

# 1. INTRODUCTION

## 1.1 Project Overview

Booknest is a full-stack web application developed using the MERN stack—MongoDB, Express.js, React.js, and Node.js—that serves as a modern online bookstore platform. The goal of this project is to allow users to browse, wishlist, and purchase books from anywhere and enable administrators to manage books, users, and orders efficiently via an admin dashboard.

Built as a solo development initiative, Booknest demonstrates complete ownership over the software development lifecycle, including system architecture, frontend development, backend APIs, database modeling, and deployment readiness.

The application offers a seamless experience for both general users (readers/customers) and admin users (store managers). Features like book cover uploads, category filtering, secure login with JWT, and role-based access control ensure the platform is feature-rich and production-ready.

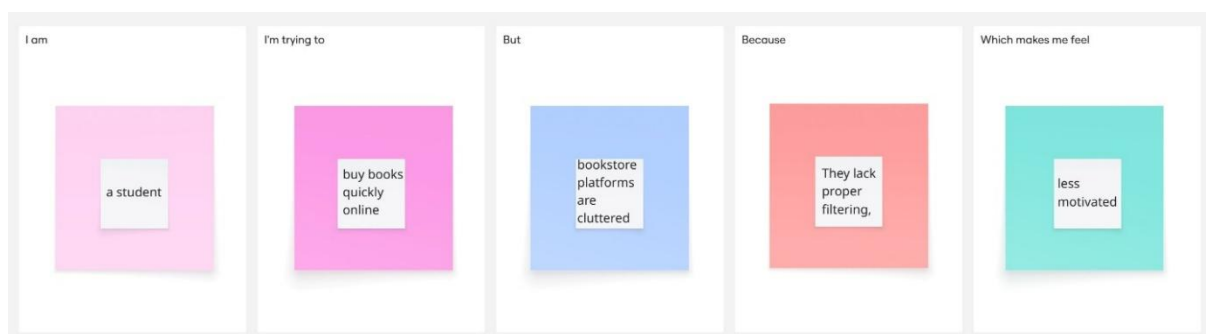
## 1.2 Purpose

The purpose of Booknest is to provide a simple, efficient, and user-friendly online bookstore platform where users can browse, wishlist, and order books, while administrators manage inventory and users through a secure dashboard.

It solves the need for convenient book access by eliminating the constraints of physical shopping and offering a structured digital alternative. Additionally, it showcases full-stack development skills using the MERN stack, covering frontend, backend, authentication, and database integration in a single cohesive application.

