



An E-commerce Platform With Smart Features

Submitted By

Md.Towfique Hasan Jayead

Student Id: 202-35-3120

Supervised By

Dr. Engr. Abdul Kadar Muhammad Masum

Professor

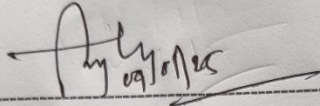
This project report has been submitted in fulfilment of the requirements for the degree of **Bachelor of Science in Software Engineering**

@ All right Reserved by Daffodil Internation University

APPROVAL

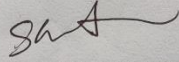
This project titled on "An E-commerce Platform with Smart Feature", submitted by **Md.Towfique Hasan Jayead (ID: 202-35-3120)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



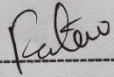
Chairman

Professor Dr. Engr. AKM Masum
Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



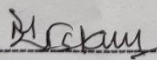
Internal Examiner 1

Md. Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



Internal Examiner 2

Dr. Marzia Ahmed
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

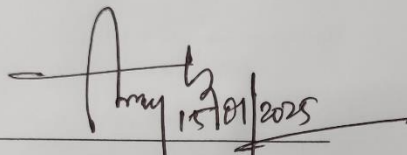


External Examiner

Dr. Md. Monowarul Islam
Associate Professor
Department of Computer Science & Engineering
Jagannath University

SUPERVISOR'S DECLARATION

I hereby declare that I have checked the named "An E-commerce Platform With Smart Features" project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Science.

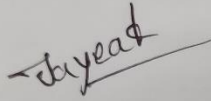


(Supervisor's Signature)

Full Name : Abdul Kadar Muhammad Masum, PhD
Position : Professor
Date : 31/12/2024

STUDENT'S DECLARATION

I hereby declare that the work in this project named “**An E-commerce Platform With Smart Features**” is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Daffodil International University or any other institution.



(Student's Signature)

Full Name : **Md. Towfique Hasan Jayead**

ID Number : 202-35-3120

Date : 31 December 2024

An E-commerce Platform With Smart Features

Md.Towfique Hasan Jayead

Project submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Science

Department of Software Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

OCTOBER 2024

ACKNOWLEDGEMENTS

I wish to convey my sincere appreciation to everybody who contributed to the successful execution of this Multi-Vendor E-Commerce Platform project.

I express my profound gratitude to my supervisor, Dr. Engr. Abdul Kadar Muhammad Masum, Professor in the Department of Software Engineering at Daffodil International University, for his important direction, support, and constructive input during the project's progress. Their experience and insights were important in defining the scope and implementation of this project.

I express my profound gratitude to the instructors and staff of the Software Engineering department at Daffodil International University for fostering a conducive learning atmosphere and supplying the essential resources for this effort. Their commitment to scholarly distinction and creativity has consistently served as a source of motivation.

I wish to express my gratitude for the steadfast support of my family, whose encouragement and confidence in my capabilities have been a motivating factor throughout my academic pursuits. Their patience and comprehension during the arduous stages of this undertaking were indispensable.

I would want to extend my appreciation to all developers, researchers, and contributors in e-commerce, web development, and software engineering. The open-source tools, frameworks, and expertise provided by the developer community constituted a robust foundation for this project.

I appreciate your support, guidance, and inspiration, which have rendered this endeavor feasible.

DEDICATION

I therefore declare that I have done this project under the oversight of “**Dr. Engr. Abdul Kadar Muhammad Masum**”, “**Professor**”, Department of Software Engineering, Daffodil International University. Also declare that neither entire record nor any portion of this record has been submitted somewhere else for my degree.

ABSTRACT

The Multi-Vendor E-Commerce Platform is a comprehensive web application intended to deliver an integrated solution for buyers, merchants, and administrators, facilitating a seamless and scalable online marketplace. This project seeks to optimize e-commerce operations by allowing different sellers to oversee their products, inventories, and orders, while providing customers with an extensive shopping experience.

The platform utilizes contemporary web technologies, including HTML, CSS, JavaScript, and React for the frontend, with Node.js and Express for the backend, to provide a responsive and user-friendly experience. A robust and efficient database, utilizing MongoDB, guarantees dependable data storage and retrieval for various e-commerce operations.

The platform includes essential functionality such as user registration and authentication, product browsing and filtering, a shopping cart, secure checkout and payment integration, order tracking, and ratings and reviews. Vendors utilize technologies for inventory management, order processing, and analytics, whereas administrators can monitor user activity, categories, reviews, and produce comprehensive reports to uphold platform integrity and efficiency.

The project, developed under Agile approach, emphasizes iterative enhancements and ongoing input, facilitating prompt problem resolution and alignment with user expectations. The establishment of secure payment gateways, strong authentication systems, and performance enhancement highlights the platform's dedication to security, dependability, and scalability.

The Multi-Vendor E-Commerce Platform is engineered to accommodate the dynamic requirements of e-commerce, with prospective improvements including mobile application integration, sophisticated customisation, and AI-driven recommendations. This project signifies a substantial advancement in establishing a dynamic and user-focused marketplace that empowers sellers and improves the shopping experience for consumers.

TABLE OF CONTENT

**DECLARATION
TITLE PAGE**

ACKNOWLEDGEMENTS **VI**

DEDICATION **VII**

ABSTRACT **VIII**

TABLE OF CONTENT **ixii**

LIST OF TABLES **xii**

LIST OF FIGURES **xii**

LIST OF APPENDICES **xiv**

CHAPTER 1 INTRODUCTION **15**

1.1 Background **12**

1.1.1 Context and Relevance **12**

1.1.2 Problem Identification **12**

1.1.3 Purpose and Justification **12**

1.1.4 Scope **13**

1.2 Project Planning and Initiation **13**

Feasibility Study (Step-by-Step) **13**

1.3 Target User Profile and Tentative Elicitation Process **15**

1.3.1 Target User **15**

1.3.2 User profile **15**

1.3.3 Elicitation Process **17**

1.4 Project Block Diagram **20**

1.5 System Requirements	21
1.5.1 Hardware Requirements	21
1.5.2 Software Requirements	21
1.5.3 Constraints and Dependencies	22
1.6 Project Scheduling	22
1.7 Summary	25
CHAPTER 2 DESIGN AND IMPLEMENTATION	26
2.1 Introduction	26
2.2 Functional Requirements	26
2.3 Non-Functional Requirements	29
2.3.1 Performance	30
2.3.2 Reliability	30
2.3.3 Portability	30
2.4 Object-oriented System design using UML	33
2.4.1 Use Case Diagram	33
2.4.2 Case Description	34
2.4.3 Activity Diagram	52
2.4.4 Sequence Diagram	68
2.4.5 Class Diagram	77
2.4.6 ER Diagram	78
2.5 Coding: Appendix A	78
2.6 Summary	78
CHAPTER 3 SOFTWARE TESTING	79
3.1 Introduction	79
3.2 Testing Features	79

3.2.1 Feature to Be Tested	79
3.3 Testing Strategies	80
3.3.1 Test Approach	80
3.3.2 Pass/Fail Criteria	80
3.4 System Testing (Test Cases with Report)	80
3.5 Summary	82
CHAPTER 4 DEPLOYMENT AND MAINTENANCE	82
4.1 Introduction	82
4.2 Try to follow the SRLC (software release life cycle)	82
CHAPTER 5 USER MANUAL	84
5.1 Introduction	84
5.2 Project Functionalities	84
5.3 Summary	88
CHAPTER 6 PROJECT SUMMARY	89
6.1 Introduction	89
6.2 Project Limitation	90
6.3 Scope	90
6.4 Future Work	91
6.5 Conclusion	91
REFERENCES	92
APPENDICES	95

LIST OF TABLES

Table 1.1: User profile for Customer	15
Table 1.2: User profile for Vendor	16
Table 1.3: User profile for Admin	16
Table 1.4: Time Frame	23
Table 1.5: Risk Management	24
Table 3.1: Test Case	80

LIST OF FIGURES

Figure 1.1: System Block Diagram	20
Figure 1.2: E-commerce Platform Deployment Schedule	23
Figure 2.1: Use Case Diagram	33
Figure 2.2: Registration / Login	52
Figure 2.3: Browse Products	53
Figure 2.4: Search Products	54
Figure 2.5: Add To cart	55
Figure 2.6: Checkout	56
Figure 2.7: Manage Profile	57
Figure 2.8: View Order History	58
Figure 2.9: Rate and Review	59
Figure 2.10: Manage Products (Vendor)	60
Figure 2.11: Process Orders (Vendor/Admin)	61
Figure 2.12: Manage Inventory	62
Figure 2.13: View Analytics	63
Figure 2.14: Manage Users (Admin)	64
Figure 2.15: Manage Categories	65
Figure 2.17: Moderate Reviews	66
Figure 2.18: Generate Reports	67
Figure 2.19: Registration	68
Figure 2.20: Browse Products	69
Figure 2.21: Search Products	69
Figure 2.22: Add To cart	70
Figure 2.23: Checkout	70
Figure 2.24: Manage Profile	71
Figure 2.25: View Order History	71
Figure 2.26: Rate and Review	72
Figure 2.27: Manage Products (Vendor)	72
Figure 2.28: Process Orders (Vendor)	73
Figure 2.29: Manage Inventory	73
Figure 2.30: View Analytics	74

Figure 2.31: Manage Users (Admin)	74
Figure 2.32: Manage Categories	75
Figure 2.33: Moderate Reviews	75
Figure 2.34: Generate Reports	76
Figure 2.35: Class Diagram	77
Figure 2.35: ER Diagram	78
Figure 5.1: Customer / vendor / Admin Login	84
Figure 5.2: Browse Product	85
Figure 5.3: Add to cart	86
Figure 5.4: Buy Product	87
Figure 5.5: Check Out	87
Figure 5.6: Manage Order	89

LIST OF APPENDICES

Appendix A: Title	19
Appendix B: Title	20

CHAPTER 1 INTRODUCTION

The swift advancement of technology has altered business operations and consumer interactions with products and services. E-commerce, as a fundamental aspect of the digital revolution, has emerged as a vital platform for global economic transactions. Multi-vendor marketplaces have evolved as a potent solution within the diverse models of e-commerce, providing a centralized platform for suppliers and customers to connect and transact effectively.

The Multi-Vendor E-Commerce Platform aims to deliver a smooth, scalable, and secure environment for several vendors to display and manage their items, while allowing customers to explore, purchase, and evaluate a diverse array of offerings. The platform facilitates transactions and connects merchants with customers through interactive features including ratings, reviews, and personalized recommendations.

The aim of this project is to develop a resilient web-based application that satisfies the requirements of many stakeholders: customers, vendors, and administrators. The platform encompasses functionalities including user registration and authentication, product catalog management, a shopping cart, secure checkout, and comprehensive order tracking. Vendors possess capabilities for inventory management, order processing, and sales analytics, whereas administrators uphold platform integrity by controlling users, categories, reviews, and creating reports.

The platform, constructed with contemporary web development technologies such as React, Node.js, and MongoDB, guarantees a responsive and user-centric experience for all participants. Implementing an Agile technique in development facilitates incremental improvements and the incorporation of user feedback, guaranteeing that the system progresses in accordance with stakeholder expectations.

This paper offers a comprehensive examination of the platform's development process, encompassing its functional and non-functional requirements, system architecture, implementation specifics, testing methodologies, and prospective enhancement strategies. The Multi-Vendor E-Commerce Platform seeks to transform the online marketplace by providing a comprehensive and scalable solution that tackles the issues encountered by both vendors and customers in the expanding e-commerce environment.

1.1 Background

The Multi-Vendor E-Commerce Platform exemplifies a novel strategy for online commerce, catering to the increasing need for adaptable and scalable markets. This project integrates technology innovations with commercial requirements to establish a platform that facilitates smooth connections between vendors and customers. The platform facilitates inventory management, order processing, and secure transactions, so assisting both small and large enterprises in effectively reaching their target audience.

1.1.1 Context and Relevance

E-commerce has transformed the retail sector, emerging as an essential conduit for business activities in the digital era. The swift expansion of online shopping has led to the rise of multi-vendor platforms, which provide a consolidated marketplace for various sellers. These platforms cater to consumer desire for convenience and diversity while allowing sellers to broaden their reach with little overhead expenses. Current systems frequently exhibit deficiencies in user-friendly interfaces, extensive vendor tools, and robust administrative management, underscoring the necessity for enhanced solutions.

1.1.2 Problem Identification

Current e-commerce platforms frequently fail to deliver an equitable experience for all parties involved. Vendors encounter obstacles include ineffective inventory management, restricted sales analytics, and inadequate customer engagement tools. Customers, however, face challenges such as insufficient personalization, untrustworthy feedback, and restricted product diversity. Administrators face challenges in preserving the platform's integrity due to insufficient moderating tools and analytics. These deficiencies require a complete system that successfully addresses the needs of vendors, customers, and administrators.

1.1.3 Purpose and Justification

The main goal of the Multi-Vendor E-Commerce Platform is to address the deficiencies observed in existing systems. The initiative seeks to equip suppliers with sophisticated tools for product management, order processing, and performance analysis, while improving the client shopping experience via tailored recommendations, safe payment methods, and effective order tracking. The platform provides administrators with comprehensive moderating tools and analytics to maintain platform integrity. This project holds considerable potential to enhance the efficiency and scalability of online commerce, benefiting both firms and consumers..

1.1.4 Scope

This project involves the creation of an extensive multi-vendor e-commerce platform. It encompasses functionalities such as user registration and authentication, product catalog exploration, order administration, payment integration, and rating/review systems. Vendors can oversee inventories, evaluate sales metrics, and process orders, whilst administrators supervise user actions, regulate reviews, and provide reports. The system is designed for scalability and security, guaranteeing adaptation to changing business requirements. The platform provides a solid basis for enterprises aiming to succeed in the competitive e-commerce environment.

1.2 Project Planning and Initiation

The creation of a Multi-Vendor E-Commerce Platform necessitates meticulous planning and analysis to guarantee its viability and congruence with corporate objectives. This section outlines the sequential feasibility study conducted during the project planning and initiation phase.

Feasibility Study (Step-by-Step)

1. Phase 1: Preliminary Analysis & Project Scope Definition

The initial phase concentrated on delineating the primary objectives of the project and establishing its scope. The primary objective was to develop a multi-vendor e-commerce platform serving three stakeholder groups: customers, vendors, and administrators. Stakeholder requirements were collected, encompassing intuitive navigation, secure payment methods, sophisticated vendor management capabilities, and effective administrative controls. An analysis of competitor platforms was conducted to find gaps and opportunities. The scope encompassed critical functionalities including product browsing, order administration, payment integration, inventory control, and analytics.

2. Phase 2: Market Feasibility Analysis (or Market Research)

Market research was conducted to evaluate the demand for multi-vendor e-commerce platforms. The results revealed an increasing tendency in e-commerce and a notable inclination towards platforms that provide a variety of products from several sellers. Surveys were administered to small and medium-sized enterprises to ascertain their difficulties in embracing e-commerce. Customers indicated the necessity for a dependable platform offering tailored purchasing experiences. The research confirmed

the project's ability to address market deficiencies and emphasized the significance of scalability and user contentment.

3. Phase 3: Technical Feasibility Analysis

The technical feasibility analysis assessed the development requirements of the platform. Contemporary technologies, like React for the frontend, Node.js for the backend, and a scalable database such as MongoDB, were chosen to guarantee optimal performance and scalability. The project aimed to utilize cloud hosting for efficient deployment and operational dependability. Essential technical considerations encompassed security (to avert data breaches), performance optimization (to manage heavy traffic), and modular architecture (to facilitate future expansions). The analysis verified that the necessary technical resources and experience were present within the team..

4. Phase 4: Financial Feasibility Analysis

A comprehensive cost analysis was conducted to evaluate the project's financial sustainability. Essential cost components encompassed development (team salary, software tools, and infrastructure), deployment (cloud hosting and domain registration), and maintenance (bug fixes, updates, and customer support). Strategies for revenue generation, including vendor subscription plans and commission-based models, were examined to ensure sustainability. The analysis forecasted a positive return on investment within two years, rendering the project financially viable.

1.3 Target User Profile and Tentative Elicitation Process

1.3.1 Target User

The principal users and stakeholders of the Multi-Vendor E-Commerce Platform are delineated as follows :

- Customers: Individuals engaging in the exploration and acquisition of products.
- Vendors: Entities or individuals overseeing their product inventories and sales transactions.
- Administrators: Platform managers accountable for system maintenance, user oversight, and analytics.

1.3.2 User profile

Table 1.1: User Profile for Customer

User Class	Note on Characteristics
Type of user	Customers browsing and purchasing products.
Age range	18–65 years.
Frequency of use	Daily or occasional.
Mandatory	Optional but necessary for product browsing and purchases.
Computer experience	Basic familiarity with e-commerce platforms.
Education	No specific educational requirement.
goal	Find, compare, and purchase products.
Language skills	Basic English and Bangla
Number of users	Estimated to grow with platform adoption.
Training	Not required; user-friendly interface provided.
Others system use	May use other e-commerce or digital payment platforms.
Way of working	Individual users accessing via web or mobile.

Table 1.2: User Profile for Vendors

User Class	Note on Characteristics
Type of user	Vendors managing their product catalog and sales.
Age range	22–60 years.
Frequency of use	Daily or multiple times a week.
Mandatory	Required to operate their online store.
Computer experience	Moderate familiarity with business tools and online platforms.
Education	Basic education with business knowledge preferred.
goal	List products, manage sales, and increase revenue.
Language skills	Basic English or other supported local languages (Bangla).
Number of users	Predicted growth with platform expansion.
Training	Minimal training for dashboard and tools.
Others system use	May use inventory or financial management tools.
Way of working	Access via a dashboard for product and order management.

Table 1.3: User Profile for Administrators

User Class	Note on Characteristics
Type of user	Administrators managing the platform.
Age range	25–50 years.
Frequency of use	Daily.
Mandatory	Required for system monitoring and maintenance.
Computer experience	Advanced proficiency with backend systems and analytics.
Education	Degree in IT, computer science, or relevant fields preferred.
goal	Maintain platform stability, manage users, and analyze system data.
Language skills	Proficient in Bangla and English.

Number of users	Limited to a few platform operators.
Training	Required for system administration tools.
Others system use	May use analytics, security tools, and CMS platforms.
Way of working	Operational management through an admin dashboard.

1.3.3 Elicitation Process

To obtain thorough and actionable user needs for the Multi-Vendor E-Commerce Platform, the subsequent elicitation approaches were employed:

1. Interviews

Description:

Interviews were performed with representatives from each stakeholder category (customers, vendors, and administrators). This approach facilitated comprehensive dialogues regarding user expectations, obstacles, and feature inclinations.

- **Participants:** A cohort of habitual online consumers, proprietors of small enterprises, and information technology administrators
- **Objective:** To comprehend particular requirements, challenges, and feature recommendations.
- **Outcome:** Understanding of intuitive interfaces, payment alternatives, and inventory management systems.

2. Surveys

Description:

Online surveys were disseminated to a broader audience to collect quantitative data. The questions were framed to emphasize desirable e-commerce functionalities and anticipated user experiences.

- **Participants:** General users, vendors, and prospective administrators.
- **Objective:** To ascertain prevalent features such as search filters, secure payment mechanisms, and dashboard functionalities.

- **Outcome:** Prioritization of critical characteristics according to response frequency.

3. Focus Groups

Description:

Focus groups convened a variety of users to deliberate on the platform's concept, difficulties, and possible solutions.

- **Participants:** A diverse group comprising customers, vendors, and subject matter experts.
- **Objective:** To promote the exchange of ideas and comprehend collaboration requirements.
- **Outcome:** Recognition of distinctive attributes such as vendor-specific discounts and analytical dashboards.

4. Observation

Description:

An examination of user behavior on current e-commerce platforms was performed to comprehend real-time interactions and usability issues.

- **Participants:** Discreet observation of customer and vendor interactions on rival platforms.
- **Objective:** To discover challenges include sluggish product searches, inadequate navigation, or restricted payment alternatives.
- **Outcome:** Suggestions for optimized navigation, expedited loading times, and secure multi-vendor integrations.

5. Document Analysis

Description:

An analysis of existing documentation from comparable systems was conducted to obtain insights into conventional practices and prevalent concerns.

- **Sources:** User manuals, reports, and feedback from prominent e-commerce sites
- **Objective:** To evaluate the platform in comparison to industry norms.

- **Outcome:** Development of a preliminary feature list, encompassing multi-currency functionality and user review mechanisms.

Summary

The elicitation techniques guaranteed that requirements were precise, thorough, and consistent with stakeholder expectations. Each strategy facilitated a comprehensive grasp of the system's objectives and obstacles, propelling the creation of a user-focused e-commerce platform.

1.4 Project Block Diagram

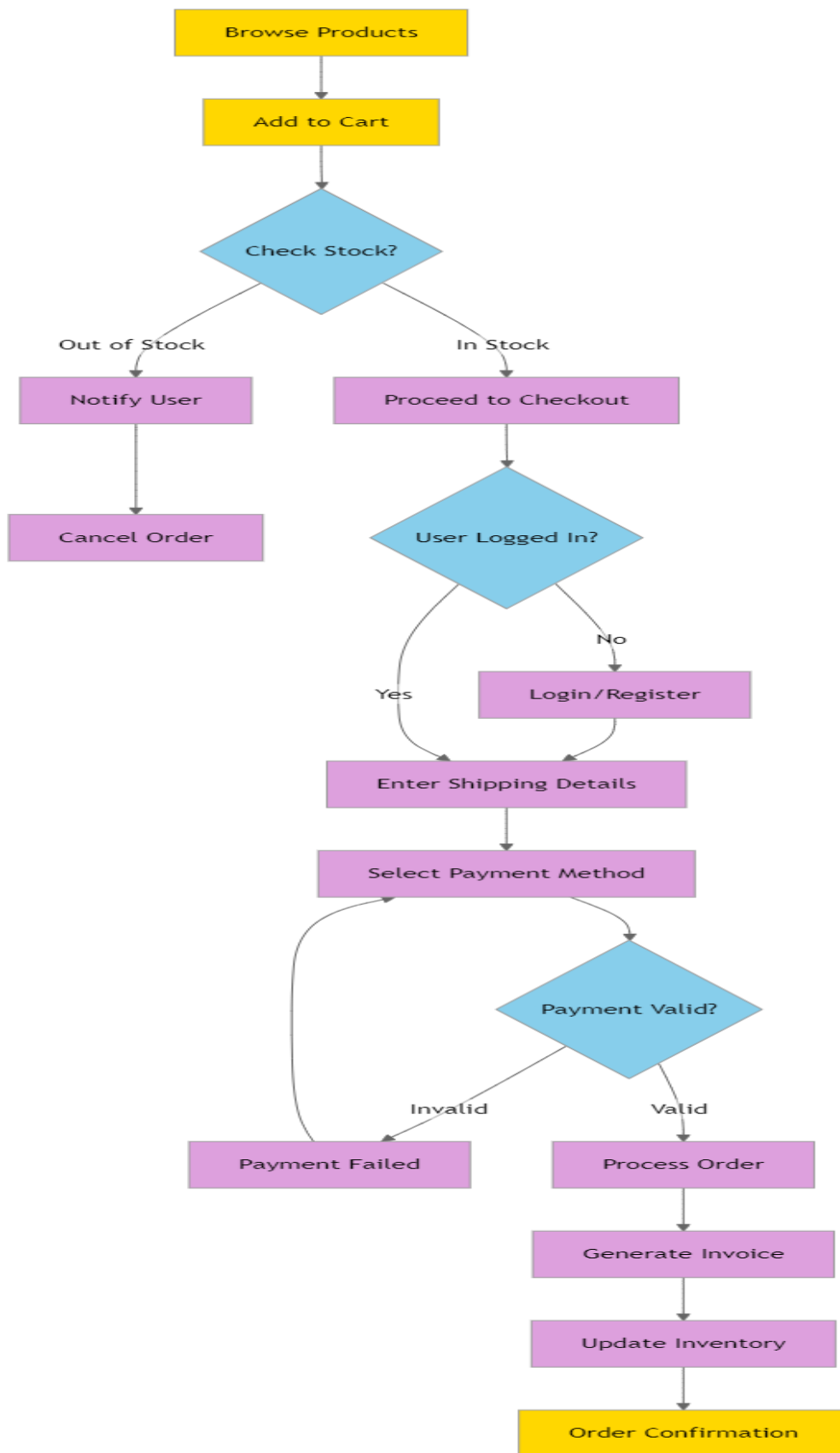


Figure 1.1: System Block Diagram

1.5 System Requirements

1.5.1 Hardware Requirements

The following hardware criteria are necessary for the efficient operation and maintenance of the Multi-Vendor E-Commerce Platform:

- **Client-Side Hardware:**
 - Processor: Minimum Intel Core i3 or equivalent
 - RAM: 4 GB or higher
 - Storage: 10 GB free space
 - Display: Minimum 1366x768 resolution
 - Network: Broadband internet connection
- **Server-Side Hardware:**
 - Processor: Quad-core Intel Xeon or equivalent
 - RAM: 16 GB or higher
 - Storage: SSD with at least 500 GB capacity
 - Network: High-speed internet with 1 Gbps bandwidth
 - Backup: External storage for backups (1 TB or more)

1.5.2 Software Requirements

The project requires specific software environments to function properly:

- **Client-Side Software:**
 - Operating System: Windows 10 or later, macOS, or Linux
 - Web Browser: Google Chrome, Mozilla Firefox, or Microsoft Edge
 - Additional Tools: JavaScript-enabled browser
- **Server-Side Software:**
 - Operating System: Ubuntu Server 20.04 LTS or equivalent
 - Web Server: Apache or Nginx
 - Application Framework: Node.js (v16 or later)
 - Database: MySQL 8.0 or MongoDB
 - Programming Languages: JavaScript, HTML5, CSS3, Python for auxiliary scripts
 - Additional Tools: Docker for containerization, Redis for caching
- **Development Tools:**

- IDE: Visual Studio Code
- Version Control: Git
- Testing Tools: Selenium, Postman
- Deployment Tools: AWS EC2 or similar cloud services

1.5.3 Constraints and Dependencies

- **Constraints:**

- Internet Dependency: The platform requires a stable internet connection to ensure real-time interactions between customers, vendors, and administrators.
- Performance Scalability: Hardware and software must handle increased traffic during sales or promotions.
- Data Privacy: Compliance with data protection regulations such as GDPR or CCPA is mandatory.

