



Instituto Tecnológico y de Estudios Superiores de Monterrey

Escuela de Ingeniería y Ciencias

*Proyecto Integrador*

Autor

Serna Mendoza Ximena - A01749870 - Campus Estado de México

Profesor Tutor - Dr. Roberto Martínez Román

Programación Orientado a Objetos

Sábado 17 de junio de 2023

## ÍNDICE

Introducción.....	3
Diagrama de clases.....	4
Ejemplo de ejecución.....	5 y 6
Identificación de casos.....	7
Conclusión Personal.....	8
Referencias consultadas.....	9

## Introducción

En los últimos años, los servicios de streaming de video bajo demanda han experimentado un notable crecimiento. Plataformas como Netflix, Disney Plus y HBO Max han transformado la forma en que consumimos entretenimiento, ofreciendo acceso ilimitado a películas y series en cualquier momento y lugar.

Ante esta realidad, surge la necesidad de desarrollar un sistema de consulta especializado para proveedores de streaming. Nuestro proyecto busca crear una versión simplificada de esta herramienta, enfocándonos en películas y episodios de series.

El sistema permitirá cargar datos en formato CSV, mostrar videos según calificación y género especificados por el usuario, buscar episodios de una serie en particular y listar películas con una calificación superior a un valor determinado.

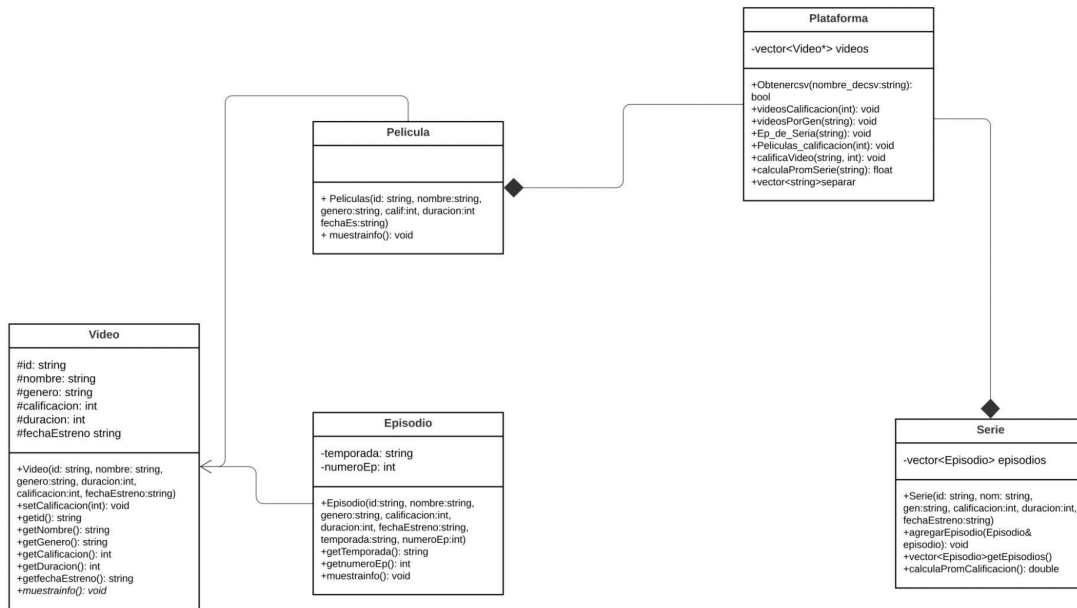
Además, los usuarios podrán calificar videos y calcular el promedio de calificaciones de una serie.

Con este proyecto, pretendemos adquirir experiencia en el diseño y desarrollo de sistemas de consulta para servicios de streaming, mejorando la experiencia del usuario y facilitando la gestión del contenido audiovisual.

