

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,  
управління та адміністрування  
Кафедра інформаційних технологій

**Кваліфікаційна робота бакалавра**

на тему: Розробка веб-сервісу «Книжкова крамниця»

---

Виконав студент групи КН-20  
спеціальності 122 Комп'ютерні науки  
Костишин Андрій Юрійович

Керівник к.т.н., доцент  
Гнатовська Ганна Арнольдівна

Рецензент к.геогр.н., доцент  
Кузніченко Світлана Дмитрівна

Одеса 2023

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ТЕРМІНІВ І ПОЗНАЧЕНЬ .....	6
ВСТУП.....	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ ...	9
1.1 Дослідження предмета, цілей і особливостей додатку для книжкової крамниці .....	9
1.2 Аналіз існуючих систем.....	11
1.3 Переваги та особливості використання електронної комерції для реалізації книжкової продукції.....	15
1.4 Принцип функціонування книжкових інтернет-крамниць .....	16
1.5 Постановка завдання .....	17
2 ВИБІР АРХІТЕКТУРИ ТА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ ВЕБ-СЕРВІСУ .....	21
2.1 Загальна архітектура веб-сервісу.....	21
2.2 Вибір Web-сервера.....	23
2.3 Вибір системи управління базами даних.....	25
2.4 Вибір мови програмування.....	28
2.4.1 Мова сценаріїв PHP.....	28
2.4.2 Мова сценаріїв JavaScript.....	31
2.4.3 Мова розмітки гіпертексту HTML .....	33
2.4.4 Мова розмітки CSS .....	34
3 ПРОЕКТУВАННЯ ВЕБ-СЕРВІСУ «КНИЖКОВА КРАМНИЦЯ» .....	36
3.1 Проектування веб-додатку за допомогою методології SADT .....	36
3.1.1 Використання мови IDEF0 для проектування веб-додатку.....	36
3.2 Проектування веб-додатку за допомогою нотації IDEF3 .....	42
3.3 Проектування веб-додатку за допомогою нотації DFD.....	45
3.4 Проектування функціональних вимог і бази даних додатку .....	48
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБ-ДОДАТКУ «КНИЖКОВА КРАМНИЦЯ» .....	57
4.1 Керівництво користувача-клієнта.....	57
4.1.1 Загальні функції інтерфейсу системи.....	57
4.1.2 Реалізація механізму авторизації у системі.....	59

	5
4.1.3 Складові інтерфейсу користувача.....	61
4.1.4 Організація пошуку у додатку .....	67
4.2 Керівництво користувача-адміністратора .....	69
4.2.1 Реалізація механізму авторизації у системі.....	69
4.2.2 Складові інтерфейсу-адміністратора .....	70
ВИСНОВКИ.....	76
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ .....	77

## ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ТЕРМІНІВ І ПОЗНАЧЕНЬ

IT – інформаційні технології.

Хостинг – послуга, яка надає можливість для розміщення фізичної інформації.

CSS (Cascading Style Sheets) – каскадні таблиці стилів.

HTML (Hypertext Markup Language) – мова розмітки гіпертексту.

HTTP (Hypertext Transfer Protocol) – протокол передачі гіпертексту.

JS (JavaScript) – мова програмування для створення динамічного веб-контенту.

JSX (JavaScript XML) – розширення JavaScript, що дозволяє використовувати синтаксис XML в JavaScript-кодi.

PPC (Pay-Per-Click) – система рекламного збору плати за кожне клікання.

SADT (Structured Analysis and Design Technique) – метод структурованого аналізу та проектування.

SEO (Search Engine Optimization) – оптимізація пошукової системи.

WEB (World Wide Web) – глобальна мережа, що з'єднує мільйони веб-сторінок та додатків.

XML (eXtensible Markup Language) – розширювана мова розмітки.

## ВСТУП

У сучасному світі Інтернет став невід'ємною частиною нашого життя. Він відкриває безмежні можливості для розвитку бізнесу та взаємодії зі споживачами. Віртуальні магазини стали нормою для багатьох людей, а книжкові крамниці не є винятком.

Сьогодні ефективна взаємодія між бізнесменами та клієнтами відбувається за допомогою Інтернету. Людям зручно купувати та продавати товари в Інтернеті через простоту, з якою завдяки простоті, з якою це робиться. Веб-сайти, що стосуються купівлі-продажу в Інтернеті, набувають все більшої популярності завдяки швидкому розвитку технологій [1].

Електронна комерція актуальна практично для будь-якого типу бізнесу: усі процеси можна оптимізувати та налаштувати канали збуту, залучити цільову аудиторію, налаштувати SEO, PPC (Google або Facebook Ads), YouTube, TikTok та інші. Однак електронна комерція – не лише для фізично існуючих товарів. Тип торгівлі може набувати різних форм [2]:

- роздрібна торгівля B2C;
- оптовий продаж B2B;
- реалізація нематеріальних послуг.
- дропшипінг – процес продажу, в якому бере участь третя сторона;
- краудфандинг – збір грошей зі споживачів до того, як товар вийшов на ринок (з метою отримання стартового капіталу для подальшої появи на ринку). Приклад – Kickstarter;
- підписка – автоматична повторювана покупка через певні проміжки часу (SaaS);
- продаж цифрових продуктів – програми, додатки, онлайн-курси та інше, де клієнт завантажує або встановлює цифровий продукт (Udemy і Apple Store);

- товари Б/У (eBay);
- товари унікального виробництва (Etsy).

У межах одного бізнесу часто комбінуються кілька форм електронних транзакцій – особливо, якщо ви надаєте і товари, і послуги.

Метою даної бакалаврської кваліфікаційної роботи є створення веб-сервісу «Книжкова крамниця», який дозволить користувачам зручно і швидко замовляти та придбати засобами мережі Інтернет книги он-лайн. Основні функції, які має виконувати веб-сервіс, полягають у можливості перегляду каталогу книг, пошуку книг за різними параметрами, замовленні книг, здійснення оплати замовлення та отримання інформації про стан виконання замовлення, що дозволить реалізувати можливості продажу та розповсюдження книжок засобами мережі Інтернет.

Створення веб-сервісу «Книжкова крамниця» має на меті спростити процес купівлі книг, забезпечити швидкий та зручний доступ до потрібних видань та покращити якість обслуговування клієнтів.

Дипломна робота містить в собі 78 сторінки, 12 таблиць, 47 рисунків, 13 посилань.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

### 1.1 Дослідження предмета, цілей і особливостей додатку для книжкової крамниці

Книжкові інтернет-крамниці працюють у сфері онлайн-продажу книжкової продукції. Цей бізнес не є дуже успішним, зважаючи на конкуренцію, спричинену присутністю на ринку великих гравців. Однак кожна книжкова інтернет-крамниця має власну клієнтську базу, яку можна охопити за допомогою добре розроблених і зручних для користувача веб-додатків.

Мета інтернет-крамниці – мінімізувати витрати і максимізувати прибуток. Необхідно знайти найефективніший спосіб управління інформацією про постійних клієнтів, товари в каталозі, замовлені товари, персонал і постачальників. Необхідно вирішити такі завдання, як реєстрація замовлень, розрахунок витрат, виставлення рахунків, створення каталогів та облік платежів клієнтів.

Відповідно до системного підходу, інтернет-крамниця – це сукупність об'єктів, які взаємодіють один з одним для досягнення певних цілей [2].

Інтернет-крамниця складається з наступних елементів:

- веб-додаток;
- система обробки замовлень;
- система доставки товару;
- система роботи з постачальниками.

Основні аспекти діяльності інтернет-крамниці:

- прийом персоналу на роботу;
- закупка товару;
- замовлення товарів клієнтами;
- повернення пошкодженого товару;
- повернення товару, якщо не сподобався;
- сплата податків;

– підтримка та оновлення додатку.

Замовити продукцію можна наступним чином. Каталог продукції опублікований у додатку крамниці. Тільки зареєстровані користувачі можуть обирати та замовляти товари з каталогу. Товари можна замовити через додаток або за телефоном. Замовлення пакуються на складі і доставляються клієнту обраним способом (кур'єром або поштою) [2].

Якщо товар надійшов пошкодженим, його можна повернути або обміняє товар на інший товар тієї ж вартості. Магазин огляне товар і поверне гроші або поміняє товар або відмовить у поверненні. Якщо вам не сподобався товар, ви можете повернути його протягом 30 днів або обміняти на інший товар тієї ж вартості.

Оскільки велику кількість замовлень потрібно обробляти одночасно, може бути зручно використовувати автоматизацію процесів. Коли надходить замовлення, автоматизована торгова система перевіряє дані про наявність товару на складі без залучення персоналу і повідомляє клієнта про те, що дата доставки може бути змінена, якщо його немає в наявності.

Останнім часом з'являється все більше платіжних систем, які полегшують рух грошей в Інтернеті. Загалом, існує два основних способи оплати товарів, придбаних в інтернет-магазинах: оплата при отриманні товару та передоплата. Передоплата передбачає оплату шляхом переказу грошей через платіжну систему.

Система доставки для інтернет-магазинів є одним з найважливіших структурних елементів. З одного боку, це додатковий сервіс для покупця, з іншого - додатковий ризик (терміни доставки, проблеми з надійністю, додаткові витрати). Тому цей аспект вимагає особливої уваги до послуг доставки. Наразі існує два види доставки: кур'єрська та поштова [2].



## 1.2 Аналіз існуючих систем

Для того щоб сформулювати вимоги функцій, які повинен надавати веб-сервіс «Книжкова крамниця» потрібно здійснити аналіз існуючих додатків та визначити їх слабкі та сильні сторони. Розглянемо декілька таких прикладів.

«Yakaboo.ua» – найбільший інтернет-магазин книг, дитячих товарів і подарунків в Україні. На головній сторінці користувач може побачити новинки та новини магазину. Головна сторінка представлена на рисунку 1.1.

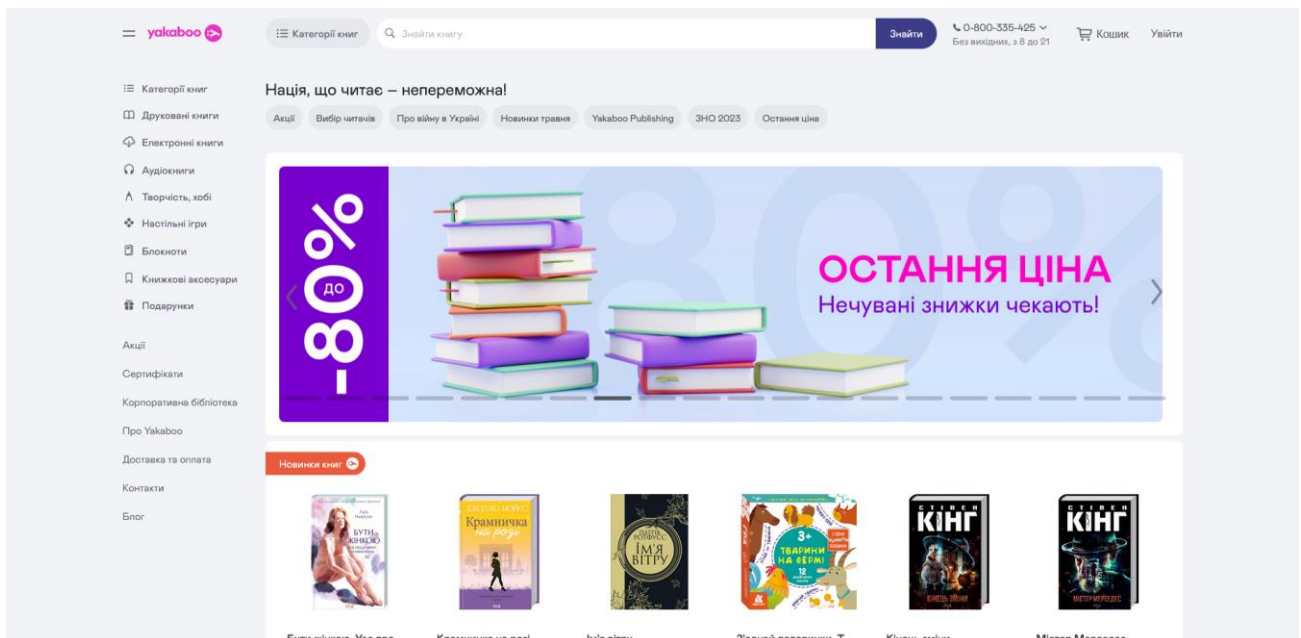


Рисунок 1.1 – Головна сторінка магазину «Yakaboo.ua»

Перевагами цього додатку є:

- широкий асортимент товарів;
- зручний дизайн ПК-версії;
- зручний пошук.

Недоліками цього додатку є:

- дизайн на мобільних пристроях може бути некоректним;

- на деяких сторінках сайту використовуються шрифти, які можуть бути складні для читання;
- на деяких сторінках дизайну використовуються надмірні ефекти або графічні елементи, що можуть зменшувати швидкість завантаження сторінок.

Наступний додаток, який аналізується – це «grenka.ua».

«Grenka.ua» – найбільший інтернет магазин книг, подарунків, ігор в Україні. На головній сторінці можна побачити слайдер з новинами, під слайдером розміщений блок з лічильниками. Також є блок з новинками та бестселерами (рис. 1.2).



Рисунок 1.2 – Головна сторінка додатку «grenka.ua»

До переваг додатку цього Інтернет-магазину можливо віднести:

- наявність зворотного зв'язку;
- функціональний пошук;
- наявність декількох способів доставки.

Недоліки додатку:

- перевантажений дизайн;
- вузький макет;
- дублювання деяких елементів;
- деякі елементи некоректно відображаються.

Наступний приклад «Nashformat.ua».

«Nashformat.ua» – це онлайн-магазин, який пропонує широкий вибір книжок, електронних книг та аудіо-книг. Сайт працює в Україні та пропонує клієнтам швидку та зручну доставку товарів по всій країні. Головна сторінка представлена на рис. 1.3.

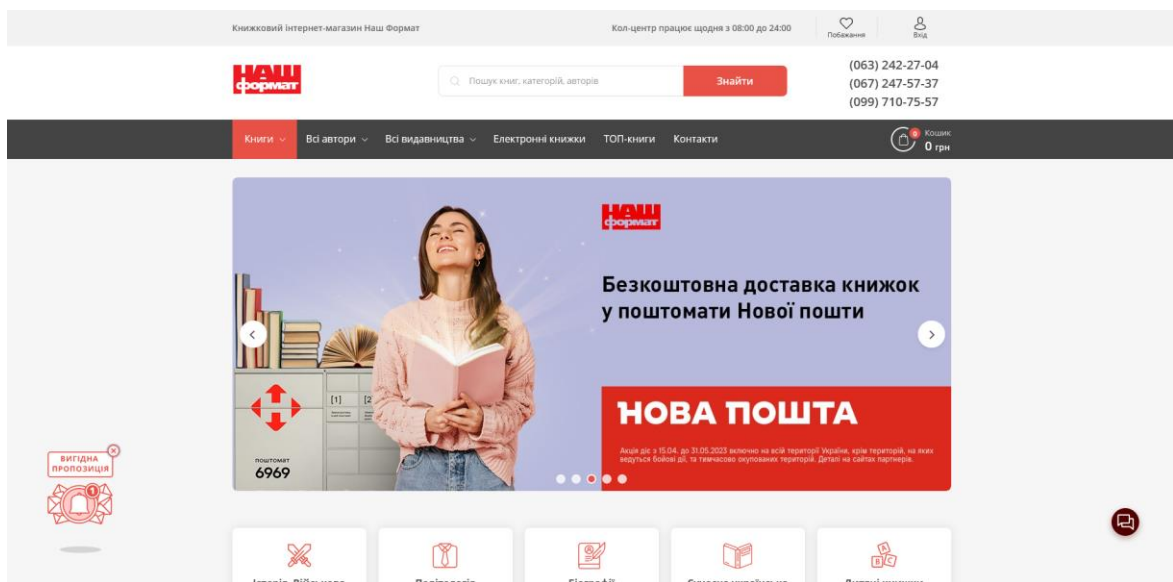


Рисунок 1.3 – Головна сторінка додатку «nashformat.ua»

При розгляді цього ресурсу буди відзначені наступні переваги:

- привабливий дизайн;
- зручна навігація;
- широкий функціонал.

Недоліками цього додатку є:

- на головній сторінці сайту може бути занадто багато інформації;
- деякі елементи дизайну можуть бути недостатньо виділені на фоні загального дизайну сайту;
- некоректний адаптивний дизайн;
- сайт може бути трохи повільним у відгуках.

У результаті проведеного аналізу існуючих додатків, з множини електронних реалізацій інтернет-крамниць перевага віддається тій, яка найбільшою мірою задовольняє потреби користувача. Потрібно урахувати зручність, дизайн, наявність адаптивного дизайну для мобільних пристроїв та швидкий відгук.

Виходячи з аналізу існуючих систем, можна виділити функції які повинен мати додаток «Книжкова крамниця»:

- легкий дизайн – додаток повинен мати привабливий та не перевантажений дизайн, для того щоб користувачу було комфортно ним користуватись;
- швидкість – додаток повинен бути швидким в роботі;
- адаптивний дизайн – додаток повинен коректно відображатись як на настільних системах, так і на мобільних
- функціональний пошук – додаток повинен мати зручний та функціональний пошук, щоб користувач міг легко знайти те що йому потрібно;
- зручна навігація – додаток повинен мати інтуїтивно зрозумілу навігацію;
- можливість реєстрації користувача – користувач повинен зареєструватися щоб зробити замовлення;
- наявність системи оплати – додаток повинен підтримувати декілька методів оплати;
- систему відгуків та рейтингів – додаток повинен мати систему рейтингів та відгуків;
- можливість попереднього перегляду книг.

