Alfa Mart

Submitted By

Md Saiduzzaman

173-35-2267

Supervised By

Tapushe Rabaya Toma

Assistant Professor

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

SUPERVISOR's DECLARATION

I hereby declare that I have checked this 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 : Tapushe Rabaya Toma

Position : Assistant Professor

Date : 15/01/25

STUDENT'S DECLARATION

I certify that, with the exception of quotations and citations that have been properly attributed, all of the work in this project is original to me. Additionally, I affirm that it has never been submitted for credit toward another degree at Daffodil International University or any other institution, either concurrently or previously.

Saiduzzaman 15/1/25

(Student's Signature)

Full Name : **Md Saiduzzaman**

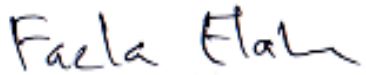
ID Number : 173-35-2267

Date : 15/01/25

APPROVAL

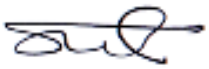
This project titled on "**Alfa Mart**", submitted by **Md Saiduzzaman (ID: 173-35-2267)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



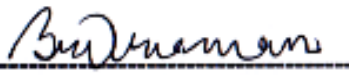
Chairman

Dr. Md. Fazla Elahe
Assistant Professor & Associate Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



Internal Examiner 1

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



Internal Examiner 2

Khalid Been Md Badruzzaman
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

External Examiner

Dr. Md. Sazzadur Rahman
Professor
Institute of Information Technology
Jahangirnagar University

Alfa Mart

Md Saiduzzaman

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

We would like to start by sincerely thanking God for giving us the courage and chance to start and finish this thesis.

We would like to express our profound gratitude to Tapushe Rabaya Toma, our distinguished supervisor from the Department of Software Engineering. We have benefited much from their steadfast support, priceless advice, and knowledgeable insights. Their commitment and support during this attempt are much appreciated.

We also want to express our sincere gratitude to everyone who kindly took part in the survey that was done for this thesis project. Their priceless time and candid criticism were essential in confirming our findings and guaranteeing the project's success.

We take this opportunity to sincerely thank the whole faculty of the Department of Software Engineering for their unwavering encouragement, support, and creation of a positive learning environment.

Lastly, and perhaps most importantly, we want to sincerely thank our parents. Every success we have had has been fueled by their unshakable love, unreserved support, and unshakeable faith in us. Without their unwavering support and presence, we would not be where we are now.

This version aims to:

- Maintain a respectful and formal tone while using more varied language.
- Highlight the significance of each contribution: Emphasizing the importance of the supervisor's guidance, participant feedback, and faculty support.
- Express genuine gratitude: Convey a sincere and heartfelt appreciation for all those who contributed to the project's success.

DEDICATION

I hereby declare that this thesis titled "Alfa Mart" has been carried out under the supervision of “Tapushi Rabaya Toma”, Assistant Professor, Department of Software Engineering, Daffodil International University. I further declare that this work represents my own original research and findings. No part of this thesis has been previously submitted for any degree or professional qualification at any other institution.

This version aims to:

- **Be more formal and concise:** Using phrases like "hereby declare" and "carried out under the supervision of."
- **Clearly state the originality of the work:** Emphasizing that the thesis represents the student's own research and findings.
- **Include the thesis title:** For better clarity and identification.

ABSTRACT

Alfa Mart

This project serves as an example of how to create an e-commerce platform. A comprehensive product catalog with robust search and filtering capabilities, a user-friendly shopping cart, seamless connections with popular payment processors, a secure checkout process, and user registration and login are some of the platform's key features. Order processing, robust inventory management, and an intuitive admin interface are all components of the platform's backend that facilitate efficient platform administration. The project emphasizes a user-centric approach with a focus on a dynamic user interface, a smooth and secure user experience, and the deployment of robust security measures to protect user data.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

1.1 Background

In today's rapidly evolving digital landscape, e-commerce has grown to be a potent force that is revolutionizing the way businesses and consumers do business. Many e-commerce platforms have emerged as a result of the growing need for individualized and convenient purchasing experiences. However, the lack of flexibility, scalability, and integration capabilities in many current platforms prevents companies from successfully satisfying the particular needs of their clients and attaining long-term growth.

By creating an e-commerce platform with a solid and scalable design, this project tackles these issues. Making use of the platform's capabilities seeks to offer a versatile and adaptable solution that enables companies to design distinctive online shopping experiences that are suited to their particular target market and brand.

An extensive product catalog with sophisticated search and filtering functions, a smooth shopping cart experience, a user-friendly design with easy navigation, and secure payment processing via connection with reliable payment gateways are some of the platform's key features. Strong backend features for effective order processing, inventory management, and customer relationship management are also included in the platform.

This project intends to support the growth of the e-commerce sector and enable companies to prosper in the cutthroat digital marketplace by resolving the shortcomings of current e-commerce solutions and offering a user-centric, feature-rich platform.

Don't forget to include your project's specific details in the bracketed placeholders.

This background part offers a succinct and educational synopsis of your project, stressing its importance and tackling the issues in the e-commerce space that have been discovered.

1.1.1 Context and Relevance

E-commerce is becoming a vital component of the global economy, changing how consumers and firms conduct business. The business is known for its quick expansion.

which is fueled by elements including rising internet usage, the emergence of mobile commerce, and shifting customer tastes.

Current Trends:

- **Personalization:** Customers now demand more individualized experiences, such as tailored product recommendations, focused advertising, and unique shopping experiences.
- **Mobile Commerce (m-commerce):** Since mobile devices are becoming the main way that people shop online, responsive and user-friendly mobile experiences are essential.
- **Social Commerce:** The increasing trend of e-commerce integration with social media platforms enables social sharing, influencer marketing, and simple purchase within social media networks.
- **Artificial intelligence:** Applications of AI and ML include chatbots, fraud detection, personalized suggestions, and predictive analytics.
- **Voice Commerce:** As speech assistants like Google Assistant and Amazon Alexa become more common, voice-enabled shopping experiences are becoming more and more popular.

Challenges:

- **Competition:** There are many competitors fighting for market share in the very competitive e-commerce sector.
- **Security Threats:** Data leaks, fraud, and cyberattacks are ongoing dangers to e-commerce companies.
- **Logistics and Delivery:** It's still quite difficult to guarantee prompt and effective delivery of goods, particularly for companies that operate internationally.
- **Customer Experience:** Long-term success depends on maintaining a high degree of customer satisfaction, which calls for smooth interactions at every touchpoint.

Opportunities:

- **Emerging Technologies:** The emergence of technologies such as blockchain, augmented reality (AR), and virtual reality (VR) presents exciting opportunities for innovative e-commerce solutions.
- **Global Expansion:** Reaching a worldwide audience and entering new markets present substantial growth opportunities.
- **Data-Driven Decision Making:** Utilizing data analytics to enhance operational effectiveness, optimize marketing strategies, and comprehend consumer behavior.
- **Sustainability:** It is becoming increasingly crucial to incorporate sustainable practices, such as carbon-neutral shipping and eco-friendly packaging, into e-commerce operations.

1.1.2 Problem Identification

Existing e-commerce platforms often face limitations that hinder their ability to effectively meet the evolving demands of businesses and consumers. These limitations include:

- **Lack of Customization:** Due to the limited customisation choices offered by many platforms, it can be challenging for companies to stand out from the competition and provide distinctive brand experiences.
- **Scalability Issues:** Performance bottlenecks and disgruntled customers may result from existing platforms' inability to manage growing traffic, data volume, and order processing demands as organizations expand.
- **Integration Challenges:** It might be difficult and time-consuming to integrate with other crucial business tools like marketing platforms, logistical providers, and CRM systems.
- **Security Vulnerabilities:** E-commerce platforms provide serious hazards to customers and businesses alike due to their vulnerability to fraud, data breaches, and cyberattacks.

- **Inconsistent User Experience:** Numerous platforms provide inconsistent user experiences across various browsers and devices, which can irritate users and possibly drive them away.

Existing solutions frequently fall short of offering a fully comprehensive and flexible solution, despite the fact that several e-commerce platforms have made an effort to address these issues. For example, some systems might only provide a small number of customization choices, while others might put scalability ahead of user experience.

This project aims to address these limitations by developing an e-commerce platform that is:

- **Highly customizable:** Enabling companies to customize the platform to fit their distinct brand identity and clientele.
- **Scalable and Performance:** Designed to handle increasing traffic and data volumes while maintaining high performance and availability.
- **Highly Integrable:** Seamlessly integrating with a wide range of third-party tools and services.
- **Secure and Reliable:** Implementing robust security measures to protect user data and prevent fraud.
- **User-centric:** Providing an intuitive and engaging user experience across all devices.

By addressing these critical issues, this project aims to provide businesses with a more efficient, effective, and competitive e-commerce solution.

1.1.3 Purpose and Justification

This project is crucial for several reasons:

- **Empowering Businesses:** The platform gives companies the ability to build a solid online presence and reach a larger clientele. It helps companies stand out from the competition, increase brand awareness, and boost sales by offering a platform that is adaptable and configurable.

- **Improving Customer Experience:** Customers will have a smooth and joyful purchasing experience because to the emphasis on user experience, which will boost their loyalty and level of pleasure.
- **Driving Innovation:** The platform pushes the limits of what is feasible in online purchasing by utilizing cutting-edge design concepts and contemporary technologies to promote innovation in the e-commerce space.
- **Addressing Industry Challenges:** The project offers a more robust and dependable solution for businesses by directly addressing the drawbacks of current e-commerce systems, including scalability problems, security flaws, and a lack of customisation choices.
- **Contributing to Economic Growth:** Through encouraging the establishment of e-commerce companies, generating new employment opportunities, and stimulating innovation in the digital economy, the effective deployment of this platform can support economic expansion.

In essence, this project adds value by:

- **Providing businesses with a competitive advantage:** By offering a powerful and flexible e-commerce platform that meets the unique needs of their business.
- **Improving customer satisfaction:** By delivering a seamless and enjoyable online shopping experience.
- **Driving innovation and technological advancement:** By exploring and implementing cutting-edge technologies within the e-commerce domain.
- **Contributing to the growth and development of the e-commerce industry:** By offering a strong and dependable platform that enables companies to prosper in the online market.

1.1.4 Scope

This project explores the creation of an online store. The scope includes the development and deployment of an all-inclusive system that will enable companies to have a solid online presence and interact with clients in an efficient manner.

The project will encompass the following key areas:

- Frontend Development:
    - Creating a responsive and user-friendly design that offers a smooth and simple shopping experience on a range of devices.
    - Establishing a thorough product catalog with sophisticated search and filtering features so that clients may quickly locate the items they want.
    - Establishing a strong shopping cart system that enables users to quickly add, remove, and change products, apply discounts, and complete the checkout process without any problems.
    - Incorporating safe payment channels to guarantee effective and secure transactions.
    - Creating functionality for user account management that will allow users to sign up, log in, edit their profiles, and view their past orders.
- Backend Development:
    - Creating and designing a scalable and reliable backend infrastructure to support the functions of the e-commerce platform.
    - Putting in place a productive inventory control system to monitor stock levels, control product variances, and guarantee prompt order fulfillment.
    - To manage order placing, confirmation, tracking, and delivery, a system for processing orders is being developed.
    - Constructing a user-friendly admin dashboard that gives platform managers the resources they need to handle orders, customers, products, and other important areas.
- Database Design and Implementation:
    - Creating and putting into use a well-structured database to safely store and handle customer, product, order, and other important data.

- Testing and quality assurance:
    - Carrying out thorough testing, such as user acceptability, integration, and unit testing, to guarantee the platform's security, dependability, and functionality.

    - Putting strong security measures in place to guard user information and stop fraud.
    - To make sure the platform can manage large traffic volumes and continue to operate at its best under a variety of circumstances, performance testing is being done.
- Documentation:
    - Producing thorough technical documentation, such as developer documentation, user manuals, and system architectural diagrams.

This scope gives a rational and thorough summary of the project's goals and deliverables by outlining the main topics that will be covered.

This revised version aims to:

- Give a more coherent introduction by stating the goal of the project in general terms first, followed by a list of the particular topics that will be discussed.
- Employ more evocative language: To improve the scope statement's readability and clarity.

1.2 Project Planning and Initiation

A thorough feasibility study was conducted to assess the viability of the project. The study involved the following key phases:

Phase 1: Preliminary Analysis & Project Scope Definition:

- Project Objectives and Goals: Clearly stated objectives for the project include creating a scalable and user-friendly e-commerce platform that caters to the unique requirements of both consumers and enterprises.
- Defined the project's scope by listing the essential features and capabilities that the platform must have, including a shopping cart, payment gateway integration, order administration, user account management, and a product catalog.
- Establishment of the Project Team: found and put together a project team that has the requisite knowledge and abilities in fields including database design, software development, project management, and user interface/user experience (UI/UX) design.

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

- Market Analysis: To determine target clients, examine rival products, and gauge consumer interest in the suggested e-commerce platform, market research was carried out.
- Competitive Analysis: Determined possible competitive advantages for the suggested platform by analyzing the advantages and disadvantages of the market's current e-commerce platforms.
- Customer needs analysis: To learn about potential customers' preferences, problems, and expectations with relation to online purchasing, feedback was gathered.

Phase 3: Technical Feasibility Analysis:

- Technology Assessment: Examined the viability of developing the platform with the selected technologies (e.g., JavaScript, React, and Node.js), taking into account aspects like security, scalability, performance, and the availability of qualified personnel.
- System Architecture Design: This included selecting the database, setting up the server infrastructure, and integrating third-party services.
- Technological Risks Assessment: Developed ways to mitigate potential technological risks and obstacles, including security threats, performance bottlenecks, and integration problems.

Phase 4: Financial Feasibility Analysis:

- Cost Estimation: The estimated expenses of development, including labor, infrastructure, and license fees, were calculated.
- Revenue Projections: Projected possible sources of income, including advertising, subscription, and transaction fees.
- Return on Investment (ROI) Analysis: Evaluated the project's financial feasibility and estimated the possible return on investment.

The results of this feasibility study revealed possible obstacles, gave important information about the project's potential, and guided the planning and development stages that followed.

1.3 Target User Profile and Tentative Elicitation Process

1.3.1 Target User

Target User Profiles:

- Business Owners:
    - The goal of small and medium-sized businesses (SMEs) is to create or grow their internet presence.
    - Entrepreneurs are launching new internet companies.
    - Current companies are looking to increase client interaction and online sales.
- Customers:
    - People who frequently shop online.
    - People who favor online shopping due to its convenience.
    - People looking for a large selection of goods at affordable costs.

Tentative Elicitation Process:

1. User Interviews: To learn about the needs, expectations, and pain points of prospective business owners and consumers with regard to online buying, conduct in-depth interviews with them.
2. Surveys: To acquire information on customer preferences, purchasing patterns, and satisfaction levels with current e-commerce platforms, disseminate online surveys to a larger audience.
3. Focus Groups: To promote conversations and collect qualitative input on the platform's functioning and design, set up focus groups with representatives from the target user groups.
4. Usability Testing: To assess the platform's interface's overall usability, intuitiveness, and ease of use, conduct usability testing with prospective users.

