

**A PROJECT REPORT**  
**“BOOK ORDERING SYSTEM”**

*Submitted by:*

**Md. Imran Hossen (ID:21222203001)**

**Rajesh Chandra Barman Roni (ID:21222203002)**

**Istiak Ahmed Arnob (ID:21222203028)**

**Bikash Chandra Ray (ID:21222203045)**

**Rohan Rashid (ID:21222203050)**

**A Project Submitted in Partial Fulfillment of the Requirements for the  
Degree of Bachelor of Science in Computer Science of the Bangladesh  
University of Business and Technology (BUBT)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**BANGLADESH UNIVERSITY OF BUSINESS AND TECHNOLOGY (BUBT)**  
**DHAKA-BANGLADESH**

**MAY, 2025**

## DECLARATION

I hereby declare that the project “Book Ordering System” submitted for the degree of Bachelor of Science Engineering in Computer Science and Engineering in the faculty of Computer Science and Engineering of Bangladesh University of Business and Technology (BUBT), is our original work and that it contains no material which has been accepted for the award to the candidates of any other degree or diploma, except where due reference is made in the next of the project to the best of our knowledge, it contains no materials previously published or written by any other person except where due reference is made in this project.

---

**Md. Imran Hossen**  
**ID: 21222203001**  
**Intake: 41**  
**Section:01**

---

**Rajesh Chandra**  
**Barman Roni**  
**ID: 21222203002**  
**Intake: 41**  
**Section:01**

---

**Istiak Ahmed Arnob**  
**ID: 21222203028**  
**Intake: 41**  
**Section:01**

---

**Bikash Chandra Ray**  
**Id: 21222203045**  
**Intake: 41**  
**Section:01**

---

**Rohan Rashid**  
**Id: 21222203050**  
**Intake: 41**  
**Section:01**

## APPROVAL

This project “Book Ordering System” report submitted by Md. Imran Hossen, Rajesh Chandra Barman Roni, Istiak Ahmed Arnob, Bikash Chandra Ray, and Rohan Rashid, students of Department of Computer Science and Engineering, Bangladesh University of Business and Technology (BUBT), under the supervision of Mr. Ali Azgar, Assistant Professor, Department of Computer Science and Engineering, has been accepted as satisfactory for the partial requirements for the degree of Bachelor of Science in Computer Science and Engineering.

---

**(Mr. Ali Azgar)**

**Assistant Professor & Project  
Supervisor  
Department of CSE, BUBT**

## ACKNOWLEDGEMENTS

We are deeply thankful to Bangladesh University of Business and Technology (BUBT) for providing us such a wonderful environment to peruse our project. We would like to express our sincere gratitude to Ali Azgar, Assistant Professor, CSE, BUBT. We have completed our project with his help. We found the project area, topic, and problem with his suggestions. He guided us with our study, and supplied us many articles and academic resources in this area. He is patient and responsible. When we had questions and needed his help, he would always find time to meet and discuss with us no matter how busy he was. We also want to give thanks to our CSE department. Our department provide us logistic supports to complete our project with smoothly. We would also like to acknowledge our team members for supporting each other and be grateful to our university for providing this opportunity for us.

## List of Figures

Figure 3.2.1 User Registration (User).....	5
Figure 3.2.2 User Login (User) .....	6
Figure 3.2.3 Book Search and Browsing (User).....	6
Figure 3.2.4 Ordering Books (User) .....	7
Figure 3.2.5 Cart (User) .....	7
Figure 3.2.6 Order Tracking and Feedback (User).....	8
Figure 3.2.7 Order Tracking and Feedback (User).....	8
Figure 3.2.8 Book Management (Admin).....	9
Figure 3.2.9 Order Management (Admin) .....	9
Figure 3.6.1 Waterfall Model.....	11
Figure 3.7.1 Activity Diagram of DBOS.....	12
Figure 3.7.2 Use Case Diagram.....	13
Figure 3.7.3 Database (Schema Diagram) .....	14
Figure 3.7.4 ER Diagram of Database.....	14
Figure 4.8.1 Fragmented Database (Localhost) .....	17

## ABSTRACT

The is a web-based platform developed to simplify the process of purchasing books online through a user-friendly interface and an efficient backend supported by a distributed database system. The main objective of this project is to modernize the traditional book purchasing experience and provide a seamless solution for both customers and administrators across multiple regional branches. The motivation behind this project stems from the inefficiencies in current systems, where physical visits to bookstores are required, inventory is managed manually, and there is no real-time visibility of stock. Our system addresses these problems by offering a 24/7 online portal, allowing customers to browse books by category, search using keywords such as title, author, or ISBN, manage a shopping cart, and place orders remotely. Admin users can manage inventory, add or remove books, and process orders efficiently. The system is implemented using HTML, CSS, PHP, and MySQL, with a distributed database model across three branches—Dhaka, Rajshahi, and Chittagong—to ensure better data availability and load balancing. A modular design pattern was followed to separate front-end and back-end functionalities, and the waterfall model was adopted for the development lifecycle. The system includes key components such as registration/login, search functionality, category filters, shopping cart, checkout, admin dashboard, and customer feedback. Testing was conducted in two phases: unit testing for individual modules and integration testing to ensure all modules work together correctly. The system passed all test cases and demonstrated stability, usability, and responsiveness under various conditions. The final outcome is a scalable, distributed, and fully functional that can be further extended to include features like mobile applications, online payment integration, and AI-based book recommendations. This project contributes to the ongoing digital transformation of retail systems and offers a practical solution for improving customer experience and inventory management in the book retail industry.

## LIST OF ABBREVIATIONS AND ACRONYMS

Abbreviation	Full Form
<b>DDBS</b>	<b>Distributed Database System</b>
<b>DBOS</b>	<b>Distributed Book Ordering System</b>
<b>XAMPP</b>	<b>Cross-platform, Apache, MySQL, PHP, and Perl</b>
<b>LAMP</b>	<b>Linux, Apache, MySQL, PHP/Python/Perl</b>
<b>PC</b>	<b>Personal Computer</b>
<b>ISBN</b>	<b>International Standard Book Number</b>
<b>RAM</b>	<b>Random Access Memory</b>
<b>HTML</b>	<b>HyperText Markup Language</b>
<b>CSS</b>	<b>Cascading Style Sheets</b>
<b>Java</b>	<i>(Not an acronym; programming language)</i>
<b>SCRIPT</b>	<i>(General term for scripting languages; not an acronym)</i>
<b>PHP</b>	<b>Hypertext Preprocessor</b> <i>(formerly Personal Home Page)</i>
<b>UX</b>	<b>User Experience</b>
<b>UI</b>	<b>User Interface</b>
<b>MySQL</b>	<i>(“My” is a name + SQL: Structured Query Language)</i>
<b>DFD</b>	<b>Data Flow Diagram</b>

# TABLE OF CONTENTS

<b>Declaration .....</b>	<b>ii</b>
<b>Approval .....</b>	<b>iii</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>iv</b>
<b>List of Figures.....</b>	<b>v</b>
<b>Abstract .....</b>	<b>vi</b>
<b>LIST OF ABBREVIATIONS AND ACRONYMS .....</b>	<b>vii</b>
 <b>Chapter 1: INTRODUCTION.....</b>	 <b>1-2</b>
1.1 Introduction .....	1
1.2 Problem statements.....	1
1.3 Current Systems.....	1
1.4 Problem with the Current Systems.....	1
1.5 Proposed System .....	2
1.6 Project Objectives.....	2
1.7 Motivations of the Project.....	2
1.8 Project Contributions.....	2
1.9 Organization of Report.....	2
1.10 Summary .....	
 <b>Chapter 2: LITERATURE REVIEW.....</b>	 <b>3-4</b>
2.1 Introduction .....	3
2.2 Related Works.....	4
2.3 Problem Analysis.....	4
2.3 Summary.....	4
	2
 <b>Chapter 3: Proposed Model/System.....</b>	 <b>5-15</b>
3.1 Introduction.....	5
3.2 Requirement Analysis.....	5-9
3.3 Hardware Requirement Analysis.....	10
3.3.1 Computer.....	10
3.3.2 Server.....	10
3.4 Software Requirement Analysis.....	10
3.4.1 HTML.....	10
3.4.2 CSS.....	10
3.4.3 PHP.....	10
3.4.4 Database.....	10
3.5 System Architecture.....	10-11
3.6 Methodology.....	11-12
3.7 Data Flow Diagram.....	12-14



