

Proyecto Integrador

José Ángel García Gómez

A01745865

Erika Marlene García Sánchez

A01745158

Junio 2021

Profesor: Antonio Mejia

Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Estado de México

Escuela de Ingeniería y Ciencias

Programación orientada a objetos (Gpo 303)

Índice de contenido

Portada	1
Introducción	2
Diagrama de clases UML	3
Ejemplo de ejecución	10
Conclusiones individuales	31
Referencias	32

Introducción

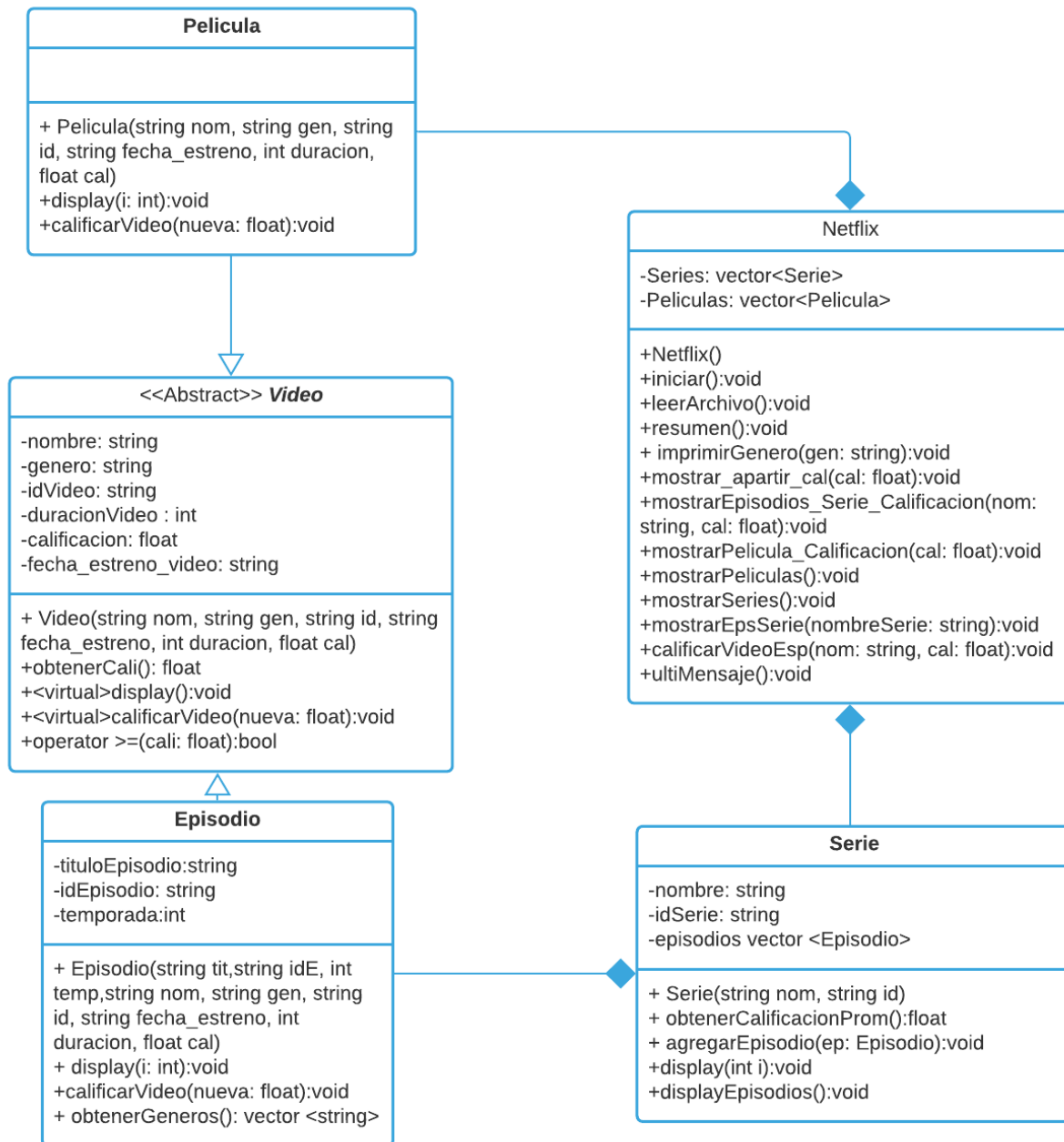
Los avances tecnológicos han cambiado la manera en la que consumimos el entretenimiento, es por eso que se han creado las plataformas de streaming donde los usuarios pueden acceder a películas, series y documentales a la hora que deseen a través de sus televisiones o con sus diferentes dispositivos digitales, siempre y cuando tengan acceso a internet o hayan descargado el contenido. Las plataformas de streaming no solo le dan a sus usuarios un gran contenido para ver sino también les facilita el escoger una película o serie que sea de su agrado ya que a través de cómo califican el contenido que ven le es más fácil a estas plataformas el poder ofrecerle a sus usuarios contenido que sea de su agrado.

En esta situación problema debemos diseñar un modelo de streaming que trabaje con dos diferentes tipos de videos: películas y series. Además estas cuentan con un ID, nombre, duración, género, calificación y fecha de estreno que las diferencia. Las series a su vez cuentan con episodios que tienen su propio nombre, ID, temporada y número de episodio.

Con este sistema seremos capaces de mostrar los videos en general, tanto películas como series, con sus respectivas calificaciones o por género y mostrar su información, mostrar los episodios de una serie en específico con sus calificaciones y mostrar las películas con sus calificaciones.

