Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

КУРСОВА РОБОТА

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліни “Об’єктно-орієнтоване програмування”

ФІЛЬМОТЕКА

Керівник, проф. Бондарєв В. М.

Студент гр. ПЗПІ-23-4 Прядко М. А

Комісія:

проф. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бондарєв В. М.

доц. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Кравець Н. С.

cт. викл. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Черепанова Ю. Ю.

Харків 2024

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра *програмної інженерії*

Рівень вищої освіти *перший (бакалаврський)*

Дисципліна *Об’єктно-орієнтоване програмування*

Спеціальність *121 Інженерія програмного забезпечення*

Освітня програма *Програмна інженерія*

Курс *1* Група *ПЗПІ-23-4* Семестр *2*

***ЗАВДАННЯ***

***на текстовий проєкт студента***

Прядко Максим Андрійович

(Прізвище, Ім’я, По батькові)

1. Тема проєкту:

Фільмотека rrtrtrrtrtrtrtrttrrrtrrtrrtrtrtrtrtrtrtrrtttrt

1. Термін здачі студентом закінченого проекту: ***“08” - червня - 2024 р.***
2. Вихідні дані до проекту:

*База відеофайлів: назва фільму, студія, жанр, рік випуску, режисер, виконавці головних ролей, короткий зміст, суб'єктивна оцінка фільму, розташування, розмір. Перевірка наявності фільму в відеотеці. Підбір фільму для перегляду по ряду критеріїв, підбір. Складання вибірки ряду фільмів за часом відтворення, розміру зовнішнього носія, жанру і акторської і режисерському складу.*

Зміст розрахунково-пояснювальної записки:

*Вступ, опис вимог, проєктування програми, інструкція користувача, висновки*

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |
| --- | --- | --- |
| *№* | *Назва етапу* | *Термін виконання* |
| 1 | Видача теми, узгодження і затвердження | 13.02.2024 - 15.03.2024 р. |
| 2 | Формулювання вимог до програми | 01.03.2024 – 05.03.2024 р. |
| 3 | Розробка підсистеми зберігання та завантаження даних | 13.05.2024 –13.05.2024 р. |
| 4 | Розробка функцій додавання, редагування та видалення фільмів | 13.05.2024 –14.05.2024 р. |
| 5 | Розробка підбору фільмів та складання вибірки | 15.05.2024 –17.05.2024 р. |
| 6 | Тестування і доопрацювання розробленої програмної системи | 18.05.2024 –19.05.2024 р. |
| 7 | Оформлення пояснювальної записки, додатків, графічного матеріалу | 19.05.2024 – 21.05.2024 р. |
| 8 | Захист | 03.06.2024–08.06.2024 р. |

Студент Прядко Максим Андрійович

Керівник Бондарєв Володимир Михайлович

«21» лютого 2024 р.

**РЕФЕРАТ**

Пояснювальна записка до курсової роботи: 25 с., 21 рис., 0 табл., 0 додатки, 0 джерел.

БІБЛІОТЕКА, ФІЛЬМИ, ЗВІТ, ООП, .NET, МОВА C#

Метою роботи є розробка програми «Фільмотека», яка буде надавати користувачу інформацію про різні кінокартини, колекціонерів, музеї, комісійні магазини, аукціони та картини.

В результаті отримана програма, що дозволяє додавати фільми та їх  
характеристики, такі як: назва, студія, жанр, рік випуску, режисер виконавці  
головних ролей, короткий зміст, суб’єктивна оцінка, розташування на диску,  
розмір. Є можливість підбирати фільм за критеріями та створити вибірку. Також, є функція, видаляти та редагувати додані фільми

В процесі розробки використано середовище розробки Microsoft Visual Studio 2022, фреймворк Windows Forms, платформа .NET 8.0, мова програмування C#.

**ЗМІСТ**

[Вступ 7](#_Toc74841188)

[1 Опис вимог 8](#_Toc74841189)

[1.1 Опис сценаріїв. 8](#_Toc74841190)

[2 Проєктування програми 10](#_Toc74841192)

[3 Інструкція користувачу 20](#_Toc74841195)

[Висновок 25](#_Toc74841198)

**ВСТУП**

В тему курсового проекту входить програма, дані якого зберігаються у форматі JSON.

Багато людей мають власну колекцію фільмів. І було б доволі зручно мати повну інформацію про кожну кінокартину в одному місці. Створення такої програми значно полегшить процес організації та перегляду фільмів

Програма, розроблена в рамках цієї курсової роботи, допоможе створити персоналізовану бібліотеку фільмів. Вона не тільки надає детальну інформацію про кожен фільм, включаючи жанр, режисера, склад акторів, студію та сюжет, але й має функцію підбирання фільмів.

