

Nombres y Apellidos: Cristian Yanqui

Asignatura: [PROGRAMACION ORIENTADA A OBJETOS 2-SOF-3A](#)

Fecha: 15/02/2026

Actividad: Aprendizaje Autónomo 2

## **Informe de Desarrollo: Sistema de Gestión de Libros Electrónicos**

**Materia:** Programación Orientada a Objetos (2-SOF-3A) **Entregable:** Avance de Proyecto (Pasos 1 y 2)

### **1. Planificación e Identificación de Clases (Paso 1)**

Cumpliendo con la instrucción 1.3, se han identificado las estructuras de datos fundamentales para el sistema, acoplándolas a las funcionalidades de gestión bibliotecaria:

- **Estructura Base: publicacion**
  - Actúa como la superclase que contiene los atributos compartidos: título, autor y año.
- **Estructura: Ebook**
  - Especialización para libros de texto digital. Añade atributos como formato (PDF/EPUB) y número de páginas.
- **Estructura: AudioLibro**
  - Especialización para contenido multimedia narrado. Añade atributos como narrador y duración en minutos.

### **2. Implementación Técnica (Unidad 3 - Paso 2)**

Siguiendo los temas de la tercera unidad, se han integrado las siguientes capacidades al software:

#### **A. Encapsulamiento**

Se protegió la integridad de la información declarando los atributos de las estructuras en **minúscula** (ej: titulo, autor). Esto garantiza que los datos solo puedan ser modificados a través de **Setters** controlados, evitando que el sistema entre en estados inconsistentes (como un año negativo).

#### **B. Manejo de Errores**

Nombres y Apellidos: Cristian Yanqui

Asignatura: [PROGRAMACION ORIENTADA A OBJETOS 2-SOF-3A](#)

Fecha:15/02/2026

Actividad: Aprendizaje Autónomo 2

Se implementó un sistema de validación robusto mediante el tipo error nativo de Go.

- **Constructores:** Las funciones NuevoEbook y NuevoAudioLibro validan cada entrada. Si el usuario intenta registrar un libro del futuro o un formato no soportado, el sistema genera una excepción controlada que es capturada en la interfaz de usuario.

### C. Interfaces y Polimorfismo

Se definió la interfaz **Digital**. Gracias a esto, el sistema puede manejar una lista heterogénea ([]Digital) que contiene tanto E-books como Audiolibros de forma indistinta. El polimorfismo se evidencia en el método ObtenerFicha(), donde cada objeto sabe qué información específica mostrar sin necesidad de usar condicionales if/else complejos.

### 3. Documentación de Uso (Manual)

1. **Registro:** El usuario puede alimentar el sistema con nuevos títulos validando su formato.
2. **Consulta:** Se presenta un catálogo ordenado aplicando polimorfismo.
3. **Acceso:** Se simula la interacción con el contenido digital (descarga o streaming).