Project Report On

# ShopVerse – Multi-Vendor E-Commerce Platform

Submitted in partial fulfillment for the award of

**Post Graduate Diploma in Advanced Computing** from

**C-DAC ACTS (Pune)**

**Guided by**

**Mr. Vinu Josy**

**Presented By**


**Divyani Nale- 250840120051**

**Komal Dhande 250840120082**

**Onkar Gaikwad- 250840120110**

**Prasanna Munde- 250840120129**

**Chandrashekhar Patil- 250840120040**


**Centre of Development of Advanced Computing (C-DAC), Pune**

# CERTIFICATE

**TO WHOMSOEVER IT MAY CONCERN**

**This is to certify that**

**Divyani Nale- 250840120051**

**Komal Dhande 250840120082**

**Onkar Gaikwad- 250840120110**

**Prasanna Munde- 250840120129**

**Chandrashekhar Patil- 250840120040**

**have successfully completed their project titled**

# "ShopVerse – Multi-Vendor E-Commerce Platform"

**Under the Guidance of Mr. Vinu Josy**

**Project Guide** **HOD ACTS**

# ACKNOWLEDGEMENT

This project “ShopVerse – Multi-Vendor E-Commerce Platform” was a great learning experience for us and we are submitting this work to Advanced Computing Training School (CDAC ACTS).

We all are very glad to mention the name of **Mr. Vinu Josy** for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

Our most heartfelt thank goes to Ms **Swati mam** (Course Coordinator, PGDAC) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC ACTS, Pune.

Divyani Nale- 250840120051

Komal Dhande 250840120082

Onkar Gaikwad- 250840120110

Prasanna Munde- 250840120129

Chandrashekhar Patil- 250840120040

# TABLE OF CONTENTS

# 1. Introduction

Online shopping has become an important part of everyday life, and there is a growing need for reliable and scalable e-commerce platforms. **ShopVerse** is a **multi-vendor e-commerce platform** designed to allow multiple sellers to sell products online while customers can browse, purchase, and manage their orders in a secure and easy way.

This project is developed using **Spring Boot with a microservices architecture**, which helps in building a modular and scalable system. The platform supports different user roles such as **Admin, Seller, and Customer**. Sellers can add and manage products, customers can browse products, add items to cart or wishlist, and place orders, while the admin can manage users, sellers, and overall system activities.

The system uses **Apache Kafka** for handling important events like order creation, payment confirmation, and user activity tracking. This helps in smooth communication between different services and improves system performance. **Secure online payments** are integrated using **Stripe or Razorpay**, ensuring safe and reliable transactions.

Core e-commerce features such as **product management, cart, wishlist, order processing, reviews, and payments** are included in the project. The application uses a **relational database** with a properly designed **Entity-Relationship (ER) Diagram**, which defines entities like User, Product, Order, Cart, Wishlist, and Review to maintain structured and consistent data.

The entire application is containerized using **Docker** and deployed on **AWS cloud infrastructure**, making it scalable and suitable for real-world usage. Overall, this project provides a complete understanding of how a modern multi-vendor e-commerce system works using current backend technologies.

## 2. Software/Hardware Requirement

**Server:**

Processor: Intel Core i5 or equivalent AMD processor.

RAM: Minimum 8GB RAM.

Storage: SSD storage for improved performance.

Network: Ethernet or Wi-Fi connectivity.

Operating System: Linux distribution (Ubuntu, CentOS) preferred for server deployment.

**Client Devices:**

Processor: Dual-core processor or higher.

RAM: Minimum 4GB RAM.

Storage: Sufficient storage for caching and local data.

Network: Ethernet or Wi-Fi connectivity.

Browser: Compatible with latest versions of popular browsers like Google Chrome, Mozilla Firefox, and Safari.

## 3. Tools and technologies used

- SpringBoot
- SpringDataJPA
- RESTful Web
- SpringWeb
- MySql
- JWT
- Git
- Spring Security
- React JS
- HTML and CSS
- Axios
- Stripe payment integration
- Material UI

1. Spring Boot: Spring Boot is used to develop the backend of the application. It helps in creating independent services for handling users, products, orders, and payments. It simplifies configuration and allows fast development of the e-commerce system.

2. Spring Data JPA: Spring Data JPA is used for database operations. It helps in performing CRUD (Create, Read, Update, Delete) operations on entities like User, Product, Order, Cart, Wishlist, and Review without writing complex SQL queries.

3. RESTful Web Services: RESTful APIs are created to enable communication between the frontend and backend. These APIs handle requests such as user registration, product listing, adding items to cart, placing orders, and processing payments.

4. Spring Web: Spring Web is used to build REST controllers and handle HTTP requests like GET, POST, PUT, and DELETE. It manages request routing, validation, and response handling for the application.

5. MySQL: MySQL is used as the relational database to store all application data such as user details, product information, orders, cart items, wishlists, and reviews. It ensures data consistency and structured storage.

6. JWT (JSON Web Tokens): JWT is used for secure authentication and authorization. After login, a token is generated and sent to the client. This token is used to access protected APIs without requiring the user to log in repeatedly.

7. Axios: Axios is used in the React application to communicate with backend REST APIs. It helps in sending HTTP requests (GET, POST, PUT, DELETE) and handling responses such as fetching products or submitting orders.

8. React: React JS is used to build the frontend of the application. It creates a responsive and interactive user interface for browsing products, managing cart and wishlist, placing orders, and viewing order history.

9. HTML and CSS: HTML and CSS are used to design the structure and layout of the web pages. They help in building forms, product cards, navigation bars, and responsive layouts.

