

Universidad Tecnológica Nacional

Tecnicatura Universitaria en Programación a Distancia



Título del trabajo: **“Trabajo Final Integrador (TFI) - Programación 2”**

Alumnos:

Eric Suárez Dubs, ericgodzilladubs@gmail.com - Comisión 13

Gonzalo Vega, gonzaarg03@gmail.com - Comisión 11

Federico Iacono, iaconofede@gmail.com - Comisión 6

Mateo Serafini, matuserafini@gmail.com - Comisión 5

Materia: **Programación 2**

Profesor coordinador: Carlos Martinez

Fecha de entrega: 20 de noviembre de 2025

Video: [Link YouTube](#)

Índice

1. Introducción. Propósito del proyecto y funcionalidad general.

2. Diseño del Programa y Dominio

2.a. Integrantes y Roles. Detalle de los miembros del equipo y asignación de tareas.

2.b. Elección del Dominio y Justificación. Por qué Producto → CódigoBarras.

2.c. Diagrama UML y Decisiones Clave. Referencia al Diagrama UML y explicación de la relación 1-a-1 (FK en productos).

3. Arquitectura y Funcionalidad

3.a. Arquitectura por Capas y Responsabilidades. Explicación de las 5 capas (entities, dao, service, main, utils).

3.b. Persistencia Transaccional y Atomicidad. Detalle del commit/rollback en la capa Service y la operación compuesta de creación.

3.c. Validaciones y Robustez del Sistema. Reglas de negocio (CHECK, unicidad), InputValidator, ManejadorExcepciones.

4. Pruebas y Conclusión

4.a. Pruebas Realizadas (Evidencia CRUD). Capturas de pantalla del menú (CRUD, errores de entrada).

4.b. Conclusiones y Mejoras Futuras. Aprendizaje y líneas de acción futuras (ej. migrar a ORM/API REST).

5. Anexo

5.a. Fuentes y Herramientas. Tabla de software y evidencia de IA.

1. Introducción: Propósito del Proyecto y Funcionalidad General

El presente Trabajo Final Integrador (TFI) tiene como propósito fundamental **integrar los conocimientos de arquitectura de software y persistencia de datos** mediante el desarrollo de una aplicación modular en Java.

El objetivo es modelar y gestionar de manera eficiente la relación crítica **Producto → Código de Barras** a través de una arquitectura por capas.

Funcionalidad General del Programa

La aplicación resultante es un sistema de inventario que provee todas las funcionalidades CRUD (Crear, Leer, Actualizar, Eliminar) para ambas entidades. Sus capacidades principales incluyen:

- **Persistencia Segura:** Conexión a la base de datos MySQL mediante **JDBC**, utilizando **PreparedStatement** para garantizar la seguridad contra inyección SQL.
- **Gestión Transaccional Atómica:** El sistema garantiza la **integridad y consistencia** de los datos. Las operaciones compuestas, como la creación de un Producto y un Código de Barras nuevo de forma simultánea, son manejadas dentro de una única transacción (commit/rollback).
- **Manejo de la Relación 1-a-1:** Permite asignar, reasignar o anular la vinculación de un código de barras a un producto, haciendo cumplir la restricción **UNIQUE** para la unicidad del código.
- **Robustez Operativa:** El programa ha sido diseñado con una capa de utilidades (com.utils) que previene fallos de aplicación por entradas inválidas del usuario (ej. texto en campos numéricos), manteniendo la estabilidad operativa del sistema.

Este proyecto sirve como una demostración de la implementación de una arquitectura limpia (DAO/Service) que aísla la lógica de negocio de la capa de persistencia y la presentación.

2. Diseño del Programa

El diseño de la aplicación se centró en dos pilares fundamentales: el cumplimiento de los requisitos de la consigna (como la relación unidireccional) y la implementación de una arquitectura robusta por capas que garantizara la separación de responsabilidades, la mantenibilidad y la escalabilidad.

Decisiones Clave de Diseño: Implementación de la Relación 1-a-1

La consigna del TFI exigía una **relación 1-a-1 unidireccional** donde la clase "A" (Producto) referencia a la clase "B" (CodigoBarras), pero la clase "B" no debe tener conocimiento de "A".

La relación se implementó de la siguiente manera:

- **En Java (Capa de Entidades):**
 - La clase Producto.java contiene el atributo private CodigoBarras codigoBarras;.
 - La clase CodigoBarras.java **no** contiene ningún atributo de tipo Producto. Cumpliendo con el requisito de unidireccionalidad $A \rightarrow B$.
- **En la Base de Datos (SQL):**
 - La clave foránea (FOREIGN KEY) se colocó en la tabla productos.
 - Se creó la columna codigos_barras_id BIGINT en la tabla productos, la cual referencia a codigos_barras(id).
 - Para forzar la cardinalidad **1-a-1** (y no 1-a-N), se añadió una restricción **UNIQUE** a esta columna codigos_barras_id, impidiendo que dos productos diferentes puedan apuntar al mismo código de barras.

Utilización de Clave Foránea (FK) en productos

Esta fue una decisión de diseño crucial basada en dos razones:

- **Coherencia con la Unidireccionalidad:** Poner la FK en la tabla productos modela perfectamente la relación unidireccional. La fila del producto "conoce" o "apunta" al código de barras que le pertenece. Si la FK estuviera en codigos_barras, la relación sería $B \rightarrow A$, lo cual violaría la consigna.
- **Flexibilidad Lógica:** Este diseño permite que un CodigoBarras exista en la base de datos de forma independiente, sin estar asignado a un producto. Esto es lógicamente correcto (se pueden pre-cargar códigos en el sistema). Además, al permitir que la columna