- **Dependencies:**

- Payment Gateways: Integration with external payment systems like Stripe, PayPal, or Razorpay for secure transactions.
- Third-Party APIs: Use of APIs for functionalities such as currency conversion, shipping tracking, and analytics.
- Hosting Services: Dependency on cloud providers (e.g., AWS, Azure, or Google Cloud) for scalability and uptime.
- Open-Source Libraries: Usage of external libraries for frontend (React) and backend (Express.js) development.

1.6 Project Scheduling

Time Frame/ Gantt Chart/Risk Management.

1.6.1 Time Frame

The Multi-Vendor E-Commerce Platform project is scheduled from July 1, 2024, to December 25, 2024, utilizing an Agile development process. The project is organized into eight sprints, each targeting distinct targets, so facilitating incremental advancement and flexibility.

Table 1.4: Time Fream

Sprint	Duration	Focus
Sprint 1	1st July – 15th July	Project planning, environment setup, and UI/UX design.
Sprint 2	16 th July – 30 th July	Product catalog creation and search feature integration.
Sprint 3	1st August – 15th August	Shopping cart development and checkout system.
Sprint 4	16th August – 31st August	User profile and order history management.
Sprint 5	1st September – 15th September	Reviews and vendor inventory management.
Sprint 6	16th September – 30th September	Admin features and analytics dashboard.
Sprint 7	1st October – 20th October	Integration testing, bug fixes, and optimization.
Sprint 8	21st October – 30th November	Final testing, UAT, and preparation for deployment.
Sprint 9	1st December – 15th December	Production deployment and post-deployment maintenance.

1.6.2 Gantt Chart

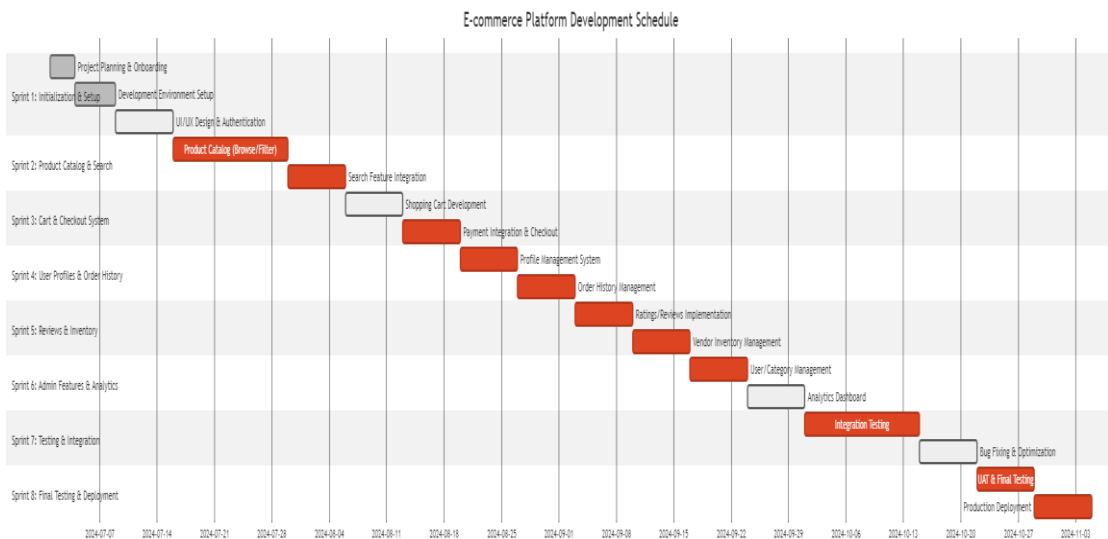


Figure 1.2:Deployment Schedule

1.6.3 Risk Management

Table 1.5: Risk Management

Risk	Likelihood	Impact	Mitigation Strategy
Delays in Requirement Gathering	High	High	Conduct stakeholder interviews early; set clear deadlines.
Scope Creep	Medium	High	Implement strict change management processes and regular scope reviews.
Performance Bottlenecks	Medium	Medium	Perform load testing during development; optimize backend queries.
Payment Gateway Issues	Low	High	Partner with reliable payment service providers; test integration extensively.
Security Vulnerabilities	Medium	High	Regular security audits- implement strong encryption and access controls.
Integration Challenges with Third-Party APIs	Low	Medium	Test API integration early; maintain up-to-date API documentation.
Lack of User Engagement Post-Launch	Low	Medium	Plan and execute a post-launch marketing campaign; gather user feedback.

1.7 Summary

The introductory chapter establishes the groundwork for the Multi-Vendor E-Commerce Platform project by offering a comprehensive review of its background and significance. The project begins by delineating its context, highlighting the limitations of conventional e-commerce systems, and underscoring the necessity for a multi-vendor solution to enhance interactions between vendors and users. The purpose and justification emphasize the project's aim to improve user experience, scalability, and vendor management for a diverse audience. The scope details the functionalities, encompassing user and admin modules, product administration, and efficient transactions.

The chapter examines the feasibility analysis, assessing the project's viability and its congruence with company objectives. A detailed user profile and requirement elicitation procedure delineate the identification and translation of user demands into system requirements. The project block diagram delineates the system's architecture and operational flow. Furthermore, system requirements encompassing hardware, software, and limitations are delineated to guarantee the project's technological viability.

A clearly delineated project timeline employing Agile approach guarantees methodical advancement through iterative sprints, emphasizing punctual delivery and quality. This chapter finishes with a summary that encapsulates the foundation established for the project's effective implementation.

CHAPTER 2 DESIGN AND IMPLEMENTATION

2.1 Introduction

This chapter delineates the design and execution of the multi-vendor e-commerce platform. It offers a comprehensive summary of the architectural framework, system models, and technical selections employed during development. The discussion encompasses essential design ideas, user interface factors, and backend functionalities to guarantee the platform adequately fulfills both functional and non-functional criteria. The implementation process delineates the sequential methodology employed to construct and integrate diverse system components, guaranteeing a cohesive and resilient solution for all stakeholders.

2.2 Functional Requirements

The Functional Requirements delineate the precise functionalities that the multi-vendor e-commerce platform must deliver to satisfy user and stakeholder requirements. These specifications encompass functionalities for customers, vendors, and administrators to guarantee seamless contact, effective management, and secure operations.

Key functionalities include:

- **Customer Features:** User registration, product exploration, search and filtering capabilities, shopping cart functionality, order placing, payment processing, and review submission
- **Vendor Features:** Product administration, inventory monitoring, order processing, and analytical insights.
- **Administrator Features:** User administration, vendor authorization, category oversight, platform analytics, and comprehensive system monitoring.

The objective of these objectives is to provide a resilient, user-centric platform that serves all stakeholders while guaranteeing scalability, security, and dependability.

FR01	User Registration and Authentication
-------------	---------------------------------------------

Description	The system must allow customers, vendors, and admins to register and authenticate securely using an email/password or third-party services (Google, Facebook).
Stakeholder	Customer, Vendor, Admin

FR02	Product Catalog
Description	Customers must be able to browse and search the product catalog based on various filters (price, category, rating) to find desired products.
Stakeholder	Customer , Vendor (Vendor's products will be listed in the catalog)

FR03	Shopping Cart and Checkout
Description	Customers must be able to add products to their shopping cart, review them, and proceed to checkout. The system must support payment integration for completing purchases.
Stakeholder	Customer, Vendor (Processes customer orders)

FR04	Order Management
Description	Vendors must be able to manage customer orders, update statuses (e.g., shipped, delivered), and handle returns or cancellations.
Stakeholder	Vendor, Customer (Tracks their own orders)

FR05	Vendor Management
Description	Admins must be able to manage vendors, approve their registrations, and monitor their product listings and order fulfillment performance.
Stakeholder	Vendor, Admin

FR06	Reviews and Ratings
Description	Customers must be able to rate and review products. Vendors can view the reviews of their products, and admins can moderate reviews to ensure compliance with guidelines.

Stakeholder	Customer, Vendor, Admin
--------------------	-------------------------

FR07	User Roles and Permissions
Description	The system must define distinct roles for customers, vendors, and admins, with specific access rights and permissions based on their roles.
Stakeholder	Customer, Vendor, Admin

FR08	Inventory Management
Description	Vendors must be able to manage their product inventory, updating stock levels, pricing, and availability.
Stakeholder	Vendor, Admin (Monitors inventory activities)

FR09	Analytics and Reporting
Description	Vendors and admins must have access to analytics dashboards to track sales performance, customer behavior, and product demand.
Stakeholder	Vendor, Admin

FR10	Moderation
Description	Admins must have the ability to moderate user-generated content such as reviews, comments, and vendor product listings to ensure compliance with platform rules.
Stakeholder	Customer (Interacts with moderated content) Admin, Vendor

FR11	Notifications
Description	The system must notify customers about order updates, vendors about new orders, and admins about system alerts or approvals needed.

Stakeholder	Customer, Vendor, Admin
--------------------	-------------------------

FR12	Payment Integration
Description	The system must support multiple payment gateways for customers to securely complete their purchases, with payment tracking available to vendors.
Stakeholder	Customer, Vendor, Admin

FR13	Security and Data Privacy
Description	The system must ensure the security of user data (personal information, payment details) and provide options for account recovery, password resets, and privacy settings.
Stakeholder	Customer, Vendor, Admin

FR14	Product Categories and Management
Description	Admins must be able to manage product categories, ensuring products are appropriately categorized for ease of search and discovery.
Stakeholder	Vendor (Ensures their products are listed in the right category)

2.3 Non-Functional Requirements

This section addresses the non-functional requirements that delineate the overall quality and performance benchmarks of the multi-vendor e-commerce platform. These requirements pertain to essential elements such as system stability, scalability, security, usability, and performance efficiency. By complying with these standards, the platform guarantees an uninterrupted user experience, stringent security protocols, and the capacity to manage substantial traffic and extensive datasets efficiently. Non-functional needs are crucial for determining the platform's long-term sustainability and user pleasure.

2.3.1 Performance

The system must provide a cohesive and responsive user experience under diverse conditions. It must facilitate simultaneous user access with minimal latency, guaranteeing rapid loading times

for product catalogs, search outcomes, and checkout procedures. Benchmarking must guarantee that the response time remains within 3 seconds for 95% of requests during peak loads.

2.3.2 Scalability

The platform must accommodate a growing number of users, merchants, and transactions without a decline in performance. It must facilitate both horizontal and vertical scaling to support business expansion, encompassing the integration of additional features, vendors, and customers.

2.3.3 Security

The system must guarantee the protection of user data, encompassing personal and financial information. Implement security techniques such as SSL/TLS for data transmission, effective password encryption, and adherence to standards like PCI DSS for payment data. Routine vulnerability assessments and penetration testing must also be performed.

2.3.4 Reliability

The platform must function reliably without unforeseen interruptions. It must attain a minimum uptime of 99.9% to guarantee uninterrupted availability. Mechanisms for error detection and recovery must be implemented to mitigate disruptions.

2.3.5 Availability

The system must be available around the clock to consumers across several time zones. Cloud hosting solutions must be employed to guarantee high availability and redundancy, incorporating failover procedures to address server outages.

2.3.6 Usability

The platform must be accessible, with an easy interface that accommodates all stakeholders, including customers, vendors, and administrators. Accessibility features, including screen reader compatibility and keyboard navigation, must be implemented.

2.3.7 Maintainability

The platform's codebase must be modular and thoroughly documented to enable efficient updates and problem resolutions. Routine maintenance tasks, including database optimization and software updates, must not interfere with the system's operations.

2.3.8 Portability

The solution must be deployable in many contexts, including on-premises and cloud, with minimal setup adjustments. It should facilitate migration to upgraded technologies as necessary.

2.3.9 Data Integrity

The system must guarantee the precision, coherence, and legitimacy of all data. Transactional actions, including order placement and inventory updates, must be atomic to prevent mistakes during concurrent usage.

2.3.10 Compliance

The platform must adhere to legal and regulatory obligations, including GDPR for user data protection and local tax legislation. Periodic audits must be performed to guarantee compliance with established standards.

2.3.11 Backup and Recovery

The system must execute automated daily backups and establish a comprehensive recovery procedure to avert data loss. Recovery from hardware or software problems should not exceed 4 hours.

2.3.12 Responsiveness

All processes, including as loading websites, processing payments, and updating inventory, must be performed with minimal latency. The system must deliver instantaneous updates regarding stock availability and order statuses.

2.3.13 Localization and Internationalization

The platform must accommodate several languages and currencies to serve a diversified user base across different geographies. Date, time, and currency formats must be presented according to the user's locale.

2.3.14 Compatibility with Different Devices and Browsers

The solution must operate seamlessly across all principal web browsers and be optimized for mobile devices, tablets, and desktops. Adherence to responsive design principles is essential for maintaining consistency in user experience.

2.3.15 Auditability and Monitoring

The platform must record all essential operations, including user actions, transactions, and system faults, for auditing purposes. Real-time monitoring instruments must be employed to assess system performance and usage trends.

2.3.16 Error Management and Documentation

The system must identify and manage failures effectively, delivering helpful alerts to users while safeguarding sensitive information. Logs must document comprehensive information regarding errors to facilitate debugging and enhance system performance.

2.3.17 Load Distribution

The system must allocate incoming traffic uniformly among servers to provide stability and performance during peak demand. Load balancing solutions must be scalable to manage abrupt traffic surges efficiently.

2.3.18 Service Continuity and Disaster Recovery

The platform must possess a disaster recovery plan to mitigate downtime and data loss during catastrophic situations. Redundant systems and data replication must be utilized to guarantee service continuity.

2.4 Object-oriented System design using UML

2.4.1 Use Case Diagram

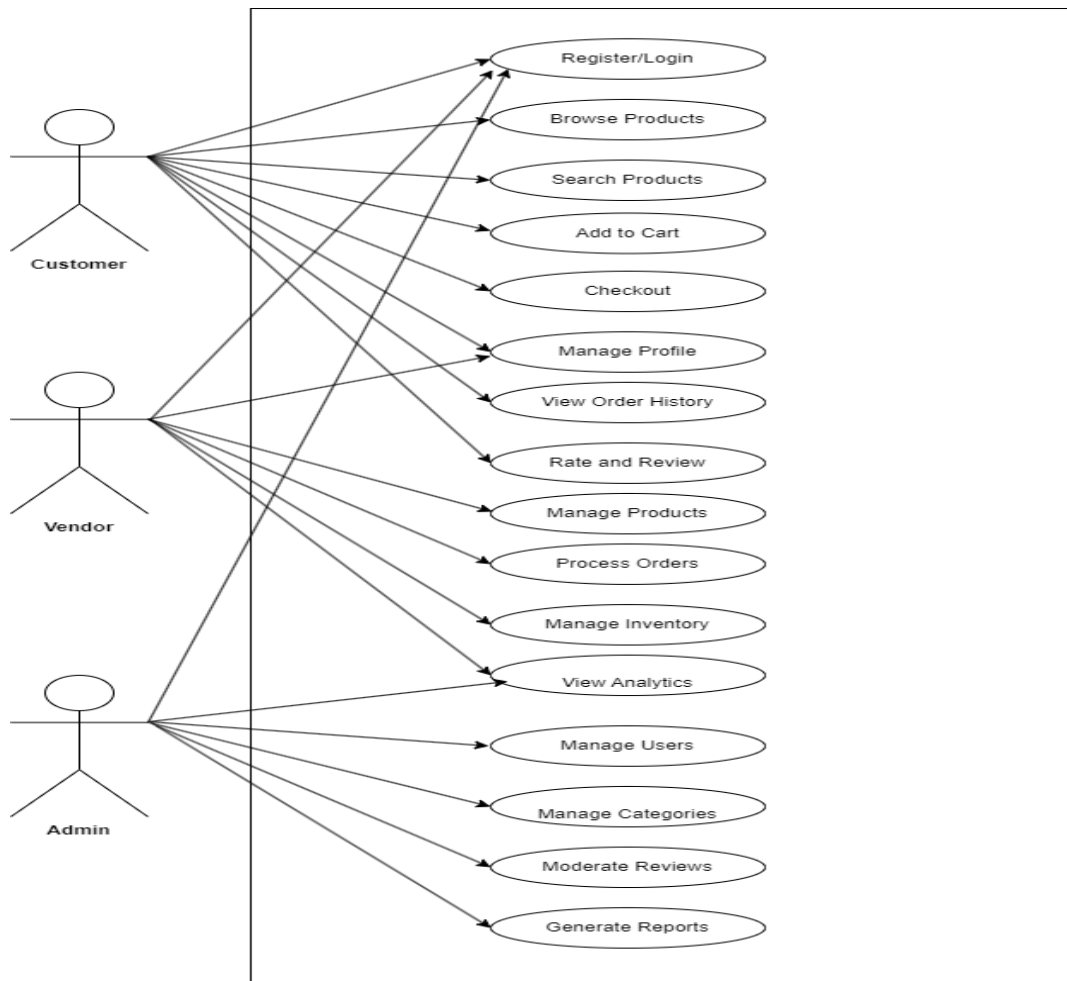


Figure 2.1: Use case Diagram

2.4.2 Case Description

Case Description-01: Registration

Use Case	Registration												
Goal	Users can register to sign in to the system.												
Precondition	Users must install the Hotel Management app for registration.												
Success End Condition	Notification: !!!Successfully Registered!!!												
Failed End Condition	Notification: "Submission Not Submitted"												
Primary Actors:	Customer, Vendor												
Secondary Actors:	Admin												
Trigger	User will request a registration form to fill up												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Press "Registration" Button</td> </tr> <tr> <td>2.</td> <td>Provide registration form</td> </tr> <tr> <td>3.</td> <td>Enter Information</td> </tr> <tr> <td>4.</td> <td>Press "Submit" Button.</td> </tr> <tr> <td>5.</td> <td>Information saved</td> </tr> <tr> <td>6.</td> <td>The system saves the details and shows them !!! Successfully Registered!!! Notify</td> </tr> </table>	1.	Press "Registration" Button	2.	Provide registration form	3.	Enter Information	4.	Press "Submit" Button.	5.	Information saved	6.	The system saves the details and shows them !!! Successfully Registered!!! Notify
1.	Press "Registration" Button												
2.	Provide registration form												
3.	Enter Information												
4.	Press "Submit" Button.												
5.	Information saved												
6.	The system saves the details and shows them !!! Successfully Registered!!! Notify												

Alternative Flows	1.1	System Error
		1.1.a. Try Again!!
	4.1	The user Did not fill up the details!
		4.1.a. Checked By the system & Notify by “Please! Fill Up the Box”.
	5.1	The system did not respond
		5.1.a. Show Error Message.
	6.1	The system Doesn’t save the details.
		6.1.a. Notification: “Details did not Save”
Quality Requirements	The user Will fill up all the details in 30 minutes.	

Case Description-02: Login

Use Case	Login
Goal	Users can log into their accounts.
Precondition	Users must have registered successfully.
Success End Condition	Notification: “Login Successful”
Failed End Condition	Notification: “Invalid Credentials”
Primary Actors:	Customer, Vendor, Admin
Secondary Actors:	None
Trigger	Users will provide login credentials.

Description / Main Success Scenario	1.	Press the "Login" button.
	2.	Provide username and password.
	3.	Submit credentials.
	4.	System validates the credentials.
	5.	Login is successful, and the user is redirected to the dashboard.
Alternative Flows	1.1	Invalid credentials entered.
		1.1.a. Notify: "Invalid Username or Password."
	2.1	System Error.
		2.1.a Notify: "Please Try Again."
Quality Requirements	The login process must complete within 10 seconds.	

Case Description-03: Browse Products

Use Case	Browse Products
Goal	Users can view and browse the product catalog.
Precondition	Products must be added by vendors and approved by the admin.
Success End Condition	The product list is successfully viewed by users.
Failed End Condition	Error: "Unable to Load Products"
Primary Actors:	Customer
Secondary Actors:	Vendor, Admin

Trigger	Users go to the catalog of products.								
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access the "Products" page.</td> </tr> <tr> <td>2.</td> <td>System loads and displays the product catalog.</td> </tr> <tr> <td>3.</td> <td>Users can filter and sort products by categories, price, and ratings.</td> </tr> <tr> <td>4.</td> <td>Users select a product to view details.</td> </tr> </table>	1.	Access the "Products" page.	2.	System loads and displays the product catalog.	3.	Users can filter and sort products by categories, price, and ratings.	4.	Users select a product to view details.
1.	Access the "Products" page.								
2.	System loads and displays the product catalog.								
3.	Users can filter and sort products by categories, price, and ratings.								
4.	Users select a product to view details.								
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>System Error while loading.</td> </tr> <tr> <td></td> <td>1.1.a. Notify: "Error Loading Products. Try Again Later."</td> </tr> </table>	1.1	System Error while loading.		1.1.a. Notify: "Error Loading Products. Try Again Later."				
1.1	System Error while loading.								
	1.1.a. Notify: "Error Loading Products. Try Again Later."								
Quality Requirements	Products should load within 5 seconds.								

Case Description-04: Search Products

Use Case	Search Products
Goal	Users can search for specific products.
Precondition	Users can search for specific products.
Success End Condition	Search results are displayed based on the entered query.
Failed End Condition	Notification: "No Results Found"
Primary Actors:	Customer
Secondary Actors:	Vendor, Admin

Trigger	Users enter a search query.								
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Enter a search query in the search bar.</td> </tr> <tr> <td>2.</td> <td>Press the "Search" button.</td> </tr> <tr> <td>3.</td> <td>System retrieves matching products.</td> </tr> <tr> <td>4.</td> <td>Display search results to the user.</td> </tr> </table>	1.	Enter a search query in the search bar.	2.	Press the "Search" button.	3.	System retrieves matching products.	4.	Display search results to the user.
1.	Enter a search query in the search bar.								
2.	Press the "Search" button.								
3.	System retrieves matching products.								
4.	Display search results to the user.								
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>No matching products found.</td> </tr> <tr> <td></td> <td>1.1.a. Notify: "No Results Found for Your Query."</td> </tr> <tr> <td>2.1</td> <td>System Error.</td> </tr> <tr> <td></td> <td>2.1.a Notify: "Search Feature Unavailable. Try Again Later."</td> </tr> </table>	1.1	No matching products found.		1.1.a. Notify: "No Results Found for Your Query."	2.1	System Error.		2.1.a Notify: "Search Feature Unavailable. Try Again Later."
1.1	No matching products found.								
	1.1.a. Notify: "No Results Found for Your Query."								
2.1	System Error.								
	2.1.a Notify: "Search Feature Unavailable. Try Again Later."								
Quality Requirements	Search results should be displayed within 3 seconds.								

Case Description-05: Add to Cart

Use Case	Add to Cart
Goal	Items can be added by customers to their shopping cart.
Precondition	The product has to be stocked.
Success End Condition	Notification: "Item Added to Cart"
Failed End Condition	Notification: "Unable to Add Product to Cart"
Primary Actors:	Customer
Secondary Actors:	None

Trigger	The "Add to Cart" option is chosen by users.										
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Select a item.</td> </tr> <tr> <td>2.</td> <td>Click on the "Add to Cart" button.</td> </tr> <tr> <td>3.</td> <td>System checks stock availability.</td> </tr> <tr> <td>4.</td> <td>System adds the product to the cart.</td> </tr> <tr> <td>5.</td> <td>Notify: "Product Added to Cart."</td> </tr> </table>	1.	Select a item.	2.	Click on the "Add to Cart" button.	3.	System checks stock availability.	4.	System adds the product to the cart.	5.	Notify: "Product Added to Cart."
1.	Select a item.										
2.	Click on the "Add to Cart" button.										
3.	System checks stock availability.										
4.	System adds the product to the cart.										
5.	Notify: "Product Added to Cart."										
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Item out of stock.</td> </tr> <tr> <td></td> <td>1.1.a. Notify: "Product is Out of Stock."</td> </tr> <tr> <td>2.1</td> <td>System Error.</td> </tr> <tr> <td></td> <td>2.1.a Notify: "Unable to Process Request. Try Again Later."</td> </tr> </table>	1.1	Item out of stock.		1.1.a. Notify: "Product is Out of Stock."	2.1	System Error.		2.1.a Notify: "Unable to Process Request. Try Again Later."		
1.1	Item out of stock.										
	1.1.a. Notify: "Product is Out of Stock."										
2.1	System Error.										
	2.1.a Notify: "Unable to Process Request. Try Again Later."										
Quality Requirements	The process should complete within 3 seconds.										

