

@doorz – A Multi-Vendor Website

By

MUHAMMAD ABUBAKAR SIDDIQUE

2022-GCUF-05733

NAILA SHAFIQUE

2022-GCUF-05736

SAUD AHMAD

2022-GCUF-05737

BACHELOR OF SCIENCE IN COMPUTER SCIENCE



**DEPARTMENT OF COMPUTER SCIENCE
Government College University Faisalabad**

2024

Acknowledgement

In the name of Allah, the Most Merciful and the Most Gracious, we are incredibly grateful to Allah Almighty for giving us the courage, perseverance, and knowledge necessary to start and finish this project. His immense knowledge and guidance have served as the cornerstone of our academic career and have been essential in helping us overcome the obstacles we have faced.

We are grateful to our supervisor, Dr. Uzma Jameel, for her constant support, eloquent advice, and invaluable comments throughout this endeavor. Her knowledge and support have been instrumental in helping us improve our work and ensure that we meet the highest requirements for academic qualifications. His guidance has improved our technical expertise and inspired us to drive innovation and ongoing development.

We would like to express our sincere gratitude to the instructors and staff of the Department of Computer Science, Government College University, Faisalabad, who provided us with the tools and supportive environment that helped us complete this project. His commitment to fostering a culture of research and learning has greatly aided our academic and personal growth.

We also want to express our sincere gratitude to our families, who have been our greatest source of support with their love, tolerance, and sacrifices. Their unwavering encouragement and support have served as the foundation around which our academic success has been built. This endeavor would never have been possible without their undying faith in our abilities.

We would especially like to thank our friends and co-workers who have inspired us and offered us invaluable advice and support during this endeavor. This journey has been memorable and enriching because of their friendship and spirit of cooperation.

Finally, we thank Allah for His blessings, including our good health, our ability to persevere, and the opportunity to learn and grow. We pray that He will continue to guide and help us in all that we do in the future.

Dedication

We dedicate this effort to Almighty Allah, the source of all wisdom and knowledge, whose immense mercies and blessings have made it possible for us to reach this important achievement. We have faced many difficulties along the way, but with His heavenly guidance, we have been able to overcome them and come out strong and determined.

Our success is largely due to the love, sacrifices, and perseverance of our parents, to whom we dedicate this effort. Our motivation comes from their unwavering support and belief in our abilities. This success is a testament to their commitment and the morals they have taught us. We are incredibly grateful to them for being our rock and for their endless tolerance and compassion.

We dedicate this effort to our siblings and extended family as a thank you for their unwavering encouragement and support. Their belief in us has motivated us to strive for excellence in all that we do and has been a source of inspiration.

This commitment also goes to our wonderful supervisor, Dr. Uzma Jameel, whose guidance was very helpful. His advice, tolerance, and insightful criticism have been invaluable to the success of our project. We sincerely appreciate the help and expertise that he has shared with us.

Additionally, we dedicate this initiative to our friends and co-workers who have supported us along the way. Their cooperation, support, and helpful criticism have been invaluable in bringing this idea to fruition. It has been a rewarding and enriching experience because of their friendship and support.

Finally, we dedicate this work to all the teachers, mentors, and friends who have helped us in our academic careers. His commitment to education and his love of learning have inspired us to work hard and honor our goals. This project is an expression of their impact on our lives and our commitment to preserve knowledge and creative heritage.

