

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

Кафедра інженерії програмного забезпечення

КУРСОВА РОБОТА

з курсу: «Об'єктно-орієнтоване програмування»

на тему: «Розроблення програмного забезпечення з використанням
об'єктно-орієнтованої парадигми»

Виконав: студент (ка) 2 курсу групи 611пст
Спеціальність 121 – Інженерія програмного
забезпечення

(код та найменування)

Хвостовця К.К.

Керівник: _____
(прізвище й ініціали студента(ки))

Лучшев П.О.

(прізвище й ініціали)

Національна шкала _____

Кількість балів: _____

Оцінка: ECTS _____

Члени комісії _____ Шевченко І.В.

(підпис)

(прізвище та ініціали)

_____ Дем'яненко В.А.

(підпис)

(прізвище та ініціали)

Харків – 2024

ТИПОВЕ ЗАВДАННЯ НА КУРСОВУ РОБОТУ

Варіант 18 Платні навчальні курси

Критерій оцінювання	Бали	Штрафи
Покрокове виконання курсової роботи:		
Розділ 1 (крайній термін – 3 тиждень)	0..15	-3
Розділ 2 (крайній термін – 6 тиждень)	0..15	-3
Розділ 3 (крайній термін – 9 тиждень)	0..15	-3
Розділ 4 (крайній термін – 12 тиждень)	0..15	-3
Розділ 5 (крайній термін – 13 тиждень)	0..5	-1
Оформлення пояснювальної записки (термін – 14 тиждень)	0..3	-1
Підготовка до захисту курсової роботи:		
доповідь з презентацією (термін – 15 тиждень)	0..1	
відеоролик з демонстрацією роботи програми (термін – 15 тиждень)	0..1	
Захист курсової роботи:		
доповідь з презентацією	0..5	
демонстрація роботи програми і github-репозиторію	0..5	
відповіді на питання	0..5	
використання*: 1) інкапсуляції 2) статичних членів класів 3) інтерфейсів 4) абстрактних класів 5) спадкування 6) поліморфізму 7) .NET-delegates і events 8) сереалізації/десереалізації об'єктів 9) Collections.Generic 10) LINQ * мають бути представленні на захисті курсової роботи у презентації	0..15	
Зсього за курсову роботу:	0..100	0..14

Зміст

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ	4
ВСТУП	5
1 ФУНКЦІОНАЛЬНІ ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	6
1.1 Аналіз функціональності програмних аналогів	6
1.2 Побудова Mind Map для заданої предметної області	6
1.3 Виділення ролей користувачів і формулювання функціональних вимог	7
1.4 Проектування інтерфейсу користувача	8
1.5 Розроблення функціональних тестів	9
1.5.1 Функціональні тести для ролі користувача «Адміністратор»	9
1.5.2 Функціональні тести для ролі користувача «Зареєстрований користувач»	10
1.5.3 Функціональні тести для ролі користувача «Гість»	10
2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ	12
2.1 Виділення і опис класів предметної області	12
2.2 Встановлення зв'язків між класами	13
3 ПРОГРАМНА РЕАЛІЗАЦІЯ КЛАСІВ ПРЕДМЕТНОЇ ОБЛАСТІ І ЇХ ТЕСТУВАННЯ	14
3.1 Структура проекту з реалізацією класів предметної області	14
3.2 Реалізація інтерфейсів і каркасів класів предметної області	14
3.3 Розроблення unit-тестів для класів предметної області	14
3.4 Повна реалізація класів предметної області	15
3.5 Результати unit-тестування класів предметної області	15
4 ПРОЄКТУВАННЯ І ПРОГРАМНА РЕАЛІЗАЦІЯ КЛАСІВ ІНТЕРФЕЙСУ КОРИСТУВАЧА	16
4.1 Структура проекту з реалізацією класів інтерфейсу користувача	16
4.2 Виділення класів для реалізації інтерфейсу користувача	16
4.3 Програмна реалізація класів інтерфейсу користувача	16
5 ФУНКЦІОНАЛЬНЕ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	17
5.1 Функціональне тестування для ролі користувача «Адміністратор»	17
5.2 Функціональне тестування для ролі користувача «Зареєстрований користувач»	18
5.3 Функціональне тестування для ролі користувача «Гість»	18
ВИСНОВКИ	19
ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАНЬ	20
ДОДАТОК А. Лістинг класів предметної області	21
ДОДАТОК Б. Лістинг класів інтерфейсу користувача	22
ДОДАТОК В. Лістинг класів unit-тестів	23
ДОДАТОК Г. Назва додатку	24

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

- CRUD – (англ. Create Read Update Delete) основні функції управління даними «створення, читання, оновлення і вилучення».
- ООП – об'єктно-орієнтоване програмування
- ...

ВСТУП

Чому тема є актуальною?

Хто є потенційним користувачем?

Навести лідерів серед програмних аналогів.

Зазначити стек-технологій, який буде використаний.

У сучасному світі, навчання та розвиток навичок стали важливою складовою успіху в будь-якій сфері діяльності. З цим пов'язано зростання популярності платних навчальних курсів, що надають можливість здобути нові знання та вдосконалити навички в зручний для користувача спосіб. Ця тема є актуальною через постійний розвиток технологій та зміну вимог ринку праці, що ставлять перед людьми необхідність постійного самовдосконалення.

Потенційними користувачами платних навчальних курсів є люди з різних сфер діяльності та рівнів кваліфікації, які мають бажання вдосконалювати свої навички, отримувати нові знання або перекваліфікуватися для успішної кар'єри чи особистого розвитку.

Серед лідерів серед програмних аналогів можна виокремити такі платформи, як Udey та Coursera, які надають доступ до великого вибору курсів у різних галузях знань та мають високу репутацію серед користувачів.

У даній курсовій роботі буде розглянуто стек технологій, що використовуються для створення та розвитку платних навчальних курсів, включаючи такі складові, як веб-фреймворки, мови програмування, системи управління контентом та інші інструменти, що забезпечують ефективне функціонування та взаємодію з користувачами.

Не забуваємо наводити правильно оформлені посилання. Для ручного оформлення посилань звернутися до [1], для автоматичного – до [2] із «Перелік джерел і посилань».

Вступ має займати не більше 1 стор.

1 ФУНКЦІОНАЛЬНІ ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Аналіз функціональності програмних аналогів

1.1.1 Udemу

Посилання на сервіс: <https://www.udemy.com/>

Udemу - це платформа з онлайн-навчання, де користувачі можуть знайти та купити курси з різних тематик, що варіюються від програмування до мистецтва. Кожен курс на Udemу створений і ведеться незалежним викладачем або експертом відповідної галузі. Платформа пропонує більше 150 000 курсів у різних мовах.

Сервіс дозволяє:

- користувачам придбати доступ до широкого спектру онлайн-курсів з різних областей знань.
- Кожен курс має відеоуроки, текстові матеріали та тести для оцінки знань.
- Користувачі можуть самостійно вибирати курси, вивчати матеріали у зручний для них час та темп, а після успішного завершення отримувати сертифікати.

