SCHOOL OF TECHNOLOGY

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

KCAVIBES E-COMMERCE WEB APPLICATION

BY

AWANZI HASSAN MUNENE (21/04609)

PRESENTED TO: Prof. PATRIC OGAO

SYSTEM REQUIREMENT SPECIFICATION DOCUMENT

Contents

1. INTRODUCTION

In the ever-evolving landscape of technology, e-commerce businesses are harnessing the power of the latest advancements to enhance their online marketplace services. As a response to these changes, we propose the development of a sophisticated Multi-vendor e-commerce web application that will revolutionize and streamline the operations of our platform.

Historically, our platform has relied on conventional e-commerce methods. However, the current technological shifts necessitate the integration of a cutting-edge system, addressing the challenges posed by the traditional approach.

The foremost challenge ecommece platforms face now days is the lack of effective vendor onboarding poor user experiences.

These challenges have proven to be hindering service delivery, and thus making the adoption of a multi-vendor ecommerce platform imperative.

2. PURPOSE

This E-commerce Platform is designed to facilitate the operations and transactions of multiple vendors and customers in an online marketplace. This platform serves the following core functions:

Integration with Existing Systems: The platform will seamlessly integrate with existing e-commerce systems, enabling vendors to transition smoothly to the online marketplace.

Inventory Management: The system will automatically generate purchase orders for items out of stock, such as products from different vendors. Vendors will have the capability to manage their stock efficiently.

Customer Traffic Tracking: The platform will track and provide insights into the number of customers visiting the marketplace on a daily basis, minimizing data redundancy and facilitating data-driven decisions.

Records Management: The system will maintain secure and organized records for customers, vendors, products, and transactions, ensuring data integrity and accessibility.

User Management: The platform will handle user registration, providing options for both customers and vendors to create accounts.

AIMS OF SRS

The System Requirements Specification (SRS) for the Platform serves several important aims to ensure the successful development and management of the system. These aims include:

Reduce Development Effort:

The SRS forces a comprehensive and rigorous consideration of all requirements before the development phase begins. This minimizes the need for later redesign, recoding, and retesting.By carefully reviewing the requirements in the SRS, it helps identify omissions, misunderstandings, and inconsistencies early in the development cycle, allowing for more straightforward corrections.

Provide a Basis for Estimating Costs and Schedules:

The detailed product description in the SRS serves as a realistic foundation for estimating project costs.It can be used to obtain approvals for bids, pricing estimates, and resource allocations.

Provide a Baseline for Validation and Verification:

The SRS facilitates the development of effective validation and verification plans. It ensures that the final product aligns with the specified requirements.

It acts as a contract between stakeholders and developers, providing a baseline against which compliance can be measured.

Facilitate Transfer:

The SRS simplifies the process of transferring the software product to new users or different computing environments.

Owners find it more manageable to transition the software to other parts of their organization, ensuring a smooth transfer process.

Serve as a Basis for Enhancement:

The SRS, which focuses on the product itself rather than the development project, serves as a foundation for future enhancements.

While the SRS may require updates or modifications, it provides a stable reference point for ongoing product evaluation and improvements.

The aims of this SRS document are crucial for maintaining the efficiency, quality, and adaptability of the Multi-Vendor E-commerce Platform throughout its development lifecycle and beyond.

3. SCOPE

The scope of this E-commerce Platform encompasses its coverage, objectives, and key functionalities. This platform is designed to serve multiple vendors and customers within the e-commerce domain. Its scope includes:

Vendor Management: The platform will enable vendors to register, create and manage their online stores, add and update product listings, view and fulfill customer orders, and monitor their sales performance.

Customer Experience: Customers will be able to browse a wide range of products, search for specific items, add products to their cart, make secure payments, track order statuses, and provide ratings and reviews for products and vendors.

Administrator Functions: The platform will provide administrators with tools to manage user accounts, oversee the platform's overall operation, and enforce quality and policy standards.

Order and Inventory Management: Both vendors and administrators can efficiently manage orders, track order history, update inventory, and receive notifications for low stock levels.

Secure Transactions: The system will incorporate secure payment processing methods to ensure safe transactions and customer data protection.

Rating and Review System: Customers can provide feedback on products and vendors through a rating and review system, which will help maintain quality standards on the platform.

Reporting and Analytics: Vendors and administrators will have access to data and analytics to monitor sales, customer behavior, and the performance of the platform as a whole.

