AY 2021/2022

Politecnico di Milano

RASD: Requirement Analysis and Specification Document

Ottavia Belotti Alessio Braccini Riccardo Izzo

Professor
Elisabetta Di Nitto

Version 1.0
November 1, 2021

Contents

1 Introduction

Data-dRiven PrEdictive FArMing, also known as *DREAM*, is a project presented by UNDP India and Healthsites initiative, promoted by Telangana's government. The aim of the project is to enhance the farm system and the entire food supply chain with an IT supporting application. This arises from modern challenges like climate change and the foreseen population growth that have underlined the critical issues of the modern system making necessary a complete overhaul.

1.1 Purpose

DREAM aims to support work categories involved into the farming industry by providing them relevant and up-to-date data about the farm activity's performance. The main stakeholders are: Telangana's policy makers, farmers and agronomists. The goal is to develop a data-driven application with the help of IT partners. Telangana's state already collect important data concerning wheather forecast, these data are publicly available with a live rainfall map on the official government website. Other data can be collected through humidity sensors deployed all over the territory and through the water irrigation system.

Agriculture has a main role in India's economy, more than half of the population depends on it and about a fifth is below the poverty line. Furthermore, as a significant increment in population is expected for 2050 (*UN*'s esteem), food demand is going to significantly increase. Telangana needs an efficient application to increase the general productivity of the farm system.

The user base is expected to be the entire population of Telangana, starting with those who works in the agricolture sector up to normal citizens.

1.2 Scope

Phenomena controlled by the Machine

ID	Phenomenom	Shared
M1	Check username and password	No
M2	Analysis of best practices	No
M3	Analysis of weather data	No
M4	Visualize data concerning weather, land, performance	Yes

Phenomena controlled by the World

ID	Phenomenom	Shared
W1	User login	Yes
W2	User share best practice	Yes
W3	User ask for help on forum	Yes
W4	Collect land data from sensor	Yes
W5	User create topic in forum	Yes
W6	User insert post	Yes
W7	User reply to a post	Yes
W8	User update daily plan	Yes
W9	User check weather forcast	Yes

1.2.1 Goals

Telangana's policy makers

1. **Identification of well-performing farmers**
   Main goal of the policy makers is to identify farmers that are resilient to meteorological adverse events. This can be done comparing the productivity ratio defined as the produced amount per product in adverse condition over the amount in standard conditions. This farmers will receive special incentives and will be asked to help other farmers with useful practices.

2. **Identification of bad-performing farmers**
   Identify farmers that are performing bad using the productivity ratio, they are the ones that need to be helped by the well-performing farmers.

3. **Visualize the results of steering initiatives**
   Visualize and evaluate the results produced by the steering initiatives from agronomists and good farmers.

Farmers

1. **Visualize data**
   Visualize important data like weather forecast and personalized suggestion about specific crops or fertilizers. All data are based on location and type of production.

2. **Insert data**
   Insert data about their production, report every type of problems.

3. **Request for help/suggestion**
   Farmers can request help with a text message that will be sent directly to the agronomists responsible of the area.

4. **Create discussion forums**
   Create forums to discuss with the other farmers. In this section the creator can choose the name of the forum and invite all the desirable partecipants.

Agronomists

1. **Insert area**
   Insert the area of responsibility for the agronomist.

2. **Receive request for help/suggestion**
   Here the agronomist can manage all the incoming request for help or suggestion. This can be done with a specific section where the agronomist can visualize the message and answer it.

3. **Visualize area stats**
   Visualize data about whether forecast or a list of best-performing farmers. The list of best-performing farmers is based on the productivity over a selected period of time.

4. **Visualize and update daily plan**
   The daily plan consists in a list of farms to be visited during the day. Every farm must be visited at least twice a year with particular attention to the under-performing ones that should be visited more often.

5. **Confirm the daily plan**
   Confirm the daily plan at the end of the day or update it in case of deviations.

1.3 Definitions, acronyms, abbreviations

Definitions

Acronyms

- **RASD**: Requirement Analysis and Specification Document

- **DREAM**: *Data-driven predictive farming* project
- **Telangana**: Indian state promoting the *DREAM* project

1.4 Revision history

1.5 Reference documents

- Specification document: "Assignment RDD AY 2021-2022"
- Alloy documentation: https://alloytools.org/documentation.html
- UML documentation: https://www.uml-diagrams.org/
- BPMN documentation: https://www.bpmn.org/
- Paper: "The World and the Machine" by M. Jackson and P. Zave

1.6 Document structure

- **Section 1** gives an introduction about the problem to tackle and about which functionalities will be implemented in the final product in order to solve it.
- **Section 2** contains the overall description of the whole project, presenting it in a more formal way through class diagrams which will contain the backbone blocks that will build the final application. Furthermore, there will be presented the so-called *actors* who are the ones that will use the application, the expected functionalities and the domain assumptions taken in consideration throughout the whole project, from the specification phase to the actual developing phase.
- **Section 3** delves deeply into the technical aspects of the topics presented in *Section 2*, in order to be more useful for the development, by providing a standard interfaces' system *a priori* to stick to during the project implementation. It will show functional and non-functional requirements. The former will be presented through some use-cases and scenarios as meaningful examples; while the latter will be disclosed by analysing performance, design and software system features that the project will have.

- **Section 4** presents the Alloy code briefly explainig the purpose of it in modeling the given problem.

2 Overall Description

2.1 Product perspective

DREAM is a system that offers the functionalities described in the *Product Functions (2.2)* section. It manages to use some external interfaces, for more details refers to the *Software Interfaces (3.1.3)* section. We can see below an high-level class diagram that shows the basic structure of the system.