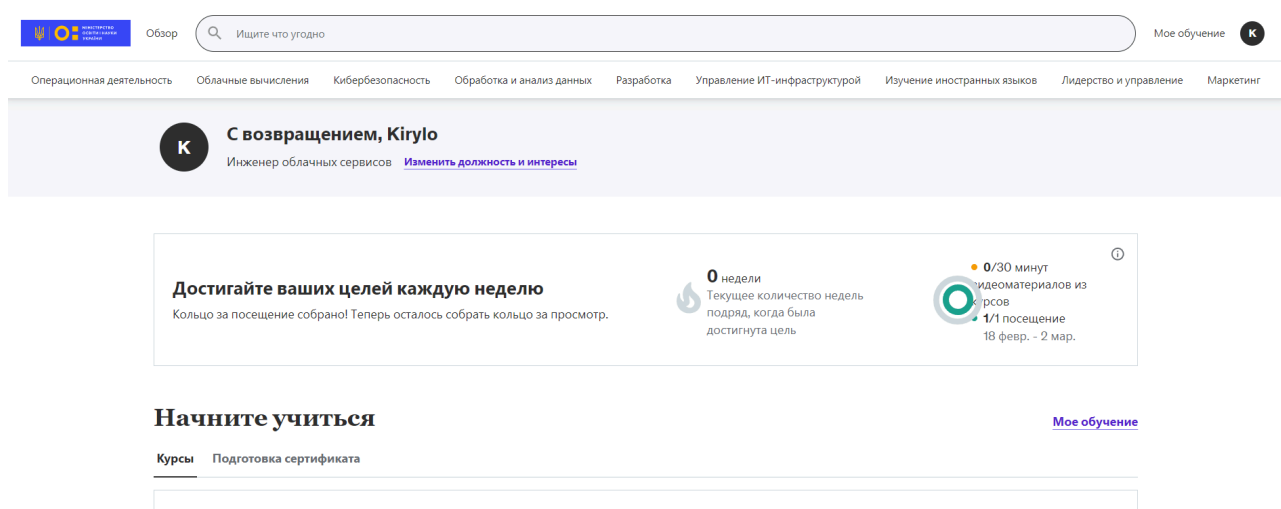


Рисунок 1.1 – Головна сторінка Udemу

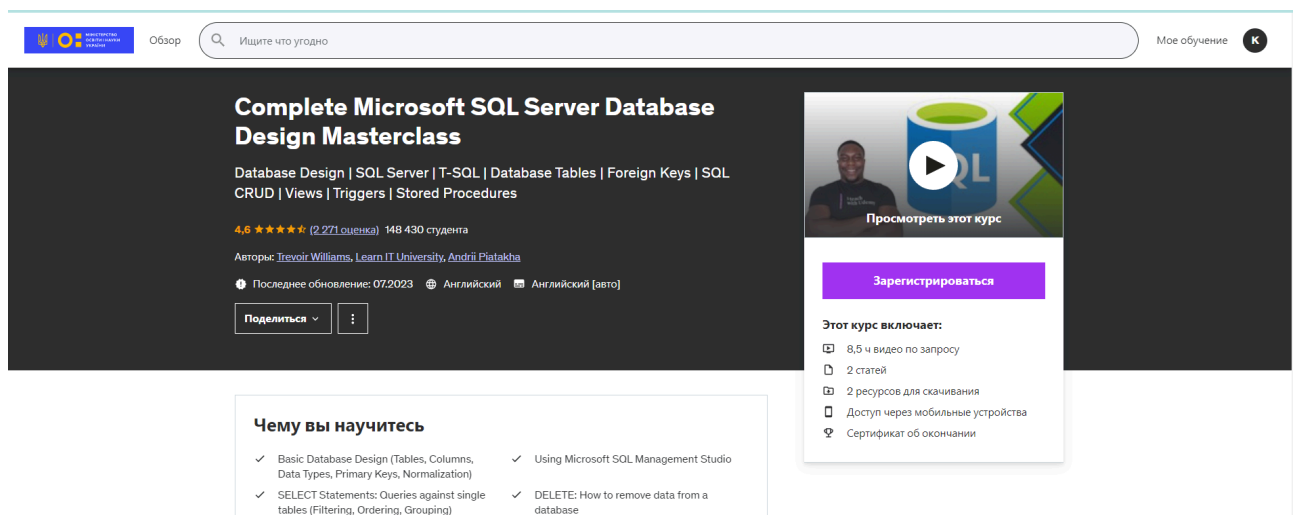


Рисунок 1.2 – Сторінка курсу у Udemy

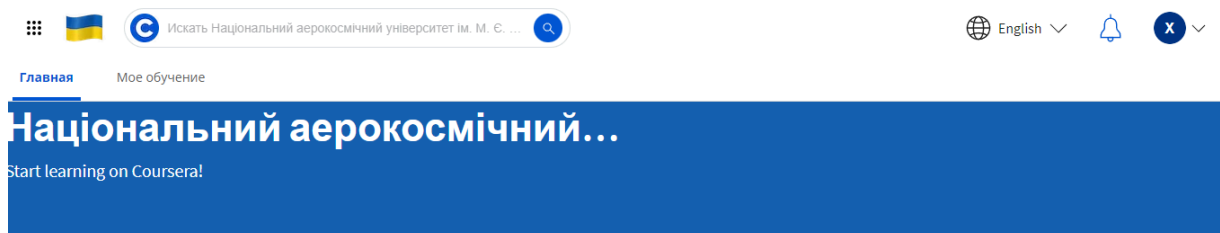
1.1.2 Coursera

Посилання на сервіс: <https://www.coursera.org/>

Coursera - це платформа з онлайн-навчання, яка співпрацює з університетами та вищими навчальними закладами для надання курсів відомими викладачами та експертами. Курси на Coursera зазвичай розроблені університетами та інститутами світового класу, такими як Стенфордський, Єльський, та іншими. Платформа пропонує більше 4 000 курсів у різних галузях знань.

Сервіс дозволяє:

- користувачам отримувати доступ до курсів відомих університетів та коледжів з усього світу.
- Кожен курс має лекції, практичні завдання та тести для перевірки знань.
- Користувачі можуть вивчати матеріали у своєму темпі, отримувати сертифікати від університету та брати участь у спеціалізованих програмах для глибокого вивчення конкретних тематик.



Рекомендации от Coursera

Зарегистрируйтесь на популярные курсы с учетом данных о регистрации и оценок учащихся из различных отраслей на Coursera.

