

Customer Review System on Amazon Bookstore Application

Joshnavi Chenreddy

Ashwitha Eravelly

Singireddy sahithi

Sai Sruthi Thileti

Table of Contents

1. Introduction	4
2. Customer Statement of Requirements	4
2.1 Target Audience	4
2.2 Set of Requirements	5
2.3. Use Case Scenarios	6
3. Functional Requirements Specification	6
3.1 Stakeholders	6
3.2 Actors and Goals	7
3.3. Use Cases	7
5. Nonfunctional Requirements	12
6. Domain Analysis	12
6.1 Concept Definitions.....	12
6.2 Association Definitions	13
6.3 Attribute Definitions	13
7. System Operation Contracts	14
7.1 Adding a New Book.....	14
7.2 Updating Book Information	14

8.	User Interface Design	16
9.	Preliminary Design	18
10.	Plan of Work	19
11.	References	20

1. Introduction

The Amazon Bookstore application is a powerful tool for bookstore managers seeking to efficiently manage book inventory. This comprehensive report sheds light on the application's design, functionality, and its exclusive role in simplifying the daily tasks of bookstore managers. In the dynamic world of retail, bookstore managers face numerous challenges, from managing diverse book inventories to ensuring seamless sales operations. The Amazon Bookstore application has been tailored to cater exclusively to their needs, making it a vital tool for the effective management of bookstores.

This application equips bookstore managers with the power to maintain an up-to-date inventory with ease, facilitating the addition of new books and updates to existing ones. The intuitive user interface is designed to enhance the manager's experience, simplifying operations and streamlining the sales process. Through this application, managers can efficiently track and record sales transactions while maintaining accurate inventory data.

2. Customer Statement of Requirements

2.1 Target Audience

The primary user of the Amazon Bookstore application is the bookstore manager. This application empowers them with the capability to manage the entire inventory efficiently, keeping book data accurate, and streamlining sales processes.

2.2 Set of Requirements

As per the manager's requirements, the Amazon Bookstore application offers key features tailored to their daily operations, including:

- **Addition of Books:** The manager can add new books to the inventory, filling in essential details such as title, author, price, and quantity. This feature ensures that the inventory remains up to date.
- **Updating Book Information:** With this functionality, the manager can make changes to book information, including price and quantity, ensuring that inventory data is always accurate.
- **Selling Books:** The manager can facilitate book sales efficiently, reducing stock with each sale.
- **Listing Available Books:** The application offers a readily accessible list of available books, making it easier for the manager to monitor inventory.

2.3. Use Case Scenarios

The Amazon Bookstore application revolves around the needs of the bookstore manager:

- I. Adding a New Book: In this scenario, the manager adds a new book to the inventory, ensuring that all essential book details are recorded accurately.
- II. Updating Book Information: For existing books, the manager can easily update price and quantity, allowing for the constant maintenance of accurate inventory data.
- III. Selling a Book: When sales transactions occur, the manager can efficiently record these sales and reduce the stock, which is essential for effective inventory management.
- IV. Listing Available Books: The manager can readily access a list of books, making it easier to make informed decisions about inventory orders and sales strategies.

3. Functional Requirements Specification

3.1 Stakeholders

The primary stakeholder in the Amazon Bookstore application is the bookstore manager. Their role is central to the functioning of the application. They manage the entire process, from book additions to sales tracking.

3.2 Actors and Goals

- Admin: Admin is responsible for managing all aspects of the application. Their primary goal is to streamline inventory management and sales processes effectively.
- Customer: They use the system to purchase the books, submit reviews, rate products or services

3.3. Use Cases

Use Case 1: Adding a New Book

- ❖ Casual Description: The store manager adds a new book to the inventory, ensuring that all essential book details are accurately recorded, maintaining the uptodate inventory.
- ❖ Fully Description: The manager inputs the title, author, price, and quantity for the new book, and the application adds it to the store's inventory.

Use Case 2: Updating Book Information

- ❖ Casual Description: The manager can adjust the price and quantity of existing books to maintain accurate inventory data.