### 1.3 Переваги та особливості використання електронної комерції для реалізації книжкової продукції

Торгівля включає купівлю, продаж та обмін товарами. Тому її можна визначити як обмін товарами або всю діяльність, пов'язану з переміщенням товарів від виробників до споживачів. Комерція була важливою частиною людського життя з самого початку історії. Впровадження Інтернету призвело до зміни парадигми ведення бізнесу сьогодні. Останнє десятиліття стало свідком появи нового виду комерції, відомого як електронна комерція.

Відповідно до веб-сайту Європейського Союзу, електронна комерція – це концепція, що стосується будь-якої форми ділових операцій або обміну інформацією, що здійснюється за допомогою інформаційно-комунікаційних технологій (ІКТ), між компаніями, компаніями та їхніми клієнтами, або компаніями та державними адміністраціями. Існує безліч додатків для електронної комерції, які постійно впливають на тенденції та перспективи бізнесу. Основними сферами застосування електронної комерції є бізнес-споживач (B2C), бізнес-бізнес (B2B), споживач-споживач (C2C) та споживач-бізнес (C2B). B2C – скорочення для позначення бізнесів, що продають товари та послуги клієнтам для особистого користування. Це може бути і великий гіпермаркет, і маленька перукарня або консультаційний центр. Сфера охоплює торгові точки, заклади харчування, послуги для споживачів (від прокату авто до салону краси), інтернет комерцію (платні онлайн послуги, інтернет-магазини і т.д.) [2].

Книжкова інтернет-крамниця відноситься до комерції типу бізнес-споживач (B2C).

Продаж книжкової продукції через Інтернет має важливі правові аспекти, зокрема, щодо авторського права. Інтернет надає можливість широкого розповсюдження та доступу до літературних творів, але це не звільняє авторів від дотримання правових норм, що захищають їхні права. Авторське право є невід'єм-

ною частиною онлайн-продажу книг. Згідно з авторським правом, автори мають виключне право використовувати, поширювати, передавати і продавати свої твори. Однак автори можуть передавати або ліцензувати ці права третім особам, наприклад, видавцям або книготорговцям в Інтернеті. Система онлайн-продажу книг складається з кількох етапів. По-перше, автор або видавець повинні мати права на продаж книги в електронному форматі. Якщо автор передав свої права видавництву, то видавництво може мати виключне право на продаж електронних копій твору. Автори, видавці, інтернет-магазини та інші продавці пропонують свої книги на власних веб-сайтах або платформах електронних книг. Потім клієнти можуть придбати книгу, зазвичай за допомогою електронної платіжної системи. Юридичні аспекти включають захист авторських прав і захист прав споживачів. Що стосується авторського права, важливо, щоб продавець мав дозвіл від автора або мав право продавати книгу. Продавець також повинен уникати порушення авторських прав. Порушенням авторських прав є несанкціоноване копіювання або розповсюдження книг. Для захисту прав споживачів важливо, щоб продавець чітко вказав умови продажу, включаючи ціну, умови повернення, інформацію про авторські права та ліцензійні обмеження. Важливо також забезпечити захист персональних даних покупців і вільний доступ до придбаних ними електронних книг [3].

#### 1.4 Принцип функціонування книжкових інтернет-крамниць

Книжкові інтернет-магазини працюють за принципом електронної комерції, тобто продажу товарів через Інтернет. Основна мета книжкового інтернет-магазину – запропонувати клієнтам можливість швидко і легко купувати книги та інші видання за допомогою онлайн-замовлень. Книжковий інтернет-додаток може надавати наступні послуги:

- на сайтах книжкових магазинів користувачі можуть знайти потрібну книгу, ввівши назву або автора книги;
- книгарні надають каталог книг, які можна класифікувати за жанром, автором, видавництвом тощо;
- користувачі можуть отримати інформацію про книги, таку як рейтинг, опис та обкладинка;
- користувач може додати вибрані книги до кошика і оформити замовлення, надавши необхідну інформацію, таку як адреса доставки та спосіб оплати;
- книги доставляються користувачеві поштою або кур'єром;
- користувач може повернути книгу, якщо вона не відповідає його очікуванням або є бракованою;
- книгарні можуть надавати клієнтську підтримку для вирішення будь-яких питань або проблем, пов'язаних із замовленням або доставкою.

Основною перевагою книжкових інтернет-магазинів є їхня зручність. Користувач може легко знайти потрібну книжку та оформити замовлення, не покидаючи дому чи офісу. Крім того, книжковий інтернет-магазин може запропонувати більший вибір книжок, ніж звичайна книжкова крамниця, що дає змогу знайти потрібну книжку в більшому розмаїтті.

### 1.5 Постановка завдання

Повинен бути створений веб-додаток у якості сервісу який забезпечує виконання наступних функцій, для різних категорій користувачів. В системі повинні буди такі види користувачів:

- користувач-адміністратор;
- користувач-гість;
- користувач-клієнт.

Вони повинні мати наступні функції.

Користувач-адміністратор:

- додавати, редагувати та видаляти категорії книг;
- додавати, редагувати та видаляти книг ізлюбих категорій;
- додавати, редагувати та видаляти новини інтернет-кramниці;
- додавати та видаляти користувачів додатку.

Користувач-клієнт:

- дивитись категорії книг або конкретну книгу;
- дивитись інформацію про книгу;
- авторизуватись у додатку;
- робити замовлення;
- виконувати пошук за категоріями;
- зв'язуватись з адміністрацією для вирішення питань.

Користувач-гість:

- дивитись категорії книг або конкретну книгу;
- дивитись інформацію про книгу;
- реєструватись у додатку;
- виконувати пошук за категоріями.

Додаток має бути клієнт-серверним застосунком. Він повинен включати в себе механізм пошуку, оплати, зворотного зв'язку. Користувачу повинна надаватись можливість попереднього перегляду книг.

Розробити систему категоризації та тегування книжок для полегшення пошуку. Забезпечити можливість перегляду детальної інформації про книжку (автор, видавництво, рейтинги, опис тощо). Забезпечити можливість додавання книжок до списку бажань або порівняння.

Розробити механізми пошуку за назвою, автором, жанром, ISBN та іншими параметрами. Забезпечити функцію автодоповнення для полегшення вве-



дення пошукових запитів. Реалізувати фільтри для точнішого обмеження результатів (ціна, рейтинг, новинки тощо).

Розробити функціонал корзини, де користувачі зможуть додавати книжки перед оформленням замовлення. Забезпечити можливість перегляду та редагування замовлення перед підтвердженням. Забезпечити різні методи оплати та доставки.

Реалізувати систему реєстрації та авторизації користувачів. Забезпечити персоналізований користувацький профіль з можливістю збереження історії замовлень, список бажань та власні рецензії. Дозволити користувачам залишати оцінки та коментарі до книжок, а також читати відгуки інших користувачів.

Розробити систему рекомендацій на основі попередніх покупок та історії переглядів користувачів. Забезпечити можливість персоналізованих рекомендацій на основі інтересів та вподобань користувача.

Розробити панель адміністратора для керування каталогом книжок, управління замовленнями та користувачами. Забезпечити можливість додавання, редагування та видалення книжок із каталогу. Забезпечити статистичний аналіз продажів, рейтингів та іншої корисної інформації для адміністратора.

Забезпечити коректну працю та функціонування як на веб-платформах (десктоп, мобільні браузері), так і на мобільних платформах (Android та iOS). Забезпечити адаптивний дизайн для оптимального відображення на різних розмірах екранів.

Застосувати надійні механізми захисту персональних даних користувачів та даних про оплату. Забезпечити шифрування передачі даних та захист від можливих атак на систему.

Інтегрувати додаток з популярними платіжними системами для забезпечення зручної та безпечної оплати замовлень. Забезпечити інтеграцію з провідними службами доставки для забезпечення ефективного та надійного доставлення книжок замовникам.

Провести широке тестування додатку з метою виявлення та виправлення помилок, а також оптимізації продуктивності та швидкодії. Забезпечити сумісність додатку з різними браузерами, операційними системами та пристроями.

## 2 ВИБІР АРХІТЕКТУРИ ТА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ ВЕБ-СЕРВІСУ

### 2.1 Загальна архітектура веб-сервісу

Сучасні веб-додатки досягли високого рівня розвитку, і термін і поняття «архітектура» можна застосовувати для грамотної побудови та проектування веб-додатків як єдиного цілого і забезпечення їхньої ефективної та надійної роботи.

Архітектура веб-додатка – це його концепції, що визначають моделі, структури, функції та взаємозв'язок компонентів.

Щоб побудувати правильну і надійну архітектуру, необхідно суворо дотримуватися останніх стандартів у цих галузях. Без цього ви, швидше за все, створите архітектуру, яка не зможе розвиватися і задовольняти зростаючі потреби користувачів ІТ.

Програмні системи можна класифікувати відповідно до їхньої архітектури таким чином:

- централізована архітектура;
- архітектури файлових серверів;
- дворівневі клієнт-серверні архітектури;
- багаторівневі клієнт-серверні архітектури;
- архітектури розподілених систем;
- архітектури веб-додатків;
- сервіс-орієнтована архітектура.

Трирівнева клієнт-серверна архітектура є однією з найпоширеніших архітектур в розробці програмного забезпечення. Її функція полягає у розподілі завдань між різними компонентами системи, щоб забезпечити ефективну обробку даних та зменшити навантаження на сервер [4].

У трирівневій клієнт-серверній архітектурі є три основні компоненти:

- клієнт (tier 1) – програмне забезпечення, яке виконується на стороні користувача та відповідає за взаємодію з сервером. Клієнт зазвичай надає інтерфейс користувача, щоб користувач міг взаємодіяти з системою;
- сервер (tier 2) – програмне забезпечення, яке виконується на стороні сервера та відповідає за зберігання та обробку даних. Сервер взаємодіє з клієнтом, щоб отримати та обробити запити користувачів;
- база даних (tier 3) – місце зберігання даних, до якого мають доступ як клієнт, так і сервер. База даних може бути локальною, розташованою на сервері або розподіленою по кількох серверах.

Переваги трирівневої клієнт-серверної архітектури включають [4]:

- розподілення завдань між різними компонентами системи зменшує навантаження на сервер та дозволяє більш ефективно використовувати ресурси;
- масштабованість – система може бути розширена шляхом додавання додаткових серверів або клієнтів, що дозволяє збільшувати її продуктивність;
- безпека – доступ до бази даних можна обмежити тільки до авторизованих користувачів;
- «тонкий» клієнт знижує вартість апаратного забезпечення робочої станції;

Трирівнева клієнт-серверна архітектура застосовується в багатьох різних системах, таких як веб-додатки, електронні поштові сервіси, онлайн ігри та багатокористувацькі програми. Вона дозволяє ефективно розподіляти завдання між компонентами системи та забезпечувати швидкий доступ до даних для користувачів. Схема трирівневої архітектури «Клієнт-сервер» представлена на рис. 2.1.

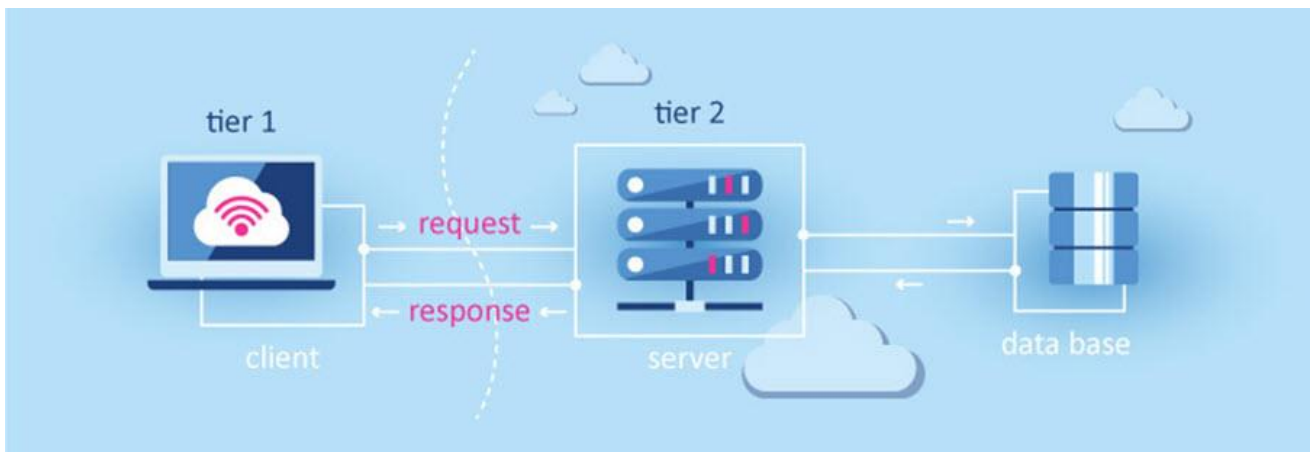


Рисунок 2.1 – Тривірнева архітектура «Клієнт сервер»

Вибір тривірневої клієнт-серверної архітектури для реалізації веб-додатку «Книжкова крамниця» обґрунтовується її перевагами в масштабованості, гнучкості, безпеці, легкій інтеграції та розподіленості. Ця архітектура дає змогу ефективно керувати навантаженням, забезпечувати модульність і розподіл відповідальності між компонентами системи, а також забезпечує безпеку та захист даних. Завдяки своїй гнучкості, тривірнева архітектура легко підтримує зміни та розвиток системи, а її розподіленість дає змогу забезпечити відмовостійкість і доступність системи для користувачів. У підсумку, тривірнева клієнт-серверна архітектура є надійним і ефективним вибором для розробки та розгортання веб-додатків.

## 2.2 Вибір Web-сервера

Для розробки додатку «Книжкова крамниця» був обраний веб-сервер Apache.

HTTP-сервер Apache – це безкоштовне програмне забезпечення з відкритим вихідним кодом для крос-платформного веб-сервера, випущене на умовах

ліцензії Apache License 2.0. Apache розробляється і підтримується відкритою спільнотою розробників під егідою Apache Software Foundation [5].

Головна перевага сервера Apache полягає в тому, що це програма з відкритим вихідним кодом, яка безплатно поширюється в Інтернеті. Програмісти постійно додають нові функції до сервера, використовуючи API Apache для створення модулів, які працюють з основним кодом Apache. Це робить Apache потужнішим і стабільнішим.

Основні переваги Web-сервера Apache [6]:

- доступність – це програмне забезпечення з відкритим вихідним кодом. Отже, його може безкоштовно використовувати або модифікувати будь-хто. Розробники по всьому світу створюють конфігурації та модулі веб-сервера для своїх специфічних потреб. З цієї ж причини Apache регулярно отримує корисні доповнення, що розширюють його базовий функціонал;
- гнучкість налаштування – Apache використовує кілька конфігураційних файлів для керування веб-сервером. Це дає змогу налаштувати ПЗ під вузькоспрямовані завдання;
- функціональність – у Apache динамічна модульна структура. Можна швидко підключати додатковий функціонал у вигляді модулів, які завантажуються, навіть без звернення до зовнішніх джерел. Це дає змогу вирішувати цілий комплекс найважливіших завдань у сфері безпеки, кешування, редагування URL, розподілу навантаження. Завдяки гібридним модулям MPM, Apache може однаково успішно обслуговувати статичний і динамічний контент. Є можливість оперативно відключати непотрібні модулі та прискорювати роботу веб-сервера;
- кросплатформеність – Apache працює як на Windows, так і на всіх Unix-подібних системах. Адміністрування веб-сервером не має серйозних ві-

- дмінностей на різних ОС. Індивідуальний тільки процес встановлення і розташування директорій з файлами програми;
- сумісність – Apache працює на базі скриптових або веб-орієнтованих мов (PHP, Python, Tcl, Ruby, Perl, ASP), що робить його сумісним із найширшим спектром баз даних і серверного ПЗ. Багато веб-додатків та інструментів відразу виходять із засобами запуску з-під Apache у вигляді PHP-модуля. Веб-сервер підтримує технології FastCGI і CGI, що дають змогу користуватися програмними продуктами на об'єктно-орієнтованих мовах Java, sh, C, C++;
  - масштабованість – підходить для веб-ресурсів будь-якого масштабу. Apache добре працює як на односторінковому сайті (лендінгу), так і на багатосторінковому сайті зі щоденною аудиторією в десятки тисяч відвідувачів;
  - підтримка користувачів – Apache утримує першість популярності серед веб-серверів з 1996 року. За минулий час для нього створено величезну базу документації – як офіційної, так і створеної сторонніми розробниками. Готові, детально описані керівництва можна знайти практично на будь-який сценарій.

Насамкінець, вибір Apache як веб-сервера для розробки веб-додатку «Книжкова крамниця» обґрунтовується його надійністю, відкритим вихідним кодом, потужними функціями і модульністю, крос-платформеністю, підтримкою спільноти та інтеграцією з іншими компонентами.

### 2.3 Вибір системи управління базами даних

Для здійснення розробки веб-додатку «Книжкова крамниця» необхідно здійснити вибір СУБД, яка дозволить реалізувати зберігання, пошук, швидкий доступ до всієї інформації, яка потрібна для функціонування додатку.