Case Description-06: Checkout

Use Case	Checkout
Goal	Customers can purchase the items in their cart.
Precondition	Customers must have items in their cart.
Success End Condition	Notification: "Order Placed Successfully"
Failed End Condition	Notification: "Unable to Place Order"
Primary Actors:	Customer
Secondary Actors:	Payment Gateway

Trigger	Users proceed to checkout.												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access the "Cart" page.</td> </tr> <tr> <td>2.</td> <td>Click on the "Checkout" button.</td> </tr> <tr> <td>3.</td> <td>Provide shipping and payment details.</td> </tr> <tr> <td>4.</td> <td>Confirm the order.</td> </tr> <tr> <td>5.</td> <td>System processes the payment.</td> </tr> <tr> <td>6.</td> <td>Notify: "Order Placed Successfully."</td> </tr> </table>	1.	Access the "Cart" page.	2.	Click on the "Checkout" button.	3.	Provide shipping and payment details.	4.	Confirm the order.	5.	System processes the payment.	6.	Notify: "Order Placed Successfully."
1.	Access the "Cart" page.												
2.	Click on the "Checkout" button.												
3.	Provide shipping and payment details.												
4.	Confirm the order.												
5.	System processes the payment.												
6.	Notify: "Order Placed Successfully."												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Payment gateway error.</td> </tr> <tr> <td></td> <td>1.1.a. Notify: "Payment Failed. Try Again."</td> </tr> <tr> <td>2.1</td> <td>System Error.</td> </tr> <tr> <td></td> <td>2.1.a Notify: "Unable to Process Order. Try Again Later."</td> </tr> </table>	1.1	Payment gateway error.		1.1.a. Notify: "Payment Failed. Try Again."	2.1	System Error.		2.1.a Notify: "Unable to Process Order. Try Again Later."				
1.1	Payment gateway error.												
	1.1.a. Notify: "Payment Failed. Try Again."												
2.1	System Error.												
	2.1.a Notify: "Unable to Process Order. Try Again Later."												
Quality Requirements	The checkout process should complete within 2 minutes.												

Case Description-07: Manage Profile

Use Case	Manage Profile
Goal	Users can view and update their profile information
Precondition	User must be logged into the system
Success End Condition	Notification: "Profile Updated Successfully!"
Failed End Condition	Notification: "Update Failed"
Primary Actors:	Customer, Vendor
Secondary Actors:	Admin

Trigger	Notification: "Update Failed"	
Description / Main Success Scenario	1.	Click "Profile" button.
	2.	System displays current profile information
	3.	User modifies desired information and click Update.
	4.	System validates information
	5.	System saves updated information
	6.	System displays success message
Alternative Flows	1.1	Invalid input format
		System displays format error message
	2.1	Network connection lost
		2.1.a. System saves draft locally
	3.1	Validation fails
		3.1.a System highlights invalid fields
Quality Requirements	Profile updates should be processed within 5 seconds	

Case Description-08: View Order History

Use Case	View Order History
Goal	Users can view their past orders and order status
Precondition	User must be logged in and have at least one order
Success End Condition	Order history displayed successfully
Failed End Condition	"Unable to retrieve order history" message
Primary Actors:	Customer
Secondary Actors:	Admin

Trigger	User selects "Order History" option												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Click "Order History" button</td> </tr> <tr> <td>2.</td> <td>System retrieves order list</td> </tr> <tr> <td>3.</td> <td>System displays orders in chronological order</td> </tr> <tr> <td>4.</td> <td>User can select specific order for details</td> </tr> <tr> <td>5.</td> <td>System displays detailed order information</td> </tr> </table>	1.	Click "Order History" button	2.	System retrieves order list	3.	System displays orders in chronological order	4.	User can select specific order for details	5.	System displays detailed order information		
1.	Click "Order History" button												
2.	System retrieves order list												
3.	System displays orders in chronological order												
4.	User can select specific order for details												
5.	System displays detailed order information												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>No orders found</td> </tr> <tr> <td></td> <td>1.1.a Display "No orders yet" message</td> </tr> <tr> <td>2.1</td> <td>Connection error</td> </tr> <tr> <td></td> <td>2.1.a Show connection error message.</td> </tr> <tr> <td>3.1</td> <td>Order details unavailable</td> </tr> <tr> <td></td> <td>3.1.a. Show "Details temporarily unavailable"</td> </tr> </table>	1.1	No orders found		1.1.a Display "No orders yet" message	2.1	Connection error		2.1.a Show connection error message.	3.1	Order details unavailable		3.1.a. Show "Details temporarily unavailable"
1.1	No orders found												
	1.1.a Display "No orders yet" message												
2.1	Connection error												
	2.1.a Show connection error message.												
3.1	Order details unavailable												
	3.1.a. Show "Details temporarily unavailable"												
Quality Requirements	Order history should load within 3 seconds												

Case Description-09: Rate and Review

Use Case	Rate and Review
Goal	Users can rate products and leave reviews
Precondition	User must be logged in and have purchased the product
Success End Condition	Review posted successfully
Failed End Condition	Review submission failed
Primary Actors:	Customer
Secondary Actors:	

Trigger	User selects "Write a Review" option on a purchased product												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Select product to review</td> </tr> <tr> <td>2.</td> <td>System verifies purchase</td> </tr> <tr> <td>3.</td> <td>Display review form</td> </tr> <tr> <td>4.</td> <td>User enters rating (1-5 stars)</td> </tr> <tr> <td>5.</td> <td>User writes review text and submit</td> </tr> <tr> <td>6.</td> <td>The system saves the details and shows them !!! Successfully Registered!!! Notify</td> </tr> </table>	1.	Select product to review	2.	System verifies purchase	3.	Display review form	4.	User enters rating (1-5 stars)	5.	User writes review text and submit	6.	The system saves the details and shows them !!! Successfully Registered!!! Notify
1.	Select product to review												
2.	System verifies purchase												
3.	Display review form												
4.	User enters rating (1-5 stars)												
5.	User writes review text and submit												
6.	The system saves the details and shows them !!! Successfully Registered!!! Notify												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Purchase not verified</td> </tr> <tr> <td></td> <td>1.1.a Show "Purchase required" message</td> </tr> <tr> <td>2.1</td> <td>Review too short</td> </tr> <tr> <td></td> <td>2.1.a. Request for minimum length</td> </tr> <tr> <td>3.1</td> <td>Inappropriate content</td> </tr> <tr> <td></td> <td>3.2.a Flag for moderation</td> </tr> </table>	1.1	Purchase not verified		1.1.a Show "Purchase required" message	2.1	Review too short		2.1.a. Request for minimum length	3.1	Inappropriate content		3.2.a Flag for moderation
1.1	Purchase not verified												
	1.1.a Show "Purchase required" message												
2.1	Review too short												
	2.1.a. Request for minimum length												
3.1	Inappropriate content												
	3.2.a Flag for moderation												
Quality Requirements	Reviews must be at least 20 characters long												

Case Description-10: Manage Products

Use Case	Manage Products
Goal	Vendors can add, edit, and delete their products
Precondition	User must be logged in as vendor
Success End Condition	Product successfully added/updated/deleted
Failed End Condition	Product operation failed
Primary Actors:	Vendor
Secondary Actors:	Admin

Trigger	Vendor selects "Manage Products" option												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access product management dashboard</td> </tr> <tr> <td>2.</td> <td>Choose operation (Add/Edit/Delete)</td> </tr> <tr> <td>3.</td> <td>Enter/modify product details</td> </tr> <tr> <td>4.</td> <td>Upload product images and Set pricing and inventory</td> </tr> <tr> <td>5.</td> <td>Submit changes</td> </tr> <tr> <td>6.</td> <td>System validates information and update product database</td> </tr> </table>	1.	Access product management dashboard	2.	Choose operation (Add/Edit/Delete)	3.	Enter/modify product details	4.	Upload product images and Set pricing and inventory	5.	Submit changes	6.	System validates information and update product database
1.	Access product management dashboard												
2.	Choose operation (Add/Edit/Delete)												
3.	Enter/modify product details												
4.	Upload product images and Set pricing and inventory												
5.	Submit changes												
6.	System validates information and update product database												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Invalid product details</td> </tr> <tr> <td></td> <td>1.1.a. Show validation errors</td> </tr> <tr> <td>2.1</td> <td>Image upload fails</td> </tr> <tr> <td></td> <td>42.1.a Retry upload option.</td> </tr> <tr> <td>3.1</td> <td>Duplicate product SKU</td> </tr> <tr> <td></td> <td>3.1.a Request unique SKU</td> </tr> </table>	1.1	Invalid product details		1.1.a. Show validation errors	2.1	Image upload fails		42.1.a Retry upload option.	3.1	Duplicate product SKU		3.1.a Request unique SKU
1.1	Invalid product details												
	1.1.a. Show validation errors												
2.1	Image upload fails												
	42.1.a Retry upload option.												
3.1	Duplicate product SKU												
	3.1.a Request unique SKU												
Quality Requirements	Product updates should be live within 2 minutes												

Case Description-11: Process Orders

Use Case	Process Orders
Goal	Vendors can manage and process customer orders
Precondition	User must be logged in as vendor with pending orders
Success End Condition	Order status updated successfully
Failed End Condition	Order processing failed
Primary Actors:	Vendor
Secondary Actors:	Customer

Trigger	New order notification or vendor accesses order management	
Description / Main Success Scenario	1.	View pending orders
	2.	Select order to process
	3.	Review order details
	4.	Update inventory
	5.	Generate shipping label
	6.	Update order status and notify customer.
Alternative Flows	1.1	Out of stock
		1.1.a. Notify customer of delay
	2.1	Shipping label error
		2.1.a Manual label generation
	3.1	Status update fails
		3.1.a. Retry update
Quality Requirements	Orders must be processed within 24 hours	

Case Description-12: Manage Inventory

Use Case	Manage Inventory
Goal	Vendors can track and update product inventory
Precondition	User must be logged in as vendor
Success End Condition	Inventory updated successfully
Failed End Condition	Inventory update failed
Primary Actors:	Vendor
Secondary Actors:	System

Trigger	Vendor selects "Inventory Management" option												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access inventory dashboard</td> </tr> <tr> <td>2.</td> <td>View current stock levels</td> </tr> <tr> <td>3.</td> <td>Select products to update</td> </tr> <tr> <td>4.</td> <td>Enter new quantities</td> </tr> <tr> <td>5.</td> <td>Set low stock alerts and submit changes .</td> </tr> <tr> <td>6.</td> <td>System updates database</td> </tr> </table>	1.	Access inventory dashboard	2.	View current stock levels	3.	Select products to update	4.	Enter new quantities	5.	Set low stock alerts and submit changes .	6.	System updates database
1.	Access inventory dashboard												
2.	View current stock levels												
3.	Select products to update												
4.	Enter new quantities												
5.	Set low stock alerts and submit changes .												
6.	System updates database												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Invalid quantity</td> </tr> <tr> <td></td> <td>1.1.a. Show error message</td> </tr> <tr> <td>2.1</td> <td>Update conflict</td> </tr> <tr> <td></td> <td>2.1.a. Resolve conflicts</td> </tr> <tr> <td>3.1</td> <td>Database error</td> </tr> <tr> <td></td> <td>3.1.a. Retry update</td> </tr> </table>	1.1	Invalid quantity		1.1.a. Show error message	2.1	Update conflict		2.1.a. Resolve conflicts	3.1	Database error		3.1.a. Retry update
1.1	Invalid quantity												
	1.1.a. Show error message												
2.1	Update conflict												
	2.1.a. Resolve conflicts												
3.1	Database error												
	3.1.a. Retry update												
Quality Requirements	Real-time inventory synchronization within 1 minute												

Case Description-13: View Analytics

Use Case	View Analytics
Goal	Users can view business performance metrics
Precondition	User must be logged in as vendor or admin
Success End Condition	Analytics data displayed successfully
Failed End Condition	Unable to load analytics
Primary Actors:	Vendor, Admin
Secondary Actors:	System

Trigger	User accesses analytics dashboard												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Select date range</td> </tr> <tr> <td>2.</td> <td>Choose metrics to view</td> </tr> <tr> <td>3.</td> <td>System generates reports</td> </tr> <tr> <td>4.</td> <td>Display visualizations</td> </tr> <tr> <td>5.</td> <td>Option to export data</td> </tr> <tr> <td>6.</td> <td>View detailed breakdowns</td> </tr> </table>	1.	Select date range	2.	Choose metrics to view	3.	System generates reports	4.	Display visualizations	5.	Option to export data	6.	View detailed breakdowns
1.	Select date range												
2.	Choose metrics to view												
3.	System generates reports												
4.	Display visualizations												
5.	Option to export data												
6.	View detailed breakdowns												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Invalid date range</td> </tr> <tr> <td></td> <td>1.1.a. Request valid dates</td> </tr> <tr> <td>2.1</td> <td>Data unavailable</td> </tr> <tr> <td></td> <td>2.1.a Show partial data</td> </tr> <tr> <td>3.1</td> <td>Export fails</td> </tr> <tr> <td></td> <td>3.1.a Retry export</td> </tr> </table>	1.1	Invalid date range		1.1.a. Request valid dates	2.1	Data unavailable		2.1.a Show partial data	3.1	Export fails		3.1.a Retry export
1.1	Invalid date range												
	1.1.a. Request valid dates												
2.1	Data unavailable												
	2.1.a Show partial data												
3.1	Export fails												
	3.1.a Retry export												
Quality Requirements	Analytics should load within 10 seconds												

Case Description-14: Manage Users (Admin)

Use Case	Manage Users
Goal	Admin can view, edit, and manage all system users
Precondition	User must be logged in as admin
Success End Condition	User management action completed successfully
Failed End Condition	User management action failed
Primary Actors:	Admin
Secondary Actors:	None

Trigger	Admin selects "User Management" option												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access user management dashboard</td> </tr> <tr> <td>2.</td> <td>View list of all users</td> </tr> <tr> <td>3.</td> <td>Search/filter users to select to manage</td> </tr> <tr> <td>4.</td> <td>Perform action (edit/block/delete)</td> </tr> <tr> <td>5.</td> <td>Confirm action and System updates user status</td> </tr> <tr> <td>6.</td> <td>Notification sent to affected user</td> </tr> </table>	1.	Access user management dashboard	2.	View list of all users	3.	Search/filter users to select to manage	4.	Perform action (edit/block/delete)	5.	Confirm action and System updates user status	6.	Notification sent to affected user
1.	Access user management dashboard												
2.	View list of all users												
3.	Search/filter users to select to manage												
4.	Perform action (edit/block/delete)												
5.	Confirm action and System updates user status												
6.	Notification sent to affected user												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>No matching users</td> </tr> <tr> <td></td> <td>1.1.a. Show "No results found"</td> </tr> <tr> <td>2.1</td> <td>Invalid action</td> </tr> <tr> <td></td> <td>2.1.a Show error message</td> </tr> <tr> <td>3.1</td> <td>Update fails</td> </tr> <tr> <td></td> <td>3.1.a. Retry update.</td> </tr> </table>	1.1	No matching users		1.1.a. Show "No results found"	2.1	Invalid action		2.1.a Show error message	3.1	Update fails		3.1.a. Retry update.
1.1	No matching users												
	1.1.a. Show "No results found"												
2.1	Invalid action												
	2.1.a Show error message												
3.1	Update fails												
	3.1.a. Retry update.												
Quality Requirements	User actions must be logged for audit purposes												

Case Description-15: Manage Categories

Use Case	Manage Categories
Goal	Admin can create and manage product categories
Precondition	Login as admin
Success End Condition	Category successfully created/updated/deleted
Failed End Condition	Category operation failed
Primary Actors:	Admin
Secondary Actors:	None

Trigger	Admin selects "Category Management" option												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Access category management</td> </tr> <tr> <td>2.</td> <td>View category hierarchy</td> </tr> <tr> <td>3.</td> <td>Select operation (add/edit/delete)</td> </tr> <tr> <td>4.</td> <td>Enter category details</td> </tr> <tr> <td>5.</td> <td>Set parent category if applicable</td> </tr> <tr> <td>6.</td> <td>Submit changes and update category structure</td> </tr> </table>	1.	Access category management	2.	View category hierarchy	3.	Select operation (add/edit/delete)	4.	Enter category details	5.	Set parent category if applicable	6.	Submit changes and update category structure
1.	Access category management												
2.	View category hierarchy												
3.	Select operation (add/edit/delete)												
4.	Enter category details												
5.	Set parent category if applicable												
6.	Submit changes and update category structure												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Duplicate category</td> </tr> <tr> <td></td> <td>1.1.a Request unique name</td> </tr> <tr> <td>2.1</td> <td>Products in deleted category</td> </tr> <tr> <td></td> <td>2.1.a Reassign products prompt</td> </tr> <tr> <td>3.1</td> <td>Update error</td> </tr> <tr> <td></td> <td>3.1.a Rollback changes</td> </tr> </table>	1.1	Duplicate category		1.1.a Request unique name	2.1	Products in deleted category		2.1.a Reassign products prompt	3.1	Update error		3.1.a Rollback changes
1.1	Duplicate category												
	1.1.a Request unique name												
2.1	Products in deleted category												
	2.1.a Reassign products prompt												
3.1	Update error												
	3.1.a Rollback changes												
Quality Requirements	Maximum of 3 hierarchy levels allowed												

Case Description-16: Moderate Reviews

Use Case	Moderate Reviews
Goal	Admin can review and moderate user product reviews
Precondition	User must be logged in as admin
Success End Condition	Review moderated successfully
Failed End Condition	Moderation action failed
Primary Actors:	Admin
Secondary Actors:	Customer

Trigger	Admin accesses review moderation queue												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>View pending reviews</td> </tr> <tr> <td>2.</td> <td>Select review to moderate and read content</td> </tr> <tr> <td>3.</td> <td>Check against guidelines</td> </tr> <tr> <td>4.</td> <td>Choose action (approve/reject/edit)</td> </tr> <tr> <td>5.</td> <td>Add moderation note if needed</td> </tr> <tr> <td>6.</td> <td>Submit decision and notify review author </td> </tr> </table>	1.	View pending reviews	2.	Select review to moderate and read content	3.	Check against guidelines	4.	Choose action (approve/reject/edit)	5.	Add moderation note if needed	6.	Submit decision and notify review author
1.	View pending reviews												
2.	Select review to moderate and read content												
3.	Check against guidelines												
4.	Choose action (approve/reject/edit)												
5.	Add moderation note if needed												
6.	Submit decision and notify review author												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>No pending reviews</td> </tr> <tr> <td></td> <td>1.1.a Show empty queue message</td> </tr> <tr> <td>2.1</td> <td>Multiple violations</td> </tr> <tr> <td></td> <td>2.1.a Flag user account</td> </tr> <tr> <td>3.1</td> <td>System error</td> </tr> <tr> <td></td> <td>3.1.a Save draft decision</td> </tr> </table>	1.1	No pending reviews		1.1.a Show empty queue message	2.1	Multiple violations		2.1.a Flag user account	3.1	System error		3.1.a Save draft decision
1.1	No pending reviews												
	1.1.a Show empty queue message												
2.1	Multiple violations												
	2.1.a Flag user account												
3.1	System error												
	3.1.a Save draft decision												
Quality Requirements	Reviews must be moderated within 24 hours												

Case Description-17: Generate Reports

Use Case	Generate Reports
Goal	Admin can generate various system reports
Precondition	Login as admin
Success End Condition	Report Generate Successfully
Failed End Condition	Generate failed
Primary Actors:	Admin
Secondary Actors:	System

Trigger	Admin selects "Generate Reports" option	
Description / Main Success Scenario	1.	Select report type
	2.	Choose date range and set filters/parameters
	3.	Select output format and Generate report
	4.	Preview report
	5.	Download or share report
Alternative Flows	1.1	Invalid date range
		1.1.a Request valid dates
	2.1	Insufficient data
		2.1.a Show partial report
	3.1	Download fail
		3.1.a Retry download
Quality Requirements	Reports must be generated within 5 minutes	

