

POLITECNICO

MILANO 1863

Software Engineering 2 project

Academic Year 2021-2022

DREAM - Data-dRiven PrEdictive FArMing in Telengana

Requirement Analysis and Specification Document

Version 0.2 – 18/12/2021

Authors:

Xu Qiongjie

Zhang Rui

Hu Zhijun

Professor:

Di Nitto Elisabetta

Content

1. INTRODUCTION	4
1.1 Purpose	4
1.1.1 Purpose	4
1.1.2 Goals	4
1.2 Scope	5
1.2.1 World phenomena	5
1.2.2 Shared phenomena.....	5
1.3 Definitions, Acronyms, Abbreviations	6
1.3.1 Definition	6
1.3.2 Acronyms	7
1.3.3 Abbreviations.....	7
1.4 Revision history	7
1.5 Reference Documents.....	7
1.6 Document Structure	8
2. OVERALL DESCRIPTION	9
2.1 Product perspective.....	9
2.1.1 Scenarios.....	9
2.1.2 Class Diagram.....	12
2.1.3 State Charts.....	13
2.2 Product functions	14
2.3 User characteristics	19
2.4 Assumptions, dependencies and constraints.....	20
2.4.1 Domain Assumptions	20
2.4.2 Dependencies.....	21
3. SPECIFIC REQUIREMENTS:.....	22
3.1 External Interface Requirements.....	22
3.1.1 User Interfaces.....	22
3.1.2 Hardware Interfaces.....	22
3.1.3 Software Interfaces.....	22
3.1.4 Communication Interfaces	23

3.2 Functional Requirements.....	23
3.3 Performance Requirements	46
3.4 Design Constraints	47
3.4.1 Standards compliance	47
3.4.2 Hardware limitations.....	47
3.4.3 Any other constraint	47
3.5 Software System Attributes	47
3.5.1 Reliability.....	47
3.5.2 Availability	47
3.5.3 Security.....	47
3.5.4 Maintainability	48
3.5.5 Portability.....	48
4. FORMAL ANALYSIS USING ALLOY	49
4.1.1 Signatures.....	50
4.1.2 Facts	52
4.1.3 Assertions	55
4.1.4 Analysis results	56
5. EFFORT SPENT:	57
6. REFERENCES	58

1. INTRODUCTION

1.1 Purpose

1.1.1 Purpose

Agriculture plays a vital role in India's economy. However, with food demand increasing, climate change and COVID-19 pandemic, agriculture in Telengana India is facing severe challenges, like unstable food supply chains, vulnerabilities of marginalized communities and smallholders.

Telengana's government wants to build anticipatory governance models for food systems using digital public goods and community-centric approaches to strengthen data-driven policy making in Telengana.

This document is addressed to the developers who have to implement the requirements and could be used as an agreement among policy makers, farmers and agronomists.

1.1.2 Goals

- G.1: Allow policy makers to identify those farmers who are performing well or badly.
- G.2: Allow policy makers to understand whether the steering initiatives carried out by agronomists produce significant results. ¹
- G.3: Allow farmers to visualize data relevant to them.
- G.4: Allow farmers to provide information about their production.
- G.5: Allow farmers to ask for help and suggestion from agronomists.
- G.6: Allow farmers to create discussion forums with the other farmers.
- G.7: Allow agronomists to receive requests for help and answer to these requests.
- G.8: Allow agronomists to visualize data concerning weather forecasts in the responsible area and the best performing farmers in the responsible area.
- G.9: Allow agronomists to create and modify a daily plan to visit farms in the responsible area.
- G.10: Allow agronomists to confirm the execution of the daily plan at the end of each day or specify the deviations from the plan.

¹ Because "with the help of good farmers" is not explicitly mentioned in the work of agronomists, this part is deleted for consistency.

1.2 Scope

DREAM is an effective application which aims to settle for various queuing problems faced by farmers, policy makers and agronomists.

Between farmers and policy makers, it allows policy makers identify farmers who are good at dealing with challenges or who need to help. Between farmers and agronomists, it allows they can learn each other's requests and solutions quickly, accurately and visually. Between policy makers and agronomists, it allows policy makers evaluate ability of agronomists.

The system also contains functions creates discussion forums with the other farmers, and allows users access information of weather, crop, water irrigation system and soil.

1.2.1 World phenomena

- WP.1: Farmers know their farms' location.
- WP.2: Weather forecasts are available by Telengana.
- WP.3: Farmers grow specific crops.
- WP.4: Farmers face a problem.
- WP.5: Farmers carry out suggestions.
- WP.6: Farmers know the produced amount per product they plant.
- WP.7: Water irrigation system monitors the amount of water used by each farmer.
- WP.8: Sensors deployed on the territory measure humidity of soil.
- WP.9: the policy makers know how to rate performance of farmers.
- WP.10: the policy makers understand the initiatives from Agronomists.
- WP.11 the policy makers make policies.
- WP.12: Agronomists know their responsible areas.
- WP.13: Agronomists research and develop solutions.
- WP.14: Agronomists visit each farm at least twice per year.
- WP.15: Each farmer's farms are in a same area in charge of one agronomist.

1.2.2 Shared phenomena

- SP.1: The policy makers visualize the information about farmers. (**world**)
- SP.2: The policy makers visualize the information about geospatial information. (**world**)
- SP.4: The policy makers rate performance of farmer in the system. (**world**)
- SP.5: The policy makers visualize daily plan from Agronomists. (**world**)
- SP.6: Agronomists insert the area they are responsible of. (**world**)
- SP.7: Agronomists receive information about farmers' requests for help. (**machine**)
- SP.8: Agronomists answer to farmers' requests. (**world**)
- SP.9: Agronomists check data concerning weather forecasts. (**world**)
- SP.10: Agronomists check the best performing and under-performing farmers in the area.

(world)

SP.11: Agronomists visualize and update a daily plan to visit farms in the area. (world)

SP.12: Agronomists confirm the execution of the daily plan at the end of each day or specify the deviations from the plan. (world)

SP.13: The farmers check local weather forecasts. (world)

SP.14: The farmers search for personalized suggestions concerning specific crops to plant or specific fertilizers to use based on the type of product. (world)

SP.15: The farmers insert the data about the production (types of products, produced amount per product) into the system. (world)

SP.16: The farmers insert the problems they face into the system. (world)

SP.17: The system shows the answer to the farmer's request. (machine)

SP.18: The farmers create a discussion forum in the system. (world)

SP.19: The system shows the discussion forum to all farmers. (machine)

SP.20: The farmers comment a discussion in the system. (world)

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definition

- Marginalized communities: Socially excluded groups of people for different reasons, such as age, physical or mental disabilities, economic status, access to education, or live in isolated places or depressed areas.
- Smallholders: Smallholders are usually farms supporting a single family with a mixture of cash crops and subsistence farming.
- Challenge: It may cause exceptionable influence from human or natural factors.
- Telengana: Telengana is the 11th largest state in India with a geographical area of 112,077 km² and 35,193,978 residents (data from 2011)
- DREAM System (or "The System"): refers to the whole system to be developed.
- DREAM Services (or "Services"): refers to the functionalities offered by the DREAM System, such as daily plan management mechanism and the visualization service.
- DREAM Application (or "The Application"): refers to the application that makes DREAM Services available everywhere. In this document, this term is intentionally used in a generic way. How the application will actually be delivered (e.g., as a web app or as a mobile app) will be defined later in the Design Document of DREAM.
- Geospatial information: geospatial information is data about the water irrigation system concerning the amount of water used by each farmer, sensors deployed on the territory and measuring the humidity of soil, and weather forecasts
- Agricultural Product Reference System: A system helps governments and development agencies coordinate their activities to improve and develop agriculture. It also conducts research, provides technical assistance to projects, operates educational and training programs, and collects data on agricultural output, production, and development.

1.3.2 Acronyms

- WP: World phenomena
- SP: Shared phenomena
- G: Goal
- RASD: Requirement Analysis and Specification Document.
- UML: Unified Modelling Language.
- GPS: Global Positioning System.

1.3.3 Abbreviations

- G.i: i-th goal.
- WP.i: i-th world phenomena.
- SP.i: i-th shared phenomena.
- R.i: i-th requirement.
- D.i: i-th domain assumption.
- UC.i: i-th use case.

1.4 Revision history

Vision	Date	Content
0.1	30/11/21	Finish Section1
0.2	18/12/21	Finish Section2 and modify Section1 because of the decision we make about some ambiguous parts.
0.3	23/12/21	Finish Section3, adjust the details of Section1 and Section2.
0.4	23/12/21	Finish Section 4, adjust the details of Section 1 ~ 3

1.5 Reference Documents

- Project assignment specification document.
- ISO/IEC/IEEE 29148 - Systems and software engineering.
- Course slides on WeBeeP

1.6 Document Structure

This document is presented as it follows:

1. **Introduction:** contains a brief description of problems to be solved, purpose of system, and goals which the application must reach. Meanwhile, it describes scope, which gives more details about goals and shows world or shared phenomena.
2. **Overall Description:** gives a general description of the system, focusing on its functions and constraints. Moreover, it provides the domain assumptions of the analyzed world.
3. **Specific Requirements:** explains in detail the functional and non-functional requirements. It lists the possible interactions with the system in the form of scenarios, use cases and sequence diagrams.
4. **Formal Analysis Using Alloy:** contains the Alloy model of some critical aspects of the system and an example of the generated world.
5. **Effort Spent:** keeps track of the time spent to complete this document.
6. **References:** lists the references used in this document.

2. OVERALL DESCRIPTION

2.1 Product perspective

2.1.1 Scenarios

- **Scenario 1: Farmer Registration**

Nita is very worried about the recent weather because there had not been any rain in the past month, which has seriously affected the crops she planted. To make matters worse, the lockdown caused by the COVID-19 pandemic caused prices on local farms to plummet. Therefore, when Nita knows about DREAM, she feels very happy. In fact, she immediately signs up as a farmer by filling out a simple form with her personal data including name, password, and phone number. After she accepts the Terms and Conditions of the service and the Privacy Statement, Nita completes the registration.

- **Scenario 2: Information Retrieval**

Kumar plants potato in Telengana and wants to know if tomorrow is a good time to fertilize his fields. Because of the government's publicity, he knows he could find some suggestions on DREAM. He opens DREAM and enters the information search interface.

The System displays a list of production types and the location of Kumar.

Kumar selects potato in the list, confirms that the location is the place where his potatoes are grown, and submits.

The System gives him the weather forecast for next few days which includes the information about temperature and rainfall. The System also shows that it is advised for application of fertilizer as well as earthing up based on the weather. In addition, the System suggests to do constant monitoring in potato because the infection of blight may occur due to high relative humidity.

Kumar takes the suggestions and decide to do the fertilization tomorrow.

- **Scenario 3: Problem Raising and Solving**

Balaiah is a farmer in Telengana. Recently, the cotton he plants is affected by bacterial leaf spot and the fungicide he used in the past is not very effective. Therefore, Balaiah decides to ask for help from others. He logs into DREAM System and creates a new message in which he describes the symptoms and signs of his crops. He confirms his

input and waits for a reply.

Sanjay, an agronomist, is responsible of Balaiah's area. One day, he receives a message submitted by Balaiah in DREAM System. Based on his previous research, combined with historical weather, irrigation system and soil moisture data, he enters details of suggestion about this question, including types, quantities and methods of use of pesticides. Then Sanjay confirms all contents and press down 'Submit' button.

Balaiah checks the suggestions and applies them to his cotton. After then, his cotton is recovered from the disease and Balaiah is very happy about the result. He opens DREAM again, navigates to the problem he opened before and gives Sanjay's answer a 9/10 score.

- **Scenario 4: Production report**

Reddy has a bumper paddy harvest recently. After selling out all his paddy, Reddy decides to report his production. He logs in DEARM, chooses production report interface, selects "paddy" in the list of production types, enters the production amount, the paddy plantation area and the plantation period, then submits his input.

- **Scenario 5: Discussion with other farmers**

Surinder grows red chilies on her 2.5-acre farm in Telangana and uses DREAM almost every day. Recently, the government published some policies to help the farmers. Surinder wants to know what others think about the government's compensation measures for red chilies. So, she opens DREAM and creates a discussion in the forum. She enters the title and a brief description of the discussion. In the description part, she attaches the news about these measures. She confirms her input and waits for other farmers' comments.

Anil sees Surinder's post when he browses the forum in DREAM. He reviews the information posted by Surinder and thinks the measures are good for them. He writes a comment in which he explains his opinion.

Surinder sees Anil's comment and agrees with him.

- **Scenario 6: Agronomist's registration**

Raj is a doctor of agronomist. After graduation, he devoted himself to solving the problems of agroecology encountered in the development of India. Due to his many influential articles in top publications, he was invited by Telangana to become a distinguished expert to help establish and improve agriculture of Telangana through DREAM System. Then, he was assigned to the region with the worst environment in

Telangana Province where extreme weather often occurred, which was a big challenge for him.

At the beginning, Raj has to register in DREAM System. He opens it, then presses down 'sign up' button of agronomist, and follows entering all parts, like e-mail, password etc. Moreover, as an agronomist, he must select the responsible area. Finally, he successfully registers as an agronomist in DREAM System.

- **Scenario 7: Manage daily plan**

Kamla is another agronomist aimed more at environment and is also invited by Telangana government as an expert.

Today she logs in DREAM system and it displays her responsible farmers, the time of last visit and farmers' performance.

She finds it has been a long time since last time she visited Kumar. Moreover, she has to pay more attention to Kumar who are under-performing. She logs in system and opens homepage. The system shows, weather forecast, irrigation system and sensor data of soil humidity in this area. So according to this information Kamla builds a daily plan, then enters title, date, name of farmers and content. She confirms contents and clicks 'submit' button.