Table of Content

Chapter 1	7
Introduction to the Problem	7
1.1 Introduction	7
1.2 Background	7
1.3 Purpose	8
1.4 Scope	8
1.5 Objective	9
1.6 Intended Audience and Reading Suggestions	9
1.7 Documentation Conventions	10
Chapter 2:	12
Software Requirements Specification	12
2.1 Overall Description	12
2.1.1 Product Perspective	12
2.1.2 Product Features	13
2.1.3 Design and Implementation Constraints	17
2.1.4 Assumptions and Dependencies	18
2.2 System Features	20
2.2.1 System Feature 1: Vendor Management System	20
2.2.2 System Feature 2: Product Management System	20
2.2.3 System Feature 3: Customer Account Management	20
2.2.4 System Feature 4: Search and Navigation	21
2.2.5 System Feature 5: Shopping Cart and Checkout Process	21
2.2.6 System Feature 6: Payment System Integration	21
2.2.7 System Feature 7: Order Management and Fulfillment	22
2.2.8 System Feature 8: Review and Feedback Mechanism	22
2.2.9 System Feature 9: Administrative Control Panel	22
2.2.10 System Feature 10: Event Management System	23
2.2.11 System Feature 11: Communication and Messaging	23
2.2.12 System Feature 13: Inventory Management	23
2.2.13 System Feature 14: Shipping and Delivery Management	23
2.2.14 System Feature 15: User Activation and Verification	24
2.2.15 System Feature 16: Wishlist Management	24
2.3 External Interface Requirements	25
2.3.1 User Interfaces	25
2.3.2 Hardware Interfaces	25
2.3.3 Software Interfaces	25
2.3.4 Communications Interfaces	25
2.4 Other Nonfunctional Requirements	26
2.4.1 Performance Requirements	26
2.4.2 Safety Requirements	26
2.4.3 Security Requirements	26

List of Tables

No table of figures entries found.

List of Figures

No table of figures entries found.

Chapter 1

Introduction to the Problem

1.1 Introduction

Due to the rapid growth of the online industry, e-commerce platforms are now essential for both customer delight and business expansion. The need for a stable and dynamic multi-vendor e-commerce platform has never been greater. This need is met by the @doorz initiative, which creates a vast online marketplace aimed at promoting seamless communication between buyers and sellers. The project uses Redux for state management in conjunction with the MERN stack, which consists of MongoDB, Express.js, React, and Node.js. These technological advances guarantee that the platform is user-friendly, scalable, and efficient, meeting the demands of both modern suppliers and consumers.

@doorz aims to revolutionize the online shopping experience by providing a flexible and welcoming environment that fosters the growth of small and medium-sized enterprises (SMEs). It is not just another e-commerce platform. @doorz will bridge the gap between suppliers and buyers using innovative technology solutions and a simple and efficient marketplace that promotes company expansion and enhances customer happiness.

1.2 Background

E-commerce has emerged as a major sales channel as a result of the fundamental transformation of the retail industry as a result of the rapid spread of mobile and Internet technologies. Online marketplaces are gradually replacing or complementing traditional brick-and-mortar retailers as they provide unparalleled accessibility and convenience. As they can bring together a wide range of sellers and products, multi-vendor e-commerce platforms have become increasingly popular in this context, providing consumers with a one-stop shop for all their shopping needs.

However, there are often issues with scalability, security, and user experience with the multivendor platforms that now exist. Many small and medium-sized businesses (SMEs) find it difficult to build a solid online presence due to their expensive operating costs and lack of technical know-how. By offering SMEs an easy-to-use, scalable, and secure platform, @doorz aims to solve these problems while also improving the overall shopping experience for customers.

1.3 Purpose

The primary goal of the @doorz project is to use the MERN stack (MongoDB, Express, React, and Node.js) as well as Redux for state management to design and build a robust multi-vendor e-commerce platform. The platform strives to:

- **Empower Vendors:** Give SMEs the tools and assets they need to monitor their stores, monitor revenue, and grow their customer base with minimal overhead.
- **Improve the Customer experience:** Provide a secure checkout procedure, an intuitive user interface, and advanced search features to guarantee client happiness.
- **Leverage Advanced technologies:** Build a scalable, responsive application with real-time updates and the ability to handle large levels of traffic using the MERN stack.

