

Міністерство освіти і науки України

Відокремлений структурний підрозділ «Тернопільський фаховий коледж
Тернопільського національного технічного університету імені Івана Пулюя»

(повне найменування вищого навчального закладу)

Відділення телекомунікаційних та електронних систем

(назва відділення)

Циклова комісія комп'ютерних наук

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

фахового молодшого бакалавра

(освітньо-професійного ступеня)

на тему: “Розробка інтернет-магазину вживаних книг «BookSwar»”

Виконав: студентка IV курсу, групи КН-423

Спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)

Олексій БАЛАБАН

(ім'я та прізвище)

Керівник Володимир СЕРБІН

(ім'я та прізвище)

Рецензент _____
(ім'я та прізвище)

Тернопіль – 2026

ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ТЕРНОПІЛЬСЬКИЙ ФАХОВИЙ КОЛЕДЖ
ТЕРНОПІЛЬСЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ
ІМЕНІ ІВАНА ПУЛЮЯ»

Відділення телекомунікацій та електронних систем
Циклова комісія комп'ютерних наук
Освітньо-професійний ступінь «фаховий молодший бакалавр»
Спеціальність 122 Комп'ютерні науки
Галузь знань 12 Інформаційні технології

ЗАТВЕРДЖУЮ

Голова циклової комісії
комп'ютерних наук

_____ Галина МАРЦІЯШ

« 02 » березня 2026 року

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТЦІ**

_____ Балабану Олексію Ігоровичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка інтернет-магазину вживаних книг «BookSwap»

керівник роботи _____ Сербін Володимир Сергійович _____,
(прізвище, ім'я, по батькові)

затверджені наказом вищого навчального закладу № 4/9-132 від 27.02.2026 р.

2. Строк подання студенту роботи: 19.06.2026 р.

3. Вихідні дані до роботи: технічне завдання на розробку програмного забезпечення, мови програмування: TypeScript; фреймворк Next.js (App Router), бібліотеки: Prisma (ORM), Tailwind CSS, shadcn/ui, NextAuth.js, Bun. стандарти IEEE 29148-2018, IEEE 29119, ДСТУ 8302:2015.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

- 1 Загальний розділ
 - 1.1 Аналітичний огляд існуючих рішень
 - 1.2 Технічне завдання
 - 1.2.1 Найменування та область застосування
 - 1.2.2 Призначення розробки
 - 1.2.3 Вимоги до функціоналу вебзастосування
 - 1.2.4 Вимоги до програмної документації
 - 1.2.5 Техніко-економічні показники
 - 1.2.6 Стадії та етапи розробки
 - 1.2.7 Порядок тестування та прийому

- 2 Розробка технічного та робочого проєкту
 - 2.1 Розробка структури інтернет-магазину і вебсторінок
 - 2.2 Створення та верстка сторінок інтернет-магазину
 - 2.3 Розробка структури бази даних інтернет-магазину
 - 2.4 Програмування інтернет-магазину
 - 2.4.1 Написання клієнтської частини
 - 2.4.2 Написання серверної частини
 - 2.5 Тестування інтернет-магазину
- 3 Спеціальний розділ
 - 3.1 Інструкція з розгортання інтернет-магазину
 - 3.2 Інструкція з обслуговування та наповнення інтернет-магазину
 - 3.3 Інструкція з популяризації та підтримки інтернет-магазину
- 4 Економічний розділ
 - 4.1 Визначення стадій технологічного процесу та загальної тривалості

проведення розробки інтернет-магазину

- 4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи
- 4.3 Розрахунок витрат на електроенергію
- 4.4 Розрахунок суми амортизаційних відрахувань
- 4.5 Обчислення накладних витрат
- 4.6 Складання кошторису витрат та визначення собівартості інтернет-магазину
- 4.7 Розрахунок ціни інтернет-магазину
- 4.8 Визначення економічної ефективності і терміну окупності капітальних

вкладень

- 5 Охорона праці, техніка безпеки та екологічні вимоги
 - 5.1 Перевірка знань працівників з питань охорони праці.
 - 5.2 Оптимальні кліматичні умови. Таблиця.
- 6 Висновки

Додаткові вказівки: виконання кваліфікаційної роботи із розробкою програмного продукту – інтернет-магазину.

5. Перелік графічного матеріалу:

1. Схема структурна клієнтської частини
2. UML-діаграма варіантів використання
3. ER-діаграма бази даних інтернет магазину
4. Таблиця техніко-економічних показників

6. Консультанти розділів роботи

Розділ	Ім'я, прізвище та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний розділ	Любов КАЛУШКА		
Охорона праці, техніка безпеки та екологічні вимоги	Генадій ГОРЯЧЕК		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Отримання і аналіз технічного завдання	20.03.2026	
2	Збір і узагальнення інформації	01.05.2026	
3	Написання першого розділу	15.05.2026	
4	Розробка технічного та робочого проекту	29.05.2026	
5	Написання спеціального розділу	05.06.2026	
6	Розрахунок економічної частини	08.06.2026	
7	Написання розділу охорони праці	09.06.2026	
8	Виконання графічної частини	10.06.2026	
9	Оформлення пояснювальної записки	11.06.2026	
10	Погодження нормоконтролю	12.06.2026	
11	Попередній захист кваліфікаційної роботи	.06.2026	
12	Захист кваліфікаційної роботи	.06.2026	

7. Дата видачі завдання: 05 березня 2026 р.

Студент

_____ (підпис)

Олексій БАЛАБАН

Керівник роботи

_____ (підпис)

Володимир СЕРБІН

ЗМІСТ

АНОТАЦІЯ.....	7
ABSTRACT.....	8
ВСТУП.....	9
1 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ.....	11
1.1 Аналітичний огляд існуючих рішень.....	11
1.2 Технічне завдання.....	15
1.2.1 Найменування та область застосування.....	15
1.2.2 Призначення розробки.....	16
1.2.3 Вимоги до функціоналу інтернет-магазину.....	16
1.2.4 Вимоги до програмної документації.....	18
1.2.5 Техніко-економічні показники.....	18
1.2.6 Стадії та етапи розробки.....	19
1.2.7 Порядок контролю та прийому.....	20
2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ.....	21
2.1 Розробка структури інтернет-магазину і web-сторінок.....	21
2.2 Створення та верстка сторінок інтернет-магазину.....	26
2.3 Розробка структури бази даних інтернет-магазину.....	37
2.4 Програмування інтернет-магазину.....	40
2.4.1 Написання клієнтської частини.....	42
2.4.2 Написання admin-частини.....	45
2.5 Тестування інтернет-магазину.....	47
3 СПЕЦІАЛЬНИЙ РОЗДІЛ.....	56
3.1 Інструкція з розміщення інтернет-магазину в Інтернеті.....	56
3.2 Інструкція з обслуговування та наповнення інтернет-магазину.....	59
3.3 Інструкція з популяризації та підтримки інтернет-магазину.....	61

					<i>2026.КВР.122.423.01.00.00 ПЗ</i>		
Зм.	Арк.	№ докум.	Підпис	Дата			
Розроб.		<i>Балабан О.І.</i>			Літ.	Арк.	Аркуші
Перевір.		<i>Сердін В.С.</i>				5	
Реценз.					<i>ВСП ТФК ТНТУ КН-423 м. Тернопіль</i>		
Н. Контр.		<i>Приймак В.А.</i>					
Затверд.							

**Розробка інтернет-магазину
вживаних книг «BookSwar»
Пояснювальна записка**

4	ЕКОНОМІЧНИЙ РОЗДІЛ.....	64
4.1	Визначення стадій технологічного процесу та загальної тривалості проведення розробки інтернет-магазину.....	64
4.2	Визначення витрат на оплату праці та відрахувань на соціальні заходи 65	
4.3	Розрахунок витрат на електроенергію	67
4.4	Розрахунок суми амортизаційних відрахувань інтернет-магазину	68
4.5	Обчислення накладних витрат	68
4.6	Складання кошторису витрат та визначення собівартості	69
4.7	Розрахунок ціни інтернет-магазину	70
4.8	Визначення економічної ефективності і терміну окупності капітальних вкладень	70
5	ОХОРОНА ПРАЦІ, ТЕХНІКА БЕЗПЕКИ ТА ЕКОЛОГІЧНІ ВИМОГИ	72
5.1	Перевірка знань працівників з питань охорони праці	72
5.2	Оптимальні кліматичні умови	73
	ВИСНОВКИ.....	74
	ПЕРЕЛІК ПОСИЛАНЬ	76
	ДОДАТКИ.....	78
	Додаток А. Лістинг файлу «app/layout.tsx»	78
	Додаток Б. Лістинг файлу «app/auth/login/page.tsx».....	79
	Додаток В. Лістинг файлу «book-card.tsx»	82
	Додаток Г. Лістинг файлу «actions/book.ts»	87

АНОТАЦІЯ

Тема кваліфікаційної роботи: Розробка інтернет-магазину вживаних книг «BookSwap»

Головною ціллю роботи є створення зручного програмного продукту для організації процесу вторинного використання літератури та побудови прямої взаємодії між читачами.

Пояснювальна записка містить п'ять окремих розділів.

У першому розділі наведено аналітичний огляд наявних торгових майданчиків та проведено аналіз технічного завдання. Визначено головні недоліки універсальних дощок оголошень та обґрунтовано потребу у створенні окремого нішевого рішення для ринку книг.

Другий розділ розкриває процес безпосереднього створення програми та містить пояснення щодо вибору архітектурних рішень. Описано структуру таблиць бази даних, методи обробки вхідної інформації, наведено алгоритми роботи окремих модулів та етапи тестування написаного коду.

Спеціальна частина пояснює процедуру розгортання сайту на хмарному сервері та містить детальну інструкцію з щоденної експлуатації системи. Текст розділу дає чіткі вказівки для адміністратора щодо керування опублікованим вмістом та обліковими записами відвідувачів магазину.

Розрахунок витрат на написання програми та оцінка фінансової доцільності проєкту наведені в четвертому розділі кваліфікаційної роботи.

В останньому п'ятому розділі розглянуто питання охорони праці та техніки безпеки.

Загальний обсяг пояснювальної записки складає __ сторінки. До складу роботи також входить графічна частина на аркушах формату А3. В неї включено діаграму варіантів використання, структурну схему сайту, діаграму бази даних, таблиця техніко-економічних показників.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

ABSTRACT

Thesis Topic: Development of “BookSwap,” an Online Store for Used Books

The main objective of this thesis is to create a user-friendly software product designed to facilitate the reuse of books and foster direct interaction among readers.

This thesis consists of five separate chapters.

The first section provides an analytical overview of existing trading platforms and analyzes the technical specifications. It identifies the main shortcomings of general-purpose classifieds platforms and justifies the need to create a dedicated niche solution for the book market.

The second section details the process of actually developing the program and explains the choice of architectural solutions. It describes the structure of the database tables, methods for processing input data, and outlines the algorithms for individual modules as well as the stages of testing the written code.

A special section explains the procedure for deploying the website on a cloud server and provides detailed instructions for the day-to-day operation of the system. The text of this section offers clear guidance for the administrator on managing published content and the accounts of store visitors.

The calculation of software development costs and the assessment of the project’s financial feasibility are presented in the fourth chapter of the thesis.

The final, fifth chapter addresses occupational health and safety issues.

The total length of the explanatory note is ___ pages. The thesis also includes a graphical section on A3-sized sheets. It contains a site structure diagram, a database diagram, a use case diagram, and technical and economic indicators.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Сучасний рівень розвитку інформаційних технологій відкриває нові можливості для зручного обміну товарами та послугами між звичайними людьми. Традиційне читання паперових книг залишається популярним дозвіллям серед багатьох верств населення України. Проте вартість нових друкованих видань постійно зростає і стає серйозним фінансовим тягарем для багатьох читачів. Велика кількість прочитаної літератури часто просто припадає пилом на домашніх полицях без подальшого використання. Створення спеціалізованого майданчика для обміну та продажу вживаних книг дозволяє значно знизити витрати людей на придбання потрібної літератури.

Необхідність проведення цієї роботи зумовлена відсутністю зручних нішевих інструментів саме для ринку книг. Універсальні дошки оголошень часто мають занадто складний інтерфейс та перевантажені великою кількістю зайвих товарів. Спеціалізовані старі ресурси зазвичай страждають від застарілого дизайну та повної відсутності адаптації під сучасні мобільні пристрої. Розробка проєкту BookSwar вирішує ці проблеми шляхом створення чистого та функціонального вебдодатка. Використання сучасних засобів програмування дозволяє досягти високої швидкості роботи сайту та зручності для кожного відвідувача.

Головна мета кваліфікаційної роботи полягає у розробці та практичній реалізації сучасної вебплатформи для ефективного книгообміну. Для досягнення цієї мети потрібно успішно виконати наступний перелік важливих технічних завдань. Спершу необхідно провести аналіз аналогічних систем та обрати надійний технологічний стек для написання коду. Потім потрібно спроектувати структуру бази даних та створити зручні макети користувацького інтерфейсу. Реалізація функціоналу каталогу літератури та системи пошуку книг за певними критеріями складає основу програмної частини роботи. Студент також повинен розробити надійні інструменти для авторизації користувачів та управління особистими оголошеннями.

Галузь можливого використання розроблених програмних рішень охоплює

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

освітні заклади та різноманітні читацькі спільноти. Студенти вищих навчальних закладів можуть успішно використовувати платформу для обміну підручниками та навчальними матеріалами. Окремі книжкові клуби отримають зручний інструмент для координації спільного читання та обміну цікавими новинками. Запропоноване програмне рішення також підходить для невеликих крамниць як базова система обліку та продажу товарів через мережу. Використання цієї платформи сприяє розвитку культури повторного використання ресурсів та популяризації читання серед молоді.

Розроблена платформа має великий потенціал для подальшого розширення та додавання нових корисних функцій. Впровадження таких рішень робить літературу більш доступною для кожного громадянина незалежно від його рівня доходу. Побудова довірливих стосунків між продавцями та покупцями стає головним пріоритетом під час роботи над проєктом. Текст пояснювальної записки детально розкриває всі технічні аспекти втілення цієї ідеї в життя. Результати роботи демонструють можливість створення якісного програмного продукту з використанням мінімальних фінансових ресурсів.

Подальші розділи роботи містять детальні розрахунки та описи кожного етапу розробки. Кожен крок проєктування супроводжується поясненнями та обґрунтуваннями обраних методів. Фінальний продукт готовий до реального використання та показує високі результати під час тестування. Таким чином робота має велике практичне значення та відповідає всім сучасним стандартам якості розробки програмного забезпечення. Чистота коду та зрозумілість архітектури роблять підтримку проєкту простою та ефективною. Це дозволяє команді розробників швидко вносити зміни та адаптувати платформу до нових потреб ринку. Кінцевий результат повністю виправдовує витрачений час та зусилля на проведення цього дослідження.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

1 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ

1.1 Аналітичний огляд існуючих рішень

Покупка та обмін книгами через інтернет стає звичним явищем серед читачів. Люди постійно шукають зручні способи отримати нову літературу або позбутися вже прочитаних примірників. Створення спеціалізованого майданчика для таких потреб вирішує багато проблем звичайних користувачів. Під час розробки власного проекту BookSwar дуже корисно проаналізувати схожі платформи. Вони пропонують різні підходи до організації продажу та обміну книгами в мережі.

Популярним варіантом є використання великих маркетплейсів для розміщення оголошень. Платформа OLX дозволяє користувачам продавати та купувати книги у відповідній категорії товарів. Сторінка сайту зображена на рисунку 1.1. Цей сервіс має величезну аудиторію та стабільно працює по всій країні. Люди публікують фотографії товару та додають короткий опис стану кожної книги.

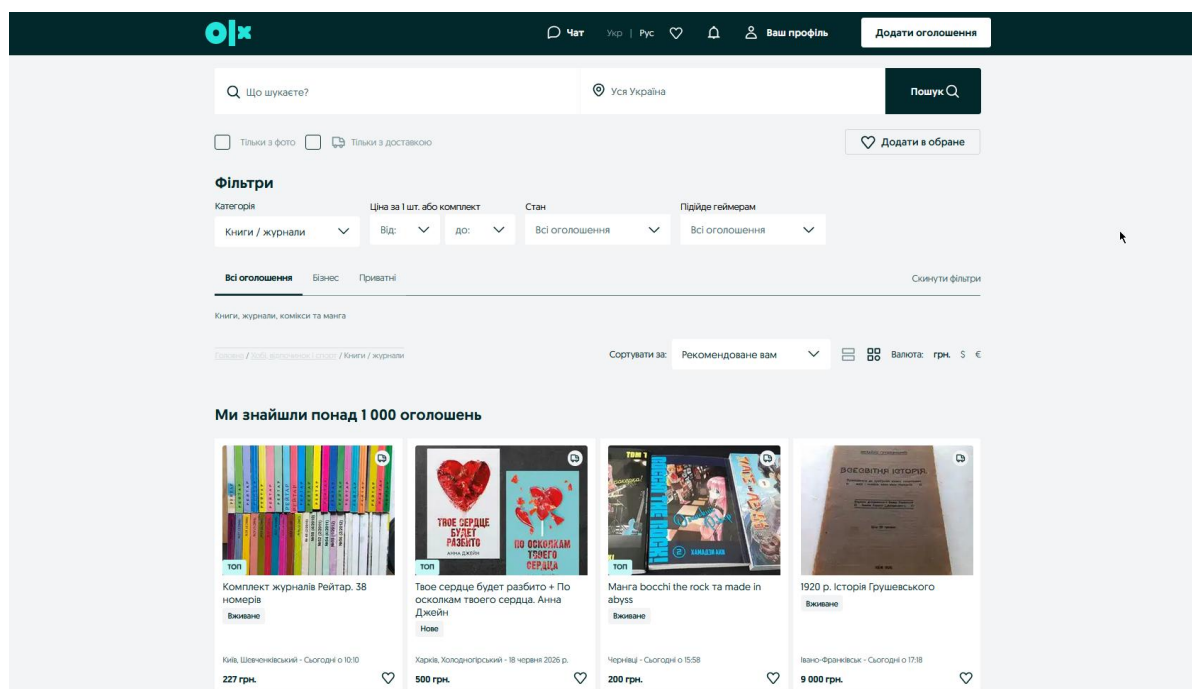


Рисунок 1.1 - Розділ книг на платформі OLX

									Арк.
									11
Зм.	Арк.	№ докум.	Підпис	Дата	2026.KBP.122.423.01.00.00 ПЗ				

Проте великі маркетплейси мають значні недоліки для справжніх любителів читання. Вони не пропонують специфічних інструментів для пошуку за автором або конкретним жанром. Користувачам доводиться дуже довго гортати загальну стрічку оголошень у пошуках потрібного видання. Також повністю відсутній вбудований механізм прямого обміну книгами між двома читачами. Люди змушені самостійно домовлятися про обмін у приватних повідомленнях. Це створює додаткові незручності та певні ризики для обох сторін угоди.

Спеціалізовані платформи для книголюбів існують переважно на західному ринку. Відомим прикладом є старий сервіс BookMooch. Її головна сторінка зображена на рисунку 1.2. Ця система базується на внутрішній економіці балів за відправлені книги. Користувач отримує бали, коли надсилає свою книгу іншій людині звичайною поштою. Потім він витрачає ці зароблені бали на замовлення нових книг у інших учасників спільноти.

