

Full Stack Development with MERN:

Project Documentation

1. Introduction

- **Project Title :** BookNest – Where Stories Nestle
 - **Team Id :** LTVIP2025TMID55365
 - **Team Members:**
 - **Pattapu Chohitha – Team Leader**
Led the project from ideation to deployment. Managed planning, development, and integration of all modules. Took primary responsibility for the frontend, backend, database, and documentation.
 - **Pantula Vindhya Kanaka Sai Pavani – Team Member, Frontend Developer (Support Role)**
Assisted in designing UI components and maintaining visual consistency.
 - **Panguluri Vishnu Vardahn – Team Member, Backend Developer (Support Role)**
Provided support in building REST APIs and implementing user authentication logic.
 - **Pamarthi Sri Lakshmi – Team Member, Database Developer (Support Role)**
Contributed to initial schema discussions and data modeling.
-

2. Project Overview

- **Purpose:**
The primary goal of BookNest is to create a seamless, user-friendly online bookstore that caters to modern readers and local sellers. It aims to bridge the gap between traditional book shopping and digital convenience by offering personalized browsing, role-based dashboards, and efficient order management. This project not only simplifies the book-buying experience for readers like Sarah (our scenario user) but also empowers sellers to manage their inventory

efficiently. To provide a seamless and immersive digital bookstore experience where users can explore, search, and buy books, and sellers/admins can manage book listings and analytics.

- **Key Features:**

- **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.
-

3. Architecture

- **Frontend (React):**
 - Built with React functional components and React Router.
 - Styled with Tailwind CSS/Bootstrap.
 - Responsive design for all screen sizes.
 - **Backend (Node.js + Express):**
 - RESTful API endpoints.
 - Modular architecture separating controllers, routes, and middleware.
 - Integrated JWT for authentication.
 - **Database (MongoDB):**
 - Schemas for Users, Books, Orders, and Admin roles.
 - Mongoose ODM for model interaction.
 - Efficient querying and schema relationships.
-

4. Setup Instructions

- **Prerequisites:**
 - Node.js (v18+)
 - MongoDB Atlas/local instance
 - Git
- **Installation:**

1. Clone the repo:

```
git clone [your-repo-url]
```

2. Navigate into directories:

```
cd client
```

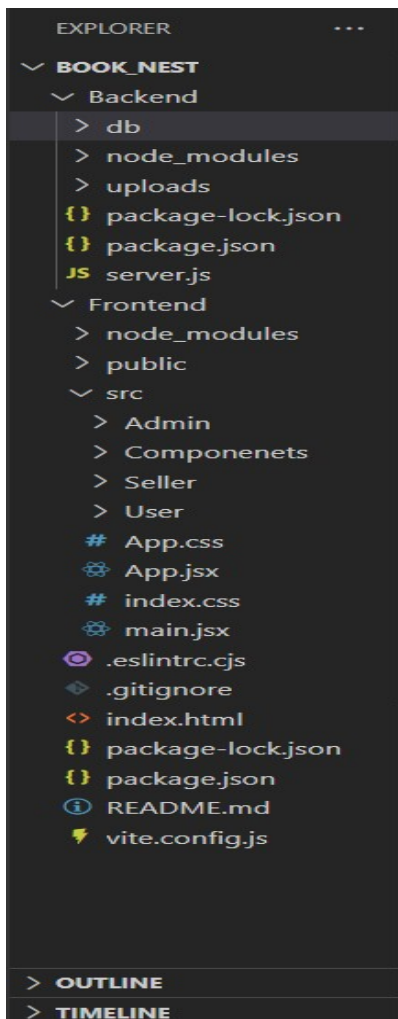
```
npm install
```

```
cd ../server
```

```
npm install
```

3. Add .env file with Mongo URI and JWT_SECRET in server folder.

5. Folder Structure



- **Client (React Frontend):**
 - src/components: Reusable UI components
 - src/pages: Route-based pages (Home, Login, etc.)
 - src/App.js: Root component with routing logic
 - **Server (Node Backend):**
 - routes/: Route definitions
 - controllers/: Business logic
 - models/: Mongoose schemas
 - middleware/: JWT verification and error handlers
-

6. Running the Application

- **Frontend:**

```
cd client
npm start
```

- **Backend:**

```
cd server
npm start
```

7. API Documentation

Endpoint	Method	Description
/api/register	POST	Register new user
/api/login	POST	Authenticate user
/api/books	GET	Fetch all books
/api/cart	POST	Add book to cart
/api/orders	POST	Place order
/api/admin/books	CRUD	Admin book management

