*A project report on*

ECOMMERCE STORE

*Submitted in partial fulfillment for the award of the degree of*

**Integrated MTech CSE**

*by*

**21MIC7045 – CHAVA RISHIK**

**21MIC7114 – S JAYADRITHA**



**AMARAVATI**

**DEPARTMENT OF SCOPE**

July-2024

**CERTIFICATE**

This is to certify that the thesis entitled "ECOMMERCE STORE" submitted by CHAVA RISHIK(21MIC7045), S. JAYADRITHA(21MIC7114 scope, VIT-AP, for the award of the Summer Internship for the bonafide work carried out by him/her under my supervision.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The Project report fulfils the requirements and regulations of VIT-AP and in my opinion meets the necessary standards for submission.

**Signature of the Guide**

**CONTENTS**

**ABSTRACT…………………………………………………………………….…5**

**ACKNOWLEDGEMENT………………………………………………….…….6**

**INTRODUCTION** ………………………………………………......................... 7

1.1 Motivation……………………….......................................................... 7  
1.2 Problem Statement …………............................................................... 7  
1.3 Overview of the Project ……................................................................. 8  
1.4 Challenges …………………….............................................................. 8  
1.5 Objective of the Project ………....……………………......................... 9  
1.6 Scope …………………………..……………………........................... 10

1. **LITERATURE SURVEY** ......................................................................... 11

2.1 Related Work ......................................................................................... 11

1. **SYSTEM ANALYSIS** ................................................................................ 12

3.1 Existing System ..................................................................................... 12  
3.2 Disadvantages ........................................................................................ 13  
3.3 Proposed System .................................................................................... 13  
3.4 Advantages ............................................................................................. 13  
3.5 Work Flow of Proposed System ........................................................... 14

1. **REQUIREMENT ANALYSIS** ................................................................. 16

4.1 Functional and Non-functional Requirements ………………………... 17  
4.2 Hardware Requirements ......................................................................... 17  
4.3 Software Requirements .......................................................................... 18  
4.4 Architecture ………………………….................................................... 21

1. **METHODOLOGY** ..................................................................................... 22

5.1 YOLOv8 ................................................................................................ 23

1. **SYSTEM DESIGN** ..................................................................................... 26

6.1 Introduction to Input Design …………………………........................... 27  
6.2 UML Diagrams …………………………………...………………........ 28  
6.3 Data Flow Diagram ………………………….......................................... 36

1. **IMPLEMENTATION** .................................................................................. 39

7.1 Modules ................................................................................................... 39

1. **SYSTEM STUDY AND TESTING** ............................................................ 41

8.1 Feasibility Study ....................................................................................... 41  
8.2 Types of Tests ........................................................................................... 43

1. **CONCLUSION** ............................................................................................. 45
2. **FUTURE ENHANCEMENT** ....................................................................... 45
3. **REFERENCES** .............................................................................................. 46
4. **Appendix 1** …………………………………………………………………...47
5. **Appendix 2 …………………………………………………………………... 47-52**

**ABSTRACT**

In the digital age, eCommerce has revolutionized the way consumers shop and businesses operate. This eCommerce store is designed to provide a seamless, user-friendly online shopping experience, offering a diverse range of products to cater to various customer needs. The platform leverages advanced technology to ensure secure transactions, efficient logistics, and personalized customer service. Through innovative marketing strategies and data-driven insights, the store aims to enhance customer satisfaction and foster brand loyalty. The integration of AI and machine learning facilitates personalized recommendations, optimizing the shopping experience. Additionally, the eCommerce store is committed to sustainability and ethical practices, ensuring that all operations adhere to environmental and social responsibility standards. The ultimate goal is to create a trusted online marketplace where customers can enjoy convenient, reliable, and enjoyable shopping.

**ACKNOWLEDGEMENT**

It is my pleasure to express with deep sense of gratitude to Prof. Udit Narayana Kar, Designation, Scope, VIT-AP, for his constant guidance, continual encouragement, understanding; more than all, he taught me patience in my endeavour. My association with him is not confined to academics only, but it is a great opportunity on my part of work with an intellectual and expert in the field of SCOPE.

I would like to express my gratitude to DR. G. Vishwanathan Sir ,Dr . S. V. Kota Reddy, DR. CH. Pradeep Reddy, SCOPE, for providing with an environment to work in and for his inspiration during the tenure of the course.

In jubilant mood I express ingeniously my whole-hearted thanks to Dr. Rohit Kumar Das. Program chair pf MTSE, all teaching staff and members working as limbs of our university for their not-self-centred enthusiasm coupled with timely encouragements showered on me with zeal, which prompted the acquirement of the requisite knowledge to finalize my course study successfully. I would like to thank my parents for their support.

It is indeed a pleasure to thank my friends who persuaded and encouraged me to take up and complete this task. At last, but not least, I express my gratitude and appreciation to all those who have helped me directly or indirectly toward the successful completion of this project

Place: Amaravathi

Date:

Name of the student

**Chava Rishik**

**S. Jayadritha**

**1.INTRODUCTION**

1.1MOTIVATION

The rise of online shopping has transformed consumer habits, highlighting the need for a user-friendly eCommerce store that provides convenience and accessibility anytime, anywhere. By launching a digital platform, businesses can reach a global audience, boost revenue potential, and cut operational costs compared to physical stores.

An effective eCommerce app enhances brand visibility and fosters customer engagement through personalized experiences and targeted marketing. Leveraging technology allows for streamlined operations and innovative shopping experiences, which are essential in a market dominated by online retailers.

Moreover, the ability to collect valuable customer data empowers businesses to adapt their offerings to meet changing consumer demands. This app aims to create a seamless shopping experience by providing detailed product information alongside features like recommendations, cart management, and user preferences. With a focus on an intuitive interface and a strong backend, we’re committed to enhancing the overall online shopping journey.

* 1. PROBLEM STATEMENT

The primary challenge in retail today is the need to adapt to the evolving demands of consumers who seek convenient and efficient shopping experiences. Many traditional brick-and-mortar retailers find it difficult to scale their operations and expand their reach due to physical constraints and logistical issues.

As competition increases, businesses must differentiate themselves by enhancing customer engagement and providing tailored services. This shift necessitates innovative approaches to reach customers effectively. Establishing an eCommerce platform offers a solution by enabling businesses to overcome these limitations, reduce operational costs, and leverage valuable customer data. This not only enhances their ability to refine product offerings but also allows for more effective marketing strategies. Ultimately, addressing these challenges is crucial for delivering a shopping experience that meets the expectations of modern consumers.

1.3 OVERVIEW OF THE PROJECT

Developing an eCommerce store involves creating a digital platform where businesses can sell products or services online, aiming to reach a broader audience and meet the rising consumer demand for online shopping convenience. The process encompasses several key steps: conducting market research to understand target customers, selecting a suitable eCommerce platform or building a custom website, designing an intuitive and engaging user interface, setting up secure payment gateways, and implementing effective logistics and inventory management systems.

Additionally, integrating marketing tools for SEO, social media, and email campaigns is crucial to drive traffic and boost sales. Continuous analysis and optimization based on customer feedback and data insights ensure the store remains competitive and responsive to market trends. This comprehensive approach ultimately enables businesses to enhance their market presence, improve customer satisfaction, and achieve sustainable growth.

* 1. CHALLENGES PRESENT

ECommerce store presents several challenges, starting with the need for a deep understanding of the target market to ensure the platform meets consumer expectations. Technical hurdles include selecting the right eCommerce platform, ensuring seamless integration of payment gateways, and setting up robust security measures to protect customer data. Additionally, creating a user-friendly and visually appealing interface requires careful design and usability testing.

Logistics and inventory management can be complex, particularly for businesses with a large product range or global shipping requirements. Marketing and driving traffic to the site necessitate effective SEO strategies, engaging content, and continuous social media and email marketing efforts. Moreover, the competitive nature of the online marketplace demands constant innovation and adaptation to changing trends and technologies. Balancing these technical, operational, and strategic aspects while maintaining a focus on customer experience and satisfaction is a significant challenge in the development and ongoing management of an eCommerce store.

* 1. **OBJECTIVES**

Developing an eCommerce store is motivated by various factors that align with market demands and business goals.

1. **Market Demand:**  
   The shift in consumer behavior towards online shopping is increasingly evident, with more people preferring the convenience of browsing and purchasing products from the comfort of their homes. The ability to shop 24/7 from any location caters to the modern consumer's lifestyle, making online platforms an essential part of retail.
2. **Business Growth:**  
   Establishing an eCommerce store offers significant opportunities for business growth. Without the geographical limitations of a physical store, businesses can scale operations more easily and reach a broader audience. Additionally, eCommerce generally incurs lower operational costs compared to traditional retail, allowing businesses to allocate resources more efficiently.
3. **Revenue Potential:**  
   The potential for increased revenue is a strong motivator for developing an online store. By tapping into a global market, businesses can significantly boost sales and expand their customer base. Furthermore, an eCommerce platform allows for a more diverse range of products and services, catering to various consumer needs and preferences.
4. **Brand Presence:**  
   Enhancing brand presence in the digital marketplace is crucial for long-term success. An effective eCommerce platform increases online visibility, making it easier for potential customers to discover and engage with the brand. Personalized experiences and targeted marketing strategies help build stronger relationships with customers, fostering loyalty and encouraging repeat business.
5. **Technological Advancements:**  
   Utilizing the latest technology plays a vital role in improving the shopping experience. Innovations such as AI-driven recommendations and virtual try-ons enhance customer engagement and satisfaction. Additionally, automating processes like inventory management, order processing, and customer service streamlines operations, allowing businesses to focus on growth.
6. **Competitive Advantage:**  
   In an industry increasingly dominated by online retailers, staying competitive is essential. Developing an eCommerce store provides a significant market edge by allowing businesses to adapt to changing consumer expectations and market trends. By embracing digital transformation, companies can remain relevant and competitive.
7. **Customer Insights:**  
   An eCommerce platform offers valuable opportunities for data collection, providing insights into customer preferences, behavior, and trends. This data can be leveraged to personalize marketing strategies and product offerings, ensuring that businesses meet the specific needs of their customers and enhance overall satisfaction.
8. **Flexibility and Adaptability:**  
   The flexibility of an eCommerce store allows for easier updates and expansions of product ranges based on market demand. Additionally, businesses can implement dynamic pricing, flash sales, and other promotional strategies more efficiently, responding quickly to changing consumer behavior and maximizing sales potential.

**SCOPE OF THE PROJECT**

The scope of the eCommerce store project encompasses the comprehensive process of researching the market to identify target audiences and industry trends, selecting and setting up the appropriate eCommerce platform, and designing a user-friendly and responsive interface. It involves integrating essential tools such as secure payment gateways and inventory management systems while ensuring robust security measures and compliance with data protection regulations. The project includes creating and managing content, rigorous testing for functionality and performance, and launching the store with a strategic promotion plan. Additionally, the scope covers ongoing customer support, regular maintenance and updates, and implementing analytics to track user behaviour and sales performance. This holistic approach ensures the eCommerce store is well-equipped to meet customer needs, drive sales, and support business growth.

9.4 INTRODUCTION

The eCommerce store project is designed to create a dynamic and accessible online platform for the purchase and sale of products or services, catering to the increasing consumer preference for shopping from anywhere at any time. This digital solution is essential for businesses seeking to expand their market reach and enhance their operational efficiency.

An eCommerce store offers significant advantages, including the ability to operate 24/7, thereby providing convenience and flexibility for both customers and businesses. It reduces the overhead costs associated with physical storefronts, allowing for broader market access without geographical limitations. Additionally, the store can leverage advanced analytics to gain insights into customer behaviour, personalize marketing efforts, and optimize inventory management. The integration of secure payment systems and effective customer support ensures a trustworthy and seamless shopping experience, fostering customer loyalty and driving growth. Overall, an eCommerce store is a crucial tool for modernizing business operations and capturing opportunities in the expanding digital marketplace.

**2.LITERATURE SURVEY**

**2.1 RELATED WORK**

The landscape of eCommerce is rapidly evolving, driven by technological advancements, changing consumer behaviors, and emerging market trends. Research indicates that consumer behavior is significantly influenced by convenience, product variety, and price competitiveness, with a growing preference for online shopping linked to the desire for instant gratification and easy product comparisons. Technological innovations, particularly in artificial intelligence and machine learning, are crucial for personalizing shopping experiences through recommendation systems, while virtual and augmented reality enhance product visualization. However, security concerns remain a significant barrier to online transactions, highlighting the importance of building trust through secure payment gateways and transparent privacy policies. Effective digital marketing strategies, such as targeted campaigns and social media engagement, drive traffic and boost sales, with personalization based on data analytics increasing conversion rates. Additionally, efficient logistics and supply chain management are critical for timely order fulfillment, and automated inventory systems can streamline operations. Customer feedback plays a vital role in continuous improvement, as businesses that actively seek and respond to feedback are more likely to enhance offerings and build loyalty. Looking ahead, the future of eCommerce will be shaped by trends such as mobile commerce growth and increasing consumer demand for sustainability, prompting businesses to adapt to these changes.

**3. SYSTEM ANALYSIS**

**3.1Existing-System**  
Current eCommerce systems primarily consist of traditional online stores that offer limited customization and rely heavily on basic functionalities. These systems often lack advanced features such as personalized recommendations, seamless user interfaces, and robust data analytics capabilities. Many existing platforms face challenges in managing logistics, inventory, and customer relationships effectively, leading to inefficiencies in operations and suboptimal user experiences.

**Disadvantages of Existing System:**

* **Limited Personalization:** Existing systems often provide generic product recommendations, resulting in lower conversion rates and customer engagement.
* **User Experience Issues:** Many platforms lack intuitive designs, making navigation cumbersome and frustrating for users.
* **Security Concerns:** Current systems may not implement the latest security measures, leaving customer data vulnerable to breaches.
* **Inefficient Logistics Management:** Existing solutions often struggle with timely order fulfillment and effective inventory management, leading to delays and customer dissatisfaction.
* **Inflexibility:** Many systems are rigid in terms of features and customization, limiting businesses' ability to adapt to changing market trends.

**3.2Proposed-System**  
The proposed eCommerce system aims to address the limitations of existing solutions by incorporating advanced technologies and features that enhance user experience and operational efficiency. This system will include personalized product recommendations driven by AI, a user-friendly interface, secure payment processing, and automated logistics management. Additionally, the proposed system will leverage data analytics for better insights into customer behavior and preferences, enabling businesses to make informed decisions.