для успішної реалізації цього застосунку необхідно мати навички об'єктно-орієнтованого програмування. Цей підхід дозволить створити систему, яку можна буде легко розширювати та модифікувати, забезпечуючи подальшу підтримку в майбутньому.

1. **ОПИС ВИМОГ**

1.1 Опис сценаріїв

**Сценарій 1. Додавання фільму**

*Передумова*

Користувач запустив програму та обрав у меню пункт додавання фільму.

*Основний сценарій*

1. Програма відкриває вікно додавання фільму.
2. Користувач заповнює поля у вікні додавання і натискає кнопку ”OK”.
3. Програма перевіряє коректність даних в полях, створює новий фільм і додає його до списку.

*Додатковий сценарій*

1. Програма відкриває вікно додавання фільму.
2. Користувач заповнює поля у вікні додавання і натискає кнопку ”OK”.
3. Програма встановлює, що дані не коректні.
4. Програма сповіщає користувача про помилку в даних і про те, яким чином її виправити.

**Сценарій 2. Редагування фільму**

*Передумова*

Користувач запустив програму, обрав фільм зі списку та натиснув у меню пункт редагування фільму.

*Основний сценарій*

1. Програма відкриває вікно редагування фільму.
2. Користувач змінює інформацію у полях і натискає кнопку ”OK”.
3. Програма перевіряє коректність даних в полях і оновлює інформацію про фільм.

*Додатковий сценарій*

1. Програма відкриває вікно редагування фільму.
2. Користувач заповнює поля у вікні додавання і натискає кнопку ”OK”.
3. Програма встановлює, що дані не коректні.
4. Програма сповіщає користувача про помилку в даних і про те, яким чином її виправити.

**Сценарій 3. Видалення фільму**

*Передумова*

Користувач запустив програму, обрав фільм зі списку та натиснув у меню пункт видалення фільму.

*Основний сценарій*

1. Програма відкриває вікно у якій питає користувача чи хоче він видалити фільм.
2. Користувач натискає кнопку ”OK”.
3. Програма видаляє фільм зі списку разом з усією пов’язаною інформацією до нього.

**Сценарій 4. Підбір фільму**

*Передумова*

Користувач запустив програму та натиснув у меню пункт ”Підбір фільму”.

*Основний сценарій*

1. Програма відкриває вікно з полями критеріїв.
2. Користувач заповнює поля.
3. Користувач натискає кнопку ”OK”.
4. Програма підбирає фільми та відображає їх в окремому списку

*Додатковий сценарій*

1. Програма відкриває вікно з полями критеріїв.
2. Користувач заповнює поля.
3. Програма встановлює, що немає збігів.
4. Програма повертає користувачу пустий список.

**Сценарій 5. Створення вибірки** *Передумова*

Користувач запустив програму та натиснув у меню пункт ”Створити вибірку”.

*Основний сценарій*

1. Програма відкриває вікно з полями критеріїв.
2. Користувач заповнює поля.
3. Користувач натискає кнопку ”OK”.
4. Програма підбирає фільми та відображає їх в окремому списку

*Додатковий сценарій*

1. Програма відкриває вікно з полями критеріїв.
2. Користувач заповнює поля.
3. Програма встановлює, що немає збігів.
4. Програма повертає користувачу пустий список.

**2 ПРОЄКТУВАННЯ ПРОГРАМИ**

В даній програмі містяться форми, та класи, які взаємодіють з даними формами (рисунок 2.1).

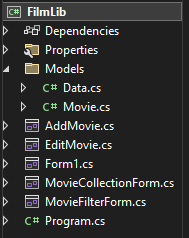


Рисунок 2.1– Архітектура програми.

При написанні програми реалізовані такі основні класи, як «Data» та «Movies».

Клас «Data» (рисунок 2.2) є основним класом для збереження та завантаження даних. Цей клас дозволяє зберігати інформацію про фільми у файлі JSON, і завантаження їх до програми.

Клас «Movies» (рисунок 2.3) містить властивості для зберігання інформації про фільм. Кожна властивість класу представляє такі аспекти фільму як: назва, студія, рік випуску, жанр, режисера, головних акторів і тд.

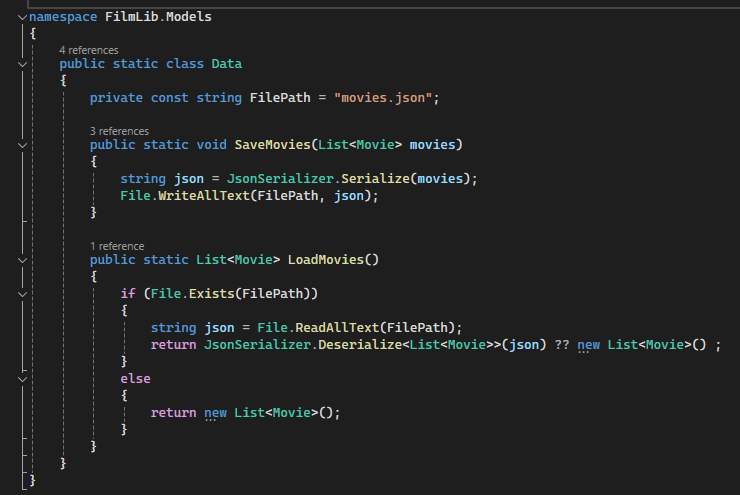


Рисунок 2.2– Реалізація класу «Data».

