Politecnico di Milano

DREAM

Requirements Analysis and Specification Document

Jiayan Cui
10756377
Yudong Wang
10829382

January 5, 2022

POLITECNICO
MILANO 1863

Contents

1. Introduction

1.1 Purpose

The COVID-19 causes many challenges to our society. For the agriculture, it has greatly highlighted the massive disruption caused in food suppl y chains exposing the vulnerabilities of marginalized communities, small holder farmers and the importance of building resilient food systems. In addition, climate changes also impact everything from productivity to livelihoods across food and farm systems and is predicted to result in a 4%-26% loss in net farm income towards the end of the century. Therefore, it is important for us to develop innovative methodologies and technologies that can help bolster countries against food supply shocks and challenges.
This is why we develop DREAM, an application helped not only farmer, but also agronomists and policy makers to work together so that they can stabilize their food supply.
The goal of this document is to show a fully system description about the main goals, functionalities and scenarios, then going more into details describing the functional and nonfunctional requirements that the system should be fulfill.

1.1.1 Goals

ID	Goal
G1	Allow policy maker to get information provided by the farmers about their production
G2	Allow policy maker to get farmers' production ranking.
G3	Allow farmers to get relevant weather forecasts and suggestions based on the farmer's location and production information.
G4	Allow farmers to upload their production and problems.
G5	Allow farmers to create discussion forums with the other farmers.
G6	Allow farmers to respond to other farmers' questions.
G7	Allow agronomists to respond to the farmers' questions.

1.2 Scope

1.2.1 Product

DREAM is an organic system of policy producers, agronomists and farmers to solve farmers' land yield problems, report problems to policy makers, agronomists and get feedback, and DREAM is also a forum among farmers.

DREAM allows policy producers to go into the data reported by farmers to rank them and thus identify those farmers who are performing well, especially if they show resistance to adverse weather events, as these farmers will be specially rewarded and asked to provide useful best practices to other farmers. Identify those farmers who need help because they are performing particularly poorly. Find out if the mentoring initiatives carried out by agronomists with the help of good farmers have produced significant results.

As for farmers, DREAM can visualize data relevant to them based on their location and type of production - for example, weather forecasts, personalized advice on planting specific crops or using specific fertilizers. DREAM can also allow farmers to insert information about their own products as well as post problems they are having and allow them to ask for help from agronomists or other farmers.

1.2.2 World and Shared phenomena

1.2.2.1 World phenomena

ID	Phenomenon
WP1	Farmers record information about their production.
WP2	Farmers use the recommendations to optimize their crop plans
WP3	The government wants to help farmers improve their production through technical guidance.
WP4	Policy makers need to analyze farmers' performance to identify good and poor performers.
WP5	Policy makers need to let agronomists and good-production farmers to develop policies helping poor-performance farmers.
WP6	Agronomists develop plan to help poorly Performing farmers.
WP7	High-performing farmers develop plan to help low-performing farmers.

1.2.2.2 Shared phenomena - controlled by the World

ID	Phenomenon
SP1	Farmers upload their own production information.
SP2	Farmers discuss in the community section in DREAM.
SP3	Agronomists guide farmers on their problems in DREAM.
SP4	Policy makers get ranked by farmers' performance.
SP5	Policy makers need to discern and analyze whether policies are effective or not.
SP6	Agronomists answer questions from farmers in the DREAM system.
SP7	Policy makers receive farmer performance rankings in the DREAM system.

1.2.2.3 Shared phenomena - controlled by the Machine

ID	Phenomenon
SP8	The system ranks the performance of farmers.
SP9	The system shows the farmers' performance ranking.
SP10	The system shows the production information of farmers.
SP11	The system receives production information uploaded by farmers.
SP12	The system receives weather forecast information.
SP13	The system provides weather forecast information to farmers.
SP14	The system feeds the agronomist's recommendations to the farmer.

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

GPS A system that can determine the exact location

Result Judgment Whether the steering initiatives have produced significant results or not.

Ranking Farmer rankings based on farmer production and weather conditions.

1.3.2 Acronyms

DREAM Data-Driven Predictive Farming in Telangana.

G Goal.

IT device Information Technology device.

R Requirement.

RASD Requirements Analysis and Specification Document.

SP Shared Phenomenon.

WP World Phenomenon.

1.3.3 Abbreviations

1.4 Revision history

Version	Date	Notes
V1.0	December 10, 2021	Initial release.
V1.1	December 23, 2021	Some corrections to UML diagrams.

1.5 Reference documents

- **Alloy documentation**
- **R&DD Assignment AY 2021-2022**
- **UML documentation**

1.6 Document structure

This document is structured in the following way:

1. The first chapter is an introduction and overview of the project, setting the context that led to its development, the goals to be achieved, and a general description of its functionality.
2. The second chapter is a formal description of the domain model and the project through the extensive use of class diagrams and state machine diagrams. The class diagram provides a high-level description of the domain entities and their relationships, while the state machine diagram focuses on modeling the most important entities through their state transitions. All functional requirements and domain assumptions are also presented here to achieve the previously stated goals.
3. The third chapter presents the non-functional requirements, which are deepened thanks to the description of possible use cases using natural language and sequence or activity diagrams, and illustrates the design constraints.
4. The fourth and final chapter presents a formal analysis of the model through the use of the open source Alloy language and tool. Some of the configurations created by the tool are included.

2. Overall description

2.1 Product perspective

2.1.1 Class diagram

A UML class diagram describing the main entities involved in the system follows.

Farmers use the system to get weather information for their current location and to get advice on planting crops. Farmers can also create discussion forums like a forum where they can upload their current problems in the system.

Telengana's policy makers have access to the farmer performance rankings in the DREAM system to distinguish between farmers who are performing well and those who are not. The effectiveness of the policy is determined by comparing the ranking of the badly performing farmers over time.

Agronomists can answer farmers' questions and get the current weather data of the region and the best performing farmers in the system. They can also access local farmers according to the daily plan and update the daily plan daily.

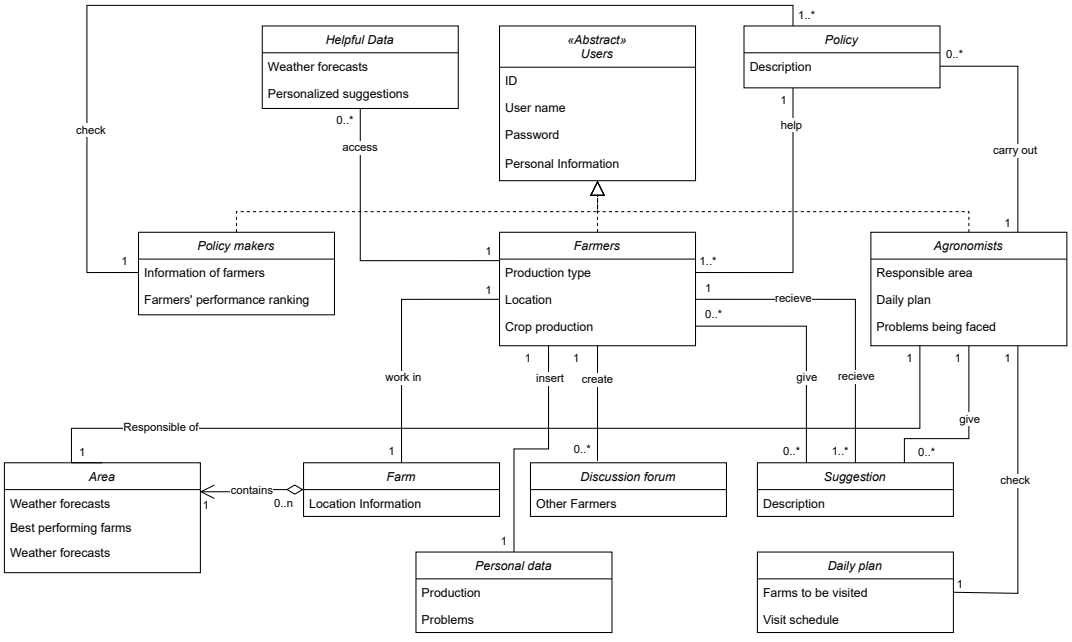


Figure 2.1: Class Diagram

2.1.2 State chart diagrams

The internal state of the main entities of the domain is better defined in the following UML state diagrams.

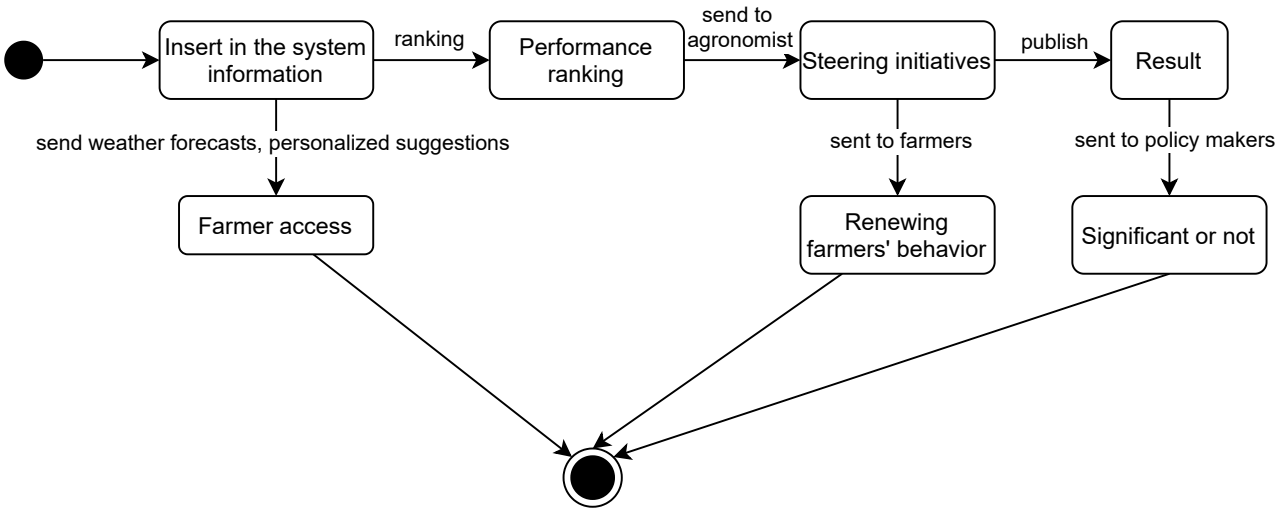


Figure 2.2: Statechart.

2.1.3 Scenarios

2.1.3.1 Giving special incentive to the performing well farmers.

Martini, a policy maker at Telengana, noticed that the weather conditions in the area where Farm A is located were not very good in recent days, with unstable temperatures, but Farm A had the highest production. After observing this phenomenon, he contacted Alex, the farmer in charge of Farm A, through the DREAM system, and rewarded him for being able to maintain high production despite the bad weather conditions, and encouraged Alex to share his farming experience.
Alex is happy to receive the special incentives and share his experience to every farmer on the DREAM.

