

**Situación problema: modelado de servicio de streaming**

Mariana Balderrábano Aguilar A01749581  
17 de junio de 2023

Programación Orientada a objetos  
Roberto Martínez Román

## Índice

Introducción .....	3
Diagrama de clases .....	3
Ejemplos de ejecución .....	4
Identificación de casos que harían que el proyecto deje de funcionar. ....	10
Conclusión personal.....	10
Referencias.....	12

## Introducción

Con el paso del tiempo, las plataformas de videos para ver series o películas (de streaming) han crecido exponencialmente, de acuerdo con BBVA, “El crecimiento de plataformas como Netflix, Twitch, Amazon Prime en los últimos años ha alcanzado tal punto que llegan a representar un 25% del mercado televisivo” (Plataformas de Streaming: ¿Cómo Ha Sido Su Crecimiento Con El Pasar de Los Años?, 2022). Es por ello por lo que este proyecto se centra en el desarrollo de un programa en el lenguaje de c++ con ayuda de programación orientada a objetos. Modelar este tipo de plataformas nos ayudará en un futuro a poder seguir innovando, a su vez nos podría ayudar a hacer más eficientes los procesos de los códigos y de esta manera finalmente brindar la mejor calidad de plataformas de streaming a los usuarios.

## Diagrama de clases

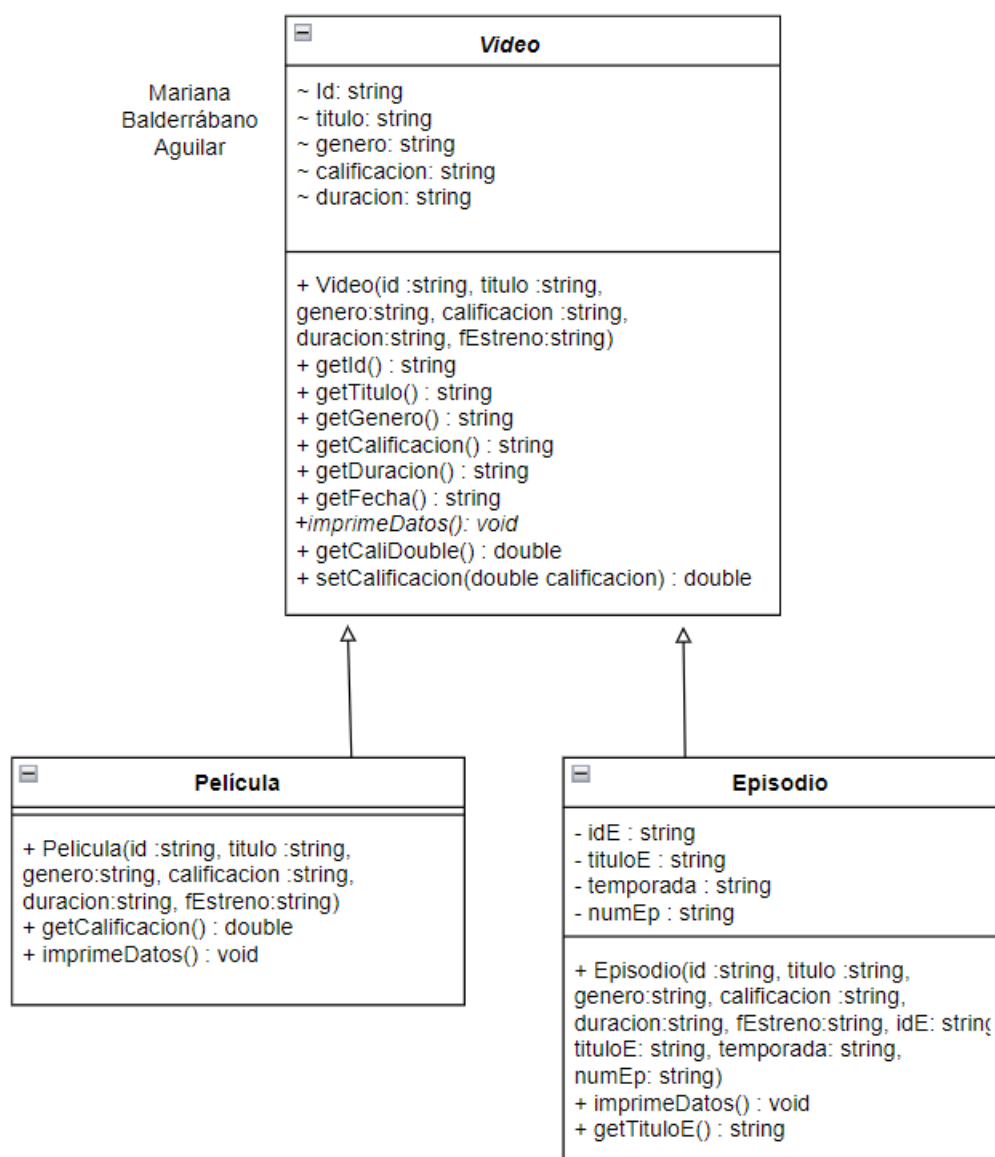


Figura 1. Diagrama de clases en lenguaje UML

En la Figura 1 podemos apreciar que para este proyecto se decidió trabajar con tres grandes clases, las cuales fueron, la clase Video, Película y Episodio.

La clase Video es la superclase o clase base, es decir que la clase Película y Episodio heredarán más adelante los atributos y métodos de la clase Video, esta misma clase contiene los atributos en común en todos los videos (película y episodio). En la parte de la superclase se definieron los atributos que describen a un objeto de tipo Video, es por ello que tenemos el id, título o nombre del video, el género, la calificación y la duración del video, en este caso, todos los atributos mencionados anteriormente son de tipo string, posteriormente tenemos a los getters (nos ayudarán a obtener el string de los atributos y en el caso de getCaliDouble, nos regresará la calificación en tipo double) y setCalificacion, que nos ayuda a definir el valor de calificación con la calificación dada en los parámetros, así como el constructor de Video con todos los atributos dentro del mismo y tenemos al método abstracto ImprimeDatos, lo cual hace que la clase Video, sea una clase abstracta. Más adelante, el último método mencionado se sobrescribirá de acuerdo a su tipo de dato.

