

UCS1617 Mini Project

AgriTech System

Date : 22-03-2021

Name : Lakshmi Priya B

Register Number : 185001083

Other member : Nivedhitha D (185001104)



Table of Contents

<u>Table of Contents</u>	<u>2</u>
<u>Problem Statement</u>	<u>5</u>
<u>Software Requirements Specification</u>	<u>6</u>
<u>Revision History</u>	<u>6</u>
<u>Introduction</u>	<u>7</u>
<u>Purpose</u>	<u>7</u>
<u>Scope</u>	<u>7</u>
<u>Existing System</u>	<u>8</u>
<u>Objectives of the system</u>	<u>10</u>
<u>Definitions, Acronyms and Abbreviations</u>	<u>12</u>
<u>References</u>	<u>13</u>
<u>Overview</u>	<u>13</u>
<u>Overall Description</u>	<u>14</u>
<u>Product Perspective</u>	<u>14</u>
<u>Product Functions</u>	<u>15</u>
<u>User characteristics</u>	<u>15</u>
<u>Constraints</u>	<u>16</u>
<u>Assumptions and dependencies</u>	<u>16</u>
<u>Specific Requirements</u>	<u>17</u>
<u>Functionality</u>	<u>17</u>
<u>Usability</u>	<u>17</u>
<u>Reliability</u>	<u>17</u>
<u>Performance</u>	<u>18</u>
<u>Supportability</u>	<u>18</u>
<u>Design Constraints</u>	<u>19</u>
<u>Online User Documentation and Help System Requirements</u>	<u>19</u>
<u>Purchased Components</u>	<u>19</u>
<u>Interfaces</u>	<u>19</u>
<u>Licensing Requirements</u>	<u>20</u>
<u>Legal, Copyright, and Other Notices</u>	<u>20</u>
<u>Use Case Model</u>	<u>21</u>
<u>Aim</u>	<u>21</u>

<u>Notations</u>	22
<u>Identification of Actors</u>	23
<u>Identification of Scenarios</u>	24
<u>Relating Use Cases</u>	25
<u>Notations Guide</u>	25
<u>User goal : Main Success Scenario</u>	26
<u>Alternate Scenario : Failure Scenario</u>	27
<u>Sub-function: Buy</u>	28
<u>Sub-function: Sell</u>	28
<u>Sub-function: Agri-info</u>	29
<u>Sub-function: Resource Manager</u>	30
<u>Sub-function: Discussion Forum</u>	31
<u>Use Case Diagrams</u>	32
<u>User goal : Main Success</u>	32
<u>Alternate Scenario : Failure Scenario</u>	33
<u>Sub-function : Buy</u>	34
<u>Sub-function : Sell</u>	35
<u>Sub-function : Agri-info</u>	36
<u>Sub-function : Resource Manager</u>	37
<u>Sub-function : Discussion Forum</u>	38
<u>Fully Dressed Use Case Description</u>	39
<u>Main Success</u>	39
<u>Failure</u>	41
<u>Buy</u>	43
<u>Sell</u>	45
<u>Agri-info</u>	46
<u>Resource Manager</u>	48
<u>Discussion Forum</u>	49
<u>Documentation</u>	50
<u>Domain Model and Class Diagram</u>	51
<u>Aim</u>	51
<u>UML Notations for Domain Model Diagram</u>	51
<u>UML Notations for Class Diagram</u>	53
<u>Identification of Classes</u>	55
<u>Identification of Noun Phrases</u>	55
<u>Conceptual Class Category List</u>	56
<u>Identification of Associations</u>	57

<u>Association Category List</u>	57
<u>Definition of Associations and their notations</u>	58
<u>Multiplicity based associations</u>	59
<u>Domain Model Diagram</u>	60
<u>Class Diagram</u>	61
<u>Documentation</u>	62
<u>Sequence Diagram</u>	63
<u>Aim</u>	63
<u>UML Notations for Sequence Diagram</u>	63
<u>Sequence Diagrams</u>	64
<u>Documentation</u>	71
<u>State Machine and Activity Diagram</u>	72
<u>Aim</u>	72
<u>UML Notations for State Machine</u>	72
<u>UML Notations for Activity Diagram</u>	73
<u>Identification of states</u>	74
<u>State Machine Diagram</u>	75
<u>Activity Diagram</u>	76
<u>Documentation</u>	77
<u>Package Diagram</u>	78
<u>Aim</u>	78
<u>UML Notations for Package Diagram</u>	78
<u>Package Diagram</u>	82
<u>Documentation</u>	83
<u>Component Diagram and Deployment Diagram</u>	84
<u>Aim</u>	84
<u>UML Notations for Component Diagram</u>	85
<u>List of components identified from class and use-case diagram</u>	86
<u>Component Diagram</u>	87
<u>UML Notations for Deployment Diagram</u>	88
<u>List of nodes identified from Component diagram</u>	88
<u>Deployment Diagram</u>	89
<u>Documentation</u>	90

Problem Statement

Nearly **58%** of the Indian workforce is engaged in **agriculture** and yet India is not self-reliant in various essential agriculture commodities and depends heavily on imports. Furthermore, the share of Indians working in agriculture is declining even though it is the **backbone of our economy**. The bill on the Agriculture market seeks to allow farmers to sell their produce outside 'mandis' to customers of their choice. Farmers will get better prices through competition and cost-cutting on transportation. Thus, the task of achieving **self-reliance** and self-sustainability in agriculture entails the collaboration with **technology to manage everything a farmer may need from seed-to-sale** and much more.

The **Agritech system** should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. It should assist in **crop selection, track finances, manage inventory, market produce and sell online**, on a **user-friendly platform** available on any device from anywhere. Furthermore, a **collaboration forum** amongst farmers will facilitate a crowd-sourced knowledge base. The Agritech system should eliminate the need for middlemen, increasing farmer profits.

The system must enable marketing and selling directly to customers online. This **Smart-Selling-Anywhere-Anytime(SSAA) feature** automates customer communications, manages pickup options and immensely simplifies revenue and financial reporting. The system uses **hotspot(short ranges)** and **GPS(long ranges)** for scoping customers or other farmers in the vicinity. The information accumulated in the system can be used to normalize the availability of produce in the geographical area which in turn **eradicates the supply-demand problem**.

The **resource management feature** for agriculture will allow efficient and effective data collection and tracking of customer and supplier lifecycles to manage financials, inventory and crops. By integrating **crop management, financials and inventory management** together, the farmer is in control of his entire workflow.

The Agritech system uses easily available and cheap hotspots to identify supply and demand points by mapping requirements stored in a database. It does not incur much costs as no special hardware is required apart from a computing device like a **phone, laptop, desktop or tablet** with a **hotspot, stable internet connection and GPS**. The software is one secure, user-friendly and cross-form, ensuring that the farmer is **self-reliant** and in **control of his entire workflow from seed-to-sale**.

Software Requirements Specification

Agritech System	Version: 1.0
Software Requirements Specification	Date: February 1, 2021

Revision History

Date	Version	Description	Author
01/01/2021	1.0	Software Requirements Specification Document Initial Release	Lakshmi Priya B Nivedhitha D

1. Introduction

1.1. Purpose

The purpose of Software Requirements Specification (SRS) document is to describe the external behavior of the Agritech System. Requirements Specification defines and describes the operations, interfaces, performance, and quality assurance requirements of the Agritech System. The document also describes the non-functional requirements such as the user interfaces. It also describes the design constraints that are to be considered when the system is to be designed, and other factors necessary to provide a complete and comprehensive description of the requirements for the software. The SRS captures the complete software requirements for the system, or a portion of the system. Requirements described in this document are derived from the TNAU Agritech Portal.

1.2. Scope

The SRS captures all the requirements in a single document. The Agritech System that is to be developed provides exceptional features such as **Crop Selection Assistance, Smart-Selling-Anytime-Anywhere (SSAA), Resource Management, Accounting, Discussion Forum** and much more. The Agritech portal is supposed to have the following features:

- The product provides the members with an online portal that is up and running all day to manage everything a farmer may need from seed-to-sale on any device with a stable internet connection
- The system provides a log in facility for three categories of members including farmers, customers and administrators
- The system provides the members with the option to log into their account to:
 - get crop assistance, planning and tracking
 - perform produce management
 - market and sell produce online using SSAA
 - utilize resource management for crop, inventory and finance management
 - utilize the discussion forum to interact with other members

-
- The system provides accurate and reliable details of various crops to aid farmers during crop selection to simplify crop planning and seed orders
 - The system automates yield estimates, easily identifies trends and potential problems
 - The system serves as a digital store for optimized planning, tracking and managing crops, equipment, customers, orders, and finances all in one place so that information is never lost
 - The system supports built-in accounting, automates profit and loss statements and helps gain new insights into farmer operations to improve decision making through personalization
 - The system is equipped with SSAA feature which enables:
 - customers to buy the products directly from the farmers at a reasonable price
 - farmers to improve sales and marketing by simplifying order tracking, handling customer records
 - The system has a collaboration forum that serves as a communication channel between the members:
 - farmers can publicly post a query which can be answered by other farmers or advisors publicly
 - customers can post their queries or product requirements publicly which can be answered by any farmer to the customer publicly or privately

The features described here meet the needs of all the users. The success criteria for the system is based on the level up to which the features described in this document are implemented in the system.

1.3. Existing System

With emergence of new and cost effective solutions in the **Information and Communications Technology (ICT)** field, traditional methods for information access and service delivery are either becoming obsolete or it has become necessary to augment these methods with the new ICT tools. As the ICT is spreading its wings at the grassroots level, more and more population is becoming aware of the benefits of ICT. Their expectations about the service delivery from the Government are increasing. They expect sophistication, timeliness and easy accessibility (anytime, anywhere) in service delivery. Timely access to information and service delivery is critical and of utmost importance in this sector in view of time bound farm activities involved in all stages of the crop cycle. To fulfill the expectations of the farm sector for provision of timely access of information and services, many e-Governance initiatives have been started in past few years, by Government as well as by Public and Private sectors. Under these initiatives, large number websites providing information and services in the agricultural sector have been launched. The Ministry of Agriculture itself has a number of websites for different divisions, directorates and projects. There have been many websites and portals published by Central Government Departments (Agriculture, Irrigation, Fertilisers, Cooperation, **ICAR**, **DADF**, Planning Commission, Commerce, Agencies , Public Sector Banks, **NABARD**, etc), State Government Departments (Sub-sectors), State level Agencies, Commodity wise portals, **NGOs**, **CMIE**, etc.

However these websites do not share web services among them and hence contents are static, non-consistent, non-integrity etc. Many times farmers and other stakeholders in the agricultural sector have to visit multiple websites to trace the desired piece of information or to avail a single service. Different websites have different look and feel, presentation style, structure, and color schema as these follow different technology standards, design lay-outs and navigation architecture etc. It results in a lot of inconvenience to the user and requires a lot of learning on their part to access the information and services. On the other hand it is a challenging job for the owners of these websites also to keep all the information updated and in sync at multiple websites all the time. This results in duplicate efforts, outdated content, multiple sources of information, mismatch of the information finally confusing the service consumer.

To make all the Agricultural information and services accessible in an easy and convenient manner, a need has been felt to establish a one-stop source for all government information and services in the Agricultural Sector. It will save farmers and other stakeholders from the hassles of searching the desired information across a large number of websites, with diverse design and navigation patterns. Providing relevant information and required services to all

the important stakeholders in this sector especially to the farming communities is the major objective of this Agritech System. This system also aims to overcome various shortcomings of **TNAU Agritech portal** which includes lack of SSAA features, product management, collaboration forum and much more.

1.4. Objectives of the system

The major objectives of Agritech System are:

- To provide a **one-stop, single-window solution** to all stakeholders, especially, farmers for accessing information and services in the agricultural sector eliminating the need to remember multiple URLs, to navigate multiple websites or applications
- To provide **timely, reliable and accurate** agricultural expert advice
- To act as a single gateway for the delivery of valuable information and services in agricultural sector from various channels
- To deliver information regarding agricultural related government services
- To complement and enhance service delivery channels which may already exist at the central government level
- To build a comprehensive **knowledge management system for Indian agriculture** by providing platform to collaborate and share knowledge
- To act as **bridge between farmers and the government bodies at the highest decision-making** level by streamlining and simplifying their interactions with government entities reducing service cycle time and providing enriching experience
- To provide mechanism for **grievance redressal and management** in agricultural sector
- To provide **advisory services to the farmers during each stage of the entire crop cycle** while on the other hand to provide the platform for the domain experts to receive and address specific queries of the farmers
- To build a **National Farmers Database** to understand the information requirements of the farmers and to provide relevant services through various delivery channels
- To provide secure, cross-platform, single window delivery of government services to agri-business industry comprising of dealers, wholesalers, retailers of the agricultural inputs and exporters, importers and traders of the agricultural products
- To streamline and simplify their interactions with government entities reducing service cycle time and providing enriching experience
- To provide consistent and uniform experience to users in terms of presentation, standards, design, layouts and navigation architecture

1.5. Salient Features

The focus of the Agritech system are the farmers and as most farmers are not experienced Internet users they find it very difficult to find the relevant information. So, the services provided by the system should be easily understandable by them. As such, the following is a list of proposed features for the Agritech System:

- Single window access of information and services in Agricultural Sector in India for farmers and other stakeholders
- Use of icons / pictures / images and graphical interfaces to represent links and information for quicker understanding of the farmers
- Consistent and Easy to Use interface with consistent design of the web pages for common look and feel. Design in a way so that the desired document can be traced in minimum number of navigations.
- Easy to navigate, search and browse.
- Adaption of the look highlighting contents of importance as per the location of the user, crop and current stage of crop life cycle as per season.
- Interface for the updation of information and service delivery.
- Well designed home page conveying theme and purpose with aesthetic and ergonomic design.
- Single Sign-on to access all information and services on the portal.
- Digital Dashboards to display key information on a single screen and allow an overall idea of the current agricultural scenario in the country/state.
- Discussion Boards to allow all kinds of users like farmers, experts, government officers, Agricultural University faculty/scientists, Students, Research Scholars etc. to interact with each other.
- Comprehensive Content Management System to provide accurate and up-to-date content with a farmer-orientation. Role based access for designated Content Management officers of the government.
- Feedback and Analysis mechanism for constant improvement and enrichment of the portal for user satisfaction.

1.6. Definitions, Acronyms and Abbreviations

- | | | |
|---------|--------|---|
| 1.6.1. | SRS | : Software Requirements Specification |
| 1.6.2. | TNAU | : Tamil Nadu Agricultural University |
| 1.6.3. | NGO | : Non-governmental Organization |
| 1.6.4. | ICAR | : Indian Council of Agricultural Research |
| 1.6.5. | CMIE | : Centre For Monitoring Indian Economy |
| 1.6.6. | NABARD | : National Bank for Agriculture and Rural Development |
| 1.6.7. | DADF | : Department of Animal Husbandry and Dairying |
| 1.6.8. | SSAA | : Smart Selling - Anytime, Anywhere |
| 1.6.9. | ICT | : Information and Communications Technology |
| 1.6.10. | MTBF | : Mean Time Between Failures |
| 1.6.11. | MTTR | : Mean Time To Repair |
| 1.6.12. | LAN | : Local Area Network |
| 1.6.13. | MDN | : Mozilla Developer Network |
| 1.6.14. | IDE | : Integrated Development Environment |
| 1.6.15. | HTML | : Hypertext Markup Language |
| 1.6.16. | CSS | : Cascading Style Sheets |
| 1.6.17. | TCP/IP | : Transmission Control Protocol/Internet Protocol |

Provided wherever necessary in the document.

1.7. References

- 1.7.1. Central Agricultural Portal Software Requirement Specifications
- 1.7.2. Providing Information on Crops, Farm Machinery, Training & Good Agricultural Practices (Service 3) Software Requirement Specifications
- 1.7.3. TNAU Agritech Portal

1.8. Overview

The SRS will provide a detailed description of the Agritech System. This document will provide the outline of the requirements, overview of the characteristics and constraints of the system:

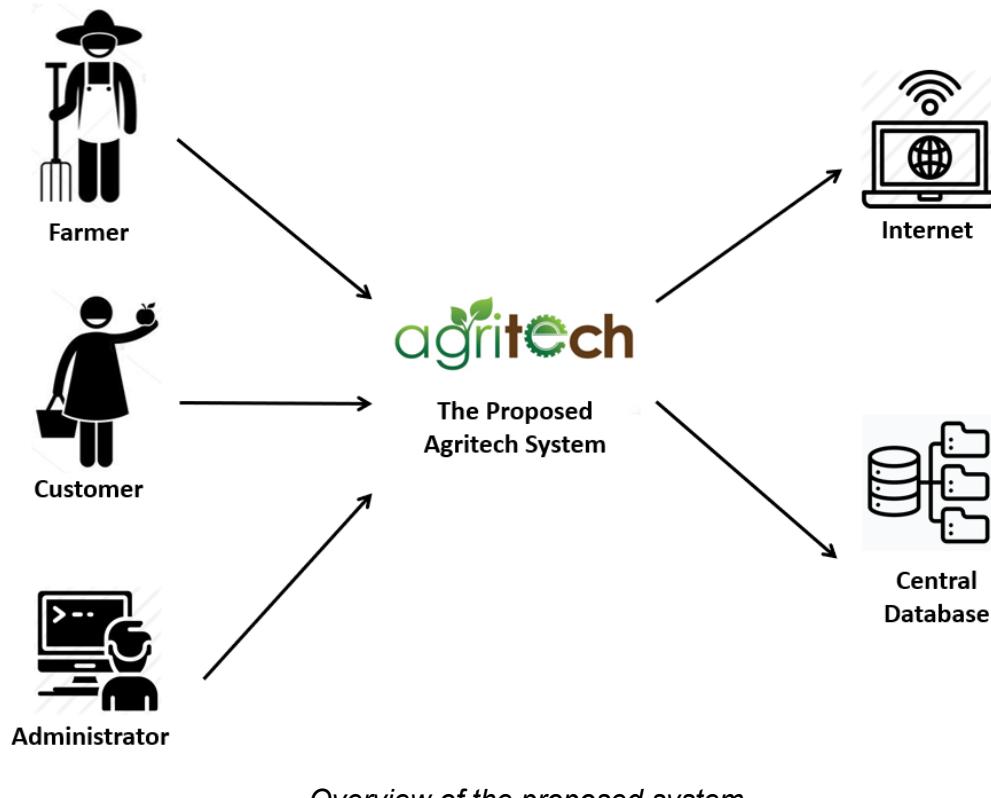
- 1.8.1. **Section 2:** This section of the SRS will provide the general factors that affect the product and its requirements. It provides the background for those requirements. The items such as product perspective, product function, user characteristics, constraints, assumptions and dependencies and requirements subsets are described in this section.
- 1.8.2. **Section 3:** This section of SRS contains all the software requirements mentioned in section 2 in detail sufficient enough to enable designers to design the system to satisfy the requirements and testers to test if the system satisfies those requirements.

2. Overall Description

2.1. Product Perspective

The Agritech System is a product to be used by all the important stakeholders in the agriculture sector to ease the entire workflow from seed-to-sale and by the consumers to acquire high quality farm produce at reasonably low prices. The Agritech System to be developed equally benefits both farmers and consumers. The system provides crop information to farmers and helps them decide on the crops to be sown for the particular season and particular soil type. The administrator must keep the crop information up-to-date all the time so that the farmers get the timely, relevant and accurate information.

The complete overview of the system is shown in the diagram below:



The product to be developed has interactions with the users: Farmer, Customer and Administrator. The product has to interact with other systems like the Internet and Central database.