Самые популярные сертификаты

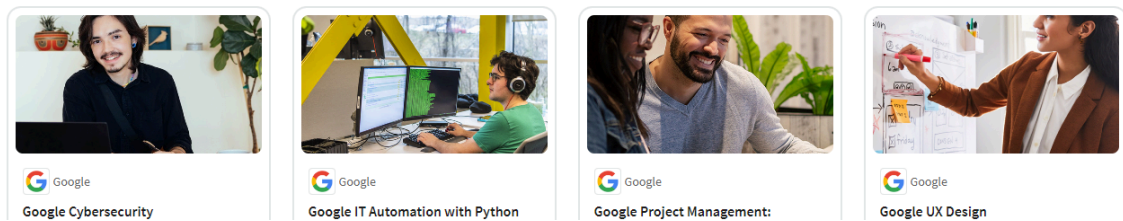


Рисунок 1.3 – Головна сторінка Coursera

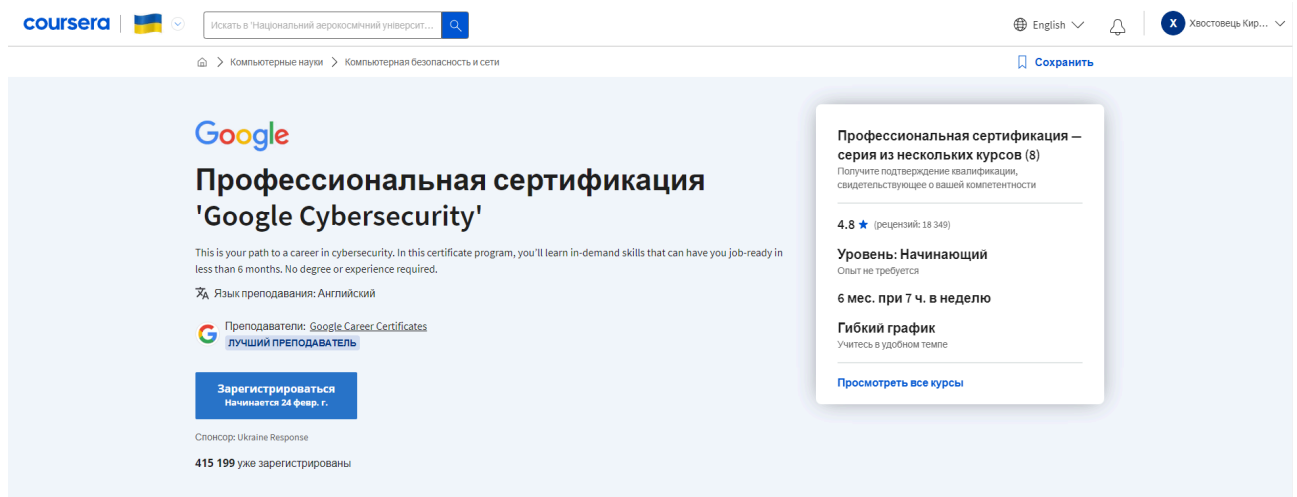


Рисунок 1.4 – Сторінка курсу у Coursera

1.2 Побудова Mind Map для заданої предметної області

Для систематизації інформації, побудуємо Mind Map для застосунку платних курсів Learnoria.

Mind Map була побудована за допомогою онлайн-інструментів MindMeister (<https://www.mindmeister.com>)

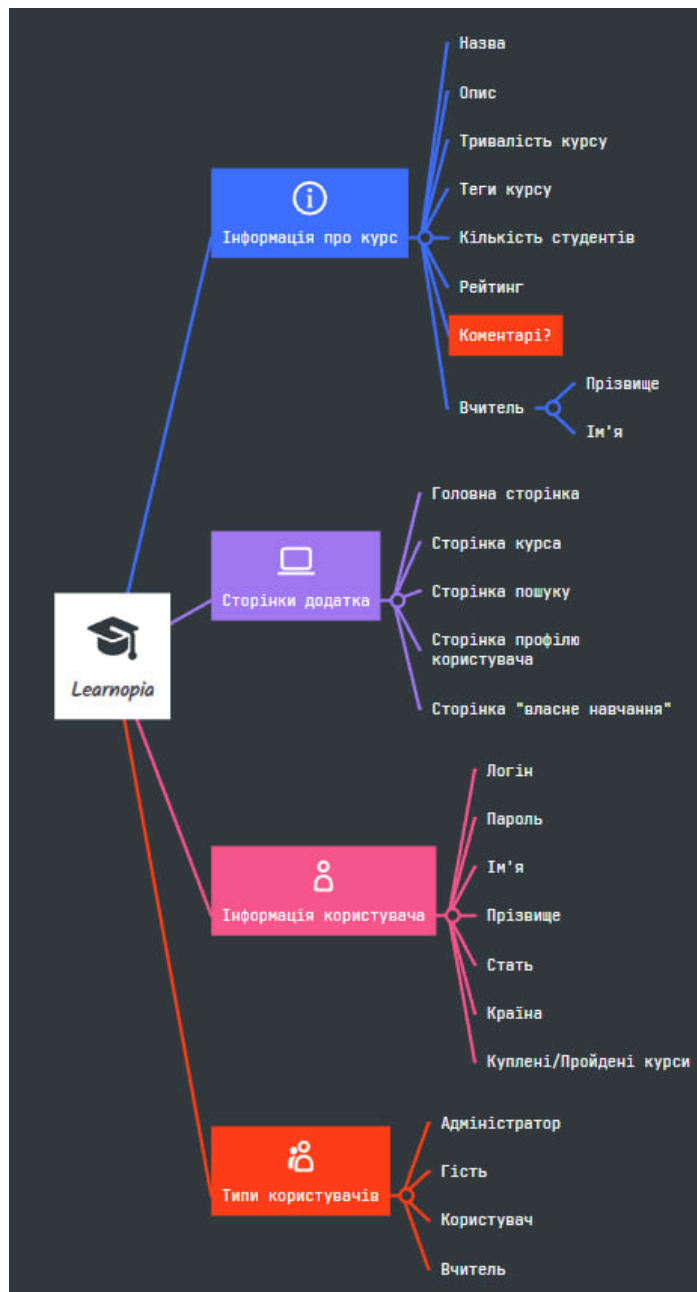


Рисунок 1.5 – Mind map платформи платних курсів Learnopia

Посилання на розроблений Mind map:

<https://mm.tt/app/map/3174780711?t=aLYzuWDjnd>

1.3 Виділення ролей користувачів і формулювання функціональних вимог

Були виділені три чотири користувачів: «Адміністратор», «Зареєстрований користувач», «Викладач», «Гість».

Опишемо функціональні вимоги кожного типу користувачів.

Таблиця 1.1 – Функціональні вимоги користувача з роллю «Адміністратор»

Іденти-фікатор	Функціональні вимоги
A.1.	Адміністратор може додати/редагувати/видалити курси на платформі платних курсів. <i>Примітка:</i> якщо адміністратор додає курс, то у курсу немає викладача
A.2.	Адміністратор може переглядати загальну статистику по усій платформі: активності користувачів, кількість і зареєстрованих користувачів, кількість придбаних курсів, кількість відвідувань тощо.
A.3.	Адміністратор може змінювати користувачів
A.4.	Можливість зберігати та завантажувати інформацію про курси та користувачів у форматі JSON для архівування або резервного копіювання

Таблиця 1.2 – Функціональні вимоги користувача з роллю «Зареєстрований користувач»

Іденти-фікатор	Функціональні вимоги
ЗК.1.	Зареєстрований користувач може авторизуватися. <i>Примітка:</i> не можна авторизуватися з невірним логіном і паролем.
ЗК.2.	Зареєстрований користувач може переглядати курси.
ЗК.3.	Зареєстрований користувач може шукати курси.
ЗК.4.	Зареєстрований користувач може покупати курси <i>Примітка:</i> курс може бути безкоштовним.
ЗК.5.	Зареєстрований користувач може моніторити власний прогрес проходження курсів. <i>Примітка:</i> прогрес можна дивитися тільки у тих курсів, що проходяться
ЗК.6.	Зареєстрований користувач має особистотий профіль, який він може змінювати.
ЗК.7.	Зареєстрований користувач може оцінювати курси. <i>Примітка:</i> оцінювати можна тільки ті курси, що пройшов користувач

Таблиця 1.3 – Функціональні вимоги користувача з роллю «Гість»

Іденти-фікатор	Функціональні вимоги
Г.1.	Гість може зареєструватися, вказавши логін і пароль. <i>Примітка:</i> 1) логін не має збігатися з існуючими; 2) довжина логіну не менше п'яти символів;

	3) довжина пароллю не менше п'яти символів;
Г.2.	Гість може переглядати курси
Г.3.	Гість може шукати курси

Таблиця 1.4 – Функціональні вимоги користувача з роллю «Викладач»

Іденти-фікатор	Функціональні вимоги
В.1.	Викладач може зареєструватися, вказавши логін і пароль. <i>Примітка:</i> 1) логін не має збігатися з існуючими; 2) довжина логіну не менше п'яти символів; 3) довжина пароллю не менше п'яти символів;
В.2.	Викладач може створити та редагувати свої курси
В.3.	Викладач може переглядати учасників курсів та їх прогрес проходження курсу <i>Примітка:</i> перегляд користувачів тільки власних курсів
В.4.	Можливість зберігати свої курси та інформацію про учасників у форматі JSON

1.4 Проєктування інтерфейсу користувача

Для реалізації замовником було обрано desktop-застосунок.