Posteriormente tenemos la subclase Película, la cual dentro del recuadro de atributos no tiene nada, puesto que todos los atributos que necesita ya han sido heredados de la clase Video, por la parte de los métodos, tenemos el constructor con todos los atributos dentro, así como el getCalificacion que nos regresará el valor de la calificación con el tipo de dato double y por último ImprimeDatos, que es un método sobrescrito que nos ayudará a imprimir los datos, en este caso de la Película con un formato específico y de esta forma usamos polimorfismo.

Por otra parte, tenemos a la subclase Episodio, la cual heredó de Video, por lo que tendrá esos datos, pero de igual manera necesita más datos, por lo que en los atributos agregué idE, el cual es el id del episodio, así como tituloE, título del episodio, también temporada y numEp (número de episodios de la serie), todo con tipo de dato string. Por la parte de los métodos, tenemos el constructor Episodio, el cual tiene dentro los datos necesarios, los cuales son los atributos heredados de Video y los descritos anteriormente (id del episodio, título del episodio, temporada de la serie y número de episodios). Asimismo, tenemos el método imprimeDatos que es sobrescrito para que imprima los datos de tipo Episodio de una serie con formato y de esta manera usar polimorfismo. Finalmente, tenemos el método getTituloE, el cual nos regresará el título del episodio.

### Ejemplos de ejecución

1.- Carga el archivo csv

File Edit Selection View Go Run Terminal Help

Video.h - situacionProblema - Visual Studio Code

EXPLORER

OPEN EDITORS

DatosPelículas.csv

DiagramaDeClases.drawio.png U

C Video.h M X

leerArchivo.cpp M

Película.cpp

Episodio.cpp

Episodio.h

Episodio.h

Video.cpp

SITUACIONPROBLEMA

.vscode

doc

.gitignore

a.exe

a.exe.stackdump

DatosPelículas.csv

DiagramaDeClases.drawio.png U

Episodio.cpp

Episodio.h

leerArchivo.cpp M

Película.cpp

Película.h

README.md

Video.cpp

Video.h M

OUTLINE

TIMELINE

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS

PS C:\Users\maria\Desktop\cpp\Ejercicios\situacionProblema> g++ -std=c++11 \*.cpp