The screenshot shows the BookMooch website interface. At the top, there is a search bar containing 'JavaScript' and a 'SEARCH >' button. Below the search bar are navigation buttons: HOME, BROWSE, ABOUT, JOIN, and LOG IN. The main content area displays search results for 'JavaScript'. It includes a filter bar with options like 'SHOW UNMOOCHABLE +', 'AUTHOR >', 'TITLE >', 'TOPIC >', and 'PUBLISHER >'. Below this, there are buttons for 'RELATED SEARCHES >', 'ADVANCED SEARCH >', and 'SHOW COVERS +'. The results are presented in two tables. The first table lists books by author and title, with a 'Copies' column. The second table lists books by topic and title, also with a 'Copies' column. At the bottom, there are two search bars: one for 'BookMooch' and one for 'Amazon', both with 'JavaScript' entered and search buttons.

Title		Copies
David Flanagan	JavaScript: The Definitive Guide	1
Gordon McComb	JavaScript SourceBook - Crie Programas Interativos em JavaScript para a WWW	1
David Flanagan	Javascript Pocket Reference	1
Young Rewired State	Get Coding! Learn HTML, CSS, and JavaScript and Build a Website, App, and Game	1
John Deubert	Extending Acrobat Forms with JavaScript	1

Topic		Copies
Elizabeth Castro	Perl and CGI for the World Wide Web (Visual QuickStart Guide)	2
Patrick Kunert	Tapestry.	1
Eric C. Herrmann	Mastering Perl 5	1
Jim Conallen	Building Web Applications with UML (The Addison-Wesley Object Technology Series)	1
John Sharp	Microsoft Visual C# 2013 Step by Step (Step By Step (Microsoft))	1

Рисунок 1.2 - Головна сторінка сервісу BookMooch

Такий підхід дійсно стимулює людей ділитися своїми бібліотеками з іншими читачами. Але сервіс BookMoosh має застарілий інтерфейс та принципово не підтримує прямі грошові операції. Користувачі не можуть просто купити книгу за гроші, якщо у них на рахунку немає балів. Це жорстко обмежує аудиторію лише тими людьми, які готові постійно відправляти свої книги іншим.

Ще одним популярним прикладом є глобальна платформа Goodreads. Вона чудово підходить для створення особистих списків читання та публікації розгорнутих відгуків. Інтерфейс платформи подано на рисунку 1.3. Люди додають прочитані книги до свого публічного профілю та ставлять їм оцінки.

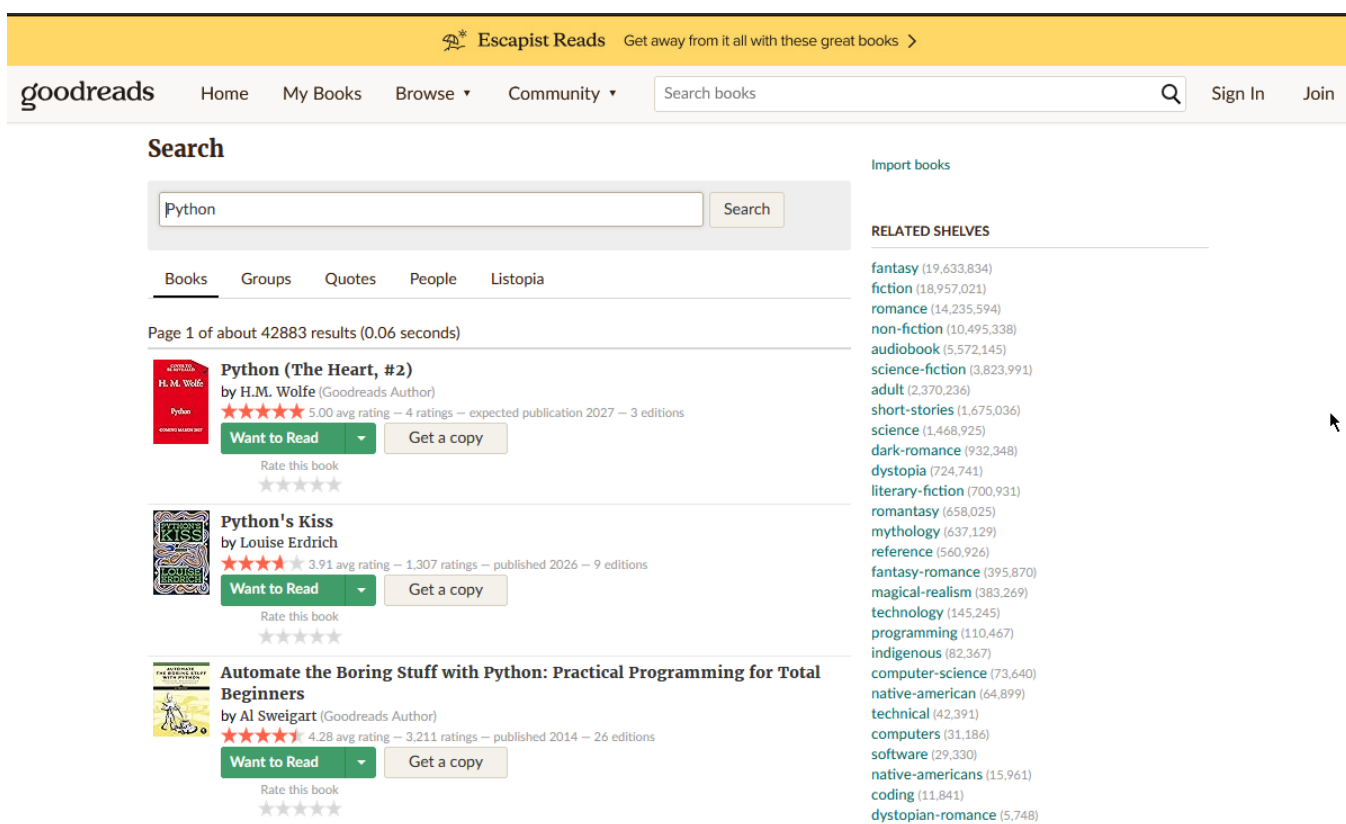


Рисунок 1.3 - Сторінка книги на платформі Goodreads

Але програма Goodreads взагалі не працює як повноцінний магазин чи платформа для обміну. Вона лише дає партнерські посилання на зовнішні магазини для покупки електронних або паперових версій. Користувачі не мають змоги взаємодіяти між собою для фізичного обміну примірниками.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

сайтів та готових шаблонів [1]. Команда розробників отримує повний контроль над дизайном [2] та внутрішньою логікою роботи програми. Власна база даних надійно зберігатиме всю інформацію про користувачів та їхні замовлення. Це дасть змогу без проблем розширювати платформу в майбутньому та додавати нові функції.

1.2 Технічне завдання

1.2.1 Найменування та область застосування

Програмний продукт, що розробляється в межах даної кваліфікаційної роботи, має повну назву «Інтернет-магазину вживаних книг «BookSwar»».

Скорочена назва проєкту для технічної документації та використання в інтерфейсі - BookSwar.

Система являє собою сучасний Інтернет-магазин торгового та соціального призначення. Програма забезпечує організацію процесу купівлі, продажу та прямого обміну прочитаною літературою між зареєстрованими користувачами. Проєкт створює зручне цифрове середовище для вторинного використання паперових видань та популяризації читання через зниження фінансових витрат на придбання нових книг.

Програмний виріб призначений для надання користувачам інструментів пошуку потрібних творів, управління власними оголошеннями та безпечної взаємодії з іншими учасниками платформи. Інтернет-магазин дозволяє продавцям швидко розміщувати інформацію про наявні примірники, а покупцям - легко знаходити літературу за конкретними жанрами, авторами або станом видання.

Область застосування охоплює сферу електронної комерції, букіністичної справи та соціальних сервісів спільного споживання. Інтернет-магазин може успішно використовуватися як самостійний торговий майданчик для звичайних громадян, так і допоміжний інструмент для невеликих приватних книгарень. Проєкт має велике значення для студентських спільнот, де постійно існує попит на обмін навчальною літературою та підручниками.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

Об'єктами використання програмного продукту є персональні комп'ютери, ноутбуки та портативні мобільні пристрої відвідувачів. Для стабільної роботи системи необхідний доступ до глобальної мережі Інтернет та наявність будь-якого сучасного веб-браузера. Платформа працює без встановлення додаткового програмного забезпечення на клієнтські пристрої користувачів.

1.2.2 Призначення розробки

Головна мета створення інтернет-магазину BookSwar полягає у розробці зручної онлайн платформи для читачів. Люди зможуть швидко знаходити цікаві книги та легко пропонувати свої власні примірники для продажу іншим учасникам. Програма має вирішити проблему пошуку вживаної літератури без використання незручних дошок оголошень.

Важливим завданням є надання користувачам зрозумілого інтерфейсу для швидкого пошуку потрібних видань у загальному каталозі. Система повинна мати гнучке фільтрування книг за жанрами, ціною та фізичним станом. Кожне видання вимагає відображення детального опису з кількома фотографіями та повною інформацією про автора книги.

Особлива увага приділяється питанню безпеки. Програма має налагодити безпечне збереження паролів та особистих даних усіх зареєстрованих користувачів. Також потрібно створити надійну систему оформлення замовлень за допомогою віртуального кошика покупок. Для підтримки порядку на платформі розробляється панель адміністрування для перевірки нових оголошень перед їх публікацією. Успішна реалізація цих завдань допоможе створити дійсно корисний інструмент для любителів читання. Платформа BookSwar задовольнить всі базові потреби користувачів та працюватиме максимально стабільно.

1.2.3 Вимоги до функціоналу інтернет-магазину

Розробка платформи BookSwar вимагає дуже чіткого дотримання

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	Арк.
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		16

визначених функціональних вимог. Програма повинна безперебійно виконувати заявлені дії без жодних помилок та системних збоїв. Спочатку створюється структурований каталог книг із логічним розподілом за популярними жанрами. Кожна окрема книга обов'язково має свою сторінку з якісними фотографіями, актуальною ціною, роком видання та точним описом стану.

Звичайні користувачі зможуть додавати вибрані товари до кошика та вільно змінювати їх кількість. Система оформлення замовлення вимагає від покупця введення адреси доставки та вибору зручного способу оплати. Вебсайт підтримує швидку реєстрацію нових користувачів та безпечну авторизацію через електронну пошту. Авторизовані клієнти отримують можливість самостійно додавати власні книги на продаж через спеціальну форму.

Програма також вимагає обов'язкової перевірки нових книг адміністратором перед їх появою в загальному каталозі. Вбудована функція пошуку дозволяє швидко знаходити книги за назвою або прізвищем автора. Інтерфейс вебдодатку автоматично підлаштовується під розміри екранів мобільних телефонів та планшетів. Додатково користувачі зможуть залишати свої відгуки про роботу платформи у спеціальному розділі.

Діаграма варіантів використання інтернет-магазину представлено на плакаті 2026.КВР.122.423.01.00.00 ДВ.

Вхідні дані надходять від користувачів переважно через різні вебформи. Люди самостійно вводять текстову інформацію, вибирають параметри зі списків та завантажуватимуть файли зображень обкладинок. Вихідні дані відображатимуться на сторінках сайту у вигляді зручних карток товарів, таблиць замовлень та зрозумілих текстових повідомлень [3].

Інтернет-магазин повинен блискавично реагувати на всі дії користувача. Час завантаження сторінки каталогу не повинен перевищувати три секунди при нормальній швидкості домашнього інтернету. Пошукові запити мають оброблятися максимально швидко для забезпечення зручності відвідувачів. Система вимагає надійного захисту бази даних від будь-якого стороннього

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

втручання. Паролі користувачів завжди зберігатимуться виключно у зашифрованому вигляді. Доступ до панелі адміністрування матимуть лише спеціальні акаунти з відповідними правами.

1.2.4 Вимоги до програмної документації

Після завершення написання основного програмного коду потрібно обов'язково підготувати необхідну документацію. Вона допоможе іншим розробникам швидко зрозуміти логіку роботи Інтернет-магазину. Документація містить детальне керівництво з розгортання Інтернет-магазину на хостингу в мережі Інтернет. Також вона описує процес налаштування з'єднання з базою даних та хмарним сховищем зображень.

Окремо формується повний перелік усіх потрібних залежностей та бібліотек для успішного запуску проєкту на новому сервері. Додається детальна інструкція для адміністратора сайту щодо правильної модерації книг та перегляду списку замовлень. Важливим пунктом є рекомендації щодо створення регулярних резервних копій бази даних на випадок аварії. Документація також описує структуру бази даних із поясненням всіх зв'язків між таблицями. Для звичайних клієнтів створюється коротка інструкція щодо додавання книг та оформлення нових замовлень.

Самий програмний код має обов'язково містити коментарі до складних функцій та неочевидних компонентів. Коментарі чітко пояснюють призначення функції та очікувані вхідні параметри. Вони мають бути лаконічними та максимально зрозумілими для інших програмістів. Кожна функція повинна завжди повертати передбачуваний результат або генерувати стандартне повідомлення про помилку.

1.2.5 Техніко-економічні показники

Розрахунок витрат на створення нового інтернет-магазину BookSwar

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<i>18</i>

відбуватиметься суворо згідно з правилами складання економічної частини кваліфікаційної роботи. Потрібно об'єктивно оцінити час розробки та вартість оренди всіх сторонніх сервісів. Загальний час на написання програмного коду становитиме близько 170 людино-годин.

Окремо враховуються витрати на оренду продуктивного сервера та доменного імені на перший рік роботи магазину. Також береться до уваги час на детальне проектування інтерфейсу та створення макетів сторінок у графічному редакторі. Частина технікоекономічних показників:

- очікуваний термін повної окупності проєкту складає 2,1 років;
- собівартість – 93201грн;
- ціна - 143 834грн.

Точні суми ретельно розраховуватимуться у відповідному економічному розділі кваліфікаційної роботи. Вони базуватимуться на середніх ринкових цінах послуг хостингу та актуальній вартості робочого часу програміста. Техніко економічні показники представлені на плакаті 2026.КВР.122.423.01.00.00 ТП.

1.2.6 Стадії та етапи розробки

Процес створення нового вебдодатку BookSwar логічно ділиться на окремі етапи. Це дуже допомагає контролювати виконання роботи та вчасно виявляти будь-які проблеми. Спочатку відбувається збір вимог та складання детального плану роботи над усім проєктом. Далі йде проектування структури бази даних для правильного зберігання користувачів та книг.

Наступним кроком є створення візуальних макетів інтерфейсу для різних розмірів екранів пристроїв. Після цього починається написання серверної логіки для обробки запитів та постійної взаємодії з базою даних. Розробка клієнтської частини програми виконується за допомогою фреймворку Next.js [4] на базі бібліотеки React [5]. Окремим етапом йде підключення зовнішнього хмарного сервісу Cloudinary для надійного зберігання фотографій книг.

Потім відбувається об'єднання клієнтської та серверної частин у повністю

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		19

готовий додаток. Проводиться ретельне тестування всіх функцій програми на наявність логічних помилок. Завершальними кроками є розгортання готового вебдодатку на сервері в мережі Інтернет та написання самої пояснювальної записки. Кожен етап вимагає обов'язкової перевірки результатів перед переходом до наступного кроку. Такий підхід завжди гарантує високу якість кінцевого продукту.

1.2.7 Порядок контролю та прийому

Готовий інтернет-магазин BookSwar пройде ретельне тестування перед початком використання. Перевірка підтвердить готовність програми та її здатність працювати без жодних збоїв. Процес тестування починається з ручної перевірки реєстрації нових користувачів та входу в систему. Потім тестується додавання нової тестової книги та завантаження фотографій на сервер.

Далі відбувається перевірка роботи кошика покупок та спроба створення нового тестового замовлення. Тестувальник здійснює вхід під обліковим записом адміністратора та перевіряє механізм схвалення тестової книги. Обов'язково проводиться тестування роботи програми на різних мобільних пристроях та у різних популярних браузерях. Також відбувається вимірювання реальної швидкості завантаження головної сторінки та великого каталогу товарів. Наостанок ретельно перевіряється робота системи фільтрування та пошуку книг за різними ключовими словами.

Після успішного завершення всіх етапів тестування відбудеться офіційний прийом роботи комісією. Демонструється робота готового інтернет-магазину наживо. Комісія перевірить відповідність проекту затвердженому технічному завданню та виставить підсумкову оцінку.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						20
Зм.	Арк.	№ докум.	Підпис	Дата		

2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ

2.1 Розробка структури інтернет-магазину і web-сторінок

Під час визначення головного призначення майбутнього інтернет-магазину було обрано формат спеціалізованої веб-платформи для зручного обміну та швидкого продажу вживаних книг прямо між самими читачами на території всієї нашої країни. Через це веб-проект поєднує функції великої бібліотеки. Для задоволення потреб любителів читання розроблено зручну систему пошуку літератури за назвою чи авторами та створено спеціальний каталог із багатьма популярними групами нових видань у мережі інтернет на нашому сайті. З метою залучення нових відвідувачів на стартовій сторінці необхідно показати списки останніх книжкових надходжень разом із фото обкладинок від продавців. Без реєстрації можна переглядати наявні пропозиції. Завдяки цьому вдасться суттєво спростити первинний контакт нових гостей із вмістом нашої платформи та відразу створити гарні умови для наступної тривалої роботи з меню нашої нової системи магазину.

У процесі аналізу завдання та проектування інтерфейсу було створено повну карту сайту, яку показано на плакаті структурної схеми 2026.КВР.122.423.01.00.00 СС.

Для обґрунтування архітектурних рішень інтерфейсу та вибору паттернів проектування були використані рекомендації щодо розробки програмного забезпечення. Через це інтерфейс має чіткий та логічний вигляд. У межах проекту вся система чітко ділиться на відкриту публічну зону для перегляду та закриту частину для авторизованих користувачів і адміністраторів системи для захисту бази даних у мережі інтернет надовго.

Щодо зовнішнього вигляду нашого ресурсу треба створити простий стиль із вибором світлої та темної тем для зручності читачів у будь-який час доби. Проектування зручного інтерфейсу для користувачів та вибір гармонійної колірної гами виконувалися на основі базових ергономічних правил веб-дизайну.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						21
Зм.	Арк.	№ докум.	Підпис	Дата		

Для текстових блоків планується обрати чіткі сучасні шрифти без зарубок, які забезпечать швидке читання описів книг відвідувачами нашого нового сайту у мережі. Через це дизайн буде виглядати дуже привабливо. Завдяки гнучкій адаптивній верстці сторінки мають однаково зручно відображатися як на великих екранах комп'ютерів, так і на маленьких дисплеях мобільних телефонів користувачів у мережі інтернет. Для створення цілісного уявлення про побудову інтерфейсу нижче наведено детальний опис структури ключових веб-сторінок, які необхідно реалізувати:

1) Гловна сторінка. У верхній частині головної сторінки нашого сайту необхідно реалізувати зручне меню навігації для переходу між усіма основними вікнами готового ресурсу у мережі інтернет без довгих пошуків потрібних кнопок. Під цим меню планується розмістити великий банер для показу нових пропозицій та популярних категорій видань на нашому новому майданчику для покупців книг. Через це вибір книг стане простішим. Для залучення нових читачів у нижній частині вікна має бути розташована сітка останніх книжкових надходжень із фотографіями обкладинок та реальними оцінками від покупців у нашій новій великій системі.

