

# FULL STACK DEVELOPMENT WITH MERN

## INTRODUCTION:

**Title:** Book Store

**Team Members:**

- 1. Surya I
- 2. Kumaran M
- 3. Vignesh M
- 4. Sadha Priyan

## PROJECT OVERVIEW:

**Purpose:**

BookStore is your one-stop destination for exploring a vast collection of books across genres. From personalized profiles to seamless browsing, purchasing, and order management, enjoy an effortless e-bookstore experience.

**Features:**

- 1. **Extensive Book Collection**  
Explore a diverse range of books across genres, including fiction, non-fiction, academic, and more. New arrivals and bestsellers are just a click away.
- 2. **Personalized Profiles**  
Create and manage your profile to customize your reading experience. Update personal information, preferences, and access your order history effortlessly.
- 3. **Easy Book Browsing**  
Navigate through intuitive categories and filters to discover your next favorite book. View book details such as title, author, genre, and price with ease.
- 4. **Convenient Book Purchase**  
Add books to your cart, specify quantities, and choose formats like e-books or special editions. A seamless checkout process ensures a hassle-free purchase.
- 5. **Order Management**  
Access current and past orders at any time. Check order status, payment details, and manage your purchase history directly within the app.
- 6. **Order Confirmation & Notifications**  
Get instant confirmation for new orders with detailed summaries. Stay updated with real-time notifications

## Architecture

### 1. Frontend: React.js

The frontend is developed using React.js to ensure a dynamic, interactive, and responsive user experience. The interface is divided into three main segments, each designed to cater to specific user roles:

#### a. Buyers

- i. **Signup & Login Pages:** Users can create accounts or log in to access personalized features
- ii. **Add-to-Cart Page:** Buyers can select books and add them to a cart for seamless purchasing.
- iii. **Browsing Features:** Buyers can filter books by genre, author, price, and ratings.

#### b. Sellers

- i. **Signup & Login Pages:** Sellers can register or log in to manage their profiles.
- ii. **Book Management:** Sellers have permission to add, edit, or delete books they list for sale.
- iii. **Order Insights:** Sellers can view and manage orders placed by buyers for their books.

#### c. Admin

1. **Admin-Specific Functionalities:** Admins can oversee user and seller activities, manage book listings globally, and monitor overall platform activity.
2. **Dashboard:** A comprehensive admin panel to view and control all segments.

Each segment is modular, ensuring scalability and maintainability. React's component-based architecture is leveraged for reusability across the app.

### 2. Backend: Node.js and Express.js

- a. The backend is built using **Node.js** and **Express.js**, enabling efficient handling of API requests

and server-side logic. Key backend features include:

- b. **RESTful API Architecture:** Ensures smooth communication between the frontend and the database.
- c. **Role-Based Access Control:** Implements distinct permissions for buyers, sellers, and admins.
  - i. **Buyers:** Access profile, browse books, place orders, and view order history.
  - ii. **Sellers:** Manage books and view orders placed for their inventory.
  - iii. **Admins:** Full control over user and seller data, as well as system-wide management.
- d. **Error Handling & Validation:** Ensures data integrity and user input validation for all API calls.
- e. **Middleware:** Custom middleware for authentication (e.g., JWT), logging, and request validation.

### 3. Database: MongoDB

MongoDB serves as the database for managing the app's data due to its scalability and flexibility. Key database design elements include:

- i. **User Collection:** Stores details for buyers, sellers, and admins with role-based fields.
- ii. **Book Collection:** Manages book data, including title, author, genre, price, quantity, ratings, and seller details.
- iii. **Order Collection:** Tracks order information, including buyer details, purchased books, order status, and payment history.
- iv. **Session Management:** Stores user sessions for security and persistence.

#### SETUP INSTRUCTION:

##### Prerequisites

- **Software Dependencies:**

- Node.js: v14+
- MongoDB: v4.4+
- npm: v7+
- React: v18+

## Installation Steps

### Clone the Repository:

bash

Copy code

```
git clone https://github.com/username/BookStore.git
cd BookEase
```

### 1. Install Dependencies:

Navigate to the frontend (`client`) and backend (`server`) directories and install the necessary packages:

bash

Copy code

```
cd Frontend
npm install
cd ../Backend
npm install
```

### 2. Set Up Environment Variables:

Create a `.env` file in the `server` directory with the following variables:

makefile

Copy code

```
PORT=5000
MONGO_URI=mongodb+srv://<username>:<password>@cluster.mongodb.net/BookEase
```

## TESTING STRATEGY:

### 1. Unit

### Testing

Verify individual components like React UI elements, API endpoints, and database queries to ensure they function as expected.

