CHAPTER 1

COMPANY PROFILE

1.1 Rooman Technologies

Rooman Technologies is an Indian IT training and services company headquartered in Bengaluru, Karnataka. Established in the late 1990s, Rooman has grown to become a prominent player in the field of IT education and training in India. The company specializes in providing a wide range of courses, including hardware and networking, software development, cybersecurity, cloud computing and Global Data. Rooman Technologies has partnered with various governmental and non-governmental organizations to deliver skill development programs aimed at enhancing employability among youth. Their collaborations include initiatives with the National Skill Development Corporation (NSDC) and other state-level skill development missions. Through these partnerships, Rooman has trained thousands of students, contributing significantly to the Skill India mission. Rooman Technologies has partnered with various governmental and non-governmental organizations to deliver skill development programs aimed at enhancing employability among youth. Their collaborations include initiatives with the National Skill Development Corporation (NSDC) and other state-level skill development missions. Through these partnerships, Rooman has trained thousands of students, contributing significantly to the Skill India mission.

In addition to training, Rooman offers IT services such as system integration, network solutions, and managed services to a diverse clientele, including educational institutions, corporate organizations, and government agencies. Their commitment to quality education and services has earned them recognition and awards in the industry. In addition to training, Rooman offers IT services such as system integration, network solutions, and managed services to a diverse clientele, including educational institutions, corporate organizations, and government agencies. Their commitment to quality education and services has earned them recognition and awards in the industry.

1.2 IBM (International Business Machines Corporation)

Founded in 1911 as the Computing-Tabulating-Recording Company (CTR) and renamed IBM in 1924, International Business Machines Corporation is a global technology leader headquartered in Armonk, New York, USA. Operating in over 170 countries with a workforce of more than 300,000, IBM has played a pivotal role in the evolution of software and application development.

IBM supports modern web and mobile application development through its comprehensive suite of tools and cloud-based services. The IBM Cloud platform enables developers to build, test, and deploy scalable applications using technologies like Kubernetes, containers, and serverless computing. Its focus on hybrid cloud architecture allows seamless integration between on-premises systems and public cloud services, making it ideal for enterprise-grade web and mobile solutions.

The $34 billion acquisition of Red Hat in 2019 significantly enhanced IBM's open-source and containerization capabilities. Through Red Hat OpenShift, developers can create cloud-native applications using microservices and DevOps practices. IBM's tools like Mobile Foundation, API Connect, and Cloud Code Engine empower developers to build cross-platform apps with robust backend integration and secure APIs.

IBM supports a wide range of languages and frameworks such as Java, Node.js, Swift, and Kotlin, enabling responsive and high-performance application development. The company also invests in developer education through IBM Developer and SkillsBuild platforms, offering free resources, code patterns, and hands-on labs.

Under CEO Arvind Krishna's leadership since 2020, IBM has focused on advancing digital transformation by supporting agile, full-stack development across web and mobile environments. The company continues to innovate with tools that simplify the development lifecycle and enhance user experience, reinforcing its commitment to empowering developers and driving enterprise modernization.

1.3 Wadhwani Foundation

The Wadhwani Foundation is a global not-for-profit organization established in 2000 by Dr. Romesh Wadhwani, a Silicon Valley entrepreneur and philanthropist. Headquartered in the United States, the Foundation operates in several countries across Asia, Africa, and Latin America, with a mission to accelerate economic development by creating high- quality jobs and promoting entrepreneurship.The Foundation's initiatives support digital innovation and modern workforce readiness, offering rich opportunities for professionals in web and mobile application development.

Entrepreneurship:

Through its National Entrepreneurship Network (NEN), the Foundation empowers aspiring entrepreneurs and startups with training, mentorship, and resources. These programs often include support for digital product development, encouraging the use of web and mobile technologies to create scalable business solutions. Developers are engaged in building MVPs, launching tech-driven startups, and creating mobile-first platforms for real-world impact.

Skilling:

The Wadhwani Opportunity initiative equips job seekers with industry-relevant digital and technical skills, including web and mobile development. The program emphasizes employability training in areas like front-end development, backend technologies, app development frameworks, and responsive design. Learners gain practical skills through project-based learning and developer-centric content tailored to the modern job market.

Innovation & Research:

Focused on fostering technological innovation, the Foundation supports academic and institutional research that integrates digital tools, mobile solutions, and software platforms. This opens avenues for developers to collaborate on impactful web-based applications and mobile interfaces that support health tech, education, and community development.

CHAPTER 2

SERVICES AND ACTIVITIES AT THE COMPANY

2.1 Departments and Services Offered at IBM, Roman and Wadhwani Technologies

IBM, Roman Technologies, and Wadhwani Foundation are globally recognized organizations known for their multifaceted contributions to technology, innovation, consulting, and digital transformation. These companies operate across various departments, offering scalable and high-impact solutions to enterprises, startups, and institutions. Their services range from software engineering and AI-driven analytics to training, workforce development, and cloud platform engineering—delivering end-to-end solutions tailored for enterprise needs.

The Software Development Divisions of IBM and Roman Technologies focus on building secure, scalable, and modular platforms. These include full-stack web and mobile application development, cloud-native SaaS applications, and enterprise-level systems. These companies integrate automation, microservices, and secure API connectivity in their development life cycles to support rapid deployment and reduced operational costs. Subscription-based, ready-to-use application services help businesses launch faster with reduced time-to-market.

In parallel, Wadhwani Foundation and Roman Technologies also invest in Outsourced IT Consulting and Project Execution Models, enabling organizations to delegate software development tasks to expert teams and trained student developers. This structure provides mutual benefits—enterprises minimize hiring and training expenses while students and interns gain exposure to real-time industrial projects, mentorship, and skill-building experiences.

The AI & Research Divisions at IBM and Wadhwani Foundation work on building and deploying AI solutions across sectors like healthcare, finance, and education. They develop applications such as predictive analytics tools, intelligent automation systems, and image recognition models.

Additionally, they are actively involved in academic and industrial research, collaborating with universities, government bodies, and international investors to explore

breakthroughs in artificial intelligence, quantum computing, and machine learning.

Together, these departments reflect the companies' commitments to technological innovation, business empowerment, and skill-based ecosystem development, aligning with modern trends in digital transformation.

Roman Technologies contributes to the tech ecosystem by supporting local startups through product incubation and MVP (Minimum Viable Product) development services. Their cross-functional teams work with clients from ideation to deployment, ensuring product-market fit, UI/UX standards, and backend scalability. Their flexible project engagement models include fixed-cost, milestone-based, and resource augmentation contracts, providing options for businesses of all sizes.

Across all three organizations, there is a common goal: to foster **innovation-driven ecosystems** that empower businesses, uplift communities, and prepare the next generation of technologists. By combining technical excellence with strategic consulting and grassroots-level impact, these organizations continue to play a pivotal role in shaping the future of the digital world.

CHAPTER 3

INTRODUCTION

3.1 Introduction to Application Developer – Web and Mobile

An Application Developer – Web and Mobile is responsible for building responsive, secure, and scalable applications that operate efficiently across web browsers and mobile devices. This role encompasses both frontend (client-side) and backend (server-side) development, involving user interface creation, real-time data handling, authentication, database integration, and deployment processes. The developer ensures seamless performance and interaction across devices while adhering to modern development practices and architecture.

Key Components of Web and Mobile Application Development

Key components of web and mobile application development include frontend and backend development, database integration, and responsive UI/UX design. These components work together to ensure seamless functionality, user interaction, and real-time data handling across platforms.

1. Frontend Development (Client-Side)

- **Overview:**

  The frontend is the visible part of the application where users interact with features such as product browsing, login/register forms, and cart management.

- **Technologies Used**
  - **HTML, CSS:** Structured layout and styling of the platform.
  - **JavaScript (ES6+):** Handles dynamic behavior and interactivity.
  - **Frontend Frameworks/Libraries:**
    - **React.js:** Used to build reusable UI components and manage state.
    - **Bootstrap, Google Fonts:** For responsive and aesthetic design.
  - **State Management:**
    - Context API: Handles global state like login visibility.

- **Outcome:**

  A dynamic, mobile-responsive, and user-friendly interface for e-commerce interaction.

2. Backend Development (Server-Side)

- **Overview:**

  Backend manages data processing, authentication, and real-time synchronization between users, vendors, and admin systems.

- **Technologies Used:**
  - **Node.js with Express.js:** API development and server logic.
  - **Authentication & Security:**
    - **JWT:** Secure token-based session management.
    - **bcrypt.js:** Hashes passwords before database storage.
    - **Middleware:** Protects routes and handles user roles (e.g., admin).
- **Outcome:**

  A secure backend system for handling user sessions, order management, and admin operations.

3. Database Management

- **Overview:**

  Data such as user details, product info, and cart items are stored and retrieved from a structured database.

