

Prueba - Productos de Limpieza

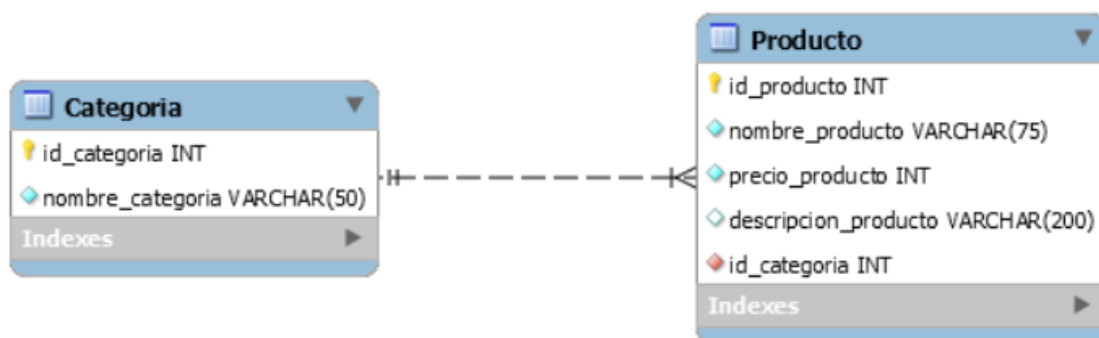
- Para realizar esta prueba debes haber estudiado previamente todo el material disponible en el LMS correspondiente al módulo.
- Una vez terminada la prueba, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el .zip en el LMS.
- Puntaje total: 10 puntos
- Desarrollo prueba: Individual

Habilidades a evaluar

- Crear y configurar un proyecto web dinámico utilizando el lenguaje Java, definiendo la estructura del proyecto y agregando las librerías a utilizar.
- Diseñar e implementar el mecanismo de procesamiento de solicitudes HTTP de una aplicación web dinámica.
- Generar las vistas necesarias para el correcto funcionamiento de una aplicación web dinámica.
- Configurar la conexión a base de datos de una aplicación web dinámica, y definir la estructura de la capa de persistencia de la aplicación.
- Implementar la capa de persistencia de datos, de tal forma que la aplicación pueda realizar operaciones en una base de datos e interactuar con datos relacionados.

Descripción

Una empresa dedicada a la venta de artículos de aseo y cuidado personal está buscando implementar un sistema web que le ayude a administrar su negocio. Para eso, han solicitado que en una primera etapa, la aplicación sea capaz de registrar y mostrar los productos que ellos ofrecen, los cuales están asociados a una categoría de productos. El equipo encargado de la estructura de la base de datos ha construido el siguiente esquema, que cumple con los requerimientos de esta primera etapa:



También han construido los comandos para crear esta base de datos, que viene con algunos productos y categorías ya cargados. El script está disponible en el archivo llamado “Apoyo Prueba - Productos de Limpieza”.

La aplicación debe estar desarrollada usando el lenguaje Java, y debe ser capaz de registrar, mostrar, modificar y eliminar los productos. Para eso, se debe crear un proyecto web dinámico con nombre a libre elección, donde el paquete raíz sea “com.desafiolatam”, y dentro de él se encuentren los demás paquetes del proyecto, siguiendo el patrón de diseño MVC.

Además, se debe generar las vistas y controladores que sean necesarios para poder realizar las operaciones descritas anteriormente, junto con una página principal que permita acceder a todas estas funcionalidades. Esta página principal debe ser la primera página en mostrarse al acceder a la aplicación web.

Las vistas deben estar conectadas por medio de enlaces que permitan la navegación entre ellas. Esto implica que siempre debe existir al menos una ruta que permita desplazarse entre todas y cada una de las páginas que conforman el sitio.

Al mostrar los datos del producto, se debe poder visualizar el id de producto, el nombre, la descripción y el precio. La aplicación también debe ser capaz de mostrar el nombre de la

categoría a la que pertenece cada producto, y de registrar a qué categoría pertenece cada producto nuevo que se agrega. No es necesario que la aplicación cree, modifique o elimine esas categorías.

También, se debe construir el modelo que refleje la estructura de la base de datos, y que permita realizar las operaciones descritas e interactuar con las relaciones entre el producto y su categoría asociada. Los nuevos productos deben registrarse con un identificador autoincremental, que genere el id según el último registro ingresado. Al modificar registros, no se debe poder modificar el id del producto.

Requerimientos

1. Crear un proyecto web dinámico en Java, estructurando el proyecto en paquetes según el patrón de diseño MVC, y configurando las librerías que utilizará el proyecto (JSTL y driver de conexión a base de datos).
(2 Puntos)
2. Crear la estructura de la capa de persistencia de datos (modelo), siguiendo el patrón DAO y utilizando una conexión de tipo Singleton para acceder a la base de datos.
(2 Puntos)
3. Implementar la capa de persistencia de datos, de tal forma que la aplicación sea capaz de crear, leer, modificar y eliminar registros, manteniendo y mostrando las relaciones entre los distintos datos.
(2 Puntos)
4. Diseñar e implementar el mecanismo de procesamiento de solicitudes HTTP, usando Servlets que tengan responsabilidad única, y no incluyan lógica de negocio en ellos.
(2 Puntos)
5. Generar las vistas a utilizar en la aplicación, usando JSP y JSTL para desplegar datos de forma dinámica.
(2 Puntos)

Requerimientos opcionales

1. Utilizar Bootstrap para definir el layout de las vistas, y estilizar los diferentes componentes de la interfaz web de la aplicación (botones, enlaces, menús, tablas, títulos, etc.).
2. Agregar una clase adicional a la aplicación siguiendo el patrón Facade, para conectar las capas de acceso a datos y el controlador. Esta clase debe permitir el acceso a todas las operaciones que puede realizar la capa de acceso a datos, concentrándose en un solo punto de acceso.
3. Implementar un mecanismo de autenticación, que restrinja el acceso a la página de bienvenida, solicitando nombre de usuario y contraseña. No es necesario que tenga mecanismo de cierre de sesión, que restrinja el acceso a otras páginas del sitio, o que extraiga la información del usuario o contraseña desde una base de datos.
4. Utilizar cookies para persistir un dato en el navegador del cliente, y mostrar ese dato almacenado en alguna de las vistas.