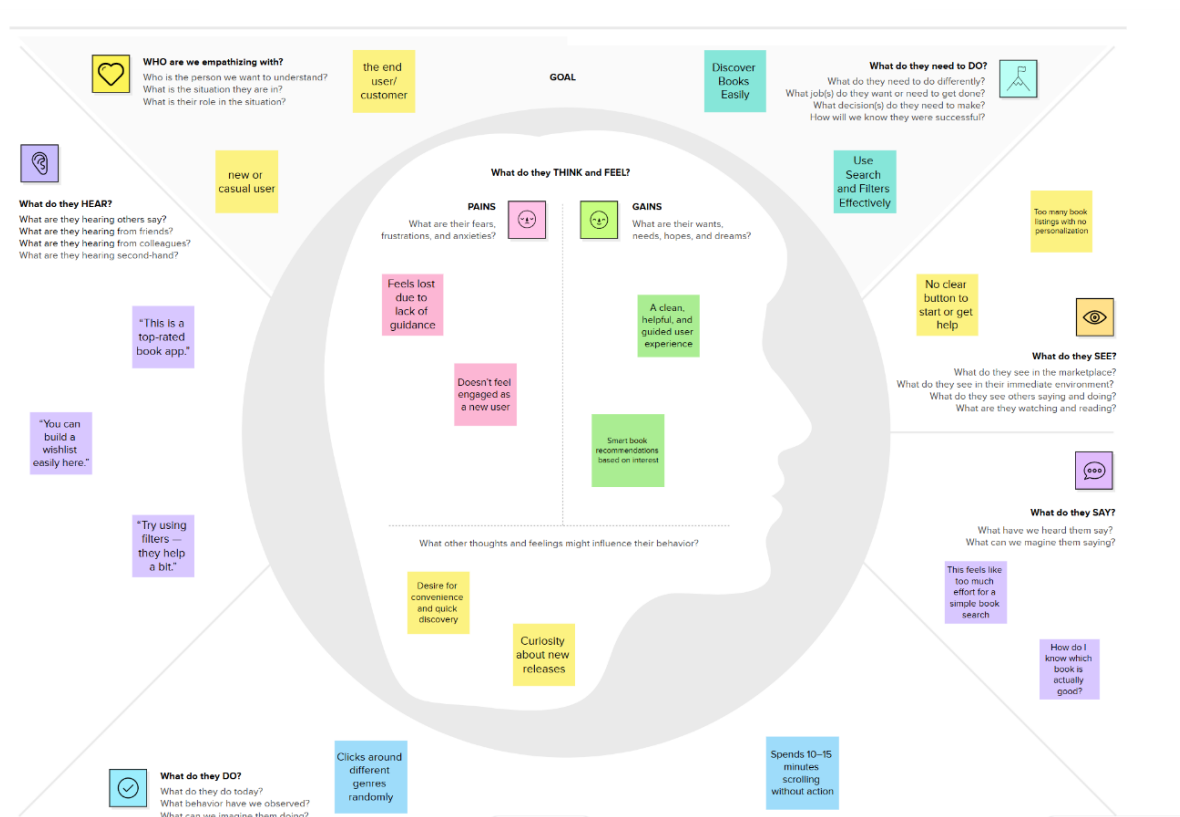
# 2. IDEATION PHASE

## 2.1 Problem Statement



| Problem Statement (PS) | I am (Customer)                  | I'm trying to  | But   | Because  | Which makes me feel                             |
|------------------------|----------------------------------|--|---|--|---|
| PS-1                   | A busy student who loves reading | Find and buy books quickly online                            | Most bookstore platforms are cluttered and time-consuming | They lack proper filtering, personalization, and smooth UX | Frustrated and overwhelmed                      |
| PS-2                   | An admin managing book inventory | Keep the online store updated with books, prices, and orders | Manual updates take time and often cause errors           | There's no efficient dashboard or automation               | Drained and less motivated to maintain the site |

## 2.2 Empathy Map Canvas



## 2.3 Brainstorming (Sticky Note Ideas)

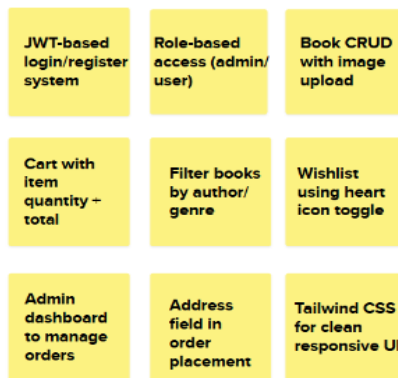
2

### Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

#### PAVANI



## 3. REQUIREMENT ANALYSIS

### 3.1 Customer Journey map

| Scenario:<br>[Existing experience through a product or service]  | Enter<br>How easy/difficult is it to use the product?               | Engage<br>How easy/difficult is it to use the product?              | Exit<br>What is the exit process like?                              | Extend<br>What is the exit process like?                            |
|--|---|---|---|---|
| <b>Experience steps</b><br>What does the process look like at the moment of use? (Typically, experience is based on steps)                   | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |
| <b>Interactions</b><br>What are the key touchpoints at each step? (Typically, interactions are based on touchpoints)                         | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |
| <b>Goals &amp; motivations</b><br>What are the goals and motivations of the user? (Typically, goals and motivations are based on user needs) | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |
| <b>Positive moments</b><br>What are the positive moments of the user? (Typically, positive moments are based on user satisfaction)           | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |
| <b>Negative moments</b><br>What are the negative moments of the user? (Typically, negative moments are based on user dissatisfaction)        | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |
| <b>Areas of opportunity</b><br>What are the areas of opportunity for the user? (Typically, areas of opportunity are based on user feedback)  | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management | 1. User registration<br>2. User login<br>3. User profile management |

### 3.2 Solution Requirement

#### Functional Requirements:

Following are the functional requirements of the proposed solution.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task)  |
|--------|-------------------------------|---|
| FR-1   | User Registration             | <ul style="list-style-type: none"><li>• Registration through Form</li><li>• Registration through Gmail</li></ul>    |
| FR-2   | User Confirmation             | <ul style="list-style-type: none"><li>• Confirmation via Email</li></ul>  |
| FR-3   | Book Browsing & Wishlist      | <ul style="list-style-type: none"><li>• Browse books by genre/author</li><li>• Add/Remove from wishlist</li></ul>   |
| FR-4   | Cart & Order Management       | <ul style="list-style-type: none"><li>• Add to cart</li><li>• Place order with delivery address</li></ul>           |
| FR-5   | Admin Panel                   | <ul style="list-style-type: none"><li>• Manage books</li><li>• Manage users</li><li>• Update order status</li></ul> |
| FR-6   | Profile Management            | <ul style="list-style-type: none"><li>• Update name/email/password</li></ul>  |
| FR-7   | Book Image Upload             | <ul style="list-style-type: none"><li>• Upload book front cover while adding/editing books</li></ul>                |

#### Non-functional Requirements:

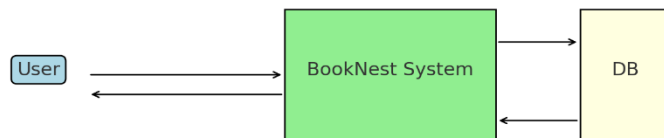
Following are the non-functional requirements of the proposed solution.