A través de todo lo aprendido a lo largo de este curso es que pudimos solucionar esta situación problema, hicimos uso de nuestro conocimiento en herencia, polimorfismo y sobrecarga de operadores.

Diagrama de clases UML



Nuestro diagrama UML está compuesto por una clase abstracta llamada **Video** la cual se compone de atributos tipo string *nombre* la cual es el nombre de las películas y episodios, *genero* la cual tiene géneros desde Action hasta War, *idVideo* el cual es un id diferente para cada video, *fecha_estreno_video* la cual tiene las fechas de estreno en el formato 00/00/0000; de la variable tipo int *duracionVideo* que contiene la duración en minutos y de la variable tipo float *calificacion* la cual contiene calificaciones del 1 al 10. Esta función tiene un constructor donde se le pasan como parámetros los atributos anteriormente mencionados, también tiene un método llamado *obtenerCali* que regresa un flotante con la calificación del archivo en escala del 1 al 5, también cuenta con dos métodos virtuales que no regresan nada llamados *display* el cual muestra la información de las películas y episodios con todos los métodos que se mencionaron, *calificarVideo* al cual se le pasa como parámetro una variable llamada *nueva* la cual es de tipo flotante este método se encarga de sumarle la nueva calificación que dio el usuario a la calificación que ya tenía con anterioridad para sacarle un promedio y obtener una calificación nueva y el método *operator >=* el cual recibe la variable *cali* que es la calificación de tipo flotante y regresa un booleano, esta función se encarga de checar si la calificación es mayor o igual que la calificación que solicitó el usuario regresando true o false.

Tenemos una clase llamada **Pelicula** la cual no tiene atributos ya que los hereda de la clase **Video** de forma pública. Esta clase tiene un constructor al cual se le pasan los mismos atributos que a la clase **Video**, también tiene un método llamado *display* a la que se le pasa un parámetro de tipo int llamado *i* y no regresa nada, este método se encarga de usar la función *display* de la clase **Video** por lo que regresa la información que se mencionó anteriormente con un subtítulo en la parte de arriba que dice “-----Pelicula-----”, por último tiene el método llamado *calificarVideo* que usa la función del mismo nombre de la clase **Video** por lo que obtiene una nueva calificación para la película y esta la imprime diciendo el nombre de la película y la nueva calificación (igual en escala del 1 al 5).

Tenemos la clase **Episodio** que hereda de igual manera de la clase **Video** de manera pública, además cuenta con los atributos de tipo string llamados *tituloEpisodio* y *idEpisodio* así como con un atributo de tipo int llamado *temporada*. Esta función tiene un constructor al que se le pasan como parámetros sus propios atributos más los atributos de **Video**, tiene un método llamado *obtenerGeneros* el cual regresa un vector lleno de strings, en esta función pusimos un

ciclo while para separar todos los géneros que tienen las películas y episodios a partir de la coma (,) para guardarlos en el vector de strings *generos*. Después tenemos el método *display* en el cual se muestra el subtítulo “-----Episodio ----” junto con su respectivo número, abajo el nombre del episodio, su ID y el número de temporada. Nuestro último método es *calificarVideo* donde se le pasa un parámetro float llamado *nueva* la cual es la nueva calificación que le quiere dar el usuario desplegando “El episodio *tituloEpisodio* de la serie *Video::getNombre* cambió de calificación a: *Video::obtenerCali()*”.

La clase **Serie** compone a la clase **Episodio**, es decir Serie tiene Episodio. Tiene atributos de tipo string llamados *nombre*, *idSerie* y un vector llamado *episodios*. El constructor de **Serie** tiene como parámetros *nom* y *id*, ambos son strings. Después tenemos como método *obtenerCalificacionProm* donde de nuestro vector *episodios* sacamos cada calificación y las vamos sumando para después dividirlo entre el número de episodios y regresar un flotante con la calificación final de la serie. Posteriormente tenemos el método *agregarEpisodio* donde al vector *episodios* le agregamos objetos de la clase **Episodio**. Aquí también tenemos el método *display* que muestra el subtítulo “----Serie----” junto con su respectivo número de serie, abajo se muestra el nombre de la serie, su ID, el número de capítulos que tiene y la calificación promedio de la serie. Por último tenemos el método *displayEpisodios* que muestra el mensaje “Los episodios de la serie *nombre* son: ” y tenemos un ciclo for que va desde 0 y sigue mientras sea menor que el tamaño del vector *episodios* este vector se encarga de desplegar los episodios que tiene la serie llamando a la función *display*.

Nuestra última clase es **Netflix** la cual está en composición con las clases **Pelicula** y **Serie**, es decir Netflix tiene películas y series. Sus atributos son dos vectores, el primero está lleno de objetos de la clase **Serie** y el segundo objetos de la clase **Pelicula**. Tiene un constructor vacío, un método llamado *leerArchivo* el cual es llamado desde main y tiene como variables locales *id*, *nombre*, *duracion*, *genero*, *calificacion*, *fecha_estreno*, *nombreEp*, *idEp*, *temporada*, *linea_numEp*, *comilla_numEp* y *numEp*; esta función se encarga de leer el archivo “BasePelículas4.csv” separándolo cada que encuentra un punto y coma (;), también si en la variable *numEp* encuentra que está vacía entonces creará un objeto de la clase **Pelicula** pasándole como parámetro las variables *nombre*, *genero*, *id*, *fecha_estreno*, *duracion_i* y *calificacion_i*, posteriormente procederá a agregar estos objetos en el vector *Peliculas*, si la