2.2. Product Functions

The Agritech System provides online, real-time guidance that aids crop selection, efficient resource management and SSAA. The product functions are more or less the same as described in the product perspective. The functions of the system include the system providing different types of services based on the type of users [Farmer/Consumer/Administrator]:

- The farmers will always be provided with the up-to-date information regarding various crops and farm practices from the database which can help them extensively in crop selection
- The farmers can utilize the system for crop, inventory and finance management
- Provisions for the farmers to sell their produce to gain better margins, if all the other required rules hold good
- Provisions for the consumers to buy produce at reasonable rates, if all the other required rules hold good
- The members are given the provision to check their account information and change the details any time in the given valid period
- The members are able to post and view messages in the public discussion forum/advice section
- The administrator is provided with an interface of tools to moderate the system and preserve the consistency of the system by keeping it updated in timely manner
- The farmer is provided with an interface of tools to ease his entire workflow from seed-to-sale
- The customer is provided with an interface of tools to buy fresh produce at reasonable prices within specified geographical limit

2.3. User characteristics

The users of the system are farmers, customers and the administrators who maintain the system. The farmers and customers are assumed to have basic knowledge of the computers and Internet browsing. The administrators of the system must have knowledge of the internals of the system and should be able to rectify the problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The user interface, user manual, online help and the guide to maintain the system must be sufficient to educate members on how to use the system without much difficulty and troubleshoot.

2.4. Constraints

- The user of the system must access the Agritech System from devices such as mobile, laptop, tablet or desktop that has Internet browsing capabilities and a stable internet connection
- The information of all the farmers, customers and other records must be stored in a database that is accessible by the Agritech System
- The Agritech System server must be live throughout ensuring high reliability and fault tolerance
- The users must provide their login credentials, that is, a valid username-password pair to enter into the Agritech System
- The Agritech System demands extensive use of icons, pictures, images and graphical interfaces to represent links and information for quicker understanding of the farmers

2.5. Assumptions and dependencies

- The users have internet connectivity
- The users have sufficient knowledge on how to use mobile, laptop, tablets or desktop
- The Agritech system should have Internet connection and Internet server capabilities
- The system can access the created central database
- Content generation which helps in crop selection will be updated by the system administrators
- Updation by each farmer on the crops available with them for sale will be done timely
- To operate as a team, every stakeholder at Centre, State, District, Block and other entities such as ICAR Institutes, Agricultural Universities and partners will need to accept the responsibility for information management
- Commitment from information resources will be required

3. Specific Requirements

This section describes in detail all the functional and non-functional requirements:

3.1. Functionality

3.1.1. Logon capabilities

The system shall provide the users with logon capabilities.

3.1.2. Device compatibility

The system is also supported on mobiles, tablets, laptops and desktops.

3.1.3. Tooltip and interactive navigation

The system provides extensive visual cues to guide new users.

3.2. Usability

- The system shall allow the users to access the system from the Internet using HTML or its derivative technologies and uses a web browser as an interface (WebApp)
- Since all users are familiar with the general usage of browsers, no specific training is required
- The system is user friendly and self-explanatory

3.3. Reliability

The system has to be extremely reliable due to the importance and sensitivity of data stored in the database. The damage caused by incorrect or incomplete data will be devastating making reliability indispensable. Updation of information, especially the crop availability details, should happen in real time.

3.3.1. Availability

The system is available 100% for the user and is used 24 hours a day and on all days of the year. The system shall be operational always.

3.3.2. Mean Time Between Failures (MTBF)

The system will be developed in such a way that it may fail once in a year.

3.3.3. Mean Time To Repair (MTTR)

Even if the system fails, the system will be recovered back up within an hour or less.

3.3.4. Accuracy

The accuracy of the system is limited by the accuracy of the speed at which the farmers update the details regarding available crops for sale and the administrators update content of the system.

3.3.5. Access reliability

The system shall provide 100% access reliability.

3.4. Performance

3.4.1. *Response Time*

The information page should be able to be downloaded quickly and crisp details must be updated in real time at regular intervals. The information is refreshed every two minutes. The access time for a mobile device should be less than a minute. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs.

3.4.2. *Administrator/Farmer Response*

The system shall take as less time as possible to provide service to the administrator or the farmer.

3.4.3. *Throughput*

The number of transactions is directly dependent on the number of users, who may be the administrators, farmers, and also the customers who use the portal for buying produce.

3.4.4. *Capacity*

The system should at least handle 250 users at a time.

3.4.5. *Resource Utilization*

The resources are modified according to the user requirements and also according to the requests of the users, if possible.

3.5. *Supportability*

The system designers shall take into considerations the following supportability and technical limitations:

3.5.1. *Internet Protocols*

The system shall comply with the TCP/IP protocol standards and shall be designed accordingly.

3.5.2. *Maintenance*

The maintenance of the system shall be done as per the maintenance contract.

3.5.3. *Standards*

The coding standards and naming conventions will be as per the creators' standards or the coding style of the language used.

3.6. Design Constraints

3.6.1. Software Language Used

The frontend of the system shall be developed using a combination of HTML, CSS, and the various of Javascript with support from various frameworks like Bootstrap. The backend will be developed using python and MongoDB.

3.6.2. Development Tools

The undertaking requires MDN as the online reference manual for developing the frontend. The IDE to integrate the project will be PyCharm or Visual Studio Code. The project will be tested on both Windows and Linux platforms. Git and GitHub will be the primary version control softwares.

3.6.3. Class Libraries

Various Node.js and python modules along with electron may be used.

3.7. Online User Documentation and Help System Requirements

Online help, visual cues and tooltips are provided for all the features in the Agritech System. The nature of these subsystems is unique to application development as they combine aspects of programming (hyperlinks, etc) with aspects of technical writing (organization, presentation).

The User Manual describes the use of the system to aforementioned categories of users in a simple and understandable manner. It describes how to use the system. The user manual will be available in digital form.

3.8. Purchased Components

If the system is required to handle more than 250 users at a time, a separate server will need to be purchased and server administrators will need to be trained to maintain the system.

3.9. Interfaces

3.9.1. User Interfaces

- Logon Screen Prototype:
- Homepage of Agritech System Prototype:
- Farmer Dashboard Prototype:
- New Contact Record Prototype:
- New Equipment Record Prototype:
- New Transaction Record Prototype:
- Customer Dashboard Prototype:
- Discussion Forum Prototype:

3.9.2. *Hardware Interfaces*

The existing LAN will be used for collecting data from the users and also for updating the data.

3.9.3. *Software Interfaces*

Login credentials will be verified with the server to prevent unauthorized access to the system.

3.9.4. *Communication Interfaces*

The Agritech System will be connected to the World Wide Web.

3.10. **Licensing Requirements**

The usage is restricted to the geographic boundaries of India and only to those users who have accounts with the system who have agreed to the terms and conditions of the usage contract.

3.11. **Legal, Copyright, and Other Notices**

The Agritech System project is an undertaking made for educational purposes and must not be copied or recreated without the creators' consent.

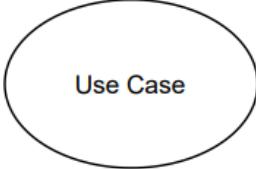
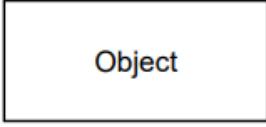
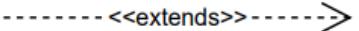
Use Case Model

Aim

To create a UML use case model for the problem domain that includes:

- researching the business domain
- documenting UML notations
- identifying use cases
- identifying different types of actors
- identifying scenarios
- identifying sub-functions
- determining relationship between use cases
- determining relationship between use cases and actors
- creating use case diagrams adhering to UML standards for the identified scenarios
- expanding identified sub-functions by creating use case diagrams adhering to UML standards
- providing a fully dressed use case description for each scenario
- documenting observations on the use case model of the system

Notations

Figure	Notation	Explanation
	Use Case	Represents various functionality of the system
	Actor	Represents the various types of users of the system
	Object	Represents an object associated with the system
	Relationship	Represents “includes” relationship between use cases
	Relationship	Represents “extends” relationship between use cases
	Relationship	Represents generalization between actors and between use cases
	Relationship	Represents association between use cases

Identification of Actors

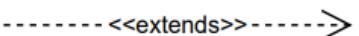
Actor	Scenario	Description
Farmer	<ul style="list-style-type: none"> ● Main Success ● Failure ● Agri-info ● Buy ● Sell 	The primary user of the system to provide automation of tasks from seed-to-sale
Customer: <ul style="list-style-type: none"> ● Retailer ● Wholesaler ● End consumer 	<ul style="list-style-type: none"> ● Main Success ● Failure ● Buy 	Purchases from the farmer and falls into different categories depending on the context
Admin	<ul style="list-style-type: none"> ● Main Success ● Failure ● Forum 	Oversees the system operation and maintenance, pertains to the context
Content Manager	<ul style="list-style-type: none"> ● Main Success ● Failure ● Agri-info 	Oversees content management, pertains to the context
Agricultural Expert/ Officer	<ul style="list-style-type: none"> ● Agri-info ● Discussion Forum ● Resource Manager 	Some users are verified, registered and granted special privileges to participate in certain contexts
User	<ul style="list-style-type: none"> ● Discussion Forum ● Resource Manager 	Relies on the functionality of the system, pertains to context

Identification of Scenarios

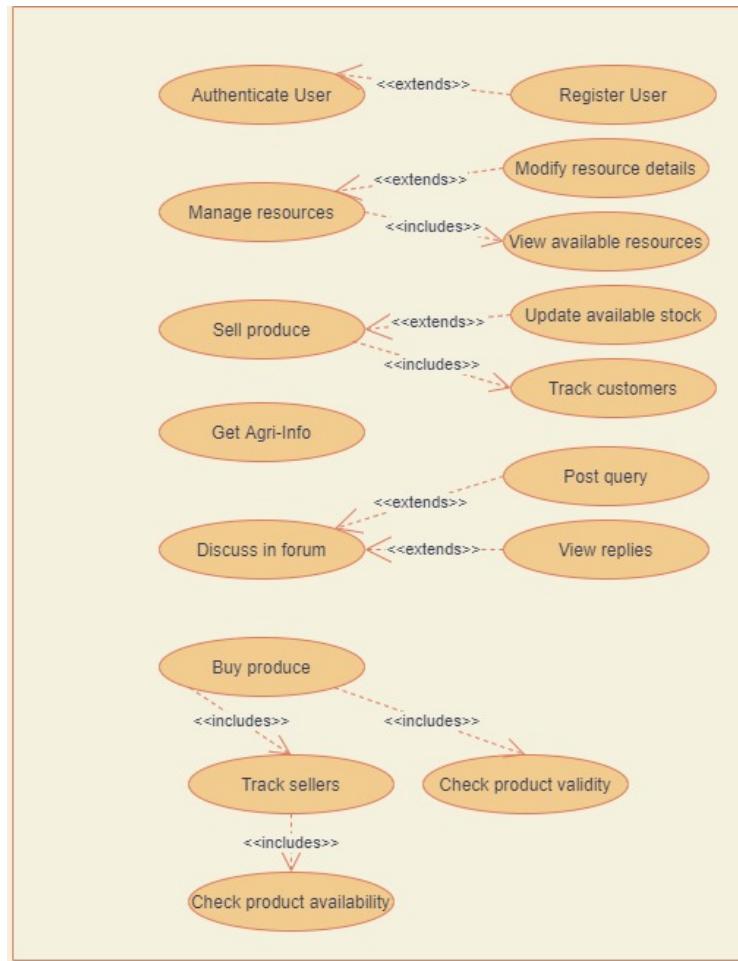
Scenario	Type	Description
Main Success	User goal scenario	Complete model inclusive of all the system functionalities and behaviour
Failure	Alternate scenario	Describes the failure model
Buy	Sub-function	Farmer or customer is connected to a seller of choice from a list generated by the algorithm
Sell	Sub-function	Farmer is connected to a buyer of choice from a list generated by the algorithm
Agri-info	Sub-function	Details of various crops, governmental policies and other expert curated resources to aid decision making for precision farming
Resource Manager	Sub-function	Smart system that aggregates data from CRM, financials and inventory to boost sales and gain insights
Discussion Forum	Sub-function	Communication channel in private or public mode for user interaction supplemented with on-demand expert advisory

Relating Use Cases

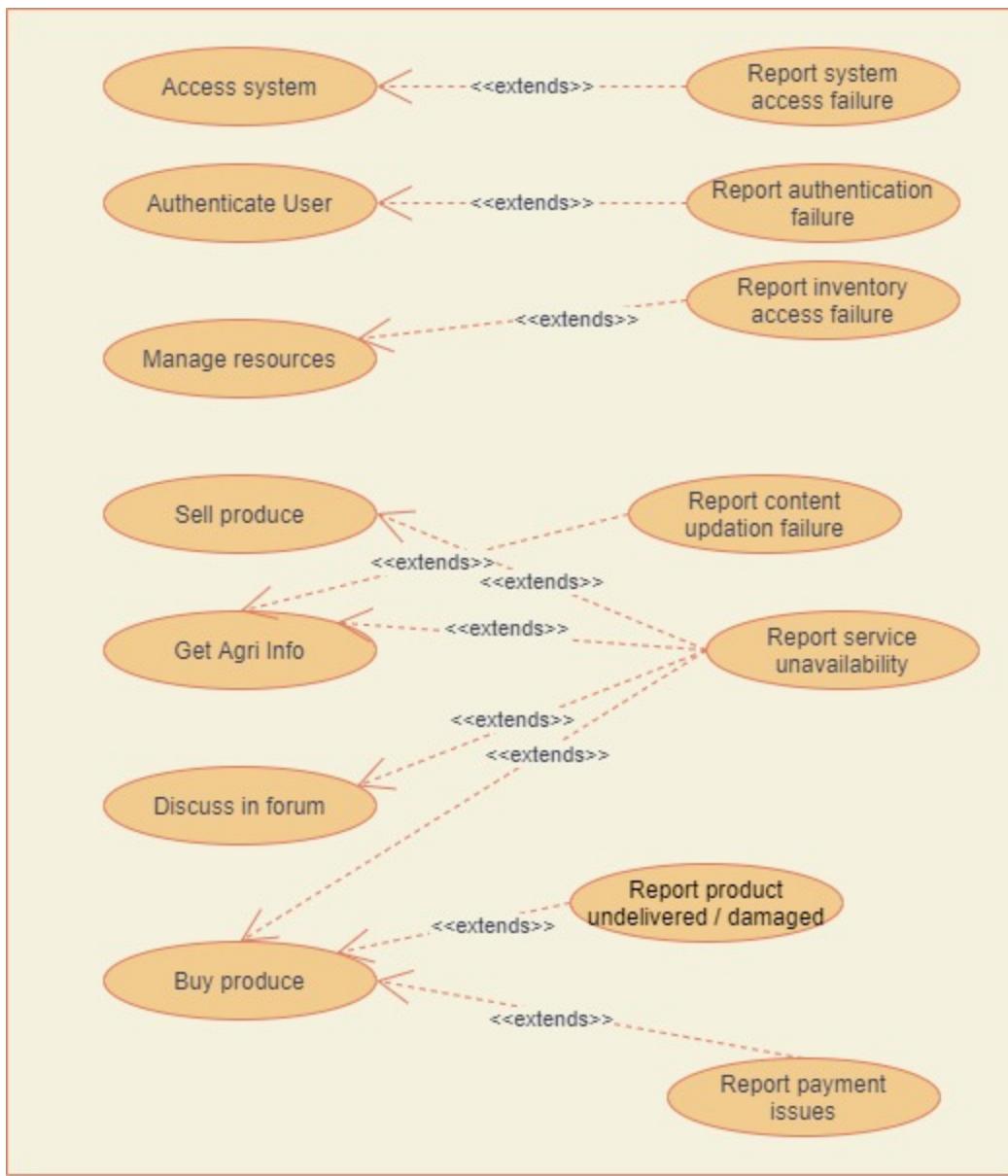
Notations Guide

Figure	Notation	Explanation
	Relationship	Represents “includes” relationship between use cases
	Relationship	Represents “extends” relationship between use cases
	Relationship	Represents generalization between actors and between use cases
	Relationship	Represents association between use cases