2.1.1 Class diagram

A UML class diagram that describes the main entities in the system. Three types of actors can access the system: farmers, policy makers and agronomists. During sign up they all create a user account with basic informations such as username, password and email. At this point users are requested to insert other informations based on their type of account. For example, to complete the registration, farmers need to specify the address of their farm and agronomists must declare their area of responsability. The system provides several functionalities, one of this is the forum. When the farmers select the forum section they basically access the topics that are characterized by a list of messages where the sender is specified. The agronomists can confirm and update a daily plan. This is composed by the current date, the list of farms visited and a status flag that indicates whether the plan has already been confimed or not. Moreover they have access to the list of farmers currently under their area of responsability. Finally there are various type of data that can be visualized and analyzed by the users, which ones to display dependes on the type of account. There are four types of data:

- Sensor data
  Acquired by the sensors distributed all over the territory, characterized by a value that is the humidity of soil.

- Weather forecast data
  Meteorological short-term and long-term forecast acquired by the Telangana's government and displayed on their official website. The data to display to the user are temperature, rainfall, wind and humidity.

- Water irrigation data
  Acquired by the water irrigation system installed in the farm, the only significant value is the amount of water.

- Production data
  Provided by the farmers, this type of data is displayed as a table that map the type of product to the amount produced.

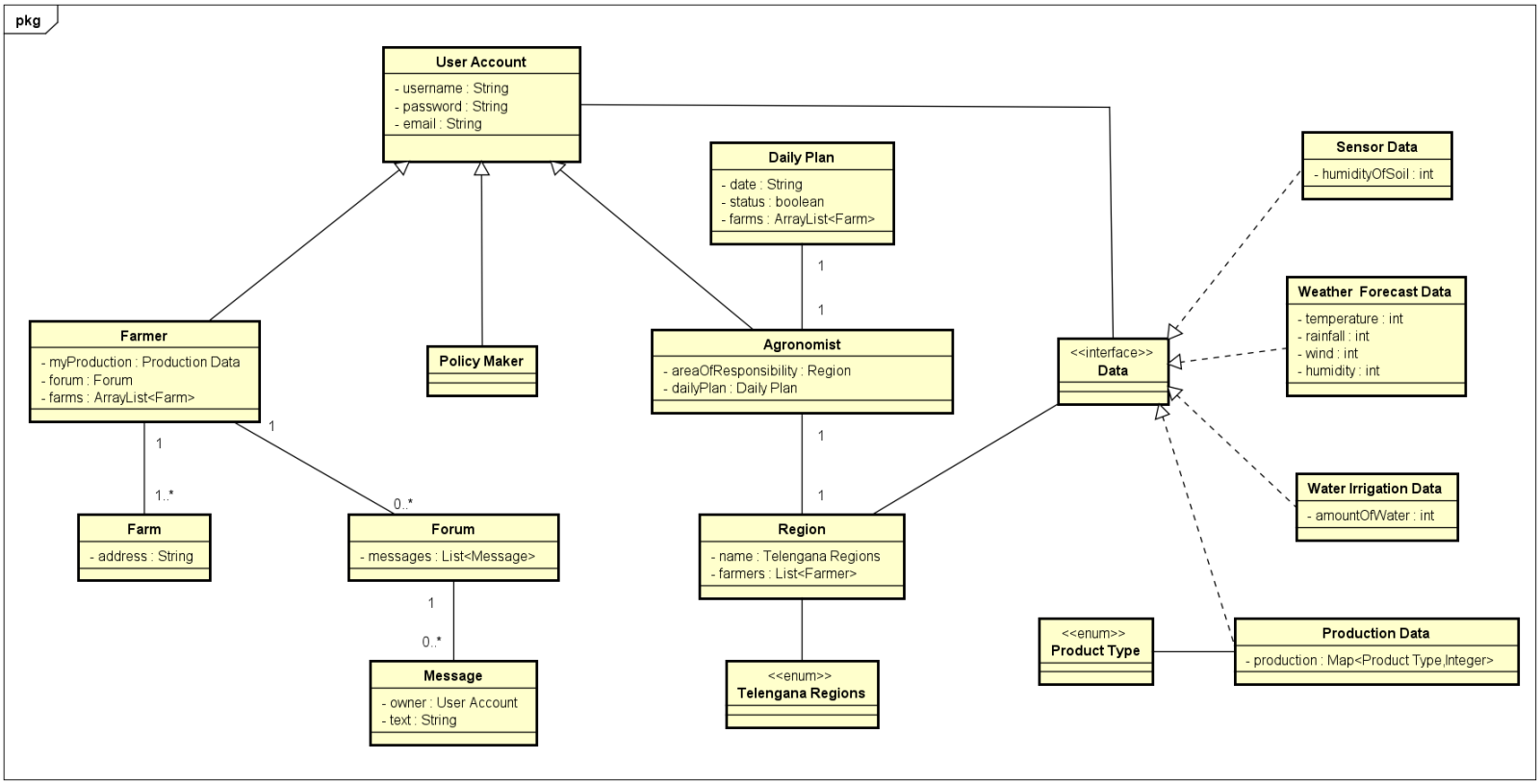

Figure 1: High-level UML

2.2 Product functions

2.2.1 Sign-up and shared functions

- **Sign-up:** let the user sign-up thorugh an email and a password, creating a profile tailored for the user's job. Specify the area in which they live and what type of cultivation they manage.

