

Державний торговельно-економічний університет
Кафедра інженерії програмного забезпечення та кібербезпеки

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЄКТ

на тему:

«Програмний додаток реалізації цифрового медіа контенту»

Студента 4 курсу, 6 групи,
спеціальності 121 «Інженерія
програмного забезпечення»
освітньої програми «Інженерія
програмного забезпечення»

Гармати Богдана
Юрійовича

підпис студента

Науковий керівник
кандидат технічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Власенко Лідія
Олександрівна

підпис керівника

Гарант освітньої програми
кандидат технічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Рзаєва Світлана
Леонідівна

підпис гаранта

КИЇВ – 2023

Державний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь бакалавр

Спеціальність 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії програмного
забезпечення та кібербезпеки

Криворучко О. В.

«14» листопада 2022 р.

Завдання

на випускний кваліфікаційний проєкт студентів

Гармати Богдана Юрійовича

(прізвище, ім'я, по батькові)

1. Тема випускного кваліфікаційного проєкту «Програмний додаток
реалізації цифрового медіа контенту»

Затверджена наказом ректора від «6» грудня 2022 р. № 3288

2. Строк здачі студентом закінченого проєкту 5 червня 2023

3. Цільова установка та вихідні дані до проєкту

Мета проєкту розробка та реалізація програмного додатку, який
забезпечує функціональність цифрового медіа контенту.

Об'єкт дослідження процес розробки та реалізації програмного додатку
реалізації цифрового медіа контенту.

Предмет дослідження розробка програмного додатку реалізації цифрового
медіа контенту.

4. Консультанти проєкту із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускного кваліфікаційного проєкту (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ВИКОРИСТАННЯ ДОДАТКУ РЕАЛІЗАЦІЇ МЕДІА КОНТЕНТУ

1.1. Аналіз проблеми

1.2. Безпека цифрового медіа контенту

1.3. Технічне завдання

1.4. Висновок до розділу 1

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ

2.1. Вибір інструментів та шаблону проектування

2.2. Способи інтеграції з платіжними системами

2.3. Архітектура додатку

2.4. Архітектура бази даних

2.5. Висновок до розділу 2

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО ДОДАТКУ

3.1. Інтерфейс користувача

3.2. Серверна частина

3.3 Висновок до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

6. Календарний план виконання проєкту

№ пор.	Назва етапів випускного кваліфікаційного проєкту	Строк виконання етапів проєкту	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускного кваліфікаційного проєкту</i>	21.09.2022	
2.	<i>Розробка та затвердження завдання на проєкт</i>	14.11.2022	
3.	<i>Вступ та перелік літературних джерел</i>	23.12.2022	
4.	<i>Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ВИКОРИСТАННЯ ДОДАТКУ РЕАЛІЗАЦІЇ МЕДІА КОНТЕНТУ</i>	27.01.2023	
5.	<i>Розділ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ</i>	03.03.2023	
6.	<i>Розділ 3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО ДОДАТКУ</i>	14.04.2023	
7.	<i>Висновки</i>	28.04.2023	
8.	<i>Здача випускного кваліфікаційного проєкту на кафедрі (перша перевірка)</i>	17.05.2023	
9.	<i>Підготовка автореферату та презентації доповіді</i>	26.05.2023	
10.	<i>Попередній захист випускного кваліфікаційного проєкту</i>	29.05.2023 – 02.06.2023	
11.	<i>Зовнішнє рецензування випускного кваліфікаційного проєкту</i>	05.06.2023	05.06.2023
12.	<i>Здача прошитого випускного кваліфікаційного проєкту на кафедрі</i>	05.06.2023	05.06.2023
13.	<i>Публічний захист випускного кваліфікаційного проєкту</i>		

7. Дата видачі завдання «14» листопада 2022 р.

8. Науковий керівник випускного кваліфікаційного проєкту _____

Власенко Л. О

(прізвище, ініціали, підпис)

9. Гарант освітньої програми _____

Рзаєва С.Л.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент _____

Гармата Б. Ю.

(прізвище, ініціали, підпис)

Відповідно до мети дослідження робота присвячена розробці та реалізації програмного додатку, який забезпечує реалізацію цифрового медіа контенту.

В результаті порівняльного аналізу аналогічних рішень визначено, що схожі додатки відсутні. Або знаходяться в вигляді сайтів

Розробка серверної частини виконана у середовищі Microsoft SQL Server

Готовий програмний додаток було успішно протестовано

Ключові слова: Об'єктно орієнтоване програмування, MVVM, C#, медіа контент, UML

ABSTRACT

According to the purpose of the research, the work is devoted to the development and implementation of a software application that provides the implementation of digital media content.

As a result of the comparative analysis of similar solutions, it was determined that there are no similar applications. Or are in the form of sites.

The development of the server part was carried out in the Microsoft SQL Server environment.

The finished software application has been successfully tested.

Keywords: Object-oriented programming, MVVM, C#, media content, UML

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

MVVM – Model-View-ViewModel

ООП – Об’єктно – орієнтоване програмування



					<i>ДТЕУ 121 06-08.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		14.04.23	«Програмний додаток реалізації цифрового медіа контенту»	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Власенко Л. О.		14.04.23		ПС	2	51
Гарант		Рзаєва С.Л.		14.04.23		Факультет інформаційних технологій 4 курс, 6 група		
Розробив		Гармата Ю. Ю.		14.04.23				
					<i>Перелік умовних скорочень</i>			

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ВИКОРИСТАННЯ ДОДАТКУ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ	7
1.1.Аналіз проблеми.....	7
1.2.Безпека цифрового медіа контенту	9
1.3.Технічне завдання	12
1.4. Висновки до розділу 1	16
РОЗДІЛ 2 ПРОЕКТУВАННЯ ТА РОЗРОБКА ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ	18
2.1. Вибір інструментів та шаблону проектування.....	18
2.2.Способи інтеграції з платіжними системами	20
2.3.Архітектура додатку	21
2.4.Архітектура бази даних	27
2.5. Висновки до розділу 2	29
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ДОДАТКУ	30
3.1. Інтерфейс користувача	30
3.2. Серверна частина	33
3.3. Висновок до розділу 3.....	35
ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....	37
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	38
ДОДАТКИ	40

					<i>ДТЕУ 121 06-08.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	«Програмний додаток реалізації цифрового медіа контенту»	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			23.12.22		3	3	50
Керівник	Власенко Л. О.			23.12.22		Факультет інформаційних технологій 4 курс, 6 група		
Гарант	Рзаєва С.Л.							
Розробив	Гармата Б. Ю.			23.12.22	<i>Зміст</i>			

КЕРІВНИЦТВО

КОРИСТУВАЧА.....

ДОДАТКИ.....

.....



							Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-08.БР		5

ВСТУП

Актуальність. На сьогоднішній день досить стрімко розвивається ІТ технології за допомогою яких розповсюджують різні типи послуг та товарів. Також розвиток ІТ технологій спровокував створення нових типів товарів які існують тільки в цифровому вигляді. Наприклад створення різних програмних додатків, зображень в інтернеті, відео уроків для навчання якоїсь професії, а також створення розважальних відео. Такі типи товарів почали називати цифровим медіа контентом і він включає в себе досить обширний перелік товарів таких як відео-, аудіо- контент, зображення та інші. Монетизація таких товарів є досить різною наприклад для додатків існують різні магазини в інтернеті також використовуються сайти, але за свої послуги вони вимагають сплачувати процент з продажів або оплатити деяку суму для додавання свого товару на цю площадку для продажів. Тому досить активно почали створюватись додатки по продажу для конкретної компанії або людини, що створює продукт та потребує в його реалізації

Мета дослідження: є розробка та реалізація програмного додатку, який забезпечує функціональність цифрового медіа.

Об'єкт дослідження: процес розробки та реалізації програмного додатку реалізації цифрового медіа контенту

Предмет дослідження: розробка програмного додатку реалізації цифрового медіа контенту

У відповідності з метою дослідження поставлені наступні завдання:

- Розробка функціонального програмного додатка;
- Розгляд способів захисту цифрового вмісту;
- Вивчення інтеграції з платіжними системами .;
- Розробка інтерфейсу користувача.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 06-08.БР			
Зав. каф.		Криворучко О.В.		23.12.22	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Керівник		Власенко Л. О.		23.12.22		В	5	51
Гарант		Рзаєва С.Л.		23.12.22		Факультет інформаційних технологій		
Розробив		Гармата Б. Ю.		23.12.22		4 курс, 6 група		
					Вступ			

Методи дослідження Аналіз існуючих програмних додатків, літературний огляд статей

Наукова новизна дослідження полягає в створенні нового програмного додатку для реалізації медіа контенту

Практичне значення дослідження: Розробка ефективного та зручного програмного додатка



					ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		5

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ВИКОРИСТАННЯ ДОДАТКУ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ

1.1. Аналіз проблеми

На сьогоднішня досить поширеним стало реалізовувати медіа контент через інтернет оскільки це швидше дешевша та надійніше ніж реалізовувати його через фізичні носії або спеціальні магазини.

Цифрове розповсюдження – це спосіб розповсюдження, за якого вміст доставляється без використання фізичних носіїв, зазвичай шляхом завантаження з Інтернету безпосередньо до споживача додому.

Окрім того початківці що створюють цифровий медіа контент в інтернеті можуть на різних площадках показувати свій контент, а також монетизувати свою роботу, але деякі площадки мають поріг входження.

Наприклад на YouTube для початку отримування грошей потрібно виконувати такі вимоги:

Стати учасником Партнерської програми YouTube можуть автори з доброю репутацією, які дотримуються всіх Правил монетизації контенту YouTube, проживають у країні або регіоні, де доступна Партнерська програма YouTube, і мають дійсний пов'язаний обліковий запис AdSense. Крім цього, авторам потрібно мати:

1000 підписок і 4000 годин перегляду загальнодоступних відео за останні 12 місяців або

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 06-08.БР</i>			
Зав. каф.		Криворучко О.В.		27.01.23	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Керівник		Власенко Л. О.		27.01.23		РІ	7	51
Гарант		Рзаєва С.Л.		27.01.23		Факультет інформаційних технологій 4 курс, 6 група		
Розробив		Гармата Б. Ю.		27.01.23				
					Аналіз предметної області використання додатку реалізації цифрового медіа контенту			

1000 підписок та 10 мільйонів годин перегляду загальнодоступних відео YouTube Shorts за останні 90 днів [1].

Але на деяких платформах навіть така можливість не доступна, як от наприклад на платформі Twitch оплата монетизація контенту автора не проводиться і користувачам доводиться розраховувати або на донати від глядачів, або показом реклами на своєму каналі.

Крім того для виведення грошей з цих площадок потрібно, щоб на рахунку аккаунту була якась конкретна сума для виводу.

Через такі вимоги багато хто вирішує створювати власні додатки з реалізацією медіа контенту для додаткової монетизації своєї діяльності.