	3.8 Summary.....	15
<b>Chapter 4:</b>	<b>IMPLEMENTATION AND TESTING .....</b>	<b>15-18</b>
	4.1 Introduction.....	15
	4.2 User Interface.....	15
	4.3 Categories.....	15
	4.4 Registration Form.....	16
	4.5 Cart.....	16
	4.6 Search .....	16
	4.7 Admin Control Panel .....	17
	4.8 Database Design.....	17
	4.9 Testing .....	18
	4.10 Summary .....	18
<b>Chapter 5:</b>	<b>USER MANUAL .....</b>	<b>19-20</b>
	5.1 Introduction .....	19
	5.2 Home page.....	19
	5.3 Login.....	19
	5.4 Order.....	19
	5.5 Payment Option.....	20
	5.6 Confirm Order Option.....	20
	5.7 Search.....	20
	5.8 Feedback.....	20
<b>Chapter 6:</b>	<b>CONCLUSION AND FURTHER WORK.....</b>	<b>21</b>
	6.1 Conclusion .....	21
	6.2 Future plan.....	21
	<b>REFERENCES.....</b>	<b>22</b>

# Chapter 1: Introduction

---

## 1.1 Introduction

The is a web-based platform designed to simplify the process of purchasing books online.[1] It enables users to browse, search, and order books conveniently from their devices without needing to visit physical stores. This system is built on a distributed architecture that manages data across multiple branches—Dhaka, Rajshahi, and Chittagong—ensuring real-time synchronization and availability of inventory. Key challenges addressed during development include maintaining data consistency across branches and designing a user-friendly interface that supports smooth navigation and ordering.[2]

## 1.2 Problem Statements

- Customers cannot access book inventories remotely and must visit stores physically.
- Manual inventory management leads to inaccuracies and delays in stock updates.
- Lack of real-time order tracking and updates causes customer dissatisfaction.
- Admins face difficulties managing orders and stock efficiently due to lack of centralized control.

## 1.3 Current Systems

Currently, book purchases are done primarily through physical bookstores. Inventory management is handled manually at each branch, which causes delays and errors in stock updates. Customers have no online access to view or order books, limiting convenience and reach.[3]

## 1.4 Problems with the Current Systems

- No online portal for customers to browse or order books.
- Inventory is updated manually, often leading to inconsistencies.
- Customers cannot check book availability or place orders remotely.
- The entire purchasing process is time-consuming and inefficient for both customers and staff.

## 1.5 Proposed System

The proposed system introduces an online book ordering platform with the following features:

- A web-based portal accessible anytime and anywhere for customers.
- Distributed databases across the Dhaka, Rajshahi, and Chittagong branches, synchronized regularly.
- Real-time inventory updates and search functionality.
- User-friendly interface with easy navigation, registration, and secure ordering.
- Admin panel for efficient stock and order management.

## 1.6 Project Objectives

- Develop a functional, responsive web-based .
- Implement distributed data management to synchronize branch inventories.
- Ensure a smooth, user-friendly experience for both customers and administrators.
- Automate and simplify the ordering and inventory tracking processes.

## 1.7 Motivations of the Project

The motivation behind this project is to modernize book purchasing by addressing the limitations of the existing manual system. Increasing internet penetration has created demand for an online platform that is efficient, reliable, and scalable. The project seeks to improve customer convenience, reduce manual errors, and enable better management for store admins.[4]

## 1.8 Project Contributions

- A centralized yet distributed database system providing real-time book availability.
- Secure and fast online ordering capabilities.
- Administrative tools for streamlined order processing and stock management.
- A scalable system capable of handling growth in users and inventory.

## 1.9 Organization of Report

- **Chapter 2:** Literature review and related works.
- **Chapter 3:** System analysis and design details.
- **Chapter 4:** Implementation and testing.
- **Chapter 5:** User manual describing system operation.
- **Chapter 6:** Conclusion and future work recommendations.

## 1.10 Summary

This chapter introduced the , outlining the problems with current manual processes, the motivation for developing an online distributed system, and the objectives and contributions of the project. The following chapters will explore the system design, development, and usage in detail.

# CHAPTER 2: LITERATURE REVIEW

---

## 2.1 Introduction

This chapter analyzes prior works and existing systems related to book ordering and inventory management. It aims to identify the strengths and limitations of current approaches and justify the need for our proposed distributed .[5]

**Yunkai et al.** have explored the enhancement of order processing systems through effective status tracking mechanisms. Their research shows that dividing orders into five key statuses—Pending, Processed, Shipped, Delivered, and Cancelled—improves clarity and operational efficiency. This structure allows both customers and system managers to monitor the progress of orders in real time. They emphasize that real-time tracking, when integrated into online bookstore platforms, significantly enhances user satisfaction and reduces management complexity. Their approach highlights the value of a status-based model for scalable and user-friendly order processing. [6]

**Dhanush Kumar et al.** propose an online bookstore system combining intelligent book recommendations and inventory optimization. The system uses collaborative filtering to suggest books based on user preferences and includes a dashboard for tracking and managing stock levels. Prior research highlights the importance of personalized recommendations in enhancing user engagement and the role of data analytics in improving inventory decisions. This integrated approach aims to improve both user satisfaction and operational efficiency. [7]

**Gnana Singh et al.** in "Distributed Database for Bookstore Application" propose a distributed database to address issues in centralized systems like data loss during crashes and access bottlenecks. The system stores bookstore data across multiple sites, ensuring high availability, fault tolerance, and accessibility. Using MySQL and SQL, it maintains ACID properties and supports data security, working even offline. This approach improves the efficiency and reliability of managing large data in bookstores. [8]

**Ezéchiél Et Al.** (2019) Reviewed Distributed Database Systems (DDBS) Focusing On Three Strategies: Data Fragmentation, Allocation, And Replication. Key Challenges Include Query Optimization For Fragmented Data, Np-hard Allocation Problems, And Synchronization In Replicated Systems. The Paper Proposes Solutions For Efficient Synchronization, Particularly In Decentralized Peer-to-peer (P2P) Setups, Highlighting The Need For Scalability And Autonomy.[9]

**Santiago Et Al.** (2021) Explored The Integration Of Distributed Database Systems (DDBS) In Data Security, Emphasizing Scalable And Expandable Access Control Models. The Paper Discusses The Challenges Of Implementing Access Control, Such As The Complexity Of DDBS, Data Integrity, And The Need For Skilled Management. It Also Examines Real-world Applications In Sectors Like IT And Social Media, Stressing The Importance Of Robust Security Measures And Future Research In Enhancing Data Protection.[10]

## 2.2 Related Works

Several online book-selling platforms exist both globally and locally, such as Amazon, Rokomari, and Daraz. These platforms use centralized database systems, which manage user registration, book inventory, and orders from a single data center.[11]

- **Amazon** uses a powerful cloud-based infrastructure but focuses more on global distribution.
- **Rokomari** is a Bangladeshi online bookstore that offers a centralized model with basic features like user registration, ordering, and delivery.
- **Daraz** operates across product categories, including books, but lacks book-specific features like library categorization or educational tagging.

These systems provide automation but fall short when it comes to **regional optimization**, **system fault tolerance**, and **scalability** under distributed environments.

## 2.3 Problem Analysis

Despite the existence of such systems, many challenges remain unaddressed:

- **Single Point of Failure:** Centralized systems are vulnerable. If the main server fails, the whole platform becomes inaccessible.
- **Scalability Issues:** As traffic increases, centralized systems struggle with load balancing.
- **Regional Accessibility:** There's no localized data management, causing latency and slower access for users outside the primary data center's location.
- **No Branch-Level Customization:** Book availability and pricing cannot be efficiently managed per branch in centralized systems.

