

## **Taller Taquilla Del Cine**

María Paz Puerta Acevedo

Heily Yohana Rios Ayala

Universidad Autónoma

Programación Orientada a Objetos

Carlos Andres Rojas

31/03/2025

Manizales, Caldas, Colombia

## Grupo de trabajo

El taller se realizará en grupos de dos estudiantes.

## Enunciado

Se desea desarrollar un software orientado a objetos que permita facturar las entradas a un cine bajo la siguientes condiciones:

- 1- El cine tiene en su cartelera activa varias películas.
- 2- Cada película tiene un costo base independiente.
- 3- Las películas se exhiben en funciones y cada función tiene un porcentaje de descuento.
- 4- Las funciones existentes son: primera función (50%), tarde (10%), noche (0%).
- 5- Los usuarios se catalogan según su tipo, los cuales determinan un valor de descuento.
- 6- Los tipos de usuarios son: Niños (\$500), Adultos (\$0), Mayores (\$1000).
- 7- Nunca una entrada al cine podrá valer menos de \$0.

La aplicación le permitirá al dependiente de la taquilla realizar las siguientes acciones:

- 1- Vender boletas.
- 2- Generar factura.
- 3- Gestionar películas.
- 4- Gestionar usuarios.

Una venta podrá incluir una o más boletas y discriminar los costos unitarios con sus correspondientes descuentos descuentos que apliquen para calcular así su valor total.

Cualquier inquietud que se tenga acerca de los requerimientos del sistema se deberá contactar al gerente general del cine: su querido profesor.

## Entregables

1. Código fuente publicado oportunamente en UAMVirtual
2. Diagrama de clases

polimorfismo a través de clases abstractas(diagrama de clases, capa negocio)

## 1. Identificación de clases

### Clases

- Película

#### Atributos

- ☐ costoBase : double
- ☐ titulo : String

#### Métodos

- ☐ getCostoBase(): double
- ☐ getTitulo(): String

### Métodos

- Boleta

#### Atributos

- ☐ precioFinal : float

#### Métodos

- ☐ calcularPrecioFinal():void
- ☐ getPrecioFinal():Float

- Venta

#### Atributos

- ☐ totalVenta: float
- ☐ fecha : String

#### Métodos

- ☐ calcularTotal(precio:float):void
- ☐ getTotalVenta():float

- Cine

### **Atributos**

### **Métodos**

- ☐ agregarUsuario(): void
- ☐ eliminarUsuario(): void
- ☐ mostrarUsuario():void
- ☐ agregarPelicula():void
- ☐ eliminarPelicula(): void
- ☐ mostrarPelicula(): void
- ☐ agregarFuncion(Funcion) : void
- ☐ venderBoleta(): void
- ☐ calcularTotalGeneralDeBoletas() : double
- ☐ generarFactura(): void

- Cartelera

### **Atributos**

### **Métodos**

- ☐ crearPelicula():void
- ☐ buscarPelicula(): void
- ☐ actualizarPelicula ():void
- ☐ eliminarPelicula():void

## **Clases abstractas**

- Funcion (Abstracta)

### **Atributos**

- ☐ porcentajeDescuento: double

### **Métodos abstractos**

- ☐ getDescuento ():float

☐ calcularPorcentajeDescuento(precioBase: double): double

### Subclases de Funcion

- PrimerFuncion

#### Atributos

☐ double porcentajeDescuento = 0.50

#### Métodos

☐ getDescuento ():float

☐ calcularPorcentajeDescuento(precioBase : double): double

- FuncionTarde

#### Atributos

☐ porcentajeDescuento = 0.10

#### Métodos

☐ getDescuento ():float

☐ calcularPorcentajeDescuento(precioBase : double): double

- FuncionNoche

#### Atributos

☐ porcentajeDescuento= 0.0

#### Métodos

☐ getDescuento ():float

☐ calcularPorcentajeDescuento(precioBase : double): double

- Usuario (Abstracta)

