

**Jordan University of Science and Technology**

**College of Computer Sciences & Information Technology**

**“Organizing Agricultural Production”**

*A project submitted*

*in partial fulfillment of the requirements for the degree of*

*Bachelor in Software Engineering*

**by**

Ibrahim Raid Ibrahim Hajras (145978)

Hesham Ahmed Mohammad alqaoud(146496)

**Supervised by**

Dr. Zakaria AL SHARA

**Committee Member Names**

Dr. Mohammad Radaideh

Dr. Mohammad Malkawi

**June 2023**

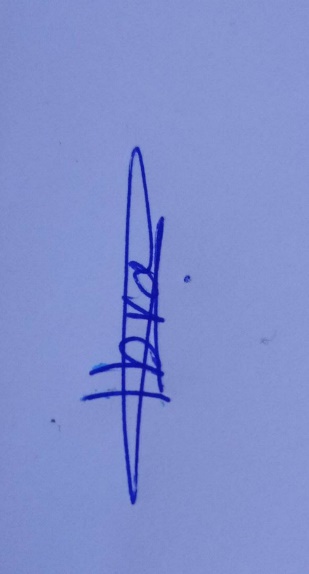
## Undertaking

This is to declare that the project entitled “**Organizing Agricultural Production**” is an original work done by undersigned, in partial fulfillment of the requirements for the degree “bachelor’s in software engineering” at Software Engineering Department, College of Computer and Information Technology, Jordan University of Science and Technology.

All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

Student 1 : Ibrahim raid Ibrahim Hajras

Student 2 : Hesham Ahmed Mohammad Alqaoud

A close-up of a drawing

Description automatically generated with low confidence

## ABSTRACT

The application aims for a way of communication between farmers and to organizing agricultural operations in all regions of the Kingdom of Jordan, and this idea was thought of because of the problems that occur in the markets of the difference in prices between goods, one of the main problems faced by farmers is the absence of agricultural production organization, there is no guide for farmers what and when to plant a specific crop.

## Acknowledgment

We want to sincerely thank Dr. Zakaria for his invaluable advice and assistance during this effort. His knowledge and advice have been crucial in assisting us in reaching our objectives. We also acknowledge and appreciate the ongoing support and inspiration from our families and friends. Finally, we would like to express our gratitude to our professors for their constructive criticism and suggestions, which have enabled us to enhance our work.

# Content

[Undertaking ii](#_Toc86397196)

[ABSTRACT iii](#_Toc86397197)

[Acknowledgment iv](#_Toc86397198)

[List of Tables vi](#_Toc86397199)

[List of Figures vii](#_Toc86397200)

[List of Acronyms and Abbreviations viii](#_Toc86397201)

[CHAPTER 1: INTRODUCTION 9](#_Toc86397202)

[1.1 Overview 9](#_Toc86397203)

[1.2 Project Motivation 9](#_Toc86397204)

[1.3 Problem Statement 10](#_Toc86397205)

[1.4 Project Aim and Objectives 10](#_Toc86397206)

[CHAPTER 2: Planning phase 11](#_Toc86397207)

[2.1 Scope of the project 11](#_Toc86397208)

[2.2 Project risks and Product risks 11](#_Toc86397209)

[2.3 Feasibility study 11](#_Toc86397210)

[2.4 Project Schedule 12](#_Toc86397211)

[2.5 Project Software and Hardware Requirements 12](#_Toc86397212)

[CHAPTER 3: Related Existing System 13](#_Toc86397213)

[CHAPTER 4: Requirement Engineering and Analysis 14](#_Toc86397214)

[4.1 Used Techniques 14](#_Toc86397215)

[4.3 Functional Requirement & Modelling 14](#_Toc86397216)

[4.3.1 Use Case Diagram 14](#_Toc86397217)

[4.3.2 Use Cases: Description & Details 14](#_Toc86397218)

[4.4 Nonfunctional Requirements: Quality & Constriants 15](#_Toc86397219)

[CHAPTER 5: Architecture & Design 16](#_Toc86397220)

[5.1 Software Architecture 16](#_Toc86397221)

[5.2 Software Detailed Design 17](#_Toc86397222)

[5.2.1 Use Cases Internal Interactions 17](#_Toc86397223)

[5.2.2 Class Diagram 17](#_Toc86397224)

[5.2.3 State transition Diagram 17](#_Toc86397225)

[5.2.4 Data Storage Organization 17](#_Toc86397226)

[CHAPTER 6: Implementation Plan & Prototyping 18](#_Toc86397227)

[6.1 Introduction 18](#_Toc86397228)

[6.2 Prototyping 18](#_Toc86397229)

[CHAPTER 6: Testing Plan 19](#_Toc86397230)

[6.1 Tools 19](#_Toc86397231)

[6.2. System Testing Plan (Black box) 19](#_Toc86397232)

[Conclusions 20](#_Toc86397233)

[REFERENCES 21](#_Toc86397234)

[Appendix-A: Manual 22](#_Toc86397235)

## List of Tables

Table 1: Table Example 15

## List of Figures

[Figure 1: System Overview 16](#_Toc86397237)

## List of Acronyms and Abbreviations

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | API | Application Programming Interface | | REST | Representational State Transfer WS | | RPC | Remote Procedure Call | |  |
| Table 1 |  |
|  |  |

## CHAPTER 1: INTRODUCTION

One of the main problems faced by farmers is the absence of agricultural production organization. Here we mean that there is no organization of production of various plant crops, especially vegetables. More in details, there is no guide for farmers what and when to plant a specific crop, including the presence of an abundance amounts of some agricultural products, this leads to a decrease in their prices, because the large supply in an unorganized way, which in turn leads to huge losses for farmers producing these crops. and thus, lead to an increase in the prices of those products, which causes a burden on the consumers’ budget. Therefore, to economize consumers budget and to minimize farmers losses, the idea of organizing agricultural production in Jordan must be worked on. Create electronic application to regulate the production of agricultural crops in Jordan and to supervise it by specialists who ensure that the instructions for the cultivated crops and their planting times are followed and weather condition.

### 1.1 Overview

### Work will be done to activate a special application under the name Organizing Agricultural Production, and the application will contain a set of instructions for farmers and set the goal of the application to help farmers know the suitable crop for cultivation. He developed several information about agricultural crops, how they are available in Jordan, their seasons, cultivation, etc., and reformed the marketing process inside Jordan and Observing weather and weather conditions in all regions of Jordan.

### 1.2 Project Motivation

We came up with this idea due to there is no guide for farmers on what and when to plant a specific crop, including the presence of an abundance amounts of some agricultural products. This project is one of the planning projects which proposed a new idea for Jordan such Organizing Agricultural Production, importance of the project Improving and maintaining agricultural products and their availability in all markets on an ongoing basis and Giving each agricultural product its right in the markets and Developing agriculture in Jordan in general.

### 1.3 Problem Statement

The lack of structure in agricultural output, namely the cultivation of diverse plant crops, particularly vegetables, in Jordan, is the issue this project seeks to address. The biggest problem is the lack of instructions for farmers on what and when to grow a specific crops, and not observing weather and weather conditions in all city of Jordan. which causes several issues. This includes an overstock of some agricultural goods that lowers prices because of an unchecked growth in supply, resulting in large losses for farmers. Contrarily, poor production management can result in a shortage of some agricultural products or their unavailability at certain times of the year, raising costs and straining customers' budgets.

### 1.4 Project Aim and Objectives

The aim of the "Organizing Agricultural Production " project is to address the problem of unregulated agricultural production in Jordan. The project aims to develop and implement application that will provide farmers with guidance on what to plant and when to plant it, based on the local weather and soil conditions, in order to optimize crop yields and reduce losses. The platform will also allow for better coordination between farmers, distributors, and consumers, and will ultimately contribute to a more stable and sustainable agricultural sector in Jordan.