variable *numEp* no se encuentra vacía entonces creará un objeto de la clase **Episodio** pasándole como parámetro las variables *nombreEp*, *idEp*, *temp_i*, *nombre*, *genero*, *id*, *fecha_estreno*, *duracion_i* y *calificacion_i*; después se encuentra un condicional donde si el tamaño del vector *Series* es mayor a 0 y coincide el nombre con el que se le pasa como parámetro entonces a esa serie en específico por medio del método *agregarEpisodio* de la clase **Serie** es que se agregará este objeto; si no coincide el nombre entonces se creará un objeto de la clase **Serie** pasándole como parámetro el *nombre* y el *id*, este objeto por medio de la función *agregarEpisodio* de la clase **Serie** y con un *push_back* lo agrega al vector *Series*; si el tamaño del vector *Series* no es mayor a 0 entonces de igual manera hará lo del paso anterior. Finalmente se cierra el archivo.

También tiene un método llamado *iniciar* el cual no regresa nada, este método tiene una variable local llamada *op* la cual contiene un número entero de la opción que elija el usuario, aquí tenemos nuestro menú donde al principio mostramos un mensaje de bienvenida que es “-- Bienvenido a su servicio de Streaming ---”, después despliega las 7 opciones que tenemos que son las siguientes:

1. - Mostrar resumen de la lectura de la base de datos.
2. - Mostrar los videos en general de un cierto género.
3. - Mostrar los videos en general a partir de una cierta calificación.
4. - Mostrar los episodios de una determinada serie con una calificación determinada.
5. - Mostrar las películas a partir de cierta calificación.
6. - Calificar un video en específico.
7. - Terminar

Y al final muestra el siguiente mensaje: “Por favor ingrese el número de la opción que desee desplegar: ”.

Si eligen la primer opción primero por medio de un bloque de try donde prueba nuestro método *resumen* que tiene dos variables locales llamadas, *contEp* y *conTot*. Por medio de un ciclo for de 0 y mientras sea menor al tamaño del vector *Series* a la variable *contEp* se le sumará ella misma más el tamaño del vector *Episodios*. Posteriormente la variable *conTot* es la suma de *contEp* más el tamaño del vector *Peliculas* para que de esta manera sepamos el número de videos que tiene.

Aquí mostramos el número de vídeos, el número de películas y episodios que tiene la plataforma de streaming. Luego despliega un mini menú con las dos opciones siguientes:

1. - Mostrar las películas disponibles.
2. - Mostrar las series disponibles.

Además muestra la instrucción de que ingrese la opción que quiera desplegar, el número que el usuario ingrese se guardará en la variable local *opres*. Si el usuario ingresa un número diferente de 1 y 2 entonces arrojará el siguiente mensaje “ - Error: opción no válida -”. Si ingresa 1 entonces irá a su método *mostrarPelículas* en donde por medio de un ciclo for usa la función *display* de la clase **Pelicula** donde de su vector *Peliculas* irá mostrando todas las películas que tiene la plataforma. Si elige 2 entonces irá a su método *mostrarSeries* donde por medio de un ciclo for usa la función *display* de la clase **Serie** donde de su vector *Series* irá mostrando todas las series que tiene la plataforma. Se despliega un mini menú con las dos opciones siguientes:

1. - Mostrar los episodios de una serie en específico.
2. - Regresar al menú principal.

Además muestra la instrucción de que ingrese la opción deseada, el número que el usuario ingrese se guardará en la variable local *opEp*. Si el usuario ingresa un número diferente a 1 y 2 entonces arrojará el siguiente mensaje “ - Error: opción no válida -”. Si ingresa 1 entonces pedirá al usuario que ingrese el nombre de la serie que quiere ver y lo guardará en su variable local *nombreSerie*, por medio de *.ignore* y *getline* logamos leer todo lo que escribe el usuario incluyendo los espacios en blanco posteriormente por medio de un bloque try nos dirigimos al método *mostrarEpsSerie* pasándole como parámetro el nombre de la serie que se requiere, en la función *mostrarEpsSerie* tenemos una variable local llamada *evalua* la cual es booleano y lo inicializamos como false, posteriormente por medio de un ciclo for y un condicional de que si el parámetro que se le pasó es igual a alguno de los nombres de las series que están en el vector *Series* entonces convertirá nuestra variable *evalua* a true y por medio del método *displayEpisodios* mostrará todos los episodios de la serie que solicitó el usuario, si *evalua* es igual a false entonces significa que la serie que busca el usuario no se encuentra en la base de datos o lo escribió mal así que lanza un error “- Error: Serie no válida-”. Si del mini menú que se mencionó anteriormente ingresa 2 entonces le vuelve a mostrar el menú general.

Si en el menú general el usuario ingresa la opción 2 entonces le pedirá que ingrese el género (En ingles y con la primer letra en mayúscula) que quiera ver, el género se guarda en la variable local *gene*, después en un bloque try prueba el método *imprimirGenero* pasándole como parámetro la variable *gene*. En la función *imprimirGenero* por medio de ciclos y condicionales mostramos las series y películas del género que eligió el usuario, si el género que ingresó no se encuentra en nuestra base de datos entonces se arroja el error “- Error: Genero no válido-”. Posteriormente por medio de un bloque try se prueba la función *ultiMensaje* la cual muestra un mini menú con las dos opciones siguientes:

3. - Volver a mostrar el menú.
4. - Salir.