PS C:\Users\maria\Desktop\cpp\Ejercicios\situacionProblema> ./a.exe

-----Menú-----

Opciones

1 cargar el archivo csv

2 Mostrar videos en general con:

3 Mostrar episodios de una serie

4 Mostrar películas con calificación mayor al número ingresado

5 Calificar un video

6 Calcula el promedio de la calificación de una serie

7 Salir

Introduzca un número del 1 al 6 de acuerdo con su elección

1

Archivo cargado con éxito

-----Menú-----

Opciones

1 cargar el archivo csv

2 Mostrar videos en general con:

3 Mostrar episodios de una serie

4 Mostrar películas con calificación mayor al número ingresado

5 Calificar un video

6 Calcula el promedio de la calificación de una serie

7 Salir

Introduzca un número del 1 al 6 de acuerdo con su elección

1

Ln 25, Col 5 Spaces: 4 UTF-8 CRLF (1) C++ Win32

## 2.- Muestra videos con calificación mayor a la dada

File Edit Selection View Go Run Terminal Help

Video.h - situacionProblema - Visual Studio Code

EXPLORER

OPEN EDITORS

DatosPelículas.csv

DiagramaDeClases.drawio.png U

C Video.h M X

leerArchivo.cpp M

Película.cpp

Episodio.cpp

Episodio.h

Episodio.h

Video.cpp

SITUACIONPROBLEMA

.vscode

doc

.gitignore

a.exe

a.exe.stackdump

DatosPelículas.csv

DiagramaDeClases.drawio.png U

Episodio.cpp

Episodio.h

leerArchivo.cpp M

Película.cpp

Película.h

README.md

Video.cpp

Video.h M

OUTLINE

TIMELINE

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS

1 mostrar videos con una calificación mayor o igual a un valor teclado

2 mostrar videos con un género teclado por el usuario

1

Videos con calificación mayor o igual a: 6

Videos con calificación mayor a 6:

Id serie: tt8944947

Título serie: Game Of Thrones

Duración del episodio: 57

Género de la serie: Accion&Aventura&Drama&Fantasy&Romance

Calificación del episodio: 6.3

Fecha de estreno del episodio:

Id del episodio: tt1668899

Título del episodio: Winter Is Coming

Temporada: 1

Número del episodio: 1

Id serie: tt8944947

Título serie: Game Of Thrones

Duración del episodio: 57

Género de la serie: Accion&Aventura&Drama&Fantasy&Romance

Calificación del episodio: 6.2

Fecha de estreno del episodio:

Id del episodio: tt1668746

Título del episodio: The Kingsroad

Temporada: 1

Número del episodio: 2

Id serie: tt8944947

Título serie: Game Of Thrones

Duración del episodio: 57

Género de la serie: Accion&Aventura&Drama&Fantasy&Romance

Calificación del episodio: 6.1

Fecha de estreno del episodio:

Id del episodio: tt1829962

Título del episodio: Lord Snow

Temporada: 1

Número del episodio: 3

Id serie: tt8944947

Ln 25, Col 5 Spaces: 4 UTF-8 CRLF (1) C++ Win32

```

1  /*
2  3  Autor: Mariana Balderrábano Aguilar
3  4  Definición de superclase Video (clase abstracta)
4  5  */