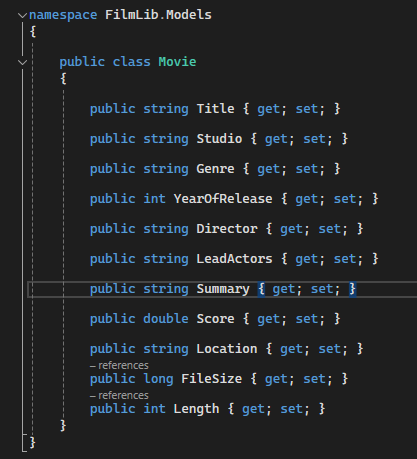


Рисунок 2.3– Реалізація класу «Movies».

Форма «Form1» робить усі необхідні дії на головній сторінці. Натискання на кнопку, додавання, редагування, видалення фільму та інші методи були реалізовані саме тут.

При запуску програми, вона одразу завантажує інформацію з JSON файлу, та відображає її у списку за допомогою методу «PopulateListBox» (Рисунок 2.4).

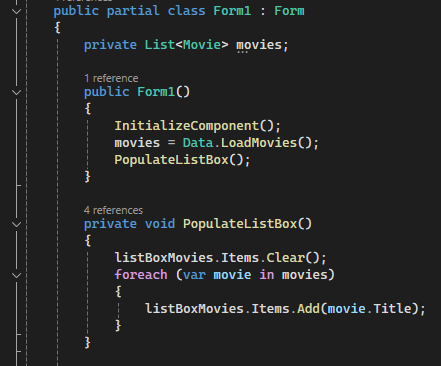
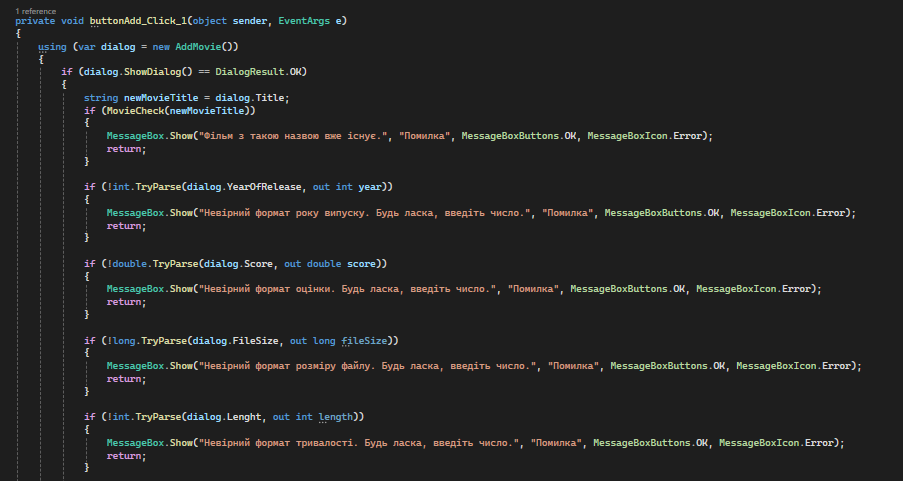


Рисунок 2.4– Завантаження даних, та відображення у списку

При натисканні кнопки «Додати», програма викликає форму «AddMovie» (Рисунок 2.7), у якій користувач вводить інформацію у відповідні поля. Після цього програма створює новий об’єкт «Movie» та додає його до списку. (Рисунок 2.5)

У проміжку між натисканням кнопки додавання у формі, і появою фільму у списку, програма перевіряє його назву на співпадіння з існуючими фільмами, використовуючи метод «MovieCheck» (Рисунок 2.6). У разі випадку, коли користувач заповнив поля для року випуску, тривалості, оцінки або розміру файлу некоректно, програма сповістить користувача про це, та пояснить як саме треба виправити.



