

Київський національний торговельно-економічний університет
Кафедра інженерії програмного забезпечення та кібербезпеки

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розробка веб-сайту інтернет-магазину продажу книг

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

підпис студента

Голікова Івана
Володимировича

Науковий керівник
кандидат технічних наук,
доцент

підпис керівника

Рзаєва Світлана
Леонідівна

Гарант освітньої програми
кандидат технічних наук,
доцент

підпис керівника

Цензура Микола
Олександрович

КИЇВ – 2021

Київський національний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь бакалавр

Спеціальність 121 Інженерія програмного забезпечення

Затверджую

Зав. кафедри інженерії
програмного забезпечення
та кібербезпеки

Криворучко О. В.

"20" жовтня 2020 р.

Завдання

на випускна кваліфікаційна робота студентів

Голікова Івана Володимировича

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи Розробка веб-сайту інтернет-магазину продажу книг

Затверджена наказом ректора від "30" жовтня 2020 р. № 3225

2. Строк здачі студентом закінченої роботи _____

3. Цільова установка та вихідні дані до роботи

Мета роботи - створення веб-сайту інтернет-магазину, що спеціалізується на продажу книг та має в собі функціонал, достатній для продуктивної роботи магазину

Об'єкт дослідження - діяльність книжкового магазину

Предмет дослідження - розробка веб-сайту інтернет-магазину продажу книг

4. Консультанти роботи із зазначенням розділів, які консультують:

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

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1 ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНОЇ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-САЙТІВ КНИЖКОВИХ МАГАЗИНІВ

Концепція діяльності книжкового інтернет-магазину

Аналіз методів дослідження діяльності інтернет-магазину

Аналіз предметної області веб-ресурсу

Порівняльний аналіз аналогів

Технічне завдання на веб-сайт книжкового магазину

Висновки до розділу 1

РОЗДІЛ 2 МОДЕЛЮВАННЯ СТРУКТУРИ ПЗ

Функціональна характеристика книжкового інтернет-магазину

Побудова моделі бази даних

Створення таблиць

Висновки до розділу 2

РОЗДІЛ 3 РОЗРОБКА ВЕБ-САЙТУ ІНТЕРНЕТ-МАГАЗИНУ

Головна сторінка сайту

Сторінка сайту «Про нас»

Сторінка каталогу товарів

Сторінка кошику замовлень

Сторінка зворотного зв'язку

Сторінка авторизації

Висновки до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

6. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	21.09.2020	21.09.2020
2.	<i>Вступ та перелік літературних джерел</i>	14.12.2020	14.12.2020
3.	<i>Розділ 1. «Дослідження функціональної структури існуючих веб-сайтів книжкових магазинів»</i>	19.02.2021	19.02.2021
4.	<i>Розділ 2. «Моделювання структури ПЗ»</i>	05.03.2021	05.03.2021
5.	<i>Розділ 3. «Розробка веб-сайту інтернет-магазину»</i>	10.04.2021	10.04.2021
6.	<i>Висновки</i>	24.04.2021	24.04.2021
7.	<i>Здача випускної кваліфікаційної роботи на кафедру (перша перевірка)</i>	14.05.2021	14.05.2021
8.	<i>Підготовка автореферату та презентації доповіді</i>	17.05.2021	17.05.2021
9.	<i>Попередній захист випускної кваліфікаційної роботи</i>	18.05.2021 – 21.05.2021	21.05.2021
10.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	01.06.2021	01.06.2021
11.	<i>Здача прошого випускної кваліфікаційної роботи на кафедру</i>	02.06.2021	02.06.2021
12.	<i>Публічний захист випускної кваліфікаційної роботи</i>		

7. Дата видачі завдання «20» жовтня 2020 р.

8. Науковий керівник випускної кваліфікаційної роботи

Рзасва С. Л.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми

Цензура М. О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент Голіков І. В.

(прізвище, ініціали, підпис)

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

(*nidnuc*, *data*)

Рзаева С. Л.

(ПИБ, підпис, дата)

Випускна кваліфікаційна робота студента Голікова І. В.
(прізвище, ініціали)

Цензура М. О.

(прізвище, ініціали, підпис)

Криворучко О. В.

(підпис, прізвище, ініціали)

« _____ » 20__ г.

АНОТАЦІЯ

Випускна кваліфікаційна робота присвячена проектуванню та розробці веб-сайту інтернет-магазину з продажу книг. Під час виконання етапів випускної кваліфікаційної роботи було проаналізовано основні тенденції у розробці інтернет-ресурсів комерційного спрямування, розглянуто сайти-аналоги вже працюючих інтернет-магазинів з подібним напрямком роботи, виявлено їх недоліки, які було враховано при створенні веб-сайту згідно із заявленим технічним завданням. Було розроблено структуру сайту та створено всі необхідні сторінки проєктуемого веб-ресурсу, які необхідні для забезпечення всіх функціональних засобів, які заявлені в технічному завданні до проєкта.

ANNOTATION

The graduation qualification work is devoted to the design and development of the website for the online book store. During the stages of the final qualification work, the main trends in the development of commercial Internet resources were analyzed, analogous sites of existing online stores with a similar direction of work were considered, their shortcomings were identified and considered during creating a website according to the technical requirements. The structure of the site was developed and all the necessary pages of the projected web resource were created. These pages are necessary to provide all the functional means that are claimed in the project technical requirements.

ЗМІСТ

Перелік умовних скорочень.....	3
ВСТУП.....	4
РОЗДІЛ 1 ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНОЇ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-САЙТІВ КНИЖКОВИХ МАГАЗИНІВ.....	6
1.1. Концепція діяльності книжкового інтернет-магазину.....	6
1.2. Аналіз методів дослідження діяльності інтернет-магазину.....	9
1.3. Аналіз предметної області веб-ресурсу.....	9
1.4. Порівняльний аналіз аналогів.....	11
1.5. Технічне завдання на веб-сайт книжкового магазину.....	14
1.6. Висновки до розділу 1.....	15
РОЗДІЛ 2 МОДЕЛЮВАННЯ СТРУКТУРИ ПЗ.....	17
2.1. Функціональна характеристика книжкового інтернет-магазину..	17
2.2. Побудова моделі бази даних.....	22
2.3. Створення таблиць.....	24
2.4. Висновки до розділу 2.....	26
РОЗДІЛ 3 РОЗРОБКА ВЕБ-САЙТУ ІНТЕРНЕТ-МАГАЗИНУ.....	27
3.1. Головна сторінка сайту.....	27
3.2. Сторінка сайту «Про нас».....	28
3.3. Сторінка каталогу товарів.....	28
3.4. Сторінка кошику замовлень.....	29
3.5. Сторінка зворотного зв'язку.....	29
3.6. Сторінка авторизації.....	30
3.7. Висновки до розділу 3.....	31
ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....	32
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	33
ДОДАТКИ.....	35

					КНТЕУ 121 06-05.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка веб-сайту інтернет-магазину продажу книг	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		20.10.20		Зміст	2	34
Керівник		Рзаєва С.Л.		20.10.20		Факультет інформаційних технологій, 4 курс, 6 група		
Гарант		Цензура М.О.		20.10.20				
Розроб		Голіков І.В.		20.10.20				
					Зміст			

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

Рис. – Рисунок.

СКБД – Система керування базою даних.

ПЗ – Програмне забезпечення.

ISBN – International Standard Book Number – міжнародний стандартний номер книги.

УДК – Універсальна десяткова класифікація.

IBM – International Business Machines Corporation – американська електронна корпорація.

WPT – Web Performance Tool.

SQL – Structured query language – мова структурованих запитів.

MS – Microsoft – американська корпорація.

IDE – Integrated Development Environment – інтегроване середовище розробки.

HTML – Hyper Text Markup Language – мова розмітки гіпертексту.

CSS – Cascading Style Sheets – каскадні таблиці стилів.

JS – JavaScript – мова програмування.

					КНТЕУ 121 06-05.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка веб-сайту інтернет-магазину продажу книг	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		20.10.20		ПС	3	34
Керівник		Рзаєва С.Л.		20.10.20		Факультет інформаційних технологій, 4 курс, 6 група		
Гарант		Цензура М.О.		20.10.20				
Розроб		Голіков І.В.		20.10.20	ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ			

ВСТУП

В сучасному світі науково-технічний прогрес змушує застосовувати передові досягнення та розробки вчених практично у всіх сферах народного господарства.

На сьогоднішній день інтернет-комерція отримала широкий розвиток. У мережі Інтернет можна купити практично будь-яку продукцію або послугу.

Сайт інтернет-магазину є, по суті, звичайним магазином, перенесеним в Інтернет. Зазвичай інтернет-магазин надає можливість ознайомитися з переліком товарів і тут же зробити замовлення вподобаного товару.

Щороку кількість інтернет-магазинів збільшується, оскільки це дійсно прибутково і зручно для покупця, а також інтернет-магазин економить бюджет і час. Інтернет-магазин працює цілодобово і може продавати певні товари в автоматичному режимі без участі продавця.

В юридичній площині діяльність об'єктів інтернет-комерції має бути визначена згідно класифікації видів економічної діяльності (КВЕД).

КВЕДи які застосовуються для здійснення торгівлі через інтернет:

КВЕД 47.91: Роздрібна торгівля що здійснюється фірмами поштового замовлення або через мережу Інтернет.

КВЕД 46.19: Діяльність посередників у торгівлі товарами широкого асортименту.

КВЕД 63.12: Веб-портали. (Керування веб-сайтами, які використовують пошукові механізми задля створення та підтримки великих баз даних інтернет-адрес і контенту в зручному для пошуку форматі, керування іншими веб-сайтами, що функціонують як портали в мережі інтернет, таких як медійні сайти, що періодично поновлюють інформаційний контент).

					<i>КНТЕУ 121 06-05.БР</i>			
Зм.	Аркуш	№ докум	Підпис	Дата	<i>Розробка веб-сайту інтернет-магазину продажу книг</i>	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		14.12.20		В	4	34
Керівник		Рзаєва С.Л.		14.12.20		Факультет інформаційних технологій, 4 курс, 6 група		
Гарант		Цензура М.О.		14.12.20				
Розроб		Голіков І.В.		14.12.20				
					<i>ВСТУП</i>			

Світовий ринок автоматизованих систем теж займається розробкою програмних продуктів в галузі книжкової торгівлі. Так компанією IBM® створено додаток “книжковий магазин” IBM® zipSeries® за допомогою сімейства програмних продуктів IBM® WebSphere®. В даний час воно працює на серверах IBM Series p® з ПЗ IBM® AIX® V 5, хоча створювалося для роботи на будь-якому класі серверів IBM (в тому числі IBM Series i, p, x, z®). До складу IBM® Web Performance Tool (WPT) входять автоматичні Web-клієнти (роботи), що моделюють дії кінцевих користувачів. У випадку з додатком «книжковий магазин» zipSeries, ці роботи звертаються до edge-серверів WebSphere і моделюють дії користувачів (покупців книгарні), виконуючи пошук і покупку книг.

Актуальністю розробки проєкту є створення повноцінної інформаційної системи, яка поєднує в собі інтерфейс користувача та бази даних книжкового магазину.

Мета дослідження – створення веб-сайту інтернет-магазину, що спеціалізується на продажу книг та має в собі функціонал, достатній для продуктивної роботи магазину.

Об’єктом дослідження є діяльність книжкового магазину.

Предмет дослідження – веб-сайт інтернет-магазину, розроблений з використанням технології, структури, засобів та методів розробки.

Завдання випускної кваліфікаційної роботи:

- огляд існуючих веб-сайтів книжкового ринку;
- розробка технічного завдання проєкту;
- проєктування структури веб-сайту;
- розробка веб-сайту книжкового магазину, який би відповідав визначеним критеріям та дозволяв отримати бажаний результат в процесі роботи магазину.

Процес розробки сайту книжкового магазину відбувається із застосуванням таких методів дослідження, як аналіз, синтез та системний метод.

									Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата					5

РОЗДІЛ 1

ДОСЛІДЖЕННЯ ФУНКЦІОНАЛЬНОЇ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-САЙТІВ КНИЖКОВИХ МАГАЗИНІВ

1.1. Концепція діяльності книжкового інтернет-магазину

Так як книжковий магазин є суб'єктом господарської діяльності, що займається реалізацією друкованої продукції, то ми маємо проаналізувати усі аспекти, що відносяться до зазначеної галузі. Саме дослідивши та проаналізувавши їх можна буде визначити чіткі критерії для створення програмного продукту, що задовольнятиме усім сформованим вимогам.