4. INTENDED AUDIENCE

The KcaVibes E-commerce Platform's intended audience includes various stakeholders who will interact with the system at different stages of its lifecycle:

Platform Managers: The managers responsible for overseeing the platform's day-to-day operations. They will benefit from the system's efficient administration tools and performance monitoring capabilities.

Vendors**:** These are the individuals or businesses that use the platform to sell their products. Vendors will utilize the platform for product listings, order management, and sales tracking.

Customers: The end-users who shop on the platform. Customers will interact with the platform to browse products, place orders, and provide feedback through reviews and ratings.

System Designers: These professionals are tasked with designing and translating the system requirements into logical designs for the platform. The SRS will serve as the basis for creating these designs.

System Developers: The developers responsible for coding and building the platform. They will rely on this document to ensure that the system they create aligns with the specified requirements.

System Implementers and Testers: Implementers will be responsible for deploying the system, and testers will validate its functionality using predefined test plans to guarantee that user requirements are met. Their work ensures a successful platform launch.

5. OVERALL DESCRIPTION

This E-commerce Platform serves as a standalone system designed to facilitate the operations of a multi-vendor e-commerce marketplace. Unlike the existing systems, this platform is independent and not interfaced with any other system. The e-commerce platform is a novel solution intended to overcome the limitations of manual vendor management and enhance the efficiency and effectiveness of multi-vendor e-commerce services.

6. PRODUCT FEATURES

Vendor Registration and Product Listing:

Vendors will have the capability to register on the platform and list their products.

Product Management:

The system will offer automated product management, allowing vendors to update, edit, and remove their product listings.

Order Processing and Fulfillment:

The platform will process customer orders and support order fulfillment for vendors.

Secure Payment Processing:

The system will provide secure payment processing, allowing customers to make purchases with confidence.

Customer and Vendor Ratings and Reviews:

Users will be able to rate and review both vendors and products, promoting transparency and trust within the platform.

Search and Filter Functionality:

The platform will enable users to search for products and filter results based on various criteria.

Shopping Cart and Checkout:

Customers can add products to their shopping carts, review their selections, and complete the checkout process.

User Account Management:

Users can create accounts, manage their profiles, track orders, and view order histories.

Reporting and Analytics:

The system will provide vendors with insights into their sales performance and customers' preferences through reporting and analytics features.

User Classes and Characteristics:

The users of the platform vary in their roles and familiarity with the system. User classes and their characteristics include:

Platform Administrators:

These are the super administrators responsible for managing the entire system. They have the authority to create and configure user accounts, manage system settings, and ensure that the platform operates smoothly. They monitor system health and resolve issues.

Vendors:

Vendors are the primary sellers on the platform. They require vendor-specific accounts to list and manage their products. They must be trained on the platform's vendor management features.

Customers:

Customers are end-users who visit the platform to browse and purchase products. They create customer accounts, manage their profiles, and engage in online shopping. User guides and user-friendly interfaces are provided to assist them, but generally, they are expected to have basic IT literacy.

Design and Development Team:

System designers and developers are responsible for designing the platform's features and functionalities. They are involved in translating the software requirements into logical designs and then into working systems.

System Implementers and Testers:

These users are responsible for the actual implementation of the system and the validation of its functionality through test plans. They ensure that

the platform aligns with user requirements and resolves any issues that arise during testing.

7. OPERATING ENVIRONMENT

The KcaVibes e-commerce Web Application is designed to operate in a specific environment that includes the hardware platform, operating systems, and other software components. This section outlines the requirements for the operating environment in which the application must seamlessly function.

Hardware Platform:

The e-commerce web application is expected to operate on standard hardware configurations commonly used in the industry. The platform must be compatible with the following hardware specifications:

- Web Servers**:** The application requires a web server that can handle HTTP requests and responses efficiently. Commonly used web server software such as Apache, Nginx, is preferred.
- Database Servers**:** The system relies on a MongoDB for data storage and retrieval. It should be compatible with widely adopted database servers, Microsoft SQL Server, or Oracle Database.

Operating System and Versions:

The application must be compatible with the following operating systems and their respective versions:

- Server Operating Systems**:** The platform should be deployable on major server operating systems, including but not limited to:

  Windows Server (versions 2016, 2019, and later)

  Linux distributions (e.g., Ubuntu, CentOS)

  Unix-based systems (e.g., FreeBSD)

- Client Operating Systems**:** End-users will access the application from various client devices, including Windows, macOS, Linux, and mobile operating systems (iOS and Android). Therefore, the application should be accessible from common web browsers and operating systems used by customers.

