



Universidad Autónoma de Nuevo León
Facultad De Ciencias Fisicomatemáticas

Licenciatura en
Ciencias Computacionales

Proyecto Final

Manual de usuario

Laboratorio de Programación Orientada a Objetos

Grupo:

033

Equipo:

Alumno	Matrícula
Alejandro Beltrán Alvarado	1967181
Miguel Giovanny Vargas Cantú	1954779

Docente:

José Luis Candelario Tovar

San Nicolás de los Garza, Nuevo León

A 26 de octubre de 2021

Índice

Introducción	3
Recorrido.....	4
<i>Documentación de código</i>	19
Diagrama.....	24
Conclusión	25
Anexos	26

Introducción

Este prototipo es nuestro Proyecto Final de la materia impartida por el docente José Luis Candelario Tovar que lleva por nombre Laboratorio de Programación Orientada a Objetos, dicho proyecto es acerca de la venta de boletos (de manera presencial) en un cine, podríamos decir que es un software para empresas que tienen un cine, donde tendrán varias opciones que tocaremos mas a fondo más adelante.

Una de las razones por la cual decidimos hacer este proyecto es que ambos somos fan del cine, y hasta en el proceso tuvimos duda de como era dado que la fecha es Octubre del 2021, y en nuestro país, México se sigue en pandemia, y no hemos podido salir al cine en todo este tiempo, aparte nos surgieron varios temas dado que el cine poco a poco esta siendo reemplazado por plataformas de streaming, pero aun así fue un buen proyecto dado que nos dio una perspectiva de lo que pudiese ser crear proyectos para empresas que tienen una gran utilidad en el mercado y muchos pagarían por ello, claro esta que solo rascamos la superficie, pero podría decirse que esto ya es utilizable, es mejorable, por eso lo consideramos un prototipo, pero es utilizable debido a que cumple sus funciones como tal.

En lo siguiente con respecto al manual de usuario lo que se podrá apreciar en este mismo será una explicación sobre las distintas funcionalidades que te posee el software, así como una pequeña explicación más que nada conceptual de lo que sería el código que existe por atrás, y su diagrama de clases para una visualización clara.

Desarrollo

Recorrido

Iniciemos mostrándonos las distintas funcionalidades y que nos ofrece este software, todo esto inicia desde consola, al iniciarlo vemos lo siguiente:

```
run:
  Iniciar sesion
-----
Usuario:
```

Como vemos nos solicita un Usuario, si usted es su primer contacto con el software, es importante saber como iniciarlo, ya que este software este hecho de forma que no sea necesario que usted en su local tenga a su disposición alguien de sistemas, simplemente iniciará colocando Admin, con A mayúscula, y después le aparecerá contraseña, ahí colocaremos Password.

```
run:
  Iniciar sesion
-----
Usuario: Admin
Contraseña: Password
```

Podremos darle enter y nos aparecerá lo siguiente:

```
.
  Aplicacion cinepolis
-----
Bienvenido Admin!

Que desea hacer?
1. Venta de boletos
2. Busqueda de Ticket
3. Panel de Administracion
4. Cerrar sesion
5. Apagar sistema

Opcion: |
```

Dado que es el primer contacto con el software no hay nada preestablecido, solo el Admin y Password, por lo tanto, requerimos crear las funciones, agregar los usuarios, las películas, etc...

Pero primero demos de alta a los empleados, estos vendrían a ser los usuarios que tendrán acceso al software, podemos darlos de alta de la siguiente forma:

- Escribimos 3 que es el Panel de Administración en el menú.

```
-----
Bienvenido Admin!

Que desea hacer?
1. Venta de boletos
2. Busqueda de Ticket
3. Panel de Administracion
4. Cerrar sesion
5. Apagar sistema

Opcion: 3
```

- Escribimos 1 que es Usuarios en el submenú.

```
-----
Panel de Administracion
-----
A que recurso desea acceder?

1. Usuario
2. Peliculas
3. Salas
4. Salir

Seleccione una opcion: 1
```

- Escribimos 1 de nuevo dado que es Agregar en el submenú.

```
-----
Menu de Usuario
-----
Que desea hacer?

1. Agregar
2. Modificar
3. Ver informacion
4. Salir

Seleccione una opcion: 1
```