Ключову роль у функціонуванні книготорговельної мережі відіграє асортиментна робота, адже саме вона визначає ступінь відповідності представлених товарів попиту населення. Чим ефективніший процес формування асортименту книготорговельного підприємства, тим кращою буде його діяльність за рахунок збільшення виручки від реалізації та обсягу прибутків.

Товарна політика є однією з найважливіших складових комплексу маркетингу торговельного підприємства. Поняття асортименту загалом характеризує склад товарної маси, що реалізується торговельним підприємством, та являє собою повний перелік товарів у розрізі груп, підгруп, видів, різновидів тощо. Відповідно, асортиментна політика книготорговельного підприємства – це комплекс заходів щодо управління асортиментом товарів та інструмент реалізації його стратегічних і тактичних цілей, який включає принципи добору друкованої продукції з урахуванням специфіки роботи книгарень. Розробка асортиментної політики – один з основних етапів комерційної роботи книгарні. При дослідженні питання сучасної асортиментної політики необхідно визначити ступінь розробки даної теми.

					<i>КНТЕУ 121 06-05.БР</i>		
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка веб-сайту інтернет-магазину продажу книг Дослідження функціональної структури існуючих веб-сайтів книжкових магазинів Стадія РІ Аркуш 6 Аркуші в 34 Факультет інформаційних технологій, 4 курс, 6 група		
Зав. каф.		Криворучко О.В.		19.02.21			
Керівник		Рзаєва С.Л.		19.02.21			
Гарант		Цензура М.О.		19.02.21			
Розроб		Голіков І.В.		19.02.21			

Досить важливою для функціонування книгарні є нормативно-правова база, що регламентує її діяльність у правовому полі держави. В Україні вона ґрунтується на переліку законів та підзаконних актів, поданих в таблиці 1.1.

Таблиця 1.1.

Нормативні документи, що регламентують діяльність інтернет-магазину

Нормативний документ	Призначення
1	2
Закон Про державну реєстрацію юридичних осіб, фізичних осіб-підприємців та громадських формувань	Основний нормативний акт, який регулює діяльність суб'єктів господарювання в Україні
Закон Про електронну комерцію	Базовий нормативний документ, який регламентує здійснення інтернет-торгівлі, правові вимоги до неї
Закон Про захист прав споживачів	Регулює відносини між споживачами товарів, робіт і послуг та виробниками і продавцями товарів, виконавцями робіт і надавачами послуг різних форм власності, встановлює права споживачів, а також визначає механізм їх захисту
Правила № 103	Визначають відносини між споживачем і суб'єктом господарювання – продавцем товарів на замовлення та поза торговельними або офісними приміщеннями на підставі договору купівлі-продажу, укладеного на відстані чи поза торговельними або офісними приміщеннями, а також регламентують вимоги в дотриманні прав споживачів щодо належної якості, безпеки і належного торговельного обслуговування
Правила № 104	Регламентують порядок приймання, зберігання, підготовки до продажу та продажу непродовольчих товарів через роздрібну торговельну мережу, а також визначають вимоги щодо дотримання прав споживачів стосовно належної якості та безпеки товарів і рівня торговельного обслуговування

1	2
Правила № 185	Регламентують порядок приймання, зберігання, підготовки до продажу і продаж продовольчих товарів через роздрібну торговельну мережу, а також визначають вимоги в дотриманні прав споживачів щодо якості й безпеки товару та рівня торговельного обслуговування
Порядок № 833	Визначає загальні умови провадження торговельної діяльності суб'єктами оптової торгівлі, роздрібною торгівлі, закладами ресторанного господарства, основні вимоги до торговельної мережі, мережі закладів ресторанного господарства і торговельного обслуговування споживачів (покупців)

Ключові проблеми асортиментної роботи в роздрібній книготорговельній мережі досить чисельні. Високі показники господарсько-фінансової діяльності та культура торговельного обслуговування прямо залежать від постійної наявності в продажу широкого і стійкого асортименту товарів. Саме тому питання асортиментної діяльності виходить на провідні позиції при вивченні діяльності книжкових магазинів. Найпершою складовою товарної політики підприємства є формування асортименту. Нині варто наголосити на потребі впровадження фахової термінології. Термін «асортимент» (від французького «assortiment») означає підбір різних видів і сортів товарів. Торговельний асортимент – це набір товарів, сформований організацією торгівлі з урахуванням її спеціалізації, споживчого попиту і матеріально-технічної бази. Вважаємо, що формування асортименту – це процес добору й визначення номенклатури товарів, тематичних розділів, типів, видів і різновидів літератури за різними ознаками, спрямований на задоволення попиту населення та забезпечення високої прибутковості роботи підприємств торгівлі. Для прибуткової діяльності книгарні він повинен відповідати вимогам

максимального задоволення попиту населення з мінімальними можливими витратами часу на придбання товарів.

1.2. Аналіз методів дослідження діяльності інтернет-магазину

Досліджуючи діяльність роботи книжкового магазину методом аналізу ми відокремлюємо різні сегменти його діяльності. Так першочерговим заходом є оцінка екземпляра книжки з урахуванням її цінності та ринкової вартості, потім розраховується націнка та додається дана книжка до асортиментного переліку. Коли з'являється покупець на даний товар – проходить етап розрахунку та виведення даної позиції з бази доступних до продажу.

Якщо для дослідження брати окремі елементи ланцюжка побудови структури діяльності книжкового магазину, такі як прийняття на баланс відповідного примірника, виставлення його у продаж і сама операція реалізації, то досліджуючи ці елементи разом із зв'язками взаємодії – отримаємо метод синтезу. Результатом синтезу є абсолютно нове утворення, властивості якого є не тільки зовнішнє з'єднання властивостей компонентів, але також і результат їх внутрішнього взаємозв'язку і взаємозалежності.

Якщо організовувати побудову системи взаємозв'язків елементів та складових об'єкта, ми отримаємо системний метод дослідження. У випадку з розробкою книжкового інтернет-магазину при застосуванні такого методу потрібно виділити усі елементи системи, вирахувати усі можливі взаємозв'язки між цими складовими і запустити ці процеси, як єдину систему.

1.3. Аналіз предметної області веб-ресурсу

На сьогодні практично кожна комерційна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій – це необхідний чинник існування, що дозволяє розширити межі комерційної діяльності та залучити тим самим нових клієнтів. А для інтернет-магазинів це основний засіб представлення товару покупцям.

Структурована інформація сайту дає змогу партнерам і клієнтам отримати повне уявлення про підприємство і його діяльність. Споживач одержує з рекламних банерів сайту і комерційних та ділових видань багато

					КНТЕУ 121 06-05.БР	Аркуш
						9
Зм.	Аркуш	№ докум	Підпис	Дата		

корисних Веб-адрес, з яких можна дізнатися ціни, замовити товари. Такий спосіб передачі повідомлень потенційним клієнтам ефективніший за звичайні засоби комунікації: пошту, телефон, факс. За допомогою таких обов'язкових елементів, як інтерфейс або навігація, клієнт може самостійно вибрати необхідну для вивчення інформацію. Сайт є доповненням до рекламної кампанії, забезпечуючи можливість зворотного зв'язку й інтерактивної роботи з даними підприємства.

Процес створення і розробки сайтів включає в себе:

затвердження первинного технічного завдання на розробку сайту;

визначення структурної схеми сайту – розташування інформаційних розділів, контенту і навігації;

веб-дизайн – створення графічних елементів макету сайту, стилів і елементів навігації;

розробку програмного коду, модулів, бази даних та інших елементів сайту необхідних в проєкті;

тестування і підключення сайту до мережі Інтернет.

Взявши за приклад сайти вже існуючих інтернет-магазинів з продажу книг, розробимо свій, врахувавши вимоги ТЗ до представленого проєкту.

Для початку розглянемо такі поняття як: електронний бізнес, електронна комерція, електронна торгівля та інтернет-магазин.

Електронний бізнес – модель організації бізнесу, в якій бізнес-процеси, обмін бізнес-інформацією та комерційні транзакції працюють за допомогою інформаційних систем. Значна частина рішень використовує інтернет-технології для передачі даних і надання веб-сервісів.

Транзакції – група послідовних операцій, яка являє собою логічну одиницю роботи з даними. Транзакція може бути виконана або цілком і успішно, дотримуючись цілісності даних, або не виконана взагалі, тоді вона не зробить ніякої дії.

Електронна комерція (e-commerce) – термін, що використовується для позначення комерційної активності в мережі Інтернет. Забезпечує можливість

					КНТЕУ 121 06-05.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		10

здійснення покупок, продажів, сервісного обслуговування, проведення маркетингових заходів шляхом використання комп'ютерних мереж. Електронна комерція (у широкому сенсі) – підприємницька діяльність по здійсненню комерційних операцій з використанням електронних засобів обміну даними. Об'єктами електронної комерції є те, на що спрямована діяльність систем електронної комерції. До них можна віднести різні товари, послуги та інформацію.

Електронна торгівля є частиною електронної комерції, а значить і електронного бізнесу. Електронна торгівля (англ. Electronic trading, eTrading, e-Trading) – здійснення торговельно-закупівельної діяльності через Інтернет.

Основна мета будь-якого комерційного сайту полягає в обслуговуванні користувачів способом, який принесе йому вигоду. Мета поширення інформації шляхом подання на сайті полягає в тому, щоб спробувати підштовхнути людей придбати продукцію або послуги компанії. Кінцевий результат використання інтернет-магазинів полягає в заохоченні інтересу до здійснення ділової операції.

1.4. Порівняльний аналіз аналогів

На сьогоднішній день існує велика кількість інтернет-магазинів з продажу книг. Кожен з них індивідуальний, так як має своє спрямування та концепцію.

Представлений інтернет-магазин проєктується для запровадження бізнесу з реалізації книг.

Проаналізуємо аналоги веб-сайтів трьох інтернет-магазинів у галузі продажу книг: «Буква», «Книгарня Є», «Книжечка».

Розглянемо Інтернет-магазин «Буква». Веб-адреса ресурсу <https://bukva.ua>. Інтерфейс сайту максимально простий та зрозумілий, що зменшує кількість помилок при користуванні сайтом. Дизайн виконаний в стриманому стилі, відсутнє нагромадження елементів, які б відволікали увагу відвідувачів сайту не зважаючи на те, що в асортименті магазину є товари, не пов'язані з літературою, бо основний напрямок роботи це продаж книг.

Сайт є досить функціональним. На головній сторінці присутній фільтр виводу новинок. Також присутній слайдер, який виводить акції магазину. Присутня можливість знаходження магазину по містах. Форма замовлення є досить інтуїтивна та зрозуміла. Також на сайті наявний кошик, що значно спрощує здійснення замовлень. Головна сторінка сайту «Буква» відображена на рис. 1.1.



Рис. 1.1. Головна сторінка інтернет-магазину «Буква»

Наступним аналогом до порівняння вибрано інтернет-магазин «Книгарня Є». Веб-адреса сайту <https://book-ye.com.ua>. Кольорова гама сайту мінімалістична, проте дизайн сайту є не досить естетичним. Рекламний слайдер, що має велику швидкість зміни зображень займає більшу частину сторінки. Відвідувач, дивлячись на даний сайт, не може зосередити свою увагу, що відвертає від головної мети сайту (рис. 1.2).

В центральній частині сайту знаходиться вікно пошуку. Поруч розташовані позначки кошика, вподобання та особистого кабінета.

