**BOOKNEST**

**(WHERE STORIES NESTLE)**

**ADITYA COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SUBMITTED BY**

|  |  |
| --- | --- |
| MALLELA KOUSHIK | 22MH1A05B8 |
| MATTA GAYATRI | 22MH1A0533 |
| SUDEER KUMAR | 23MH5A0502 |
| JEELAKARRA BALA SATYA KOTESWARA RAO | 23MH5A0508 |

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.NO | CONTENT | PAGE.NO |
| 1 | INTRODUCTION | 1 |
| 2 | TECHNICAL ARCHITECTURE | 2-3 |
| 3 | FEATURES OF BOOKNEST | 4-5 |
| 4 | TECHNOLOGY STACK | 6 |
| 5 | PROJECT STRUCTURE | 7-8 |
| 6 | BACKEND | 8-9 |
| 7 | FRONTEND | 9-10 |
| 8 | PROJECT IMPLEMENTATION | 11-16 |
| 9 | SUMMARY | 17 |

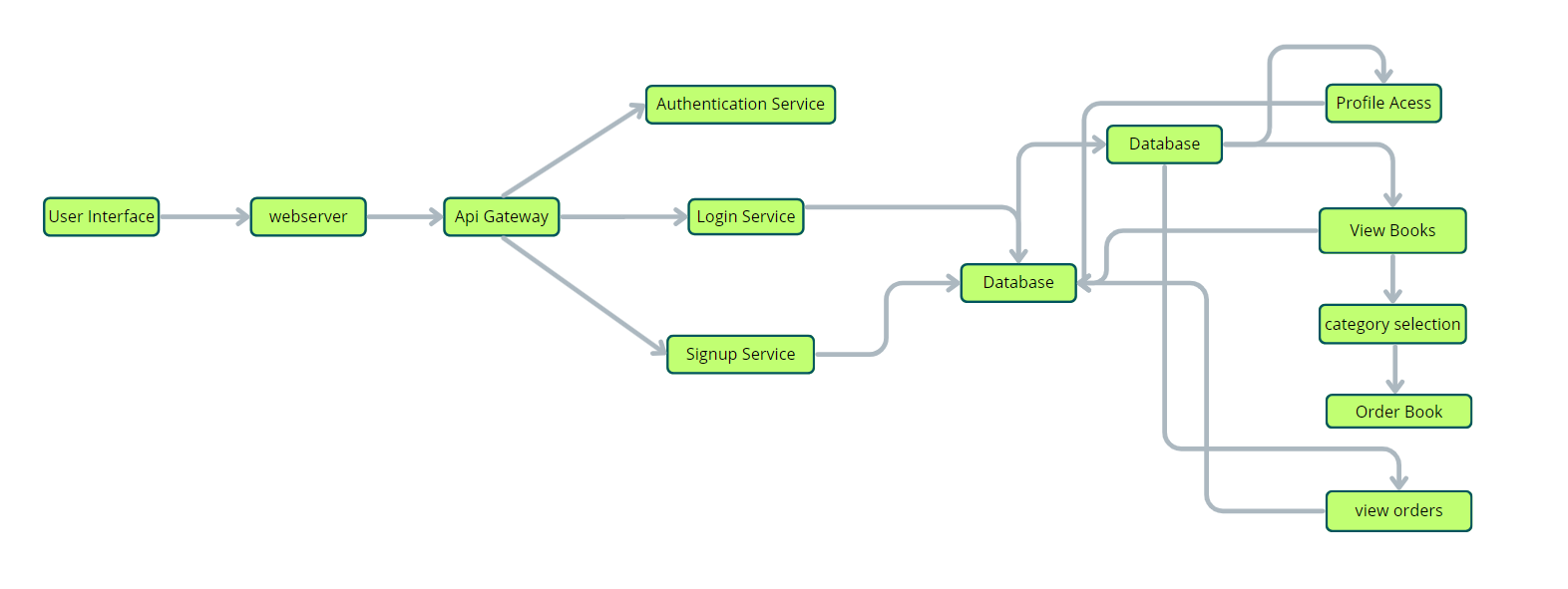
**INTRODUCTION**

Introducing the BookNest, the doorway to an amazing world of books! Our website is made to meet all of your bookish needs, whether you're a voracious reader, a casual browser, or someone looking for the ideal present. It has never been simpler to find your next great read because to the extensive choice of books that span genres, authors, and languages. Read reviews, look through suggested books, and easily buy or reserve books for pickup whenever it's convenient for you. Use the BookNest Website to immerse yourself in the exciting world of books, where each page becomes an adventure!

**Description:**

The BookNest was developed with the MERN Stack, an effective combination of modern technology (MongoDB, Express.js, React, and Node.js). Book enthusiasts now have a whole new way to locate, peruse, and enjoy books from anywhere thanks to this app, which combines the pleasure of reading with the convenience of technology.  
The app is robust and user-friendly, and it was created with today's readers in mind. Our app offers you a seamless and customized experience that suits your reading preferences, whether you're searching for the newest books or your classic favorites.  
MongoDB works in the background to store and arrange a sizable book collection so you can locate what you're looking for fast. React ensures that everything looks fantastic and functions properly on screen, while Express.js and Node.js keep the application operating quickly and seamlessly. Because our software is compatible with desktops, tablets, and phones, you may read at any time and from any location. It's easy to search, browse, and purchase books with a few clicks thanks to the responsive, simple, and clean design.