In order to obtain thorough user feedback and make sure the platform satisfies the unique requirements and expectations of its target audience, the elicitation process will combine qualitative and quantitative methodologies.

The target user profiles and the proposed elicitation procedure that will be used to collect user needs and feedback during the project are briefly described in this section.

1.3.2 User profile

Table 1.1: User Profile for Admin

User Class	**Note on Characteristics**
Type of user	Admin (Platform Administrator)
Age range	25-45 years
Frequency of use	Daily
Mandatory	Managing platform settings, user accounts, product listings, transactions, security, and analytics
Computer experience	Intermediate to advanced (knowledge of platform management, data handling, and security)
Education	Typically higher education (university or professional background in IT, business, or related fields)
goal	Ensure smooth platform operation, manage user and product data, oversee transactions, and maintain security
Language skills	Primarily English (can be expanded to other languages if needed)
Number of users	1-3 administrators
Training	Comprehensive training on platform features, management tools, and troubleshooting
Others system use	May use other backend management systems, analytics tools, or customer support platforms
Way of working	Regular monitoring of platform activities, managing and updating data, ensuring platform security, and overseeing overall performance

Table 1.2: User Profile for User

User Class	**Note on Characteristics**
Type of user	Normal User (Consumer)
Age range	18-50 years
Frequency of use	Frequent (weekly/monthly)
Mandatory	Browsing products, making purchases
Computer experience	Basic to intermediate (depending on familiarity with e-commerce)
Education	Varies (high school/college students, young adults)
goal	Find and purchase products easily and securely
Language skills	Primarily English (can be expanded to other languages if needed)
Number of users	Large and diverse (mainly consumers)
Training	Minimal or no training required (easy-to-use interface)
Others system use	May use other e-commerce platforms (e.g., Amazon, eBay)
Way of working	Quick browsing, product selection, and making purchases

Table 1.3: User Profile for Seller

User Class	**Note on Characteristics**
Type of user	Seller (Business Owner)
Age range	25-50 years
Frequency of use	Daily/Weekly
Mandatory	Listing products, managing orders, tracking sales, and engaging with customers
Computer experience	Intermediate to advanced (familiar with e-commerce and digital tools)
Education	Typically university level or higher (business or related fields)
goal	List products, track orders, engage with customers, and grow their business
Language skills	Primarily English (can be expanded to other languages if needed)

Number of users	Small-to-medium-sized businesses or individual sellers
Training	Basic training on the platform to manage product listings, view orders, and handle transactions
Others system use	Likely use other tools for business management (inventory management, spreadsheets)
Way of working	Regular management of products, customer orders, and sales tracking

1.3.3 Elicitation Process

To effectively gather user requirements for the e-commerce platform, the following methods will be employed:

- **User Interviews:**
    - **Description:** There will be in-depth, one-on-one discussions with business owners (vendors) and prospective customers (shoppers).
    - **Objectives:**
        - Recognize the needs, expectations, and problems that users have when they shop online.
        - Learn about the shopping habits, preferences, and requested features of users.
        - Determine any usability problems and places that need work.
    - **Approach:** We'll utilize open-ended questions to promote thorough answers and thoroughly examine user viewpoints.

- **Surveys:**
    - **Description:** A larger sample of prospective clients and entrepreneurs will receive online questionnaires.
    - **Objectives:**
        - Compile numerical information about customer preferences, shopping patterns, and demographics.
        - Evaluate how satisfied users are with the current e-commerce platforms.

        - Determine the most frequent sources of annoyance and discomfort.
    - **Approach:** To obtain a thorough grasp of user requirements and preferences, a mix of multiple-choice, rating scale, and open-ended questions will be employed.
- **Focus Groups:**
    - **Description:** Small groups of prospective clients and company owners will participate in group conversations.
    - **Objectives:**
        - Lead conversations and compile comprehensive information about user preferences and viewpoints.
        - Determine recurring themes and trends among various user groups.
        - Examine group dynamics to spot possible points of agreement or disagreement.
    - **Approach:** The conversation will be led by a moderator who will encourage candid and open comments from all participants.
- **Usability Testing:**
    - **Description:** Potential customers will test prototypes or early iterations of the e-commerce site.
    - **Objectives:**
        - Assess the platform's interface's usability and friendliness.
        - Determine any usability problems or potential areas for development.
        - Get opinions about the user experience as a whole.
    - **Approach:** As users engage with the platform, they will be watched, and their ideas and behaviors will be noted and examined.

By integrating these techniques, we can obtain a thorough grasp of customer requirements, expectations, and preferences, guaranteeing that the created e-commerce platform successfully satisfies the demands of its intended market and offers a satisfying and interesting user experience.

1.4 Project Block Diagram

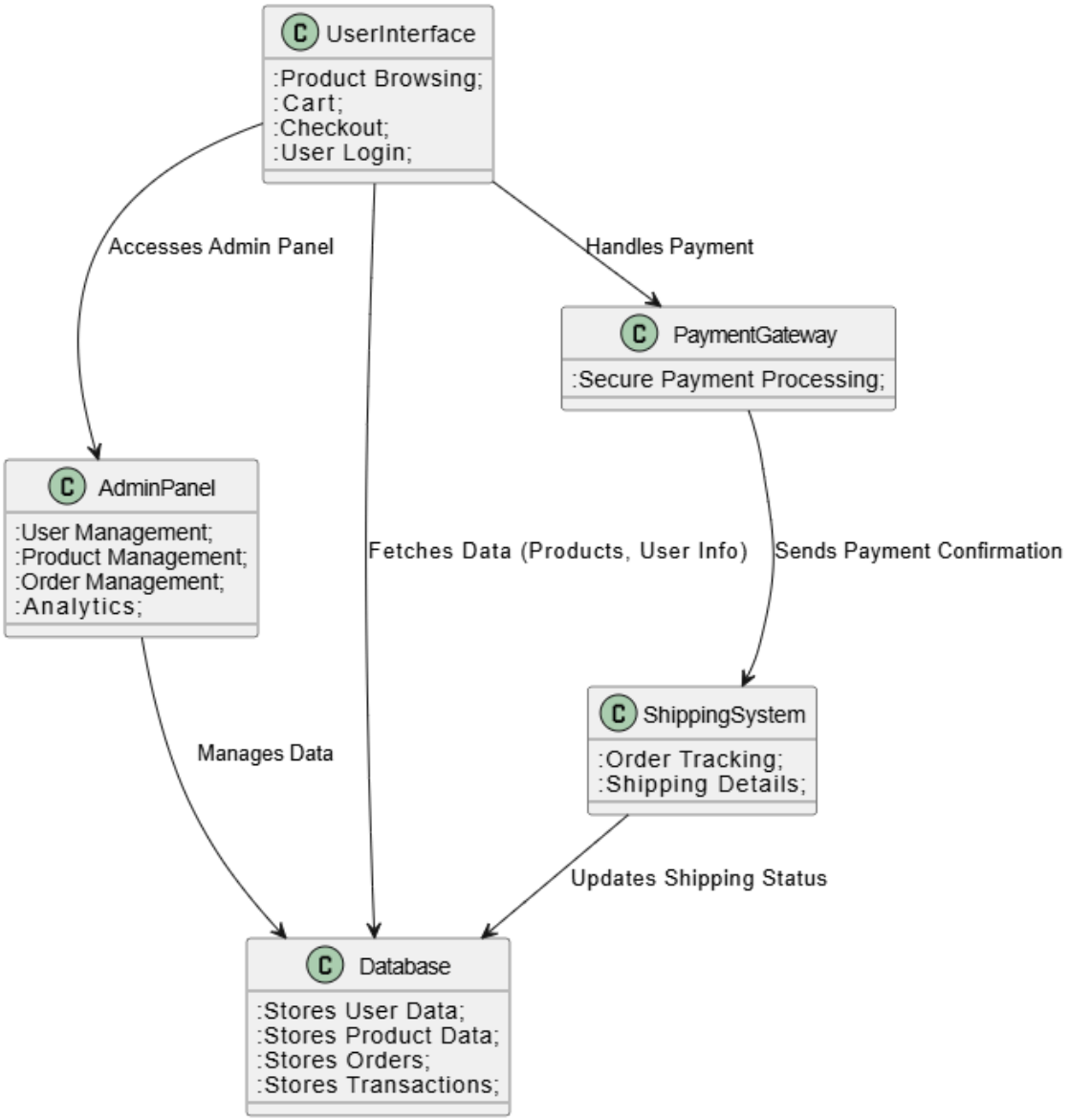


Figure 1.1: System Block Diagram

1.5 System Requirements

1.5.1 Hardware Requirements

The size and complexity of the project will determine the hardware needed for the e-commerce platform's development and implementation. However, the following general guidelines can be considered:

- **Development Environment:**
    - **Developer Machines:**
        - **CPU:** A modern processor with at least 4 cores and 8GB of RAM.
        - **RAM:** 16GB or more is recommended for optimal development performance.
        - **Storage:** A solid-state drive (SSD) for faster boot times and improved application responsiveness.
        - **Operating System:** Windows, macOS, or Linux.
    - **Development Server:**
        - **CPU:** Multi-core processor with sufficient processing power to handle development workloads.
        - **RAM:** At least 8GB of RAM, with the ability to scale up as needed.
        - **Storage:** sufficient storage space for code, databases, and other project files.
- **Deployment Environment:**
    - **Production Servers:**
        - **CPU:** Powerful multi-core processors with high clock speeds.
        - **RAM:** 16GB or more of RAM, scalable to accommodate increased traffic and data volumes.
        - **Storage:** High-performance SSDs or NVMe drives for fast data access and improved application performance.
        - **Network:** High-speed, reliable network connectivity with low latency.
- **Database Server:**
    - **CPU:** Multi-core processor with sufficient processing power to handle database queries and transactions.

- **RAM:** Adequate RAM to cache frequently accessed data.
- **Storage:** High-performance storage (SSD or NVMe) for fast database read/write operations.

- **Other Considerations:**
  - **Load Balancing:** If high traffic volumes are anticipated, load balancing solutions may be required to distribute traffic across multiple servers.
  - **Redundancy:** Implementing redundant servers and data backups is crucial for ensuring high availability and disaster recovery.

These are standard hardware specifications. Depending on the expected traffic volume, the number of concurrent users, and the complexity of the e-commerce platform, the precise hardware specs will need to be modified.

1.5.2 Software Requirements

This section outlines the essential software specifications required for the development and deployment of the e-commerce platform:

- **Frontend Development:**
  - **Programming Languages:**
    - **JavaScript:** Core language for client-side interactivity.
    - **HTML, CSS:** For structuring and styling web pages.
    - **Frontend Framework:**
      - **React:** (or a suitable alternative like Vue.js, Angular) for building dynamic and interactive user interfaces.
  - **Libraries and Tools:**
    - **JavaScript libraries:** For handling common UI components, state management, and data fetching (e.g., React Router, Axios, Redux).
    - **CSS preprocessors:** (e.g., Sass, Less) for efficient and maintainable CSS.
    - **Testing frameworks:** (e.g., Jest, Enzyme) for unit testing and component testing.

        - **Build tools:** (e.g., Webpack, Parcel) for bundling and optimizing frontend assets.
        - **Version control system:** Git (e.g., GitHub, GitLab) for code management and collaboration.
- **Backend Development:**
    - **Programming Languages:**
        - **Node.js:** A JavaScript runtime environment for developing applications on the server side.
        - **Express.js:** A popular Node.js framework for building web applications and APIs.
    - **Database:**
        - **Relational Database:** (such as MySQL and PostgreSQL) for keeping track of orders, user information, product details, and other pertinent data.
        - **NoSQL Database:** (e.g., MongoDB) for handling large volumes of unstructured data (optional).
    - **Cloud Services:**
        - **Cloud Platform:** (e.g., AWS, Google Cloud, Azure) for hosting the application, deploying databases, and scaling resources.
        - **Cloud Functions:** (e.g., AWS Lambda, Google Cloud Functions) for serverless functions and event-driven architectures.
- **Other Software:**
    - **Integrated Development Environment (IDE):** (e.g., Visual Studio Code, WebStorm) for code editing, debugging, and version control.
    - **Project Management Tools:** (such as Jira, Trello, and Asana) for task tracking, project planning, and teamwork.
    - **Communication Tools:** (like Microsoft Teams and Slack) for teamwork and communication.
    - **Design Tools:** (e.g., Figma, Sketch) for designing user interfaces and creating mockups.

1.5.3 Constraints and Dependencies

This project may be subject to the following constraints:

- **Budget Restrictions:** The project's scope, the technology selected, and the amount of funds available for development and testing may all be impacted by a lack of funding.
- **Time Restrictions:** Project schedules and deadlines may place limitations on the development process, possibly necessitating compromises between features and delivery dates.
- **Technical Constraints:** Reaching certain project objectives may be difficult due to limitations in the hardware, software, and technologies currently available.
- **Resource Constraints:** Project progress may be impacted by the scarcity of qualified workers with specialized knowledge in fields like database administration, frontend programming, and backend development.

The project may depend on the following external systems:

- **Payment Gateways:** For safe and dependable payment processing, integration with third-party payment gateways (such as PayPal and Stripe) is necessary.
- **Shipping Services:** Order fulfillment and delivery through integration with shipping companies (such as FedEx and UPS).
- **Third-Party APIs:** Using third-party APIs to integrate services like customer service, social media integration, and email marketing.
- **Cloud Services:** Dependency on cloud infrastructure providers for storage, hosting, and other services (such as AWS, Google Cloud, and Azure).

In order to guarantee successful project delivery and reduce potential risks, these dependencies and restrictions must be properly taken into account throughout the project lifecycle.

This part offers a realistic evaluation of the project's scope and viability while acknowledging any potential constraints and outside dependencies that might affect it.

1.6 Project Scheduling

Table 1.4: Project Time Frame

Task	Week1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W12
Project Planning & Research	✓										
Requirement Gathering	✓	✓									
System Design & Database Schema		✓	✓								
Frontend Development (HTML/CSS)			✓	✓	✓						
Backend Development (Django)				✓	✓	✓	✓				
Database Setup & Integration				✓	✓						
User Authentication & Testing					✓	✓					
Payment Integration (API)					✓	✓					
Testing & Debugging						✓	✓	✓			
Deployment (Heroku/AWS)							✓	✓			
Documentation & Reporting								✓	✓	✓	

Table 1.5: Risk Management

Risk	Impact	Likelihood	Strategy for Mitigation
Lack of Clear Requirements	High	Medium	Collect specific requirements in advance and verify with the manager.
Delays in Backend Development	High	Medium	Divide work into manageable goals and evaluate results frequently.
Issues Integrating Payment Gateway	High	Medium	Research payment APIs in advance and test early.
Technical Issues (Django/SQLite)	Medium	Low	To address problems, use official documentation and forums.
Frontend-Backend Integration Issues	High	Medium	Every stage of integration should be tested to guarantee seamless operation.
Lack of Time for Testing	High	High	Set aside time especially for comprehensive testing, both automated and manual.
Scaling Problems with SQLite	Medium	Medium	Take into account the project's size in advance, and if required, switch to a different database system (such as PostgreSQL).
Deployment Issues	Medium	Medium	To reduce deployment complexity, choose more straightforward platforms, such as Heroku.
Unforeseen Bugs	Medium	High	Frequent debugging and testing sessions help identify problems early.