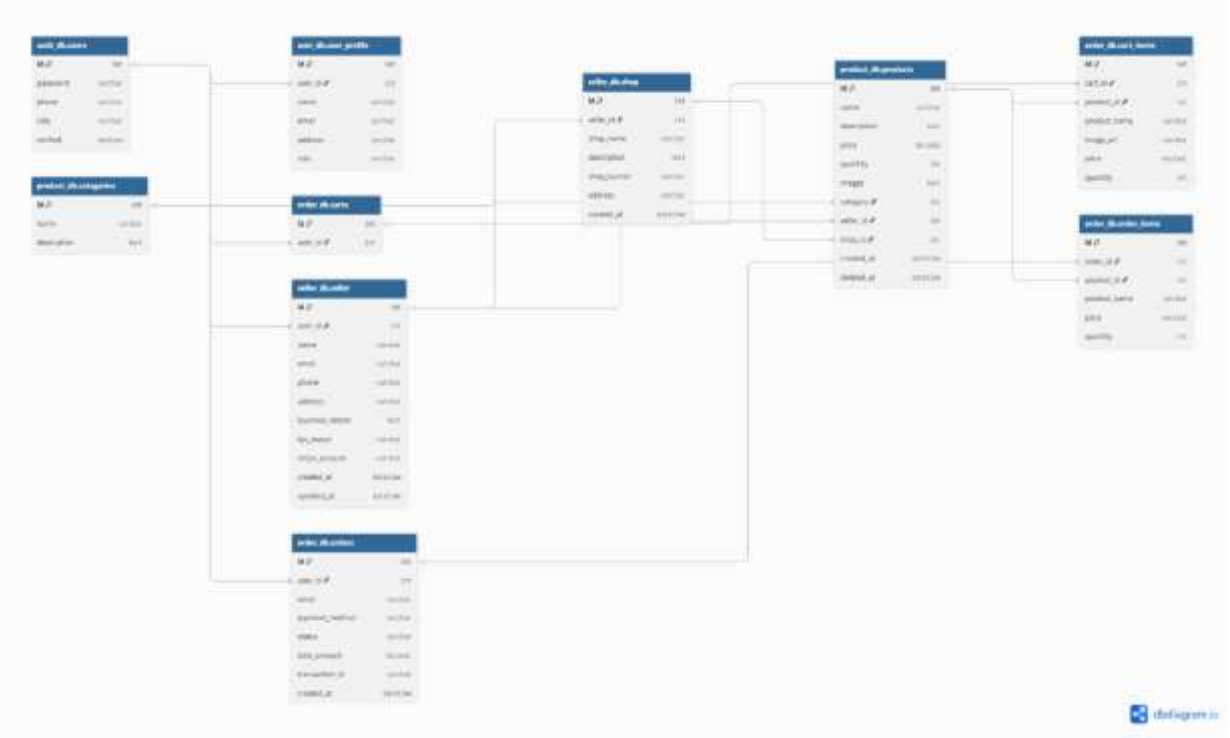
10. Material UI: Material UI is used to design modern and user-friendly UI components in React. It provides ready-made components like buttons, forms, dialogs, cards, and tables, improving the look and usability of the application.

11. Git: Git is used for version control. It helps track code changes, manage different branches, and collaborate efficiently during project development.

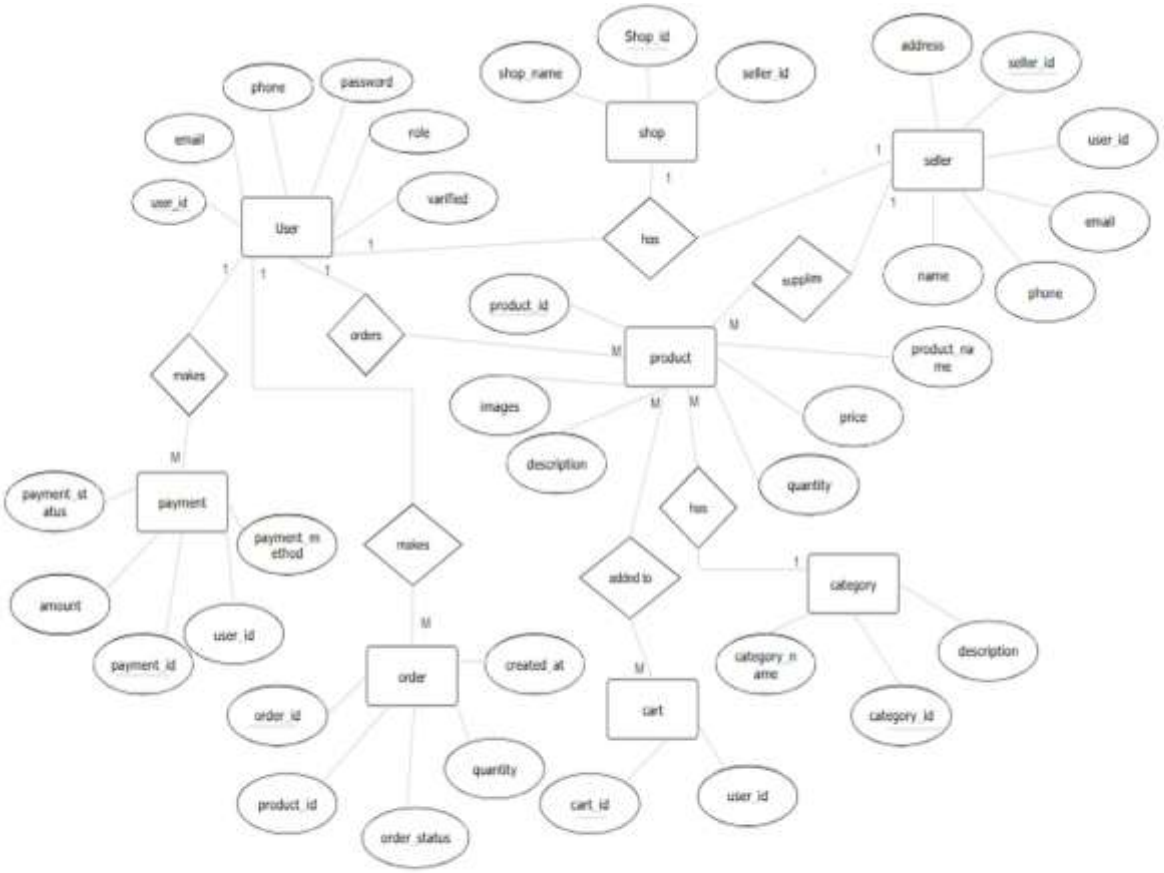
12. Stripe Payment Integration: Stripe is integrated to handle secure online payments. When a user checks out, payment details are sent to Stripe, and on successful payment, the order is confirmed and stored in the database.