Досить багато таких програмних рішень створюються у вигляді сайту або веб додатку але доступ до них можливий тільки з доступом до інтернету, що іноді досить незручно, хоча й досить дешево та для виробництва застосовується менше часу.

Окрім того у вигляді сайтів такі програмні рішення мають деякі обмеження оскільки можуть підходити не для всіх пристроїв або операційних систем та працювати некоректно. Це створює деякі незручності під час користування та отримання послуги про покупку продукту.

Також такі рішення мають досить погану продуктивність та зламатися від одночасного великого напливу людей. Також такий варіант повинен досить часто перевірятися на правильність виведеної інформації.

В свою чергу в десктоп додатку дані можна підв'язати через базу даних, що робить їх більш захищеними, а також через базу можна контролювати виведення коректної інформації. Через те, що додаток буде знаходитись на комп'ютері користувача йому не потрібно буде досить довго шукати його в браузері. Ще можливість підключити онлайн програвач для попереднього перегляду медіа контенту.

								ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата					8

Але в свою чергу такі десктоп рішення є досить дорогим в виробництві, оскільки над їх створенням проводиться більше часу та зусиль, а також можуть не підходити під деякі гаджети та мати технічні проблеми при довгому не оновлені програми такі як відсутність нового медіа контенту, відсутність підтримки онлайн програвача в додатку і на самий кінець неможливість запуснути додаток

Через такі нюанси було вирішено, що це буде саме десктоп додаток оскільки надання інформації через нього є більш надійним.

1.2. Безпека цифрового медіа контенту

У наш час досить поширеною проблемою є використання медіа контенту без відома автора цього контенту що частіше за все відбувається в мережі інтернет оскільки там не зрозуміло як захистити свій контент.

Будь-яка розумова діяльність повинна бути захищена від можливості шахраїв привласнити її собі і на допомогу цьому йде авторське право.

Авторське право — галузь цивільного права та права інтелектуальної власності, що регулює відносини, які виникають у зв'язку зі створенням і використанням творів науки, літератури і мистецтва, захищає особисті (немайнові) і майнові права авторів та їхніх правонаступників[2].

Контролюється воно законом країни в якому створений продукт розумової діяльності. У цей список продуктів входять багато предметів таких як: книги, фільми, серіали, мультиплікація, відеоігри, музика, програмні додатки, збірки оповідань, віршів та багато інших видів.

Цифровий медіа контент також входить до цього списку, що підпадає під захист авторських прав.

У нашій країні поки нема закону про цифровий контент тому законодавством цифровий контент не захищений і його захист гарантує

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			9

тільки авторське право. Але все ж є деякі методики для захисту від неправомірного розповсюдження цифрового медіа контенту.

Одним із видів захисту медіа контенту є нанесення водяних знаків на медіа контент.

Водяний знак, або Філігрань (італ. / filigrana, від лат. filum: нитка і granum: зерно), — прозоре зображення, відтворюване (насамперед на особливо важливих документах) на папері під час його виробництва [3]. В цифровому варіанті це напівпрозорий текст що неможливо прибрати не пошкодивши оригінальний цифровий контент.

Іншим прикладом захисту може бути показання повідомлення про авторське право.

Повідомлення про авторське право - це заява, яка вказує на те, що певний твір захищений законами про авторське право. Зазвичай воно містить символ авторського права (©), рік публікації та ім'я власника авторських прав [4].

Такий варіант захисту також є досить дієвим оскільки повідомляє звичайного користувача, що вирішив подивитись контент, що він має автора і його використання без автора є незаконним.

Можна запросити отримання значка DMCA(Digital Millenium Copyright Act) – закон про авторське право в цифрову епоху, що буде показувати, що на сайті або в додатку знаходиться тільки авторський контент. Такий вид захисту допоможе передбачити копіювання сайту, додатку або контенту на ньому стороннім особам оскільки користувачі по цьому значку будуть бачити, що контент розміщений саме автором. Приклад значка DMCA показано на Рис 1.1.

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			10



Рис 1.1. Значок DMCA

Але DMCA використовується виключно для сайтів, а не для десктоп додатків, але є доступним по ціні.

Умови використання сайту – це документ, завданням якого є юридично-правильне встановлення правил для користувачів та зменшення рівня відповідальності адміністрації сайту (правовласника) [5].

Умови використання також є досить поширеним методом захисту оскільки допомагає користувачам зрозуміти як та де можна використовувати цифровий медіа контент користувачу. Вони також є досить дієвими тому що по ним легше доказати, що авторський контент був викрадений.

Ось кілька основних пунктів передбачених в документі «Умов використання»:

- a) визначити типи контенту, на які поширюється угода "Умови використання", наприклад, текст, зображення, відео тощо. (“6 способів запобігти копіюванню вашого контенту”)
- b) вказати дозволені способи використання контенту - що користувачі можуть робити з контентом, наприклад, переглядати, ділитися або завантажувати його для особистого використання. (“6 способів запобігти копіюванню вашого контенту”)

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			11

с) зазначити, що будь-яке несанкціоноване використання контенту, наприклад, копіювання, відтворення або розповсюдження без дозволу, суворо заборонено.

д) включити пункт про порушення та примусове виконання, який описує наслідки будь-якого несанкціонованого використання контенту, такі як припинення доступу, судовий позов тощо [4].

Ще одним методом захисту контенту це захистити паролем весь медіа контент, щоб зловмисники, або небажані особи не мали доступу до авторського медіа контенту в додатку або на сайті. Це також допоможе надавати доступ тільки перевіреним особам. Але проблемою такого способу є те, що при зламу паролю доступ до контенту отримують зловмисники та небажані користувачі, а можливо навіть і втратити доступ до контенту.

Це тільки основні види захисту медіа контенту, але ще є досить багато сервісів які організують захист контенту і відповідають за те, щоб контент не потрапив до зловмисників та небажаних користувачів. Із вище зазначеного видно, що обов'язковими є повідомлення про авторське право, водяний знак та пароль оскільки є загальнодоступними та дуже добре захищають медіа контент від зловмисників та недобросовісних користувачів.

1.3. Технічне завдання

1. Загальні відомості : Програмний додаток створюється для реалізації цифрового медіа контенту

1.1 Найменування системи: «AndroShop»

1.2 Планові терміни початку та закінчення роботи: 12.03.2023 – 30.03.2023

							ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата				12

1.3 Головні бенефіціари та потенційні користувачі програми: Програма розрахована на користувачів, що в основному переглядають відео ігровий контент.

2. Мета та призначення створення системи

2.1 Призначення системи: Система створюється для реалізації медіа контенту

2.2 Мета створення системи: Надання доступності медіа контенту та його реалізації шляхом розміщення його у додатку.

3. Вимоги до системи

3.1 Вимоги до системи в цілому

3.1.1 Вимоги до структури та функціонування системи перелік підсистем

3.1.1.1 Вимоги до способів і засобів інформаційного обміну між компонентами системи: Обмін влаштований таким чином, що інформація пов'язана з медіа контентом знаходиться на сервері в базі даних з якої додаток бере цю інформацію як і інформація пов'язана з користувачами.

3.1.1.2 Вимоги до режимів функціональності системи: Додаток повинен мати такі режими функціонування:

- Режим для користувачів, в якому надається доступ до перегляду загальної інформації та можливості придбання цифрового медіа контенту.

3.1.1.3 Вимоги до режимів управління системою: Програма матиме такі режими управління:

- Управління для користувача, що передбачає перегляд загальної інформації про товар, можливість придбати та залишити відгук до продукту

3.1.2 Показники призначення

								ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата					13

3.1.2.1 Параметри, що характеризують ступінь відповідності системи:

3.1.2.2 Вимоги до пристосованості системи до змін: програма повинна мати можливість змінюватись під зміни якщо вони необхідні

3.1.3 Вимоги до ергономіки та технічної естетики

3.1.4 Вимоги до експлуатації технічного обслуговування, ремонту і зберігання компонентів системи:

- Експлуатація користувачем передбачена переглядом загальної інформації про контент, можливість придбати продукт та залишити відгук на продукт
- Технічне обслуговування повинен проводити адміністратор через базу даних та додаток
- Ремонт повинен проводити розробник для більш якісного та швидкого відновлення працездатності

Компоненти системи зберігаються в папці проекту та на базі даних проекту

3.1.5 Вимоги до захисту інформації від не санкціонованого доступу: доступ до бази даних повинен захищатися логіном та паролем, файл з кодом програми повинен бути недоступним до редагування звичайному користувачу

3.1.5.1 Вимоги до інформаційної безпеки вся інформація про користувача не зберігається і використовується тільки для передачі продукту

3.1.5.2 Розмежування відповідальності ролей при доступі: Користування програмою розмежується на такі ролі:

- Користувач – доступно тільки перегляд продуктів придбання та можливість залишити відгук

								Аркуш
								14
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-08.БР			

- Адміністратор – доступно зміни в базі даних як додавання зміни та видалення даних а також відповідь на питання користувачів
- Розробник – перевірка працездатності, створення оновлень та їх введення в програму, а також відновлення працездатності програмного продукту

3.1.6 Вимоги до безпеки:

- Оплата продукту повинна проводитись через зовнішню платіжну систему PayPal
- Дані користувача не повинні зберігатися і використовуються тільки для передачі продукту
- Доступ до бази даних повинен мати тільки адміністратор
- Логін та пароль до бази даних повинні складатися з мінімум

12 символів та включати:

- Всі букви повинні бути латинськими
- Повинен бути як мінімум один символ верхнього регістру
- Повинна бути мінімум одна цифра
- Повинен бути мінімум один спеціальний символ

3.2 Вимоги до видів забезпечення

3.2.1 Вимоги до інформаційного забезпечення

3.2.1.1 Вимоги до складу, структури і способів організації даних в системі:

3.2.1.2 Вимоги до інформаційного обміну між компонентами системи: Дані зберігаються в базі даних з якої програмний додаток бере та виводить їх в вигляд для користувача

3.2.1.3 Вимоги щодо застосування систем управління базами даних: Обов'язковою є використання СУБД для зберігання медіа контенту,

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			15

назви продукту, дату створення, ціну зображення для подачі користувачу

3.2.1.4 . Вимоги до захисту даних від руйнувань при аваріях і збоях в електроживленні системи: Дані повинні додатково зберігатися на головному комп'ютері, а також на каналі YouTube та на хмарному сховищі

4. Вимоги до програмного забезпечення: Програмний додаток повинен мати зручний та зрозумілий інтерфейс

На початковій сторінці знаходиться перелік продуктів, що доступні.

