

# **Project Documentation: Quasar Books —**

## **MERN Stack Online Book Store**

### **1. Introduction**

**Project Title:** Quasar Books

**Team Members:** **SWTID1744137899**

- Anvesha Rathore
- Diya Mittal
- Shreya Sahu
- Suhani Singh

**Links:**

- To all the project phase documents: [git-hub folder](#)
- **GitHub:** <https://github.com/shreying/QMERN-store>

### **2. Project Overview**

#### **Purpose**

Quasar Books is an online bookstore providing users with a seamless platform to browse, buy, and manage books. Admins can manage inventory, users, and orders efficiently.

You can sign up, log in, and browse books by genre, author, or popularity. When you find something you like, just add it to your cart and place the order. You'll get a confirmation, and you can always check your order history later. There's also an admin panel where books can be added or updated, inventory is managed, and orders are tracked in real time.

#### **Key Features**

- **User Roles**
  - *Users*: Browse, add to cart/favorites, place orders
  - *Admins*: Add/edit/delete books, manage users and orders
- **Authentication**: Secure login/signup with JWT
- **Dynamic UI**: Real-time cart and favorites updates
- **Responsive Design**: Optimized for all devices

## 3. Architecture

### Frontend (React)

- **Framework**: React with Vite
- **State Management**: Redux Toolkit
- **Styling**: Tailwind CSS
- **Routing**: React Router
- **API Integration**: Axios

### Backend (Node.js + Express)

- **Framework**: Express.js
- **Authentication**: JWT + Bcrypt
- **Routing**: RESTful API for users, books, and orders

### Database (MongoDB)

- **Collections**:
  - *Users*: Credentials, roles, cart, favorites
  - *Books*: Title, author, price, genre, image, etc

- *Orders*: Book IDs, user ID, total, status

## 4. Setup Instructions

### Prerequisites

- Node.js v16+
- MongoDB Atlas or local instance
- Git

### Installation

#### Clone the Repo

```
1 git clone https://github.com/shreying/QMERN-store.git
2 cd QMERN-store
```

#### Backend Setup

```
1 cd backend
2 npm install
3 cp .env.example .env # Add MONGO_URI, JWT_SECRET
4 npm start
5 nodemon app.js
```

#### Frontend Setup

```
1 cd ../frontend
2 npm install
3 npm run dev
```

## 5. Folder Structure

### Frontend

*src/*

```
|── assets/          # Static assets (images, etc.)  
|── components/    # UI Components  
|── pages/          # Route pages  
|── store/          # Redux slices  
|── App.css         # Global styles  
|── App.jsx         # App Router  
|── index.css       # Tailwind base styles  
|── main.jsx        # Entry Point
```

### Backend

*backend/*

```
|── conn/           # DB Connection Logic  
|   └── conn.js  
|── models/         # Mongoose Schemas (User, Book, Order)  
|   ├── book.js  
|   ├── order.js  
|   └── user.js  
|── routes/         # Express Routes  
|   ├── book.js  
|   ├── cart.js  
|   ├── favourite.js  
|   ├── order.js  
|   ├── user.js  
|   └── userAuth.js  
|── app.js          # Server Setup  
|── .env             # Environment Config
```

```
└── package.json    # Backend Dependencies
```

## 6. Running the Application

- **Frontend:**

`npm run dev` (Runs on <http://localhost:5173>)

- **Backend:**

`npm start` (Runs on <http://localhost:1000>)

## 7. API Documentation

### Base URL

`http://localhost:<PORT>/api/v1`

### Endpoint Summary Table

Category	Method	Endpoint	Description
Auth	POST	/sign-up	Create a new user
Auth	POST	/sign-in	User login and JWT retrieval
User	GET	/get-user-information	Get user info
User	PUT	/update-address	Update user address

Books	POST	/add-book	Add book (Admin)
Books	PUT	/update-book/:id	Update book (Admin)
Books	DELETE	/delete-book/:id	Delete book (Admin)
Books	GET	/get-book	Get all books
Books	GET	/get-recent-books	Get recently added books
Books	GET	/get-book-by-id/:id	Get book by ID
Cart	PUT	/add-to-cart	Add book to cart
Cart	PUT	/remove-from-cart/:bookid	Remove book from cart
Cart	GET	/get-user-cart	Get user's cart
Favourites	PUT	/add-book-to-	Add book to

		avourite	favourites
Favourites	PUT	/remove-book-from-favourite	Remove book from favourites
Favourites	GET	/get-favourite-books	Get user's favourite books
Orders	POST	/place-order	Place a new order
Orders	GET	/get-order-history	Get user's order history
Orders (Admin)	GET	/get-all-orders	Get all orders (Admin)
Orders (Admin)	PUT	/update-order-status/:id	Update order status (Admin)