- **Technologies Used:**
  - **Database:** MongoDB for flexible NoSQL data storage.
  - **ORM:** Mongoose used for defining schemas and database interaction.
- **Entities:**
  - **Users, Products, Orders, Categories:** Represent key objects in the system.
- **Outcome:**

  Efficient data handling with real-time update capability and structured access control.

4. DevOps & Deployment

- **Overview:**

  Ensures application is hosted, versioned, and accessible for users with minimal downtime.

- **Technologies Used:**
  - **Version Control:** Git, GitHub for code management.

- **Hosting & Deployment:**
  - **Cyclic:** Used for deployment and feedback collection.
  - **GitHub Repository:** Project source code and updates.

- **Outcome:**

  A live, deployed e-commerce platform accessible for testing and usage.

5. Architecture & Design Patterns

- **Overview:**
  - **Architecture Style:**
    - **Modular Full Stack Architecture** with separation of frontend, backend, and database.
  - **Design Pattern:**
    - **MVC (Model-View-Controller):** Applied for clear separation of data models, user interface, and control logic.
- **Outcome:**

  A scalable architecture supporting real-time e-commerce operations.

6. Testing & Debugging

- **Overview:**

  Application was tested for API integrity, security, and user interaction.
- **Testing Strategies:**
  - **API Testing:**
    - **Postman:** Used to test authentication, product updates, and payment workflows.
- **Outcome:**

  A functional and tested platform ready for real-world usage with tested login, product, and payment operations.

CHAPTER 4

SYSTEM ANALYSIS

4.1 Existing System

In the traditional e-commerce environment, many platforms suffer from **inefficient data synchronization** and lack **real-time updates**. These systems often:

- Rely on **manual refresh** or polling to update inventory and cart data.
- Lack integration of **real-time order tracking** and **live communication** between users and vendors.
- Use outdated UI/UX design principles, which results in poor user experience, especially on mobile devices.
- Offer **insecure authentication systems**, which expose user data to potential breaches.
- Often lack **cross-platform compatibility**, making the system rigid for use across devices.

Furthermore, these systems generally do not provide a seamless experience across roles (admin, and customer), resulting in data inconsistency, delayed order updates, and poor user satisfaction.

4.2 Proposed System

To address the limitations of the existing systems, our project proposes a real-time, full-stack multi- vendor e-commerce platform that is secure, scalable, and responsive across all devices.

Key Features of the Proposed System:

- **Real-Time Synchronization:** Using WebSocket (Socket.io) for real-time updates of cart, orders, and inventory.
- **Modern Frontend:** Built using React.js, ensuring fast rendering and responsive design using Bootstrap and Google Fonts.
- **Backend Security:** Implemented using Node.js with Express.js, integrated with JWT for token-based authentication and bcrypt.js for password hashing.

- **Database:** Uses MongoDB, offering flexible schema and scalability for large datasets like users, orders, and products.

- **Role-Based Access:** Admin panel, vendor dashboard, and customer interface each have specific functionalities.
- **Secure Payments:** Integrated with Stripe and PayPal for secure transactions.
- **Deployment:** Hosted on GitHub and deployed via Cyclic for live testing and feedback collection.

By implementing this architecture, the proposed system ensures a reliable, user-friendly**,** and real-time shopping experience that supports customers, vendors, and administrators in a unified interface.

4.3 Objective of the System

The main objective of this project is to develop a **synchronized, real-time e-commerce platform** using **React.js and Node.js** that supports a **multi-vendor marketplace model**. **Specific Goals:**

- To design a **responsive and scalable UI/UX** for online shoppers.
- To implement **secure user authentication** using **JWT** and **bcrypt.js**.
- To allow **real-time order tracking** and inventory updates across user and vendor dashboards.
- To facilitate **seamless payments** using **Stripe and PayPal** integrations.
- To provide **admin-level control** over system operations, including product management and user monitoring.
- To **simulate a professional e-commerce business** with real-world functionalities like product listings, filters, secure login, payment processing, and responsive design.

By achieving these goals, the system will deliver a smooth and secure online shopping experience while offering the flexibility and control expected in a modern e-commerce solution.

CHAPTER 5

REQUIREMENT ANALYSIS

5.1 Hardware Requirement Specification

The hardware requirements for the proposed E-commerce platform are determined considering performance, scalability, and efficiency during both development and deployment phases. As this is a full-stack web application built using **React.js (frontend)** and **Node.js with MongoDB (backend)**, the following hardware configurations are recommended:

1. Server-Side (Hosting Environment)

- **Processor:** Dual-core Intel Core i3 or equivalent (2.0 GHz or higher)
- **RAM:** Minimum 4 GB
- **Hard Disk:** 100 GB SSD or higher
- **Network:** 1 Gbps Ethernet Adapter
- **Operating System:** Ubuntu 20.04 LTS / Windows Server 2019
- **Deployment Platform:** GitHub (source control), Cyclic (hosting & feedback)

2. Client-Side (User Systems)

- **Processor:** Intel Pentium or equivalent
- **RAM:** 2 GB minimum
- **Hard Disk:** 50 GB free space
- **Operating System:** Windows 10 / Ubuntu
- **Browser:** Latest versions of Google Chrome, Mozilla Firefox, Edge
- **Connectivity:** Stable internet connection for real-time updates and sync
- **Backup & Restore:** Regular Git-based code backup and versioning

5.2 Software Requirement Specification

A] Functional Requirements

- **User Authentication:**

  Secure login and registration with **JWT token-based authentication** and **bcrypt.js** password encryption.
- **Cart & Order Management:**

  Users can add to cart, place orders, and track order status in real time.

- **Payment System:**

  Integrated **Stripe** and **PayPal** for secure online transactions with support for credit card payments.

- **Database:**

  MongoDB database with schemas for Users, Products, Orders, and Categories.

- **UI/UX Features:**

  Dynamic product recommendations, flash sales (gamification), and responsive design using **React.js**, **Bootstrap**, and **Google Fonts**.

- **Reports & Monitoring:**

  Admin can view order statuses (success/processing), and vendors can see live dashboard updates.

- **Testing & Feedback:**

  Tested using **Postman**;

B] Non-Functional Requirements

- **Availability:**

  Uptime of **99.5%** targeted; application tested live on Cyclic with user interaction and load scenarios.

- **Accessibility:**

  Responsive and mobile-friendly design using Bootstrap; supports access from various devices and screen sizes.

- **Security:**
  - Passwords hashed using **bcrypt.js**
  - Sessions protected using **JWT tokens**
  - Admin/user access roles enforced via **middleware**
  - Secure payment flow with **2FA concept** and **HTTPS**
- **Performance:**
  - Real-time updates via **WebSocket**
  - Response time optimized (<2 seconds)
  - Efficient API design and minimal latency

Usability:

Clean UI using React components and Context API for smooth transitions; minimal training required for users.

- **Scalability:**

  Built with **MongoDB** and modular backend (**Node.js + Express.js**), easily scalable for more users.

- **Portability:**

  Cross-platform compatibility with major browsers and operating systems (Windows, Linux, Android).

- **Maintainability:**

  Modular architecture (separate frontend/backend)

  GitHub version control

  Logging and structured API routes for easier debugging

CHAPTER 6

DESIGN ANALYSIS

The below figure represents the **System Architecture** of the multi-vendor e-commerce platform. It is divided into four main components: **Front-End (React.js)**, **Back-End (Node.js/Express.js)**, **Database (MongoDB)**, and **WebSocket (Socket.io)** for real-time communication. The front-end manages the user interface, state, and JWT-based authentication, interacting with the backend via REST APIs and WebSocket events. The backend handles business logic, authentication, API processing, and communicates with both the database and payment gateways (Stripe/PayPal). MongoDB stores user profiles, products, orders, and payment details. Socket.io ensures real-time updates for cart synchronization, inventory, and order status across all user roles.