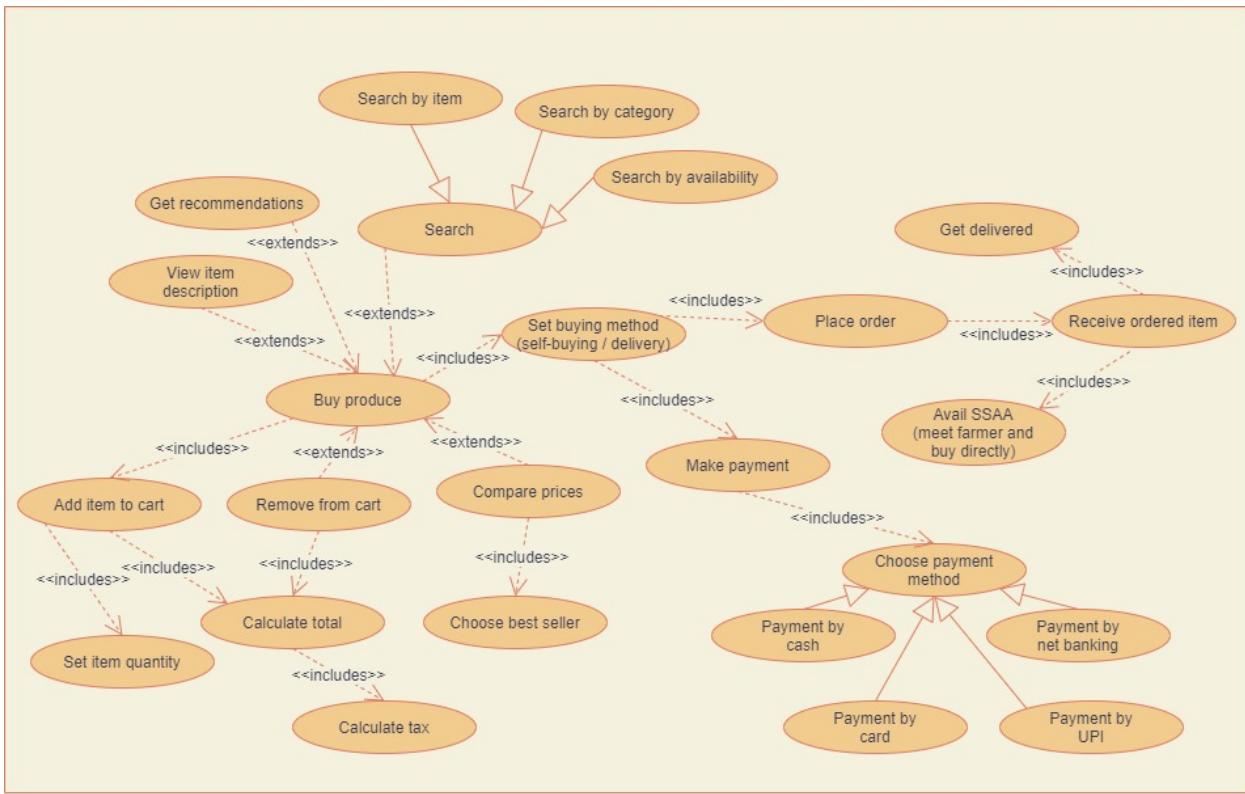
User goal : Main Success Scenario



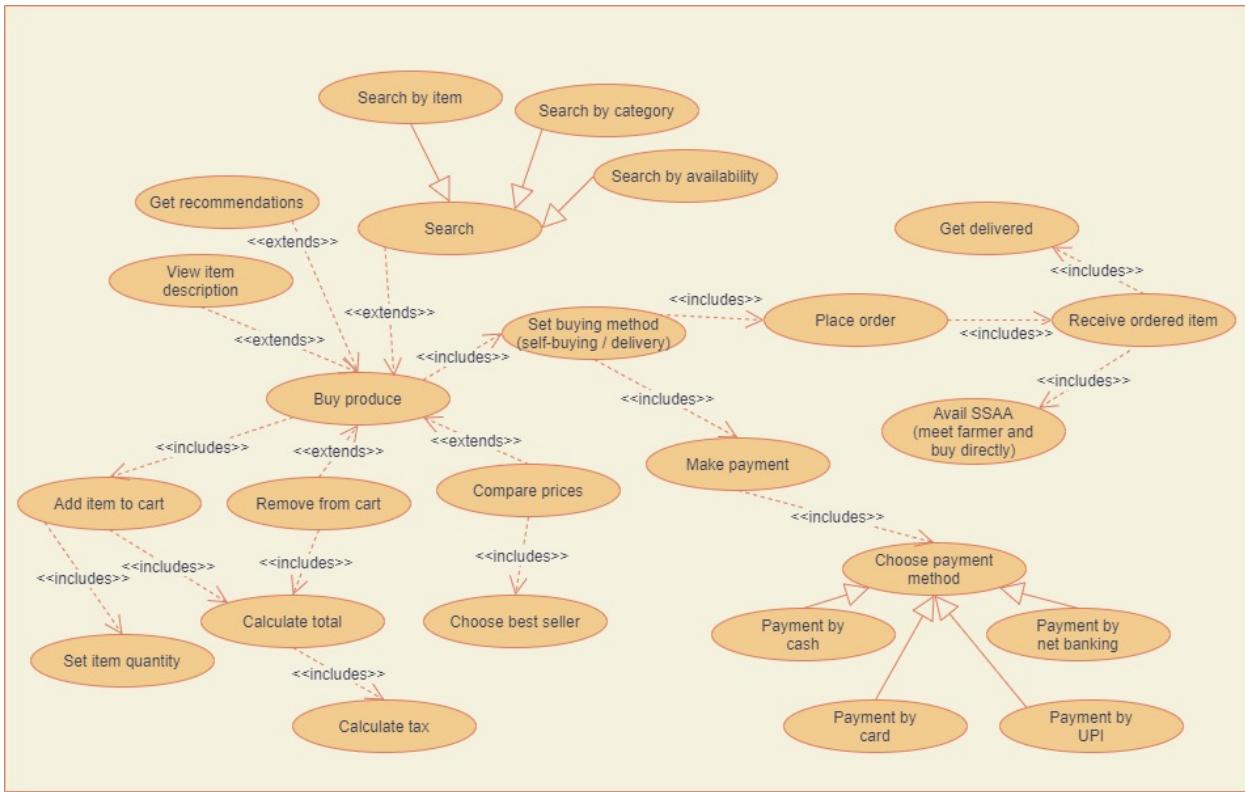
Alternate Scenario : Failure Scenario



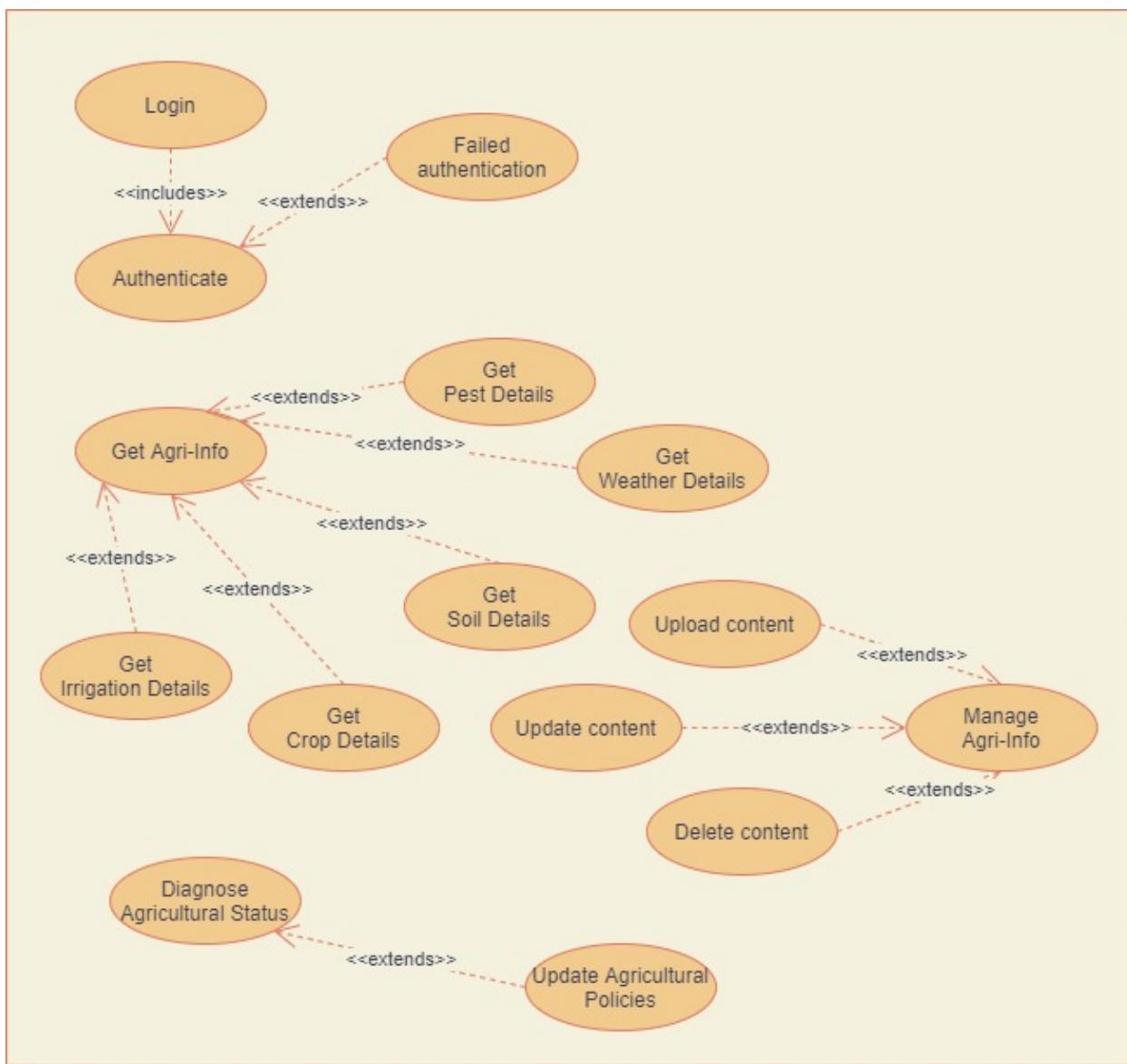
Sub-function: Buy



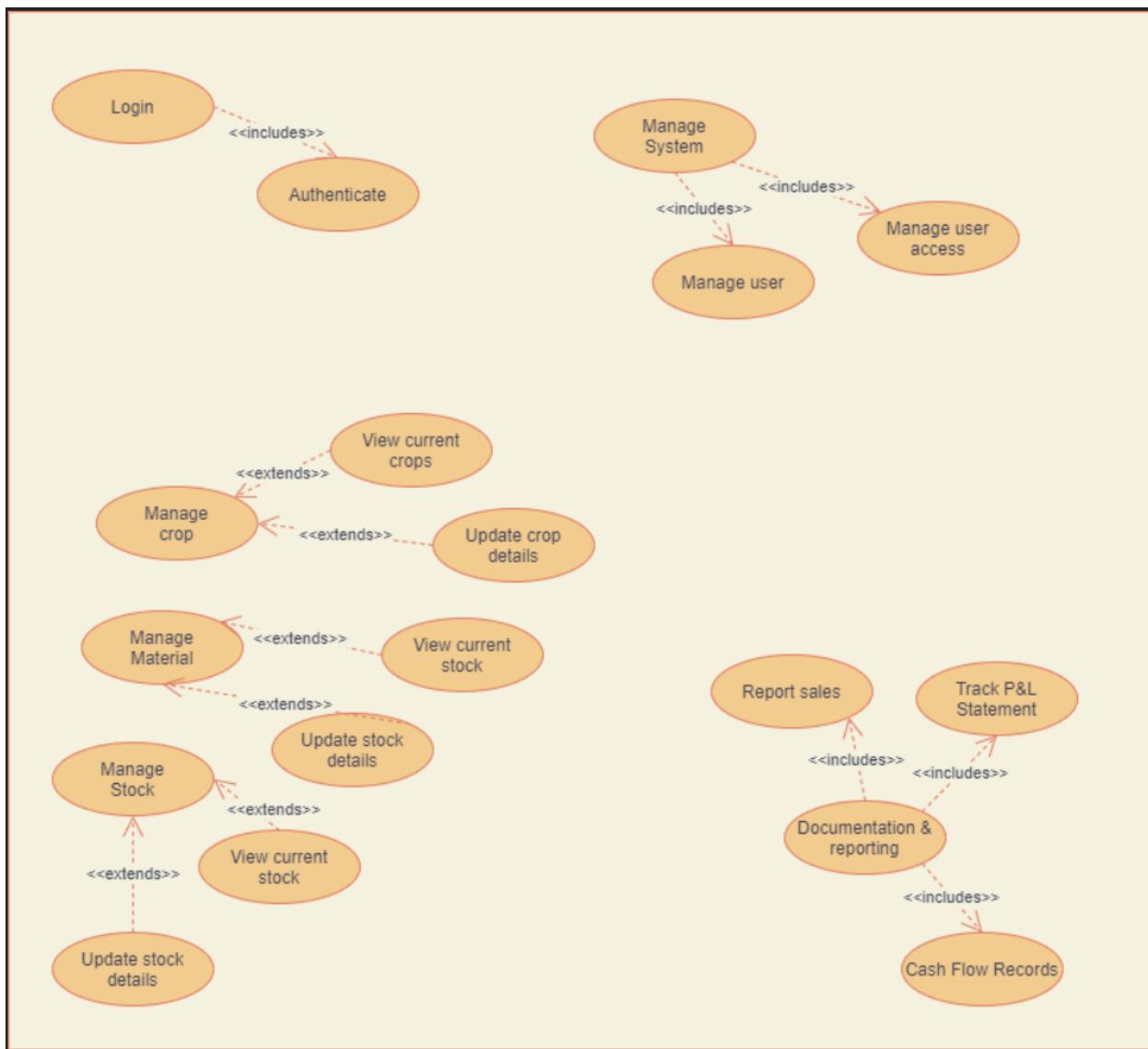
Sub-function: Sell



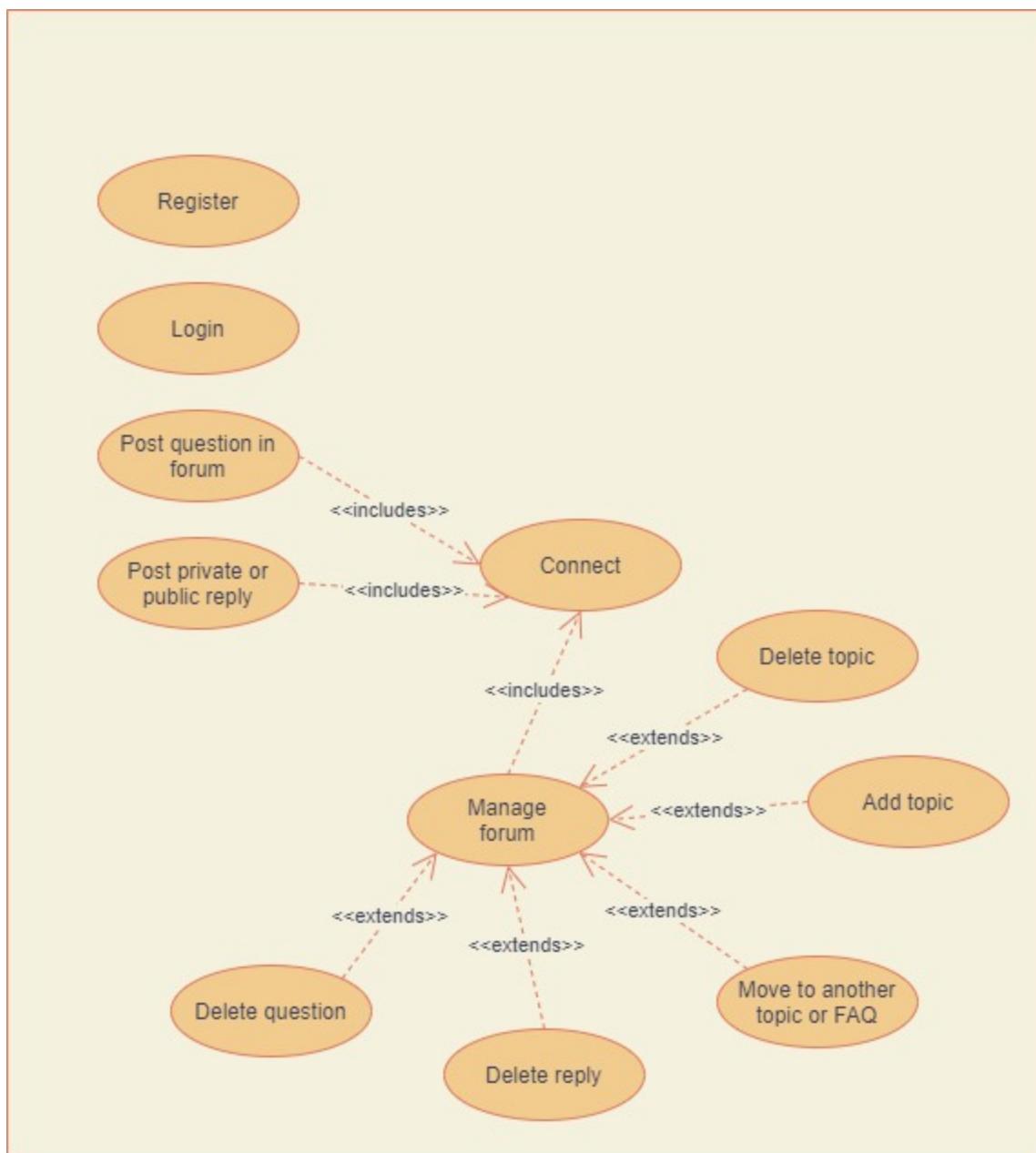
Sub-function: Agri-info



Sub-function: Resource Manager

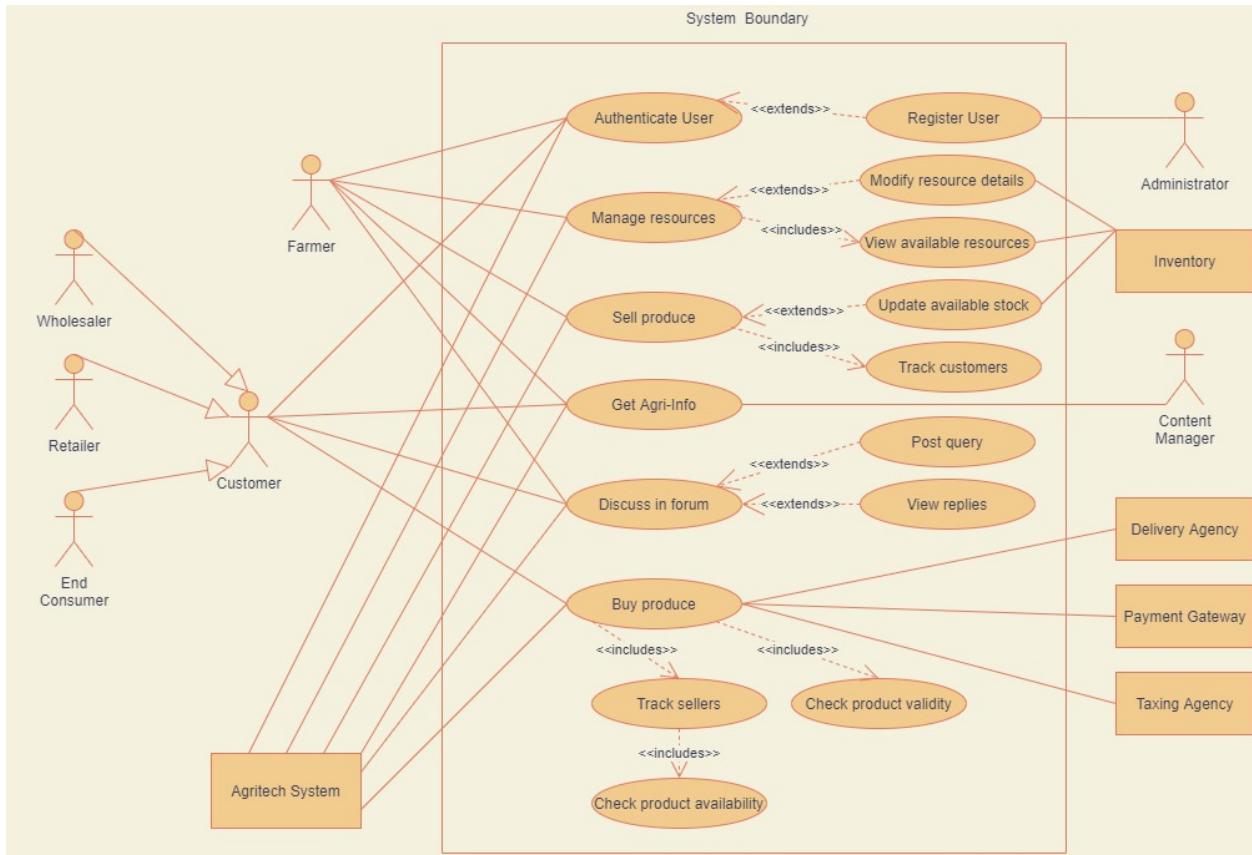


Sub-function: Discussion Forum

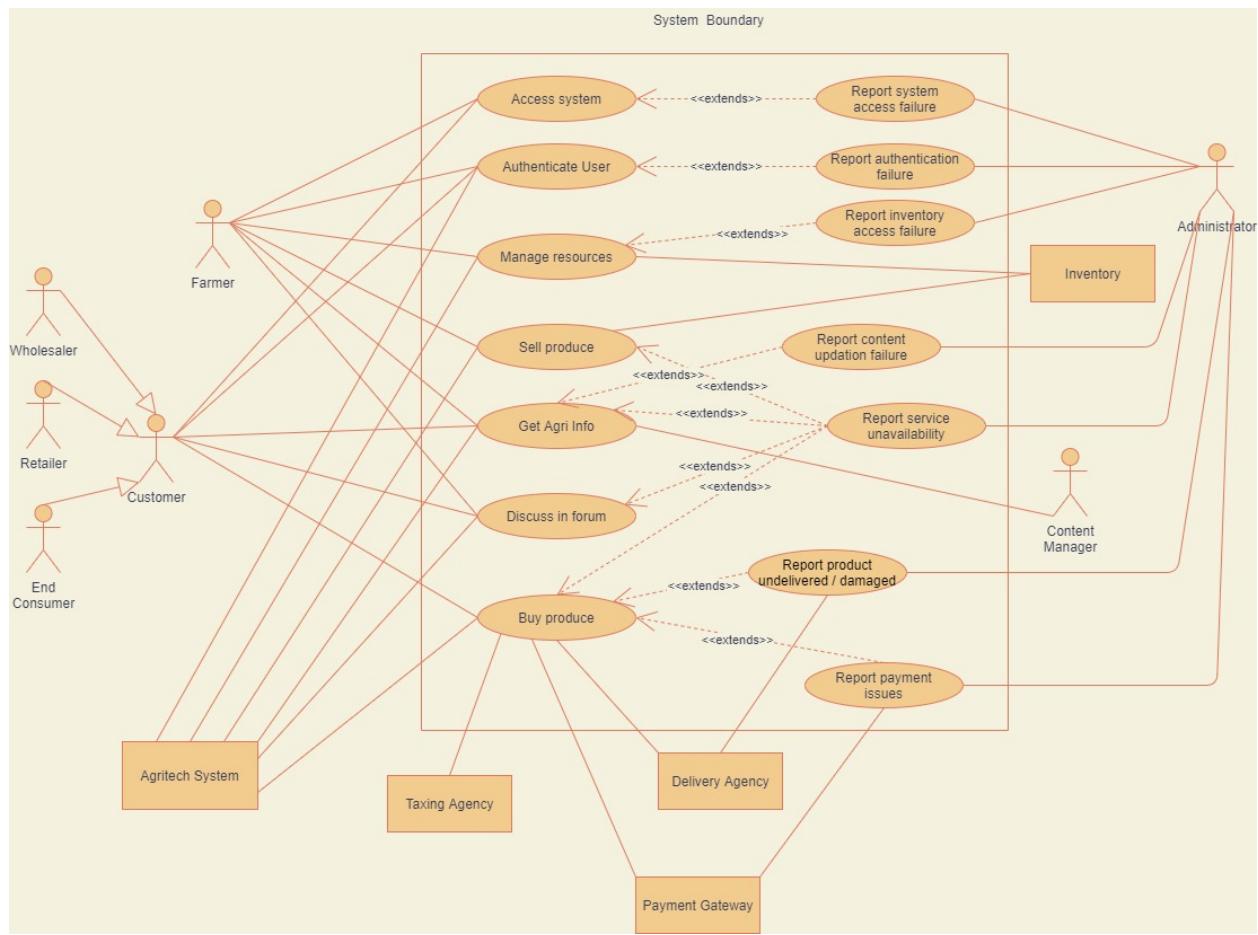


Use Case Diagrams

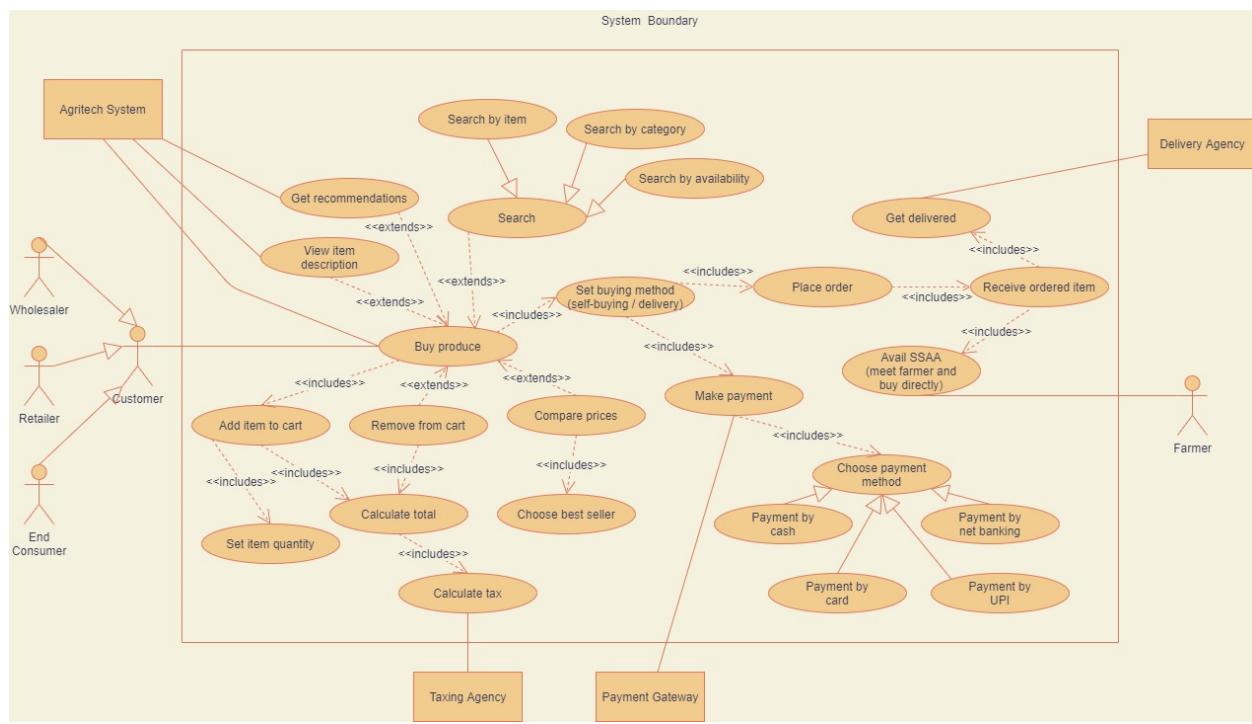
User goal : Main Success



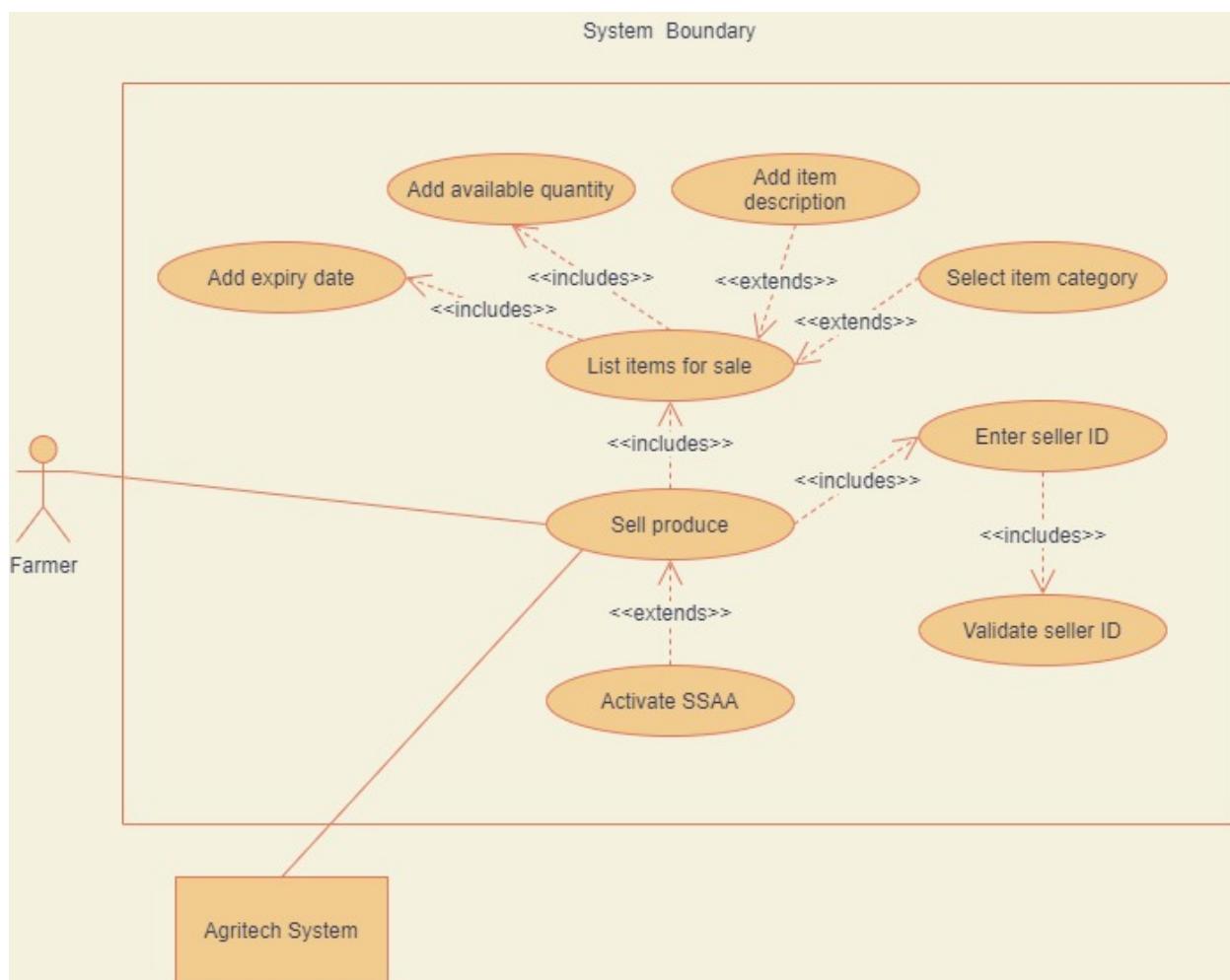
Alternate Scenario : Failure Scenario



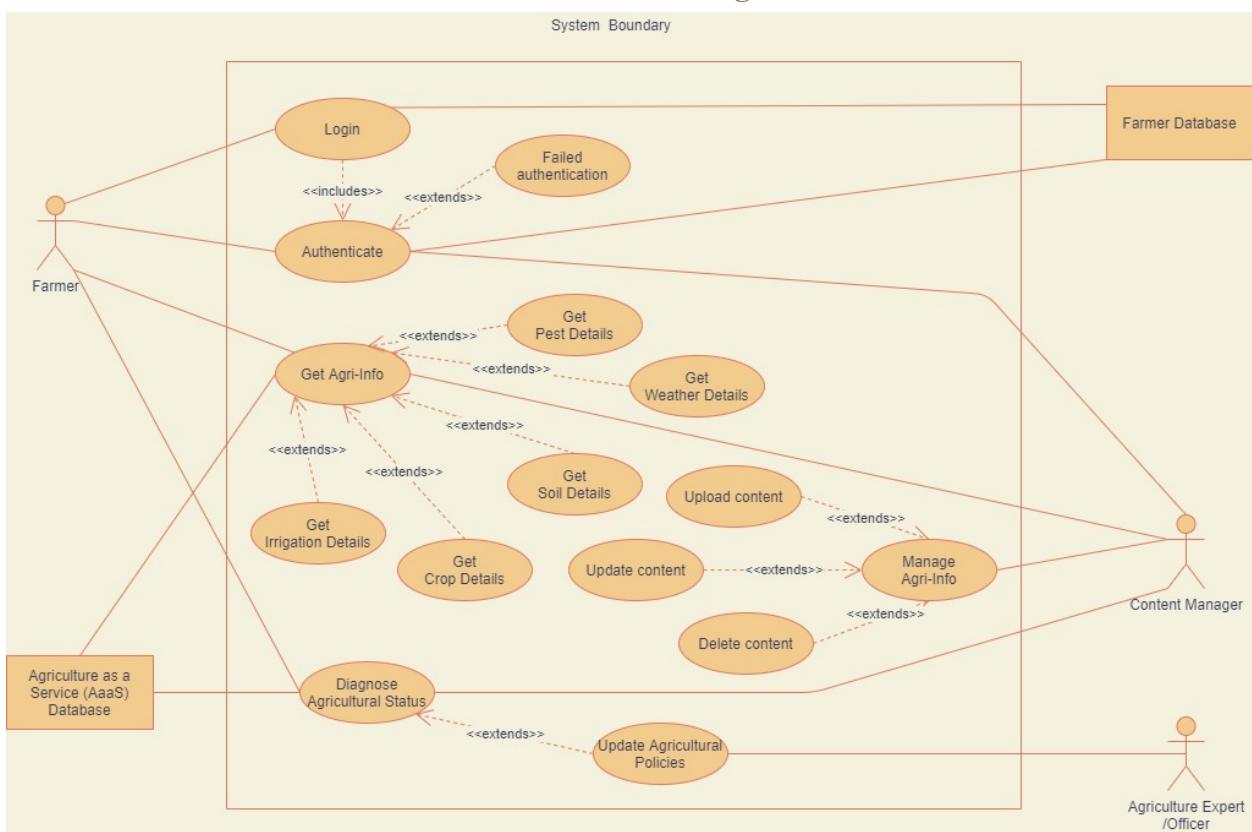
Sub-function : Buy



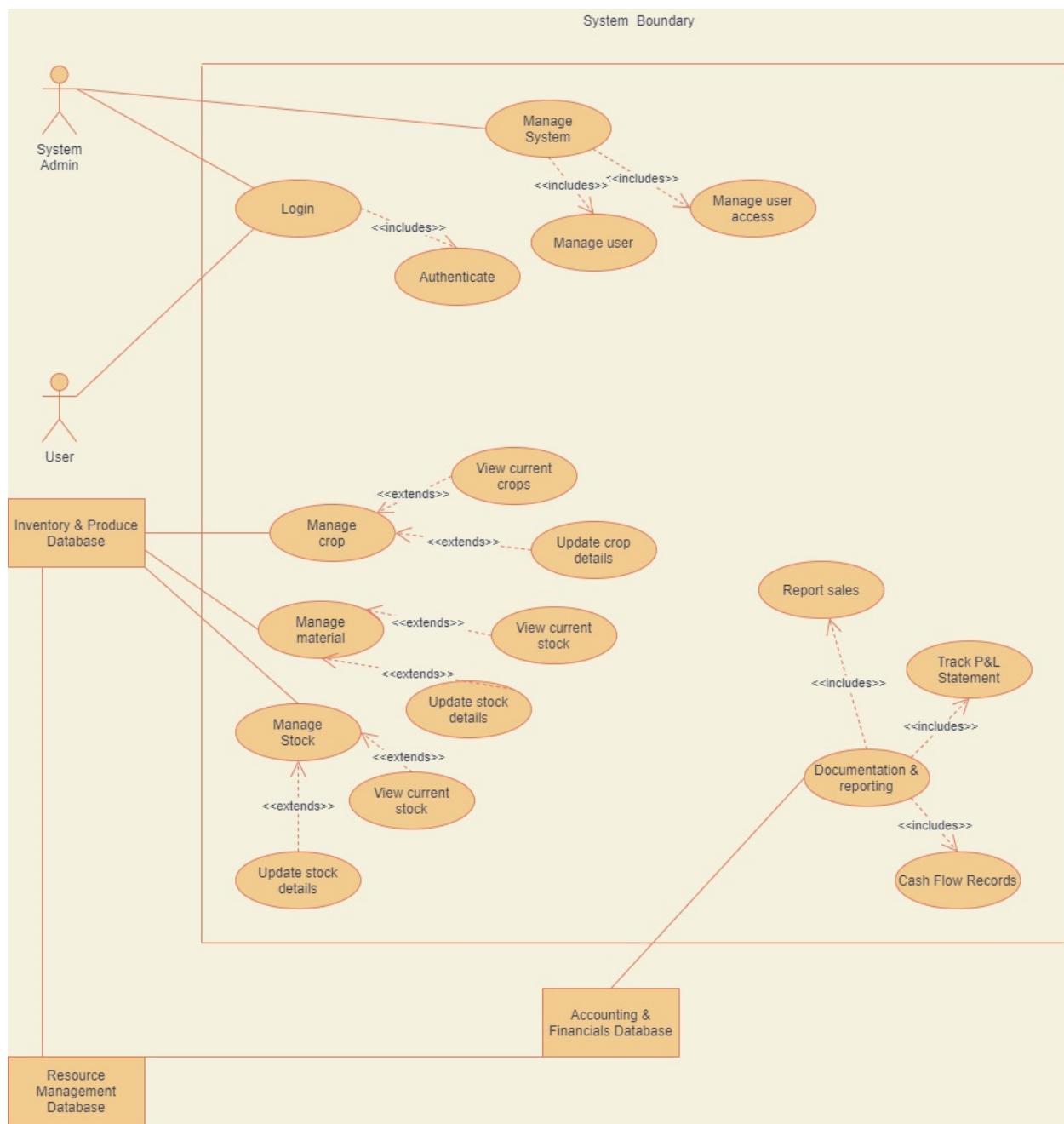
Sub-function : Sell



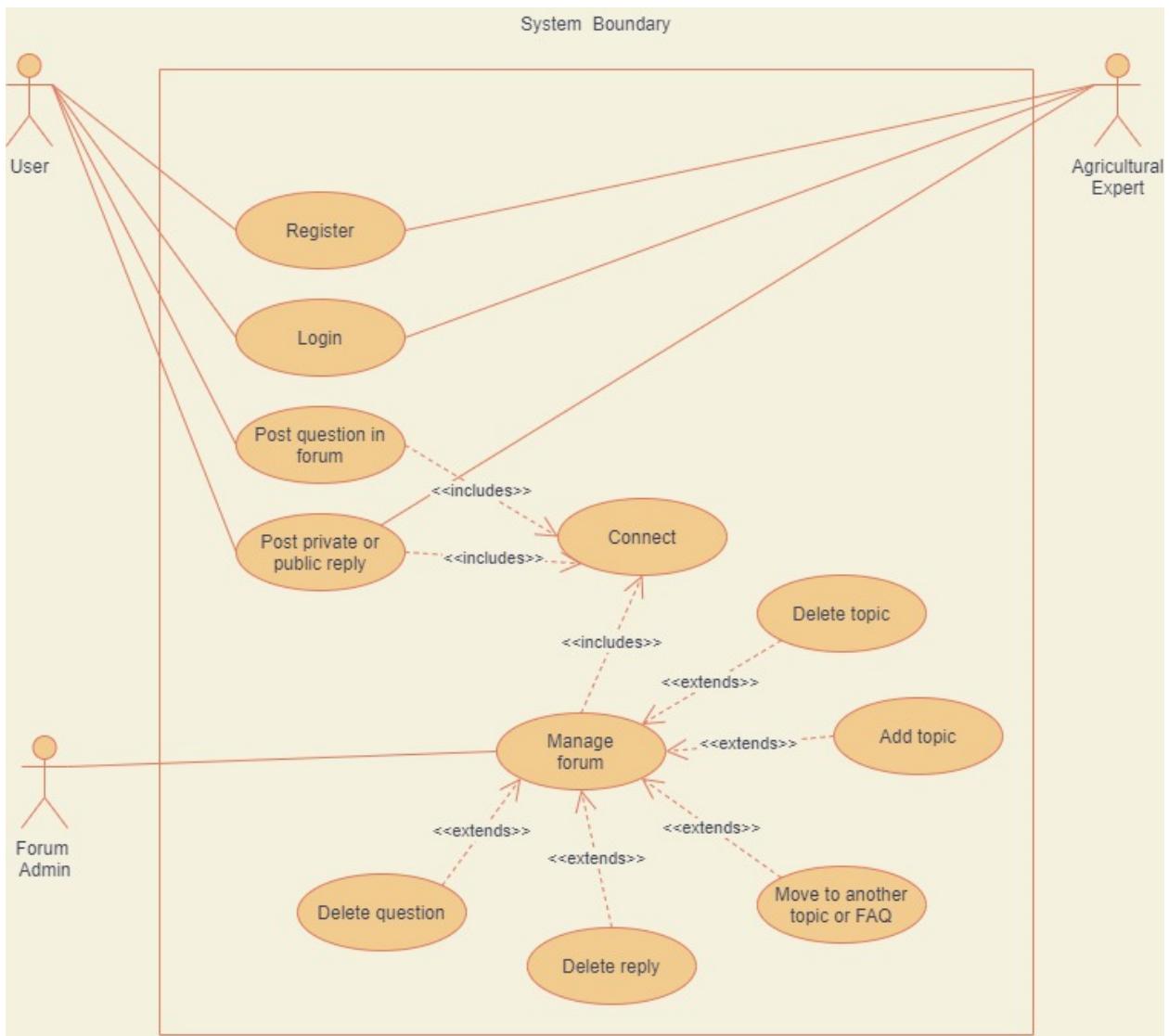
Sub-function : Agri-info



Sub-function : Resource Manager



Sub-function : Discussion Forum



Fully Dressed Use Case Description

Main Success

Scope	: Any agritech portal or system
Level	: User goal
Primary actors	: Farmer, Customer

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Customer** : Wants to buy fresh and high-quality products at reasonable prices (end consumer, retailer or wholesaler)
- **Administrator** : Wants to keep the system functional and running smoothly all the time
- **Content manager** : Wants updated, relevant and accurate information related to agriculture which could be used to provide useful details to users and crop assistance to the farmers
- **Government tax agencies** : Want to collect tax from every sale (may be multiple agencies, such as national, state, and county)
- **Payment authorization service** : Wants to receive digital authorization requests in the correct protocol and manage accounting