13. Spring Security: Spring Security is used to secure the application. It handles user authentication, role-based access control (Admin, Seller, Customer), password encryption, and API protection using JW

# 4. Project Database Diagram



# 5.Project E-R(Entity relationship) Diagram




(ShopVerse ER Diagram)

# 6. Advantages

1. **Multi-Vendor Support**

   The project allows multiple sellers to register and sell their products on a single platform. This makes the system scalable and suitable for real-world e-commerce use.

2. **Secure Authentication**

   JWT and Spring Security are used to provide secure login and role-based access for Admin, Seller, and Customer. User data and system resources are well protected.

3. **Scalable Architecture**

   The use of Spring Boot and microservices architecture makes the application scalable. New features or services can be added easily without affecting existing functionality.

4. **User-Friendly Interface**

   React JS with Material UI provides a clean, modern, and responsive user interface, improving the overall user experience.

5. **Secure Online Payments**

   Stripe payment integration ensures safe and reliable online transactions, increasing trust and usability of the platform.

6. **Smooth Frontend–Backend Communication**

   RESTful APIs and Axios enable efficient communication between the frontend and backend, making data flow smooth and reliable.

7. **Real-World Application**

   The project follows industry standards and reflects real-world e-commerce systems, making it valuable for practical learning and professional use.

8. **Cloud & Deployment Ready**

The project is designed to be deployable using Docker and cloud platforms, making it suitable for production environments.