| NFR No. | Non-Functional Requirement | Description  |
|---------|----------------------------|--|
| NFR-1   | Usability                  | User-friendly UI with simple navigation and responsive design              |
| NFR-2   | Security                   | JWT authentication, role-based access, and secure password handling        |
| NFR-3   | Reliability                | Consistent functionality under different user loads and inputs             |
| NFR-4   | Performance                | Fast response times for browsing, filtering, and dashboard loading         |
| NFR-5   | Availability               | 24/7 uptime with cloud-hosted backend                                      |
| NFR-6   | Scalability                | Modular architecture to support future user growth and additional features |

### 3.3 Data Flow Diagram

#### Level-0 Data Flow Diagram-Context Level:

DFD Level-0 – BookNest



- User sends request (register, login, browse books, order, wishlist).
- System responds with book list, dashboard, confirmation
- System stores/updates data in MongoDB

#### Level 1 DFD – Functional Decomposition

DFD Level-1 – BookNest



#### Data flows:

- From user to system through login, book filters, wishlist, and cart interactions
- System reads/writes to MongoDB (user, book, order, wishlist collections)

### 3.4 Technology Stack

| Component           | Description                  | Technology Used                                |
|---------------------|------------------------------|--|
| Frontend (UI)       | Web interface for users      | React.js, Tailwind CSS                         |
| Backend (API Logic) | Business logic, routing      | Node.js, Express.js                            |
| Database            | Data storage                 | MongoDB, MongoDB Atlas                         |
| Authentication      | Login and token management   | JWT, Bcrypt.js                                 |
| File Upload         | Book images                  | Multer, Cloudinary (optional)                  |
| Hosting             | Frontend and Backend hosting | Vercel (Frontend), Render or Railway (Backend) |