These issues highlight the importance of moving towards **distributed database systems**, particularly for geographically distributed services like branch-wise bookstores.

## 2.4 Summary

The review shows that while existing systems are functional, they do not fully utilize the potential of **distributed computing** in book ordering services. Our proposed system introduces a **branch-wise distributed architecture**, ensuring fault tolerance, regional customization, and efficient data access. It addresses the core challenges found in related works and sets the foundation for scalable, real-time book ordering across multiple locations.

# CHAPTER 3: PROPOSED SYSTEM

---

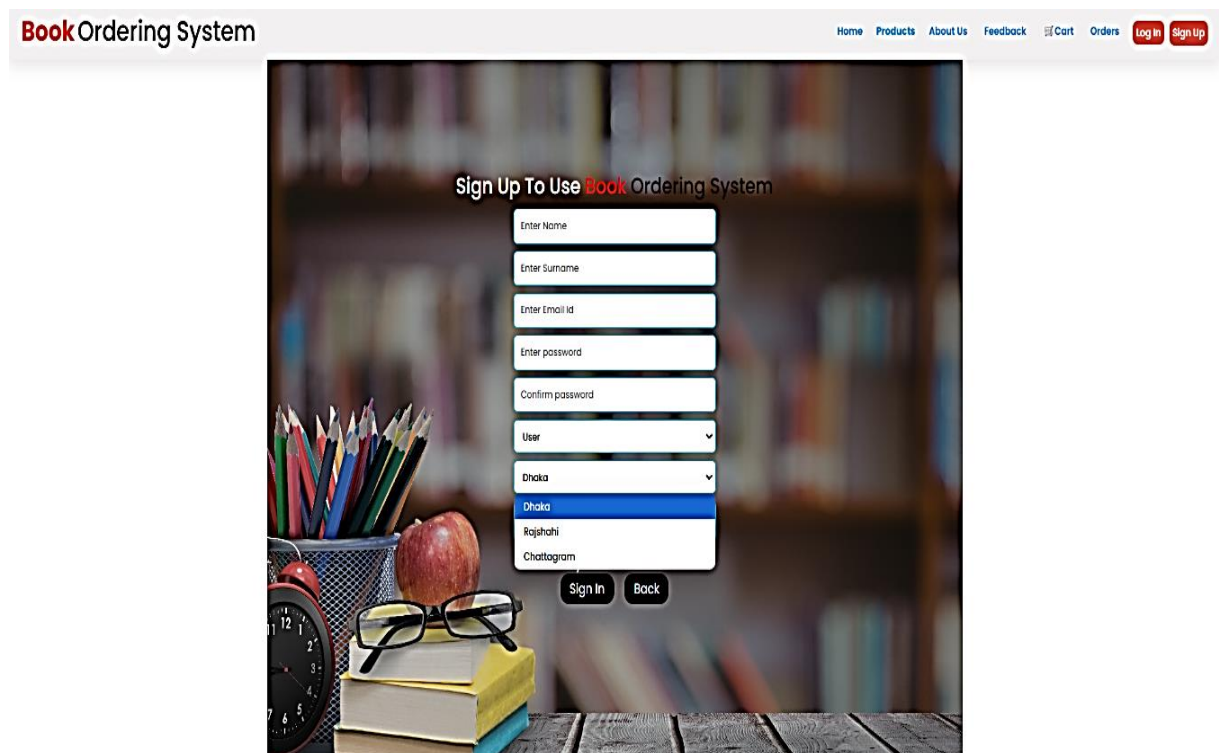
## 3.1 Introduction

This chapter explains the proposed system architecture, development approach, hardware/software requirements, and the design methodology adopted for the – “Book Ordeing System.” It covers how we approached building a scalable, distributed, and user-friendly platform to meet the identified needs of both customers and administrators across multiple locations.

