

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Кафедра інженерії програмного забезпечення

КУРСОВИЙ ПРОЕКТ (РОБОТА)
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

з дисципліни: «Об'єктно-орієнтоване програмування»

на тему:

“Каса кінотеатру”

студента I курсу групи ІПЗ-21-5
спеціальності 121 «Інженерія програмного
забезпечення»

Гейни Владислава Сергійовича

(прізвище, ім'я та по-батькові)

Керівник старший викладач кафедри ІПЗ

Олег ВЛАСЕНКО

Дата захисту: " ____ " _____ 20__ р.

Національна шкала _____

Кількість балів: _____

Оцінка: ECTS _____

Члени комісії

(підпис)

(підпис)

(підпис)

Тетяна ВАКАЛЮК

(прізвище та ініціали)

Олексій ЧИЖМОТРА

(прізвище та ініціали)

Дмитро ПРОХОРЧУК

(прізвище та ініціали)

Житомир – 2022

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Факультет інформаційно-комп'ютерних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень: бакалавр

Спеціальність 121 «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»

Зав. кафедри

_____ Андрій МОРОЗОВ

“ _____ ” _____ 20__ р.

ЗАВДАННЯ
НА КУРСОВИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ
Гейні Владиславу Сергійовичу

- Тема роботи: Розробка програми “Каса кінотеатру”,
керівник роботи: старший викладач кафедри інженерії програмного забезпечення Власенко Олег Васильович
- Строк подання студентом: “ _____ ” _____ 20__ р.
- Вихідні дані до роботи: Розробити програму “Каса кінотеатру”
- Зміст розрахунково-пояснювальної записки(перелік питань. Які підлягають розробці)
 - аналіз методів та засобів вирішення задачі
 - проектування та розробка програмного додатку
 - опис роботи з програмними додатком та його тестування
- Перелік графічного матеріалу(з точним зазначенням обов'язкових креслень)
 - Презентація до курсового проекту
 - Посилання на репозиторій: <https://gitlab.com/2021-2025/ipz-21-5/geyna-vladyslav/courseworks/first-course/second-semester/coursework-cinema-box-office>
- Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посади консультанта	Підпис, дата	
		завдання видав	завдання прийняв

- Дата видачі завдання “ 11 ” лютого 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів курсового проекту (роботи)	Строк виконання етапів проекту	Примітки
1	Постановка задачі	11.02.2022	Виконано
2	Пошук, огляд та аналіз аналогічних розробок	18.02.2022	Виконано
3	Формулювання технічного завдання	26.02.2022	Виконано
4	Опрацювання літературних джерел	07.03.2022	Виконано
5	Проектування структури	15.03.2022	Виконано
6	Написання програмного коду	25.03.2022	Виконано
7	Відлагодження	03.06.2022	Виконано
8	Написання пояснювальної записки	13.06.2022	Виконано
9	Захист	19.07.2022	Виконано

Студент

(підпис)

Керівник проекту

(підпис)

Владислав ГЕЙНА

(прізвище та ініціали)

Олег ВЛАСЕНКО

(прізвище та ініціали)

РЕФЕРАТ

Завданням на курсовий проект (роботу) була розробка програми “Каса кінотеатру”.

Пояснювальна записка до курсового проекту (роботи) на тему «Розробка програми “Каса кінотеатру”» складається з вступу, трьох розділів, висновків, списку використаної літератури та додатків.

Текстова частина викладена на 36 сторінках друкованого тексту.

Пояснювальна записка має 24 сторінки додатків. Список використаних джерел містить 12 найменувань і займає 1 сторінку. В роботі наведено 35 рисунків. Загальний обсяг роботи – 62 сторінки.

Ключові слова: C#, БАЗИ ДАНИХ, SQLITE, LINQ, WPF, ORM, MATERIAL DESIGN, XAML, ENTITY FRAMEWORK, ПРОГРАМА, ПЗ, ДОДАТОК, ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ, ПАТЕРН.

					ДУ «Житомирська політехніка».22.121.3.000 - ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка програми “Каса кінотеатру”			Літ.	Арк.	Аркушів		
Розроб.		Владислав ГЕЙНА								4	62	
Перевір.		Олег ВЛАСЕНКО						ФІКТ Гр. ІПЗ-21-5[2]				
Керівник												
Н. контр.												
Зав. каф.		Андрій МОРОЗОВ										

ЗМІСТ

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧ.....	7
1.1 Аналіз задачі, засобів та методів її вирішення.	7
1.2 Аналіз існуючого програмного забезпечення за тематикою курсового проекту	8
РОЗДІЛ 2 ПРОЕКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	12
2.1. Проектування загального алгоритму роботи програми	12
2.2. Розробка функціональних алгоритмів роботи програми.....	13
2.3. Розробка програмного забезпечення.....	16
РОЗДІЛ 3 ОПИС РОБОТИ З ПРОГРАМНИМ ДОДАТКОМ ТА ЙОГО ТЕСТУВАННЯ	24
3.1. Опис роботи з програмним додатком	24
3.2 Тестування роботи програмного забезпечення.....	32
ВИСНОВКИ.....	36
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	37
ДОДАТКИ.....	38

ВСТУП

На сьогоднішній день кінематограф стає все більш популярним, з кожним роком підвищується кількість та якість фільмів, а кінотеатрів стає все більше. У зв'язку з цим підвищення автоматизації роботи кінотеатру є доволі актуальною проблемою, вирішенням якої є розробка ПЗ для продажу квитків і автоматичного оформлення замовлення.

Актуальність теми даного курсового проекту полягає у тому, що використання цієї системи кінотеатрами значно спрощує багато процесів, які пов'язані з обробкою замовлень та знижує витрати на такі речі, як: принтери, касові апарати, комп'ютери та утримання працівників – касирів. Також великою перевагою є зменшення багатьох довгих та рутинних операцій, пов'язаних з оформленням замовлення, що значно збільшує прибуток компанії.

Що стосується користувача, то дана програма дозволить йому у зручній формі ознайомитись з фільмом, переглянувши трейлер та основні деталі, а також швидко й просто зробити замовлення завдяки зрозумілому та не складному інтерфейсу.

Метою даного курсового проекту є розробка програми для перегляду деталей про фільм та оформлення замовлення, тобто квитка, з можливістю додавання товарів, таких як, наприклад: 3D окуляри, вода, чи попкорн.

Об'єктом дослідження є кінотеатри та їх система продажу квитків.

Предметом дослідження є вивчення можливостей програмування мовою C#, інтерфейсу програмування додатків WPF та використання цього для створення каси кінотеатру для оформлення замовлення та продажу квитків.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				6
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧ

1.1 Аналіз задачі, засобів та методів її вирішення.

Задачею курсового проекту є реалізація завершеного програмного продукту “Каса кінотеатру”. Він повинен мати повноцінний функціонал для отримання доступу до всіх потрібних функцій, бути інтуїтивно зрозумілим для усіх користувачів, простим у використанні та мати приємний та зручний інтерфейс. Також додаток має коректно працювати при введенні чи виборі неправильних значень.

Отож, даний додаток повинен додавати фільми та товари в обране, надавати інформацію про конкретний фільм, не допускати помилки в ході користування програмою та створювати квиток, де міститиметься все, що було додано в обране та інша інформація, яка була зібрана протягом формування замовлення.

Для реалізації даного програмного продукту буде використовуватись інтегроване середовище розробки програмного забезпечення Microsoft Visual Studio 2022, а саме платформа Windows Presentation Foundation (WPF), що дуже розширить наші можливості та дозволить більш гнучко оформити інтерфейс додатку, а також бібліотека Material Design In XAML Toolkit, яка дозволить створити дуже привабливий дизайн додатку за рахунок готових стилів для різноманітних елементів. Також буде обрано платформу .NET Framework, об'єктно-орієнтовану мову програмування C#, та систему управління базами даних SQLite. Для взаємодії з базою даних та її таблицями буде використано Entity Framework. Він представляє спеціальну об'єктно-орієнтовану технологію для роботи з даними та дозволяє абстрагуватись від самої бази даних, передбачаючи роботу не з таблицями, а з об'єктами та їх наборами. Таким чином він нам значно полегшить роботу із БД.

При реалізації слід притримуватись об'єктно-орієнтованого підходу, адже це допомагає уникнути деяких помилок, та у подальшому дозволяє з легкістю підтримувати сам проект.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				
Змн.	Арк.	№ докум.	Підпис	Дата		7

1.2 Аналіз існуючого програмного забезпечення за тематикою курсового проекту

Зараз, на жаль, існує не велика кількість додатків для оформлення замовлення в кінотеатрах. Більшість з них представлені саме як веб-сайти.

Провівши пошук, можна виділити чотири сервіси для купівлі квитків, а саме: Multiplex, SmartCinema, Планета Кіно та vkino.

Усі вищевказані сервіси дозволяють забронювати місце, купити квиток онлайн, в них є пошук міста, перегляд інформації про фільм.

Сервіс SmartCinema

Посилання на веб-сайт: <https://smartcinema.ua/>.

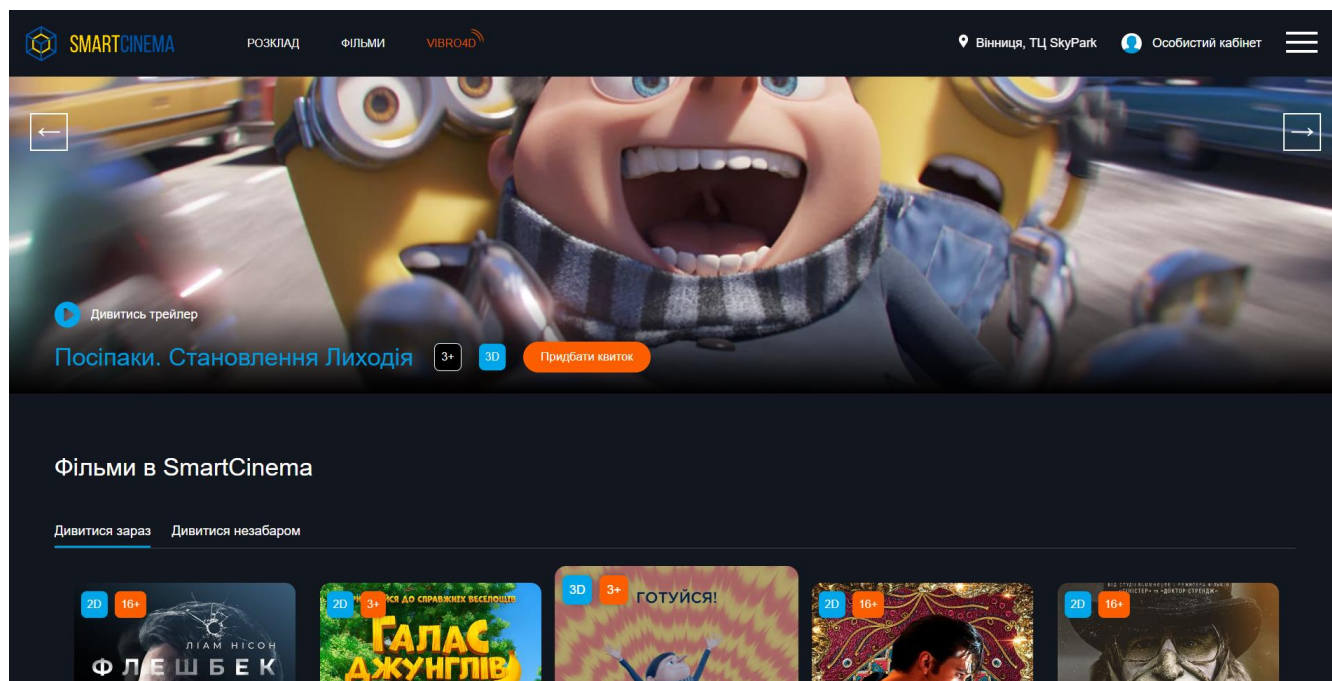


Рис. 1.1. Вигляд головної сторінки веб-сайту SmartCinema

Переваги:

- Є можливість обрати кінотеатр.
- Можна обрати режим перегляду (2D або 3D), фільм, а також сеанс.
- Є коротка інформація про фільм та можливість перегляду трейлеру.

Недоліки:

- Немає товарів.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				8
Змн.	Арк.	№ докум.	Підпис	Дата		

- Погана валідація вікового обмеження.
- Для певної групи користувачів інтерфейс може бути не зручним.

Сервіс vkino

Посилання на веб-сайт: <https://vkino.com.ua/ua>.

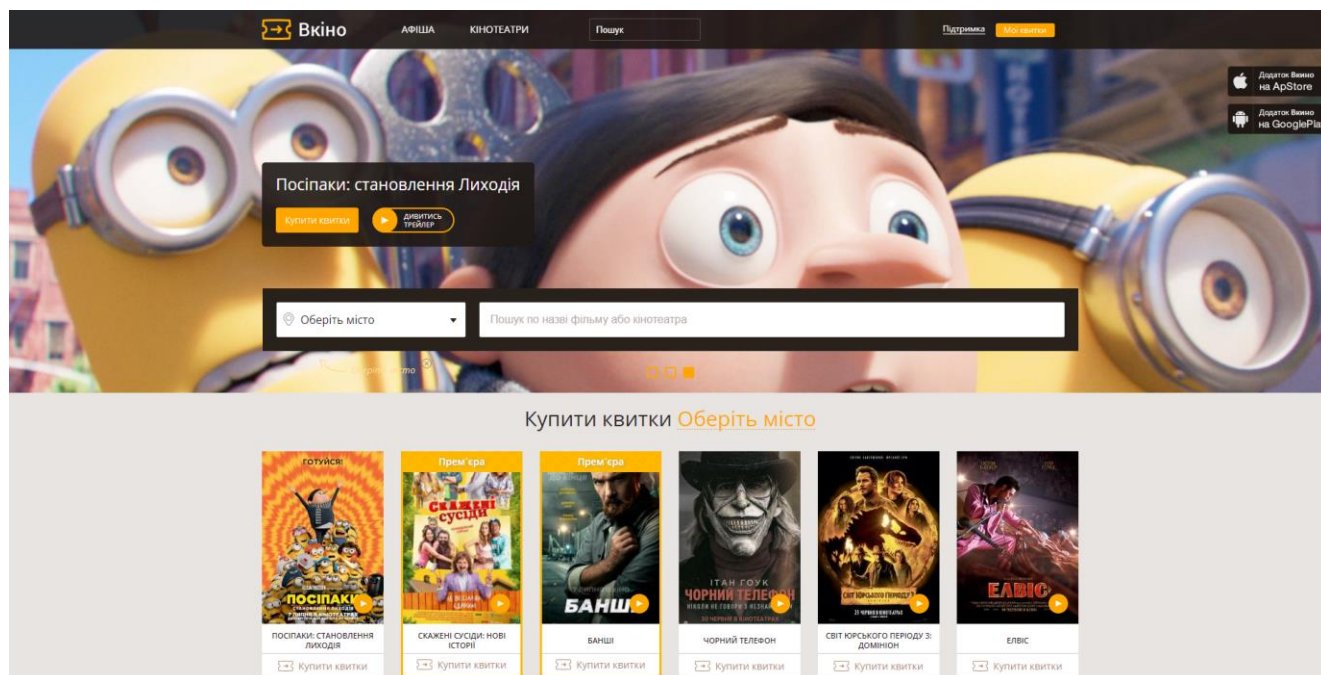


Рис. 1.2. Вигляд головної сторінки веб-сайту vkino

Переваги:

- Можливість вибору кінотеатру.
- Найбільший вибір міст та кінотеатрів.
- Можливість повернути квитки.
- Можна обрати режим перегляду (2D або 3D), фільм, а також сеанс.
- Можна переглянути інформацію про фільм та трейлер.

Недоліки:

- Для того, щоб зробити замовлення потрібно зареєструватись.
- Немає товарів.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				9
Змн.	Арк.	№ докум.	Підпис	Дата		

Сервіс Планета Кіно

Посилання на веб-сайт: <https://planetakino.ua/>.

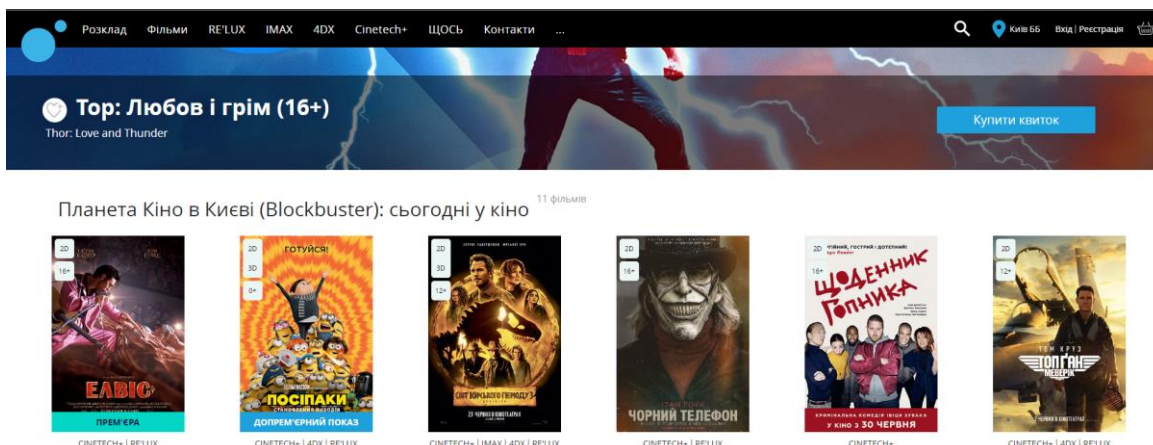


Рис. 1.3. Вигляд головної сторінки веб-сайту Планета Кіно

Переваги:

- Можливість вибору кінотеатру.
- Можливість вибору міста.
- Можна обрати режим перегляду (2D або 3D), фільм, а також сеанс.
- Можливість оформити замовлення як гість, без реєстрації.
- Є інформація про фільм та трейлер.

Недоліки:

- Немає товарів.
- Немає перевірки, вікового обмеження.

Сервіс Multiplex

Посилання на веб-сайт: <https://multiplex.ua/>.