## 4. PROJECT DESIGN

### 4.1 Problem Solution Fit

Problem-Solution fit canvas 2.0

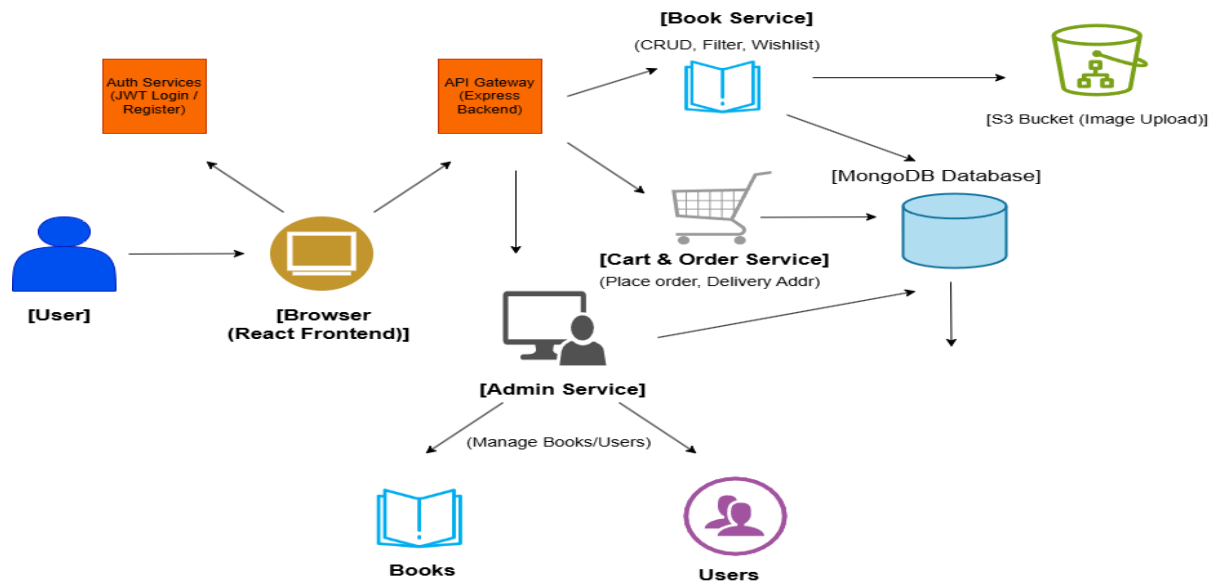
Purpose / Vision

|                         |  |  |  |                                       |
|-------------------------|--|--|--|---------------------------------------|
| Define CS, fit into     | <b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span><br>Who is your customer?<br><ul style="list-style-type: none"> <li>Busy students, working professionals, and casual readers who prefer online access to books.</li> <li>Admins or bookstore owners managing inventory and customer orders digitally.</li> </ul>  | <b>6. CUSTOMER</b> <span>CC</span><br>What constraints prevent your customers from taking action or limit their choices of solutions?<br><ul style="list-style-type: none"> <li>Limited budget or time to explore options.</li> <li>Low digital literacy (for some users).</li> <li>No account created — fear of data misuse.</li> </ul>   | <b>5. AVAILABLE SOLUTIONS</b> <span>AS</span><br>Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have?<br><ul style="list-style-type: none"> <li>Amazon, Flipkart, physical bookstores.</li> <li>Pros: Trusted brands, wide range.</li> <li>Cons: Overwhelming UX, not tailored for book-lovers, lacks wishlist or admin control.</li> </ul>  | Explore A.S.                          |
|                         | <b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span><br>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.<br><ul style="list-style-type: none"> <li>Users want to browse and buy books online easily.</li> <li>Users want to save books for later (wishlist).</li> <li>Admins need to manage books, users, and orders efficiently.</li> </ul> | <b>9. PROBLEM ROOT CAUSE</b> <span>RC</span><br>What is the real reason that this problem exists? What is the back story behind the need to do this job?<br><ul style="list-style-type: none"> <li>Digital platforms prioritize transactions over user experience.</li> <li>Admins lack dedicated dashboards tailored for bookstore operations.</li> <li>Readers have too many choices, but lack direction.</li> </ul> | <b>7. BEHAVIOUR</b> <span>BE</span><br>What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)<br><ul style="list-style-type: none"> <li>Use online platforms (Amazon) to search titles.</li> <li>Visit bookstores for personalized help.</li> <li>Ask friends for recommendations.</li> </ul>  | Focus on J&P, tap into BE, understand |
| Identify strong TR & EM | <b>3. TRIGGERS</b> <span>TR</span><br>What triggers customers to act?<br><ul style="list-style-type: none"> <li>Need to buy a book urgently (for study, interest, gifting).</li> <li>Seeing friends or peers using online book platforms.</li> </ul>   | <b>10. YOUR SOLUTION</b> <span>SL</span><br><ul style="list-style-type: none"> <li>A clean, responsive MERN-based bookstore app.</li> <li>Features: Secure login, wishlist, filters, cart, order with address, admin panel.</li> <li>Easy for users to browse and buy; powerful tools for admins to manage everything.</li> </ul>  | <b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span><br><b>8.1 ONLINE</b><br>What kind of actions do customers take online? Extract online channels from #7<br>1. Google/book searches    2. Adding to cart or wishlist<br>3. Browsing genres/filter    4. Order placement & reviews<br><b>8.2 OFFLINE</b><br>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.<br><ul style="list-style-type: none"> <li>Visiting bookstores</li> <li>Discussing with peers</li> </ul> | Extract online & offline CH of BE     |
|                         | <b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span><br>How do customers feel when they face a problem or a job and afterwards?<br><ul style="list-style-type: none"> <li>Before: Confused, overwhelmed, frustrated, unmotivated, skeptical.</li> <li>After: Empowered, satisfied, in control, more engaged with reading.</li> </ul>   |  |  |                                       |

## 4.2 Proposed Solution

| S.No. | Parameter                                    | Description   |
|-------|--|---|
| 1     | <b>Problem Statement</b>                     | In today's busy lifestyle, book lovers often struggle to find time to visit physical stores. Moreover, local availability and selection are often limited. There is a need for a centralized online platform to browse, wishlist, and purchase books conveniently, with the ability for admins to manage inventory efficiently.                   |
| 2     | <b>Idea / Solution Description</b>           | Booknest is a full-stack MERN web application that serves as an online bookstore. It allows users to register/login, browse/filter books by genre or author, add to wishlist/cart, and place orders. Admins can upload books, manage inventory, and track orders through a secured dashboard. The system is role-based, responsive, and scalable. |
| 3     | <b>Novelty / Uniqueness</b>                  | Unlike static book listings or basic e-commerce demos, Booknest includes features like image upload, dynamic wishlist toggling, admin-controlled order status updates, and JWT-based secure role access. It simulates real-world functionality typically found in large e-commerce platforms but built fully as a solo MERN stack project.        |
| 4     | <b>Social Impact / Customer Satisfaction</b> | The solution empowers readers with easier access to diverse book collections and allows book sellers/admins to manage inventory digitally. It supports learning, literacy, and digital convenience—especially helpful for students, working professionals, and rural readers.   |
| 5     | <b>Business Model (Revenue Model)</b>        | Booknest can adopt a commission-based model for authors/vendors, ad monetization, or subscription-based premium access (e.g., early releases, personalized recommendations). Affiliate partnerships with publishers can also be considered.   |
| 6     | <b>Scalability of the Solution</b>           | Built with a modular, RESTful, and stateless architecture using the MERN stack, Booknest is fully scalable. It can be extended to support multi-vendor management, mobile apps, payment integration, analytics, and even cloud-native deployment using services like Vercel, Render, or AWS.  |

