



# **AgroMart 360 Digital Marketplace for Agricultural Products**

**Submitted By**

MD. Sultanul Alam

211-35-3172

**Supervised By**

Dr. Md. Fazla Elahe

Assistant Professor & Associate Head

Department of Software Engineering

Daffodil International University

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

## APPROVAL

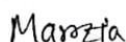
This project titled on “AgroMart 360 Digital Marketplace for Agricultural Products”, submitted by Md.Sultanul Alam (ID: 211-35-3172) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

### BOARD OF EXAMINERS



Chairman

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



Internal Examiner 1

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



Internal Examiner 2

**Dr. Shabnom Mustary**  
Assistant Professor  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University



Internal Examiner 3

**Md. Rajib Mia**  
Lecturer (Senior Scale)  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University



External Examiner

**Mohammad Abul Kashem, PhD**  
Professor  
Department of Computer Science and Engineering  
DUET, Bangladesh

## DAFFODIL INTERNATIONAL UNIVERSITY

### DECLARATION OF PROJECT AND COPYRIGHT

Author's Full Name : MD. Sultanul Alam  
Date of Birth : 08-06-2002  
Title : AgroMart 360 Digital Marketplace for Agricultural Products.  
Academic Session : 2021-2025

I declare that this thesis is classified as:

- CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)\*  
 RESTRICTED (Contains restricted information as specified by the organization where research was done)\*  
 OPEN ACCESS I agree that my project to be published as online open access (Full Text)

I acknowledge that Daffodil International University reserves the following rights:

1. The Project is the Property of Daffodil International University.
2. The Library of Daffodil International University has the right to make copies of the Project for the purpose of research only.
3. The Library of Daffodil International University has the right to make copies of the Project for academic exchange.

Certified by:

Sultanul Alam

211-35-3172

Date: 28-12-2025

Fazla Elahe

Dr. Md. Fazla Elahe

Date: 28-12-2025

## SUPERVISOR DECLARATION

I hereby declare that I have check this project and in my opinion, this project is adequate in terms of quality for the award of the degree of Bachelor of Science.

Fazla Elahe

Full Name : Dr. Md. Fazla Elahe

Position : Assistant Professor & Associate Head Department of Software Engineering.

Date : 28 December 2025

## STUDENT DECLARATION

I hereby declare that the work in this project 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.

*Sultanul Alam*

---

Full Name : MD. Sultanul Alam

ID Number : 211-35-3172

Date : 28 December 2025

# ACKNOWLEDGEMENTS

First of all, we are grateful to The Almighty Allah for making us eligible to complete this project documentation.

Then we would like to thank you our supervisor Dr. Md. Fazla Elahe, Assistant Professor & Associate Head, Department of Software Engineering his excellent direction, tremendous help, and encouragement have been invaluable in the realization of this goal. I owe him a huge debt of my gratitude for providing me with ideas and his belief in my capacity.

I would also like to thank them who were involved in the survey for this project.

We take this opportunity to record our sincere thanks to all the faculty members of the Department of Software Engineering for their help and encouragement.

Last but not least, we would like to thank to our parents, for their unconditional support, love and without this we would not have come this far.

## **DEDICATION**

I therefore declare that I have done this project under the oversight of Dr. Md. Fazla Elahe , Assistant Professor & Associate Head , Department of Software Engineering , Daffodil International University . Also declare that neither entire record or any portion of this record has been submitted somewhere else for my degree.

# ABSTRACT

AgroMart 360 is a comprehensive digital agricultural marketplace developed to address the critical challenges faced by farmers in Bangladesh when accessing quality farming inputs and financial support. Traditional agricultural supply chains in rural areas often suffer from limited availability, price manipulation, counterfeit products, and lack of transparency. AgroMart 360 aims to overcome these issues by providing a reliable, technology-driven platform where farmers can easily purchase genuine agricultural inputs such as certified seeds, fertilizers, pesticides, farming tools, and modern agricultural equipment at fair and competitive prices.

Beyond e-commerce functionality, AgroMart 360 introduces an innovative Mini Loan Feature to support farmers during critical farming periods such as planting or harvesting seasons. Many small and marginal farmers face short-term financial constraints that prevent them from purchasing essential inputs on time. Through this feature, farmers can apply for small, short-term loans directly within the platform to purchase necessary agricultural supplies. The digital loan process is designed to be simple, transparent, and farmer-friendly, enabling quick approval and access to funds without complex paperwork.

By integrating agricultural e-commerce with digital financial services, AgroMart 360 creates a holistic agricultural ecosystem that not only simplifies purchasing but also enhances financial inclusion for farmers. The platform aims to reduce input costs, minimize dependency on local middlemen, improve decision-making through access to reliable information, and ultimately increase agricultural productivity and income stability. Through its technology-driven approach, AgroMart 360 contributes to sustainable agricultural development and supports the long-term growth of Bangladesh's farming community.

## TABLE OF CONTENT

TITLE PAGE	i
APPROVAL	ii
DECLARATION OF COPYRIGHT	iii
SUPERVISOR DECLARATION	iv
STUDENT DECLARATION	v
ACKNOWLEDGEMENTS	vi
DEDICATION	vii
ABSTRACT	viii
TABLE OF CONTENT	ix
LIST OF TABLES	xii
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xiv
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Background	1
1.1.1 Context and Relevance	1
1.1.2 Problem Identification	1
1.1.3 Purpose and Justification	2
1.1.4 Scope	2
1.2 Project Planning and Initiation	3
Feasibility Study (Step-by-Step)	3
1.3 Target User Profile and Tentative Elicitation Process	4
1.3.1 Target User	4
1.3.2 User profile	5
1.3.3 Elicitation Process	6

1.4	Project Block Diagram	7
1.5	System Requirements	7
1.5.1	Hardware Requirements	7
1.5.2	Software Requirements	7
1.5.3	Constraints and Dependencies	7
1.6	Project Scheduling	8
1.7	Summary	8
<b>CHAPTER 2 DESIGN AND IMPLEMENTATION</b>		<b>9</b>
2.1	Introduction	9
2.2	Functional Requirements	9
2.3	Non-Functional Requirements	10
2.3.1	Performance	10
2.3.2	Reliability	11
2.3.3	Portability	11
2.3.4	Security	11
2.3.5	Scalability	11
2.3.6	Usability	111
2.4	Object-oriented System design using UML	11
2.4.1	Use Case Diagram	12
2.4.2	Case Description	12
2.4.3	Activity Diagram	21
2.4.4	Sequence Diagram	27
2.4.4	ER Diagram	31
2.5	Summary	31

<b>CHAPTER 3 SOFTWARE TESTING</b>	32
3.1 Introduction	32
3.2 Testing Features	32
3.2.1 Feature to Be Tested	32
3.3 Testing Strategies	32
3.3.1 Test Approach	32
3.3.2 Pass/Fail Criteria	32
3.4 System Testing (Test Cases with Report)	33
3.5 Summary	35
<b>CHAPTER 4 DEPLOYMENT AND MAINTENANCE</b>	36
4.1 Introduction	36
4.2 Try to follow the SRLC (software release life cycle)	36
<b>CHAPTER 5 USER MANUAL</b>	37
5.1 Introduction	37
5.2 Project Functionalities	37
5.3 Summary	43
<b>CHAPTER 6 PROJECT SUMMARY</b>	44
6.1 Introduction	44
6.2 Project Limitation	44
6.3 Scope	44
6.4 Future Work	44
6.5 Conclusion	45
REFERENCES	46

## LIST OF TABLES

Table 1.3.1: User Profile for farmer	4
Table 1.3.2: User Profile for Admin	5
Table 2.4.1: Case Description-01 User Registration	12
Table 2.4.2: Case Description-02: User Login	13
Table 2.4.3: Case Description 03: Product Cart Management	14
Table 2.4.4: Case Description-04: Checkout	15
Table 2.4.5: Case Description-05: Take Loan	16
Table 2.4.6: Case Description-06: Admin Portal Login	18
Table 2.4.7: Case Description-07: Add, Update, and Delete Products	18
Table 2.4.8: Case Description-08: Order Management	19
Table 3.4.1: Test Case Report for User Registration	33
Table 3.4.2: Test Case Report for User Login	33
Table 3.4.3: Test Case Report for User Add to Cart	34
Table 3.4.4: Test Case Report for User Checkout	35

## LIST OF FIGURES

Figure 1.4.1: System Block Diagram	7
Figure 1.6.1: Project Scheduling Gantt Chart	8
Figure 2.4.1: Use case Diagram	12
Figure 2.4.2: Activity Diagram for User Registration	21
Figure 2.4.3: Activity Diagram for User Login	22
Figure 2.4.4: Activity Diagram for Checkout	23
Figure 2.4.5: Activity Diagram for Add, Update & Delete Products	24
Figure 2.4.6: Activity Diagram for Manage Orders	25
Figure 2.4.7: Activity Diagram for Take Loan	26
Figure 2.4.8: Activity Diagram for Logout	26
Figure 2.4.9: Sequence Diagram for User Registration	27
Figure 2.4.10: Sequence Diagram for User Login	27
Figure 2.4.11: Sequence Diagram for User Checkout	28
Figure 2.4.12: Sequence Diagram for Add to Cart	28
Figure 2.4.13: Sequence Diagram Add, Update & Delete	29
Figure 2.4.14: Sequence Diagram for Take Loan	30
Figure 2.4.15: Entity Relationship Diagram	31
Figure 4.2.1: Software Release Life Cycle Diagram	36
Figure 5.2.1: User Registration	37
Figure 5.2.2: User Login	38
Figure 5.2.3: User Dashboard	38
Figure 5.2.4: Products List	39
Figure 5.2.5: Products Cart List	39
Figure 5.2.6: Payment History	40
Figure 5.2.7: Loan Dashboard	40
Figure 5.2.8: Admin Dashboard	41
Figure 5.2.9: Admin Add Products	41
Figure 5.2.10: Admin Update & Delete products	42
Figure 5.2.11: Admin See Loan Details	42

## **LIST OF ABBREVIATIONS**

MERN	MongoDB, Express.js, React.js, Node.js
ER	Entity Relationship
SRLC	Software Release Life Cycle

# CHAPTER 1

## INTRODUCTION

### 1.1 Background

Agriculture is one of the key sectors in Bangladesh, but farmers continue to have problems accessing high quality agricultural inputs such as seed, fertiliser and other agrochemicals, tools and equipment at reasonable prices throughout the country. With the penetration of smartphones and digital services in rural Bangladesh, there is a good potential to provide a reliable and efficient digital platform which will enable easy purchase of agricultural products. The proposed AgroMart 360 Digital Marketplace aims to ensure that the farmers of Bangladesh can purchase genuine agricultural inputs directly from trusted sellers through an easy-to-use online marketplace. It will bring in transparency about pricing, product availability and also which product has the best guaranteed delivery options to rural areas, thus saving time for farmers and keeping them away from fake or sub-standard products. Through the use of technology to better connect farmers with vital farming goods, the project aims to promote improved agricultural productivity and support further development of Bangladesh's farming sector.

#### 1.1.1 Context and Relevance

Agriculture continues to be the backbone of Bangladesh's economy, employing a great number of its people and making an enormous contribution to national food security. But the lack of quality agricultural inputs, including seeds, fertilisers, pesticides as well as farm machinery and tools remain a huge challenge for farmers – particularly in rural hinterland. Traditionally, the input supply end of the chain is infested with intermediaries leading to high prices during peak times, paucity during cropping seasons and offering spurious or low quality products. Meanwhile, Bangladesh is undergoing a swift digital revolution with smartphone penetration rapidly rising, low cost internet connectivity now widely available and the government's Smart Agriculture – among other digital service initiatives – flourishing. These developments have also opened the door to potential transformations in digital platforms for agricultural supply. The worldwide shift towards e-commerce has not excluded Agri-business and farming as most countries move to online marketplaces to guarantee transparency, vast product range availability, the ability to directly deal with certified sellers. Under this circumstance, a new specific agricultural e-commerce application for purchasing of agricultural products is quite relevant for Bangladesh. It provides the opportunity to provide farmers timely access to inputs, and farmer-centric digital platforms that reduce their dependence on middlemen and enhance productivity and income.

### **1.1.2 Problem Identification**

Although agriculture is a dominant economic sector in Bangladesh, farmers still encounter substantial problems during the purchase of necessary agricultural inputs such as seeds, fertilizers, pesticides, and implements. The whole chain of supply at present is very much chained to local dealers, retailers and agents who tend to regulate both availability and what customers pay. This leads to a number of issues such as high prices, lack of product diversity and fake or substandard quality agricultural products. The lack of genuine and reliable input products, however, is forcing small farmers in rural areas to travel long distances to make hard-to-pay off additional expenses in both time costs and money that can ultimately cut down crop production as well as farming income. Though there have been some initiatives and digital platforms to facilitate online agricultural services in Bangladesh but these are more concentrated towards the selling of crops, advisory services or general information and not concentrated much to develop a complete system dedicated only to purchase of inputs for agriculture. They also have less stringent mechanisms for verifying whether the product is authentic or how to ensure delivery to remote rural areas and typically overlook the farmers' unique requirements, digital skills levels and language preference. Thus, a definite void is still visible in the area of offering user-friendly, authentic and farmer oriented e-commerce based unit where Bangladeshi farmers can buy original agricultural products easily with reasonable price. This project hopes to mitigate some of these shortcomings by developing a trustworthy digital market place that is customized for the purchasing needs of farmers with transparency, authenticity and accessibility in all the regions of Bangladesh.

### **1.1.3 Purpose and Justification**

This project aims to build an effective, robust and handy agricultural eCommerce-based platform which would allow the farmers from Rangpur division of Bangladesh to consistently buy genuine quality agricultural input through online and get delivered at their doorstep efficiently. This project uses digital technology to streamline the process of input acquisition, facilitate fair pricing, and contribute to easy access to a wide variety of important farming items. This platform is also created to tackle some of the old hurdles in traditional supply chain, unavailability of the product, untrustworthiness in quality and influence of middle men who adds up cost and reduces transparency.

The rationale was to develop this project with the hope of having a significant impact on agricultural production and farmer income. In a fast growing digital landscape in Bangladesh, an online marketplace for farmers will make difference and trigger growth. It saves travel time transporting inputs and lowers the risk of buying fake or poor-quality inputs, while giving farmers confidence through vetted sellers and transparent information. It reinforces the national vision of Smart Agriculture by introducing technology in rural areas and helping farmers to choose wisely. The project brings significant value to the agriculture sector while promoting sustainable rural development in Bangladesh through convenience, affordability and its originality.

### **1.1.4 Scope**

This project - farmers only e-commerce 2.1 Project overview This project scope is to provide a dedicated agricultural based e-commerce website where Bangladeshi farmer can directly purchase their necessary inputs, and goods/services from authentic suppliers/manufacturers/industries. The project is concerned with purchase of agricultural produce only and not the sale of crops, availability of advisory services, or general agricultural know-how. Farmers will have access to a wide range of inputs including seeds, fertilizers, pesticides, farm machinery and farming equipment which are listed in an online marketplace that is easy for farmers to use.

The website will support basic product listing, detailed product details and descriptions, price visibility and security order placement and tracking in order to meet specifications for lead-time. One area of focus is on creating an easy-to-use interface for anti-nutrient detection, which can then be used by the Bangladeshi farmer whose technology literacy is not quite up to speed and achieve usability even with all instructions being given in Bangla. Seller identity and product quality seriously will also be checked to ensure no scrappy or materials of some poor quality. The platform is not going to provide smart agricultural analytics, crop advisory systems or financial tools other than facilitating the payment for his products. As a whole the project seeks to designing reliable, user-friendly and efficient farmers-to-the-buyer e-commerce platform, which will elevate buying experience for Bangladeshi farmer and at the same time improve nation's agricultural productivity. And the best part is online loan process.

## **1.2 Project Planning and Initiation**

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

#### **Phase 1 Project Scope Definition:**

This phase deals with the main purpose, necessity and limitations of the agricultural e-commerce website intended to be developed for Bangladeshi farmer community. The preliminary assessment pinpoints the problem: farmers are unable to access quality agricultural inputs “ seeds, fertilisers, pesticides and tools “ because they cannot find it in their vicinity; they can only buy from money lenders; or they get fakes. The project scope is to create a MEAN-stack website for farmers to look and shop their genuine AGRO products from listed seller. The system will be user friendly, support Bangla language, payment security and product delivery track. It will not include crop selling, advisory services or advanced data analytics. Articulating what the product will do and not doing, this phase helps support a disciplined approach to development focused on farmer needs.