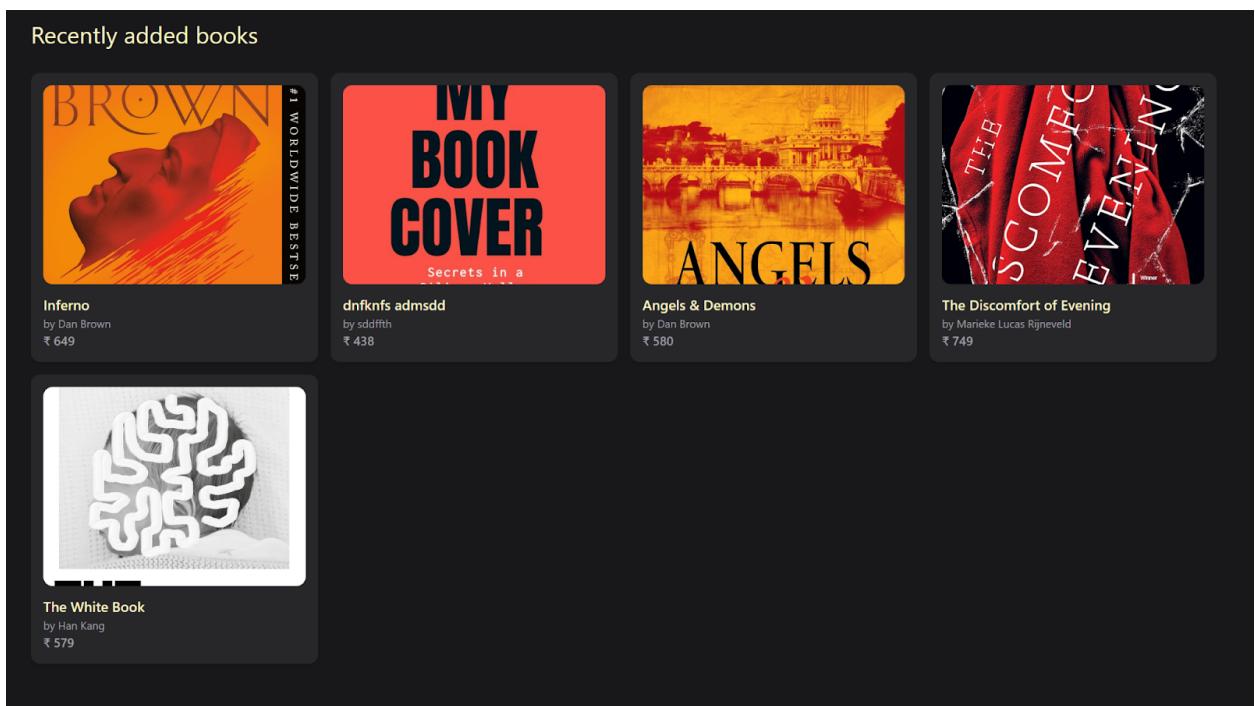
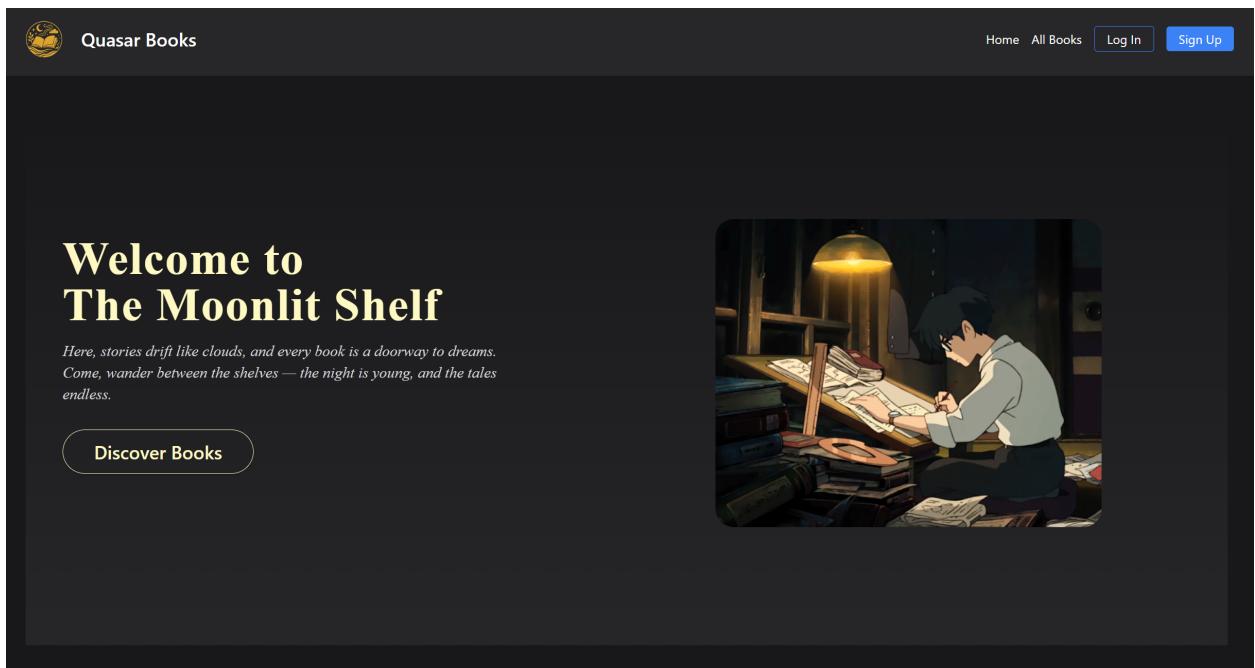
Full API docs available in [api.md](#)

