

Department of Software Engineering

University of Gujarat

AGROCONNECT



Session : BSSE 2021

Project Advisor: Dr. M. Usman Sana

Submitted By

Ahmad Mubashar 21011598-028

Ahsan Ullah 21011598-056

Ahmad Atta-Ur-Rehman 21011598-062

Department of Software Engineering
University of Gujarat

STATEMENT OF SUBMISSION

This is certify that _____ Roll No. _____ , _____ Roll No. _____ and _____ Roll No. _____ has successfully completed the final year project named as _____ at the Department of Information Technology, University of Gujrat, to fulfill the requirement of the degree of _____ **in Software Engineering.**

Project Supervisor

Project Coordination Office
Faculty of C&IT -UOG

Head of the Department

Acknowledgement

We truly acknowledge the cooperation and help make by _____, Chairman, Department of Software Engineering, University of Gujrat. He has been a constant source of guidance throughout the course of this project. We would also like to thank _____ for his help and guidance throughout this project. We are also thankful to our friends and families whose silent support led us to complete our project.

Date:

Abstract

AgroConnect is an innovative web-based platform designed to bridge the gap between farmers and consumers, enabling direct crop sales and eliminating the need for farmers to visit physical markets. The platform features crop listings, real-time market price updates, and an AI-powered assistant that provides personalized farming advice. This advice includes recommendations on best practices, suitable fertilizers, and optimal insecticide usage, all aimed at improving crop yield and quality. AgroConnect also integrates weather data, helping farmers make informed decisions about planting and harvesting. Additionally, the platform keeps farmers informed about relevant government schemes and subsidies that support agricultural activities. With an intuitive and user-friendly interface, AgroConnect ensures accessibility for farmers with varying levels of digital literacy. Secure authentication and payment systems are implemented to ensure safe transactions. By promoting sustainable practices and improving the efficiency of the agricultural supply chain, AgroConnect empowers farmers and fosters a more profitable, transparent, and sustainable agricultural ecosystem.

TABLE OF CONTENTS

CHAPTER 1: PROJECT FEASIBILITY REPORT.....	7
1.1. INTRODUCTION.....	8
1.2. PROJECT/PRODUCT FEASIBILITY REPORT.....	8
1.2.1. <i>Technical Feasibility</i>	8
1.2.2. <i>Operational Feasibility</i>	8
1.2.3. <i>Economic Feasibility</i>	8
1.2.4. <i>Schedule Feasibility</i>	8
1.2.5. <i>Specification Feasibility</i>	9
1.2.6. <i>Information Feasibility</i>	9
1.2.7. <i>Motivational Feasibility</i>	9
1.2.8. <i>Legal & Ethical Feasibility</i>	9
1.3. PROJECT/PRODUCT SCOPE	9
1.4. PROJECT/PRODUCT COSTING	9
1.4.1. <i>Project Cost Estimation By Function Point Analysis</i>	10
1.4.2. <i>Project Cost Estimation by using COCOMO'81 (Constructive Cost Model)</i>	14
1.4.3. <i>Activity Based Costing</i>	15
1.5. TASK DEPENDENCY TABLE.....	16
1.6. CPM - CRITICAL PATH METHOD.....	17
1.7. GANTT CHART	21
1.8. INTRODUCTION TO TEAM MEMBER AND THEIR SKILL SET	22
1.9. TASK AND MEMBER ASSIGNMENT TABLE.....	22
1.10. TOOLS AND TECHNOLOGY WITH REASONING	23
1.11. VISION DOCUMENT	24
1.12. RISK LIST	24
1.13. PRODUCT FEATURES/ PRODUCT DECOMPOSITION	24
CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION (FOR OBJECT ORIENTED APPROACH)	25
2.1 INTRODUCTION:	26
2.1.1 <i>Systems Specifications</i>	27
3.1.2. <i>Identifying External Entities</i>	29
2.1.3. <i>Context Level Data Flow Diagram:</i>	30
2.1.4. <i>Capture "shall" Statements:</i>	31
2.1.5. <i>Allocate Requirements:</i>	32
2.1.6. <i>Prioritize Requirements:</i>	33
2.1.7. <i>Requirements Trace-ability Matrix:</i>	34
2.2.10. <i>High Level Usecase Diagram:</i>	35
2.2.11. <i>Analysis Level Usecase Diagram:</i>	36
2.2.12. <i>Usecase Description</i>	36
CHAPTER 3: DESIGN DOCUMENT (FOR OBJECT ORIENTED APPROACH)	40
3.1. INTRODUCTION:	41
3.2. DOMAIN MODEL	41

3.3. SYSTEM SEQUENCE DIAGRAM	42
3.4. SEQUENCE DIAGRAM	43
3.4.1. <i>Defining a Sequence diagram</i>	43
3.4.2. <i>Basic Sequence Diagram Symbols and Notations</i>	43
3.5. COLLABORATION DIAGRAM.....	48
3.6. OPERATION CONTRACTS	48
3.7. DESIGN CLASS DIAGRAM.....	50
3.8. STATE CHART DIAGRAM	51
3.9. DATA MODEL.....	53

Chapter 1: Project Feasibility Report

1.1. Introduction

1.2. Project/Product Feasibility Report

The feasibility report evaluates the practicality of implementing the AgroConnect platform, assessing its technical, operational, financial, and market feasibility. It aims to determine whether the project is viable within the given constraints, ensuring that the objectives of connecting farmers with consumers and providing valuable resources for agricultural improvement can be successfully achieved.

There are many types of feasibilities:

- Technical
- Operational
- Economic
- Schedule
- Specification
- Information
- Motivational
- Legal and Ethical

1.2.1. Technical Feasibility

AgroConnect's technical feasibility is ensured through the use of scalable web technologies like React and Node.js. AI-driven farming advice can be implemented using existing machine learning models, while real-time market data can be fetched via APIs. Secure payment gateways and data protection measures will ensure safe transactions. Cloud services will provide reliable storage and server management for platform growth.

1.2.2. Operational Feasibility

AgroConnect is designed to be user-friendly, ensuring accessibility for farmers with minimal digital literacy. The AI-driven farming advice and simple navigation make it easy for users to interact with the platform. Customer support will be available to assist users with any operational issues. Regular updates will ensure platform remains efficient.

1.2.3. Economic Feasibility

AgroConnect's economic feasibility is supported by a sustainable business model, including revenue from transaction fees, subscription services, and potential partnerships. The initial costs include platform development, marketing, and server infrastructure. Over time, the platform aims to become self-sustaining through increased user adoption and partnerships with agricultural organizations. Cost-effective cloud services and efficient operational management will ensure long-term profitability.

1.2.4. Schedule Feasibility

AgroConnect's development is feasible within the proposed timeline, with key milestones set for each phase. The platform's core features and AI integration can be implemented on schedule. Adequate resources and planning ensure a timely and efficient launch.

1.2.5. Specification Feasibility

The platform can support features like crop listings, AI-powered advice, and real-time market prices using existing technologies. Additionally, the system's design and infrastructure will accommodate future scalability and integration of additional features.

1.2.6. Information Feasibility

AgroConnect can access reliable data for market prices, weather updates, and farming advice through APIs and public sources. The platform will integrate accurate, real-time information for informed decision-making. AI-driven recommendations will be based on available agricultural datasets, ensuring relevance and accuracy.

1.2.7. Motivational Feasibility

AgroConnect is highly motivating for farmers as it empowers them to access better market prices, improve crop yields, and reduce dependency on intermediaries. The platform offers personalized AI advice, which increases farmers' confidence in adopting new farming practices. By providing direct connections to consumers and supporting financial growth, AgroConnect enhances farmers' motivation to embrace digital solutions and improve their agricultural productivity.

1.2.8. Legal & Ethical Feasibility

AgroConnect ensures legal feasibility by complying with data protection laws, including GDPR, to protect user privacy and transaction information. The platform will follow relevant agricultural and e-commerce regulations, ensuring fair trade practices. Ethically, it promotes transparency, supports sustainable farming practices, and empowers farmers without exploiting their resources, fostering trust and accountability in the agricultural sector.

1.3. Project/Product Scope

The AgroConnect project aims to develop an online platform that connects farmers directly with consumers, enabling the sale of crops without intermediaries. Key features include crop listings, real-time market prices, AI-powered farming advice, weather integration, and access to government schemes. The platform will cater to farmers of varying digital literacy levels, with a user-friendly interface and secure payment systems. It will support scalability to accommodate future features, such as product management, order tracking, and payment integration. The project focuses on improving the efficiency and profitability of farmers while promoting sustainable agricultural practices.

1.4. Project/Product Costing

The project/product costing for AgroConnect involves several components, including development, operational, and maintenance costs. Initial development costs cover expenses related to software development, design, and infrastructure setup, acquiring necessary technology, and ensuring secure systems. Operational costs include ongoing expenses for cloud server hosting, security measures, and customer support services. Marketing and promotional activities are also an essential part of the operational budget, aimed at reaching a wider audience and increasing user adoption. Maintenance costs will cover regular

updates, bug fixes, system improvements, and AI model training to ensure the platform remains efficient and up-to-date. Transparent pricing mechanisms will be essential for the platform's sustainability, balancing affordability for users while generating enough revenue to support continuous development and operational costs. The overall costing strategy aims to create a profitable yet user-friendly platform, ensuring long-term growth and value for both farmers and consumers.

1.4.1. Project Cost Estimation By Function Point Analysis

Function point analysis is the process of sizing software based on the number of business functions an application must accomplish. It eliminates the need to tediously go through each line of code to determine if an application meets business needs and has the required quality.

Table 1.4.1.1: Unadjusted function points

Type of Component	Complexity of Component			
	Low	Average	High	Total
External Inputs		31 x 4 =		124
External Outputs		6 x 5 =		30
External Inquiries		4 x 4 =		16
Internal Logical Files		5 x 10 =		50
External Interface Files		3 x 7 =		21
Total number of unadjusted function points				241

Information domain values are defined in the following manner:

User Inputs (External Inputs):

User Registration

- Full Name
- Email
- Phone Number
- Password
- Confirm Password
- Account Type (Farmer or Customer)

- Province
- City

Login

- Email
- Password

Order Management:

- Customer Name
- Order Id
- Crop Ordered
- Quantity
- Order Status (Completed/Pending)

Product Listing (For Farmers)

- Crop Name
- Quantity Available
- Price Per Unit

Order Placements (For Customers)

- Selected crop
- Quantity
- Payment Method

AI Farming Advice Request

- Current Weather Data (optional)
- Detect Disease
- Provide Medical Suggestion

Feedback Submission

- Rating
- Review Text

Government Scheme Information:

- Scheme Availability
- Eligibility Criteria

Payment Integration:

- Amount to Pay
- Payment Method
- Payment Confirmation

External Outputs:**Farmer Dashboard:**

- Crop Listings
- Order Status

Market Price Updates

- Average market prices of major crops

Order Confirmation

- Order Details (Id, Crop, Quantity, Price)

AI Farming Advice

- Suggested Fertilizer and Insecticides

Payment Confirmation:

- Payment Status and Amount

External Inquiries:**Incorrect Login Information**

- Error message for invalid credentials

Out of Stock Notification

- Message indicating the requested crop is unavailable

Weather Data Retrieval Failure

- Message when weather data fails to load

Connection Termination

- Notification if the session times out or disconnects

Internal Logical Files (ILF):**Farmer Profiles**

- Details on registered farmers, including contact and farm information

Customer Profiles

- Customer account details and purchase history

Crop Listing Database

- Crop listings with quantity, price, and quality information

Order Records

- Detailed records of all placed orders, status, and transactions

AI Advice Logs

- Logs of AI-generated farming advice to track and improve recommendations

External Interface Files (EIF):

Weather API Interface

- Real-time data interface to fetch local weather conditions

Market Price API Interface

- External source for retrieving average crop market prices

Payment Gateway Interface

- Payment processing system to handle transactions securely

To compute function points (FP), the following relationship is used:

$$\text{FP est.} = \text{Count Total} * [0.65 + 0.01 * (Fi)]$$

Table 1.4.1.2: VAF (Valued Adjustment Factor)

Sr. No.	Value Adjustment Factor (VAF)	Values
1.	Data communication	5
2.	Distributed data processing	4
3.	Performance	4
4.	Heavily used configuration	3
5.	Transaction rate	3
6.	Online data entry	2
7.	End-user efficiency	4
8.	Online update	2
9.	Complex processing	4
10.	Reusability	3
11.	Installation ease	4
12.	Operational ease	4
13.	Multiple sites	1
14.	Facilitate change	3
	Total VAF	46

$$\begin{aligned}\text{AFP} &= \text{UFP} \times [0.65 + 0.01 \times \text{VAF}] \\ &= 241 \times (0.65 + 0.01 \times 46) \\ &= 241 \times 1.11 = 267.51\end{aligned}$$

Lines of Code (LOC) Estimation:

$$\text{LOC} = 25 \times 267.51 = 6687.51 \approx 6688 \text{ LOC} \\ = 6.688 \text{ KLOC}$$

Finally, Total Project Cost and Total Project Effort are calculated given the average productivity parameter for the system.

The formulae are given as follows:

$$\text{Cost / FP} = \text{labor rate / productivity parameter} \quad \text{Total Project Cost} = \text{FP est.} * (\text{cost / FP})$$

$$\text{Total Estimated Effort} = \text{FP est. / productivity parameter}$$

1.4.2. Project Cost Estimation by using COCOMO'81 (Constructive Cost Model)

Boehm's COCOMO model is one of the mostly used models commercially. The first version of the model delivered in 1981 and COCOMO II is available now. COCOMO 81 is a model that allows one to estimate the cost, effort, and schedule when planning a new software development activity, according to software development practices that were commonly used in the 1970s through the 1980s. It exists in three forms, each one offering greater detail and accuracy the further along one is in the project planning and design process. Listed by increasing fidelity, these forms are called Basic, Intermediate, and Detailed COCOMO. However, only the Intermediate form has been implemented by USC in a calibrated software tool.

