

# Informe de Diseño del Sistema de Gestión de Tienda de Discos

Base Conceptual: Modelo orientado a objetos siguiendo principios SOLID, DRY, y Domain-Driven Design (DDD).

## 1. Nomenclatura de Clases y Variables

Elemento	- Razón del Nombre	- Ejemplo
Store	- Representa la entidad física de la tienda	- name, address
Employee	- Modela a los trabajadores	- employeeNumber
MusicalProduct	- Clase abstracta para CD/DVD	- mediaType (Enum)
RecordingService	- Servicio de grabación personalizada	- dataCostPerGB
MultimediaContent	- Interfaz para canciones y videos	- calculateSize()

## 2. Tipos de Datos y Validaciones

Variable	- Tipo	- Razón
employeeNumber	- String	- Formato específico (no numérico puro)
mediaType	- Enum	- Valores fijos para soporte
baseCost	- double	- Manejo de decimales en costos
collaborators	- List	- Relación con artistas

Ejemplo de Validación:

validarCarnetIdentidad(String carnet) - valida formato YYMMDD#####

## 3. Uso de Interfaces y Abstracción

Elemento	- Tipo	- Razón
MultimediaContent	- Interfaz	- Polimorfismo con Song y VideoClip
PricingStrategy	- Interfaz	- Patrón Strategy para precios
MusicalProduct	- Clase abstracta	- Lógica común para CD/DVD

## 4. Enumeraciones (Enum)

MediaType: CD, DVD

## 5. Relaciones entre Clases

Relación	- Tipo	- Cardinalidad	- Razón
Store -> Employee	- Composición	- 1..*	- Empleados pertenecen a tienda
Disc -> MultimediaContent	- Agregación	- 1..*	- Disco contiene contenido
Song -> Album	- Asociación	- 1 -> 1	- Una canción pertenece a un álbum

## 6. Diseño de Reportes

Reporte	- Complejidad	- Base
Tendencias de Géneros	- Media	- Song.genre y VideoClip.genre
Distribución de Servicios	- Baja	- Agrupa RecordingService.mediaType
Impacto de Colaboradores	- Media-Alta	- Cuenta en MultimediaContent.collaborators
Costo-Beneficio de Servicios	- Alta	- baseCost + (dataCostPerGB * dataSizeGB)

## 7. Operaciones CRUD

Clase: EmployeeDAO

create(emp): valida letras y employeeNumber, persiste en BD

## 8. Principios de Diseño Aplicados

- SOLID: Responsabilidad única, Open/Closed con MultimediaContent
- DRY: Métodos en Utils reutilizados
- Encapsulación: salary con getters

## **Informe de Diseño del Sistema de Gestión de Tienda de Discos**

Conclusión:

- Abstracciones precisas, patrones probados, validaciones robustas
- Escalabilidad y mantenibilidad aseguradas

Nota: Considerar PostgreSQL, API REST, pruebas JUnit