E-Commerce Store

A PROJECT REPORT

By

Srinivas Sobhit Kintali (RA2111031010041)

Under the Guidance of

Mr. J.Prabakaran

(Assistant Professor, Department Networking and Communication)

In partial fulfillment of the requirements for the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

with specialization in Information Technology

DEPARTMENT OF NETWORKING AND COMMUNICATIONS

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR-603 203

MAY 2024

BONAFIDE CERTIFICATE

Certified that this course project of Database Management System (18CSC303J) report titled "**E-Commerce Store Management System** " is the Bonafide work of **Srinivas Sobhit Kintali (RA2111031010041)** who carried out the project work under my supervision.

Certified further, that to the best of my knowledge the work reported here in does not form part of any other course project report on the basis of which a degree was conferred on an earlier occasion for this or any other candidate.

Mr. Prabakaran J
Assistant Professor
Department of Networking and Communications
SRM Institute of Science and Technology, Kattankulathur

Dr. Annapurani Panaiyappan K
Professor and Head of the Department
Department of Networking and Communications
SRM Institute of Science and Technology, Kattankulathur

Acknowledgement

I would like to express our gratitude to our Assistant Professor, **Mr. Prabakaran J** who gave us the golden opportunity to do this wonderful project on the topic **"E-Commerce Store Management System"** which also helped us in doing a lot of research and we came to know about so many new things we are really thankful to him.

I would also thankful to all the other faculty, teaching and non-teaching staff members of our department for their kind co-operation and help.

Lastly, I would also like to thank our friends who helped us a lot in finishing this project within the limited time. We are making this project not only for marks but to also increase our knowledge.

SRINIVAS SOBHIT KINTALI (RA2111031010041)

Table of Contents

S.No	Chapter	Page No
1	Introduction	1
2	Scope	2
3	Requirement Analysis	4
4	Data Modeling	6
5	Software Requirements	7
6	Sample Source Code	10
7	Sample Screen shots	12
8	Conclusion	14
9	References	14

Introduction:

The E-commerce Store Management System (ESMS) project endeavors to modernize retail operations by harnessing digital solutions to enhance customer experiences, streamline business processes, and drive sustainable growth in the online marketplace. In response to the evolving landscape of retail and consumer behavior, there is a critical need for innovative platforms that can seamlessly integrate sales, inventory management, and customer engagement to meet the demands of today's discerning shoppers.

This project addresses these challenges by developing a comprehensive database management system tailored specifically for e-commerce businesses. By leveraging advanced technologies and industry best practices, the ESMS aims to empower retailers with tools and insights to optimize sales, manage inventory effectively, and deliver personalized shopping experiences to customers.

Scope:

The scope of the project encompasses the development of a versatile e-commerce platform with features tailored for both customers and administrators. Here is a detailed outline of the project's scope:

1. **Customer Experience Enhancement:** The ESMS includes modules for browsing products, adding items to cart, and completing purchases seamlessly. Customers can create accounts, manage profiles, track orders, and receive personalized recommendations based on their purchase history and preferences.
2. **Administrative Tools:** The system provides robust backend functionalities for administrators, including inventory management, order processing, sales analytics, and customer relationship management (CRM). Administrators can manage product listings, track inventory levels, analyze sales performance, and communicate with customers efficiently.
3. **Security and Compliance:** The ESMS prioritizes data security and compliance with regulations such as GDPR (General Data Protection Regulation) and PCI DSS (Payment Card Industry Data Security Standard). It implements encryption protocols, access controls, and regular audits to safeguard sensitive customer information and ensure regulatory compliance.
4. **Scalability and Flexibility:** The platform is designed to accommodate the evolving needs of e-commerce businesses, including scalability to handle increased traffic and transaction volumes, as well as flexibility to integrate with third-party services and adapt to changing market trends.
5. **User Experience Optimization:** The ESMS focuses on delivering an intuitive and engaging user experience across desktop and mobile devices. It incorporates responsive design principles, intuitive navigation, and personalized content to enhance user satisfaction and retention.
6. **Reporting and Analytics:** The system provides robust reporting and analytics capabilities, allowing hospital administrators to generate various reports such as patient demographics, financial summaries, operational metrics, and performance indicators. These reports help in monitoring key performance indicators (KPIs),

identifying trends, and making data-driven decisions.