2.4.3 Activity Diagram

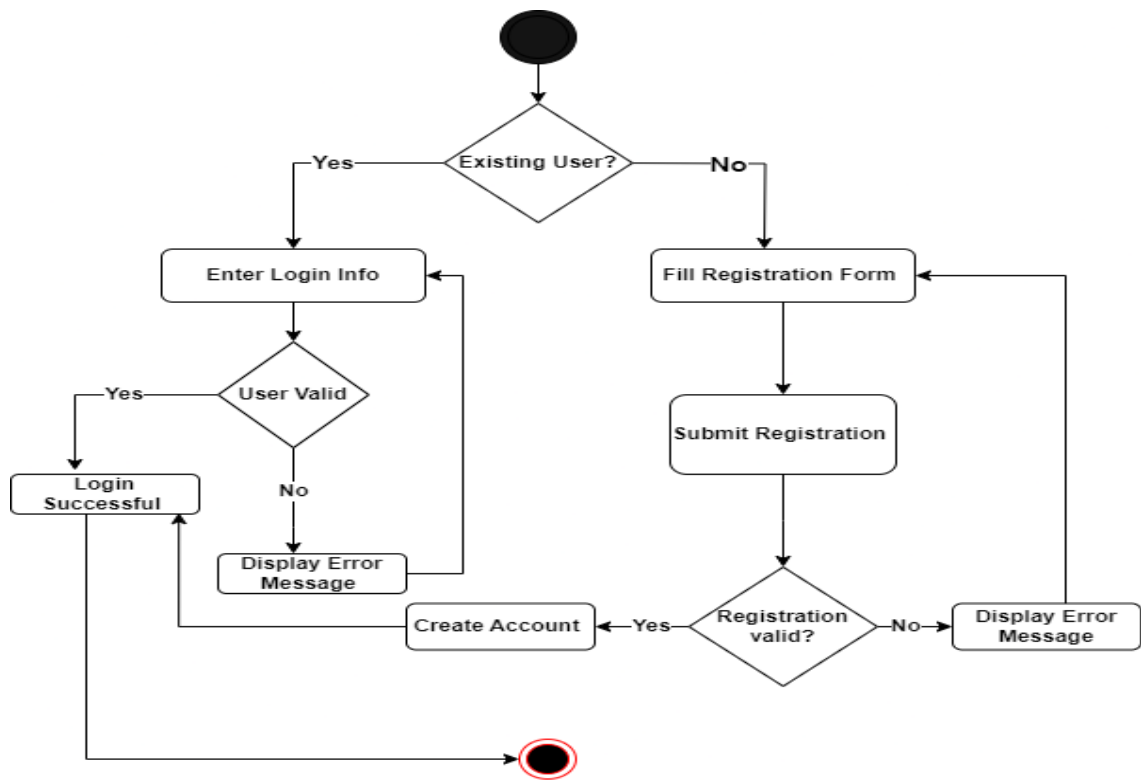


Figure 2.2: Registration / Login

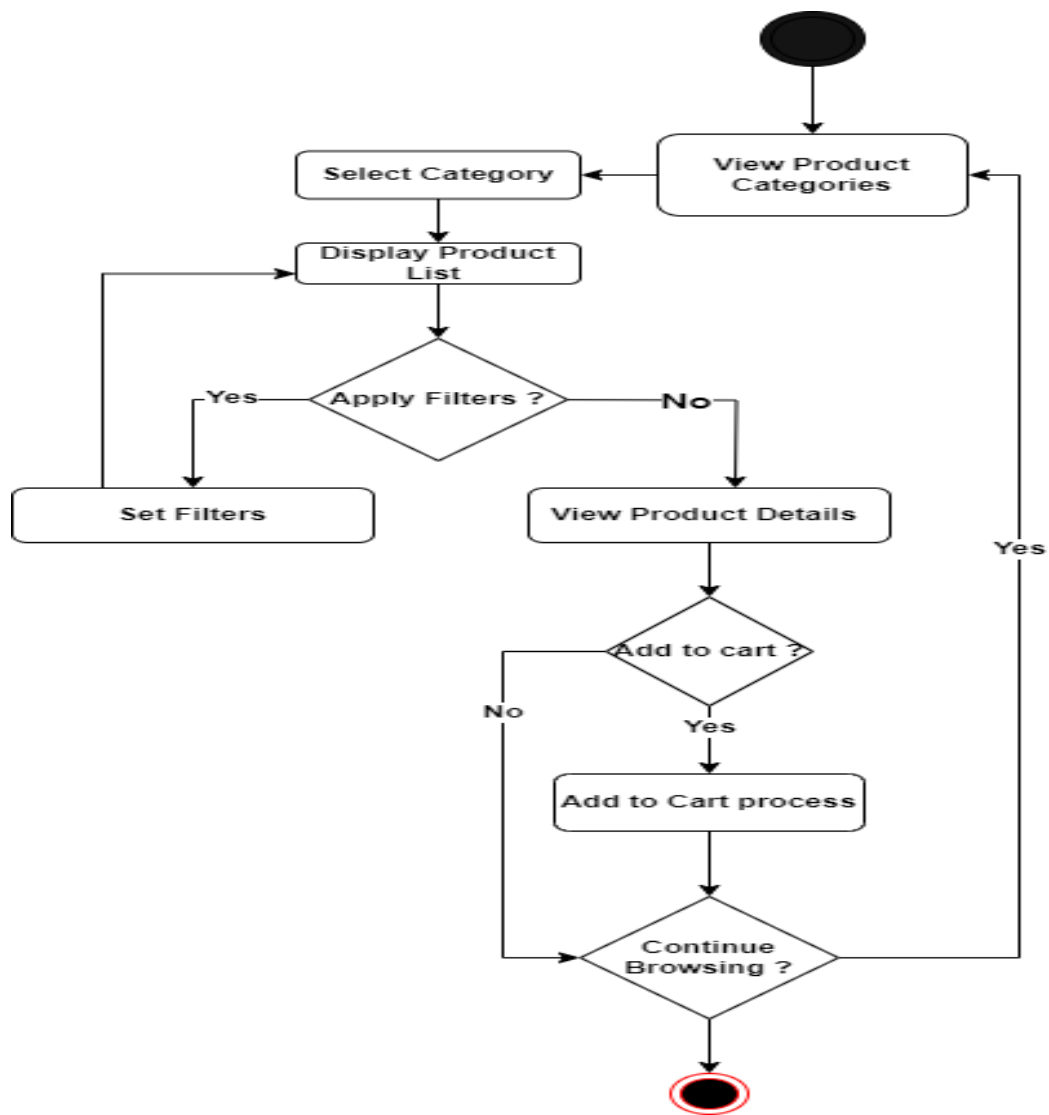


Figure 2.3: Browse Products

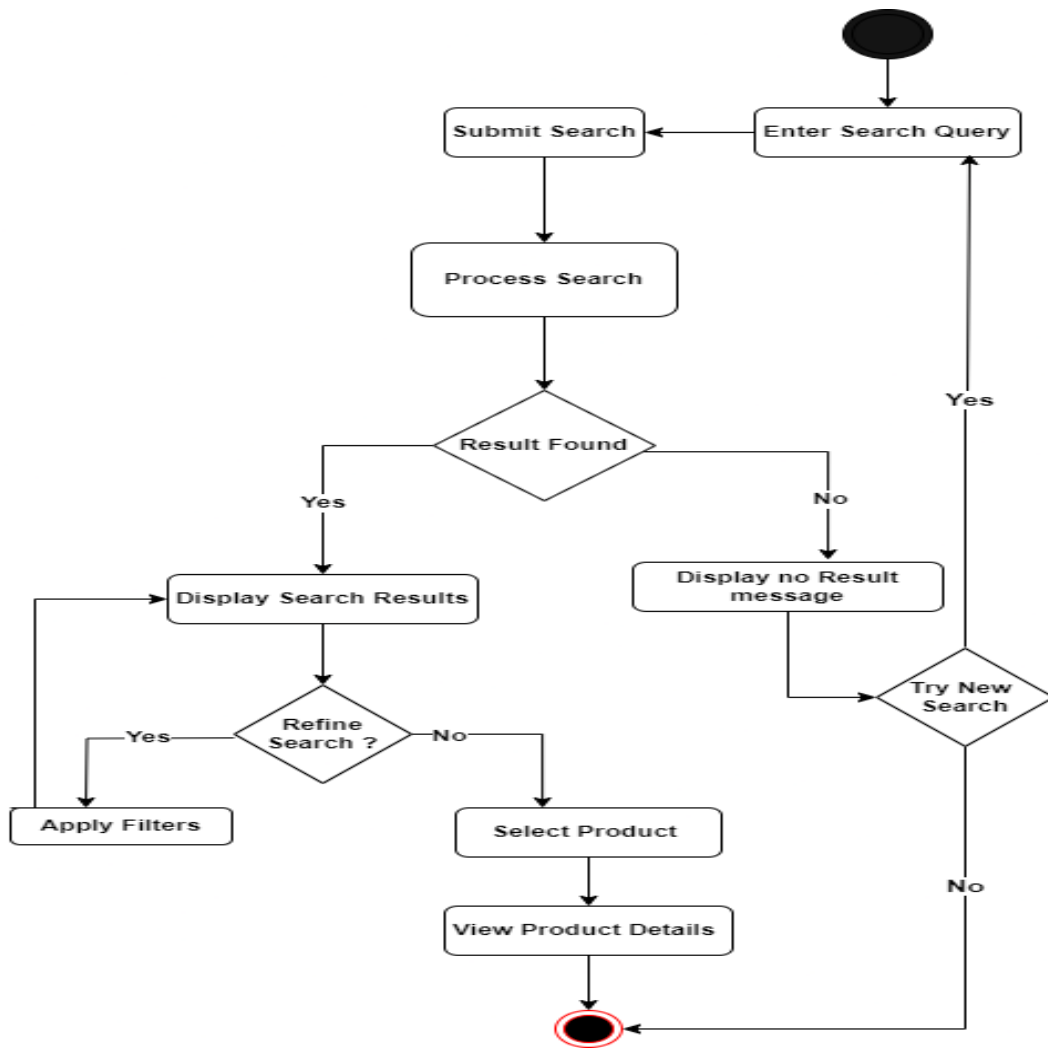


Figure 2.4: Search Products

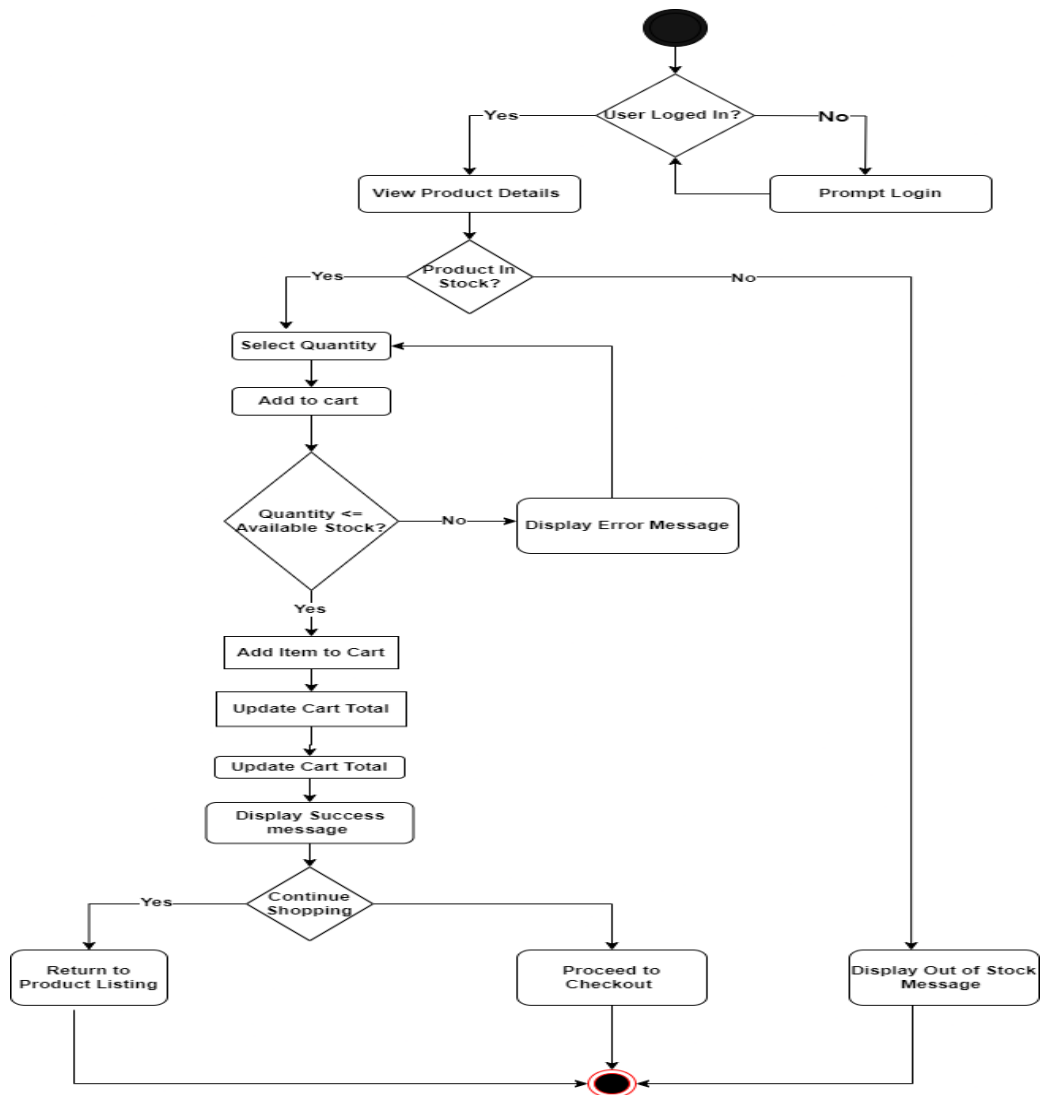


Figure 2.5: Add to Cart

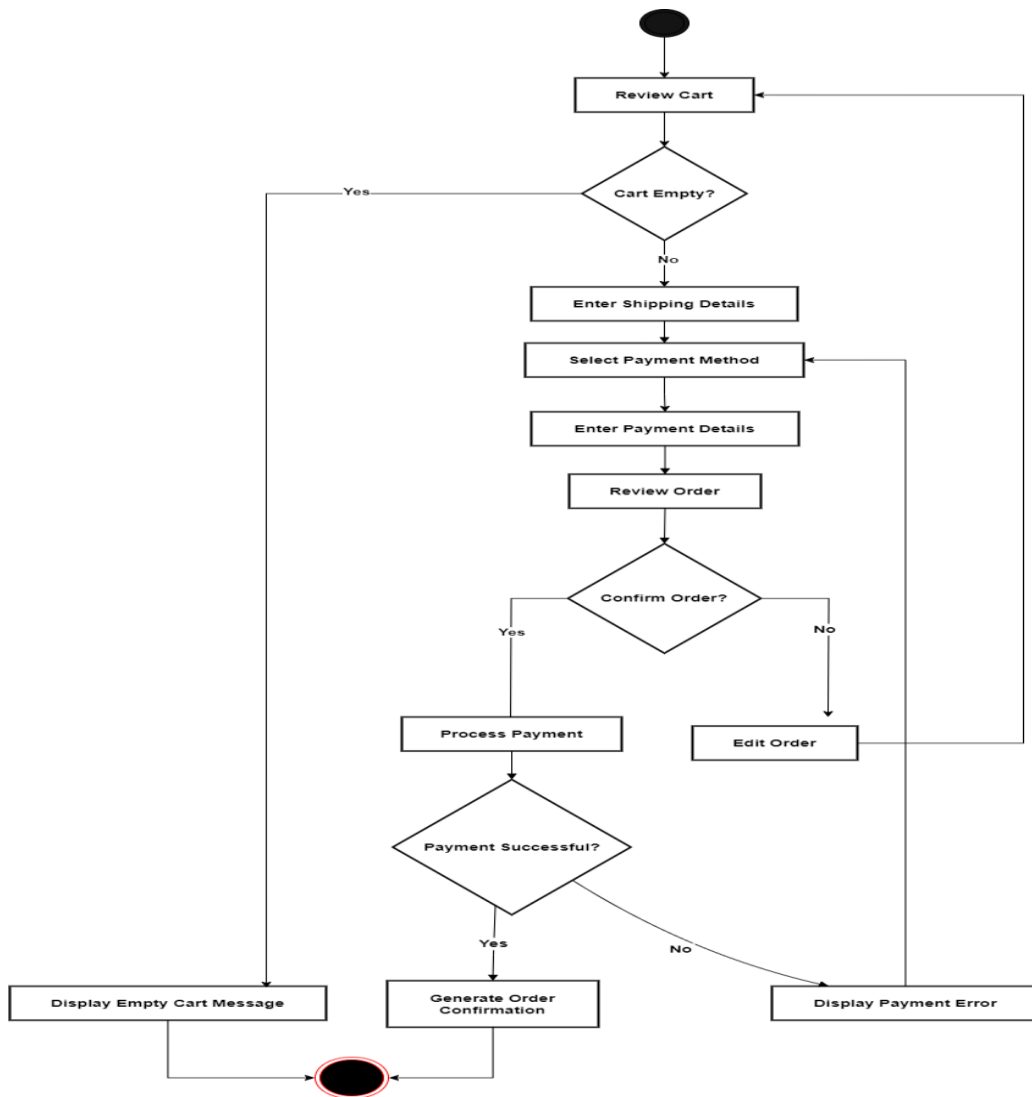


Figure 2.6: Checkout

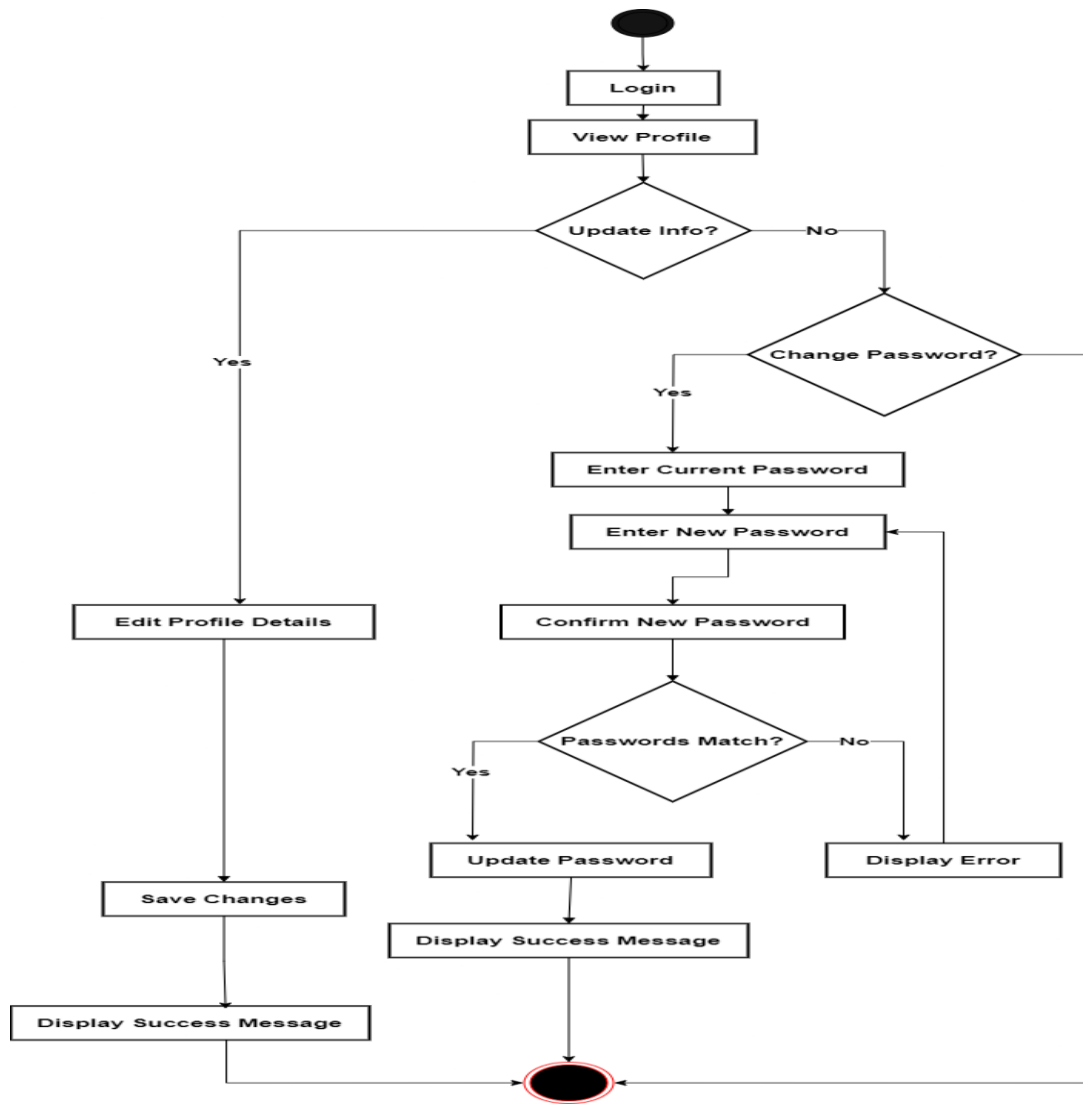


Figure 2.7: Manage Profile

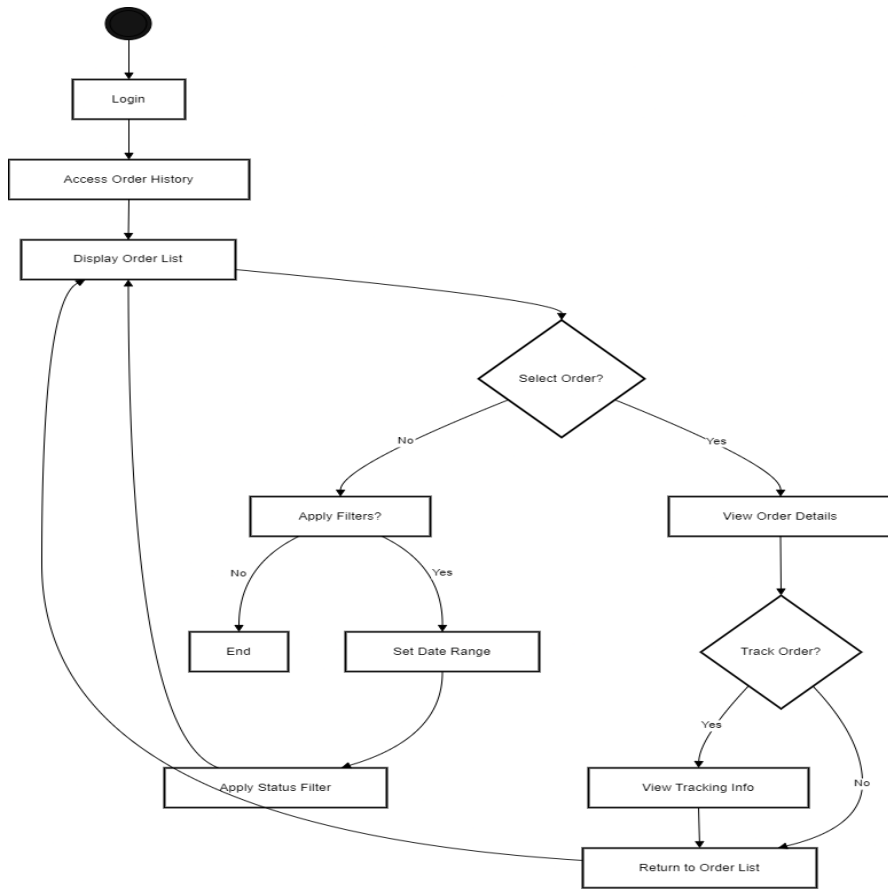


Figure 2.8: For View Order History

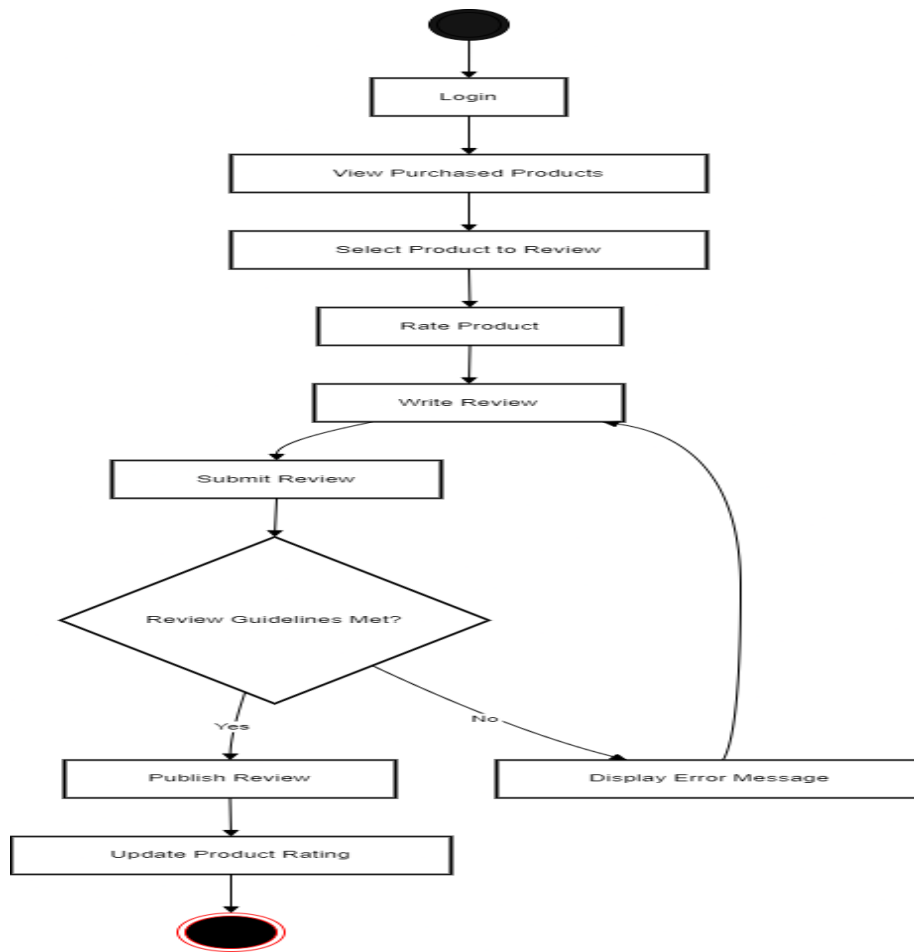


Figure 2.9: Rate and Review

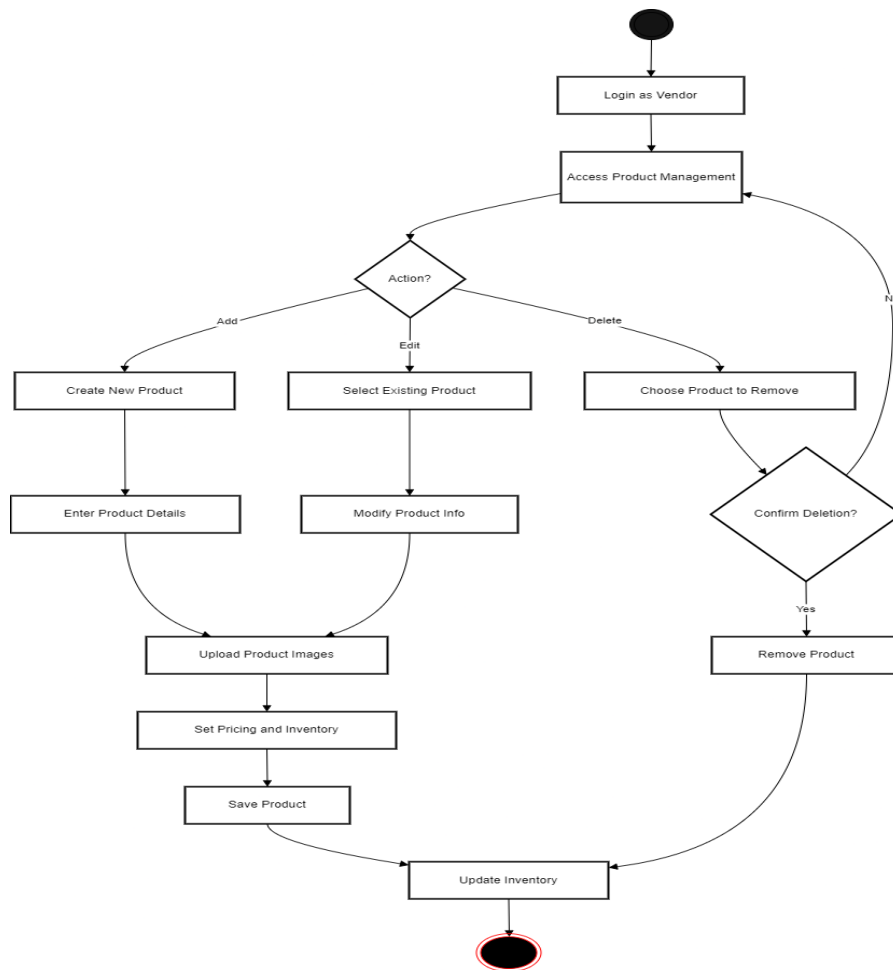


Figure 2.10: Manage Products (Vendor)

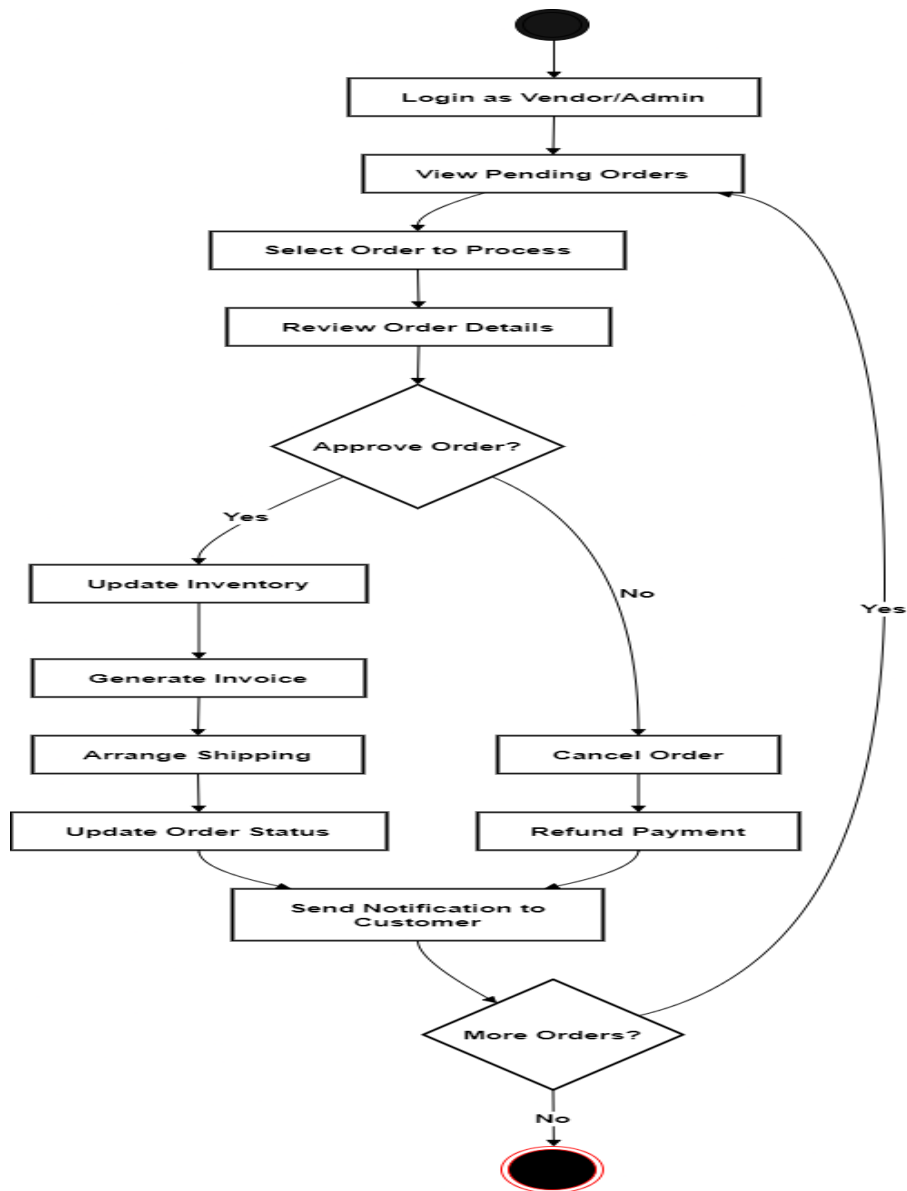


Figure 2.11: Process Orders (Vendor/Admin)