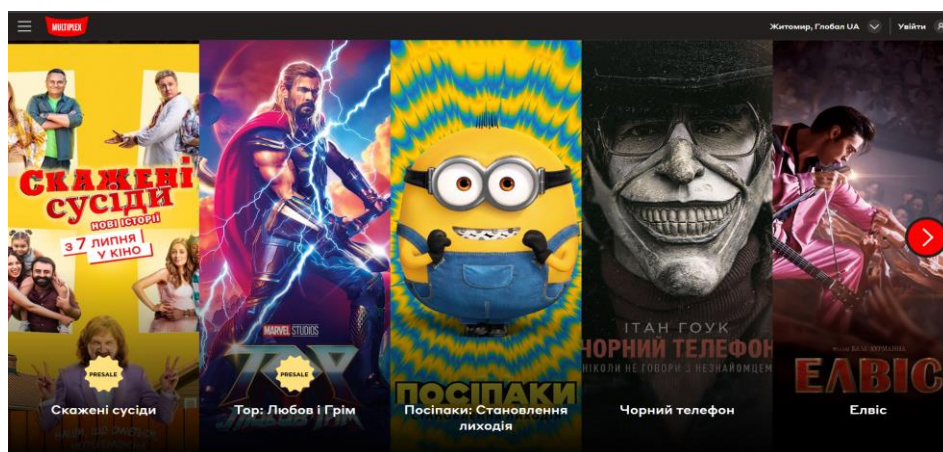


Рис. 1.4. Вигляд головної сторінки веб-сайту Multiplex

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Переваги:

- Можливість вибору міста.
- Можливість оформити замовлення як гість, без реєстрації.
- Є детальна інформація про фільм та трейлер.
- Є товари.
- Є тарифи та знижки.
- Є різноманітні способи оплати.
- Можна обрати режим перегляду (2D або 3D), фільм, а також сеанс.
- Інтуїтивно зрозумілий та зручний інтерфейс.

Недоліки:

- Для оформлення замовлення є всього сім хвилин.

Отже, найкращою онлайн-касою для купівлі квитків є сервіс Multiplex. Він найбільш зручний у користуванні, має магазин з товарами. Також порівняно з іншими сервісами Multiplex має набагато менше недоліків.

Висновки до першого розділу:

В першому розділі було поставлено задачу проекту, яким має бути функціонал, обрано платформу та мову розробки, обрано підхід для реалізації задля уникнення помилок. Також було розглянуто існуюче програмне забезпечення зі схожою тематикою до курсового проекту.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				11
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2 ПРОЕКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1. Проектування загального алгоритму роботи програми

На етапі проектування програмного забезпечення потрібно скласти загальний алгоритм роботи програми, провести дослідження яким чином краще сформулювати її структуру.

Після запуску додатку відкривається вікно реєстрації. Якщо користувач вже має акаунт, то він може одразу перейти до вікна авторизації.

Далі можуть бути виконані такі дії:

Дії на сторінці з фільмами “Зараз у прокаті”:

- При натисканні на кнопку “Детальніше” відкривається вікно із детальної інформації про фільм, яке користувач може закрити, або продовжити ознайомлення.
- В вікні користувач обирає дату та час сеансу та переходить до вибору місця в залі.
- Користувачем обирається ряд та місце в залі, після чого він додає цей фільм до обраного.

Дії на сторінці з товарами “Смаколики та інше”:

- За допомогою кнопок “+”, “-” або введення числа в текстове поле користувач може обрати потрібну кількість товару.
- Після чого він додає обрану кількість даного товару до обраного.

Дії на сторінці з обраним “Обране”:

- Якщо обране не є порожнім, то користувач за бажанням може його очистити, тобто видалити всі товари або ж сформувати замовлення.

Отже, загальна схема (алгоритм) роботи програмного додатку, з урахуванням всіх функцій, які він має виконувати зображена на рис. 2.1.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				12
Змн.	Арк.	№ докум.	Підпис	Дата		

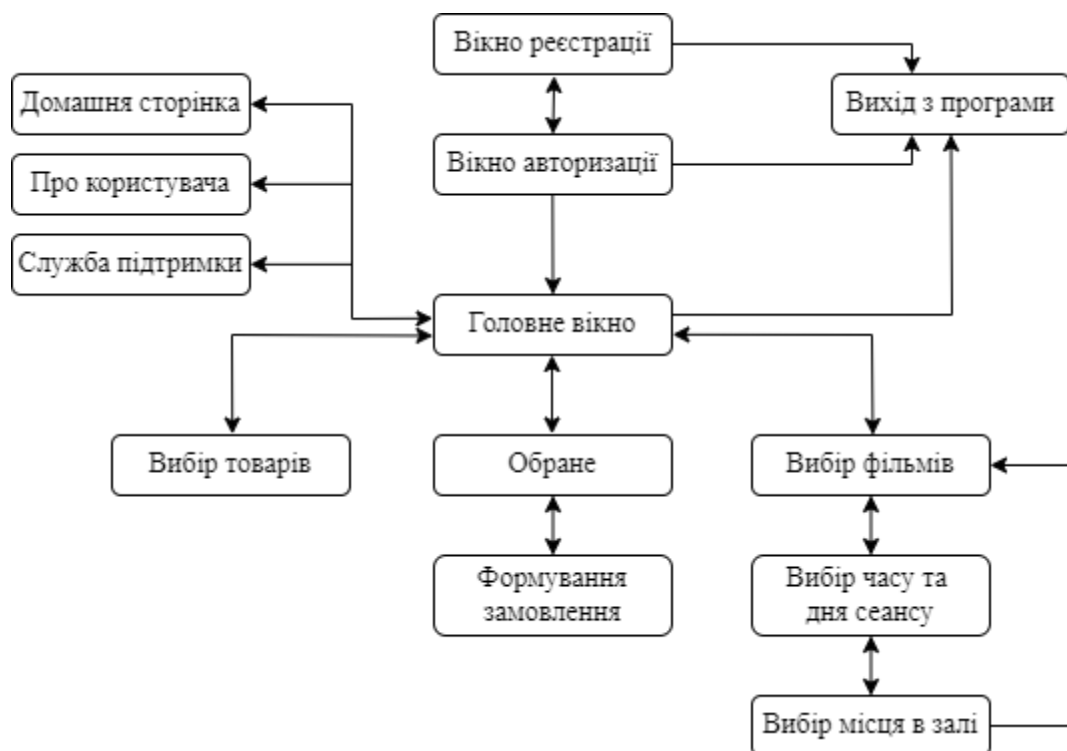


Рис. 2.1. Загальний алгоритм роботи програми

2.2. Розробка функціональних алгоритмів роботи програми

Після етапу проектування переходимо до етапу розробки ПЗ. Заздалегідь створений загальний алгоритм роботи програми полегшує написання програмного коду.

Для початку, потрібно створити базу даних, та три таблиці в ній (рис. 2.2.), в яких зберігатимуться всі дані користувача. В подальшому, за допомогою Entity Framework та запитів LINQ ми будемо звертатися до бази даних та її таблиць.

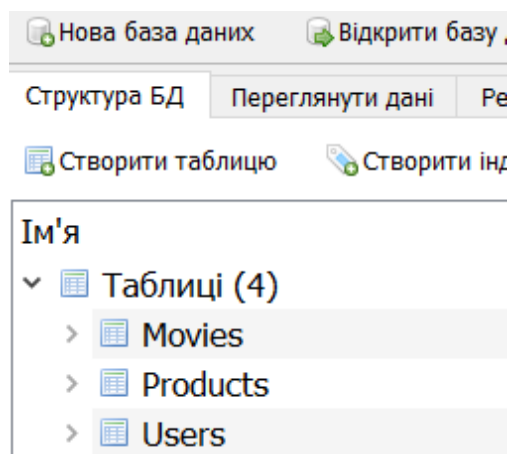


Рис. 2.2. Створені таблиці

Тепер безпосередньо перейдемо до реалізації самого додатку. Звичайно, основне його призначення – це додавання товарів, фільмів до обраного та формування замовлення, тобто створення квитка.

По-перше, так як ми використовуємо Entity Framework, нам треба створити класи моделі (класи, які представлятимуть таблиці БД як об’єкти), тобто User, Product, Movie, а також один із найголовніших класів, який буде виконувати роль підключення до БД – ApplicationContext. При створенні класу ApplicationContext був використаний патерн Singleton (одинак), що дозволить в межах програми користуватися єдиним його екземпляром, не створюючи багато зайвих.

Тепер треба створити класи, які не стосуватимуться БД, а будуть використовуватися безпосередньо в цілях роботи додатку. Це будуть класи, які представлятимуть фільм – CinemaMovie, товар – CinemaShopProduct, для яких батьківським є абстрактний CinemaProduct. Ще створимо клас який міститиме інформацію про замовлення – CinemaOrder, а також допоміжний клас, який представлятиме місце в залі – CinemaPlace. Детальніше діаграму класів показано на рис. 2.4.

Сам алгоритм полягає в тому, що при додаванні товару в обране створюється екземпляр класу CinemaShopProduct в конструктор якого передаються дані про товар. Далі отримується екземпляр класу ApplicationContext, за допомогою статичного методу GetInstance (клас реалізовано за допомогою патерна Singleton) і потім товар додається до БД і відповідно – обраного.

Що стосується додавання фільму до обраного, то при натисканні кнопки “Детальніше” відкривається вікно з детальною інформацією про фільм. Там обирається дата та час сенсу, та при можливості один з доступних тарифів (пенсіонер, учень, студент). Вартість сеансу розраховується в залежності від дня тижня, часу та тарифу. Після натискання кнопки “Обрати місце” відкривається вікно із вибором ряду та місця в залі. Далі при натисканні на кнопку “В обране” фільм додається до БД та відповідно до обраного. Слід зазначити, що на відміну від товарів, фільм до обраного можна додати лише один. Це пов’язано з тим, що квиток, який в кінці буде згенеровано буде стосуватися лише однієї людини та

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				14
Змн.	Арк.	№ докум.	Підпис	Дата		

одного фільму. Також якщо вік користувача є менший за мінімально допустимий для цього фільму, то відповідно він не додається до обраного.

Щодо обраного, то всі дані, які відображаються там “витягуються” з БД зверненням до неї з використанням запитів LINQ. Очищення обраного (видалення товарів та фільмів) відбувається обернено до запису (замість методу Add використовується – Remove).

Процес створення квитка відбувається за допомогою використання бібліотеки, що дозволяє працювати з додатком Microsoft Word за допомогою коду. Суть створення полягає в тому, що в тимчасовому файлі знаходяться певні слова, входження яких перевіряється в коді, та замінюється на ті слова, що формують собою замовлення.

Якщо говорити про реєстрацію та авторизацію, то ці процеси є аналогічними, наприклад, до процесів знаходження та додавання фільмів, розглянутих вище.

Отже, на рис. 2.3. зображено алгоритм формування замовлення та додавання до обраного товарів та фільмів.

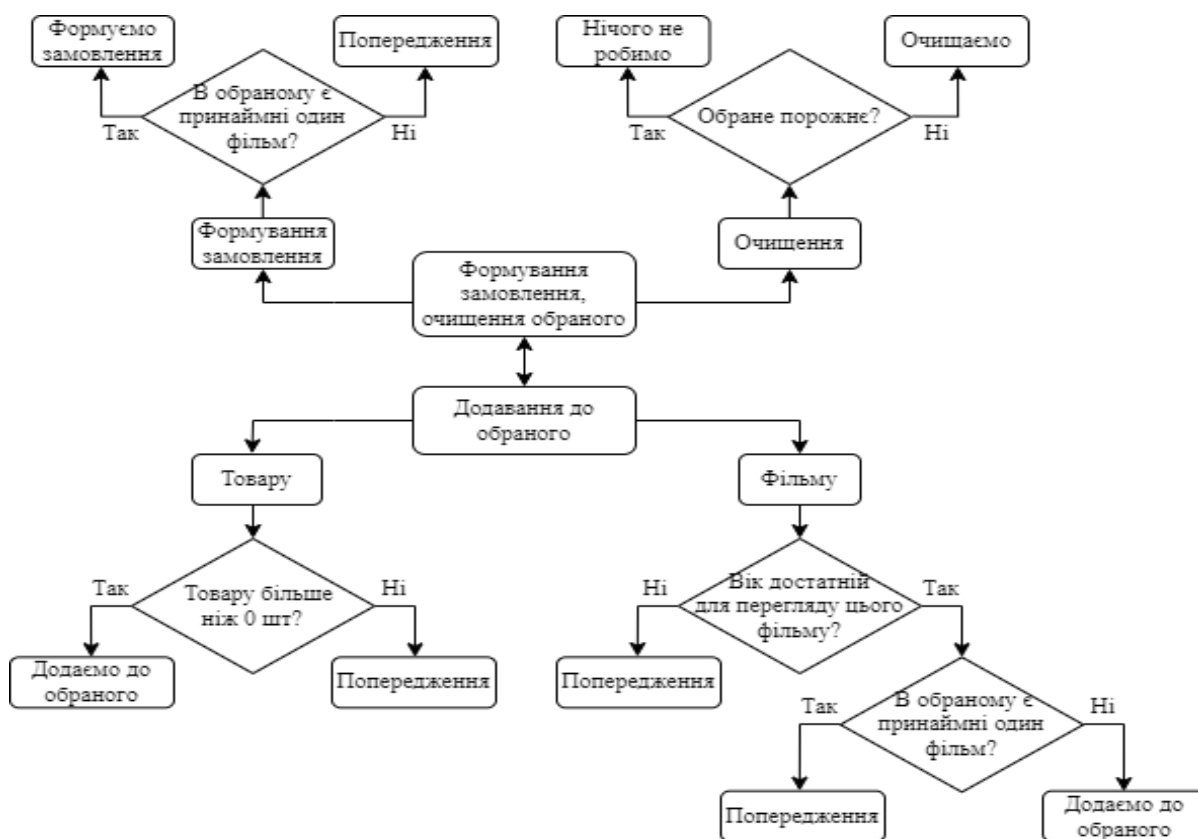


Рис. 2.3. Алгоритм формування замовлення

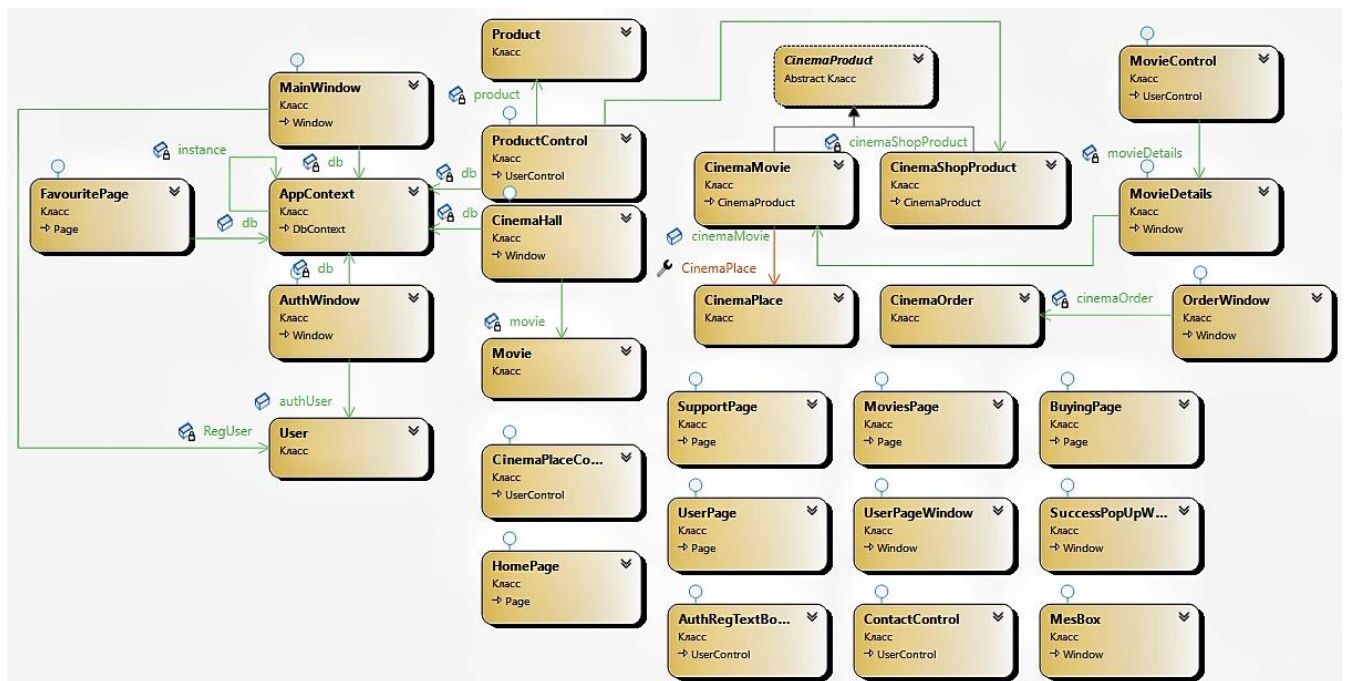


Рис. 2.4. Діаграма класів

На рис. 2.4. зображено наочну ієрархію класів та компонентів. Як бачимо, всього є 9 класів, створених нами, та 19, які автоматично створились при додаванні нових вікон (Window), сторінок (Page) та елементів керування користувача (UserControl).