**TECHNICAL ARCHITECTURE**



**User Interface:**

This is the part of the app that customers see and use. It lets them browse books, search for titles or authors, read book details, and make purchases. It’s designed to be clean, simple, and easy to use so that anyone can explore and shop without confusion.

**Web Server:**

The web server runs the app and shows users the pages they need. It sends and receives information between the app and users, making sure everything loads quickly and works smoothly while browsing or buying books.

**API Gateway:**

When the app gets a request (like searching for a book or placing an order), the API Gateway sends it to the right part of the system. It handles all communication between users and the app’s services.

**Authentication Service:**

This part makes sure users are who they say they are. It handles login, sign-up, and protects user information. It keeps everything safe while users are browsing or making purchases.

**Database:**The database stores all important information like book titles, authors, prices, availability, and user data such as account details and past orders. It helps the app remember everything and find it quickly when needed.

**View Books:**

This feature lets users see the entire collection of books. They can scroll through and find books they like, or discover new ones by simply browsing.

**Category Selection:**

Users can filter books by genres or categories—like fiction, history, or science. This helps them find books that match their interests more easily.

**Inventory Management Service:**

This keeps track of all the books in the store—how many are available, their ratings, and stock status. It updates automatically so users always see the correct info about each book.

**Order Management Service:**

This part handles the entire buying process. Users can add books to their cart, choose how many they want, and place an order. It also updates users about their order status, like when it's confirmed or shipped.

**FEATURES OF BOOKNEST**

**User Registration and Authentication:**

Allow users to register accounts securely, log in, and authenticate their identity to access the book store platform.

**Book Listings:**

Display a comprehensive list of available books with details such as title, author, genre, description, price, and availability status.

**Book Selection:**

Provide users with options to select their preferred books based on factors like genre, author, ratings, and popularity.

**Purchase Process:**

Allow users to add books to their cart, specify quantities, and complete purchases securely. Upon successful completion, an order is generated, and the inventory is updated accordingly.

**Order Confirmation:**

Provide users with a confirmation page or notification containing details of their order, including book information, total price, and order ID.

**Order History:**

Allow users to view their past and current orders, providing options to track shipments, review purchased books, and rate their shopping experience.

**Organizer Dashboard**:

Offer administrators an interface to manage book listings, inventory levels, user accounts, orders, and other platform-related activities.

**Create Item:**

Organizer can create items and add new items and he can get the items and he can update items.

**Admin Dashboard:**

Offer administrators an interface to manage book listings, inventory levels, user accounts, orders, and other platform-related activities. Manage the users and organizers.

**Reporting and Analytics:**

Generate reports and analytics on book sales, popular genres, user demographics, and other relevant metrics to gain insights into platform usage and performance.

**Integration with External APIs:**

Integrate with third-party APIs for services like payment processing, shipping logistics, and book recommendations to enhance the functionality and user experience of the book store platform.

**TECHNOLOGY STACK**

**Frontend:**

1. React
2. Tailwind CSS
3. Bootstrap

**Backend:**

1. Node.js
2. Express.js

**Database:**

1. MongoDB

**Other:**

1. Axios
2. Multer – for image uploading
3. Recharts – for analytics
4. React Router DOM

**PROJECT STRUCTURE**

BOOKNEST/

│

├── Backend/

│ ├── db/

│ │ ├── Seller/

│ │ ├── Users/

│ │ └── config.js

│ ├── node\_modules/

│ ├── uploads/

│ ├── server.js

│ ├── package.json

│ └── package-lock.json

│

├── frontend/

│ ├── node\_modules/

│ ├── public/

│ ├── src/

│ │ ├── Admin/

│ │ ├── Seller/

│ │ ├── User/

│ │ ├── Components/

│ │ ├── App.jsx

│ │ ├── main.jsx

│ │ ├── App.css

│ │ └── index.css

│ ├── index.html

│ ├── vite.config.js

│ ├── package.json

│ └── package-lock.json

**BACKEND**

The key files used in the Backend development are

* **server.js:** Main file to initialize the server, connect MongoDB, and handle routes.
* **config.js:** Holds the MongoDB URI and connection logic.
* **db/Seller & db/Users:** Contains Mongoose schemas and route handlers for user and seller operations.
* **uploads/:** Stores book images uploaded by sellers.

**Technologies Used:**

* **Node.js** – Runtime to run JavaScript on the server
* **Express.js** – Framework for creating REST APIs
* **MongoDB** – NoSQL database
* **Multer** – Middleware for image/file upload

**The backend handles:**

* API endpoints for books, users, and sellers
* File upload (book images)
* Database operations for CRUD
* Authentication and role-based access

**FRONTEND**

**1.Technologies Used:**

* React.js – For building user interfaces
* Vite – Fast build tool for React
* Tailwind CSS / Bootstrap – For responsive design and styling
* Axios – For API requests
* React Router DOM – For routing between pages

**2. Key Folders:**

* src/Admin/ – Components for admin tasks (e.g., manage users, books)
* src/Seller/ – Allows sellers to add, edit, and delete book listings
* src/User/ – Interfaces for browsing books, managing cart, and purchasing
* src/Components/ – Common components like Navbar, Footer, BookCards