- ❖ Fully Description: The manager selects an existing book, updates the price and quantity, and the system updates the database accordingly.

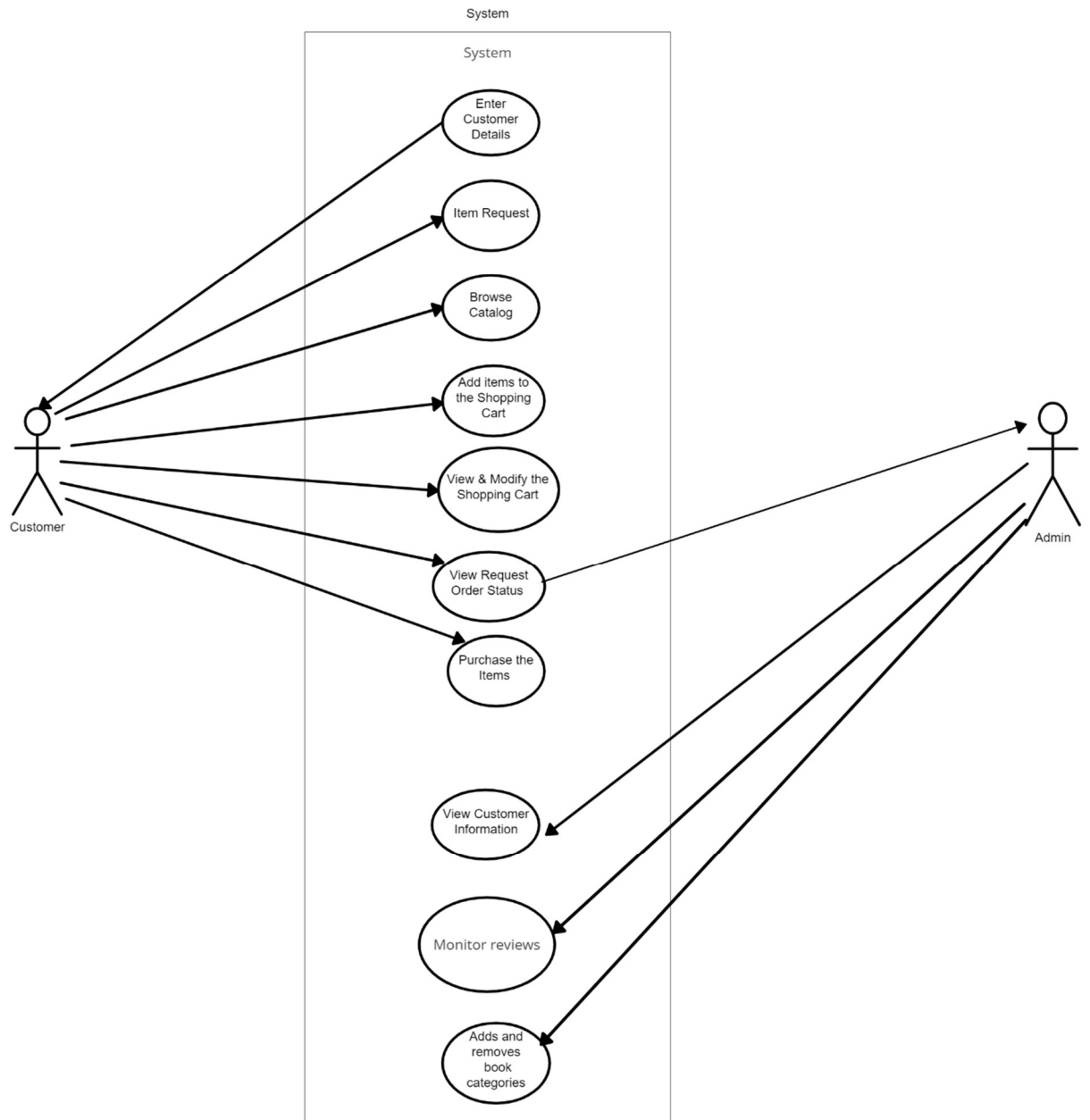
Use Case 3: Selling a Book

- ❖ Casual Description: Sales transactions are easily recorded, and the manager can efficiently reduce stock with each sale.
- ❖ Fully Description: The manager records sales by selecting the book, specifying the quantity sold, and the application reduces stock accordingly.