7. **Security and Compliance:** The system prioritizes data security and compliance with industry standards such as PCI DSS and GDPR. It implements encryption, authentication, and authorization mechanisms to protect sensitive data and ensure regulatory compliance in handling customer information and payment transactions.
8. **Usability and Scalability:** The e-commerce platform emphasizes user-friendly design and scalability. Its intuitive interface and responsive layout enable smooth navigation across devices. Leveraging cloud infrastructure, the platform ensures seamless performance, even during peak traffic. Customization features empower users to personalize their experience, while feedback mechanisms drive ongoing enhancements for optimal usability and scalability.

Requirement Analysis:

3.1. Functional Requirements:

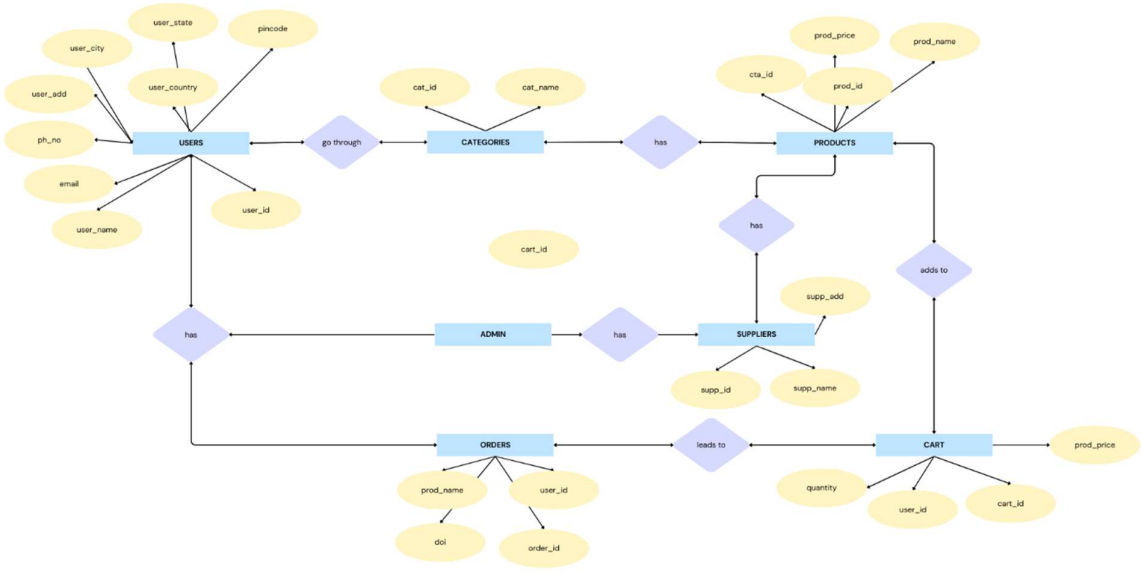
1. **Customer Management Module:**
   - Ability to register new customers and manage customer profiles.
   - Options for customers to update personal information, view order history, and track shipments.
   - Integration with third-party authentication services for streamlined login and registration processes.
2. **Product Management Module:**
   - Use Tools for administrators to add, edit, and delete product listings.
   - Support for categorizing products, assigning attributes, and managing inventory levels.
   - Features for displaying product details, images, pricing, and availability status.
3. **Order Processing Module:**
   - Functionality for customers to add items to cart, proceed to checkout, and complete purchases securely.
   - Integration with payment gateways for accepting multiple payment methods, including credit/debit cards, digital wallets, and online banking.
   - Automated order confirmation emails, shipping notifications, and tracking information for customers.
4. **Inventory Management Module:**
   - Tools for tracking inventory levels, receiving stock, and updating product availability.
   - Features for setting reordering thresholds, generating purchase orders, and managing supplier relationships.
   - Real-time inventory updates to prevent overselling and stockouts, ensuring seamless order fulfillment.
5. **Inventory Management Module:**
   - Dashboard with key performance indicators (KPIs) such as sales revenue, order volume, and customer acquisition metrics.
   - Customizable reports for analyzing sales trends, product performance, and customer behavior.
   - Integration with analytics tools for advanced data visualization and predictive insights.

