AGRICULTURE PRODUCT SUPPLY CHAIN MANAGEMENT APPLICATION

BY

YAP ZI XIN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS) BUSINESS INFORMATION SYSTEMS

Faculty of Information and Communication Technology

(Kampar Campus)

JUNE 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Agriculture Product Supply Chain Management Application

Academic Session: June 2024

I YAP ZI XIN

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:
No.8, Jalan Damai Perdana 2/6e,
Bandar Damai Perdana,
56000, Cheras, Kuala Lumpur.

Tan Teik Boon

Date: 29 August 2024

Date: 13/9/24

FACULTY OF INFORMATION COMMUNICATION AND TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 29 August 2024

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ***Yap Zi Xin*** (ID No: ***20ACB05249***) has completed this final year project entitled "*Agriculture Product Supply Chain Management Application*" under the supervision of Ts. Tan Teik Boon (Supervisor) from the Department of Computer Science, Faculty of Information Communication and Technology.

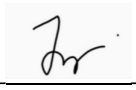
I understand that University will upload softcopy of my final year in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

(*Yap Zi Xin*)

DECLARATION OF ORIGINALITY

I declare that this report entitled "**AGRICULTURE PRODUCT SUPPLY CHAIN MANAGEMENT**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : Yap Zi Xin

Date : 29 August 2024

ACKNOWLEDGEMENTS

I would like to express thanks and appreciation to my supervisor, Ts. Tan Teik Boon and my moderator, Ms. Jin, Tong who have given me a golden opportunity to involve in the application development field study. Besides that, they have given me a lot of guidance in order to complete this project. When I was facing problems in this project, the advice from them always assists me in overcoming the problems. Again, a million thanks to my supervisor and moderator.

To a very special person in my life, Yap Zi Heng, for her patience, unconditional support and love, and for standing by my side during hard times. Finally, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

This Agricultural Product Supply Chain Management Application is a web application for farmers to manage and operate their farming activities and business systematically and efficiently. It plays a vital role in ensuring the sustainability of food production in the agriculture sector around the globe. The aim is to develop this web application for the farmer to enhance their farming productivity and management to ensure the fruits can be supplied in the market to fulfil the customers' demand in every season. This abstract provides an overview of this agriculture product supply chain management application development. It discusses the issues that farmers might encounter in existing supply chain management (SCM) application which may cause a shortage of food supplies. It emphasizes the measures of enhancing farmers' decision-making, expediting food production, and reducing the risk of food shortage to ensure food security. This web application integrates with digital technologies to facilitate monitoring and tracking the estimated harvest quantity of agricultural products and real-time inventory level to improve the accuracy precision of resource allocation and prediction of market demand. Besides, this web application can determine the cycle stage of agricultural product based on the estimated days of each stage. Additionally, task management is a crucial module in this web application as it streamlines the farming system operation, optimizes farmers' productivity, and ensures a well-planned farming activity. By implementing these objectives into this agriculture product SCM application, can ensure the yielding, lower production cost, and sustainability in agriculture sectors. Overall, this web application is useful for the farmer to improve their farm operations and ensure the sustainability of the supply chain to the market.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

SCM	Supply Chain Management
ERD	Entity Relationship Diagram
UID	User Interface Design
AI	Artificial Intelligent
IOT	Internet of Things
CRUD	Create, Retrieve, Update, Delete

Chapter 1
Introduction

Supply Chain Management (SCM) is an all-encompassing measure used to manage the movement of data, resources, and services from suppliers to end consumers by integrating with distribution channels to improve delivery efficiency as well as a system to optimize the production, and distribution and reduce cost while increasing supply chain productivity [1]. SCM has covered the activities from managing the raw material, products manufacturing, warehouse operations, stock taking, and logistic service where to deliver final products to either intermediary's channel or final consumers from the suppliers.

SCM is crucial to the agriculture sector where it can ensure food security for the public due to it assists the farmers in improving farming operations and making better decision-making on supply planning and resource allocations used to fulfil the predicted market demands [2]. However, in this technological advancement era, it is necessary to integrate with technology development to improve farming operations efficiency. It is helpful for farmers to improve their productivity by optimizing supply chain management with high automation, predicted and actual stock amount tracking, and data-driven decision-making [3]. With the technology advancement, it increased the crop yielding sustainability and productivity to fulfil the market demand at every season. This has led to ensuring food security in the global market and minimizing the waste of resources by reducing the harmful environmental impact. This report focuses on developing an SCM application for agriculture sectors for the improvement of farming business operations and management performance purposes.

1.1 Problem Statement and Motivation

In this day and age, the population has increased drastically around the globe which led to a higher demand for agricultural products to ensure the public's food security. Due to this increment in population, the shortage of agricultural products has become a serious issue around the globe as it causes the farmers unable to supply sufficient amounts of agricultural products for them. This is due to the lack of inventory estimation to forecast the total stock of the agriculture products which has lowered accuracy precision of stock allocation, and prediction of market demand. This may cause delays in supplying and delivering agricultural products into the market regardless during the ordinary seasons or in specific seasons due to

underestimation of stock and lead to running out of stock at the time of product delivery preparation. Thus, integrating product stock amount and harvest amount of agricultural products estimation in this web application is necessary to improve the accuracy of prediction of stock availability to be supplied.

Besides, lacking well-planned tasks of farming activities to cultivate agricultural products may reduce farmers' efficiency in performing the activities. Without a well-planned task, the farmers are unable to perform the related works as well as affecting the product yielding and timeline of product delivery. This is because poor planning may cause uncertainty to the product growth due to not assigning the task on time which causes the farmer not to notice that tasks have been assigned leading to overdue tasks. Every delay in tasks may have a negative impact on the growth of agricultural products. Therefore, task assignment with frequency is critical in this web application, as it can automatically calculate the due date of each task by assigning tasks to the selected assignee based on the selected timeline and task frequency every time a farmer assigns a task. This can prevent the farmer from forgetting to assign tasks to the other farmers and ensure the task will be completed on time as well as improve yielding performance.

Lastly, without the calculation to determine the growth stage of agricultural products, it is inconvenient for the farmer to plan the activities to cultivate the agricultural products. This is because the farmers have to observe the products manually frequently to ensure that they can capture the beginning of each stage on time. This has increased farmers' workload and also consumed the farmers a lot of time observing the stage. To reduce the workload on this, this web application must have the function of determining the cycle stage automatically. Thus, the farmer does not need to visit the field and observe the growth condition of the agricultural products product to determine the cycle stage.

1.2 Objectives

Nowadays, due to the higher market demand for agricultural products, the farmers are required to ensure the stock is available at all times to deliver the products to the market to improve the public's food security and reduce hunger. To approach this, the inventory estimation and estimated harvest quantity of agricultural products by month are implemented into this application. It allows to improve the precision of estimation of the stock amount in the inventory that can be delivered. Implementing this estimation function in both harvesting amount and stock level of agricultural products can optimize farming plans to overcome the

issue of underestimating stock amount as well as reduce the lead time of supplying and delivering the products to the market [4].

Besides, improving farmers' productivity in agricultural production performance is important as it is favourable for plant growth. Meanwhile, it can also improve the accuracy of the agricultural products to be harvested within the expected period. Thus, the objective used to approach these is implementing automatic calculation of task due dates based on selected timelines and frequency in task management. It is convenient for the farmers to assign multiple tasks with a selected task type and field at once rather than assigning the task one at a time, which has prevented the problem of forgetting to assign tasks. In addition, the application allows farmers to keep track of and monitor every task progress, and notify the assignee to complete the task to prevent the occurrence of overdue tasks that might affect the growth of the agricultural products and extend agricultural products supply and delivery lead time.

Lastly, the objective of the application is to implement automatic calculation of the cycle growth stage of agricultural products based on the estimation of days for each growth stage. The motivation for implementing this objective is to reduce the frequency of the farmer visits to the field physically and determine the stage of the agriculture products manually. This function saves farmers time to conduct other activities or decision planning as well as to reduce the farmers' workload. On the other hand, determination of the agricultural product growth stage is also crucial to alert the farmer to identify potential environmental issues [10]. This is useful for the farmer to adjust their decision planning to fit with the current situation and manage to solve the issue to reduce the impact of the agricultural product growth. Hence, the farmer can ensure to sustain their crops yielding for their farming business.

1.3 Project Scope and Direction

This project is a web application development for the farmer to manage their farming operation efficiently and smoothly while improving the sustainability of the production of agricultural products to reduce and overcome the issue of food shortage and food security. Thus, in this project, the logistic management, customer management, procurement, and financing management will not be covered.

This project will cover user authentication, field management, task management, inventory management, pest management, resource management, chat, and announcement. Each module except for user authentication and chat, has included create, read, update, and delete (CRUD) operations functions for the farmer to manage and handle the overall operations easily. Other

than CRUD operation functions, the searching, viewing dashboard, set announcement, completion status, filtering, export data, or send notifications functions also will be included in these modules.

The first scope is for user authentications which is required for every farmer to register their account before logging in to this web application. This module has also included a change password function for the farmer to change their password when they forget their password.

The second scope is farm management which includes 2 modules which are fruit or vegetable, and land. The fruit or vegetable module is for the farmer to key in the product planting details such as product name, fruit or vegetable category, growth cycle stage, soil type, water requirement, humidity range, temperature range, sunlight requirement, harvest indicator, harvest method, post-harvest handling, carbon footprint, and sustainability practice. This module will allow the user to conduct CRUD operations and search the product planting details list. The land module, allows the user to view the list of each land with land name, land area, and its location longitude, and latitude. The farmer can add the land by searching for a location plotting an area in the map and key in the land name. While the farmer can also update and delete the land. They can also filter the land list by land name. This module also included a sub-module which is the field module. This sub-module is for the farmer to view all the lists of fields in the selected land with its basic information. However, the farmer can view the detailed information in the dashboard. The farmer can create a new field by plotting an area within the selected land and key in the required data. They can also update the field details and delete the field. It has also included a filter function to filter the list by field name and set announcements for the selected field.

The third scope is task management includes a task module that allows the farmer to view all the tasks, and assign tasks to the other farmer. This module also included notification features to remind the assigned farmer to complete their task as well as notify them if they have overdue tasks to remind them to complete the tasks as soon as possible. After completing the task, the farmer can click on a checkbox to update their task status from pending to completed. The farmer can also click on the task dashboard to view task information.

The fourth scope is inventory management including the inventory module which is used for the farmer to check the actual amount and estimated amount of the products stored in the inventory. This module includes the export of inventory list information into a CSV file, filtering the list by product category and CRUD operations features.

The fifth scope is pest management for the farmer to key in the pests that have been found in the field to aware others of solving the issues as well as preventing the pest that is harmful to the plant and its growth. This module allows the farmer to conduct CRUD operations to manage all the pest issues that occur in every field. The farmer can also upload the image of the pest they captured from the field, and key in the related information. When the issue is handled, the farmer can check on the checkbox to update the treatment status to notify the others that the issue is solved.

The sixth scope is resources which includes 2 modules which are "seeds" and "fertilizers". Both module that other than including the CRUD operations function, both modules also include the replenish resources function to allow the farmer to increase the resource amount. The purpose of this function is to ensure the farmers have sufficient fertilizers and seeds to plant the agricultural products to prevent the risk of a shortage of agricultural products. These modules can also allow the farmers to key in the number of packs of fertilizers or seeds that are being used and update the resource amount.

The seventh scope is a message including a message module. This module is implemented to be convenient for the farmer to communicate with others and share some information through this application.

The last scope is the notification including the announcement module. All the announcements made from the field sub-module will be displayed in this module and only the farmer who created the announcement can update and delete it. Not only can create the announcement from the sub-module, but can also create it in this announcement module.

1.4 Contributions

This web application, "Agriculture Product Supply Chain Management Application" can simplify farming operations and improve efficiency. Digitalizing farming operations allows the farmers to manage their farming business effectively and reduces their workload as the majority of the manual work has been eliminated. For instance, this web application will store and update the information in a database where the farmer can view all the latest information and prevent any data loss. Furthermore, this web application has also assisted the farmer in calculating the period of the agricultural product growth stage; thus, the farmer does not need to observe the plant and determine the growth stage manually and can know the current stage quickly and accurately. In short, by using this web application, the farmers were able to find that there is a significant improvement in their farming operation efficiency.

1.5 Report Organization

In this report, it will cover into 6 chapters. Chapter 1 is Introduction and 5 sections have been covered which are problem statement and motivation, project objective, project scope, contributions, and report organization. Chapter 2 is Literature Review which is a summarization of 3 reviewed existing systems and these 3 reviewed systems will be differentiated into 3 sub-sections of review existing system section. Chapter 3 is system methodology or approaches which includes the section, system design diagram with 3 sub-sections which are the system architecture diagram, use case diagram, and activity diagram to interpret this web application flow. Chapter 4 is about system design and it includes entity relationship diagram (ERD), block diagram, and user interface design (UID). Chapter 5 is system implementation which covers 3 sections, software setup, setting configuration and implementation issues and challenges. Chapter 6 is system evaluation. This chapter will be done with a system testing section and a project challenge section. The last chapter is Chapter 7, Conclusion to conclude the whole project development.

Chapter 2
Literature Review

2.1 Review Existing System

2.1.1 iAgri

iAgri is a web application software for farmers to manage and operate farming processes and business efficiency to deliver agricultural products to market and consumers on time. This software allowed the farmers to manage crop management, business financial statements, labor management, inventory management, etc. These have enabled farmers to maximize farming efficiency by reducing the labor workforce and cost of production. This software has reduced the paperwork for the farmer to manage farm information as the farmers can record all the information and data regarding the farm into this software. This function is convenient for the farmers to keep track of all the records for all time and also to prevent the loss of farming records and other related data. This software has been integrated with various services including social, mapping, cropping, financial, animal management, field map, labor management, hazard and incidents, farmer training, fleet, and task management [6]. Thus, the farmers can control their farming operations smoothly.

This software has also provided beneficial features for the farmer to manage their farm. In this software, the farmer can record the amount that they received and used with a few taps of finger; then, this software will calculate and update the latest financial statement analysis immediately. Emphasising financial statement analysis in farm management is crucial for the farmers to analyze current farming performance to improve farm production and operation [7]. Besides, this software provides a map-dragging feature for the farmer to drag a field for planting. It is convenient for the farmer to plot the field area without affecting the other's field as the farmer can use a different colour to drag the area to differentiate it. Moreover, this software allows one to manage the other's farmer account data such as updating the farmer's details and registering an account for the farmer. Last, the farmers can assign tasks to the other farmers to maximise the farmer's productivity and to track completion status.

This software has provided several strengths for the farmer to operate. This software used satellite maps for the farmer to drag their fields. This satellite map can determine the soil condition and whether it is suitable for planting. This can prevent the farmer plant in unhealthy soil and causing environmental waste. Besides, the farmer can also import their map into the

software to replace the original map in the software; however, the dragged field areas will still appear in the imported map. Moreover, the farmer can use this software without an internet connection to manage their farm operations. Furthermore, it includes analysis of cropping records in every field to allow the farmer to determine the production efficiency of the field they observed. The farmers can also export the analysis into PDF, Excel, CSV, etc to convenient them to track multiple cropping analyses in one file.

Even though this software has several strengths for the farmers, it has also had some weaknesses when the farmer uses it to manage the farm. This software will not assist the farmers in determining the agricultural product stage; hence, the farmers are required to observe the products manually to determine and estimate the current stage of the agricultural product in frequency. Moreover, this software is unable to estimate the number of agricultural products harvested by month and their stock amount in the inventory system where it is difficult for the farmer to plan for product delivery due to the farmer can only track the current stock amount. On the other hand, this software is unable to calculate the task due date based on the selected period and task frequency automatically and assign all the tasks to the farmer at once. Hence, the farmers are required to assign the tasks one by one which might cause the farmer to miss the task if the task assigned on that day is due on that day. Worse still, the farmer might forget to assign the task to the other farmers, and might cause some impact on the agricultural product growth.

2.1.2 Conservis

Conservis is a farming management software for every farmer to collaborate to manage and operate their farming business smoothly. This software has covered a wide range of services for farmers to improve their production productivity and efficiency. The services include work planning, labor communication channels, task management, inventory management, yielding analysis, and financial analysis [8]. These services provided in the software have simplified farmers' working procedures as all the work can be accomplished at once by using this software as well as reduced labor workforce. Besides, this software has included a data-collection function for the farmer to record the farming data and keep track of real-time cropping and yielding information for all time.

This software provided various useful features to assist the farmers in managing their farm operations. This software provides automation of financial services for the farmer to record all the financial data into the software and generate financial statements for the farmers to review

current yielding performance and plan for future production. Other than generating financial statements, this software will be based on the latest cropping data to determine the budget for farming operations and financial planning to enable the farmer to improve their yielding performance in the future. Furthermore, the farmer can keep track of the real-time data of the farms for all time to ensure operation stability and productivity. In addition, this software allows the farmers to upload receipts or lists of the products they wish to add to the system. Thus, the farmer does not need to enter it manually. Moreover, this software allows the farmer to use different colours to drag field areas to differentiate the field from the others.

This software has also benefited the farmers while using it to manage their farming business as it can analyse the farming and yield data to determine the cost of production. This can prevent the farmers from underestimating or overestimating the cost of production and causing a loss of profitability. Furthermore, this software can be used without an internet connection and out of the farming range without any inaccuracy of data as all the data will be synchronised. Additionally, this software will generate a real-time analysis report to observe and reflect the performance of the crop yield data of the field. This allows the farmers based on the analysis to improve the production performance in the next production. While operating the farms, the software will synchronise the rate of resource usage at all times to ensure that the farmer uses the resource wisely.

Besides the benefits of using this software, it has also had some weaknesses existed in this software. This software is unable to allow the users to key in the estimated day of each cycle stage for the different agricultural products. Thus, this software is unable to calculate and determine the current stage of the agriculture of products for the farmers. Furthermore, to fulfil the market and consumer demand for consuming fresh agricultural products, the farmer is required to estimate the stock amount including the amount of the product to be harvested. Unfortunately, this software does not include this inventory estimation function for the farmer to estimate the amount of the agricultural product harvested and its total amount able to be delivered and schedule the delivery period to deliver the products into the market. Lastly, the task module of this software does not include the task period and frequency of task selection to calculate the due date of the selected task and assign the tasks at once. This may be inconvenient for the farmers to determine the due date for the tasks and assign them one by one.

2.1.3 365 Crops

365 Crops is a farming software developed by 365 FarmNet for farmers to manage their farming business operations smoothly and cost-efficiently. This software provides a data-collection function for the farmer to key in farming data into the software to prevent the loss of records as well as ease for the farmer to track the previous and real-time farming data. This software includes 6 internal components such as access right, crop view, application map, iso-xml for creating dynamic maps, active document, and field planner. The other 16 components from external parties have been integrated into this software to allow the farmer to manage to enhance their operation efficiently and productivity [5].

This software provides several features for the farmer to manage their farm. Since this software is supported by multiple connections with the other stakeholders or farmers; the farmer can drag and plot a land either for personal or the stakeholder. This feature is useful for enhancing the efficiency of farming activities as the farmer can keep track of the stakeholder's land real-time data and observe the condition of the plant. Moreover, the farmer can also trace other's farm cropping status to analyse the yielding data able to be generated by using this software. The farmer can base this analysis on providing feedback to the farmer to improve the crop yielding. Furthermore, this software allows the farmers to overview the status of agriculture product production in either personal or other farmer's fields to analyse the product growing stage and condition used to estimate the harvesting date and total yielding of the product after harvesting.

This software includes various strengths for the farmer to use this software to manage the farm. This software has provided an auto-fill function where the software is based on previously saved data to fill up the related info into the software. Besides, the farmers can use this software in offline conditions to manage their farm activities and track the real-time data of the farm. Additionally, this software includes a weather forecast feature that it uses to enhance the plan of farming activities to assign suitable tasks for the farmer to complete [9]. This software will according to the weather conditions stated in the weather forecast to suggest the farmer which activities are suitable to be conducted. These features are convenient for the farmer to key in the data by clicking the auto-fill selection if the data has been stored before and to assign suitable tasks to the farmer based on the weather conditions to ensure productivity.

On the other hand, the weaknesses of this software still exist which may cause the reduction of the farming efficiency. This software does not contain a module to key in the planting details of the agricultural products and the estimated days of each cycle stage. Due to the lack of this

module in this software, it is unable to calculate and determine the current cycle stage of the agricultural products. Moreover, this software is unable to display the location of the plotted land in the dashboard which might difficult for the farmers to find the land. Furthermore, this software does not provide stock estimation features and estimate the amount of the agricultural product harvested. This has caused them to raise their challenges in planning for supplying agricultural products to the market. Lastly, the task management module of this software allows the farmers to assign tasks for the other; however, it does not have the frequency of task and period selection for the farmer to calculate the due date of tasks when assigning the task for the farmer. This may cause the assigned farmer to put off the task reducing the farmer's efficiency and performance of agricultural product production.