### 4.3 Solution Architecture



## 5. PROJECT PLANNING & SCHEDULING

### 5.1 Project Planning

| Sprint   | Functional Requirement (Epic) | User Story Number | User Story / Task   | Story Points | Priority | Team Members  |
|----------|-------------------------------|-------------------|---|--------------|----------|---------------|
| Sprint-1 | Registration                  | USN-1             | As a user, I can register using email, password, and confirmation password. | 2            | High     | Yoga srivalli |
| Sprint-1 |                               | USN-2             | As a user, I will receive a confirmation email after registering.           | 1            | High     | Yoga srivalli |
| Sprint-1 |                               | USN-3             | As a user, I can register using Gmail authentication.                       | 2            | Medium   | Yoga srivalli |
| Sprint-1 | Login                         | USN-4             | As a user, I can log in using email and password.                           | 1            | High     | Yoga srivalli |
| Sprint-2 | Dashboard                     | USN-5             | As a user, I can view a personalized dashboard with my books and orders.    | 3            | High     | Yoga srivalli |
| Sprint-2 | Book Browsing                 | USN-6             | As a user, I can browse books and filter by genre and author.               | 3            | High     | Yoga srivalli |
| Sprint-2 | Wishlist                      | USN-7             | As a user, I can add or remove books from my wishlist using a heart icon.   | 2            | Medium   | Yoga srivalli |
| Sprint-3 | Cart & Order                  | USN-8             | As a user, I can add books to cart and place an order with address.         | 4            | High     | Yoga srivalli |
| Sprint-3 | Profile Management            | USN-9             | As a user, I can edit my profile details (name, email, password).           | 2            | Medium   | Yoga srivalli |
| Sprint-4 | Admin Dashboard               | USN-10            | As an admin, I can manage all books, users, and orders from a single panel. | 5            | High     | Yoga srivalli |
| Sprint-4 | Admin Order Update            | USN-12            | As an admin, I can update order statuses (Processing, Shipped, Delivered).  | 3            | Medium   | Yoga srivalli |
| Sprint-4 | Book Image Upload             | USN-13            | As an admin, I can upload book front cover images when adding books.        | 2            | Medium   | Yoga srivalli |



Project Tracker, Velocity & Burndown Chart: (4 Marks)

| Sprint   | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint Release Date (Actual) |
|----------|--------------------|----------|-------------------|---------------------------|---|------------------------------|
| Sprint-1 | 6                  | 6 Days   | 01 Jun 2025       | 06 Jun 2025               | 6   | 06 Jun 2025                  |
| Sprint-2 | 8                  | 6 Days   | 07 Jun 2025       | 12 Jun 2025               | 8   | 12 Jun 2025                  |
| Sprint-3 | 6                  | 6 Days   | 13 Jun 2025       | 18 Jun 2025               | 6   | 18 Jun 2025                  |
| Sprint-4 | 10                 | 6 Days   | 19 Jun 2025       | 24 Jun 2025               | 10  | 24 Jun 2025                  |

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

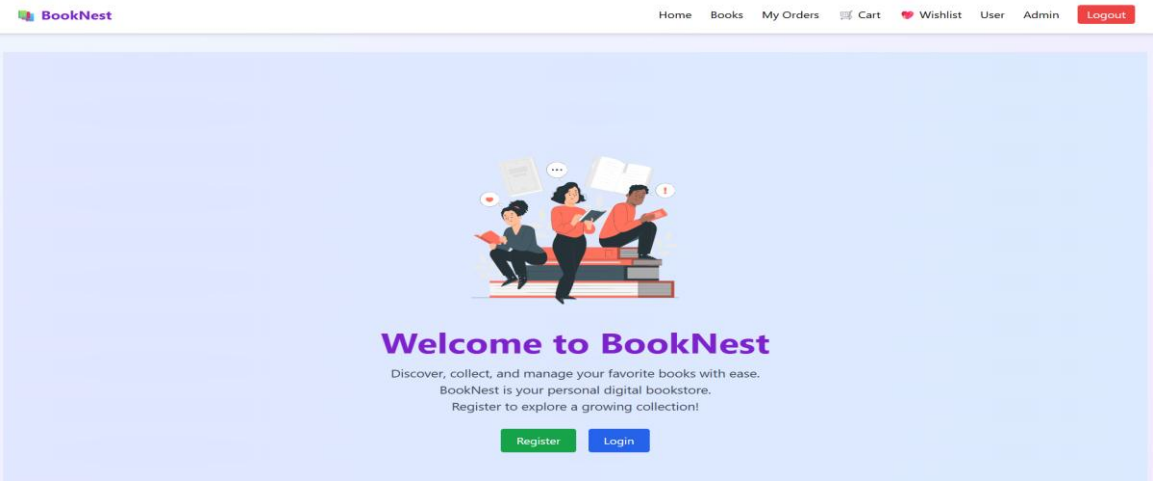
Performance testing is essential to evaluate how the BookNest application behaves under expected workloads. Although BookNest is primarily tested manually during development, key performance metrics were observed to ensure smooth user experience and backend responsiveness.