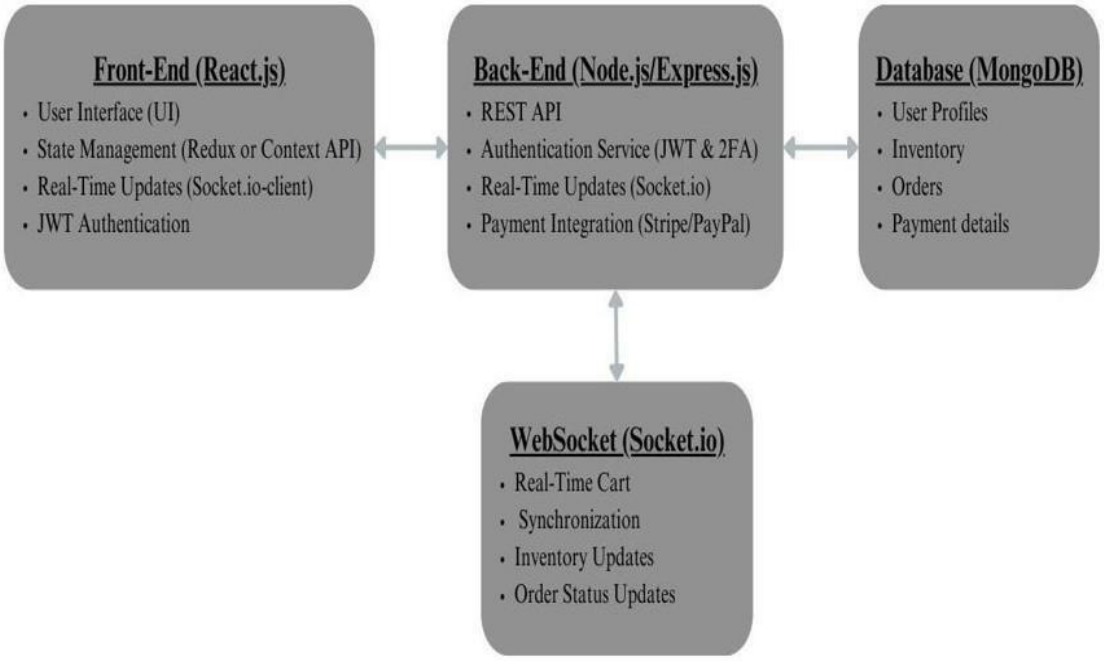

Fig 6.1 : How system works

6.1 System Architecture :

The system architecture of our e-commerce platform follows a layered full- stack design using React.js for the frontend, Node.js/Express.js for the backend, MongoDB for the database for real-time communication. The frontend handles UI, state while the backend

manages RESTful APIs, authentication, and payment processing. MongoDB stores structured data like user profiles, products, orders, and categories. Socket.io is used.

A visual representation of the architecture includes components such as:

- React.js frontend (UI, state management, JWT auth)
- Node.js/Express backend (API, auth, Stripe/PayPal integration)
- MongoDB database (user, product, order collections)
- Socket.io layer for real-time cart, inventory, and order status sync

6.2 Technology Stack :

Frontend:	React.js, Bootstrap, Google Fonts, Context API, Axios.
Backend:	Node.js, Express.js, JWT, bcrypt.js, Socket.io
Database	MongoDB, Mongoose ORM
Payment	Stripe, PayPal
Other Tools	Postman (API Testing), GitHub (Version Control),

6.3 Key Modules and Design Considerations :

6.3.1 User Management

- Secure **JWT-based authentication**
- **bcrypt.js** used to hash passwords
- Middleware for protected routes and admin roles

6.3.2 Inventory Module

- Vendors can **add/update/delete** product stock
- Real-time inventory sync using **Socket.io**
- Product categories and filters for easy search

6.3.3 Order & Payment Module

- Add to cart functionality with session-based tracking
- Payment processing via **Stripe** and **PayPal**
- Order status updates (Processing, Success)
- Admin view of all orders and sales data

6.3.4 Admin Dashboard

- Role-based admin access
- Manage users, view orders, and add new products
- Dashboard with **order tracking**, **delivery status**, and **product control**

6.4 Database Schema Overview (Core Tables)

- **User:** name, email, hashed_password, role (customer/vendor/admin)
- **Product:** name, description, price, quantity, vendor_id, category
- **Order:** items[], user_id, payment_status, delivery_status, timestamps
- **Category:** name, description

All collections are designed using Mongoose schemas with proper validation and indexing for optimized queries.

6.5 Security & Performance Considerations

- JWT for secure token-based authentication
- Bcrypt.js for password hashing
- Middleware for protected routes (admin/user roles)
- Stripe/PayPal use tokenized and encrypted APIs

Performance:

- WebSocket (Socket.io) reduces delay via live event broadcasting
- MongoDB indexing for fast retrieval

- Load distribution across components through modular routing and async processing

6.6 Scalability and Maintainability

Scalability:

- MongoDB supports sharding for distributed data handling
- Socket.io handles multiple real-time user connections simultaneously

Maintainability:

- React component-based design ensures reusable UI parts
- Express middleware and route separation allow clean API structure
- Easily extendable database with flexible NoSQL schema (MongoDB)

CHAPTER 7

IMPLEMENTATION

7.1 Frontend Implementation (React.js):

The frontend of the e-commerce platform was built using React.js, focusing on responsiveness and a clean user experience. Components like Header, Footer, Login/Register, Home Page, Search Bar, and Product Listings were created using reusable React components.

- UI Design was implemented using Bootstrap and Google Fonts for responsiveness and style consistency.
- State Management was handled using the Context API, which allowed for global state sharing like hiding login/registration once the user logs in.
- Axios was used to make HTTP requests to the backend for actions such as user login, product fetching, and cart operations.
- Socket.io-client was integrated to manage real-time updates for cart synchronization, inventory status, and order confirmation.

8.2 Backend Implementation (Node.js/Express.js)

The backend was developed using Node.js and Express.js, structured around RESTful APIs and secure user management.

- Authentication was handled using JWT (JSON Web Tokens) for secure sessions and bcrypt.js to hash passwords.
- Middleware was implemented for:

  Token verification for protecting user endpoints.

  Admin access control to protect sensitive admin-only routes.
- API Endpoints were created for:

  /api/auth/register – Registering a new user

  /api/orders – Placing and managing orders CRUD operations for products and categories
- Forgot Password: Instead of OTP, the app uses a security question-based system to reset the password.

- Role-Based Access: Users can either log in as customers or admins. Admins can manage orders and product listings.

8.3 Payment Integration

The system supports both Stripe and PayPal for secure payment handling.

- Initially, PayPal integration encountered sandbox-related errors (e.g., Braintree configuration).
- The team resolved this by switching to Stripe, which allowed smoother handling of card payments.
- Token-based secure payment was established to process transactions and update order statuses in real-time.

8.4 Deployment and Testing

- Postman was used to test backend APIs, ensuring endpoints for login, product retrieval, and order placement function correctly.
- Feedback was collected from users who accessed the deployed site, helping refine the UI and improve payment flow.
- Manual Testing of payment workflows confirmed successful Stripe-based order placement.

For Source code and code review check the link below

GitHub Link -
https://github.com/AbhinandanGaikwad02/Ecommerce-MERN-Stack-.git

CHAPTER 8

SNAPSHOT

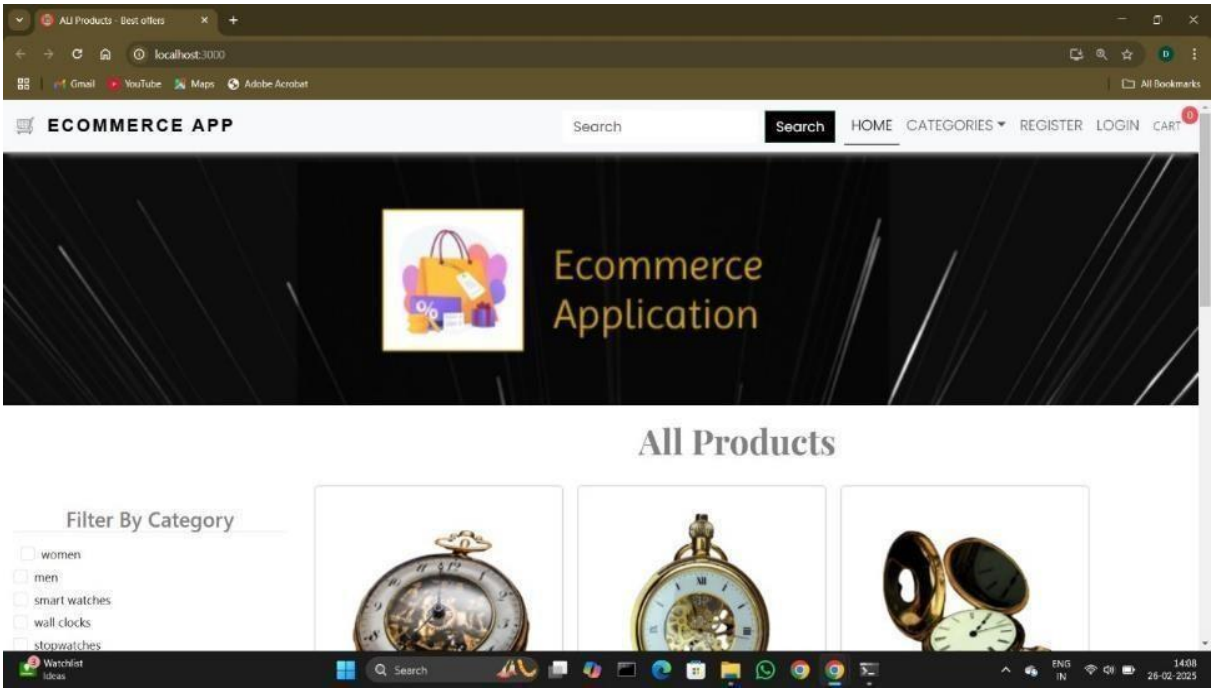

Fig 8.1 : Front Page of Ecommerce app, for users interaction.