2.1.3.2 Helping farmer who needs help

Ann, also a policy maker at Telengana, noticed from the DREAM ranking table that the production of the farms managed by farmer Bob were far below the normal production. Noticing this phenomenon, she promptly contacted Bob and directed him to connect with the appropriate agronomists and quality farmers to provide him with technical guidance on planting the appropriate crops. At the same time, Ann added Bob to her watching list to monitor Bob's production in real time.
Two months later, Bob's production were significantly higher than average.

2.1.3.3 Helping farmers to solve their problems

Mike owns a cornfield. In preparation for planting, Mike logged into his DREAM account, checked the weather information for his field, followed the system's recommendations for planting, and prepared the type of fertilizer that the system suggested. About a month after planting, Mike found that the corn in the field was not growing as expected. He opened DREAM and described his problem. Mike learned that the unsuitable humidity was

causing the corn to grow too slowly, and he followed the advice in the forum to adjust the humidity of the land, and finally the corn grew at a normal rate.
In the end he got more harvest than ever before.

2.1.3.4 Discussing with other farmers

Scarlett owned some fields, but she didn't know what kind of crops were appropriate to grow. She thought of DREAM, a software specifically for farmers, and that she might be able to get some useful information from the software. So she registered and logged into her DREAM account, opened the forum, shared information about the location of her farmland in the forum, and asked other farmers for their opinions. Soon there were many farmers sharing their experiences and opinions. Scarlett selected the opinions of most of these farmers and decided to grow soybeans on her own farm because they were best suited to the conditions of her field.

2.1.3.5 Agronomist supervision

Federico is a well-known agronomist who was invited to join the DREAM platform to help improve regional crop yields. One day he noticed that the eggplants planted by farmer Gary were rotting heavily, affecting the production of the farm. He looked at the weather conditions in Gary's area, the humidity of Gary's land and Gary's production information and discovered the problems Gary was having with the planting process. He reached out to Gary through the DREAM platform and instructed Gary to change his planting method.
Gary finally managed to save his eggplant field.

2.2 Product functions

The main production functions are identified as follows:

Telengana's policy makers will be able to:

- Register with their work number.
- Viewing production Ranking.
- Seeing the production of farmer who helped by agronomists.

Farmers will be able to:

- Register with personal data.
- Visualize data relevant to them, based on their location and type of production.
- Upload their production.
- Create discussion forums with the other farmers.
- Request for help and suggestions by agronomists and other farmers.

Agronomists will be able to:

- Register providing a set of areas of responsibility.
- Receive helping request from farmers and seeing their related situations.

2.3 User characteristics

The following users are addressed by the system:

- **Telengana's policy makers**

  Register as an administrator and have access to view detailed data on farmer production and farmers who have been helped.

- **Farmers**

  Register with personal data, upload produce information and production, get help in the app, and post questions in the forum.

- **Agronomists**

  Register with areas of responsibility and help farmers who are facing problems.

2.4 Assumptions, dependencies and constraints

2.4.1 Domain assumptions

ID	Domain assumption
D1	The number of policy makers is sufficient to identify and reward all farmers who rank high.
D2	Policy makers will help every farmer who asks for help.
D3	All farmers are truthful and regularly upload their production.
D4	All farmers will allow the software to access farm location information.
D5	Farmers will answer questions they know in the forum.
D6	Agronomists will provide adequate help to farmers who need it.
D7	All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

2.4.2 Constraints

In order to improve the ease of use of the system, all users will not authenticate to the system. For all the functionalities that require to know the identity of the users, the system will use:

- when users access the system with an IT device, a device identifier that the system generates and binds to the device itself
- when users access the system with a standard telephone line, the caller identifier of the telephone line (telephone number)

For the authentication of policy makers, instead, the system will integrate with identity systems.

3. Specific requirements

3.1 External interface requirements

3.1.1 Hardware interfaces

Policymakers need to have an Internet-enabled IT device to access the DREAM system for management. Farmers need GPS-enabled mobile devices to locate the managed fields and to access the weather conditions of their areas from the Internet.

Also, the corresponding farmland should have a moisture detector to detect soil moisture. The detector should have Internet capability to transmit monitoring data back to the software in real time. The detector can be powered by a mobile power source such as a battery, as it is difficult to achieve a fixed power source in a farm field.

3.1.2 Software interfaces

The weather forecast function is very important in order to help farmers to better improve their production. The software provides an API to obtain weather forecasts for short and long term weather forecasts provided by Telengana website.

In the mean time, the system should provide an external interface to provide location services to determine the location information of the user's farm. The client should remind the user of all the APIs used in the system during the registration process.

3.2 Functional requirements

3.2.1 Mapping on Requirements

- **G1**:Allow policy maker to get information provided by the farmers about their production.
  — **R1**:The system shall allow an unregistered policy maker to register.
  — **R2**:The system should be able to store the production information offered by farmers.
  — **R3**:The system should allow policy makers to extract the production information.
  — **R4**:The system should get farmers' GPS information and weather conditions there to show to policy makers.
  — **D3**:All farmers are truthful and regularly upload their production.
  — **D3**:All farmers are truthful and regularly upload their production.
  — **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

- **G2**:Allow policy maker to get farmers' production ranking.

  — **R3**:The system should allow policy makers to extract the production information.

  — **R4**:The system should get farmers' GPS information and weather conditions there to show to policy makers.

  — **R5**:The system should ranking farmers' production and show to policy makers.

  — **R6**:The system should let policy makers know how to contact each farmer.

  — **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

- **G3**:Allow farmers to get relevant weather forecasts and suggestions based on the farmer's location and production information.

  — **R7**:The system shall allow an unregistered farmer to register.

  — **R8**:After the farmer has successfully entered all the information required for registration, the system will send him/her a cell phone verification code to complete the registration process.

  — **R9**:The system must remember each farmers' field location.

  — **D1**:The number of policy makers is sufficient to identify and reward all farmers who rank high.

  — **D2**:Policy makers will help every farmer who asks for help.

  — **D4**:All farmers will allow the software to access farm location information.

- **G4**:Allow farmers to upload their production and problems.

  — **R10**:The system must remember each farmers' problems data.

  — **R11**:The system must allow farmers to upload different type of problem, such as picture, text and video.

  — **R12**:The system shall rank problems by uploading time.

  — **R13**:The system shall allow farmers to delete their problems on forums.

  — **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

- **G5**:Allow farmers to create discussion forums with the other farmers.

  — **R14**:The system will allow all farmers and all agronomist having access to all problems.

  — **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

- **G6**:Allow farmers to respond to other farmers' questions.

  — **R13**:The system shall allow farmers to delete their problems on forums.

  — **R14**:The system will allow all farmers having access to all problems.

  — **D5**:Farmers will answer questions they know in the forum.

  — **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

- **G7**:Allow agronomists to respond to the farmers' questions.

  — **R15**:The system will allow an unregistered Agronomist to register..

  — **R16**:After the agronomist has successfully entered all the information required for registration, the system will send him/her a cell phone verification code to complete the registration process.

— **R17**:The system will allow all agronomists to respond to the farmers' questions.

— **R18**:The system will allow all agronomists to have access to all farmer's detail information such as production, location weather and so on.

— **R17**:The system will allow all agronomists to respond to the farmers' questions.

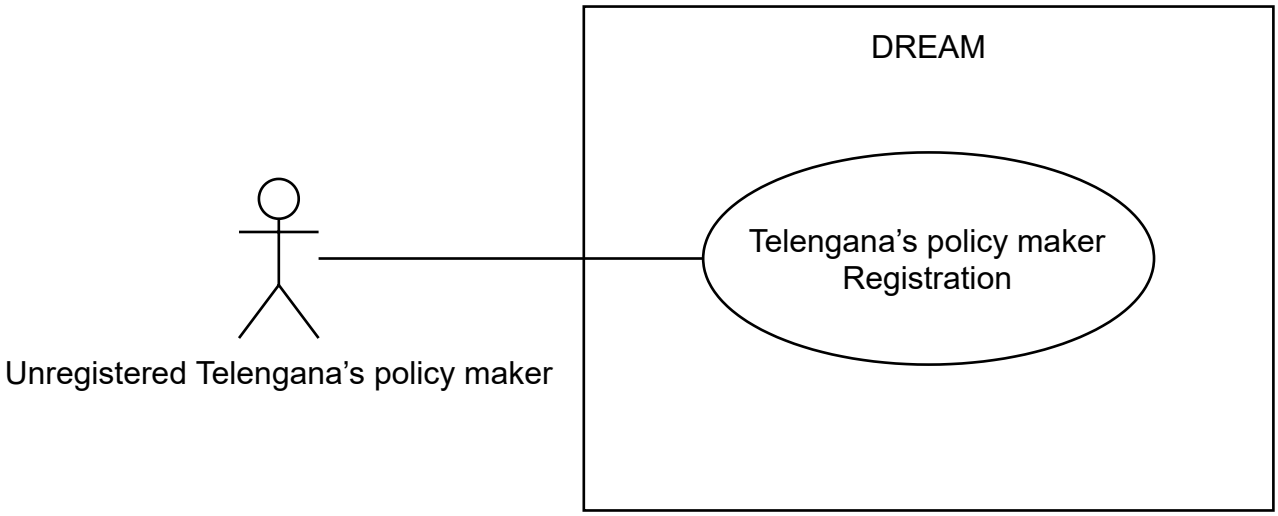
— **D3**:All farmers are truthful and regularly upload their production.

— **D6**:Agronomists will provide adequate help to farmers who need it.

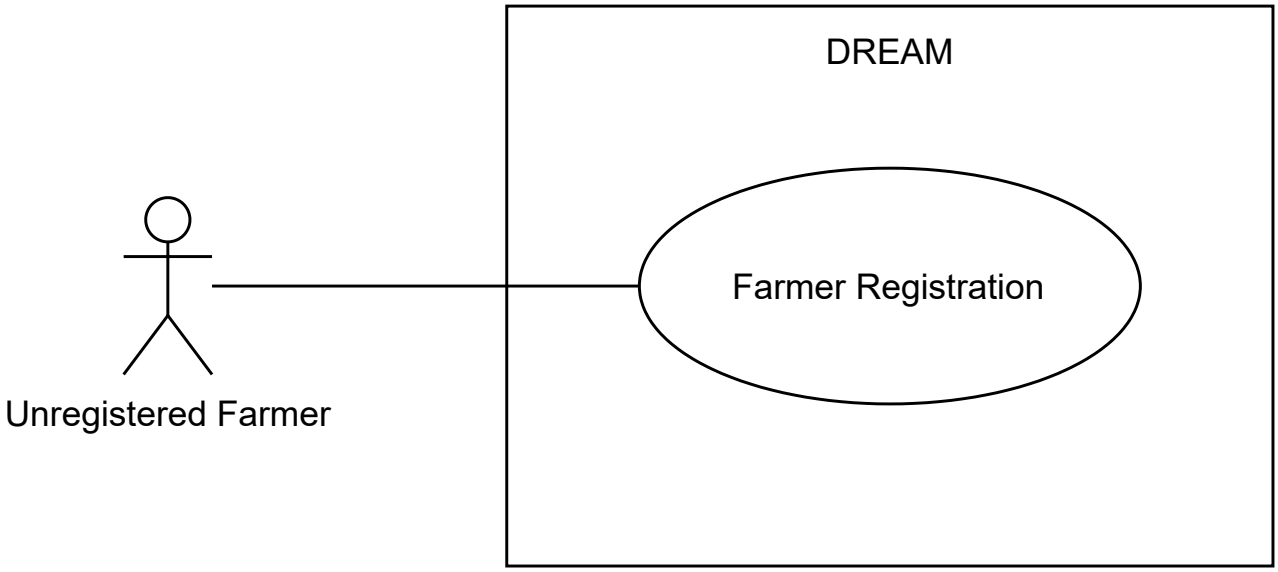
— **D7**:All DREAM users have an IT device with support for Internet connectivity or a standard telephone line.