Goals of Performance Testing

- Ensure fast page load times and API response times
- Validate server handling capacity for multiple concurrent requests
- Identify and resolve performance bottlenecks
- Maintain responsiveness during user actions such as login, wishlist toggle, cart operations, and order placement

7. RESULTS

Home Page:



## Login Page:

localhost:5173/login

BookNest

Home Books My Orders Cart Wishlist User Admin Logout

### Login

Email

Password

Login

## Registration Page:

localhost:5173/register

BookNest

Home Books My Orders Cart Wishlist User Admin Logout

### Register

Name

Email

Password

Register


## Books Page:

BookNest

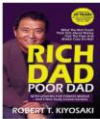
Home Books My Orders Cart Wishlist User Admin Logout

### Available Books

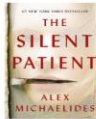
All Genres All Authors Reset Filters




**The Little Prince**  
A cosmic journey revealing life's unseen truths through a child's eyes.  
Author: Antoine de Saint-Exupéry  
Genre: Fantasy, Children's Fiction, Philosophical Allegory  
Price: ₹599  
[Add to Cart](#)




**Rich Dad Poor Dad**  
"Rich Dad Poor Dad" contrasts the financial philosophies of two fathers to underscore the importance of financial literacy, teaching readers to acquire income-generating assets rather than liabilities to achieve wealth.  
Author: Robert Kiyosaki (with Sharon Lechter)  
Genre: Personal Finance, Non-Fiction, Self-Help, Business, Investment  
Price: ₹399  
[Add to Cart](#)




**The Silent Patient**  
A psychological thriller where a woman's act of violence against her husband and her refusal to speak leads to a gripping mystery.  
Author: Alex Michaelides  
Genre: Thriller  
Price: ₹399  
[Add to Cart](#)