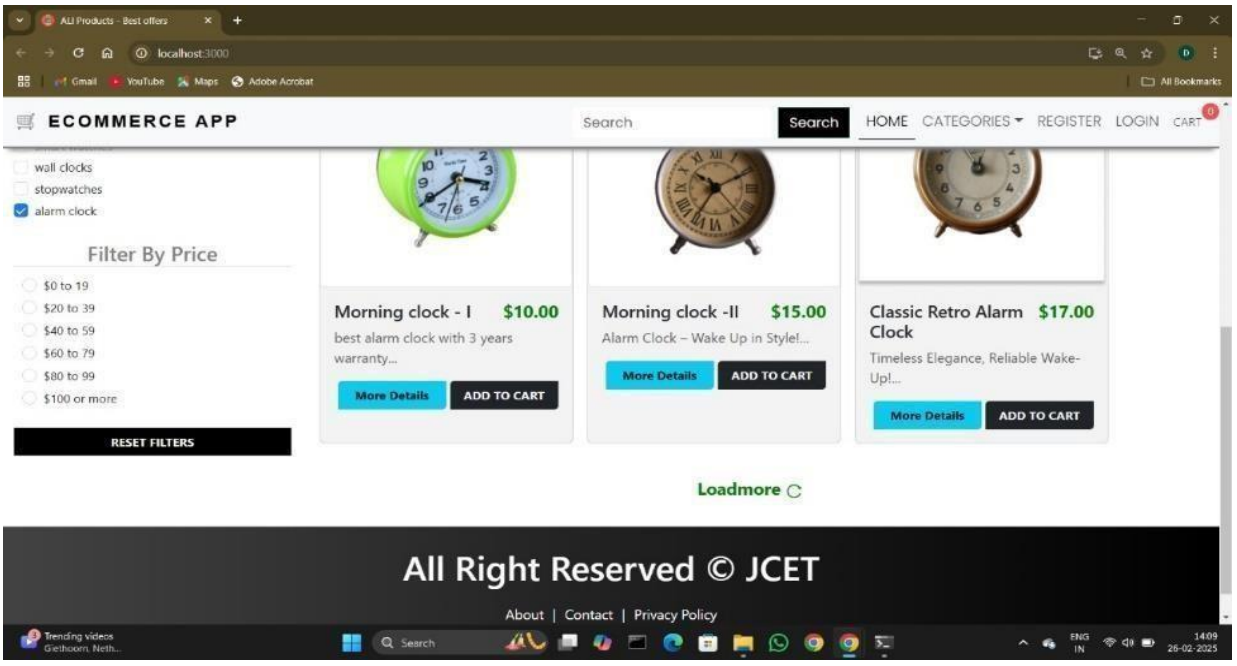

Fig 8.2 : The admin dashboard provides an overview of total products, sales, low-stock alerts, and access to all product management.

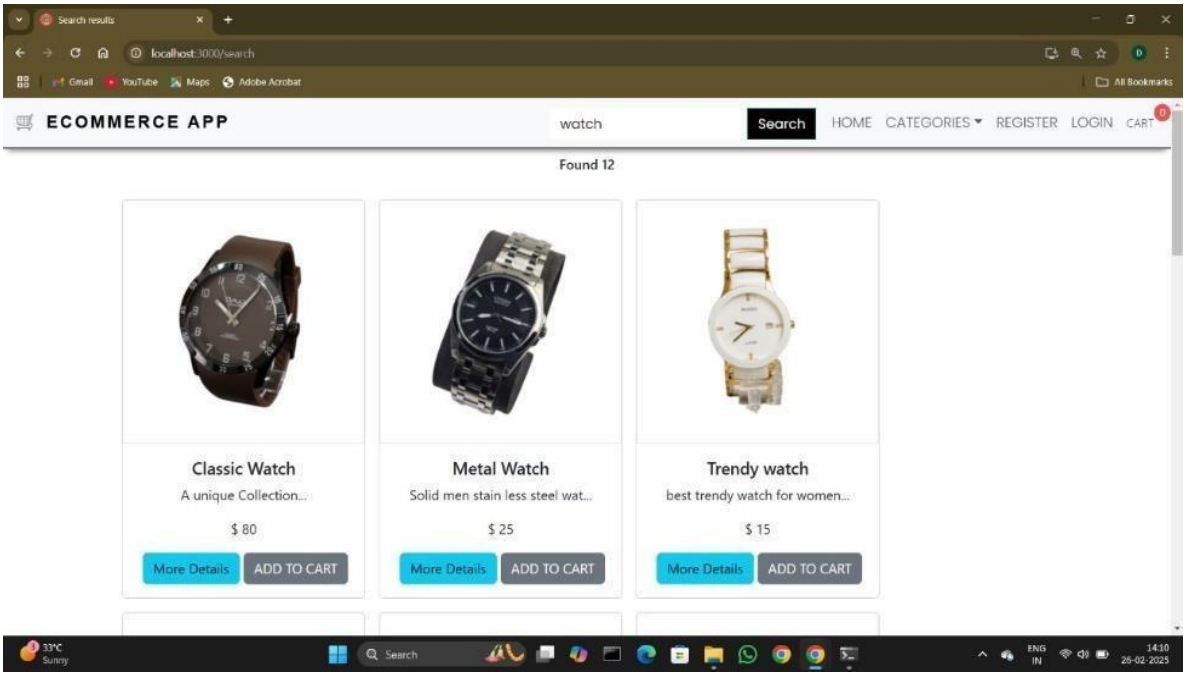

Fig 8.3 : The admin dashboard provides an overview of total products, sales, low-stock alerts, and access to key modules such as product management, billing, and reports.

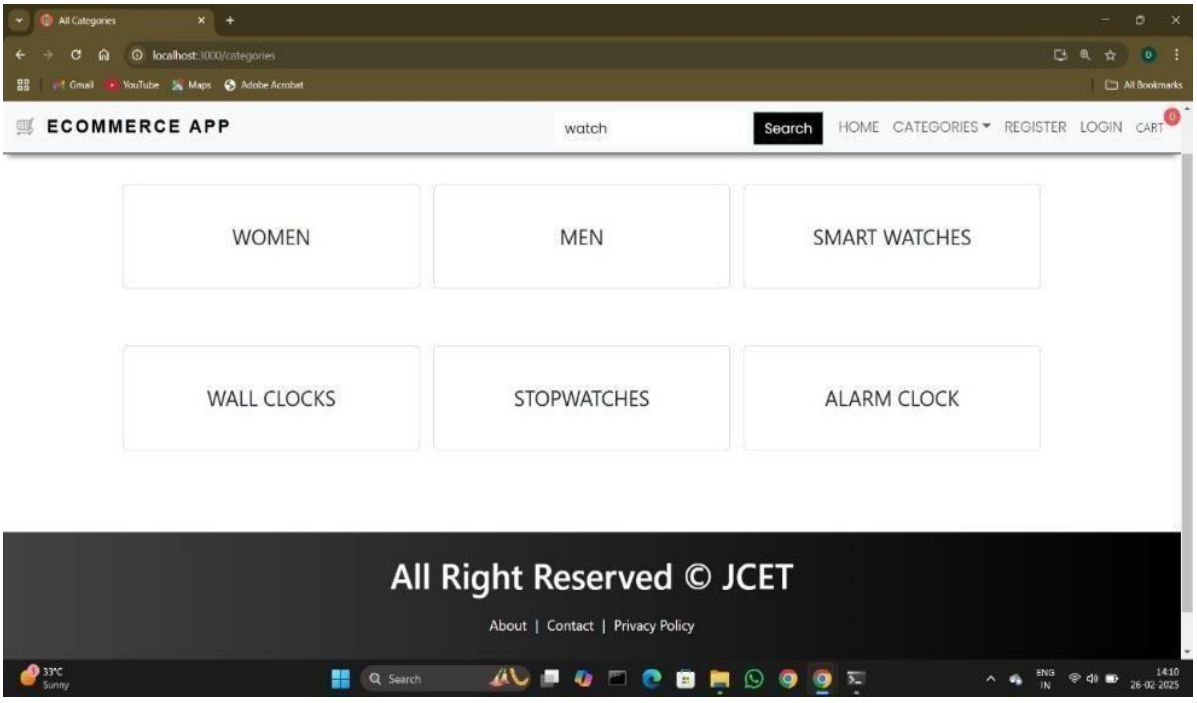

Fig 8.4 : Displays options for making easy to search watches and clocks.

