

Міністерство освіти і науки України
Український державний університет науки і технологій

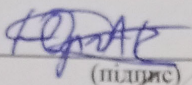
Факультет «Комп'ютерні технології і системи»
Кафедра «Комп'ютерні інформаційні технології»

Пояснювальна записка
до кваліфікаційної роботи бакалавра

на тему: «Розробка автоматизованої системи продажу квитків в кінотеатрі»
за освітньою програмою: «12 Інженерія програмного забезпечення»
зі спеціальності: «121 Інженерія програмного забезпечення»
Виконав: студент групи «ПЗ1911»

Керівник:

Нормоконтролер:


(підпис)

/Артем ЮРТЕЄВ/
(Ім'я ПРІЗВИЩЕ)

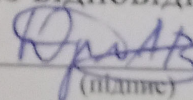

(підпис)

/доц. Олександр ІВАНОВ/
(посада, Ім'я ПРІЗВИЩЕ)


(підпис)

/доц. Світлана ВОЛКОВА/
(посада, Ім'я ПРІЗВИЩЕ)

Засвідчую, що у цій роботі немає запозичень з
праць інших авторів без відповідних посилань.
Студент


(підпис)

Ministry of Education and Science of Ukraine
Ukrainian State University of Science and Technologies

Faculty «Computer technologies and systems»
Department «Computer information technology»

Explanatory Note to Bachelor's Thesis

on the topic: « Development of an automated ticket sales system in the cinema »
according to educational curriculum «Software engineering»
in the Speciality: «121 Software engineering»

Done by the student of the group PZ1911:

/Artem Yrtieiev/

Scientific Supervisor:

/Oleksandr IVANOV/

Normative controller:

/Svitlana VOLKOVA/

Міністерство освіти і науки України
Український державний університет науки і технологій

Факультет: Факультет «Комп'ютерні технології і системи»
Кафедра: «Комп'ютерні інформаційні технології»
Рівень вищої освіти: бакалавр
Освітня програма: «Інженерія програмного забезпечення»
Спеціальність: «121 Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри КІТ
_____/Вадим ГОРЯЧКІН/
(підпис)

Дата _____

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра
студенту Юртеєву Артему Володимировичу

1. Тема роботи: «Розробка автоматизованої системи продажу квитків в кінотеатрі»

Керівник роботи: Іванов Олександр Петрович, доцент
затверджені наказом № 77 ст від 08.12.2021

2. Строк подання студентом роботи: _____.____.202_ р.

3. Вихідні дані до роботи: _____

4. Зміст пояснювальної записки (перелік питань, які потрібно опрацювати):

Вступ

Збір та аналіз вимог до мобільного додатку

Проектування програмного продукту

Проектування прототипу програмного продукту

Розробка прототипу програмного продукту

Розробка програмного продукту

Тестування програмного продукту

Загальні висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

Презентація

Відео роботи програми

КАЛЕНДАРНИЙ ПЛАН

Стадія	Зміст	Строки виконання
Технічне завдання	Постановка задачі, збір інформації, вибір та обґрунтування критеріїв розробки. Попередній вибір методів рішення задач. Визначення вимог до технічних засобів. Узгодження і затвердження технічного завдання.	31.01.23 - 18.02.23
Робочий проект	Програмування та відлагодження програми.	19.02.23 - 20.05.23
	Тестування програми	20.05.23 - 27.05.23
	Розробка, узгодження і затвердження програмної документації.	27.05.23 - 12.06.23
	Подання кваліфікаційної роботи до кафедри	07.06.23
	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	26.06.23

Студент

Артем ЮРТЕЕВ

(підпис)

(Ім'я ПРІЗВИЩЕ)

Керівник роботи

доц. Олександр ІВАНОВ

(підпис)

(Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка складається з 7 розділів:

- вступ – в даному розділі описується сутність розробки, її актуальність.

Складається з 1 сторінки;

- збір та аналіз вимог до мобільного додатку – у цьому розділі ми розглядаємо та описуємо аналоги, аналізуємо їх оцінки доповнюючи це плюсами та мінусами та складаємо постановку подальшої роботи. Складається з 5 сторінок;

- проектування– у цьому розділі описуються задачі проекту, його функціональні вимоги, вхідні та вихідні дані, описується вибір мови програмування та підходу розробки, проектується та розробляється прототип мобільного додатку який буде основою нашого мобільного додатку. Складається з 26 сторінок;

- розробка – у цьому розділі, на основі прототипу, розробляється динаміка системи, екрани мобільного додатку, інтерфейс мобільного додатку, система навігації та дизайн. Складається з 8 сторінок;

Тестування та налагодження – у цьому розділі розроблений мобільний додаток тестується методами білої та чорної скриньки. Складається з 6 сторінок.

- загальні висновки – підсумки всієї роботи. Складається з 1 сторінки;

- список використаних джерел – включає в себе бібліографічний список використаної літератури. Складає 1 сторінки;

- додатки – містить технічне завдання.

Кількість таблиць: 14 штук. Кількість рисунків: 41 штуки.

Об'єктом розроблення є програмний продукт «Автоматизована система продажу квитків в кінотеатрі».

Метою роботи є розробка програмного продукту.

Методом для розв'язання задачі є використання об'єктно орієнтованого програмування.

Отриманим результатом є розроблений програмний продукт.

Значенням роботи є розробка програми, яка зможе автоматизувати роботу працівників кінотеатру.

ЗМІСТ

ВСТУП	7
1. РОЗДІЛ. ЗБІР ТА АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ПРОДУКТУ	8
1.1. Опис аналогів	8
1.2. Постановка задачі	11
Висновки до розділу:	11
2. РОЗДІЛ. ПРОЕКТУВАННЯ	13
2.1. Задачі проекту	13
2.2. Функціональні вимоги.....	13
2.3. Вхідні та вихідні дані	14
2.4. Вибір мови програмування	17
2.5. Проектування архітектури системи	18
2.5.1. Моделювання словника системи.....	18
2.5.2. Моделювання розподілу обов’язків у системі.....	19
2.5.3. Моделювання залежностей.....	20
2.6. Проектування екранів програми	24
2.7. Проектування інтерфейсу користувача	24
Висновки до розділу:	28
3. РОЗДІЛ. РОЗРОБКА	30
3.1. Розробка інтерфейсу користувача.....	30
3.2. Розробка методів та конструювання алгоритмів	33
Висновки до розділу.....	37
4. РОЗДІЛ. ТЕСТУВАННЯ ТА НАЛАГОДЖЕННЯ	38
4.1. Вибір методів тестування.....	38
4.2. Тестування	38
Висновки до розділу.....	45
Загальні висновки	46
Список використаних джерел	47
Додаток А	50
Додаток Б	63

ВСТУП

«Automated ticket sales system in the cinema» - це самостійно розроблена програма, призначенням якої є автоматизація роботи працівників кінотеатру. А саме автоматизація продажу квитків. Програма призначена допомагати у роботі працівникам кінотеатру. Користувач програми може додавати до неї кінофільми, створювати їх короткий опис і на основі доданих фільмів створювати розклад сеансів, додавати вартість квитків та продавати їх.

Для продажу квитків в програмі створена схема кінозалу. Вона показує зайняті та вільні місця на певний сеанс фільму, що допомагає відвідувачам легше обирати собі місце для перегляду фільму, а також бачити вартість обраних ними місць. Реалізована і можливість повернення квитків на сеанс.

Функціональне призначення – програмний продукт має додавати, зберігати та видаляти дані щодо доданих фільмів, сеансів, їх розкладу та білетів. Відтворювати розклад, схему кінозалу, опис фільмів, вартість квитків.

Експлуатаційне призначення – за допомогою програмного продукту відбувається створення розкладу сеансів у кінотеатрі та продаж квитків до них, що автоматизує роботу працівників кінотеатру, покращує якість обслуговування відвідувачів.

Актуальність роботи: на сьогоднішній день кінофільми, мультфільми та серіали є одним із найпоширеніших видів дозвілля людей. Для їх перегляду багато людей щодня відвідують кінотеатри. Програмний продукт задовольняє потреби, як у професіональній діяльності так і у проведенні дозвілля людей.

1. РОЗДІЛ. ЗБІР ТА АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ПРОДУКТУ

1.1.Опис аналогів

Пошук аналогів відбуватиметься в пошуковій системі інтернет «Google»

1. Програма SERVIO Tickets – це сучасна система автоматизації кінотеатрів, театрів, концерт-холів та інших розважальних комплексів, діяльність яких пов'язана з реалізацією квитків. SERVIO Tickets складається з комплексу програмних інструментів, що дозволяють вирішити оперативні завдання підприємства, налагодити роботу кожного з підрозділів (продаж квитків, валідатор, інформаційна панель для відвідувачів) та об'єднати їх у єдиний механізм, що працює.

Загальний опис функціоналу:

- гнучка система налаштування сеансів, розкладу фільмів, вистав, концертів;
- швидкий доступ касира до інформації про фільм чи спектакль, тривалість сеансів;
- зручне відображення вільних/зайнятих місць у залах, їхнє бронювання;
- інформаційний монітор для відвідувачів із виведенням доступних місць у залі;
- синхронізація із сайтом;
- відображення інформації про сеанси на ТВ-панелях;
- мобільний додаток-валідатор квитків по штрих-кодам для проходу відвідувачів до зали;
- автоматичне вивантаження даних у міжнародну систему ComScore;
- звітність з продажу, сеансів.

Вигляд інтерфейсу представлений на Рисунок 1. 1 та Рисунок 1. 2



Рисунок 1. 1 – Огляд першого аналогу, формат №1



Рисунок 1. 2 – Огляд першого аналогу, формат №2

2. Квитковая система управления "SaleTicket" - программа автоматизации продажи билетов для кинотеатров, театров, стадионов, цирков та інших підприємств, відвідування яких здійснюється за вхідними квитками. Програма вирішує завдання комплексного управління, у тому числі мережами та складними багатозальними структурами.