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

This stage assess what demand and market potential there is for the agricultural e-commerce website. Study shows that the farmers of Bangladesh are more and more encounter problems to have access to good quality agro-inputs at reasonable prices. Current digital platforms primarily cater to crop selling or advisory services and less

so as input buying solutions. Considering the increasing penetration of the Internet, smartphone and government support for Smart Agriculture, there is strong market potential for a dedicated online input-purchasing platform. Investors have just spot-checked a few of the assumptions behind this industry, such as farmers' openness to digital solutions if the platform is easy-to-use, cost-effective and reliable. Competitive market analysis discloses very few native e-commerce applications focused solely on purchasing of input into Agriculture, thereby resulting in a large untapped segment. Accordingly, high feasibility is applicable to the market of this project and it can be adopted positively.

### **Phase 3 Technical Feasibility Analysis:**

Phase 1 determines if the technology and product can be successfully developed. The MEAN stack MongoDB, Express. js, Angular, and Node. js, which makes it a great choice for developing expensive and large scale e-commerce systems. WordPress is a good name, but since it's a standalone page (with or without advanced WP functionality), MySQL won't indexed it as well as a Node and MongoDB can. js and Express. js enable high-speed backend operations. Interactive and responsive UI is one of the best features Angular that fits for farmers using mobile or low spec devices. Use of cloud servers and deployment technologies are available in the market. It also needs basic web development skills that most people have access to. Given the above, this is a technologically feasible project that can be developed on time using current technology and methods.

### **Phase 4 Financial Feasibility Analysis:**

This step considers if the project is financially feasible and robust. As the MEAN stack is open source, and hence development costs are relatively low initially, mostly including domain registration, web hosting charges, developer salaries and security integration if possible. The running costs but not too much maintenance, customer and marketing support. The platform could monetize things such as seller's fees, promoted product listings, partnerships with agriculture suppliers, and service charges. For users, the outcome is higher profits for our farmers as they have first access to quality inputs at affordable prices and reduced time spent on travel and middlemen. Socio-economic sustainability Digital agricultural solutions are in greater demand, which enhances long-term financial viability of the business in Bangladesh. In summary, the project is economically viable with promising potential for expansion and financial returns.

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

### **1.3.1 Target User**

1. Farmers
2. Administrator

### 1.3.2 User profile

Table 1.3.1: User Profile for farmer

User Class	Note on Characteristics
Type of user	Buyers of agricultural products.
Age range	Typically 18-55 years, includes young.
Frequency of use	Mainly during crop seasons
Mandatory	Not mandatory but highly beneficial for easier and reliable purchasing
Computer experience	Low to moderate. Many farmers may have basic smartphone or internet skill.
Education	Varies widely some have basic literacy.
goal	To purchase genuine seeds, medicine and equipment easily.
Language skills	Mostly Bangla speakers and limited English.
Number of users	Potentially thousands across rural and semi-urban areas of Bangladesh.
Training	Minimal training required, simple and intuitive interface preferred.
Others system use	May use basic website, mobile banking
Way of working	Practical, task focused users

Table 1.3.2: User Profile for Admin

User Class	Note on Characteristics
Type of user	Manages the website
Age range	25-50 years
Frequency of use	Continuous monitoring and management
Mandatory	Yes, essential for the operation and security of the the website
Computer experience	Moderate to high
Education	At least Bachelor's degree in Software Engineer or related field
goal	Ensure smooth website operation, manage products, maintain security.

Language skills	Proficient in Bangla and English
Number of users	1-3 administrators depending on scale
Training	Moderate training on the system backend, security protocols and seller verification procedures
Others system use	Familiar with other e-commerce or content management system
Way of working	Task oriented and systematic, monitors transaction, resolves issues.

### 1.3.3 Elicitation Process

#### 1. Interviews

Farmers, suppliers and administrators will be interviewed through structured and semi-structured interviews. Farmers will also be questioned on their experiences in purchasing agricultural inputs, how and where they prefer to buy them, and their level of digital literacy. Suppliers will share information about product, inventory and transaction management. Administrators will communicate system control, security and workflow needs. Interviews provide the opportunity for in-depth, qualitative understanding of user desires.

#### 2. Surveys/Questionnaires

More (other) farmers from rural and semiurban zone will be surveyed to measure the preferences, purchase frequency of input, and willingness to use an internet platform. Surveys produce quantifiable feedback that can reveal commonalities and trends like the most popular product categories, their delivery preferences or the need for language accessibility.

#### 3. Focus Groups

Farmers and suppliers as well as delivery partners will participate in focus group discussions. These sessions promote open discussions and reveal work-down concepts, challenges, which are your preferred working methods against the platform. Focus groups are particularly well suited to the evaluation of early design concepts and behaviour.

#### 4. Observation

Watching framers in action – observing at local markets and agri-supply store would be key as that will provide insights into the current buying process, pain points and purchase patterns. There is a lot of insight to be gained from observational data in order to create an intuitive website flow.

## 5. Document Analysis

Published papers, market reportages, government agricultural census and studies on ICT adoption in Bangladesh would be reviewed. This gives a background on market situation, legal regulations and successful examples of agricultural e-commerce.

## 6. Prototyping and Feedback

Selected farmers, suppliers and administrators will be presented with early test prototypes or wireframes for feedback on ease of use, navigation and functionality. Feedback Needs are Met Early and Won't Lead to Future Redesign Activities Iterative feedback will enable the platform to satisfy user needs, resulting in lower cost of subsequent redesign.

### 1.4 Project Block Diagram

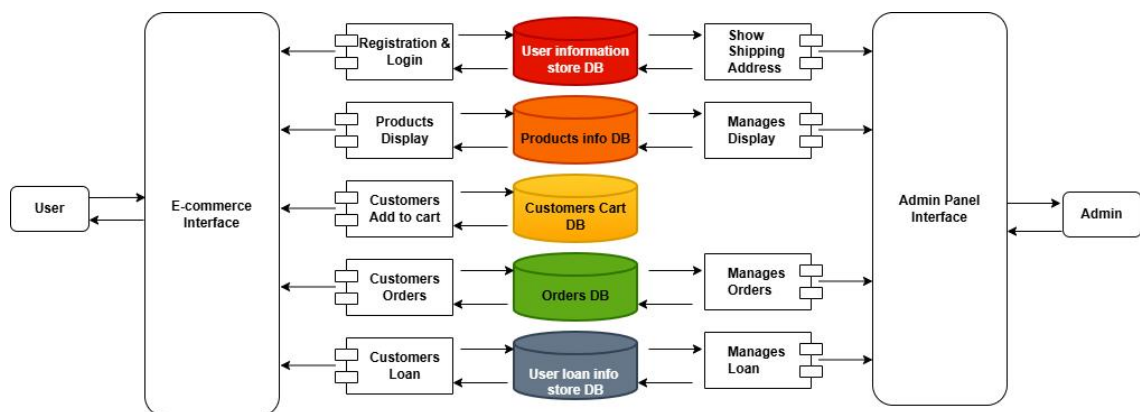


Figure 1.4.1: System Block Diagram

## 1.5 System Requirements

### 1.5.1 Hardware Requirements

I a computer with Intel Core i5 CPU, 8 GB of memory and 256 GB SSD.

II. For server hosting: 2–4 vCPU, 4–8 GB RAM, 50 GB SSD.

III. Overseas candidates should have reliable internet connection for online booking and database access.

### 1.5.2 Software Requirements

- Operating System: Windows / Linux / macOS
- MERN Stack: MongoDB, Express.js, React.js, Node.js

c) Tools: VS Code, Chrome browser

### 1.5.3 Constraints and Dependencies

For its functioning, the site relies on a stable internet connection, stable hosting and third-party payment gateways Stripe as well as delivery partners. It was a combination of budget constraints, hardware capability and limited development time.

## 1.6 Project Scheduling

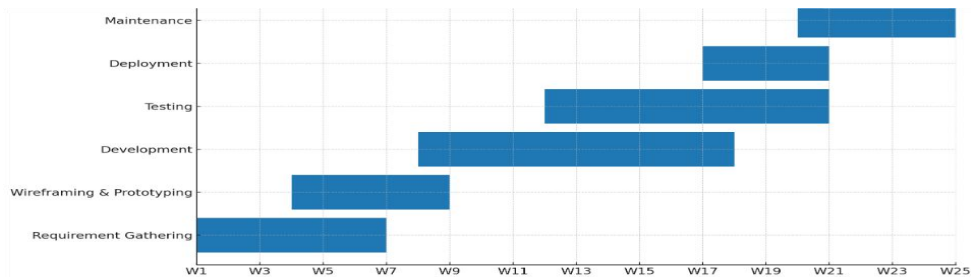


Figure 1.6.1: Project Scheduling Gantt Chart.

## 1.7 Summary

This chapter described the planning and commencement of the AEW project. It described the project background, objectives, and scope, highlighting a need of facilitating Bangladeshi farmer to access true HDU conveniently. The proof-of-concept showed that the MEAN stack can be technical, economical, operational and financial viable. Users of the platform, such as farmers, suppliers, administrators, investors and delivery partners were researched by their profiles, needs and usage behaviors. The process of Elicitation described various methods to effectively collect user requirements such as interviews, surveys, focus groups and so on /observation/prototyping. Lastly, the hardware and software specifications as well as limitations and dependencies were introduced to guarantee success in both software development and implementation. In summary, this chapter provides a good starting point to develop the robust and easy-to-use e-commerce system for social commerce with scalability in Bangladesh agricultural domain.

# CHAPTER 2

## DESIGN AND IMPLEMENTATION

### 2.1 Introduction

In the following section, we are only going to discuss about how we designed our website. It details how the system is to be built in order that it satisfies user needs and accomplishes project goals. In this chapter, we present the general system architecture, database model, user interface (UI) design, and functionality of the accompanying web-based platform with a focus on offering an easy-to-use and secure online market place allowing farmers to buy agricultural inputs.

### 2.2 Functional Requirements

<b>FR01</b>	<b>Registration</b>
<b>Description</b>	Allows users to create an account on this platform.
<b>Stakeholder</b>	Customers, Admin

<b>FR02</b>	<b>Login</b>
<b>Description</b>	Customers and Admin must login before using this platform
<b>Stakeholder</b>	Customers, Admin

<b>FR03</b>	<b>Add, Update, and Delete Products</b>
<b>Description</b>	Add new products, update existing product details, or remove products from the platform.
<b>Stakeholder</b>	Admin

<b>FR04</b>	<b>Product Categorization</b>
<b>Description</b>	Organize products into categories
<b>Stakeholder</b>	Admin

<b>FR05</b>	<b>Order Management</b>
<b>Description</b>	Allows users to place orders, view order status and manage order history.
<b>Stakeholder</b>	Admin

<b>FR06</b>	<b>Add to Cart</b>
<b>Description</b>	Farmers to add products to the cart, update quantities, and remove items before checkout.
<b>Stakeholder</b>	Customers

<b>FR07</b>	<b>Checkout</b>
<b>Description</b>	Customer can place an order by providing shipping details and selecting a payment method.
<b>Stakeholder</b>	Customers

<b>FR08</b>	<b>Take Loan</b>
<b>Description</b>	Allows farmers to apply for loans, submit necessary documents, and track loan approval status.
<b>Stakeholder</b>	Customers

<b>FR09</b>	<b>Logout</b>
<b>Description</b>	Allows users to securely log out of their account and end their active session.
<b>Stakeholder</b>	Customers, Admin

## 2.3 Non-Functional Requirements

These are the quality requirements, which describe how well the Agricultural E-Commerce Website operates. Such requirements are there to make sure that your application/software product is working properly, under a safe environment and its user experience was not being compromised at any moment. Instead, they concentrate on the way in which the system works.

### **2.3.1 Performance**

It should be a fast and responsive system to use. Every single page on the web must perform acceptably, even when it pins CPU to 100% at peak use times. It also needs to be able to deal with several simultaneous users, process orders instantly and perform at peak ability when busy.

### **2.3.2 Reliability**

It has to run smoothly without crashes every few minutes. It needs to stay up under all circumstances with little downtime. There have to be tools to stop data from being lost and keep the system stable so users can rely on it 24×7.

### **2.3.3 Portability**

The website must be available in multiple platforms and devices. It needs to work seamlessly on desktops, notepads, tablets as well as cell phones. It should also work great across all major browsers such as Chrome, Firefox, Edge and Safari etc.

### **2.3.4 Security**

Protection of sensitive user data, as well as financial transactions inevitably implies secure means for authentication and data protection. Sensitive features should be limited to specific roles, and the system should use secure login, encryption, protection policies and security controls to prevent unauthorized access.

### **2.3.5 Scalability**

Not necessarily without limit, but the system is tied to the growth that customer needs demand. It should be able to grow with more user, product amount and orders, or new module. The design should also be such that it is readily adaptable for improvement in the future without major structural mods.

### **2.3.6 Usability**

The UI must be clean, user-friendly and friendly to even the most proficient farmer. The navigation should be intuitive, instructions easy on the eye and flow of interaction in general must be simple and user-friendly.

## 2.4 Object-oriented System design using UML

### 2.4.1 Use Case Diagram

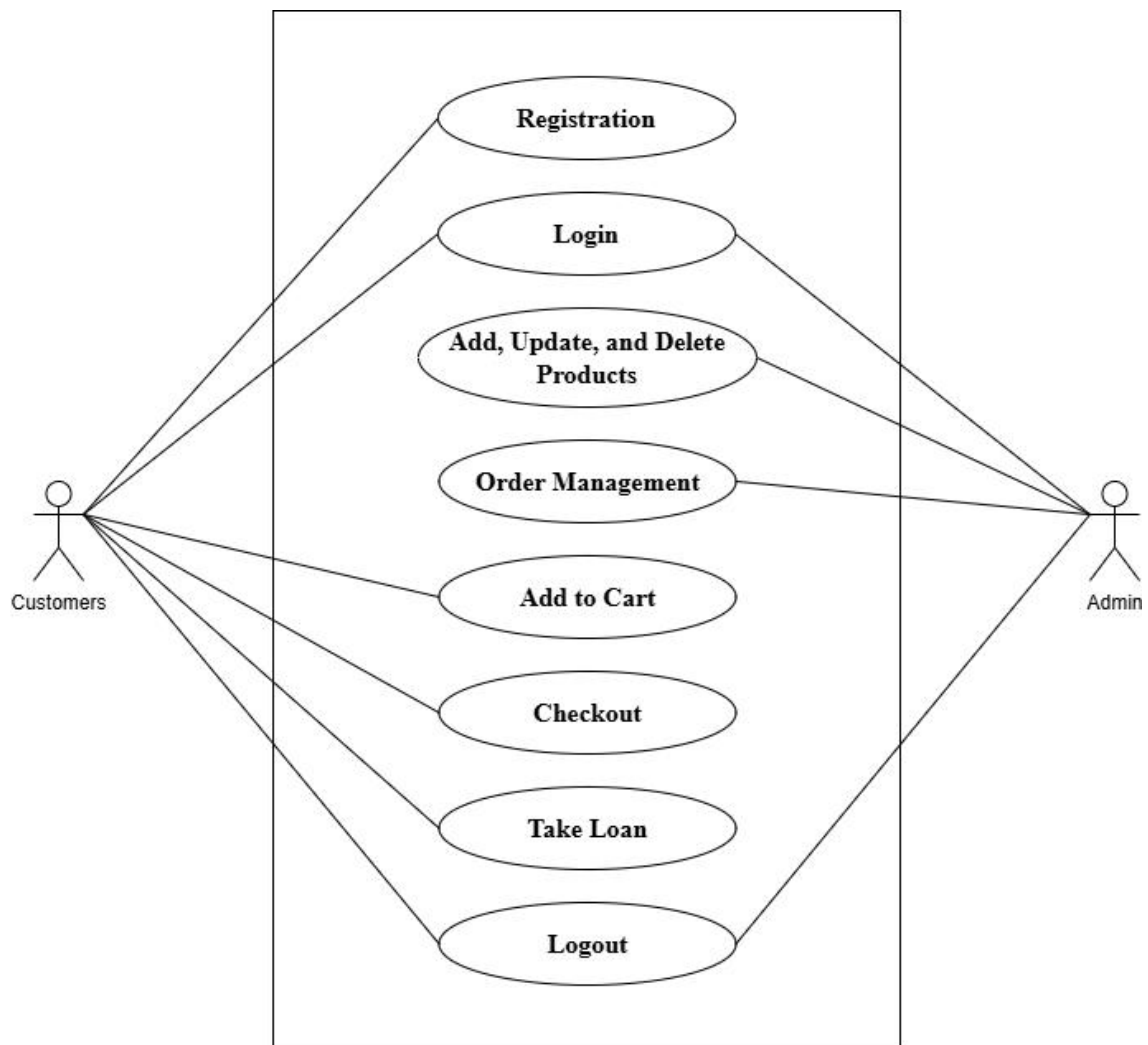


Figure 2.4.1: Use case Diagram

### 2.4.2 Case Description

Table 2.4.1: Case Description-01 User Registration

Use Case	User Registration
Goal	Users can registration to sign in to this system.
Precondition	Users must access the registration page.
Success End Condition	Notification: "Successfully Registered"

