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.
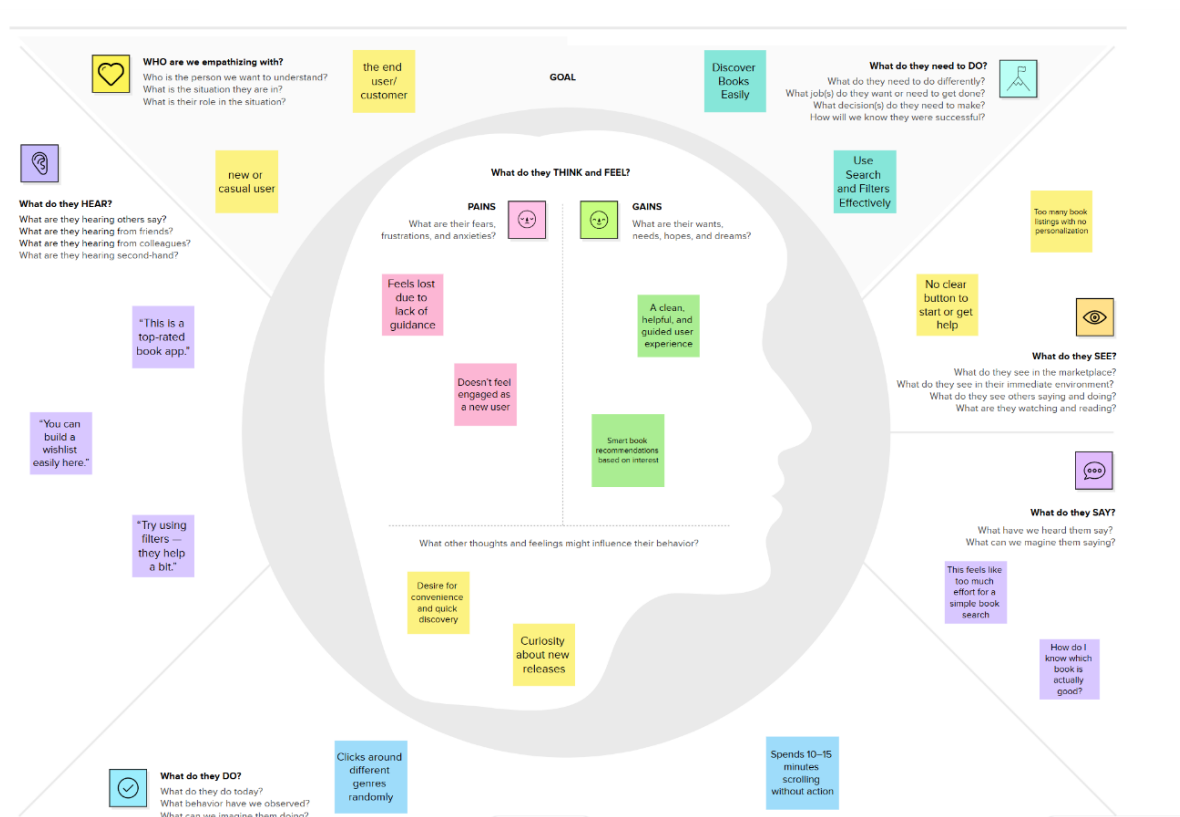
2. IDEATION PHASE

