

Práctica de diseño de aplicaciones web en entorno servidor (Curso 2019/20)

FP DUAL (DUW2)

Descripción

Se trata de la implementación y documentación de la funcionalidad básica típica de una aplicación web de comercio electrónico. Para ello, se provee un proyecto ecommerce con la siguiente estructura de paquetes:

```
ecommerce [ecommerce master]
├── Deployment Descriptor: ecommerce
├── JAX-WS Web Services
└── Java Resources
    ├── src/main/java
    │   ├── com.wirtz.ecommerce.config
    │   ├── com.wirtz.ecommerce.controllers
    │   │   ├── ProductController.java
    │   │   ├── UserProfileController.java
    │   │   ├── com.wirtz.ecommerce.model.cartline
    │   │   ├── com.wirtz.ecommerce.model.cartservice
    │   │   ├── com.wirtz.ecommerce.model.order
    │   │   ├── com.wirtz.ecommerce.model.orderline
    │   │   ├── com.wirtz.ecommerce.model.orderservice
    │   ├── com.wirtz.ecommerce.model.product
    │   │   ├── Product.java
    │   │   ├── ProductDao.java
    │   │   ├── ProductDaoHibernate.java
    │   ├── com.wirtz.ecommerce.model.productservice
    │   │   ├── ProductBlock.java
    │   │   ├── ProductService.java
    │   │   ├── ProductServiceImpl.java
    │   ├── com.wirtz.ecommerce.model.userprofile
    │   │   ├── LoginUserGroupValidator.java
    │   │   ├── RegisterUserGroupValidator.java
    │   │   ├── UserProfile.java
    │   │   ├── UserProfileDao.java
    │   │   ├── UserProfileDaoHibernate.java
    │   ├── com.wirtz.ecommerce.model.userservice
    │   ├── com.wirtz.ecommerce.model.util
    │   ├── com.wirtz.ecommerce.modelutil
    │   ├── com.wirtz.ecommerce.modelutil.dao
    │   └── com.wirtz.ecommerce.modelutil.exceptions
```

Funcionalidad

La tienda permitirá a los usuarios comprar productos. La información de cada producto incluye:

- nombre
- precio
- fecha de alta
- número de unidades de stock

Cada usuario puede buscar productos. Además podrá añadirlos a su carrito de la compra si ha iniciado sesión. Una vez en el carrito puede eliminarlos o modificar el número de unidades que desea adquirir. El carrito de la compra será persistente, es decir, si un usuario abandona la sesión sin comprar los productos que añadió al carrito, la siguiente vez que entre en la tienda, el carrito contendrá estos productos.

Cuando el usuario decide comprar los productos de su carrito, se generará un pedido, que se compondrá de varias líneas de pedido. Por cada pedido se guardará la fecha en la que se hizo la compra. Por cada línea de pedido se guardará el número de unidades pedidas del producto y el precio unitario.

Un usuario autenticado podrá ver el número total de productos de su carrito en todo momento.

Casos de uso

Para no alargar innecesariamente la práctica, el proyecto proporcionado implementa los siguientes casos de uso:

- Registro de usuarios
- Inicio y fin de sesión con autenticación basada en formulario. Si la autenticación es exitosa, se almacena en la sesión el identificador de usuario y su nombre.
- Búsqueda de productos: Cualquier usuario podrá buscar por palabras clave. Las palabras tienen que estar todas contenidas en el nombre del producto y en cualquier orden. Los productos que aparecen como resultado de una búsqueda se visualizan mostrando su nombre, precio y un enlace para añadirlo al carrito (para añadirlo al carrito es preciso estar autenticado).

