

# **TFC-J2EE Tienda Virtual.**

**Gustavo Adolfo Visier Cabezudo.**

**ETIG / ETIS**

**14 de Enero de 2.008.**

# INDICE

<b>1</b>	<b>Introducción</b>	<b>4</b>
1.1	Justificación del TFC	4
1.2	Objetivos del TFC	4
1.2.1	Generales	4
1.2.2	Específicos	4
1.3	Enfoque y método de trabajo	4
1.4	Planificación del proyecto	6
1.5	Productos obtenidos	9
1.6	Descripción de otros capítulos de la memoria.	9
<b>2</b>	<b>Descripción del sistema y Revisión de los casos de uso.</b>	<b>9</b>
2.1	Descripción del sistema.	9
2.2	Proceso de la conexión.	9
2.3	Mantenimiento de los datos.	10
2.3.1.	Gestión de Usuarios	10
2.3.2	Gestión de libros.	10
2.3.3	Gestión de pedidos.	11
2.4	Búsquedas de datos.	12
2.5	Carro de la compra.	12
<b>3</b>	<b>Actores.</b>	<b>13</b>
<b>4</b>	<b>Esquema de la aplicación.</b>	<b>14</b>
4.1	Detalle de funcionalidades.	15
4.2	Casos de uso según el actor.	19
<b>5</b>	<b>Especificación formal de los casos de uso.</b>	<b>20</b>
5.1	Caso de uso Mantenimiento.	20
5.1.1	Diagramas de colaboración.	21
5.2	Caso de uso Búsquedas	22
5.3	Caso de uso compra.	24
<b>6</b>	<b>Datos a tratar en la aplicación.</b>	<b>25</b>
6.1	Usuarios.	25
6.2	Libros.	25
6.3	Pedidos.	26
6.4	Diagrama de Entidades.	27
<b>7</b>	<b>Diseño.</b>	<b>28</b>
7.1	J2EE standard de desarrollo en aplicaciones Web.	28
7.2	Estructura J2EE.	29
7.3.	Diseño arquitectónico del sistema.	30
7.3.1	Nivel de presentación.	31
7.3.2	Nivel de negocio.	32
7.3.3	Nivel de datos.	36
7.4	Conclusiones del diseño.	37
<b>8</b>	<b>Tecnología a utilizar.</b>	<b>38</b>
8.1	Recursos necesarios.	38
8.2	Tecnología de desarrollo utilizada.	38
8.3	Diseño de la interfaz de Usuario.	38
8.3.1	Normas básicas.	38
8.3.2	Descripción de la interfaz de usuario.	39

<b>8.4 Presentación de páginas de ejemplo.</b>	<b>40</b>
8.4.1 Página de Inicio.	40
8.4.2 Página de búsqueda de libros.	41
8.4.3 Pantalla de edición de los datos de un libro.	42
8.4.4 El carro de la compra.	43
8.4.5 Pantalla de edición de datos del usuario.	44
8.4.6 Mensajes de confirmación.	45
8.4.7 Control de errores	46
<b>9 Estructura del proyecto.</b>	<b>47</b>
9.1 Estructura de directorios.	47
9.2 Detalles de implementación.	48
9.2.1 Detalles generales	48
9.2.2 Empaquetado y Despliegue de la aplicación	49
9.3 Valoración económica.	49
<b>10 Conclusiones.</b>	<b>50</b>
10. Valoración del diseño.	50
<b>11 Bibliografía.</b>	<b>51</b>
<b>Anexo A. Guía de Instalación.</b>	<b>52</b>

## **1 Introducción:**

### **1.1 Justificación del TFC.**

He elegido la opción propuesta por el profesor ya que no tengo ninguna preferencia sobre el programa a realizar. Así cubriré todos los objetivos propuestos. Además creo que se trata más de utilizar los conceptos de Ingeniería del Software, aplicar la plataforma J2EE y utilizar bases de datos, independientemente de los datos que se graben en la base de datos.

Hace mucho tiempo que las tiendas de Internet proliferan y van comiendo terreno a las tiendas clásicas y por lo tanto existen múltiples desarrollos sobre este tema.

### **1.2 Objetivos del TFC**

#### **1.2.1 Generales:**

Crear una tienda virtual de libros en Internet, en la que los internautas podrán realizar sus pedidos mediante el sistema de carro de la compra.

#### **1.2.2 Específicos:**

La tienda estará dividida en 2 áreas la pública y la privada:

- En la pública los clientes podrán consultar las obras disponibles por categorías y hacer sus pedidos mediante el sistema de carro de la compra.
- El área privada será para los administradores. Podrán consultar los usuarios que se han dado de alta, los stocks de libros y los pedidos realizados además de su estado.

### **1.3 Enfoque y Método seguido en el proyecto.**

Se realizará una planificación inicial para comenzar el proyecto, pero ya sabemos que a lo largo de su desarrollo habrá modificaciones sobre la marcha según vaya apareciendo nueva información. Por lo tanto la planificación se irá adaptando a los nuevos datos y escenarios que nos encontremos.

Para el desarrollo del software seguiremos un ciclo de vida como el Racional Unified Process, iterativo e incremental. Se distinguen estas cuatro fases:

- 1) Inicio, en la que se establece la justificación económica del software y se delimita el alcance del proyecto.

2) Elaboración, en la cual se estudia el dominio del problema (parte de la actividad de la empresa dentro de la cual se utilizará el software) y se tienen en cuenta muchas de las necesidades de información, eventuales requisitos, restricciones, se establece la arquitectura general del software y se realiza una planificación del proyecto.

3) Construcción, en la que se desarrolla todo el producto de forma iterativa e incremental, tiene en cuenta todas las necesidades de información que debe satisfacer y desarrolla la arquitectura obtenida en la fase anterior.

4) Transición, que comprende la entrega del producto al cliente y el comienzo de su utilización; aunque es posible que sea necesario hacer retoques. La programación exploratoria consiste en elaborar una primera versión del software, o de una parte de éste, enseñarla a los usuarios para que la critiquen y a continuación, hacerle los cambios que éstos sugieran, proceso que se repetirá tantas veces como sea necesario. En cada una de estas fases se llevan a término (en diferentes proporciones) las siguientes tareas:

- recogida de requisitos (requirement capture)
- análisis y diseño
- realización (implementation)
- prueba (test)

Para el desarrollo del software utilizaremos el *Unified Modeling Language* (UML) que es un modelo para la construcción de *software* orientado a objetos que ha sido propuesto como estándar de ISO por el OMG. Consta de un conjunto de tipos de diagramas interrelacionados, dentro de los cuales se utilizan elementos del modelo, que sirven para describir distintos aspectos de la estructura y la dinámica del *software*.

## **1.4 Planificación del proyecto.**

Quisiera hacer notar que de forma paralela a la realización del proyecto debo aprender toda la arquitectura J2EE y todos los aspectos técnicos relacionados con el desarrollo en esta plataforma. La planificación siguiente está adaptada a las fechas de entrega de las pac. En principio estoy utilizando un módulo de J2EE de la UOC como introducción y a partir de ahí utilizaré los libros J2EE 1.4 Bible y Mastering Enterprise Java Beans los dos de Wiley.

### **Plan de trabajo.**

#### **(PAC1 Entrega 28 de Septiembre).**

- Análisis general del proyecto.
- Determinación del tiempo necesario.
- Definición del plan del proyecto.
- Plan de trabajo. PEC1.

### **Análisis/Requisitos y Diseño del software.**

#### **(PAC2 Entrega 29 de Octubre).**

#### **Análisis.**

- Revisión de los casos de uso.
- Paquetes de análisis y de servicios.
- Identificación de las clases de entidades.
- Especificación de los atributos de las clases de entidades.
- Especificación de las relaciones entre las clases.
- Identificación de las clases de frontera, control y operaciones.
- Especificación formal de los casos de uso.
- Glosario.
- Funcionalidades versiones siguientes.

- Análisis completado.

### **Diseño.**

- Revisar las especificaciones preliminares.
- Diseño arquitectónico del sistema.
- Diseño de los casos de uso.
- Diagrama estático de diseño.
- Diseño de la persistencia.
- Diseño de la interfaz de usuario.
- Diseño de los subsistemas.
- Integración de documentos.
- Diseño completo.

### **Implementación y Entrega parcial. (PAC 3 Entrega 17 de Diciembre).**

- Revisar los casos de uso.
- Implementación del diseño de la persistencia.
- Desarrollo código de las distintas clases.
- Revisión.
- Integración de documentos.

### **Implementación y Entrega. (Final Entrega 14 de Enero).**

### **Testing y calidad.**

- Desarrollar planes de unidades de test.

- Desarrollar planes de integración de los test.

#### **Unidad de test.**

- Revisar código modular.
- Revisar los módulos de los componentes y el ajuste a las especificaciones.
- Identificar las discrepancias con las especificaciones.
- Modificar el código.
- Volver a comprobar el código modificado.

#### **Comprobar la integración.**

- Comprobar la integración de los módulos.
- Identificar las discrepancias con las especificaciones.
- Modificar el código.
- Volver a comprobar el código modificado.

#### **Documentación.**

- Desarrollar los manuales de usuarios.
- Revisar toda la documentación del usuario.

#### **Entrega.**

- Comprobar toda la documentación.
- Instalar/distribuir el software.
- Evaluar toda la información.
- Enviar entrega final.

## **1.5 Productos obtenidos.**

Desarrollo de una tienda virtual utilizando el standard J2EE.  
Memoria del proyecto.  
Presentación de diapositivas.

## **1.6 Descripción de otros capítulos de la memoria.**

La descripción general se realiza en el capítulo 2 y de esta emanan los casos de uso que se utilizan en el desarrollo de la aplicación. Luego se tratan los actores en el capítulo 3, se desglosa en detalle las funcionalidades en el 4 y se ven en detalle los casos de uso en el 5. En el 6 se ven los datos que se utilizan y su persistencia y luego se pasa al Diseño, la tecnología a utilizar y la interface de usuario.

## **2 Descripción del sistema y Revisión de los casos de uso:**

### **2.1 Descripción del sistema:**

La tienda virtual de libros se ejecutará en uno o varios servidores dependiendo de la afluencia de clientes. Estos serán accedidos localmente (intranet) o remotamente (Internet) a través de un navegador Web. La información se almacenará en un servidor de base de datos.

Se pretende la gestión de de la compra-venta de libros a través de Internet posibilitando las consultas de libros disponibles, el registro de los clientes y su mantenimiento, la selección de los libros requeridos mediante el método del carro de la compra y gestión integral de los pedidos.

### **2.2 Proceso de la conexión:**

Primer acceso: Al instalar la aplicación ya viene dado de alta un usuario administrador, para que se pueda empezar a dar de alta otros administradores e introducir los datos del stock de libros que se van a poner a la venta en la tienda.

Acceso de los clientes potenciales: Los clientes pueden examinar los libros que están en venta en la tienda antes de registrarse y/o autenticarse pulsando en la opción Libros que aparece en la parte superior. O pasar a registrarse o autenticarse en la parte izquierda de la pantalla de bienvenida o en la de lista de libros.

Acceso clientes registrados: Los clientes ya autenticados podrán realizar el pedido mediante el sistema de carro de la compra.

## **2.3 Mantenimiento de los datos**

### **2.3.1. Gestión de Usuarios**

El administrador además de las funcionalidades mencionadas, debe de poder hacer las altas, bajas y modificaciones de los usuarios.