**3. Main Files:**

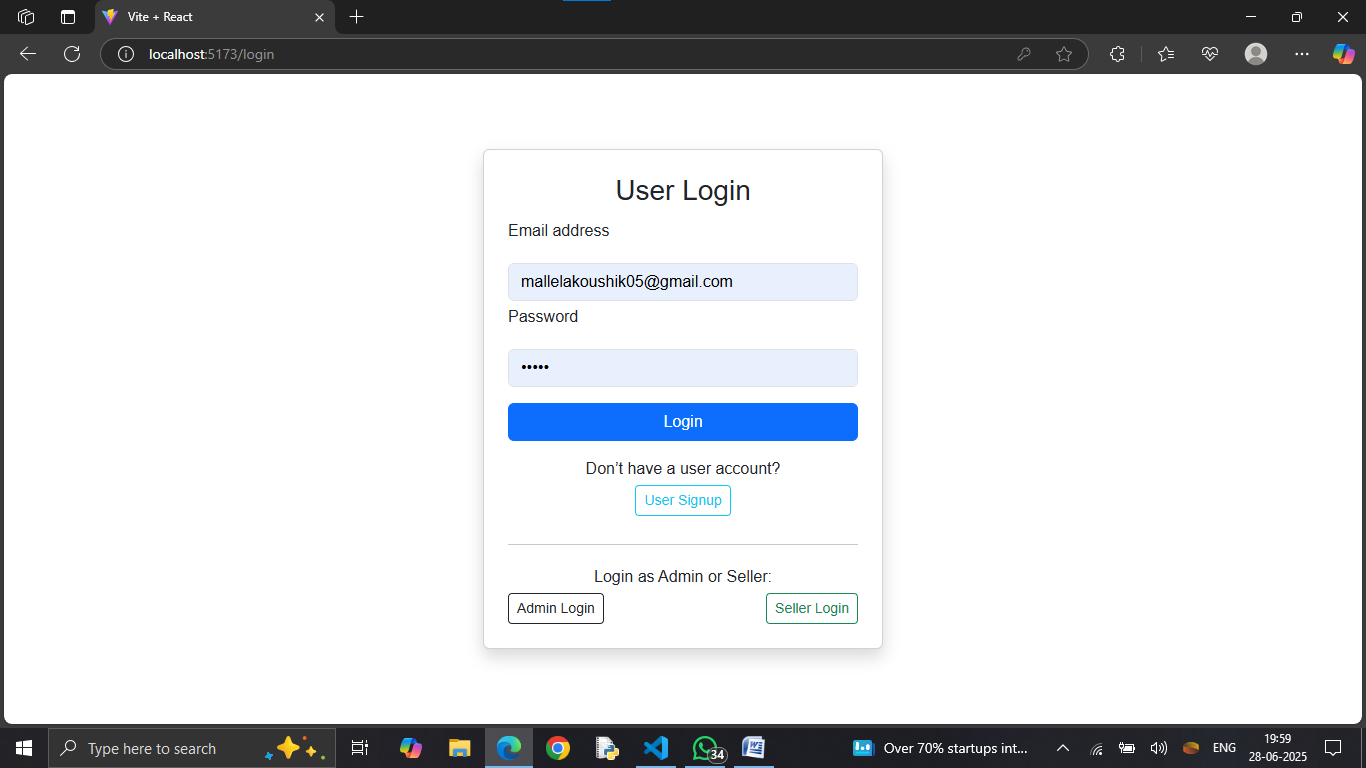
* App.jsx: Main component defining the route structure.
* main.jsx: React entry point that renders<App />into the DOM.
* App.css / index.css: Global and default styling.
* vite.config.js: Configuration for the Vite build process.
* index.html: Template used by Vite to inject React content