MySQL – це реляційна система управління базами даних і СУБД з відкритим вихідним кодом, створена в 1995. Метою створення MySQL було збільшення швидкості обробки великих обсягів даних. Вона підтримується багатьма відомими мовами програмування. MySQL є найбільш широко використовуваною серверною базою даних. MySQL часто використовується у веб-додатках і сервісах [7].

Якщо потрібна безкоштовна або недорога система управління базами даних, є кілька варіантів на вибір: MySQL, mSQL, Postgres, один з безкоштовних, але не підтримуваних двигунів від комерційних постачальників тощо. Коли ви порівнюєте MySQL з іншими системами баз даних, подумайте про те, що для вас є найбільш важливим. Продуктивність; підтримка; можливості (відповідність SQL, розширення тощо); ліцензійні умови та обмеження; ціна - все це фактори, які слід брати до уваги. Враховуючи ці фактори, MySQL має багато привабливих можливостей, які можна запропонувати [7]:

- швидкість. MySQL працює швидко. Розробники стверджують, що MySQL – це найшвидша база даних, яку ви можете отримати.
- простота використання. MySQL – це високопродуктивна, але відносно проста система управління базами даних, яку набагато легше налаштувати та адмініструвати, ніж більші системи. – Вартість. MySQL є безкоштовною для більшості внутрішніх потреб;
- підтримка мов запитів. MySQL розуміє мову SQL (Structured Query Language), яка є основною мовою для всіх сучасних систем баз даних. Ви також можете отримати доступ до MySQL за допомогою програм, які підтримують ODBC (Open Database Connectivity), протокол зв'язку з базами даних, розроблений Microsoft;
- можливості. Багато клієнтів можуть підключатися до сервера одночасно. Клієнти можуть використовувати декілька баз даних одночасно. Ви можете отримати доступ до MySQL в інтерактивному режимі за допо-



- могою декількох інтерфейсів, які дозволяють вводити запити і переглядати результати: клієнти командного рядка, веб-браузери або клієнти X Window System. Крім того, доступні різноманітні інтерфейси програмування для таких мов, як C, Perl, Java, PHP і Python. Таким чином, у вас є вибір між використанням готового клієнтського програмного забезпечення або написанням власного для індивідуальних додатків;
- підключення та безпека. MySQL повністю мережева, і до баз даних можна отримати доступ з будь-якої точки Інтернету, тому ви можете ділитися своїми даними з будь-ким і будь-де. Але MySQL має контроль доступу, щоб люди, які не повинні бачити ваші дані, не могли їх побачити;
  - переносимість. MySQL працює на багатьох різновидах UNIX, а також на інших не UNIX-системах, таких як Windows і OS/2. MySQL працює на обладнанні від домашніх комп'ютерів до серверів високого класу;
  - відкрите розповсюдження. MySQL легко отримати; просто використуйте ваш веб-браузер. Якщо ви не розумієте, як щось працює, або вам цікавий алгоритм, ви можете отримати вихідний код і розібратися в ньому. Якщо вам не подобається, як щось працює, ви можете це змінити.

Виходячи з цих факторів можна зробити висновок, що MySQL надає простоту використання і гнучкість, що робить його привабливим вибором для розробників усіх рівнів навичок. Крім того, підтримка й активність спільноти MySQL забезпечують доступ до великих ресурсів і допомоги, що може бути цінним під час розробки. MySQL пропонує надійну та ефективну базу даних, яка може бути адаптована під ваші потреби. Вибір MySQL забезпечує стабільність, розширюваність і можливість розвитку вашого проекту в міру його зростання і зміни вимог.

## 2.4 Вибір мови програмування

### 2.4.1 Мова сценаріїв PHP

В якості мови сценаріїв для розробки веб-додатку «Книжкова крамниця» була обрана скриптова мова PHP.

Мова програмування PHP – серверна мова, за допомогою якої можна створювати Web-сайти, причому як невеликі лендінги, що складаються з однієї сторінки, так і гігантські системи, що використовують сотні й тисячі серверів. Електронна енциклопедія Wikipedia, соціальна мережа Facebook створені з використанням PHP. Будучи однією з найперших мов програмування, орієнтованих на Web-розробку, PHP пройшла тривалий шлях практично з самого початку зародження Web. Тому у світі вона залишається однією з найпопулярніших і найзатребуваніших мов [8].

В основі популярності PHP лежать такі переваги:

- орієнтація на Web-розробку – PHP створювали, розвивали і підтримували як мову для створення Web-сайтів. Багато конструкцій і рішень у ній створено для зручності роботи у Web-середовищі;
- кросплатформеність – PHP перенесено на всі основні операційні системи: можна розробляти сайт у Windows, Mac OS X, а експлуатувати на Linux-сервері. Складнощі перенесення будуть мінімальні і нівелюватися мовою;
- безкоштовність – PHP є розробкою зі світу вільного програмного забезпечення, не потрібно платити ні за саму мову, ні за більшість супутніх програм (редактори, Web-сервери, бази даних). До того ж більшість програмних продуктів, з якими доведеться мати справу, матимуть доступний для вивчення і модифікації вихідний код. Вкладення можуть знадобитися під час оренди доменного імені та сервера для публікації сайту в Інтернеті. Однак вивчати PHP можна, не вкладаючи ні копійки;

- низький поріг входу – вивчити PHP і почати створювати на ньому готові додатки набагато простіше, ніж із використанням конкуруючих технологій (.NET, Python, Ruby, Go). Вивчення PHP не закриває для розробника інші технології, у Web сама мова – значна, але менша частина використовуваних технологій. Знання, прийоми роботи, супутні технології (Web-сервери, бази даних, бібліотеки, допоміжні мови) стануть у пригоді і в будь-якій іншій екосистемі, відмінній від PHP. При створенні власного бізнесу зібрати команду PHP-розробників часто простіше і дешевше за все.

PHP має не тільки переваги, але й недоліки:

- відсутність лідера – багато технологій і мов мають лідера, архітектора, який визначає вигляд технології, задає вектор розвитку, ухвалює рішення про те, що має бути обов'язково, а чого не буде ніколи (Linux, Python, Ruby тощо). У PHP лідера немає, багато рішень і конструкцій – це компроміс зацікавлених груп та реалій, що історично склалися;
- непослідовний синтаксис – при вивченні мови PHP, особливо старої частини, заснованої на функціях, можна помітити, що частина функцій має префікси `array_`, `str_`, частина не має. Параметри функцій можуть бути розташовані не зовсім логічно і не так, як в іншій функції цієї ж групи;
- PHP – мова, що вже досить довго живе. Коли мова тільки з'являється, вона доволі елегантна і внутрішньо узгоджена. У міру життєвого циклу мова обростає додатковими ключовими словами, артефактами, застарілими конструкціями, які начебто є, працюють, але якими не рекомендується користуватися. У PHP була досить бурхлива молодість, під час якої було скасовано масу директив і прийомів, які, на перший погляд, мали полегшувати розробку, а на практиці оберталися серйозними проблемами безпеки. Сам PHP, що стартував як необ'єктно-орієнтована

мова, зараз став повноцінною об'єктно-орієнтованою мовою. Однак у ній повно старих процедурних артефактів, якими доведеться користуватися;

- співтовариство PHP-розробників велике і роз'єднане, тому що PHP – це одна з перших технологій для розроблення Web-проектів, половину Інтернету створено за її участю. У PHP-розробку одночасно була залучена величезна кількість програмістів по всьому світу. Усе це породило велику кількість найрізноманітніших підходів, фреймворків і несумісних одна з одною екосистем. Ба більше, завдяки зусиллям потужних і впливових соціальних мереж (насамперед Facebook) з'явилися альтернативні реалізації PHP. Це погано, тому що багато екосистем усередині PHP не сумісні, а співтовариство роздроблене і витрачає сили на створення одних і тих самих бібліотек у рамках різних груп. Ситуацію виправляють і за допомогою PSR-стандартів. Розробники домовляються про єдині правила та інтерфейси, що забезпечують сумісність фреймворків, але цей процес ще на початку шляху, тоді як конкуруючі технології (.NET, Ruby) вже мають єдину платформу для всіх фреймворків.

Мова PHP має ядро і розширення мови. Між ядром і розширеннями досить важко провести межу, тому що багато розширень давно включені до складу ядра або поширюються у вигляді скомпільованих бінарних бібліотек і легко встановлюються.

Є й інша частина – код, створений на PHP, який умовно можна поділити на такі типи:

- компоненти – бібліотеки на PHP, які збираються за допомогою менеджера пакетів Composer;
- фреймворки – готові збірки, найчастіше з компонентів, за допомогою яких можна створювати сайти будь-якого ступеня складності.

– готові додатки – готові до використання розробки на PHP. Це системи управління контентом (Wordpress, Drupal), форуми (phpBB), Web-інтерфейси управління базами даних (phpMyAdmin, pgAdmin).

На закінчення, вибір PHP як мови програмування для розробки веб-додатку «Книжкова крамниця» виправданий з низки причин. PHP має широке поширення і величезну спільноту розробників, що забезпечує доступ до багатого вибору ресурсів і підтримки. Простий і зрозумілий синтаксис PHP робить його привабливим для початківців-розробників, даючи їм змогу швидко почати створення функціональних веб-додатків. Завдяки своїй розширюваності та можливостям інтеграції з іншими технологіями, PHP надає гнучку платформу для створення потужних веб-сайтів і додатків з динамічним вмістом. Крім того, PHP має багатий набір бібліотек і розширень, які полегшують розробку і додавання нових функціональних можливостей. Незалежно від масштабу проекту, PHP надає інструменти та ресурси для створення ефективних і успішних веб-додатків. Через усі ці переваги PHP є найкращим вибором при розробці додатку «Книжкова крамниця».

#### 2.4.2 Мова сценаріїв JavaScript

Також для розробки веб-додатку «Книжкова крамниця» була обрана мова JavaScript.

JavaScript – це вбудована, мультиплатформна, інтерпретована мова, створена компанією Netscape. Як і HTML, вона сумісна з усіма платформами.

JavaScript дає змогу розширювати функціональні можливості, вбудовуючи прикладний код безпосередньо в код HTML. За допомогою цієї мови можна створювати додатки з широким діапазоном: починаючи від застосування засобів взаємодії користувачів на і закінчуючи обміном інформацією з базами даних. Існують два методи вбудовування коду JavaScript у Web-сторінки: включення

операторів мови до коду, що виконується на клієнтській платформі, і включення операторів мови до коду, що виконується на серверній платформі. Обидва методи містять один і той самий базовий набір засобів, визначених ядром мови. Водночас кожен метод має у своєму розпорядженні і власний, розширений, набір мовних об'єктів і засобів. Клієнтські додатки JavaScript являють собою сценарії, вбудовані безпосередньо в і оброблювані браузером користувача в процесі завантаження сторінки. Щонайменше 90 % усіх сценаріїв, що застосовуються у Web, належать до подібної категорії. Коли браузер користувача викликає HTML-сторінку з вбудованим у неї сценарієм JavaScript, інтерпретатор JavaScript браузера читає сценарій від верху до низу, виконуючи оператори в міру їхньої появи. Однією з переваг клієнтського застосування сценарію є те, що він дає змогу розпізнати події, спричинені діями користувача (наприклад, введення даних у формі, натискання на кнопку, поява курсору миші над конкретним об'єктом тощо). Потім, отримуючи інформацію про дії користувача, він може викликати інші частини сценарію і виконувати іншу роботу, не звертаючись знову до Web-сервера і не вимагаючи додаткової інформації [9].

В якості доповнення до JavaScript при розробці веб-додатку «книжкова інтернет-крамниця» була обрана популярна бібліотека React.

React.js, більш відомий як React, – це безкоштовна бібліотека JavaScript з відкритим вихідним кодом. Вона найкраще підходить для створення користувацьких інтерфейсів, об'єднуючи частини коду (компоненти) у повноцінні веб-сайти. Спочатку її створив Facebook, а зараз її підтримують Meta та спільнота розробників з відкритим вихідним кодом. Однією з переваг React є те, що ви можете використовувати його так багато або так мало, як вам потрібно! Наприклад, ви можете побудувати весь сайт на React або використовувати лише один React-компонент на одній сторінці. React.js створений з використанням JSX – комбінації JavaScript та XML. Елементи створюються за допомогою JSX, а потім за допомогою JavaScript відображаються на вашому сайті. Хоча React має

круту криву навчання для молодших розробників, він швидко перетворюється на одну з найпопулярніших і затребуваних бібліотек JavaScript.

Насамкінець, вибір JavaScript і React JS як основних інструментів розробки веб-додатку «Книжкова крамниця» може бути обґрунтований їхньою популярністю, ефективністю, модульністю, одностороннім зв'язуванням даних, розширюваністю і кросплатформеністю. JavaScript є однією з найпопулярніших мов програмування, має широку екосистему і підтримку спільноти розробників. React JS, своєю чергою, пропонує ефективний спосіб створення компонентів користувацького інтерфейсу, модульну архітектуру, зручне керування станом і багаті можливості для створення інтерактивних додатків.

#### 2.4.3 Мова розмітки гіпертексту HTML

HTML – це мова розмітки тексту, яка зараз використовується у Всесвітній павутині. Це відкрита мова розмітки, за допомогою якої ви вказуєте веб-браузерам, як відображати веб-сторінки.

Невидима розмітка, що використовується в текстових редакторах, схожа на розмітку HTML, яка використовується для створення веб-сторінок. У випадку з HTML, команди розмітки, спрямовані на веб-вміст, повідомляють програмному забезпеченню браузера про структуру документа і, де це можливо, про те, як він повинен відображатися. Коли браузер читає документ, що містить HTML-розмітку, він визначає, як відобразити його на екрані, беручи до уваги елементи HTML, вбудовані в документ. HTML-документ – це просто текстовий файл, який містить інформацію, яку потрібно опублікувати. Він також містить вбудовані інструкції, які називаються елементами, що вказують, як веб-браузер повинен структурувати або представляти документ. Елементи HTML вказують на «розмітку» оточуючого тексту. Вони можуть вказувати на значення вкладеної інформації (наприклад, цитата) або на те, як вона має бути відображена (напри-

клад, напівжирним шрифтом). Елементи HTML зазвичай складаються з пари тегів, які називаються тегами-контейнерами, оскільки вміст знаходиться між ними. Однак деякі елементи, такі як тег горизонтального вирівнювання або розрив строки, не мають відповідного кінцевого тегу [9].

HTML є кращим вибором для розмітки веб-сторінок при розробці веб-додатку «Книжкова крамниця» з таких причин: його універсальність і підтримка всіма веб-браузерами, простота використання та сумісність з іншими технологіями, підтримка семантики та мультимедіа, а також наявність широкого співтовариства розробників і великих ресурсів для підтримки й навчання. Усі ці фактори роблять HTML зручним і потужним інструментом для створення веб-сторінок і додатків.

#### 2.4.4 Мова розмітки CSS

У той час як HTML використовується, щоб описати вміст веб-сторінки, саме каскадні таблиці стилів (Cascading Style Sheets, CSS) впливають на те, який вигляд має контент. Говорячи про веб-дизайн, спосіб, яким представлена сторінка, відомий як її представлення. Це означає, що шрифтами, кольорами, фоновими зображеннями, інтервалами між рядками, макетом сторінки та іншим. За допомогою новітньої версії (CSS3) можна додавати на сторінку навіть спеціальні ефекти та просту анімацію. Каскадні таблиці стилів також керують поданням документів не тільки в браузерах, а й у таких контекстах, як друк і пристрої з екранами з малою діагоналлю. Крім того, у таблицях стилів існують правила для визначення невізуального представлення документів, наприклад, як звучатиме текст під час зчитування його програмою екранного доступу (однак вони не дуже добре підтримуються). Таблиці стилів – чудовий інструмент, що дає змогу автоматизувати процес розроблення, тому що ви можете вносити зміни,



що стосуються всіх сторінок сайту, редагуючи один-єдиний документ таблиці стилів. Вони певною мірою підтримуються всіма сучасними браузерами [9].

CSS є найкращим вибором для стилізації веб-сторінок при розробці веб-додатку «Книжкова крамниця» з таких причин: його універсальність і підтримка всіма веб-браузерами, простота використання і можливість легкого застосування стилів до елементів HTML, гнучкість і потужні можливості налаштування зовнішнього вигляду веб-сторінок, а також наявність великого набору стильових властивостей і селекторів для досягнення бажаних ефектів. CSS дає змогу створювати естетично привабливий та сучасний дизайн, а також забезпечує легкість підтримки та модифікації стилів на різних сторінках і проектах.

На закінчення, можна сказати що HTML, CSS, JS, PHP та ReactJS є загальними та широко використовуваними технологіями веб-розробки. HTML – це основна мова розмітки, яка визначає структуру веб-сторінки. CSS використовується для стилізації та естетичного оформлення елементів на сторінці. JS – це мова програмування, яка додає інтерактивності та функціональності веб-сторінкам, забезпечуючи динамічне створення контенту та взаємодію з користувачем; PHP – це серверна мова програмування, яка обробляє дані на сервері, включаючи маніпуляції з базами даних та створення динамічних веб-додатків. ReactJS – популярна бібліотека JavaScript для розробки користувацьких інтерфейсів, що полегшує створення масштабованих і багаторазових компонентів та полегшує розробку складних веб-додатків. Ці технології обирають за їх широкий спектр можливостей, спільноти розробників, великі ресурси для навчання та підтримки, а також за їх здатність створювати високоякісні, інтерактивні веб-додатки.