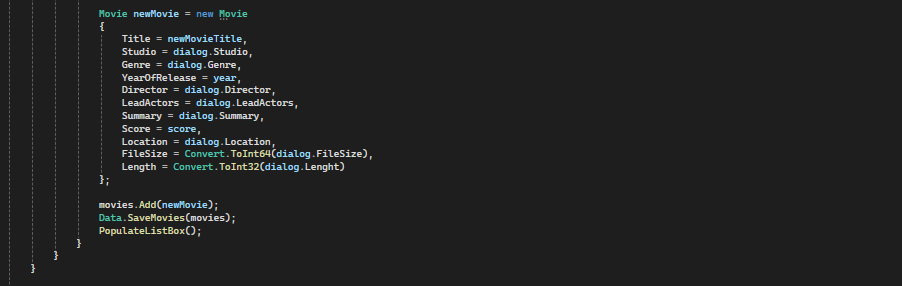


Рисунок 2.5– Реалізація функції додавання фільму

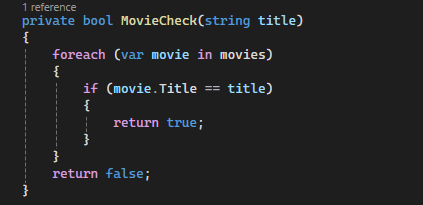
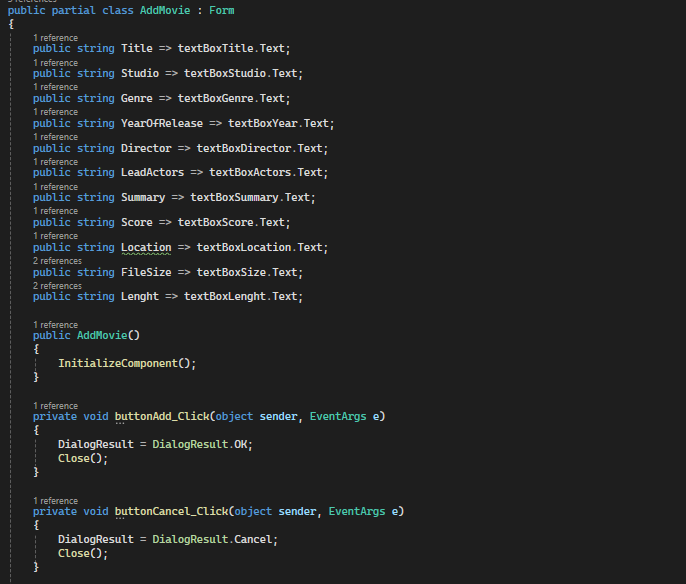


Рисунок 2.6– Реалізація методу «MovieCheck»



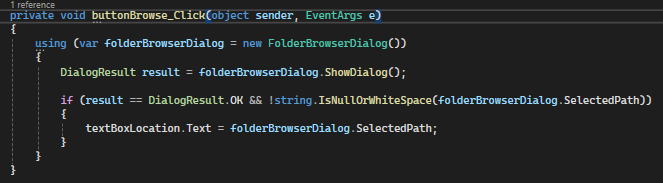


Рисунок 2.7 – Реалізація форми «AddMovie»

Виділивши фільм у списку і натиснувши кнопку редагування, відкривається вікно «EditMovie» (Рисунок 2.9) у якій користувач може змінити інформацію про фільм (Рисунок 2.8).