Three levels:

Basic: Is used mostly for rough, early estimates.

Intermediate: Is the most commonly used version, includes 15 different factors to account for the influence of various project attributes such as personnel capability, use of modern tools, hardware constraints, and so forth.

Detailed: Accounts for the influence of the different factors on individual project phases: design, coding/testing, and integration/testing. Detailed COCOMO is not used very often.

Each level includes three software development types:

1. **Organic:** Relatively small software teams develop familiar types of software in an in-house environment. Most of the personnel have experience working with related systems.
2. **Embedded:** The project may require new technology, unfamiliar algorithms, or an innovative new method
3. **Semi-detached:** Is an intermediate stage between organic and embedded types.

Basic COCOMO

Type	Effort	Schedule
Organic	PM= 2.4 (KLOC) ^{1.05}	TD= 2.5(PM) ^{0.38}
Semi-Detached	PM= 3.0 (KLOC) ^{1.12}	TD= 2.5(PM) ^{0.35}
Embedded	PM= 2.4 (KLOC) ^{1.20}	TD= 2.5(PM) ^{0.32}

PM= person-month (effort)

KLOC= lines of code, in thousands

TD= number of months estimated for software development (duration)

Our project falls under **Organic class**:

Now we will put values in the given formulas:

Calculate Effort:

$$PM = 2.4 * (KLOC)^{1.05} \quad PM = 2.4 * (6.688)^{1.05}$$

$$PM = 2.4 * 7.35463421$$

$$PM = 17.6511221$$

Calculate Schedule:

$$TD = 2.5 * (PM)^{0.38}$$

$$TD = 2.5 * (17.6511221)^{0.38}$$

$$TD = 2.5 * 2.9769633$$

$$TD = 7.44240825 \text{ Number of Months}$$

Calculate People Required:

$$\text{People Required} = \text{Effort} / \text{Duration}$$

$$= 17.6511221 / 7.44240825$$

$$= 2.37169496$$

Calculate Productivity:

$$\text{Productivity} = \text{FP} / \text{Effort}$$

$$\text{Productivity} = 267.51 / 17.6511221$$

$$\text{Productivity} = 15.1554104$$

1.4.3. Activity Based Costing

Table 1.4.3 Activity Based Costing

No.	Activity Name	Duration (Days)	Cost
1	Requirement Analysis	14	7,000
2	Front-end Development	28	15,000
3	Backend Development	28	25,000
4	Integration (Frontend & Backend)	14	10,000

5	AI Bot Implementation	21	12,000
6	System Testing	10	10,000
7	Issues Resolved	7	5,000
8	Deployment	5	20,000

1.5. Task Dependency Table

Table 1.5: Task Dependency Table

Task No.	Tasks	Duration	Dependencies
T1	Idea selection and proposal submission		None
T2	Requirement Analysis	14 days	T1
T3	SRS Submission	21 days	T2
T4	Design Document	28 days	T3
T5	Front-end Development	28 days	T4
T6	Backend Development	21 days	T5
T7	Integration (Frontend & Backend)	14 days	T6
T8	System Testing	7 days	T7
T9	Issues Found	2 days	T8
T10	Issues Resolved	7 days	T9
T11	Verification	7 days	T10
T12	Deployment	3 days	T11

1.6. CPM - Critical Path Method

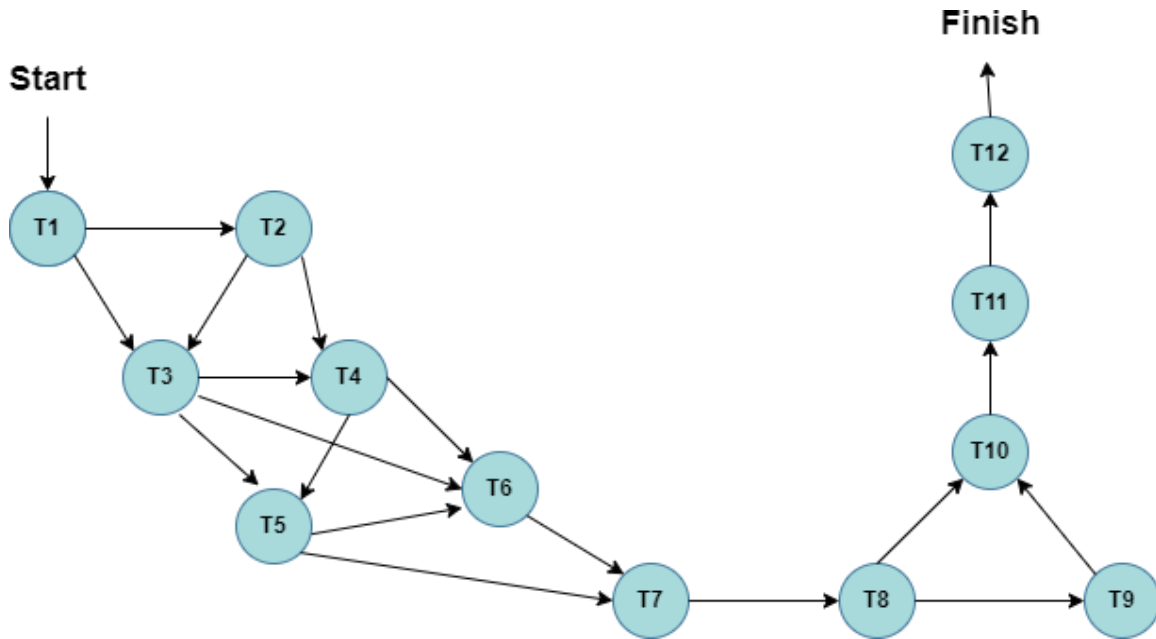


Figure 1.6: Network Diagram

1. Specify the Individual Activities

- 1) Idea Selection and proposal submission
- 2) Req. gathering and analysis
- 3) Design Document
- 4) SRS Submission
- 5) Front-end Development
- 6) Backend Development
- 7) Integration (Frontend & Backend)
- 8) System Testing
- 9) Issues Found
- 10) Issues Resolved
- 11) Verification
- 12) Deployment

2. Determine the Sequence of the Activities

Table 1.6.1: Sequence Activity

T.ID	Activity	Duration (Days)	Immediate Predecessors
T1	A	14	None
T2	B	14	A
T3	C	21	A, B
T4	D	21	B, C
T5	E	28	C, D
T6	F	28	C, D, E
T7	G	14	E, F
T8	H	7	G
T9	I	2	H
T10	J	7	H, I
T11	K	7	J
T12	L	21	K

3. Draw the Network Diagram

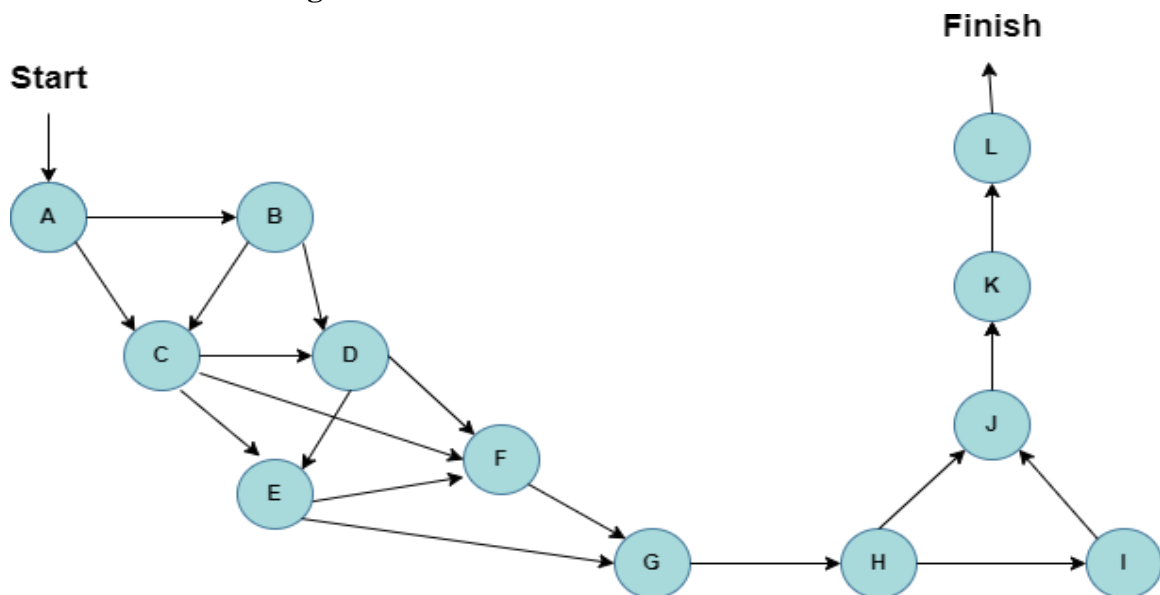


Figure 1.6.1: CPM Network Diagram

4. Estimate Activity Completion Time

Table 1.6.2: Activity Completion Time Table

T.ID	Activity Name	Duration (Days)
T1	Idea Selection and proposal Submission	14
T2	Requirement Analysis	14
T3	SRS Submission	21
T4	Design Document	21
T5	Front-end Development	28
T6	Backend Development	28
T7	Integration (Frontend & Backend)	14
T8	System Testing	7
T9	Issues Found	2
T10	Issues Resolved	7
T11	Verification	7
T12	Deployment	21

5. Identify the Critical Path

Table 1.6.3: CPM Table

T. ID	Activi ty	Duration (Days)	ES	EF	LS	LF	TS
T1	A	14	0	14	0	14	0
T2	B	14	14	28	14	28	0
T3	C	21	28	49	28	49	0
T4	D	21	49	70	49	70	0
T5	E	28	70	98	70	98	0
T6	F	28	98	126	98	126	0
T7	G	14	126	140	126	140	0
T8	H	7	140	147	140	147	0
T9	I	2	147	149	147	149	0
T10	J	7	149	15	149	156	0
T11	K	7	156	163	156	163	0

As the critical path is the path with the highest value of duration, so:

The **critical path** is:

A-B-C-D-E-F-G-H-I-J-K-L= 184 Days

With T.ID:

T1-T2-T3-T4-T5-T6-T7-T8-T9-T10-T11-T12= 184 Days

6. Update CPM Diagram

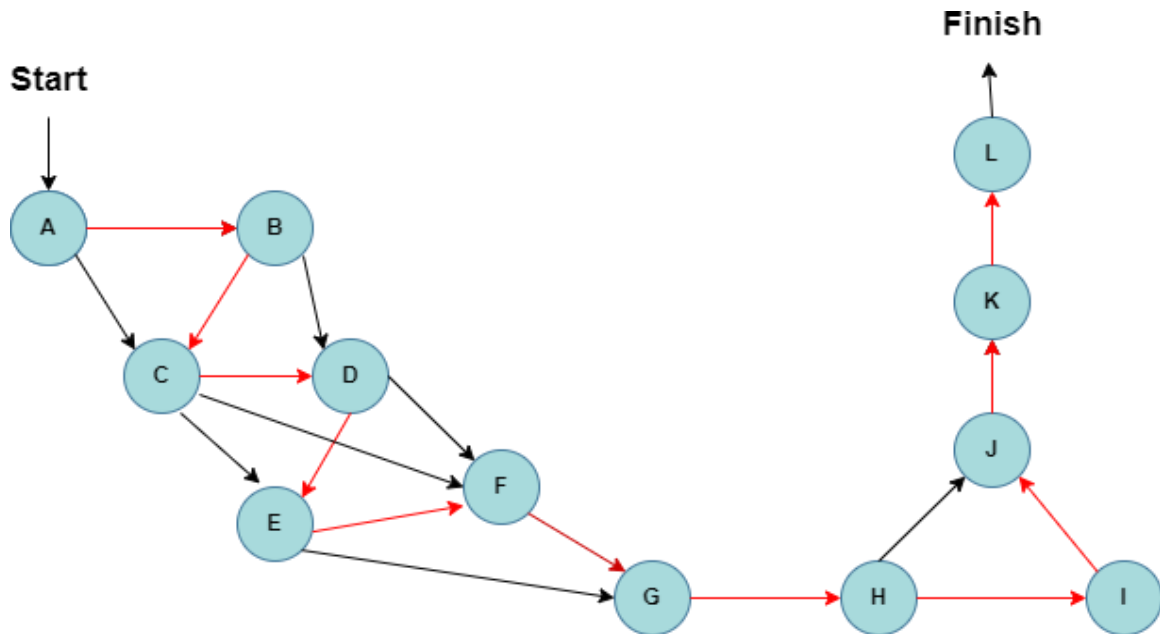


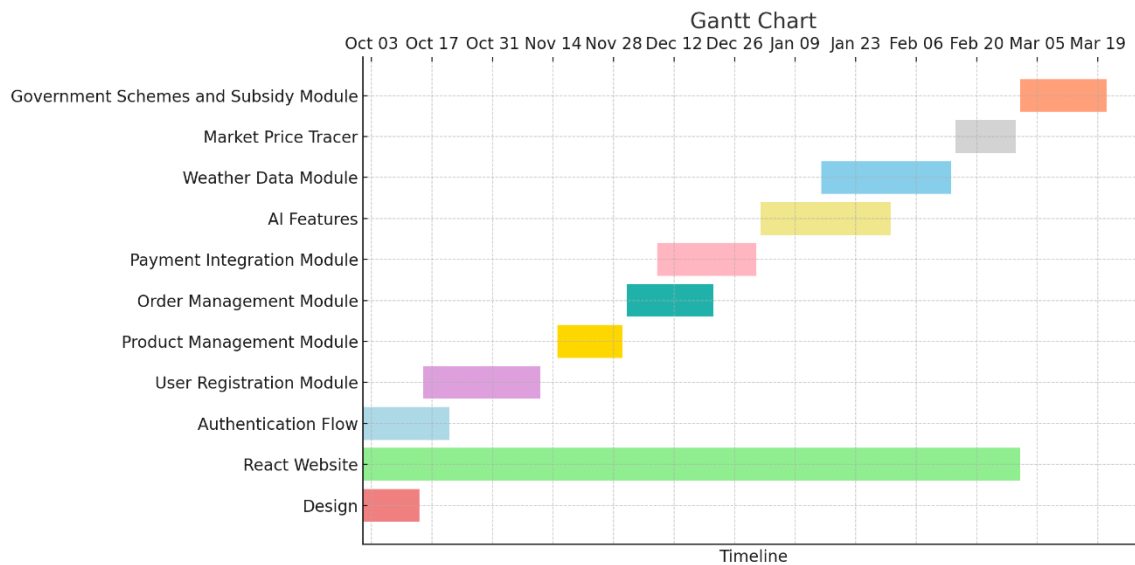
Figure1.6.2: Network Diagram with Critical point