Зверху повинна знаходитись панель з кнопками переходу до домашньої сторінки та корзини

Повинні підтримуватись дві мови Англійська та Українська мови

Оплата повинна проводитись через платіжну систему PayPal

5. Вимоги до технічного забезпечення: програма повинна бути створена на Visual Studio мовою програмування C# за допомогою шаблону проектування MVVM візуальна частина створена . База даних повинна бути створена за допомогою СУБД Microsoft SQL Server

1.4. Висновки до розділу 1

Виходячи з вище переліченого досить поширеною стала практика створення власного програмного рішення для реалізації авторського медіа контенту для отримання прибутку та збільшення аудиторії. Сервіси, що надають доступ розміщати авторський контент, або надають мале грошове заохочення або взагалі його не мають через, що авторам доводиться шукати інші методи. Через цю проблему і створюється даний програмний продукт

Але при створенні такого продукту виникає інша проблема як захистити медіа контент від зловмисників та недобросовісних користувачів. І з оглядом

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			16

на вище зазначене було вибрано як види захисту водяний знак, пароль та повідомлення про авторське право, а також бажаним буде створити умови користування.



								Аркуш
								ДТЕУ 121 06-08.БР
Зм.	Аркуш	№ докум	Підпис	Дата				17

РОЗДІЛ 2

ПРОЕКТУВАННЯ ТА РОЗРОБКА ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ ЦИФРОВОГО МЕДІА КОНТЕНТУ

2.1. Вибір інструментів та шаблону проектування

На сьогодні жодний програмне забезпечення не обходиться без проектування та створення діаграм через які заказники можуть побачити, що та як буде працювати в програмі. Але також важливим є вибір інструментів та мови програмування для створення цього додатку.

Є досить багато мов програмування для створення програмних продуктів для персональних комп'ютерів. Як видно з рис.2.1 найпопулярнішою мовою програмування на 2023 р. є JavaScript другою є Java третьою є Python на четвертому місці C# і на п'ятому TypeScript.



Рис 2.1 найпопулярніші мови програмування

Вони всі є досить хорошими зі своїми фішками і способами створення різних програмних рішень. В свою чергу для створення програмного продукту реалізації цифрового медіа контенту потрібна мова програмування з відповідними критеріями:

- наявність можливості розробляти програмний продукт за допомогою парадигми програмування ООП (Об'єктно – орієнтоване програмування) і як похідне розробка шаблоном програмування MVVM

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 06-08.БР			
Зав. каф.		Криворучко О.В.		03.03.23	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Керівник		Власенко Л. О.		03.03.23		P2	18	51
Гарант		Рзаєва С.Л.		03.03.23		Факультет інформаційних технологій 4 курс, 6 група		
Розробив		Гармата Б. Ю.		03.03.23				
					Проектування та розробка додатку для реалізації цифрового медіа контенту			

- b) можливість підключити і працювати з базами даних
- c) мова програмування для розробки десктоп додатку
- d) мова програмування для Full Stack розробки

За вище зазначеними критеріями з найпопулярніших мов програмування відпадають дві мови програмування це JavaScript та TypeScript оскільки це мови програмування для розробки сайтів та веб-додатків. Для вибору між мовами, що залишились потрібно подивитись ще одну статистику, а саме зображену на Рис. 2.2.

Мови програмування за сферами використання



Рис.2.2. Найпопулярніші мови програмування в Full Stack сегменті

Як видно з графіку то C# в сегменті Full Stack розробки лідирує з великим відривом Java стоїть на п'ятому місці, а Python навіть не входить в топ 5. З цього можна винести підсумки, що C# ідеально підходить для розробки даного програмного рішення. Окрім того до C# досить легко може бути підключено базу даних з СУБД Microsoft SQL Server, що має досить великі можливості для створення та обслуговування баз даних.

MVVM — шаблон проектування архітектури додатку. В його основі відділення коду призначеного для користувача інтерфейсу (UI) від решти, і створення слабкого зв'язку для синхронізації [6]. Вибір шаблону програмування MVVM було вибрано через те що він розділяє проект на три частини Model, View та View Model, що розділяють логіку додатку від інтерфейсу для користувача, що допоможе ще більше захистити дані від втручання користувачів та до втручання в роботу програми.

Основна логіка програми знаходиться в частині Model де знаходиться вся логіка програми, її обрахунки, формули та бізнес логіка з підключенням до бази даних.

Інтерфейс користувача різні графічні фішки та UI/UX дизайн знаходиться в частині View де розроблюється Front-end додатку, зовнішній вигляд та взаємодія для користувач.

Роль з'єднувача між цими двома частинами виконує View Model, що логіку додатку переводить в вигляд зрозумілий для користувача і виводить його в частину View.

Також важливою частиною розробки є середовище в якому вона розроблюється. Оскільки програмний продукт розроблюється мовою програмування C# то вибір середовища досить сильно зменшується. А також враховуючи, що додатку потрібне підключення баз даних то вибір середовища зупиняється на середовищі від Microsoft а саме Visual Studio, оскільки там крім того що він досить зручний для написання коду мовою програмування C# так в ньому також підтримується під'єднання баз даних до програмного продукту а також простий спосіб передачі даних між базою даних та додатком.

2.2.Способи інтеграції з платіжними системами

На сьогодні є досить багато компаній для реалізації різного вигляду контенту та послуг. З розвитком ІТ – технологій багато компаній переходять в інтернет для більш швидкого та зручного надання продуктів або послуг. Для такого виду діяльності потрібен метод щоб клієнт окрім швидкого доступу мав також можливість швидко оплатити товар для цього створені платіжні системи.

Платіжна система - система для виконання платіжних операцій із формальними та стандартизованими домовленостями і загальними

						ДТЕУ 121 06-08.БР	Аркуш
							20
Зм.	Аркуш	№ докум	Підпис	Дата			

правилами щодо процесингу, клірингу та/або виконання розрахунків між учасниками платіжної системи [7].

Платіжні системи є різних видів і підходять вони для різних задач по переказу грошей. Для бізнесу що знаходиться на сайті, або в додатку потрібен такий інструмент, як платіжна система, оскільки це допомагає клієнту швидко оплатити контент або послугу.

Для підключення платіжної системи використовують різні методи: хостингові платіжні шлюзи або інтегрування платіжної системи через API.

Використання хостингового платіжного шлюзу передбачає те що користувач переходить до платформи платіжного сервісу де він оплачує продукт або послугу.

Використання інтегрування платіжної системи через API передбачає що за допомогою розробників потрібні інструменти вбудовуються напряму на сайт або додаток.

Крім того, що вони мають плюси вони також мають мінуси. Для інтегрування платіжної системи через API потрібні розробники, що вбудують його в додаток, що робить цей метод досить затратний на гроші окрім того саме підприємство повинно забезпечити захист передачі особистих даних користувача. В цей же час хостинговий платіжний шлюз змушує користувача робити додаткові кроки для оплати продукту, або послуги.

Але для даного програмного продукту більш підходящим буде саме інтеграція платіжної системи через хостингові шлюзи оскільки це зменшує ціну за додаток та захист особистих даних користувача бере на себе платіжна система.

2.3. Архітектура додатку

Коли було обрано мову програмування парадигму розробки та шаблон час приходить для створення архітектури додатку, адже краще завчасно

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			21

передбачити більшість функціоналу, можливих проблем та варіантів розробки ніж в процесі стикатися з різними видами проблем.

Для розробки архітектури додатку використовують досить багато методів, але самим дієвим і популярним досі залишається створення UML діаграм. UML (Unified Modeling Language) — уніфікована мова моделювання, що використовується розробниками програмного забезпечення для візуалізації процесів та роботи систем [8]. Але UML це не мова програмування а набір правил для створення діаграм. Написання UML діаграм уніфікує їх та робить зрозумілими логіку додатку для тих хто не розбирається в програмуванні.

Є досить багато діаграм, але для різних додатків потрібні різні діаграми для пояснення структури та логіки додатку. Але все таки є декілька обов'язкових діаграм при створенні додатку. Це Use Case діаграма та діаграма класів бо вони описують як програма біде взаємодіяти з користувачем та те як програма буде працювати з середини між своїми компонентами.

Також досить корисною є написання діаграми алгоритму додатку для описання алгоритму роботи додатку та можливостей взаємодії. Також досить корисною є діаграма послідовностей для опису послідовності роботи операцій та підсистем програми а також приблизний час роботи операцій під час роботи програми.

Діаграма варіантів використання — (діаграма прецедентів, сценарій використання, use case) — дозволяє уявити типи ролей та їх взаємодію із системою. Проте не показує порядок виконання кроків. Зображує функціональні вимоги (те, що система може зробити) з точки зору користувача.[8]

Як зображено на рис.2.3 то користувач відкривши додаток переглядає продукти, після чого в нього є можливість придбати контент, або залишити

						ДТЕУ 121 06-08.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			22

коментар. При залишенні коментарю його переглядає адміністратор, якщо це запитання він відповідає в додатку.

При покупці контенту користувач вводить свою електронну пошту, а потім додаток переходить до платіжної системи через яку буде проводитись оплата. Після чого додаток отримує підтвердження оплати та перекидає відповідний контент на електронну пошту. Також при перегляді контенту користувач може переглянути трейлер відео контенту через вбудований програвач.

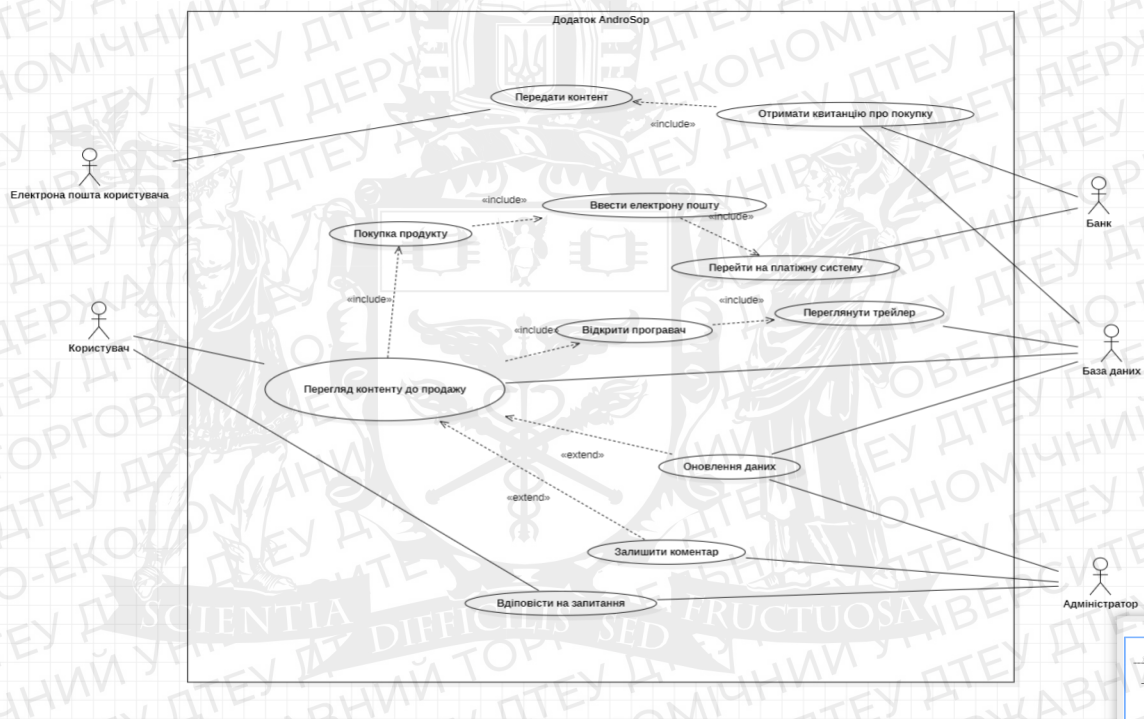


Рис.2.3. Use Case діаграма.