## 3.2 Requirement Analysis

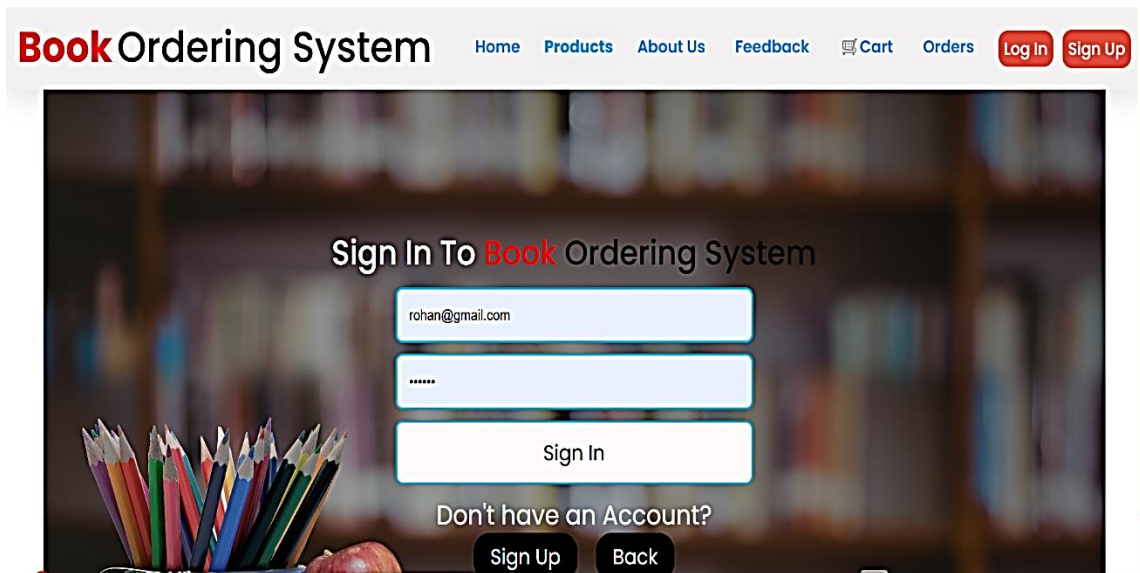
**User Requirements:**

**User Registration:** Customers must be able to create accounts based on branch selection.



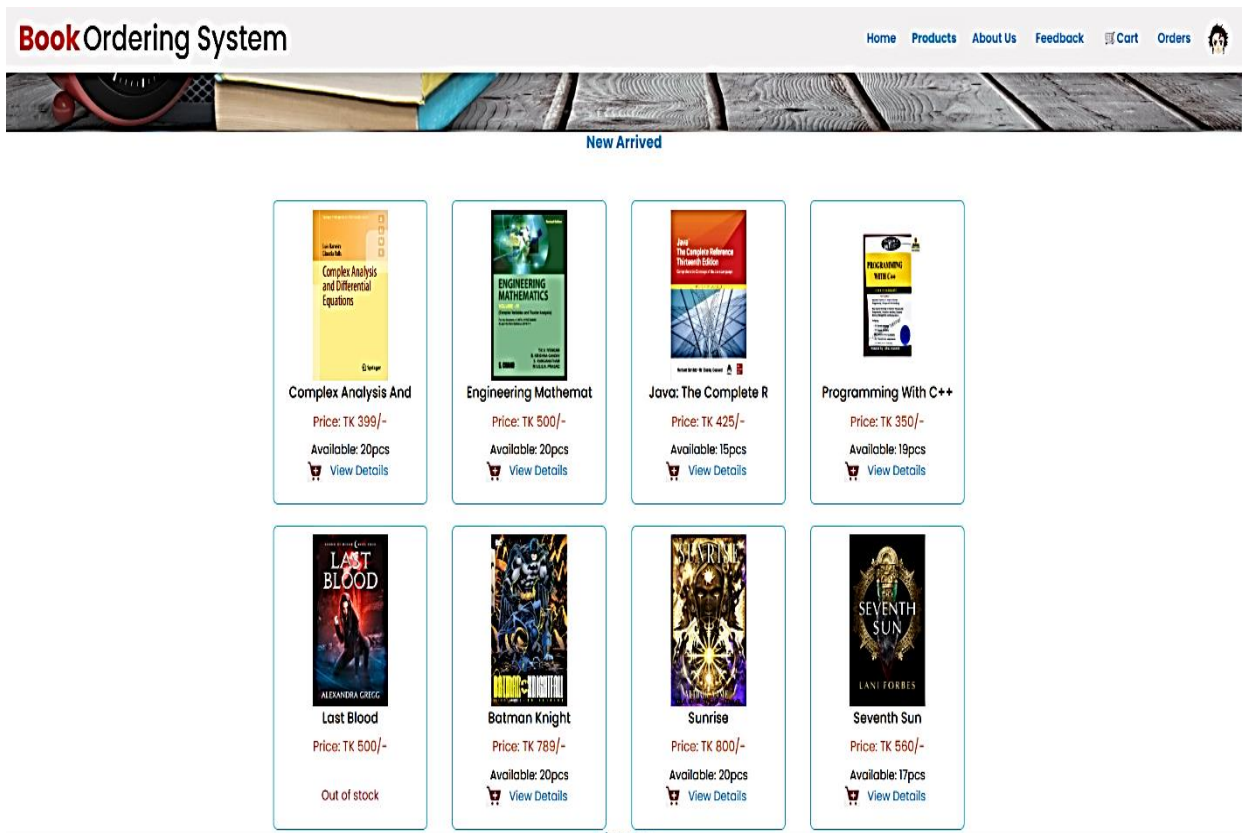
The screenshot displays the 'Sign Up To Use Book Ordering System' interface. The page features a header with the site name 'Book Ordering System' and navigation links: Home, Products, About Us, Feedback, Cart, Orders, Log In, and Sign Up. The main content area is a sign-up form with the following fields: Enter Name, Enter Surname, Enter Email id, Enter password, Confirm password, User (dropdown menu), and Dhaka (dropdown menu). The 'Dhaka' dropdown is open, showing options: Dhaka, Rajshahi, and Chattogram. At the bottom of the form are 'Sign In' and 'Back' buttons. The background of the form is a blurred image of a desk with books, a clock, and a pencil holder.

**Fig: 3.2.1**



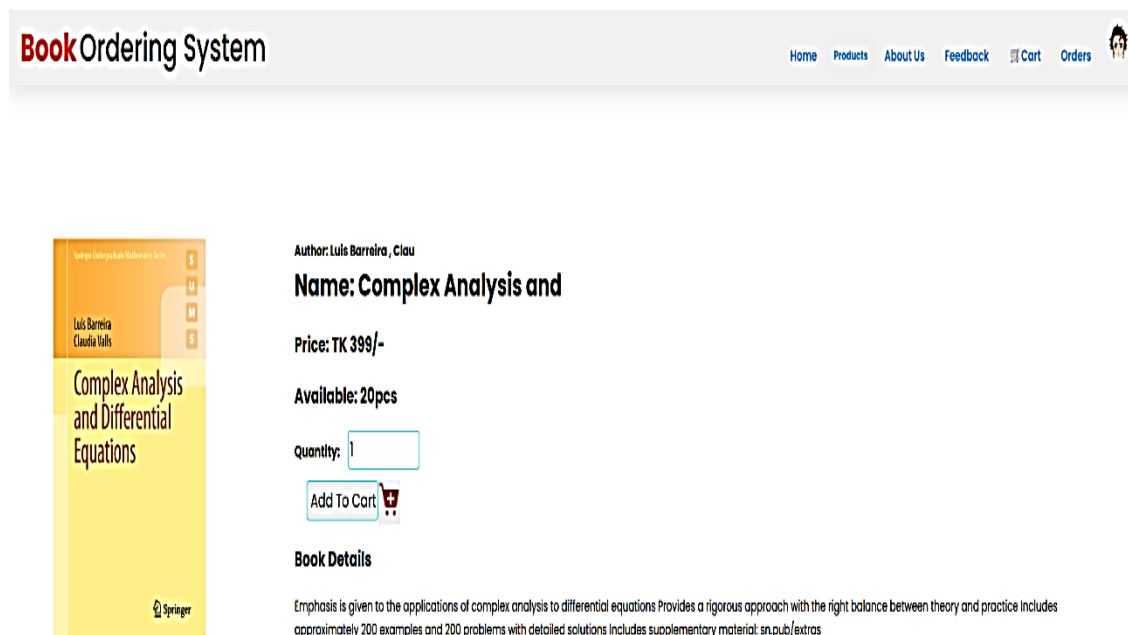
**Fig: 3.2.2**

- **Book Search and Browsing:** Users should be able to browse and search books by category, author, or keyword.

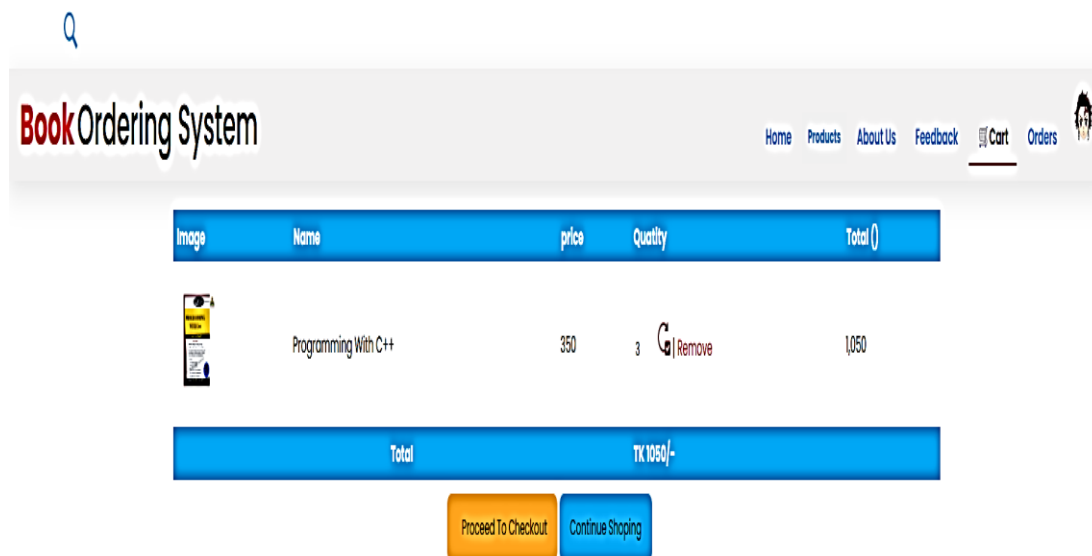


**Fig: 3.2.3**

- **Ordering Books:** Customers must add books to the cart, place orders, and receive order confirmation.