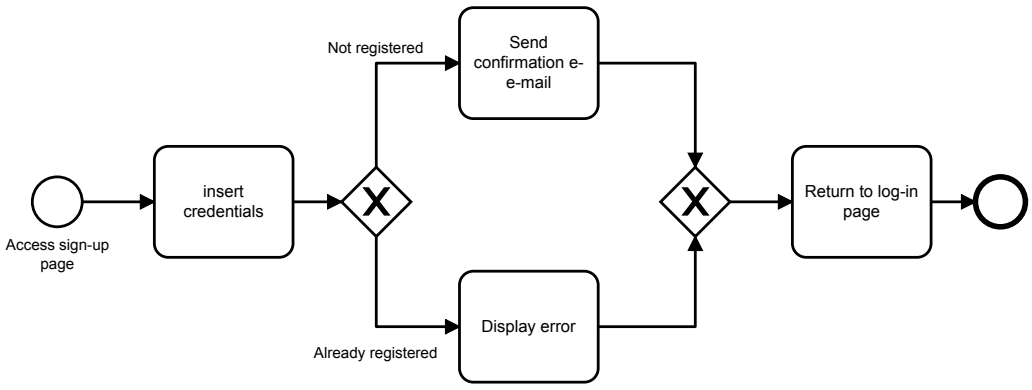

Figure 2: Sign Up BPMN

2.2.2 Policy makers functions

- **Visualize relevant data and initiative:** let the policy makers know a variety of different data like the performances of the farmers by grouping them in a rank to know who are the farmers that are performing well and who are the worst one based on informations insert by them. Policy makers can also visualize the steering iniziative presented by the agronomist in a specific subsection of their system.

2.2.3 Farmers functions

- **Profile edit:** allow the farmers change their profile in order to upgrade information like:

  Area: allow to change the area where farmers have their plantation;

Plant type: allow to change the type of plant;

Username and password: allow to change their username and password.

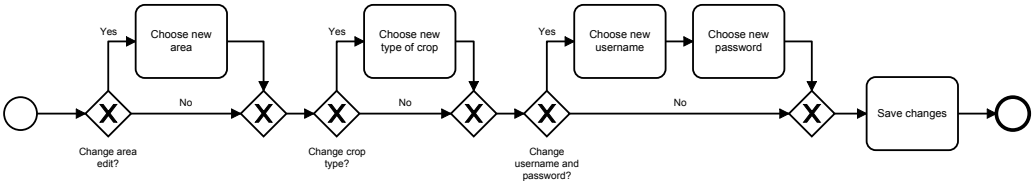

Figure 3: Profile edit BPMN

- **Manipulation of informations:** allow farmers to visualize every kind of their interest like the wheather forecast for the day or for the next few days or some suggestions about own crops and specific fertilizers.
- **Send message:** allow farmers to send messages to the agronomist.
- **Usage of the forum:** allow farmers to create a new topic or reply to a message in the dedicated forum.

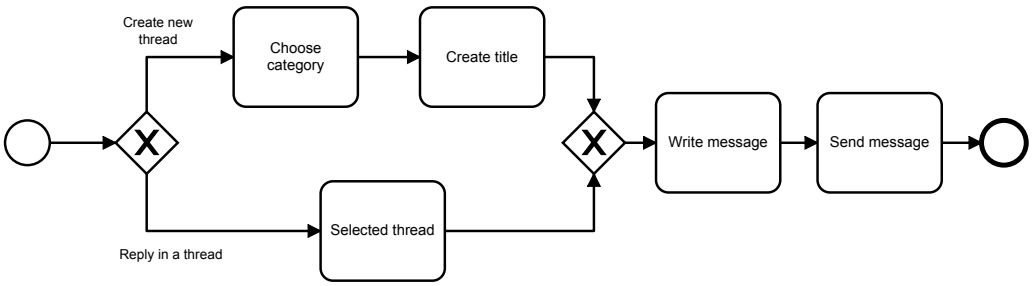

Figure 4: Forum BPMN

•

2.2.4 Agronomists functions

- **Area functions:** allow the agronomist to insert their responsability area and visualize the correspondant data like wheather forecast or the rank of the farmers.
- **Manage farmers requests:** farmers can send help or suggestion request to whom agronomist have to reply.
- **Manage daily plan:** allow the agronomist to make their daily plan by registring the incoming visits to the farmers. At the end of the day a farmers must confirm if they were able to complete the plan or specify the deviations occurred during the day.

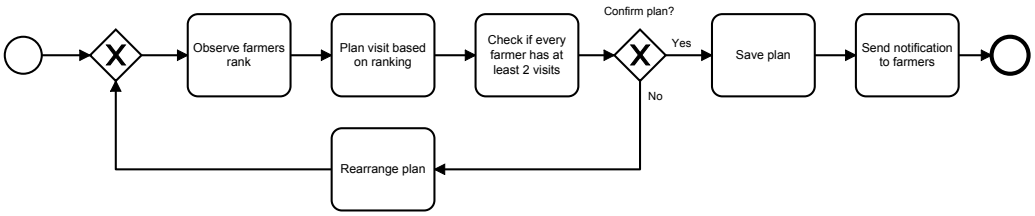

Figure 5: Daily plan BPMN

2.3 User characteristics

The application has been thought for the three different user categories that follows:

- **Policy makers** are government's employees that are in charge of analysing the general agricolture trends among all the districts in Telangana, then promote based state-wide policies to better the whole food system. Their main goal is not only to secure the current provision, but also to identify now the best practices that will lead to a flourishing food production in the future. By doing so, the plan is to grow more resiliant and profitable crops and prepare the lands to face future menaces, for instance the climate that is getting more hostile or the

foreseen increment of the food demand. As a consequence, they want to be notified about the best performing farmers in order to contact them and get more insight from them about their procedures, with the aim to acquire best practises to be shared and applied on a larger scale. At the same time, they need to know who, on the other hand, is performing particularly badly, so that they can be given the help they need to better their results, since obtaining the foresaid goal requires the structure to run smoothly in all its parts. Policy makers also need a feedback system that let them be aware of the true impact *a posteriori* of the initiatives carried out by the agronomists in collaboration with the knowledge and practice of the best farmers.