- Altas de usuarios. Esta opción permitirá la introducción del identificador del usuario, si ya está en el sistema, se indicará esta circunstancia y se permitirá un nuevo intento. En caso de que el identificador no conste todavía en el sistema, se permitirá la introducción de los otros datos.
- Bajas de usuarios. Esta opción permitirá eliminar a un usuario de la base de datos. Se solicitará su identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario y se pedirá confirmación para borrarlo.
- Modificación de los usuarios. Esta opción permitirá modificar los datos de los usuarios bien porque se haya cometido algún error en la introducción de datos, bien porque hayan cambiado los datos desde el momento en que se introdujeron. Se solicitará el identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario que se podrán modificar. Una vez hechas las modificaciones será necesario apretar un botón para hacerlas efectivas.

### **2.3.2 Gestión de libros.**

- Altas de libros. Esta opción permitirá la introducción del identificador del libro, si ya está en el sistema, se indicará esta circunstancia y se permitirá un nuevo intento. En caso de que el identificador no conste todavía en el sistema, se permitirá la introducción de los demás datos.
- Bajas de libros. Esta opción permitirá eliminar a un libro de la base de datos. Se solicitará su identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario y se pedirá confirmación para borrarlo.
- Modificación de los datos de libros. Esta opción permitirá modificar los datos de los libros bien porque se haya cometido algún error en la introducción de datos, bien porque hayan cambiado desde el momento en que se introdujeron. Se solicitará el identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario que se podrán modificar. Una vez hechas las modificaciones será necesario apretar un botón

para hacerlas efectivas.

### **2.3.3 Gestión de pedidos.**

Se permitirán igualmente altas, bajas y modificaciones como se detallará a continuación, para que los administradores puedan mantener los datos referentes a los pedidos aunque normalmente serán introducidos por los clientes mediante el sistema de carro de la compra.

- Carro de la compra: Se trata de ciertas opciones disponibles en las pantallas de consulta de libros (tanto de lista de libros como de detalle) que permiten añadir un libro al carro de la compra y otra pantalla en la que se podrán eliminar libros del carro y confirmar el pedido en cuestión. Claro está que esto lo podrá realizar sólo un usuario registrado del que se tengan todos sus datos.
- Altas de pedidos. Esta opción permitirá la introducción del identificador del pedido, si ya está en el sistema, se indicará esta circunstancia y se permitirá un nuevo intento. En caso de que el identificador no conste todavía en el sistema, se permitirá la introducción de los demás datos.
- Bajas de pedidos. Esta opción permitirá eliminar a un pedido de la base de datos. Se solicitará su identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario y se pedirá confirmación para borrarlo.
- Modificación de los datos de pedidos. Esta opción permitirá modificar los datos de los pedidos bien porque se haya cometido algún error en la introducción de datos, bien porque hayan cambiado desde el momento en que se introdujeron. Se solicitará el identificador y, si no estuviera en la base de datos, se indicará esta circunstancia. En caso de estar, se mostrará el resto de los datos del usuario que se podrán modificar. Una vez hechas las modificaciones será necesario apretar un botón para hacerlas efectivas.

## 2.4 Búsquedas de datos.

Las habrá tanto de libros como de usuarios y pedidos, servirán para buscar y encontrar los registros de datos requeridos. Constarán de un listado por columnas. Para buscar el registro en cuestión dispondremos de dos métodos:

- Filtrado por palabras: En la cabecera de cada columna se podrán introducir palabras con las que se realizará un filtrado para que sólo aparezcan los registros que en dicho campo contenga dichas palabras.
- Ordenar datos por columnas: Si se pincha en la cabecera de la columna el listado se ordenará por dicha columna.

Una vez tenemos el libro en cuestión que estábamos buscando pincharemos en el botón de detalle que aparece en cada registro para pasar a la pantalla de detalle donde tendremos todos los datos relativos al registro.

Tendremos búsquedas relativas a libros, pedidos y usuarios.

## 2.5 Carro de la compra.

Ciertas opciones disponibles en las pantallas de consulta de libros (tanto en la búsqueda de libros como de en la pantalla de detalle) que permitirán añadir un libro al carro de la compra y otra pantalla denominada "carro de la compra" en la que se podrán:

- Eliminar libros del carro de entre los que hemos ido añadiendo anteriormente.
- Cambiar las unidades deseadas de cada libro en cuestión: Al añadir un libro al carro de la compra se añade una unidad de cada libro. En esta pantalla podremos cambiar el número de unidades, sin tener que pinchar repetidas veces en "añadir al carro de la compra".
- Confirmar el pedido en cuestión: Una vez confeccionado el pedido tendremos a la vista los importes de cada libro el total y el importe del iva. El pedido que hasta ahora es provisional se puede pasar a ser confirmado pulsando en el enlace "confirmar pedido". Es entonces cuando el pedido queda grabado en en la base de datos. Claro está que esto lo podrá realizar sólo un usuario registrado del que se tengan todos sus datos.

### 3 Actores:

Los actores que van a participar en el proceso básicamente son:

- Los clientes que consultarán los libros y harán pedidos.
  - **Cientes sin registrar:** Son los usuarios que acceden al sitio web y no se han identificado, ya sea porque no están registrados o porque se trata de clientes o administradores que todavía no se han identificado.
  - **Cientes registrados:** Son clientes que se han identificado, se les permitirá consultar los libros que están disponibles a la venta y podrán realizar pedidos mediante el sistema de carro de la compra.
  - **Administradores:** Se encargan de mantener el sistema realizando las siguientes tareas:
    - Introduciendo los libros en stock y eliminando los que no se vayan a vender.
    - Añadiendo y modificando los clientes y nuevos administradores que pueda tener el sistema. Aunque normalmente los clientes se darán de alta ellos mismos registrándose en el sistema.
    - Control de los pedidos, modificación de los datos o eliminación de la base de datos si procede.

## **4 Esquema de la aplicación:**

Este esquema refleja las pantallas o páginas de las que se va a componer la aplicación. Las páginas que se encuentran en un nivel inferior se pueden acceder desde las páginas que las preceden. En cada página se introduce la gestión de los permisos es decir el tipo de usuario que va a poder acceder a dichas páginas. Aparecerán duplicadas las páginas a las que puedan acceder distintos usuarios ya que serán variaciones de la misma página pero aparecerán distintas opciones según el usuario que acceda a las mismas.

### **Página Inicio.**

**Identificación de Usuarios (Cliente sin registrar).**

**Registro de usuarios(Cliente sin registrar).**

### **Búsqueda de libros.**

**I Identificación de Usuarios (Cliente sin registrar).**

**Registro de usuarios(Cliente sin registrar).**

**Consulta de Libro (Cliente sin registrar).**

**Consulta de Libro (Cliente Registrado)**

**Añadir al carro de la compra, (Cliente Registrado)**

**Ver Carro de la compra. (Cliente Registrado)**

**Consulta Detalle de Libro (Administrador)**

**Añadir al carro de la compra, (Administrador)**

**Ver Carro de la compra. (Administrador)**

**Búsqueda de pedidos. (Cliente Registrado).**

**Consulta de pedidos. (Cliente registrado, sólo pedidos propios).**

**Búsqueda de pedidos (Administrador) (Todos los pedidos).**

**Consulta de pedidos (Administrador).**

**Búsqueda de Usuarios (Administrador).**

**Consulta de Usuarios (Administrador).**

#### 4.1 Detalle de funcionalidades.

##### **Inicio. (Cliente sin registrar).**

Es la primera pantalla que aparece al entrar en la web en la cual aparecerán datos y comentarios de la empresa. Encontraremos un apartado para la identificación del usuario (login y contraseña)

##### **Opciones:**

Identificación de usuario **(Cliente sin registrar).**

Registrarse como usuario nuevo **(Cliente sin registrar).**

Búsqueda de Libros **(Cliente sin registrar).**

##### **Inicio. (Cliente Registrado).**

Es la primera pantalla que aparece al entrar en la web en la cual aparecerán datos y comentarios de la empresa. Encontraremos un apartado para la identificación del usuario (login y contraseña)

##### **Opciones:**

Búsqueda de Libros **(Cliente Registrado).**

##### **Inicio. (Administrador).**

Es la primera pantalla que aparece al entrar en la web en la cual aparecerán datos y comentarios de la empresa. Encontraremos un apartado para la identificación del usuario (login y contraseña)

##### **Opciones:**

Búsqueda de Libros **(Administrador).**

Búsqueda de Pedidos **(Administrador).**

Búsqueda de Usuarios **(Administrador).**

##### **Registro de usuario:**

En esta pantalla se introducen los datos del usuario, con la contraseña por duplicado para confirmación. Se verifica que el usuario no exista, se graba el nuevo usuario y se vuelve a la pantalla de inicio como cliente registrado. Si no se consigue realizar el registro se vuelve a la pantalla de inicio como cliente sin registrar.

##### **Opciones:**

Búsqueda de Libros **(Cliente sin registrar).**

##### **Búsqueda de Libros. (Cliente sin registrar).**

Se trata de una lista de libros que se puede filtrar por diversos campos (Título, Autor, Tema, etc). También se puede ordenar pinchando en la cabecera

de cada columna.

**Opciones:**

Detalle de Libro (**Ciente sin registrar**).

Identificación de usuario (**Ciente sin registrar**).

Registrarse como usuario nuevo (**Ciente sin registrar**).

**Búsqueda de Libros (Ciente Registrado).**

Se trata de una lista de libros que se puede filtrar por diversos campos (Título, Autor, Tema, etc). También se puede ordenar pinchando en la cabecera de cada columna.

**Opciones:**

Consulta de Libro (**Ciente Registrado**).

Búsqueda de Libros (**Ciente Registrado**).

Búsqueda de Pedidos (**Ciente Registrado**).

Añadir al Carro de la Compra (**Ciente**

**Registrado/Administrador**).

Ver Carro de la Compra (**Ciente Registrado/Administrador**).

**Búsqueda de Libros (Administrador).**

Se trata de una lista de libros que se puede filtrar por diversos campos (Título, Autor, Tema, etc). También se puede ordenar pinchando en la cabecera de cada columna.

**Opciones:**

Consulta de Libro (**Ciente Registrado**).

Búsqueda de Pedidos (**Administrador**).

Búsqueda de Usuarios (**Administrador**).

Añadir al Carro de la Compra (**Ciente**

**Registrado/Administrador**).

Ver Carro de la Compra (**Ciente Registrado/Administrador**).

Editar libro (**Administrador**).

Añadir Libro (**Administrador**).

**Consulta de Libros. (Ciente sin registrar).**

Se visualizan todos los datos de los libros pero sin poderse modificar.

**Opciones:**

Búsqueda de Libros (**Ciente sin Registrar**).

**Consulta de Libros (Ciente Registrado).**

Se visualizan todos los datos de los libros pero sin poderse modificar.

**Opciones:**

Búsqueda de Libros (**Ciente Registrado**).

Búsqueda de Pedidos **(Cliente Registrado)**.  
Añadir al Carro de la Compra **(Cliente Registrado/Administrador)**.  
Ver Carro de la Compra **(Cliente Registrado/Administrador)**.

### **Consulta de Libros (Administrador)**.

Mantenimiento de libros completo. Altas, bajas, modificaciones.

#### **Opciones:**

Búsqueda de Libros **(Cliente Registrado)**.  
Búsqueda de Pedidos **(Cliente Registrado)**.  
Búsqueda de Usuarios **(Cliente Registrado)**.  
Añadir al Carro de la Compra **(Cliente Registrado/Administrador)**.  
Ver Carro de la Compra **(Cliente Registrado/Administrador)**.  
Editar libro **(Administrador)**.

### **Ver carro de la compra (Cliente Registrado / Administrador)**.

Se visualizan los datos de los libros añadidos previamente al carro de la compra con sus importes unidades, total y cálculo de iva. Se permiten cambiar las unidades de cada libro, eliminar libros, recalcular total y confirmar pedido.

#### **Opciones:**

Búsqueda de Libros **(Administrador)**.  
Búsqueda de Pedidos **(Administrador)**.  
Búsqueda de Usuarios **(Administrador)**.

