

# Tienda online

Desarrollo de Aplicaciones Web Avanzadas

**Autores:**

Isabel Rodríguez Iglesias

Manuel Santiago Saborido Caamaño

Rosa Elena Veiga Otero

Eloy Rubiños Alonso



Tabla de contenido

1    Objetivo de la práctica ..... 1

2    Descripción del equipo..... 1

3    Requisitos funcionales ..... 1

4    Requisitos no nominales ..... 1

5    Requisitos de calidad del servicio ..... 2

6    Detalles técnicos de interés ..... 2

Anexo I – Documentos adjuntos..... 3



## 1 Objetivo de la práctica

Esta práctica consiste en ampliar las funcionalidades de una tienda online implementada durante prácticas anteriores. El resultado de esta práctica debe ser una tienda funcional con ciertos requisitos básicos mínimos y que cumpla ciertos criterios de escalabilidad.

## 2 Descripción del equipo

Los cuatro integrantes del equipo nos repartiremos el trabajo a partes iguales y trabajamos colaborativamente en la implementación de los distintos módulos de la aplicación.

## 3 Requisitos funcionales

Tenemos que extender la funcionalidad de los clientes y crear un tipo de usuario administrador e implementar funcionalidades para este.

En cuanto a los clientes, tenemos que permitir autenticación y registro; a mayores, tenemos que implementar filtros para la tienda y la funcionalidad completa de la tienda (compras con impacto en la BD, confirmaciones, factura, etc...). Por último de entre lo más reseñable, tenemos que implementar con JavaMail el envío de un correo de confirmación de la compra.

**Decidimos implementar las funcionalidades opcionales** (vista e introducción de comentarios y valoraciones).

En cuanto a los administradores, tenemos que implementar gestión básica de la tienda: introducir, modificar y eliminar productos, y modificar y eliminar usuarios.

## 4 Requisitos no nominales

**NN1:** Cuando no encontramos coincidencias entre el nombre de usuario introducido en el login y la BD, enviamos una página de error informando de la eventualidad.

**NN2:** Cuando todos los datos de un nuevo CD sean idénticos (menos el id) a los de uno ya existente, no lo insertaremos en la BD y lanzaremos una página de error. De todas formas gestionamos los productos de la BD a través de un campo id del que, debido a nuestra implementación, nunca puede haber duplicados, así que no llegaría a haber inconsistencias.

**NN3:** Implementamos el lanzamiento de una página de error durante el registro cuando los datos introducidos son incorrectos o se produce una excepción SQL al introducir el nuevo usuario en la BD, aunque es altamente improbable que se den estas situaciones ya que los campos de registro están protegidos por patrones de expresiones regulares.

## 5 Requisitos de calidad del servicio

**CS1:** El tipo de usuario es escalable ya que se usa una clase abstracta para definir los atributos y métodos básicos necesarios. Esta clase es extendida por los clientes y los administradores, y podría ser extendida por otros tipos de usuario.

**CS2:** Afrontamos la escalabilidad de la base de datos mediante el uso de índices para los productos y las ventas (y líneas de venta), que son las tablas más susceptibles de crecer mucho. Todas las operaciones que tienen que ver con estas tablas se basan en sus índices, que son números enteros, lo cual hace las consultas mucho más rápidas en ciertos casos.

**CS3:** Todos los JSP están “automatizados” en el sentido de que generan los contenidos para cualquier cantidad de elementos que queramos mostrar, usando tablas fácilmente ampliables para acomodar más datos. Cuando queramos ampliar funcionalidades nos resultará bastante sencillo gracias a esto.

**CS4:** Tenemos funcionalidades para manejar la inserción/modificación/borrado en BD en masa, así que sería fácil conectarnos a aplicaciones externas e importar sus datos.

Por otro lado, el acceso a datos está separado de los objetos de transferencia (DAO + VO), por lo que también sería posible que la tienda trabajase con varias fuentes de datos simultáneamente. La tienda verá siempre el mismo tipo de objeto y siempre funcionará con ellos.

**CS5:** En las transacciones más importantes y con un mínimo volumen de uso se usan Prepared Statements, y cuando se van a producir cambios en bloque en la BD se usan batches.

## 6 Detalles técnicos de interés

El script de creación de la BD que se proporciona contiene los siguientes usuarios con el objetivo de facilitar el uso y la prueba de la aplicación:

Usuario: cliente      Contraseña: cliente

Usuario: admin      Contraseña: admin

Como resulta evidente, ‘cliente’ es un usuario de tipo cliente con acceso a la tienda, y ‘admin’ es un usuario de tipo administrador con acceso a la administración de la tienda.

A mayores incluye otra serie de usuarios.

Contiene los productos que se nos proporcionaron para la primera versión de la tienda.

No tiene ventas ni líneas de venta, pero sí comentarios y valoraciones para probar cómo se ven en los detalles de los productos.

Los DAO funcionan con la url “jdbc:mysql://localhost:3306/bd\_dawa”, y obtienen el usuario y contraseña de acceso a la BD durante su instanciación, como parámetros “USUARIO” y “PASS” declarados globales al inicio de cada uno de los cuatro servlets.

Los datos con que lo enviamos por defecto son:

Usuario: daw      Contraseña: etse

## Anexo I – Documentos adjuntos

Nombre	Descripción
<b>Modelos.mdj</b>	Documento StarUML 2 que contiene los siguientes diagramas: diagrama ER, diagrama de clases, diagrama de arquitectura y diagramas de secuencia.
<b>Diagramas</b>	Carpeta que contiene PNGs de todos los diagramas, por comodidad.
<b>create_schema.sql</b>	Script de creación de la base de datos MySQL.
<b>insert_initial_data.sql</b>	Script de inserción de datos de prueba.
<b>proyecto_Eloy_RubiñosAlonso.war</b>	Aplicación web desarrollada.