```

Título película: Pulp Fiction  
 Duración de la película: 154  
 Género de la película: Crimen&Drama  
 Calificación de la película: 6.2  
 Fecha de estreno de la película:  
 Id serie: tt2085859  
 Título serie: Black Mirror  
 Duración del episodio: 60  
 Género de la serie: Drama&Sci-Fi&Suspense  
 Calificación del episodio: 6.2  
 Fecha de estreno del episodio:  
 Id del episodio: tt2542420  
 Título del episodio: White Bear  
 Temporada: 2  
 Número del episodio: 2  
 Id película: tt8468569  
 Título película: The Dark Knight  
 Duración de la película: Accion&Crimen&Drama&Suspense  
 Género de la película: 152  
 Calificación de la película: 6.3  
 Fecha de estreno de la película:  
 Id serie: tt4574334  
 Título serie: Stranger Things  
 Duración del episodio: 51  
 Género de la serie: Drama&Fantasy&Terror&Misterio&Sci-Fi&Suspense  
 Calificación del episodio: 6.2  
 Fecha de estreno del episodio:  
 Id del episodio: tt4593132  
 Título del episodio: Chapter Six: The Monster  
 Temporada: 1  
 Número del episodio: 6  
 -----Menú-----  
 Opciones  
 1 cargar el archivo csv  
 2 Mostrar videos en general con:  
 3 Mostrar episodios de una serie  
 4 Mostrar películas con calificación mayor al número ingresado

## 2.- Muestra videos con el género dado

```

1  /*
2  3  Autor: Mariana Balderrábano Aguilar
3  4  Definición de superclase Video (clase abstracta)
4  5  */

```

5 calificar un video  
 6 calcula el promedio de la calificación de una serie  
 7 Salir  
 Introduzca un número del 1 al 6 de acuerdo con su elección  
 1  
 Archivo cargado con éxito  
 -----Menú-----  
 Opciones  
 1 cargar el archivo csv  
 2 Mostrar videos en general con:  
 3 Mostrar episodios de una serie  
 4 Mostrar películas con calificación mayor al número ingresado  
 5 calificar un video  
 6 calcula el promedio de la calificación de una serie  
 7 Salir  
 Introduzca un número del 1 al 6 de acuerdo con su elección  
 2  
 1 mostrar videos con una calificación mayor o igual a un valor tecleado  
 2 mostrar videos con un género tecleado por el usuario  
 2  
 Ejemplos de géneros:  
 Romance Accion Aventura Drama Fantasy Suspense Sci-Fi  
 Comedia Misterio Animacion Crimen Drama Biografia Familiar  
 Género: Romance  
 Videos con género Romance:  
 Id serie: tt8944947  
 Título serie: Game Of Thrones  
 Duración del episodio: 57  
 Género de la serie: Accion&Aventura&Drama&Fantasy&Romance  
 Calificación del episodio: 6.3  
 Fecha de estreno del episodio:  
 Id del episodio: tt1668899  
 Título del episodio: Winter Is Coming  
 Temporada: 1  
 Número del episodio: 1

## 3.- Muestra episodios de una serie

```

1  /*
2  Autor: Mariana Balderrábano Aguilar
3  Definición de superclase Video (clase abstracta)
4  */
5  Calificar un video
6  Calcular el promedio de la calificación de una serie
7  Salir
8  Introduzca un número del 1 al 6 de acuerdo con su elección
9
10 3
11  Game Of Thrones  The Big Bang Theory  Memories Of The Alhambra  Death Note  Fate Zero  Unbreakable Kimmy Schmidt
12  Black Mirror  Chilling Aventura of Sabrina  Stranger Things  Maniac  The Simpsons  The 100  Modern Familiar  Friends
13  Introduzca una serie de la que quiere ver los datos de sus episodios: The Simpsons
14  Id serie: tt0096697
15  Título serie: The Simpsons
16  Duración del episodio: 22
17  Género de la serie: Animacion&Comedia
18  Calificación del episodio: 5.4
19  Fecha de estreno del episodio:
20  Id del episodio: tt0756593
21  Título del episodio: Bart The Genius
22  Temporada: 1
23  Número del episodio: 1
24  Id serie: tt0096697
25  Título serie: The Simpsons
26  Duración del episodio: 22
27  Género de la serie: Animacion&Comedia
28  Calificación del episodio: 5.4
29  Fecha de estreno del episodio:
30  Id del episodio: tt0756399
31  Título del episodio: There No Disgrace Like Home
32  Temporada: 1
33  Número del episodio: 3
34  Id serie: tt0096697
35  Título serie: The Simpsons
36  Duración del episodio: 22
37  Género de la serie: Animacion&Comedia
38  Calificación del episodio: 5.2
39  Fecha de estreno del episodio:
40  Id del episodio: tt0701124
41  Título del episodio: Homer Odyssey