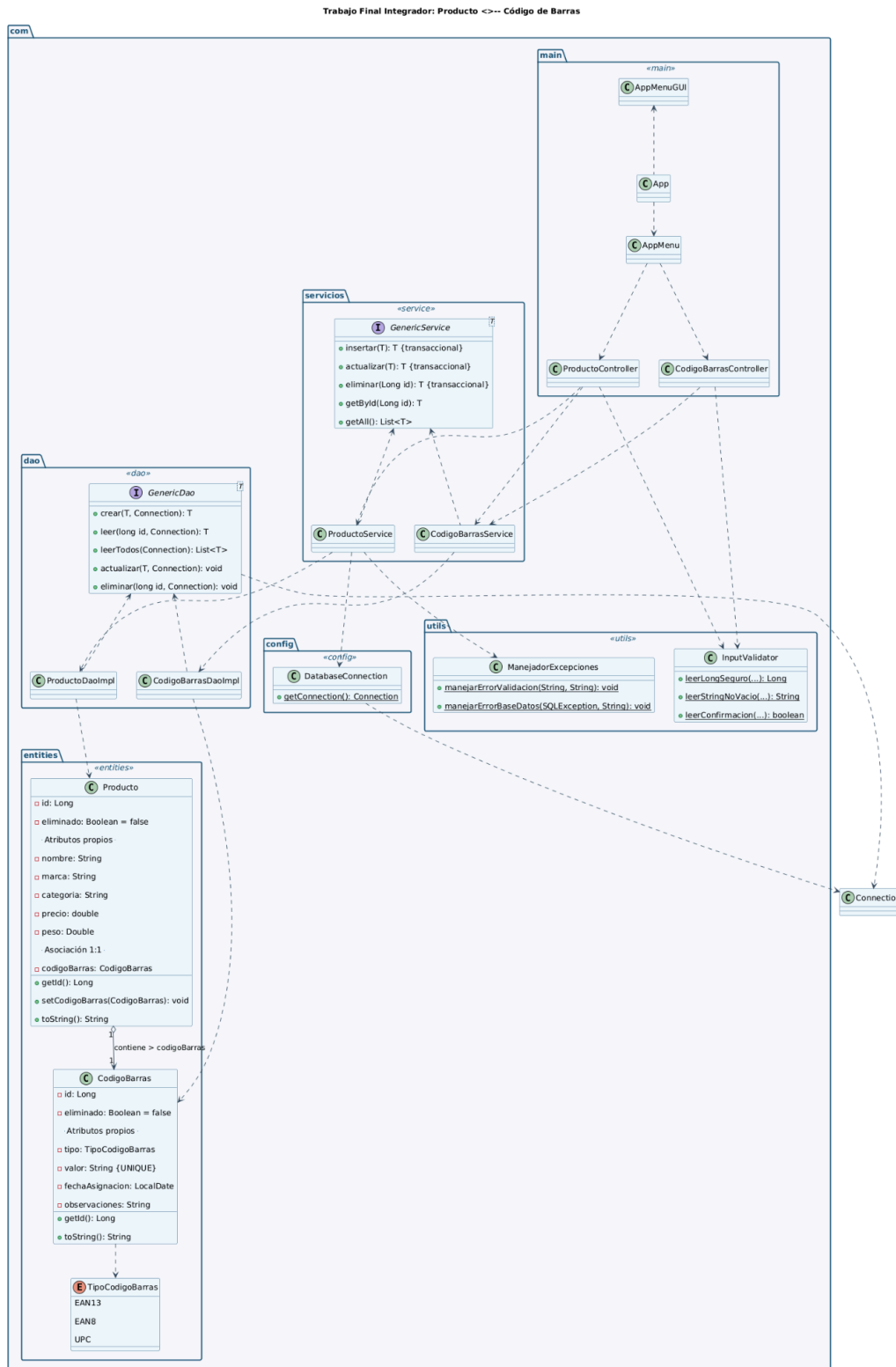
codigos_barras_id sea **nulable** (NULL), el sistema puede crear un Producto sin un código de barras asignado, permitiendo que se asigne posteriormente.

Integrantes y Roles

La estructura del equipo se definió para abarcar las distintas capas de la aplicación y coordinar los trabajos con la materia Bases de Datos I.

| Nombre | Rol Principal | Foco de la Implementación |
|-------------------------|--|--|
| Federico Iacono | Arquitecto de Persistencia / Estructura de capas | Diseño de entidades, conexión con Base de Datos, DAOs Genericos, DAO productos, DAO código de barras. Parte de la documentación técnica. |
| Gonzalo Vega | Ingeniero de Backend y Transacciones | Desarrollo de la Capa de Servicio, Manejo centralizado de errores, Interfaz gráfica y lógica de Atomicidad en servicios. Parte de la documentación técnica. |
| Eric Suárez Dubs | Desarrollador de Interfaces y Robustez | Implementación de la Capa de Utilidades (com.utils) y controladores de entrada/salida. Merge al final a rama main. Parte de la documentación técnica. |
| Mateo Serafini | Debugging, Arquitectura BBDD, UML y Documentación Técnica. | Diseño de BBDD. Diseño UML. Testing y Debugging. Parte de la documentación técnica. |