- **Farmers** are interested in functionalities that will help with their day-to-day life at work, so they would like to receive in one place all the information about the weather to plan before hand the work day and useful data, like suggestions and news about the specific crop they cultivate, if some crop's illness is spreading in their area and how to treat it, which fertilizers boost the plant's production, etc. Moreover, they should insert data about their own production and ask for help to a regional agronomist through the app if it's needed. Being part of a larger community of people that share the same purpose (such as being more productive) brings more knowledge in general, so it's easier for the farmers to get in touch with their collegues that grow the same crops and might have faced the same challenges they do through the in-app forum. The feature allows them to enlarge their pool of acquaintances and brings them together online, even though they might be kilometers away from each other.

- **Agronomists** are the experts in the agricolture field, so the main function needed for them is the possibility of helping out the farmers that reach out to them. Each agronomist is in charge of a specific geographical area in Telangana, in order to be efficiently present on the territory in a fair and useful way according to the actual helping demand. In fact, they visit each farm spread among their area at least twice a year. That said, agronomists would like some functionalities that help them planning out their trips on the field in an simple yet flexible way. Furthermore, they would like to be notified of the farms' performace, especially the ones scoring poor results in order to plan

their visits more often for those, depending on the problem their facing. Nonetheless, in order to make a complete and axhaustive report about the area productivity for the central government, they are also interested into acknowledging the top performing farms.

2.4 Assumptions, dependencies and constraints

- D1: Agronomists actually stick to the scheduled daily plan
- D2: Agronomists correct their schedule at the end of the day if deviations occur
- D3: Each user registers according to their role and always feeds correct information to the app
- D4: The internet connection works properly
- D5: The sensors measuring humidity level of the soil work properly
- D6: Weather forecasts are accurate up to a 80% for the short-term and up to 60% on the long-term
- D7: Water irrigation system send the information accurately, with an error of at most 1%
- D8: Agronomists always replay to farmers requests of help
- D9: Farmers periodically insert data about their crop status and resulting production

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The system should interface with users through devices which must be connected to the Internet. Everyone that needs to use this service would connect to it through a web broswer from an existing domain, for example: dream.com. It must be easy to use as it will have to be used from different kind of people, sometimes with a not great affinity with technologies.

3.1.2 Hardware Interfaces

The system "as it is" does not provide specific hardware equipment in order to access to web site. To use all functionalities it's required to have an IT devices connected to the internet, eventually a user can decide to use GPS to insert his own location and only proper devices with GPS equipment can do this.

Soil data, obtained from sensors, are collect by someone else and will be added by the policy makers, also this not requires any special kind of hardware.

3.1.3 Software Interfaces

In order to use the system it's required the use of a web broswer, on that will run the application.

System will also use some important interfaces in order to accomplish its functionalities.

First is an external API to access maps which will use the position acquired by the user settings to return an interactive map with a marker on that exact position.

Second API service used is the one of the DBMS system which will be adopted in order to query the internal database.

Another API will be use to access the external database containing data reguarding the sensor data and the other external data that the system need.

3.1.4 Communication Interfaces

Internet connection is mandatory in order to access to every formation that the system will display. Users that want full functionalities have to have a GPS system on their device in order to guarantee a certain level of precision in the localization phase.

3.2 Functional Requirements

3.2.1 List of Requirements

ID	Requirement

R1	The system can be used only by registered and logged-in users.
R2	If the IT device is equipped with GPS technology, then the system must suggest the users the area in which they are during the registration phase. In any case, the final decision will be up to the registering user.
R3	The system must allow the users to change their location in any moment, except when they have connectivity issues.
R4	The system must allow Policy Makers to see up-to-date statistic data, accordingly to the database, about water irrigation systems and soil humidity sensors.
R5	The system must give Policy Makers up-to-date ranking of the best and worst performing farmers.
R6	The system must use a fair and scientific score to rank the Farmers in order to represent the real situation.
R7	The system must let Farmers insert their production information every day, even without connection.
R8	If a Farmer has temporary connectivity issues while trying to upload production data, the system must keep them locally stored until connection is restablished. Once there is connectio, data must be uploaded to the database.
R9	The system must suggest a list of agronomists appointed to the area of the Farmer requesting professional help.
R10	The system must notify Agronomists of unresolved requests of help from Farmers.
R11	The system must show weather forcasts relevant to the area concerning the Farmer or the Agronomist.
R12	The system must present news concerning crop only if relevant to the Farmer's own crop type.
R13	If users are experimenting connectivity issues, the system must inform them to try later.

R14	The system must allow every user registered as Farmer to access the forum, to create new discussions and to post replies to already existing ones.
R15	The system must suggest to the Agronomists which farms to visit (among the ones in their area of competence) while planning the Daily Plan based upon the last visit day following a FIFO policy. Exceptions to the FIFO policy are the farms under-performing, which shall has higher priority.
R16	The system must allow the Agronomists to add a schedule for the day each day, even without connection.
R17	The system must allow the Agronomists to update their schedule during the day and to confirm the execution before the end of the day. If the user doesn't confirm before 23:59, the system must register the already uploaded schedule as definitive.