Si el usuario ingresa una opción diferente a 1 y 2 entonces arroja el siguiente error “- Error: Opcion no valida, voy a terminar el programa. -”. Si ingresa 1 entonces vuelve a mostrar el menú general y si ingresa el número 2 muestra el siguiente mensaje de despedida “Gracias por preferir nuestro servicio de Streaming...”.

Si en el menú general el usuario ingresa la opción 3 le pedirá al usuario que ingrese la calificación de la que desea ver videos en una escala del 1 al 5 aquí se va a la función *mostrar_apartir_cal* pasándole como parámetro la calificación que ponga el usuario, en esa función por medio de ciclos for y condiciones se despliega la información de las películas y episodios cuya calificación es igual o mayor a la que ingresa el usuario y si no encuentra ningún video con la calificación que da el usuario entonces se despliega un error.

Si en el menú general el usuario ingresa la opción 4 le pedirá que ingrese el nombre de la serie a desplegar y posteriormente le pedirá que ingrese la calificación de la cual quiere ver los episodios, posteriormente manda esos parámetros a la función *mostrarEpisodios_Serie_Calificacion* en donde por medio de for's y condicionales busca si el nombre de la serie se encuentra en el catálogo de la plataforma, de igual manera busca si cuenta con episodios mayores o igual a la calificación solicitada, si cumple con esas condiciones entonces despliega la información de los episodios, de no ser así arroja mensajes de error.

Si en el menú general el usuario ingresa la opción 5 le pedirá que ingrese la calificación de las películas que quiere ver y por medio del método *mostrarPelicula_Calificacion* y por ciclos for y condicionales es que checa si la plataforma cuenta con películas con esa calificación o con alguna mayor para desplegar su información y de no ser así arroja un mensaje de error.

Si en el menú general el usuario ingresa la opción 6 le pedirá que ingrese el nombre del video que quiere calificar y posteriormente que ingrese la calificación que le quiere asignar (en una escala de 1 a 5), por medio de su método *calificarVideoEsp* y con ciclos for y condicionales es que checa si la plataforma cuenta con el nombre ingresado en sus películas o episodios, de ser así a través de sus funciones *calificarVideo*, (tanto de la clase **Pelicula** como de la clase **Episodio**) es que promedia la calificación que ya tenía el video con la nueva que ingresa el usuario para mostrarle al usuario la nueva calificación y si lo que ingresa el usuario no cumple con las especificaciones mencionadas anteriormente entonces arrojará un mensaje de error.

Si en el menú general el usuario ingresa la última opción, que es 7, entonces mostrará el siguiente mensaje de despedida “Gracias por preferir nuestro servicio de Streaming...”.

Si en el menú general el usuario ingresa cualquier cosa diferente a las opciones dadas entonces se mostrará el siguiente mensaje “- Error: opción inválida, voy a terminar el programa. -”.

En el archivo main se crea un objeto de la clase **Netflix**, después se hace uso de la función *leerArchivo* y por último de la función *iniciar*.

Ejemplo de ejecución

```
----- Bienvenido a su servicio de Streaming -----

1 - Mostrar resumen de la lectura de la base de datos.
2 - Mostrar los videos en general de un cierto género.
3 - Mostrar los videos en general a partir de una cierta calificacion.
4 - Mostrar los episodios de una determinada serie con una calificación determinada.
5 - Mostrar las peliculas a partir de cierta calificacion.
6 - Calificar un video en específico.

7 - Terminar

-----

Por favor ingrese el número de la opción que desee desplegar: 1
----- Resumen -----
Se leyeron 38 videos de los cuales: 17 son peliculas y 21 son episodios que forman parte de 5 series.
-----

1 - Mostrar las peliculas disponibles.
2 - Mostrar las series disponibles.

-----

Ingresa la opcion que quiere desplegar: 1
```

En la imagen se muestra el menú con 7 opciones y la instrucción de que ingrese el número de la opción que quiere seleccionar. Posteriormente se ve el resumen que dice el número de videos, películas, episodios y series que tiene la base de datos. Después muestra 2 opciones donde la primera es mostrar las películas disponibles y la segunda es mostrar las series disponibles, en este caso se escogió la primera opción.

- - - - - Pelicula 1 - - - -

Nombre: Jurassic Park
Generos: Action, Adventure, Sci-Fi, Suspense
Id: tt0107290
Duracion: 127
Calificacion: 4.05/5
Fecha de estreno: 6/11/1993

- - - - - Pelicula 2 - - - -

Nombre: Star Trek
Generos: Action, Adventure, Sci-Fi
Id: tt0796366
Duracion: 127
Calificacion: 3.95/5
Fecha de estreno: 5/8/2009

- - - - - Pelicula 3 - - - -

Nombre: Mulan
Generos: Animation, Adventure, Family, Fantasy, Musical, War
Id: tt0120762
Duracion: 88
Calificacion: 3.8/5
Fecha de estreno: 6/19/1998

- - - - - Pelicula 4 - - - -

Nombre: The Lion King 2: Simba's Pride
Generos: Animation, Adventure, Drama, Family, Musical, Romance
Id: tt0120131
Duracion: 81
Calificacion: 3.25/5
Fecha de estreno: 10/27/1998

- - - - - Pelicula 5 - - - -

Nombre: Superman
Generos: Action, Adventure, Drama, Sci-Fi
Id: tt0078346
Duracion: 143
Calificacion: 3.65/5
Fecha de estreno: 12/15/1978

En esta captura de pantalla se muestran las películas disponibles con toda su información.

