

B.Tech. CSE (III YEAR – V SEM) (2025-2026)

DEPARTMENT OF COMPUTER ENGINEERING & APPLICATIONS



GLA University
17km Stone, NH-19, Mathura-Delhi Road P.O.
Chaumuhan, Mathura - 281406 (Uttar Pradesh) India

Project Title: ShopSmart AI
(AI Powered E-Commerce Website)

Team Member 1: Rahul Gupta	UR: 2315800067
Team Member 2: Harsh Gupta	UR: 2315800032
Team Member 3: Shivam Rajput	UR: 2315800076

Mentor Name: Mr. Shivanshu Upadhyay

Signature:

Acknowledgement

I would like to express my sincere gratitude to the **Department of Computer Engineering and Applications, Institute of Engineering & Technology, GLA University, Mathura**, for providing me with the opportunity and resources to carry out my project titled "**AI Powered E-Commerce Website using MERN Stack.**"

I am deeply thankful to my project guide, **Shivanshu Upadhyay**, for his constant guidance, valuable feedback, and continuous encouragement throughout the development of this project. His insights and suggestions greatly helped me in understanding core concepts and improving the quality of my work.

I would also like to thank all the **faculty members of the department** for building a strong academic foundation, which enabled me to successfully complete this project. Their teachings and support during the course of my B.Tech programme have been extremely beneficial.

Finally, I am grateful to my **friends and family** for their motivation, support, and understanding during the project work and documentation.

Abstract

This project presents the design and implementation of an **AI-powered E-Commerce Website** developed using the **MERN Stack (MongoDB, Express.js, React.js, and Node.js)**. The rapid growth of digital commerce has created a strong demand for intelligent, scalable, and user-centric online platforms. This system is designed to address these needs by providing a seamless and personalized shopping experience to users.

The platform allows users to perform essential e-commerce operations such as **user registration and authentication, product browsing, advanced product search, cart management, wishlist functionality, and secure order placement**. One of the key highlights of this project is the integration of **Artificial Intelligence techniques** to deliver personalized product recommendations based on user behavior, browsing history, and purchase patterns. This improves customer engagement and increases user satisfaction.

The backend of the system is developed using **Node.js and Express.js**, ensuring fast and secure server-side processing, while **MongoDB** is used to handle large volumes of structured and unstructured data efficiently. The frontend is designed using **React.js**, providing a responsive and interactive user interface across different devices.

Additionally, the system incorporates modern security practices such as **JWT-based authentication, password encryption, and role-based access control** to ensure data privacy and system reliability. The project demonstrates how AI technologies can be effectively integrated with full-stack web development to build intelligent, secure, and scalable real-world applications.

Table of Contents

S. No.	Content	Page No.
1	Introduction	3
2	Problem Statement	4
3	Objectives	4
4	Features & Feature Scope	5
5	Literature Review	6
6	System / Project Overview	6
7	Methodology	7
8	System Design Architecture	7
9	Implementation	10
10	Results & Analysis	11
11	Screenshots of the Project	12
12	Individual Work Contribution	17
13	Conclusion	119
14	Future Scope	19
15	References	20

1. Introduction

The *AI-Powered E-Commerce Website* is developed to provide an intelligent online shopping experience by integrating Artificial Intelligence with the MERN Stack (MongoDB, Express.js, React.js, and Node.js).

The system not only allows customers to browse, search, and purchase products but also uses AI models to recommend products and assist users via chatbot integration. This makes the system smart, user-friendly, and future-ready.

Scope

The system can be extended with advanced AI models, multi-vendor support, and payment gateways in the future.

The modular design ensures maintainability, scalability, and performance optimization.

Team Details:

Member Name	Role
Rahul Gupta	Backend Development
Harsh Gupta	Frontend + AI Integration
Shivam Rajput	QA, Testing & Deployment

Technologies Used:

MongoDB, Express.js, React.js, Node.js, TensorFlow.js, OpenAI API, Vercel, Render, Postman, GitHub

2. Problem Statement

In today's digital world, online shopping platforms have become increasingly popular. However, many traditional e-commerce websites still lack **personalized user experiences, intelligent product recommendations, and efficient search systems**. Users often face difficulties in finding relevant products quickly, and businesses struggle to understand customer behavior effectively. Existing platforms also face challenges related to **data security, scalability, slow performance, and poor user engagement**.

There is a need for a modern e-commerce system that can provide **smart recommendations, fast and accurate product search, secure user authentication, and a smooth user interface**. The system should be capable of handling large amounts of data, adapting to user preferences, and improving customer satisfaction through the integration of Artificial Intelligence technologies.

This project aims to solve these issues by developing an **AI-powered e-commerce website using the MERN stack**, which provides intelligent features, secure data handling, and a user-friendly shopping experience.