Para la realización de la práctica se pide la implementación de los siguientes casos de uso:

1. Visualización de la información del producto desde los resultados de búsqueda de productos: El nombre del producto deberá estar asociado a un enlace que permitirá ver los detalles del producto.
2. Gestión del carrito de la compra: Un usuario autenticado debe poder:
 1. visualizar su carrito: La visualización del carrito incluye por cada producto: nombre (con enlace a ver sus detalles), precio unitario, número de unidades y precio total.
 2. eliminar productos del carrito
 3. modificar el número de unidades que desea adquirir de un producto
 4. añadir otros productos (mediante los enlaces presentes en los resultados de las búsquedas)

5. comprar todos los productos del carrito. Una vez realizada la compra, el carrito se vacía.
6. ver el número total de productos de su carrito en todo momento (si está autenticado)
3. Visualización de pedidos: Un usuario autenticado podrá visualizar todos los pedidos que ha hecho. Por cada pedido se mostrará la fecha y un enlace para ver detalles del pedido.

Implementación

Para la implementación se usarán las siguientes tecnologías:

- JSP 2.3.1
 - JSTL 1.2.1
 - Spring MVC 5.1.0.RELEASE
 - Hibernate 5.3.5.Final
 - SQL Server 2017
- Se utilizará, por tanto, el patrón de arquitectura MVC ayudados del framework Spring MVC.
 - Además usaremos el patrón DAO (Data Access Object) que se implementarán en función de los DAO genéricos proporcionados. En los paquetes *com.wirtz.ecommerce.modelutil.dao* y *com.wirtz.ecommerce.modelutil.exceptions* encontraréis los DAO genéricos que se han utilizado en los casos de uso ya implementados. Tenéis un ejemplo en ProductDAO:

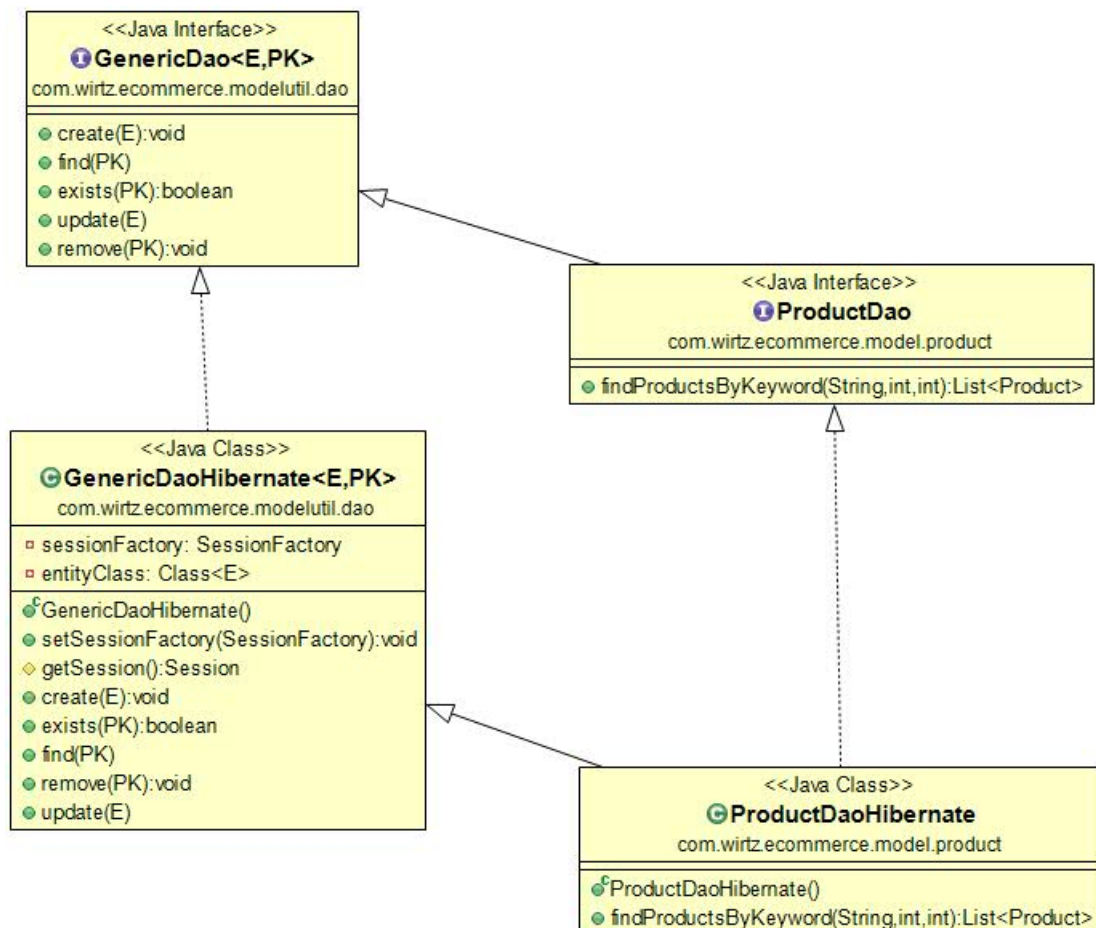


Figura 1: Implementación de ProductDao a partir de los Daos genéricos de *com.wirtz.ecommerce.modelutil.dao*

Para la implementación de la práctica se pide lo siguiente:

1. Creación de las clases persistentes E y su anotación para utilizar Hibernate y SQL Server 2017.
2. Configurar el acceso a la base de datos mediante un recurso jndi. Modifica los ficheros de tu contenedor de aplicaciones (Tomcat) para exponer el recurso con nombre **jdbc/Ecommerce**, la base de datos ha de llamarse **ecommerce**, crea un usuario **ecommerce** con contraseña **ecommerce**.
3. Por cada clase persistente E, creación de una interfaz EDao y su implementación EdaoHibernate a partir de los DAOS Genéricos proporcionados. Para métodos diferentes a los existentes en el GenericDao, se deberán crear métodos específicos usando HQL o [JPA 2 Criteria API](#). Tenéis un ejemplo HQL en ProductDaoHibernate: `List<Product> findProductsByKeyword(String keyword)`
4. Diseñar las interfaces de los servicios a utilizar y desarrollar sus implementaciones en función de los Daos creados.
5. Documentar los métodos de la implementación de CartService en la clase Java.
6. Definir y desarrollar los controladores para implementar los casos de uso propuestos
7. Definir las vistas JSP para implementar los casos de uso propuestos

Documentación

Se ha de redactar un documento con el siguiente nombre AWDS_Memoria_Práctica1_Nombre_Apellidos.pdf. El documento ha de contener las siguientes secciones:

1. Índice
2. Indica las modificaciones introducidas en los ficheros de configuración de Tomcat para publicar los recursos Jndi.
3. Un diagrama¹ de clases persistentes y sus relaciones
4. Un diagrama de Daos similar al proporcionado en 1 para una entidad que hayáis creado.
5. Un diagrama con todas las interfaces de Servicios de la aplicación.
6. Un diagrama que ilustre las clases que intervienen en el caso de uso “realizar pedido” así como sus relaciones. Podría ser un [diagrama de secuencia UML](#) o un diagrama más sencillo de forma similar al diagrama proporcionado en la implementación de Movies con Servlets y JSP . Ha de distinguirse claramente:
 1. qué componentes juegan el papel de Modelo, Vista y Controlador
 2. dónde situarías el dispatcher servlet de Spring
 3. no es necesario que muestre las implementaciones de los servicios y Daos, pero sí ha de ilustrarse las interfaces de los servicios y de los Daos.
7. Indica en qué consiste la inyección de dependencias o inversión de control característica de Spring y pon un ejemplo en el código donde se esté aprovechando esta característica.
8. Indica en qué consiste la programación orientada a aspectos y pon un ejemplo en el código donde se aproveche esta característica.
9. Problemas conocidos: Lista los errores que se conoce que tiene el código.