- - - - - Pelicula 6 - - - - -

Nombre: Batman
Generos: Action, Adventure
Id: tt0096895
Duracion: 126
Calificacion: 3.75/5
Fecha de estreno: 6/23/1989

- - - - - Pelicula 7 - - - - -

Nombre: The Avengers
Generos: Action, Adventure, Sci-Fi
Id: tt0848228
Duracion: 143
Calificacion: 4/5
Fecha de estreno: 5/4/2012

- - - - - Pelicula 8 - - - - -

Nombre: End Game
Generos: Action, Crime, Drama, Mystery, Thriller
Id: tt0415778
Duracion: 93
Calificacion: 2.55/5
Fecha de estreno: 7/21/2006

- - - - - Pelicula 9 - - - - -

Nombre: Django Unchained
Generos: Drama, Western
Id: tt1853728
Duracion: 165
Calificacion: 4.2/5
Fecha de estreno: 12/25/2012

- - - - - Pelicula 10 - - - - -

Nombre: Love Rosie
Generos: Comedy, Romance
Id: tt1638002
Duracion: 102
Calificacion: 3.6/5
Fecha de estreno: 10/22/2014

En esta captura de pantalla se muestran las películas disponibles con toda su información.

- - - - - Pelicula 11 - - - -

Nombre: Back to the Future
Generos: Adventure, Comedy, Sci-Fi
Id: tt0088763
Duracion: 116
Calificacion: 4.25/5
Fecha de estreno: 7/3/1985

- - - - - Pelicula 12 - - - -

Nombre: The Greatest Showman
Generos: Biography, Drama, Musical
Id: tt1485796
Duracion: 105
Calificacion: 3.8/5
Fecha de estreno: 12/20/2017

- - - - - Pelicula 13 - - - -

Nombre: The Lego Batman Movie
Generos: Animation, Action, Comedy, Family
Id: tt4116284
Duracion: 104
Calificacion: 3.65/5
Fecha de estreno: 2/10/2017

- - - - - Pelicula 14 - - - -

Nombre: Ad Astra
Generos: Adventure, Drama, Mystery, Sci-Fi, Thriller
Id: tt2935510
Duracion: 123
Calificacion: 3.3/5
Fecha de estreno: 9/20/2019

- - - - - Pelicula 15 - - - -

Nombre: 1917
Generos: Drama, War
Id: tt8579674
Duracion: 119
Calificacion: 4.15/5
Fecha de estreno: 1/10/2020

En esta captura de pantalla se muestran las películas disponibles con toda su información.

```
- - - - - Pelicula 14 - - - -  
Nombre: Ad Astra  
Generos: Adventure, Drama, Mystery, Sci-Fi, Thriller  
Id: tt2935510  
Duracion: 123  
Calificacion: 3.3/5  
Fecha de estreno: 9/20/2019  
  
- - - - - Pelicula 15 - - - -  
Nombre: 1917  
Generos: Drama, War  
Id: tt8579674  
Duracion: 119  
Calificacion: 4.15/5  
Fecha de estreno: 1/10/2020  
  
- - - - - Pelicula 16 - - - -  
Nombre: Jojo Rabbit  
Generos: Comedy, Drama, War  
Id: tt2584384  
Duracion: 108  
Calificacion: 3.95/5  
Fecha de estreno: 11/8/2019  
  
- - - - - Pelicula 17 - - - -  
Nombre: The Hobbit: An Unexpected Journey  
Generos: Adventure, Family, Fantasy  
Id: tt0903624  
Duracion: 169  
Calificacion: 3.9/5  
Fecha de estreno: 12/14/2012  
  
- - - - - Elija una opcion - - - - -  
  
1 - Volver a mostrar el menu.  
2 - Salir.  
  
- - - - -
```

En esta captura de pantalla se muestran las películas disponibles con toda su información. Abajo se encuentran dos opciones, la primera es volver a mostrar el menú y la segunda es salir. Aquí se eligió la primera opción.

```

1 - Mostrar resumen de la lectura de la base de datos.
2 - Mostrar los videos en general de un cierto género.
3 - Mostrar los videos en general a partir de una cierta calificacion.
4 - Mostrar los episodios de una determinada serie con una calificación determinada.
5 - Mostrar las peliculas a partir de cierta calificacion.
6 - Calificar un video en específico.

7 - Terminar

-----

Por favor ingrese el número de la opción que desee desplegar: 1

----- Resumen -----
Se leyeron 38 videos de los cuales: 17 son peliculas y 21 son episodios que forman parte de 5 series.
-----

1 - Mostrar las peliculas disponibles.
2 - Mostrar las series disponibles.

-----

Ingresa la opcion que quiere desplegar: 3
-----
- - - - Error: opcion no válida - - - -
- - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.
2 - Salir.

```

Después se eligió la opción 1 y después se colocó el número 3 para ver qué pasa cuando se ingresa una opción no válida, el programa arroja un mensaje de error y pide elegir entre 1. Volver a mostrar el menú y 2. Salir.

```

-----
4
- - - - Error: Opcion no valida, voy a terminar el programa. - - - -

```

Se puso el número 4 para ver qué pasa cuando se pone una opción que no es válida y podemos ver que arroja un mensaje de error y se termina el programa.