2. c. Diagrama UML

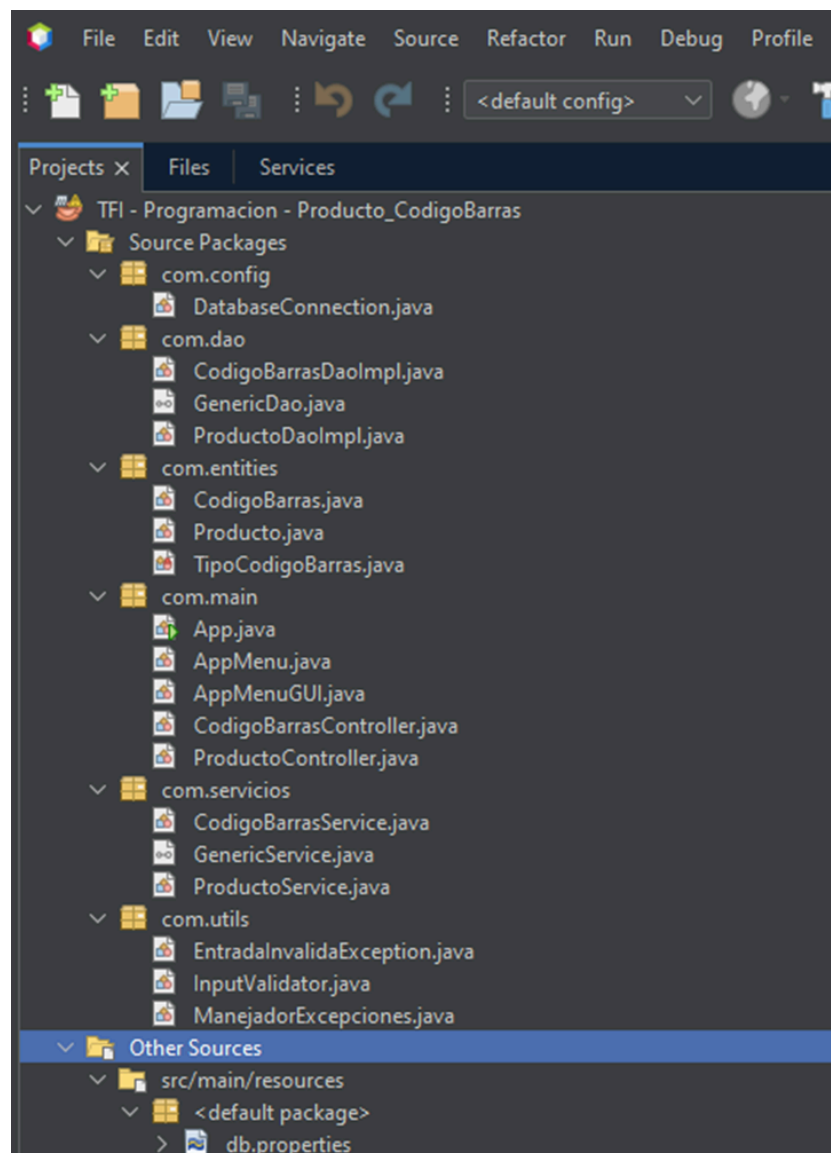


Para ver esto en mejor calidad, busque la imagen en el repositorio.(uml.png)

3. Arquitectura y Funcionalidad

Arquitectura por Capas

El proyecto está implementado siguiendo una arquitectura estricta de 5 capas, donde cada paquete (com.*) tiene una responsabilidad única y bien definida. Esto asegura el desacoplamiento: los cambios en una capa (como cambiar la base de datos en el DAO) no afectan a las otras (como la lógica de negocio en el Service).



Las responsabilidades de cada paquete son:

- **com.entities (Capa de Modelo):**
 - Representa los objetos de dominio del negocio (el "Qué").
 - Contiene solo POJOs (Plain Old Java Objects): Producto, CodigoBarras y TipoCodigoBarras.
 - No contiene lógica de negocio, validaciones ni acceso a datos.
- **com.dao (Capa de Acceso a Datos - DAO):**
 - Es la única capa que "sabe" hablar SQL y se comunica con la base de datos.
 - Su única responsabilidad es la persistencia: ejecutar consultas SELECT, INSERT, UPDATE mediante PreparedStatement.
 - Implementa la interfaz GenericDao y **no maneja transacciones**; recibe la Connection abierta desde la capa de servicio.
- **com.servicios (Capa de Lógica de Negocio - Service):**
 - Es el **cerebro** de la aplicación (el "Cómo").
 - Orquesta las operaciones llamando a uno o más métodos del DAO.
 - **Maneja las Transacciones:** Es la capa responsable de la atomicidad (setAutoCommit(false), commit(), rollback()), garantizando la integridad de operaciones compuestas (como crear un producto y su nuevo código de barras atómicamente).
 - Contiene todas las **validaciones y reglas de negocio** (ej. esValorDuplicado, validarFormatoSegunTipo).
- **com.main (Capa de Presentación y Control):**
 - Es la cara visible para el usuario (la "Vista" y el "Controlador").
 - App.java: Es el punto de entrada que permite elegir la interfaz (consola o GUI).
 - AppMenu.java (Vista): Muestra el menú de consola, captura la opción y delega la acción.
 - AppMenuGUI.java (Vista): Implementación de una interfaz gráfica Swing que consume los mismos servicios.
 - ProductoController.java y CodigoBarrasController.java (Controladores): Orquestan el flujo de la interfaz. Toman los datos de la vista (validados por InputValidator), los pasan a la capa Service y devuelven la respuesta al usuario.
- **com.utils (Capa de Utilidades):**
 - Paquete transversal que da soporte a las otras capas para mantenerlas limpias.

- `InputValidator.java`: Provee métodos robustos para leer la entrada del usuario en la consola, manejando errores (`NumberFormatException`) y reintentos, evitando que la aplicación falle.
- `ManejadorExcepciones.java`: Centraliza el manejo de errores, traduciendo códigos de error de SQL (1062, 1452) en mensajes amigables.
- `EntradaInvalidaException.java`: Excepción personalizada para un control de flujo de errores de usuario más limpio.

Persistencia, Validaciones y Robustez del Sistema

El programa garantiza la integridad de los datos y la estabilidad operativa a través de un diseño defensivo implementado en todas sus capas:

1. A Nivel de Base de Datos (SQL)

- **Integridad de Entidad:** Uso de PRIMARY KEY en todas las tablas para garantizar que cada registro sea único e identificable.
- **Integridad Referencial:** Implementación de FOREIGN KEY (`productos.codigos_barras_id`) que asegura que un producto solo pueda referenciar a un código de barras que realmente existe.
- **Restricciones de Unicidad (Regla de Negocio):**
 - Uso de UNIQUE en `codigos_barras.valor` para impedir códigos de barras duplicados.
 - Uso de UNIQUE en `productos.codigos_barras_id` para forzar la relación 1 a 1.
- **Restricciones de Dominio (CHECK):** Uso de CHECK (`precio > 0`, `categoria IN (...)`, etc.) para asegurar que solo datos lógicamente válidos puedan ser almacenados.

2. A Nivel de Persistencia (Capa DAO)

- **Prevención de Inyección SQL:** Uso exclusivo de `PreparedStatement` en todas las consultas SQL, parametrizando las entradas del usuario para neutralizar ataques.
- **Manejo de Relaciones Opcionales:** Uso de LEFT JOIN (en `leer/leerTodos`) y `setNull` (en `crear/actualizar`) en `ProductoDaoImpl` para manejar correctamente productos que no tienen un código de barras asignado.
- **Preservación de Datos:** Implementación de baja lógica (`UPDATE ... SET eliminado = true`) en lugar de DELETE físico.