Chapter 3

System Methodology/Approach

The methodology to be used to develop this agricultural product supply chain management application is Agile Methodology. It is a process of developing software or applications from planning, design, development, testing, and review. It provides more clarifications for the whole development process from the beginning of the process until the final product is developed. It is an iterative and incremental approach where the developers can return to the previous process to make changes and improve the parts in the middle of developing the applications. This approach can enhance the quality of the application.

This web application will be started on the planning by researching the current situation and reviewing for existing system to identify the issues required to be solved in the system. By conducting research and reviewing of existing system, all the information was transformed into a proposal to describe the planning of developing the system. Then, based on the system planning design a prototype for the web application and develop it. If there are any issues found in the prototype in the middle of developing the web application such as system logic, it is allowed to return to the design stage to design the prototype again to solve the issue. While completed development, the web application will go through a testing phase by using a web browser. If there is any issue found in the testing process, it is required to return to the development stage to analyse the code of the web application. After completing the testing phase, it is time to review and evaluate this web application to improve this application in the future.

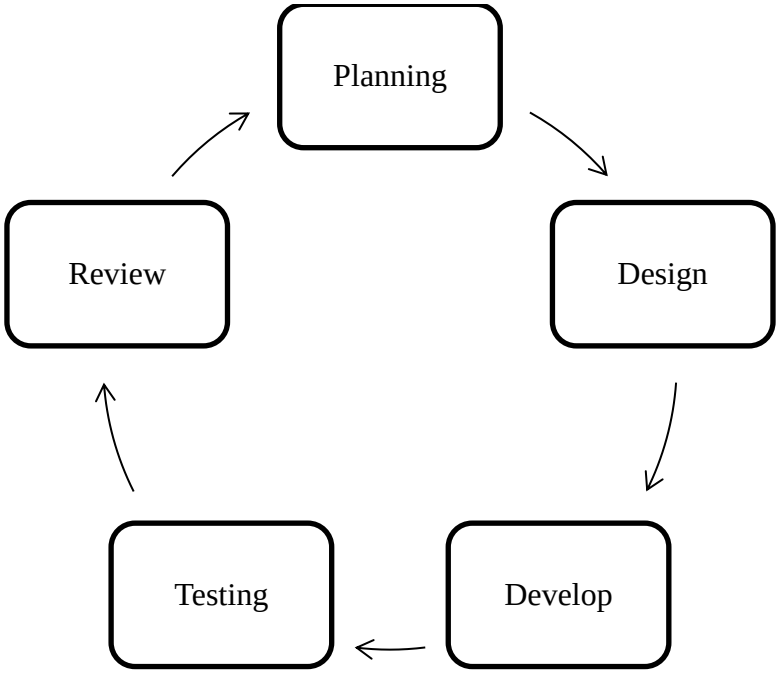


Figure 3.0 Process of Agile Methodology

3.1 System Design Diagram

3.1.1 System Architecture Diagram

Figure 3.1.1 System Architecture Diagram of "Agriculture Product SCM Application"

Figure 3.1.1 above shows the system architecture diagram of the "Agriculture Product SCM Application". This diagram shows the ways of this web application to interact with the users and servers. The visual interface of this web application is built by a hypertext markup language (HTML), cascading style sheet (CSS), and JavaScript. These will allow the farmer to manipulate and interact from the visual interface. This visual interface includes data visualisation, data key-in, etc.

For the backend, the ASP.NET framework is integrated with this web application. The web server will transfer the HTTPS request sent by the farmers to ASP.NET to process it. For instance, user verification, data processing, and so on. After processing the request, it will pass the result to the web server and respond to the interface.

The database is used to store all the data and farming records such as land data, inventory tasks, etc. The database server used to store and manage the data is the MySQL database server. By using a structured query language (SQL), the farmers can store, update, and retrieve data based on the farmers' requests and respond to the results on the webpage through the web server.

3.1.2 Use Case Diagram and Description

Use Case Diagram

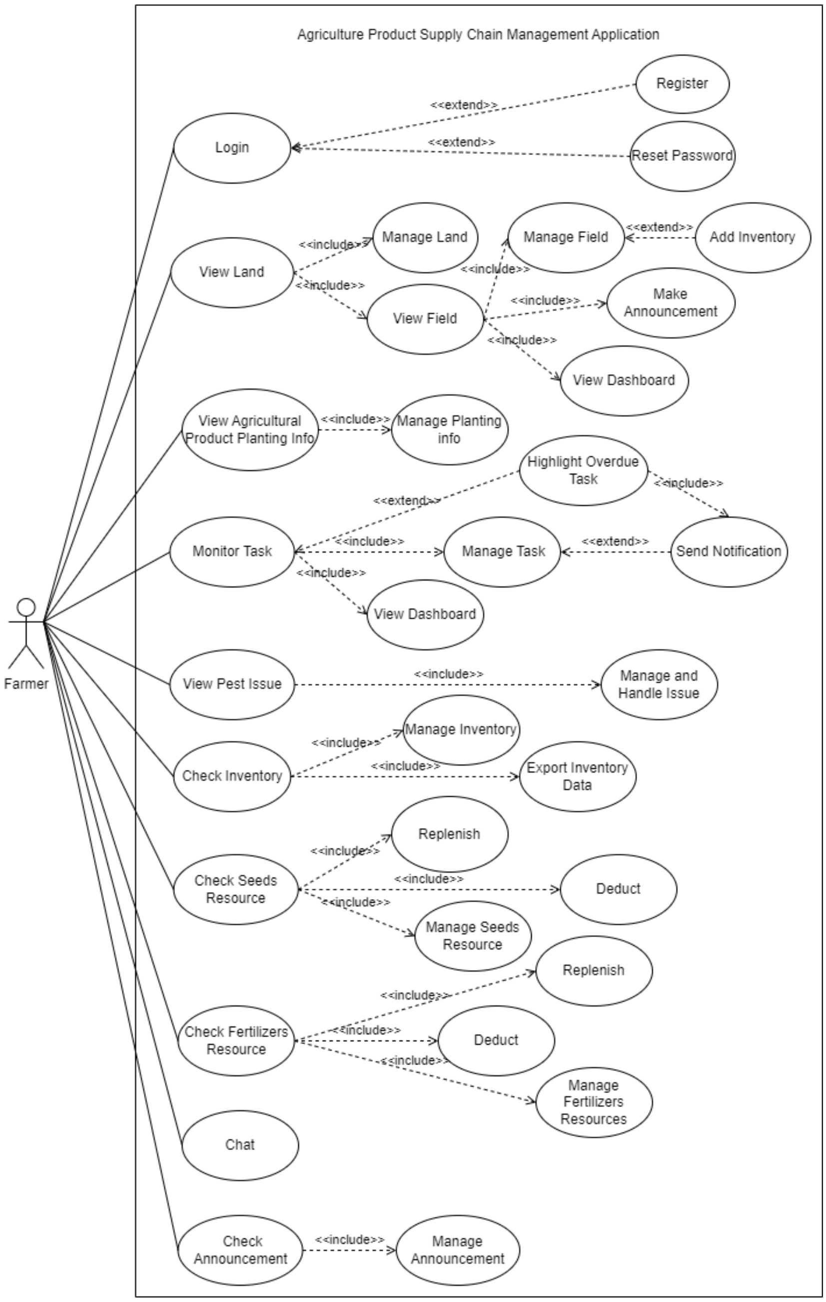


Figure 3.1.2 Use Case Diagram of "Agriculture Product SCM Application"

Use Case Description

Use Case Name: Login	ID: 1	Importance Level: High
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – Login into the web application to manage the farm		
Brief Description: To describe the ways for the farmer to log into the web application		
Trigger: Farmers enter their email and password Type: Internal		
Relationships: Association: Farmer Include: - Extend: Register, Reset Password Generalization: User Information		
Normal Flow of Events: 1. Farmer login into their web application account with their email and password 2. If the farmer has no account yet, execute the Register case 3. If the farmer wishes to change the password, execute the Reset password case		
Sub-Flows: -		
Alternate/Exceptional Flows: -		

Table 3.1.2.1 Use Case Description of Login Case

Use Case Name: View Land	ID: 2	Importance Level: High
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – Wants to view or manage the land and its land		
Brief Description: To describe how the farmer can view or manage the land and its field		
Trigger: Farmer viewed the land information or selected either add, update, delete land, or view and manage field Type: Internal		
Relationships: Association: Farmer Include: Manage Land, View Field Extend: Generalization: Manage Land, Manage Field		

Normal Flow of Events:

1. The farmer views all the created land lists.
2. The farmer selects add new land, update or delete existing land, or view and manage its fields.
    - If the farmer selects add new land, the manage land case is executed, and the S-1: add new land sub-flow is performed.
    - If the farmer selects update existing land, the manage land case is executed, and the S-2: update existing land sub-flow is performed.
    - If the farmer selects delete existing land, the manage land case is executed, and the S-3: delete existing land sub-flow is performed.
    - If the farmer selects to view and manage its field, the field case is executed.
    - If the farmer selects to add a new field, the manage field case is executed, and the S-4 add new field sub-flow is performed.
    - If the farmer selects to update the existing field, the manage field case is executed, and the S-5: update existing field sub-flow is performed.
    - If the farmer selects to delete the existing field, the manage field case is executed, and the S-6: delete existing field sub-flow is performed.
    - If the farmer wants to make an announcement, the farmer can select add announcement and execute its use case.
    - If the farmer wants detailed field information, the farmer can select and execute the view dashboard use case.

Sub-Flows:

S-1: Add New Land: The farmer enters the necessary details of the land.

S-2: Update Existing Land: Farmer re-enters and updates necessary details of the selected existing land.

S-3: Delete Existing Land: Farmer reviews and deletes selected existing land.

S-4: Add New Field for Selected Land: The Farmer enters the necessary details of the field. If the agricultural product planned to be planted in this field does not exist in inventory, will automatically be added into inventory.

S-5: Update Existing Field: Farmer re-enters and updates necessary details of the selected existing field. If new changed agricultural product planned to be planted in this field does not exist in the inventory, will automatically be added to the inventory.

S-6: Delete Existing Field: Farmer reviews and deletes selected existing fields.
Alternate/Exceptional Flows: -

Table 3.1.2.2 Use Case Description of View Land Case

Use Case Name: View Agricultural Product (Fruit/Vegetable) Planting Information	ID: 3	Importance Level: High
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – View and manage agricultural products planting info		
Brief Description: To describe how the farmer can manage the planting information by adding, updating, or deleting agricultural products.		
Trigger: The farmer can select to either add a new agricultural product with its planting information or update or delete the selected agricultural product. Type: Internal and External		
Relationships: Association: Farmer Include: Manage Planting Information Extend: - Generalization: Manage Agricultural Products Planting Info		
Normal Flow of Events: 1. Farmers can view all existing agricultural product planting information 2. Farmers select to add new agricultural products, or update or delete existing agricultural products. • If the farmer selects to add a new agricultural product, the manage planting info case is executed, and the S-1: add new agricultural product sub-flow is performed. • If the farmer selects to update the existing agricultural products, the manage planting info case is executed, and the S-2: update existing agricultural product sub-flow is performed. • If the farmer selects to delete the existing agricultural product, the manage planting info case is executed, and the S-3: delete existing agricultural product sub-flow is performed.		
Sub-Flows:		

S-1: Add New Agricultural Product: The farmer enters the necessary planting information for the agricultural product. S-2: Update Existing Agricultural Product: The farmer reviews and updates the selected existing agricultural product by re-entering the updated necessary information. S-3: Delete Existing Agricultural Product: The farmer reviews and deletes the selected existing agricultural product.
Alternate/Exceptional Flows: -

Table 3.1.2.3 Use Case Description of View Agricultural Product Planting Information

Use Case Name: Monitor Task	ID: 4	Importance Level: High
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – Wants to assign and manage the task to the other farmers		
Brief Description: To describe how the farmer monitors the task performance and status and manages every assignee		
Trigger: The farmer can monitor the performance of each task assign a new task or update or delete the assigned task. Type: Internal		
Relationships: Association: Farmer Include: manage task, view dashboard, send notification Extend: Send notification, highlight overdue task Generalization:		
Normal Flow of Events: 1. Farmers can view all assigned tasks and their status, and check overdue tasks. 2. Farmers can select to assign a new task, update assigned incomplete task details, update task status, or delete the assigned task. • If the farmer selects to assign a new task, the manage task case is executed, and the S-1: assign new task sub-flow is performed. • If the farmer selects to update task status for the selected task, the manage task case is executed, and the S-2: update task status sub-flow is performed.		

• If the farmer selects to update the assigned incomplete task, the manage task case is executed, and the S-3: update assigned incomplete task sub-flow is performed. • If the farmer selects to delete the assigned task, the manage task case is executed, and the S-4: delete assigned task sub-flow is performed. 3. If the farmer selects to view the dashboard, execute the view dashboard case.
Sub-Flows: S-1: Assign New Task: The farmer enter new task details to assign the task for the farmer and sends notification to alert the farmer. S-2: Update Task Status: The farmer checks on the task status checkbox to update the task status from "pending" to "completed" and remove the notification. The update button for the completed task will be unavailable. S-3: Update Incomplete Task: The farmer selects an incomplete task to edit the task details and update the notification if the assignee changes to another farmer. S-4: Delete Assigned Task: The farmer reviews and deletes the selected assigned task, and removes the notification if the deleted task is incomplete.
Alternate/Exceptional Flows: -

Table 3.1.2.4 Use Case Description of Monitor Task Case

Use Case Name: View Pest Issue	ID: 5	Importance Level: Medium
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – wants to observe and solve pest issues for each field		
Brief Description: To describe how the farmer manages to handle the pest issue that occurred in the field		
Trigger: Farmers observe all pest issues found enter new issues, or update or delete found issue Type: Internal		
Relationships: Association: Farmer Include: Manage and handle Issue Extend: - Generalization: Handle Pest Issue		
Normal Flow of Events:		

1. Farmers observe all the issues found in every field. 2. Farmer selects to add new captured pest issue, update treatment status, update untreated issue details, or delete issue. • If the farmer selects to add a new captured pest issue, the manage and handle issue case is executed, and the S-1: add new issue sub-flow is performed. • If the farmer selected to update selected untreated pest issue information, manage and handle issue case is executed, the S-2: update selected untreated issue sub-flow is performed. • If the farmer selected to update the treatment status for the selected issue, manage and handle issue case is executed, the S-3: update treatment status sub-flow is performed. • If the farmer selects to delete the selected issue, manage and handle issue case is executed, and the S-4: delete selected pest issue sub-flow is performed.
Sub-Flows: S-1: Add New Pests Issue: The farmer enters the information related to the pest. S-2: Update Selected Untreated Pest Issue: Farmer re-enters the updated details of the selected issue. S-3: Update Pest Treatment Status: The farmer checks the checkbox on the treatment status column for the newly handled pest issue. The update button of the treated issue will be unavailable. S-4: Delete Selected Pest Issue: The farmer reviews and deletes the pest's issue.
Alternate/Exceptional Flows: -

Figure 3.1.2.5 Use Case Description of View Pest Issue Case

Use Case Name: Check Inventory	ID: 6	Importance Level: High
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – Wants to check or manage the agricultural product in the inventory		
Brief Description: To describe how the farmer can check, add, update, or delete the agricultural product in the inventory and export inventory data.		

Trigger: The farmer views the inventory stock and selects an action either to add new agricultural product stock, update, delete existing agricultural products, or export inventory data into a CSV file. Type: Internal
Relationships: Association: Farmer Include: Manage inventory, Export CSV file Extend: - Generalization: Manage Inventory
Normal Flow of Events: 1. Farmers view all the agricultural product stock in inventory. 2. Farmers can choose to add new agricultural products to their inventory or update or delete agricultural products in inventory. • If the farmer selects to add a new agricultural product in inventory, the manage inventory case is executed, and the S-1: add new agricultural product in inventory sub-flow is performed. • If the farmer selects to update the existing agricultural product in inventory, manage inventory case is executed, and the S-2: update existing agricultural product in inventory sub-flow is performed. • If the farmer selects to delete an existing agricultural product in inventory, manage inventory case is executed, and the S-3: delete existing agricultural product from inventory sub-flow is performed. 3. If the farmer decides to export the CSV file of the inventory data. The export inventory data case is executed.
Sub-Flows: S-1: Add New Agricultural Product in Inventory: The farmer asked to enter the product name, and upload the product image, quantity, and price. S-2: Update Existing Agricultural Product in Inventory: The farmer asked to update any necessary product details in inventory for the selected existing product wish to update. S-3: Delete Existing Agricultural Product from Inventory: The farmer reviews and deletes the selected existing agricultural product from inventory.
Alternate/Exceptional Flows: -

Table 3.1.2.6 Use Case Description of Check Inventory Case

<table>
<tr><td>Use Case Name: Check Seeds Resource</td><td>ID: 7</td><td>Importance Level: Low</td></tr>
<tr><td>Primary Actor: Farmer</td><td colspan="2">Use Case Type: Detail, Essential</td></tr>
<tr><td colspan="3">Stakeholders and Interests: The farmer – wants to manage the seed resource</td></tr>
<tr><td colspan="3">Brief Description: To describe how the farmer can ensure the sufficiency of the seeds for planting agricultural products.</td></tr>
<tr><td colspan="3">Trigger: The farmer views all available seeds, and decides either to add new seeds update delete existing seed details, or replenish or deduct selected seeds stock quantity.
Type: Internal</td></tr>
<tr><td colspan="3">Relationships:
Association: Farmer
Include: Manage seed resource, replenish, deduct
Extend: -
Generalization: Seeds Resources</td></tr>
<tr><td colspan="3">Normal Flow of Events:
1. Farmers view all the seeds of agricultural products.
2. The farmer selects the action to add new seeds, update existing seed details, delete existing seeds, or replenish or deduct seed stock quantity.
• If the farmer selects to add a new seed into resources, the manage seeds resource case is executed, and the S-1: add new seeds sub-flow is performed.
• If the farmer selects to update the existing seed’s details, the manage seeds resource case is executed, and the S-2: update existing seeds sub-flow is performed.
• If the farmer selects to delete existing seeds from resources, the manage seeds resource case is executed, and the S-3: delete existing seeds sub-flow is performed.
• If the farmer selects to replenish the stock quantity of seeds in resources, the replenishment case is executed.
• If the farmer selects to deduct the quantity of seeds in resources, the deduct case is executed.</td></tr>
<tr><td colspan="3">Sub-Flows:
S-1: Add New Seeds: The farmer entered the necessary information about the seeds</td></tr>
</table>

S-2: Update Existing Seeds: The farmer re-entered and updated the details of the existing seed in resource assets. S-3: Delete Existing Seeds: The farmer reviews and deletes the selected seeds from the resource assets.
Alternate/Exceptional Flows: -

Table 3.1.2.7 Use Case Description of Check Seeds Case

Use Case Name: Check Fertilizers Resource	ID: 8	Importance Level: Low
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: The farmer – wants to manage the fertilizers resource		
Brief Description: To describe how the farmer can ensure the sufficiency of the fertilizers for cultivating agricultural products.		
Trigger: Farmer views all available fertilizers, and decides either to add new fertilizers, update, delete existing fertilizers details, or replenish or deduct selected fertilizers stock quantity. Type: Internal		
Relationships: Association: Farmer Include: Manage fertilizers resource, replenish, deduct Extend: - Generalization: Fertilizers Resource		
Normal Flow of Events: 1. Farmers view all the fertilizers have been stored in the resource assets. 2. Farmers select the action to add new seeds, update existing fertilizer details, delete existing fertilizers, or replenish or deduct fertilizer stock quantity. • If the farmer selects to add new fertilizers into resources, the manage fertilizers resource case is executed, and the S-1: add new fertilizers sub-flow is performed. • If the farmer selects to update the existing fertilizer's details, the manage seeds resource case is executed, and the S-2: update existing fertilizers sub-flow is performed.		