Ahora nos aparecerá Ingrese su nombre de usuario, así también la contraseña que tendrá este usuario, aquí es a elección del cliente, en este manual usaremos como ejemplo el siguiente usuario:

Nombre de usuario: Jose

Contraseña: 1928

Agregar usuario

Ingrese su nombre de usuario: Jose

Ingrese su contraseña: 1928

- Se confirma la adición de un nuevo usuario y nos solicita presionar enter

Agregar usuario

Ingrese su nombre de usuario: Jose

Ingrese su contraseña: 1928

Usuario Agregado exitosamente

Digite enter para continuar: |

- Ahora podemos dar 4 de manera repetida para lograr salir del todo, hasta llegar al menú inicial. Ahora lo que haremos será verificar que funciona este usuario cerrando sesión con 4.

Aplicacion cinepolis

Bienvenido Admin!

Que desea hacer?

1. Venta de boletos
2. Busqueda de Ticket
3. Panel de Administracion
4. Cerrar sesion
5. Apagar sistema

Opcion: 4|

- Y ingresamos el nuevo usuario que tenemos

Iniciar sesion

Usuario: Jose

Contraseña: 1928

- Como vemos funciona a la perfección

```
Aplicacion cinepolis
```

```
-----
Bienvenido Jose!
```

```
Que desea hacer?
```

1. Venta de boletos
2. Busqueda de Ticket
3. Panel de Administracion
4. Cerrar sesion
5. Apagar sistema

```
Opcion: |
```

- Podemos también desactivarlo ingresando al Menú de Usuario, y en este mismo presionamos 2 que sería Modificar en el submenú.

```
Menu de Usuario
```

```
-----
Que desea hacer?
```

1. Agregar
2. Modificar
3. Ver informacion
4. Salir

```
Seleccione una opcion: 2
```

- Ahora se nos solicitara el usuario, para ello ingresaremos Jose

```
Modificar usuario
```

```
-----
Digite el usuario que desea modificar: Jose
```

- Podemos modificar el nombre, contraseña y cambiar el estado de la cuenta, supongamos que Jose se lastimo y no podrá ir al trabajo, podemos pasar el estado a inactivo haciendo imposible su ingreso en el acceso de la aplicación, así que presionamos 3.

```
Modificar usuario
```

```
-----
Que desea modificar?
```

- 1) Nombre de usuario
 - 2) Contraseña
 - 3) Cambiar estado de la cuenta
- ```
Ingrese una opcion: 3|
```

- Se confirma y nos pide presionar enter

```

Modificar usuario

Que desea modificar?

1) Nombre de usuario
2) Contraseña
3) Cambiar estado de la cuenta
Ingrese una opcion: 3

El usuario ha cambiado de status correctamente
Digite enter para continuar:

```

- Procederé a salir de sesión e intentar entrar de nueva cuenta, veremos como nos salta un error de usuario inactivo.

|                                                                  |                                                                                   |                                                                       |
|------------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| <pre> Iniciar sesion ----- Usuario: Jose Contraseña: 1928 </pre> |  | <pre> El usuario está inactivo Iniciar sesion ----- Usuario:   </pre> |
|------------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------|

## Películas

Para las películas iremos al Panel de Administración y presionaremos 2 que es el correspondiente a Películas en el submenú.

```

Panel de Administracion

A que recurso desea acceder?

1. Usuario
2. Peliculas
3. Salas
4. Salir

Seleccione una opcion: 2

```

Veremos este menú, podemos agregar películas, modificarlas y ver información de las películas que tenemos actualmente en nuestro cinema.

```

Menu de Pelicula

Que desea hacer?

1. Agregar
2. Modificar
3. Ver informacion
4. Salir

Seleccione una opcion:

```



## 1. Agregar

Para agregar presionamos 1, acto seguido aparecerá lo siguiente:

```
Nueva película

1. Subtitulada
2. Doblada

Ingrese el tipo de película:
```

Podemos seleccionar como sería la nueva película, seleccionare subtitulada, acto seguido nos solicita un título, proporcionare una película de Damien Chazelle, La La Land.

```
Nueva película

1. Subtitulada
2. Doblada

Ingrese el tipo de película: 1
Ingrese el título de la película: La La Land
```

Se confirma la adición.

```
Nueva película

1. Subtitulada
2. Doblada

Ingrese el tipo de película: 1
Ingrese el título de la película: La La Land

Película añadida exitosamente
Digite enter para continuar:
```

Y con esto tendríamos todo con respecto a Agregar, veremos Ver Información para verificar que fue agregada solamente.

