



Tecnológico de Monterrey

Proyecto Integrador

Programación Orientada a Objetos

Adrián Güereque de la Garza - A01722394

02 de Febrero de 2023

Índice:

Introducción.....	3
Diagrama de clases UML.....	3
Ejemplo de ejecución.....	4-5
Argumentación/Explicación.....	6
Limitaciones.....	6
Conclusión.....	6

Introducción

Hace unas décadas, la gente contrataba proveedores de un servicio de televisión de paga satelital o por cable, modalidad en la que existen varios canales que transmiten contenidos de entretenimiento programados en horarios y días específicos. Sin embargo, ahora ha cambiado sus preferencias de entretenimiento de tal forma que sus preferencias ya no son los programas de televisión tradicionales, con sus horarios y fechas programadas; ahora los usuarios prefieren disfrutar de una base de datos con contenidos específicos acoplados a sus necesidades y distribuidos por un proveedor o servidor a través de internet.

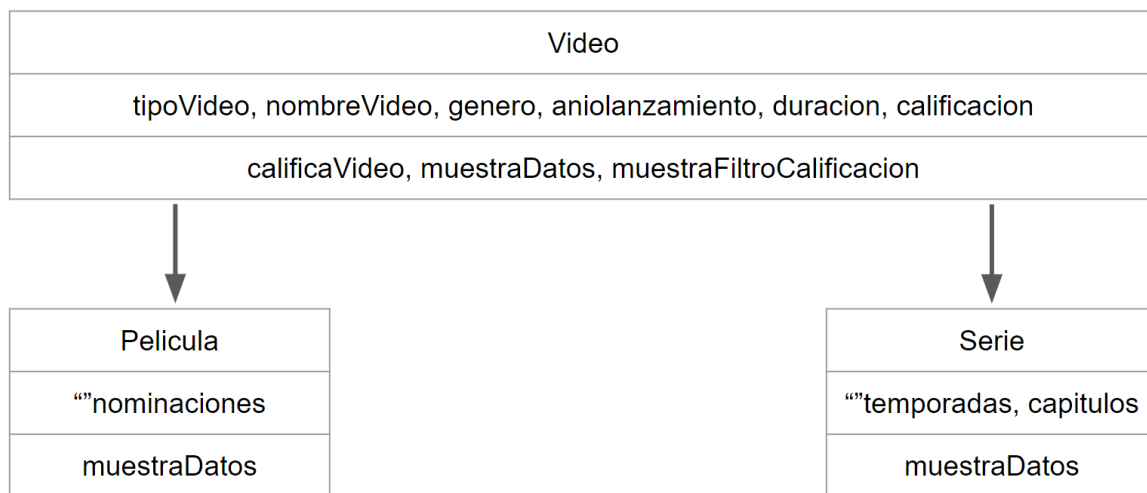
Esto es mejor conocido como: servicios de Streaming.

Esto es una nueva forma de consumir contenidos, por ejemplo: Música, vídeos, series, documentales, películas, juegos, etc. En nuestros días, podemos encontrar varios proveedores como: Netflix, Amazon Prime Video, Disney Plus, entre otros. Algunos se especializan por el volumen de vídeos que proporcionan a sus usuarios, mientras que otros se han puesto el reto de mostrar solamente vídeos de su propia marca.

Se requiere de una infraestructura amplia con una gran inversión en tecnología. Es por ello que se requiere que plantees una versión ligera de una aplicación de apoyo para un nuevo proveedor de este tipo de servicios.

Esto nos lleva a ahora. Lo que vamos a tratar de hacer es una tipo simulación de uno de estos servicios de streaming, pero a mucho menor escala. Crearemos tres clases. Una principal de Videos, que es más general, y dos derivadas de esta de Películas y de Series. Veremos que tan complicado sería hacer algo así, y a ver si nos podría jalar a una escala diminuta.

Diagrama de clases UML



En este diagrama podemos ver los atributos y métodos de cada clase. Todos los atributos y métodos de la clase **Video** son heredados por las subclases. Los atributos adicionales en las subclases son propios de la clase, y el método de `muestraDatos` se cambia en las clases para mostrar toda la información deseada con el uso de polimorfismo.

