BookNest – MERN Stack Book Store Application

1. Introduction

BookNest is a full-stack web application developed using the **MERN Stack. (MongoDB, Express.js, React.js, Node.js)**. The system provides a modern online bookstore platform where users can browse books, sellers can manage inventory, and administrators can monitor system activity.

The project aims to digitize the traditional bookstore experience by offering secure authentication, attractive UI, and efficient book management.

2. Objectives

- Develop a role-based bookstore platform (User, Seller, Admin).
- Provide secure authentication and authorization.
- Enable online book browsing, cart management, and ordering.
- Allow sellers to manage inventory.
- Provide admins with order monitoring and platform control.

3. Technologies Used

Frontend

- React.js
- Vite
- Bootstrap & React-Bootstrap
- Axios
- React Router DOM
- Framer Motion (animations)
- React Icons
- TSParticles (floating particles background)

Backend

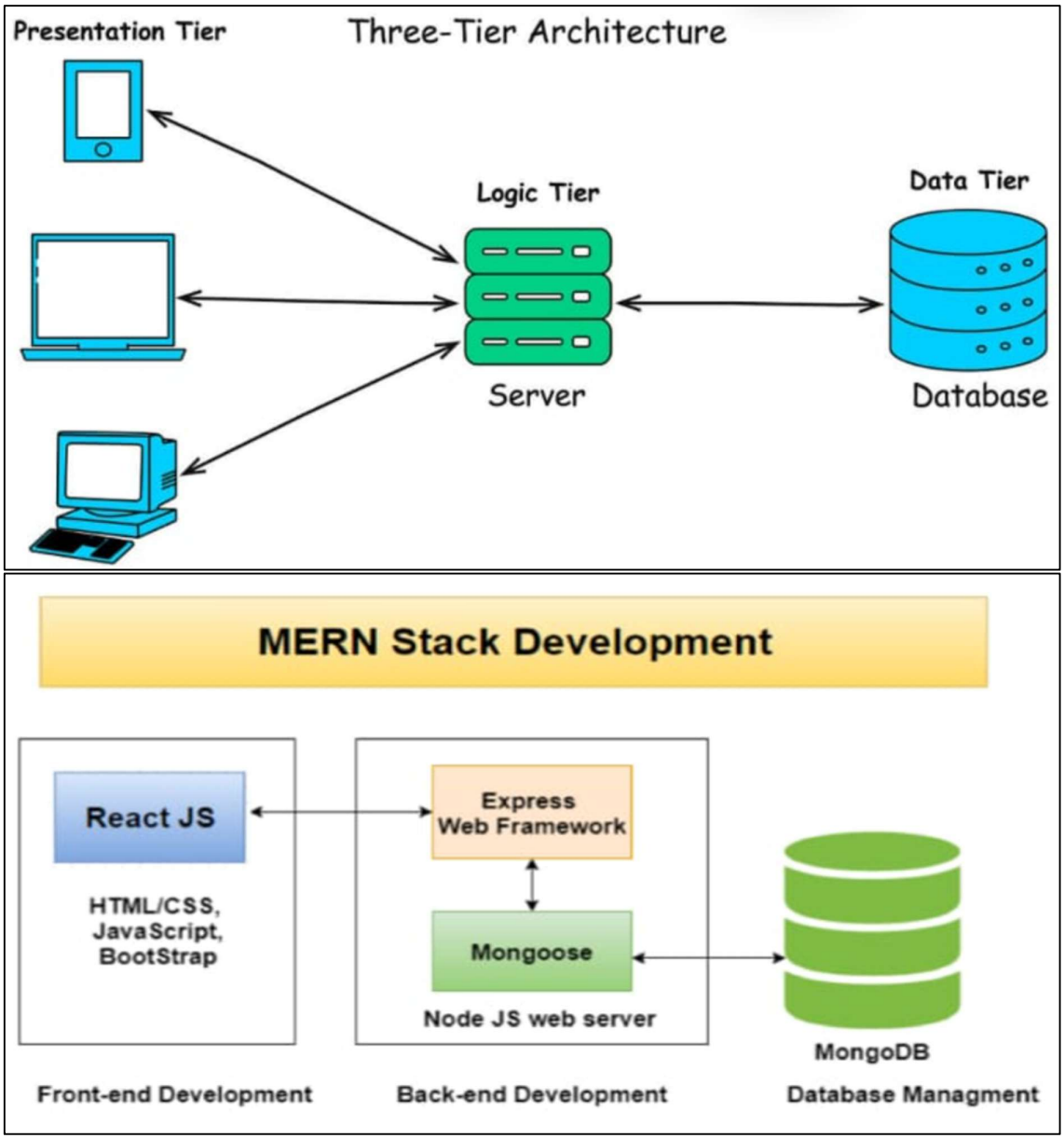
- Node.js
- Express.js
- JWT Authentication
- bcryptjs (password hashing)
- CORS