## CHAPTER 2: Planning phase

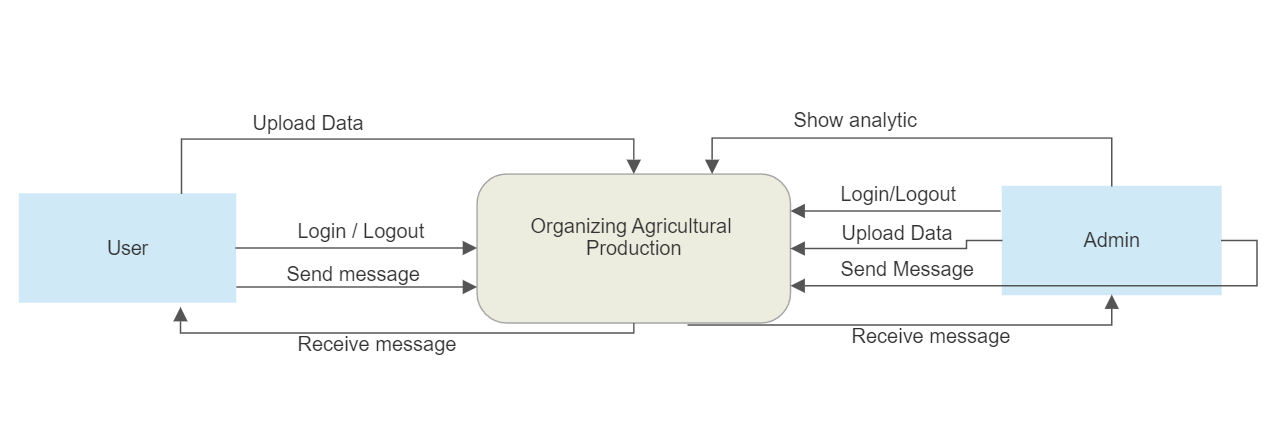
### 2.1 Scope of the project

•It the application is a tool for analyzing agricultural crop production quantities, their availability in the market, and the quantities consumed during the annual seasons (winter and summer) in previous years for each region in Jordan. It also monitors the weather conditions in all regions across Jordan.

•The farmer can see the cultivated quantity in dunums for each crop in the current season for each region in Jordan, and to give the farmer some advice about agriculture, the Ministry of Agriculture in Jordan.

• One of our project’s features is that it shows the percentage required of each crop in the months of the year and the percentage of the number of farmers who will plant each crop and monitor Weather in all cities Jordan.

**Figure 1** shows the context diagram for the system. Farmers can login and logout, then can upload data about how much they will plant and which kind of plant they will plant, and they can send messages and receive messages. The Ministry of Agriculture in Jordan can show analytics about crops and send messages and receive messages with farmers.



**Figure 1: Context Diagram**

### 2.2 Project risks and Product risks

various issues arise in our system, and these issues are as follows:

• Data accuracy.

• A farmer can add land, this land is accessed using the coordinates of a location installed on those lands; if the coordinates are incorrect, there is a problem.

• To access and benefit from system functions, this system necessitates the user's

requirement for Internet access.

• Requires Regular Maintenance.

These are only a few of the issues that can arise in the system and must be addressed.

### 2.3 Feasibility study

Through discussion and after extensive investigation and analysis, our team feels that this system is practicable. We discussed the constraints of available time, resources, and budget to build this system.

**1. Financial Feasibility:**

We believe that this project does not have any financial risks because it does not offer any product and the only financial requirement is the cost of hiring a development team to implement it, the project is financially viable.

**2.Technical Feasibility:**

Our team believes that this project requires technologies reachable for every developer. which are: *Android Studio*: we will use this technology to implement the system using language dart with framework (flutter) The previous language is used to implement the project’s Frontend, in addition the Backend of the project which we will use (PHP) to implement. *Data Base*: We will use (MySQL) to design the required database. The project is technically feasible.

**3.Legally Feasibility:** We need the approval of the Ministry of Agriculture.

### 2.4 Project Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| Chapter: Task | Duration | Start | End |
| introduction, Overview, Motivation, Problem, Aim. | One week | 5-April-2023 | 12-April-2023 |
| Scope, Risks, Feasibility, Software, and hardware requirements. | One week | 13-April-2023 | 20-April-2023 |
| Existing Systems. | One week | 21-April-2023 | 28-April-2023 |
| Used Techniques, Functional requirements, use case diagram, Use case description and details, Nonfunctional. | Two weeks | 5 - May -2023 | 17-May-2023 |
| Software Architecture | Two weeks | 17-May-2023 | 30 -May-2023 |
| Prototyping | 3 days | 31-May-2023 | 3- Jun -2023 |
| Testing  Table 2 | 3 days | 3 - Jun -2023 | 7-Jun-2023 |

### 

### 2.5 Project Software and Hardware Requirements

• For developers Software requirements:

1- android studio

2- PhpMyAdmin.

3-MySql

4-POSTMAN

Hardware requirements:

1-Laptop / PC.

2-At least 4GB of RAM.

3-Internet.

• For users

Software requirements:

Internet browser.

Hardware requirements:

1-Smartphone or Tablet device.

2-At least 2GB of RAM.

3-Internet.

## CHAPTER 3: Related Existing System

Existing Systems:

**3.1 AGMRI**

**Figure 2** shows the AGMRI application.

AGMRI is an agricultural technology company that provides an app called "AGMRI - Automated Crop Intel." The AGMRI app is designed to help farmers and agronomists monitor and manage their crops more efficiently using advanced analytics and remote sensing technologies Some of the key features of the AGMRI app may include: Crop Monitoring and Analytics and Insights and Field Management and Notifications and Alerts.

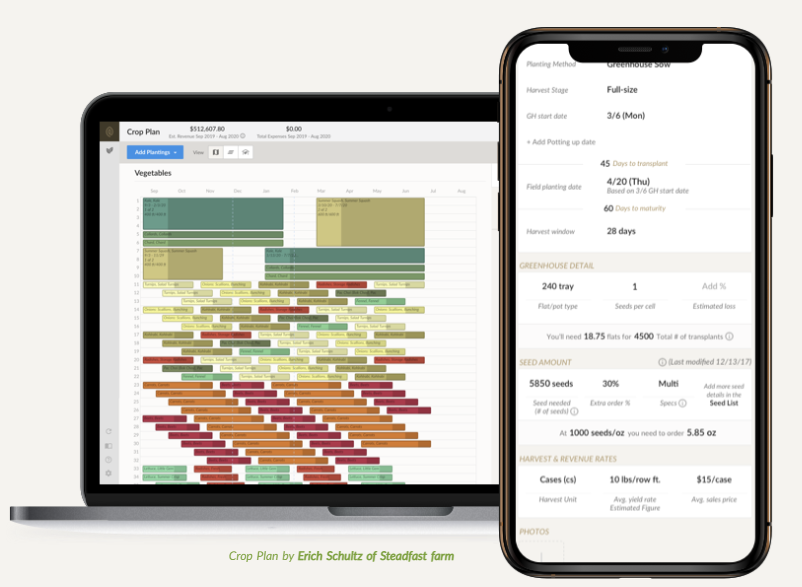


**Figure 2**

**3.2 Tend**

**Figure 3** shows the Tend platform

Tend is an agricultural management platform that provides an app for organizing and overseeing farm operations. It includes features such as task management, record-keeping, inventory tracking, and labor management. The app offers real-time insights and analytics to optimize productivity and streamline operations.