Ejemplo de ejecución (menús)

Menú principal

```
PS C:\Users\aguer\Desktop\Programacion\TEC\C++\1er semestre 3er parcial\POO Invierno> & .\"Avance_proyecto_poli
morfismo.exe"
Videos miscellaneos (1)

Series (2)

Películas (3)

█
```

Si le picas 1:

```
Consultar videos (1)

Calificar videos (2)

█
```

Si le picas 1 de nuevo

```
Filtrar por calificacion? (s/n)

█
```

Si mandas "s"

```
Introduce la calificacion: █
```

Te manda los datos filtrados y te pregunta si deseas continuar

```
Introduce la calificacion: 8
Los videos miscellaneos con esa calificacion o mayor son:
Ori and the Blind forest speedrun es un/una Gameplay genero speedrun
publicado en 2019 con duracion de 23 minutos

Su calificacion es de 10/10

Deseas continuar? (s/n)
█
```

Si "s" te regresa al inicio, si no, sale

```
Deseas continuar? (s/n)
s
Videos miscellaneos (1)

Series (2)

Películas (3)

█
```

Picandole 2

```
2
Consultar series (1)

Calificar series (2)
```

2 de nuevo y te salen los objetos de la clase series

```
Cambiar MiSerie1 (1), MiSerie2 (2) o MiSerie3 (3)?
```

Cambiando la 3 e ingresando 6

```
Cambiar MiSerie1 (1), MiSerie2 (2) o MiSerie3 (3)?
3
Introduce la calificacion:
6
Deseas continuar? (s/n)
```

Si me voy a consultar series de nuevo

```
6
Deseas continuar? (s/n)
s
Videos miscelaneos (1)

Series (2)

Películas (3)

2
Consultar series (1)

Calificar series (2)

1
Filtrar por calificacion? (s/n)
```

Y le mando "n", me muestra todas las series, incluyendo al que le cambiamos la calificación

```
Filtrar por calificacion? (s/n)
n

Stranger Things es un/una Serie genero Suspenso/Horror
publicado en 2018 con duracion de 800 minutos

Tiene 4 temporadas y 40 capitulos

Grey's Anatomy es un/una Serie genero Drama
publicado en 2004 con duracion de 800000 minutos

Su calificacion es de 3/10

Tiene 16 temporadas y 800000 capitulos

Attack On Titan es un/una Serie genero Accion
publicado en 2014 con duracion de 900 minutos

Su calificacion es de 6/10

Tiene 4 temporadas y 90 capitulos

Deseas continuar? (s/n)
```

Finalmente, es lo mismo con las otras opciones.

Argumentación/Explicación

Clases

Se crearon tres clases, una superclase y dos derivadas. No hay nada en protected porque los atributos no se manipulan en las derivadas. Si se requiere cambiar una calificación, se cambia mediante el método que está en la clase principal. Hay una clase virtual en la principal porque en las derivadas si se requiere llamar el método, necesita mostrar cosas adicionales de la superclase, y por ende se crea de nuevo el método en las clases derivadas pero con la información adicional. Finalmente, hay una función que recibe una calificación, y si la calificación del objeto es mayor o igual al recibido, se muestra la información del objeto. Esto sirve para filtrar cuáles objetos tienen cierta calificación. Ya que se llama con `this->muestraDatos`, sin importar de qué clase el objeto sea, se mostrará el `muestraDatos` de esa clase. El "this" se refiere al objeto.

Funciones

Hay una función que pide un 1 o un 2 y lo regresa, otra función que pide un 1,2 o 3 y lo regresa y una clase que pide "s" o "n" y lo regresa. Se usan para poder reutilizarse para los menús. Finalmente, hay una función que pide la calificación y la regresa. Esto es para salvar líneas de código y hacerlo más legible, ya que se utilizarán varias veces las funciones.

Main

Primero creamos los objetos, y luego ya empezamos con los menús. Siempre se va a tener que poner 1,2,3,"s" o "n". No hay otra opción excepto cuando introduces una nueva calificación. Se repite mediante while; mientras que el usuario ponga "s" cuando se le pide si quiere continuar.

Sobrecarga

La sobrecarga no se utilizó en el código, pero sí se pudiera emplear en esta situación. Si por ejemplo quisiéramos checar si algún objeto tiene calificación o no, pudiéramos sobrecargar el `==`, y si sí tiene calificación, que te diga que sí y te regrese la misma calificación.

Limitaciones/maneras que se pudiera romper el código

Siento que el código sí es legible, pero también que se podría hacer mejor. Honestamente no veo errores en el código y no debería de parar mientras que no se ponga "n" al final, pero pudiera estar mal. Esto es porque las funciones validan que el dato sea correcto. Limitaciones del código incluye que no se pueden realmente manipular otros atributos menos que la calificación, y que no se puede crear objetos fluidamente mientras corre. Es un programa medio básico pero bien hecho.

Conclusión

En conclusión, aunque no se pudo hacer una versión como los sistemas de ahora, fue una manera muy buena de aprender sobre clases, cómo funcionan, cómo manipular los atributos y métodos, cómo usar herencia, polimorfismo y sobrecarga (aunque no se usó en el código). Las clases estuvieron perfectas y la verdad no tuve problemas, pero estoy seguro de que si no hubiera puesto atención, sí estaría muy perdido. Es un tema algo avanzado y estoy alegre de que le pude entender perfectamente.