3. A Nivel de Lógica de Negocio (Capa Service)

- **Transacciones Atómicas (ACID):** Implementación rigurosa de `setAutoCommit(false)`, `commit()` y `rollback()` en todos los métodos de servicio que modifican datos.
- **Orquestación de Operaciones Compuestas:** El método `ProductoService.insertar` maneja atómicamente la lógica compuesta de crear un `CodigoBarras` y un `Producto` en una sola transacción.
- **Validación de Reglas de Negocio (Previas):**
 - **Formato:** `validarFormatoSegunTipo` (ej. EAN-13 de 13 dígitos) y `validarCamposObligatorios`.
 - **Duplicidad:** `esValorDuplicado` (para `CodigoBarras.valor`) y `esCodigoYaAsignado` (para la relación 1-a-1), que consultan la base de datos antes de intentar un commit.

4. A Nivel de Presentación y Utilidades (Capas Main/Utils)

- **Robustez en la Captura de Datos:** La clase `InputValidator` previene que entradas inválidas (texto por número, valores vacíos, números negativos) lleguen a las capas inferiores.
- **Sistema de Reintentos:** El `InputValidator` ofrece múltiples intentos al usuario, mejorando la experiencia sin sacrificar la robustez.
- **Manejo de Fallos de Usuario:** El programa no se detiene si el usuario agota sus intentos; es capturado por los `Controllers` y `AppMenu`, y la aplicación continúa funcionando.
- **Manejo de Excepciones Centralizado:** La clase `ManejadorExcepciones` traduce códigos de error de SQL (1062 - Duplicado, 1452 - FK Rota) en mensajes claros, facilitando la depuración y la comprensión del error.

4. Pruebas y conclusiones

Inicio de aplicación.

```
????????????????????????????????????????????????????????????
?  SISTEMA DE GESTI?N DE PRODUCTOS      ?
????????????????????????????????????????????????????????????

Seleccione modo de ejecuci?n:
  1. Modo Consola
  2. Modo Gr?fico

Opci?n:
```

Aplicación por consola.

```
Aplicaci?n iniciada desde AppLauncher.

=== MEN? PRINCIPAL ===
1. Crear nuevo producto
2. Editar producto existente
3. Crear nuevo c?digo de barras
4. Editar c?digo de barras existente
5. Buscar producto
6. Buscar c?digo de barras
7. Asignar c?digo de barras a producto
8. Eliminar producto
9. Eliminar c?digo de barras
10. Listar productos
11. Listar c?digos de barras
12. Buscar por categoria de producto
13. Buscar por tipo de c?digos de barras
0. Salir

Seleccione una opci?n:
```

Creación de producto.

```
--Creación de nuevo producto--
Por favor, ingrese los datos del producto:
Nombre:
Mouse
Marca:
Logitech
Categoría:
Informatica
Precio:
100
¿Desea ingresar el peso del producto?
(S/N):
n
¿Desea asignar un código de barras al producto? (S/N):
s
Opciones:
1) Seleccionar código de barras existente por ID
2) Crear nuevo código de barras
Seleccione opción (1-2):
2
--Creación rápida de código de barras--
Observaciones (opcional):
ninguna
Seleccione el tipo de código de barras:
A) EAN13
B) EAN8
C) UPC
Ingrese su opción (A, B, C, etc.):
b
Valor (8 dígitos):
87654321
Código de barras creado: CodigoBarras{id=10015, eliminado=false, tipo=EAN8, valor='876
54321', fechaAsignacion=2025-11-20, observaciones='ninguna'}
Se creó correctamente su producto: Producto{id=10002, eliminado=false, nombre='Mouse',
marca='Logitech', categoria='INFORMATICA', precio=100.0, peso=null, codigoBarrasId=100
15}
```

Edición de producto.

```
Seleccione una opción:
2
--Edición de producto--
Ingrese el ID del producto a editar:
10002
[Producto encontrado: Producto{id=10002, eliminado=false, nombre='Mouse', marca='Logitech', categoria
='INFORMATICA', precio=100.0, peso=0.0, codigoBarrasId=10015}
Ingrese los nuevos datos del producto (deje en blanco para mantener el valor actual):
Nuevo nombre:
(Dejar vacío para mantener: Mouse)

Nueva marca:
(Dejar vacío para mantener: Logitech)

Nueva categoría:
(Dejar vacío para mantener: INFORMATICA)

Nuevo precio:
(Dejar vacío para mantener: 100.0)
150
Nuevo peso:
(Dejar vacío para mantener: 0.0)

¿Desea modificar el código de barras? (S/N):
n
[Producto actualizado correctamente: Producto{id=10002, eliminado=false, nombre='Mouse', marca='Logi
tech', categoria='INFORMATICA', precio=150.0, peso=0.0, codigoBarrasId=10015}
```

Lectura de producto.

```
Seleccione una opción:
5
--Búsqueda de producto--
Ingrese el ID del producto a buscar:
10002
[Producto encontrado: Producto{id=10002, eliminado=false, nombre='Mouse', marca='Logitech', categori
a='INFORMATICA', precio=150.0, peso=0.0, codigoBarrasId=10015}
```

Borrado de producto (Lógica).

```
--Eliminación de producto--
Ingrese el ID del producto a eliminar:
10002
[Producto a eliminar: Producto{id=10002, eliminado=false, nombre='Mouse', marca='Logitech', categoria
='INFORMATICA', precio=150.0, peso=0.0, codigoBarrasId=10015}
¿Está seguro de que desea eliminar este producto? (S/N):
s
[Producto eliminado correctamente: Producto{id=10002, eliminado=true, nombre='Mouse', marca='Logitec
h', categoria='INFORMATICA', precio=150.0, peso=0.0, codigoBarrasId=10015}
```

Listado por Categoría de productos.

