Sistema de Gestion y Produccion de Contenido musical

Plataforma Desarrollo: C++

Componente Evaluados : Clases Album, Artista, Cancion, Menu y Precarga

27 de Octubre de 2025

**1. Introducción**

El presente informe documenta el proceso de desarrollo e implementación de un sistema de gestión musical basado en consola, utilizando el lenguaje C++ y un enfoque de Programación Orientada a Objetos (POO). El proyecto se centra en la precarga de datos desde archivos de texto, la gestión de álbumes y canciones, la administración de usuarios y la simulación de funcionalidades de reproducción, como la selección aleatoria de canciones del catálogo o de la lista de favoritos.

2. **Objetivo**

El objetivo de este documento es detallar la estructura de la solución, la lógica de los algoritmos no triviales y las consideraciones clave para su implementación y evolución futura.

3. **Análisis del Problema y Solución Propuesta**

**a. Análisis del Problema**

El problema principal radica en cómo gestionar y relacionar grandes volúmenes de datos jerárquicos (Artistas -> Álbumes -> Canciones) y datos de usuarios (Usuarios -> Favoritos) de manera eficiente en un entorno de C++ sin bases de datos complejas.

**Requerimientos funcionales identificados:**

1. **Precarga de Datos:** Los datos (artistas, álbumes, canciones y usuarios) deben cargarse desde archivos de texto planos al inicio de la ejecución.
2. **Estructura Jerárquica:** Las canciones deben estar asociadas a álbumes, y los álbumes deben estar asociados a artistas.
3. **Gestión de Usuarios:** El sistema debe permitir el ingreso de un usuario y cargar su lista de canciones favoritas.
4. **Funcionalidades de Reproducción:** Implementar la reproducción simulada de canciones aleatorias, ya sea desde el catálogo completo o de la lista de favoritos del usuario activo.
5. La solución se basó en el paradigma de POO, utilizando clases específicas para modelar las entidades y arreglos estáticos para el almacenamiento, lo cual es apropiado para un proyecto de alcance controlado:

|  |  |  |
| --- | --- | --- |
| **Clase** | **Responsabilidad** | **Solución** |
| Álbum | Almacenar datos del álbum y las canciones que contiene. | Contiene arreglos internos para canciones, relacionándose a ellas por ID. Mantiene la duración total de forma acumulada. |
| Usuario | Almacenar datos del usuario y su lista de favoritos. | Utiliza un arreglo de enteros (idsCancionesFavoritas) para almacenar los IDs de las canciones, logrando una relación simple y directa con la clase Album. |
| Artista | Almacenar datos del artista y sus álbumes. | Simplemente agrega un nivel de jerarquía superior para búsquedas y gestión. |
| Precarga | Carga inicial de datos desde archivos | Incluye la lógica para descomponer las líneas de texto basadas en el delimitador |

d. Problemas de desarrollo que afrontó.

- **Crashed**: Al ejecutar algunas veces para hacer pruebas el programa, nos ocurría un crash debido al desbordamiento del buffer

- **Funciones no declaradas:** Algunos métodos de las clases implementadas no estaban declaradas en h, lo que nos causaba errores de vinculación.

- **Modulación:** Los archivos de texto plano en la fase de prueba no fueron definidos en la carpeta correspondiente impidiendo la lectura correcta de los mismos.

- **Control de Versiones:** Presentamos problemas de incompatibilidad.

Diagrama

El contenido generado por IA puede ser incorrecto.