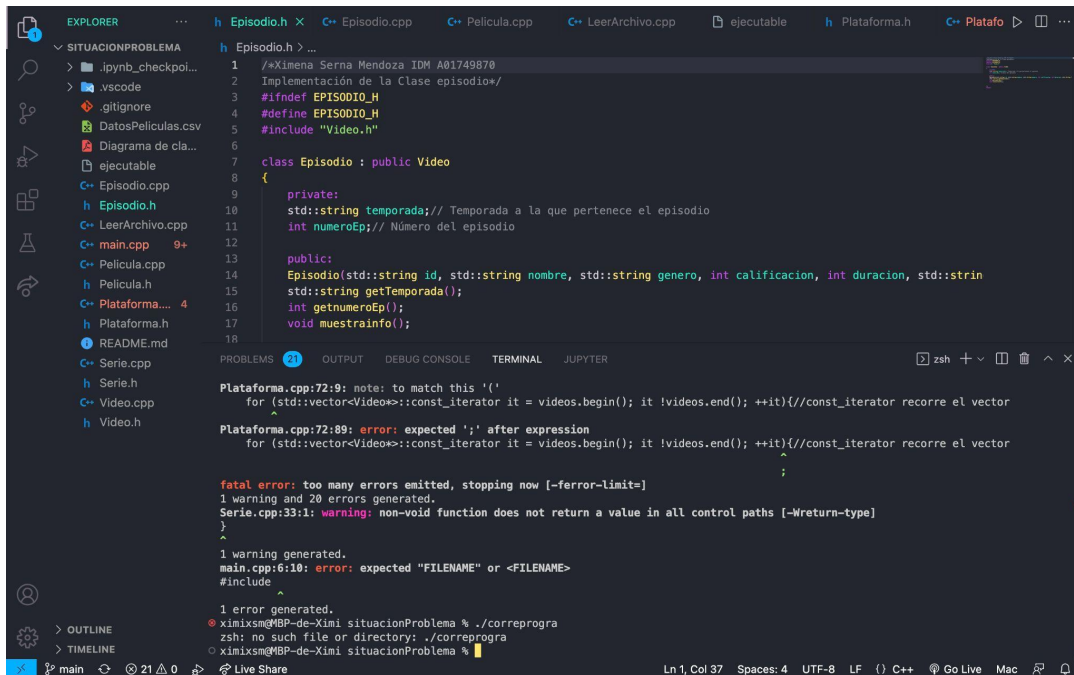
El informe detalla el proceso de desarrollo, incluyendo el diseño del programa, la implementación de funcionalidades, la manipulación de archivos y las pruebas realizadas.

Buscamos ofrecer una solución eficiente y efectiva para la consulta de contenido audiovisual en servicios de streaming.

## Diagrama de clases



## Ejecución del programa



The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying a project structure. The main editor window shows the `Episodio.h` file with the following code:

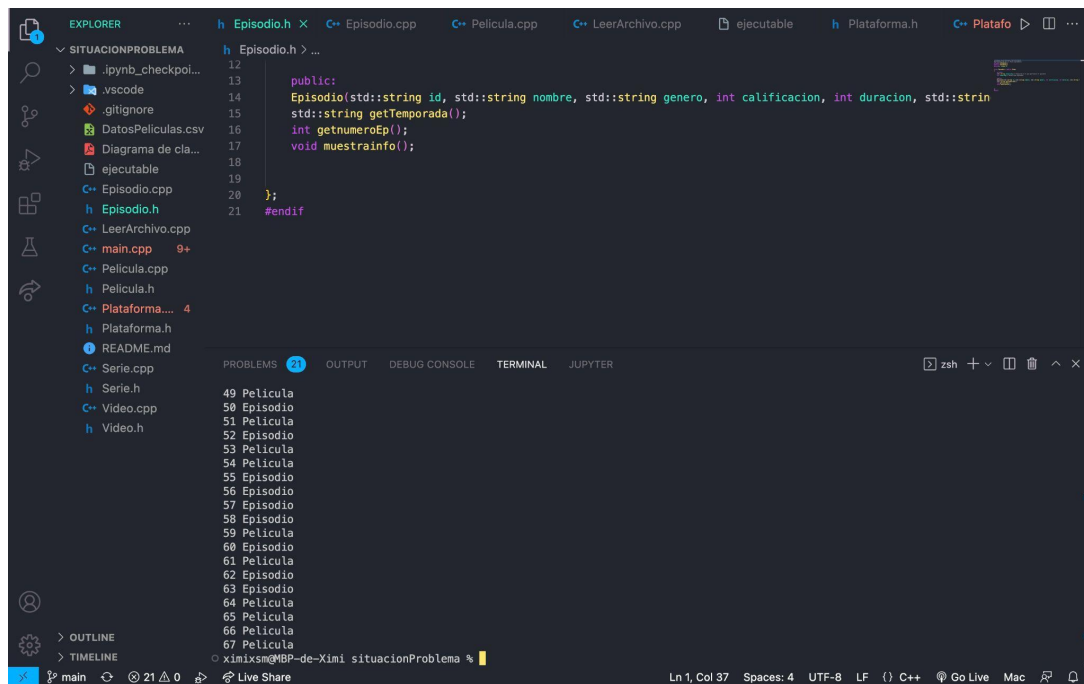
```
1 /*Ximena Serna Mendoza IDM A01749870
2 Implementación de la Clase episodio*/
3 #ifndef EPISODIO_H
4 #define EPISODIO_H
5 #include "Video.h"
6
7 class Episodio : public Video
8 {
9 private:
10     std::string temporada; // Temporada a la que pertenece el episodio
11     int numeroEp; // Número del episodio
12
13 public:
14     Episodio(std::string id, std::string nombre, std::string genero, int calificacion, int duracion, std::string temporada);
15     std::string getTemporada();
16     int getnumeroEp();
17     void muestrainfo();
18 }
```

The bottom panel shows the TERMINAL output with the following errors and warnings:

```
Plataforma.cpp:72:9: note: to match this '('
for (std::vector<Video>::const_iterator it = videos.begin(); it != videos.end(); ++it){//const_iterator recorre el vector
^
Plataforma.cpp:72:89: error: expected ';' after expression
for (std::vector<Video>::const_iterator it = videos.begin(); it != videos.end(); ++it){//const_iterator recorre el vector
                                                                    ^
;
fatal error: too many errors emitted, stopping now [-ferror-limit=]
1 warning and 20 errors generated.
Serie.cpp:33:1: warning: non-void function does not return a value in all control paths [-Wreturn-type]
}
^
1 warning generated.
main.cpp:6:10: error: expected 'FILENAME' or '<FILENAME>'
#include
^
1 error generated.
ximixsm@MBP-de-Ximi situacionProblema % ./correprogra
zsh: no such file or directory: ./correprogra
ximixsm@MBP-de-Ximi situacionProblema %
```

Al compilar el programa ejecutando `./correprogra`, se han encontrado varios errores que impidieron una compilación exitosa debido a problemas en ciertas partes del código. Sin embargo, es importante destacar que la parte del programa encargada de identificar las series y películas funciona correctamente cuando se ejecuta el archivo ejecutable `./`.

Estos errores pueden ser resultado de problemas en la sintaxis, falta de declaración de variables, llamadas incorrectas a funciones o cualquier otro error que afecte la compilación del programa en su totalidad. Para solucionar estos errores, se recomienda revisar detenidamente el código fuente y corregir los problemas identificados, asegurándose de tener todas las dependencias y bibliotecas necesarias correctamente instaladas.



```
12
13 public:
14 Episodio(std::string id, std::string nombre, std::string genero, int calificacion, int duracion, std::string temporada) {
15     id = id; nombre = nombre; genero = genero; calificacion = calificacion; duracion = duracion; temporada = temporada;
16 }
17 int getnumeroEp();
18 void muestrainfo();
19
20 };
21 #endif
```

PROBLEMS 21

- 40 Pelicula
- 50 Episodio
- 51 Pelicula
- 52 Episodio
- 53 Pelicula
- 54 Pelicula
- 55 Episodio
- 56 Episodio
- 57 Episodio
- 58 Episodio
- 59 Pelicula
- 60 Episodio
- 61 Pelicula
- 62 Episodio
- 63 Episodio
- 64 Pelicula
- 65 Pelicula
- 66 Pelicula
- 67 Pelicula

Es fundamental tener en cuenta que la ejecución exitosa de la parte del programa relacionada con la identificación de series y películas en el archivo ejecutable "/" indica que dicha funcionalidad ha sido implementada de manera correcta. Sin embargo, se deben abordar y resolver los errores presentes en el proceso de compilación para asegurar el funcionamiento integral del programa en su totalidad.