3.2.2 Use case diagrams

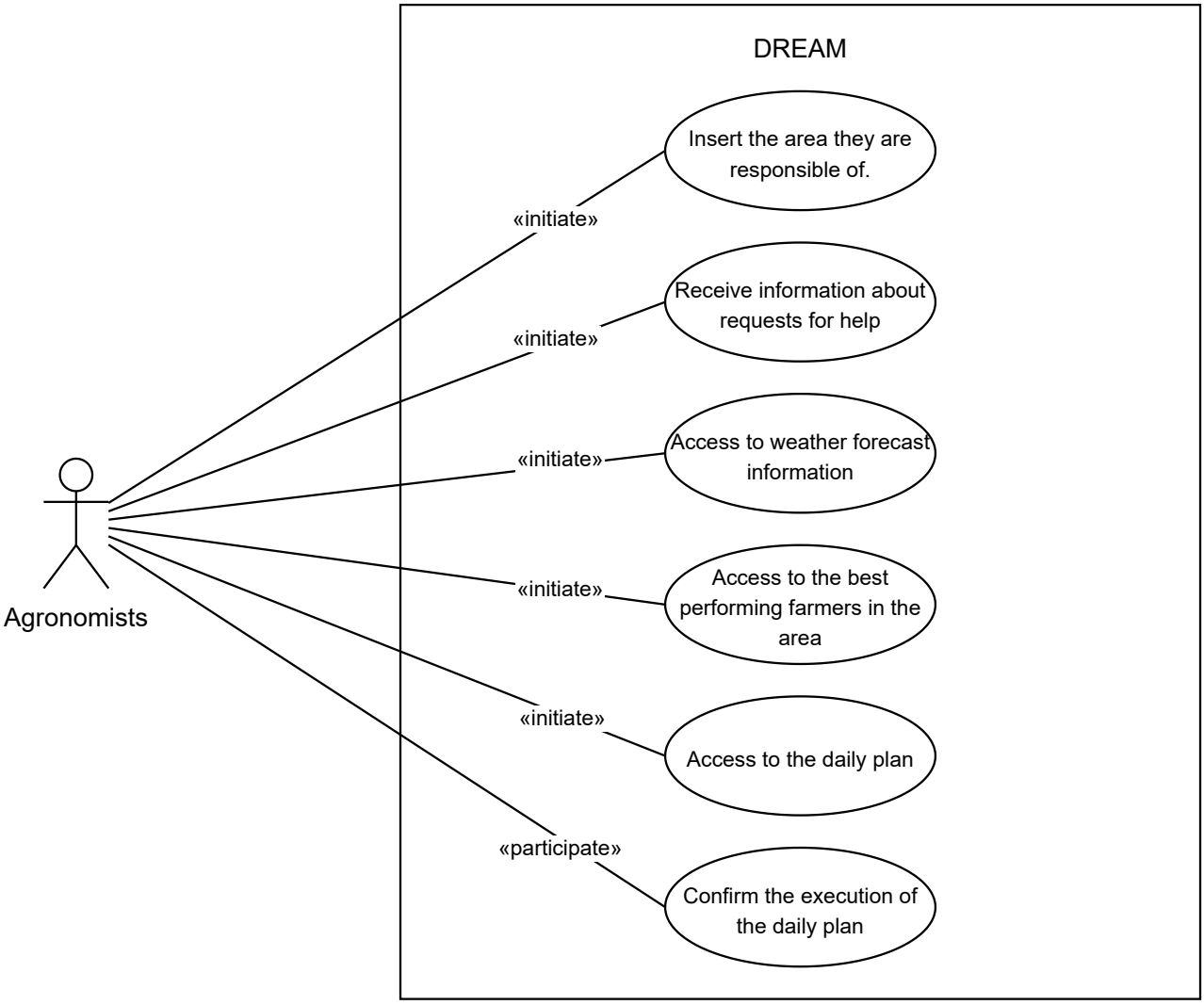
- **unregistered policy maker**



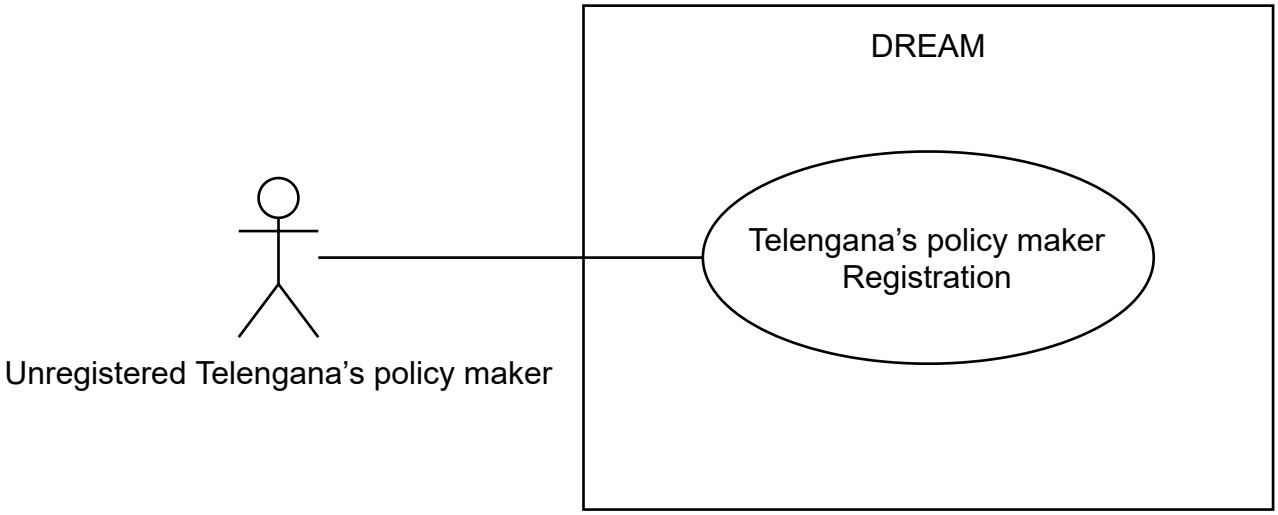
- **unregistered farmer**



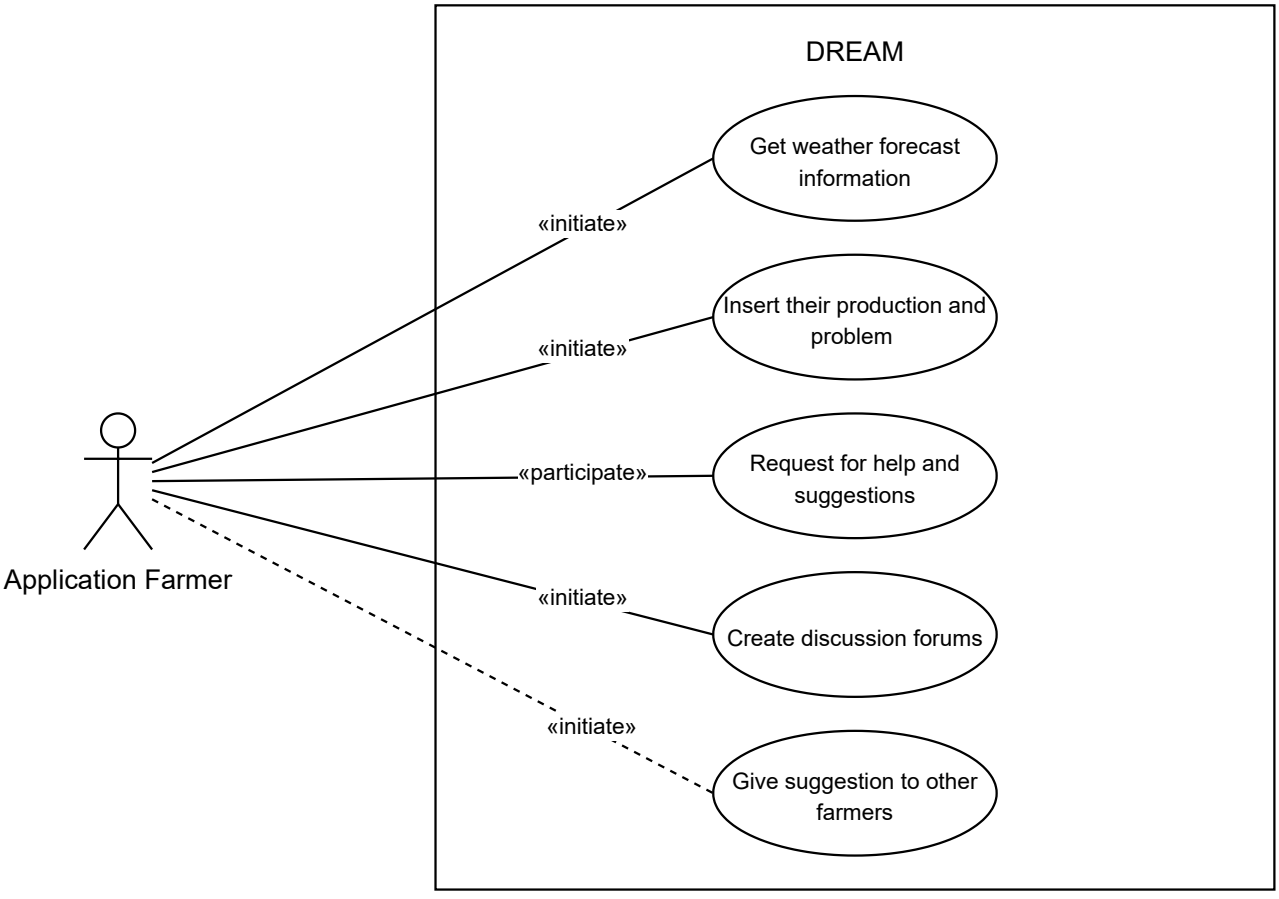
- **unregistered agronomist**



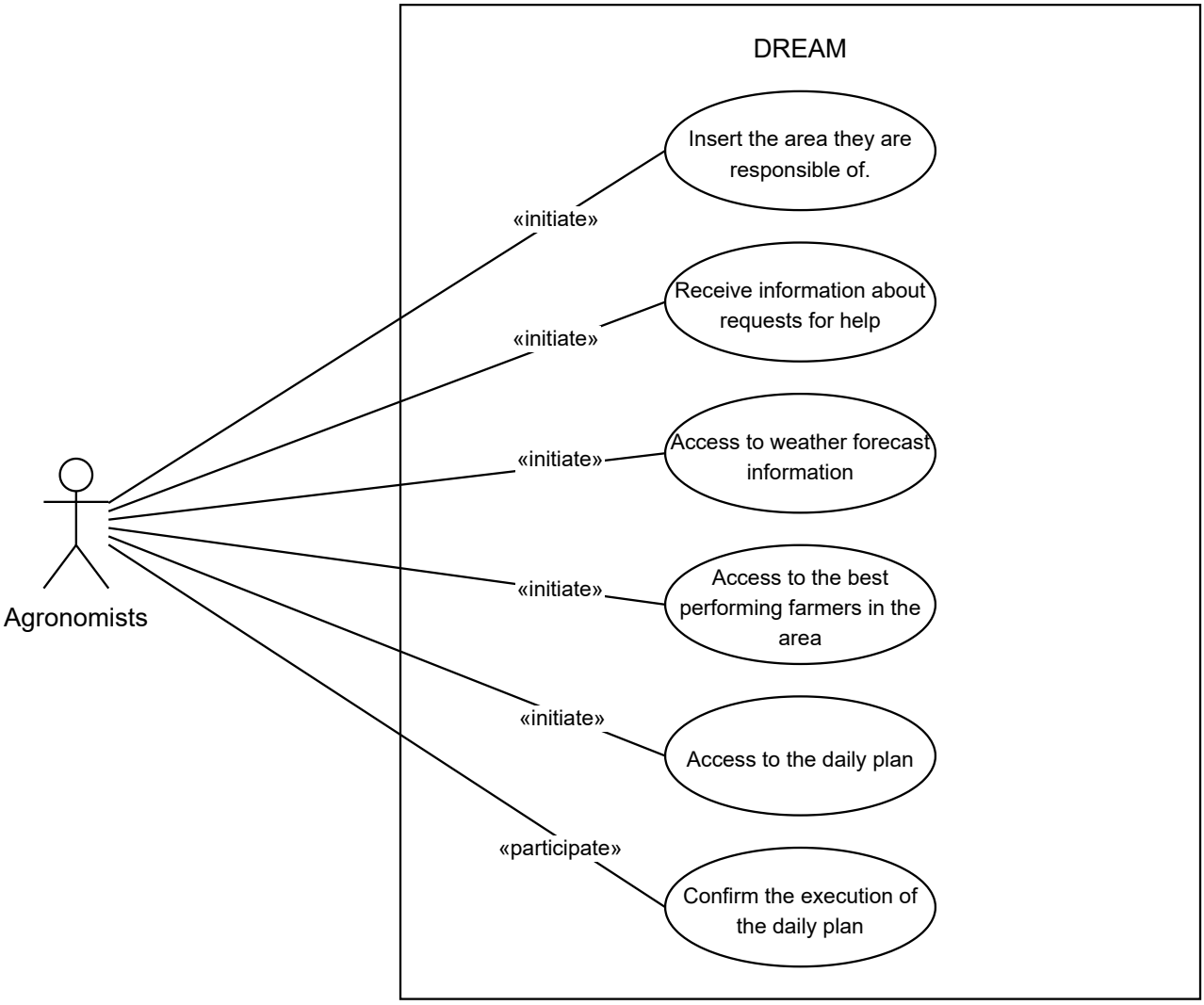
- **application policy maker**



- **application farmer**



- **application agronomist**


DREAM
Insert the area they are responsible of.
«initiate»
Receive information about requests for help
«initiate»
Access to weather forecast information
«initiate»
Access to the best performing farmers in the area
«initiate»
Access to the daily plan
«participate»
Confirm the execution of the daily plan
Agronomists

3.2.3 Use cases and Sequence diagrams

3.2.3.1 Policy maker registration

Use Case	Policy maker registration
Actor	Policy maker
Entry condition	Policy maker wants to use DREAM system to manage farmers.