3.2.2 Mapping

3.2.3 Use Cases

3.2.3.1 Use Cases Diagram

3.2.3.2 Use Cases Description

- **Shared Use Cases**

Sign Up

Use Case	Sign Up
Actor	Policy Maker, Farmer, Agronomist
Entry condition	User want to register in the system
Flow of events	1. User press the sign up button 2. User select its username, password and role
Exit condition	The system shows a confirmation message to the user
Exceptions	1. If the user insert an already taken username 2. If the user press the cancel button 3. Internet connection isn't working

Log In

Use Case	Log In
Actor	Policy Maker, Farmer, Agronomist
Entry condition	User want to log in the system
Flow of events	1. User put its own username and password
Exit condition	The system correctly log in the system
Exceptions	1. If the user insert a wrong username and password 2. Internet connection isn't working

- **Policy Makers**

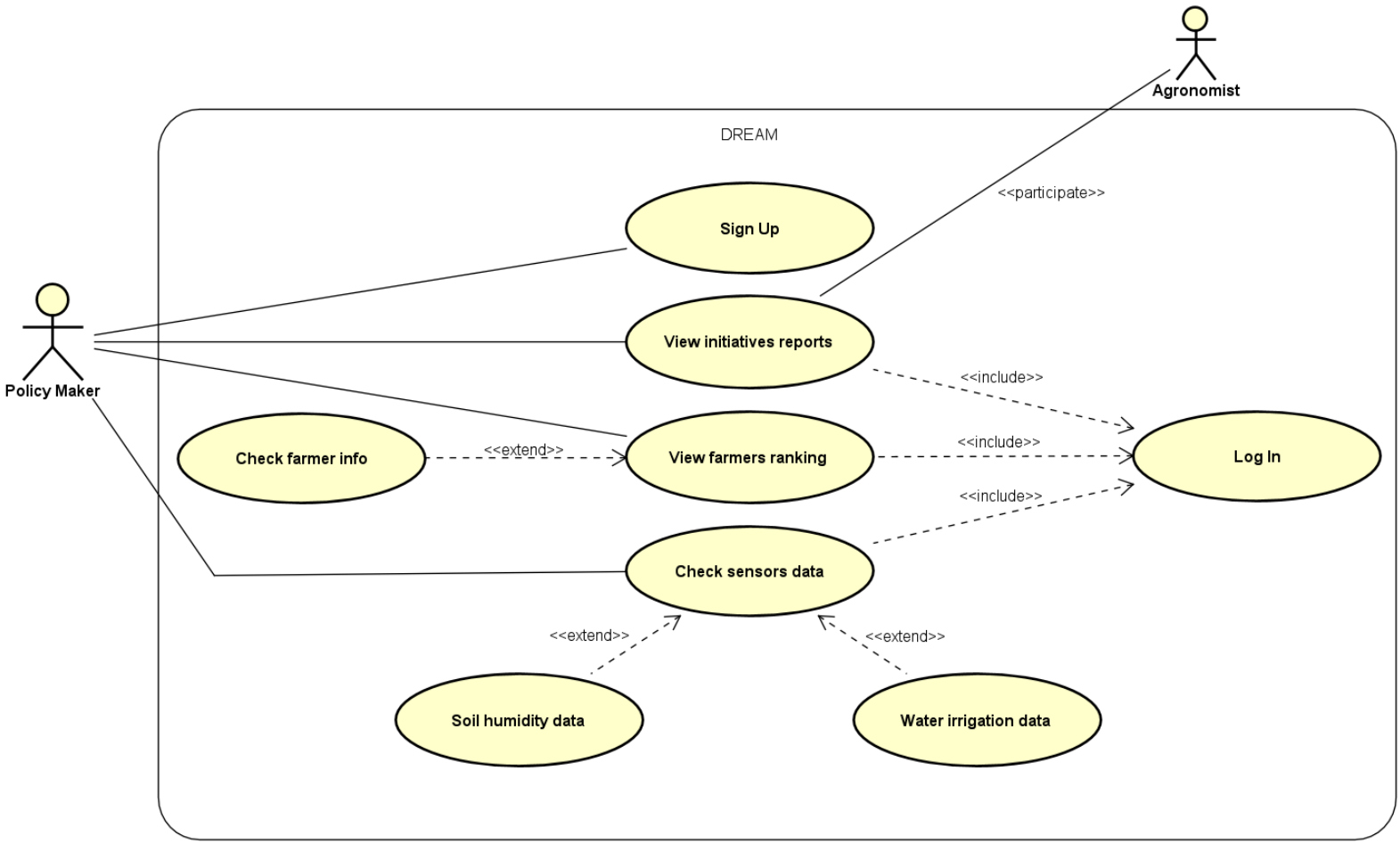

Figure 6: Policy Maker - Use Case Diagram

View Initiatives Reports

Use Case	View Initiatives Reports
Actor	Policy Maker
Entry condition	User wants to check the Steering Initiative proposed by agronomists
Flow of events	1. User press the "View Initiatives Reports" button 2. System send to the user the steering initiative insert by the agronomist
Exit condition	User press the back button
Exceptions	

Check Soil Humidity Data

Use Case	Check Soil Humidity Data
Actor	Policy Maker
Entry condition	User wants to check the soil humidity data
Flow of events	1. User press the "Check Sensors Data" button 2. User press the soil humidity data button 3. System retrive soil humidity data and send them back to the user 4. System display the data
Exit condition	User press the "Home" button
Exceptions	

Check Water Irrigation Data

Use Case	Check Water Irrigation Data
Actor	Policy Maker
Entry condition	User wants to check the water irrigation data
Flow of events	1. User press the "Check Sensors Data" button 2. User press the water irrigation button 3. System retrive water irrigation data and send them back to the user 4. System display the data
Exit condition	User press the "Home" button
Exceptions	

View Farmers Ranking

Use Case	View Farmers Ranking
Actor	Policy Maker
Entry condition	User wants to see the farmers ranking
Flow of events	1. User press the best or the worst performing ranking 2. System retrive data on farmers ranking 3. System show ranking data
Exit condition	User press the "Back" button
Exceptions	

View Specific Farmer Informations (View Farmers Ranking)