### Identificación de casos

Al ejecutar el programa, se han detectado errores en varias secciones, especialmente en el archivo "plataforma.cpp" y dentro del archivo "main.cpp". Estos errores están afectando el correcto funcionamiento del programa en su totalidad.

Los errores específicos dentro de la clase "Plataforma" pueden requerir una revisión más exhaustiva, ya que implican problemas en los métodos y funcionalidades implementadas en dicha clase. Resolver estos errores puede llevar más tiempo, ya que pueden involucrar errores de sintaxis, problemas en la lógica de programación o falta de implementación de ciertas funciones requeridas.

Es importante destacar que, a pesar de los errores encontrados en la clase "Plataforma" y en el archivo "main.cpp", las demás clases del programa parecen estar implementadas correctamente. Sin embargo, para lograr un funcionamiento completo y libre de errores, es necesario abordar y corregir los problemas identificados en la clase "Plataforma" y en otras áreas del programa, si corresponde.

Se recomienda revisar cuidadosamente el código en "plataforma.cpp" y dentro del archivo "main.cpp", prestando especial atención a los mensajes de error proporcionados durante la compilación. Estos mensajes pueden brindar información útil sobre la naturaleza de los errores y ayudar a localizar y solucionar los problemas de manera más efectiva.

Dedicar tiempo y esfuerzo para resolver los errores en la clase "Plataforma" y en el archivo "main.cpp" es fundamental para garantizar el correcto funcionamiento del programa en su totalidad y lograr los resultados deseados.

## Conclusiones

En lo personal, quiero destacar que la clase Plataforma fue mi mayor reto en este proyecto y también la que más me generó estrés. Aún se me dificulta comprender completamente el concepto del polimorfismo, especialmente debido a la rapidez con la que hemos abordado los temas en el curso. Sin embargo, valoro el desafío que representó, ya que me permitió identificar áreas de mejora en mi comprensión y aplicación del polimorfismo.

Por otro lado, quiero resaltar que el diagrama de clases resultó ser una herramienta muy útil para mí. Pude organizar de manera clara y visual la estructura del programa, lo cual me brindó una mayor claridad sobre cómo podía aplicar los conceptos aprendidos. El diagrama de clases me facilitó la planificación y el diseño de la implementación, lo cual considero una fortaleza en mi proceso de programación.

En resumen, este proyecto me ha dejado en claro la importancia de profundizar en el entendimiento del polimorfismo y de abordar los conceptos a mi propio ritmo. Asimismo, he aprendido a aprovechar el uso de herramientas como el diagrama de clases para organizar y planificar mis proyectos de manera más efectiva. Estoy motivada a seguir practicando y ampliando mis conocimientos en este campo, con el objetivo de dominar el polimorfismo y aplicarlo de manera exitosa en futuros proyectos.



## Referencias

- Botero, T.R., Castro, C.C. y Parra, C. J. (abril, 2006). Método integrado de programación secuencial y programación orientada a objetos para el análisis, diseño y elaboración de algoritmos – MIPS00. Revista Virtual Universidad Católica del Norte, 17. Recuperado de <https://www.redalyc.org/pdf/1942/194220465010.pdf>
- Román, Z.C.A. (2017). Bloques y sentencias: la sentencia switch. Universidad Nacional Autónoma de México. Recuperado de [http://profesores.fi-b.unam.mx/carlos/java/java\\_basico2\\_7\\_2.html](http://profesores.fi-b.unam.mx/carlos/java/java_basico2_7_2.html)