1.4 Scope

The scope of the @doorz project includes building an e-commerce platform with multiple vendors that has the following features:

- **Vendor Dashboard:** Vendor dashboards are comprehensive control panels that allow vendors to monitor orders, inventory, and sales data.
- **Customer Accounts:** Personalized shopping experiences with wish lists, transaction histories, and order monitoring are available through customer accounts.
- **Product Listings:** Comprehensive product pages that include dynamic pricing options and multimedia support.
- **Search and Filters:** Advanced search features and filters are provided to help users find products quickly and efficiently.
- **Checkout and Shopping Cart:** Simple, secure procedures that help customers complete their transactions.
- **Payment Gateway Integration:** To enable smooth transactions, several secure payment methods are provided.
- **Order Management:** Reliable back-end systems that allow suppliers to handle orders, track status updates, and monitor shipping.

- **Review and Rating System:** Mechanisms for users to rate and review products and vendors, fostering community trust and engagement.
- **Responsive Design:** Ensuring compatibility with various devices, including desktops, laptops, tablets, and smartphones.

1.5 Objective

The objectives of the @doorz project are as follows:

- **To create an efficient, scalable, multi-vendor e-commerce platform** that facilitates the expansion of small and medium enterprises (SMEs) by offering a comprehensive, user-friendly environment for managing their online presence. does.
- **To improve the online shopping experience for consumers** by providing a secure transaction processing system, advanced search features, and an easy-to-use interface.
- **Strong security mechanisms** should be in place to protect user information and guarantee the integrity of transactions, instilling trust and reliability in the process.
- **To give suppliers access to comprehensive analytics tools** that will help them understand market trends, consumer behavior, and sales success. This will allow them to make data-driven decisions and improve their business.
- **Offering a long-term and innovative solution** for online sales to guarantee that the platform is flexible enough for future expansion as well as advancements in technology.

1.6 Intended Audience and Reading Suggestions

The target audience for this paper is broad and includes:

- **Project Supervisor:** Charged with monitoring and ensuring that the project meets technical and academic requirements.
- **Developers:** Developers are charged with implementing the platform's technical framework. They will discover the comprehensive technical specifications, schematics, and installation instructions necessary to build the platform.
- **Graphic Designers:** Graphic designers are tasked with creating user interfaces and user experiences. The document will be used by designers to understand the functional specifications and design principles that guide the creation of the platform's interactions and visual components.
- **Project Manager:** responsible for overall project coordination, resource allocation, risk management, and planning. The document will serve as the project manager's reference for comprehensive project schedules, resource requirements, and risk mitigation strategies.

Reading Suggestions:

- **Overview:** Provides all readers with a basic understanding of the project's purpose by providing a quick explanation of the project's vision, objectives, and impact
- **Project Description:** For most project managers and developers, this outlines goals, expected results, and techniques.
- **Design and Development:** Describes specific processes, tools, and design philosophies that developers and designers use.
- **Project Management:** For most project managers, this focuses on resource allocation, risk management, and project timeframes.
- **Testing and Quality Assurance:** For most developers and project managers, this section covers specific techniques for assuring software quality and reliability.

1.7 Documentation Conventions

This document follows specific conventions to ensure clarity, consistency, and professionalism. The conventions used throughout the document are as follows:

- **Font and formatting:** Times New Roman is used throughout the work, with body text set at 12 points, main headings set at 16 points, and subheadings set at 14 points. Section and subsection headings are distinguished by bolding and capitalization.

- **Lists:** Products, features, and benefits are presented using bulleted lists. Sequences of actions or systematic steps are represented using numbered lists.
- **Emphasis:** Key words and phrases are underlined in the text to draw attention to them. Instructions and important parts can also be underlined for emphasis.
- **Tables and Figures:** All tables and figures in the paper have a sequential numbering system and are properly referenced in the text. This guarantees that every visual aid can be referenced and easily identified.

Chapter 2:

Software Requirements Specification

2.1 Overall Description

The @doorz platform is a multi-vendor e-commerce solution designed to cater to the needs of both vendors and consumers. Unlike traditional e-commerce websites, @doorz allows multiple vendors to manage their storefronts, products, and sales under a single platform. This project leverages the MERN stack—comprising MongoDB, Express.js, React, and Node.js—along with Redux for state management. The choice of this technology stack ensures a scalable, flexible, and efficient online marketplace that can handle a high volume of transactions and user interactions seamlessly.