After Kamla visited Kumar's farm, Kamla analysis the deviations from the plan and write deviation about this daily plan in system. She confirms the content and clicks 'confirm' button.

- **Scenario 8: Aamir participants in DREAM**

Aamir, a government office worker, was given the task by his supervisor, that was the first time he heard of "Dream". In fact, he immediately signs up as a policymaker by filling out a simple form with his name, email and password. The young man, finally, completes the registration by accepting the Terms and Conditions of the service and the Privacy Statement and finishes the step of sign up.

- **Scenario 9: Identify the performance of farmers**

Kumar is a new government employee. He feels full of enthusiasm about all new things. His superior sent him a task about identifying the performance of farmers. First, he logs in the Dream as the role of policymaker, and he can see the whole map of the Telangana. Then he clicks the interested area, and he can see a detail interface, the System will show farm's detail to him, including short-term and long-term forecasts, water irrigation information, territory and soil humidity information. After the farmers upload their production information on the harvest season, the System will display the relative

information to Kumar on the details interface, then he gives each farmer a score according to the policy standard. After calculation, Kumar enters the System, inputs the score results into the detail interface and confirms.

- **Scenario 10: Understand the incentive**

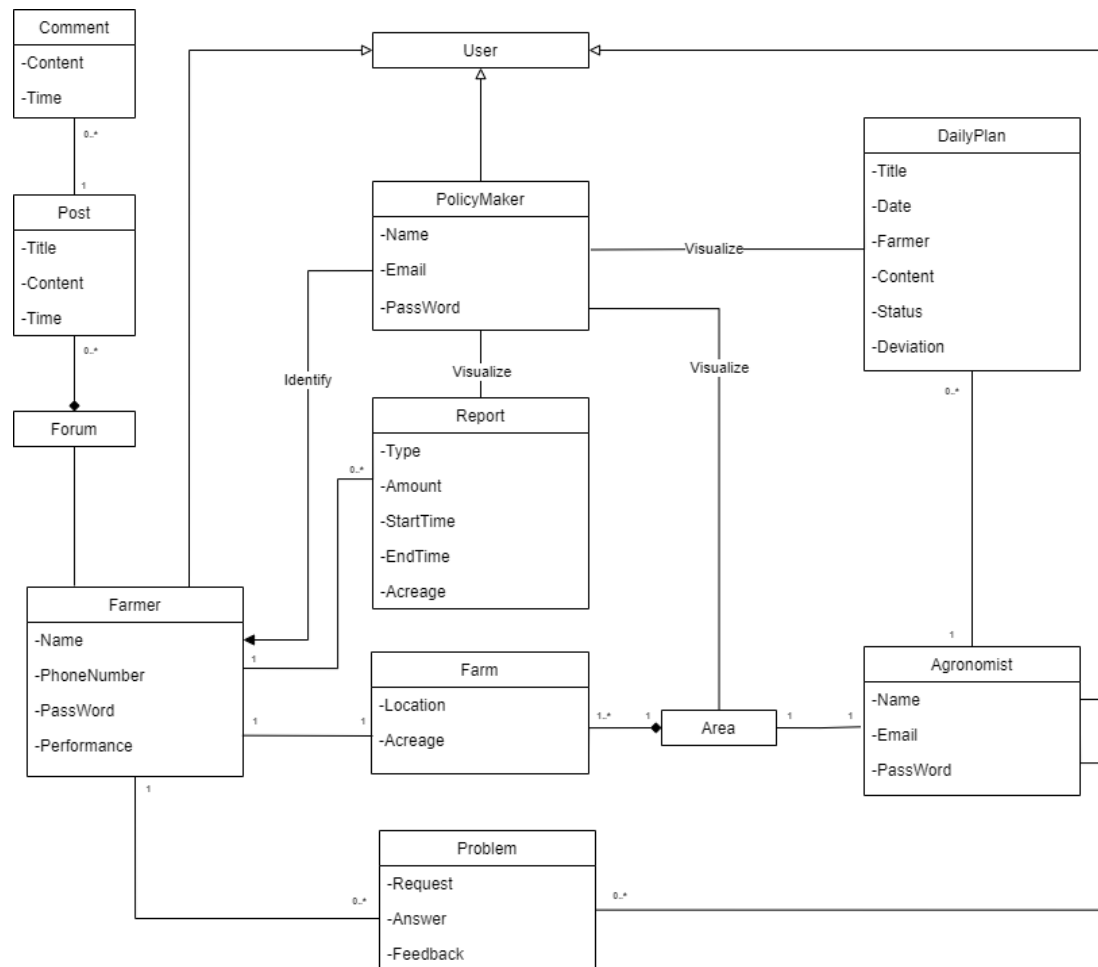
Kiran is a well-experience worker in the Telangana government, under the influence of COVID-19, the farm situation of Telangana got worse and worse, she worried about the farmers and wanted to help them. Then Kiran is assigned to the management part of agronomists. She needs to pay attention to the daily plans of agronomists.

First, she logins into the System as a policymaker, then she can see the whole map of Telangana, and she notices that one agronomist list. Now she wants to check the daily plan of every agronomist so that she clicks the name of them, and then the lists of daily plans show out. After the deviation were entered and confirmed by agronomist, Kiran needs to collect relevant information from details and feed it back to her superiors, and finally comprehensively evaluate the agronomists' daily plans, check if the productivity has been effectively improved due to the suggestions of agronomist.

2.1.2 Class Diagram

This section details the class diagram of the System. It provides a high-level view of the main concepts on which the system is focused.

The main elements in the class diagram are:

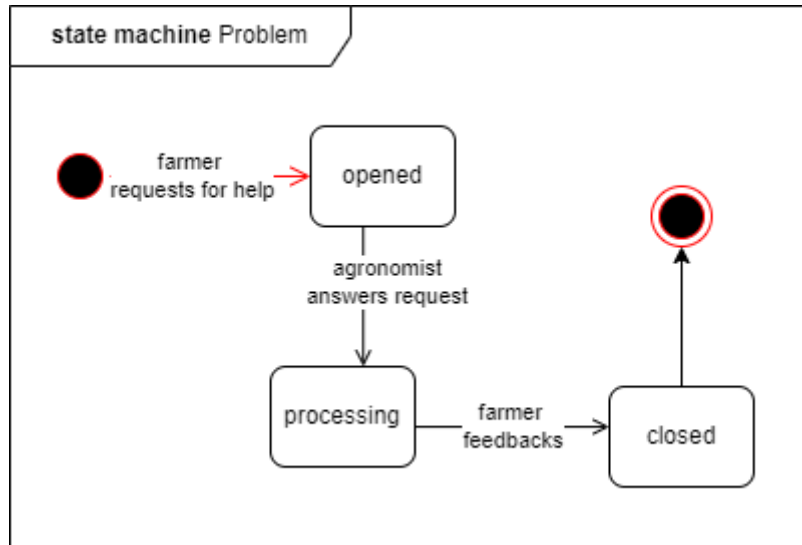


Class diagram

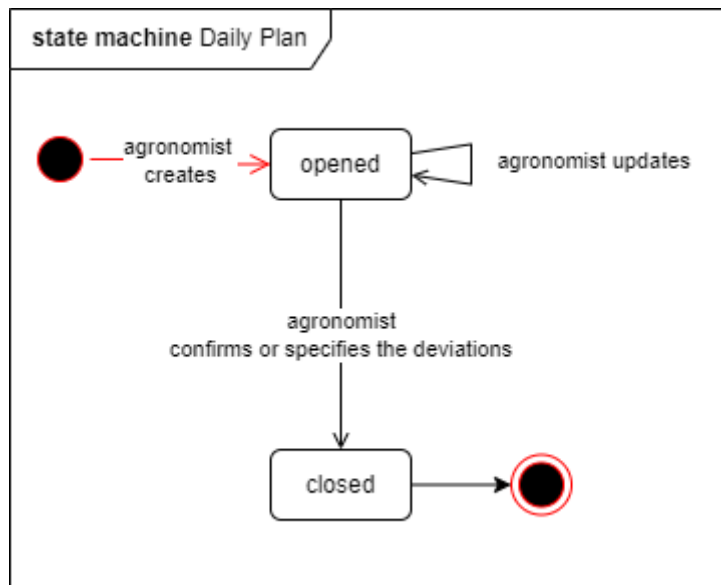
2.1.3 State Charts

State charts describe the behavior of the system while considering all possible states the objects can have when an event occurs. This analysis helps to clarify the most critical aspects of the system.

In the following, two simple state diagrams related to some concepts introduced in the previous section are shown. In particular, the first shows the state transitions of a problem, while the second one focuses on the states of a daily plan.



State Chart 1: Problem



State Chart 2: Daily Plan

2.2 Product functions

- **Farmer Register:** DREAM Application will allow Farmer to register by entering all the required information listed in Table 1. When registering to the application, firstly, he/she will declare to have read the Privacy Statement and secondly, he/she will have to accept the Terms and Conditions, which specially include his/her consent to the acquisition and processing of his/her data.

Information	Description	Mandatory
Full name	The Farmer's full name	Yes

Password	A password that meets the minimum-security requirements, i.e., it is at least 8 characters long and contains at least one number and one letter	Yes
PhoneNumber	The Farmer's phone number	Yes

Table 1

- **Farmer Login:** DREAM Application will allow Farmer to login by entering all the required information listed in Table 2.

Information	Description	Mandatory
PhoneNumber	Farmer's phone number which is used when he/she registers	
Password	Farmer's password which is used when he/she registers	

Table 2

- **Retrieve Information:** Farmer can select the location manually or use the location provided by GPS. If Farmer want to visualize the suggestions about his/her type of production, he/she must specify the type of production from the list of production types displayed in the System. Otherwise, System will display the weather only.

Information	Description	Mandatory
Location	The location of the Farmer or the production	Yes
ProductionType	The Farmer's type of production	No

Table 3

- **Ask for Help from Agronomist:** DREAM will allow Farmer to ask for help. Farmer can create a message by inputting the title and the description of the problems and submit. Then the message, together with information of the Farmer, will be visible to the agronomists.

Information	Description	Mandatory
Title	The title of the problem	Yes
Description	The detail of the problem	Yes

Table 4

- **Feedback the problem-solving result:** After Farmer applies the suggestion from Agronomist, he/she can feedback the result. Farmer can open the problem he/she created before and give a score.

Information	Description	Mandatory
Problem	Associated problem	Yes

Score	The feedback of the result after applying the suggestion.	Yes
-------	---	-----

Table 5

- **Report Production:** The Farmer can report the production when the harvest has been gathered in. After registration to the System and creating a report, he/she need to specify information listed in table 6 accordingly.

Information	Description	Mandatory
Type	Type of product	Yes
Amount	Produced amount per product	Yes
Acreage	Plantation area of the product	Yes
StartTime	When start to plant the product	Yes
EndTime	When the product harvests	Yes

Table 6

- **Create a post:** If Farmer wants to discuss or asks for help from other farmers, he/she can create a post in the forum by inputting the parameters listed in Table 7. After being created, the post would be visible to all Farmers.

Information	Description	Mandatory
Title	The title of the discussion	No
Content	The detail of the discussion	Yes

Table 7

- **Leave a comment:** If Farmer wants to comment on the post created by another Farmer, he/she can create a comment by inputting the parameters listed in Table 8. After being created, the comment would be appended to the related post and visible to all Farmers.

Information	Description	Mandatory
Post	The post which Farmer wants to comment	Yes
Content	The content of the comment	Yes

Table 8

- **Policy maker Register:** DREAM System will allow Policy maker to register. He/she can register by entering all the required information listed in Table 9. When registering to the System, He/she will first declare to have read the Privacy Statement, and secondly, they will have to accept the Terms and Conditions, which specifically include their consent to the acquisition and processing of their data.

Information	Description	Mandatory
-------------	-------------	-----------

Name	The Policy maker's full name	Yes
Email	The Policy maker's email	Yes
Password	A password that meets the minimum-security requirements, i.e., it is at least 8 characters long and contains at least one number and one letter	Yes

Table 9

- **Policy maker Logins:** DREAM System will allow Policy maker to log in. He/she can register by entering all the required information listed in Table 10.

Information	Description	Mandatory
Email	The Policy maker's email	Yes
Password	The password with regard to the registered password	Yes

Table 10

- **Visualize basic farm data:** DREAM System will allow policy makers to visualize farm information in interests of whole Telangana area, including short-term and long-term forecasts, water irrigation information, territory, soil humidity information and production information, which helps policy maker to identify the performance of farmers.

Information	Description	Mandatory
Location	The interested area of policymaker	Yes

Table 11

- **Input the score of farmers:** After calculating the final score of performance, policy maker can input the score into the System.

Information	Description	Mandatory
Famer	The personal information of farmers	Yes
Performance	The final score for the performance of farmers	Yes

Table 12

- **Visualize agronomist:** When the policy maker logins into the System, he/she can select the agronomist who he/she interested, then find basic information of daily plan, like title, date, farmer, execution, deviation and so on.

Information	Description	Mandatory
Name	The interested agronomist.	Yes

Table 13

- **Visualize the details of daily plan:** If the policy maker wants to check the details of daily plan of Agronomist, then he/she can click the "Title" of the daily plan to acquire

exact situation of execution.

Information	Description	Mandatory
Title	The title of the interested daily plan	Yes

Table 14

- **Register as agronomist:** DREAM Application will allow agronomists to register by entering all the required information listed in Table 15 When registering to the System firstly, he/she will declare to have read the Privacy Statement, and secondly, he/she will have to accept the Terms and Conditions, which specially include his/her consent to the acquisition and processing of his/her data.