The **critical path** is:

A-B-C-D-E-F-G-H-I-J-K-L= 184 Days

1.7. Gantt chart

Table 1.7 Gantt Chart



1.8. Introduction to Team member and their skill set

Ahsan Ullah (21011598-056)	Ahmed Mubasher (21011598-028)	Ahmad Atta (21011598-062)
Design Document	Front End Development	Document Testing
Requirements Gathering	Requirements Gathering	Backend Development
MS Word Expert	MS Word Expert	MS Word Expert
Front End Development	Testing Document	Integration

1.9. Task and Member Assignment Table

Ahsan Ullah = M1
 Ahmad Mubashar = M2
 Ahmad Atta = M3

Table 1.9: Task Assign

T.ID	Activity Name	Duration (Days)	Depende ncies	Members
T1	Idea Selection and proposal Submission	14	None	M1, M2, M3
T2	Req. Gathering and Analysis	14	T1	M1, M2
T3	SRS Submission	21	T2	M1, M2, M3
T4	Design Document	21	T3	M1
T5	Front-end Development	28	T4	M1, M2
T6	Back-end Development	28	T5	M3
T7	Integration	14	T6	M1, M2, M3
T8	System Testing	7	T7	M1, M2, M3
T9	Issue Found	2	T8	M1, M2, M3
T10	Issue Solve	7	T9	M1,M2, M3
T11	Verification	7	T10	M1, M2, M3
T12	Deployment	21	T11	M1, M2, M3

1.10. Tools and Technology with reasoning

Tools and Technology with reasoning:

- Visual Studio Code
- Git Hub
- Figma

Languages/Frameworks/Packages for Frontend:

Name	Reasoning
React.js	React.js provides a component-based structure, ensuring a dynamic and responsive user interface suitable.
Python	Python libraries simplify implementing AI models for farming advice and predictive analytics.
Figma	Figma provides an intuitive platform for designing responsive and user-friendly interfaces.

Languages/Packages for Backend:

Name	Reasoning
Django	Offer rapid development, scalability, and built-in security features to manage user data, crop listings, and payment processing.
PostgreSQL	Provide robust data integrity, scalability, and advanced querying capabilities to handle large datasets like crop listings, market prices, and user transactions.

1.11. Vision Document

AgroConnect envisions empowering farmers by providing a platform to sell crops directly to customers, eliminating middlemen, and increasing profitability. The website offers real-time crop prices, AI-driven farming advice on fertilizers and pest control, and easy access to government schemes and subsidies. Stakeholders include farmers, customers, and agricultural experts, with benefits like transparent pricing, optimized farming practices, and secure transactions. The project assumes farmers have basic internet access and customers are willing to purchase fresh produce directly. AgroConnect addresses inefficiencies in the agricultural market by combining technology with practical solutions for farming challenges.

1.12. Risk List

1. Technical Risks

- **System Downtime:** Potential server outages affecting platform availability.
- **Data Security:** Risk of data breaches exposing sensitive farmer or customer information.
- **AI Model Accuracy:** Inaccurate recommendations from the AI farming assistant.

2. Financial Risks

- **Budget Overrun:** Exceeding the allocated project budget due to unforeseen expenses.
- **Payment Failures:** Issues with payment processing, leading to financial disputes.

3. User Adoption Risks

- **Farmer Resistance:** Reluctance from farmers to adopt digital platforms due to lack of trust or technical knowledge.
- **Customer Retention:** Difficulty in maintaining a steady customer base due to market competition.

4. Market Risks

- **Inconsistent Market Data:** Delays or inaccuracies in fetching real-time crop prices.
- **Regulatory Changes:** Sudden policy shifts affecting platform operations or compliance.

5. Operational Risks

- **Team Coordination:** Miscommunication or delays within the development team.
- **Resource Availability:** Unavailability of skilled personnel or tools during critical phases.

1.13. Product Features/ Product Decomposition

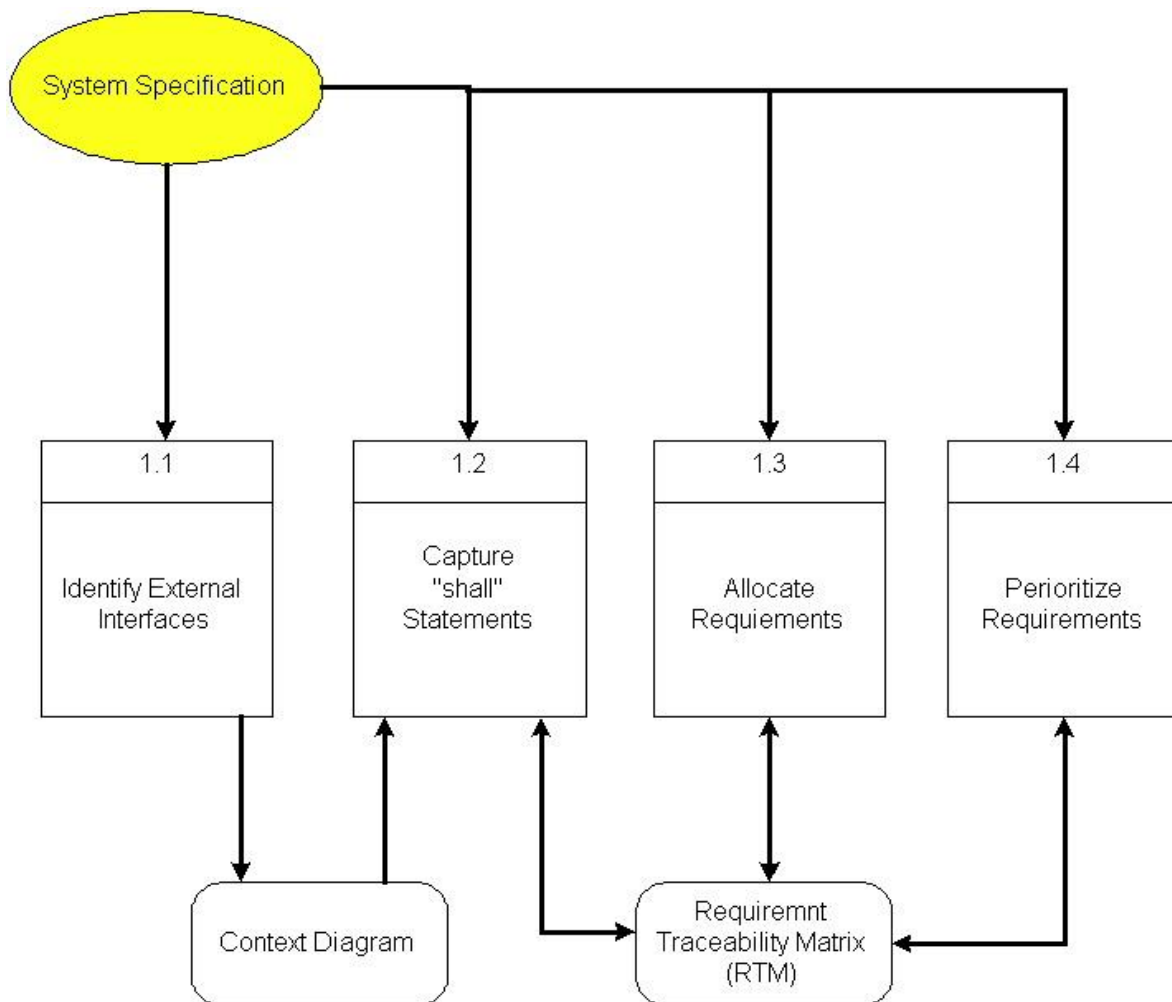
- User Management
- Crop Marketplace
- Real-Time Market Prices
- AI Farming Assistant
- Order Management
- Payment Integration

Chapter 2: Software Requirement Specification (For Object Oriented Approach)

2.1 Introduction:

Requirements engineering process provides the appropriate mechanism for understanding what the customer wants, analyzing need, assessing feasibility, negotiating a reasonable solution, specifying the solution unambiguously, validating the specification and managing the requirements as they are transformed into an operational system. The task of capturing, structuring, and accurately representing the user's requirements so that they can be correctly embodied in systems which meet those requirements (i.e. are of good quality).

- Requirements elicitation
- Requirements analysis and negotiation
- Requirements specification
- System modeling
- Requirements validation
- Requirements management



Here, requirements specification is to be discussed. Requirements specification would lead to the following four steps:

- Identify external interfaces
- Development of context diagram
- Capture “shall statements
- Allocate requirements
- Prioritize requirements
- Development of requirements traceability matrix

2.1.1 Systems Specifications

The following are the clauses that must be included while describing the system specifications.

Introduction

AgroConnect is an innovative web-based platform designed to transform the agricultural market by providing a seamless connection between farmers and customers. The system enables farmers to list and sell their crops directly to consumers, eliminating middlemen and ensuring better pricing for both parties. AgroConnect aims to empower local farmers with real-time market prices, AI-driven farming advice, and access to government schemes. The platform also helps farmers optimize their operations by providing personalized recommendations for fertilizers, pest control, and weather updates. With a focus on improving transparency, efficiency, and profitability, AgroConnect is poised to make a significant impact on the agricultural industry, providing farmers with tools to increase productivity and sustainability.

Existing System

The existing systems in the agricultural market are primarily characterized by traditional farming practices, middlemen, and limited access to technology. Farmers often rely on local markets or intermediaries to sell their crops, which leads to inefficiencies, price manipulation, and reduced profitability. In terms of market prices, farmers typically rely on word-of-mouth or outdated information, resulting in a lack of transparency. Additionally, many farmers struggle with optimizing their farming practices due to a lack of access to tailored advice on soil health, pest control, and suitable fertilizers.

Organizational Chart

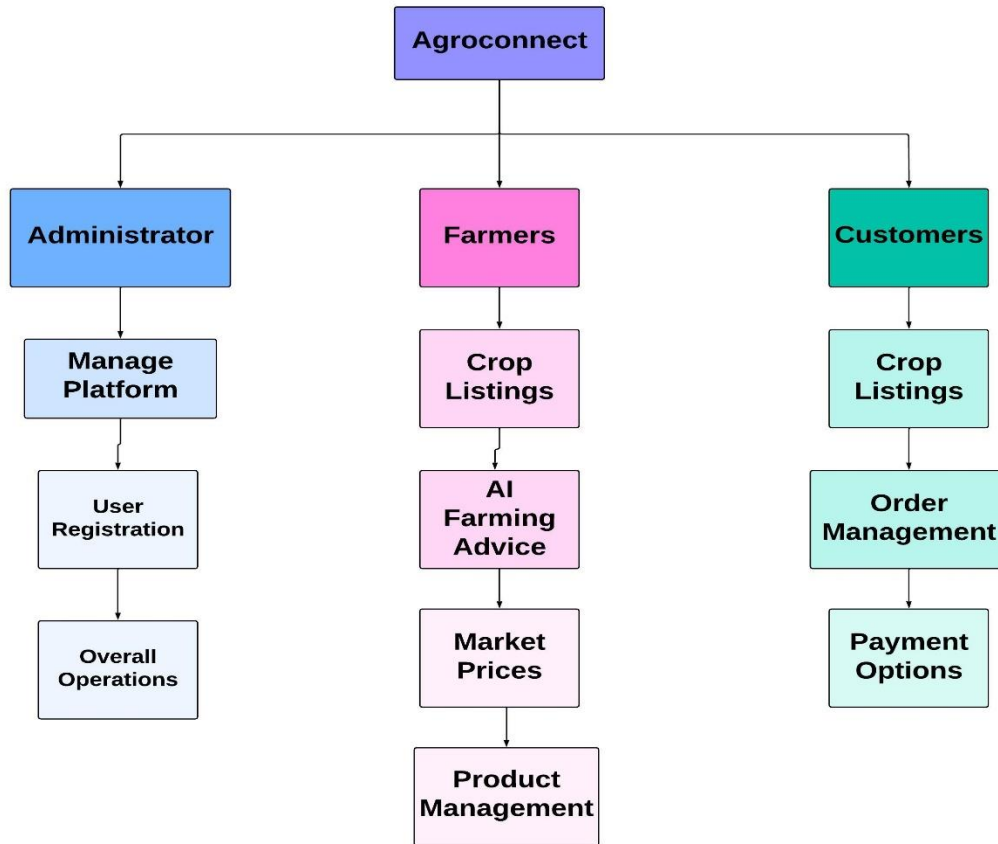


Figure 2.1.1: Organizational Chart

Scope of the System

The scope of AgroConnect encompasses features tailored for both farmers and customers. Farmers can directly list their crops for sale, access AI-driven farming advice, and view real-time market prices to make informed decisions. Customers can browse detailed crop listings and purchase products directly from farmers, ensuring fair pricing and transparency. Additional modules include secure payment integration, weather and climate data for agricultural planning, and information on government schemes and subsidies. Designed for scalability, AgroConnect can expand to accommodate more crops, users, and regions, creating a robust and comprehensive platform for modern agriculture.