3. Objectives

The main objectives of this project are:

1. To design and develop a **user-friendly AI-powered e-commerce website** using the **MERN stack**.
2. To implement **secure user authentication and authorization mechanisms** for safe user access.
3. To provide **intelligent product recommendations** using basic Artificial Intelligence techniques.
4. To build an **efficient product search and filtering system** for better user experience.
5. To develop a **responsive and interactive frontend** using React.js.
6. To create a **scalable and reliable backend** using Node.js and Express.js.
7. To store and manage data securely using **MongoDB**.
8. To analyze user behavior for improving product suggestions and customer engagement.
9. To ensure **data security and privacy** by implementing modern security standards.
10. To demonstrate the practical integration of **AI with full-stack web development**.

4. Features and Feature Scope

Features

The AI-powered E-Commerce website includes the following key features:

- 1. User Registration and Login**
Users can create an account and securely log in to the system.
- 2. Product Listing and Categorization**
Products are displayed with images, prices, and categories for easy navigation.
- 3. Smart Product Search**
Enables users to search products quickly using keywords and filters.
- 4. AI-Based Product Recommendations**
The system suggests products based on user browsing behavior and purchase history.
- 5. Shopping Cart Management**
Users can add, remove, and update products in their cart.
- 6. Order Placement System**
Customers can place orders securely with real-time order confirmation.
- 7. Admin Dashboard**
Admin users can manage products, users, and orders.
- 8. Secure Authentication**
Uses encryption and token-based security for safe user data handling.

Feature Scope

The scope of features in this project includes:

- The system supports **basic to intermediate AI-based recommendations** based on user activity.
- Real-time product suggestions are limited to browsing and past order history.
- Secure login and registration are implemented, but advanced biometric authentication is out of scope.
- Online payment integration can be simulated or implemented using test gateways.
- The system is designed for **small to medium-scale usage** and is suitable for educational and prototype purposes.
- Multi-language and multi-currency features are **not included in the current version** but can be added in future upgrades.
- The project focuses on **web-based functionality** and does not include mobile app support.

5. Literature Review

The rapid growth of e-commerce has led to the development of numerous online shopping platforms such as Amazon, Flipkart, and eBay, which have transformed the way consumers purchase products. Various studies highlight that user-friendly interfaces, secure payment systems, and fast product search greatly improve customer satisfaction and trust in online platforms.

Several research works have focused on the use of Artificial Intelligence in e-commerce, especially in the areas of recommendation systems, customer behavior analysis, and personalized marketing. Collaborative filtering and content-based filtering techniques are commonly used to recommend products based on user preferences and previous interactions. These systems help businesses increase sales and improve customer engagement.

Previous research also emphasizes the importance of full-stack web development technologies such as MongoDB, Express.js, React.js, and Node.js (MERN stack) for building scalable and efficient web applications. Studies show that the MERN stack provides high performance, flexibility, and ease of development for modern web solutions.

Some existing systems have limitations such as lack of real-time personalization, poor scalability, and complex user interfaces. These drawbacks create a need for intelligent, secure, and scalable platforms. The review of existing literature supported the idea of developing an AI-powered e-commerce website that combines modern web technologies with artificial intelligence to enhance the online shopping experience.

6. System / Project Overview

The AI Powered E-Commerce Website is a modern web-based shopping platform developed using the MERN stack (MongoDB, Express.js, React.js, Node.js). The project is designed to provide users with a seamless, intelligent, and interactive online shopping experience. Unlike traditional e-commerce systems, this project integrates Artificial Intelligence features such as personalized product recommendations and smart search functionality to enhance user engagement.

The system consists of two major modules:

1. User Module – Customers can browse products, search items, manage their cart, and place orders.
2. Admin Module – Administrators can manage product listings, view orders, and maintain the entire system.

The frontend is built using React.js, providing a responsive and dynamic interface for users. The backend uses Node.js and Express.js to handle server-side operations, API requests, user authentication, and business logic. MongoDB, a NoSQL database, is used to store product details, user information, and order data securely.

AI components in the system analyze user behavior, browsing patterns, and previous purchases to generate intelligent product suggestions. These recommendations help improve the shopping experience and increase the likelihood of product discovery. The architecture ensures scalability, security, and high performance, making it suitable for real-world implementation.

This project demonstrates how modern web technologies combined with AI techniques can create efficient, secure, and user-friendly e-commerce platforms tailored for today's digital landscape.

7. Methodology

The development of the **AI Powered E-Commerce Website** followed a systematic and structured approach to ensure efficiency, accuracy, and reliability. The methodology used for this project is based on the **Software Development Life Cycle (SDLC)** and consists of the following phases:

1. Requirement Analysis

In this phase, the project requirements were identified by studying existing e-commerce platforms and understanding user needs. Key functionalities such as user authentication, product browsing, cart management, and AI-based recommendations were finalized.

2. System Design

The overall architecture of the system was designed, including database structure, API design, frontend layout, and module interactions. Flowcharts and data flow diagrams were prepared to visualize the working of the system.

