BookNest: Where Stories Nestle

Full Stack Development With MERN

Project Title:

BookNest – Online Book Shopping Platform

Team Members:

- M.Sai Santosh Vamsi (Full Stack Developer)
- D.V.J.Kranthi Kumar(Full Stack Developer)
- D.S.k.prameela (Full Stack Developer)
- D.Varthika (Full Stack Developer)

Project Overview

Purpose:

BookNest is a feature-rich online bookstore platform that offers seamless experiences for readers and book sellers. The platform includes an engaging UI/UX, secure login, inventory and order management for sellers, review/rating features for customers, and integrated payment options.

Key Goals:

- Enable users to easily discover and purchase books
- Allow sellers to efficiently manage their book listings and orders
- Provide a secure and responsive e-commerce experience

Key Features

Customer Features

- Browse books with search & category filtering
- Shopping cart with quantity and coupon support
- Cashfree Payment Gateway Integration
- User Ratings and Book Reviews
- Wishlist and Order History
- Supports INR (₹)

Seller Features

- Seller Dashboard for inventory and order management
- Book CRUD operations with image uploads
- Real-time Revenue Tracking
- Stock Monitoring and Alerts
- Order lifecycle: Pending → Processing → Shipping → Delivered

Architecture

Frontend (React + TypeScript):

- Build Tool: Vite
- Styling: TailwindCSS + ShadCN UI
- Routing: React Router
- State Management: Local state and Context API
- Key Pages: Home, Login, Register, Books, Cart, Dashboard
- Components: Navbar, ReviewSection, Cards, Toasts
- Services: Axios API handler

Backend (Node.js + Express.js):

- Main File: index.js
- Routes: auth, user, book, order, review, analytics
- Controllers: authController.js, bookController.js, etc.
- Middleware: JWT-based authentication
- Database: MongoDB using Mongoose

Database Schema

MongoDB Collections

- User: email, password, name, role (customer/seller/admin)
- Book: title, author, description, price, image, stock, seller ID
- Order: user, orderItems, payment info, order status, timestamps

Setup Instructions

Prerequisites:

- Node.js v16+
- MongoDB v4.4+
- npm or yarn
- Cashfree & Cloudinary accounts

Installation:

```bash
git clone <repository-url>
cd BookNest

# Backend
cd server
npm install

# Frontend
cd ../Client
npm install
```

Folder Structure

```
BookNest/
├── Client/       # Frontend
│    ├── src/
│    │    ├── components/
│    │    ├── pages/
│    │    ├── services/
│    │    └── App.tsx
├── server/       # Backend
│    ├── controllers/
│    ├── models/
│    ├── routes/
│    ├── middleware/
│    └── index.js
```

```
├── .git
├── README.md
```

API Documentation

Authentication

- POST /api/auth/register – Register a new user
- POST /api/auth/login – User login
- GET /api/auth/me – Get current user

Books

- GET /api/books – List all books
- GET /api/books/:id – View single book
- POST /api/books – Add a book (seller)
- PUT /api/books/:id – Edit book (seller)
- DELETE /api/books/:id – Delete book (seller)

Orders

- GET /api/orders – Filtered by user/seller
- POST /api/orders/create-after-payment
- POST /api/orders/cashfree – Payment
- PUT /api/orders/:id/status – Update order

Reviews

- POST/PUT/DELETE /api/reviews – Manage reviews
- GET /api/reviews/book/:id – Reviews for a book

Users

- Wishlist operations via /api/users/:id/wishlist

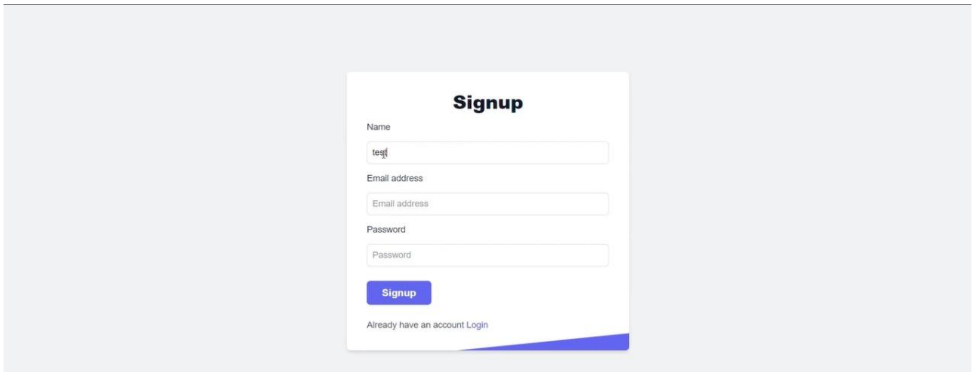
Authentication and Security

- JWT-based secure login
- Tokens generated on login
- Protected routes based on roles: admin, seller, customer