Загальний опис функціоналу:

- підвищення продуктивності праці та зниження ймовірності помилок персоналу, зайнятого продажем та розповсюдженням квитків, за рахунок надійного оперативного інформаційного забезпечення та автоматизації рутинних операцій на всіх етапах;
- підвищення якості та швидкості обслуговування глядачів за рахунок впровадження нових технологій, зменшення черг та часу очікування клієнта біля каси, впровадження нових способів продажу квитків онлайн через інтернет, мобільні програми;
- поліпшення іміджу кінотеатру в очах клієнтів, формування образу прогресивної та високотехнологічної установи;
- точний та надійний метод ведення обліку та контролю продажу квитків;
- припинення всіляких форм зловживань персоналом, розкрадань та недобросовісного виконання своїх обов'язків;
- інформаційна підтримка прийняття маркетингових рішень за рахунок розвиненого аналітичного інструментарію та автоматизації складання різних звітів;
- забезпечення обліку коштів від реалізації квитків та отримання звітів в автоматичному режимі;
- забезпечення ефективного контролю відвідуваності кінотеатру та одержаного доходу, управління репертуаром, дисконтно-бонусними програмами;

Вигляд інтерфейсу представлений на Рисунок 1. 3



Рисунок 1. 3 – Огляд другого аналогу

1.2. Постановка задачі

Програмний продукт «Автоматизована системи продажу квитків в кінотеатрі», що розробляється, має виконувати функції створення розкладу сеансів показу фільмів, та продажу квитків для кінотеатру.

В залежності від введених даних програма має їх виводити на екран та зберігати.

Користувач, використовуючи цю програму, може додавати фільму та опис до них. Створювати розклад сеансів показу фільмів, ціну на квитки та продавати їх. Продаж здійснюється шляхом виведення на екран схеми кінозалу, де відвідувач бачить схему кінозалу, опис фільму, вільні та зайняті місця, їх вартість, і виходячи з цих даних може придбати для себе квиток.

Висновки до розділу:

Під час опису аналогів було проаналізовано роботу програмних продуктів та їх функціональні можливості. Опираючись на них ми можемо скласти список основних функціональних вимог та описати план для подальшого проектування та розробки програмного продукту.

2. РОЗДІЛ. ПРОЕКТУВАННЯ

2.1. Задачі проекту

На основі аналізу аналогів виділяємо наступні задачі проекту:

- проектування архітектури системи;
 - проектування та створення класів що будуть позначати білет, фільм, та сеанс показу фільму;
- проектування інтерфейсу користувача;
- створення форм які являють собою інтерфейс користувача;
- розробка логіки програми;

2.2. Функціональні вимоги

Програмний продукт повинен:

- забезпечувати можливість додавання фільмів та їх описів до програми;
- забезпечувати можливість видалення фільмів та їх описів до програми;
- забезпечувати можливість створення розкладу сеансів на показ кінофільмів і встановлення їх вартості;
- забезпечувати можливість видалення розкладу сеансів на показ кінофільмів;
- забезпечувати можливість вибору фільму з розкладу та виводити опис цього фільму екран, відображати на схемі кінозалу вільні та зайняті місця, а також їх вартість;
- забезпечувати можливість повернення придбаних відвідувачами квитків;
- забезпечувати збереження введених даних;

Розроблено діаграму прецедентів, див. Рисунок 2. 1

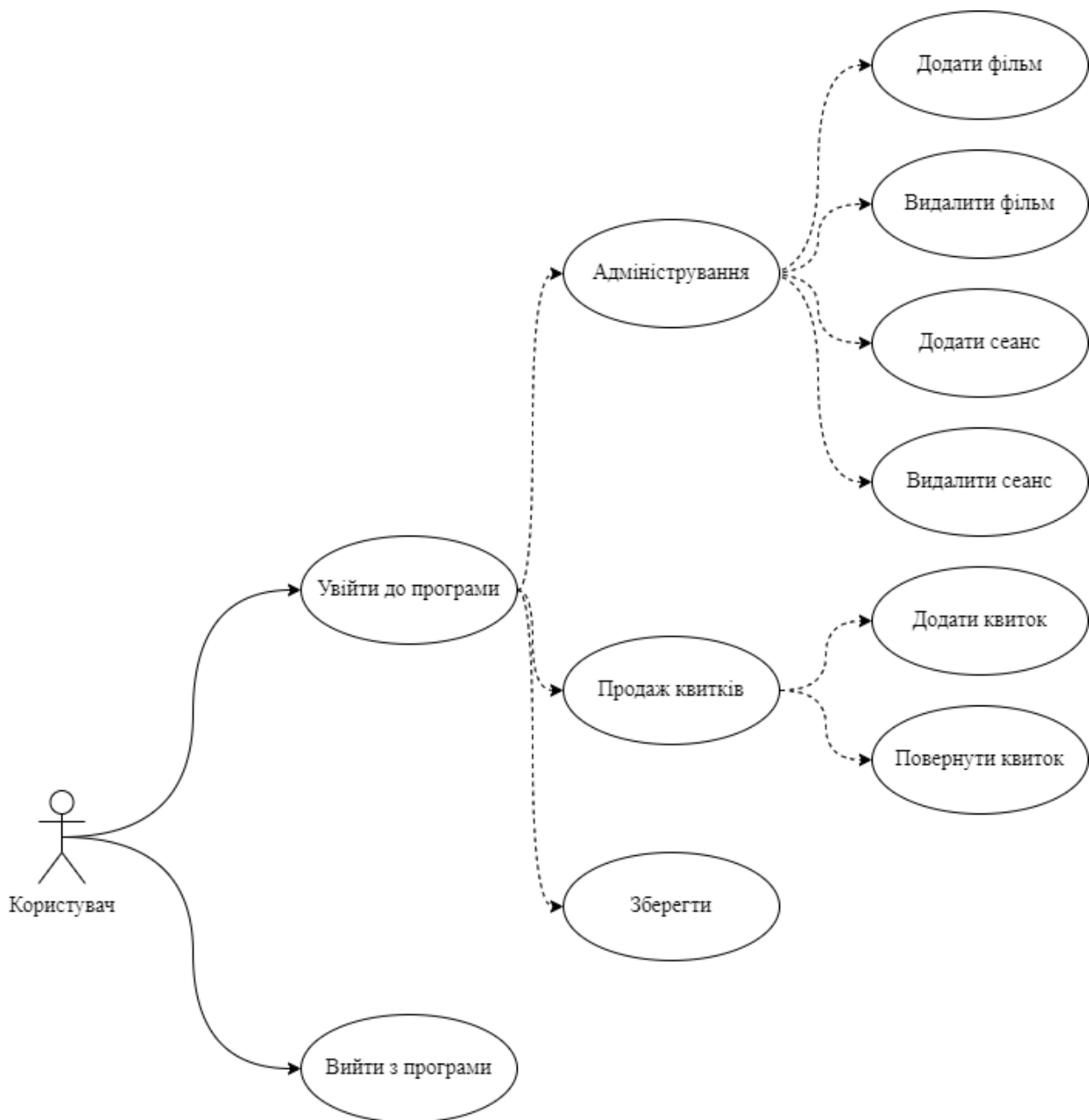


Рисунок 2. 1 – Діаграма прецедентів

2.3.Вхідні та вихідні дані

Вхідні данні:

Вхідні данні частини програми, де виконується додавання фільму, див. Таблиця 2. 1

Додавання фільму			
Номер	Назва	Тип	Вимоги
1	Назва	string	Не пуста строка
2	Вік		

3	Жанр		
4	Країна		
5	Рік випуску		
6	Режисер		

Таблиця 2. 1 – Вхідні данні частини програми, де виконується додавання фільму

Вхідні дані видалення фільму див. Таблиця 2. 2

Видалення фільму			
Номер	Назва	Тип	Вимоги
1	Назва	Елемент випадаючого меню	

Таблиця 2. 3 – Вхідні дані видалення фільму

Вхідні данні до частини програми, де виконується додавання сеансу див. Таблиця 2.

3

Додавання сеансу			
Номер	Назва	Тип	Вимоги
1	Назва фільму	Елемент випадаючого меню	
2	Формат		
3	Дата	string	День тижня, число. Прик: Понеділок, 22
4	Час сеансу	Заповнення або вибір часу в елементі	Можливості обмежені елементом форми
5	Вартість квитка	int	Будь-яке число

Таблиця 2. 4 - Вхідні данні до частини програми, де виконується додавання сеансу

Вхідні дані видалення сеансу див. Таблиця 2. 4

Видалення сеансу			
Номер	Назва	Тип	Вимоги
1	Дата	Елемент випадаючого меню	
2	Час	Елемент випадаючого меню	

Таблиця 2. 5 – Вхідні дані видалення сеансу

Вхідні данні до частини програми, де обирається сеанс, для подальшого вибору місць в кінозалі див. Таблиця 2. 5

Вибір сеансу			
Номер	Назва	Тип	Вимоги
1	Дата	Елемент випадального меню	
2	Фільм		
3	Час сеансу		

Таблиця 2. 6 – Вхідні данні до частини програми, де обирається сеанс, для подальшого вибору місць в кінозалі

Обрання місць в кінозали проходить шляхом натискання кнопок, що симулюють місця в кінозалі. Після обрання місць натискається кнопка купити

Вхідні дані повернення квитків див. Таблиця 2. 6

Повернення квитків			
Номер	Назва	Тип	Вимоги
1	Дата	Елемент випадального меню	
2	Фільм		
3	Час сеансу		
4	Ряд		
5	Місце		

Таблиця 2. 7 – Вхідні дані повернення квитків

Вихідні данні:

Вихідні данні частини програми, де виконується додавання фільму:

- Таблиця з даними про доданий фільм.

Вихідні данні до частини програми, де виконується додавання сеансу:

- Таблиця з даними про доданий сеанс.

Вихідні данні до частини програми, де обирається сеанс, для подальшого вибору місць в кінозалі:

- Дані про фільм в таблиці.

Зміна кольору кнопок(місць в кінозалі) при їх натисканні. Кнопки, місця до яких вже куплені також відрізняються кольором та недоступні.

2.4. Вибір мови програмування

Для розробки програми була використана мова програмування C# з використанням інструменту Windows Forms.