**Figure 3**

**3.3 Agworld**

**Figure 4** shows the **Agworld** application

Agworld is an application that enables farmers to plan, track, and manage their farming activities. It offers features like field mapping, crop planning, task scheduling, input tracking, and financial management. It allows collaboration and communication among team members, improving coordination and efficiency.



**Figure 4**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Advantages** | **AGMRI** | **Tend** | **Agworld** | **Organizing agricultural production**  **(our project)** |
| View the proportions of crops grown in country | **✔** | ✖ | ✖ | **✔** |
| Notifications and Alerts | ✔ | ✖ | ✖ | ✔ |
| communicate with each other(farmers) | ✖ | ✖ | ✖ | ✔ |
| Provide analyzes about crops in the last years | ✖ | ✖ | ✖ | ✔ |
| Provide what the market needs of crops | ✖ | ✖ | ✖ | ✔ |
| monitored weather in every day | ✖ | ✖ | ✖ | ✔ |
| optimize productivity and streamline operations about crops | ✔ | ✔ | ✔ | ✔ |
| support a lot of languages | ✖ | ✖ | ✖ | ✔ |
| Generate reports about crops and weather every day | ✖ | ✖ | ✖ | ✔ |

Table 3

distinguish project from other existing systems.

## CHAPTER 4: Requirement Engineering and Analysis

### 4.1 Used Techniques

**• Brainstorming**

* This helped us when we thought of providing new ideas or solutions to problems that faced our farmers in the last few years.

**• Monitoring**

* For three months, we asked the farmers about the results that confirm the problem that we identified by using Brainstorming.
* We asked the farmers in Jordan about the problem to collect as much information as possible.

**• Group interview**

* Our collaborative team engaged in weekly interviews to explore innovative solutions to this problem, this approach allowed us to gather diverse perspectives from various stakeholders.

### 4.2 Functional Requirement & Modelling

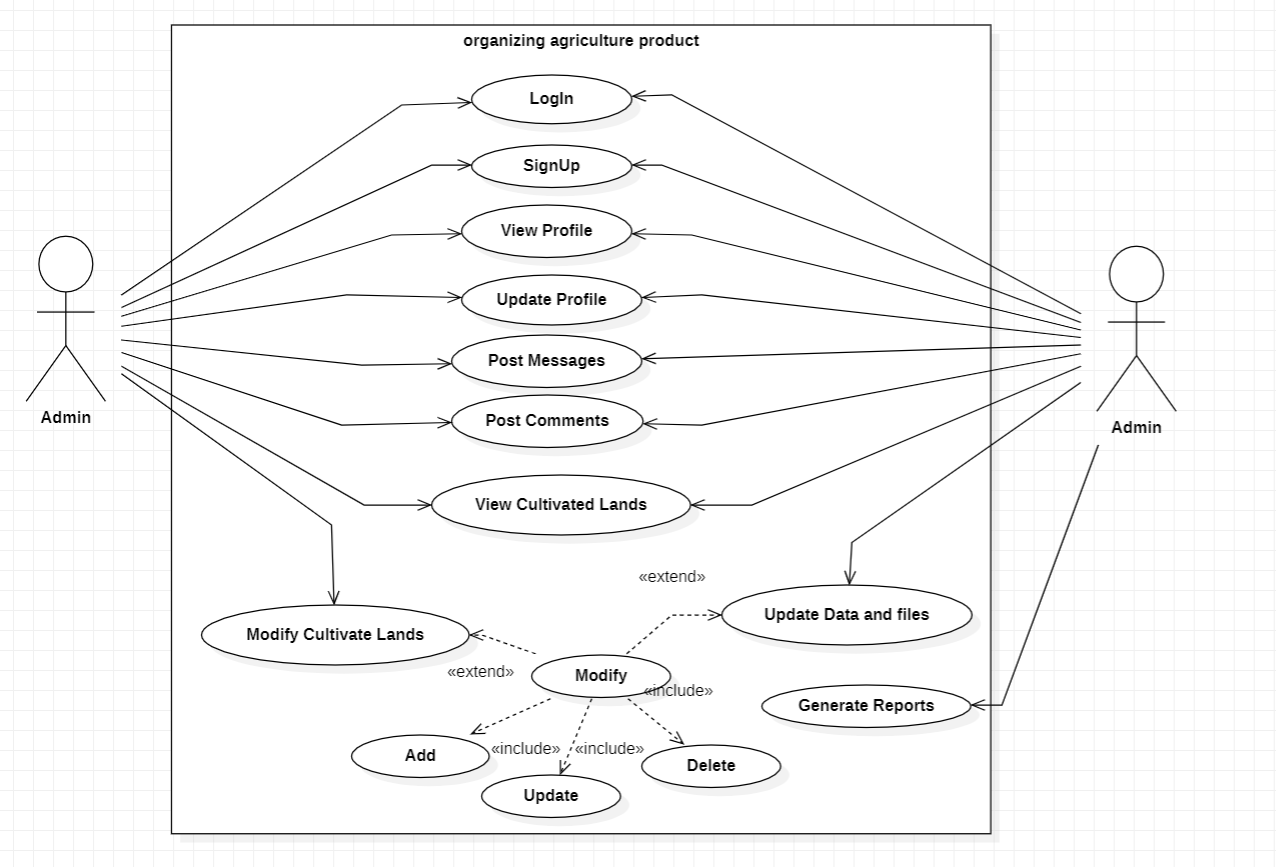
|  |  |
| --- | --- |
| Actor  (User) | Functional Requirement |
| 1 | The system shall allow the user to log in with his information to the system, if he is singed up before. |
| 2 | The system shall allow the user to see their profile. |
| 3 | The system shall allow the user to update their profile. |
| 4 | The system shall allow the user to change password by E-mail or phone number when they forgot it. |
| 5 | The system shall allow the user to see the list of cultivated land for each crops and city. |
| 6 | The system shall allow the user to Post messages and add comments on other posts. |
| 7 | The system shall allow the user track and display the quantity of vegetables and fruits produced per city in Jordan. |
| 8 | The system shall allow the user should be able to add, update, and delete information about the crops they want to grow. |
| 9 | The system shall allow the user should be able to see weather condition every day |
| 10 | the system shall allow help and support services to both users. |
| 11 | the system shall allow the user switch language in Application |
| 12 | the system shall allow the user to view the location land address using google maps |

Table 4

|  |  |
| --- | --- |
| Actor  (Admin) | Functional Requirement |
| 1 | The system shall allow the admin to see their profile. |
| 2 | The system shall allow the admin to see the list of available and cultivate land |
| 3 | The system shall allow the admin to check cultivate land |
| 4 | The system shall allow the admin to send Emails to the users |
| 5 | The system shall allow the admin to generate reports |
| 6 | The system shall allow the admin to Post messages |
| 7 | The system shall allow the admin to add, update and remove data |
| 8 | The system shall allow the admin to comment to other messages |
| 9 | The system shall allow the admin to modify land |
| 10 | the system shall allow the admin switch language in Application |

Table 5

#### 4.2.1 Use Case Diagram



**Figure 5**

#### 4.2.2 Use Cases: Description & Details

**1.Sign up**

|  |  |
| --- | --- |
|  | Sign Up |
| Actor(s) | User |
| Precondition | The system displays an account registration page. |
| Normal flow | 1.The user enters the username, e-mail, phone number, password, confirm password.  2.The user chooses to sign up as farmer.  3.The system shall verify the correctness of the information  4.The system shall open log in page. |
| Post condition | The account has been successfully registered. |
| Exception | If the user enters the password and it does not match the confirmation password, the system shall ask the user to enter the password and confirm it again.  If the email or phone have invalid format the system, ask the user to enter it again |