Screenshots or Demo

https://booknest.vercel.app

Screenshots



Signup

Name

test

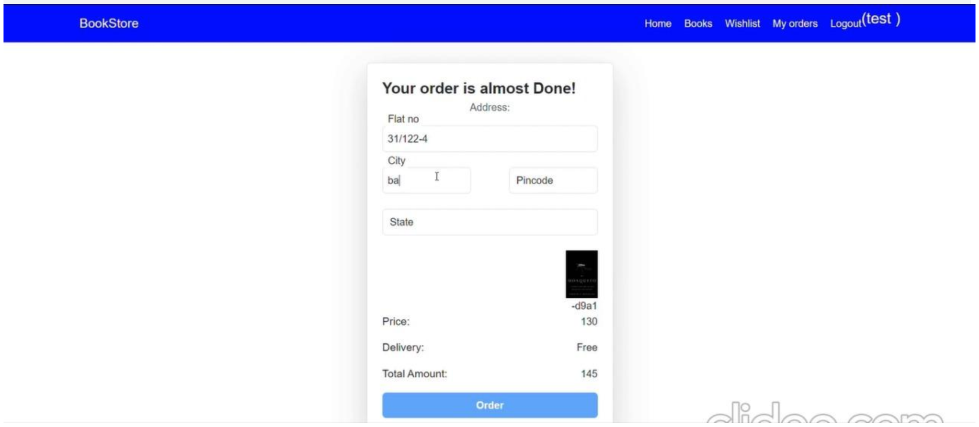
Email address

Email address

Password

Password

Already have an account Login



Your order is almost Done!

Address:

Flat no

31/122-4

City

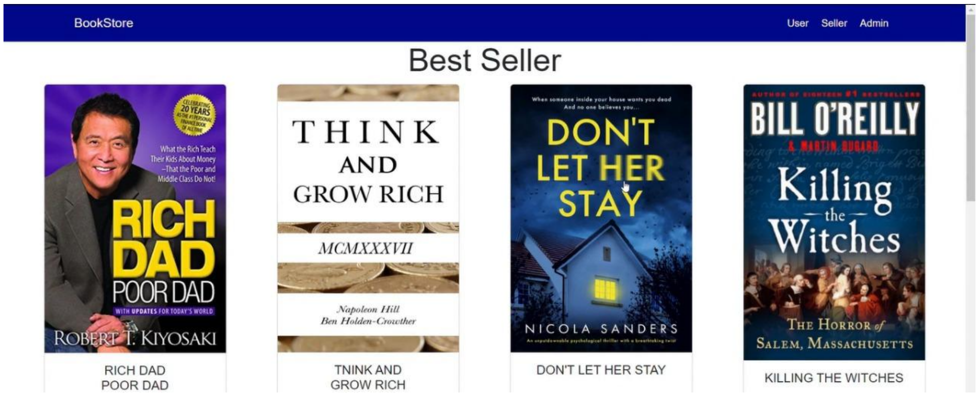
ba

Pincode

State

-d9a1

Price:	130
Delivery:	Free
Total Amount:	145



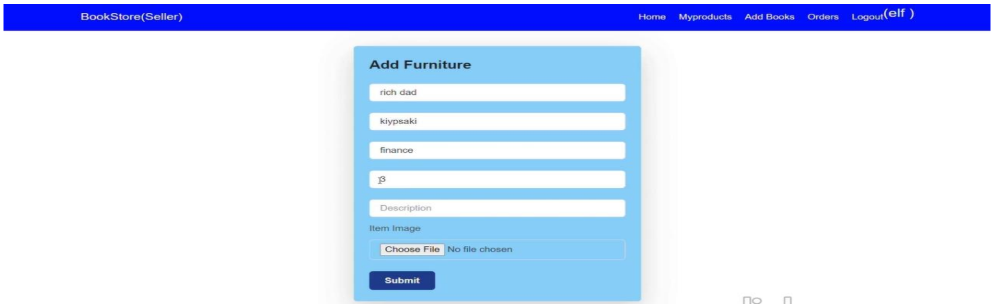
Best Seller

RICH DAD POOR DAD

TNINK AND GROW RICH

DON'T LET HER STAY

KILLING THE WITCHES



Add Furniture

rich dad

kiypsaki

finance

Description

Item Image

Choose File No file chosen

Known Issues

- Image Uploads May Fail on Slow Networks
- Wishlist Not Synced Across Devices.

Future Enhancements

- Razorpay integration
- React Native Mobile App
- Live Order Tracking
- Notifications for sellers and users