### 3 ПРОЕКТУВАННЯ ВЕБ-СЕРВІСУ «КНИЖКОВА КРАМНИЦЯ»

#### 3.1 Проектування веб-додатку за допомогою методології SADT

##### 3.1.1 Використання мови IDEF0 для проектування веб-додатку

При проектуванні веб-додатку «Книжкова крамниця» була обрана методологія функціонального моделювання SADT (Structured Analysis and Design Technique) – стандарт IDEF0.

Нотації SADT складаються з діаграм зі стрілками (блоків), з чотирма стрілками на кожній стороні, визначеними як: вхід, вихід, контроль і механізм, і однією діяльністю посередині. Їх визначення складаються з наступного [10]:

**Діяльність:** Діяльність – це будь-яка функція або процес, який слугує для перетворення входів у виходи.

**Вхідні дані:** Дані/інформація, необхідні для того, щоб розпочати процес перетворення.

**Вихід:** дані/інформація, вироблені діяльністю в результаті цього перетворення.

**Контроль:** Будь-яке обмеження, яке певним чином впливає на поведінку діяльності.

**Механізм:** Люди, ресурси або будь-які засоби, необхідні для виконання діяльності.

Контекстна діаграма – це високорівнева модель, яка показує систему або процес у контексті зовнішнього середовища. Це перше представлення системи, що дозволяє зрозуміти взаємодію із зовнішніми факторами і позначити межі системи. Вона не надає детальної інформації про внутрішню структуру і функціонування системи, але слугує загальним оглядом і контекстом для розуміння системи та її взаємодії із зовнішніми факторами. Вона є відправною точкою для більш детального моделювання та аналізу функцій і процесів системи [10].

Контекстна діаграма веб-додатку «Книжкова крамниця» наведена на рис.

### 3.1.

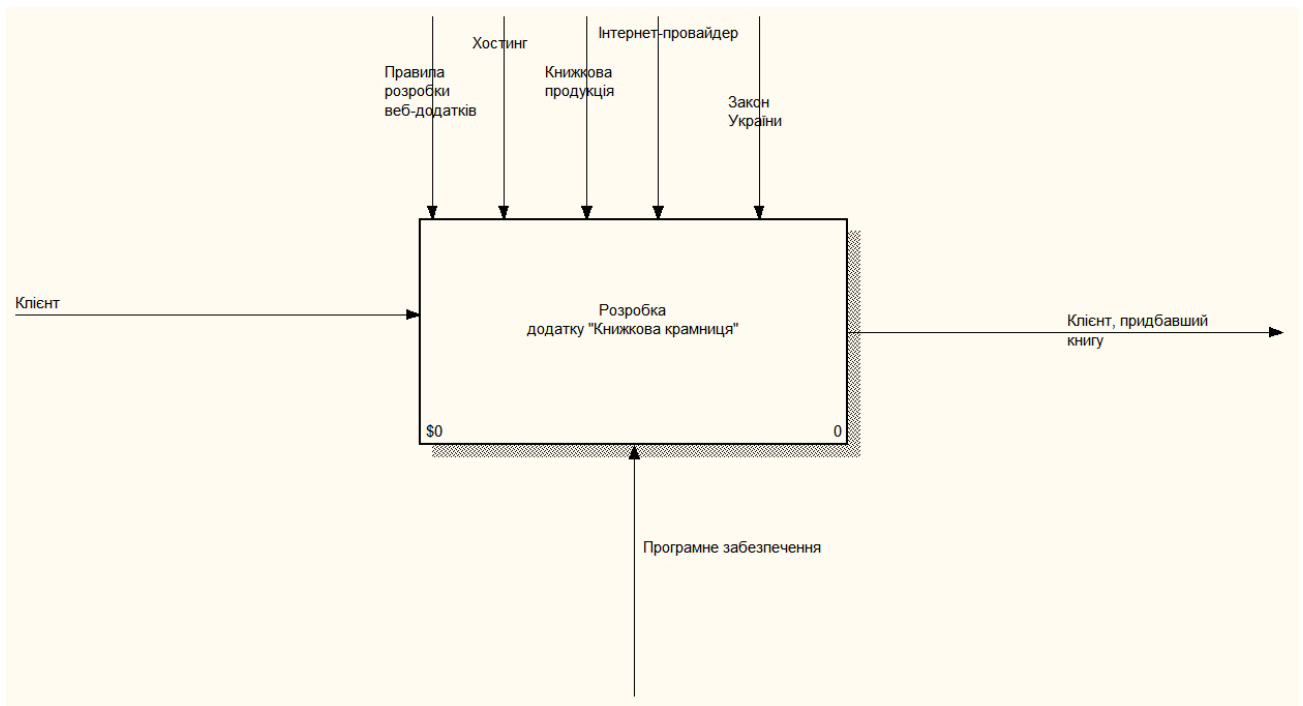


Рисунок 3.1 – Контекстна діаграма веб-додатку «Книжкова крамниця»

На контекстній діаграмі відображається головний процес веб-додатку «Книжкова крамниця». На вході є інформація про клієнта. Головна робота керується: правилами розробки веб-додатків, наявністю книжкової продукції, хостингом, Інтернет-провайдером, та законами України. На виході – клієнт, який придбав книгу.

Після того, як система визначена як єдине ціле, її розбивають на менші частини, іншими словами проводиться декомпозиція системи. Декомпозиція – це процес розбиття системи на менші компоненти та підзадачі з метою більш детального аналізу та розуміння її структури та функцій. Це дозволяє уточнити моделі і створити більш детальні функціональні моделі для кожної частини системи. Декомпозиція полегшує системний аналіз, проектування та документу-

вання, розбиваючи складні системи на більш керовані та зрозумілі компоненти. Кожен рівень декомпозиції забезпечує глибший рівень деталізації, що дозволяє точніше ідентифікувати функції, процеси та взаємозв'язки в системі. На різних рівнях декомпозиції синтаксис, що використовується у функціональній моделі, може бути однаковим, але може бути модифікований для врахування більш детальної інформації про компоненти системи [10].

Після того як, була проведена декомпозиція контекстної діаграми, з'явилися три блоки. Ці блоки описують основні роботи.

Блок «Аналіз вимог» включає в себе аналіз існуючих систем та формулювання вимог. На вході нічого немає. Керується правилами розробки веб-додатків. Механізмом слугує програмне забезпечення. На виході отримуємо сформульоване завдання.

Блок «Розробка додатку» включає в себе повну розробку веб-додатку «Книжкова крамниця». Включає в себе розробку основних функцій, користувачького інтерфейсу, та розробку бази даних. Входом є сформульоване завдання, на виході отримуємо розроблений додаток. Керується правилами розробки веб-додатків. Механізмом слугує програмне забезпечення.

Останнім блоком є «Розміщення додатку». Він включає в себе отримання доменного імені, публікацію додатку на сервері, наповнення бази даних та реалізацію. На вході є інформація про клієнта та розроблений додаток. Виходом є клієнт, який придбав книгу.

Діаграма декомпозиції розробки веб-додатку «Книжкова крамниця» наведена на рис. 3.2.

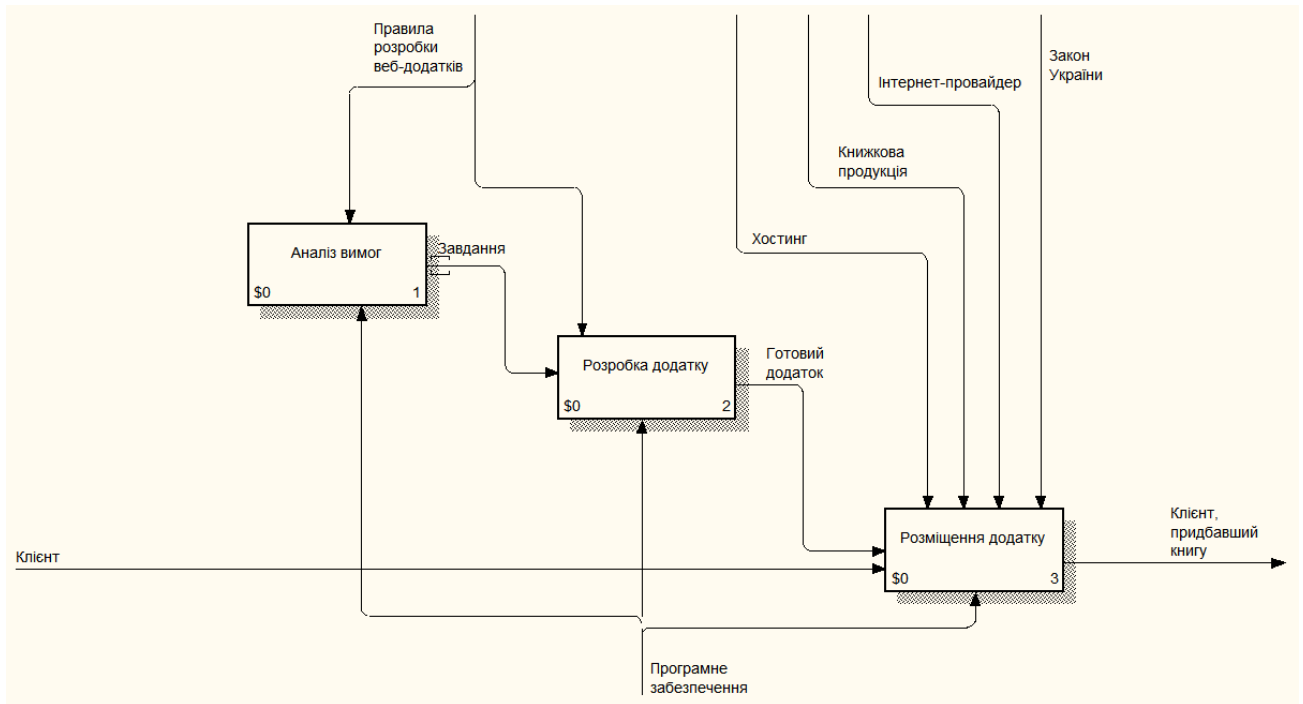


Рисунок 3.2 – Діаграма декомпозиції розробки додатку «Книжкова крамниця»

При декомпозиції блоку «Розробка додатку», отримуємо наступні блоки: «Розробка інтерфейсу». На цьому етапі виконується розробка користувацького інтерфейсу. входом слугує сформульоване завдання. Робота керується правилами розробки веб-додатків. Механізмом є програмне забезпечення. На виході отримуємо розроблений інтерфейс.

«Розробка функціональності». На цьому етапі виконується реалізація основних функцій веб-додатку. Входом є розроблений інтерфейс та завдання. Робота керується правилами розробки веб-додатків. Механізмом є програмне забезпечення. На виході отримуємо готову функціональність додатку.

«Розробка бази даних». На цьому етапі виконується проектування логічної та фізичної моделі даних. Входом слугує розроблена функціональність додатку, та завдання. Робота керується правилами розробки веб-додатків. Механізмом є програмне забезпечення. На виході отримуємо готовий розроблений додаток. Діаграма декомпозиції блоку «Розробка додатку» наведена на рис. 3.3.

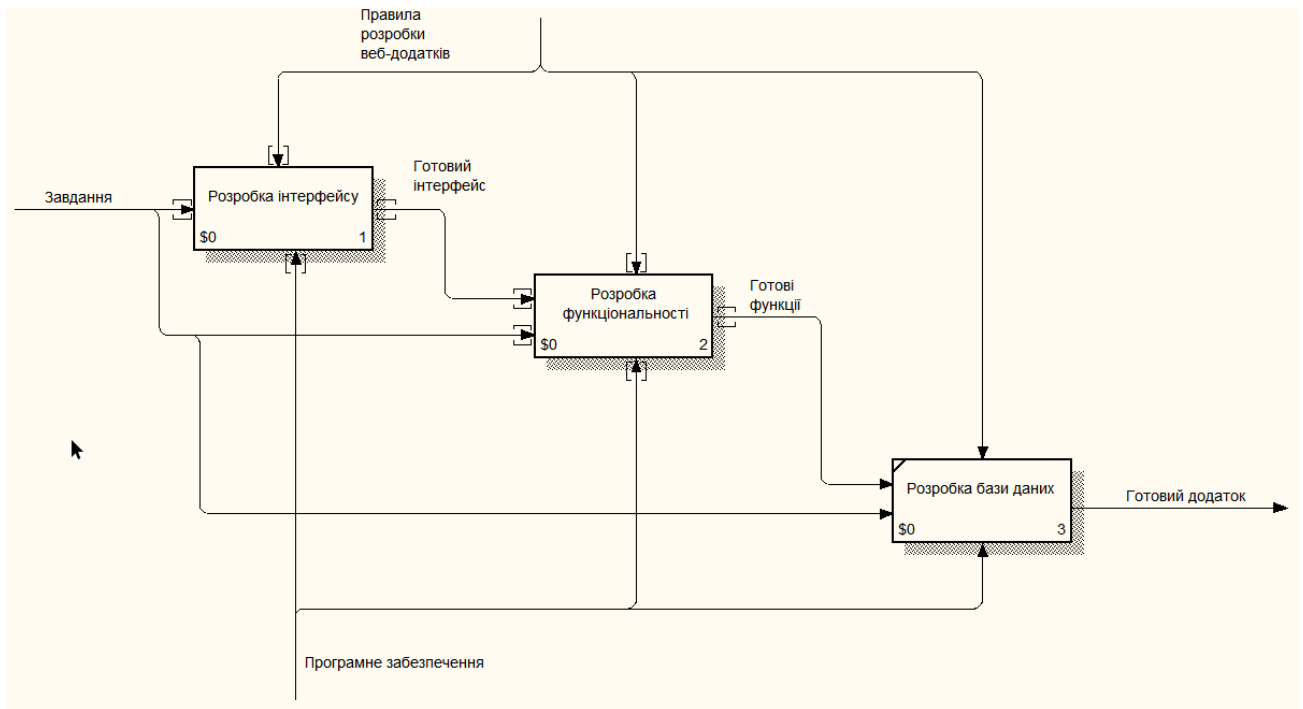


Рисунок 3.3 – Діаграма декомпозиції блоку «Розробка додатку»

Останнім блоком є «Розміщення додатку». При його декомпозиції, отримуємо чотири нових блоки:

«Отримання доменного імені». На цьому етапі отримується доменне ім'я для розміщення додатку. Робота керується хостингом та Інтернет-провайдером. Механізмом є програмне забезпечення. На виході є отримане доменне ім'я.

«Публікація додатку на сервері». На цьому етапі виконується налаштування серверу та розміщення на ньому додатку. Входом є розроблений додаток та отримане доменне ім'я. Робота керується хостингом та Інтернет-провайдером. Механізмом слугує програмне забезпечення. На виході отримуємо опублікований додаток на сервері.

«Наповнення бази даних». На цьому етапі виконується заповнення бази даних інформацією про книжкову продукцію. Робота керується хостингом, Ін-

тернет-провайдером, та інформацією про книжкову продукцію. Механізмом є програмне забезпечення. На виході – додаток з наповненою базою даних.

«Реалізація». На цьому етапі виконується введення додатку в експлуатацію. Входом є інформація про клієнта, та готовий додаток з книжною продукцією. Робота керується хостингом, Інтернет-провайдером, та діючими законами України. Механізмом, у даному блоці слугує програмне забезпечення. На виході – клієнт, який придбав книгу.

Діаграма декомпозиції блоку «Розміщення додатку» наведена на рис. 3.4.

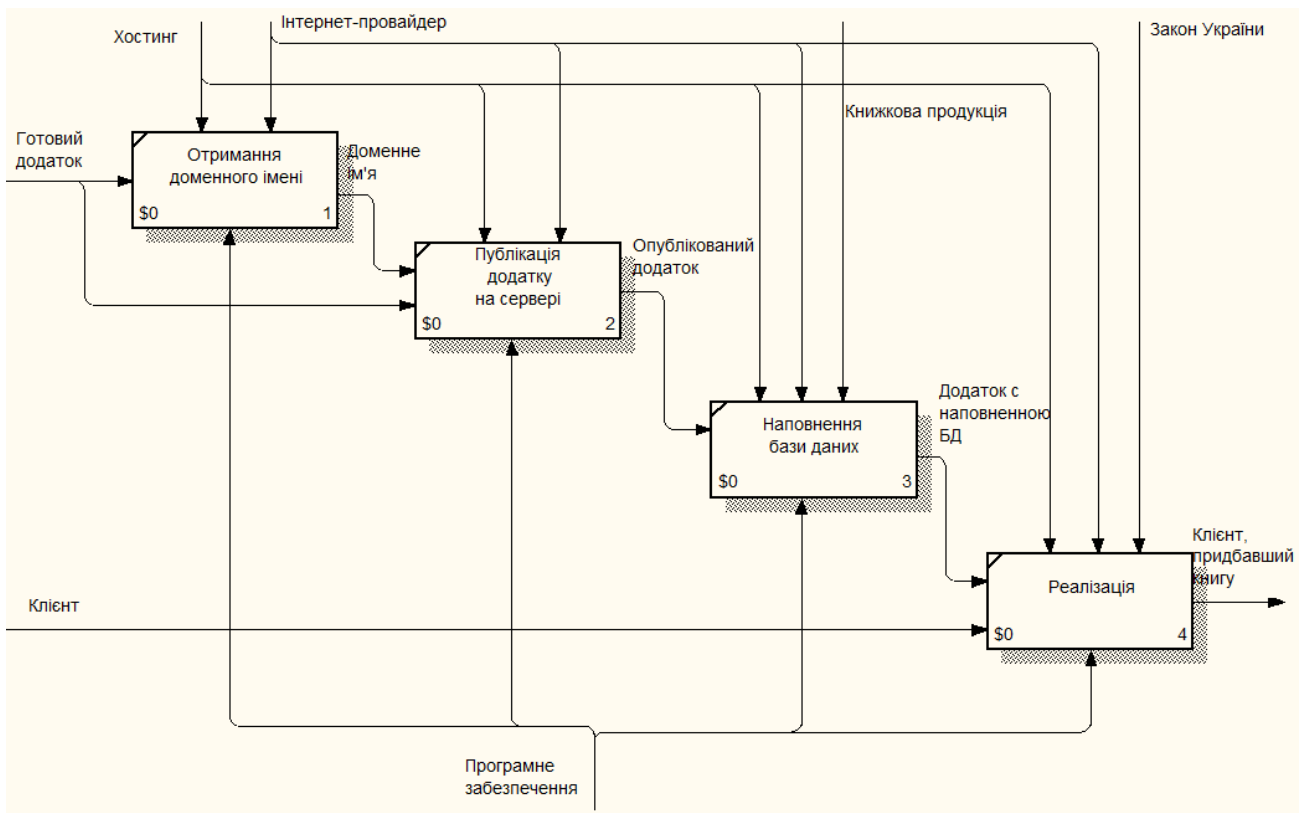


Рисунок 3.4 – Діаграма декомпозиції блоку «Розміщення додатку»

### 3.2 Проектування веб-додатку за допомогою нотації IDEF3

Проектування веб-додатку «Книжкова крамниця» було здійснено за допомогою послідовного виконання процесів Workflow Diagramming (стандарт IDEF3).