**Advantages of Proposed System:**

* **Enhanced Personalization:** The use of AI-driven recommendations will increase customer engagement and conversion rates by offering tailored product suggestions.
* **Improved User Experience:** An intuitive and responsive design will make navigation seamless, providing users with a satisfying shopping experience.
* **Robust Security Features:** Implementing the latest security protocols will protect customer data and build trust in the platform.
* **Efficient Logistics Management:** Automated inventory and logistics systems will streamline operations, ensuring timely order fulfillment and reducing costs.
* **Adaptability and Flexibility:** The proposed system will be more customizable, allowing businesses to adapt quickly to market changes and consumer demands.

**3.5 WORK FLOW OF PROPOSED SYSTEM**

**1. User Access and Interaction**

* **User Visits App**: The customer accesses the eCommerce store through a mobile app.
* **Account Creation/Login**: Users can create an account or log in to an existing account to access personalized features and saved information.

**2. Product Discovery**

* **Browse and Search**: Users browse through product categories or use the search function to find specific items.
* **Product Details**: Users view detailed product pages, including descriptions, images, prices, and reviews.

**3. Shopping Cart Management**

* **Add to Cart**: Users select products and add them to their shopping cart.
* **View Cart**: Users review their cart items, adjust quantities, or remove items if needed.
* **Cart Summary**: Displays a summary of selected items, subtotal, shipping costs, and estimated total.

**4. Checkout Process**

* **Initiate Checkout**: Users proceed to checkout, where they provide shipping information and choose a delivery method.
* **Payment Information**: Users enter payment details or select from saved payment methods.
* **Review Order**: Users review the final order details, including shipping address, payment method, and order summary.
* **Place Order**: Users confirm and place the order.

**5. Order Processing**

* **Order Confirmation**: The system sends an order confirmation email or notification to the user, including an order number and summary.
* **Inventory Update**: The system updates inventory levels based on the purchased items.
* **Payment Processing**: The payment gateway processes the payment and confirms the transaction.

**6. Fulfillment and Shipping**

* **Order Preparation**: The order is prepared for shipping, which may involve picking, packing, and labeling.
* **Shipping**: The order is dispatched to the chosen delivery carrier or logistics partner.
* **Tracking**: Users receive tracking information and can monitor the status of their shipment.

**7. Delivery**

* **Order Delivery**: The product is delivered to the customer’s specified address.
* **Delivery Confirmation**: The system may update the order status to "Delivered" and send a delivery confirmation to the user.

**8. Post-Delivery Support**

* **Customer Feedback**: Users can leave reviews or feedback about their purchase experience and the product.
* **Returns and Exchanges**: If needed, users initiate returns or exchanges through the customer service or returns process.

**9. Customer Service**

* **Support Requests**: Users can contact customer support for assistance with issues such as order status, refunds, or product inquiries.
* **Issue Resolution**: Customer service resolves issues and provides solutions based on the nature of the request.

**10. Analytics and Reporting**

* **Data Collection**: The system collects data on user behavior, sales performance, and other key metrics.
* **Analysis and Reporting**: Administrators analyze data to generate reports, track performance, and identify opportunities for improvement.

**4.REQUIREMENT ANALYSIS**

**4.1 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS**

**FUNCTIONAL COMPONENTS OF THE PROJECT**

**1. User Interface (UI) and User Experience (UX)**

* **Features**: Intuitive design, responsive layout, easy navigation, personalized user experience.
* **Function**: Provides an engaging and user-friendly shopping experience, ensuring customers can easily browse products, find information, and complete purchases.

**2. Product Catalog Management**

* **Features**: Organized product categories, detailed product pages with descriptions, images, prices, and reviews.
* **Function**: Manages and displays product information effectively, allowing customers to explore and choose products based on their preferences.

**3. Shopping Cart and Checkout System**

* **Features**: Persistent cart, easy-to-use interface, order summary, support for multiple payment options, guest checkout.
* **Function**: Enables customers to add products to their cart, review their selections, and complete purchases securely and efficiently.

**4. Payment Gateway Integration**

* **Features**: Support for various payment methods (credit/debit cards, digital wallets, bank transfers), secure encryption, fraud detection.
* **Function**: Facilitates safe and efficient payment processing, ensuring customer transactions are secure and compliant with financial regulations.

**5. Order Management System (OMS)**

* **Features**: Real-time order tracking, automated order processing, inventory synchronization, return and refund handling.
* **Function**: Manages the order lifecycle from placement to delivery, ensures accurate inventory levels, and handles customer returns and refunds efficiently.

**6. Inventory Management**

* **Features**: Stock level monitoring, low stock alerts, automatic replenishment, warehouse management integration.
* **Function**: Keeps track of product availability, prevents stockouts and overstock situations, and optimizes inventory control.

**7. Customer Relationship Management (CRM)**

* **Features**: Customer data storage, interaction tracking, personalized marketing, customer segmentation.
* **Function**: Manages customer interactions and data, supports targeted marketing campaigns, and enhances customer service and engagement.

**8. Content Management System (CMS)**

* **Features**: Easy content creation and editing, SEO-friendly, multimedia support, customizable templates.
* **Function**: Allows for the management and publication of website content, supports blogging and content marketing efforts, and ensures content is optimized for search engines.

**9. Analytics and Reporting**

* **Features**: Real-time data tracking, customizable reports, performance metrics (sales, traffic, conversion rates), integration with analytics platforms.
* **Function**: Provides insights into customer behavior, sales performance, and marketing effectiveness, supporting data-driven decision-making and strategy optimization.

**10. Security Features**

* **Features**: SSL encryption, secure payment processing, regular security audits, compliance with data protection laws.
* **Function**: Protects customer data, prevents unauthorized access, and ensures the overall security of the eCommerce platform.

**NON-FUNCTIONAL REQUIREMENTS**

Non-functional components of an eCommerce application refer to the attributes and criteria that describe the system's operation rather than specific behaviors or functions. These components are essential for ensuring the application meets performance, security, and reliability standards. Here are key non-functional components for an eCommerce application:

**1. Performance**

* **Scalability**: The ability of the application to handle increasing numbers of users and transactions without compromising performance.
* **Response Time**: The speed at which the application responds to user inputs, including page load times and checkout processing times.
* **Throughput**: The number of transactions the system can process within a given time frame.

**2. Reliability**

* **Availability**: The degree to which the system is operational and accessible when required for use (typically measured as a percentage of uptime).
* **Fault Tolerance**: The ability of the system to continue operating properly in the event of a failure of some of its components.
* **Recovery**: The system’s ability to recover quickly from failures, including data recovery and service restoration.

**3. Security**

* **Data Protection**: Ensuring customer and transaction data is protected through encryption, secure storage, and compliance with data protection regulations.
* **Access Control**: Mechanisms for authenticating and authorizing users to ensure that only legitimate users can access the system and perform permitted actions.
* **Vulnerability Management**: Regularly identifying, assessing, and mitigating security vulnerabilities in the application.

**4. Usability**

* **User-Friendly Interface**: Ensuring the application is easy to use and navigate, providing a pleasant user experience.
* **Accessibility**: Ensuring the application is usable by people with various disabilities, adhering to accessibility standards such as WCAG.
* **Consistency**: Providing a consistent user interface and experience across different pages and functions within the application.

**5. Maintainability**

* **Modularity**: Designing the application in a modular way so that individual components can be updated or replaced without affecting the entire system.
* **Documentation**: Providing comprehensive and up-to-date documentation for developers and administrators to facilitate maintenance and upgrades.
* **Code Quality**: Ensuring that the codebase is clean, well-documented, and follows best practices for software development.

**4.2 HARDWARE CONFIGURATION:**

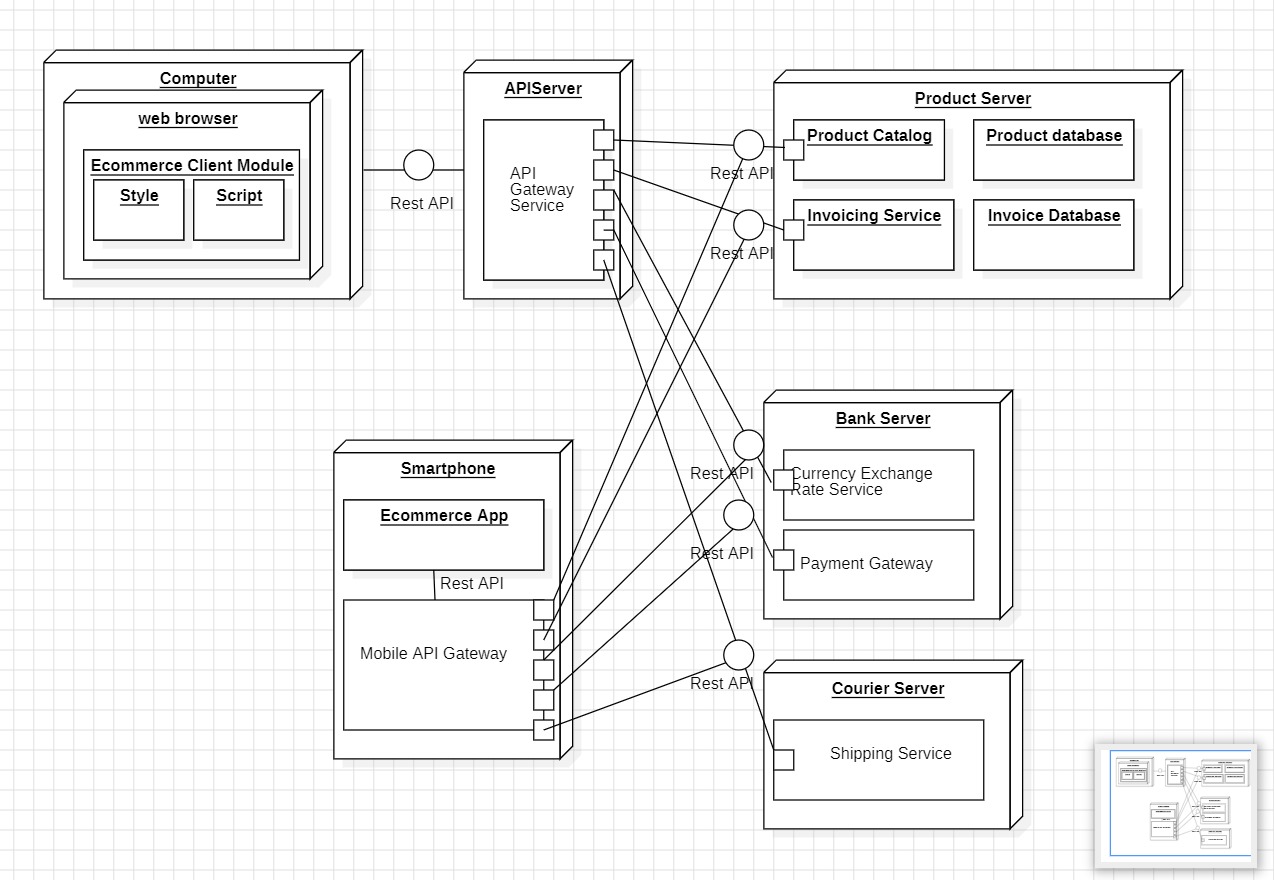
* **Server Requirements:**
  + **Processor:** Multi-core processor (e.g., Intel Xeon or AMD EPYC) to handle multiple requests and processes simultaneously.
  + **RAM:** Minimum of 16 GB, expandable to 32 GB or more for handling increased traffic and concurrent users.
  + **Storage:** SSD storage (at least 500 GB) for faster data retrieval and improved application performance, with options for additional storage as needed.
  + **Network:** Reliable and high-speed internet connection with redundancy to ensure continuous availability.
* **Client Requirements:**
  + **User Devices:** Compatibility with various devices, including desktops, laptops, tablets, and smartphones.
  + **Minimum Device Specifications:** Devices should have at least a dual-core processor, 4 GB RAM, and a stable internet connection.

**4.2 SOFTWARE CONFIGURATION:**

The software configuration will include various tools and technologies to support the development, deployment, and maintenance of the eCommerce app:

* **Backend Development:**
  + **Programming Language:** Dart for Flutter development, with additional support for Python (e.g., Flask or Django) for backend services.
  + **Database Management System:** MongoDB for storing product, user, and transaction data, providing flexibility and scalability.
* **Frontend Development:**
  + **Framework:** Flutter for creating a responsive and intuitive user interface across platforms (iOS and Android).
  + **UI/UX Design Tools:** Figma
* **Server and Hosting:**
  + **Web Server:** Nginx
  + **Cloud Hosting:** Digital Ocean
* **Payment Processing:**
  + **Payment Gateway:** RazorPay
* **Security:**
  + **SSL Certificate:** Implementation of SSL certificates for secure communication between users and the server(SSL Certificate as been obtained from Digital Ocean).

**4.4 ARCHITECTURE**



**5.METHODOLOGY**

The development of the eCommerce store utilizes the Incremental Model, emphasizing the construction of the application in small, manageable increments. This approach facilitates the early delivery of functional components, enabling continuous feedback and iterative improvements throughout the project lifecycle. The methodology consists of several key phases: planning and requirements gathering, design, development, testing, deployment, and post-launch activities.

**Incremental Approach Overview**

This project adheres to the Incremental Model, breaking the development process into multiple iterations or increments. Each increment delivers a specific set of features, allowing for user feedback and adjustments to be incorporated into subsequent iterations.

**1. Planning and Requirements Gathering**

The initial phase involved several activities:

* **Identifying Objectives:** The primary goals of the eCommerce store were established, including a target to increase sales by 30% within the first year, improve the customer experience through a streamlined checkout process, and expand market reach to include international customers.
* **Market Research:** Competitive analysis and target audience studies were conducted to understand market needs and identify gaps that the new store could fill.
* **Feature List:** Essential features for the platform were collaboratively outlined, focusing on user registration, product catalog, shopping cart, secure payment gateway integration (Razorpay), order tracking, and customer reviews.
* **Technology Stack:** An appropriate technology stack was selected, including Flutter for frontend development, Node.js with Express.js for backend development, and MongoDB for database management.
* **Budget and Timeline:** The project budget was estimated at 50,000, and a timeline for each increment was established to ensure the timely delivery of features.