```

#### 4.- Muestra películas con una calificación mayor o igual a la dada

```

12  using namespace std;
13
14  //Declara clase video
15  class Video
16  {
17  //Atributos de video
18
19  3  Mostrar episodios de una serie
20  4  Mostrar películas con calificación mayor al número ingresado
21  5  Calificar un video
22  6  Calcular el promedio de la calificación de una serie
23  7  Salir
24  Introduzca un número del 1 al 6 de acuerdo con su elección
25
26  4
27  Películas con calificación mayor a:
28  6.3
29  Id película: tt0088763
30  Título película: Back To The Future
31  Duración de la película: Aventura&Comedia&Sci-Fi
32  Género de la película: 116
33  Calificación de la película: 6.9
34  Fecha de estreno de la película:
35  Id película: tt0468569
36  Título película: The Dark Knight
37  Duración de la película: Accion&Crimen&Drama&Suspense
38  Género de la película: 152
39  Calificación de la película: 6.3
40  Fecha de estreno de la película:
41  -----Menú-----
42  Opciones
43  1  cargar el archivo csv
44  2  Mostrar videos en general con:
45  3  Mostrar episodios de una serie
46  4  Mostrar películas con calificación mayor al número ingresado
47  5  Calificar un video
48  6  Calcular el promedio de la calificación de una serie
49  7  Salir
50  Introduzca un número del 1 al 6 de acuerdo con su elección

```

#### 5.- Califica un video y define la nueva calificación

Visual Studio Code interface showing the file explorer on the left with the project 'SITUACIONPROBLEMA' expanded. The main editor displays the 'DatosPelículas.csv' file. The terminal shows the output of a program that prompts the user to enter a movie title and a rating, then displays the movie details and the rating.

```

17 tt0848228,The Avengers,143,Acción&Aventura&Sci-Fi,5.6,05/04/2012,,,,
18 tt0415778,End Game,93,Acción&Crímen&Drama&Misterio&Suspense,3.6,07/21/2006,,,,
19 tt0108778,Friends,22,Comedia&Romance,5.9,09/22/1994,tt0583459,The One Where Monica Gets A Roommate,1,1
20 tt0108778,Friends,22,Comedia&Romance,5.7,09/29/1994,tt0583647,The One With The Sonogram At The End,1,2
21 tt0108778,Friends,22,Comedia&Romance,5.7,10/06/1994,tt0583653,The One With The Thumb,1,3
22 tt1853728,Django Unchained,165,Drama&Western,5.9,12/25/2012,,,,
23 tt1442437,Modern Family,22,Comedia&Drama&Romance,5.9,09/23/2009,tt1444504,Pilot,1,1

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS

```

7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección
5
Ingrese el título de la película o nombre del capítulo de la serie a calificar: Mulan
Ingresa la nueva calificación(del 1 al 7): 3.9
Calificación actualizada correctamente
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calcular el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección
5
Ingrese el título de la película o nombre del capítulo de la serie a calificar: End Game
Ingresa la nueva calificación(del 1 al 7): 6.7
Calificación actualizada correctamente
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calcular el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

```

Visual Studio Code interface showing the file explorer on the left with the project 'SITUACIONPROBLEMA' expanded. The main editor displays the 'leerArchivo.cpp' file. The terminal shows the output of a program that prompts the user to enter a movie title and a rating, then displays the movie details and the rating.