Information	Description	Mandatory
Name	Name of agronomist	Yes
Email	Email of agronomist	Yes
Password	A password to verify that account is used correctly. It has at least 8 alphanumeric characters, 2 numeric characters.	Yes
Responsible area	The responsible area of agronomist	Yes

Table 15

- **Login as agronomist:** DREAM system allows agronomist to login System. In text table, he/she enters Email and Password.

Information	Description	Mandatory
Email	Email of agronomist	Yes
Password	password regarding the registered password	Yes

Table 16

- **Visualize farmer's basic data:** DREAM system allows agronomist to visualize farmers' basic information in his/her responsible area. After agronomist logs into System, he/she can choose basic information table about farmer. In table about farmer, system displays farmer's name, performance and the time of last visit.
- **Visualize area's basic data:** DREAM system allows agronomist to visualize basic information in his/her responsible area. After agronomist logs into System, he/she can see the information including weather, water irrigation, etc.

Information	Description	Mandatory
Area	area which agronomist is responsible for	Yes

Table 17

- **Answer to requests:** DREAM system will allow agronomist to answer farmer's request. After Agronomist receives a request from farmer who is in agronomist's responsible area, he/she searches relative information, gives answer to farmer's request.

Information	Description	Mandatory
Answer	Answer to requests	Yes

Table 18

- **Create a daily plan:** DREAM system will allow agronomist to create a new daily plan. When agronomist decides to create a new daily plan, he/she has to log in System then specify the following required information listed in Table 19.

Information	Description	Mandatory
Title	Title of daily plan	Yes
Date	Date of daily plan	Yes
Farmer	Farmer involved in this plan	Yes
Content	Details about daily plan	Yes

Table 19

- **Modify a daily plan:** DREAM system will allow agronomist to modify his/her daily plan. When agronomist change his/her decision about daily plan, he/she can open existing daily plan and modify one or more elements in Table 20.

Information	Description	Mandatory
Title	Title of daily plan	No
Date	Date of daily plan	No
Farmer	Farmer involved in this plan	No
Content	Details about daily plan	No

Table 20

- **Complete a daily plan:** After the agronomist creates a daily plan and visits the farmer, he/she has to change the status of execution and enter deviation in the System.

Information	Description	Mandatory
Status	State of plan	Yes
Deviation	Differences between plan and reality	Yes

Table 21

2.3 User characteristics

The actors of the application are the following:

- **Unregistered User:** a single person who has not yet registered.

- **Farmer:** a single person who has registered to the DREAM System and can use the Services it offers to visualize data, ask for help, report production, and discuss with other farmers.
- **Policy maker:** a single person who has registered as a Policy maker, to visualize the basic farm data, score the performance of farmers and visualize the daily plan of Agronomist.
- **Agronomist:** a single person who has registered as an agronomist in DREAN system, to browse information, answer requests and manage daily plan.

2.4 Assumptions, dependencies and constraints

2.4.1 Domain Assumptions

- D.1 The external service used by the System to retrieve the weather forecasts is assumed to work properly.
- D.2 Farmer owns a working smartphone which supports GPS.
- D.3 The GPS is assumed to be accurate.
- D.4 Data given by Farmers are assumed to be correct.
- D.5 The external service used by the System to show the information about the water usage of each farmer is assumed to be accurate.
- D.6 The external service used by the System to show the information about the humidity of soil is assumed to be accurate.
- D.7 Every character owns a working smartphone or computer which has access to internet connection
- D.8 Policy maker knows the standard of recognizing the performance of farmers.
- D.9 If a Farmer does not report his/her production information, his/her performance cannot be identified.
- D.10 Policy maker knows how to understand the initiatives carried out by Agronomist based on his/her daily plans
- D.11 Agronomist is responsible for only one area
- D.12 An area is associated to one and only one Agronomist
- D.13 One area contains multiple farms
- D.14 A farm is associated to one and only one area
- D.15 Agronomist makes no more than one daily plan for a day.
- D.16 Agronomist can have multiple daily plans
- D.17 A daily plan can involve multiple farmers
- D.18 Agronomist visits each farmer at least twice a year.
- D.19 Agronomist visits under-performing farmer more often.
- D.20 Agronomist knows how to solve requests from farmers.
- D.21 A Farmer has one and only one farm.
- D.22 A Farmer can have multiple production
- D.23 A Farmer's request can only be handled by the Agronomist who is responsible for the area where is the Farmer's farm.

- D.24 A Farmer can have multiple discussion with other Farmers
- D.25 Agricultural Product Reference System provides suggestion about planting.
- D.26 Data given by Agronomist are assumed to be correct.
- D.27 Agronomist's email must be unique
- D.28 Policy maker's email must be unique
- D.29 Farmer's phone number must be unique

2.4.2 Dependencies

- The System will use the GPS of the Farmer's computer/smartphone.
- The System will use the Internet connectivity of the Farmer's computer/smartphone.
- The System will use an external service to retrieve meteorological short-term and long-term forecasts
- The System will use an external service to retrieve water usage per farmer.
- The System will use an external service to retrieve the data about the humidity of soil.
- The System will use an external service to retrieve personal suggestions concerning specific crops

3. SPECIFIC REQUIREMENTS:

3.1 External Interface Requirements

The following section will give a more detailed description, in terms of hardware, software, and communication interfaces.

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

Policy maker must have a computer equipped network in order to join the service, acquire the data and so on.

Farmers must have a computer/smartphone equipped with GPS and network in order to use the DREAM System and all its functionalities.

Agronomist must have the computer/smartphone equipped with network for updating the daily plan and so on.

3.1.3 Software Interfaces

The DREAM System relies on various external services accessible via API. These services are:

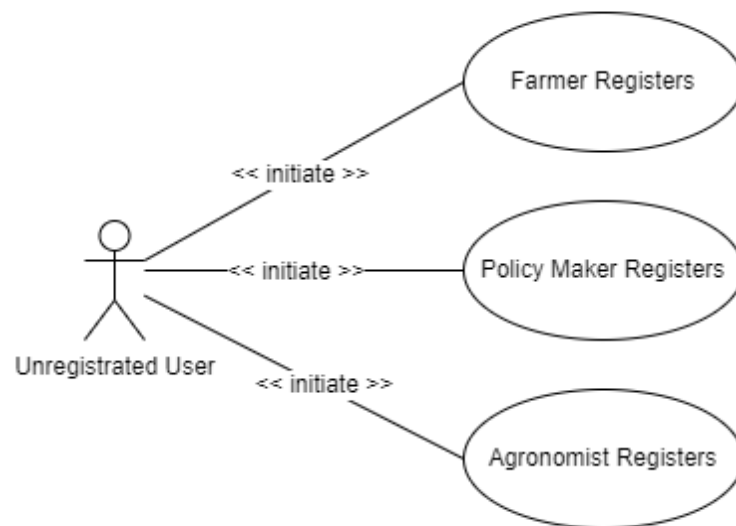
- **Weather Service:** the DREAM System relies on it for meteorological short-term and long-term forecasts.
- **Soil Sensor System:** the DREAM System relies on it to provide information about the humidity of soil.
- **Water Irrigation System:** the DREAM System relies on it to provide information concerning the amount of water used by each farmer.
- **GPS Service:** the DREAM System relies on it to obtain the location of the Farmer.
- **Agricultural Product Reference System:** the DREAM System relies on it to provide production suggestion to Farmer.

3.1.4 Communication Interfaces

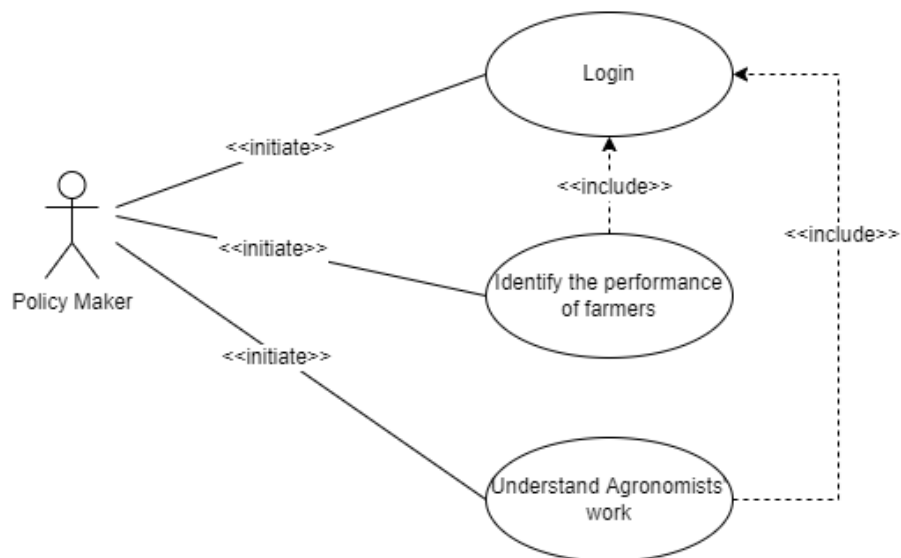
The various devices connect to the DREAM System via Internet connection.

3.2 Functional Requirements

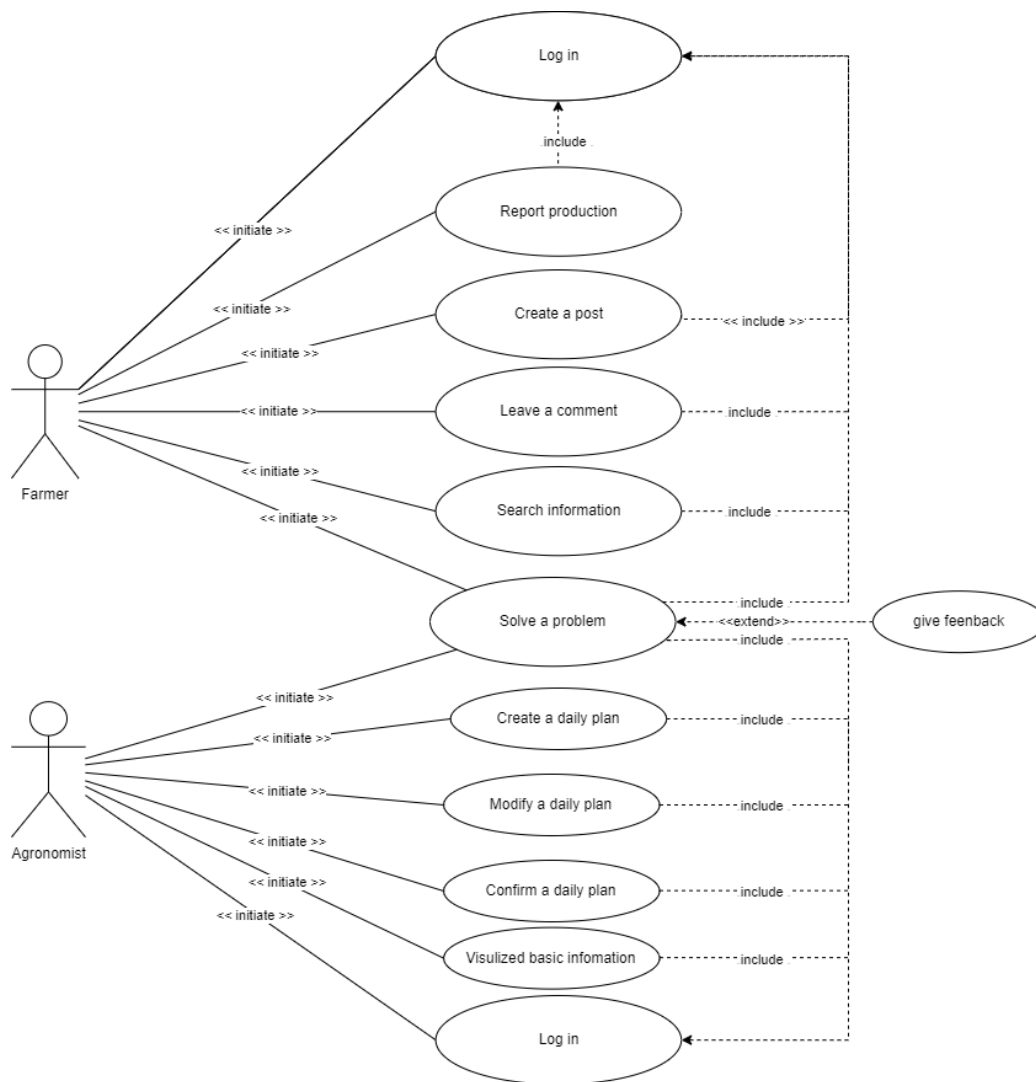
3.2.1 Use Case Diagrams



Use Case Diagram 1: Unregistered User



Use Case Diagram 2: Policy Maker



Use Case Diagram 3: Agronomist and Farmer

3.2.2 Use Case Analysis

Name	[UC.1] Agronomist Registers
Actors	Unregistered User
Entry Conditions	Unregistered User wants to join DREAM
Events Flow	<ol style="list-style-type: none"> 1. Unregistered User opens the System and clicks 'Sign up' to open the Agronomist account creation page. 2. Unregistered User enters his/her user name in the appropriate fields of creation page. 3. Unregistered User enters his/her email. 4. Unregistered User enters a password.

	<ol style="list-style-type: none"> 5. Unregistered User select response area. 6. Unregistered User checks the "Accept Privacy Statement" checkbox. 7. Unregistered User checks the "Accept Terms and Conditions" checkbox. 8. Unregistered User clicks the "create an account" button. 9. The System saves the information. 10. The System sends the registration result.
Exit Conditions	Unregistered User successfully registered on DREAM
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Not all required fields are correctly filled in. • Inserted password does not comply with the minimum-security standards. • "Accept Terms and Conditions" checkbox is not checked. • "Accept Privacy Statement" checkbox is not checked. • Email is already in use.