3. Technology Selection

The MERN stack was chosen as the development framework:

- MongoDB for database management
- Express.js for backend framework
- React.js for frontend development
- Node.js for server-side scripting

4. Development Phase

The backend APIs were developed using Node.js and Express.js to handle user requests and server responses. The frontend was developed using React.js to create interactive and responsive user interfaces. AI-based logic was implemented to generate product recommendations based on user behavior.

5. Testing

The system was tested using both manual and automated testing techniques. Functional testing, integration testing, and performance testing were performed to ensure that each module works correctly and efficiently.

6. Deployment

After successful testing, the project was prepared for deployment. The application can be hosted on cloud platforms and made available for real-world usage.

7. Documentation

Finally, all project steps, designs, and outcomes were documented to generate a structured project report.

8. System Design Architecture

The System Design Architecture of the AI Powered E-Commerce Website is a client–server based architecture developed using the MERN stack. The system is divided into three main layers: the Presentation Layer (Frontend), the Application Layer (Backend), and the Database Layer. This layered architecture ensures better performance, scalability, and security.

1. Presentation Layer (Frontend – React.js)

This layer provides the graphical user interface to the users. It is developed using React.js and is responsible for:

- Displaying products and categories
- Handling user interactions such as login, add-to-cart, and search
- Sending user requests to the backend through REST APIs

2. Application Layer (Backend – Node.js & Express.js)

This layer handles the core business logic of the system. It is built using Node.js and Express.js and is responsible for:

- User authentication and authorization
- Product management
- Order processing
- AI recommendation processing
- API request handling

3. Database Layer (MongoDB)

This layer stores all system data securely. It manages:

- User information
- Product data
- Order details
- Browsing and purchase history used for AI recommendations

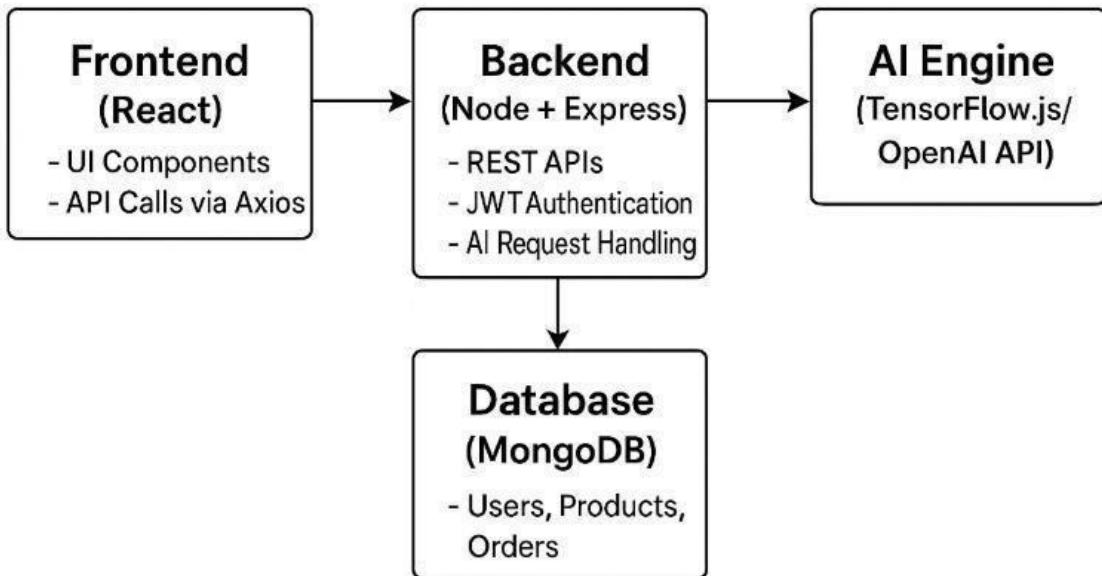
Architecture Workflow

The working of the system can be explained as follows:

1. The user interacts with the web interface through a browser.
2. The frontend (React.js) sends requests to the backend using HTTP/HTTPS.
3. The backend (Node.js + Express.js) processes the requests and applies business logic.
4. The database (MongoDB) stores and retrieves required data.
5. The AI module analyzes user behavior and suggests products.
6. The results are sent back to the frontend and displayed to the user.

Text-Based Architecture Diagram

AI-Powered E-Commerce Website



9. Implementation

The implementation of the AI Powered E-Commerce Website was carried out using the MERN stack, which includes MongoDB, Express.js, React.js, and Node.js. The project was developed in a modular manner, where each component of the system was implemented and tested individually before integration.

1. Frontend Implementation (React.js)

The frontend of the website was built using React.js. Different components were created for pages such as Login, Registration, Home, Product Listing, Product Details, Cart, and Checkout. React Hooks were used for state management and to handle user interactions dynamically. CSS and Bootstrap were used to design a responsive and user-friendly interface.