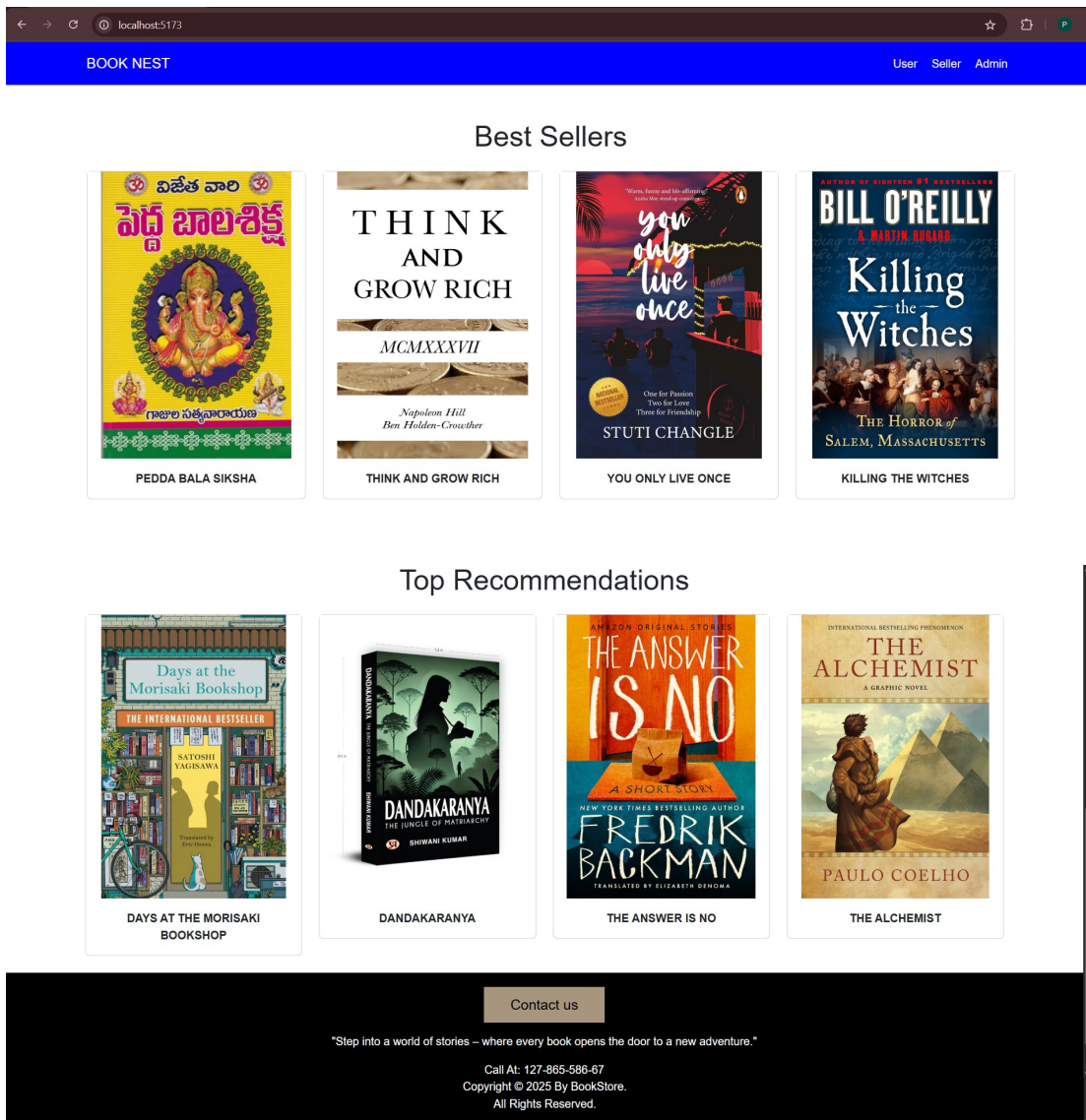
Note: Full request/response examples in Postman collection.

8. Authentication

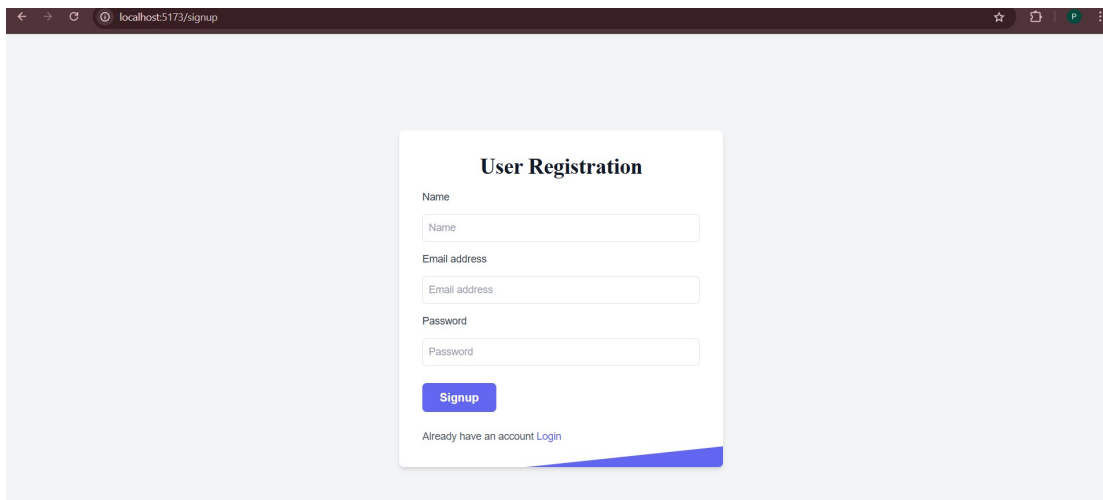
- Uses **JWT tokens** stored in local storage.
- Backend middleware validates tokens on protected routes.
- Separate login access for users, sellers, and admin roles.

9. User Interface

- Fully responsive design using TailwindCSS.
- **Landing Page**



• User Registration Page



A screenshot of a web browser at localhost:5173/signup. The page features a central white registration form with a blue 'Signup' button and a link to the login page.