Name	[UC.2] Agronomist Logins
Actors	Agronomist
Entry Conditions	<ul style="list-style-type: none"> ● User already registers as Agronomist in DREAM System
Events Flow	<ol style="list-style-type: none"> 1. Agronomist opens the System and clicks "Agronomist" link, entering Agronomist login page. 2. Agronomist enters email. 3. Agronomist enters password. 4. Agronomist clicks 'Login' button. 5. The System checks if the given login parameters are valid. 6. The System creates a new session for the user.
Exit Conditions	<ul style="list-style-type: none"> • Agronomist successfully logged in DREAM System
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Agronomist enters invalid email. • Agronomist enters invalid password.

Name	[UC.3] Agronomist creates the daily plan
Actors	Agronomist
Entry Conditions	<ul style="list-style-type: none"> • Agronomist successfully logged into DREAM • Agronomist wants to create the daily plan
Events Flow	<ol style="list-style-type: none"> 1. Agronomist gets into the Homepage. 2. The System shows the basic information of response area to Agronomist, including "Location", "Weather", "Irrigation System", "oil Moisture data", "Farmer Information" and "Daily Plan". 3. Agronomist clicks "Create New" on the "Daily Plan" tab, entering NewDailyPlan page.

	<ol style="list-style-type: none"> The System shows daily plan creation page. Agronomist inputs the "title" of daily plan. Agronomist selects the "date" of daily plan. Agronomist selects the name of "farmers". Agronomist inputs the "content" of daily plan. Agronomist confirms the plan and clicks "submit". The System saves the daily plan and redirects to the homepage.
Exit Conditions	Agronomists create the daily plan.
Exceptions	<p>In the following situations, the System shows an error message on the NewDailyPlan page.</p> <ul style="list-style-type: none"> Inserted date is not valid Inserted deviation is not valid Daily plan for inserted date has been existed.

Name	[UC.4] Agronomist modifies the daily plan
Actors	Agronomist
Entry Conditions	<ul style="list-style-type: none"> Agronomist successfully logged into DREAM Agronomist wants to modify the existing daily plan
Events Flow	<ol style="list-style-type: none"> Agronomist gets into the Homepage. Agronomist clicks the "modify" on the HomePage, entering ModifyDailyPlan page. Agronomist changes the "title" of daily plan. Agronomist changes the "date" of daily plan. Agronomist changes the name of "farmers". Agronomist changes the "content" of daily plan. Agronomist confirms the plan and clicks "modify". The System saves the daily plan and redirects to the homepage.
Exit Conditions	Agronomists modify the daily plan.
Exceptions	<p>In the following situations, the System shows an error message on the ModifyDailyPlan page.</p> <ul style="list-style-type: none"> Inserted date is not valid Inserted deviation is not valid Daily plan for inserted date has been existed.

Name	[UC.5] Agronomist confirms the daily plan
Actors	Agronomist
Entry Conditions	<ul style="list-style-type: none"> Agronomist successfully logged into DREAM Agronomist wants to confirm the existing daily plan
Events Flow	<ol style="list-style-type: none"> Agronomist gets into the Homepage. Agronomist clicks the "confirm" on the HomePage, entering ConfirmDailyPlan page. Agronomist inputs the deviation if needed.

	<ol style="list-style-type: none"> 4. Agronomist confirms the plan and clicks "confirm". 5. The System saves the daily plan and redirects to the homepage.
Exit Conditions	Agronomists confirm the daily plan.
Exceptions	<p>In the following situations, the System shows an error message on the ConfirmDailyPlan page.</p> <ul style="list-style-type: none"> • Inserted deviation is not valid

Name	[UC.6] Farmer Registers
Actors	Unregistered User
Entry Conditions	Unregistered User wants to join DREAM
Events Flow	<ol style="list-style-type: none"> 1. Unregistered User opens the System and clicks Farmer 'Sign up' to open the Farmer account creation page. 2. Unregistered User enters his/her user name in the appropriate fields of creation page. 3. Unregistered User enters his/her phone number. 4. Unregistered User enters a password. 5. Unregistered User enters his/her own farms. 6. Unregistered User checks the "Accept Privacy Statement" checkbox. 7. Unregistered User checks the "Accept Terms and Conditions" checkbox. 8. Unregistered User clicks the "Create an account" button. 9. The System saves the information. 10. The System sends the registration result.
Exit Conditions	Unregistered User successfully registered on DREAM
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Not all required fields are correctly filled in. • Inserted password does not comply with the minimum-security standards. • "Accept Terms and Conditions" checkbox is not checked. • "Accept Privacy Statement" checkbox is not checked. • Phone number is already in use.

Name	[UC.7] Farmer Logins
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> ● User successfully registers as farmer in System
Events Flow	<ol style="list-style-type: none"> 1. Farmer opens the System and clicks "Farmer" link, entering Farmer login page. 2. Farmer enters phone number in the appropriate field. 3. Farmer enters password in the appropriate field. 4. Farmer clicks 'Login' button. 5. The System checks if the given login parameters are valid

	6. The System creates a new session for the user
Exit Conditions	<ul style="list-style-type: none"> Farmer successfully logged in DRAEM System
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Inserted phone number is not valid Inserted password does not match the inserted phone number

Name	[UC.8] Farmer searches information
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> Farmer successfully logged in System Farmer wants to search information about planation
Events Flow	<ol style="list-style-type: none"> In the homepage, farmer clicks 'Search' button. In Search page, system automatically displays Farmer's location by GPS. Farmer selects his/her interested location. Farmer selects type of production which he/she plants in the location. Farmer clicks 'Submit' button. System displays weather forecast, planting suggestion to Farmer.
Exit Conditions	<ul style="list-style-type: none"> Farmer successfully finished searching
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Inserted location is not valid Inserted type is not valid <p>If Farmer doesn't choose production, then system will show weather forecast only.</p>

Name	[UC.9] Farmer report production
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> Farmer successfully logged in System Farmer wants to report his/her production
Events Flow	<ol style="list-style-type: none"> In the homepage, farmer clicks 'Report' button. Farmer selects type of production. Farmer enters amount of production. Farmer selects planting area of production. Farmer selects planting start time of production. Farmer selects planting end time of production. Farmer confirms information and clicks 'Submit' button. The System saves the report. The System redirect to the Report Page.
Exit Conditions	<ul style="list-style-type: none"> Farmer successfully reports production
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Inserted type is not valid

	<ul style="list-style-type: none"> • Inserted amount is not valid • Inserted area is not valid • Inserted start time is not valid • Inserted end time is not valid
--	--

Name	[UC.10] Solve a problem
Actors	<ul style="list-style-type: none"> • Farmer • Agronomist
Entry Conditions	<ul style="list-style-type: none"> ● Farmer successfully logged in System ● Farmer wants to ask for help ● Agronomist successfully logged in System
Events Flow	<ol style="list-style-type: none"> 1. In the homepage, farmer clicks 'Request' button. 2. In Request page, farmer clicks 'Create' button in Request List. 3. In Create page, Farmer enters title of request. 4. Farmer enters content of request. 5. Farmer confirms content and clicks 'Submit' button. 6. System saves the request and redirect the Farmer to Request page 7. System associated the request to the Agronomist. 8. On the homepage, Agronomist clicks the request notification button. 9. The System shows the requests to Agronomist. 10. Agronomists enters answer to the request and clicks "submit". 11. System saves the answer. 12. In Request page, farmer clicks interested request in Request List. 13. System displays answer to the request.
Exit Conditions	<ul style="list-style-type: none"> • Farmer successfully gets answer
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Inserted request is not valid • Inserted answer is not valid

Name	[UC.11] Farmer gives feedback
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> ● Farmer successfully logged in System ● Farmer gets the answer ● Farmer wants to give feedback
Events Flow	<ol style="list-style-type: none"> 1. In Request page, farmer clicks interested request in Request List. 2. System displays information about interested request. 3. Farmer enters score. 4. Farmer clicks 'Submit feedback'

	<ol style="list-style-type: none"> The System saves the score. The System redirect to Request Page
Exit Conditions	<ul style="list-style-type: none"> A feedback score is saved
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Score is invalid

Name	[UC.12] Farmer create a post
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> Farmer successfully logged in System Farmer wants to discuss with other farmers
Events Flow	<ol style="list-style-type: none"> In the homepage, farmer clicks 'Forum' button. In Forum page, Farmer enters title of post in appropriate filed. Farmer enters content of post in appropriate filed. Farmer confirms content and clicks 'Submit' button. System saves the post. System redirects to Forum Page.
Exit Conditions	<ul style="list-style-type: none"> Farmer successfully creates a post
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Inserted title is not valid Inserted content is not valid

Name	[UC.13] Farmer leaves a comment
Actors	Farmer
Entry Conditions	<ul style="list-style-type: none"> Farmer successfully logged in System Farmer wants to create a comment of other farmer's post
Events Flow	<ol style="list-style-type: none"> In the homepage, farmer clicks 'Forum' button. System displays farmer the list of latest post titles. Farmer clicks interested post. System displays farmer title and detail content of the interested post. Farmer enters comment in appropriate area. Farmer confirm comment and clicks 'Submit' button. System saves the comment.
Exit Conditions	<ul style="list-style-type: none"> Farmer successfully sends a comment
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> Inserted comment is not valid

Name	[UC.14] Policy Maker Registers
Actors	Unregistered User
Entry Conditions	<ul style="list-style-type: none"> Unregistered User wants to join DREAM
Events Flow	<ol style="list-style-type: none"> Unregistered User opens the Application and clicks "sign up", entering Policy Maker account creation page

	<ol style="list-style-type: none"> 2. Unregistered User enters his/her username in the appropriate fields 3. Unregistered User enters his/her email 4. Unregistered User enters a password 5. Unregistered User checks the "Accept Privacy Statement" checkbox 6. Unregistered User checks the "Accept Terms and Conditions" checkbox 7. Unregistered User clicks the "Create an account" button 8. The System saves the information 9. The System sends the registration result
Exit Conditions	<ul style="list-style-type: none"> • Unregistered User successfully registered on DREAM
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Not all required fields are correctly filled in. • Inserted password does not comply with the minimum-security standards • "Accept Terms and Conditions" checkbox is not checked • "Accept Privacy Statement" checkbox is not checked • Email is already in use

Name	[UC.15] Policy Maker Logins
Actors	Policy Maker
Entry Conditions	<ul style="list-style-type: none"> • Policy Maker is successfully registered to DREAM
Events Flow	<ol style="list-style-type: none"> 1. Policy Maker opens the System and clicks "Policy Maker" link, entering policy maker login page 2. Policy Maker enters his/her email 3. Policy Maker enters his/her password 4. Policy Maker clicks the "Login" button 5. The System checks if the given login parameters are valid 6. The System creates a new session for the user
Exit Conditions	<ul style="list-style-type: none"> • Policy Maker successfully logged in
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Policy Maker enters invalid email • Policy Maker enters invalid password

Name	[UC.16] Policy Maker identifies the performance of farmers
Actors	Policy Maker
Entry Conditions	<ul style="list-style-type: none"> • Policy Maker successfully logged into DREAM • Policy Maker wants to identify the performance of farmers • Farmers already upload their production information
Events Flow	<ol style="list-style-type: none"> 1. Policy Maker clicks the interested area in the whole map of

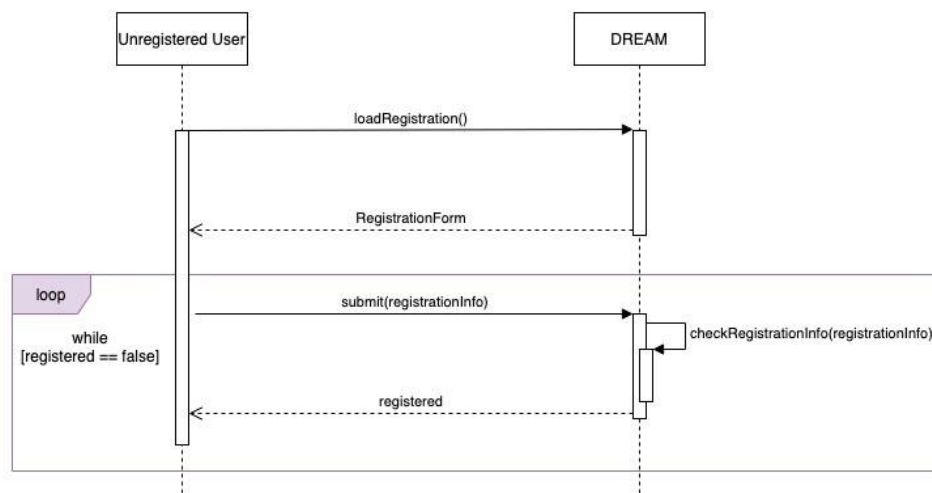
	<p>the Telangana showing in the homepage</p> <ol style="list-style-type: none"> 2. The System shows farms' detail regarding to that area, including short-term and long-term forecasts, water irrigation information, territory and soil humidity information, and farmers' production information. 3. Policy Maker clicks "Update Performance" button, entering performance update page. 4. The System shows a list of farmers in that area. 5. Policy Maker inputs the score for each farmer 6. Policy Maker clicks "Submit" button 7. The System saves the performance information and redirect to the page shown in step 2
Exit Conditions	The performance information is successfully saved.
Exceptions	<p>In the following situations, the System shows an error message.</p> <ul style="list-style-type: none"> • Policy Maker enters invalid performance score