Summary of Requirements: (Initial Requirements)

User Accounts:

- Farmers and customers must be able to register and log in securely.

Farmers' Portal:

- Features for crop listings, AI farming advice, market price updates, and product management.

Customers' Portal:

- Access to crop listings, order placement, and payment options.

Administrator Module:

- Manage users, products, and overall platform operations.

AI Integration:

- Provide personalized farming recommendations and resource optimization.

Payment System:

- Secure and user-friendly payment gateway integration.

Weather and Climate Module:

- Real-time weather updates and climate data for agricultural planning.

Government Schemes:

- Provide information on subsidies and support programs.

3.1.2. Identifying External Entities

Farmers:

- Primary users who list crops, access AI farming advice, and manage their products.

Customers:

- End-users who browse crop listings, place orders, and make payments.

Admin:

- Provide data on schemes, subsidies, and support programs integrated into the platform.

Weather Data Providers:

- External systems offering real-time weather and climate information.

Payment Gateways:

- Third-party services facilitating secure online transactions.

Market Data Sources:

- External databases providing up-to-date crop prices.

2.1.3. Context Level Data Flow Diagram:

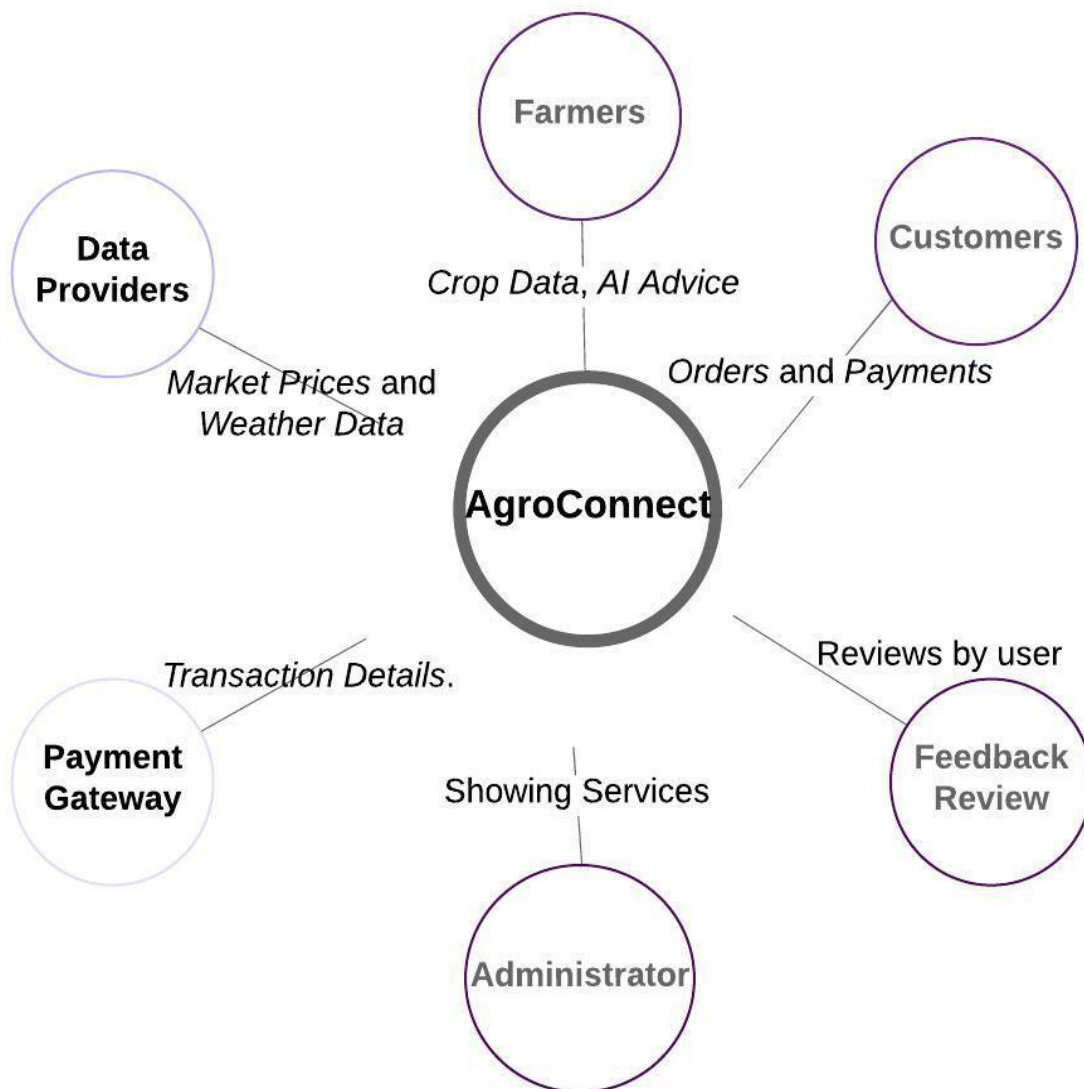


Figure 2.1.3: Context Level Data Flow Diagram

2.1.4. Capture "shall" Statements:

Table for shall statements is as follows:

Table 2.1.4: Capture "shall" Statements

Para#	Initial Requirements
1.0	User shall register to the AgroConnect platform with their email.
1.0	User shall log in to the AgroConnect platform.
1.0	User shall update their profile by entering general details.
1.0	User shall access crop listings and market prices upon login.
1.0	User shall check the availability of products near them.
1.0	User shall place orders for crops directly from farmers
1.0	User shall give feedback based on product quality and delivery.
1.0	User shall track orders and payments.
2.0	Farmer shall register to the AgroConnect platform with their email.
2.0	Farmer shall log in to the AgroConnect platform.
2.0	Farmer shall post their crop listings with prices and availability.
2.0	Farmer shall manage orders and respond to customer inquiries.
2.0	Farmer shall update their offerings based on customer feedback.
2.0	Farmer shall provide advice on farming practices through the AI bot.
3.0	The platform shall manage user profiles for both customers and farmers.
3.0	The platform shall manage farmer crop listings, product availability, and prices.
3.0	The platform shall allow users to view market prices and check the average crop prices.
3.0	The platform shall provide an AI-based farming advice system for farmers based on their inputs.
3.0	The platform shall allow farmers to track orders and manage payments.
3.0	The platform shall enable users to leave feedback and ratings on crop quality and service.

2.1.5. Allocate Requirements:

Table 2.1.5: Allocate Requirements

Para #	Initial Requirements	UC Name
1.0	User shall register to the AgroConnect platform with their email.	UC Registration
1.0	User shall log in to the AgroConnect platform.	UC Login
1.0	User shall update their profile by entering general details.	UC-Update
1.0	User shall access crop listings and market prices upon login.	UC-Listing
1.0	User shall check the availability of products near them.	UC-Avail.
1.0	User shall place orders for crops directly from farmers	UC-Order
1.0	User shall give feedback based on product quality and delivery.	UC-Feedback
1.0	User shall track orders and payments.	UC-Track
2.0	Farmer shall register to the AgroConnect platform with their email.	UC-Registration
2.0	Farmer shall log in to the AgroConnect platform.	UC-Login
2.0	Farmer shall post their crop listings with prices and availability.	UC-Listing
2.0	Farmer shall manage orders and respond to customer inquiries.	UC-Respond
2.0	Farmer shall update their offerings based on customer feedback.	UC-Update
2.0	Farmer shall provide advice on farming practices through the AI bot.	UC-Advice
3.0	The platform shall manage user profiles for both customers and farmers.	UC-Manage User
3.0	The platform shall manage farmer crop listings, product availability, and prices.	UC-Manage price
3.0	The platform shall allow users to view market prices and check the average crop prices.	UC-Allow
3.0	The platform shall provide an AI-based farming advice system for farmers based on their inputs.	UC-Advice
3.0	The platform shall allow farmers to track orders and manage payments.	UC-Track
3.0	The platform shall enable users to leave feedback and ratings on crop quality and service.	UC-Feedback

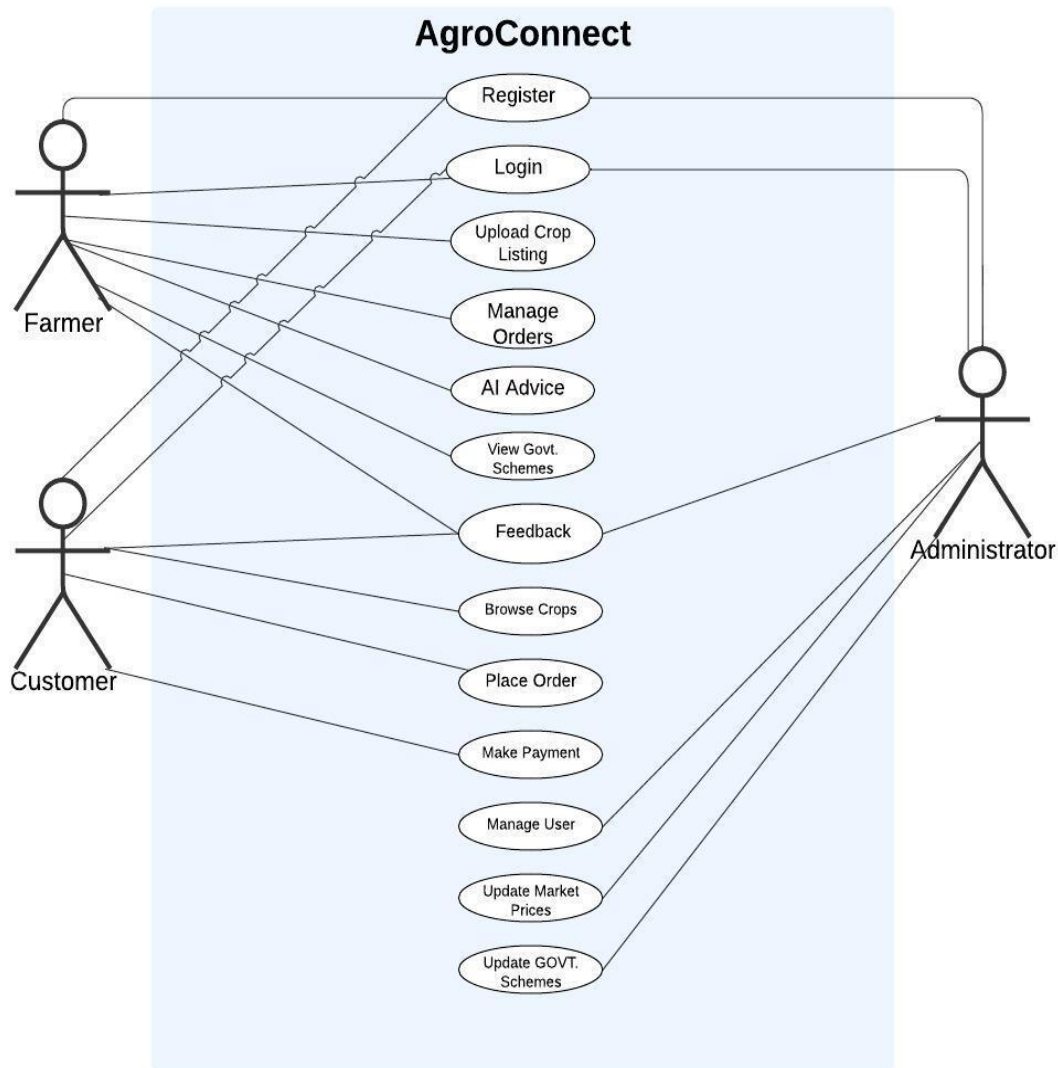
2.1.6. Prioritize Requirements:

Para#	Rank	Initial Requirements	UC Name	UC ID
1.0	Highest	User shall register to the platform with their email.	UC Register	UC-1
1.0	Highest	User shall log in to the platform.	UC Login	UC-2
1.0	Medium	User shall update their profile.	UC-Update	UC-3
1.0	Highest	User shall access crop listings.	UC-Listing	UC-4
1.0	Highest	User shall check the availability of products.	UC-Avail.	UC-5
1.0	Highest	User shall place orders directly from farmers.	UC-Order	UC-6
1.0	Highest	User shall give feedback based on product quality.	UCFeedback	UC-7
1.0	Highest	User shall track orders and payments.	UC-Track	UC-8
2.0	Highest	Farmer shall register to the platform with their email.	UC-Register	UC-1
2.0	Highest	Farmer shall log in to the platform.	UC-Login	UC-2
2.0	Highest	Farmer shall post their crop listings.	UC-Listing	UC-4
2.0	Highest	Farmer shall respond to customer inquiries.	UC-Respond	UC-9
2.0	Highest	Farmer shall update their offerings.	UC-Update offering	UC-10
2.0	Highest	Farmer shall provide advice on through the AI bot.	UC-Advice	UC-11
3.0	Highest	The platform shall manage user profiles.	UC-Manage User	UC-12
3.0	Highest	The platform shall manage farmer crop listings, product availability, and prices.	UC-Manage price	UC-13
3.0	Highest	The platform shall allow users to view market prices	UC-Allow	UC-14
3.0	Highest	The platform shall provide an AI-based farming advice system	UC-Advice	UC-11
3.0	Highest	The platform shall allow farmers to track orders.	UC-Track	UC-8
3.0	Highest	The platform shall enable users to leave feedback.	UCFeedback	UC-7