User Registration

Name

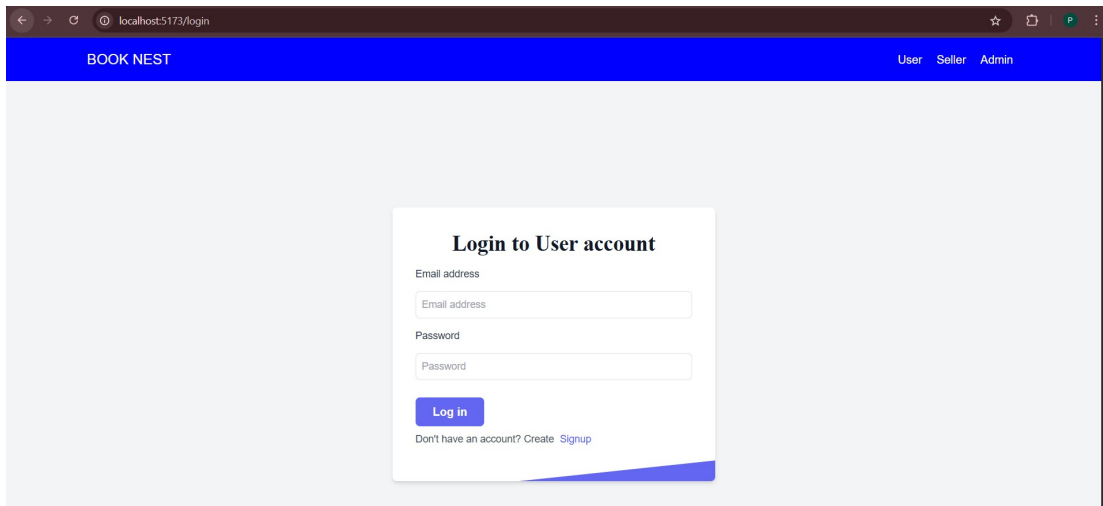
Email address

Password

Signup

Already have an account [Login](#)

• User Login Page



A screenshot of a web browser at localhost:5173/login. The page has a blue header with 'BOOK NEST' and navigation links. The login form is centered with a blue 'Log in' button and links for account creation.

BOOK NEST [User](#) [Seller](#) [Admin](#)

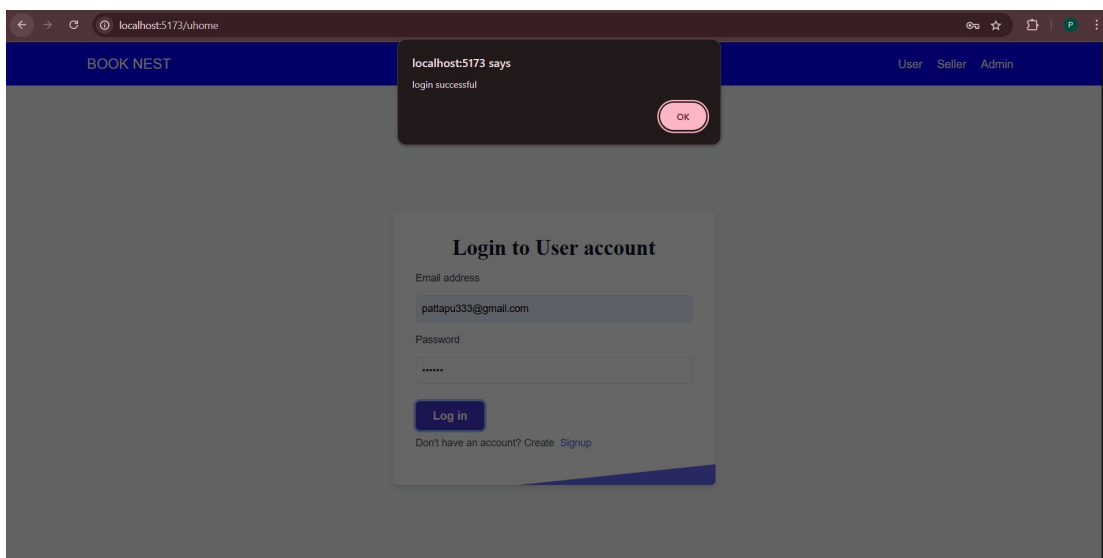
Login to User account

Email address

Password

Log in

Don't have an account? [Create](#) [Signup](#)



A screenshot of the same login page after a successful login. A dark toast notification is displayed at the top, and the login form is dimmed in the background.

BOOK NEST [User](#) [Seller](#) [Admin](#)

localhost:5173 says
login successful **OK**

Login to User account

Email address

Password

Log in

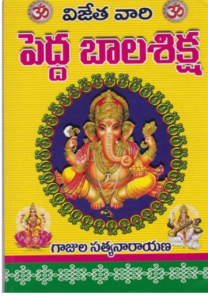
Don't have an account? [Create](#) [Signup](#)

• User Dashboard

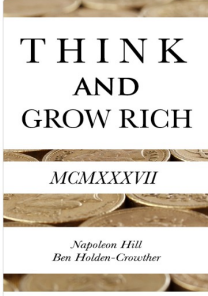
BookStore

Home Books Wishlist My orders Logout(PATTAPU CHOHITHA)