8. SOFTWARE COMPONENTS AND APPLICATIONS

The e-commerce web application is designed to peacefully coexist with several software components and applications. These include:

Web Browsers: Users will interact with the application through popular web browsers, including but not limited to Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge, and other modern browsers. The application's user interface must be responsive and compatible with these browsers.

External Services and APIs: The platform may integrate external services and APIs for various functions, such as payment processing, shipping services, and analytics. The specific APIs to be integrated will be determined during the development and integration phases to ensure compatibility and security.

Payment Gateways: To facilitate secure payment processing, the application must be capable of integrating with a variety of payment gateway providers that comply with industry standards for secure online transactions.

The above operating environment requirements are essential to ensure the smooth operation and compatibility of the Multi-Vendor E-commerce Web Application with commonly used hardware and software components. Compliance with these operating environment specifications will help guarantee a seamless user experience and reliable system performance.

9. DESIGN AND IMPLEMENTATION CONSTRAINTS

Corporate Policies and Regulatory Requirements:The development and operation of the e-commerce platform must comply with relevant corporate policies and industry-specific regulations. This may include data protection regulations (e.g., GDPR), tax laws, and industry-specific standards. Deviations from these policies and regulations can lead to legal consequences.

Hardware Limitations:The platform must adhere to specific hardware limitations to ensure optimal performance. For example, it should operate efficiently within defined timing requirements and memory constraints to prevent performance degradation.

Integration with Third-Party Applications:The platform may need to interface with third-party applications, such as payment gateways, shipping services, or analytics tools. Compatibility and integration with these external systems are necessary constraints. Any changes or updates to these third-party systems may impact the platform's functionality.

Choice of Technologies, Tools, and Databases:The selection of technologies, tools, and databases should align with predefined constraints. For example, if the organization has standardized on specific programming languages, databases, or development tools, the development team must adhere to these choices.

Parallel Operations:The system should be capable of handling concurrent operations and multi-user interactions. This constraint may affect the architectural design and database management to prevent conflicts and data consistency issues during simultaneous user actions.

Language Requirements:If the organization has a language preference for system development, it should be considered a constraint. For example, the organization may require that the platform's user interface and documentation be available in multiple languages to support a diverse customer base.

Communications Protocols:The platform must adhere to specific communication protocols for secure data exchange. This may involve using industry-standard encryption methods, data transfer protocols, and secure network communication to protect sensitive information and maintain data integrity.

Security Considerations:Security cnstraints are of paramount importance. The system should comply with established security policies, including data encryption, user authentication, authorization mechanisms, and protection against common security threats (e.g., SQL injection, cross-site scripting). It should also be designed to undergo periodic security assessments and audits.

Design Conventions and Programming Standards:The development team should adhere to specific design conventions and programming standards. These may include coding guidelines, naming conventions, code documentation practices, and architectural patterns defined by the organization or industry best practices.

Maintenance Responsibilities:The organization's policies may specify that it is responsible for maintaining the delivered software. This constraint may affect the choice of technologies, tools, and platforms used in development, as they should be manageable and maintainable by the organization's IT team.

10. USER DOCUMENTATION

User Manual:

A comprehensive user manual will be provided to guide both vendors and customers on how to use the platform. It will include step-by-step instructions, illustrations, and explanations of various features and functionalities.

- Online Help:

The application will feature an online help system accessible directly from the platform's user interface. Users can access context-specific help and guidance while using the system.

- Tutorials:

  Interactive tutorials will be available to walk users through specific tasks or processes, helping them understand how to perform common actions on the platform. These tutorials will be available in video or interactive formats.

- Frequently Asked Questions **(FAQ):**

A frequently asked questions section will be included in the user documentation to address common queries and concerns. It will provide quick answers to common issues that users may encounter.

- Troubleshooting Guide:

In case users encounter technical issues or errors, a troubleshooting guide will be available to help them diagnose and resolve these problems. It will offer solutions to common technical challenges.

- Glossary of Terms:A glossary will be included to define key terms and phrases used within the application. This will help users understand the platform's terminology and features.
- Accessibility Documentation: If the application is designed to be accessible to users with disabilities, documentation on how to use accessibility features will be provided.