- dotenv

Database

- MongoDB
- Mongoose ODM

4. System Architecture



Architecture Diagram

Frontend

- Developed using React.
- Component-based architecture.
- Responsive dashboard layout.
- Role-based navigation.

Backend

- REST API built using Express.
- Route separation for:
    - Authentication
    - Books
    - Orders

Database

MongoDB stores:

- Users
- Books
- Orders

5. User Roles

User

- Signup & Login
- Browse books
- Add to cart
- Checkout orders
- View UI dashboard

Seller

- Login
- Add new books
- Manage inventory

Admin

- Login
- View orders
- Monitor system

6. Authentication System

- JWT token based authentication.
- Passwords encrypted using bcrypt.
- Protected routes ensure secure access.

7. Key Features

✓ User Authentication

- Login & Signup system.
- Role-based access control.

✓ Book Listings

- Display books with:
  - Title
  - Author
  - Price
  - Description

✓ Cart System

- Add books to cart.
- Dynamic price calculation.

✓ Checkout & Orders

- Order stored in MongoDB.
- Admin can view all orders.

✓ Seller Dashboard

- Add books through UI form.

✓ Admin Dashboard

- View order list and totals.

✓ Modern UI Features

- Attractive theme
- Animated transitions
- Floating particles background
- Sidebar navigation

- Responsive layout

8. Project Folder Structure

```
booknest/
│
├── client/            (React Frontend)
│   ├── src/
│   │   ├── components/
│   │   ├── pages/
│   │   ├── App.jsx
│   │   └── main.jsx
│
├── server/            (Node Backend)
│   ├── models/
│   ├── routes/
│   ├── middleware/
│   ├── index.js
│   └── .env
```

9. Installation & Setup

1 Clone or Extract Project

```
booknest/
```

2 Backend Setup

```
cd server
npm install
npx nodemon index.js
```

3 Frontend Setup

```
cd client
npm install
npm run dev
```

4 MongoDB

Start MongoDB server:

```
mongod
```

10. API Endpoints

Auth

- POST `/api/auth/register`
- POST `/api/auth/login`

Books

- GET `/api/books`
- POST `/api/books`

Orders

- POST `/api/orders`
- GET `/api/orders`

11. Database Models

User Model

- name
- email
- password
- role

Book Model

- title
- author
- price
- description
- stock

Order Model

- items
- total
- created At

12. UI Enhancements

- Bootstrap dark-light blended theme
- Animated page transitions
- Sidebar dashboard design
- Floating particle background

- Icon-based navigation

13. Future Enhancements

- Payment gateway integration
- Search & filters
- Book ratings & reviews
- Analytics dashboard
- Deployment to cloud
- Email verification

14. Conclusion

BookNest successfully demonstrates a full-stack MERN implementation combining modern UI design with secure backend architecture.
The project provides a complete online bookstore solution with role-based dashboards and scalable architecture.