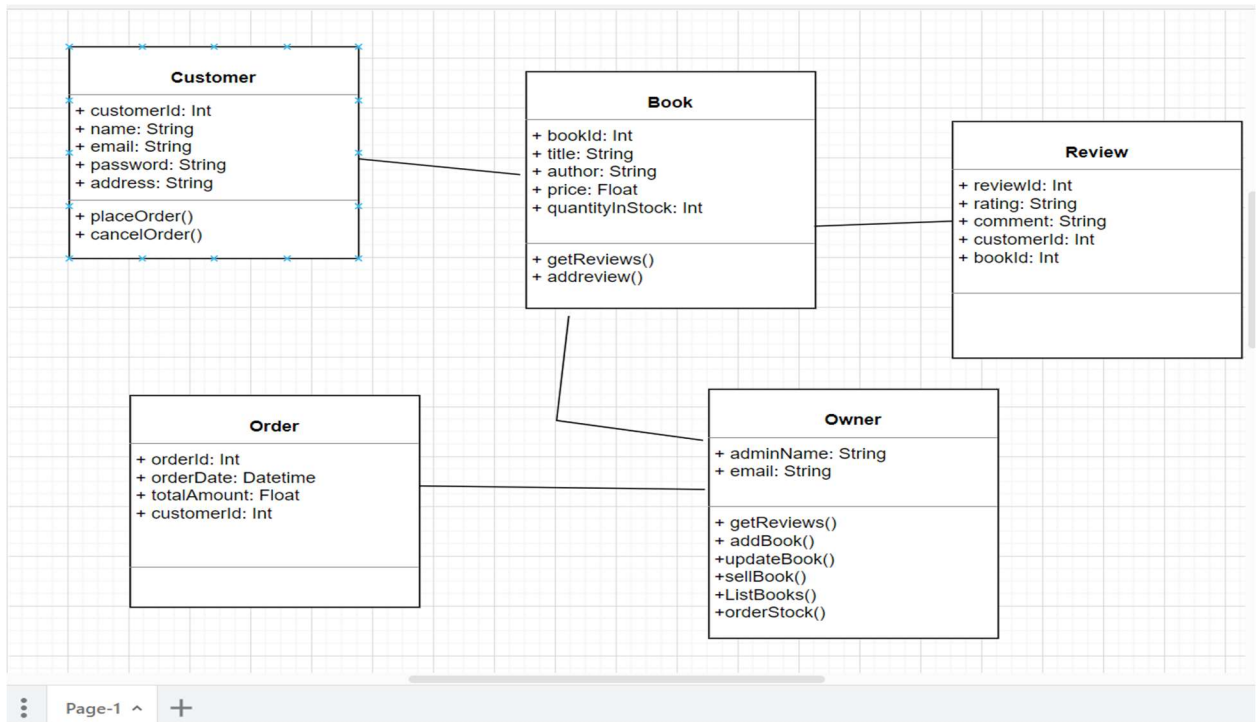
Use Case 4: Listing Available Books

- ❖ Casual Description: The manager can readily access a list of books, aiding in inventory decisions and sales strategies.
- ❖ Fully Description: The manager accesses a comprehensive list of available books, facilitating datadriven decisions.

UseCase Diagram:



Class Diagram:



Class Descriptions:

Customer:

Attributes: customerId, name, email, password, address

Methods: placeOrder(Book book, int quantity), cancelOrder(Order order),
writeReview(Review review)

Book:

Attributes: bookId, title, author, price, quantityInStock

Methods: getReviews(), addReview(Review review)

Review:

Attributes: reviewId, rating, comment, customerId, bookId

Note: customerId and bookId are foreign keys referencing Customer and Book classes, respectively.