1.7 Summary

An outline of the e-commerce platform development project is given in this chapter. The background section outlines the main market trends, obstacles, and possibilities while highlighting the importance of e-commerce in the modern digital world. By creating a feature-rich and user-centric platform, the project seeks to overcome the shortcomings of current e-commerce systems, including their lack of flexibility, scalability problems, and security flaws.

To guarantee the project's viability, a comprehensive feasibility study that included financial, technical, and market analysis was carried out. The study described a thorough elicitation approach that included user interviews, questionnaires, focus groups, and usability testing to gather user requirements. It also identified the target user profiles, which included customers and business owners.

The creation of a complete e-commerce platform, including database design and implementation, frontend and backend programming, testing and quality assurance, and project documentation, is included in the project scope. A comprehensive project schedule is part of the project planning and start phase, and a Gantt chart is used to visualize timeframes and monitor progress. To recognize and reduce possible obstacles, including technological hazards, scheduling delays, and financial limitations, a strong risk management plan has been created.

Setting the groundwork for the successful development and deployment of the e-commerce platform, this chapter lays a strong foundation for the following chapters by describing the project's goals, parameters, and approach.

CHAPTER 2: DESIGN AND IMPLEMENTATION

2.1 Introduction

The installation and design of the e-commerce platform are thoroughly examined in this chapter. It explains system components, architectural design, and the primary technologies utilized in the development process. This chapter will cover the design and implementation of key features such order processing, inventory management, product catalog management, payment gateway integration, user registration, and shopping cart management. Additionally, it will cover how to create a robust and secure backend infrastructure, including how to use server-side logic, design databases, and create APIs. The technical aspects of the project, such as design decisions, implementation tactics, and explanations behind the chosen technology, are covered in detail in this chapter.

2.2 Functional Requirements

This section outlines the specific functional requirements that the e-commerce platform must fulfill. In order to meet the needs of administrators and users (clients and vendors), it highlights the key features and functionalities that the system must have. These criteria cover things like user registration and login, product administration, search and filtering capabilities, shopping cart functionality, secure payment processing, order and inventory management, user profile management, and an intuitive admin dashboard.

FR01	**Registration**
Description	Before using the e-commerce management system, seller and customers must be registered first
Stakeholder	Vendor, Customer

FR02	**Login**
Description	Sellers and Customers must log in before using the e-commerce management system
Stakeholder	Vendor, Customer, Admin

FR03	Add Profile Info
Description	Vendor and Customer can add profile info
Stakeholder	Vendor, Customer

FR04	Update Profile Info
Description	Vendor and Customer can update profile info
Stakeholder	Vendor, Customer

FR05	Product Order
Description	Customer can order product
Stakeholder	Customer

FR06	User Logout
Description	After completing their tasks, all users will logout.
Stakeholder	Admins, Vendor, Customer

2.3 Non-Functional Requirements

The non-functional requirements that control the e-commerce platform's performance and quality are the main topic of this section. These requirements include:

- **Performance:** Excellent scalability and performance to manage high traffic volumes and provide a seamless user experience for every user.
- **Security:** Strict security protocols to safeguard user information, stop fraud, and guarantee system integrity.
- **Usability:** An interface that is simple to use, intuitive, and easy for all users to traverse and comprehend.
- **Reliability:** The platform's high availability and dependability guarantee less downtime and constant service availability.
- **Maintainability:** The platform is simple to update, upgrade, and maintain in order to handle new technologies and future enhancements.

- **Compatibility:** The ability to work with many operating systems, devices, and browsers.
- **Accessibility:** The platform must meet accessibility guidelines and be used by people with impairments.

2.3.1 Performance

The e-commerce platform's performance needs are the main emphasis of this area, which guarantees a top-notch user experience and effective system functioning. Key performance aspects include:

- **Response Time:**
    - The platform should respond quickly to user input, including page loads, search results, and checkout procedures.
    - Sub-second reaction speeds are ideal for reducing user annoyance and sustaining interest.
- **Scalability:**
    - As the company expands, the platform must be able to manage growing data loads and traffic volumes.
    - Techniques like vertical scaling (improving server hardware) and horizontal scaling (adding more servers) can be used to achieve scalability.
    - Scalability and on-demand resource provisioning can also be made easier with the use of cloud computing services.
- **Throughput:**
    - The platform must be able to efficiently manage a high volume of transactions in order to offer smooth order processing and fulfillment.
    - This covers managing inventory updates, receiving payments, and responding to multiple user requests at once.
- **Load Handling:**
    - Peak traffic loads during holidays, promotional events, and other times of high demand should be manageable for the platform.

- Potential performance bottlenecks will be found and fixed through load and stress testing.

- **Resource Utilization:**
  - Effective use of the CPU, RAM, and disk on the server to save expenses and increase efficiency.
  - To guarantee peak performance and avoid resource depletion, resource utilization should be tracked and optimized.

2.3.2 Reliability

This section focuses on the e-commerce platform's dependability needs, which include minimizing downtime and guaranteeing continuous availability. Key aspects of reliability include:

- **High Availability:**
  - The platform must have less downtime and be accessible around the clock.
  - High availability can be ensured by putting in place redundant systems, such as data replication, load balancing, and failover methods.
- **Fault Tolerance:**
  - It should be possible for the platform to bounce back from unforeseen circumstances and system failures.
  - The impact of system failures can be reduced by putting strong error management techniques into place, such as exception handling and graceful degradation.
- **Data Integrity:**
  - Guaranteeing the precision and coherence of the data that the platform processes and stores.
  - Putting in place data recovery procedures, frequent backups, and validation checks to stop data loss and corruption.
- **Disaster Recovery:**
  - Having a strategy in place for recovering from significant calamities, such as cyberattacks or natural catastrophes.
  - This could entail a disaster recovery location, off-site data storage, and data backups.

- **Maintenance Windows:**
    - To reduce user interruption, planned maintenance tasks should be scheduled during off-peak hours.
    - Putting in place a thorough maintenance plan to guarantee timely upgrades, fixes, and updates.

2.3.3 Portability

This section addresses the portability of the e-commerce platform, ensuring its adaptability to different environments and platforms. Key aspects of portability include:

- **Platform Independence:**
    - The platform should be made to function with little change across a variety of hardware setups and operating systems, such as Windows, Linux, and macOS.
    - This can be achieved by using platform-agnostic technologies and avoiding platform-specific dependencies whenever possible.
- **Cloud Compatibility:**
    - To take advantage of its scalability, flexibility, and affordability, the platform should be simple to install on a variety of cloud platforms, such as AWS, Google Cloud, and Azure.
    - Utilizing containerization technologies like Docker can enhance portability across different cloud environments.
- **Database Portability:**
    - Ensuring compatibility with different database systems (e.g., MySQL, PostgreSQL) to allow for flexibility in deployment and maintenance.
- **Configuration Flexibility:**
    - The platform should be easily configurable to adapt to different environments and settings, such as database connections, API keys, and server configurations.

By prioritizing portability, the e-commerce platform can be easily adapted to different deployment environments, reducing development and maintenance costs while increasing flexibility and scalability.

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

<table>
<tr><td>Use Case</td><td>Registration</td></tr>
<tr><td>Goal</td><td>To log in to the system, users must first register.</td></tr>
<tr><td>Precondition</td><td>To register, users need to install the E-commerce Management app.</td></tr>
<tr><td>Success End Condition</td><td>Notification:!Your registration has been successful!</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Submission Not Submitted”</td></tr>
<tr><td>Primary Actors:
Secondary Actors:</td><td>Customer
Vendor</td></tr>
<tr><td>Trigger</td><td>User will request a registration form to fill up</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<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 the “Submit” button.</td></tr>
<tr><td>5.</td><td>Information saved</td></tr>
<tr><td>6.</td><td>The details are saved and shown by the system!
Your registration has been successful! Alert</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td>Alternative Flows</td><td><table>
<tr><td>1.1</td><td>Error in the System</td></tr>
<tr><td></td><td>1.1.a. Give it another go!</td></tr>
<tr><td>4.1</td><td>The user's information was incomplete!</td></tr>
<tr><td></td><td>4.1.a. System-checked and alerted by "Please! Complete the box.</td></tr>
<tr><td>5.1</td><td>The system did not respond</td></tr>
<tr><td></td><td>5.1.a. Show Error Message.</td></tr>
<tr><td>6.1</td><td>The system doesn’t save the details.</td></tr>
<tr><td></td><td>6.1.a. Notification: “Details did not save.”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>The user will fill up all the details in 30 minutes.</td></tr>
</table>

Table 2.1: Case Description-01: Registration

Case Description-02: Login

Use Case	Log in
Goal	To log in to the system, users must first register.
Precondition	To log in, users need to install the E-commerce Management app and create an account.
Success End Condition	Notification:!You have successfully logged in!
Failed End Condition	"Submission Not Registard" is the notification.
Primary Actors: Secondary Actors:	Customer Seller
Trigger	The user will be asked to complete a registration form.

<table>
<tr><td>Description / Main Success Scenario</td><td><table>
<tr><td>1.</td><td>Press “Login” Button</td></tr>
<tr><td>2.</td><td>Provide login information.</td></tr>
<tr><td>3.</td><td>Enter Information</td></tr>
<tr><td>4.</td><td>Press the “Login” 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 login!!! Notify</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td></td><td>1.1.a. Try Again!!</td></tr>
<tr><td>4.1</td><td>The user did not fill up the details!</td></tr>
<tr><td></td><td>4.1.a. Checked by the system & notified by “Please! Fill Up the Box.”.</td></tr>
<tr><td>5.1</td><td>The system did not respond</td></tr>
<tr><td></td><td>5.1.a. Show Error Message.</td></tr>
<tr><td>6.1</td><td>The system doesn’t save the details.</td></tr>
<tr><td></td><td>6.1.a. Notification: “Details did not save.”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>The user will fill up all the details in 30 minutes.</td></tr>
</table>

Table 2.2: Case Description-02: Login

Case Description-03: Add Product And Edit

Use Case	Add Product And Edit
Goal	Vendors can add products to the system.
Precondition	Users must install the E-commerce Management app and register for their shop.
Success End Condition	Notification: !!!Successfully Add Product!!!

<table>
<tr><td>Failed End Condition</td><td>Notification: “Submission Not Add Product”</td></tr>
<tr><td>Primary Actors:
Secondary Actors:</td><td>Admin
Vendor</td></tr>
<tr><td>Trigger</td><td>The user will request an Add Product to fill up</td></tr>
<tr><td>Description / Main Success Scenario</td><td><table>
<tr><td>1.</td><td>Press “Add” Button</td></tr>
<tr><td>2.</td><td>Provide product add 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 Add Product!!! Notify</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td></td><td>1.1.a. Try Again!!</td></tr>
<tr><td>4.1</td><td>The user did not fill up the details!</td></tr>
<tr><td></td><td>4.1.a. Checked by the system & notified by “Please! Fill Up the Box.”</td></tr>
<tr><td>5.1</td><td>The system did not respond</td></tr>
<tr><td></td><td>5.1.a. Show Error Message.</td></tr>
<tr><td>6.1</td><td>The system doesn’t save the details.</td></tr>
<tr><td></td><td>6.1.a. Notification: “Details did not save.”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>The user will fill up all the details in 30 minutes.</td></tr>
</table>

Table 2.3: Case Description-03: Add Product And Edit

Case Description-04: Edit Profile

<table>
<tr><td>Use Case</td><td>Edit Profile</td></tr>
<tr><td>Goal</td><td>Customer and vendor can edit profiles in the system.</td></tr>
<tr><td>Precondition</td><td>Users must install the E-commerce Management app for registration.</td></tr>
<tr><td>Success End Condition</td><td>Notification: !!!Successfully Edit Profile!!!</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Submission Not Submitted”</td></tr>
<tr><td>Primary Actors:
Secondary Actors:</td><td>Customer
Vendor</td></tr>
<tr><td>Trigger</td><td>The user will request an Edit Profile to fill up</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>Press “Edit Profile” Button</td></tr>
<tr><td>2.</td><td>Provide Edit Profile form</td></tr>
<tr><td>3.</td><td>Enter Information</td></tr>
<tr><td>4.</td><td>Press the “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 Update Profile!!! Notify</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td></td><td>1.1.a. Try Again!!</td></tr>
<tr><td>4.1</td><td>The user did not fill up the details!</td></tr>
<tr><td></td><td>4.1.a. Checked by the system & notified by “Please! Fill Up the Box.”</td></tr>
<tr><td>5.1</td><td>The system did not respond</td></tr>
<tr><td></td><td>5.1.a. Show Error Message.</td></tr>
<tr><td>6.1</td><td>The system doesn’t save the details.</td></tr>
<tr><td></td><td>6.1.a. Notification: “Details did not save.”</td></tr>
</table>
</td></tr>
</table>

Quality Requirements	The user will fill up all the details in 30 minutes.

Table 2.4: Case Description-04: Edit Profile

Case Description-06: Search Product

Use Case	Search Product
Goal	The user can search products in the system.
Precondition	Users must install the E-commerce Management app for registration.
Success End Condition	Notification: !!!N/A!!!
Failed End Condition	Notification: “N/A”
Primary Actors: Secondary Actors:	All User
Trigger	The user will request a Search bar to fill up
Description / Main Success Scenario	1. Press “Search Bar.” 2. Provide Product-Information 3. Enter Information 4. Press the “Search” button. 5. The system searches for the details and shows them!!!