3.2. Non-Functional Requirements:

1. **Performance:**
    - Response time for key functions like product search, checkout process, and order confirmation should be within seconds.
    - Scalability to handle peak traffic loads during promotional events and holiday seasons without performance degradation.
    - Load testing and performance optimization to ensure consistent performance under varying load conditions.
2. **Security:**
    - Implementation of robust data encryption mechanisms to protect patient information at rest and in transit.
    - Access controls based on roles and privileges to ensure only authorized personnel can access sensitive data.
    - Regular security audits and vulnerability assessments to identify and mitigate potential risks.
3. **Usability:**
    - Intuitive user interface design with easy navigation and minimal training requirements for users.
    - Support for multiple devices such as desktops, tablets, and smartphones for accessibility from anywhere within the hospital premises.
    - Customizable dashboards and reporting options for personalized user experience.
4. **Reliability:**
    - High availability with minimal downtime for system maintenance and upgrades.
    - Automated backup and recovery procedures to prevent data loss in case of system failures.
    - Failover mechanisms to ensure continuity of critical operations during unexpected outages.

Data Modeling:

a. ER Diagram:



Software Requirements:

a. Front End:

1. **User Interface Design:**
   - Responsive and user-friendly interface for seamless navigation.
   - Customizable dashboards for different user roles (administrators, healthcare providers, patients, etc.).
   - Intuitive forms for data entry and editing patient information, appointments, and medical records.
2. **Technologies:**
   - HTML5, CSS3, and JavaScript for front-end development.
   - Front-end development tools such as VS Code used for code editing and debugging.
   - Version control systems (e.g., Git) for collaborative development and code management.
   - Browser developer tools for testing, debugging, and optimizing web applications across different browsers and devices.
3. **User Authentication:**
   - Secure login and authentication system with role-based access control.
   - Password encryption and hashing techniques for user credential protection.
   - Optional two-factor authentication for enhanced security.

b. Back End:

1. **Application Logic:**
   - Server-side scripting and business logic implementation using PHP.
   - Integration with external systems for data exchange (e.g., diagnostic systems, payment gateways).
   - Error handling and exception management for robust application performance.
2. **Technologies used:**
   - Database management systems (DBMS) such as MySQL, PostgreSQL, or MongoDB for storing and managing e-commerce data.
   - ORM (Object-Relational Mapping) libraries (e.g., Sequelize, SQLAlchemy) for simplified database interactions and data modeling.
   - Containerization technologies (e.g., Docker) for packaging and deploying microservices and server-side applications
3. **Security Measures:**
   - Implementation of secure coding practices to prevent vulnerabilities like SQL injection, XSS attacks, etc.
   - Regular code reviews and security testing to identify and mitigate potential risks.
   - Logging and monitoring mechanisms for detecting and responding to suspicious activities.

c. Database Connectivity:

1. **Database Management System (DBMS):**
   - Selection of a suitable DBMS based on data modeling requirements, scalability considerations, and performance benchmarks.
   - Installation, configuration, and optimization of the DBMS for efficient data storage, retrieval, and management.

2. **Data Schema Design:**
   - Designing normalized relational database schemas based on the application's data model and requirements.
   - Indexing, partitioning, and optimization strategies for improving query performance and resource utilization.
3. **Database Connectivity:**
   - Establishing secure connections between the back-end application and the database server using encrypted protocols (e.g., SSL/TLS).
   - Implementing connection pooling, caching, and query optimization techniques to minimize latency and maximize throughput.
   - Backup and recovery procedures to ensure data integrity and availability in case of system failures or data loss events.

User Side Features:

1. **Browsing and Shopping**: Users can browse products, view details, and add items to their cart for purchase.
2. **Account Management:** Users can create accounts, manage personal information, and track order history.
3. **Checkout Process**: Seamless checkout process with secure payment options for completing purchases.
4. **Cart Availability:** Users can add the products to cart and proceed batchwise to buy.
5. **Order History:** Users can see the history of purchases.

