



**DISEÑO E IMPLEMENTACIÓN
DE UN
SISTEMA PDM**

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per
Francisco Expósito Aguilera
i dirigit per
Josep Maria Ganyet
Bellaterra, 11 de juny de 2007

El sotasingnat, Jose Maria Ganyet

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en Francisco Expósito Aguilera

I per tal que consti firma la present.

Signat: Josep Maria Ganyet

Bellaterra, 14 de juny de 2007

A Chelo

Índice general

1. Introducción	1
1.1. Presentación	1
1.2. Organización de la memoria	6
1.3. Motivo y justificación del proyecto	7
2. Fundamentos teóricos	8
3. Análisis de requerimientos	12
3.1. Requerimientos funcionales	13
3.2. Requerimientos no funcionales	16
4. Diseño e implementación	18
4.1. Patrón de arquitectura	18
4.1.1. Tipos de patrones	18
4.1.2. Patrón seleccionado para la aplicación	20
4.1.3. Herramienta de implementación del patrón MVC	22
4.2. Almacenamiento de la información	23
4.2.1. Almacenamiento en Oracle	25
4.2.2. Diseño del almacenamiento	30
4.3. Implementación de la aplicación	40
4.3.1. Modelo	40
4.3.2. Vista	43
4.3.3. Controlador	52
5. Pruebas	54

6. Coste económico	63
7. Conclusiones	66
A. Casos de uso	70
B. Implementación base de datos	121
Bibliografía	132

Índice de figuras

1.1. Participantes durante el ciclo de vida de un producto	3
3.1. Diagrama de caso de uso del menú Admin	13
3.2. Diagrama de caso de uso del menú Editar	14
3.3. Diagrama de caso de uso del menú Crear	14
3.4. Diagrama de caso de uso del menú Buscar	15
4.1. Patrón de diseño web: Modelo 1	19
4.2. Patrón de diseño web: Modelo 2 (MVC)	20
4.3. Capas del modelo MVC	22
4.4. Funcionamiento del modelo MVC	23
4.5. Almacenamiento en Oracle: tablespace	27
4.6. Almacenamiento en Oracle: datafile	27
4.7. Almacenamiento en Oracle: segmento	28
4.8. Almacenamiento en Oracle: bloque de datos	29
4.9. Acceso a la aplicación	44
4.10. Menú principal	45
4.11. Extensión del menú principal	45
4.12. Formulario de búsqueda de masterdatas	46
4.13. Formulario de creación de usuarios	47
4.14. Errores de base de datos	47
4.15. Búsqueda de ficheros con único resultado	48
4.16. Búsqueda de ficheros con varios resultados	48
4.17. Uso de uitags en interfaz gráfica	49
4.18. Creación de un fichero nuevo	50

4.19. Creación de la versión de un fichero	50
4.20. Descarga de ficheros	50
4.21. Tipos de descarga de asociaciones	51
5.1. Descarga de fichero en personal sin ser propietario	59
5.2. Descarga de fichero no en personal sin ser propietario	59
5.3. Descarga de asociación sin permisos en algún documento	60
5.4. Error en cambio de contraseña	61
5.5. Búsqueda de masterdata	61
5.6. Búsqueda de masterdata-b	62

Agradecimientos

Me gustaría dar las gracias a todas aquellas personas que en algún momento me han animado y ayudado a terminar este trabajo, especialmente

a Josep Maria Ganyet, profesor que me dio la oportunidad de realizar el Trabajo Final de Carrera en su departamento.

a mis compañeros y amigos de universidad.

a mi familia, y en especial a Chelo, por tener tanta paciencia.

Capítulo 1

Introducción

1.1. Presentación

En el entorno económico y empresarial actual, las organizaciones implicadas en la distribución de productos deben conseguir que éstos sean cualitativamente superiores a los productos de las empresas de la competencia y se comercialicen en un periodo de tiempo sensiblemente menor.

Los productos suelen atravesar varias etapas:

- Etapa de desarrollo de un nuevo producto:
 1. es muy caro
 2. no se perciben ingresos por venta
 3. es un periodo de pérdidas netas
- Etapa de introducción en el mercado
 1. supone un coste muy alto
 2. el nivel de ventas es bajo
 3. el balance es de pérdidas netas

- Etapa de crecimiento
 1. se reducen los costes debido a la realización de economías de escala
 2. los volúmenes de ventas aumentan significativamente
 3. se empiezan a percibir beneficios
- Etapa de madurez
 1. los costes son muy bajos
 2. se alcanzan los niveles máximos de ventas
 3. los precios tienden a caer debido a la proliferación de productos competitivos
 4. se alcanza la mayor rentabilidad
- Etapa de declive
 1. las ventas caen
 2. los precios bajan
 3. los beneficios se reducen

Además, las empresas necesitan administrar la comunicación y la información con sus clientes (Customer Relationship Management, CRM), con sus proveedores (Supply Chain Management, SCM) y de los recursos de la empresa (Enterprise Resource Planning, ERP).

Para mejorar el producto se debe gestionar su ciclo de vida (Product Lifecycle Management, PLM), de modo que se consiga:

- Acelerar el ciclo de vida del producto.
- Reducir el tiempo de comercialización.
- Garantizar el cumplimiento de los requisitos legales y de fabricación.
- Evitar retrasos en la publicación de productos.

- Reducir los gastos indirectos y la falta de eficacia derivados de los procesos descentralizados.
 - Aumentar la aceptación del producto y la satisfacción del cliente.

PLM es una estrategia empresarial que aplica un conjunto de soluciones informáticas para crear más colaborativamente la información de definición del producto, en aras a su mejor gestión y óptima diseminación en la empresa extendida, desde el concepto inicial hasta el final de la vida del producto, integrando las personas, los procesos, los sistemas y la información.

Esta estrategia empresarial no equivale a un único software, sino a la integración de una colección de herramientas de software y métodos de trabajo de manera que se gestione el proceso completo, como se puede ver en la figura 1.1. El principal objetivo de PLM es recopilar información que pueda ser reutilizada para otros proyectos y coordinar simultáneamente desarrollos de varios productos.

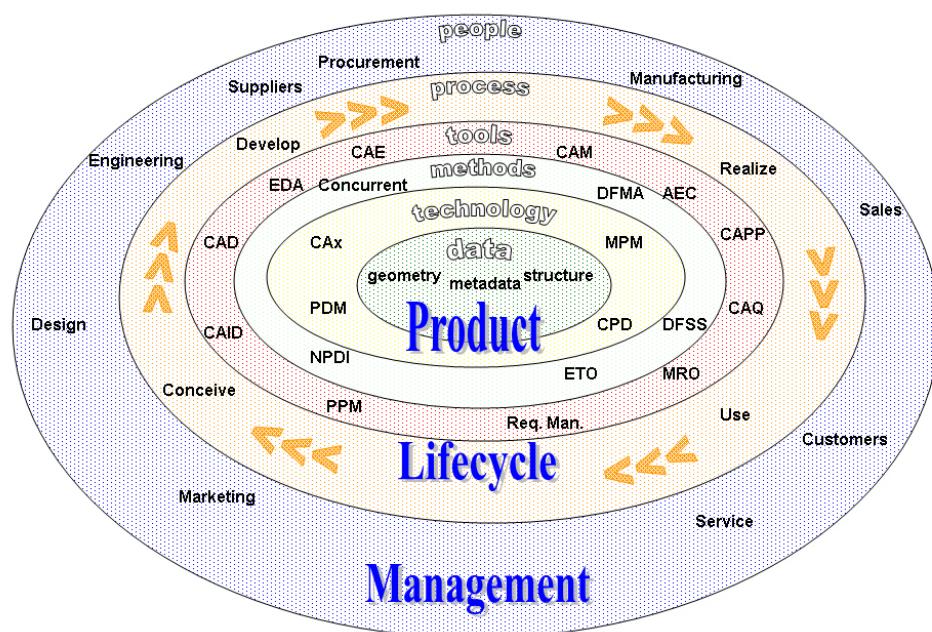


Figura 1.1: Participantes durante el ciclo de vida de un producto

Las fases del ciclo de vida de un producto suelen ser:

- Concepción: se basa en imaginar, especificar, planear e innovar. En esta etapa se definen los requisitos del producto basados en un cliente, compañía, mercado, etc.
- Diseño: se basa en describir, definir, desarrollar, testear, analizar y validar. En esta fase se empieza a diseñar y desarrollar el producto, progresando hasta un test del prototipo. En este punto se incluyen diferentes disciplinas, como ingenieros mecánicos, eléctricos, electrónicos o de software.
- Realización: se basa en fabricación, producción, venta y entrega. Aquí se incluye la creación física del producto, simulaciones para operaciones de moldeado, prensiones,... mediante el uso de CNC. Simultáneamente se genera la documentación de marketing y la configuración de la venta del producto.
- Servicio: se incluye el uso, mantenimiento, soporte, retirada paulatina, reciclaje y eliminación.

Ninguna de estas etapas está aislada del resto. De hecho un proyecto no sigue unos pasos secuenciales o independientes, sino que la información se mueve entre diferentes usuarios y sistemas. La mayor parte de PLM se ocupa de la coordinación y la gestión de la información: toda información debe ser gestionada. En el nivel del departamento de ingeniería se encarga el software PDM (Product Data Management), en el nivel corporativo el software EDM (Enterprise Data management), ... Habitualmente estas dos definiciones tienden a superponerse, aunque en general una organización tiene dos o más sistemas de gestión de la información. Estos sistemas están también enlazados con otros sistemas corporativos, como SCM, CRM y ERP, todos ellos mencionados anteriormente.

En definitiva, las cinco áreas de aplicación de PLM son:

- Gestión de necesidades de clientes (Customer Needs Management, CNM)
- Gestión de la cartera de productos (Product Portfolio Management, PPM)
- Diseño colaborativo de productos (Collaborative Product Design, CPD)
- Gestión de información de producto (Product Data Management, PDM)
- Fuente de materias primas directas (Direct Materials Sourcing, DMS)

Por su parte, PDM se utiliza para controlar la información relacionada con el producto: crea y gestiona relaciones entre conjuntos de datos que definen un producto y guarda sus relaciones en una base de datos. Se trata de una de las herramientas más importantes en la gestión del ciclo de vida de un producto. Algunas de las ventajas de PDM son:

- Proteger ficheros de proyecto con seguridad simplificada.
- Disminuir la pérdida de tiempo en la organización de los datos de diseño.
- Mejorar la productividad a través de la reutilización de información.
- Aumentar la colaboración.

Con este proyecto se pretende diseñar una aplicación web mediante la cual se pueda gestionar la información de un producto:

- Almacenamiento y recuperación de información.
- Manipulación del flujo del proceso de negocio mediante control de revisiones y estados.
- Gestión de la estructura de producto.

1.2. Organización de la memoria

De acuerdo con la finalidad del proyecto, el resto de la memoria se estructura de la siguiente manera:

2. Fundamentos teóricos: se efectuará una explicación detallada del funcionamiento teórico de la aplicación.

3. Análisis de requerimientos: en este capítulo se explicarán los requerimientos necesarios para el funcionamiento de la aplicación, mediante el uso de la notación UML.

4. Diseño e implementación: se describirá el diseño que se ha realizado, razonando el uso de la tecnología utilizada, así como la implementación de la aplicación.

5. Pruebas: se explicarán las diferentes pruebas que se han realizado para demostrar el correcto funcionamiento de la aplicación.

6. Coste económico: se realizará un coste aproximado del trabajo realizado en función de los perfiles necesarios en un proyecto.

7. Conclusiones: en este capítulo serán extraídas las conclusiones del trabajo realizado, además de plantear líneas de continuación futuras a partir del proyecto realizado.

Apéndices: en esta sección se colocará información secundaria pero importante que, debido a su extensión, es preferible extraerla de los apartados principales. En particular, se trata de los casos de uso necesitados y la implementación de la base de datos.

1.3. Motivo y justificación del proyecto

Existen dos motivos principales por los que se ha escogido este proyecto. El primero de ellos es la intención de realizar un proyecto que pueda tener alguna utilidad en el mundo empresarial en el que vivimos actualmente y, en este caso, que mejore la productividad y colaboración.

A pesar de ser el anterior un motivo más que suficiente para realizar un PFC, sin embargo, la principal razón para seleccionar este trabajo ha sido adquirir ciertos conocimientos en las tecnologías más utilizadas del momento, como son Java, Oracle y el framework Struts, de modo que no se considere el PFC como un mero trámite para obtener el título de ingeniería superior en informática, sino también obtener beneficios en forma de conocimientos técnicos que puedan suponer, por ejemplo, una mejora laboral o simplemente una ampliación de la información obtenida a lo largo de los estudios universitarios.

Capítulo 2

Fundamentos teóricos

El objetivo de esta aplicación consiste en ayudar, en especial, al departamento técnico de cualquier empresa, de modo que pueda mejorar la productividad y aumentar la colaboración, tanto de los usuarios del departamento técnico como de los proveedores, personal subcontratado, etc.

Normalmente, cada empresa trabaja en un número variado de proyectos en un mismo espacio de tiempo, por lo que los recursos deben ser distribuidos de la forma que se crea conveniente. Esto significa que los empleados trabajarán para uno o varios proyectos sin tener, por tanto, que recibir acceso a todos los proyectos en los que la empresa esté involucrada. Del mismo modo, puede haber proveedores o personal subcontratado que sólo pueda tener acceso a la información del producto con el que estén relacionados.

Además, aún teniendo acceso a un proyecto, cada usuario podrá realizar según qué operaciones en función de la etapa de desarrollo o de la propiedad del documento. En este caso la etapa de desarrollo viene definida por los estados del documento, que pueden ser:

- Personal: sólo el propietario puede ver el documento y realizar operaciones con él.

- Work: todos los usuarios que tengan permiso al proyecto al que esté asignado podrán descargar el documento, pero sólo el propietario podrá realizar operaciones con él. Este estado se utiliza en el proceso de desarrollo.
- Under review: todos los usuarios con permiso al proyecto pueden ver el documento pero nadie puede realizar operaciones con él. Este estado se utiliza para que los cambios queden en la historia permanentemente y que ningún usuario pueda modificarlo.
- Release: todos los usuarios con permisos en el proyecto pueden verlo, pero no se puede realizar ninguna operación sobre él. Este estado significa que está en producción.
- Support: todos los usuarios pueden ver el documento, pero nadie modificarlo. Este estado significa que este documento alguna vez ha estado en release y que ha sido sustituido por otro documento.
- Obsolete: es visible a todos los usuarios, pero quiere decir que el documento ya no se utiliza.

La colaboración en este tipo de organizaciones es muy importante para la finalización de un proyecto en los tiempos especificados, por lo que el trabajo en equipo es fundamental. Podemos poner como ejemplo una empresa del sector de la automoción que se dedique a diseñar vehículos. Obviamente, el diseño de un vehículo puede ser desglosado en varias subfunciones: chasis, motor, caja de cambio, etc; sin embargo, todos ellos forman un conjunto y cada una de estas subfunciones están formadas por más de una pieza. Esto significa que la aplicación debe permitir que se formen estructuras de información; es decir, una caja de cambio está formada por varias piezas:

- Cabeza, que a su vez contiene válvulas y rebajes
- Bloque de cilindros, formada por los pistones, bielas, etc.
- Cámara de combustión, formado por un pistón, un eje al cigüeñal, etc

- Sistema de bombeo
- Sistema de alimentación
- ...

Este hecho significa que la aplicación debe permitir que haya una relación de hasta n niveles (asociación) entre todos estos elementos del estilo:

1. Cabeza
 - a) válvula
 - b) rebaje
2. Bloque de cilindros
 - a) pistón
 - b) biela
3. Cámara de combustión
 - a) pistón
 - b) eje
 - c) tornillo
4. Sistema de combustión
 - a) ...
5. Sistema de alimentación
 - a) ...
6. ...

Además, el desarrollo de cada una de las piezas lleva un tiempo y rara vez la primera aproximación es la definitiva, por lo que la aplicación también debe permitir la creación de versiones de cada pieza o documento creado.

Otro de los procesos típicos durante el desarrollo de una pieza es el cambio de propietario. Muchas veces la persona que se iba a encargar del proceso de creación del documento no es la que lo termina, por lo que la aplicación debe permitir que se realicen cambios de propietario.

En definitiva, esta aplicación intenta proveer soluciones a los problemas típicos de los departamentos de I+D de cualquier empresa, aunque puede ser extrapolado a otros departamentos.

Capítulo 3

Análisis de requerimientos

En este capítulo se exponen los requerimientos de la aplicación, tanto funcionales como no funcionales.

- Requerimientos funcionales: son las acciones que debe realizar el software al recibir información, procesarla y producir los resultados. Es decir, en este punto se prepara el futuro comportamiento del software. En estas acciones se incluye:
 - Comprobación de validez de las entradas.
 - Secuencia de operaciones.
 - Generación de salidas.
 - Respuesta a situaciones anormales.

Para ello se ha utilizado el Lenguaje Unificado de Modelado (UML) que, aunque aún no es estándar, está apoyado en gran medida por OMG (Object Management Group). Debido a la extensión de este punto, sólo quedarán expuestos los diagramas de casos de uso necesarios, introduciendo en el Apéndice A los casos de uso y el resto de diagramas que han sido generados en esta parte del proceso de desarrollo del software.

- Requerimientos no funcionales: son aquéllos que indican necesidades no relacionadas directamente con lo que el software hace, sino cómo lo hace.

3.1. Requerimientos funcionales

Se ha generado un diagrama de caso de uso para cada uno de los menús principales de la aplicación, de modo que el diagrama sea claro y legible. En cada uno de estos diagramas de caso de uso se da por supuesto que el usuario se ha validado y que cualquier operación termina en una modificación en la base de datos.

El primer diagrama de caso de uso, mostrado en la figura 3.1, está relacionado con las operaciones que un administrador puede realizar (“Menú Admin”).

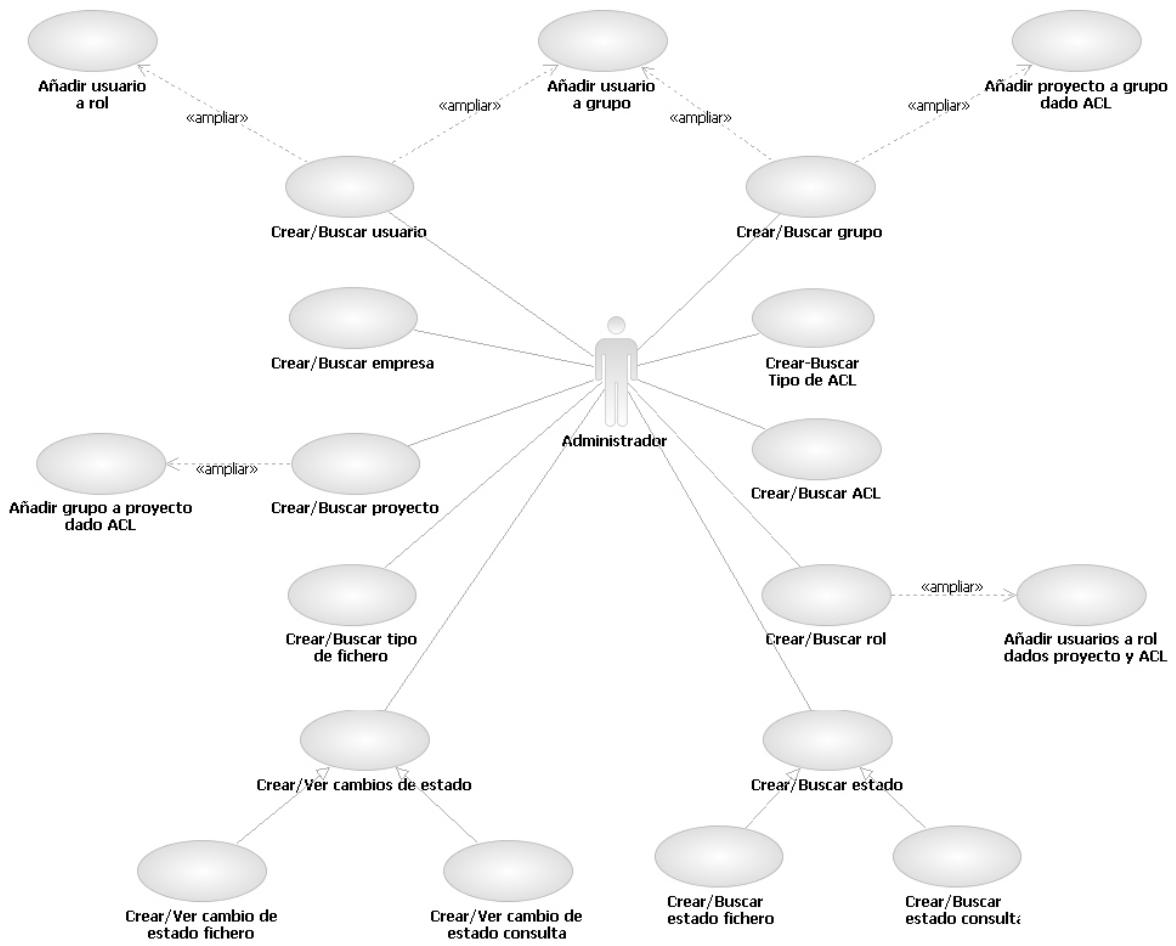


Figura 3.1: Diagrama de caso de uso del menú Admin

El segundo diagrama -figura 3.2- se refiere al menú “Editar”, mediante el cual los usuarios podrán modificar sus perfiles.

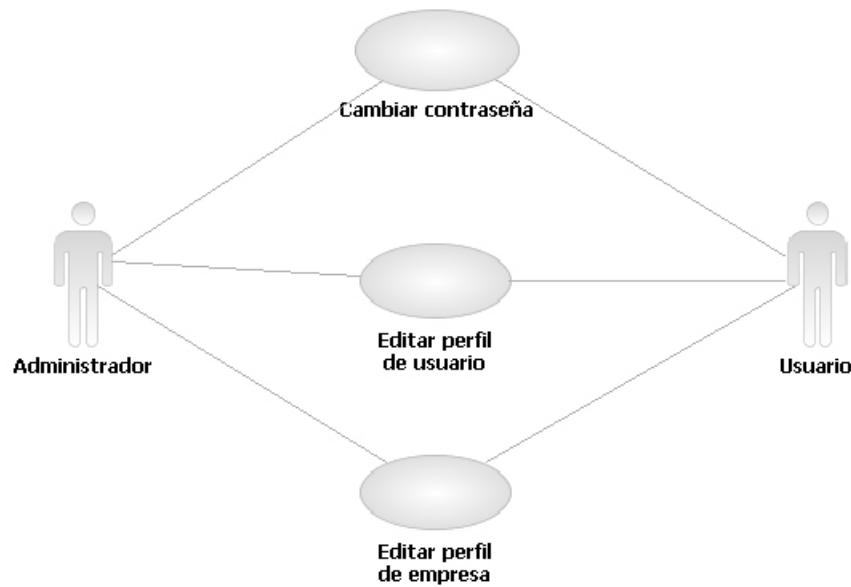


Figura 3.2: Diagrama de caso de uso del menú Editar

El tercer diagrama está relacionado con la creación de información por parte de los usuarios o administradores mediante la aplicación (“Menú Crear” en la figura 3.3).

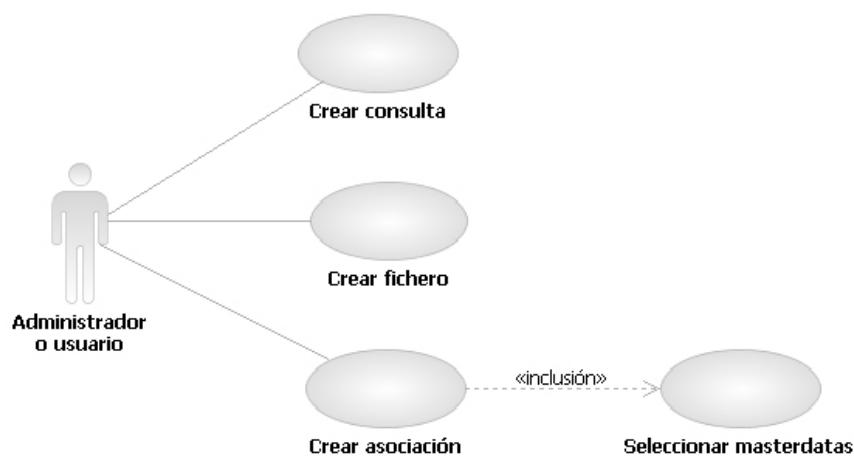


Figura 3.3: Diagrama de caso de uso del menú Crear

Por último está el diagrama en el que se muestran las operaciones posibles de los usuarios en la búsqueda de documentación -figura 3.4-. Este es el diagrama más extenso, ya que es el que se utilizará de forma más frecuente y el que provee mayor número de funciones (“Menú Buscar”).