2) Каталог книг. Для швидкого пошуку літератури інтерфейс цієї сторінки має складатися з бічної панелі з різними фільтрами за авторами чи жанрами для зручності відвідувачів нашого нового сайту у мережі інтернет. У правій частині сторінки необхідно реалізувати велику сітку карток книг для показу всіх наявних пропозицій від продавців нашого нового сайту. Усередині кожної картки мають бути такі елементи:

- фотографія обкладинки видання;
- назва книги та ім'я автора;
- стан примірника;
- вартість книги;
- кнопка швидкого перегляду.

Без наявності цих даних пошук книг буде дуже важким для покупців, тому

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		22

реалізація цих елементів є обов'язковою для нашої нової системи магазину у мережі інтернет. Завдяки такому простому рішенню кожен гість зможе швидко знайти найкращу пропозицію для покупки у нашому новому магазині без зайвих зусиль у будь-який зручний час.

3) Сторінка книги. Під час перегляду деталей окремого видання у лівій половині робочого поля має бути розташовано велике фото обкладинки для огляду стану книги покупцями нашого нового сайту у мережі. У правій частині сторінки необхідно реалізувати блок із назвою книги, описом стану примірника, ціною та даними продавця для швидкого контакту з покупцем. На цій сторінці мають бути такі елементи:

- велике фото обкладинки;
- детальний опис книги;
- стан примірника;
- вартість книги;
- кнопка додавання до кошика;
- блок відгуків покупців.

Для здійснення покупки трохи нижче описів планується додати великі та помітні кнопки для відправки книги у кошик та написання відгуку про якість цього конкретного примірника у системі. Завдяки такому продуманому розташуванню елементів кожен покупець зможе швидко зв'язатися з продавцем та оформити покупку без будь-яких додаткових труднощів чи затримок у роботі.

4) Профіль продавця. У верхній зоні сторінки користувача планується розмістити великий банер із фотографією продавця та коротким текстом про його інтереси щодо читання різних книг у нашому новому книжковому веб-сайті магазину. Під цим банером необхідно реалізувати сітку всіх оголошень цього продавця з позначенням статусів наявності чи продажу конкретних книжкових видань у нашій системі. Через це каталог виглядає дуже просто покупцям. Завдяки такому підходу покупці зможуть легко бачити весь асортимент конкретної людини та купувати кілька книжок одночасно без тривалого пошуку по всьому великому

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<i>23</i>

каталогу нашого нового сайту у мережі.

5) Кошик та оформлення замовлення. У лівій частині екрана планується відобразити детальний список вибраних книг із мініатюрами обкладинок, цінами та кнопками вилучення зайвих товарів для покупців нашого нового сайту у мережі інтернет надовго. У правій частині необхідно розмістити зручну форму для швидкого введення даних доставки посилки поштою для майбутніх покупців нашої нової системи. У середині цієї форми планується зробити такі поля:

- список вибраних видань;
- ціна кожної книги;
- поля для імені покупця;
- поле для номера телефону;
- варіанти поштових відділень.

Для завершення покупки покупцям необхідно буде вказати правильний телефон, ім'я та адресу найближчого поштового відділення для надійної доставки книг продавцем без зайвих затримок у системі. Завдяки такому простому рішенню процес покупки стане набагато швидшим для кожного відвідувача нашої нової системи магазину в мережі інтернет на довгі роки роботи.

б) Кабінет продавця. У лівій частині кабінету користувача планується розмістити вертикальне навігаційне меню для переходу між вкладками профілю, оголошень та замовлень для швидкої роботи користувачів нашого нового веб-сайту у мережі інтернет. У центральній зоні має бути виведено список книг продавця з кольоровими індикаторами статусу перевірки кожної книги адміном нашої нової системи. Для додавання книг планується така зручна форма:

- назва книги;
- опис книги;
- стан примірника;
- вартість;
- файл зображення.

Для заповнення цієї форми продавцеві необхідно вказати назву книги, опис

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		24

стану примірника, вартість та завантажити файл зображення обкладинки для швидкої публікації оголошення у системі. Завдяки такому простому підходу продавці зможуть легко керувати своїми оголошеннями та відстежувати статус перевірки книг адміністрацією нашого нового магазину у мережі інтернет.

7) Панель адміністратора. У центральній зоні панелі адміна необхідно створити зручні прості таблиці для модерації нових книг, зміни бази людей та швидкого опису статусів замовлень у нашій новій готовій системі магазину. Під час модерації адмін має отримати можливість схвалювати або відхиляти оголошення прямо з таблиці списку книг для швидкого покращення роботи нашої системи. Без цього контроль буде дуже повільним. Завдяки такому розміщенню основних блоків вікон адмін зможе швидко керувати вмістом сайту та берегти стабільну роботу всього нашого книжкового майданчика у мережі інтернет без будь-яких додаткових затримок часу.

У процесі взаємодії з сайтом гості мають заповнити кілька важливих інтерактивних форм для збору особистих даних та запису своїх дій у нашій новій системі магазину для обміну. Через це вся інформація буде гарантовано збережена. Для створення нового облікового запису користувача треба створити форму реєстрації з повним описом імені та прізвища разом із поштою та паролем для захисту особистого профілю. Без пошти вхід буде відразу повністю заблоковано. З метою швидкого наповнення каталогу книг для продажу продавець має заповнювати форму з текстовим полем назви, повним описом стану, вартістю та модулем для завантаження фотографій обкладинки книг у свій кабінет.

Під час фінального етапу оформлення доставки гість має написати форму з обов'язковим вказанням міста, адреси, телефону та імені людини для швидкого відправлення посилки поштою у будь-який регіон. Без телефону відправка буде повністю скасована поштовою організацією. Для адміністрування бази даних треба створити прості форми зміни категорій та жанрів літератури з метою швидкого оновлення швидких фільтрів нашого нового сайту магазину для читачів. Через це пошук книг стане набагато простішим гостям. Завдяки успішному проектуванню

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						25
Зм.	Арк.	№ докум.	Підпис	Дата		

цих форм користувачі зможуть швидко знаходити потрібні книги, а адмін отримає зручні інструменти для роботи з усіма оголошеннями нашого нового магазину у мережі інтернет без зайвих проблем.

2.2 Створення та верстка сторінок інтернет-магазину

Для швидкої побудови сторінок книжкової платформи вибрано відоме середовище Next.js на базі коду React для прискорення завантаження всіх її сторінок у мережі. У браузері файли коду завантажуються дуже швидко. Для швидкого встановлення коду системи застосовано швидкодіючий інструмент Bun з метою економії часу під час роботи над кожною частиною сайту. Завдяки цьому збірка застосунку завершується швидко.

Після запуску звичайної інсталяційної директиви в робочій папці сама постала тут первинна структура всіх файлів та каталогів, яка зображена на рисунку 2.1. У підсумку проста схема спрощує роботу для програміста. Для швидкої модифікації стилів блоків використано готову базу Tailwind CSS [6] з метою швидкого проектування темної гами та гарного розміщення тут всіх елементів. У робочому файлі опис всіх стилів міститься повністю.

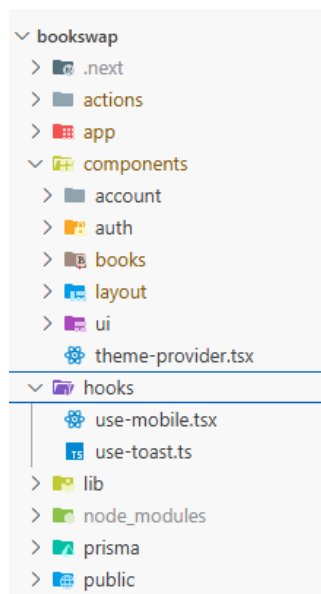


Рисунок 2.1 - Початкова структура каталогів створеного проекту

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

Для побудови загальної оболонки всіх сторінок сайту розроблено простий макет у файлі `app/layout.tsx`, де додано коди шрифтів та базових провайдерів для простого перемикання теми. У додатку А поданий код цього макета. У верхній частині побудованої сторінки сайту розміщено хедер з кнопками для переходу між сторінками та перегляду списку доданих книг у користувацький кошик. На рисунку 2.2 зображено зовнішній вигляд хедера.

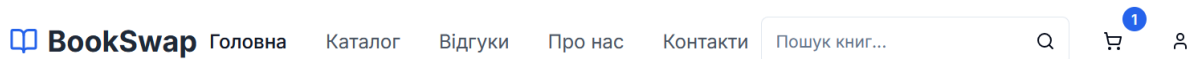


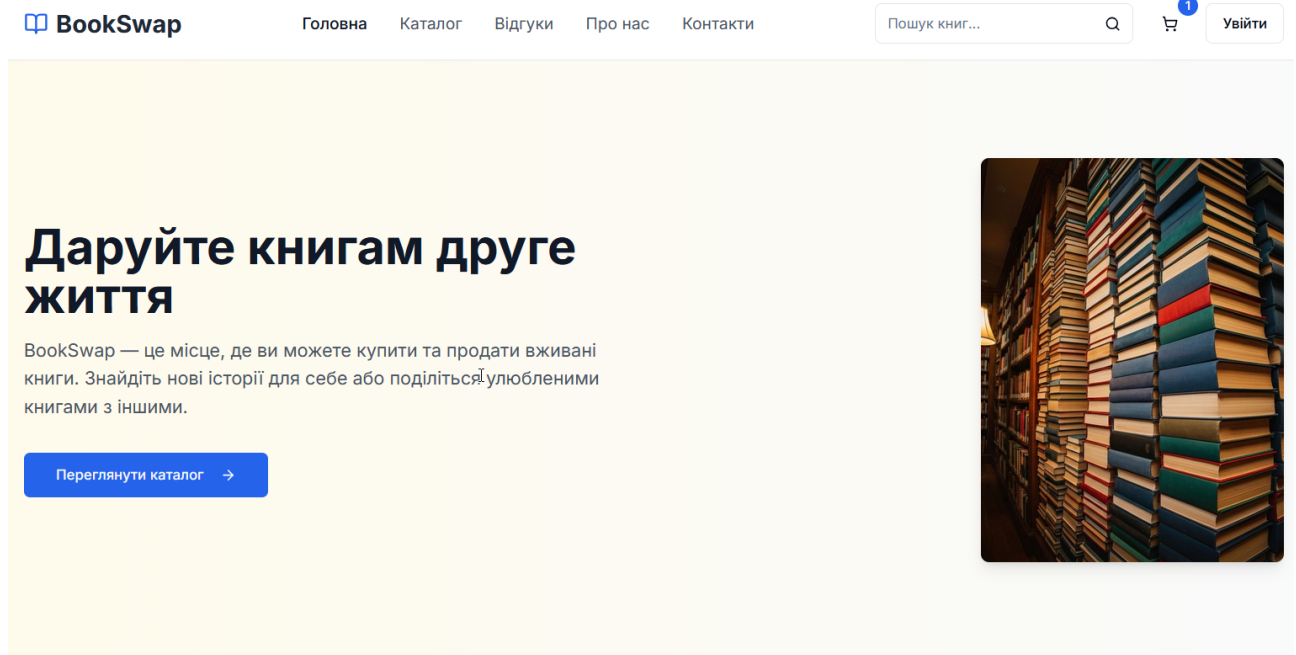
Рисунок 2.2 - Зовнішній вигляд навігаційного меню шапки сайту

Для авторизації та входу користувача створено форму у файлі `app/auth/login/page.tsx` для ідентифікації пошти та пароля під час входу на сайт. У додатку Б подано її повний код. Завдяки підключенню коду валідації на стороні клієнта блок забезпечує автоматичну перевірку форми перед надсиланням будь-яких запитів до бази даних. На рисунку 2.3 зображено форму входу до системи.

The image shows a login form titled 'Вхід до акаунту'. Below the title is the instruction 'Увійдіть, щоб отримати доступ до вашого акаунту'. There are two input fields: 'Email' with the placeholder 'your@email.com' and 'Пароль' with a masked password '.....'. Below the fields is a blue button labeled 'Увійти'. At the bottom, there is a link: 'Немає акаунту? Зареєструватися'.

Рисунок 2.3 - Зовнішній вигляд форми входу в обліковий запис

Для вибору книг розроблена головна сторінка, де розташована велика хіро секція та зручні кнопки категорій для переходу до каталогів різних книг. Тут є кнопки для вибору категорій. На рисунку 2.4 зображено загальний інтерфейс головної сторінки з хіро секцією та кнопками категорій для перегляду тут нових груп доступних книг. При кліці на кнопку.переходід працює швидко.



Категорії книг

Поезія

Драма

Пригоди

Роман

Історичний роман

Повість

Рисунок 2.4 - Інтерфейс головної сторінки з хіро секцією та кнопками категорій

У лівій зоні інтерфейсу розміщена панель фільтрів для сортування тут літературних творів за ціною та жанрами або визначеним станом цієї книги. Поруч побудовано сітку для показу каталогу. На рисунку 2.5 зображено каталог книг з інтерактивною панеллю фільтрів для розумного вибору тут потрібних видань перед відправленням у інтернет-кошик. При кліках фільтри відпрацьовують дуже швидко.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

Каталог книг

Знайдено 5 книг

Фільтри X Скинути

Пошук

Пошук за назвою, автором...

Жанр

Всі жанри

Стан книги

Будь-який стан

Ціна

0 € - 1000 €

0 1000

Сортування

За назвою (А-Я)



Кобзар

Тарас Шевченко
2015 • Поезія

300 €

Деталі

Як нова



Місто

Валер'ян Підмогильний
2021 • Українська література

220 €

Деталі

Як нова



Тигролови

Іван Багряний
2020 • Пригодницький роман

250 €

Деталі

Гарний

Рисунок 2.5 - Каталог книг з бічною панеллю фільтрів

Для гарного показу кожного літературного видання розроблено окрему картку книги з її обкладинкою та ціною і назвою автора та спеціальною позначкою стану. У додатку В подано її код. На рисунку 2.6 зображено тут інтерфейс картки книги з ціною та визначеним станом примірника для перегляду покупцями перед покупкою у книжковому магазині. При показі макет виглядає дуже гарно.



Пантелеймон Куліш

ЧОРНА РАДА

Чорна рада234234 3 потертостями

Пантелеймон Куліш
2018 • Історичний роман

180 €

Деталі

Рисунок 2.6 - Інтерфейс картки книги з ціною та станом

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

При переміщенні на окрему сторінку книги користувач бачить повний опис твору з відгуками інших читачів та формою для написання власної рецензії. Через форми відгуку дані передаються швидко. На рисунку 2.7 зображено сторінку книги з детальним текстовим описом

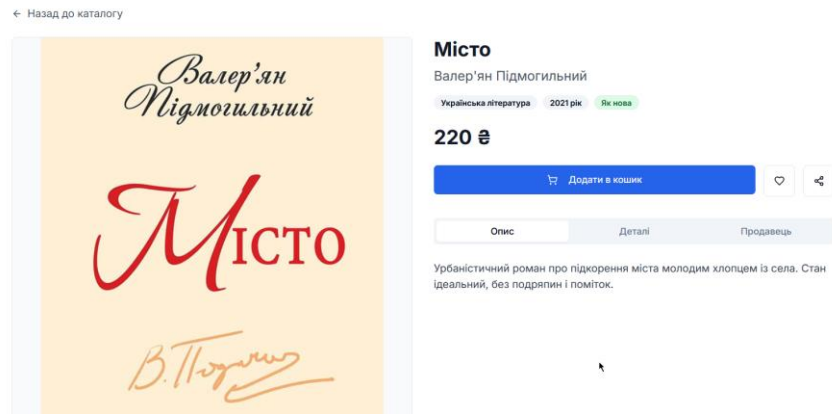


Рисунок 2.7 - Сторінка книги з детальним описом

Для збору книг реалізовано кошик з швидким розрахунком загальної ціни та простою можливістю зміни списку перед придбанням чи обміном книгами. Після вибору товари потрапляють до бази. На рисунку 2.8 зображено тут інтерфейс кошика з повним переліком всіх вибраних видань та зазначенням їхньої вартості перед початком майбутньої оплати людини. Через кошик покупки проходять дуже швидко.

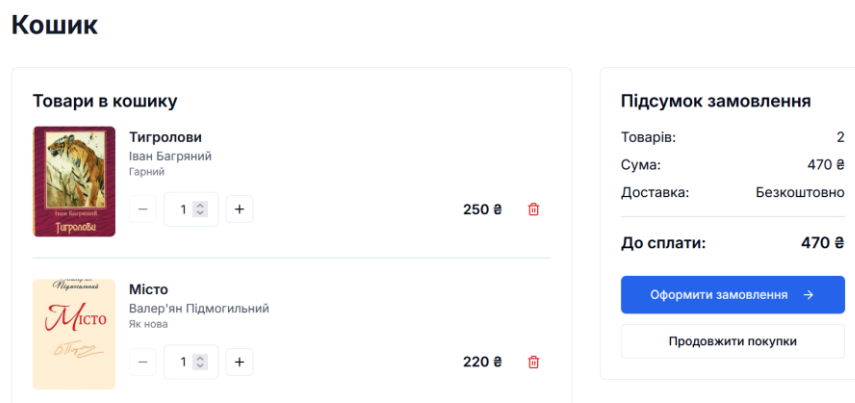


Рисунок 2.8 - Інтерфейс кошика з переліком вибраних видань

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

Після переходу на сторінку оформлення замовлення тут відкривається окрема форма для внесення адреси доставки та вибору способу розрахунку між двома різними користувачами. При цьому замовлення записуються в базу. На рисунку 2.9 зображено сторінку оформлення замовлення з формою доставки та переліком поштових відділень для швидкого отримання книг покупцем у рідному місті. Через сервіс доставка діє дуже швидко.

← Назад до кошика

Оформлення замовлення

Контактна інформація Ім'я * <input type="text" value="Іван"/> Прізвище * <input type="text" value="Користувач"/> Email * <input type="text" value="user@gmail.com"/> Телефон * <input type="text"/>		Ваше замовлення Тигролови x 1 250 ₪ Місто x 1 220 ₪ Сума: 470 ₪ Доставка: 80 ₪ До сплати: 550 ₪ <input type="button" value="Підтвердити замовлення"/>
Доставка Спосіб доставки <input checked="" type="radio"/> Нова Пошта (80 ₪)		

Рисунок 2.9 - Сторінка оформлення замовлення з формою доставки

Для показу власних даних створено сторінку профілю з відображенням електронної пошти та імені користувача. Через паролі профілі захищаються дуже надійно. На рисунку 2.10 зображено сторінку профілю користувача. При переході кабінети відкриваються дуже швидко.

Мій акаунт

Вітаємо, Іван Користувач!

Профіль Керуйте своїми особистими даними Ім'я: Іван Користувач Email: user@gmail.com <input type="button" value="Редагувати профіль"/>	Мої книги Переглядайте та керуйте вашими оголошеннями <input type="button" value="Переглянути книги"/>	+ Додати книгу Створіть нове оголошення про продаж <input type="button" value="Додати книгу"/>
Мої замовлення Історія покупок та поточні замовлення <input type="button" value="Переглянути замовлення"/>		

Рисунок 2.10 - Сторінка профілю користувача з балансом та даними


При переході на сторінку редагування профілю користувач може самостійно змінити своє ім'я або обрати секретний пароль та вказати діючий номер мобільного телефону. Після цього дані оновлюються в базі. На рисунку 2.11 зображено сторінку редагування профілю користувача з формою для оновлення своїх відомостей та запису пароля у новій базі. При цьому зміни зберігаються тут відразу.