Admin Side Features:

1. **Product Management:** Admins can add, edit, and delete products, including details like name, price, and description.
2. **Order Management:** Ability to view and manage orders, including processing payments, updating order status, and generating invoices.
3. **User Management:** Admins can manage user accounts, including creating new accounts, resetting passwords, and handling account-related issues.
4. **Inventory Management:** Admins can track inventory levels, receive notifications for

low stock items, and manage restocking activities.

5. **Reporting and Analytics:** Access to sales reports, inventory insights, and other analytics to monitor business performance and make data-driven decisions.
6. **Content Management:** Admins can manage static content such as homepage banners, promotional messages, and site policies.
7. **Customer Support:** Ability to respond to customer inquiries, resolve issues, and provide assistance as needed to ensure a positive shopping experience.

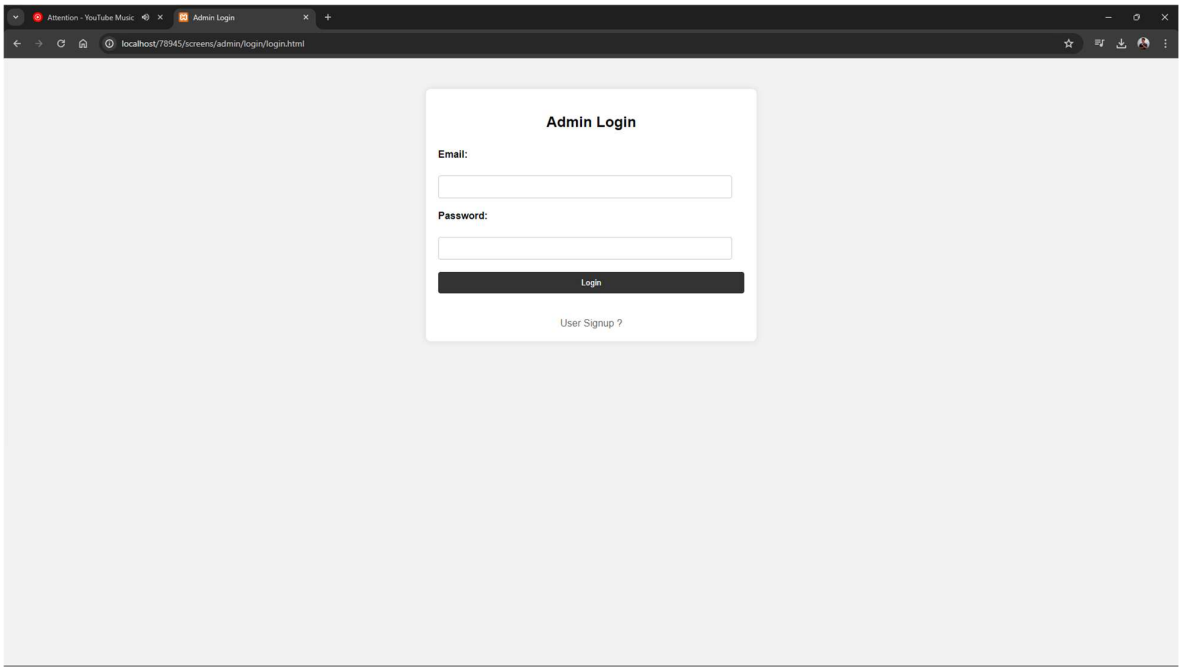
Source code:

```sql
CREATE TABLE users (
    USER_ID INT PRIMARY KEY,
    USER_NAME VARCHAR(25),
    PH_NO VARCHAR(15),
    EMAIL VARCHAR(25),
    USER_ADD VARCHAR(15),
    USER_CITY VARCHAR(15),
    USER_STATE VARCHAR(15),
    USER_COUNTRY VARCHAR(15),
    USER_PINCODE VARCHAR(10)
);

CREATE TABLE categories (
    CAT_ID INT PRIMARY KEY,
    CAT_NAME VARCHAR(25)
);

CREATE TABLE products (
    PROD_ID INT PRIMARY KEY,
    PROD_NAME VARCHAR(25),
    CAT_ID INT,
    PROD_PRICE DECIMAL(10, 2),
    SUPP_ID INT,
    FOREIGN KEY (CAT_ID) REFERENCES categories(CAT_ID),
    FOREIGN KEY (SUPP_ID) REFERENCES suppliers(SUPP_ID)
);

CREATE TABLE suppliers (
    SUPP_ID INT PRIMARY KEY,
    SUPP_NAME VARCHAR(25),
    SUPP_ADD VARCHAR(50)
);

CREATE TABLE cart (
    CART_ID INT PRIMARY KEY,
    PROD_ID INT,
    QUANTITY INT,
    TOTAL_AMT DECIMAL(10, 2),
    USER_ID INT,
    FOREIGN KEY (PROD_ID) REFERENCES products(PROD_ID),
    FOREIGN KEY (USER_ID) REFERENCES users(USER_ID)
);
```