Use Case	View Specific Farmer Informations (View Farmers Ranking)
Actor	Policy Maker
Entry condition	User wants to see the specific informations of a farmer
Flow of events	1. First three events are the same of the "View Farmers Ranking" use case 2. User press on a farmer name 3. System retrive farmer data 4. System display farmer data
Exit condition	User press "Back" button
Exceptions	

- **Farmers**

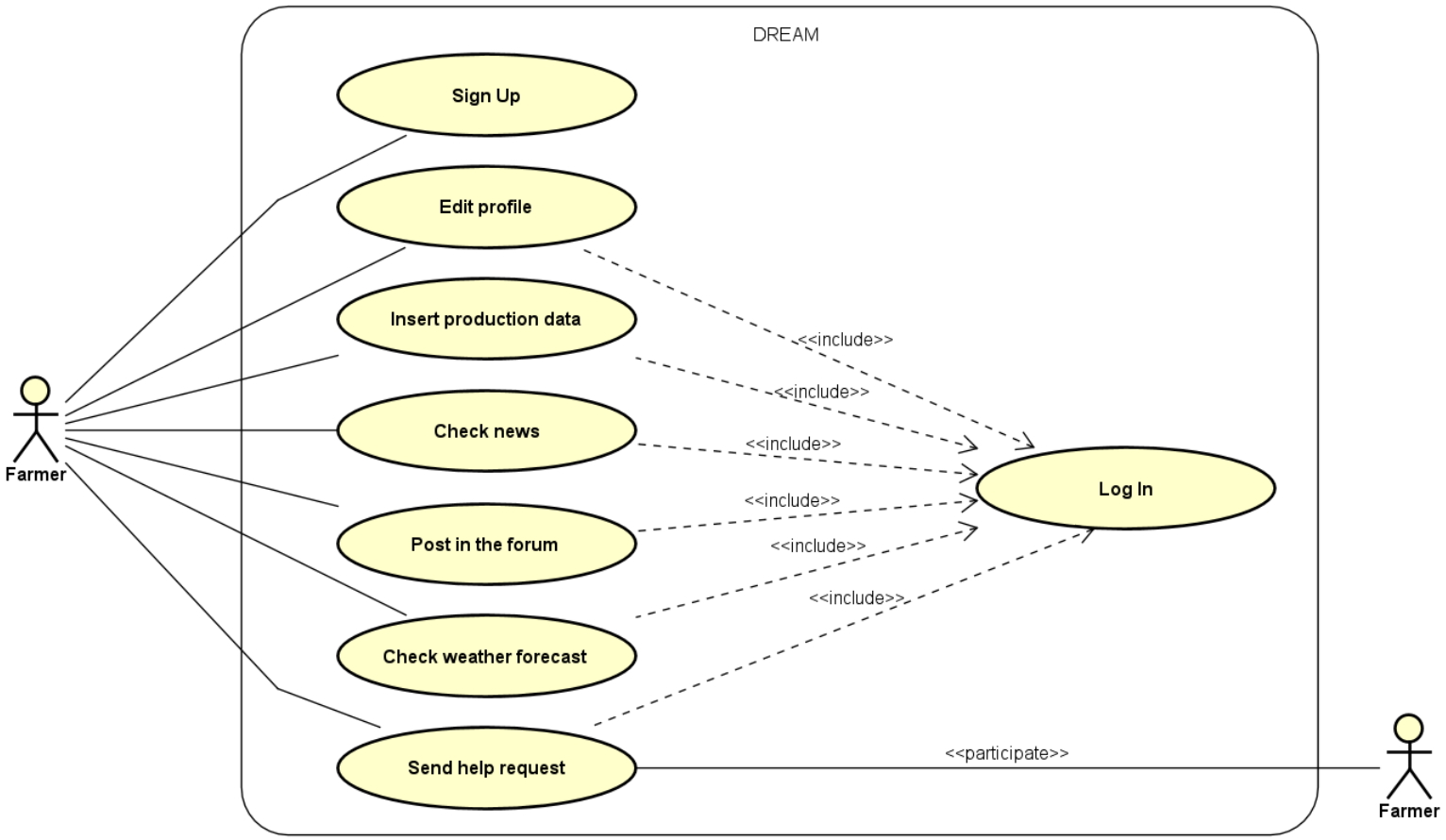

Figure 7: Farmer - Use Case Diagram

Profile Edit

Use Case	Profile Edit
Actor	Farmers
Entry condition	User wants to modify or update its profile
Flow of events	1. User press the "Settings" button on its home page 2. User modify the interest data (username, password, crop type or area) 3. User press "Confirm" button
Exit condition	User receive a confirmation message
Exceptions	1. User choose an already taken username 2. User choose the old password

Insert Production Data

Use Case	Insert Production Data
Actor	Farmers
Entry condition	User wants to insert the production data
Flow of events	1. User press the "Insert Production Data" button 2. User insert data reguarding its coultivation performance 3. User press "Confirm" button 4. System collect and analiyze data
Exit condition	User receive a confirmation message
Exceptions	1. User insert data in a wrong manner 2. User press "Back" button

Check News

Use Case	Check News
Actor	Farmers
Entry condition	User want to check the news
Flow of events	1. User press the news box 2. The system retrive data reguarding news
Exit condition	User press "Back" button
Exceptions	

Post in the Forum

Use Case	Post in the Forum
Actor	Farmers
Entry condition	User wants to post in the forum
Flow of events	1. User press "Forum" button 2. User choose a thread to reply in or the creation of new thread 3. User write the text
Exit condition	User press "Post" button
Exceptions	1. User exceed the maximum number of character