Failed End Condition	Notification: “Registered Failed”																
Primary Actors:	Customer																
Secondary Actors:	System																
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 first</td> </tr> <tr> <td>2.</td> <td>Than provide registration form</td> </tr> <tr> <td>3.</td> <td>Enter Information all</td> </tr> <tr> <td>4.</td> <td>Press “Submit”</td> </tr> <tr> <td>5.</td> <td>Information saved</td> </tr> <tr> <td>6.</td> <td>The system saves details and shows “Successfully Registered”</td> </tr> </table>	1.	Press “Registration” Button first	2.	Than provide registration form	3.	Enter Information all	4.	Press “Submit”	5.	Information saved	6.	The system saves details and shows “Successfully Registered”				
1.	Press “Registration” Button first																
2.	Than provide registration form																
3.	Enter Information all																
4.	Press “Submit”																
5.	Information saved																
6.	The system saves details and shows “Successfully Registered”																
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>System Error</td> </tr> <tr> <td></td> <td>1.1.a. “System Error please try again later”</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 &amp; notify by “please fill in all required fields”</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>	1.1	System Error		1.1.a. “System Error please try again later”	4.1	The user Did not fill up the details		4.1.a. Checked & notify by “please fill in all required fields”	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”
1.1	System Error																
	1.1.a. “System Error please try again later”																
4.1	The user Did not fill up the details																
	4.1.a. Checked & notify by “please fill in all required fields”																
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 2 minutes																

Table2.4.2: Case Description-02: User Login

Use Case	User Login
Goal	Users can log in to access the system.
Precondition	User must already be registered.
Success End Condition	Notification: “Successfully Login”
Failed End Condition	Notification: “Invalid credentials”

Primary Actors:	Customer												
Secondary Actors:	System												
Trigger	User provides login credentials.												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>User presses the "Login" button.</td> </tr> <tr> <td>2.</td> <td>User enters their email.</td> </tr> <tr> <td>3.</td> <td>User enters their password.</td> </tr> <tr> <td>4.</td> <td>The system validates the credentials.</td> </tr> <tr> <td>5.</td> <td>If credentials are correct, grant access to the user.</td> </tr> <tr> <td>6.</td> <td>Notify the user: "Login Successful!"</td> </tr> </table>	1.	User presses the "Login" button.	2.	User enters their email.	3.	User enters their password.	4.	The system validates the credentials.	5.	If credentials are correct, grant access to the user.	6.	Notify the user: "Login Successful!"
1.	User presses the "Login" button.												
2.	User enters their email.												
3.	User enters their password.												
4.	The system validates the credentials.												
5.	If credentials are correct, grant access to the user.												
6.	Notify the user: "Login Successful!"												
Alternative Flows	<table border="1"> <tr> <td>2.1</td> <td>Invalid Email</td> </tr> <tr> <td></td> <td>2.1.a. Email is not registered. 2.1.b. Notify: "Invalid Credentials."</td> </tr> <tr> <td>3.1</td> <td>Invalid Password</td> </tr> <tr> <td></td> <td>3.1.a. Password does not match the email. 3.1.b. Notify: "Invalid Credentials."</td> </tr> <tr> <td>3.2</td> <td>System Error</td> </tr> <tr> <td></td> <td>3.2.a. System cannot process the login due to technical issues. 3.2.b. Notify: "System Error! Please try again."</td> </tr> </table>	2.1	Invalid Email		2.1.a. Email is not registered. 2.1.b. Notify: "Invalid Credentials."	3.1	Invalid Password		3.1.a. Password does not match the email. 3.1.b. Notify: "Invalid Credentials."	3.2	System Error		3.2.a. System cannot process the login due to technical issues. 3.2.b. Notify: "System Error! Please try again."
2.1	Invalid Email												
	2.1.a. Email is not registered. 2.1.b. Notify: "Invalid Credentials."												
3.1	Invalid Password												
	3.1.a. Password does not match the email. 3.1.b. Notify: "Invalid Credentials."												
3.2	System Error												
	3.2.a. System cannot process the login due to technical issues. 3.2.b. Notify: "System Error! Please try again."												
Quality Requirements	Login process must complete within 5 seconds.												

Table2.4.3: Case Description 03: Product Cart Management

Use Case	Cart Management
Goal	Customers can add desired products to their Wishlist or shopping cart.
Precondition	The customer is logged into the system.
Success End Condition	Products are successfully added to the Wishlist or shopping cart.

Failed End Condition	Notification: "Failed to Add Product to Cart."												
Primary Actors:	Customer												
Secondary Actors:	System												
Trigger	The customer selects the "Add to Cart" button.												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>The customer selects a product.</td> </tr> <tr> <td>2.</td> <td>The system displays the "Add to Cart" option.</td> </tr> <tr> <td>3.</td> <td>The customer clicks the chosen option.</td> </tr> <tr> <td>4.</td> <td>The system verifies product availability.</td> </tr> <tr> <td>5.</td> <td>The system adds the product to the Wishlist or shopping cart.</td> </tr> <tr> <td>6.</td> <td>The system notifies the user: "Product Successfully Added to Cart!"</td> </tr> </table>	1.	The customer selects a product.	2.	The system displays the "Add to Cart" option.	3.	The customer clicks the chosen option.	4.	The system verifies product availability.	5.	The system adds the product to the Wishlist or shopping cart.	6.	The system notifies the user: "Product Successfully Added to Cart!"
1.	The customer selects a product.												
2.	The system displays the "Add to Cart" option.												
3.	The customer clicks the chosen option.												
4.	The system verifies product availability.												
5.	The system adds the product to the Wishlist or shopping cart.												
6.	The system notifies the user: "Product Successfully Added to Cart!"												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>System Error</td> </tr> <tr> <td></td> <td>1.1.a. The system encounters a technical issue. 1.2.b. Notify the user: "System Error! Please try again."</td> </tr> <tr> <td>4.1</td> <td>No Matching Product Found</td> </tr> <tr> <td></td> <td>4.1.a. The product is not available. 4.1.b. Notify the user: "This Product is Currently Out of Stock."</td> </tr> <tr> <td>5.1</td> <td>Request Timeout</td> </tr> <tr> <td></td> <td>5.1.a. The system fails to save the product to the cart. 5.1.b. Notify the user: "Failed to Add Product! Please Try Again."</td> </tr> </table>	1.1	System Error		1.1.a. The system encounters a technical issue. 1.2.b. Notify the user: "System Error! Please try again."	4.1	No Matching Product Found		4.1.a. The product is not available. 4.1.b. Notify the user: "This Product is Currently Out of Stock."	5.1	Request Timeout		5.1.a. The system fails to save the product to the cart. 5.1.b. Notify the user: "Failed to Add Product! Please Try Again."
1.1	System Error												
	1.1.a. The system encounters a technical issue. 1.2.b. Notify the user: "System Error! Please try again."												
4.1	No Matching Product Found												
	4.1.a. The product is not available. 4.1.b. Notify the user: "This Product is Currently Out of Stock."												
5.1	Request Timeout												
	5.1.a. The system fails to save the product to the cart. 5.1.b. Notify the user: "Failed to Add Product! Please Try Again."												
Quality Requirements	Products should be added to the Wishlist or shopping cart within 2 seconds.												

Table2.4.4: Case Description-04: Checkout

Use Case	Checkout
Goal	Customers can purchase products.

Precondition	Products must be in the shopping cart.												
Success End Condition	Notification: "Order Placed Successfully!"												
Failed End Condition	Notification: "Order Failed."												
Primary Actors:	Customer												
Secondary Actors:	System, Bank, Payment Gateway												
Trigger	User initiates checkout.												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>User clicks the "Checkout" button.</td> </tr> <tr> <td>2.</td> <td>The system verifies cart details.</td> </tr> <tr> <td>3.</td> <td>User enters the shipping address.</td> </tr> <tr> <td>4.</td> <td>User provides payment information.</td> </tr> <tr> <td>5.</td> <td>The system processes the payment and confirms the order.</td> </tr> <tr> <td>6.</td> <td>The system notifies the user: "Order Placed Successfully!"</td> </tr> </table>	1.	User clicks the "Checkout" button.	2.	The system verifies cart details.	3.	User enters the shipping address.	4.	User provides payment information.	5.	The system processes the payment and confirms the order.	6.	The system notifies the user: "Order Placed Successfully!"
1.	User clicks the "Checkout" button.												
2.	The system verifies cart details.												
3.	User enters the shipping address.												
4.	User provides payment information.												
5.	The system processes the payment and confirms the order.												
6.	The system notifies the user: "Order Placed Successfully!"												
Alternative Flows	<table border="1"> <tr> <td>2.1</td> <td>Empty Cart</td> </tr> <tr> <td></td> <td>2.1.a. Cart has no products. 2.1.b. Notify the user: "Cart is empty. Add items before checkout."</td> </tr> <tr> <td>3.1</td> <td>Missing Shipping Address</td> </tr> <tr> <td></td> <td>3.1.a. User does not provide a shipping address. 3.1.b. Notify the user: "Couldn't find shipping address."</td> </tr> <tr> <td>4.1</td> <td>Payment Failure</td> </tr> <tr> <td></td> <td>4.1.a. Payment cannot be processed. 4.1.b. Notify the user: "Payment Failed! Please retry."</td> </tr> </table>	2.1	Empty Cart		2.1.a. Cart has no products. 2.1.b. Notify the user: "Cart is empty. Add items before checkout."	3.1	Missing Shipping Address		3.1.a. User does not provide a shipping address. 3.1.b. Notify the user: "Couldn't find shipping address."	4.1	Payment Failure		4.1.a. Payment cannot be processed. 4.1.b. Notify the user: "Payment Failed! Please retry."
2.1	Empty Cart												
	2.1.a. Cart has no products. 2.1.b. Notify the user: "Cart is empty. Add items before checkout."												
3.1	Missing Shipping Address												
	3.1.a. User does not provide a shipping address. 3.1.b. Notify the user: "Couldn't find shipping address."												
4.1	Payment Failure												
	4.1.a. Payment cannot be processed. 4.1.b. Notify the user: "Payment Failed! Please retry."												
Quality Requirements	Checkout process must complete within 3 minutes.												

Table2.4.5: Case Description-05: Take Loan

Use Case	Take Loan
----------	-----------

Goal	Customers can apply for a loan to purchase products.												
Precondition	The customer must be logged into the system.												
Success End Condition	Notification: "Loan Application Submitted Successfully!"												
Failed End Condition	Notification: "Loan Application Failed."												
Primary Actors:	Customer												
Secondary Actors:	System,Bank												
Trigger	Customer selects the "Take Loan" option.												
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Customer selects the "Take Loan" option from the system.</td> </tr> <tr> <td>2.</td> <td>The system displays the loan application form.</td> </tr> <tr> <td>3.</td> <td>Customer enters the required details (loan amount, purpose, documents, etc.).</td> </tr> <tr> <td>4.</td> <td>Customer clicks "Submit Application".</td> </tr> <tr> <td>5.</td> <td>The system processes the application and sends it to the bank/loan provider.</td> </tr> <tr> <td>6.</td> <td>The system notifies the customer: "Loan Application Submitted Successfully!"</td> </tr> </table>	1.	Customer selects the "Take Loan" option from the system.	2.	The system displays the loan application form.	3.	Customer enters the required details (loan amount, purpose, documents, etc.).	4.	Customer clicks "Submit Application".	5.	The system processes the application and sends it to the bank/loan provider.	6.	The system notifies the customer: "Loan Application Submitted Successfully!"
1.	Customer selects the "Take Loan" option from the system.												
2.	The system displays the loan application form.												
3.	Customer enters the required details (loan amount, purpose, documents, etc.).												
4.	Customer clicks "Submit Application".												
5.	The system processes the application and sends it to the bank/loan provider.												
6.	The system notifies the customer: "Loan Application Submitted Successfully!"												
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Loan Service Unavailable</td> </tr> <tr> <td></td> <td>1.1.a. Loan functionality is temporarily unavailable. 1.1.b. Notify: "Loan Service Currently Unavailable. Try again later."</td> </tr> <tr> <td>3.1</td> <td>Missing Required Information</td> </tr> <tr> <td></td> <td>3.1.a. Customer leaves required fields empty. 3.1.b. Notify: "Please fill in all required fields."</td> </tr> <tr> <td>4.1</td> <td>System Error</td> </tr> <tr> <td></td> <td>4.1.a. The system encounters an error during loan submission. 4.1.b. Notify: "Loan Application Failed! Please try again."</td> </tr> </table>	1.1	Loan Service Unavailable		1.1.a. Loan functionality is temporarily unavailable. 1.1.b. Notify: "Loan Service Currently Unavailable. Try again later."	3.1	Missing Required Information		3.1.a. Customer leaves required fields empty. 3.1.b. Notify: "Please fill in all required fields."	4.1	System Error		4.1.a. The system encounters an error during loan submission. 4.1.b. Notify: "Loan Application Failed! Please try again."
1.1	Loan Service Unavailable												
	1.1.a. Loan functionality is temporarily unavailable. 1.1.b. Notify: "Loan Service Currently Unavailable. Try again later."												
3.1	Missing Required Information												
	3.1.a. Customer leaves required fields empty. 3.1.b. Notify: "Please fill in all required fields."												
4.1	System Error												
	4.1.a. The system encounters an error during loan submission. 4.1.b. Notify: "Loan Application Failed! Please try again."												
Quality Requirements	Loan application submission must complete within 30 seconds.												

Table2.4.6: Case Description-06: Admin Portal Login

Use Case	Admin Portal Login										
Goal	Admins can log in to manage the system.										
Precondition	Admin must have valid login credentials.										
Success End Condition	Notification: "Login Successful!"										
Failed End Condition	Notification: "Login Failed."										
Primary Actors:	Admin										
Secondary Actors:	None										
Trigger	Admin provides login credentials.										
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Admin enters username and password.</td> </tr> <tr> <td>2.</td> <td>The system verifies credentials.</td> </tr> <tr> <td>3.</td> <td>If credentials are correct, grant access to the admin portal.</td> </tr> <tr> <td>4.</td> <td>The system fetches relevant administrative data (e.g., product details, orders).</td> </tr> <tr> <td>5.</td> <td>Notify the admin: "Login Successful!"</td> </tr> </table>	1.	Admin enters username and password.	2.	The system verifies credentials.	3.	If credentials are correct, grant access to the admin portal.	4.	The system fetches relevant administrative data (e.g., product details, orders).	5.	Notify the admin: "Login Successful!"
1.	Admin enters username and password.										
2.	The system verifies credentials.										
3.	If credentials are correct, grant access to the admin portal.										
4.	The system fetches relevant administrative data (e.g., product details, orders).										
5.	Notify the admin: "Login Successful!"										
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>Invalid Credentials</td> </tr> <tr> <td></td> <td>1.1.a. Username or password is incorrect. 1.1.b. Notify: "Login Failed! Please try again."</td> </tr> </table>	1.1	Invalid Credentials		1.1.a. Username or password is incorrect. 1.1.b. Notify: "Login Failed! Please try again."						
1.1	Invalid Credentials										
	1.1.a. Username or password is incorrect. 1.1.b. Notify: "Login Failed! Please try again."										
Quality Requirements	Admin portal login must complete within 1 minutes.										

Table2.4.7: Case Description-07: Add, Update, and Delete Products

Use Case	Add, Update, and Delete Products
Goal	Admins can manage product details.

Precondition	Admins must be logged in to the portal.										
Success End Condition	Notification: "Product Added Successfully!", "Product Updated Successfully!", "Product Deleted Successfully!"										
Failed End Condition	Notification: "Operation Failed."										
Primary Actors:	Admin										
Secondary Actors:	System										
Trigger	Admin selects a product management option.										
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Admin selects "Add", "Update", or "Delete" option.</td> </tr> <tr> <td>2.</td> <td>The system provides relevant forms or displays product details.</td> </tr> <tr> <td>3.</td> <td>Actor enters new product information or modifies existing details.</td> </tr> <tr> <td>4.</td> <td>Actor clicks "Submit" to confirm changes.</td> </tr> <tr> <td>5.</td> <td>System processes the operation and notifies: "Operation Successful!"</td> </tr> </table>	1.	Admin selects "Add", "Update", or "Delete" option.	2.	The system provides relevant forms or displays product details.	3.	Actor enters new product information or modifies existing details.	4.	Actor clicks "Submit" to confirm changes.	5.	System processes the operation and notifies: "Operation Successful!"
1.	Admin selects "Add", "Update", or "Delete" option.										
2.	The system provides relevant forms or displays product details.										
3.	Actor enters new product information or modifies existing details.										
4.	Actor clicks "Submit" to confirm changes.										
5.	System processes the operation and notifies: "Operation Successful!"										
Alternative Flows	<table border="1"> <tr> <td>1.1</td> <td>System Error</td> </tr> <tr> <td></td> <td>           1.1.a. The system encounters an error during the operation.            1.1.b. Notify the user: "Operation Failed! Please retry."         </td> </tr> </table>	1.1	System Error		1.1.a. The system encounters an error during the operation. 1.1.b. Notify the user: "Operation Failed! Please retry."						
1.1	System Error										
	1.1.a. The system encounters an error during the operation. 1.1.b. Notify the user: "Operation Failed! Please retry."										
Quality Requirements	Product changes must reflect immediately in the system.										

