



Situación Problema

POO. Modelado de servicio de streaming

Carolina Ortega Barrios

a01025254@itesm.mx

*Escuela de Ingeniería, Instituto Tecnológico de Estudios
Superiores de Monterrey, México, México, Campus Santa Fe*

Fecha de entrega : Sábado 13 de junio de 2020

ÍNDICE DE CONTENIDO

- Introducción (Planteamiento del problema) ----- 1
- Diagrama de clases de tipo UML ----- 2
- Ejemplo de ejecución ----- 3
- Partes del proyecto ----- 4
- Conclusión ----- 5
- Referencias ----- 6

Introducción (Planteamiento del problema)

El objetivo de la situación problema de la materia de Programación Orientada a Objetos consistió en ser capaces de programar una simulación de un programa de streaming como Netflix . Netflix ha tenido un gran auge en los últimos años porque fue una de las primeras plataformas que fue capaz de mostrar a sus usuarios todo el catálogo de películas , series , documentales , etcétera que tienen disponible y además poder opinar al ponerle una calificación a ese video. Hoy en día hay tantas opciones que no podemos ver todo lo que existe y muchas veces nos guiamos por las recomendaciones de otras personas. El algoritmo de la plataforma nos enseña en nuestro inicio varias cosas , como por ejemplo :

- 1) Lo popular en la plataforma
- 2) Lo recomendado para ti con base en lo que has visto
- 3) Lo más popular en tu país

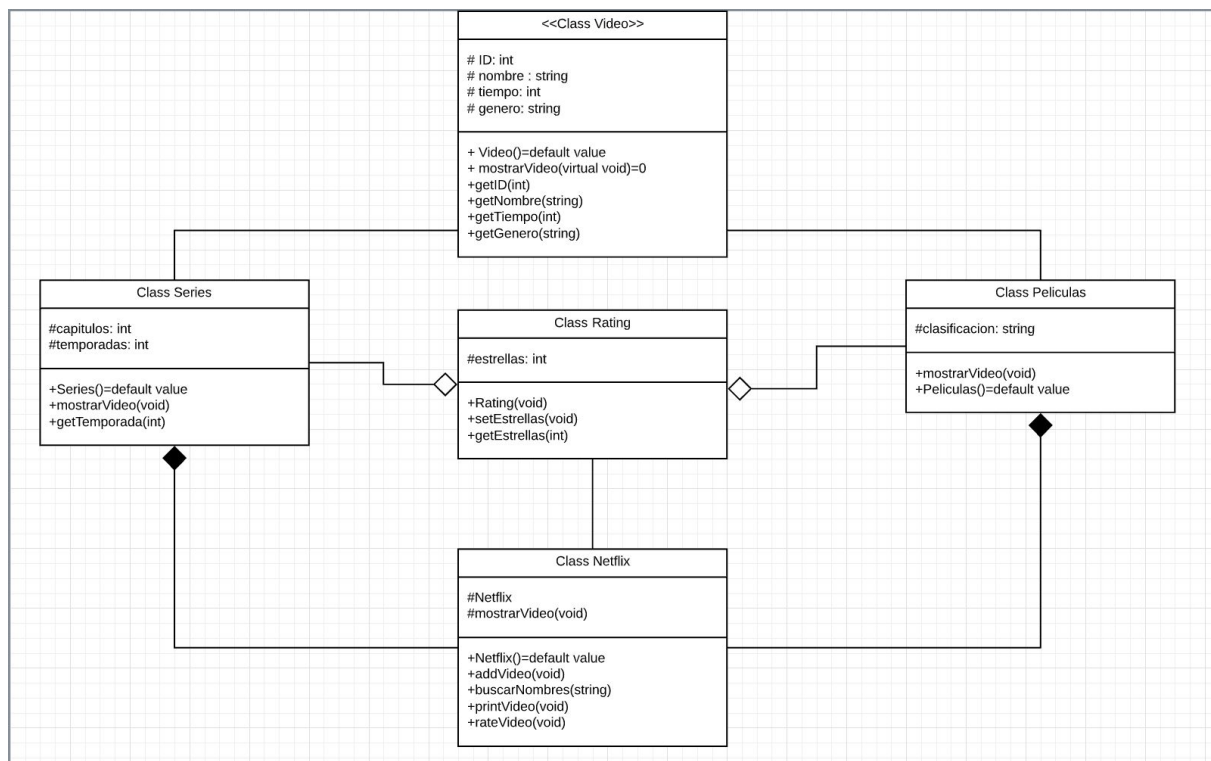
Todo lo que sale en tu inicio es escogido gracias a ciertos algoritmos pero lo que también influye en lo que vas a decidir ver es la calificación que tiene el título. Cada vídeo se puede calificar con cualquier usuario y se basa en una calificación de 0 a 5 estrellas.

Para poder aplicar y reflejar esto en la simulación para la situación problema de la materia nos basamos en describir dos tipos de vídeos : Series y Películas.

El usuario va a decidir que tipo de video quiere calificar y por eso cada Serie y Película debe poder ser calificada por el usuario en una escala del 1 al 5. El sistema también debe de desplegar las opciones que tiene el usuario para calificar y devolver el título que se calificó y la calificación que se le otorgó.

Diagrama de clases de tipo UML

Los diagramas UML sirven para representar de manera gráfica , un modelo para poder describir tu “hipótesis” de cómo va a estar conformado tu sistema. Se pueden representar las clases con sus atributos y métodos (-privados,+públicos o #protegidos) y el tipo de relación que tendrán (herencia , agregación , composición o asociación).