**2. Design**

The design phase included the following steps:

* **Wireframes and Mockups:** Wireframes for critical pages (home, product detail, cart, and checkout) were created, followed by high-fidelity mockups illustrating the user interface. Stakeholder approval was obtained for the designs.
* **User Experience (UX):** Usability testing on wireframes was conducted to ensure an intuitive application, and feedback was gathered to inform design iterations in subsequent increments.
* **User Interface (UI):** An attractive and responsive UI was designed to reflect the brand identity, ensuring compatibility across various devices.

**3. Development**

During the development phase, several activities were executed:

* **Increment Planning:** The planning for app development in incremental approach is as follows:

**Increment 1 –** Develop the backend API using NodeJs and ExpressJs - ZEROTH REVIEW

**Increment 2 –** Develop the app with all features except recommendations – 1st REVIEW

**Increment 3 –** Develop the complete app including recommendations – FINAL REVIEW

* **Frontend Development:** UI components were coded using Flutter, ensuring responsiveness and adherence to design specifications.
* **Backend Development:** Server-side logic and APIs were developed using Node.js and Express.js, with a focus on user authentication, product management, and order processing.
* **Database Design:** The MongoDB database schema was established, ensuring efficient data retrieval and storage.
* **Integration:** Third-party services, such as Razorpay for payment processing and email services, were integrated within each increment.
* **Version Control:** Git was utilized for code management, allowing collaboration among team members and tracking changes.

**4. Testing**

The testing phase comprised:

* **Incremental Testing:** Each increment was tested as it was developed, focusing on unit testing individual components.
* **Integration Testing:** All components developed in an increment were tested together to verify seamless interactions.
* **User Acceptance Testing (UAT):** Testing with actual users was conducted at the end of each increment to gather feedback and make necessary adjustments based on real-world interactions.
* **Performance Testing:** The application’s performance was assessed under various conditions at the end of each increment to ensure it met user expectations.

**5. Deployment**

In the deployment phase, the following steps were taken:

* **Choose Hosting Provider:** Digital Ocean was selected for hosting due to its scalability and reliability.
* **Continuous Integration/Continuous Deployment (CI/CD):** CI/CD pipelines were established using tools like GitHub Actions for automated testing and deployment, ensuring rapid iteration and delivery.
* **Domain and SSL:** The domain (chavarishik.online) was registered with GoDaddy, and SSL certificates were implemented from Digital Ocean for secure user data and transactions.
* **Launch:** The application was deployed to the production environment at the end of each increment, with all configurations validated.

**6. Post-Launch Activities**

Following the launch, several activities were essential for ongoing success:

* **Continuous Maintenance:** Regular monitoring of the application was established for performance issues, with a dedicated team ready to address bugs and vulnerabilities.
* **User Feedback Incorporation:** User feedback was actively collected through surveys and reviews, guiding future enhancements and feature additions in subsequent increments.
* **Regular Updates:** Scheduled updates were implemented quarterly to introduce new features and improve existing functionality.

**6.SYSTEM DESIGN**

**6.1 INPUT AND OUTPUT**

CLIENT-SIDE APPLICATION (FLUTTER):

INPUTS:

* **Registration and Login**: User enters email, password, and other registration details.
* **Product Search**: User inputs search keywords.
* **Filter and Sort**: User selects filter options (e.g., category, price range) and sorting preferences.
* **Add to Cart**: User selects products and quantities to add to the cart.
* **Checkout**: User inputs shipping details, selects payment method, and confirms the order.
* **Profile Management**: User updates profile information and addresses.
* **Product Reviews**: User submits product ratings and reviews.

OUTPUTS:

* **Product Listings**: Display a list of products based on search and filter criteria.
* **Product Details**: Show detailed information about a selected product.
* **Shopping Cart**: Display items added to the cart and update totals.
* **Order Confirmation**: Show order summary and confirmation message after checkout.
* **Profile Page**: Display updated user profile information.
* **Notifications**: Show notifications for order updates, promotions, etc.

SERVER-SIDE API(NODE.JS):

INPUTS:

* **User Authentication**: Requests for user registration, login, and password recovery.
* **Product Management**: Requests to fetch, create, update, and delete product information.
* **Order Processing**: Requests to place orders, update order status, and process payments.
* **User Management**: Requests to update user profiles and manage addresses.
* **Review Submission**: Requests to submit and fetch product reviews.

OUTPUTS:

* **Authentication Responses**: Token generation for successful login, error messages for failures.
* **Product Data**: JSON responses containing product details, lists, and search results.
* **Order Data**: Confirmation of order placement, order status updates.
* **User Data**: User profile information, update confirmations.
* **Error Handling**: Error messages for invalid requests, authentication failures, etc.

ADMIN PANEL (FLUTTER):

INPUTS:

* **User Management**: Admin inputs to create, update, or delete user accounts and assign roles.
* **Product Management**: Admin inputs to add new products, update product details, and manage inventory.
* **Order Management**: Admin actions to view, update, and process orders.
* **Promotions and Discounts**: Admin inputs to create and manage discount codes and promotional offers.
* **Content Management**: Admin inputs to update informational pages, banners, and other content.

OUTPUTS:

* **User Account Management**: Display list of users, details of selected user, and confirmation of actions.
* **Product Management**: Show product lists, detailed product information, and update confirmations.
* **Order Management**: Display orders, status updates, and processing details.
* **Analytics and Reports**: Show sales reports, inventory status, and other key metrics.
* **Content Management**: Display current content and confirmation of updates.

**6.2 UML DIAGRAMS**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artefacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS**

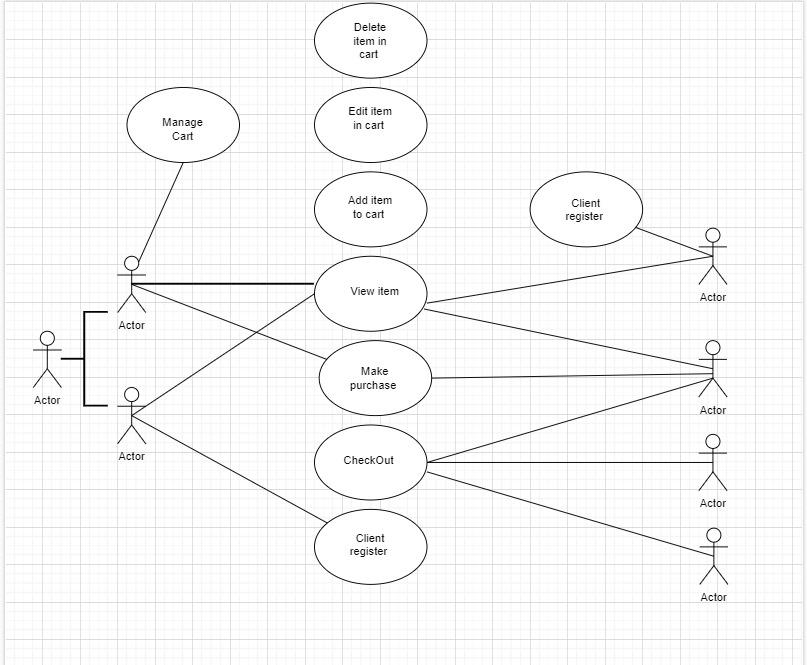
* The Primary goals in the design of the UML are as follows:
* Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
* Provide extendibility and specialization mechanisms to extend the core concepts.
* Be independent of particular programming languages and development process.
* Provide a formal basis for understanding the modelling language.
* Encourage the growth of OO tools market.
* Support higher level development concepts such as collaborations, frameworks, patterns and components.
* Integrate best practices.

**USE CASE DIAGRAM**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram created from a use-case analysis. Its primary purpose is to provide a graphical overview of the functionality provided by a system, illustrating the interactions between actors (users or other systems) and their goals (use cases).

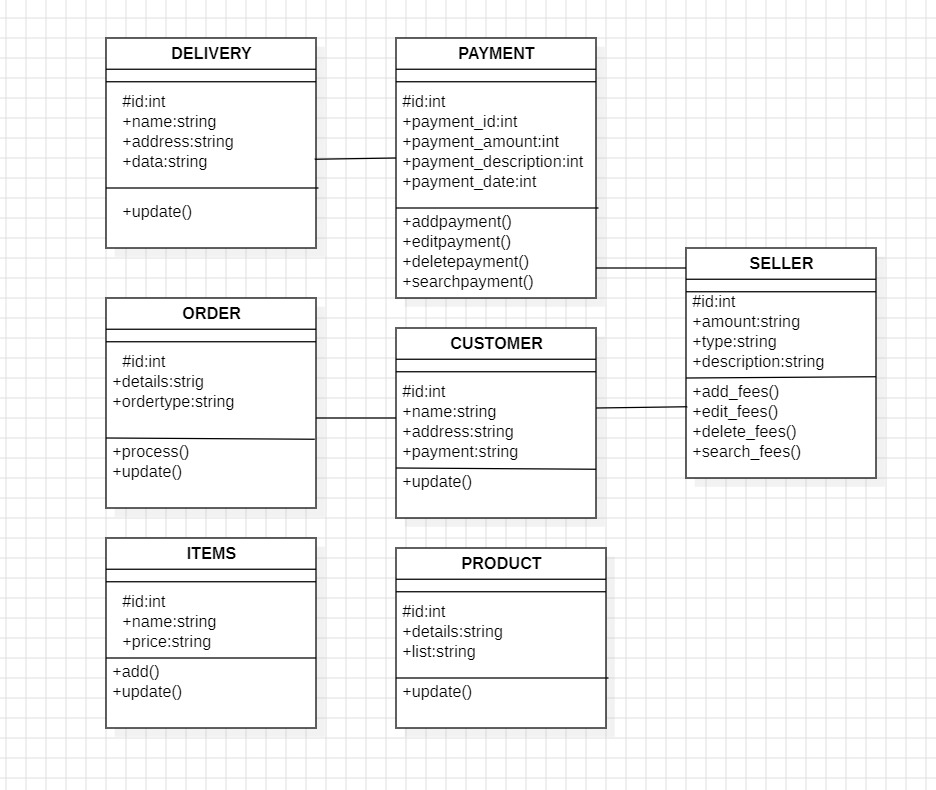
The diagram shows what system functions are performed for which actor, clearly depicting the roles of the actors within the system. Actors represent users or external systems that interact with the system, and each has specific goals depicted as use cases. Use cases represent the distinct functionalities or services the system provides.

Additionally, use case diagrams highlight dependencies between use cases, such as inclusion, extension, and generalization relationships. These relationships help understand how different use cases are connected and how they contribute to the overall functionality of the system. By visually representing these elements, use case diagrams aid in requirements gathering, system design, and communication among stakeholders, ensuring a clear understanding of the system's behavior and interactions.



**CLASS DIAGRAM**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



**ER DIAGRAM**

The Entity-Relationship Diagram (ERD) for the e-commerce application illustrates the relationships and attributes of key entities within the system. Here's a detailed description:

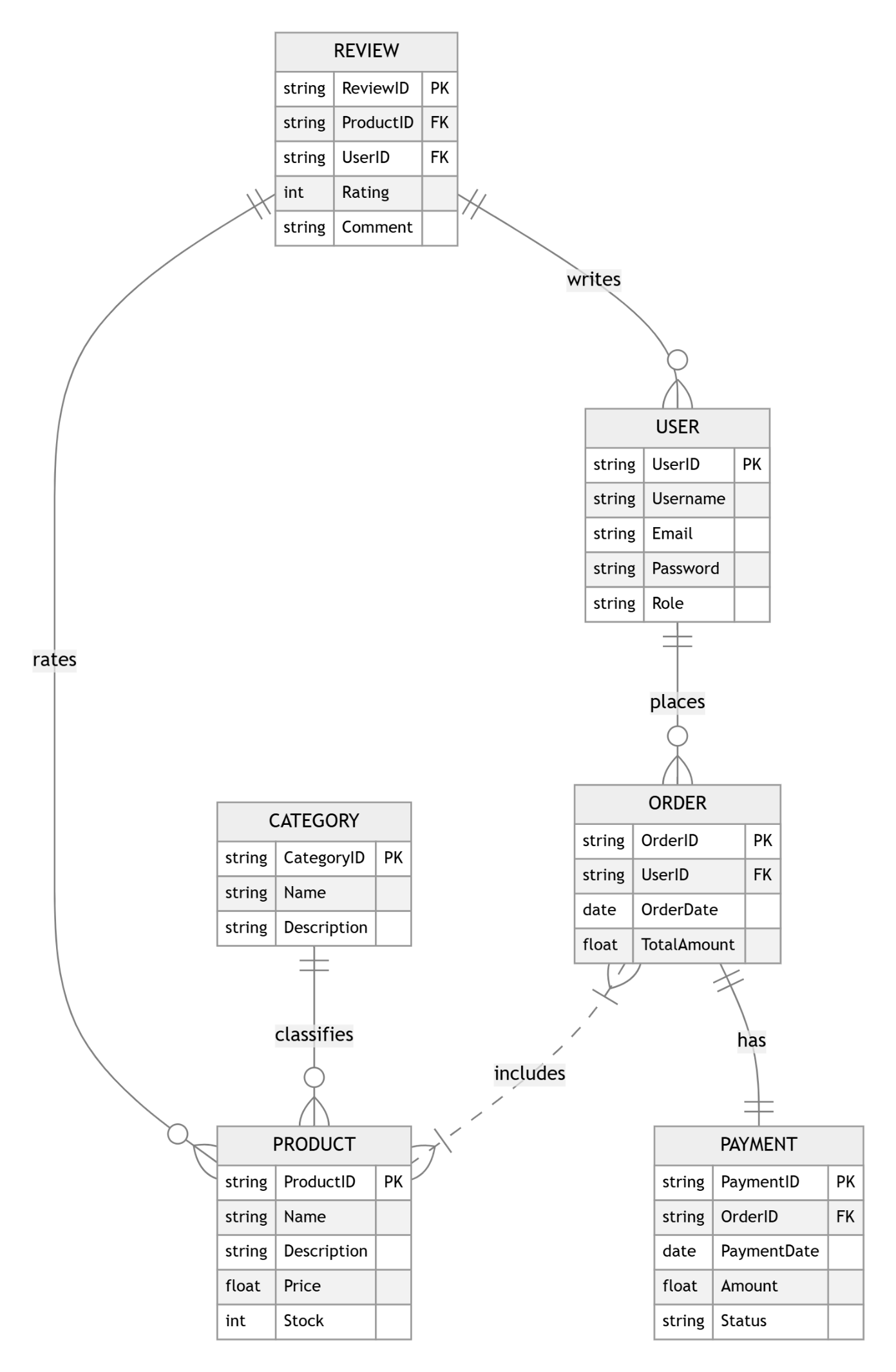
1. **Entities:**
   * **User:** Represents individuals who interact with the system, including customers and administrators. Attributes include UserID (primary key), Username, Email, Password, Role (e.g., customer, admin).
   * **Product:** Represents items available for purchase in the e-commerce platform. Attributes include ProductID (primary key), Name, Description, Price, Stock.
   * **Order:** Represents a purchase request made by a customer. Attributes include OrderID (primary key), UserID (foreign key to User), OrderDate, TotalAmount.
   * **Category:** Represents the classification of products into different categories. Attributes include CategoryID (primary key), Name, Description.
   * **Payment:** Represents the financial transaction details associated with an order. Attributes include PaymentID (primary key), OrderID (foreign key to Order), PaymentDate, Amount, Status.
   * **Review:** Represents customer reviews and ratings for products. Attributes include ReviewID (primary key), ProductID (foreign key to Product), UserID (foreign key to User), Rating, Comment.
2. **Relationships:**
   * **User-Order:** One-to-Many relationship where a User can place multiple Orders, but each Order belongs to one User.
   * **Product-Category:** Many-to-One relationship where multiple Products belong to one Category, and each Product belongs to only one Category.
   * **Order-Product:** Many-to-Many relationship indicating that an Order can include multiple Products, and a Product can be included in multiple Orders.
   * **Order-Payment:** One-to-One relationship where each Order has one Payment associated with it.
   * **Review-Product:** Many-to-One relationship where multiple Reviews are associated with one Product, and each Review is for only one Product.
   * **Review-User:** Many-to-One relationship where multiple Reviews are written by one User, and each Review is written by only one User.
3. **Cardinality and Constraints:**
   * **User-Order:** One User can place Many Orders (1

). OrderID is mandatory for User, but User can exist without placing an Order.