#### Atributos

☐ valorDescuento : double

☐ nombre : String

☐ numTelefono: String

#### Métodos

☐ getDescuento():double

☐ calcularDescuentoFinal(precioBase:double):double

### Subclases de Usuario

- UsuarioNiño

#### Atributos

☐ valorDescuento= 500

#### Métodos

☐ getDescuento():double

☐ calcularDescuentoFinal(precioBase:double):double

- UsuarioAdulto

#### Atributos

☐ valorDescuento= 0

#### Métodos

☐ getDescuento():double

☐ calcularDescuentoFinal(precioBase:double):double

- UsuarioMayor

#### Atributos

☐ valorDescuento= 1000

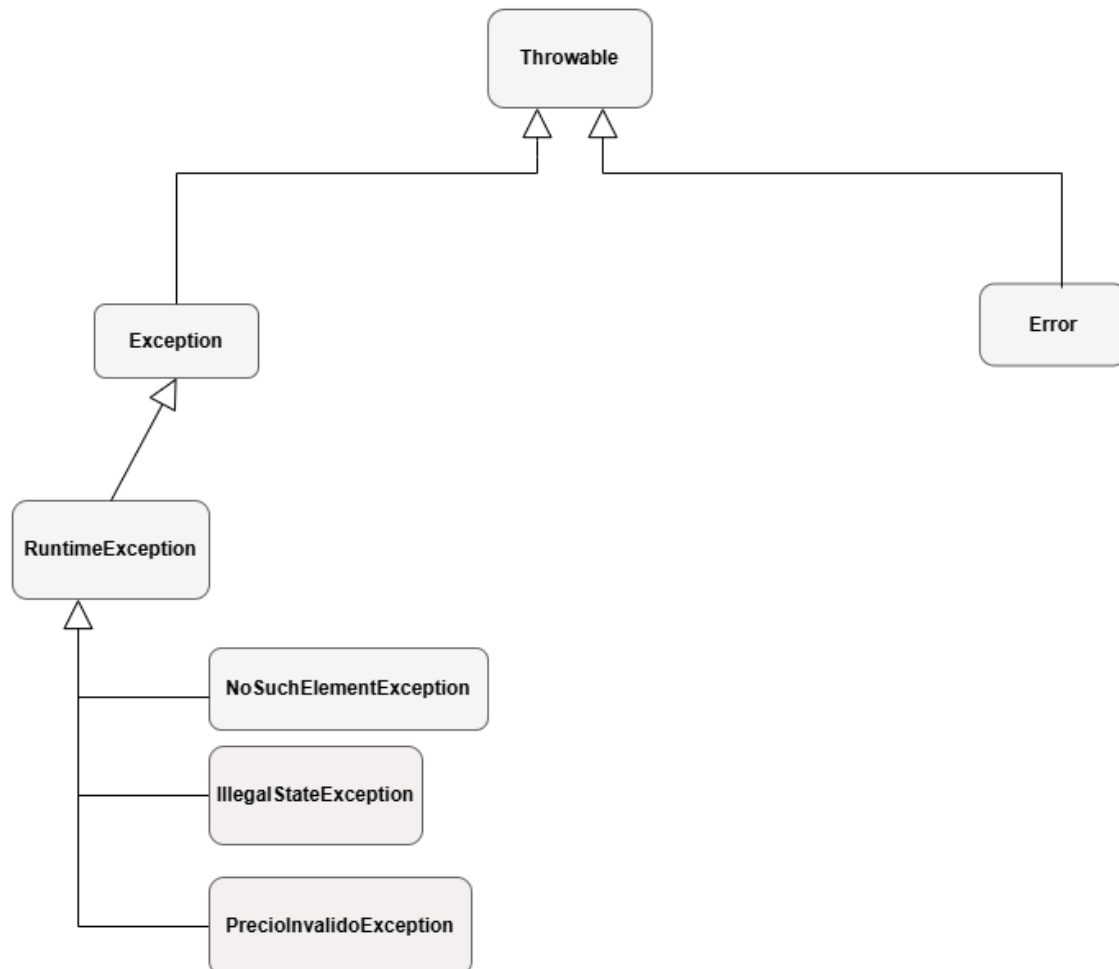
#### Métodos

☐ getDescuento():double

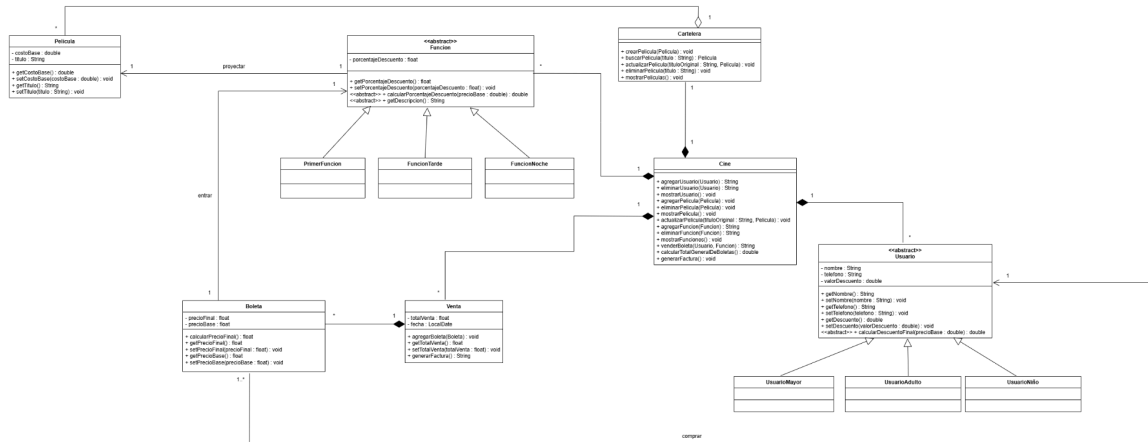
☐ calcularDescuentoFinal(precioBase:double):double

## 2. Lista excepciones

- NoSuchElementException : Se lanzará esta excepción cuando una película no sea encontrada.
- IllegalStateException : Se lanzará esta excepción cuando no haya películas en cartelera.
- PrecioInvalidoException: Se lanzará esta excepción, cuando por ejemplo el total sea -0.



### 3. Diagrama de clases



4. Enlace git <https://github.com/MariaPazPuertaa/TaquillaCine.git>