На основі сформульованих функціональних вимог були розроблені екранні форми, які наведені на рис. 1.6. – 1.11.

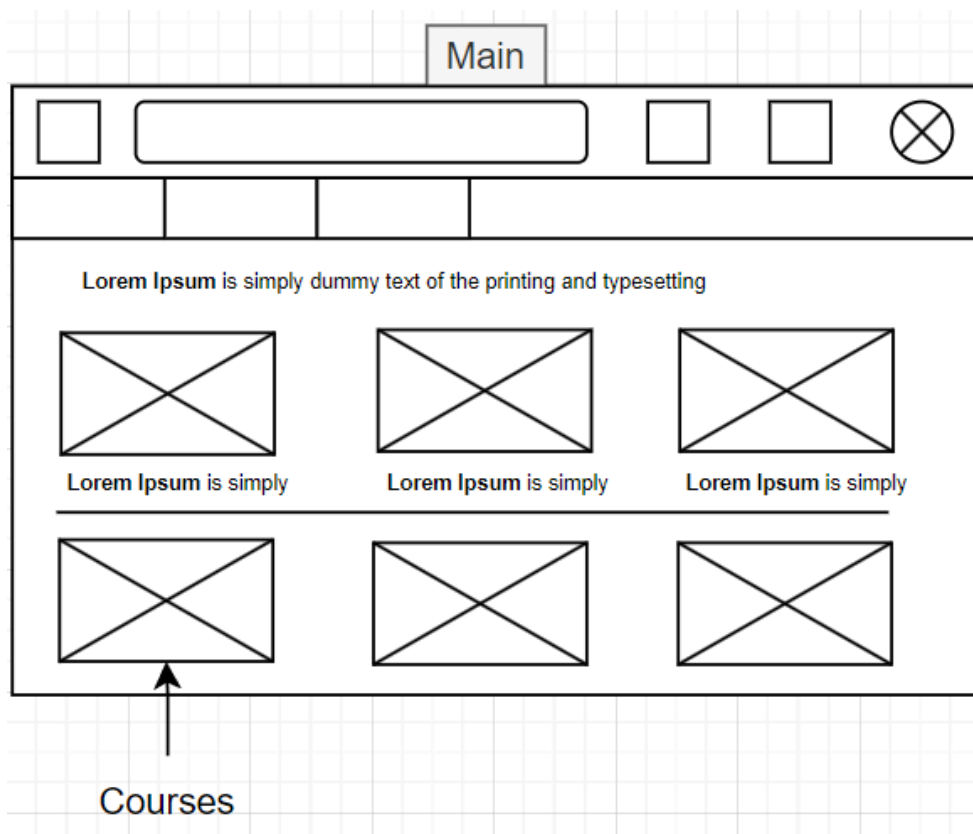


Рисунок 1.6 – Макет екранної форми «Головна сторінка»

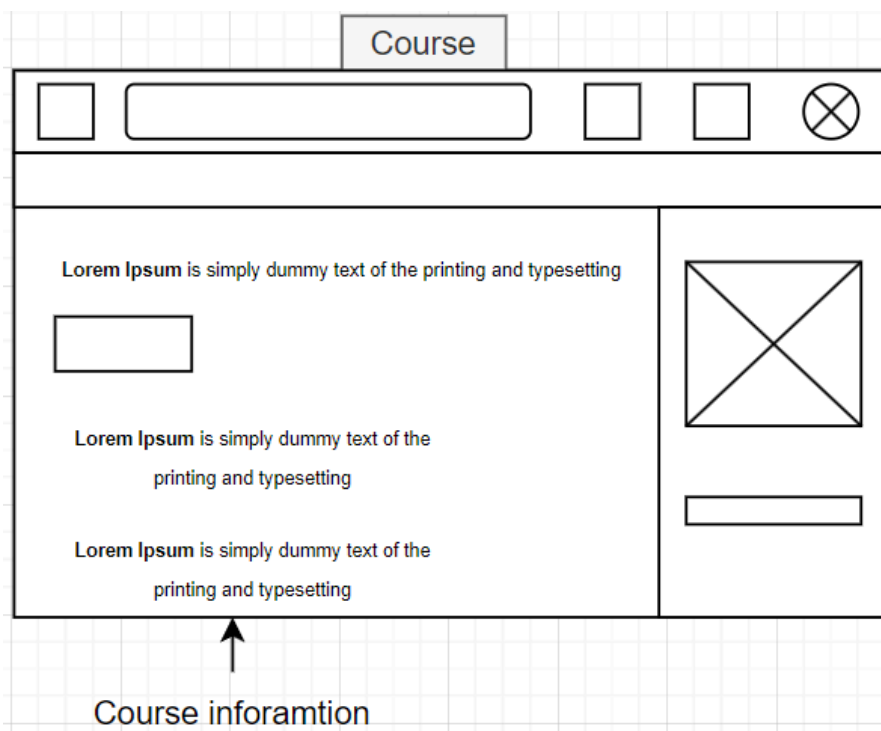


Рисунок 1.7 – Макет екранної форми «Сторінка курсу»

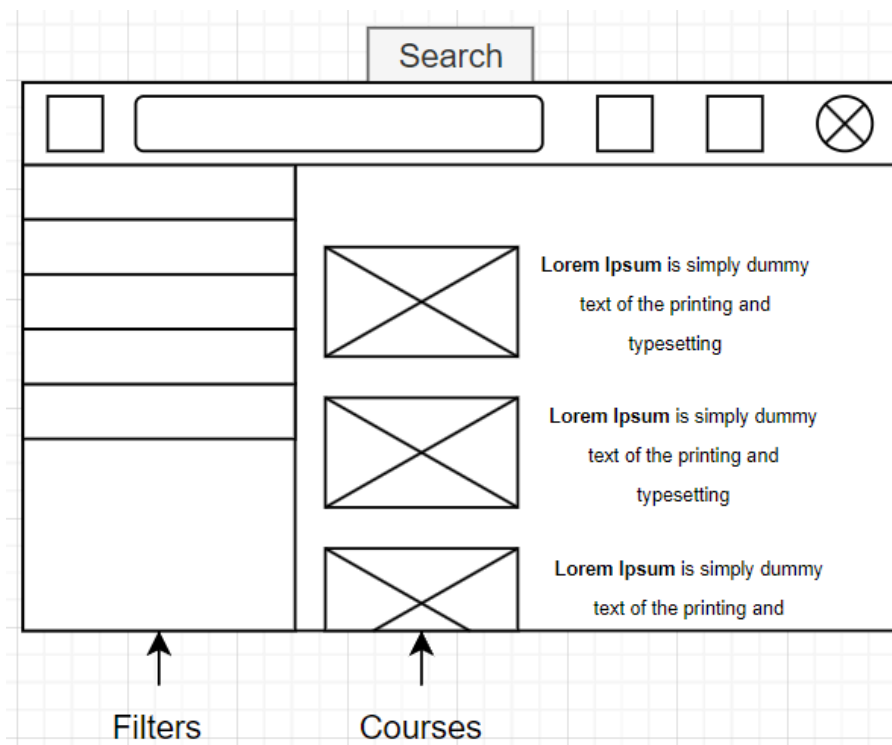


Рисунок 1.8 – Макет екранної форми «Пошук»