### **Búsqueda de pedidos (Cliente Registrado)**.

Sólo aparecen los pedidos del propio cliente registrado. Podrá eliminarse o cancelarse el pedido si no hay fecha de salida registrada.

#### **Opciones:**

Consulta de Pedido. **(Cliente Registrado)**.  
Búsqueda de Libros. **(Cliente Registrado)**.  
Búsqueda de Pedidos. **(Cliente Registrado)**.  
Eliminar Pedido. **(Cliente Registrado)**.

### **Búsqueda de pedidos (Administrador)**.

Aparecen todos los pedidos. Con filtro por Cliente, Fechas, etc.

#### **Opciones:**

Consulta de Pedido. **(Cliente Registrado)**.  
Búsqueda de Libros. **(Administrador)**.

Búsqueda de Usuarios. **(Administrador)**.  
Eliminar Pedido. **(Cliente Registrado)**.  
Editar Pedido. **(Administrador)**.

#### **Consulta de Pedidos (Cliente Registrado)**

Se muestran los datos del pedido. Se podrá cancelar el pedido si no hay fecha de salida.

##### **Opciones:**

Búsqueda de Libros **(Cliente Registrado)**.  
Búsqueda de Pedidos **(Cliente Registrado)**.  
Eliminar Pedido **(Cliente Registrado)**.

#### **Consulta de Pedidos (Administrador)**

Se muestran los datos del pedido. El pedido podrá ser anulado en cualquier caso.

##### **Opciones:**

Búsqueda de Libros **(Administrador)**.  
Búsqueda de Usuarios **(Administrador)**.  
Búsqueda de Pedidos **(Administrador)**.  
Eliminar Pedido **(Cliente Registrado)**.  
Editar Pedido **(Administrador)**.

#### **Búsqueda de Usuarios (Administrador)**

Aparecen todos los pedidos. Con filtro por Nombre, Dni, etc.

##### **Opciones:**

Consulta de Usuario **(Cliente Registrado)**.  
Búsqueda de Libros **(Administrador)**.  
Búsqueda de Pedidos **(Administrador)**.  
Eliminar Pedido **(Cliente Registrado)**.  
Editar Usuario **(Administrador)**.  
Añadir Usuario **(Administrador)**.

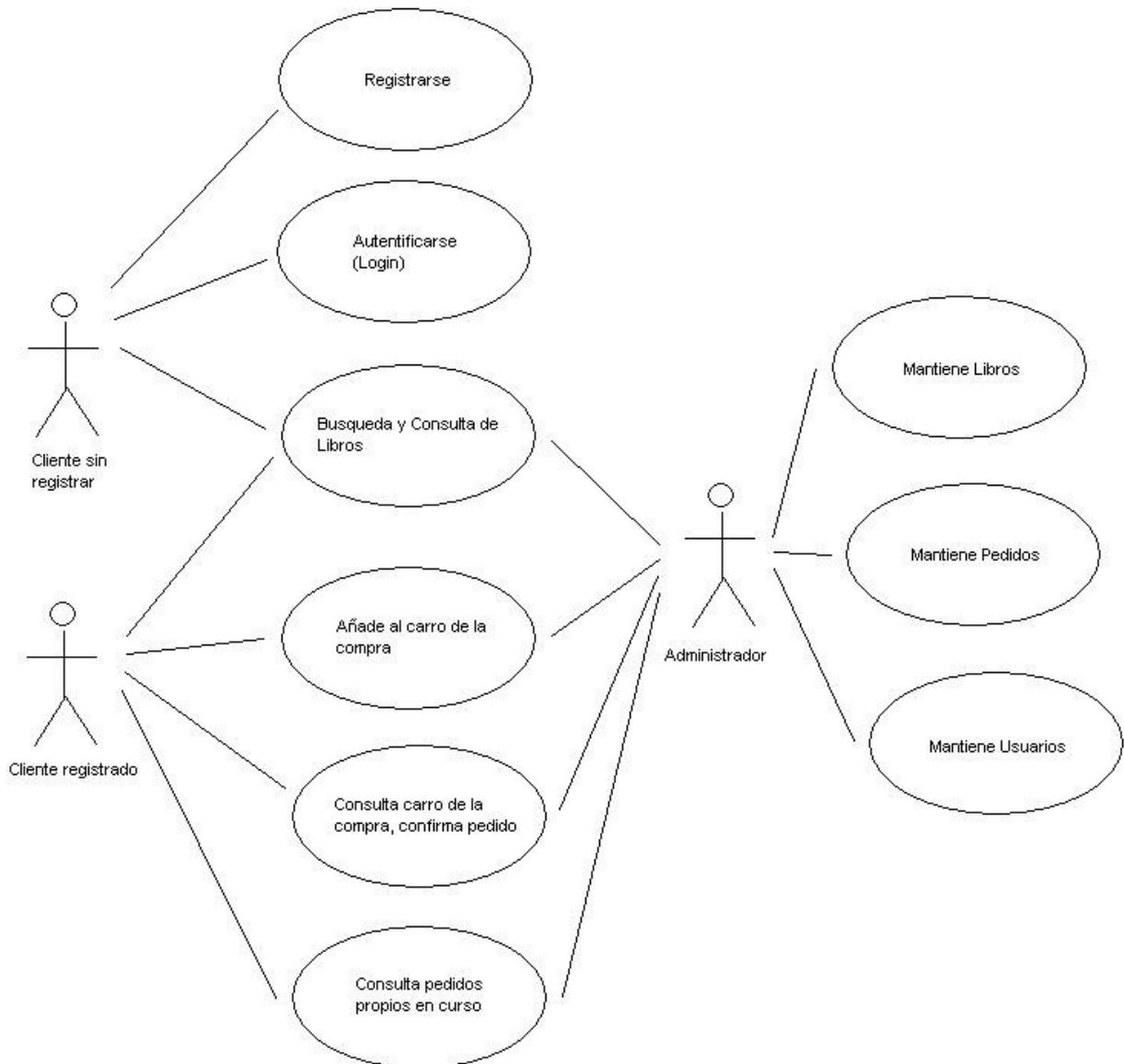
#### **Consulta de Usuarios (Administrador)**

Se muestran todos los datos del usuario.

##### **Opciones:**

Búsqueda de Libros **(Administrador)**.  
Búsqueda de Usuarios **(Administrador)**.  
Búsqueda de Pedidos **(Administrador)**.

## 4.2 Casos de uso según el actor.



## 5 Especificación formal de los casos de uso:

Para no introducir información redundante voy a agrupar los casos de uso en dos tipos:

- Los mantenimientos: Que son los encargados de mantener tablas de la base de datos con sus altas, bajas y modificaciones. En este tipo se engloban todos los casos de uso que tratan de añadir, modificar o borrar los datos de las tablas libros, usuarios y pedidos.
- Las Búsquedas: Se encargan de buscar datos mediante listas filtradas por ciertos campos. Su objetivo es encontrar un registro en concreto.

### 5.1 Caso de uso Mantenimiento:

**Funcionalidad:** Introduce, modifica, borra y/o consulta a usuarios, libros o pedidos de la base de datos.

**Papel dentro del trabajo del usuario:** Estos mantenimientos son utilizados por los distintos actores según se indica en el apartado Esquema de la aplicación. Hay dos tipos el de consulta y el de edición que como indican sus nombres permiten consultar sus datos y modificar respectivamente.

**Actores:** Cliente sin registrar, Cliente registrado, Administrador.

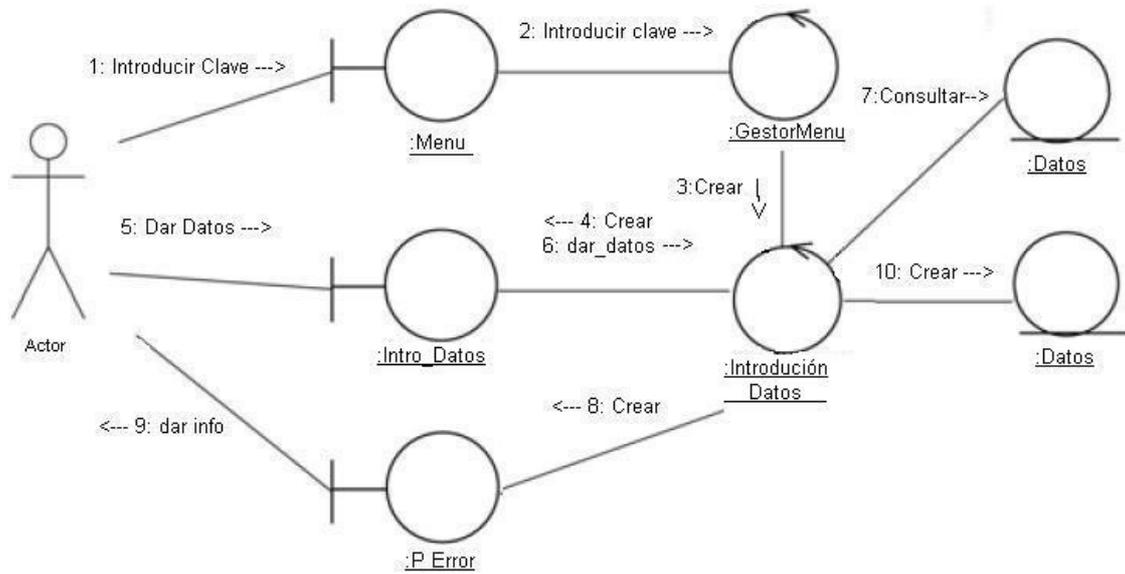
**Precondición:** El registro no está creado en la base de datos en el caso de alta. En el resto de casos debe estar ya creado en la base de datos.

**Postcondición:** El registro está incorporado en la base de datos en caso de creación. En caso de modificación, eliminación, consulta o bloqueo, se efectúa la operación o se emite un mensaje explicativo de las razones por las que no se ha podido llevar a cabo.

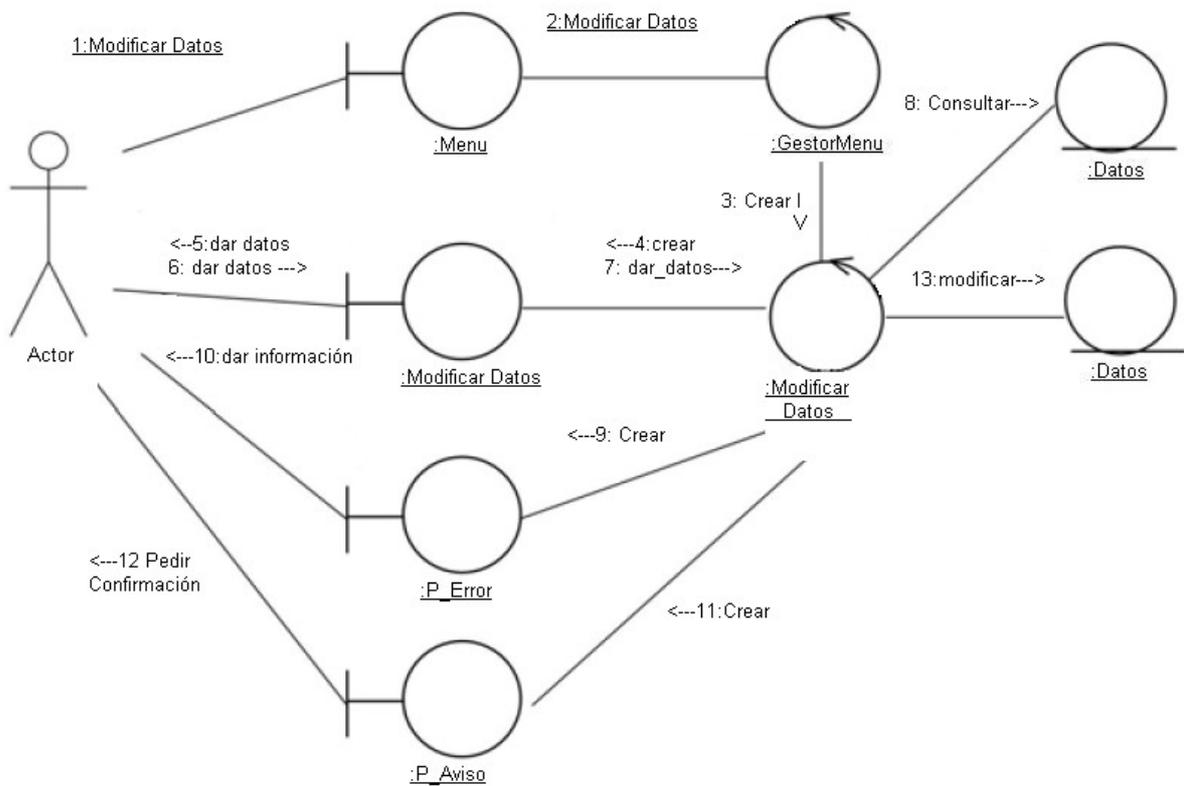
**Descripción:** Se puede haber seleccionado el registro a Modificar/Borrar en una búsqueda en cuyo caso aparecería el registro en pantalla y se permitiría modificar o pulsar el botón Borrar si el actor en cuestión pudiera realizar esas opciones