## 2. Ver Información

En Menú de Película presionamos 3, acto seguido aparece lo siguiente:

```
Películas en el sistema:
1. Película
 Título: La La Land
 Idioma: Subtitulada
Digite enter para continuar: |
```

Confirmando la correcta adición de la nueva película La La Land.

### 3. Modificar

Podemos modificar las películas desde el Menú de Película presionando 2, acto seguido aparecerá lo siguiente:

```
Modificar pelicula

Peliculas en el sistema:
1. Pelicula
 Titulo: La La Land
 Idioma: Subtitulada
```

Elige una pelicula:

---

Seleccionamos La La Land siendo la Película número 1.

```
Modificar pelicula

Peliculas en el sistema:
1. Pelicula
 Titulo: La La Land
 Idioma: Subtitulada
```

Elige una pelicula: 1|

---

Podemos modificar el nombre y el tipo, hagamos ambos.

```
Modificar pelicula

Que desea modificar?

1) Titulo
2) Tipo
3) Eliminar la pelicula
Ingrese una opcion:
```

Presionare 1 para el Titulo primero, aparece lo siguiente:

```
Cambiar nombre de la pelicula
```

```

Ingrese el nuevo nombre de la pelicula: |
```

Ingresaremos otra obra de Damien Chazelle esta vez la famosa Whiplash.

```
Cambiar nombre de la pelicula
```

```

Ingrese el nuevo nombre de la pelicula: Whiplash
```

```
Cambio de titulo exitoso
```

```
Digite enter para continuar:
```

Ahora presionamos 2 y 2 de nuevo dado que escogeremos doblada para ver el correcto cambio de Tipo de Idioma.

```
Cambiar tipo de pelicula
```

```

1. Subtitulada
```

```
2. Doblada
```

```
Ingrese el tipo de pelicula: 2
```

```
Cambio de idioma exitoso
```

```
Digite enter para continuar: |
```

Confirmaremos mediante Ver Información, herramienta extremadamente útil es una de las razones por las cuales decidimos optar por Agregarla dado que siempre es un buen habito tener una herramienta que te guie y te mantenga actualizado de los cambios y la información.

```
Peliculas en el sistema:
```

```
1. Pelicula
```

```
Titulo: Whiplash
```

```
Idioma: Doblada
```

```
Digite enter para continuar:
```

Como vemos se realizó de manera exitosa todo, quedaría Eliminar Película, solo escoges la película después de seleccionar Eliminar Película y la elimina por completo.

## Función

Presionamos 3 en el Panel de Administración.

```
Panel de Administracion

A que recurso desea acceder?

1. Usuario
2. Peliculas
3. Funcion
4. Salir

Seleccione una opcion: 3
```

Nos muestra el siguiente Menú.

```
Menu de Funcion

Que desea hacer?

1. Agregar
2. Eliminar Funcion
3. Ver Funciones
4. Ver Asientos
5. Salir

Seleccione una opcion: |
```

Vamos a Agregar La La Land a la Sala 1, para ello hay que ir a Agregar presionando el botón 1.

```
.
Agregar funcion

Que sala desea elegir?:
1. Sala 1 6. Sala 6
2. Sala 2 7. Sala 7
3. Sala 3 8. Sala 8
4. Sala 4 9. Sala 9
5. Sala 5 10. Sala 10

Opcion: |
```

Aparece lo anterior, las salas tienen distintos precios dado que cada una es distinta, siendo la 1,7,8,9 Normal, la 2 KIDS, la 3,5 MACROXE, la 4,6,10 4D cada una tiene su distinto precio, siendo normales 60, MACROXE 100, la 4D 200 y KIDS 120, esto es modificable desde el código en la clase Ticket para el precio y Función para el tipo de sala.