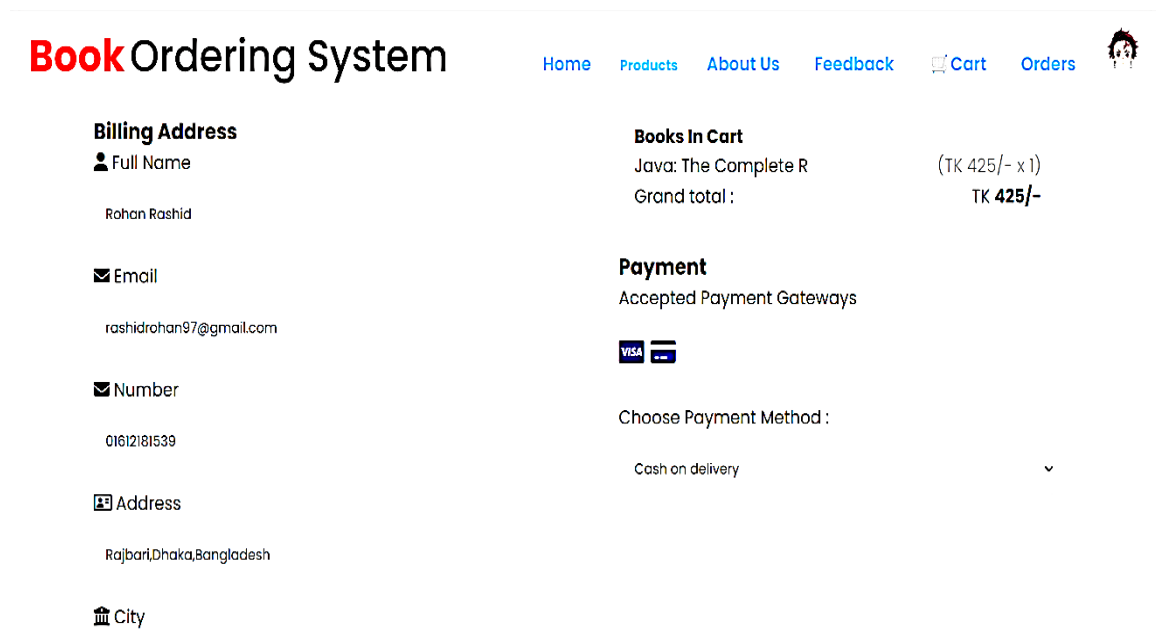
**Fig: 3.2.4**



**Fig: 3.2.5**



- **Order Tracking and Feedback:** Users can check order status and provide feedback or reviews.



**Book Ordering System**

Home Products About Us Feedback Cart Orders

**Billing Address**

Full Name  
Rohan Rashid

Email  
rashidrohan97@gmail.com

Number  
01612181539

Address  
Rajbari, Dhaka, Bangladesh

City

**Books In Cart**

Java: The Complete R (TK 425/- x 1)  
Grand total : TK 425/-

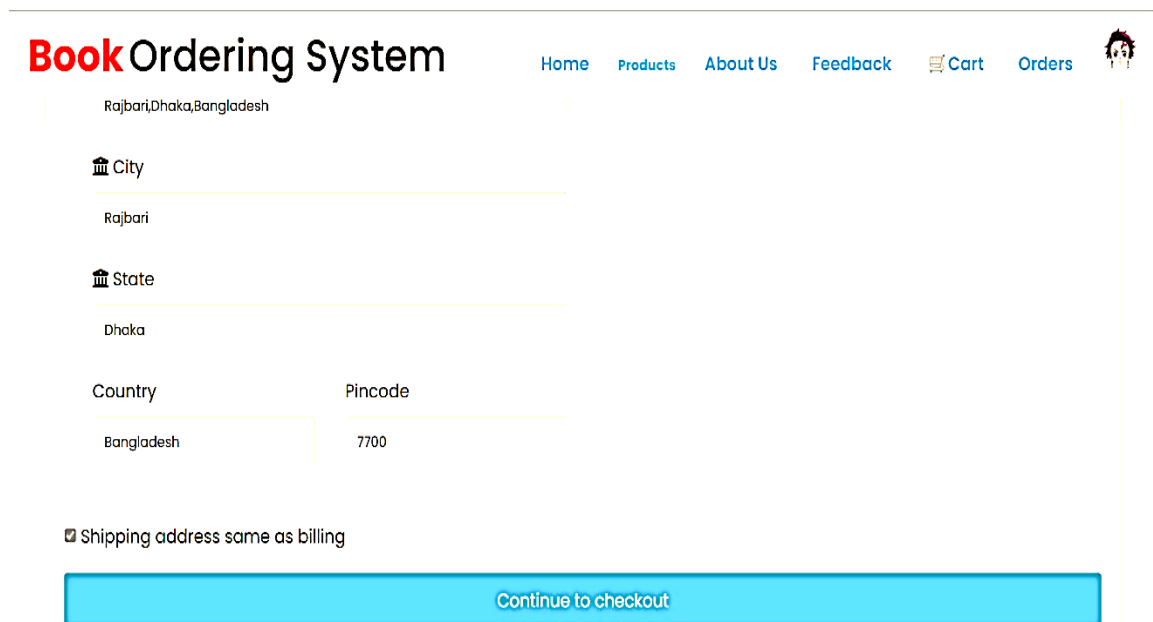
**Payment**

Accepted Payment Gateways

VISA MasterCard

Choose Payment Method :  
Cash on delivery

**Fig: 3.2.6**



**Book Ordering System**

Home Products About Us Feedback Cart Orders

Rajbari, Dhaka, Bangladesh

City  
Rajbari

State  
Dhaka

Country  
Bangladesh

Pincode  
7700

☒ Shipping address same as billing

Continue to checkout

**Fig: 3.2.7**

## Admin Requirements:

- **Book Management:** Admins can add, edit, and remove book entries including title, author, price, and stock.

The screenshot shows the 'Book Ordering System' Admin Panel. At the top, there's a navigation bar with links: Home, Add Books, Orders, Message, Users. The user is logged in as 'admin' with a 'logout' link. Below the navigation bar, there's a 'See All Books' button. The main section is titled 'Add Books To Books For You'. It contains a form with the following fields: 'Enter book Name', 'Enter Author name', 'Enter product price', a dropdown menu for 'Adventure', 'Enter book description', 'Enter product quantity', and a file upload section with 'Choose File' and 'No file chosen' buttons. Below the form is an 'Add Book' button. At the bottom, there's a list of books. The first book is 'Complex Analysis And' by 'Luis Barreira, Clau' with a price of 'TK 399/-'. The second book is 'Engineering Mathematics' by 'Dr. T. K. V. Iyengar, D' with a price of 'TK 500/-'. Each book entry has 'Update' and 'Delete' buttons.

Fig: 3.2.8

- **Order Management:** Admins view, approve, or cancel customer orders and track delivery status.

The screenshot shows the 'PLACED ORDERS' section. It contains two order details. The first order is dated '19-May-2025', has an 'Order Id' of '#35', and is for 'Rone'. The second order is dated '24-May-2025', has an 'Order Id' of '#0', and is for 'Mr. Ali Azgar'. Both orders show the 'Payment Method' as 'cash on delivery' and the 'Payment Status' as 'pending'. Each order entry has 'Update' and 'Delete' buttons.

Fig: 3.2.9

- **User Management:** Admins can manage registered users and their activity.
- **Reporting Tools:** Admins can generate reports such as top-selling books, revenue summaries, and user activity logs.

### 3.3 Hardware Requirements

#### Client-Side (User Device):

- Minimum RAM: 2 GB
- Browser: Latest version of Chrome, Firefox, or Edge
- Device: PC, tablet, or smartphone

#### Server-Side:

- Processor: Minimum 2 GHz dual-core CPU
- Memory: 4 GB RAM or more
- Storage: 50 GB minimum
- Web Server: Apache Server (via XAMPP or LAMP stack)
- Network: Stable internet connection for continuous database syncing across branches

### 3.4 Software Requirements

- **Frontend Technologies:** HTML, CSS, JavaScript – used for designing responsive and interactive user interfaces.
  - **Backend Language:** PHP – server-side scripting to handle business logic and database operations.
  - **Database System:** MySQL – deployed in a distributed fashion to support multiple branches.
  - **Server Environment:** XAMPP (for Windows) or LAMP (for Linux) to provide the Apache, MySQL, and PHP stack.
  - **Version Control:** Git for source code management and collaboration.
- 

### 3.5 System Architecture

The system is built on a **three-tier architecture**:

1. **Presentation Layer (Frontend):**
  - Interface for users and administrators.
  - Designed with HTML, CSS, and JavaScript.
  - Responsible for collecting inputs and displaying outputs.
2. **Logic Layer (Backend):**
  - Handles user authentication, book operations, and order processing.
  - Implemented in PHP.
  - Communicates with the database layer to store and retrieve data.
3. **Data Layer (Database):**