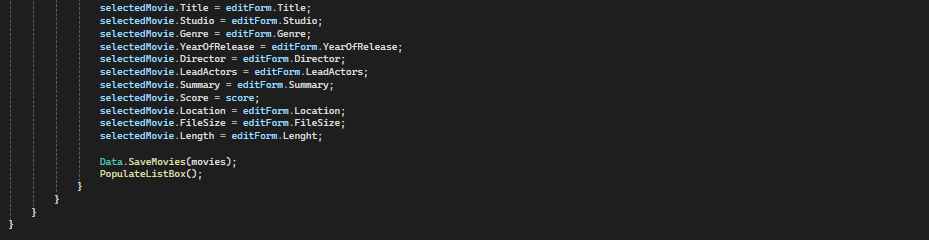
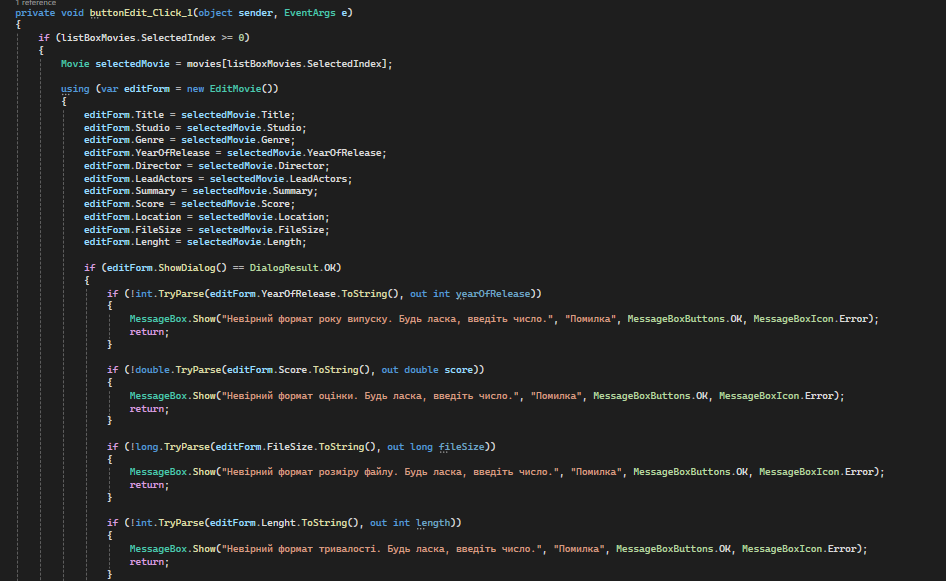
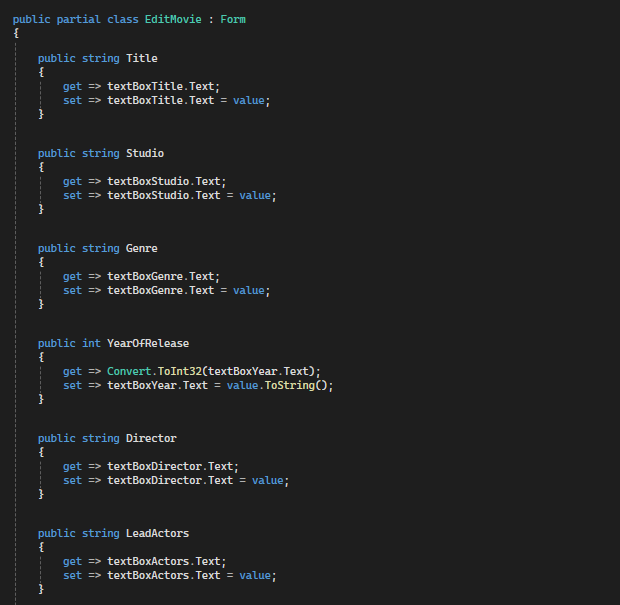


Рисунок 2.8 – Реалізація функції редагування.



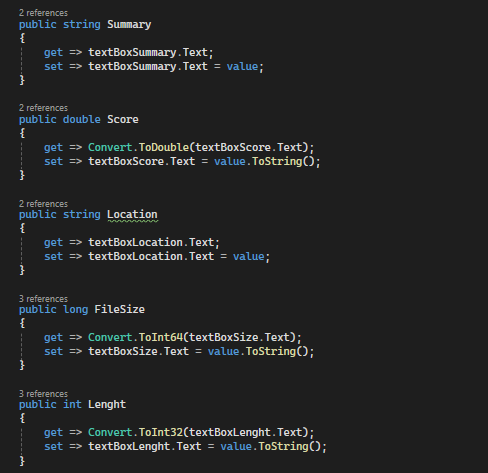


Рисунок 2.9 – Реалізація форми «EditMovie»

Виділивши фільм, і натиснувши кнопку видалення, з’являється вікно, у якому користувач повинен підтвердити свій вибір перед повним видаленням фільму зі списку. (Рисунок 2.10).

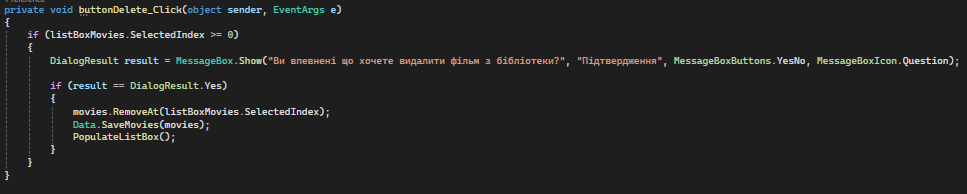


Рисунок 2.10 – Реалізація функції видалення.

Обравши фільм у списку, інформація про нього відображається в об’єкті «dataGridViewDetails». В коді задано створення стовпців, та зміна їх параметрів для того, щоб текст який не вміщається переносився на новий рядок. Також реалізовано форматування параметру «Length» щоб у таблиці він відображався у форматі «години:хвилини:секунди» (Рисунок 2.11).

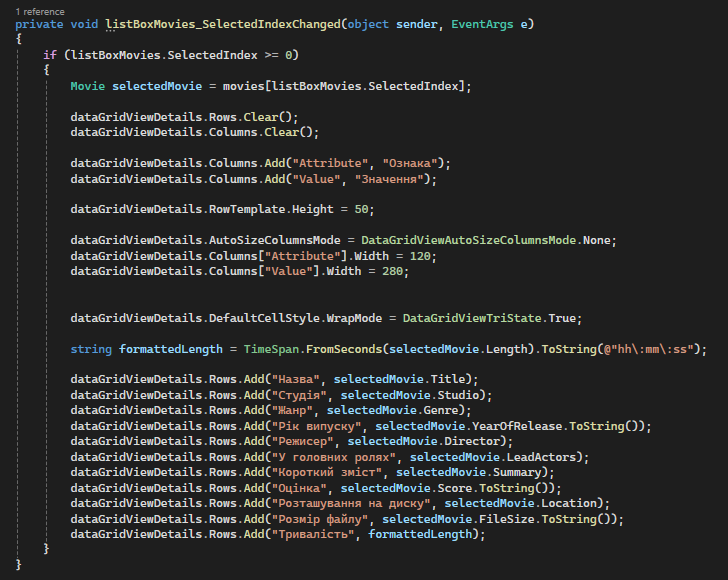


Рисунок 2.11 – Реалізація відображення інформації у таблиці.