### 5.1.1 Diagramas de colaboración:

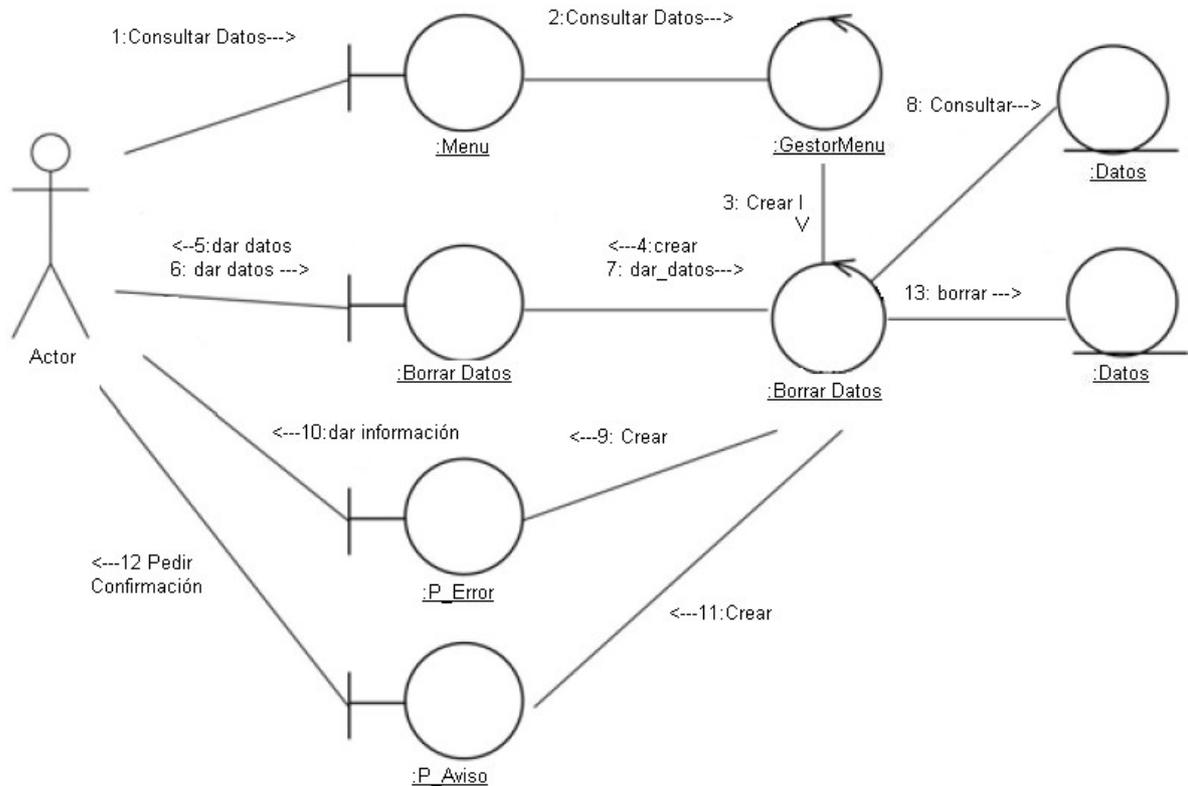
#### Alta:



#### Modificación:



### Borrar Registro:



### 5.2 Caso de uso Búsquedas:

**Funcionalidad:** Se trata de consultar listado de registros ya sea para ver dicha lista o para buscar un registro en concreto. Se podrán aplicar filtros y búsquedas por palabras.

**Papel dentro del trabajo del usuario:** Estas búsquedas se utilizan junto con los registros de usuarios, libros y pedidos.

**Actores:** Cliente sin registrar, Cliente registrado, Administrador.

**Precondición:** Debe haber registros creados ya que si no, no tendría utilidad.

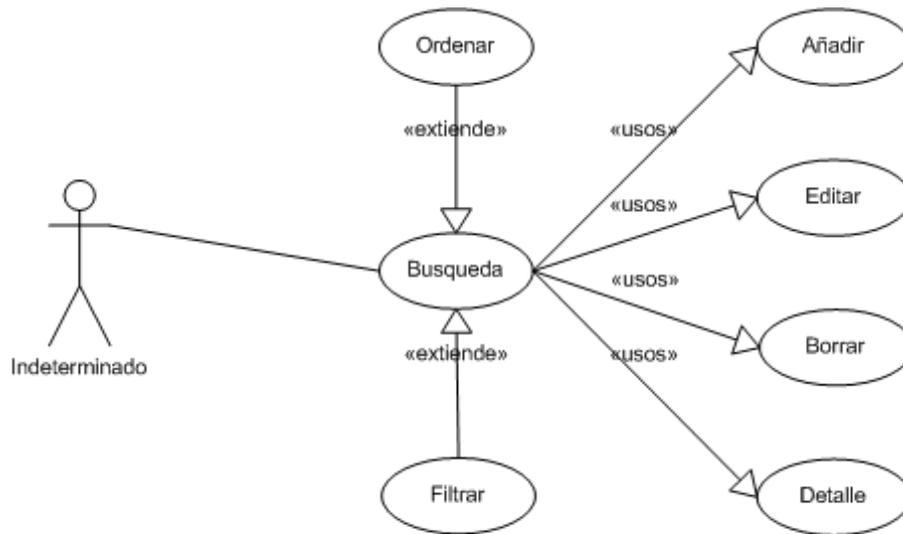
**Postcondición:** Ninguna.

**Descripción:** Aparece una lista de registros a modo de tabla. Con las siguientes funcionalidades que aparecen o no según el usuario que esté accediendo a la aplicación:

- Ordenar: Pulsando en las cabeceras se permite la ordenación por los

distintos campos.

- Filtrar: Encima de las cabeceras se podrán introducir palabras para aplicar filtros y seleccionar así el grupo de registros que nos interesa.
- Detalles: Para ver todos los campos del registro en cuestión.
- Editar: Para cambiar los datos del registro.
- Eliminar: Borrar definitivamente el registro de datos.
- Añadir nuevo: Nos permite introducir los campos de un nuevo registro.
- Añadir al carro de la compra: En el caso de libros se pueden añadir al carro de la compra.
- Identificarse o registrarse: En caso de no haberlo hecho anteriormente.



### 5.3 Caso de uso compra:

**Funcionalidad:** Consulta y realización de pedidos.

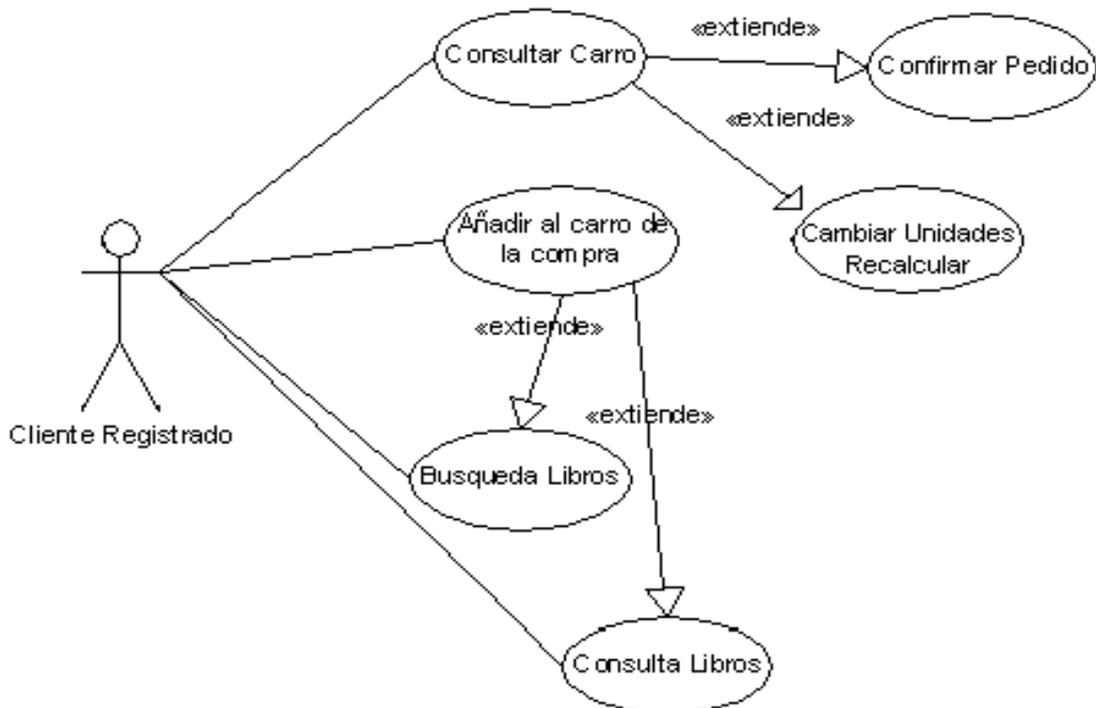
**Papel dentro del trabajo del usuario:** Se utiliza para realizar un pedido de libros en la tienda.

**Actores:** Cliente registrado, Administrador.

**Precondición:** El usuario debe estar registrado. Debe haber registros de libros creados ya que si no no tendría utilidad.

**Postcondición:** Ninguna.

**Descripción:** El usuario consulta la lista de libros, puede seleccionar un registro para ver los detalles. Si alguno le interesa comprarlo lo añade al carro de la compra. Cuando termina de revisar los libros si ha añadido alguno y le interesa puede pasar a ver el carro de la compra. En dicha pantalla aparece la lista de libros que se ha ido añadiendo, se podrá modificar el número de unidades de cada volumen, eliminar libros de la lista, recalcular los importes y confirmar el pedido en cuestión.



## 6 Datos a tratar en la aplicación.

### 6.1 Usuarios:

Se registran en la web dándose de alta a sí mismos. Darlos de baja será tarea del administrador. Por lo tanto damos lugar a dos casos de uso:

Registro Usuarios.  
Consulta y Baja.  
Login (identificación de usuario).

#### Atributos:

Nombre.  
Apellidos  
Dirección Entrega  
Calle  
Número  
Población  
Código Postal.  
Provincia.  
Correo electrónico.  
Contraseña.  
Administrador (SI/NO).

### 6.2 Libros:

El mantenimiento de los libros como altas, aumentos de stock (entradas de almacén) serán llevadas a cabo por el administrador aunque cuando los clientes realicen pedidos se descontarán automáticamente las unidades pedidas. Por lo tanto en relación a los libros tendríamos los siguientes casos de uso:

Mantenimiento de Libros (Altas, Bajas, Modificaciones).  
Consulta de libros.  
Confirmación, consulta y edición de pedidos.