```

1 // leerArchivo.cpp
2 #include <iostream>
3 #include <string>
4 #include <vector>
5 #include <fstream>
6 #include <sstream>
7 #include <algorithm>
8 #include <limits>
9 #include <memory>
10 #include <string.h>
11 #include <conio.h>
12 #include <windows.h>
13 using namespace std;
14
15 struct Video {
16     string titulo;
17     int duracion;
18     float calificacion;
19 };
20
21 vector<Video> videos;
22
23 void cargarArchivo() {
24     ifstream archivo("DatosPelículas.csv");
25     if (!archivo.is_open()) {
26         cout << "Error al cargar el archivo." << endl;
27         return;
28     }
29     string linea;
30     while (getline(archivo, linea)) {
31         stringstream ss(linea);
32         string id, titulo, genero, rating, fecha, duracion;
33         ss >> id >> titulo >> genero >> rating >> fecha >> duracion;
34         Video v;
35         v.titulo = titulo;
36         v.duracion = atoi(duracion.c_str());
37         v.calificacion = atof(rating.c_str());
38         videos.push_back(v);
39     }
40     archivo.close();
41 }
42
43 void mostrarVideos() {
44     if (videos.empty()) {
45         cout << "No hay videos cargados." << endl;
46         return;
47     }
48     for (int i = 0; i < videos.size(); i++) {
49         cout << i << ": " << videos[i].titulo << ", " << videos[i].duracion << " minutos, " << videos[i].calificacion << " de calificación." << endl;
50     }
51 }
52
53 void calificarVideo() {
54     int i;
55     for (i = 0; i < videos.size(); i++) {
56         cout << i << ": " << videos[i].titulo << ", " << videos[i].duracion << " minutos, " << videos[i].calificacion << " de calificación." << endl;
57     }
58     int indice;
59     cout << "Ingrese el índice del video a calificar: ";
60     while (true) {
61         indice = getche();
62         if (indice < '0' || indice > '9') continue;
63         break;
64     }
65     if (indice < 0 || indice > videos.size() - 1) {
66         cout << "Índice fuera de rango." << endl;
67         return;
68     }
69     float calificacion;
70     cout << "Ingrese la nueva calificación (del 1 al 7): ";
71     while (true) {
72         calificacion = getche();
73         if (calificacion < '0' || calificacion > '9') continue;
74         break;
75     }
76     videos[indice].calificacion = atof(calificacion.c_str());
77     cout << "Calificación actualizada correctamente." << endl;
78 }
79
80 void calcularPromedio() {
81     float suma = 0;
82     for (int i = 0; i < videos.size(); i++) {
83         suma += videos[i].calificacion;
84     }
85     float promedio = suma / videos.size();
86     cout << "El promedio de la calificación de una serie es: " << promedio << endl;
87 }
88
89 void menu() {
90     cout << "-----Menú-----" << endl;
91     cout << "Opciones" << endl;
92     cout << "1 cargar el archivo csv" << endl;
93     cout << "2 Mostrar videos en general con:" << endl;
94     cout << "3 Mostrar episodios de una serie" << endl;
95     cout << "4 Mostrar películas con calificación mayor al número ingresado" << endl;
96     cout << "5 Calificar un video" << endl;
97     cout << "6 Calcular el promedio de la calificación de una serie" << endl;
98     cout << "7 Salir" << endl;
99 }
100
101 int main(int argc, char const * argv[]) {
102     cargarArchivo();
103     menu();
104     int opcion;
105     while (true) {
106         opcion = getche();
107         switch (opcion) {
108             case '1':
109                 cargarArchivo();
110                 break;
111             case '2':
112                 mostrarVideos();
113                 break;
114             case '3':
115                 mostrarVideos();
116                 break;
117             case '4':
118                 mostrarVideos();
119                 break;
120             case '5':
121                 calificarVideo();
122                 break;
123             case '6':
124                 calcularPromedio();
125                 break;
126             case '7':
127                 return 0;
128             default:
129                 menu();
130         }
131     }
132 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS

```

1 Archivo cargado con éxito
-----Menú-----
Opciones
1 Cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calcular el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección
5
Ingrese el título de la película o nombre del capítulo de la serie a calificar: Come Fly With Me
Ingresa la nueva calificación (del 1 al 7): 5
Calificación actualizada correctamente
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calcular el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