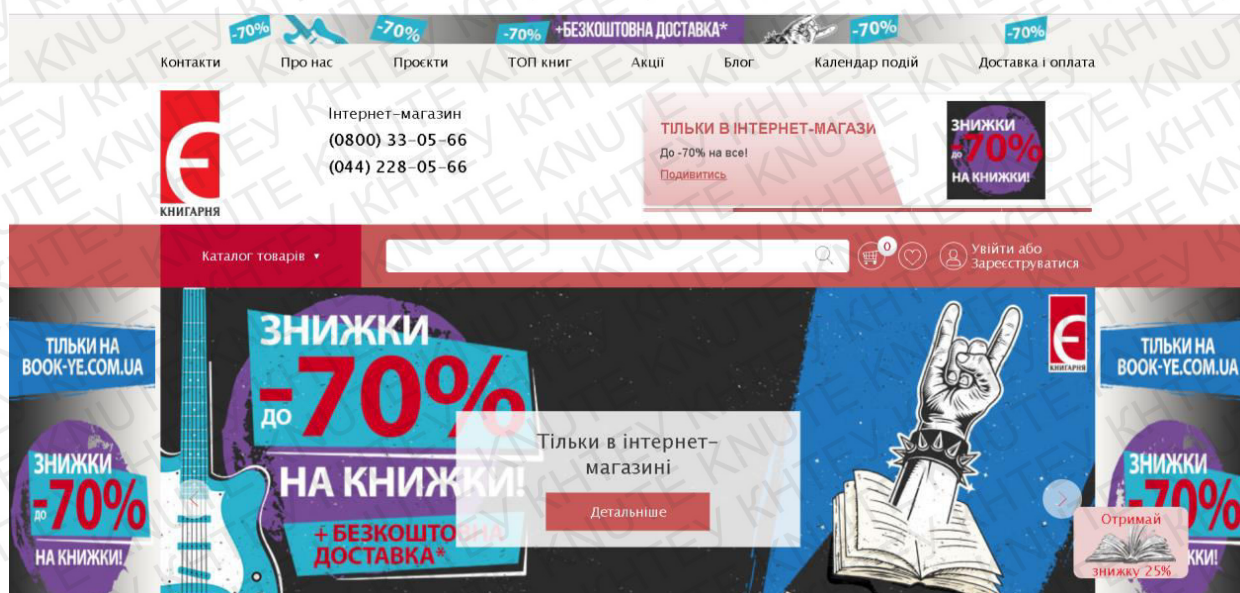


Рис. 1.2. Головна сторінка інтернет-магазину «Книгарня Є»

Розглянемо сайт інтернет-магазину «Книжечка». Веб-адреса сайту <https://in-market.kiev.ua/>. Дизайн сайту має мінімальне оформлення. На головній сторінці присутній рекламний слайдер, рядок розділів сайту, пошукове вікно та адреса офісу. Відсутня авторизація та кошик, а також перемикання мови подання інформації. Безпосередньо сам контент та заголовки відображаються і українською, і російською мовами на одній сторінці. Веб-адреса та назва магазину не співпадають, що створює незручності в пошуку саме цього магазину в мережі. Головна сторінка сайту «Книжечка» зображена на рис. 1.3.

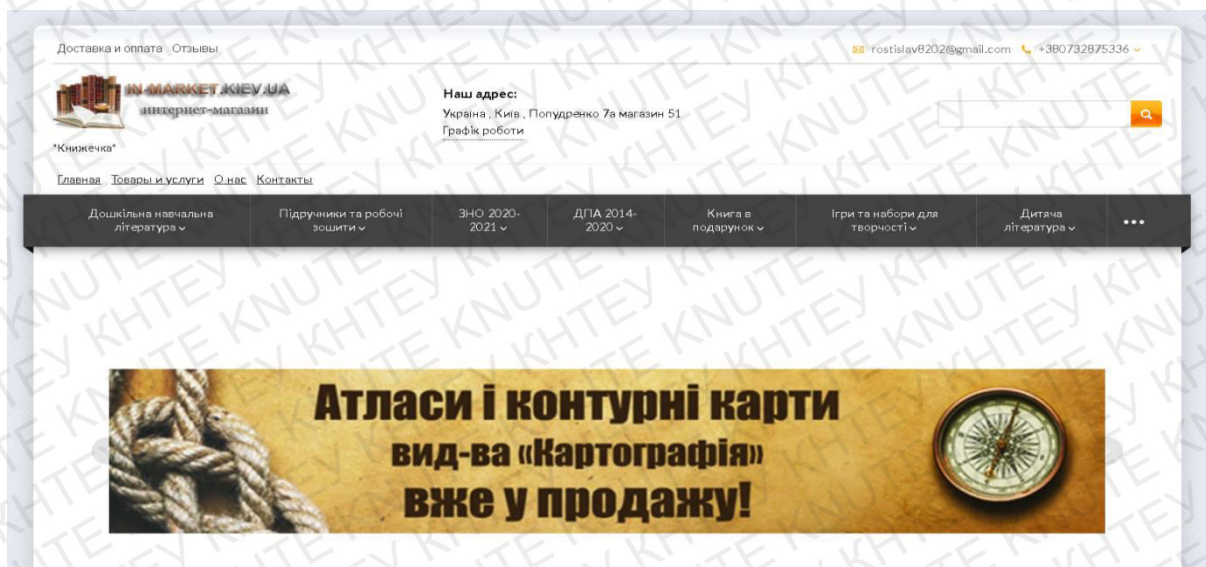


Рис. 1.3. Головна сторінка інтернет-магазину «Книжечка»

Зм.	Аркуш	№ докум	Підпис	Дата

КНТЕУ 121 06-05.БР

Аркуш

13

Усі перераховані веб-сайти дозволяють здійснювати торгівлю в мережі Інтернет. Порівняльний аналіз аналогів наведено в таблиці 1.2.

Таблиця 1.2.

Порівняльний аналіз аналогів

Характеристика	«Буква»	«Книгарня Є»	«Книжечка»
Дизайн	+	-	-
Кошик	+	+	-
Слайдер	+	+	+
Фільтри	+	+	+
Особистий кабінет	+	+	-
Пошук	+	+	+

1.5. Технічне завдання на веб-сайт книжкового магазину

Завданням проєкту є розробка веб-сайту, призначеного для роботи інтернет-магазину з продажу книг.

1.5.1. Вимоги до оформлення та структури сайту.

1.5.1.1. Рядок навігації має бути темно-синього кольору, а активний його елемент засвічуватись яскраво-блакитним кольором.

1.5.1.2. При заповненні форми покупцем біля вікна введення імені користувача має бути символ «?», при наведенні на який на сірому фоні видається консультативна інформація, щодо заповнення даного вікна.

1.5.1.3. При введенні коректного паролю має відображатись елемент зеленого кольору, праворуч від вікна введення, а при не коректному введенні – червоного кольору.

1.5.1.4. Поділ товару за категоріями: художня література, дитяча література, технічна література, учбова література.

1.5.1.5. Можливість переміщення вибраних позицій до кошика.

1.5.1.6. Автоматичний підрахунок суми при оформленні замовлення.

1.5.1.7. Реалізація системи пошуку товару.

1.5.1.8. Наявність футеру в нижньому рядку сторінки з інформацією про назву магазину.

1.5.2. Вимоги до структури головної сторінки сайту.

1.5.2.1. Структура головної сторінки згідно макету (рис. 1.4)

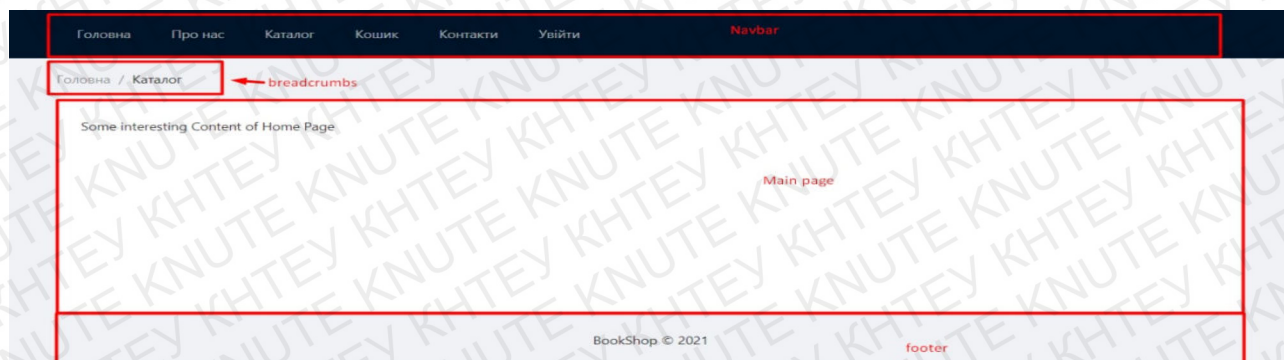


Рис. 1.4. Макет сторінки сайту

1.5.3. Вимоги до представлення товарів на сайті.

1.5.3.1. Кожен товар має містити коротку інформацію у списку товарів.

1.5.3.2. Кожен товар має містити детальний опис на сторінці товару.

1.5.4. Вимоги до мови подання інформації.

1.5.4.1. За замовчуванням при відкритті сторінки сайту має бути встановлено українську мову.

1.6. Висновки до розділу 1

Розглянувши принцип функціонування книжкового магазину, визначивши основні критерії, необхідні для його роботи було окреслено пріоритетні складові, які найбільше впливають на ефективну роботу даного об'єкта. Також враховано специфічність асортименту магазину, а саме – продаж букіністичної літератури.

Було проаналізовано кілька методів дослідження, такі як метод аналізу, метод синтезу та системний метод і враховуючи масштабність проєкту, що розглядається, визначено метод синтезу, як такий, що найбільш ефективно дозволить реалізувати поставлене завдання.

Було проведено порівняльну характеристику аналогів веб-сайтів, що займаються реалізацією книжок та виявлено їх основні недоліки, такі як відсутність, або недосконалість зрозумілого інтерфейсу. Відсутність кошика у одного з магазинів та двомовне відображення інформації на одній сторінці є суттєвим недоліком сайту. Також відмічено не співпадіння назви магазину та імені у веб-адресі, що може заплутати покупця при відвідуванні інтернет-магазину.

Провівши аналіз методів розв'язання поставленої задачі створення веб-сайту було визначено веб-технології, що використовуватимуться при розробці. Обрано мову програмування Python [3], та систему керування базами даних SQLite [1]. Розроблено технічне завдання проєкту.

РОЗДІЛ 2

МОДЕЛЮВАННЯ СТРУКТУРИ ПЗ

2.1. Функціональна характеристика книжкового інтернет-магазину

Постановка та розв'язування задач (зокрема, побудова інформаційних і комп'ютерних моделей) пов'язані із заглибленням у певну предметну область, яка складається з реальних об'єктів (наприклад, автомобілі, люди, країни, товари) та об'єктів абстрактних (наприклад, інтервал часу). Такі об'єкти називають сутностями. Об'єкти кожної предметної області характеризуються сукупністю параметрів (атрибутів). Наприклад, об'єкт Автомобіль може мати такі атрибути, як модель, рік виготовлення, потужність двигуна, тип коробки передач, а об'єкт Художник – прізвище, ім'я, рік народження, біографія, стиль живопису.

Для візуалізації процесу функціонування інформаційної системи застосовуватимемо UML-діаграми. Поведінка акторів і прецедентів системи відображаються їх відносинами, які показані на діаграмі прецедентів ідеального сценарію реалізації товарів (рис. 2.1)

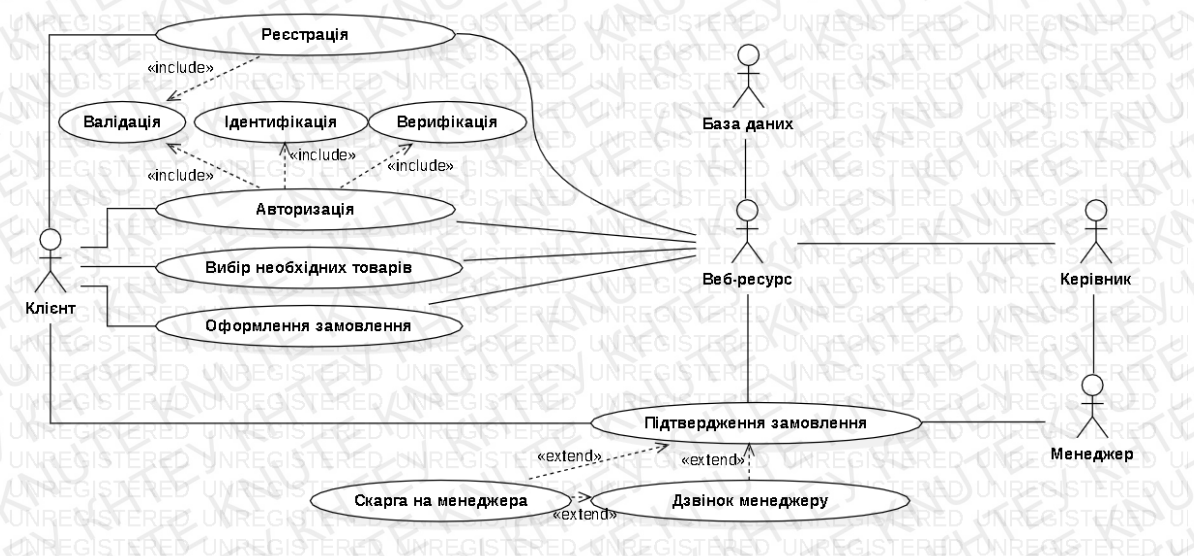


Рис. 2.1. Структура взаємодії клієнта з сайтом магазину

					<i>КНТЕУ 121 06-05.БР</i>		
Зм.	Аркуш	№ докум	Підпис	Дата			
Зав. каф.		Криворучко О.В.		05.03.21	Розробка веб-сайту інтернет-магазину продажу книг	Стадія	Аркуш
Керівник		Рзаєва С.Л.		05.03.21		P2	17
Гарант		Цензура М.О.		05.03.21		Факультет інформаційних технологій, 4 курс, 6 група	
Розроб		Голіков І.В.		05.03.21	Моделювання структури ПЗ		

Діаграма станів на концептуальному рівні дозволяє описувати можливі послідовності станів і переходів окремого прецеденту, які характеризують його динамічну поведінку. Діаграма стану системи зображена на рис. 2.2.