IDEF3 (Integration Definition for Process Description Capture Method) – це методологія та нотація для моделювання та опису процесів в організації. IDEF3 розроблено для опису функцій, потоків даних та їхніх взаємозв'язків у системі або організації. Нотація IDEF3 являє собою мережу функціональних блоків, які моделюють процеси та їхню взаємодію. IDEF3 дає змогу створювати докладні моделі процесів, які можуть бути використані для аналізу, оптимізації та поліпшення бізнес-процесів. Нотація IDEF3 є частиною більш загальної родини нотацій IDEF, розроблених для моделювання та опису інформаційних систем і процесів. Функціональні блоки в IDEF0 можуть бути виражені як процес або послідовність процесів за допомогою IDEF3. IDEF0 описує, що робиться в системі, в той час як IDEF3 описує, як це робиться. Контекстні діаграми IDEF3 надають загальну картину процесу або системи і фокусуються на його взаємозв'язках із зовнішніми об'єктами і зовнішнім середовищем. Відображається взаємодія процесу або системи із зовнішніми об'єктами, а також основні входи і виходи процесу або системи. Контекстні діаграми IDEF3 дозволяють швидко оцінити загальну картину процесу або системи та його взаємодію із зовнішнім середовищем [11].

Контекстна діаграма додатку у нотації IDEF3 складається з єдиної роботи – «Розробка веб-додатку «Книжкова крамниця»».

Контекстна діаграма представлена на рис. 3.5.



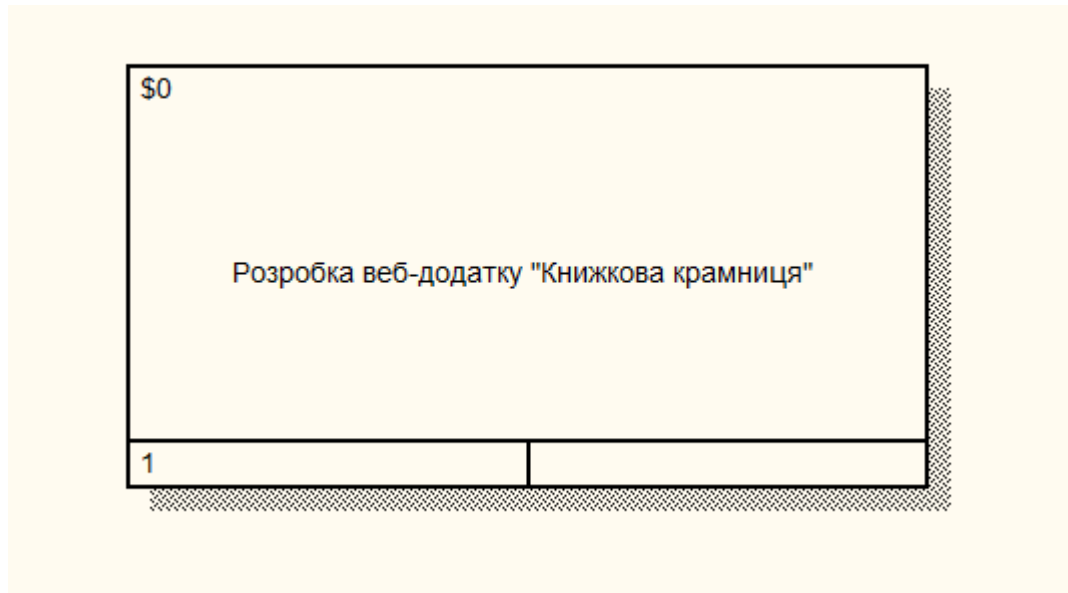


Рисунок 3.5 – Контекстна діаграма розробки веб-додатку «Книжкова крамниця»

Після декомпозиції контекстної діаграми з'являється послідовність процесів (робіт). Першою роботою є «Аналіз вимог». Після неї йде робота «Розробка додатку». Останньою роботою є «Розміщення додатку».

Діаграма декомпозиції контекстної діаграми представлена на рис. 3.6.

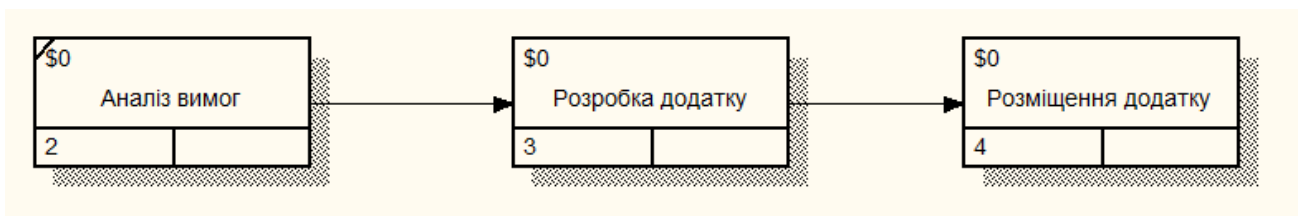


Рисунок 3.6 – Діаграма декомпозиції контекстної діаграми «Розробка веб-додатку «Книжкова крамниця»»

При декомпозиції роботи «Розробка додатку» отримали три блоки з двома перетинами (перехрестями).

Перехрестя використовуються для відображення логіки взаємодії стрілок (потоків) під час злиття та розгалуження або для відображення безлічі подій, що можуть або мають бути завершені перед початком наступної роботи. Розрізняють перехрестя для злиття (Fan-in Junction) і розгалуження (Fan-out Junction) стрілок. Перехрестя не може використовуватися одночасно для злиття і для розгалуження [11].

Перший перетин, «Асинхронне І», означає, що інші роботи можуть не розпочатися одночасно, але обов'язково повинні бути початі. Це роботи «Розробка інтерфейсу» та «Розробка функціональності». При з'єднанні стрілок, які виходять з цих процесів, також було використано перетин «асинхронне І». Це означає, що роботи повинні бути завершені, але не обов'язково одночасно. Останньою йде робота «Розробка бази даних».

Діаграма декомпозиції роботи «Розробка додатку» представлена на рис. 3.7.

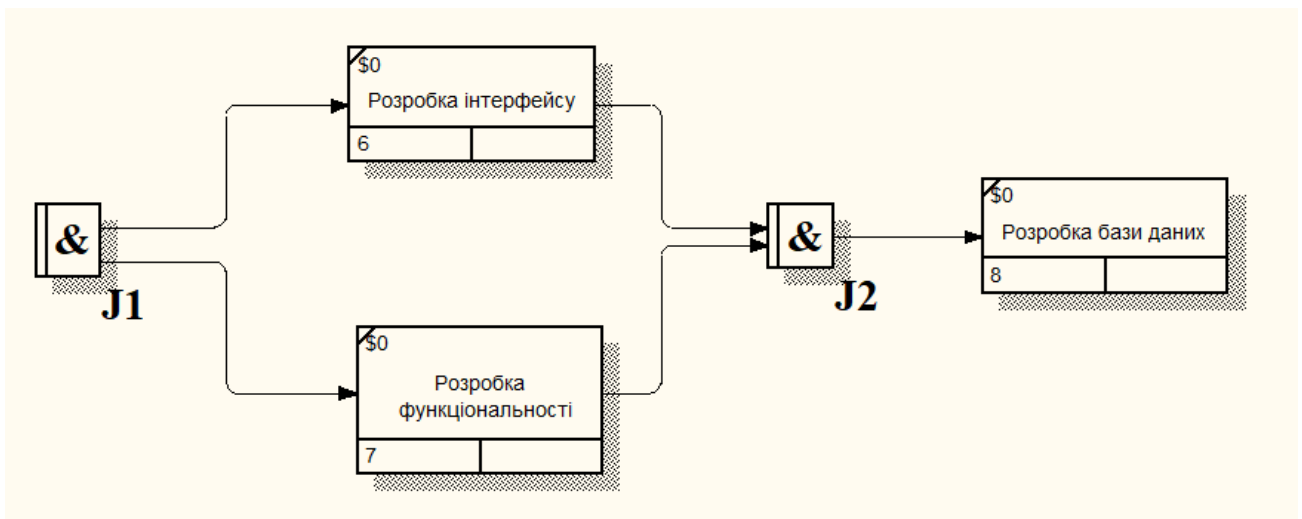


Рисунок 3.7 – Діаграма декомпозиції процесу «Розробка додатку»

Після декомпозиції роботи «Розміщення додатку», з'явилися чотири блоки робіт. Ці роботи пов'язані між собою старшим зв'язком і та йдуть послідовно

одна за одну. А саме: «Отримання доменного імені», «Публікація додатку на сервері», «Наповнення бази даних» та «Реалізація».

Діаграма декомпозиції, показана на рис. 3.8.

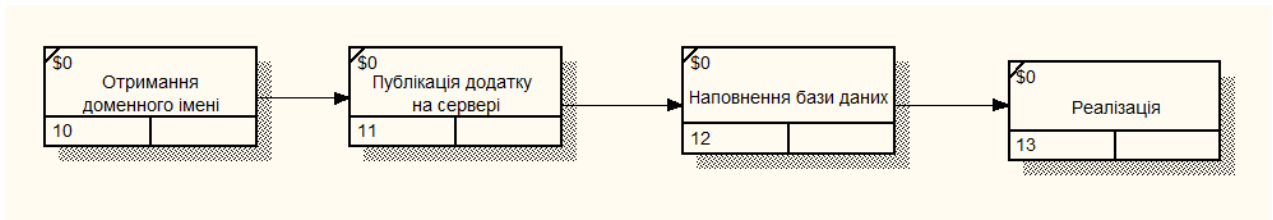


Рисунок 3.8 – Діаграма декомпозиції роботи «Розміщення додатку»

### 3.3 Проектування веб-додатку за допомогою нотації DFD

DFD (Data Flow Diagram) – це графічний метод представлення та моделювання потоків даних в інформаційній системі. DFD-діаграми використовують для візуалізації та аналізу процесів, а також для проектування нових систем.

DFD-діаграми мають різні рівні деталізації, починаючи з контекстної діаграми (Context Diagram), яка представляє загальну картину системи та її взаємодії із зовнішніми сутностями, і далі розбиваються на більш деталізовані рівні. Ця декомпозиція продовжується, створюючи багаторівневу ієрархію діаграм, поки процес не стане примітивним і не досягне рівня, який не може бути деталізований. Нотація DFD є корисним інструментом для аналізу і проектування систем, оскільки вони допомагають виявити потоки даних, взаємодію між компонентами системи і виділити ключові процеси. Вони також допомагають в ідентифікації вузьких місць, проблем у продуктивності або можливостей для оптимізації процесів. DFD-діаграми широко застосовують у сфері системного аналізу та проектування, бізнес-моделювання та розроблення програмного забезпе-

чення для представлення й аналізу інформаційних потоків у різних типах систем [12].

У контекстній діаграмі головним процесом є «Розробка веб-додатку «Книжкова крамниця»». Зовнішніми сутностями, які впливають на систему, є: «Замовник», «Клієнт» та «Хостинг». Також існує блок типу «Сховище даних» – «Дані». «Бажання» та «Бюджет» слугують зв'язком між замовником та головною роботою. Із системи йдуть дані в зовнішні сутності «Клієнт» – Книга, та в «Хостинг» – Розроблений додаток. Сховище даних «Дані» зв'язано з системою двонаправленими стрілками, це означає що дані передаються в обох напрямках. Контекстна діаграма системи представлена на рис. 3.9.

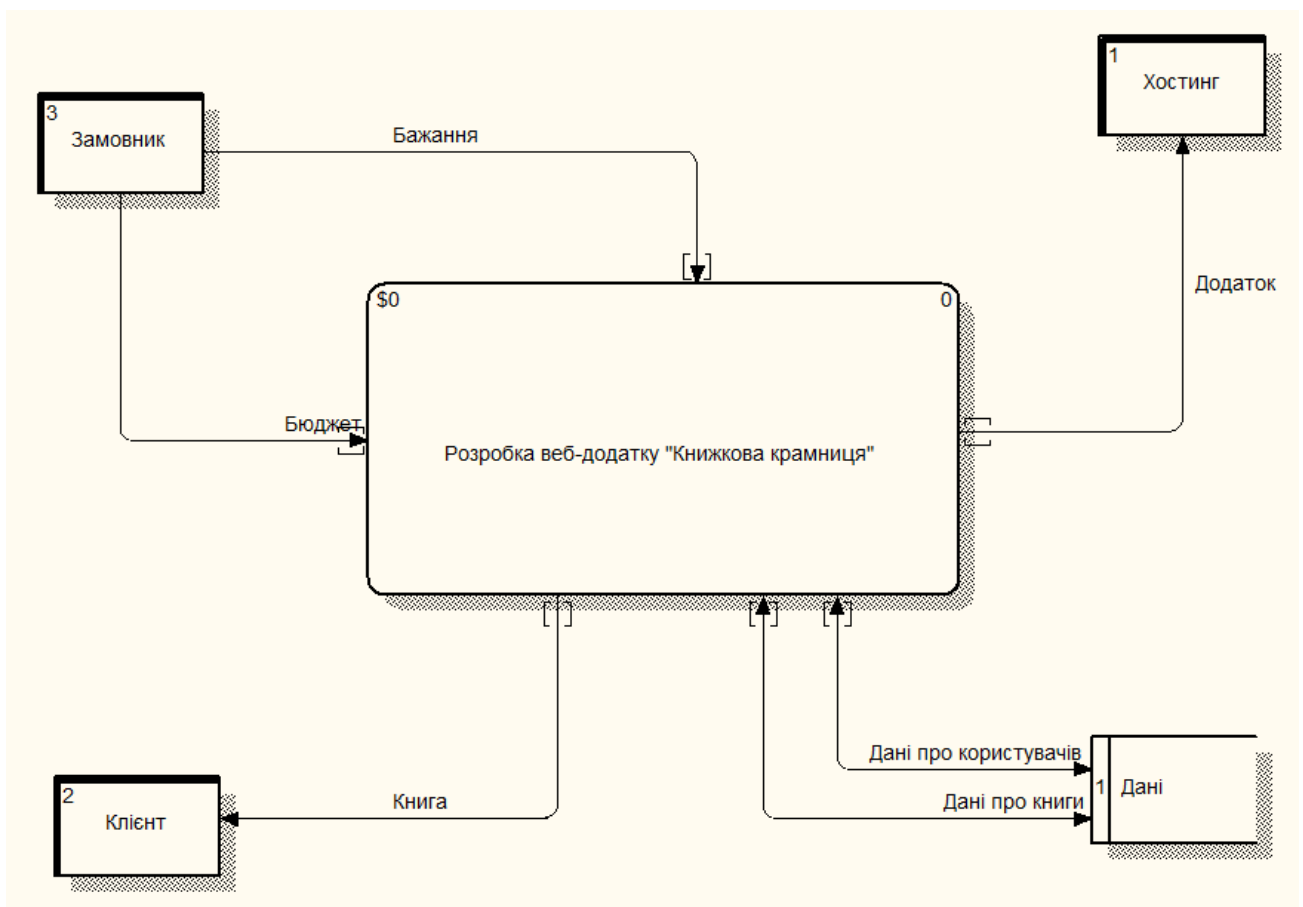


Рисунок 3.9 – Контекстна діаграма розробки веб-додатку «Книжкова крамниця»

Після декомпозиції, перший рівень відображає основні внутрішні процеси та відповідні зовнішні сутності.

Перший процес «Розробка веб-додатку» приймає потік даних з зовнішньої сутності «Замовник» – «Бюджет» і «Бажання», і з сховища даних – «Дані про книги» та «Дані про користувачів». Вихідний потік даних з цього блоку це «Інтерфейс системи», а вихідний потік даних – «Додаток».

Другий блок – «Розміщення додатку на хостингу» також приймає потік даних з зовнішньої сутності «Замовник» – Бюджет і Бажання. Також приймає потік з сутності «Хостинг» – «Особливості». Вхідний потік даних подається з блоку «Розробка веб-додатку». Вихідний потік «Книга». Вихідний потік «Додаток» приймає зовнішня сутність «Хостинг».