```

6.- Calcula el promedio de una serie y la regresa



```

17 tt0448228,The Avengers,143,Acción&Aventura&Sci-Fi,5.6,05/04/2012,,,,,
18 tt0415778,End Game,93,Acción&Crimen&Drama&Historio&Suspense,3.6,07/21/2006,,,,,
19 tt0108778,Friends,22,Comedia&Romance,5.9,09/22/1994,tt0583459,The One Where Monica Gets A Roommate,1,1
20 tt0108778,Friends,22,Comedia&Romance,5.7,09/29/1994,tt0583647,The One With The Sonogram At The End,1,2
21 tt0108778,Friends,22,Comedia&Romance,5.7,10/06/1994,tt0583653,The One With The Thumb,1,3
22 tt1853728,Django Unchained,165,Drama&Western,5.9,12/25/2012,,,,,
23 tt1442437,Modern. Familiar,22,Comedia&Drama&Romance,5.9,09/23/2009,tt1444504,Pilot,1,1

```

```

7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

1
Archivo cargado con éxito
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calula el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

6
Game Of Thrones The Big Bang Theory Memories Of The Alhambra Death Note Fate Zero Unbreakable Kimmy Schmidt
Black Mirror Chilling Aventura of Sabrina Stranger Things Maniac The Simpsons The 100 Modern Familiar Friends
Nombre de la serie de la que desea obtener el promedio:
Friends
El promedio de las calificaciones de la serie Friends es de: 5.76667
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calula el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

```

## 7.- Salir

```

343 {
344     cout << "Serie no encontrada, intente otra vez" << endl;
345 }
346 break;
347 }
348 }
349 //salir
350 case 7:
351     cout << "Ha salido, hasta pronto :)"<<endl;
352     x = false;
353     break;
354 }
355 default:
356     cout << "argumento no válido" << endl;
357     break;
358 }
359 }
360 }
361 }
362 vector<string> separar(string linea)
363 {

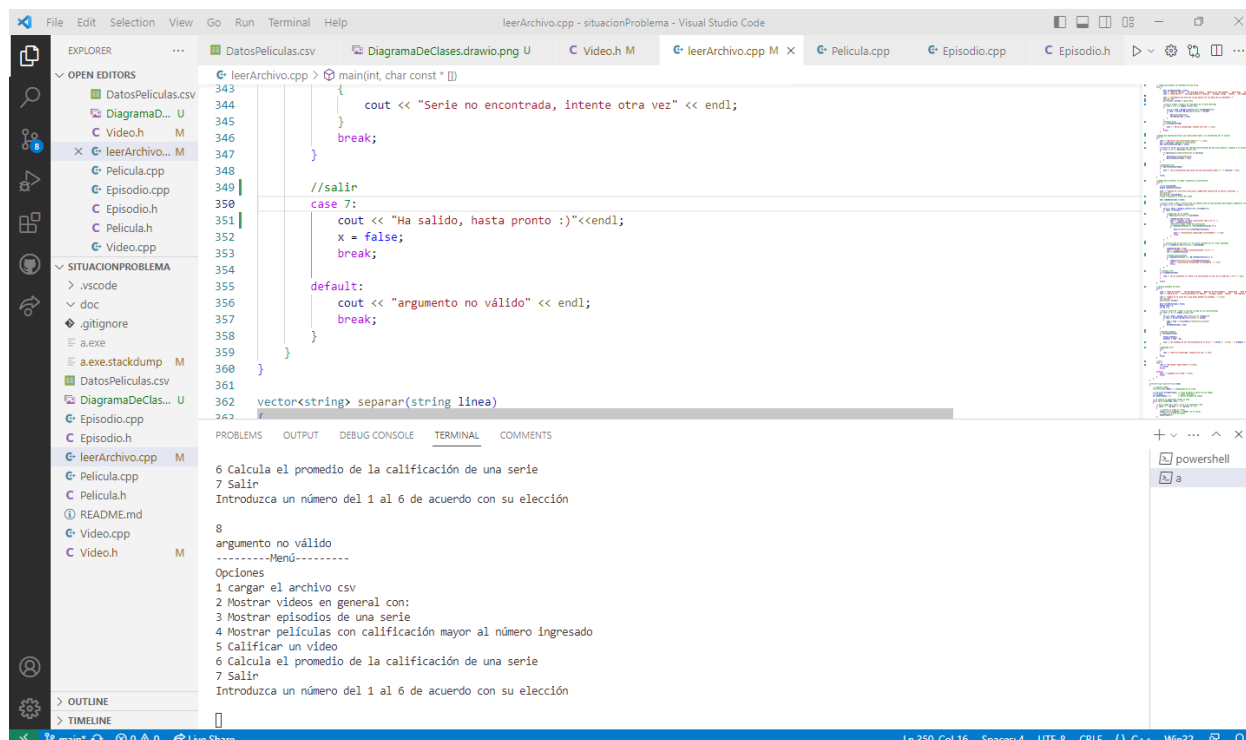
```

```

PS C:\Users\maria\Desktop\cpp\Ejercicios\situacionProblema> g++ -std=c++11 *.cpp
PS C:\Users\maria\Desktop\cpp\Ejercicios\situacionProblema> ./a.exe
-----Menú-----
Opciones
1 cargar el archivo csv
2 Mostrar videos en general con:
3 Mostrar episodios de una serie
4 Mostrar películas con calificación mayor al número ingresado
5 Calificar un video
6 Calula el promedio de la calificación de una serie
7 Salir
Introduzca un número del 1 al 6 de acuerdo con su elección

7
Ha salido, hasta pronto :)
PS C:\Users\maria\Desktop\cpp\Ejercicios\situacionProblema>