11. ASSUMPTIONS FOR THE E-COMMERCE WEB APPLICATION

Internet Connectivity: It is assumed that users will have stable internet connectivity to access and use the application. Any disruptions in internet connectivity may affect the platform's usability.
Security Compliance: The application assumes that third-party payment gateways and external services used for payment processing and analytics comply with industry-standard security measures to protect customer data.
User Device Compatibility: It is assumed that users' devices, including computers, smartphones, and tablets, meet the minimum system requirements and have compatible web browsers for accessing the application.
Availability of External APIs: The successful integration of external services and APIs is dependent on the availability and proper functioning of those services. Any downtime or changes to these external services may impact the application's functionality.
Regulatory Compliance: The application assumes that it will operate within the regulatory requirements and policies governing e-commerce and online transactions, without any significant changes or constraints.

12. DEPENDENCIES FOR THE E-COMMERCE WEB APPLICATION:

Payment Gateways: The application depends on third-party payment gateways for secure payment processing. Any changes or issues with these payment gateways may affect payment processing on the platform.
External APIs: The application relies on external services and APIs for features like payment processing, shipping services, and analytics. The availability, reliability, and updates to these external APIs are external dependencies.
Server Hosting: The hosting environment, whether on a cloud server or in-house server, is a critical dependency. Any server-related issues, downtimes, or infrastructure changes can impact the availability and performance of the application.

Database Management System: The application depends on a MongoDB for data storage. Any changes or issues with the MongoDB can affect data integrity and access.
Client Devices: The application depends on users' devices for access. Any changes or constraints on client devices, including web browsers, could impact user experience.
Compliance with Security Standards: The application relies on security measures for data protection. Any changes in security standards or regulations may require updates to the platform.

13. EXTERNAL INTERFACE REQUIREMENTS

User Interfaces:

The Multi-Vendor E-commerce Platform features a user-friendly and intuitive interface, catering to the needs of different user classes:

Vendor Dashboard: Vendors will access a dedicated dashboard to register, manage products, process orders, and review their performance within the platform.

Customer Portal: Customers interact with a user-friendly portal to explore, order, and manage products. This portal includes a shopping cart and checkout process.

Platform Administrator Control Panel: Platform administrators use a control panel to manage the system, user accounts, and oversee its smooth operation. They also have access to reporting and analytics features.

Reporting and Analytics Dashboard: Vendors can access a dashboard to analyze sales performance and customer preferences, making informed decisions.

14. Hardware Interfaces

The Multi-Vendor E-commerce Platform operates as a web-based application, making it compatible with various hardware configurations. It requires internet connectivity and a web browser for access. There are no specific hardware dependencies.

15. Software Interfaces

The platform interfaces with various software components:

Web Browsers: Users interact with the platform via web browsers such as Chrome, Firefox, Safari, and Edge.

Payment Gateway: Secure payment processing is facilitated through an industry-compliant payment gateway.

16. COMMUNICATION INTERFACES

The Multi-Vendor E-commerce Platform communicates through standard communication protocols:

Hypertext Transfer Protocol (HTTP**)**: The primary protocol for data exchange between clients and the platform's web servers.

Secure Sockets Layer (SSL)/Transport Layer Security (TLS**)**: Employed for secure data transmission, ensuring the confidentiality and integrity of sensitive information.

Email: The platform can send automated emails for various purposes, including order user registration confirmations and notifications.

17. External APIs

The platform may integrate external services and APIs to enhance features such as payment processing, shipping services, and analytics. The selection of specific APIs will be determined during the development and integration phases, ensuring compatibility and security.

The Multi-Vendor E-commerce Platform is designed for seamless operation within its external environment, providing users with efficient communication and compatibility, regardless of their hardware or software preferences. These interfaces are structured to maintain clarity, security, and reliability throughout the platform's lifecycle.

18. OTHER NON-FUNCTIONAL REQUIREMENTS

Performance Requirements:

Performance requirements are crucial to ensure that the Multi-Vendor E-commerce Web Application meets user expectations and operates efficiently. These requirements help guide the development and design choices to provide a responsive and reliable user experience. The following performance requirements are specified:

Response Time:

- *Requirement:* The application should have an average response time for page loading of 2 seconds or less.
- *Rationale*: A fast response time ensures that users can quickly browse products and place orders, enhancing the user experience and potentially increasing sales.