**PROJECT IMPLEMENTATION**

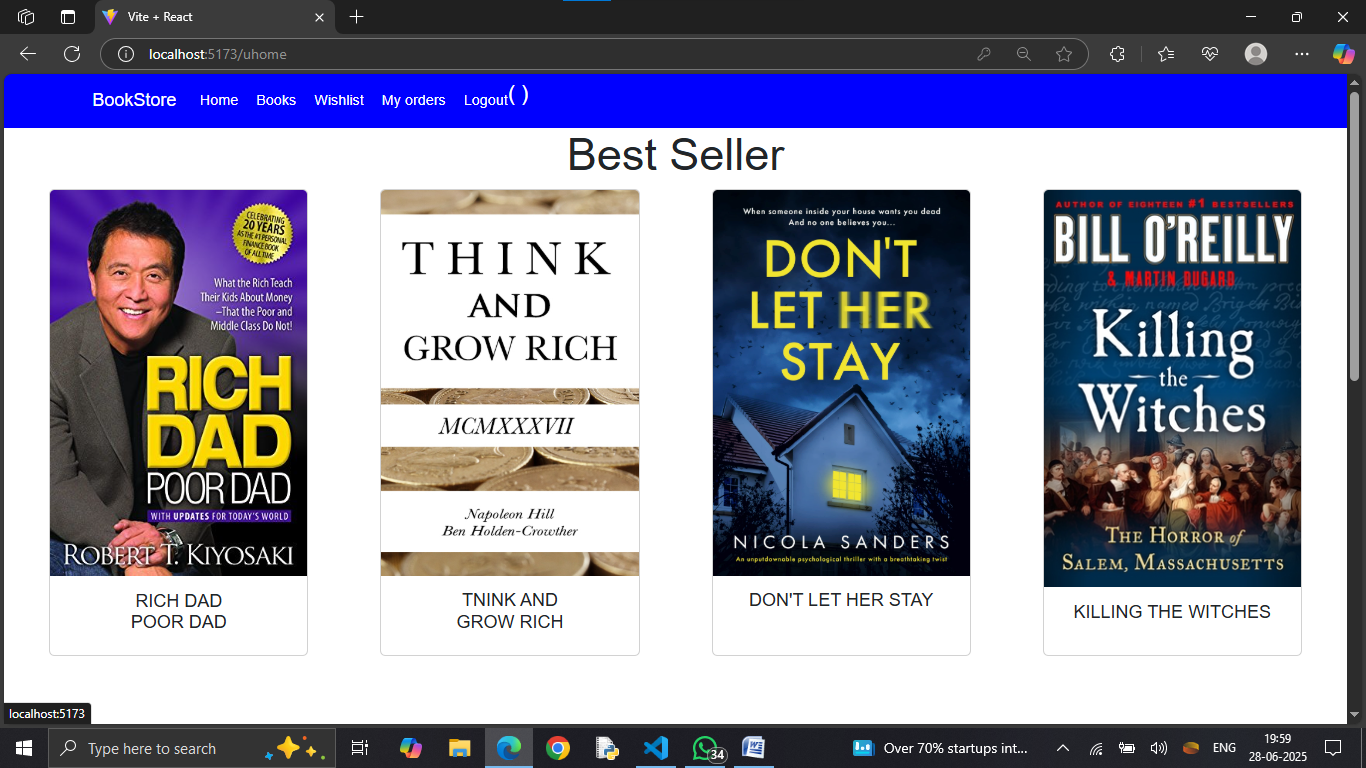
**Landing page:-**

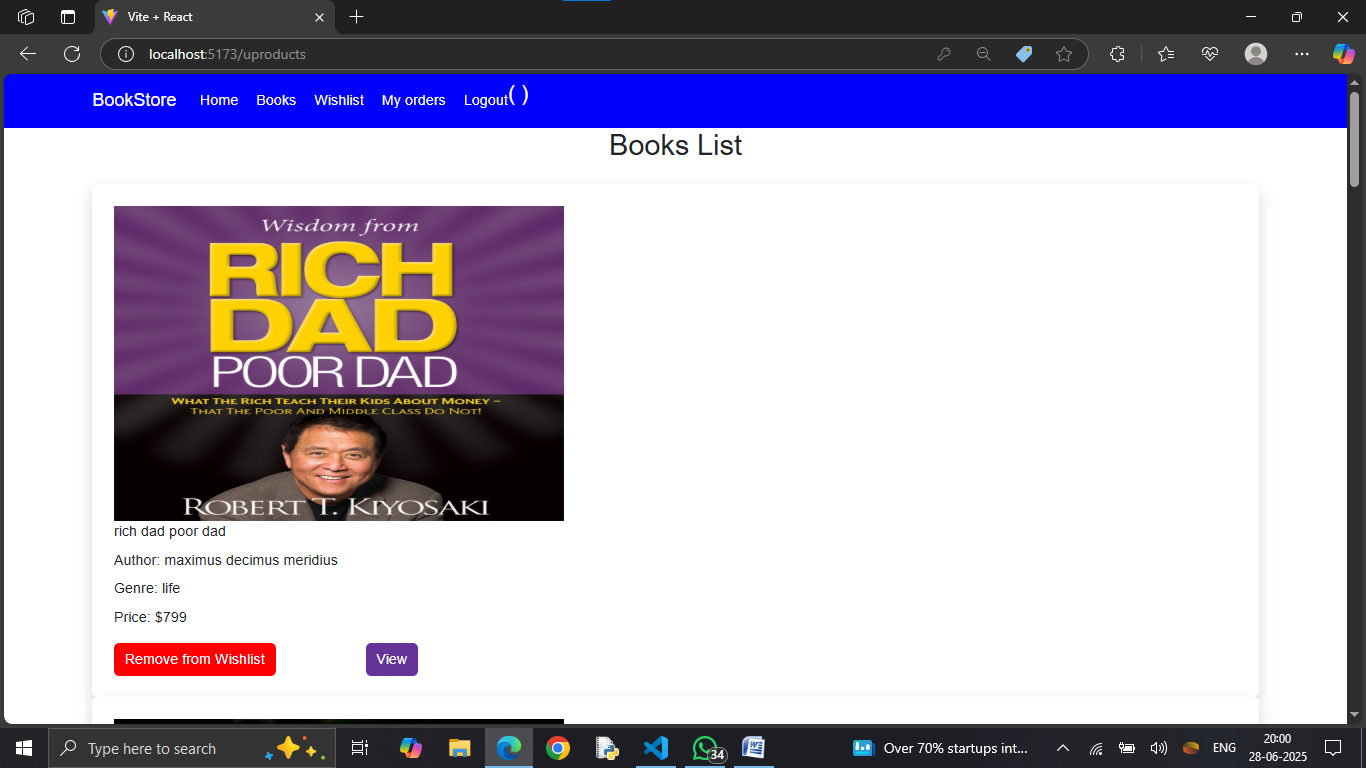
****

**Login Page:-**

****

**Home Page:-**

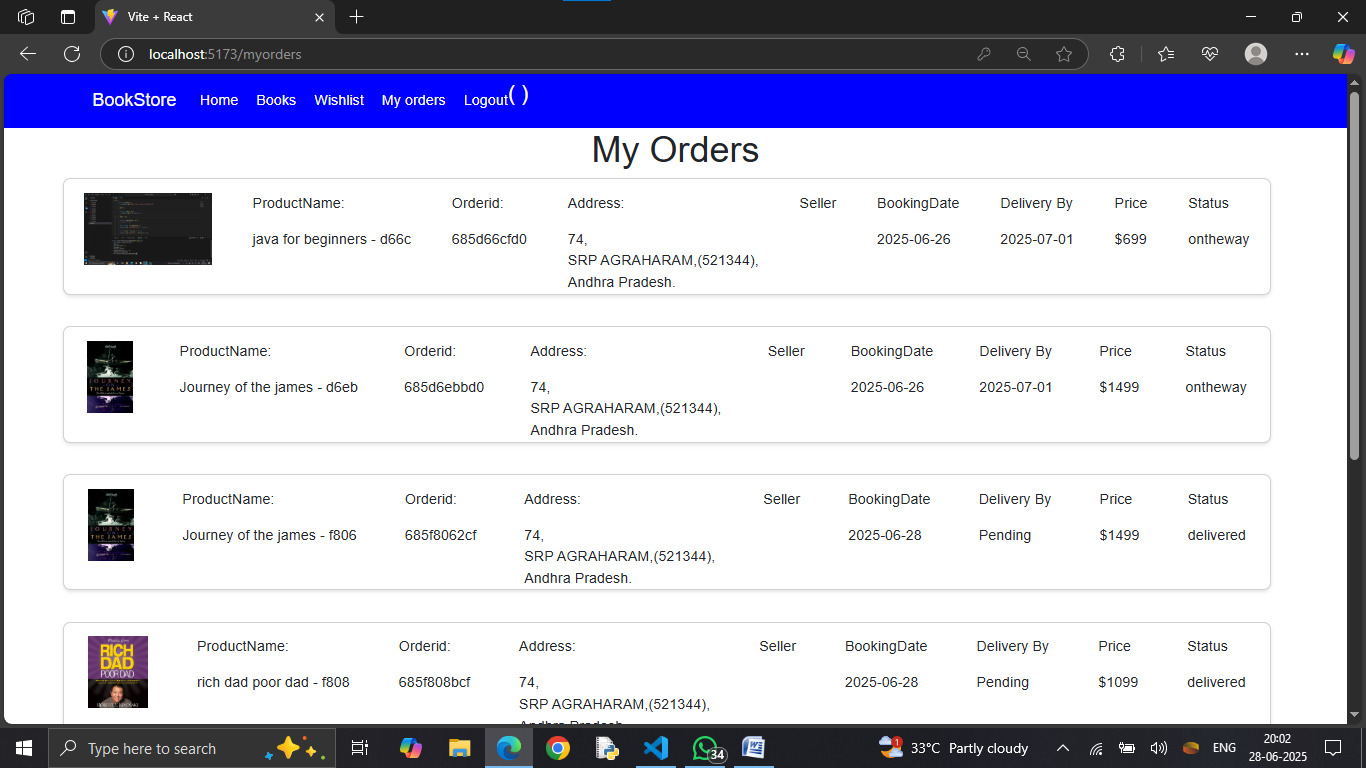
****

**Books Page:- **

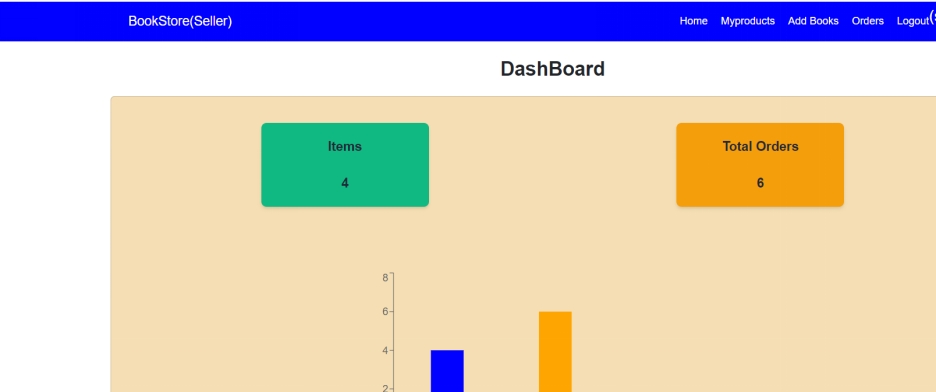
**Wishlist Page:-**

****

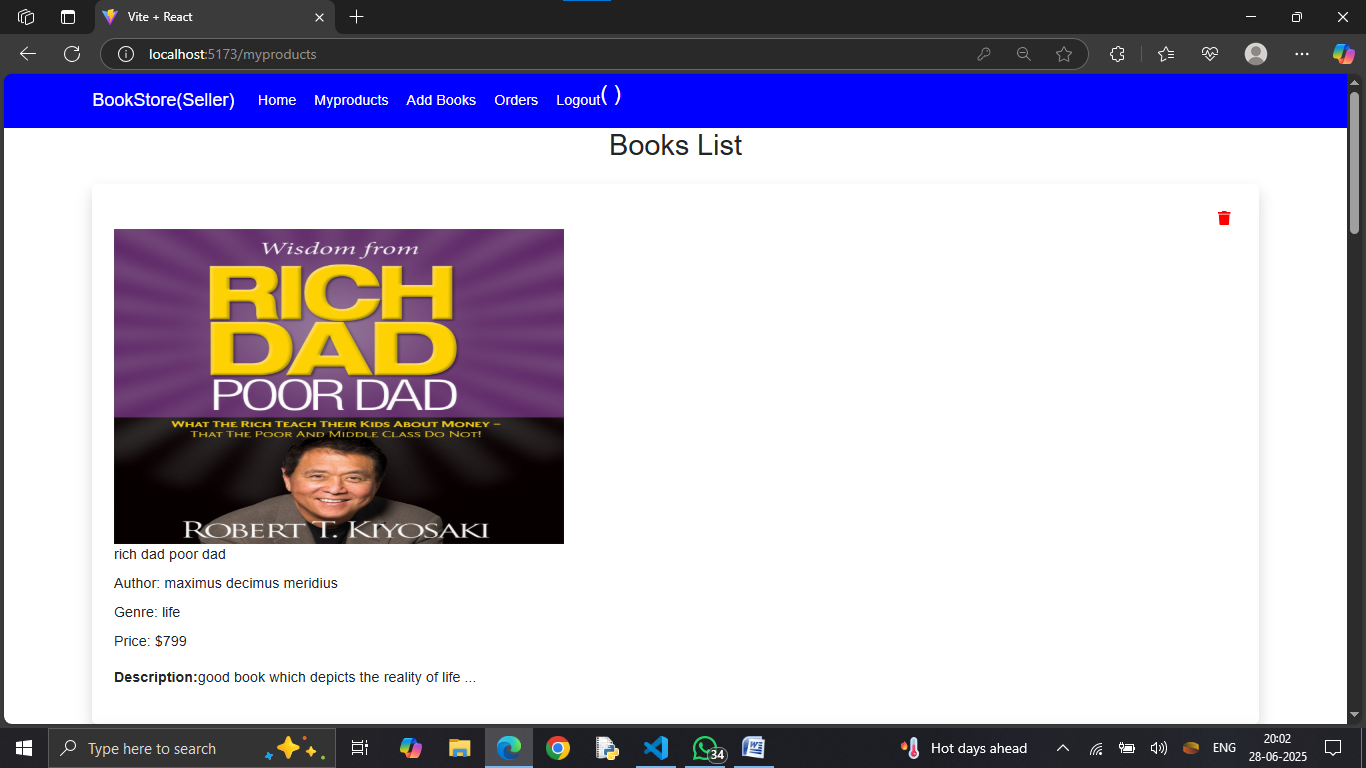