• If the farmer selects to delete existing fertilizers from resources, the manage fertilizers resource case is executed, and the S-3: delete existing fertilizers sub-flow is performed. • If the farmer selects to replenish the stock quantity of fertilizers in resources, the replenishment case is executed. • If the farmer selects to deduct the quantity of fertilizers in resources, the deduct case is executed.
Sub-Flows: S-1: Add New Seeds: The farmer entered the necessary information about the fertilizers S-2: Update Existing Seeds: The farmer re-entered and updated the details of the existing fertilizers in resource assets. S-3: Delete Existing Seeds: The farmer reviews and deletes the selected fertilizers from the resource assets.
Alternate/Exceptional Flows: -

Table 3.1.2.8 Use Case Description of Check Fertilizers Case

Use Case Name: Chat	ID: 9	Importance Level: Low
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – wants to check messages or chat with other farmers		
Brief Description: To describe how farmers can chat with others using this web application		
Trigger: The farmer selects a farmer to check message or chat Type: Internal		
Relationships: Association: Farmer Include: - Extend: - Generalization: Chat		
Normal Flow of Events: 1. Farmers select the farmer that he or she wants to chat with. 2. Read the unread message sent by the farmer. 3. Continue the chat with the farmer.		
Sub-Flows: -		
Alternate/Exceptional Flows: -		

Table 3.1.2.9 Use Case Description of Chat Case

Use Case Name: Check Announcement	ID: 10	Importance Level: Low
Primary Actor: Farmer	Use Case Type: Detail, Essential	
Stakeholders and Interests: Farmer – wants to check or manage the announcement		
Brief Description: To describe the farmer to check, make, update, or delete announcement		
Trigger: Farmer views all announcements, makes announcements, updates announcement content, or deletes announcements. Type: Internal		
Relationships: Association: Farmer Include: Manage announcement Extend: - Generalization: Make an announcement		
Normal Flow of Events: 1. Farmers check all the announcements created by themselves or by others. 2. Farmers can decide to make new announcements, update announcement content, or delete announcements. • If the farmer chooses to make a new announcement, manage announcement case is executed, and the S-1: make new announcement sub-flow is performed. • If the farmer chooses to update announcement content, manage announcement case is executed, and the S-2: update announcement content sub-flow is performed. • If the farmer chooses to delete the announcement, manage announcement case is executed, and the S-3: delete announcement sub-flow is performed.		
Sub-Flows: S-1: Make New Announcement: The farmer asked to enter the new announcement. S-2: Update Announcement Content: The farmer asked to enter updated announcement content into the created announcement. S-3: Delete Announcement: The farmer deleted the selected announcement from the announcement list.		
Alternate/Exceptional Flows: -		

Table 3.1.2.10 Use Case Description of Check Announcement Case

3.1.3 Activity Diagram

User Authentication Module Activity Diagram

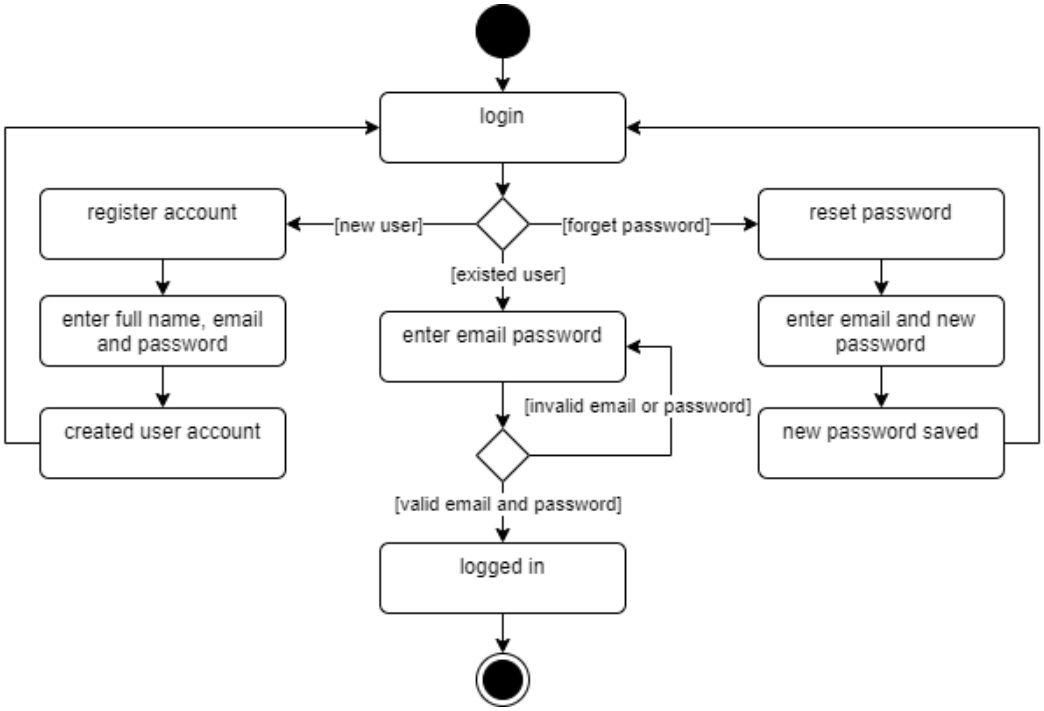


Figure 3.1.3.1 User Authentication Activity Diagram

Figure 3.1.3.1 above shows the user authentication module activity diagram. This module begins from the login page by default, the farmer then can log in by key in email and address. The web application will validate the user by comparing the key in the email and hashed password with the email and hashed password stored in the database. If the email and password are valid, the farmer will successfully log into the web application; otherwise, the web application will show an error message and either ask the farmer to re-enter the email and password or register an account if the account does not exist. Once the farmer registers an account, it will redirect the farmer to the login page and enter a valid email and address to log into the web application. The farmers can also reset their password if they forget their password or wish to change the password. The farmer is required to key in their email and new password for the application and update their password in the database.

Land Module Activity Diagram

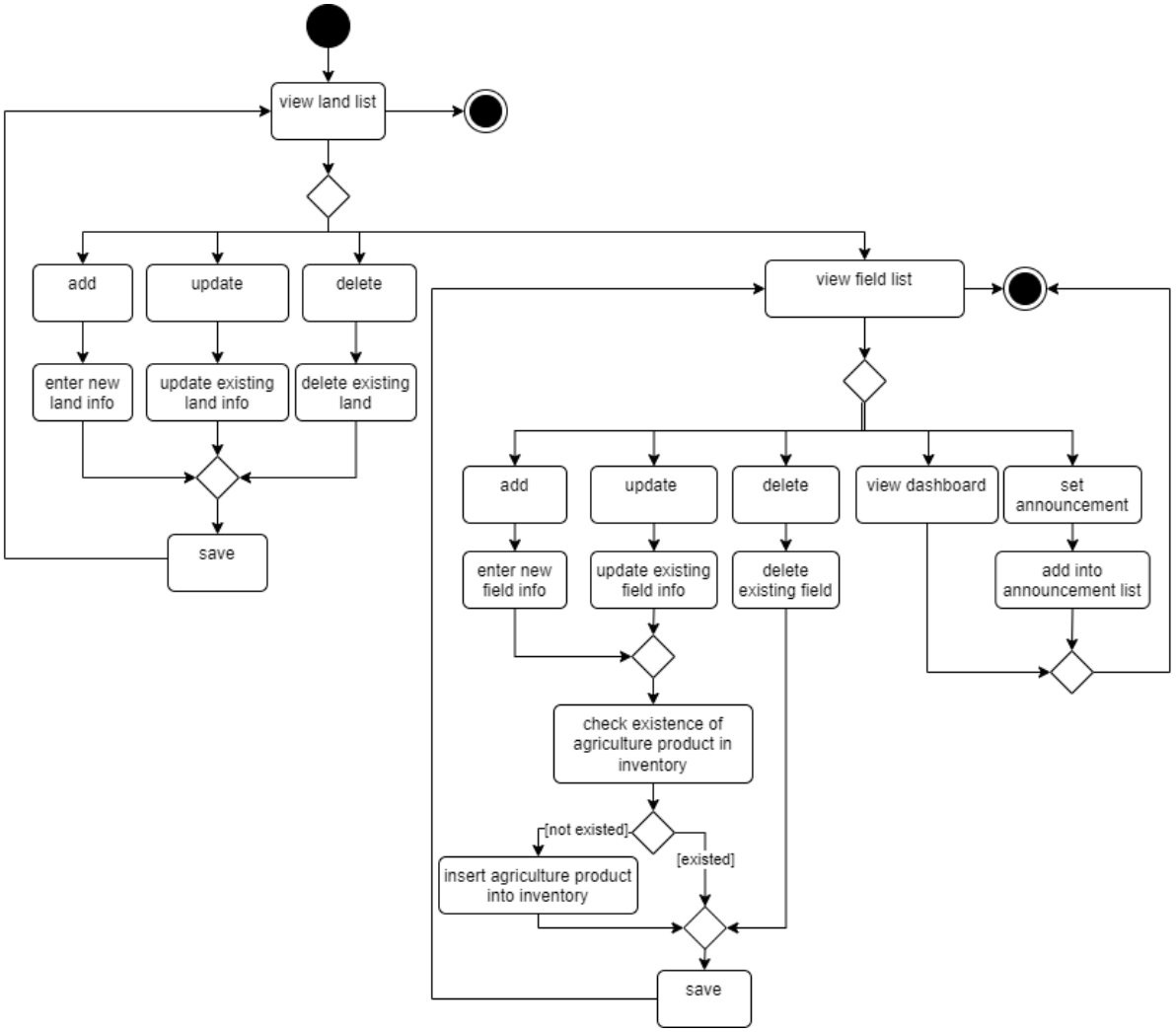


Figure 3.1.3.2 Land Module Activity Diagram

Figure 3.1.3.2 above shows the land module activity diagram. In this land module, the web application initially will show the main page of the land module which is the list of lands for the farmer. The farmer can then choose to add new land information, update and delete selected land information. The application will redirect the farmer to each respective page to add, update, or delete the land. After saving the land information, the farmer will be redirected back to the main page of the land module.

Besides, it has also included a sub-module which is the field module; thus, the activity diagram of the field module will be included in the land module. The farmer can view the field that is under the selected land by selecting the "field" button to access the field module. The main page of this field module is a list of fields recorded in a table. After being redirected to this main page, the farmer can decide to add new field information, or update or delete selected field information. The application will redirect the farmer to each respective page to add, update, or delete the field. When the farmer chooses to add a new field or update an existing

field, this application will check the agricultural product that exists in the inventory. If it exists in inventory the application will directly save the field; otherwise, the application will add it into the inventory. This is convenient for farmers to enter agricultural products into inventory automatically. After completing any CRUD action, the farmer will be redirected back to the field webpage with updated information. The farmer can also view each of the field dashboards for more detailed information related to the field to be convenient for them for decision-making to plan for the activities. Other than that, the farmer can also set an announcement for the selected field to notify the other regarding the agricultural product growth stage. It is also useful to remind the farmers who managing this field to monitor it.

Fruit or Vegetable Planting Data Module Activity Diagram

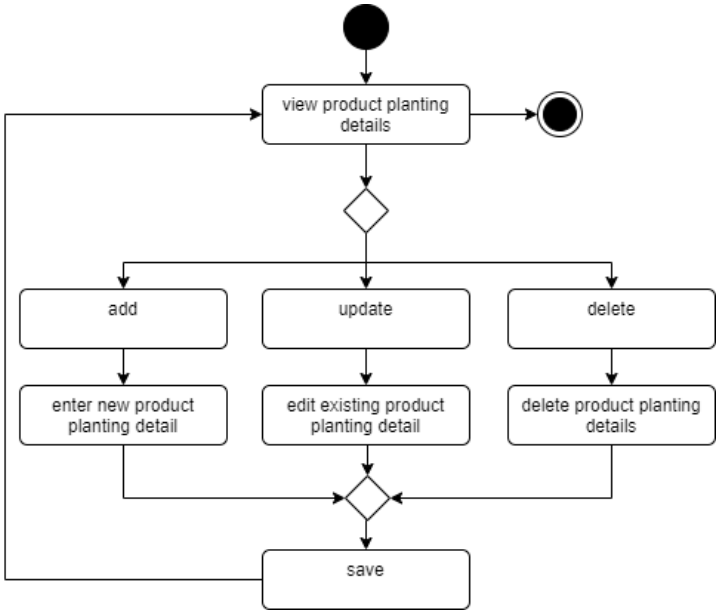


Figure 3.1.3.3 Fruit or Vegetable Planting Data Module Activity Diagram

Figure 3.1.3.3 above shows the activity diagram of the fruit or vegetable planting data module. When the farmer selects this module, the farmer will be redirected to the main page of this module and it will list all of the planting information for every agricultural product. The farmer can add new agricultural products to the list if the product does not exist. The farmer can also select which product needs to be updated or deleted from the list. The application will redirect the farmer to each respective page to add, update, or delete the planting information of the agricultural product. After the data is saved and updated in the database, the farmer will be redirected back to the updated product list webpage.

Task Module Activity Diagram

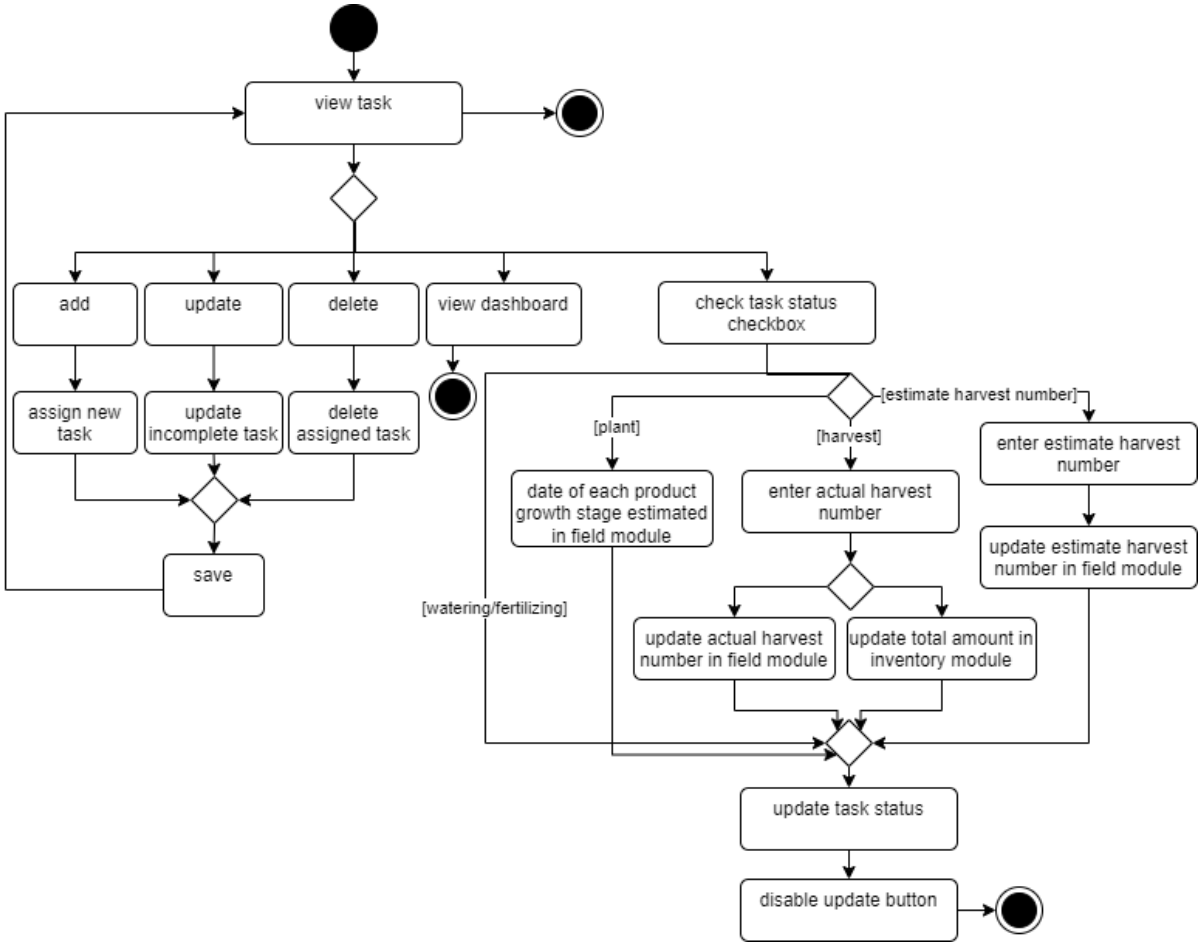


Figure 3.1.3.4 Task Module Activity Diagram

Figure 3.1.3.4 above shows the activity diagram of the task module to interpret the flow of how the farmer can manage the task. The farmer initially will be redirected to the task main page and view all of the tasks that have been assigned. The farmer can assign new tasks by selecting the add function and entering every required information. The farmer can also update the information of the selected task if the task is incomplete, and also delete the task. The application will redirect the farmer to each respective page to add, update, or delete the task. After saving the information in the database, it will redirect the farmer back to the main page of the task module. The farmer can also update the task status when they complete the task by checking on the checkbox. If the task type of the completed task is harvest, the application will open a modal for the farmer to enter the actual harvested number of the agricultural products. Meanwhile, it will also update the total quantity in the inventory automatically and update the actual number in the respective field. If the task type is estimating harvest number, the application will open a modal to ask the farmer to enter the estimated number. After entered, it will update the estimated harvest number in the respective field as well as the product estimation in the inventory. The date of the product growth stage will be estimated in the field

module if the task type is “plant”. Once the checkbox is checked, the farmer is not allowed to update the completed task due to the button being disabled. Lastly, the farmer is allowed to view the dashboard of the task to view the visualized task information.

Pest Management Module Activity Diagram

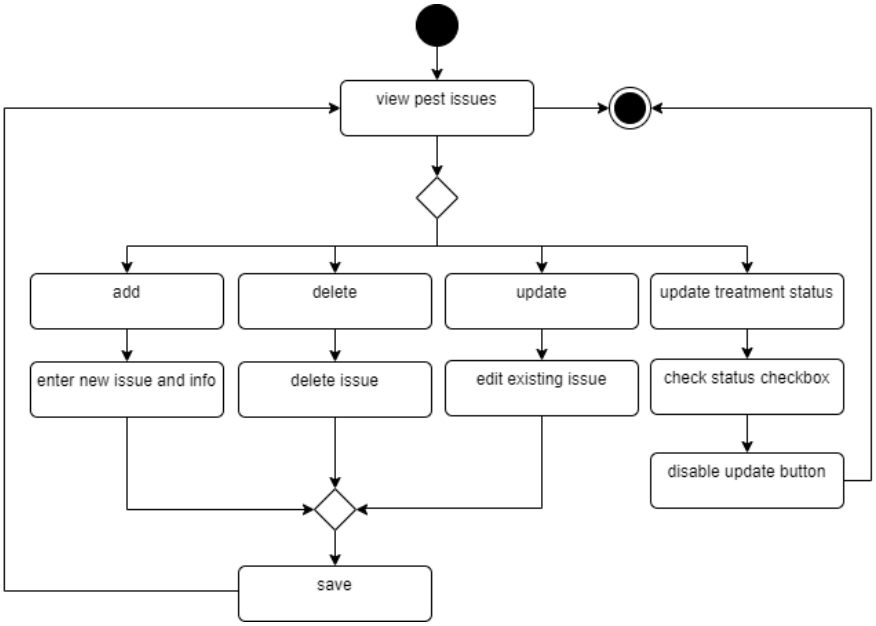


Figure 3.1.3.5 Pest Management Module Activity Diagram

Figure 3.1.3.5 above shows the activity diagram of the pest management module for the farmer to update and handle the issue of pests in the field. The application will initially redirect the farmer to the main page of pest management and it will display all the issues in a table. Once the farmer has successfully handled the issue of the pest listed in the table, the checkbox of field treatment status should be checked as “treated” and once checked and the farmer is not able to update the information of the handled issue as the update button for the issue is disabled. Furthermore, the farmer can also add new issues, update untreated issues, and delete issues. The application will redirect the farmer to each respective page to add, update, or delete information on pest issues. Once completed adding, updating, or deleting, the application will redirect the farmer back to the main page of the pest management module.

Inventory Management Module Activity Diagram

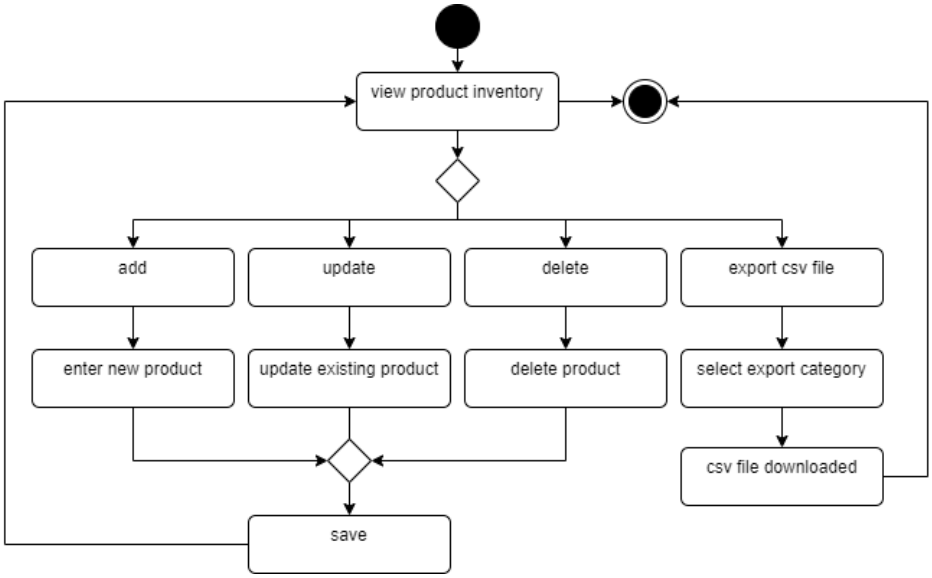


Figure 3.1.3.6 Inventory Management Module Activity Diagram

Figure 3.1.3.6 above shows an activity diagram of the inventory module for the farmer to manage their agricultural products in the inventory. When a farmer redirects to the inventory module, it will show all the agricultural products in a table with its information on its main page. In this module, the farmer can add new agricultural products, and update and delete selected existing agricultural products. The application will redirect the farmer to each respective page to add, update, or delete the product. After completing to add, update, or delete of the product, the application will redirect back to the main page of the inventory module. The farmer can also export the inventory of agricultural products based on its category. This may be a useful function for the farmer to take stock when facing internet connection issues.