The platform's architecture is designed to facilitate easy integration with third-party services such as payment gateways and shipping providers, ensuring secure transactions and real-time tracking capabilities. Additionally, the responsive design ensures that the platform is accessible and user-friendly across various devices and screen sizes, from desktops to smartphones.

2.1.1 Product Perspective

The @doorz project seeks to provide an innovative, multi-vendor e-commerce platform that improves the user experience by leveraging contemporary web technologies and addressing the shortcomings of existing solutions. The platform will act as a dynamic marketplace where customers can have a seamless and secure shopping experience and vendors can manage their products, track sales, and interact with customers.

- **Market Positioning:** With an emphasis on scalability, security, and user-friendliness, @doorz aims to be a preeminent multi-vendor platform. Its target market includes small and medium-sized businesses (SMEs) as well as a wide range of consumers looking for a variety of products.
- **Stakeholder Participation:** Key stakeholders involved are:
 - **Vendors:** business owners who need a stable platform to manage and list their goods.
 - **Consumers:** end users looking for a safe and convenient way to shop online.
 - **Administrators:** platform managers in charge of managing user complaints, ensuring compliance, and monitoring operations.

- **Developers:** The technical group in charge of building and managing the platform.
- **Technical integration:** The platform guarantees a unified development process by leveraging the MERN stack. React provides a dynamic user interface, Express.js is a robust server framework, MongoDB provides a configurable database solution, and Node.js facilitates efficient server-side operations. Redux optimizes state management to guarantee a reliable and consistent user experience.
- **Scalability and adaptability:** The platform architecture is designed to withstand increasing traffic and data volumes. Its modular design allows for future improvements and the easy inclusion of new features.

2.1.2 Product Features

1. Vendor Dashboard:

A comprehensive system that enables vendors to manage their profiles, product listings, and orders efficiently.

- **Overview:** The vendor dashboard is a centralized control panel where vendors can manage their entire store. This includes inventory management, order processing, sales tracking, and customer interactions.
- **Sales Analytics:** Vendors can access detailed analytics, including sales reports, product performance metrics, customer demographics, and behavior patterns. This data helps vendors make informed decisions to optimize their sales strategies.
- **Order Management:** Vendors can view and manage all orders, from new orders to processing, shipping, and completed orders. The system supports order status updates, handling returns, and managing refunds.
- **Inventory Management:** Vendors can add new products, update existing product details, manage stock levels, and set dynamic pricing options. The system supports bulk product uploads through CSV files for ease of management.
- **Communication Tools:** Vendors have a built-in messaging system to communicate with customers directly, addressing their queries and resolving issues promptly.

2. Customer Accounts:

A secure and personalized space for customers to manage their information, orders, and interactions with vendors.

- **Registration and Profile Management:** Customers can create accounts, update personal information, and manage their profiles. The registration process includes email verification for added security.
- **Order Tracking:** Customers can track the status of their orders in real-time, from the moment they place an order until it is delivered.
- **Wishlist:** Customers can add products to their wishlist for future reference and easy access.
- **Reviews and Ratings:** Customers can rate and review products they have purchased, providing feedback to other customers and vendors. Reviews are moderated to ensure quality and relevance.

3. Product Listing:

A robust system that allows vendors to create, update, and manage their product listings with ease.

- **Multimedia Support:** Vendors can upload high-resolution images, videos, and detailed descriptions for each product. This includes support for 360-degree product views and augmented reality previews.
- **Dynamic Pricing:** Vendors can set regular prices, discounts, promotional prices, and schedule price changes in advance.
- **Categorization and Tags:** Products can be categorized into multiple categories and tagged with relevant keywords to enhance searchability and navigation.

4. Search:

An intuitive search engine with advanced options to help customers find products quickly and easily.

- **Advanced Search:** The search engine supports full-text search capabilities with real-time auto-complete suggestions based on user input.

5. Shopping Cart and Checkout:

A seamless process for customers to add products to their cart and complete purchases securely.