#### Atributos:

ISBN  
Título  
Autor  
Tema  
Año Publicación  
Descripción.  
Stock.  
Precio.  
Unidades Vendidas.

### 6.3 Pedidos:

Los pedidos serán realizados por los clientes en dos fases con el sistema de carro de la compra. Se consultarán libros y se irán añadiendo al carro de la compra. La segunda fase será confirmar el pedido. Los casos de uso relacionados son:

Consulta de libros (Añadir al carro de la compra).  
Confirmación, consulta, anulación y edición de pedidos.  
Selección de pedido a consultar (Clientes. Filtrado por clientes.  
Administradores. Todos).

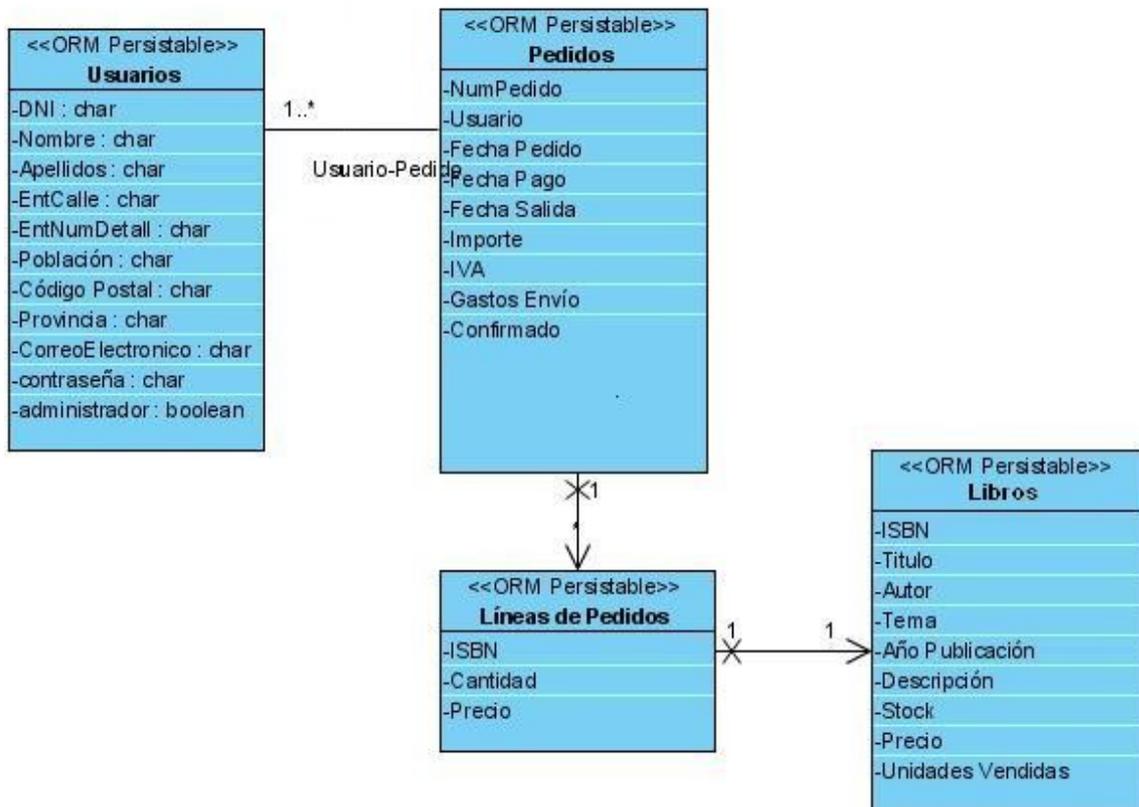
#### Atributos:

Número de Pedido.  
Usuario.  
Fecha Pedido.  
Fecha Pago.  
Fecha Salida.  
Importe  
IVA  
Gastos de Envío  
Confirmado Si/No.

#### Líneas de Pedido:

ISBN.  
Cantidad.  
Precio.

## 6.4 Diagrama de Entidades:



## 7 . Diseño.

### 7.1 J2EE standard de desarrollo en aplicaciones Web.

Dentro del mundo del desarrollo software profesional de grandes aplicaciones, JAVA es la principal opción hoy en día y dentro del amplio abanico de posibilidades, vamos a utilizar la tecnología J2EE, parte especialmente orientada al desarrollo de aplicaciones de servidor en entornos web.

Tan importante como la elección del lenguaje, es la elección de los objetos sobre los que va a funcionar. Para poder montar un sistema con J2EE, necesitamos un servidor de aplicaciones J2EE. Los servidores de aplicaciones para J2EE populares y de uso frecuente son:

- Oracle Application Server (Oracle).
- WebSphere (IBM).
- EAServer (Sybase Inc).
- WebLogic, (BEA Systems).
- JOnAS, no propietario.
- JBoss, no propietario.

Un servidor muy conocido y muy utilizado es el servidor Tomcat. Este servidor no permite el uso de objetos EJB como marca la tecnología J2EE pero muchas veces no se implementan estos objetos ya que en numerosos foros de debate se duda de su utilidad práctica. De todas formas yo utilizaré Jboss e implementaré EJBs de Sesión.

Si pasamos a comentar la parte del cliente que utilizará la aplicación final, éste solo necesitará de un navegador web para poder acceder (aunque también se le puede dotar de aplicaciones finales y dispositivos móviles).

Las interfaces web estarán desarrolladas en HTML, CCS ... en la parte cliente y JSP en la parte del servidor.

La puerta de entrada al sistema, por la parte del cliente, es a través de un servidor web con capacidad para gestionar Servlets y JSP. Como norma general, todo servidor de aplicaciones lleva incorporado un servidor web.

Finalmente, los datos se almacenan en un sistema gestor de base de datos. Hay multitud de SGBD en el mercado, entre los cuales se pueden destacar:

- Microsoft SQL Server, (Microsoft).
- Oracle, (Oracle Corporation).
- MySQL, no propietario
- PostgreSQL, no propietario

Todo esto viene a describir, de una manera simple, los componentes necesarios para desarrollar una aplicación J2EE. Ahora vamos a pasar a describir la arquitectura J2EE para el desarrollo de aplicaciones empresariales.

## 7.2 Estructura J2EE.

J2EE se basa, en su forma más extensa, en una arquitectura multicapa de cuatro niveles (puede hacerse una arquitectura de tres niveles dependiendo de las necesidades):

- **Capa Cliente:** Es la que proporciona la interfaz de usuario. En esta capa se recogen datos de usuario y se pasan a la capa siguiente, que los procesa. Se suele comunicar con la capa web, pero se pueden utilizar otras arquitecturas que permitan comunicarse directamente con la capa de negocio o de datos.
- **Capa Web:** Esta capa recibe los datos de usuario y se comunica con la capa de negocio, escogiendo los elementos que sean necesarios para procesar la petición. Es el puente entre el usuario y la capa de negocio.
- **Capa de Negocio:** Aquí es donde se almacenan las reglas de negocio. Esta capa es la que transforma las peticiones del cliente en respuestas, pasando previamente por la capa de datos para consultar los datos necesarios.
- **Capa de Datos:** Aquí es donde está almacenada la información de la empresa (bases de datos, ERPs...)

Para concluir con esta introducción a J2EE, se ha de recalcar que J2EE se basa en componentes, contenedores y conectores.

Los componentes que se utilizan a nivel de capa de negocio son los EJB, y a nivel de capa web son los Servlets y JSP. Los contenedores es el software encargado de gestionar los componentes.

### 7.3 Diseño arquitectónico del sistema.

El usuario interactúa con la aplicación Web a través del navegador. Como consecuencia de la actividad del usuario, se envían peticiones al servidor, donde se aloja la aplicación y que hace uso de una base de datos que almacena toda la información relacionada con la misma. El servidor procesa la petición y devuelve la respuesta al navegador que la presenta al usuario. Por tanto, el sistema se distribuye en tres componentes:

- El navegador, que presenta la interfaz al usuario.
- La aplicación, que se encarga de realizar las operaciones necesarias según las acciones llevadas a cabo por éste.
- La base de datos, donde la información relacionada con la aplicación se hace persistente.

Esta distribución se conoce como el modelo o *arquitectura de tres capas*.

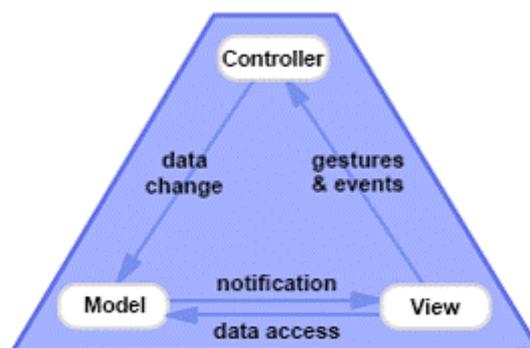
En este sistema y ortogonalmente a cada una de las capas de despliegue comentadas, podemos dividir la aplicación en tres áreas o niveles:

- Nivel de presentación: es el encargado de generar la interfaz de usuario en función de las acciones llevadas a cabo por este.
- Nivel de negocio: contiene toda la lógica que modela los procesos de negocio y es donde se realiza todo el procesamiento necesario para atender a las peticiones del usuario.
- Nivel de administración de datos: encargado de hacer persistente toda la información, suministra y almacena información para el nivel de negocio.

### 7.3.1 Nivel de presentación.

En el *nivel de presentación*, utilizaremos el modelo vista controlador 2 (MVC-2), una variación del clásico modelo MVC, que proporciona una multitud de beneficios al diseño, ya que separa preocupaciones del diseño (persistencia de datos y comportamiento, presentación, y control), disminuyendo la duplicación de código, centralizando el control, y haciendo la aplicación modificable de una manera mas fácil. Esta variación del MVC, introduce un servlet controlador entre el navegador y las páginas JSP o servlet de contenido.

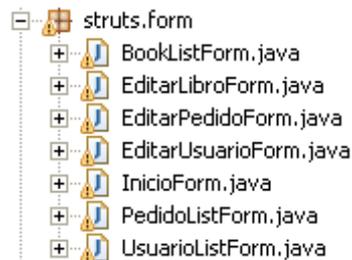
El controlador centraliza la lógica para expedir peticiones a la siguiente vista basada en la petición URL, parámetros de entrada, y estado de la aplicación. El controlador también maneja la selección de la vista, que desacopla las páginas JSP y servlets unos de otros. Las aplicaciones MVC2 son fáciles de mantener y extender, porque las vistas no se refieren unas a otras directamente. El servlet controlador del Modelo 2 proporciona un punto único de control para seguridad y registro, y a menudo encapsula datos de entrada en un formulario utilizable por el modelo MVC de fondo. Por estas razones, el Modelo 2 arquitectónico es recomendado para la mayoría de aplicaciones interactivas.



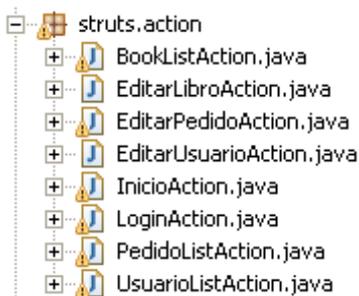
Este modelo estará implementado mediante el patrón Struts en el cual la entidad **Vista** está formada por páginas JSP, que utilizan html, css y la inserción de datos se realizará mediante las etiquetas de Struts. Las páginas que necesitaremos en la aplicación serán las siguientes:

- bookList.jsp
- detalleLibro.jsp
- editarLibro.jsp
- editarPedido.jsp
- editarUsuario.jsp
- index.jsp
- NuevoLibro.jsp
- NuevoUsuario.jsp
- PedidoList.jsp
- UsuarioList.jsp
- verCarro.jsp

El **Modelo** lo forman las clases que almacenan y sirven los datos a los componentes de la vista, durante la interacción con los usuarios. Por lo tanto deben tener propiedades para almacenar los datos y los métodos get y set adecuados para que los utilice struts.