Order:

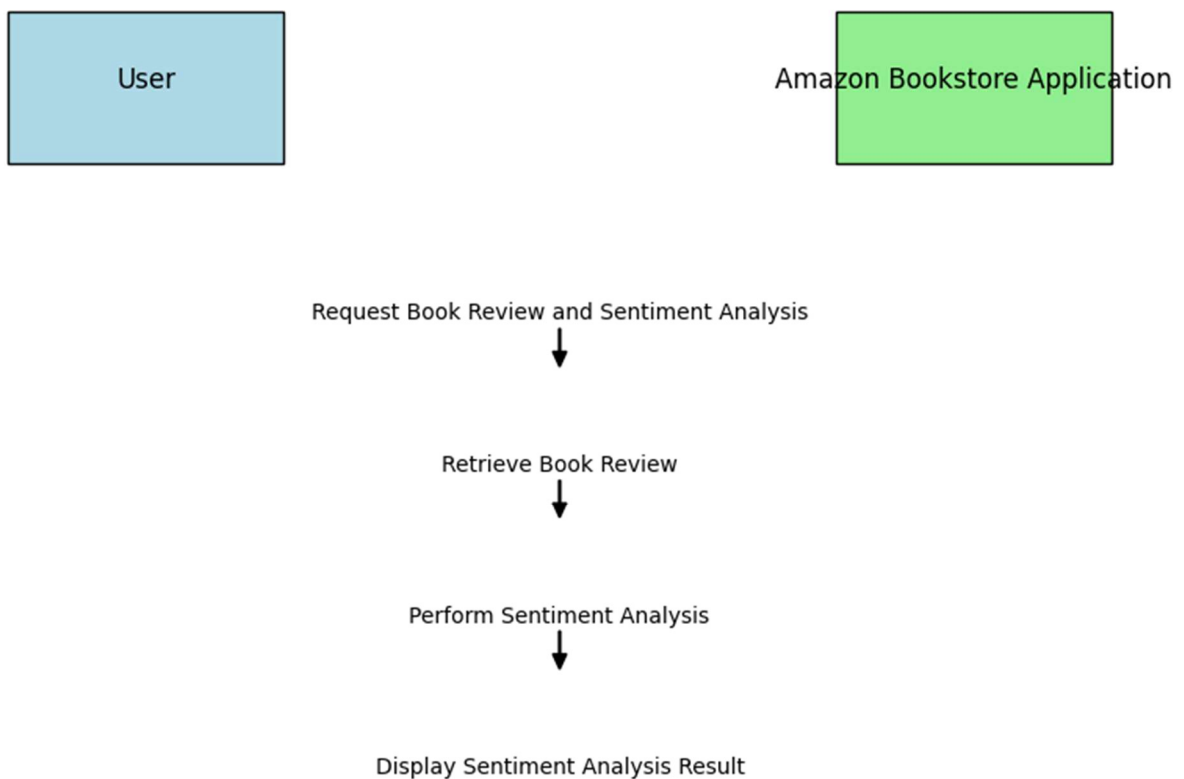
Attributes: orderId, orderDate, totalAmount, customerId

Note: customerId is a foreign key referencing the Customer class.

4. System Sequence Diagrams

These diagrams illustrate the interactions between the bookstore manager and the system in the processes of adding, updating, selling, and listing books.

Book Review and Sentiment Analysis - System Sequence Diagram



5. Nonfunctional Requirements

The Amazon Bookstore application is committed to delivering an intuitive and user-friendly interface that allows managers to navigate effortlessly. Robust security measures ensure the confidentiality of inventory data and customer information. Additionally, efficient performance is a key nonfunctional requirement, ensuring that book data can be managed without delays or system hiccups. These requirements collectively empower managers to oversee their bookstores with confidence, knowing that the application is designed to streamline operations while safeguarding sensitive information and providing a responsive and reliable user experience.

6. Domain Analysis

6.1 Concept Definitions

- I. Book: Represents a specific book in the inventory. It includes attributes such as Title, Author, Price, and Quantity.

- II. Inventory: Represents the entire collection of books available for sale in the bookstore. It has associations with Book entities.
- III. Sales Transaction: Represents a record of each book sale, including details such as the Book sold, the Quantity sold, and the Date of the transaction.