Después de elegir la sala 1 y la película 1 nos pedirá el horario siendo estas las siguientes opciones, escogeré la opción 2 dicha opción es 12-2pm

```
.
 Agregar funcion

Que horario desea elegir?:
1. 10am-12pm
2. 12pm-2pm
3. 2pm-4pm
4. 4pm-6pm
5. 6pm-8pm

Opcion: 2
Digite enter para continuar: |
```

Podemos eliminarla también presionando 2 y escogiendola la función, eliminándola por completo.

Al escoger ver funciones vemos lo siguiente:

```
.
 Funciones

Funciones en el sistema:
1. Funcion
 Sala: 1 Tipo: Normal
 Pelicula: Whiplash
 Idioma: Doblada
 Horario: 12pm-2pm

Digite enter para continuar:
```

Como vemos todo es perfecto, procedamos a ver los asientos, dichos deberían estar disponibles todos de color verde.

```
 Ver asientos

Funciones en el sistema:
1. Funcion
 Sala: 1 Tipo: Normal
 Pelicula: Whiplash
 Idioma: Doblada
 Horario: 12pm-2pm

Elige una Funcion para inspeccionar: 1
```

Como vemos todos disponibles.

.

|                              | Funcion           |
|------------------------------|-------------------|
|                              | -----             |
|                              | 1A 1B 1C 1D 1E 1F |
|                              | 2A 2B 2C 2D 2E 2F |
|                              | 3A 3B 3C 3D 3E 3F |
|                              | 4A 4B 4C 4D 4E 4F |
|                              | 5A 5B 5C 5D 5E 5F |
| Ocupado                      | Disponibile       |
| Digite enter para continuar: |                   |
| <hr/>                        |                   |

## Venta

Presionamos 1 en el menú inicial y nos desplegara lo siguiente:

```
.
.
 Venta

Nombre del cliente: |
```

Escribimos el Nombre del cliente, hagamos Pepe esta vez.

```
.
 Venta

Nombre del cliente: Pepe
```

Escogeremos la Función

```
.
Funciones en el sistema:
1. Funcion
 Sala: 1 Tipo: Normal
 Pelicula: Whiplash
 Idioma: Doblada
 Horario: 12pm-2pm

Elige una Funcion: 1|
```

Ingresamos cantidad, escogemos 2

```
.
Funciones en el sistema:
1. Funcion
 Sala: 1 Tipo: Normal
 Pelicula: Whiplash
 Idioma: Doblada
 Horario: 12pm-2pm

Elige una Funcion: 1
Ingrese la cantidad de boletos: 2
```

## Y saldrá esto

```
.
1A 1B 1C 1D 1E 1F
2A 2B 2C 2D 2E 2F
3A 3B 3C 3D 3E 3F
4A 4B 4C 4D 4E 4F
5A 5B 5C 5D 5E 5F

Ocupado Disponible

Elija un asiento: |
```

---

## Escogemos 1A

```
1A 1B 1C 1D 1E 1F
2A 2B 2C 2D 2E 2F
3A 3B 3C 3D 3E 3F
4A 4B 4C 4D 4E 4F
5A 5B 5C 5D 5E 5F

Ocupado Disponible

Elija un asiento: 1A
```

## Y escogemos 1B

```
1A 1B 1C 1D 1E 1F
2A 2B 2C 2D 2E 2F
3A 3B 3C 3D 3E 3F
4A 4B 4C 4D 4E 4F
5A 5B 5C 5D 5E 5F

Ocupado Disponible

Elija un asiento: 1B
```



## No queremos más boletos

```
.
 1A 1B 1C 1D 1E 1F
 2A 2B 2C 2D 2E 2F
 3A 3B 3C 3D 3E 3F
 4A 4B 4C 4D 4E 4F
 5A 5B 5C 5D 5E 5F

Ocupado Disponible

Elija un asiento: 1B
Desea comprar más boletos?

1) Si
2) No
```

Opcion: 2

---

## Y se nos proporciona el ticket, así como el boleto

```
.
 Ticket

Nombre el cliente: Pepe
Fecha: 2021/10/26
Codigo: 1

Boletos:

Sala: 1 Tipo: Normal Precio: 60.0
Horario: 12pm-2pm
Película: Whiplash Idioma: Doblada
2 Asientos: 1A, 1B.
Total: 120.0

Total a pagar: 120.0

Ticket guardado correctamente
Presione enter para continuar: |
```