Flow of events	• Policy maker start to sign up. • The system shows sign up view. • Policy maker create credentials(username, password) • The system will check username if it is available. • The system shows to policy makers user details • Policy maker confirm sign up.
Exit condition	The system shows a confirmation message to policy maker.
Exceptions	If the username has been used by other policy maker, system will let you to change another username.

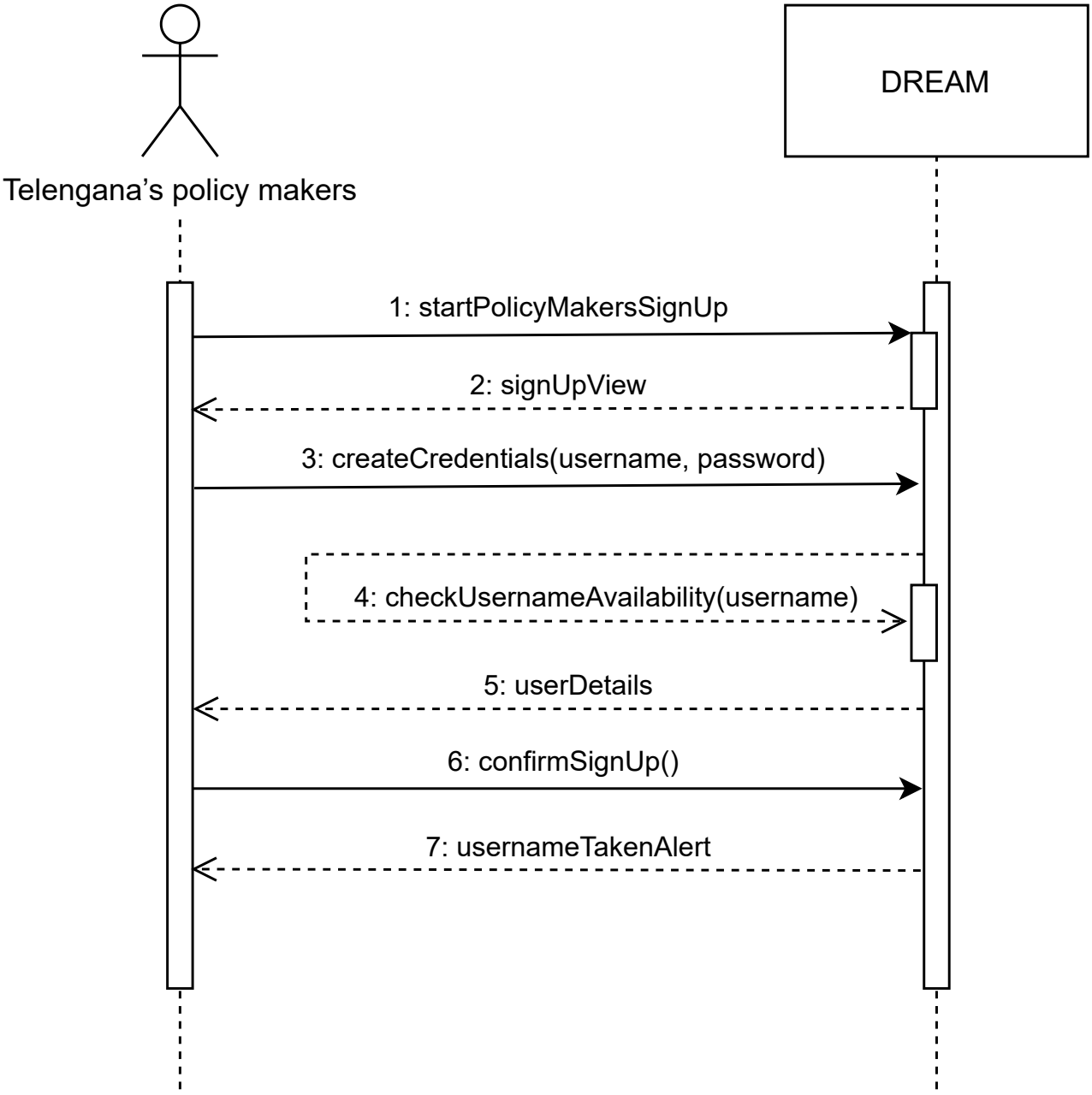


Figure 3.1: Policy maker registration sequence diagram

3.2.3.2 Farmer registration

Use Case	Farmer registration
Actor	Farmer
Entry condition	The farmer wants to use DREAM.
Flow of events	• The farmer start to sign up. • The system shows sign up view. • The farmer create credentials(username, password) • The system will check username if it is available. • The system shows to farmer user details • The farmer confirm sign up.
Exit condition	The system shows a confirmation message to farmer
Exceptions	If the username has been used by other farmer, system will let you to change another username.