Table2.4.8: Case Description-08: Order Management

Use Case	Order Management
Goal	Admins can manage customer orders.
Precondition	Orders must exist in the system.
Success End Condition	Notification: "Order Updated Successfully!"
Failed End Condition	Notification: "Order Update Failed."

Primary Actors:	Admin										
Secondary Actors:	None										
Trigger	Admin selects an order management option.										
Description / Main Success Scenario	<table border="1"> <tr> <td>1.</td> <td>Admin selects the "Manage Orders" option.</td> </tr> <tr> <td>2.</td> <td>The system displays the list of existing customer orders.</td> </tr> <tr> <td>3.</td> <td>The actor selects an order and updates its status or details.</td> </tr> <tr> <td>4.</td> <td>The actor clicks the "Save Changes" button.</td> </tr> <tr> <td>5.</td> <td>The system processes the update and notifies: "Order Updated Successfully!"</td> </tr> </table>	1.	Admin selects the "Manage Orders" option.	2.	The system displays the list of existing customer orders.	3.	The actor selects an order and updates its status or details.	4.	The actor clicks the "Save Changes" button.	5.	The system processes the update and notifies: "Order Updated Successfully!"
1.	Admin selects the "Manage Orders" option.										
2.	The system displays the list of existing customer orders.										
3.	The actor selects an order and updates its status or details.										
4.	The actor clicks the "Save Changes" button.										
5.	The system processes the update and notifies: "Order Updated Successfully!"										
Alternative Flows	<table border="1"> <tr> <td>2.1</td> <td>No Orders Available</td> </tr> <tr> <td></td> <td>2.1.a. No orders exist in the system. 2.1.b. Notify: "No orders available to manage."</td> </tr> </table>	2.1	No Orders Available		2.1.a. No orders exist in the system. 2.1.b. Notify: "No orders available to manage."						
2.1	No Orders Available										
	2.1.a. No orders exist in the system. 2.1.b. Notify: "No orders available to manage."										
Quality Requirements	The order search or browsing process must complete within 5 seconds.										

### 2.4.3 Activity Diagram

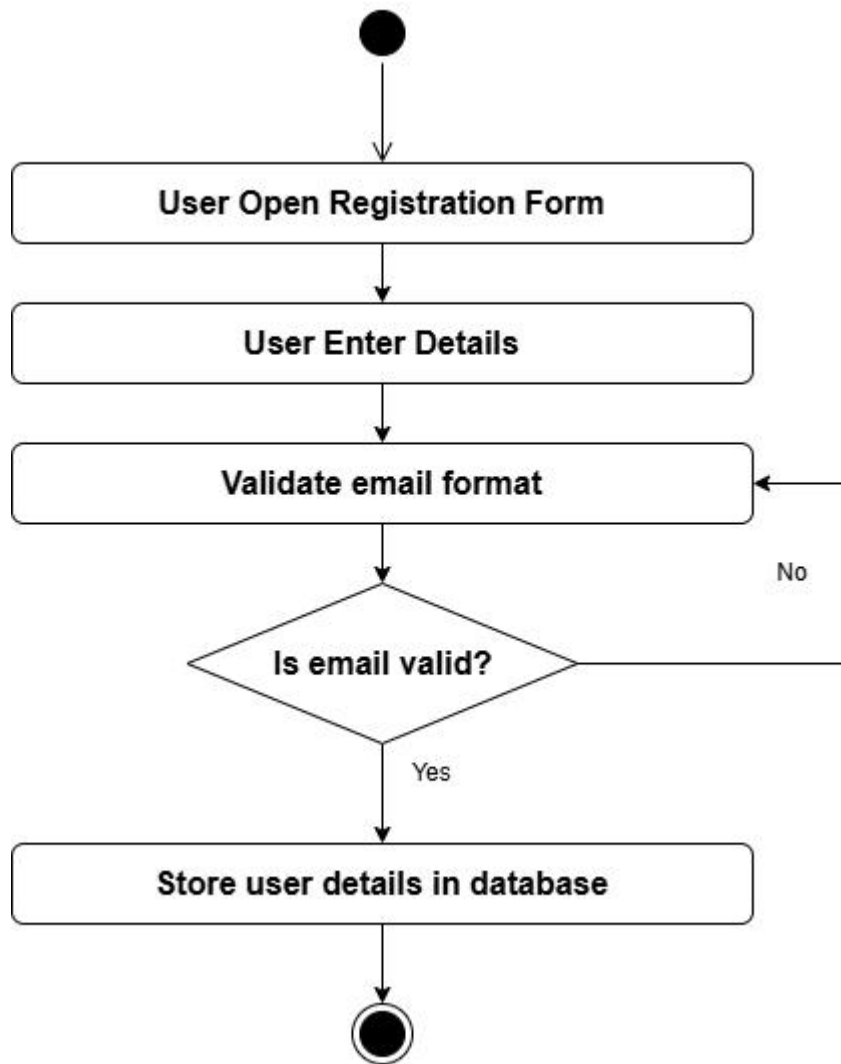


Figure 2.4.2: Activity Diagram for User Registration

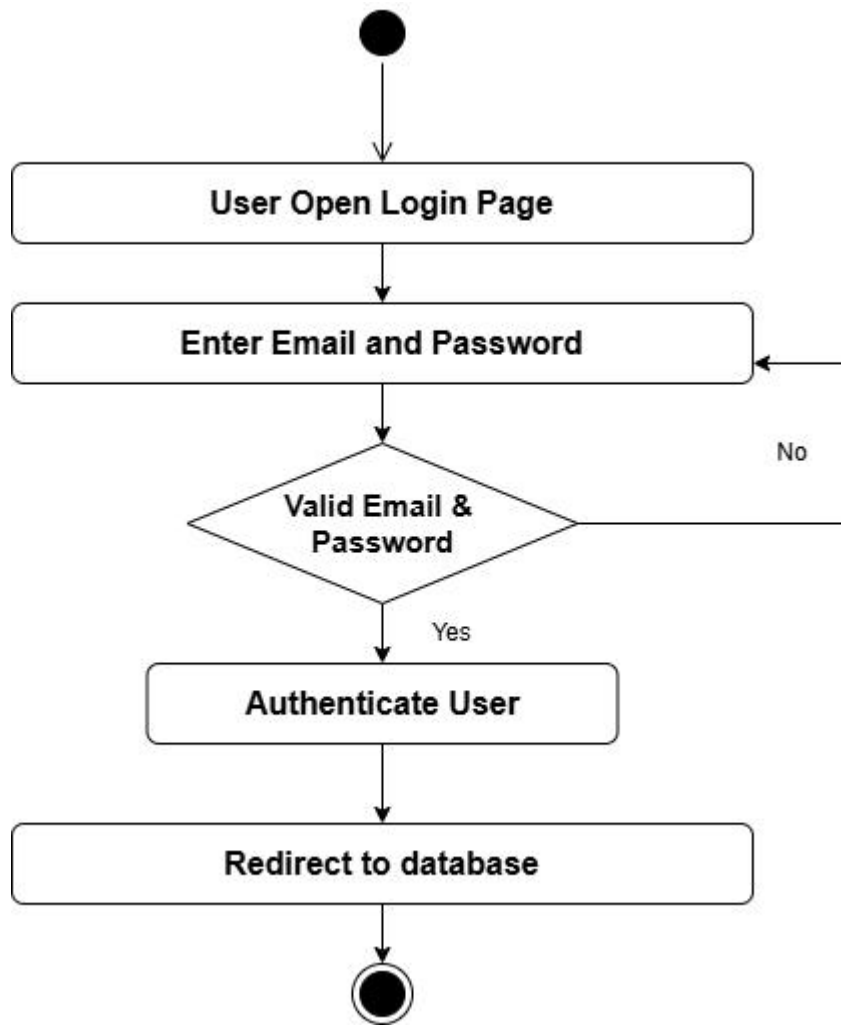


Figure 2.4.3: Activity Diagram for User Login

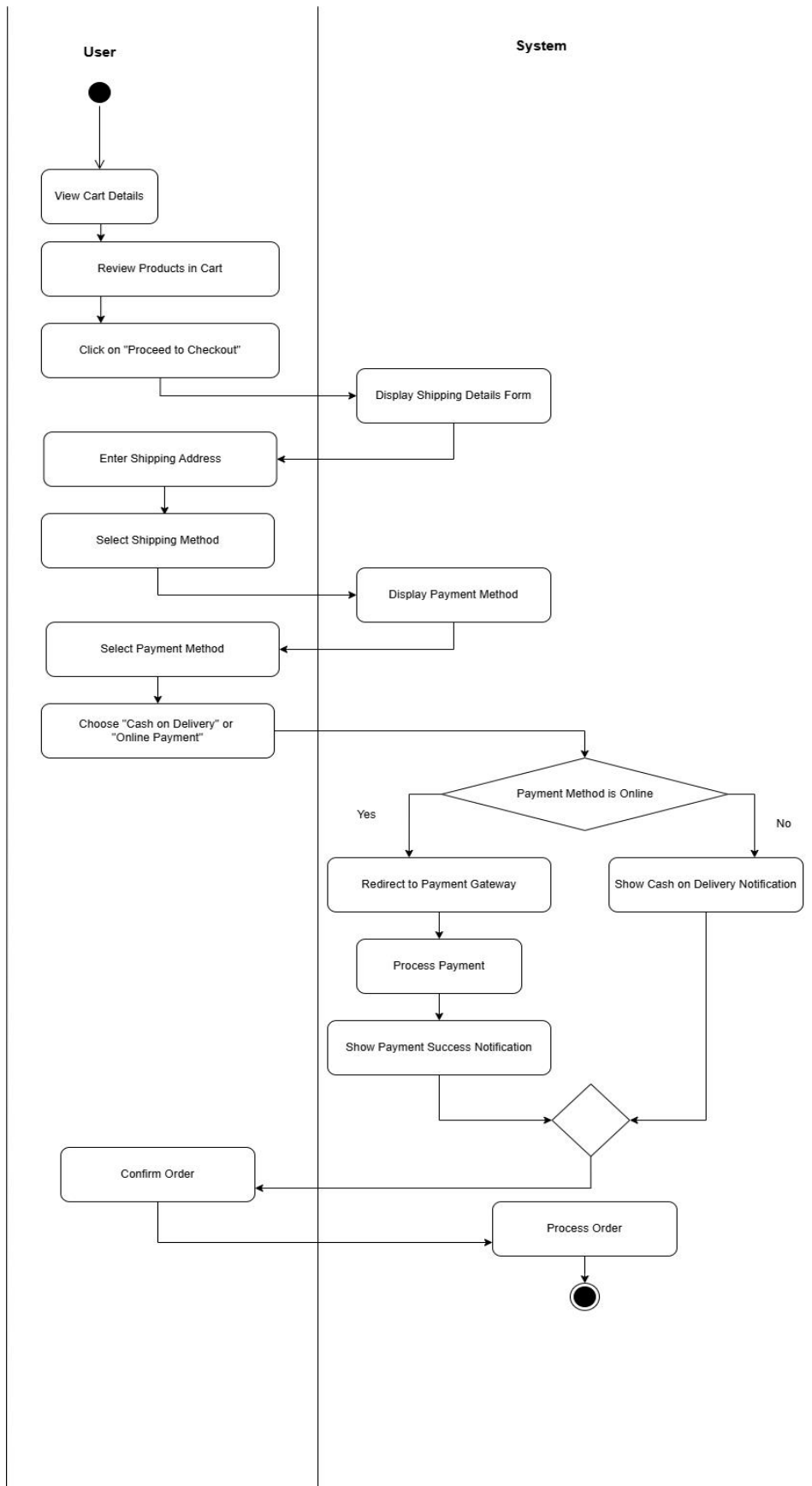


Figure 2.4.4: Activity Diagram for Checkout

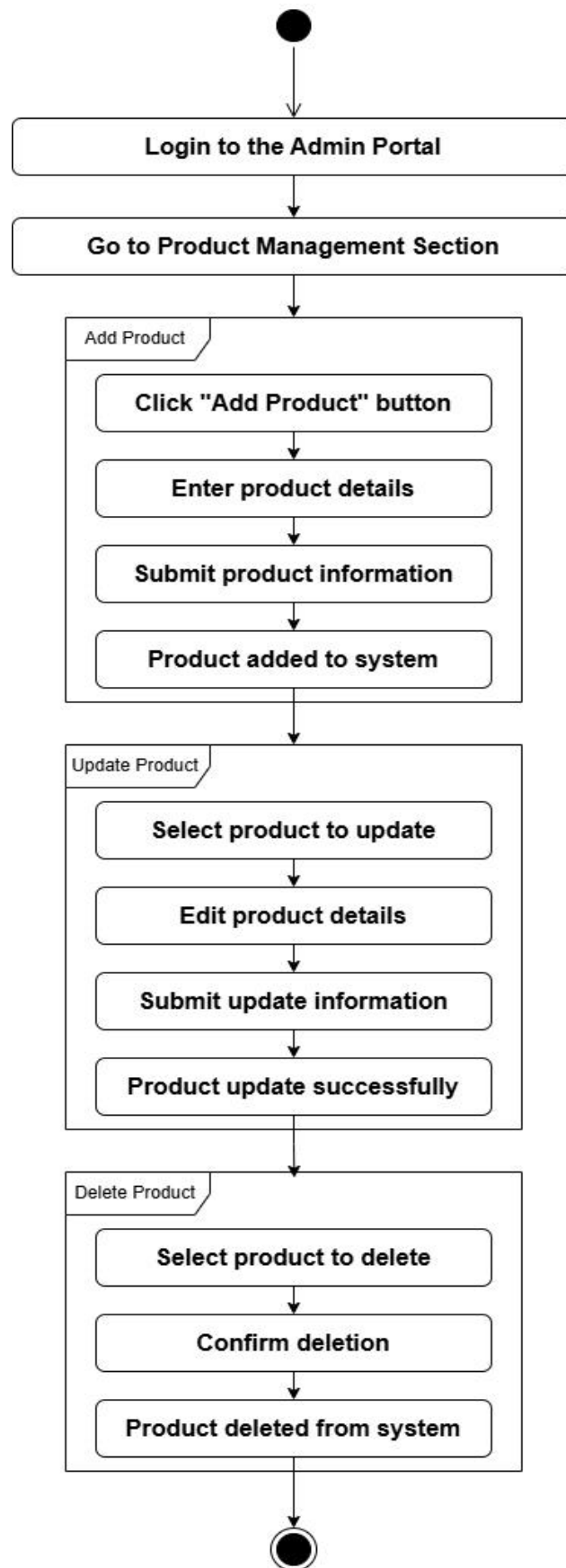


Figure 2.4.5: Activity Diagram for Add, Update & Delete Products.