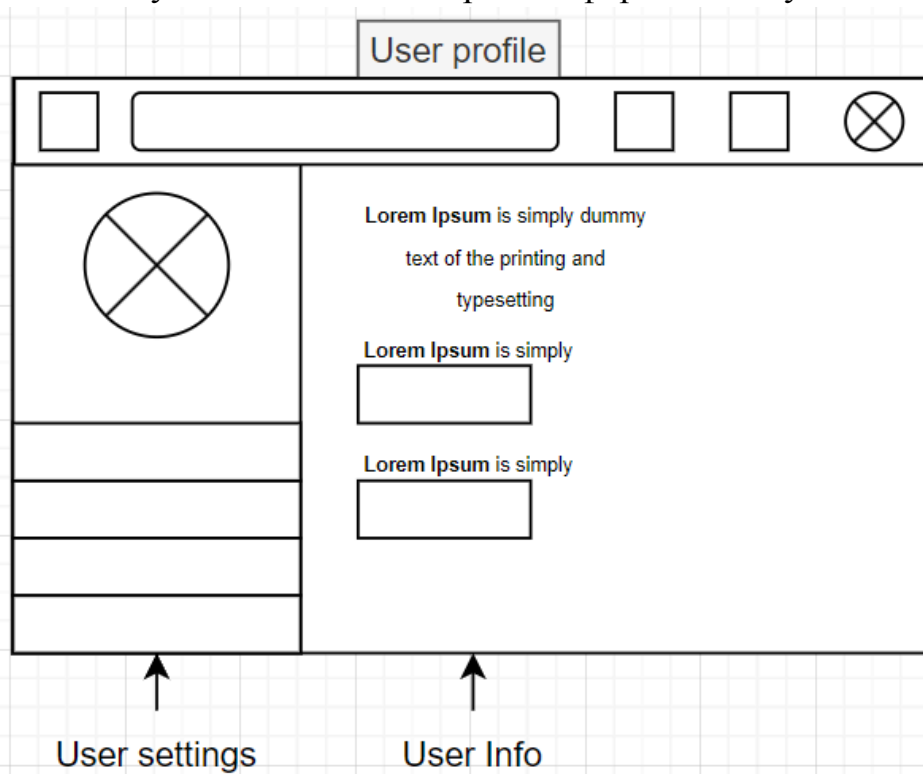


Рисунок 1.9 - Макет екранної форми «Профіль користувача»

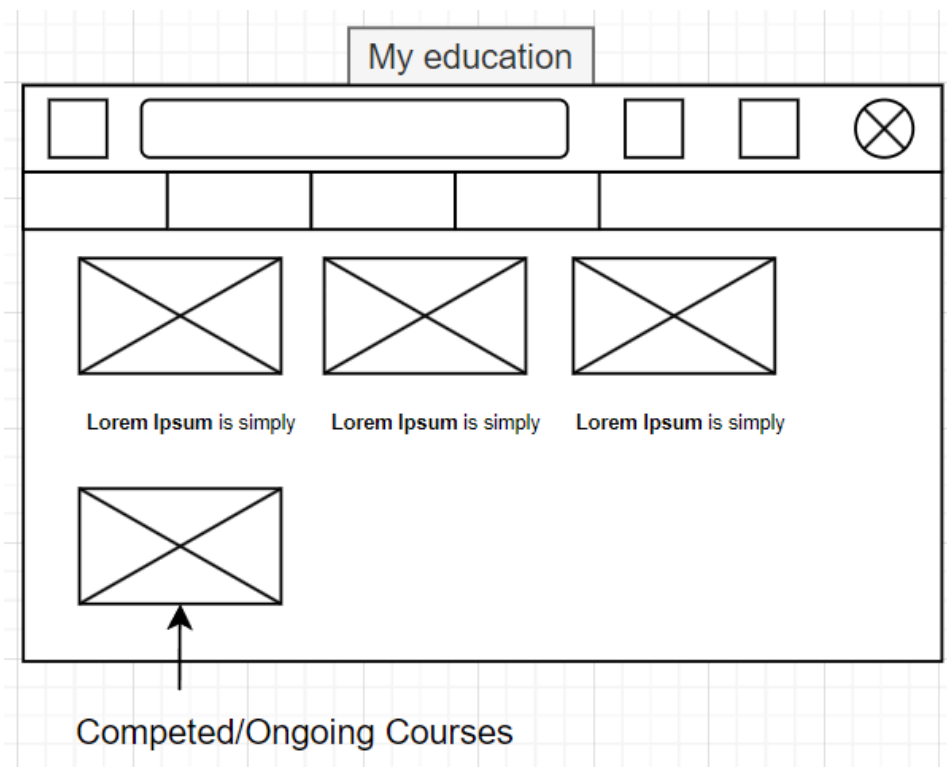


Рисунок 1.10 - Макет екранної форми «Моє навчання»

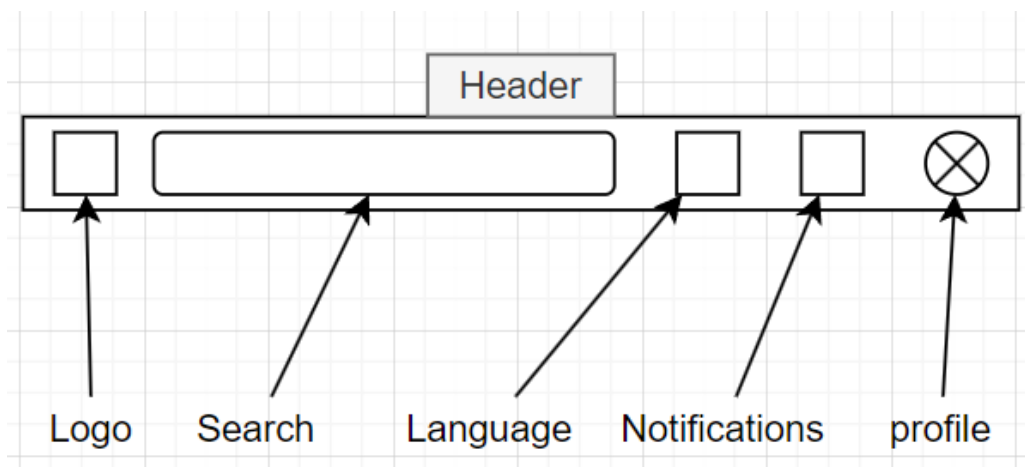


Рисунок 1.11 - Макет елемента «Хедер»

1.5 Розроблення функціональних тестів

1.5.1 Функціональні тести для ролі користувача «Адміністратор»

На основі опису предметної області, а також функціональних вимог і екранних форм були розроблені функціональні тести для ролі користувача «Адміністратор» (див. табл. 1.4).