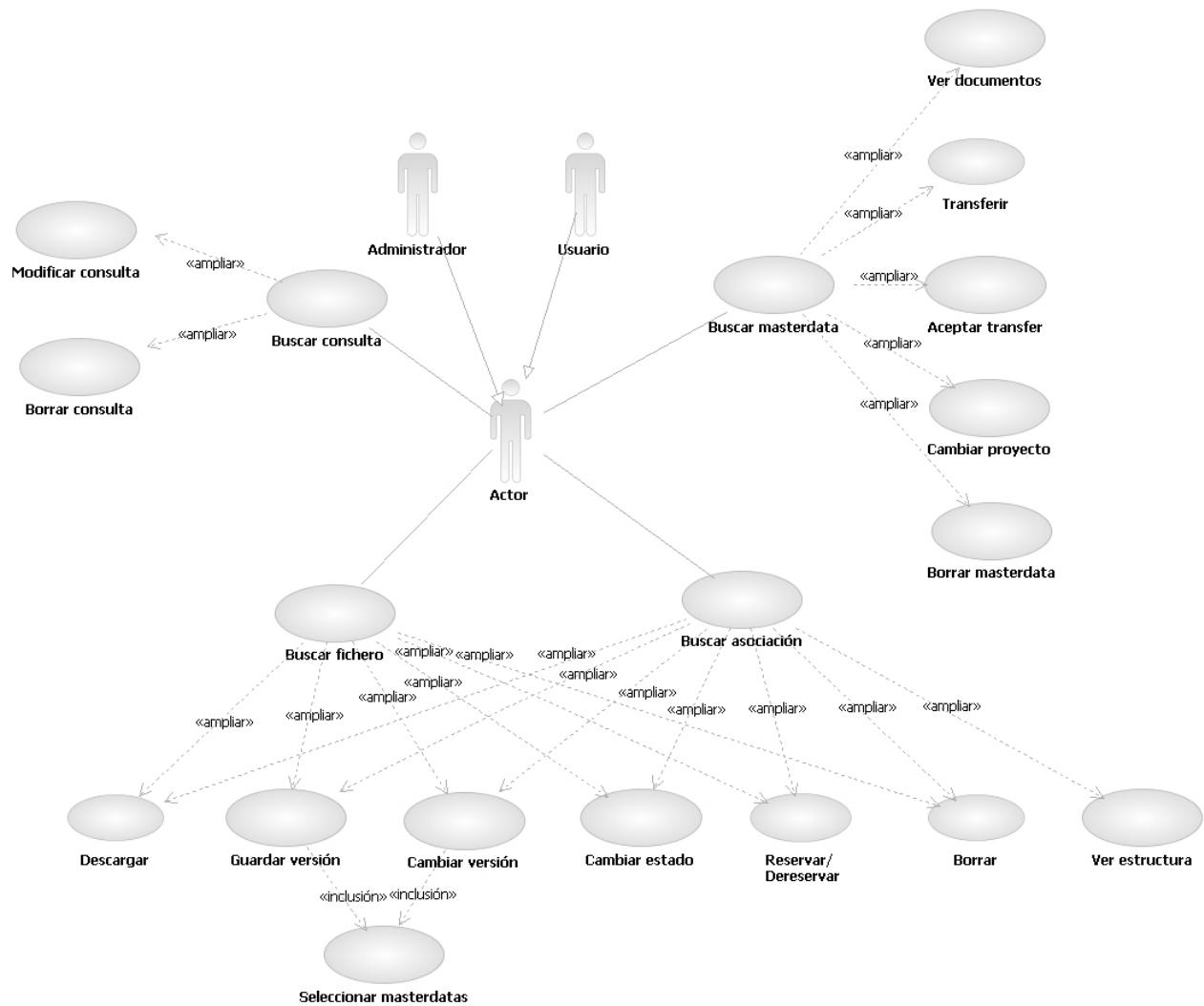


Figura 3.4: Diagrama de caso de uso del menú Buscar

Debido a su extensión, se han descrito los casos de uso que se han utilizado para implementar la aplicación en el apéndice A.

3.2. Requerimientos no funcionales

Rendimiento El sistema debe permitir el uso de alrededor de 100 usuarios, siendo este número bastante elevado para la suma del departamento de ingeniería más el departamento de SCM. En el caso de necesidad de uso de un mayor número de usuarios concurrentes, se podría optar por una solución ofrecida en el apartado de conclusiones. Puesto que se pueden guardar ficheros de hasta 2GB, el tiempo de transacción será variable y en este punto obtiene gran importancia el hardware escogido y el ancho de banda de la conexión. El tema de hardware queda exento de explicación en este proyecto, ya que de este punto se debe encargar el departamento de IT de la empresa en la que se vaya a implantar el sistema. Del rendimiento de la base de datos se encargará el administrador de ésta.

Seguridad Cada usuario deberá acceder a la aplicación mediante el uso de un login y una contraseña, los cuales tendrán unos permisos determinados y, en función de ellos, podrá realizar según qué operaciones. Esta parte será función del administrador de la base de datos. Respecto al servidor de ficheros, será función del departamento de sistemas, quienes definirán el acceso posible a los directorios sin tener que utilizar la aplicación. En principio estos permisos deben ser nulos excepto para los administradores pertenecientes al departamento de sistemas. En este apartado también debe ser remarcada la necesidad de realizar copias de seguridad, tanto en la parte del servidor de la base de datos como del servidor de ficheros. Puesto que se actualiza la información diariamente y habitualmente en cantidades de información extensas, se puede optar por realizar copias de seguridad incrementales cada día y realizar una copia completa cada semana, siempre en horario en el que no afecte a los usuarios.

Disponibilidad En el entorno empresarial actual es habitual que los servidores tengan una disponibilidad total, siendo únicamente posible su desconexión por tareas de mantenimiento. Por temas de disponibilidad sería importante clonar ambos servidores, por si el principal sufre algún problema.

Mantenibilidad El mantenimiento del sistema será función compartida entre el administrador de Oracle, el administrador de la aplicación y el departamento de sistemas. Será fundamental el mantenimiento de los discos físicos, debiéndose comprobar el espacio libre para poder incrementarlo cuando éste sea del 80%. El mantenimiento del código debe ser simplificado de la mejor manera posible, aunque se necesite mayor desarrollo de código a la hora de generar la aplicación.

Portabilidad El porcentaje de código dependiente del servidor debe ser mínimo, por lo que el traslado a otras plataformas debería ser sencillo. La única especificación técnica necesaria para la portabilidad de la aplicación sería que en el sistema de destino hubiera un gestor de base de datos Oracle -si se quiere evitar una migración de sistema gestor-. Puesto que Oracle está disponible en la mayoría de S.O., éste no debería ser un obstáculo para cambiar de plataforma.

Comunicaciones Debido al volumen de los ficheros a cargar y guardar, en la intranet sería importante una conexión mínima a 100Mb y para la conexión en redes externas una conexión mínima de ADSL, siendo ésta última opción insuficiente para cargar o bajar archivos de gran tamaño.

Capítulo 4

Diseño e implementación

4.1. Patrón de arquitectura

4.1.1. Tipos de patrones

En el diseño de aplicaciones web existen varios patrones comunes:

a) **Modelo 1:** presentación, negocio y persistencia están mezclados.

- Se utilizan páginas JSP-HTML para interactuar con el usuario (lógica de presentación).
- Las páginas JSP son las responsables de controlar el flujo de la aplicación: recibir peticiones, direccionar a la siguiente página (lógica de control).
- El acceso a los datos se lleva a cabo desde la misma página JSP o con JavaBeans.

En la figura 4.1. se puede ver el funcionamiento de este patrón de diseño.

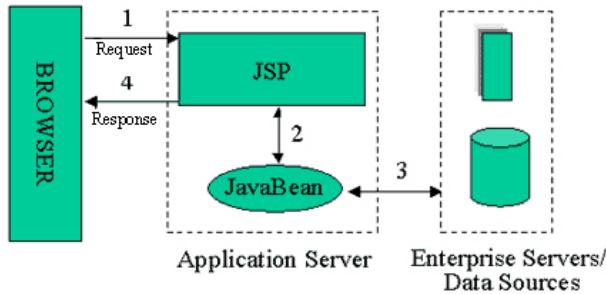


Figura 4.1: Patrón de diseño web: Modelo 1

La principal ventaja del este modelo es la simplicidad a la hora de programar y que permite al creador de las páginas generar contenido dinámico fácilmente, basándose en la petición y el estado de la aplicación. Sin embargo tiene varios inconvenientes, destacando el problema de mantenimiento cuando el programa crece, debido a la cantidad de scriptlets incrustados en las páginas JSP. Otro inconveniente es la reducción de la modularidad y de oportunidades de reutilización de código, además de que proporciona unas separaciones de papeles de desarrollo muy pobres.

- b) **Modelo 1.5:** se separa la presentación, por una parte, y la lógica de negocio y los datos por otro. Tiene los mismos inconvenientes que el Modelo 1, aunque con una importancia algo menor.

- c) **Modelo 2:** utiliza el patrón de diseño MVC para separar las capas de presentación, de negocio y de persistencia. Su funcionamiento aparece en la figura 4.2.
 - Las páginas JSP se utilizan para generar el nivel de presentación y los servlets para realizar las tareas que requieren procesamiento.
 - Además el controlador gestiona la navegación, decidiendo a qué página debe redireccionarse a continuación.

- No aparece lógica de procesamiento dentro de la presentación JSP: simplemente accede a los JavaBeans que previamente se han cargado para extraer dinámicamente su contenido.

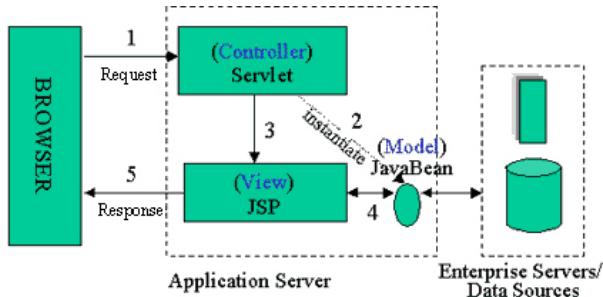


Figura 4.2: Patrón de diseño web: Modelo 2 (MVC)

- d) **Modelo 2X:** es una evolución del modelo 2 para construir aplicaciones multi canal.

4.1.2. Patrón seleccionado para la aplicación

En este proyecto se ha utilizado el Modelo 2, el cual permite que la aplicación sea totalmente escalable ya que independiza completamente las tres capas o componentes que forman todo desarrollo web:

- **Modelo:** en esta capa reside la lógica de negocio de la aplicación. Encapsula las reglas de negocio en componentes que son fáciles de probar, permiten mejorar la calidad del software y promueven la reutilización. Aquí se trata la comunicación a todos los niveles con la base de datos. Está formado por dos tipos de componentes:
 - *Componentes de estado:* define el conjunto actual de valores del modelo e incluye métodos para cambiar estos valores. La construcción independiente de estos componentes permite que las cualidades del diseño sean la reutilización, la calidad y la robustez. Normalmente se utilizan los JavaBeans

para implementar este componente.

- *Componentes de proceso:* definen los cambios permitidos para los estados en respuesta a los eventos.
- **Vista o interfaz de usuario:** constituye la lógica de presentación de una aplicación. Aquí es donde se unen los datos con el diseño elegido para que el usuario acceda a la información. Los componentes de la vista obtienen el estado actual del sistema del modelo y proporcionan el interfaz de usuario. Como parte de la generación del interfaz de usuario, la vista presenta los eventos que el usuario puede activar en cada momento. Generalmente se utiliza JSP.
- **Controlador o lógica de control:** es el que proporciona unión a toda la arquitectura, definiendo el workflow de la aplicación; es decir, el encargado de que se produzca la comunicación del usuario con la aplicación y responsable de:
 - Recibir eventos.
 - Determinar cual es el manejador apropiado.
 - Invocar a dicho manejador.
 - Determinar la generación de la respuesta apropiada.

Habitualmente se utilizan los servlets para la tecnología del controlador.

De este modo la lógica de negocio y el acceso a datos es completamente independiente de la presentación de la información y la interacción del usuario, introduciendo entre ambos la figura del controlador, como se puede ver en la figura 4.3, que muestra la independencia de las tres capas y las tareas que cada una de ellas debe realizar.

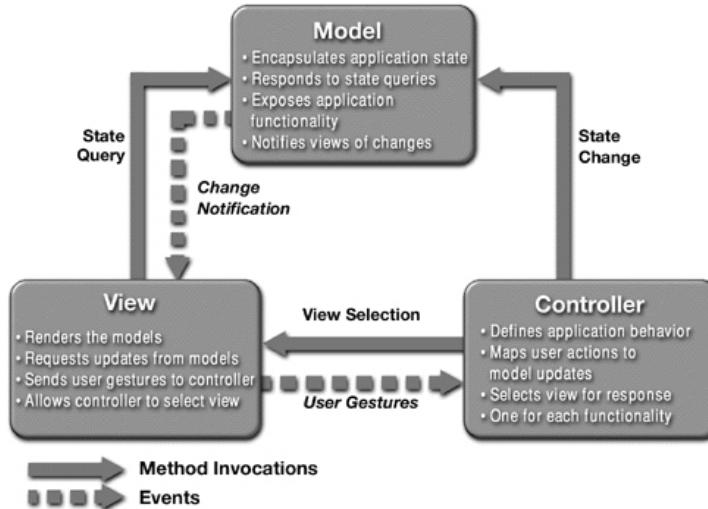


Figura 4.3: Capas del modelo MVC

Esta separación de capas produce dos ventajas importantes relacionadas con la modificación del software:

- Al realizar un cambio en una de las tres capas, sólo se modifica la capa en cuestión.
- Es posible modificar uno de los componentes sin conocer el funcionamiento del resto.

4.1.3. Herramienta de implementación del patrón MVC

Con la finalidad de implementar el patrón MVC en Java se ha optado por la utilización del framework Struts, el cual se desarrollaba como parte del proyecto Jakarta de la Apache Software Foundation, pero que actualmente es un proyecto independiente conocido como Apache Struts.

Struts está basado en tecnologías estándar como Java Servlets, JavaBeans, ResourceBundles y XML, así como de varios paquetes de Jakarta Commons, como BeanUtils. Además provee su propio controlador y se puede integrar con otras tecnologías para conseguir las capas de Modelo y Vista. Para el Modelo, este framework puede interactuar

con tecnologías estándar de acceso a datos como JDBC y EJB, así como paquetes de otros proveedores, como Hibernate e iBATIS. Para la Vista, el framework puede utilizar JavaServer Pages, incluyendo JSTL y JSF, ó Velocity, XSTL, etc.

Con el uso de este patrón, el funcionamiento de la aplicación -figura 4.4- sigue los siguientes pasos: el usuario interactúa con la interfaz de usuario, provocando que el navegador genere una solicitud que es atendida por el controlador (un servlet especializado). Éste se encarga de analizar la solicitud, seguir la configuración que se le ha programado en su XML y llamar al Action correspondiente pasándole los parámetros enviados. El Action instanciará y/o utilizará los objetos de negocio para concretar la tarea. Según el resultado que devuelva el Action, el Controller derivará la generación de interfaz a una o más JSPs, las cuáles podrán consultar los objetos del Model a fines de realizar su tarea.

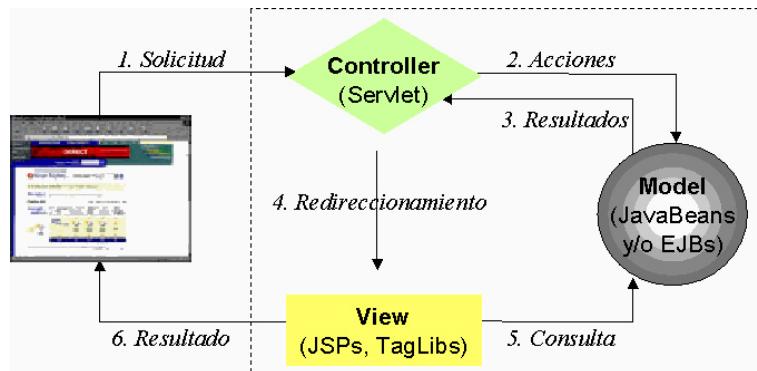


Figura 4.4: Funcionamiento del modelo MVC

4.2. Almacenamiento de la información

En este sistema se debe almacenar información de dos tipos: por una parte los ficheros físicos que el usuario quiere guardar y, por otra parte, la información que define todos los campos relacionados con los ficheros que, de algún u otro modo, sean necesarios en algún momento para poder trabajar con dichos ficheros. Además también debe almacenar esta última parte todo lo relacionado con la seguridad de la aplicación.

Puesto que el tamaño de los ficheros puede tener una variabilidad alta, se ha optado por no almacenar los ficheros en la base de datos, sinó que se utilizarán dos tipos de servidores:

- a) Servidor de base de datos: guarda y gestiona la información lógica, entendiéndose por información lógica aquella que es necesaria para trabajar con los ficheros y tratar la seguridad de la aplicación.
- b) Servidor de ficheros: guarda y gestiona los ficheros físicos.

En nuestro caso nos ocuparemos del servidor de base de datos, mientras que la configuración del servidor de ficheros se deja como trabajo para el equipo de infraestructura.

Con la finalidad de organizar, administrar, usar, compartir y proteger la información se ha decidido utilizar sistema gestor de base de datos relacional. Existen varios sistemas de este tipo (Microsoft SQL Server, Oracle, DB2, PostgreSQL, MySQL, FireBird).

Las principales ventajas del uso de bases de datos son:

- Globalización de la información: permite a los diferentes usuarios considerar la información como un recurso corporativo.
- Eliminación de información redundante: información duplicada que puede generar inconsistencias en la base de datos.
- Eliminación de información inconsistente.
- Permite compartir información: varios sistemas o usuarios pueden utilizar una misma entidad.
- Permite mantener la integridad en la información: sólo se almacena la información correcta.

- Independencia de datos y tratamiento: la independencia de datos implica un divorcio entre programas y datos; es decir, se pueden hacer cambios a la información que contiene la base de datos o tener acceso a ésta de diferente manera, sin hacer cambios en las aplicaciones o en los programas. Esto implica menor coste de mantenimiento.
- Restricciones de seguridad: en lo relacionado con el acceso de usuarios a los datos y operaciones sobre los datos.

Se ha optado por escoger Oracle como sistema gestor de base de datos relacional (RDBMS). Es robusto, tiene muchas características que nos garantizan la seguridad e integridad de los datos; que las transacciones se ejecuten de forma correcta, sin causar inconsistencias; ayuda a administrar y almacenar grandes volúmenes de datos; estabilidad, escalabilidad y es multiplataforma. De hecho, es el dominante en el mercado de servidores empresariales.

4.2.1. Almacenamiento en Oracle

En el almacenamiento de Oracle hay cinco entidades a tener en cuenta:

- a) **Tablespaces o espacios de tablas:** un tablespace es una unidad lógica de almacenamiento de datos. No se trata de un fichero físico, sino de un conjunto de propiedades de almacenamiento que se aplican a los objetos (tablas, secuencias,...) que se van a crear en la base de datos bajo el tablespace indicado. Su estructura puede verse en la figura 4.5.

Las propiedades que se asocian a un tablespace son:

- Localización de los ficheros de datos.
- Especificación de máximas cuotas de consumo de disco

- Control de la disponibilidad de los datos (en línea o fuera de línea).
- Backup de datos

Cuando un objeto se crea dentro de un tablespace, este objeto adquiere todas las propiedades descritas en él.

Un tablespace imprescindible es SYSTEM, el cual se crea automáticamente al crearse la base de datos. Es obligatorio que esté siempre on-line ya que contiene la siguiente información:

- El diccionario de la base de datos
- Los segmentos de rollback temporales.
- Aquellas tablas que el producto ORACLE utilice para su funcionamiento.

El uso de múltiples tablespaces permite mayor flexibilidad en operaciones con la base de datos, como por ejemplo:

- Separar datos de usuario de datos del diccionario para reducir el contenido de los objetos de diccionario y objetos de esquema para los mismos archivos.
- Guardar diferentes archivos de datos en discos separados para reducir lectura de entrada/salida.
- Realizar backup de espacios de tablas individuales.

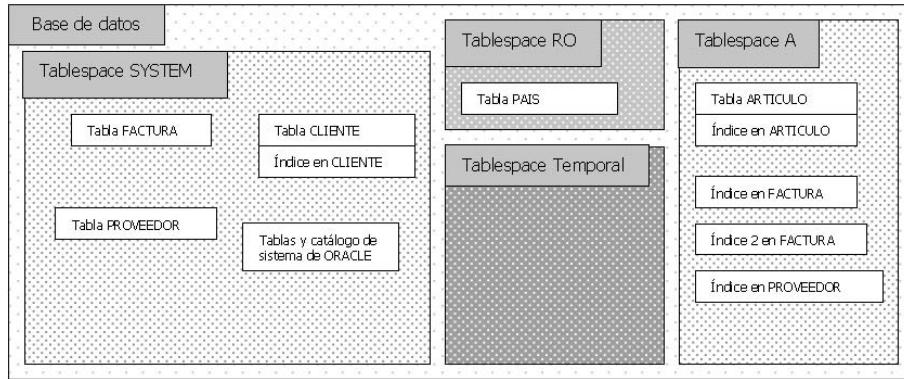


Figura 4.5: Almacenamiento en Oracle: tablespace

- b) **Fichero de datos o datafile:** es la representación física de un tablespace -ver figura 4.6-; es decir, dónde se almacena la información físicamente. Un datafile tiene un tamaño predefinido en su creación, aunque puede ser alterado en cualquier momento. Al crearlo, éste ocupará tanto espacio en disco como se haya indicado en su creación, aunque internamente está completamente vacío. De este modo se reserva espacio continuo en disco y se evita la fragmentación. Los datafiles tienen una propiedad llamada autoextend que, si está activa, se encarga de que el datafile crezca automáticamente (según un tamaño indicado) cada vez que se necesite espacio y no exista.

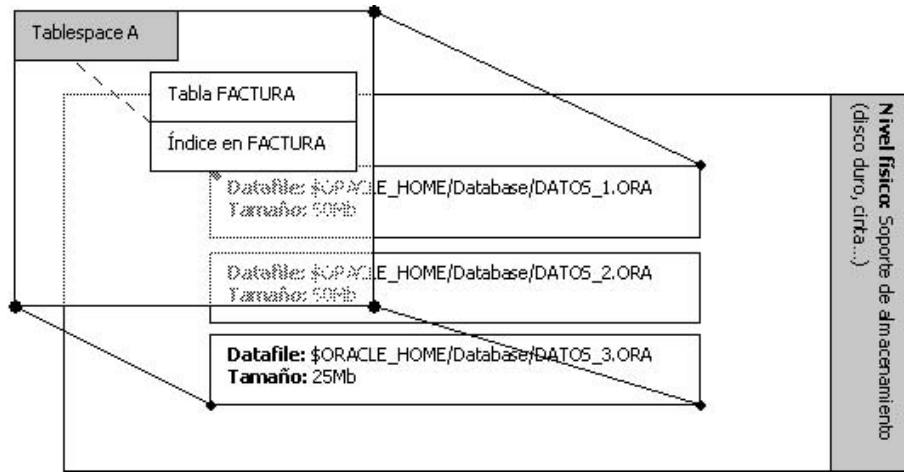


Figura 4.6: Almacenamiento en Oracle: datafile

c) **Segmento:** Un segmento es el espacio reservado por la base de datos, dentro de un datafile, para ser utilizado por un solo objeto. Físicamente, cualquier objeto de la base de datos es un segmento dentro de un datafile. En la figura 4.7 se relacionan los conceptos anteriores con el concepto de segmento.