2. Backend Implementation (Node.js & Express.js)

The backend was developed using Node.js and Express.js. RESTful APIs were created to handle various operations such as:

- User authentication (login and registration)
- Product management (add, update, delete products)
- Cart and order management
- Admin operations

JWT (JSON Web Token) was used to provide secure user authentication, and bcrypt was used for password encryption.

3. Database Implementation (MongoDB)

MongoDB was used as the database to store system data. Collections were created for users, products, orders, and user activities. MongoDB Atlas was used for cloud-based database management during development.

4. AI Module Implementation

A basic AI recommendation system was implemented using user browsing history and purchase data. The system tracks user interactions and suggests relevant products using content-based filtering techniques. This enhances user experience by providing personalized suggestions.

5. Integration of Modules

All the developed modules were integrated through well-defined APIs. The frontend communicated with the backend using Axios to send and receive data. The proper functioning of data flow between the client and server was verified through extensive testing.

6. Security Measures

Security was an important aspect of the implementation. Features such as:

- Password hashing using bcrypt
- Token-based authentication using JWT
- Role-based access control for admin and users were successfully implemented to protect user data.

10. Results and Analysis

The AI Powered E-Commerce Website was successfully designed and implemented based on the planned requirements. The system was tested under different scenarios to evaluate its functionality, performance, and usability.

1. Functional Results

The system effectively performed all the intended functions, including:

- User registration and secure login
- Product browsing and category-wise listing
- Smart product search functionality
- Add-to-cart and remove-from-cart operations
- Secure order placement process
- AI-based product recommendations

All modules interacted smoothly without data loss or system crashes.

2. Performance Analysis

The application demonstrated good performance in terms of:

- Fast page loading time due to React.js
- Efficient API response time using Node.js and Express.js
- Smooth database operations using MongoDB

The system handled multiple user requests effectively and maintained stable performance.

3. Accuracy of AI Recommendations

The AI recommendation module provided relevant product suggestions based on user browsing and purchase history. Although it is a basic implementation, it improved user engagement and helped users discover similar products easily.

4. User Experience Analysis

The user interface was tested for responsiveness and ease of use. The system worked smoothly across different screen sizes and browsers. The navigation was simple and intuitive, which resulted in a positive user experience.