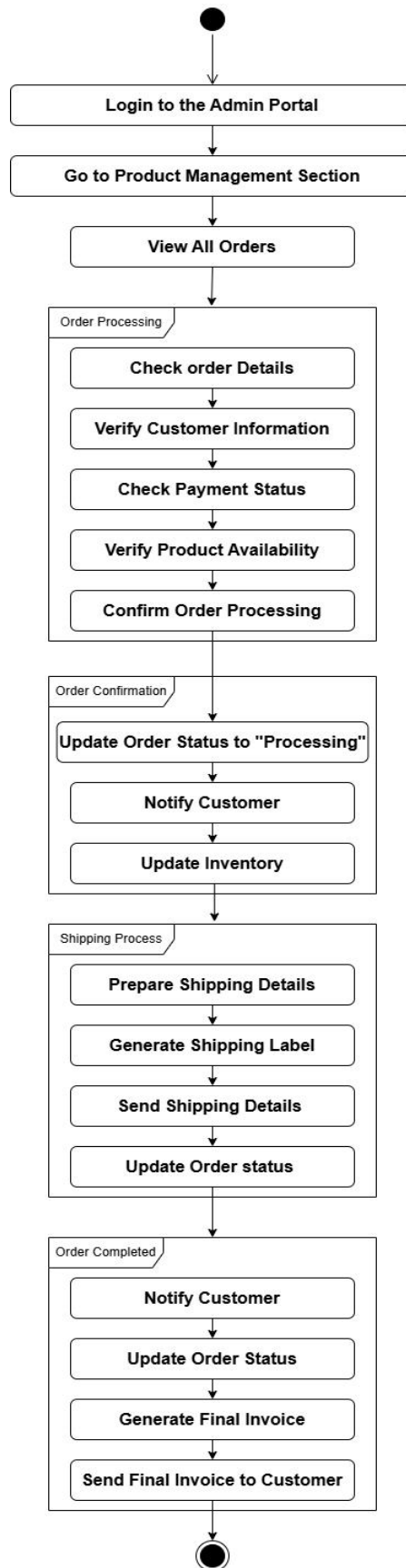


Figure 2.4.6: Activity Diagram for Manage Orders

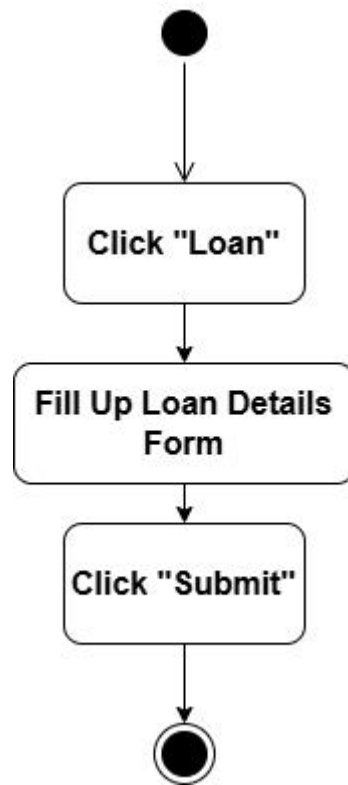


Figure 2.4.7: Activity Diagram for Take Loan

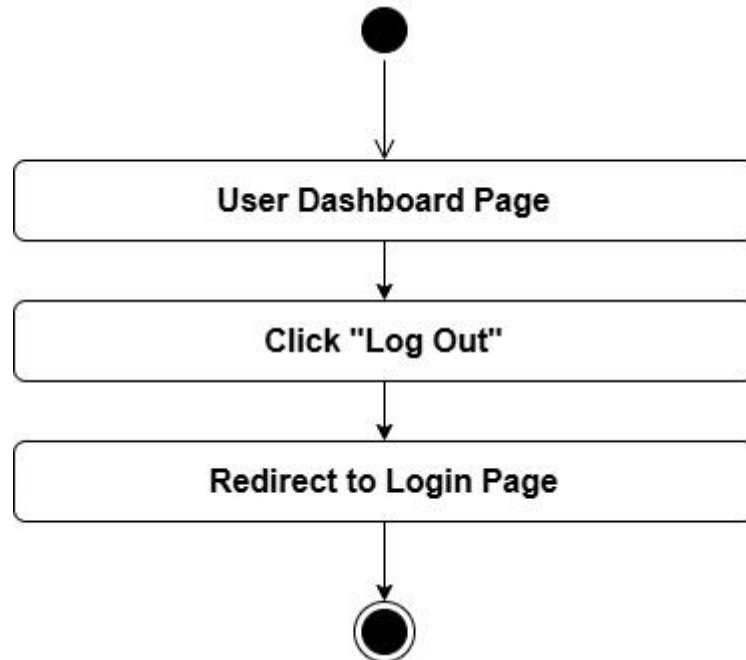


Figure 2.4.8: Activity Diagram for Logout

## 2.4.4 Sequence Diagram

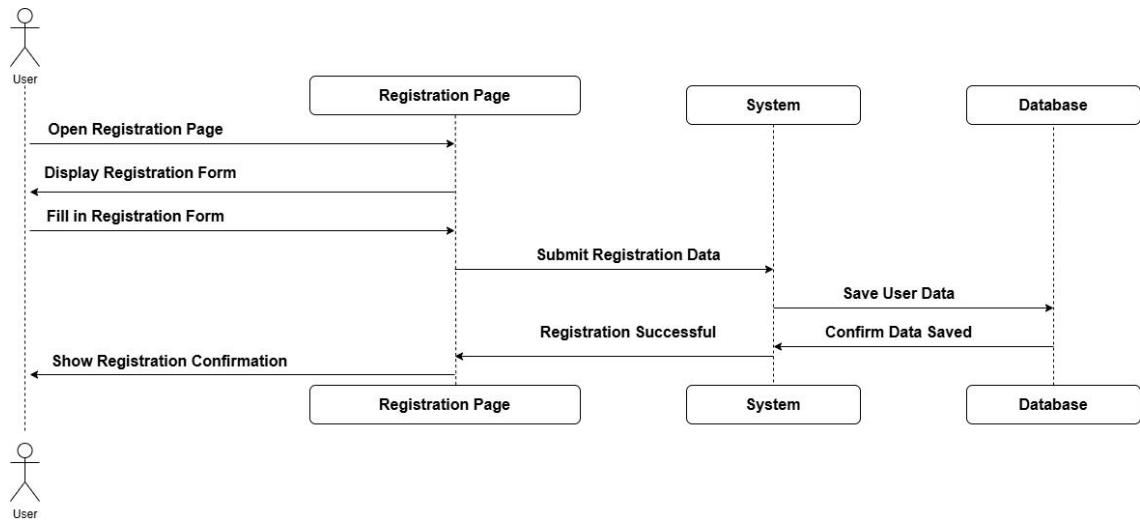


Figure 2.4.9: Sequence Diagram for User Registration

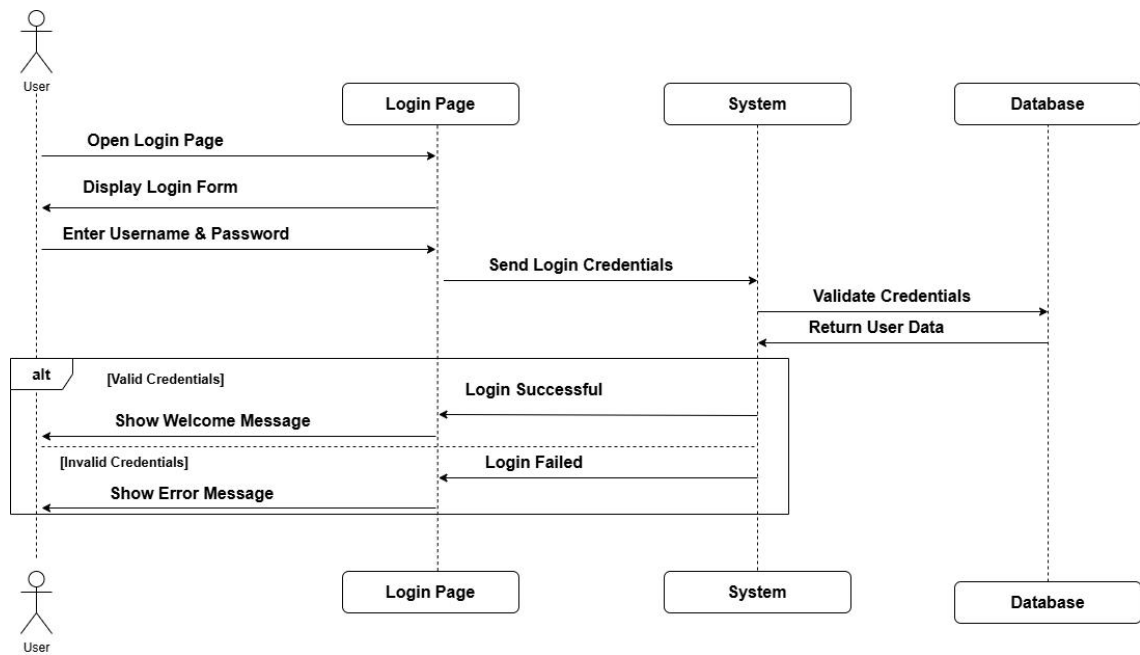


Figure 2.4.10: Sequence Diagram for User Login

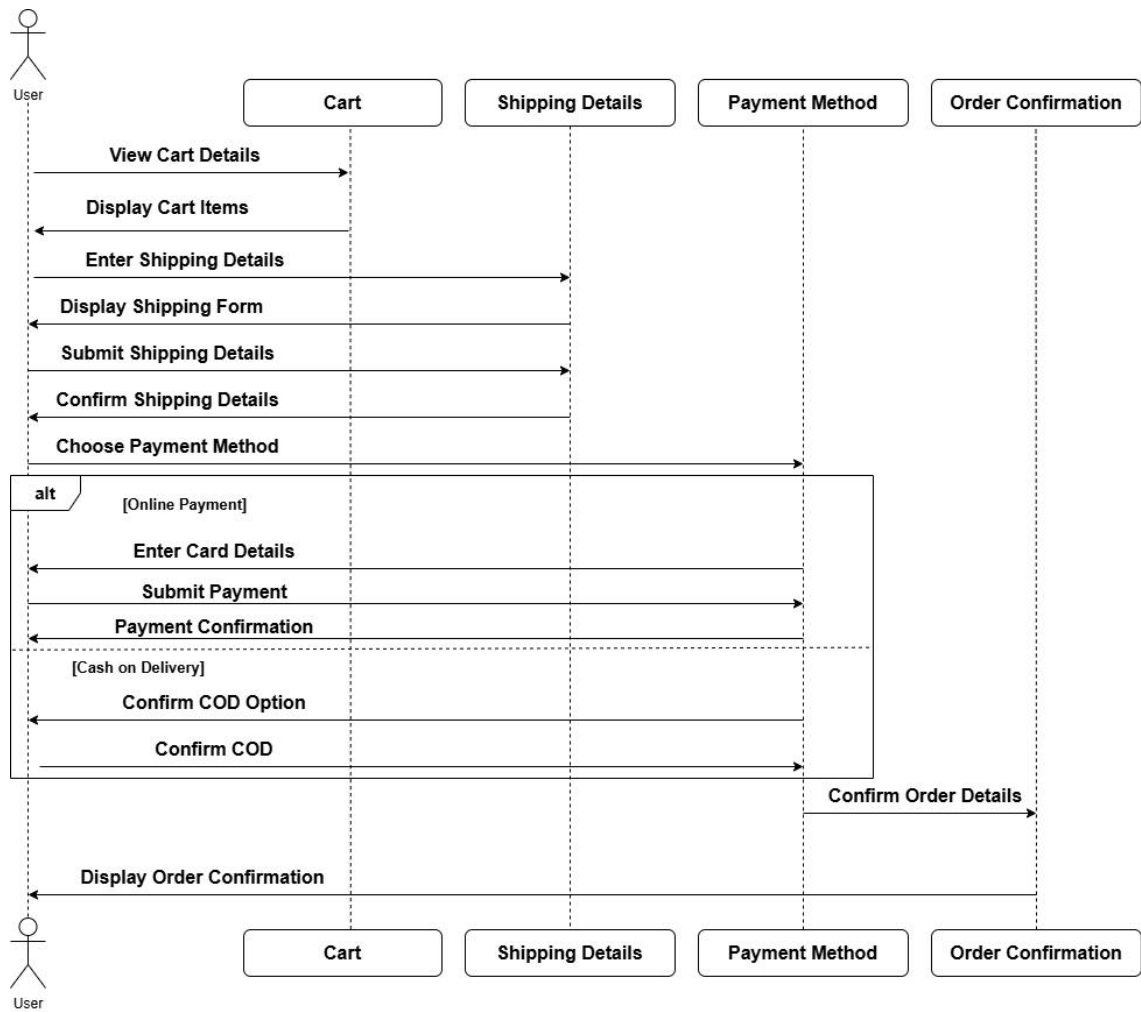


Figure 2.4.11: Sequence Diagram for User Checkout

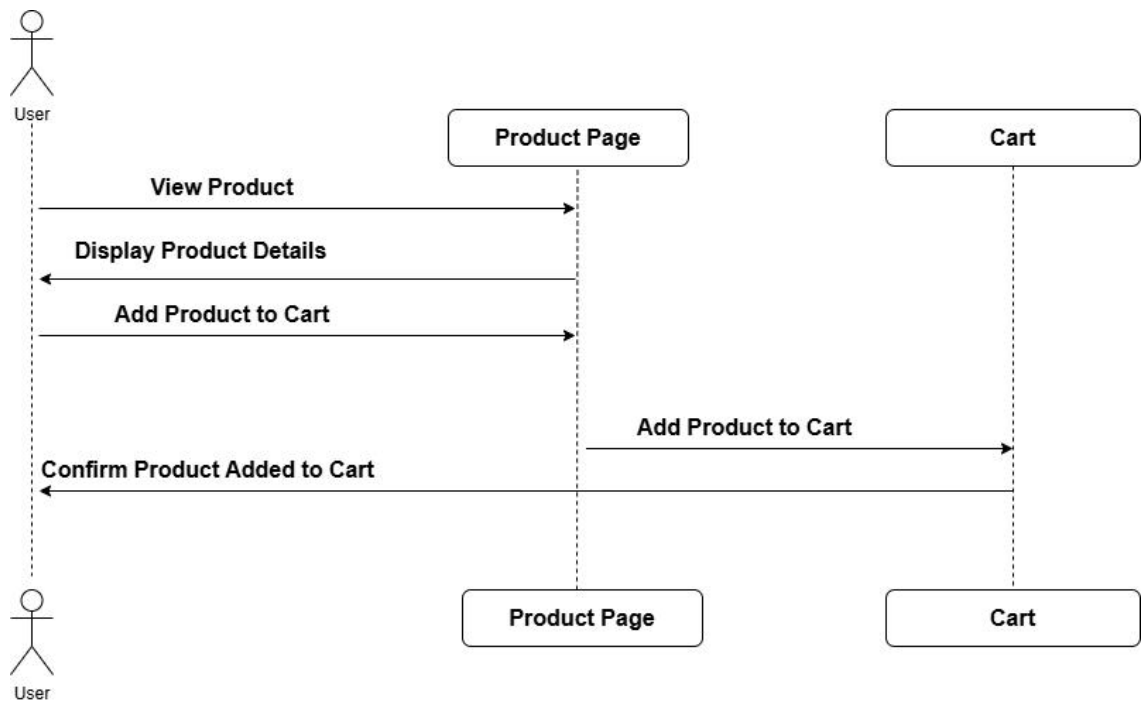


Figure 2.4.12: Sequence Diagram for Add to Cart

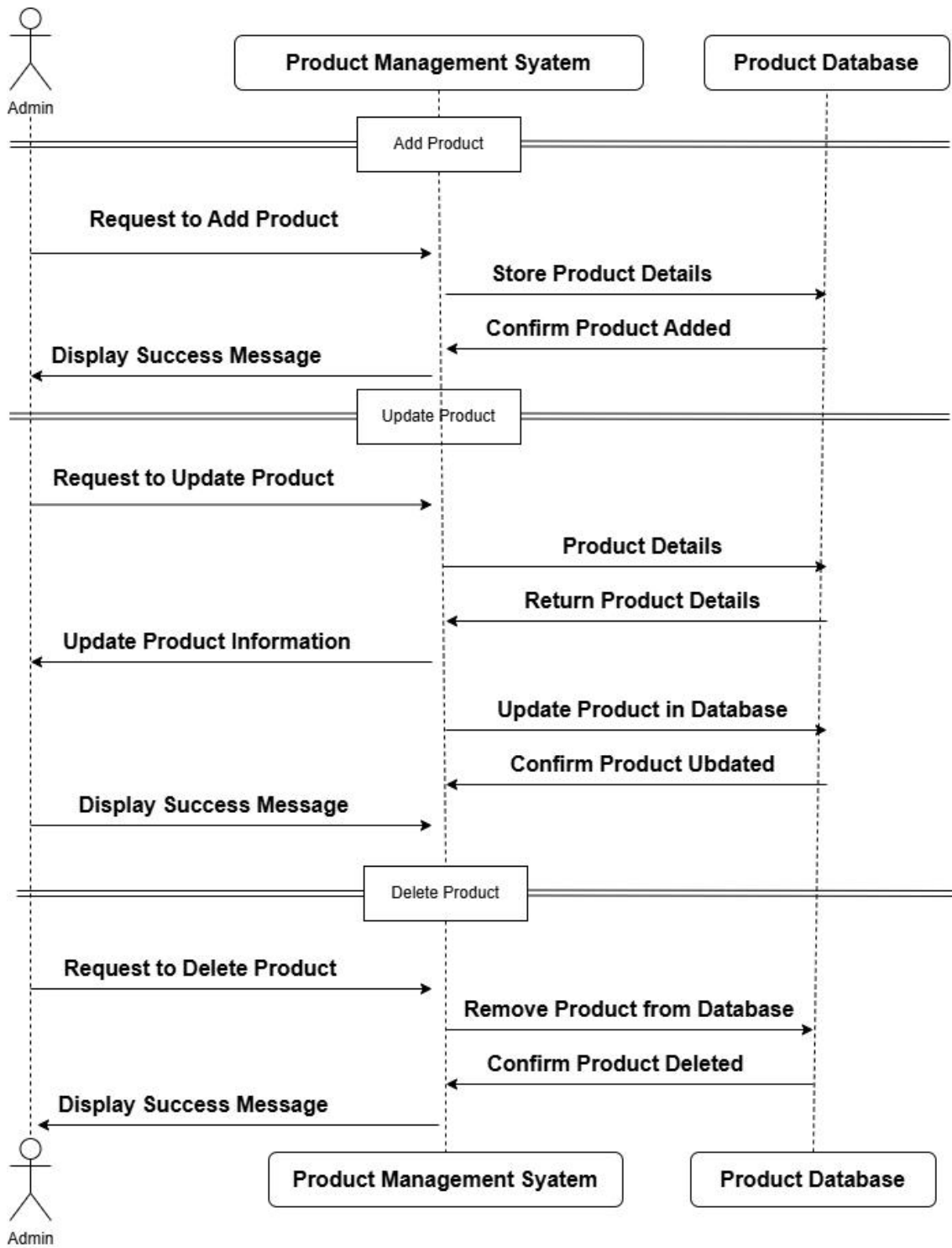


Figure 2.4.13: Sequence Diagram Add, Update & Delete

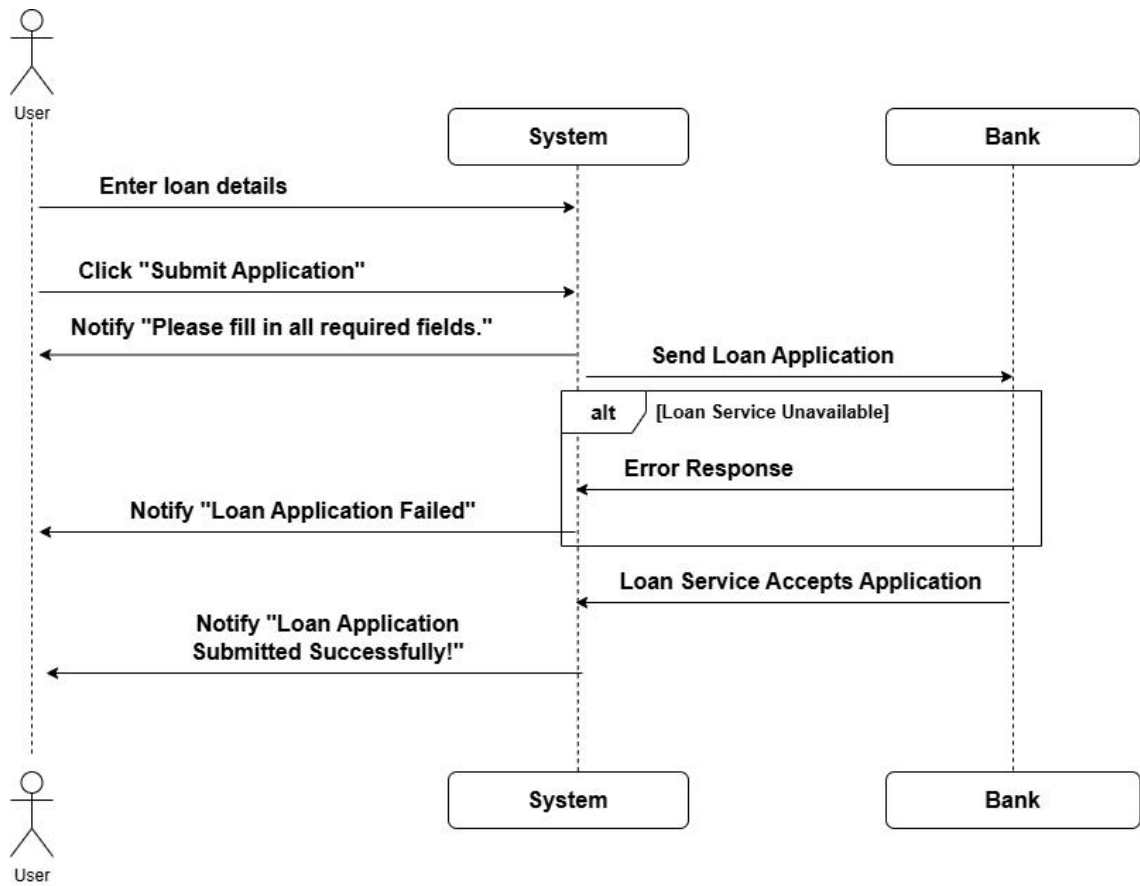


Figure 2.4.14: Sequence Diagram for Take Loan

## 2.4.5 ER Diagram

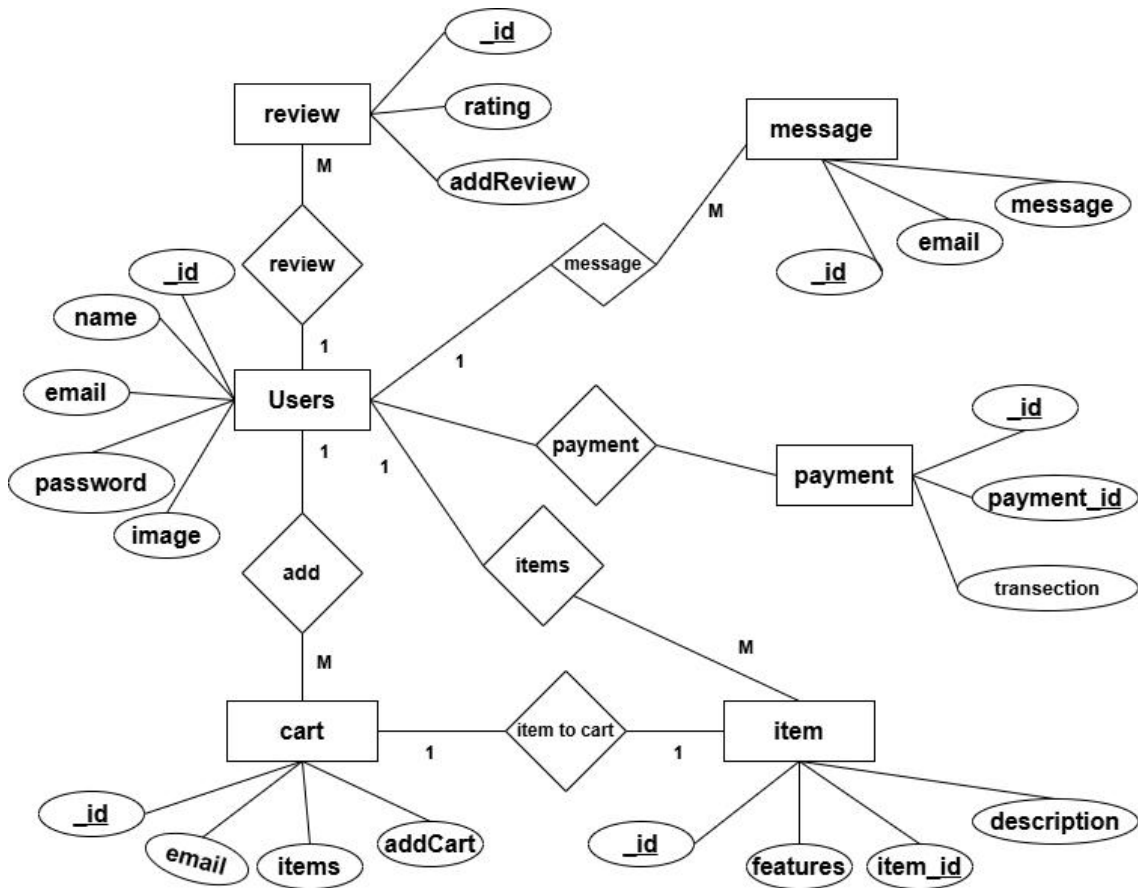


Figure 2.4.15: Entity Relationship Diagram

## 2.5 Summary

Overall view of the Agricultural E-Commerce Website ( I ) Design and realization of Program. It was initiated with the functional and non-functional requirements of the system which described what important features were required to support users effectively and how well it must work. Finally this chapter introduced object-oriented system approach to design with the UML such as Use Case Diagram, use case scenarios, Activity diagram, Sequence diagram, Class diagram and ER diagram. These are a group of models that together, depict the workings, relationships, architecture and database design of the system. On the whole, this chapter puts forth a well-defined architecture plan for the development and efficiency, usability, reliability of the platform Runway.

# Chapter 3

## Software Testing

### 3.1 Introduction

Software testing is an important phase in the development of AgroMart360 Agricultural-E-Commerce System which ensures that all the system components are working properly and also satisfies the user's requirements. The chapter presents the testing approach used to test the system, which was aimed at ensuring that the system is reliable, accurate and works efficiently. Different components and functionalities including member's login, product browsing, cart management and order placement were tested to detect any bugs, confirm the expected outcome of each process, and make sure that everything is running as it should. The objective of the tests is to ensure that the system functions properly in diverse situations and assists end-users (farmers, in particular) when they want to buy agricultural products.

### 3.2 Testing Features

#### 3.2.1 Feature to Be Tested

- a. User Registration
- b. User Login
- c. Add to Cart
- d. Checkout

### 3.3 Testing Strategies

#### 3.3.1 Test Approach

The testing strategy of AgroMart360 is intended to be also designed in such a manner that the entire software with all the functional and non-functional requirements are evaluated successively. Testing was performed to ensure that every element of the system (login, product management, user registration etc.) works as per the requirement documentation. The method also relies on black-box testing, that is, the check of system outputs for different inputs without exploiting any knowledge about the internal structure of the code. This guarantees that user interaction, validation and error handling are working as intended. Usability and workflow could also be tested through manual tests to mimic real-life cases. Positive testing, confirmation of the correct behavior for valid inputs, as well as negative testing to check error handling (invalid or incomplete input), were included in order to ensure reliability and stability of the system.

### 3.3.2 Pass/Fail Criteria

The pass/fail criterion specifies what a test must do in order for it to be considered passing. A pass of a test case is the system outputs the correct result, performs its function without error and also fulfils the specification. On the other hand, a test is fail if the system returns an incorrect output, fails to reach some error states, or does not have correct behavior. Validation checks check for such things as whether all required fields are complete, the second password is a match to the first, email addresses are in correct form, product quantities are valid and payment and shipping information is properly filled out. A feature is not considered completed until all subsequent test cases pass without major issues.

### 3.4 System Testing (Test Cases with Report)

Table 3.4.1: Test Case Report for User Registration.

<b>Test Case:</b> 3.4.1			<b>Test Case Name:</b> User Registration				
<b>System:</b> AgroMart360			<b>Subsystem:</b> User Authentication				
<b>Designed by:</b> MD. Sultanul Alam			<b>Design Date:</b> 24/11/2025				
<b>Executed by:</b> MD. Sultanul Alam			<b>Execution Date</b> 24/11/2025				
<b>Description:</b> This test case verifies whether a new user can successfully register on the AgroMart360 platform by providing valid information							
<b>Pre-condition:</b> The user have access to the registration page.							
Step	Name	Email	Password	Re-password	Response	Pass/Fail	Comment
1	Sultanul	sultanul@gmail.com	123456	123456	Registration Successfully	Pass	Registration is successful with valid input