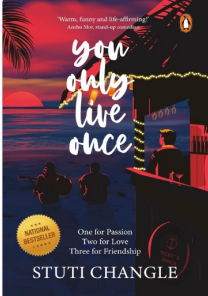
Best Seller



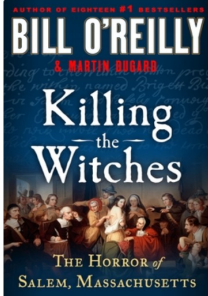
PEDDA BALA SIKSHA



TNINK AND GROW RICH



YOU ONLY LIVE ONCE



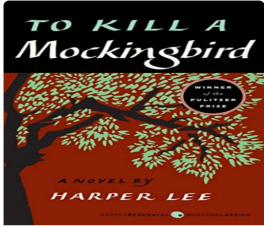
KILLING THE WITCHES

• Book List Page

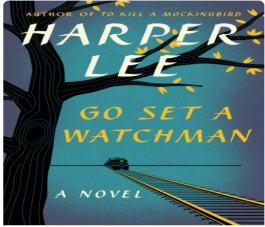
BookStore

Home Books Wishlist My orders Logout(PATTAPU CHOHITHA)

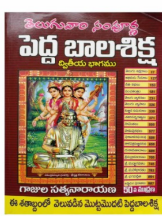
Books List



To Kill A Mocking Bird
Author: Haper Lee
Genre: Fiction
Price: \$349
[Add to Wishlist](#) [View](#)



Go Set A Watchman
Author: Haper Lee
Genre: Fiction
Price: \$549
[Add to Wishlist](#) [View](#)



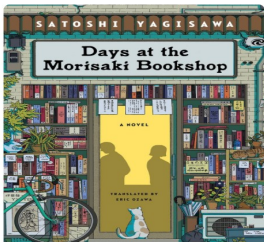
Pedda Bala Siksha
Author: Satya Narayana
Genre: All
Price: \$600
[Add to Wishlist](#) [View](#)

• User Wishlist Page

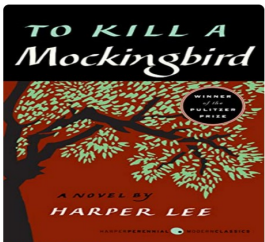
BookStore

Home Books Wishlist My orders Logout(PATTAPU CHOHITHA)

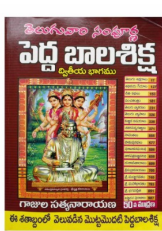
Wishlist



Days at the Morisaki Bookshop
Author:
Genre:
Price: \$
[Remove from Wishlist](#) [View](#)



To Kill A Mocking Bird
Author:
Genre:
Price: \$
[Remove from Wishlist](#) [View](#)

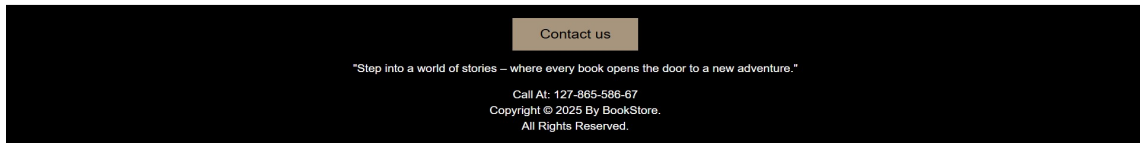


Pedda Bala Siksha
Author:
Genre:
Price: \$
[Remove from Wishlist](#) [View](#)

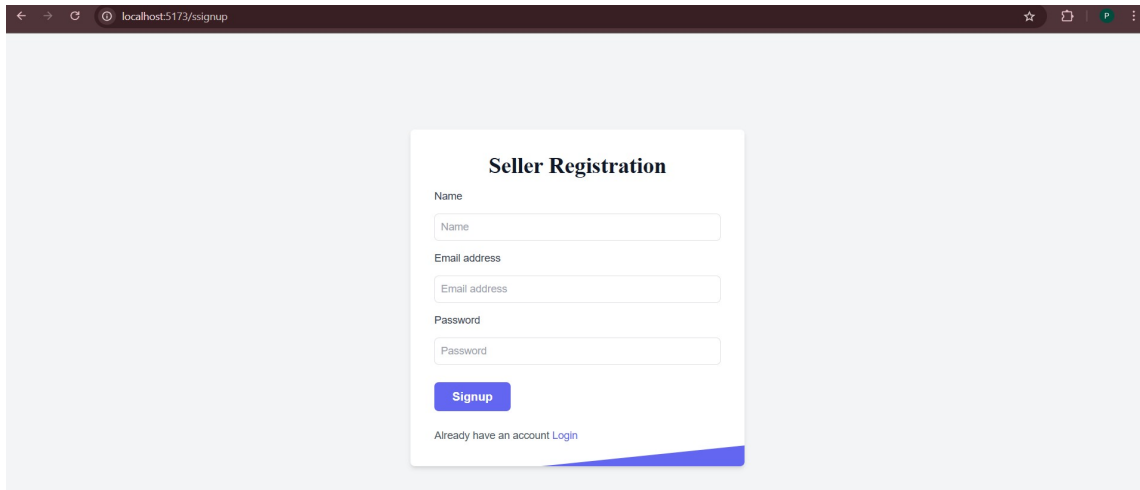
• User Orders Page



ProductName:	Orderid:	Address:	Seller	BookingDate	Delivery By	Price	Status
 -1882	685f882f7e	12/234, Gudivada,(521301), AP,	Satoshi Yagisawa	28/6/2025	7/5/2025	\$112	ontheway