**Atomic Habits**

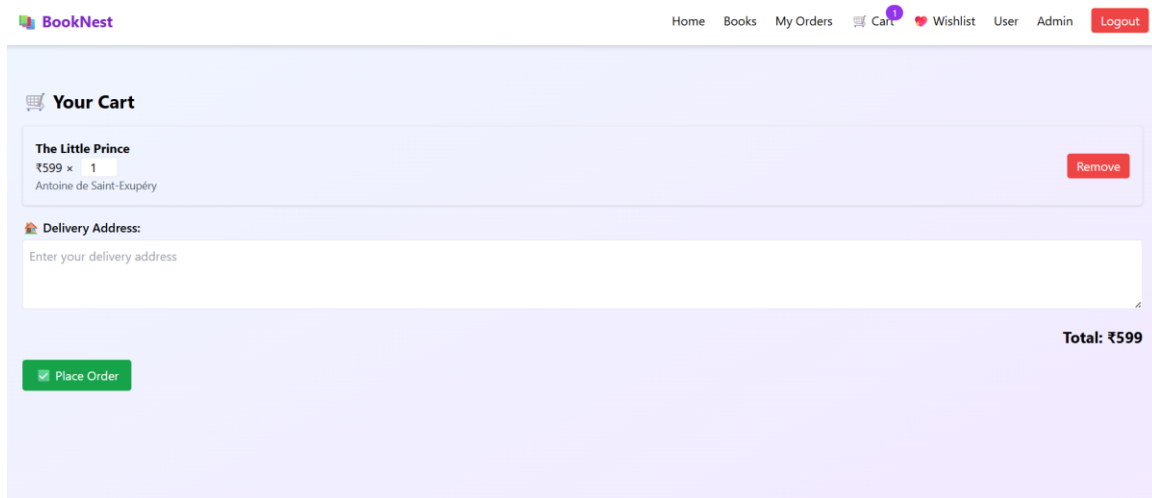


**The Alchemist**

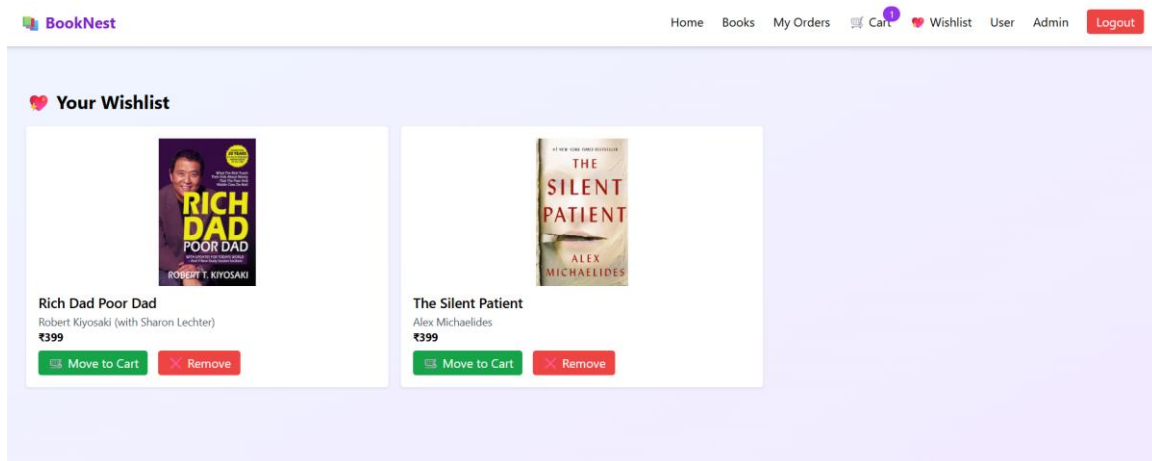


**Stephen Hawking**

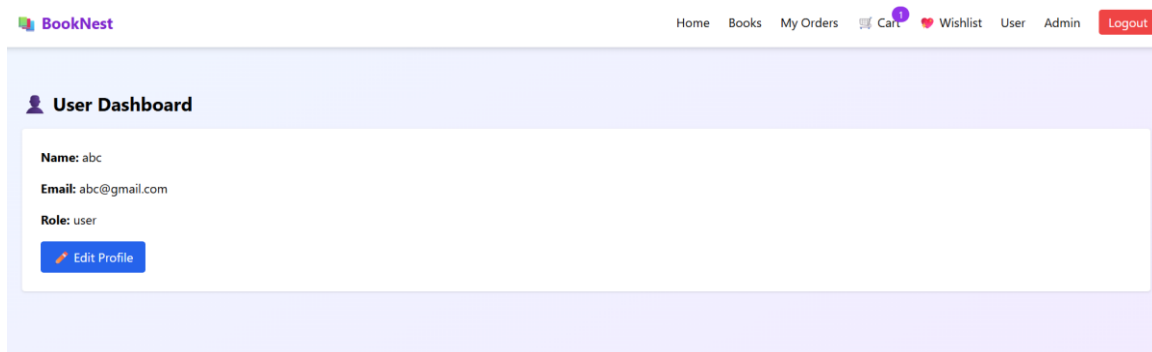
### Cart Page:



### Wishlist Page:



## Admin Dashboard:



## Your Orders Page:

BookNest

HomeBooksMy OrdersCart1WishlistUserAdminLogout

Your Orders


Order ID: 685b0ab6a66514c63665d5c1

Status: Shipped

Total Amount: ₹399

Ordered on: 25/6/2025, 1:59:42 am

Address: Rajendra Nagar, Gudivada, Andhra Pradesh, India.



Rich Dad Poor Dad

Author: Robert Kiyosaki (with Sharon Lechter)

Price: ₹399

Quantity: 1

## Admin Dashboard:

BookNest

HomeBooksMy OrdersCartWishlistUserAdminLogout

Admin Dashboard

Total Users  
3

Total Books  
10

Total Orders  
1

## Admin Viewing all the Users:

BookNest

HomeBooksMy OrdersCartWishlistUserAdminLogout

All Users

| Name  | Email             | Role  | Created   |
|-------|-------------------|-------|-----------|
| Leelu | leelu@example.com | user  | 20/6/2025 |
| Admin | admin@example.com | admin | 21/6/2025 |
| abc   | abc@gmail.com     | user  | 22/6/2025 |

## Admin can perform CRUD operations on books:

BookNest


HomeBooksMy OrdersCartWishlistUserAdminLogout

Manage Books

+ Add Book

| Title                                 | Author                                | Genre  | Price | Actions               |
|---------------------------------------|---------------------------------------|--|-------|-----------------------|
| The Little Prince                     | Antoine de Saint-Exupéry              | Fantasy, Children's Fiction, Philosophical Allegory            | ₹599  | <div>EditDelete</div> |
| Rich Dad Poor Dad                     | Robert Kiyosaki (with Sharon Lechter) | Personal Finance, Non-Fiction, Self-Help, Business, Investment | ₹399  | <div>EditDelete</div> |
| The Silent Patient                    | Alex Michaelides                      | Thriller   | ₹399  | <div>EditDelete</div> |
| Atomic Habits                         | James Clear                           | Self-Help  | ₹499  | <div>EditDelete</div> |
| The Alchemist                         | Paulo Coelho                          | Fiction  | ₹350  | <div>EditDelete</div> |
| A Brief History of Time               | Stephen Hawking                       | Science  | ₹599  | <div>EditDelete</div> |
| Wings of Fire                         | A.P.J. Abdul Kalam                    | Autobiography  | ₹299  | <div>EditDelete</div> |
| The Hobbit                            | J.R.R. Tolkien                        | Fantasy  | ₹450  | <div>EditDelete</div> |
| The Subtle Art of Not Giving a Fck*   | Mark Manson                           | Self-Help  | ₹399  | <div>EditDelete</div> |
| Harry Potter and the Sorcerer's Stone | J.K. Rowling                          | Fantasy  | ₹550  | <div>EditDelete</div> |

## Admin updating the order status:



The screenshot shows the BookNest Admin Panel. At the top, there's a navigation bar with links: Home, Books, My Orders, Cart, Wishlist, User, Admin, and a Logout button. Below the navigation bar, the 'All Orders' section is displayed. It contains a table with the following data:

| Order ID                 | User | Books                 | Total Price | Address  | Date                  | Status  |
|--------------------------|------|-----------------------|-------------|--|-----------------------|---------|
| 685b0ab6a66514c63665d5c1 | abc  | Rich Dad Poor Dad × 1 | ₹399        | Rajendra Nagar, Gudivada, Andhra Pradesh, India. | 25/6/2025, 1:59:42 am | Shipped |

A dropdown menu is open for the 'Shipped' status of the first order, showing the following options: Processing, Shipped, Delivered, and Cancelled.