2.1 Problem Statement
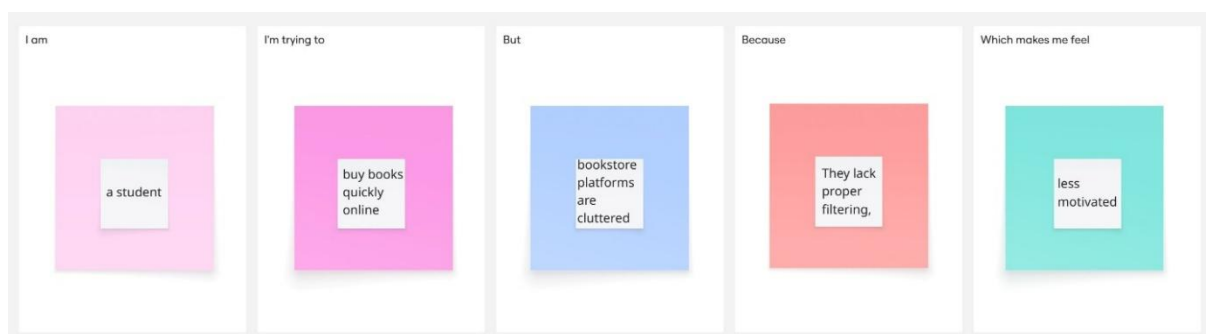
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	A busy student who loves reading	Find and buy books quickly online	Most bookstore platforms are cluttered and time-consuming	They lack proper filtering, personalization, and smooth UX	Frustrated and overwhelmed
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

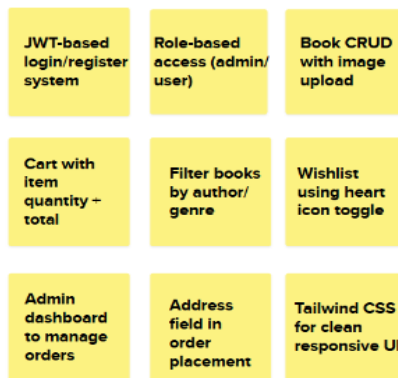## 2.2 Empathy Map Canvas

2.3 Brainstorming (Sticky Note Ideas)

3. REQUIREMENT ANALYSIS

3.1 Customer Journey map

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	• Registration through Form • Registration through Gmail
FR-2	User Confirmation	• Confirmation via Email
FR-3	Book Browsing & Wishlist	• Browse books by genre/author • Add/Remove from wishlist
FR-4	Cart & Order Management	• Add to cart • Place order with delivery address
FR-5	Admin Panel	• Manage books • Manage users • Update order status
FR-6	Profile Management	• Update name/email/password
FR-7	Book Image Upload	• Upload book front cover while adding/editing books

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
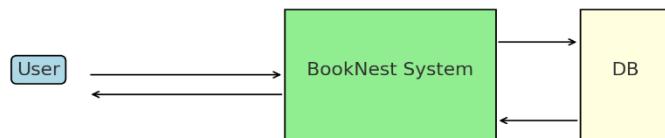
- User sends request (register, login, browse books, order, wishlist).

- 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

4.2 Proposed Solution

S.No.	Parameter	Description
1	**Problem Statement**	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	**Idea / Solution Description**	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	**Novelty / Uniqueness**	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	**Social Impact / Customer Satisfaction**	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	**Business Model (Revenue Model)**	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	**Scalability of the Solution**	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
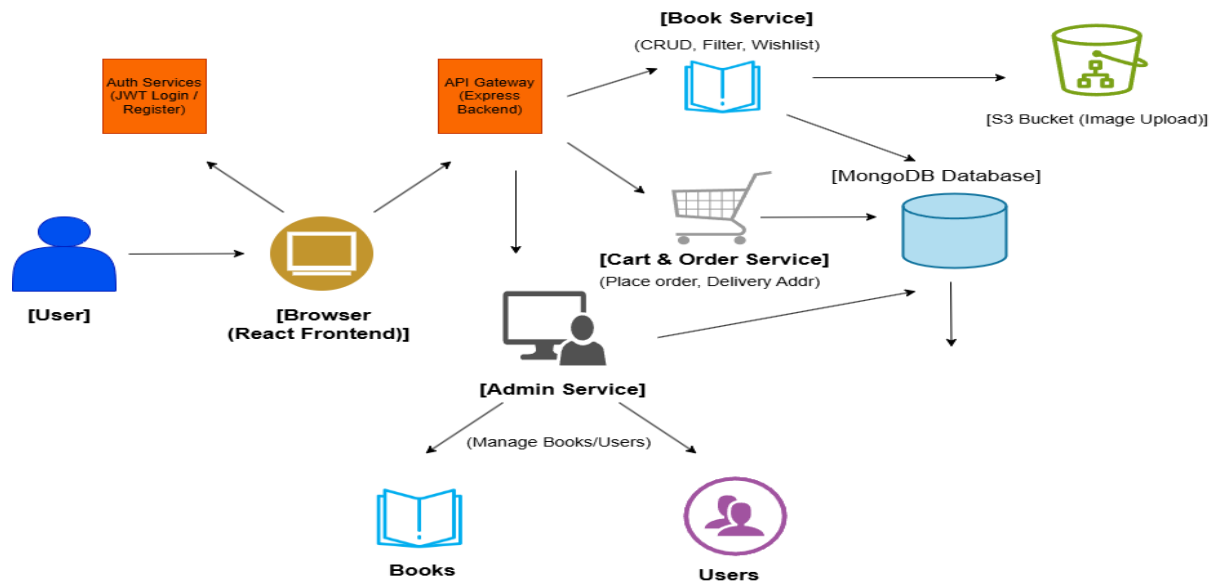
5. PROJECT PLANNING & SCHEDULING