## 8. Authentication

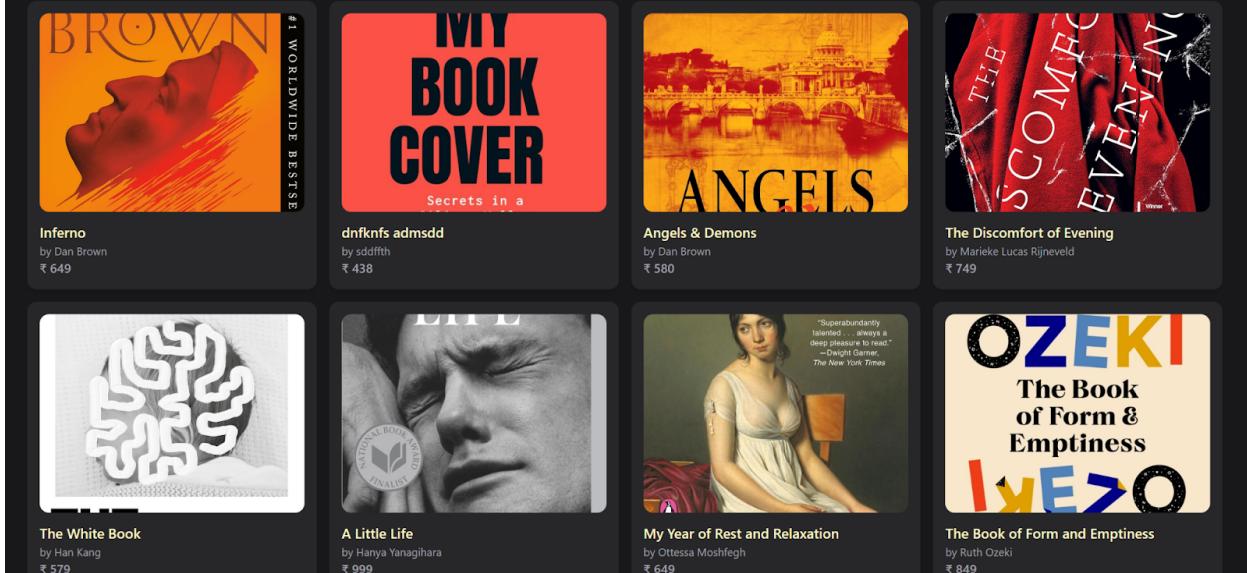
- JWT tokens issued on login (valid for 1hour)
- Protected routes via middleware
- Role-based access control (RBAC) for admin routes