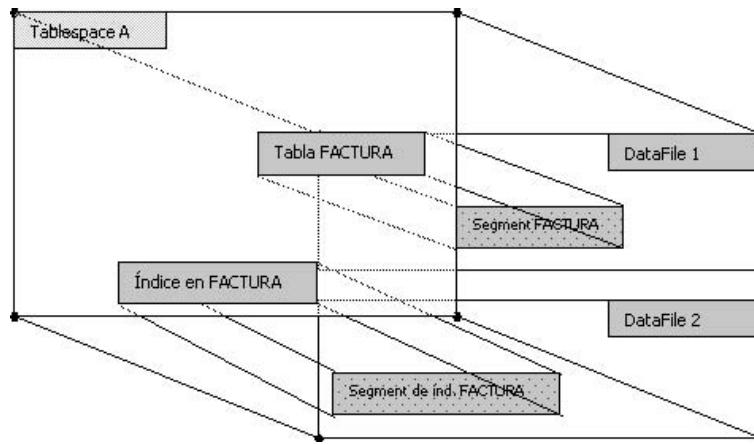


Figura 4.7: Almacenamiento en Oracle: segmento

d) **Extensión:** es un espacio en disco que se reserva de una sola vez, un segmento que se reserva en un momento determinado de tiempo. El concepto de extent es un concepto físico, unos están separados de otros dentro del disco. Cada segmento se compone de varias extensiones y puede ser reservado de una vez (10Mb de golpe) o en varias veces (5Mb hoy y 5Mb mañana). Cada una de las veces que se reserva espacio se denomina extensión.

Las extensiones se crean en momentos distintos de tiempo, por lo que pueden estar fragmentadas de otras. Un objeto de la base de datos no reside todo junto dentro del bloque, sino que residirá en tantos bloques como extensiones tenga. Por este motivo es crítico definir un buen tamaño de extensión inicial ya que, si es lo suficientemente grande, el objeto nunca estará fragmentado y si es demasiado grande, se pierde gran cantidad de espacio. Si el objeto tiene muchas extensiones y están separadas en disco, las consultas pueden retardarse considerablemente, ya que las cabezas lectoras tienen que dar saltos continuamente.

El tamaño de las extensiones se define durante la creación del objeto y no puede ser modificado después de la creación. Oracle recomienda que el tamaño del INITIAL EXTENT sea igual al tamaño del NEXT EXTENT.

La mejor solución es calcular el tamaño que tendrá el objeto (tabla o índice), multiplicando el tamaño de cada fila por una estimación del número de filas. Cuando hemos hecho este cálculo, debemos utilizar este tamaño como extensión INITIAL y NEXT, y tendremos prácticamente la certeza de que no se va a producir fragmentación en ese objeto. En caso de detectar más de 10 extensiones en un objeto Oracle recomienda recrear el objeto desde cero (aplicando el cálculo anterior) e importar de nuevo los datos.

- e) **Bloque de datos:** Es un concepto físico ya que representa la mínima unidad de almacenamiento que es capaz de manejar Oracle. Cada segmento (o cada extensión) se almacena en uno o varios bloques de datos -ver figura 4.8-, dependiendo del tamaño definido para la extensión y el bloque de datos.

El tamaño de un bloque de datos tiene que ser múltiplo del tamaño de una unidad de asignación del sistema operativo.

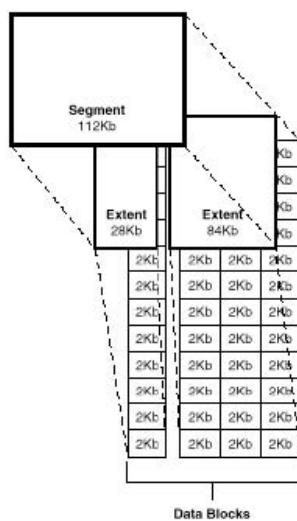


Figura 4.8: Almacenamiento en Oracle: bloque de datos

4.2.2. Diseño del almacenamiento

A partir del análisis de requerimientos y de la selección de Oracle, se ha diseñado una base de datos para almacenar los datos que proporciona la aplicación, de modo que se distribuya la información de un modo lógico, eficiente y rápidamente accesible. Para ello se ha hecho uso de:

- **Tablespaces y datafiles:** Algunos parámetros que se han utilizado en la creación de los espacios de tablas son:
 - *LOGS:*
 - ◊ LOGGING: se usa para indicar si las operaciones que van a realizarse sobre los objetos del tablespace deben registrarse en los ficheros de redo log (ficheros de almacenamiento de cambios en la BD).
 - ◊ FORCE LOGGING: fuerza a que todos los objetos se registren en los redo log incluso aunque al crearlos se les haya dicho lo contrario.
 - ◊ NOLOGGING: no se generan undo ni redo logs para las operaciones que soporten la opción NOLOGGING en tablas, índices y particiones del tablespace.
 - *AUTOEXTEND:* habilita (ON) o no (OFF) la extensión automática del fichero de datos.
 - *NEXT:* determina el espacio en disco que será asignado a un fichero de datos cuando sea necesaria una nueva extensión.
 - *MAXSIZE:* determina el máximo espacio que podrá ser asignado a un fichero de datos.

- *EXTENT MANAGEMENT*: es la forma de manejar las extensiones:
 - ◊ DICTIONARY: las extensiones son gestionadas en tablas de data dictionary de forma que, cuando un objeto necesita otra extensión, debe preguntar al sistema. Oracle entonces va a las tablas de data dictionary, ejecuta algunas queries, encuentra el espacio (o no) y actualiza una fila en una tabla (o la elimina) e inserta una fila en otra.
 - ◊ LOCAL: las extensiones son gestionadas mediante un bitmap en el propio tablespace. Para obtener una extensión, sólo se necesita poner un bit a 1 en el bitmap y para liberar espacio ponerlo a 0. En comparación es increíblemente más rápido.
- *SEGMENT SPACE MANAGEMENT*: es la forma de manejar el espacio libre y usado:
 - ◊ MANUAL: el espacio libre en los segmentos se gestiona usando listas libres (listas de bloques de datos que tienen espacio disponible para insertar filas). Es necesario especificar varios parámetros de almacenamiento (PCTUSED, PCTFREE y FREELISTS GROUPS) al crear objetos en ese tablespace.
 - ◊ AUTO: el espacio libre se gestiona mediante bitmaps. En este caso, cada bitmap describe el estado de cada bloque de datos con respecto a la cantidad de espacio disponible en el bloque. Es una forma más simple y eficiente de gestionar el espacio de los segmentos y elimina la necesidad de especificar los parámetros de almacenamiento anteriores. Está comprobado que, para cargas de trabajo estándar, el rendimiento de la aplicación es superior si se gestiona automáticamente.

Una vez explicados los parámetros anteriores, se ha optado por seleccionar las opciones Extent Management Local y Segment Space Management Auto ya que mejoran el rendimiento del sistema, facilita las tareas de administración y reducen la fragmentación. Además, el hecho de habilitar varios tablespaces

es beneficioso si se quieren mover tablas específicas a discos físicos diferentes o si se quieren realizar backups de tablas individuales sin interrumpir el uso del resto. La otra opción que se ha implementado es colocar los tablespaces en varios discos físicos, de forma que se mejore la E/S. La configuración que se ha propuesto es la siguiente:

- *Tablespace para las tablas empresa y usuario*

```
CREATE TABLESPACE "TAB_USUARIO_EMPRESA"  
LOGGING  
DATAFILE '/nfs/pdm/info_a/TAB_USUARIO_EMPRESA.ora'  
SIZE 5M AUTOEXTEND  
ON NEXT 1M MAXSIZE 50M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para la tabla fichero*

```
CREATE TABLESPACE "TAB_FICHERO"  
LOGGING  
DATAFILE "/nfs/pdm/info_b/TAB_FICHERO.ora"  
SIZE 20M AUTOEXTEND  
ON NEXT 5M MAXSIZE 500M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para las tablas relacionadas con grupos (grupo_usuario, grupos, grupos_por_proyecto)*

```
CREATE TABLESPACE "TAB_GRUPOS"  
LOGGING  
DATAFILE "/nfs/pdm/info_a/TAB_GRUPOS.ora"  
SIZE 5M AUTOEXTEND  
ON NEXT 1M MAXSIZE 20M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para la tabla acl*

```
CREATE TABLESPACE "TAB_ACL"  
LOGGING  
DATAFILE "/nfs/pdm/info_a/TAB_ACL.ora"  
SIZE 5M AUTOEXTEND  
ON NEXT 1M MAXSIZE 20M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO ;
```

- *Tablespace para la tabla masterdata*

```
CREATE TABLESPACE "TAB_MASTERDATA"  
LOGGING  
DATAFILE "/nfs/pdm/info_c/TAB_MASTERDATA.ora"  
SIZE 10M AUTOEXTEND  
ON NEXT 5M MAXSIZE 500M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para otros datos de ficheros (proyecto, tipo_fichero, estado_fichero, estado_consulta)*

```
CREATE TABLESPACE "TAB_OTROS_DATOS"  
LOGGING  
DATAFILE "/nfs/pdm/info_a/TAB_OTROS_DATOS.ora"  
SIZE 5M AUTOEXTEND  
ON NEXT 1M MAXSIZE 50M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para los índices*

```
CREATE TABLESPACE "TAB_INDICES"  
LOGGING  
DATAFILE "/nfs/pdm/info_c/TAB_INDICES.ora"
```

```
SIZE 10M AUTOEXTEND  
ON NEXT 5M MAXSIZE 100M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para las consultas*

```
CREATE TABLESPACE "TAB_CONSULTAS"  
LOGGING  
DATAFILE "/nfs/pdm/info_a/TAB_CONSULTAS.ora"  
SIZE 20M AUTOEXTEND  
ON NEXT 5M MAXSIZE 500M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para los roles y tablas relacionadas con ellos.*

```
CREATE TABLESPACE "TAB_ROLES"  
LOGGING  
DATAFILE "/nfs/pdm/info_a/TAB_ROLES.ora"  
SIZE 5M AUTOEXTEND  
ON NEXT 1M MAXSIZE 20M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- *Tablespace para las asociaciones*

```
CREATE TABLESPACE "TAB_ASOCIACIONES"  
LOGGING  
DATAFILE "/nfs/pdm/info_b/TAB_ASOCIACIONES.ora"  
SIZE 20M AUTOEXTEND  
ON NEXT 1M MAXSIZE 20M EXTENT MANAGEMENT LOCAL SEGMENT  
SPACE MANAGEMENT AUTO;
```

- **Tablas:** el conjunto de tablas que se han diseñado son las siguientes:
 - *Tipo_ACL*: cada ACL puede ser, en principio, de dos tipos. Puede ser un ACL definido para los grupos o para los roles.
 - *Estado_fichero*: cada uno de los ficheros puede pasar por diferentes estados, en función del proceso de desarrollo. Es en esta tabla en la que se guardan dichos estados.
 - *Estado_consulta*: cada una de las consultas puede pasar por dos estados, abierta o cerrada.
 - *ACL*: la seguridad de la aplicación está diseñada mediante listas de control de acceso. Por defecto un usuario nuevo no tiene acceso a realizar ninguna operación, de modo que se le tienen que dar permisos para lo que sea necesario. Se trata de permisos acumulativos, ya que un usuario puede estar en grupos o en roles y tener diferentes permisos en cada uno de ellos, por lo que los permisos reales serán la suma de ellos; es decir, si un usuario rige sus operaciones por dos acls diferentes, sus permisos completos se definen mediante una operación lógica de tipo OR de ambos acls. Cada acl está diferenciado del resto mediante el tipo de acl, el nombre, el estado actual del fichero y si es o no propietario del fichero.
 - *Grupos*: con ellos se pretende reunir a un conjunto de usuarios que deban realizar funciones parecidas. Mantiene una relación directa con los proyectos y las directivas de acceso ya que cada grupo tendrá unos permisos específicos dependiendo del proyecto en el que esté.
 - *Rol*: con ellos se le da a un usuario concreto unos permisos específicos en un proyecto determinado.

- *Empresa*: se definen todas las empresas que tienen alguna relación con la aplicación, incluyendo los datos relevantes de cada una de ellas.
- *Usuario*: se definen todos los usuarios que tendrán acceso a la aplicación, incluyendo los datos más importantes.
- *Proyecto*: en esta tabla se crean los proyectos de la aplicación. Cada uno de los ficheros que se creen deben estar incluidos en un proyecto.
- *Tipo_fichero*: en esta tabla se guardan todos los tipos de ficheros que existen en la base de datos. Cada fichero tiene una extensión y en esta tabla se muestra la aplicación con la que se puede abrir/editar.
- *Grupo_usuario*: en esta tabla se crean los enlaces que hay entre los grupos y los usuarios. Cada usuario puede estar en uno o varios grupos, del mismo modo que un grupo puede estar formado por uno o varios usuarios.
- *Grupos_por_proyecto*: en esta tabla están las relaciones que hay entre los grupos, los proyectos y los ACLs, de modo que un grupo dentro de un proyecto sólo puede tener un ACL.
- *Masterdata*: se crea toda la información relacionada con los masterdatas. Se trata de un contenedor en el cuál están todas las versiones de un documento, ya sea fichero o asociación.

- *Fichero*: tabla en la que se almacenan todos los datos de un fichero. Cada fichero tiene un masterdata, un estado y un propietario. Además, debe estar asignado a un proyecto. Pueden existir varias versiones para cada fichero, por lo que la dupla nombre/versión debe ser única.
- *Consulta*: se guardan todas las consultas que los usuarios pueden realizar. Éstas serán contestadas por los administradores de la aplicación.
- *Cambio_estado_consulta*: en esta tabla se define la posibilidad o no de realizar un cambio de estado de las consultas, dado el estado actual. Por ejemplo, se puede cerrar una consulta abierta, pero no lo contrario.
- *Cambio_estado_fichero*: en esta tabla se define la posibilidad o no de realizar un cambio de estado de los ficheros, dado el estado actual. Por ejemplo, se puede poner en release una pieza en work, pero no al revés.
- *Usuario_rol_proyecto*: tabla en la que se define el ACL que tiene un usuario dado un rol y trabajando en un proyecto en concreto.
- *Asociación*: tabla en la que se crean todos los datos relativos a una asociación, que consiste en un conjunto de ficheros, en un conjunto de ficheros o asociaciones o en un conjunto de asociaciones, pudiendo cada asociación tener n niveles en su estructura.
- *Masterdata_en_asociación*: se definen las relaciones que tiene cada asociación con los documentos que lo forman. Las relaciones están definidas por el masterdata, ya que cada fichero o cada asociación tiene un masterdata diferente.

Se han escogido varios tipos de datos como claves en las tablas definidas. Algunas son introducidas por los usuarios o administradores, como es el caso del DNI de un usuario, y otras deben ser insertadas automáticamente por la base de datos al realizar una inserción. Debido a este último caso, se han tenido que generar secuencias (tipo de objetos utilizados para establecer y asociar una numeración correlativa de manera automática en las columnas que se quiera) y disparadores con la finalidad de conseguir claves incrementales para aquéllas que deben ser definidas durante la inserción. La descripción de cada una de las tablas de la base de datos, así como las secuencias y disparadores quedan expuestos en el Apéndice B.

- **Índices:** los índices en bases de datos son estructuras de datos que mejoran la velocidad de operaciones en una tabla o más tablas. En el caso de Oracle, por defecto crea árboles-B, los cuales mantienen los datos ordenados y las inserciones y eliminaciones se realizan en tiempo logarítmico amortizado (encontrar el tiempo medio de proceso por operación sobre una secuencia de operaciones en el peor de los casos; garantiza el tiempo por operación sobre el rendimiento en el peor caso). Los árboles-B, por tanto, son usados para evitar largas operaciones de búsqueda, de modo que el sistema no tenga que realizar un escaneo completo de la/s tabla/s involucradas.

Sin embargo, el uso de los índices tiene ventajas y desventajas:

Ventajas:

- acceso rápido para consultas con matching exacto y búsqueda por rango.
- No se requiere espacio para almacenar los ROWID (identificador único para cada fila de la base de datos) de los registros.

Inconveniente:

- cada vez que se hace una inserción, actualización o borrado sobre una tabla, la base de datos tiene que actualizar cualquier índice en la tabla para reflejar los cambios en los datos.

La razón para tener un índice en una columna es permitir que se ejecuten las búsquedas tan rápido como sea posible y evitar los escaneos completos de tablas. Además, si el porcentaje de entradas duplicadas en la tabla es demasiado alto, no habrá ninguna mejora sustancial con el uso de un índice.

Otro tema a considerar es la frecuencia de uso de los índices. Oracle usará un índice para una columna en particular si dicha columna aparece en la cláusula WHERE en una consulta. Si rara vez se usa una columna en la cláusula WHERE, no tiene mucho sentido indexar dicha columna, siendo más eficiente sufrir el escaneo completo de la tabla las raras ocasiones en que se use esta columna en una consulta, que estar actualizando el índice cada vez que cambien los datos de la tabla.

Por defecto, Oracle crea automáticamente los índices de los campos definidos como clave primaria ó únicos. Sin embargo, los campos definidos como claves foráneas no generan índices automáticamente. El hecho de que las claves foráneas no estén indexadas, pueden provocar problemas de rendimiento (gran cantidad de joins entre las tablas se harán basándose en las columnas correspondientes a las claves foráneas), problemas de concurrencias y bloqueos que degradarán enormemente el rendimiento de la actividad de actualizaciones de la base de datos. El problema es que cualquier actualización sobre la tabla padre deberá bloquear toda la tabla hija si no hay un índice sobre las columnas que corresponden a la clave foránea, de modo que se asegure la integridad.

Normalmente los índices son creados en un tablespace específico distinto al de datos y situados en diferentes discos, ya que incrementa la accesibilidad de los datos, aunque se pierde facilidad en el mantenimiento de la base de datos para realizar backups. Debido a todos estos datos, se ha optado por indexar varios campos de la base de datos, además de las claves primarias y las claves únicas, que se indexan automáticamente. Los campos que se han indexado quedan expuestos en el Apéndice B.

4.3. Implementación de la aplicación

4.3.1. Modelo

Esta capa queda dividida en dos partes: los componentes de estado y de proceso. En toda aplicación que deba trabajar con una base de datos relacional aparece el problema de equivalencia entre el modelo de objetos con el modelo de tablas o relacional; es decir, la integración con la base de datos para guardar, actualizar y recuperar la información que utiliza la aplicación. Se llama persistencia de los objetos a su capacidad para guardarse y recuperarse desde un medio de almacenamiento. En nuestro caso la persistencia ha sido implementada mediante el patrón DAO (Data Access Object), que mapea objetos Java a la base de datos, aunque se podría haber escogido algún framework que automatiza el proceso a partir de mapeos, como Hibernate. DAO consiste en generar una clase que contenga un atributo para cada campo de la tabla y otra clase que contenga los métodos para insertar, actualizar, seleccionar y eliminar registros de dicha tabla. Las principales ventajas de este patrón de diseño son:

- Simple
- Separación de datos o componentes de estado (DTO, Objeto de transferencia de datos) y comportamiento o proceso (DAO)
- Código eficiente, siempre que se diseñe utilizando las habilidades de la BD.

También tiene alguna desventaja:

- Requiere implementar grandes volúmenes de código.
- Escribir código relacionado con el modelo de objetos requiere conocimiento del esquema de la base de datos.

La parte de separación de datos (DTO) se ha implementado mediante Java Beans serializables, uno para cada tabla de la base de datos creada. Un Java Bean es una clase Java que sigue ciertas normas, incluyendo los métodos que usa. Son muy útiles ya que son portables, reutilizables e independientes de la plataforma. Los Java Beans están formados por una serie de atributos privados que pueden ser accedidos mediante el uso de funciones get y set definidas en la misma clase. Además debe tener un constructor.

Un ejemplo de Java Bean sería:

```
/*Objeto de Transferencia de Datos para la tabla USUARIO*/
package project.struts.OTD;
import java.io.Serializable;

public class OTDUsuario implements Serializable
{
    private String dni = null;
    private String passwd = null;
    private String nombre = null;
    private String apellido1 = null;
    private String apellido2 = null;
    private String tlffijo = null;
    private String tlfmovil = null;
    private String mail = null;
    private String rEmpresa = null;

    //CONSTRUCTOR
    public OTDUsuario(String dni, String passwd, String nombre, String ciudad, String apellido1, String apellido2,
                      String tlffijo, String tlfmovil, String mail, String rEmpresa)
    {
        this.dni = dni;
        this.passwd = passwd;
        this.nombre = nombre;
        this.apellido1 = apellido1;
        this.apellido2 = apellido2;
        this.tlffijo = tlffijo;
        this.tlfmovil = tlfmovil;
        this.mail = mail;
        this.rEmpresa = rEmpresa;
    }

    public OTDUsuario()
    {

    }

    //ARCHIVOS GET
    public String getDni() {return dni;}
    public String getPasswd() {return passwd;}
    public String getNombre() {return nombre;}
    public String getApellido1() {return apellido1;}
    public String getApellido2() {return apellido2;}
    public String getTlffijo() {return tlffijo;}
    public String getTlfmovil() {return tlfmovil;}
```

```

public String getMail() {return mail;}
public String getrEmpresa() {return rEmpresa;}

//ARCHIVOS SET
public void setDni(String string) {dni = string;}
public void setPasswd(String string) {passwd = string;}
public void setNombre(String string) {nombre = string;}
public void setApellido1(String string) {apellido1 = string;}
public void setApellido2(String string) {apellido2 = string;}
public void setTlfFijo(String string) {tlfFijo = string;}
public void setTlfMovil(String string) {tlfMovil = string;}
public void setMail(String string) {mail = string;}
public void setrEmpresa(String string) {rEmpresa = string;}
}

```

Para la parte de comportamiento se han definido varios Beans que acceden a la base de datos mediante JDBC (API de Java que define cómo una aplicación cliente accederá a la base de datos, independientemente del motor de base de datos) y se encargan de las inserciones, búsquedas, actualizaciones, eliminaciones, etc.

```

public class CambiarPasswdBean
{
    private DataSource dataSource = null;

    public CambiarPasswdBean(DataSource dataSource)
    {
        this.dataSource = dataSource;
    }

    public OTDUsuario changePasswd (String dni, String passwd) throws ModuleException
    {
        OTDUsuario usuario = null;
        Connection con = null;
        Statement stmt = null;
        ResultSet rs = null;
        String sQuery =;

        try
        {
            con = dataSource.getConnection();
            stmt = con.createStatement();
            //Crea la query para insertar en la base de datos
            sQuery = "UPDATE USUARIO SET Passwd ='" + passwd + "' WHERE Dni= '" + dni + "'";
            //Ejecuta la inserción de la query
            int resultado = stmt.executeUpdate(sQuery);
            if (resultado==1)
            {
                //Si se ha creado el proyecto, se busca ese proyecto y se obtienen todos los datos
                sQuery = "SELECT * FROM USUARIO " + "WHERE dni = '" + dni + "'";
                rs = stmt.executeQuery(sQuery);
                if (rs.next())
                {
                    usuario = new OTDUsuario();
                    usuario.setDni(rs.getString("Dni"));
                    usuario.setPasswd(rs.getString("Passwd"));
                    usuario.setNombre(rs.getString("Nombre"));
                    usuario.setApellido1(rs.getString("Apellido1"));
                    usuario.setApellido2(rs.getString("Apellido2"));
                }
            }
        }
        catch (SQLException se)
        {
            se.printStackTrace();
            ModuleException me =new ModuleException("error.db.sql");
            throw me;
        }
        catch (Exception e)
        {

```

```

        e.printStackTrace();
        ModuleException me =new ModuleException(.error.db.sql");
        throw me;
    }
    finally
    {
        try
        {
            if (rs != null) rs.close();
        }
        catch (SQLException se)
        {
            se.printStackTrace();
            ModuleException me =new ModuleException(.error.db.sql");
            throw me;
        }
        try
        {
            if (stmt != null) stmt.close();
        }
        catch (SQLException se)
        {
            se.printStackTrace();
            ModuleException me =new ModuleException(.error.db.sql");
            throw me;
        }
        try
        {
            if (con != null) con.close();
        }
        catch (SQLException se)
        {
            se.printStackTrace();
            ModuleException me =new ModuleException(.error.db.sql");
            throw me;
        }
    }
}
return usuario;
}
}

```

Además de estos tipos de clases se han creado varias que definen diferentes procesos que se han necesitado, como una clase en la que se implementan varias operaciones con strings o una clase que define el formato de una fecha y operaciones relacionadas.