- **Delivery agencies** : Wants to deliver the product to the customer in a timely manner without any damage to the products

Preconditions : Administrators and content managers are identified and authenticated

Success guarantee : Users are able to access the application and avail services like Agri-info, SSAA, resource management, discussion forum, etc without any difficulty

Main Success Scenario :

1. User opens the Agritech application
2. Users access various agriculture related information and benefit from that information
3. User logs in as a farmer or customer after authentication or else registers as a new user
4. Farmers can use the Agri-info for crop assistance
5. Farmer manages and keeps track of all the resources owned by him like crops, fertilizers, equipment, finances, etc
6. Farmers can analyse various resources held so as to optimally use the remaining resources
7. Farmers can post queries in the discussion forum which can be answered by anyone
8. Farmers can answer any queries posted in the discussion forum
9. Farmers can list all the produce available with him for sale
10. Farmers can also activate the SSAA feature which will enable the farmer to meet the customer directly and sell the produce
11. Customers can use the Agri-info to gain insight about agriculture
12. Customers can buy fresh produce
13. Customers can either place orders for products which will be delivered by the delivery agency or else they can avail the SSAA feature to meet the farmers nearby directly and buy the product themselves without the help of the delivery agency
14. Customers can post any agriculture related queries in the discussion forum
15. User logs out of the application after availing the services without facing any issues