<table>
<tr><td>Alternative Flows</td><td><table>
<tr><td>1</td><td>The user did not fill up the details!</td></tr>
<tr><td></td><td>4.1.a. Checked by the system & notified by “Please! Fill Up the Box.”.</td></tr>
<tr><td>2</td><td>The system did not respond</td></tr>
<tr><td></td><td>Show error message.</td></tr>
<tr><td>3</td><td>The system doesn’t save the details.</td></tr>
<tr><td></td><td>Notification: “Details did not save.”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>The user will fill up all the details in 30 minutes.</td></tr>
</table>

Table 2.4: Case Description-06: Search Product

2.4.3 Activity Diagram

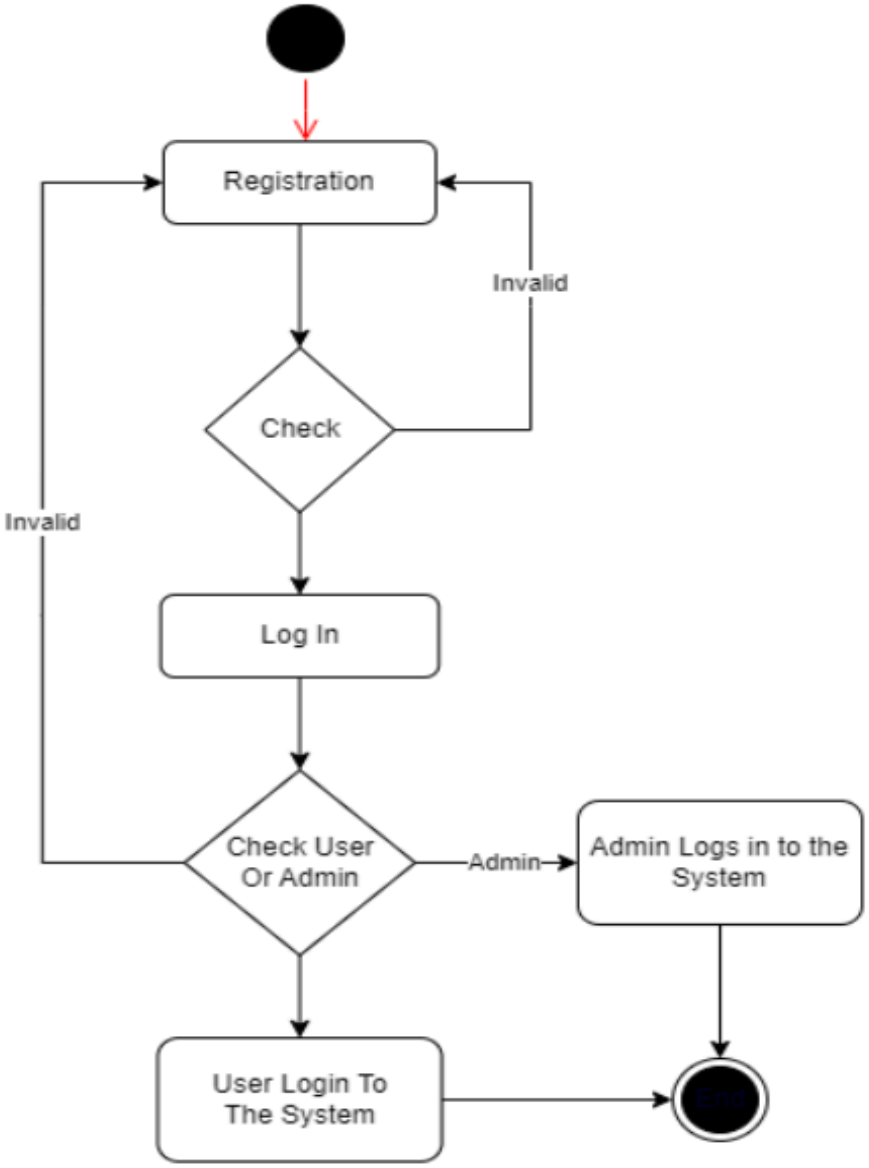


Figure 2.2: Sign Up

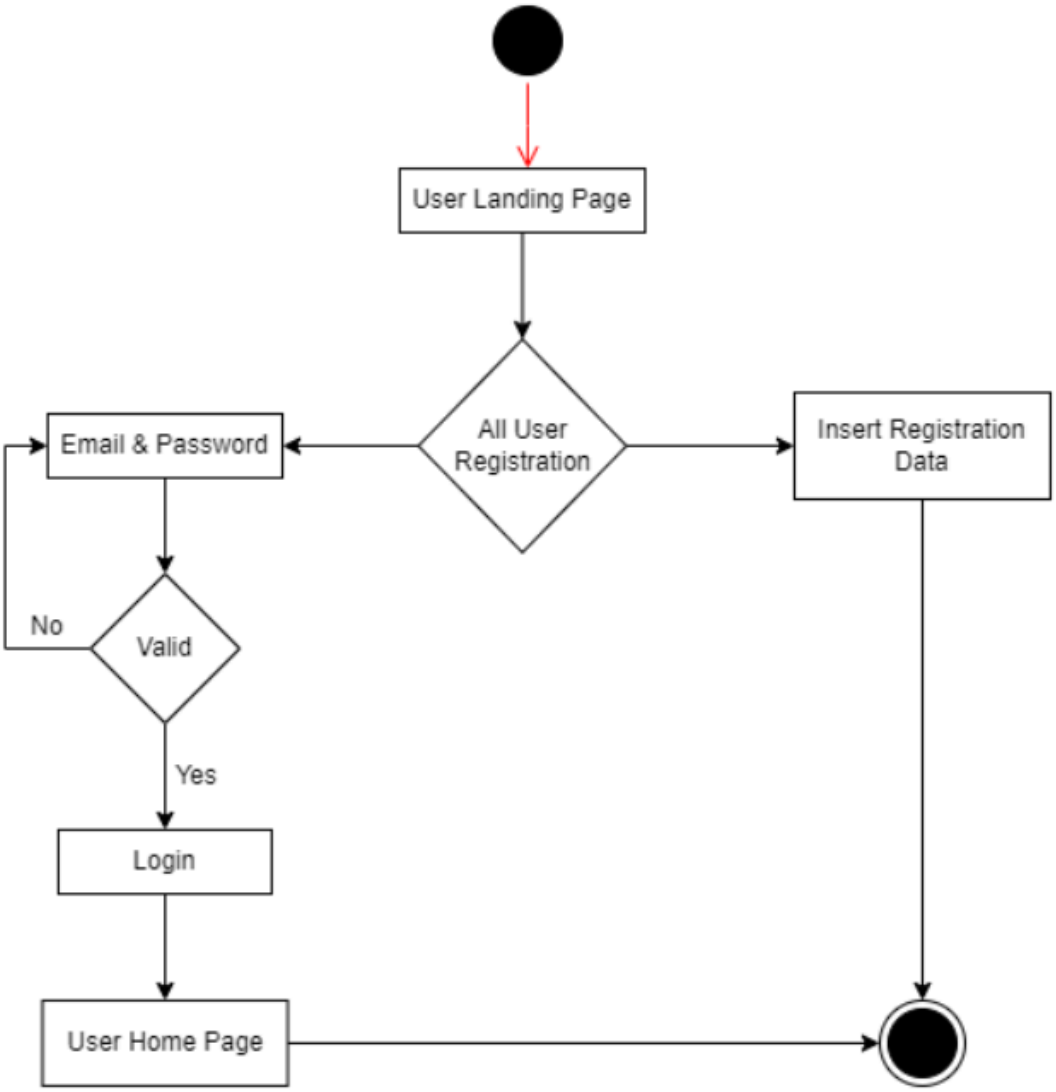


Figure 2.3: Login

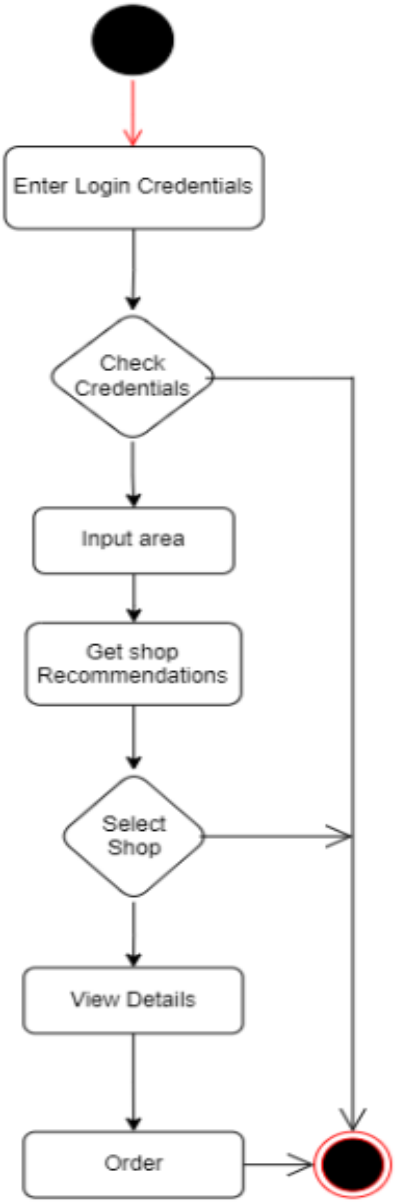


Figure 2.4: Order

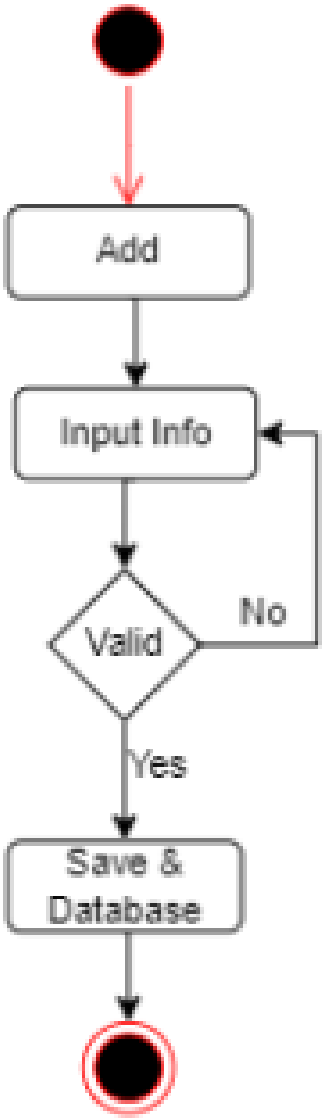


Figure 2.5: Add

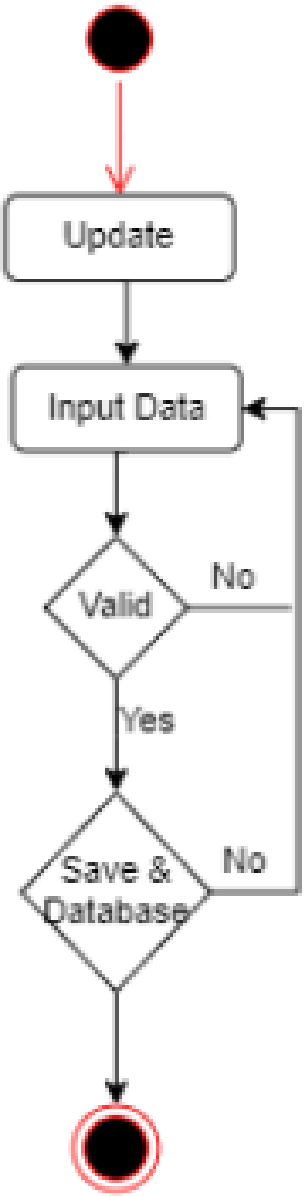


Figure 2.6: Update

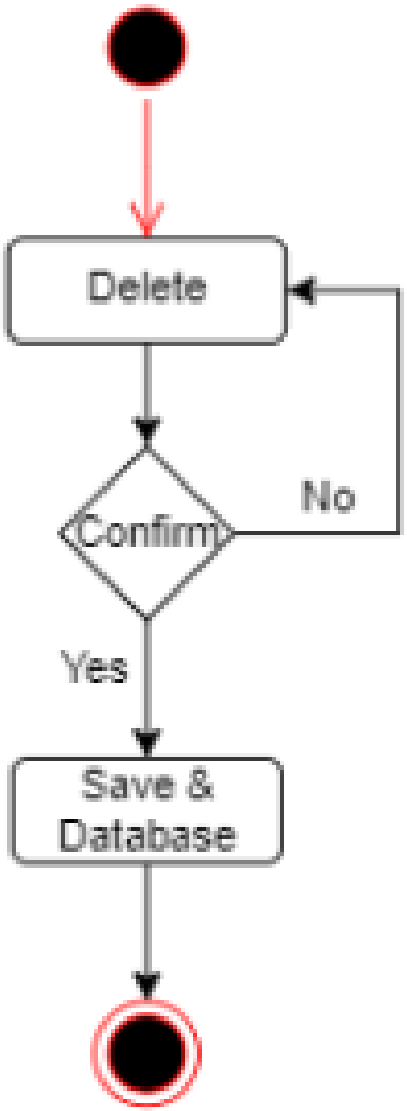


Figure 2.7: Delete

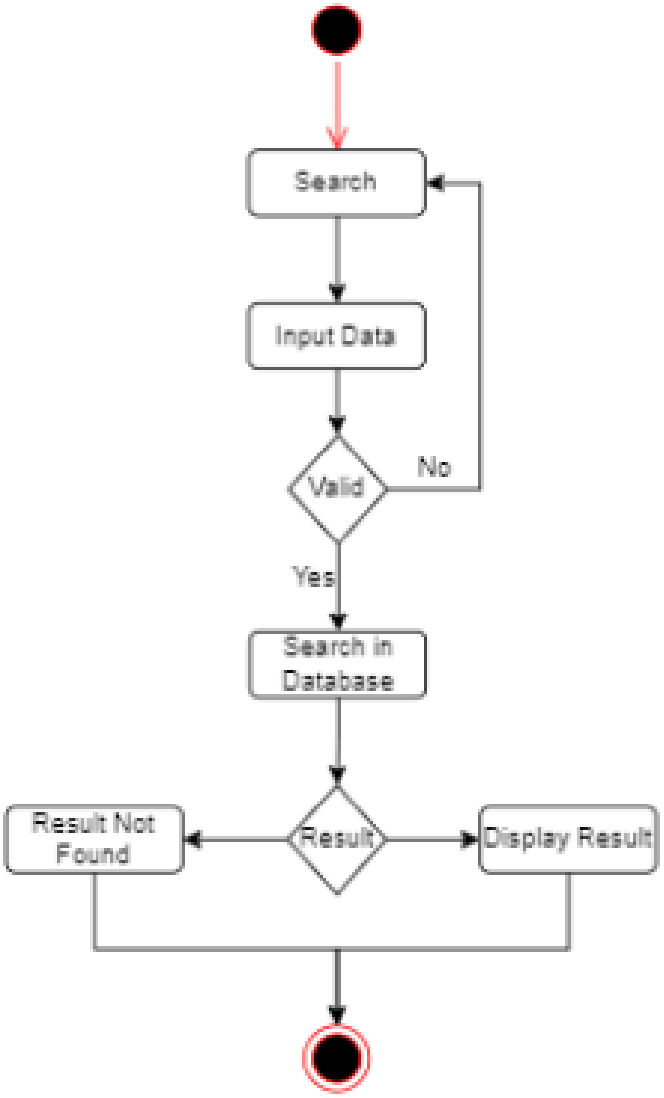


Figure 2.8: Search

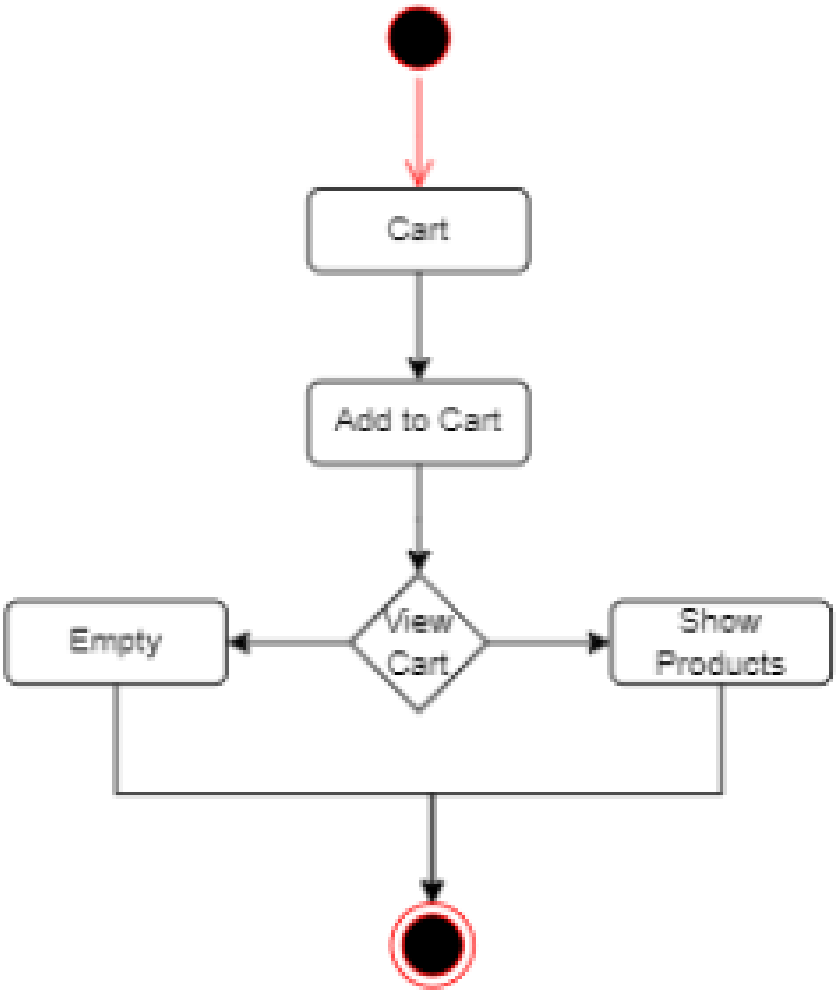


Figure 2.9: Cart

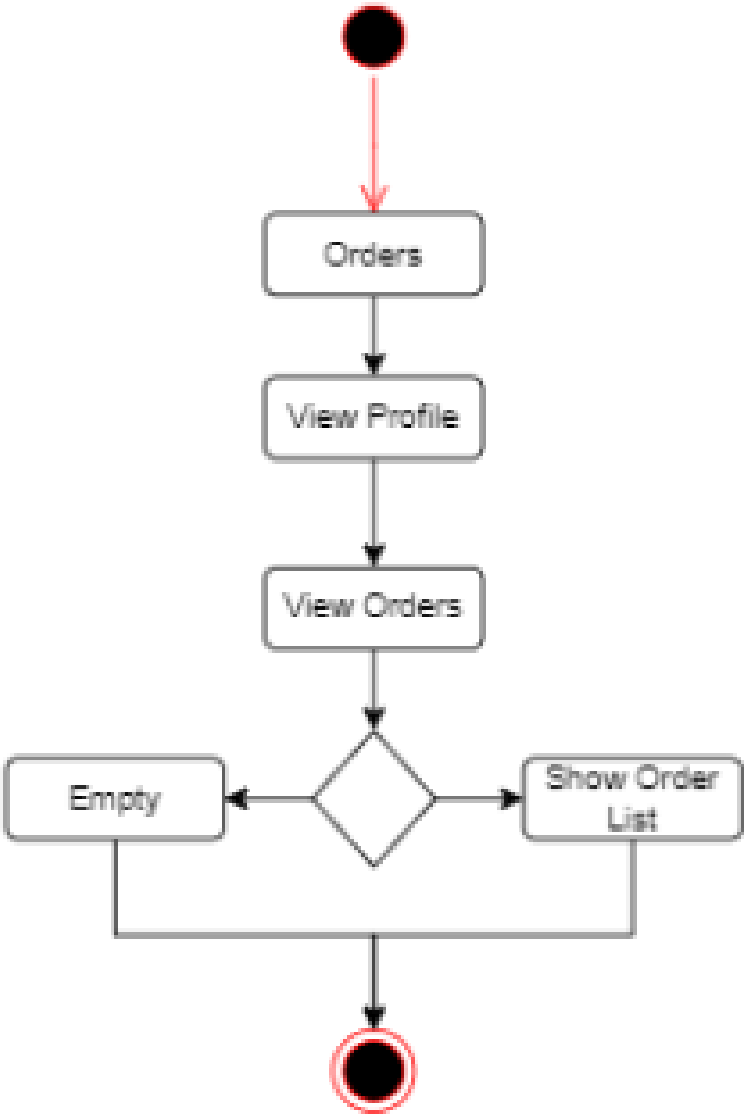


Figure 2.10: Order History

2.4.4 Sequence Diagram

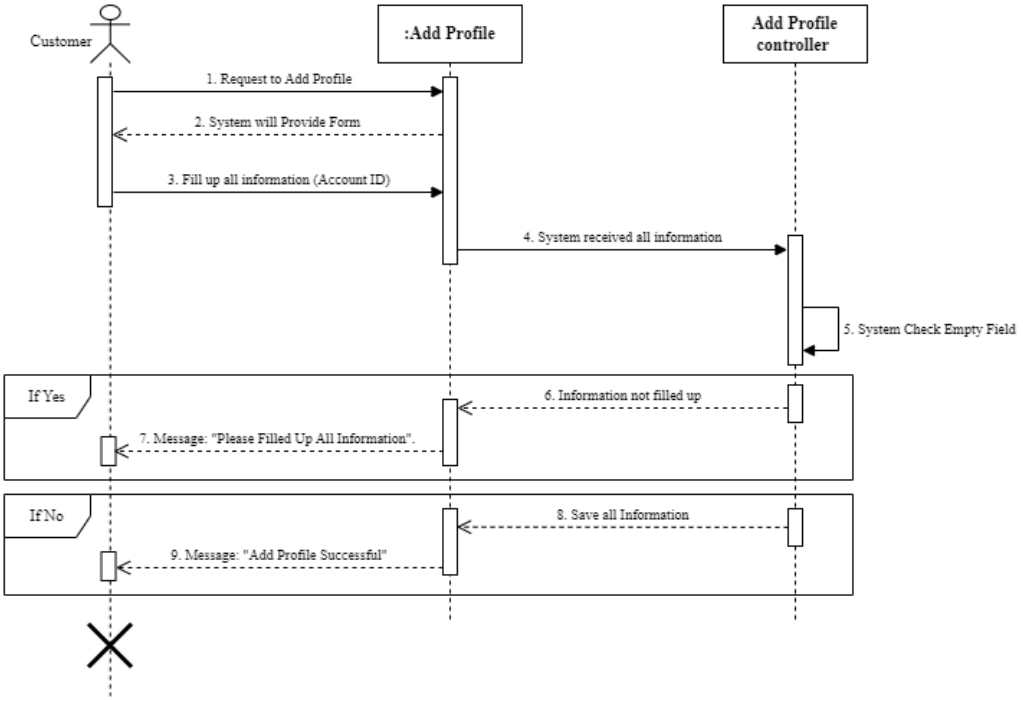


Figure 2.11: Add Profile

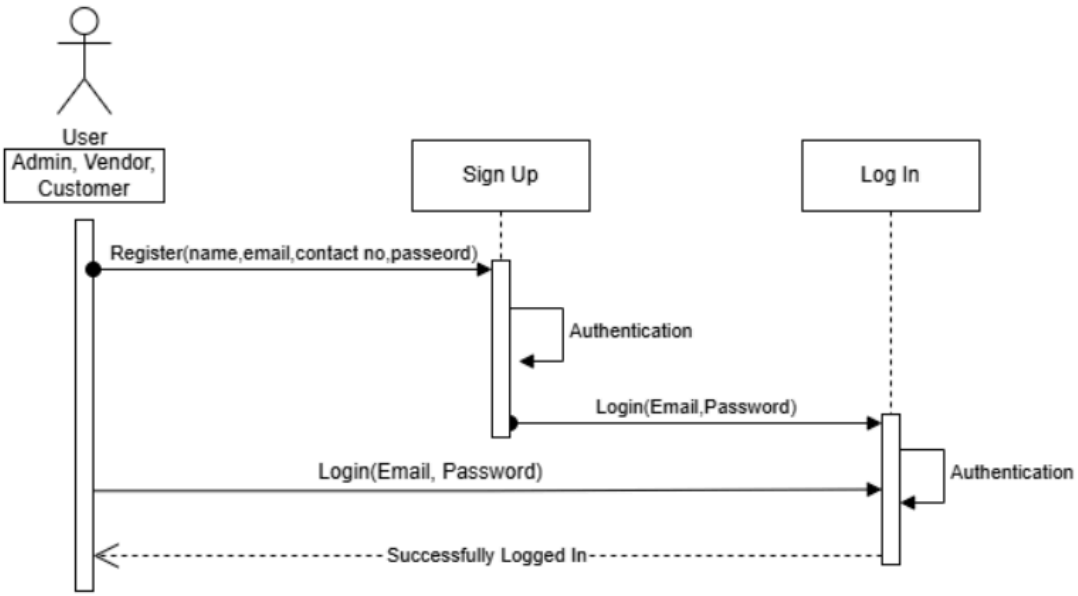


Figure 2.12: User

2.4.5 Class Diagram

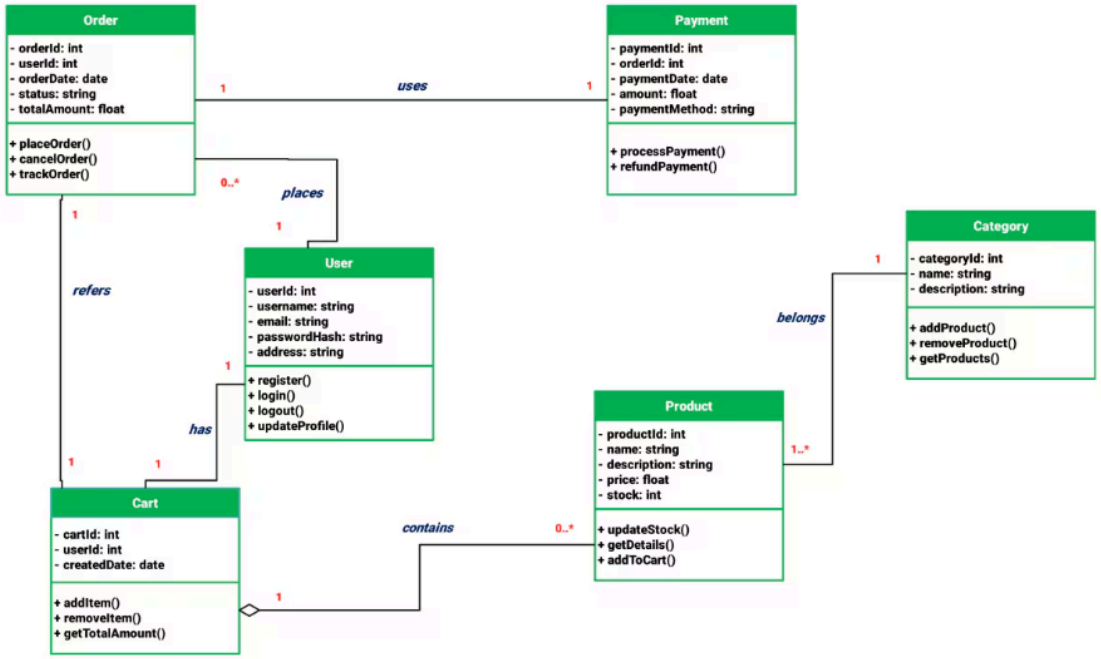


Figure 2.13: Class Diagram

2.4.6 ER Diagram

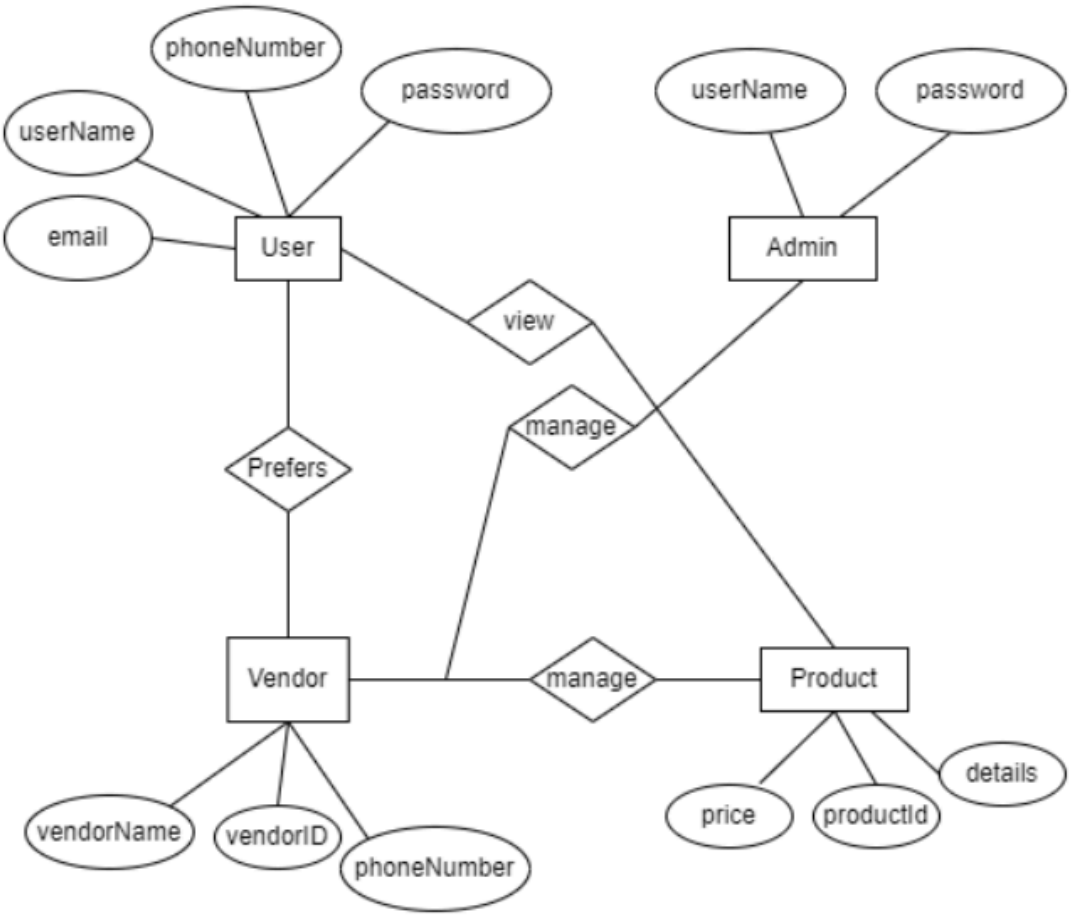
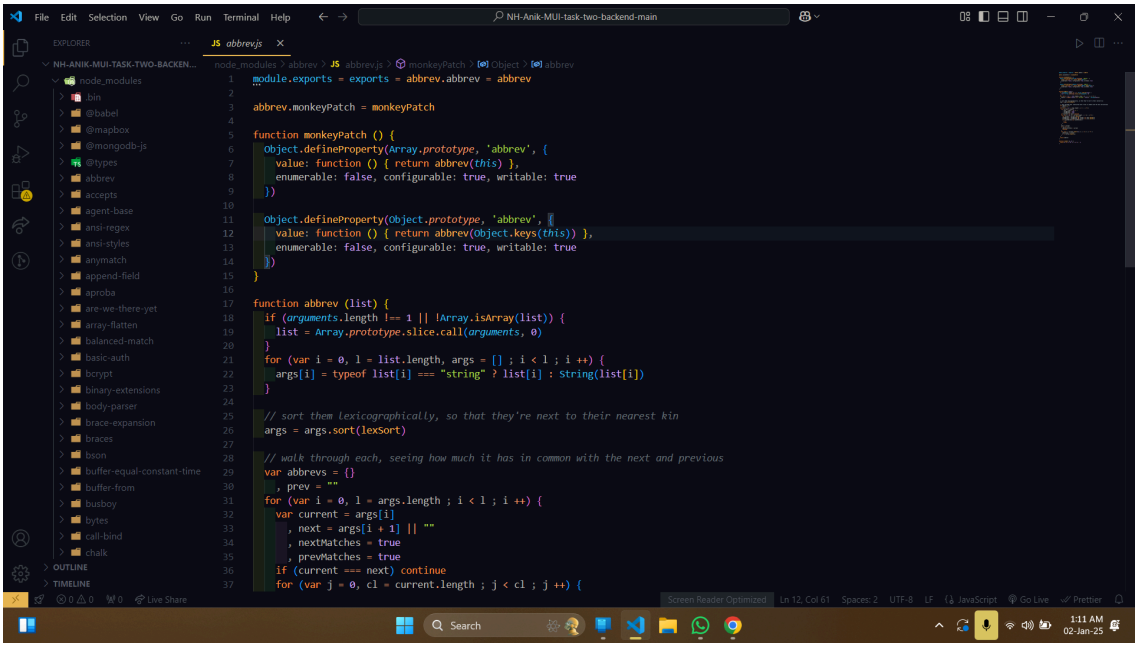


Figure 2.16: ER Diagram

2.5 Coding: Appendix A

Code sample



2.6 Summary

This chapter goes into great length on the e-commerce platform's architecture and implementation. Along with the system architecture and key elements, it explains the primary functions, including user administration, product catalog, shopping cart, and order processing. This chapter also covers database architecture, API development, and the implementation of a robust and secure backend infrastructure. It provides a comprehensive overview of the technical aspects of the project, including design decisions, implementation tactics, and the rationale behind the technology used.

Chapter 3: Software Testing

3.1 Introduction

The rigorous testing techniques used to ensure the performance, quality, and dependability of the developed e-commerce platform are the main topic of this chapter. The testing process is crucial for identifying and resolving any defects or issues before the platform is placed into production. The different testing approaches used, such as user acceptability testing, system testing, integration testing, and unit testing, will be described in this chapter. It will outline the test cases created to cover a range of scenarios, such as regression testing, non-functional testing (security, performance, and usability), and functional testing. Additionally, the usage of test automation tools to improve efficiency and streamline the testing process will be covered in this chapter. The testing process's outcomes will be examined and recorded, emphasizing any problems found and how they were fixed. Through extensive testing, this project seeks to produce a top-notch e-commerce platform that satisfies stakeholders' and users' expectations.

3.2 Testing Features

3.2.1 Feature to Be Tested

This section outlines the key features of the e-commerce platform that will undergo rigorous testing:

- **User Registration:**
  - successfully creating an account with legitimate credentials.
  - input field validation (e.g., email, password, username).
  - handling of errors resulting from multiple accounts or incorrect input.
  - email delivery confirmation and account activation.
- **User Login:**
  - Successfully logged in using legitimate credentials.
  - Handling of errors resulting from improper credentials (wrong password or username).
  - Feature for recovering passwords.
  - Recall my functioning.
- **Vendor Recommendation:**
  - Correctness and applicability of vendor suggestions.
  - The suggestion system's ease of use.

- Influence of browser history and user preferences on suggestions.

- **Admin Login:**
  - Administrators with valid credentials have successfully logged in.
  - Access control for various administrative positions is dependent on roles.
- **Admin Dashboard:**
  - Functionality of every dashboard element, including sales reports, user administration, inventory management, and order management.
  - Quality and integrity of the dashboard's data.
  - Simplicity of use and navigation on the dashboard.
- **Add Vendor (Admin):**
  - New vendors have been successfully added to the platform.
  - Verification of vendor data throughout the registration procedure.
  - Granting new vendors the proper responsibilities and authorizations.
- **User Logout:**
  - Both users' and administrators' logouts are successful.
  - After logging out, cookies are deleted, and the session is invalidated.

The project team can guarantee the e-commerce platform's usability, performance, and dependability by methodically testing these essential elements.

3.3 Testing Strategies

3.3.1 Test Approach

To provide thorough coverage of the e-commerce platform's functionalities, the testing procedure will combine automated and manual testing approaches.

- **Manual Testing:**
  - **Exploratory Testing:** In order to find possible problems and opportunities for development, testers will investigate the platform's features and functionalities in an unstructured way.