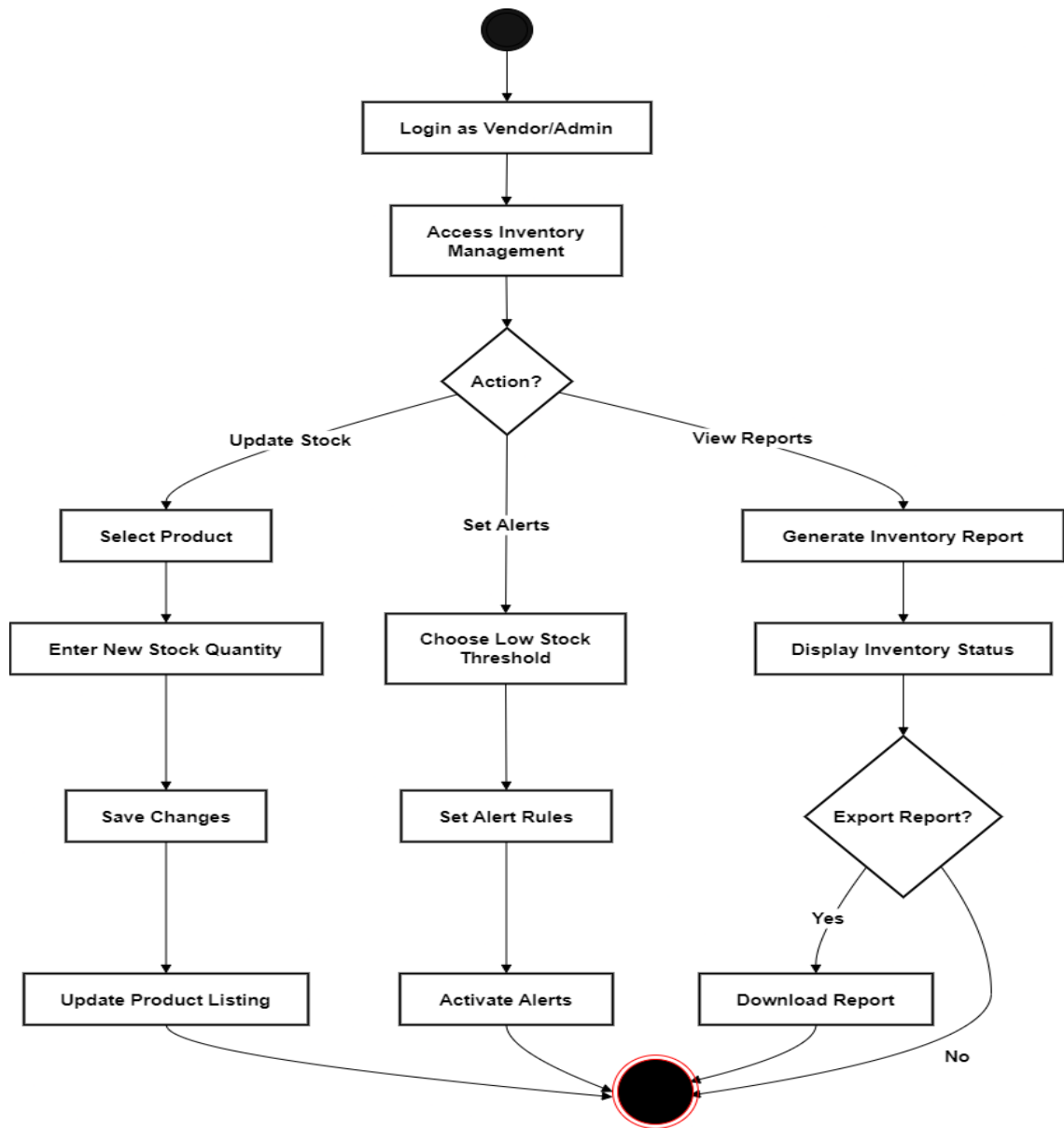


Figure 2.12: Manage Inventory

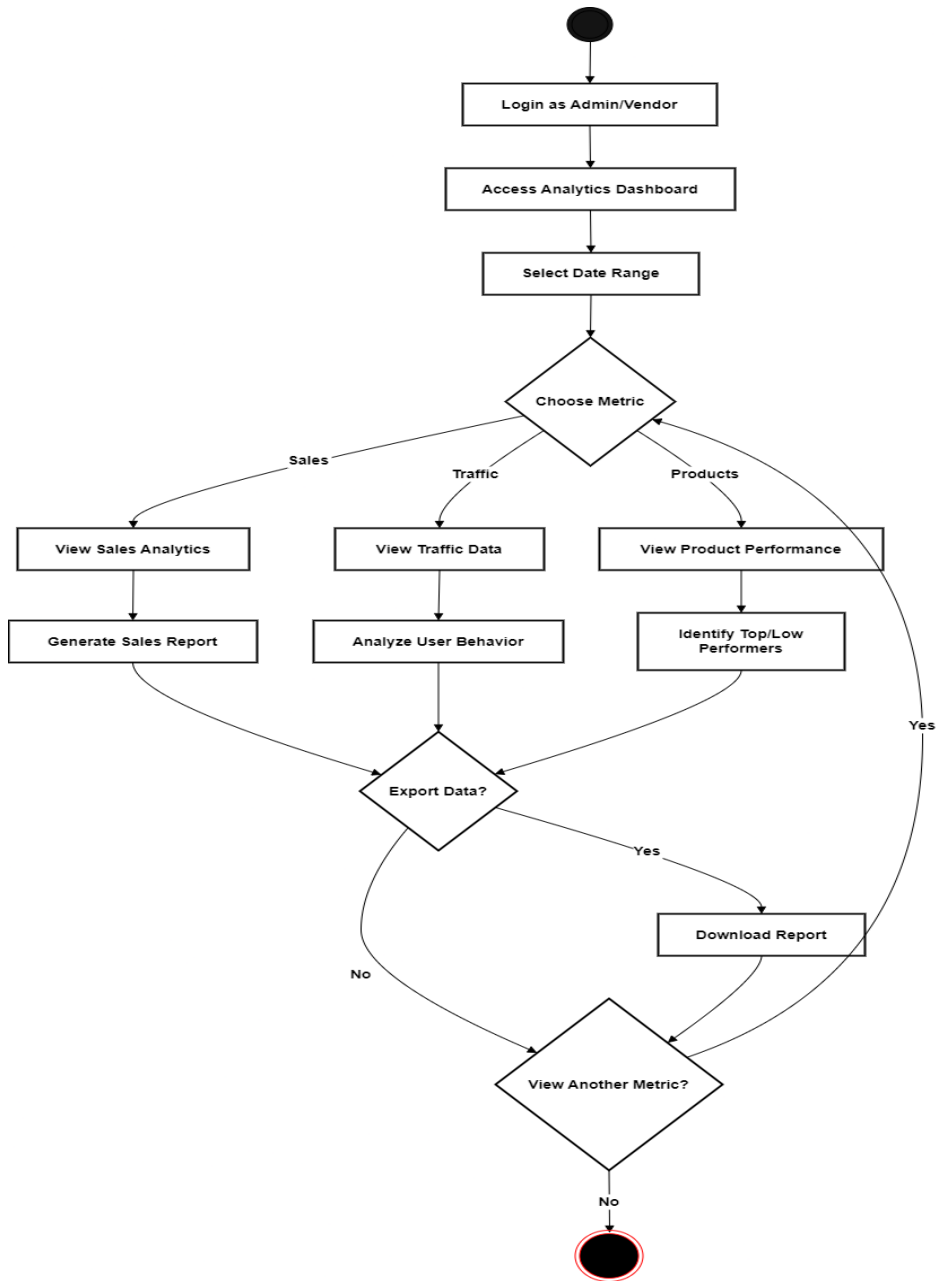


Figure 2.13: View Analytics

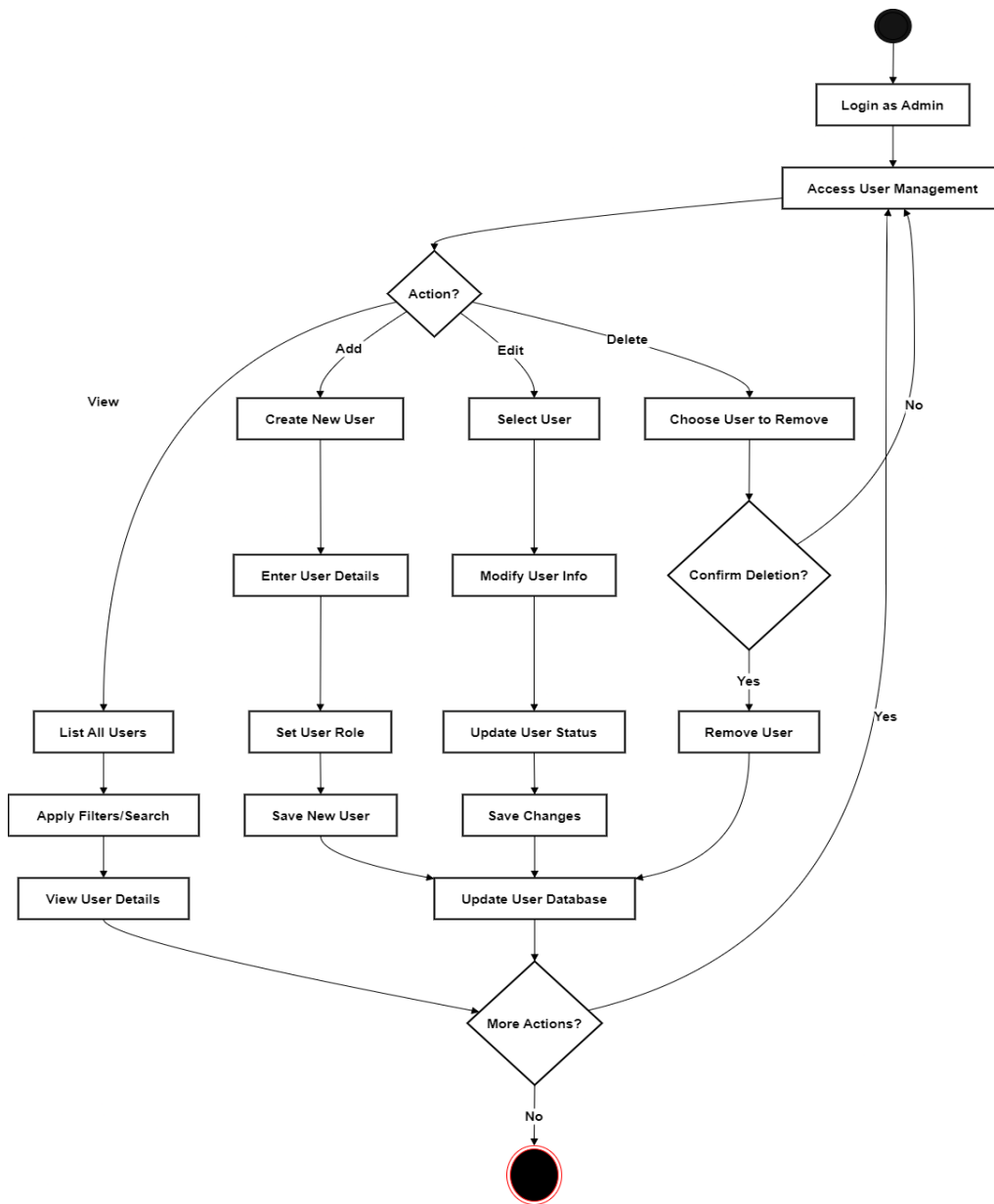


Figure 2.14: Manage Users (Admin)