Ingresa la opcion que quiere desplegar: 2

- - - - -

- - - Serie 1 - - -

Nombre de la Serie: Game of Thrones
Id Serie: tt0944947
Capitulos: 9
Calificacion Promedio de la Serie: 4.52222

- - - Serie 2 - - -

Nombre de la Serie: Friends
Id Serie: tt0108778
Capitulos: 3
Calificacion Promedio de la Serie: 4.11667

- - - Serie 3 - - -

Nombre de la Serie: Modern Family
Id Serie: tt1442437
Capitulos: 3
Calificacion Promedio de la Serie: 4.1

- - - Serie 4 - - -

Nombre de la Serie: The 100
Id Serie: tt2661044
Capitulos: 3
Calificacion Promedio de la Serie: 3.81667

- - - Serie 5 - - -

Nombre de la Serie: The Simpsons
Id Serie: tt0096697
Capitulos: 3
Calificacion Promedio de la Serie: 3.8

1 - Mostrar los episodios de una serie en específico.

2 - Regresar al menu principal

Después se elige la opción 2 del menú de la opción 1 y nos muestra todas las series que tiene la plataforma, luego aparecen dos opciones, la primera es mostrar los episodios de una serie en específico y la segunda es regresar al menú principal.


```
- - - - -
Ingresar la opcion deseada: 1
- - - - -

Ingrese el nombre de la serie: Modern Family
- - - - -

Los episodios de la serie Modern Family son:

- - - - Episodio 1 - - - -

Nombre del Episodio: Piloto
Id Episodio: tt1444504
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4.2/5
Fecha de estreno: 9/23/2009

- - - - Episodio 2 - - - -

Nombre del Episodio: The Bicycle Thief
Id Episodio: tt1492032
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4.1/5
Fecha de estreno: 9/23/2009

- - - - Episodio 3 - - - -

Nombre del Episodio: Come Fly with Me
Id Episodio: tt1520618
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4/5
Fecha de estreno: 10/7/2009
```

Ahí se elige la opción 1 y después se ingresa el nombre de la serie Modern Family, el programa arroja los episodios de dicha serie junto con su respectiva información.

```
- - - - - Elij a una opcion - - - - -  
  
1 - Volver a mostrar el menu.  
2 - Salir.  
- - - - -  
1  
----- Bienvenido a su servicio de Streaming -----  
  
1 - Mostrar resumen de la lectura de la base de datos.  
2 - Mostrar los videos en general de un cierto g nero.  
3 - Mostrar los videos en general a partir de una cierta calificacion.  
4 - Mostrar los episodios de una determinada serie con una calificaci n determinada.  
5 - Mostrar las peliculas a partir de cierta calificacion.  
6 - Calificar un video en espec fico.  
  
7 - Terminar
```

Vuelve a mostrar las 2 opciones de volver a mostrar el men  o salir, se escoge la primera y posteriormente muestra el men  principal.

1 - Mostrar los episodios de una serie en específico.

2 - Regresar al menu principal

Ingresar la opcion deseada: 1

Ingrese el nombre de la serie: Lucifer

- - - - - Error: Serie no válida - - - - -

- - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.

2 - Salir.



En esta captura de pantalla se ingresa el nombre de una serie que no está disponible en la plataforma por lo que arroja el error de que la serie ingresada no es válida.

```

Por favor ingrese el número de la opción que desee desplegar: 2
- - - - - Mostrar videos por genero - - - - -
Ingresa el género (En ingles y con la primer letra en mayúscula) que quieres desplegar: War
- - - - - Genero War - - - - -

- - - - - Pelicula 3 - - - - -

Nombre: Mulan
Generos: Animation, Adventure, Family, Fantasy, Musical, War
Id: tt0120762
Duracion: 88
Calificacion: 3.8/5
Fecha de estreno: 6/19/1998

- - - - - Pelicula 15 - - - - -

Nombre: 1917
Generos: Drama, War
Id: tt8579674
Duracion: 119
Calificacion: 4.15/5
Fecha de estreno: 1/10/2020

- - - - - Pelicula 16 - - - - -

Nombre: Jojo Rabbit
Generos: Comedy, Drama, War
Id: tt2584384
Duracion: 108
Calificacion: 3.95/5
Fecha de estreno: 11/8/2019

- - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.
2 - Salir.
- - - - -

```

En esta captura de pantalla se seleccionó la opción 2 del menú general así que posteriormente se ingresa el género del que se quieren ver los videos en general, el programa arroja tanto las películas como los episodios que sean del género solicitado.

```
Por favor ingrese el número de la opción que desee desplegar: 2
- - - - - Mostrar videos por genero - - - - -
Ingresa el género (En ingles y con la primer letra en mayúscula) que quieres desplegar: Comedia
- - - - - Genero Comedia - - - - -

- - - - - Error: Genero no válido- - - - -

- - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.
2 - Salir.
- - - - -
```