Таблиця 1.5 – Функціональні тести для ролі користувача «Адміністратор»

Ідентифікатор тесту	Послідовність дій користувача	Очікуваний результат
<i>А.1. Адміністратор може додати/редагувати/видалити курси на платформі платних курсів.</i>		
A.1.1	1. Ввести назву курсу. 2. Ввести опис курсу. 3. Додати тему курсу. 4. Ввести ціну курсу. 5. Натиснути кнопку «Додати».	Успішне додавання курсу
A.1.2	1. У пошуку знайти потрібний курс. 2. Натиснути на сторінку курсу. 3. Натиснути кнопку «Змінити» 4. Змінити назву курсу. 5. Натиснути кнопку «Зберегти».	Успішне редагування курсу
A.1.3	1. У пошуку знайти потрібний курс. 2. Натиснути на сторінку курсу. 3. Натиснути кнопку «Видалити».	Успішне видалення курсу
<i>А.2 Адміністратор може переглядати загальну статистику по усій платформі: активності користувачів, кількість і зареєстрованих користувачів, кількість придбаних курсів, кількість відвідувань тощо.</i>		
A.2.1	1. Перейти до секції статистики. 2. Переглянути загальну кількість користувачів.	Коректне відображення статистичних даних
<i>А.3 Адміністратор може змінювати користувачів</i>		
A.3.1	1. Перейти до налаштувань адміністратора. 2. Обрати розділ користувачів. 3. Знайти потрібного користувача 4. Змінити його логін 5. Натиснути кнопку «Змінити»	Успішна зміна облікового запису користувача
<i>А.4 Можливість зберігати та завантажувати інформацію про курси та користувачів у форматі JSON для архівування або резервного копіювання</i>		
A.4.1	6. Перейти до налаштувань адміністратора. 7. Обрати опцію «Архівувати». 8. Задати назву файлу 9. Підтвердити дію.	Успішне архівування інформації
A.4.2	10. Перейти до налаштувань адміністратора. 11. Обрати опцію «Завантажити архів». 12. Задати назву файлу 13. Підтвердити дію.	Успішне завантаження інформації

1.5.2 Функціональні тести для ролі користувача «Зареєстрований користувач»

На основі опису предметної області, а також функціональних вимог і екранних форм були розроблені функціональні тести для ролі користувача «Зареєстрований користувач» (див. табл. 1.6).

Таблиця 1.6 – Функціональні тести для ролі користувача «Зареєстрований користувач»

Ідентифікатор тесту	Послідовність дій користувача	Очікуваний результат
<i>ЗК.1. Зареєстрований користувач може авторизуватися</i>		
ЗК.1.1	1. Ввести правильний логін. 2. Ввести правильний пароль 3. Натиснути кнопку "Увійти".	Успішна авторизація користувача
ЗК.1.2	1. Ввести правильний логін. 2. Ввести неправильний пароль 3. Натиснути кнопку "Увійти".	Помилка: Невірний пароль
<i>ЗК.2 Зареєстрований користувач може переглядати курси.</i>		
ЗК.2.1	1. Клікнути на потрібний курс	Перегляд інформації курсу
<i>ЗК.3. Зареєстрований користувач може шукати курси.</i>		
ЗК.3.1	1. Перейти на сторінку пошуку 2. Ввести ключове слово у поле пошуку.	Успішне виведення результатів пошуку відповідно до введенного ключового слова.
<i>ЗК.4. Зареєстрований користувач може покупати курси</i>		
ЗК.4.1	1. Перейти на сторінку курсу 2. Натиснути на кнопку "Зареєструватися". 3. Підтвердити покупку курсу	Успішне придбання курсу
<i>ЗК.5. Зареєстрований користувач може моніторити власний прогрес проходження курсів</i>		
ЗК.5.1	1. Перейти до "Власного навчання" 2. Перейти до курсу, що проходиться 3. Натиснути кнопку "Прогрес"	Успішне вивід свого прогресу курсу
<i>ЗК.6. Зареєстрований користувач має особистотий профіль, який він може змінювати</i>		
ЗК.6.1	1. Перейти до особистого профілю. 2. Змінити своє ім'я 3. Натиснути кнопку "Зберегти".	Успішне оновлення інформації у профілі користувача.
<i>ЗК.7. Зареєстрований користувач може оцінювати курси.</i>		
ЗК.7.1	1. Перейти на потрібну сторінку курсу 2. Натиснути кнопку "Поставити рейтинг" 3. Оцінити курс за 10-бальною системою 4. Натиснути кнопку "Підтвердити"	Успішне оцінювання курсу

1.5.3 Функціональні тести для ролі користувача «Гість»

На основі опису предметної області, а також функціональних вимог і екранних форм були розроблені функціональні тести для ролі користувача «Гість» (див. табл. 1.7).

Таблиця 1.7 – Функціональні тести для ролі користувача «Гість»

Ідентифікатор тесту	Послідовність дій користувача	Очікуваний результат
<i>Г.1.Гість може зареєструватися, вказавши логін і пароль</i>		
Г.1.1	<ol style="list-style-type: none"> 1. Ввести унікальний логін (не збігається з існуючими) довжиною не менше п'яти символів. 2. Ввести пароль довжиною не менше п'яти символів. 3. Ввести ім'я. 4. Ввести прізвище 5. Натиснути кнопку "Зареєструватися". 	Успішна реєстрація гостя з новим.
<i>Г.2. Гість може переглядати курси</i>		
Г.2.1	<ol style="list-style-type: none"> 2. Клікнути на потрібний курс 	Перегляд інформації курсу
<i>Г.3.Гість може шукати курси</i>		
Г.3.1	<ol style="list-style-type: none"> 3. Перейти на сторінку пошуку 4. Ввести ключове слово у поле пошуку. 	Успішне виведення результатів пошуку відповідно до введенного ключового слова.

1.5.4 Функціональні тести для ролі користувача «Викладач»

На основі опису предметної області, а також функціональних вимог і екранних форм були розроблені функціональні тести для ролі користувача «Викладач» (див. табл. 1.8).