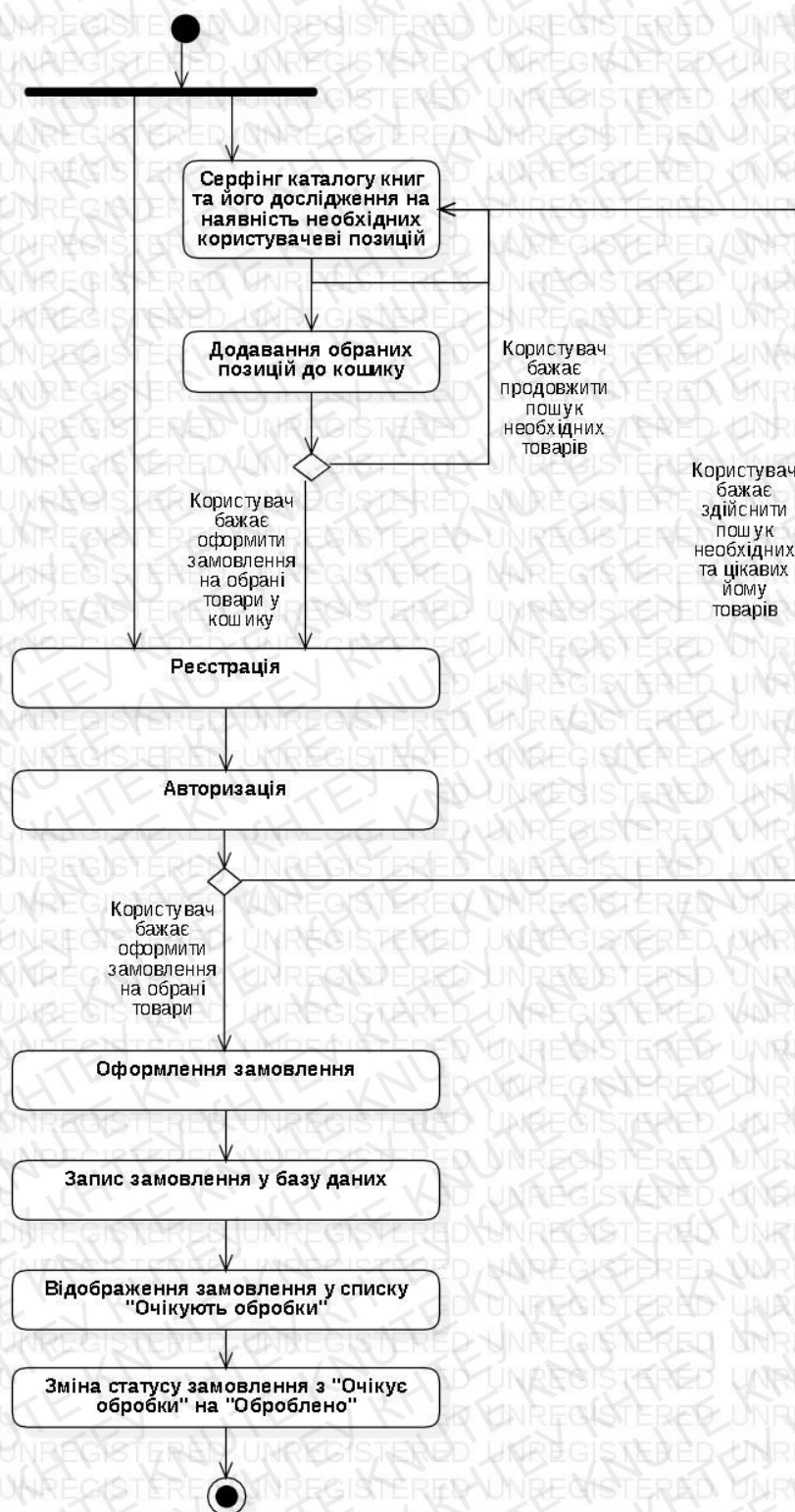


Рис. 2.2 Діаграма станів системи

Для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування служить діаграма класів. На цій діаграмі показано класи, інтерфейси, об'єкти й кооперації, а також їхні відносини в додатку django (рис. 2.3).

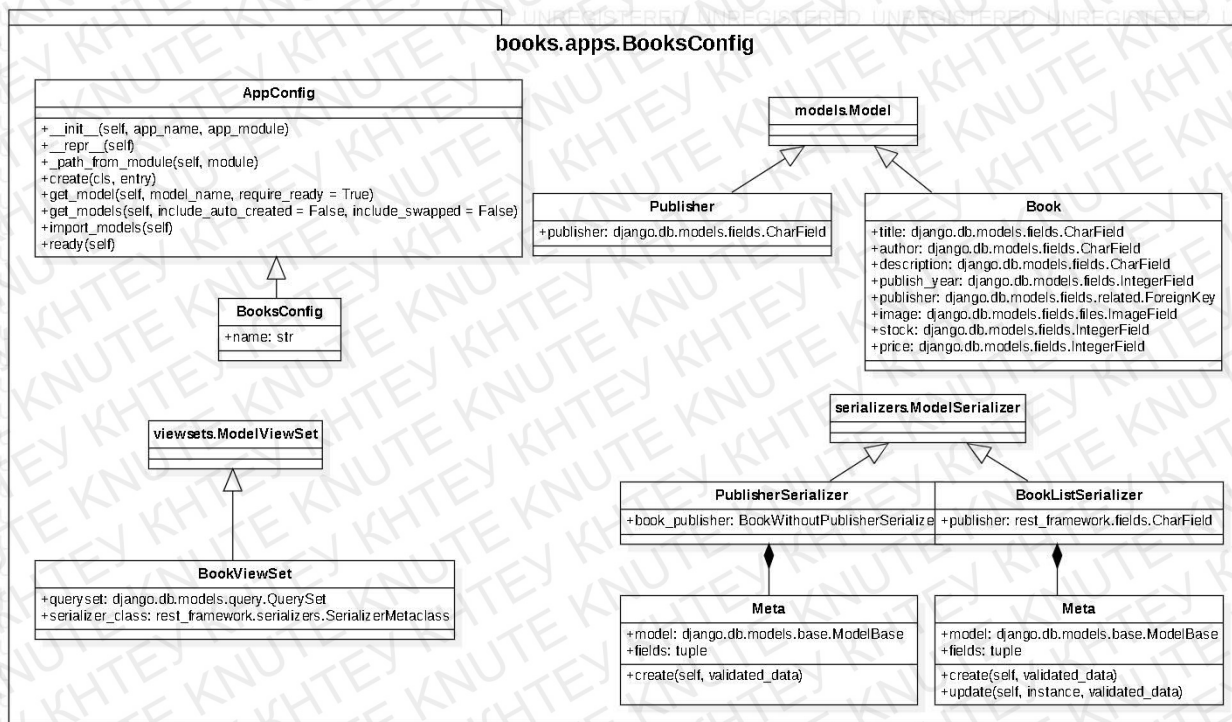


Рис. 2.3 Структура класів додатку django

При моделюванні поведінки проєктованої або аналізованої системи виникає необхідність не тільки уявити процес зміни її станів, але і деталізувати особливості алгоритмічної і логічної реалізації виконуваних системою операцій. Традиційно для цієї мети використовувалися блок-схеми або структурні схеми алгоритмів. Для моделювання процесу виконання операцій в UML використовуються діаграми діяльності. (Рис. 2.4.)

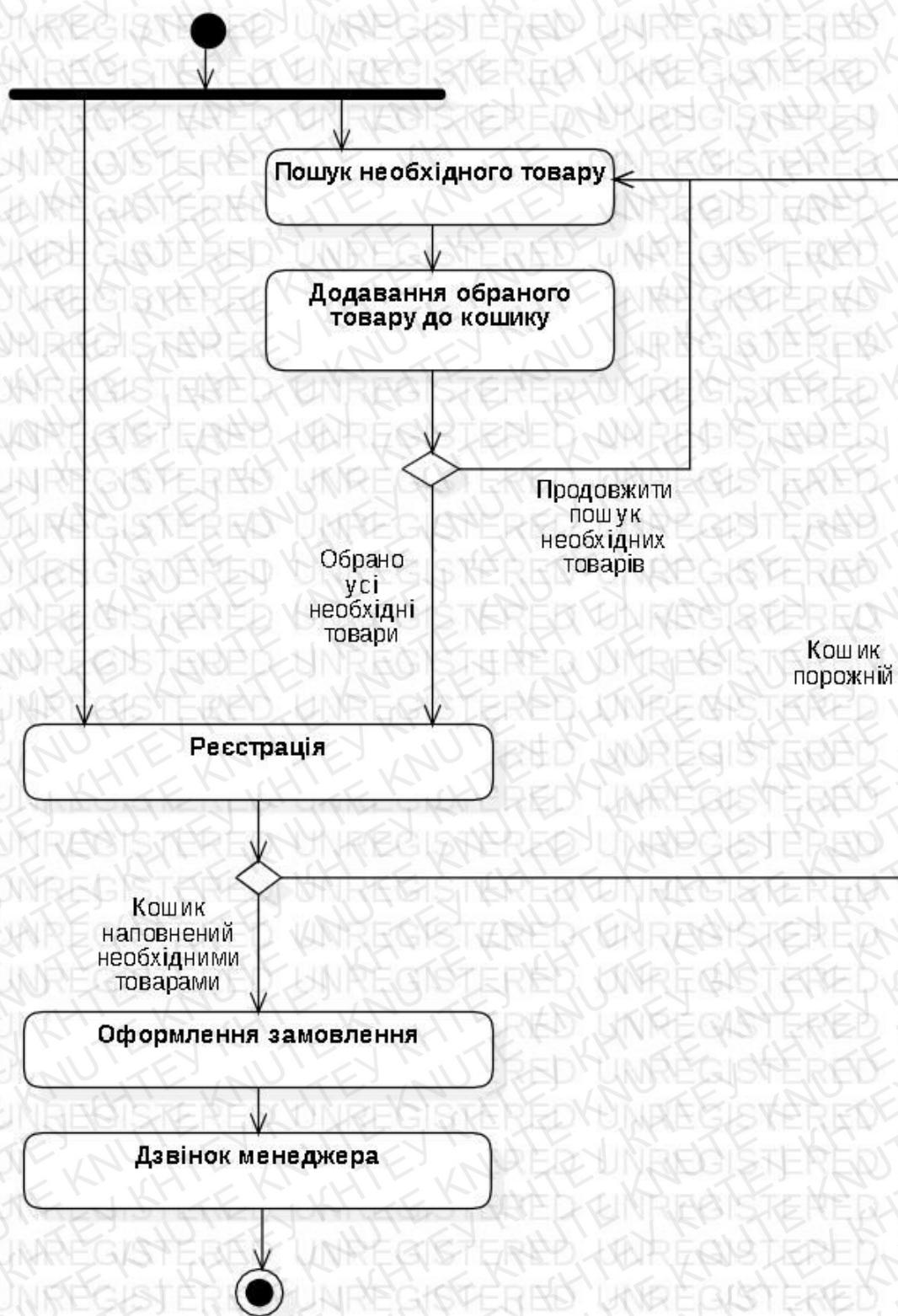


Рис. 2.4. Процес оформлення он-лайн замовлення клієнтом

Проведення аналізу предметної області в інтересах подальшого проектування сайту є завданням, що формує єдиний погляд на відомості, які в предметній області обробляються, враховуючи не тільки їх структури, а й правила зберігання і обробки, що відбиваються у виділених функціях і завданнях.

Діяльність інтернет-магазину слугуватиме предметною областю для розробки веб-додатку.

Для побудови веб-ресурсу необхідно добре розуміти процеси, що відбуваються з даними протягом їх обробки і перетворення. Такі процеси представляються моделями потоків даних.

Формування дерева зв'язків функцій обумовлено можливостями реалізації багатьох процесів роботи з даними в рамках СКБД, і не переносити їх на рівень додатків, відокремлюючи, тим самим, завдання роботи з даними від завдань організації інтерфейсу.

По суті, проєктуємий програмний додаток може бути реалізований на основі трьох умовно-незалежних програмних модулів (рис. 2.5).