- **Cart Management:** Customers can add, remove, and modify products in their shopping cart. The cart is saved across sessions, allowing users to continue shopping at their convenience.
- **Multiple Checkout Steps:** The checkout process is divided into multiple steps: entering shipping information, selecting a payment method, and reviewing the order before final confirmation.
- **Payment Methods:** The platform integrates with multiple payment gateways, including Stripe for credit/debit card payments, PayPal, and Cash on Delivery (COD) for local purchases.
- **Order Confirmation:** Customers receive email notifications upon successful order placement, including order details and estimated delivery times.

6. Payment Gateway Integration

Secure and flexible payment gateway integration to facilitate various payment methods.

- **Support for Major Payment Gateways:** Integration with widely-used payment gateways such as Stripe, PayPal, and traditional banking APIs.
- **Encryption and Security Compliance:** All transactions are encrypted using SSL/TLS protocols to ensure data security and compliance with industry standards.
- **Real-time Payment Confirmation:** Immediate confirmation of payment status, allowing both customers and vendors to see real-time updates on transaction statuses.
- **Multiple Payment Options:** Support for various payment methods including credit/debit cards, PayPal, and COD.

7. Order Management and Fulfillment

Efficient back-end system for processing orders and managing fulfillment.

- **Order Receipt and Confirmation:** Automated systems generate and send order confirmations to customers and vendors.
- **Status Updates:** Real-time updates on order status including processing, shipping, and delivery. Notifications are sent at each stage.
- **Returns and Refunds:** Streamlined process for handling returns and refunds, including automated updates to stock levels and financial records.

8. Review and Feedback Mechanism

Allows users to leave reviews and ratings for products and vendors, fostering trust and community.

- **Product Reviews:** Customers can leave detailed reviews and ratings for products they have purchased. Reviews can include a rating system, written feedback, and the option to upload images.
- **Vendor Feedback:** Customers can provide feedback on their shopping experience with specific vendors. This helps build trust and credibility on the platform.

9. Responsive Design

Ensuring the platform is accessible and user-friendly across various devices and screen sizes.

- **Cross-Device Compatibility:** The platform is designed to work seamlessly on desktops, laptops, tablets, and smartphones, providing a consistent user experience across devices.
- **Adaptive UI/UX:** The user interface adapts to different screen sizes and resolutions, ensuring optimal usability regardless of the device used.
- **Touch-Friendly Interfaces:** The platform includes touch-friendly interfaces for mobile devices, making it easy to navigate and interact with on smartphones and tablets.

10. Administrative Control Panel

A central dashboard for administrators to manage the platform, including users, vendors, and content.

- **User and Vendor Management:** Tools for managing user and vendor accounts, including approval processes, role assignments, and access controls.
- **Content Moderation:** Systems for monitoring and moderating product listings, reviews, and user interactions to ensure compliance with platform policies.
- **System Configuration:** Settings for configuring platform-wide features, including payment options, shipping methods, and site policies.

2.1.3 Design and Implementation Constraints

Several limitations affect the design and implementation of the @doorz platform:

1. Technology Stack:

- The application is built using the MERN stack (MongoDB, Express.js, React, Node.js), which may restrict the use of certain third-party tools and libraries not compatible with these technologies.
- Redux is used for state management across the application to maintain a predictable and consistent state.

2. Deployment and Scalability:

- The initial deployment will focus on a specific market or region with plans to expand geographically in subsequent phases.
- The system must be designed to scale efficiently to handle increased traffic and data load without compromising performance.

3. Security and Compliance:

- Strict security measures need to be implemented to protect user data, including encryption for data at rest and in transit, and compliance with industry standards like GDPR.
- Role-based access control is essential to ensure users only have access to their permitted features and data.

4. Performance Requirements:

- The application should support a high number of concurrent users (e.g., 1,000 vendors and 10,000 customers) without performance degradation.
- Page load times should be optimized to ensure a smooth user experience, targeting load times of less than 3 seconds for key pages.

5. Cross-Platform Compatibility:

- The front-end must be responsive, providing a seamless user experience across various devices and screen sizes, including desktops, tablets, and smartphones.