Windows Forms — інтерфейс програмування додатків (API), відповідальний за графічний інтерфейс користувача і є частиною Microsoft .NET Framework. Даний інтерфейс спрощує доступ до елементів інтерфейсу Microsoft Windows за допомогою створення обгортки для Win32 API в керованому коді.

Перевагами Windows Forms є:

Простота вивчення та використання: Windows Forms надає простий та інтуїтивно зрозумілий спосіб створення інтерфейсу користувача за допомогою графічного дизайнера та програмування мовою C# або іншими мовами .NET. Він має звичний для Windows підхід до побудови інтерфейсу, заснований на елементах управління (кнопки, текстові поля, списки і т. д.), що робить його легким для використання розробниками.

Швидка розробка: Windows Forms пропонує набір готових елементів керування та функцій, які можна легко додавати та налаштовувати на формах програми.

Широка підтримка: Windows Forms є частиною платформи .NET Framework, яка має велику підтримку з боку Microsoft та спільноти розробників. Це дає швидкий доступ до ресурсів та документації.

Інтеграція з Windows: Windows Forms тісно інтегрований з операційною системою Windows і надає доступ до багатьох можливостей платформи, таких як робота з файлами, реєстром, мережею, базами даних тощо. Він також підтримує використання функцій Windows API та COM-компонентів.

Можливості налаштування та розширення: Windows Forms володіє гнучкою системою налаштування зовнішнього вигляду елементів керування та підтримкою власних розширень. Ви можете створювати власні елементи керування, налаштовувати їх зовнішній вигляд, створювати власні теми та дизайн програми.

2.5.Проектування архітектури системи

2.5.1. Моделювання словника системи

Ідентифіковані сутності: Film, Ticket, Session, Form1, Form2, Program.

Ідентифіковані обов'язки:

- Film: збереження даних про додані фільми;
- Ticket: збереження даних про додані білети;
- Session: збереження даних про додані сеансів;
- Form1: виведення даних про розклад сеансів, опис фільмів, схему кінозалу, введення даних про білети, що купили та повернули;
- Form2: введення та виведення даних про фільми та сеанси;
- Program: запуск програми.

Атрибути та операції, необхідні для виконання обов'язків кожної

сутності-класу наведемо у Таблиця 2. 7

Сутність	Атрибути	Методи
Film	name – назва фільму, тип даних - String; MPA_rating_system – віковий рейтинг фільму, тип даних - String; genre – жанр фільму, ; country – країна виробник фільму, тип даних - String; year_of_issue – рік випуску фільму, тип даних - String; director – режисер фільму тип даних - String;	Створення сутності
Ticket	film_name – назва фільму на який придбаний білет, тип даних - String; seat_code – код місця в залі, тип даних - String; ticket_value – вартість білету, тип даних - int; date – дата сеансу, тип даних - String; time – час сеансу, тип даних - String;	Створення сутності
Session	film_name – назва фільму на для сеансу, тип даних - String;	Створення сутності

	date – дата сеансу, тип даних - String; time – час сеансу, тип даних - String; ticket_value – вартість квитка, тип даних - int; format – формат фільму «2D» «3D» , тип даних - String;	
Form1	випадаючі списки; кнопки; поля для виведення даних; seat – список який зберігає «код» місця в кінозалі, тип даних – List«string»; seat_number – частина «коду» місця в кінозалі, номер в ряду, тип даних - int; seat_row – частина «коду» місця в кінозалі, номер ряду, тип даних - int; ticket_value – вартість квитка, змінна виводиться в інтерфейс користувача тип даних - int; film – список доданих фільмів, тип даних - List«string»; session – список доданих сеансів, тип даних - List«string»;	Збереження Завантаження Купити квиток Повернути квиток Перехід к Form2 Натискання на кнопки «Схеми кінозалу»
Form2	випадаючі списки; кнопки; поля для виведення даних; film_counter – кількість фільмів, тип даних - int; session_counter – кількість сесій, тип даних - int;	Збереження Завантаження Додати сеанс Видалити сеанс Додати фільм Видалити фільм Перехід к Form1
Program		Запуск програми

Таблиця 2. 8 – Сутність-клас

2.5.2. Моделювання розподілу обов’язків у системі

- Form1, Film – виведення даних, додавання та видалення;
- Form1, Session – виведення даних, додавання та видалення;
- Form1, Ticket – виведення даних, додавання та видалення;
- Form2, Film – виведення даних, додавання та видалення;
- Form2, Session – виведення даних, додавання та видалення;
- Program, Form1 – метод класу Program завантажує інтерфейс Form1.

2.5.3. Моделювання залежностей

Моделювання залежностей див. Таблиця 2. 8

Клас який зв'язується	Клас з яким зв'язується	Тип зв'язку
Program	Form1	Залежність
Form1	Ticket	Асоціація
	Session	
	Film	
Form1	Form2	Асоціація
Form1	Film	Асоціація
	Session	

Таблиця 2. 9 – Моделювання залежностей

Представимо заключний результат моделювання у вигляді діаграми

класів див. Рисунок 2. 2

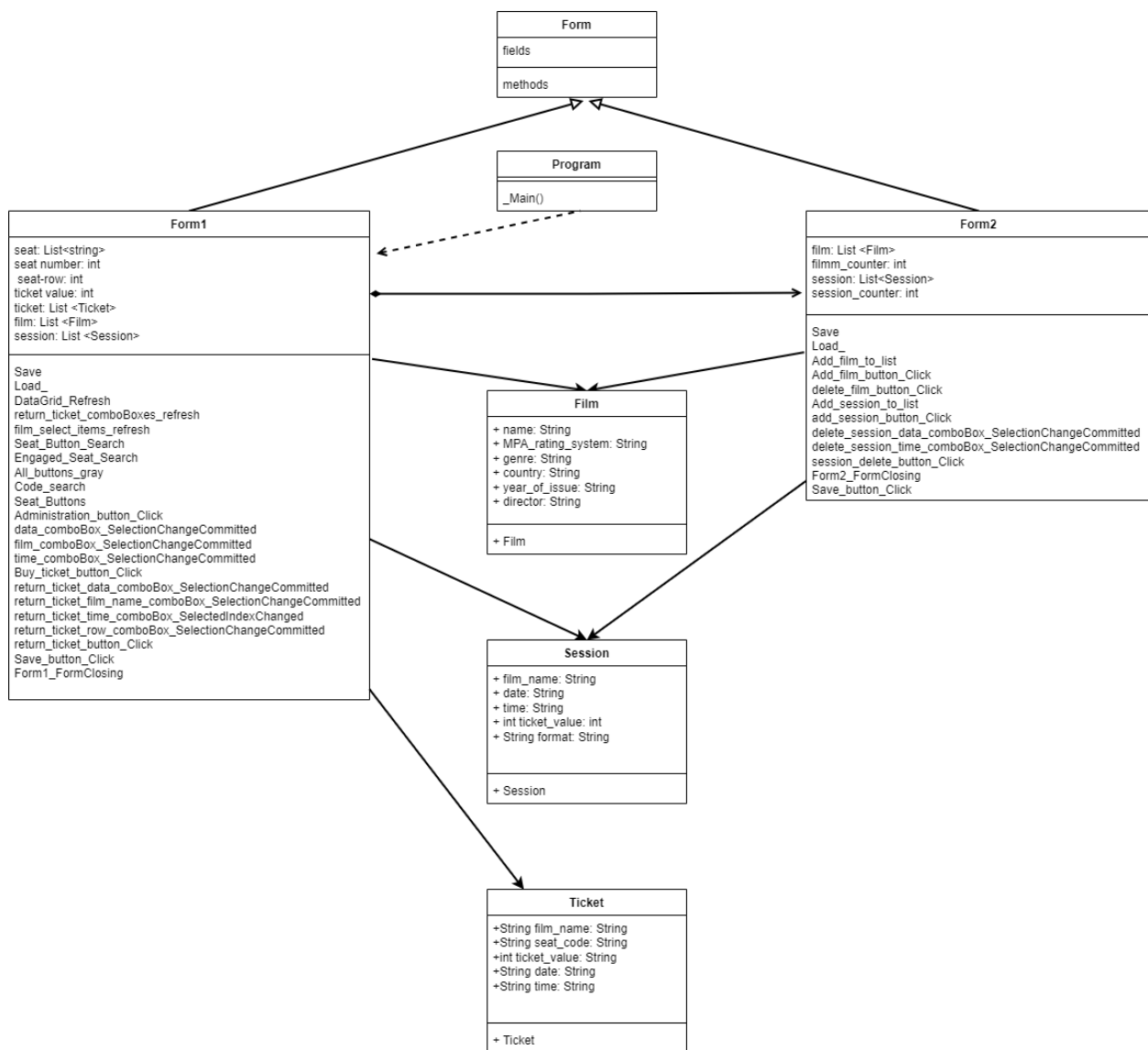


Рисунок 2. 3 – Діаграма класів

CRC картка для класу Program, див. Таблиця 2. 9.

Базовий клас	Похідні класи
Form	Відсутні
Обов'язки	Зв'язки
Запуск програми, шляхом запуску Form1	Form1

Таблиця 2. 10 – CRC картка для класу Program

CRC картка для класу Form1, див. Таблиця 2. 10.

Базовий клас	Похідні класи
Form	Відсутні
Обов'язки	Зв'язки

Збереження об'єкти класу Ticket в бінарний файл Завантаження об'єкти класу Ticket з бінарного файлу Завантаження об'єкти класу Session з бінарного файлу Завантаження об'єкти класу Film з бінарного файлу Створювати список об'єктів класу Ticket Створювати список об'єктів класу Session Створювати список об'єктів класу Film Реалізація методів компонентів форми Створювати список «кодів» місць в кінозалі Введення та виведення на екран інформації	Form2, Session, Ticket, Film
--	------------------------------

Таблиця 2. 11 – CRC картка для класу Form1

CRC картка для класу Form2, див. Таблиця 2. 11.