Таблиця 1.7 – Функціональні тести для ролі користувача «Викладач»

Ідентифікатор тесту	Послідовність дій користувача	Очікуваний результат
<i>В.1. Викладач може зареєструватися, вказавши логін і пароль.</i>		
В.1.1	<ol style="list-style-type: none"> 1. Ввести унікальний логін (не збігається з існуючими) довжиною не менше п'яти символів. 2. Ввести пароль довжиною не менше п'яти символів. 3. Ввести ім'я. 4. Ввести прізвище 5. Натиснути кнопку "Зареєструватися". 	Успішна реєстрація викладача з новим логіном і паролем.
<i>В.2. Викладач може створити та редагувати свої курси</i>		
В.2.1	<ol style="list-style-type: none"> 1. Увійти до особистого кабінету викладача. 2. Обрати опцію "Створити новий курс". 3. Заповнити всі необхідні поля для курсу. 4. Натиснути кнопку "Зберегти зміни" 	Успішне створення курсу викладачем
В.2.2	<ol style="list-style-type: none"> 1. Увійти до особистого кабінету викладача. 2. Вибрати потрібний курс та клікнути на нього. 	Успішне редагування курсу викладачем

	3. Змінити ім'я курсу 4. Натиснути кнопку "Змінити"	
<i>В.3. Викладач може переглядати учасників курсів та їх прогрес проходження курсу</i>		
В.3.1	1. Перейти до списку власних курсів. 2. Вибрати потрібний курс..	Список учасників курсу та їх прогрес.
<i>В.4. Можливість зберігати свої курси та інформацію про учасників у форматі JSON</i>		
В.4.1	1. Увійти до особистого кабінету викладача. 2. Обрати опцію "Зберегти курс". 3. Підтвердити дію.	Успішне збереження даних у форматі JSON

2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

2.1 Виділення і опис класів предметної області

Абстрактний клас «User» – користувач

Таблиця 2.1 – Виділення характеристик класу «User»

Назва характеристик	Тип значення	Обмеження
Логін	Рядок	Не пустий рядок
Пароль	Рядок	Не пустий рядок
Ім'я	Рядок	Не пустий рядок, довжина: хоча б 1 символ
Прізвище	Рядок	Не пустий рядок, довжина: хоча б 1 символ
Повідомлення	Масив рядків	
Статичний лічильник	Ціле число	Не менше нуля
Змінити пароль	Подія	

Клас «Authorised» – авторизований користувач

Таблиця 2.2 – Виділення характеристик класу «Authorised»

Назва характеристик	Тип значення	Обмеження
Придбані курси	Course[]	

Таблиця 2.3 – Виділення поведінки класу «Authorised»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Відкрити свої придбані курси	void		

Клас «Teacher» – вчитаель

Таблиця 2.4 – Виділення характеристик класу «Teacher»

Назва характеристик	Тип значення	Обмеження
Створені курси	Ціле число	Не менше нуля

Таблиця 2.5– Виділення поведінки класу «Teacher»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Створити курс	Void	True/False	
Видалити свій курс	ідентифікатор	True/False	

Клас «Admin» – адміністратор

Таблиця 2.6 – Виділення характеристик класу «Admin»

Назва характеристики	Тип значення	Обмеження
Створені курси	Ціле число	Не менше нуля

Таблиця 2.7 – Виділення поведінки класу «Admin»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Подивитися статистику	Void	Void	
Створити користувача	Void	True/False	
Видалити користувача	ідентифікатор	True/False	
Створити курс	Void	True/False	
Видалити свій курс	ідентифікатор	True/False	

Клас «Course» – курс

Таблиця 2.8 – Виділення характеристик класу «Course»

Назва характеристики	Тип значення	Обмеження
Ідентифікатор	Ціле число	Не менше нуля
Назва курсу	Рядок	Не пустий рядок
Опис курсу	Рядок	Не пустий рядок
Ціна	Гроші	
Чи виконаний курс	Логічне	
Модулів виконано	Ціле число	
Модулі	Масив модулів	Більше нуля
Теги	Масив енам теги	Більше нуля
Творець курсу	User	Не може бути null

Статичний лічильник	Ціле число	Не менше нуля
---------------------	------------	---------------

Таблиця 2.9 – Виділення поведінки класу «Course»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Розпочати N модуль	Номер модуля	void	...
Купити курс	Void	True/False	...

Клас «Module» – модуль курсу

Таблиця 2.10 – Виділення характеристик класу «Module»

Назва характеристики	Тип значення	Обмеження
Назва модуля	Рядок	Не пустий рядок
Опис модуля	Рядок	Не пустий рядок
Кількість виконаних уроків	Ціле число	Не менше 0
Чи виконаний модуль	Логічне	
Уроки	Масив уроків	
Статичний лічильник	Ціле число	Не менше 0

Таблиця 2.11 – Виділення поведінки класу «Module»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Почати урок №N	Номер уроку	Lesson, що є у масиві уроків	

Клас «Lesson» – урок

Таблиця 2.12 – Виділення характеристик класу «Lesson»

Назва характеристики	Тип значення	Обмеження
Назва уроку	Рядок	Не пустий рядок
Матеріали	Рядок	Не пустий рядок
Чи виконаний урок	Логічне	
Статичний лічильник	Ціле число	Не менше 0

Таблиця 2.13 – Виділення поведінки класу «Lesson»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Виконати урок	Void	Void	
Завершити урок	Void	Void	

Клас «CoursesApp» – додаток курсів

Таблиця 2.14 – Виділення характеристик класу «CoursesApp»

Назва характеристики	Тип значення	Обмеження
Поточний користувач	User	Не може бути null

Таблиця 2.15 – Виділення поведінки класу «CoursesApp»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Пошук курсів	Ввід користувача	Course[]	
Пошук курсів	Теги	Course[]	
Завантажити курс	Курс	Void	

Інтерфейс «IStudyable» – вміння вчитися

Таблиця 2.16 – Виділення поведінки інтерфейсу «IStudyable»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Відкрити свої курси	Void	int[]	

Інтерфейс «ITeachable» – вміння навчати

Таблиця 2.17 – Виділення поведінки інтерфейсу «ITeachable»

Назва поведінки	Опис вхідних параметрів	Тип значення, що повертається	Примітка
Створити курс	Void	Void	
Видалити курс	Void	Void	

2.2 Встановлення зв'язків між класами

На рисунку 2.1 наведена діаграма класів, яка була побудована з використанням сервісу <https://www.lucidchart.com>.

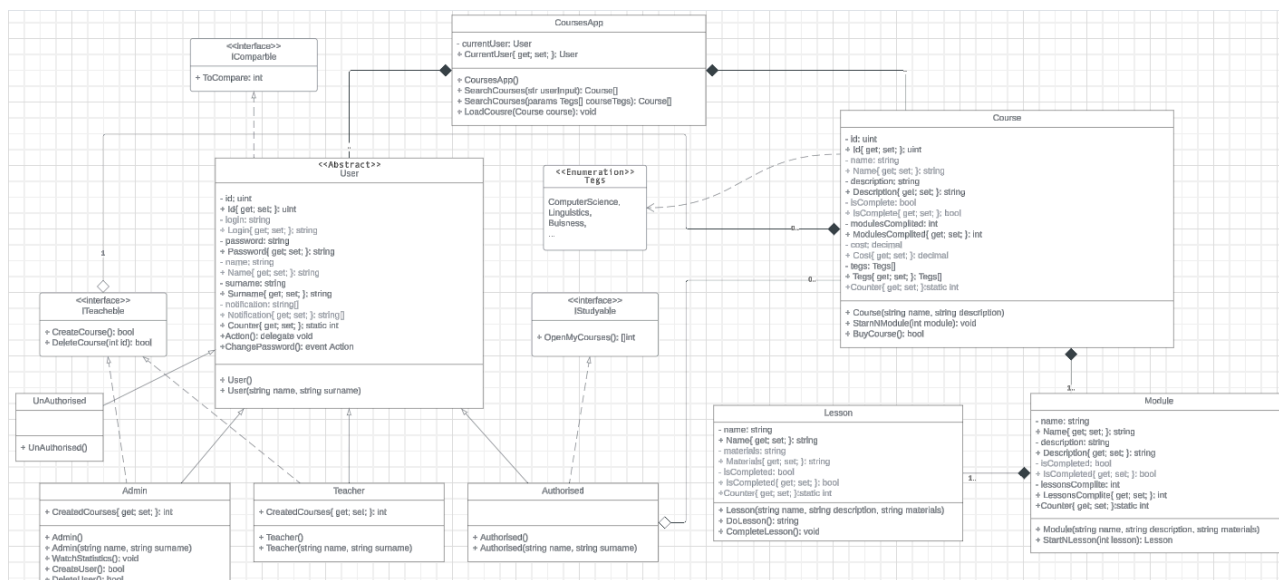


Рисунок 2.1 – Діаграма класів предметної області “Платні навчальні курси”

Всю діаграму можна умовно розділити на “користувачі” та “курси”.