Name	[UC.17] Policy Maker understand Agronomists' work
Actors	Policy Maker
Entry Conditions	<ul style="list-style-type: none"> • Policy Maker successfully logged into DREAM • Policy Maker wants to understand if the initiatives carried out by agronomists produce significant results
Events Flow	<ol style="list-style-type: none"> 1. In the homepage, Policy Maker clicks one of the agronomists' names in the agronomist list 2. The System shows all daily plans for the choosing agronomist 3. Policy Maker chooses one of the daily plans he/she wants to see the details by clicking the date in daily plan list, entering daily plan detail page 4. The System displays details including date, farmer, content, deviation and state related to that daily plan
Exit Conditions	<ul style="list-style-type: none"> • The details of the daily plan are shown and can be analyzed by the Policy Maker
Exceptions	<p>The System cannot find any daily plan for the choosing agronomist. In this case, the daily plan page regarding to that agronomist will not show any daily plan and display a notice.</p> <p>The System finds that the choosing agronomist does not exist. In this case, the System will redirect to the homepage and shows the error message.</p> <p>The System cannot find selected daily plan. In this case, the System will redirect to the daily plan page regarding to that agronomist and shows the error message.</p>

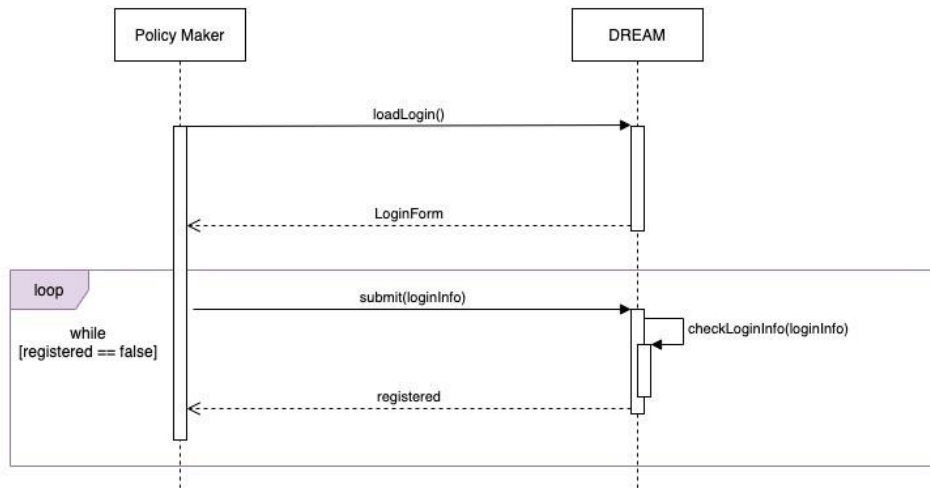
Name	[UC.18] Agronomist visualizes information
Actors	Agronomist
Entry Conditions	<ul style="list-style-type: none"> ● Agronomist successfully logged in System ● Agronomist wants to visualize information
Events Flow	<ol style="list-style-type: none"> 1. Agronomist clicks the tabs in the homepage 2. The System shows relative details regarding to different tabs, including location, weather, water irrigation, territory and soil humidity information, farmers' information and daily plan.
Exit Conditions	<ul style="list-style-type: none"> ● Agronomist successfully visualizes the information

3.2.3 Sequence Diagrams

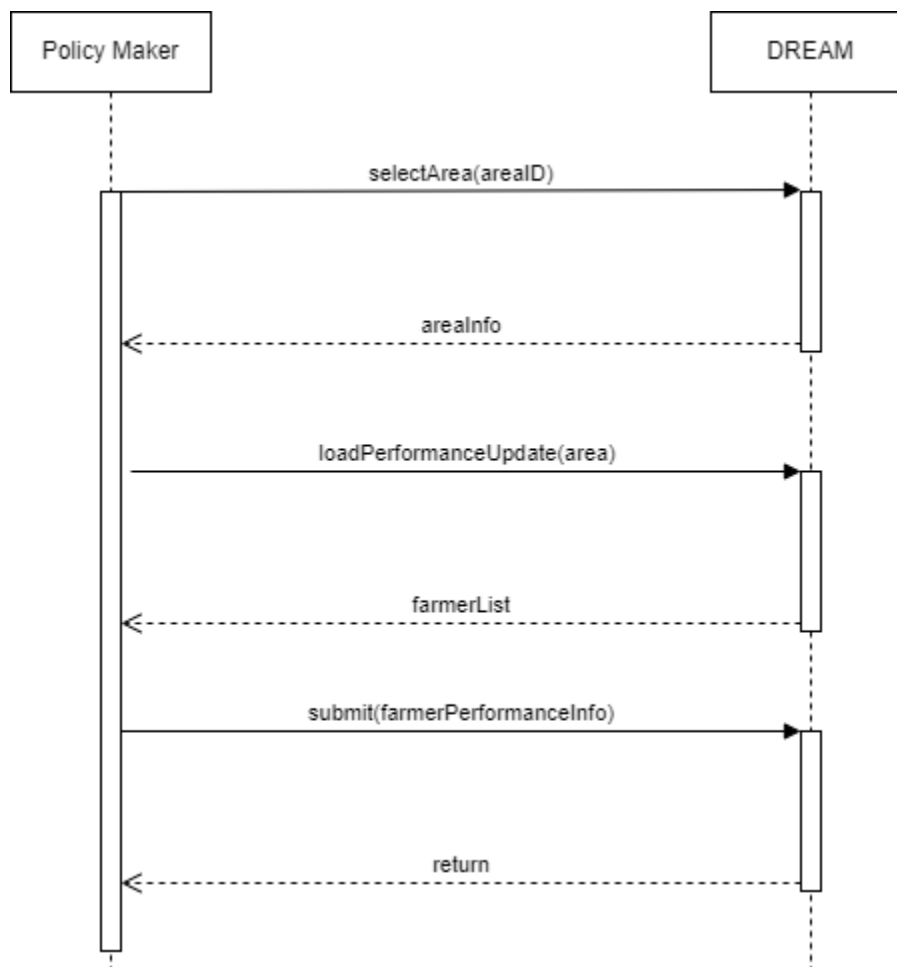
This section provides a high-level representation of the interaction processes between the various actors and the System. They constitute a visual representation of some of the previously described use cases.



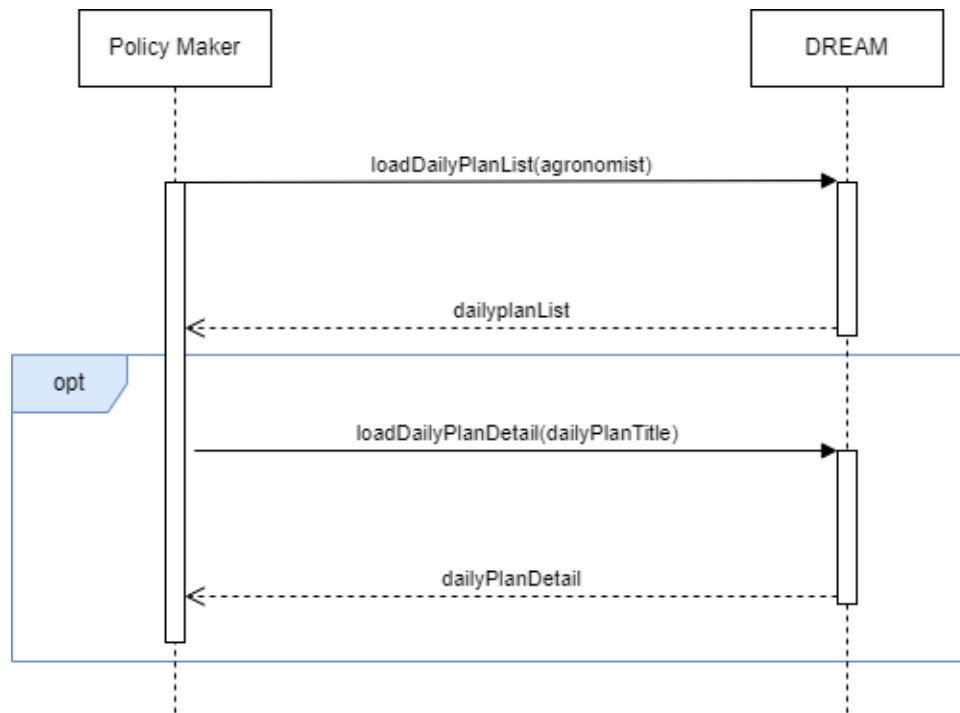
Sequence Diagram 1: Registration



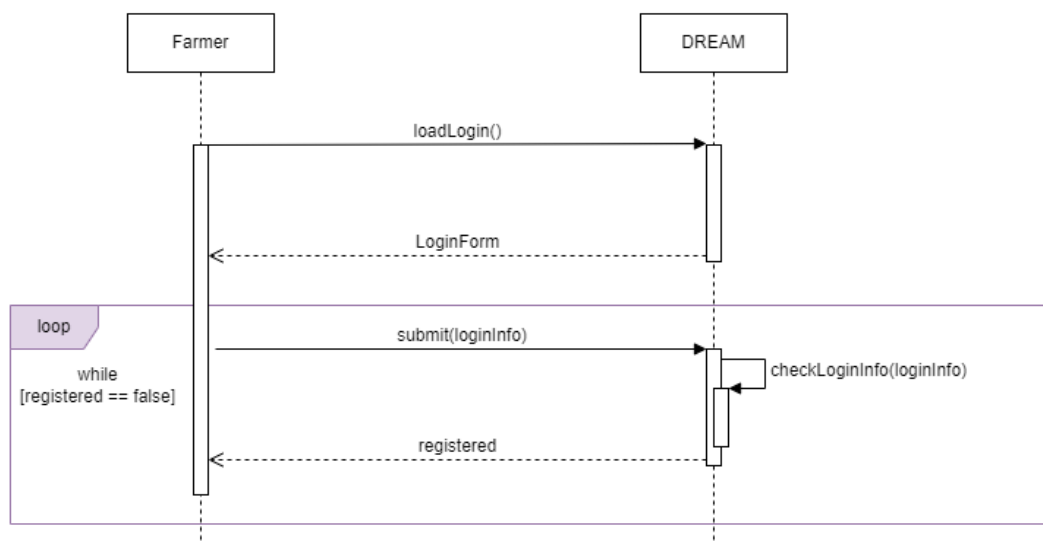
Sequence Diagram 2: Policy Maker Login



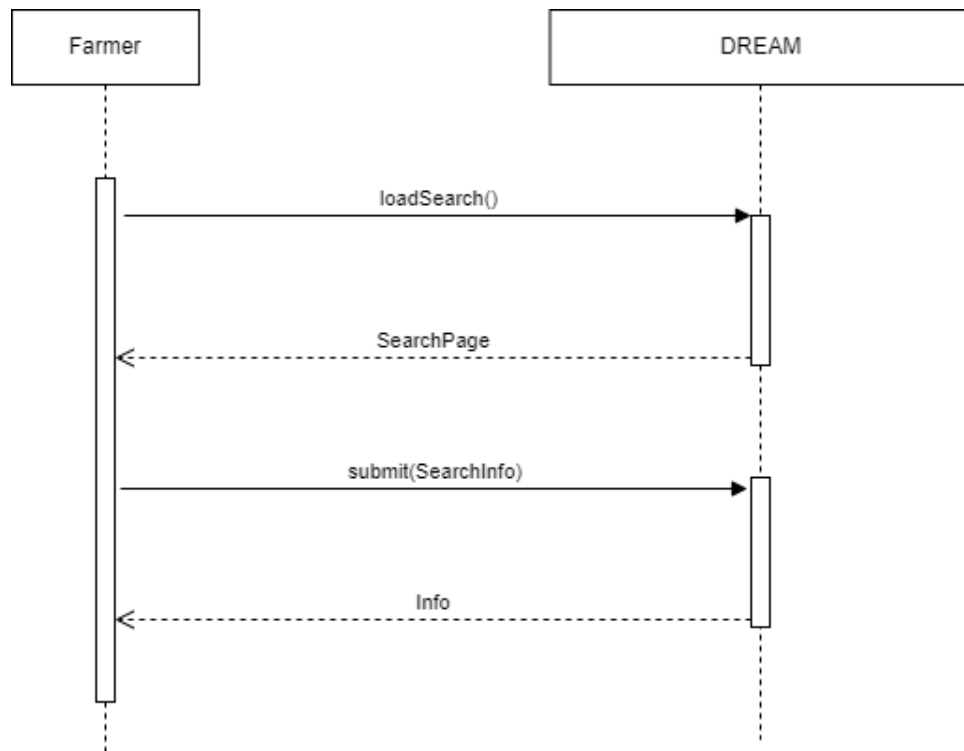
Sequence Diagram 3: Policy Maker Identifies the Performance of Farmer