2.3. Розробка програмного забезпечення

Почнемо розробку програми з підключення бази даних до проекту. Для цього після її створення та збереження, копіюємо файл БД (database.db) до нашого проекту (рис. 2.5.). Наступним кроком буде додання рядка підключення та провайдерів до файлу App.config:

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data Source=.\database.db"
    providerName="System.Data.SQLite" />
</connectionStrings>
<providers>
  <provider invariantName="System.Data.SQLite"
    type="System.Data.SQLite.EF6.SQLiteProviderServices, System.Data.SQLite.EF6"/>
  <provider invariantName="System.Data.SqlClient"
    type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer" />
  <provider invariantName="System.Data.SQLite.EF6"
    type="System.Data.SQLite.EF6.SQLiteProviderServices, System.Data.SQLite.EF6" />
</providers>
```

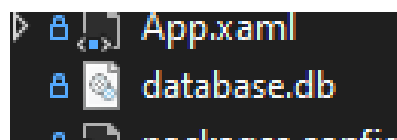


Рис. 2.5. Файл БД, доданий до проекту

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				16
Змн.	Арк.	№ докум.	Підпис	Дата		

Далі нам потрібно через меню управління пакетами NuGet додати бібліотеку System.Data.SQLite, яка дозволить в подальшому працювати із БД. Тепер створимо клас за рахунок якого ми зможемо до неї підключитись. Для його реалізації скористаємось патерном Singleton:

```
public class AppContext : DbContext
{
    private AppContext() : base("DefaultConnection") { }
    private static AppContext instance;
    private static readonly object _lock = new object();
    public static AppContext GetInstance()
    {
        if (instance == null)
        {
            lock (_lock)
            {
                if (instance == null)
                {
                    instance = new AppContext();
                }
            }
        }
        return instance;
    }
}
```

Реалізуємо в ньому три колекції на основі класу DbSet, елементами яких будуть класи моделі наших таблиць БД. Згодом в них будуть зберігатись елементи, витягнуті з конкретної таблиці:

```
public DbSet<User> Users { get; set; }
public DbSet<Product> Products { get; set; }
public DbSet<Movie> Movies { get; set; }
```

Далі коротко розглянемо як реалізовано процес реєстрації:

```
private AppContext db;
private User RegUser;
db = AppContext.GetInstance();
RegUser = new User(login, password, name, age);
db.Users.Add(RegUser);
db.SaveChanges();
```

Також при реєстрації робиться перевірка, чи є вже користувач з таким логіном в БД:

```
private bool IsUserExist()
{
    User user = null;
    user = db.Users.Where(b => b.Login ==
textBoxLogin.textBoxInput.Text.Trim()).FirstOrDefault();
    if (user != null)
        return true;
    else
        return false;
}
```

І проходить процес хешування паролю:

```
public static string GetHash(string text)
{
    var md5 = MD5.Create();
    var hash = md5.ComputeHash(Encoding.UTF8.GetBytes(text));
    return Convert.ToBase64String(hash);
}
```

Тепер розглянемо процес авторизації за логіном та паролем:

```
private AppContext db;
db = AppContext.GetInstance();
public static User authUser = null;
authUser = db.Users.Where(b => b.Login == login && b.Password ==
password).FirstOrDefault();
```

В цьому випадку ми спершу хешуємо введений пароль, а потім порівнюємо його із паролями у БД. Відповідно якщо після цього authUser не буде дорівнювати null, то користувача знайдено, інакше – ні.

Перейдемо до реалізації основних функцій програми. Розглянемо процес додавання товару до обраного. При натисненні на кнопку “В обране” спочатку робиться перевірка, чи кількість товару більша ніж 1. Якщо ні, то користувачу з’являється відповідне попередження, інакше, виконується наступний код і товар додається до обраного:

```
private AppContext db;
private Product product;
private CinemaShopProduct cinemaShopProduct;
cinemaShopProduct = new CinemaShopProduct(ProductName, int.Parse(ProductPrice),
int.Parse(TextBlockProductCount.Text));
db = AppContext.GetInstance();
product = new Product(cinemaShopProduct.Name, cinemaShopProduct.Price,
AuthWindow.authUser.id, cinemaShopProduct.Amount);
db.Products.Add(product);
db.SaveChanges();
```

Тепер розглянемо процес додавання фільму до обраного. Спочатку, коли користувач натискає на кнопку “Детальніше”, то з’являється вікно із інформацією про даний фільм. Відбувається це рахунок того, що кожен фільм (на сторінці “В прокаті”) являє собою UserControl (MovieControl) із картинкою, назвою фільму та кнопкою (рис. 2.6.). Додатково власноруч були створені властивості для даного MovieControl, тобто ті, що будуть містити деталі про конкретний фільм. Таким чином коли в XAML додається MovieControl, то в його властивостях вказується інформація про фільм:

```
<controls:MovieControl YearOfIssue="2019" MovieDuration="1год 43хв"
CountryOfProduction="США" MovieStudio="Disney" MinAge="0" MovieGenre="Сімейний,
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				18
Змн.	Арк.	№ докум.	Підпис	Дата		

```
природи" LinkTrailer="https://www.youtube.com/watch?v=Zi4LMpSDccc"
MovieImageSource="/Images/frozen.jpg" MovieName="Крижане серце 2">
</controls:MovieControl>
```

Далі після натискання кнопки “Детальніше” створюється екземпляр класу вікна MovieDetails, в елементи вікна якого присвоюється інформація, що була вказана у властивостях MovieControl та викликається метод, що відкриває дане вікно:

```
private MovieDetails movieDetails;
movieDetails = new MovieDetails();
movieDetails.ImageMovie.Source = MovieImageSource;
movieDetails.textBlockNameOfMovie.Text = MovieName;
movieDetails.textBlockMinAge.Text = minAge.ToString();
movieDetails.textBlockYearOfIssue.Text = yearOfIssue.ToString();
movieDetails.textBlockMovieDuration.Text = movieDuration;
movieDetails.textBlockCountryOfProduction.Text = countryOfProduction;
movieDetails.textBlockMovieGenre.Text = movieGenre;
movieDetails.textBlockMovieStudio.Text = movieStudio;
movieDetails.LinkTrailer = linkTrailer;
movieDetails.ShowDialog();
```

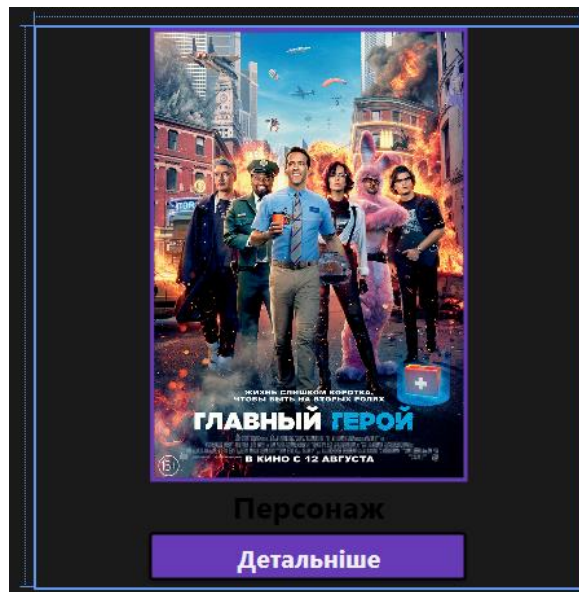


Рис. 2.6. Елемент керування користувача MovieControl

В даному вікні очікується активність користувача щодо вибору дати та часу сеансу, а також одного з тарифів (студент, школяр, пенсіонер), що є не обов’язковим. По цим параметрах буде визначатися ціна за сеанс:

```
double priceOfSession;
bool isAnyChecked = false;
if (radioButtonPension.IsChecked == true || radioButtonSchoolChild.IsChecked == true
|| radioButtonStudent.IsChecked == true)
isAnyChecked = true;
DateTime dateTime = datePickerDate.SelectedDate.Value;
if (dateTime.DayOfWeek == DayOfWeek.Sunday || dateTime.DayOfWeek ==
DayOfWeek.Saturday)
{
    if (comboBoxTime.SelectedIndex >= 3)
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				19
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            if (isAnyChecked)
                priceOfSession = 100;
            else
                priceOfSession = 110;
        }
        else
        {
            if (isAnyChecked)
                priceOfSession = 80;
            else
                priceOfSession = 90;
        }
    }
}

else
{
    if (comboBoxTime.SelectedIndex >= 3)
    {
        if (isAnyChecked)
            priceOfSession = 80;
        else
            priceOfSession = 90;
    }
    else
    {
        if (isAnyChecked)
            priceOfSession = 60;
        else
            priceOfSession = 70;
    }
}
}

```

Далі при натисненні на кнопку “Обрати місце” проходить перевірка, чи обрано дату та час. Якщо ні, то з’являється попередження, якщо так – перевіряється, чи вік користувача є допустимим для цього фільму. Якщо результат позитивний, то створюється екземпляр класу, що представляє собою фільм, в який записуються обрані дані, та відкривається вікно із вибором місця:

```

public static CinemaMovie cinemaMovie = null;
string plan = null;
if (radioButtonPension.IsChecked == true)
    plan = "Пенсіонер";
else if (radioButtonSchoolChild.IsChecked == true)
    plan = "Школяр";
else if (radioButtonStudent.IsChecked == true)
    plan = "Студент";
cinemaMovie = new CinemaMovie(textBlockNameOfMovie.Text, MakePriceOfSession(),
datePickerDate.SelectedDate.Value, comboBoxTime.Text, plan);
this.Close();
CinemaHall cinemaHall = new CinemaHall();
cinemaHall.ShowDialog();

```

Тут при натисненні на кнопку “В обране” робиться перевірка, чи було обрано ряд та місце. Якщо ні, то з’явиться відповідне попередження, якщо так – так виконається перевірка, чи є вже принаймні один фільм в обраному:

```

private AppContext db;
db = AppContext.GetInstance();
List<Movie> movies = db.Movies.Where(b => b.User_Id == AuthWindow.authUser.id).ToList();

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				20
Змн.	Арк.	№ докум.	Підпис	Дата		

Відповідно якщо `movies.Count >= 0`, то з'явиться попередження, що не можна додати більше одного фільму. Інакше, ми звертаємось до об'єкту, створеного в попередньому вікні та записуємо в його властивість, що представляє собою місце в залі, обране користувачем місце та додаємо даний фільм до БД та обраного:

```
private Movie movie;
MovieDetails.cinemaMovie.CinemaPlace = new CinemaPlace(comboBoxRow.SelectedIndex + 1,
comboBoxColumn.SelectedIndex + 1);
movie = new Movie(MovieDetails.cinemaMovie.Name,
MovieDetails.cinemaMovie.TimeOfSession, MovieDetails.cinemaMovie.DateOfSession.Day,
MovieDetails.cinemaMovie.DateOfSession.Month,
MovieDetails.cinemaMovie.DateOfSession.Year, MovieDetails.cinemaMovie.CinemaPlace.Row,
MovieDetails.cinemaMovie.CinemaPlace.Column, MovieDetails.cinemaMovie.Price,
AuthWindow.authUser.id, MovieDetails.cinemaMovie.Plan);
db.Movies.Add(movie);
db.SaveChanges();
```

Перейдемо до сторінки з обраним. Для того, щоб структуровано показати список із усіма товарами створимо в XAML шаблон, по якому буде відбуватись вивід кожного з них:

```
<ListView>
    <ListView.ItemTemplate>
        <DataTemplate>
            <WrapPanel>
                <TextBlock Text="{Binding Name}"></TextBlock>
                <StackPanel>
                    <TextBlock Text="{Binding Price}"></TextBlock>
                    <TextBlock Text="грн"></TextBlock>
                </StackPanel>
                <TextBlock Text="{Binding Amount}"></TextBlock>
                <TextBlock Text="шт"></TextBlock>
            </WrapPanel>
        </DataTemplate>
    </ListView.ItemTemplate>
</ListView>
```

Тепер зробимо це ж саме для виводу фільмів:

```
<ListView>
    <ListView.ItemTemplate>
        <DataTemplate>
            <WrapPanel>
                <TextBlock Text="{Binding Name}"></TextBlock>
                <StackPanel>
                    <TextBlock Text="{Binding Price}"></TextBlock>
                    <TextBlock Text="грн"></TextBlock>
                </StackPanel>
                <StackPanel>
                    <TextBlock Text="{Binding DayOfSession}"></TextBlock>
                    <TextBlock Text="."></TextBlock>
                    <TextBlock Text="{Binding MonthOfSession}"></TextBlock>
                    <TextBlock Text="."></TextBlock>
                    <TextBlock Text="{Binding YearOfSession}"></TextBlock>
                </StackPanel>
                <TextBlock Text="{Binding TimeOfSession}"></TextBlock>
            </WrapPanel>
        </DataTemplate>
    </ListView.ItemTemplate>
</ListView>
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				21
Змн.	Арк.	№ докум.	Підпис	Дата		

```
</ListView.ItemTemplate>
</ListView>
```

Тепер перейдемо до коду. Створимо список з фільмів та товарів, та за допомогою запитів LINQ “витягнемо” їх із БД. Кожен запис має поле, яке є ідентифікатором конкретного користувача. Таким чином, ми можемо відібрати тільки ті записи, які належать саме йому:

```
public static AppContext db;
public static List<Product> products;
public static List<Movie> movies;
products = db.Products.Where(b => b.User_Id == AuthWindow.authUser.id).ToList();
movies = db.Movies.Where(b => b.User_Id == AuthWindow.authUser.id).ToList();
```

Тепер, якщо розмір списку не дорівнює нулю, тобто є принаймні один запис, то ми ці дані записуємо в шаблон, розглянутий вище.

При очищенні обраного також проходить перевірка, чи існують записи.

Якщо так, то видаляємо їх:

```
foreach (Movie movie in movies)
    db.Movies.Remove(movie);
db.SaveChanges();
foreach (Product product in products)
    db.Products.Remove(product);
db.SaveChanges();
```

Перейдемо до формування замовлення та створення квитка. Слід зазначити, що якщо користувач не обрав ні одного фільму, то при натисненні на кнопку “Сформувати замовлення” з’явиться відповідне попередження. Інакше, обирається спосіб оплати, генерується квиток та очищається обране.