## 7. Screenshots

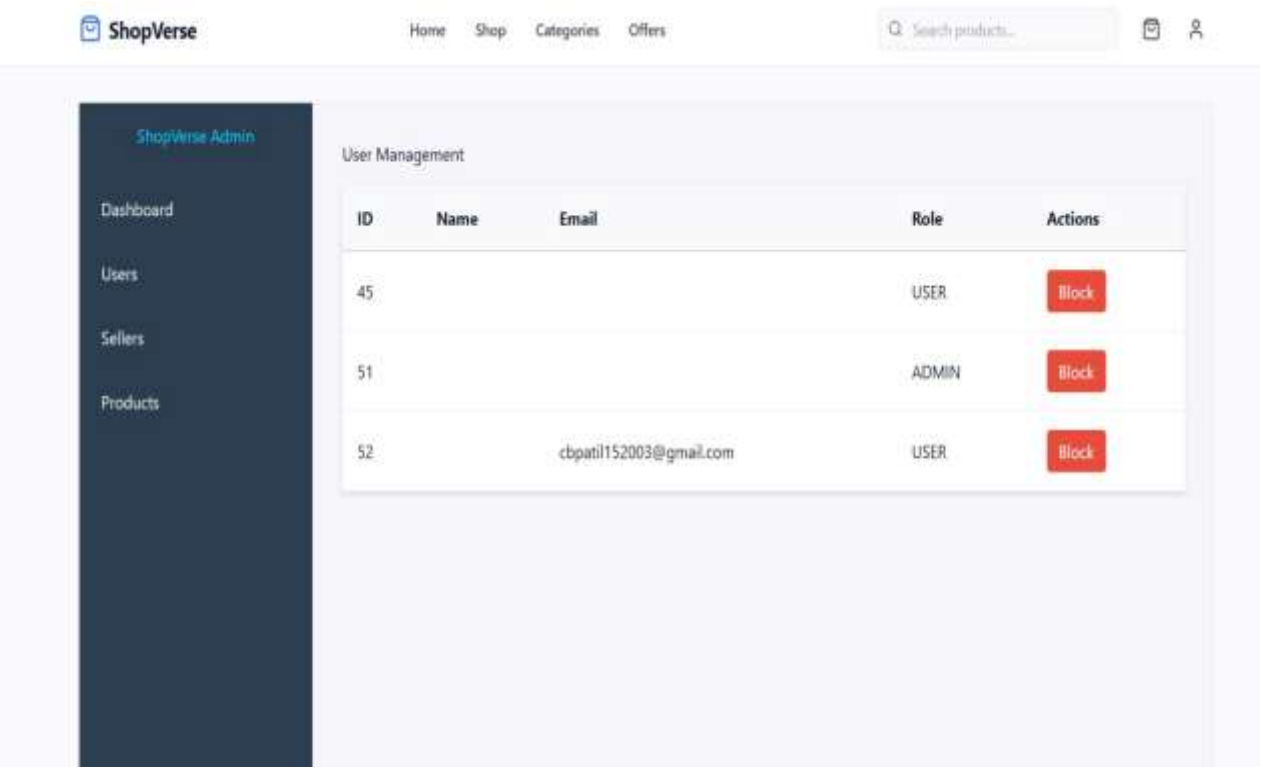



**Fig1. Admin Dashboard**

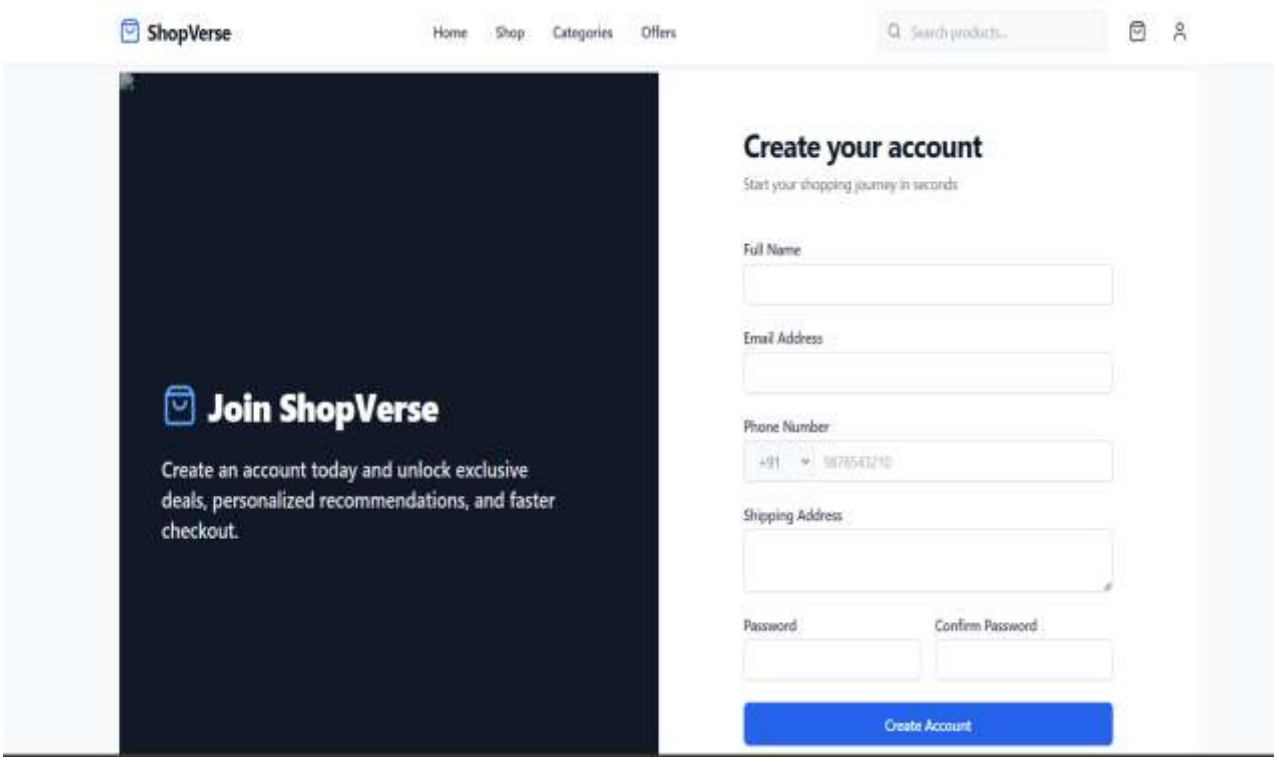



**Fig2. Registration Page**

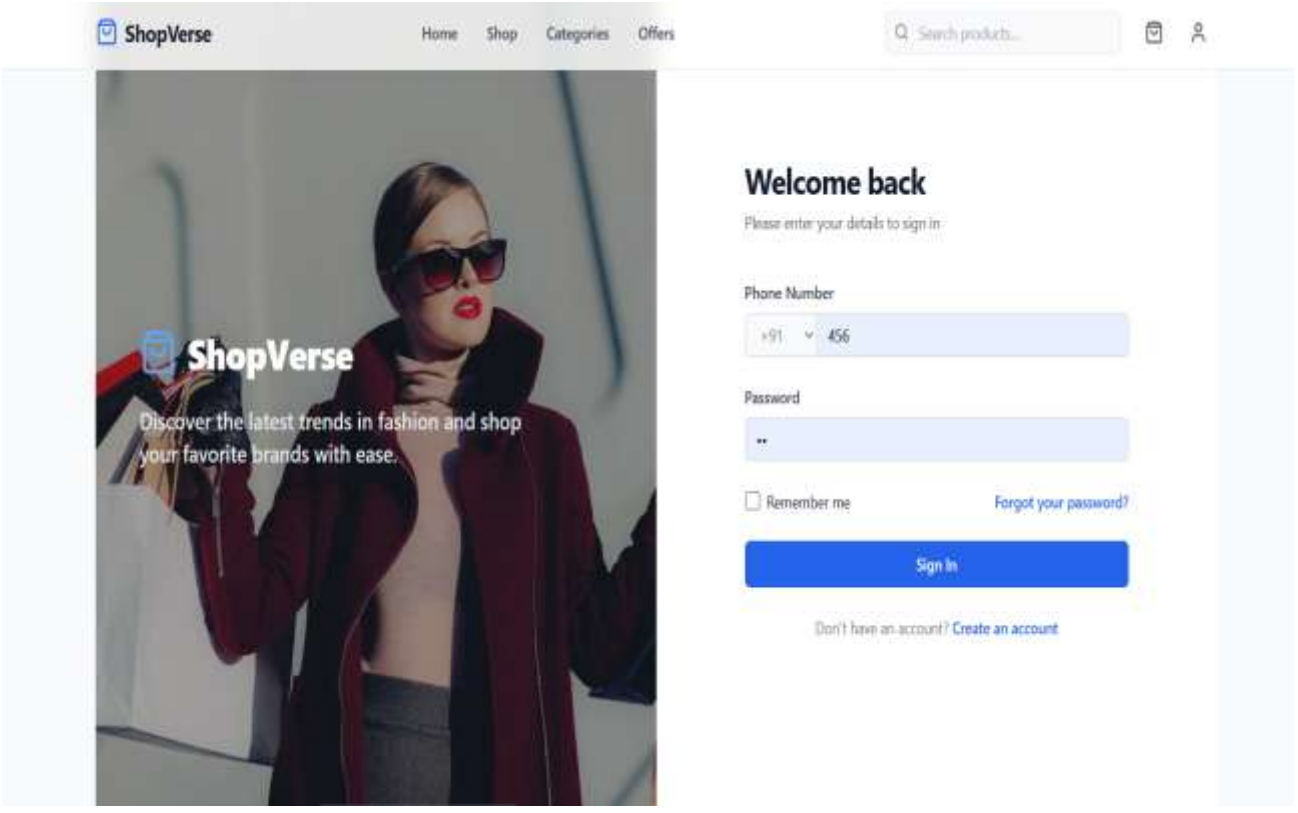



**Fig3. Login Page**

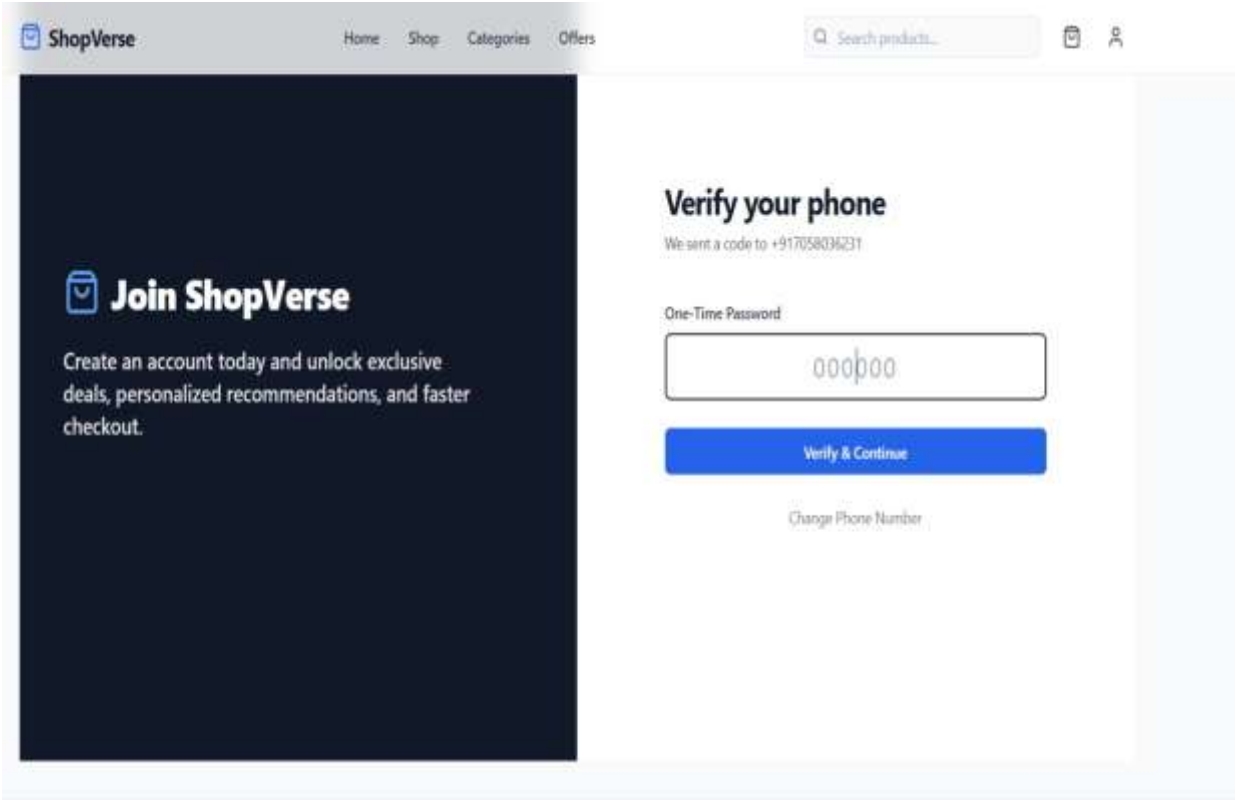



**Fig4.User Authentication Page**

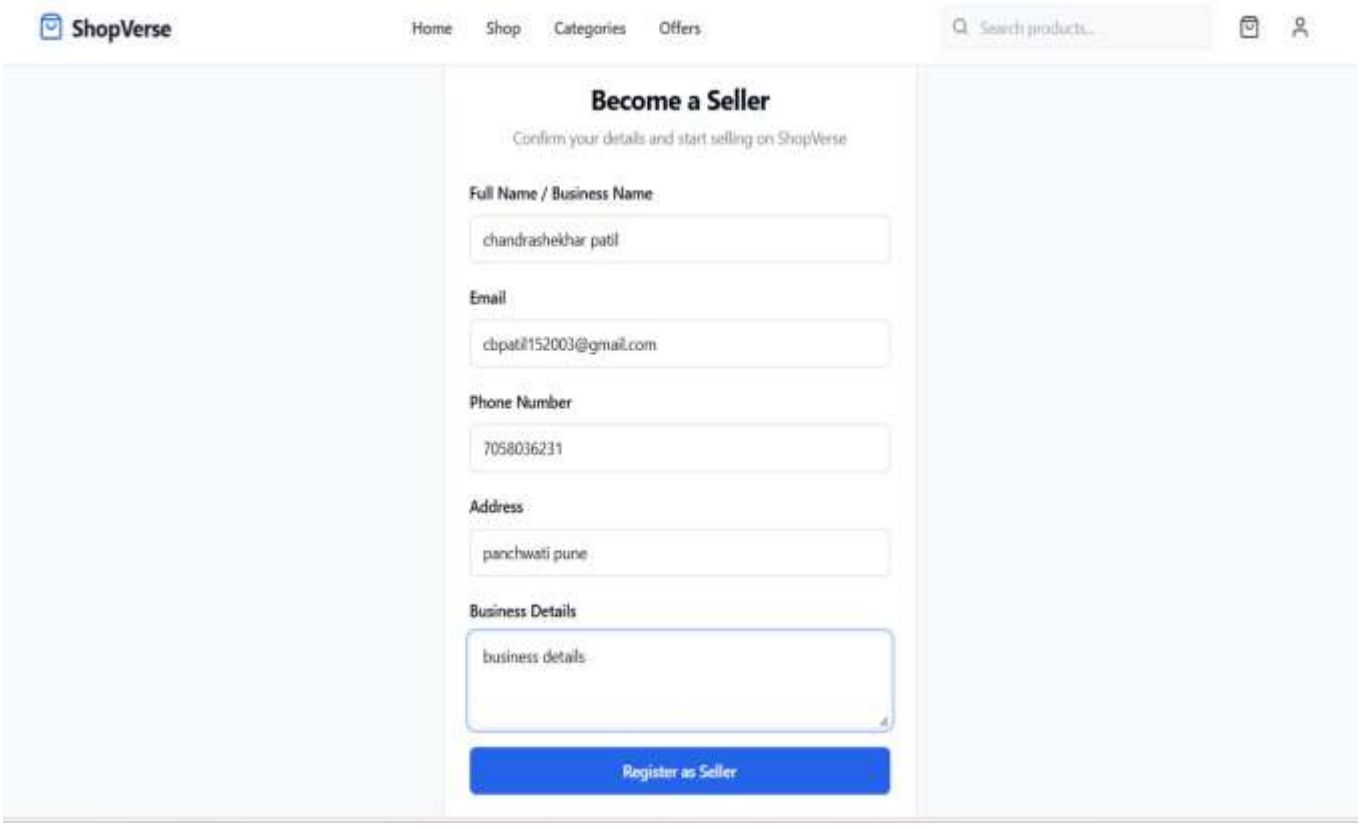



**Fig5. Add Seller**

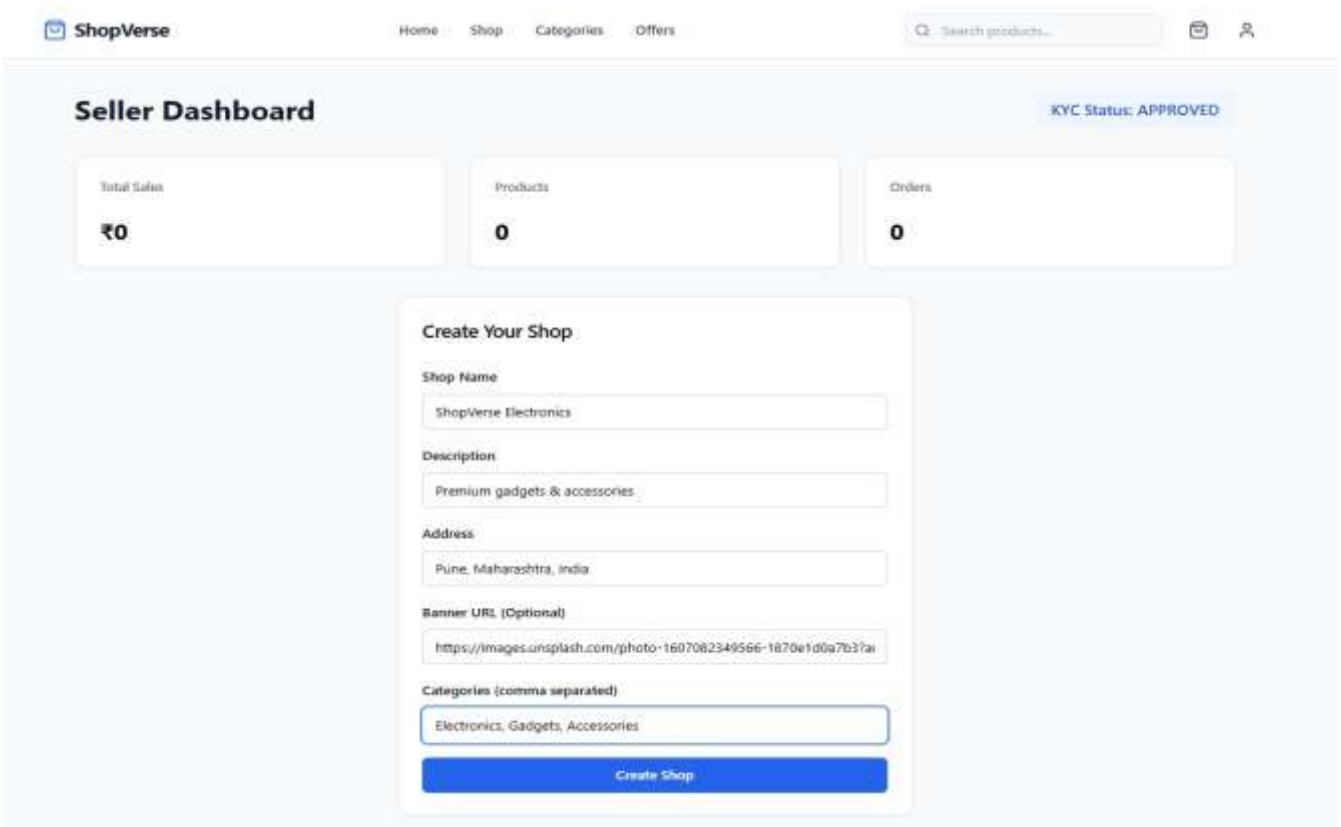