Seeds Module Activity Diagram

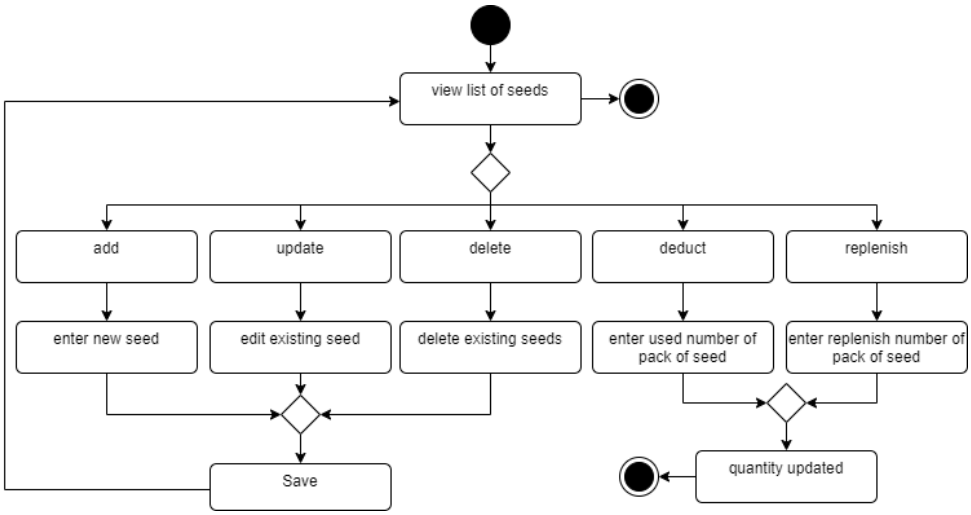


Figure 3.1.3.7 Seeds Module Activity Diagram

Figure 3.1.3.7 shows the activity diagram of the seeds module which is a flow for the farmer to manage it in resource assets. The farmer will first redirect to its main page which will show all the existing seeds of every agricultural product. The farmer can add new seeds of agricultural product, and update or delete existing seeds. The application will redirect the farmer to each respective page to add, update, or delete the seeds. After adding, updating, or deleting the seeds, the application will be redirected back to the main page of the seeds module. Besides, the farmer can also select any seeds to replenish the seeds or vice versa. This is convenient for the farmer to update the quantity of the seeds without recalculating the quantity. Since replenish and deduct quantity will pop out a modal unlike create update and delete seeds, after the quantity is entered and the modal, it will remain on the main page and end the activity.

Fertilizers Module Activity Diagram

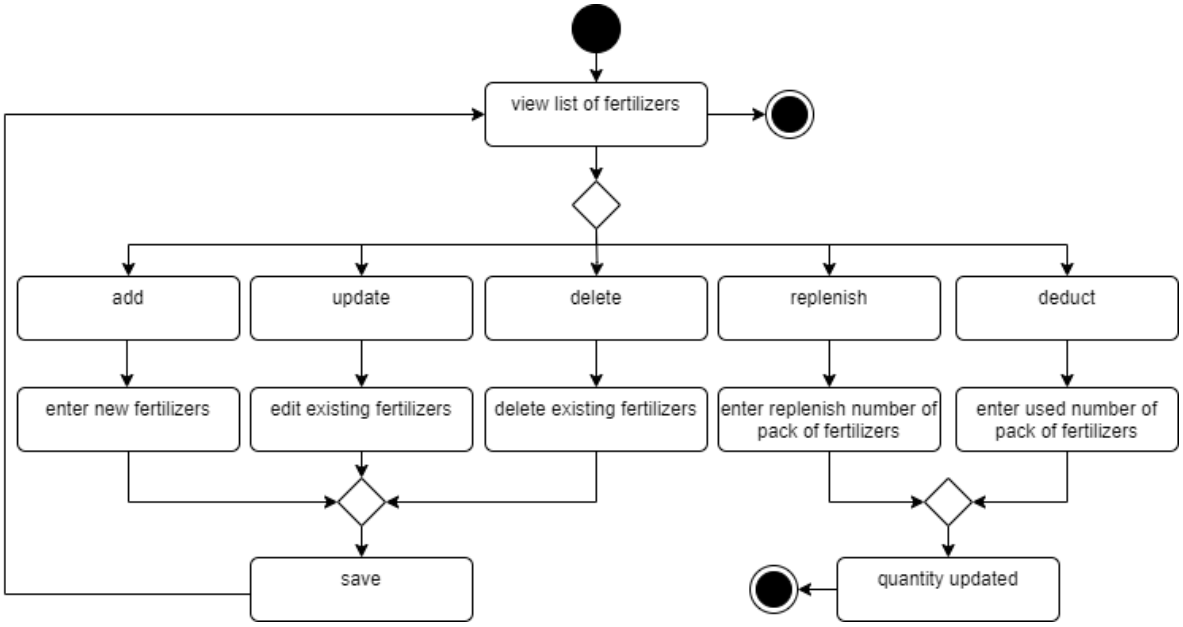


Figure 3.1.3.8 Fertilizers Module Activity Diagram

Figure 3.1.3.8 shows the activity diagram of the fertilizers module which is a flow for the farmer to manage it in resource assets. As usual, the farmer will be firstly redirected to the main page of the fertilizers module to view all the fertilizers in a table. The farmer can select to add new fertilizers, update existing fertilizers, or delete fertilizers. By choosing any of these 3 actions, the application will redirect the farmer to each respective page to add, update, or delete the fertilizers. After completion, the application will redirect back to the main page of the fertilizers module. For replenishing and updating fertilizers, the application will pop out a modal for the farmer to enter the quantity of pack of fertilizers required to replenish or deduct.

Since it will not be redirected to the other pages, it will end the activity after the quantity is updated.

Announcement Module Activity

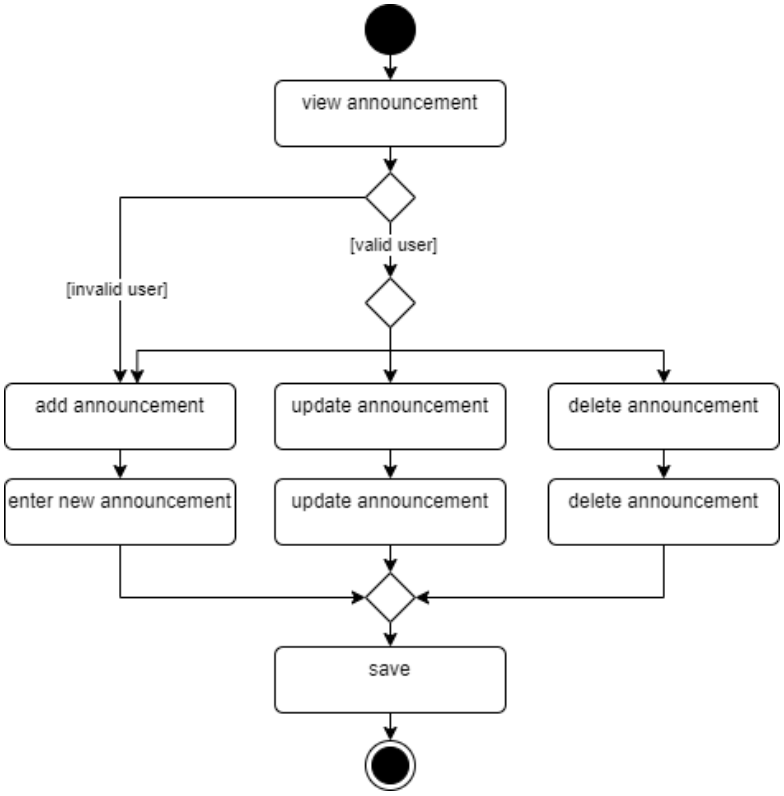


Figure 3.1.3.9 Announcement Module Activity Diagram

Figure 3.1.3.9 above shows the activity diagram of the announcement module for this application. This module only allows the farmer who created the announcement to update and delete it. Otherwise, the other farmer can only read the announcement. However, every farmer is allowed to make new announcements. The application will open a modal for the farmer to make, update, or delete announcements. After completed and saved, the modal will close and end the activity.

Chat Module Activity Diagram

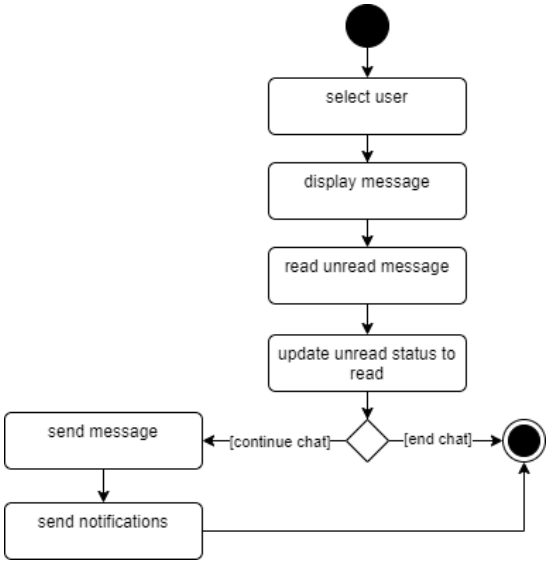


Figure 3.1.3.10 Chat Module Activity Diagram

Figure 3.1.3.10 above shows the activity diagram of the chat module for this application. When the farmer has selected on chat module, the application will redirect the farmer to the page of the chat module. The farmer is required to select one of the conversations in the user list. The application will then display all of their conversations. When the unread message is viewed, the application will update the status. If the farmer wishes to continue the conversation, the farmer can send a message to the selected farmer and the application will send notifications to the respective farmer. Otherwise, the farmer can end with the chat activity.

Chapter 4

System Design

4.1 Entity Relationship Diagram (ERD)

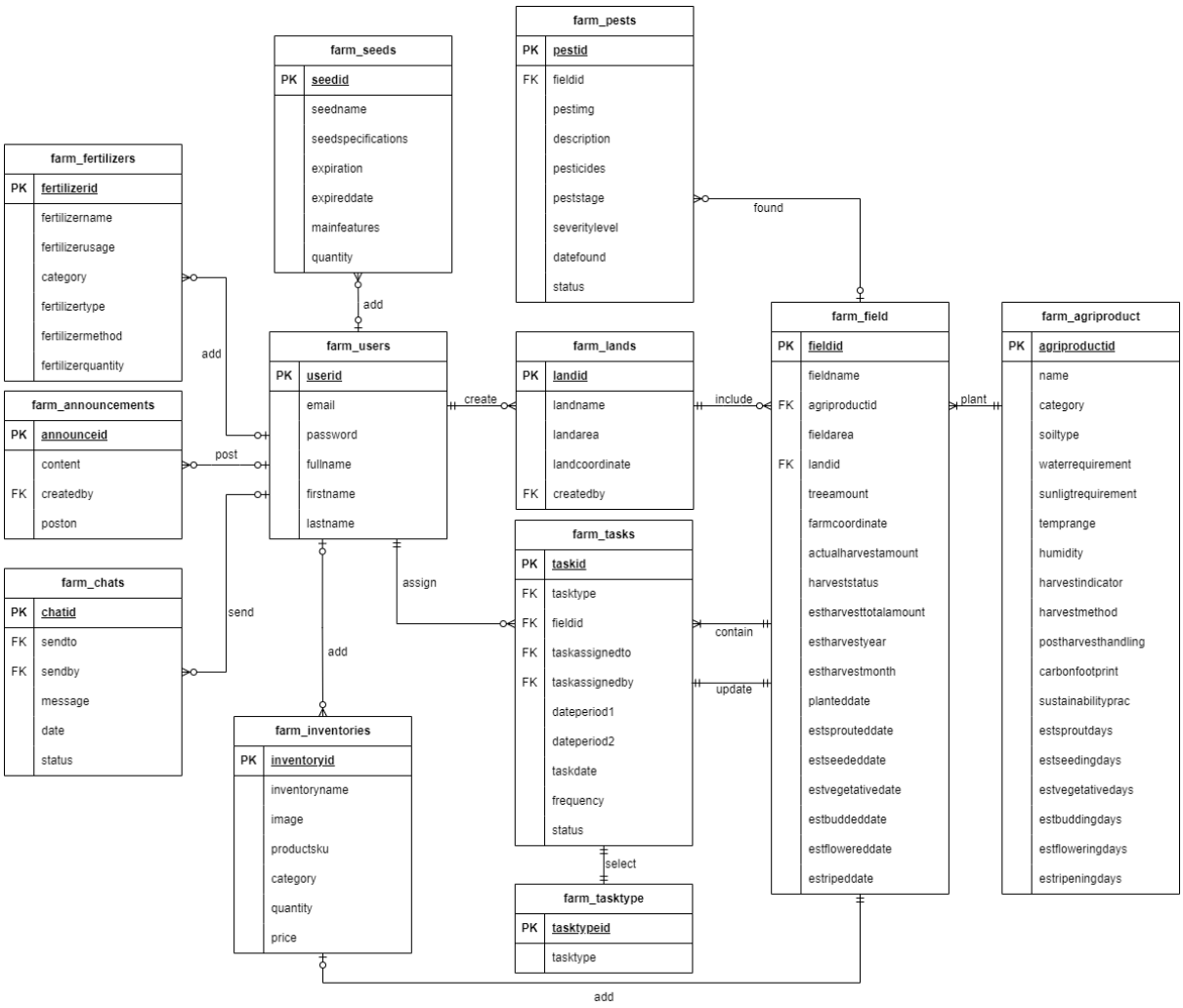


Figure 4.1 Entity Relationship Diagram (ERD)

Figure 4.1 above shows the entity relationship diagram (ERD) of this "Agriculture Product Supply Chain Management Application". Each of the entities has at least one and above relationship with other entities. Before using this application, every farmer is required to have a user account as it is a main entity to extend the relationship with other entities. After creating a user account, all their required user information will be stored in the "farm_users" entity.

From Figure 4.1 above, the ERD shows that every "farm_users" can optionally create more than one "farm_lands". After creating a land the farmer can plot their fields inside the created land. Thus, each of the "farm_land" is allowed to include either more than one "farm_field".

To begin the agricultural product production, the farmer can select one of the agricultural products stored in the “farm_agriproduct” entity to plant in the “farm_field”. While plotting the “farm_field” to cultivate the agricultural product, the application will check the existence of the agriculture product in “farm_inventories”. If it does not exist, it will be added to “farm_inventories”.

Besides, assigning tasks is important for the farmer to cultivate the agricultural product in “farm_field”. Thus, each “farm_users” can assign more than one “farm_tasks” and each “farm_task” can only select one “farm_tasktype” to the farmer. Every “fam_field” may contain several “farm_tasks” and every time completed “farm_task”, it will update the information of “farm_field” based on “farm_tasktype”. When the actual harvest number of the agricultural product is updated in “farm_field”, it will also add quantity in the “farm_inventories”. Other than adding the agricultural product and its quantity automatically from the “farm_field”, the”farm_users” can add the agricultural product into “farm_inventories” also. Moreover, from the “farm_field”, the farmer may either have found no or more occurrences of pest issues and record the issue in the “farm_pests” entity.

To ensure sufficient resources to cultivate the agricultural product, the “farm_users” can add seeds of agricultural products in the “farm_seeds” entity and several fertilizers in the “farm_fertilizers” entity. Furthermore, the “farm_users” can post multiple “farm_announcements” to notify them of the latest news. Lastly, the “farm_users” can communicate in this application with the others by sending “farm_chat”.

4.2 Block Diagram

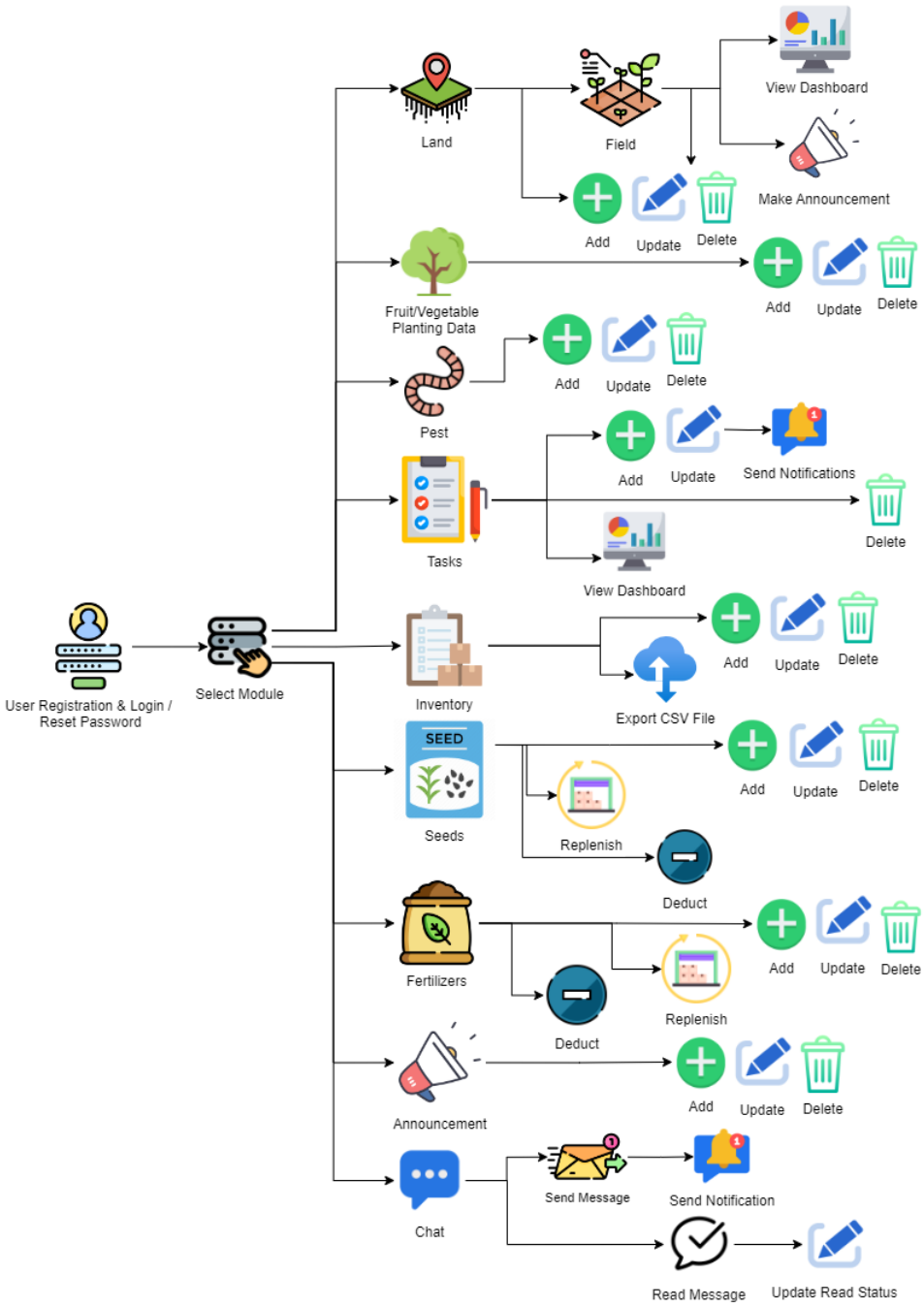


Figure 4.2 Block Diagram

Figure 4.2 above shows the block diagram of this "Agriculture Product Supply Chain Management Application". The block diagram shows after the farmer creates an account and logs in to the web application, the farmer can select a module. The "land" module includes the add, update, and delete functions, and a sub-module which is the "field" module. This "field" module can also allow the farmer to either add, update, or delete the field as well as to view the dashboard and make an announcement of the selected field.

For “fruit/vegetable planting data” module is for recording the planting information for every agricultural product that the farmer currently cultivating. This module allows the farmer to add new agriculture products, update existing product information, or delete information. The “pest” module, is a module for the farmer to record, manage, and handle the issue of pests that occurred in the field. The farmer can add a new issue, update details of the issue or treatment status, or delete the issue.

In the “tasks” module, the farmer can assign a new task for the farmer to complete and also update the task information if the task is incomplete or update the task status from pending to completed. After being assigned or updated, the web application will send a notification to the farmer who has been assigned to complete the task. The farmer can also delete the selected task if they wish to delete it. This module also provided a dashboard to visualize the task information to be convenient for the farmer to analyze the overall assigned task performance.

For the “Inventory” module, the farmer can add new agricultural products to this module to manage its available quantity and its estimated quantity. The farmer can also update and delete the selected agricultural product. Not only to add, update, and delete function included in this module, but also allows the farmer to export the CSV file for agricultural product data by category.