Scalability:

- *Requirement*: The platform must support a minimum of 10,000 simultaneous users without significant degradation in performance.
- *Rationale*: Scalability is important to handle increased user loads during peak shopping periods, preventing slowdowns or downtime.

Database Query Response Time:

- *Requirement*: Database queries should execute in under 200 milliseconds on average.
- *Rationale*: Efficient database query response times are essential for seamless product searches and inventory management.

Concurrent Order Processing:

- **Requirement:** The system must process a minimum of 500 customer orders per minute without errors.
- **Rationale:** Efficient order processing is vital to prevent order bottlenecks during high-traffic periods.

Image Loading Time:

- **Requirement:** Product images must load within 1 second on product detail pages.
- **Rationale:** Fast image loading is essential for a visually appealing shopping experience and reducing bounce rates.

Safety Requirements:

While the Multi-Vendor E-commerce Web Application is not directly associated with physical safety concerns, there are safety requirements related to data and user interactions:

Data Safety:

- **Requirement:** The application must implement regular data backups to prevent data loss in case of unexpected incidents.
- **Rationale:** Data safety is crucial to prevent the loss of critical information, ensuring business continuity.

Security Requirements:

Security requirements are paramount to protect user data, privacy, and prevent unauthorized access. The Multi-Vendor E-commerce Web Application must comply with security standards and regulations:

User Authentication:

- **Requirement:** User authentication must include strong password policies, including the use of a combination of upper and lower-case letters, numbers, and special characters.
- **Rationale:** Strong user authentication helps prevent unauthorized access to accounts and ensures data security.

Data Encryption:

- **Requirement:** User data and payment information must be encrypted during transmission and storage using industry-standard encryption protocols such as SSL/TLS.
- **Rationale:** Data encryption is crucial to protect sensitive information from interception and unauthorized access.

Regulatory Compliance:

- **Requirement:** The application must adhere to relevant data protection regulations, such as GDPR (General Data Protection Regulation) and PCI DSS (Payment Card Industry Data Security Standard).
- **Rationale:** Compliance with data protection regulations is essential to maintain user trust and prevent legal repercussions.

Session Management:

- **Requirement:** Implement secure session management to protect user sessions from session hijacking and session fixation attacks.

- **Rationale:** Secure session management ensures that users remain authenticated and protected during their interactions with the platform.

19. SYSTEM FEATURES

The e-commerce platform includes a range of essential features that enhance its functionality and user-friendliness. These features are designed to streamline the shopping and selling experience for both vendors and customers. They are presented in the form of use cases.

Use Case: **User Login**

Primary Actor: User

Stakeholders: Database

Description and Priority: This use case describes the process of a user logging into the system. It is of high priority as it is the initial interaction point for users.

Stimulus/Response Sequences:

The user initiates the login process.

The user provides login credentials (username and password).

The database validates the provided credentials.

If the credentials are valid, the application server uploads the main system interface to the user.

If the credentials are invalid, an error message is displayed.

Functional Requirements**:**

REQ-1: **Application server services**: To execute this use case, the user must have access to the application server services to successfully access the database.

REQ-2: **LAN services**: The user needs Local Area Network (LAN) connectivity to establish a connection between the workstation, application server, and database server.

Use Case: **User Adds New Entry**

Primary Actor: User

Stakeholders: Database

Description and Priority: This use case involves a user registering a product entry in the system, adding a new transaction to the database.

Stimulus/Response Sequences**:**

The user connects to the system and accesses the product details form.

The user initiates the process by clicking the "search" button.

If it's new product, the user selects the "add product" option.

The user fills in the required details.

The user clicks the "save" button.

The application server checks if all required fields contain data.

If all required fields contain data, the application server adds the data to the database.

If any required field is empty, the application server returns the form to the user with a message indicating missing information.

Functional Requirements:

REQ-1: Application server services: To execute this use case, the user must have access to the application server services to successfully access the database.

REQ-2: LAN services: The user requires Local Area Network (LAN) connectivity to establish a connection among the workstation, application server, and database server.

Use Case**: Update Entry**

Primary Actor: User

Stakeholders: Database

Description and Priority: This use case describes a user updating an existing entry in the database.

Stimulus/Response Sequences**:**

The user connects to the system and accesses the services.

The user selects the "Update Entry" option.

The application server presents a form to the user.

The user selects the drop-down button.

The application server retrieves and displays all product details.

The user selects the correct product name.