Table 6

**2.Log-in**

|  |  |
| --- | --- |
|  | Log-in |
| Actor(s) | User, Admin |
| Precondition | The system display log in page. |
| Normal flow | 1.The user entered email and password.  2.The system shall verify the E-mail and password.  3.The system shall open the home page. |
| Post condition | The user has been successfully login. |
| Exception | 1. If the user log in for the first time, the system shall prompt the user to sign up. 2. If the user enters the wrong e-mail or password, the system shall ask the user to enter it again. |

Table 7

**3.View profile page**

|  |  |
| --- | --- |
|  | View profile page |
| Actor(s) | User, Admin. |
| Precondition | The actors are logged in |
| Normal flow | 1.The actors shall click on “Profile” icon.  2.The actors (User, Admin) can see/update the information |
| Post condition | The profile page has been successfully displayed |
| Exception |  |

Table 8

**4. view cultivated lands and crop percentage**

|  |  |
| --- | --- |
|  | view cultivated lands and crop percentage |
| Actor(s) | User, Admin |
| Precondition | The actors enter to the home page. |
| Normal flow | 1.The actors click on “show analytic” button.  2.The system shows the cultivated lands and crop percentage. |
| Post condition | Available analytic page displayed successfully. |
| Exception | If the Internet interrupt from the user, the system cannot show the  available analytic. |

Table 9

**5.modfiy the cultivated lands**

|  |  |
| --- | --- |
|  | modify the cultivated lands |
| Actor(s) | User |
| Precondition | The actors enter to the home page. |
| Normal flow | 1.The actors click on “modify my land” button.  2.The system shows the three button add and remove and update.  3. add the crop that will be cultivate and land area or remove information about crops or update information about all crops and area |
| Post condition | Available modify my land displayed successfully. |
| Exception | If the Internet interrupt from the user, the system cannot show the available analytic. |

Table 10

**6. Generate Reports**

|  |  |
| --- | --- |
|  | Generate Reports |
| Actor(s) | Admin |
| Precondition | The user reserved. |
| Normal flow | 1. The admin enter the reports Date/Time.  2. The system displays the desired report.  3. The admin generate reports. |
| Post condition | The reports successfully done. |
| Exception |  |

Table 11

**7.Update my profile**

|  |  |
| --- | --- |
|  | Update my profile |
| Actor(s) | User, Admin. |
| Precondition | The Actors logged in. |
| Normal flow | 1. Actors shall go to the profile page.   2. Actors shall click on “Edit profile” button.  3. Actors can change their personal information.  4. Actors can confirm the new information by click on “Change” button. |
| Post condition | The profile has been updated successfully. |
| Exception | If the edited information has wrong format looks like Email the  system ask to enter the wrong information again or save the old info |

Table 12

**8.Post Message**

|  |  |
| --- | --- |
|  | Update my profile |
| Actor(s) | User, Admin. |
| Precondition | The Actors logged in |
| Normal flow | 1. Actors shall write any message.   2.Actors shall Post this message. |
| Post condition | The Message successfully Posted. |
| Exception | If the messages is empty ask the user to write a message. |

Table 13

**9.Log Out**

|  |  |
| --- | --- |
|  | Log Out |
| Actor(s) | User, Admin. |
| Precondition | The actors shall already log in |
| Normal flow | 1. Actors shall go to their profile.   2.Actors shall click on “log out” button. |
| Post condition | The log out successfully done. |
| Exception |  |

Table 14

**10. Modify Cultivated Land**

|  |  |
| --- | --- |
|  | Modify Cultivated Land |
| Actor(s) | User |
| Precondition | The actors shall already log in |
| Normal flow | 1. Actors shall go to their profile. 2. Actors shall click on the Lands button. 3. Choose to remove or add or update. |
| Post condition | Successfully modified. |
| Exception |  |

Table 15

**11. update data**

|  |  |
| --- | --- |
|  | update data |
| Actor(s) | Admin |
| Precondition | The actors shall already log in |
| Normal flow | 1. Actors click the statistics button. 2. Actors choose to add or remove or update date |
| Post condition | Successfully updated. |
| Exception |  |

Table 16

### 4.4 Nonfunctional Requirements: Quality & Constriants

|  |  |  |
| --- | --- | --- |
| Nonfunctional Requirements | Explain | Use Technics |
| Performance | The app should be responsive and perform efficiently to handle many users and data. | 1.Optimize-Network-Requests:  -lower the payload size by compressing data or using more efficient data formats like JSON instead of XML.  -Implement intelligent caching mechanisms to avoid redundant requests and serve cached data when appropriate.  2. Efficient Flutter UI:  -improve Flutter UI by minimizing unnecessary widget rebuilds and updates. the appropriate widget lifecycle methods (didUpdateWidget, shouldRebuild, etc.) to control when widgets should update.  -will Employ the Flutter ListView.builder and GridView.builder constructors to efficiently load and display large amounts of data. |
| Reliability | The app should be available and reliable, with minimal downtime and the ability to recover from failures. | 1.Automated Testing:  -Implement automated testing for both your Flutter app and PHP backend. we will Use unit tests, integration tests, and end-to-end tests to verify the reliability and functionality of your codebase.  -we will write test cases that cover various scenarios, including edge cases, error conditions, and data inconsistencies, to ensure robustness and reliability. |
| Security | The app should ensure the confidentiality, integrity, and privacy of user data, including authentication and access control measures. | 1.Protection Against Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF):  Implement measures on the server-side (PHP) to prevent XSS attacks by sanitizing user-generated content, using output encoding, and applying Content Security Policy (CSP) headers to restrict the execution of malicious scripts.  lower CSRF attacks by implementing CSRF tokens and validating them on each request to ensure that requests originated from your application and are not forged.  2. Secure Storage:  sure sensitive data, such as user credentials or API keys, are securely stored on the device using encryption techniques provided by Flutter, such as the flutter\_secure\_storage package, and avoid storing sensitive data in plain text or in insecure locations. |
| Usability | The app should be user-friendly, intuitive, and accessible to users with different levels of technical expertise. | 1.Intuitive and Consistent User Interface:  we will design a user interface (UI) that is intuitive and easy to navigate. and use familiar UI patterns and adhere to platform-specific design guidelines (e.g., Material Design for Android, Cupertino for iOS) to provide a consistent and familiar experience.  and sure clear and concise labeling of buttons, menus, and navigation elements to help users understand their purpose and functionality.  2.improve the app performance of to provide a smooth and responsive user experience. Minimize loading times, reduce latency, and optimize animations and transitions to create a snappy and enjoyable user interface. |
| Scalability | The app should be able to accommodate an increasing number of users and data without compromising performance. | we will use technologies such as **load balancers and reverse proxies** to distribute incoming requests across multiple servers, ensuring even load distribution and increasing the capacity to handle more concurrent users.  Implement caching mechanisms (such as Redis or Memcached) to reduce database load and improve response times for frequently accessed data. |
| Availability | The app should be accessible to users with minimal downtime, ensuring uninterrupted service. | 1.Design for Redundancy: Implement redundancy at various levels of your application architecture to minimize the impact of failures. This includes redundant servers, load balancers, databases, and network infrastructure. we will send across multiple servers or data centers to ensure that if one component fails, the application can continue to function.  2. Automatic Failure Detection and Recovery: Implement monitoring systems that constantly monitor the health and availability of your application components, including servers, databases, and network connections. |