6.2 Association Definitions

- ❖ Inventory-Book: Describes the association between the Inventory and individual Book entities. Each Inventory contains multiple Books, and each Book belongs to a single Inventory.
- ❖ Sales Transaction-Book: Represents the relationship between Sales Transactions and Books, indicating which Book was sold in each transaction.

6.3 Attribute Definitions

- ❖ Book Attributes: Include Title (string), Author (string), Price (double), and Quantity (integer) to describe each book's characteristics.
- ❖ Sales Transaction Attributes: Include Date (date), Book Sold (Book entity), and Quantity Sold (integer) to record sales details.

7. System Operation Contracts

Operation contracts detail the preconditions, postconditions, and sequence of steps for each operation, focusing on the manager's use cases.

7.1 Adding a New Book

Preconditions: The manager is logged into the system. The book's title, author, price, and quantity are provided.

Steps:

1. The manager enters the book details.
2. The system validates the input.
3. The system adds the book to the inventory.

Postconditions: The new book is included in the inventory.

7.2 Updating Book Information

Preconditions: The manager is logged in, selects an existing book, and provides new price and quantity.

Steps:

1. The manager selects a book.
2. The manager updates the price and quantity.
3. The system validates the input.

4. The system updates the book information.

Postconditions: The book's information is updated in the inventory.

7.3 Selling a Book

Preconditions: The manager is logged in, selects a book, specifies the quantity sold.

Steps:

1. The manager selects a book.
2. The manager enters the quantity sold.
3. The system validates the input.
4. The system records the sales transaction.
5. The system updates the inventory to reflect the sale.

Postconditions: The sale is recorded, and the inventory is adjusted accordingly.

7.4 Listing Available Books

Preconditions: The manager is logged into the system.

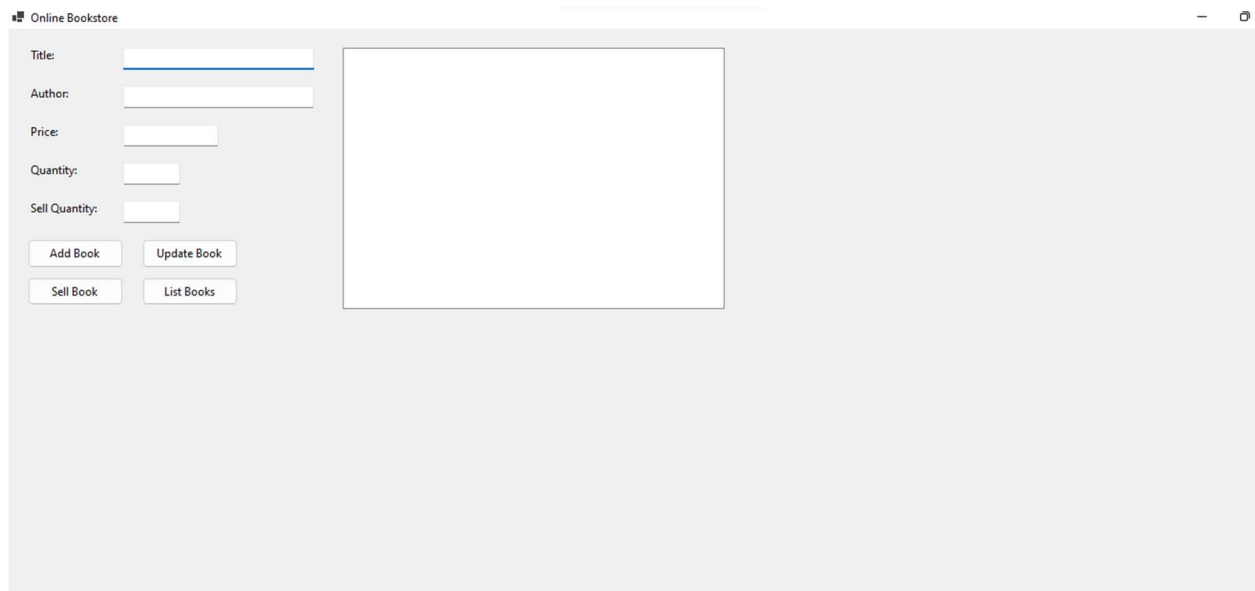
Steps:

1. The manager requests a list of available books.
2. The system retrieves the list from the inventory.

Postconditions: A list of available books is displayed for the manager.