* + **Product-Category:** Many Products can belong to One Category (N:1). CategoryID is mandatory for Product.
  + **Order-Product:** Many Orders can include Many Products (M

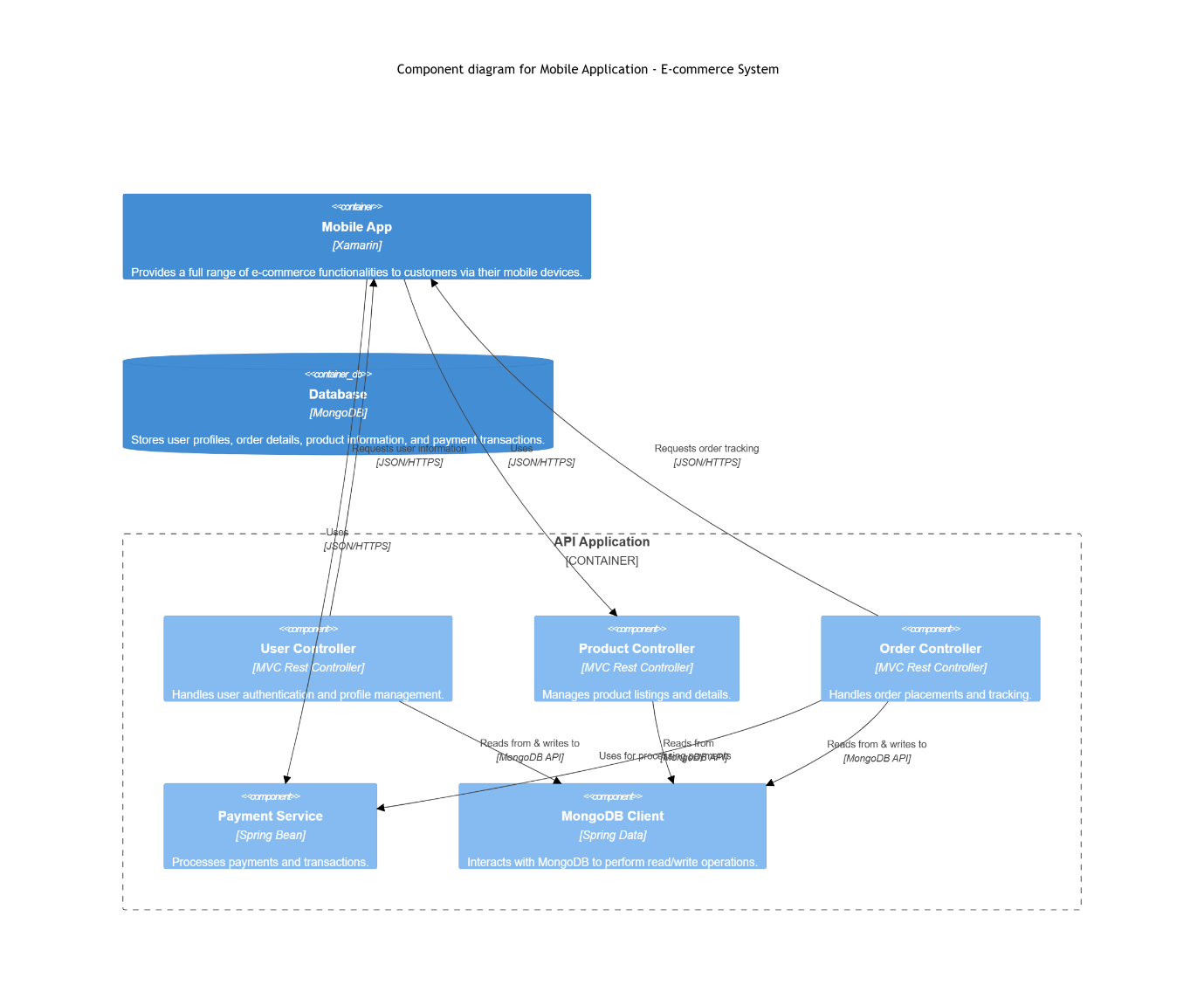
). Both OrderID and ProductID are mandatory in the junction table.

* + **Order-Payment:** One Order has One Payment (1:1). Both OrderID and PaymentID are mandatory and unique.
  + **Review-Product:** Many Reviews for One Product (N:1). ProductID is mandatory for Review.
  + **Review-User:** Many Reviews by One User (N:1). UserID is mandatory for Review.



**COMPONENT DIAGRAM:**

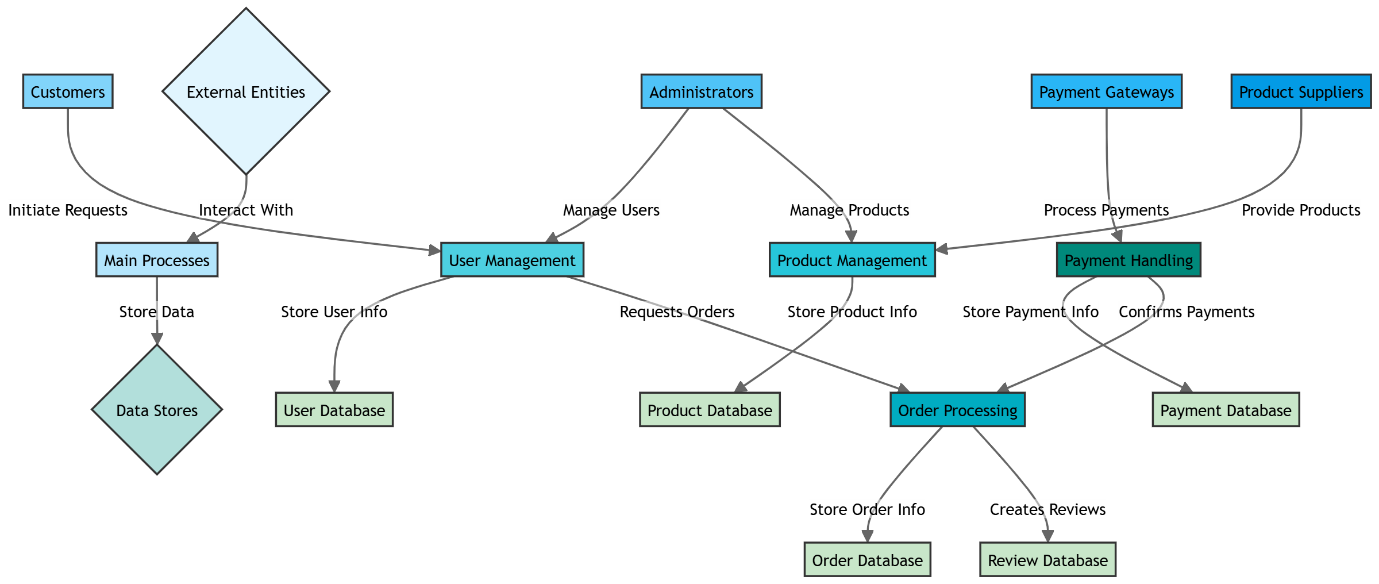
A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.



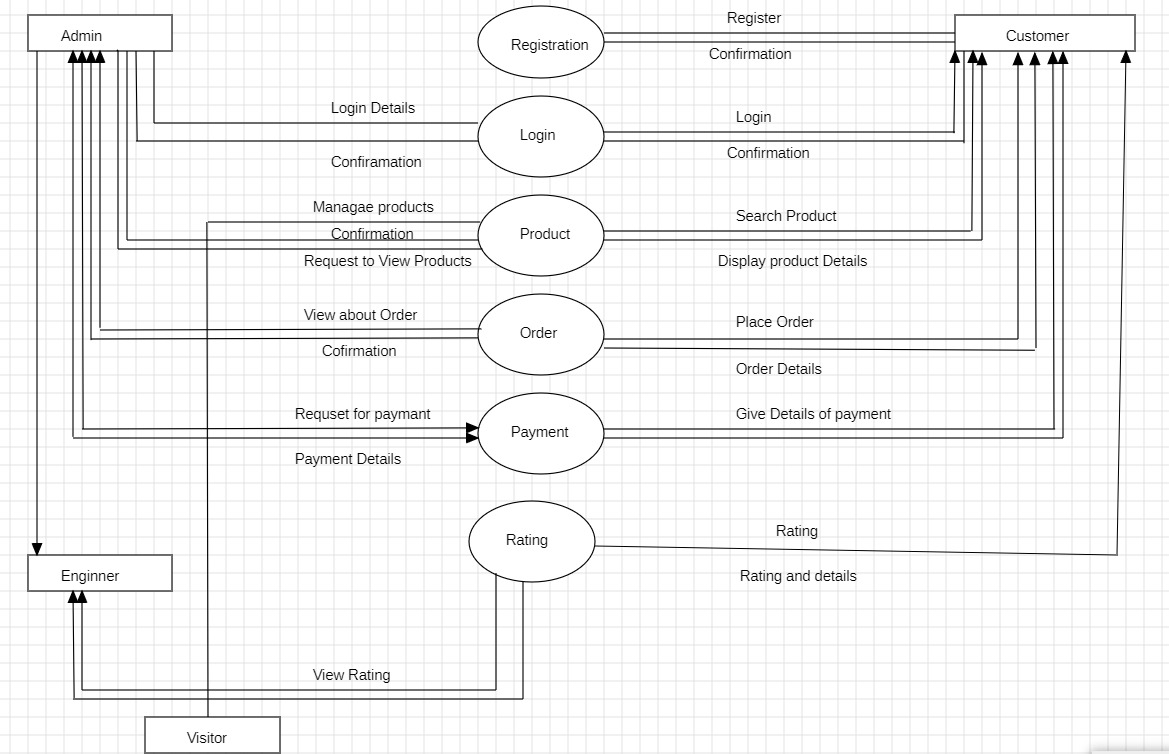
**DFD DIAGRAM**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

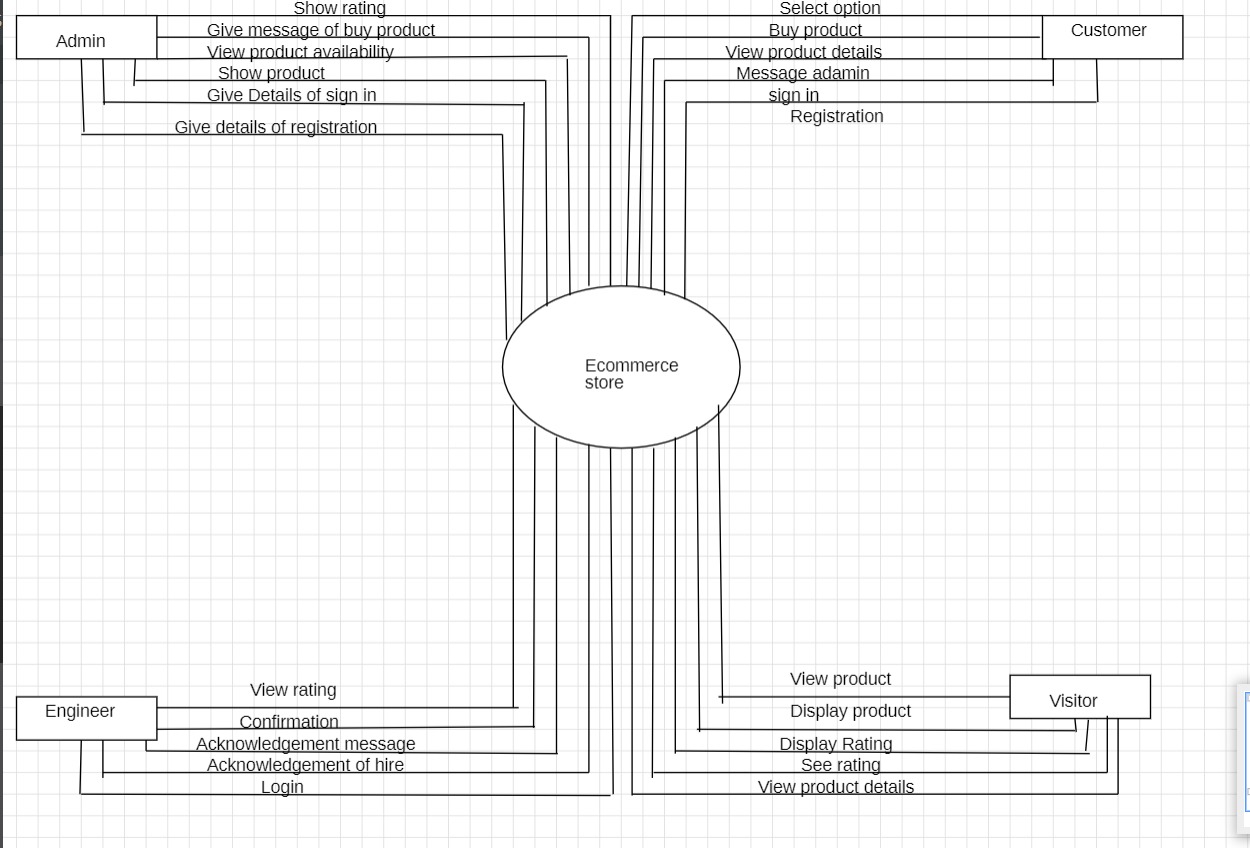
1. **Level 0 DFD:**
   * **Main Processes:** Includes high-level processes such as User Management, Product Management, Order Processing, and Payment Handling.
   * **External Entities:** Customers, Administrators, Payment Gateways, and Product Suppliers.
   * **Data Stores:** Databases (User, Product, Order, Payment, Review) storing persistent data.
   * **Data Flows:** Communication paths between entities and processes, indicating the flow of information.