Table 17

## CHAPTER 5: Architecture & Design

### 5.1 Software Architecture

**Figure 6** shows the Logical View.

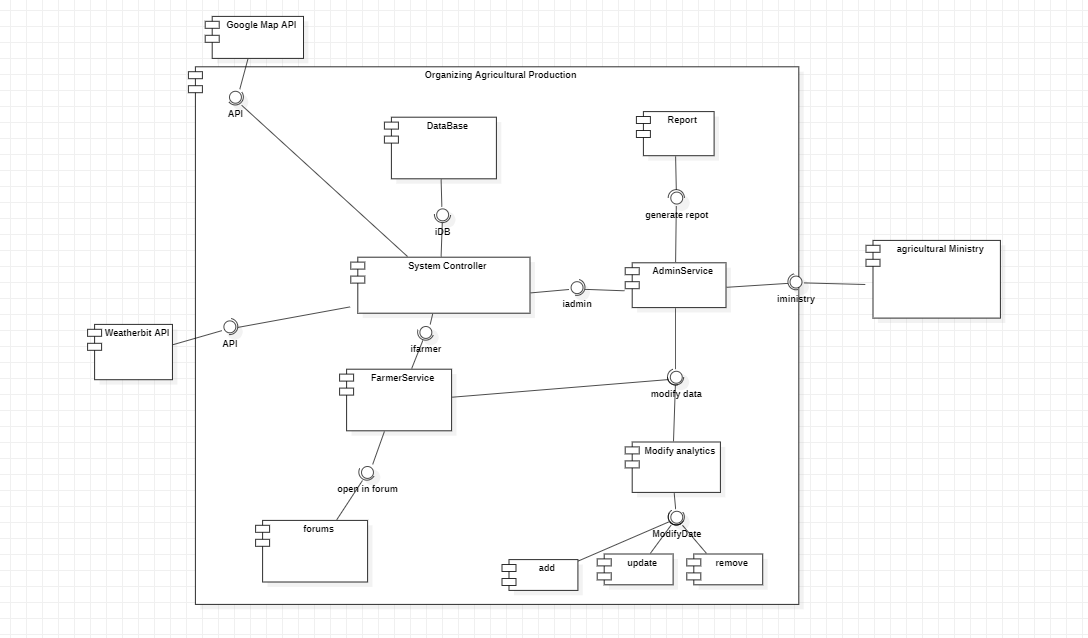
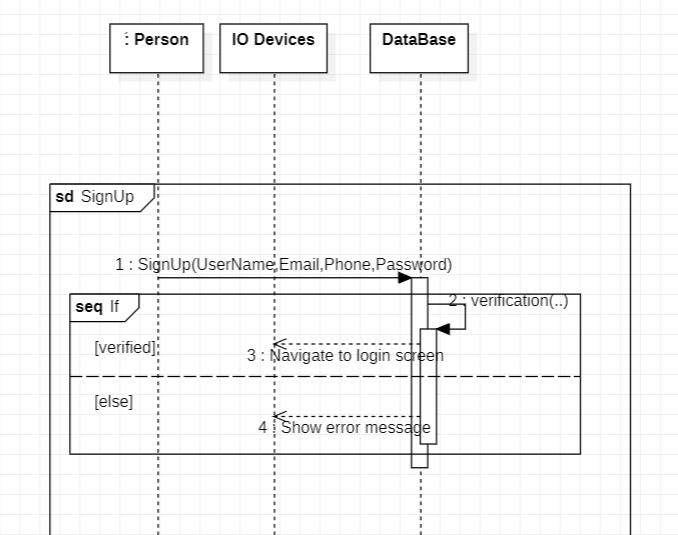
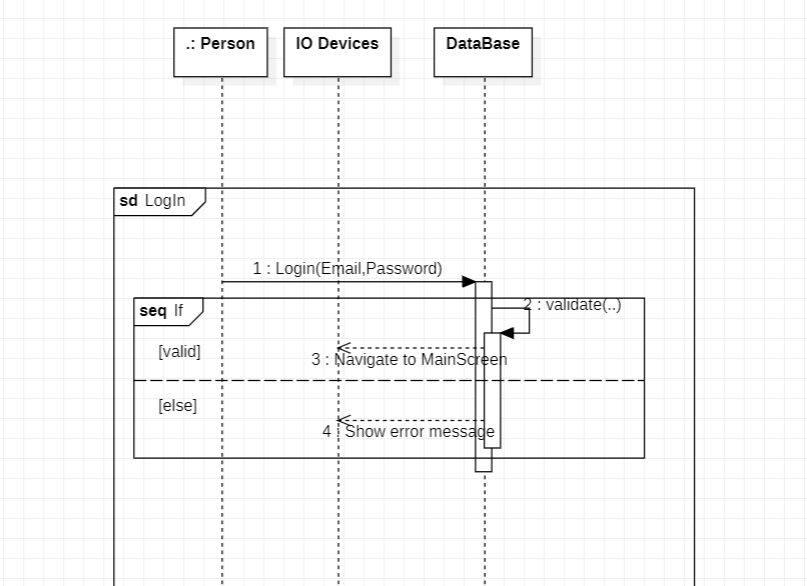