6. Database Management:

- MongoDB is used for database management, requiring careful schema design to handle the hierarchical data structure efficiently.

7. Integration with Third-Party Services:

- The system relies on third-party services for payment processing (Stripe, PayPal) and shipping integrations, which necessitates reliable and secure API integrations.
- Any changes or outages in these third-party services could impact the platform's functionality.

8. User Interface and Experience:

- The UI should be intuitive and user-friendly to accommodate users with varying levels of technical proficiency.
- Consistency in design and navigation across the platform is crucial to enhance usability.

9. Testing and Quality Assurance:

- Comprehensive testing, including unit tests, integration tests, and user acceptance tests, must be conducted to ensure the reliability and quality of the application.
- Automated testing tools should be employed where possible to streamline the testing process.

10. Maintenance and Updates:

- The codebase should follow clean code principles and be well-documented to facilitate easy updates and maintenance.
- A version control system (e.g., Git) should be used to manage code changes and collaboration among developers.

2.1.4 Assumptions and Dependencies

1. User Familiarity:

- It is assumed that target users (vendors and customers) have a basic understanding of online shopping and e-commerce platforms.

2. Technical Infrastructure:

- The performance and scalability of the platform depend on the underlying cloud hosting services (e.g., AWS, Azure) used for deployment.
- Reliable internet connectivity is required for seamless operation and access to the platform.

3. Third-Party Service Availability:

- The platform's functionality depends on the availability and reliability of third-party services such as payment gateways (Stripe, PayPal) and shipping providers.

- Continuous updates and support from these services are assumed for uninterrupted platform operations.

4. Regulatory Compliance:

- The platform must comply with relevant data protection and privacy laws (e.g., GDPR), and it is assumed that necessary legal consultations will be undertaken to ensure compliance.
- Regular audits and updates may be required to stay compliant with changing regulations.

5. User Engagement:

- The platform's success relies on active engagement from both vendors and customers. It is assumed that there will be sufficient marketing and outreach efforts to attract and retain users.
- User feedback mechanisms will be in place to continuously improve the platform based on user needs and preferences.

6. Development Resources:

- The project assumes availability of skilled developers proficient in the MERN stack and Redux for both initial development and ongoing maintenance.
- Adequate resources (time, budget, personnel) will be allocated to meet project milestones and deadlines.

7. Scalability Planning:

- It is assumed that the initial architecture will be designed with scalability in mind, allowing for seamless expansion as user numbers grow.
- Future scalability plans will include provisions for additional server resources, database optimization, and load balancing solutions.

These sections address the necessary constraints and assumptions for the design and implementation of the multi-vendor platform, ensuring a robust, scalable, and user-friendly solution.

2.2 System Features

2.2.1 System Feature 1: Vendor Management System

Description: A comprehensive system that allows vendors to create and manage their profiles, products, and orders.

Features:

- **Vendor Registration and Login:** Vendors can create accounts, log in, and manage their profiles.
- **Dashboard:** Vendors have access to a dashboard to manage their inventory, track sales, and view analytics.
- **Product Management:** Vendors can add, update, and delete product listings with detailed descriptions, images, and pricing.
- **Order Management:** Vendors can view and manage orders, update order status, and handle returns and refunds.
- **Communication Tools:** Vendors can communicate with customers and site administrators via integrated messaging.

2.2.2 System Feature 2: Product Management System

Description: This feature allows vendors to create, update, and manage their product listings, ensuring accurate and detailed product information.

Features:

- **Product Listings:** Vendors can create product listings with comprehensive details, including titles, descriptions, images, and prices.
- **Pricing Options:** Support for dynamic pricing, discounts, and promotional offers.
- **Categorization:** Products can be categorized for easy navigation and searchability.
- **SEO Optimization:** Fields for meta titles, descriptions, and keywords to enhance search engine visibility.

2.2.3 System Feature 3: Customer Account Management

Description: A secure and personalized space for customers to manage their profiles, orders, and communications with vendors.