Клас (class) — категорія речей, що мають загальні атрибути та операції. Сама діаграма класів являє собою набір статичних, декларативних елементів моделі. (“Діаграми UML для моделювання процесів і архітектури проекту”) Вона дає нам найбільш повне і розгорнуте уявлення про зв'язки в програмному кодї, функціональність та інформацію про окремі класи [9]. Як зображено на Рис.2.4 діаграма класів представляє собою діаграму на якій зображено всі класи додатку. Всього є 4 класи «MediaContent», «Model», «ViewModel» та «View».

Клас «MediaContent» слугує для прийняття даних від бази даних про медіа контент, а саме його назву, короткий опис ціну контенту та відгуки. Клас «Model» слугує для того, щоб переводити дані з «MediaContent» в вигляд списку для легшого представлення великої кількості контенту в вигляді зрозумілому для користувача. Клас ViewModel створений як проміжний клас між баченням користувача в класу View та сирими даними з класу Model також додає команди для вибору окремого контенту покупки контенту та пошуку контенту. Клас View створений для представлення даних у прийнятному вигляді для користувача наслідуючи клас ViewModel.

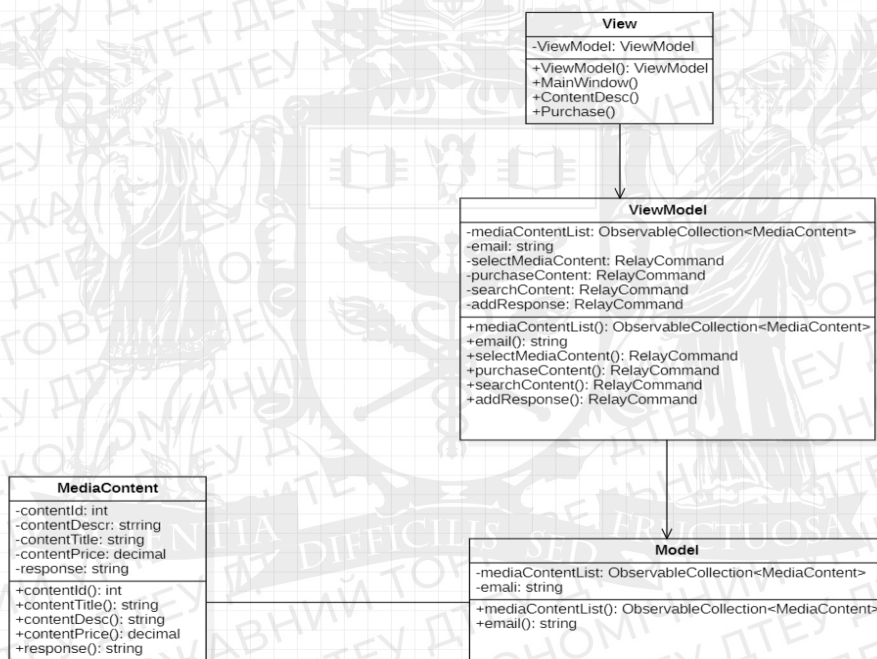


Рис.2.4. Діаграма класів

Діаграма алгоритму створюється для розуміння всіх варіантів дії додатку. Це допомагає більш зрозуміло представити логіку додатку для людей які не розбираються в програмуванні. Як видно з розробленого алгоритму на Рис.2.5. то спочатку відкривається головне вікно з списком контенту на продаж. Після цього користувач вибирає контент, що йому сподобався та якщо захоче придбати то його спочатку попросить ввести електрону пошту. Потім він переходить до платіжної системи де оплачує

товар. Після цього він на раніше введenu електрону пошту отримує посилання на файл, щоб скачати його. Після цього йому запропонують залишити відгук до контенту. В кінці його перекине на головний екран де якщо він хоче то продовжує покупки, а якщо ні закриває додаток.

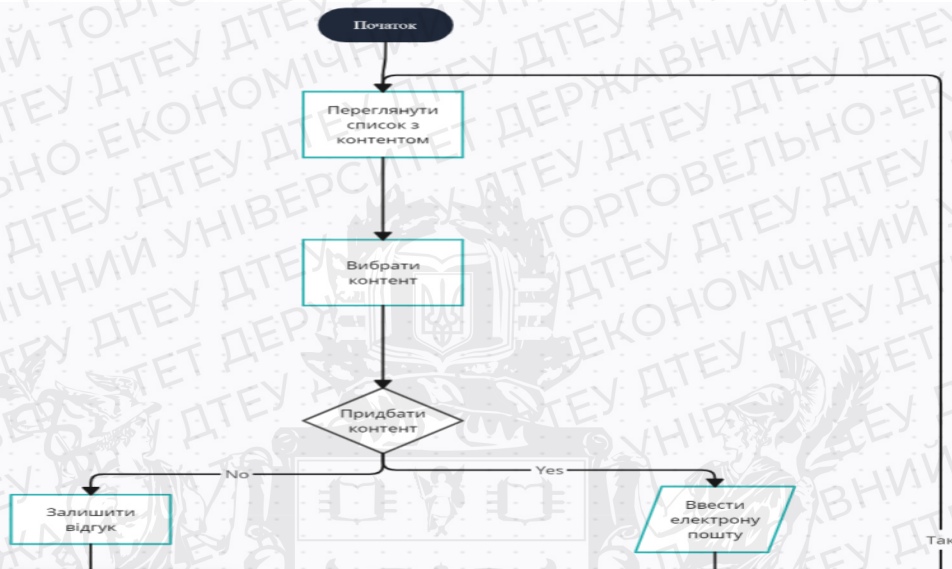


Рис. 2.5. Алгоритм аркуш 1

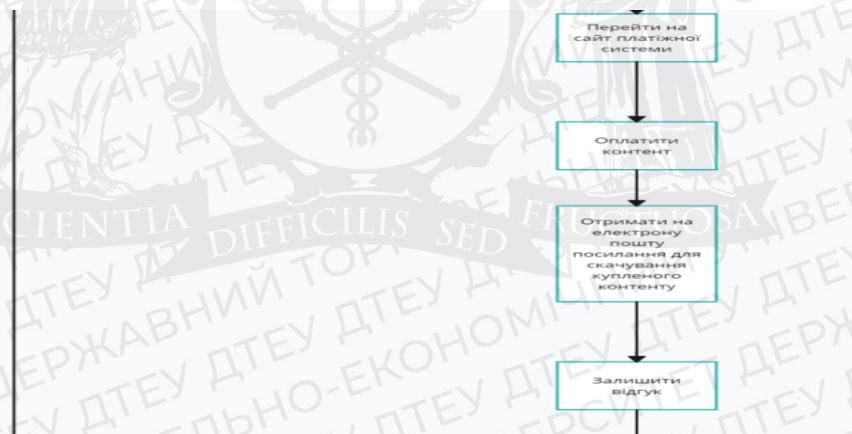


Рис.2.5. Алгоритм аркуш 2

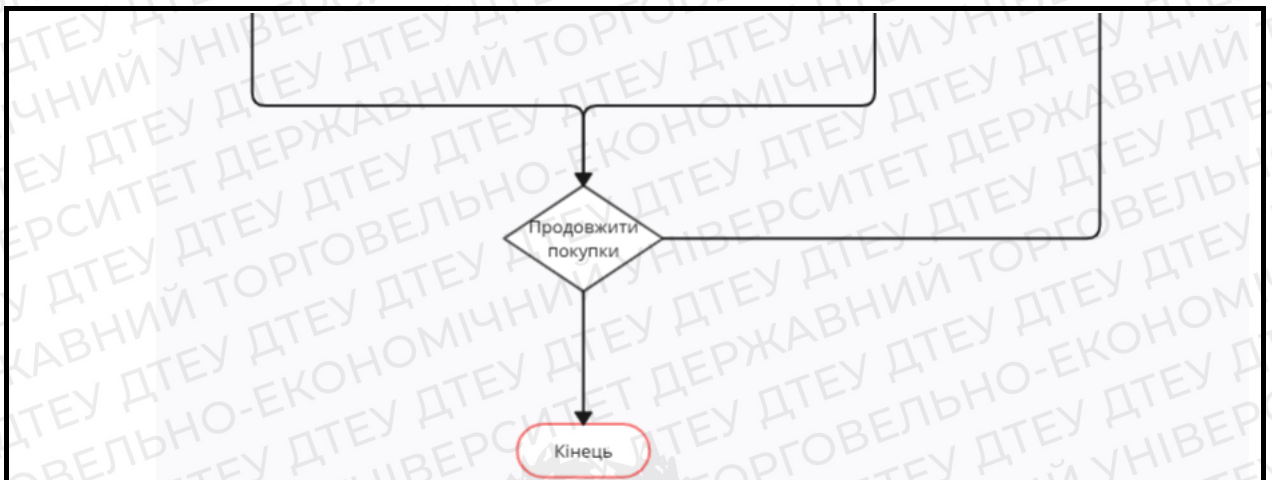


Рис.2.5. Алгоритм аркуш 3

"Діаграма послідовності (Sequence Diagram) — показує часові особливості передачі і прийому повідомлень об'єктами." ("Інструкція, як будувати UML-діаграми | DOU") Впорядкованість за часом слід розуміти як послідовність дій і не плутати з часовими діаграмами [8]. Це досить важлива діаграма оскільки показує в який період часу з'являється той чи інший об'єкт або екземпляр під час роботи додатку.

Як зображено на Рис.2.6 то видно що програма при запуску звертається до бази даних щодо списку контенту для програми. Після чого бере дані з відповідних класів потім користувач вибирає контент що сподобався та може залишити коментар який зберігається в базі та відображається для всіх користувачів. При спробі придбати товар додаток просить ввести електрону пошту, а потім переходить до платіжної системи де оплачує товар. Після того як додаток отримає квитанцію то відбувається запит на відправку посилання на електрону пошту користувача, що він вводив раніше для завантаження контенту.