Базовий клас	Похідні класи
Form	Відсутні
Обов'язки	Зв'язки
Збереження об'єкти класу Session в бінарний файл Збереження об'єкти класу Film в бінарний файл Завантаження об'єкти класу Session з бінарного файлу Завантаження об'єкти класу Film з бінарного файлу Створювати список об'єктів класу Session Створювати список об'єктів класу Film Реалізація методів компонентів форми Введення та виведення на екран інформації	Session, Ticket

Таблиця 2. 12 – CRC картка для класу Form2

CRC картка для класу Film, див. Таблиця 2. 12.

Базовий клас	Похідні класи
Відсутні	Відсутні
Обов'язки	Зв'язки
Збереження назви фільму Збереження вікової категорії фільму Збереження жанру фільму збереження країни виробництва фільму Збереження року випуску фільму Збереження режисера фільму	

Таблиця 2. 13 – CRC картка для класу Film

CRC картка для класу Session, див. Таблиця 2. 13.

Базовий клас	Похідні класи
Відсутні	Відсутні
Обов'язки	Зв'язки
Збереження назви фільму для сеансу Збереження дати сеансу Збереження часу сеансу Збереження вартості квитка на сеанс Збереження формату сеансу	

Таблиця 2. 14 – CRC картка для класу Session

CRC картка для класу Ticket, див. Таблиця 2. 14.

Базовий клас	Похідні класи
Відсутні	Відсутні
Обов'язки	Зв'язки
Збереження назви фільму для квитка Збереження коду місця Збереження вартості квитка Збереження дати сеансу для квитка Збереження часу сеансу для квитка	

Таблиця 2. 15 – CRC картка для класу Ticket

2.6.Проектування екранів програми

Інтерфейс користувача буде представлений двома формами:

- форма продажу квитків;
- форма адміністрування.

2.7.Проектування інтерфейсу користувача

Прототип програми повинен мати такі компоненти:

- “ComboBox”;
- “DataGridView”;
- “Button”;
- “GroupBox”;
- “TabControl”;
- “DateTimePicker”;
- “TextBox”;
- “NumericUpDown”;
- “Label”.

Компонент ComboBox – компонент інтерфейсу, при натисканні на який з’являється випадаюче меню.

Ескіз елементу ComboBox, зображено на Рисунок 2. 3.



Рисунок 2. 4 – Ескіз елементу ComboBox

Компонент DataGridView – компонент інтерфейсу, який виводить дані на форму.

Ескіз елементу DataGridView, зображено на Рисунок 2. 4.

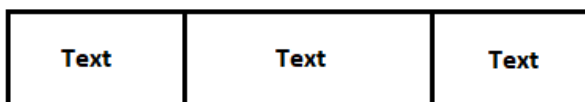


Рисунок 2. 5 – Ескіз елементу DataGridView

Компонент Button – компонент інтерфейсу, при натисканні на який виконуються прописані дії.

Ескіз елементу Button, зображено на Рисунок 2. 5.



Рисунок 2. 6 – Ескіз елементу Button

Компонент GroupBox – компонент інтерфейсу, який створений для групування інших елементів на формі.

Компонент TabControl – компонент інтерфейсу, для створення вкладок та вкладеного інтерфейсу.

Компонент DateTimePicker – компонент, при натисканні на який надається можливість встановити час або дату на елементі.

Ескіз елементу DateTimePicker, зображено на Рисунок 2. 6.

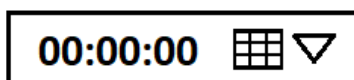


Рисунок 2. 7 – Ескіз елементу DateTimePicker

Компонент TextBox – компонент, який створений для введення даних.

Ескіз елементу TextBox, зображено на Рисунок 2. 7.

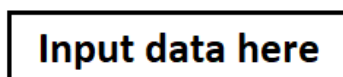


Рисунок 2. 8 – Ескіз елементу TextBox

Компонент `NumericUpDown` – компонент який дозволяє користувачеві вибирати числове значення з обмеженого діапазону.

Ескіз елементу `NumericUpDown`, зображено на Рисунок 2. 8.



Рисунок 2. 9 – Ескіз елементу `NumericUpDown`

Компонент `Label` – компонент, який створений для відображення текстової інформації у графічному інтерфейсі.

На Рисунок 2. 9 представлено ескіз форми для продажу квитків

 A rectangular form for purchasing movie tickets. At the top right, there are two buttons: 'Зберегти' (Save) and 'Адміністрування' (Administration). The main area contains a large box labeled 'Схема кінозалу' (Cinema layout) and a smaller box to its right labeled 'Місце ряд вартість' (Seat row price). Below the layout box is a 'Купити' (Buy) button. At the bottom, there are five dropdown menus labeled 'Дата' (Date), 'Фільм' (Movie), 'Час сеансу' (Showtime), 'Ряд' (Row), and 'Місце' (Seat), followed by a 'Повернути' (Return) button.

Рисунок 2. 10 – Ескіз форми для продажу квитків

На Рисунок 2. 10 представлено ескіз форми для додавання сеансів

До квитків		Зберегти	
Сеанс		Фільм	
Назва фільму	<input type="text"/>	Вартість	<input type="text"/>
Дата	<input type="text"/>	Формат	<input type="text"/>
Час	<input type="text"/>	Додати	
Сеанси			
Дата	Час	Видалити	
<input type="text"/>	<input type="text"/>		

Рисунок 2. 11 – Ескіз форми для додавання сеансів

На Рисунок 2. 11 представлено ескіз форми для додавання фільмів

До квитків		Зберегти	
Сеанс		Фільм	
Назва	<input type="text"/>	Країна	<input type="text"/>
Вік	<input type="text"/>	Рік випуску	<input type="text"/>
Жанр	<input type="text"/>	Режиссер	<input type="text"/>
Опис фільмів			Додати
Дата	Видалити		
<input type="text"/>			

Рисунок 2. 12 – Ескіз форми для додавання фільмів

На Рисунок 2. 12 приведена діаграма станів

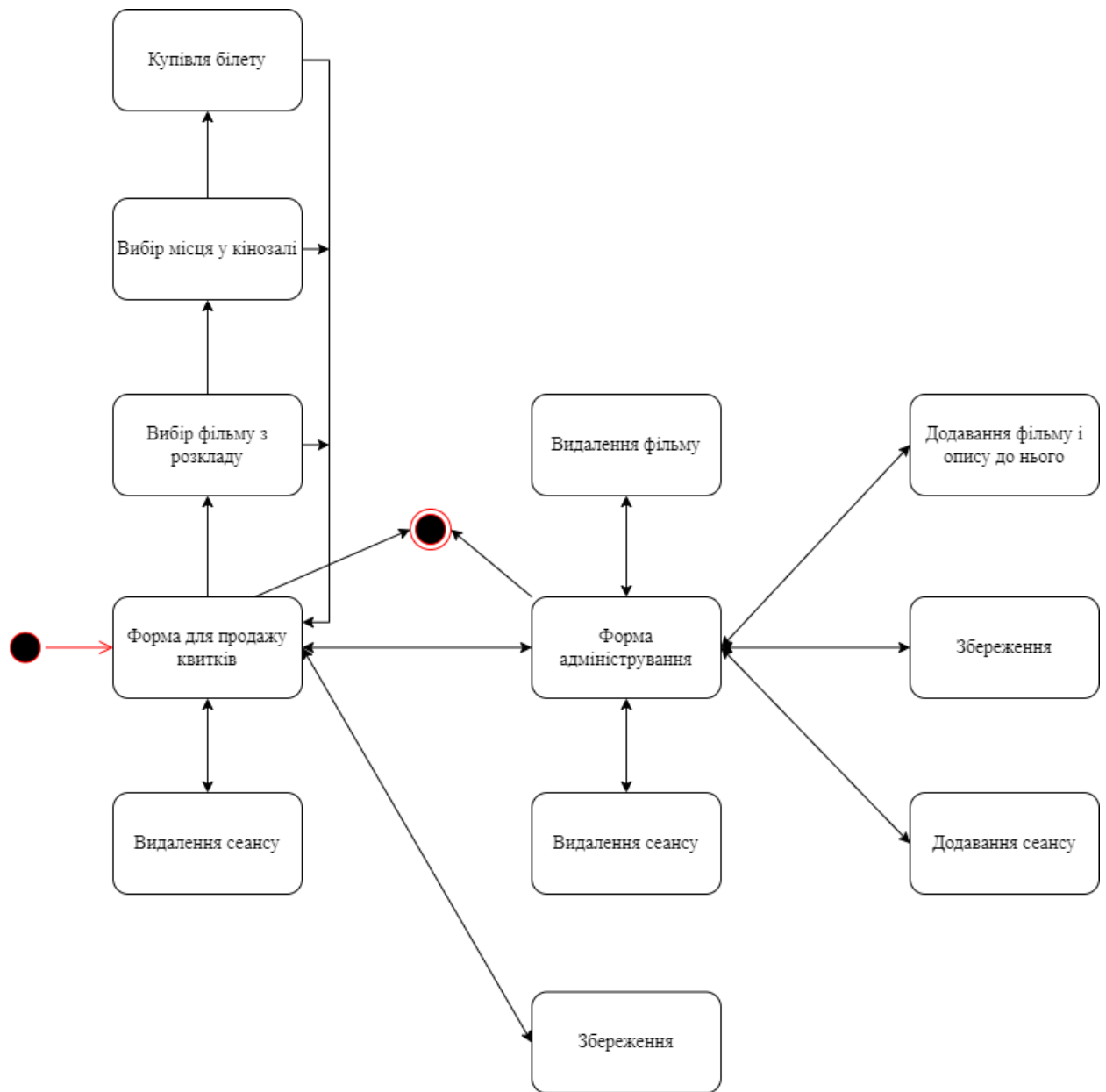


Рисунок 2. 13 – Діаграма станів

Висновки до розділу:

Після опису аналогів, в процесі проектування, було складено задачі проекту та описані функціональні вимоги разом з вхідними та вихідними даними до мобільного додатку.

До описаних характеристик була підібрана мова програмування та інструмент з якими найкраще можна виконати цю задачу.