Рис.2.5. Дерево зв'язків функцій веб-додатку

При роботі з веб-додатком користувач не взаємодіє з нею безпосередньо, а посилає через інтерфейс програми (так звана front-end складова проєкту) повідомлення про необхідність виконати будь-яку глобальну дію і передати відповідний результат, представлений на екран. Обробка запиту проходить на back-end складовій і за необхідності вносить запис чи редагування до бази даних. Це дозволяє чітко розділити окремі етапи обробки даних і явно вказати модулі програмного забезпечення, що відповідають за відповідні завдання:

- інтерфейс користувача:
 - а) уявлення екранних форм для введення вихідних даних,
 - б) уявлення результату обробки даних в зручному для користувача вигляді,
 - в) візуалізація інтерфейсу користувача;
- інтерфейс програми:
 - а) формування структури і наповнення екранних форм користувача,
 - б) передача відомостей від користувача в обробку на рівень СКБД,
 - в) передача користувачеві результату обробки в СКБД,
 - г) специфічна обробка даних з використанням механізмів, що не реалізуються в СКБД;
- робота з базою даних:
 - а) модифікація даних в базі даних,
 - б) вибірка даних по інформаційним потребам користувачів,
 - в) алгоритмічна обробка даних в базі даних з використанням інструментів програмування.

2.2. Побудова моделі бази даних

Створення бази даних ґрунтується на основі логічної моделі. Логічна модель – це загальний погляд на дані та модель даних в певній предметній області. Інколи її називають концептуальною моделлю даних. Такою моделлю і є модель типу сутність-зв'язок.

Логічні моделі даних засновуються на структурах, визначених у попередній концептуальній моделі даних, оскільки вона описує семантику інформаційного контексту, яку логічна модель повинна також відображати. На рис.2.6 представлена логічна модель бази даних книжкового інтернет-магазину.

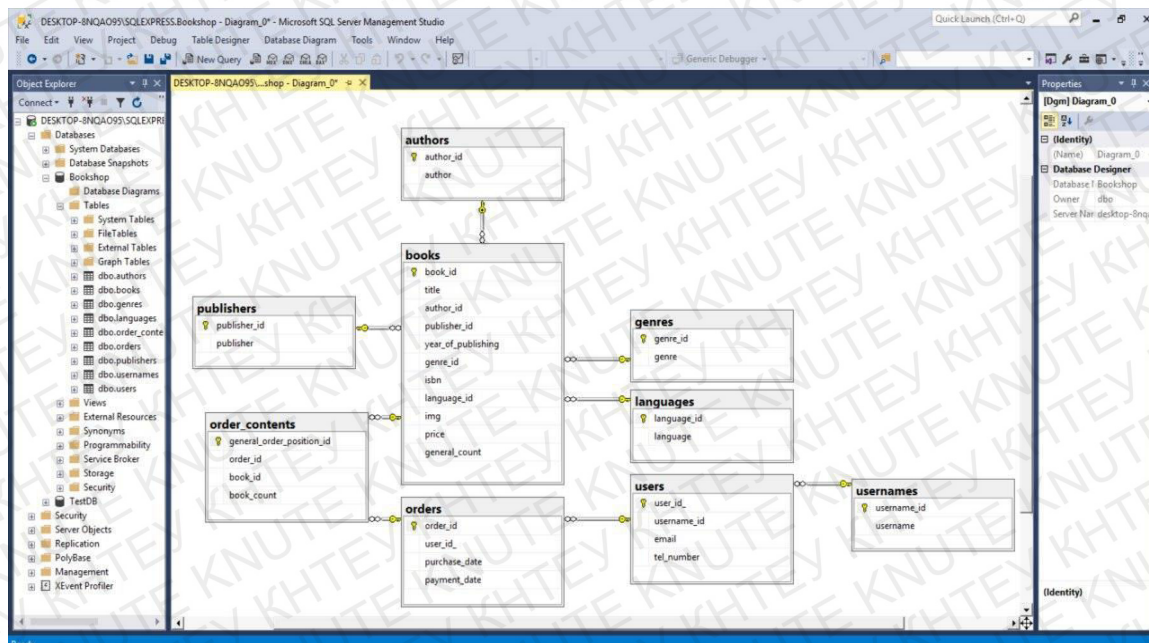


Рис. 2.6. Логічна модель бази даних

Фізична модель даних типово походить від логічної моделі. Завершена фізична модель даних проєкту зображена на рис. 2.7.

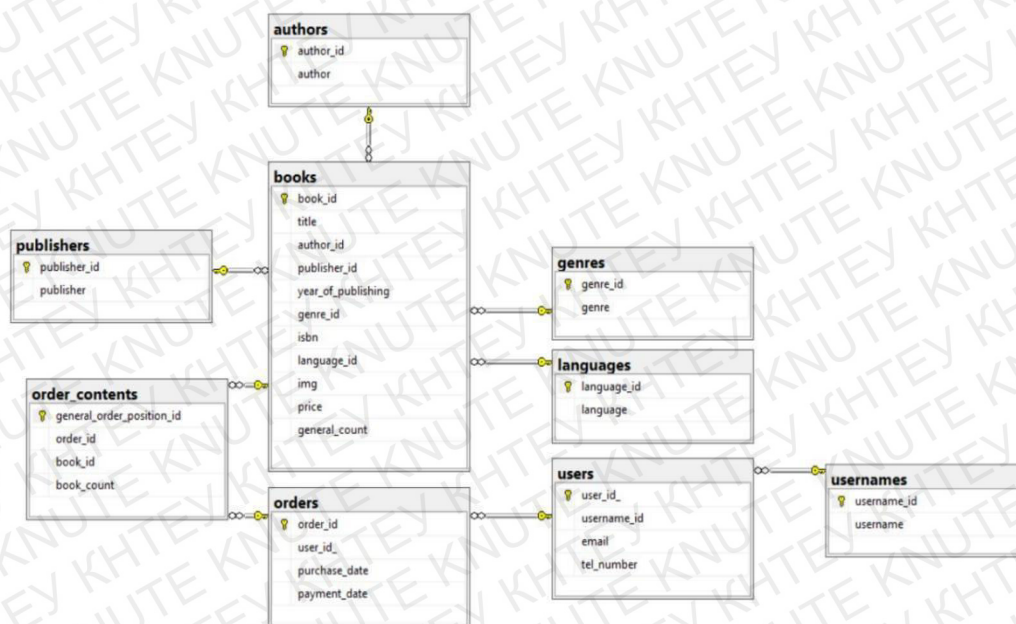


Рис. 2.7. Фізична модель даних

2.3. Створення таблиць

При розгляді основних інформаційних сутностей, які мають бути застосовані для зберігання інформації в системі з'являться додаткові, більш деталізовані елементи, які заносяться до таблиць.

- Таблиця «books» (книги). (Рис. 2.8)

Дана таблиця відображає одну з основних сутностей в даній роботі – дані про книгу. Поле «book_id» є первинним ключем таблиці.

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer						NULL	
2	title	varchar (100)						NULL	
3	author	varchar (250)						NULL	
4	description	varchar (3000)						NULL	
5	isbn	varchar (25)						NULL	
6	publish_year	integer						NULL	
7	number_of_pages	integer						NULL	
8	language	varchar (2)						NULL	
9	cover	varchar (5)						NULL	
10	stock	integer						NULL	
11	price	integer						NULL	
12	image	varchar (100)						NULL	
13	publisher_id	integer						NULL	

Рис. 2.8. Таблиця «books»

- Таблиця «authors» (автори) зберігає дані про авторів. (Рис. 2.9)

authors	
	author_id
	author

Рис. 2.9. Таблиця «authors» (автори)

- Таблиця «publishers» (видавництва) інформує про збережені дані видавництв. (Рис. 2.10)

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer						NULL	
2	publisher	varchar (100)						NULL	

Рис. 2.10. Таблиця «publishers» (видавництва)

- Таблиця «genres» (жанри). В даному полі зберігаються дані про жанри книг магазину. (Рис. 2.11)

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	genre	varchar (250)							NULL

Рис. 2.11. Таблиця «genres» (жанри)

- Таблиця «order_contents» (вміст замовлення). (Рис. 2.12)

order_contents	
	general_order_position_id
	order_id
	book_id
	book_count

Рис. 2.12. Таблиця «order_contents» (вміст замовлення)

- Таблиця «orders» (замовлення). В даному полі містяться дані конкретного замовлення. (Рис. 2.13)

orders	
	order_id
	user_id
	purchase_date
	payment_date

Рис. 2.13. Таблиця «orders» (замовлення)

- Таблиця «users» (користувачі). Зберігає накопичену інформацію про покупців. (Рис. 2.14)

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	password	varchar (128)							NULL
3	last_login	datetime							NULL
4	is_superuser	bool							NULL
5	username	varchar (150)							NULL
6	first_name	varchar (150)							NULL
7	last_name	varchar (150)							NULL
8	email	varchar (254)							NULL
9	is_staff	bool							NULL
10	is_active	bool							NULL
11	date_joined	datetime							NULL
12	gender	varchar (1)							NULL
13	birth_date	date							NULL
14	tel	bigint							NULL
15	location	varchar (150)							NULL
16	status	varchar (50)							NULL
17	modified_at	datetime							NULL

Рис. 2.14. Таблица «users» (користувачі)

2.4. Висновки до розділу 2

Проведено комплекс конструкторських рішень в області розробки програмного забезпечення, моделювання бізнес-процесів, системного проектування та відображення організаційних структур в площині об'єктно-орієнтованого програмування.

При проведенні аналізу предметної області визначено об'єкти, їх атрибути та зв'язки між ними. За цими даними створено дерево зв'язків функцій, побудовано логічну та фізичну моделі бази даних, створено ряд таблиць, необхідних для функціонування системи. Також за допомогою уніфікованої мови моделювання UML (Unified Modeling Language) створено UML-діаграми для візуалізації процесу формування та функціонування проектуємої інформаційної системи, а саме: структура взаємодії клієнта з сайтом магазину, діаграма станів системи, структура класів додатку django, процес оформлення он-лайн замовлення клієнтом.

РОЗДІЛ 3

РОЗРОБКА ВЕБ-САЙТУ ІНТЕРНЕТ-МАГАЗИНУ

3.1. Головна сторінка сайту

Інтернет-магазин являє собою складний сайт з великою кількістю сторінок. Важливо, щоб стартова сторінка ресурсу була правильно структурована. На ній повинні бути такі блоки:

- Каталог товарів.
- Самі товари.
- Кращі пропозиції.
- Форма для замовлення.
- Зворотній зв'язок.

Головна (домашня) сторінка – це сторінка, яка першою завантажується при відкритті сайту.

Вигляд домашньої сторінки проєктуємого сайту зображено на рис. 3.1



Рис. 3.1. Головна сторінка сайту інтернет-магазину

В верхній частині розташована панель навігації, що має темно-синій колір і завдяки цьому візуально виділяється на світло-сірому фоні самої сторінки.

сторінки.

					КНТЕУ 121 06-05.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка веб-сайту інтернет-магазину продажу книг	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		10.04.21		РЗ	27	34
Керівник		Рзаєва С.Л.		10.04.21		Факультет інформаційних технологій, 4 курс, 6 група		
Гарант		Цензура М.О.		10.04.21				
Розроб		Голіков І.В.		10.04.21				
					Розробка веб-сайту інтернет-магазину			

Нижче виводиться назва «Головна сторінка» і додається текстовий елемент, в якому можна викласти коротку інформацію, щодо самого інтернет-магазину. Під текстом розташовується слайдер, який може містити інформацію про анонси, новини, рекламу, тощо. В низу сторінки знаходиться футер із зазначенням прав на даний сайт.

