patón Controlador ya que se usa en el patrón de arquitectura>

## 10.2.1. Nombre del patrón 1

## 10.2.2. Nombre del patrón 2

## 10.2.3. Nombre del patrón 3

Contenido

[11. Informe de gestión del proyecto 1](#_Toc85792891)

[11.1. Organización interna del proyecto 1](#_Toc85792892)

[11.2. Planificación y seguimiento del proyecto 1](#_Toc85792893)

[11.2.1. Informe de seguimiento del tiempo 1](#_Toc85792894)

[11.2.2. Informe de incidencias creadas recientemente 1](#_Toc85792895)

[11.2.3. Informe Diagrama de flujo acumulado 2](#_Toc85792896)

# 11. Informe de gestión del proyecto

# 11.1. Organización interna del proyecto

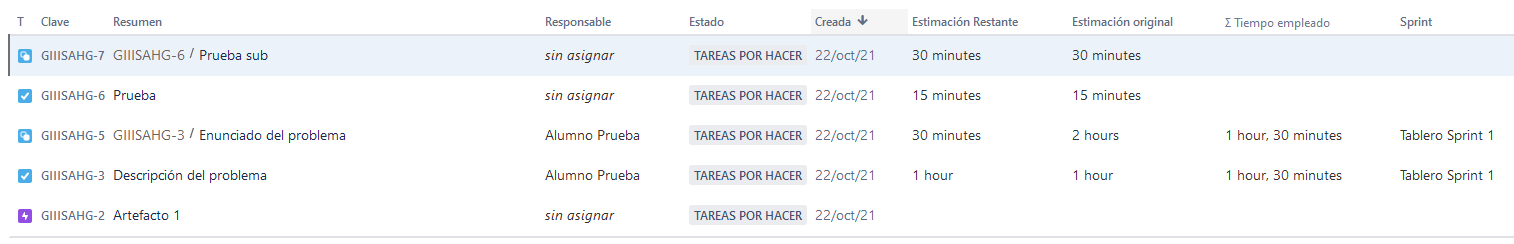
<Explicación textual de la gestión de las versiones de los entregables, de los ficheros de los proyecto de Rational y de TOAD y de los datos en Jira, especificando la forma de comunicación entre los miembros del equipo y la organización para el acceso y actualización de la información compartida.>

# 11.2. Planificación y seguimiento del proyecto

<Incluir los siguientes informes obtenidos de la herramienta JIRA. Se pueden añadir otros si se considera conveniente.>

## 11.2.1. Informe de seguimiento del tiempo

<En Filtros / Todas las incidencias, en Columnas marcar los campos que se muestran a continuación. Pantallazo de la información mostrada de todas las incidencias.

 >

## 11.2.2. Informe de incidencias creadas recientemente

<En la opción Paneles, Crear panel con el nombre del proyecto y permitir lectura y edición al proyecto. En el panel Añadir gadget y elegir Gráfico creado recientemente. Seleccionar el proyecto, con agrupación semanal y los días para cubrir todo el periodo lectivo. Pantallazo del gráfico y de la tabla de datos que se muestra al lado.>

## 11.2.3. Informe Diagrama de flujo acumulado

<Con columnas Por hacer, En progreso y Hecho y con periodo de tiempo: Iteración 1, desde el martes de la segunda semana del curso hasta la fecha de entrega de la iteración 1; Iteración 2, desde la fecha de entrega de la iteración 1 hasta la fecha de entrega de la iteración 2; Iteración 3, desde la fecha de entrega de la entrega de la iteración 2 hasta la fecha de entrega de la iteración 3.>