4.3.2. Vista

La parte de la vista está construida usando JavaScript, JavaServer Pages, JSTL y la API de Struts. Además también se han utilizado varias librerías de código abierto: uitags y displaytag.

Uno de los ficheros más importantes es ApplicationResources.properties, en el cual quedan definidos todos los textos que aparecen en la aplicación. De este modo, si se quiere cambiar algún texto en la aplicación, no hay que buscar en todos los ficheros para ver dónde

está, sino que es completamente seguro que está en este fichero. Separando los textos del código provoca que sea más fácil cambiar cualquier texto, y por lo tanto, hace la aplicación más fácil de mantener. Además este tipo de ficheros también se puede utilizar para tener la aplicación en varios idiomas. Simplemente es necesario crear un fichero equivalente a cada idioma, pero renombrando el fichero correctamente; por ejemplo, si se quisiera utilizar en inglés, se podría definir el fichero ApplicationResources_en.properties, el cual tendría los mismos campos definidos, pero los textos serían en el idioma especificado.

```
#Página Menus-;Buscar-;Masterdata
buscarMasterdata.titulo=Buscar masterdata
listarMasterdatas.titulo=Masterdatas encontrados
verDatosMasterdata.titulo=Datos del masterdata
cambiarProyecto.titulo=Cambiar proyecto
transferirMasterdata.titulo=Transferir masterdata
masterdatas.nombre=Nombre
masterdatas.propietario=Propietario
masterdatas.proyecto=Proyecto
masterdatas.fechaCreacion=Fecha de creación
masterdatas.fechaActualizacion=Fecha de actualización
masterdatas.masterdataTransfer=Masterdata transferido
masterdatas.descripcion=Descripción
masterdatas.transferAceptada=Transfer aceptada
```

La primera página que el usuario ve es la de acceso a la aplicación -figura 4.9-, en la que deberá introducir los datos que definen su cuenta.



Figura 4.9: Acceso a la aplicación

El resto de las páginas que forman la aplicación tienen en la parte superior un menú implementado en JavaScript -figura 4.10-.



Figura 4.10: Menú principal

En la figura 4.11 se puede observar como a partir de este menú se puede ir a las páginas de cada una de las opciones principales.



Figura 4.11: Extensión del menú principal

Una vez situado en las páginas principales de cada menú se realizan las operaciones que se quiera, siempre y cuando el usuario conectado tenga permiso para ello.

En la mayoría de dichas páginas existen formularios con los que el usuario deberá interactuar para obtener o crear la información necesaria, como se puede ver en la figura 4.12, dónde aparece un formulario con varios campos que el usuario puede utilizar, ya sea para realizar una búsqueda o para crear información.

Buscar masterdata

Nombre	<input type="text"/>
Propietario	<input type="button" value="Seleccionar"/>
Proyecto	<input type="button" value="Seleccionar"/>
Fecha de creación	<input type="text"/>
Fecha de actualización	<input type="text"/>
Masterdata transferido	<input type="button" value="Seleccionar"/>
Descripción	<input type="text"/>

Figura 4.12: Formulario de búsqueda de masterdatas

Obviamente cada formulario debe realizar validaciones de los datos introducidos por el usuario. Para ello se utiliza la validación propia de Struts (Validator). Éstas están definidas en el fichero validation.xml.

```
<!-- Validaciones para el formulario de Crear Usuario -->
<form name="crearUsuarioForm">
  <field depends="required,mask" property="dni">
    <arg0 key="error.dni"/>
    <msg key="error.dni.mascara" name="mask" resource="true"/>
    <var>
      <var-name>mask</var-name>
      <var-value>${dni}</var-value>
    </var>
  </field>
  <field depends="required,maxlength" property="passwd">
    <arg0 key="error.usuarios.password"/>
    <arg1 key="${var:maxlength}" resource="false"/>
    <var>
      <var-name>maxlength</var-name>
      <var-value>8</var-value>
    </var>
  </field>
</form>
```

Si alguna de éstas falla, la aplicación genera la misma página HTML, pero mostrando todos los errores de validación -ver figura 4.13-. Si la validación es correcta, el controlador continúa con el proceso

The screenshot shows a user creation form titled "Crear usuario". The fields and their validation errors are:

- DNI: Campo DNI es obligatorio
- Contraseña: Campo Contraseña es obligatorio
- Nombre: Campo Nombre es obligatorio
- Primer apellido: Campo Primer apellido es obligatorio
- Segundo apellido:
- Teléfono fijo:
- Teléfono móvil:
- Dirección de correo: Campo Dirección de correo es obligatorio
- Empresa: Arion Data Systems

At the bottom are "Reset" and "Crear" buttons.

Figura 4.13: Formulario de creación de usuarios

La segunda parte de la validación está relacionada con las operaciones realizadas contra la base de datos. En este caso los errores se muestran tras validar información en la base de datos, obteniendo el error en el mismo formulario que se ha enviado. En la figura 4.14 se puede ver como en el momento de intentar crear un usuario, se recibe un error que informa de que el dni que se ha introducido ya existe.

The screenshot shows a user creation form titled "Crear usuario". The fields and their values are:

- DNI: 44016794S
- Contraseña: *****
- Nombre: Francisco
- Primer apellido: Exposito
- Segundo apellido: Aguilera
- Teléfono fijo:
- Teléfono móvil:
- Dirección de correo: fooexposito@mail.com
- Empresa: Arion Data Systems

At the bottom are "Reset" and "Crear" buttons. A message at the bottom states: "* El dni del usuario ya existe"

Figura 4.14: Errores de base de datos

Respecto a las páginas en las que se realizan búsquedas, hay tres opciones:

- No se encuentra ningún resultado: se muestra un mensaje de error del tipo “operaciones realizadas contra la base de datos”.
- Se encuentra un resultado (figura 4.15): en este caso se muestran los datos que definen el elemento buscado y las operaciones posibles.

Datos del fichero	
Nombre	Federol
Versión	1.1
Masterdata	Federol
Tipo	txt
Estado	Personal
Descripción	Dese1
Fecha de creación	29/03/2007
Fecha de actualización	29/03/2007
Tamaño (bytes)	0
Proyecto	Golden Gate
Propietario	Francisco Exposito
Reservado	Sí
<input type="button" value="Descargar"/> <input type="button" value="Guardar versión"/> <input type="button" value="Cambiar versión"/> <input type="button" value="Cambiar estado"/> <input type="button" value="Borrar"/> <input type="button" value="Dereservar"/>	

Figura 4.15: Búsqueda de ficheros con único resultado

- Se encuentra más de un resultado (figura 4.16): en este caso se muestra una tabla generada mediante el uso de displaytag, existiendo la posibilidad de realizar una o más operaciones con cada elemento encontrado.

ID	Nombre	Operaciones
66	Administrators	Editar Usuarios Proyectos Borrar
61	Designers	Editar Usuarios Proyectos Borrar
62	Designers Flex	Editar Usuarios Proyectos Borrar
63	Designers Venture	Editar Usuarios Proyectos Borrar
64	Managers	Editar Usuarios Proyectos Borrar
65	Viewers	Editar Usuarios Proyectos Borrar

6 registros encontrados

Figura 4.16: Búsqueda de ficheros con varios resultados

En la parte de administración se ha utilizado también uitags, mediante el cuál hemos añadido la posibilidad de, dadas dos colecciones de datos en forma de columnas, pasar información de una a otra. De este modo hemos configurado, por ejemplo, añadir un usuario a uno o varios grupos, como se muestra en la figura.

En el caso mostrado en la figura 4.17, hay una columna en la que aparecen todos los grupos existentes y otra columna en la que se muestran los grupos a los que ya pertenece el usuario, pudiendo realizar cambios mediante el uso de los botones que hay entre ambas columnas.



Figura 4.17: Uso de uitags en interfaz gráfica

Una de las partes más importantes de la aplicación es la carga y descarga de ficheros. Hay dos tipos de carga de ficheros:

- Al crear un documento nuevo (figura 4.18): sólo son necesarios un proyecto, una descripción y la selección del fichero, siendo el resto de la información creada automáticamente al enviar el formulario.
- Al crear una versión nueva o modificar una existente (figura 4.19): son necesarios tres campos, del mismo modo que al crear un documento nuevo; en este caso, el proyecto ya queda definido, puesto que es el mismo que el de la versión de la que proviene; sin embargo, falta por definir qué tipo de versión será creada.

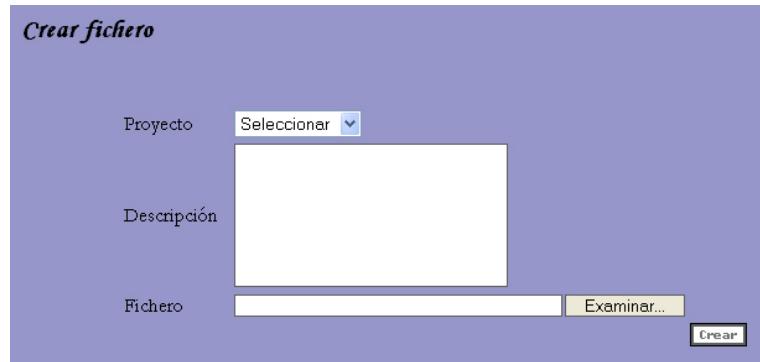


Figura 4.18: Creación de un fichero nuevo

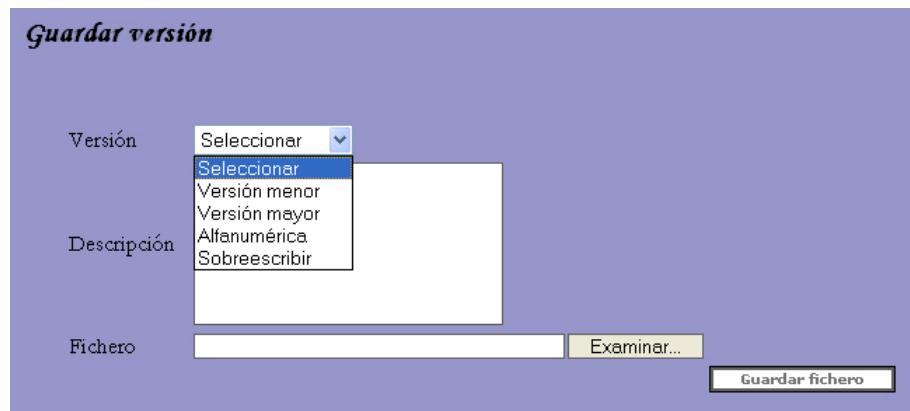


Figura 4.19: Creación de la versión de un fichero

En lo que se refiere a descargar ficheros, al pulsar el botón Descargar, se obtiene una ventana típica en la que se puede escoger entre abrir o guardar el fichero seleccionado -ver figura 4.20-.



Figura 4.20: Descarga de ficheros

Por otra parte, existen las asociaciones. Éstas son conjuntos de ficheros o de otras asociaciones, que pueden tener hasta n niveles de estructura. En ellas hay dos posibilidades de descarga, que aparecen en el momento de intentar descargar la asociación: descargar la versión más alta o descargar el estado más alto. Un ejemplo de estos tipos de descarga se muestra en la figura 4.21.

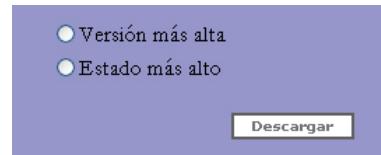


Figura 4.21: Tipos de descarga de asociaciones

Suponiendo que la asociación A1 está formada por los masterdatas F1, F2 y A2, donde A2 está formada por los masterdatas F3 y F4, con las siguientes versiones:

```

F1:
v1 / 1.1 / work
v2 / 1.2 / release

F2:
v1 / 1.1 / release
v2 / 1.2 / work

F3:
v1 / 1.1 / work
A.1/ 1.2 / under review

F4:
v1 / 1.1 / work
v2 / 1.2 / release
v3 / 1.3 / work
  
```

Si se escoge descargar versión más alta obtendremos los ficheros F1 v2, F2 v2, F3 v2 y F4 v3, mientras que si escogemos descargar estado más alto obtendremos F1 v2, F2 v1, F3 v2 y F4 v2.

4.3.3. Controlador

La parte del controlador está enfocada en las solicitudes recibidas desde el cliente, decidiendo qué función de la lógica de negocio se va a realizar, y a continuación delegando la responsabilidad para producir la siguiente fase del interface de usuario en un componente vista apropiado. En Struts, el componente principal del controlador es un servlet de la clase ActionServlet, el cual está configurado definiendo un conjunto de ActionMappings. El fichero en el que se definen todos los mapeos de direcciones recibe el nombre de struts-config.xml. Sin embargo, con la finalidad de independizar estos mapeos, se ha optado por crear varios ficheros xml, uno para cada opción del menú, además del struts-config.xml, y para activarlos son todos llamados desde el fichero web.xml. En el fichero struts-config.xml destaca la configuración del pool de conexiones, implementada utilizando Jakarta Commons DBCP, la definición del FormBean, mapeos de acceso al sistema y la ruta del fichero de validación que ha sido nombrado anteriormente. Por último, también queda definido el tamaño máximo de los ficheros que pueden ser guardados y que ha sido definido en 2GB.

En el resto de archivos xml en que hemos dividido el struts-config.xml quedan expuestos los FormBeans y los ActionMappings de cada acción.

En los FormBeans se definen todas las propiedades que son susceptibles de ser transferidas al sistema desde el formulario.

```
<form-bean name="crearAsociacionForm" type="org.apache.struts.validator.DynaValidatorForm">
  <form-property name="idMasterdata" type="java.lang.String[]"/>
  <form-property name="idAsociacion" type="java.lang.String"/>
  <form-property name="nombre" type="java.lang.String"/>
  <form-property name="version" type="java.lang.String"/>
  <form-property name="rPropietario" type="java.lang.String"/>
  <form-property name="rProyecto" type="java.lang.String"/>
  <form-property name="fechaCreacion" type="java.lang.String"/>
  <form-property name="fechActualizacion" type="java.lang.String"/>
  <form-property name="rMasterdata" type="java.lang.String"/>
  <form-property name="descripcion" type="java.lang.String"/>
  <form-property name="estructura" type="java.lang.String"/>
</form-bean>
```

Un ActionMapping define un path que se compara contra la URI solicitada de la petición entrante, y normalmente especifica el nombre totalmente cualificado de clase de una clase Action. Las acciones encapsulan la lógica de negocio, interpretan la salida y, por último, despachan el control al componente vista apropiado para la respuesta creada.

Por tanto, en los ActionMappings se definen los mapeos que puede sufrir una página dependiendo del botón pulsado y, dada la respuesta de la función que se ha ejecutado, mostrará una u otra página.

```
<action input="/Menus/Crear/crearAsociacion.jsp" name="crearAsociacionForm"
    path="/crearAsociacion" scope="request" parameter="accion"
    type="project.struts.Actions.Crear.CrearAsociacionAction" validate="false">
    <forward name="goSearchMasterdataToAdd" path="/Menus/Crear/buscarMasterdataAregarAsociacion.jsp"/>
    <forward name="notGoSearchMasterdataToAdd" path="/Menus/Crear/crearAsociacion.jsp"/>
    <forward name="showMasterdatasfounded" path="/Menus/Crear/listarMasterdatasEncontrados.jsp"/>
    <forward name="notShowMasterdatasfounded" path="/Menus/Crear/buscarMasterdataAregarAsociacion.jsp"/>
    <forward name="seeInfoAsociacionToSave" path="/Menus/Crear/verDatosNuevaAsociacion.jsp"/>
    <forward name="notSeeInfoAsociacionToSave" path="/Menus/Crear/crearAsociacion.jsp"/>
    <forward name="asociacionSaved" path="/general.jsp"/>
    <forward name="asociacionNotSaved" path="/Menus/Crear/verDatosNuevaAsociacion.jsp"/>
    <forward name="created" path="/general.jsp"/>
    <forward name="notCreated" path="/Menus/Crear/crearAsociacion.jsp"/>
</action>
```

Capítulo 5

Pruebas

En este apartado se explican las pruebas a las cuales ha sido sometida la aplicación.

En primer lugar, cada una de las funciones u operaciones que se han definido han sido probadas individualmente. Para ello se ha creado una serie de información, de manera que la seguridad de la aplicación quede chequeada. La información creada ha sido:

1. Usuarios de la aplicación: se muestran los campos necesarios para el chequeo de la aplicación.

USUARIOS			
DNI	Contraseña	Nombre	Apellido1
99999999A	admin	Administrador	Admin
44016794S	pacoexpo	Francisco	Expósito
74122055D	090182	Chelo	Figueroa
11111111A	pa2primo	Josep	Sierra
11111111B	asdfpoiuy	Marta	Taba
11111111C	2wsxzaq1	Jerry	Class
11111111D	mnbvasdf	Agusti	Gutierrez
11111111E	1qaz2wsx	Curt	Targ
11111111F	bon2sort	Josep	Ort
11111111G	salamina	Ismael	Perez
11111111H	sanjordi	Eduardo	Martinez
11111111I	ronillo1	Antonio	Menendez

2. Proyectos que tendrán documentos.

PROYECTOS
A
B
C
D
E

3. Grupos y usuarios que pertenecen a cada uno de los grupos.

<i>Grupos</i>	<i>Usuarios en grupos</i>			
Designers	Ismael Perez	Eduardo Martinez		
Designers1	Jerry Class	Agusti Gutierrez		
Designers2	Curt Tart	Josep Ort		
Managers	Antonio Menendez	Josep Sierra		
Viewers	Chelo Figueroa	Francisco Exposito		
Administrators	Administrador	Marta Taba		

4. Roles.

Roles
Releasers

5. Tipo de ACL.

Tipos de ACL
Grupos
Roles

6. Estados posibles de los ficheros y cambios posibles de un estado actual a un estado futuro.

Estados de fichero	Cambios de estado de ficheros	
	Origen	Destino
Personal	Personal	Work
Work	Personal	Under review
Under Review	Personal	Release
Release	Personal	Obsolete
Support	Work	Under review
Obsolete	Work	Release
	Work	Obsolete
	Under review	Release
	Under review	Obsolete
	Release	Support
	Release	Obsolete
	Support	Obsolete

7. ACLs: debido a la extensión de la tabla, se han obviado los permisos de cada uno de los ACLs.

ACLS			
Nombre	TipoAcl	Estado	Propietario
DEFAULT_ADMIN_ACL	Grupos	Personal	Si
DEFAULT_ADMIN_ACL	Grupos	Work	Si
DEFAULT_ADMIN_ACL	Grupos	Under review	Si
DEFAULT_ADMIN_ACL	Grupos	Release	Si
DEFAULT_ADMIN_ACL	Grupos	Support	Si
DEFAULT_ADMIN_ACL	Grupos	Obsolete	Si
DEFAULT_ADMIN_ACL	Grupos	Personal	No
DEFAULT_ADMIN_ACL	Grupos	Work	No
DEFAULT_ADMIN_ACL	Grupos	Under review	No
DEFAULT_ADMIN_ACL	Grupos	Release	No
DEFAULT_ADMIN_ACL	Grupos	Support	No
DEFAULT_ADMIN_ACL	Grupos	Obsolete	No
DEFAULT_PROJECT_ACL	Grupos	Personal	Si
DEFAULT_PROJECT_ACL	Grupos	Work	Si
DEFAULT_PROJECT_ACL	Grupos	Under review	Si
DEFAULT_PROJECT_ACL	Grupos	Release	Si
DEFAULT_PROJECT_ACL	Grupos	Support	Si
DEFAULT_PROJECT_ACL	Grupos	Obsolete	Si
DEFAULT_PROJECT_ACL	Grupos	Personal	No
DEFAULT_PROJECT_ACL	Grupos	Work	No
DEFAULT_PROJECT_ACL	Grupos	Under review	No
DEFAULT_PROJECT_ACL	Grupos	Release	No
DEFAULT_PROJECT_ACL	Grupos	Support	No
DEFAULT_PROJECT_ACL	Grupos	Obsolete	No
DEFAULT_ELEMENT_ACL	Grupos	Personal	Si
DEFAULT_ELEMENT_ACL	Grupos	Work	Si
DEFAULT_ELEMENT_ACL	Grupos	Under review	Si
DEFAULT_ELEMENT_ACL	Grupos	Release	Si
DEFAULT_ELEMENT_ACL	Grupos	Support	Si
DEFAULT_ELEMENT_ACL	Grupos	Obsolete	Si
DEFAULT_ELEMENT_ACL	Grupos	Personal	No
DEFAULT_ELEMENT_ACL	Grupos	Work	No
DEFAULT_ELEMENT_ACL	Grupos	Under review	No
DEFAULT_ELEMENT_ACL	Grupos	Release	No
DEFAULT_ELEMENT_ACL	Grupos	Support	No
DEFAULT_ELEMENT_ACL	Grupos	Obsolete	No

ACLS			
Nombre	TipoAcl	Estado	Propietario
DEFAULT_MASTERDATA_ACL	Grupos	Personal	Si
DEFAULT_MASTERDATA_ACL	Grupos	Work	Si
DEFAULT_MASTERDATA_ACL	Grupos	Under review	Si
DEFAULT_MASTERDATA_ACL	Grupos	Release	Si
DEFAULT_MASTERDATA_ACL	Grupos	Support	Si
DEFAULT_MASTERDATA_ACL	Grupos	Obsolete	Si
DEFAULT_MASTERDATA_ACL	Grupos	Personal	No
DEFAULT_MASTERDATA_ACL	Grupos	Work	No
DEFAULT_MASTERDATA_ACL	Grupos	Under review	No
DEFAULT_MASTERDATA_ACL	Grupos	Release	No
DEFAULT_MASTERDATA_ACL	Grupos	Support	No
DEFAULT_MASTERDATA_ACL	Grupos	Obsolete	No
ONLY_READ_ACL	Grupos	Personal	Si
ONLY_READ_ACL	Grupos	Work	Si
ONLY_READ_ACL	Grupos	Under review	Si
ONLY_READ_ACL	Grupos	Release	Si
ONLY_READ_ACL	Grupos	Support	Si
ONLY_READ_ACL	Grupos	Obsolete	Si
ONLY_READ_ACL	Grupos	Personal	No
ONLY_READ_ACL	Grupos	Work	No
ONLY_READ_ACL	Grupos	Under review	No
ONLY_READ_ACL	Grupos	Release	No
ONLY_READ_ACL	Grupos	Support	No
ONLY_READ_ACL	Grupos	Obsolete	No
RELEASE_ACL	Roles	Personal	Si
RELEASE_ACL	Roles	Work	Si
RELEASE_ACL	Roles	Under review	Si
RELEASE_ACL	Roles	Release	Si
RELEASE_ACL	Roles	Support	Si
RELEASE_ACL	Roles	Obsolete	Si
RELEASE_ACL	Roles	Personal	No
RELEASE_ACL	Roles	Work	No
RELEASE_ACL	Roles	Under review	No
RELEASE_ACL	Roles	Release	No
RELEASE_ACL	Roles	Support	No
RELEASE_ACL	Roles	Obsolete	No

8. Permisos que tiene un usuario en un proyecto dado un rol determinado.

USUARIO_ROL_PROYECTO			
Usuario	Proyecto	Rol	Acl
Marta Taba	2	releasers	release_acl
Marta Taba	4	releasers	release_acl
Josep Sierra	1	releasers	release_acl
Josep Sierra	2	releasers	release_acl
Marta Taba	3	releasers	release_acl
Josep Sierra	3	releasers	release_acl

9. Permisos que tiene un grupo en un proyecto determinado.

		GRUPOS_POR_PROYECTO			
Grupo	Proyecto	Acls			
Designers	A	default_element_acl	default_masterdata_acl		
Designers1	A	No pueden ver nada			
Designers2	A	default_element_acl	default_masterdata_acl		
Managers	A	only_read_acl			
Viewers	A	only_read_acl			
Administrators	A	default_element_acl	default_masterdata_acl	default_admin_acl	default_project_acl
Designers	B	default_element_acl	default_masterdata_acl		
Designers1	B	default_element_acl	default_masterdata_acl		
Designers2	B	No pueden ver nada	default_masterdata_acl		
Managers	B	only_read_acl			
Viewers	B	only_read_acl			
Administrators	B	default_element_acl	default_masterdata_acl	default_admin_acl	default_project_acl
Designers	C	default_element_acl	default_masterdata_acl		
Designers1	C	default_element_acl	default_masterdata_acl		
Designers2	C	default_element_acl	default_masterdata_acl		
Managers	C	only_read_acl	default_masterdata_acl		
Viewers	C	only_read_acl			
Administrators	C	default_element_acl	default_masterdata_acl	default_admin_acl	default_project_acl
Designers	D	default_element_acl	default_masterdata_acl		
Designers1	D	default_element_acl	default_masterdata_acl		
Designers2	D	No pueden ver nada	default_masterdata_acl		
Managers	D	only_read_acl			
Viewers	D	only_read_acl			
Administrators	D	default_element_acl	default_masterdata_acl	default_admin_acl	default_project_acl
Designers	E	default_element_acl	default_masterdata_acl		
Designers1	E	No pueden ver nada	default_masterdata_acl		
Designers2	E	default_element_acl	default_masterdata_acl		
Managers	E	only_read_acl			
Viewers	E	only_read_acl			
Administrators	E	default_element_acl	default_masterdata_acl	default_admin_acl	default_project_acl

Toda esta información ha sido almacenada desde la aplicación y no directamente en Oracle, por lo que mediante dicha introducción se ha comprobado el correcto funcionamiento de la parte de administración de la aplicación. Una vez añadidos estos datos en la base de datos se procedió a realizar las pruebas de cada una del resto de las funciones con cada uno de los usuarios creados.