  - **Usability Testing:** To assess the platform's overall usability, intuitiveness, and ease of use, user interviews and observations will be carried out.
  - **Compatibility Testing:** The platform will be tested across a range of devices, including desktops, laptops, tablets, smartphones, and operating systems, as well as browsers, including as Chrome, Firefox, Safari, and Edge, to ensure compatibility and a consistent user experience.

- **Automated Testing:**
    - **Unit Testing:** To guarantee proper operation, each platform module and component will be tested separately.
    - **Integration Testing:** To make sure various modules and components interact as intended, their interactions will be examined.
    - **Regression Testing:** To make sure that current functionality is maintained, automated tests will be run following every code modification or problem patch.
    - **API Testing:** To make sure that APIs are operating as intended and returning the anticipated data, their functionality will be examined.
- **Test Automation Tools:**
    - **Selenium:** For doing functional testing and automating web browser interactions.
    - **Jest:** For JavaScript code unit testing.
    - **Postman:** For testing APIs, use Postman.
- **Test Environments:**
    - To reduce the possibility of affecting production systems, distinct test environments will be established for development, testing, and staging.

3.3.2 Pass/Fail Criteria

Each test case's pass/fail status will be decided based on the following criteria:

- **Functionality:**
    - According to the functional requirements, every essential feature of the e-commerce platform must operate as intended.
    - Every user interaction needs to be completed successfully, including registration, login, product browsing, shopping cart activities, and checkout.
    - Authorized administrators must be able to access and use all administrative features, including order, product, and user management.
- **Performance:** The platform must fulfill the specified performance standards, including load handling capacity, throughput, and reaction times.

    - To make sure the platform can manage high traffic loads and maintain respectable response times, performance testing will be done.
- **Usability:**
    - The platform should be simple to use and navigate for all users, including administrators and clients.
    - Usability testing will be done to identify and address any usability issues or areas that require improvement.
- **Security:**
    - The platform must adhere to the established security standards, which include fraud prevention, access control, and data protection.
    - To find and fix any security flaws, security testing will be carried out.
- **Reliability:**
    - The platform must exhibit constant performance, low downtime, and high availability and reliability.
    - To evaluate the platform's capacity to bounce back from malfunctions and keep services available, reliability testing will be carried out.
- **Compatibility:**
    - The platform needs to work with a variety of devices, operating systems, and browsers.
    - Compatibility testing will be done to provide a consistent user experience and performance across all platforms.
- **Accessibility:**
    - The platform must comply with accessibility guidelines (such as WCAG) and be usable by people with disabilities.
    - To find and fix any accessibility issues, accessibility testing will be done.

3.4 System Testing (Test Cases with Report)

Table 3.1: Test Case For Login

Test case #1	**Test case name: Log in**
System: Alfa Mart	**Subsystem: N/A**
Designed by: Md Saiduzzaman	**Designed date:** 02/08/2024
Executed by: Md Saiduzzaman	**Executed date:** 03/08/2024
Short description: If the user enters the right info, they will be able to successfully log in.	
Pre-conditions: • Admin must add the user. • Assume that the password is "12345678" and the email address is "niamul35-2252@diu.edu.bd."	

Step	**Email**	**Password**	**Expected result**	**Pass/Fail**	**Comment**
1	saiduzzaman35@	pass	These credentials don't match what we have on file.	Fail	
2	saiduzzaman35-2267 @diu.edu.bd		Password required.	Fail	
3		12345678	Email field is required.	Fail	

4	saiduzzaman35-2267@diu.edu.bd	12345678	Login successfully	Pass	
Post-conditions: User may access homepage and several choices after logging in.					

Table 3.2: Test Case For Add User

Test case #2		**Test case name:** Add User		
System: Alfa Mart		**Subsystem: N/A**		
Designed by: Md Saiduzzaman		**Designed date:** 10/08/2024		
Executed by: Md Saiduzzaman		**Executed date:** 11/08/2024		
Short description: Initially, only the administrator may add a client or worker.				
Pre-conditions: Admin must be login.				
Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	clients To add workers or Clients, click Admin.	Complete all necessary fields	Pass	
2	After selecting Add, a new user will be added.	To the same page, redirect	Pass	
Post-conditions: Pass				

Table 3.3: Test Case For Update User Profile

Test case #3		**Test case name:** Update user profile		
System: Alfa Mart		**Subsystem: N/A**		
Designed by: Md Saiduzzaman		**Designed date:** 20/08/2024		
Executed by: Md Saiduzzaman		**Executed date:** 21/08/2024		
Short description: Each user can update the necessary information.				
Pre-conditions: • Logging in is required. • That user must fill out all essential fields.				
Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	Not yet filled out are all necessary fields.	Complete all necessary fields	Pass	
2	The user fills up every input field.	That data will be saved by the program.	Pass	
Post-conditions: The application program's database is correctly configured to hold information about profile updates.				

Table 3.4: Test Case For Make Income Report

Test case #4		**Test case name:** Make income report		
System: Hire a Worker		**Subsystem: N/A**		
Designed by: Md Saiduzzaman		**Designed date:** 22/09/2024		
Executed by: Md Saiduzzaman		**Executed date:** 23/09/2024		
Short description: After completing all necessary areas, the admin can report sales.				
Pre-conditions: • Admin and Vendor must be login.				
Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	Some of the essential fields have not yet been completed.	Fields should not be empty.	Pass	
2	The sales completes all sales inputs.	The application will store that information.	Pass	
Post-conditions: Admin information stored in the application program's database.				

Table 3.5: Test Case For Admin can view all reports

Test case #5	**Test case name:** Admin can view all reports
System: Alfa Mart	**Subsystem: N/A**
Designed by: Md Saiduzzaman	**Designed date:** 27/09/2024
Executed by: Md Saiduzzaman	**Executed date:** 28/09/2024

Short description: All reports given by workers or clients can be viewed by the administrator.

Pre-conditions:

- Admin must login first.

Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	The report is not seen.	Please try again.	Fail	
2	The report is seen to be a success.	All of the data will be visible to the application.	Pass	

Post-conditions: administrator access worker or customer data.

Table 3.6: Test Case For Customer buy product

Test case #6		**Test case name:** Customer buy product		
System: Alfa Mart		**Subsystem: N/A**		
Designed by: Md Saiduzzaman		**Designed date:** 17/10/2024		
Executed by: Md Saiduzzaman		**Executed date:** 18/10/2024		
Short description: after publishing an offer and filling out all required elements.				
Pre-conditions: • The program requires the user to log in and authenticate their identity.				
Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	There are still some necessary fields that need to be filled out.	Fields cannot be left unfilled.	Pass	
2	The vendor completes the inputs that are provided.	The software will save such details.	Pass	
Post-conditions: The database of the application software appropriately maintains client data.				

Table 3.7: Test Case For Monthly Report

Test case #7		**Test case name:** Monthly Report		
System: Alfa Mart		**Subsystem: N/A**		
Designed by: Md Saiduzzaman		**Designed date:** 25/10/2024		
Executed by: Md Saiduzzaman		**Executed date:** 26/1/2024		
Short description: The vendor can produce the equipment income report after filling out all the essential data.				
Pre-conditions: • Using the program, the vendor must log in and establish their identity.				
Step	**Email**	**Response**	**Pass/Fail**	**Comment**
1	Some of the essential fields have not yet been completed.	Fields cannot be left unfilled.	Pass	
2	The vendor completes all supplied inputs.	The software will save such details.	Pass	
Post-conditions: In his panel, the user will successfully log in.				

3.5 Summary

This chapter describes the thorough testing approach used to guarantee the created e-commerce platform's performance, quality, and dependability. The automated and manual testing techniques utilized in the testing process include user acceptability testing, system testing, integration testing, and unit testing.

The chapter outlines the main functionalities that will be examined, including order management, product catalog, shopping cart, payment gateway integration, and user registration. It describes the testing methodology, which includes the use of automated testing tools such as Selenium and Jest, exploratory testing, and usability testing.

To guarantee that the platform satisfies the specified functional and non-functional requirements, such as performance, security, usability, dependability, and compatibility, explicit pass/fail criteria have been set for every test scenario.

By conducting comprehensive testing at every level of development, the project aims to identify and address any issues early on, ensuring the delivery of an excellent, dependable, and user-friendly e-commerce platform that meets the needs and expectations of businesses and customers alike.

The main points discussed in the "Software Testing" chapter are succinctly summarized here, emphasizing the significance of thorough testing for the e-commerce platform project's success.

Chapter 4: Deployment and Maintenance

4.1 Introduction

The implementation and continuous upkeep of the created e-commerce platform are the key topics of this chapter. It describes the deployment process, which includes choosing an appropriate hosting environment, following the deployment steps, and setting up the required infrastructure.

This chapter also covers the platform's continuous maintenance needs, such as frequent updates, security patches, bug fixes, and performance enhancements. In order to guarantee the platform's continuous stability, dependability, and performance, it talks about how crucial monitoring and maintenance tasks are.

This chapter offers a thorough rundown of the project's deployment and maintenance components, guaranteeing the e-commerce platform's long-term, profitable operation.