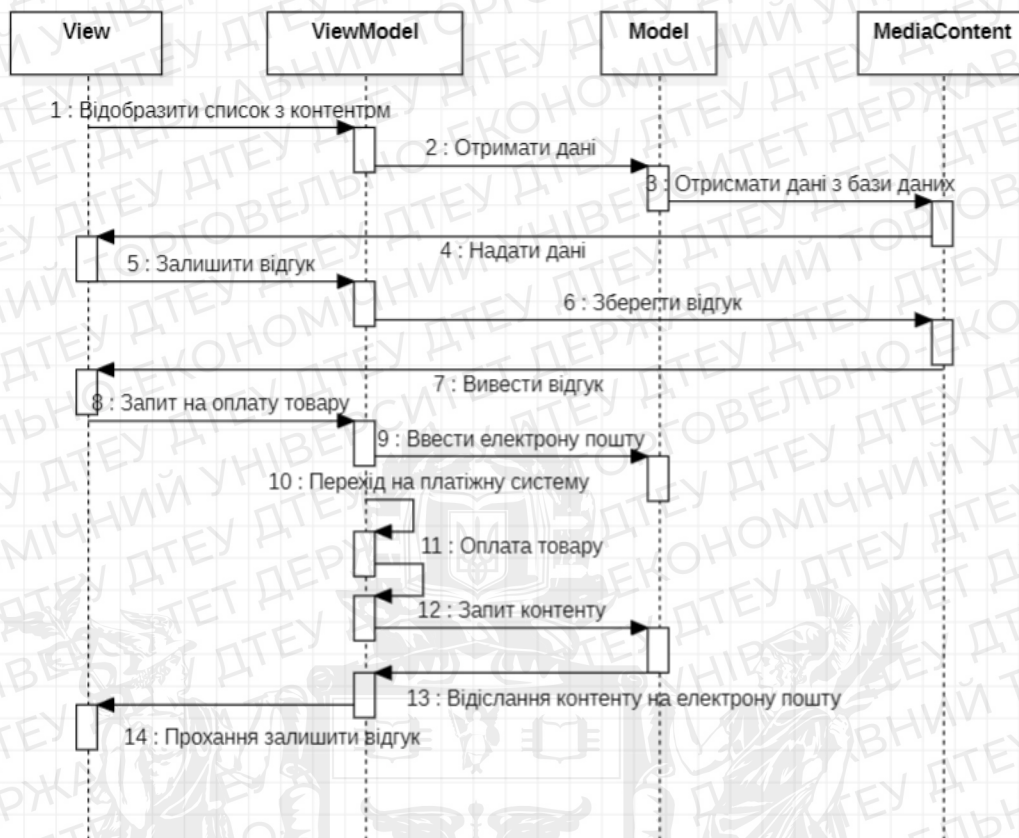


Рис.2.6. Діаграма послідовності

2.4.Архітектура бази даних

Але архітектуру потрібно розробляти не тільки для додатку, але й для бази даних, оскільки неправильно розроблена база даних може потім створити багато проблем з виведенням даних та може створити плутанину між таблицями. Для розробки архітектури бази даних як і для проектування архітектури додатку використовують діаграми, але коли для архітектури додатку використовувалися UML діаграми то для баз даних використовуються всього три діаграми а саме логічна модель, фізична модель та концептуальна модель баз даних.

Концептуальна модель – це модель для опису бази даних на базовому рівні без поглиблення в зв'язки чи атрибути. Як видно з Рис.2.7. то база даних має три таблиці User, Purchase та MediaContent. До таблиці User відноситься дані користувача, а саме електронна пошта. До таблиці MediaContent відноситься

основна інформація про контент таку як назву, опис, ціна. Таблиця Purchase об'єднує ці таблиці та вказує інформацію про придбаний товар

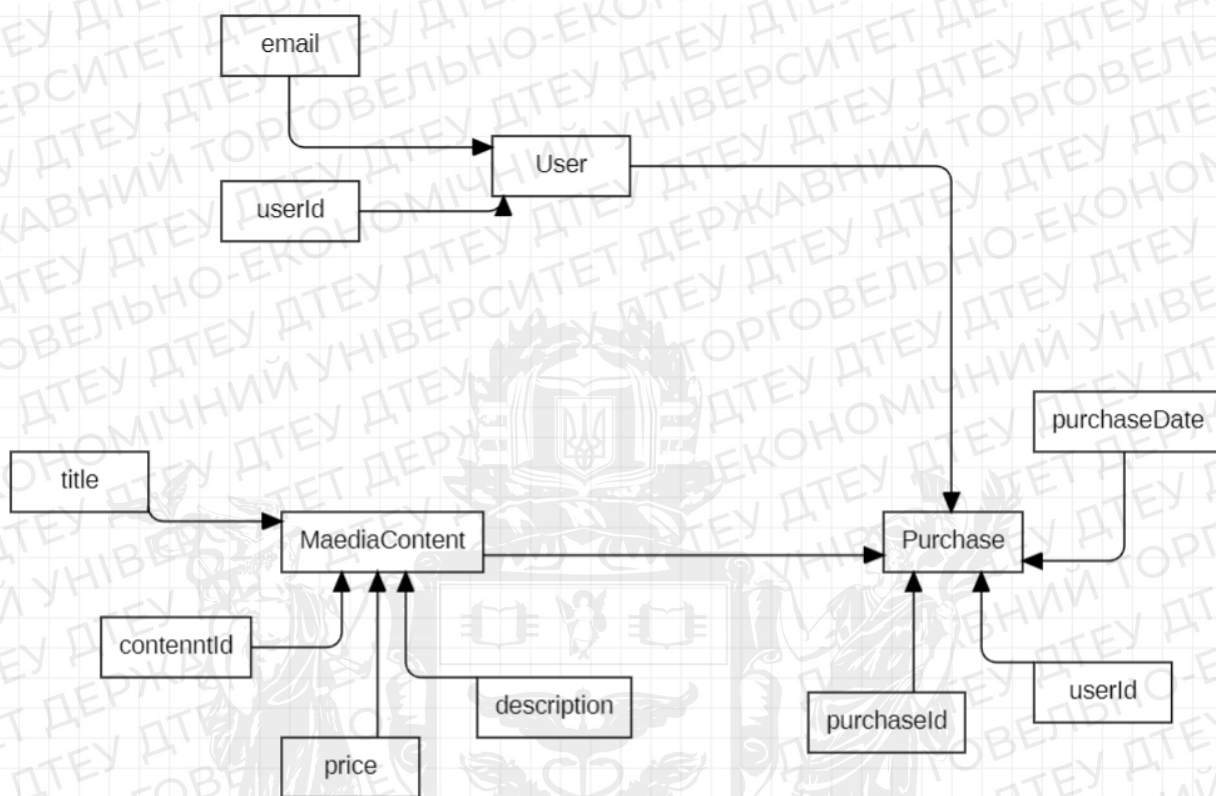


Рис.2.7. Концептуальна модель

Логічна модель є основою бази даних, вона повинна відображати взаємозв'язки між реляційними таблицями [10]. Під час створення бази даних логічна модель є найважливішою оскільки показує зв'язки між таблицями в середині бази. Як видно з Рис.2.8 то у базі даних знаходиться три таблиці. MediaContent, Users та Purchase. Таблиця MediaContent це таблиця в якій зберігається дані про контент його опис, назву в полі title, ціну в полі price, опис в полі description та посилання в полі link на контент та ключове поле contentId. Також є таблиця Users в котрій зберігається електрона пошта користувача в полі email та ключове поле userId. Ці дві таблиці об'єднує таблиця Purchase в котрій зберігаються дані про те який

користувач придбав який контент та дата покупки в полі purchaseDate та ключове слово purchaseId.

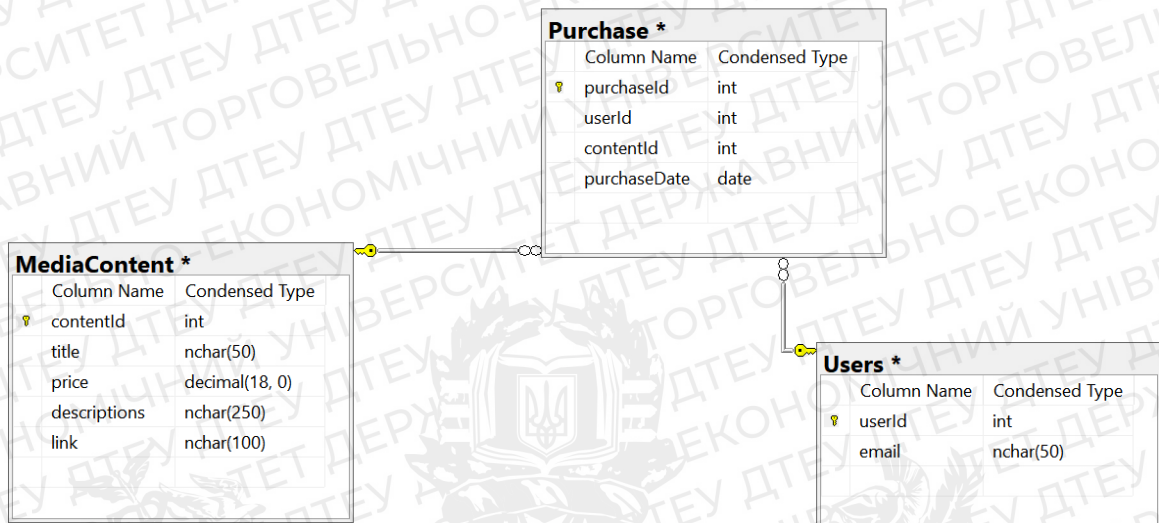


Рис.2.8. Логічна модель

2.5. Висновки до розділу 2

Отже, підсумовуючи вище викладене, можна визначити, що найбільш підходящим для розробки додатку виявилась мова програмування C# на платформі Visual Studio з розробкою бази даних та підключенням хостингового шлюзу платіжної системи. Також для правильного створення додатку було розроблено чотири UML діаграми для представлення логіки додатку в більш зрозумілому вигляді. Також було розроблено концептуальну та фізичну модель для створення бази даних та її представлення.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ПРОГРАМНОГО ДОДАТКУ

3.1. Інтерфейс користувача

При створенні будь-якого додатку важливим пунктом є розроблення інтерфейсу для користувача. Інтерфейс повинен бути зручним, інтуїтивним та зрозумілим для звичайного користувача.

На головній сторінці продукту знаходиться перелік контенту справа знаходиться поле для пошуку за назвою. Зверху знаходиться меню з кнопками Main для переходу на головне вікно. Саме вікно розміром 450 пікселів в довжину та 800 пікселів в ширину, щоб не займати багато місця при роботі з додатком як зображено на рис.3.1.