Algunas de las pruebas realizadas son las siguientes:

- a) Intento de un usuario de descargar un fichero que está en estado “personal” y no es el propietario: en este caso no tiene permiso para realizar ninguna operación, ni siquiera descargar el fichero, ya que sólo el propietario de un documento en estado personal puede hacer operaciones con él. En la figura 5.1 se

puede ver como todos los botones de la página generada están desactivados.

Datos del fichero	
Nombre	<input type="text" value="Fichero1"/>
Versión	<input type="text" value="1.1"/>
Masterdata	<input type="text" value="Fichero1"/>
Tipo	<input type="text" value="txt"/>
Estado	<input type="text" value="Personal"/>
Descripción	
Fecha de creación	<input type="text" value="29/03/2007"/>
Fecha de actualización	<input type="text" value="29/03/2007"/>
Tamaño (bytes)	<input type="text" value="0"/>
Proyecto	<input type="text" value="C"/>
Propietario	<input type="text" value="Francisco Exposito"/>
Reservado	<input type="text" value="Sí"/>
<input type="button" value="Descargar"/> <input type="button" value="Guardar versión"/> <input type="button" value="Cambiar versión"/> <input type="button" value="Cambiar estado"/> <input type="button" value="Borrar"/>	

Figura 5.1: Descarga de fichero en personal sin ser propietario

En el caso de que el fichero estuviera en cualquier otro estado que no fuera personal, entonces el usuario podría únicamente descargar el fichero, como se puede ver en la figura 5.2.

Datos del fichero	
Nombre	<input type="text" value="Fichero3"/>
Versión	<input type="text" value="1.1"/>
Masterdata	<input type="text" value="Fichero3"/>
Tipo	<input type="text" value="txt"/>
Estado	<input type="text" value="Work"/>
Descripción	
Fecha de creación	<input type="text" value="29/03/2007"/>
Fecha de actualización	<input type="text" value="05/06/2007"/>
Tamaño (bytes)	<input type="text" value="0"/>
Proyecto	<input type="text" value="C"/>
Propietario	<input type="text" value="Francisco Exposito"/>
Reservado	<input type="text" value="Sí"/>
<input type="button" value="Descargar"/> <input type="button" value="Guardar versión"/> <input type="button" value="Cambiar versión"/> <input type="button" value="Cambiar estado"/> <input type="button" value="Borrar"/>	

Figura 5.2: Descarga de fichero no en personal sin ser propietario

b) En el caso de que un usuario intente descargar una asociación, si dicha asociación está en un estado accesible para el usuario, además deberá tener permisos para descargar todos los ficheros que hay en el interior de dicha asociación. Si suponemos una asociación con la siguiente estructura:

- Asociación2 en work
 - Fichero3 en work
 - Fichero4 en personal (sólo una versión)

Si la persona que lo intenta cargar es el propietario del Fichero4, será posible descargarlo; en el caso contrario, no podrá y recibirá el mensaje de error de la figura 5.3.

* No tiene permisos para descargar algún fichero de la
asociación

Figura 5.3: Descarga de asociación sin permisos en algún documento

El tema de las asociaciones varía en función del tipo de descarga que se elija y del número de versiones que existan. Dependiendo de estos dos términos un usuario que tiene acceso al proyecto podrá o no descargar la asociación.

- c) Cambio de la contraseña por parte de un usuario. En la imagen inferior se puede ver el mensaje de error que recibe el usuario cuando intenta cambiar su contraseña y los valores introducidos no son correctos.



Figura 5.4: Error en cambio de contraseña

- d) Búsqueda de masterdatas por parte de un usuario. Aparecen sólo los masterdatas a cuyo proyecto tiene acceso el usuario que ha realizado la búsqueda. En el caso que se muestra a continuación -figura 5.5 y 5.6-, en total hay 25 ficheros en la base de datos, pero hay uno al que no tiene acceso el usuario.

Masterdatas encontrados		
ID	Nombre	Operaciones
226	Asociacion1	Ver
227	Asociacion2	Ver
228	Asociacion3	Ver
241	Asociacion4	Ver
242	Asociacion5	Ver
123	CAPITULO-8_ED b	Ver
301	commons-lang-2.3	Ver
288	Estranera	Ver
221	Fichero1	Ver
222	Fichero2	Ver
223	Fichero3	Ver
224	Fichero4	Ver
225	Fichero5	Ver
289	Musica	Ver
287	Nacional	Ver
161	Nuevo Documento de texto	Ver
321	setupesp	Ver
302	Test	Ver
261	Unnombremaslargoqueprobartabla	Ver
284	01-joaquin_sabina-pajaros_de_portugal	Ver

24 registros encontrados [Primero/Anterior] 1, 2 [Siguiente/Último]

Figura 5.5: Búsqueda de masterdata

<i>Masterdatas encontrados</i>		
ID	Nombre	Operaciones
226	Asociacion1	Ver
227	Asociacion2	Ver
228	Asociacion3	Ver
241	Asociacion4	Ver
242	Asociacion5	Ver
123	CAPITULO-8_ED b	Ver
301	commons-lang-2.3	Ver
288	Extranjera	Ver
221	Fichero1	Ver
222	Fichero2	Ver
223	Fichero3	Ver
224	Fichero4	Ver
225	Fichero5	Ver
289	Musica	Ver
287	Nacional	Ver
161	Nuevo Documento de texto	Ver
321	setupesp	Ver
302	Test	Ver
261	Unnombremaslargoaparprobartabla	Ver
281	01 - Diego Torres - Andando	Ver

25 registros encontrados [Primero/Anterior] [1](#), [2](#) [Siguiente/Último]

Figura 5.6: Búsqueda de masterdata-b

Además de estas pruebas individuales, se ha testeado el conjunto de la aplicación mediante el acceso concurrente de cinco usuarios diferentes, los cuales han realizado búsquedas, creado asociaciones y ficheros, guardado ficheros de tamaños variables, creado versiones, etc.

Capítulo 6

Coste económico

Todo proyecto tiene a su vez un coste económico. En este caso el coste está determinado por la implementación de la aplicación como por el hardware necesario para su funcionamiento.

En la parte de hardware podemos considerar el coste de los servidores a utilizas, tanto el de la base de datos como el servidor de ficheros, el cuál estará formado por varios discos para mejorar la E/S, como se ha explicado en el apartado de diseño del almacenamiento. Además se puede considerar coste de hardware la licencia correspondiente del gestor de base de datos, debido a la selección de Oracle. El resto de herramientas utilizadas son libres, por lo que no tienen coste alguno.

La parte de coste de un servidor debe ser minuciosamente pensada por parte del departamento de sistemas y queda fuera del estudio.

En la parte de software podemos considerar los siguientes perfiles:

a) Jefe de proyecto:

- Colaboración con el cliente en la definición y concreción de los objetivos del proyecto.
- Planificación del proyecto en todos sus aspectos, identificando las activi-

dades a realizar, los recursos a poner en juego, los plazos y los costes previstos.

- Dirección y coordinación de todos los recursos empleados en el proyecto.
- Toma de decisiones necesarias para conocer en todo momento la situación en relación con los objetivos establecidos.
- Adopción de las medidas correctoras pertinentes para poner remedio a las desviaciones que se hubieran detectado.
- Proponer, en su caso, modificaciones a los límites u objetivos básicos del proyecto cuando concurren circunstancias que así lo aconsejen.

b) Analista:

- Identificar los requerimientos de información. El analista determina qué información se requiere para poder hacer el análisis.
- Planear las actividades de colección de información. El analista determina cómo obtendrá la información que requiere. En un plan de colección, el analista enumera la información necesaria y su importancia relativa para el análisis, lugar más probable de obtención y como se obtiene.
- Actividades de colección. El analista obtiene la información requerida y determina su utilidad y credibilidad.
- Preparación del producto. El analista verifica su trabajo y lo amolda a las necesidades y requerimientos del usuario. Este producto puede ser un patrón de actividad más reciente, un informe de inteligencia, o cualquier colección de información procesada.
- Diseminación del Producto. El analista comunica el producto de inteligencia a través de resúmenes, informes,...

c) Programador:

- Determinar, en colaboración con el analista, los objetivos perseguidos con los distintos programas, la naturaleza y fuentes de datos que habrá que introducir y ordenar, y establecer los controles necesarios.
- Elaborar gráficos y diagramas para describir y determinar en qué secuencias habrá que proceder al registro y tratamiento de los datos.
- Desarrollar y proporcionar documentación detallada sobre los programas informáticos, utilizando para ello diversos lenguajes de programación.
- Probar los programas elaborados para eliminar o corregir deficiencias o errores.

d) Implantador:

- Puesta en marcha de la aplicación
- Reporte de errores
- Formación de usuarios
- Soporte al cliente

Dados los perfiles que actuan en el desarrollo de la aplicación, podemos considerar la siguiente distribución en función de las tareas:

ANÁLISIS DE COSTES (en euros)			
<i>Perfil</i>	<i>Horas</i>	<i>Precio/hora</i>	<i>Coste total</i>
Jefe de proyecto	40	30	1200
Analista	70	20	1400
Programador	150	12	1800
Implantador	24	15	360
Presupuesto total	284	77	4760

Capítulo 7

Conclusiones

En este trabajo se ha desarrollado un sistema PDM vía web. Para llevar a cabo este desarrollo, en primer lugar se ha realizado el análisis de requerimientos necesario para la creación de cualquier código. A continuación se han escogido el gestor de base de datos, los lenguajes de programación a utilizar y el patrón de diseño para, a partir de éstos, realizar el diseño que ha servido para implementar finalmente el proyecto. Como conclusión del trabajo realizado resaltamos los siguientes puntos:

- Para la gestión de la base de datos ha sido elegida uno de los mejores, sino el mejor, gestor de base de datos, Oracle en su última versión.
- El patrón de diseño escogido ha sido MVC, mediante el cuál quedan completamente independientes las tres capas de una aplicación web (modelo, vista y controlador), provocando con ello mayor flexibilidad y mantenibilidad.
- La programación interna se ha realizado en Java, quizás el lenguaje de programación más en auge de la actualidad.
- Con este proyecto se pretende gestionar la información de un producto, de modo que se mejore la productividad y se aumente la colaboración durante el proceso, hecho que provoca una mayor reutilización de elementos, disminuyendo así la duplicación de información.

Los objetivos propuestos inicialmente se han cumplido satisfactoriamente. No obstante, esta implementación es sólo una solución a los problemas planteados y, en un futuro, sería posible mejorar y añadir nuevas características.

Las posibles continuaciones de este proyecto podrían tratar problemas o mejoras como:

- Incremento del número de usuarios situados geográficamente en zonas distintas: este hecho provocaría que los usuarios situados lejos de los servidores tuvieran un rendimiento bajo debido a que se trata de una aplicación en la que se cargar/descargan ficheros que pueden tener un tamaño extenso. Algunas soluciones posibles a dicha disminución de rendimiento podrían ser:
 - Colocar varios servidores de base de datos con sus respectivos servidores de ficheros en situaciones estratégicas de modo que el rendimiento no baje, siempre realizando un estudio rendimiento/coste. En este caso habría que tener en cuenta la replicación entre las bases de datos para que la información en una y otra base de datos sea la misma en el mínimo periodo de tiempo y sin perjudicar el rendimiento del sistema. En este caso se podrían crear varios ciclos de replicación entre los distintos servidores. Se trata de un proceso complejo que podría implicar problemas de replicación e inconsistencias en la base de datos.
 - Colocar un servidor de base de datos y varios servidores de ficheros repartidos en las situaciones geográficas estratégicas. De este modo la replicación consistiría sólo en copiar ficheros de un servidor de ficheros a otro. Puesto que el mayor tiempo se emplea en la carga o descarga de ficheros, se puede pensar que realizar la búsqueda de la información en un servidor de datos alejado geográficamente y descargar el fichero del servidor local podría ser una opción viable.

- Se podrían implementar algunas operaciones para mejorar la aplicación o, al menos, hacer la vida más fácil al usuario. Por ejemplo, que cada vez que se haga una transferencia de masterdata se envíe un correo electrónico a la persona que lo ha recibido o que cada vez que un usuario quiera realizar un cambio en un documento que no es suyo, pueda pedir al actual propietario mediante un mail la propiedad del documento.
- Se podrían crear conjuntos de usuarios de modo que cada usuario de un conjunto pueda coger la propiedad de un documento para modificarlo, aunque dicho documento pertenezca a otro usuario del mismo grupo. De este modo se reduce el tiempo de desarrollo del documento si hay algún problema con el propietario actual como, por ejemplo, si está de baja. Además, el caso de los conjuntos de usuario no implica ningún problema ya que se supone que entre los usuarios de un mismo conjunto existe cierta comunicación; es decir, se trataría de conjuntos de usuarios que forman un círculo cerrado alrededor de una función específica.
- Podrían crearse también enlaces con programas de CAD, de modo que la información pudiera ser grabada y cargada desde la misma aplicación que utilizan los usuarios para diseñar, si es el caso.
- Encriptación del nombre de los ficheros en el servidor de ficheros de modo que sólo se pueda saber qué documento estas cargando a través de la aplicación.
- Implementación de una interfaz mediante Java Swing.
- ...

Obviamente se puede mejorar la aplicación incrementando sus funcionalidades de modo que se añadan operaciones que puedan ser importantes para los usuarios. Para ello sería imprescindible la participación de los propios usuarios, de tal modo que se produzcan reuniones en períodos de tiempo previamente establecidos para mostrar las posibles mejoras o partes que puedan ser modificadas. Estos cambios o mejoras pueden ser consideradas tareas de mantenimiento de la aplicación y el software desarrollado debe permitir su ampliación sin muchas dificultades debido al patrón MVC por el que se ha optado.

Apéndice A

Casos de uso

Listado de casos de uso

<i>Id</i>	<i>Caso de uso</i>
0.	Validar usuario
1.	Buscar empresa
2.	Buscar grupo
3.	Buscar proyecto
4.	Buscar usuario
5.	Buscar tipo de fichero
6.	Buscar tipo de acl
7.	Buscar acl
8.	Buscar roles
9.	Buscar estados de consulta
10.	Ver cambios de estado de consulta
11.	Buscar estados de fichero
12.	Ver cambios de estado de fichero
13.	Crear empresa
14.	Crear grupo
15.	Crear proyecto
16.	Crear usuario
17.	Crear tipo de fichero
18.	Crear tipo de acl
19.	Crear acl
20.	Crear rol
21.	Crear estados de consulta
22.	Crear cambio de estado de consulta
23.	Crear estados de fichero
24.	Crear cambio de estado de fichero
25.	Cambiar contraseña
26.	Editar perfil de usuario
27.	Editar perfil de empresa
28.	Crear consulta
29.	Crear fichero
30.	Crear asociación
31.	Buscar consulta
32.	Buscar masterdata
33.	Transferir masterdata
34.	Ver ficheros
35.	Aceptar transfer
36.	Cambiar proyecto
37.	Borrar masterdata
38.	Buscar asociación
39.	Ver estructura
40.	Guardar versión asociación
41.	Cambiar versión asociación
42.	Cambiar estado asociación
43.	Descargar asociación
44.	Borrar asociación
45.	Buscar fichero
46.	Guardar versión fichero
47.	Cambiar versión fichero
48.	Cambiar estado fichero
49.	Descargar fichero
50.	Borrar fichero

ID	0		
Nombre	Validar usuario		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	9/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cualquier usuario debe validarse antes de poder utilizar la aplicación
Evento inicial	El caso de uso empieza cuando el actor accede a la página del gestor.
Precondiciones	El actor no debe de estar validado en la aplicación
Flujo principal	P.0.0. El sistema pide NIF y contraseña. Una vez el actor ha introducido los datos, debe pulsar el botón “Acceder”.
Flujos alternativos	
Poscondiciones	P.0.0. Se muestra la pantalla general del gestor.
Excepciones	P.0.0.E.1. Los datos introducidos por el usuario no son correctos: se muestra el mensaje de error “DNI o contraseña incorrectos”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	1		
Nombre	Buscar empresa		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	9/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	Los administradores pueden buscar información de las empresas.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Empresas → Buscar empresa
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.1.0. El sistema pide varios datos de la empresa, decidiendo el actor el criterio de la búsqueda y pulsa el botón “Buscar”: a) NIF b) Nombre c) Dirección d) Ciudad e) Código postal f) Teléfono fijo g) Teléfono móvil h) Horario

	<ul style="list-style-type: none"> i) Contrato en vigor j) Hotline k) Notas <p>P.1.1. Mostrar resultados.</p> <p>P.1.1.1. Ningún criterio encontrado</p> <p>P.1.1.2. Sólo cumple criterio una empresa</p> <p>P.1.1.3. Cumplen criterios varias empresas</p>
Flujos alternativos	<p>A.1.0. Modificar empresa: permite al administrador modificar los datos relacionados con la empresa.</p> <p>A.1.1. Eliminar empresa: permite al administrador eliminar la empresa de la base de datos.</p>
Poscondiciones	<p>P.1.1.2. Se muestra una plantilla con todos los datos de la empresa (los mismos campos que en P.1.0), pudiendo el administrador modificar los datos o eliminar la empresa.</p> <p>P.1.1.3. Se muestra una plantilla con las empresas que cumplen los requisitos y el actor puede elegir la que quiera ver, mostrándose entonces una plantilla del estilo “Sólo cumple criterio una empresa”</p> <p>A.1.0. Se muestra la plantilla “Sólo cumple criterio una empresa” con los datos actualizados.</p> <p>A.1.1. Se muestra una página con las empresas que hay en la base de datos.</p>
Excepciones	<p>P.1.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.1.1.1.E.1. Se muestra el mensaje “No existe ninguna empresa con los datos especificados”.</p> <p>A.1.1.E.1. La empresa está formada por algún usuario. Se muestra el mensaje de error “La empresa tiene uno o más usuarios”</p>
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	2		
Nombre	Buscar grupo		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	9/10/2006	Fecha última actualización	2/12/2006

Actores	Administrador
Descripción	El sistema mantiene la información relacionada con los usuarios, permisos y proyectos mediante el uso de grupos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Grupos → Buscar grupo
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.2.0. El sistema pide el nombre del grupo y pulsa el botón “Buscar”.</p> <p>P.2.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.2.1.1. Ningún grupo encontrado con el criterio dado P.2.1.2. Sólo cumple criterio un grupo P.2.1.3. Cumplen criterios varios grupos
Flujos alternativos	<p>A.2.0. Editar: permite al administrador modificar los datos relacionados con el grupo.</p> <p>A.2.1. Usuarios: permite al administrador añadir usuarios a un grupo. Se muestran dos listas en las que están todos los usuarios y todos los usuarios dentro del grupo. Se pueden añadir y eliminar usuarios. Para terminar la operación, pulsar el botón “Añadir usuarios”.</p> <p>A.2.2. Proyectos: permite al administrador añadir proyectos a un grupo dado un ACL. En primer lugar hay que seleccionar el ACL y una vez escogido se muestran dos listas con todos los proyectos y los proyectos en ese ACL y grupo. Se pueden añadir y eliminar proyectos para ese grupo y ACL. Para terminar la operación, pulsar el botón “Añadir grupo”.</p> <p>A.2.3. Borrar: permite al administrador eliminar el grupo de la base de datos.</p>
Poscondiciones	<p>P.2.1.2. Se muestra una plantilla con todos los datos del grupo:</p> <ul style="list-style-type: none"> a) Id grupo b) Nombre <p>El administrador puede modificar los datos, añadir o eliminar usuarios, añadir o eliminar proyectos del grupo en función de un ACL o eliminar dicho grupo.</p> <p>P.2.1.3. Se muestra una plantilla con los grupos que cumplen los requisitos y el actor puede elegir entre editar, añadir usuarios, añadir proyectos o borrar el grupo.</p> <p>A.2.0. Se muestra la plantilla “Sólo cumple criterio un grupo”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.2.1. Se muestra la plantilla “Sólo cumple criterio un grupo”.</p>

	A.2.2. Se muestra la plantilla “”Sólo cumple criterio un grupo””.
Excepciones	P.2.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.2.1.1.E.1. Se muestra el mensaje “No existe ningún grupo con los datos especificados”. A.2.3.E.1. El grupo tiene alguna relación y no se puede borrar. Se muestra el error “El grupo tiene alguna relación”
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	3		
Nombre	Buscar proyecto		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	10/10/2006	Fecha última actualización	2/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar proyectos, debido a que cada grupo y cada documento está asignado a un proyecto.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Proyectos → Buscar proyecto
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.3.0. El sistema pide el nombre del proyecto decidiendo el actor los criterios de la búsqueda y pulsa el botón “Buscar”: a) Id b) Nombre c) Fecha de creación d) Descripción P.3.1. Mostrar resultados. P.3.1.1. Ningún proyecto encontrado con el criterio dado P.3.1.2. Sólo cumple criterio un proyecto P.3.1.3. Cumplen criterios varios proyectos
Flujos alternativos	A.3.0. Editar: permite al administrador modificar la información relacionada con el proyecto. A.3.1. Grupos: permite al administrador añadir grupos al proyecto con el ACL seleccionado en la página siguiente. Se muestran dos listas en las que están todos los grupos y todos los grupos dentro del proyecto. Se pueden añadir y eliminar grupos. Para terminar la operación, pulsar el botón

	<p>“Añadir grupo”.</p> <p>A.3.2. Borrar: permite al administrador eliminar el proyecto de la base de datos.</p>
Poscondiciones	<p>P.3.1.2. Se muestra una plantilla con todos los datos del proyecto, pudiendo el administrador modificar los datos, añadir grupos en función de un ACL o eliminar el proyecto.</p> <p>P.3.1.3. Se muestra una plantilla con los grupos que cumplen los requisitos y el actor puede elegir entre editar, añadir grupos o borrar el proyecto.</p> <p>A.3.0. Se muestra la plantilla “Sólo cumple criterio un proyecto”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.3.1. Se muestra la plantilla “Sólo cumple criterio un proyecto”.</p> <p>A.3.2. Se muestra la plantilla “Sólo cumple criterio un proyecto”.</p>
Excepciones	<p>P.3.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.3.1.1.E.1. Se muestra el mensaje “No existe ningún proyecto con los datos especificados”.</p> <p>A.3.2.E.1. El proyecto tiene alguna relación y no se puede borrar. Se muestra el error “El proyecto tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	4		
Nombre	Buscar usuario		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	10/10/2006	Fecha última actualización	3/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar usuarios, debido a que la mayoría de elementos de la base de datos son propiedad de un usuario.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Usuarios → Buscar usuario
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.4.0. El sistema pide una serie de datos, decidiendo el actor los criterios de la búsqueda y pulsa el botón “Buscar”:

	<ul style="list-style-type: none"> a) DNI b) Nombre c) Primer apellido d) Segundo Apellido e) Teléfono fijo f) Teléfono móvil g) Dirección de correo h) Empresa <p>P.4.1. Mostrar resultados.</p> <p>P.4.1.1. Ningún usuario encontrado con el criterio dado</p> <p>P.4.1.2. Sólo cumple criterio un usuario</p> <p>P.4.1.3. Cumplen criterios varios usuarios</p>
Flujos alternativos	<p>A.4.0. Editar: permite al administrador modificar la información relacionada con el usuario.</p> <p>A.4.1. Grupos: permite al administrador añadir el usuario a uno o varios grupos. Se muestran dos listas en las que están todos los grupos y todos los grupos a los que pertenece el usuario. Se puede añadir y eliminar el usuario a uno o más grupos. Para terminar la operación, pulsar el botón “Añadir”.</p> <p>A.4.2. Roles: permite al administrador añadir el usuario a uno o varios roles. En la siguiente página se pide al administrador que seleccione el proyecto y acl que relacionará con el usuario y rol. Se muestran dos listas en las que están todos los roles y todos los roles a los que pertenece el usuario dados proyecto y acl. Se puede añadir y eliminar el usuario a uno o más roles. Para terminar la operación, pulsar el botón “Añadir roles”.</p> <p>A.4.3. Borrar: permite al administrador eliminar el usuario de la base de datos.</p>
Poscondiciones	<p>P.4.1.2. Se muestra una plantilla con todos los datos del usuario, pudiendo el administrador modificar los datos, añadirlo a grupos o roles en función de un ACL o eliminar el usuario.</p> <p>P.4.1.3. Se muestra una plantilla con los usuarios que cumplen los requisitos y el actor puede elegir entre editar, añadirlo a grupos o roles en función de un ACL o eliminar el usuario.</p> <p>A.4.0. Se muestra la plantilla “Sólo cumple criterio un usuario”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.4.1. Se muestra la plantilla “Sólo cumple criterio un usuario”.</p> <p>A.4.3. Se muestra la plantilla “Sólo cumple criterio un usuario”.</p>
Excepciones	<p>P.4.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.4.1.1.E.1. Se muestra el mensaje “No existe ningún proyecto con los datos especificados”.</p>

	A.4.3.E.1. El usuario tiene alguna relación y no se puede borrar. Se muestra el error “El usuario tiene alguna relación”
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	5		
Nombre	Buscar tipo de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	10/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los tipos de ficheros que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Tipos de fichero → Buscar tipo de fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.5.0. El sistema pide una serie de datos, decidiendo el actor los criterios de la búsqueda y pulsa el botón “Buscar”:</p> <ul style="list-style-type: none"> a) Extensión b) Programa <p>P.5.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.5.1.1. Ningún tipo de fichero encontrado con el criterio dado P.5.1.2. Sólo cumple criterio un tipo de fichero P.5.1.3. Cumplen criterios varios tipos de fichero
Flujos alternativos	<p>A.5.0. Editar: permite al administrador modificar la información relacionada con el tipo de fichero.</p> <p>A.5.1. Borrar: permite al administrador eliminar el tipo de fichero de la base de datos.</p>
Poscondiciones	<p>P.5.1.2. Se muestra una plantilla con todos los datos del tipo de fichero, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.5.1.3. Se muestra una plantilla con los tipos de fichero que cumplen los requisitos y el actor puede elegir entre editar o eliminar.</p> <p>A.5.0. Se muestra la plantilla “Sólo cumple criterio un tipo de fichero”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p>

Excepciones	<p>A.5.1. Se muestra la plantilla “Sólo cumple criterio un tipo de fichero”.</p> <p>P.5.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.5.1.1.E.1. Se muestra el mensaje “No existe ningún tipo de fichero con los datos especificados”.</p> <p>A.5.1.E.1. El tipo de fichero tiene alguna relación y no se puede borrar. Se muestra el error “El tipo de fichero tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	6		
Nombre	Buscar tipo de acl		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	10/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los tipos de acl que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → ACLs → Buscar tipo de ACL
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.6.0. El actor introduce el nombre del tipo de ACL y pulsa el botón “Buscar”.</p> <p>P.6.1. Mostrar resultados.</p> <p>P.6.1.1. Ningún tipo de acl encontrado con el criterio dado</p> <p>P.6.1.2. Sólo cumple criterio un tipo de acl</p> <p>P.6.1.3. Cumplen criterios varios tipos de acl</p>
Flujos alternativos	<p>A.6.0. Editar: permite al administrador modificar la información relacionada con el tipo de acl.</p> <p>A.6.1. Borrar: permite al administrador eliminar el tipo de acl de la base de datos.</p>
Poscondiciones	<p>P.6.1.2. Se muestra una plantilla con todos los datos del tipo de acl, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.6.1.3. Se muestra una plantilla con los tipos de acl que cumplen los requisitos y el actor puede elegir entre editar o eliminar.</p>

	A.6.0. Se muestra la plantilla “Sólo cumple criterio un tipo de acl”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.6.1. Se muestra la plantilla “Sólo cumple criterio un tipo de acl”.
Excepciones	P.6.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.6.1.1.E.1. Se muestra el mensaje “No existe ningún tipo de acl con los datos especificados”. A.6.1.E.1. El tipo de acl tiene alguna relación y no se puede borrar. Se muestra el error “El tipo de acl tiene alguna relación”
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	7		
Nombre	Buscar acl		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los acl que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → ACLs → Buscar ACL
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.7.0. El actor introduce los datos del ACL a partir de los cuáles quiere realizar la búsqueda y pulsa el botón “Buscar”: a) Nombre b) Tipo de acl c) Estado d) Es propietario e) Crear proyecto f) Modificar proyecto g) Borrar proyecto h) Leer Masterdata i) Modificar masterdata j) Borrar masterdata k) Cambiar proyecto

	<ul style="list-style-type: none"> l) Leer fichero m) Escribir fichero n) Modificar fichero o) Borrar fichero p) Cambiar estado a q) Cambiar estado desde r) Reservar <p>P.7.1. Mostrar resultados.</p> <p>P.7.1.1. Ningún acl encontrado con el criterio dado</p> <p>P.7.1.2. Sólo cumple criterio un acl</p> <p>P.7.1.3. Cumplen criterios varios acl</p>
Flujos alternativos	<p>A.7.0. Editar: permite al administrador modificar la información relacionada con el acl.</p> <p>A.7.1. Borrar: permite al administrador eliminar el acl de la base de datos.</p>
Poscondiciones	<p>P.7.1.2. Se muestra una plantilla con todos los datos del acl, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.7.1.3. Se muestra una plantilla con los acl que cumplen los requisitos y el actor puede elegir entre editar o eliminar.</p> <p>A.7.0. Se muestra la plantilla “Sólo cumple criterio un acl”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.7.1. Se muestra la plantilla “Sólo cumple criterio un acl”.</p>
Excepciones	<p>P.7.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.7.1.1.E.1. Se muestra el mensaje “No existe ningún acl con los datos especificados”.</p> <p>A.7.1.E.1. El acl tiene alguna relación y no se puede borrar. Se muestra el error “El acl tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	8		
Nombre	Buscar roles		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	3/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los roles que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Roles → Buscar rol
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.8.0. El actor introduce los datos del rol a partir de los cuáles quiere realizar la búsqueda y pulsa el botón “Buscar”:</p> <ul style="list-style-type: none"> a) Nombre b) Descripción <p>P.8.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.8.1.1. Ningún rol encontrado con el criterio dado P.8.1.2. Sólo cumple criterio un rol P.8.1.3. Cumplen criterios varios roles
Flujos alternativos	<p>A.8.0. Editar: permite al administrador modificar la información relacionada con el rol.</p> <p>A.8.1. Usuarios: permite al administrador añadir o eliminar usuarios del rol en función del proyecto y acl. En la siguiente pantalla se muestran dos listas desplegables para seleccionar un proyecto y un acl. A continuación se muestran, opr una parte, todos los usuarios, y por la otra los usuarios que en este momento están dentro de dicho rol con el proyecto y acl dados. En ese momento se pueden añadir o eliminar los usuarios.</p> <p>A.8.2. Borrar: permite al administrador eliminar el rol de la base de datos.</p>
Poscondiciones	<p>P.8.1.2. Se muestra una plantilla con todos los datos del rol, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.8.1.3. Se muestra una plantilla con los roles que cumplen los requisitos y el actor puede elegir entre editar, añadir usuarios al rol o eliminar.</p> <p>A.8.0. Se muestra la plantilla “Sólo cumple criterio un rol”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.8.1. Se muestra la plantilla “Sólo cumple criterio un rol”.</p> <p>A.8.2. Se muestra la plantilla “Sólo cumple criterio un rol”.</p>
Excepciones	<p>P.8.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.8.1.1.E.1. Se muestra el mensaje “No existe ningún rol con los datos especificados”.</p> <p>A.8.2.E.1. El rol tiene alguna relación y no se puede borrar. Se muestra el</p>

	error “El rol tiene alguna relación”
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	9		
Nombre	Buscar estados de consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los estados posibles que puede tener cada consulta
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Consultas → Buscar estado consulta
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.9.0. El actor introduce el nombre del estado y pulsa el botón “Buscar”.</p> <p>P.9.1. Mostrar resultados.</p> <p>P.9.1.1. Ningún estado encontrado con el criterio dado</p> <p>P.9.1.2. Sólo cumple criterio un estado</p> <p>P.9.1.3. Cumplen criterios varios estados</p>
Flujos alternativos	<p>A.9.0. Editar: permite al administrador modificar la información relacionada con el estado.</p> <p>A.9.1. Borrar: permite al administrador eliminar el estado de la base de datos.</p>
Poscondiciones	<p>P.9.1.2. Se muestra una plantilla con todos los datos del estado, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.9.1.3. Se muestra una plantilla con los estados que cumplen los requisitos y el actor puede elegir entre editar o eliminar.</p> <p>A.9.0. Se muestra la plantilla “Sólo cumple criterio un estado”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.9.1. Se muestra la plantilla “Sólo cumple criterio un estado”.</p>
Excepciones	<p>P.9.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.9.1.1.E.1. Se muestra el mensaje “No existe ningún estado con los datos</p>

	especificados”.
	A.9.1.E.1. El estado tiene alguna relación y no se puede borrar. Se muestra el error “El estado tiene alguna relación”
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	10		
Nombre	Ver cambios de estado de consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los cambios de estado posibles que puede tener cada consulta
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Consultas → Ver cambios de estado
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.10.0. El sistema muestra todos los cambios de estado que puede tener cada consulta y permite eliminarla.
Flujos alternativos	A.10.0. Borrar: permite al administrador eliminar el cambio de estado.
Poscondiciones	A.10.0. Se muestran los cambios de estado que hay disponibles tras eliminar la seleccionada.
Excepciones	P.10.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	11		
Nombre	Buscar estados de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los estados posibles que puede tener cada fichero
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Ficheros → Buscar estado fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.11.0. El actor introduce el nombre del estado y pulsa el botón “Buscar”.</p> <p>P.11.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.11.1.1. Ningún estado encontrado con el criterio dado P.11.1.2. Sólo cumple criterio un estado P.11.1.3. Cumplen criterios varios estados
Flujos alternativos	<p>A.11.0. Editar: permite al administrador modificar la información relacionada con el estado.</p> <p>A.11.1. Borrar: permite al administrador eliminar el estado de la base de datos.</p>
Poscondiciones	<p>P.11.1.2. Se muestra una plantilla con todos los datos del estado, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>P.11.1.3. Se muestra una plantilla con los estados que cumplen los requisitos y el actor puede elegir entre editar o eliminar.</p> <p>A.11.0. Se muestra la plantilla “Sólo cumple criterio un estado”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.11.1. Se muestra la plantilla “Sólo cumple criterio un estado”.</p>
Excepciones	<p>P.11.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.11.1.1.E.1. Se muestra el mensaje “No existe ningún estado con los datos especificados”.</p> <p>A.11.1.E.1. El estado tiene alguna relación y no se puede borrar. Se muestra el error “El estado tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	12		
Nombre	Ver cambios de estado de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	11/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los cambios de estado posibles que puede tener cada fichero
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Ficheros → Ver cambios de estado
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.12.0. El sistema muestra todos los cambios de estado que puede tener cada fichero y permite eliminarla.
Flujos alternativos	A.12.0. Borrar: permite al administrador eliminar el cambio de estado.
Poscondiciones	A.12.0. Se muestran los cambios de estado que hay disponibles tras eliminar la seleccionada.
Excepciones	P.12.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”
Casos de uso incluidos	
Prioridad	Baja
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	13		
Nombre	Crear empresa		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	12/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	Los administradores pueden crear información de las empresas que tendrán usuarios que accedan al gestor.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Empresas → Crear empresa
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.13.0. El sistema pide varios datos de la empresa (los mismos que en el caso de uso 1.Buscar empresa). Una vez añadidos, el actor debe pulsar el botón “Crear”
Flujos alternativos	A.13.0. Modificar empresa: permite al administrador modificar los datos relacionados con la empresa.

	A.13.1. Eliminar empresa: permite al administrador eliminar la empresa de la base de datos.
Poscondiciones	<p>P.13.0.Se muestra una plantilla con todos los datos de la empresa, pudiendo el administrador modificar los datos o eliminar la empresa.</p> <p>A.13.0. Se muestran la misma plantilla que en P.1.0 con los datos actualizados.</p> <p>A.13.1. Se muestra una página con las empresas que hay en la base de datos.</p>
Excepciones	<p>P.13.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.13.0.E.2. El nif de la empresa ya existe: se muestra el mensaje de error “El nif de la empresa ya existe”</p> <p>P.13.0.E.3. El nombre de la empresa ya existe: se muestra el mensaje de error “El nombre ya está activo para una empresa”</p> <p>A.13.1.E.1. La empresa está formada por algún usuario. Se muestra el mensaje de error “La empresa tiene uno o más usuarios”</p>
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	14		
Nombre	Crear grupo		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	12/10/2006	Fecha última actualización	4/12/2006

Actores	Administrador
Descripción	El sistema mantiene la información relacionada con los usuarios, permisos y proyectos mediante el uso de grupos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Grupos → Crear grupo
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.14.0. El sistema pide el nombre del grupo y pulsa el botón “Crear”.
Flujos alternativos	<p>A.14.0. Editar: permite al administrador modificar los datos relacionados con el grupo.</p> <p>A.14.1. Usuarios: permite al administrador añadir usuarios a un grupo. Se muestran dos listas en las que están todos los usuarios y todos los usuarios</p>

	<p>dentro del grupo. Se pueden añadir y eliminar usuarios. Para terminar la operación, pulsar el botón “Añadir usuarios”.</p> <p>A.14.2. Proyectos: permite al administrador añadir proyectos a un grupo dado un ACL. En primer lugar hay que seleccionar el ACL y una vez escogido se muestran dos listas con todos los proyectos y los proyectos en ese ACL y grupo. Se pueden añadir y eliminar proyectos para ese grupo y ACL. Para terminar la operación, pulsar el botón “Añadir grupo”.</p> <p>A.14.3. Borrar: permite al administrador eliminar el grupo de la base de datos.</p>
Poscondiciones	<p>P.14.0. Se muestra una plantilla con todos los datos del grupo, pudiendo el administrador modificar los datos, añadir o eliminar usuarios, añadir o eliminar proyectos del grupo en función de un ACL o eliminar dicho grupo.</p> <p>A.14.0. Se muestra la plantilla con los datos del grupo. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.14.1. Se muestra la plantilla con los datos del grupo.</p> <p>A.14.2. Se muestra la plantilla con los datos del grupo.</p>
Excepciones	<p>P.14.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.14.0.E.2. El grupo ya existe: se muestra el mensaje de error “El grupo ya existe”</p> <p>A.14.3.E.1. El grupo tiene alguna relación y no se puede borrar. Se muestra el error “El grupo tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	15		
Nombre	Crear proyecto		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	12/10/2006	Fecha última actualización	4/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear proyectos, debido a que cada grupo y cada documento está asignado a un proyecto.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Proyectos → Crear proyecto
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.15.0. El sistema pide los datos del proyecto y una vez introducidos el actor debe pulsar el botón “Crear”: a) Nombre b) Descripción
Flujos alternativos	A.15.0. Editar: permite al administrador modificar la información relacionada con el proyecto. A.15.1. Grupos: permite al administrador añadir grupos al proyecto con el ACL seleccionado en la página siguiente. Se muestran dos listas en las que están todos los grupos y todos los grupos dentro del proyecto. Se pueden añadir y eliminar grupos. Para terminar la operación, pulsar el botón “Añadir grupo”. A.15.2. Borrar: permite al administrador eliminar el proyecto de la base de datos.
Poscondiciones	P.15.0. Se muestra una plantilla con todos los datos del proyecto, pudiendo el administrador modificar los datos, añadir grupos en función de un ACL o eliminar el proyecto. A.15.0. Se muestra la plantilla con los datos del proyecto. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.15.1. Se muestra la plantilla con los datos del proyecto. A.15.2. Se muestra la plantilla con los datos del proyecto.
Excepciones	P.15.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.15.0.E.2. El proyecto ya existe: se muestra el mensaje de error “El proyecto ya existe” A.15.2.E.1. El proyecto tiene alguna relación y no se puede borrar. Se muestra el error “El proyecto tiene alguna relación”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas

	operaciones
Requisitos no funcionales	
Comentarios	

ID	16		
Nombre	Crear usuario		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	4/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear usuarios, debido a que la mayoría de elementos de la base de datos son propiedad de un usuario.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Usuarios → Crear usuario
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.16.0. El sistema pide una serie de datos (los mismos que en el caso de uso 4.Buscar usuario). Una vez introducidos, el actor pulsa el botón “Crear”.
Flujos alternativos	<p>A.16.0. Editar: permite al administrador modificar la información relacionada con el usuario.</p> <p>A.16.1. Grupos: permite al administrador añadir el usuario a uno o varios grupos. Se muestran dos listas en las que están todos los grupos y todos los grupos a los que pertenece el usuario. Se puede añadir y eliminar el usuario a uno o más grupos. Para terminar la operación, pulsar el botón “Añadir”.</p> <p>A.16.2. Roles: permite al administrador añadir el usuario a uno o varios roles. En la siguiente página se pide al administrador que seleccione el proyecto y acl que relacionará con el usuario y rol. Se muestran dos listas en las que están todos los roles y todos los roles a los que pertenece el usuario dados proyecto y acl. Se puede añadir y eliminar el usuario a uno o más roles. Para terminar la operación, pulsar el botón “Añadir roles”.</p> <p>A.16.3. Borrar: permite al administrador eliminar el usuario de la base de datos.</p>
Poscondiciones	<p>P.16.0. Se muestra una plantilla con todos los datos del usuario, pudiendo el administrador modificar los datos, añadirlo a grupos o roles en función de un ACL o eliminar el usuario.</p> <p>A.16.0. Se muestra la plantilla con los datos del usuario. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.16.1. Se muestra la plantilla con los datos del usuario.</p> <p>A.16.2. Se muestra la plantilla con los datos del usuario.</p>

	A.16.3. Se muestra la plantilla con los datos del usuario.
Excepciones	<p>P.16.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”</p> <p>P.16.0.E.2. El usuario ya existe: se muestra el mensaje de error “El dni del usuario ya existe” o “El mail ya está activo para un usuario”</p> <p>A.16.3.E.1. El usuario tiene alguna relación y no se puede borrar. Se muestra el error “El usuario tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Alta
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	17		
Nombre	Crear tipo de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los tipos de ficheros que hay definidos en la base de datos. También se crean automáticamente cuando un usuario graba un fichero con una extensión que no existe.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Tipos de fichero → Crear tipo de fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.17.0. El sistema pide una serie de datos (los mismos que en el caso de uso 5.Buscar tipo de fichero). Una vez introducidos el actor pulsa el botón “Crear”.
Flujos alternativos	<p>A.17.0. Editar: permite al administrador modificar la información relacionada con el tipo de fichero.</p> <p>A.17.1. Borrar: permite al administrador eliminar el tipo de fichero de la base de datos.</p>
Poscondiciones	<p>P.17.0. Se muestra una plantilla con todos los datos del tipo de fichero, pudiendo el administrador modificar los datos o eliminarlo.</p> <p>A.17.0. Se muestra la plantilla con los datos del tipo de fichero. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados.</p> <p>A.17.1. Se muestra la plantilla con los datos del tipo de fichero.</p>
Excepciones	P.17.0.E.1. El usuario no es administrador: se muestra el mensaje de error

	<p>“Es necesario ser administrador”</p> <p>P.17.0.E.2. El tipo de fichero ya existe: se muestra el mensaje de error “El tipo de fichero ya existe”</p> <p>A.17.1.E.1. El tipo de fichero tiene alguna relación y no se puede borrar. Se muestra el error “El tipo de fichero tiene alguna relación”</p>
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Media
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	18		
Nombre	Crear tipo de acl		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los tipos de acl que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → ACLs → Crear tipo de ACL
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.18.0. El actor introduce el nombre del ACL y pulsa el botón “Crear”. A.18.0. Editar: permite al administrador modificar la información relacionada con el tipo de acl. A.18.1. Borrar: permite al administrador eliminar el tipo de acl de la base de datos.
Flujos alternativos	
Poscondiciones	P.18.0. Se muestra una plantilla con todos los datos del tipo de acl, pudiendo el administrador modificar los datos o eliminarlo. A.18.0. Se muestra la plantilla “Sólo cumple criterio un tipo de acl”. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.18.1. Se muestra la plantilla “Sólo cumple criterio un tipo de acl”.
Excepciones	P.18.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.18.0.E.2. El tipo de acl ya existe: se muestra el mensaje de error “El tipo de acl ya existe”

	A.18.1.E.1. El tipo de acl tiene alguna relación y no se puede borrar. Se muestra el error “El tipo de acl tiene alguna relación”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	19		
Nombre	Crear acl		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los acl que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → ACLs → Crear ACL
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.19.0. El actor introduce los datos del ACL (los mismos que en laco de uso 7.Buscar acl) y pulsa el botón “Crear”. A.19.0. Editar: permite al administrador modificar la información relacionada con el acl. A.19.1. Borrar: permite al administrador eliminar el acl de la base de datos.
Poscondiciones	P.19.0. Se muestra una plantilla con todos los datos del acl, pudiendo el administrador modificar los datos o eliminarlo. A.19.0. Se muestra la plantilla con los datos del acl. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.19.1. Se muestra la plantilla con los datos del acl.
Excepciones	P.19.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.19.0.E.2. El usuario no ha seleccionado tipo de acl o estado: se muestra el mensaje de error “Es necesario seleccionar los campos tipo acl y estado” P.19.0.E.3. El acl ya existe: se muestra el mensaje de error “Es acl ya existe” A.19.1.E.1. El acl tiene alguna relación y no se puede borrar. Se muestra el error “El acl tiene alguna relación”

Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	20		
Nombre	Crear rol		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	5/12/2006

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de buscar los roles que hay definidos en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Roles → Buscar rol
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.20.0. El actor introduce los datos del rol a partir de los cuáles quiere realizar la búsqueda y pulsa el botón “Buscar”: a) Nombre b) Descripción
Flujos alternativos	A.20.0. Editar: permite al administrador modificar la información relacionada con el rol. A.20.1. Usuarios: permite al administrador añadir o eliminar usuarios del rol en función del proyecto y acl. En la siguiente pantalla se muestran dos listas desplegables para seleccionar un proyecto y un acl. A continuación se muestran, por una parte, todos los usuarios, y por la otra los usuarios que en este momento están dentro de dicho rol con el proyecto y acl dados. En ese momento se pueden añadir o eliminar los usuarios. A.20.2. Borrar: permite al administrador eliminar el rol de la base de datos.
Poscondiciones	P.20.0. Se muestra una plantilla con todos los datos del rol, pudiendo el administrador modificar los datos o eliminarlo. A.20.0. Se muestra la plantilla con los datos del rol. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.20.1. Se muestra la plantilla con los datos del rol. A.20.2. Se muestra la plantilla con los datos del rol.
Excepciones	P.20.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador”