1. **Level 1 DFDs:**
   * **User Management:** Processes related to user authentication, registration, and profile management.
   * **Product Management:** Processes related to adding new products, updating product details, and managing inventory.
   * **Order Processing:** Processes involved in order placement, order fulfillment, and order status updates.
   * **Payment Handling:** Processes related to processing payments, verifying transactions, and handling refunds.
   * **Review Management:** Processes related to submitting reviews, rating products, and displaying reviews.



1. **Level 2 DFDs:**
   * **Detailed Processes:** Further breakdown of Level 1 processes into detailed sub-processes.
   * **Input/Output Data:** Specification of data inputs and outputs for each process.
   * **Data Stores:** Refinement of data storage locations and data access patterns.
   * **Data Flows:** Elaboration of data flows, including triggers, transformations, and data validation steps.



**7.IMPLEMANTATION**

**MODULES**

**7.1 User Authentication Module**

The User Authentication Module is the foundation of user security and account management within the eCommerce store. It enables users to create accounts through a straightforward registration process, where they provide essential details like email and password. Once registered, users can securely log in to their accounts. Additionally, the module includes a password recovery feature, allowing users to reset forgotten passwords through email verification. This emphasis on security and user accessibility is crucial in building trust and encouraging user engagement with the platform.

**7.2 Product Catalog Module**

The Product Catalog Module serves as the storefront of the eCommerce platform, showcasing products with comprehensive details that assist users in making informed purchasing decisions. It includes product listings that feature essential information such as product names, prices, high-quality images, and descriptions. To enhance user experience, this module incorporates category filtering, allowing users to narrow down their choices based on specific criteria. Furthermore, a robust search functionality enables users to quickly find particular products using keywords, making navigation efficient and user-friendly.

**7.3 Shopping Cart Module**

The Shopping Cart Module plays a pivotal role in the purchasing process by enabling users to manage selected items they intend to buy. Users can effortlessly add products to their cart from various sections of the site, view a comprehensive summary of their selected items, and modify quantities or remove items as desired. This module ensures a seamless transition from product selection to checkout, helping users keep track of their intended purchases while fostering a sense of control over their shopping experience.

**7.4 Checkout Module**

The Checkout Module is integral to the completion of the purchasing journey, guiding users through the final steps of placing an order. It presents an order summary that details items in the cart, shipping options, and total costs. The module integrates with the Razorpay payment gateway to facilitate secure and efficient payment processing, ensuring that users can complete transactions with confidence. Upon successful payment, users receive confirmation emails, providing reassurance and documentation of their purchases.

**7.5 User Profile Module**

The User Profile Module empowers users to manage their personal information and preferences conveniently. Within this module, users can view and edit their profile details, including contact information and saved shipping addresses, which streamlines future purchases. Additionally, the module offers access to order history, enabling users to track past orders and revisit their favorite items. By allowing users to personalize their accounts, this module enhances user engagement and satisfaction.

**7.6 Review and Rating Module**

The Review and Rating Module fosters a community-driven atmosphere by allowing users to provide feedback on products they have purchased. Users can submit detailed reviews and assign ratings based on their experiences, contributing to a wealth of user-generated content that assists others in making purchasing decisions. The module also includes an admin moderation feature, enabling the management team to oversee submissions and ensure that reviews maintain a standard of quality and relevance, which ultimately enhances the credibility of the product offerings.

**7.7 Admin Dashboard Module**

The Admin Dashboard Module serves as a control center for managing the eCommerce store. It provides administrators with tools to oversee product management, allowing them to add, update, or delete products from the catalog easily. This module also includes order management capabilities, enabling admins to track order statuses, process shipments, and handle customer inquiries efficiently. Furthermore, the User Management feature allows administrators to view and manage user accounts, ensuring that the platform runs smoothly and customer support is readily available.

**7.8 Recommendation Engine Module**

The Recommendation Engine Module enhances the shopping experience by providing personalized product suggestions based on user behavior and preferences. This module analyzes data from user interactions, such as browsing history and previous purchases, to deliver tailored recommendations that align with individual interests. By presenting users with relevant products, this module not only improves user engagement but also increases the likelihood of additional sales, thereby contributing to overall revenue growth.

**8.SYSTEM STUDY AND TESTING**

**8.1 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

**8.1.1 Technical Feasibility**

The technical feasibility of the eCommerce store was assessed to determine the adequacy of the chosen technology stack and infrastructure. The application leverages Flutter for frontend development, which allows for a responsive and intuitive user interface across both iOS and Android platforms. For backend services, Node.js and Express.js were selected due to their scalability and efficiency in handling asynchronous requests. Additionally, MongoDB was chosen as the database management system for its flexibility in storing unstructured data and ability to scale horizontally, accommodating increasing data loads as the business grows. The technical team evaluated existing hardware and software requirements, confirming that the necessary resources and expertise were available within the organization to support the implementation and ongoing maintenance of the application.

**8.1.2 Economic Feasibility**

The economic feasibility analysis focused on evaluating the project’s financial aspects to ensure it is a worthwhile investment. A comprehensive cost-benefit analysis was conducted, estimating the total project budget at $50,000, which includes expenses for development, hosting, marketing, and ongoing maintenance. The anticipated revenue projections, based on market research and analysis of competitors, suggest a potential sales growth of 30% within the first year of operation, driven by improved customer experience and expanded market reach. Additionally, the analysis considered cost savings associated with lower operational expenses compared to maintaining physical storefronts, reinforcing the economic viability of launching the eCommerce store. This financial assessment indicates that the potential returns justify the investment, making the project economically feasible.

**8.1.3 Operational Feasibility**

Operational feasibility examined the organization’s capacity to support the eCommerce platform post-launch. This evaluation confirmed that the team possesses the necessary skills and resources to manage the application effectively. Key personnel, including developers, designers, and marketing specialists, have been identified and will be trained to handle their respective responsibilities. Furthermore, existing operational processes were reviewed to ensure they align with the requirements of an online business, including customer service protocols, order fulfillment, and inventory management. The analysis also highlighted the importance of implementing robust support systems, such as training for customer service staff and establishing clear communication channels for addressing user inquiries and issues. Overall, the operational feasibility study confirmed that the organization is well-equipped to maintain and grow the eCommerce platform, ensuring long-term sustainability and success.

8.2 TESTING

**8.2.1 Functional Testing**

Functional testing was conducted to ensure that each feature of the eCommerce mobile app operates as intended. This phase involved testing critical functionalities such as user registration, product search, shopping cart management, checkout processes, and payment gateway integration. Test cases were created to validate user inputs, ensuring that the system responds appropriately to both valid and invalid data. Each module underwent rigorous testing to confirm that all features worked correctly across various mobile devices and operating systems. The testing team documented any discrepancies found during this process, allowing developers to address issues before the final launch. By thoroughly testing the functional aspects of the application, we ensured that users would experience a seamless and intuitive shopping journey on their mobile devices.

**8.2.2 Performance Testing**

Performance testing was executed to evaluate the mobile app’s responsiveness and stability under different load conditions. This involved simulating various user traffic scenarios to assess how the application handles peak loads, ensuring it could accommodate increased usage during promotional events or holiday seasons. Key metrics measured included app launch times, page load speeds, and transaction processing times. Load testing tools were utilized to simulate concurrent users, revealing potential bottlenecks in the system. Stress testing was also performed to determine the application's limits and ensure it remains operational under extreme conditions. The performance testing phase provided valuable insights into system capabilities, allowing for optimizations to enhance speed and reliability, ultimately improving user satisfaction.

**8.2.3 Notifications Testing**

Notifications testing was performed to ensure that users receive timely and accurate alerts related to their activities within the mobile app. This included testing various types of notifications, such as order confirmations, shipping updates, promotional offers, and account-related alerts. The testing team verified that notifications were triggered correctly based on user actions, ensuring that they were delivered through the appropriate channels, including push notifications and in-app alerts. Each notification was evaluated for content accuracy, formatting, and delivery timing to ensure a positive user experience. Additionally, the performance of the notification system was tested under high-load scenarios to confirm that it remained operational during peak usage. Thorough testing of notifications is crucial for maintaining effective communication with users and enhancing overall engagement with the app.

**8.2.4 Security Testing**

Security testing was conducted to identify vulnerabilities within the mobile app and ensure the protection of sensitive user data. This included assessments of authentication mechanisms, authorization controls, and data encryption methods.

**8.2.5 Usability Testing**

Usability testing was performed to evaluate the user experience of the eCommerce mobile app. This involved gathering feedback from real users as they interacted with the application, focusing on ease of navigation, intuitiveness of the interface, and overall satisfaction. Participants were observed while completing key tasks, such as browsing products, adding items to the cart, and completing the checkout process. Feedback collected during this phase provided valuable insights into areas for improvement, allowing the design team to refine the user interface and enhance overall usability. The goal of usability testing was to ensure that the app meets user expectations and provides a positive shopping experience.

**8.2.6 Compatibility Testing**

Compatibility testing was conducted to ensure that the eCommerce mobile app functions correctly across various devices, operating systems, and screen sizes. This phase involved testing the application on different mobile platforms, including Android and iOS, as well as on various smartphone and tablet models. The testing team evaluated the app's performance on popular mobile devices to ensure consistent functionality and appearance. This comprehensive testing approach aimed to identify and resolve any discrepancies that may affect user experience, ensuring that the mobile app is accessible to a broad audience regardless of their device choice

**9.CONCLUSION**

The development of the eCommerce mobile app has successfully created a platform that addresses the growing demand for convenient and accessible online shopping. By employing an incremental development approach, the project has ensured that key features are delivered in manageable phases, allowing for continuous feedback and improvement. The thorough testing processes—encompassing functional, performance, security, usability, and compatibility testing—have validated the app's functionality and user experience, ensuring a robust and user-friendly product. Overall, this eCommerce app not only meets the current market needs but also lays the groundwork for future enhancements and expansion in the dynamic online retail landscape.

**10.FUTURE SCOPE**

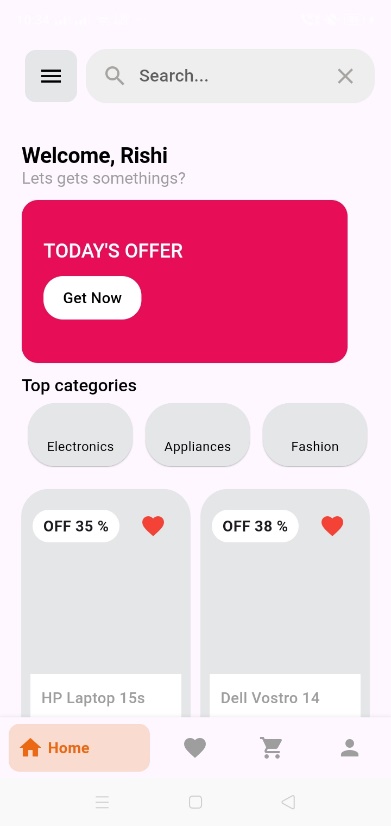
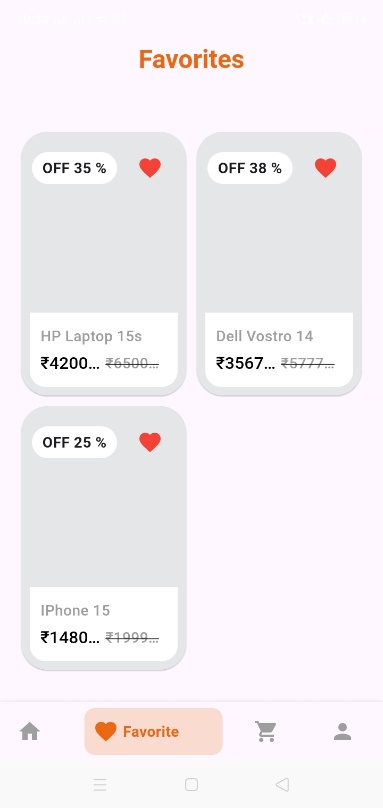
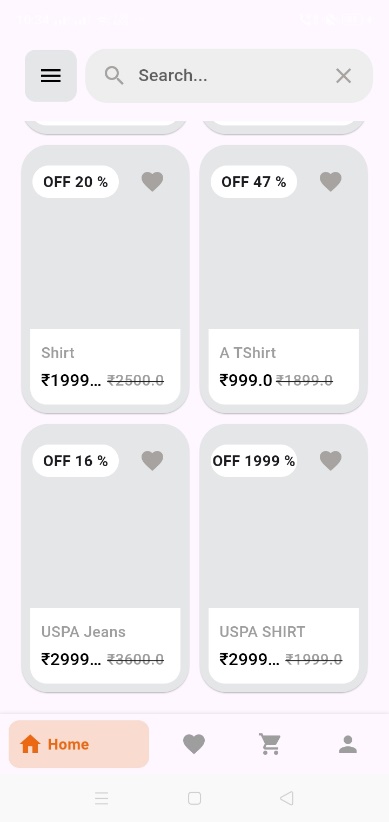
The future scope of the eCommerce mobile app includes several potential enhancements and features aimed at improving user experience and expanding market reach. Key areas for development include the integration of advanced AI-driven recommendation systems that provide personalized shopping experiences based on user behavior and preferences. Additionally, the implementation of augmented reality (AR) features could allow users to visualize products in their environment before making a purchase, enhancing decision-making. Expanding the app to support multi-language and multi-currency options will also cater to a broader international audience. Furthermore, exploring partnerships with social media platforms for social commerce features could leverage the growing trend of shopping through social channels. Continuous updates and improvements based on user feedback will be essential to keep the app competitive and relevant in the fast-evolving eCommerce landscape.