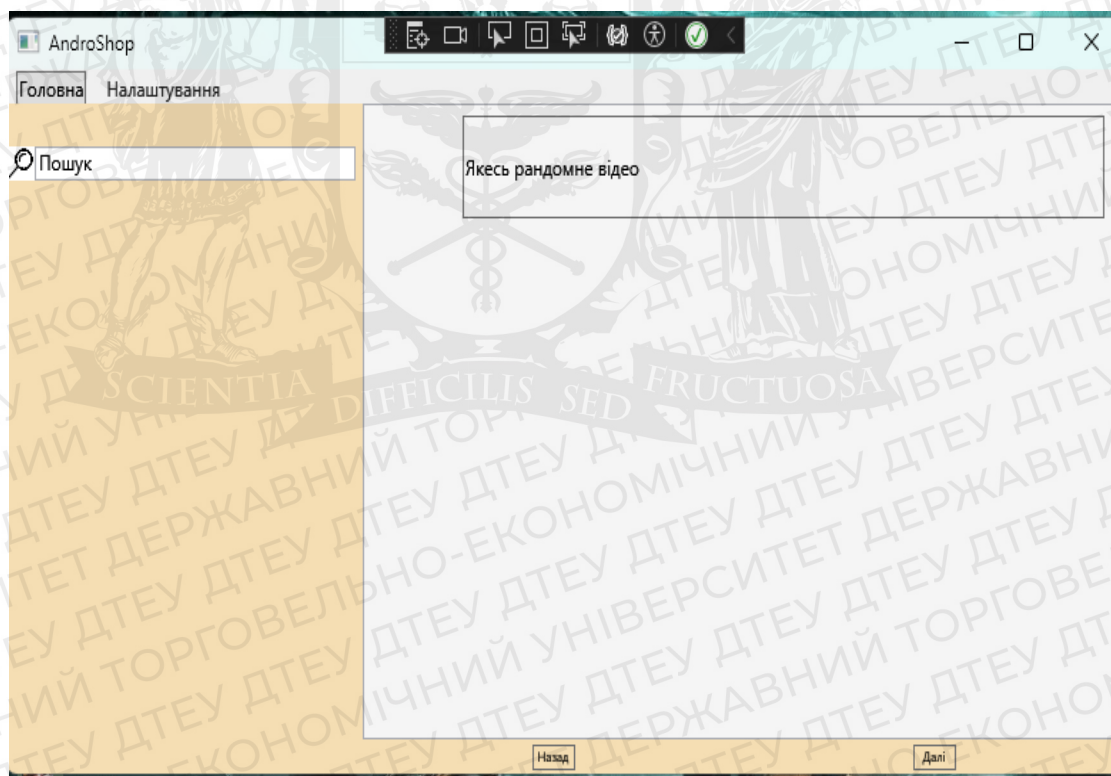


Рис.3.1. Головна сторінка

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 06-08.БР			
Зав. каф.		Криворучко О.В.		14.04.23	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Керівник		Власенко Л. О.		14.04.23		РЗ	30	51
Гарант		Рзаєва С.Л.		14.04.23		Факультет інформаційних технологій		
Розробив		Гармата Б. Ю.		14.04.23		4 курс, 6 група		

При введенні в пошук бажаного запиту будуть відображатися тільки контент, що має відповідну назву. Кнопка Options де можна змінити тему додатку з темної на світлу, що зображено на Рис3.2.

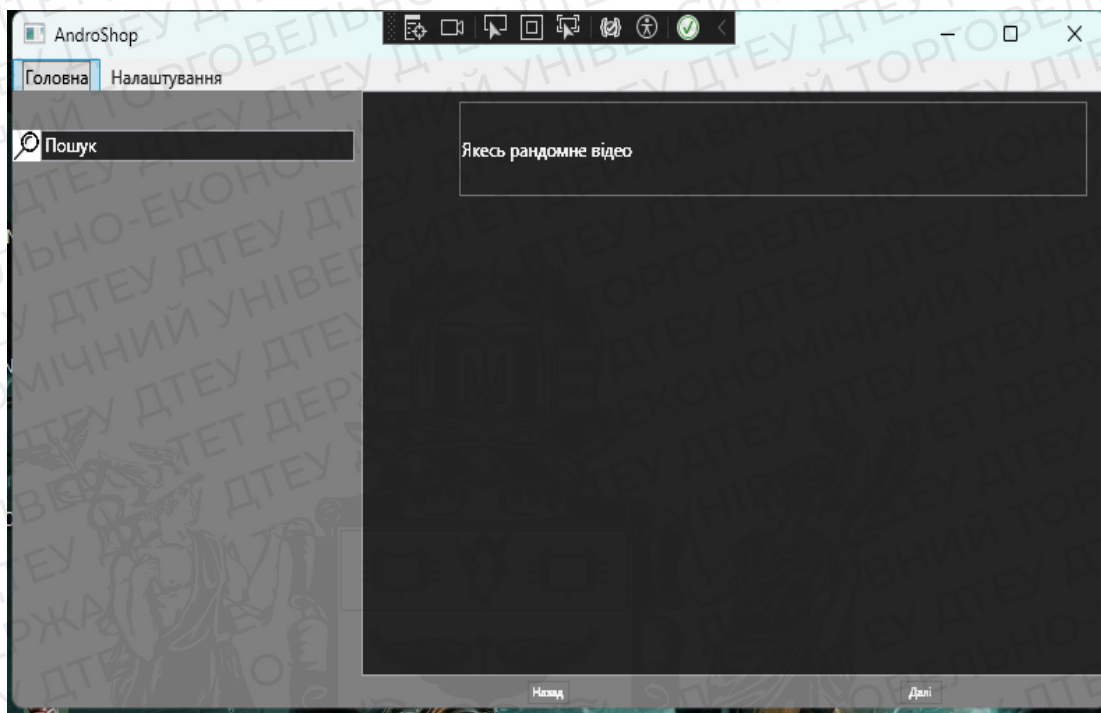


Рис.3.2. Темна тема додатку

При виборі контенту, що зацікавив користувача відкривається нове вікно з детальним переглядом продукту де знаходиться назва, опис контенту відгуки до нього та програвач для перегляду трейлеру до відео контенту. Також зверху знаходиться меню з двома кнопками Main для повернення на головну сторінку та Option для зміни теми додатку з темної на світлу. Також на екрані є кнопка для покупки контенту, як зображено на Рис3.3.

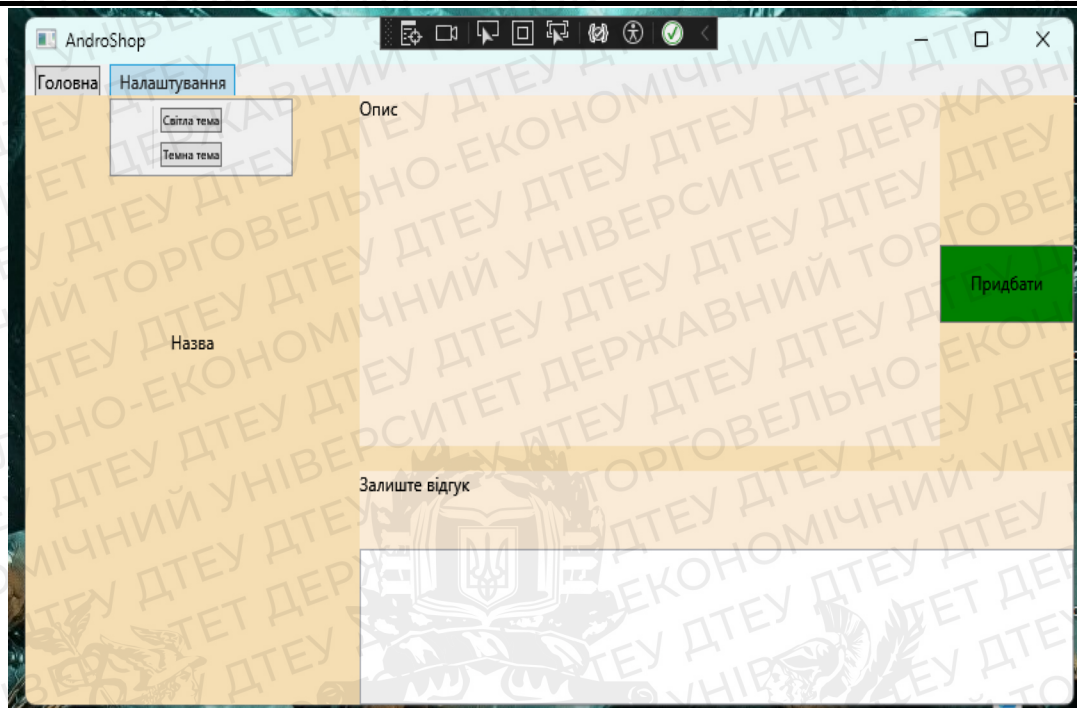


Рис3.3. Вікно вибраного продукту

При натисканні на кнопку «Придбати» додаток відкриє вікно з проханням ввести електронну пошту користувача, як на Рис3.4.

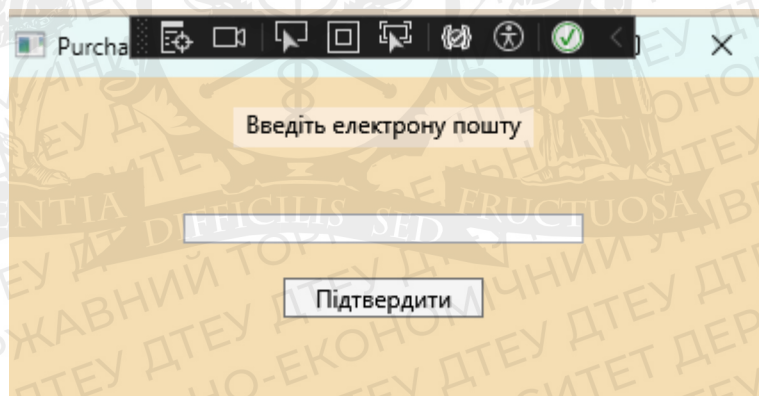


Рис.3.4. Вікно з введенням електронної пошти

Після введення електронної пошти додаток відкриє браузер та перейде на сторінку платіжної системи для оплати продукту. Коли адміністратор побачить перехід коштів на картку та візьме посилання на відповідний файл та відправить його на електронну пошту, що ввів користувач. Після цього користувачу залишається тільки перейти на свою електронну пошту та завантажити відео, що показано на Рис3.5.

