**Documento Técnico: Arquitectura de la Aplicación**

**Introducción**

Este documento técnico describe la arquitectura de la aplicación desarrollada utilizando Java EE con las tecnologías JSF, PrimeFaces, Hibernate, y JPA. La aplicación es un sistema que permite realizar operaciones CRUD Crear, Leer( las demás operaciones están en el código pero por tiempo no fueron implementadas) sobre una entidad llamada "Producto". A continuación, se explican las principales características de la arquitectura y las decisiones de diseño tomadas durante el desarrollo.

**Arquitectura General**

La arquitectura de la aplicación sigue un patrón de diseño típico de aplicaciones web Java EE. Está basada en el Modelo-Vista-Controlador (MVC), donde:

**Modelo:** Representado por las entidades JPA, en este caso, la entidad "Producto". Estas clases mapean objetos Java a tablas de la base de datos y definen la lógica de negocio.

**Vista:** Representada por las páginas JSF y PrimeFaces. Estas páginas proporcionan la interfaz de usuario con la que los usuarios interactúan para realizar operaciones CRUD en los productos.

**Controlador:** Representado por las clases de controlador JSF. Estas clases gestionan las solicitudes de los usuarios, interactúan con el modelo (las entidades) y actualizan la vista según sea necesario.

**Componentes Principales**

A continuación, se describen los componentes principales de la aplicación:

**Entidades JPA:** La entidad principal es "Producto," que está mapeada a una tabla en la base de datos. Esta entidad almacena la información sobre los productos, como nombre, descripción y precio.

**Controladores JSF:** Los controladores JSF, como "ProductoController," gestionan las operaciones CRUD para los productos. Estas clases interactúan con el EJB ProductoFacadeLocal, que proporciona métodos para acceder a la base de datos.

**EJB (Enterprise JavaBeans):** La clase ProductoFacade es un EJB que encapsula la lógica de acceso a la base de datos utilizando JPA. Proporciona métodos para crear, leer, actualizar y eliminar productos, así como una función para probar la conexión a la base de datos.

**Páginas JSF y PrimeFaces:** Las páginas JSF (como "index.xhtml" y "productos.xhtml") junto con los componentes de PrimeFaces, proporcionan la interfaz de usuario para interactuar con la aplicación. Se utilizan para mostrar la lista de productos y permitir la creación de nuevos productos.

**Base de Datos:** La aplicación utiliza una base de datos MySQL para almacenar los datos de los productos. La configuración de la conexión a la base de datos se realiza en el archivo "persistence.xml."

**Decisiones de Diseño**

Tecnologías Utilizadas: Se seleccionaron tecnologías estándar de Java EE, como JSF, PrimeFaces, Hibernate y JPA, para desarrollar una aplicación web robusta y escalable.

Patrón MVC: Se siguió el patrón Modelo-Vista-Controlador para separar claramente la lógica de la aplicación, la interfaz de usuario y el control de las solicitudes del usuario.

Uso de EJB: Se utilizó un EJB (ProductoFacade) para encapsular la lógica de acceso a la base de datos. Esto permite una gestión eficiente de las transacciones y el control de la base de datos.

Seguridad: La aplicación no implementa características de seguridad en este documento.

Mensajes de Retroalimentación: Se implementaron mensajes de retroalimentación para informar al usuario sobre el resultado de las operaciones, como la creación exitosa de un producto o errores en la conexión.

**Instrucciones de Instalación y Ejecución de la Aplicación**

**Requisitos Previos**

Antes de comenzar con la instalación y ejecución de la aplicación, asegúrese de tener instalado lo siguiente:

**Java Development Kit (JDK):** Asegúrese de tener instalado Java SE Development Kit 8 (o una versión compatible).

**Servidor de Aplicaciones:** Necesitará un servidor de aplicaciones Java EE compatible. En este documento, se asume que está utilizando **WildFly**. Puede descargarlo desde WildFly Downloads.

**Gestor de Bases de Datos:** Asegúrese de tener instalada una base de datos MySQL. Puede descargarla desde **MySQL** Downloads.

**Herramienta de Construcción:** Se utiliza Maven como herramienta de construcción para administrar las dependencias del proyecto. Asegúrese de tener Maven instalado.

**Pasos de Instalación y Ejecución**

A continuación, se detallan los pasos para instalar y ejecutar la aplicación:

1. Clonar el Repositorio

Clone el repositorio de la aplicación desde GitHub:

**2. Importar el Proyecto**

Abra su IDE de desarrollo preferido (por ejemplo, NetBeans, Eclipse o IntelliJ IDEA) e importe el proyecto clonado como un proyecto Maven existente.

**3. Configurar la Base de Datos**

Cree una base de datos MySQL con el nombre "pruebatecnica" (o el nombre que desee) y creamos la tabla: CREATE TABLE productos (

id BIGINT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255) NOT NULL,

descripcion TEXT,

precio DECIMAL(10, 2) NOT NULL

);.

Edite el archivo src/main/resources/META-INF/persistence.xml para configurar la conexión a la base de datos. Asegúrese de actualizar la URL, el nombre de usuario y la contraseña según su configuración de MySQL.

**4. Configurar el Servidor de Aplicaciones (WildFly)**

Inicie su servidor WildFly.

Cree un DataSource para la base de datos en WildFly. Puede hacerlo utilizando la consola de administración de WildFly. Aquí hay un ejemplo de configuración:

JNDI Name: java:/yourDataSource

Connection URL: La URL de conexión a su base de datos MySQL.

Username y Password: Sus credenciales de MySQL.

**5. Compilar el Proyecto**

Utilice Maven para compilar el proyecto. Abra una terminal en el directorio raíz del proyecto y ejecute:

mvn clean install

**6. Implementar la Aplicación**

Después de compilar con éxito, despliegue la aplicación en su servidor WildFly. Esto suele hacerse copiando el archivo WAR generado (PruebaTecnicaJavaWeb-1.0-SNAPSHOT.war) en el directorio de despliegue de WildFly.

**7. Acceder a la Aplicación**

Abra un navegador web y vaya a la siguiente dirección: http://localhost:8080/PruebaTecnicaJavaWeb-1.0-SNAPSHOT/.

Debería ver la página principal de la aplicación con un botón para "ver el listado de productos".

**8. Utilizar la Aplicación**

Haga clic en el botón "ver listado de productos" para acceder a la página de productos.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Puede agregar nuevos productos ingresando los detalles en el formulario y haciendo clic en "Agregar".

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Y al agregar el producto se muestra un mensaje que se han agregado con éxito y se refresca la tabla para que se vea el producto agregado  
Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente con confianza media

La librería la importamos en Dependecies y usamos sus métodos en ProductoFacade

Texto

Descripción generada automáticamente