• Seller Registartion Page



Seller Registration

Name

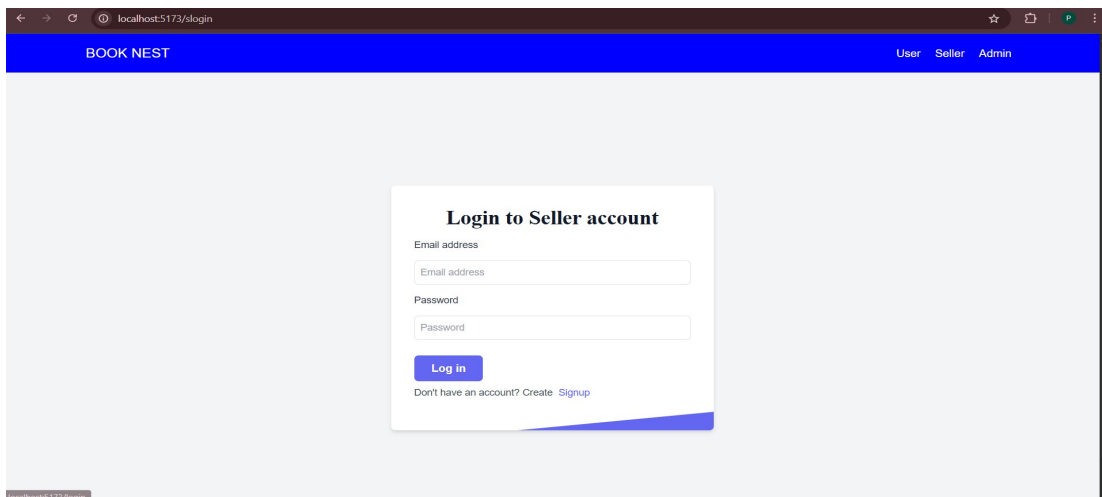
Email address

Password

Signup

Already have an account Login

• Seller Login Page



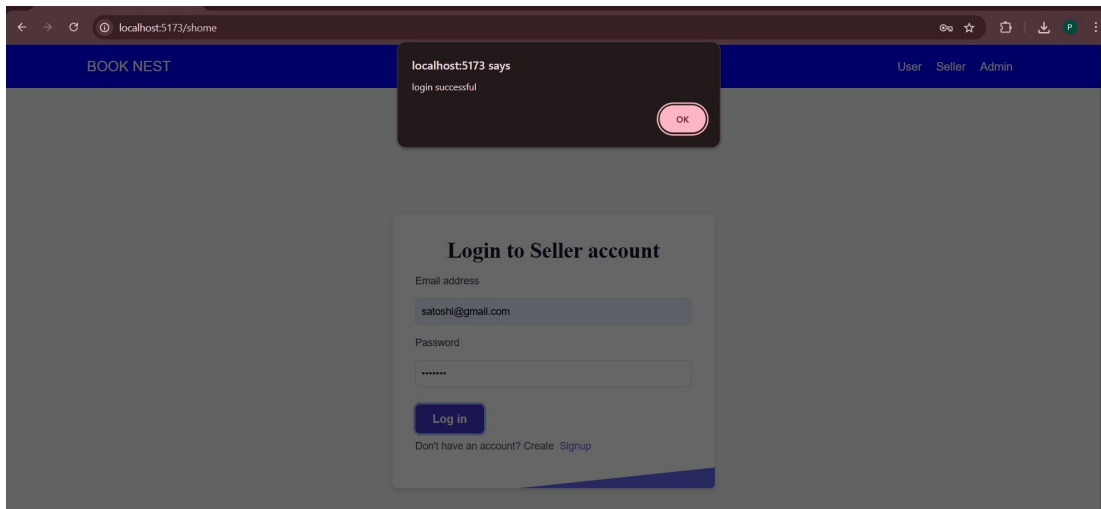
Login to Seller account

Email address

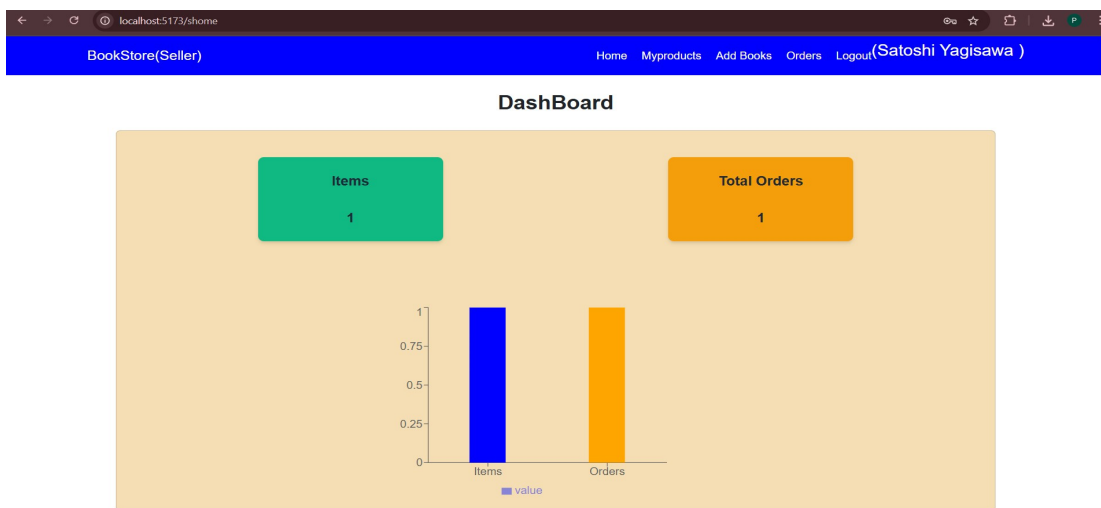
Password

Log in

Don't have an account? Create Signup

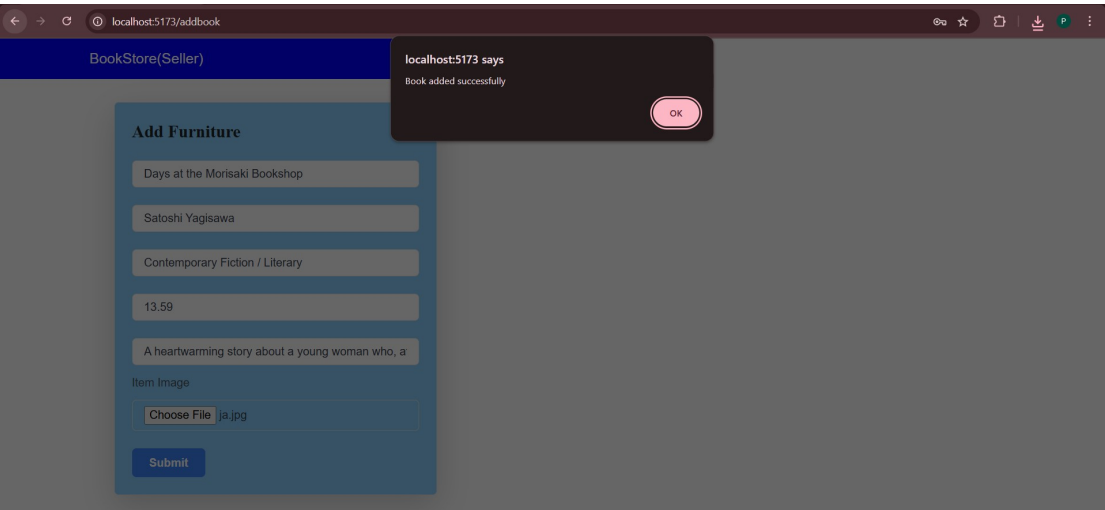


• Seller Dashboard

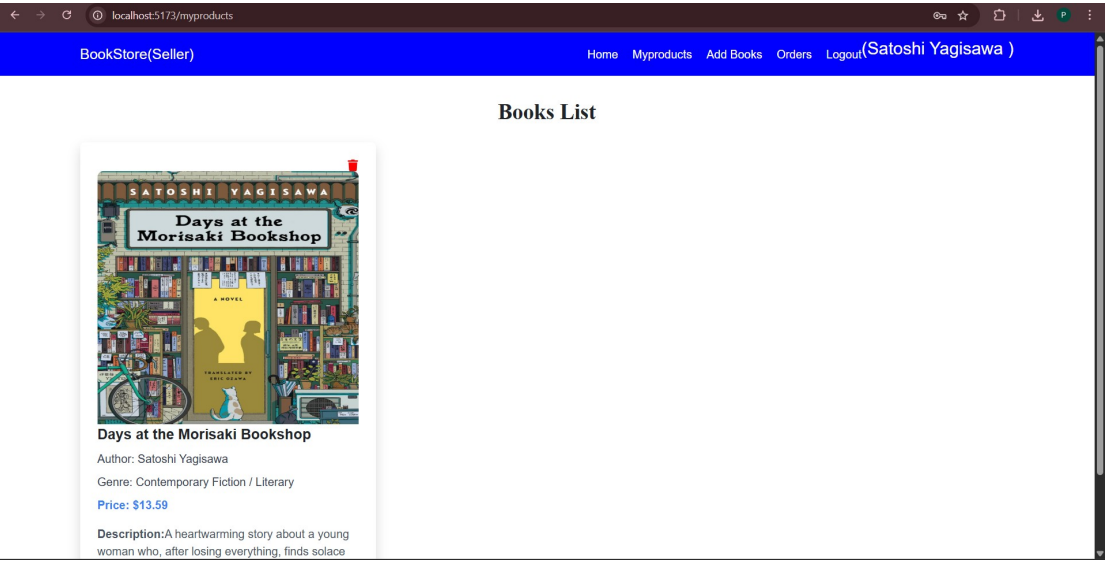


• Seller Add Book Page

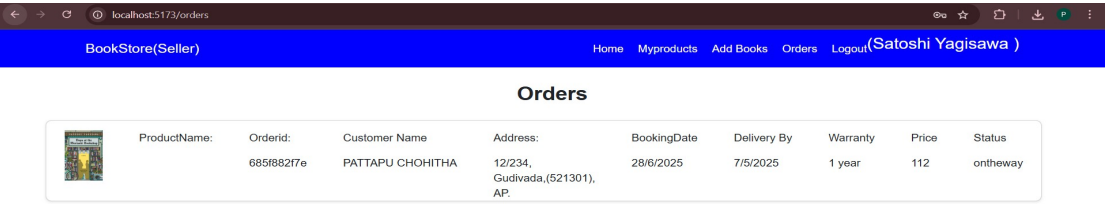
The screenshot shows the "Add Furniture" form on the "BookStore(Seller)" page at localhost:5173/addbook. The form is a light blue box with the following fields: "title", "author", "genre", "Price", and "Description". Below these is an "Item Image" section with a "Choose File" button and the text "No file chosen". A "Submit" button is at the bottom of the form.