## 8. ADVANTAGES & DISADVANTAGES

### Advantages

- Full-Stack Control – Built entirely using the MERN stack, allowing unified JavaScript codebase across client and server.
- Modular Design – Separation of frontend, backend, and database ensures maintainability and scalability.
- Secure Authentication – Uses JWT for secure user login, with role-based access for admin functionalities.
- Responsive UI – Built with Tailwind CSS and React, the interface is fast, responsive, and user-friendly across devices.
- Wishlist and Cart Functionality – Enhances user experience by allowing item tracking and personalized browsing.
- Admin Panel – Allows book and user management, and order status tracking by privileged users.
- NoSQL Flexibility – MongoDB allows easy schema evolution and scaling of collections like books and orders.

## **Disadvantages**

- No Real Payment Integration – Orders are placed without actual payment gateway support.
- Admin Role Manual Setup – Admin accounts must be assigned manually in the database.
- Token Expiry Handling Not Implemented – JWT refresh tokens are not in place, requiring users to log in again after expiration.
- No Mobile App – The solution is currently web-only, limiting reach for mobile-first users.
- Limited Testing Coverage – Lacks automated unit or end-to-end test cases, making regression testing manual.

## **9. CONCLUSION**

Booknest is a fully functional, scalable, and secure online bookstore platform developed using the MERN stack. It offers a seamless user experience, allowing customers to browse, wishlist, and purchase books, while providing admins with full control over content, users, and orders.

The project demonstrates complete ownership over all layers of web application development—from system design and frontend development to backend API creation and database integration. The platform can serve as a foundation for real-world deployment and further enhancements such as payment integration, mobile support, and cloud deployment.

Through Booknest, the goal of delivering a modern, responsive, and modular online bookstore solution has been successfully achieved.

## **10. FUTURE SCOPE**

The Booknest project can be significantly expanded and improved in the following ways:

### **Online Payment Gateway**

- Integrate Razorpay or Stripe to support secure real-time payments.

### **Mobile Application (React Native)**

- Develop a cross-platform app for better mobile usability.

### **Review and Rating System**

- Allow users to review and rate books, improving decision-making and engagement.

### **Advanced Search and Filters**

- Add autocomplete search, price range filters, and personalized recommendations.

### **Email/SMS Notifications**

- Notify users on order confirmation, shipping, and delivery updates.

### **Role Management UI**

- Enable admin role assignment via the frontend rather than manual DB updates.

### **Cloud Deployment**

- Host the frontend (Vercel/Netlify) and backend (Render/Heroku) for public access.

### **Analytics Dashboard for Admins**

- Visualize sales, inventory, and user metrics using charts and reports.

### **AI-Based Recommendations (Future)**

- Suggest books to users based on purchase history or interests using ML.

### **Multilingual Support**

- Add regional language support to improve accessibility.

## **11. APPENDIX**

### **Source Code:**

#### **Home.jsx:**

```
// src/pages/Home.jsx
```

```
import { Link } from "react-router-dom";
```

```
const Home = () => {
```

```
return (
```

```
  <div className="flex items-center justify-center min-h-screen bg-gradient-to-r from-indigo-100 to-blue-100">
```

```
    <div className="text-center px-4">
```

```
      
```

```
    <h1 className="text-4xl md:text-5xl font-bold text-purple-700 mb-4">
```

```
      Welcome to BookNest
```

```
    </h1>
```

```
    <p className="text-lg text-gray-700 max-w-xl mx-auto mb-6">
```

```
      Discover, collect, and manage your favorite books with ease. <br />
```

```
      BookNest is your personal digital bookstore. <br />
```

```
      Register to explore a growing collection!
```

```
    </p>
```

```
    <div className="flex justify-center gap-4">
```

```
      <Link
```

```
        to="/register"
```

```
        className="bg-green-600 hover:bg-green-700 text-white px-5 py-2 rounded"
```

```
      >
```

```
        Register
```

```
      </Link>
```

```
    <Link
```



```
        to="/login"

        className="bg-blue-600 hover:bg-blue-700 text-white px-5 py-2 rounded"

    >

        Login

    </Link>

</div>

</div>

</div>

);

};
```

export default Home;

**NOTE:** Entire source code files are uploaded in the GitHub repository

**GitHub Link:** <https://github.com/Yogasrivalli/BookNest-Where-Stories-Nestle>

**Demo**

**Link:**[https://drive.google.com/file/d/1QqQpnmea\\_b1MMM1O40HeEfmi92c9vscj/view?usp=sharing](https://drive.google.com/file/d/1QqQpnmea_b1MMM1O40HeEfmi92c9vscj/view?usp=sharing)