						Аркуш
					ДТЕУ 121 06-08.БР	32
Зм.	Аркуш	№ докум	Підпис	Дата		

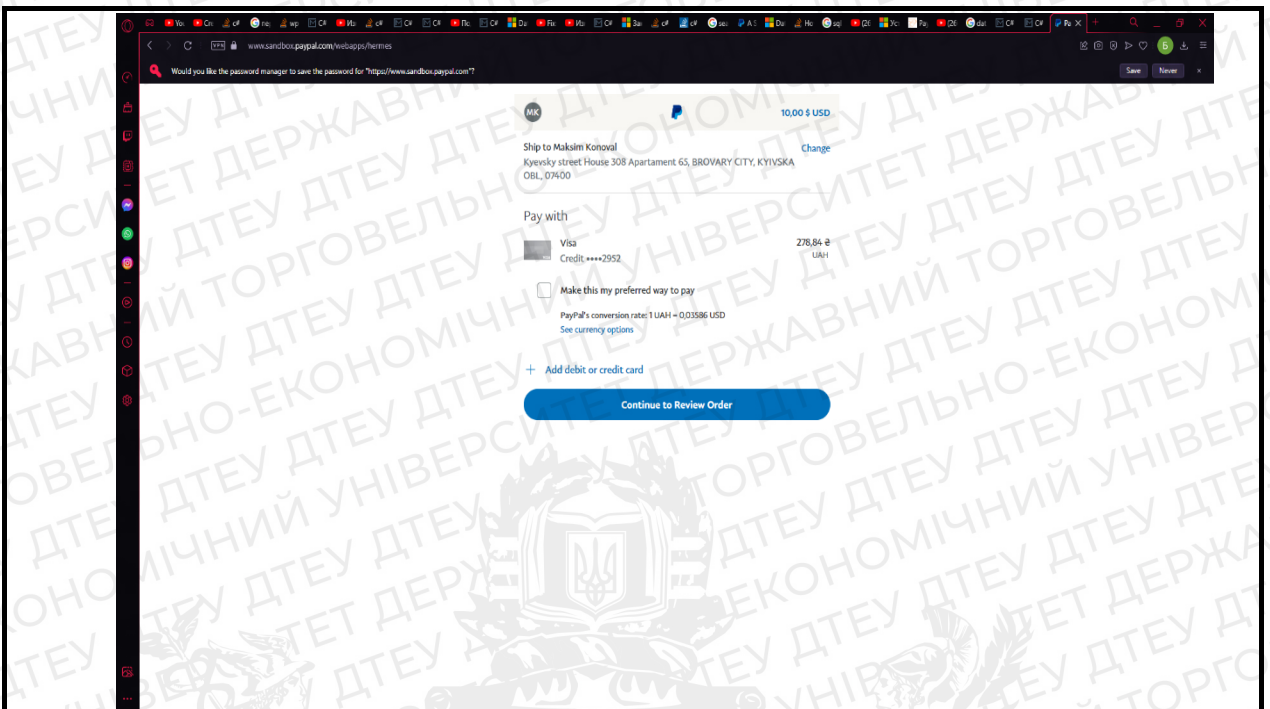


Рис.3.5.Вікно оплати продукту

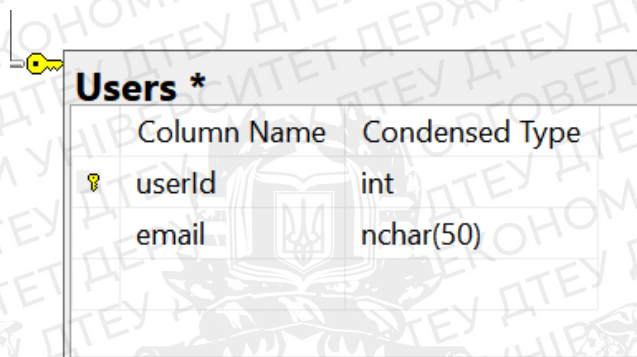
Потім додаток автоматично повернеться на головний екран для подальших покупок.

3.2. Серверна частина

При створенні додатку було вирішено, що для більш зручної роботи з великою кількістю контенту потрібно розробити базу даних. База даних складається з трьох таблиць MediaContent, Purchase, Users відповідно для зберігання інформації про відео контент, про покупки контенту та для зберігання електронної пошти користувачів, що взаємозв'язані між собою. MediaContent зв'язана з Purchase зв'язком один до багатьох. Що означає що в таблиці Purchase може бути багато екземплярів одного і того ж контенту. а в таблиці MediaContent тільки один такий екземпляр з унікальним номером. Таблиця Users зв'язана з Purchase тим же зв'язком але передається тільки індивідуальний номер користувача, що дозволяє відслідкувати хто та коли придбав відповідний контент.

					Аркуш
					33
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-08.БР

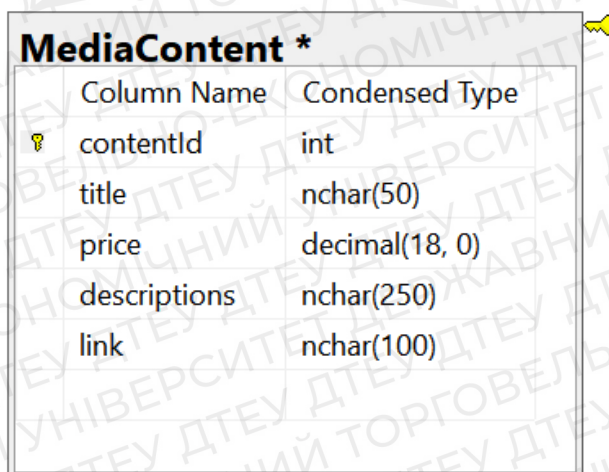
В таблиці Users що зображена на Рис.3.6. знаходиться інформація про користувача, а саме електрона пошта на яку буде надходити посилання на контент та унікальний номер користувача для легшої обробки інформації. Також зберігання електронної пошти допоможе якщо користувача виникли проблеми це дозволяє тримати зв'язок з клієнтом.



Column Name	Condensed Type
userid	int
email	nchar(50)

Рис.3.6. Таблиця Users

В таблиці MainContent зображеної на Рис.3.7. знаходиться інформація про контент, а саме його унікальний номер, його назва, його опис, ціна контенту, посилання на контент.



Column Name	Condensed Type
contentId	int
title	nchar(50)
price	decimal(18, 0)
descriptions	nchar(250)
link	nchar(100)

Рис.3.7. Таблиця MediaContent

В таблиці Purchase зображеної на Рис.3.8 знаходиться інформація про покупки користувача та зберігається дата покупки для моніторингу інформації та унікальні номери контенту та користувача. Щоб побачити коли та який користувач придбав той чи інший контент. Також таке збереження допоможе контролювати який контент подобається більше, а який менше, що дозволить робити більш популярний контент для користувачів та піднімати продажі.

Purchase *	
Column Name	Condensed Type
purchaseId	int
userId	int
contentId	int
purchaseDate	date

Рис.3.8. таблиця Purchase

Всі ці таблиці мають дані що повинні бути захищеними тому для них створюється логін та пароль, котрий знає лише адміністратор, що відповідальний за оновлення інформації. Це використовується для захисту особистих даних користувачів та захисту медіа контенту.

3.3. Висновок до розділу 3

Виходячи з вище зазначеного було розроблено інтерфейс для користувача з вимогою, щоб він був зручним та інтуїтивним для нього. Інтерфейс дозволяє переглядати контент, залишити відгук до контенту, переглянути трейлер контенту та придбати контент. Також було розроблено серверну частину для оновлення даних та зберігання інформації про

користувача та передачі інформації в додаток для представлення контенту користувачу.



								Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-08.БР			36

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

З проведених досліджень можна зробити наступні висновки:

1. Проведено аналіз ринку реалізації медіа контенту з якого видно, що досить вигідною є політика створення власної платформи для реалізації цифрового контенту.
2. В якості операційних систем обрано Windows оскільки вона є найбільш розповсюдженою в світі, що робить її більш рентабельною для поширення власного медіа контенту
3. Проведений аналіз виявив, що безкоштовні платформи для розповсюдження контенту, або мають досить високі вимоги для початку монетизації власного контенту або навіть не мають токової взагалі, що робить створення власної платформи для розповсюдження медіа контенту більш рентабельним та прибутковим
4. В процесі виконання проекту розроблено програмний додаток для реалізації медіа контенту з назвою «AndroShop»
5. Розроблена система повністю задовольняє вимогам до додатку згідно технічного завдання, зокрема, можливість переглянути контент, переглянути опис до контенту, придбати контент та залишити відгук до вибраного контенту, а також пошук відповідного контенту.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 06-08.БР			
Зав. каф.		Криворучко О.В.		28.04.23	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Керівник		Власенко Л. О.		28.04.23		ВП	37	51
Гарант		Рзаєва С.Л.		28.04.23		Факультет інформаційних технологій		
Розробив		Гармата Б. Ю.		28.04.23		4 курс, 6 група		
					Висновки та пропозиції			

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. YouTube Creators «Як заробляти гроші на YouTube» [Електронний ресурс]. – Режим доступу: https://www.youtube.com/intl/uk_ALL/creators/how-things-work/video-monetization/ (дата звернення 21.05.2023)
2. ВУЕ – Велика українська енциклопедія «Авторське право» [Електронний ресурс]: Режим доступу – https://vue.gov.ua/Авторське_право (дата звернення 21.05.2023)
3. Літературознавча енциклопедія: у 2 т. / авт.-уклад. Ю. І. Ковалів. — Київ : ВЦ «Академія», 2007. — Т. 1 : А — Л. — С. 199.»
4. Ranktraker «6 способів запобігти копіюванню вашого контенту» авт. James Westfield [Електронний ресурс] – Режим доступу: <https://www.ranktracker.com/uk/blog/6-ways-to-prevent-your-content-from-being-copied/> (дата звернення 21.05.2023)
5. Legal Support «Умови використання сайту / цифрового сервісу» [електронний ресурс] – Режим доступу: [https://legal-support.top/umovi-vikoristania/#:~:text=Умови%20використання%20сайту%20-%20це%20документ,відповідальності%20адміністрації%20сайту%20\(правовласника\).](https://legal-support.top/umovi-vikoristania/#:~:text=Умови%20використання%20сайту%20-%20це%20документ,відповідальності%20адміністрації%20сайту%20(правовласника).) (дата звернення 21.05.2023)
6. Brander «розробка мобільних додатків Technologies MVVM» [Електронний ресурс] – Режим доступу: <https://brander.ua/technologies/mvvm> (дата звернення 21.05.2023)
7. Про платіжні послуги: Закон від 12.01.2023.№ 2888 XI
8. DOU «Як будувати UML-діаграми. Розбираємо три найпопулярніші варіанти: авт. Юлія Каграманова»[Електронний ресурс] – Режим доступу: <https://dou.ua/forums/topic/40575/> (дата звернення 23.05.2023)

					<i>ДТЕУ 121 06-08.БР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата	«Програмний додаток реалізації цифрового медіа контенту»	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		23.12.22		СВД	38	51
Керівник		Власенко Л. О.		23.12.22		Факультет інформаційних технологій 4 курс, 6 група		
Гарант		Рзаєва С.Л.		23.12.22				
Розробив		Гармата Б. Ю.		23.12.22	<i>Список використаних джерел</i>			

9. Evengreen «UML для бізнес-моделювання: для чого потрібні діаграми процесів авт: Олександр Марголін» Режим доступу – <https://evergreens.com.ua/ua/articles/uml-diagrams.html> (дата звернення 26.05.2023)

10. StudFiles «6.8. Побудова логічної моделі даних» [Електронний ресурс] – Режим доступу: <https://studfile.net/preview/7144845/page:29/> (дата звернення 23.05.2023)