Діаграма декомпозиції показана на рис. 3.10.

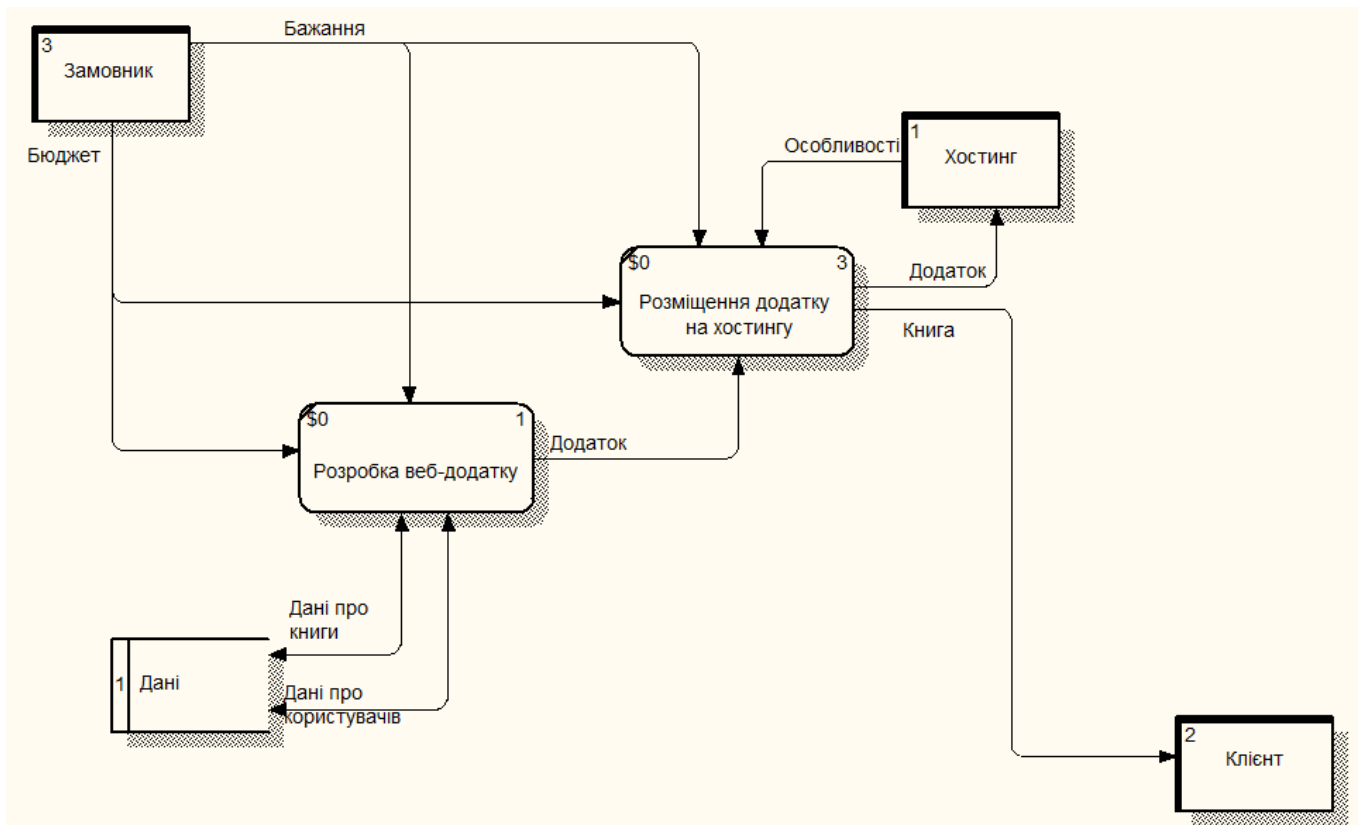


Рисунок 3.10 – Діаграма декомпозиції розробки веб-додатку

В результаті процесу проектування веб-додатку «Книжкова крамниця» були створені діаграми IDEF0, IDEF3 та DFD для представлення веб-додатку у вигляді структурованої моделі та опису її функцій і взаємодії між компонентами.

Діаграма IDEF0 надала детальний опис функцій, взаємозв'язків та ієрархічної структури системи. Діаграми надали більш точний опис потоків даних, інформації, ресурсів та контрольних точок всередині системи.

Діаграма IDEF3 сфокусована на моделюванні процесів та інформаційних потоків всередині системи. Вона деталізує процедури та алгоритми, що використовуються в системі, і описує, які дані обробляються і як вони передаються між різними компонентами системи.

DFD (діаграми потоків даних) були використані для моделювання потоку та обробки даних в системі. Це допомогло визначити зовнішні сутності, які взаємодіють з системою, та визначити потік даних між ними.

Для подальшої програмної реалізації веб-додатку «Книжкова крамниця» діаграми мають бути використані як основа для розробки архітектури додатку та написання коду: Діаграми IDEF0, IDEF3 та DFD деталізують функції, процеси та потоки даних в системі, що полегшує розробку та дозволяє більш ефективно реалізувати систему в реальному середовищі.

### 3.4 Проектування функціональних вимог і бази даних додатку

Для представлення логічної структури веб-додатку «Книжкова крамниця» обрана модель «Сутність-зв'язок».

Сутнісно-реляційні моделі призначені для логічного представлення даних на основі важливої семантичної інформації про реальний світ. Вони визначають значення даних по відношенню до інших даних. Сутності – це об'єкти, які мож-

на описати таким чином, щоб відрізнити їх від інших об'єктів. Сутність – це фактично набір атрибутів, які описують властивості всіх членів даної множини сутностей. Множина (діапазон) значень атрибута називається його областю (доменом). Зв'язок – це відношення, встановлене між декількома сутностями [13].

Для бази даних веб-додатку «Книжкова крамниця» виділимо наступні сутності: Genres (жанри), Authors (автори), Publishers (видавці), Binding (типи обкладинки), Ages (вікові категорії), Users (користувачі), Books (книги), BookTypes (типи книг), Orders (замовлення), Review (відгуки), FavoriteLists (листи бажуючого), UserTypes (типи користувачів), Cities (міста).

Кожна сутність повинна містити атрибути або групи атрибутів, які однозначно ідентифікують кожен екземпляр цієї сутності. Такі атрибути називаються первинними ключами.

При проектуванні бази даних веб-додатку «Книжкова крамниця» були розглянуті сутності та їх атрибути.

Сутність «Authors» (автори) містить наступні атрибути (табл. 3.1):

- author\_id – ідентифікатор автора;
- author\_secondName – прізвище автора;
- author\_firstName – ім'я автора;
- author\_thirdName – по-батькові автора.

Таблиця 3.1 – Сутність «Authors»

№	Атрибут	Тип
1	author_id	integer
2	author_secondName	varchar(50)
3	author_firstName	varchar(50)
4	author_thirdName	varchar(50)

Сутність «Genres» (жанри) містить наступні атрибути (табл. 3.2):

- genre\_id – ідентифікатор жанру;
- genre\_name – назва жанру.

Таблиця 3.2 – Сутність «Genres»

№	Атрибут	Тип
1	genre_id	integer
2	genre_name	varchar(50)

Сутність «Publishers» (видавництва) містить наступні атрибути (табл. 3.3):

- publisher\_code – код видавництва (ідентифікатор);
- publisher\_name – назва видавництва;
- city\_id – ідентифікатор міста видавництва;
- publisher\_email – адреса електронної пошти видавництва.

Таблиця 3.3 – Сутність «Publishers»

№	Атрибут	Тип
1	publisher_code	integer
2	publisher_name	varchar(50)
3	publisher_phone	varchar(50)
4	city_id	integer
5	publisher_email	varchar(50)

Сутність «Binding» (типи обкладинки) містить наступні атрибути (табл. 0):

- binding\_id – ідентифікатор типу обкладинки;
- binding\_type\_name – назва типу обкладинки.



Таблиця 3.4 – Сутність «Binding»

№	Атрибут	Тип
1	binding_id	integer
2	binding_type_name	varchar(50)

Сутність «Orders» (замовлення) містить наступні атрибути (табл. 3.5):

- order\_id – ідентифікатор замовлення;
- book\_id – ідентифікатор книги;
- quantity – кількість книг;
- total\_price – загальна вартість;
- shipping\_info – інформація про доставку;
- order\_info – додаткова інформація про замовлення.

Таблиця 3.5 – Сутність «Orders»

№	Атрибут	Тип
1	order_id	integer
2	book_id	integer
3	quantity	integer
4	total_price	integer
5	shipping_info	varchar(100)
6	order_info	varchar(100)

Сутність «Users» (користувачі) містить наступні атрибути (табл. 3.6):

- user\_id – ідентифікатор користувача;
- user\_sname – прізвище користувача;
- user\_fname – ім'я користувача;
- user\_tname – по-батькові користувача;
- user\_sex – стать користувача;
- user\_pass – пароль користувача;

- user\_phone – телефон користувача;
- user\_email – електронна пошта користувача;
- user\_birthDate – дата народження користувача;
- user\_regDate – дата реєстрації користувача;
- userType\_id – ідентифікатор типу користувача;
- city\_id – ідентифікатор міста видавництва.

Таблиця 3.6 – Сутність «Users»

№	Атрибут	Тип
1	user_id	integer
2	user_sname	varchar(50)
3	user_fname	varchar(50)
4	user_tname	varchar(50)
5	user_sex	varchar(50)
6	user_login	varchar(50)
7	user_pass	varchar(50)
8	user_phone	varchar(50)
9	user_email	varchar(50)
10	user_birthDate	date
11	user_regDate	date
12	userType_id	integer
13	city_id	integer

Сутність «Books» (книги) містить наступні атрибути (табл. 3.7):

- book\_id – ідентифікатор книги;
- genre\_id – ідентифікатор жанру;
- book\_title – назва книги;
- author\_id – ідентифікатор автора;
- publisher\_code – код видавця;
- book\_year – рік видання;

- book\_pages – кількість сторінок;
- book\_type\_id – ідентифікатор типу книги;
- binding\_id – ідентифікатор типу обкладинки
- book\_isbn – код ISBN книги;
- book\_quantity – кількість книг у наявності.

Таблиця 3.7 – Сутність «Books»

№	Атрибут	Тип
1	book_id	integer
2	genre_id	integer
3	book_title	varchar(50)
4	author_id	integer
5	publisher_code	integer
6	book_year	integer
7	book_pages	integer
8	book_type_id	integer
9	binding_id	integer
10	book_isbn	varchar(50)
11	book_quantity	integer

Сутність «BookTypes» (типи книг) містить наступні атрибути (табл. 3.8):

- book\_type\_id – ідентифікатор типу книги;
- book\_type\_name – назва типу книги.

Таблиця 3.8 – Сутність «BookTypes»

№	Атрибут	Тип
1	book_type_id	integer
2	book_type_name	varchar(50)

Сутність «Reviews» (відгуки) містить наступні атрибути (табл. 3.9):

- review\_id – ідентифікатор відгуку;
- book\_id – ідентифікатор книги;
- user\_id – ідентифікатор користувача;
- review\_text – текст відгуку;
- review\_rate – оцінка.

Таблиця 3.9 – Сутність «Reviews»

№	Атрибут	Тип
1	review_id	integer
2	book_id	integer
3	user_id	integer
4	review_text	varchar(500)
5	review_rate	integer

Сутність «UserTypes» (списки бажуючого) містить такі атрибути (табл. 3.10):

- userType\_id – ідентифікатор типу користувача;
- type\_name – назва типу користувача;
- access\_to\_admin\_panel – доступ до адмін-панелі;
- access\_to\_app – доступ до авторизації;
- access\_to\_database – доступ до бази даних;

Таблиця 3.10 – Сутність «UserTypes»

№	Атрибут	Тип
1	userType_id	integer
2	type_name	varchar(50)
3	access_to_admin_panel	yes/no
4	access_to_app	yes/no
5	access_to_database	yes/no

Сутність «FavoriteLists» (списки бажаного) містить такі атрибути (табл. 3.11):

- list\_id – ідентифікатор списку;
- book\_id – ідентифікатор книги;
- user\_id – ідентифікатор користувача;
- listName – назва списку.

Таблиця 3.11 – Сутність «FavoriteLists»

№	Атрибут	Тип
1	list_id	integer
2	book_id	integer
3	user_id	integer
4	listName	varchar(50)

Після того, як були визначили всі основні сутності та атрибути бази даних, можна визначити зв'язки між сутностями. Представимо базу даних у вигляді моделі «сутність-зв'язок» (рис. 3.11).

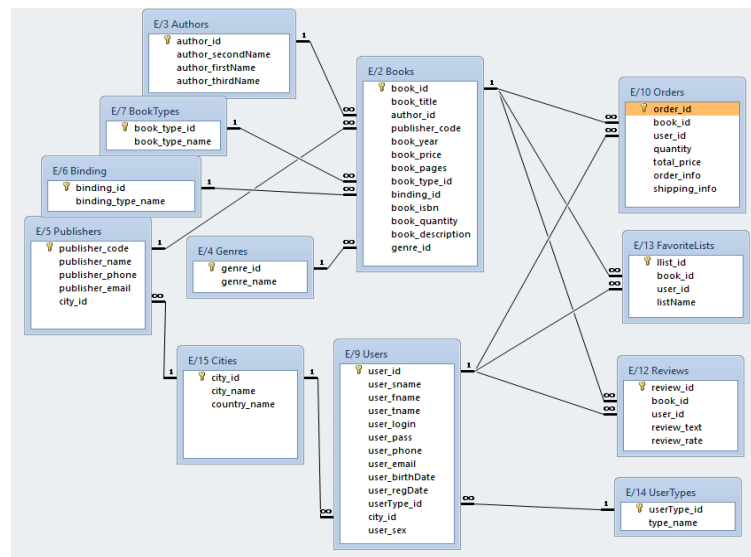


Рисунок 3.11 – Діаграма «сутність зв'язок»

На основі проведеного аналізу та розробки діаграми «сутність-зв'язок» для веб-додатка «Книжкова крамниця» можна зробити такі висновки. База даних являє собою добре структуровану систему, що відображає основні сутності, їхні атрибути та зв'язки між ними. Розроблена модель бази даних враховує основні компоненти книжкової крамниці, такі як книги, автори, видавництва, жанри та клієнти. Зв'язки між сутностями дають змогу ефективно організувати зберігання і доступ до інформації, а також підтримувати цілісність даних. Діаграма «сутність-зв'язок» є основою для створення та подальшого розвитку бази даних, що відповідає потребам веб-додатку.

## 4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБ-ДОДАТКУ «КНИЖКОВА КРАМНИЦЯ»

### 4.1 Керівництво користувача-клієнта

#### 4.1.1 Загальні функції інтерфейсу системи

Коли користувач заходить в додаток, він бачить головну сторінку. Її зміст та вигляд повинен бути зрозумілим користувачу. У верхній частині розміщена шапка з меню навігації, пошуком, кошиком замовлень та кнопкою управління акаунтом користувача, вона присутня на всіх сторінках додатку. На головній сторінці під шапкою користувача зустрічає слайдер з вітанням, кнопкою швидкого перегляду та інформацію про поточні знижки. Головна сторінка веб-додатку «Книжкова крамниця» показана на рис. 4.1.

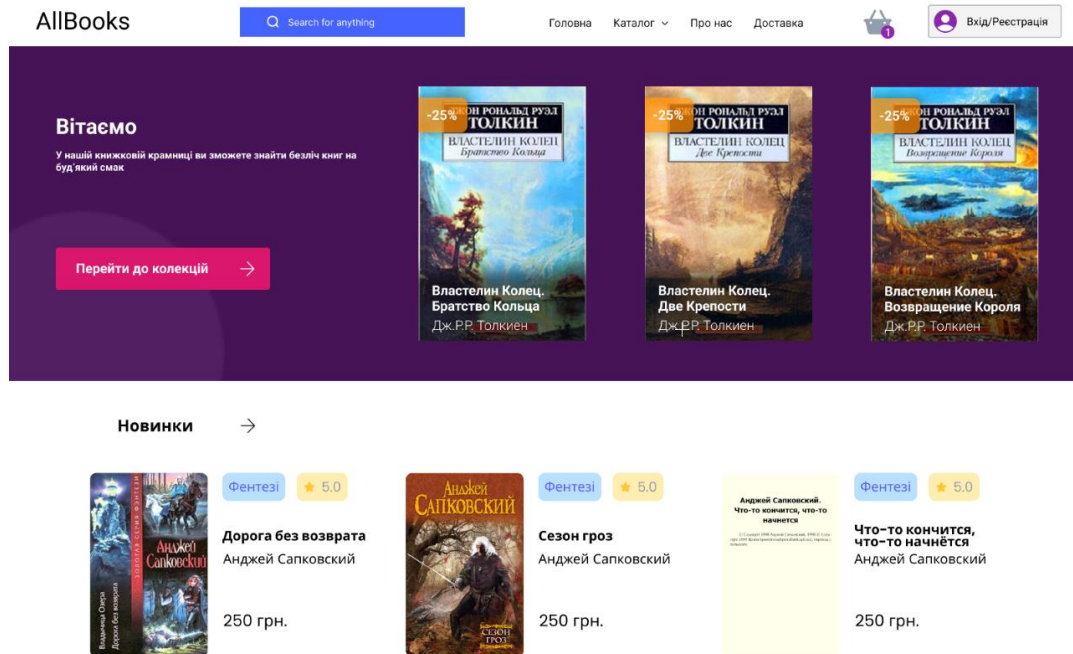


Рисунок 4.1 – Головна сторінка веб-додатку «Книжкова крамниця»

Після слайдеру розміщений блок з новинками. Користувачу представлені три будь-які нових книги. Користувач може натиснути на кнопку «Новинки» та переглянути всі нещодавно додані книги. Якщо користувач наведе курсор на обкладинку він може натисканням на обкладинку перейти до опису книги. Поведінка при наведенні курсору показана на рис. 4.2.