Обравши у верхньому меню пункт «Підбір фільму» (Рисунок 2.12) або «Створити вибірку» (Рисунок 2.13) з’являться відповідні вікна, у яких користувач, заповнивши відповідні поля, може підібрати фільм за критеріями. Програма порівнює введену інформацію з тією, що прив’язана до фільму і виводить знайдені кінокартини у списку. Програма успішно покаже результати, навіть якщо критерій було вписано частково. Додавання відфільтрованих фільмів до окремого списку реалізовано методом «PopulateFilter» (Рисунок 2.14).

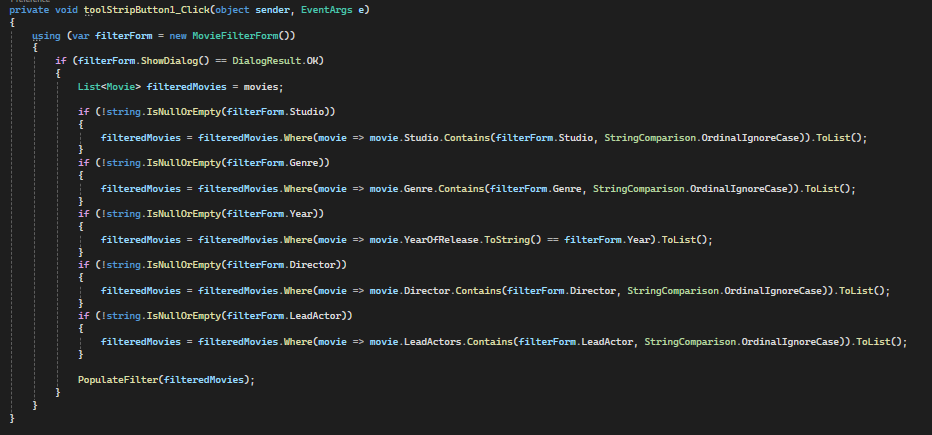


Рисунок 2.12 – Реалізація функції «Підбір фільму»

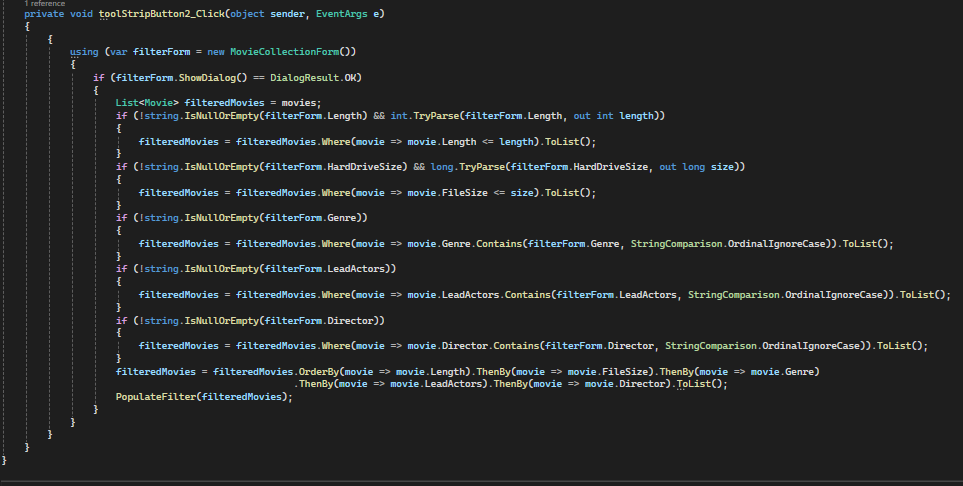


Рисунок 2.13 – Реалізація функції «Створити вибірку»

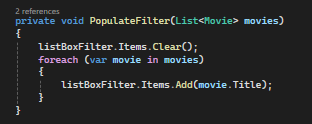


Рисунок 2.14 – Реалізація методу «PopulateFilter»

**3 ІНСТРУКЦІЯ КОРИСТУВАЧУ**

Запускаємо програму. Відкривається головне меню (Рисунок 3.1)