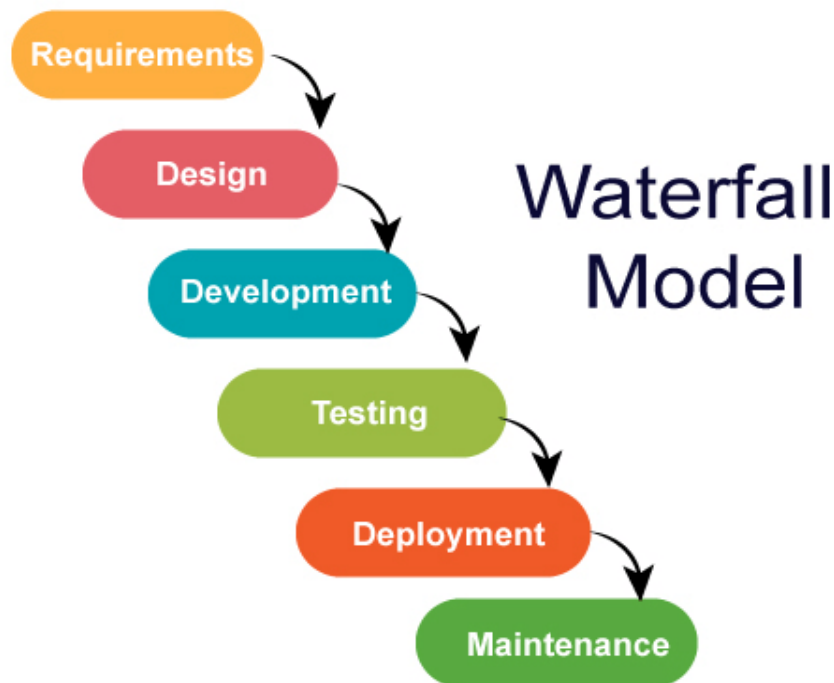
- Uses distributed MySQL databases with identical schemas across the three branches (Dhaka, Rajshahi, Chittagong).
- Maintains book data, user records, and transaction logs.

This architecture separates concerns for scalability, easier maintenance, and better performance across branches.

---

### 3.6 Methodology

The **Waterfall Model** was chosen for development due to its simplicity and structured approach. The phases followed were:



**Fig: 3.6.1**

1. **Requirement Gathering:** Defined system goals and functional needs from both users and admins.
2. **System Design:** Designed UI mockups, database schema, and flow diagrams.
3. **Implementation:** Developed modules for user and admin features with database integration.
4. **Testing:** Performed unit testing, integration testing, and system testing to ensure functionality.

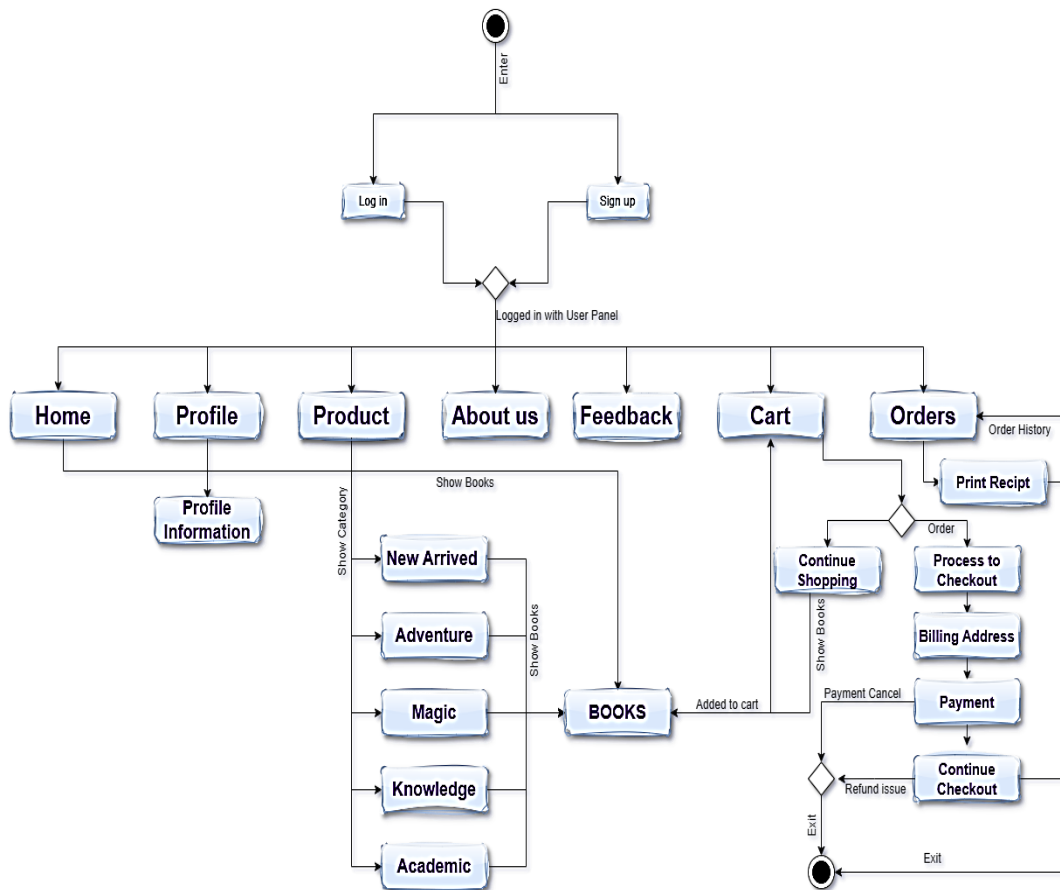
5. **Deployment and Maintenance:** Deployed on localhost for simulation and prepared for future hosting.

This model ensures clarity of objectives at each stage, suitable for academic and systematic project development.

### 3.7 Diagram :

#### 3.7.1 Showing a activity diagram of how data flows between modules.

**Activity Diagram of Boook Ordering Sytem**



**Figure:3.7.1:** Activity diagram Of BOS

The system DFD illustrates the flow of data within the application. It includes:

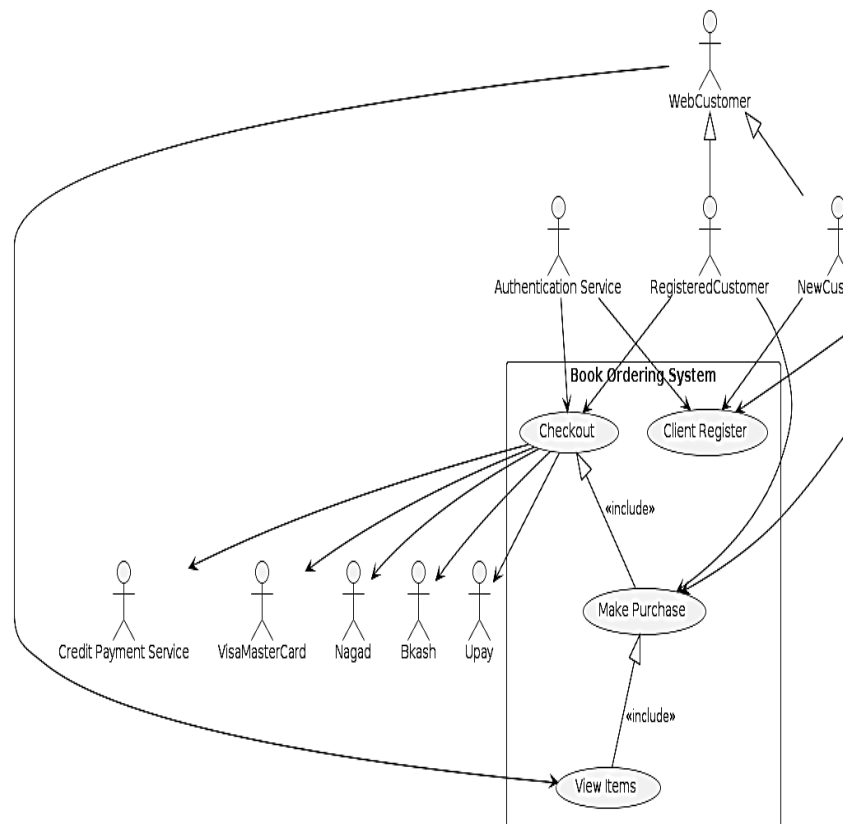
1. **User Login Process:**

- Input: Username & Password

- Output: Session initiated upon successful validation.
- 2. **Book Browsing & Searching:**
  - Query from the user to the database
  - Response with matching book listings
- 3. **Cart Management:**
  - User adds selected books to the cart
  - Backend stores this temporarily until checkout
- 4. **Order Placement:**
  - Cart items sent to the order processing module
  - Order record saved in the database
- 5. **Database Update:**
  - Inventory updated after order confirmation
  - Admin notified for processing and dispatch

A detailed diagram (not included in text format) can represent this flow using standard DFD notations.