2	Sultanul	sultanul@gmail.com	123456	123457	Password mismatch	Fail	Password is not same
3	Sultanul		123456	123456	Email filed is empty	Fail	Fill up empty field
4		sultanul@gmail.com	123456	123456	Name filed is empty	Fail	Fill up empty field
<b>Post-condition:</b> The user is successfully registered with valid information, and the process is complete.							

Table 3.4.2: Test Case Report for User Login.

<b>Test Case:</b> 3.4.2		<b>Test Case Name:</b> User Login			
<b>System:</b> AgroMart360		<b>Subsystem:</b> User Authentication			
<b>Designed by:</b> MD. Sultanul Alam		<b>Design Date:</b> 24/11/2025			
<b>Executed by:</b> MD. Sultanul Alam		<b>Execution Date</b> 24/11/2025			
		<b>Description:</b> 'Login User' Functionality Description: This test case tests that you can login in to the AgroMart360 website. It tests if users are capable of logging in with the right credentials and properly manages incorrect or missing login activity.			
		<b>Pre-condition:</b> The user must be already registered in the system.			
Step	Email	Password	Response	Pass/Fail	Comment
1	sultanul@gmail.com	123456	Login Successful	Pass	User logged in successfully with valid credentials
2	sultanul@gmail.com	000000	Incorrect Password	Fail	System denies login due to wrong password
3		123456	Email filed is empty	Fail	Fill up empty field
4	sultanul@gmail.com		Password Field is Empty	Fail	Fill up empty field
5	gfcchgcgh@gmail.com	123456	User Not Found	Fail	Email does not exist in the system
<b>Post-condition:</b> The user is successfully login with valid information, and the process is complete.					

Table 3.4.3: Test Case Report for User Add to Cart

<b>Test Case:</b> 3.4.3		<b>Test Case Name:</b> Add to Cart			
<b>System:</b> AgroMart360		<b>Subsystem:</b> Cart Management			
<b>Designed by:</b> MD. Sultanul Alam		<b>Design Date:</b> 24/11/2025			
<b>Executed by:</b> MD. Sultanul Alam		<b>Execution Date</b> 24/11/2025			
		<b>Description:</b> This test case tests customer has been able to add product and check-out it from shopping cart while allowing customer off working with the invalid operation of not found product inside shopping cart.			
		<b>Pre-condition:</b> User must be logged into the system.			
Step	Product Name	Quantity	Response	Pass/Fail	Comment
1	Rooton	2	Added to Cart Successfully	Pass	Valid product and quantity
2	NAC Salt - 84 SL	0	Invalid	Fail	Quantity must be at least 1

			Quantity		
3	Grip 52.5 WDG	5	Product Out of Stock	Fail	Cannot add unavailable product
4		1	Product Not Selected	Fail	User must select a product before adding to cart
<b>Post-condition:</b> The selected product is added to the cart with the correct quantity, and invalid inputs are rejected with appropriate messages.					

Table 3.4.4: Test Case Report for User Checkout.

<b>Test Case:</b> 3.4.4		<b>Test Case Name:</b> Checkout			
<b>System:</b> AgroMart360		<b>Subsystem:</b> Order Processing			
<b>Designed by:</b> MD. Sultanul Alam		<b>Design Date:</b> 24/11/2025			
<b>Executed by:</b> MD. Sultanul Alam		<b>Execution Date</b> 24/11/2025			
		<b>Description:</b> This test case verifies the checkout process, including entering shipping details, selecting payment methods, and confirming the order.			
		<b>Pre-condition:</b> User must be logged into the system.			
Step	Shipping Address	Payment Method	Response	Pass/Fail	Comment
1	Dhaka, Bangladesh	Cash on Delivery	Order Placed Successfully	Pass	Valid address and payment
2		Cash on Delivery	Address Field is Empty	Fail	Shipping address required
3	Dhaka, Bangladesh		Payment Method Not Selected	Fail	User must choose a payment option
4	Dhaka, Bangladesh	Online Payment	Payment Failed	Fail	Payment gateway error / invalid transaction
<b>Post-condition:</b> A successful checkout generates an order entry with shipping details and selected payment method. Failed attempts provide appropriate error messages and do not create an order					

### 3.5 Summary

This chapter presented the software testing of the AgroMart360 Agricultural E-Commerce System. It started with the significance of testing system functionality to guarantee a reliable user experience. Critical functionality such as user registration, login process, product addition to the cart and checkout have been identified and tested to check if these operations are working correctly on both legitimate and illegitimate input. The chapter also discussed the testing methodologies such as black-box, functional and manual-level testing and working on positive or negative test cases focussing into. Parametric criteria were described to assess the success of each test case.

# Chapter 4

## Deployment and Maintenance

### 4.1 Introduction

This chapter is a discussion on the deployment and maintenance of AgroMart360 Agricultural E-Commerce System. Deployment means handing the system over to end users so they can use it, and making sure that it works once delivered. Maintenance involves running the system after it is installed, repairing breakdowns and applying updates to enhance features or security. The chapter gives an overview of the early release of the system, perception it operates in and subsequent action to maintain reliability, effectiveness and user satisfaction.

