

Universidad Mariano Gálvez

Facultad de Ingeniería en Sistemas

Programación I

Ing. Alan G. Ucelo Morán



Manual Técnico

Oscar René Gonzalez Rojas

9390 - 24 – 8224

26/05/2025

Índice

Índice	2
Introducción	3
Detalle técnico	4
Requerimientos mínimos del sistema.....	5
Variables y tipos de datos	6
Procedimientos y funciones importantes	8
Diagrama de clases	10
Glosario de términos	11

Introducción

El programa desarrollado implementa un sistema de control de un Cinema. En este manual se describe el diseño, la estructura y el funcionamiento interno del sistema de gestión para un cine, desarrollado en lenguaje C++ utilizando Windows Forms como la interfaz gráfica de usuario. Este sistema fue creado con el objetivo de brindar una solución completa y automatizada para la administración de las principales operaciones de un cine, incluyendo la gestión de películas, salas, asignaciones, clientes, boletos y usuarios del sistema.

El sistema cuenta con persistencia de datos a través de archivos, y la interfaz gráfica para la interacción con los usuarios del sistema. Además, se ha implementado la autenticación de usuarios, control de acceso según el tipo de usuario, los cuales pueden ser operadores o administradores, validación de datos y la generación de reportes HTML.

Este documento está dirigido principalmente a desarrolladores, técnicos y personal encargado del mantenimiento del sistema, brindando una visión detallada de la arquitectura del software, sus componentes, el flujo de datos, y las tecnologías utilizadas. Asimismo, servirá como guía para futuras actualizaciones, depuraciones o ampliaciones del sistema.

Detalle técnico

El sistema fue desarrollado utilizando C++ como lenguaje de programación principal complementado con Windows Forms (C++/CLI) para la creación de la interfaz gráfica de usuario. Esta combinación permite integrar la potencia y eficiencia del lenguaje C++ con la facilidad de diseño visual proporcionada por el entorno de desarrollo de Microsoft Visual Studio.

- Lenguaje de Programación C++

C++ es un lenguaje de programación de propósito general que ofrece programación orientada a objetos, gestión eficiente de memoria, y un alto rendimiento. En este proyecto, se utiliza para implementar la lógica de negocio del sistema, la manipulación de estructuras de datos dinámicas (como arreglos de punteros a objetos), el manejo de archivos, y la interacción entre clases.

Características Aplicadas:

- **Programación Orientada a Objetos (POO):** Uso de clases, herencia, encapsulamiento y métodos.
- **Manejo dinámico de memoria:** Uso de punteros para gestionar los arreglos de objetos.
- **Persistencia:** Guardado y carga de datos utilizando archivos .csv.
- **Diseño por módulos:** El código está estructurado en archivos .cpp y .h para mantener una estructura clara y mantenible.

- C++/CLI con Windows Forms

Para la interfaz gráfica se emplea C++/CLI, una extensión de C++ que permite el uso de la plataforma .NET. Mediante Windows Forms se construyó una interfaz amigable y funcional que incluye controles como pestañas, botones, cuadros de texto, combos y tablas (DataGridView), brindando así una experiencia visual estructurada y fluida.

Los beneficios que se obtuvieron al trabajar con estas tecnologías son

- **Diseño visual simplificado:** El editor gráfico de Windows Forms en Visual Studio permite arrastrar y soltar controles (botones, tablas, cuadros de texto, entre otros) lo cual acelera el desarrollo del sistema.

Requerimientos mínimos del sistema

Para garantizar el correcto funcionamiento del sistema de gestión de cine, es importante cumplir con una serie de requisitos de hardware y software en el equipo donde será instalado. A continuación, se detallan los aspectos mínimos necesarios:

- Requisitos de Hardware:
 - Procesador: Intel Core i3 o superior
 - Memoria RAM: Mínimo 4G
 - Espacio en disco duro: Al menos 500 mb de espacio libre para la aplicación y archivos generados como por ejemplo los reportes.
 - Resolución de pantalla: 1280x720 o superior.
- Requisitos de Software:
 - Sistema Operativo: Microsoft Windows 10 o superior, se recomienda una versión de 64 bits.
 - Entorno de ejecución: .NET Framework 4.7.2, se requiere por Windows Forms en C++/CLI.
- Archivos Necesarios:

Se debe contar con todos los archivos .csv para cargarlos al sistema.