En esta captura de pantalla también se seleccionó la opción 2 del menú general, posteriormente se ingresa el género del que se quieren ver los videos en general y aquí ingresamos el género en español por lo que arroja un error ya que el género no es válido.

```

Por favor ingrese el número de la opción que desee desplegar: 3
- - - - - Mostrar videos apartir de una calificacion - - - - -
Ingrese la calificacion(en un rango de 1 a 5): 4.2
- - - - - Calificacion mayor o igual 4.2 - - - - -

- - - - - Pelicula 9 - - - - -
Nombre: Django Unchained
Generos: Drama, Western
Id: tt1853728
Duracion: 165
Calificacion: 4.2/5
Fecha de estreno: 12/25/2012

- - - - - Pelicula 11 - - - - -
Nombre: Back to the Future
Generos: Adventure,Comedy,Sci-Fi
Id: tt0088763
Duracion: 116
Calificacion: 4.25/5
Fecha de estreno: 7/3/1985

- - - - - Episodio 1 - - - - -
Nombre del Episodio: Winter Is Coming
Id Episodio: tt1253648
Temporada: 1
Nombre: Game of Thrones
Generos: Action, Adventure, Drama, Fantasy, Romance
Id: tt0944947
Duracion: 57
Calificacion: 4.5/5
Fecha de estreno: 4/17/2011

- - - - - Episodio 2 - - - - -
Nombre del Episodio: The One Where Monica Gets a RoommateT
Id Episodio: tt0583459
Temporada: 1
Nombre: Friends
Generos: Comedy, Romance
Id: tt0108778

```

En esta captura de pantalla se eligió la opción 3 del menú general, luego se ingresó la calificación y el programa nos mostró tanto las películas como los episodios cuya calificación es mayor o igual a la que ingresamos.

Nombre: Friends
Generos: Comedy, Romance
Id: tt0108778
Duracion: 22
Calificacion: 4.2/5
Fecha de estreno: 9/22/1994

- - - - Episodio 3 - - - -

Nombre del Episodio: Piloto
Id Episodio: tt1444504
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4.2/5
Fecha de estreno: 9/23/2009

- - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.

2 - Salir.

- - - - -

En esta captura de pantalla se muestra la continuación de lo anterior.

Por favor ingrese el número de la opción que desee desplegar: 4

- - - - Mostrar los episodios de una determinada serie con una calificación determinada - - - -

Ingrese el nombre de la serie a desplegar: Modern Family

Ingrese la calificacion que quiere desplegar(en un rango de 1 a 5): 3.4

- - - Estos son los episodios de la serie Modern Family con calificacion mayor o igual 3.4 - - -

- - - - Episodio 1 - - - -

Nombre del Episodio: Piloto
Id Episodio: tt1444504
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4.2/5
Fecha de estreno: 9/23/2009

- - - - Episodio 2 - - - -

Nombre del Episodio: The Bicycle Thief
Id Episodio: tt1492032
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4.1/5
Fecha de estreno: 9/23/2009

- - - - Episodio 3 - - - -

Nombre del Episodio: Come Fly with Me
Id Episodio: tt1520618
Temporada: 1
Nombre: Modern Family
Generos: Comedy, Drama, Romance
Id: tt1442437
Duracion: 22
Calificacion: 4/5
Fecha de estreno: 10/7/2009

En esta captura de pantalla se eligió la opción 4 del menú general, se ingresó la serie Modern Family y como calificación 3.4 por lo que el programa nos mostró todos los episodios cuya calificación es igual o mayor a esa, además de desplegar su información.


```

-----
Por favor ingrese el número de la opción que desee desplegar: 4
- - - - Mostrar los episodios de una determinada serie con una calificación determinada - - - -
Ingrese el nombre de la serie a desplegar: Modern Family
Ingrese la calificacion que quiere desplegar(en un rango de 1 a 5): 6
- - - - Error: Calificacion no valida - - - -

- - - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.
2 - Salir.
-----

```

En esta captura de pantalla se eligió la opción 4 del menú general y de igual manera se ingresó la serie Modern Family pero con una calificación fuera del rango que se puede ingresar por lo que arroja un error de calificación no válida.

```

-----
Por favor ingrese el número de la opción que desee desplegar: 4
- - - - Mostrar los episodios de una determinada serie con una calificación determinada - - - -
Ingrese el nombre de la serie a desplegar: Modern Family
Ingrese la calificacion que quiere desplegar(en un rango de 1 a 5): 5
- - - - Estos son los episodios de la serie Modern Family con calificacion mayor o igual 5 - - - -

- - - - Error: No se encuentran episodios que cumplan con la calificacion ingresada - - - -

- - - - - - Elija una opcion - - - - -

1 - Volver a mostrar el menu.
2 - Salir.
-----

```

En esta corrida del programa se ingresó la opción número 4 del menú general y de igual manera se ingresó la serie Modern Family pero con 5 de calificación así que nos arroja un error de que no se encuentran episodios que cumplan con la calificación ingresada.

Por favor ingrese el número de la opción que desee desplegar: 5

----- Mostrar las peliculas a partir de cierta calificacion -----

Ingrese la calificacion a desplegar(en un rango de 1 a 5): 4.2

----- Estos son las peliculas con calificacion mayor o igual a 4.2 -----

----- Pelicula 9 -----

Nombre: Django Unchained
Generos: Drama, Western
Id: tt1853728
Duracion: 165
Calificacion: 4.2/5
Fecha de estreno: 12/25/2012

----- Pelicula 11 -----

Nombre: Back to the Future
Generos: Adventure,Comedy,Sci-Fi
Id: tt0088763
Duracion: 116
Calificacion: 4.25/5
Fecha de estreno: 7/3/1985

----- Elija una opcion -----

1 - Volver a mostrar el menu.

2 - Salir.