These operation contracts ensure that the manager can effectively execute critical tasks while maintaining the integrity of the inventory and sales transactions without the need for complex mathematical models.

8. User Interface Design

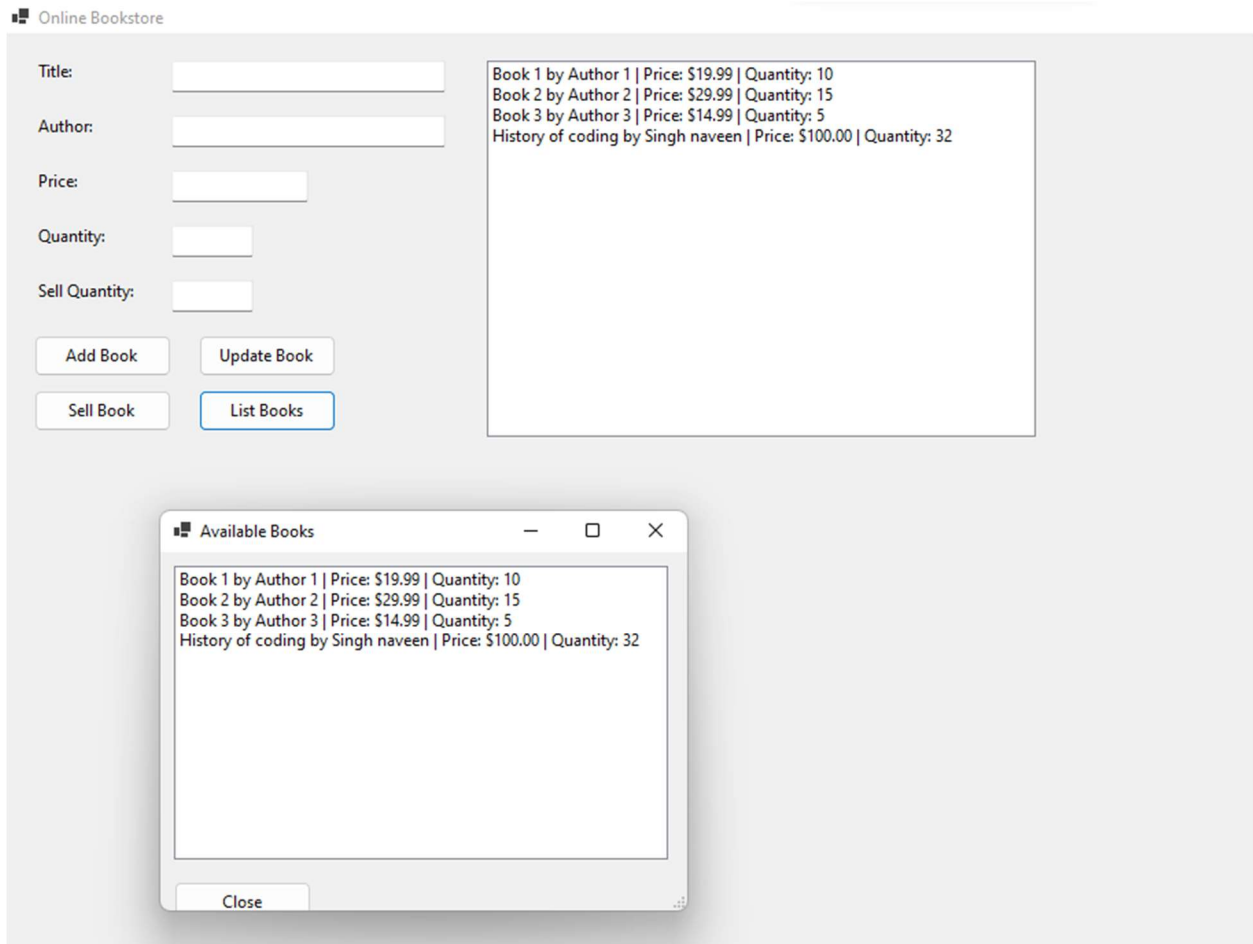


The image shows a user interface for an "Online Bookstore". The interface is divided into a left sidebar and a main content area. The sidebar contains input fields for "Title:", "Author:", "Price:", "Quantity:", and "Sell Quantity:", each followed by a text input box. Below these fields are four buttons: "Add Book", "Update Book", "Sell Book", and "List Books". The main content area is a large, empty rectangular box, likely intended for displaying book details or a list of books. The top of the interface has a title bar with "Online Bookstore" on the left and standard window controls (minimize, maximize, close) on the right.

Available Books

Book 1 by Author 1 | Price: \$19.99 | Quantity: 10
Book 2 by Author 2 | Price: \$29.99 | Quantity: 15
Book 3 by Author 3 | Price: \$14.99 | Quantity: 5

Close



9. Preliminary Design

The preliminary design focuses on creating an intuitive and user-friendly interface for bookstore managers. It aims to streamline the manager's tasks by providing easy navigation and clear information presentation. The design ensures that managers can efficiently add and update

books, record sales, and access a list of available books. With a user-centric approach, the design prioritizes simplicity and functionality, enhancing the overall user experience.

10. Plan of Work

In the coming stages of the Amazon Bookstore application development, our team is committed to enhancing the software to better serve the needs of bookstore managers. The following plan outlines our key objectives and tasks:

- i. **Enhanced User Interface:** We will focus on refining the user interface to ensure an even more intuitive and efficient experience for managers. This includes improving the layout, navigation, and overall user-friendliness.
- ii. **Sales Reporting:** We will introduce a sales reporting feature that will enable managers to track and analyze sales data. This will aid in decision-making and inventory management.