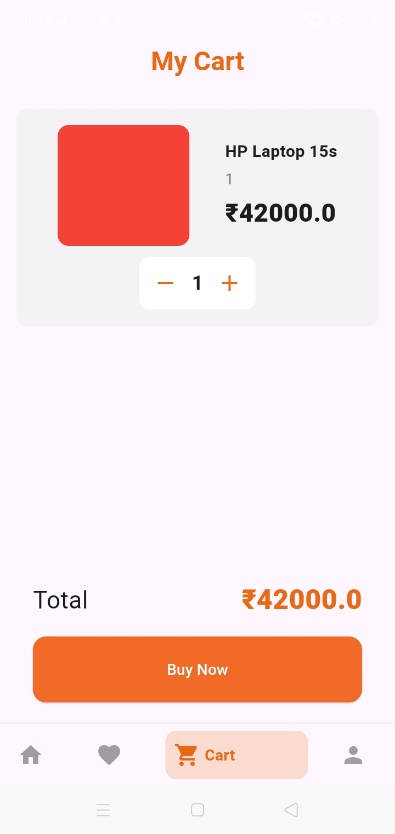
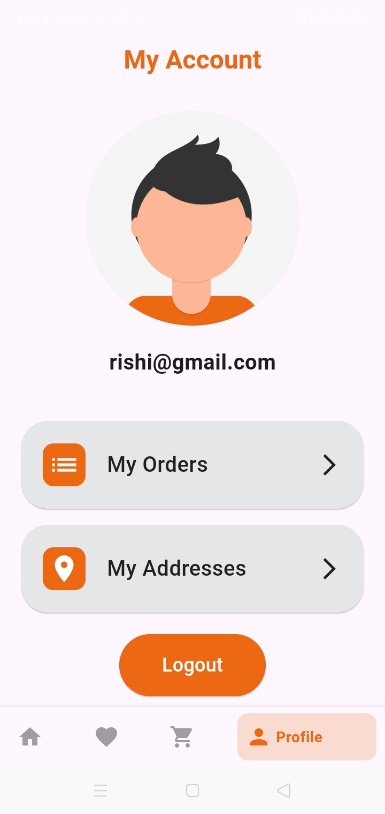
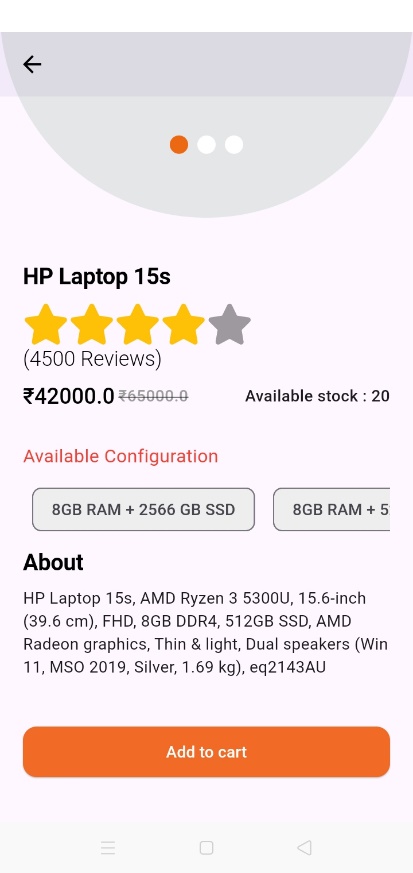
**11.REFERENCES**

1. **B2B vs. B2C eCommerce Models**:
   * Latham, D. (2022). “Understanding the Differences Between B2B and B2C eCommerce Models.” *Journal of Business Research*, 128, 140-150.
2. **Mobile eCommerce Trends**:
   * Smith, J. (2023). “The Rise of Mobile Commerce: Trends and Strategies for Success.” *eCommerce Times*. Retrieved from eCommerce Times.
3. **User Experience in Mobile Apps**:
   * Nielsen Norman Group. (2022). “Mobile User Experience: 10 Usability Heuristics for Mobile Apps.” Retrieved from Nielsen Norman Group.
4. **Security in eCommerce**:
   * Gupta, R. (2023). “Security Challenges in eCommerce: A Comprehensive Review.” *International Journal of Information Security*, 22(1), 1-16.
5. **Agile Development Methodologies**:
   * Beck, K., et al. (2021). “Manifesto for Agile Software Development.” Retrieved from Agile Alliance.
6. **Performance Testing Techniques**:
   * Gupta, S., & Kumar, R. (2021). “Performance Testing of Web Applications: Techniques and Tools.” *International Journal of Software Engineering*, 12(2), 45-60.
7. **Future of eCommerce**:
   * KPMG. (2022). “The Future of Retail: How Technology is Transforming the Shopping Experience.” Retrieved from KPMG.