Extension : If the farmer or customer is unable to avail any service, then the farmer or customer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Robust recovery when access to remote services such as inventory systems, payment gateway, delivery agency and taxing agency are failing
- Reporting to system administrators within 10 minutes of issues or errors

Failure

Scope	: Any agritech portal or system
Level	: Frequent alternate scenario
Primary actors	: Administrator, Farmer, Customer

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Customer** : Wants to buy fresh and high-quality products at reasonable prices (end consumer, retailer or wholesaler)
- **Administrator** : Wants to keep the system functional and running smoothly all the time
- **Content manager** : Wants updated, relevant and accurate information related to agriculture which could be used to provide useful details to users and crop assistance to the farmers
- **Government tax agencies** : Want to collect tax from every sale (may be multiple agencies, such as national, state, and county)
- **Payment authorization service** : Wants to receive digital authorization requests in the correct protocol and manage accounting
- **Delivery agencies** : Wants to deliver the product to the customer in a timely manner without any damage to the products

Preconditions : Administrators and content managers are identified and authenticated

Success guarantee : Users are able to access the application and avail services like Agri-info, SSAA, resource management, discussion forum, etc without any difficulty

Frequent Alternate Scenario :

1. User opens the Agritech application
2. User is unable to access various agriculture related information and so reports the issue to the administrator
3. Users are unable to register or log in as a farmer or customer
4. Users are unable to authenticate themselves
5. Farmers are unable to use the Agri-info
6. Farmers are unable to manage resources
7. Farmers are unable to analyse various resources
8. Farmer is unable to post queries in the discussion forum
9. Farmer is unable to answer any queries posted in the discussion forum
10. Farmer is unable to list all the produce available with him for sale
11. Farmers are unable to activate the SSAA feature which will enable the farmer to meet the customer directly and sell the produce
12. Customers are unable to use the Agri-info to gain insight about agriculture
13. Customers are unable to place orders to buy fresh produce
14. Customer is unable to complete the payment process
15. Customers are unable to track the undelivered products
16. Customers receive damaged products
17. Customers are unable to post any agriculture related queries in the discussion forum
18. Content manager is unable to update the Agri-info and so reports content updation failure to the administrator
19. Administrator can also be reported regarding the system access failure, authentication failure, inventory access failure, service unavailability, product undelivered / damage, payment issues, etc. by the users

Extension : If the farmer or customer is unable to avail any service, then the farmer or customer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Robust recovery when access to remote services such as inventory systems, payment gateway, delivery agency and taxing agency are failing
- Reporting to system administrators within 10 minutes of issues or errors

Buy

Scope	: Any online shopping system
Level	: Sub-function
Primary actor	: Customer

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Customer** : Wants to buy fresh and high-quality products at reasonable prices (end consumer, retailer or wholesaler)
- **Government tax agencies** : Want to collect tax from every sale (may be multiple agencies, such as national, state, and county)
- **Payment authorization service** : Wants to receive digital authorization requests in the correct protocol and manage accounting
- **Delivery agencies** : Wants to deliver the product to the customer in a timely manner without any damage to the products

Preconditions : Customer is registered and has logged in after authentication

Success guarantee :

- Customers are able to place items in the cart and the orders which will be delivered by the delivery agency
- Avail SSAA feature which will enable the customer to meet the nearby farmers directly and get the items without the intervention of the delivery agency

Buy Sub-function :

1. Customer opens the Agritech application
2. Customer logs in after authentication or else registers as a new user
3. Customer can get recommendations regarding various products available for sale
4. Customer can search by item, category or availability
5. Customer can view the item description of any produce
6. Customer can add item to cart and specify the quantity required
7. Customer can remove item from the cart
8. Customer can compare prices and choose the best seller
9. System calculates the total including the tax
10. Customer can set buying method as self-buying or delivery
11. Customer can either place orders for products which will be delivered by the delivery agency or else they can avail the SSAA feature to meet the farmers nearby directly and buy the product themselves
12. Customer can make payment by choosing payment by cash, card, upi or net banking
13. Customer logs out of the application after availing the services without facing any issues
14. Customer receives ordered items undamaged and without much delay

Extension : If the customer is unable to avail any service, then the customer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Robust recovery when access to remote services such as inventory systems, payment gateway, delivery agency and taxing agency are failing
- Reporting to system administrators within 10 minutes of issues or errors

Sell

Scope	: Any online shopping system
Level	: Sub-function
Primary actor	: Farmer

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit

Preconditions : Farmer is registered and has logged in after authentication

Success guarantee : Farmer is able to list the items for sale and activate SSAA feature to sell their produce at good rates

Sell Sub-function :

1. Farmer opens the Agritech application
2. Farmer logs in after authentication or else registers as a new user
3. Farmer lists the available with him/her for sale
4. Farmer mandatorily mentions the expiry date of the particular item and the quality available for sale
5. Farmer can optionally add item description and select item category
6. After listing all items, farmer enters unique seller ID assigned to him/her
7. System updates the database which contains items for sale after validating the seller ID
8. Farmer can also activate the SSAA feature to meet the nearby customers directly and sell the items to them
9. Farmer logs out of the application after availing the services without facing any issues

Extension : If the farmer is unable to avail any service, then the farmer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Robust recovery when access to remote services such as inventory systems, payment gateway, delivery agency and taxing agency are failing
- Reporting to system administrators within 10 minutes of issues or errors

Agri-info

Scope	: Any curated content service system
Level	: Sub-function
Primary actors	: Farmer, Agriculture Expert

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Administrator** : Wants to keep the system functional and running smoothly all the time
- **Content manager** : Wants updated, relevant and accurate information related to agriculture which could be used to provide useful details to users and crop assistance to the farmers
- **Agricultural Expert/Officer** : Wants to support content manager with updated, relevant and accurate information related to agricultural policies which provides assistance to the farmers and reviews the validity of content published

Preconditions	: Administrators and content managers are identified and authenticated
Success guarantee	: Users are able to access the application and the avail services of Agri-info portal

Agri-info Sub-function :

1. User opens the Agritech application
2. Users access various agriculture related information and benefit from that information
3. User logs in as a farmer or customer after authentication or else registers as a new user
4. Farmers can use the Agri-info for crop assistance
5. Farmers can use the Agri-info for expert advice
6. Content manager maintains the database
7. Agricultural Expert gives timely replies and notifies content manager regarding policy changes
8. User logs out of the application after availing the services without facing any issues

Extension : If the farmer is unable to avail any service, then the farmer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Reporting to system administrators within 10 minutes of issues or errors

Resource Manager

Scope : Any resource management system

Level : Sub-function

Primary actors : Farmer

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Administrator** : Wants to keep the system functional and running smoothly all the time

Preconditions : Administrators and farmers are identified and authenticated

Success guarantee : Users are able to access the application and the avail services of the resource manager, namely, crop, stock and inventory management

Resource Manager Sub-function :

1. User opens the Agritech application
2. Users access various agriculture related information and benefit from that information
3. User logs in as a farmer after authentication or else registers as a new user
4. Farmers can analyse various resources held so as to optimally use the remaining resources
5. Farmers can track equipment and production stock reserves
6. User logs out of the application after availing the services without facing any issues

Extension : If the farmer is unable to avail any service, then the farmer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Robust recovery when access to remote services such as inventory systems, payment gateway, delivery agency and taxing agency are failing
- Reporting to system administrators within 10 minutes of issues or errors

Discussion Forum

Scope : Any discussion or communication forum

Level : Sub-function

Primary actors : Farmer, Agriculture Expert

Stakeholders and Interests :

- **Farmer** : Wants user-friendly, all-in-one application which can aid from seed-to-sale and help in maximizing yield and profit
- **Administrator** : Wants to keep the system functional and running smoothly all the time
- **Agricultural Expert** : Wants to respond to queries with updated, relevant and accurate information related to agriculture which provides assistance to the farmers

Preconditions : Farmers, administrators and content managers are identified and authenticated

Success guarantee : Users are able to access the application and the avail services of the discussion forum

Discussion Forum Sub-function :

1. User opens the Agritech application
2. Users access various agriculture related information and benefit from that information
3. User logs in as a farmer or customer after authentication or else registers as a new user
4. Farmers can post and answer queries in the discussion forum which can be answered by anyone in private or public channels
5. Customers can post any agriculture related queries in the discussion forum in private or public channels
6. Agricultural experts can answer any queries posted in the discussion forum in private or public channels
7. User logs out of the application after availing the services without facing any issues

Extension : If the farmer is unable to avail any service, then the farmer can report to the administrator

Special Requirements :

- Internet connectivity
- System response time should be within 30 seconds 90% of the time
- Reporting to system administrators within 10 minutes of issues or errors

Documentation

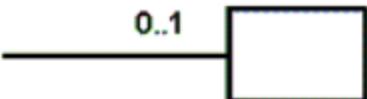
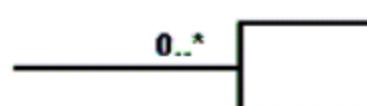
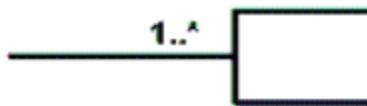
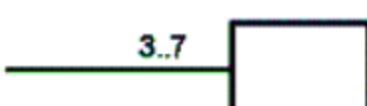
This serves as the UML use case model for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. The **Main Success Scenario** and **Alternate Scenario** describe in detail the goals and shortcomings of the system in detail respectively. For assisting in crop selection the **Agri-info portal sub-function** has been expanded in the use case model. The **Buy sub-function** and **Sell sub-function** detail the process SSAA. The **Resource Manager sub-function** handles tracking crops and managing inventory. Furthermore, a collaboration forum amongst farmers will facilitate a crowd-sourced knowledge base that is handled by a **Discussion Forum sub-function**.

Domain Model and Class Diagram

Aim

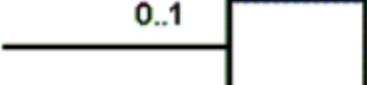
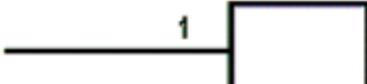
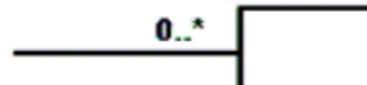
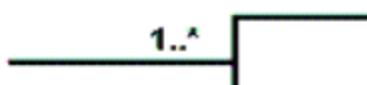
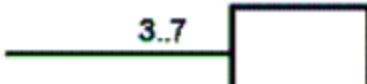
- To map the Agritech System to a UML Domain Model
- To create a UML Class Diagram for the Agritech System

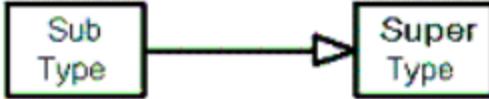
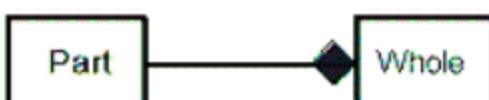
UML Notations for Domain Model Diagram

Notation	UML
Multiplicities	
Zero or one	 A line segment ending in an open rectangle, with the text "0..1" written above it.
One only	 A line segment ending in an open rectangle, with the text "1" written above it.
Zero or more	 A line segment ending in an open rectangle, with the text "0..*" written above it.
One or more	 A line segment ending in an open rectangle, with the text "1..*" written above it.
Specific range	 A line segment ending in an open rectangle, with the text "3..7" written above it.

Attributes						
Names	attributeName: Type					
Methods	+ getIrrigationInfo()					
Associations						
Subtyping						
Aggregation						
Composition						
Class						
Class	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">Stock</td> </tr> <tr> <td style="padding: 5px;">+ name: String</td> </tr> <tr> <td style="padding: 5px;">+ quantity: Integer</td> </tr> <tr> <td style="padding: 5px;">+ price: float</td> </tr> <tr> <td style="padding: 5px;">+ description: String</td> </tr> </table>	Stock	+ name: String	+ quantity: Integer	+ price: float	+ description: String
Stock						
+ name: String						
+ quantity: Integer						
+ price: float						
+ description: String						

UML Notations for Class Diagram

Notation	UML
Multiplicities	
Zero or one	 A line connects to a rectangle. Above the line is the text "0..1".
One only	 A line connects to a rectangle. Above the line is the text "1".
Zero or more	 A line connects to a rectangle. Above the line is the text "0..*".
One or more	 A line connects to a rectangle. Above the line is the text "1..*".
Specific range	 A line connects to a rectangle. Above the line is the text "3..7".
Attributes	
Names	attributeName: Type
Methods	+ getIrrigationInfo()
Associations	

Subtyping	
Aggregation	
Composition	
Class	
Class	<pre> class AgrilInfo { + crop: String + irrigation: String + soil: String + weather: String + pest: String + getIrrigationInfo() + getCropInfo() + getSoilInfo() + getWeatherInfo() + getPestInfo() } </pre>

Identification of Classes

Identification of Noun Phrases

The **Agritech system** should be an all-in-one software that helps **farmers** be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive **farm**. It should assist in **crop** selection and tracking, manage **inventory**, market **stock** and sell online, on a user-friendly platform available on any device from anywhere. Furthermore, a **collaboration forum** amongst **farmers** will facilitate a crowd-sourced knowledge base. The Agritech system should eliminate the need for **middlemen**, increasing **farmer profits**.

The system must enable marketing and selling directly to **customers** online. This Smart-Selling-Anywhere-Anytime(SSAA) feature automates **customer** communications, manages pickup options and immensely simplifies **revenue** and financial reporting. The system uses hotspot(short ranges) and GPS(long ranges) for scoping **customers** or other **farmers** in the vicinity. The **information** accumulated in the system can be used to normalize the availability of **produce** in the **geographical area** which in turn eradicates the supply-demand problem.

The **Resource Manager** feature for agriculture will allow efficient and effective data collection and tracking of **customer** and **supplier** lifecycles to manage **crops**, **inventory** and other farm **resources**. By integrating **crops**, **accounting** and **inventory** management together, the **farmer** is in control of his entire workflow.

Conceptual Class Category List

CATEGORY LIST

- Red - Roles of peoples
- Orange - Catalog
- Green - Transaction Item
- Blue - Physical / Tangible Items
- Violet - Financial Transaction
- Pink - Container

CONCEPTUAL CLASSES

- User
 - Customer
 - Farmer
 - Administrator
 - Content Manager
- Agri-info
- Item
- Resource
 - Material
 - Stock
 - Crop
- Discussion Forum

Identification of Associations

Association Category List

- An association is a relationship between classes that indicates some meaningful and interesting connection
- Each end of an association is called a role
- In the UML, associations are defined as “the semantic relationship between two or more classifiers that involve connections among their instances
 - In UML, simple line is an **association**

- A simple line with an arrowhead denotes **generalization**



- A black-filled diamond denotes a **composition**, a part, unique to this whole



- A white-empty diamond denotes an **aggregation**, a part, but not unique to this whole



Definition of Associations and their notations

GENERALIZATION

- In UML modeling, a generalization relationship is a relationship that implements the concept of object orientation called inheritance
- The generalization relationship occurs between two entities or objects, such that one entity is the parent, and the other one is the child
- The child inherits the functionality of its parent and can access as well as update it
- Generalization relationship is utilized in class diagrams to specify that the child inherits actions, characteristics, and relationships from its parent



AGGREGATION

- An aggregation is a subset of association, which represents has a relationship
- It is more specific than association
- It defines a part-whole or part-of relationship
- In this kind of relationship, the child class can exist independently of its parent class



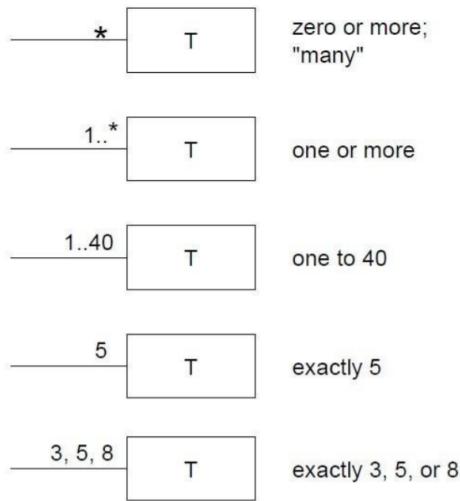
COMPOSITION

- The composition is a subset of aggregation
- It portrays the dependency between the parent and its child, which means if one part is deleted, then the other part also gets discarded
- It represents a whole-part relationship