Moreover, the farmer can also add and manage their farming resources to ensure sufficient resource allocation for agricultural product production. The farmer can manage the resources in the “seeds” module and “fertilizers” module. Both of the modules allow the farmer to add new resources, and update or delete the selected resource. If the quantity of seeds or fertilizers is low, the farmer can select to replenish them to increase the quantity of the resources. The farmer can also select the deduct function to enter the quantity of the resources and deduct it from the module.

For the “announcement” module, the farmer can make any announcement to notify all the farmers of any latest updated information or news. This can allow the farmer to determine and modify their decision-planning on the agricultural product production on time. The farmer can also update and delete the announcement that is created by themselves.

Lastly, for the “chat” module, the farmer can communicate with others using this web application. When the message is sent, the web application will send notifications to the farmer. Once the farmer opens the conversation and reads the message, the message status will be updated from “unread” to “read”.

4.3 User Interface Design

User interface design (UID) has defined the ways how this web application can build an interaction with the farmer. In addition, UID is used to maximize the usability and the user experience of software or hardware. A user-friendly UID allows the farmer to interact with the whole web application easily and improve the efficiency of managing the farming operations.

UID of Login Page

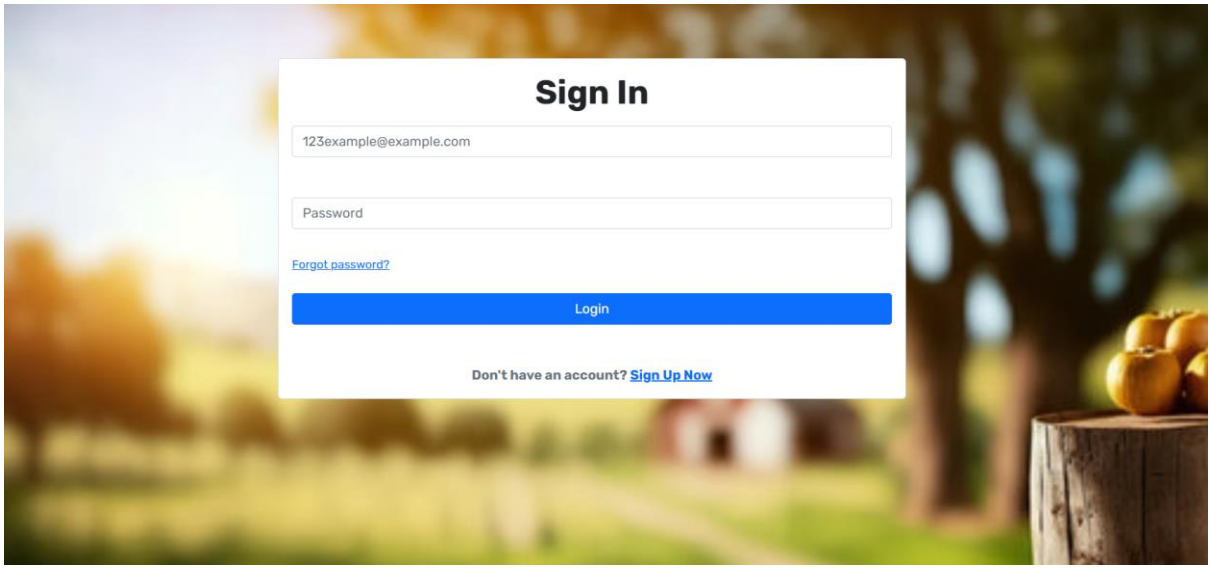


Figure 4.3.1 UID of Login Page

UID of Register Page

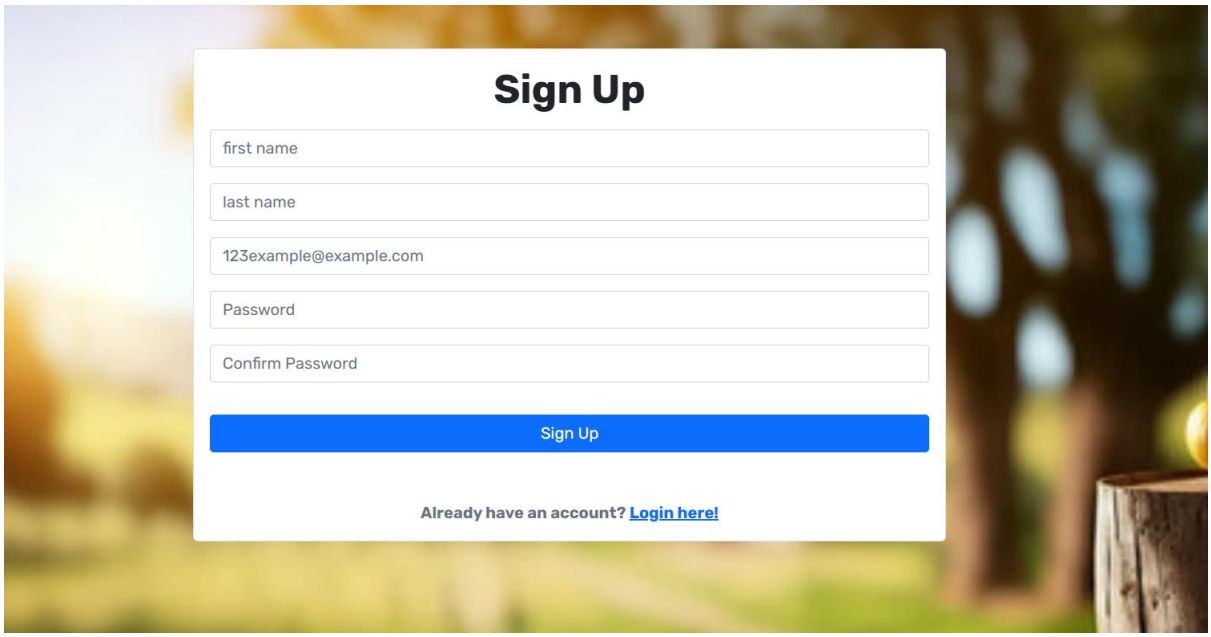


Figure 4.3.2 UID of Signup Page

UID of Change Password Page

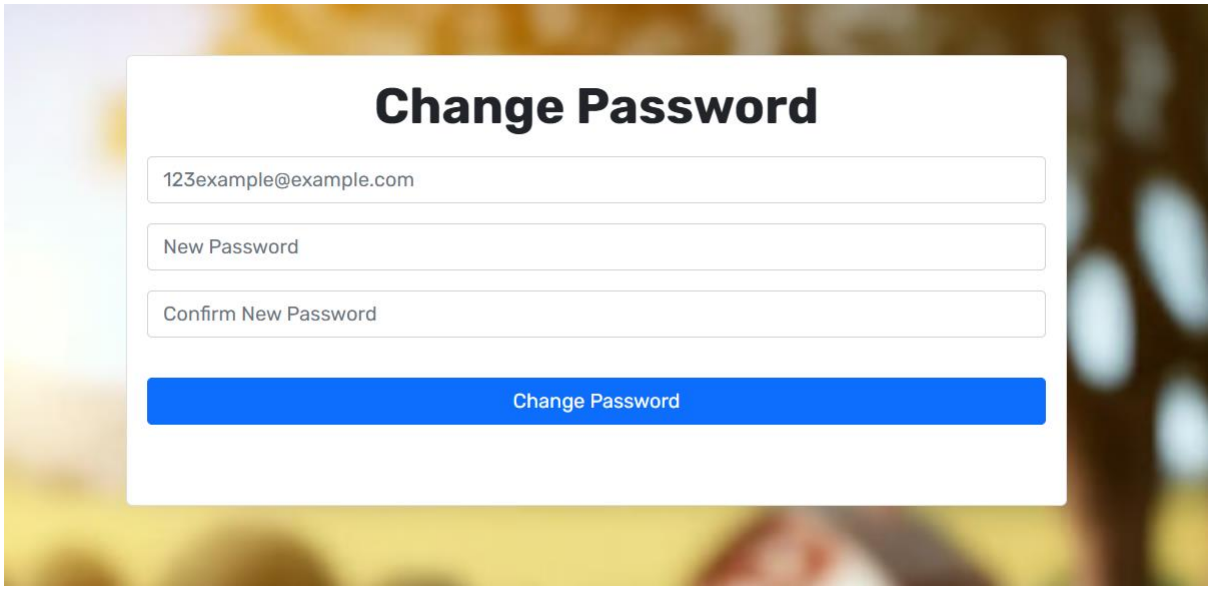


Figure 4.3.3 UID of Change Password Page

UID of Homepage

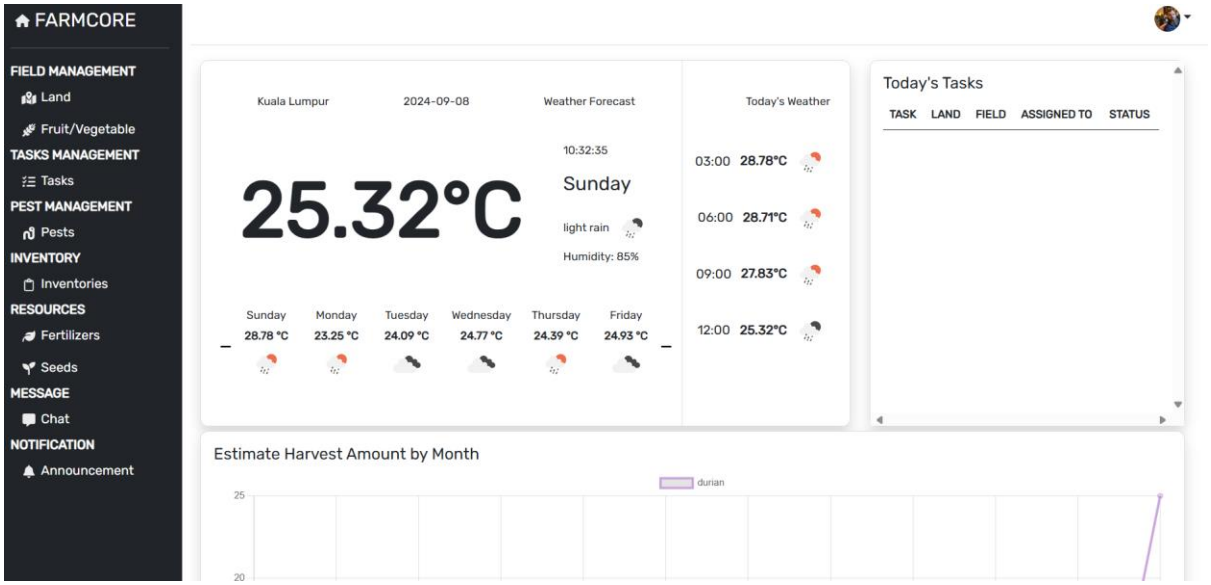
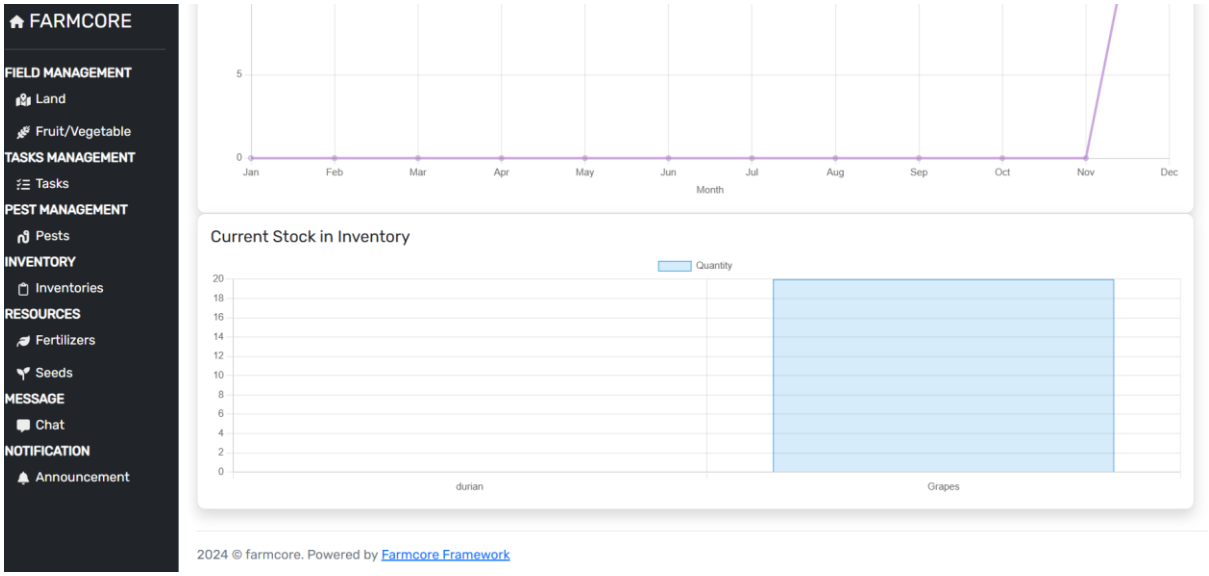