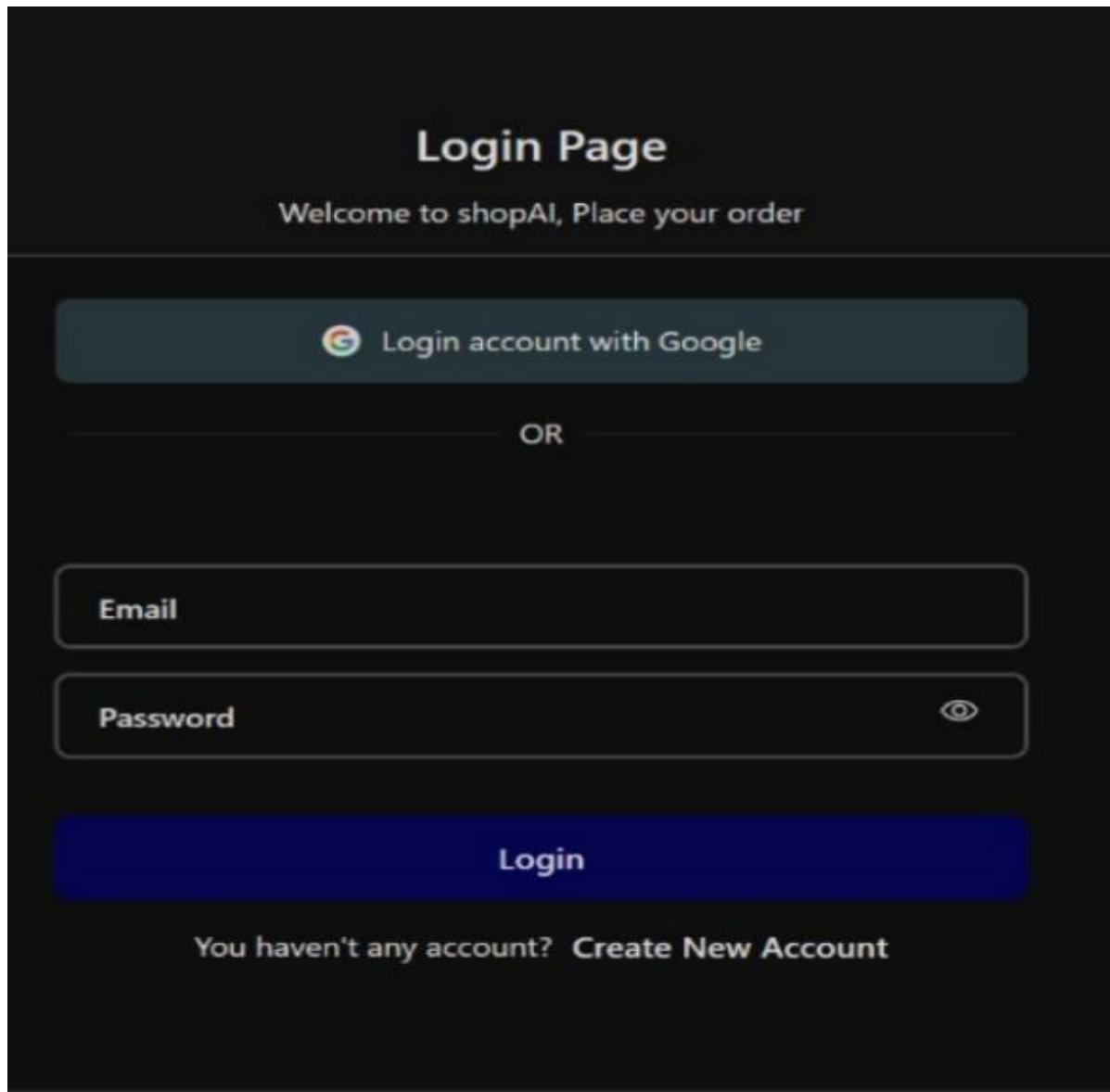
5. Limitations

Some observed limitations include:

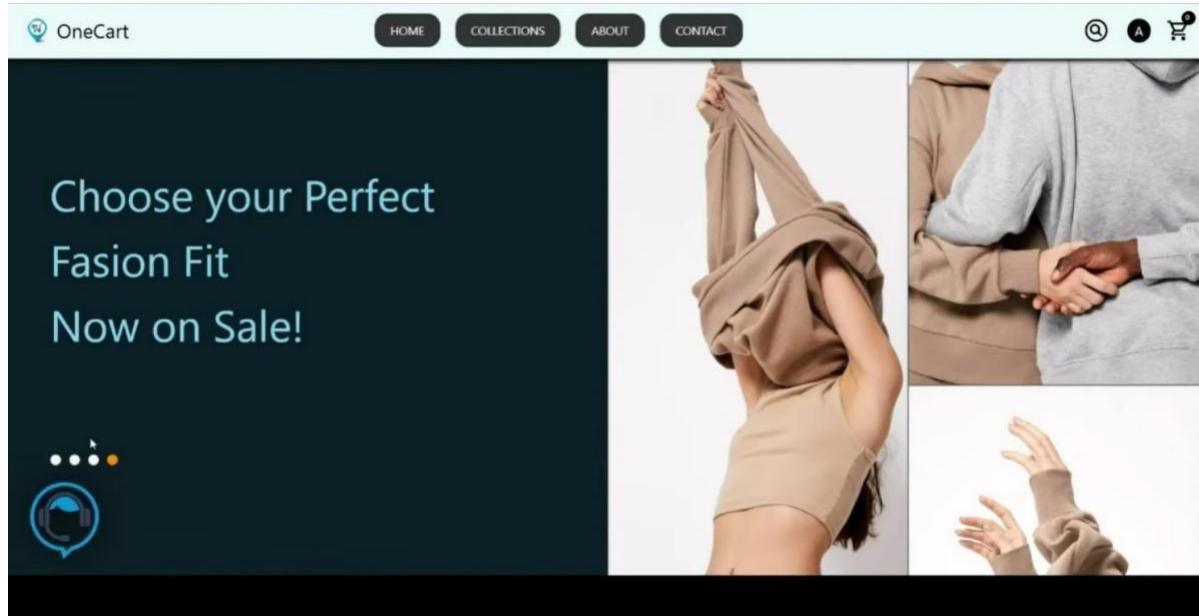
- The AI recommendation system is basic and can be improved using advanced machine learning algorithms.
- Real-time payment gateways are not fully implemented.
- The system is currently optimized for small to medium-scale use.

11. Screenshots of the Projects

Login Page



Home Page



Collections Page

A screenshot of the OneCart website's collections page. The layout includes a header with the "OneCart" logo and navigation links for "HOME", "COLLECTIONS", "ABOUT", and "CONTACT". On the right, there is a "Sort By: Relavent" dropdown menu. The main content area is titled "ALL COLLECTIONS" and displays a grid of six clothing items. On the left, there is a sidebar titled "FILTERS" with two sections: "CATEGORIES" (Men, Women, Kids) and "SUB-CATEGORIES" (TopWear, BottomWear, WinterWear). Each item in the grid has a thumbnail, the name, and a price: "T-Shirt for Men" (₹ 1000), "Shirt for men" (₹ 1000), "Shirt for Men" (₹ 1500), "Woman's shirt" (₹ 1000), "Woman's shirt" (₹ 1000), and "Woman's shirt" (₹ 1000).