Nos faltaría la parte del **controlador** que en struts está formado por los objetos Action. Se encargan principalmente de extraer los datos de los EJBs cargarlos en los objetos Form del modelo y lanzar el objeto vista jsp que corresponda.

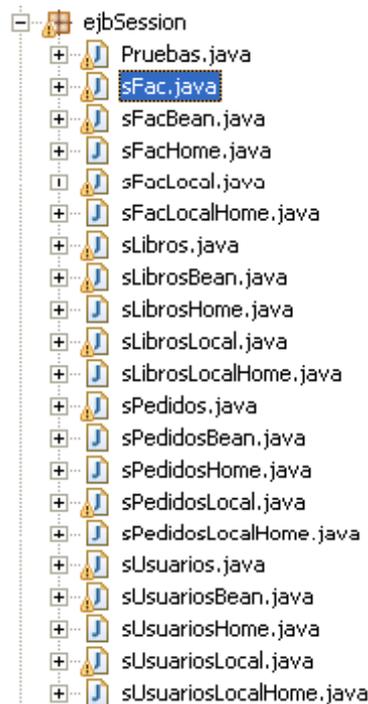


### 7.3.2 Nivel de negocio.

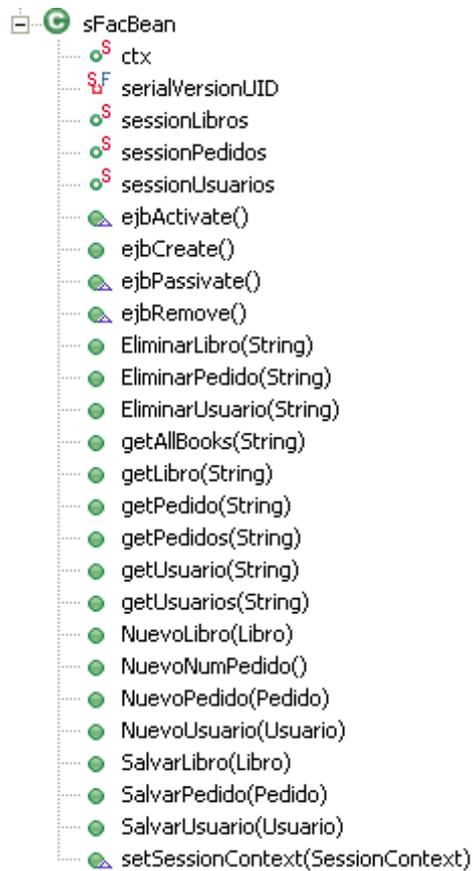
En el **nivel de negocio**, utilizaremos el patrón Session Facade. Session Facade encapsula la complejidad de las interacciones entre los objetos de negocio. Session Facade administra los objetos de negocio y proporciona una capa de servicio uniforme a los clientes. Este patrón abstrae las interacciones subyacentes entre los objetos de negocios y proporciona una capa que muestra solamente las interfaces precisas. Así, oculta al cliente la vista de las complejas interacciones entre los participantes.

Estos objetos de “fachada” interactuarán con EJBs de sesión que encapsularán la lógica de negocio y que pueden aportar valiosos servicios a la aplicación funcionando sobre el servidor JBOSS por ejemplo tolerancia a fallos, clustering, etc. Estos servicios no se tratarán en este TFC.

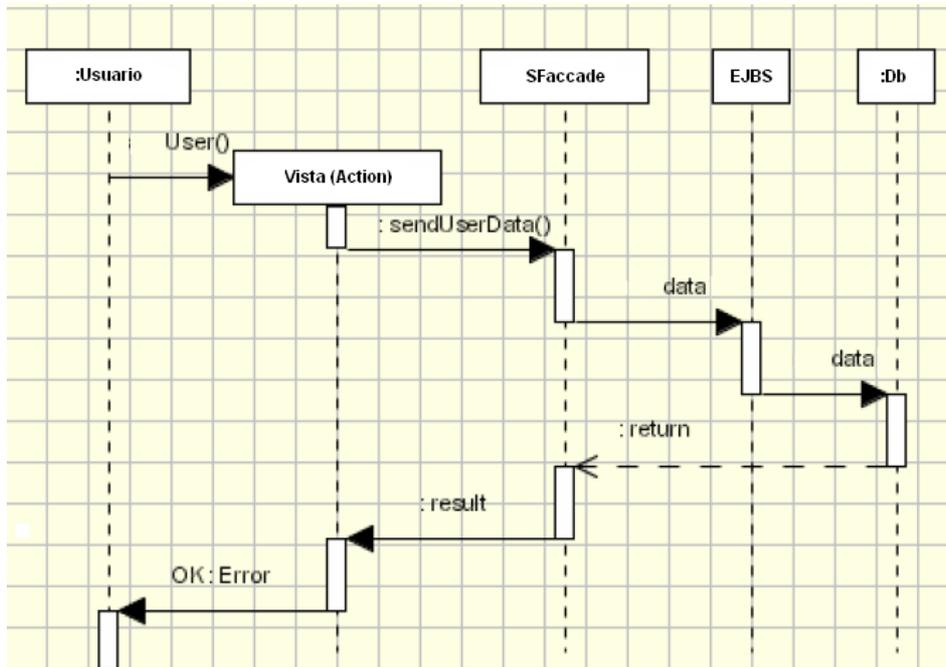
Este patrón estará implementado en nuestra aplicación por el EJB sFac y este se comunicará con los objetos de sesión de libros, usuarios y pedidos.



Es interesante ver aquí la lista de métodos que contiene el objeto sfac ya que representa todas las interacciones entre la vista y los objetos de negocio, al igual que representa el flujo de los objetos de datos que se obtienen de la base de datos y también los que se almacenan.



Ahora incluyo un diagrama de secuencia que expone el flujo de datos entre los distintos niveles y objetos del que hablábamos al exponer los métodos del objeto sFac.



En esta tabla vamos a reflejar las acciones que se van a realizar en cada capa de la aplicación en base a las paginas que puede utilizar el usuario.

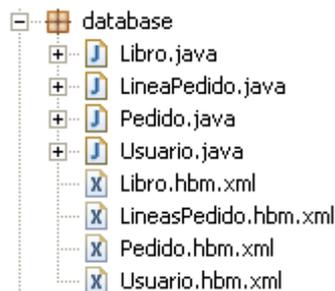
Vista	Negocio	Datos	EJB
Inicio(Login)	Validar Usuario	getUsuario	sLibros
Inicio			
Registrarse	Validar datos Usuario	GetUsuario	sLibros
	Guardar nuevo Usuario	NuevoUsuario	sLibros
BusquedaLibros	Lista de libros	getBooks	sLibros
Detalle de Libros	Un Libro	getBook	sLibros
	Nuevo Libro	Añadir Libro	sLibros
	Modificar Libro	setBook	sLibros
Añadir al Carro	Añadir Libro al carro	NuevoPedido	sPedidos
		AñadirLibroPedido	sPedidos
Ver Carro de la compra	Lista Pedido	getPedido	sPedidos

	Confirmar Pedido	setPedido	sPedidos
	Eliminar Libro del carro	EliminarLibroPedido	sPedidos
BusquedaPedidos	Lista de Pedidos	getPedidos	sPedidos
Detalle de Pedidos	Un Pedido	getPedido	sPedidos
	Nuevo Pedido	Añadir Pedido	sPedidos
	Modificar Pedido	setPedido	sPedidos
BusquedaUsuarios	Lista de Usuarios	getUsuarios	sUsuarios
Detalle de Usuarios	Un Usuario	getUsuario	sUsuarios
	Nuevo Usuario	Añadir Usuario	sUsuarios
	Modificar Usuario	setUsuario	sUsuarios

### 7.3.3 Nivel de datos.

En el **nivel de datos**, utilizaremos una herramienta ORM (Object Relational Mapping) llamada Hibernate, que nos sirva de puente entre el mundo de los objetos y de las bases de datos relacionales, suministrándonos persistencia y servicios de consulta. El sistema de gestión de base de datos SQL se apoya en JDBC. La base de datos que voy a utilizar va a ser MySql.

Hibernate utiliza unos ficheros de configuración para establecer una equivalencia entre registros de la base de datos y objetos. Por lo tanto cuando realizamos consultas, en vez de obtener conjuntos de registros, obtenemos colecciones de objetos. Los objetos de datos y las definiciones de hibernate hbm serán los siguientes:



#### **7.4 Conclusiones del diseño.**

En definitiva con este diseño maximizamos la reutilización del código y minimizamos la duplicación del mismo. Por ejemplo cada entidad del negocio tiene un simple componente en la capa de negocio, y todos los componentes de la capa de presentación que quieren acceder a esta entidad de negocio lo hacen a través del mismo componente de la capa de negocio.

También es posible modificar la aplicación haciendo solo cambios en una capa, dejando las otras dos inalteradas por el cambio. Por ejemplo es posible modificar la capa de presentación de una aplicación web sin afectar a las capas de negocio y acceso a datos.

Como cada capa es ahora independiente de las demás ello permite emplear diferentes grupos de programadores, con diferentes especialidades, para cada capa. Esto reduce la necesidad de desarrolladores especializados en distintos campos. Por ejemplo, podemos tener desarrolladores especializados en la capa de presentación (HTML y CSS), y desarrolladores especializados en la capa de negocio (Java).

## 8 Tecnología a utilizar:

### 8.1 Recursos necesarios.

- Un servidor de bases de datos.
- Un ordenador o más centrales que hacen las funciones de servidores de aplicaciones ejecutando la aplicación.
- N ordenadores clientes para conectarse mediante un navegador Web a los ordenadores centrales.
- Además, todos los ordenadores tienen que estar conectados a los ordenadores centrales mediante una red local o una red de gran alcance.

### 8.2 Tecnología de desarrollo utilizada.

Todo el proyecto se va a mover sobre el estándar Java 2 Enterprise Edition, utilizando el diseño multicapa ya descrito. Voy a enumerar las tecnologías utilizadas por capas:

- **Capa de presentación:** He optado por el patrón Struts para implementar MVC2. Es el patrón más utilizado actualmente y aunque existen evoluciones como Java Server Faces quizás son demasiado nuevas para igualar la fiabilidad de Struts.
- **Capa de negocio:** En esta capa utilizaré los Enterprise Java Beans de sesión ya que aunque esta aplicación no es grande en principio puede irse ampliando y/o recibir un aumento del número de visitas que requiera de servicios de balanceo de carga o la necesidad de implementar un servicio de alta disponibilidad.

Para implementar la tecnología de EJBs es necesario utilizar un servidor de aplicaciones que lo soporte. En este caso utilizaremos JBOSS.

- **Capa de Datos:** Utilizo Hibernate para la conversión o mapeo objeto-relacional y como medio de acceso a la base de datos MySQL. Ambas tecnologías son referentes indiscutibles del software libre.

**IDE::** Eclipse. Es el entorno con el que estoy familiarizado y del que tengo mejores referencias. Junto con el uso de Jboss-IDE y la herramienta packaging como método de despliegue.

### 8.3 Diseño de la interfaz de Usuario.

#### 8.3.1 Normas básicas:

El diseño de la interfaz de usuario vendrá limitado por el entorno de presentación del cliente, que será un navegador Web.

Para que la interfaz de usuario sea coherente, es necesario que cada dato se presente de la misma manera en todos los formularios en los que aparece.