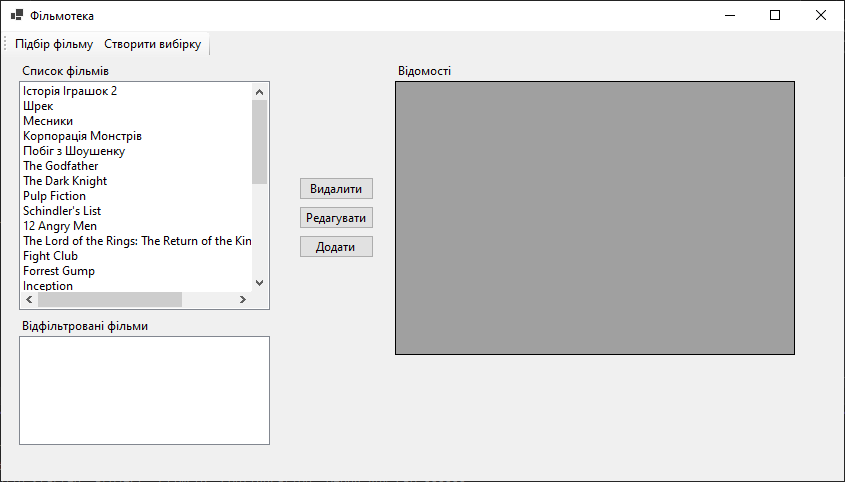


Рисунок 3.1– Головне меню.

Додавання нового фільму до бібліотеки:

* У головному меню натиснути кнопку «Додати»;
* У відкритому вікні заповнити усі поля (Рисунок 3.2);
* натискаємо на кнопку «Додати» з лівого нижнього блоку;
* після успішного додавання інформації фільм з’явиться у списку;

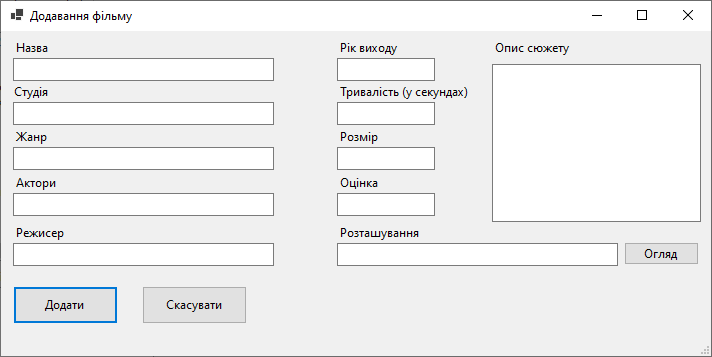


Рисунок 3.2 – Меню додавання фільму.

Змінення будь-якої інформації про фільм:

* У головному меню обрати фільм зі списку;
* натискаємо кнопку «Редагувати»;
* У відкритому вікні змінюємо інформацію на бажану (Рисунок 3.3);
* натискаємо на кнопку «Зберегти» з лівого нижнього блоку;
* після успішного зберігання, інформація про фільм оновиться;

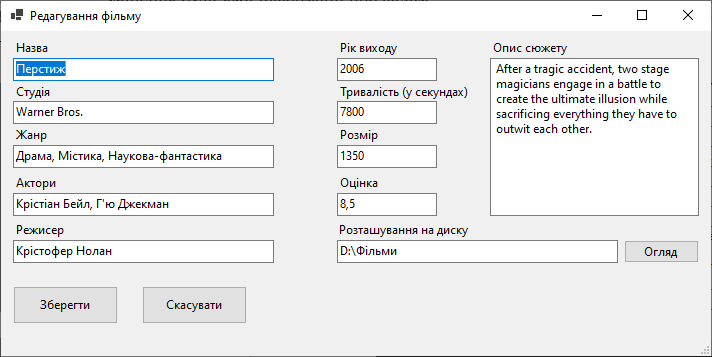


Рисунок 3.3 – Меню редагування фільму.

Видалення фільму зі списку:

* У головному меню обрати фільм зі списку;
* натискаємо кнопку «Видалити»;
* У діалоговому вікні підтверджуємо свій вибір, натиснувши «Так» (Рисунок 3.4);
* після видалення, фільм зникне зі списку;