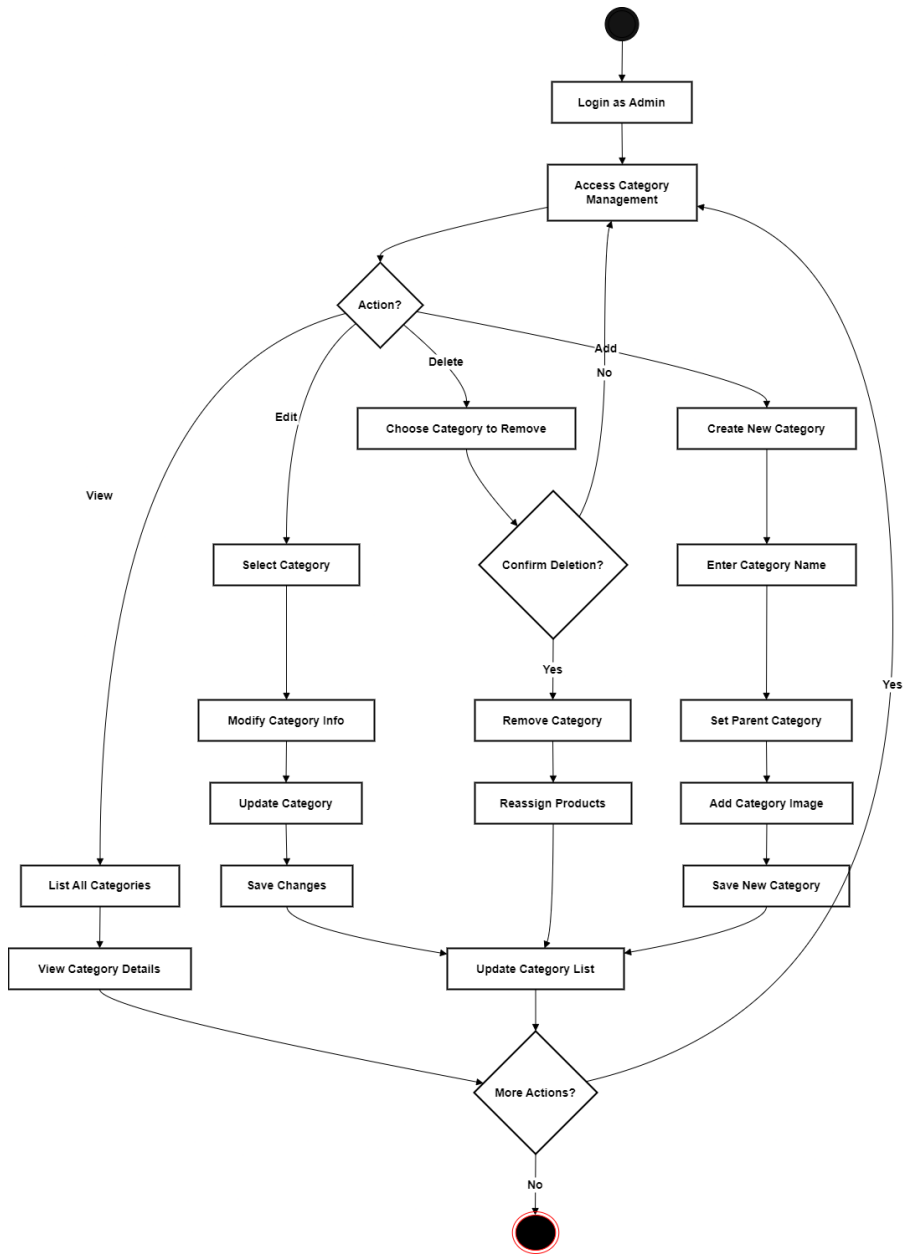


Figure 2.15: Manage Categories

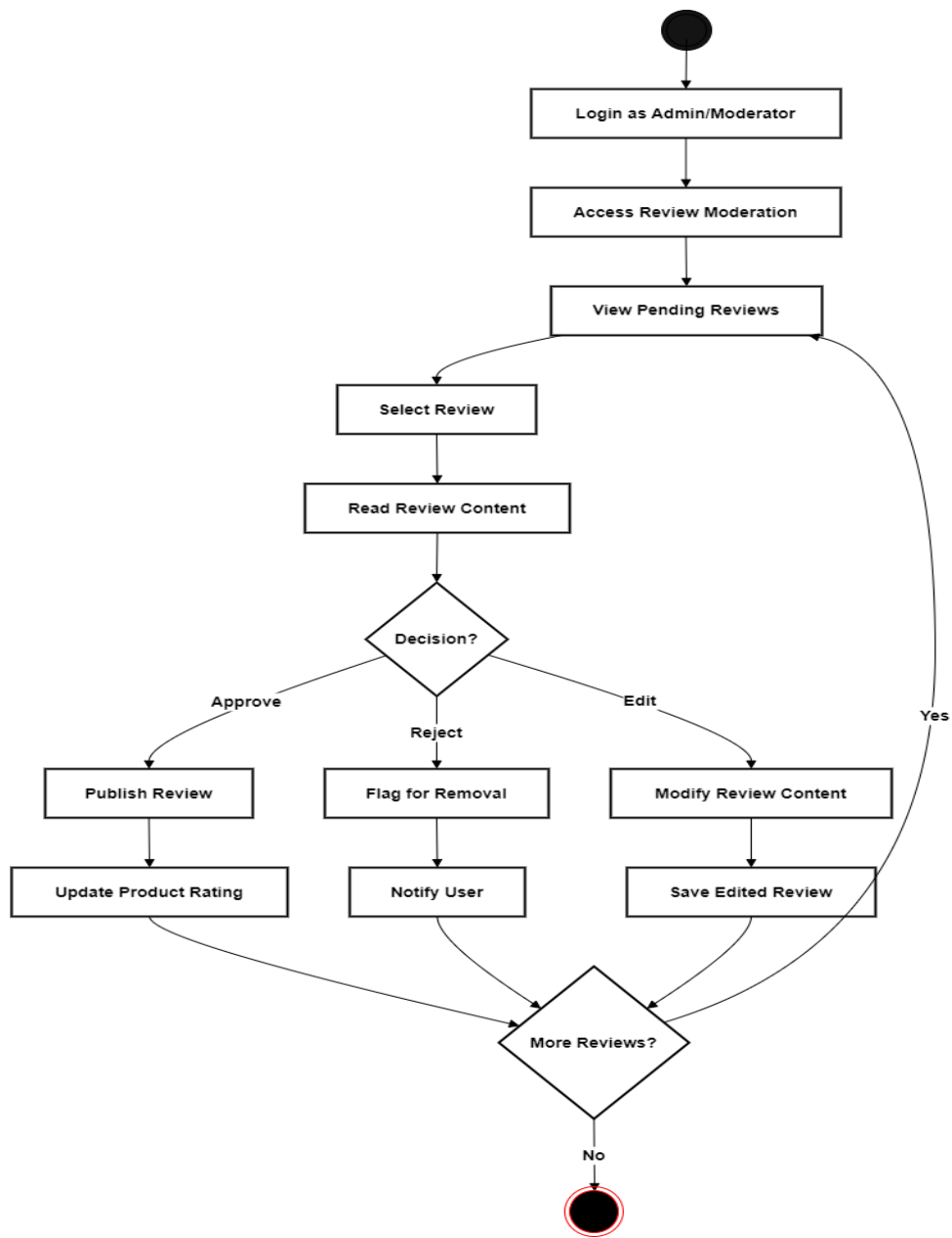


Figure 2.17: Moderate Reviews

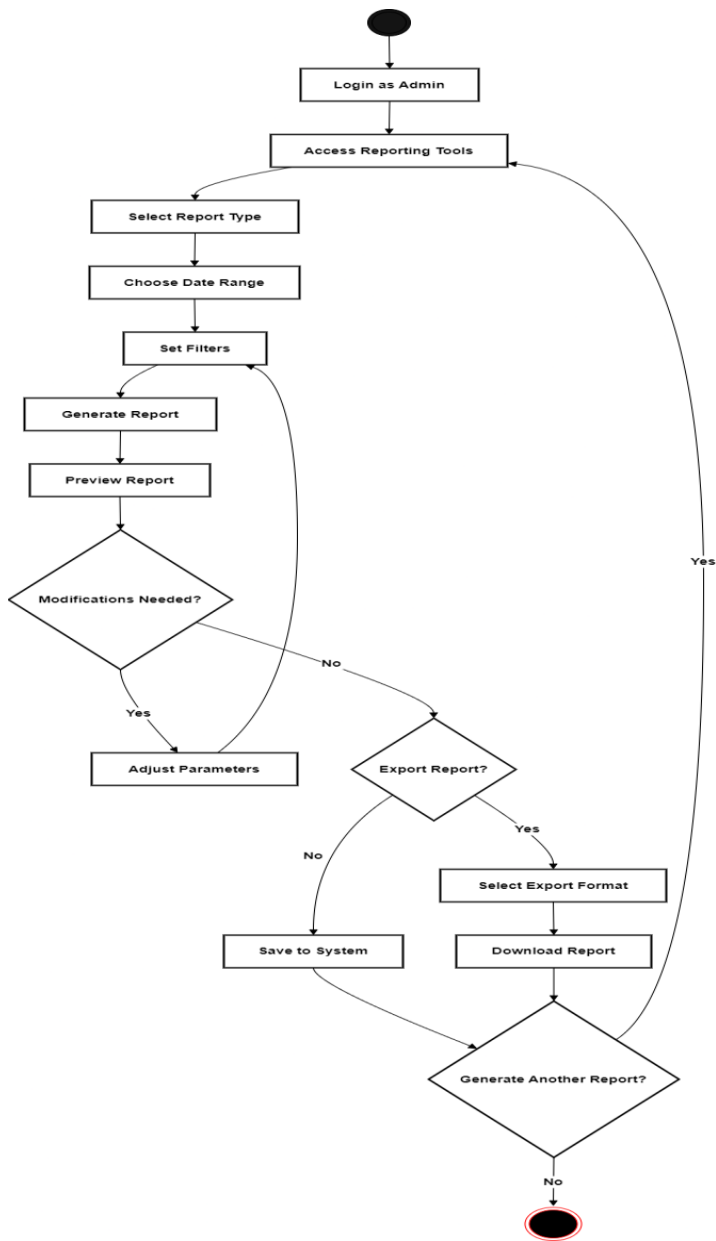


Figure 2.18: Generate Reports

2.4.4 Sequence Diagram

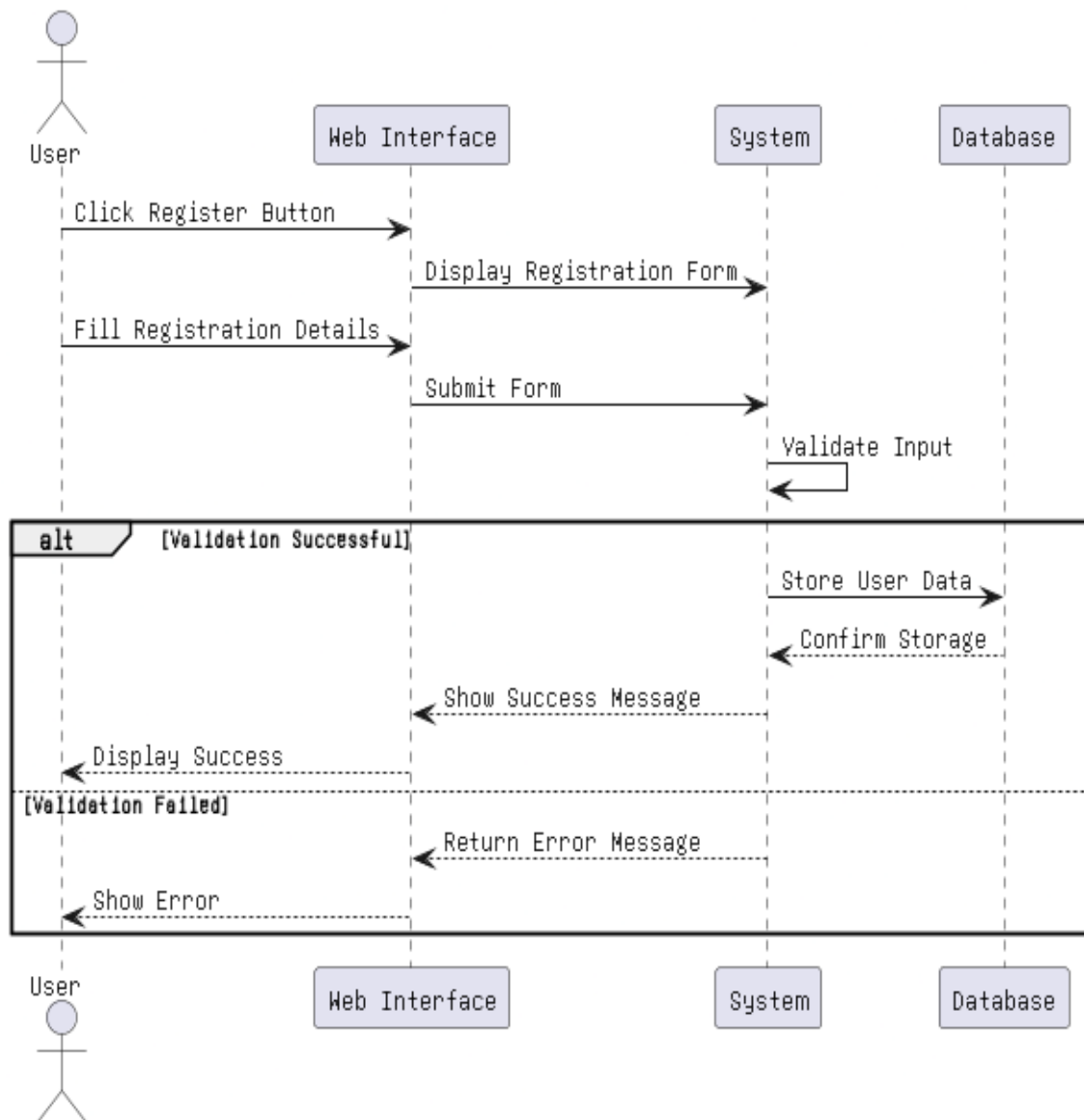


Figure 2.19: Registration

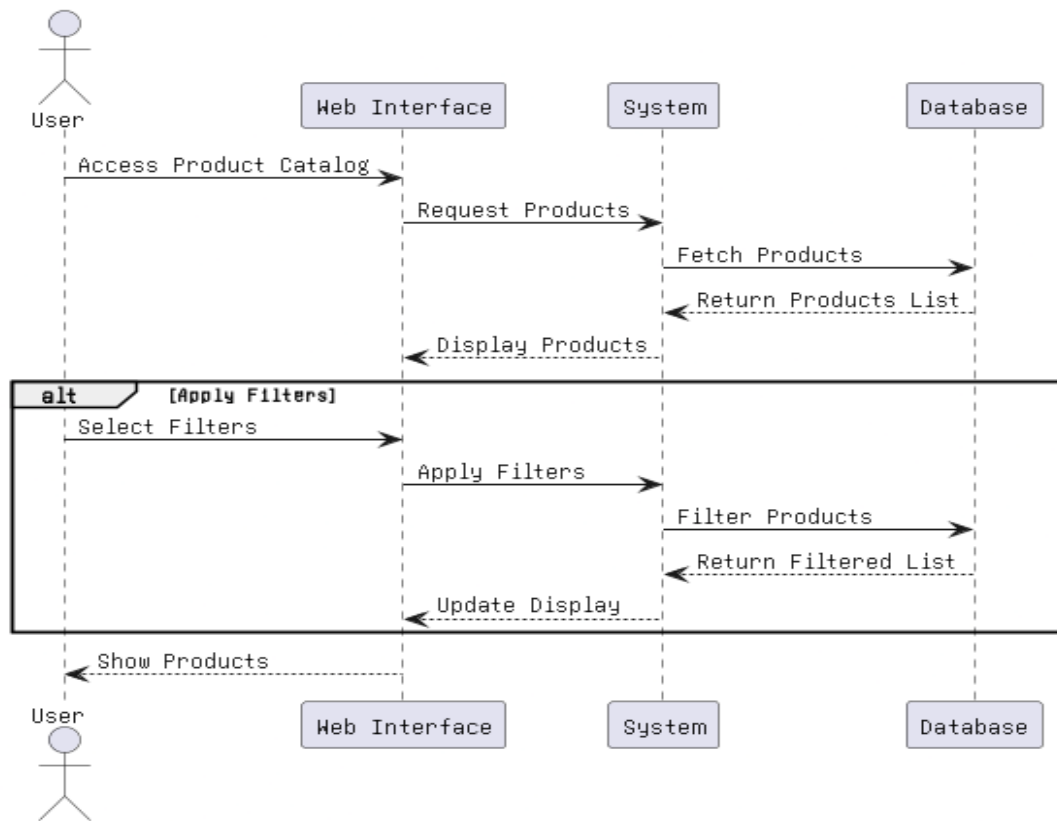


Figure 2.20: Browse Products

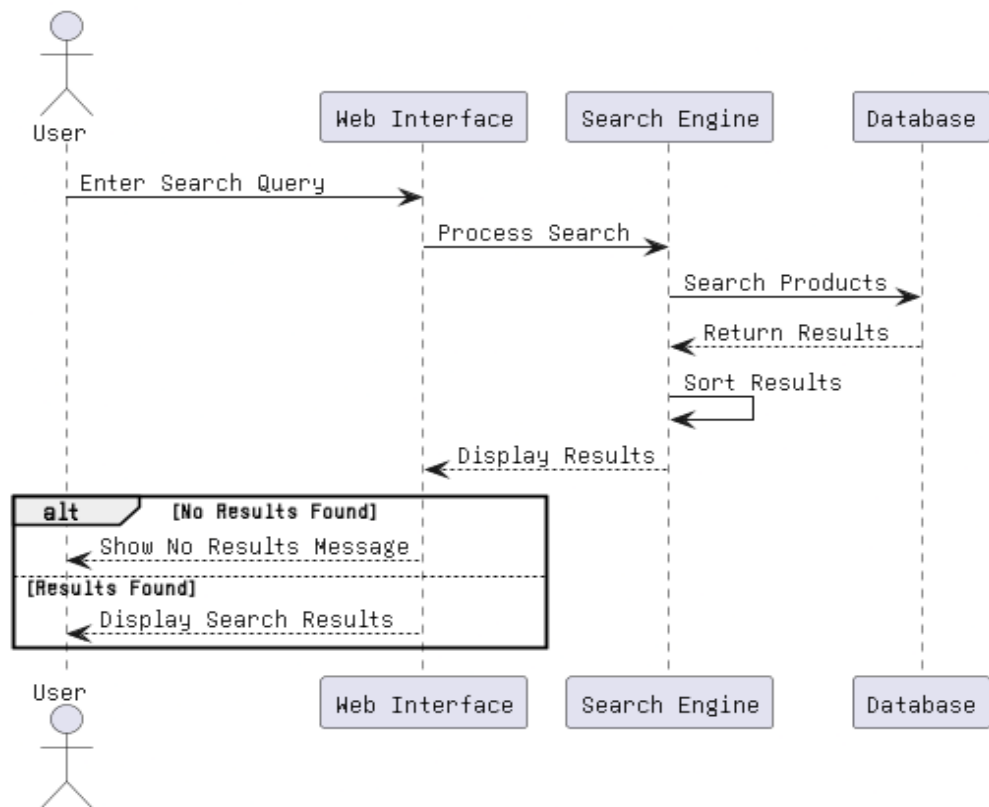


Figure 2.21: Search Product

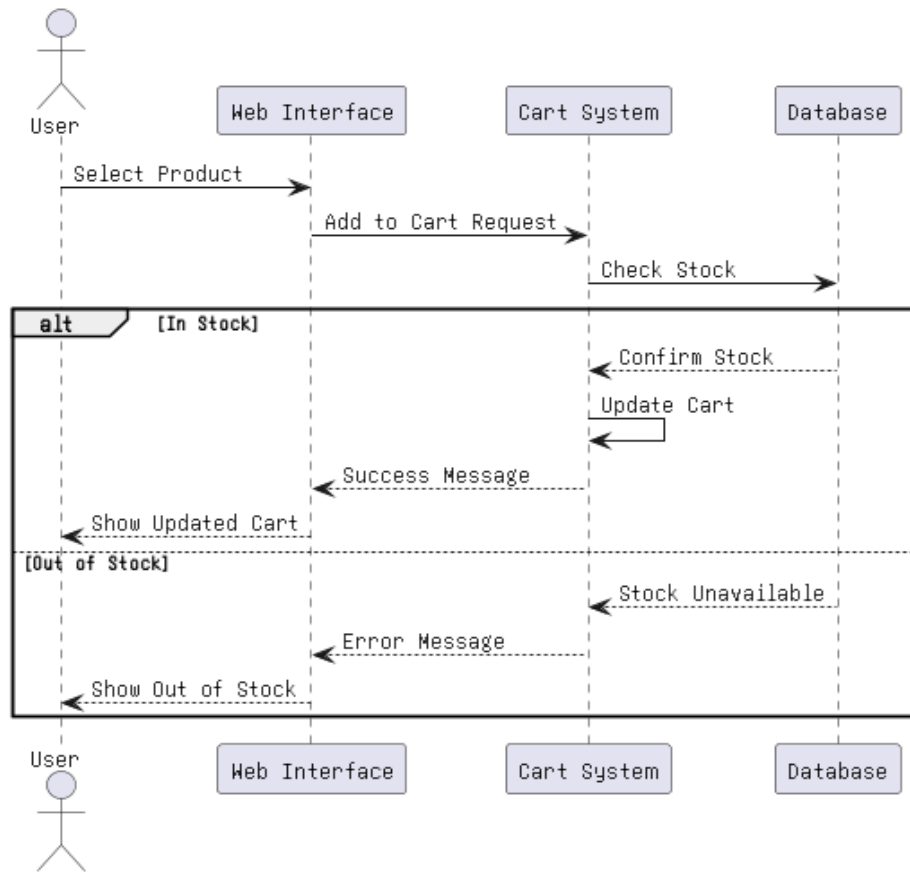


Figure 2.22: Add To Cart

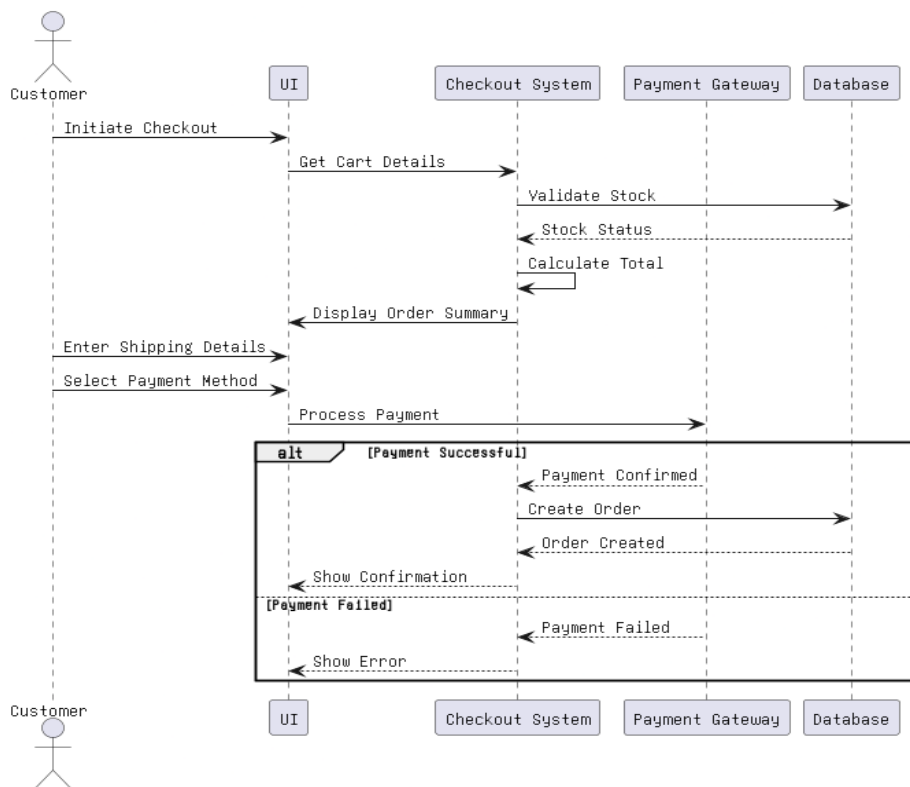


Figure 2.23: Checkout

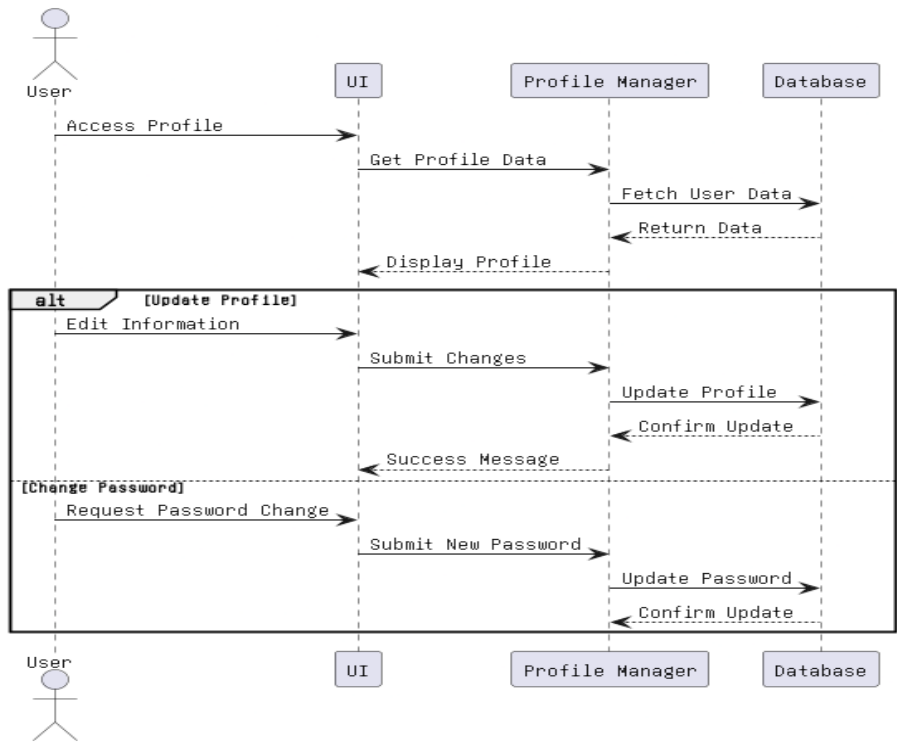


Figure 2.24: Manage Profile

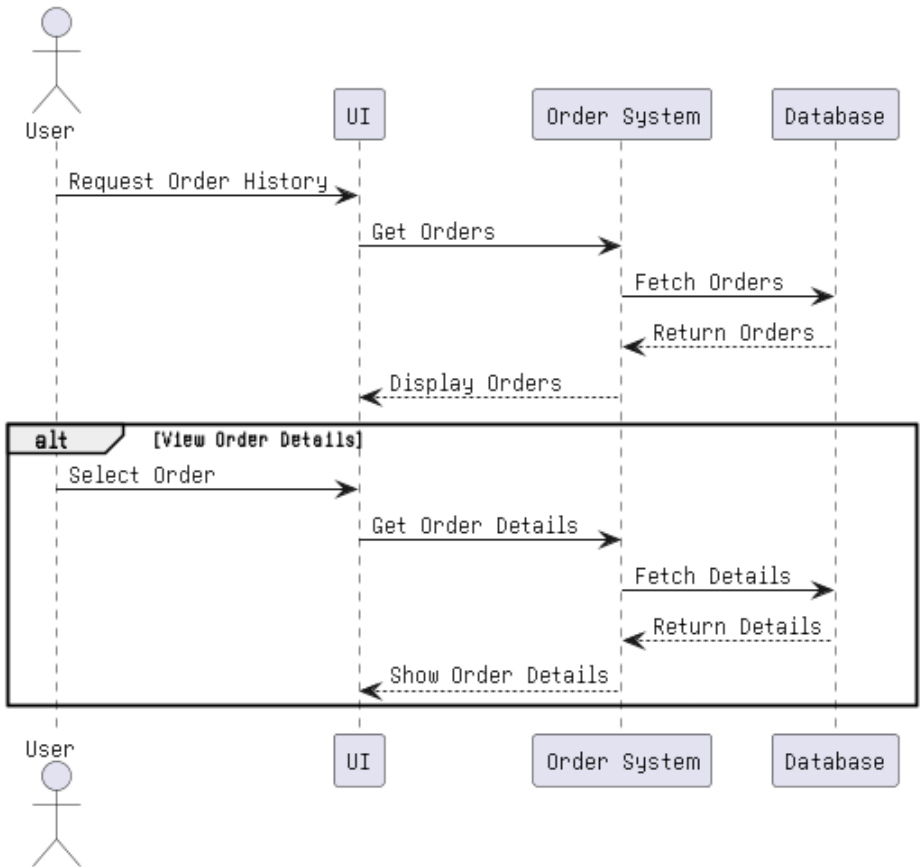


Figure 2.25: View Order History

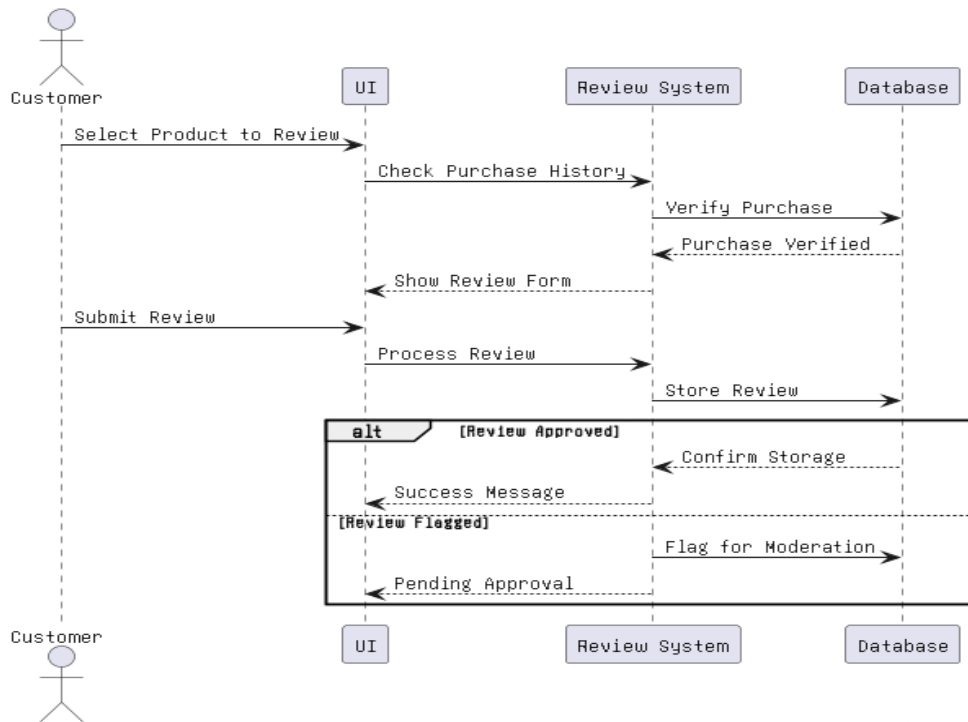


Figure 2.26: Rate and Review

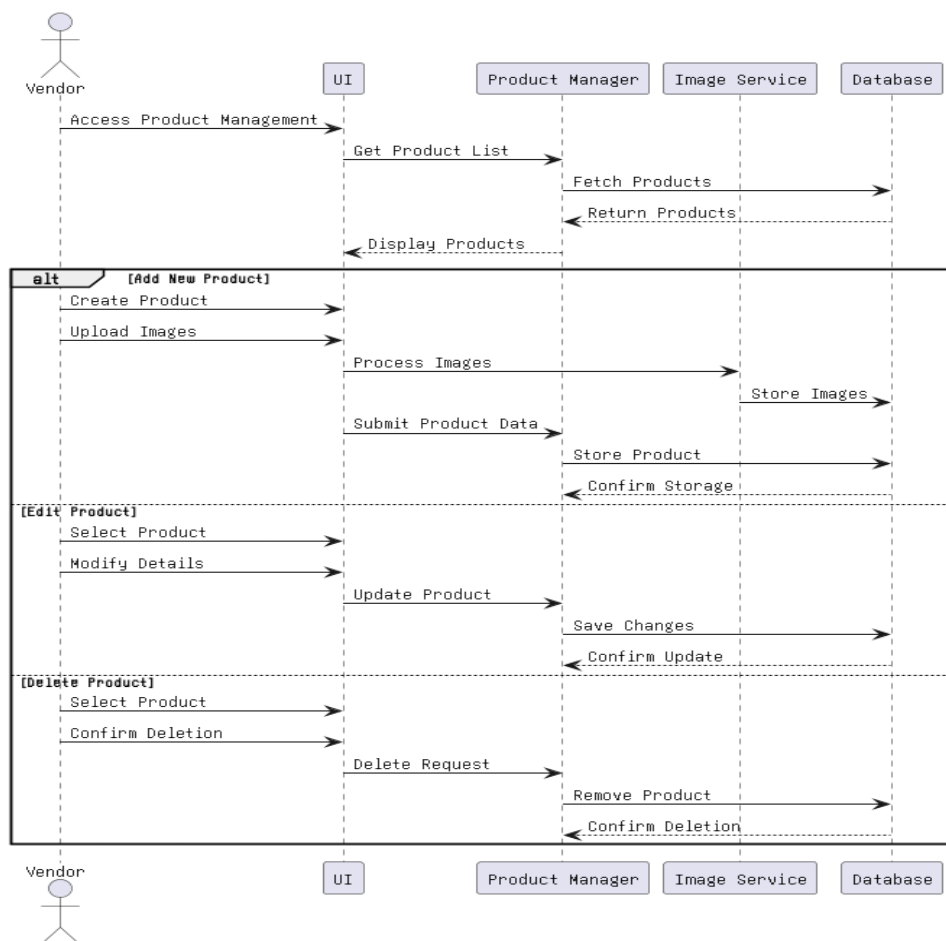


Figure 2.27: Manage Products (Vendor)

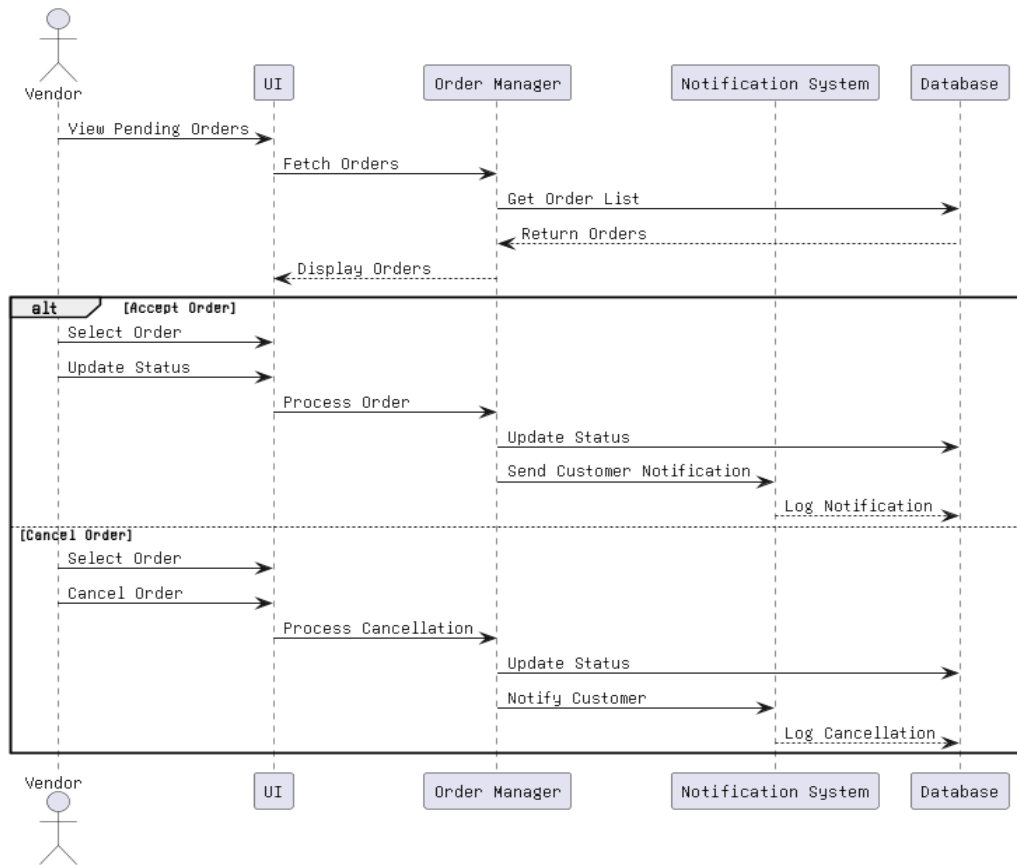


Figure 2.28: Process Orders (Vendor)