Метод, який шукає входження слова в документі та замінює його на інше:

```
private void FindAndReplace(Word.Application wordApp, object toFindText, object
replaceWithText)
{
    object matchCase = true;
    object matchWholeWord = true;
    object matchWildCards = false;
    object matchSoundLike = false;
    object nmatchAllforms = false;
    object forward = true;
    object format = false;
    object matchKashida = false;
    object matchDiacritics = false;
    object matchAlefHamza = false;
    object matchControl = false;
    object read_only = false;
    object visible = true;
    object replace = 2; object wrap = 1;
    wordApp.Selection.Find.Execute(ref toFindText, ref matchCase, ref
matchWholeWord, ref matchWildCards, ref matchSoundLike, ref nmatchAllforms, ref
forward, ref wrap, ref format, ref replaceWithText, ref replace, ref matchKashida, ref
matchDiacritics, ref matchAlefHamza, ref matchControl);
}
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				22
Змн.	Арк.	№ докум.	Підпис	Дата		

Загальний алгоритм полягає в тому, що ми за допомогою коду працюємо з додатком Microsoft Word та, замінюючи слова, формуємо квиток. Детальніше цей процес наведено в додатку А (OrderWindow.xaml.cs).

Висновки до другого розділу:

В другому розділі було розглянуто основний та функціональний алгоритми роботи програми, спроектовано схеми та діаграму класів. Описано основний функціонал додатку, пояснено алгоритми важливих методів та наведено їх код. Роз'яснено деякі нюанси роботи програми.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				23
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ОПИС РОБОТИ З ПРОГРАМНИМ ДОДАТКОМ ТА ЙОГО ТЕСТУВАННЯ

3.1. Опис роботи з програмним додатком

Після запуску програми ми бачимо вікно реєстрації (рис. 3.1.), на якому розміщені поля вводу та кнопки. Якщо ми ще не маємо акаунту, то можемо його створити. В разі помилки введення поле підсвітиться червоним кольором та при наведенні з'явиться підказка.

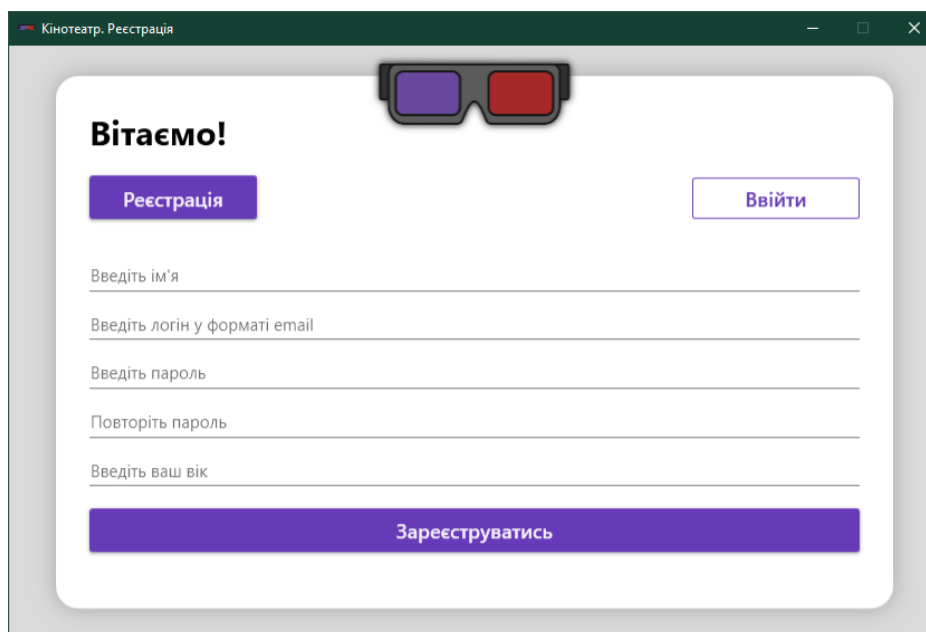


Рис. 3.1. Вікно реєстрації

Після реєстрації ми переносимось до вікна авторизації (рис. 3.2.) де вводимо свої дані та входимо, натискаючи кнопку “Ввійти”, що знаходиться під текстовими полями.

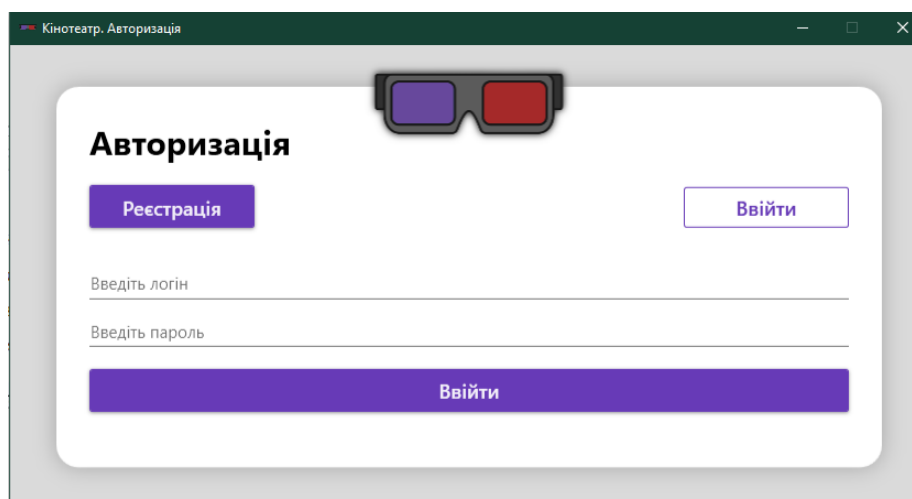


Рис. 3.2. Вікно авторизації

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				24
Змн.	Арк.	№ докум.	Підпис	Дата		

Аналогічно в разі помилки введення ми будемо про це повідомлені зміною кольору текстового поля та підказкою.

Після входу, з'являється головне вікно нашого додатку з відкритою домашньою сторінкою, де знаходиться текст з привітанням, а також певна актуальна інформація, яка може бути корисною (рис. 3.3.). З лівої сторони знаходиться навігаційне меню, яке можна розкрити, щоб ознайомитися із усіма можливими наступними діями (рис. 3.4).

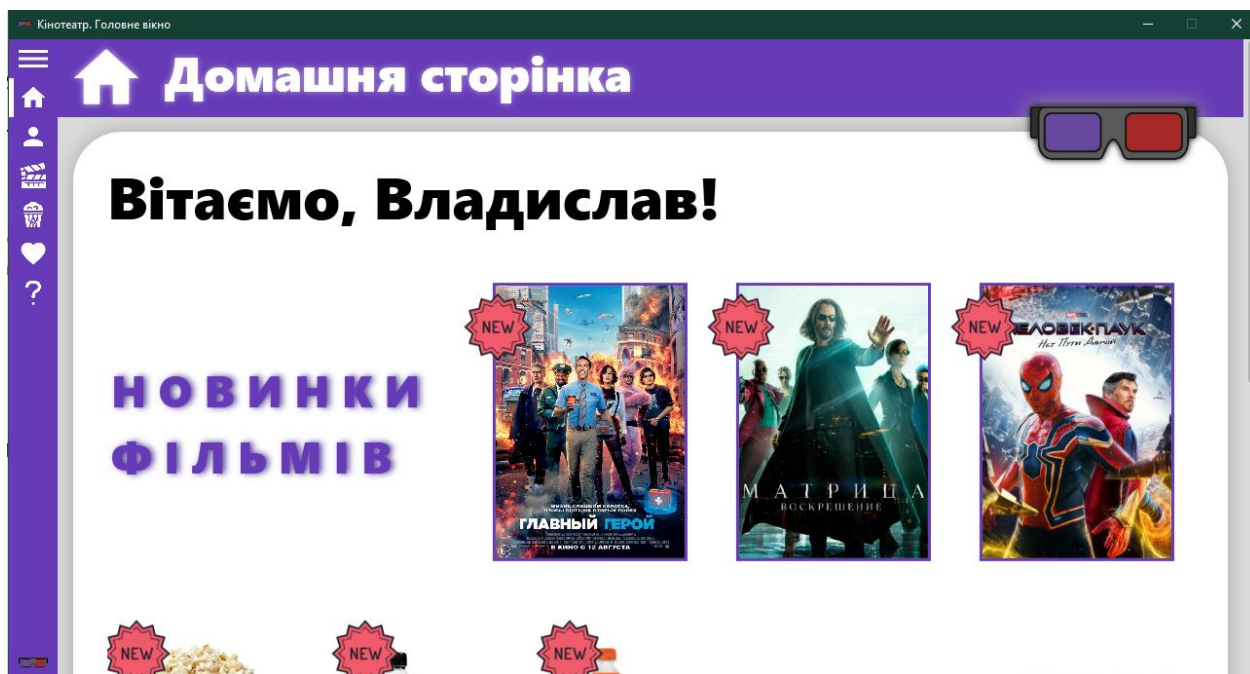


Рис. 3.3. Головне вікно додатку

Меню включає в себе 6 пунктів. Натиснувши на один з них, в нас відкриється відповідна сторінка.

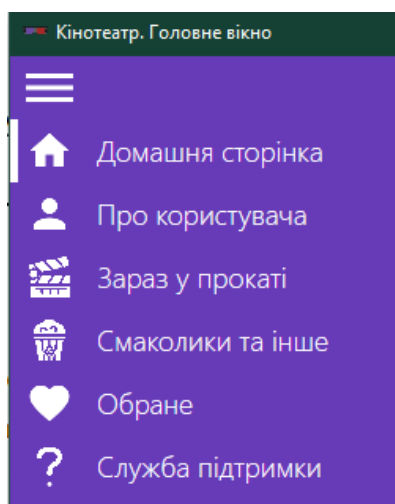


Рис. 3.4. Навігаційне меню

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк. 25
		Олег ВЛАСЕНКО				
Змн.	Арк.	№ докум.	Підпис	Дата		

Відкриємо сторінку “Про користувача”. Тут ми можемо побачити логін, ім’я, а також вік, які ми ввели при реєстрації (рис. 3.5.).

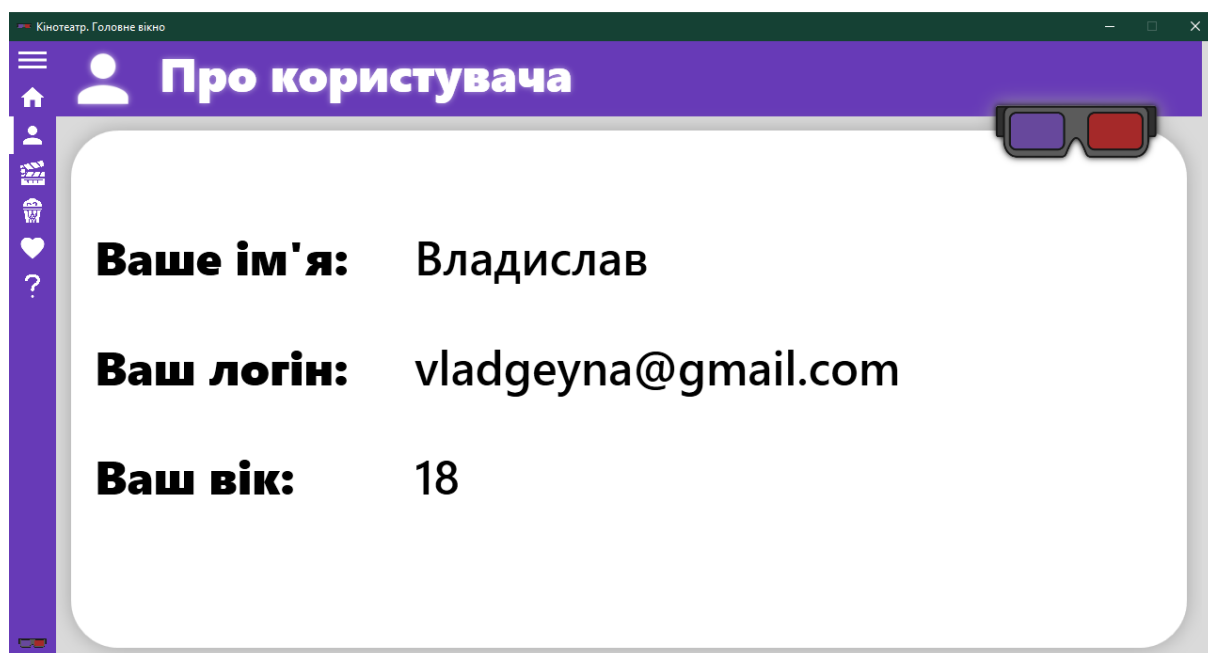


Рис. 3.5. Вигляд сторінки “Про користувача”

На останній сторінці знаходиться служба підтримки із контактами (рис. 3.6.).

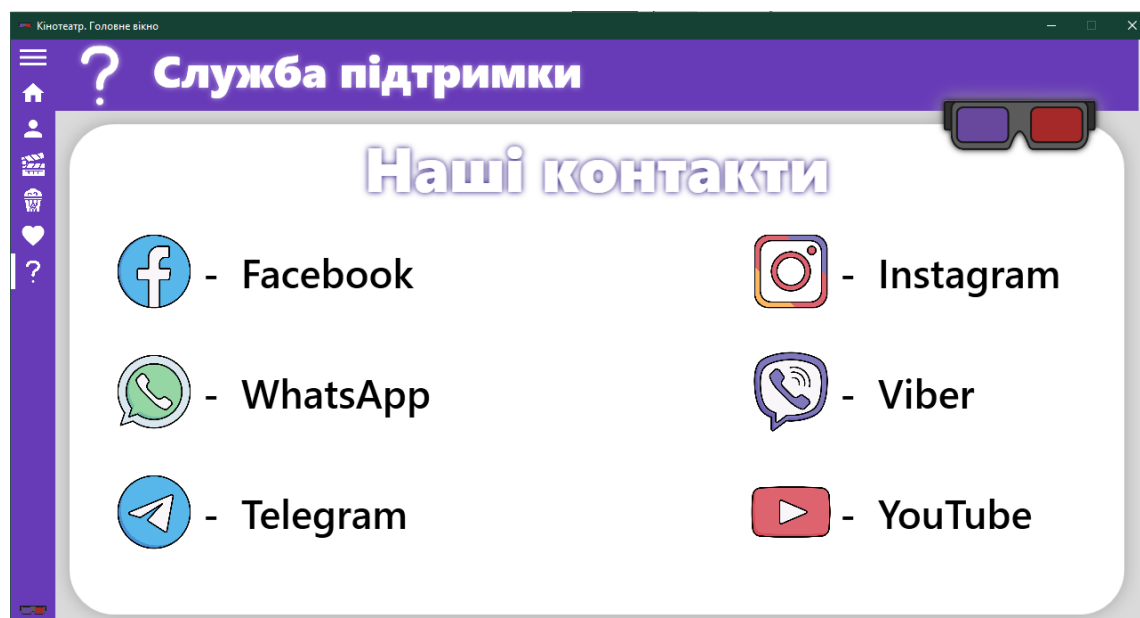


Рис. 3.6. Вигляд сторінки “Служба підтримки”

Перейдемо до сторінки “Смаколики та інше”, та спробуємо додати товар. Як бачимо, кількість товару ми можемо задавати або ввівши число з клавіатури, або за допомогою кнопок “+” та “-”. Спробуємо це зробити (рис. 3.7.).

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				26
Змн.	Арк.	№ докум.	Підпис	Дата		

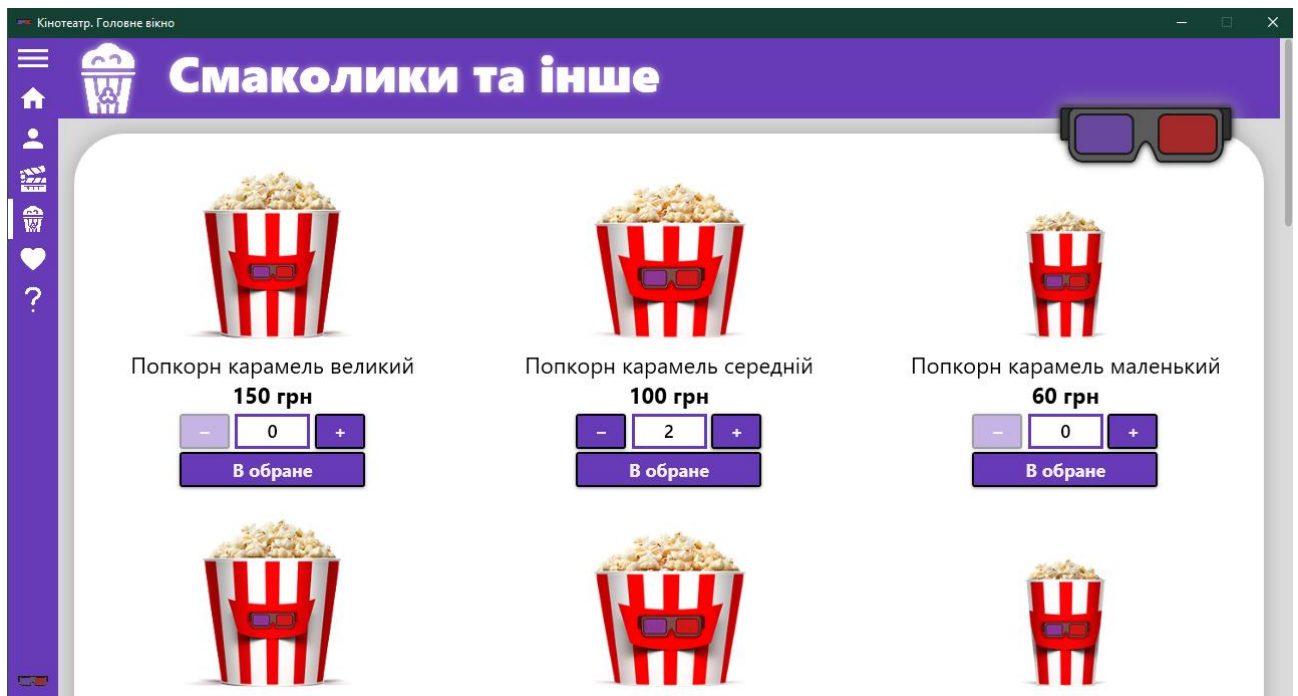


Рис. 3.7. Додавання запису

Як бачимо в текстовому полі товару “Попкорн карамель середній” з’явилося число “2”. Натиснемо на кнопку “В обране” та в разі успіху побачимо коротку анімацію, на місці товару (рис. 3.8.).

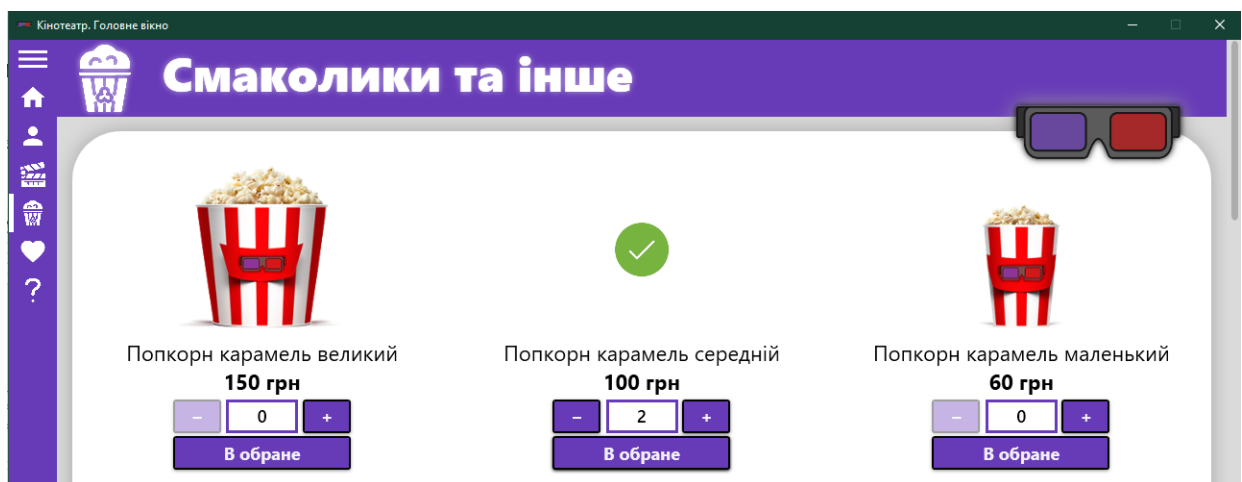


Рис. 3.8. Товар успішно додався до обраного

Перейдемо до сторінки “Обране” та перевіримо, чи з’явився там товар (рис. 3.10.), адже до цього вона була порожньою (рис. 3.9.).

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк. 27
		Олег ВЛАСЕНКО				
Змн.	Арк.	№ докум.	Підпис	Дата		

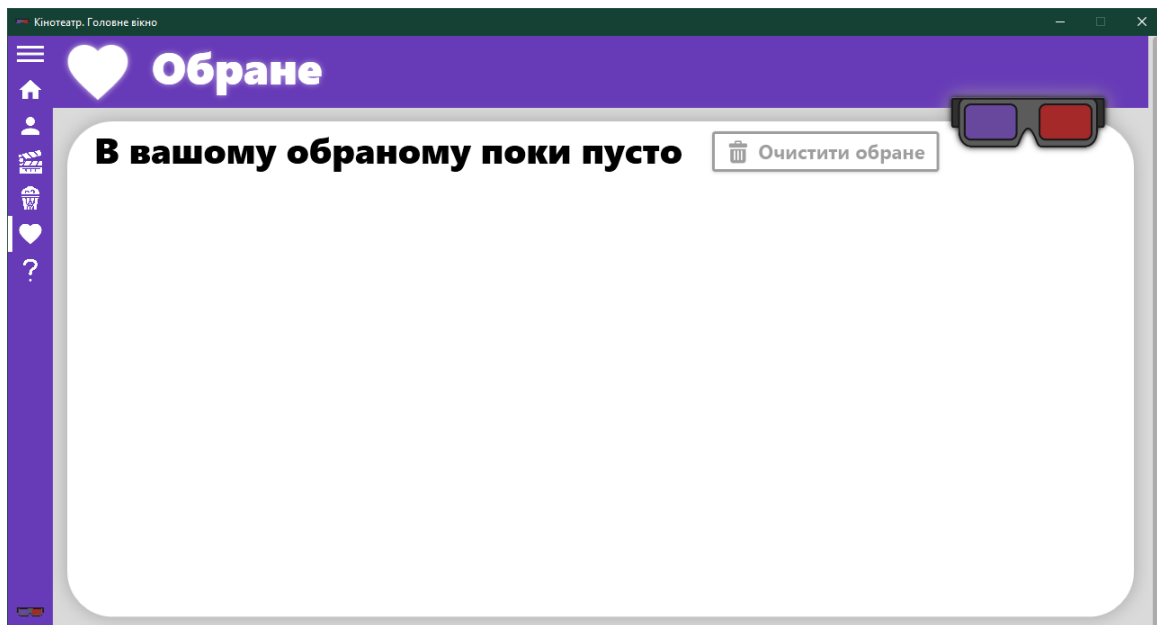


Рис. 3.9. Вигляд сторінки “Обране” без товарів

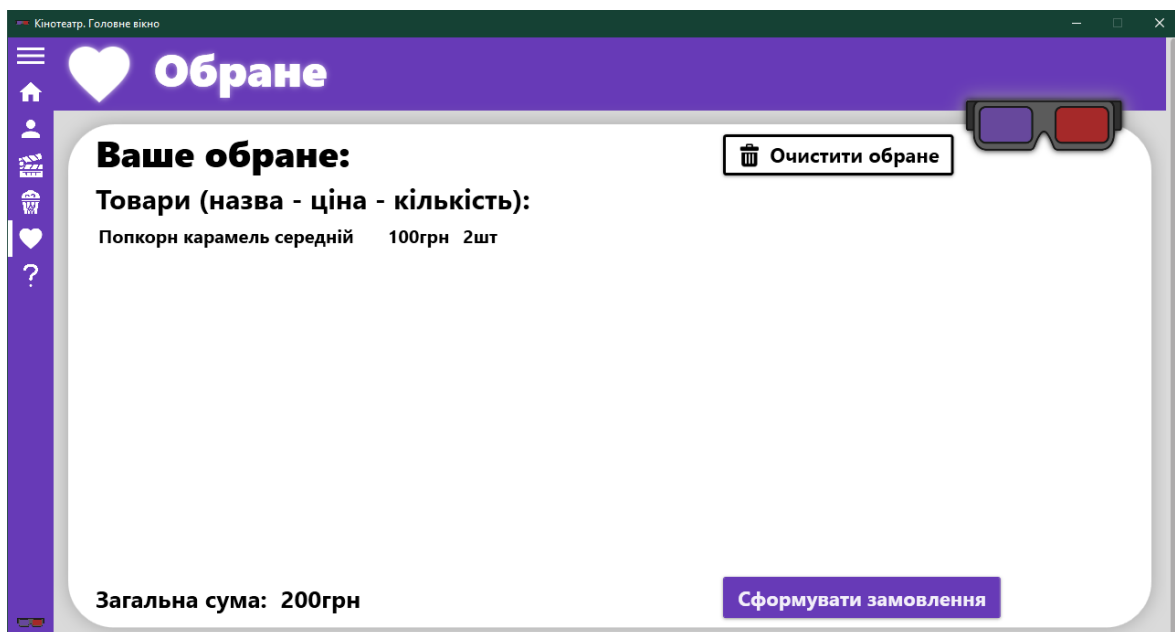


Рис. 3.10. Вигляд сторінки “Обране” після додавання товару

Для того, щоб ми могли сформувати замовлення, нам треба обов’язково додати фільм, отже, зробимо це. Відкриємо сторінку “Зараз у прокаті” (рис. 3.11.) та нажмемо на кнопку “Детальніше”.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				28
Змн.	Арк.	№ докум.	Підпис	Дата		

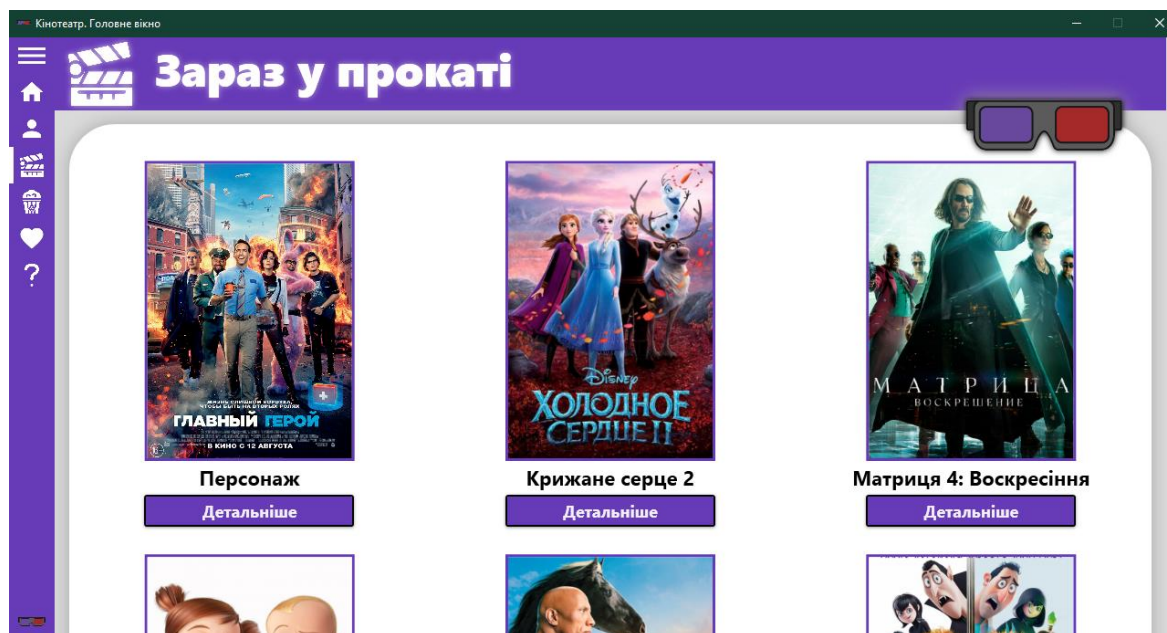


Рис. 3.11. Вигляд сторінки “Зараз у прокаті”

Бачимо, що з’явилося вікно із інформацією про фільм. Тут ми маємо обрати дату та час сеансу, а також при бажанні один з доступних тарифів (рис. 3.12.). Ще тут можна переглянути кінцеву ціну на обраний сеанс із врахуванням тарифу, натиснувши на кнопку “Оновити вартість”. Окрім інформації, даної в вікні, можна переглянути трейлер, натиснувши на кнопку “Дивитись трейлер”.

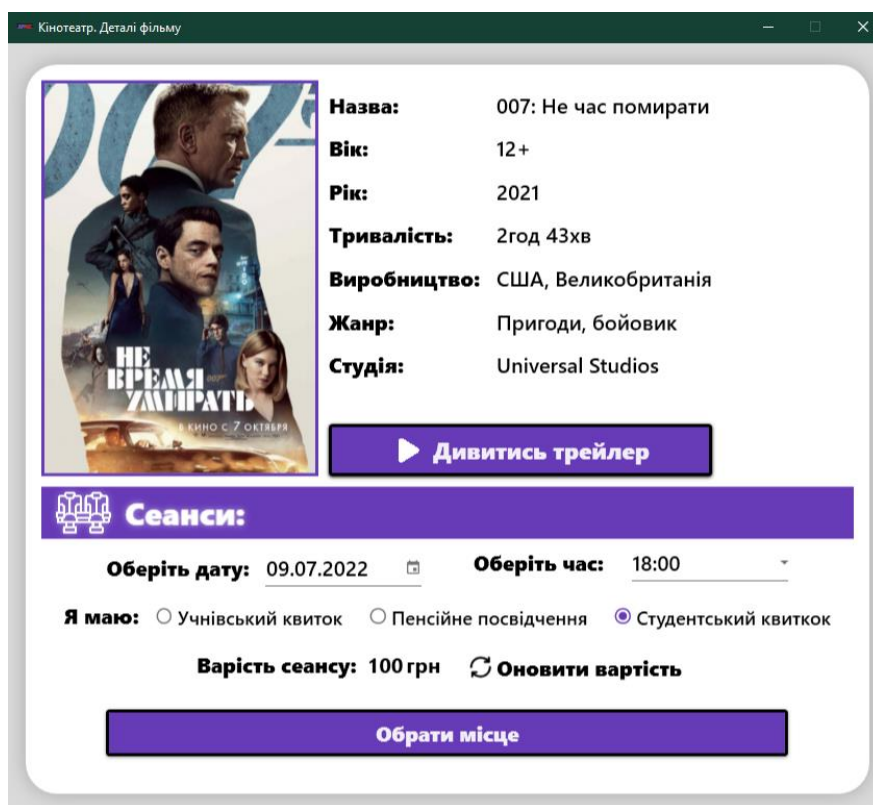


Рис. 3.12. Вікно з детальною інформацією про фільм

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				
Змн.	Арк.	№ докум.	Підпис	Дата		29

Далі переходимо до вибору місця. Натискаємо на кнопку “Обрати місце” та потрапляємо до відповідного вікна (рис. 3.13.). Тут для зручності, при наведенні на кожне місце, з’являється підказка, де точно вказується його розташування. Знизу ми можемо побачити два випадаючі списки, де ми відповідно можемо обрати ряд та місце.

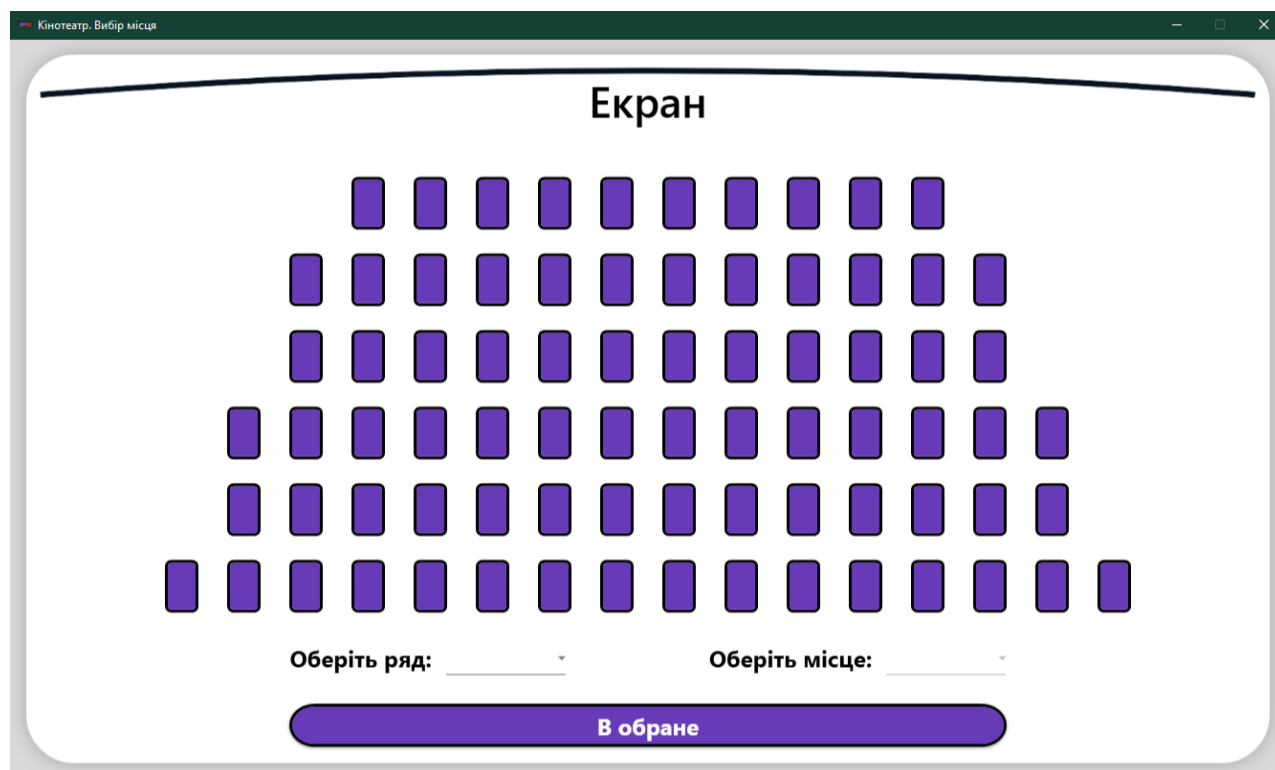


Рис. 3.13. Вікно з вибором місця в залі

Після вибору натискаємо на кнопку “В обране”, щоб додати фільм до обраного. В разі успіху бачимо відповідне спливаюче вікно (рис. 3.14.).



Рис. 3.14. Фільм успішно доданий до обраного

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				30
Змн.	Арк.	№ докум.	Підпис	Дата		

Перейдемо до сторінки “Обране” та перевіримо, чи з’явився там, щойно доданий нами фільм (рис. 3.15.).

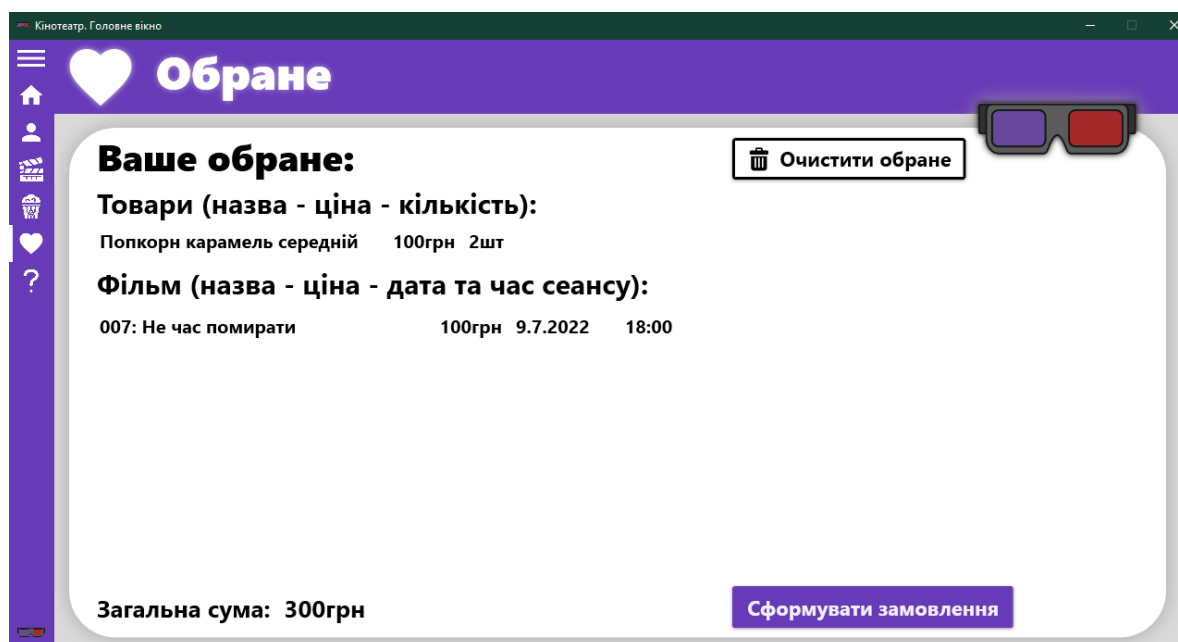


Рис. 3.15. Вигляд сторінки “Обране” після додавання товару та фільму

Як бачимо на даній сторінці присутній товар, доданий нами раніше та фільм, який ми тільки що додали. В кожному пункті обраного ми можемо окремо побачити його ціну, а в товару додатково ще й кількість. Зліва знизу ми можемо побачити загальну суму замовлення. При бажанні змінити свій вибір щодо фільму та товарів ми можемо очистити обране, натиснувши на кнопку зверху “Очистити обране”.