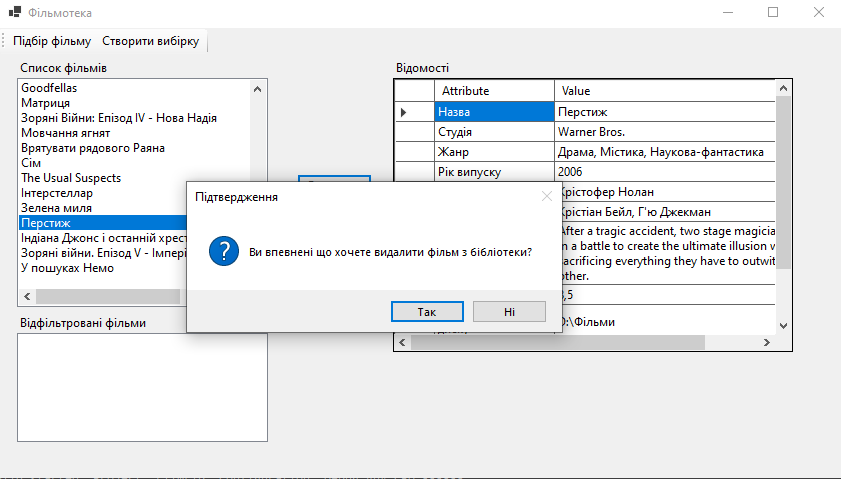


Рисунок 3.4 – Діалогове вікно про підтвердження видалення фільму.

Підбір фільму:

* У верхньому меню натиснути кнопку «Підбір фільму»;
* У відкритому вікні задаємо бажані критерії (Рисунок 3.5);
* натискаємо на кнопку «Підібрати» з правого нижнього блоку;
* після цього фільм, або фільми які підпадають під критерії з’являться у списку з відфільтрованими фільмами (Рисунок 3.6);

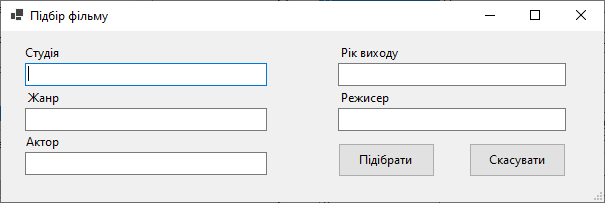


Рисунок 3.5 – Меню підбору фільму.

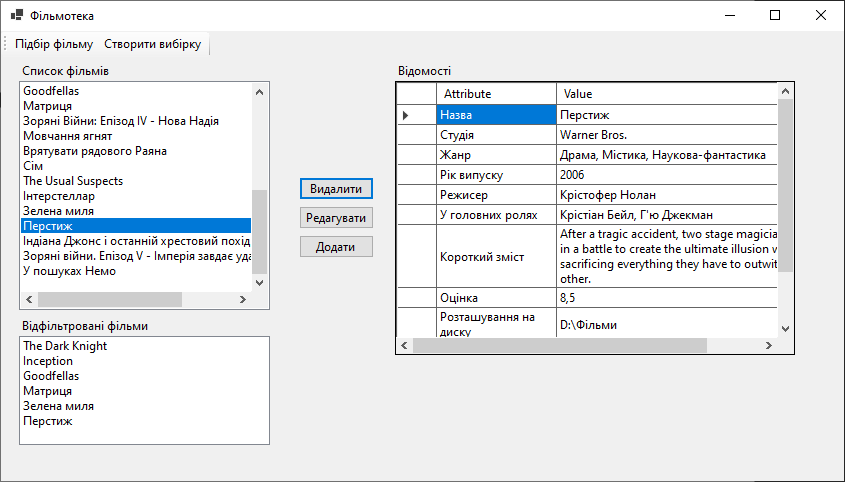


Рисунок 3.6 – Головне меню з відфільтрованими фільмами.

Створення вибірки фільмів:

* У верхньому меню натиснути кнопку «Створити вибірку»;
* У відкритому вікні задаємо бажані критерії (Рисунок 3.7);
* натискаємо на кнопку «ОК» з правого нижнього блоку;
* після цього фільм, або фільми які підпадають під критерії з’являться у списку з відфільтрованими фільмами;

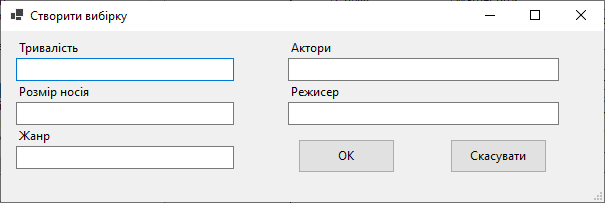


Рисунок 3.7 – Меню створення вибірки.

**ВИСНОВОК**

Під час виконання курсової роботи була розроблена програма «Фільмотека», яка дозволяє створювати список фільмів. В ході виконання роботи були реалізовані наступні методи: перегляд інформації про фільми, редагування інформації, підбір, створення вибірки, збереження фільму до файлу JSON.

В результаті написання цієї курсової роботи була створена повноцінна програма, з повним набором функції. Всі данні розроблені на програмній мові С#, а також вивчені та розібрані різні аспекти об’єктно-орієнтованого програмування, розробка програм у середовищі .NET.

В ході виконання курсової роботи отримано досвід розробки програм, вивчені стандарти програмного забезпечення, а також планування ходу розробки.

Програма не є ідеальною, тому є можливості для подальшого вдосконалення.