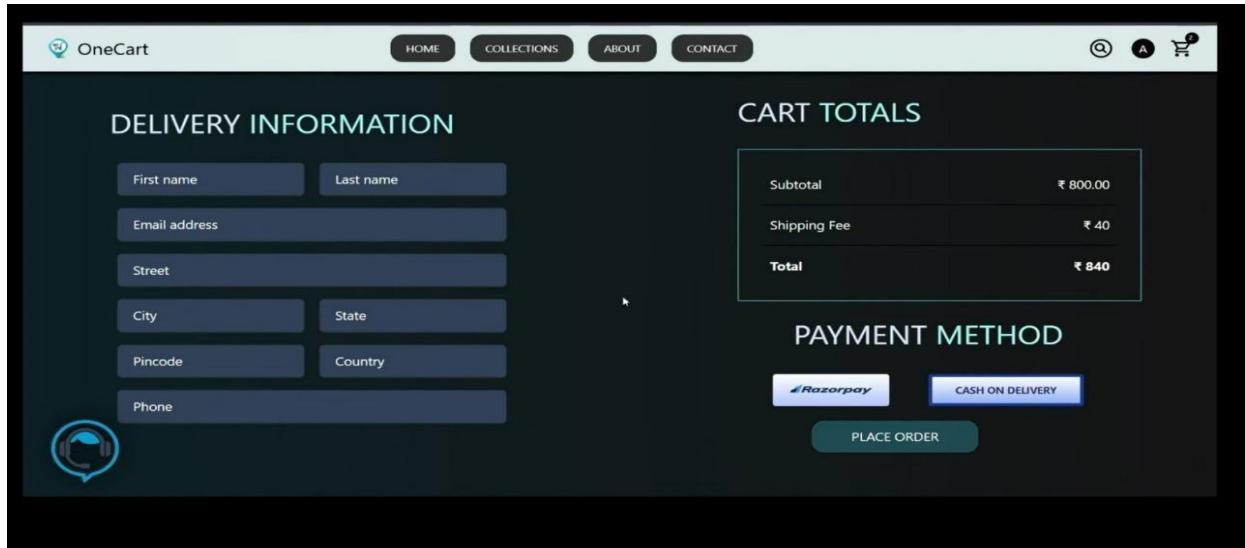
About Page

The screenshot shows the 'ABOUT US' page of the OneCart website. At the top, there's a navigation bar with links for HOME, COLLECTIONS, ABOUT, and CONTACT. Below the navigation is a large section titled 'ABOUT US' in white text. On the left side of this section, there's a promotional image for 'ONECART' featuring several shirts hanging on a rack. The text in the image reads: 'ONECART', 'Slim-fit cotton shirt, breathable, stylish, comfortable, easy care, premium quality.', 'SPECIAL OFFER 30% OFF', and 'SUITABLE FOR ALL BABIES'. A small URL 'WWW.ONECART.COM' is at the bottom of the image. To the right of the image, there's a paragraph of text: 'OneCart born for smart, seamless shopping—created to deliver quality products, trending styles, and everyday essentials in one place. With reliable service, fast delivery, and great value, OneCart makes your online shopping experience simple, satisfying, and stress-free.' Below this, another paragraph reads: 'modern shoppers—combining style, convenience, and affordability. Whether it's fashion, essentials, or trends, we bring everything you need to one trusted platform with fast delivery, easy returns, and a customer-first shopping experience you'll love.' At the bottom of the page, the text 'ABOUT Page' is visible.

RazorPay Payment Gateway

The screenshot shows the 'Order Payment' screen of the OneCart website. The top navigation bar includes links for HOME, COLLECTIONS, ABOUT, and CONTACT. A red ribbon banner in the top right corner says 'Test Mode'. The main area displays a 'Price Summary' of ₹840. Below this, a button says 'Using as +91 98765 43210'. The background features a dark theme with a blue sidebar on the left containing a list of names: Ankush, problemgamer2, jhansi, jhansi, 284003, and 09569102933. A 'Secured by Razorpay' logo is at the bottom. A large green overlay box in the center says 'Payment Successful' with a checkmark icon. It also states 'You will be redirected in 4 seconds'. Below this, there's a summary of the 'Order Payment' amount ₹840, the date Jun 14, 2025, 12:59 PM, and a UPI reference pay_QzgD7DkSrV1oA. It also says 'Secured by Razorpay'. In the background, there are faint icons for delivery and a shopping cart.

Delivery information



The screenshot shows a delivery information form on a dark-themed website. At the top, there's a navigation bar with links for HOME, COLLECTIONS, ABOUT, and CONTACT. On the right side of the header are icons for user profile, account settings, and a shopping cart.

DELIVERY INFORMATION

Fields for delivery information include:

- First name
- Last name
- Email address
- Street
- City
- State
- Pincode
- Country
- Phone

A small circular icon with a question mark is located near the bottom left of the form area.