	P.20.0.E.2. El rol ya existe: se muestra el mensaje de error “El rol ya existe” A.20.2.E.1. El rol tiene alguna relación y no se puede borrar. Se muestra el error “El rol tiene alguna relación”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	21		
Nombre	Crear estados de consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	13/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los estados posibles que puede tener cada consulta
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Consultas → Crear estado consulta
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.21.0. El actor introduce el nombre del estado y pulsa el botón “Crear”. A.21.0. Editar: permite al administrador modificar la información relacionada con el estado. A.21.1. Borrar: permite al administrador eliminar el estado de la base de datos.
Flujos alternativos	
Poscondiciones	P.21.0. Se muestra una plantilla con todos los datos del estado, pudiendo el administrador modificar los datos o eliminarlo. A.21.0. Se muestra la plantilla con los datos del estado. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.21.1. Se muestra la plantilla con los datos del estado.
Excepciones	P.21.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.21.0.E.2. El estado ya existe: se muestra el mensaje de error “El estado ya existe” A.21.1.E.1. El estado tiene alguna relación y no se puede borrar. Se muestra el error “El estado tiene alguna relación”
Casos de uso incluidos	

Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	22		
Nombre	Crear cambio de estado de consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	14/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los cambios de estado posibles que puede tener cada consulta. De este modo se define, dado un estado, a qué estados puede cambiar.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Consultas → Crear cambios de estado
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.22.0. El sistema muestra dos listas desplegables con todos los estados en cada una de ellas. El actor debe escoger un estado en cada una de las listas desplegables y pulsar el botón “Crear”
Flujos alternativos	A.22.0. Borrar: permite al administrador eliminar el cambio de estado.
Poscondiciones	A.22.0. Se muestran los cambios de estado que hay disponibles tras eliminar la seleccionada.
Excepciones	P.22.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.22.0.E.2. El cambio de estado ya existe: se muestra el mensaje de error “El cambio de estado ya existe”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	23		
Nombre	Crear estados de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	14/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los estados posibles que puede tener cada fichero
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Ficheros → Crear estado fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.23.0. El actor introduce el nombre del estado y pulsa el botón “Crear”.
Flujos alternativos	A.23.0. Editar: permite al administrador modificar la información relacionada con el estado. A.23.1. Borrar: permite al administrador eliminar el estado de la base de datos.
Poscondiciones	P.23.0. Se muestra una plantilla con todos los datos del estado, pudiendo el administrador modificar los datos o eliminarlo. A.23.0. Se muestra la plantilla con los datos del estado. Si se modifican los datos se vuelve a mostrar la misma plantilla con los datos actualizados. A.23.1. Se muestra la plantilla con los datos del estado.
Excepciones	P.23.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.23.0.E.2. El estado ya existe: se muestra el mensaje de error “El estado ya existe” A.23.1.E.1. El estado tiene alguna relación y no se puede borrar. Se muestra el error “El estado tiene alguna relación”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	24		
Nombre	Crear cambio de estado de fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	14/10/2006	Fecha última actualización	

Actores	Administrador
Descripción	El sistema ofrece la posibilidad de crear los cambios de estado posibles que puede tener cada fichero. De este modo se define, dado un estado, a qué estados puede cambiar.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Admin → Estados → Ficheros → Crear cambios de estado
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.24.0. El sistema muestra dos listas desplegables con todos los estados en cada una de ellas. El actor debe escoger un estado en cada una de las listas desplegables y pulsar el botón “Crear”
Flujos alternativos	A.24.0. Borrar: permite al administrador eliminar el cambio de estado.
Poscondiciones	A.24.0. Se muestran los cambios de estado que hay disponibles tras eliminar la seleccionada.
Excepciones	P.24.0.E.1. El usuario no es administrador: se muestra el mensaje de error “Es necesario ser administrador” P.24.0.E.2. El cambio de estado ya existe: se muestra el mensaje de error “El cambio de estado ya existe”
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	25		
Nombre	Cambiar contraseña		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	18/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema le ofrece la posibilidad a todos los usuarios de cambiar la contraseña de acceso.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Editar → Cambiar contraseña
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.25.0. El sistema muestra tres campos: a) Antigua contraseña b) Nueva contraseña

	c) Repetición de la nueva contraseña. Tras introducir los tres campos el usuario debe pulsar el botón “Cambiar”.
Flujos alternativos	
Poscondiciones	A.25.0. Se muestra la página general.
Excepciones	P.25.0.E.1. Los dos campos que definen la nueva contraseña no son iguales. Se muestra el error “Los campos nueva contraseña y repetir contraseña deben ser iguales”. P.25.0.E.2. La antigua contraseña es la misma que la nueva: se muestra el mensaje de error “No se puede repetir la última contraseña” P.25.0.E.3. La antigua contraseña introducida por el usuario no es correcta: se muestra el mensaje de error “La contraseña actual no es correcta”
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	26		
Nombre	Editar perfil de usuario		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	18/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema le ofrece la posibilidad a todos los usuarios de modificar algunos de los datos que le identifican como usuario.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Editar → Perfil de usuario
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.26.0. El sistema muestra la información del usuario que está actualmente en sesión. a) DNI: no modificable b) Nombre c) Primer apellido d) Segundo apellido e) Teléfono fijo f) Teléfono móvil g) Dirección de correo h) Empresa: no modificable Tras realizar las modificaciones el usuario debe pulsar el botón “Cambiar”.
Flujos alternativos	
Poscondiciones	A.26.0. Se muestra la página con los datos del usuario actualizados.

Excepciones	
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	27		
Nombre	Editar perfil de empresa		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	18/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema le ofrece la posibilidad a todos los usuarios de modificar algunos de los datos de la empresa a la que pertenecen.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Editar → Perfil de empresa
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.27.0. El sistema muestra la información del usuario que está actualmente en sesión.</p> <ul style="list-style-type: none"> a) NIF: no modificable b) Nombre c) Dirección d) Ciudad e) Código postal f) Teléfono fijo g) Teléfono móvil h) Horario i) Contrato en vigor: no modificable j) Hotline: no modificable k) Notas <p>Tras realizar las modificaciones el usuario debe pulsar el botón “Cambiar”.</p>
Flujos alternativos	
Poscondiciones	P.27.0. Se muestra la página con los datos de la empresa, a la que pertenece el usuario conectado, actualizados.
Excepciones	
Casos de uso incluidos	
Prioridad	Media
Frecuencia de uso	Baja
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	28		
Nombre	Crear consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	19/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de crear consultas relacionadas con cualquier duda que tengan.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Crear → Consulta
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.28.0. El sistema muestra los campos que debe llenar el actor: a) Título b) Proyecto c) Pregunta Una vez ha introducido los datos debe pulsar el botón “Crear”.
Flujos alternativos	
Poscondiciones	A.28.0. Se muestra la página de entrada al gestor.
Excepciones	P.28.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto” P.28.0.E.2. La consulta ya existe. El sistema muestra el mensaje de error: “La consulta ya existe”.
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	29		
Nombre	Crear fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	19/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de guardar ficheros en la base de datos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Crear → Fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.29.0. El sistema muestra los campos que debe llenar el actor: a) Proyecto b) Descripción

	c) Fichero Una vez ha introducido los datos y seleccionado el fichero que quiere guardar, debe pulsar el botón “Crear”.
Flujos alternativos	
Poscondiciones	A.29.0. Se muestra la página de entrada al gestor.
Excepciones	P.29.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto” P.29.0.E.2. El usuario no ha escogido ningún fichero. El sistema muestra el mensaje de error: “Falta escoger el fichero”. P.29.0.E.3. El fichero ya existe. El sistema muestra el mensaje de error: “El fichero ya existe”. P.29.0.E.4. El usuario no tiene permisos para crear el archivo. El sistema muestra el mensaje de error: “No tiene suficientes permisos”. P.29.0.E.5. El fichero tiene un nombre con una longitud que excede del máximo. El sistema muestra el mensaje de error: “Nombre demasiado largo”.
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	30		
Nombre	Crear asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	20/10/2006	Fecha última actualización	6/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de crear asociaciones, las cuales estarán compuestas por ficheros o por otras asociaciones.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Crear → Asociación
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.30.0. El sistema muestra los campos que debe rellenar el actor: a) Nombre b) Proyecto c) Descripción Una vez ha introducido los datos tiene que buscar los masterdatas que se añadirán a la asociación. Para ello debe pulsar el botón “Buscar”

	<p>masterdatas”.</p> <p>P.30.1. Se muestra la página para realizar la búsqueda de masterdatas a añadir:</p> <ul style="list-style-type: none"> a) Nombre b) Propietario c) Proyecto d) Descripción <p>P.30.2. Se crea la asociación con los datos escogidos a través de los flujos alternativos. Se debe pulsar el botón “Crear” tras el flujo A.30.0.</p>
Flujos alternativos	<p>A.30.0. El usuario puede seleccionar los masterdatas que quiere añadir a la asociación. Para ello debe pulsar el botón “Seleccionar”</p> <p>A.30.1. El usuario puede seleccionar varios masterdata y realizar otra búsqueda para añadir más elementos a la asociación. Para ello debe pulsar el botón “Seleccionar y buscar”</p> <p>A.30.2. Si el usuario no obtiene los resultados esperados en la búsqueda, puede realizar una nueva búsqueda. Para ello debe pulsar el botón “Nueva búsqueda”</p> <p>A.30.3. Si el usuario quiere empezar a añadir masterdatas a la asociación, puede empezar una nueva búsqueda. Para ello debe pulsar el botón “Empezar búsqueda”</p>
Poscondiciones	<p>P.30.0. Se muestra la página para realizar la búsqueda de masterdatas.</p> <p>P.30.1. Se muestran los resultados de la búsqueda de masterdatas.</p> <p>A.30.0. Se muestra la página con los datos de la asociación y los masterdatas que se han escogido:</p> <ul style="list-style-type: none"> a) Nombre b) Proyecto c) Descripción d) Ficheros escogidos <p>A.30.1. Se muestra la página para realizar otra búsqueda de masterdatas.</p> <p>A.30.2. Se muestra la página para realizar otra vez la búsqueda de masterdatas.</p> <p>A.30.3. Se muestra la página para realizar la búsqueda de masterdatas.</p>
Excepciones	<p>P.30.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”</p> <p>P.30.0.E.2. El usuario no ha escogido ningún fichero. El sistema muestra el mensaje de error: “Falta escoger el fichero”.</p> <p>P.30.2.E.1. El masterdata ya existe. Se muestra el mensaje de error “Existe masterdata, asociación o fichero con ese nombre”</p> <p>P.30.2.E.2. El actor no tiene permisos para guardar la asociación. Se muestra el mensaje de error “No tiene permisos para crear la asociación”</p>

Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	Sólo los administradores de la base de datos pueden realizar estas operaciones
Requisitos no funcionales	
Comentarios	

ID	31		
Nombre	Buscar consulta		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	20/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de buscar consultas. En función del estado en el que esté la consulta y del propietario, se podrán realizar unas u otras operaciones.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Buscar → Consulta
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.31.0. El sistema muestra los campos que puede llenar el actor:</p> <ul style="list-style-type: none"> a) ID consulta b) Título c) Propietario d) Proyecto e) Pregunta f) Respuesta g) Estado <p>Una vez ha introducido los criterios de búsqueda debe pulsar el botón “Buscar”.</p> <p>P.31.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.31.1.1. Ningún criterio encontrado P.31.1.2. Sólo cumple criterio una consulta P.31.1.3. Cumplen criterios varias consultas
Flujos alternativos	<p>A.31.0. El usuario puede modificar la consulta.</p> <p>A.31.1. El usuario puede eliminar la consulta.</p>
Poscondiciones	<p>P.31.1.2. Se muestra una plantilla con todos los datos de la consulta (los mismos que al realizar la búsqueda, pero añadiendo los campos Fecha pregunta y Fecha respuesta), pudiendo el propietario modificar la pregunta o un administrador contestarla, siempre y cuando no esté cerrada.</p> <p>P.31.1.3. Se muestra una plantilla con las consultas que cumplen los requisitos y el actor puede elegir la que quiera ver, mostrándose entonces</p>

	<p>una plantilla del estilo “Sólo cumple criterio una consulta”</p> <p>A.31.0. Se muestra la plantilla “Sólo cumple criterio una consulta” con los datos actualizados.</p> <p>A.31.1. Se muestra la página general del gestor.</p>
Excepciones	<p>P.31.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”</p> <p>P.31.1.1.E.1. El sistema no ha encontrado ninguna consulta con la información que ha dado el actor. Se muestra el mensaje de error “No existe ninguna consulta con los datos especificados”.</p> <p>P.31.1.2.E.1. El usuario no es el propietario ni administrador o la consulta esta cerrada. Se muestra el mensaje de error “Se necesitan permisos para actualizar datos o no se puede modificar una consulta cerrada”.</p> <p>A.31.0.E.1. El usuario no es el propietario ni administrador. Se muestra el mensaje de error “No es administrador ni propietario de la consulta”.</p> <p>A.31.0.E.2. La consulta ya está cerrada. Se muestra el mensaje de error “No se puede modificar una consulta cerrada”.</p> <p>A.31.0.E.3. La consulta está abierta, el usuario es administrador pero no propietario de la consulta y ha dejado el campo de respuesta vacío. Se muestra el mensaje de error “Es necesario introducir una respuesta”.</p> <p>A.31.1.E.1. La consulta está contestada o el usuario conectado no es el propietario de la consulta. Se muestra el mensaje de error “La consulta no se puede eliminar porque tiene una respuesta o no eres el propietario”</p>
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	32		
Nombre	Buscar masterdata		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	7/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de buscar masterdatas y realizar varias operaciones con ellos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Buscar → Masterdata
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.32.0. El sistema muestra los campos que puede llenar el actor:</p> <ul style="list-style-type: none"> a) Nombre b) Propietario c) Proyecto d) Fecha creación e) Fecha actualización f) Masterdata transferido g) Descripción <p>Una vez ha introducido los criterios de búsqueda debe pulsar el botón “Buscar”.</p> <p>P.32.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.32.1.1. Ningún criterio encontrado P.32.1.2. Sólo cumple criterio un masterdata P.32.1.3. Cumplen criterios varios masterdatas
Flujos alternativos	Ver casos de uso incluidos
Poscondiciones	<p>P.32.1.2. Se muestra una plantilla con todos los datos del masterdata, pudiendo el actor realizar varias operaciones.</p> <p>P.32.1.3. Se muestra una plantilla con los masterdatas que cumplen los requisitos y el actor puede elegir el que quiera ver, mostrándose entonces una plantilla del estilo “Sólo cumple criterio un masterdata”</p>
Excepciones	<p>P.32.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”</p> <p>P.32.1.1.E.1. El sistema no ha encontrado ningún masterdata con la información que ha dado el actor. Se muestra el mensaje de error “No existe ningún masterdata con los datos especificados”.</p>
Casos de uso incluidos	33. Transferir masterdata, 34. Ver ficheros, 35. Aceptar transfer, 36. Cambiar proyecto, 37. Borrar masterdata
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	33		
Nombre	Transferir masterdata		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cada uno de los masterdatas tiene un propietario, el cuál puede cambiar. Para ello el actual propietario debe hacer una transferencia al nuevo propietario.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Transferir masterdata” en la página que muestra los datos de dicho masterdata.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.33.0. El sistema muestra los campos de información del masterdata y el actor debe escoger en una lista desplegable el usuario al que quiere transferirlo. Una vez ha seleccionado el usuario debe pulsar el botón “Transfer”.
Flujos alternativos	
Poscondiciones	P.33.0. Se muestra la plantilla con todos los datos del masterdata.
Excepciones	P.33.0.E.1. El actor no es el propietario del masterdata. El sistema muestra el mensaje de error: “No es el propietario del masterdata” P.33.0.E.2. El masterdata no tiene ningún documento. El sistema muestra el mensaje de error: “El masterdata no tiene documentos”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	34		
Nombre	Ver ficheros		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cada uno de los masterdatas está formado por documentos. Éstos se pueden descargar.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Ver ficheros” en la página que muestra los datos de dicho masterdata.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.34.0. El sistema muestra los documentos que forman el masterdata. Los documentos pueden ser ficheros o asociaciones.

Flujos alternativos	
Poscondiciones	<p>P.34.0. Si los documentos son ficheros:</p> <p>P.34.0.1. Si hay un fichero se muestra una plantilla con todos sus datos y las operaciones relacionadas.</p> <p>P.34.0.2. Si hay más de un fichero se muestra una lista de los ficheros y se puede escoger el que se quiera ver.</p> <p>P.34.1. Si los documentos son asociaciones:</p> <p>P.34.1.1. Si hay una asociación se muestra una plantilla con todos sus datos y las operaciones relacionadas.</p> <p>P.34.1.2. Si hay más de una asociación se muestra una lista de las que contiene y se puede escoger la que se quiera ver.</p>
Excepciones	P.34.0.E.1. El masterdata no contiene ningún documento. El sistema muestra el mensaje de error: “El masterdata no tiene documentos”
Casos de uso incluidos	<p>Punto P.38.1.2. del caso de uso P.38 Buscar asociación</p> <p>Punto P.45.1.2 del caso de uso P.45 Buscar fichero</p>
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	35		
Nombre	Aceptar transfer		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cada uno de los masterdatas puede haber sido transferido por otro usuario. Este comando acepta dicha transferencia.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Aceptar transfer” en la página que muestra los datos del masterdata.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.35.0. Al pulsar el botón se acepta la transferencia.
Flujos alternativos	
Poscondiciones	P.35.0. Se muestra la plantilla con los datos del masterdata actualizados.
Excepciones	<p>P.35.0.E.1. La transferencia del masterdata está ya aceptada. Se muestra el mensaje de error “Transferencia ya aceptada o no es necesario aceptarla”</p> <p>P.35.0.E.2. El usuario que quiere aceptar la transferencia no es el propietario. Se muestra el mensaje de error “No es el propietario del masterdata”</p> <p>P.35.0.E.3. El masterdata no contiene ningún documento. Se muestra el mensaje de error “El masterdata no tiene documentos”</p>

Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	36		
Nombre	Cambiar proyecto		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cada uno de los masterdatas está asignado a un proyecto, pero existe la posibilidad de cambiarlo de proyecto.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Cambiar proyecto” en la página que muestra los datos de dicho masterdata.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.36.0. El sistema muestra los campos de información del masterdata y el actor debe escoger en una lista desplegable el proyecto al que quiere asignarlo. Una vez ha seleccionado el usuario debe pulsar el botón “Actualizar proyecto”.
Flujos alternativos	
Poscondiciones	P.36.0. Se muestra la plantilla con todos los datos del masterdata actualizados.
Excepciones	P.36.0.E.1. El actor no tiene permisos para cambiar el proyecto (es necesario que todas las versiones de fichero o asociación estén en personal o work). El sistema muestra el mensaje de error: “No tiene permisos para cambiar el proyecto”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Media
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	37		
Nombre	Borrar masterdata		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	21/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	Cada uno de los masterdatas puede ser eliminado de la base de datos.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Borrar” en la página que muestra los datos del masterdata.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.37.0. Al pulsar el botón se elimina el masterdata de la base de datos.
Flujos alternativos	
Poscondiciones	P.37.0. Se muestra la página general del gestor.
Excepciones	<p>P.37.0.E.1. El masterdata aún contiene ficheros. Se muestra el mensaje de error “El masterdata tiene ficheros”</p> <p>P.37.0.E.2. El masterdata está relacionado con alguna asociación. Se muestra el mensaje de error “El masterdata está en una asociación”</p> <p>P.37.0.E.3. El actor no tiene permisos para eliminar el masterdata. El sistema muestra el mensaje de error: “No tiene permisos para eliminar el masterdata”</p>
Casos de uso incluidos	
Prioridad	Alta
Frecuencia de uso	Baja
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	38		
Nombre	Buscar asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	23/10/2006	Fecha última actualización	8/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de buscar masterdatas y realizar varias operaciones con ellos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Buscar → Masterdata
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.38.0. El sistema muestra los campos que puede llenar el actor: a) Nombre b) Proyecto c) Propietario d) Estado

	<p>e) Descripción f) Fecha creación g) Fecha actualización h) Reservado</p> <p>Una vez ha introducido los criterios de búsqueda debe pulsar el botón “Buscar”.</p> <p>P.38.1. Mostrar resultados. P.38.1.1. Ningún criterio encontrado P.38.1.2. Sólo cumple criterio una asociación P.38.1.3. Cumplen criterios varias asociaciones</p>
Flujos alternativos	Ver casos de uso incluidos
Poscondiciones	<p>P.38.1.2. Se muestra una plantilla con todos los datos de la asociación, pudiendo el actor realizar varias operaciones. A los datos de P.38.0 hay que añadirle:</p> <p>a) Versión b) Masterdata</p> <p>P.38.1.3. Se muestra una plantilla con las asociaciones que cumplen los requisitos y el actor puede elegir la que quiera ver, mostrándose entonces una plantilla del estilo “Sólo cumple criterio una asociación”</p>
Excepciones	<p>P.38.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”</p> <p>P.38.1.1.E.1. El sistema no ha encontrado ninguna asociación con la información que ha dado el actor. Se muestra el mensaje de error “No existe ninguna asociación con los datos especificados”.</p>
Casos de uso incluidos	<p>39. Ver estructura 40. Guardar versión asociación, 41. Cambiar versión asociación, 42. Cambiar estado asociación, 43. Descargar asociación, 44. Borrar asociación</p>
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	39		
Nombre	Ver estructura		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	23/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de ver la estructura multinivel de cualquier asociación.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Ver estructura” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.39.0. Al pulsar el botón se ejecuta la acción para ver la estructura.
Flujos alternativos	Ver casos de uso incluidos
Poscondiciones	P.39.0. Se muestra la estructura de la asociación indentada en niveles.
Excepciones	P.39.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	40		
Nombre	Guardar versión asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	23/10/2006	Fecha última actualización	8/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de guardar una nueva versión de la asociación.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Guardar versión” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.40.0. El sistema muestra información de la asociación. El usuario debe escoger el tipo de versión que quiere y, además, permite realizar varias operaciones.
Flujos alternativos	A.40.0. Actualizar la asociación A.40.1. Añadir información a la estructura. A.40.2. Eliminar información de la estructura
Poscondiciones	A.40.0. Se muestra la página general del gestor A.40.1. Busca masterdatas y los añade a la asociación. Ver punto P.30.1 y los puntos A.30.X del caso de uso 30. Crear asociación A.40.2. Elimina masterdatas de la asociación