У процесі проектування було спроектовано прототип програмного продукту. Для нього було створено:

- 2 форми;
- ескізи компонентів;
- спроектований інтерфейс користувача.

3. РОЗДІЛ. РОЗРОБКА

3.1. Розробка інтерфейсу користувача

Програма налічує дві форми:

- форма для продажу квитків;
- форма для адміністрування,

Форма для продажу складається з 57 компонентів:

- 8 компонентів “Label”, які використовуються для позначення випадających списків;
- 8 компонентів “ComboBox”;
- 4 компонентів “Button”;
- 28 компонентів “Button”, кожен з яких симулює місце в кінозалі;
- 2 компоненти DataGridView;
- компонент “GroupBox”, що групує схему кінозалу;
- компонент “GroupBox”, використаний в якості симуляції екрану в кінозалі.

Інтерфейс користувача при використанні форми для продажу квитків зображений на Рисунок 3. 1

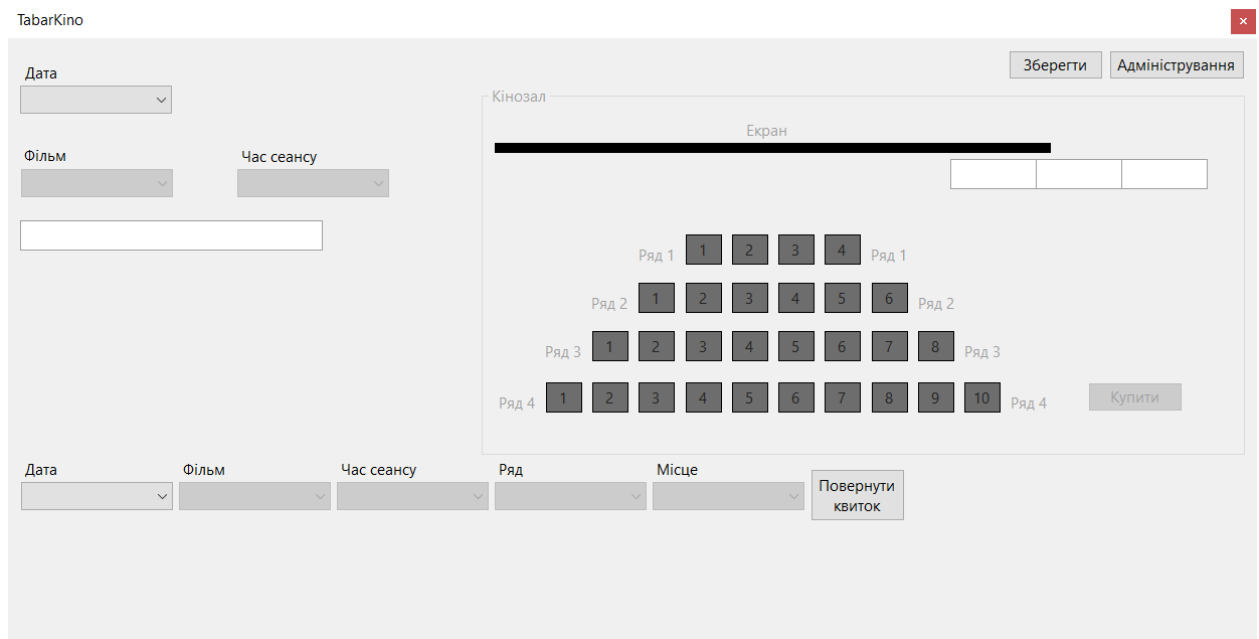


Рисунок 3. 1 – Інтерфейс користувача при використанні форми для продажу квитків

Форма для адміністрування складається з 3 компонентів

- 2 компоненти “Button”;
- компонент “GroupBox”.

Компонент “GroupBox” складається з двох вкладок «Сеанс» та «Фільм»

Вкладка «Сеанс» складається з 17 компонентів:

- 7 компонентів “Label”;
- 4 компоненти “ComboBox”;
- компонент “DateTimePicker”;
- компонент “NumericUpDown”;
- 2 компоненти “Button”;
- компонент “DataGridView”;
- компонент “TextBox”.

Інтерфейс користувача при використанні форми для адміністрування на вкладці «Сеанс» зображений на Рисунок 3. 2.

TabarKino

До квитків Зберегти

Сеанс Фільм

Назва фільму Вартість квитка 0

Дата(День тижня, число) Формат

Час сеансу 00:00:00 Додати

--	--	--	--	--

Дата Час Видалити сеанс

Рисунок 3. 2 – Інтерфейс користувача при використанні форми для адміністрування на вкладці «Сеанс»

Вкладка «Фільм» складається з 17 компонентів:

- 7 компонентів “Label”;
- 6 компонентів “TextBox”;
- 2 компонента “Button”;
- компонент “DataGridView”;
- компонент “ComboBox”.

Інтерфейс користувача при використанні форми для адміністрування на вкладці «Фільм» зображений на Рисунок 3. 3.

TabarKino

До квитків Зберегти

Сеанс **Фільм**

Назва Країна

Вік Рік випуску

Жанр Режисер

--	--	--	--	--	--

Назва

Рисунок 3. 3 – Інтерфейс користувача при використанні форми для адміністрування на вкладці «Фільм»

3.2.Розробка методів та конструювання алгоритмів

У цьому розділі будуть розглянуті основні розроблені методи та сконструйовані алгоритми, а також представлені різні підходи до створення оптимальних та надійних алгоритмічних рішень.

Метод «Пошуку кнопок місць в кінозалі»

Для реалізації схеми кінозалу, що зображена на рис.3.1. було додано 28 компонентів “Button”. Для того щоб об’єднати реалізацію всіх компонентів, і було розроблено метод, блок-схему якого зображено на Рисунок 3. 4.

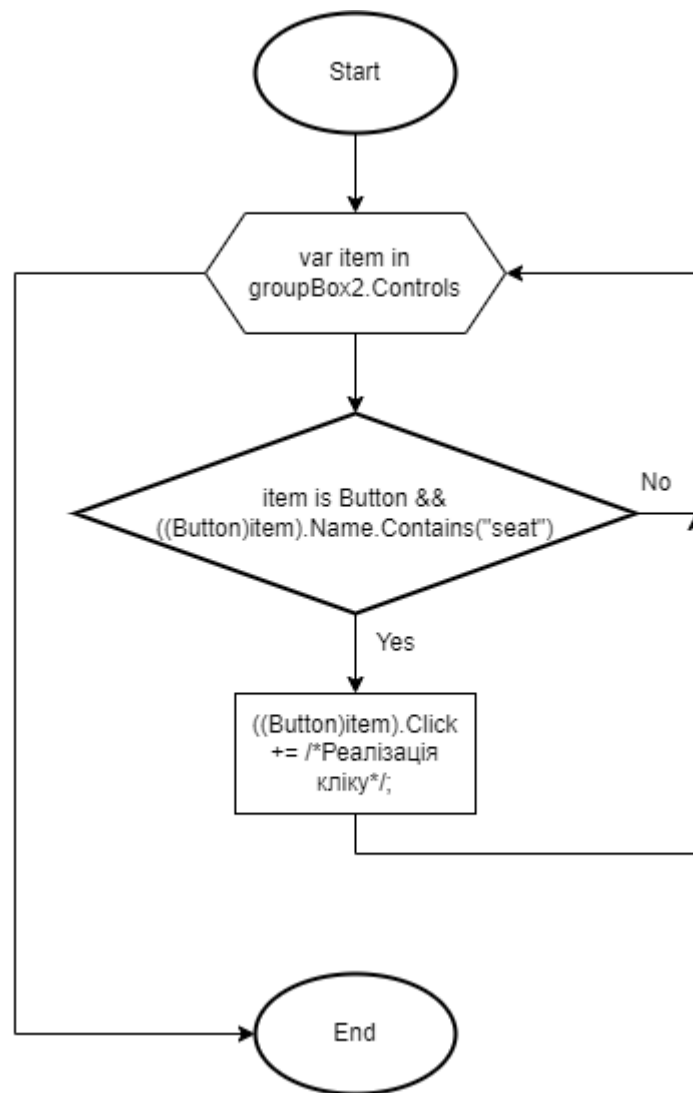


Рисунок 3. 4 – Блок-схема реалізації алгоритму схеми кінозалу

Метод «Купити квиток»

Метод запускається після натискання на кнопку «Купити» на формі для продажу квитків див. рис. 3.1. Вхідні дані отримуються випадających списків та після натискань кнопок на схемі кінозалу. Блок-схема методу на Рисунок 3. 5.