Сформуємо замовлення. Натиснемо на кнопку “Сформувати замовлення” та побачимо відповідне вікно де нам потрібно обрати спосіб оплати та погодитись з правилами (рис. 3.16.).

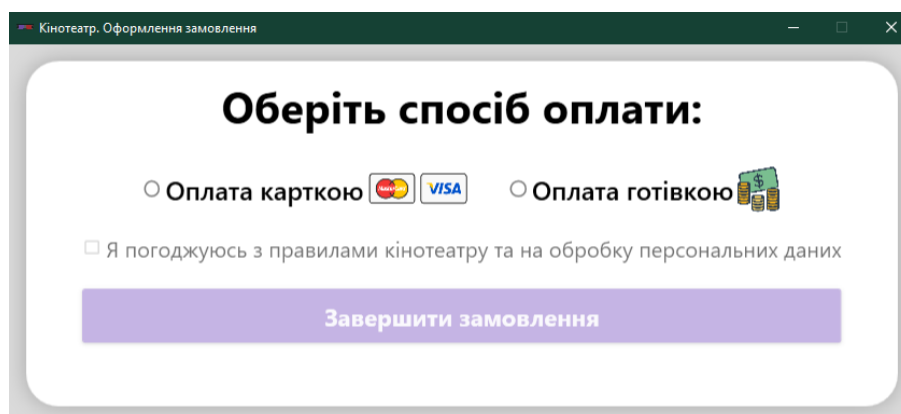


Рис. 3.16. Вигляд сторінки “Обране” після додавання товару та фільму

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк. 31
		Олег ВЛАСЕНКО				
Змн.	Арк.	№ докум.	Підпис	Дата		

Після вибору та погодження натискаємо на кнопку “Завершити замовлення”. У разі успіху бачимо спливаюче вікно, що квиток було створено, після чого обране самостійно очищується. Переглянемо щойно створений квиток. Він знаходиться в папці “Документи”. Бачимо, що тут є усі дані, які були зібрані під час формування замовлення (рис. 3.17).

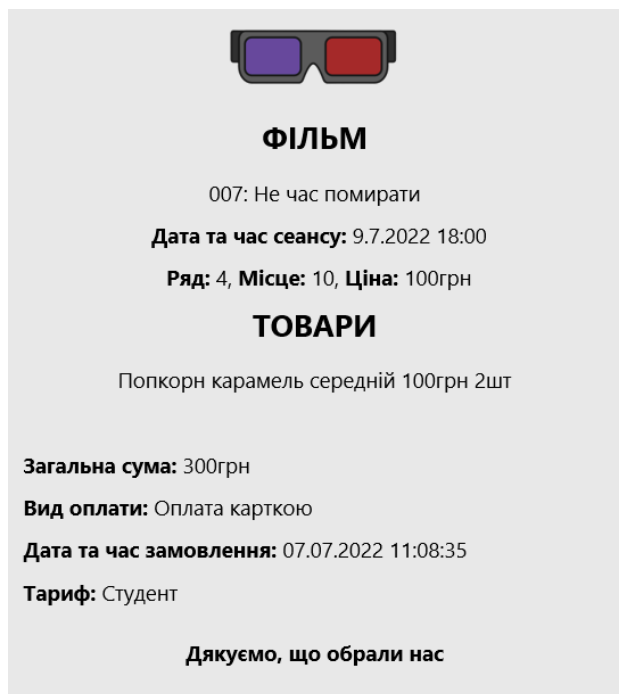


Рис. 3.17. Вигляд квитка, що створився після завершення замовлення

3.2 Тестування роботи програмного забезпечення

Програма передбачає помилки при введенні, а також вибору різних пунктів користувачем, для зручного користування програмою. Отож, перевіримо готовий додаток на коректність усіх функцій.

Наприклад, при реєстрації та авторизації при некоректно введених даних відповідне поле підсвічується червоним кольором (рис. 3.18).

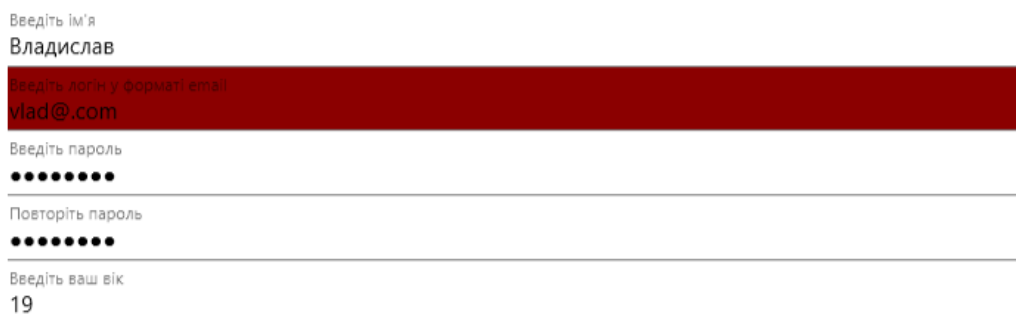


Рис. 3.18. Некоректне введення даних

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				32
Змн.	Арк.	№ докум.	Підпис	Дата		

Також якщо при реєстрації буде виявлено, що такий користувач вже є або навпаки, при авторизації, що такого не існує, з'являться відповідні спливаючі вікна (рис. 3.18., рис. 3.20.).

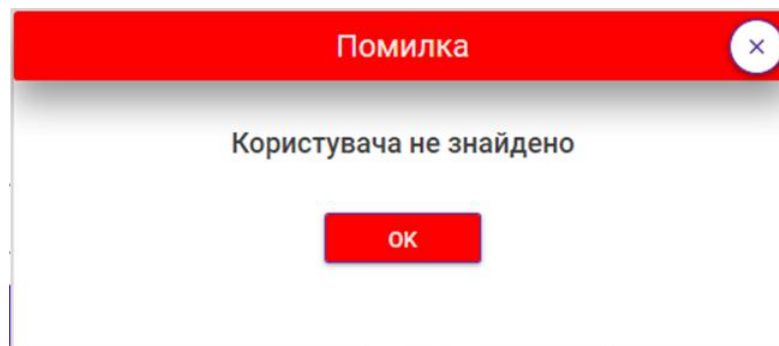


Рис. 3.19. Помилка при авторизації

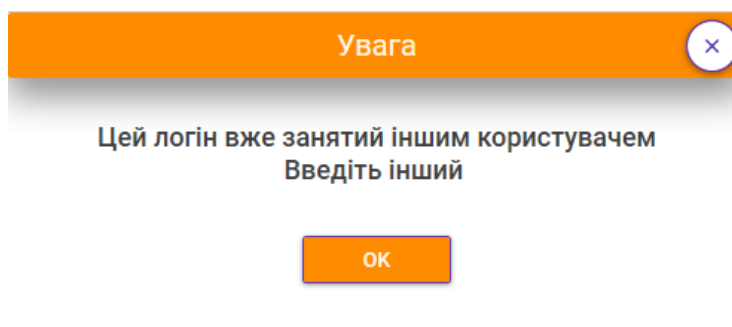


Рис. 3.20. Попередження при реєстрації

Якщо користувач спробує додати товар кількістю 0 штук він отримає наступне попередження:

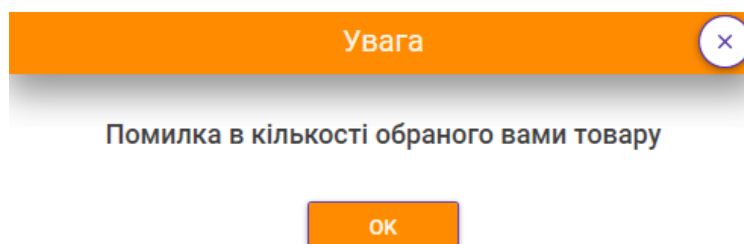


Рис. 3.21. Попередження при додаванні товару до обраного

Аналогічно, якщо при виборі дати та часу сеансу, користувач захоче оновити ціну, не обравши якогось з пунктів, йому буде про це повідомлено (рис. 3.22.). Це ж саме відбудеться коли він буде переходити до вибору місця або додавати фільм до обраного, не обравши ряд та місце в залі.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				33
Змн.	Арк.	№ докум.	Підпис	Дата		

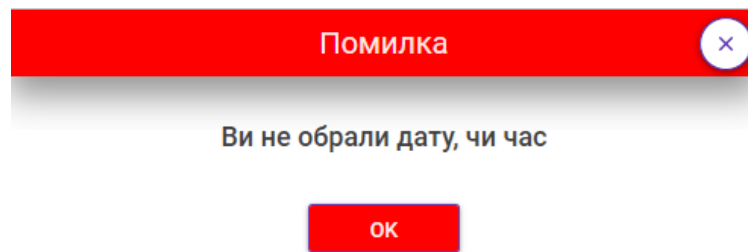


Рис. 3.22. Помилка при оновленні ціни

Також, якщо вік користувача недостатньо великий для перегляду даного фільму, він не зможе обрати його:

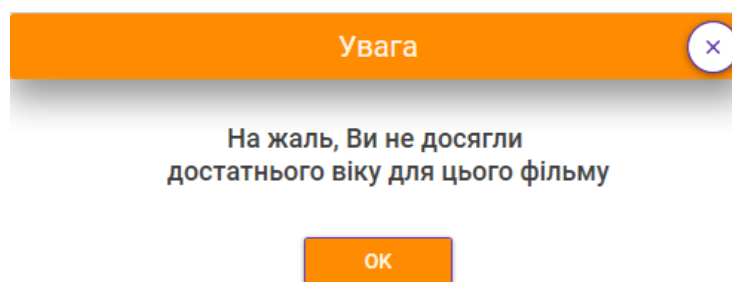


Рис. 3.23. Попередження про недостатній вік для фільму

Оскільки до обраного можна додати лише один фільм, то при спробі додати більше, користувача буде повідомлено наступним чином:

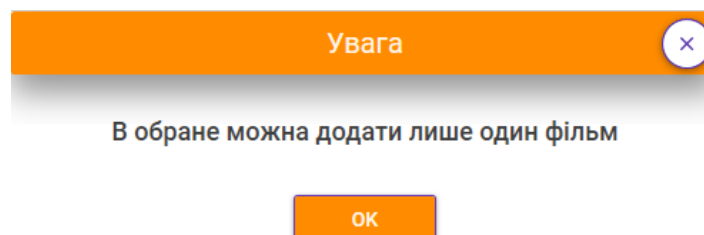


Рис. 3.24. Попередження про додавання більше одного фільму до обраного

Якщо буде спроба оформити замовлення, не обравши жодного фільму, з'явиться попередження про це (рис. 3.24.).

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				34
Змн.	Арк.	№ докум.	Підпис	Дата		

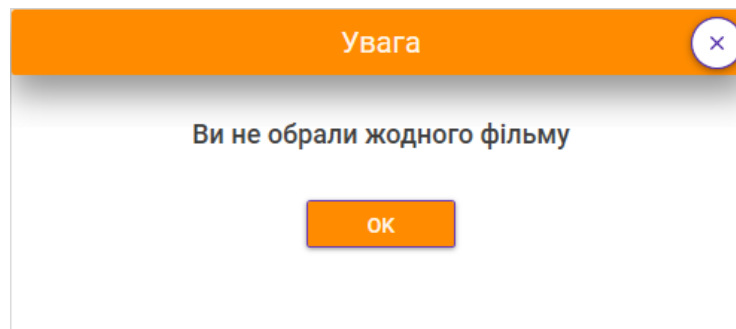


Рис. 3.24. Попередження при оформленні замовлення

Також задля уникнення помилок, текстові поля, які мають приймати лише числа, ігнорують введення всього окрім них. Це було реалізовано завдяки регулярним виразам.

Висновки до третього розділу

У третьому розділі було детально описано роботу програмного додатку, очікувані дії користувача та їх результати. Було детально протестовано роботу програми, розписано моменти, де ймовірно могла би виникнути помилка, що дозволило уникнути багатьох неправильних та нелогічних дій користувача, а також серйозних помилок. Для кращого розуміння до кожного пункту було додано відповідну ілюстрацію у вигляді скріншоту.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				35
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В результаті виконання курсового проекту було отримано навички роботи з середовищем розробки Windows Presentation Foundation, мовою програмування C#, ORM інструментом Entity Framework та БД. Було створено програму «Каса кінотеатру», яка фактично не має жодних аналогів, у вигляді десктопних додатків.

У першому розділі були розглянуті аналоги представлені у вигляді веб-сайтів зі схожою тематикою. Було вирішено яким має бути функціонал, проаналізовано та визначено, яку структуру повинна мати майбутня програма та де вона буде застосовуватись.

У другому розділі було розглянуто та спроектовано загальну схему та алгоритм програми, а також діаграму класів. Було визначено та детально описано можливості програми. Розглянуто структуру бази даних, пояснено алгоритми основних методів та фрагментів коду програми та приведено схему формування замовлення та додавання до обраного товарів та фільмів.

У третьому розділі було детально описано роботу програмного додатку та як ним користуватись, очікувані дії користувача та їх результати. Проведено тестування роботи програмного забезпечення, задля уникнення хибних дій користувача та помилок. Були додані ілюстрації у вигляді скріншотів для кращого розуміння написаного.

Отже, в результаті виконання курсового проекту було створено справну робочу програму, яка виконує усі бажані та потрібні функції, коректно працює при введенні різних значень та непередбачуваних дій користувача.

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				36
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. METANIT.COM, Руководство по WPF [Електронний ресурс] – Режим доступу до ресурсу: <https://metanit.com/sharp/wpf/>.
2. Мак-Дональд М. WPF. Windows Presentation Foundation в .NET 3.5 / с примерами на C# 2008 для профессионалов / Мэтью Мак-Дональд.
3. C# documentation [Електронний ресурс] – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/csharp/>.
4. METANIT.COM, Введение в C# [Електронний ресурс] – Режим доступу до ресурсу: <https://metanit.com/sharp/tutorial/1.1.php>.
5. Об'єктно-орієнтоване програмування [Електронний ресурс] / Освітній портал ДУ «Житомирська політехніка» – 2022. – Режим доступу: <https://learn.ztu.edu.ua/course/view.php?id=4035>.
6. Шилдт Герберт C# 4.0: полное руководство.: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2011. — 1056 с.: ил.
7. SQL Tutorial [Електронний ресурс] – Режим доступу: <https://www.w3schools.com/sql/>.
8. DB Browser for SQLite [Електронний ресурс] – Режим доступу: <https://sqlitebrowser.org/>.
9. SQLite Documentation [Електронний ресурс] – Режим доступу: <https://www.sqlite.org/docs.html>.
10. Рихтер Д. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C# / Джеффри Рихтер. – Питер, 2013. – 896 с. – (4-е изд).
11. Фленов М. Библия C# / Михаил Фленов. – Питер: БХВ-Петербург, 2011. – 541 с. – (2-е изд).
12. Material Design In XAML Toolkit [Електронний ресурс] – Режим доступу: <http://materialdesigninxaml.net/>.

ДОДАТКИ

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				38
Змн.	Арк.	№ докум.	Підпис	Дата		

Вихідний код додатку

Файл MainWindow.xaml.cs