Для сторінки редагування профілю передбачено такі поля:

- нове ім'я користувача;
- адреса електронної пошти;
- пароль для захисту кабінету;
- номер мобільного телефону.

Редагувати профіль

Аватар



Особиста інформація

Повне ім'я *	Email *
<input type="text" value="Іван Користувач"/>	<input type="text" value="user@gmail.com"/>
Телефон	
<input type="text" value="+380..."/>	

Адреса

Повна адреса


Рисунок 2.11 - Сторінка редагування профілю користувача з формою

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

Для перегляду всіх транзакцій створено сторінку історії замовлень з повним списком тут придбаних книг та показом їхньої ціни та її поточного статусу. У цій таблиці дані виводяться дуже наочно. На рисунку 2.12 зображено сторінку історії замовлень з переліком покупок та загальною сумою оплати за кожне виставлене продавцем книжкове у цій базі. Усі записи зберігаються у новій базі.



← Назад до акаунту


Мої замовлення

 **Замовлення #6ac4d8a4**


19 червня 2026 р. о 17:08

В обробці

	Тигролови Іван Багряний	x1 250 ₴
	Місто Валер'ян Підмогильний	x1 220 ₴

 **Доставка**


Нова Пошта
dddddddddd, ddddddddddddddd

 **Оплата**

Накладний платіж


Загальна сума:


550 ₴

 **Замовлення #сссвес46**


9 червня 2026 р. о 12:49

В обробці

	Тигролови Іван Багряний	x1 250 ₴
---	----------------------------	-------------

 **Доставка**

Нова Пошта
sdfsdfsdf, ddddddddddddddd

 **Оплата**

Накладний платіж

Загальна сума:

330 ₴

Рисунок 2.12 - Сторінка історії замовлень з переліком транзакцій

Для перегляду своїх книг створено сторінку мої книги з повним списком виставлених на продаж або обмін тут усіх книг цього конкретного користувача. Тут можна легко змінити кожне оголошення.

На рисунку 2.13 зображено сторінку мої книги з виставленими оголошеннями для продажу та швидкого відстеження статусів перевірки книг адміном цієї нової платформи. Через меню книги оновлюються дуже швидко.

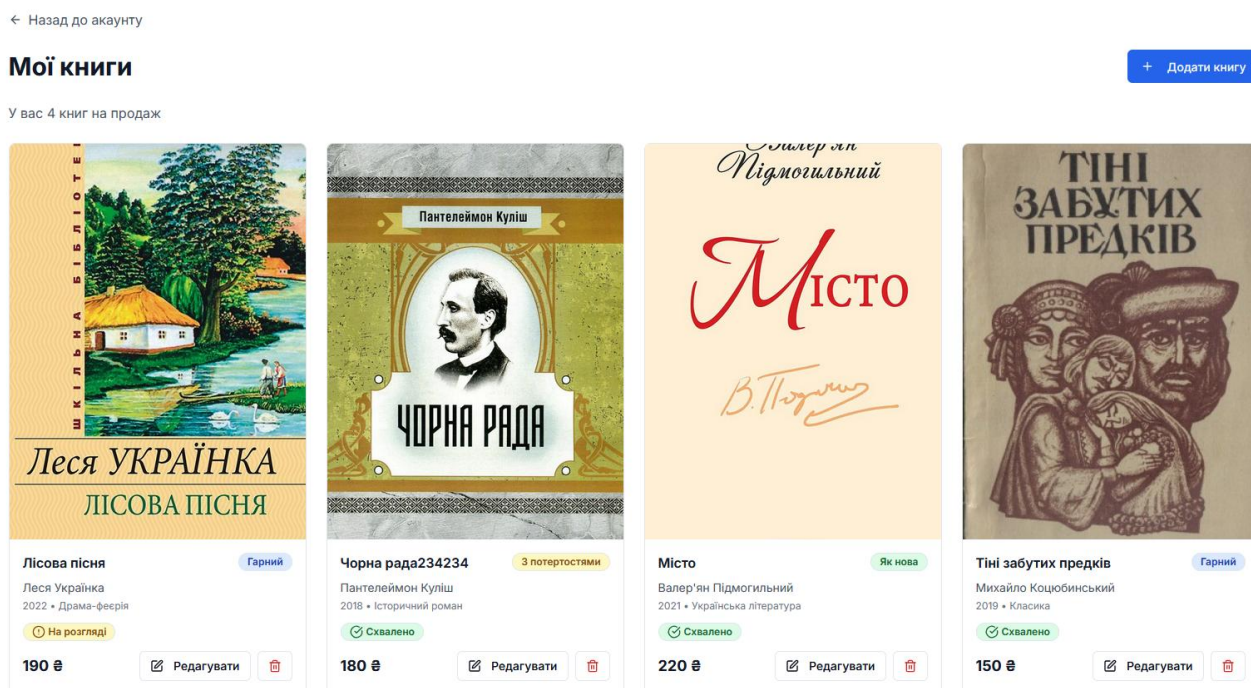


Рисунок 2.13 - Сторінка мої книги з виставленими оголошеннями

Для додавання нового оголошення створено форму додавання нової книги з полями для назви та опису та завантаження тут фотографій цієї книги на сайт. При введенні дані перевіряються дуже швидко. На рисунку 2.14 зображено форму додавання нової книги з текстовими полями та зручним вибором стану та ціни для публікації пропозиції цього користувача. Усі поля заповнюються тут дуже просто.

При додаванні нової книги заповнюються такі елементи:

- назва книжкового видання;
- автор;
- рік видання;
- жанр;
- ціна;
- стан книжкового примірника;

- ISBN номер;
- опис;
- фотографії обкладинки книги.

Додати книгу на продаж

Заповніть форму, щоб створити оголошення про продаж вашої книги

Назва книги *

Автор *

Рік видання *

Жанр *

Ціна (€) *

Стан книги *

ISBN (необов'язково)

Зображення обкладинки

Підтримуються формати JPG, PNG, WEBP. Максимум 5 фото.

Опис книги *

Рисунок 2.14 - Форма додавання нової книги з текстовими полями

Для зміни вже опублікованих книг створено форму редагування конкретної книги з швидким показом тут раніше введених даних про зареєстровану книгу з

загальної бази. Через форми зміни записуються дуже швидко. На рисунку 2.15 зображено форму редагування існуючої книги з наявними даними для швидкого внесення виправлень продавцем та оновлення інформації в загальній базі. При цьому база оновлюється тут миттєво.

Редагувати книгу ×

Внесіть зміни до інформації про вашу книгу.

Назва книги *

Автор *

Рік видання *

Жанр *

Ціна (€) *

Стан книги *

ISBN (необов'язково)

Зображення обкладинки (залиште порожнім, щоб не змінювати)

Підтримуються формати JPG, PNG, WEBP. Максимум 5 фото.

Опис книги *

Рисунок 2.15 - Форма редагування існуючої книги з наявними даними

Для простого підсумку розробки клієнтської частини було виконано це

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

повну перевірку всіх готових тут сторінок та елементів вигляду нової книжкової платформи. У підсумку сторінки працюють дуже правильно. При тестуванні інтерфейсу кожна окрема кнопка та кожне поле вводу даних тут відразу реагували на дії користувачів без будь-яких системних помилок. Після цього завершено поточний етап розробки.

2.3 Розробка структури бази даних інтернет-магазину

Для забезпечення надійного збереження даних та підтримки високої швидкості обробки запитів у системі BookSwar було обрано відому систему керування базами даних MySQL [7]. У підсумку робота сервісу стає дуже стабільною. Завдяки використанню цієї обраної технології постає повна цілісність коду та гнучкість при швидкому зв'язуванні між усіма робочими таблицями бази. У підсумку схема побудована дуже просто. У графічній частині кваліфікаційної роботи під кодом 2026.КВР.122.423.01.00.00 БД представлена детальна схема бази із зазначенням всіх зв'язків та типів полів.

Для логічного представлення даних у межах платформи було розроблено набір таблиць, опис структури яких наведено нижче:

Таблиця користувачів (users) - призначена для збереження облікових даних та профільної інформації покупців і продавців:

- id (Int, первинний ключ) - унікальний ідентифікатор користувача в базі даних;
- name (String) - повне ім'я та прізвище користувача;
- email (String, унікальне) - електронна пошта користувача для входу;
- password (String) - зашифрований пароль для перевірки доступу;
- phone (String, може мати порожнє значення) - контактний номер мобільного телефону;
- address (String, може мати порожнє значення) - адреса доставки посилки поштою;

					<i>2026.КВР.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		<i>37</i>

– role (String, значення за замовчуванням user) - роль користувача для перевірки прав доступу;

– createdAt, updatedAt (DateTime) - мітки часу створення та зміни облікового запису.

Таблиця авторів (authors) - зберігає перелік письменників для класифікації літератури:

– id (Int, первинний ключ) - унікальний ідентифікатор письменника в базі даних;

– name (String, унікальне) - повне ім'я автора книги.

Таблиця жанрів (genres) - містить категорії книг за літературними напрямками:

– id (Int, первинний ключ) - унікальний ідентифікатор жанру;

– name (String, унікальне) - назва літературного жанру.

Таблиця книг (books) - зберігає повний перелік виставлених для продажу чи обміну видань:

– id (Int, первинний ключ) - унікальний ідентифікатор книги;

– title (String) - назва книжкового видання;

– year (Int) - рік видання книги;

– price (Float) - встановлена вартість книги;

– description (String, тип Text) - детальний опис книги від продавця;

– isbn (String, може мати порожнє значення) - міжнародний стандартний номер книги;

– condition (Enum BookCondition) - фізичний стан книги (нова, хороша, задовільна, погана);

– status (Enum BookStatus, значення за замовчуванням PENDING) - статус перевірки оголошення адміном (очікує, схвалено, відхилено);

– authorId (Int, зовнішній ключ) - зв'язок із таблицею авторів;

– genreId (Int, зовнішній ключ) - зв'язок із таблицею жанрів;

– sellerId (Int, зовнішній ключ) - зв'язок із таблицею користувачів;

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						38
Зм.	Арк.	№ докум.	Підпис	Дата		

– createdAt, updatedAt (DateTime) - мітки часу публікації та оновлення оголошення.

Таблиця зображень книг (book_images) - призначена для зберігання посилань на файли обкладинок:

- id (Int, первинний ключ) - унікальний ідентифікатор зображення;
- url (String) - пряме посилання на файл у хмарному сховищі;
- bookId (Int, зовнішній ключ) - зв'язок із відповідною книгою.

Таблиця замовлень (orders) - зберігає інформацію про купівлю книг користувачами:

- id (String, первинний ключ, UUID) - унікальний ідентифікатор замовлення;
- userId (Int, зовнішній ключ) - зв'язок із покупцем із таблиці користувачів;
- totalPrice (Float) - загальна вартість куплених книг;
- deliveryPrice (Float) - вартість доставки посилки поштою;
- status (String, значення за замовчуванням pending) - поточний статус виконання замовлення;
- city (String) - місто доставки посилки;
- postalCode (String, може мати порожнє значення) - поштовий індекс відділення;
- deliveryMethod (String) - обраний спосіб доставки посилки;
- paymentMethod (String) - обраний спосіб оплати замовлення;
- notes (String, тип Text) - додаткові побажання користувача;
- createdAt, updatedAt (DateTime) - мітки часу оформлення та зміни статусу покупки.

Таблиця елементів замовлення (order_items) - зв'язує замовлення із конкретними книгами:

- id (Int, первинний ключ) - унікальний ідентифікатор рядка;
- orderId (String, зовнішній ключ) - посилання на таблицю замовлень;
- bookId (Int, зовнішній ключ) - посилання на таблицю книг;

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						39
Зм.	Арк.	№ докум.	Підпис	Дата		

- quantity (Int) - кількість замовлених книг;
- price (Float) - ціна книги на момент покупки.

Таблиця відгуків (reviews) - містить оцінки та коментарі покупців про книги:

- id (Int, первинний ключ) - унікальний ідентифікатор відгуку;
- bookId (Int, зовнішній ключ) - зв'язок із книгою;
- userId (Int, зовнішній ключ) - зв'язок із користувачем;
- rating (Int) - оцінка книги від одного до п'яти;
- comment (String, тип Text) - текстовий коментар покупця;
- createdAt (DateTime) - дата створення відгуку.

Для об'єднання таблиць у єдину систему в межах бази даних MySQL було спроектовано логічні зв'язки між усіма визначеними сутностями цієї платформи. У роботі зв'язки підтримують цілісність даних. Зокрема зв'язок один до багатьох побудовано між користувачами та книгами і замовленнями чи відгуками для швидкої адресації всіх запитів до бази. У базі зв'язки діють дуже наочно.

Для зв'язування книг з іншими блоками діє тут зв'язок один до багатьох між авторами чи літературними жанрами та виставленими на продаж книжковими виданнями. У базі зв'язки зберігаються постійно. У базі подібний зв'язок діє між книгами та файлами або окремими елементами замовлень для збереження тут всієї історії книжкових продажів на цьому веб-сайті. При цьому база працює швидко.

У підсумку побудована схема зв'язків підтримує цілісність даних та заважає виникненню колізій під час спільної роботи програми тут багатьох окремих користувачів. При цьому робота коду стає стабільною.

2.4 Програмування інтернет-магазину

Процес програмування та налаштування програмного коду платформи виконувався у середовищі розробки Visual Studio Code. Цей редактор було обрано через наявність великої кількості розширень, зручну роботу з терміналом та вбудовану підтримку Git. Для створення проекту було використано платформу

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40

Next.js на базі мови TypeScript. Такий підхід дозволив забезпечити строгу типізацію даних та уникнути поширених помилок ще на етапі написання коду.

Для стилізації інтерфейсу застосовано інструмент Tailwind CSS, який інтегрується безпосередньо у структуру Next.js. Усі стилі описуються за допомогою утилітарних класів, що значно прискорює верстку та оптимізує підсумковий розмір CSS файлів. Встановлення та керування залежностями проекту здійснювалося за допомогою менеджера пакетів Bun, який характеризується високою швидкістю виконання команд порівняно зі стандартним npm.

Зв'язок із реляційною базою даних MySQL налаштовано за допомогою Prisma ORM. Використання Prisma [8] дозволяє взаємодіяти з базою даних через типізований клієнт без написання прямих SQL запитів. Налаштування підключення до бази даних міститься у файлі конфігурації оточення .env, де вказано адресу сервера, ім'я користувача, пароль та назву бази даних.

Для ініціалізації та експорту єдиного екземпляра Prisma Client створено допоміжний файл у каталозі lib/prisma.ts. Це запобігає створенню додаткових з'єднань з базою даних під час перезапуску сервера в режимі розробки. Код цього файлу наведено у лістингу 2.1.

```
import { PrismaClient } from "@prisma/client"
const globalForPrisma = global as unknown as { prisma: PrismaClient }
export const prisma = globalForPrisma.prisma || new PrismaClient()
if (process.env.NODE_ENV !== "production") globalForPrisma.prisma = prisma
```

Лістинг 2.1 - Налаштування клієнта Prisma

Опис структури таблиць бази даних міститься у файлі schema.prisma. Після опису моделей у цьому файлі було виконано команду `npm run prisma db push` для синхронізації схеми з базою даних MySQL та генерації типів клієнта. Для автоматичного наповнення бази даних тестовими даними розроблено скрипт `prisma/seed.ts`, який запускається командою `npm run prisma db seed` та створює початковий набір жанрів, авторів та книг.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

Архітектура проекту базується на концепції App Router у Next.js, де файлова структура визначає маршрутизацію сайту. Усі сторінки, макети та компоненти згруповано у логічні каталоги. Структура сценаріїв та каталогів розробленого проекту містить такі основні елементи:

- app - папка з маршрутами сторінок, макетами (layouts) та API ендпоінтами;
- components - каталог перевикористовуваних React компонентів інтерфейсу (кнопки, картки, форми);
- actions - серверні дії (Server Actions) для виконання операцій з базою даних на стороні сервера;
- lib - утиліти та конфігураційні файли, зокрема налаштування Prisma;
- prisma - файли схеми бази даних та скрипти початкового наповнення;
- public - статичні ресурси, такі як зображення, іконки та шрифти.

Завдяки такій структурі код проекту є модульним та легким для підтримки. Локальний сервер розробки запускається за допомогою команди bun dev, після чого проект стає доступним за локальною адресою в браузері для тестування та налагодження.

2.4.1 Написання клієнтської частини

Клієнтська частина веб-сайту розроблена з використанням React компонентів та механізмів Next.js. Усі сторінки розділено на серверні та клієнтські. За замовчуванням усі компоненти в App Router є серверними (Server Components). Вони рендеряться на сервері, що зменшує обсяг JavaScript [9], який надсилається клієнту, та покращує швидкість завантаження сторінок і пошукову оптимізацію (SEO). Клієнтські компоненти (Client Components) позначені директивою use client на початку файлу та використовуються для забезпечення інтерактивності, наприклад, для обробки подій кліків, стану форм або взаємодії з браузерними API.

Для автентифікації користувачів розроблено форми входу та реєстрації.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						42
Зм.	Арк.	№ докум.	Підпис	Дата		

Валідація введених даних на стороні клієнта реалізована за допомогою бібліотеки Zod. Це дозволяє перевіряти коректність полей перед відправкою запиту на сервер та миттєво виводити повідомлення про помилки. Код схеми валідації реєстраційного запису наведено у лістингу 2.2.

```
const registerSchema = z.object({
  name: z.string().min(2, "Ім'я повинно містити принаймні 2 символи"),
  email: z.string().email("Некоректний email"),
  password: z.string().min(6, "Пароль повинен містити принаймні 6 символів"),
})
```

Лістинг 2.2 - Схема валідації реєстрації користувача

Після успішної валідації форми дані надсилаються до серверної дії `registerUser`, яка виконує шифрування пароля за допомогою бібліотеки `bcryptjs` та створює новий запис у таблиці користувачів. Сесія авторизованого користувача зберігається та контролюється за допомогою інструменту `Next-Auth` [10]. Захист приватних маршрутів, таких як кабінет користувача чи сторінка додавання нової книги, реалізовано у файлі `middleware.ts`, який перевіряє сесійний токен і перенаправляє неавторизованих користувачів на сторінку входу.

У кабінеті користувача передбачено можливість оновлення контактної інформації. Метод `updateProfile` приймає ідентифікатор користувача та нові дані (ім'я, телефон, адреса), оновлюючи відповідні поля в базі даних. Для відображення книг, виставлених користувачем на продаж чи обмін, реалізовано сторінку з викликом функції, яка фільтрує книги за ідентифікатором продавця.

Для відображення каталогу книг на головній сторінці та сторінці пошуку розроблено функцію `getApprovedBooks`. Вона підтримує фільтрацію за жанром, мінімальною та максимальною ціною, фізичним станом примірника, а також текстовий пошук за назвою чи автором. Окрім цього, реалізовано сортування книг за датою додавання, ціною та алфавітом. Усі параметри фільтрації синхронізуються

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

з URL-адресою сторінки через `searchParams`, що дозволяє користувачеві зберігати стан пошуку та ділитися посиланнями.

Для роботи з кошиком створено `CartContext` з використанням `React Context API`. Дані про додані книги, їх кількість та ціну зберігаються в пам'яті та синхронізуються з локальним сховищем браузера `LocalStorage`. Це дозволяє зберігати обрані товари навіть після перезавантаження сторінки. При оформленні замовлення клієнтська форма збирає дані про доставку та оплату, після чого викликає серверну дію `createOrder`.