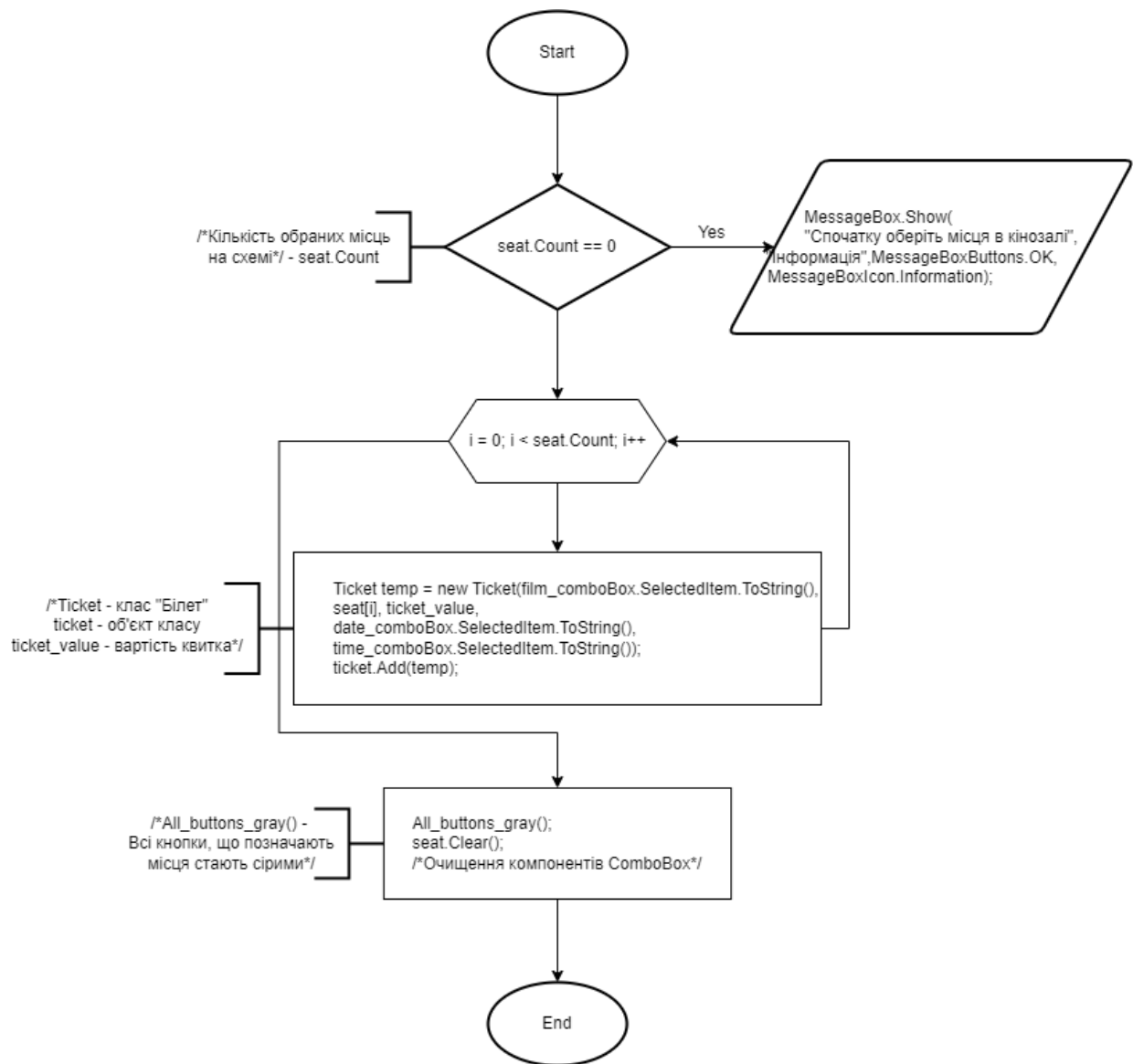


Рисунок 3. 5 – Блок-схема методу «Купити квиток»

Метод «Завантаження»

Метод запускається при завантаженні форми. Завантажуються об'єкти класів з бінарних файлів. Блок-схема методу на Рисунок 3. 6.

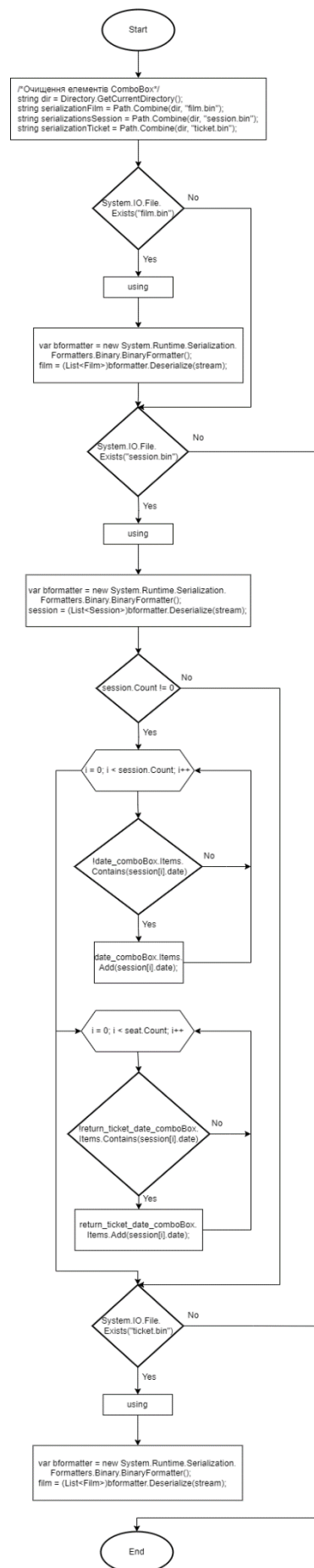


Рисунок 3. 6 – Блок-схема методу «Завантаження»

Метод «Збереження»

Метод запускається при натисканні кнопки зберегти. Зберігає об'єкти класів в бінарні файли. Блок-схема методу на Рисунок 3. 7.

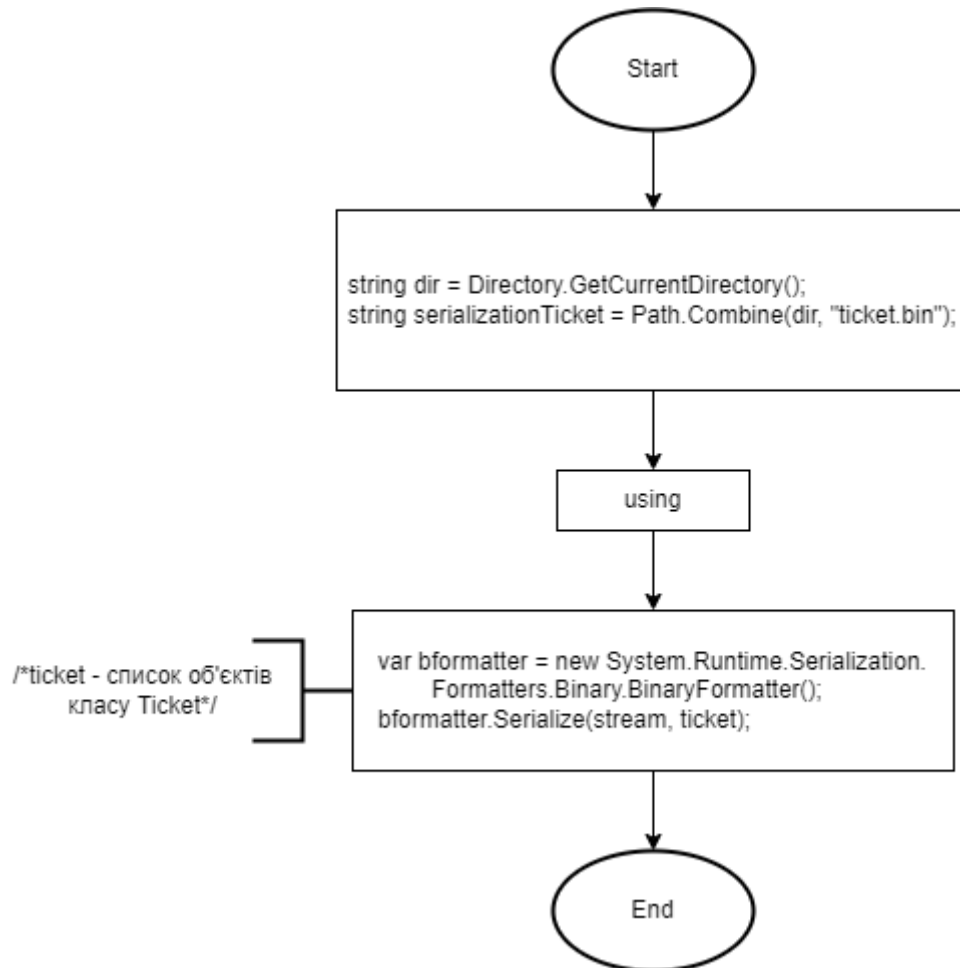


Рисунок 3. 7 – Блок-схема методу «Збереження»

Висновки до розділу

Після опису аналогів, та проектування було розроблено інтерфейс користувача та архітектуру системи.

У процесі розробки був розроблений робочий програмний продукт.

4. РОЗДІЛ. ТЕСТУВАННЯ ТА НАЛАГОДЖЕННЯ

4.1. Вибір методів тестування

Для тестування функцій було вибрано два методи:

- тестування чорною скринькою;

Тестування чорною скринькою базується на тому що нам не відомо як влаштована система – засноване на тестуванні роботи з зовнішніми інтерфейсами.

За допомогою цього метода ми зможемо знайти помилки таких категорій:

- помилкові реалізації функцій, або повна їх відсутність;
- помилки інтерфейсу системи, яку тестуємо;
- помилкова організація структури даних, або організації доступу;
- помилкова поведінка, або повна відсутність продуктивності системи.

Таким чином ми можемо концентруватись на роботі програми, а не на тому, як вона це робить.

У методі тестування чорною скринькою можна виділити такі переваги та недоліки:

Переваги:

- при тестуванні не потрібно знати мову програмування та вивчати особливості реалізації;
- тестування кінцевого продукту дає змогу визначити неточності у специфікації.

Недоліки:

- тестуються не всі шляхи виконання програми;
- деякі тести можуть виявитись надлишковими.

4.2. Тестування

Тест 1.

Очікуваний результат: програма спрацює коректно, білет буде придбаним.

Вхідні дані:

1. Обрані елементи з випадаючих списків див. Рисунок 4. 1.

Рисунок 4. 1 – Тест 1, обрані елементи з випадаючих списків

2. Обране місце в кінозалі на схемі шляхом натискання на кнопку, що позначає місце в кінозалі, кнопка змінила колір на зелений, див. Рисунок 4. 2.

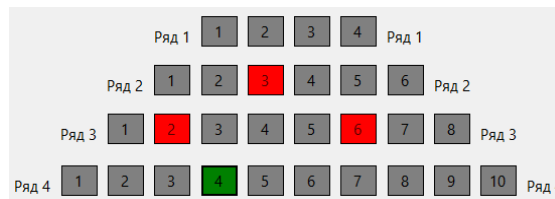


Рисунок 4. 2 – Тест 1, обране місце в кінозалі

Вихідні дані:

1. Після обрання елементів з випадаючих списків, на екран вивівся опис фільму, див. Рисунок 4. 3.

Рисунок 4. 3 – Тест 1, вихідні дані, опис фільму

2. На схемі кінозалу виводиться інформація щодо вартості обраного місця, див. Рисунок 4. 4.