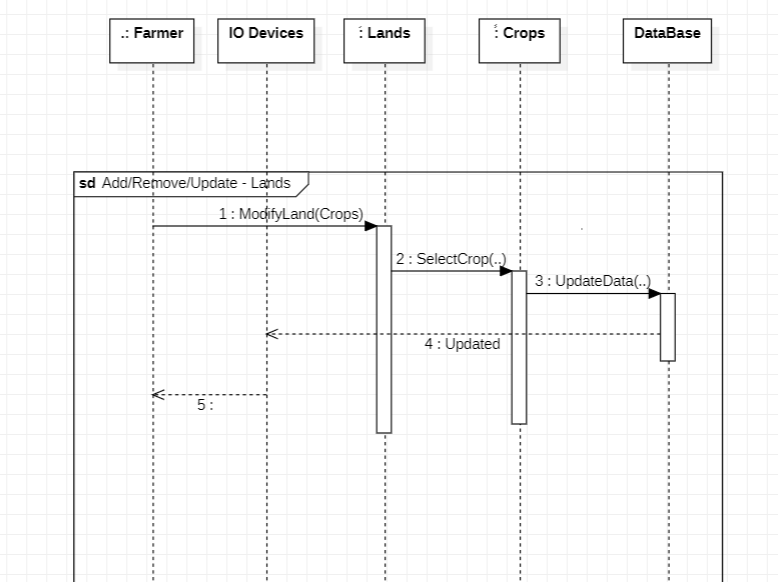
Figure 6: Architecture

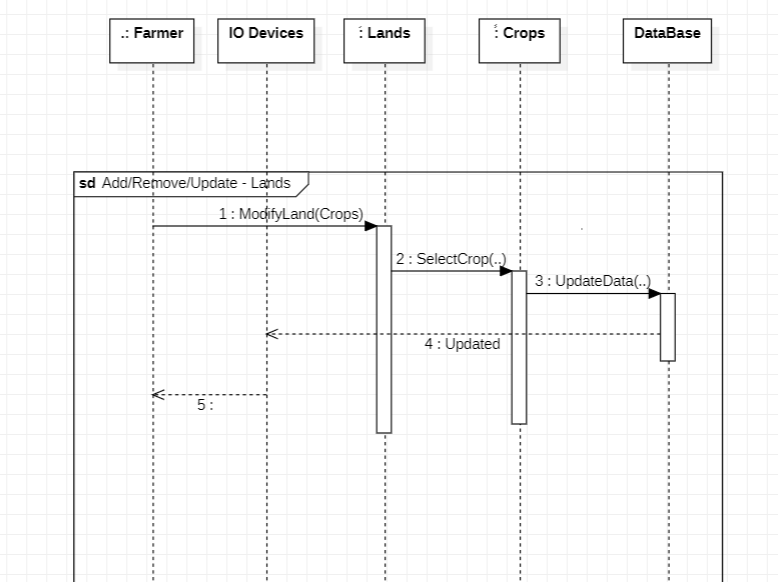
### 5.2 Software Detailed Design

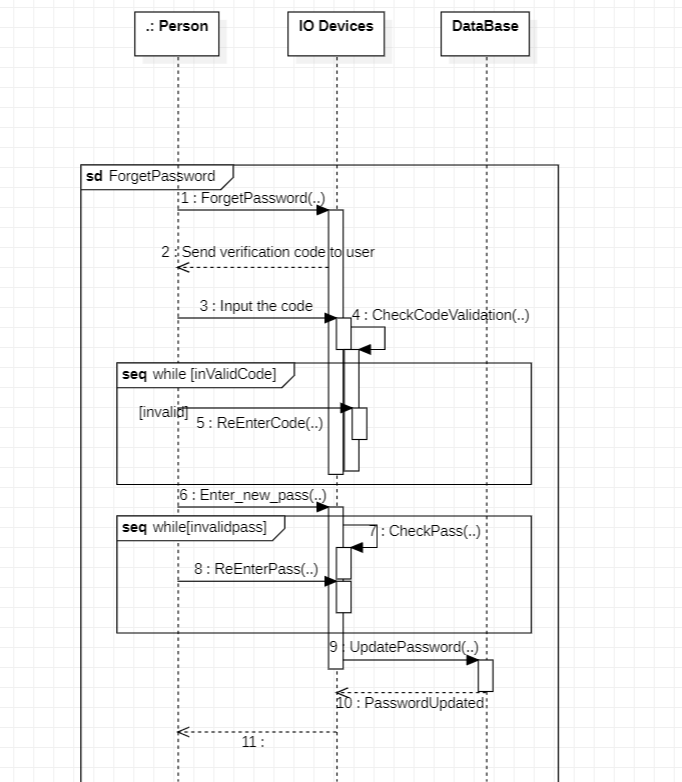
#### 5.2.1 Use Cases Internal Interactions