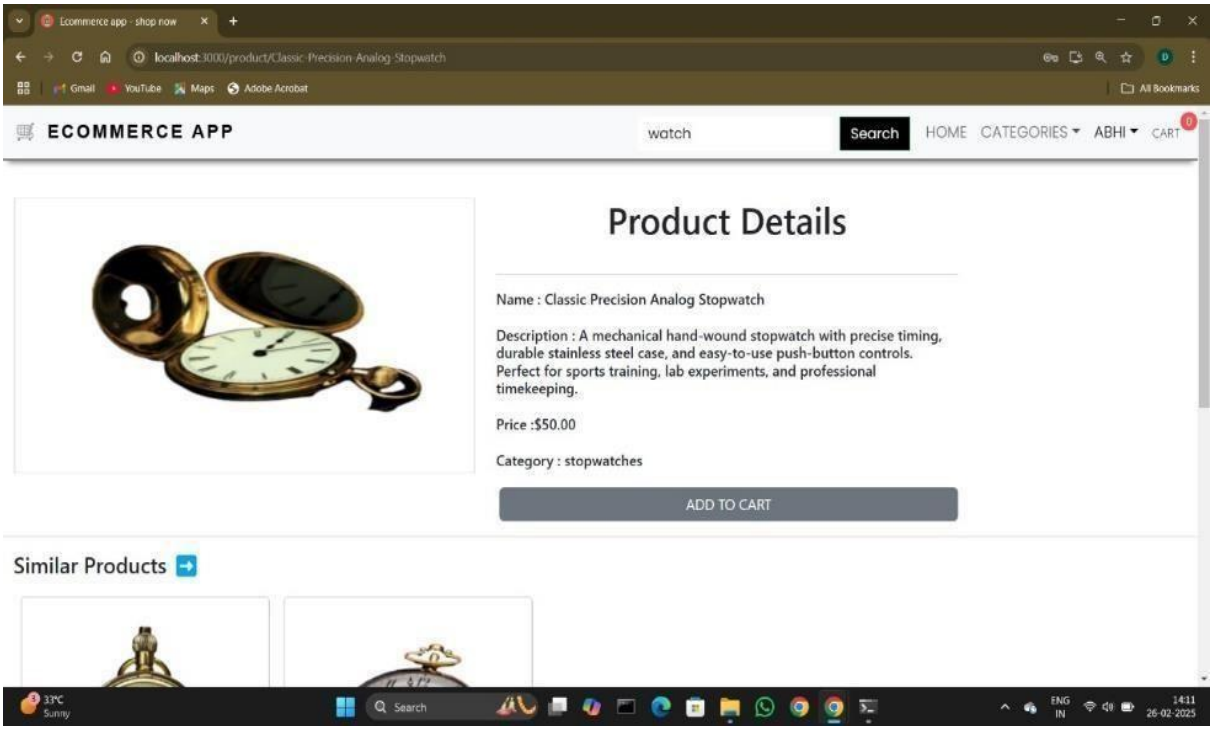

Fig 8.5 : Managing and providing product details Product name, Description, Price, Category.

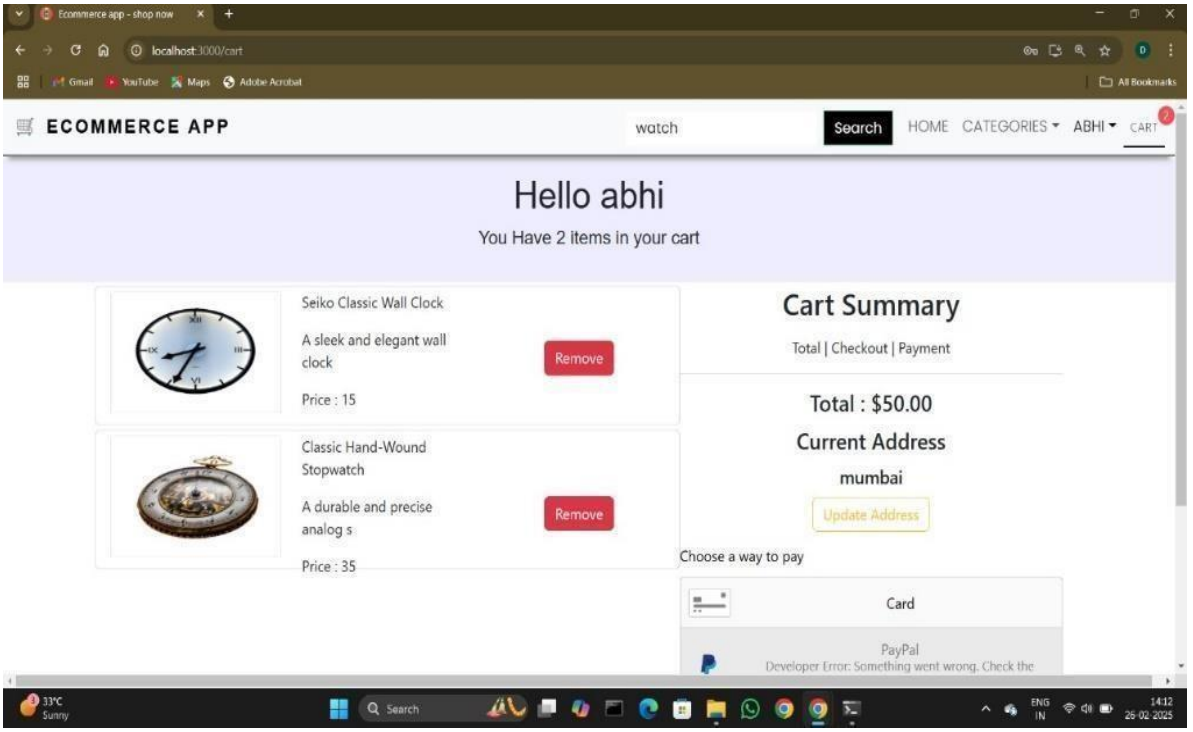

Fig 8.6 : Displays and manages customer information, including cart summary.

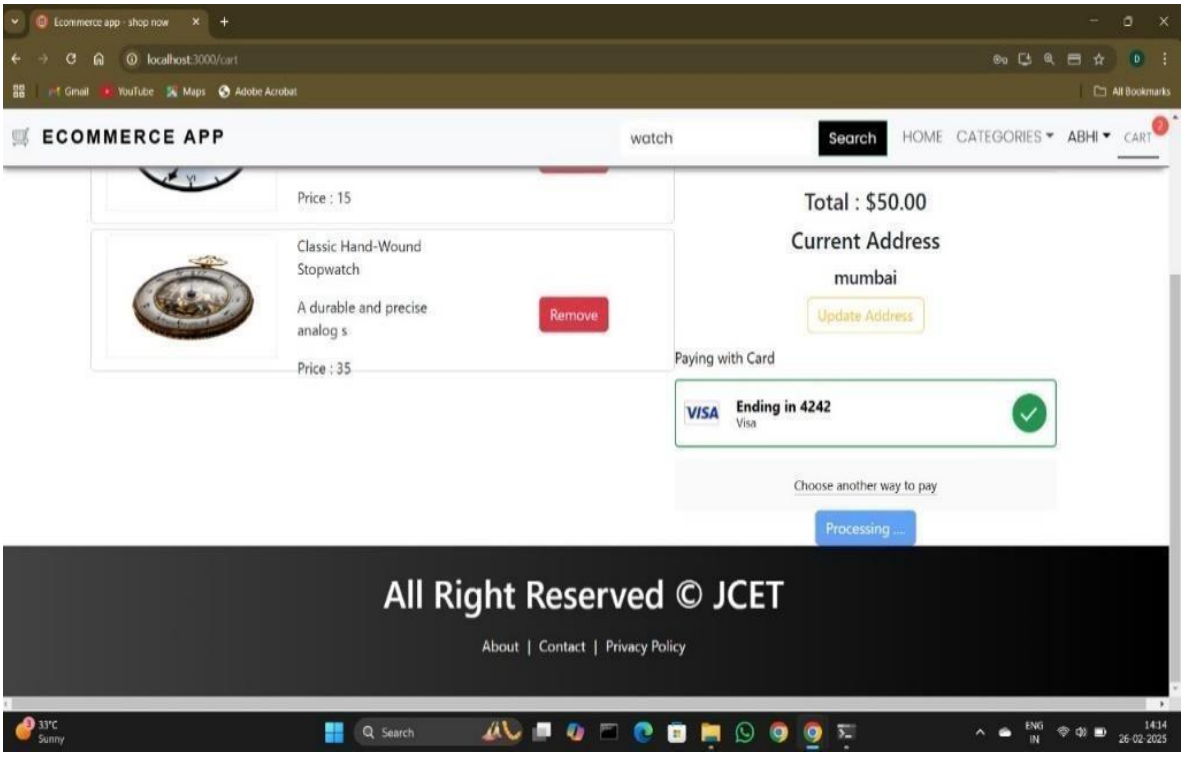

Fig 8.7 : The billing module enables users to select which payment they want to pay.