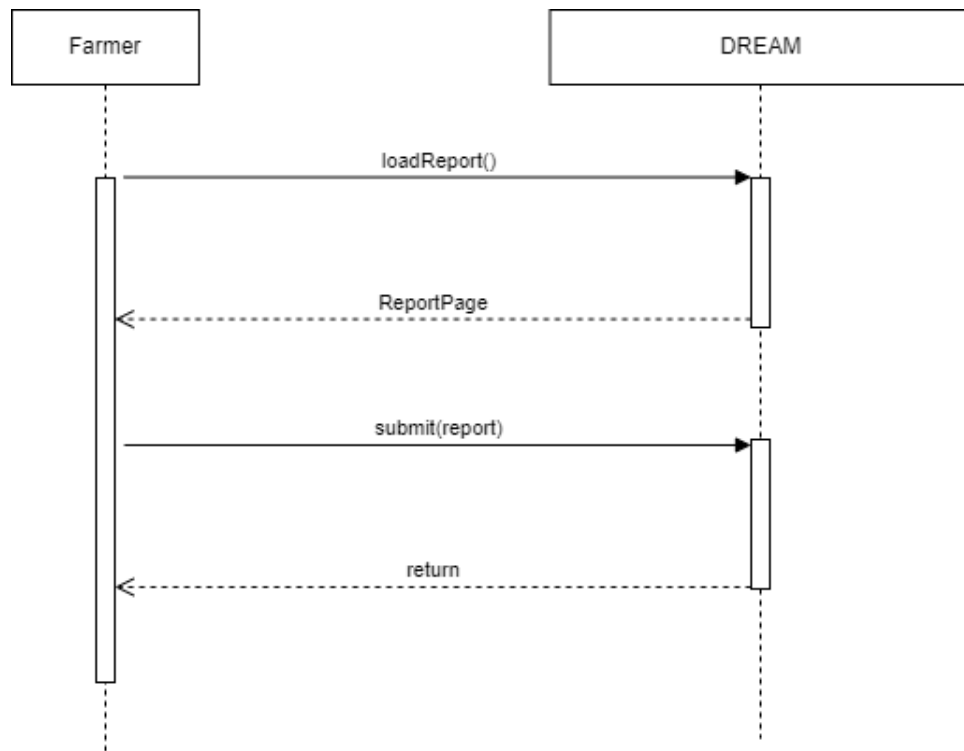
Sequence Diagram 4: Policy Maker understands Agronomists' work



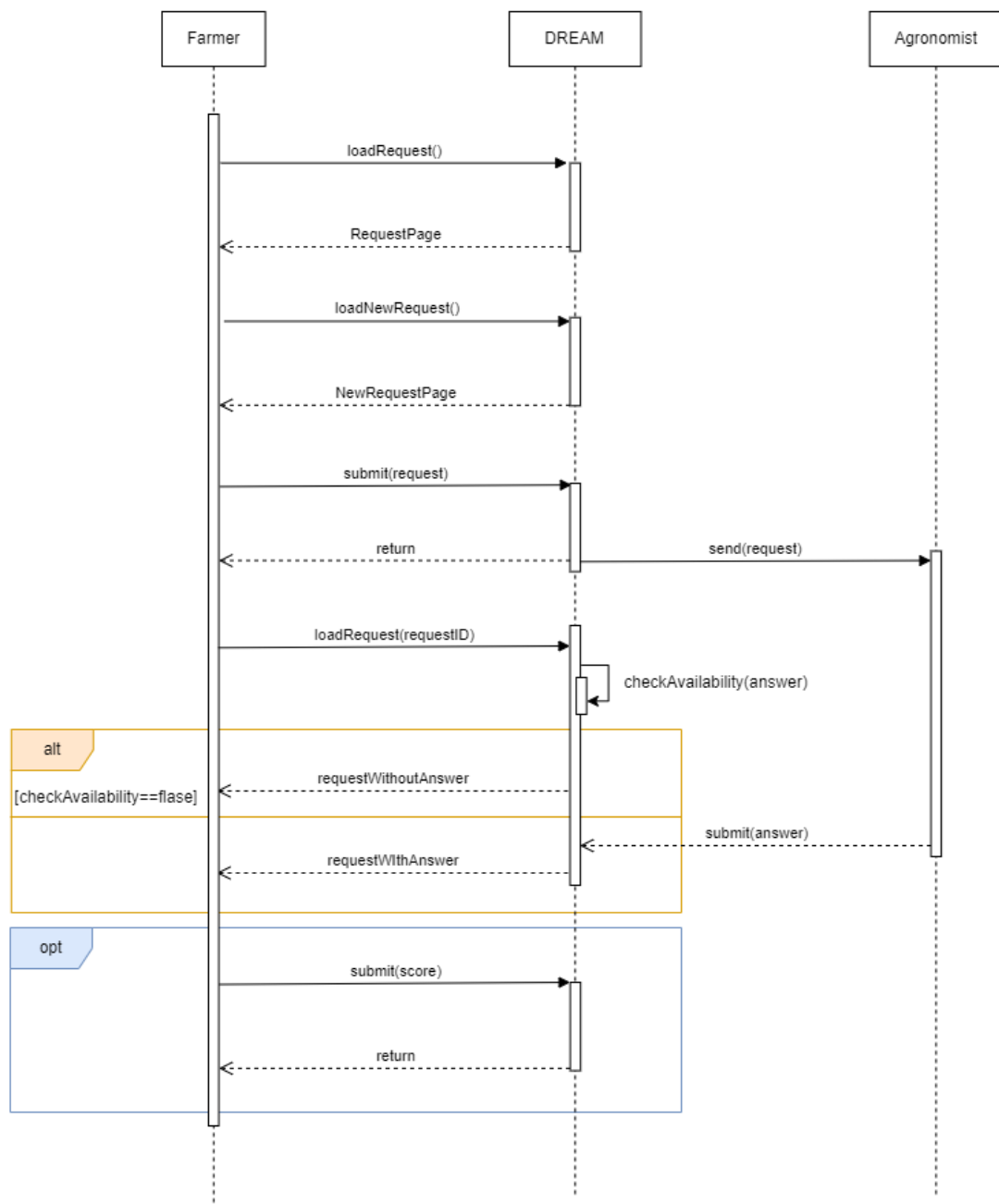
Sequence Diagram 5: Farmer Login



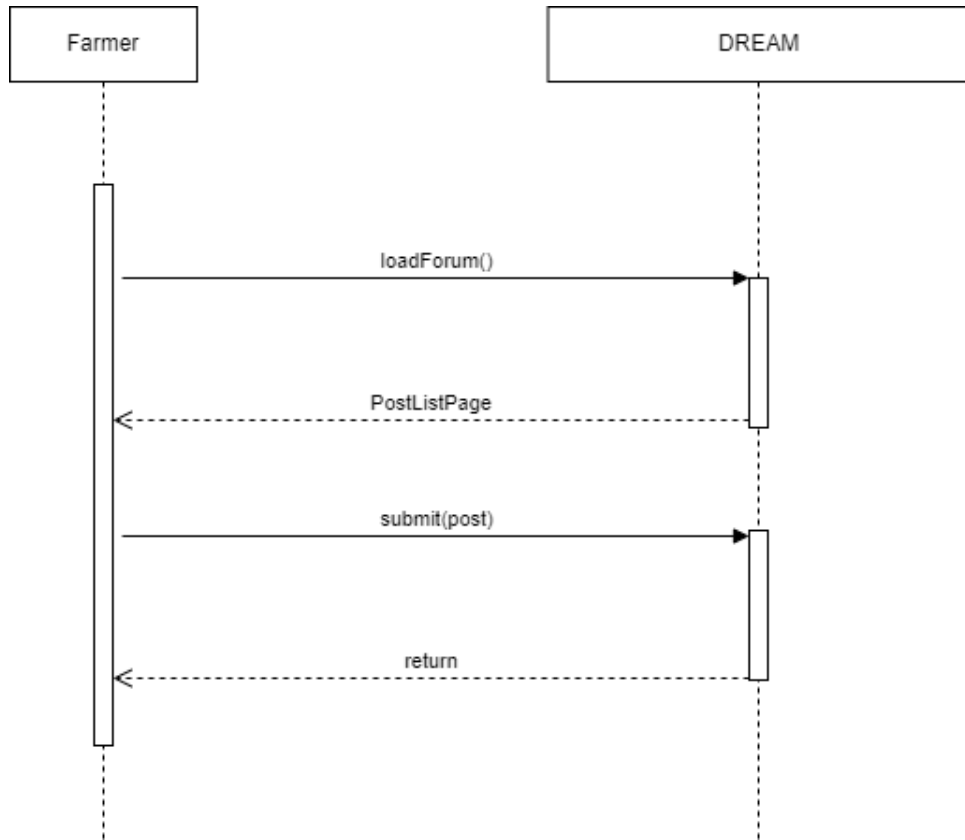
Sequence Diagram 6: Farmer Search Information



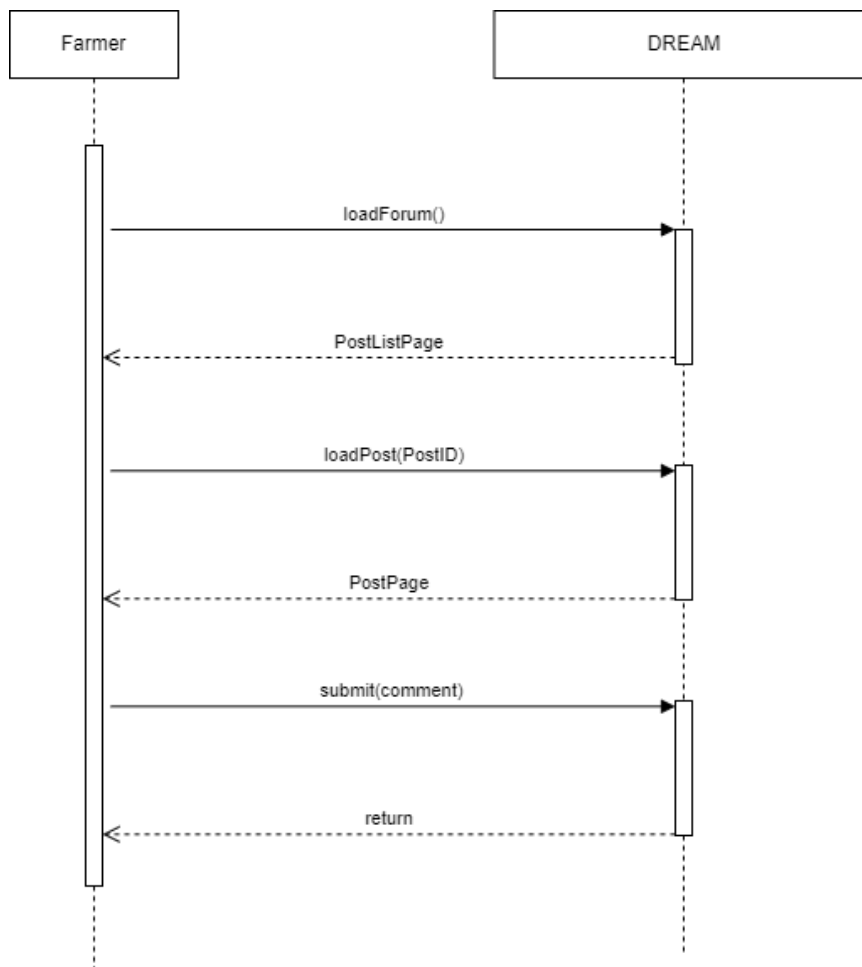
Sequence Diagram 7: Farmer creates a production report



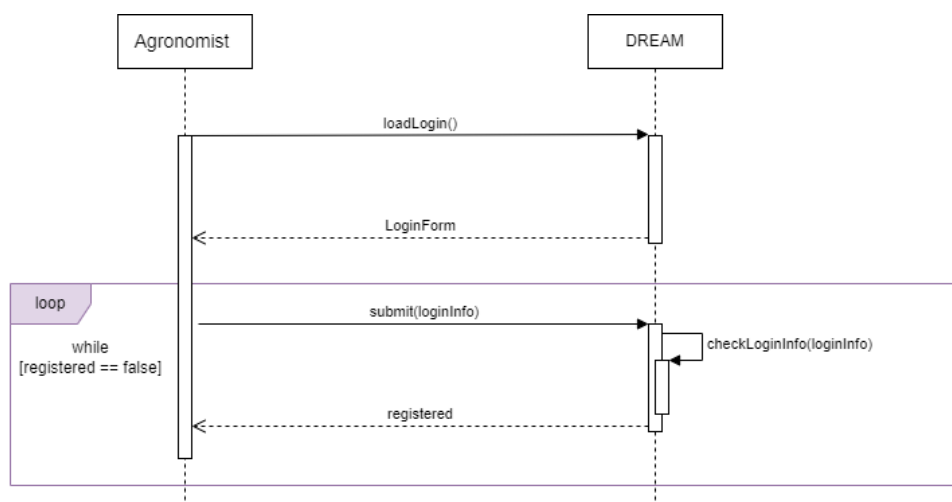
Sequence Diagram 8: Farmer asks for help



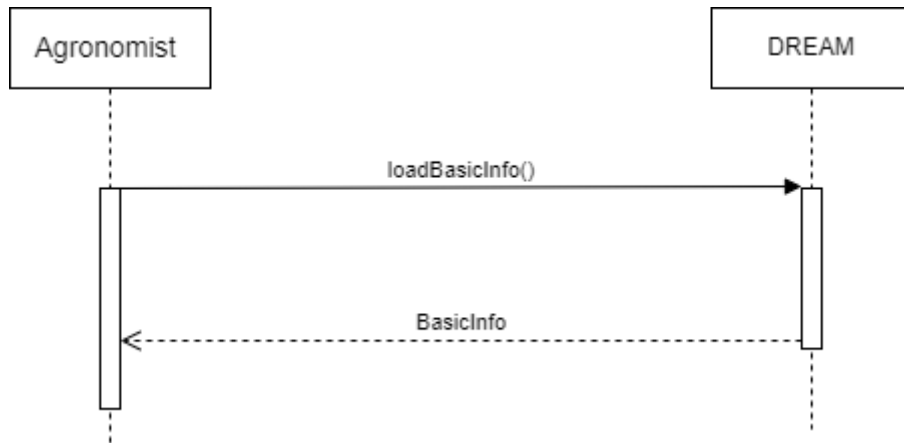
Sequence Diagram 9: Farmer creates new post