### 3.7.2 Show interaction between users, administrators, and the system



**Figure:3.7.2: Use Case Diagram**

### 3.7.3 Highlight table structure for Users, Books, and Orders.

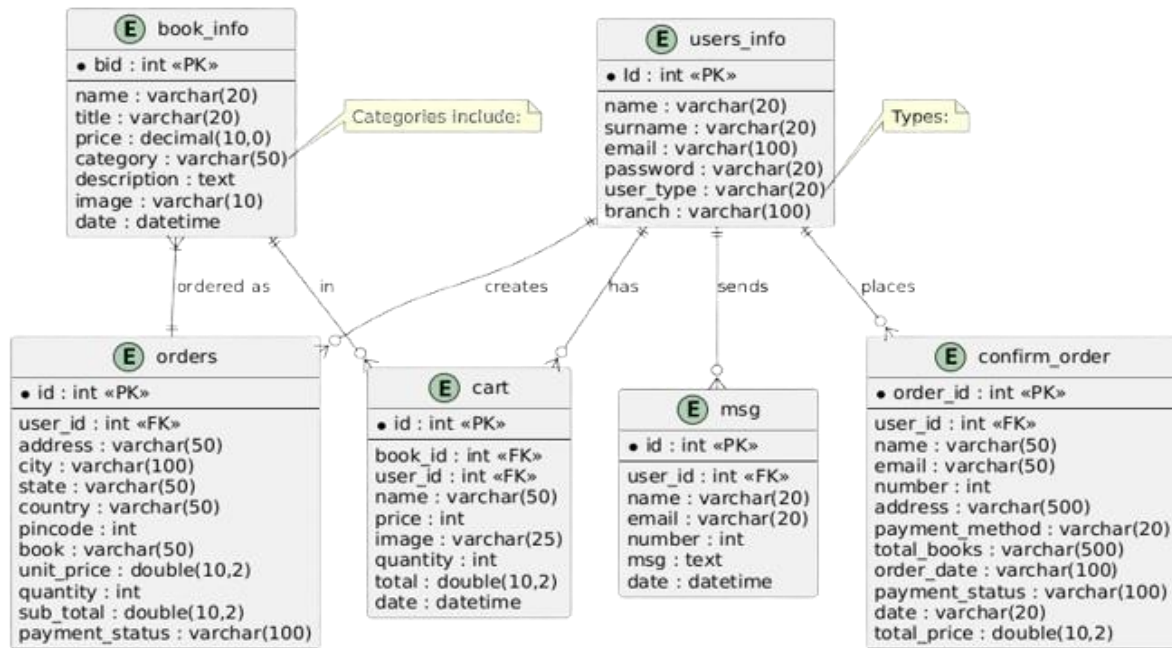


Figure:3.7.3: Database (Schema Diagram)

### 3.7.4 Highlight relationships between key entities like Users, Books, and Orders

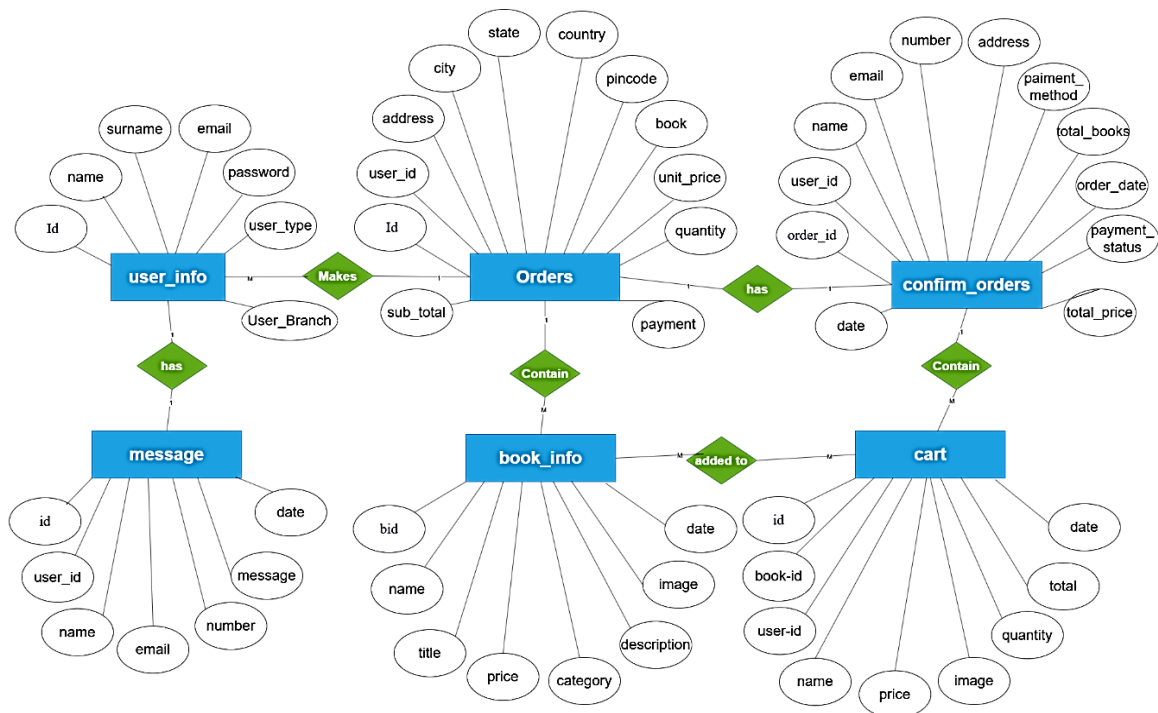


Figure:3.7.4: ER Diagram of Database

## 3.8 Summary

This chapter described the architectural and technological foundation of the **Book Ordering System**. It outlined hardware/software needs, described the layered design, and explained the development methodology. The system was structured for flexibility, ease of maintenance, and enhanced performance using distributed databases and a three-tier web-based approach.

# Chapter 4: Implementation and Testing

---

## 4.1 Introduction

This chapter discusses the practical implementation of the , highlighting both frontend (user interface) and backend functionalities. It also covers the cart system, admin features, distributed database setup, and the testing processes used to validate the system.

## 4.2 User Interface

The system offers a **simple, user-friendly, and responsive UI**. Key interface elements include:

- A consistent **navigation bar** with links to Home, Categories, Cart, Contact, and Login/Register pages.
- A dynamic **book display area** that uses cards to show book covers, prices, and “Add to Cart” buttons.
- A sticky header for improved UX on mobile and desktop devices.
- Alerts and pop-ups to confirm actions like adding items to the cart or placing an order.

## 4.3 Categories

Books are classified into multiple **categories** to improve browsing and filtering:

- Fiction
- Academic
- Biography



- Technology
- Magic
- Others

Each category groups relevant books, allowing users to quickly locate their interests. Filtering is implemented using simple SQL WHERE clauses based on the category field in the book\_info table.

## 4.4 Registration

User registration is **mandatory for placing orders**. Users must provide:

- Full name
- Email address
- Password
- Contact number

On successful registration, users can log in and access features like adding to cart, checking out, viewing past orders, and sending messages.

Password hashing is implemented in the backend for security.

## 4.5 Cart System

The **cart system** is fully functional and allows users to:

- Add books to the cart
- Update quantity
- Remove books
- View subtotal and total
- Proceed to checkout

Cart data is stored in the cart table, and is linked to each specific user\_id. The cart system also handles duplicate entries by updating quantities instead of duplicating rows.