1 Para la realización de los diagramas podréis hacer uso de herramientas como el plugin [ObjectAid UML Explorer](#).

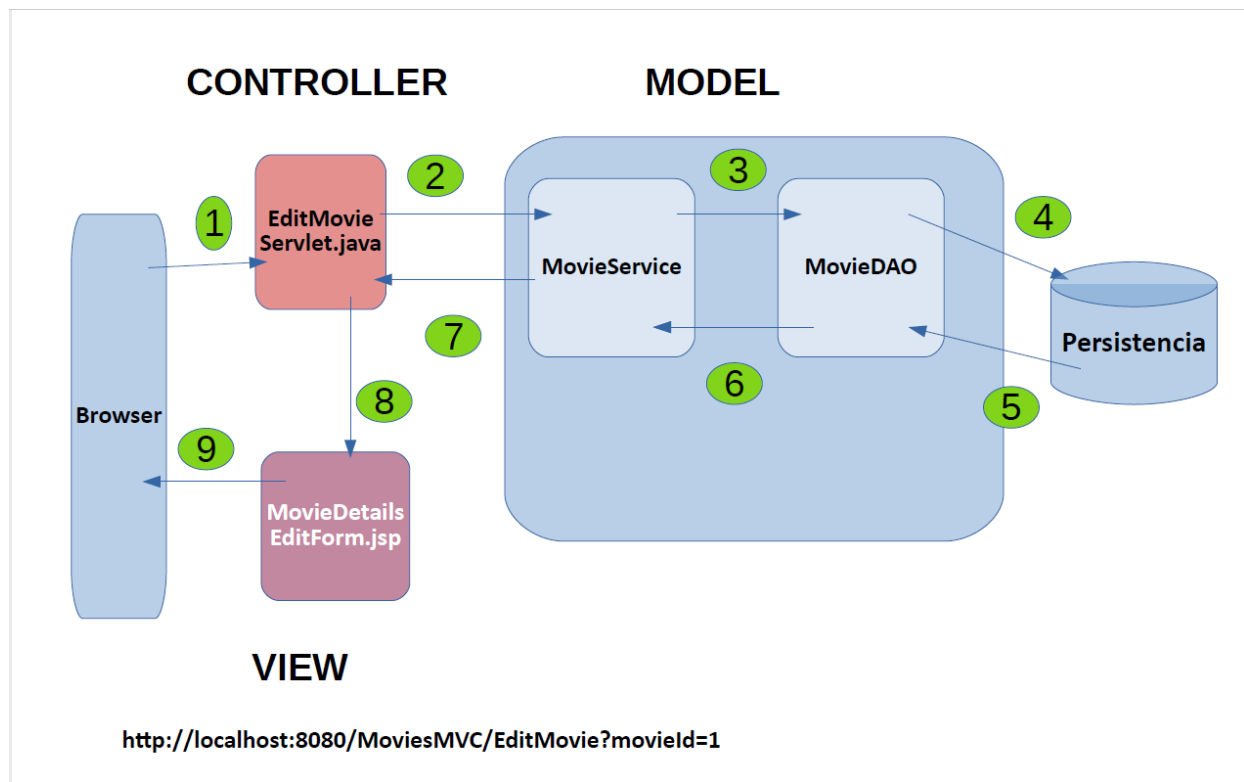


Figura 2: Diagrama proporcionado para la implementación de Movies con Servlets y JSP. Es sólo una referencia, no tiene por qué seguir este formato

Productos entregables

- 1 Un archivo AWDS_Práctica1_Nombre_Apellidos.zip que contenga:
 - 1.1 el proyecto ecommerce exportado en sistema de ficheros comprimido ecommerce_Nombre_Apellidos.zip
 - 1.2 El documento AWDS_Memoria_Práctica1_Nombre_Apellidos.pdf

Fecha de entrega:

Fecha límite de entrega será el día **11/12/2019 a las 20:00h**. Las defensas tendrán lugar el día 11 y 12 de diciembre durante las horas de clase.