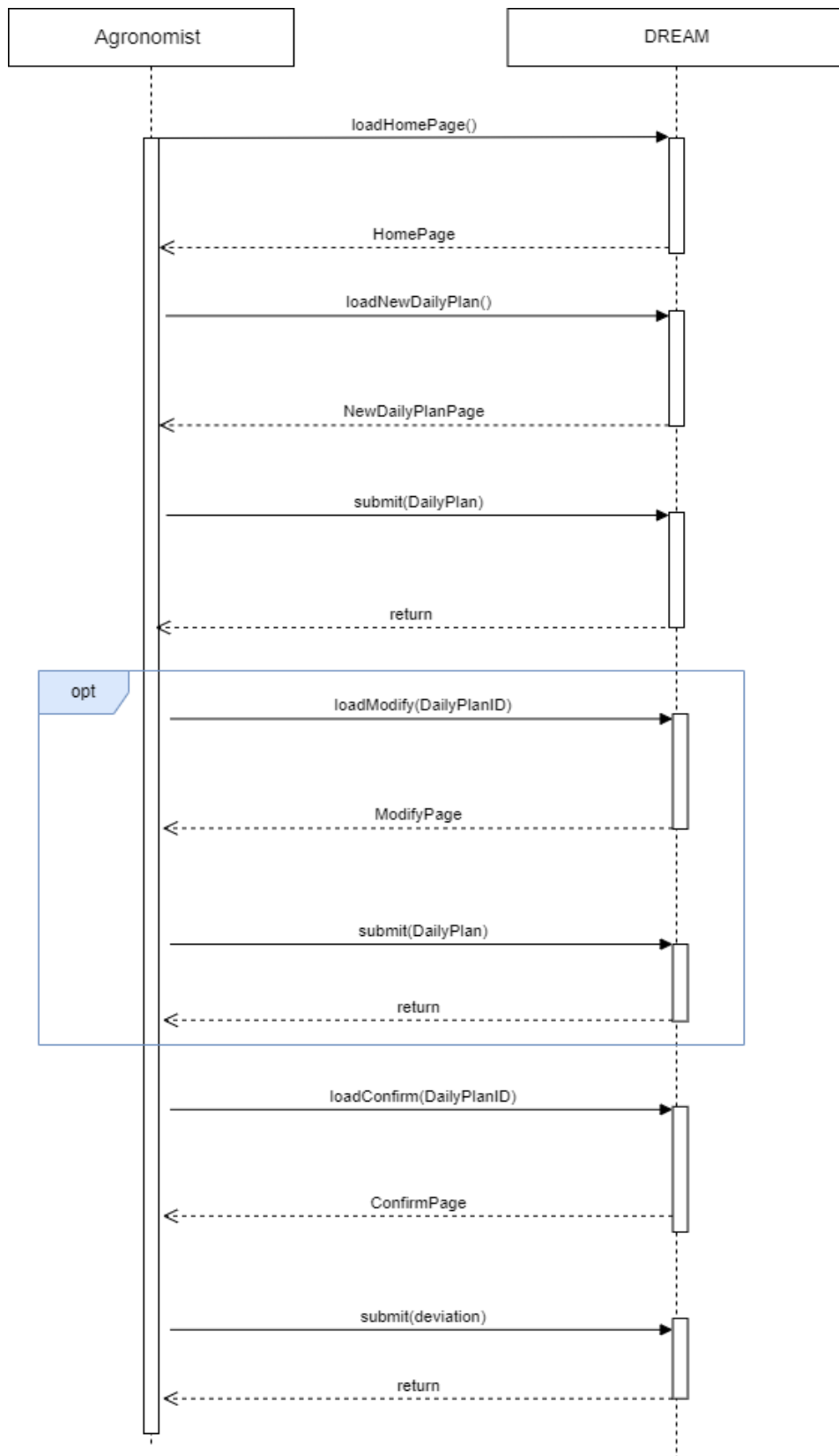
Sequence Diagram 10: Farmer leaves a comment



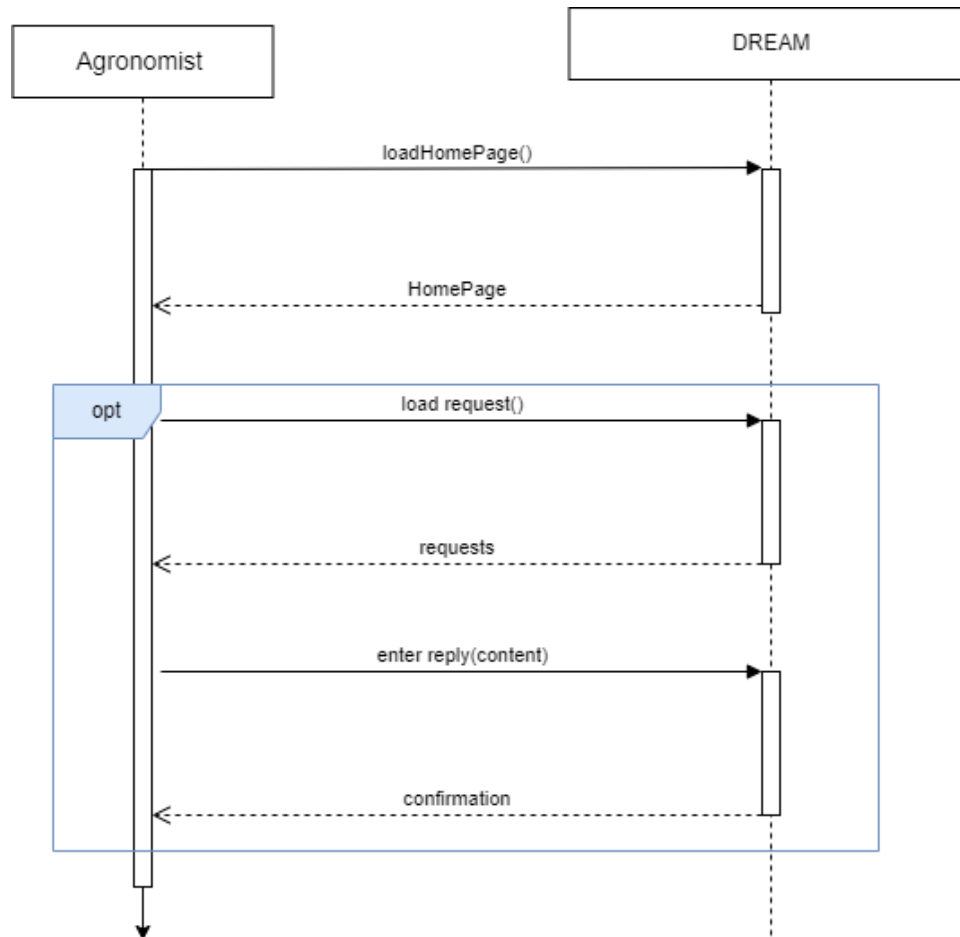
Sequence Diagram 11: agronomist Login



Sequence Diagram 12: agronomist visualize data



Sequence Diagram 13: complete Daily Plan



Sequence Diagram 14: agronomist reply to the request

3.2.4 Requirements

This section lists all the requirements that the DREAM System must meet in order to achieve the set goals. A detailed mapping between them will be provided in section 3.2.5.

R.1 The System must allow Policy Maker to register DREAM by filling in a form containing a set of fields

- R.1.1 Policy Maker must be able to indicate his/her username
- R.1.2 Policy Maker must be able to indicate his/her email
- R.1.3 Policy Maker must be able to indicate his/her password

- R.1.4 Policy Maker must be able to read and accept the Terms and Conditions
- R.1.5 Policy Maker must be able to read and accept the Privacy Statement
- R.2 The System must store the personal data relating to Policy Maker
- R.3 The System must allow Policy Maker to login into DREAM by entering his email and password.
- R.4 The System must allow Policy Maker to visualize the data about his/her interested area in Telegana.
 - R.4.1 Policy Maker must be able to visualize whole areas in Telegana.
 - R.4.2 Policy Maker must be able to visualize the weather forecast
 - R.4.3 Policy Maker must be able to visualize the water irrigation information
 - R.4.4 Policy Maker must be able to visualize the soil humidity information
 - R.4.5 Policy Maker must be able to visualize Farmer's production information
- R.5 The System must be able to store Farmer's performance
- R.6 The System must allow Policy Maker to input Farmer's performance
- R.7 The System must allow Policy Maker to visualize daily plans of Agronomist
 - R.7.1 Policy Maker must be able to visualize the daily plan list for his/her interested Agronomist
 - R.7.2 Policy Maker must be able to visualize the detail of daily plan for his/her interested Agronomist
- R.8 The System must allow Farmer to register DREAM by filling in a form containing a set of fields
 - R.8.1 Farmer must be able to indicate his/her username
 - R.8.2 Farmer must be able to indicate his/her phone number
 - R.8.3 Farmer must be able to indicate his/her password
 - R.8.4 Farmer must be able to read and accept the Terms and Conditions
 - R.8.5 Farmer must be able to read and accept the Privacy Statement
- R.9 The System must store the personal data relating to Farmer
- R.10 The System must allow Farmer to login into DREAM by entering his phone number and password.
- R.11 The System must allow Farmer to search the data
 - R.11.1 Farmer must be able to indicate location
 - R.11.2 Farmer must be able to indicate type of production
 - R.11.3 Farmer must be able to visualize the weather forecast
 - R.11.4 Farmer must be able to visualize the suggestions
- R.12 The System must allow Farmer to ask for help
 - R.12.1 Farmer must be able to indicate his/her problem
 - R.12.2 Farmer must be able to visualize answer to the problem
 - R.12.3 Farmer must be able to indicate feedback, a score, to agronomist's answer
- R.13 The System must store the problem data
- R.14 The System must allow Farmer to report his/her production
 - R.14.1 Farmer must be able to indicate his/her type of production
 - R.14.2 Farmer must be able to indicate his/her amount of production
 - R.14.3 Farmer must be able to indicate his/her planting area of production
 - R.14.4 Farmer must be able to indicate his/her planting period of production

- R.15 The System must store the production report data
- R.16 The System must allow Farmer to create a post
 - R.16.1 Farmer must be able to indicate his/her title
 - R.16.2 Farmer must be able to indicate content of post
- R.17 The System must store the new post
- R.18 The System must allow Farmer to visualize posts list
- R.19 The System must allow Farmer to visualize post details
- R.20 The System must allow Farmer to leave a comment
 - R.20.1 Farmer must be able to indicate his/her comment
- R.21 The System must store the new comment
- R.22 The System must allow agronomist to register to DREAM by filling in a form containing a set of fields
 - R.22.1 Agronomist must be able to indicate his/her name
 - R.22.2 Agronomist must be able to indicate an email
 - R.22.3 Agronomist must be able to indicate a password
 - R.22.4 Agronomist must be able to indicate a response area
 - R.22.5 Agronomist must be able to read and accept the Terms and Conditions and the Privacy Statement
- R.23 The System must store the personal data relating to the Agronomist
- R.24 The System must allow Agronomist to login to DREAM by entering his/her email and password
- R.25 The System must allow Agronomist to visualize the data about his/her interested area in Telegana.
 - R.25.1 Agronomist must be able to visualize the location of response area.
 - R.25.2 Agronomist must be able to visualize the weather forecast
 - R.25.3 Agronomist must be able to visualize the water irrigation information
 - R.25.4 Agronomist must be able to visualize the soil humidity information
 - R.25.5 Agronomist must be able to visualize Farmer information
- R.26 The System must allow Agronomist to create the daily plan
 - R26.1 Agronomist must be able to indicate the title of daily plan
 - R26.2 Agronomist must be able to indicate date of daily plan
 - R26.3 Agronomist must be able to indicate relative farmer of daily plan
 - R26.4 Agronomist must be able to indicate the content of daily plan
- R.27 Agronomist must be able to visualize the detail of the daily plan
- R.28 Agronomist must be able to visualize the list of the daily plan
- R.29 The System must store the daily plan data
- R.30 The System must allow Agronomist to modify the daily plan
 - R.30.1 Agronomist must be able to modify the title of daily plan
 - R.30.2 Agronomist must be able to modify date of daily plan
 - R.30.3 Agronomist must be able to modify relative farmer of daily plan
 - R.30.4 Agronomist must be able to modify the content of daily plan
- R.31 The System must allow Agronomist to confirm the daily plan
 - R.31.1 Agronomist must be able to indicate the deviation
- R.32 Agronomist must be able to visualize the status of daily plan

- R.33 The System must allow Agronomist to answer the requests
- R.34 The System must allow Agronomist to visualize the list of the requests
- R.35 The System must allow Agronomist to visualize the details of the requests
- R.36 The System must store the answer to the request

3.2.5 Traceability Matrix

This section provides a summary of the mapping between the goals and the requirements that guarantee their achievement under some domain assumptions.