- iii. **Performance Optimization:** We will work on optimizing the application's performance to ensure swift responses and efficient book data management.
- iv. **Security Enhancements:** Our team will strengthen security measures to safeguard sensitive inventory and sales data, prioritizing data integrity and customer privacy.

- v. Testing and Quality Assurance: Rigorous testing will be conducted to identify and rectify any potential issues, ensuring a robust and reliable application.
- vi. Documentation: Comprehensive documentation will be maintained and updated, providing support for users and future development.
- vii. Training and Support: We will offer training sessions to managers and provide ongoing support to address any queries or issues.

Our team remains dedicated to tailoring the Amazon Bookstore application to the unique needs of bookstore managers, ensuring that it continues to be a valuable tool for their daily operations.

11. References

- [1] Zhang, X., Zhang, L., Yang, L., Chen, Y., & Li, P. (2022). Design of an Online Book Purchasing Platform with. Net and SQL Server. *International Core Journal of Engineering*, 8(8), 997-1001.
- [2] Echchaoui, H., & Boudour, R. (2023, August). Check for updates Online Book Selling Dapp Based on Smart Contracts. In *Artificial Intelligence Doctoral Symposium: 5th Doctoral Symposium, AID 2022, Algiers, Algeria, September 18–19, 2022, Revised Selected Papers* (p. 178). Springer Nature.

- [3] Ferrone, H. (2021). *Learning C# by Developing Games with Unity 2021: Kickstart your C# programming and Unity journey by building 3D games from scratch*. Packt Publishing Ltd.
- [4] Echchaoui, H., & Boudour, R. (2022, September). Online Book Selling Dapp Based on Smart Contracts. In *Artificial Intelligence Doctoral Symposium* (pp. 178-191). Singapore: Springer Nature Singapore.
- [5] Price, M. J. (2019). *C# 8.0 and .NET Core 3.0—Modern Cross-Platform Development: Build applications with C#, .NET Core, Entity Framework Core, ASP.NET Core, and ML.NET using Visual Studio Code*. Packt Publishing Ltd.