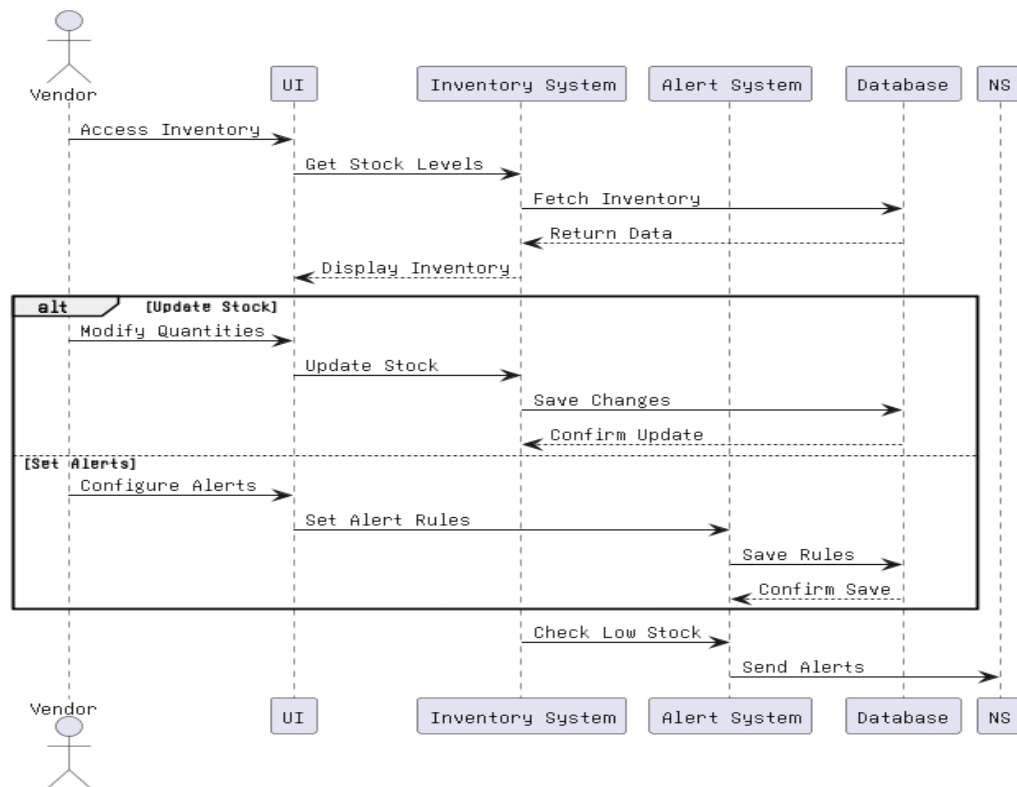


Figure 2.29: Manage Inventory

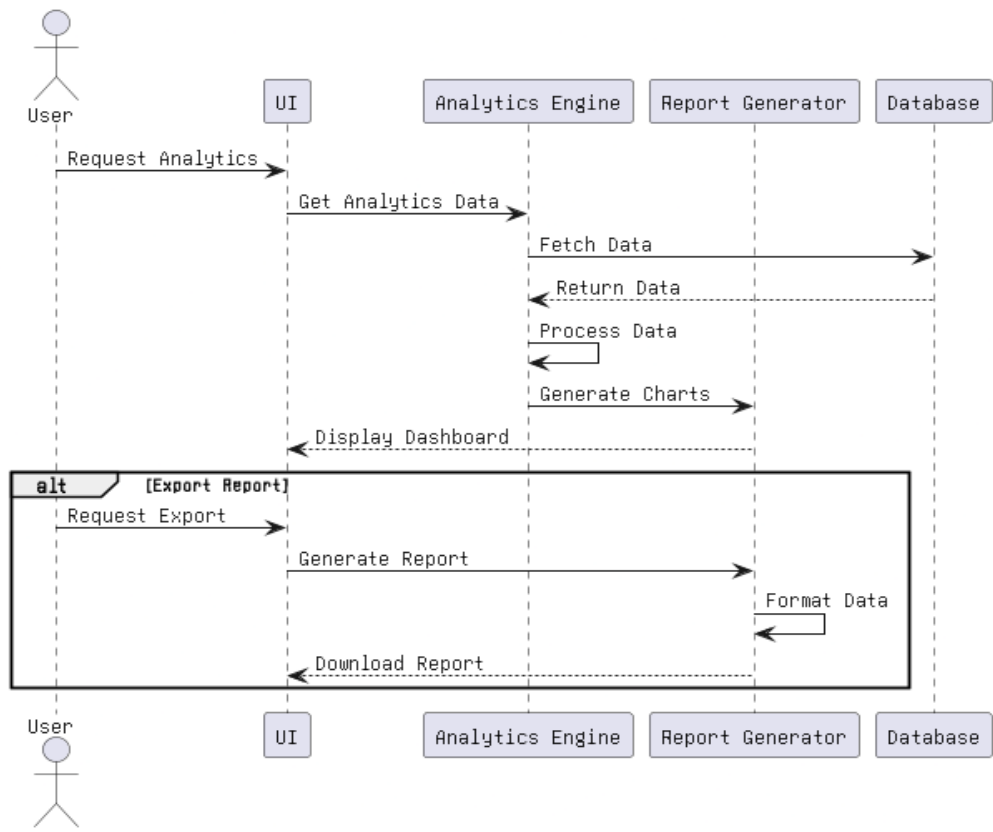


Figure 2.30: View Analytics

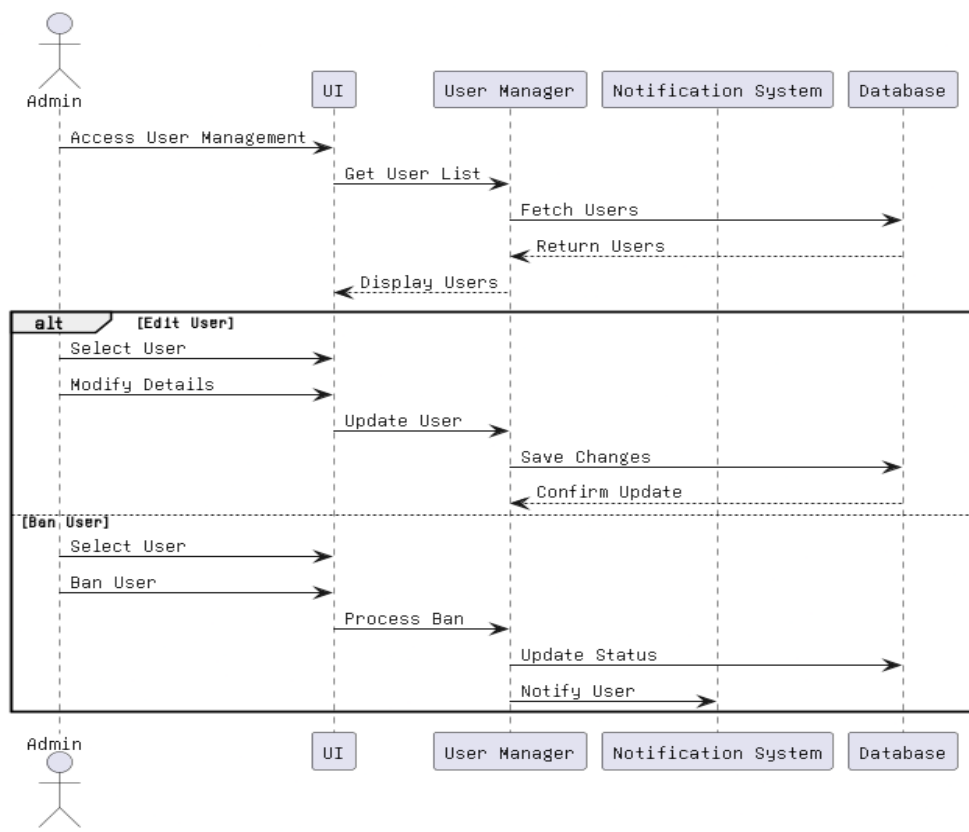


Figure 2.31: Manage Users (Admin)

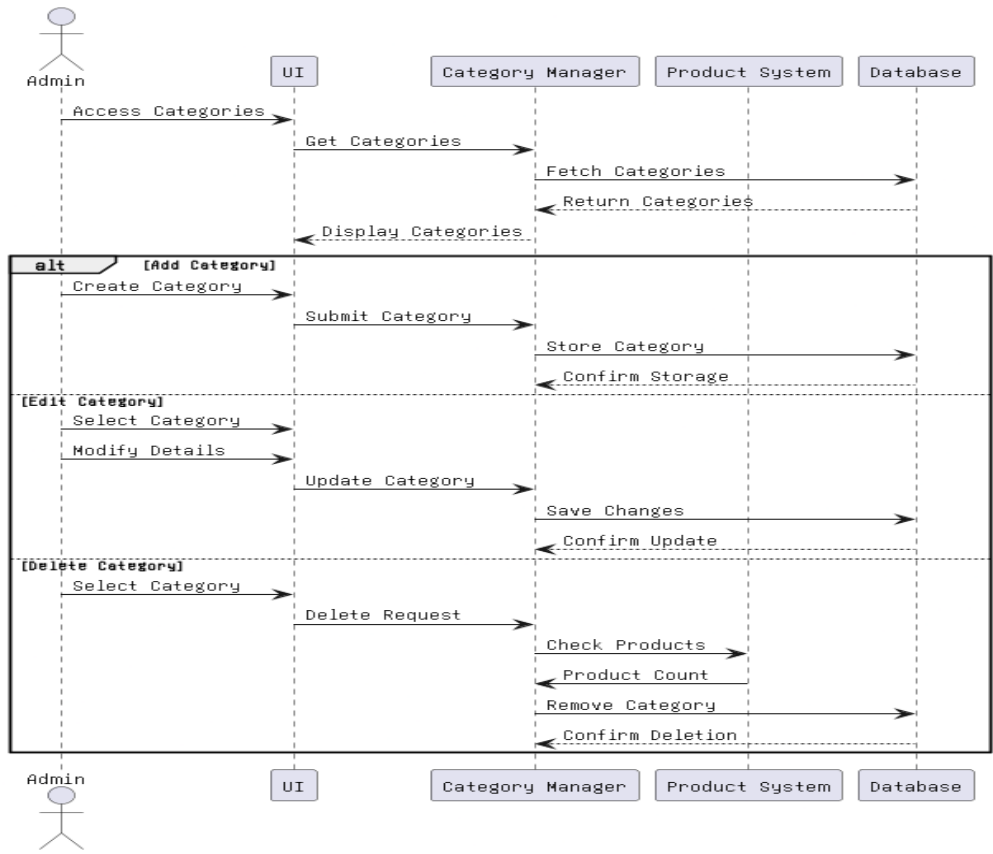


Figure 2.32: Manage Categories

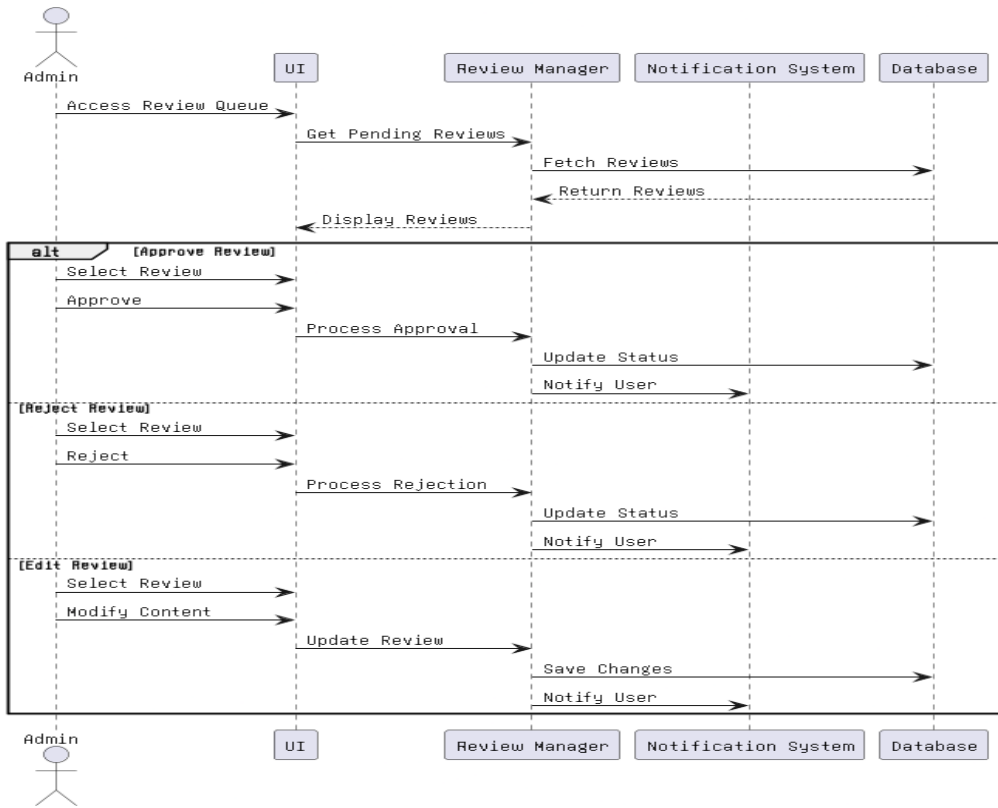


Figure 2.33: Moderate Reviews

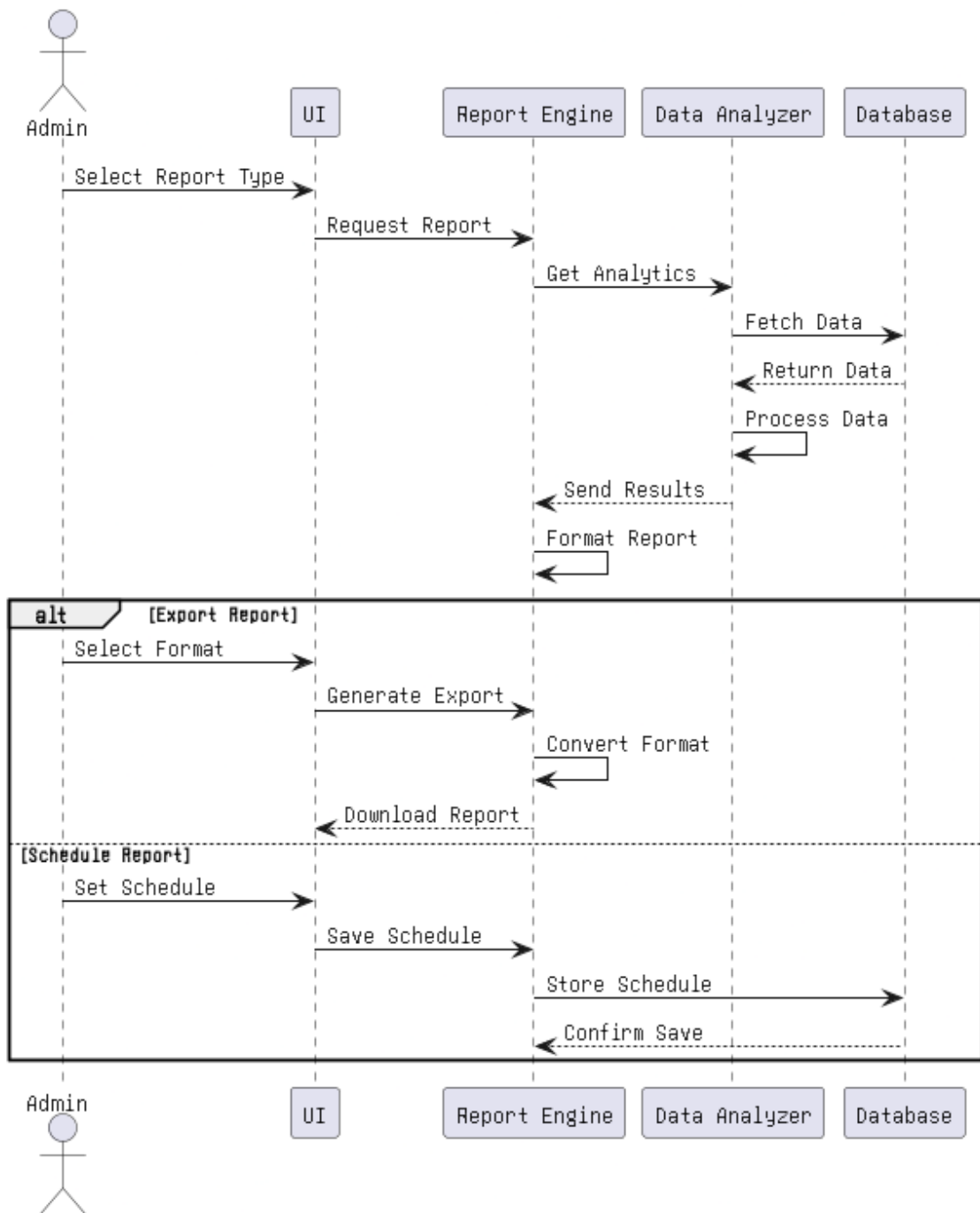


Figure 2.34: Generate Reports

2.4.5 Class Diagram

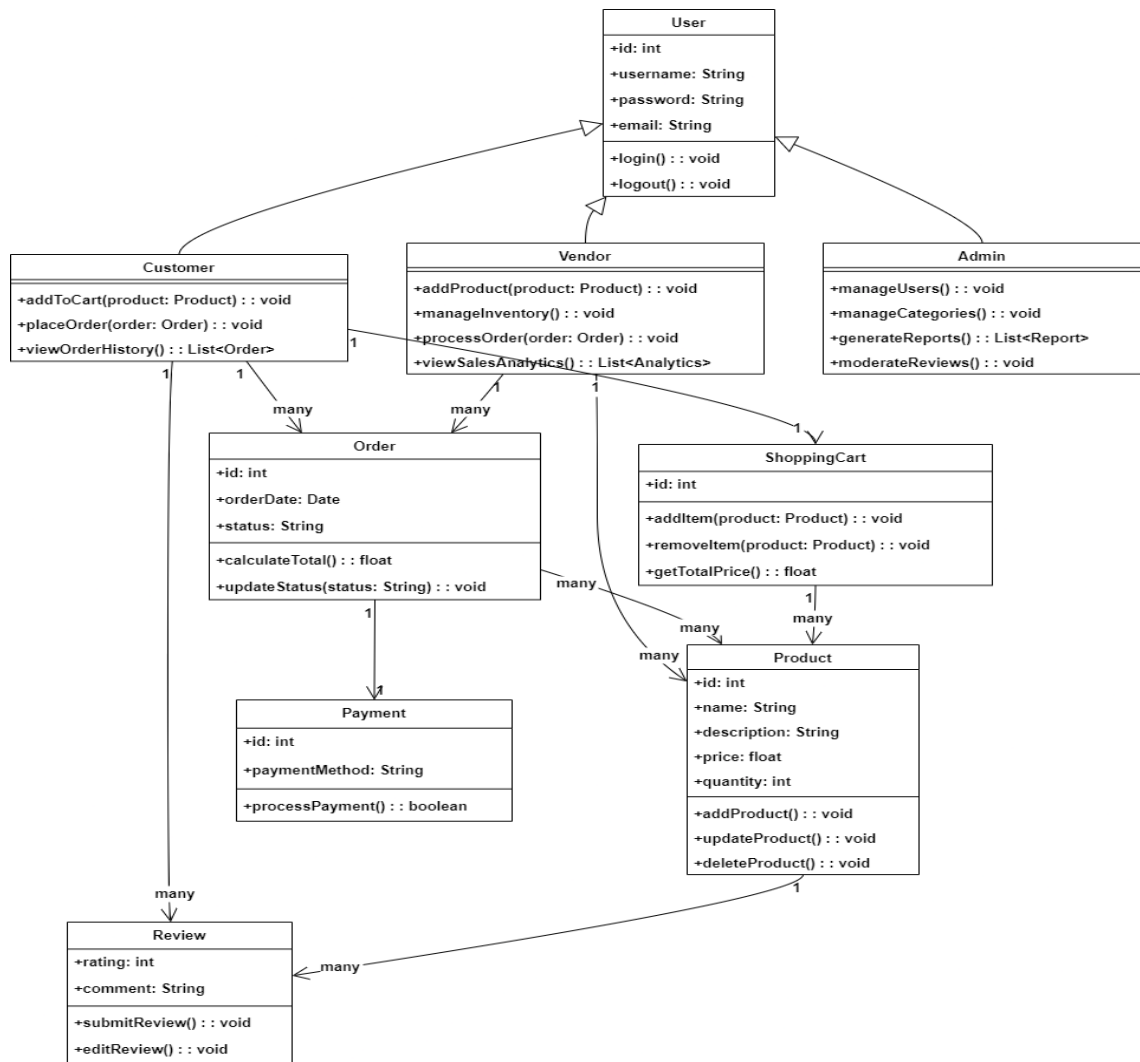


Figure 2.35: Class Diagram

2.4.6 ER Diagram

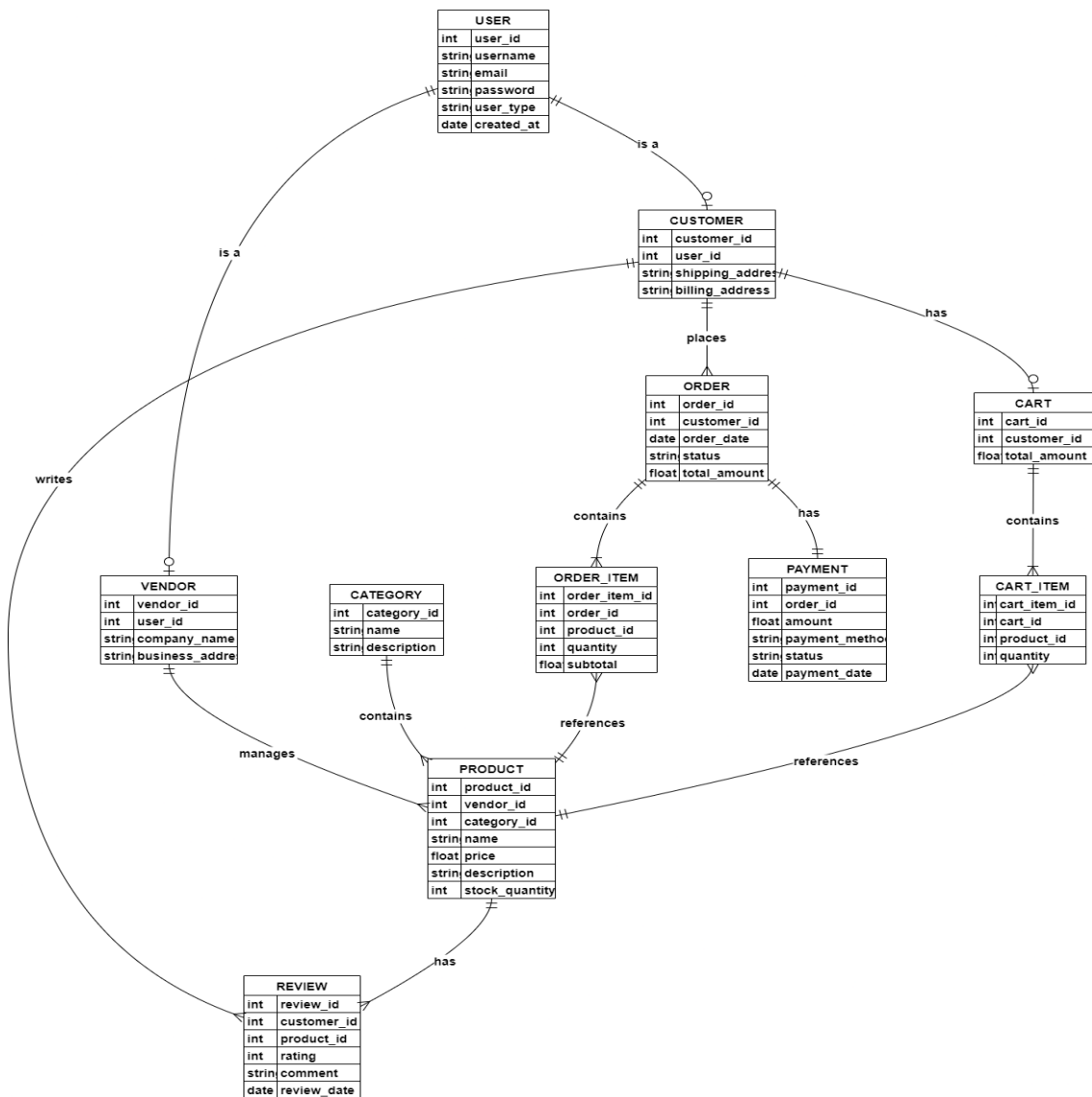


Figure 2.36: ER Diagram

2.6 Summary

This chapter included a comprehensive analysis of the design and implementation stages for the Multi-Vendor E-Commerce Platform. The process commenced by delineating functional and non-functional requirements, so providing the groundwork for system development. Diagrams for object-oriented system design, such as use case, activity, sequence, and ER diagrams, were employed to represent the platform's workflow and architecture.

Documented coding examples illustrate the real implementation of functionality like user registration and product search, showcasing best practices and technologies. This chapter connects theoretical design with practical development, facilitating a smooth transition to testing and deployment phases.

Chapter 3 Software Testing

3.1 Introduction

Software testing is an essential stage in the software development lifecycle, verifying that the system functions as intended and fulfills user expectations. The testing method seeks to detect and rectify faults, ensuring the reliability, functionality, and performance of the multi-vendor e-commerce platform. This chapter delineates the testing methodologies employed, the aspects evaluated, and the outcomes attained throughout the process. The emphasis is on verifying both functional and non-functional requirements to guarantee the system's resilience and user contentment. The testing strategy encompasses both human and automated testing techniques to provide thorough coverage of all system components.

3.2 Testing Features

3.2.1 Feature to Be Tested

- a.** User Registration
- b.** User Login
- c.** Product Browsing
- d.** Product Search
- e.** Cart Management
- f.** Checkout process
- g.** Admin Login
- h.** Admin Dashboard
- i.** Mange Vendors
- j.** Add products
- k.** User login
- l.** Admin Logout
- m.** Order History
- n.** Notifications

3.3 Testing Strategies

3.3.1 Testing Methodology

The testing methodology for the multi-vendor e-commerce platform employs a blend of manual and automated testing methodologies. Unit testing occurs during the development phase to verify individual components, whereas integration testing guarantees smooth interactions among modules. Functional testing checks the system's compliance with requirements, whereas usability testing reviews the user interface and overall experience. Load and performance testing assess the system's responsiveness under different loads, whereas security testing verifies the safeguarding of important user data. End-to-end testing is conducted to verify the entire workflow, confirming that the platform functions as designed.

3.3.2 Pass/Fail Criteria

The pass/fail criteria for testing are defined as follows:

- **Pass:** A test case is marked as passed if all expected outcomes match the actual results without any deviations.
- **Fail:** A test case is marked as failed if the actual results deviate from the expected outcomes, or if the test scenario cannot be executed due to system errors.
- Severity levels (Critical, Major, Minor) are assigned to failed test cases to prioritize fixes during iterative development.

3.4 System Testing (Test Cases with Report)

Sample Test Cases:

Test Case ID	Feature	Test Scenario	Expected Result	Actual Result	Status
TC-001	User Registration	Verify user registration with valid details	User account created successfully	As expected,	Pass

TC-002	User registration	Verify registration with missing mandatory fields	Error message: "Please fill all required fields"	As expected,	Pass
TC-003	User Login	Validate login with correct credentials	User logged in successfully	As expected,	Pass
TC-004	User Login	Validate login with incorrect credentials	Error message: "Invalid username or password"	As expected,	Pass
TC-005	Product Browsing	Ensure product catalogue loads successfully	Products displayed within 3 seconds	As expected,	Pass
TC-006	Shopping Cart Operations	Add product to the shopping cart	Product added successfully	As expected,	pass
TC-007	Checkout Process	Verify payment with valid card details	Payment processed, order confirmed	As expected,	Pass
TC-008	Vendor Product Management	Vendor updates product price	Product price updated successfully	As expected,	Pass
TC-009	Admin Dashboard	Ensure admin can view user statistics	Dashboard displays accurate user data	As expected,	Pass

3.5 Summary

This chapter delineated the extensive testing approach and implementation employed for the multi-vendor e-commerce platform. It emphasized the methodical strategy employed to guarantee the platform's operation, performance, and security. The successful completion of all test cases indicates the platform's preparedness for deployment and its capacity to provide a seamless user experience. Subsequent testing will concentrate on resolving issues detected after deployment and improving the platform according to user feedback.

Chapter 4 Deployment and Maintenance

4.1 Introduction

The deployment and maintenance phases are essential in the lifecycle of a multi-vendor e-commerce platform. Deployment entails rendering the platform accessible to consumers while assuring its seamless operation in a live environment. Maintenance emphasizes the surveillance, updating, and improvement of the platform after launch to accommodate changing user requirements and resolve technical issues. This chapter delineates the deployment strategy and maintenance protocols in accordance with the Software Release Life Cycle (SRLC) for efficient implementation and sustainability.

