

NAAN MUDHALVAN
MERN STACK
FUNDAMENTALS
ONLINE BOOK STORE
0004-Madras Institute of Technology

TEAM MEMBERS:

Devadharshini B – 2021506015 (80F25D1CD7080B8A35A46F2602A90258)

Dhanushri S – 2021506017 (DA79BDE488BE71A89164F7052B6F0C59)

Jaikiran R - 2021506029 (106211811A9ACE9BA91430944BCD4835)

Padmapriya S – 2021506104 (5CCB063A8923BA09E5348F9BE7FB7A23)

INTRODUCTION:

This report details the development of an online bookstore application built using HTML, CSS, JavaScript, React, Node.js, Express.js, and MongoDB. The application provides users with a seamless online book-shopping experience, enabling them to browse books, manage wishlists and carts, filter and sort products, securely log in to their accounts, and process payments. It features a responsive design to ensure compatibility across various devices, enhancing accessibility.

TECHNICAL IMPLEMENTATION:

FRONTEND:

HTML:

- Structured web pages such as the homepage, product listing page, cart, and login page.
- Integrated HTML forms for user input and interaction.

CSS:

- Styled the web pages for a visually appealing and intuitive interface.
- Used CSS for layout design, colors, fonts, and spacing to enhance usability.

JAVASCRIPT:

- Implemented dynamic interactions on the website, including:
 - Form validation.
 - AJAX requests for real-time updates.
 - DOM manipulation to update the UI dynamically.
 - Event listeners for user actions like button clicks.
- Developed client-side search functionality to filter books by name or author.

REACT:

- Created reusable components such as product cards, filters, and navigation bars.
- Used React hooks for state management and handling component lifecycle events.
- Implemented routing for seamless navigation between pages.
- Developed a custom toast component for success, error, warning, and information messages.

BACKEND:

MongoDB:

- Stored data for books, user accounts, wishlists, cart items, and orders.
- Designed efficient schemas for data storage and retrieval.
- Implemented CRUD operations to handle database interactions.

Node.js and Express.js:

- Set up a backend server to handle API requests and process user data.
- Implemented authentication and authorization mechanisms using JWT for secure user sessions.
- Developed RESTful APIs to manage product listings, user accounts, orders, and payments.

Deployment:

- Deployed the backend server on Vercel for scalability and easy access.

FEATURES:

Homepage:

- Displays curated collections, including categories like New Arrivals.
- Features a search bar for quick navigation by book name or author.

Product Listing:

- Displays books with detailed descriptions, images, prices, and ratings.
- Offers sorting by price (low-to-high, high-to-low) and filtering by:
 - Price range (minimum and maximum).
 - Book genres.
 - Average product rating.
 - Stock availability.
 - Fast delivery options.
- Allows clearing all applied filters.

Wishlist Management:

- Enables users to add or remove books from their wish list.

Cart Management:

- Allows users to:
 - Add books to the cart.
 - Update quantities or remove books.
 - Move items to the wish list from the cart.
 - Apply coupon codes.

Single Product Page:

- Displays detailed information about a book and allows adding it to the cart or wish list.

Checkout and Payment:

- Integrates Razorpay for secure payment processing.
- Provides an order summary with details of purchased items.

Orders Page:

- Displays users' past orders and their details.

Custom Toast Notifications:

- Includes four types of notifications: Success, Error, Warning, and Information.

Pagination:

- Handles large product lists by displaying books in a paginated format.

FUTURE ENHANCEMENTS:**Mobile Optimization:**

- Ensure optimal usability on mobile devices.

User Reviews and Ratings:

- Enable users to leave feedback and rate books.

Personalized Recommendations:

- Implement a recommendation engine based on user purchase history and preferences.

Push Notifications:

- Notify users about order updates, promotions, and new arrivals.

Advanced Search:

- Add more robust search options with filters for price, genre, and ratings.

CONCLUSION:

This online bookstore application effectively combines HTML, CSS, JavaScript, React, Node.js, Express.js, and MongoDB to deliver a feature-rich and user-friendly shopping platform. It offers functionalities such as user authentication, product browsing, cart and wishlist management, secure payments, and order history tracking. By incorporating planned enhancements, this application can elevate user satisfaction and expand its reach.