Метод `createOrder` виконує створення замовлення у базі даних `MySQL`. Для забезпечення цілісності даних процес реалізовано у межах однієї транзакції `Prisma`. Спочатку оновлюються контактні дані користувача, потім створюється запис у таблиці `orders`, після чого додаються відповідні записи до таблиці `order_items` для кожної придбаної книги. Якщо на будь-якому етапі виникає помилка, транзакція повністю скасовується, що запобігає появі частково збережених або некоректних замовлень.

Для написання відгуків розроблено інтерактивну форму з вибором рейтингу від одного до п'яти зірок та текстовим коментарем. Після відправки форми викликається серверна дія `addReview`. Для миттєвого оновлення інтерфейсу без повного перезавантаження сторінки використовується функція `revalidatePath`, яка змушує `Next.js` оновити кеш сторінки книги на сервері та надіслати оновлений вміст клієнту.

Завантаження фотографій обкладинок книг реалізовано за допомогою інтеграції з хмарним сервісом `Cloudinary`. Серверна дія `uploadImage` приймає об'єкт `FormData` з файлом зображення, перетворює його у двійковий буфер та передає в `Cloudinary` через потік завантаження. Отримане у відповідь посилання записується в базу даних `MySQL`. Код цієї дії наведено у лістингу 2.3.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

```

export async function uploadImage(formData: FormData) {
  const file = formData.get('file') as File
  const bytes = await file.arrayBuffer()
  const buffer = Buffer.from(bytes)

  const uploadResult = await new Promise((resolve, reject) => {
    cloudinary.uploader.upload_stream(
      { folder: 'BookSwap' },
      (error, result) => {
        if (error) reject(error)
        else resolve(result)
      }
    ).end(buffer)
  })

  return { success: true, url: (uploadResult as any).secure_url }
}

```

Лістинг 2.3 - Функція завантаження зображень

Для підвищення зручності користувацького інтерфейсу на стороні клієнта інтегровано систему спливаючих сповіщень. При успішному виконанні будь-яких дій, як-от додавання книги до кошика чи реєстрація профілю, на екрані відображаються інформаційні повідомлення. Під час завантаження медіафайлів чи надсилання форм кнопки автоматично блокуються з показом анімованого індикатора завантаження, що запобігає повторній відправці даних та виключає можливість надсилання дублюючих запитів користувачем.

2.4.2 Написання admin-частини

Адміністративна панель призначена для модерації вмісту сайту та контролю за процесами покупок і продажів. Вона розташована за маршрутом `app/admin`.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

Доступ до цього розділу суворо обмежений на рівні middleware та серверних компонентів. При спробі неавторизованого доступу користувача з роллю user система автоматично повертає помилку доступу або перенаправляє його на головну сторінку.

Головна сторінка адмін-панелі містить загальну статистику системи. За допомогою агрегованих запитів Prisma розраховує загальну кількість користувачів, обсяг продажів у грошовому еквіваленті, кількість активних книг та замовлень. Такий аналітичний блок дозволяє адміністратору швидко оцінювати активність на платформі.

Для модерації оголошень про продаж книг розроблено інтерфейс, у якому відображаються картки видань зі статусом PENDING (очікує перевірки). Адміністратор може детально переглянути інформацію про книгу, її зображення та опис, після чого схвалити (APPROVED) або відхилити (REJECTED) оголошення. Зміна статусу книги виконується за допомогою серверної дії updateBookStatus. Код цієї функції наведено у лістингу 2.4.

```
export async function updateBookStatus(id: number, status: BookStatus) {
  try {
    await prisma.book.update({
      where: { id },
      data: { status },
    })
    revalidatePath("/admin")
    revalidatePath("/books")
    return { success: true }
  } catch (error) { return { error: "Помилка при оновленні статусу" } }}
```

Лістинг 2.4 - Функція оновлення статусу книги

Завдяки використанню методу revalidatePath зміни статусу відразу відображаються у каталозі книг для звичайних користувачів та зникають зі списку модерації в адмін-панелі. Це виключає потребу ручного оновлення сторінок у браузері. Повний код файлу керування книгами поданий у додатку Г.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

Для адміністрування замовлень реалізовано метод `getAllOrders`, який зчитує з бази даних повний список оформлених покупок. Функція завантажує дані про покупця (ім'я, пошта, телефон), загальну вартість замовлення, обраний метод доставки та перелік замовлених книг. Адміністратор має можливість змінювати статус замовлення, наприклад, переводити його у статус відправлено або виконано. Зміна статусу оновлює відповідні записи в базі даних та автоматично відображає актуальний стан замовлення у кабінетах покупців на сторінці історії покупок.

Для обробки виняткових ситуацій та помилок у всіх серверних діях та методах взаємодії з базою даних застосовано конструкції `try-catch`. Це запобігає падінню сервера та забезпечує повернення структурованих об'єктів помилок на клієнтську частину для відображення зрозумілих повідомлень користувачам.

У результаті виконання етапу програмування клієнтська та адміністративна частини були об'єднані в єдину працездатну систему. Розроблені скрипти та зв'язки забезпечують стабільну взаємодію користувача з базою даних MySQL, швидке завантаження контенту та надійну роботу всіх функціональних можливостей платформи.

2.5 Тестування інтернет-магазину

Задля підтвердження повної працездатності всіх розроблених модулів та перевірки їх відповідності технічному завданню було проведено комплексне тестування всієї платформи BookSwar. Основним завданням цього етапу стало виявлення та своєчасне усунення потенційних дефектів у логіці обробки даних. Система пройшла повну перевірку надійності. Додатково досліджено візуальне відображення всіх елементів інтерфейсу на різних екранах.

Під час верифікації функціоналу було застосовано декілька видів тестування, що дало змогу охопити всі аспекти роботи вебресурсу. Зокрема, було реалізовано такі підходи:

– функціональне тестування, яке дозволило перевірити коректність роботи всієї системи автентифікації, процесів додавання книг користувачами та

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

механізмів формування замовлень;

– тестування користувацького інтерфейсу (UI/UX), спрямоване на дослідження точності відображення розроблених макетів сторінок, плавності переходів та зручності навігації між розділами каталогу книг;

– кросбраузерне та адаптивне тестування, у межах якого виконано перевірку відображення сторінок у популярних браузерах Google Chrome, Mozilla Firefox та Safari на пристроях із різною діагоналлю екрана;

– тестування безпеки, що дозволило перевірити стійкість системи до неавторизованого доступу до адмін-панелі та коректність роботи посередників для захисту приватних даних користувачів.

Задля систематизації процесу перевірки було розроблено детальний контрольний список (checklist), який містить усі ключові параметри для повної оцінки якості розробленого програмного продукту.

У межах контрольного списку перевірено такі основні елементи:

– реєстрація нового користувача, яка очікувано завершується успішним створенням відповідного запису в реляційній базі даних;

– додавання нової книги користувачем з появою завантаженої обкладинки у хмарному сховищі Cloudinary [11] та запису в таблицях бази даних;

– фільтрація у загальному каталозі книг з очікуваним оновленням списку видань на сторінці відповідно до обраних параметрів;

– оформлення замовлення користувачем з успішним збереженням адреси доставки, способу оплати та суми у базі даних через транзакцію;

– адаптивність навігаційного меню та кошика з коректним згортанням елементів інтерфейсу на мобільних пристроях і планшетах;

– авторизація адміністратора з наданням доступу до спеціальної панелі модерації доданих користувачами оголошень;

– оновлення даних профілю користувача з надійним збереженням усіх внесених змін у базі даних MySQL;

– перевірка завантаження обкладинки книги з коректним відображенням

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						48
Зм.	Арк.	№ докум.	Підпис	Дата		

прев'ю на формі створення оголошення;

- робота з кошиком товарів, що передбачає додавання та видалення книг, а також автоматичний перерахунок загальної вартості;
- створення seed-даних для наповнення бази даних базовими категоріями, жанрами, авторами та користувачами.

Процес тестування включав імітацію критичних навантажень та введення некоректних даних у форми для пошуку потенційних проблем. Зокрема, було перевірено реакцію системи на спроби завантаження файлів великого об'єму або введення неправильного формату електронної пошти. Завдяки впровадженій валідації Zod усі помилки обробляються коректно, а користувач отримує зрозумілі підказки щодо їх вирішення.

Покроковий опис процесу тестування функціоналу BookSwar:

1) Реєстрація нового користувача:

Для початку тестування необхідно зареєструвати нового користувача.

- відкрито головну сторінку BookSwar;
- натиснуто кнопку Увійти та обрано реєстраційне меню;
- заповнено поля коректними даними: ім'я; електронна адреса; пароль.
- натиснуто кнопку Створити акаунт.

Очікується успішне створення нового облікового запису в системі.

Користувач автоматично перенаправляється на головну сторінку, як це зображено на рисунку 2.16.

Після успішного створення акаунту або для повторного входу здійснюється перевірка системи автентифікації.

- відкрито сторінку логіну;
- введено зареєстровану електронну адресу та пароль;
- натиснуто кнопку «Увійти».

Очікуваний результат: користувач успішно проходить авторизацію, система зберігає сесію, а у верхній панелі сайту (хедері) з'являється аватар користувача та доступ до особистого кабінету

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		49

Реєстрація

Створіть новий акаунт для доступу до всіх функцій

Ім'я та прізвище *

Іван Петренко

Please fill out this field.

Email *

your@email.com

Пароль *

.....

Підтвердження паролю *

.....

Телефон

+380991234567

Адреса

м. Київ, вул. Хрещатик, 1

Зареєструватися

Вже маєте акаунт? [Увійти](#)

Рисунок 2.16 - Процес успішної реєстрації нового користувача

2) Публікація нового оголошення про продаж книги:

Після реєстрації користувач повинен мати можливість додати свою першу книгу.

- здійснено вхід під новим обліковим записом продавця;
- натиснуто кнопку Додати книгу в навігаційному меню або кабінеті;
- заповнено форму, яка містить назву книги, опис, рік видання, ціна, стан примірника та isbn;
- завантажено зображення обкладинки книги;
- натиснуто кнопку Надіслати книгу.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		50

Очікується повідомлення про успішне надсилання книги на модерацію, форму заповнення якого наведено на рисунку 2.17.

Додати книгу на продаж

Заповніть форму, щоб створити оголошення про продаж вашої книги

Назва книги *	Автор *
<input type="text"/>	<input type="text"/>
Рік видання *	Жанр *
<input type="text"/>	Оберіть жанр
Ціна (₴) *	Стан книги *
<input type="text"/>	Оберіть стан
ISBN (необов'язково)	
<input type="text" value="978-966-01-0561-3"/>	
Зображення обкладинки	
<input type="text" value="Browse... No files selected."/>	
Підтримуються формати JPG, PNG, WEBP. Максимум 5 фото.	
Опис книги *	
<input type="text" value="Розкажіть про книгу, її стан, особливості..."/>	
<input type="button" value="↑ Додати книгу"/>	<input type="button" value="Скасувати"/>

Рисунок 2.17 - Форма додавання нової книги

Після успішного відправлення оголошення користувач переходить у свій кабінет. Додана книга з'являється у профілі користувача зі статусом очікування модерації, що продемонстровано далі на рисунку 2.18.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		51



Рисунок 2.18 - Картка книги зі статусом очікування

Перевірка оголошення адміністратором та його схвалення:

Адміністратор відіграє ключову роль у підтримці якості контенту на платформі.

здійснено вхід під обліковим записом адміністратора;

перейдено до панелі адміністратора та відкрито вкладку модерації оголошень;

здійснено пошук нової книги та перегляд її деталей;

натиснуто кнопку Схвалити книгу.

Процес перевірки заявки та інтерфейс прийняття рішення адміністратором показано на рисунку 2.19.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52



Рисунок 2.19 - Інтерфейс модерації та схвалення книги адміністратором

Очікується, що статус книги зміниться на схвалено, і вона стане доступною в загальному каталозі. Зміна статусу картки книги представлена на рисунку 2.20.

Мої книги

У вас 4 книг на продаж



Рисунок 2.20 - Картка схваленої книги в профілі користувача

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

Перевірка наявності книги в каталозі:

Фінальна перевірка полягає у відображенні оголошення для всіх користувачів сайту.

здійснено вихід з облікового запису адміністратора або перехід на головну сторінку як гість;

перейдено до розділу каталогу та використано фільтри чи пошуковий рядок для знаходження книги.

Очікується, що додана книга відображається у загальному каталозі та доступна для перегляду та купівлі всім відвідувачам платформи, що зображено на рисунку 2.21.



Рисунок 2.21 - Відображення схваленої книги в загальному каталозі

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

Додаткові методи тестування:

- тестування продуктивності, в межах якого здійснено вимірювання швидкості завантаження сторінок та часу відповіді сервера при збільшенні кількості користувачів за допомогою інструментів Google Lighthouse та PageSpeed Insights, результати яких показали високу оптимізацію ресурсів;
- тестування безпеки (Security Testing), яке включало аналіз на наявність поширених вразливостей (SQL Injection, XSS) за допомогою автоматизованих сканерів OWASP ZAP, де завдяки валідації Zod та механізмам JWT усі ризики вдалося мінімізувати;
- тестування зручності використання (Usability Testing), проведеного для оцінки інтуїтивності інтерфейсу та простоти виконання ключових завдань (пошук, додавання в кошик) з фокус-групою, яка зафіксувала високий рівень задоволеності користувачів;
- тестування доступності (Accessibility Testing), спрямоване на перевірку відповідності вебстандартам WCAG 2.1 для людей з обмеженими можливостями за допомогою інструментів Chrome DevTools Lighthouse, де система отримала високу оцінку.

Під час перевірки продуктивності було встановлено, що використання Next.js та оптимізованих запитів Prisma до бази даних MySQL забезпечує високу швидкість завантаження контенту навіть при низькій швидкості інтернет-з'єднання. Платформа продемонструвала стабільність під час роботи з багатьма одночасними запитами. Результати проведених заходів дозволяють зробити висновок про повну готовність BookSwar до експлуатації в реальних умовах. Усі виявлені дрібні недоліки інтерфейсу та обробки даних були успішно усунені під час фінальної ітерації налагодження перед публікацією проекту.

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		55

3 СПЕЦІАЛЬНИЙ РОЗДІЛ

3.1 Інструкція з розміщення інтернет-магазину в Інтернеті

Вибір надійного серверного середовища прямо впливає на стабільність роботи магазину. Для розгортання веб-сайту BookSwar було обрано хмарну платформу Vercel [12], яка чудово підходить для Next.js додатків. Хмарна інфраструктура автоматизує процеси налаштування оточення, оптимізує ресурси та забезпечує швидке завантаження сторінок.

Процес розгортання проекту розпочинається зі створення бази даних MySQL на хмарному сервісі Railway. Для налаштування бази даних виконайте наступні покрокові дії:

- зареєструйте обліковий запис на офіційному сайті Railway;
- створіть новий проект у панелі керування та оберіть опцію розгортання порожньої бази даних MySQL;
- зачекайте запуску контейнера бази даних та перейдіть у вкладку Variables для перегляду конфігурацій;
- скопіюйте згенерований платформою рядок підключення DATABASE_URL для подальшого використання у налаштуваннях.

Панель створення та налаштування нової бази даних Railway детально зображено на рисунку 3.1.

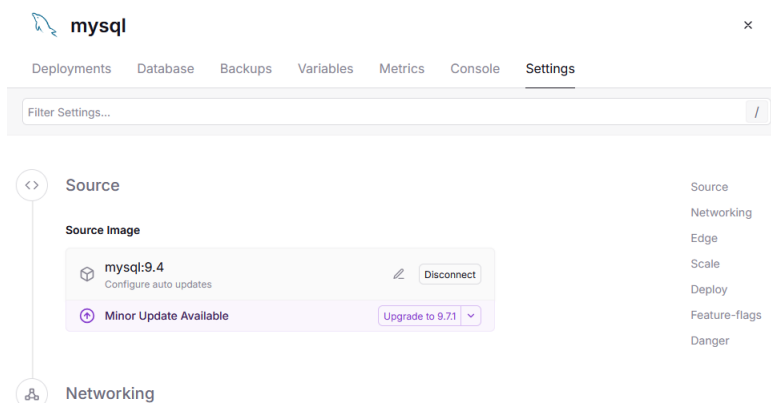


Рисунок 3.1 - Панель створення нової бази даних Railway

Після успішного створення бази даних виконайте розгортання коду проекту на сервісі Vercel. Для деплою коду виконайте такі кроки:

- відкрийте панель керування Vercel та авторизуйтеся за допомогою облікового запису GitHub;
- натисніть кнопку імпорту проекту та оберіть потрібний репозиторій BookSwar зі списку доступних сховищ;
- перейдіть до вікна налаштувань конфігурацій та додайте необхідні змінні середовища;
- вставте скопійовану адресу підключення до бази даних Railway у поле під назвою DATABASE_URL;
- пропишіть секретний ключ для авторизації користувачів у змінній NEXTAUTH_SECRET;
- натисніть кнопку Deploy для початку автоматичного збирання та компіляції файлів проекту.

Вікно налаштування змінних середовища та параметрів проекту на Vercel представлено на рисунку 3.2.

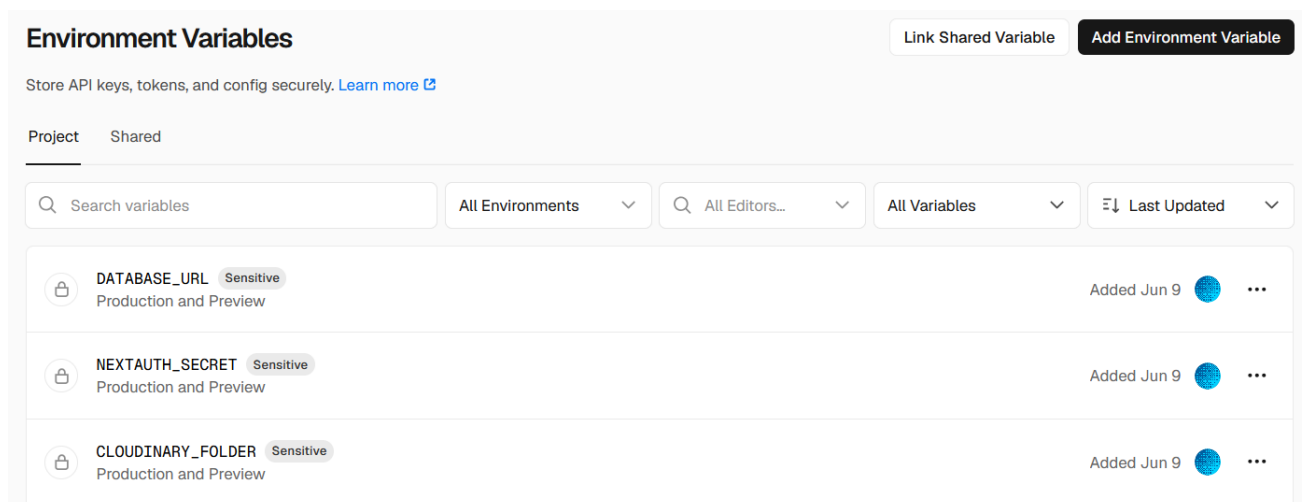


Рисунок 3.2 - Вікно налаштування змінних середовища Vercel

Під час розгортання сервіс Vercel автоматично встановлює необхідні залежності, компілює файли сторінок та оптимізує клієнтський JavaScript. Після

завершення процесу збирання система генерує тимчасову адресу для перевірки роботи сайту. Для прив'язки власного доменного імені до створеного проекту виконайте такі кроки:

- відкрийте вкладку налаштувань доменів у панелі керування проектом Vercel;
- введіть придбане доменне ім'я у відповідне текстове поле та натисніть кнопку підтвердження;
- скопіюйте запропоновані системою значення для DNS записів типу A та CNAME;
- перейдіть до панелі реєстратора доменів та додайте ці дані у налаштуваннях DNS зони;
- зачекайте оновлення записів та автоматичної генерації безкоштовного SSL сертифіката від Vercel.