```

using System;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Text.RegularExpressions;
using System.Windows;
using System.Windows.Input;
using System.Windows.Media;

namespace usersApp.Windows
{
    public partial class MainWindow : Window
    {
        private AppContext db;
        private User RegUser;
        public MainWindow()
        {
            InitializeComponent();
            db = AppContext.GetInstance();

            private void Button_Reg_Click(object sender, RoutedEventArgs e)
            {
                string login = textBoxLogin.textBoxInput.Text.Trim();
                string password = textBoxPassword.Password.Trim();
                string passwordConfirm = textBoxPasswordConfirm.Password.Trim();
                string name = textBoxName.textBoxInput.Text.Trim();
                bool checkAge = int.TryParse(textBoxAge.textBoxInput.Text.Trim(), out int
age);
                Regex regex = new Regex("^([a-z0-9_-]+\\.)*[a-z0-9_-]+@[a-z0-9_-]+(\\.([a-
z0-9_-]+)*\\.([a-z]{2,6})$)", RegexOptions.Compiled);
                Match match = regex.Match(login);
                if (name.Length <= 1)
                {
                    textBoxName.textBoxInput.ToolTip = "Некоректно введено ім'я";
                    textBoxName.textBoxInput.Background = Brushes.DarkRed;
                    return;
                }
                else if (!match.Success)
                {
                    textBoxLogin.textBoxInput.ToolTip = "Некоректно введений логін";
                    textBoxLogin.textBoxInput.Background = Brushes.DarkRed;
                    return;
                }
                else if (password.Length < 5)
                {
                    textBoxPassword.ToolTip = "Мінімальна довжина пароля 5 символів";
                    textBoxPassword.Background = Brushes.DarkRed;
                    return;
                }
                else if (passwordConfirm != password)
                {
                    textBoxPasswordConfirm.ToolTip = "Ви помилилися при другому введені
пароля";
                    textBoxPasswordConfirm.Background = Brushes.DarkRed;
                    return;
                }
                else if (checkAge == false)

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				39
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    {
        textBoxAge.textBoxInput.ToolTip = "Некоректно введений вік";
        textBoxAge.textBoxInput.Background = Brushes.DarkRed;
        return;
    }
    else if (age < 6 || age > 100)
    {
        textBoxAge.textBoxInput.ToolTip = "Вік занадто малий або великий";
        textBoxAge.textBoxInput.Background = Brushes.DarkRed;
        return;
    }
    else if (IsUserExist())
    {
        textBoxLogin.textBoxInput.ToolTip = "Цей логін вже зайнятий іншим користувачем. Введіть інший";
        textBoxLogin.textBoxInput.Background = Brushes.DarkRed;
        new MessageBox($"Цей логін вже зайнятий іншим користувачем\n{"Введіть інший", 41}", MessageType.Warning, MessageButtons.Ok).ShowDialog();
        return;
    }
    else
    {
        password = GetHash(password);
        RegUser = new User(login, password, name, age);
        db.Users.Add(RegUser);
        db.SaveChanges();
        new MessageBox("Реєстрація пройшла успішно", MessageType.Success, MessageButtons.Ok).ShowDialog();
        AuthWindow authWindow = new AuthWindow();
        authWindow.Show();
        this.Close();
    }
}

private void Button_Window_Auth_Click(object sender, RoutedEventArgs e)
{
    AuthWindow authWindow = new AuthWindow();
    authWindow.Show();
    this.Close();
}

private bool IsUserExist()
{
    User user = null;
    user = db.Users.Where(b => b.Login ==
textBoxLogin.textBoxInput.Text.Trim()).FirstOrDefault();
    if (user != null)
        return true;
    else
        return false;
}

public static string GetHash(string text)
{
    var md5 = MD5.Create();
    var hash = md5.ComputeHash(Encoding.UTF8.GetBytes(text));
    return Convert.ToBase64String(hash);
}

private void textBoxAge_PreviewTextInput(object sender,
TextCompositionEventArgs e)
{
    e.Handled = new Regex("[^0-9]+").IsMatch(e.Text);
}
}
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				40
Змн.	Арк.	№ докум.	Підпис	Дата		

Файл AuthWindow.xaml.cs

```

using System.Linq;
using System.Text.RegularExpressions;
using System.Windows;
using System.Windows.Media;

namespace usersApp.Windows
{
    public partial class AuthWindow : Window
    {
        private AppContext db;
        public AuthWindow()
        {
            InitializeComponent();
            db = AppContext.GetInstance();
        }
        public static User authUser = null;
        private void Button_Auth_Click(object sender, RoutedEventArgs e)
        {
            string login = textBoxLogin.textBoxInput.Text.Trim();
            string password = textBoxPassword.Password.Trim();
            Regex regex = new Regex("^([a-z0-9_-]+\\.)*[a-z0-9_-]+@[a-z0-9_-]+(\\.([a-z0-9_-]+)*\\.([a-z]{2,6})$)", RegexOptions.Compiled);
            Match match = regex.Match(login);
            if (!match.Success)
            {
                textBoxLogin.textBoxInput.ToolTip = "Некоректно введений логін";
                textBoxLogin.textBoxInput.Background = Brushes.DarkRed;
            }
            else if (password.Length < 5)
            {
                textBoxPassword.ToolTip = "Мінімальна довжина пароля 5 символів";
                textBoxPassword.Background = Brushes.DarkRed;
            }
            else
            {
                password = MainWindow.GetHash(password);
                authUser = db.Users.Where(b => b.Login == login && b.Password == password).FirstOrDefault();
                if (authUser != null)
                {
                    new MessageBox("Авторизація пройшла успішно", MessageType.Success,
                        MessageButtons.Ok).ShowDialog();
                    UserPageWindow userPageWindow = new UserPageWindow();
                    userPageWindow.Show();
                    this.Close();
                }
                else
                {
                    new MessageBox("Користувача не знайдено", MessageType.Error,
                        MessageButtons.Ok).ShowDialog();
                }
            }
            private void Button_Red_Click(object sender, RoutedEventArgs e)
            {
                MainWindow mainWindow = new MainWindow();
                mainWindow.Show();
                this.Close();
            }
        }
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				41
Змн.	Арк.	№ докум.	Підпис	Дата		

Файл UserPageWindow.xaml.cs

```
using System.Collections.Generic;
using System.Windows;
using System.Windows.Controls;
using usersApp.Pages;

namespace usersApp.Windows
{
    public partial class UserPageWindow : Window
    {
        public static List<Page> navigationPages;
        public UserPageWindow()
        {
            InitializeComponent();
            navigationPages = new List<Page>() { new HomePage(), new UserPage(), new
FavouritePage(), new MoviesPage(), new BuyingPage(), new SupportPage() };
        }
        private void Ham1_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[0];
        }
        private void Ham2_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[1];
        }
        public void Ham3_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[2];
        }
        private void Ham4_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[3];
        }
        private void Ham5_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[4];
        }
        private void Ham6_Selected(object sender, RoutedEventArgs e)
        {
            ContentFrame.Content = navigationPages[5];
        }
    }
}
```

Файл MesBox.xaml.cs

```
using System.Windows;
using System.Windows.Input;
using System.Windows.Media;

namespace usersApp.Windows
{
    public partial class MesBox : Window
    {
        public MesBox(string Message, MessageType Type, MessageButtons Buttons)
        {
            InitializeComponent();
            txtMessage.Content = Message;
            switch (Type)
            {
                case MessageType.Success:
                {
                    txtTitle.Text = "Успіх";
                    cardHeader.Background = Brushes.Green;
                }
            }
        }
    }
}
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				42
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        btnOk.Background = Brushes.Green;
    }
    break;
case MessageType.Warning:
    txtTitle.Text = "Увага";
    cardHeader.Background = Brushes.DarkOrange;
    btnOk.Background = Brushes.DarkOrange;
    break;
case MessageType.Error:
    {
        txtTitle.Text = "Помилка";
        cardHeader.Background = Brushes.Red;
        btnOk.Background = Brushes.Red;
    }
    break;
}
switch (Buttons)
{
    case MessageButtons.Ok:
        btnOk.Visibility = Visibility.Visible;
        break;
}
}
private void btnOk_Click(object sender, RoutedEventArgs e)
{
    this.DialogResult = true;
    this.Close();
}
private void btnClose_Click(object sender, RoutedEventArgs e)
{
    this.DialogResult = false;
    this.Close();
}
private void cardHeader_MouseDown(object sender, MouseButtonEventArgs e)
{
    if(e.LeftButton == MouseButtonState.Pressed)
    {
        DragMove();
    }
}
}
public enum MessageType
{
    Success,
    Warning,
    Error,
}
public enum MessageButtons
{
    Ok,
}
}

```

Файл CinemaHall.xaml.cs

```

using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using usersApp.Classes.CinemaClasses;
using usersApp.Classes.DataBaseClasses;
using System.Threading;
using usersApp.Pages;