Figure 4.3.4 UID of Homepage



(cont.) Figure 4.3.4 UID of Homepage

UID of Fruit or Vegetable Planting Data Page

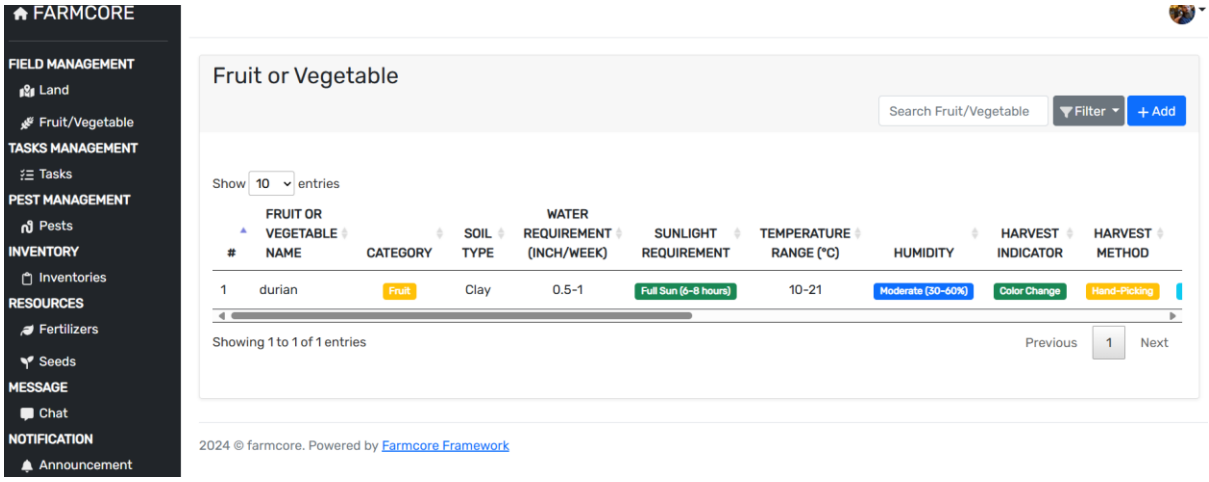


Figure 4.3.5 UID of Fruit or Vegetable Planting Data Page

UID of Add New Fruit or Vegetable Planting Data Page

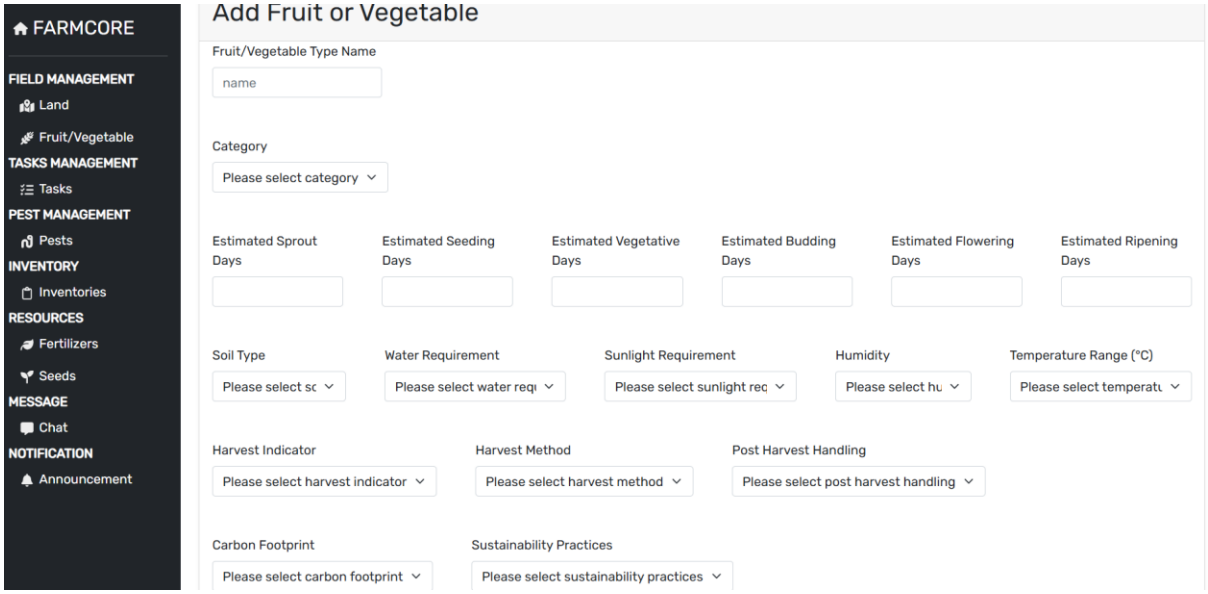


Figure 4.3.6 UID of Add New Fruit or Vegetable Planting Data

UID of Update Fruit or Vegetable Planting Data

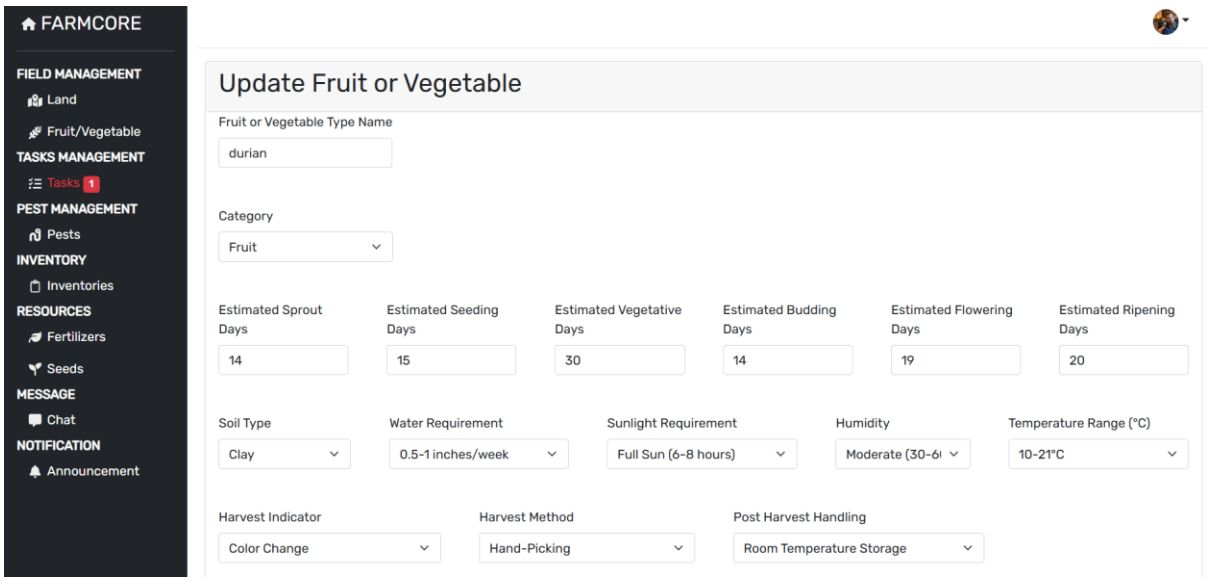


Figure 4.3.7 UID of Update Fruit or Vegetable Planting Data

UID of Delete Fruit or Vegetable Planting Data

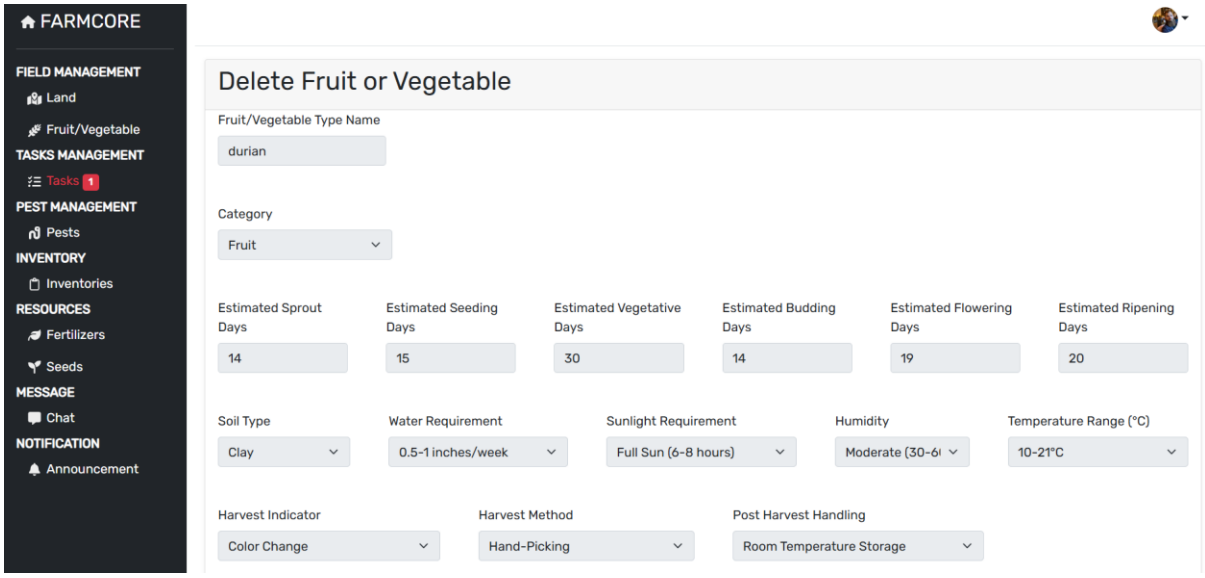


Figure 4.3.8 UID of Delete Fruit or Vegetable Planting Data

UID of Land Data Page

Figure 4.3.9 UID of Land Data Page

UID of Add New Land Page

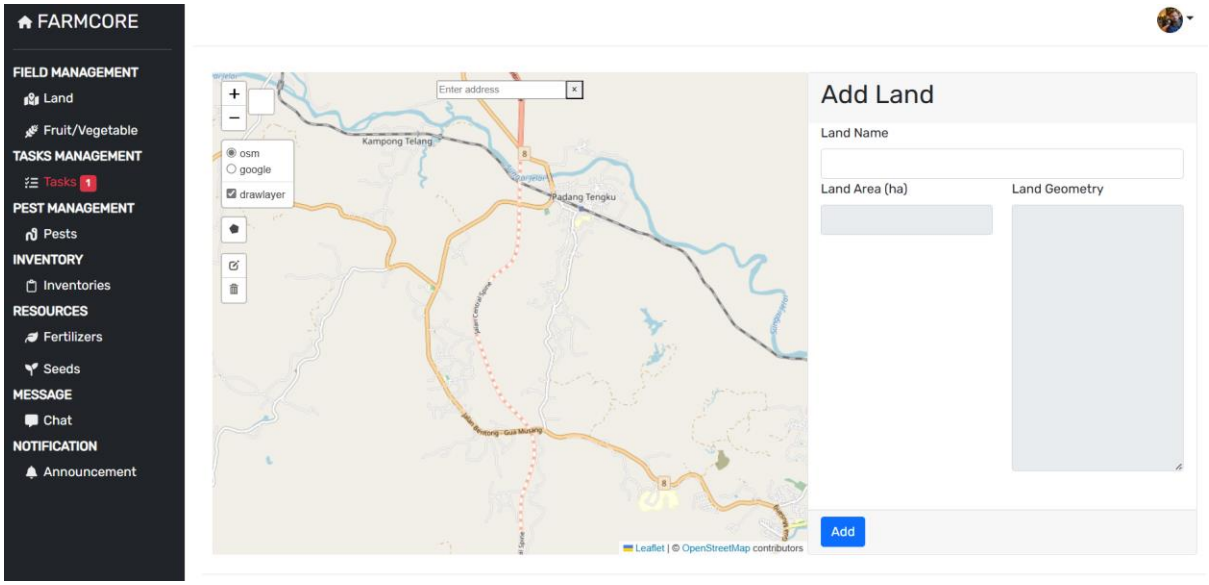


Figure 4.3.10 UID of Add New Land Page

UID of Update Land Page

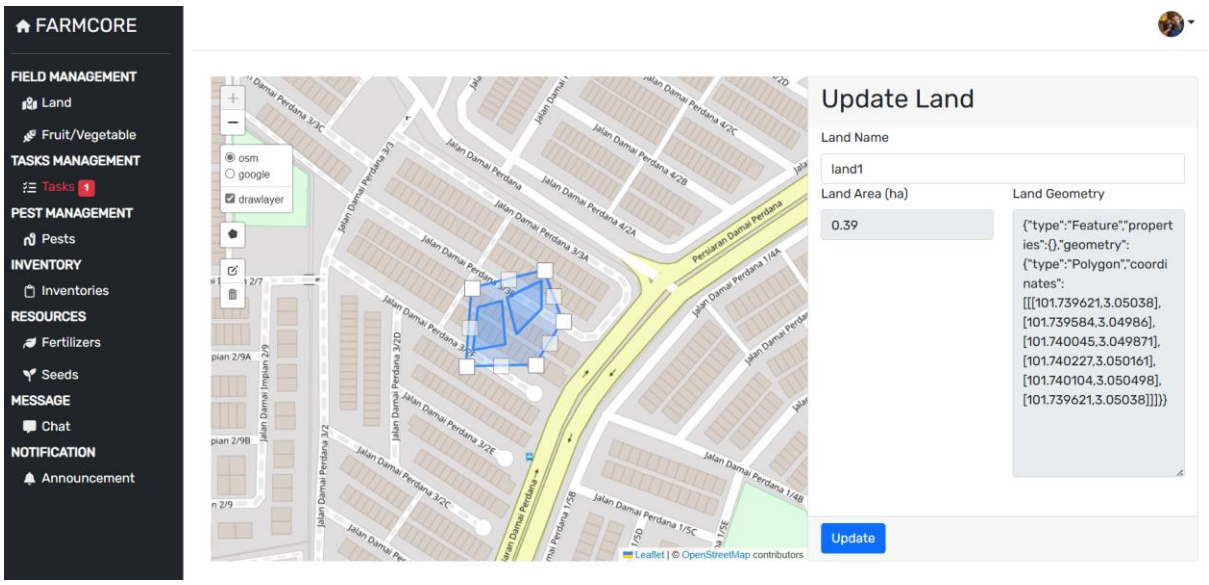


Figure 4.3.11 UID of Update Land Page

UID of Delete Land Page

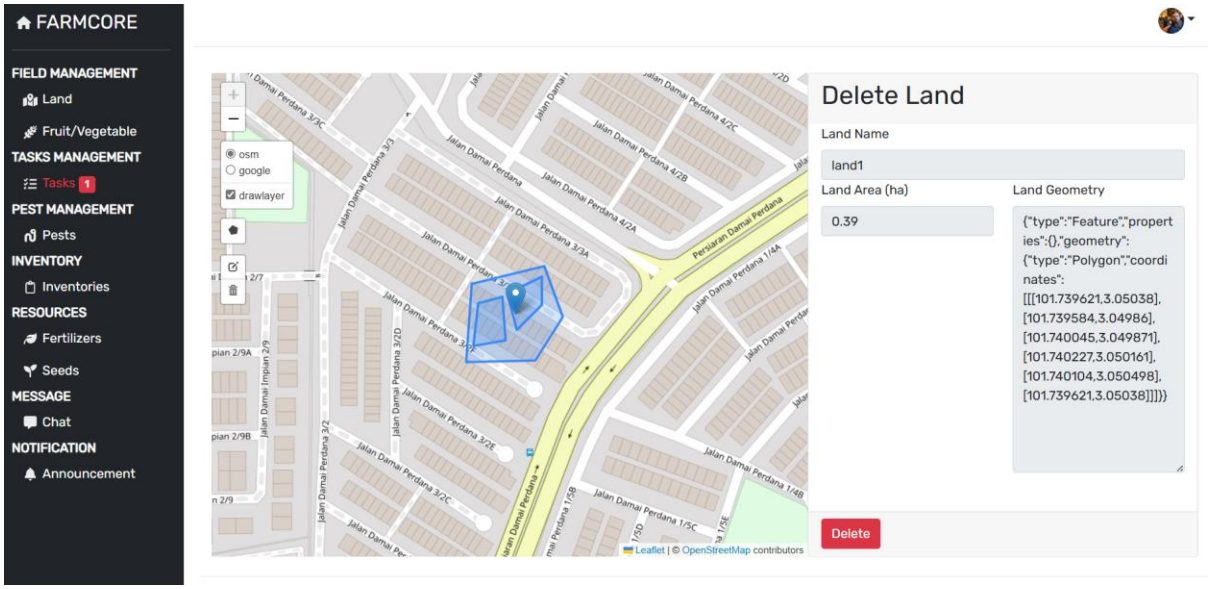


Figure 4.3.12 UID of Delete Land Page

UID of Field Data Page

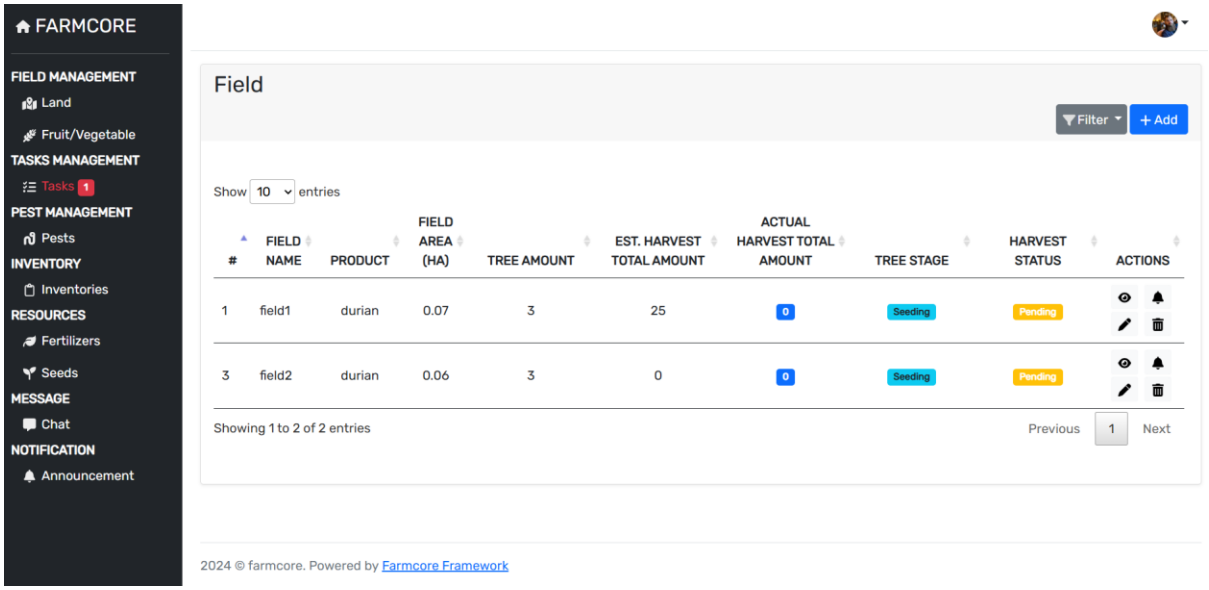


Figure 4.3.13 UID of Field Data Page

UID of Add New Field Page

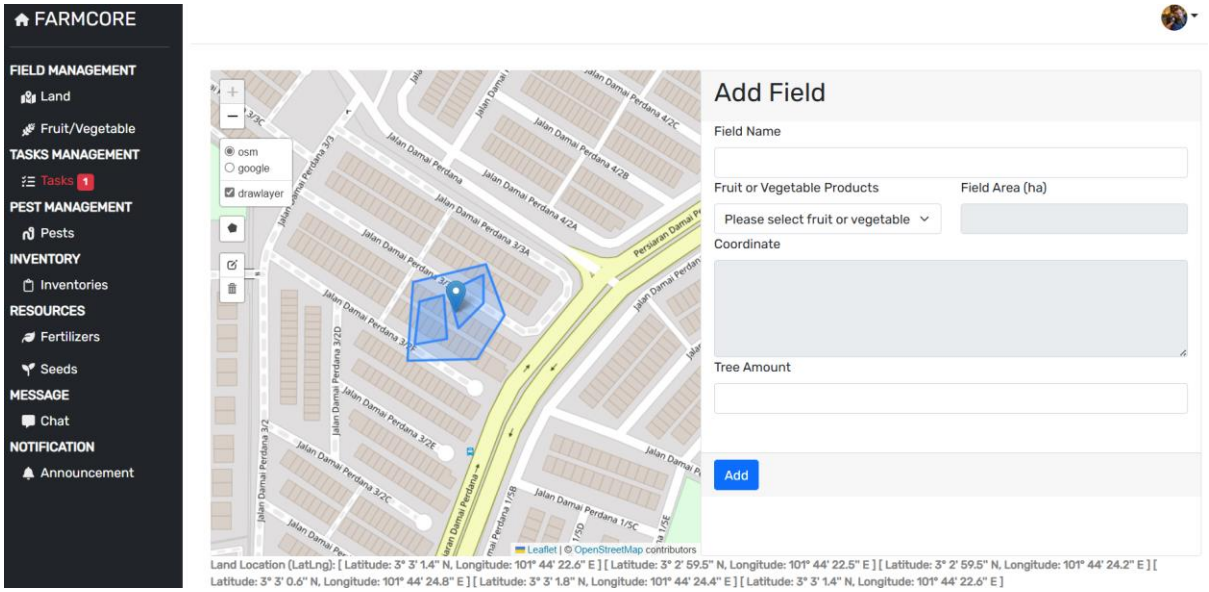


Figure 4.3.14 UID of Add New Field Page

UID of Update Field Page

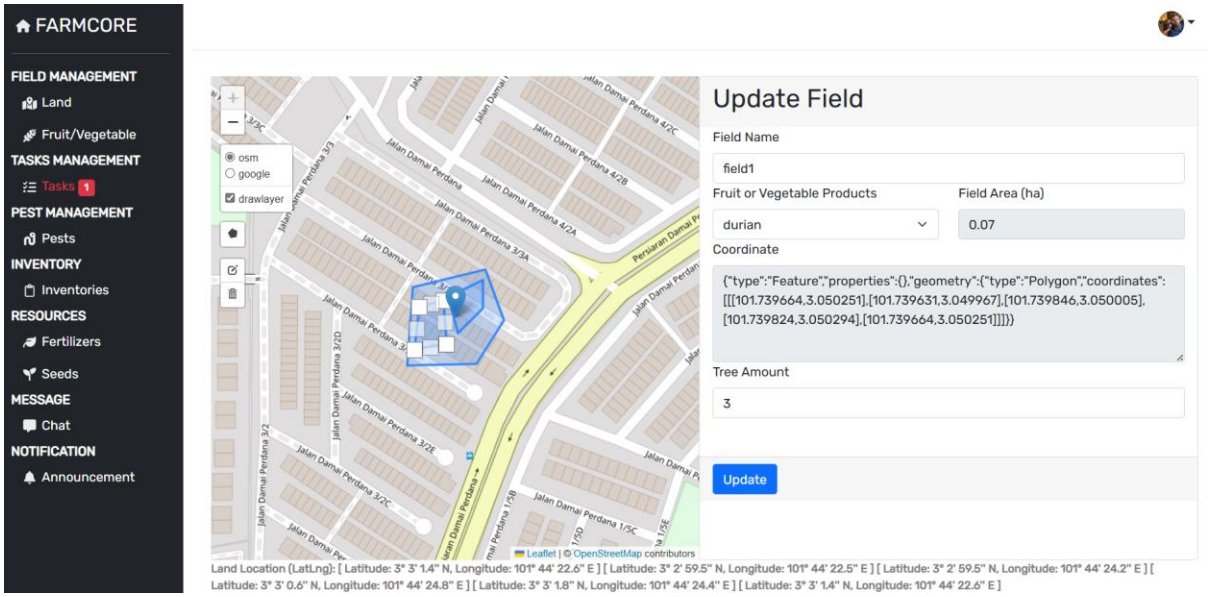


Figure 4.3.15 UID of Update Field Page

UID of Delete Field Page

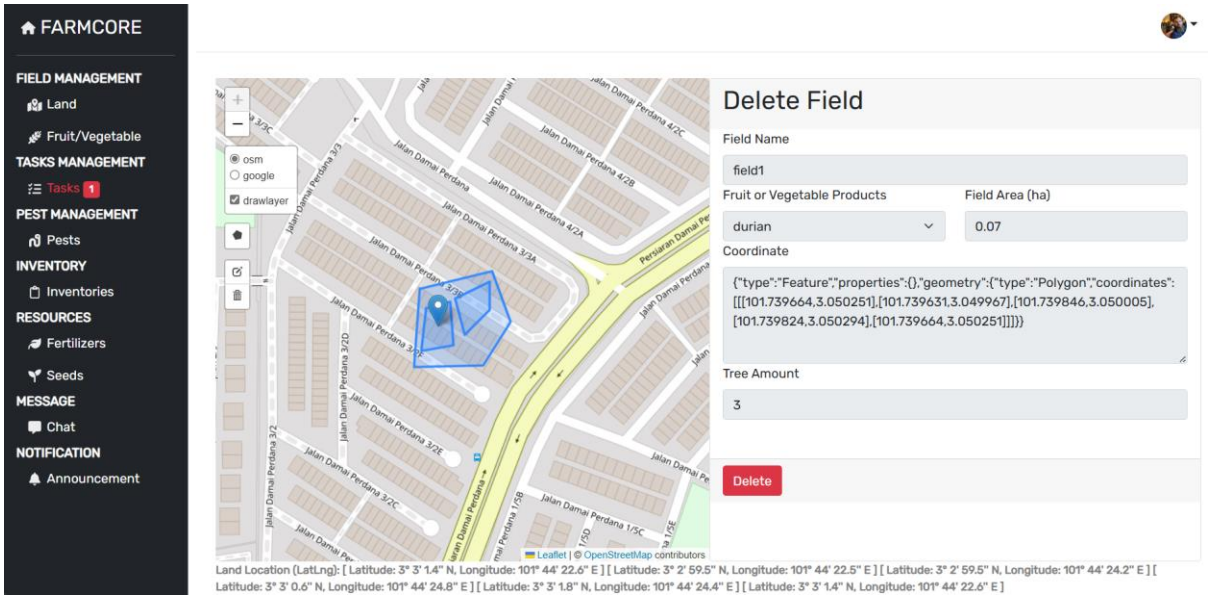


Figure 4.3.16 UID of Delete Field Page

UID of Field Dashboard Page

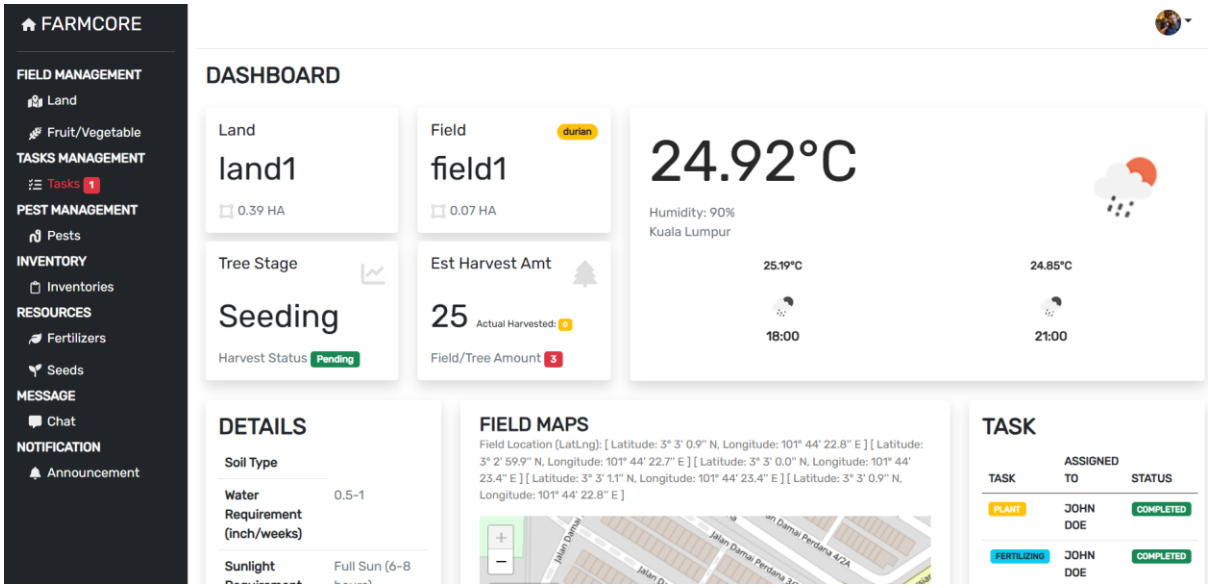
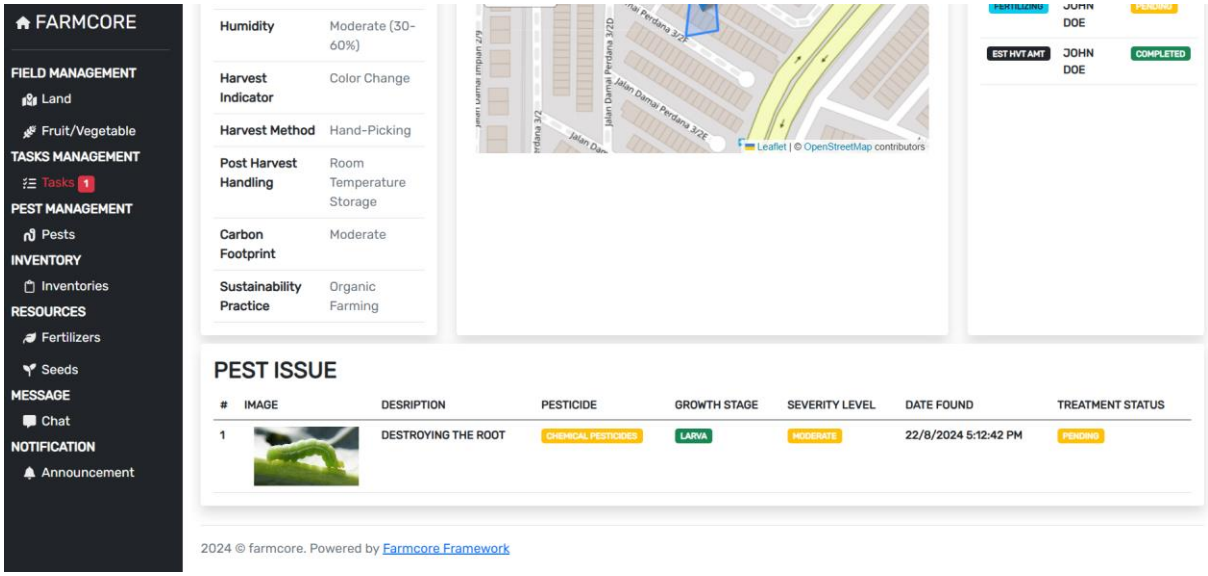