2.1.7. Requirements Trace-ability Matrix:

Para#	Rank	Initial Requirements	UC Name	Build	UC ID
1.0	Highest	User shall register to the platform with their email.	UC Register	B1	UC-1
1.0	Highest	User shall log in to the platform.	UC Login	B1	UC-2
1.0	Medium	User shall update their profile.	UC-Update	B1	UC-3
1.0	Highest	User shall access crop listings.	UC-Listing	B1	UC-4
1.0	Highest	User shall check the availability of products.	UC-Avail.	B1	UC-5
1.0	Highest	User shall place orders directly from farmers.	UC-Order	B1	UC-6
1.0	Highest	User shall give feedback based on product quality.	UCFeedback	B1	UC-7
1.0	Highest	User shall track orders and payments.	UC-Track	B1	UC-8
2.0	Highest	Farmer shall register to the platform with their email.	UC-Register	B1	UC-1
2.0	Highest	Farmer shall log in to the platform.	UC-Login	B1	UC-2
2.0	Highest	Farmer shall post their crop listings.	UC-Listing	B1	UC-4
2.0	Highest	Farmer shall respond to customer inquiries.	UC-Respond	B1	UC-9
2.0	Highest	Farmer shall update their offerings.	UC-Update offering	B1	UC-10
2.0	Highest	Farmer shall provide advice on through the AI bot.	UC-Advice	B1	UC-11
3.0	Highest	The platform shall manage user profiles.	UC-Manage User	B1	UC-12
3.0	Highest	The platform shall manage farmer crop listings, product availability, and prices.	UC-Manage price	B1	UC-13
3.0	Highest	The platform shall allow users to view market prices	UC-Allow	B1	UC-14
3.0	Highest	The platform shall provide an AI-based farming advice system	UC-Advice	B1	UC-11
3.0	Highest	The platform shall allow farmers to track orders.	UC-Track	B1	UC-8
3.0	Highest	The platform shall enable users to leave feedback.	UCFeedback	B1	UC-7

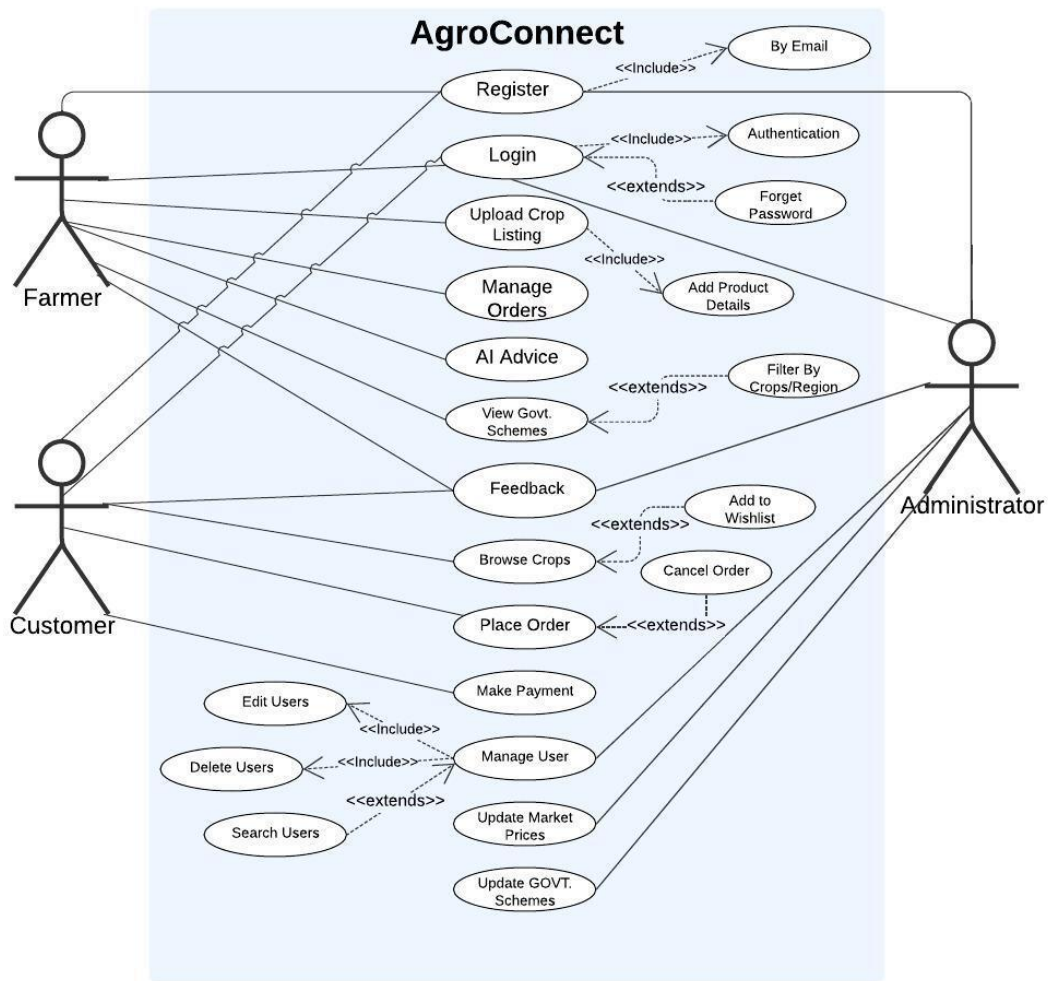
2.2.10. High Level Usecase Diagram:



2.2.11. Analysis Level Usecase Diagram:

Analysis level usecase diagram is actually the explanation of high level usecas diagram. In this diagram high level usecases are expanded in a way that exhibit how high level usecases will reach to their functionality. Two types of relationships are used in this diagram. Which are:

- Extend
- Include



2.2.12. Use case Description

Table 2.2.12.1: UC Description of Registration

Use Case Name	Register
Use Case Id.	UC-1
Actors	Admin, Customers and Farmers
Pre-Conditions	User must have a valid email address and internet connection.
Post-Conditions	User account is created, and the user can log in.

Table 2.2.12.2: UC Description of Login

Use Case Name	Login
Use Case Id.	UC-2
Actors	Admin, Customers and Farmers
Pre-Conditions	User must have a registered account.
Post-Conditions	User successfully logs into the platform.

Table 2.2.12.3: UC Description of Upload Crop Listing

Use Case Name	Crop Listing
Use Case Id.	UC-3
Actors	Farmers
Pre-Conditions	Farmer must be logged in and profile must be complete.
Post-Conditions	Crop listing is successfully posted and visible to customers.

Table 2.2.12.4: UC Description of Order Management

Use Case Name	Order Management
Use Case Id.	UC-4
Actors	Farmers
Pre-Conditions	Farmer must be logged in and have active orders.
Post-Conditions	Order statuses are updated, and customers are notified accordingly.

Table 2.2.12.5: UC Description of Feedback

Use Case Name	Feedback
Use Case Id.	UC-5
Actors	Farmers, Customers, Administrator
Pre-Conditions	User must have completed an order or interaction.
Post-Conditions	Feedback is saved and visible to relevant parties.

Table 2.2.12.6: UC Description of AI Advice

Use Case Name	AI Advice
Use Case Id.	UC-6
Actors	Farmers
Pre-Conditions	Farmer must be logged in and profile must be complete.
Post-Conditions	Advice for farming provided to farmer successfully.

Table 2.2.12.7: UC Description of Place Order

Use Case Name	Place Order
Use Case Id.	UC-7
Actors	Customers
Pre-Conditions	Customer must be logged in and crop listings must be available.
Post-Conditions	Order is placed, and both customer and farmer are notified.

Table 2.2.12.8: UC Description of Browse Crops

Use Case Name	Browse Crops
Use Case Id.	UC-8
Actors	Customers
Pre-Conditions	Customer must be logged in and profile must be complete.
Post-Conditions	Crops listings provided successfully.

Table 2.2.12.9: UC Description of Manage Users

Use Case Name	Manage Users
Use Case Id.	UC-9
Actors	Administrator
Pre-Conditions	Admin must be logged in with the appropriate permissions.
Post-Conditions	User data is updated or removed successfully.

Table 2.2.12.10: UC Description of Update Market Prices

Use Case Name	Monitor Transactions
Use Case Id.	UC-10
Actors	Administrator
Pre-Conditions	Payment data must be available in the system.
Post-Conditions	Transactions are reviewed, and any discrepancies are logged.

Table 2.2.12.11: UC Description of Government Schemes and Subsidies

Use Case Name	Government Schemes and Subsidies
Use Case Id.	UC-11
Actors	Administrator and Farmer
Pre-Conditions	Farmer and Administrator must be logged in and profile must be complete.

Post-Conditions	Govt. Schemes and subsidies updated and provided to farmers successfully.
------------------------	---

Table 2.2.12.12: UC Description of Make Payments

Use Case Name	Make Payments
Use Case Id.	UC-12
Actors	Customer
Pre-Conditions	Customer must be logged in and profile must be complete.
Post-Conditions	Payment of order made successfully.

Table 2.2.12.13: UC Description of Make Payments

Use Case Name	Market Prices
Use Case Id.	UC-13
Actors	Administrator
Pre-Conditions	Administrator must be logged in and profile must be complete.
Post-Conditions	Market Prices updated successfully.

Chapter 3: Design Document (For Object Oriented Approach)

3.1. Introduction:

Third deliverable is all about the software design. In the previous deliverable, analysis of the system is completed. So we understand the current situation of the problem domain. Now we are ready to strive for a solution for the problem domain by using object-oriented approach. Following artifacts must be included in the 3rd deliverable.

1. Domain Model
2. System Sequence Diagram
3. Sequence Diagram
4. Collaboration Diagram
5. Operation Contracts
6. Design Class Diagram
7. State Transition Diagram
8. Data Model

Now we discuss these artifacts one by one as follows:

3.2. Domain Model

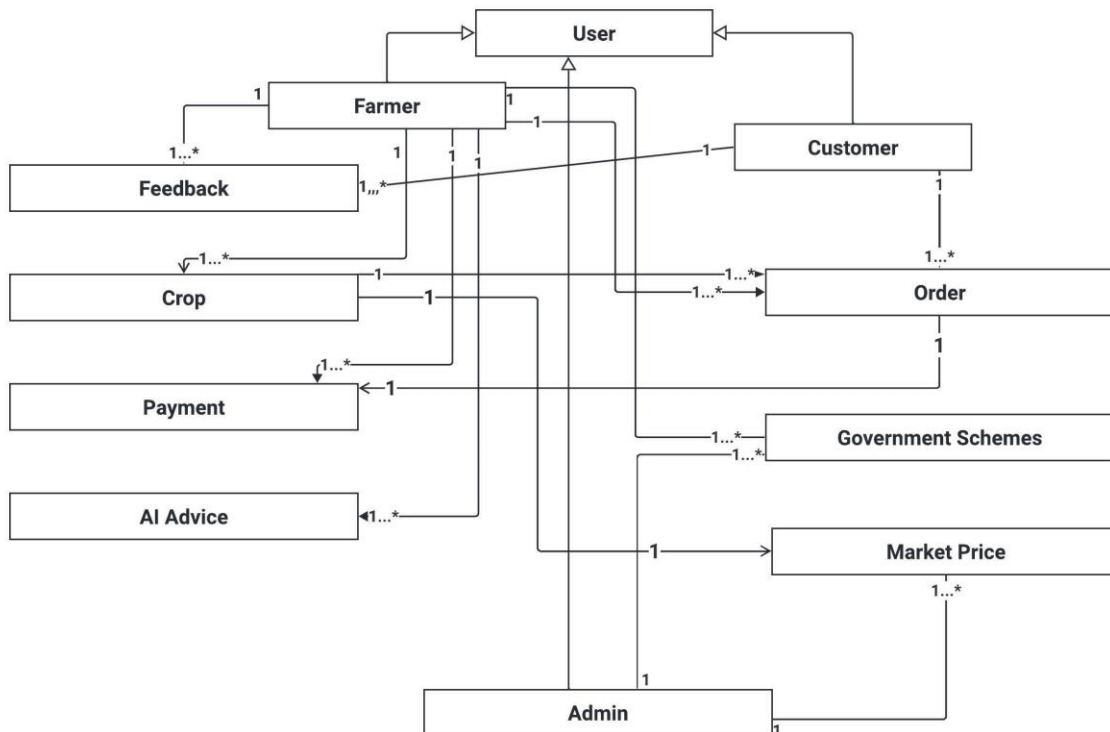
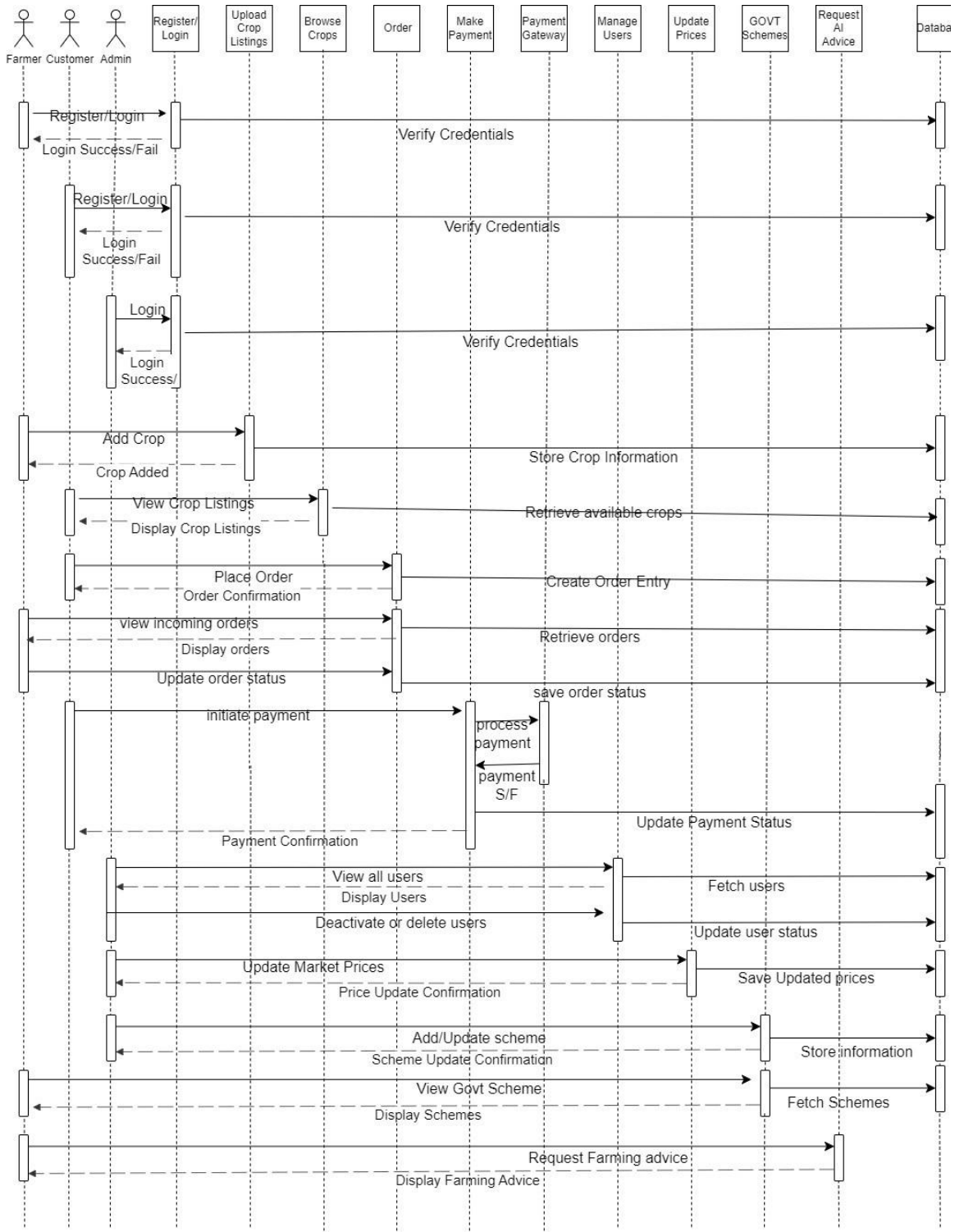