Calificación

La calificación total de la práctica supondrá el 30% de la nota de la primera evaluación.

La calificación de la práctica se dividirá de la siguiente forma:

1. Implementación (60%)
2. Documentación (30%)
3. Defensa (10%)

Una práctica copiada será calificada con un 0.

A continuación se encuentran desglosados los pesos de cada una de las partes en el caso de un desempeño máximo:

IMPLEMENTACIÓN (60% del total de la práctica)		
Apartados de la práctica		Peso en caso de obtener el nivel máximo de desempeño
Aspectos funcionales		
Muestra la información del producto desde los resultados de búsqueda de productos: El nombre del producto deberá estar asociado a un enlace que permitirá ver los detalles del producto.		5,00 %
Para un usuario autenticado:		
Muestra su carrito: La visualización del carrito incluye por cada producto: nombre (con enlace a ver sus detalles), precio unitario, número de unidades y precio total.		8,00 %
Permite eliminar productos del carrito		8,00 %
Permite modificar el número de unidades que desea adquirir de un producto		8,00 %
Permite añadir otros productos (mediante los enlaces presentes en los resultados de las búsquedas)		10,00 %
Permite comprar todos los productos del carrito. Una vez realizada la compra, el carrito se vacía.		10,00 %
Muestra el número total de productos de su carrito en todo momento		5,00 %
Conserva el contenido del carrito tras cerrar sesión y volver a autenticarse.		2,00 %
Muestra el histórico de pedidos realizados. Por cada pedido muestra la fecha y un enlace para ver detalles del pedido		8,00 %
Aspectos técnicos		
Utiliza jndi para acceder a la base de datos		3,00 %
Utiliza los Daos Genéricos proporcionados para la implementación de los Daos de clases persistentes		5,00 %
Utiliza HQL o JPA Criteria para las consultas particulares de cada DAO que no están recogidas en los DAOs genéricos		5,00 %
Establece las dependencias en función de las interfaces (de Daos y Servicios)		3,00 %
Documenta los métodos de la implementación de CartService en la clase Java.		5,00 %
Desarrolla controladores usando las anotaciones pertinentes de SpringMVC		5,00 %
Crea las vistas JSPs utilizando librerías JSTL		5,00 %
Presenta el código fuente de forma ordenada		5,00 %
Total		100,00 %

DOCUMENTACIÓN (30% del total de la práctica)		
Apartados de la práctica		Peso en caso de obtener el nivel máximo de desempeño
Aspectos funcionales		
Incluye índice bien estructurado y actualizado		2,50 %
Indica las modificaciones introducidas en los ficheros de configuración de Tomcat para publicar los recursos Jndi.		2,50 %
Incluye un diagrama de clases persistentes y sus relaciones que se corresponde con la implementación		10,00 %
Incluye un diagrama de Daos similar al proporcionado en Figura 1 para una entidad que creada por el alumno que se corresponde con la implementación		10,00 %
Incluye un diagrama con todas las interfaces de Servicios de la aplicación.		10,00 %
Incluye un diagrama que ilustre las clases que intervienen en el caso de uso “realizar pedido” así como sus relaciones donde distinguirse claramente:		30,00 %
qué componentes juegan el papel de Modelo, Vista y Controlador	10,00 %	
la función del dispatcher servlet de Spring	10,00 %	
ilustra las interfaces de los servicios y de los Daos.	10,00 %	
Indica en qué consiste la inyección de dependencias o inversión de control característica de Spring y pone un ejemplo en el código donde se esté aprovechando esta característica.		10,00 %
Indica en qué consiste la programación orientada a aspectos y pone un ejemplo en el código donde se aproveche esta característica.		10,00 %
Lista los problemas conocidos que se conoce que tiene el código.		5,00 %
Presentación (consistencia, orden)		10,00 %
Consistencia estética	3,00 %	
Expresión cuidada	3,00 %	
Ortografía	4,00 %	
Total		100,00 %