Check Weather Forecast

Use Case	Check Weather Forecast
Actor	Farmers
Entry condition	User wants to check weather forecast
Flow of events	1. User press on the weather forecast widget
Exit condition	User press "Home" button
Exceptions	

Send an Help Request

Use Case	Send an Help Request
Actor	Farmers
Entry condition	User needs the agronomist help
Flow of events	1. User press "Contact Agronomist" button 2. User choose the object of its help request 3. User write the message 4. User send the request by pressing "Send request" button
Exit condition	The system shows a confirmation message to the user
Exceptions	1. User doesn't fill every mandatory boxes

- **Agronomists**

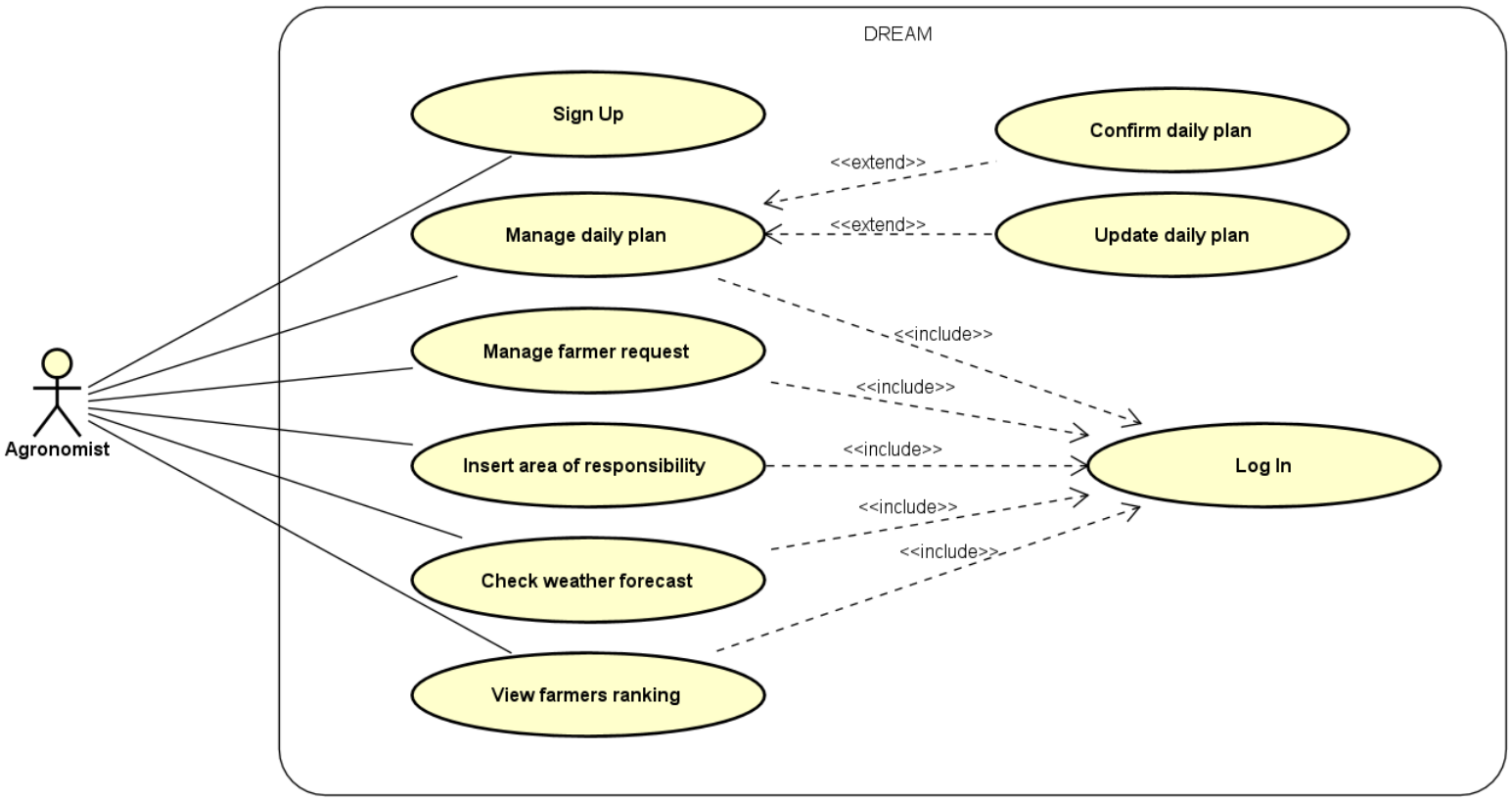

Figure 8: Agronomist - Use Case Diagram

3.2.4 Scenarios

1. Sahil is a farmer that would like to share some best-practices about a particular product that he has cultivated during the last year. He decides to open a new discussion on the forum and invites all his colleagues.

    - He opens the app on his device, if he is already registered he logs in
    - The system shows the homepage for a farmer user
    - He clicks on the "Forum" and access the forum section

- Now he clicks on the "Create new topic" where he can select the name of the discussion
- Finally he writes a new message in the discussion

2. Pajeet is one of the main policy makers in the state of Telangana. He wants to distribute special incentives to the best farmers and to do this he decides to open the app and visualize the data.

    - He opens the app on his device, if he is already registered he logs in
    - The system shows the homepage for a policy maker user
    - He visualize the ranking of the well-performing farmers
    - He clicks on the name of a farmer
    - Now the system shows the profile of the selected farmer
    - Here the policy maker can find useful contact informations such as phone number and email
    - He takes note of the desirable contact informations and close the app

3. Shaleena is an agronomist responsible of Mahbubnagar, one of the biggest region in Telangana. She wants to compose the daily plan to decides which farms to visit today and for this reason she decides to open the app.