Налаштування записів домену в панелі керування реєстратора зображено на рисунку 3.3.

l.verc
Learn

Add Domains

Add one or more domains to this project and choose one destination for all of them. Domains connected to an environment are aliased to the most recent deployment in that environment.

Or paste multiple domains, one per line.

Connect to an environment

Production

Redirect to Another Domain

307 Temporary Redirect No Redirect

Cancel Add 0 Domains

Рисунок 3.3 - Налаштування записів домену в панелі реєстратора

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

Після оновлення DNS записів та активації SSL сертифіката сайт стає доступним за постійною адресою в мережі Інтернет. Для фінальної верифікації працездатності розгорнутої платформи виконайте такі перевірочні кроки:

- введіть адресу сайту в новій вкладці браузера та переконайтеся в наявності захищеного SSL з'єднання;
- протестуйте навігацію між категоріями та сторінками для перевірки з'єднання з базою даних Railway;
- зареєструйте тестовий профіль користувача для підтвердження коректної роботи авторизації.

Завдяки успішному виконанню всіх етапів інструкції веб-сайт BookSwar повністю розгорнуто в мережі Інтернет для загального доступу користувачів.

3.2 Інструкція з обслуговування та наповнення інтернет-магазину

Стабільне функціонування книжкової платформи BookSwar потребує належних серверних ресурсів для забезпечення швидкої обробки запитів та збереження медіафайлів. Мінімальна конфігурація сервера для хостингу та розгортання бази даних MySQL вимагає наступних параметрів:

- один потік процесора з частотою від двох гігагерц для обробки запитів користувачів;
- один гігабайт оперативної пам'яті для стабільної роботи Next.js додатку та кешування даних;
- двадцять гігабайтів вільного простору на швидкому твердотільному накопичувачі SSD;
- встановлена операційна система Linux для забезпечення надійності та безпеки середовища.

Для керування вмістом платформи розроблено зручні інструменти як для звичайних користувачів, так і для адміністраторів сайту.

Для додавання нової книги на продаж або обмін звичайному користувачеві

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		59

необхідно виконати такі покрокові дії:

- перейдіть до особистого кабінету користувача після проходження авторизації в системі;
- натисніть кнопку Додати книгу в навігаційному меню або на сторінці кабінету;
- заповніть текстові поля форми, вказавши назву книги, опис сюжету, рік видання та міжнародний номер isbn;
- вкажіть бажану ціну книги та оберіть її фізичний стан із випадального списку варіантів;
- натисніть на область завантаження зображення та оберіть файл обкладинки книги на комп'ютері;
- натисніть кнопку Надіслати для передачі даних на сервер та автоматичного відправлення оголошення на модерацию.

Для редагування раніше опублікованого оголошення про продаж книги користувачеві необхідно виконати наступні дії:

- відкрийте сторінку Мої книги в особистому кабінеті користувача;
- оберіть потрібне видання зі списку та натисніть кнопку Редагувати на картці книги;
- внесіть необхідні зміни у полях форми, оновивши ціну, опис, стан або замінивши фото обкладинки;
- натисніть кнопку Зберегти для запису оновленої інформації у базу даних.

Для видалення оголошення про продаж книги користувачеві або адміністратору необхідно виконати такі дії:

- відкрийте список власних оголошень в особистому кабінеті або перейдіть до загального списку в адмін-панелі;
- знайдіть необхідну книгу та натисніть кнопку Видалити;
- підтвердіть видалення оголошення у спливаючому модальному вікні запиту;

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підпис	Дата		

– дочекайтеся автоматичного оновлення списку та видалення зображень книги з хмарного сховища.

Для модерації (схвалення або відхилення) оголошення адміністратору платформи необхідно виконати наступні дії:

– авторизуйтеся в системі під обліковим записом з правами адміністратора;

– перейдіть до панелі керування за адресою `app/admin` та відкрийте розділ модерації книг;

– оберіть нове оголошення зі списку очікування, перевірте правильність заповнення текстових полей та фотографію;

– натисніть кнопку Схвалити для публікації книги в загальному каталозі або кнопку Відхилити для видалення оголошення з черги перевірки.

Завдяки впровадженню описаних інструкцій процес обслуговування сайту та наповнення його контентом є простим та інтуїтивно зрозумілим для всіх груп користувачів книжкової платформи.

3.3 Інструкція з популяризації та підтримки інтернет-магазину

Створення якісної книжкової платформи є першим кроком до залучення аудиторії та розвитку культури обміну літературою. Після успішного запуску веб-сайту BookSwar важливим етапом стає впровадження довгострокового плану з просування ресурсу в мережі Інтернет та забезпечення технічної підтримки користувачів. Ефективна стратегія популяризації проекту має поєднувати сучасні інструменти інтернет-маркетингу, роботу з громадою читачів та офлайн-заходи для постійного збільшення аудиторії.

Перспективним напрямком просування є пошукова оптимізація (SEO), яка закладена на рівні архітектури сайту завдяки технологіям Next.js. У майбутньому планується розширення семантичного ядра проекту шляхом аналізу низькочастотних запитів користувачів, пов'язаних із пошуком та обміном

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

конкретних видань. Для цього передбачається створення тематичного блогу з оглядами літературних новинок та рекомендаціями книг. Це дозволить отримувати безкоштовний органічний трафік із пошукових систем Google та Bing завдяки релевантному текстовому контенту.

Важливу роль у формуванні лояльної спільноти має відігравати маркетинг у соціальних мережах (SMM). На базі сторінок платформи в Instagram, Facebook та Telegram планується регулярна публікація підбірок популярної літератури, відгуків користувачів та інтерактивних матеріалів. Для швидкого залучення підписників передбачається проведення щомісячних розіграшів книжкових новинок та надання бонусів за активність у соціальних мережах.

Для швидкого залучення перших покупців та продавців планується запуск платних рекламних кампаній. Контекстна реклама Google Ads дозволить показувати оголошення гарячій аудиторії, яка шукає можливості придбати або продати вживані книги. Додатково планується використання таргетованої реклами у соціальних мережах з орієнтацією на інтереси любителів читання, студентів та молоді. На рисунку 3.4 зображено рекламний кабінет Facebook як один зі способів створення таргетованої реклами.

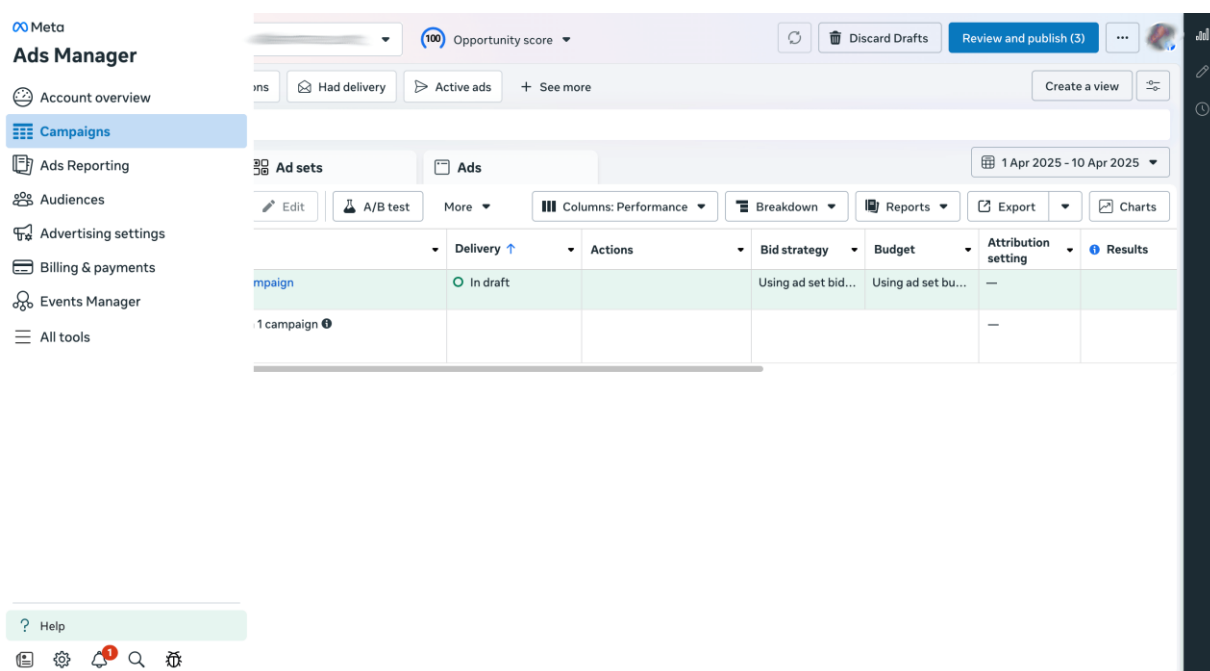


Рисунок 3.4 - Приклад рекламного кабінету Facebook

Для залучення офлайн-аудиторії та популяризації платформи серед молоді планується реалізація таких кроків:

- розповсюдження брендovаних буклетів та паперові закладок з QR-кодом платформи біля вищих навчальних закладів;
- розміщення спеціальних полиць для буккросингу з логотипом BookSwap у популярних міських кав'ярнях та бібліотеках;
- проведення спільних зустрічей та обговорень під егідою місцевих читацьких клубів;
- виділення книг як подарунків для спонсорської підтримки студентських та культурних заходів.

Наступним кроком для підвищення активності на сайті має стати розробка інструментів утримання користувачів. Автоматизована система збору електронних адрес під час реєстрації дозволить налагодити пряму комунікацію з аудиторією.

Для утримання наявної аудиторії та стимулювання повторної активності на сайті планується впровадити такі технічні рішення:

- запуск автоматичної системи тригерних розсилок про статус модерації оголошень та нові повідомлення;
- інтеграція реферальної програми з нарахуванням бонусних балів за запрошення нових активних учасників;
- формування та розсилка персональних книжкових рекомендацій на основі історії пошуку та інтересів користувача.

Важливим аспектом тривалої підтримки платформи є надання швидкої технічної допомоги користувачам. Адміністрація сайту має оперативно реагувати на звернення щодо помилок авторизації, проблем із завантаженням фото обкладинок або труднощів під час оформлення замовлень. Постійне оновлення модулів безпеки, резервне копіювання бази даних MySQL та оптимізація швидкодії сервера гарантуватимуть високу надійність роботи проекту BookSwap у майбутньому.

					<i>2026.KBP.122.423.01.00.00 ПЗ</i>	Арк.
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		63

4 ЕКОНОМІЧНИЙ РОЗДІЛ

Метою економічної частини даного дипломного проєкту є проведення економічних розрахунків, спрямованих на визначення економічної ефективності розробки інтернет-магазину вживаних книг «BookSwar», прийняття рішення про подальший розвиток і впровадження або ж недоцільність проведення відповідної розробки.

Об'єктом розробки є інтернет-магазин «BookSwar».

Розрахунок вартості розробки виконується в декілька етапів:

- описати технологічний процес розробки із зазначенням трудомісткості кожної операції;
- визначити суму витрат на оплату праці основного і допоміжного персоналу, включаючи відрахування на соціальні заходи;
- обчислити витрати на електроенергію;
- нарахувати суму амортизаційних відрахувань;
- визначити суму накладних витрат;
- скласти кошторис та визначити собівартість робіт;
- розрахувати ціну робіт;
- визначити економічну ефективність та термін окупності.

4.1 Визначення стадій технологічного процесу та загальної тривалості проведення розробки інтернет-магазину

В цьому підрозділі розглянемо основні етапи технологічного процесу для розробки інтернет-магазину. Для визначення загальної тривалості проведення робіт доцільно дані витрат часу по окремих операціях технологічного процесу звести у таблицю 4.1.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64

Таблиця 4.1 - Середній час виконання робіт по обслуговуванню та стадії (операції) технологічного процесу

№ п/п	Назва операції (стадії)	Виконавець	Середній час виконання операції, год.
1	Планування та аналіз	Керівник проєкту	15
2	Розробка технічного завдання	Керівник проєкту	10
		Розробник	5
3	Дизайн інтерфейсу	Розробник	20
4	Розробка функціоналу	Розробник	70
5	Тестування та відладка	Тестувальник	25
6	Документування	Керівник проєкту	5
		Розробник	5
7	Розгортання та підтримка	Розробник	10
8	Управління проєктом	Керівник проєкту	5
Разом			170

Сумарний час виконання операцій технологічного процесу становить 170 години.

4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи

У даному підрозділі проводиться аналіз і розрахунок витрат, пов'язаних з оплатою праці та відрахуваннями на соціальні заходи, що необхідні для розробки інтернет-магазину.

Розмір заробітної плати залежить від складності та умов виконуваної роботи, професійно-ділових якостей працівника, результатів його праці та діяльності підприємства.

Основна заробітна плата розраховується за формулою:

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						65
Зм.	Арк.	№ докум.	Підпис	Дата		

$$Z_{\text{осн.}} = T_c * K_r \quad (4.1)$$

де:

– T_c – тарифна ставка, грн. (приймаємо для керівника проекту - 400 гривень за годину, розробника - 300 гривень за годину, тестувальника - 150 гривень за годину);

– K_r – кількість відпрацьованих годин.

Отже, основна заробітна плата для:

- Керівника проекту $Z_{\text{осн.}} = 35 * 400 = 14\,000$ гривень;
- Розробника $Z_{\text{осн.}} = 110 * 300 = 33\,000$ гривень;
- Тестувальника $Z_{\text{осн.}} = 25 * 150 = 3\,750$ гривень.

Сумарна основна заробітна плата становить:

$$Z_{\text{осн.}} = 14\,000 + 33\,000 + 3\,750 = 50\,750 \text{ гривень.}$$

Додаткова заробітна плата становить 10 – 15 % від суми основної заробітної плати.

$$Z_{\text{дод.}} = Z_{\text{осн.}} \cdot K_{\text{допл.}} \quad (4.2)$$

де: $K_{\text{допл.}}$ – коефіцієнт додаткових виплат працівникам.

Отже додаткова заробітна плата по категоріях працівників становить:

- Керівника проекту $Z_{\text{дод.2}} = 14\,000 * 0,1 = 1\,400$ гривень;
- Розробника $Z_{\text{дод.3}} = 33\,000 * 0,1 = 3\,300$ гривень;
- Тестувальника $Z_{\text{дод.4}} = 3\,750 * 0,1 = 375$ гривень.

Загальна додаткова заробітна плата становить:

$$Z_{\text{дод.}} = 1\,400 + 3\,300 + 375 = 5\,075 \text{ гривень.}$$

Звідси загальні витрати на оплату праці ($V_{\text{о.п.}}$) визначаються за формулою:

$$V_{\text{о.п.}} = Z_{\text{осн.}} + Z_{\text{дод.}} \quad (4.3)$$

$$V_{\text{о.п.}} = 50\,750 + 5\,075 = 55\,825 \text{ гривень.}$$

Єдиний соціальний внесок (ЄСВ – 22%) визначається за формулою:

$$V_{\text{ЄСВ}} = V_{\text{о.п.}} * 0,22 \quad (4.4)$$

$$V_{\text{ЄСВ}} = 55\,825 * 0,22 = 12\,282 \text{ гривні.}$$

Проведені розрахунки витрат на оплату праці наведено у таблиці 4.2.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						66
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.2 – Зведені розрахунки витрат на оплату праці

№ п/ п	Категорія працівників	Основна заробітна плата, грн.			Додаткова заробітна плата, грн.	ЄСВ, грн.	Всього витрати на оплату праці, грн. 6 = 3+4+5
		Тарифна ставка, грн.	К-сть годин	Фактично нарах. зарплати, грн.			
		1	2	3	4	5	6
1	Керівник проєкту	400	35	14 000	1 400		
2	Розробник	300	110	33 000	3 300		
3	Тестувальник	150	25	3 750	375		
Разом				50 750	5 075	12 282	68 107

Отже, загальні витрати на оплату праці становлять 68 107 грн.

4.3 Розрахунок витрат на електроенергію

Розрахуємо вартість електроенергії. Затрати на електроенергію 1-ці обладнання визначаються за формулою:

$$Z_{\text{в}} = W * T * S \quad (4.5)$$

де: W – необхідна потужність, кВт; T – кількість годин роботи обладнання;
 S – вартість кіловат-години електроенергії (приймаємо 15,94 грн).

В нашій системі є 1 ПК. Витрати на електроенергію для цього комп'ютера обчислимо окремо, взявши за основу, що час роботи обладнання обчислюється в залежності від виконуваних робіт (згідно табл. 4.1) і споживані потужності наступні: комп'ютер – 0,4 кВт/год.

$$Z_{\text{ек}} = 0,4 * 170 * 15,94 = 1 084 \text{ гривні}$$

Витрати на електроенергію становлять 1 084 грн.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		67

4.4 Розрахунок суми амортизаційних відрахувань інтернет-магазину

Характерною особливістю застосування основних фондів у процесі виробництва є їх відновлення. Для відновлення засобів праці у натуральному виразі необхідне їх відшкодування у вартісній формі, яке здійснюється шляхом амортизації.

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення

Комп'ютери та оргтехніка належать до четвертої групи основних фондів.

Амортизація на них нараховується лише в випадку, якщо мінімально допустимі строки їх корисного використання 2 роки. Для визначення амортизаційних відрахувань застосовуємо формулу:

$$A = \frac{B_B * H_A}{100\%} * T, \quad (4.6)$$

де:

- А – амортизаційні відрахування за звітний період, грн.;
- B_B – балансова вартість групи основних фондів на початок звітного періоду, грн.;
- H_A – норма амортизації, 0,04 %.

Оскільки для написання програми та її тестування використовується один ПК, вартістю 15 000 грн., то сума амортизаційних відрахувань становитиме:

$$A = \frac{15\,000 * 0,04}{150} * 170 = 680 \text{ грн.}$$

4.5 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління підприємства (фірми) та створення необхідних умов праці.

В залежності від організаційно-правової форми діяльності господарюючого суб'єкта, накладні витрати можуть становити 20–60 % від суми основної та

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		68

додаткової заробітної плати працівників.

$$H_B = V_{o.п.} * 0,2..0,6 \quad (4.7)$$

де: H_B – накладні витрати.

$$H_B = 55\,825 * 0,4 = 22\,330 \text{ грн.}$$

4.6 Складання кошторису витрат та визначення собівартості

Для складання кошторису витрат та визначення собівартості, результати проведених вище розрахунків зведемо у таблиці 4.4.

Таблиця 4.4 - Кошторис витрат BookSwap

	Зміст витрат	Сума, грн.	В % до загальної суми
1.	Витрати на оплату праці	68 107	74
2.	Витрати на електроенергію	1 084	1
3.	Амортизаційні відрахування	680	1
4.	Накладні витрати	22 330	24
5.	Собівартість	92 201	100

Собівартість (C_B) НДР розрахуємо за формулою:

$$C_B = V_{o.п.} + V_{c.з} + 3e + A + H_B \quad (4.8)$$

Отже, собівартість дорівнює

$$C_B = 55\,825 + 12\,282 + 1\,084 + 680 + 22\,330 = 92\,201 \text{ грн.}$$

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		69

4.7 Розрахунок ціни інтернет-магазину

Розрахунок ціни науково-дослідної роботи включає в себе урахування різноманітних факторів, таких як рівень рентабельності, собівартість та податкова ставка.