Ряд 4	Місце 4	100 грн

Рисунок 4. 4 – Тест 1, вихідні дані, інформація щодо вартості обраного місця

3. Після натискання кнопки купити, кнопка, що позначає обране місце змінюється на червоний, місце в кінозалі «бронюється», див. Рисунок 4. 5.

		Ряд 1	1	2	3	4		Ряд 1			
	Ряд 2	1	2	3	4	5	6	Ряд 2			
Ряд 3	1	2	3	4	5	6	7	8	Ряд 3		
Ряд 4	1	2	3	4	5	6	7	8	9	10	Ряд 4

Рисунок 4. 5 – Тест 1, вихідні дані, придбане місце

Результат тесту: тест виконано коректно.

Тест 2.

Очікуваний результат: подальший вибір місць в кінозалі не можливий, елементи не активні, адже не заповнені всі випадаючі списки.

Вхідні дані:

1. Обрані всі елемент випадаючих списків окрім одного, див. Рисунок 4. 6.

Дата	
Понеділок, 20	
Фільм	Час сеансу
Кіно	
<div> <div>Назва: Кіно</div> <div>Рік виходу: 2000</div> <div>Віковий рейтинг: 12+</div> <div>Жанр: Драма</div> <div>Режисер: Режисер</div> <div>Країна: США</div> </div>	

Рисунок 4. 6 – Тест 2, вхідні дані

Вихідні дані:

1. Після обрання елементів з випадаючих списків, на екран вивівся опис фільму.

Результат тесту: тест виконано коректно, див. Рисунок 4. 7.

The screenshot shows a cinema booking interface. On the left, there are dropdown menus for 'Дата' (Monday, 20), 'Фільм' (Кіно), and 'Час сеансу'. Below these is a table with movie details: 'Назва: Кіно', 'Рік виходу: 2000', 'Віковий рейтинг: 12+', 'Жанр: Драма', 'Режисер: Режисер', and 'Країна: США'. On the right, there is a 'Кінозал' (Cinema Hall) section with a diagram of the hall and a seating chart. The chart shows four rows (Ряд 1 to Ряд 4) with seats numbered 1 to 10. A 'Купити' (Buy) button is visible at the bottom right of the seating chart.

Рисунок 4. 7 – Тест 2, результат тесту

Тест 3

Очікуваний результат: квиток буде повернуто, заброньоване місце в залі звільниться.

Вхідні дані:

1. Заброньоване місце, ряд 4, місце 4, кнопка підсвічена червоним кольором, див.

Рисунок 4. 8;

The screenshot shows the cinema seating chart from Figure 4.7. The seat at Row 4, Seat 4 is highlighted in red, indicating it is the selected seat for the test.

Рисунок 4. 8 – Тест 3, вхідні дані, заброньоване місце

2. Обрані елементи випадаяючих списків, див. Рисунок 4. 9;

The screenshot shows the cinema booking form with the following selected elements: 'Дата' (Monday, 20), 'Фільм' (Кіно), 'Час сеансу' (15:00:00), 'Ряд' (4), and 'Місце' (4). A 'Повернути квиток' (Return ticket) button is visible at the bottom right.

Рисунок 4. 9 – Тест 3, вхідні дані, обрані елементи випадаяючих списків

Вихідні дані:

1. Місце в кінозалі звільнилося, кнопка підсвічена сірим кольором, див. Рисунок 4.

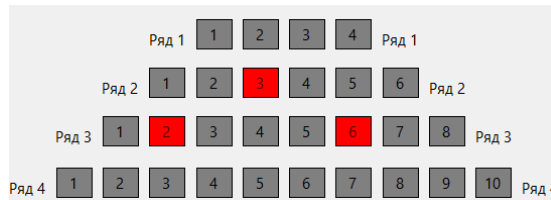


Рисунок 4. 10 – Тест 3, результат тесту

Результат тесту: тест виконано коректно.

Тест 4.

Очікуваний результат: виведення повідомлення про помилку.

Вхідні дані:

1. Обрані всі елементи з випадаяючих списків окрім одного, див. Рисунок 4. 11.

Дата	Фільм	Час сеансу	Ряд	Місце	Повернути квиток
Понеділок, 20	Кіно	15:00:00	3		

Рисунок 4. 11 – Тест 4, вхідні дані

Вихідні дані:

1. Інформаційне повідомлення про необхідність заповнити всі випадаяючі списки, див. Рисунок 4. 12.

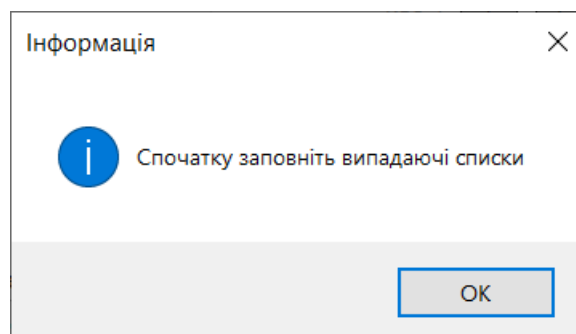


Рисунок 4. 12 – Тест 4, вихідні дані

Результат тесту: тест виконано коректно.

Тест 5.

Очікуваний результат: додано новий фільм, виведено опис фільму

Вхідні дані:

1. Заповнено текстові поля, див. Рисунок 4. 13.

Назва	<input type="text" value="Кіно1"/>	Країна	<input type="text" value="Франція"/>
Вік	<input type="text" value="12+"/>	Рік випуску	<input type="text" value="2002"/>
Жанр	<input type="text" value="Драма"/>	Режисер	<input type="text" value="Режисер1"/>

Рисунок 4. 13 – Тест 5, вхідні дані

Вихідні дані:

1. Після натискання кнопки додати виведено опис фільму, див. Рисунок 4. 14.

Кіно	12+	Драма	США	2000	Режисер
Кіно1	12+	Драма	Франція	2002	Режисер1

Рисунок 4. 14 – Тест 5, вихідні дані

Результат тесту: тест виконано коректно

Тест 6.

Очікуваний результат: інформаційне повідомлення про необхідність заповнення всіх полів

Вхідні дані:

1. Заповнено всі текстові поля окрім одного, див. Рисунок 4. 15.

Назва	<input type="text" value="Кіно1"/>	Країна	<input type="text" value="Франція"/>
Вік	<input type="text" value="12+"/>	Рік випуску	<input type="text" value="2002"/>
Жанр	<input type="text"/>	Режисер	<input type="text" value="Режисер1"/>

Рисунок 4. 15 – Тест 6, вхідні дані

Вихідні дані:

1. Після натискання кнопки додати, на екран виведено інформаційне повідомлення, див. Рисунок 4. 16.

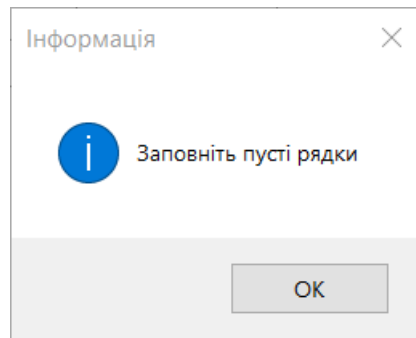


Рисунок 4. 16 – Тест 6, вихідні дані

Результат тесту:

Тест 7.

Очікуваний результат: фільм буде видалено

Вхідні дані:

1. Заповнено випадаючий список, див. Рисунок 4. 17.

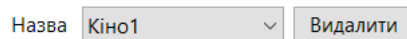


Рисунок 4. 17 – Тест 7, вхідні дані

Вихідні дані:

1. Інформаційне повідомлення з попередженням про видалення, та можливістю змінити вибір, див Рисунок 4. 18.

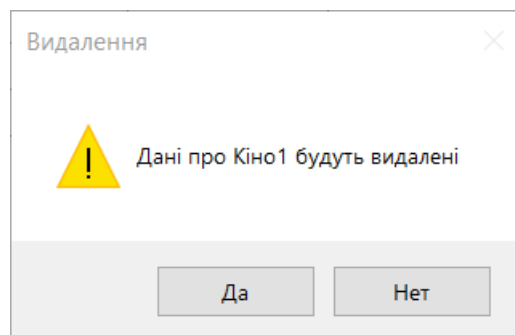


Рисунок 4. 18 – Тест 7, вихідні дані

2. Поле для виводу очищено, див Рисунок 4. 19.

Кіно	12+	Драма	США	2000	Режисер

Рисунок 4. 19 – Тест 7, очищення поля виводу

Результат тесту: тест виконано коректно.

Висновки до розділу

Було протестовано роботу програми методом чорної скриньки на можливі помилки.

Загальні висновки

Результатом роботи є програмний продукт, розроблений на мові С# з використання інструменту WindowsForms, який виконує функцію автоматизації продажу квитків в кінотеатрі.

На основі розглянутих та описаних аналогів, з оцінкою їх плюсів та мінусів, були поставлені задачі проектування та розробки проекту.

На етапі проектування були визначені вхідні та вихідні умови, функціональні вимоги, спроектовані екрани програми з їх інтерфейсом, було спроектовано та розроблено код програми.

Програмний продукт було протестовано методом чорної скриньки, що дало можливість оцінити функціонал програми.