### 2. Integration

### Testing

Test interactions between the frontend, backend, and database to ensure smooth

communication and data flow.

### 3. End-to-End

Testing

Simulate real-world user journeys, such as browsing books, adding them to the cart, and completing a purchase.

### 4. Performance

Testing

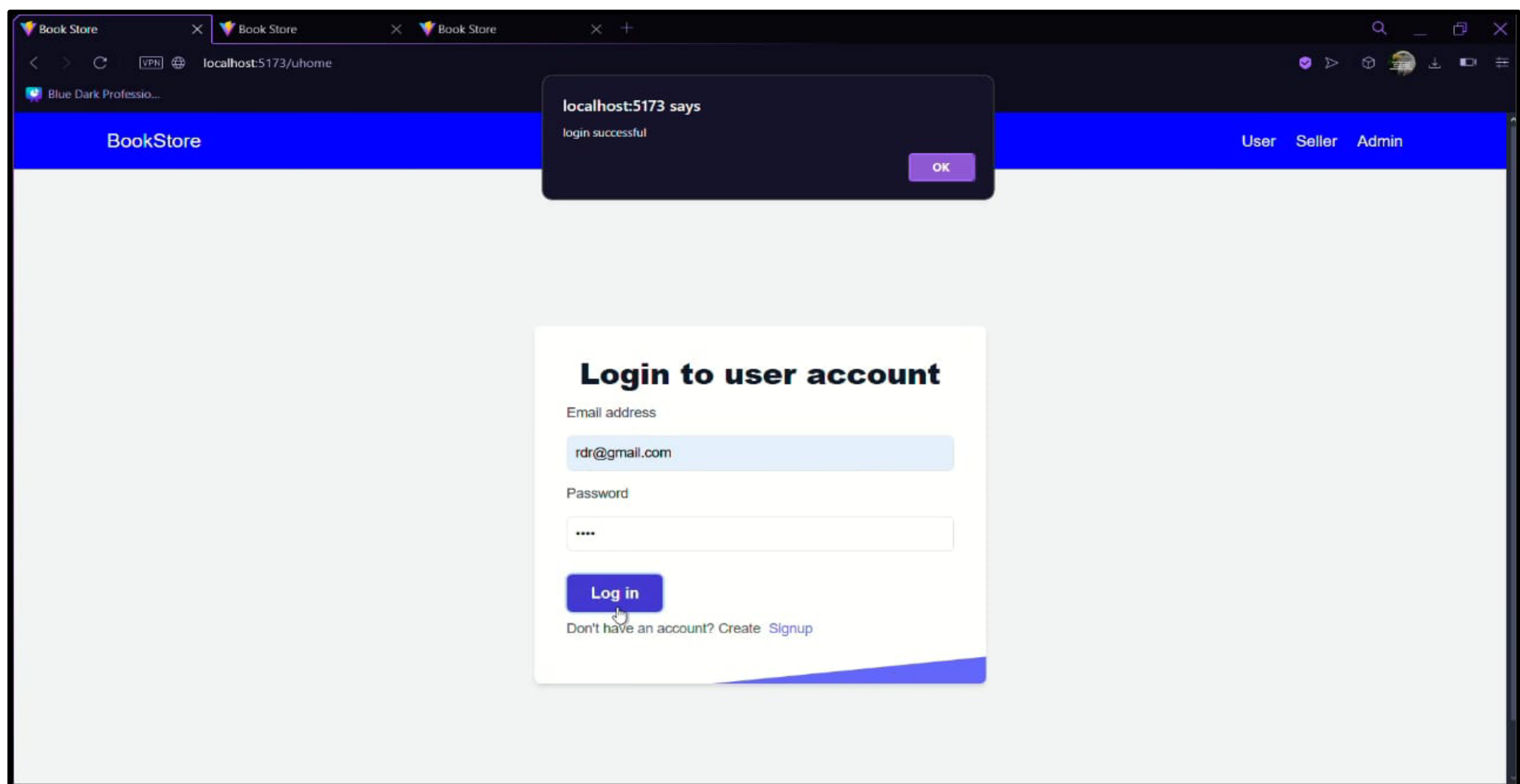
Assess application stability and responsiveness under varying load conditions to ensure optimal performance.