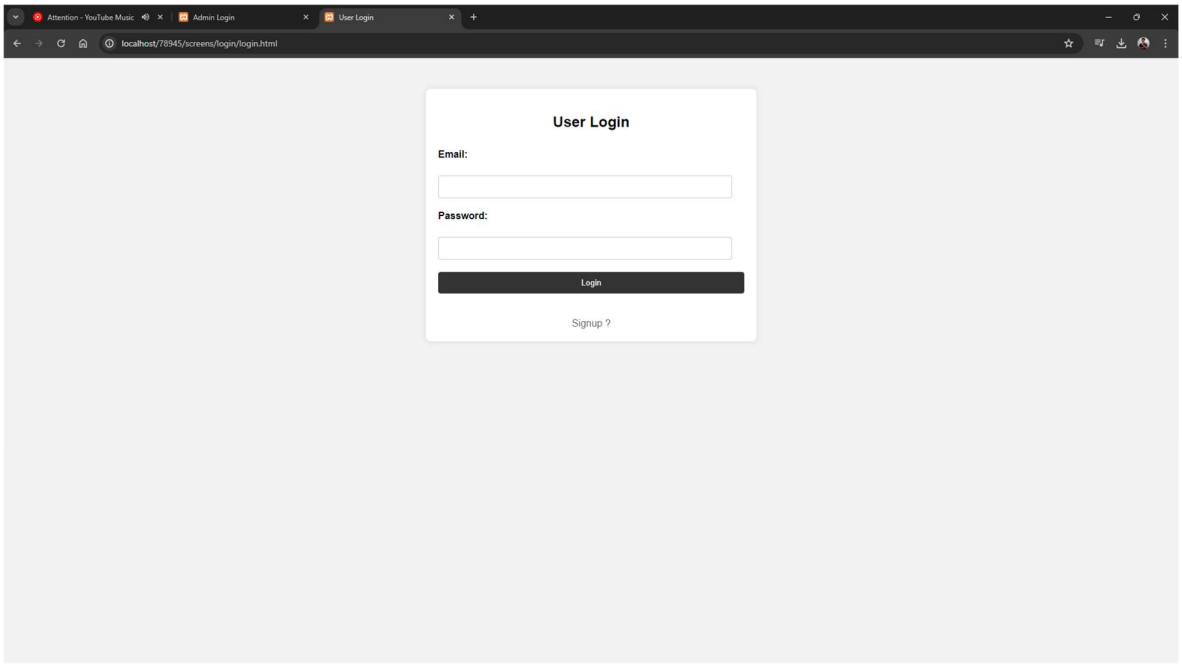
```sql
CREATE TABLE invoice (
    INVOICE_ID INT PRIMARY KEY,
    DOI DATE,
    CART_ID INT,
    PAYMENT_MODE VARCHAR(7),
    USER_ID INT,
    FOREIGN KEY (CART_ID) REFERENCES cart(CART_ID),
    FOREIGN KEY (USER_ID) REFERENCES users(USER_ID)
);
```

Sample Screenshots:

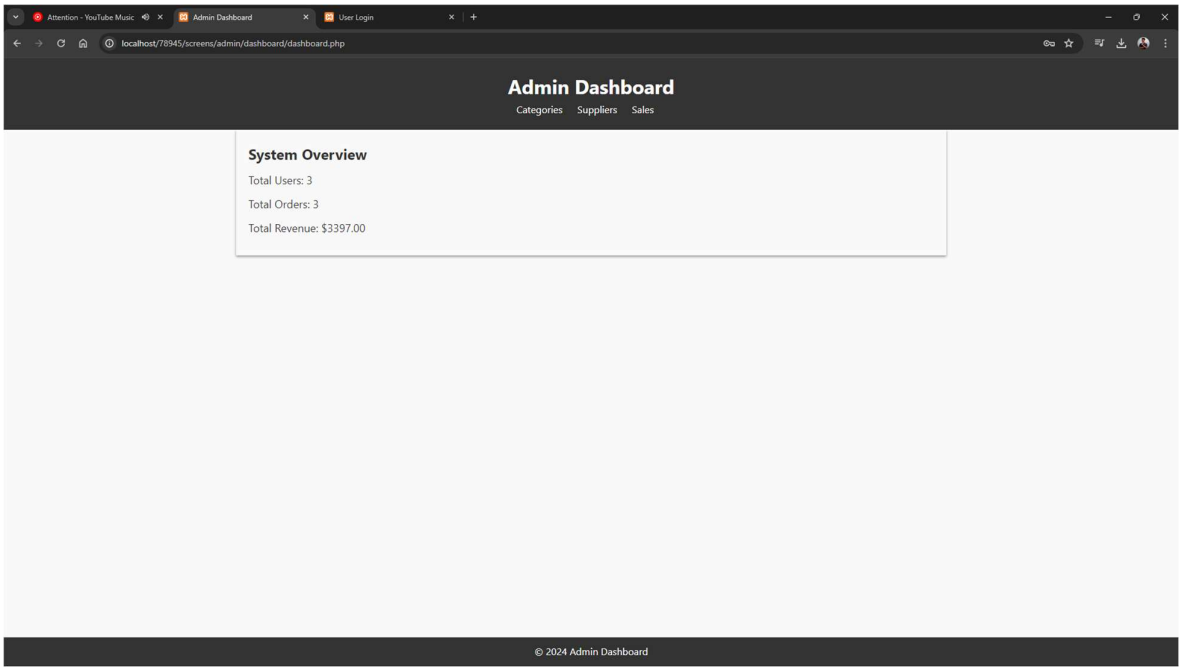
ADMIN SIDE LOGIN:



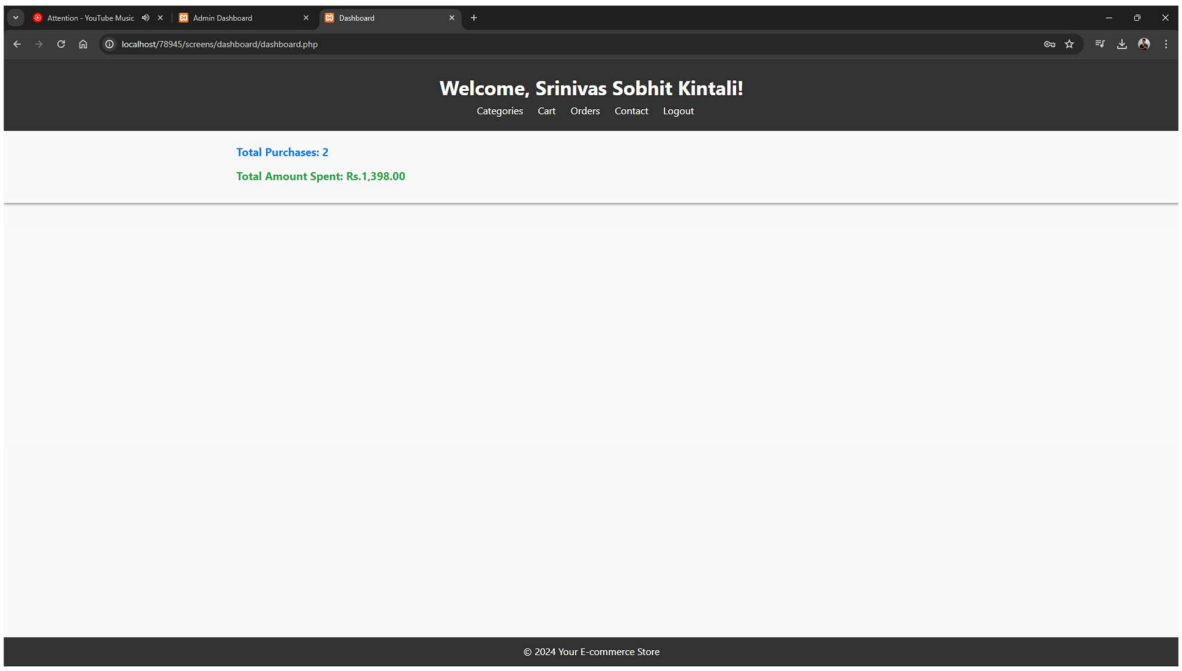
CUSTOMER SIDE LOGIN:



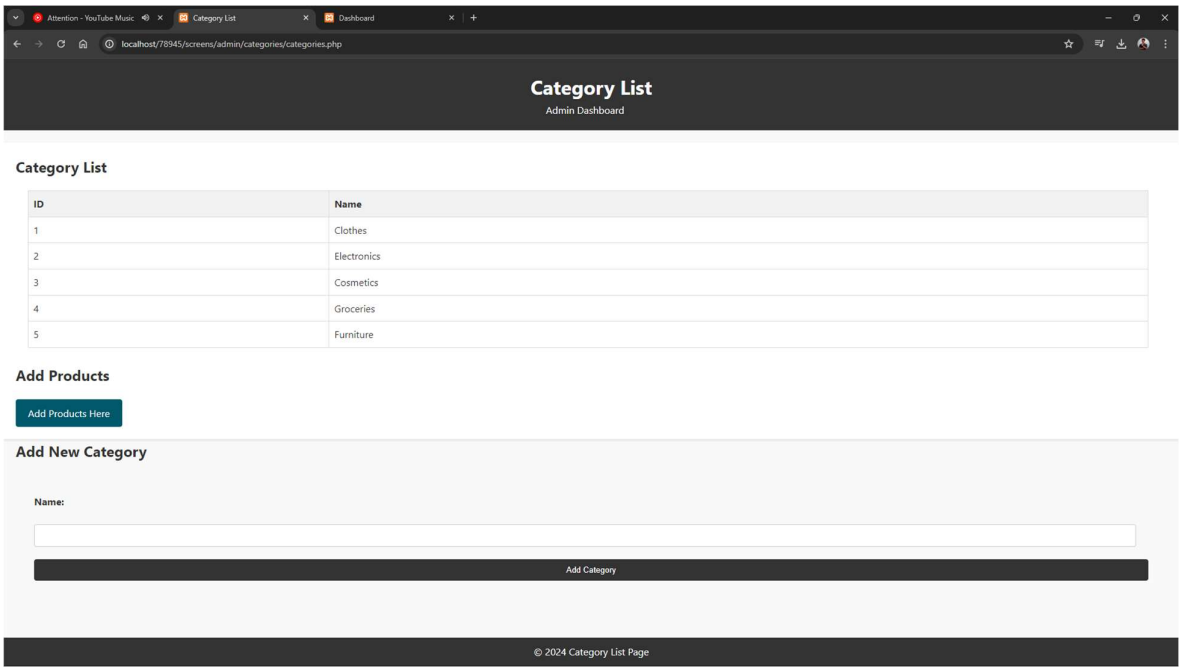
ADMIN DASHBOARD:



USER DASHBOARD:



CATEGORIES PAGE ON ADMIN SIDE:



Conclusion:

The E-commerce Store Management System project represents a significant milestone in modernizing retail operations and enhancing customer experiences in the digital age. By leveraging cutting-edge technologies, robust software architecture, and user-centric design principles, this project aims to empower retailers with tools and insights to thrive in today's competitive e-commerce landscape.

Through meticulous requirement analysis, data modeling, and software development, the ESMS offers a scalable, secure, and feature-rich platform for retailers to manage their online stores effectively, drive sales growth, and build lasting relationships with customers. By embracing innovation and embracing change, this project sets the stage for the future of e-commerce, where agility, adaptability, and customer-centricity are paramount to success.

References:

1. https://www.w3schools.com/sql/sql_syntax.asp
2. https://www.worldscientific.com/doi/abs/10.1142/9789811245992_0015