Sistema Gestor de Cinema - Documentación Técnica

Bibliotecas Utilizadas

El sistema utiliza las siguientes bibliotecas principales:

iostream

Esta biblioteca proporciona las funcionalidades básicas de entrada y salida estándar en C++. Se utiliza para la interacción con el usuario a través de la consola, permitiendo mostrar información (`cout`) y recibir datos (`cin`).

pqxx/pqxx

Esta es la biblioteca oficial de C++ para PostgreSQL. Proporciona una interfaz orientada a objetos para interactuar con bases de datos PostgreSQL, permitiendo realizar conexiones, ejecutar consultas y manejar transacciones de manera segura y eficiente.

cstdlib

Proporciona funciones generales de utilidad, incluyendo gestión de procesos y sistema. En este proyecto se utiliza principalmente para la función `system()` que permite ejecutar comandos del sistema operativo.

Funciones de Utilidad

Funciones de Interfaz

limpiarPantalla()

void limpiarPantalla() {

system("clear");

}

Esta función limpia la pantalla de la consola utilizando el comando "clear" del sistema. Se utiliza para mantener una interfaz limpia y organizada durante la ejecución del programa.

limpiarEntrada()

void limpiarEntrada() {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

}

Esta función limpia el buffer de entrada, eliminando cualquier carácter residual que pudiera quedar después de una operación de lectura. Es crucial para evitar problemas en la lectura secuencial de datos.

esperarEntrada()

void esperarEntrada() {

cout << "Presione Enter para continuar...";

cin.get();

}

Esta función pausa la ejecución del programa hasta que el usuario presione Enter. Se utiliza para dar tiempo al usuario de leer los mensajes en pantalla antes de continuar.

Funciones de Validación

validarClasificacion()

bool validarClasificacion(const string& clasificacion)

Esta función verifica si una clasificación de película es válida según las opciones predefinidas del sistema. Las clasificaciones permitidas son:

- "Apta todo público"

- "+9 años"

- "+15 años"

- "+18 años"

formatearPaises()

string formatearPaises(const string& paises)

Esta función procesa una cadena de texto que contiene países separados por comas y la convierte en un formato de array compatible con PostgreSQL. Realiza las siguientes operaciones:

1. Limpia espacios en blanco innecesarios

2. Separa los países por comas

3. Añade comillas dobles a cada país

4. Genera una cadena en formato {país1,país2,país3}

Funciones de Conexión y Base de Datos

iniciarConexion()

connection iniciarConexion()

Esta función establece la conexión con la base de datos PostgreSQL. Sus características principales son:

- Configura los parámetros de conexión (nombre de base de datos, usuario, contraseña, host, puerto)

- Verifica que la conexión se establezca correctamente

- Maneja errores de conexión de manera apropiada

- Retorna un objeto connection que se utilizará en todo el programa

ejecutarSQL()

bool ejecutarSQL(connection &c, const string &query)

Esta función es un wrapper para ejecutar consultas SQL de manera segura:

- Acepta una conexión y una consulta SQL como parámetros

- Ejecuta la consulta dentro de una transacción

- Maneja errores y excepciones

- Retorna true si la ejecución fue exitosa, false en caso contrario

inicializarTablas()

void inicializarTablas(connection &c)

Esta función crea la estructura de la base de datos si no existe. Crea las siguientes tablas:

1. Movies: Almacena información general de películas

2. Directors: Información de directores

3. Actors: Información de actores

4. Characters: Personajes interpretados por actores

5. Theaters: Información de complejos de cine

6. Rooms: Salas de cine

7. Shows: Funciones programadas

8. Promotions: Promociones disponibles

9. Reviews: Reseñas de películas

Funciones de Visualización

mostrarCatalogo()

void mostrarCatalogo(connection &c)

Esta función muestra el catálogo completo de películas con toda su información:

- Utiliza joins para relacionar películas con directores

- Muestra información detallada incluyendo título, género, idioma, subtítulos, países de origen, etc.