Рисунок 4.2 – Приклад поведінки при наведенні курсору на обкладинку

Далі розміщений блок з бестселерами. Користувач також може натиснути на кнопку «Бестселери» та переглянути всі популярні книги. Блок «Бестселери» показано на рис. 4.3.



Рисунок 4.3 – Блок головної сторінки «Бестселери»



Останнім блоком на головній сторінці є «Футер». У ньому розміщені посилання на сторінки у соціальних мережах, продубльовано меню навігації, а також способи зв'язку з адміністрацією та фізична адреса магазину. У нижній частині футеру є копірайт та рік. Цей блок також присутній на всіх сторінках додатку. Футер показаний на рис. 4.4.

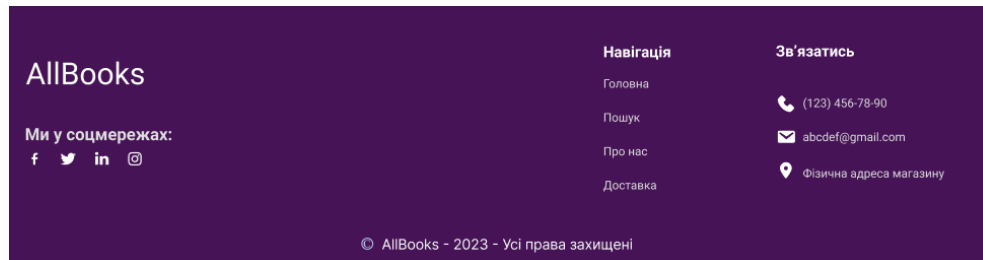


Рисунок 4.4 – Блок «Футер»

#### 4.1.2 Реалізація механізму авторизації у системі

Користувачі, які вперше отримують доступ до додатку, повинні пройти процес реєстрації. Користувач повинен заповнити обов'язкові поля, такі як адреса електронної пошти, пароль, ім'я, прізвище, ім'я користувача (якщо є), номер телефону та іншу особисту інформацію. Користувач повинен створити унікальні облікові дані, які будуть використовуватися для подальшої автентифікації. Після успішної реєстрації користувач може увійти в додаток, використовуючи свої облікові дані. На екрані автентифікації користувач повинен ввести адресу електронної пошти та пароль. Система перевіряє ці дані, збережені під час реєстрації, і, якщо вони правильні, надає доступ до особистого кабінету. Також можна увійти використовуючи обліковий запис Google. На сторінці присутні посилання для реєстрації, відновлення паролю та допомоги, якщо виникли труднощі при авторизації. Форма авторизації показана на рис. 4.5.

Рисунок 4.5 – Форма авторизації у додаток «Книжкова крамниця»

Якщо введені дані будуть не вірні, або користувач не зареєстрований, йому буде виведене повідомлення (рис. 4.6).

Рисунок 4.6 – Повідомлення про невірні дані

Після того як користувач увійде до облікового запису, він буде перенаправлений на головну сторінку. А у шапці з'явиться ім'я користувача (рис. 4.7).



Рисунок 4.7 – Ім'я авторизованого користувача у шапці

При натисканні на ім'я користувача відкриться панель керування обліковим записом. У цій панелі є посилання на поточні замовлення користувача, списком улюблених книг та налаштування облікового запису.

#### 4.1.3 Складові інтерфейсу користувача

Користувач має можливість переглядати книги на головній сторінці або у каталозі. При натисканні на зображення книги відкривається детальна інформація про книгу – її опис, відгуки та інша інформація. а сторінці опису є назва книги, автор та велике зображення, при натисканні на нього, відкриється модальне вікно з фотографіями книги. У правій частині є перемикач типу книги, паперова або електронна, під ним є блок з оцінкою, кількістю відгуків, та кількістю раз додавання до списку бажаного. Під цим блоком є детальні характеристики книги, а біля характеристик присутній блок «Придбати» з статусом, ціною, бажаною кількістю (рис. 4.8).

The screenshot shows the product page for the book "Сезон гріз" by Andrey Sapkovskiy on the AllBooks website. The page layout includes a navigation bar at the top with links for "Головна", "Каталог", "Про нас", and "Доставка", along with a shopping cart icon and a user profile button for "Андрій". The main content area features the book title and author name, a book cover image, and a toggle switch for "Паперова" (Paper) and "Електронна" (Electronic) formats. Below the toggle, there is a rating section showing a 5.0 star rating, 1 review, and 0 likes. A "Характеристики:" (Characteristics) section lists details such as author, publisher, genre (Fantasy), type (Paperback), cover type (Hardcover), page count (384), publication year (2016), language (Ukrainian), ISBN (978-5-17-083919-3), format (84x108/32 (130x200 mm)), and age restriction (16+). To the right, a modal window displays the book's status as "В наявності" (In stock), the price as "250 грн", a quantity selector set to "1", and a purple "Придбати" (Buy) button.

Рисунок 4.8 – Вигляд сторінки детальної інформації

Після зображення та характеристик йде блок з описом, відгуками (рецензіями) та підбіркою схожих книг. Зміна відображення виконуються шляхом перемикача на верхній частині блоку.

Блок «Опис книги» наведено на рис. 4.9.

#### Опис книги Відгуки (1) Схожі

**Сезон гріз** (пол. Sezon burz) - роман у жанрі фентезі польського письменника-фантаста Анджея Сапковського, восьма книга в серії "Відьмак" про пригоди Геральта з Рівії. Роман було вперше випущено польською мовою видавництвом superNOWA 6 листопада 2013 року. "Сезон гроз" не є продовженням роману "Володарка Озера"; його дія відбувається в тому самому часовому проміжку, що й дія оповідань зі збірки "Останнє бажання", між подіями, описаними в оповіданнях "Останнє бажання" та "Відьмак". Частково про деякі події роману згадується в оповіданні "Щось більше".

Події розвиваються в тому ж вигаданому світі, що й інші книги саги про Відьмака: головна дія відбувається в королівстві Керак. Головний герой - відьмак Геральт із Рівії - мисливець за монстрами за наймом, підданий мутаціям і натренований у цитаделі відьмаків Каер Морхені, якого (через спричинену мутаціями зміну вигляду та фізіології) в суспільстві сприймають у кращому разі як менше зло.

Рисунок 4.9 – Блок опису книги

У блоці відгуків є середня оцінка та кількість рецензій з конкретною оцінкою. Натиснувши на кнопку «Написати відгук» користувач може написати рецензію на дану книгу. З кількості відгуків вираховується середня оцінка книги.

Блок «Відгуки» наведено на рис. 4.10.

#### Опис книги Відгуки (1) Схожі

5.0/5



Написати відгук

Андрій



31/05/2023

Остання книга саги про відьмака Геральта з Рівії, хоча насправді не остання, а радше одна з перших. Написана через 14 років після виходу "Володарки озера", вона відрізняється від решти книг серії і сюжетом, і стилем. Мені здалося, що цю книгу написав не сам пан Сапковський, а це фанфік, проте на обкладинці зазначено саме його ім'я. У "Сезоні гроз" буде все, чого душа забажає, крім уже звичного Геральтового характеру і чарівливості оповіді. Тут будуть крадіжка і калажка, наклепи і доноси, інтрижки і зваблення, криваві бійні і розчленування, демони і їхня відсутність, лисиці-перевертні, аукціон, державна зрада і двірцевий переворот, та, зрештою, буря і спасіння лотопельників. Не обійдеться і без жертв і чарівних рятівних втручань. Загалом весь сюжет виглядає якось дуже зіжмакано і рвано, з купкою гачечків цікавіших історій, які так і не були сповна розкриті. Трішки нагадує стиль перших книг циклу, але там принаймні були саме окремі завершені історії, а тут ніби епізоди однієї, проте якось зовсім між собою не пов'язані. Сам Геральт теж якийсь твердолобий і дерев'яний, не схожий на того, до якого ми звикли за 7 книг (хай навіть він час від часу і скиглив щодо своєї місії). Тож якщо вам сподобався основний цикл, "Сезон гроз" можна прочитати по діагоналі, не особливо шукаючи логіки, просто як можливість ще раз побачити улюблених персонажів.

Рисунок 4.10 – Блок відгуків (рецензій)

Останнім блоком є блок «Схожі» який показує три книги у яких такий самий автор, та три книги за таким самим жанром.

Блок «Схожі» наведено на рис. Рисунок 4.11

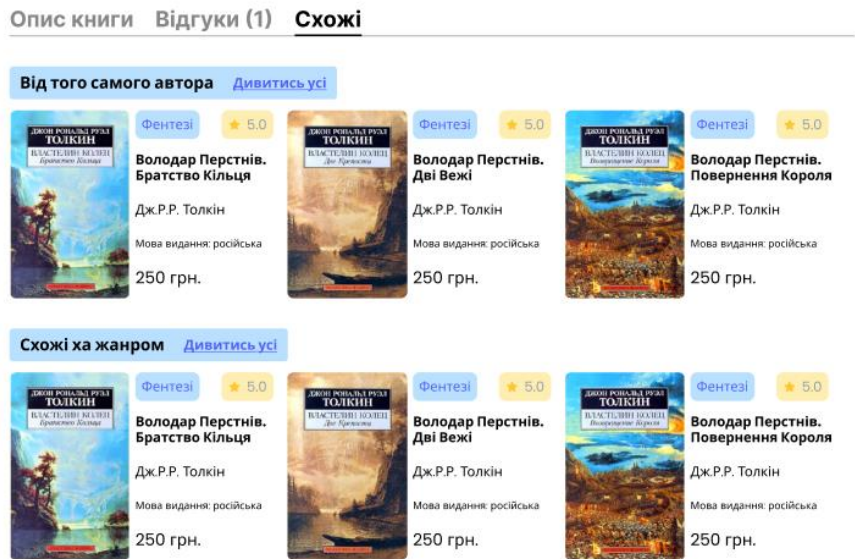


Рисунок 4.11 – Блок «Схожі»

Якщо користувач хоче купити книгу, він натискає на кнопку «Придбати», і після цього, товар додається у кошик, вигляд якого наведено на рис. 4.12.

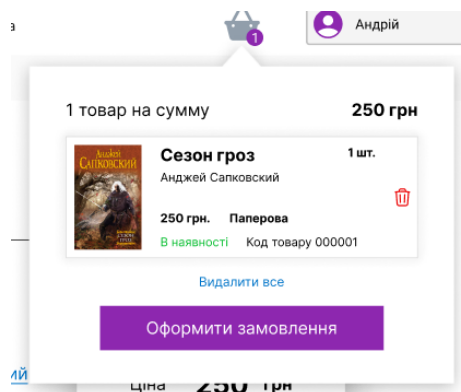


Рисунок 4.12 – Вигляд кошику замовлень

Оформлення замовлення відбувається натисканням кнопки «Оформити замовлення». Якщо користувач авторизований, він потрапить на сторінку оформлення замовлення, у протилежному випадку його він буде перенаправлений на сторінку авторизації.

Сторінка «Оформлення замовлення» представлена на рис. 4.13.

**AllBooks** Головна Каталог Про нас Доставка Андрій

Кошик / Оформлення замовлення

## Оформлення замовлення

**Контактні дані**

Ім'я \*


Прізвище \*

Номер телефон \*

Електронна пошта \*

Передзвоніть мені

**1 товар у кошику** [Редагувати](#)

 **Сезон гроз** 1 шт.  
Андрей Сапковский

250 грн. Паперова  
В наявності Код товару 000001

Разом 250 грн.

**Доставка**

Країна \*

Місто \*

Спосіб доставки \*  
 Відділення Нова Пошта 75 грн.

Адреса відділення \*

**До сплати 325 грн**

1 товар 250 грн  
Доставка 75 грн

Відправляючи замовлення, я підтверджую, що прочитав і згоден(а) з [Умовами використання](#)

**Підтвердити замовлення**

Рисунок 4.13 – Загальний вигляд сторінки «Оформлення замовлення»

На сторінці оформлення у лівій частині замовлення користувач обов'язково повинен заповнити поля відмічені зірочкою, а саме: ім'я, прізвище, телефон, електронну пошту, вибрати спосіб та адресу доставки (рис. 4.14). Якщо користувач бажає щоб йому передзвонив оператор, то йому необхідно відмітити пункт «Передзвоніть мені» у блоці «Контактні дані».

**Доставка**

**Країна \*** Україна

**Місто \*** Одеса

**Спосіб доставки \***

**Відділення Нова Пошта**  
75 грн.

**Адреса відділення \***

Адреса відділення

**Відділення Укрпошта**  
40 грн.

**Поштомат Нова Пошта**  
Безкоштовно

**Кур'єр Нова Пошта**  
95 грн.

**Кур'єр Укрпошта**  
60 грн.

Рисунок 4.14 – Блок з інформацією про доставку

Далі користувач повинен обрати спосіб оплати (рис. 4.15).

**Спосіб оплати**

**Оплата онлайн**

**При отриманні товару**

Рисунок 4.15 Блок вибору способу оплати

Також користувач може за бажанням залишити коментар до замовлення (рис. 4.16).

Рисунок 4.16 – Блок коментарю до замовлення

У правій частині знаходиться блок з підсумком замовлення, та блок з загальною сумою до сплати.

Після підтвердження замовлення, якщо була обрана оплата картою, користувач перейде до сторінки вводу платіжних даних. На сторінці є форма для введення платіжних даних, та інформація щодо замовлення та вартості.

На цьому етапі ще доступне редагування замовлення, і якщо, користувач бажає додати або видалити книги з кошику, то він може це виконати шляхом натискання на посилання редагувати у блоці товару.

Оплата здійснюється шляхом натискання кнопки «Оплатити».

Сторінка оплати замовлення наведена на рис. 4.17.

### Оплата замовлення

Оплата картою	
Введіть дані платіжної картки та натисніть "Оплатити"	
Ім'я власника картки	Andrii Kostyshyn
Номер картки	1234 4567 8900 0000
Дійсна до	CVC
03 / 25	123
<b>Оплатити</b>	

1 товар у кошику	
	<b>Сезон гроз</b> 1 шт. Анджей Сапковский
250 грн.	Паперова
<a href="#">В наявності</a>	Код товару 000001
Разом	250 грн.
<b>До сплати</b>	<b>325 грн</b>
1 товар	250 грн
Доставка	75 грн

Рисунок 4.17 – Вигляд сторінки оплати замовлення



Якщо усі дані вірні та вистачає коштів, оплата здійсниться, користувач придбає товар і побачить сторінку з успішністю оплати. У протилежному випадку оплата не пройде і користувач буде про це сповіщений.

Повідомлення «Оплата успішна» наведене на рис. 4.18.

## Дякуємо за замовлення

Оплата Успішна



Чекайте повідомлення від оператора

[Повернутись до списку замовлень](#)

Рисунок 4.18 – Вигляд повідомлення успішності оплати

Після успішності оплати користувач має чекати повідомлення від оператора. Повернутись до списку замовлень можливо натисканням кнопки «Повернутись до списку замовлень»

### 4.1.4 Організація пошуку у додатку

Так як веб-додаток повинен містити багато різноманітних книжок, доречно буде організувати зручний пошук. У першу чергу це потрібно користувачу для того щоб знайти бажану книгу.

Коли користувач знаходиться на головній сторінці, він має панель пошуку у шапці, коли користувач введе туди пошуковий запит та натисне на клавіатурі

клавішу Enter, відкриється сторінка «Каталог» з результатами пошуку та фільтрами. Також перехід у каталог здійснюється натисканням кнопки у шапці.

У лівій частині сторінки «Каталог» є панель фільтрів, яка дозволяє відфільтрувати пошук за певними критеріями, такими як: жанр, автор, видавництво, мова видання, рік видання, тип обкладинки, тип книги, рейтинг, та ціновий діапазон. У панелі фільтрів, у полі пошуку можна знайти потрібні критерії (наприклад, жанр або автор) та додати їх до відповідного списку, натисканням на перехрестя критерії можна видалити. В деяких полях, якщо критерії не задані, то відображається параметр «Усі» (рис. 4.19).

**Фільтр**

**Жанр**

Знайти

× Фентезі

× Казки

**Автор**

Знайти

× Анджей Сапkowski

× Дж.Р.Р. Толкієн

**Видавництво**

Знайти

Усі

**Мова видання**

Знайти

Усі

Рисунок 4.19 – Фільтри пошуку

Результати пошуку відображаються у правій частині сторінки. Також зверху є кнопки категорій та панель пошуку (рис. 4.20).