**My Bookings Page :-**

****

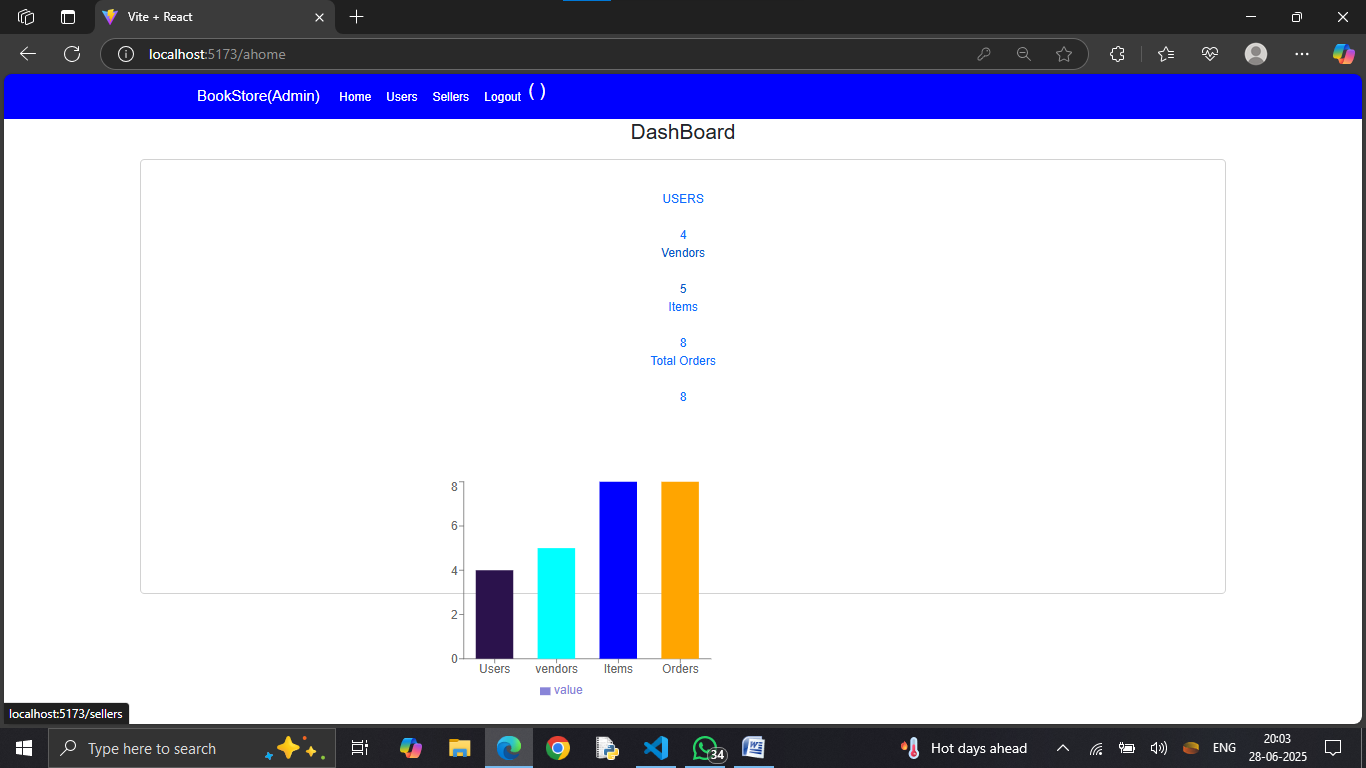
**Seller Dashboard:-**

****

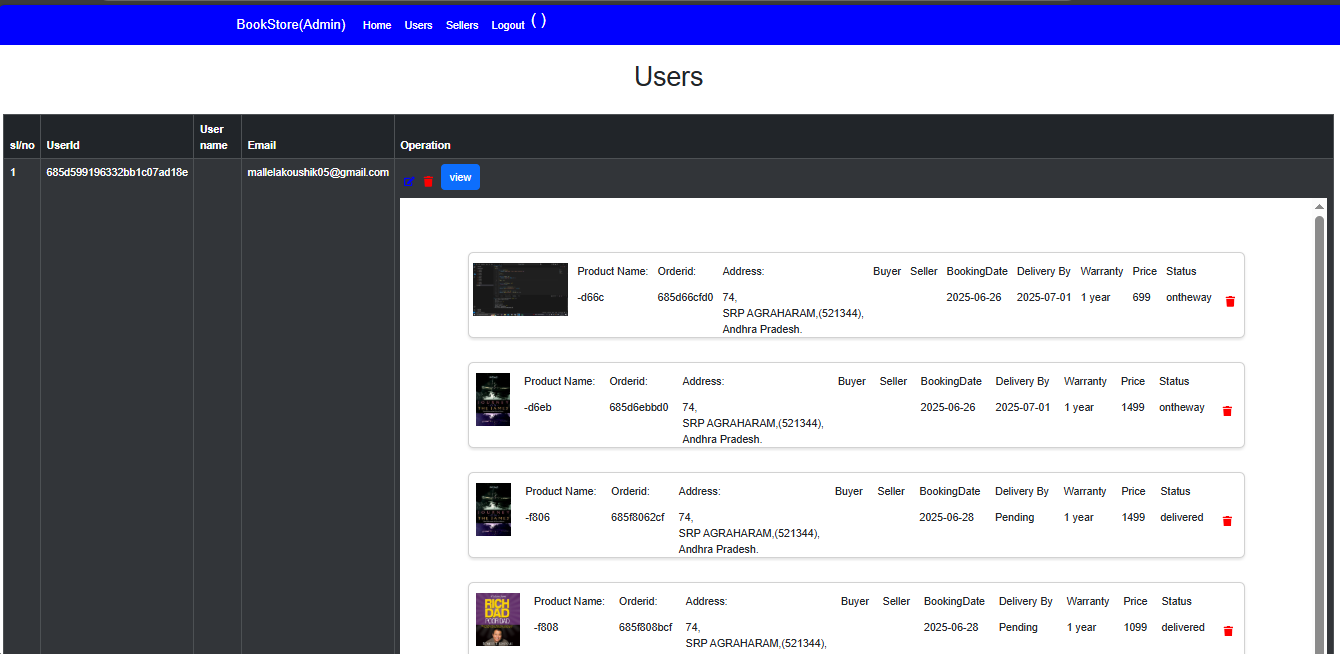
**Seller Items:**

****

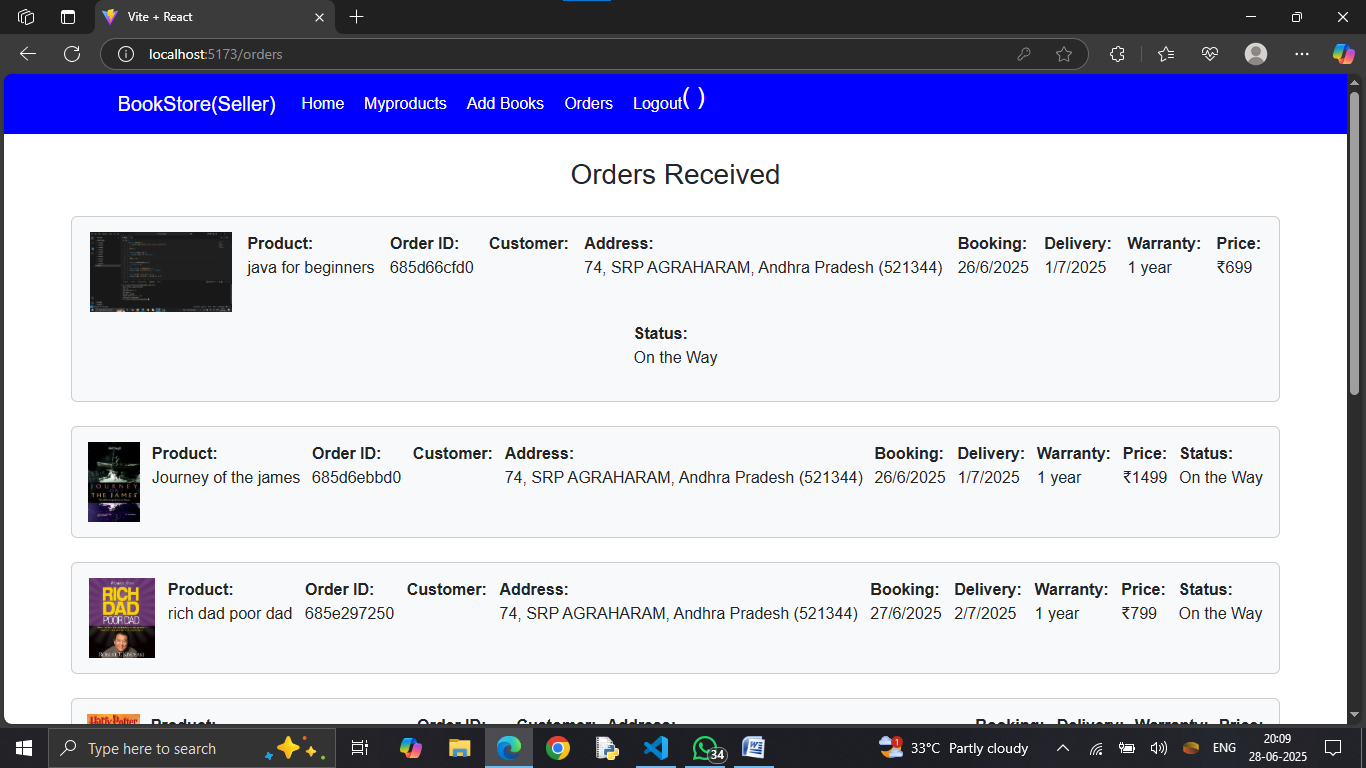
**Admin Dashboard:-**

****

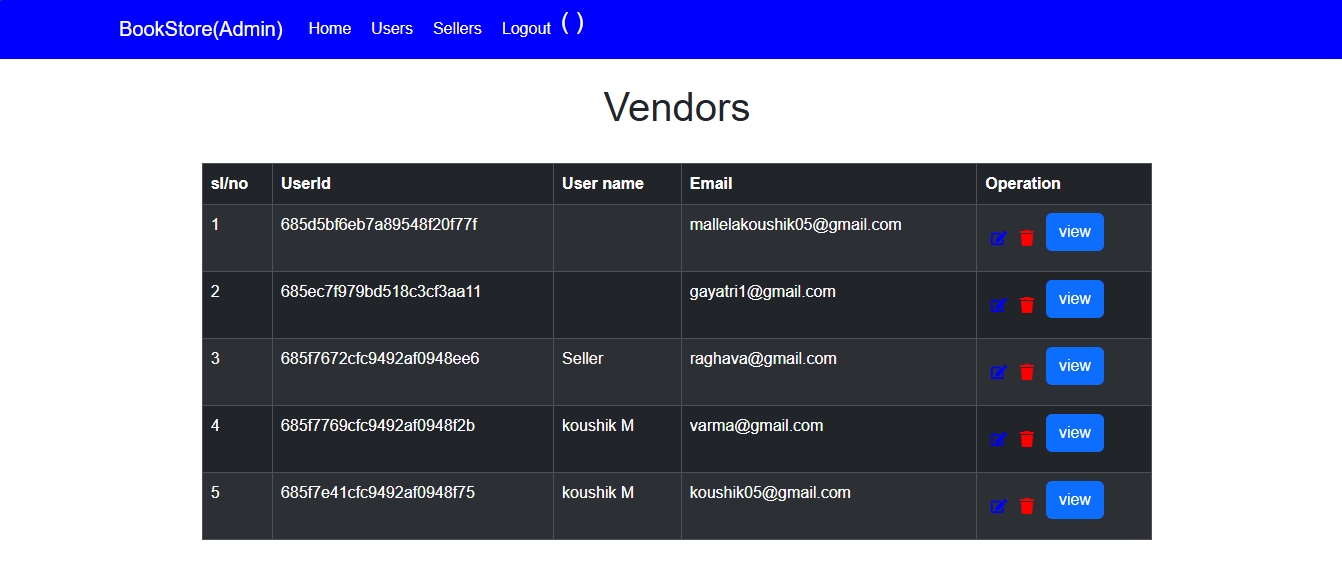
**Users Page:**

****

**User Orders:**

****

**Sellers Page:**

****

**SUMMARY**

BookNest is a full-stack online bookstore web application built using the MERN stack (MongoDB, Express.js, React, and Node.js). It provides users with a seamless platform to browse, search, and purchase books online. The platform includes key features such as secure user registration and login, detailed book listings with genre and author filters, cart management, and order history tracking. Users can also maintain wishlists and manage their bookings. The responsive design ensures accessibility across desktops, tablets, and mobile devices, offering a smooth and user-friendly experience.

The application also includes dedicated dashboards for sellers and administrators. Sellers can add, edit, and manage their book inventory, while admins have control over users, sellers, book listings, and overall analytics. Technologies like Axios handle API requests, Multer supports image uploads, and Recharts is used for displaying analytics. With its clean UI and strong backend architecture, BookNest serves as a comprehensive and efficient solution for managing an online bookstore**.**