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				43
Змн.	Арк.	№ докум.	Підпис	Дата		

```

namespace usersApp.Windows
{
    public partial class CinemaHall : Window
    {
        private AppContext db;
        private Movie movie;
        public CinemaHall()
        {
            InitializeComponent();
        }
        private async void CinemaPlaceFavorite_Click(object sender, RoutedEventArgs e)
        {
            if (comboBoxRow.SelectedIndex == -1)
            {
                comboBoxRow.ToolTip = "Ви не обрали дату";
                comboBoxRow.Background = Brushes.DarkRed;
                return;
            }
            else if (comboBoxColumn.SelectedIndex == -1)
            {
                comboBoxColumn.ToolTip = "Ви не обрали час";
                comboBoxColumn.Background = Brushes.DarkRed;
                return;
            }
            else
            {
                if (AuthWindow.authUser != null)
                {
                    db = AppContext.GetInstance();
                    List<Movie> movies = db.Movies.Where(b => b.User_Id ==
AuthWindow.authUser.id).ToList();
                    if(movies.Count >= 1)
                    {
                        new MesBox("В обране можна додати лише один фільм",
MessageType.Warning, MessageBoxButtons.Ok).ShowDialog();
                        this.Close();
                        return;
                    }
                    else
                    {
                        if(MovieDetails.cinemaMovie != null)
                        {
                            MovieDetails.cinemaMovie.CinemaPlace = new
CinemaPlace(comboBoxRow.SelectedIndex + 1, comboBoxColumn.SelectedIndex + 1);
                            movie = new Movie(MovieDetails.cinemaMovie.Name,
MovieDetails.cinemaMovie.TimeOfSession, MovieDetails.cinemaMovie.DateOfSession.Day,
MovieDetails.cinemaMovie.DateOfSession.Month,
MovieDetails.cinemaMovie.DateOfSession.Year, MovieDetails.cinemaMovie.CinemaPlace.Row,
MovieDetails.cinemaMovie.CinemaPlace.Column, MovieDetails.cinemaMovie.Price,
AuthWindow.authUser.id, MovieDetails.cinemaMovie.Plan);
                            db.Movies.Add(movie);
                            db.SaveChanges();
                            UserPageWindow.navigationPages[2] = new FavouritePage();
                            SuccessPopUpWindow successPopUpWindow = new
SuccessPopUpWindow();
                            successPopUpWindow.Show();
                            await Task.Factory.StartNew(() => Pause(3));
                            successPopUpWindow.Close();
                            this.Close();
                        }
                    }
                }
            }
        }
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				44
Змн.	Арк.	№ докум.	Підпис	Дата		

```

private void Pause(int T)
{
    Thread.Sleep(T * 1000);
}
private void comboBoxRow_SelectionChanged(object sender,
SelectionChangedEventArgs e)
{
    comboBoxRow.ToolTip = null;
    comboBoxRow.Background = Brushes.Transparent;
    comboBoxColumn.IsEnabled = true;
    List<string> list = new List<string>();
    if (comboBoxRow.SelectedIndex == 0)
    {
        for (int i = 0; i < 10; i++)
            list.Add($"{i + 1} місце");
        comboBoxColumn.ItemsSource = list;
    }
    else if (comboBoxRow.SelectedIndex == 1 || comboBoxRow.SelectedIndex == 2)
    {
        for(int i = 0; i < 12; i++)
            list.Add($"{i + 1} місце");
        comboBoxColumn.ItemsSource = list;
    }
    else if (comboBoxRow.SelectedIndex == 3 || comboBoxRow.SelectedIndex == 4)
    {
        for (int i = 0; i < 14; i++)
            list.Add($"{i + 1} місце");
        comboBoxColumn.ItemsSource = list;
    }
    else if (comboBoxRow.SelectedIndex == 5)
    {
        for (int i = 0; i < 16; i++)
            list.Add($"{i + 1} місце");
        comboBoxColumn.ItemsSource = list;
    }
}
private void comboBoxColumn_SelectionChanged(object sender,
SelectionChangedEventArgs e)
{
    comboBoxColumn.ToolTip = null;
    comboBoxColumn.Background = Brushes.Transparent;
}
}
}

```

Файл MovieDetails.xaml.cs

```

using System;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using System.Diagnostics;
using usersApp.Classes.CinemaClasses;

namespace usersApp.Windows
{
    public partial class MovieDetails : Window
    {
        public static CinemaMovie cinemaMovie = null;
        public MovieDetails()
        {
            InitializeComponent();
            datePickerDate.DisplayDateStart = DateTime.Now;
            datePickerDate.DisplayDateEnd = DateTime.Now.AddDays(6);
        }
        private string linkTrailer;
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				45
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public string LinkTrailer
{
    get { return linkTrailer; }
    set { linkTrailer = value; }
}
private void watchTrailer_Click(object sender, RoutedEventArgs e)
{
    Process.Start(linkTrailer);
}
private void buttonChoosePlace_Click(object sender, RoutedEventArgs e)
{
    if (datePickerDate.Text == "")
    {
        datePickerDate.ToolTip = "Ви не обрали дату";
        datePickerDate.Background = Brushes.DarkRed;
        return;
    }
    else if (comboBoxTime.SelectedIndex == -1)
    {
        comboBoxTime.ToolTip = "Ви не обрали час";
        comboBoxTime.Background = Brushes.DarkRed;
        return;
    }
    else
    {
        if (AuthWindow.authUser != null)
        {
            if (AuthWindow.authUser.Age >= int.Parse(textBlockMinAge.Text))
            {
                string plan = null;
                if (radioButtonPension.IsChecked == true)
                    plan = "Пенсіонер";
                else if (radioButtonSchoolChild.IsChecked == true)
                    plan = "Школяр";
                else if (radioButtonStudent.IsChecked == true)
                    plan = "Студент";
                cinemaMovie = new CinemaMovie(textBlockNameOfMovie.Text,
                MakePriceOfSession(), datePickerDate.SelectedDate.Value, comboBoxTime.Text, plan);
                this.Close();
                CinemaHall cinemaHall = new CinemaHall();
                cinemaHall.ShowDialog();
            }
            else
            {
                new MesBox($"{"На жаль, Ви не досягли", 34}\n{"достатнього віку для цього фільму", 36}", MessageType.Warning, MessageButtons.Ok).ShowDialog();
                return;
            }
        }
    }
}

private void comboBoxTime_SelectionChanged(object sender,
SelectionChangedEventArgs e)
{
    comboBoxTime.ToolTip = null;
    comboBoxTime.Background = Brushes.Transparent;
}

private void datePickerDate_CalendarClosed(object sender, RoutedEventArgs e)
{
    if (datePickerDate.Text != "")
    {
        datePickerDate.ToolTip = null;
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				46
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        datePickerDate.Background = Brushes.Transparent;
    }
}

private void buttonReloadPrice_Click(object sender, RoutedEventArgs e)
{
    if (datePickerDate.Text == "" || comboBoxTime.SelectedIndex == -1)
    {
        new MesBox("Ви не обрали дату, чи час", MessageType.Error,
MessageButtons.Ok).ShowDialog();
        return;
    }
    else
    {
        textBlockPriceOfSession.Visibility = Visibility.Visible;
        textBlockUAH.Visibility = Visibility.Visible;
        MakePriceOfSession();
    }
}

private double MakePriceOfSession()
{
    double priceOfSession;
    bool isAnyChecked = false;
    if (radioButtonPension.IsChecked == true ||
radioButtonSchoolChild.IsChecked == true || radioButtonStudent.IsChecked == true)
        isAnyChecked = true;
    DateTime dateTime = datePickerDate.SelectedDate.Value;
    if (dateTime.DayOfWeek == DayOfWeek.Sunday || dateTime.DayOfWeek ==
DayOfWeek.Saturday)
    {
        if (comboBoxTime.SelectedIndex >= 3)
        {
            if (isAnyChecked)
                priceOfSession = 100;
            else
                priceOfSession = 110;
        }
        else
        {
            if (isAnyChecked)
                priceOfSession = 80;
            else
                priceOfSession = 90;
        }
    }
    else
    {
        if (comboBoxTime.SelectedIndex >= 3)
        {
            if (isAnyChecked)
                priceOfSession = 80;
            else
                priceOfSession = 90;
        }
        else
        {
            if (isAnyChecked)
                priceOfSession = 60;
            else
                priceOfSession = 70;
        }
    }
    textBlockPriceOfSession.Text = priceOfSession.ToString();
    return priceOfSession;
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				47
Змн.	Арк.	№ докум.	Підпис	Дата		

```
}
}
```

Файл OrderWindow.xaml.cs

```
using System;
using System.IO;
using System.Reflection;
using System.Linq;
using System.Windows;
using usersApp.Classes.CinemaClasses;
using Word = Microsoft.Office.Interop.Word;
using usersApp.Classes.DataBaseClasses;
using SautinSoft.Document;
using usersApp.Pages;

namespace usersApp.Windows
{
    public partial class OrderWindow : Window
    {
        private CinemaOrder cinemaOrder;
        public static bool IsOrderMade = false;
        public OrderWindow()
        {
            InitializeComponent();
        }
        private void buttonCompleteOrder_Click(object sender, RoutedEventArgs e)
        {
            IsOrderMade = true;
            string paymentMethod = "";
            if (radioButtonCardPay.IsChecked == true)
                paymentMethod = radioButtonCardPay.Content.ToString();
            else if (radioButtonCashPay.IsChecked == true)
                paymentMethod = radioButtonCashPay.Content.ToString();
            cinemaOrder = new CinemaOrder(DateTime.Now, paymentMethod);
            string savePath =
Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments) +
$"\\CinemaOrder{AuthWindow.authUser.Login}Ticket.docx";

CreateWordDocument(BingPathToAppDir("OrderTicketsFiles\\TemporaryFiles\\TemporaryTicke
t.docx"), savePath);
            this.Close();
        }
        private void FindAndReplace(Word.Application wordApp, object toFindText,
object replaceWithText)
        {
            object matchCase = true;
            object matchWholeWord = true;
            object matchWildCards = false;
            object matchSoundLike = false;
            object nmatchAllforms = false;
            object forward = true;
            object format = false;
            object matchKashida = false;
            object matchDiactitics = false;
            object matchAlefHamza = false;
            object matchControl = false;
            object read_only = false;
            object visible = true;
            object replace = 2;
            object wrap = 1;
            wordApp.Selection.Find.Execute(ref toFindText, ref matchCase, ref
matchWholeWord, ref matchWildCards, ref matchSoundLike, ref nmatchAllforms, ref
forward, ref wrap, ref format, ref replaceWithText, ref replace, ref matchKashida, ref
matchDiactitics, ref matchAlefHamza, ref matchControl);
        }
    }
}
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				48
Змн.	Арк.	№ докум.	Підпис	Дата		


```

private void CreateWordDocument(object filename, object SaveAs)
{
    Word.Application wordApp = new Word.Application();
    object missing = Missing.Value;
    Word.Document myWordDoc = null;
    if (File.Exists((string)filename))
    {
        object readOnly = false;
        object isVisible = false;
        wordApp.Visible = false;
        myWordDoc = wordApp.Documents.Open(ref filename, ref missing, ref
readOnly, ref missing, ref missing, ref missing, ref missing, ref
missing, ref missing, ref missing, ref missing, ref missing, ref missing,
ref missing);
        myWordDoc.Activate();
        this.FindAndReplace(wordApp, "<movie>",
FavouritePage.movies.FirstOrDefault().Name);
        this.FindAndReplace(wordApp, "<dateOfSession>",
FavouritePage.movies.FirstOrDefault().DayOfSession.ToString() + "." +
FavouritePage.movies.FirstOrDefault().MonthOfSession.ToString() + "." +
FavouritePage.movies.FirstOrDefault().YearOfSession.ToString());
        this.FindAndReplace(wordApp, "<timeOfSession>",
FavouritePage.movies.FirstOrDefault().TimeOfSession);
        this.FindAndReplace(wordApp, "<placeRow>",
FavouritePage.movies.FirstOrDefault().RowPlace.ToString());
        this.FindAndReplace(wordApp, "<placeColumn>",
FavouritePage.movies.FirstOrDefault().ColumnPlace.ToString());
        this.FindAndReplace(wordApp, "<moviePrice>",
FavouritePage.movies.FirstOrDefault().Price.ToString() + "грн");
        if (FavouritePage.products.Count != 0)
        {
            string tmp = "";
            foreach (Product product in FavouritePage.products)
                tmp += product.Name + " " + product.Price + "грн" + " " +
product.Amount + "шт" + Environment.NewLine;
            this.FindAndReplace(wordApp, "<productsTitle>", "ТОВАРИ");
            this.FindAndReplace(wordApp, "<products>", tmp);
        }
        else
        {
            this.FindAndReplace(wordApp, "<products>", "");
            this.FindAndReplace(wordApp, "<productsTitle>", "");
        }
        this.FindAndReplace(wordApp, "<totalPrice>",
FavouritePage.TotalPrice.ToString() + "грн");
        this.FindAndReplace(wordApp, "<paymentMethod>",
cinemaOrder.PaymentMethod);
        this.FindAndReplace(wordApp, "<dateAndTimeOfOrder>",
cinemaOrder.DateTimeOfOrder.ToString());
        if (FavouritePage.movies.FirstOrDefault().Plan != null)
        {
            this.FindAndReplace(wordApp, "<planTitle>", "Тариф:");
            this.FindAndReplace(wordApp, "<plan>",
FavouritePage.movies.FirstOrDefault().Plan);
        }
        else
        {
            this.FindAndReplace(wordApp, "<planTitle>", "");
            this.FindAndReplace(wordApp, "<plan>", "");
        }
    }
    else
    {
        new MesBox("Файл не знайдено", MessageType.Error,
MessageButtons.Ok).ShowDialog();
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				49
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        return;
    }
    myWordDoc.SaveAs2(ref SaveAs, ref missing, ref missing, ref missing, ref
missing, ref missing, ref missing, ref missing, ref missing, ref missing,
ref missing, ref missing, ref missing, ref missing, ref missing);
    myWordDoc.Close();
    wordApp.Quit();
    new MesBox("Ваш квіток створено", MessageType.Success,
MessageButtons.Ok).ShowDialog();
}
private void checkBoxAcceptRules_Click(object sender, RoutedEventArgs e)
{
    if (checkBoxAcceptRules.IsChecked == true)
        buttonCompleteOrder.IsEnabled = true;
    else
        buttonCompleteOrder.IsEnabled = false;
}
private void radioButtonCardPay_Checked(object sender, RoutedEventArgs e)
{
    if(!checkBoxAcceptRules.IsEnabled)
        checkBoxAcceptRules.IsEnabled = true;
}
private void radioButtonCashPay_Checked(object sender, RoutedEventArgs e)
{
    if (!checkBoxAcceptRules.IsEnabled)
        checkBoxAcceptRules.IsEnabled = true;
}
public static string BingPathToAppDir(string localPath)
{
    string currentDir = Environment.CurrentDirectory;
    DirectoryInfo directory = new
DirectoryInfo(Path.GetFullPath(Path.Combine(currentDir, @"..\..\\" + localPath)));
    return directory.ToString();
}
}
}

```

Файл FavouritePage.xaml.cs

```

using System.Collections.Generic;
using System.Linq;
using System.Windows;
using System.Windows.Controls;
using usersApp.Classes.DataBaseClasses;
using usersApp.Windows;

namespace usersApp.Pages
{
    public partial class FavouritePage : Page
    {
        public static AppContext db;
        public static List<Product> products;
        public static List<Movie> movies;
        public static double TotalPrice = 0;
        private double totalMoviesPrice = 0, totalProductsPrice = 0;
        public FavouritePage()
        {
            InitializeComponent();
            if (AuthWindow.authUser != null)
            {
                db = AppContext.GetInstance();
                products = db.Products.Where(b => b.User_Id ==
AuthWindow.authUser.id).ToList();
                movies = db.Movies.Where(b => b.User_Id ==
AuthWindow.authUser.id).ToList();
                if (products.Count != 0)

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				50
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            textBlockTitleProducts.Visibility = Visibility.Visible;
            buttonRemoveFavorite.IsEnabled = true;
            textBlockTitle.Text = "База об'єктів:";
            listViewListOfProducts.ItemsSource = products;
            foreach (Product product in products)
                totalProductsPrice += product.Price * product.Amount;
        }
        if (movies.Count != 0)
        {
            textBlockTitleMovies.Visibility = Visibility.Visible;
            buttonRemoveFavorite.IsEnabled = true;
            textBlockTitle.Text = "База об'єктів:";
            listViewListOfMovies.ItemsSource = movies;
            foreach (Movie movie in movies)
                totalMoviesPrice += movie.Price;
        }
        TotalPrice = totalMoviesPrice + totalProductsPrice;
        if (TotalPrice != 0)
        {
            textBlockTotalPrice.Text = TotalPrice.ToString();
            textBlockTotalPrice.Visibility = Visibility.Visible;
            textBlockTotalPriceUAH.Visibility = Visibility.Visible;
            textBlockTotalPriceTitle.Visibility = Visibility.Visible;
            buttonMakeOrder.Visibility = Visibility.Visible;
        }
    }
}

private void buttonRemoveFavorite_Click(object sender, RoutedEventArgs e)
{
    if (AuthWindow.authUser != null)
    {
        if (movies.Count != 0)
        {
            foreach (Movie movie in movies)
                db.Movies.Remove(movie);
            db.SaveChanges();
            listViewListOfMovies.ItemsSource = null;
            textBlockTitleMovies.Visibility = Visibility.Collapsed;
        }
        if (products.Count != 0)
        {
            foreach (Product product in products)
                db.Products.Remove(product);
            db.SaveChanges();
            listViewListOfProducts.ItemsSource = null;
            textBlockTitleProducts.Visibility = Visibility.Collapsed;
        }
        textBlockTotalPriceUAH.Visibility = Visibility.Hidden;
        textBlockTotalPriceTitle.Visibility = Visibility.Hidden;
        textBlockTotalPrice.Visibility = Visibility.Hidden;
        buttonMakeOrder.Visibility = Visibility.Hidden;
        textBlockTitle.Text = "В вашому обраному поки пусто";
        buttonRemoveFavorite.IsEnabled = false;
    }
}

private void buttonMakeOrder_Click(object sender, RoutedEventArgs e)
{
    if (AuthWindow.authUser != null)
    {
        if (movies.Count != 0)
        {
            OrderWindow orderWindow = new OrderWindow();
            orderWindow.ShowDialog();
            if (orderWindow.IsOrderMade)

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				51
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            buttonRemoveFavorite_Click(sender, e);
            OrderWindow.IsOrderMade = false;
        }
    }
    else
    {
        new MesBox("Ви не обрали жодного фільму", MessageType.Warning,
MessageButtons.Ok).ShowDialog();
        return;
    }
}
}
}
}
}
}
}
}
}
}

```

Файл HomePage.xaml.cs