Figure 3.2 Domain Model Diagram

3.3. System Sequence Diagram



3.4. Sequence Diagram

3.4.1. Defining a Sequence diagram

A sequence diagram is made up of objects and messages. Objects are represented exactly how they have been represented in all UML diagrams—as rectangles with the underlined class name within the rectangle.

Sequence diagrams describe interactions among classes in terms of an exchange of messages over time.

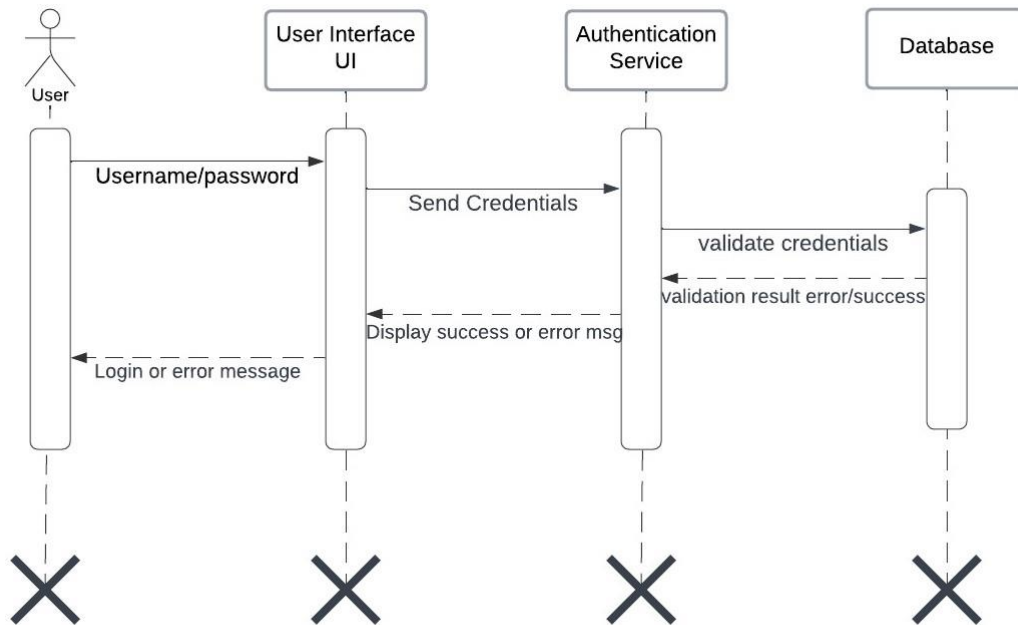


Figure 3.4.1 Login sequence Diagram

3.4.2. Basic Sequence Diagram Symbols and Notations

Class roles

Class roles describe the way an object will behave in context. Use the UML object symbol to illustrate class roles, but don't list object attributes.