## 4.6 Search Functionality

A **real-time search bar** allows users to find books by:

- Title
- Author (from the title field in the schema)
- ISBN (if implemented)

Search queries are processed using SQL LIKE operations with wildcards. This improves the discoverability of books and provides a better user experience.

## 4.7 Admin Panel

The system includes a secure **admin panel** where authorized administrators can:

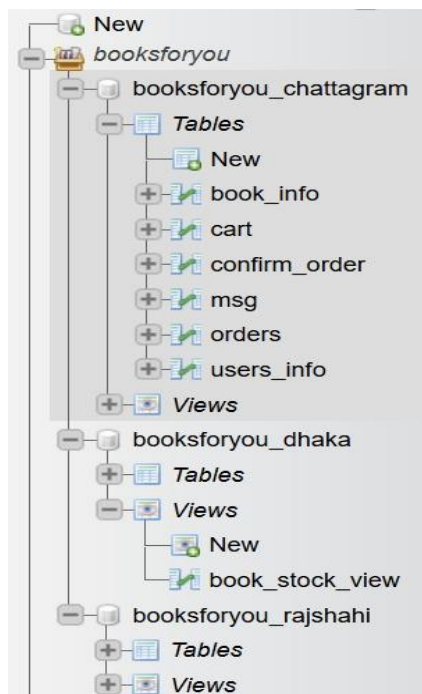
- **Add/edit/delete books** (via the book\_info table)
- **Manage stock levels**
- **View and confirm customer orders** (from the orders and confirm\_order tables)
- **Monitor messages and inquiries** (via the msg table)

Admins also have access to order statistics, and the dashboard displays real-time information on inventory and sales.

## 4.8 Database Design

The system uses a **distributed database architecture** with **three branches**:

- **Dhaka**
- **Chittagong**
- **Rajshahi**



**Fig: 4.8.1**

Each branch maintains an **identical schema** consisting of the following tables:

- book\_info
- cart
- confirm\_order
- orders
- msg
- book\_stock\_view

**Synchronization** is handled periodically using scripts or tools that ensure consistency between branches. This approach improves **availability, fault tolerance, and load balancing**.

## 4.9 Testing

**Comprehensive testing** was carried out across all modules:

- **Unit Testing:**
  - Each module (cart, login, book add/remove, etc.) was tested individually.
  - Handled boundary cases like empty inputs, invalid data, and duplicate items.
- **Integration Testing:**
  - Verified interactions between components such as placing an order (cart → orders → confirmation).
  - Database and UI interactions were tested together to ensure data integrity.

**Result:** All test cases passed. The system met functional and non-functional requirements including performance, responsiveness, and usability.

## 4.10 Summary

The was successfully implemented with all major components functioning as intended. A responsive and intuitive interface, secure backend, and distributed database setup contribute to the overall usability and stability of the system. Testing confirmed the system's readiness for deployment, and it provides a solid foundation for further enhancements like online payment integration and advanced reporting.

# Chapter 5: User Manual

---

## 5.1 Introduction

This user manual is designed to help both regular users and administrators understand how to use the . It provides step-by-step guidance on using key features such as browsing books, placing orders, managing the cart, and handling admin tasks.

## 5.2 Home Page

When you visit the website, you'll land on the **Home Page**. Here you can:

- Browse all **available books**
- View **featured categories** like Fiction, Academic, Biography, etc.
- Click on any book to see more details like price and description

It's the starting point for exploring everything the system offers.

## 5.3 Login

To place an order or use the cart:

- You must **log in** with your registered email and password
- If you're a new user, you can easily **register** by filling out a short form with your name, email, and password

Once logged in, you'll have access to all features like cart, orders, and feedback.

## 5.4 Ordering Books

To order a book:

1. Browse the book list or use the search bar
2. Click on **"Add to Cart"** next to the book you want
3. Go to your **Cart**, update quantity if needed
4. Click **"Proceed to Checkout"**

5. Confirm your shipping details
6. Click “**Place Order**”

That’s it! Your order will be placed successfully.

## 5.5 Payment

Currently, the system supports **Cash on Delivery (COD)**:

- You don’t need to pay online
- Just pay when the book arrives at your address

*Note: Online payment options (like card, mobile banking) may be added in future updates.*

## 5.6 Confirmation

After placing an order:

- You will see an **Order Confirmation Message**
- A **tracking number** will be shown to help you follow your order status
- You can also check your **order history** from your account dashboard

## 5.7 Search

To quickly find a book:

- Use the **search bar** at the top of the page
- You can search by:
  - **Book title**
  - **Author name**
  - **ISBN (if available)**

This makes it easy to locate specific books without browsing through all categories.

## 5.8 Feedback

Users can **submit feedback or reviews** in the contact or feedback section:

- Share your experience with a specific book or with the service overall
- Admins can view this feedback and respond or improve service accordingly

Your feedback helps improve the system for everyone!

# Chapter 6: Conclusion and Future Work

---

## 6.1 Conclusion

The has been successfully developed as a complete and functional platform for online book purchasing. It provides users with a **simple and user-friendly interface** for browsing books, placing orders, and tracking purchases. The admin panel ensures **easy management of inventory**, orders, and users.

One of the key strengths of the system is its **distributed database design**, allowing smooth operations across multiple branches—**Dhaka, Rajshahi, and Chittagong**. This ensures better performance, data consistency, and reliability, especially as the system scales.

Overall, the system is secure, efficient, and designed to handle day-to-day operations of an online bookstore with ease.

## 6.2 Future Plans

While the current system is robust, there are several enhancements planned for the future to improve user experience and system performance:

- **Mobile App Version**  
A dedicated mobile application for Android and iOS will be developed to make book ordering even more convenient on smartphones and tablets.
- **Integration of Online Payments**  
The current system uses cash on delivery. Future updates will include secure **online payment options** such as credit/debit cards, mobile banking, and digital wallets.
- **AI-Based Book Recommendation System**  
An intelligent recommendation engine will be added to suggest books based on user preferences, purchase history, and browsing behavior.
- **Enhanced Analytics for Admin**  
Advanced analytics and data visualization tools will be provided for admins to monitor sales trends, stock status, and user engagement in real time.

## REFERENCES

1. Shah, A.J., *Design and implementation of a web based book buying/selling system for students using three tier architecture*. 2011, California State University, Northridge.
2. Vayadande, K., et al., *Challenges Faced in Web Development: A Survey Paper*. Grenze International Journal of Engineering & Technology (GIJET), 2024. **10**.
3. Lai, J.Y. and C.Y. Chang, *User attitudes toward dedicated e-book readers for reading: the effects of convenience, compatibility and media richness*. Online information review, 2011. **35**(4): p. 558-580.
4. Niemelä, E., *A Solution Retrieval Engine for a Customer-Facing Software Project Management System*. 2022.
5. Belapurkar, A., et al., *Distributed systems security: issues, processes and solutions*. 2009: John Wiley & Sons.
6. Zhai, Y. and W. Lu. *The online bookstore*. in *MATEC Web of Conferences*. 2017. EDP Sciences.
7. Kumar, D., et al., *Online Bookstore and Management System*. International Journal of Scientific Research and Technology, 2025.
8. Gnana Singh, D., E.J. Leavline, and S. Malini, *Distributed Database for Bookstore Application*. International Journal of Advanced Research in Computer Science, 2017. **8**(3).
9. Ezéchiél, K., S. Kant, and R. Agarwal, *A systematic review on distributed databases systems and their techniques*. Journal of theoretical and applied information technology, 2019. **96**(1): p. 236-266.
10. Santiago, M., *Review on Distributed Database Systems*. Mateo Santiago.(2021). REVIEW ON DISTRIBUTED DATABASE SYSTEMS. International Journal of Innovations in Engineering Research and Technology, 2021. **8**(07): p. 334-336.
11. Ahmad, B., *Sales channel renaissance: A case study on elevating performance in small-scale publishing*. 2023.