---

## Búsqueda de ticket

Esto es sencillo en el Menú Inicial, presionamos 2 que es Búsqueda de Ticket

```
Aplicacion cinepolis

Bienvenido Admin!

Que desea hacer?
1. Venta de boletos
2. Busqueda de Ticket
3. Panel de Administracion
4. Cerrar sesion
5. Apagar sistema

Opcion: 2
```

Ingresamos el código, en este caso 1 en el ticket que creamos previamente.

```
-
-
Ingrese el codigo: 1
```

Y aparece nuestra compra

```
-
Ingrese el codigo: 1

Ticket

Nombre el cliente: Pepe
Fecha: 2021/10/26
Codigo: 1

Boletos:

Sala: 1 Tipo: Normal Precio: 60.0
Horario: 12pm-2pm
Pelicula: Whiplash Idioma: Doblada
2 Asientos: 1A, 1B.
Total: 120.0

Total a pagar: 120.0
Presione enter para continuar
```

## **Documentación del Código**

Nuestro código cuenta de 3 paquetes:

Acceso a la aplicación: El usuario accede al introducir su Usuario y Contraseña y podrá acceder al panel de administración, Venta y búsqueda de ticket.

Panel de administración: En este paquete se encuentran los recursos, se pueden agregar, modificar y eliminar.

Venta y Búsqueda: Aquí se realiza la venta de boletos y se almacena la información en tickets, también podemos buscar los tickets en el sistema.

### **Clases en nuestros paquetes**

#### **Panel de administración**

##### **Usuario**

Atributos: (-) String nombreUsuario, (-) String contraseña, (-) boolean status.

Métodos:

Constructor: (-) Usuario(String nombreUsuario, String contraseña, boolean status).

(-) void mostrarDatos(): Despliega la información de nuestro usuario.

(-) String status(): Traduce el boolean y muestra si está activo o inactivo.

(-) static boolean confirmarUser(String name): recibe como parámetro el nombre del usuario y confirma si ya existe este usuario en el sistema.

(+) static void agregar(): Agrega objetos Usuario al sistema, le pide nombre y contraseña al usuario, lo almacena en el archivo binario Usuarios.bin.

(+) static void verInformacion(): Imprime todos los usuarios en el sistema almacenados en el archivo binario Usuarios.bin.

(+) static void modificar(): Dependiendo la opción que se elija, modificará el nombre, contraseña o estado del usuario y lo modificará en el archivo binario Usuarios.bin.

## **Película**

Atributos: (#) String titulo, (#) String idioma.

Métodos:

Constructor: (+)Pelicula(String titulo, String idioma).

(-) static boolean confirmarNombre(String name): retorna true si la película ya existe.

(-) static boolean confirmarPelicula(String name, String idioma): retorna true si la película y el idioma se encuentra en el archivo binario Películas.bin.

(+) static void verInformacion(): Imprime todas las películas en el archivo binario Películas.bin.

(+) static boolean comprobarInformacion(): Comprueba si existe al menos un objeto Pelicula en el sistema.

(+) static void modificar(): Modifica las películas almacenadas en el archivo binario Películas.bin, ya sea el titulo o el idioma.

(+) static void agregar(): Almacena en el archivo binario Películas.bin objetos Pelicula.

## **Asiento**

Atributos: (-) String nombre, (-) boolean disponible.

Constructor: (+)Asiento(String nombre, boolean disponible).

## Funcion

Atributos: (-) String tipo, (-) int nsala, (-) String horario, (-) String idioma, (-) String titulo, (-) Asiento [ ] asientos.

Contantes:

(-) static final String Horario1 = "10am-12pm".

(-) static final String Horario2 = "12pm-2pm.

(-) static final String Horario3 = "2pm-4pm".

(-) static final String Horario4 = "4pm-6pm".

(-) static final String Horario5 = "6pm-8pm".

Métodos:

Constructores:

(+) Funcion(String tipo, String horario, int nsala, String titulo, String idioma).

(+) Funcion(String tipo, String horario, int nsala, String titulo, String idioma, Asiento[] asientos).

(-) static String tipoSala(int nsala): Elige la sala, devuelve la sala.

(+) static String tipoSala(int nsala): Retorna el tipo de sala dado la sala que le den.