Goal ID	Domain Assumption ID	Requirement ID	Use Case ID
<i>G1</i>	<i>D.1, D.4, D.5, D.6, D.7, D.8, D.9, D.28</i>	<i>R.1, R.2, R.3, R.4, R.5, R.6</i>	<i>UC.14, UC.15, UC.16</i>
<i>G2</i>	<i>D.7, D.10, D.26, D.28</i>	<i>R.1, R.2, R.3, R.7,</i>	<i>UC.14, UC.15, UC.17</i>
<i>G3</i>	<i>D.1, D.2, D.3, D.7, D.29</i>	<i>R.8, R.9, R.10, R.11</i>	<i>UC.6, UC.7, UC.8</i>
<i>G4</i>	<i>D.4, D.7, D.22, D.25, D.29</i>	<i>R.8, R.9, R.10, R.14, R.15</i>	<i>UC.6, UC.7, UC.9</i>
<i>G5</i>	<i>D.4, D.7, D.21, D.23, D.29</i>	<i>R.8, R.9, R.10, R.12, R.13</i>	<i>UC.6, UC.7, UC.10, UC.11</i>
<i>G6</i>	<i>D4, D.7, D.24, D.29</i>	<i>R.8, R.9, R.10, R.16, R.17, R.18, R.19, R.20, R.21</i>	<i>UC.6, UC.7, UC.12, UC.13</i>
<i>G7</i>	<i>D.2, D.4, D.7, D.20, D.23 D.26, D.27</i>	<i>R.22, R.23, R.24, R.33, R.34, R.35, R.36</i>	<i>UC.1, UC.2, UC.10</i>
<i>G8</i>	<i>D.1, D.2, D.4, D.5, D.6 D.7, D.9, D.11, D.12, D.13, D.14, D.27</i>	<i>R.22, R.23, R.24, R.25</i>	<i>UC.1, UC.2, UC.18</i>
<i>G9</i>	<i>D.4, D.7, D.11, D.15, D.16, D.17, D.18, D.19, D.20, D.26, D.27</i>	<i>R.22, R.23, R.24, R.26, R.27, R.28, R.29, R.30, R.32</i>	<i>UC.1, UC.2, UC.3, UC.4</i>
<i>G10</i>	<i>D.4, D.7, D.18, D.19, D.20 D.21, D.26, D.27</i>	<i>R.22, R.23, R.24, R.29, R.31, R.32</i>	<i>UC.1, UC.2, UC.5</i>

3.3 Performance Requirements

The System does not have specific requirements on performances such as response time.

3.4 Design Constraints

3.4.1 Standards compliance

- Production name complies with Binomial Nomenclature.

3.4.2 Hardware limitations

- User's device must be able to exchange data via the internet.
- Farmer's device must be able to get location by GPS.

3.4.3 Any other constraint

- The System should be easy and intuitive to use, because farmers are not good at use electronic product.

3.5 Software System Attributes

3.5.1 Reliability

The System must be robust and fault tolerant.

3.5.2 Availability

The System should be up for 99% of the time which means that the average time between the occurrence of a fault and service recovery (MTTR or downtime) should be not greater than 3.65 days per year.

3.5.3 Security

Only encrypted communication protocols are allowed in the System. In addition, the sensitive data stored in the System must be encrypted. Finally, the System should be protected against intrusion from agents that are not authorized to access it.

3.5.4 Maintainability

The System should be organized in modules and be well documented, in order to make maintenance, upgrades and integration of new features easy.

3.5.5 Portability

The System must be accessible on different platforms including Windows and macOS. In addition, it should be available for iOS and Android devices.

4. FORMAL ANALYSIS USING ALLOY

4.1 Alloy model

The Alloy model focuses on the following aspects of DREAM:

- Farmer's problem handling
- Agronomist's responsible area
- Agronomist's daily plan

In particular, some aspects highlighted by the model are the following:

- Agronomist only handles requests in his/her responsible area.
- Agronomist only makes daily plan involved farmers in his/her responsible area.
- PolicyMaker email must be unique.
- Agronomist email must be unique.
- Farmer phone number must be unique

4.1.1 Signatures

Here we present all the signatures composing the model.

```
-- Farm: farm in Telegana
sig Farm {}

-- Area: area in Telegana
sig Area {
    farms: some Farm
}

-- Float: positive float number
sig Float {
    left: one Int,
    right: one Int
}{
    left >= 0
    right >= 0
}

-- Report: production report
sig Report {
    -- amount: production amount
    amount: Float
}

-- ProblemStatus: the status of a problem
abstract sig ProblemStatus{}

-- Open: the problem is created by a farmer
one sig Opened extends ProblemStatus {}

-- Processing: the problem is answered by a agronomist
one sig Processing extends ProblemStatus {}
```

```

-- Closed: farmer gives the problem a feedback and close it.
one sig Closed extends ProblemStatus {}

-- Problem
sig Problem {
  -- feedback: the score is given to the solution for the problem
  feedback: lone Int,
  -- status: the status of a problem
  status: one ProblemStatus
}{
  feedback >= 0
}

-- Date
sig Date {}

-- Daily Plan
sig DailyPlan {
  date: one Date,

  -- farmer: which farmers included in this daily plan
  farmer: set Farmer
}

-- Time
sig Time {
  -- ts: timestamp
  ts: Int
}{ ts > 0}

-- Comment:
sig Comment {
  time: one Time
}

sig Post {
  time: one Time,
  comments: set Comment
}

-- Forum: in which Farmer can discuss with each other
one sig Forum {
  posts: set Post
}

-- PhoneNumber: phone number for farmer
sig PhoneNumber {}

-- AgEmail: email for agronomist
sig AgEmail {}

-- PmEmail: email for policymaker

```

```

sig PmEmail {}
-- User: who use DREAM
abstract sig User{}

-- Farmer: a farmer that uses DREAM
sig Farmer extends User{
  phonenumber: one PhoneNumber,
  performance: Int,

  own: one Farm,
  -- reports: set of production report
  reports: set Report,
  -- requests: Farmer's problem list
  requests: set Problem
}{}
-- If a Farmer does not report his/her production information,
-- his/her performance cannot be identified and be use the default value 0
#reports = 0 => performance = 0
-- performance is positive implies Farmer reports at least one production info
performance > 0 => #reports > 0
}

-- Agronomist: an agronomist that uses DREAM
sig Agronomist extends User {
  email: one AgEmail,
  -- area: the area an agronomist is responsible of. Agronomist is responsible for only one area
  area: one Area,

  dailyplans: set DailyPlan,
  answers: set Problem
}

-- PolicyMaker: a policy maker that uses DREAM
sig PolicyMaker extends User {
  email: one PmEmail
}

```

4.1.2 Facts

All the constraints needed to build the model are defined in this section.

-- Every phonenumber is associated with one and only one Farmer

```

fact phoneNumberBelongsToAFarmer {
    all pn: PhoneNumber | one f: Farmer | f.phoneNumber = pn
}

-- Every policymaker email is associated with one and only one PolicyMaker
fact emailBelongsToAPolicyMaker {
    all pe: PmEmail | one pm: PolicyMaker | pm.email = pe
}

-- Every agronomist email is associated with one and only one Agronomist
fact emailBelongsToAnAgronomist {
    all ae: AgEmail | one a: Agronomist | a.email = ae
}

-- Every Comment is associated with one and only one Post
fact commentBelongsToAPost {
    all c: Comment | one p: Post | c in p.comments
}

-- A Comment cannot be earlier than associated Post
fact noCommentEarlierThanPost {
    all p: Post | all c: p.comments | c.time.ts > p.time.ts
}

-- Every Report is associated with one and only one Farmer
fact reportBelongsToAFarmer {
    all r: Report | one f: Farmer | r in f.reports
}

-- Every Farm is associated with one and only one Farmer
fact farmBelongsToAFarmer {
    all fm: Farm | one f: Farmer | f.own = fm
}

-- Every farm is associated with one and only one Area
fact farmBelongsToAnArea {
    all fm: Farm | one a: Area | fm in a.farms
}

-- Every Problem is associated with one and only one Farmer
fact problemBelongsToAFarmer {
    all p: Problem | one f: Farmer | p in f.requests
}

```

```

-- Every Problem is associated with no more than one Agronomist
fact problemBelongsToAFarmer {
    all p: Problem | lone a: Agronomist | p in a.answers
}

-- Every Area is associated with one and only one Agronomist
fact areaBelongsToAnAgronomist {
    all a: Area | one ag: Agronomist | ag.area = a
}

-- Every Daily Plan is associated with one and only one Agronomist
fact dailyplanBelongsToAnAgronomist {
    all dp: DailyPlan | one ag: Agronomist | dp in ag.dailyplans
}

-- Every Date is associated with one and only one Daily Plan
fact dateAssociatedWithADailyplan {
    all d: Date | one dp: DailyPlan | dp.date = d
}

-- A Farmer's request can only be handled by the Agronomist who is responsible for the area where is the
Farmer's farm in.
fact problemAnswerByResponsibleAgronomist {
    all f: Farmer | all p: f.requests | (p.status in Processing or p.status in Closed)
    implies
    one a: Agronomist | p in a.answers and f.own in a.area.farms
}

-- A problem is processing if and only if it is associated to an agronomist and without feedback
fact processingProblem {
    all p: Problem |
    p.status in Processing
    iff
    all a: Agronomist |
    p in a.answers and p.feedback = 0
}

-- A problem is closed if and only if it is associated to an agronomist and with feedback
fact closedProblem {
    all p: Problem |
    p.status in Closed
    iff
    all a: Agronomist |
    p in a.answers and p.feedback > 0
}

-- A problem is opened if and only if it is not associated to any agronomist and any feedback
fact openedProblem {

```

```

    all p: Problem |
      p.status in Opened
      iff
        all a: Agronomist |
          p not in a.answers and p.feedback = 0
  }

-- Agronomist's dailyplans only involve the farmers in he/she responsible area
fact DPfarmerInAgronomistResponsibleArea {
  all a: Agronomist | all dp: a.dailyplans | #dp.farmer > 0
  implies
    all f: dp.farmer | f.own in a.area.farms
}

```

4.1.3 Assertions

Finally, the following assertions verify if the built model respects some of the most important properties that the System must have.

```

-- Every email uniquely identifies a registered agronomist
assert uniqueAgronomistEmail {
  all disj a1, a2: Agronomist | a1.email != a2.email
}
check uniqueAgronomistEmail

-- Every email uniquely identifies a registered policymaker
assert uniquePolicyMakerEmail {
  all disj pm1, pm2: PolicyMaker | pm1.email != pm2.email
}
check uniquePolicyMakerEmail

-- Every phone number uniquely identifies a registered farmer
assert uniqueFarmerPhoneNumber {
  all disj f1, f2: Farmer | f1.phonenumber != f2.phonenumber
}
check uniqueFarmerPhoneNumber

-- Agronomist only handles request from farmers in his/her responsible area
assert agronomistAnswerResponsibleFarmer {
  all a: Agronomist | all p: a.answers | all f: Farmer | p in f.requests
  implies
    f.own in a.area.farms
}

```

```
}  
check agronomistAnswerResponsibleFarmer
```

4.1.4 Analysis results

This subsection shows the results of the checked performed on the assertions.

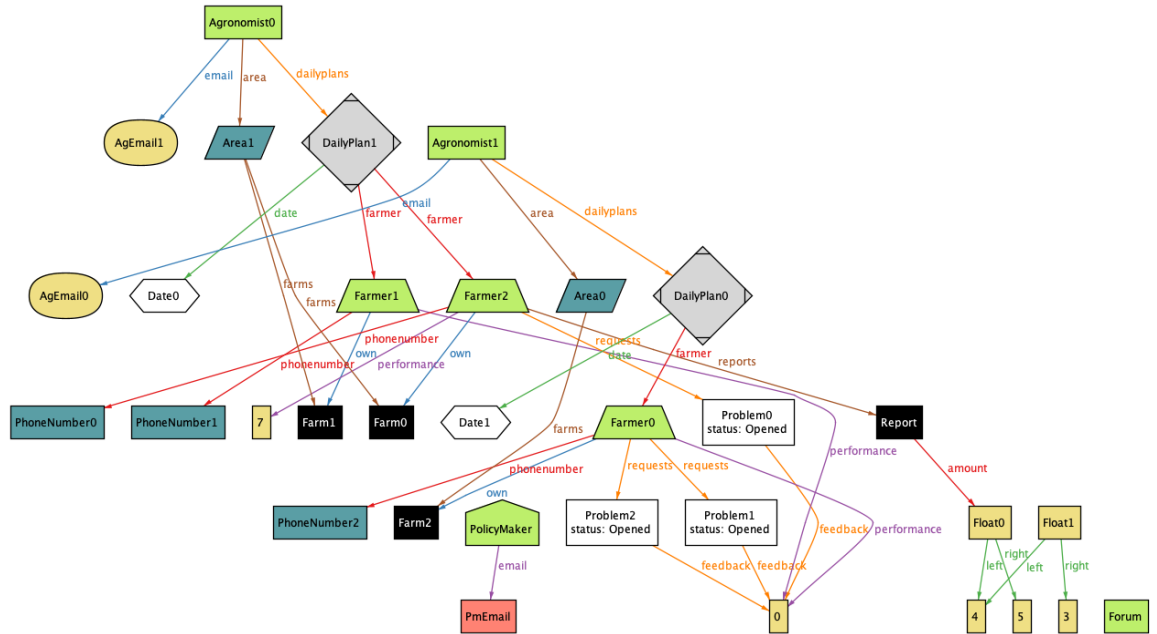
4 commands were executed. The results are:

```
#1: No counterexample found. uniqueAgronomistEmail may be valid.  
#2: No counterexample found. uniquePolicyMakerEmail may be valid.  
#3: No counterexample found. uniqueFarmerPhoneNumber may be valid.  
#4: No counterexample found. agronomistAnswerResponsibleFarmer may be valid.
```

4.2 Alloy generated Worlds

- **Simple case:** in this world, shown below, models a simple scenario in which two agronomists and three farmers are registered on DREAM. In the instant shown, one agronomist's responsible area contains one farmer, and the other one contains two farmers. All opened problems have not been sent to the responsible agronomist yet. Besides, agronomist's daily plans only involve the farmer in their responsible area.

```
pred simpleCase {  
    #Farmer = 3  
    #Agronomist = 2  
    #DailyPlan = 2  
    #Problem > 2  
}  
run simpleCase for 6
```

5. EFFORT SPENT:

[illegible]

--	--	--	--	--

6. REFERENCES

- E.Di Nitoo. Lecture Slides. Politecnico di Milano.
- E.Di Nitoo. Project Assignment AY 2021-2022. Politecnico di Milano.
- ISO/ISEC/IEEE 29148:2011. Standard on requirement engineering.
<http://standards.ieee.org/standard/29148-2011.html>.