The reader is prepared for the discussion of the e-commerce platform's deployment and maintenance features by this introduction, which offers a succinct and educational summary of the chapter's subject.

4.2 Try to follow the SRLC (software release life cycle)

1. **Planning:**

- Define the scope of your e-commerce platform (e.g., B2C, B2B, specific products or services).
- Identify key features (e.g., product catalog, shopping cart, payment gateway, user accounts).
- Estimate resources (e.g., developers, designers, testers) and timelines.

2. **Design:**

- Choose an architecture (e.g., monolithic, microservices).
- Design the database schema.
- Create UI/UX mockups.
- Select technologies (e.g., programming languages, frameworks, cloud platforms).

3. **Development:**

- Write code for core functionalities.
- Develop APIs for frontend-backend communication.
- Integrate payment gateways and other third-party services.

4. **Testing:**
- To validate individual components, run unit tests.
- To make sure parts function together, do integration tests.
- Test the system to replicate real-world use.

5. **Deployment:**
- Deploy the platform to a staging or production environment.
- Monitor system performance and user feedback.
- Provide support to users through documentation or customer service.

6. **Maintenance:**
- Take care of security flaws and bugs.
- Based on consumer input and industry trends, add new features.
- Upgrade technologies and infrastructure to improve performance and security

Chapter 5 User Manual

5.1 Introduction

A thorough user manual for the e-commerce platform is provided in this chapter. The purpose of this handbook is to instruct administrators and users alike on how to make the most of the platform's many features and capabilities.

The user manual covers a wide range of topics, including account administration, product browsing and search, shopping cart management, checkout processes, order tracking, customer support, and user registration and login. It also teaches administrators how to manage users, orders, goods, and other platform components via the admin interface.

The purpose of this user manual is to equip consumers with the information and abilities they need to make efficient use of the e-commerce platform and improve their overall online buying experience.

5.2 Project Functionalities

Login

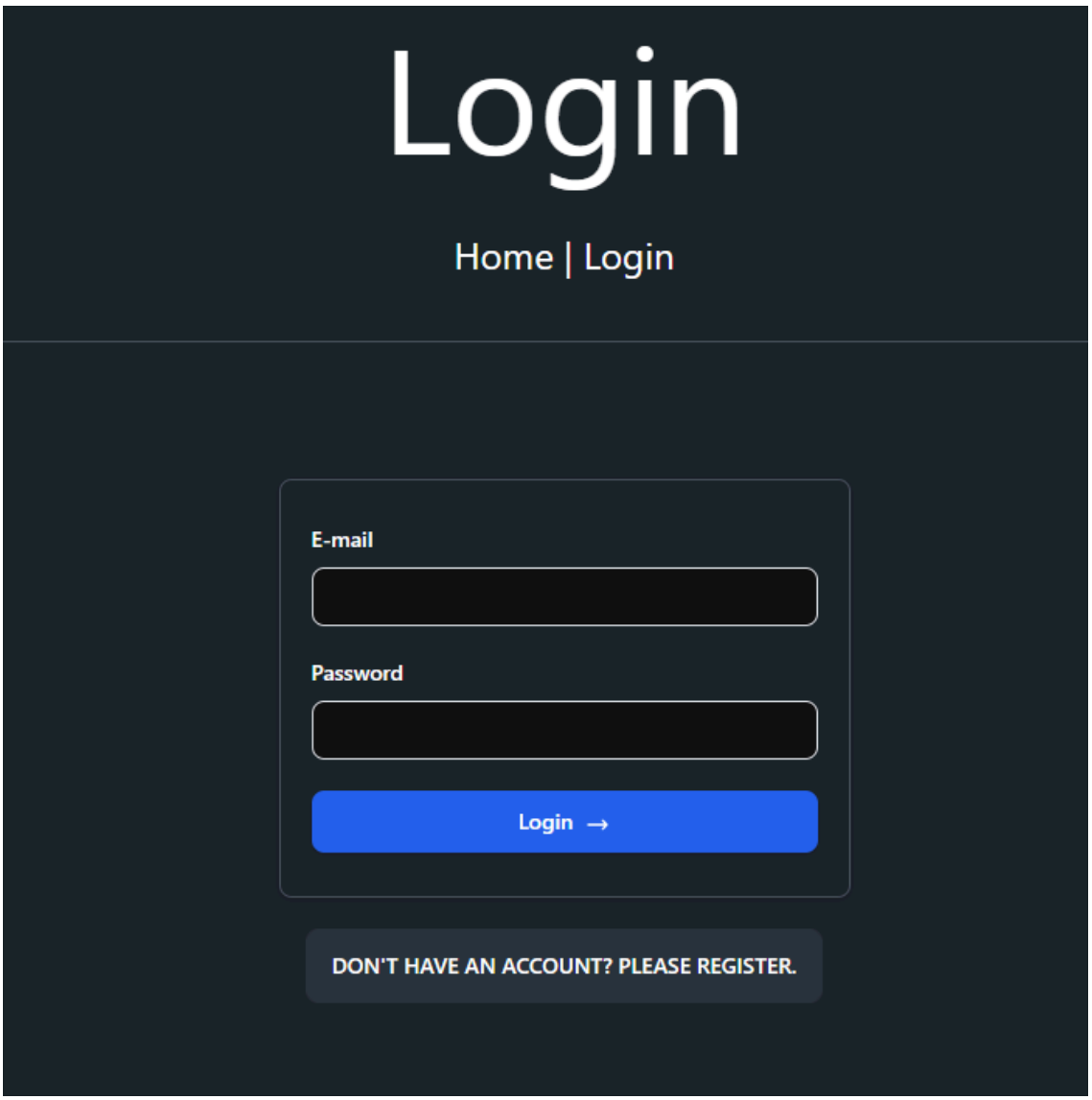


Figure 5.1: Login

Registration

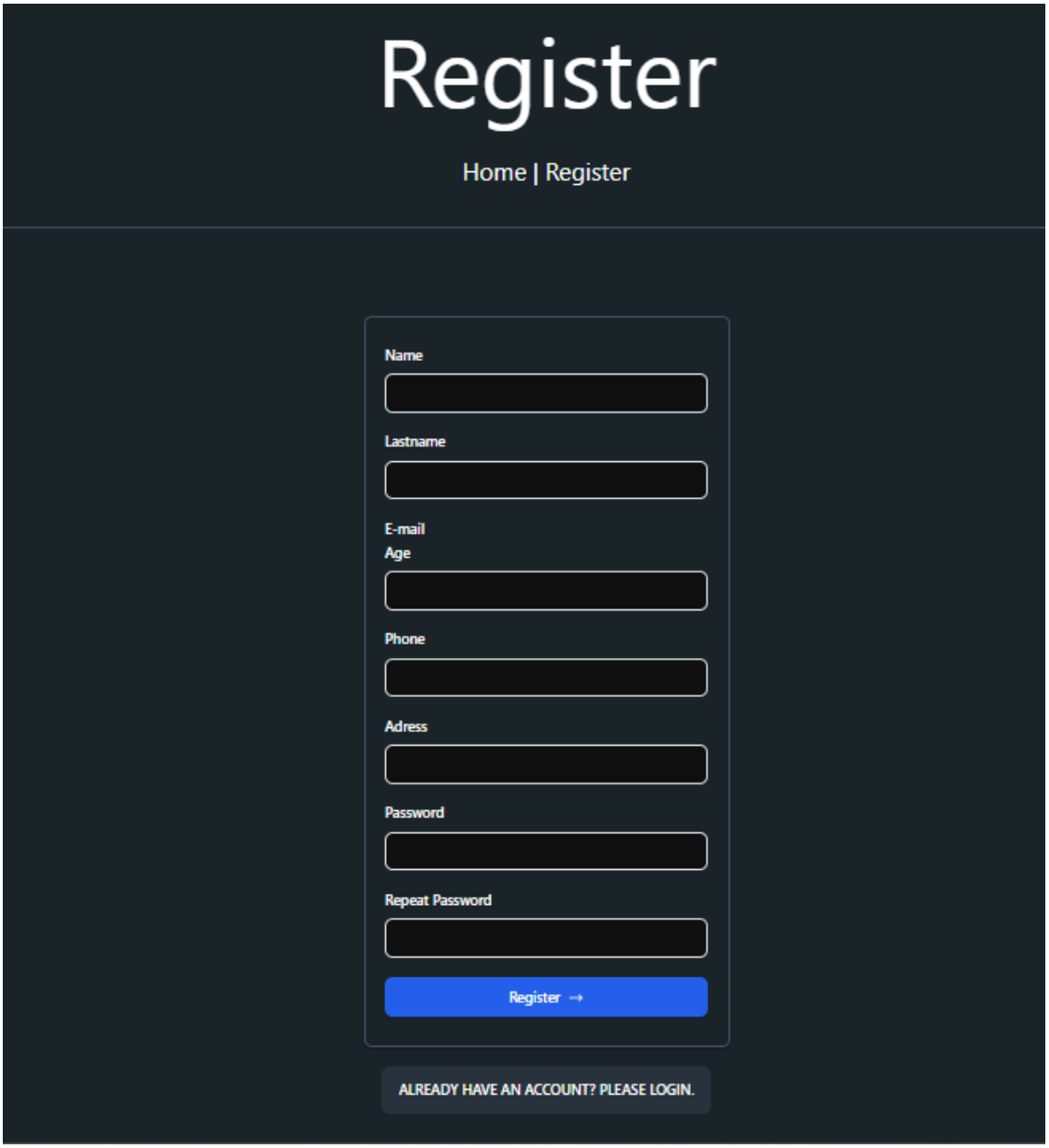


Figure 5.2: Registration

User Interface

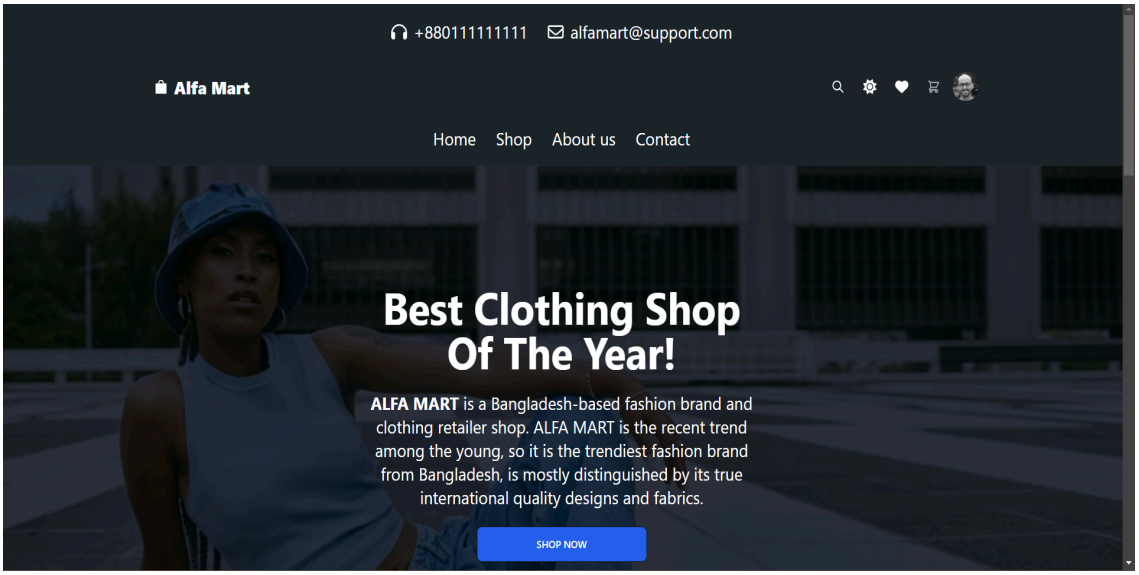


Figure 5.3: User Interface

User Interface

Figure 5.4: User Interface

User Profile

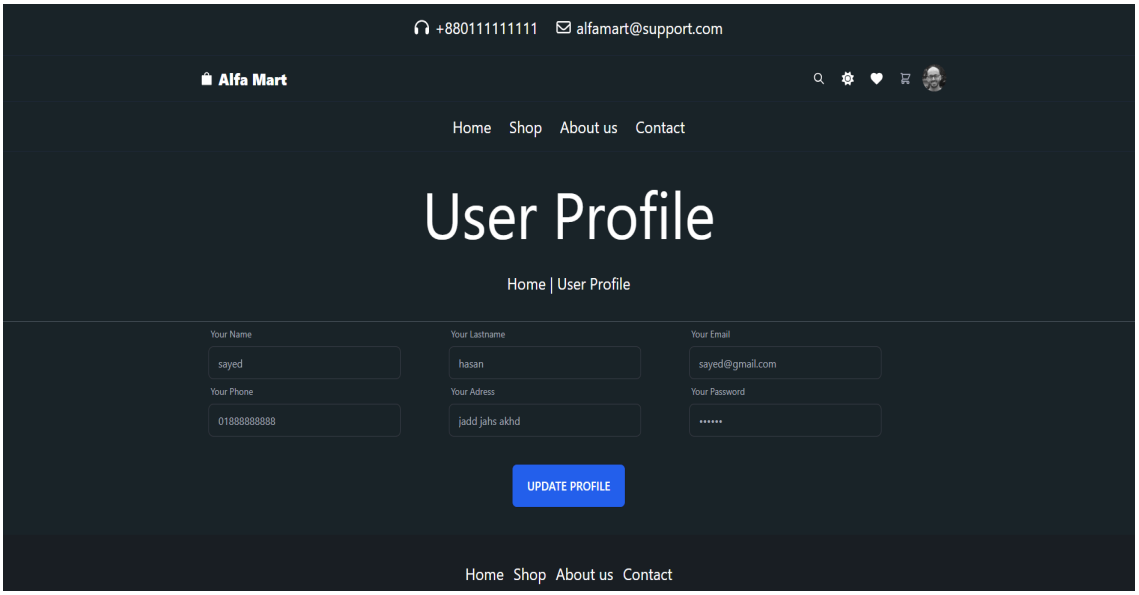


Figure 5.5: User Profile

5.3 Summary

This chapter offers crucial guidance on how to handle your Alfa Mart e-commerce account efficiently. Do not hesitate to contact customer service for assistance if you run into any problems.

Chapter 6 Project Summary

6.1 Introduction

A thorough overview of the complete e-commerce platform development project is given in this chapter. It goes over the goals of the project, its scope, its methodology, and the main conclusions drawn from its design, development, testing, and deployment phases.

The project's major accomplishments, such the platform's successful development, will be highlighted in the summary. It will also go over the difficulties that arose during the project and the methods that were used to get beyond them.

This chapter will include closing thoughts and possible future paths for the e-commerce platform, including future integrations, possible improvements, and chances for additional growth.