						Аркуш
					ДТЕУ 121 06-08.БР	39
Зм.	Аркуш	№ докум	Підпис	Дата		

ДОДАТКИ

ДОДАТОК А

Код класу ViewModel

```
public class ViewModel : Model
{
    private ObservableCollection<MediaContent> mediaContentList;
    private string email;
    private RoutedCommand searchCommand;
    private RoutedCommand purchaseCommand;
    private RoutedCommand selectMediaContentCommand;

    public void LightTheme()
    {
        var uri = new Uri("/Resource/WhiteTheme.xaml", UriKind.Relative);
        ResourceDictionary resourceDict = Application.LoadComponent(uri) as ResourceDictionary;
        Application.Current.Resources.Clear();
        Application.Current.Resources.MergedDictionaries.Add(resourceDict);
    }

    public void Blacktheme()
    {
        var uri = new Uri("/Resource/BlackTheme.xaml", UriKind.Relative);
        ResourceDictionary resourceDict = Application.LoadComponent(uri) as ResourceDictionary;
        Application.Current.Resources.Clear();
        Application.Current.Resources.MergedDictionaries.Add(resourceDict);
    }

    public void SeleyMediaContentCommand()
    {
        MainWindow mainWindow = new MainWindow();
        Product product = new Product();
        mainWindow.Close();
        product.Show();
    }

    public RoutedCommand SearchCommand
    {
        set
        {
        }
        get { return searchCommand; }
    }

    public void PurchaseCommand ()
    {
        var purchaseCommand = new PurchaseComand();
        purchaseCommand.MakePaymentAsync();
    }
}
```

Код класу PurchaseCommand

```

public class PurchaseComand
{
    public async Task MakePaymentAsync()
    {
        var clientId = "AQiP7U24EF5ZURnDxGRFZ11dIq7uqTEeYQsKI08CNdSze34hVXL_604X9O6PcycO3D2rsiipxEltiTLu";
        var clientSecret = "EEKq6sxxMKfmTH2w9wQXjKjTE2GtcJQq-hLT89L5Z0ZbWBwcZiVkrvGQ0uL6dVHV5D4B7PjdK05a8Ub";

        var environment = new SandboxEnvironment(clientId, clientSecret);
        var client = new PayPalHttpClient(environment);

        var request = new OrdersCreateRequest();
        request.Headers.Add("prefer", "return=representation");
        request.RequestBody(BuildRequestBody());

        var response = await client.Execute(request);
        var statusCode = response.StatusCode;
        var responseBody = response.Result<PayPalCheckoutSdk.Orders.Order>();

        if (statusCode == System.Net.HttpStatusCode.Created)
        {
            var orderId = responseBody.Id;
            var approvalUrl = responseBody.Links.Find(link => link.Rel.ToLower() == "approve").Href;

            Process.Start(approvalUrl);
        }
        else
        {
        }
    }

    private OrderRequest BuildRequestBody()
    {
        var order = new OrderRequest()
        {
            CheckoutPaymentIntent = "CAPTURE",
            PurchaseUnits = new List<PurchaseUnitRequest>()
            {
                new PurchaseUnitRequest()
                {
                    AmountWithBreakdown = new AmountWithBreakdown()
                    {
                        CurrencyCode = "USD",
                        Value = "10.00"
                    }
                }
            }
        };

        return order;
    }
}

```


Код xaml файлу MainWindow

```

<Window x:Class="AndroShop_v2.MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:local="clr-namespace:AndroShop_v2"
  mc:Ignorable="d"
  Title="AndroShop" Height="450" Width="800"
  HorizontalAlignment="Center"
  VerticalAlignment="Center"
  Style="{DynamicResource WindowBackgroundStyle}"
  >
  <Grid>
  <Grid.RowDefinitions>
  <RowDefinition Height="20"/>
  </Grid.RowDefinitions>
  <DockPanel>
  <Menu DockPanel.Dock="Top" Height="25">
  <Button Content="Головна" Click="Button_Click_Main"/>
  <MenuItem Header="Налаштування">
  <Button Height="16" Content="Світла тема" FontSize="8" Click="Button_Click_White"/>
  <Button Height="16" Content="Темна тема" FontSize="8" Click="Button_Click_Black"/>
  </MenuItem>
  </Menu>
  </DockPanel>
  <Grid Grid.Row="1">
  <Grid.ColumnDefinitions>
  <ColumnDefinition Width="250"/>
  </Grid.ColumnDefinitions>
  </Grid.ColumnDefinitions>
  <Grid Grid.Column="0">
  <Grid.ColumnDefinitions>
  <ColumnDefinition Width="89*"/>
  <ColumnDefinition Width="161*"/>
  </Grid.ColumnDefinitions>
  <Grid.RowDefinitions>
  <RowDefinition Height="25"/>
  <RowDefinition Height="20"/>
  </Grid.RowDefinitions>
  <Grid.RowDefinitions>
  <Grid.ColumnSpan="2" Margin="15,0,0,0">Пошук</TextBlock>
  <Image Grid.Row="1" HorizontalAlignment="Left" Width="20" Height="20" Source="/Resource/Images/SearchImage.png"
  ></Image>
  <Grid>
  <Grid.Column="1">
  <Grid.RowDefinitions>
  <RowDefinition/>
  <RowDefinition Height="20"/>
  </Grid.RowDefinitions>
  <Grid>
  <ListBox Style="{DynamicResource ListStyle}">
  <ListBoxItem Margin="3">
  <StackPanel Orientation="Horizontal">
  <Image Width="60" Height="60"/>
  <Button Style="{DynamicResource ButtonStyle}" Width="450" Click="Button_Click">
  <TextBlock Width="450">Якесь випадкове відео</TextBlock>
  </Button>
  </StackPanel>
  </ListBoxItem>
  </ListBox>
  </Grid>
  </Grid.Column>
  </Grid>
  <Grid Grid.Row="1">
  <Grid.ColumnDefinitions>
  <ColumnDefinition/>
  <ColumnDefinition/>
  </Grid.ColumnDefinitions>
  </Grid.ColumnDefinitions>

```

Продовження додатку В

```
<Button Style="{DynamicResource ButtonStyle}" Grid.Column="0" Height="15" Width="30" FontSize="8" Name="rightBtn"
Background="{x:Null}" Content="Назад" />
<Button Style="{DynamicResource ButtonStyle}" Grid.Column="1" Height="15" Width="30" FontSize="8" Content="Дані"
Name="leftbtn" Background="{x:Null}" />
</Grid>
</Grid>
</Window>
```



Код xaml файлу Product

```

<Window x:Class="AndroShop_v2.Product"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    Title="AndroShop" Height="450" Width="800"
    HorizontalAlignment="Center"
    VerticalAlignment="Center"
    Style="{DynamicResource WindowBackgroundStyle}"
>
<Grid>
<Grid.RowDefinitions>
<RowDefinition Height="20"/>
<RowDefinition/>
</Grid.RowDefinitions>
<DockPanel>
<Menu DockPanel.Dock="Top" Height="25">
<Button Content="Головна" Click="Button_Click_Main"/>
<MenuItem Header="Налаштування">
<Button Height="16" Content="Світла тема" FontSize="8" Click="Button_Click_White"/>
<Button Height="16" Content="Темна тема" FontSize="8" Click="Button_Click_Black"/>
</MenuItem>
</Menu>
</DockPanel>
<Grid Grid.Row="1">
<Grid.ColumnDefinitions>
<ColumnDefinition Width="250"/>
</ColumnDefinition>
<ColumnDefinition/>
</Grid.ColumnDefinitions>
<Grid Grid.Column="0">
<Grid.RowDefinitions>
<RowDefinition Height="150"/>
<RowDefinition/>
</Grid.RowDefinitions>
<Image Width="100" Height="100"/>
<TextBlock Grid.Row="1" TextAlignment="Center">Назва</TextBlock>
</Grid>
<Grid Grid.Column="1">
<Grid.RowDefinitions>
<RowDefinition/>
<RowDefinition Height="150"/>
</Grid.RowDefinitions>
<TextBlock Style="{DynamicResource TextBlockStyle}" Grid.Row="1" Height="50" VerticalAlignment="Top">Залиште
відгук</TextBlock>
<TextBox Style="{DynamicResource TextBoxStyle}" Grid.Row="1" Height="100" VerticalAlignment="Bottom"></TextBox>
</Grid>
<Grid.ColumnDefinitions>
<ColumnDefinition/>
<ColumnDefinition Width="100"/>
</Grid.ColumnDefinitions>
<TextBlock Height="225" VerticalAlignment="Top" Style="{DynamicResource TextBlockStyle}">Опис</TextBlock>
<Button Grid.Column="1" Height="50" Background="Green" RenderTransformOrigin="0.5,0.5"
Click="Button_Click_Payment">Придбати</Button>
</Grid>
</Grid>
</Grid>
</Window>

```

Код xaml файлу Purchase

```

<Window x:Class="AndroShop_v2.Purchase"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    HorizontalAlignment="Center"
    VerticalAlignment="Center"
    Title="Purchase" Height="200" Width="400"
    Style="{DynamicResource WindowBackgroundStyle}">
<Grid >
<Grid.RowDefinitions>
<RowDefinition Height="50"/>
<RowDefinition Height="50"/>
<RowDefinition Height="50"/>
</Grid.RowDefinitions>
<TextBlock Grid.Row="0" Style="{DynamicResource TextBlockStyle}" TextAlignment="Center"
    Width="150" Height="20">Введіть електрону пошту</TextBlock>
<TextBox Grid.Row="1" Name="emailField" Style="{DynamicResource TextBoxStyle}" Width="200" Height="15"/>
<Button Grid.Row="2" VerticalAlignment="Top" Style="{DynamicResource ButtonStyle}" Content="Підтвердити"
    Width="100" Height="20" Click="Button_Click"/>
</Grid>
</Window>

```



Код класу MainWindow

```
public partial class MainWindow : Window
{
    ViewModel commands = new ViewModel();

    public MainWindow()
    {
        InitializeComponent();
    }

    private void Button_Click(object sender, RoutedEventArgs e)
    {
        Product product = new Product();
        Close();
        product.Show();
    }

    private void Button_Click_Main(object sender, RoutedEventArgs e)
    {
        if (IsEnabled != true)
        {
            Show();
        }
        else
        {
        }
    }

    private void Button_Click_Black(object sender, RoutedEventArgs e)
    {
        commands.Blacktheme();
    }

    private void Button_Click_White(object sender, RoutedEventArgs e)
    {
        commands.LightTheme();
    }
}
```



Код класу Product

```
public partial class Product : Window
{
    MainWindow mainWindow = new MainWindow();
    ViewModel viewModle = new ViewModel();
    public Product()
    {
        InitializeComponent();
    }

    private void Button_Click_Main(object sender, RoutedEventArgs e)
    {
        mainWindow.Show();
        Close();
    }

    private void Button_Click_Black(object sender, RoutedEventArgs e)
    {
        viewModle.Blacktheme();
    }

    private void Button_Click_White(object sender, RoutedEventArgs e)
    {
        viewModle.LightTheme();
    }

    private void Button_Click_Payment(object sender, RoutedEventArgs e)
    {
        Purchase purchase = new Purchase();
        purchase.Show();
    }
}
```



Код класу Purchase

```
public partial class Purchase : Window
{
    ViewModel viewModel = new ViewModel();
    public Purchase()
    {
        InitializeComponent();
    }

    private void Button_Click(object sender, RoutedEventArgs e)
    {
        if (emailField.Text != string.Empty)
        {
            viewModel.PurchaseCommand();
            Close();
        }
        else
        {
            Close();
        }
    }
}
```