**Fig6. Seller Dashboard**

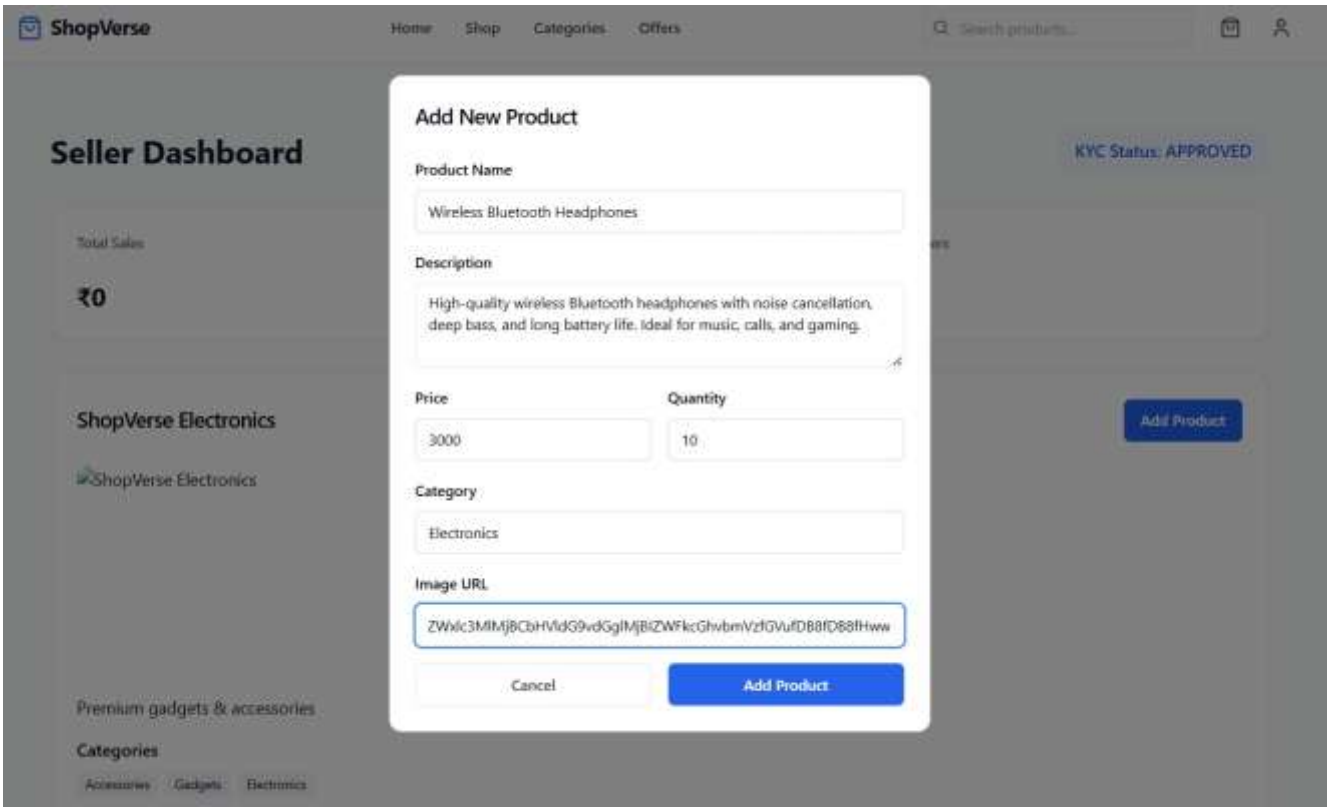



**Fig7. Add Product**

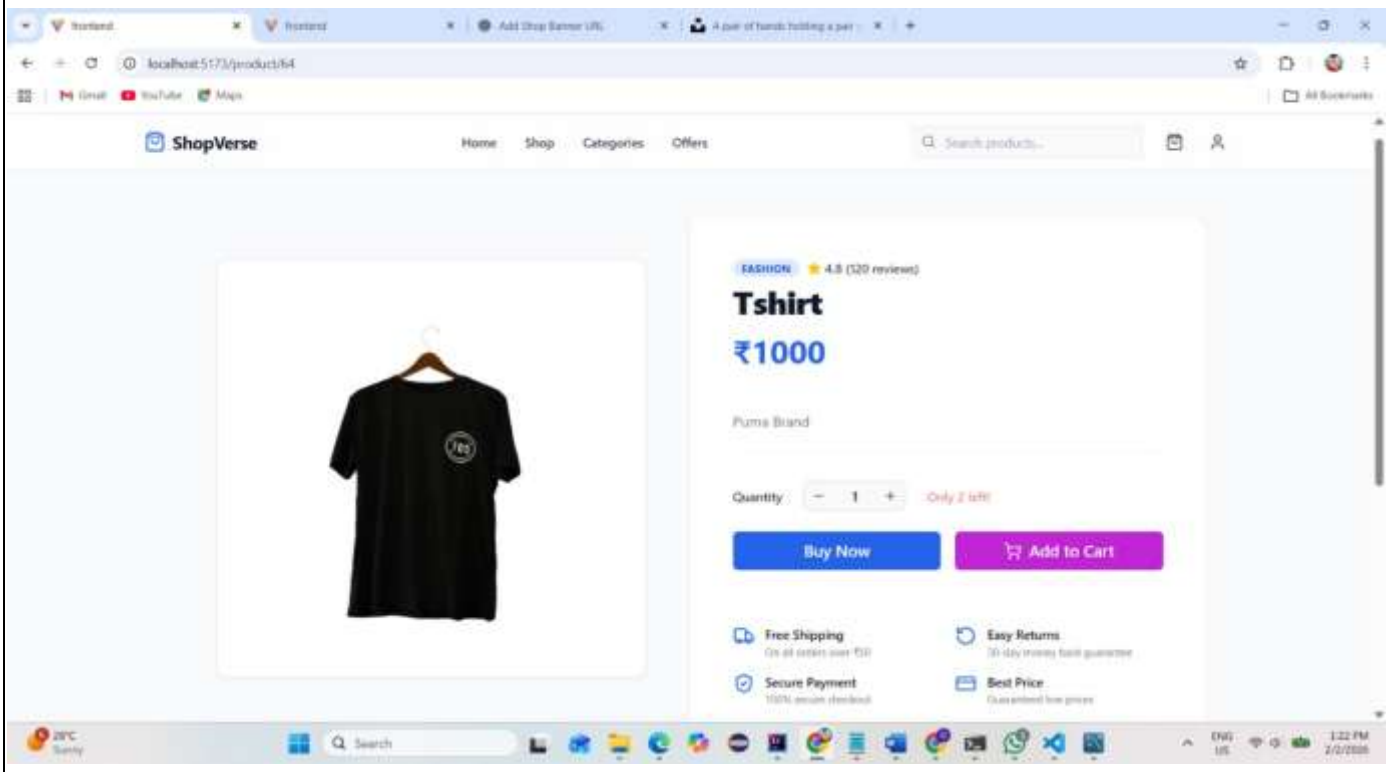



**Fig8. Product**

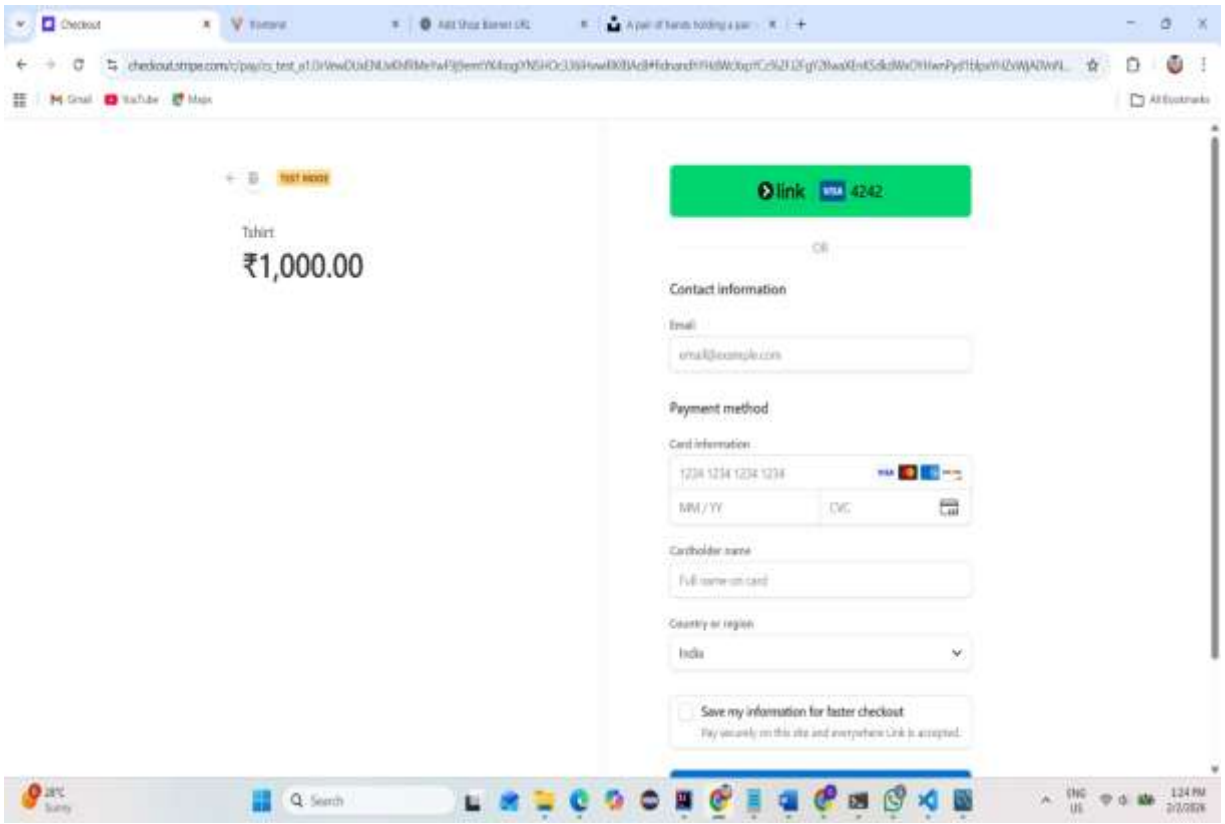
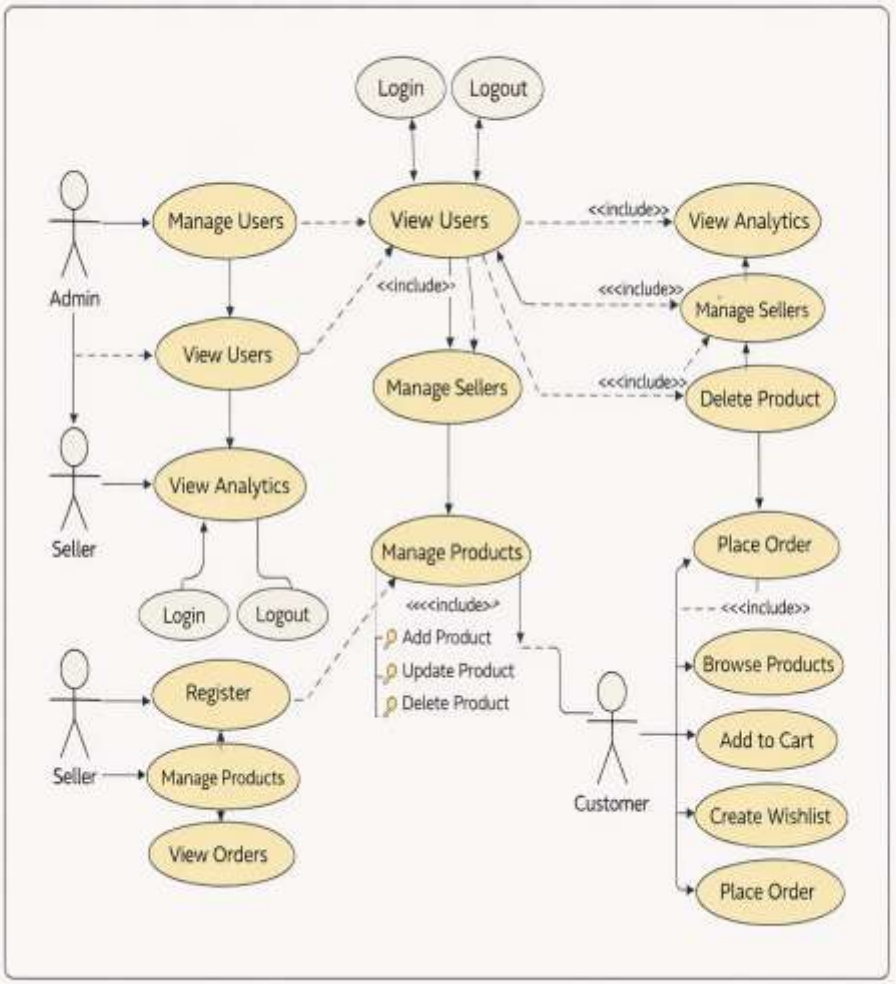



**Fig9. Transaction page**

## Use Case Diagram - Multi-Vendor E-Commerce Platform



# 8. FUTURE SCOPE:

1. **Mobile Application Development**

   The platform can be extended into a mobile application using technologies like React Native or Flutter. This will allow users to shop easily from smartphones and tablets.

2. **Advanced Search and Filters**

   More advanced search options such as price range, brand, ratings, and availability can be added to improve product discovery and user experience.

3. **Product Recommendation System**

   In the future, an intelligent recommendation system can be integrated to suggest products based on user behavior, search history, and previous purchases.

4. **Real-Time Order Tracking**

   Live order tracking can be added to show users the real-time status of their orders, from processing to delivery.

## 9. Conclusion

The ShopVerse project successfully implements a multi-vendor e-commerce platform using modern web technologies. It provides secure user authentication, product management, cart and order handling, and safe online payments. The project offers a user-friendly interface and reliable backend services, making it a practical and scalable solution. Overall, it gives strong hands-on experience in full-stack development and reflects real-world e-commerce system design.

## References


1. https://spring.io/projects/spring-boot
2. https://spring.io/projects/spring-data-jpa
3. https://restfulapi.net/
4. https://www.mysql.com/
5. https://spring.io/projects/spring-web
6. https://reactjs.org/
7. https://nodejs.org/