3.2. Сторінка сайту «Про нас»

При переході на дану сторінку користувач отримує інформацію про діяльність інтернет-магазину, історію створення та розвитку, планування заходів на майбутнє. Вигляд сторінки «Про нас» зображено на рис. 3.2

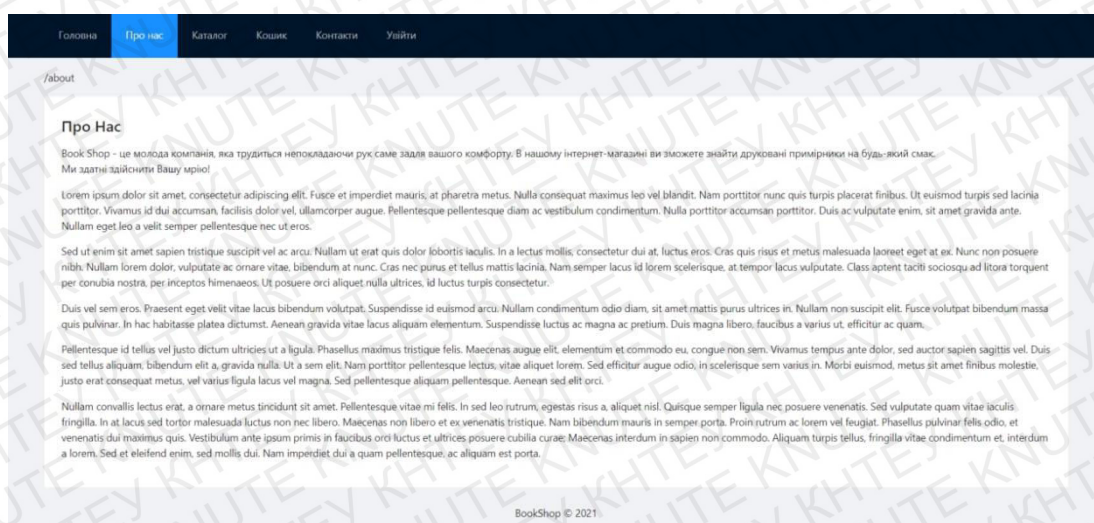


Рис. 3.2. Сторінка «Про нас» сайту інтернет-магазину

3.3. Сторінка каталогу товарів

Сторінка каталогу товарів показана на рис. 3.3

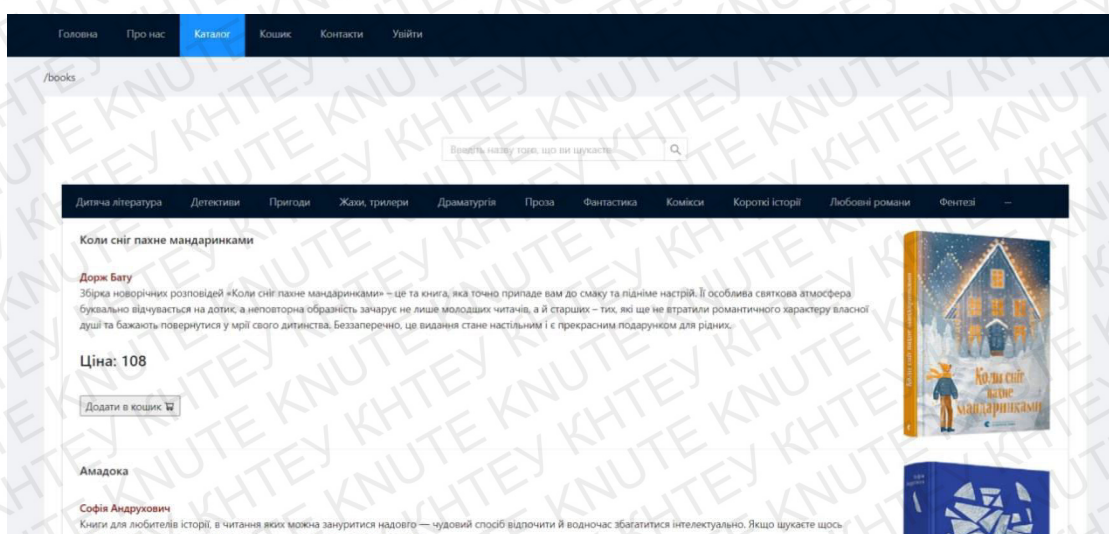


Рис. 3.3. Сторінка каталогу товарів інтернет-магазину

На даній сторінці під панеллю навігації розташовано вікно пошуку товарів.

Нижче на сторінці знаходиться фільтр товарів, що зовні та функціонально нагадує панель навігації. Перейшовши по вкладці фільтру користувач потрапляє до тематичного каталогу товарів.

Під фільтром відображається список відповідних товарних позицій у форматі: Назва примірника, автор, опис книги, ціnnик, вікно додавання позиції до кошика. В правій частині навпроти поданої інформації знаходиться мініатюра обкладинки книги.

3.4. Сторінка кошику замовлень

Кошик замовлень – це елемент сайту, в якому формується покупка. Вигляд сторінки кошику замовлень показано на рис. 3.4

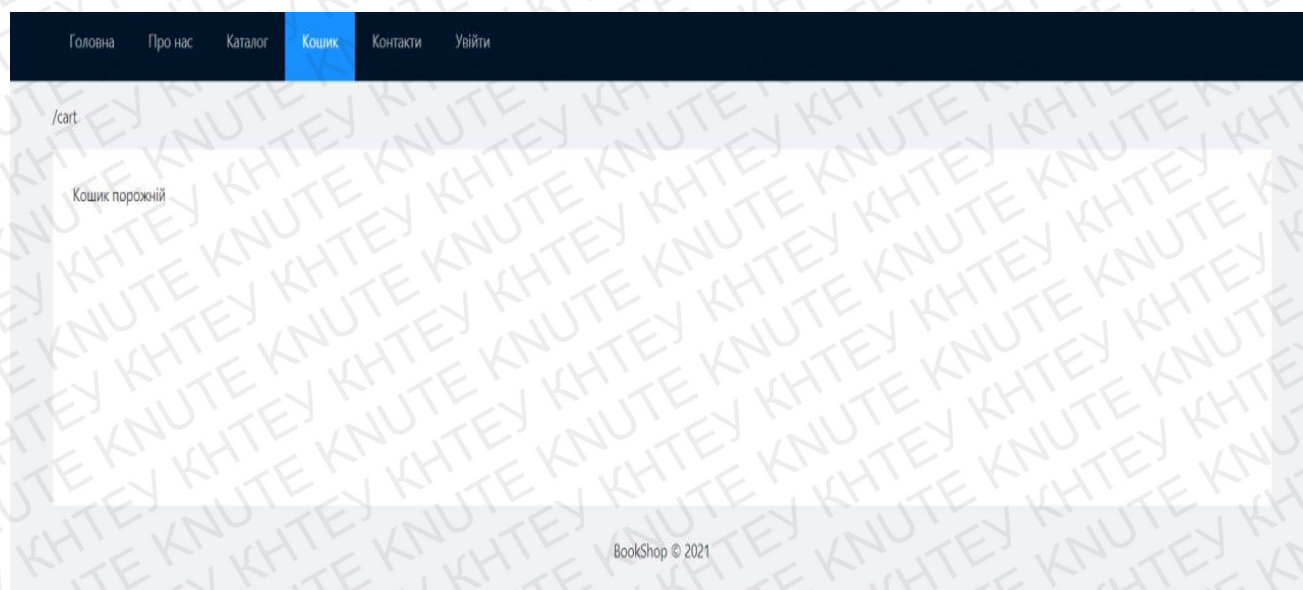


Рис. 3.4. Сторінка кошику замовлень інтернет-магазину.

Допоки до кошика не додано жодної товарної позиції, на сторінці відображатиметься повідомлення, що кошик порожній.

3.5. Сторінка зворотного зв'язку

Вигляд сторінки зворотного зв'язку зображено на рис.3.5

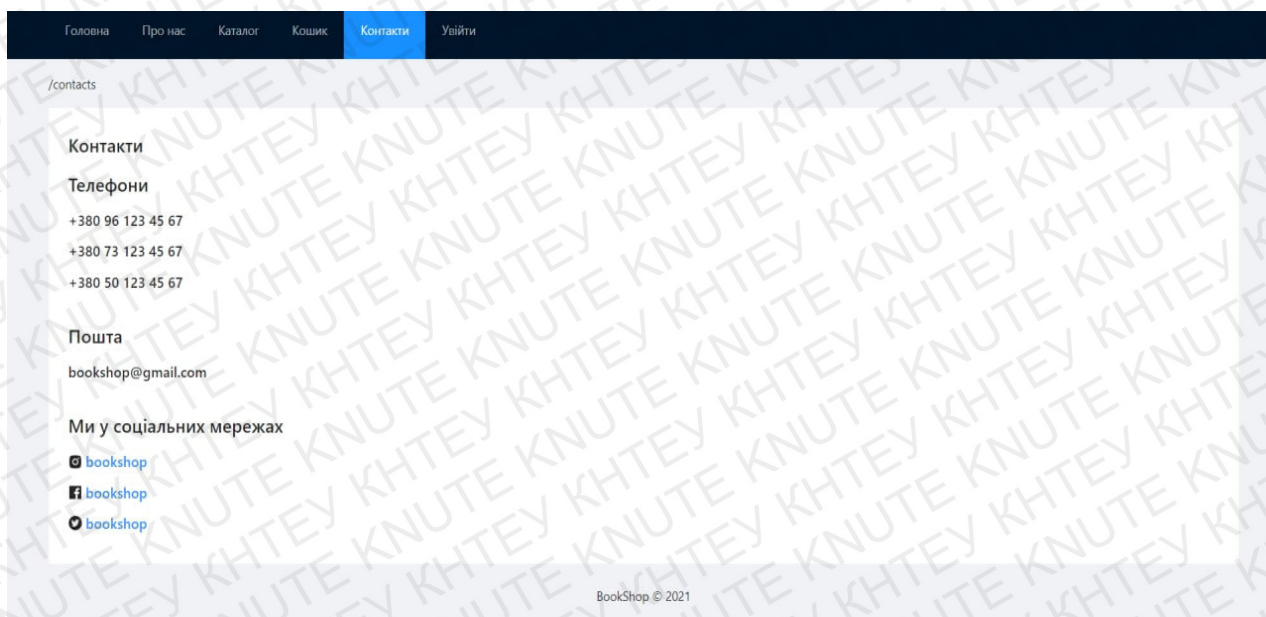


Рис. 3.5. Сторінка зворотного зв'язку інтернет-магазину

На даній сторінці міститься інформація контактів для зв'язку з представниками інтернет-магазину. Вказано номери телефонів, адресу електронної пошти та посилання на сторінки магазину в соцмережах.

3.6. Сторінка авторизації

Даний елемент сайту призначений для авторизації користувача, існуючого в базі інтернет-магазину, або ж для реєстрації нового відвідувача.

Вигляд сторінки авторизації зображено на рис.3.6

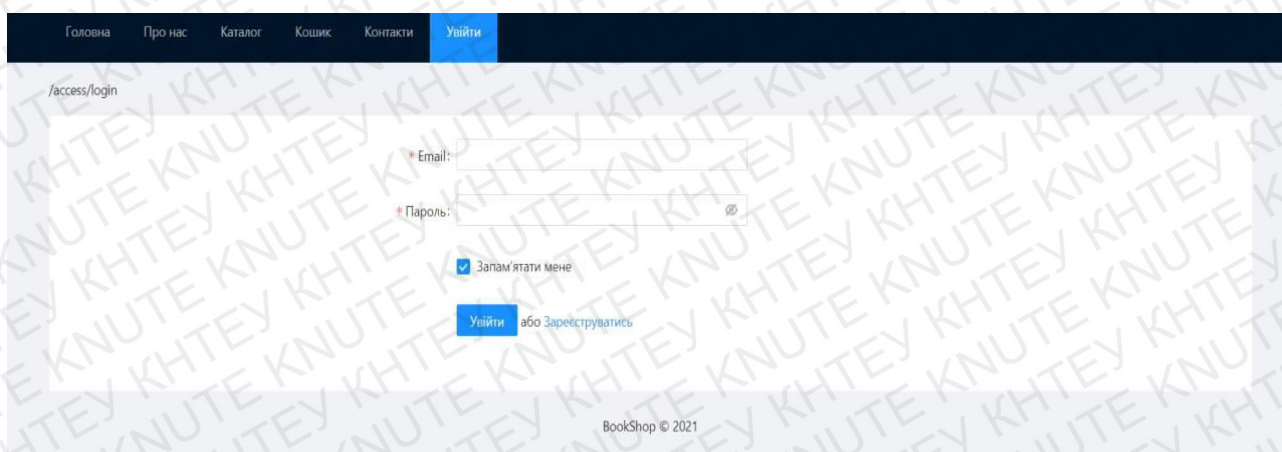


Рис. 3.6. Сторінка авторизації користувача інтернет-магазину

Якщо відвідувач не зареєстрований, потрібно натиснути кнопку «Зареєструватись». Відкриється вікно реєстрації нового користувача, (рис.3.7)

Рис. 3.7. Сторінка реєстрації нового користувача

3.7. Висновки до розділу 3

Розробка веб-сайту книжкового інтернет-магазину зводилась до створення програмного продукту, який забезпечуватиме виконання умов технічного завдання, матиме простий та зрозумілий інтерфейс, дозволить виконання основної функції, а саме забезпечуватиме процес продажу книжкової продукції.