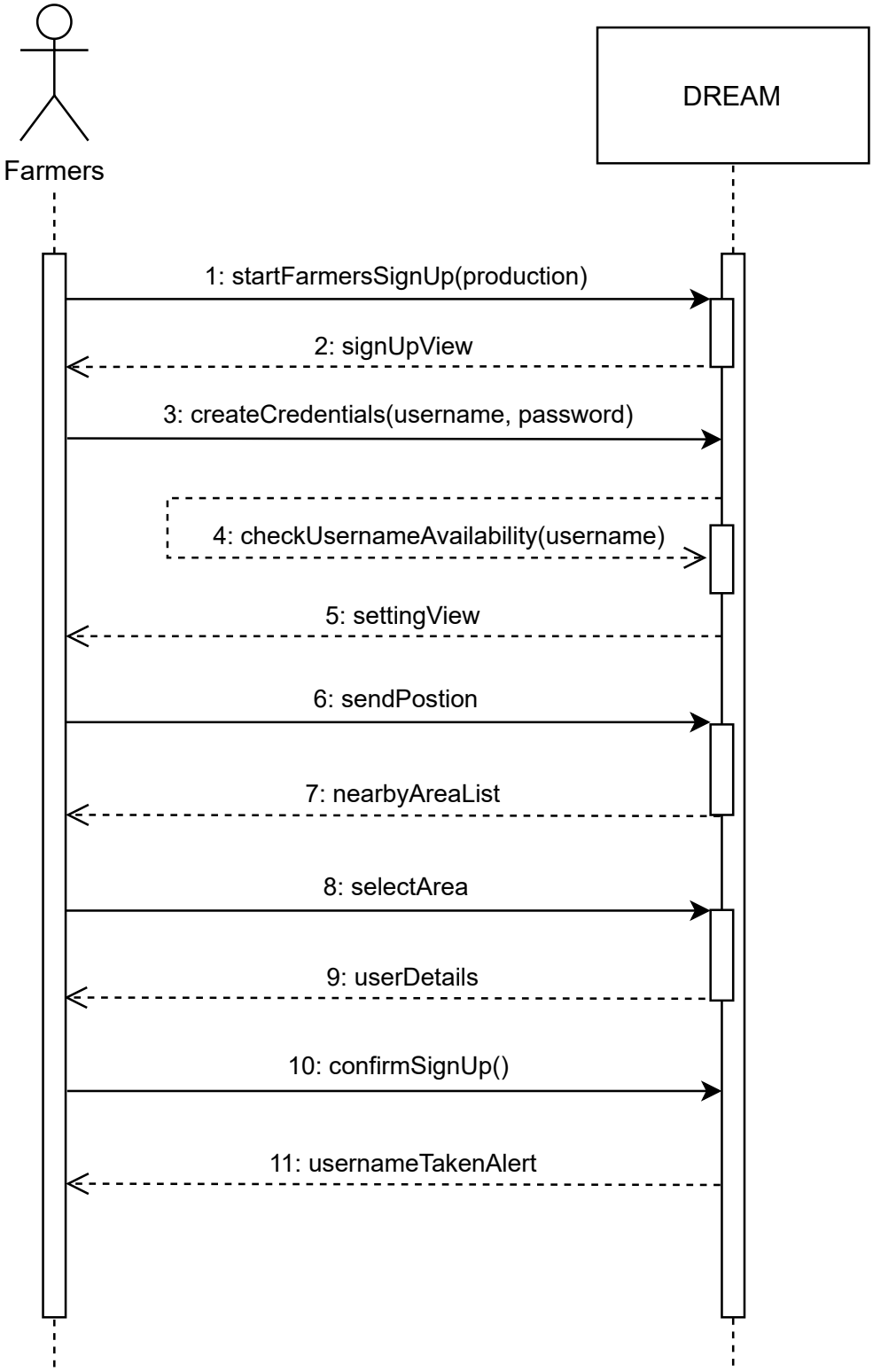


Figure 3.2: Farmer registration sequence diagram

3.2.3.3 Agronomist registration

Use Case	Agronomist registration
Actor	Agronomist
Entry condition	A customer wants to cancel their reservation
Flow of events	• The agronomist start to sign up. • The system shows sign up view. • The agronomist create credentials(username, password) • The system will check username if it is available. • The agronomist will send his/her location to DREAM. • The system will send a nearby List to agronomist. • The agronomist will choose a certain Area. • The system shows to agronomist user details • The agronomist confirm sign up.
Exit condition	The system shows a confirmation message to agronomist
Exceptions	If the username has been used by other agronomist, system will let you to change another username.

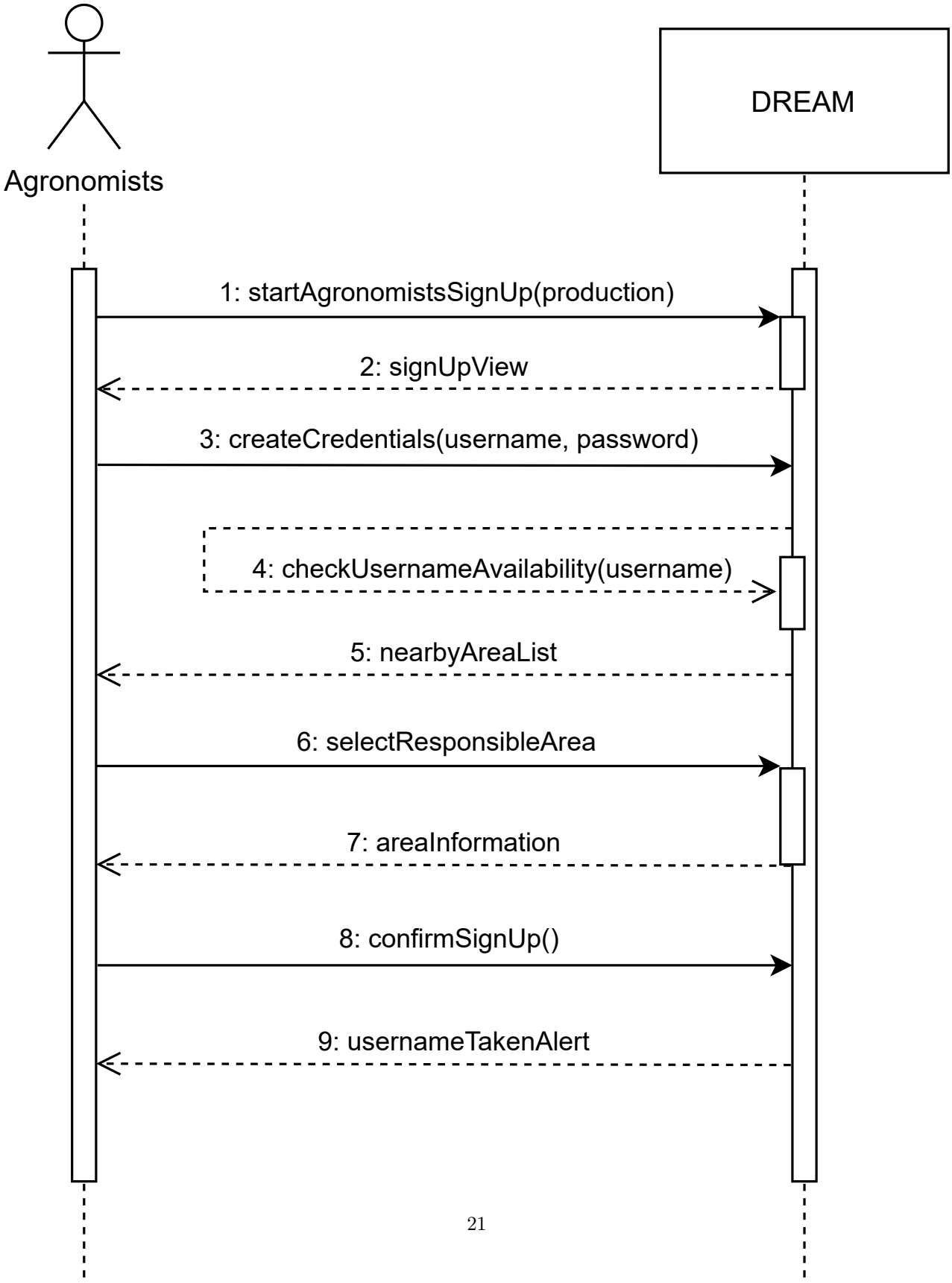


Figure 3.3: Agronomist registration sequence diagram

3.2.3.4 Identify Good Farmers

Use Case	Identify Good Farmers
Actor	Policy makers
Entry condition	Policy maker wants to know which farmer done will at their production
Flow of events	• Policy maker open ranking page. • The system shows to policy maker which farmer performing well, and shows their information.
Exit condition	Policy maker got information about well-performing farmers.
Exceptions	

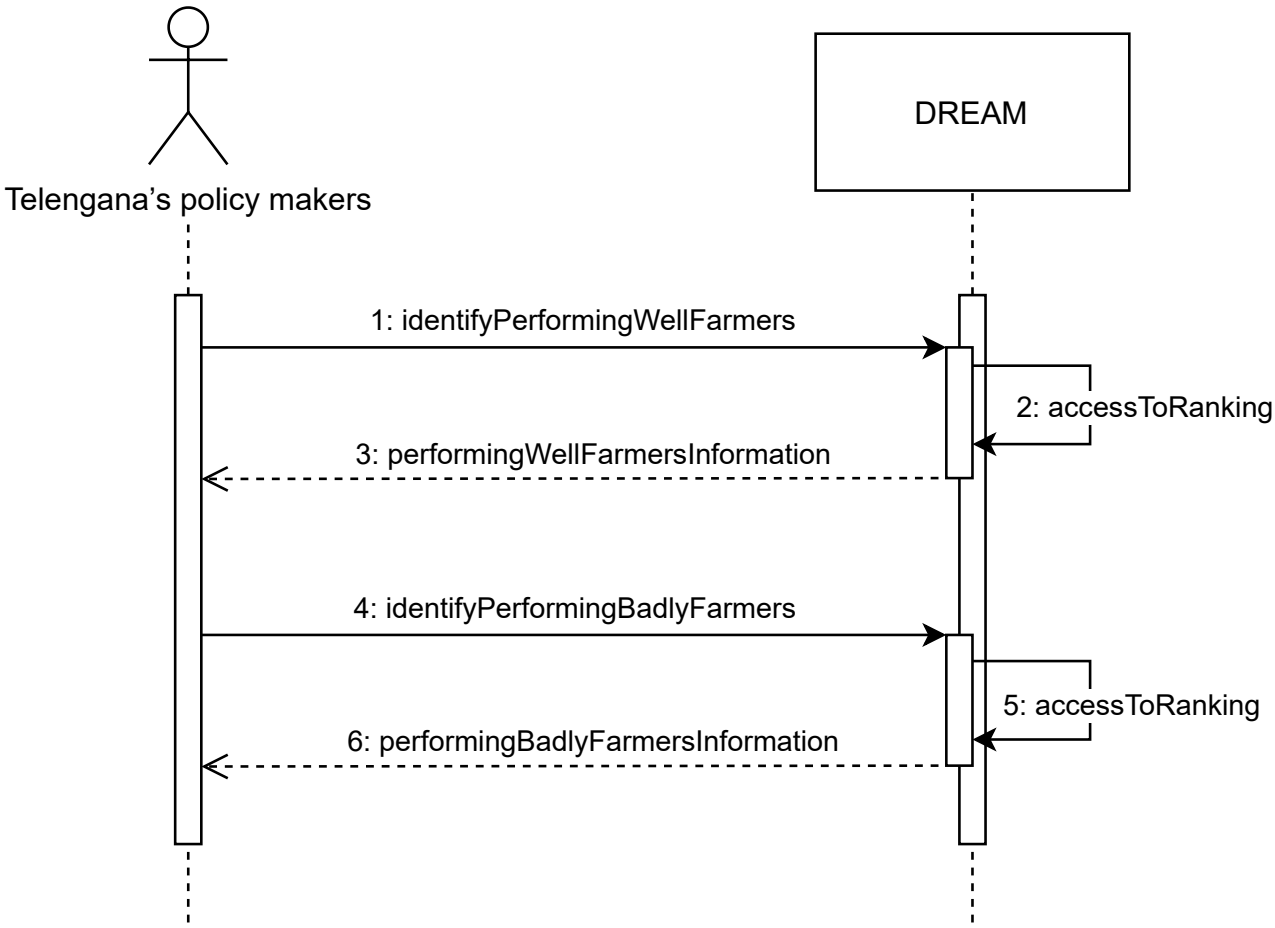


Figure 3.4: Identify Good Farmers sequence diagram

3.2.3.5 Judgement result

Use Case	Judgement result
Actor	Policy makers
Entry condition	Policy maker wants to know the production of farmers who was helped by other farmers or agronomist.
Flow of events	• Policy maker wants to know the production of farmers who was helped by other farmers or agronomist.
Exit condition	The policy maker got information.
Exceptions	