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

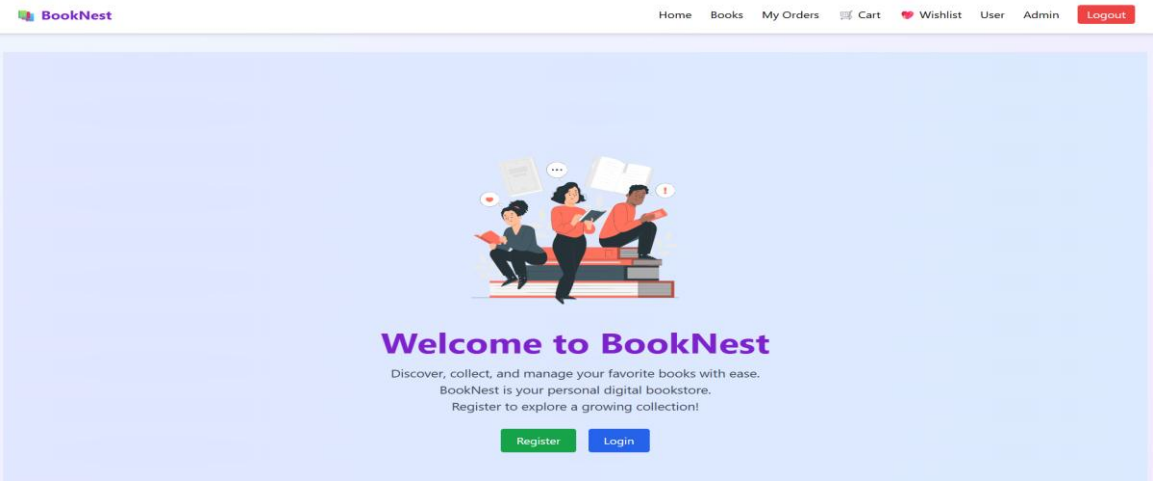# 6. FUNCTIONAL AND PERFORMANCE TESTING

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:

Registration Page:

Books Page:

Cart Page:

Wishlist Page:

Admin Dashboard:

Your Orders Page:

Admin Dashboard:

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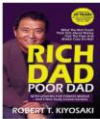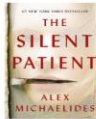
Title	Author	Genre	Price	Actions
The Little Prince	Antoine de Saint-Exupéry	Fantasy, Children's Fiction, Philosophical Allegory	₹599	Edit Delete
Rich Dad Poor Dad	Robert Kiyosaki (with Sharon Lechter)	Personal Finance, Non-Fiction, Self-Help, Business, Investment	₹399	Edit Delete
The Silent Patient	Alex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

Admin updating the order status:

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 ▾

Dropdown options: Processing / Shipped / Delivered / 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 = () => {
```

```jsx
  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">

        <img

          src="/reading-book.svg"

          alt="Reading Book"

          className="w-72 mx-auto mb-6"

        />

        <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**