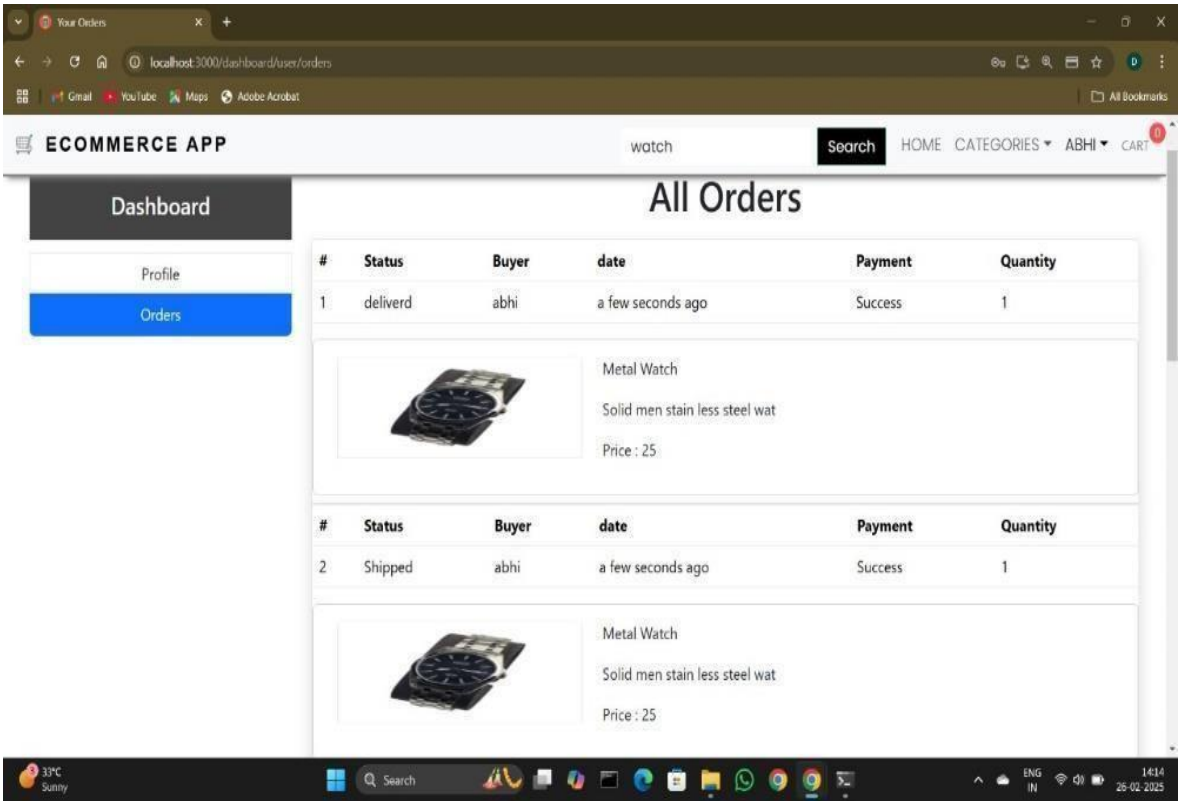

Fig 8.8 : Displays all orders history with status, buyer name, date, payment, and Quantity.

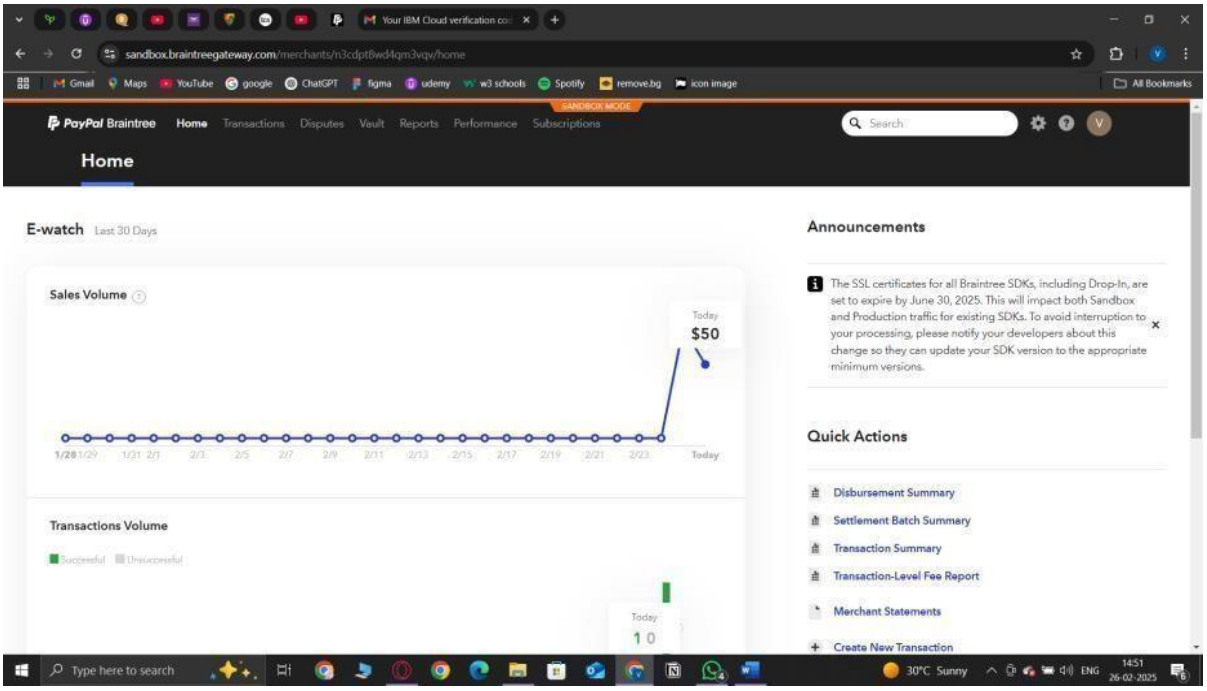

Fig 8.9: Displays a particular product sale volume and transaction volume.

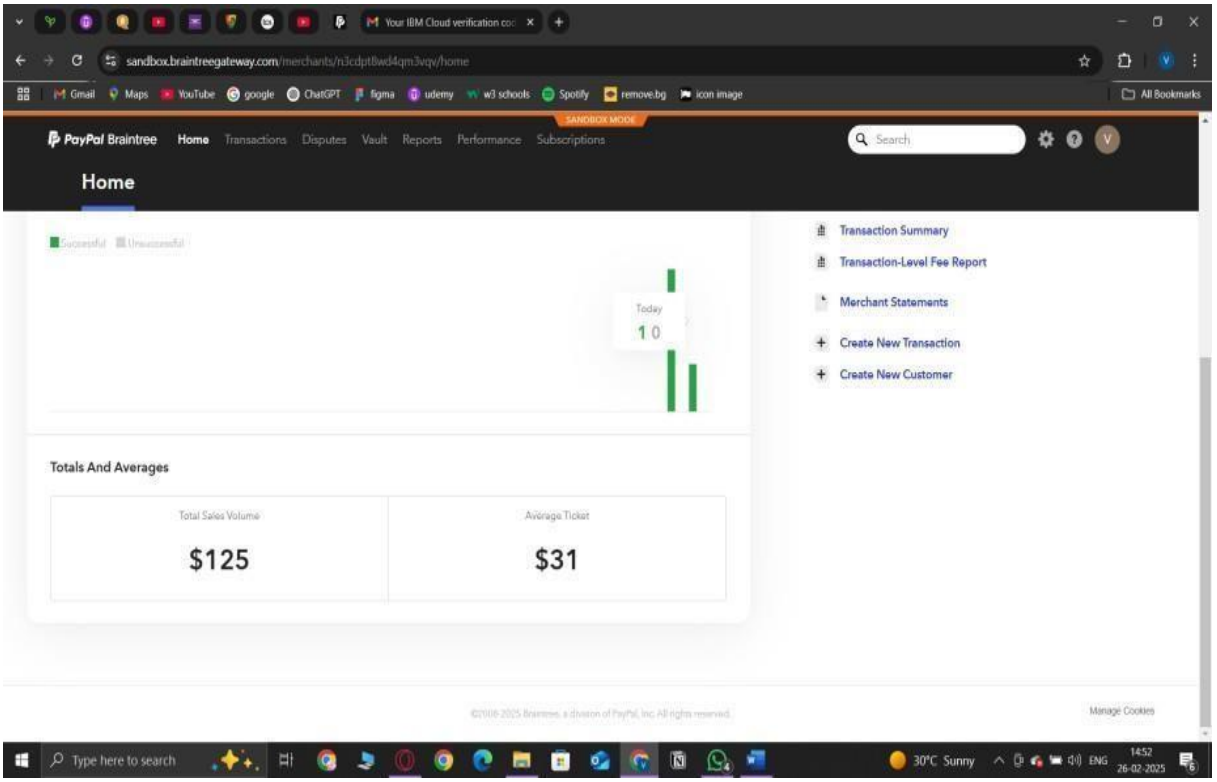

Fig 9.0 : Displays all saled product volume in graph and calender ,with transaction summary.