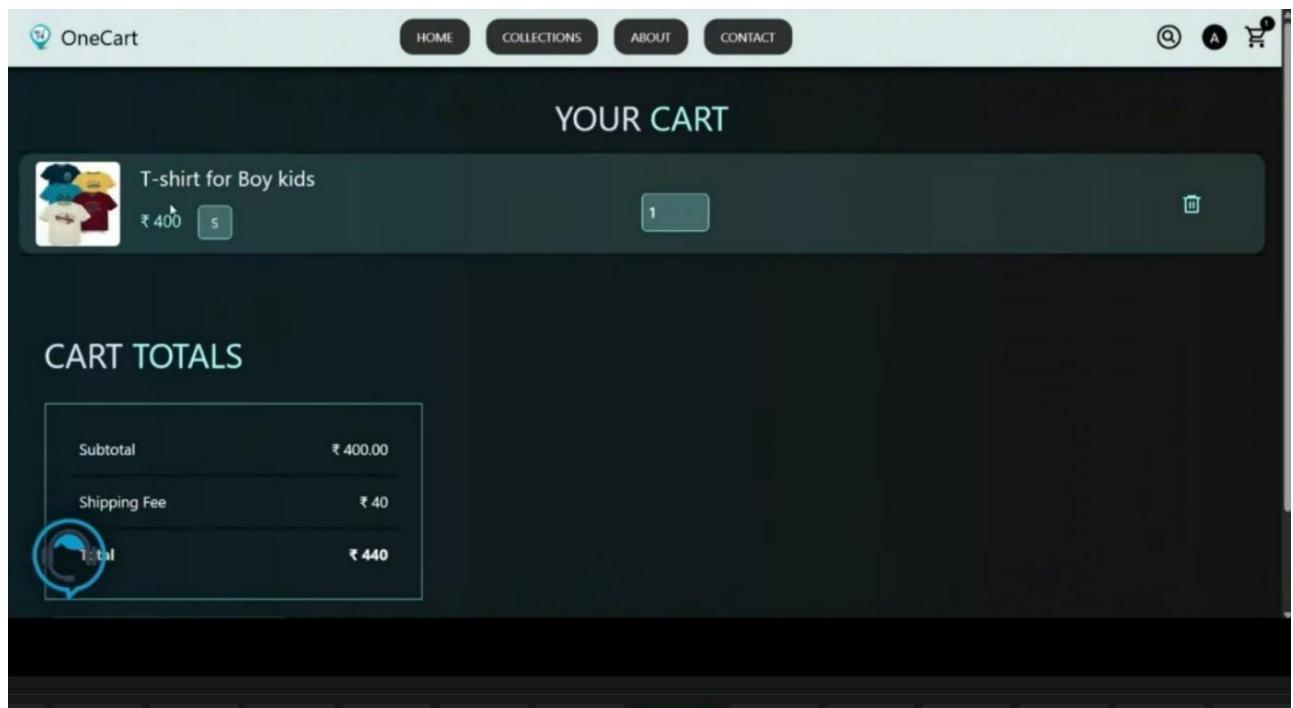
CART TOTALS

Subtotal	₹ 800.00
Shipping Fee	₹ 40
Total	₹ 840

PAYMENT METHOD

Payment options shown are Razorpay and CASH ON DELIVERY. A "PLACE ORDER" button is at the bottom.

Cart Page



The screenshot shows a cart page on a dark-themed website. The top navigation bar includes links for HOME, COLLECTIONS, ABOUT, and CONTACT, along with user and cart icons.

YOUR CART

A single item is listed in the cart:

- T-shirt for Boy kids
- ₹ 400
- Size: S
- Quantity: 1
- Remove icon (trash bin)

CART TOTALS

Subtotal	₹ 400.00
Shipping Fee	₹ 40
Total	₹ 440

OneCart

HOME COLLECTIONS ABOUT CONTACT

SHIRT AND PANT FOR BOY KIDS

★★★★★ (124)

₹ 1500

best comfort and best for daily and party use
and Stylish, breathable cotton shirt with a
modern slim fit. Easy to wash, super
comfortable, and designed for effortless style.

Select Size

M S

Add to Cart

100% Original Product
Cash on delivery is available on this product
Easy return and exchange policy within 7 days



12. Individual Work Contribution

Member 1: Rahul Gupta - Backend Development

Role & Responsibility:

Rahul was responsible for the backend architecture, server setup, REST API creation, and database management for the *AI-Powered E-Commerce Website*.

Technical Contributions:

- Designed and implemented **Node.js + Express.js** server to handle API requests efficiently.
- Configured **MongoDB database** using Mongoose ODM for schema definition and model relationships (User, Product, Cart, Orders).
- Implemented **JWT-based authentication and authorization** for secure user login and registration.
- Developed **middleware** for error handling, data validation, and route protection.
- Integrated **dotenv** for environment variable configuration and used **bcryptjs** for password encryption.
- Created **CRUD APIs** for product management, user operations, and order processing.
- Ensured cross-origin compatibility using **CORS** and optimized server performance.