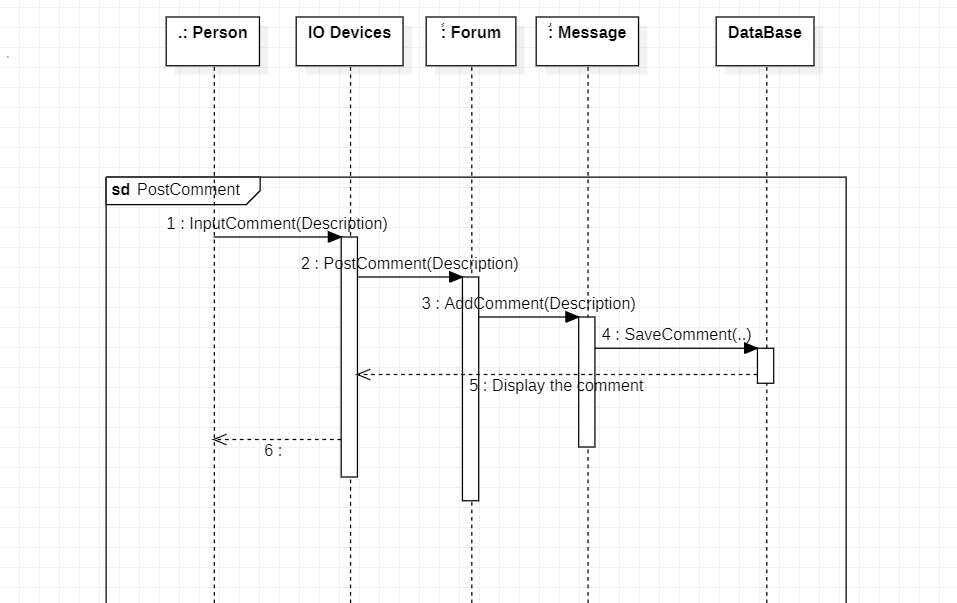


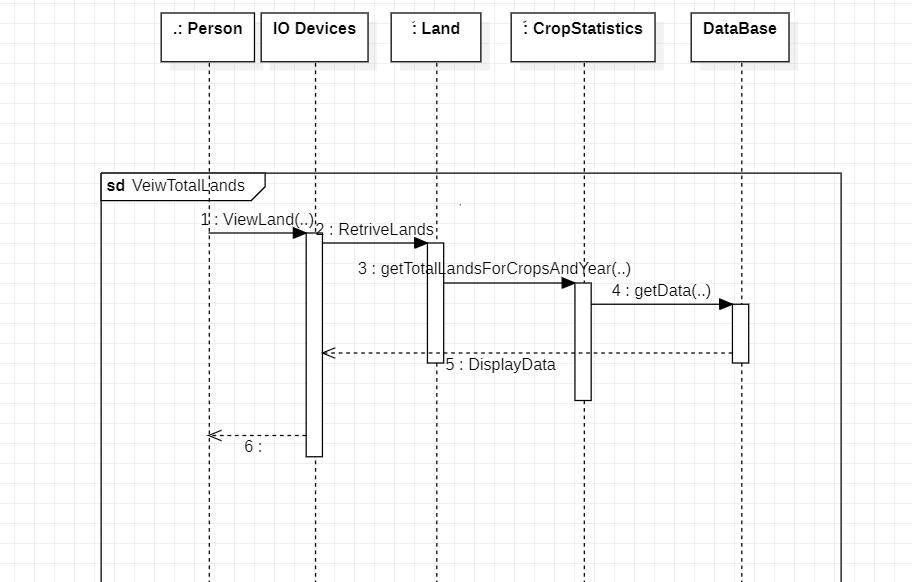


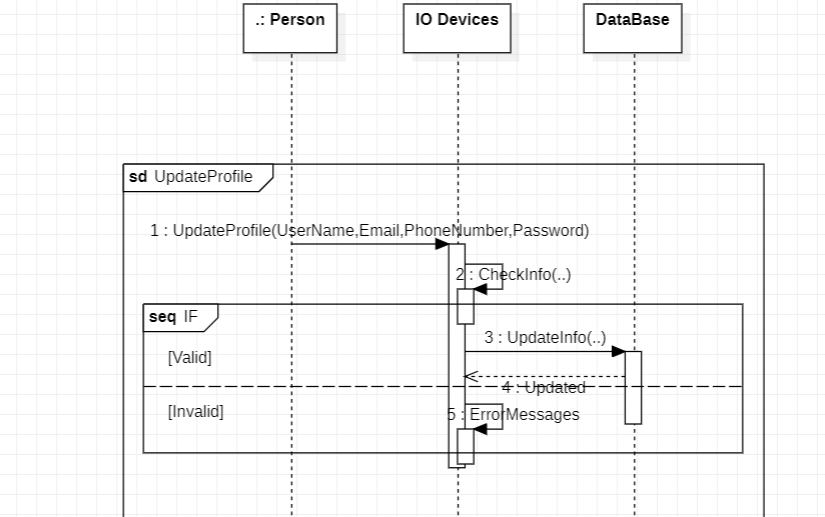










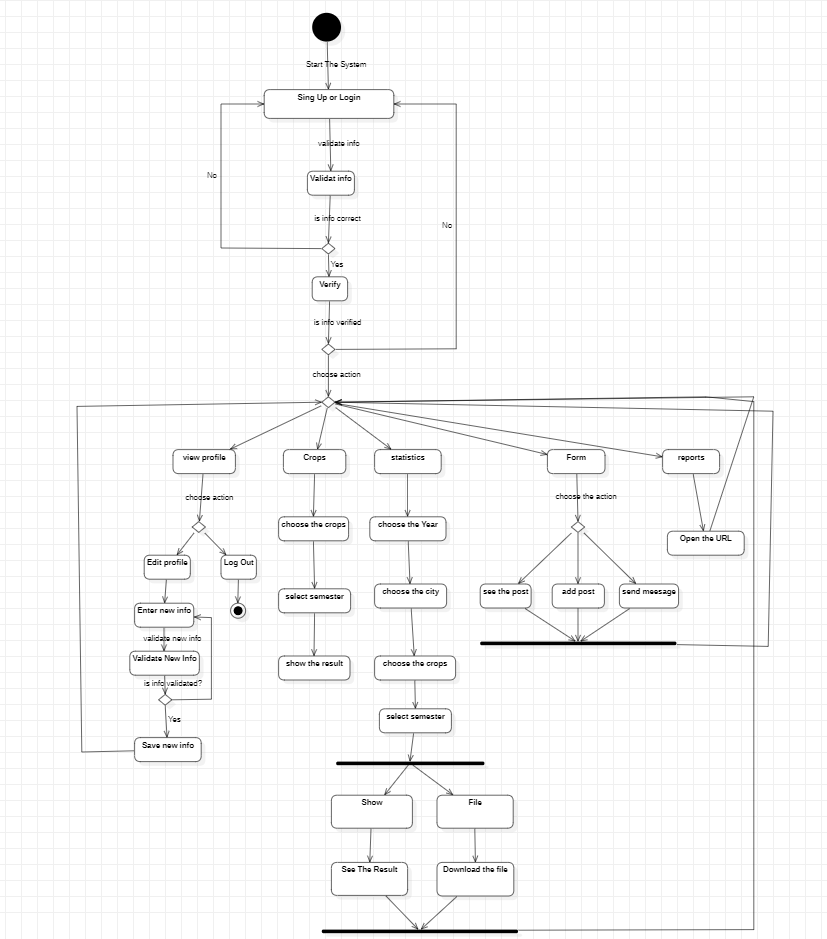




#### 5.2.2 Class Diagram

#### 

#### 5.2.3 State transition Diagram



#### 5.2.4 Data Storage Organization

You need to provide ER diagram or any other diagram to show the data structure in database.

## CHAPTER 6: Implementation Plan & Prototyping

### 6.1 Introduction

The tools, third-party apps, and languages required to build app organizing agriculture production.

1-Front End: Dart with framework Flutter

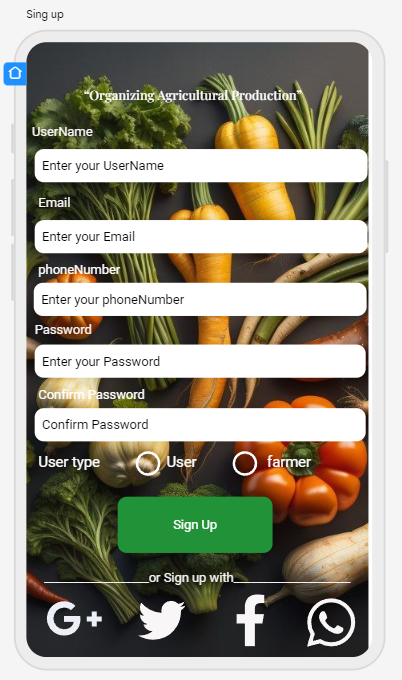
2-Back End : PHP

3-DataBase : Myphp Admin

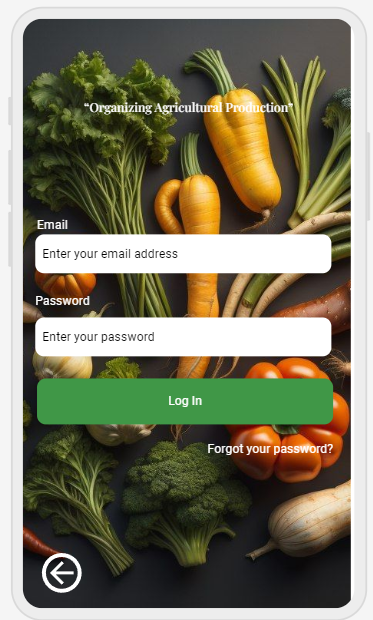
4- Third-party apps : Google Maps and WeatherBit

### 6.2 Prototyping

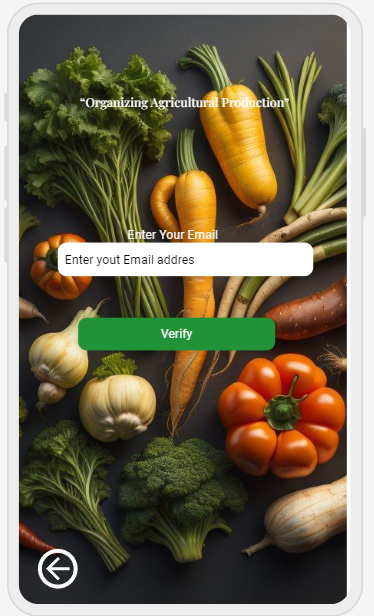
1.Sign up page



2.Log in page



3.Forgot password page.

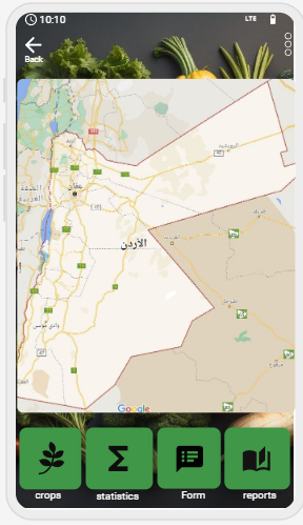


4.Verify page.

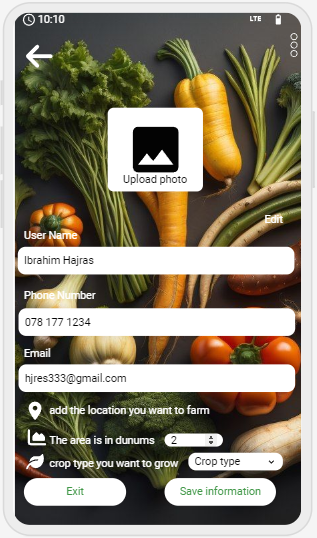
A screenshot of a cell phone

Description automatically generated with low confidence

5.Home page



6.profile page



7.crop page



8.number statistics page

A picture containing text, vegetable, carrot, food group

Description automatically generated

9. statistics page

A screenshot of a phone

Description automatically generated with medium confidence

10.Show page open statistics page



11.files page open statistics page

A screenshot of a phone

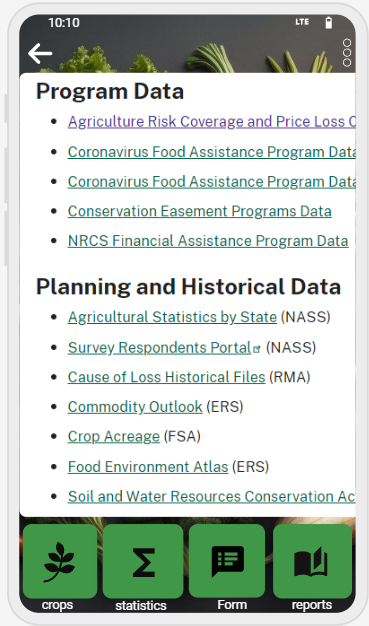
Description automatically generated with medium confidence

12.form page

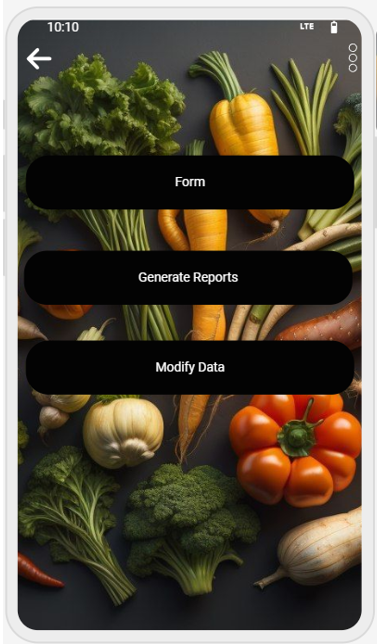
A screenshot of a phone

Description automatically generated with low confidence

13.report page



14.Home page Admin



15.Weather page



## CHAPTER 7: Testing Plan

### 6.1 Tools

1-Unit Testing.