## 9. User Interface

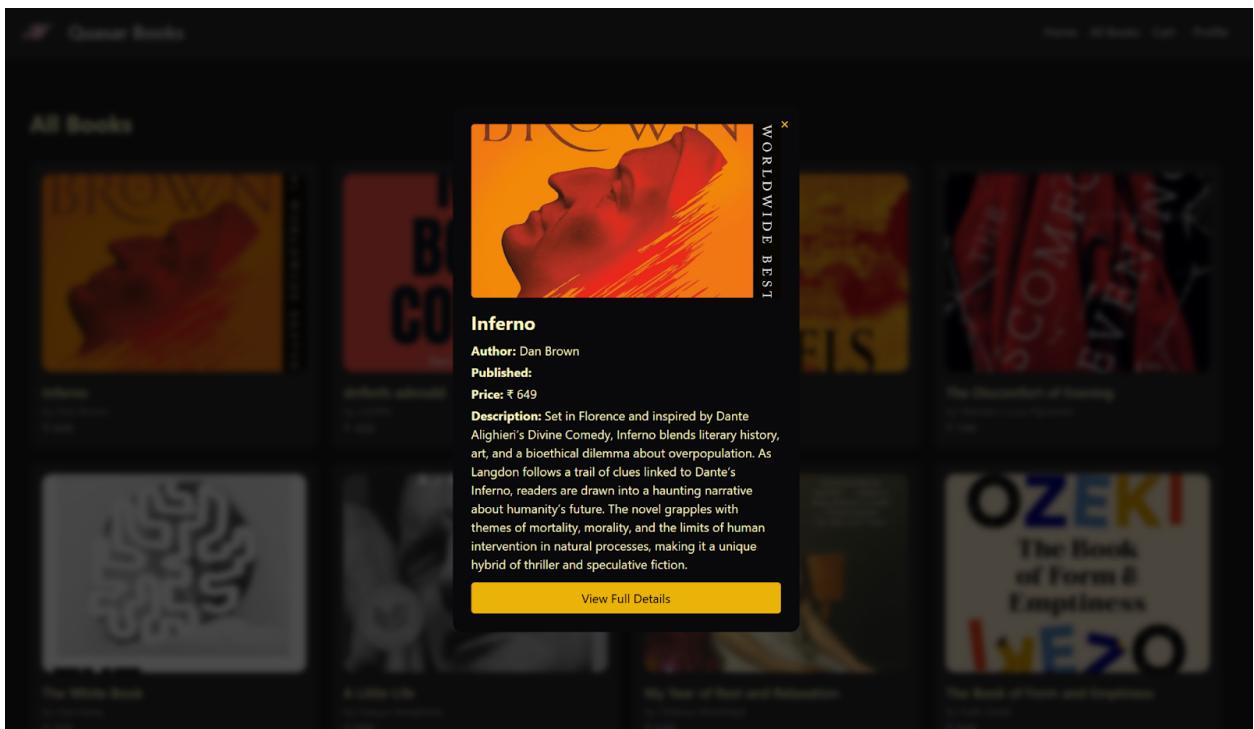
- **Homepage:** Browse books



## All Books



- **Book Details Page**



The screenshot shows a book detail page for "Angels & Demons" by Dan Brown. At the top left is the Quasar Books logo. The main image is the book cover, which features a yellow-tinted view of the Vatican City. The title "DAN BROWN" is at the top, and "ANGELS & DEMONS" is in large letters below it. To the right of the cover are two circular icons: a red heart and a blue shopping cart. The title "Angels & Demons" is displayed prominently in bold white text. Below it, the author "by Dan Brown" is written. A detailed plot summary follows: "Before The Da Vinci Code, there was Angels & Demons—where Robert Langdon uncovers a plot by a secret society, the Illuminati, to destroy the Vatican using antimatter. This fast-paced thriller intertwines modern science with ancient rituals, exploring the age-old clash between religion and scientific discovery. It's an electrifying mix of action and philosophical questioning, challenging the role of faith in an increasingly scientific world." Language information indicates "English". The price is listed as "Price: ₹580.00".

- **Book cart, favorites and order history:** orders for user

The screenshot shows the "Your Cart" page. The Quasar Books logo is at the top left. The main heading "Your Cart" is centered above a list of items. The first item is "Heaven" by Haruki Murakami, shown with a small thumbnail image of the book cover. The details "Heaven", "Price: 699 Rs", and "Quantity:" are listed next to it. On the far right of this row is a red "Remove" link. Below the cart list is a summary box containing "Total Quantity: 1" and "Total Price: 699 Rs". To the right of this summary is a blue "Proceed to Checkout" button.