    - She opens the app on his device, if she is already registered she logs in
    - The system shows the homepage for an agronomist user
    - She clicks on the "Daily plan" and access the new section
    - She can add the farms she wants to visit to the daily plan by choosing them from the list of available ones
    - Now the system shows the addresses and the date of the last visit of the selected farms
    - She takes note of the addresses of the farms to visit and close the app

4. Chaitanya is an agronomist that after a hard day of work wants to update the daily plan on the app. At the end of the day she realizes that she has visited three farms instead of the two indicated in the daily plan.

    - She opens the app on his device, if she is already registered she logs in
    - The system shows the homepage for an agronomist user
    - She clicks on the "Daily plan" and access the new section
    - Here she can visualize the daily plan
    - She can click on the "Add" or on the "Remove"
    - In case of "Add" the system would redirect her to the list of available ones, in case of "Remove" she can delete a farm from the list simply by clicking on the cross at the right
    - She clicks on the "Confirm" to validate the daily plan
    - The system returns to the homepage
    - Finally she close the app

5. Damayanti is a farmer that would like to improve the productivity of his farm. To do so he decides to consult the section regarding the suggestions on specific crops to plant or specific fertilizers to use.

    - He opens the app on his device, if he is already registered he logs in
    - The system shows the homepage for a farmer user
    - He clicks on the "News" and access the new section
    - Here he visualizes suggestions based on his type of production and on his location
    - If he is not satisfied he clicks on the "Request for suggestions" to aks agronomists or other farmers
    - Otherwise he takes note of the suggestions
    - He closes the app

6. Pradeep is an agronomist who wants to take care about help requests before going to work.

- He opens the app on his device, if he is already registered he logs in
- The system shows the homepage for an agronomist user
- He clicks on the "Mail box" and access the new section
- Here he visualizes a list of help requests
- By clicking on a help request he can visualize the request and answer
- He do this for every help request until they are finished
- Finally he closes the app and begins his working day

3.3 Performance Requirements

The majority of the user base will be represented by the farmers that cover the 58% of the entire population of Telengana. According to Unique Identification Aadhar India the Telangana population in 2021 is estimated to be 39.9 million. This means that the system is expected to be widely adopted with the registration of millions of users just in the first year. Also the average workload of the system is expected to be high, the goal is to guarantee the simultaneous connection of 200.000 individuals. A mid-term goal could be to improve the scalability of the system in order to guarantee an efficient service to all users. Finally the system should have a good response time, less than 3 seconds is reasonable.

3.4 Design Constraints

3.4.1 Standards compliance

DREAM's user data must be treated respecting the law according to the PDPA. All hardware devices distributed all over the territory (sensors) must comply the CE safety standards.

3.4.2 Hardware limitations

The software is designed to enhance the portability, in fact it is enough to have a device with a stable internet connection. In order to use the application correctly it is required to have a device equipped with GPS.

3.5 Software System Attributes

3.5.1 Reliability

To ensure data consistency the system must have a fully backup infrastructure, this allows to recover from general failures in the main system.

3.5.2 Availability

The system should be up almost 24/7 in order to allow users to use the application's services at any time. The peak of use is expected during the working hours, it means that it is better to do all the maintainance activities during the night hours.

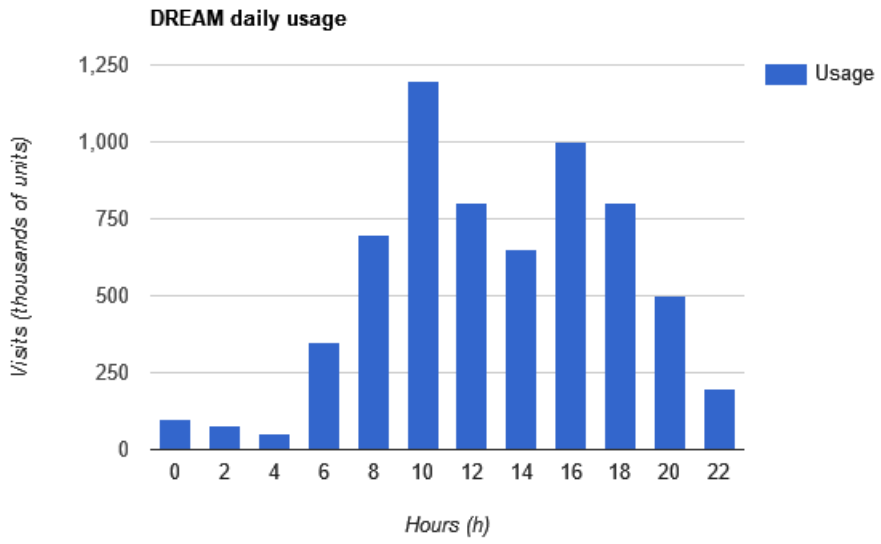

Figure 9: DREAM daily usage

3.5.3 Security

All the data stored by the system must be encrypted with a standard algorithm like the AES (Advanced Encryption Standard). The communications goes on a secure channel powered by the SSL protocol. In addition users can request a two-factor authentication (2FA) to improve the security of the personal account. All data should be stored in compliance with PDPA's regulation.

3.5.4 Maintainability

The maintainability is guaranteed by high standard of code, the use of design patterns will enhance this point and guarantee high reusability. The code must be extensively tested. The system must be extendible in order to support future updates. User's application will be regularly updated.

3.5.5 Portability

The system is designed to work on different operating systems like Windows, MacOS and Linux. All main browsers are supported.

4 Formal Analysis using Alloy

5 Effort Spent

Student	Time for S.1	S.2	S.3	S.4
Ottavia Belotti	1h	2h	?	?
Alessio Braccini	2h	2h	2h	?
Riccardo Izzo	2h	2h	2h	?

6 References