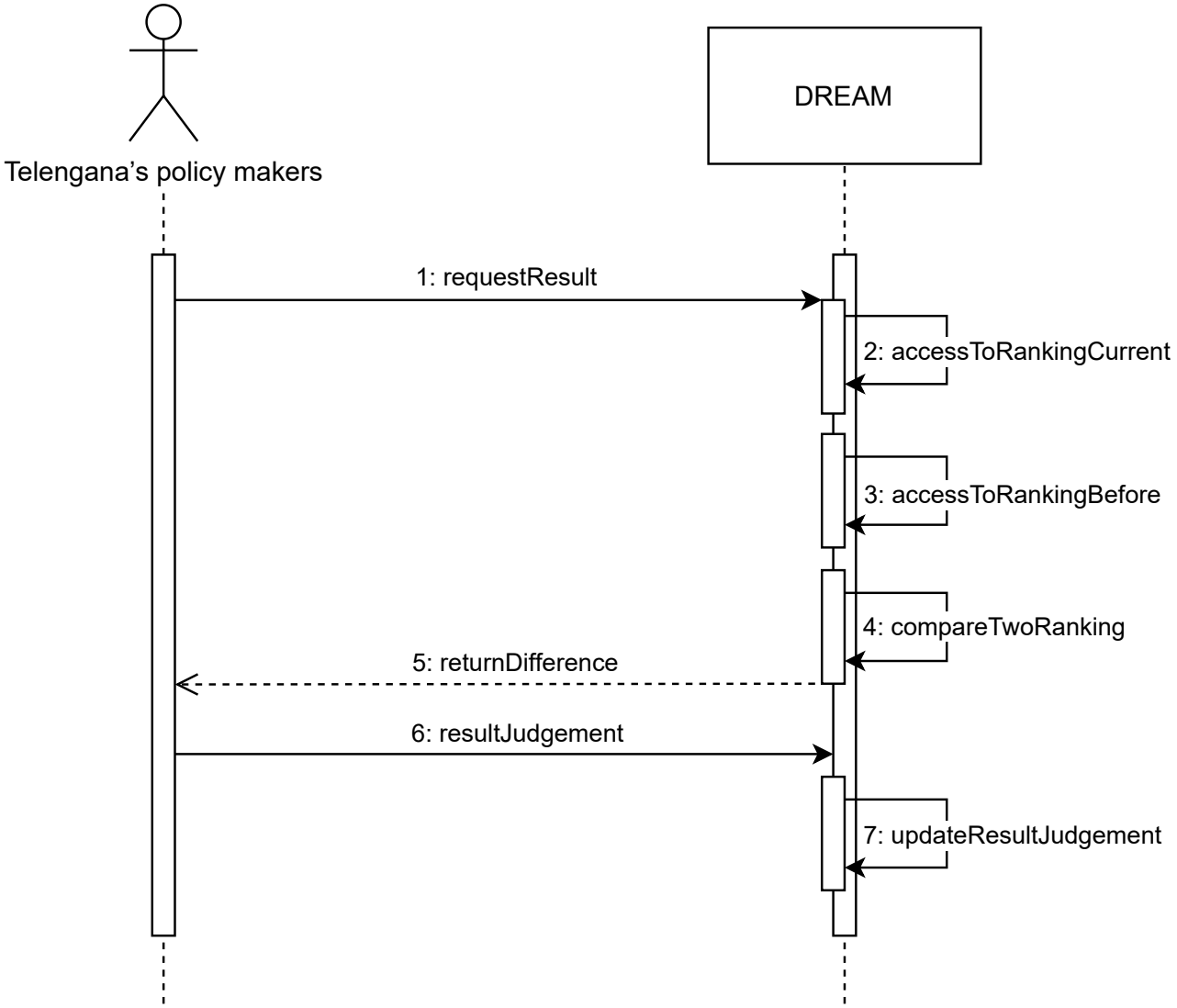


Figure 3.5: Judgement result sequence diagram

3.2.3.6 Request for suggestions

Use Case	Request for suggestions
Actor	Farmers
Entry condition	The farmer needs to be helped at DREAM.
Flow of events	• The farmer needs to be helped on DREAM and he/she write problems information on DREAM. • The system shows available suggestion way. • Farmer request suggestions. • The system returns suggestions.
Exit condition	Farmer complete request-suggestions processing.
Exceptions	

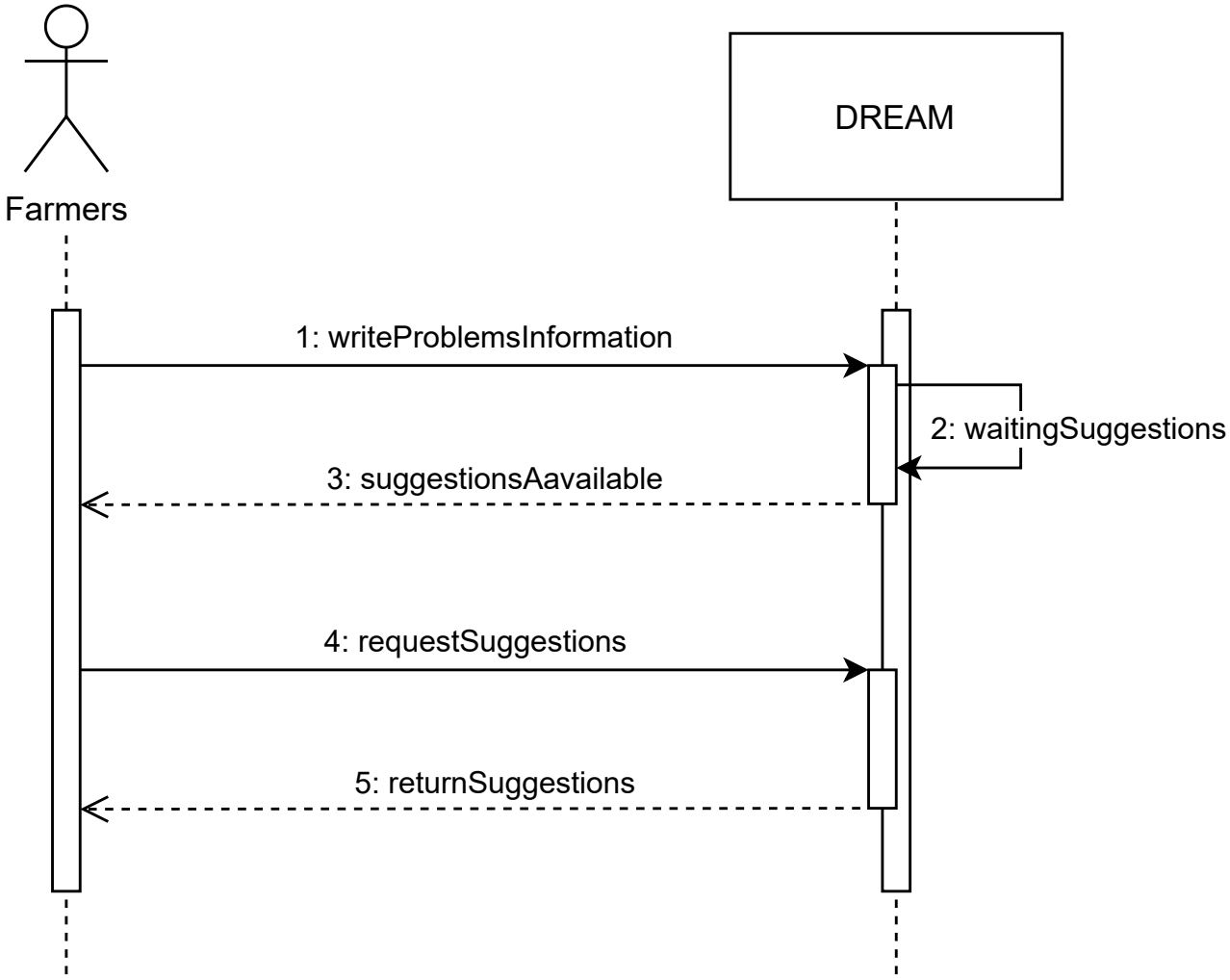


Figure 3.6: Request for suggestions sequence diagram

3.2.3.7 Create discussion

Use Case	Create discussion
Actor	Farmers
Entry condition	Farmer wants to discuss something with other farmers shopping.
Flow of events	• Farmer starts create discussion forums • select Participants And Topic
Exit condition	Farmers finished create discuss.
Exceptions	

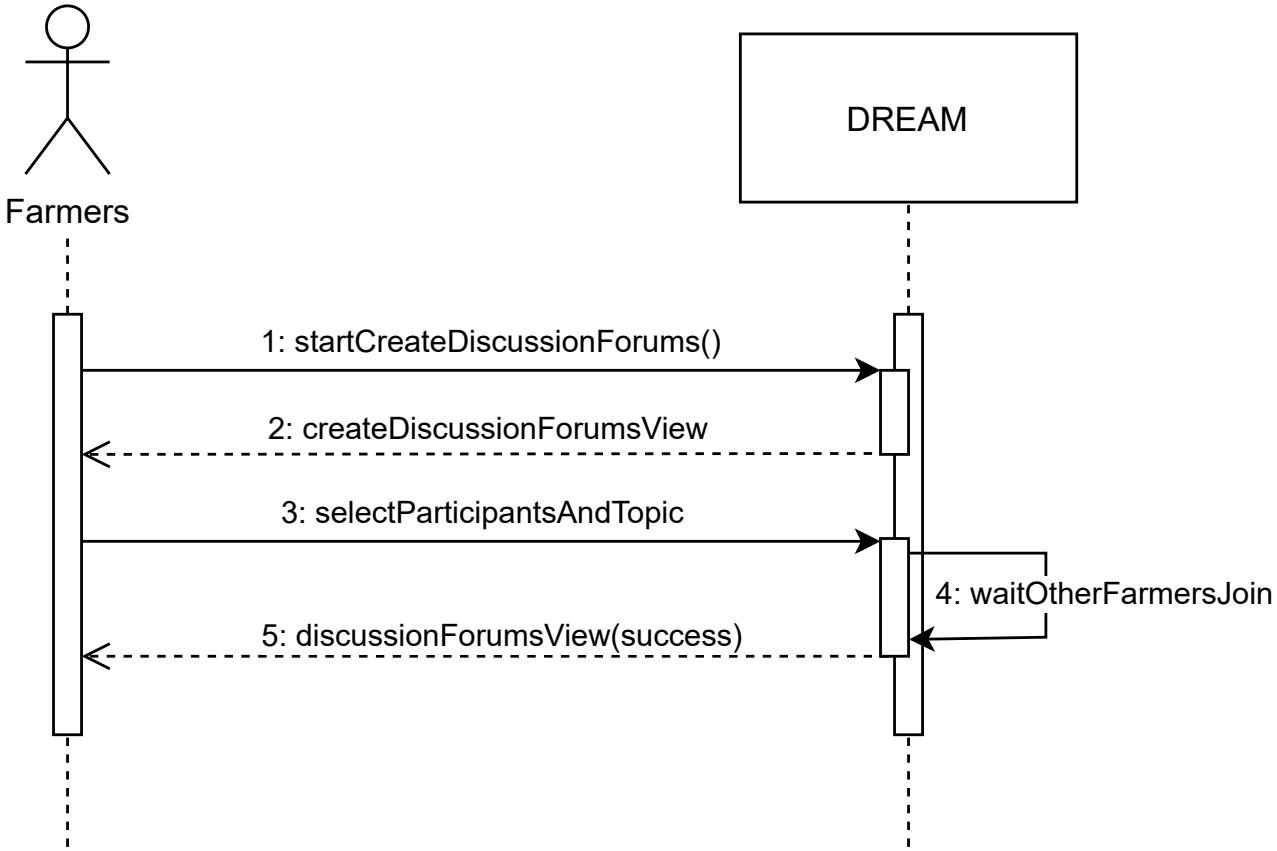


Figure 3.7: Create discussion sequence diagram

3.2.3.8 Access to data

Use Case	Access to data
Actor	Farmers
Entry condition	A customer asks a store assistant for a line up receipt
Flow of events	• Farmers have access to weather data. • Farmers need to acquire weather data. • The system return weather data. • Farmers have access to suggestions(location, production Type) • Farmers acquire Personalized Suggestions • The system return suggestions.
Exit condition	Farmers got suggestions.
Exceptions	