La interfaz va a ser muy simple basándose siempre que sea posible en enlaces de hipertexto y cajas de texto para la introducción de los datos. Si hubiera algún texto memo de varias líneas utilizaríamos una caja de texto con desplazamiento vertical, evitando también en la medida de lo posible que aparezcan barras de desplazamiento horizontal.

Los campos de longitud fija se presentarán como campo de texto cuando son sólo de salida; cuando son también de entrada, si el número de valores posibles es reducido, el valor se escogerá de una lista (una lista, combo, si se pueden añadir valores y una lista ordinaria, en caso contrario); en lugar de una lista ordinaria de menos de 5 valores fijos se utilizará un conjunto de botones de radio. Para datos booleanos, utilizaremos check boxes.

Para cambiar de una página a otra, como ya hemos dicho utilizaremos simples enlaces de hipertexto.

Los diálogos de aviso y error, se mostrarán en una ventana de mensajes modal, que nos obligará a aceptarlo para continuar.

Las listas de datos se mostrarán en formato de tabla, con al menos una columna seleccionable si fuera necesario.

### **8.3.2 Descripción de la interfaz de usuario.**

La interfaz del usuario se encuentra claramente dividida en cuatro partes:

- La parte superior, se compone de una barra de título con el anagrama de la librería y con los enlaces a los demás apartados de la aplicación.
- En la parte izquierda está el cuerpo de la página donde aparecerán todos los datos de interés para el usuario.
- En la parte derecha tendremos el dialogo de identificación cuando sea necesario.
- En la parte inferior pondremos un pie de página con texto informativo.

Siguiendo un criterio de agilidad de las páginas y eficiencia en la carga se podría pensar que el diseño debería ser minimalista, sin imágenes que retrasen la carga de la página, ni datos no necesarios que distraigan la atención del usuario. Pero el objetivo de la página que nos ocupa es vender y debe resultar un entorno agradable y estéticamente atractivo que nos incite a acudir con frecuencia y comprar. Por ello le he añadido unas imágenes para mejorar su aspecto.

## 8.4 Presentación de páginas de ejemplo:

### 8.4.1 Página de Inicio:



Librería Global inauguró su primera librería en 1923 en Madrid. En la actualidad cuenta con doce centros más: dos en Bilbao, uno en Barcelona, cuatro en Madrid, Vigo, Sevilla, Vitoria - Gasteiz, Valencia y Gijón. De la mano de Espasa-Calpe editorial a la que pertenece desde sus

La pantalla de inicio da la opción de identificarse en la misma página o pasar a registrarse como usuario nuevo o también pasar a consultar los libros disponibles sin haber hecho ninguna de las dos cosas antes mencionadas.

## 8.4.2 Página de búsqueda de libros:

Librería Internacional Global - Windows Internet Explorer

http://localhost:8080/WebLibros/bookList.do?orden=Titulo

Librería Internacional Global

Librería Internacional "Global"

Libros Usuarios Pedidos Bienvenido admin

### Listado de Libros

Titulo	Autor	Precio			
Administración de Sistemas Linux	Dee-Ann Leblanc	54.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
Angeles y Diablos	Dan Brawn	33.34	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
El Código da Vinci	Dan Brawn	43.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
Guia de Referencia y Aprendizaje	Matt Welsh	21.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
La sonrisa etrusca	Jose Luis SanPedro	42.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
Los pilares de la tierra	Ken follet	28.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
Manual de J2EE	Orreily	56.32	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
Red Hat Linux 7	Alfonso Gazo Cervero	32.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>
TCP/IP	Dr. Sidnie Feit	21.0	<a href="#">Editar</a>	<a href="#">Eliminar</a>	<a href="#">Añadir al Carro</a>

Buscar

Añadir un nuevo Libro

Hay 2 libros en el carro de la compra.  
[Ver carro de la compra](#)

Esta es la página de búsqueda accedida por un usuario administrador por eso aparecen también todas las opciones que permiten la administración de los datos sobre libros.

### 8.4.3 Pantalla de edición de los datos de un libro:

Librería Internacional Global - Windows Internet Explorer

http://localhost:8080/WebLibros/editarLibro.do?do=e... Google

Librería Internacional Global

Librería Internacional "Global"

Libros Usuarios Pedidos Bienvenido admin

### Editar Libro

Título:	Guia de Referencia y Aprendizaje
Autor:	Matt Welsh
Tema:	Linux
Publicación:	2001
Descripcion:	
Stock:	34
Precio:	21.0
Unidades Vendidas:	0

Guardar Cambios [Volver a la lista de libros](#)

Librería Global inauguró su primera librería en 1923 en Madrid. En la actualidad cuenta con doce centros más: dos en Bilbao, uno en Barcelona, cuatro en Madrid, Vigo, Sevilla, Vitoria - Gasteiz, Valencia y Gijón. De la mano de Espasa-Calpe, editorial a la que pertenece desde sus orígenes, Librería Global afronta su expansión en las principales capitales españolas. Con este proyecto Librería Global se convertirá en el primer grupo nacional de librerías. Las señas de identidad de Librería Global se mantienen inalteradas desde su fundación: librería general, por la vocación de universalidad de sus fondos; librería especializada, por la diversidad de sus cientos de miles de referencias bibliográficas; librería tradicional, por el deseo de ofrecer un lugar de encuentro de autores, lectores y libreros; librería moderna, por su atención al desarrollo de las nuevas tecnologías y su temprana asimilación de las más modernas técnicas del comercio librero.

Hay 2 libros en el carro de la compra.

Internet 100%

Esta es la pantalla de edición de los datos de un libro muy parecida a la que será la de añadir un libro.

#### 8.4.4 El carro de la compra:

Librería Internacional Global - Windows Internet Explorer

http://localhost:8080/WebLibros/editarPedido.do?do=verCarro

Librería Internacional Global

Librería Internacional "Global"

Libros Usuarios Pedidos Bienvenido admin

### Carro de la Compra

Isbn	Título	Precio	Cantidad	Importe		
3	Los pilares de la tierra	28.0	1	28.0	+ -	Eliminar
7	Angeles y Diablos	33.34	1	33.34	+ -	Eliminar
10	Administración de Sistemas Linux	54.0	1	54.0	+ -	Eliminar
11	Red Hat Linux 7	32.0	1	32.0	+ -	Eliminar
<b>Iva</b>		23,574				
<b>Importe</b>		170,914				

[Volver a la lista de libros](#)

Confirmar Pedido

Librería Global inauguró su primera librería en 1923 en Madrid. En la actualidad cuenta con doce centros más: dos en Bilbao, uno en Barcelona, cuatro en Madrid. Vigo. Sevilla. Vitoria - Gasteiz. Valencia y Gijón. De la mano de

Internet 100%

Esta es la pantalla del carro de la compra que permite cambiar las unidades de cada libro mediante los controles + y -. Cada vez que se pulsan se recalcula el IVA y el Importe. El enlace de volver a la lista de libros, realiza la función de botón cancelar saliendo de la pantalla de carro de la compra sin aceptar el pedido.

#### 8.4.5 Pantalla de edición de datos del usuario.

Librería Internacional Global - Windows Internet Explorer

http://localhost:8080/WebLibros/editarUsuario.do?do=editarUsuario&identificador=admin

Librería Internacional Global

Librería Internacional "Global"

Libros Usuarios Pedidos Bienvenido admin

### Editar Usuario

Dni:	<input type="text" value="2"/>
Nombre:	<input type="text" value="administrador"/>
Apellidos:	<input type="text" value="admin"/>
Calle:	<input type="text" value="calle"/>
Número:	<input type="text" value="nu datelle"/>
Población:	<input type="text" value="cuenca"/>
Código Postal:	<input type="text" value="16001"/>
Correo Electrónico:	<input type="text" value="admin@dipucuenca.es"/>
Identificador:	<input type="text" value="admin"/>
Password:	<input type="text" value="admin00"/>
Administrador (s/n):	<input type="text" value="s"/>

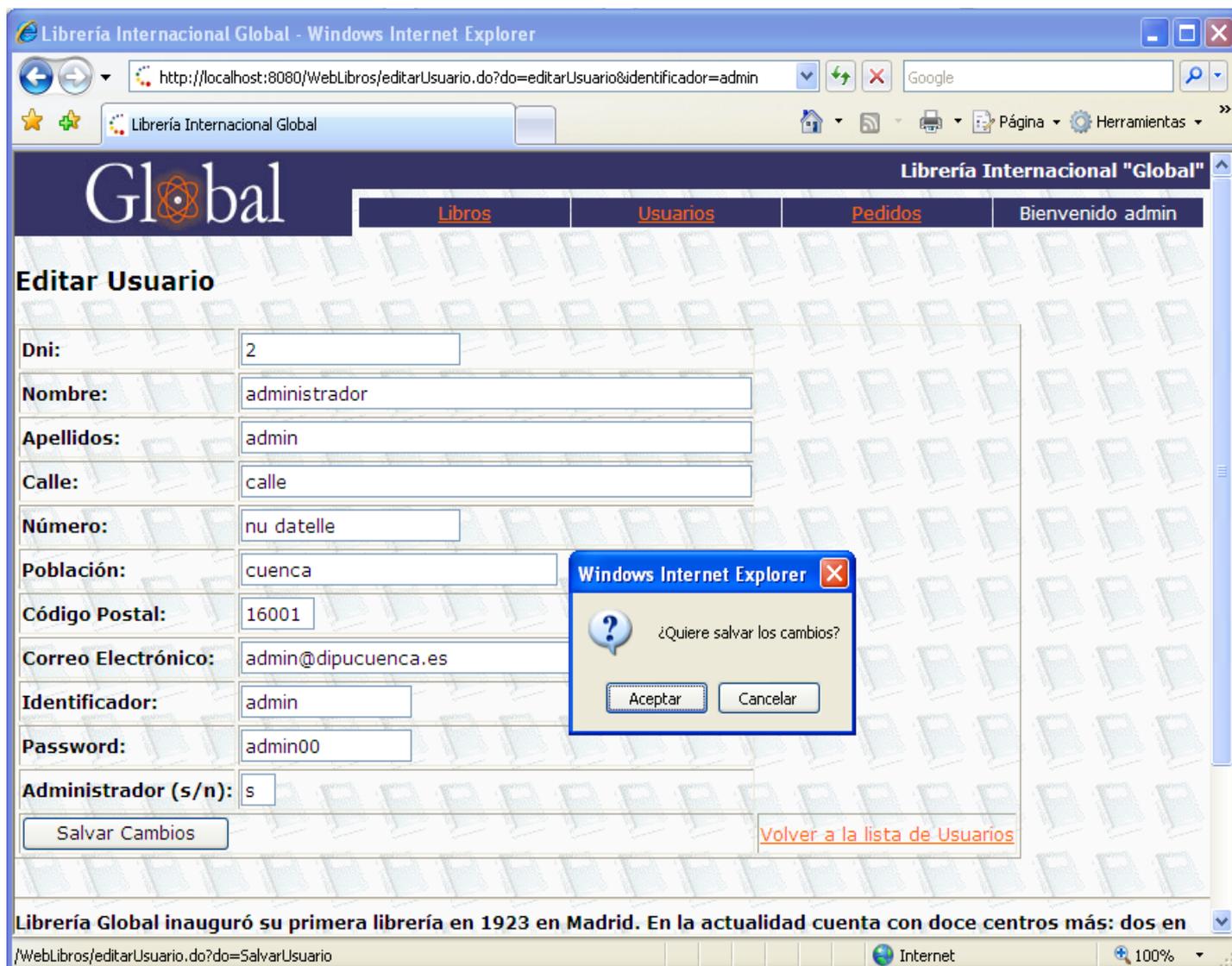
[Volver a la lista de Usuarios](#)

Librería Global inauguró su primera librería en 1923 en Madrid. En la actualidad cuenta con doce centros más: dos en

http://localhost:8080/WebLibros/UsuarioList.do?orden=Nombre

Esta sería la pantalla para editar añadir usuarios. El campo Administrador indica si el usuario es administrador o no. La pantalla de registro de usuarios será muy parecida pero este campo no aparecerá.