Features:

- **Customer Registration and Login:** Customers can create accounts, log in, and manage their profiles.

- **Order Tracking:** Customers can view their order history, track order status, and manage returns.
- **Wishlist:** Customers can save products to their wishlist for future purchases.
- **Product Reviews and Ratings:** Customers can leave reviews and ratings for purchased products.
- **Profile Management:** Customers can update their personal information, addresses, and payment methods.

2.2.4 System Feature 4: Search and Navigation

Description: An intuitive search engine that allows users to find products quickly using keywords, categories, and filters.

Features:

- **Keyword Search:** Users can search for products using keywords, with auto-complete suggestions.

2.2.5 System Feature 5: Shopping Cart and Checkout Process

Description: A seamless process that enables customers to review their selections, make changes, and complete purchases securely.

Features:

- **Add to Cart:** Users can add products to their shopping cart from product listings or detail pages.
- **Cart Summary:** A summary of the cart with options to edit quantities or remove items.
- **Checkout Process:** A multi-step checkout process including shipping details, payment options, and order review.
- **Payment Methods:** Support for multiple payment options including credit/debit cards, PayPal, and cash on delivery.
- **Order Confirmation:** Email notifications for order confirmation and updates.

2.2.6 System Feature 6: Payment System Integration

Description: A secure payment gateway system that supports various payment methods and ensures secure transaction processing.

Features:

- **Multiple Payment Gateways:** Integration with major payment gateways like Stripe and PayPal.

- **Transaction Security:** Encryption of payment data to ensure secure transactions.
- **Payment Confirmation:** Immediate confirmation of payments with detailed receipts sent to customers.

Refund Processing: Efficient processing of refunds for returned or canceled orders.

2.2.7 System Feature 7: Order Management and Fulfillment

Description: A backend system for vendors to process orders, update status, and manage fulfillment efficiently.

Features:

- **Order Processing:** Vendors can view new orders, update their status, and manage shipping details.
- **Shipping Integration:** Integration with shipping carriers for real-time tracking and shipping label generation.
- **Order History:** Detailed order history for vendors and customers, including tracking information and delivery status.
- **Return and Refund Management:** Tools for handling returns, exchanges, and refunds.

2.2.8 System Feature 8: Review and Feedback Mechanism

Description: A feature that allows users to leave reviews and ratings for products and vendors, fostering trust and community.

Features:

- **Product Reviews:** Customers can leave reviews and star ratings for products they have purchased.
- **Vendor Ratings:** Vendors can be rated based on their service quality and product offerings.
- **Feedback Moderation:** Tools for administrators to moderate and manage reviews and feedback.

2.2.9 System Feature 9: Administrative Control Panel

Description: A central dashboard for administrators to monitor the entire platform, manage users, and make site-wide changes.

Features:

- **User Management:** Tools to manage vendor and customer accounts, including approvals and suspensions.

- **Content Moderation:** Moderation of product listings, reviews, and other user-generated content.
- **System Configuration:** Settings for managing site-wide configurations, such as payment options and shipping settings.

2.2.10 System Feature 10: Event Management System

Description: A feature that allows vendors to create and manage promotional events and offers to attract customers.

Features:

- **Event Creation:** Vendors can create events and promotional offers for specific products or categories.
- **Event Scheduling:** Ability to schedule events for specific dates and times.
- **Event Management:** Tools to update, edit, or cancel events as needed.

2.2.11 System Feature 11: Communication and Messaging

Description: A robust communication system that allows vendors, customers, and administrators to interact seamlessly.

Features:

- **Vendor-Customer Messaging:** Direct messaging between vendors and customers for inquiries and support.
- **Support Tickets:** A ticketing system for customers to raise support issues, which vendors and admins can track and resolve.
- **Chat Integration:** Real-time chat support for immediate assistance.

2.2.12 System Feature 13: Inventory Management

Description: A feature for vendors to manage their inventory efficiently, ensuring products are always in stock.

Features:

- **Stock Levels:** Real-time tracking of product stock levels.