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

The AgroMart360 project uses a development life cycle that focuses on iterations incorporating, testing and deploying the product for improved stability and robustness. Testing of the system was done initially in a laboratory setting to debug functional problems. A successful system test was followed by a pilot rollout to a small number of users in the field to gain forestry-wide feedback on the production build and uncover issues ahead of full roll-out. The operational live implementation brought the system to all target users who are farmers and administrators. After deployment, the maintenance phase ensures real-time tracking of system performance, timely bug fixing and evolution based on immediate user feedback or changing requirements. This is a way of ensuring that the platform remains strong, secure, and keeps pace with its users' needs over the course of its lifespan.



Figure 4.2.1: Software Release Life Cycle Diagram

# Chapter 5

## User Manual

### 5.1 Introduction

User Manual for AgroMart360 Agricultural E-commerce System This chapter will serve as the User manual for the proposed AgroMart360 The structure of this would cover both electronic procurement and product sales. It instructs the users (farmers, administrators, etc.) on how to best communicate with the system. The manual outlines step-by-step guides of key functions including registering, logging in, searching products, adding to cart, ordering and managing loans as well as operation knowledge for other platform functions. The aim of this chapter is that users can access all the features of the application easily without having a deep understanding in how the technology works.

### 5.2 Project Functionalities

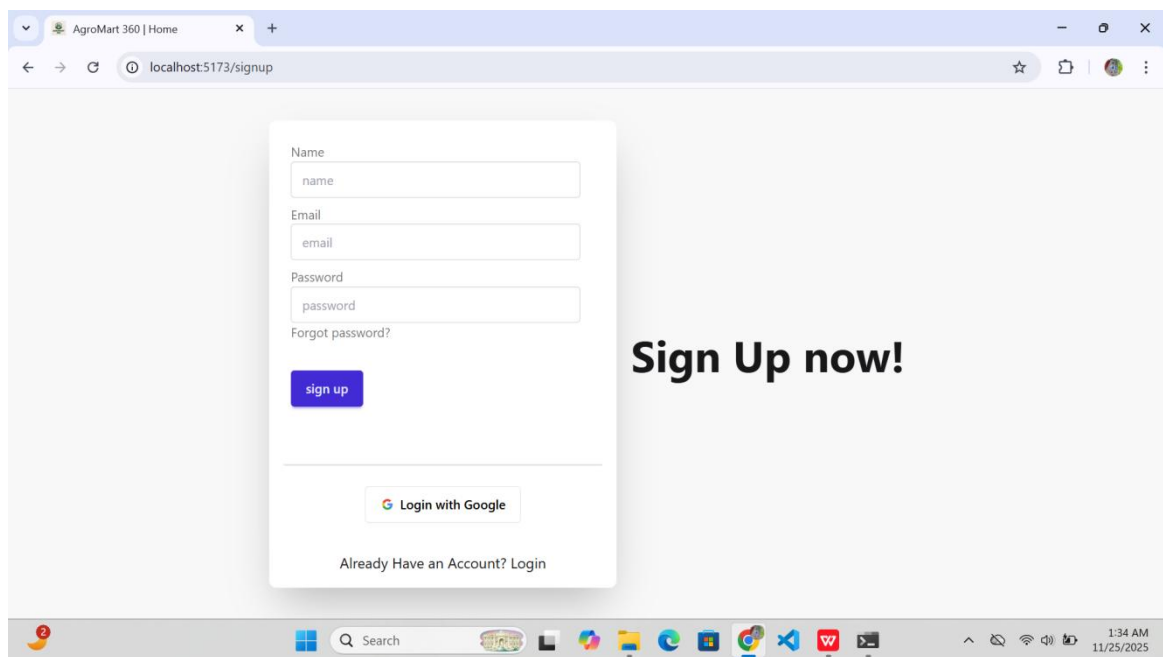


Figure 5.2.1: User Registration

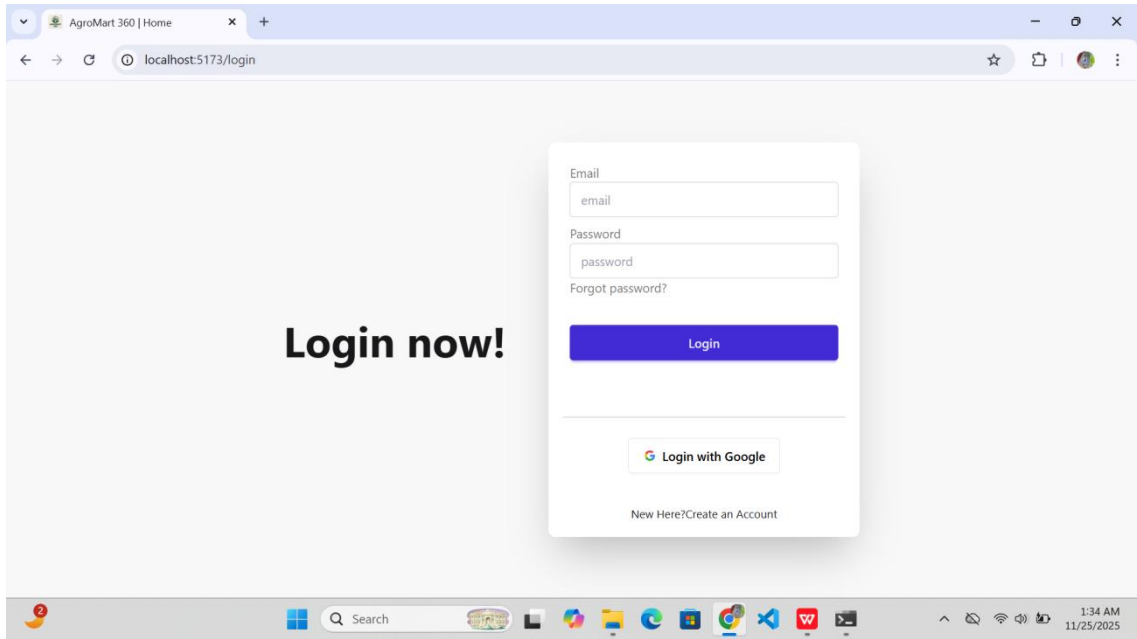


Figure 5.2.2: User Login

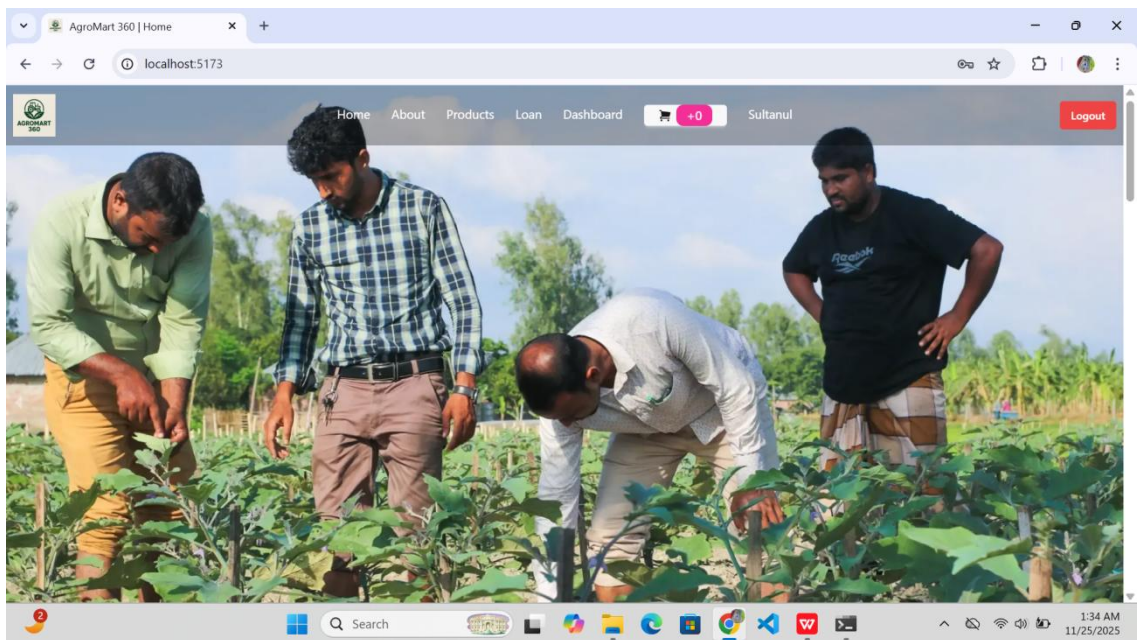


Figure 5.2.3: User Dashboard

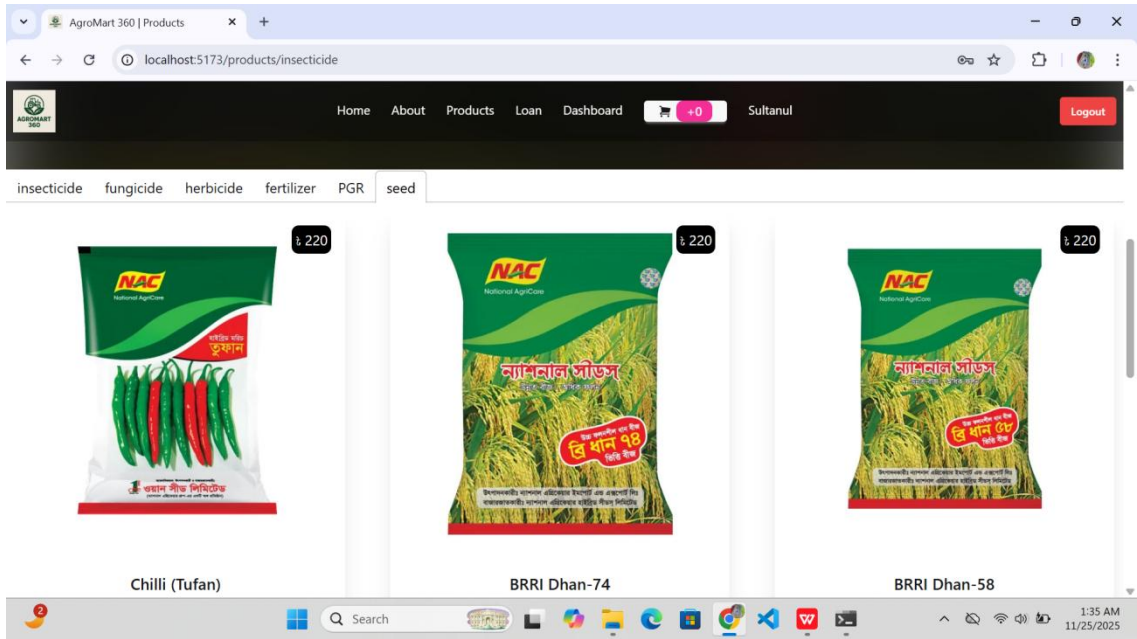


Figure 5.2.4: Products List

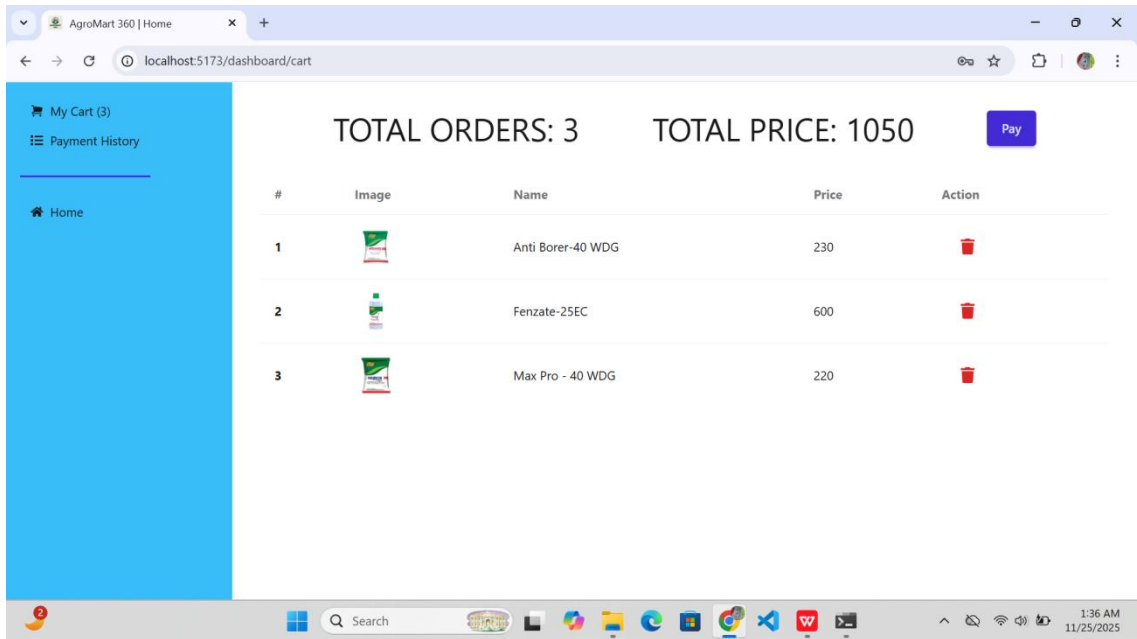


Figure 5.2.5: Products Cart List

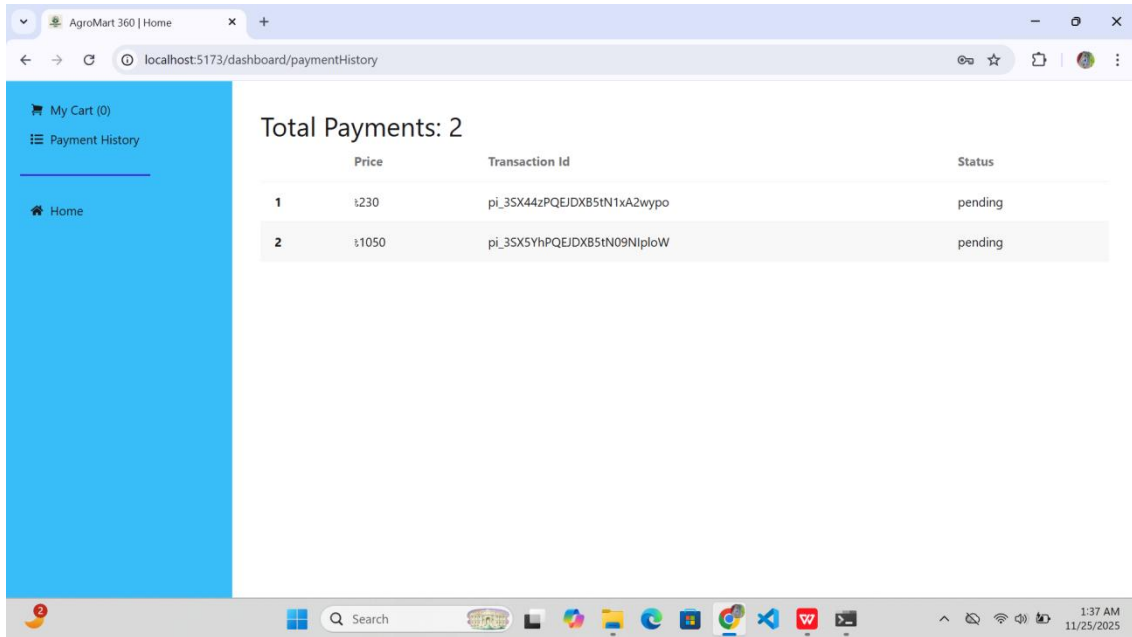


Figure 5.2.6: Payment History

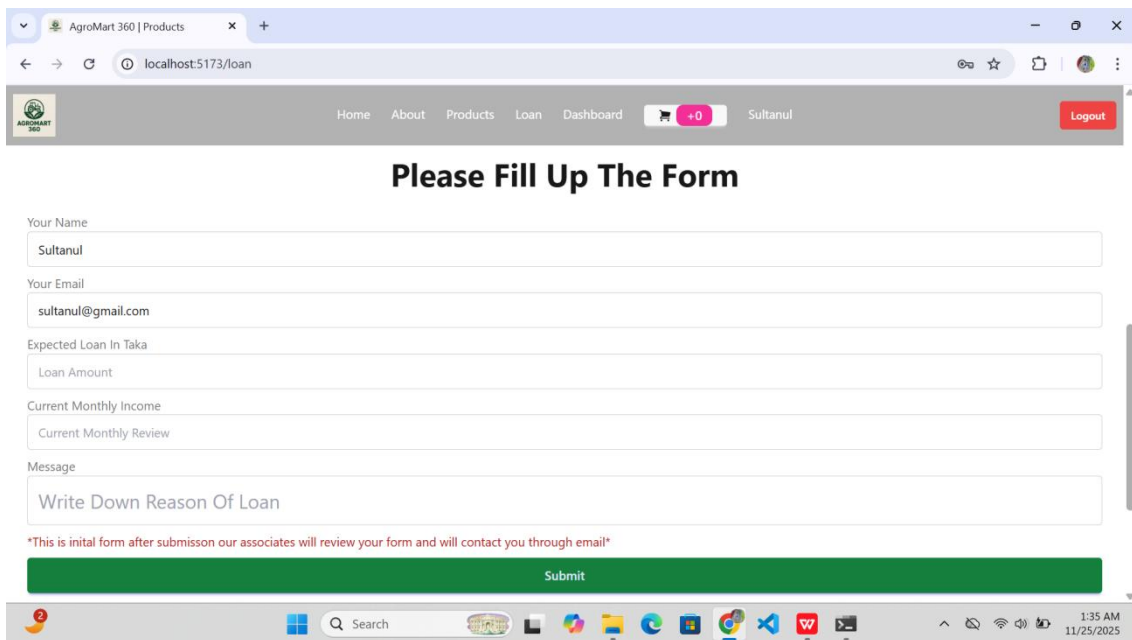


Figure 5.2.7: Loan Dashboard

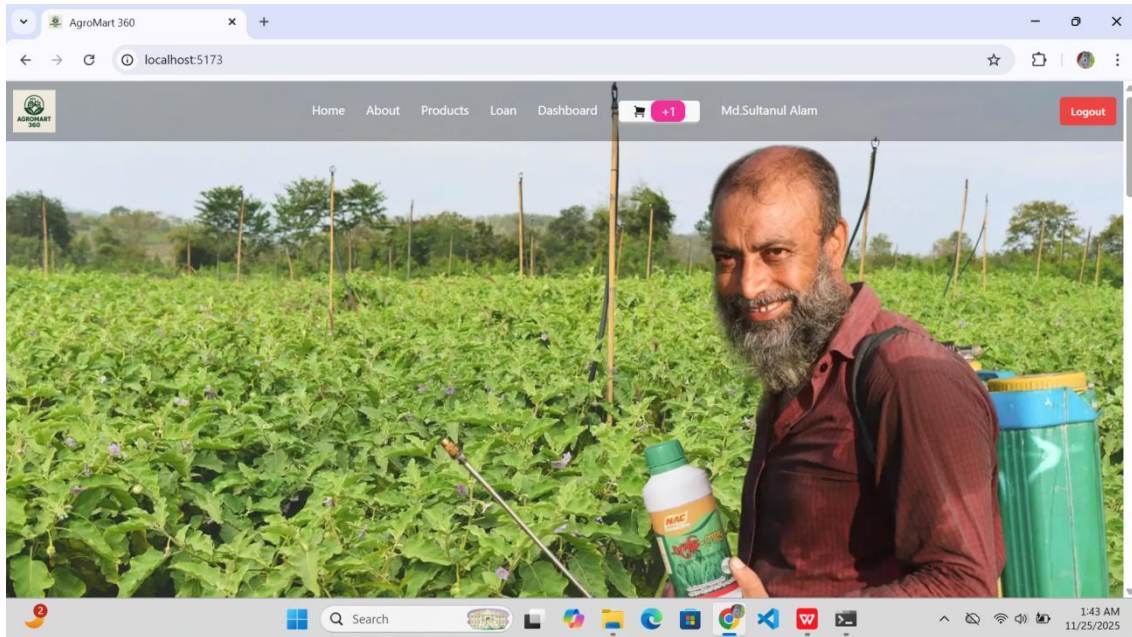


Figure 5.2.8: Admin Dashboard

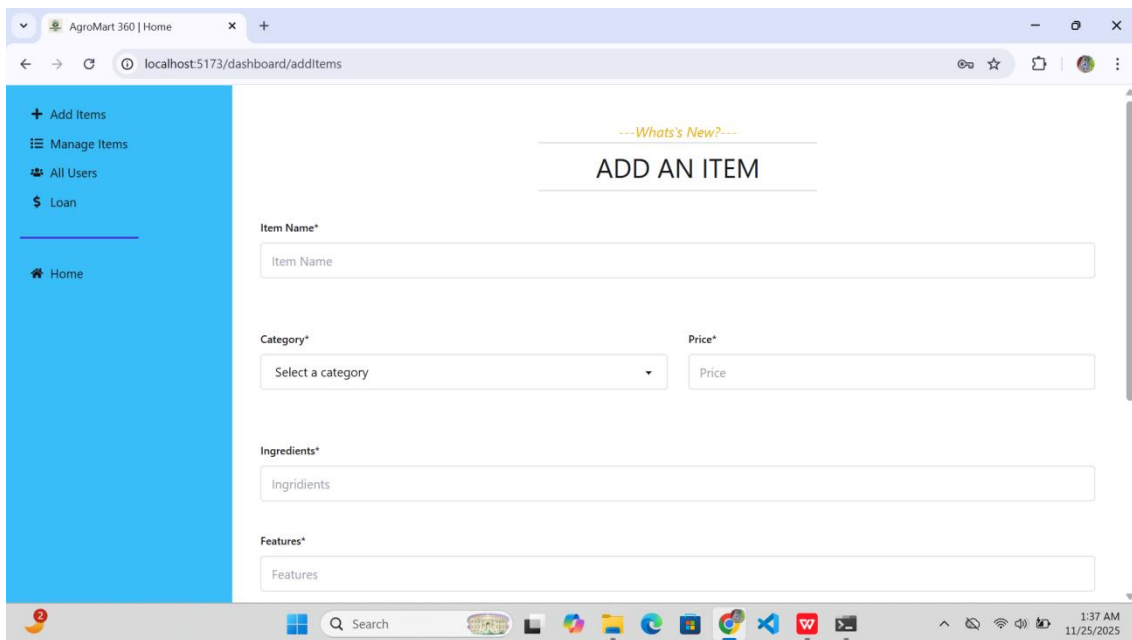


Figure 5.2.9: Admin Add Products

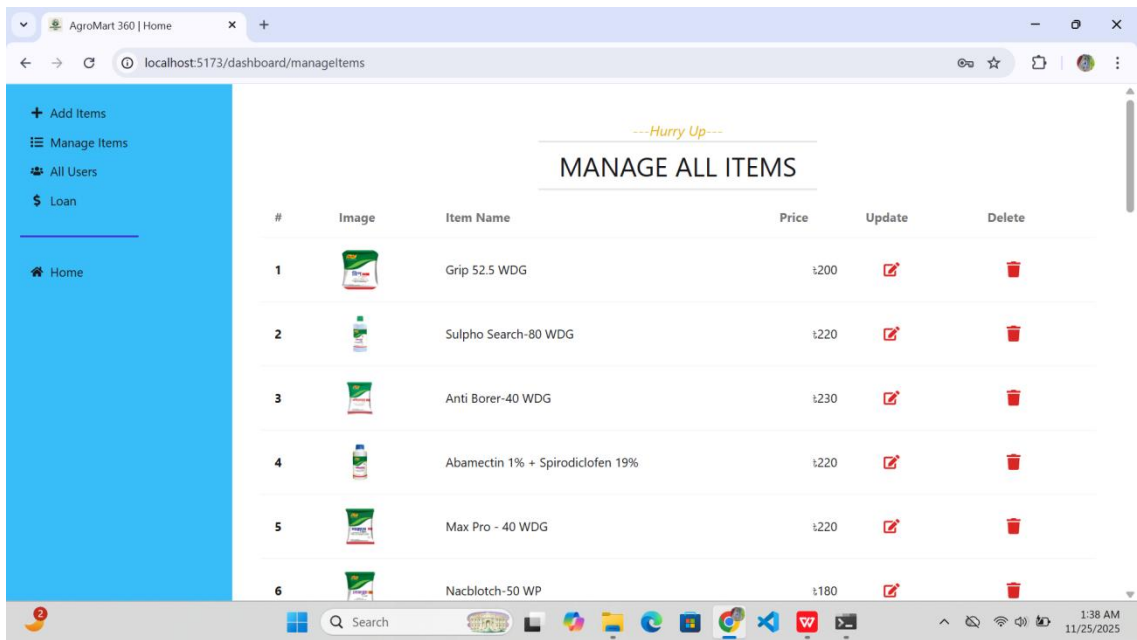


Figure 5.2.10: Admin Update & Delete products

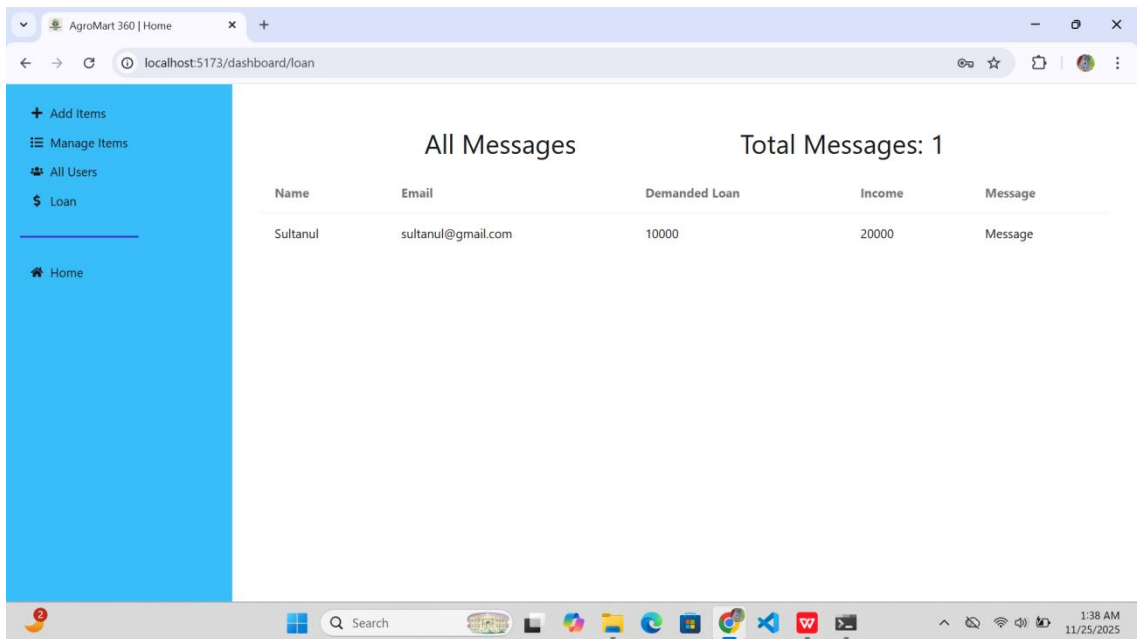


Figure 5.2.11: Admin See Loan Details

### **5.3 Summary**

This chapter provides a graphical guide of the AgroMart360 system with screen shots and short descriptions of its main functionalities. It is a good validator to show how users could achieve via your platform so that the flow and framework can be easily comprehend. With the screenshots, users can conveniently learn and control the mechanism. Such that is amenable to users, being able to save time and effort.

# Chapter 6

## Project Summary

### 6.1 Introduction

This chapter gives a brief overview of the AgroMart360 Agricultural E-Commerce System project. It involves the design, the development and the installation of phases that describe how well system works and what kind of help provides to agriculture product purchasing. The chapter also considers the limitations faced, the extent of the project, possible ways in which the system may be further developed and a final assessment about how well is project attained its aim.

### 6.2 Project Limitation

Although the AgroMart360 project was successfully developed, there were some weaknesses remarked. Due to lack of time, the depth of testing and the ability to include more advanced features was affected. Due to budget constraints cloud infrastructure also could not be extensively deployed and extra security mechanisms were not added. Infrastructure limitations, including a lack of support for integrating with third party payment gateways, also affected project scope. Therefore certain functionality like multi-language support, statistics and real time delivery tracking could not be demonstrated in this version of the system.

### 6.3 Scope

The AgroMart360 project focuses on the establishment of a digital agricultural products marketplace that provides farmers with the option of purchasing seeds, fertilizers, tools, and other inputs there. Among the features are user signup and login, product management, cart handling, order processing including loan request and checkout. It secures as well as user and device friendly. Advanced logistic management is not included in the project which is considered out of the scope of this ongoing project.

### 6.4 Future Work

Though AgroMart360 has provided a repertoire of core agricultural e-commerce and financial inclusion features, there are several areas for potential growth to help the platform become more effective, user-friendly and scalable. One significant enhancement is support for multiple secured digital payment gateways including the use of mobile financial services, bank tried to bank transfers and international payment systems. Farmers would thus have a choice and convenience in transacting at their disposal, particularly among rural farmers who prefer to use different paying systems.