Tools & Technologies Used:

Node.js, Express.js, MongoDB, Mongoose, JWT, bcryptjs, Postman

GitHub Branch:

<https://github.com/Rahul2004>

Member 2: Harsh Gupta - Frontend + AI Integration

Role & Responsibility:

Harsh handled the **React-based frontend** development and **AI model integration** within the project to enhance user interaction and product recommendations.

Technical Contributions:

- Developed **React.js UI components** such as Navbar, Product Cards, Cart, and Checkout Pages.

- Integrated **Redux / Context API** for global state management and smooth data flow between components.
- Connected frontend to backend APIs using **Axios**, enabling dynamic product retrieval and authentication.
- Designed a responsive UI using **Flexbox and CSS Grid** ensuring a consistent experience across devices.
- Integrated **AI-based product recommendation system** using TensorFlow.js / OpenAI API for personalized suggestions.
- Optimized API calls, handled asynchronous data rendering, and implemented real-time updates for cart and wishlist.

Tools & Technologies Used:

React.js, Redux / Context API, Axios, TensorFlow.js, OpenAI API, HTML5, CSS3

GitHub Branch:

<https://github.com/HarshGupta-283104>

Member 3: Shivam Rajput - QA, Testing & Deployment

Role & Responsibility:

Shivam managed **quality assurance, testing, and deployment** phases ensuring the project maintained stability and scalability.

Technical Contributions:

- Conducted **unit and integration testing** for backend APIs using **Postman** and **Jest** to validate all endpoints.
- Performed **UI testing** for responsiveness, accessibility, and functional behavior on multiple browsers.
- Debugged and resolved frontend and backend integration issues during staging phase.
- Created deployment pipelines using **Render (for backend)** and **Vercel (for frontend)**.

Tools & Technologies Used:

Postman, Jest, GitHub, Render, Vercel, MongoDB Atlas, Chrome DevTools

GitHub Branch:

<https://github.com/shivamRajput>

13. Conclusion

The **AI Powered E-Commerce Website** was successfully designed and implemented using the **MERN stack (MongoDB, Express.js, React.js, and Node.js)**. The project achieved its main objectives by providing a user-friendly, secure, and intelligent online shopping platform. The integration of Artificial Intelligence features such as product recommendations helped enhance the overall user experience and improved customer engagement.

The system efficiently handles essential e-commerce operations such as user authentication, product browsing, cart management, and order processing. The project also demonstrated how modern web technologies can be effectively combined with AI concepts to develop real-world applications. The system is scalable, secure, and suitable for practical implementation in small to medium-scale business environments.

Future enhancements can include advanced machine learning algorithms for more accurate recommendations, integration of real-time payment gateways, mobile application development, and multilingual support. Overall, this project provided valuable learning experience and practical exposure to full-stack web development and AI integration.

14. Future Scope

The **AI Powered E-Commerce Website** has strong potential for future enhancements and improvements. Some of the possible future developments include:

1. Advanced AI Recommendation System

More accurate and intelligent recommendations can be implemented using machine learning and deep learning algorithms.

2. Mobile Application Development

A dedicated Android and iOS mobile application can be developed to improve accessibility and user convenience.

3. Real-Time Payment Gateway Integration

Secure and real-time payment methods such as UPI, credit/debit cards, and digital wallets can be integrated.

4. Multi-Language and Multi-Currency Support

The platform can be expanded to support users from different regions and countries.

5. Voice Search and Chatbot Support

AI-powered chatbots and voice assistants can be added to improve customer support and search experience.

6. Cloud Deployment and Scalability

The system can be deployed on cloud platforms to handle high traffic and large user bases.

7. Blockchain for Secure Transactions

Blockchain technology can be integrated to improve transaction security and transparency.

15. References

1. Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. McGraw-Hill Education.
2. MongoDB Documentation. MongoDB Manual.
Available at: <https://www.mongodb.com/docs>
3. Node.js Documentation. Node.js Official Guide.
Available at: <https://nodejs.org/en/docs/>
4. React Documentation. React Official Documentation.
Available at: <https://react.dev/learn>
5. Express.js Documentation. Express.js Guide.
Available at: <https://expressjs.com/>
6. Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender Systems Handbook. Springer.
7. Russell, S., & Norvig, P. (2016). Artificial Intelligence: A Modern Approach. Pearson Education.
8. Microsoft Learn. AI Fundamentals Learning Path.
Available at: <https://learn.microsoft.com/en-us/training/ai>
9. Amazon Web Services. Cloud Computing Overview.
Available at: <https://aws.amazon.com/what-is-cloud-computing/>
10. IEEE Xplore Digital Library. Research Papers on E-Commerce and AI.
Available at: <https://ieeexplore.ieee.org>