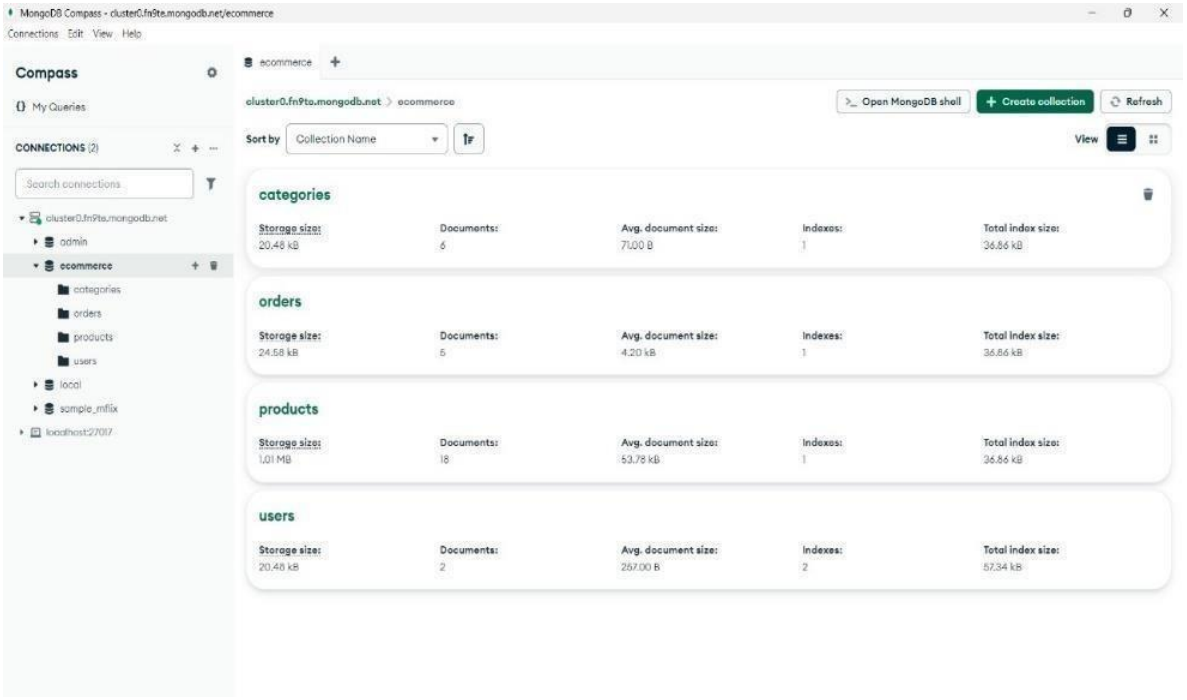

Fig 9.1 : Displays a product management which include category, orders, products, and users.

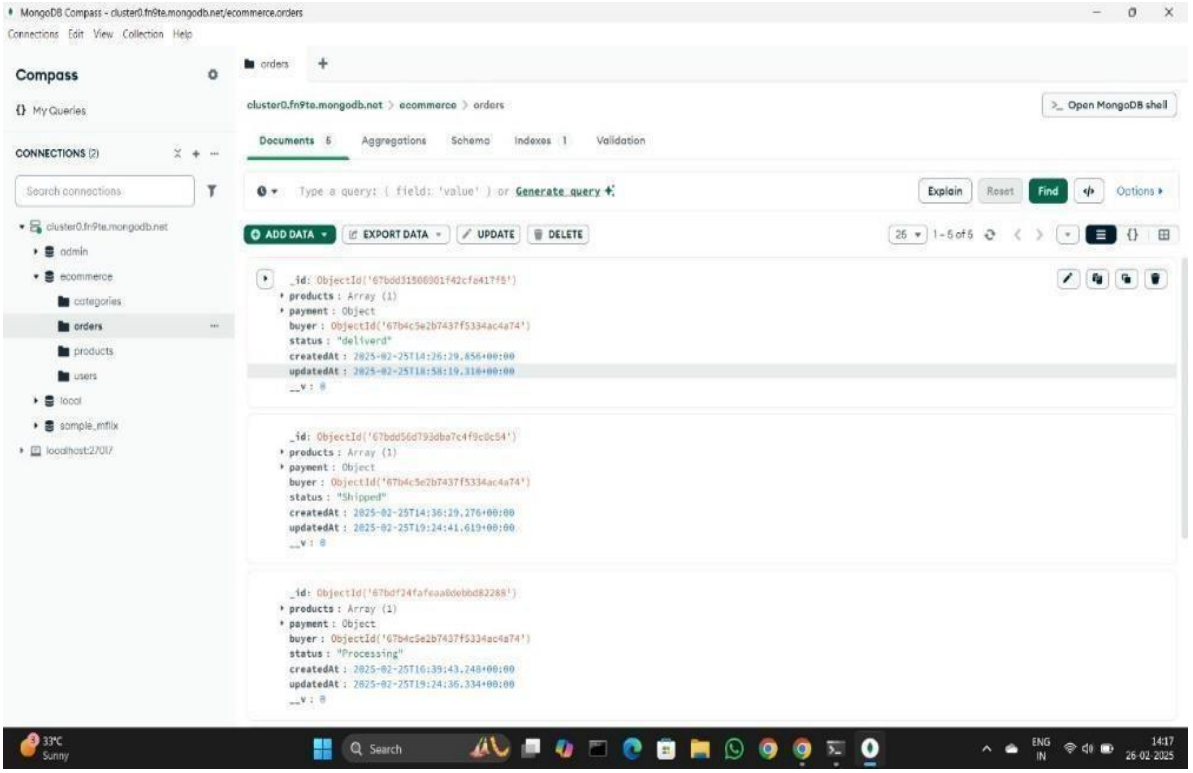

Fig 9.2 : Showing stored data and giving options to add data, export data, update, and delete.

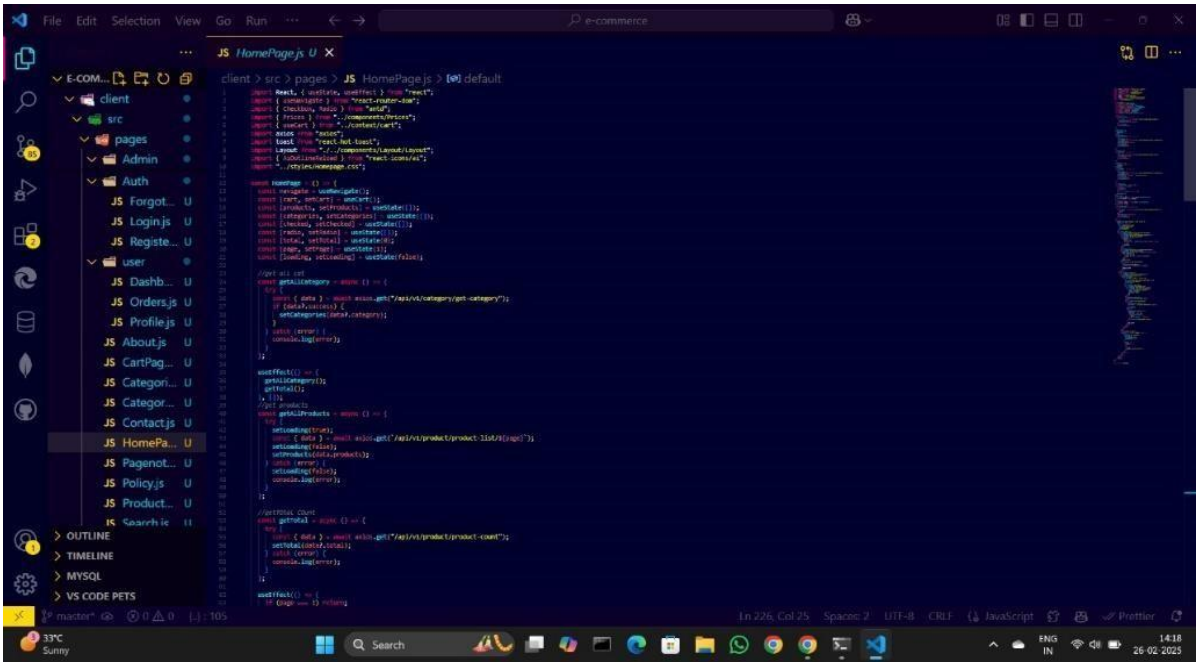

Fig 9.3 : Displaying the Home Page component which fetches product data from the backend and dynamically renders product cards with options to view individual product details.