```



Cuando un argumento no es válido, regresa a menú.

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

Algunos de los casos que harían que el proyecto deje de funcionar sería que el orden del archivo csv cambiara, que cambiara el número de datos que tiene un episodio y una película, pues no se logrará distinguir un episodio de una película, que existieran más valores por separar, que el usuario escribiera mal el género o lo escribiera con todas mayúsculas o todas minúsculas, que pusieran algún espacio de más al ingresar los títulos de las películas o ponerle acento, pues puede que en el documento csv no tengan acento.

### Conclusión personal

Esta situación problema me ayudó a mejorar mis habilidades de programación, con ella pude distinguir cosas que se me dificultan y de igual manera reforzar lo aprendido. En lo personal, la lectura de archivos se me complicó, gracias a la actividad noté que necesito practicarlo más, me pareció interesante la solución propuesta por mí, asimismo, esta actividad me desafió, pues a veces producía errores que tuve que investigar para poder resolver, gracias a esta actividad pude mejorar mi habilidad para leer y comprender los errores para poder así corregirlos. El diagrama de clases en lenguaje UML me pareció de suma importancia hacerlo al inicio, ya que me ayudó demasiado a poder tener una idea más clara del código, pero como es inevitable, a lo largo de la actividad se tuvieron que hacer algunas modificaciones. Me sorprendió mucho el avance que pude tener de primer semestre a este segundo semestre ya que pude comprender mejor los códigos y de igual forma logré entender muchas cosas que no podía el semestre pasado de programación orientada a objetos, sinceramente para mí representó un logro personal poder

desarrollar este código de manera individual, ya que el semestre pasado, estoy casi segura de que no lo hubiera logrado yo sola. Gracias a la actividad entendí la importancia del diagrama de clases, pude aplicar los conocimientos aprendidos en clase y logré obtener más conocimiento gracias a los errores que se generaban a lo largo del código.

## Referencias

DuHadway, L. (2020). The string npos static constant [YouTube Video]. In YouTube.

<https://www.youtube.com/watch?v=BqnXcsZ7-6w>

fajardo, eldonedgar. (2020). 02 Programación C++ -- Getline Cin.Ignore [YouTube Video].

In YouTube. <https://www.youtube.com/watch?v=3bbZkofM594>

Luis Cabrera Benito. (2021). Leer archivo CSV con C++ [YouTube Video]. In YouTube.

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

Plataformas de streaming: ¿Cómo ha sido su crecimiento con el pasar de los años? (2022, July

15). BBVA.CH. <https://www.bbva.ch/noticia/plataformas-de-streaming-como-ha-sido-su-crecimiento-con-el-pasar-de-los-anos/>

Programación ATS. (2016). 12. Programación en C++ || Condicionales || La sentencia switch

[YouTube Video]. In YouTube. <https://www.youtube.com/watch?v=hjDB7ev9ePM>

The Chernobyl. (2019). Dynamic Casting in C++ [YouTube Video]. In YouTube.

<https://www.youtube.com/watch?v=CiHfz6pTolQ>