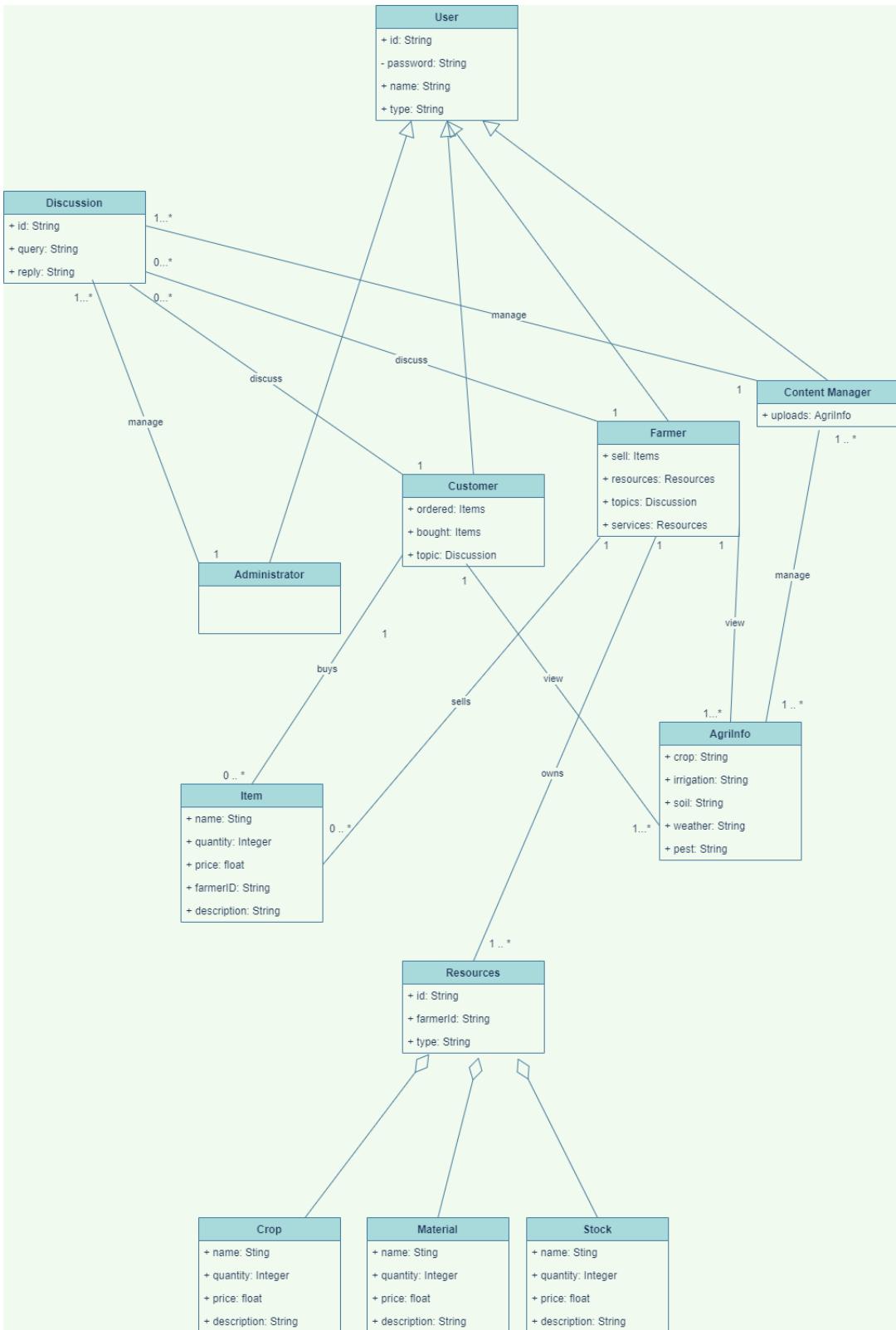


Multiplicity based associations

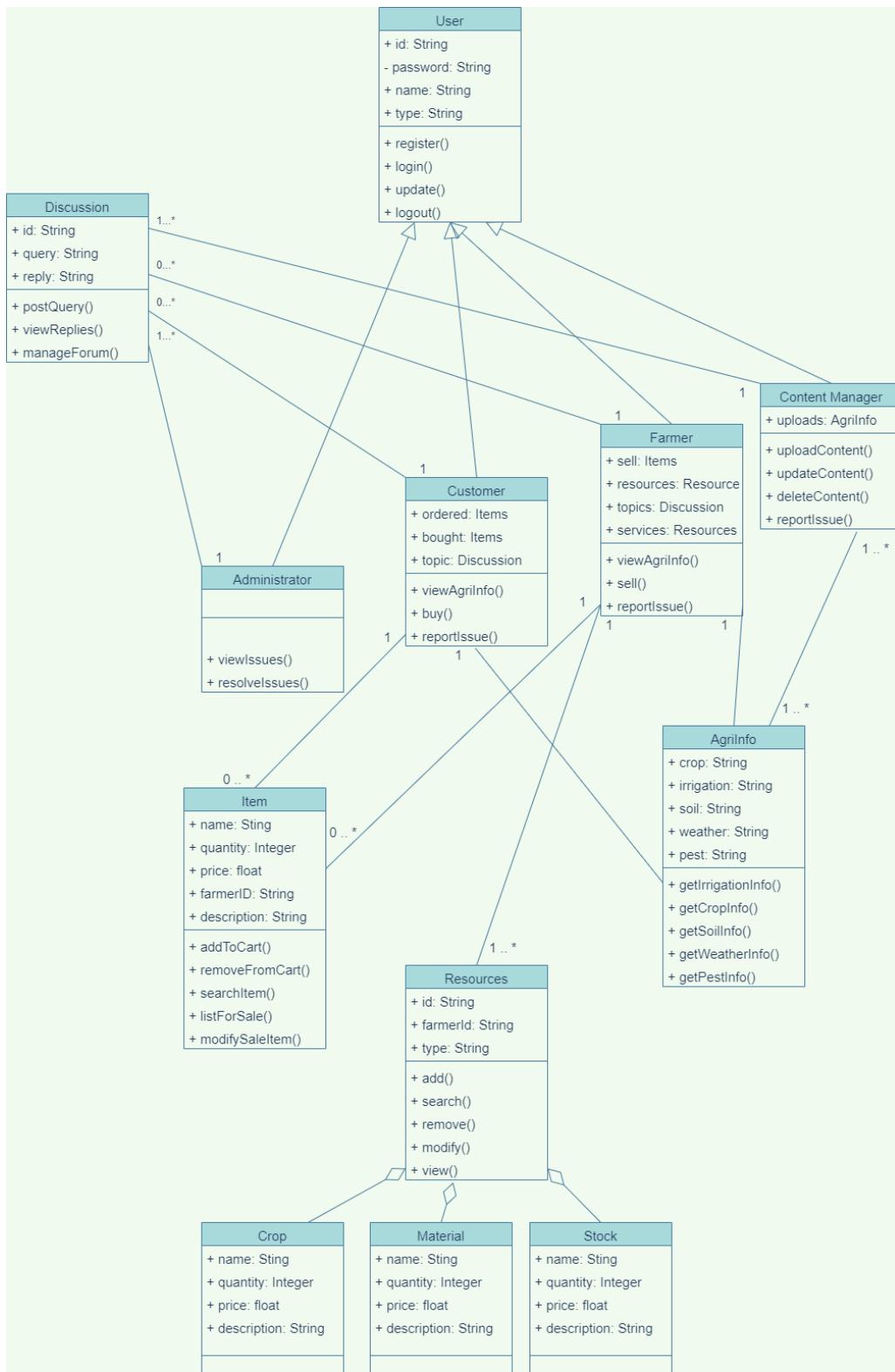
- Multiplicity defines how many instances of a class A can be associated with one instance of a class B
- It defines a specific range of allowable instances of attributes
- In case if a range is not specified, one is considered as a default multiplicity



Domain Model Diagram



Class Diagram



Documentation

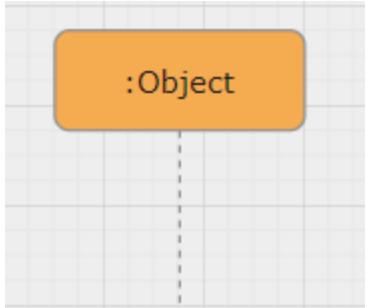
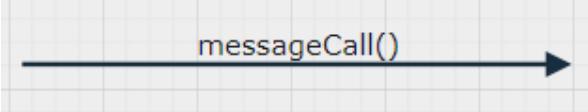
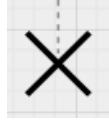
These diagrams serve as the UML domain model diagram and class diagram for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. For this purpose, we need to model a system that incorporates online selling, online buying, agri-info portal, discussion forum, Resource Manager and financial reporting. These roles have been assigned to specific classes.

Sequence Diagram

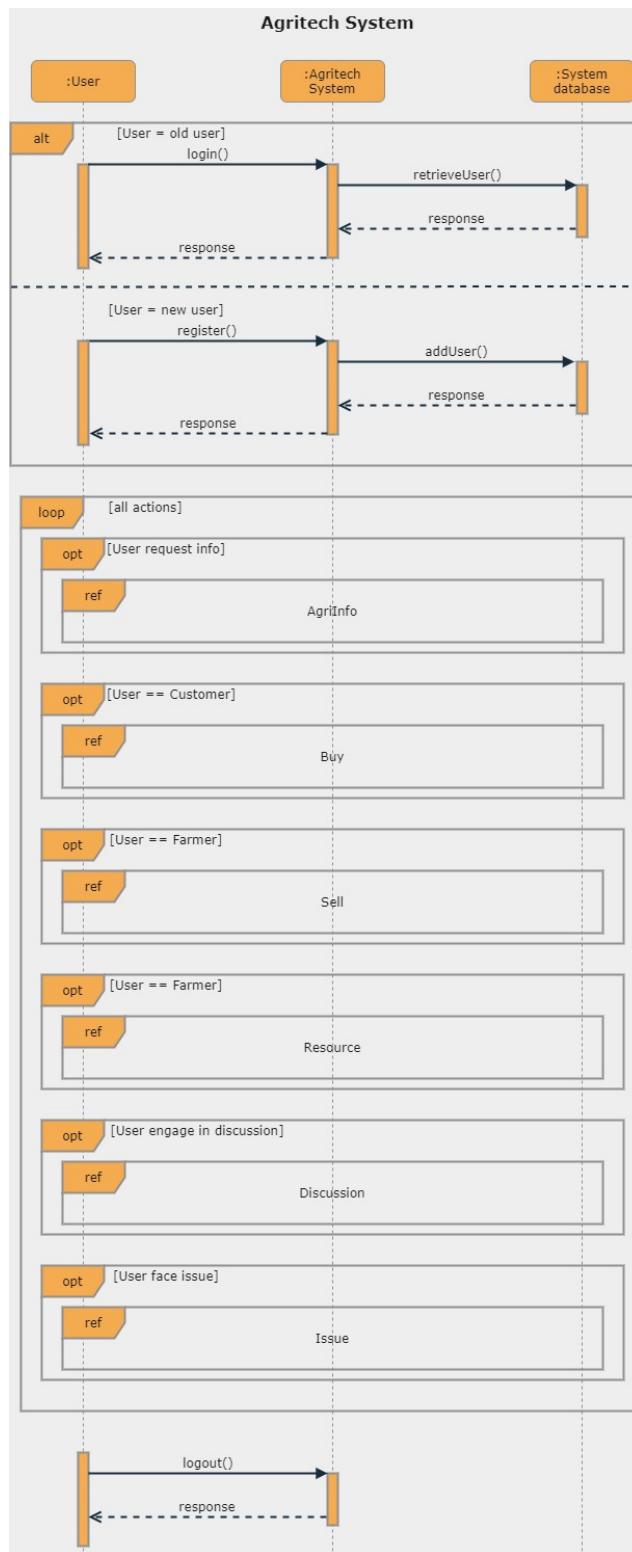
Aim

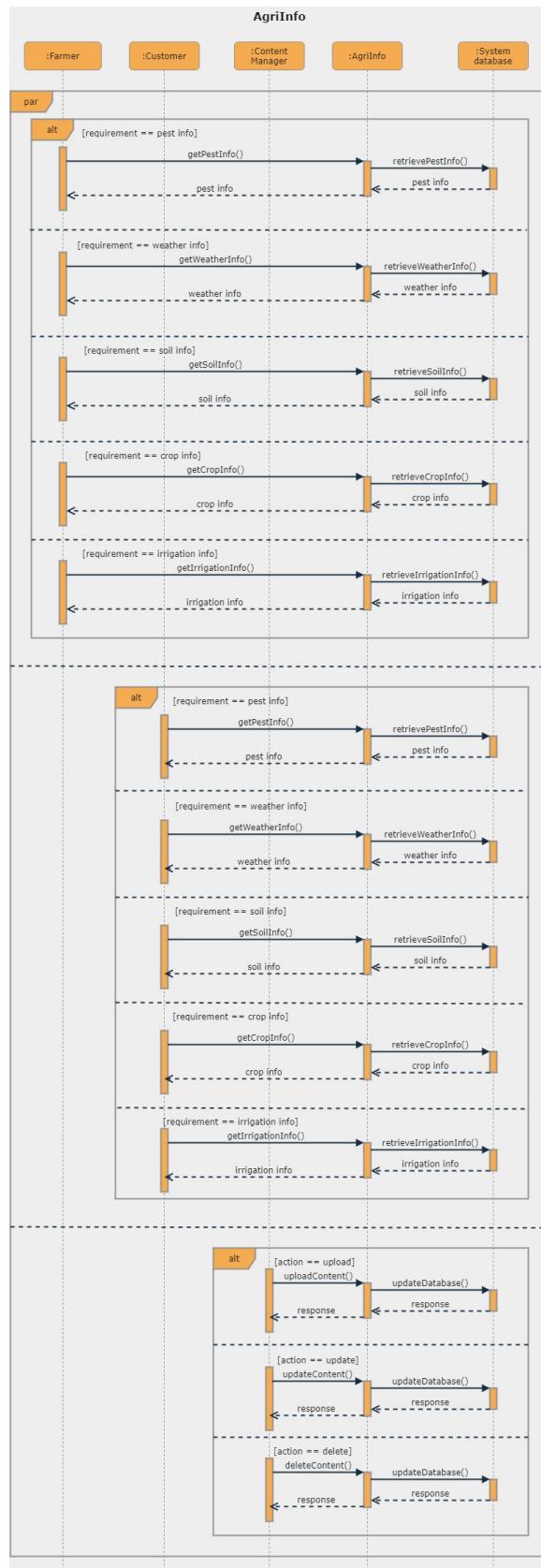
To map the core capabilities of the Agritech System to a UML Sequence Diagram.

UML Notations for Sequence Diagram

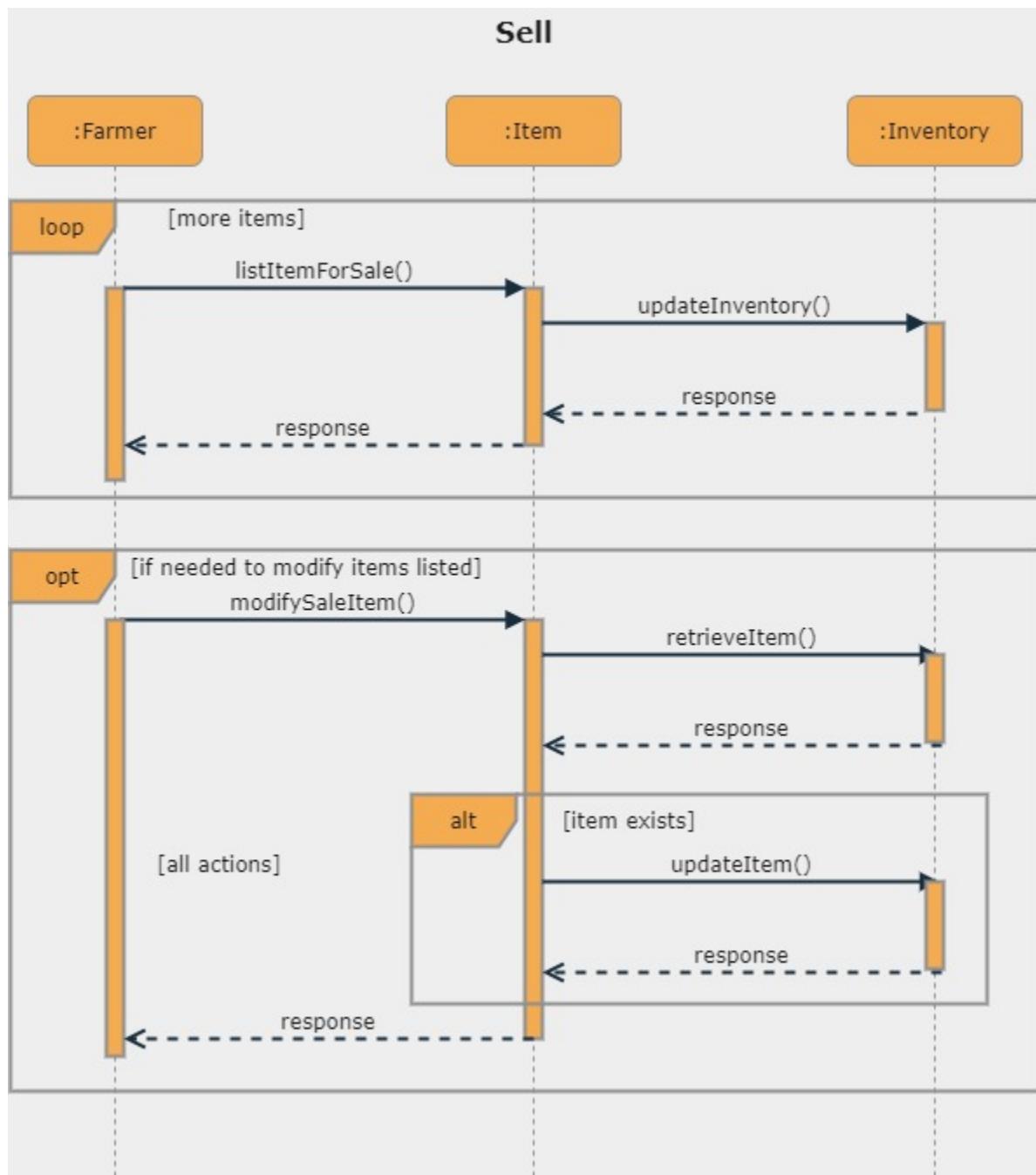
Notation	UML
Object Lifeline	
Frame <ul style="list-style-type: none">• alt• opt• loop• ref	
Message Call	
Message Return	
Destruction	
Activation	