The user updates the product details and clicks the "update" button.

Functional Requirements:

REQ-1: Application server services: To execute this use case, the user must have access to the application server services to successfully access the database.

REQ-2: LAN services: The user needs Local Area Network (LAN) connectivity to establish a connection among the workstation, application server, and database server.

Use Case: **Search for a product**.

Primary Actor: User

Stakeholders: Database

Description and Priority: This use case involves a user searching for specific product information in the database. It has high priority, especially for product information.

Stimulus/Response Sequences:

The user initiates a search for a specific product.

The application server queries the database for the requested information.

The application server returns the requested product details.

If the requested member is not found in the database, the application server displays a message indicating that the requested member cannot be found.

Requirements

Functional Requirements

Functional requirements describe the specific functions and capabilities that the system must provide to fulfill its intended purpose. For the Multi-Vendor E-commerce Platform:

The system must facilitate the registration and management of vendors and their products.

Vendors should be able to list their products and include essential information like product names, descriptions, prices, and images.

Customers must have the ability to browse products, add items to their shopping carts, and complete the checkout process.

The platform must manage and process customer orders, including updating stock levels and order statuses.

Secure payment processing is crucial, and the system must ensure the confidentiality and integrity of transactions.

Users (both vendors and customers) should be able to rate and review vendors and products.

Search and filter functionality must be provided to allow users to search for products and filter results based on various criteria.

User account management features should enable users to create, update, and manage their profiles.

Reporting and analytics features must be available to vendors for insights into sales performance and

Non-Functional Requirements**:**

Non-functional requirements encompass characteristics and constraints on the system's services or functions:

Efficiency: The platform must efficiently utilize resources to ensure fast and responsive performance, even during peak usage times.

Usability: The user interface should be intuitive and easy to understand, ensuring that users can effectively navigate and utilize the system.

Portability: The system should be designed to operate on various environments, providing flexibility in its deployment.

Timeliness: The system must ensure timely product delivery and order processing to meet user expectations.

Software Quality: The platform should be of high quality, free of significant bugs, and satisfy all customer requirements.

Maintainability: It should be easy to maintain and update, allowing for ongoing improvements and adjustments.

Back-Up: The system should provide back-up capabilities to safeguard data and maintain data integrity.

Security: The system should employ secure authentication and authorization mechanisms to protect user data and ensure confidentiality and integrity.

20. External Interface Requirements

User Interface:

The Multi-Vendor E-commerce Platform must feature an intuitive and user-friendly interface for both vendors and customers. It should allow for easy navigation, efficient use, and a smooth shopping and selling experience.

Hardware Interfaces:

The system should be compatible with standard computer components, such as keyboards, monitors, hard disks, memory, processors, and printers, ensuring ease of use on various hardware configurations.

Software Interfaces:

The platform must interface with different software components:

Web Browsers: Users will interact with the platform through common web browsers like Chrome, Firefox, Safari, and Edge.

MongoDB: The system relies on an mongodb for data storage and retrieval, adhering to industry standards.

Payment Gateway: Secure payment processing is facilitated through an industry-compliant payment gateway.

Security Requirements:

The system should validate user access through secure authentication mechanisms involving username and password verification to enhance data security and protect user privacy.

Software Benefits:

Flexibility: The system will allow for future upgrades and adapt to new ventures and changes in the business environment.

Security: It will protect sensitive data and information, ensuring confidentiality.

Maintainability: The system will be easy to maintain, reducing financial constraints.

Portability: Designed to be compatible with different Windows operating system versions.

Reliability: The system must meet needs consistently and provide reliable operations.

The need for the Multi-Vendor E-commerce Platform arises from the desire to enhance. This system will help streamline processes, reduce errors, automate manual operations, and enhance security and reliability.

21. Appendix

This section provides definitions and explanations of key terms, acronyms, and abbreviations used throughout this Software Requirements Specification for the Multi-Vendor E-commerce Platform:

SRS - Software Requirement Specification: The comprehensive document that outlines the functional and non-functional requirements of a software system.

LAN - Local Area Network: A network that connects computers and devices within a limited geographic area, typically within a single building or campus.

HTTP: Hypertext Transfer Protocol

SSL: Secure Sockets Layer

TLS: Transport Layer Security

API: Application Programming Interface

These definitions, acronyms, and abbreviations are provided to ensure clarity and a shared understanding of terminology used in this document.