Ціну робіт можна визначити за формулою:

$$Ц = C_{\text{в}} * (1 + P_{\text{рен}}) * (1 + \text{ПДВ}), \quad (4.9)$$

де:

- $C_{\text{в}}$ – собівартість;
- $P_{\text{рен}}$ – рівень рентабельності;
- ПДВ – ставка податку на додану вартість.

$$Ц = 92\,201 * (1 + 0,3) * (1 + 0,2) = 143\,834 \text{ грн.}$$

4.8 Визначення економічної ефективності і терміну окупності капітальних вкладень

Ефективність виробництва – це узагальнене і повне відображення кінцевих результатів використання робочої сили, засобів та предметів праці на підприємстві за певний проміжок часу.

Для визначення ефективності продукту розраховують чисту теперішню вартість (ЧТВ) і термін окупності (Ток).

$$\text{ЧТВ} = -C_{\text{в}} + \sum_{i=1}^t \frac{\Gamma_{\text{п}}}{(1 + i)^t}, \quad (4.10)$$

де:

- $C_{\text{в}}$ – собівартість розробки;
- $\Gamma_{\text{п}}$ – грошовий потік за t – ий рік;
- t – відповідний рік проекту;
- i – величина дисконтної ставки (10...15%).

$$\text{ЧТВ} = -92\,201 + \frac{51632}{(1 + 0,1)^1} + \frac{51632}{(1 + 0,1)^2} + \frac{51632}{(1 + 0,1)^3} = 36\,229 \text{ грн}$$

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						70
Зм.	Арк.	№ докум.	Підпис	Дата		

Якщо $ЧТВ \geq 0$, то проект може бути рекомендований до впровадження.

Термін окупності визначається за формулою:

$$T_{ок} = T_{пв} + \frac{H_B}{\Gamma_{пр}} \quad (4.11)$$

де: $T_{пв}$ – період до повного відшкодування витрат, років; H_B – невідшкодовані витрати на початок року, грн.; $\Gamma_{пр}$ – грошовий потік на початок року, грн.

$$T_{ок} = 2 + \frac{2\,592}{51\,632} = 2,1р.$$

Всі дані внесемо в зведену таблицю 4.5.

Таблиця 4.5 – Техніко-економічні показники BookSwap

№ п/п	Показник	Значення
1.	Собівартість, грн.	92 201
2.	Плановий прибуток або грошовий потік, грн.	51 632
3.	Ціна, грн.	143 834
4.	Чиста теперішня вартість, грн.	36 229
5.	Термін окупності, рік	2,1

Прибутковість проекту та термін окупності свідчать про його фінансову ефективність та здатність повернути капітальні вкладення протягом 2,1 року. Отже, на основі отриманих показників можна зробити висновок, що розробка BookSwap є доцільною з економічної точки зору.

5 ОХОРОНА ПРАЦІ, ТЕХНІКА БЕЗПЕКИ ТА ЕКОЛОГІЧНІ ВИМОГИ

5.1 Перевірка знань працівників з питань охорони праці

Організація навчання та перевірки знань з питань охорони праці є важливою складовою системи управління безпекою на будь-якому підприємстві [13]. Всі працівники під час прийняття на роботу і в процесі трудової діяльності повинні проходити інструктажі та перевірку знань відповідних нормативно-правових актів. Це дозволяє суттєво знизити рівень виробничого травматизму та забезпечити належний рівень культури безпеки на робочих місцях.

Відповідно до чинного законодавства, існують різні види контролю знань працівників, які класифікуються за часом проведення та причинами організації:

- Первинна перевірка знань проводиться перед початком самостійної роботи працівника після відповідного навчання за професією;
- Періодична перевірка знань здійснюється у визначені терміни, які безпосередньо залежать від посади працівника та специфіки виконуваних робіт;
- Позачергова перевірка знань призначається у випадках введення нових нормативних актів, модернізації обладнання або після виявлення порушень вимог безпеки.

Перевірку знань працівників з питань охорони праці здійснює спеціальна комісія, склад якої затверджується наказом керівника підприємства [14]. До складу комісії зазвичай входять фахівці з охорони праці, представники керівництва компанії та професійних спілок. Результати перевірки оформлюються відповідним протоколом, а працівникам, які успішно склали іспит, видаються посвідчення встановленого зразка.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		72

5.2 Оптимальні кліматичні умови

Мікроклімат виробничих приміщень суттєво впливає на загальне самопочуття, працездатність та здоров'я людини. Оптимальні мікрокліматичні умови забезпечують загальне відчуття теплового комфорту протягом всієї робочої зміни без напруження механізмів терморегуляції. Основними параметрами мікроклімату є температура, відносна вологість та швидкість руху повітря. Параметри мікроклімату диференціюються залежно від важкості виконуваної роботи та конкретного періоду року. Оптимальні величини цих показників для різних категорій робіт у теплий період року наведені у таблиці 5.1 [15].

Таблиця 5.1 - Оптимальні норми мікроклімату для теплого періоду року

Категорія робіт за важкістю	Температура повітря, град. С	Відносна вологість, %	Швидкість руху повітря, м/с
Легка - Іа	22-25	40-60	0.1
Легка - Іб	21-24	40-60	0.1
Середньої важкості - ІІа	18-22	40-60	0.2
Середньої важкості - ІІб	17-21	40-60	0.2
Важка - ІІІ	16-18	40-60	0.3

Дотримання зазначених параметрів у виробничих приміщеннях досягається за допомогою сучасних систем вентиляції та кондиціонування повітря. Правильно організований повітрообмін дозволяє підтримувати чистоту повітряного середовища та створює сприятливі умови для високопродуктивної праці.

ВИСНОВКИ

Виконання кваліфікаційної роботи дозволило успішно розробити та протестувати сучасну вебплатформу для обміну книгами BookSwar. Отримані результати повністю відповідають початковим вимогам технічного завдання на проектування. Програма містить усі необхідні модулі для зручної взаємодії користувачів та ефективного керування контентом. Практична реалізація проєкту підтверджує правильність обраного технологічного стеку та архітектурних рішень.

Показники продуктивності розробленого сайту демонструють високу швидкість завантаження сторінок та миттєву реакцію інтерфейсу на дії відвідувача. Використання серверної генерації вмісту дозволило значно покращити видимість ресурсу для автоматичних алгоритмів пошукових систем. Система безпеки надійно захищає приватні дані клієнтів та гарантує стабільну роботу сесій авторизації. Адаптивна верстка забезпечує правильне відображення всіх елементів керування на екранах будь-якого розміру.

Напрями подальшої роботи над темою передбачають поступове розширення функціональних можливостей платформи. Доцільно додати систему миттєвих повідомлень для прямого спілкування між покупцями та продавцями всередині сайту. Майбутня інтеграція платіжних шлюзів дозволить проводити безпечні фінансові операції безпосередньо через інтерфейс магазину. Також планується розробка окремого мобільного додатка для підвищення зручності використання сервісу власниками смартфонів.

Результати дослідження та опис ходу розробки були представлені у вигляді доповіді під час виступу перед членами комісії коледжу. Основні положення роботи та продемонстровані програмні рішення отримали позитивні відгуки фахівців. Весь обсяг робіт від початкового проектування до фінального розгортання на сервері студент виконав повністю самостійно.

У процесі виконання проєкту розробник успішно розв'язав кілька нових складних завдань. Створення гнучкої схеми бази даних на базі MySQL дозволило ефективно зберігати складні зв'язки між різними сутностями. Впровадження

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						74
Зм.	Арк.	№ докум.	Підпис	Дата		

інструментів Prisma гарантує високу надійність роботи з даними та відсутність помилок типізації. Програміст самостійно реалізував алгоритми автоматичного стиснення зображень перед їх завантаженням у хмарне сховище. Це значно прискорює роботу сайту та заощаджує вільне місце на зовнішніх серверах.

Побудована система маршрутизації дозволяє легко додавати нові розділи без порушення логіки роботи всього додатка. Використання компонентного підходу сприяє повторному використанню коду та спрощує довгострокову підтримку продукту. Кінцевий результат показує високу якість написання скриптів та відповідність сучасним стандартам вебпрограмування. Робота має вагомое практичне значення для розвитку культури обміну літературою в цифровому середовищі. Подальше впровадження рекомендацій допоможе залучити велику кількість нових активних читачів. Всі поставлені цілі досягнуті в повному обсязі та у встановлені терміни. Таким чином проєкт BookSwar є повністю завершеним та готовим до повноцінного комерційного використання. Студент отримав цінний досвід розробки повнофункціональних систем та підвищив свій професійний рівень. Даний розділ завершує пояснювальну записку та підсумовує всі проведені дослідження. Кожен етап розробки отримав своє логічне завершення та детальне пояснення в основному тексті. Висновки підтверджують успішне розв'язання всіх технічних та організаційних проблем під час створення платформи. Проєкт демонструє високу конкурентну спроможність на ринку спеціалізованих букіністичних сервісів.

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
						75
Зм.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ

- 1) Farley D. Modern Software Engineering. Boston : Addison-Wesley, 2021. 256 p.
- 2) Lidwell W., Holden K., Butler J. Universal Principles of Design. Beverly : Rockport Publishers, 2023. 400 p.
- 3) 3Shvets A. Dive Into Design Patterns. Vancouver : Refactoring.Guru, 2021. 408 p.
- 4) Next.js Documentation : вебсайт. URL: <https://nextjs.org/docs> (дата звернення: 10.06.2026).
- 5) React Documentation : вебсайт. URL: <https://react.dev> (дата звернення: 10.06.2026).
- 6) Tailwind CSS Documentation : вебсайт. URL: <https://tailwindcss.com/docs> (дата звернення: 10.06.2026).
- 7) MySQL 8.0 Reference Manual : вебсайт. URL: <https://dev.mysql.com/doc/refman/8.0/en/> (дата звернення: 10.06.2026).
- 8) Prisma Documentation : вебсайт. URL: <https://www.prisma.io/docs> (дата звернення: 10.06.2026).
- 9) Vanderkam D. Effective TypeScript. Sebastopol : O'Reilly Media, 2024. 320 p.
- 10) NextAuth.js Documentation : вебсайт. URL: <https://next-auth.js.org/getting-started/introduction> (дата звернення: 10.06.2026).
- 11) Cloudinary Documentation : вебсайт. URL: <https://cloudinary.com/documentation> (дата звернення: 10.06.2026).
- 12) Vercel Documentation : вебсайт. URL: <https://vercel.com/docs> (дата звернення: 10.06.2026).
- 13) Жидецький В.Ц., Джигирей В.С., Мельников О.В. Основи охорони праці. Львів : Афіша, 2000. 351 с.
- 14) Про затвердження Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці : вебсайт. URL:

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		76

<https://zakon.rada.gov.ua/laws/show/z0231-05> (дата звернення: 13.06.2026).

15) Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99 : вебсайт. URL: <https://zakon.rada.gov.ua/laws/show/va042282-99> (дата звернення: 13.06.2026).

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		77

ДОДАТКИ

Додаток А. Лістинг файлу «app/layout.tsx»

```
import type React from "react"
import type { Metadata } from "next"
import { Inter } from "next/font/google"
import "./globals.css"
import { ThemeProvider } from "@components/theme-provider"
import Header from "@components/layout/header"
import Footer from "@components/layout/footer"
import { AuthProvider } from "@components/auth/auth-provider"
import { CartProvider } from "@lib/cart-context"
const inter = Inter({ subsets: ["latin", "cyrillic"] })
export const metadata: Metadata = {
  title: "BookSwap - Маркетплейс вживаних книг",
  description: "Купуйте та продавайте вживані книги онлайн",}
export default function RootLayout({ children,
}: Readonly<{ children: React.ReactNode}>) {
  return (
    <html lang="uk" suppressHydrationWarning>
      <body className={` ${inter.className} min-h-screen flex flex-col`} >
        <ThemeProvider attribute="class" defaultTheme="light" enableSystem
disableTransitionOnChange>
          <AuthProvider>
            <CartProvider>
              <Header />
              <main className="flex-grow">{children}</main>
              <Footer />
            </CartProvider>
          </AuthProvider>
        </ThemeProvider>
      </body>
    </html>  )}
```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		78

Додаток Б. Лістинг файлу «app/auth/login/page.tsx»

```
"use client"
import type React from "react"
import { useState, Suspense } from "react"
import { useRouter, useSearchParams } from "next/navigation"
import Link from "next/link"
import { Button } from "@components/ui/button"
import { Input } from "@components/ui/input"
import { Label } from "@components/ui/label"
import { Card, CardContent, CardDescription, CardFooter, CardHeader,
CardTitle } from "@components/ui/card"
import { Alert, AlertDescription } from "@components/ui/alert"
import { signIn } from "next-auth/react"
function LoginForm() {
  const [email, setEmail] = useState("")
  const [password, setPassword] = useState("")
  const [error, setError] = useState("")
  const [isLoading, setIsLoading] = useState(false)
  const router = useRouter()
  const searchParams = useSearchParams()
  const registered = searchParams.get("registered")
  const handleSubmit = async (e: React.FormEvent) => {
    e.preventDefault()
    setError("")
    setIsLoading(true)
    try {
      const result = await signIn("credentials", {
        email,
        password,
        redirect: false,
      })
      if (result?.error) {
        setError("Невірний email або пароль")
      }
    }
  }
}
```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		79


```

        required
        placeholder="your@email.com"
    />          </div>          <div>
<Label htmlFor="password">Пароль</Label>
<Input
    id="password"
    type="password"
    value={password}
    onChange={(e) => setPassword(e.target.value)}
    required
    placeholder="....."
  />
</div>
<Button type="submit" className="w-full" disabled={isLoading}>
  {isLoading ? "Вхід..." : "Увійти"}
</Button>
</form>
</CardContent>
<CardFooter className="flex justify-center">
  <p className="text-sm text-gray-600">
    Немає акаунту?{" "}
    <Link href="/auth/register" className="text-blue-600
hover:text-blue-800">
      Зареєструватися    </Link>
  </p>
</CardFooter>
</Card> </div>
)
}export default function LoginPage() {
  return (
    <Suspense fallback={<div className="container mx-auto px-4 py-16 text-
center">Завантаження...</div>}>
      <LoginForm />
    </Suspense>
  )
}

```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		81

Додаток В. Лістинг файлу «book-card.tsx»

```
"use client"
import Link from "next/link"
import Image from "next/image"
import type { Book } from "@prisma/client"
import { Button } from "@components/ui/button"
import { ShoppingCart, Edit, Trash2, AlertCircle, CheckCircle, XCircle }
from "lucide-react"
import { Badge } from "@components/ui/badge"
import { useCart } from "@lib/cart-context"
import { useSession } from "next-auth/react"
import { useState } from "react"
type BookCardProps = {
  book: Book
  featured?: boolean
  onEdit?: (book: Book) => void
  onDelete?: (book: Book) => void
  showStatus?: boolean
}
export default function BookCard({ book, featured = false, onEdit, onDelete,
showStatus = false }: BookCardProps) {
  const { addItem } = useCart()
  const { data: session } = useSession()
  const [imageError, setImageError] = useState(false)
  const conditionColorMap: Record<string, string> = {
    "NEW": "bg-green-100 text-green-800",
    "GOOD": "bg-blue-100 text-blue-800",
    "FAIR": "bg-yellow-100 text-yellow-800",
    "POOR": "bg-red-100 text-red-800",
  }
  const conditionTextMap: Record<string, string> = {
    "NEW": "Як нова",
    "GOOD": "Гарний",
```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		82

```

    "FAIR": "З потертостями",
    "POOR": "Пошкоджена",
  }
  const statusColorMap: Record<string, string> = {
    "APPROVED": "bg-green-100 text-green-800",
    "PENDING": "bg-yellow-100 text-yellow-800",
    "REJECTED": "bg-red-100 text-red-800",
  }
  const statusIconMap: Record<string, React.ReactNode> = {
    "APPROVED": <CheckCircle className="h-4 w-4 mr-1" />,
    "PENDING": <AlertCircle className="h-4 w-4 mr-1" />,
    "REJECTED": <XCircle className="h-4 w-4 mr-1" />,
  }
  const statusTextMap: Record<string, string> = {
    "APPROVED": "Схвалено",
    "PENDING": "На розгляді",
    "REJECTED": "Відхилено",
  }
  const conditionColor = conditionColorMap[book.condition as string] || "bg-
  gray-100 text-gray-800"
  const conditionText = conditionTextMap[book.condition as string] ||
  book.condition

  let isOwner = false
  if (session?.user && (session.user as any).id) {
    isOwner = parseInt((session.user as any).id) === book.sellerId
  }
  const getImageSrc = () => {
    const bookAny = book as any
    if (imageError || !bookAny.images || bookAny.images.length === 0) {
      return "/abstract-book-cover.png"
    }
    return bookAny.images[0].url
  }
}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		83

```

return (
  <div
    className={`bg-white border border-gray-200 rounded-lg overflow-hidden
transition-shadow hover:shadow-md ${featured ? "h-full" : ""}`}
  >
    <Link href={` /books/${book.id}`} className="block">
      <div className="relative aspect-[3/4] overflow-hidden bg-gray-50">
        <Image
          src={getImageSrc() || "/placeholder.svg"}
          alt={`Обкладинка книги "${book.title}"`}
          fill
          className="object-cover"
          onError={() => setImageError(true)}
        />
      </div>
    </Link>
    <div className="p-4">
      <div className="flex justify-between items-start mb-2">
        <Link href={` /books/${book.id}`} className="block">
          <h3 className="font-semibold text-gray-900 hover:text-gray-700
line-clamp-2">{book.title}</h3>
        </Link>
        <Badge variant="outline" className={`ml-2 shrink-0
${conditionColor}`}>
          {conditionText}
        </Badge>
      </div>
      <p className="text-gray-600 text-sm mb-1">{(book as
any).author?.name || "Невідомий автор"}</p>
      <p className="text-gray-500 text-xs mb-3">
        {book.year} • {(book as any).genre?.name || "Без жанру"}
      </p>
      {featured && <p className="text-gray-600 text-sm mb-4 line-clamp-
2">{book.description}</p>}
    </div>
  </div>
)

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		84

```

    {showStatus && (
      <div className="mb-3">
        <Badge                                variant="outline"
className={` ${statusColorMap[book.status as string]} flex items-center w-
fit`} >
          {statusIconMap[book.status as string]}
          {statusTextMap[book.status as string]}
        </Badge>
      </div>
    )}
    <div className="flex items-center justify-between mt-2">
      <span className="text-lg font-bold text-gray-900">{book.price}
    </span>
      <div className="flex space-x-2">
        {isOwner ? (
          // Owner buttons
          <>
            <Button variant="outline" size="sm" onClick={() =>
onEdit?.(book)} >
              <Edit className="h-4 w-4 mr-1" />
              Редагувати
            </Button>
            <Button
              variant="outline"
              size="sm"
              onClick={(e) => {
                e.preventDefault();
                onDelete?.(book);
              }}
              className="text-red-600 hover:text-red-700"
            >
              <Trash2 className="h-4 w-4" />
            </Button>
          </>
        )}
      </div>
    </div>
  </div>

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		85