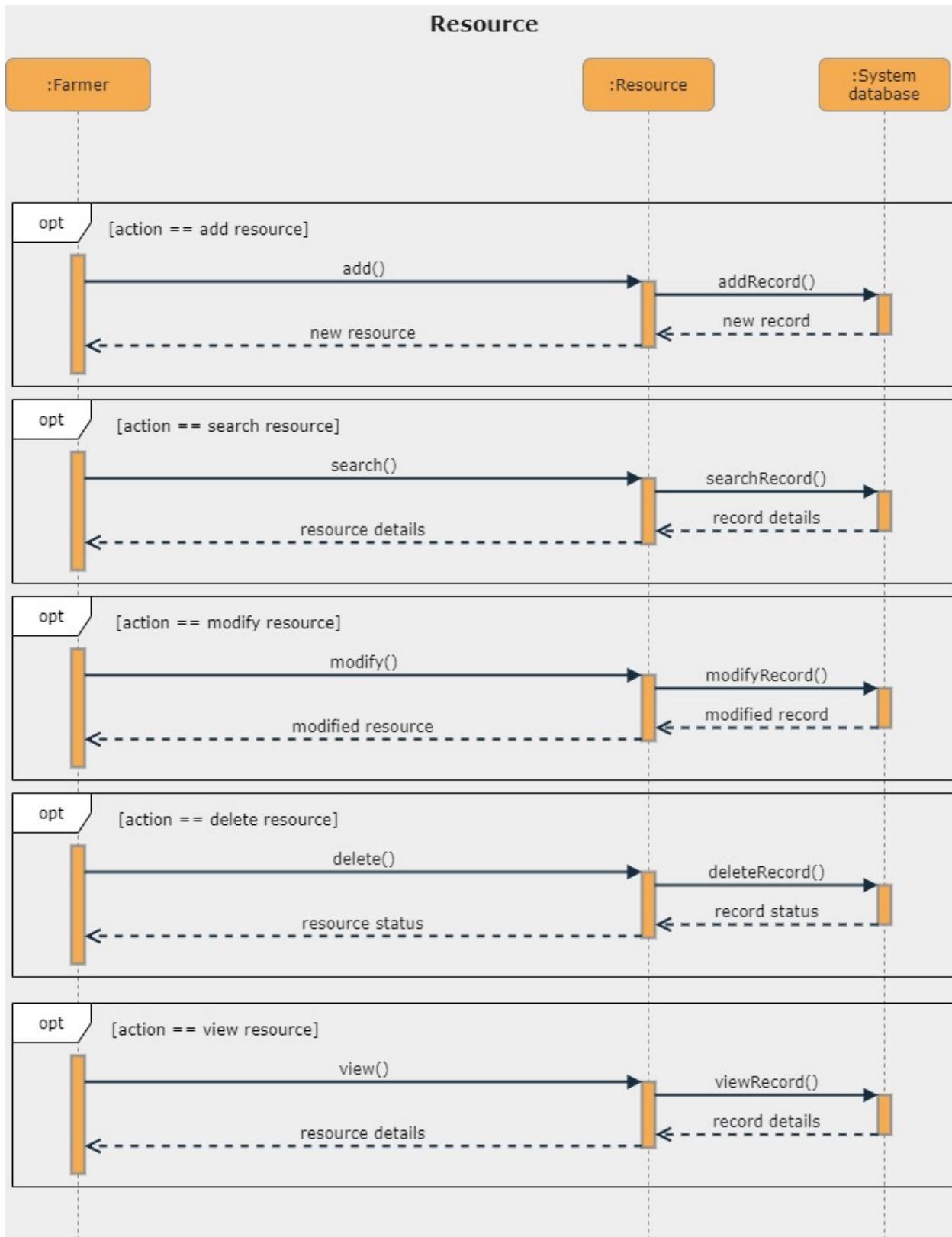
Sequence Diagrams

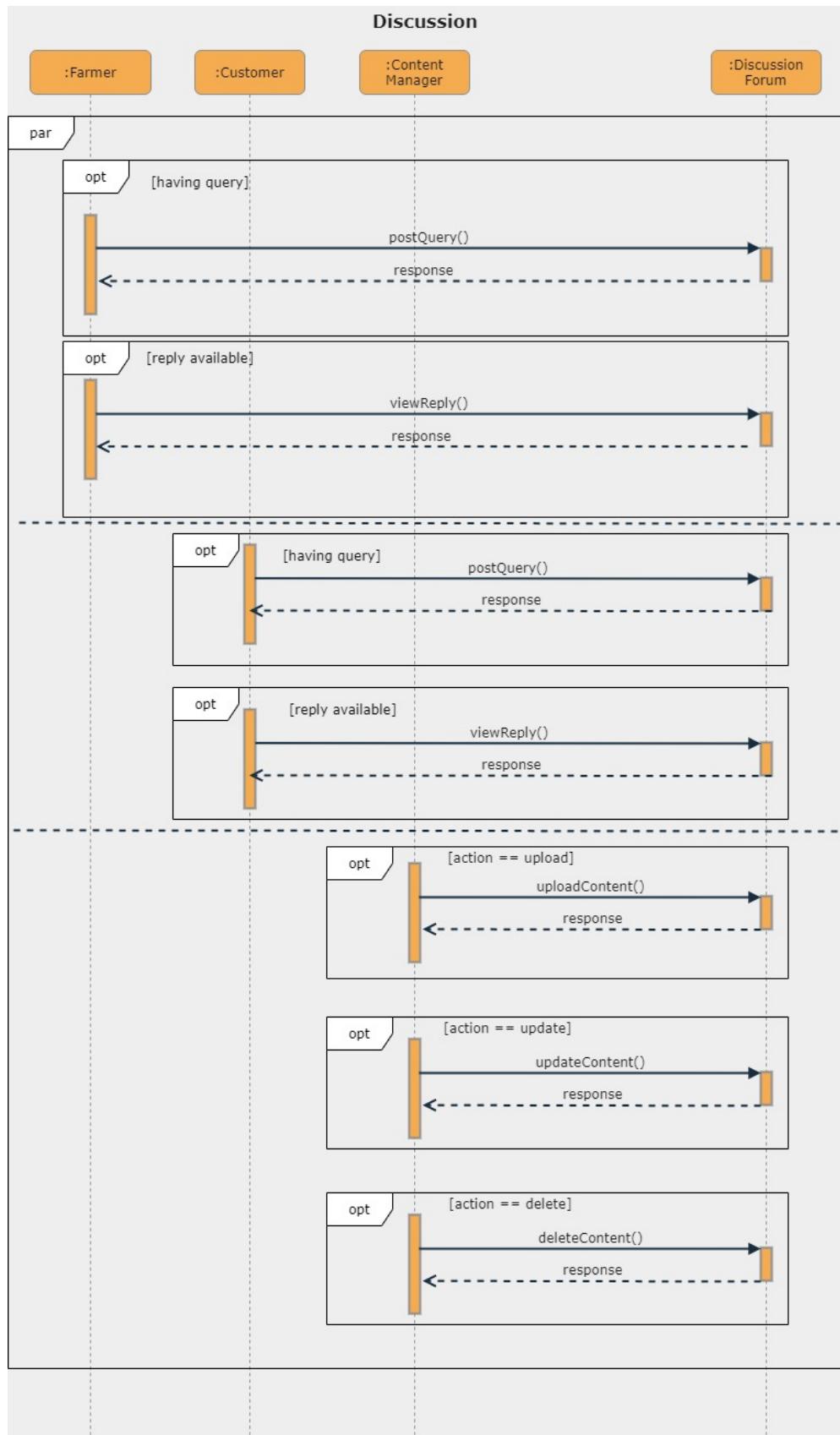


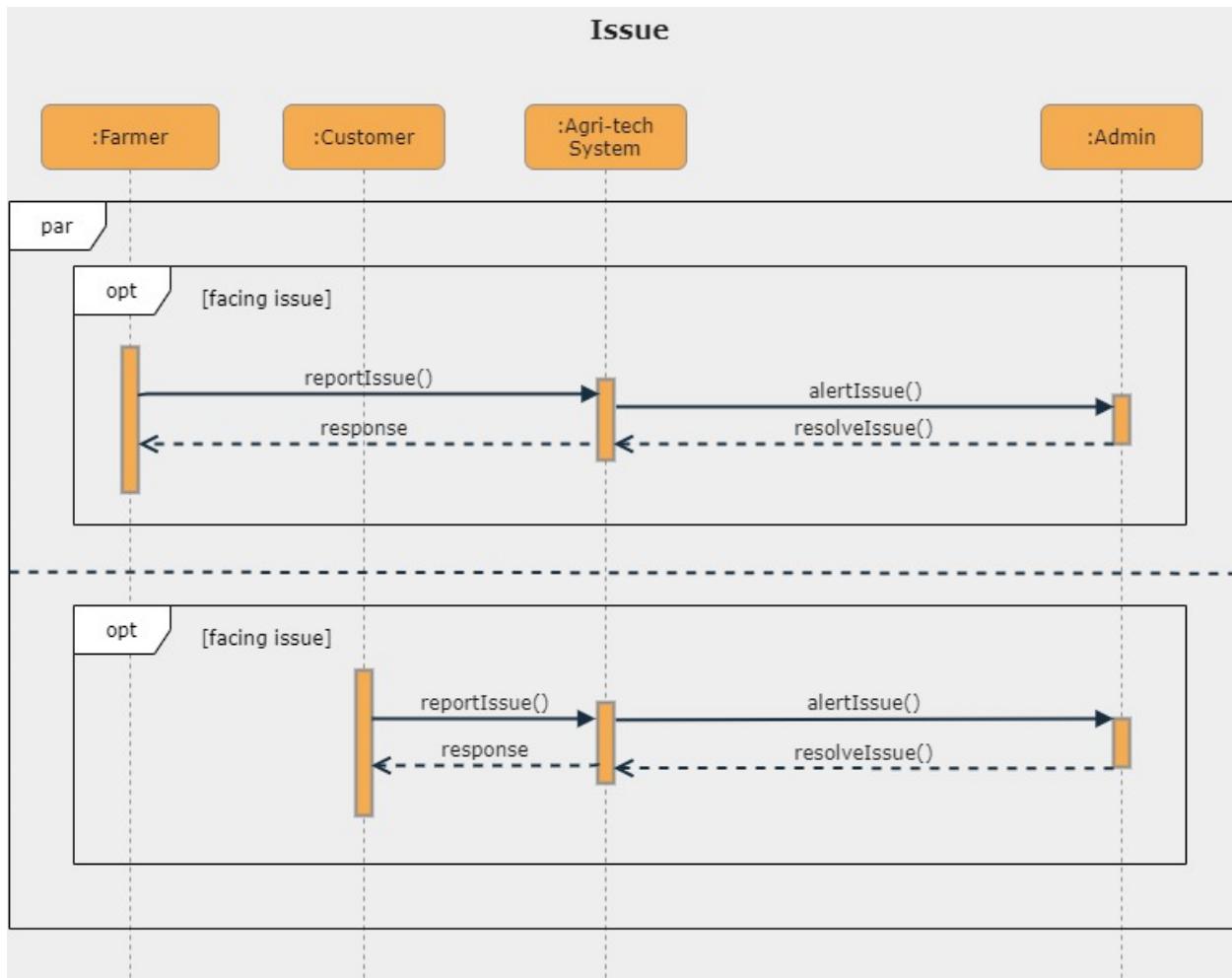












Documentation

These diagrams serve as the UML sequence diagrams for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. For this purpose, we need to model a system that incorporates online selling, online buying, agri-info portal, discussion forum, Resource Manager and financial reporting. These roles follow certain sequences and have been represented using the sequence diagram.

State Machine and Activity Diagram

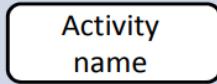
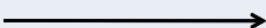
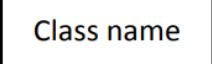
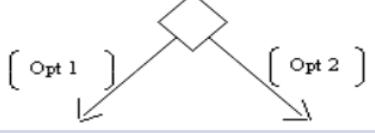
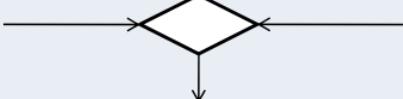
Aim

To map the core capabilities of the Agritech System to a UML State MachineDiagram and Activity Diagram.

UML Notations for State Machine

Name	Notation	Description
State		Description of a specific "time span" in which an object finds itself during its "life cycle". Within a state, activities can be executed by the object.
Transition		State transition e from a source state S to a target state T
Initial state		Start of a state machine diagram
Final state		End of a state machine diagram
Terminate node		Termination of an object's state machine diagram

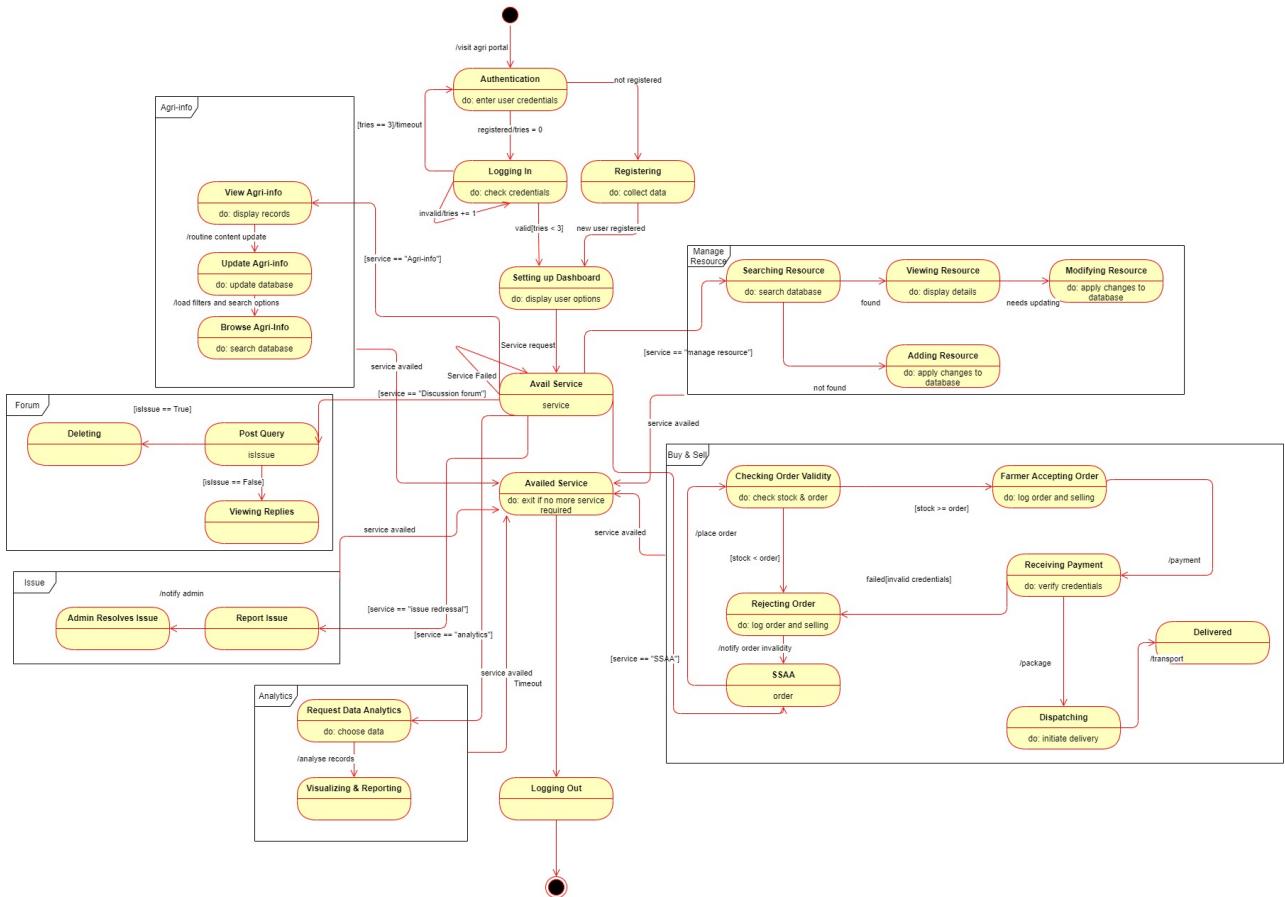
UML Notations for Activity Diagram

Description	Symbol
Activity : Is used to represent a set of actions	
A Control Flow: Shows the sequence of execution	
An Object Flow: Shows the flow of an object from one activity (or action) to another activity (or action).	
An Initial Node: Portrays the beginning of a set of actions or activities	
A Final-Activity Node: Is used to stop all control flows and object flows in an activity (or action)	
An Object Node: Is used to represent an object that is connected to a set of Object Flows.	
A Decision Node: Is used to represent a test condition to ensure that the control flow or object flow only goes down one path	
A Merge Node: Is used to bring back together different decision paths that were created using a decision-node.	
A Fork Node: Is used to split behavior into a set of parallel or concurrent flows of activities (or actions)	
A Join Node: Is used to bring back together a set of parallel or concurrent flows of activities (or actions).	
A Swimlane : A swimlane is a way to group activities performed by the same actor on an activity diagram or to group activities in a single thread	

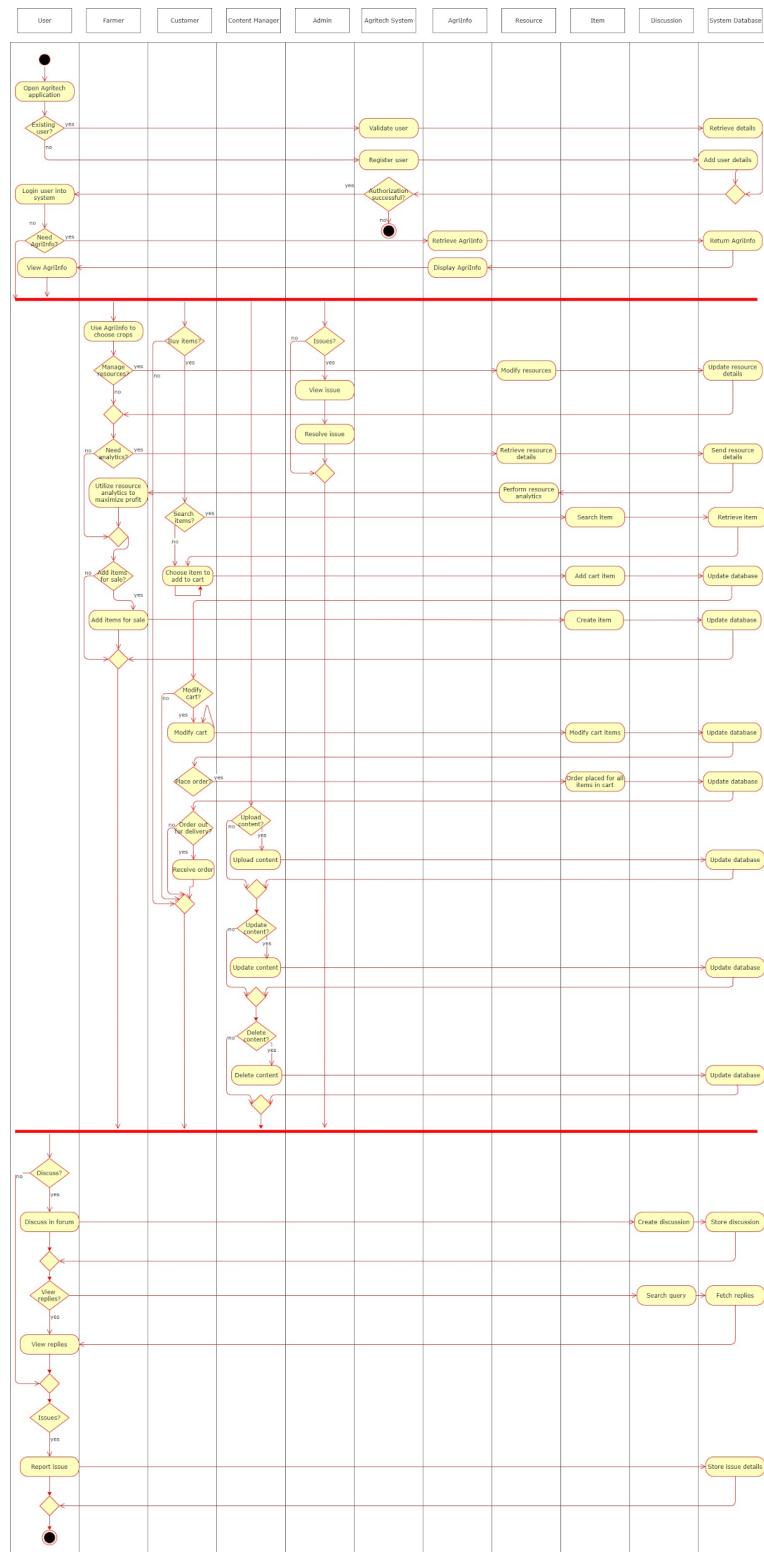
Identification of states

- Authentication
 - Logging In
 - Registering
 - Setting up Dashboard
 - Availing Service
 - Logout
- Manage Resources
 - Searching Resource
 - Adding Resource
 - Viewing Resource
 - Modifying Resource
- Buying & Selling - SSAA
 - Checking Order Validity
 - Farmer Accepts Order
 - Receiving Payment
 - Dispatching
 - Delivered
- Discussion Forum
 - Posting Query
 - Viewing Replies
 - Deleting
- Issue Redressal
 - Report Issue
 - Admin resolves issue
- Data Analytics
 - Request analytics
 - Visualizing & Reporting
- Agri-Info & Expert Column
 - Viewing Agri-info
 - Update Agri-info
 - Search Agri-info

State Machine Diagram



Activity Diagram



Documentation

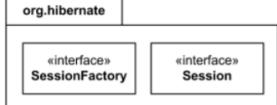
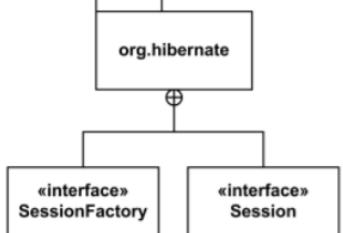
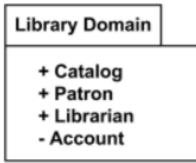
These diagrams serve as the UML state machine diagram and activity diagram for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. For this purpose, we need to model a system that incorporates online selling, online buying, agri-info portal, discussion forum, Resource Manager and financial reporting. These roles follow certain sequences which can be effectively represented by states and activities diagrammatically in UML.

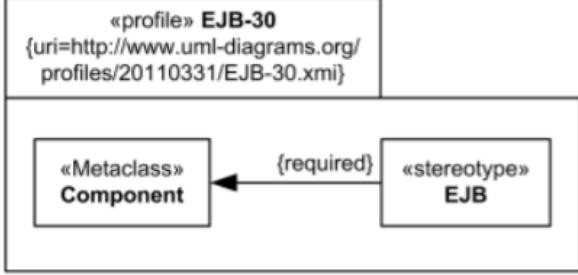
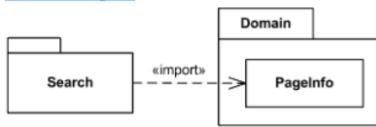
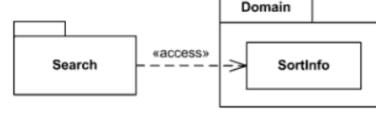
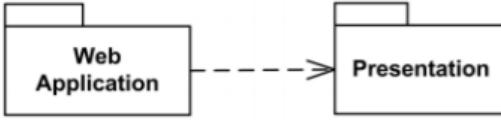
Package Diagram

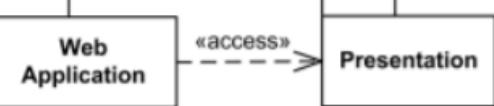
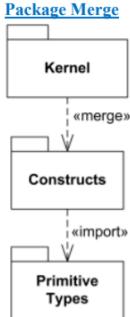
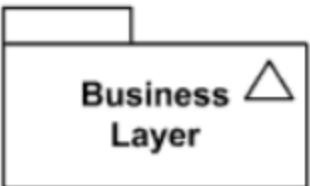
Aim

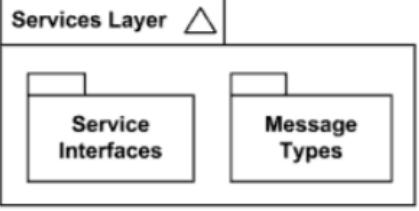
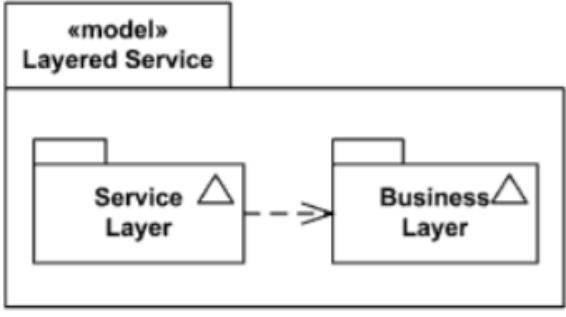
To map the core capabilities of the Agritech System to an UML package diagram.