Всі сторінки сайту виконані в одному стилі без зайвих елементів, які можуть відвертати увагу користувача від основної задачі сайту – пошук необхідної літератури та простота здійснення покупки. Особливу увагу було зосереджено при проектуванні головної сторінки. Зокрема виконані всі умови технічного завдання, щодо оформлення цієї сторінки та забезпечено інтуїтивно просту для користувача навігацію сторінками сайту.

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

В ході розробки проєкту була визначена нормативно-правова база, що регламентує діяльність інтернет-магазину у правовому полі держави. В Україні вона ґрунтується на переліку законів та підзаконних актів.

Виходячи із стрімкого переходу комерційної діяльності в інтернет та маючи розроблене технічне завдання, було проведено комплекс заходів по проєктуванню інформаційної системи, а саме веб-сайту інтернет-магазину з продажу книг. Проаналізувавши сайти вже працюючих інтернет-крамниць було виявлено ряд недоліків, які було враховано при розробці веб-сайту книжкового магазину. Побудовано модель бази даних. На основі моделі складено таблиці сутностей. Проведено процес розробки веб-сайту. Створено головну сторінку та сторінки інших розділів сайту.

Під час виконання випускної кваліфікаційної роботи було створено сайт інтернет-магазину з продажу книг. Розроблено дизайн веб-ресурсу та впроваджено необхідний функціонал.

Для візуалізації процесу функціонування інформаційної системи було застосовано UML-діаграми.

Оскільки результатом даної роботи є повністю функціонуючий веб-сайт інтернет магазину, можна пропонувати розширити каталог товарів іншими категоріями, наприклад канцелярські товари та офісне приладдя. Дана пропозиція буде корисна, в першу чергу, учням та студентам, які постійно купують ці види товарів разом з учбовою літературою.

Отже, в ході виконання випускної кваліфікаційної роботи було продемонстровано практичні навички проєктування та створення веб-сайту, набуті за час навчання.

					<i>КНТЕУ 121 06-05.БР</i>			
					<i>Розробка веб-сайту інтернет-магазину продажу книг</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>		<i>ВП</i>	<i>32</i>	<i>34</i>
<i>Зав. каф.</i>	<i>Криворучко О.В.</i>			<i>24.04.21</i>		<i>Факультет інформаційних технологій, 4 курс, 6 група</i>		
<i>Керівник</i>	<i>Рзаєва С.Л.</i>			<i>24.04.21</i>				
<i>Гарант</i>	<i>Цензура М.О.</i>			<i>24.04.21</i>				
<i>Розроб</i>	<i>Голіков І.В.</i>			<i>24.04.21</i>				
					<i>ВИСНОВКИ ТА ПРОПОЗИЦІЇ</i>			

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Балик Н. Р., Мандзюк В. І. Бази даних MySQL: Навчальний посібник. – Тернопіль: Навчальна книга – Богдан, 2010. – 160 с.
2. Берко А. Ю., Верес О. М. Організація баз даних: практичний курс: Навчальний посібник / За наукової редакції В.В. Пасічника. – Львів: Видавництво Національного університету «Львівська політехніка», 2003. – 152с.
3. Васильєв О.М. Програмування мовою Python / О. М. Васильєв. – Тернопіль: Навчальна книга – Богдан, 2019. – 504с.
4. Новожилова М. В., Резнікова С. Ю. Інформаційна безпека систем управління базами даних: Навчальний посібник. – Харків: ХДТУБА, 2004. – 185 с.
5. Пасічник В. В., Резніченко В. А. Організація баз даних та знань. – К.: Видавнича група ВНУ, 2006. – 384с.
6. Романюк О. Н., Савчук Т. О. Організація баз даних і знань. Навчальний посібник. – Вінниця: УНІВЕРСУМ-Вінниця, 2003. – 217 с.
7. Тарасов О. В. Використання мови SQL для роботи з сучасними системами керування базами даних. Практикум з навчальної дисципліни "Організація баз даних та знань" : навчально-практичний посібник / О. В. Тарасов, М. Ю. Лосєв, В. В. Федько. – Харків: ХНЕУ, 2013. – 347 с.
8. Федько В. В. Організація баз даних та знань: навчально-практичний посібник для самостійної підготовки студентів / В. В. Федько, О. В. Тарасов, М. Ю. Лосєв. – Харків: ХНЕУ, 2013. – 198с.
9. Alan Beaulieu. Learning SQL. – United States of America: O'Reilly Media, Inc., 2020. – 554p.

					<i>КНТЕУ 121 06-05.БР</i>			
Зм.	Аркуш	№ докум	Підпис	Дата				
Зав. каф.		Криворучко О.В.		24.04.21	<i>Розробка веб-сайту інтернет-магазину продажу книг</i>	Стадія	Аркуш	Аркуші в
Керівник		Рзаєва С.Л.		24.04.21		СД	33	34
Гарант		Цензура М.О.		24.04.21		Факультет інформаційних технологій, 4 курс, 6 група		
Розроб		Голіков І.В.		24.04.21				
					<i>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</i>			

10. Allen G. Taylor. SQL All-In-One. – New Jersey: John Wiley & Sons, Inc., 2019. – 739p.
11. Anthony De Barros. Practical SQL: A Beginner's Guide to Storytelling with Data. – San Francisco: No Starch Press, Inc., 2018. – 483p.
12. Bradley Beard. Beginning Backup and Restore for SQL Server. – Palm Bay, Florida, USA: Apress Media LLC, 2018. – 316p.
13. Dmitri Korotkevitch. Pro SQL Server Internals, Second Edition. – Florida, USA: Apress Media LLC, 2016. – 805p.
14. Enrico van de Laar. Pro SQL Server 2019 Wait Statistics: A Practical Guide to Analyzing Performance in SQL Server. – Drachten, The Netherlands: Apress Media LLC, 2019. – 407p.
15. Eric A. Meyer. CSS Floating. – Sebastopol: Published by O'Reilly Media, Inc., 2016. – 60p.

ДОДАТКИ

Додаток А

Лістинг моделей Django-додатку Books

```
from django.db import models

def image_upload_path(instance, filename):
    return '/'.join(['covers', str(instance.title), filename])

class Publisher(models.Model):

    publisher = models.CharField(max_length=100, default="undefined")

class Genre(models.Model):

    genre = models.CharField(max_length=250)

    def __str__(self):
        return self.genre

class Book(models.Model):

    LANGUAGES = (
        ('UA', 'Ukrainian'),
        ('EN', 'English'),
        ('RU', 'Russian'),
    )
```

```

COVERS = (
    ('SC', 'Softcover'),
    ('HCIW', 'Hardcover with ImageWrap'),
    ('HCDJ', 'Hardcover with Dust Jacket'),
)

title = models.CharField(max_length=100)
author = models.CharField(max_length=250)
description = models.CharField(max_length=3000)
isbn = models.CharField(max_length=25, default="Undefined") # #####
genre = models.ManyToManyField(Genre) # #####
publish_year = models.IntegerField()
number_of_pages = models.IntegerField(default=0) # #####
publisher = models.ForeignKey(Publisher, on_delete=models.CASCADE,
related_name="book_publisher")

language = models.CharField(max_length=2, choices=LANGUAGES) # #####
cover = models.CharField(max_length=5, choices=COVERS) # #####
stock = models.IntegerField()
price = models.IntegerField()
image = models.ImageField(max_length=100, upload_to=image_upload_path,
default="pic_absent.jpg")

```

Лістинг моделей Django-додатку Access

```
from django.db import models

from django.contrib.auth.models import AbstractUser # , User

from books.models import Book


# Create your models here.
class User(AbstractUser):
    GENDERS = (
        ('M', 'Male'),
        ('F', 'Female'),
        ('O', 'Other'),
    )

    gender = models.CharField(max_length=1, choices=GENDERS, null=True)
    birth_date = models.DateField(null=True)
    tel = models.BigIntegerField(null=True)
    location = models.CharField(max_length=150, null=True)
    status = models.CharField(max_length=50, default="customer")
    modified_at = models.DateTimeField(auto_now=True)


class Order(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE) # User
```

```
order_date = models.DateTimeField(auto_now_add=True)
```

```
class Order_content(models.Model):
```

```
    order = models.ForeignKey(Order, on_delete=models.CASCADE)
```

```
    book = models.ForeignKey(Book, on_delete=models.CASCADE)
```

```
    amount = models.IntegerField()
```

Лістинг серіалізаторів Django-додатку Books

```
from rest_framework import serializers
from books.models import Book, Publisher, Genre
from rest_framework.relations import PrimaryKeyRelatedField

class BookWithoutPublisherSerializer(serializers.ModelSerializer):

    class Meta:
        model = Book
        exclude = ('publisher',)

class PublisherSerializer(serializers.ModelSerializer):
    book_publisher = BookWithoutPublisherSerializer(many=True)

    class Meta:
        model = Publisher
        # fields = "__all__"
        fields = ('id', 'publisher')

    def create(self, validated_data):
        books_data = validated_data.pop('book_publisher')
        publisher = Publisher.objects.create(**validated_data)
```

for book in books_data:

 Book.objects.create(publisher=publisher, **book)

return publisher

class GenreSerializer(serializers.ModelSerializer):

 class Meta:

 model = Genre

 fields = '__all__'

class BookListSerializer(serializers.ModelSerializer):

 publisher = serializers.CharField(source="publisher.publisher")

 # genre = GenreSerializer(read_only=True, many=True)

 genre = PrimaryKeyRelatedField(queryset=Genre.objects.all(), many=True)

 # genre = serializers.SlugRelatedField(queryset=Genre.objects.all(),

 slug_field='genre')

 class Meta:

 model = Book

 fields = ('id',

 'title',

 'author',

 'description',

 'isbn',

 'genre',

```
'publish_year',  
'number_of_pages',  
'publisher',  
'language',  
'cover',  
'stock',  
'price',  
'image'  
)
```

```
def create(self, validated_data):
```

```
    chosen_publisher =
```

```
    Publisher.objects.get(publisher=validated_data['publisher']['publisher'])
```

```
    validated_data['publisher'] = chosen_publisher
```

```
    chosen_genres = validated_data.pop('genre')
```

```
    book = Book.objects.create(**validated_data)
```

```
    book.genre.add(*chosen_genres)
```

```
    return book
```

```
def update(self, instance, validated_data):
```

```
    chosen_publisher =
```

```
    Publisher.objects.get(publisher=validated_data['publisher']['publisher'])
```

```
    validated_data['publisher'] = chosen_publisher
```

```
    instance.title = validated_data.get('title', instance.title)
```

```
    instance.author = validated_data.get('author', instance.author)
```

```
instance.description = validated_data.get('description', instance.description)
```

```
instance.publish_year = validated_data.get('publish_year', instance.publish_year)
```

```
instance.stock = validated_data.get('stock', instance.stock)
```

```
instance.price = validated_data.get('price', instance.price)
```

```
instance.publisher = validated_data.get('publisher', instance.publisher)
```

```
instance.image = validated_data.get('image', instance.image)
```

```
instance.save()
```

```
return instance
```

Лістинг серіалізаторів Django-додатку Access

```
from rest_framework import serializers
from access.models import User, Order, Order_content
from dj_rest_auth.models import TokenModel
from django.utils.translation import ugettext_lazy as _
from allauth.account import app_settings as allauth_settings
from allauth.account.adapter import get_adapter
from allauth.account.utils import setup_user_email
from allauth.utils import email_address_exists, get_username_max_length

class RegisterSerializer(serializers.Serializer):
    username = serializers.CharField(
        max_length=get_username_max_length(),
        min_length=allauth_settings.USERNAME_MIN_LENGTH,
        required=allauth_settings.USERNAME_REQUIRED
    )
    email = serializers.EmailField(required=allauth_settings.EMAIL_REQUIRED)
    password1 = serializers.CharField(write_only=True)
    password2 = serializers.CharField(write_only=True)

    # Additional info
    gender = serializers.CharField()
    first_name = serializers.CharField()
    last_name = serializers.CharField(required=False)
```

```
birth_date = serializers.DateField(required=False)

location = serializers.CharField()

tel = serializers.IntegerField()


def validate_username(self, username):
    username = get_adapter().clean_username(username)
    return username


def validate_email(self, email):
    email = get_adapter().clean_email(email)
    if allauth_settings.UNIQUE_EMAIL:
        if email and email_address_exists(email):
            raise serializers.ValidationError(
                _("A user is already registered with this e-mail address."))
    return email


def validate_password1(self, password):
    return get_adapter().clean_password(password)


def validate(self, data):
    if data['password1'] != data['password2']:
        raise serializers.ValidationError(_("The two password fields do not match."))
    return data


def custom_signup(self, request, user):
    pass


def get_cleaned_data(self):
```

```

return {
    'username': self.validated_data.get('username', ''),
    'password1': self.validated_data.get('password1', ''),
    'email': self.validated_data.get('email', ''),

    'gender': self.validated_data.get('gender', ''),
    'first_name': self.validated_data.get('first_name', ''),
    'last_name': self.validated_data.get('last_name', ''),
    'birth_date': self.validated_data.get('birth_date', ''),
    'location': self.validated_data.get('location', ''),
    'tel': self.validated_data.get('tel', ''),
}