Excepciones	P.40.0.E.1. El usuario no tiene permisos para crear la versión. El sistema muestra el mensaje de error: “No tiene permisos suficientes” P.40.0.E.2. El usuario no ha escogido el tipo de versión que quiere guardar. El sistema muestra el mensaje de error: “Es necesario seleccionar el tipo de versión”
Casos de uso incluidos	Ver punto P.30.1 y A.30.X del caso de uso 30. Crear asociación
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	41		
Nombre	Cambiar versión asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	23/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de cambiar la versión de una asociación existente.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Cambiar versión” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña. Si el documento está reservado aparecerá la opción sobreescibir. En caso contrario no aparecerá.
Flujo principal	P.41.0. El sistema muestra la información de la asociación añadiendo una lista desplegable en la que el usuario debe seleccionar el nuevo tipo de versión que quiere. Despues debe pulsar el botón “Actualizar versión”
Flujos alternativos	
Poscondiciones	P.41.0. Se muestra la plantilla con los datos de la asociación actualizados.
Excepciones	P.41.0.E.1. El usuario no ha escogido el tipo de versión que quiere guardar. El sistema muestra el mensaje de error: “Es necesario seleccionar el tipo de versión” P.41.0.E.2. La nueva versión calculada ya existe. El sistema muestra el mensaje de error: “La nueva versión de la asociación ya existe” P.41.0.E.3. El usuario no tiene permisos para crear la versión. El sistema muestra el mensaje de error: “No tiene permisos para cambiar la versión”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	42		
Nombre	Cambiar estado asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	23/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de cambiar el estado de una asociación existente.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Cambiar estado” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.42.0. El sistema muestra la información de la asociación añadiendo una lista desplegable en la que el usuario debe seleccionar el nuevo estado. Despues debe pulsar el botón “Actualizar estado”
Flujos alternativos	
Poscondiciones	P.42.0. Se muestra la plantilla con los datos de la asociación actualizados.
Excepciones	P.42.0.E.1. El usuario no tiene permisos para cambiar el estado. El sistema muestra el mensaje de error: “No tiene permisos para cambiar el estado”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	43		
Nombre	Descargar asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	24/10/2006	Fecha última actualización	9/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de descargar los ficheros que forman la asociación en un archivo comprimido. Hay dos modos de descargarlos, como mayor versión y como mayor estado.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Descargar” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.43.0. El sistema muestra una nueva ventana con dos opciones: a. Versión más alta b. Estado más alto Despues debe pulsar el botón “Descargar”
Flujos alternativos	A.43.0. Descargar versión más alta A.43.1. Descargar estado más alto
Poscondiciones	P.43.0. Se muestra la plantilla con los datos de la asociación que se ha

	descargado
Excepciones	P.43.0.E.1. El usuario no tiene permisos para descargar la asociación. El sistema muestra el mensaje de error: “No tiene permisos para descargar algún fichero de la asociación” P.43.0.E.2. El fichero no existe. Se muestra el mensaje de error “El fichero comprimido no existe”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	44		
Nombre	Borrar asociación		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	24/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de eliminar las asociaciones.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Borrar” en la plantilla de información de una asociación.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.44.0. Una vez el usuario pulsa el botón “Borrar” la asociación se elimina de la base de datos.
Flujos alternativos	
Poscondiciones	P.44.0. Se muestra la pantalla de inicio del gestor.
Excepciones	P.44.0.E.1. El usuario no tiene permisos para borrar la asociación. El sistema muestra el mensaje de error: “No tiene permisos para borrar la asociación” P.44.0.E.2. La versión de la asociación o sus relaciones no se han podido borrar. Se muestra el mensaje de error “No se ha podido borrar la versión” P.44.0.E.3. El masterdata de la asociación no se ha podido borrar. Se muestra el mensaje de error “No se ha podido borrar el masterdata” P.44.0.E.4. La versión de la asociación es única pero pertenece a la estructura de otra asociación. Se muestra el mensaje de error “La asociación es única y el masterdata tiene una asociación”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Media
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	45		
Nombre	Buscar fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	26/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece la posibilidad a todos los usuarios de buscar ficheros y realizar varias operaciones con ellos.
Evento inicial	El caso de uso empieza cuando el actor selecciona la opción Buscar → Fichero
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	<p>P.45.0. El sistema muestra los campos que puede llenar el actor:</p> <ul style="list-style-type: none"> a) Nombre b) Proyecto c) Propietario d) Tipo e) Estado f) Tamaño g) Descripción h) Fecha creación i) Fecha actualización j) Reservado <p>Una vez ha introducido los criterios de búsqueda debe pulsar el botón “Buscar”.</p> <p>P.45.1. Mostrar resultados.</p> <ul style="list-style-type: none"> P.45.1.1. Ningún criterio encontrado P.45.1.2. Sólo cumple criterio un fichero P.45.1.3. Cumplen criterios varios ficheros
Flujos alternativos	Ver casos de uso incluidos
Poscondiciones	<p>P.45.1.2. Se muestra una plantilla con todos los datos del fichero, pudiendo el actor realizar varias operaciones.</p> <p>P.45.1.3. Se muestra una plantilla con los ficheros que cumplen los requisitos y el actor puede elegir el que quiera ver, mostrándose entonces una plantilla del estilo “Sólo cumple criterio un fichero”</p>
Excepciones	<p>P.45.0.E.1. El usuario no tiene acceso a ningún proyecto. El sistema muestra el mensaje de error: “El usuario no tiene acceso a ningún proyecto”</p> <p>P.45.1.1.E.1. El sistema no ha encontrado ningún fichero con la información que ha dado el actor. Se muestra el mensaje de error “No existe ningún fichero con los datos especificados”.</p>
Casos de uso incluidos	46. Guardar versión fichero 47. Cambiar versión fichero 48. Cambiar estado fichero 49. Descargar fichero 50. Borrar fichero
Prioridad	Muy alta

Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	46		
Nombre	Guardar versión fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	26/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de guardar una nueva versión de un fichero.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Guardar versión” en la plantilla de información de un fichero.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.46.0. El sistema muestra información de la asociación. El usuario debe escoger el tipo de versión que quiere y el fichero físico que quiere guardar.
Flujos alternativos	
Poscondiciones	
Excepciones	<p>P.46.0.E.1. El usuario no tiene permisos para crear la versión. El sistema muestra el mensaje de error: “No tiene permisos para crear la versión”</p> <p>P.46.0.E.2. El usuario no ha escogido el tipo de versión que quiere guardar. El sistema muestra el mensaje de error: “Es necesario seleccionar el tipo de versión”</p> <p>P.46.0.E.3. El usuario no ha escogido ningún fichero físico. El sistema muestra el mensaje de error: “Es necesario escoger un fichero”</p> <p>P.46.0.E.4. El fichero físico escogido no tiene el mismo nombre que el del resto de versiones. El sistema muestra el mensaje de error: “El nombre de la versión debe ser el mismo que en la versión anterior”</p> <p>P.46.0.E.5. La copia del fichero físico al servidor ha fallado. El sistema muestra el mensaje de error: “El fichero físico no se ha podido guardar”</p>
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	47		
Nombre	Cambiar versión fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	26/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de cambiar la versión de un fichero existente.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Cambiar versión” en la plantilla de información de un fichero.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña. Si el documento está reservado aparecerá la opción sobreescibir. En caso contrario no aparecerá.
Flujo principal	P.47.0. El sistema muestra la información del fichero añadiendo una lista desplegable en la que el usuario debe seleccionar el nuevo tipo de versión que quiere. Después debe pulsar el botón “Actualizar versión”
Flujos alternativos	
Poscondiciones	P.47.0. Se muestra la plantilla con los datos del fichero actualizados.
Excepciones	<p>P.47.0.E.1. El usuario no ha escogido el tipo de versión que quiere guardar. El sistema muestra el mensaje de error: “Es necesario seleccionar el tipo de versión”</p> <p>P.41.0.E.2. La nueva versión calculada ya existe. El sistema muestra el mensaje de error: “La nueva versión del fichero ya existe”</p> <p>P.41.0.E.3. El usuario no tiene permisos para crear la versión. El sistema muestra el mensaje de error: “No tiene permisos para cambiar la versión”</p> <p>P.42.0.E.3. Ha habido un error al renombrar el fichero creado. Se muestra el mensaje de error “Error al renombrar el fichero físico”</p>
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Media
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	48		
Nombre	Cambiar estado fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	26/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de cambiar el estado de un fichero existente.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Cambiar estado” en la plantilla de información de un fichero.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.48.0. El sistema muestra la información del fichero añadiendo una lista desplegable en la que el usuario debe seleccionar el nuevo estado. Después debe pulsar el botón “Actualizar estado”
Flujos alternativos	
Poscondiciones	P.48.0. Se muestra la plantilla con los datos del fichero actualizados.
Excepciones	P.48.0.E.1. El usuario no tiene permisos para cambiar el estado. El sistema muestra el mensaje de error: “No tiene permisos para cambiar el estado”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	49		
Nombre	Descargar fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	27/10/2006	Fecha última actualización	10/12/2006

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de descargar cualquier fichero del gestor.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Descargar” en la plantilla de información de un fichero.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.48.0. Al pulsar el botón se abre la ventana del explorador preguntando si se quiere abrir el fichero, guardarlo o cancelar la operación.
Flujos alternativos	
Poscondiciones	P.48.0. Se muestra la plantilla con los datos del fichero que se ha descargado
Excepciones	P.48.0.E.1. El usuario no tiene permisos para descargar el fichero. El sistema muestra el mensaje de error: “No tiene permisos para descargar el fichero” P.48.0.E.2. El fichero no existe. Se muestra el mensaje de error “No se ha

	encontrado el fichero físico”
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Muy alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

ID	50		
Nombre	Borrar fichero		
Creado por	Francisco Expósito	Actualizado por	Francisco Expósito
Fecha creación	27/10/2006	Fecha última actualización	

Actores	Usuarios, Administrador
Descripción	El sistema ofrece a los actores la posibilidad de eliminar los ficheros.
Evento inicial	El caso de uso empieza cuando el actor pulsa el botón “Borrar” en la plantilla de información de un fichero.
Precondiciones	El actor debe de haber introducido su cuenta de usuario y contraseña.
Flujo principal	P.44.0. Una vez el usuario pulsa el botón “Borrar”, el fichero se elimina de la base de datos.
Flujos alternativos	
Poscondiciones	P.44.0. Se muestra la pantalla de inicio del gestor.
Excepciones	<p>P.44.0.E.1. El usuario no tiene permisos para borrar el fichero. El sistema muestra el mensaje de error: “No tiene permisos para borrar el fichero”</p> <p>P.44.0.E.2. La información del fichero o el fichero físico no se han podido borrar. Se muestra el mensaje de error “El fichero no se ha podido borrar”</p> <p>P.44.0.E.3. El masterdata del fichero no se ha podido borrar. Se muestra el mensaje de error “No se ha podido borrar el masterdata”</p> <p>P.44.0.E.4. La versión del fichero es única pero pertenece a la estructura de una asociación. Se muestra el mensaje de error “El fichero es único y el masterdata está en una asociación”</p>
Casos de uso incluidos	
Prioridad	Muy alta
Frecuencia de uso	Alta
Normas de empresa	
Requisitos no funcionales	
Comentarios	

Apéndice B

Implementación base de datos

Tablas

En la descripción de las tablas aparece el nombre de la tabla y, entre paréntesis, el nombre del tablespace al cual está asignada dicha tabla. Los datos que se muestran son:

- Restricciones de los campos, ya sean claves primarias, foráneas o campos con valores únicos.
- Nombre del campo
- Tipo de datos
- Tamaño del tipo de datos
- Si el campo puede estar vacío.

1. Tipo_Acl

TIPO_ACL (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdTipoAcl Tipo	Number Varchar2	4 20	Sí Sí

2. Estado_fichero

ESTADO_FICHERO (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdEstado Nombre	Number Varchar2	4 16	Sí Sí

3. Estado_consulta

ESTADO_CONSULTA (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdEstado Nombre	Number Varchar2	4 16	Sí Sí

4. Acl

ACL (tab_acl)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK		IdAcl	Number	16	Sí
FK	U	rTipoAcl	Number	4	Sí
	U	Nombre	Varchar2	40	Sí
FK	U	rEstado	Number	4	Sí
	U	Propietario	Char	1	Sí
		CrearProyecto	Char	1	Sí
		ModifAtribProyecto	Char	1	Sí
		BorrarProyecto	Char	1	Sí
		LeerMasterdata	Char	1	Sí
		ModifAtribMasterdata	Char	1	Sí
		BorrarMasterdata	Char	1	Sí
		CambiarProyecto	Char	1	Sí
		LeerFichero	Char	1	Sí
		EscribirFichero	Char	1	Sí
		ModifAtribFichero	Char	1	Sí
		BorrarFichero	Char	1	Sí
		CambiarEstadoA	Char	1	Sí
		CambiarEstadoDesde	Char	1	Sí
		Reservar	Char	1	Sí

5. Grupos

GRUPOS (tab_grupos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK		IdGrupo	Number	16	Sí
	U	Nombre	Varchar2	20	Sí

6. Rol

ROL (tab_roles)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	IdRol	Number	16	Sí	
	U	Nombre	Varchar2	20	
		Descripción	Varchar2	100	Sí

7. Empresa

EMPRESA (tab_usuario_empresa)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	NifEmpresa U	Char	16	Sí	
		Nombre	Varchar2	30	Sí
		Dirección	Varchar2	50	Sí
		Ciudad	Varchar2	20	Sí
		CodPostal	Char	5	Sí
		TlfFijo	Char	9	
		TlfMovil	Char	9	
		Horario	Char	25	
		Hotline	Char	1	Sí
		ContratoVigor	Char	1	Sí
		Notas	Varchar2	100	

8. Usuario

USUARIO (tab_usuario_empresa)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	Dni U	Char	9	Sí	
		Passwd	Char	8	Sí
		Nombre	Varchar2	20	Sí
		Apellido1	Varchar2	20	Sí
		Apellido2	Varchar2	20	Sí
		TlfFijo	Char	9	Sí
		TlfMovil	Char	9	
		Mail	Varchar2	40	
FK	rEmpresa	Char	16		

9. Proyecto

PROYECTO (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdProyecto	Number	8	Sí
		Nombre	Varchar2	20	Sí
		FechaCreacion	Date		Sí
		Descripcion	Varchar2	150	

10. Tipo_fichero

TIPO_FICHERO (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdTipo	Number	16	Sí
		Extension	Varchar2	4	Sí
		Programa	Varchar2	20	

11. Grupo_usuario

GRUPO_USUARIO (tab_grupos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK,FK		IdGrupo	Number	16	Sí
PK,FK		IdUsuario	Char	9	Sí

12. Grupos_por_proyecto

GRUPOS.POR.PROYECTO (tab_grupos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK,FK		IdProyecto	Number	8	Sí
PK,FK		IdGrupo	Number	16	Sí
PK,FK		rAcl	Number	16	

13. Masterdata

MASTERDATA (tab_masterdata)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdMasterdata	Number	16	Sí
		Nombre	Varchar2	40	Sí
		FechaCreacion	Date		Sí
		FechaActualizacion	Date		Sí
FK		rPropietario	Char	9	Sí
FK		rProyecto	Number	8	Sí
		TransferAceptada	Char	1	
		Descripcion	Varchar2	100	

14. Fichero

FICHERO (tab_fichero)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK	U	IdFichero	Number	16	Sí
	U	Version	Varchar2	5	Sí
		Nombre	Varchar2	40	Sí
FK		rMasterdata	Number	16	Sí
FK		rTipo	Number	16	Sí
FK		rEstadoFichero	Number	4	Sí
		Volumen	Number	16	
		Descripcion	Varchar2	150	
		FechaCreacion	Date		
		FechaActualizacion	Date		Sí
FK		rProyecto	Number	8	Sí
FK		rPropietario	Char	9	Sí
		Reservado	Char	1	Sí

15. Consulta

CONSULTA (tab_consultas)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK		IdConsulta	Number	16	Sí
		Titulo	VARCHAR2	30	Sí
		Pregunta	VARCHAR2	400	Sí
		Respuesta	VARCHAR2	400	
		FechaPregunta	Date		Sí
		FechaRespuesta	Date		
		rProyecto	Number	8	Sí
FK		rPropietario	Char	9	Sí
FK		rEstadoConsulta	Number	4	Sí

16. Cambio_estado_consulta

CAMBIO_ESTADO_CONSULTA (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK		IdCambioEstadoConsulta	Number	4	
FK	U	rEstadoOrigen	Number	4	
FK	U	rEstadoDestino	Number	4	

17. Cambio_estado_fichero

CAMBIO_ESTADO_FICHERO (tab_otros_datos)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK		IdCambioEstadoFichero	Number	4	Sí
FK	U	rEstadoOrigen	Number	4	Sí
FK	U	rEstadoDestino	Number	4	Sí

18. Usuario_rol_proyecto

USUARIO_ROL_PROYECTO (tab_roles)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK,FK		IdProyecto	Number	8	Sí
PK,FK		IdRol	Number	16	Sí
PK,FK		IdUsuario	Char	9	Sí
PK,FK		rAcl	Number	16	Sí

19. Asociación

ASOCIACION (tab_asociaciones)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK FK FK FK FK	U	IdAsociacion	Number	16	Sí
		Nombre	VARCHAR2	40	Sí
	U	Version	VARCHAR2	5	Sí
		rPropietario	Char	9	Sí
	FK	rProyecto	Number	8	Sí
		FechaCreacion	Date		Sí
	FK	FechaActualizacion	Date		Sí
		rMasterdata	Number	16	Sí
	FK	rEstado	Number	4	Sí
		Estructura	VARCHAR2	200	
	FK	Reservado	Char	1	
		Descripcion	VARCHAR2	150	

20. Masterdata_en_asociación

MASTERDATA_EN_ASOCIACION (tab_asociaciones)					
Clave	Unicidad	Nombre	Tipo	Tamaño	No NULL
PK,FK		IdAsociacion	Number	16	Sí
PK,FK		IdMasterdata	Number	16	Sí

Secuencias y disparadores

a) IdProyecto en la tabla Proyectos

```
create sequence EXPOSITO.proyecto_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idproyecto_trigger
before insert on EXPOSITO.PROYECTO
for each row
begin
select proyecto_seq.nextval into :new.IDPROYECTO from dual;
end;
/
```

b) IdEstado en la tabla Estado_consulta

```
create sequence EXPOSITO.estadoConsulta_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idEstadoConsulta_trigger
before insert on EXPOSITO.ESTADO_CONSULTA
foreach row
begin
select estadoConsulta_seq.nextval into :new.IDESTADO from dual;
end;
/
```

c) IdEstado en la tabla Estado_fichero

```
create sequence EXPOSITO.estadoFichero_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idEstadoFichero_trigger
before insert on EXPOSITO.ESTADO_FICHERO
for each row
begin
select estadoFichero_seq.nextval into :new.IDESTADO from dual;
end;
/
```

d) IdTipo en la tabla Tipo_fichero

```
create sequence EXPOSITO.tipoFichero_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idTipoFichero_trigger
before insert on EXPOSITO.TIPO_FICHERO
foreachrow
begin
select tipoFichero_seq.nextval into :new.IDTIPO from dual;
end;
/
```

e) IdConsulta en la tabla Consulta

```
create sequence EXPOSITO.idConsulta_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idConsulta_trigger
before insert on EXPOSITO.CONSULTA
for each row
begin
select idConsulta_seq.nextval into :new.IDCONSULTA from dual;
end;
/
```

f) IdGrupo en en la tabla Grupos

```
create sequence EXPOSITO.grupos_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idgrupo_trigger
before insert on EXPOSITO.GRUPOS
for each row
begin
select grupos_seq.nextval into :new.IDGRUPO from dual;
end;
/
```

g) IdAcl en la tabla ACL

```
create sequence EXPOSITO.acls_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idacl_trigger
before insert on EXPOSITO.ACL
for each row
begin
select acls_seq.nextval into :new.IDACL from dual;
end;
/
```

h) IdRol en la tabla Rol

```
create sequence EXPOSITO.rol_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idrol_trigger
before insert on EXPOSITO.ROL
for each row
begin
select rol_seq.nextval into :new.IDROL from dual;
end;
/
```

i) IdFichero en la tabla Fichero

```
create sequence EXPOSITO.fichero_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idfichero_trigger
before insert on EXPOSITO.FICHERO
for each row
begin
select fichero_seq.nextval into :new.IDFICHERO from dual;
end;
/
```

j) IdCambioEstadoConsulta en la tabla Cambio_estado_consulta

```
create sequence EXPOSITO.cambioestadococonsulta_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idcambioestadococonsulta_trigger
before insert on EXPOSITO.CAMBIO_ESTADO_CONSULTA
for each row
begin
select cambioestadococonsulta_seq.nextval into :new.IDCAMBIOESTADOCONSULTA from dual;
end;
/
```

k) IdCambioEstadoFichero en la tabla Cambio_estado_fichero

```
create sequence EXPOSITO.cambioestadofichero_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idcambioestadofichero_trigger
before insert on EXPOSITO.CAMBIO_ESTADO_FICHERO
for each row
begin
select cambioestadofichero_seq.nextval into :new.IDCAMBIOESTADOFICHERO from dual;
end;
/
```

l) IdMasterdata en la tabla Masterdata

```
create sequence EXPOSITO.masterdata_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idmasterdata_trigger
before insert on EXPOSITO.MASTERDATA
for each row
begin
select masterdata_seq.nextval into :new.IDMASTERDATA from dual;
end;
/
```

m) IdTipoAcl en la tabla Tipo_acl

```
create sequence EXPOSITO.idtipoacl_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idtipoacl_trigger
before insert on EXPOSITO.TIPO_ACL
for each row
begin
select idtipoacl_seq.nextval into :new.IDTIPOACL from dual;
end;
/
```

n) IdAsociacion en la tabla Asociación

```
create sequence EXPOSITO.asociacion_seq
start with 1
increment by 1
nomaxvalue;
create trigger EXPOSITO.idasociacion_trigger
before insert on EXPOSITO.ASOCIACION
for each row
begin
select asociacion_seq.nextval into :new.IDASOCIACION from dual;
end;
/
```

Índices

- CREATE INDEX EXPOSITO.I_APELLIDO1 ON EXPOSITO.USUARIO(APELLIDO1) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_REMPRESA ON EXPOSITO.USUARIO(REMPRESA) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RMASTERDATA ON EXPOSITO.FICHERO(RMASTERDATA) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RTIPO EXPOSITO.FICHERO(RTIPO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RESTADOFICHERO EXPOSITO.FICHERO(RESTADOFICHERO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROYECTO_F EXPOSITO.FICHERO(RPROYECTO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROPIETARIO_F EXPOSITO.FICHERO(RPROPIETARIO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROYECTO_C EXPOSITO.CONSULTA(RPROYECTO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROPIETARIO_C EXPOSITO.CONSULTA(RPROPIETARIO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RESTADOCONSULTA_C EXPOSITO.CONSULTA(RESTADOCONSULTA) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROYECTO_M EXPOSITO.MASTERDATA(RPROYECTO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROPIETARIO_M EXPOSITO.MASTERDATA(RPROPIETARIO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROYECTO_A EXPOSITO.ASOCIACION(RPROYECTO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RPROPIETARIO_A EXPOSITO.ASOCIACION(RPROPIETARIO) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RMASTERDATA_A EXPOSITO.ASOCIACION(RMASTERDATA) TABLESPACE TAB_INDICES;
- CREATE INDEX EXPOSITO.I_RESTADO_A EXPOSITO.ASOCIACION(RESTADO) TABLESPACE TAB_INDICES;

Bibliografía

- [Hus02] Husted, Ted. *Struts in Action: A practical guide to the leading java web framework*. Manning Publications, 2003.
- [Sig02] Siggelkow, Bill. *Jakarta Struts Cookbook*. O'Reilly, 2005
- [Lon01] Loney, Kevin. *Oracle Database 10G: the complete reference*. McGraw-Hill, 2004.
- [Gar02] Garmany, John and Burleson, Donald K. *Oracle application server 10G administration handbook*. McGraw-Hill, 2004
- [Goo02] Goodwill, James. *Apache Jakarta-Tomcat*. Apress, 2002
- [Eck02] Eckel, Bruce. *Thinking in Java*. Prentice-Hall, 2005.
- [Ste02] Stevens, Perdita. *Using UML: software engineering with objects and components*. Addison-Wesley, 2006.
- [WPD] Wikipedia, The Free Encyclopedia (<http://www.wikipedia.org>)
- [ASF] Apache Software Foundation (<http://struts.apache.org> y
<http://tomcat.apache.org>)
- [SUN] Sun Microsystems (<http://www.sun.com>)
- [ORA] Oracle (<http://www.oracle.com>)
- [TEF] The Eclipse Foundation (<http://www.eclipse.org>)
- [UIT] Utags Sourceforge (<http://uitags.sourceforge.net>)
- [IRS] IBM Rational Software (<http://www.rational.com>)

Firmado: Francisco Expósito Aguilera

Bellaterra, junio 2007

Resumen

En este proyecto se ha realizado el diseño y la implementación de un sistema PDM genérico mediante el cual sea posible crear y gestionar la información relativa a un producto. A través de esta herramienta web, realizada utilizando el patrón de arquitectura MVC, que es una de las más importantes en la gestión del ciclo de vida de un producto, se pretende disminuir la pérdida de tiempo y datos en la organización de los datos de diseño, mejorar la productividad y aumentar la colaboración.

Resum

En aquest projecte s'ha realitzat el disseny i la implementació d'un sistema PDM genèric mitjançant el qual sigui possible crear i gestionar la informació relacionada amb un producte. Amb aquesta eina web realitzada fent ús del patró d'arquitectura MVC, es pretén disminuir la pèrdua de temps i de dades a la organització de dades de disseny, millorar la productivitat i augmentar la col·laboració.

Abstract

In this Project has been carried out the design and implementation of a generic PDM system. Through this application it is possible to create and manage information related to a product. This tool, which has been put into effect using the MVC web design pattern, purports to minimize the waste of time and data in the data design organization, improve the productivity and increase the collaboration.