 - películas.csv
 - salas.csv
 - asignaciones.csv
 - clientes.csv
 - boletos.csv
 - usuarios.csv

Variables y tipos de datos

En el desarrollo del sistema se han utilizado diferentes clases y tipos de datos, los cuales han sido importantes para poder presentar de forma eficiente la información del cine. A continuación, se detallan las variables y tipos de datos más importantes:

- **Pelicula**
 - **int**: código
 - **String**: nombre, género, clasificación, idioma, formato
 - **Double**: precio
 - **Bool**: estado (indica si la película está disponible o no)
- **Sala**
 - **Int**: código
 - **String**: nombre, ubicación, nombreEncargado, telefonoEncargado
 - **Int**: capacidad
- **Asignacion**
 - **Int**: código
 - **Pelicula* película, Sala* sala**: Punteros a objetos asociados.
 - **String**: horario, fechaEstreno
- **Cliente**
 - **Int**: código
 - **String**: nombre, apellido, identificación, teléfono, correo
 - **Std::tm**: fechaNacimiento (tiene estructura de tiempo)
- **Boleto**
 - **Int**: código, fila, columna
 - **String**: horario
 - **Std::tm**: fechaCompra
 - **Double**: total
 - **Pelicula*, Sala*, Cliente*** : Relaciones hacia las demás clases.
- **Usuario**
 - **Credenciales de usuario: string** usuarioLogin, contraseña, tipoUsuario
 - **String**: nombre, apellido, identificación, teléfono, correo, dirección.

Tipos de datos

- **Int**: Utilizado para identificadores, capacidad, fila y columna (asiento)
- **String**: Utilizado para los datos que contienen texto como por ejemplo nombres, descripciones, horarios.
- **double**: Para representar valores monetarios (precios, totales).

- **bool**: Indicadores de estado, como disponibilidad.
- **std::tm**: Estructura para gestionar fechas (fechaNacimiento, fechaCompra), compatible con funciones de formato de tiempo (std::put_time, std::get_time).
- **Punteros a clases**: Se utilizan Pelicula*, Sala*, entre otros, para mantener relaciones entre objetos y facilitar la reutilización de datos.

Estructuras dinámicas

- **Arreglos dinámicos de punteros**: Algunos que se utilizaron fueron Pelicula** películas, Cliente**.

Las estructuras y los tipos de datos permiten que el sistema sea escalable y fácil de mantener, a su vez también permite que se integren más áreas funcionales del cine, continuando con la misma lógica.

Procedimientos y funciones importantes

Durante el desarrollo del sistema del cine, se implementaron gran cantidad de métodos los cuales permiten la correcta administración de las diferentes entidades, validación de algunos campos, la interacción con archivos y la interfaz gráfica.

- **Funciones del CRUD:** Permiten la administración de objetos como películas, salas, clientes, boletos, usuarios y asignaciones.
 - **void agregar(Pelicula* pelicula):** Agrega una nueva película al arreglo dinámico.
 - **void eliminar(int indice):** Elimina un elemento por posición, desplazando los demás.
 - **void modificar(int indice, Cliente* nuevoCliente):** Reemplaza un objeto existente con una nueva instancia.
 - **int buscarPorCodigo(int codigo):** Devuelve el índice de un objeto cuyo código coincida; retorna -1 si no existe.
 - **bool existeCodigo(int codigo):** Valida si un código ya está registrado. Estas funciones están presentes en las clases ArrayPelículas, ArraySalas, ArrayClientes, ArrayBoletos, ArrayUsuarios, y ArrayAsignaciones.
- **Funciones de Persistencia de los datos:** Permite almacenar y recuperar los datos por medio de archivos .csv
 - **void guardarEnArchivo:** Escribe en archivo los datos actuales del arreglo correspondiente.
 - **void cargarDesdeArchivo:** Lee los datos de un archivo y los carga en memoria.

Estas funciones se implementan en todas las clases Array.