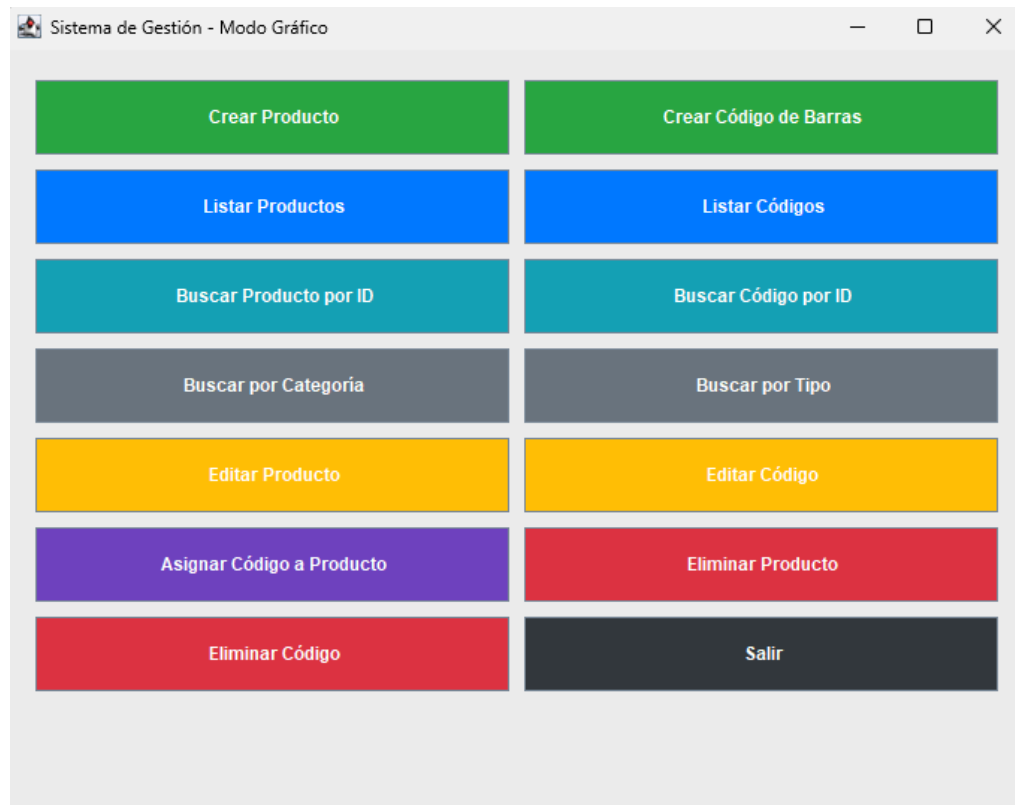
```
Seleccione una opción:
12
--Productos por Categoría--
Ingrese la categoría a filtrar:
Libros
Productos encontrados:
Producto(id=10, eliminado=false, nombre='Premium Nestle Libros', marca='Dell', categoria='Libros', precio=117.09, peso=1.239, codigoBarrasId=10)
Producto(id=11, eliminado=false, nombre='Portátil Nestle Deportes', marca='Nestle', categoria='Libros', precio=168.03, peso=6.983, codigoBarrasId=11)
Producto(id=19, eliminado=false, nombre='Avanzado Puma Electrónicos', marca='Adidas', categoria='Libros', precio=641.06, peso=8.885, codigoBarrasId=19)
Producto(id=27, eliminado=false, nombre='Premium Zara Salud', marca='Ikea', categoria='Libros', precio=955.12, peso=4.466, codigoBarrasId=27)
Producto(id=28, eliminado=false, nombre='Avanzado Ikea Juguetes', marca='Zara', categoria='Libros', precio=1126.72, peso=1.154, codigoBarrasId=28)
Producto(id=36, eliminado=false, nombre='Portátil Adidas Salud', marca='CocaCola', categoria='Libros', precio=169.23, peso=6.133, codigoBarrasId=36)
Producto(id=37, eliminado=false, nombre='Premium CocaCola Juguetes', marca='Apple', categoria='Libros', precio=499.17, peso=0.03, codigoBarrasId=37)
Producto(id=38, eliminado=false, nombre='Nuevo Under Armour Ropa', marca='Nike', categoria='Libros', precio=183.32, peso=6.37, codigoBarrasId=38)
Producto(id=39, eliminado=false, nombre='Avanzado LG Libros', marca='Adidas', categoria='Libros', precio=369.36, peso=0.055, codigoBarrasId=39)
Producto(id=44, eliminado=false, nombre='Portátil Lenovo Juguetes', marca='Adidas', categoria='Libros', precio=967.85, peso=3.694, codigoBarrasId=44)
Producto(id=53, eliminado=false, nombre='Compacto Under Armour Electrónicos', marca='Dell', categoria='Libros', precio=106.56, peso=4.837, codigoBarrasId=53)
Producto(id=82, eliminado=false, nombre='Nuevo Dell Muebles', marca='Dell', categoria='Libros', precio=155.45, peso=1.511, codigoBarrasId=82)
Producto(id=103, eliminado=false, nombre='Nuevo Puma Muebles', marca='Adidas', categoria='Libros', precio=1080.66, peso=3.375, codigoBarrasId=103)
Producto(id=112, eliminado=false, nombre='Pro Ikea Muebles', marca='Samsung', categoria='Libros', precio=56.39, peso=8.826, codigoBarrasId=112)
Producto(id=115, eliminado=false, nombre='Smart Ikea Hogar', marca='Apple', categoria='Libros', precio=214.05, peso=7.51, codigoBarrasId=115)
Producto(id=122, eliminado=false, nombre='Eco Under Armour Hogar', marca='LG', categoria='Libros', precio=187.29, peso=0.012, codigoBarrasId=122)
Producto(id=134, eliminado=false, nombre='Avanzado Zara Hogar', marca='Lenovo', categoria='Libros', precio=632.66, peso=1.192, codigoBarrasId=134)
Producto(id=137, eliminado=false, nombre='Premium HP Muebles', marca='HP', categoria='Libros', precio=91.37, peso=2.744, codigoBarrasId=137)
Producto(id=146, eliminado=false, nombre='Premium Nike Salud', marca='Sony', categoria='Libros', precio=653.12, peso=0.057, codigoBarrasId=146)
Producto(id=148, eliminado=false, nombre='Portátil CocaCola Muebles', marca='Zara', categoria='Libros', precio=326.22, peso=0.013, codigoBarrasId=148)
Producto(id=149, eliminado=false, nombre='Smart Apple Ropa', marca='Adidas', categoria='Libros', precio=907.81, peso=6.138, codigoBarrasId=149)
Producto(id=167, eliminado=false, nombre='Ultra Samsung Ropa', marca='Samsung', categoria='Libros', precio=353.77, peso=2.051, codigoBarrasId=167)
Producto(id=170, eliminado=false, nombre='Eco Samsung Muebles', marca='Under Armour', categoria='Libros', precio=705.24, peso=0.44, codigoBarrasId=170)
Producto(id=173, eliminado=false, nombre='Premium Adidas Alimentación', marca='Under Armour', categoria='Libros', precio=541.87, peso=0.352, codigoBarrasId=173)
Producto(id=178, eliminado=false, nombre='Premium Lenovo Salud', marca='Nike', categoria='Libros', precio=134.79, peso=2.029, codigoBarrasId=178)
Producto(id=183, eliminado=false, nombre='Ultra Zara Ropa', marca='Adidas', categoria='Libros', precio=404.92, peso=0.541, codigoBarrasId=183)
Producto(id=186, eliminado=false, nombre='Eco Zara Muebles', marca='Puma', categoria='Libros', precio=1150.29, peso=7.261, codigoBarrasId=186)
Producto(id=215, eliminado=false, nombre='Portátil HP Alimentación', marca='LG', categoria='Libros', precio=189.97, peso=1.272, codigoBarrasId=215)
Producto(id=220, eliminado=false, nombre='Portátil HP Libros', marca='CocaCola', categoria='Libros', precio=142.11, peso=4.616, codigoBarrasId=220)
Producto(id=227, eliminado=false, nombre='Ultra Adidas Juguetes', marca='Puma', categoria='Libros', precio=190.02, peso=4.371, codigoBarrasId=227)
Producto(id=239, eliminado=false, nombre='Pro Nike Electrónicos', marca='Zara', categoria='Libros', precio=538.29, peso=3.212, codigoBarrasId=239)
Producto(id=241, eliminado=false, nombre='Smart HP Ropa', marca='Lenovo', categoria='Libros', precio=90.09, peso=3.568, codigoBarrasId=241)
```