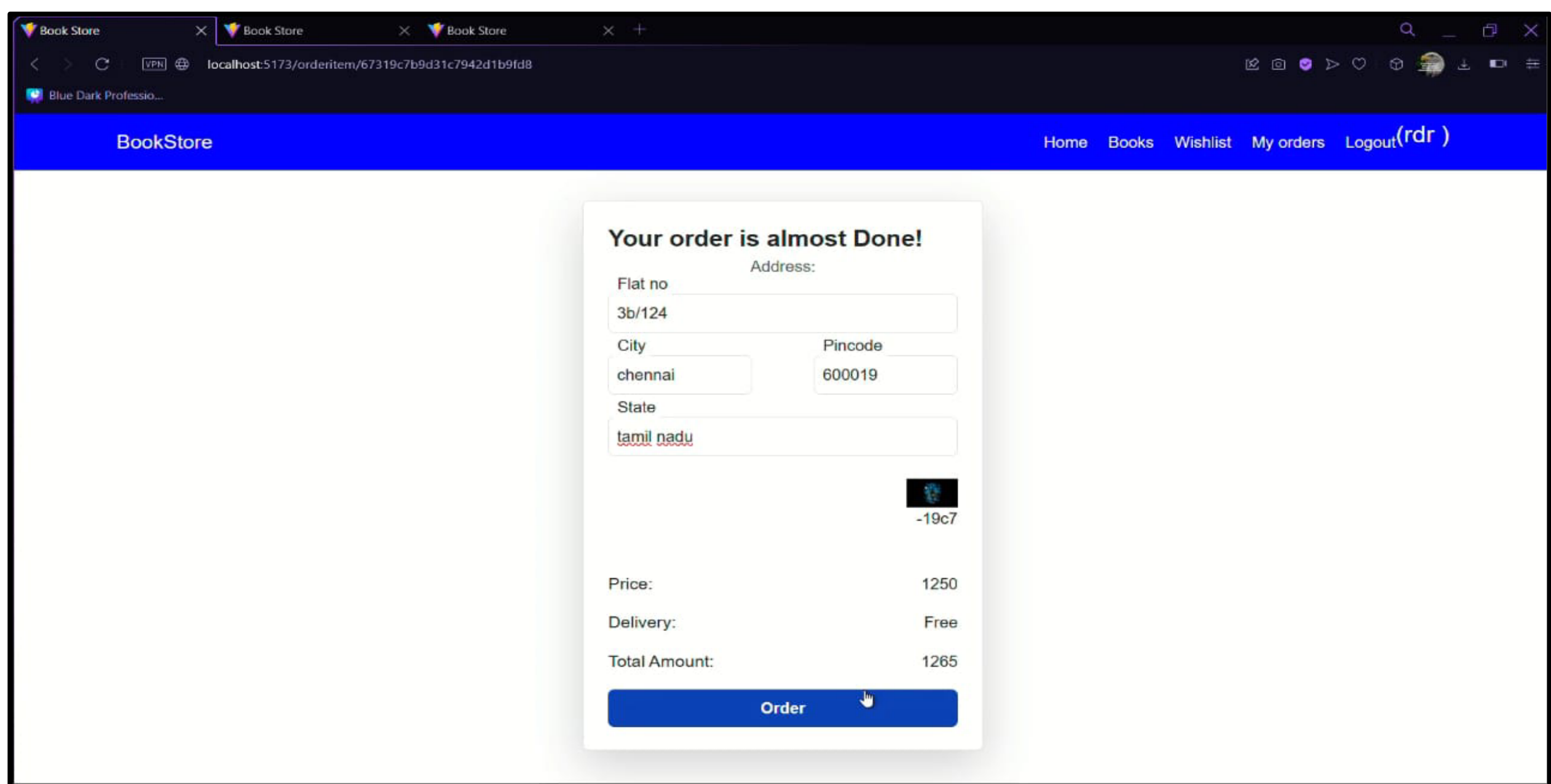