UML Notations for Package Diagram

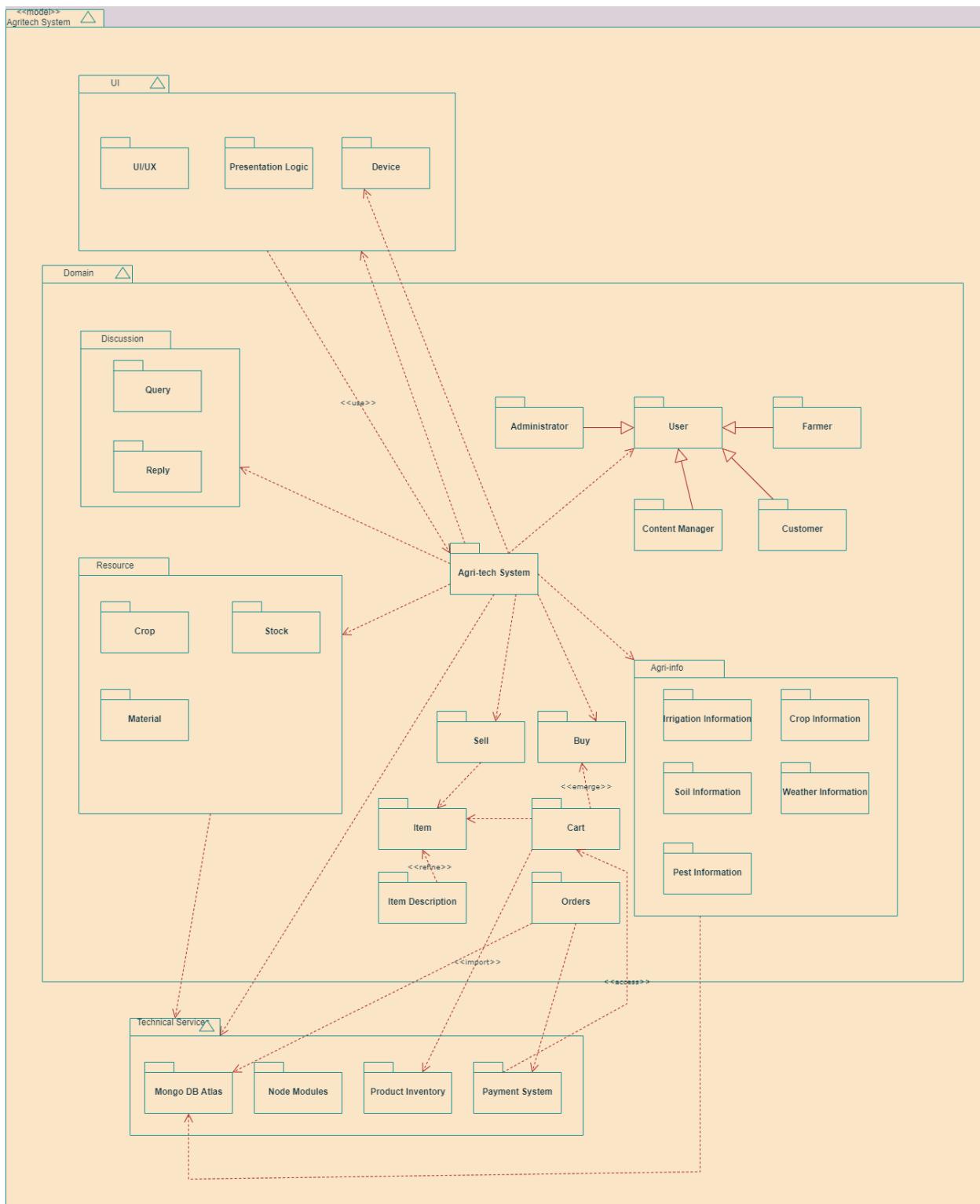
Notation	Description
<p>Package</p>  <p>org.hibernate</p> <p>Package org.hibernate</p>	<ul style="list-style-type: none">• Package is a namespace used to group together elements that are semantically related and might change together• A package could be shown as a rectangle with a small tab attached to the left side of the top of the rectangle
 <p>«interface» SessionFactory «interface» Session</p> <p>Package org.hibernate contains SessionFactory and Session.</p>	<ul style="list-style-type: none">• Members of the package may be shown within the boundaries of the package• In this case the name of the package should be placed on the tab
 <p>«interface» SessionFactory «interface» Session</p> <p>Package org.hibernate contains interfaces SessionFactory and Session.</p>	<ul style="list-style-type: none">• Members of the package may be shown outside of the package by branching lines from the package to the members• A plus sign (+) within a circle is drawn at the end attached to the namespace (package)• This notation for packages is semantically equivalent to composition (which is shown using solid diamond)
 <p>+ Catalog + Patron + Librarian - Account</p> <p>All elements of Library Domain package are public except for Account.</p>	<ul style="list-style-type: none">• If an element that is owned by a package has visibility, it could be only public(+) or private(-) visibility• Protected or package visibility is not allowed• The visibility of a package element may be indicated by preceding the name of the element by a visibility symbol

<p><u>Package URI Attribute</u></p>  <p>EJB Profile shown as a package with URI attribute.</p>	<ul style="list-style-type: none"> • Package has an optional URI attribute which serves as a unique identifier of the package • This attribute was introduced in UML 2.4 mostly to support exchange of profiles using XMI. UML 2.4 requires this URI attribute to follow the rules and syntax of the IETF URI specification RFC 2396 (while the more recent version of the URI syntax RFC 3986 released in 2005 rendered the RFC 2396 obsolete) • The URI attribute of a package may be rendered in the form {uri=} after the package name
<p><u>Element Import</u></p>  <p>Public import of PageInfo element into Search namespace from Domain package.</p>	<ul style="list-style-type: none"> • If element import is public, the imported element will be added to the namespace and made visible outside the namespace • Keyword «import» indicates public element import
 <p>Private import of SortInfo element into Search namespace from Domain package.</p>	<ul style="list-style-type: none"> • If element import is private, the imported element will be added to the namespace but will not be visible outside the namespace • Keyword «access» indicates private element import
<p><u>Package Import</u></p>  <p>WebApplication imports Presentation package with default public visibility.</p>	<ul style="list-style-type: none"> • Package import is shown using a dashed arrow with an open arrowhead from the importing namespace to the imported package • By default, the value of visibility is public, so it is the same as «import»

 <p>Public import of Domain package into WebApplication.</p>	<ul style="list-style-type: none"> If the package import is public, the imported elements will be added to the namespace and made visible outside the namespace Keyword «import» indicates public package import
 <p>Private import of Presentation package into WebApplication.</p>	<ul style="list-style-type: none"> If the package import is private, the imported elements will be added to the namespace but will not be visible outside the namespace Keyword «access» indicates private package import
 <p>UML Kernel package merges Constructs package which imports Primitive Types.</p>	<ul style="list-style-type: none"> A package merge is a directed relationship between two packages that indicates that content of one package is extended by the contents of another package Package merge is shown using a dashed line with an open arrowhead pointing from the receiving package to the merged package Keyword «merge» is shown near the dashed line
<h3><u>Model</u></h3>  <p>Business layer model</p>	<ul style="list-style-type: none"> Model is a package which captures a view of a system View is some abstraction of the system describing only those aspects of the system that are relevant to the purpose of the model, at the appropriate level of detail, describing logical or behavioral aspects of the system to a certain category of readers Model is notated using the ordinary package symbol (a folder icon) with a small triangle in the upper right corner of the large rectangle

 <p>Service Layer model contains service interfaces and message types</p>	<p>If contents of the model are shown within the large rectangle, the triangle may be drawn to the right of the model name in the tab</p>
 <p>Stereotyped model Layered Service</p>	<p>Model could be notated as a package with the keyword «model» placed above the name of the model</p>

Package Diagram



Documentation

The UML package diagram represents the various package modules for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. For this purpose, we need to model a system that incorporates online selling, online buying, agri-info portal, discussion forum, Resource Manager and financial reporting. These roles are assigned to specific packages, each interacting with others to effectively represent the Agritech system diagrammatically in UML.

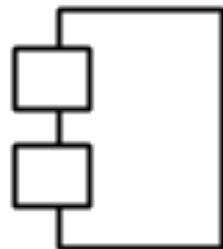
Component Diagram and Deployment Diagram

Aim

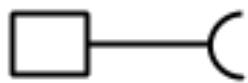
- To model the core capabilities of the Agritech System onto a UML component diagram to:
 - visualize the organization and relationships among system components
 - construct executables using forward and reverse engineering
 - represent the implementation perspective
 - reflect grouping of different design elements of the system

- To model the core capabilities of the Agritech System onto a UML deployment diagram to:
 - demonstrate the configuration of run time processing nodes
 - specify the distribution of components
 - capture the topology of the system hardware
 - identify performance bottlenecks, if any

UML Notations for Component Diagram



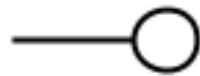
Component symbol used
for the main and sub
components



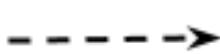
Ports sit on the boundary of the
component; can have a required (as
shown here) or provided interface



Required interface; consumes data



Provided interface; supplies data

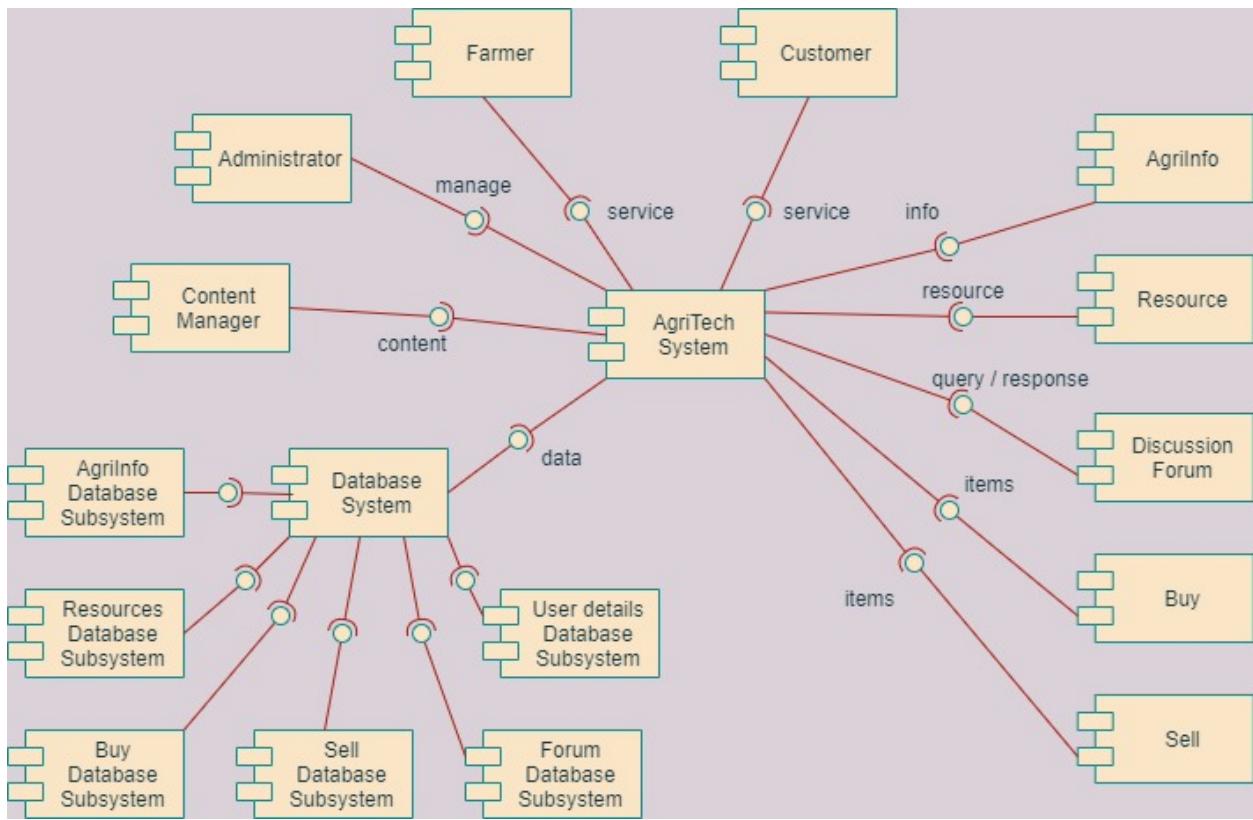


Dependency arrow; used to show
dependency between interfaces

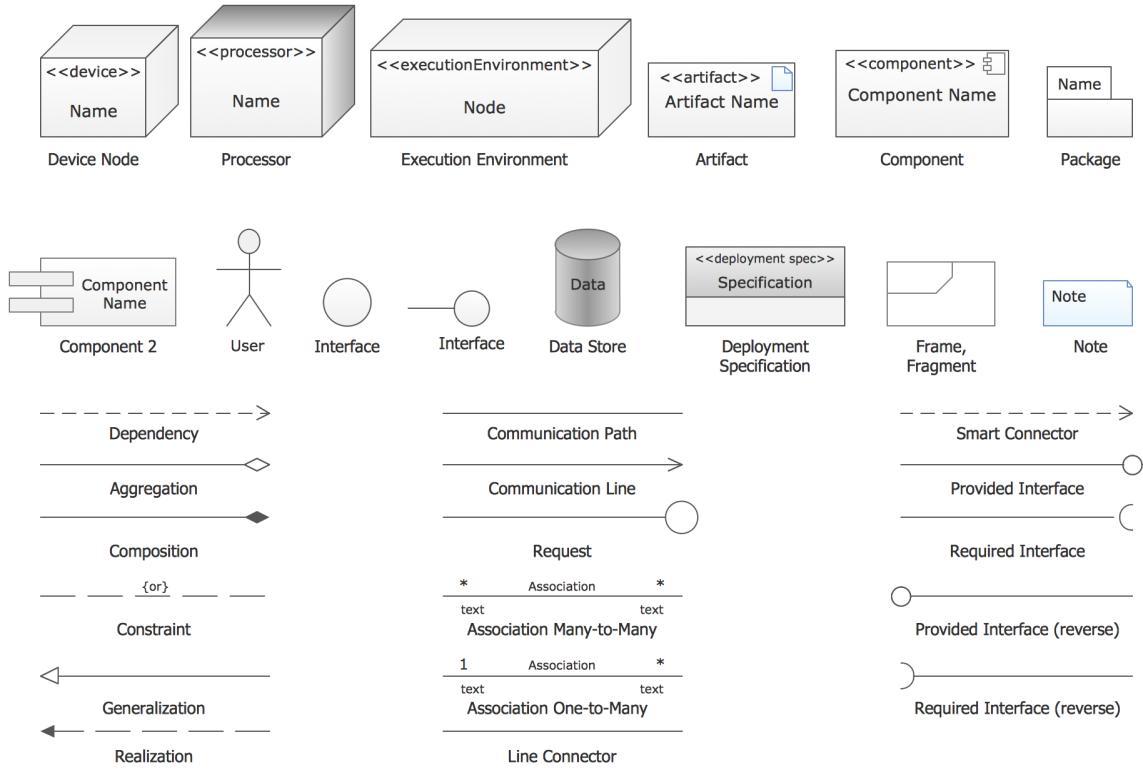
List of components identified from class and use-case diagram

- AgriTech System
- AgriInfo
- Resource
- Discussion Forum
- Buy
- Sell
- Farmer
- Customer
- Administrator
- Content Manager
- Database System
 - AgriInfo Database Subsystem
 - Resources Database Subsystem
 - Buy Database Subsystem
 - Sell Database Subsystem
 - Forum Database Subsystem
 - User Details Database Subsystem

Component Diagram



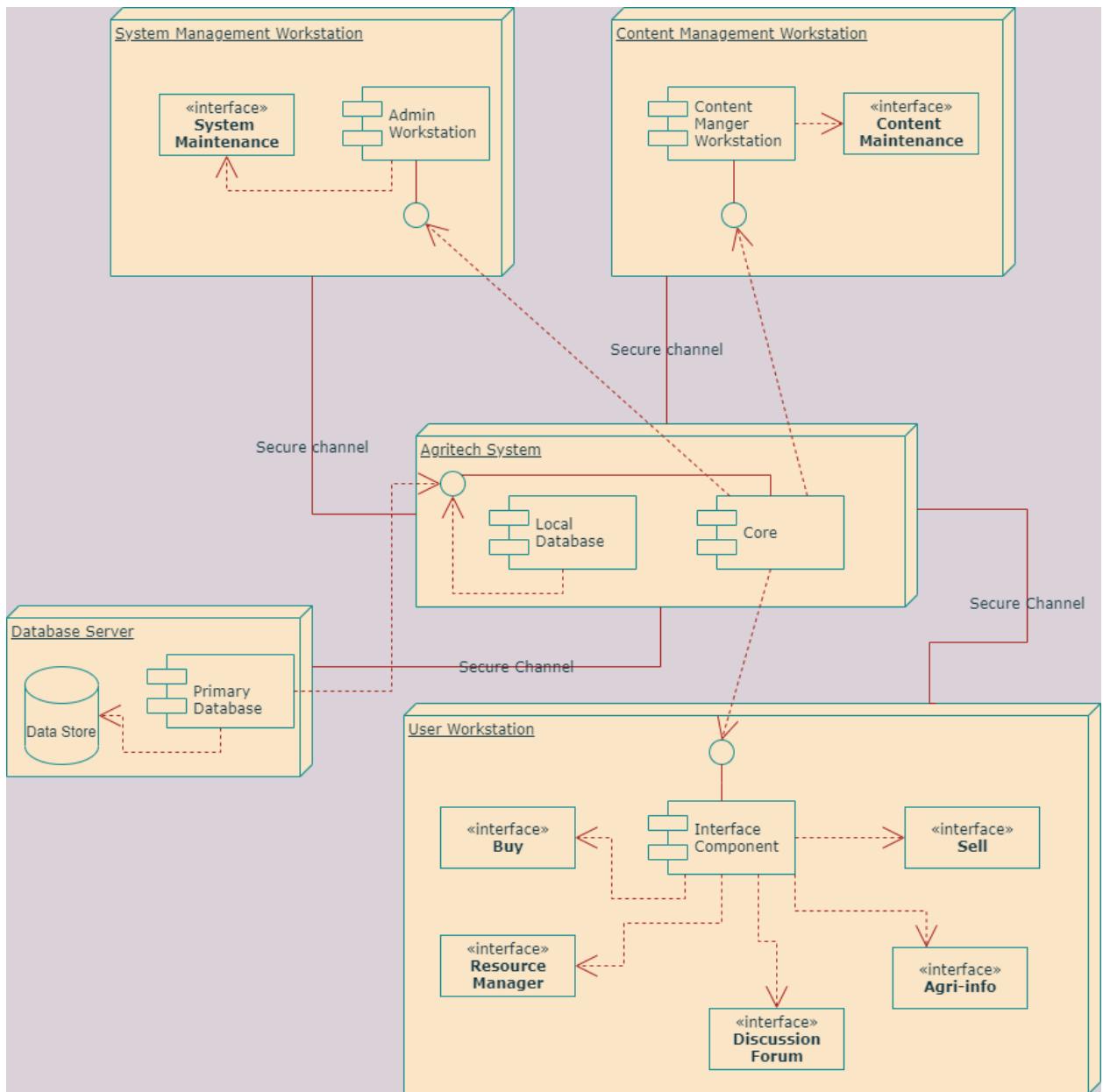
UML Notations for Deployment Diagram



List of nodes identified from Component diagram

- Agritech core system server
- Primary database server
- System management workstation
- Content management workstation
- User workstation

Deployment Diagram



Documentation

The UML package diagram represents the various package modules for the proposed Agri-tech system. The Agritech system should be an all-in-one software that helps farmers be more self-reliant, organized, sustainable and successful. The software should provide everything you need to run a more efficient and productive farm. For this purpose, we need to model a system that incorporates online selling, online buying, agri-info portal, discussion forum, Resource Manager and financial reporting. These roles are assigned to specific components, each interacting with others to effectively represent the Agritech system diagrammatically in UML. Based on these components, nodes were identified to develop a deployment diagram to specify their distribution configured as run time processing nodes, capture the topology of the system hardware and identify performance bottlenecks, if any.