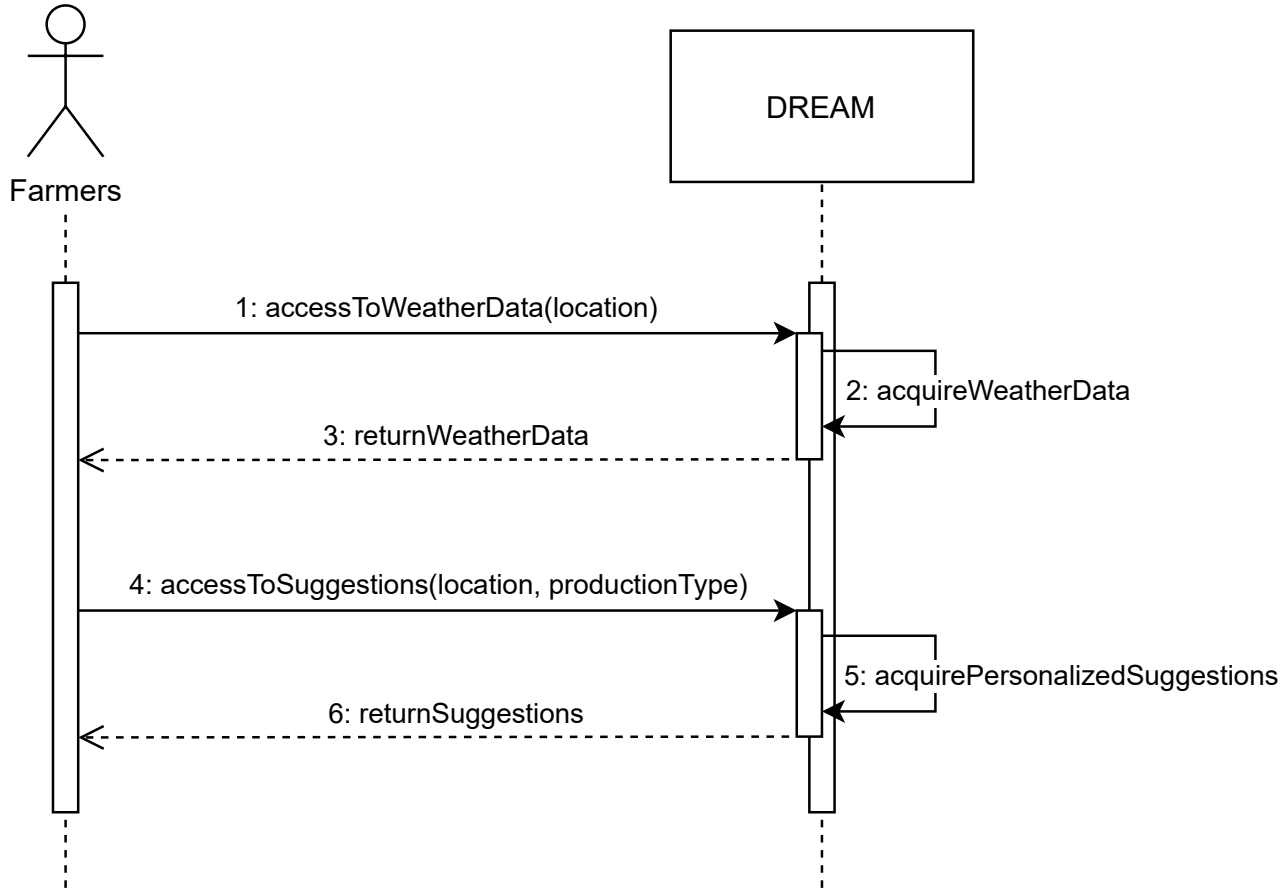


Figure 3.8: Access to data sequence diagram

3.2.3.9 Daily plan

Use Case	Daily plane
Actor	Agronomists
Entry condition	Agronomist needs to manage his/her system.
Flow of events	• Agronomists request Daily Plan. • The system will return visit list. • Agronomists have access To Ranking. • Agronomists can confirm The Execution. • The system shows daily plan view. • Agronomists can specify his/her deviations.
Exit condition	Agronomist finish his/her tasks
Exceptions	

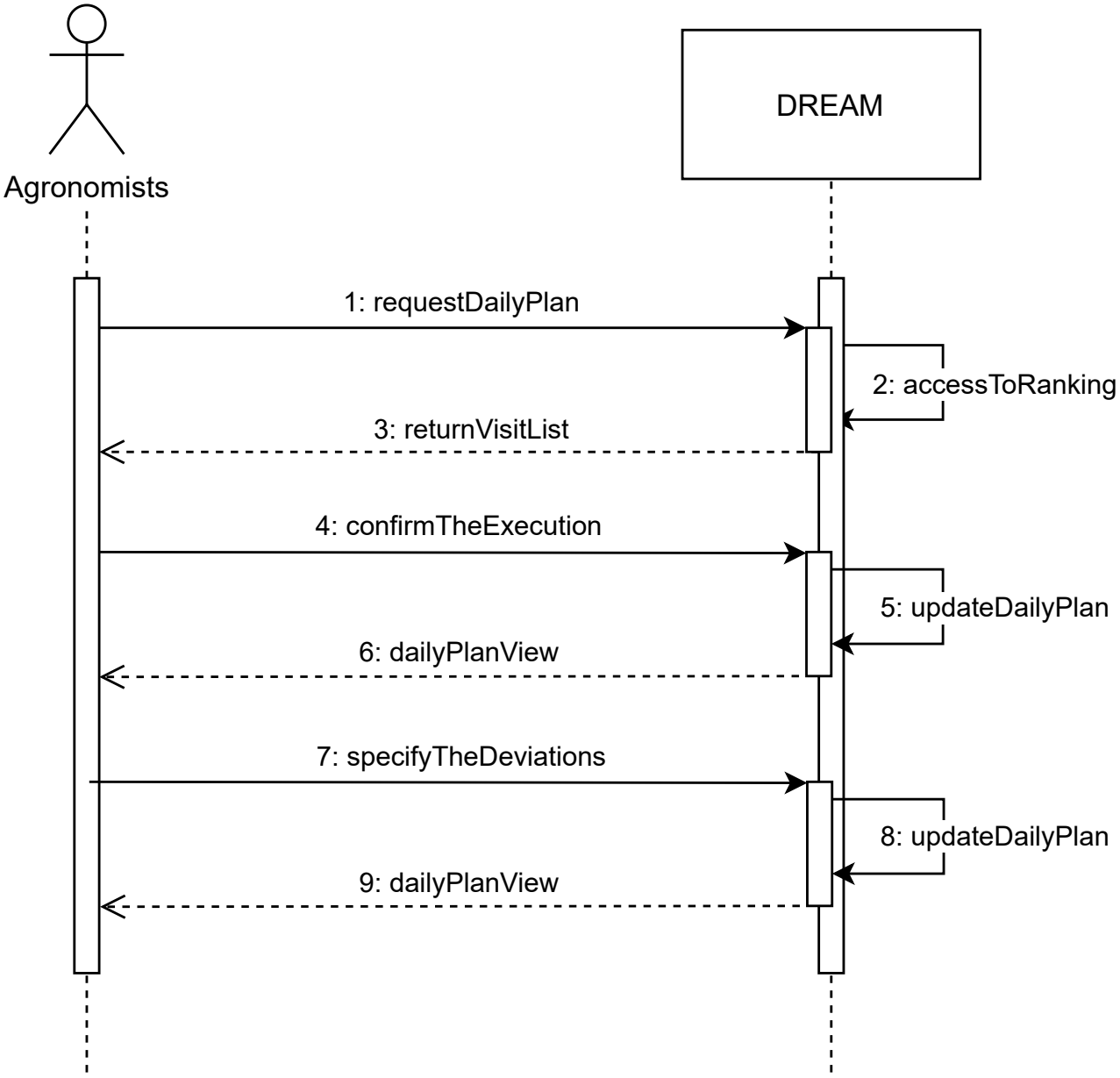


Figure 3.9: Daily plan sequence diagram

3.3 Performance requirements

- The system should be available 99% of the time.
- The system should has a backup memory to store user information and prevent data corruption.
- Data transmission delay of less than 10s for sensors.

3.4 Design constraints

3.4.1 Standards compliance

As per privacy policies, DREAM's user data must be treated respecting the law, depending on the country (i.e., the system should be GDPR compliant for all European countries).

All hardware interfaces present in field must comply the CE safety standard, in order to guarantee the maximum possible degree of safety and environmental protection.

3.4.2 Hardware limitations

Following are listed all the hardware requirements in order to use DREAM with all its functionalities:

Smartphone

- GPS antenna
- Wi-Fi or mobile 3G/4G/5G connection

Personal Computer

- Wi-Fi or Ethernet connection

Humidity Detector

- Wi-Fi, Ethernet or mobile 3G/4G/5G connection

3.5 Software system attributes

3.5.1 Reliability

The system should be nice and robust, and its structure should take into account the possible damage of system components (temperature and humidity sensors, etc.) by providing readily replaceable accessories.

The implementation of the storage system should also take into account backups in order to recover from eventual data loss.

3.5.2 Availability

Farmers may initiate requests for help and discussions at any time, so the system must have a fairly high level of reliability. Sensor-application interaction requires transmission over the network, requiring transmission with appreciable stability.

3.5.3 Security

All data inbound to DREAM services must be treated as stated by GDPR regulations.

All personnel (i.e., Policy makers, Farmers and agronomists) will authenticate to the system using Single Sign On (SSO) integrations with the stores identity systems, using either OAuth or SAML. This will allow the system to delegate the user management to external identity systems, with the benefits of:

- not having to deal with sensitive information (e.g., passwords) .
- inheriting the change of permissions of the users.
- providing the users with a unified and smooth log in experience.

Additionally, to guarantee the protection of the customer's data in between servers and the user's device, all Internet traffic must be encrypted with a modern version of TLS, and sent via HTTPS protocol.

Finally, to guarantee data protection in the backend side, data at rest encryption needs to be used: this way the system should be protected from data breaches and thefts.

3.5.4 Maintainability

Maintainability should be an issue from the very beginning of the development process. The system should strive to make the process of adapting, improving, and adding as simple as possible, and to allow easy migration to different environments. A basic rule for ensuring maintainability is to avoid using any form of deprecated software or hardware in order to keep the system scalable. In order to meet these requirements, additional appropriate design patterns must be used, but their description is beyond the purpose of this document, so they will be described in a separate document.

3.5.5 Portability

Portability means creating a program executable that can be used in different environments without major retooling of the developed software. Since DREAM can be used both as a web application and as a mobile application, it is important to separate the logic and the interface of the software from the beginning of development to ensure portability. In order to meet this requirement, a suitable development environment must be selected. A framework to ensure portability is outlined for each part of the system.