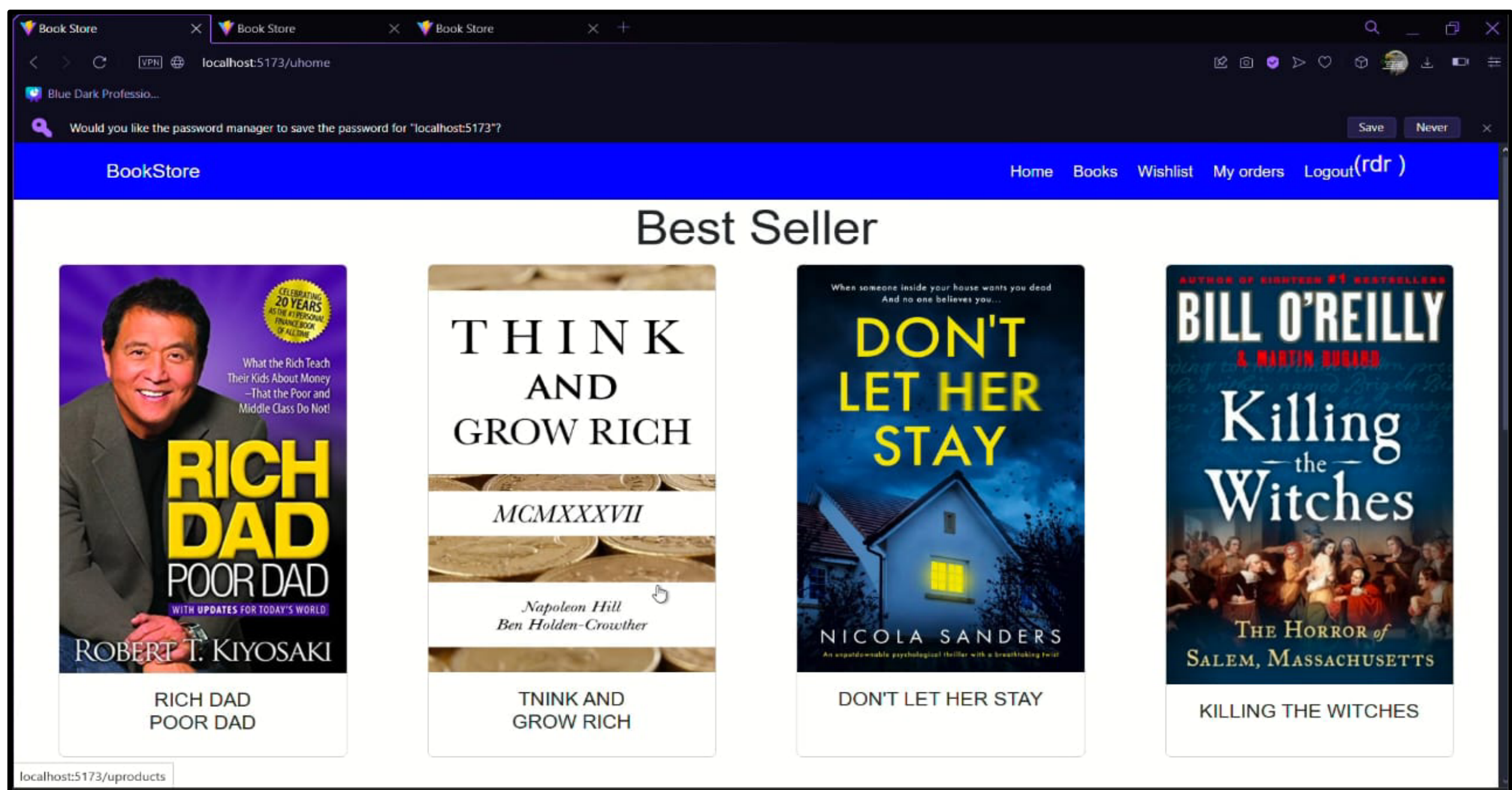
### 5. Security