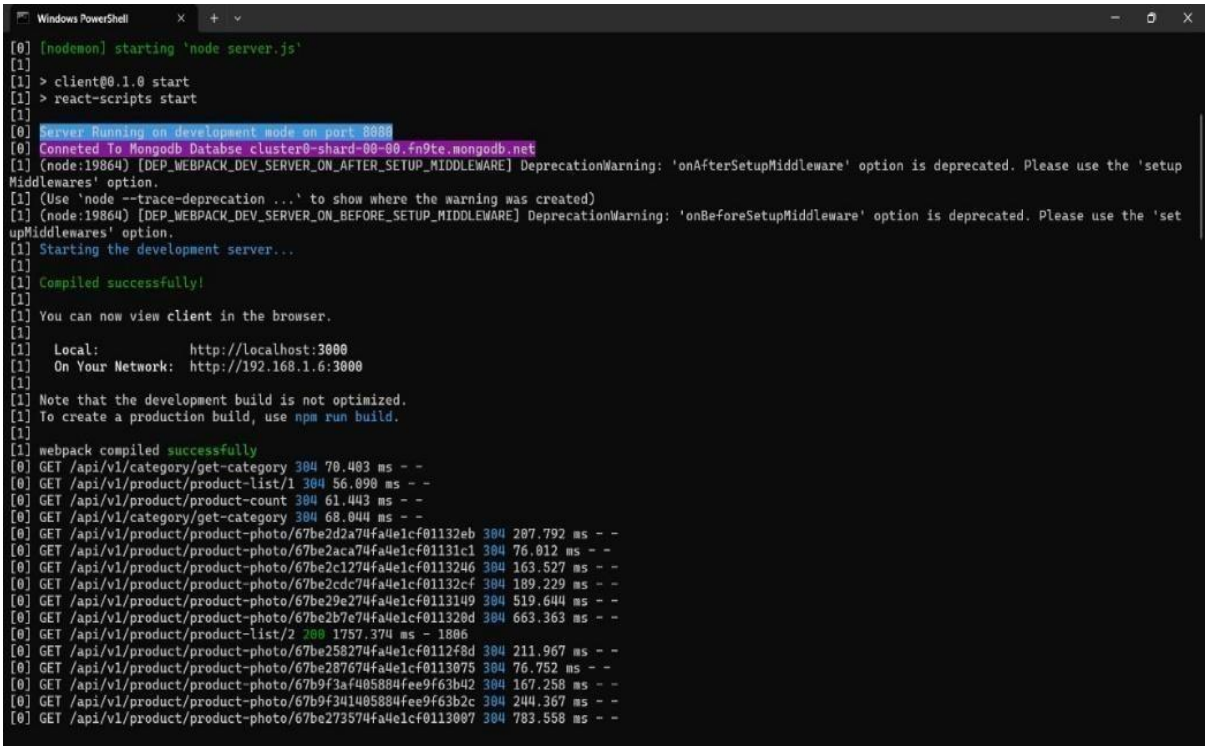

Fig 9.4 : Showing successful server startup with MongoDB connection and frontend running on localhost with product APIs loading.

CONCLUSION

The “Ecommerce Platform Integration: React.js and Node.js Synchronization” project effectively showcases the creation of a multi-vendor marketplace with scalability and security. The project emphasizes real-time synchronization of information, seamless user experience, and admin management. Implementation through React.js for the front-end and Node.js using Express.js as the back-end technology coupled with MongoDB for storing data ensures efficiency and reliability along with responsiveness. Features such as JWT-based authentication, bcrypt.js-hashed passwords, and payment gateway support through Stripe and PayPal are all integrated. Full control is also given for product management, order tracking, and user management through the admin panel. Real-time updates have been introduced for better user experience and accurate information. In spite of experiencing issues such as version compatibility and payment issues, troubleshooting issues were successfully resolved to deliver a fully operational product. Overall, the project improved our knowledge of full-stack development and gave us hands-on experience in the development of real- world web apps.

REFERENCES

[1] From MongoDB to React: Unleashing the Power of MERN in **E-commerce**
H Shwetha, D Prajwal… - … Conference on Emerging …, 2024 - ieeexplore.ieee.org

[2] Y. Fan, "Development of ecommerce app" 2010 2nd IEEE International Conference on Information Management and Engineering, 2010, pp. 207-210, doi: 10.1109/ICIME.2010.5478077.

[3] Node.js – JavaScript runtime built on Chrome's V8 JavaScript engine. https://nodejs.org

[4] Express.js – Fast, unopinionated, minimalist web framework for Node.js. https://expressjs.com

[5] T Huynh - 2020 - theseus.fi , The design and development of an e-commerce web application

[6] ReactJS. (2024). *React – A JavaScript library for building user interfaces*. Retrieved from: https://reactjs.org/docs/getting-started.html

[7] A Patil, M Patil, S Tondare… - … of **Integrated** …, 2025 - pubs.thesciencein.org
Managing **synchronous** requests was a significant task in server development, because blocking resulted in either inefficient or wasteful use of the resources.

[8] GitHub Repository – Project Source Code. https://github.com/AbhinandanGaikwad02/E-commerce-MERN-Stack-.git

[9] Stripe API – Online payment processing for internet businesses. https://stripe.com/docs/api

OJT CERTIFICATE

ROOMAN
INNOVATE INTEGRATE EMPOWER
ISO 9001 : 2005 & CMMI LEVEL - 3 COMPANY

www.rooman.com
CIN : U72900KA1999PTC025311

On the Job Training Certificate

Issued on: 03-03-2025

This is to certify that Ms/Mr **Abhinandan Gaikwad, CAN_33896109,** D/o, S/o **Shashikant Gaikwad** has successfully completed the On-the-Job training (OJT) for the course **Application Developer - Web & Mobile** conducted by **Rooman Technologies** facilitated by **TC276314,** Visvesvaraya Technological University (TP_ID: TP199028) from 12/02/2025 to 28/02/2025 at Rooman Technologies Pvt. Ltd. Bengaluru.

Sincerely,

Authorized Signatory

ROOMAN TECHNOLOGIES PVT LTD BANGALORE

Corporate Office : #130, Dr. Rajkumar Road, Ist Block Rajajinagar, Bangalore - 560010.
Phone : 0794822 1002, Email : vtu@rooman.net, Web : www.rooman.com

PMKVY CERTIFICATE

आईटी–आईटीईएस सेक्टर स्किल्स काउंसिल
IT-ITeS Sector Skill Council

राष्ट्रीय व्यावसायिक शिक्षा एवं प्रशिक्षण परिषद द्वारा मान्यता प्राप्त
Recognised by NCVET

कौशल योग्यता प्रमाणपत्र
Certificate for Skill Competency

प्रमाणपत्रसंख्या
Certificate No:
AEKAA0021QG-05-IT-00512-2023-V1.1-NASSCOM-072638

प्रमाणित किया जाता है कि श्री / सुश्री / एमएक्स
This is to certify that Mr./Ms./Mx Mr. Abhinandan Gaikwad

सुपुत्र
Son of Shashikant Gaikwad

जन्म तिथि
Date of Birth 02/04/2003

नामांकन संख्या
Enrolment No CAN_33896109

ने जॉब रोल / अर्हता का आंकलन सफलतापूर्वक
has successfully cleared the assessment in the job role/qualification Application Developer Web Mobile

अवधि
of Duration 540 Hrs

अर्जित किया
having earned 18

क्रेडिट एनसीआरएफ / एनएसक्यूएफ स्तर
Credits at NCrF/NSQF Level 5

प्रशिक्षण केन्द्र
Training Centre Jain College Of Engineering & Technology, Hubballi

जिला
District DHARWAD

राज्य
State KARNATAKA

प्रतिशत / श्रेणी के साथ उत्तीर्ण किया।
with A %/Grade

जारी करने का स्थान
Place of Issue: Uttar Pradesh

जारी करने की तिथि
Date of Issue: 19/03/2025

PMKVY

IT - ITeS SSC nasscom

नाम Name:
Sindhu Gangadharan
पद Designation:
Chairperson
हस्ताक्षर Signature:

ई-सत्यापन लिंक
e-Verification link:
https://admin.skillindiadigital.gov.in/documentverification.nsdcIndia

NCrF - National Credit Framework
NSQF - National Skills Qualification Framework

Digitally Generated Certificate

WADHWANI CERTIFICATE

WADHWANI FOUNDATION

CERTIFICATE OF COMPLETION

This is to certify that

ABHINANDAN GAIKWAD

from **Rooman Technologies Pvt Ltd**

has successfully completed

Life Skills (Jeevan Kaushal) 2.0

on **December 01, 2024**

Ajay Kela

Ajay Kela
CEO
Wadhwani Foundation

Scan & verify

This certificate confirms the completion of 93 hours of course.

ROOMAN CERTIFICATE