```

using System.Windows.Controls;
using usersApp.Windows;

namespace usersApp.Pages
{
    public partial class HomePage : Page
    {
        public HomePage()
        {
            InitializeComponent();
            if (AuthWindow.authUser != null)
                textBlockNameOfUser.Text = AuthWindow.authUser.Name;
        }
    }
}

```

Файл UserPage.xaml.cs

```

using System.Windows.Controls;
using usersApp.Windows;

namespace usersApp.Pages
{
    public partial class UserPage : Page
    {
        public UserPage()
        {
            InitializeComponent();
            if (AuthWindow.authUser != null)
            {
                textBlockUserName.Text = AuthWindow.authUser.Name;
                textBlockUserLogin.Text = AuthWindow.authUser.Login;
                textBlockUserAge.Text = AuthWindow.authUser.Age.ToString();
            }
        }
    }
}

```

Файл AuthRegTextBoxControl.xaml.cs

```

using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using MaterialDesignThemes.Wpf;

namespace usersApp.CustomControls
{
    public partial class AuthRegTextBoxControl : UserControl
    {
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				52
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    public AuthRegTextBoxControl()
    {
        InitializeComponent();
    }
    public string HintText
    {
        get { return (string)GetValue(HintTextProperty); }
        set { SetValue(HintTextProperty, value); }
    }
    public static readonly DependencyProperty HintTextProperty =
        DependencyProperty.Register("HintTextProperty", typeof(string),
typeof(AuthRegTextBoxControl), new PropertyMetadata(null, OnHintTextPropertyChanged));

    private static void OnHintTextPropertyChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
    {
        AuthRegTextBoxControl source = d as AuthRegTextBoxControl;
        HintAssist.SetHint(source.textBoxInput, (string)e.NewValue);
    }
    private void textBoxInput_TextChanged(object sender, TextChangedEventArgs e)
    {
        textBoxInput.Background = Brushes.Transparent;
        textBoxInput.ToolTip = null;
    }
}
}

```

Файл ContactControl.xaml.cs

```

using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using System.Diagnostics;

namespace usersApp.CustomControls
{
    public partial class ContactControl : UserControl
    {
        public ContactControl()
        {
            InitializeComponent();
        }
        public ImageSource ContactImageSource
        {
            get { return (ImageSource)GetValue(ContactImageSourceProperty); }
            set { SetValue(ContactImageSourceProperty, value); }
        }
        public static readonly DependencyProperty ContactImageSourceProperty =
            DependencyProperty.Register("ContactImageSourceProperty",
typeof(ImageSource), typeof(ContactControl), new PropertyMetadata(null,
OnContactImageSourcePropertyChanged));
        private static void OnContactImageSourcePropertyChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
        {
            ContactControl source = d as ContactControl;
            source.ImageContact.Source = (ImageSource)e.NewValue;
        }
        public string ContactName
        {
            get { return (string)GetValue(ContactNameProperty); }
            set { SetValue(ContactNameProperty, value); }
        }
        public static readonly DependencyProperty ContactNameProperty =
            DependencyProperty.Register("ContactNameProperty", typeof(string),
typeof(ContactControl), new PropertyMetadata(null, OnContactNamePropertyChanged));
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				53
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        private static void OnContactNamePropertyChanged(DependencyObject d,
DependencyPropertyPropertyChangedEventArgs e)
        {
            ContactControl source = d as ContactControl;
            source.TextBlockContactName.Text = (string)e.NewValue;
        }
        private string contactLink;
        public string ContactLink
        {
            get { return contactLink; }
            set { contactLink = value; }
        }
        private void HyperLinkContact_Click(object sender, RoutedEventArgs e)
        {
            Process.Start(contactLink);
        }
    }
}

```

Файл MovieControl.xaml.cs

```

using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using usersApp.Windows;

namespace usersApp.CustomControls
{
    public partial class MovieControl : UserControl
    {
        public MovieControl()
        {
            InitializeComponent();
        }
        public ImageSource MovieImageSource
        {
            get { return (ImageSource)GetValue(MovieImageSourceProperty); }
            set { SetValue(MovieImageSourceProperty, value); }
        }
        public static readonly DependencyProperty MovieImageSourceProperty =
            DependencyProperty.Register("MovieImageSourceProperty",
typeof(ImageSource), typeof(MovieControl), new PropertyMetadata(null,
OnMovieImageSourcePropertyChanged));

        private static void OnMovieImageSourcePropertyChanged(DependencyObject d,
DependencyPropertyPropertyChangedEventArgs e)
        {
            MovieControl source = d as MovieControl;
            source.ImageMovie.Source = (ImageSource)e.NewValue;
        }
        public string MovieName
        {
            get { return (string)GetValue(MovieNameProperty); }
            set { SetValue(MovieNameProperty, value); }
        }
        public static readonly DependencyProperty MovieNameProperty =
            DependencyProperty.Register("MovieNameProperty", typeof(string),
typeof(MovieControl), new PropertyMetadata(null, OnMovieNamePropertyChanged));
        private static void OnMovieNamePropertyChanged(DependencyObject d,
DependencyPropertyPropertyChangedEventArgs e)
        {
            MovieControl source = d as MovieControl;
            source.TextBlockMovieName.Text = (string)e.NewValue;
        }
        private string linkTrailer;
        public string LinkTrailer

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				54
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    get { return linkTrailer; }
    set { linkTrailer = value; }
}
private int minAge;
public int MinAge
{
    get { return minAge; }
    set { minAge = value; }
}
private int yearOfIssue;
public int YearOfIssue
{
    get { return yearOfIssue; }
    set { yearOfIssue = value; }
}
private string movieDuration;
public string MovieDuration
{
    get { return movieDuration; }
    set { movieDuration = value; }
}
private string countryOfProduction;
public string CountryOfProduction
{
    get { return countryOfProduction; }
    set { countryOfProduction = value; }
}
private string movieGenre;
public string MovieGenre
{
    get { return movieGenre; }
    set { movieGenre = value; }
}
private string movieStudio;
public string MovieStudio
{
    get { return movieStudio; }
    set { movieStudio = value; }
}
private MovieDetails movieDetails;
private void ButtonMovieDetails_Click(object sender, RoutedEventArgs e)
{
    movieDetails = new MovieDetails();
    movieDetails.ImageMovie.Source = MovieImageSource;
    movieDetails.textBlockNameOfMovie.Text = MovieName;
    movieDetails.textBlockMinAge.Text = minAge.ToString();
    movieDetails.textBlockYearOfIssue.Text = yearOfIssue.ToString();
    movieDetails.textBlockMovieDuration.Text = movieDuration;
    movieDetails.textBlockCountryOfProduction.Text = countryOfProduction;
    movieDetails.textBlockMovieGenre.Text = movieGenre;
    movieDetails.textBlockMovieStudio.Text = movieStudio;
    movieDetails.LinkTrailer = linkTrailer;
    movieDetails.ShowDialog();
}
}
}

```

Файл ProductControl.xaml.cs

```

using System;
using System.Text.RegularExpressions;
using System.Threading;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				55
Змн.	Арк.	№ докум.	Підпис	Дата		


```

using System.Windows.Input;
using System.Windows.Media;
using usersApp.Classes.CinemaClasses;
using usersApp.Classes.DataBaseClasses;
using usersApp.Pages;
using usersApp.Windows;
using XamlAnimatedGif;

namespace usersApp.CustomControls
{
    public partial class ProductControl : UserControl
    {
        public ProductControl()
        {
            InitializeComponent();
        }
        public ImageSource ProductImageSource
        {
            get { return (ImageSource)GetValue(ProductImageSourceProperty); }
            set { SetValue(ProductImageSourceProperty, value); }
        }
        public static readonly DependencyProperty ProductImageSourceProperty =
            DependencyProperty.Register("ProductImageSourceProperty",
typeof(ImageSource), typeof(ProductControl), new PropertyMetadata(null,
OnProductImageSourcePropertyChanged));
        private static void OnProductImageSourcePropertyChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
        {
            ProductControl source = d as ProductControl;
            Uri uri = new Uri(e.NewValue.ToString());
            AnimationBehavior.SetSourceUri(source.ImageProduct, uri);
        }
        public string ProductName
        {
            get { return (string)GetValue(ProductNameProperty); }
            set { SetValue(ProductNameProperty, value); }
        }
        public static readonly DependencyProperty ProductNameProperty =
            DependencyProperty.Register("ProductNameProperty", typeof(string),
typeof(ProductControl), new PropertyMetadata(null, OnProductNamePropertyChanged));
        private static void OnProductNamePropertyChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
        {
            ProductControl source = d as ProductControl;
            source.TextBlockProductName.Text = (string)e.NewValue;
        }
        public string ProductPrice
        {
            get { return (string)GetValue(ProductPriceProperty); }
            set { SetValue(ProductPriceProperty, value); }
        }
        public static readonly DependencyProperty ProductPriceProperty =
            DependencyProperty.Register("ProductPriceProperty", typeof(string),
typeof(ProductControl), new PropertyMetadata(null, OnProductPricePropertyChanged));
        private static void OnProductPricePropertyChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
        {
            ProductControl source = d as ProductControl;
            source.TextBlockProductPrice.Text = (string)e.NewValue + " грн";
        }
        private int count { get; set; }
        private void ButtonProductInc_Click(object sender, RoutedEventArgs e)
        {
            try

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				56
Змн.	Арк.	№ докум.	Підпис	Дата		


```

    {
        ButtonProductDec.IsEnabled = true;
        count = int.Parse(TextBlockProductCount.Text);
        count++;
        TextBlockProductCount.Text = count.ToString();
    }
    catch (Exception)
    {
        TextBlockProductCount.Text = "0";
        ButtonProductDec.IsEnabled = false;
    }
}
private void ButtonProductDec_Click(object sender, RoutedEventArgs e)
{
    try
    {
        count = int.Parse(TextBlockProductCount.Text);
        count--;
        TextBlockProductCount.Text = count.ToString();
        if (TextBlockProductCount.Text == "0")
            ButtonProductDec.IsEnabled = false;
    }
    catch (Exception)
    {
        TextBlockProductCount.Text = "0";
        ButtonProductDec.IsEnabled = false;
    }
}
private AppContext db;
private Product product;
private CinemaShopProduct cinemaShopProduct;
private async void ButtonToFavorites_Click(object sender, RoutedEventArgs e)
{
    if (TextBlockProductCount.Text == "" ||
    int.Parse(TextBlockProductCount.Text) <= 0)
    {
        new MessageBox("Помилка в кількості обраного вами товару",
        MessageType.Warning, MessageButtons.Ok).ShowDialog();
        return;
    }
    else
    {
        if (AuthWindow.authUser != null)
        {
            cinemaShopProduct = new CinemaShopProduct(ProductName,
            int.Parse(ProductPrice), int.Parse(TextBlockProductCount.Text));
            db = AppContext.GetInstance();
            product = new Product(cinemaShopProduct.Name,
            cinemaShopProduct.Price, AuthWindow.authUser.id, cinemaShopProduct.Amount);
            db.Products.Add(product);
            db.SaveChanges();
            UserPageWindow.navigationPages[2] = new FavouritePage();
            Uri uriSuccess = new
            Uri("pack://application:,,,/Images/success.gif");
            AnimationBehavior.SetSourceUri(ImageProduct, uriSuccess);
            await Task.Factory.StartNew(() => Pause(3));
            Uri uriCurrent = new Uri(ProductImageSource.ToString());
            AnimationBehavior.SetSourceUri(ImageProduct, uriCurrent);
            TextBlockProductCount.Text = "0";
            ButtonProductDec.IsEnabled = false;
        }
    }
}
private void Pause(int T)
{
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				57
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        Thread.Sleep(T * 1000);
    }
    private void TextBlockProductCount_PreviewTextInput(object sender,
    TextCompositionEventArgs e)
    {
        if (e.Text == "0")
            ButtonProductDec.IsEnabled = false;
        else
            ButtonProductDec.IsEnabled = true;
        e.Handled = new Regex("[^0-9]+").IsMatch(e.Text);
    }
}

```

Файл AppContext.cs

```

using System.Data.Entity;
using usersApp.Classes.DataBaseClasses;

namespace usersApp
{
    public class AppContext : DbContext
    {
        public DbSet<User> Users { get; set; }
        public DbSet<Product> Products { get; set; }
        public DbSet<Movie> Movies { get; set; }
        private AppContext() : base("DefaultConnection") { }
        private static AppContext instance;
        private static readonly object _lock = new object();
        public static AppContext GetInstance()
        {
            if (instance == null)
                lock (_lock)
                    if (instance == null)
                        instance = new AppContext();
            return instance;
        }
    }
}

```

Файл Movie.cs

```

using System;

namespace usersApp.Classes.DataBaseClasses
{
    public class Movie
    {
        public int id { get; set; }
        private string name, timeOfSession, plan;
        private int dayOfSession, monthOfSession, yearOfSession, user_id, rowPlace,
        columnPlace;
        private double price;
        public int DayOfSession
        {
            get { return dayOfSession; }
            set { dayOfSession = value; }
        }
        public int MonthOfSession
        {
            get { return monthOfSession; }
            set { monthOfSession = value; }
        }
        public int YearOfSession
        {
            get { return yearOfSession; }

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				58
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        set { yearOfSession = value; }
    }
    public int RowPlace
    {
        get { return rowPlace; }
        set { rowPlace = value; }
    }
    public int ColumnPlace
    {
        get { return columnPlace; }
        set { columnPlace = value; }
    }
    public int User_Id
    {
        get { return user_id; }
        set { user_id = value; }
    }
    public string Name
    {
        get { return name; }
        set { name = value; }
    }
    public string Plan
    {
        get { return plan; }
        set { plan = value; }
    }
    public string TimeOfSession
    {
        get { return timeOfSession; }
        set { timeOfSession = value; }
    }
    public double Price
    {
        get { return price; }
        set { price = value; }
    }
    public Movie() { }
    public Movie(string name, string timeOfSession, int dayOfSession, int
monthOfSession, int yearOfSession, int rowPlace, int columnPlace, double price, int
user_id, string plan)
    {
        this.name = name;
        this.timeOfSession = timeOfSession;
        this.dayOfSession = dayOfSession;
        this.monthOfSession = monthOfSession;
        this.yearOfSession = yearOfSession;
        this.rowPlace = rowPlace;
        this.columnPlace = columnPlace;
        this.price = price;
        this.user_id = user_id;
        this.plan = plan;
    }
}
}

```

Файл Product.cs

```

using System;

namespace usersApp.Classes.DataBaseClasses
{
    public class Product
    {
        public int id { get; set; }
        private string name;
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				59
Змн.	Арк.	№ докум.	Підпис	Дата		

```

private int amount, user_id;
private double price;
public int Amount
{
    get { return amount; }
    set { amount = value; }
}
public int User_Id
{
    get { return user_id; }
    set { user_id = value; }
}
public string Name
{
    get { return name; }
    set { name = value; }
}
public double Price
{
    get { return price; }
    set { price = value; }
}
public Product() { }
public Product(string name, double price, int user_id, int amount)
{
    this.name = name;
    this.price = price;
    this.user_id = user_id;
    this.amount = amount;
}
}
}

```

Файл User.cs

```

using System;

namespace usersApp
{
    public class User
    {
        public int id { get; set; }

        private string login, password, name;
        private int age;
        public int Age
        {
            get { return age; }
            set { age = value; }
        }
        public string Name
        {
            get { return name; }
            set { name = value; }
        }
        public string Login
        {
            get { return login; }
            set { login = value; }
        }

        public string Password
        {
            get { return password; }
            set { password = value; }
        }
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				60
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    public User() { }

    public User(string login, string password, string name, int age)
    {
        this.login = login;
        this.password = password;
        this.name = name;
        this.age = age;
    }
}

```

Файл CinemaProduct.cs

```

using System;

namespace usersApp.Classes.CinemaClasses
{
    public abstract class CinemaProduct
    {
        public string Name { get; set; }
        public double Price { get; set; }
        protected CinemaProduct(string name, double price)
        {
            Name = name;
            Price = price;
        }
        protected CinemaProduct() { }
    }
}

```

Файл CinemaMovie.cs

```

using System;

namespace usersApp.Classes.CinemaClasses
{
    public class CinemaMovie : CinemaProduct
    {
        public DateTime DateOfSession { get; set; }
        public string TimeOfSession { get; set; }
        public string Plan { get; set; }
        public CinemaPlace CinemaPlace { get; set; }
        public CinemaMovie(string name, double price, DateTime dateOfSession, string
timeOfSession, CinemaPlace cinemaPlace, string plan) : base(name, price)
        {
            DateOfSession = dateOfSession;
            TimeOfSession = timeOfSession;
            CinemaPlace = new CinemaPlace(cinemaPlace);
            Plan = plan;
        }
        public CinemaMovie(string name, double price, DateTime dateOfSession, string
timeOfSession, string plan) : base(name, price)
        {
            DateOfSession = dateOfSession;
            TimeOfSession = timeOfSession;
            Plan = plan;
        }
        public CinemaMovie() : base()
        {
            CinemaPlace = new CinemaPlace();
        }
    }
}

```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				61
Змн.	Арк.	№ докум.	Підпис	Дата		

Файл CinemaShopProduct.cs

```
using System;

namespace usersApp.Classes.CinemaClasses
{
    public class CinemaShopProduct : CinemaProduct
    {
        public int Amount { get; set; }
        public CinemaShopProduct(string name, double price, int amount) : base(name,
price)
        {
            Amount = amount;
        }
        public CinemaShopProduct() : base() { }
    }
}
```

Файл CinemaPlace.cs

```
using System;

namespace usersApp.Classes.CinemaClasses
{
    public class CinemaPlace
    {
        public int Row { get; set; }
        public int Column { get; set; }
        public CinemaPlace (CinemaPlace cinemaPlace)
        {
            Row = cinemaPlace.Row;
            Column = cinemaPlace.Column;
        }
        public CinemaPlace() { }
        public CinemaPlace(int row, int column)
        {
            Row = row;
            Column = column;
        }
    }
}
```

Файл CinemaOrder.cs

```
using System;

namespace usersApp.Classes.CinemaClasses
{
    public class CinemaOrder
    {
        public DateTime DateTimeOfOrder { get; set; }
        public string PaymentMethod { get; set; }
        public CinemaOrder(DateTime dateTimeOfOrder, string paymentMethod)
        {
            DateTimeOfOrder = dateTimeOfOrder;
            PaymentMethod = paymentMethod;
        }
        public CinemaOrder() { }
    }
}
```

		Владислав ГЕЙНА			ДУ «Житомирська політехніка».22.121.3.000 - ПЗ	Арк.
		Олег ВЛАСЕНКО				62
Змн.	Арк.	№ докум.	Підпис	Дата		