Головний клас у розділі “користувачі” - абстрактний клас User (користувач): він реалізує вбудований інтерфейс IComparable.

Від абстрактного класу наслідуються чотири класи: UnAuthorised, Authorised, Teacher та Admin.

- UnAuthorised - найпростіший, використовується перед тим, як користувач залогиниться/авторизується. Не має якогось явного функціоналу.
- Authorised - звичайний користувач. Може купувати курси та проходити їх. Реалізує інтерфейс IStudyable.
- Teacher - вчитель. Відрізняється від звичайного користувача тим, що не може проходити курси. Реалізує інтерфейс ITeacheble.
- Admin - адміністратор. Також може створювати курси, але має більш обширний функціонал: може дивитися статистику курсів та створювати користувачів. Реалізує інтерфейс ITeacheble.

У розділі курси основний класс - Course (курс). Він має багато характеристик, одна з яких множина класу Module (модуль). У свою чергу один модуль містить у собі множину класу Lesson (урок, заняття).

Ці два розділи об'єднує у собі клас CoursesApp (клас застосунку).

Поміж класів є інтерфейси `IStudyable` та `ITeachable`:

- IStudyable містить функціонал навчання.
- ITeacheble містить функціонал створення, редагування та моніторинг створених курсів

3 ПРОГРАМНА РЕАЛІЗАЦІЯ КЛАСІВ ПРЕДМЕТНОЇ ОБЛАСТІ І ЇХ ТЕСТУВАННЯ

3.1 Структура проєкту з реалізацією класів предметної області

Навести файлову/модульну структуру проєкту, яка містить реалізацію класів предметної області.

Рисунок 3.1 – Структура проєкту з реалізацією предметної області

3.2 Реалізація інтерфейсів і каркасів класів предметної області

Інтерфейс «...» – користувач

Навести програмний код інтерфейсу.

Клас «...» – зареєстрований користувач

Навести програмний код каркасу класу.

Клас «...» – адміністратор

Навести програмний код каркасу класу.

3.3 Розроблення unit-тестів для класів предметної області

Розроблені unit-тести наведені у Додатку .

Таблиця 3.1 – Покриття unit-тестами класу «User»

Назва методу	Кількість розроблених unit-тестів

Таблиця 3.2 – Покриття unit-тестами класу «Movie»

Назва методу	Кількість розроблених unit-тестів

...

3.4 Повна реалізація класів предметної області

У Додатку наведено повну реалізацію програмного коду розроблених класів предметної області.

3.5 Результати unit-тестування класів предметної області

На рисунках 0.5 – 0.7 наведено скриншоти проходження розроблених unit-тестів для класів предметної області.

Рисунок 3.2 – Скриншот проходження unit-тестів для класу

Рисунок 3.3 – Скриншот проходження unit-тестів для класу

Рисунок 3.4 – Скриншот проходження unit-тестів для класу

4 ПРОЄКТУВАННЯ І ПРОГРАМНА РЕАЛІЗАЦІЯ КЛАСІВ ІНТЕРФЕЙСУ КОРИСТУВАЧА

4.1 Структура проєкту з реалізацією класів інтерфейсу користувача

Навести файлову/модульну структуру проєкту, яка містить реалізацію класів інтерфейсу користувача.

Рисунок 4.1 – Структура проєкту з інтерфейсом користувача

4.2 Виділення класів для реалізації інтерфейсу користувача

Описати виділені характеристики і методи для класів інтерфейсу користувача.

...

4.3 Програмна реалізація класів інтерфейсу користувача

У Додатку [] наведено повну реалізацію програмного коду розроблених класів інтерфейсу користувача.

5 ФУНКЦІОНАЛЬНЕ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

5.1 Функціональне тестування для ролі користувача «Адміністратор»

Для проведення функціонального тестування для ролі «Адміністратор» були використані розроблені у п. 1.4.1. функціональні тести. Результати проведення функціонального тестування наведені нижче.

Ідентифікатор тесту: A.1.1.

Вхідні дані:

1. Ввести дату сеансу: «20.01.2023».
2. Ввести час сеансу: «10:00».
3. Обрати зал у кінотеатрі: «2».
4. Обрати фільм: «Аватар: шлях води».
5. Натиснути кнопку «Додати».

Очікуваний результат: Успішне додавання сеансу

Отриманий результат:

Рисунок 5.1 – Скриншот виконання тесту A.1.1

Ідентифікатор тесту:

Вхідні дані:

Очікуваний результат:

Отриманий результат:

Рисунок 5.2 – Скриншот виконання тесту _____

5.2 Функціональне тестування для ролі користувача «Зареєстрований користувач»

Для проведення функціонального тестування для ролі «Зареєстрований користувач» були використані розроблені у п. 1.4.2. функціональні тести. Результати проведення функціонального тестування наведені нижче.

Ідентифікатор тесту:

Вхідні дані:

Очікуваний результат:

Отриманий результат:

Рисунок 5.3 – Скриншот виконання тесту _____

5.3 Функціональне тестування для ролі користувача «Гість»

Для проведення функціонального тестування для ролі «Гість» були використані розроблені у п. 1.4.3. функціональні тести. Результати проведення функціонального тестування наведені нижче.

Ідентифікатор тесту:

Вхідні дані:

Очікуваний результат:

Отриманий результат:

Рисунок 5.4 – Скриншот виконання тесту _____

ВИСНОВКИ

У результаті виконання курсової роботи було розроблено програмне забезпечення з використанням об'єктно-орієнтованої парадигми для вказати предметну область.

В ході роботи було.... (коротко описати що саме було зроблено під час виконання курсової роботи).

...
....

Примітка:

Посилання на github-репозиторій: _____

Посилання на відеоролик: _____

ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАНЬ

1. Воробйов, Ю. А. Правила оформлення навчальних і науково-дослідних документів [Текст] : навч. посіб. / Ю. А. Воробйов, Ю. О. Сисоєв. – 4-те вид., випр. і доп. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2019. – 88 с.
2. Автоматичне оформлення джерел по ВАК України [Електронний ресурс] – Режим доступу до ресурсу: <http://vak.in.ua/do.php>.
3. Назва [Електронний ресурс] – Режим доступу до ресурсу: <http://посилання>.
4. ...

На всі наведені у цьому переліку публікації у тексті пояснювальної записці мають бути посилання.

ДОДАТОК А. Лістинг класів предметної області

Власноруч створений код програми.

ДОДАТОК Б. Лістинг класів інтерфейсу користувача

Власноруч створений код програми. Код, який було автоматично створено середовищем розробки, додавати не потрібно.

ДОДАТОК В. Лістинг класів unit-тестів

Власноруч створений код програми.



ДОДАТОК Г. Назва додатку