(-) static int elegirSala(): Elige la sala, devuelve la sala.

(-) static String elegirHorario(int nusala): Elige el horario devuelve el horario indicado.

(-) static boolean disponibilidadh(int nusala, String Horario, int h): Checa la disponibilidad de la sala dado un horario.

(+) static void agregar(): Agrega una funcion al sistema.

(-) private static Pelicula objetoPelicula(String titulo, String idioma): Devuelve la instancia de un objetoPelicula.

(+) static Pelicula elegirPelicula(int n): Elige una pelicula en el sistema.

(+) static Funcion objetoFuncion(String titulo, String idioma, String horario, String tipo, int nsala): Devuelve un objetoFuncion.

(+) static void eliminarFuncion(): Elimina una funcion.

(+) static void verAsientos(): Ve los asientos en el sistema.

(+) static void modificar(int nsala,String titulo, String horario,String idioma, int op): Elimina-modifica funciones.

(+) static void modificar(int nsala,String vt,String titulo, String horario,String idioma, int op): Modifica funciones.

(+) static void ocupar(Funcion auxx, Asiento[] asientosn): Ocupa un asiento de una funcion.

(+) static void verFunciones(): Ve las funciones en el sistema.

(-) void mostrardatos(): Muestra los datos de la funcion.

(-) static boolean salaOcupada(int nsala): Muestra si la sala está ocupada.

(+) static boolean comprobarInformacion(): Comprueba si existe un objeto en el sistema.

(+) static Asiento[] crearAsientos(Asiento[] asientos): Instancia asientos en un arreglo.

## **Venta y Búsqueda**

### **Ticket**

Atributos: (-) String ticket, (-) int matricula.

Métodos:

Constructores: (+) Ticket(String ticket, int matricula).

(-) static float precio(int nsala): Genera el precio dependiendo de la sala.

(-) static int CantidadDis(Funcion aux): Cuenta la cantidad de boletos disponibles en una funcion.

(+) static void agregar(): Agrega un Ticket en Tickets.bin.

(+) static void buscarticket(): Busca un ticket y lo imprime en pantalla.