- **Funciones de validación:** Con estas funciones podemos verificar ciertas operaciones como, por ejemplo:
 - **bool asientoOcupado:** Determina si un asiento ya ha sido ocupado en una sala y horario específico.
 - **bool salaDisponible:** Valida que no exista una asignación de película para esa sala en el horario indicado.
 - **bool existeUsuarioLogin:** Evita duplicados al crear nuevos usuarios.
- **Funciones de Reporte:** Generan salidas en formato HTML para visualizar la información de forma estructurada.
 - **void generarReporteHTML:** Se crea un archivo HTML con una tabla de registros, ordenada según los datos de cada clase.

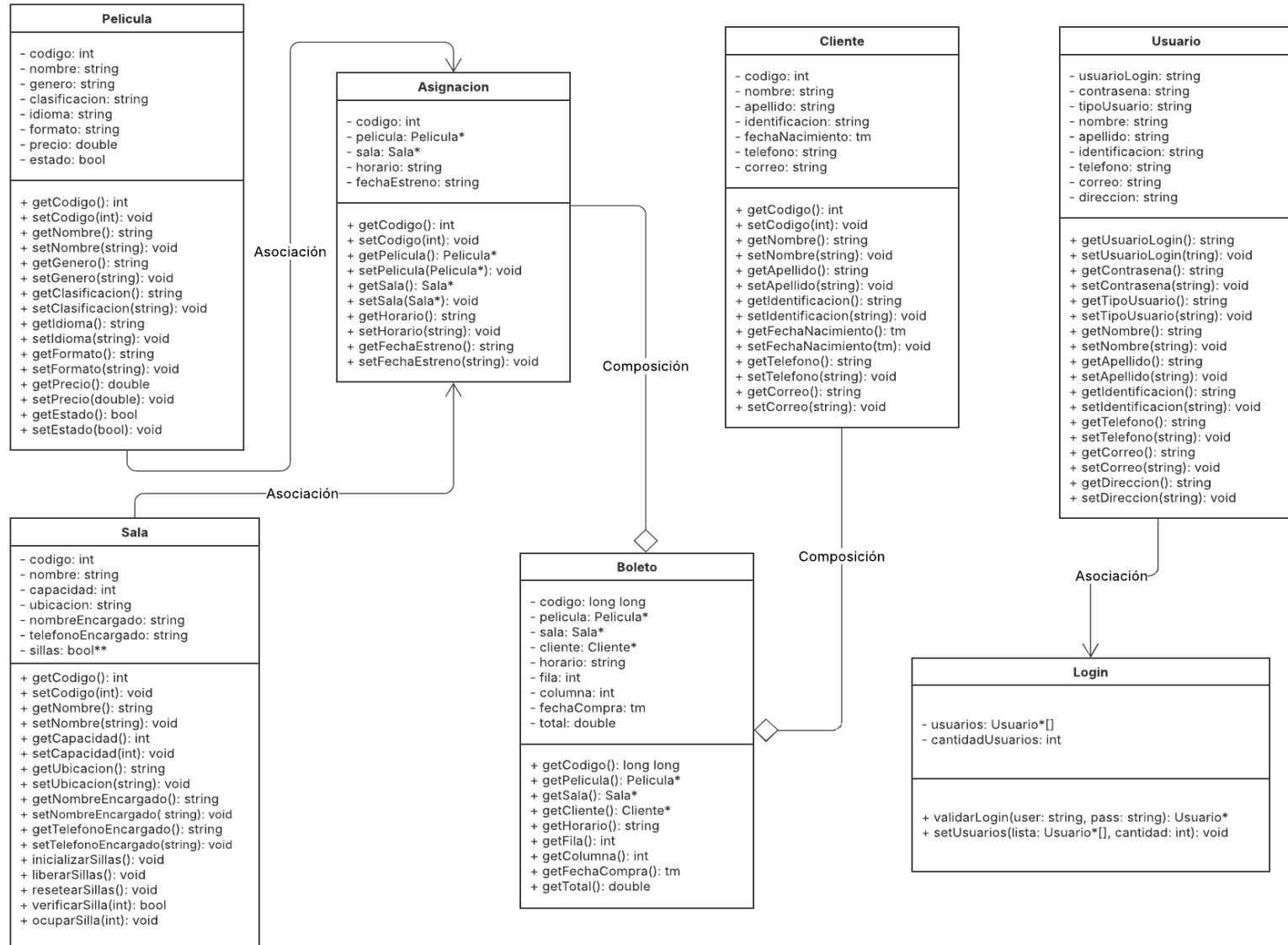
- **Funciones de la interfaz gráfica (Windows Forms):** En los formularios se manejan eventos que conectan la interfaz con la lógica.
 - **btnIngresar_Click:** Valida credenciales ingresadas por el usuario y permite el acceso dependiendo del rol.
 - **btnAgregar_Click:** Llama a los métodos correspondientes en los respectivos Array, actualizan la visualización de datos en el DataGridView.
 - **actualizarTabla():** Esta función es utilizada para recargar el contenido visual del DataGridView después de una operación de CRUD.
 - **btnGenerarReporte_Click:** Boton que invoca la función generarReporteHTML, desde el modulo que corresponde y muestra un mensaje y se abre el reporte.