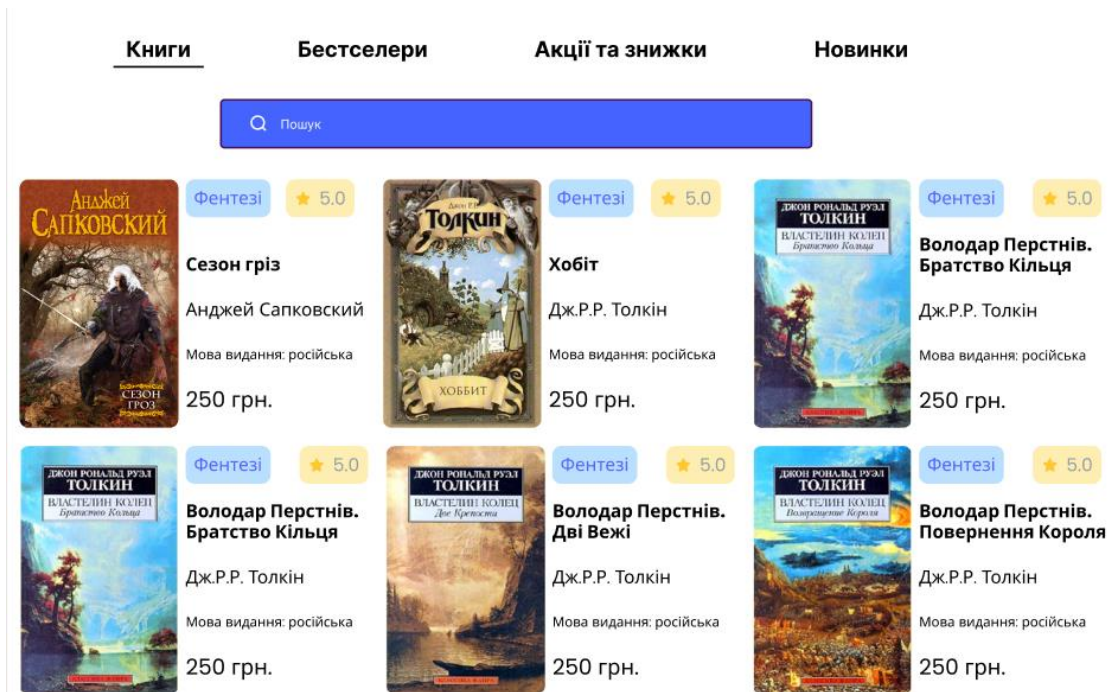


Рисунок 4.20 – Результати пошуку

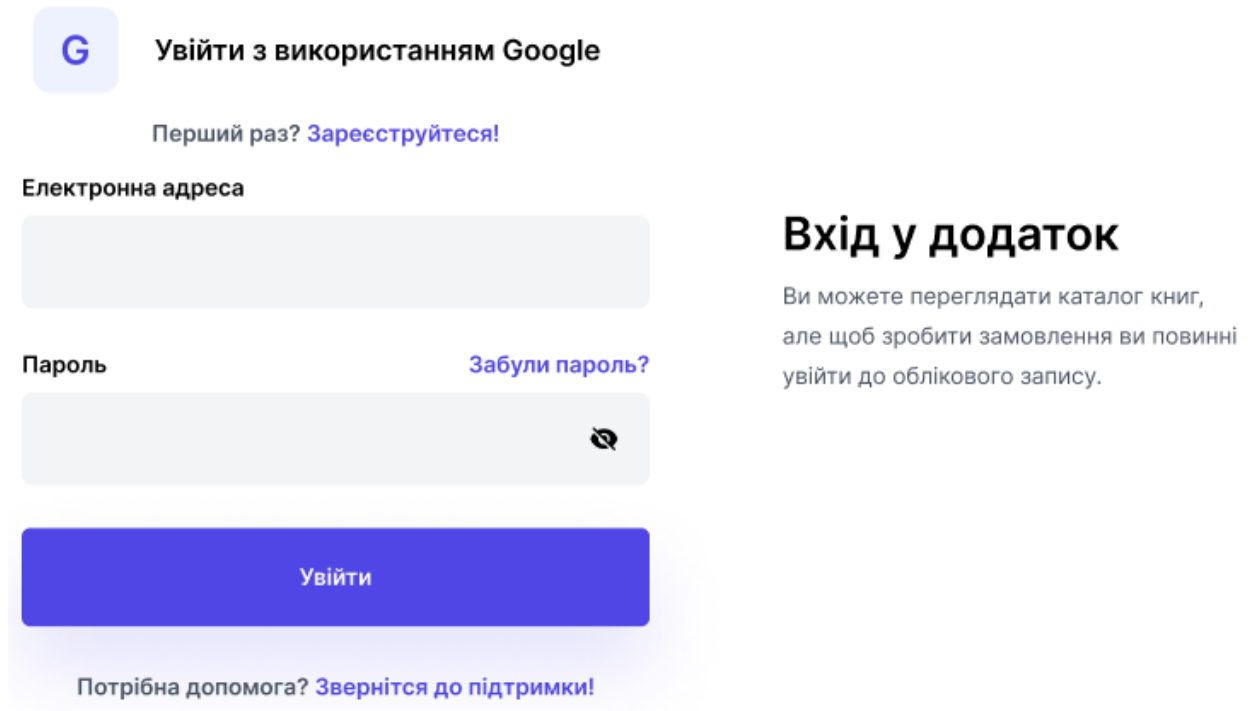
Після того, як користувач знайшов бажану книгу, він може натиснути на зображення книги та перейти до перегляду її характеристик, опису, відгуків та придбання.

## 4.2 Керівництво користувача-адміністратора

У кожному веб-додатку потрібен адміністратор, який може керувати користувачами, мати доступ до функцій обслуговування та можливості наповнення веб-додатку.

### 4.2.1 Реалізація механізму авторизації у системі

Авторизація адміністратора відбувається через форму авторизації, так само як для звичайного користувача (рис. 4.21).



**G** Увійти з використанням Google

Перший раз? [Зареєструйтеся!](#)

Електронна адреса

Пароль [Забули пароль?](#)

Увійти

Потрібна допомога? [Зверніться до підтримки!](#)

### Вхід у додаток

Ви можете переглядати каталог книг, але щоб зробити замовлення ви повинні увійти до облікового запису.

Рисунок 4.21 – Форма авторизації у веб-додатку

Після цього виконується процес перевірки облікового запису на наявність прав адміністратора, і, якщо, користувач є адміністратором то виконується перенаправлення на головну сторінку адмін-панелі.

#### 4.2.2 Складові інтерфейсу-адміністратора

На головній сторінці «Огляд» адмін-панелі присутня статистика щодо додатку. Зверху сторінки картки з кількістю записів в базі даних, а під ними розміщені два графіки зі кількістю замовлень та відвідувачів по місяцям.

Вигляд головної сторінки адмін-панелі наведено на рис. 4.22.

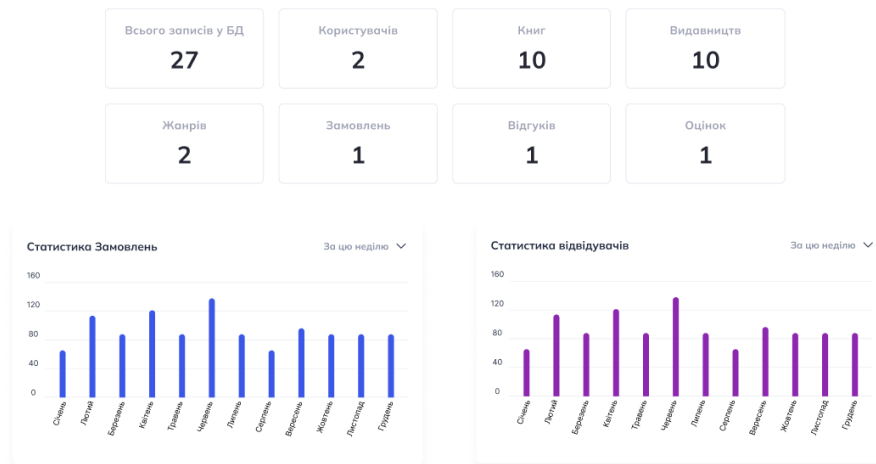


Рисунок 4.22 – Сторінка «Огляд» адмін-панелі веб-додатку

Зверху сторінок розміщена шапка з навігацією, іменем користувача, та кнопкою виходу з системи. Перехід на сторінку бази даних здійснюється через натискання кнопки на панелі навігації. Під шапкою на сторінці керування базою даних розміщена панель кнопок з назвами таблиць, за допомогою них можна перемикатися між таблицями. Вигляд сторінки бази даних наведено на рис. 4.23.

AllBooks | Admin Panel

Огляд Керування базою даних Administrator

**Керування базою даних**

Користувачі | Типи користувачів | Книги | Видавництва | Жанри | Міста | Замовлення | Відгуки | Відгуки

Пошук користувачів Редагувати

№	Прізвище	Ім'я	По-батькові	Стать	Логин	Пароль	Телефон	
1	Костишин	Андрій	Юрійович	Чоловіча	frost	frost	+380993763681	frosted.k
2	Administrator	Administrator	Administrator	Чоловіча	Admin	Admin	+380993763682	adm

< 1 >

Рисунок 4.23 – Сторінка керування базою даних, таблиця «Користувачі»

Зверху таблиці розміщене поле пошуку, яке дозволяє швидко знайти користувача. За замовчуванням адміністратор може тільки переглядати вміст таблиці. Для того щоб перейти у режим редагування таблиці треба натиснути кнопку «Редагувати» у верхньому правому куту таблиці. Після натискання, біля пошуку з'являться додаткові кнопки, такі як «Додати запис», «Видалити запис», «Зберегти зміни», та «Скасувати зміни» (рис 4.24).

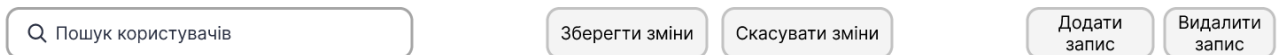


Рисунок 4.24 – Кнопки редагування записів таблиці

При натисканні на кнопку «Додати запис» у таблиці з'явиться нова строка (запис) який буде підсвічено жовтим кольором. Текст комірки відображає тип даних для цієї комірки. Зовнішній вигляд таблиці з новим записом наведено на рис. 4.25.

№	Прізвище	Ім'я	По-батькові	Стать	Логин	Пароль	Телефон	
1	Костишин	Андрій	Юрійович	Чоловіча ▾	frost	frost	+380993763681	frosted.c
2	Administrator	Administrator	Administrator	Чоловіча ▾	Admin	Admin	+380993763682	adm
*	#	Text	Text	Select ▾	Text	Text	Text	

Рисунок 4.25 – Додавання нового запису до таблиці

Деякі значення треба вибрати зі списку, такі поля мають текст «Select». При натисканні на них з'являться варіанти вибору, або, поле пошуку з результа-

том де це необхідно. Приклад випадаючого списку для поля «Стать» наведено на рис. 4.26.

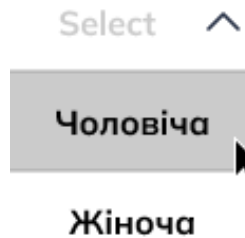


Рисунок 4.26 – Випадаючий список «Стать»

Редагування вже існуючого запису відбувається простим натисканням на нього, то введенням або вибором нового значення. При цьому запис буде помічено як редагований и змінить свій колір на синій. Редагування запису наведено на рис. 4.27.

Пошук користувачів

Зберегти зміни Скасувати зміни

Додати запис Видалити запис

	№	Прізвище	Ім'я	По-батькові	Стать	Логин	Пароль	Телефон	
	1	Редактирование1	Редактирование2	Редактирование3	Чоловіча	frost	frost	+380993763681	frosted
<input type="checkbox"/>	2	Administrator	Administrator	Administrator	Чоловіча	Admin	Admin	+380993763682	adm

< 1 >

Рисунок 4.27 – Редагування запису таблиці

У разі необхідності видалення запису з таблиці адміністратор повинен вибрати необхідні для видалення записи за допомогою чекбоксу у лівій частині таблиці, потім натиснути кнопку «Видалити запис». Після цього запис буде помічений для видалення, та змінить колір на червоний. Запис помічений для видалення наведено на рис. 4.28.

Пошук користувачів

Зберегти зміни Скасувати зміни Додати запис Видалити запис

№	Прізвище	Ім'я	По-батькові	Стать	Логин	Пароль	Телефон		
<input checked="" type="checkbox"/>	1	Костишин	Андрій	Юрійович	Чоловіча	frost	frost	+380993763681	frosted
<input type="checkbox"/>	2	Administrator	Administrator	Administrator	Чоловіча	Admin	Admin	+380993763682	adm

< 1 >

Рисунок 4.28 – Запис, помічений для видалення

Коли редагування буде завершено, потрібно зберегти зміни у таблиці. Для того щоб зберегти зміни потрібно натиснути кнопку з назвою «Зберегти зміни».

У випадку якщо нічого не потрібно змінювати, потрібно натиснути кнопку «Скасувати зміни». Це дозволить вийти з режиму редагування нічого не змінивши у таблиці.

Так як у системі повинні бути присутні користувачі з різними правами доступу, їм треба розмежувати дозвіл. Типи користувачів та їхні права доступу наведені у наступній сторінці «Типи користувачів».

Сторінка «Типи користувачів» наведена на рис. 4.29.

AllBooks | Admin Panel

Огляд Керування базою даних Administrator

Керування базою даних

Користувачі **Типи користувачів** Книги Видавництва Жанри Міста Замовлення Відгуки Відгуки

Пошук типів користувачів Редагувати

№	Назва типу	Доступ до адмін-панелі	Авторизація	Доступ до бази даних
1	Користувач	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Адміністратор	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Заблоковані	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

< 1 >

Рисунок 4.29 – Сторінка з типам користувачів



Сторінка «Типи користувачів» має назву типу: звичайний користувач, адміністратор та заблокований користувач. Звичайний користувач має звичайне право авторизації додатку. Адміністратор має право авторизації, доступ до адмін-панелі, та доступ до редагування бази даних. Заблоковані користувачі не мають доступу до авторизації у додатку.

В результаті була здійснена програмна реалізація книжкової крамниці, яка забезпечує всі необхідні функції, які були оголошені в постановці завдання.

## ВИСНОВКИ

В результаті виконання дипломної роботи було здійснено проектування та програмну реалізацію веб-сервісу «Книжкова крамниця», який надає можливість користувачам, засобами мережі Інтернет, зручно і швидко здійснювати пошук та замовлення книжок, що забезпечує спрощення процесу купівлі книг, забезпечує швидкий та зручний доступ до потрібних видань та покращує якість обслуговування клієнтів.

Основні функції, які має виконувати веб-сервіс, полягають у можливості перегляду каталогу книг, пошуку книг за різними параметрами, замовленні книг, здійснення оплати замовлення та отримання інформації про стан виконання замовлення, що дозволить реалізувати можливості продажу та розповсюдження книжок засобами мережі Інтернет.

При виконанні дипломної роботи проведено аналіз функціонуючих у мережі Інтернет аналогічних систем, визначені основні переваги та недоліки, сформульовані вимоги до веб-сервісу. Визначена архітектура та функції сервісу, здійснено проектування та розробка бази даних та інтерфейсу сервісу, розроблено механізм реєстрації, авторизації та розмежування прав доступу до інформації у додатку, здійснено проектування та програмна реалізація додатку користувачів та адміністратора сервісу.

Веб-сервіс «Книжкова крамниця» реалізовано з використанням сучасних програмних засобів: мова сценаріїв PHP та JavaScript, СУБД MySQL, Web-серверу Apache, мови розмітки HTML та технології CSS, а також бібліотеки інтерфейсів React JS що дозволяє легко розширювати та доповнювати новими функціями вже створений сервіс

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Бред Стоун. Книга Продається все. Джефф Безос та ера Amazon – Пер. з англ. Київ: Наш Формат, 2016 – 400 с.
2. B2B, B2C, B2G, C2C: сегментація та специфіка. URL: <https://avivi.pro/ua/blog/b2b-b2c-b2g-c2c-segmentatsiya-ta-spetsifika/> (дата звернення: 01.05.2023).
3. Kenneth Laudon, Carol Traver. E-commerce 2021–2022: business. technology. society., Global Edition – 17th Edition – Pearson; 17th edition (July 13, 2021) – 359 pages – ISBN 978-1292409313.
4. Тарарака В.Д. Архітектура комп'ютерних систем: навчальний посібник. – Житомир: ЖДТУ, 2018 – 383 с.
5. Scott Hawkins. Apache Web Server Administration and E-Commerce Handbook – Pearson P T R (January 1, 2000) – 359 pages – ISBN 978-0130898739.
6. Apache.org Development Team. Apache Web Server Directives Guidebook – 308 pages – ISBN 978-1583483695.
7. Бази даних MySQL: Навчальний посібник. – Тернопіль: Навчальна книга – Богдан, 2010 – 160 с. ISBN 978-966-10-0906-5.
8. Косентино Кристофер. PHP Web-професіоналам. / Пер. с англ. Київ: BHV, 2001 – 206 с.
9. Julie Meloni. HTML, CSS and JavaScript All in One, Sams Teach Yourself: Covering HTML5, CSS3, and jQuery – 2nd Edition – Sams Publishing; 2nd edition (October 1, 2014) – 704 pages ISBN 978-0672337147.
10. Гужва В. М. Інформаційні системи і технології на підприємствах: Навч. посібник. – К.: КНЕУ, 2001 – 400 с. ISBN 966–574–261–2.

11. С. В. Мінухін, О. М. Беседовський, С. В. Знахур. Методи і моделі проектування на основі сучасних CASE-засобів. Навчальний посібник. Харків: Вид. ХНЕУ, 2008 – 272 с.
12. В.С. Авраменко, А.С. Авраменко. Проектування інформаційних систем: навчальний посібник. Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017 – 434 с. ISBN 978-966-920-208-6.
13. Н. О. Харів. Бази даних та інформаційні системи: навчальний посібник. Рівне: НУВГП, 2018 – 127 с.