## Acceso a la Aplicación

### Acceso

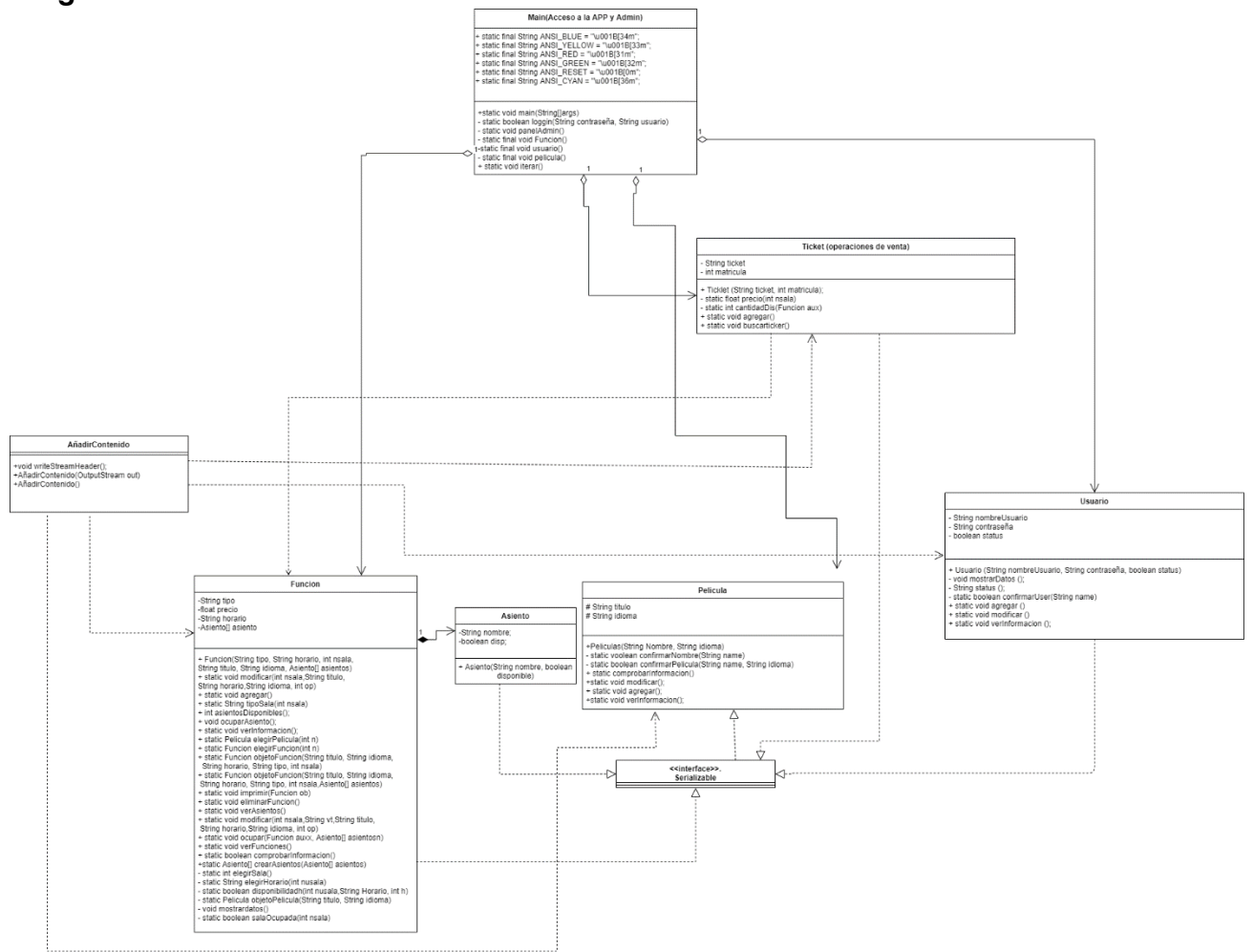
#### Constantes:

```
(+) static final String ANSI_BLUE = "\u001B[34m";
(+) static final String ANSI_YELLOW = "\u001B[33m";
(+) static final String ANSI_RED = "\u001B[31m";
(+) static final String ANSI_GREEN = "\u001B[32m";
(+) static final String ANSI_RESET = "\u001B[0m";
(+) static final String ANSI_CYAN = "\u001B[36m"
```

#### Métodos:

```
(-) static boolean login(String contraseña, String usuario):
(-) static final void panelAdmin(): Accede al panel de Acceso.
(-) static final void Funcion(): Accede a las opciones de Funcion para realizar.
(-) static final void usuario(): Accede a las opciones de usuario que se pueden realizar.
(-) static final void pelicula(): Accede a las opciones de pelicula que se pueden realizar.
(+) static void iterar(): formato solo itera saltos de línea.
(+) static void main(String[] args): corre el programa.
```

## Diagrama de clases





## **Conclusión**

### **Miguel Giovanny Vargas Cantú:**

Con este proyecto me di una mejor idea de lo que podría ser el mercado laboral, cosas que son necesarias para algunas compañías, de la misma forma que hice esto para un cine creo que ahora gracias a este proyecto puedo adaptarme con facilidad a cualquier empresa que venda cosas y tenga empleados como tal. Aprendí muchísimo en este semestre, no sabía nada respecto a Programación Orientada a Objetos, tanto practica como teoría, pero gracias a este Laboratorio y a la Clase, aprendí demasiado, ahora tengo muchas ganas de salir de vacaciones y trabajar en un proyecto mío de un videojuego, dado que programación orientada a objetos es muy cómoda para el desarrollo de videojuegos, sobre todo en Unity también que es muy amigable, disfrute crear este proyecto y solo me queda agradecerle a mi docente por todo lo que me ha enseñado, mostrándose receptivo a las dudas e incluso explicando temas fuera del temario que nos pueden servir de ayuda para el futuro, como podría serlo la globalización.

### **Alejandro Beltrán Alvarado:**

En este proyecto apliqué muchas cosas de las que aprendí a lo largo de este curso, fue un reto hacer este trabajo, me divertí mucho haciéndolo fue una gran experiencia el poder realizarlo.

**Anexo:**

Si tiene dudas respecto al uso debido del software o su software esta defectuoso puede llamar al siguiente número:

**8123445678**

O también puede mandar un e-mail a **CodersCinefilos@hotmail.es**