- Formatea la salida de manera organizada y legible

mostrarComplejos()

void mostrarComplejos(connection &c)

Muestra información sobre los complejos de cine registrados:

- Nombre del complejo

- Dirección

- Información de contacto

- Formato organizado por secciones

mostrarCartelera()

void mostrarCartelera(connection &c)

Muestra la cartelera actual de funciones:

- Agrupa las funciones por día de la semana

- Muestra horarios, películas y salas

- Utiliza múltiples joins para relacionar funciones con salas y películas

Funciones de Gestión de Películas

registrarPelicula()

void registrarPelicula(connection &c)

Esta función maneja el proceso completo de registro de una nueva película. Sus características principales incluyen:

1. Recopilación de Información General:

- Títulos (distribución y original)

- Género e idioma

- Información de subtítulos

- Países de origen

- Año de producción

- URL del sitio web

- Duración

- Clasificación de edad

- Sinopsis

2. Validaciones:

- Año de producción válido (1895-2030)

- Duración válida (horas y minutos)

- Clasificación de edad según opciones predefinidas

3. Gestión de Directores:

- Registro de nuevos directores

- Actualización de directores existentes

- Vinculación con la película

4. Gestión de Actores y Personajes:

- Registro de actores nuevos

- Actualización de actores existentes

- Registro de personajes

- Vinculación entre actores, personajes y película

5. Manejo de Transacciones:

- Uso de transacciones para garantizar la integridad de los datos

- Rollback automático en caso de error

eliminarPelicula()

void eliminarPelicula(connection &c)

Esta función gestiona la eliminación segura de películas del sistema:

1. Visualización:

- Muestra lista de películas disponibles

- Presenta información básica para identificación

2. Validaciones:

- Verifica existencia de la película

- Comprueba dependencias (funciones, reseñas)

3. Proceso de Eliminación:

- Eliminación en cascada de registros relacionados

- Orden específico para mantener integridad referencial

- Confirmación del usuario antes de eliminar

4. Seguridad:

- Manejo de transacciones para garantizar consistencia

- Verificaciones antes de eliminar

- Opción de cancelar el proceso

Sistema de Menús

menuPrincipal()

void menuPrincipal(connection &c)

Esta función implementa la interfaz principal del sistema:

1. Características:

- Interfaz clara y organizada

- Validación de entrada del usuario

- Manejo de errores robusto

- Opciones bien definidas para cada funcionalidad

2. Opciones Disponibles:

- Ver catálogo de películas

- Ver complejos de cine

- Ver cartelera actual

- Registrar nueva película

- Eliminar película

- Salir del sistema

3. Manejo de Errores:

- Validación de entrada numérica

- Captura de excepciones

- Mensajes de error descriptivos

Función Principal

main()

int main()

La función principal del sistema que:

1. Inicialización:

- Limpia la pantalla inicial

- Muestra mensaje de bienvenida

- Establece conexión con la base de datos

- Verifica y crea estructura de tablas

2. Ejecución:

- Inicia el menú principal

- Mantiene el ciclo de ejecución

- Maneja errores globales

3. Finalización:

- Limpieza de recursos

- Mensaje de despedida

- Cierre ordenado del sistema

Consideraciones de Diseño

1. Estructura Modular:

- Funciones específicas para cada tarea

- Separación clara de responsabilidades

- Código reutilizable y mantenible

2. Manejo de Errores:

- Uso consistente de try-catch

- Mensajes de error descriptivos

- Recuperación graciosa ante fallos

3. Interfaz de Usuario:

- Menús claros y organizados

- Validación robusta de entrada

- Feedback constante al usuario

4. Seguridad de Datos:

- Uso de transacciones para operaciones críticas

- Validación de datos de entrada

- Protección contra pérdida de datos

5. Usabilidad:

- Interfaz intuitiva

- Mensajes claros

- Confirmaciones en operaciones importantes