A PRELIMENERY REPORT ON

E-COMMERCE WEBSITE WITH PAYMENT GATEWAY

SUBMITTED TO THE VISHWKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY (COMPUTER ENGINEERING)

SUBMITTED BY

AAKASH DAS   22111180

MUGDHA KAMATE   22110689

RUTUJA JADHAV   22110268

ASMITA MUDASHI   22110618

DEPARTMENT OF COMPUTER ENGINEERING

BRACT'S

VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY

SURVEY NO. 3/4, KONDHWA (BUDRUK), PUNE – 411048, MAHARASHTRA (INDIA).

➢ **Introduction**

An ecommerce website with a payment gateway allows users to browse products, add items to their cart, and securely complete purchases online. It includes features like product pages, a shopping cart, checkout process, user accounts, search functionality, responsive design, security measures, payment gateway integration, order management, customer support, and analytics . The motivation behind discussing an ecommerce website with a payment gateway lies in its relevance and importance in today's digital economy. The main motive behind this project is:

- Ecommerce offers businesses of all sizes the opportunity to reach a global audience and sell products or services 24/7, without the limitations of physical storefronts.
- Online shopping provides convenience to consumers who can browse, compare, and purchase products from the comfort of their homes or on the go, saving time and effort.
- The evolution of technology and the internet has made it easier than ever to set up ecommerce platforms and integrate secure payment gateways, enabling businesses to expand their online presence.

➢ **Problems and Objectives**:

The problem we're addressing is the need for businesses to establish a robust online presence through an ecommerce website integrated with a secure payment gateway. This arises from the growing demand for online shopping and the necessity for businesses to adapt to changing consumer behaviour and market trends. However, many businesses face challenges in setting up and managing an effective ecommerce platform with seamless payment processing, leading to lost revenue opportunities and decreased competitiveness in the digital marketplace.

- Develop an ecommerce website that provides a user-friendly interface for browsing products, adding items to the cart, and completing purchases securely.
- Integrate a reliable payment gateway to facilitate seamless and secure online transactions, ensuring the protection of sensitive customer information.
- Optimize the website for various devices and screen sizes to deliver a consistent and responsive shopping experience across desktops, tablets, and smartphones.
- Implement robust security measures, including SSL encryption and PCI compliance, to safeguard customer data and instill trust in online transactions.
- Enhance user engagement and retention through features such as user accounts, order tracking, and personalized recommendations.

Methodology of Problem Solving:

Developing an e-commerce payment gateway involves several steps, from planning and design to implementation and deployment. Here's a methodology to guide you through the process:

1. Understanding Requirements:

  - Gather requirements from stakeholders: Understand the features they want, payment methods to support, security requirements, scalability needs, etc.

  - Define user stories or use cases: Break down the requirements into manageable tasks.

2. Research and Planning:

  - Research existing payment gateway solutions: Understand how they work, their features, limitations, and best practices.

  - Decide on technology stack: Choose programming languages, frameworks, libraries, and databases based on project requirements, team expertise, and scalability considerations.

  - Plan the architecture: Define the overall system architecture, including frontend, backend, database, and third-party integrations.

3. Design Phase:

  - Design database schema: Create a database schema to store user accounts, transactions, payment methods, etc.

  - Design API contracts: Define the RESTful endpoints or GraphQL queries for communication between frontend and backend.

  - UI/UX design: Design user interfaces for the website or mobile app, focusing on usability, accessibility, and responsiveness.

4. Development:

  - Frontend Development: Implement the user interfaces using HTML, CSS, and JavaScript frameworks like React, Angular, or Vue.js.

  - Backend Development: Implement the server-side logic using programming

Literature Survey:

- Ecommerce Trends and Market Analysis: Reviewing industry reports and research papers to understand the current trends, growth projections, and market dynamics of ecommerce, including the adoption of payment gateways.
- User Experience (UX) Design: Exploring literature on UX design principles, usability studies, and best practices for designing intuitive and user-friendly ecommerce websites, particularly focusing on the checkout process and payment gateway integration.
- Payment Gateway Technologies: Investigating the different types of payment gateways available (e.g., hosted vs. integrated) and their respective features, security protocols, transaction fees, and integration methods.