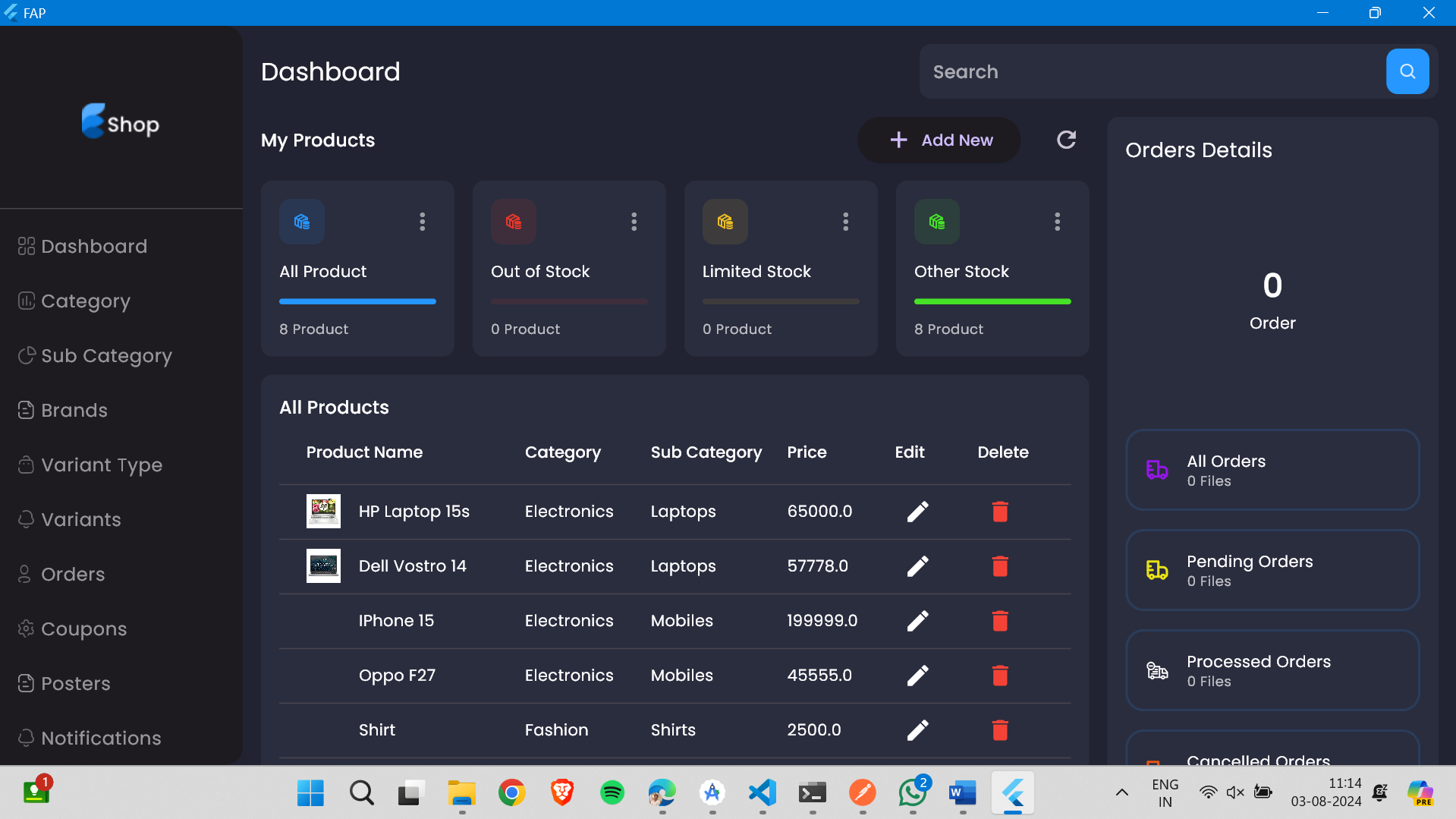
**Appendix -1**

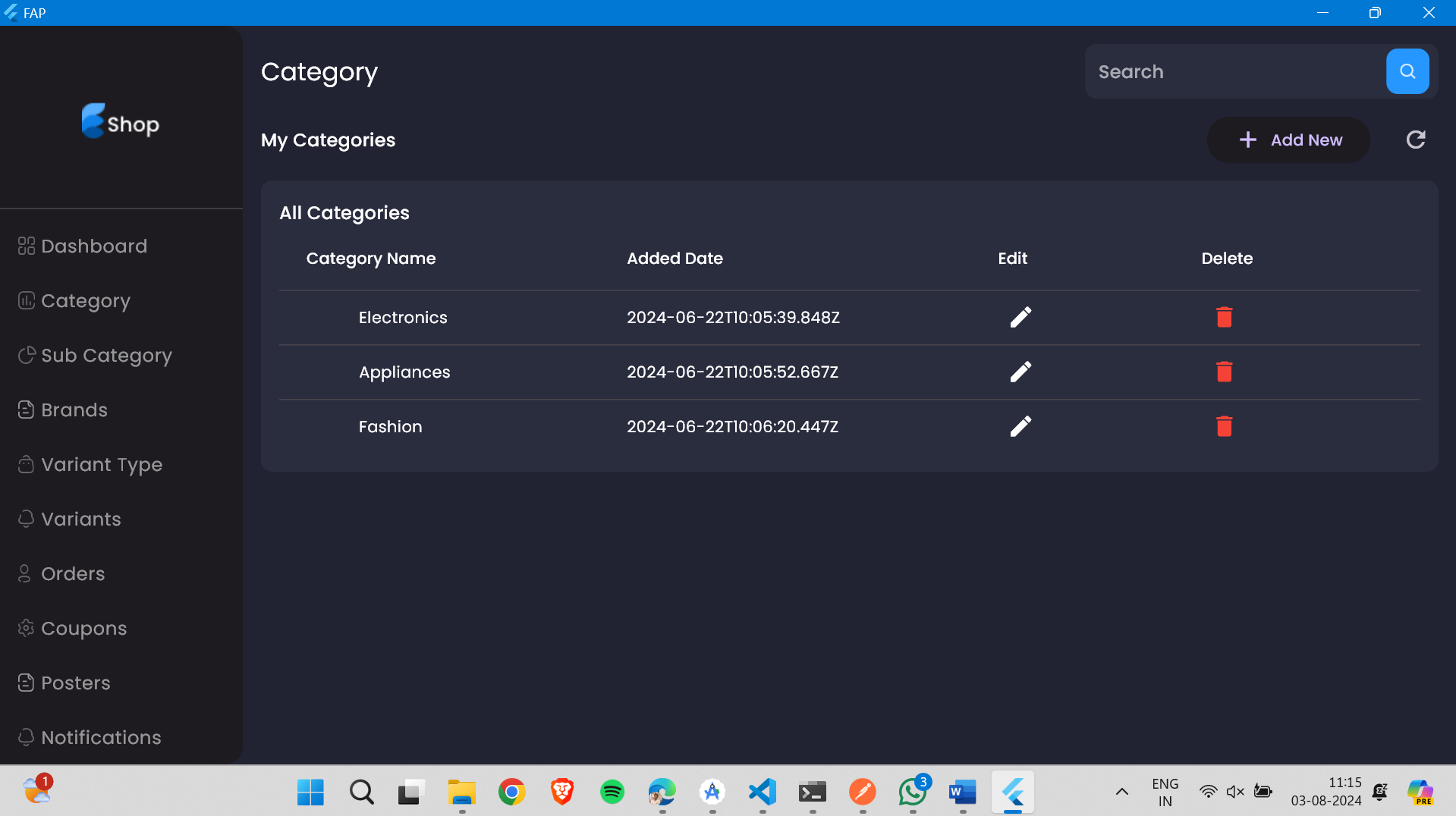
**Client App UI Visuals:**

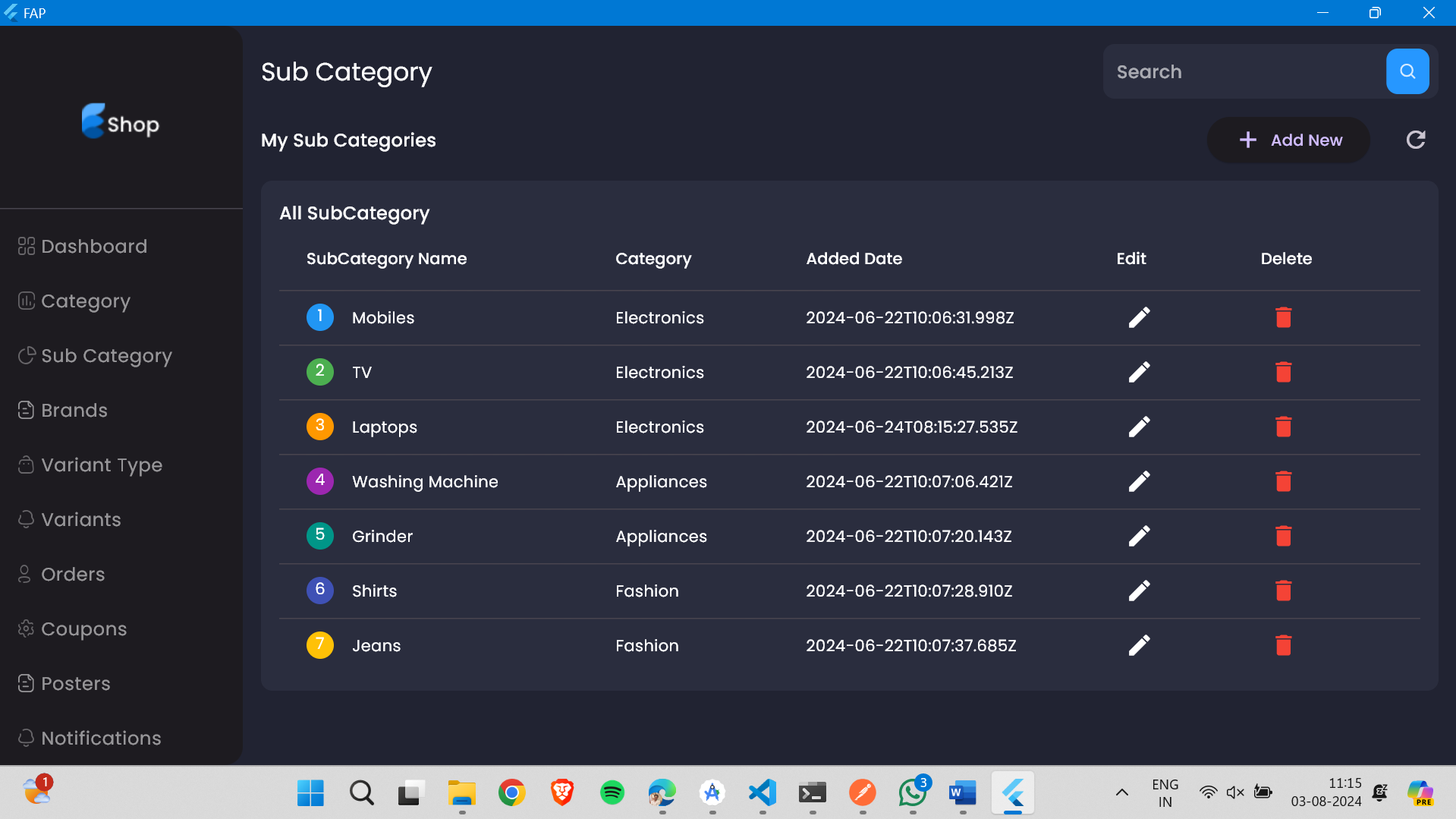


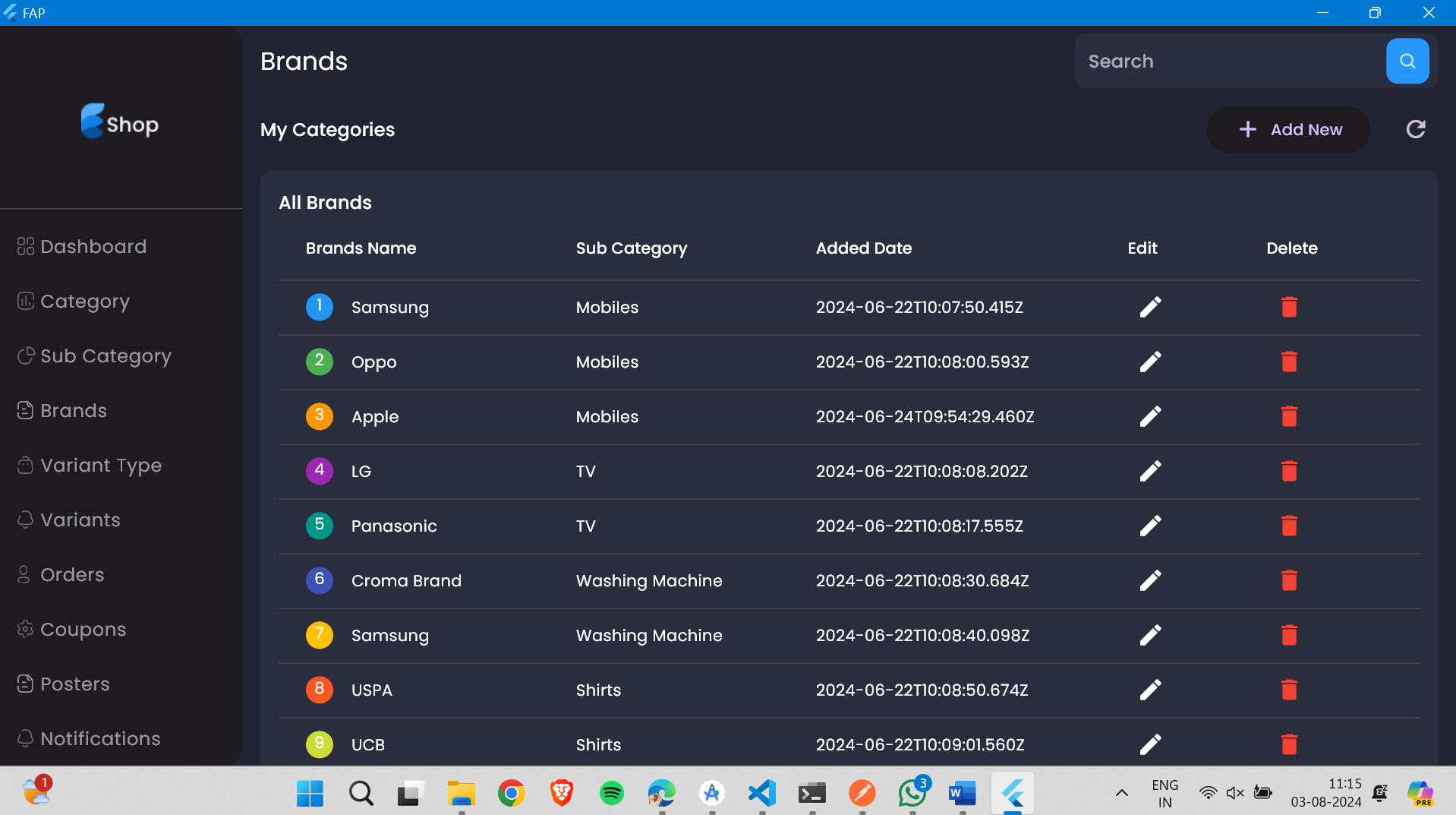


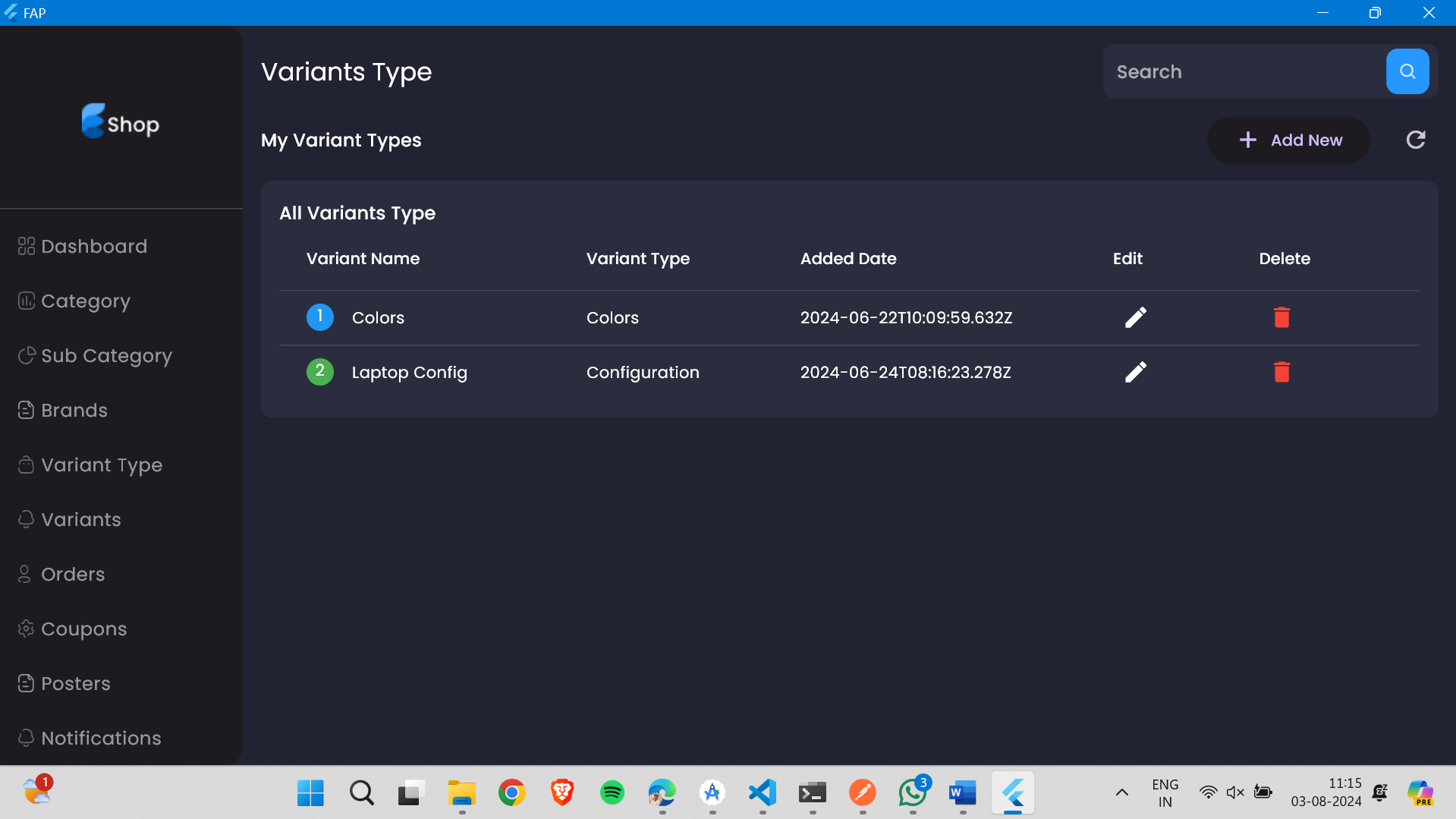
**Admin App Visuals:**

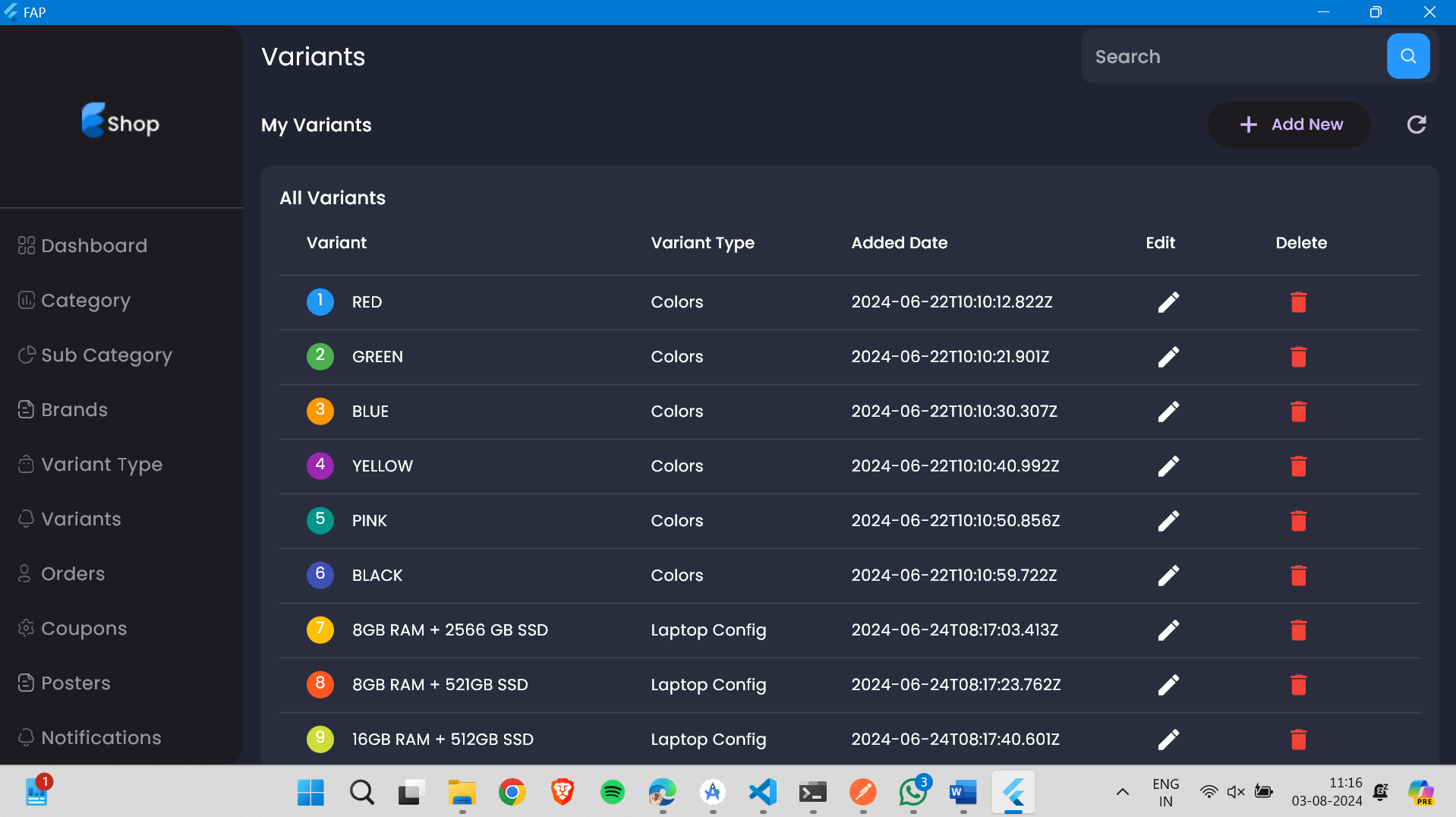
****

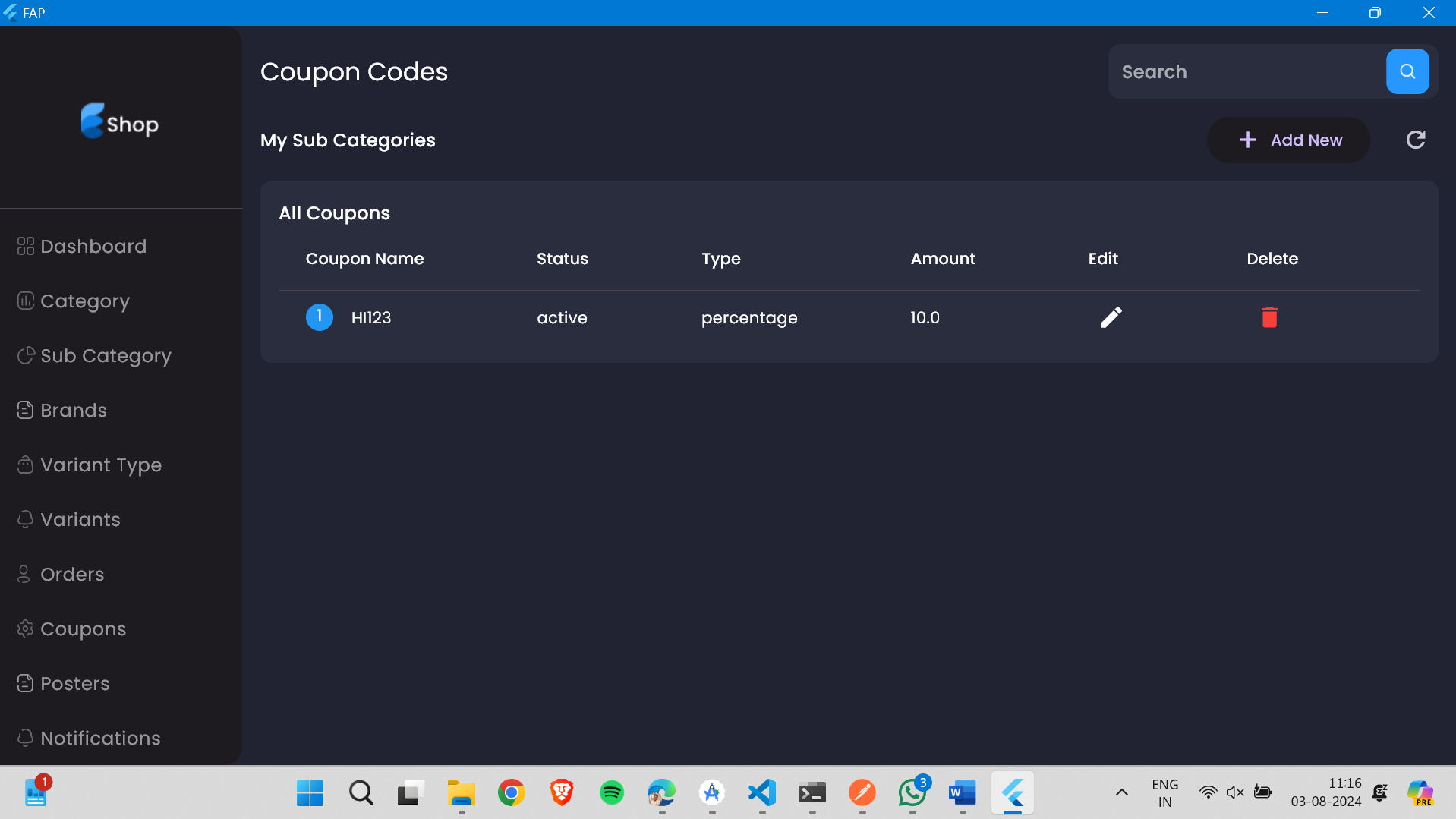
****

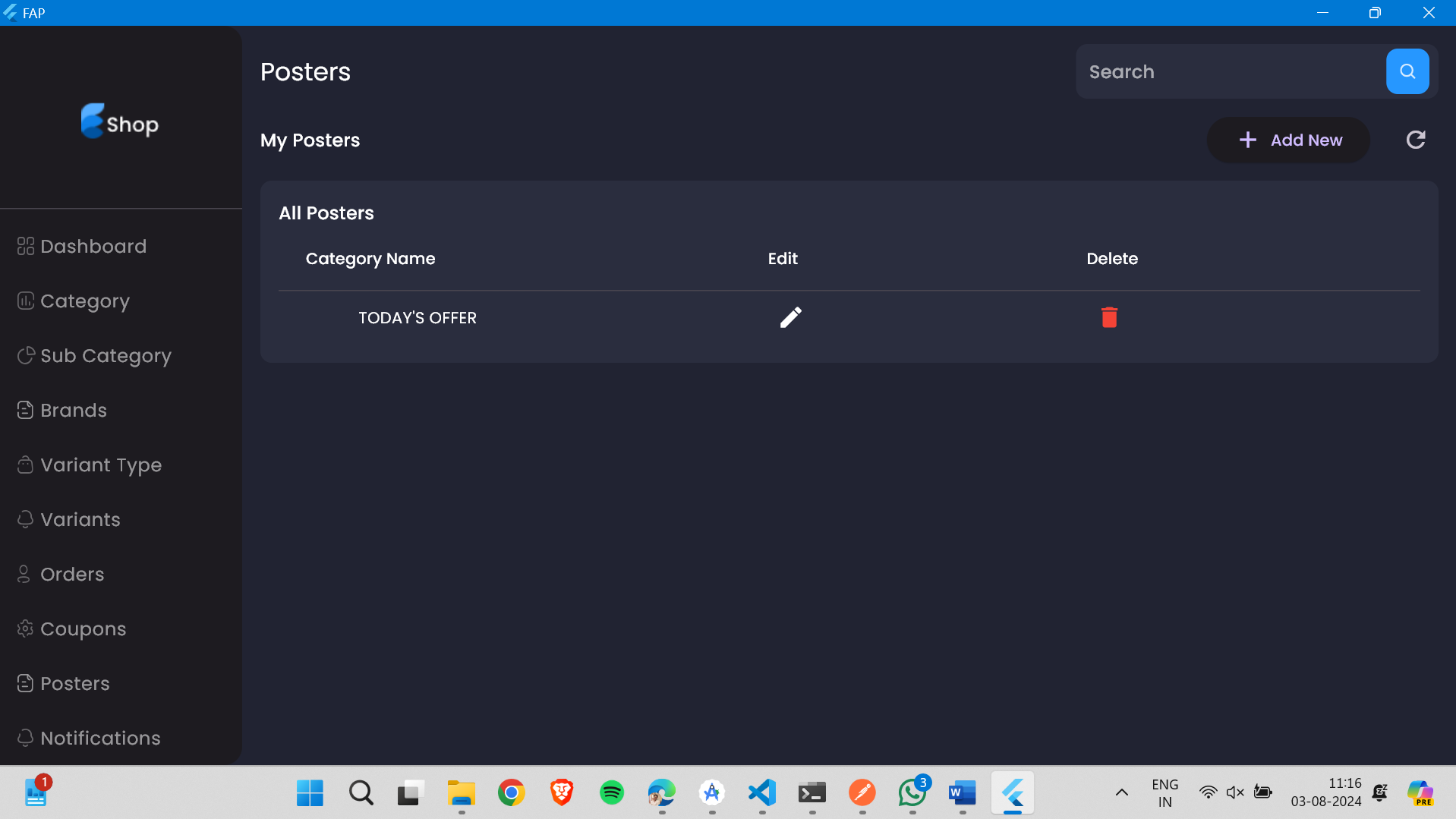
****

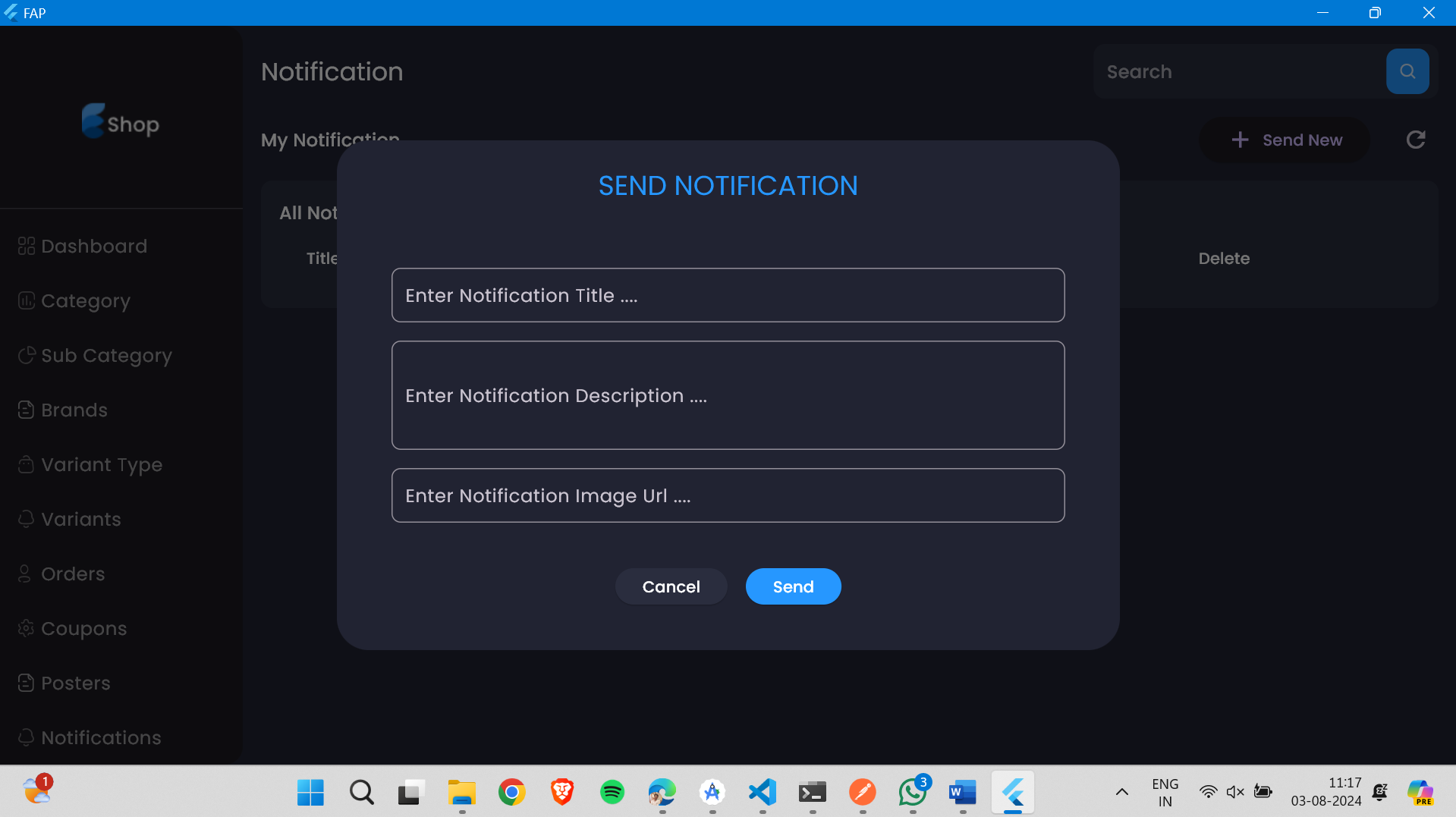
****

****

****

****

****

****

**Appendix -2**

**Codes**

1. **Content-Based Recommendations source code:**

import pandas as pd

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import linear\_kernel

from flask import Flask, jsonify

from pymongo import MongoClient

app = Flask(\_\_name\_\_)

# MongoDB connection

mongo\_url = "mongodb+srv://rishi:5USZwUU3WyYc7nnO@vijayalakshmi.ek3yx74.mongodb.net/?retryWrites=true&w=majority&appName=VijayaLakshmi"

client = MongoClient(mongo\_url)

db = client.get\_database('test')

collection = db.get\_collection('recommendation')

# Load CSV data into a DataFrame

df = pd.read\_csv('products.csv')

# Convert \_id column to string for consistency

df['\_id'] = df['\_id'].astype(str)

# Convert price-related columns to float

df['price'] = pd.to\_numeric(df['price'], errors='coerce')

df['offerPrice'] = pd.to\_numeric(df['offerPrice'], errors='coerce')

df['quantity'] = pd.to\_numeric(df['quantity'], errors='coerce')

# Ensure datetime columns are properly parsed

df['createdAt'] = pd.to\_datetime(df['createdAt'])

df['updatedAt'] = pd.to\_datetime(df['updatedAt'])

# Preprocess description column for TF-IDF (for demonstration)

df['description'] = df['description'].fillna('')

# TF-IDF Vectorizer (for demonstration)

tfidf = TfidfVectorizer(stop\_words='english')