4. Formal analysis

4.1 Overview

In this section, the entities defined in the class diagram and their relationships are formally analyzed using the Alloy declarative specification language. Each relevant entity is included in the Alloy model and its relationships are tested for consistency. The focus of the model is to describe how Telengana's policy makers and Agronomists help farmers with production improvement. Different worlds were generated to highlight specific aspects of the model.

4.2 Alloy model

4.2.1 Source code

```
open util/integer

/********************** PRIMITIVE TYPES ************************/

sig Username, Password, id, personalinformation{}

/************************ SIGNATURES ***************************/

abstract sig User {
        ID: id,
        username: Username,
        password: Password,
        PersonalInformation: personalinformation
}

sig PolicyMaker extends User {
        publish: some Policy
}
sig Farmer extends User {
        create: some DiscussionForum,
        join: some DiscussionForum,
        suggest: some Suggestion,
}
sig Agronomist extends User {
        responsibilityArea: one Area,
        suggest: some Suggestion,
        schedule: one DailyPlan
}

sig Area {
        farms: some Farm
}

sig Farm {
        farmers: one Farmer
}

sig  DiscussionForum{
```

```
} {
}

sig Policy {
}

sig DailyPlan {
        receiver:  one Agronomist
}

sig Suggestion {
        give: some Farmer
}

/************************* FACTS ******************************/

// Usernames of registered users are unique.
fact uniqueUsernames {
        no disjoint u1, u2: User |
                u1.username = u2.username
}


// A farm is always belonging to a area
fact farmAlwaysInArea {
all f: Farm | one a: Area | f in a.farms
}
//A farmer is always belonging to a farm
fact farmerAlwaysInFarm {
all f: Farmer | one a: Farm | f in a.farmers
}
// A daily plan is always belonging to a agronomist
fact dailyPlanAlwaysInAgronomist {
all d: DailyPlan | one a: Agronomist | d in a.schedule
}

// A area is always associated with only one agronomist
fact areaAlwaysAssociated
{
all a: Area | one g: Agronomist | a in g.responsibilityArea
}

// A suggestion is always associated with only one agronomist
fact suggestionAlwaysAssociated
{
all s: Suggestion | one a: Agronomist | s in a.suggest
}

// A policy is always belonging to a policy maker
fact policyAlwaysInpolicyMaker {
all p: Policy | one m: PolicyMaker | p in m.publish
}


/************************ PREDICATES ****************************/

pred show {
        #PolicyMaker = 2
        #Farmer ≥ 10
        #Agronomist = 5
        #Suggestion = 10
        #Farm ≥ 10
        #Policy = 1
        #Area = 6
}
/************************* WORLDS *****************************/
run show for 50
```

4.2.2 Predicates execution and assertions checks

Alloy Analyzer 5.1.0 built 2019-08-14T18:53:58.297Z

Warning: Alloy4 defaults to SAT4J since it is pure Java and very reliable.
For faster performance, go to Options menu and try another solver like MiniSat.
If these native solvers fail on your computer, remember to change back to SAT4J.

Executing "Run show for 50"
Solver=sat4j Bitwidth=4 MaxSeq=7 SkolemDepth=1 Symmetry=20
0 vars. 0 primary vars. 0 clauses. 875ms.
No instance found. Predicate may be inconsistent. 1ms.

Metamodel successfully generated.

Figure 4.1: Results of the execution of the Alloy predicates and assertions checks

4.2.3 Resulting worlds

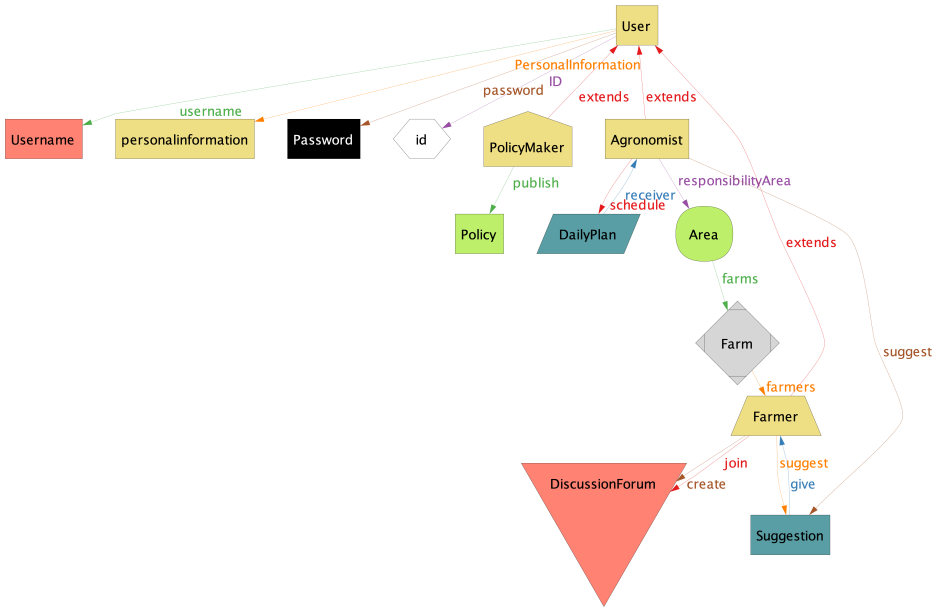


Figure 4.2: One of the worlds generated

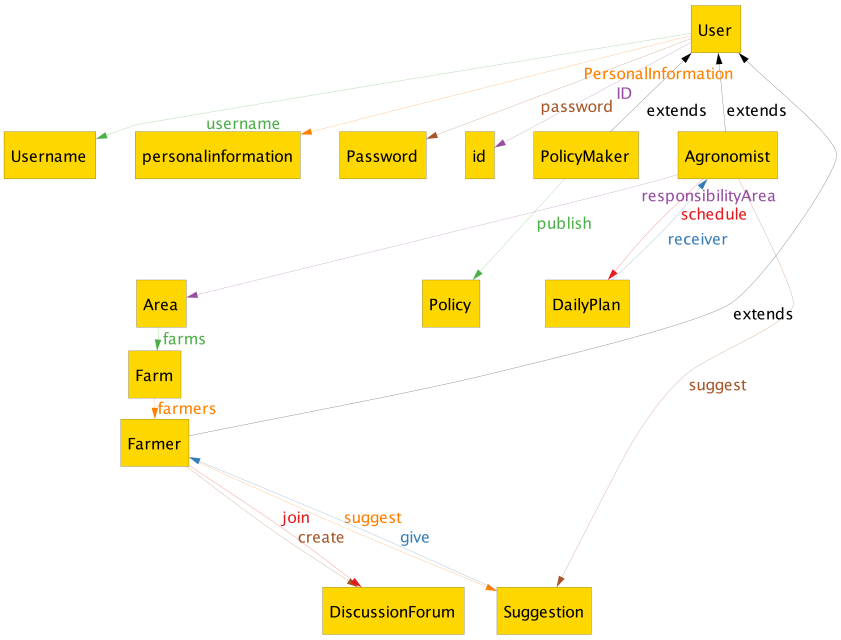


Figure 4.3: One of the worlds generated

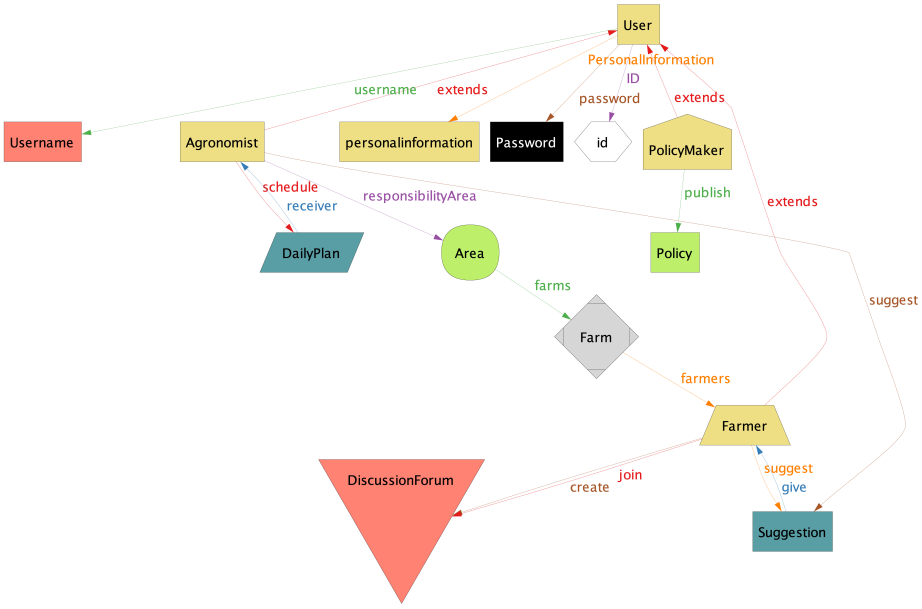


Figure 4.4: One of the worlds generated

5. Effort spent

5.1 Cui Jiayan

Date	Effort spent (h)	Notes
10/10/2021	1.0	Project set-up
17/10/2021	2.0	Introduction briefing
09/11/2021	3.0	Scope
14/11/2021	2.0	Harmonization and task appointing meeting
14/11/2021	4.0	Initial version of class diagrams
18/11/2021	0.5	Reviews
21/11/2021	2.0	Harmonization and task appointing meeting
22/11/2021	1.5	Fixes and general improvements
29/11/2021	4.0	Alloy model
29/11/2021	0.5	Improvements on constraints
04/12/2021	1.0	Reviews
05/12/2021	2.0	Alignment meeting
05/12/2021	4.0	Alloy model improvement
07/12/2021	1.0	Meeting
07/12/2021	2.0	Document refactoring and general improvements
19/12/2021	2.0	Document refactoring and general improvements
23/12/2021	0.5	Document refactoring and general improvements

5.2 Wang Yudong

Date	Effort spent (h)	Notes
10/10/2021	1.0	Project set-up
17/10/2021	2.0	Introduction briefing
17/10/2021	1.0	Purpose description
10/11/2021	1.5	Scope review
11/11/2021	2.0	Purpose description
14/11/2021	2.0	Harmonization and task appointing meeting
16/11/2021	1.5	Introduction complete and store state machine
18/11/2021	0.5	Domain assumptions review
21/11/2021	2.0	Harmonization and task appointing meeting
22/11/2021	1.0	Product functions definition
22/11/2021	0.5	PR review and insertion of images in product perspective
23/11/2021	0.5	Insertion of images in external interfaces
27/11/2021	0.5	Requirements correction and general enhancement
30/11/2021	3.0	Use cases creation
01/12/2021	1.0	Performance requirements identification
04/12/2021	5.0	Sequence diagrams creation and harmonization
05/12/2021	2.0	Harmonization and task appointing meeting
05/12/2021	1.0	Use case to requirements and goal to requirements mapping
07/12/2021	2.0	Document refactoring and general improvements
19/12/2021	2.0	Document refactoring and general improvements
23/12/2021	0.5	Document refactoring and general improvements