Listado por categoría de código de barras.

```
Seleccione una opción:
13
--Códigos de Barras por Tipo--
Seleccione el tipo de código de barras:
A) EAN13
B) EAN8
C) UPC
Ingrese su opción (A, B, C, etc.):
b
CodigoBarras(id=3, eliminado=false, tipo=EAN8, valor='840000000002', fechaAsignacion=2025-03-26, observaciones='null')
CodigoBarras(id=5, eliminado=false, tipo=EAN8, valor='770000000004', fechaAsignacion=2025-07-07, observaciones='null')
CodigoBarras(id=6, eliminado=false, tipo=EAN8, valor='770000000005', fechaAsignacion=2025-02-12, observaciones='null')
CodigoBarras(id=9, eliminado=false, tipo=EAN8, valor='840000000008', fechaAsignacion=2025-01-10, observaciones='null')
CodigoBarras(id=11, eliminado=false, tipo=EAN8, valor='840000000010', fechaAsignacion=2025-06-02, observaciones='null')
CodigoBarras(id=20, eliminado=false, tipo=EAN8, valor='840000000019', fechaAsignacion=2025-03-30, observaciones='null')
CodigoBarras(id=27, eliminado=false, tipo=EAN8, valor='770000000026', fechaAsignacion=2025-03-23, observaciones='null')
CodigoBarras(id=28, eliminado=false, tipo=EAN8, valor='840000000027', fechaAsignacion=2025-10-04, observaciones='null')
CodigoBarras(id=30, eliminado=false, tipo=EAN8, valor='770000000029', fechaAsignacion=2025-07-18, observaciones='null')
CodigoBarras(id=31, eliminado=false, tipo=EAN8, valor='840000000030', fechaAsignacion=2025-02-21, observaciones='null')
CodigoBarras(id=32, eliminado=false, tipo=EAN8, valor='770000000031', fechaAsignacion=2025-04-08, observaciones='null')
CodigoBarras(id=34, eliminado=false, tipo=EAN8, valor='770000000033', fechaAsignacion=2025-01-02, observaciones='null')
CodigoBarras(id=41, eliminado=false, tipo=EAN8, valor='840000000040', fechaAsignacion=2025-01-06, observaciones='null')
CodigoBarras(id=42, eliminado=false, tipo=EAN8, valor='770000000041', fechaAsignacion=2025-07-22, observaciones='null')
CodigoBarras(id=43, eliminado=false, tipo=EAN8, valor='770000000042', fechaAsignacion=2025-06-25, observaciones='null')
CodigoBarras(id=45, eliminado=false, tipo=EAN8, valor='770000000044', fechaAsignacion=2025-04-19, observaciones='null')
CodigoBarras(id=49, eliminado=false, tipo=EAN8, valor='770000000048', fechaAsignacion=2025-03-31, observaciones='null')
CodigoBarras(id=51, eliminado=false, tipo=EAN8, valor='840000000050', fechaAsignacion=2025-05-02, observaciones='null')
CodigoBarras(id=53, eliminado=false, tipo=EAN8, valor='770000000052', fechaAsignacion=2025-09-08, observaciones='null')
CodigoBarras(id=54, eliminado=false, tipo=EAN8, valor='840000000053', fechaAsignacion=2025-08-03, observaciones='null')
CodigoBarras(id=61, eliminado=false, tipo=EAN8, valor='840000000060', fechaAsignacion=2025-11-10, observaciones='null')
CodigoBarras(id=63, eliminado=false, tipo=EAN8, valor='770000000062', fechaAsignacion=2024-12-24, observaciones='null')
CodigoBarras(id=68, eliminado=false, tipo=EAN8, valor='840000000067', fechaAsignacion=2025-07-10, observaciones='null')
CodigoBarras(id=70, eliminado=false, tipo=EAN8, valor='770000000069', fechaAsignacion=2025-02-06, observaciones='null')
CodigoBarras(id=77, eliminado=false, tipo=EAN8, valor='840000000076', fechaAsignacion=2025-10-15, observaciones='null')
CodigoBarras(id=81, eliminado=false, tipo=EAN8, valor='770000000080', fechaAsignacion=2025-08-13, observaciones='null')
```