Testing

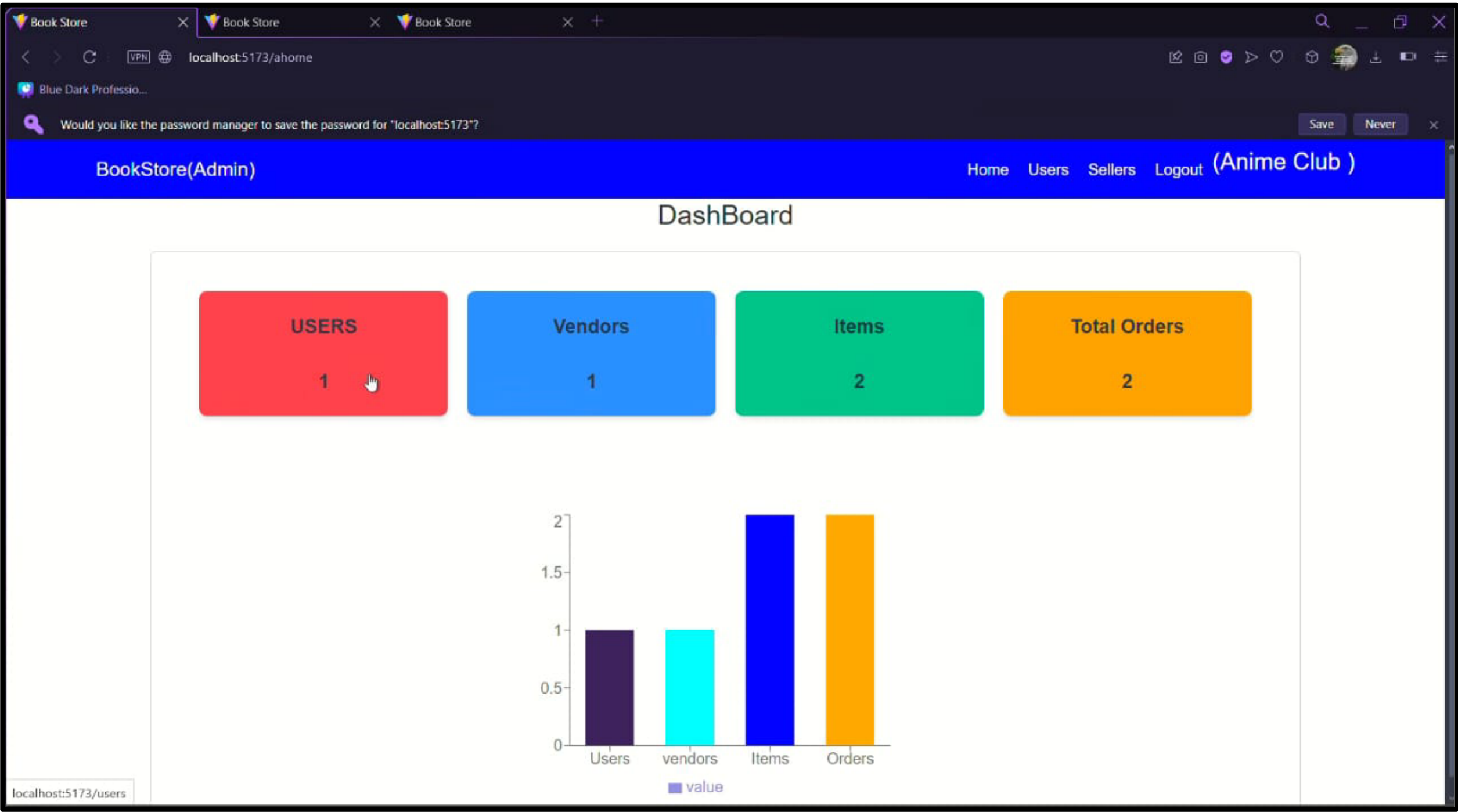
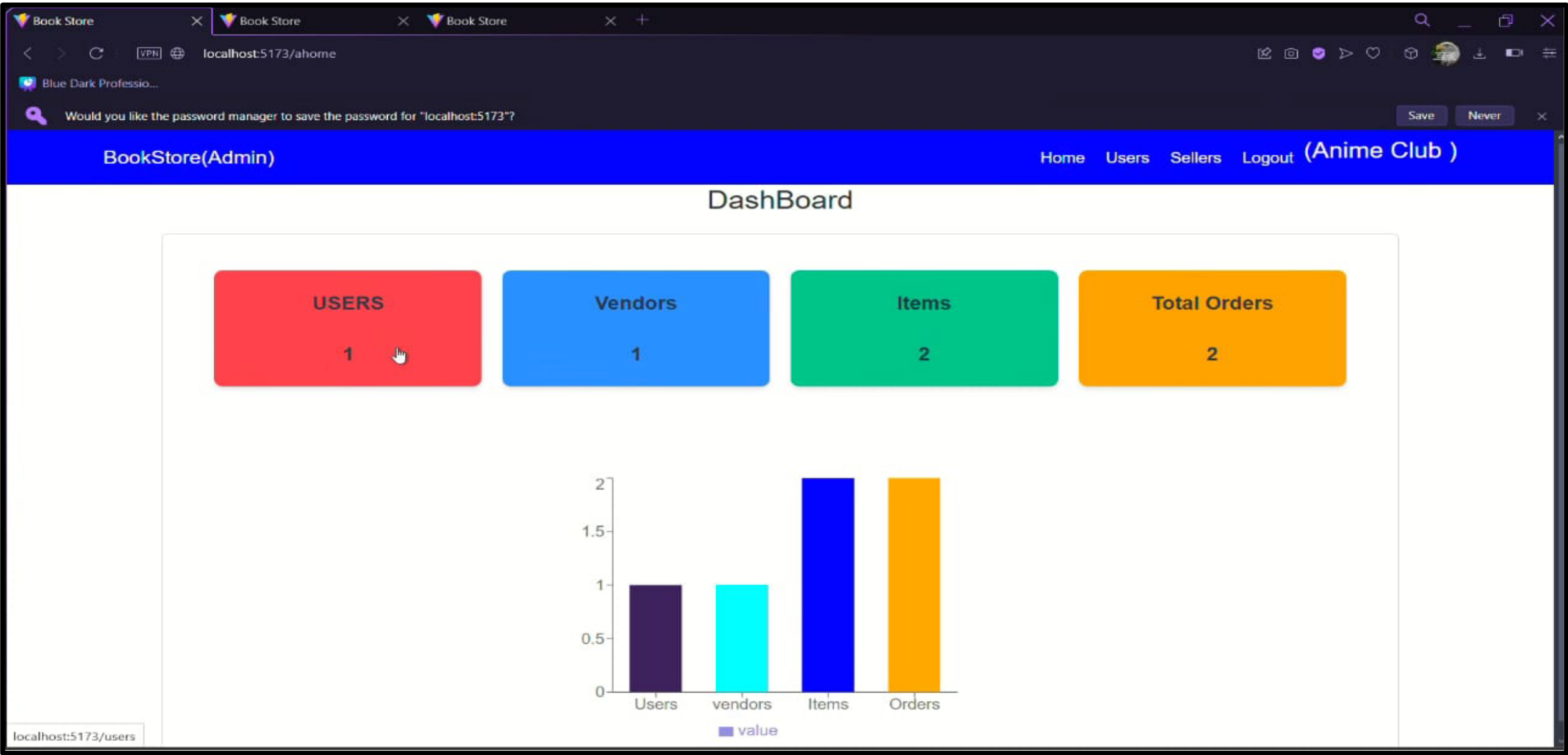
Check for vulnerabilities like unauthorized access and ensure sensitive data is encrypted and protected.