- Security and Trust: Examining research on security measures, encryption standards, and trust indicators implemented by ecommerce websites to protect customer data and enhance trust during online transactions.
- Consumer Behaviour and Trust Factors: Analyzing studies on consumer behaviour in ecommerce, including factors influencing purchase decisions, perceptions of payment security, and trust-building strategies employed by ecommerce businesses.

System design:

The system design of an ecommerce website with a payment gateway involves:

- Architecture: Client-side for UI and server-side for backend processing.
- UI Components: Homepage, product pages, shopping cart, and checkout process.
- Backend Services: User management, product management, order management, and payment gateway integration.
- Database Design: User data, product data, and order data storage.
- Security Measures: SSL encryption, PCI compliance, authentication, and authorization.
- Integration with Payment Gateway: Selection, integration, and implementation of payment processing logic.
- Scalability and Performance: Design for scalability and optimize performance.
- Monitoring and Analytics: Implement monitoring tools and analyze user behavior.
- Compliance and Regulations: Ensure compliance with relevant laws and regulations.
- Testing and Quality Assurance: Conduct comprehensive testing before deployment.

➢ **Project Implementation:**

➢ **Modules:**

User Management Module:

- User registration, login, and authentication.
- Profile management (view/update profile information).
- Address management for shipping and billing.

Product Management Module:

- Product catalogue management (add/edit/delete products).
- Category/subcategory management for organizing products.
- Inventory management (manage stock levels, track product availability).

Shopping Cart Module:

- Add/remove products to/from cart.
- Update cart quantities.
- Calculate subtotal, taxes, and total order amount.

Checkout Module:

- Shipping information collection (address, shipping method).
- Payment method selection (integration with payment gateway).
- Order review and confirmation.

Order Management Module:

- Order processing (generate order confirmation, send confirmation email).
- Order tracking (view order status and history).
- Invoice generation and management.

Payment Gateway Integration Module:

- Integration with a payment gateway provider (e.g., RazorPay).

Technologies Used:

- Frontend (React):
- Overview: React is a JavaScript library for building user interfaces. It allows for the creation of interactive and dynamic UI components.
- Component-based architecture for building reusable UI elements.
- Virtual DOM for efficient rendering and performance optimization.
- State management with hooks or context API.
- Rich ecosystem with libraries like React Router for routing and Redux for state management.

Backend (Node.js with Express.js):

- Overview: Node.js is a JavaScript runtime environment that allows for server-side scripting. Express.js is a web application framework for Node.js, providing a robust set of features for building web servers and APIs.
- Asynchronous and event-driven architecture for handling multiple concurrent requests.
- Middleware support for handling HTTP requests, parsing JSON, and implementing authentication.
- Routing system for defining API endpoints and handling requests/responses.
- Integration with MongoDB using libraries like Mongoose for data modeling and interaction.

Database (MongoDB):

- Overview: MongoDB is a NoSQL database that stores data in JSON-like documents. It offers flexibility and scalability, making it suitable for handling diverse data types and high-volume workloads.
- Document-based storage model, allowing for flexible schema design.
- Rich query language with support for complex queries, indexing, and aggregation.
- Integration with Node.js using the official MongoDB Node.js driver or Mongoose ORM for simplified data manipulation.
- ➢ **Tools and Technologies Used:**

Version Control:

- Git for version control, allowing collaboration and tracking changes to the project codebase.

Development Tools:

- Integrated development environments (IDEs) like Visual Studio Code or WebStorm for code editing.
- Package managers like npm or Yarn for managing project dependencies

Result:

➢ **Conclusion:**

In conclusion, the use of React for the frontend, Node.js with Express.js for the backend, and MongoDB for the database offers a robust and scalable tech stack for building an ecommerce website. This combination allows for the development of a dynamic and interactive user interface, efficient server-side scripting, and flexible data storage and retrieval. By leveraging these technologies, developers can create a modern ecommerce platform that delivers a seamless shopping experience for users while enabling efficient management of products, orders, and payments.

Future Work:

- Improving user experience and interface.
- Enhancing search and filtering options.
- Optimizing for mobile devices.
- Adding internationalization and localization.
- Exploring AR and VR technologies.
- Investigating blockchain for secure transactions.
- Continuously optimizing performance.
- Integrating social commerce features.
- Implementing advanced analytics for data-driven decisions.