Estas funciones y eventos son importantes para brindarle al usuario final una experiencia interactiva, conectando a su vez con la lógica del sistema.

Diagrama de clases

Diagrama de clases Cine

Oscar Gonzalez | May 31, 2025



Glosario de términos

CRUD: Significa Create, Read, Update, Delete. Representa las operaciones básicas que se pueden realizar sobre los datos en un sistema.

Película: Entidad del sistema que representa una obra audiovisual disponible para proyección en el cine.

Sala: Espacio físico del cine donde se proyectan las películas. Cada sala tiene capacidad, ubicación y asignaciones asociadas.

Cliente: Persona que asiste al cine y compra boletos. Se almacenan datos como nombre, identificación y contacto.

Boleto: Documento o registro que representa el derecho de un cliente a ingresar a una función específica en una sala y horario determinados.

Asignación: Relación entre una película, una sala y un horario. Indica qué película se proyecta en qué sala y a qué hora.

Usuario: Persona que opera el sistema. Puede tener rol de administrador u operador.

C++: Lenguaje de programación utilizado para implementar la lógica del sistema.

C++/CLI: Extensión de C++ para interoperabilidad con .NET, utilizado para la creación de la interfaz gráfica con Windows Forms.

Windows Forms: Framework de .NET para desarrollar aplicaciones de escritorio con interfaces gráficas en sistemas Windows.

Archivo CSV: Formato de archivo de texto plano que almacena datos separados por comas. Utilizado para persistencia de información.

Redimensionar: Proceso de aumentar la capacidad de un arreglo dinámico cuando se supera su límite inicial.

DataGridView: Componente gráfico de Windows Forms que muestra datos en forma de tabla.

Login: Proceso mediante el cual un usuario se autentica ingresando sus credenciales.

Reporte HTML: Archivo de salida en formato HTML que contiene datos tabulados generados por el sistema, visualizables desde un navegador web.

Identificador Único: Valor (generalmente un número entero) que permite distinguir de manera exclusiva una entidad (película, cliente, etc.).

Interfaz Gráfica (GUI): Forma visual mediante la cual el usuario interactúa con el sistema a través de ventanas, botones, cuadros de texto, etc.

Estructura Dinámica: Estructura de datos que permite modificar su tamaño durante la ejecución, como arreglos de punteros.

POO (Programación Orientada a Objetos): Paradigma de programación que organiza el código en clases y objetos. Facilita la reutilización, encapsulamiento y modularidad.

CLI (Common Language Infrastructure): Plataforma que permite ejecutar código en múltiples lenguajes, usada en C++/CLI para interoperar con .NET.

IDE (Integrated Development Environment): Entorno de desarrollo que integra editor de código, depurador y compilador. En este proyecto se utilizó Visual Studio.

Compilación: Proceso que transforma el código fuente en ejecutables que pueden correr en el sistema operativo.

Eventos: Mecanismo de Windows Forms que permite ejecutar funciones cuando el usuario interactúa con la interfaz (por ejemplo, al hacer clic en un botón).