```

    ) : (
      // Buyer buttons
      <>
        <Button variant="outline" size="sm" asChild>
          <Link href={` /books/${book.id}`}>Деталі</Link>
        </Button>
        {book.status === "approved" && (
          <Button
            size="sm"
            onClick={(e) => {
              e.preventDefault()
              addItem(book)
            }}
          >
            <ShoppingCart className="h-4 w-4 mr-1" />В кошик
          </Button>
        )}
      </>
    )}
  </div>
</div>
</div>
</div>
)
}

```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		86

Додаток Г. Лістинг файлу «actions/book.ts»

```
"use server"
```

```
import { prisma } from "@lib/prisma"
```

```
import { revalidatePath } from "next/cache"
```

```
import { BookCondition, BookStatus } from "@prisma/client"
```

```
import { z } from "zod"
```

```
const bookSchema = z.object({
```

```
  title: z.string().min(1, "Назва обов'язкова"),
```

```
  author: z.string().min(1, "Автор обов'язковий"),
```

```
  year: z.number().int().min(1000).max(new Date().getFullYear()),
```

```
  genre: z.string().min(1, "Жанр обов'язковий"),
```

```
  price: z.number().min(0),
```

```
  condition: z.enum([BookCondition.NEW, BookCondition.GOOD,  
BookCondition.FAIR, BookCondition.POOR]),
```

```
  description: z.string().min(10, "Опис повинен бути принаймні 10  
символів"),
```

```
  images: z.array(z.string()).min(1, "Додайте хоча б одне зображення"),
```

```
  isbn: z.string().optional().nullable(),
```

```
  sellerId: z.number(),
```

```
})
```

```
export async function getApprovedBooks(params?: {
```

```
  page?: number
```

```
  limit?: number
```

```
  genre?: string
```

```
  minPrice?: number
```

```
  maxPrice?: number
```

```
  condition?: string
```

```
  search?: string
```

```
  sortBy?: string
```

```
}) {
```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		87

```

const {
  page = 1,
  limit = 12,
  genre,
  minPrice,
  maxPrice,
  condition,
  search,
  sortBy = "title-asc"
} = params || {}

const skip = (page - 1) * limit

const where: any = { status: BookStatus.APPROVED }
if (genre) where.genre = { name: genre }
if (minPrice !== undefined || maxPrice !== undefined) {
  where.price = {}
  if (minPrice !== undefined) where.price.gte = minPrice
  if (maxPrice !== undefined) where.price.lte = maxPrice
}
if (condition) {
  const enumCondition = Object.values(BookCondition).find(c => c ===
condition)
  if (enumCondition) where.condition = enumCondition
}

if (search) {
  where.OR = [
    { title: { contains: search } },
    { author: { name: { contains: search } } },
  ]
}

let orderBy: any = {}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		88

```

switch (sortBy) {
  case "title-asc": orderBy = { title: "asc" }; break;
  case "title-desc": orderBy = { title: "desc" }; break;
  case "price-asc": orderBy = { price: "asc" }; break;
  case "price-desc": orderBy = { price: "desc" }; break;
  case "year-asc": orderBy = { year: "asc" }; break;
  case "year-desc": orderBy = { year: "desc" }; break;
  default: orderBy = { createdAt: "desc" };
}

const [books, total] = await Promise.all([
  prisma.book.findMany({
    where,
    orderBy,
    skip,
    take: limit,
    include: {
      author: true,
      genre: true,
      images: true,
    }
  }),
  prisma.book.count({ where })
])

return {
  books,
  totalPages: Math.ceil(total / limit),
  currentPage: page,
  totalCount: total
}
}

export async function getFeaturedBooks() {

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		89

```

return await prisma.book.findMany({
  where: { status: BookStatus.APPROVED },
  take: 4,
  orderBy: { createdAt: "desc" },
  include: { author: true, genre: true, images: true }
})
}

export async function getNewBooks() {
  return await prisma.book.findMany({
    where: { status: BookStatus.APPROVED },
    take: 8,
    orderBy: { createdAt: "desc" },
    include: { author: true, genre: true, images: true }
  })
}

export async function getBookById(id: number) {
  return await prisma.book.findUnique({
    where: { id },
    include: {
      author: true,
      genre: true,
      images: true,
      seller: {
        select: {
          name: true,
          email: true,
        },
      },
    },
  })
}
}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						90
Зм.	Арк.	№ докум.	Підпис	Дата		

```

export async function createBook(formData: z.infer<typeof bookSchema>) {
  try {
    const validatedData = bookSchema.parse(formData)

    // Check or create author
    let author = await prisma.author.findUnique({ where: { name:
validatedData.author } })
    if (!author) {
      author = await prisma.author.create({ data: { name:
validatedData.author } })
    }

    // Check or create genre
    let genre = await prisma.genre.findUnique({ where: { name:
validatedData.genre } })
    if (!genre) {
      genre = await prisma.genre.create({ data: { name: validatedData.genre
} })
    }

    const book = await prisma.book.create({
      data: {
        title: validatedData.title,
        year: validatedData.year,
        price: validatedData.price,
        condition: validatedData.condition,
        description: validatedData.description,
        isbn: validatedData.isbn,
        status: BookStatus.PENDING,
        sellerId: validatedData.sellerId,
        authorId: author.id,
        genreId: genre.id,
        images: {
          create: validatedData.images.map(url => ({ url })))
        }
      }
    })
  }
}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		91

```

    }
  },
  include: { author: true, genre: true, images: true }
})

revalidatePath("/")
revalidatePath("/books")
return { success: true, book }
} catch (error) {
  console.error("Error creating book:", error)
  if (error instanceof z.ZodError) {
    return { error: error.errors[0].message }
  }
  return { error: "Помилка при додаванні книги" }
}
}

export async function updateBook(id: number, formData: z.infer<typeof
bookSchema>) {
  try {
    const validatedData = bookSchema.parse(formData)

    // Check or create author
    let author = await prisma.author.findUnique({ where: { name:
validatedData.author } })
    if (!author) {
      author = await prisma.author.create({ data: { name:
validatedData.author } })
    }

    // Check or create genre
    let genre = await prisma.genre.findUnique({ where: { name:
validatedData.genre } })
    if (!genre) {

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
						92
Зм.	Арк.	№ докум.	Підпис	Дата		

```

    genre = await prisma.genre.create({ data: { name: validatedData.genre
} })
}

const book = await prisma.book.update({
  where: { id },
  data: {
    title: validatedData.title,
    year: validatedData.year,
    price: validatedData.price,
    condition: validatedData.condition,
    description: validatedData.description,
    isbn: validatedData.isbn,
    authorId: author.id,
    genreId: genre.id,
    // For simplicity, we only update images if new ones are provided
    ...(validatedData.images.length > 0 &&
!validatedData.images.includes("/book-cover-default.jpg") && {
      images: {
        deleteMany: {},
        create: validatedData.images.map(url => ({ url }))
      }
    })
  },
  include: { author: true, genre: true, images: true }
})

revalidatePath("/")
revalidatePath("/books")
revalidatePath(`/books/${id}`)
revalidatePath("/account/my-books")
return { success: true, book }
} catch (error) {
  console.error("Error updating book:", error)
}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		93

```

    if (error instanceof z.ZodError) {
      return { error: error.errors[0].message }
    }
    return { error: "Помилка при оновленні книги" }
  }
}

export async function updateBookStatus(id: number, status: BookStatus) {
  try {
    await prisma.book.update({
      where: { id },
      data: { status },
    })
    revalidatePath("/admin")
    revalidatePath("/books")
    return { success: true }
  } catch (error) {
    return { error: "Помилка при оновленні статусу" }
  }
}

export async function getBooksBySeller(sellerId: number) {
  return await prisma.book.findMany({
    where: { sellerId },
    orderBy: { createdAt: "desc" },
    include: { author: true, genre: true, images: true }
  })
}

export async function deleteBook(id: number) {
  try {
    // Check if book is part of any orders
    const orderItem = await prisma.orderItem.findFirst({
      where: { bookId: id }
    })
  }
}

```

					2026.KBP.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		94

```

    })
    if (orderItem) {
        return { error: "Неможливо видалити книгу, яка є частиною замовлення. Ви можете змінити її статус на 'Продано' або інший." }
    }
    await prisma.book.delete({
        where: { id },
    })
    revalidatePath("/account/my-books")
    revalidatePath("/books")
    revalidatePath("/")
    return { success: true }
} catch (error) {
    console.error("Error deleting book:", error)
    return { error: "Помилка при видаленні книги. Можливо, вона пов'язана з іншими даними." }
}
}

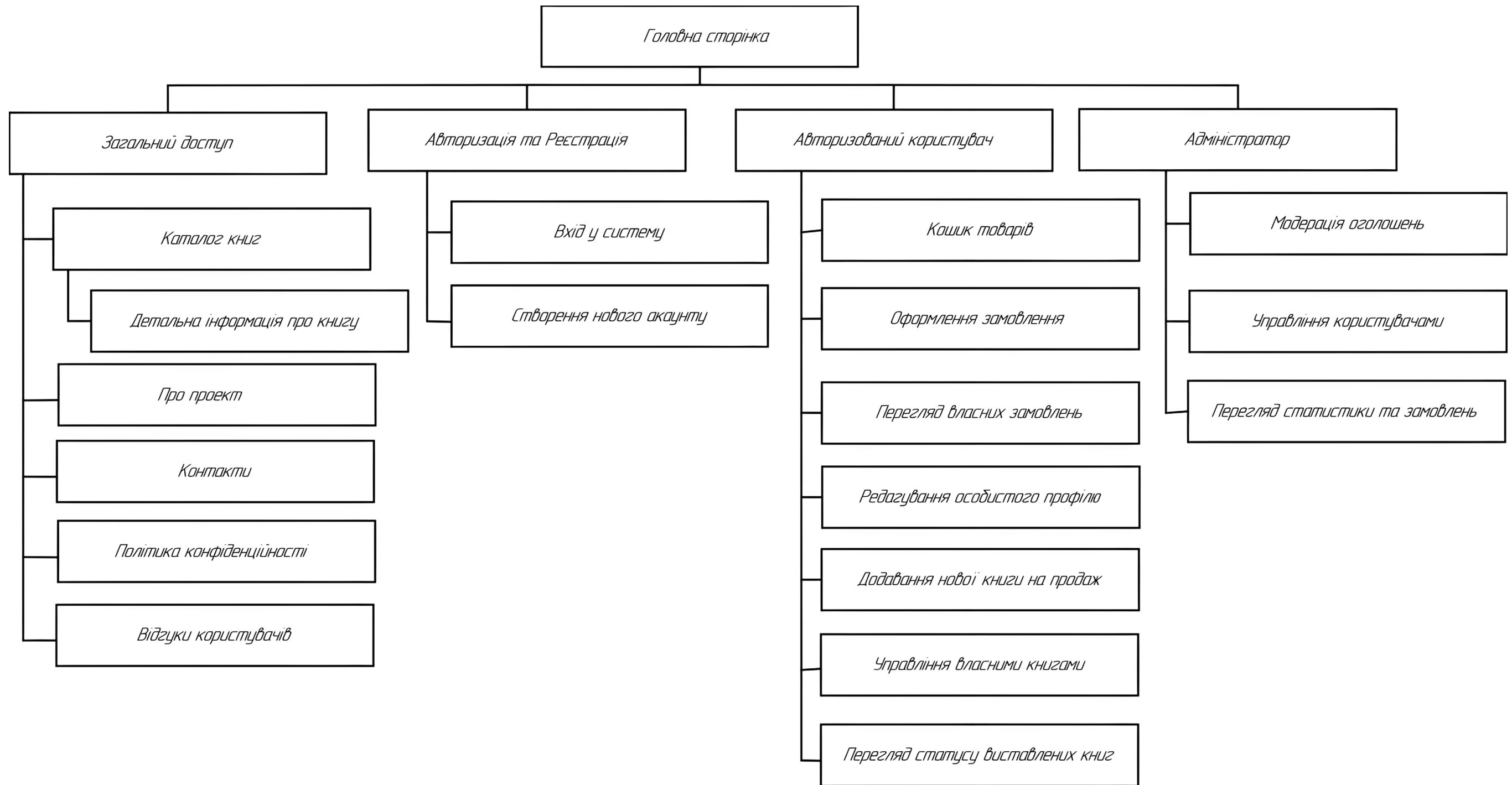
export async function getPendingBooks() {
    return await prisma.book.findMany({
        where: { status: BookStatus.PENDING },
        orderBy: { createdAt: "desc" },
        include: { author: true, genre: true, images: true, seller: true }
    })
}

export async function getRejectedBooks() {
    return await prisma.book.findMany({
        where: { status: BookStatus.REJECTED },
        orderBy: { createdAt: "desc" },
        include: { author: true, genre: true, images: true, seller: true }
    })
}
}

```

					2026.КВР.122.423.01.00.00 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		95

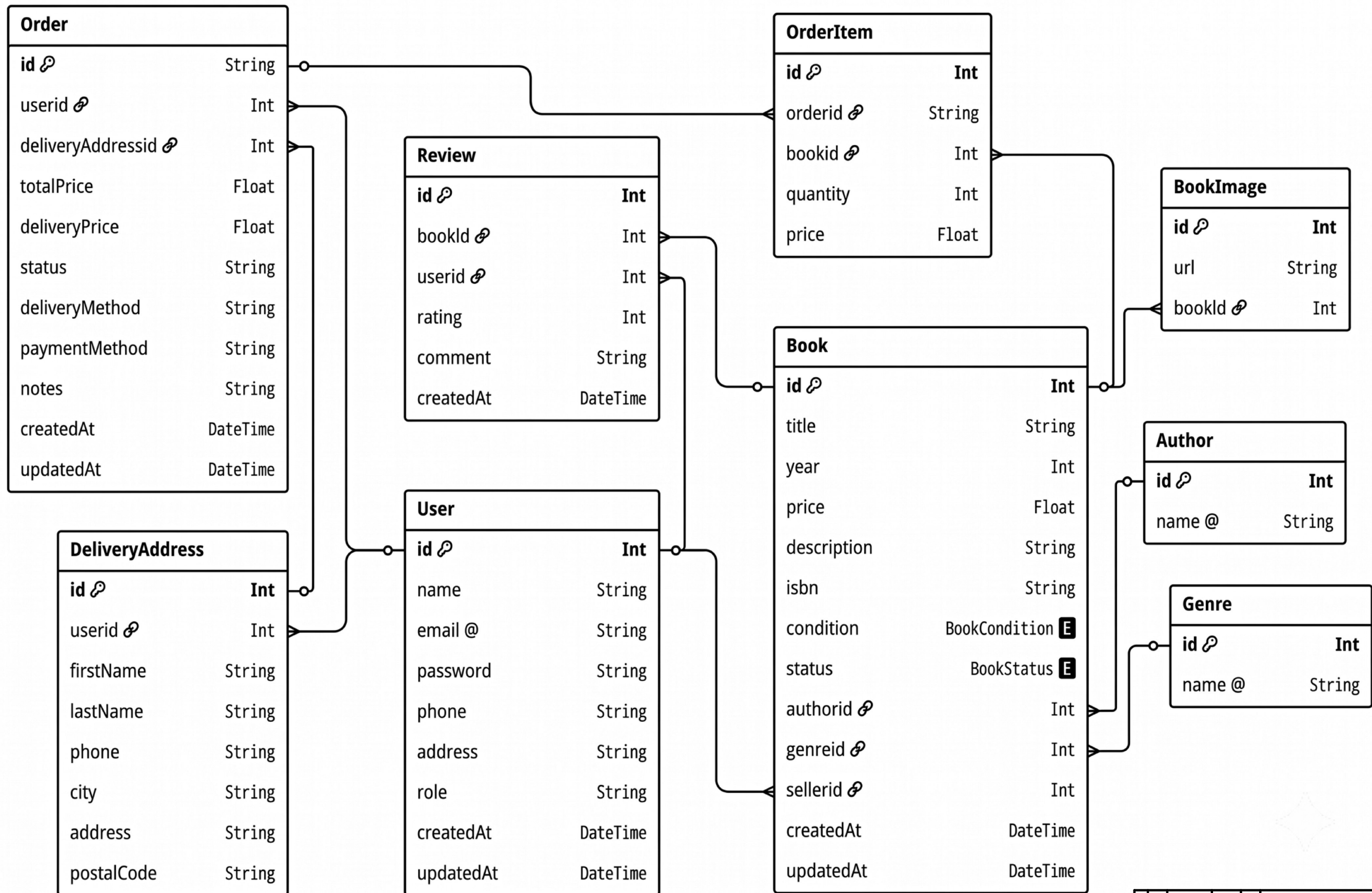
Схема структурна



Лист, примірник
Сторінка №
Листів у ділянці
Всього листів №
Листів у ділянці
Листів №

				2026.KBP.122.423.01.00.00 CC				
Изм.	Лист	№ вказ.	Лист	Листа	Разработка интернет магазина вживаних книг «BookSwap» Схема структурна	Лист	Масштаб	Масштаб
Разраб.		Богодан ОІ						
Проб.		Сердін СВ						
Текст.								
Рецензент								
Иконтр.		Пріймак В.А.						
Чит.								
						Лист	Листів	1
						ВСП ТФК ТНТУ	КН-423	
						м. Тернопіль		
						Копірабат	Формат А1	

ER-діаграма бази даних



Лист № 1
Лист № 2
Лист № 3
Лист № 4
Лист № 5
Лист № 6
Лист № 7
Лист № 8
Лист № 9
Лист № 10

UML – Діаграма варіантів використання



Лист № 1
Лист № 2
Лист № 3
Лист № 4
Лист № 5
Лист № 6
Лист № 7
Лист № 8
Лист № 9
Лист № 10
Лист № 11
Лист № 12
Лист № 13
Лист № 14
Лист № 15
Лист № 16
Лист № 17
Лист № 18
Лист № 19
Лист № 20

				2026.KBP.122.4.23.01.00.00 ДВ			
Имя	Лист	№ докум.	Лист	Дата	Разработка интернет магазина	Лист	Масштаб
Разработ		Богодан О.И.			«BookSwamp»		
Проект		Сергей С.В.			UML-Діаграма варіантів використання	Лист	Листов 1
Участник							
Рецензент							
Исполнитель		Приймак В.А.					
Чит							
Копировал						Формат А1	

Таблиця техніко-економічних показників

№	Показник	Одиниця вимірювання	Значення
1	Мова розрамування	-	TypeScript
2	Фреймворк	-	Next.js v16
3	Кількість таблиць у БД	шт	11
4	База даних	-	MySQL
5	Бібліотека стилізації	-	TailwindCSS, Radix UI
1	Собівартість	грн	92 201
2	Плановий грошовий потік	грн	51 632
3	Ціна	грн	143 229
4	Чиста теперішня вартість	грн	36229
5	Термін окупності	рік	2,1

				2026.KBP.122.423.01.00.00 ТБ			
Изм.	Лист	№ вказ.	Лист	Листа	Розробка інтернет магазину вживаних книг «BookSwag»	Лист	Масштаб
Розроб.		Богодан О.І.			Таблиця техніко-економічних показників		
Проб.		Сердюк С.В.				Лист	Листа 1
Ілюстр.							
Рецензент							
Ілюстр.		Приймак В.А.				БСП ТФК ТНТУ	КН-423
Чит.						м. Тернопіль	
				Копіював	Формат А1		