2.2.13 System Feature 14: Shipping and Delivery Management

Description: An integrated system to manage shipping and delivery processes, ensuring timely and accurate deliveries.

Features:

- **Shipping Options:** Multiple shipping options for customers, including standard and expedited shipping.

- **Shipping Label Generation:** Automatic generation of shipping labels based on carrier requirements.
- **Real-Time Tracking:** Integration with shipping carriers to provide real-time tracking information to customers and vendors.
- **Delivery Notifications:** Automated notifications for customers about shipping status and delivery updates.

2.2.14 System Feature 15: User Activation and Verification

Description: A feature to ensure the authenticity and security of user accounts through email verification and activation.

Features:

- **Email Verification:** Automated emails sent to users for account verification upon registration.
- **Activation Links:** Unique activation links for users to verify their email addresses and activate their accounts.

2.2.15 System Feature 16: Wishlist Management

Description: A convenient feature for customers to save products they are interested in purchasing later.

Features:

- **Add to Wishlist:** Customers can add products to their Wishlist directly from product listings or detail pages.
- **Manage Wishlist:** Tools for customers to view, edit, and remove items from their Wishlist.
- **Share Wishlist:** Option to share Wishlist with others via email or social media.

These system features comprehensively cover the functional aspects of the multi-vendor platform, ensuring a robust, user-friendly, and efficient e-commerce experience for vendors, customers, and administrators. The implementation of these features will leverage the strengths of the MERN stack and Redux to deliver a scalable and high-performing application.

2.3 External Interface Requirements

2.3.1 User Interfaces

The user interfaces are critical to ensuring a smooth and intuitive interaction with the system.

The main user interfaces include:

- **Vendor Interface:** A web-based dashboard allowing vendors to manage their storefronts, products, orders, and view analytics.
- **Customer Interface:** A user-friendly front-end for customers to browse products, manage their accounts, place orders, and track order status.
- **Admin Interface:** An administrative portal for site management, user and content moderation, and accessing various reports and analytics.

2.3.2 Hardware Interfaces

The system is designed to be compatible with various hardware interfaces to ensure accessibility and functionality across different devices:

- **Responsive Design:** Ensures compatibility with desktops, laptops, tablets, and smartphones.

2.3.3 Software Interfaces

The software interfaces define the interaction between different software components within the system and external services:

- **Database:** MongoDB is used for storing user data, product lists, orders, etc.
- **Server:** The backend application server is built using Express.js running on Node.js.
- **Client-side:** The front-end user interface is built using React with Redux for state management.
- **Payment Gateway:** Integration with external payment services such as PayPal, Stripe, and traditional banking APIs for secure transactions.

2.3.4 Communications Interfaces

Effective communication interfaces are necessary for smooth operations and user interactions:

- **Email System:** Integration with an email service to send order confirmations, newsletters, and marketing materials.
- **Social Media Integration:** Interfaces with social media platforms for login, sharing, and promotional purposes.
- **RESTful API:** Backend services are exposed through a RESTful API for integration with third-party applications and services.

2.4 Other Nonfunctional Requirements

2.4.1 Performance Requirements

Performance requirements ensure that the system operates efficiently under expected conditions:

- **Load Time:** Web pages should load within less than 3 seconds to ensure a smooth user experience.
- **Concurrent Users:** The system must support up to 1,000 vendors and 10,000 customers simultaneously without performance degradation.
- **Scalability:** The system architecture should support scaling to accommodate an increasing number of users and data.

2.4.2 Safety Requirements

Safety requirements are essential to protect data integrity and system stability:

Transaction Rollback: The ability to rollback transactions in case of failure to ensure data integrity.

2.4.3 Security Requirements

Security requirements ensure that the system protects user data and transactions from unauthorized access and breaches:

Authentication: Strong authentication mechanisms to prevent unauthorized access for all users.

Authorization: Role-based access control to ensure users can only perform tasks they are authorized to do.

Data Encryption: Use encryption for sensitive data both at rest and in transit using industry-standard methods.