Figure 4.3.17 UID of Field Dashboard Page



(cont.) Figure 4.3.17 UID of Field Dashboard Page

UID of Add Announcement of Field

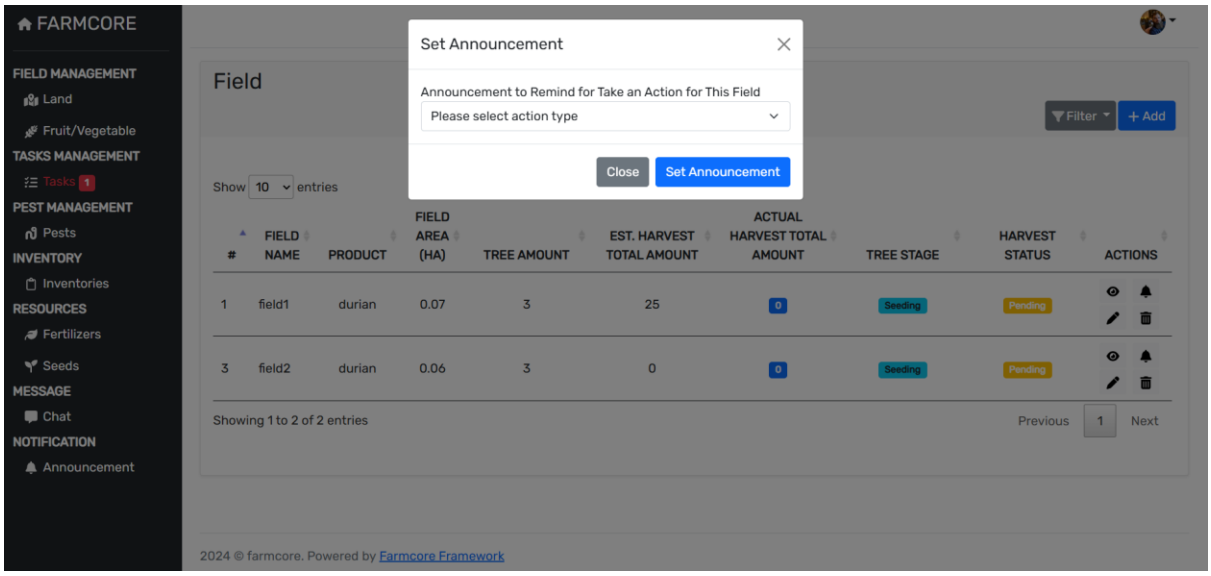


Figure 4.3.18 UID of Add Announcement of Field

UID of Tasks Page

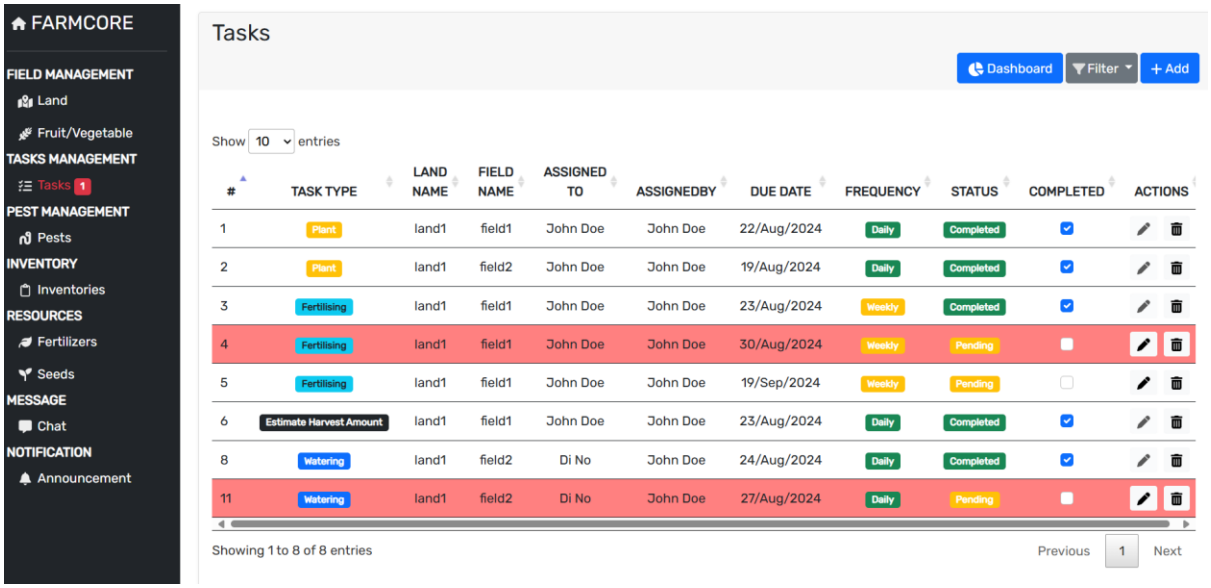


Figure 4.3.19 UID of Tasks Page

UID of Add New Task Page

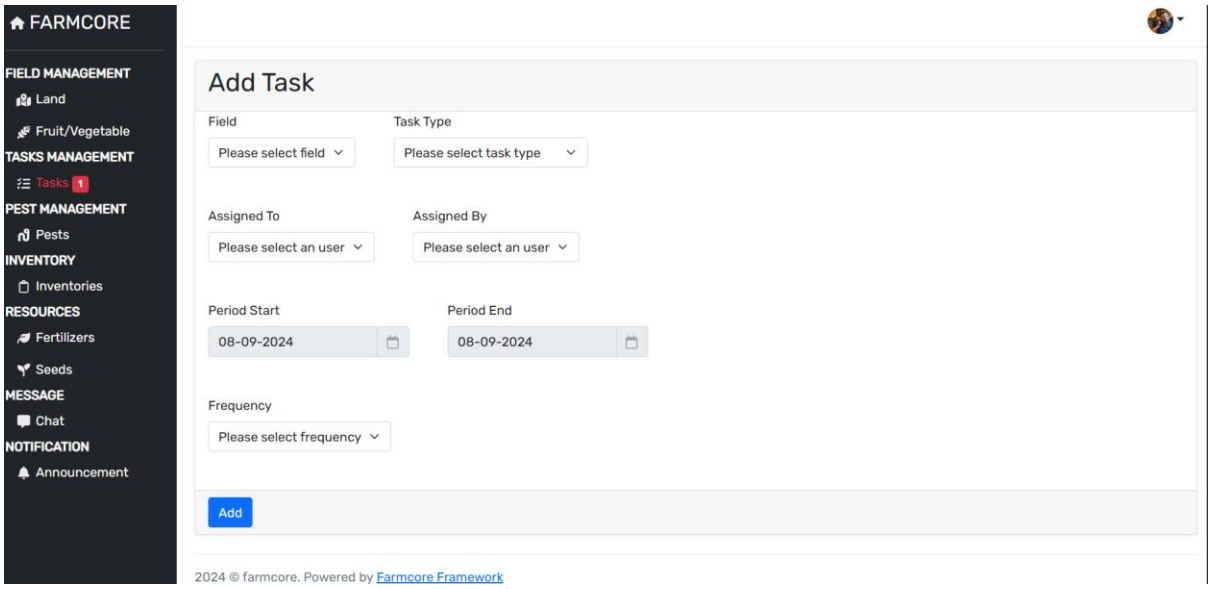


Figure 4.3.20 UID of Add New Tasks Page

UID of Update Task Page

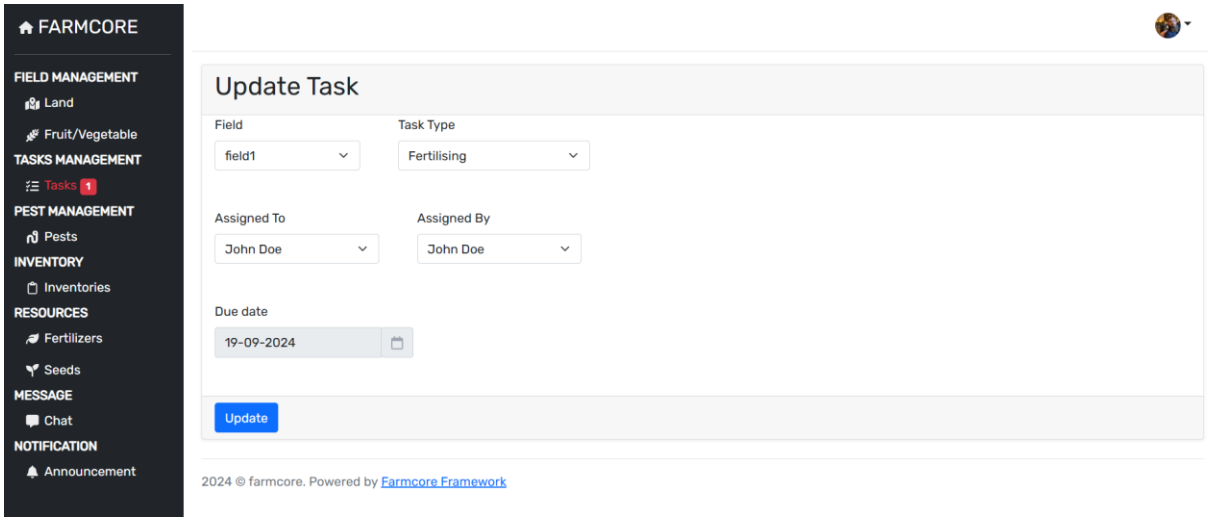


Figure 4.3.21 UID of Update Task Page

UID of Delete Task Page

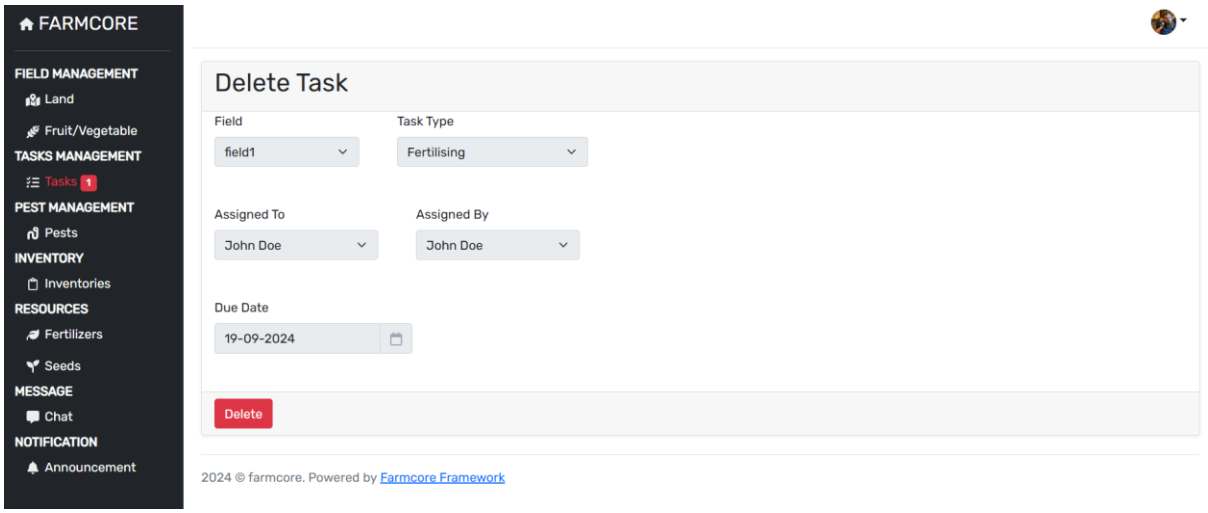


Figure 4.3.22 UID of Delete Task Page

UID of Task Dashboard

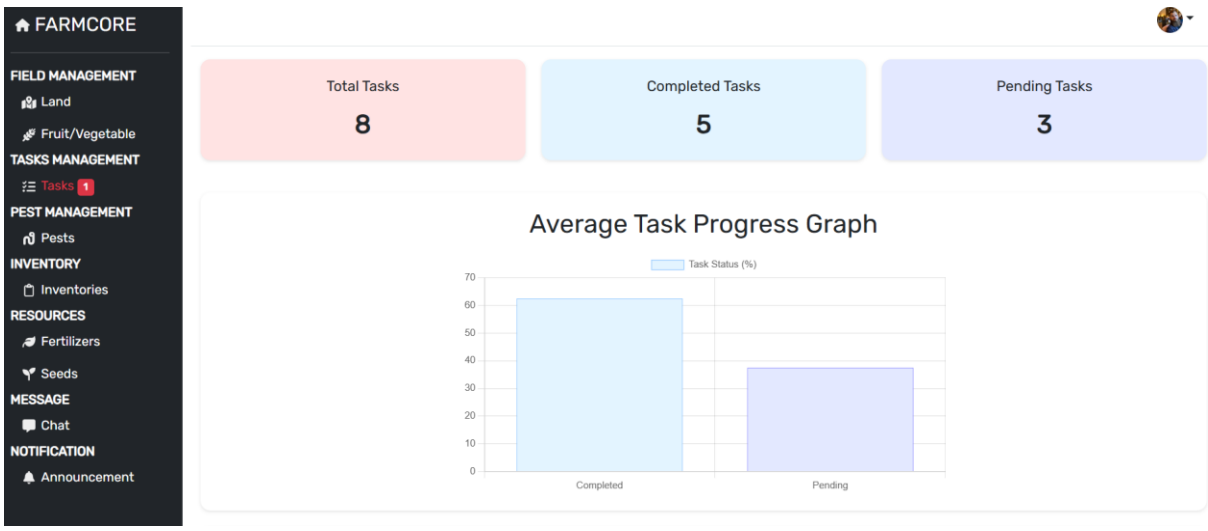
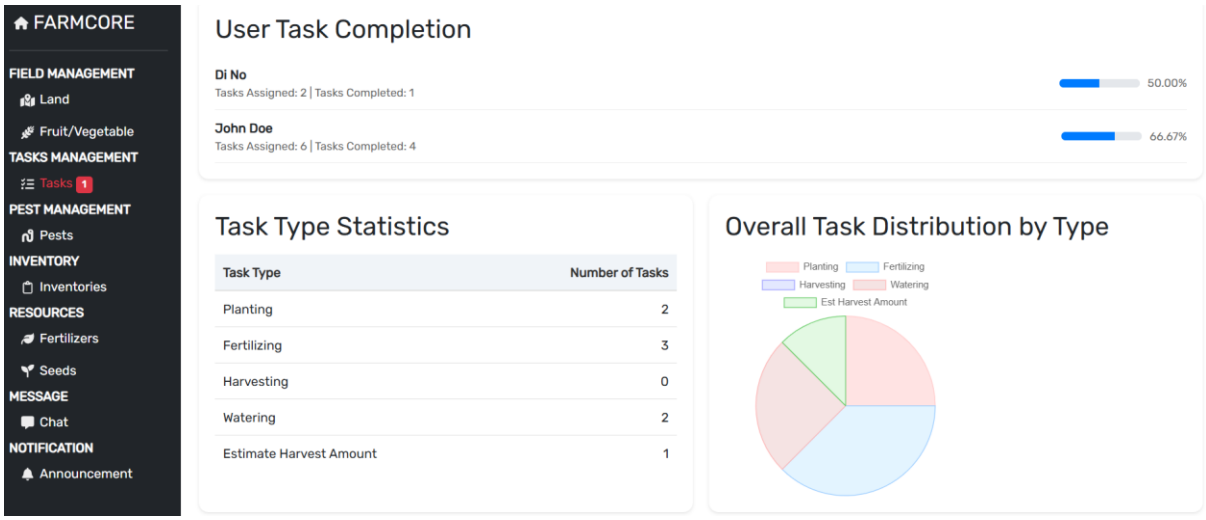
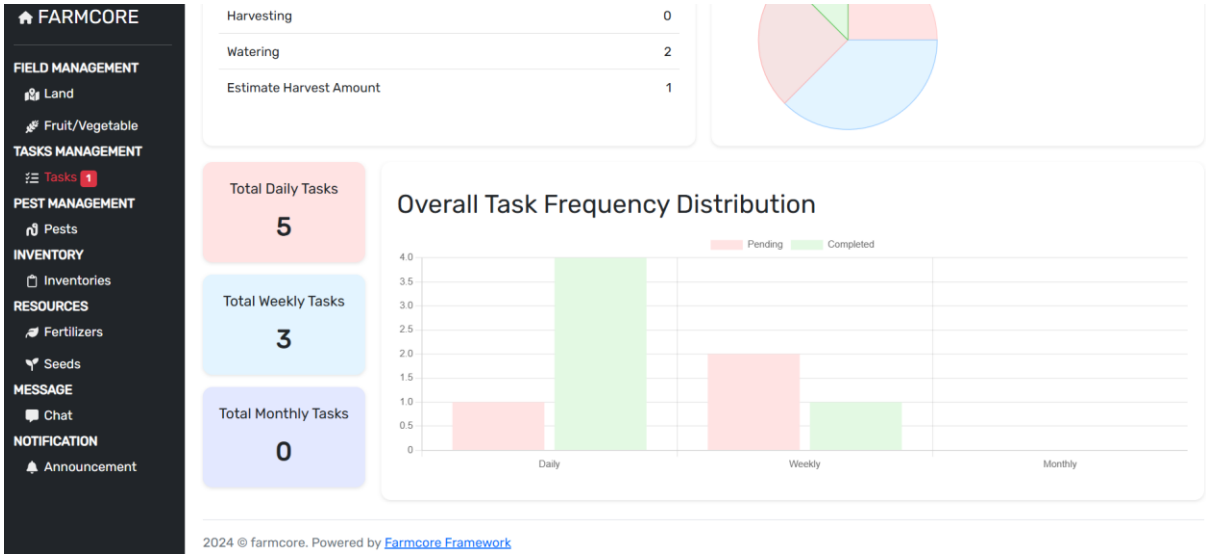