6.2 Project Limitation

While the e-commerce platform successfully incorporates many key features and functionalities, certain limitations were encountered during the development process:

- **Time Restrictions:** Due to the project's short timeline, features had to be prioritized, and certain intended functionalities might not have been able to be implemented.
- **Resource Constraints:** The project's progress was occasionally hampered by the lack of human resources with particular skill sets (such as database administrators and front-end developers).
- **Financial Restrictions:** Financial constraints might have limited the application of specific technologies or hindered the deployment of more sophisticated features, including sophisticated AI/ML capabilities for tailored suggestions.
- **Scope creep:** modifications to the project's requirements throughout the development stage could have resulted in unanticipated difficulties and affected the project's budget and schedule.

Furthermore, certain features and functionalities were either partially implemented or not implemented due to time and resource constraints:

- **Social Media Integration:** Features such as product sharing, user login, and social commerce were not completely integrated with social media platforms.
- **Advanced Analytics and Reporting:** The scope of the creation of dashboards for advanced analytics and reporting that provide detailed business insights was constrained.
- **Mobile App Development:** This project did not include the creation of a specific mobile application for the e-commerce platform.

During the project planning and development stages, these constraints were carefully taken into account, and mitigation techniques were put in place to lessen their influence on the project's overall success.

6.3 Scope

This project focuses on the development of an e-commerce platform.

Included Features and Modules:

- **User Management:** Order history, profile management, login, and user registration.
- **Product Catalog:** Product details, product classification, search and filtering, and product listing.
- **Shopping Cart:** Items can be added to the cart, removed, quantities changed, coupons applied, and shipping costs computed.
- **Checkout Process:** Order tracking, order confirmation, and secure payment processing integration.
- **Inventory Management:** Monitor inventory levels, control product variants, and send out low-stock notifications.
- **Order Management:** Process orders, manage order fulfillment, and generate invoices.
- **Admin Dashboard:**
    - Product administration includes managing product categories and adding, editing, and removing goods.
    - Order management: Track shipments, see and manage orders, and handle refunds.
    - User management: control permissions, roles, and user accounts.
    - Analytics and reporting on sales.

Excluded Features:

- **Mobile App Development:** The project does not include the development of a native mobile application for iOS or Android.
- **Social Media Integration:** Deep integration with social media platforms for login, sharing, and marketing purposes is not within the scope of this project.
- **Advanced AI/ML Features:** Advanced AI/ML capabilities such as personalized product recommendations, predictive analytics, and chatbot integration are not included in this initial phase.
- **Marketplace Functionality:** The platform does not include marketplace functionality, where third-party vendors can list and sell their products.

6.4 Future Work

This project provides a strong foundation for a robust and scalable e-commerce platform. However, several areas offer potential for future enhancements and upgrades:

- **Enhance User Experience:**
    - Put in place more sophisticated customization tools, such as tailored product suggestions based on customer browsing and purchasing patterns.
    - Incorporate virtual reality (VR) or augmented reality (AR) technology to improve the experience of perusing products.
    - Develop a more intuitive and engaging user interface with improved navigation and visual appeal.
- **Expand Functionality:**
    - Integrate with social media platforms for user login, product sharing, and social commerce features.
    - Develop a mobile app for enhanced user experience and accessibility on mobile devices.
    - Implement a loyalty program to reward repeat customers and incentivize repeat purchases.
    - Integrate a live chat feature for real-time customer support.
- **Enhance Performance and Scalability:**
    - Implement caching mechanisms to improve page load times and reduce server load.
    - Explore cloud-native technologies and serverless architectures for improved scalability and cost-efficiency.
    - Conduct load testing and performance tuning to optimize system performance and ensure high availability.
- **Improve Security:**
    - To improve the platform's security posture, put sophisticated security measures, including fraud prevention techniques and intrusion detection systems, into place.
    - To find and fix any vulnerabilities, do routine penetration tests and security audits.
- **Integrate AI/ML:**
    - Leverage AI/ML algorithms for tasks such as image recognition, sentiment analysis, and predictive analytics to improve the overall user experience and business intelligence.

- **Explore Blockchain Technology:**
    - Examine how blockchain technology may be used for supply chain management, safe and transparent transactions, and fostering consumer confidence.

These areas offer exciting opportunities for further development and improvement of the e-commerce platform, enhancing its capabilities and providing a more competitive edge in the market.

6.5 Conclusion

Incorporating essential features like user registration, product catalogs, shopping carts, secure payment processing, and order management, this project effectively created an e-commerce platform. To provide a streamlined and effective online purchasing experience, the platform makes use of a strong and scalable design.

Key achievements of the project include:

- **Development of a functional and user-friendly e-commerce platform:** Meeting the core functional and non-functional requirements outlined in the project scope.
- **Implementation of a robust and scalable backend infrastructure:** Ensuring the platform can handle increasing traffic and data volumes.
- **Integration with key third-party services:** Enabling seamless payment processing, order fulfillment, and other essential functionalities.
- **Conducting thorough testing and quality assurance:** Identifying and addressing potential issues before deployment.

Key Lessons Learned:

- **Importance of thorough planning and requirements gathering:** The importance of clearly defining project objectives, gathering comprehensive user requirements, and conducting thorough feasibility studies was emphasized throughout the project.
- **Value of agile development methodologies:** The project benefited from an agile development approach, allowing for flexibility, adaptability, and continuous improvement.
- **Critical role of effective communication and collaboration:** Maintaining clear and consistent communication among team members was crucial for project success.
- **Importance of rigorous testing:** Comprehensive testing throughout the development lifecycle played a vital role in identifying and resolving issues, ensuring the quality and reliability of the final product.

Meeting Project Objectives:

The project successfully met its primary objectives, including:

- **Developing a user-friendly and scalable e-commerce platform.**
- **Providing businesses with a platform to establish and grow their online presence.**
- **Delivering a high-quality and reliable online shopping experience for customers.**

This project has demonstrated the successful development and implementation of a functional and user-friendly e-commerce platform. The lessons learned and the valuable experience gained throughout the project will be instrumental in future software development endeavors.

REFERENCES

Use a reference manager such as *Mendeley*, *EndNote,* or any reference manager software to generate all your list of references here. Once all the references are included, then apply *Caption for Reference* style.

[1] J. K. Author, "Title of paper," *Abbreviated Title of Periodical*, vol. x, no. x, pp. xxx-xxx, Month, year, doi: xxx.

[2] J. K. Author, *Book Title*. City, State: Publisher, year.

[3] J. K. Author, "Title of paper," in *Abbreviated Name of Conf.*, (location of conference is optional), (Month and day(s) if provided) year, pp. xxx-xxx.

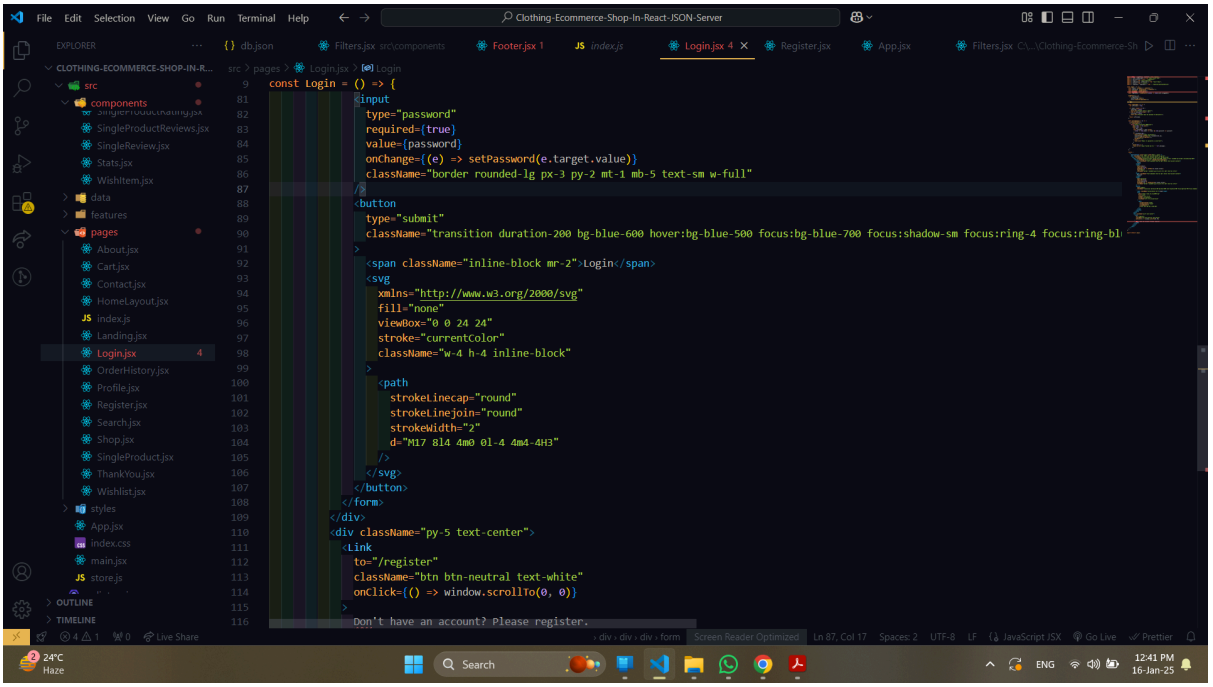
[4] First Name Initial(s). Last Name. "Page Title." *Website Title*. Web Address (accessed Date Accessed).

[5] University Course Material, "Software Engineering References and Notes," Internal documentation provided by Daffodil International University, 2024.

[6] W3Schools, "HTML, CSS, and JavaScript Tutorials," [Online]. Available: https://www.w3schools.com. [Accessed: Dec. 30, 2024].

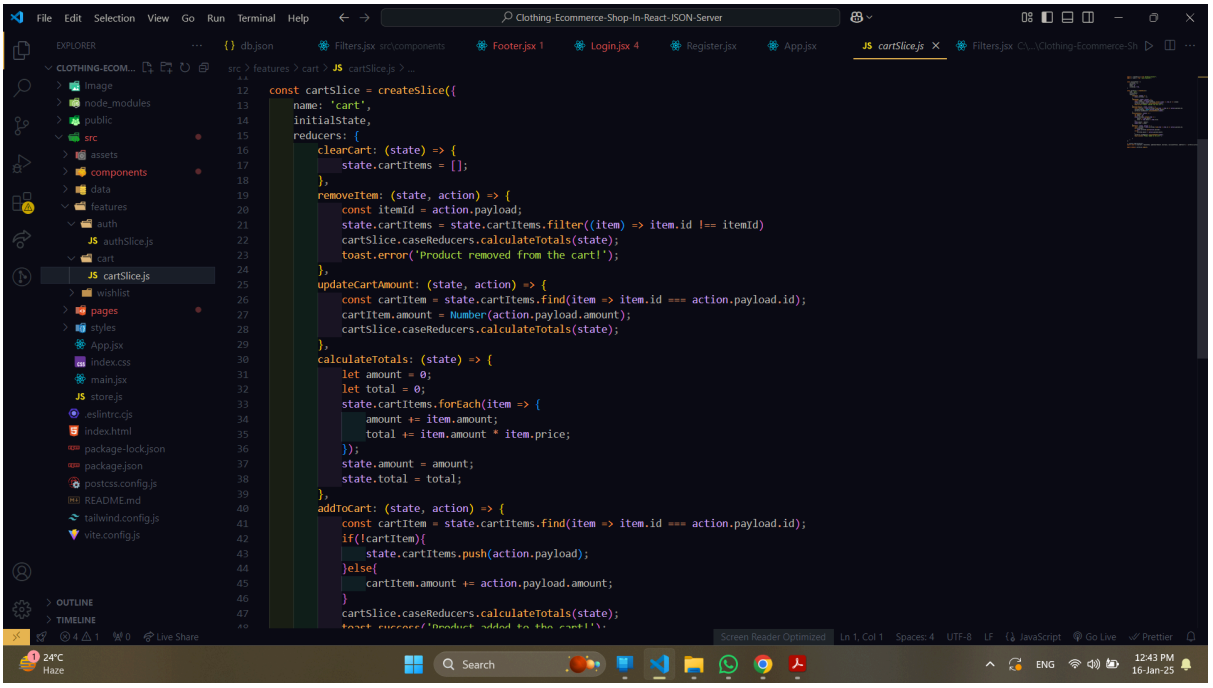
APPENDICES

Appendix B: Code Sample



```
const Login = () => {
  <input
    type="password"
    required={true}
    value={password}
    onChange={(e) => setPassword(e.target.value)}
    className="border rounded-lg px-3 py-2 mt-1 mb-5 text-sm w-full"
  />
  <button
    type="submit"
    className="transition duration-200 bg-blue-600 hover:bg-blue-500 focus:bg-blue-700 focus:shadow-sm focus:ring-4 focus:ring-bl
  >
    <span className="inline-block mr-2">Login</span>
    <svg
      xmlns="http://www.w3.org/2000/svg"
      fill="none"
      viewBox="0 0 24 24"
      stroke="currentColor"
      className="w-4 h-4 inline-block"
    >
      <path
        strokeLinecap="round"
        strokeLinejoin="round"
        strokeWidth="2"
        d="M17 8l4 4m0 0l-4 4m4-4H3"
      />
    </svg>
  </button>
</form>
</div>
<div className="py-5 text-center">
  <Link
    to="/register"
    className="btn btn-neutral text-white"
    onClick={() => window.scrollTo(0, 0)}
  >
    Don't have an account? Please register.
```

Appendix C: Code Sample



```
const cartSlice = createSlice({
    name: 'cart',
    initialState,
    reducers: {
        clearCart: (state) => {
            state.cartItems = [];
        },
        removeItem: (state, action) => {
            const itemId = action.payload;
            state.cartItems = state.cartItems.filter((item) => item.id !== itemId)
            cartSlice.caseReducers.calculateTotals(state);
            toast.error('Product removed from the cart!');
        },
        updateCartAmount: (state, action) => {
            const cartItem = state.cartItems.find(item => item.id === action.payload.id);
            cartItem.amount = Number(action.payload.amount);
            cartSlice.caseReducers.calculateTotals(state);
        },
        calculateTotals: (state) => {
            let amount = 0;
            let total = 0;
            state.cartItems.forEach(item => {
                amount += item.amount;
                total += item.amount * item.price;
            });
            state.amount = amount;
            state.total = total;
        },
        addToCart: (state, action) => {
            const cartItem = state.cartItems.find(item => item.id === action.payload.id);
            if(!cartItem){
                state.cartItems.push(action.payload);
            }else{
                cartItem.amount += action.payload.amount;
            }
            cartSlice.caseReducers.calculateTotals(state);
```

Mart

ORIGINALITY REPORT

9%	7%	1%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	4%
2	Submitted to Daffodil International University Student Paper	1%
3	123dok.com Internet Source	1%
4	Submitted to Misr International University Student Paper	<1%
5	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
6	fastercapital.com Internet Source	<1%
7	Submitted to City of Bath College, Avon Student Paper	<1%
8	Submitted to Higher Education Commission Pakistan Student Paper	<1%
9	Submitted to NCC Education	