2- Widget Testing

3- Integration Testing

4- Golden Testing

5- Mocking Frameworks

6- Flutter Driver

### 6.2. System Testing Plan (Black box)

1.Log in for User

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Conditions | Valid Partitions | Tag | Invalid Partitions | Tag | Valid Boundaries | Tag | Invalid Boundaries | Tag |
| E-Mail | 8 to 30Chars  Special Chars  0 to 9 Digits | VP1  VP2  VP3 | Lees than 8  more than 20  Invalid char  Null | IP1  IP2  IP3  IP4 | 8 Chars  30 Chars  Special Chars | VB1 VB2 VB3 | 7 Chars  31 Chars  0 char | IB1 IB2 IB3 |
| Password | 8 to 20 digit and Char  Valid digits and Chars | VP4  VP5 | Less than 8 digits and chars  More than 20 digits and chars  Nonnumeric  Non-symbol  Noncapital letter  Null | IP5  IP6  IP7 IP8 IP9  IP10 | 8 digits  20 digits | VB4 VB5 | 7 digits  21 digits | IB4 IB5 |

Table 18

Test cases for log in

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Description | Expected outcome | Tag covered |
| 1 | E-Mail: hjres333@gmail.com Password: Ibra1234\* | Log-in Successfully | VP1, VP2, VP3, VP4,VP5, ,VB1,VB3,VB4, |
| 2 | E-Mail:ibra& Password: 1234567 | Error Message | IP1,IP3,IB1,IP5, IB4,IP8,IP9. |

Table 19

**2.Sign up for user**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Conditions | Valid  Partition | Tag | Invalid  Partition | Tag | Valid  Boundary | Tag | Invalid  Boundary | Tag |
| FullName | 5 to 20 Chars  Valid Char | VP1  VP2 | Less than 5  More than 20  Null | IP1  IP2  IP3 | 5 Chars  20 Chars | VB1  VB2 | 3 Chars  21 Chars  0 Char | IB1  IB2  IB3 |
| E-mail | 8 to 30 chars  Special chars  0 to 9 Digits | VP3  VP4  VP5 | Less than 8  More than30  Invalid char  Null | IP4 IP5 IP6  IP7 | 8 Chars  30 Chars  Special Chars | VB3  VB4  VB5 | 7 Chars  31 Chars  0 Char | IB4  IB5  IB6 |
| Phone number | 10  Digit only | VP6 | Less than 10  More than 10  Special char | IP8  IP9  IP10 | 10 Digit only | VB6 | 9 Digits  11 Digits | IB7 IB8 |
| Password | 8 to 20 Digits and Chars  Valid digits and chars | VP7  VP8 | Less than 8 digits and chars  More than  20 digits and chars  Nonnumeric Non-symbol  Non-Capital letter  Null | IP11  IP12  IP13  IP14  IP15  IP16 | 8 digits  20 digits | VB7  VB8 | 7 digits  21 digits | IB9  IB10 |
| Confirm  Password | 8 to 20 Digits and Chars  Valid digits and chars | VP9  VP10 | Less than 8 digits and chars  More than  20 digits and chars  Nonnumeric Non-symbol  Non-Capital letter  Null | IP17  IP18  IP19  IP20  IP21  IP22 | 8 digits  20 digits | VB9  VB10 | 7 digits  21 digits | IB11  IB12 |

Table 20

Test cases for Sing up

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Description | Expected outcome | Tag covered |
| 1 | FullName: Ibrahim Hajras  E-Mail:  [hjres333@gmail.com](mailto:hjres333@gmail.com)  PhoneNumber:0781771234  password: Ibra1234\*  Confirm Password: Ibra1234\* | Log-in Successfully | VP1, VP2, VP3, VP4,VP5, ,VB1,VB3,VB4, |
| 2 | FullName: ibra E-Mail : hjres  PhoneNumber: 0781771234H Password: ibra12  Confirm Password: ibra123 | Error Message | IP5,IB1,IP4, IB4,IP9,IP10, IB8,IP11,IP14, IP15,IB9,IP17, IP20,IP21,IB11. |

Table 21

1. Lands - Crops.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Conditions | Valid  Partition | Tag | Invalid  Partition | Tag | Valid  Boundary | Tag | Invalid  Boundary | Tag |
| Dunums  Or  Dunums Crops | More than 0 | Vp1 | Less than 0  Equal 0 | IP1  IP2 | 1  N/A | VB1 | 0 | IB1 |

Table 22

Other attributes we will use dropdown List so the user will not enter the value.

Test case for lands

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Description | Expected outcome | Tag covered |
| 1 | Number of Dunums = 10 | add Successfully | VP1 |
| 2 | Number of dunums = 0 | Error Message | IP1 , IB1 |

Table 23

4.Crops type and Name City

Name city :

all options (Irbid , Ajloun , Jerash , Mafraq , Balqa , Amman ,Zarqa , Madaba , Karak , Tafilah , Ma'an, Aqaba) in the dropdown list are visible and readable.

that users can select an option from the dropdown list.

the options are displayed in a consistent and uniform manner.

Crops type:

all options (Tomatoes, cucumbers, potatoes, cabbage, peppers, cauliflower, zucchini, beans, Fababean , eggplant) in the dropdown list are visible and readable.

that users can select an option from the dropdown list.

the options are displayed in a consistent and uniform manner.

## Conclusions

In conclusion, the proposed system has been developed with a farmers-centered approach, considering the specific needs and requirements of the farmers. The system encompasses various aspects that cater to the needs of all farmers who will utilize it. Whether individuals have small or large farms, this system will prove to be beneficial and valuable to them. By addressing the concerns and requirements of farmers, we have ensured that the system is inclusive and relevant to a wide range of farmers.

Through the implementation of our "Organizing Agricultural Production" project, we aspire to empower farmers, enhance productivity, and contribute to the overall growth and development of Jordan's agricultural sector.

## REFERENCES

1. Babine au W., Barry P., Furness Z., "Automated Testing within the Joint Training confederation (JTC)", Proceedings of the Fall 1998 Simulation Interoperability Workshop, Orlando, FL, USA. September 1998.
2. Banks C. "Introduction to Modeling and Simulation". Chapter 1 in book “Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains”. Catherine Banks, John Sokolowski Editors. Wiley. New Jersey, 2010.
3. Booth D., Haas H., McCabe F., Newcomer E., Champion M., Ferris C., Orchard D. “Web Services Architecture”. 2004. <http://www.w3.org/TR/ws-arch/>. Accessed November 2010.

## Appendix-A: Manual