### 8.4.6 Mensajes de confirmación.



Al pulsar en los botones de salvar cambios, confirmar pedido, etc nos aparecerán unos mensajes pidiendo confirmación de la operación.

### 8.4.7 Control de errores:

Todos los accesos a la base de datos llevan un control de errores que presenta un mensaje en pantalla parecido al que se ve en la imagen. Pertenecer a un error producido al eliminar un usuario.

The screenshot shows a Windows Internet Explorer browser window displaying the 'Lista de Usuarios' page of the 'Librería Internacional Global' website. The browser's address bar shows the URL 'http://localhost:8080/WebLibros/UsuarioList.do'. The website's navigation menu includes 'Libros', 'Usuarios', 'Pedidos', and 'Bienvenido admin'. The main content area features a table of users with columns for 'Nombre', 'Apellidos', and 'Identificador', along with 'Editar' and 'Eliminar' buttons for each user. Below the table is a search box and a link to 'Añadir un nuevo Usuario'. On the right side, a grey error message box is displayed, stating: 'Se ha producido un error inesperado. Consulte con el WebMaster. Detalles: Al eliminar usuario. Index: 0, Size: 0; nested exception is: java.lang.IndexOutOfBoundsException: Index: 0, Size: 0'. Below the error message is an illustration of a cat sitting on a stack of books. At the bottom of the browser window, the status bar shows 'Internet' and '100%' zoom level.

Nombre	Apellidos	Identificador		
administrador	admin	admin	<a href="#">Editar</a>	<a href="#">Eliminar</a>
Gustavo	Visier Cabezudo	gustavo	<a href="#">Editar</a>	<a href="#">Eliminar</a>
Victoria	Berenguer Peñuelas	victoria	<a href="#">Editar</a>	<a href="#">Eliminar</a>
juan	juan	juan	<a href="#">Editar</a>	<a href="#">Eliminar</a>

[Añadir un nuevo Usuario](#)

Se ha producido un error inesperado. Consulte con el WebMaster. Detalles: Al eliminar usuario. Index: 0, Size: 0; nested exception is: java.lang.IndexOutOfBoundsException: Index: 0, Size: 0



## 9 Estructura del proyecto.

### 9.1 Estructura de directorios.

He separado la aplicación en dos proyectos. Uno con el nombre de WebLibros contiene los elementos de la vista junto con el patrón struts y el otro denominado WebLibrosEJB, lleva los EJBs de sesión junto con los objetos de base de datos e Hibernate. Esto es una práctica habitual para impedir que los ejbs aparezcan en la parte pública del servidor web.

#### WebLibros

**src:** En este directorio están los fuentes de todas las clases java, divididas en cuatro paquetes:

- **struts:** Con dos subdirectorios en uno se almacenan las clases action y en el otro los forms.

**WebRoot:** Aquí se almacenan todas las clases compiladas y los demás ficheros que componen el proyecto. Dentro de el encontramos lo siguiente:

- **jsp:** Ficheros JSP que integran las páginas web de la aplicación.
- **META-INF:** Ficheros xml sobre los EJBs y JBOSS.
- **WEB-INF:** Archivos relativos al patrón struts. Además contiene los siguientes subdirectorios:
  - **clases:** Compilados de java con la misma estructura que el directorio src con los fuentes.
  - **lib:** Todas las librerías necesarias para el funcionamiento de la aplicación.

#### WebLibrosEJB

**src:** En este directorio están los fuentes de todas las clases java, divididas en cuatro paquetes:

- **Database:** Donde están los objetos persistentes y los archivos de definición de mapeo de hibernate para dichos objetos.
- **EjbSession:** Donde se encuentran definidos los EJBs de la capa de negocios y el EJB que ejerce de Sesión Facade.
- **Útil:** Aquí encontramos un objeto de utilidad que se utiliza con hibernate.

**WebRoot:** Aquí se almacenan todas las clases compiladas y los demás ficheros que componen el proyecto. Dentro de el encontramos lo siguiente:

- **META-INF:** Ficheros xml sobre los EJBs y JBOSS.
- **WEB-INF:**
  - **clases:** Compilados de java con la misma estructura que el directorio src con los fuentes.
  - **lib:** Todas las librerías necesarias para el funcionamiento de la aplicación.

## **9.2 Detalles de implementación:**

### **9.2.1 Detalles Generales.**

#### **Vista:**

En principio he utilizado páginas html con la utilización de estilos css para fijar un diseño básico en las cuestiones de tipo de letra y colores.

A partir de ahí e incluido los tags del patrón struts para incluir los datos dinámicos procedentes de la base de datos como los mensajes y campos mostrados pertenecientes a la sesión del usuario, como son:

- El usuario en sí una vez identificado.
- El carro de la compra o pedido en curso mientras está añadiendo los libros en cuestión.
- Los mensajes de error que se puedan producir.

Para evitar la duplicación de páginas jsp he utilizado los tags de lógica de struts para que se muestren o no las distintas opciones permitidas para cada tipo de usuario.

#### **Capa de Negocios:**

En referencia a la aplicación del patrón Session Facade sólo he definido un objeto fachada ya que generar varios para una aplicación reducida como esta desvirtuaría el objetivo de este patrón de simplificar el acceso a la capa de negocios. No se descartaría ampliar o separar este objeto fachada si la aplicación creciera en gran medida.

Los EJBs se dedican mayormente a la lectura y almacenaje de los objetos de datos en la base de datos utilizando hibernate para conservar la filosofía de la orientación a objetos.

#### **Capa de datos:**

El lenguaje HQL de hibernate me ha parecido muy interesante ya que simplifica en gran medida las sentencias SQL además del tratamiento de los objetos.

#### **Control de errores:**

Los errores que se puedan producir al acceder a los EJB y a su vez a la base de datos producen excepciones que se trasladan a la capa Vista para mostrar mensajes que avisen al usuario de que se ha producido un error y por eso no se ha realizado la operación requerida.

### **9.2.2 Empaquetado y despliegue de la aplicación.**

Para montar los archivos war y jar utilizo la herramienta packaging de eclipse. La configuración de los paquetes va almacenada en el archivo packaging-build.xml que se encuentra en el directorio raíz de cada uno de los proyectos.

Si queremos generar los archivos hay que seleccionar el proyecto dentro de eclipse y con el botón derecho ejecutar Run-Pakaging (Ctrl+Mayus+F2).

Una vez generados los archivos War y rar, se pueden desplegar seleccionando los archivos y con el botón derecho Deploy to. (Ctrl+Mayus+F3).

### **9.3 Valoración económica.**

Creo que cualquier tipo de valoración económica que se llevase a cabo no sería realista ya que una gran parte del tiempo dedicado ha sido para adquirir los conocimientos necesarios sobre la tecnología y por supuesto el tener practica con las herramientas de desarrollo acorta muchísimo los plazos.

Otro dato a tener en cuenta es que la programación orientada a objetos y el entorno J2EE en general está pensado para utilizar una gran reutilización de código. Esto no ha ocurrido ya que se ha partido desde cero a excepción de la utilización del patrón struts.

Por lo tanto no me atrevo a realizar ningún cálculo sobre cuanto tiempo costará realizar el proyecto ni su valoración económica.

## **10 Conclusiones:**

Quisiera remarcar la magnitud de los conocimientos que he tenido que adquirir y aplicar en el proyecto, muchos de ellos desde cero (sobre todo para un estudiante de la ingeniería técnica). Como por ejemplo html, tags jsp, Enterprise Java Beans, patrones, diseño multicapa, hibernate etc, etc. De hecho seleccioné esta área para mi TFC porque tenía mucho interés en el aprendizaje de todas estas materias y creo que el esfuerzo requerido ha merecido la pena.

He tenido que abordar las fases de análisis y diseño llevando en paralelo la adquisición de los conocimientos técnicos requeridos. Pasando posteriormente a la fase de implementación intensiva para conseguir un aplicación que cumpla en la medida de lo posible con los requerimientos del TFC.

Sería acertado decir que no he conseguido seguir estrictamente el plan de trabajo y las tareas que propuse en la PAC1 ya que no he tenido claro el diseño y la estructura de la aplicación hasta que he adquirido una visión global de las tecnologías y patrones que se han utilizado en el proyecto.

Aunque no sé si cumplirá con las expectativas estoy satisfecho del trabajo realizado y de los conocimientos adquiridos.

### **10.1 Valoración del diseño:**

Aunque no me extendido mucho en funcionalidades necesarias para el funcionamiento de una tienda real en Internet creo que he conseguido un ejemplo del funcionamiento de todas las tecnologías requeridas para ello.

Tampoco he empleado un diseño estético que se salga de lo común. Han primado la sencillez de uso y la claridad en la presentación de los datos.

He creído importante permitir a los usuarios no identificados ver y valorar la colección de libros que ofrece la tienda ya que el paso de registrarse puede echar atrás a los usuarios antes de apreciar si les merece la pena o no entrar en el sitio web.

## 11 Bibliografía:

### **INGENIERIA DEL SOFTWARE. Un enfoque practico**

Pressman, Roger S.

ISBN/EAN: 8448132149 Año: 2001

Editorial: McGraw- Hill / Interamericana España Edición: 5

<http://www.uml.org/>

<http://java.sun.com/blueprints/>

### **PROGRAMACION JAVA SERVER CON J2EE EDICION 1.3. Profesional**

Allamaraju, Subrahmanyam / Beust, Cedric / Davies, John / Y otros

ISBN/EAN: 8441513589 Año: 2002

Editorial: Anaya, Grupo Editorial Edición: 1

<http://java.sun.com/products/jsp/>

### **JBOSS 3.2 DEPLOYMENT AND ADMINISTRATION**

Meeraj Kunumpurath

ISBN/EAN: 1590592816 Año: 2004

Edición: 1 Idioma: Inglés

## **ANEXO A. Guía de instalación.**

### **Instalación de Java.**

Se pueden consultar las páginas 4 y 5 del documento adjunto Laboratorio\_ISCSD.

### **Instalación de MySql 5.0.27 win 32.**

Instalar configuración típica y configuración standard.

En el fichero de configuración de hibernate hibernate.cfg.xml ubicado en el directorio src del proyecto WebLibrosEJB viene la configuración de acceso a la base de datos “libreria” mediante el usuario Gustavo y la contraseña mydrdown00

En el directorio raíz del fichero de instalación se encuentra el archivo sql.txt con las sentencias SQL necesarias para crear la base de datos y unos datos de prueba.

### **Instalación de Jboss 4.0.5 GA**

Seguir las instrucciones de la página 8 ,11 y 25 del documento adjunto Laboratorio\_ISCSD.

Copiar conector jdbc (mysql-connector-java.5.0.4 adjunto) al jboss/server/default/lib

Copiar WebLibros.jar y WebLibros.war a server/deafault/deploy en jboss

El classpath debe contener lo siguiente (con los directorios adecuados claro está):

```
.;  
C:\Archivos de programa\Java\jre1.5.0_10\lib\ext\QTJava.zip;  
%JBOSS_HOME%\client\log4j.jar;  
%JBOSS_HOME%\client\jbossall-client.jar;  
%JAVA_HOME%\lib\tools.jar;  
C:\Java\mysql-connector-java\mysql-connector-java-5.0.4-bin.jar
```

Arrancar el servidor Jboss con archivo .bat run.bat en el directorio bin de jboss.

Se puede acceder a la aplicación web desde <http://localhost:8080/weblibros>.

### **Datos de prueba:**

En los datos de prueba están creados los siguientes usuarios:

**Usuario administrador.**

Usuario: admin

Password: admin00

**Usuario registrado.**

Usuario: gustavo

Password: gustavo00