➢ **Applications:**

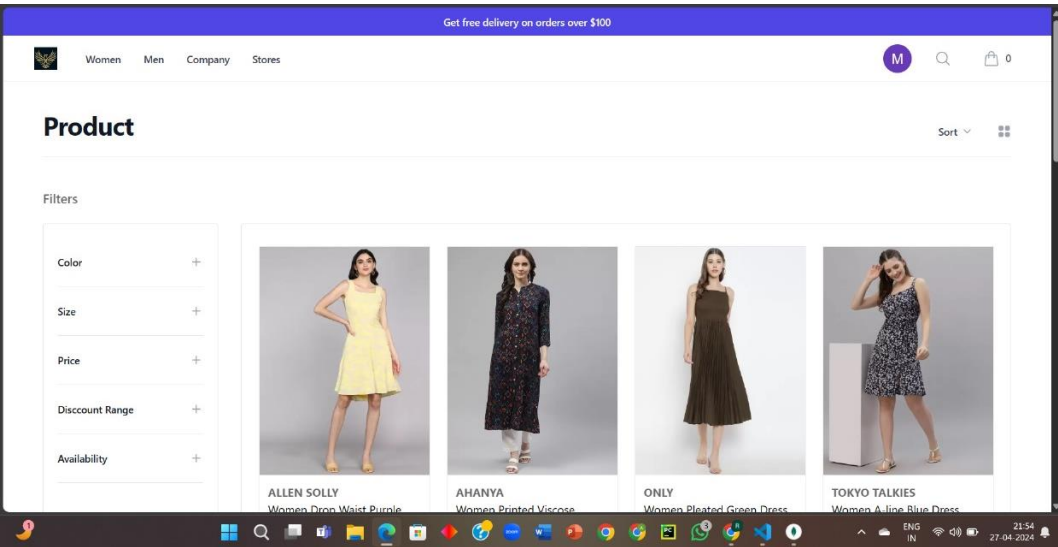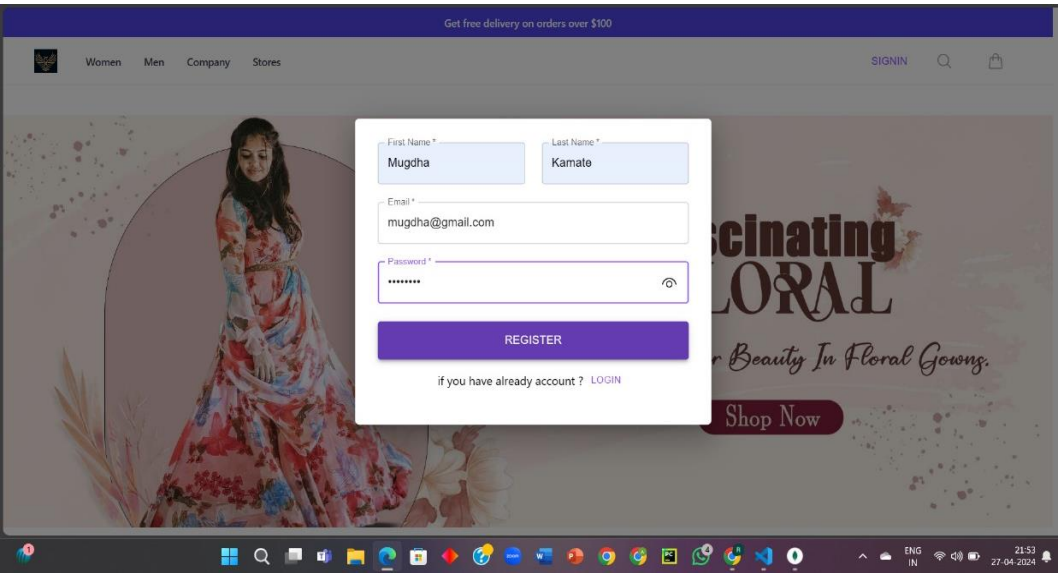For a clothing-specific ecommerce website, potential applications include:

- Online clothing store.
- Boutique fashion platform.
- Custom apparel store.
- Athleisure wear marketplace.
- Ethical and sustainable fashion platform.
- Fashion rental service.
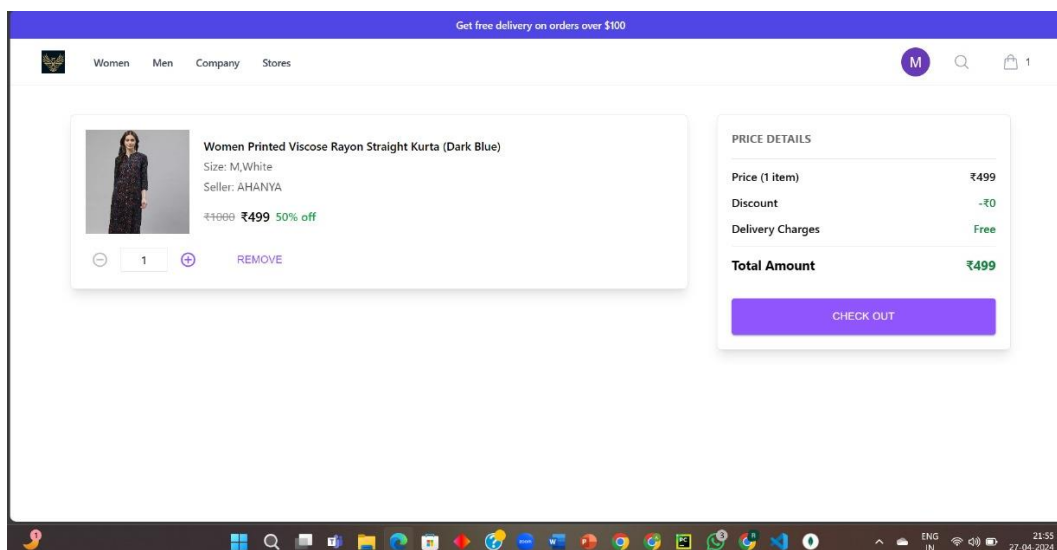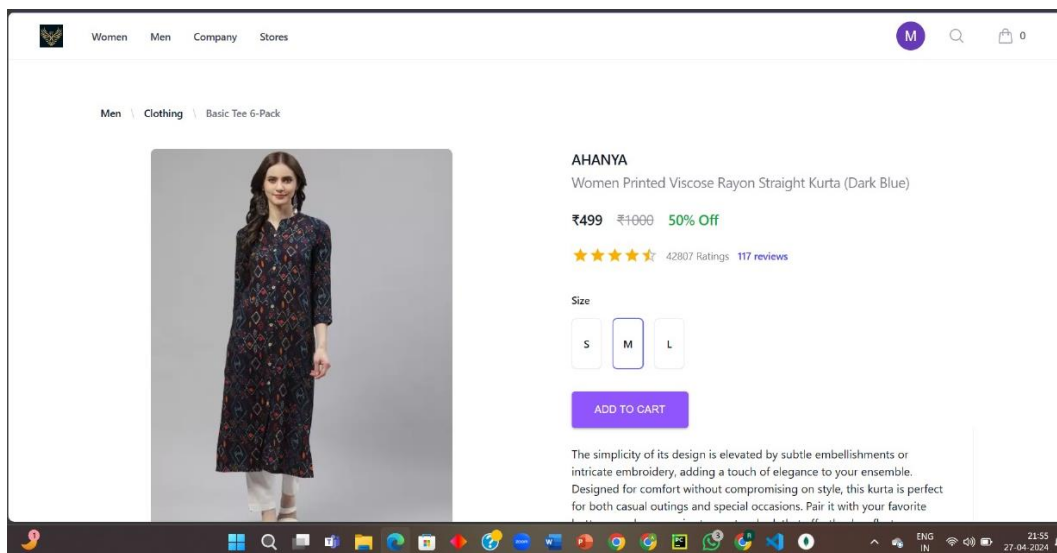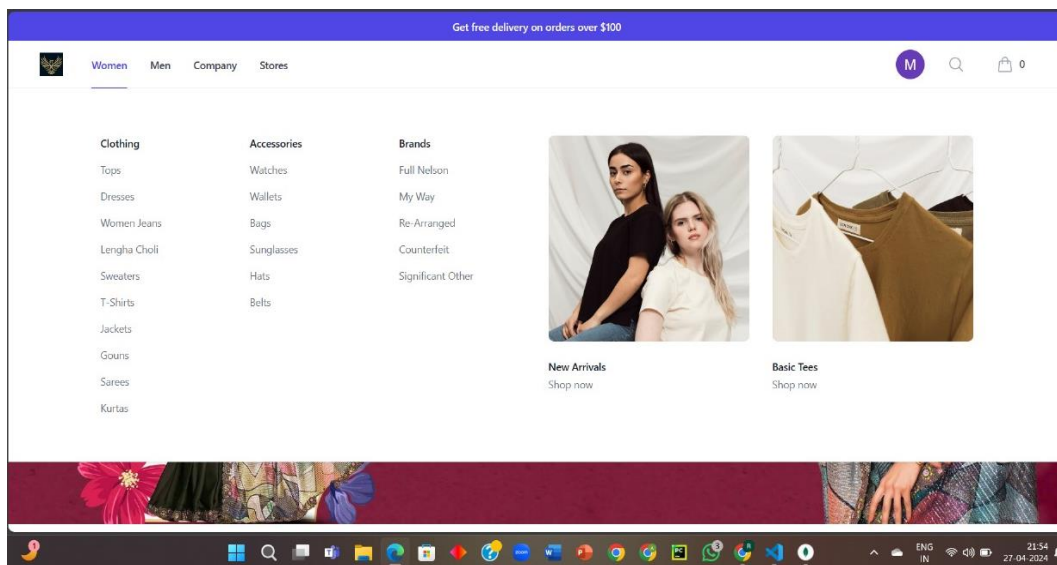- Curated fashion boxes.