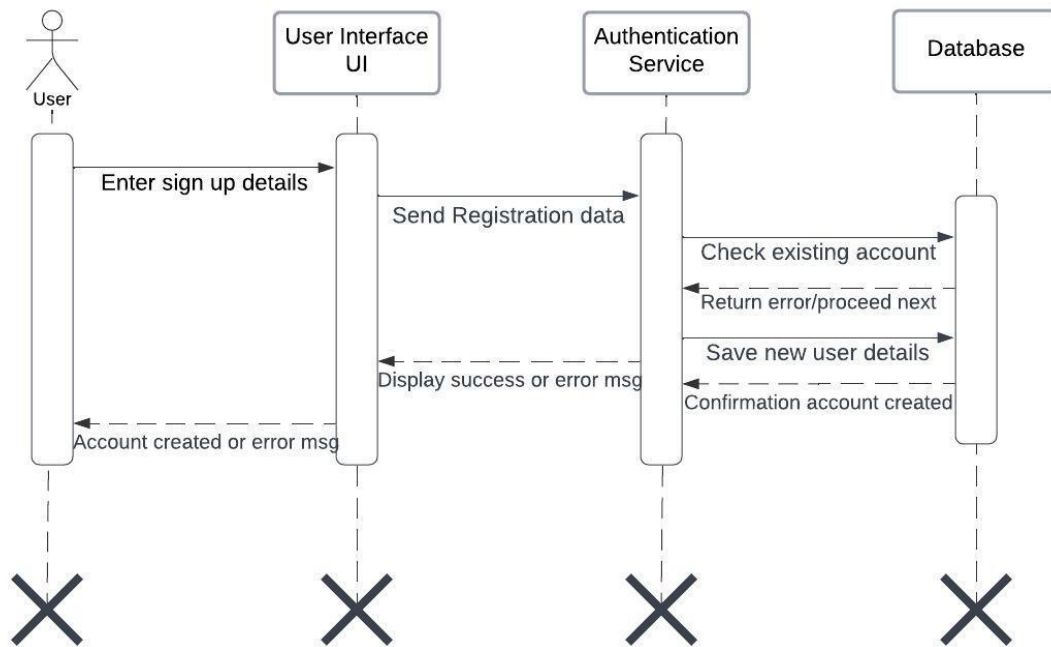


Figure 3.4.2 Sign-Up sequence Diagram

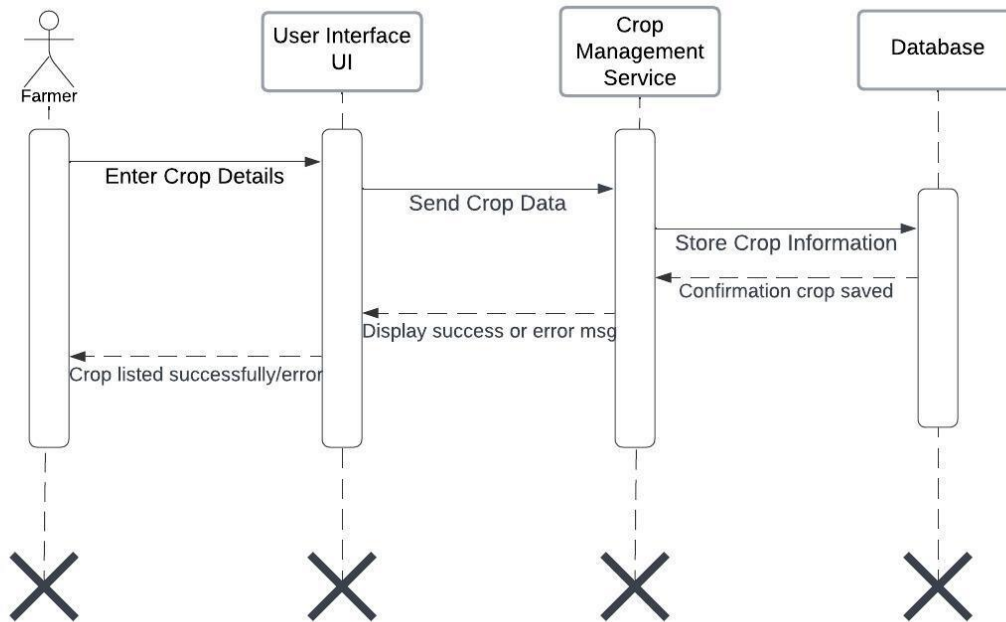


Figure 3.4.3 Crop Listing sequence Diagram

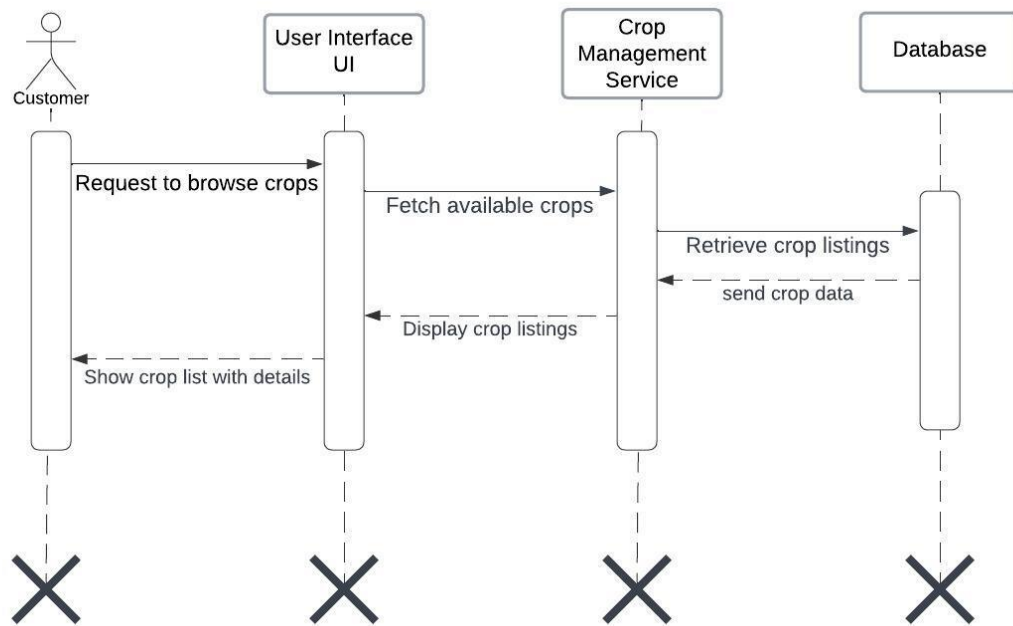


Figure 3.4.4 Browse crop sequence Diagram

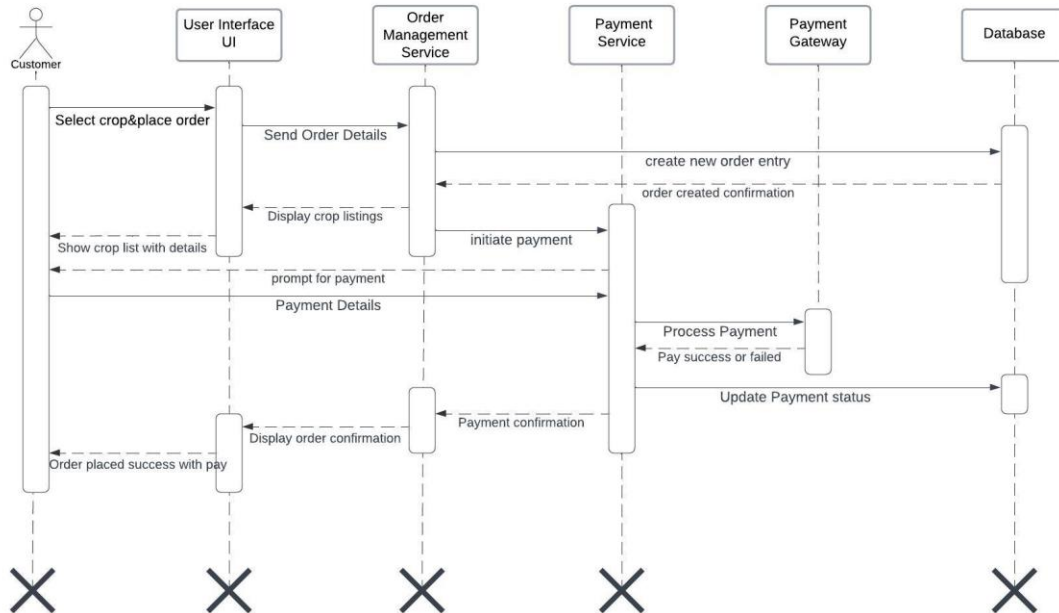


Figure 3.4.5 Place order & Payment sequence Diagram

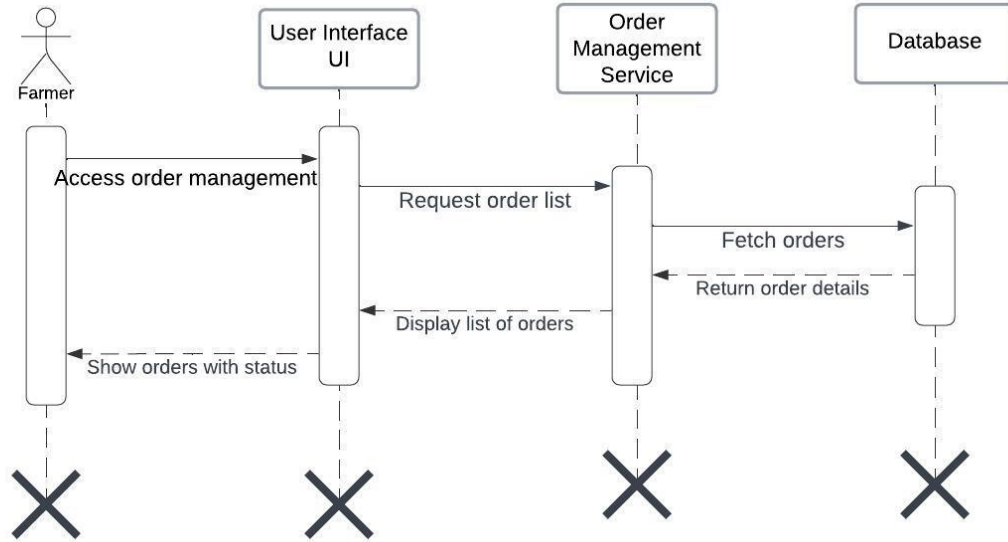


Figure 3.4.6 Manage orders sequence Diagram

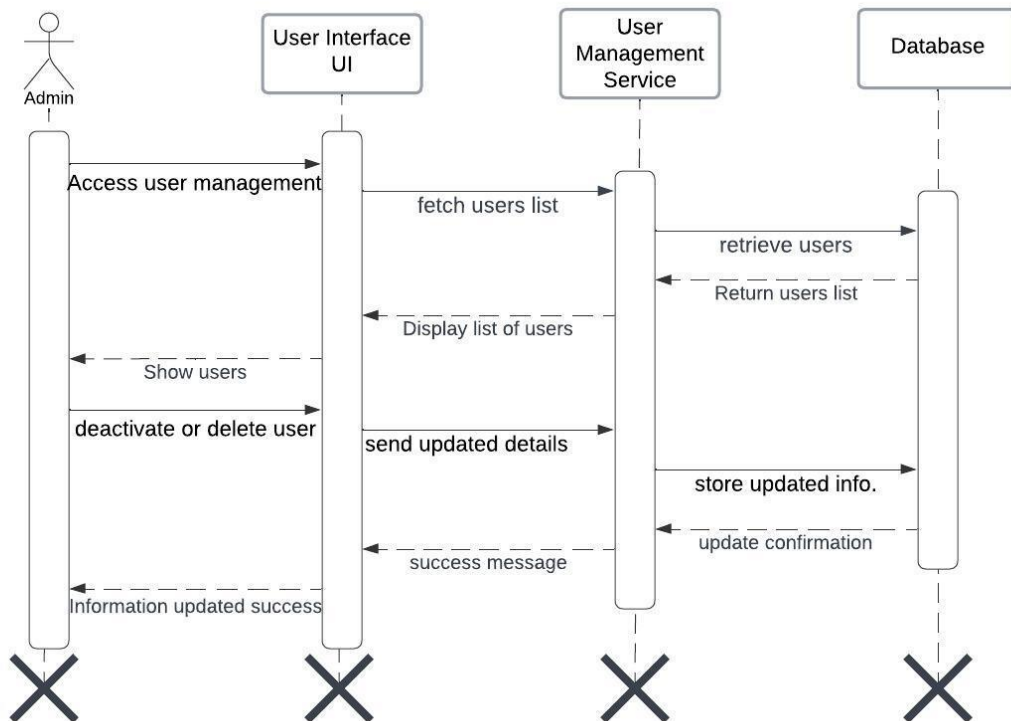


Figure 3.4.7 Manage users sequence Diagram

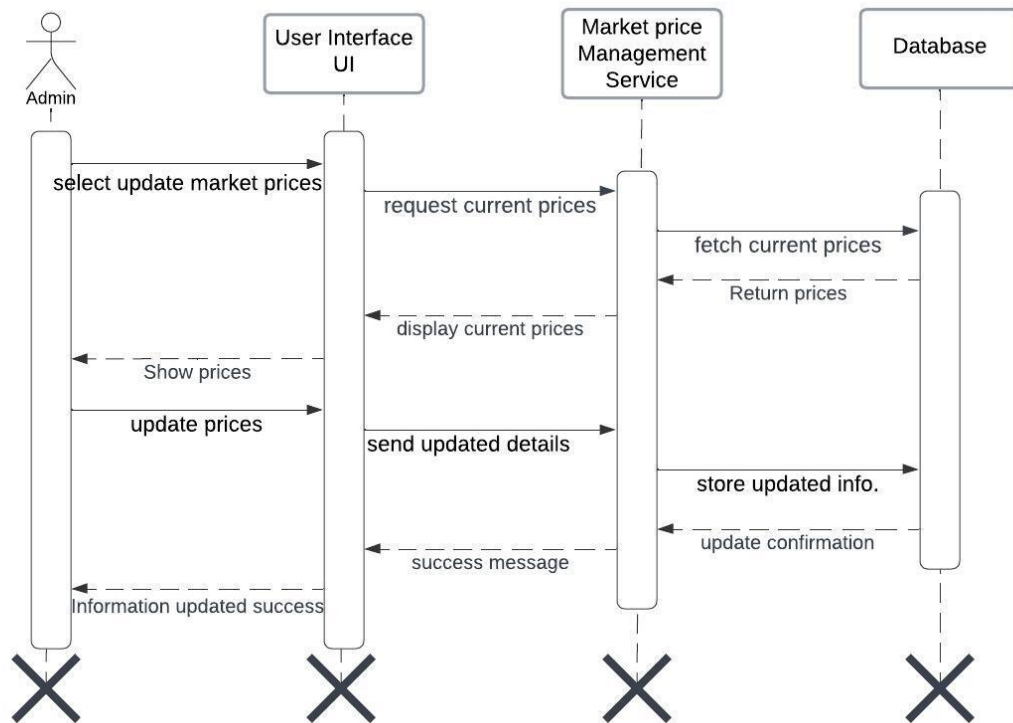


Figure 3.4.8 Update market prices sequence Diagram

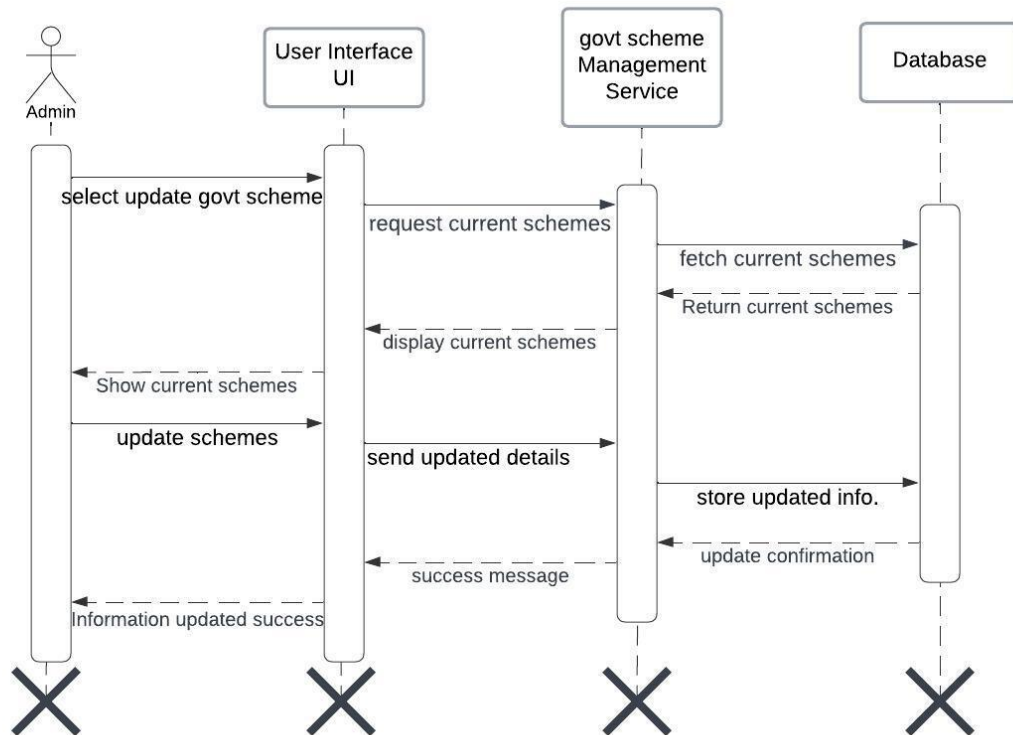
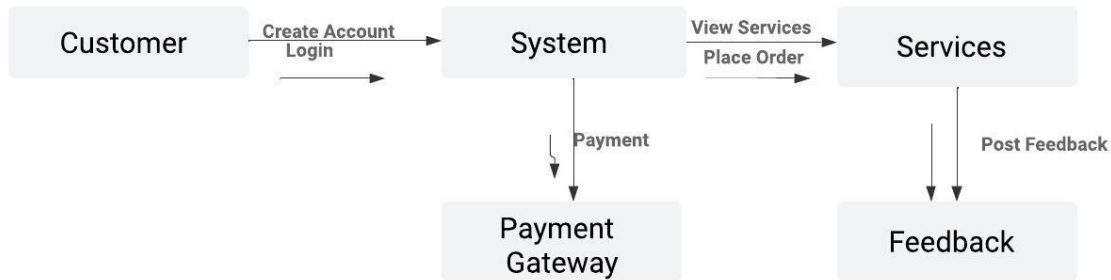


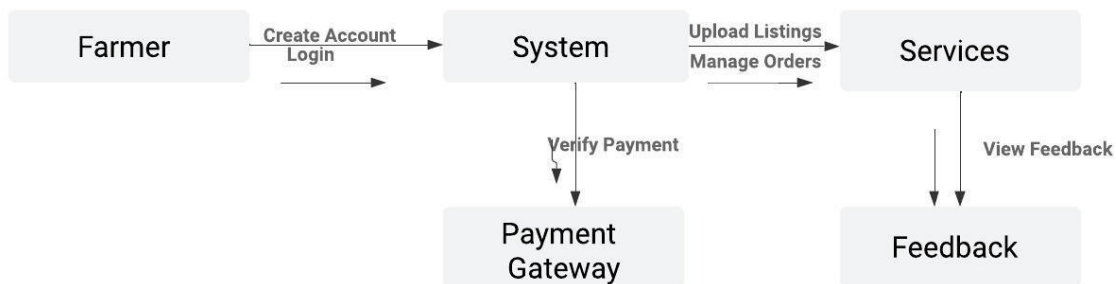
Figure 3.4.9 Update GOVT schemes sequence Diagram

3.5. Collaboration Diagram

3.5.1. Customer Collaboration diagram



3.5.2. Farmer Collaboration diagram



3.6. Operation Contracts

Table 3.6.1: Register

Name	Register ()
Responsibilities	Register user to system
Pre-Conditions	Actor should have a mobile with internet connection
Post Conditions	User can open app and successfully registered her/his account.
Exceptions	Invalid or Incomplete Details

Table 3.6.2: Crop Listings

Name	CropListings()
Responsibilities	Validate crop details
Pre-Conditions	User must be logged in to system, Crop name, price must be provided.
Post Conditions	A new crop record is created and associated with the logged-in farmer.

Exceptions	Missing or invalid crop details: Return "Invalid crop information."
-------------------	---

Table 3.6.3: Place Order

Name	PlaceOrder()
Responsibilities	Validate order details
Pre-Conditions	Customer must be logged into the system. Expected quantity must be available.
Post Conditions	A new order record is created with the status "Pending."
Exceptions	Insufficient crop quantity.

Table 3.6.4: View Market Prices

Name	ViewMarketPrices()
Responsibilities	Fetch up-to-date market price data.
Pre-Conditions	Market price data must be available in the system.
Post Conditions	The system retrieves the latest market prices and displays them to the user.
Exceptions	Market price data unavailable.

Table 3.6.5: Weather Suggestions

Name	getWeatherSuggestions ()
Responsibilities	Validate location input.
Pre-Conditions	Weather API must be functional and accessible.
Post Conditions	Weather data is retrieved from the API.
Exceptions	Invalid location.

Table 3.6.6: Fetch Schemes

Name	fetchSchemes ()
Responsibilities	Query the government portal or retrieve stored scheme data.
Pre-Conditions	The system must have access to the government portal or preloaded schemes data.
Post Conditions	Government schemes relevant to farming are retrieved and displayed to the user.
Exceptions	No relevant schemes found.

Table 3.6.7: Feedback

Name	submitFeedback ()
Responsibilities	Store feedback in the database.
Pre-Conditions	Customer must have a completed order with the crop/farmer.