En esta corrida del programa se ingresó la opción número 5 en la cual se va a mostrar las películas que tengan asignadas una calificación mayor o igual a la ingresada, en este caso se le ingreso una calificación válida para que mostrara las disponibles.

```
-----  
  
Por favor ingrese el número de la opción que desee desplegar: 6  
----- Calificar un video en específico -----  
  
Ingresa el nombre del video que quiere calificar: The Avengers  
Ingresa la calificacion que le quiere asignar (en una escala de 1 a 5): 5  
-----  
  
La nueva calificacion de la pelicula The Avengers es: 4.5  
  
----- Elija una opcion -----  
  
1 - Volver a mostrar el menu.  
2 - Salir.  
-----
```

En esta corrida del programa se ingresó la opción número 6 en la cual se va a calificar un video en específico en donde se va a ingresar un video válido y se le asignará una calificación en específico.

```
-----  
Por favor ingrese el número de la opción que desee desplegar: 6  
- - - - Calificar un video en específico - - - -  
  
Ingresa el nombre del video que quiere calificar: Bambi  
Ingresa la calificacion que le quiere asignar (en una escala de 1 a 5): 4.6  
-----  
  
- - - - Error: No se encuentra el video que desea calificar - - - -  
  
- - - - - - - Elija una opcion - - - - - - -  
  
1 - Volver a mostrar el menu.  
2 - Salir.  
-----
```

En esta corrida del programa se ingresó la opción número 6 en la cual se va a calificar un video en específico en donde se va a ingresar un video que sea válido pero se le va a asignar un video no válido por lo que va a arrojar un error en el cual se muestra que lo que se ingreso mal fue el nombre del video ya que no encontró ninguno video con ese nombre.

```
-----  
Por favor ingrese el número de la opción que desee desplegar: 6  
- - - - Calificar un video en específico - - - -  
  
Ingresa el nombre del video que quiere calificar: End Game  
Ingresa la calificacion que le quiere asignar (en una escala de 1 a 5): 10  
-----  
  
- - - - Error: Calificacion ingresada invalida - - - -  
  
- - - - - - - Elija una opcion - - - - - - -  
  
1 - Volver a mostrar el menu.  
2 - Salir.  
  
-----  
  
2  
  
Gracias por preferir nuestro servicio de Streaming...  
-----
```

En esta corrida del programa se ingresó la opción número 6 en la cual se va a calificar un video en específico en donde se va a ingresar un video que sea válido pero se le va a asignar una calificación no válida por lo que va a arrojar un error en el cual se muestra que lo que se ingreso mal fue la calificación.

```
----- Bienvenido a su servicio de Streaming -----

1 - Mostrar resumen de la lectura de la base de datos.
2 - Mostrar los videos en general de un cierto género.
3 - Mostrar los videos en general a partir de una cierta calificacion.
4 - Mostrar los episodios de una determinada serie con una calificación determinada.
5 - Mostrar las peliculas a partir de cierta calificacion.
6 - Calificar un video en específico.

7 - Terminar

- - - - -

Por favor ingrese el número de la opción que desee desplegar: 7

Gracias por preferir nuestro servicio de Streaming...
- - - - -
```

En esta imagen se muestra la corrida del programa en donde se ingresa la opción de terminar el programa desde el menú principal.

```
----- Bienvenido a su servicio de Streaming -----

1 - Mostrar resumen de la lectura de la base de datos.
2 - Mostrar los videos en general de un cierto género.
3 - Mostrar los videos en general a partir de una cierta calificacion.
4 - Mostrar los episodios de una determinada serie con una calificación determinada.
5 - Mostrar las peliculas a partir de cierta calificacion.
6 - Calificar un video en específico.

7 - Terminar

- - - - -

Por favor ingrese el número de la opción que desee desplegar: 10

- - - - - Error: opcion invalida, voy a terminar el programa. - - - -
```

En esta imagen se muestra un ejemplo de la corrida del programa en donde se ingresa una opción inválida y arroja un error.

Conclusiones individuales

Jose Angel Garcia Gomez: Gracias a este proyecto logramos entender de una manera más completa el funcionamiento de las herencias de clases, el polimorfismo y el uso de clases abstractas. Debido a que durante la realización de este reto pudimos emplear estos conceptos enfocado a un funcionamiento en el cual era necesario el uso de estos, fue que logramos identificar la manera más eficiente y el porqué es útil el dominar estos conceptos en la programación orientada a objetos.

Erika Marlene García Sánchez: Este proyecto me gustó mucho ya que está relacionado con una situación de la vida real, hoy en día las plataformas de streaming son un servicio muy rentable con millones de usuarios y poder modelar una plataforma de este tipo con todo el conocimiento adquirido en este bloque fue genial, ya que logré reforzar mi conocimiento sobre diferentes temas como herencia, polimorfismo, clases abstractas, uso de excepciones, etc. Además este proyecto me ayudó a entender de mejor manera como se programa orientado a objetos pensando en situaciones reales.

Referencias

Cabrera, L. (2021). *Leer archivo CSV con C++*. Obtenido de

<https://www.youtube.com/watch?v=bal7cn-YvfU>

Capítulo 10 del libro C++ How to Program, Ninth Edition Harvey Deitel, Paul Deitel

Prentice Hall 2013, ISBN-10: 0-13-337871-3 ISBN-13: 978-0-13-337871-9

Exception Handling in C++ - GeeksforGeeks. (2013, December 23). Retrieved June 10,

2021, from GeeksforGeeks website:

<https://www.geeksforgeeks.org/exception-handling-c/>

Inheritance in C++ - GeeksforGeeks. (2017, May 25). Retrieved June 10, 2021, from

GeeksforGeeks website: <https://www.geeksforgeeks.org/inheritance-in-c/>