tfidf\_matrix = tfidf.fit\_transform(df['description'])

# Cosine similarity matrix (for demonstration)

cosine\_sim = linear\_kernel(tfidf\_matrix, tfidf\_matrix)

@app.route('/recommendations/<product\_id>', methods=['GET'])

def get\_recommendations(product\_id):

    # Ensure product\_id is converted to string for matching with df['\_id']

    product\_id = str(product\_id)

    # Check if product\_id exists in the DataFrame

    if product\_id not in df['\_id'].values:

        return jsonify({

            "success": False,

            "message": f"Product ID {product\_id} not found.",

            "data": []

        }), 404

    # Get index of the product in DataFrame

    idx = df[df['\_id'] == product\_id].index[0]

    # Calculate cosine similarity scores

    sim\_scores = list(enumerate(cosine\_sim[idx]))

    sim\_scores = sorted(sim\_scores, key=lambda x: x[1], reverse=True)

    sim\_scores = sim\_scores[1:4]  # Limiting to 3 recommendations

    # Get indices of recommended products

    product\_indices = [i[0] for i in sim\_scores]

    # Prepare recommendations in the required format

    recommendations = []

    for index in product\_indices:

        recommendation = {

            "\_id": str(df.loc[index, "\_id"]),

            "name": df.loc[index, "name"],

            "main\_image": df.loc[index, "images[0].url"],

            "price": float(df.loc[index, "price"]),

            "offer\_price": float(df.loc[index, "offerPrice"])

        }

        recommendations.append(recommendation)

    # Return JSON response in the specified format

    response = {

        "success": True,

        "message": "Recommendations retrieved successfully.",

        "data": recommendations

    }

    return jsonify(response)

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(port=5000, debug=True)