➢ **Outcome**

HomePage
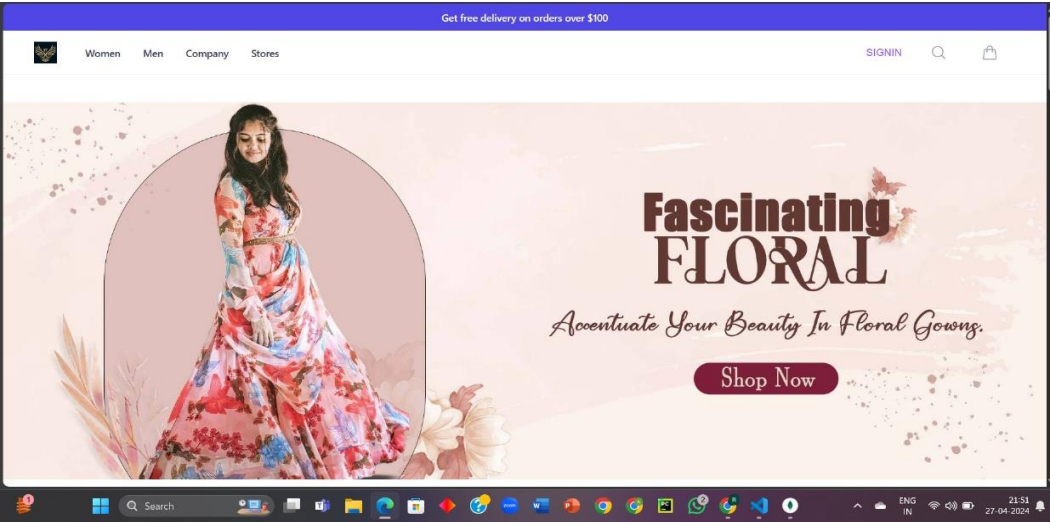
SignUp Page

This payment link is created in **Test Mode**. Only test payments can be made for this.

Payment Request from Shop With Zosh

AMOUNT PAYABLE
INR 499.00

Payment successful

₹ 499

Shop With Zosh
Apr 27, 2024 | 09:58 PM

Card | pay_O3iEpF7af1kDag
Visit razorpay.com/support for queries

Redirecting in 4 seconds...

Razorpay

UPI VISA RuPay

Want to create payment links for your business? Visit razorpay.com/payment-links and get started instantly.