Post Conditions	A new feedback entry is created and linked to the order and farmer.
Exceptions	Order not found

3.7. Design Class Diagram

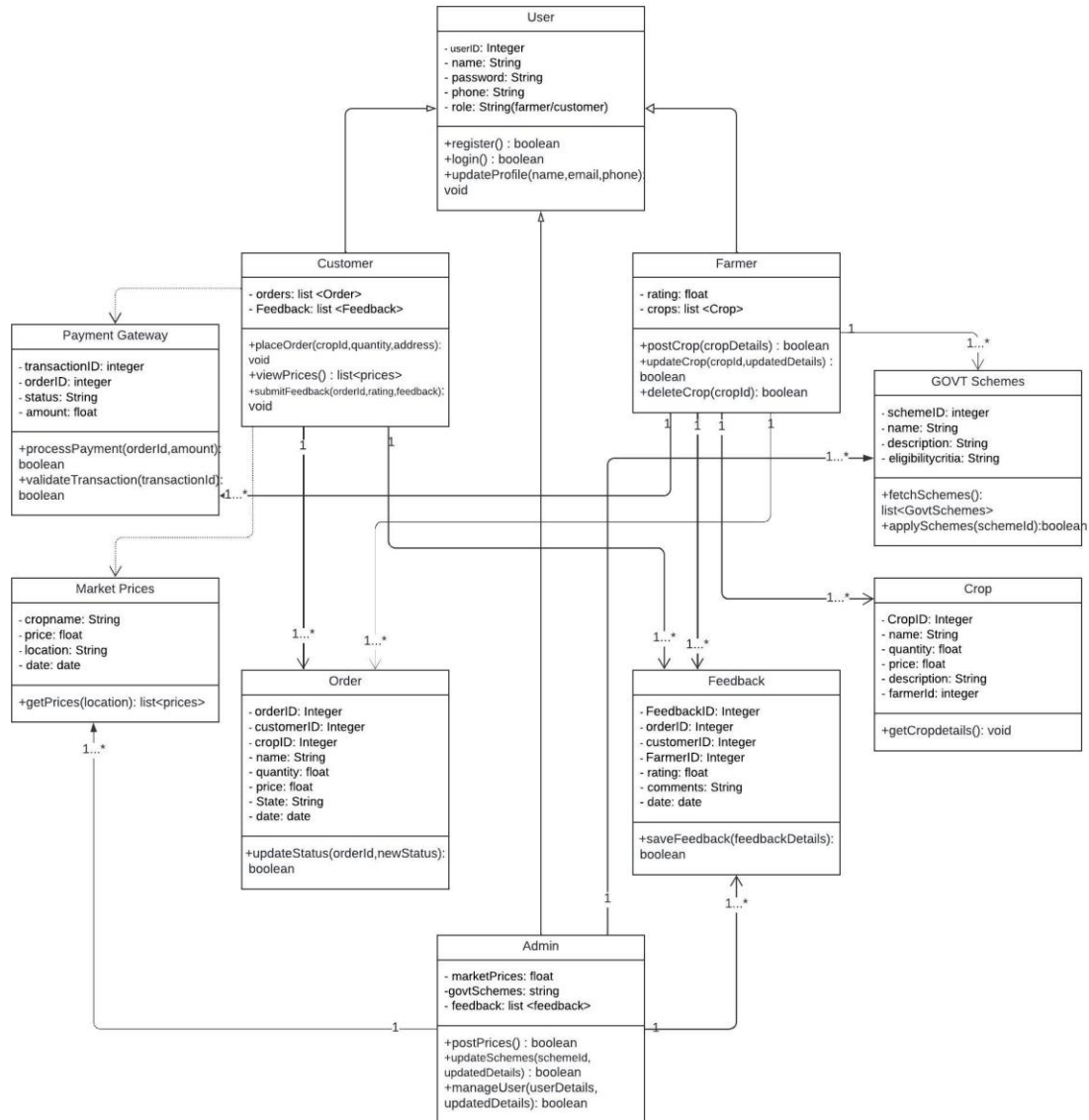


Figure 3.7 Class Diagram

3.8. State chart diagram

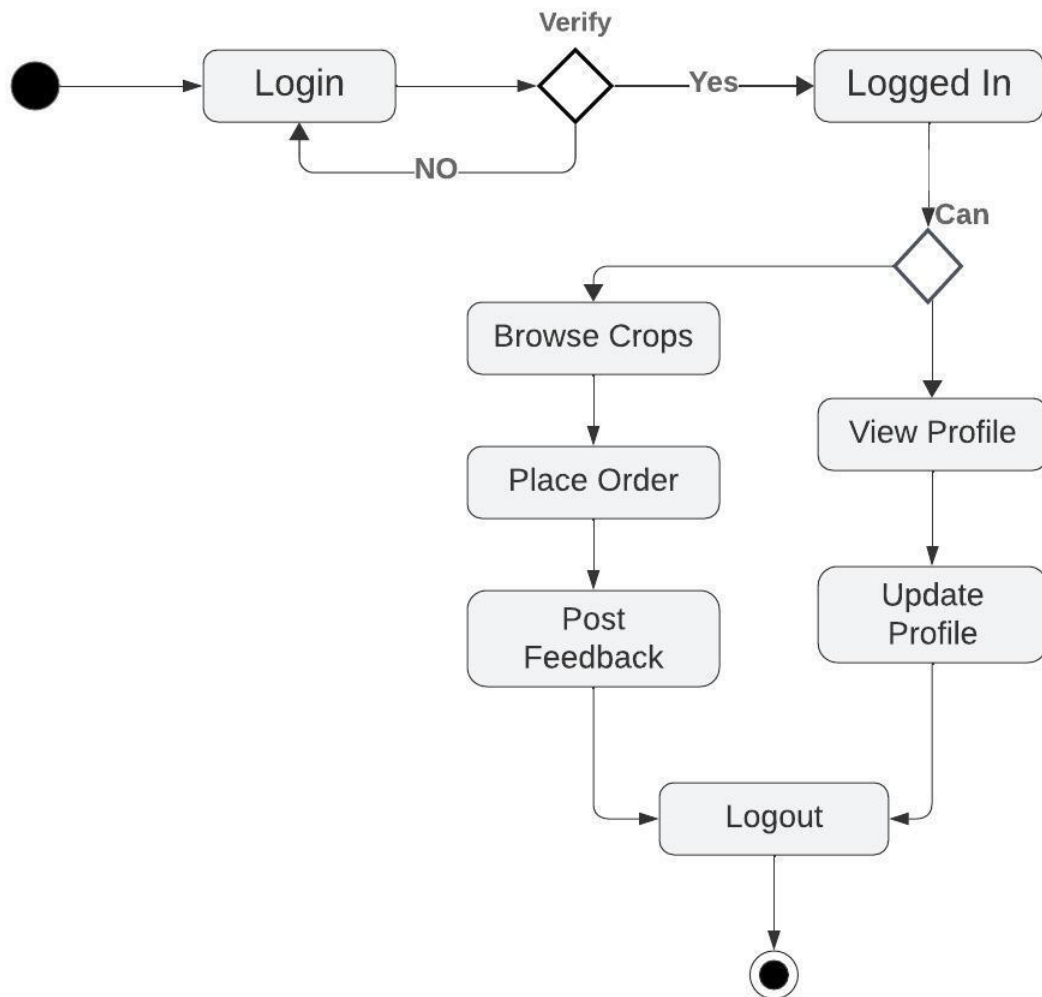


Figure 3.8.1 Customer State chart Diagram

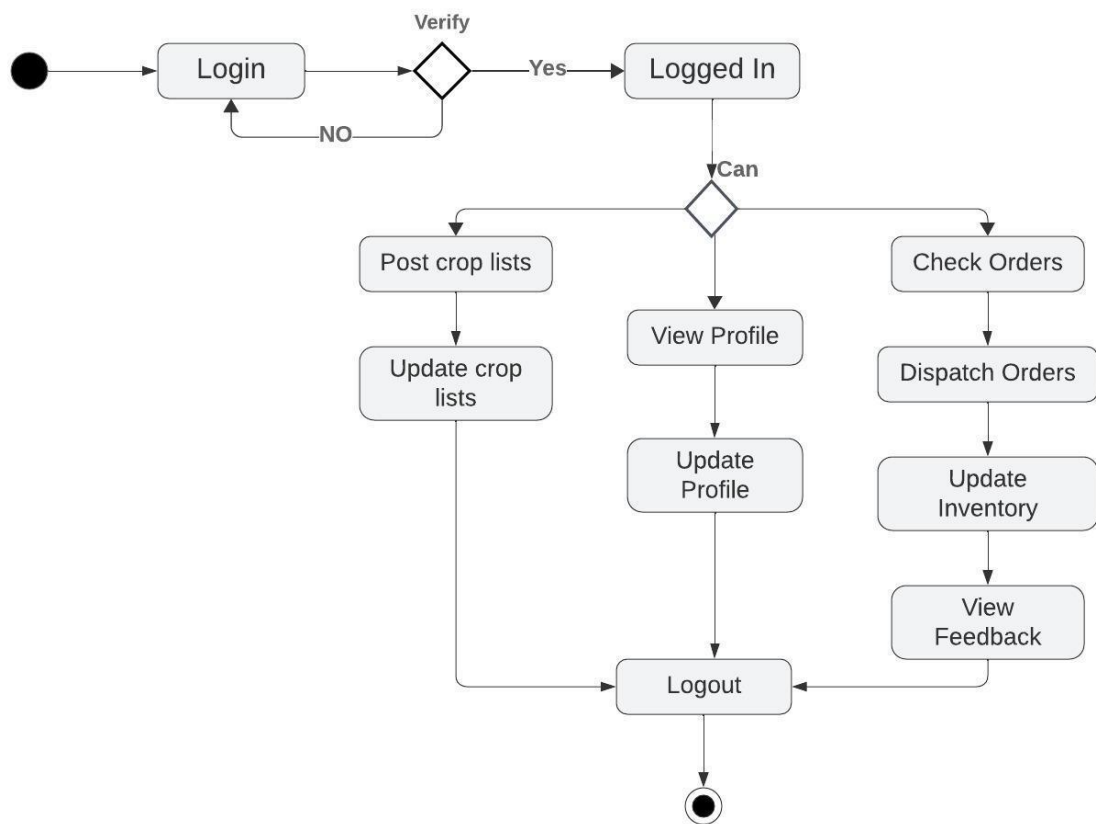


Figure 3.8.2 Farmer State chart Diagram

3.9. Data Model

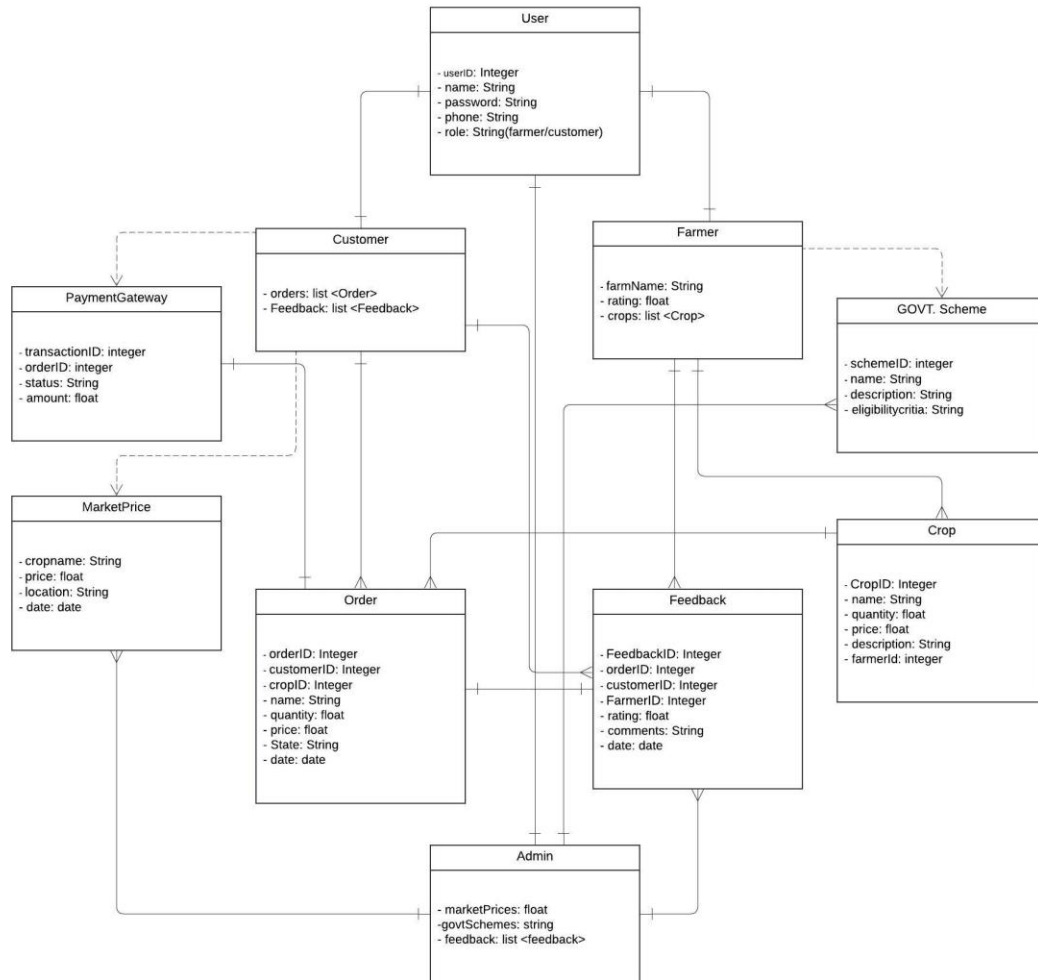


Figure 3.9.1 data model