Aplicación por pantalla

Interfaz general



Crear producto

The screenshot shows a dialog box titled "Crear Producto" with a close button (X) in the top right corner. On the left, there is a green square icon with a white question mark. To its right are five labels: "Nombre:", "Marca:", "Categoría:", "Precio:", and "Peso (opcional):". Each label is followed by a text input field. Below these fields is a checkbox labeled "¿Asignar código de barras?". At the bottom of the dialog are two buttons: "Aceptar" and "Cancelar".

Crear codigo

Crear Código de Barras

 Valor:

Tipo:

Observaciones:

Listar productos

Productos

| ID | Nombre | Marca | Categoría | Precio | Peso | Código |
|----|---------------|-------------|--------------|----------|---------|------------|
| 1 | Pro Pum... | Adidas | Alimenta... | \$186,56 | 0,79 kg | 8400000... |
| 2 | Compact... | Ikea | Bebidas | \$517,68 | 0,45 kg | 7700000... |
| 3 | Portátil C... | Nestle | Juguetes | \$67,93 | 0,40 kg | 8400000... |
| 4 | Avanzado... | Zara | Bebidas | \$154,56 | 0,24 kg | 8400000... |
| 5 | Ultra App... | Dell | Electróni... | \$823,93 | 0,14 kg | 8400000... |
| 6 | Smart LG... | Zara | Muebles | \$75,75 | 8,14 kg | 8400000... |
| 7 | Ultra Nes... | Zara | Muebles | \$664,71 | 0,78 kg | 7700000... |
| 8 | Smart Ad... | Zara | Alimenta... | \$708,29 | 2,74 kg | 7700000... |
| 9 | Nuevo N... | CocaCola | Bebidas | \$73,12 | 4,42 kg | 8400000... |
| 10 | Clásico A... | Nike | Muebles | \$51,51 | 4,21 kg | 7700000... |
| 11 | Pro Sam... | Lenovo | Bebidas | \$432,92 | 3,23 kg | 7700000... |
| 12 | Avanzado... | Zara | Libros | \$392,95 | 0,66 kg | 8400000... |
| 13 | Nuevo Ni... | Puma | Juguetes | \$187,56 | 6,59 kg | 7700000... |
| 14 | Portátil S... | LG | Juguetes | \$70,46 | 7,35 kg | 8400000... |
| 15 | Clásico ... | Under Ar... | Juguetes | \$595,29 | 1,63 kg | 8400000... |
| 16 | Premium... | Apple | Salud | \$251,83 | 3,63 kg | 8400000... |
| 17 | Pro Ikea ... | Zara | Bebidas | \$911,80 | 0,88 kg | 7700000... |
| 18 | Portátil P... | Nestle | Deportes | \$773,31 | 4,22 kg | 8400000... |
| 19 | Premium... | Samsung | Hogar | \$925,68 | 1,07 kg | 7700000... |
| 20 | Pro Coca... | Lenovo | Salud | \$983,08 | 0,53 kg | 8400000... |
| 21 | Clásico I... | Adidas | Salud | \$265,56 | 6,14 kg | 8400000... |
| 22 | Smart Ap... | Nike | Electróni... | \$333,39 | 9,04 kg | 7700000... |
| 23 | Smart Ap... | CocaCola | Alimenta... | \$172,41 | 9,76 kg | 7700000... |
| 24 | Premium... | Lenovo | Libros | \$66,54 | 8,39 kg | 7700000... |
| 25 | Compact... | Lenovo | Ropa | \$160,77 | 0,40 kg | 8400000... |

Listar códigos

| ID | Valor | Tipo | Fecha |
|----|--------------|-------|------------|
| 1 | 840000000000 | EAN8 | 27/04/2025 |
| 2 | 770000000001 | UPC | 14/08/2025 |
| 3 | 840000000002 | EAN8 | 21/11/2024 |
| 4 | 840000000003 | UPC | 09/02/2025 |
| 5 | 840000000004 | EAN13 | 01/01/2025 |
| 6 | 840000000005 | EAN13 | 09/07/2025 |
| 7 | 770000000006 | EAN13 | 06/01/2025 |
| 8 | 770000000007 | UPC | 26/09/2025 |
| 9 | 840000000008 | EAN13 | 03/12/2024 |
| 10 | 770000000009 | EAN13 | 15/11/2025 |
| 11 | 770000000010 | EAN13 | 29/03/2025 |
| 12 | 840000000011 | UPC | 19/03/2025 |
| 13 | 770000000012 | EAN13 | 22/04/2025 |
| 14 | 840000000013 | UPC | 31/05/2025 |
| 15 | 840000000014 | UPC | 16/08/2025 |
| 16 | 840000000015 | EAN8 | 25/01/2025 |
| 17 | 770000000016 | UPC | 30/08/2025 |
| 18 | 840000000017 | UPC | 20/12/2024 |
| 19 | 770000000018 | EAN13 | 09/10/2025 |
| 20 | 840000000019 | EAN13 | 21/09/2025 |
| 21 | 840000000020 | EAN13 | 11/10/2025 |
| 22 | 770000000021 | UPC | 17/05/2025 |
| 23 | 770000000022 | EAN8 | 17/11/2024 |
| 24 | 770000000023 | EAN8 | 17/02/2025 |
| 25 | 840000000024 | EAN13 | 05/09/2025 |

Aceptar

Buscar producto por id


Entrada

 Ingrese ID del producto:

Aceptar Cancelar

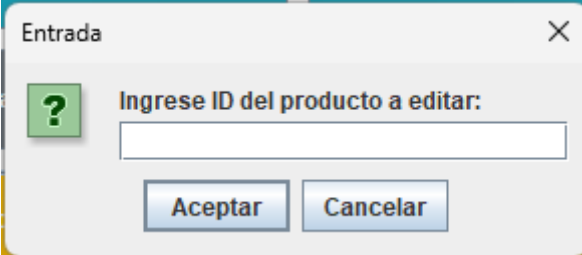
Buscar codigo por id

Entrada

 Ingrese ID del código:

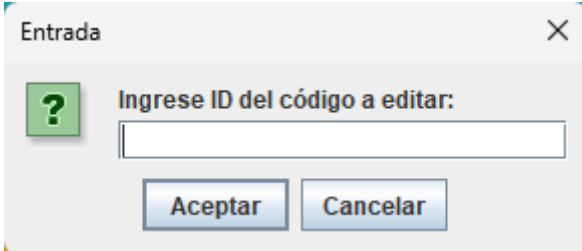
Aceptar Cancelar

Editar producto




A dialog box titled "Entrada" with a close button (X) in the top right corner. On the left, there is a green square icon containing a white question mark. To the right of the icon, the text "Ingrese ID del producto a editar:" is displayed. Below the text is a white text input field. At the bottom of the dialog, there are two buttons: "Aceptar" and "Cancelar".

Editar código



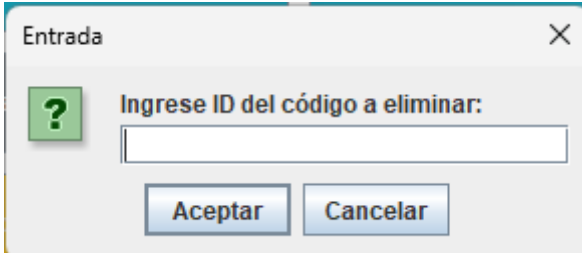
A dialog box titled "Entrada" with a close button (X) in the top right corner. On the left, there is a green square icon containing a white question mark. To the right of the icon, the text "Ingrese ID del código a editar:" is displayed. Below the text is a white text input field. At the bottom of the dialog, there are two buttons: "Aceptar" and "Cancelar".

Eliminar producto



A dialog box titled "Entrada" with a close button (X) in the top right corner. On the left, there is a green square icon containing a white question mark. To the right of the icon, the text "Ingrese ID del producto a eliminar:" is displayed. Below the text is a white text input field. At the bottom of the dialog, there are two buttons: "Aceptar" and "Cancelar".

Eliminar código



A dialog box titled "Entrada" with a close button (X) in the top right corner. On the left, there is a green square icon containing a white question mark. To the right of the icon, the text "Ingrese ID del código a eliminar:" is displayed. Below the text is a white text input field. At the bottom of the dialog, there are two buttons: "Aceptar" and "Cancelar".

Conclusión

Para concluir, hemos desarrollado una aplicación de escritorio en Java, robusta y modular, para la gestión de Productos y Códigos de Barras.

Nuestra arquitectura se basa en una clara separación de capas: Entidades que modelan los datos, DAOs que manejan el acceso a la base de datos con JDBC, y Servicios que encapsulan toda la lógica de negocio y el control de transacciones. La interacción con el usuario se gestiona a través de controladores, tanto para la interfaz de consola como para la gráfica en Swing.

Un pilar fundamental es nuestra capa de utilidades para el manejo de errores. El InputValidator asegura que la entrada del usuario sea correcta desde el principio, previniendo datos malformados. Si un error ocurre, ya sea de validación o una excepción de la base de datos, nuestro ManejadorExcepciones lo centraliza, proveyendo feedback claro y manteniendo la robustez del sistema.

Aprendizajes y Mejoras Futuras

Este proyecto ha sido fundamental para consolidar nuestros conocimientos en la arquitectura de software. Aprendimos en la práctica el valor de la separación de capas, la importancia de un manejo de errores centralizado y los retos del acceso a datos con JDBC, especialmente en lo que respecta al control explícito de transacciones y la gestión de recursos.

Este diseño no solo cumple con todos los requisitos funcionales del CRUD, sino que también crea una base sólida y mantenible. Como **mejoras futuras**, identificamos varias líneas de acción:

- **Migrar a un ORM:** Reemplazar JDBC por una solución como JPA (Hibernate) para abstraer la complejidad del SQL y mejorar la productividad.
- **Modernizar la UI:** Evolucionar la interfaz de Swing a JavaFX o, idealmente, convertir la aplicación en un servicio (API REST) con un cliente web moderno como React, Angular o Vue.
- **Seguridad:** Implementar un módulo de autenticación y autorización de usuarios para gestionar roles y permisos dentro del sistema.

5. Anexo

Fuentes y Herramientas

Las herramientas utilizadas y las fuentes de apoyo fueron cruciales para la implementación de la arquitectura de capas, las consultas complejas y la gestión transaccional.

| Categoría | Herramienta(s) Utilizada(s) | Propósito |
|-------------------------|--|--|
| Entorno de Desarrollo | NetBeans IDE (o entorno similar, como IntelliJ IDEA) | Entorno principal para el desarrollo de código Java, gestión de dependencias Maven y depuración. |
| Base de Datos (SGBD) | MySQL Server | Servidor de base de datos relacional para la persistencia de datos. |
| Gestión de DB | MySQL Workbench / phpMyAdmin | Herramientas utilizadas para la creación de esquemas, la ejecución de scripts masivos y la validación de integridad. |
| Conexión | JDBC (Java Database Connectivity) Driver | Tecnología utilizada para la conectividad directa y transaccional entre Java y MySQL (sin usar ORM). |
| Inteligencia Artificial | Google Gemini / ChatGPT | Utilizados como tutores pedagógicos para refinar la lógica de las consultas SQL, depurar el manejo de transacciones complejas en la capa Service y validar la corrección de la arquitectura. |

Uso de IA para la asistencia en la creación de Controllers, Services, Utils y AppMenu en [este Link](#).

Uso de IA para Diseño y población de Base de Datos en estos links: ([Link 1](#), [Link 2](#)).