

## **M4 AE3 ABPRO-Ejercicio grupal**

### **Integrantes:**

- Felipe Burgos,
- Catalina Villegas

### **LINK GITHUB CON DIAGRAMA Y CODIGO PARA PARTE 2:**

[https://github.com/CatalinaMonserrat/Python\\_modulo\\_4/tree/main/ejercicio\\_CineClub](https://github.com/CatalinaMonserrat/Python_modulo_4/tree/main/ejercicio_CineClub)

### **Paso 1: Análisis y Conceptualización (30 min)**

#### **¿Qué es un diagrama de flujo y para qué sirve?**

**R./** Un diagrama de clases UML es una representación visual que modela la estructura de un sistema orientado a objetos. describe clases, sus atributos, métodos y las relaciones entre ellas ( como asociación, composición y herencia).

Sirve para organizar y visualizar cómo se comportan los objetos del sistema, lo cual ayuda a planificar la arquitectura antes de escribir el código.

#### **¿Cómo puede ayudar un diagrama de clases a estructurar el sistema?**

**R./** Un diagrama de clases puede ayudar la estructura del sistema de la siguiente manera:

- Permite identificar claramente qué entidades necesita el sistema
- Ayuda a visualizar la estructura lógica y las relaciones entre los distintos elementos
- Permite definir responsabilidades de cada clase, evitando redundancias o confusiones.

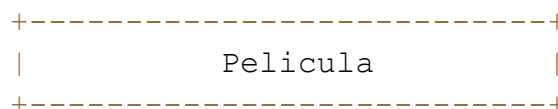
#### **¿Qué ventajas ofrece antes de la implementación?**

**R./** Dentro de las ventajas podemos encontrar:

- Claridad para el equipo: todos entienden lo que se debe construir
- Evita errores de diseño antes de programar
- Facilita la división de tareas en el equipo de desarrollo
- Sirve como guía de referencia durante la codificación.

### **Paso 2: Lectura de un Diagrama de Clases (30 min)**

Se les proporciona el siguiente diagrama de clases correspondiente al sistema CINECLUB



```

| - titulo: String      |
| - genero: String      |
| - duracion: int       |
+-----+
| + obtenerInfo(): String |
+-----+

```

(composición)

```

+-----+
|          Funcion       |
+-----+
| - fecha: Date          |
| - hora: Time           |
| - sala: int            |
+-----+
| + reservarAsiento(): bool |
+-----+

```

(colaboración)

```

+-----+
|          Usuario       |
+-----+
| - nombre: String       |
| - email: String        |
+-----+
| + comprarBoleto(): void |
+-----+

```

### ¿Cuáles son las clases principales del sistema?

R./ Las clases principiantes del sistema en el diagrama son:

1. Película
2. Función
3. Usuario

### ¿Cómo se representan los atributos y métodos?

R./

- nombre: String (atributo)

Los atributos se representan primero con un signo para definir su visibilidad (donde + es público, - es privado, # es protegido) en el caso del ejemplo planteado las 03 clases están representados con signo negativo, por ende son atributos privados, seguido de su nombre (nombre del atributo) y el tipo, si corresponde a un string, int, date, float, etc.

+ comprarBoleto(): void (metodo)

Los métodos se ubican en la parte inferior del cuadro y al igual que los atributos se comienza definiendo su visibilidad con signo, en el caso del ejemplo, las 03 clases tienen el método con signo + (público), seguido de nombre del método, parámetros y tipo de retorno.

### **¿Qué significa la composición entre Pelicula y Funcion?**

**R./** Una composición entre las clases película y función, significa que una función **no puede existir** sin una película asociada. Si la película se elimina, también la función. Es una relación de dependencia fuerte.

### **¿Cómo se representa la colaboración entre Usuario y Funcion?**

**R./** En el caso de la colaboración existente entre clase usuario y función, corresponde a una relación de asociación simple: donde el usuario interactúa con una función, por ejemplo, para comprar los boletos, pero a diferencia de la composición su dependencia no depende del otro.

### **¿Cómo se relacionan estas clases con la plataforma web que desean construir?**

**R./**

- La clase película define el contenido a proyectar
- La clase función permite gestionar las fechas, salas y horarios
- La clase usuario representa al cliente que se registra y compra boletos

Estas tres clases modelan lo que se necesita en la plataforma a construir, para reservar y administrar funciones de cine.

## **Paso 3: Diagrama de Clases**