Список використаних джерел

1. [Електронний ресурс] Windows Forms documentation -
<https://learn.microsoft.com/en-us/dotnet/desktop/winforms/?view=netdesktop-7.0>
2. [Електронний ресурс] Руководство по программированию в Windows Forms -
<https://metanit.com/sharp/windowsforms>

Додатки

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАТВЕРДЖУЮ

Проректор Українського державного
університету науки і технологій

Анатолій РАДКЕВИЧ

18.02.22

РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОДАЖУ КВИТКІВ В КІНОТЕАТРИ

Технічне завдання

ЛИСТ ЗАТВЕРДЖЕННЯ

1116130.01301-01-ЛЗ

Представники

підприємства-розробника

Завідувач кафедри КІТ

Вадим ГОРЯЧКІН

18.02.22

Керівник розробки

Олександр ІВАНОВ

18.02.22

Виконавець

Артем ЮРТЕЕВ

18.02.22

Норм-контролер

Світлана ВОЛКОВА

18.02.22

2023

ЗАТВЕРДЖЕНО

1116130.01301-01-ЛЗ

РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОДАЖУ КВИТКІВ В КІНОТЕАТРІ

Технічне завдання

1116130.01301-01-ЛЗ

Листів 16

2023

1116130.01301-01
ЗМІСТ

Додатки.....	1
1 ВВЕДЕННЯ	4
2 ПІДСТАВИ ДЛЯ РОЗРОБКИ.....	5
3 ПРИЗНАЧЕННЯ РОЗРОБКИ	6
4 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ	7
4.1 Вимоги до функціональних характеристик	7
4.2 Вимоги експлуатації	7
4.3 Вимоги до складу та параметрів технічних засобів	9
4.4 Вимоги до інформаційної та програмної сумісності	9
4.5 Вимоги до маркування і упаковки	9
4.6 Вимоги до транспортування та зберігання.....	10
5 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ.....	11
6 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ	12
7 ПОРЯДОК І КОНТРОЛЬ ПРИЙМАННЯ	13
8 БІБЛЮГРАФІЧНИЙ СПИСОК.....	14

1116130.01301-01

1 ВВЕДЕННЯ

«Розробка автоматизованої системи продажу квитків в кінотеатрі» - це самостійно розроблена програма, призначенням якої є автоматизація роботи працівників кінотеатру. А саме автоматизація продажу квитків. Програма призначена допомагати у роботі працівникам кінотеатру. Користувач програми може додавати до неї кінофільми, створювати їх короткий опис і на основі доданих фільмів створювати розклад сеансів, додавати вартість квитків та продавати їх.

Для продажу квитків в програмі створена схема кінозалу. Вона показує зайняті та вільні місця на певний сеанс фільму, що допомагає відвідувачам легше обирати собі місце для перегляду фільму, а також бачити вартість обраних ними місць. Реалізована і можливість повернення квитків на сеанс.

1116130.01301-01

2 ПІДСТАВИ ДЛЯ РОЗРОБКИ

Підставою для розробки є наказ від 08.12.21 №77ст ректора Українського державного університету науки і технологій “Про призначення наукових керівників та затвердження тем бакалаврських робіт” за спеціальністю 121 “Інженерія програмного забезпечення» факультету “Комп’ютерних технологій і систем” по кафедрі “Комп’ютерні інформаційні технології”.

Тема дипломної роботи - “РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОДАЖУ КВИТКІВ В КІНОТЕАТРІ”. Керівник - доцент Іванов О. П.

1116130.01301-01

3 ПРИЗНАЧЕННЯ РОЗРОБКИ

Функціональне призначення – програмний продукт має додавати, зберігати та видаляти дані щодо доданих фільмів, сеансів, їх розкладу та білетів. Відтворювати розклад, схему кінозалу, опис фільмів, вартість квитків.

Експлуатаційне призначення – за допомогою програмного продукту відбувається створення розкладу сеансів у кінотеатрі та продаж квитків до них, що автоматизує роботу працівників кінотеатру, покращує якість обслуговування відвідувачів.

1116130.01301-01

4 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

4.1 Вимоги до функціональних характеристик

Програмний продукт повинен:

- забезпечувати можливість додавання фільмів та їх описів до програми;
- забезпечувати можливість видалення фільмів та їх описів до програми;
- забезпечувати можливість створення розкладу сеансів на показ кінофільмів і встановлення їх вартості;
- забезпечувати можливість видалення розкладу сеансів на показ кінофільмів;
- забезпечувати можливість вибору фільму з розкладу та виводити опис цього фільму екран, відображати на схемі кінозалу вільні та зайняті місця, а також їх вартість;
- забезпечувати можливість повернення придбаних відвідувачами квитків;
- забезпечувати збереження введених даних;

4.2 Вимоги експлуатації

Програмний продукт повинен використовуватись у приміщеннях які відповідають умовам роботи ЕОМ, а саме мають такі кліматичні, санітарні та гігієнічні умови, які відповідають ДНАОП 0.00-1.13-99 (див. Рисунок 1).

Пора року	Категорія робіт згідно з ГОСТ 12.01-005-88	Температура повітря, град.С	Відносна вологість повітря, %	Швидкість руху повітря, м/с
		Оптимальна	Оптимальна	Оптимальна
Холодна	легка-1-а	22-24	40-60	0,1
	легка-1-б	21-23	40-60	0,1
Тепла	легка-1-а	23-25	40-60	0,1
	легка-1-б	22-24	40-60	0,2

Рисунок 1

1116130.01301-01

Працювати з програмою може людина, що має навички роботи з мобільними пристроями та ознайомена з керівництвом користувача програмного продукту.

1116130.01301-01

4.3 Вимоги до складу та параметрів технічних засобів

Продукт, що розробляється повинен використовуватись на ПК що мають наступні характеристики:

- 64-разрядные процесор и операційна система;
- ОС: Windows XP or later;
- Процесор: 1.5 GHz Core2Duo;
- Оперативна пам'ять: 2 GB ОЗУ;
- Відеокарта: OpenGL 1.4 or better;
- Місце на жорсткому диску: 100 MB;
- Додатково: Minimum 1280x720 screen resolution.

4.4 Вимоги до інформаційної та програмної сумісності

Програмний продукт розробляється для операційної системи Windows XP, або пізніших версій

4.5 Вимоги до маркування і упаковки

Упаковка програмного продукту, включаючи документацію повинна бути захищена від пошкоджень різного роду (механічних, кліматичних).

На упаковці повинно бути вказана назва продукту, номер версії, мінімальні системні вимоги.

На зворотній стороні упаковки вказується розробник та його юридична адреса.

<p style="text-align: center;">Програмний продукт</p> <p style="text-align: center;">РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОДАЖУ КВИТКІВ В КІНОТЕАТРИ</p> <p style="text-align: center;">Мінімальні системні вимоги</p> <ul style="list-style-type: none"> • 64-разрядные процесор и операційна система • ОС: Windows XP or later • Процесор: 1.5 GHz Core2Duo • Оперативна пам'ять: 2 GB ОЗУ • Відеокарта: OpenGL 1.4 or better • Місце на жорсткому диску: 100 MB • Додатково: Minimum 1280x720 screen resolution 	<p style="text-align: center;">Програмний продукт</p> <p style="text-align: center;">РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ПРОДАЖУ КВИТКІВ В КІНОТЕАТРИ</p> <p style="text-align: center;">Розробник: Юргеев А.В. Кафедра "КІТ", УДУНТ м.Дніпро, вул.Лазаряна 2 2022</p>
---	--

1116130.01301-01

4.6 Вимоги до транспортування та зберігання

Транспортування повинне забезпечувати збереження програмного продукту його цілісність і запобігання несанкціонованого доступу до нього. Програмний виріб міститься на фізичному носії та переданий через microUSB порт.

1116130.01301-01

5 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

До складу документації мають входити:

- специфікація;
- текст програми;
- опис програми;
- керівництво користувача.

Вся документація програмного додатку повинна задовольняти вимоги до програмної документації.

1116130.01301-01
6 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ

Стадії та етапи розробки, див. Таблиця 1.

Стадія	Зміст	Строки виконання
Технічне завдання	Постановка задачі, збір інформації, виріб та обґрунтування критеріїв розробки. Попередній вибір методів рішення задач. Визначення вимог до технічних засобів. Узгодження і затвердження технічного завдання.	31.01.23 - 18.02.23
Робочий проект	Програмування та відлагодження програми.	19.02.23 - 20.05.23
	Тестування програми	20.05.23 - 27.05.23
	Розробка, узгодження і затвердження програмної документації.	27.05.23 - 12.06.23
	Подання кваліфікаційної роботи до кафедри	07.06.23
	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	26.06.23

Таблиця 1

1116130.01301-01

7 ПОРЯДОК І КОНТРОЛЬ ПРИЙМАННЯ

Контроль за виконанням роботи здійснює керівник розробки доц. Іванов О.

П.

Прийом здійснюється комісією у складі:

- Горячкін В. М. (керівник підрозділу);
- Іванов О. П. (керівник розробки).

1116130.01301-01

8 БІБЛІОГРАФІЧНИЙ СПИСОК

- 1 І62 Інженерія програмного забезпечення [Текст]: навчальний посібник / В. М. Горячкін, О. В. Горбова, О. С. Куроп'ятник; Український державний університет науки і технологій. – Дніп-ро, 2022. – 140 с.

Program.cs

Form1.cs


```
"Ряд " + seat_row, "Місце " + seat_number);
```

```
"Ряд " + seat_row, "Місце " + seat_number, ticket_value + " грн");
```

```
Administration_button_Click(object sender, EventArgs e)
```

```
"Назва: " + film[i].name);  
"Рік виходу: " + film[i].year_of_issue);  
"Віковий рейтинг: " + film[i].MPA_rating_system);  
"Жанр: " + film[i].genre);  
"Режисер: " + film[i].director);  
"Країна: " + film[i].country);
```

```
"Спочатку оберіть місце в кінозалі",  
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);
```



```
"Спочатку заповніть випадаючі списки",  
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);
```


грн",

```
"Заповніть пусті рядки",  
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

```
"Фільм з такою назвою існує",  
"Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);
```

```
"Спочатку заповніть випадаючі списки",  
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

```
"Дані про " + film_delete_comboBox.SelectedItem.ToString()
```

```
" будуть видалені", "Видалення", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
```

```
" грн",
```

```
"Такого фільма не знайдено",  
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

" грн",

"Заповніть пусті рядки",
"Інформація", MessageBoxButtons.OK, MessageBoxIcon.Information);

"На цю годинну сеанс вже існує",
"Помилка", MessageBoxButtons.OK, MessageBoxIcon.Error);

```
"Сеанс о" + delete_session_time_comboBox.SelectedItem.ToString()  
" буде видалений", "Видалення", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
```

```
" грн",
```

Session.cs

Ticket.cs