Also, one more significant feature is live delivery tracking through GPS and logistic integration. If this feature were available, farmers could keep track of their orders

status all the way from order to delivery and would help improve transparency, trust and customer satisfaction as a whole. Moreover, if the language support will be provided in future, for example completely localized Bangla interface and optional multilingual services can mean more access even for less-educated farmers of such language-speaking groups.

In future releases of AgroMart360, the platform could be enhanced to include sophisticated data analytics capabilities for market demand analysis, seasonal trends monitoring, crop performance analysis and user behavior. Such knowledge can be used to optimize a seller's inventory, help an administrator make decisions, and offer more relevant product recommendations to farmers. In addition to this, tailored recommendation algorithms based on customer preference/user feedback and purchase history/farm requirement can greatly improve the shopping experience.

An extremely important addition in the future could be adding AI-based crop advisory for pest detection, fertilizer recommendation, weather based alerts and disease prediction of the crop. Such smart features would turn AgroMart360 from a marketplace to an entire smart agriculture support ecosystem. In addition, the Mini Loan product can be scaled through bank collaboration, microfinance institutions and farm cooperatives to provide flexible repayments & credit scoring systems.

The continued improvement of the system and frequent updates, with further enhancement of data security, system reliability, and future scalability as priority concerns. With these coming advancements, AgroMart360 will also evolve to be a national infrastructure of digital agriculture ecosystem that promotes sustainable farming and the well-being of rural economy.

## **6.5 Conclusion**

Conclusion AgroMart360 project has successfully developed a user-friendly practical and reliable agri-e-comm platform that will cater to the real-world problems of Bangladeshi farmers. It successfully satisfies its primary requirements when a secure digital marketplace facilitates easy availability to the farmers of authentic agricultural products including seeds, fertilizers, pesticides and various tools & implements etc. Making contract available, buyers can escape traditional middlemen and say goodbye to counterfeit products as well as pay the right price always.

One of the highlights of the project is the development 'Mini Loan Feature', lending short-term finance to farmers during crucial farming seasons. This boost to financial inclusion enables farmers to procure required input without any delay due to financial constraints. This is a good example of modern software development practices in use: the platform is well-architected, has safe transaction flow and it handles orders with no hiccup.

In general, AgroMart360 is a significant addition in taking the agricultural sector of Bangladesh towards digitalization. As the platform continues to grow and features mature, it could have a big impact on agricultural productivity, farmer income stability and sustainable rural development

## REFERENCES

1. Choruma, D. J., Dirwai, T. L., Mutenje, M. J., Mustafa, M., Chimonyo, V. G. P., Jacob-Matas, I., & Mabhaudhi, T. (2024). Digitalisation in agriculture: A scoping review of technologies in practice, challenges, and opportunities for smallholder farmers in Sub-Saharan Africa. *Journal of Agriculture and Food Research*, 18, 101286. <https://doi.org/10.1016/j.jafr.2024.101286>
2. FAO. (2022). *Digital agriculture and rural transformation*. Food and Agriculture Organization of the United Nations. <https://www.fao.org>
3. Hasan, M. M., Islam, M. R., & Rahman, M. A. (2021). E-commerce adoption and its impact on agricultural marketing in Bangladesh. *Journal of Agribusiness in Developing and Emerging Economies*, 11(4), 415–430. <https://doi.org/10.1108/JADEE-01-2020-0015>
4. Kumar, A., Singh, R., & Sharma, P. (2020). Role of digital platforms in transforming agricultural supply chains. *International Journal of Supply Chain Management*, 9(2), 456–464.
5. MongoDB Inc. (2023). *MongoDB documentation*. <https://www.mongodb.com/docs/>
6. Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
7. React.js. (2023). *React documentation*. Open Source. <https://react.dev/>
8. Roy, D., Datta, S., & Pal, A. (2019). Digital finance and financial inclusion in agriculture. *Economic & Political Weekly*, 54(17), 45–52.
9. Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson Education.
10. Stripe. (2024). *Accept payments with React*. <https://stripe.com/docs/stripe-js/react>
11. World Bank. (2021). *Digital development for agriculture*. World Bank Group. <https://www.worldbank.org>
12. Express.js. (2023). *Express documentation*. <https://expressjs.com>
13. Node.js Foundation. (2023). *Node.js documentation*. <https://nodejs.org>