```

```

def save(self, request):
    adapter = get_adapter()
    user = adapter.new_user(request)
    self.cleaned_data = self.get_cleaned_data()
    adapter.save_user(request, user, self)
    self.custom_signup(request, user)
    setup_user_email(request, user, [])

    # user.gender = self.cleaned_data.get('gender')
    # user.birth_date = self.cleaned_data.get('birth_date')

    return user

```

```

class UserProfileSerializer(serializers.ModelSerializer):
    class Meta:

```

```
model = User
```

```
fields = "__all__"
```

```
class OrderContentSerializer(serializers.ModelSerializer):
```

```
    """
```

```
    For order creation.
```

```
    """
```

```
class Meta:
```

```
    model = Order_content
```

```
    fields = (
```

```
        'book',
```

```
        'amount',
```

```
    )
```

```
class OrderProfileSerializer(serializers.ModelSerializer):
```

```
    """
```

```
    Orders representation for Profile.
```

```
    """
```

```
    user = serializers.CharField(source="user.username")
```

```
class Meta:
```

```
    model = Order
```

```
    fields = ('id',
```

```
              'user',
```

```
              'order_date',
```

```
    )
```

```
class OrderSerializer(serializers.ModelSerializer):
```

```
"""
```

```
    Order serializer for orders creation.
```

```
"""
```

```
books_data = OrderContentSerializer(many=True)
```

```
class Meta:
```

```
    model = Order
```

```
    fields = (
```

```
        'books_data',
```

```
    )
```

```
def create(self, validated_data):
```

```
    books_data = validated_data['books_data']
```

```
    user_id = self.context['request'].user.id
```

```
    chosen_user = User.objects.get(id=user_id)
```

```
    order = Order.objects.create(user=chosen_user)
```

```
    for book in books_data:
```

```
        Order_content.objects.create(order=order, **book)
```

```
    return validated_data
```

```
class CheckoutSerializer(serializers.ModelSerializer):
```

```
    class Meta:
```

```
        model = User
```

```
        fields = (
```

```
            'first_name',
```

```
            'tel',
```

```
            'location',
```

```
        )
```

```
class TokenSerializer(serializers.ModelSerializer):
```

```
    """
```

```
    Serializer for Token model.
```

```
    """
```

```
    is_user_staff = serializers.CharField(source="user.is_staff")
```

```
    class Meta:
```

```
        model = TokenModel
```

```
        fields = ('key', 'is_user_staff')
```

Лістинг Django файлу з налаштуваннями проєкту

"""

Django settings for BookShopDjango project.

Generated by 'django-admin startproject' using Django 3.1.2.

For more information on this file, see

<https://docs.djangoproject.com/en/3.1/topics/settings/>

For the full list of settings and their values, see

<https://docs.djangoproject.com/en/3.1/ref/settings/>

"""

import os

from pathlib import Path

Build paths inside the project like this: BASE_DIR / 'subdir'.

BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

Quick-start development settings - unsuitable for production

See <https://docs.djangoproject.com/en/3.1/howto/deployment/checklist/>

SECURITY WARNING: keep the secret key used in production secret!

SECRET_KEY = '25+&y1hu@^wn3\$wmh+_n228kt_\$atn*240=vjz#u^z)6%2o%\$'

SECURITY WARNING: don't run with debug turned on in production!

DEBUG = True

```
ALLOWED_HOSTS = []
```

```
# Application definition
```

```
INSTALLED_APPS = [
```

```
    'django.contrib.admin',
```

```
    'django.contrib.auth',
```

```
    'django.contrib.contenttypes',
```

```
    'django.contrib.sessions',
```

```
    'django.contrib.messages',
```

```
    'django.contrib.staticfiles',
```

```
    'django.contrib.sites',
```

```
    'allauth',
```

```
    'allauth.account',
```

```
    'allauth.socialaccount',
```

```
    'rest_framework',
```

```
    'rest_framework.authtoken',
```

```
    # 'rest_auth',
```

```
    # 'rest_auth.registration',
```

```
    'dj_rest_auth',
```

```
    'dj_rest_auth.registration',
```

```
    'common_tabs.apps.CommonTabsConfig',
```

```
    'books.apps.BooksConfig',
```

```
    'access.apps.AccessConfig',
```

```

'corsheaders',
]

SITE_ID = 2

AUTH_USER_MODEL = 'access.User'

REST_FRAMEWORK = {
    'DEFAULT_AUTHENTICATION_CLASSES': (
        # 'rest_framework.permissions.AllowAny',
        # 'rest_framework.authentication.SessionAuthentication',
        'rest_framework.authentication.TokenAuthentication',
    )
}

# dj-rest-auth
# default: REST_AUTH_REGISTER_SERIALIZERS =
'rest_auth.registration.serializers.RegisterSerializer'
REST_AUTH_REGISTER_SERIALIZERS = {
    "REGISTER_SERIALIZER": "access.serializers.RegisterSerializer"
}

REST_AUTH_SERIALIZERS = {
    'TOKEN_SERIALIZER': 'access.serializers.TokenSerializer',
}

# django-allauth
ACCOUNT_AUTHENTICATION_METHOD = 'email'
ACCOUNT_EMAIL_REQUIRED = True

```

```
ACCOUNT_USERNAME_REQUIRED = False
ACCOUNT_EMAIL_VERIFICATION = 'none'

ACCOUNT_ADAPTER = 'access.adapters.CustomUserAccountAdapter'

AUTHENTICATION_BACKENDS = (
    "django.contrib.auth.backends.ModelBackend",
    "allauth.account.auth_backends.AuthenticationBackend"
)

MIDDLEWARE = [
    # CORS
    'corsheaders.middleware.CorsMiddleware',
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',

    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'BookShopDjango.urls'

TEMPLATES_DIR = os.path.join(BASE_DIR, 'templates')

TEMPLATES = [
    {
```

```
'BACKEND': 'django.template.backends.django.DjangoTemplates',
'DIRS': [TEMPLATES_DIR, ],
'APP_DIRS': True,
'OPTIONS': {
    'context_processors': [
        'django.template.context_processors.media',
        'django.template.context_processors.debug',
        'django.template.context_processors.request',
        'django.contrib.auth.context_processors.auth',
        'django.contrib.messages.context_processors.messages',
    ],
},
},
]
```

```
WSGI_APPLICATION = 'BookShopDjango.wsgi.application'
```

```
# Database
```

```
# https://docs.djangoproject.com/en/3.1/ref/settings/#databases
```

```
DATABASES = {
```

```
    # 'default': {
```

```
        # 'ENGINE': 'django.db.backends.postgresql',
```

```
        # 'NAME': 'bookshop',
```

```
        # 'USER': 'laurence',
```

```
        # 'PASSWORD': 'qwerty',
```

```
        # 'HOST': '127.0.0.1',
```

```
        # 'PORT': '5432'
```

```
# }

'default': {
    'ENGINE': 'django.db.backends.sqlite3',
    'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
}

# Password validation
# https://docs.djangoproject.com/en/3.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [

    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
# https://docs.djangoproject.com/en/3.1/topics/i18n/
```

```
LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_L10N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)
```

```
# https://docs.djangoproject.com/en/3.1/howto/static-files/
```

```
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
```

```
STATIC_URL = '/static/'
```

```
# Extra places for collectstatic to find static files.
```

```
STATICFILES_DIRS = [
```

```
    os.path.join(BASE_DIR, 'static')
```

```
]
```

```
MEDIA_URL = '/media/'
```

```
MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
```

```
CORS_ORIGIN_ALLOW_ALL = False
```

```
CORS_ORIGIN_WHITELIST = (
```

```
    'http://localhost:3000',
```

```
    'http://localhost:8000',
```

```
    'http://localhost:8081',
```

)

```
CSRF_COOKIE_NAME = "csrftoken"
```

Лістинг Django файлу URL конфігурації

```
"""BookShopDjango URL Configuration
```

The `urlpatterns` list routes URLs to views. For more information please see:

<https://docs.djangoproject.com/en/3.1/topics/http/urls/>

Examples:

Function views

1. Add an import: from my_app import views
2. Add a URL to urlpatterns: path("", views.home, name='home')

Class-based views

1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path("", Home.as_view(), name='home')

Including another URLconf

1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))

```
"""
```

```
from django.contrib import admin
```

```
from django.conf import settings
```

```
from django.conf.urls.static import static
```

```
from django.urls import path, include
```

```
urlpatterns = [
```

```
    path('admin/', admin.site.urls),
```

```
    path("", include('common_tabs.urls')),
```

```
    path('books/', include('books.urls')),
```

```
    path('access/', include('access.urls')),
```

```
path('dj-rest-auth/', include('dj_rest_auth.urls')),  
path('dj-rest-auth/registration/', include('dj_rest_auth.registration.urls')),  
]  
  
if settings.DEBUG:  
    urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Лістинг тестів Django-додатку Books

```
from django.test import TestCase
from .models import Book, Publisher

# Create your tests here.
class PublisherModelTest(TestCase):

    @classmethod
    def setUpTestData(cls):
        Publisher.objects.create(publisher='Some Test publisher')

    def test_publisher_content(self):
        publisher = Publisher.objects.get(id=1)
        expected_object_name = f'{publisher.publisher}'
        self.assertEqual(expected_object_name, 'Some Test publisher')

class BookModelTest(TestCase):

    @classmethod
    def setUpTestData(cls):
        publisher = Publisher.objects.create(publisher='Some Test publisher')
        Book.objects.create(title='Some Test Book',
```

```
        author="",
        description="",
        publish_year=2021,
        stock=3,
        price=120,
        publisher=publisher)

def test_title_content(self):
    book = Book.objects.get(id=1)
    expected_object_name = f'{book.publisher.publisher}'
    print(expected_object_name)
    self.assertEqual(expected_object_name, 'Some Test publisher')
```

Лістинг тестів Django-додатку Access

```
from django.test import TestCase
from .models import User, Order, Order_content
from books.models import Book, Publisher
```

```
# Create your tests here.
```

```
class UserModelTest(TestCase):
```

```
    @classmethod
```

```
    def setUpTestData(cls):
```

```
        User.objects.create(
```

```
            username='Fant1987',
```

```
            first_name='Fant',
```

```
            last_name='Late',
```

```
            email='fant87@ukr.net',
```

```
            gender='M',
```

```
            birth_date='1990-07-18',
```

```
            tel=503520232,
```

```
            location='Lisova street, Kyiv city',
```

```
        )
```

```
    def test_user_content(self):
```

```
        user = User.objects.get(id=1)
```

```
        expected_object_location = f'{user.location}'
```

```
self.assertEqual(expected_object_location, 'Lisova street, Kyiv city')
```

```
class OrderCreationTest(TestCase):
```

```
    """
```

```
    Order And Order_content Model Test
```

```
    """
```

```
    @classmethod
```

```
    def setUpTestData(cls):
```

```
        publisher = Publisher.objects.create(publisher='Some Test publisher')
```

```
        book1 = Book.objects.create(
```

```
            title='Some Test Book',
```

```
            author='Aadwad',
```

```
            description='Adwadad',
```

```
            publish_year=2021,
```

```
            stock=3,
```

```
            price=120,
```

```
            publisher=publisher
```

```
        )
```

```
        book2 = Book.objects.create(
```

```
            title='Some Test Book2',
```

```
            author='Gdad',
```

```
            description='Dwdadafadwd',
```

```
            publish_year=2021,
```

```
            stock=3,
```

```
            price=120,
```

```

        publisher=publisher
    )
    user = User.objects.create(
        username='fafalya543',
        first_name='Alex',
        last_name='Doe',
        email='alex12355@gmail.com',
        gender='F',
    )
    order = Order.objects.create(user=user)
    Order_content.objects.create(
        order=order,
        book=book1,
        amount=3
    )
    Order_content.objects.create(
        order=order,
        book=book2,
        amount=4
    )

def test_order_creation_content(self):
    user = User.objects.get(id=1)
    order = Order.objects.get(id=1)
    order_content = Order_content.objects.filter(order=order)
    expected_user_username = f'{user.username}'
    expected_order_content_amount = f'{order_content[0].amount}'

```

```
self.assertEqual(expected_user_username, 'fafalya543')
```

```
self.assertEqual(expected_order_content_amount, 3)
```