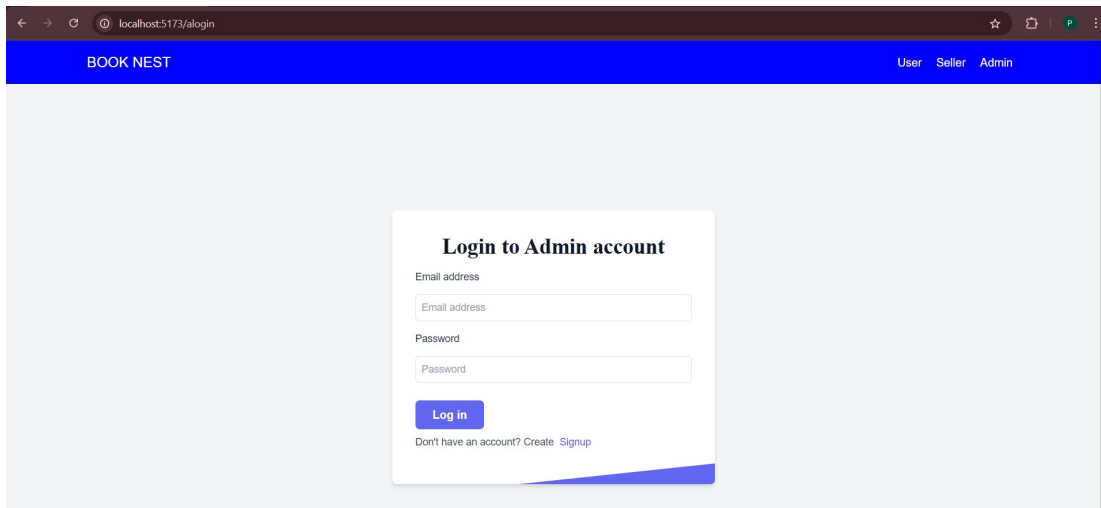
- ## Seller Book List Page



- ## Seller Order List Page



- ## Admin Login Page



A screenshot of a web browser at localhost:5173/alogin. The page has a blue header with 'BOOK NEST' on the left and 'User Seller Admin' on the right. The main content area is light gray and contains a white login form titled 'Login to Admin account'. The form has two input fields: 'Email address' and 'Password'. Below the fields is a blue 'Log in' button. At the bottom of the form, it says 'Don't have an account? Create Signup'.

BOOK NEST

User Seller Admin

Login to Admin account

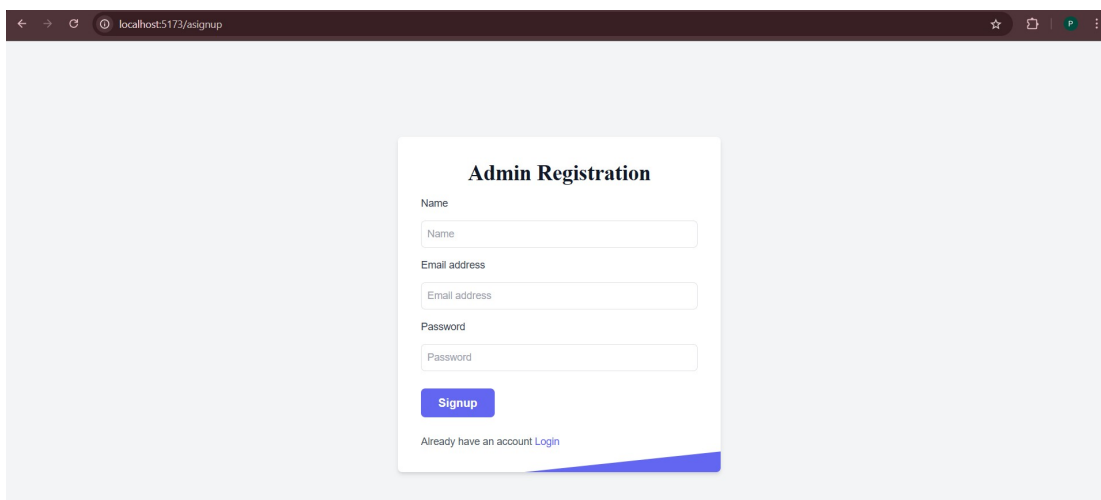
Email address

Password

Log in

Don't have an account? Create Signup

- ## Admin Registration Page



A screenshot of a web browser at localhost:5173/asignup. The page has a light gray background and a white registration form titled 'Admin Registration'. The form has three input fields: 'Name', 'Email address', and 'Password'. Below the fields is a blue 'Signup' button. At the bottom of the form, it says 'Already have an account Login'.