4.2 Software Release Life Cycle (SRLC)

The Software Release Life Cycle (SRLC) ensures systematic deployment and maintenance of the e-commerce platform. The phases include:

1. **Pre-Alpha Phase:**

- Internal development and testing by the team.
- Integration of core functionalities, such as user registration, product management, and cart operations.
- Identification and rectification of critical bugs.

2. **Alpha Phase:**

- Testing within a controlled environment with selected internal users.
- Feedback collection on usability, performance, and features.
- Initial load testing to measure system performance.

3. **Beta Phase:**

- Release to a limited group of external users for real-world testing.
- Collection of feedback on edge cases, user experience, and system reliability.
- Implementation of fixes and improvements based on beta feedback.

4. **Release Candidate Phase:**

- Preparation of the near-final version of the platform.
- Final performance testing and security checks.
- Documentation and training for support teams.

5. **Production Release (General Availability):**

- Deployment of the platform to the live environment.
- Monitoring of user interactions and system performance.
- Resolution of any immediate post-launch issues.

6. **Maintenance and Updates:**

- Continuous monitoring for bugs, security vulnerabilities, and performance bottlenecks.
- Regular updates to introduce new features, improve functionality, and enhance security.
- Feedback-driven enhancements to meet user expectations and market trends.

7. **End of Life (EOL):**

- Phasing out outdated features or versions.
- Transition planning to newer versions or platforms when necessary.

By adhering to the SRLC, the deployment process becomes structured, reducing risks and ensuring the platform remains reliable, scalable, and user-friendly throughout its lifecycle.

Chapter 5 User Manual

5.1 Introduction

The user handbook offers detailed instructions for users to effectively engage with the multi-vendor e-commerce platform. It comprises detailed instructions, feature elucidations, and troubleshooting guidance to guarantee an uninterrupted user experience. This chapter caters to both end-users and administrators, fulfilling their distinct requirements for optimal platform utilization.

5.2 Project Functionalities

User/Vendor/Admin Login

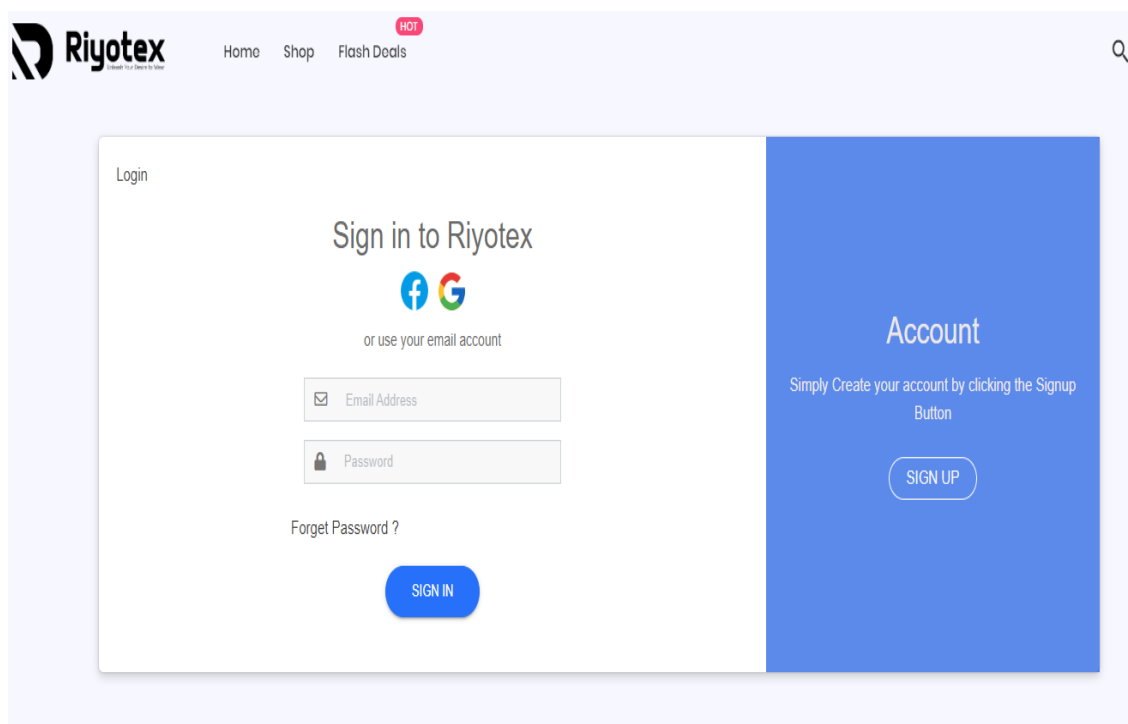
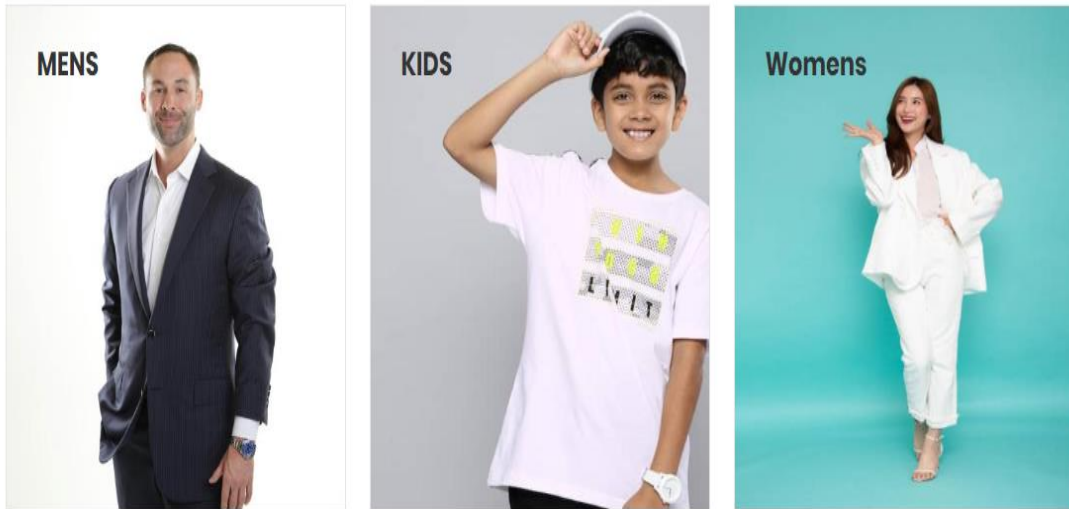


Figure 5.1: Customer / Vendor / Admin Login

Browse Product



PRODUCT OVERVIEW

[All Products](#) [MENS](#) [KIDS](#) [Womens](#) [FACE MASK](#) [SPORTS](#)

Filter

Search



Figure 5.2: Browse Product

Buy Product

The image shows a mobile shopping application interface. On the left, a product page for a 'T-shirt for Mens' is visible, partially obscured by a shopping bag overlay. The product page shows a price of ৳ 847.00, a 3-star rating, and options for size (S) and color (YellowGreen). A quantity selector is set to 1. Buttons for 'ADD TO BAG' and 'BUY NOW' are present. At the bottom, there are links for 'Size Chart', 'Delivery & Return', and 'Ask a Q'. The shopping bag overlay on the right is titled 'SHOPPING BAG' and contains three items:

- T-shirt for Mens**: ৳ 847, quantity 1.
- Premium Casual Shirt**: ৳ 530, quantity 1.
- Premium Soft T-Shirt**: ৳ 1270, quantity 1.

The subtotal for the bag is ৳ 8117 BDT. At the bottom of the bag, there are buttons for 'VIEW CART' and 'CHECK OUT'.

Figure 5.3: Add to cart

Buy Product

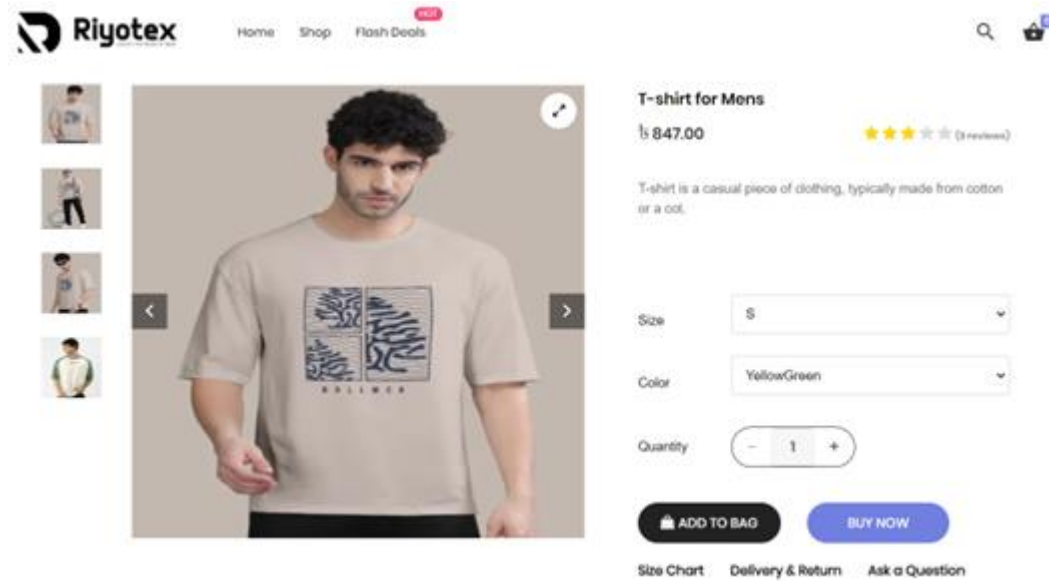


Figure 5.4: Buy Product

Checkout

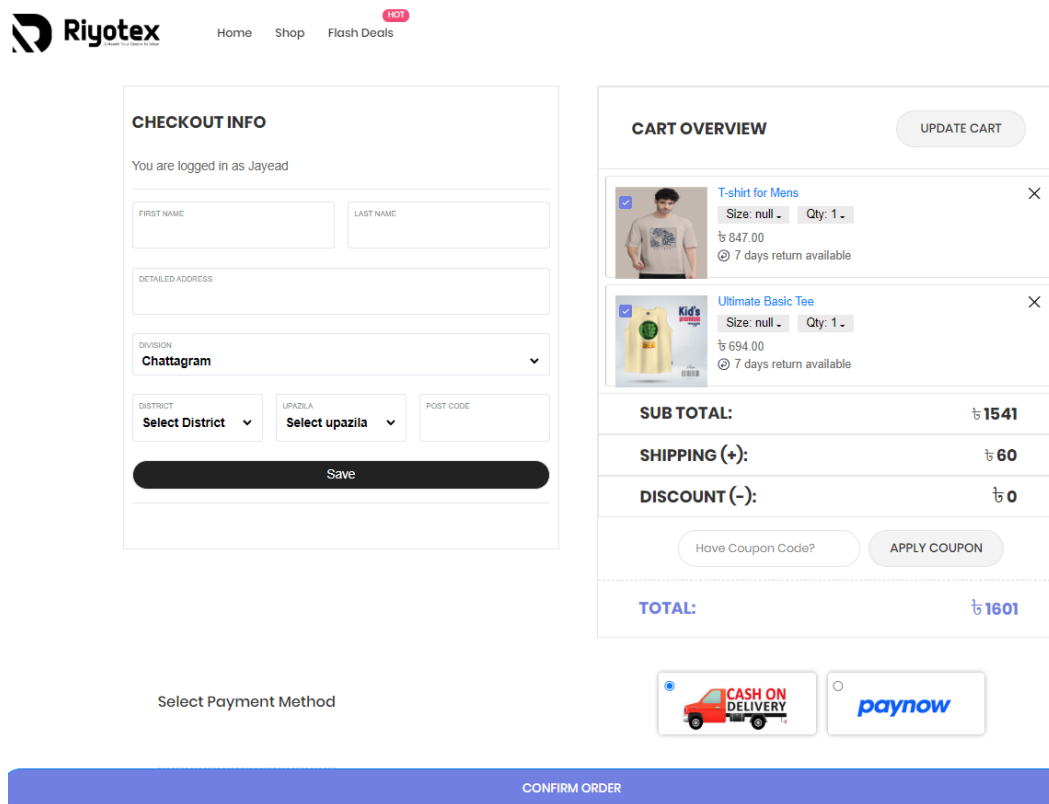
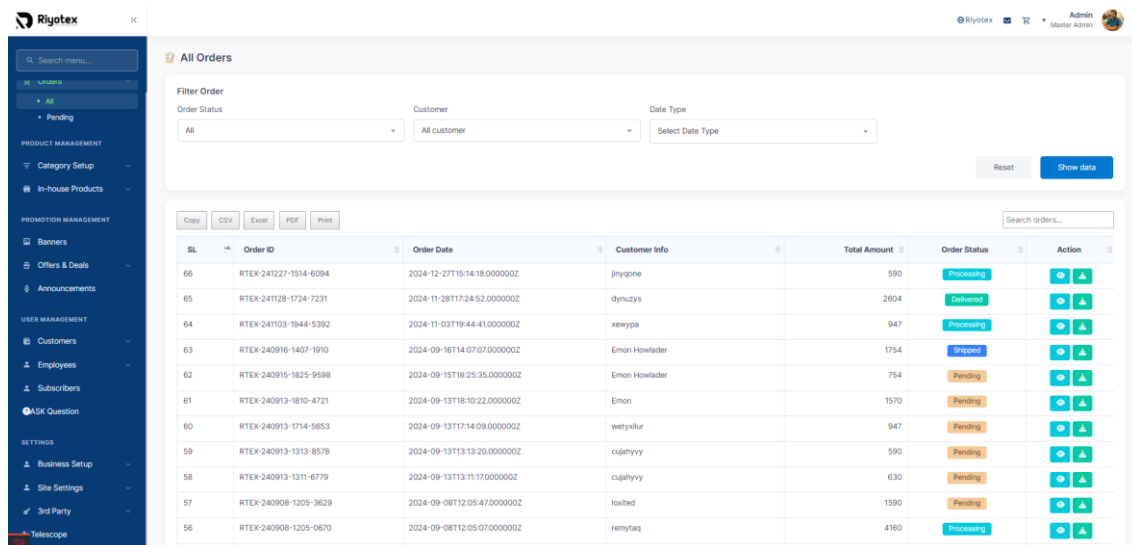


Figure 5.5: Checkout Process

Manage Order

























SL	Order ID	Order Date	Customer Info	Total Amount	Order Status	Action
66	RTEX-241227-1514-6084	2024-12-27T15:14:18.000000Z	jmyqone	590	Processing	 
65	RTEX-241128-1724-7231	2024-11-28T17:24:52.000000Z	dymzys	2604	Delivered	 
64	RTEX-241103-1944-5392	2024-11-03T19:44:41.000000Z	xewyza	947	Processing	 
63	RTEX-240916-1407-1910	2024-09-16T14:07:07.000000Z	Emon Howlader	1754	Shipped	 
62	RTEX-240915-1825-9598	2024-09-15T18:25:35.000000Z	Emon Howlader	754	Pending	 
61	RTEX-240913-1810-4721	2024-09-13T18:10:22.000000Z	Emon	1570	Pending	 
60	RTEX-240913-1714-5853	2024-09-13T17:14:09.000000Z	wetyrkur	947	Pending	 
59	RTEX-240913-1313-8578	2024-09-13T13:13:20.000000Z	cujahyvy	590	Pending	 
58	RTEX-240913-1311-6779	2024-09-13T13:11:17.000000Z	cujahyvy	630	Pending	 
57	RTEX-240908-1205-3629	2024-09-08T12:05:47.000000Z	loited	1590	Pending	 
56	RTEX-240908-1205-9670	2024-09-08T12:05:07.000000Z	remydaq	4160	Processing	 

Figure 5.6: Manage Order

5.3 Summary

The User Manual functions as an extensive resource for users, vendors, and administrators to proficiently navigate and manage the Multi-Vendor E-Commerce Platform. It offers detailed guidance for accessing and controlling diverse functions, guaranteeing a seamless and effective user experience.

The guidebook instructs users on registration, login, product search, cart addition, purchase completion, and order tracking. It delineates the logout procedure and troubleshooting methods for prevalent difficulties.

The guidebook instructs suppliers on account creation, product listing, inventory management, order tracking, and performance analytics review. It encompasses strategies for enhancing product descriptions and graphics to draw more buyers.

The handbook delineates functionalities for administrators, including user management, transaction monitoring, report generation, and platform performance oversight. It additionally addresses the management of escalations, the processing of refunds, and the assurance of conformity with platform policies.

The summary highlights user-centric design, ensuring the manual is accessible and comprehensible for all platform stakeholders, hence enabling them to optimize the platform's potential with minimal technical proficiency.

Chapter 6 Project Summary

6.1 Introduction

This chapter encapsulates the progression of creating the Multi-Vendor E-Commerce Platform. It offers a summary of the project's constraints, extent, and future potential, while contemplating the accomplishments, obstacles, and insights gained. The platform was designed to meet the requirements of sellers and users in the digital marketplace, providing a scalable and user-friendly solution for e-commerce activities.

6.2 Project Limitation

While the project achieved its primary objectives, certain limitations were encountered:

- **Time Constraints:** Due to a strict timeline, some advanced features like AI-based product recommendations and real-time analytics could not be fully developed.
- **Budget Limitations:** Restricted funding limited the integration of premium APIs and advanced cloud-hosting services.
- **Technological Constraints:** The platform currently supports a limited number of payment gateways, with certain country-specific payment methods excluded.
- **Unaddressed Requirements:** Features like multilingual support and customizable themes for vendors are planned but were not included in the current release.
- **Performance Optimization:** While performance is acceptable for moderate traffic, extensive load testing for high traffic scenarios is pending.

These limitations offer areas for future improvement to enhance the platform's functionality and scalability.

6.3 Scope

The project covers the essential features required for a multi-vendor e-commerce platform.

- **Included Features:**
 - Vendor management, including registration, product listing, and inventory tracking.
 - Customer functionalities such as product search, cart management, order placement, and payment processing.
 - Administrative tools, including user management, order tracking, and analytics dashboards.
 - Security features like authentication, authorization, and basic encryption for sensitive data.
- **Excluded Features:**
 - Advanced marketing tools like automated email campaigns or push notifications.
 - AI-based search and recommendation systems.

- Integration with advanced third-party tools for logistics and supply chain management.
- Native mobile applications for Android and iOS (only web-based responsive design provided).

The scope ensures the platform is operational, functional, and meets the needs of vendors and customers while leaving room for future enhancements.

6.4 Future Work

The Multi-Vendor E-Commerce Platform effectively achieves its goals by offering a comprehensive and intuitive solution for both vendors and customers. The project demonstrates effective vendor management, effortless product navigation, and optimized order processing. Crucial insights gained encompass the significance of early stakeholder involvement, iterative testing, and the prioritization of features informed by user feedback.

Notwithstanding its restrictions, the platform exhibits scalability and stability, providing a robust foundation for future enhancements. By implementing the suggested improvements, this project can develop into a holistic e-commerce platform serving a worldwide clientele, fulfilling the requirements of an ever-changing digital marketplace.

6.5 Conclusion

The Multi-Vendor E-Commerce Platform effectively achieves its goals by offering a comprehensive and user-friendly solution for both vendors and customers. The project demonstrates effective vendor management, effortless product navigation, and optimized order processing. Essential lessons acquired encompass the significance of early stakeholder involvement, iterative evaluation, and the prioritization of features grounded in user feedback.

Notwithstanding its restrictions, the platform exhibits scalability and stability, providing a robust foundation for future enhancements. By implementing the suggested improvements, this project can develop into a holistic e-commerce platform serving a worldwide clientele, fulfilling the requirements of an ever-changing digital marketplace..

REFERENCES

1. **Design and Implementation of E-Commerce Platform Based on Android**

Authors: Shen, J., Sun, G.L., & Li, Y.Y.

Journal: *Journal of Computer and Communications*, 2018.

Abstract: This paper discusses the design and implementation of an e-commerce platform on the Android platform, focusing on network communication, image caching, and security features.

<https://www.scirp.org/journal/paperinformation.aspx?paperid=87042>

cite turn0search6

2. **Development of Multi-Vendor E-Commerce Website**

Authors: Patel, K., & Shah, D.

Journal: *International Journal of Scientific Research and Development*, 2021.

Abstract: This paper presents the development of a multi-vendor e-commerce website, discussing the architecture, features, and technologies used to create a platform that allows multiple vendors to sell their products online.

<https://ijsrd.com/articles/IJSRDV9I100035.pdf>

cite turn0search3

3. **Design and Implementation of E-Commerce Management System**

Authors: Zhang, Y., & Wang, L.

Journal: *IOSR Journal of Electronics and Communication Engineering*, 2017.

Abstract: This paper introduces the design and implementation of an e-commerce management system, focusing on system architecture, database design, and key functionalities to support online transactions.

<https://www.iosrjournals.org/iosr-jece/papers/Vol.%2012%20Issue%201/Version-2/D1201022936.pdf>

cite turn0search4

4. **Multi-Tenant and Multi-Vendor Based E-Commerce Platform Using Spring Framework**

Authors: Nandan, R.

Journal: *The International Journal of Science & Technoledge*, 2013.

Abstract: This paper details the high-level architecture and design principles used in developing a multi-tenant, multi-vendor e-commerce platform utilizing the Spring Framework.

<https://www.internationaljournalcorner.com/index.php/theijst/article/view/128184>

□ cite □ turn0search5 □

5. **A Hybrid Model of B2B and B2C Multivendor E-commerce Platform: An Efficient Web Approach**

Authors: Singh, R., & Kaur, P.

Journal: *International Journal of Advanced Engineering and Management*, 2021.

Abstract: This paper proposes a hybrid model combining B2B and B2C functionalities in a multivendor e-commerce platform, discussing the system design and implementation strategies.

https://ijaem.net/issue_dcp/A%20Hybrid%20Model%20of%20B2B%20and%20B2C%20Multivendor%20E%20commerce%20Platform%20An%20Efficient%20Web%20Approach.pdf

□ cite □ turn0search7 □

6. **Designing and Implementing a Cloud-Based E-Commerce Platform**

Authors: Kumar, S., & Gupta, A.

Journal: *International Journal of Creative Research Thoughts*, 2023.

Abstract: This paper explores the design and implementation of a cloud-based e-commerce platform, highlighting the benefits of cloud computing in scalability and performance optimization.

<https://ijcrt.org/papers/IJCRT24A4359.pdf>

□ cite □ turn0search8 □

7. **Multi-Vendor eCommerce Solution with Spring, Angular, and Flutter**

Author: Mahfuj, H.

Platform: GitHub Repository, 2022.

Abstract: This open-source project provides a comprehensive solution for a multi-vendor e-commerce platform, utilizing Spring for the backend, Angular for the web frontend, and Flutter for the mobile application.

<https://github.com/hassanmahfuj/Multi-Vendor-eCommerce-Solution-with-Spring-Angular-Flutter>

□ cite □ turn0search17 □

8. **Managing Multi-Vendor Systems in eCommerce Marketplaces**

Author: Clarity Ventures

Platform: Clarity Ventures Blog, 2023.

Abstract: This article discusses strategies and best practices for managing multi-vendor systems within eCommerce marketplaces, focusing on vendor management, system

integration, and platform scalability.

<https://www.clarity-ventures.com/marketplace-ecommerce/managing-multi-vendor-systems-ecommerce>

An E-commerce Platform With Smart Features

ORIGINALITY REPORT

18%	9%	1%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	6%
2	umpir.ump.edu.my Internet Source	2%
3	Submitted to Midlands State University Student Paper	2%
4	Submitted to NCC Education Student Paper	2%
5	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
6	Submitted to Macquarie University Student Paper	<1%
7	Submitted to Wawasan Open University Student Paper	<1%
8	www.internationaljournalcorner.com Internet Source	<1%
9	Submitted to University of Greenwich Student Paper	<1%

10	file.scirp.org Internet Source	<1 %
11	Submitted to University of Dammam Student Paper	<1 %
12	www.jisc.ac.uk Internet Source	<1 %
13	www.ijaem.net Internet Source	<1 %
14	Submitted to University of Wales Institute, Cardiff Student Paper	<1 %
15	Submitted to Glasgow Caledonian University Student Paper	<1 %
16	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1 %
17	dspace.cuni.cz Internet Source	<1 %
18	Submitted to Blue Mountain Hotel School Student Paper	<1 %
19	Submitted to BB9.1 PROD Student Paper	<1 %
20	Submitted to Campbellsville University Student Paper	<1 %

21	Submitted to University of Westminster Student Paper	<1 %
22	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
23	Submitted to Muscat College Student Paper	<1 %
24	Submitted to University of Strathclyde Student Paper	<1 %
25	Submitted to Majmaah University Student Paper	<1 %
26	Submitted to Notre Dame of Marbel University Student Paper	<1 %
27	edf.bu.edu Internet Source	<1 %
28	www.digitaljournal.com Internet Source	<1 %
29	Submitted to Drexel University Student Paper	<1 %
30	access.atis.org Internet Source	<1 %
31	fastercapital.com Internet Source	<1 %

32	theproductmanager.com Internet Source	<1 %
33	www.carlcorway.cc Internet Source	<1 %
34	Submitted to Trine University Student Paper	<1 %
35	www.avcoe.org Internet Source	<1 %
36	Howard Anderson, Sharon Yull, Bruce Hellingsworth. "Higher National Computing - Core Units for BTEC Higher Nationals in Computing and IT", Newnes, 2004 Publication	<1 %
37	Islam, Mohammad Rubyet. "Dynamic Prognostic Health Management for Response Time Based Remaining Useful Life Prediction of Software Systems", University of Maryland, College Park, 2023 Publication	<1 %

Exclude quotes Off
Exclude bibliography Off

Exclude matches Off