## SCREENSHOTS:









Book Store

Book Store

Book Store

+


localhost:5173/orders

Blue Dark Professio...


Would you like the password manager to save the password for "localhost:5173"? Save Never

BookStore(Seller) Home Myproducts Add Books Orders Logout(rdr)

Orders



ProductName:	Orderid:	Customer Name	Address:	BookingDate	Delivery By	Warranty	Price	Status
	67319ae19d	Anime	1/1sn2wrm, tuti,(628008), tamil nadu.	11/11/2024	11/18/2024	1 year	1099	ontheway



ProductName:	Orderid:	Customer Name	Address:	BookingDate	Delivery By	Warranty	Price	Status
	67319ceb9d	rdr	3b/124, chennai,(600019), tamil nadu.	11/11/2024	11/18/2024	1 year	1349	ontheway

localhost:5173/orders

Book Store

Book Store

Book Store


+

localhost:5173/myorders

Blue Dark Professio...

BookStore Home Books Wishlist My orders Logout(rdr)

My Orders



ProducttName:	Orderid:	Address:	Seller	BookingDate	Delivery By	Price	Status
-19ce	67319ceb9d	3b/124, chennai,(600019), tamil nadu.	rdr	11/11/2024	11/18/2024	\$1349	ontheway

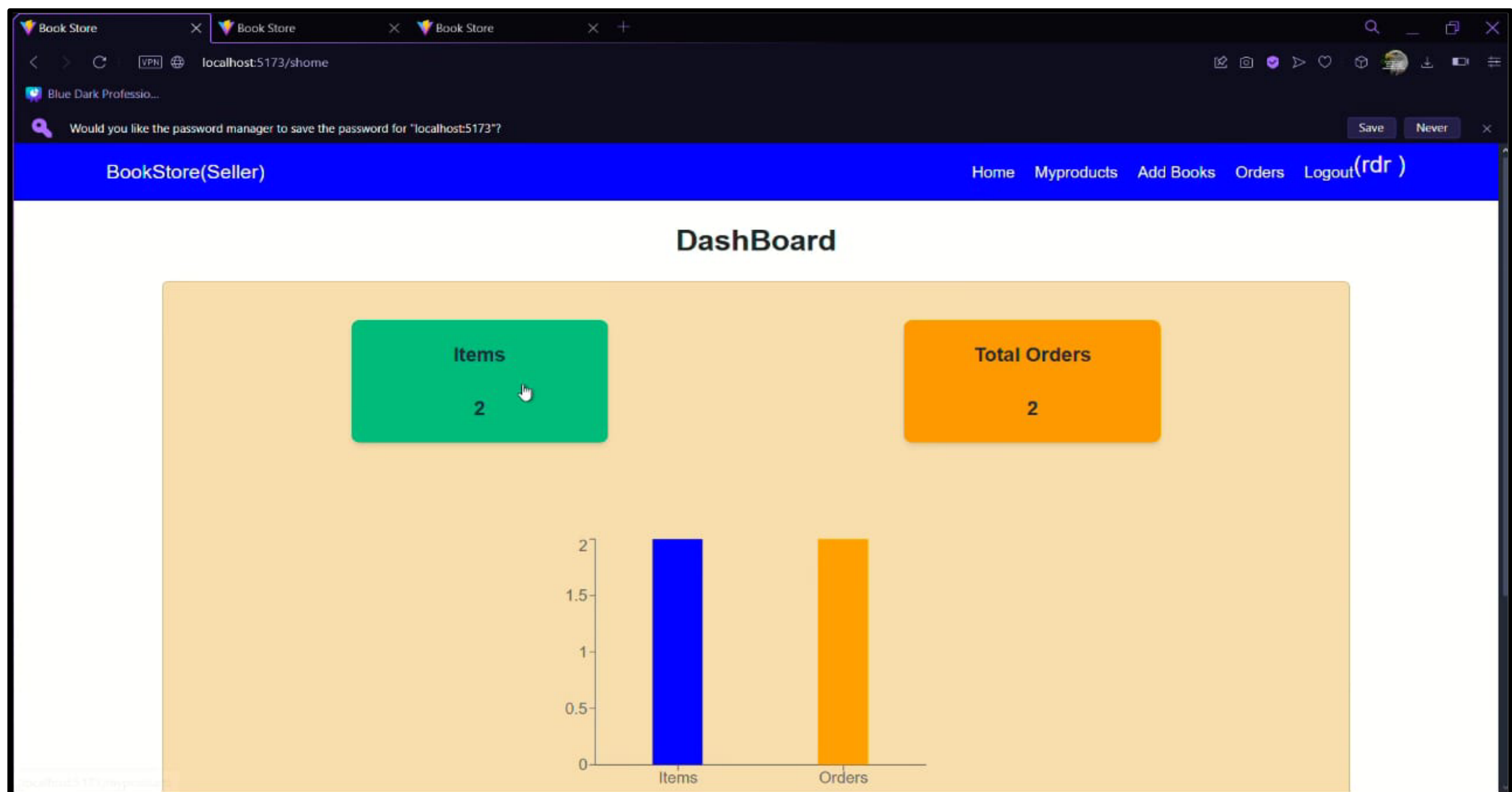
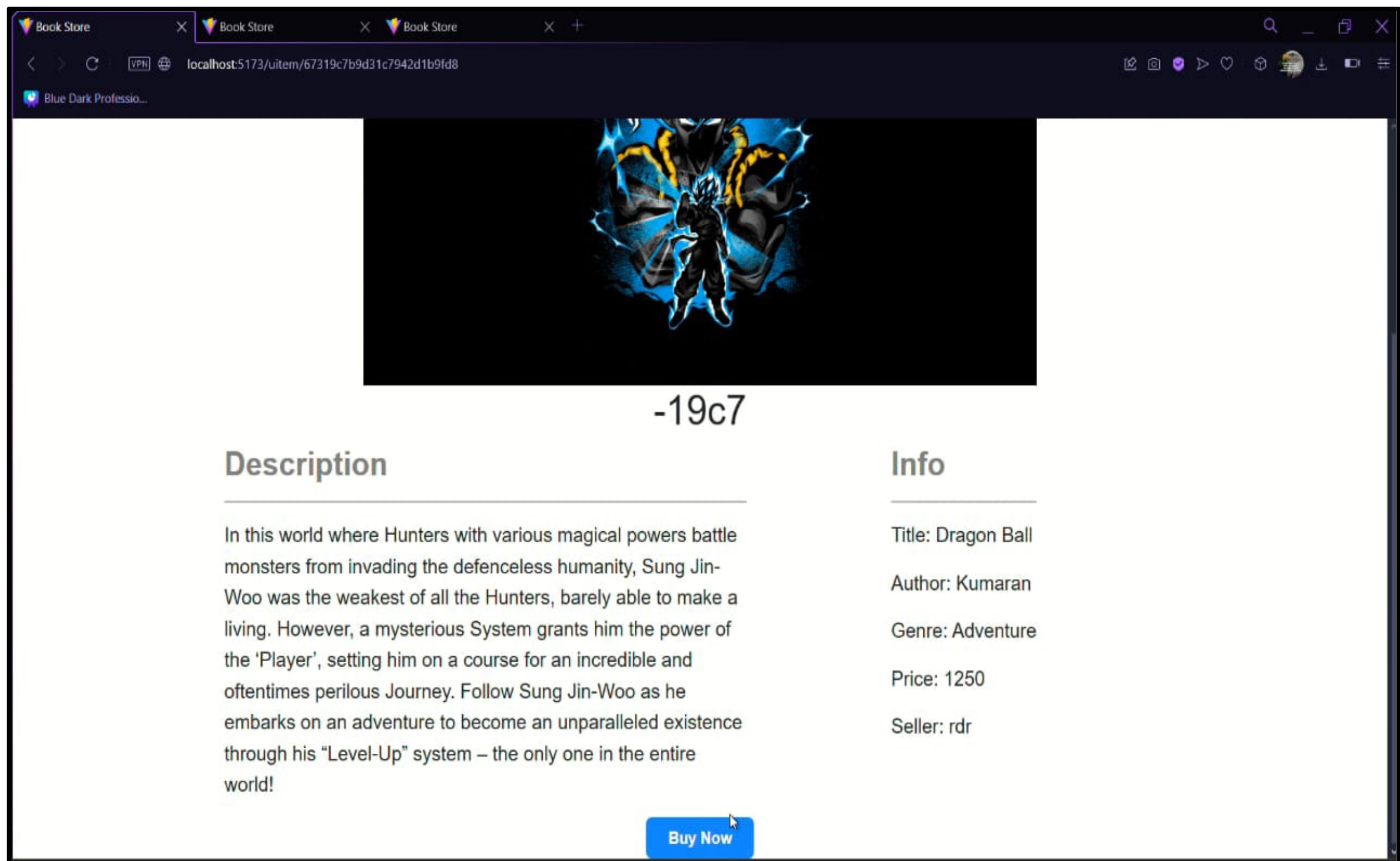
Contact us

"Embark on a literary journey with our book haven – where every page turns into an adventure!"

Call At: 127-865-586-67  
Copyright © 2024 By BookStore.  
All Rights Reserved.

localhost:5173/myorders





DEMO VIDEO LINK: [5\\_6165722159554499644.mp4](#)

FUTURE ENHANCEMENT:

1.	AI-Powered	Recommendations
	Implement a recommendation engine to suggest books based on user preferences and browsing history.	
2.	Wishlist	Feature
	Allow users to save books to a wishlist for future purchases or easy access.	
3.	Multi-Language	Support
	Expand accessibility by adding support for multiple languages in the app.	
4.	Advanced	Analytics
	Provide sellers and admins with detailed sales insights, user trends, and inventory predictions.	
5.	Mobile	Application
	Develop a mobile version of the app for Android and iOS to enhance user convenience.	