Admin Registration

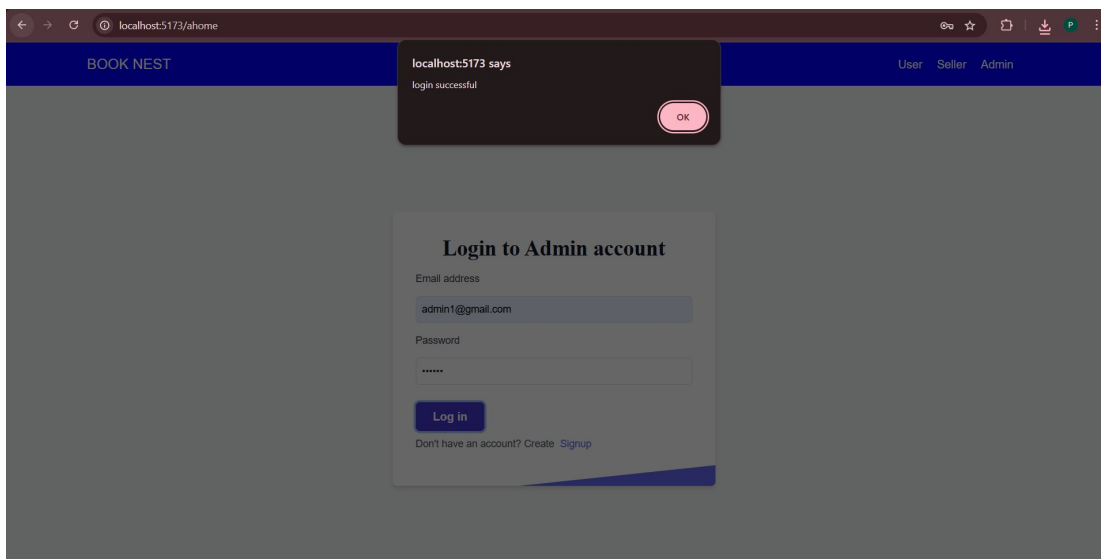
Name

Email address

Password

Signup

Already have an account Login



A screenshot of a web browser at localhost:5173/ahome. The page has a blue header with 'BOOK NEST' on the left and 'User Seller Admin' on the right. The main content area is dark gray. A dark gray modal box is centered on the screen, displaying 'localhost:5173 says' and 'login successful' with an 'OK' button. Below the modal, the 'Login to Admin account' form is visible, but the input fields are filled with text: 'Email address' is 'admin1@gmail.com' and 'Password' is '*****'. The 'Log in' button is still present.

BOOK NEST

User Seller Admin

localhost:5173 says
login successful

OK

Login to Admin account

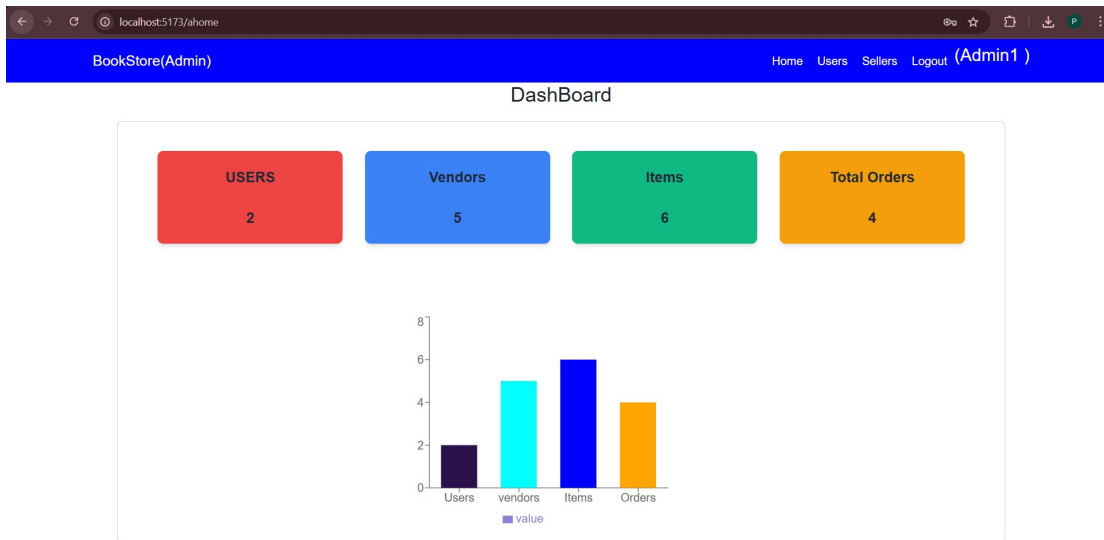
Email address

Password

Log in

Don't have an account? Create Signup

• Admin Dashboard



• Vendors List Page

The Vendors List Page (localhost:5173/sellers) displays a table of vendor information. The table has columns for sl/no, Userid, User name, Email, and Operation. Each row includes a 'view' button.

sl/no	Userid	User name	Email	Operation
1	685d48937d8ac68f06b9b65a	Rishwanth	rishi@gmail.com	view
2	685d492d7d8ac68f06b9b66a	Harper Lee	harper@gmail.com	view
3	685f845b7e95312190f4d1db	Satya Naryana	narayana@gmail.com	view
4	685f85ad7e95312190f4d110	Napoleon Hill	hill@gmail.com	view
5	685f86f47e95312190f4d202	Satoshi Yagisawa	satoshi@gmail.com	view

• Vendor Products Page

The Vendor Products Page (localhost:5173/sellers) displays a modal window titled 'Vendor Products'. The modal shows two product entries with their details and a 'Close' button at the bottom left.

Product Name:	Orderid:	Warranty	Price
-f838	685f83847e	1 year	349
-f83d	685f83d87e	1 year	549

11. Demo

<https://drive.google.com/file/d/1IV60sPoNLINH6ZdNAh3YPdjUqR8SD4d1/view?usp=sharing>

12. Known Issues

- No OTP/email verification currently.
-

13. Future Enhancements

- Add payment gateway integration (Razorpay/Stripe)
- Recommendation engine using ML
- User reviews and ratings for books
- Chatbot for book search assistance
- Mobile app version