Ejemplo de ejecución

```
Main.cpp      Netflix.h      Peliculas.h      Rating.h      Series.h      Video.h      a.out
(base) MacBook-de-Caro-2:SituacionProblema caroortega$ g++ Main.cpp
(base) MacBook-de-Caro-2:SituacionProblema caroortega$ ./a.out
Desea calificar una P(película) o S(serie):
P
Tiene estas opciones :
Nombre de película: Avengers:Los vengadores de Marvel  Género: Acción  Clasificación: PG-13  Duración: 120 minutos  ID: 1
Nombre de película: Mamma Mia  Género: Musical  Clasificación: PG-13  Duración: 78 minutos  ID: 2
Nombre de película: Becoming  Género: Documental  Clasificación: PG  Duración: 80 minutos  ID: 3
Nombre de película: Son como niños  Género: Comedia  Clasificación: PG-13  Duración: 110 minutos  ID: 4
¿Qué película desea calificar? :
Mamma Mia
¿Qué puntaje le otorga? (Puntaje del 1 al 5) :
5
Calificaste Mamma con 5/5 estrellas!!
Desea calificar una P(película) o S(serie):
```

```
5
Calificaste Mamma con 5/5 estrellas!!
Desea calificar una P(película) o S(serie):
S
Para salir escribir una letra diferente a P ó S
Tiene estas opciones :
Nombre de serie: Friends  Género de serie: Comedia  Temporadas: 10  Capítulos: 236  Duración: 24 minutos  ID: 11
Nombre de serie: La casa de papel  Género de serie: Thriller  Temporadas: 4  Capítulos: 38  Duración: 45 minutos  ID: 22
Nombre de serie: Gossip Girl  Género de serie: Drama adolescente  Temporadas: 6  Capítulos: 121  Duración: 42 minutos  ID: 33
Nombre de serie: Cake Boss  Género de serie: Telerrealidad  Temporadas: 10  Capítulos: 236  Duración: 35 minutos  ID: 44
¿Qué serie desea calificar? :
Friends
¿Qué puntaje le otorga? (Puntaje del 1 al 5) :
3
Calificaste Friends con 3/5 estrellas!!
Desea calificar una P(película) o S(serie):
```

Partes del proyecto

¿Por qué se optó por esta solución y no por otras?

Quería empezar a tener un pensamiento más general a la hora de hacer programas así que tuve que empezar con decidir qué tipo de clases quería y cómo iba a hacerlas simples pero que funcionaran con métodos generales.

Lo primero que hice fue determinar que necesitaba un main y dos de las clases más importantes para este proyecto : Series y Películas. Al empezar a llenar todos sus atributos , reflexioné que las series y películas comparten muchas características y entonces pensé que era la oportunidad perfecta para hacer un programa más general . Decidí hacer una clase video con los atributos que comparten las series y películas porque al final los dos son videos , de esta forma si algún día quisiera agregar otra clase llamada “Videos de youtube” también compartiría las características y podría usar los métodos sin cambiar el código existente. Uno de los objetivos también era poder calificar un título por eso decidí hacer una clase rating que solo se dedique a calificar . Para poder unir los métodos generales de las clases hice una clase llamada netflix la que se encarga de llamar a series y películas y hacerlas funcionar igual con los mismos métodos(los cuales llamó en el main). El main me permitió dar los toques finales de la manera en la que el programa iba a funcionar.

Identificación de casos que harían que el proyecto deje de funcionar.

Si las clases no están bien relacionadas pueden haber problemas difíciles de identificar como Diamond Ring (que mi programa tenía).El uso de los apuntadores puede ser confuso porque podrías apartar memoria que en realidad no vas a usar o llamar la variable que no quieres. Un error que me pasó fue confundir la manera correcta de poner “!=” con “=” y cosas pequeñas

como esta pueden causar que el programa no compile y como consecuencia tampoco ejecute.

También se me presentó un ciclo infinito , el cual sucede porque a veces C++ se confunde con el cin y pasa a un salto de línea.

Conclusión

Me di cuenta que al realizar este proyecto de alguna forma me superé y me sorprendí a mí misma de haber hecho este programa desde cero. Cuando empezaron las 5 semanas de esta materia no creí que el objetivo de poder llegar a hacer un programa completo de C++ fuera algo posible , ya que me sentí muy abrumada e intimidada por la cantidad de temas a cubrir en tan poco tiempo y sin la certeza de que fuera a aprender a hacerlo o aprender los conceptos. Mi maestro me ayudó muchísimo a empezar a confiar en que aunque aprenda un poco más lento puedo ser capaz de entender los temas y ponerlos en práctica.

Aunque podría mejorar la estructura en la que mis clases se relacionan para que no pase el problema de Diamond Ring y confundirme de más , pude empezar a identificar el pensamiento general que se debe de tener al crear un programa en C++ , ya que el objetivo es que tus clases sean lo más simple posibles y que los métodos sean generales y sirvan para las clases que tienes y para futuras clases sin necesidad de cambiar todo tu código.

De las cosas más importantes de las que me di cuenta fue que pedir ayuda (y más en materias como esta) es una opción que te garantiza avanzar porque las personas que saben más que tú te ayudan a entender y te impulsan a que intentes programar aunque te equivoques porque al final si es algo con lo que no puedes , ellos sí te pueden ayudar.

Referencias

N.A.. (2013). Diagramas UML. 11 de junio de 2020, de Teatro Abadia Sitio web:

https://www.teatroabadia.com/es/uploads/documentos/iagramas_del_uml.pdf