Quasar Books



shsh  
shsh@gmail.com



**Angels & Demons**  
by Dan Brown  
₹ 580

[Remove from Favourites](#)

Favourites

Order History

Settings

Home All Books Cart Profile

Quasar Books



shsh  
shsh@gmail.com

## Order History



**Angels & Demons**  
Author: Dan Brown  
Price: **580 Rs**  
Ordered on: 13 April 2025  
Status: **Order Delivered**



**Angels & Demons**  
Author: Dan Brown  
Price: **580 Rs**  
Ordered on: 13 April 2025  
Status: **Order Placed**



**The Discomfort of Evening**  
Author: Marieke Lucas Rijneveld  
Price: **749 Rs**  
Ordered on: 10 April 2025  
Status: **Order Placed**

Favourites

Order History

Settings

Home All Books Cart Profile

- **Admin Panel: Inventory & Order Management**

**All Orders**

Sr.	Book	Title	User	Price	Status
1		dnlknfs admsdd		₹438	Order Placed Order Placed ✓
2		Angels & Demons		₹580	Order Delivered Order Delivered ✓
3		Angels & Demons		₹580	Order Placed Order Placed ✓
4		dnlknfs admsdd		₹438	Order Shipped Order Shipped ✓
5		Angels & Demons		₹580	Order Placed Order Placed ✓
6		dnlknfs admsdd		₹438	Canceled Canceled ✓
7		dnlknfs admsdd		₹438	Order Placed

Order status updated to "Canceled"!

**Add a New Book**

**Image URL**  
https://example.com/image.jpg

**Title**  
Enter book title

**Author**  
Enter author name

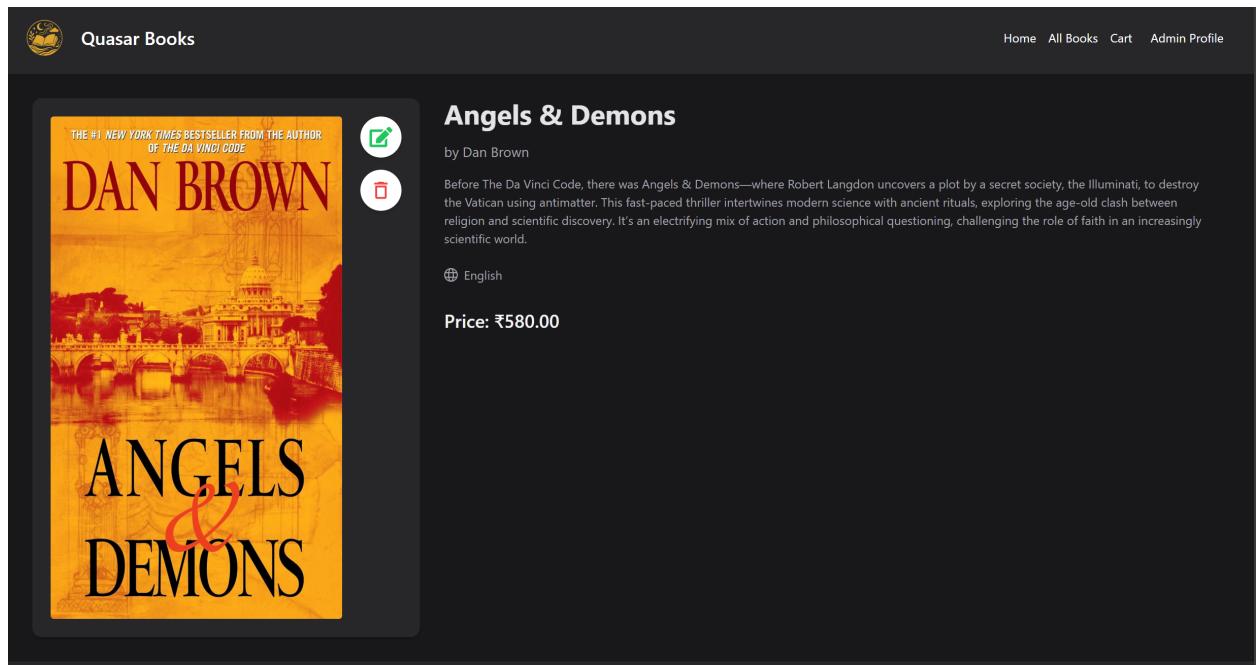
**Price (₹)**  
Enter book price

**Description**  
Enter book description

**Language**  
Enter book language

**Add Book**

Log Out



## 10. Testing

- Manual testing for user/admin flows
- API tested with Postman
- Planned: Jest + React Testing Library for frontend

## 11. Known Issues

- Cart occasionally desyncs with rapid actions
- Minor responsive glitches on very small screens

## 12. Future Enhancements

- Add search and filtering (title, author)
- Integrate Stripe/PayPal for payments
- Add ratings, reviews, and user profiles