



Universidad  
Católica del Norte

## Sistema de Gestión de Entradas para el Teatro de Antofagasta

(Informe Taller 3 parte 2)

Integrantes: Jhusel Toro

Nicolas Atienzo

Rut: 24.342.261-2

21.216.859-9

Correos: [Jhusel.toro@alumnos.ucn.cl](mailto:Jhusel.toro@alumnos.ucn.cl)

[nicolas.atienzo@alumnos.ucn.cl](mailto:nicolas.atienzo@alumnos.ucn.cl)

Profesor: Tomas Reimann

Ayudantes: Rodrigo Aguilera

Diego Cortez

## **1. Introducción:**

El programa desarrollado tiene como objetivo principal gestionar de manera eficiente la venta de entradas para el Teatro de Antofagasta durante su inauguración. Este sistema, diseñado en Java, utiliza el paradigma de programación orientada a objetos y está preparado para administrar usuarios, funciones (películas, obras de teatro y documentales), asientos, descuentos, y generar estadísticas clave. El enfoque del sistema es proporcionar una experiencia robusta mediante el uso de manejo de excepciones y técnicas de validación de datos.

Este informe detalla la estructura del programa, el diseño de las clases, la implementación de JavaDoc y la funcionalidad global del sistema.

## **2. Funcionalidades Principales:**

1. Gestión de Usuarios: El sistema permite registrar usuarios, iniciar sesión y unirse al Club Teatro Antofagasta, con un descuento exclusivo del 15%. Cada usuario se valida con un RUT único y contraseña segura.

- Registro de usuarios con validación para evitar duplicados.
- Inicio de sesión mediante RUT y contraseña.
- Posibilidad de unirse al ClubTeatroAntofagasta, otorgando un 15% de descuento en compras.

### **2. Gestión de Funciones:**

- Lectura de funciones desde un archivo de texto “funcion.txt”.
- Clasificación de funciones en películas, obras de teatro y documentales.
- Descuentos automáticos aplicados según el tipo de función:  
Películas nominadas al Oscar: 20% de descuento.  
Obras de teatro con entradas superiores a \$15,000: 10% de descuento.  
Documentales producidos antes de 2018: 10% de descuento.
- Administración de asientos mediante matrices que representan la disposición de una sala de 5x5. El sistema permite cargar estas funciones desde un archivo de texto, verificar su disponibilidad, y aplicar descuentos basados en reglas específicas.

### **3. Compra de Entradas:**

- Visualización de funciones disponibles.
- Selección de asientos con validación de disponibilidad.
- Emisión de una boleta detallada con la información de la compra.

### **4. Estadísticas:**

- Cantidad total de entradas vendidas.
- Porcentaje de usuarios registrados que se unieron al ClubTeatroAntofagasta.
- Función más popular.
- Tipo de función con mayor venta.

- Dinero recaudado por funciones realizadas en la noche.
- Ganancias totales durante la inauguración.

#### 4. Interfaz de función:

- Calcular el descuento de la función.

### 3. Estructura del Programa

El programa está diseñado en torno a la Programación Orientada a Objetos (POO), con clases que representan los elementos clave del sistema. A continuación, se detalla la estructura:

#### 1. Clases Principales:

- Usuario: Representa a un usuario del sistema, el usuario ingresando el Rut, la contraseña y el estado de membresía.
- Sala: Se utiliza una matriz de 5x5 asientos, así permitiendo reservar, liberar y verificar el estado de cada asiento.
- Función (clase abstracta): Clase base para representar una función del teatro.
- Película: Extiende función, añadiendo atributos específicos como género y nominación al Oscar.
- ObraTeatro: Extiende función, añadiendo atributos como autor y cantidad de actos.
- Documental: Extiende función añadiendo tema y año de producción.

#### 2. Clases de Contenedores:

- ContenedorTeatro: Administra las funciones disponibles.
- ContenedorUsuario: Administra los usuarios registrados.

#### 2. Clases de Utilidad:

- Estadística: Proporciona métodos para calcular estadística.
- SistemaTeatro: Implementa las operaciones principales del sistema.

## Lectura de Archivos

El sistema comienza cargando las funciones desde un archivo de texto `funcion.txt`. Este archivo contiene información como el código, nombre, duración, precio, horario y atributos específicos según el tipo de función.

El programa procesa esta información y crea instancias de las clases Película, ObraDeTeatro o Documental, almacenándolas en el ContenedorFunciones.

## Proceso de Compra

1. El usuario selecciona una función.
2. Se muestran los asientos disponibles en la sala.
3. El usuario elige la cantidad de entradas y especifica los números de asiento.
4. El sistema valida la disponibilidad de los asientos.
5. Se genera una boleta con los detalles de la compra.

## Estadísticas Generadas

El sistema proporciona las siguientes métricas al finalizar el evento:

- Entradas vendidas: Conteo total de entradas compradas.
- Porcentaje de usuarios en el club: Proporción de usuarios registrados que se unieron al ClubTeatroAntofagasta.
- Función más popular: Aquella que vendió más entradas.
- Tipo de función más vendido: Película, obra de teatro o documental con más ventas.
- Dinero recaudado en funciones nocturnas: Total de ingresos generados en las funciones nocturnas.
- Ganancias totales de la inauguración: Suma de todas las ventas realizadas.

## 4. Manejo de Excepciones:

El sistema emplea excepciones para manejar errores de manera predecible y segura. A continuación, se detallan las principales áreas donde se implementan excepciones:

1. Excepciones personalizadas:
  - Evita duplicados en el registro de usuarios.
  - Maneja intentos de reservar asientos ya ocupados.

## **5.Resultados del sistema:**

El sistema desarrollado ofrece:

1. Registro e inicio de sesión de usuarios con validación.
2. Lectura y carga dinámica de funciones desde archivos.
3. Compra de boletos con selección de asientos en tiempo real.
4. Aplicación de descuentos basados en reglas específicas.
5. Generación de estadísticas clave para la administración del teatro.

## **6.Conclusión**

El desarrollo del sistema de gestión de entradas para el Teatro de Antofagasta es un ejemplo claro de cómo la programación orientada a objetos (POO) y las buenas prácticas de desarrollo pueden converger para producir un software robusto, eficiente y adaptable. A lo largo del proyecto, se implementaron diferentes características para satisfacer las necesidades del cliente, y el resultado es un sistema que no solo cumple con los requisitos funcionales, sino que también destaca en aspectos técnicos clave.