Figure 4.3.23 UID of Task Dashboard



(cont.) Figure 4.3.23 UID of Task Dashboard



(cont.) Figure 4.3.23 UID of Task Dashboard

UID of Pests Page

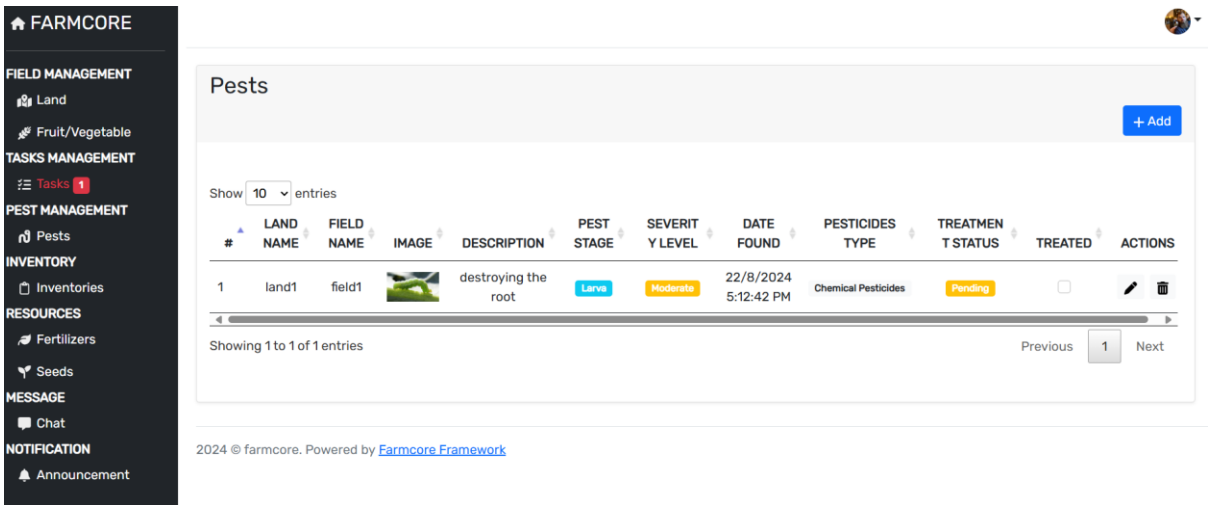


Figure 4.3.24 UID of Pests Page

UID of Add New Pests Page

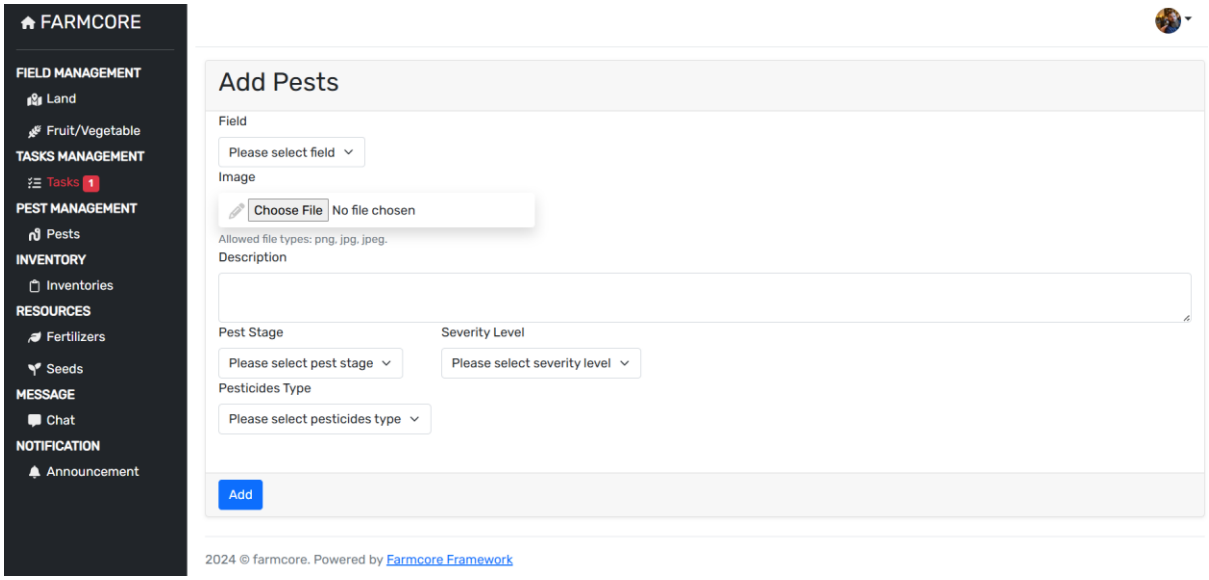


Figure 4.3.25 UID of Add New Pests Page

UID of Update Pests Page

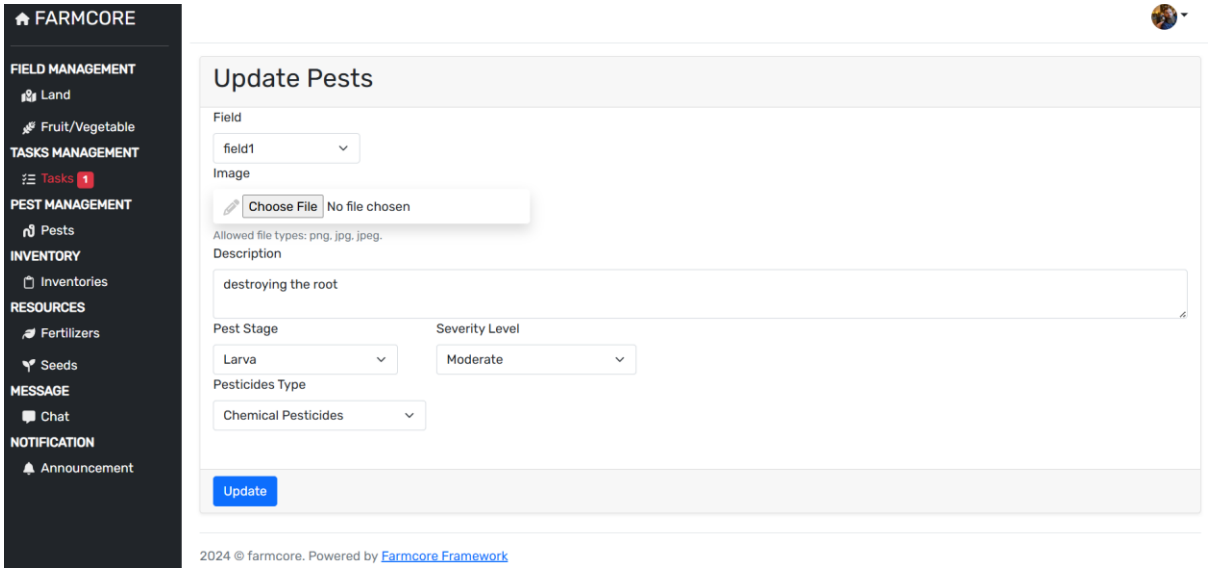


Figure 4.3.26 UID of Update Pests Page

UID of Delete Pests Page

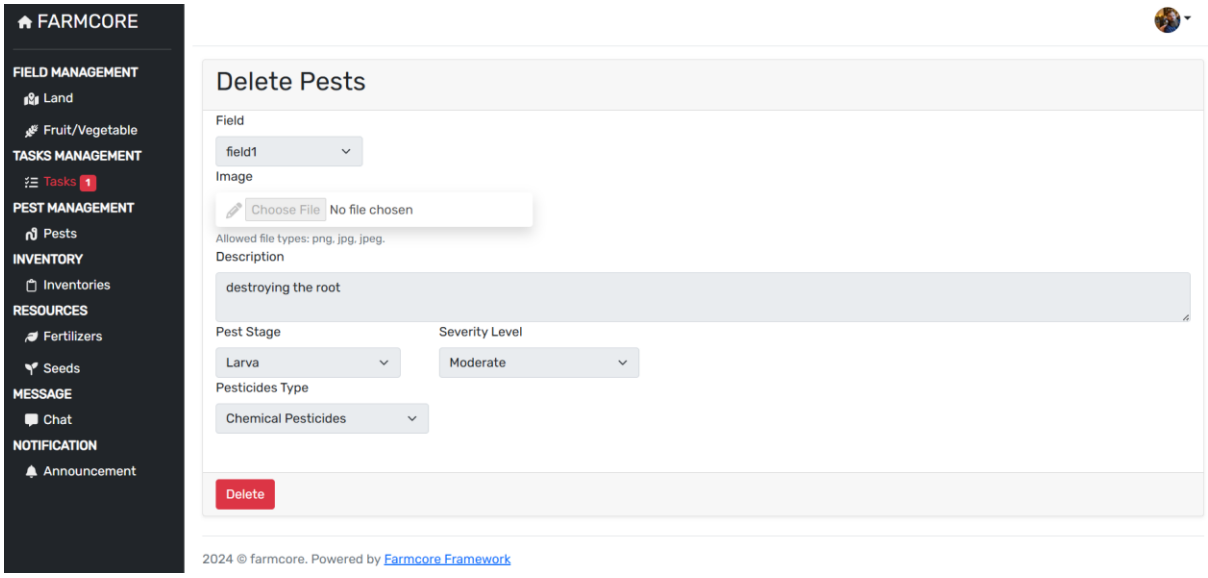


Figure 4.3.27 UID of Delete Pests Page

UID of Inventories Page

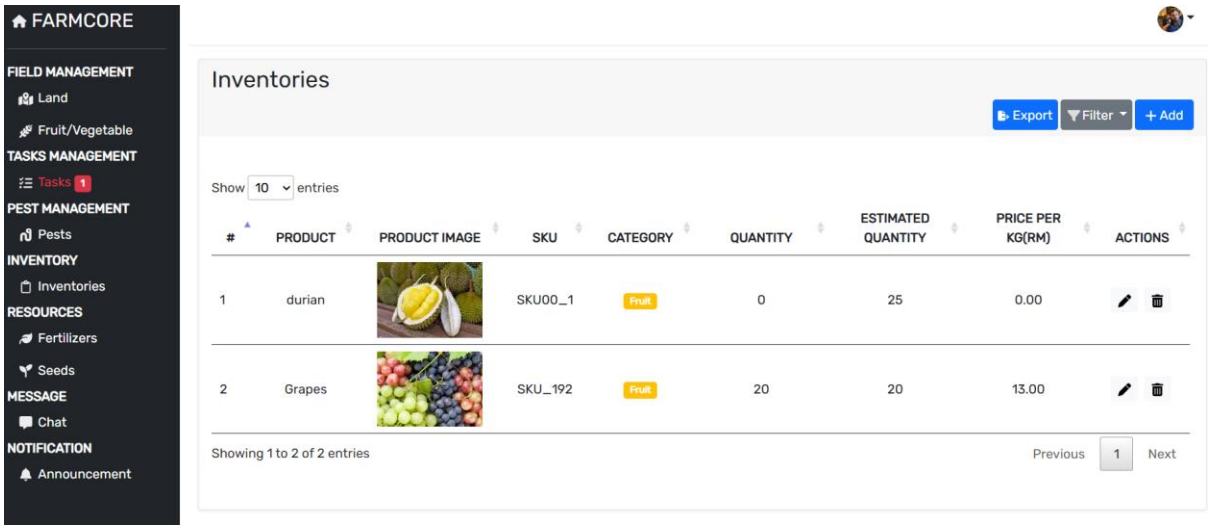


Figure 4.3.28 UID of Inventories Page

UID of Add Inventory Page

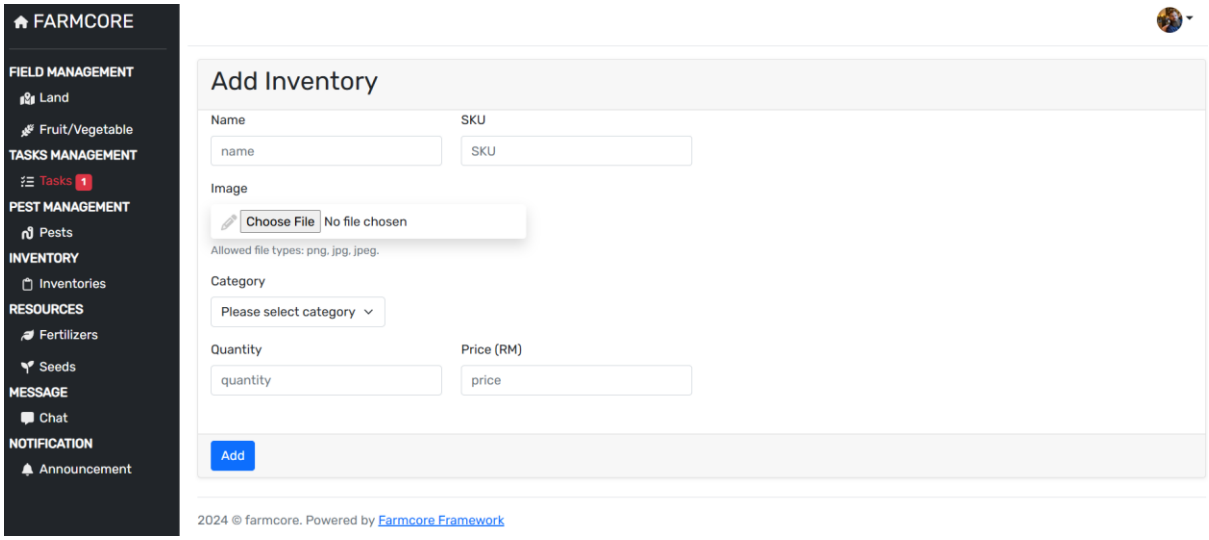


Figure 4.3.29 UID of Add Inventory Page

UID of Update Inventory Page

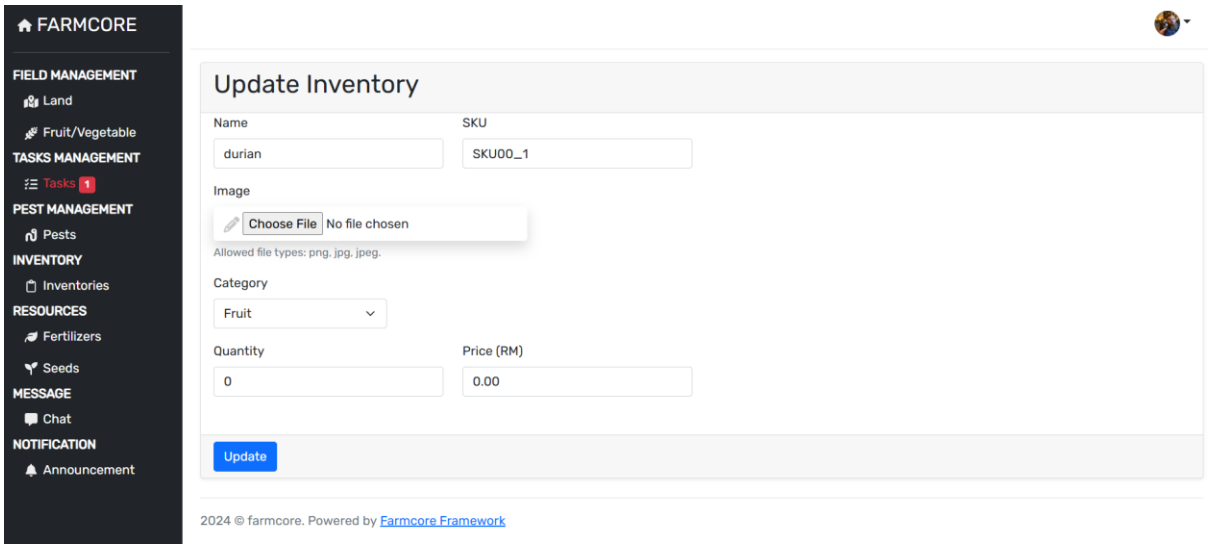


Figure 4.3.30 UID of Update Inventory Page

UID of Delete Inventory Page

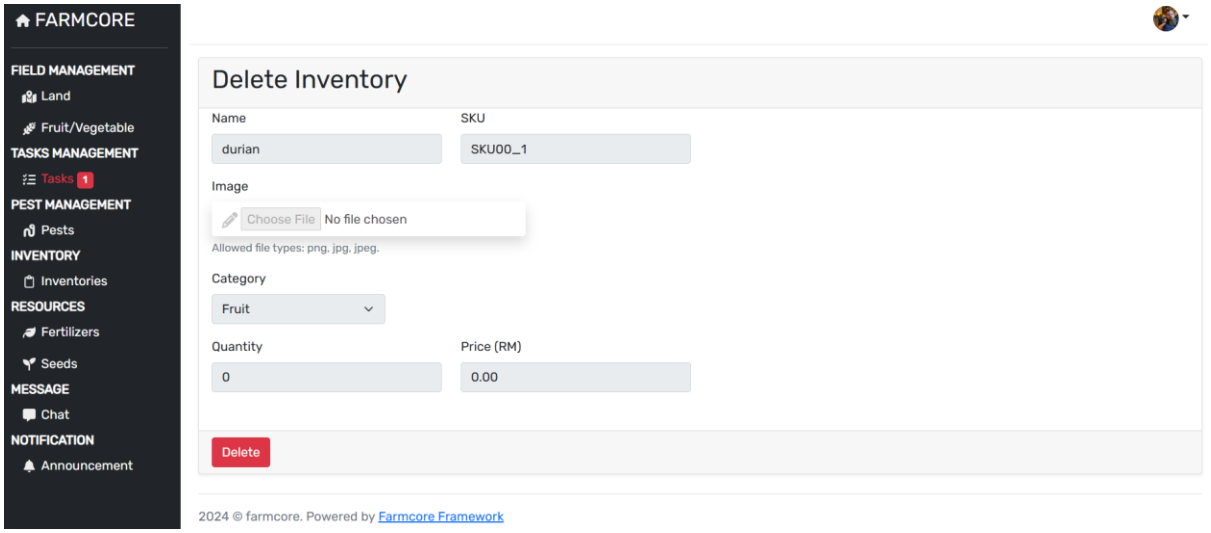


Figure 4.3.31 UID of Delete Inventory Page

UID of Export Inventory Data

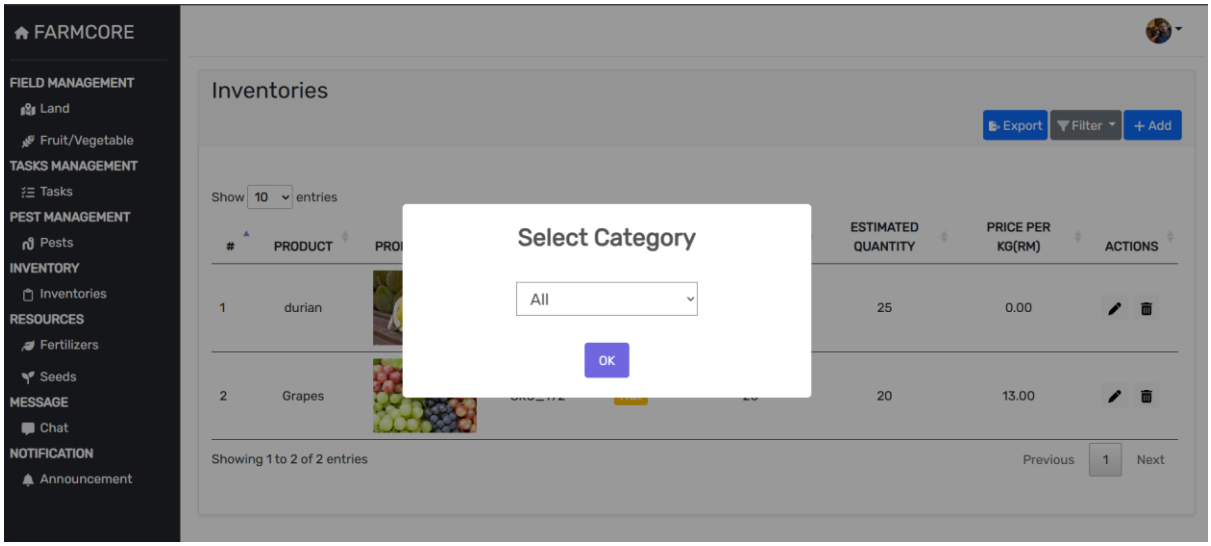


Figure 4.3.32 UID of Export Inventory Data

UID of Fertilizers Page

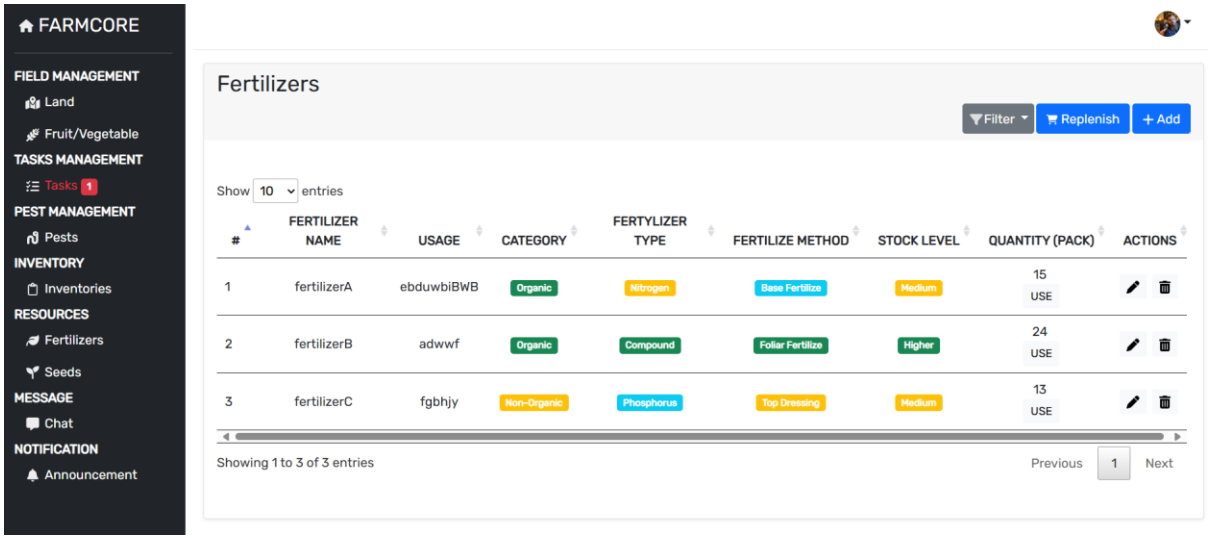


Figure 4.3.33 UID of Fertilizers Page

UID of Add New Fertilizer Page

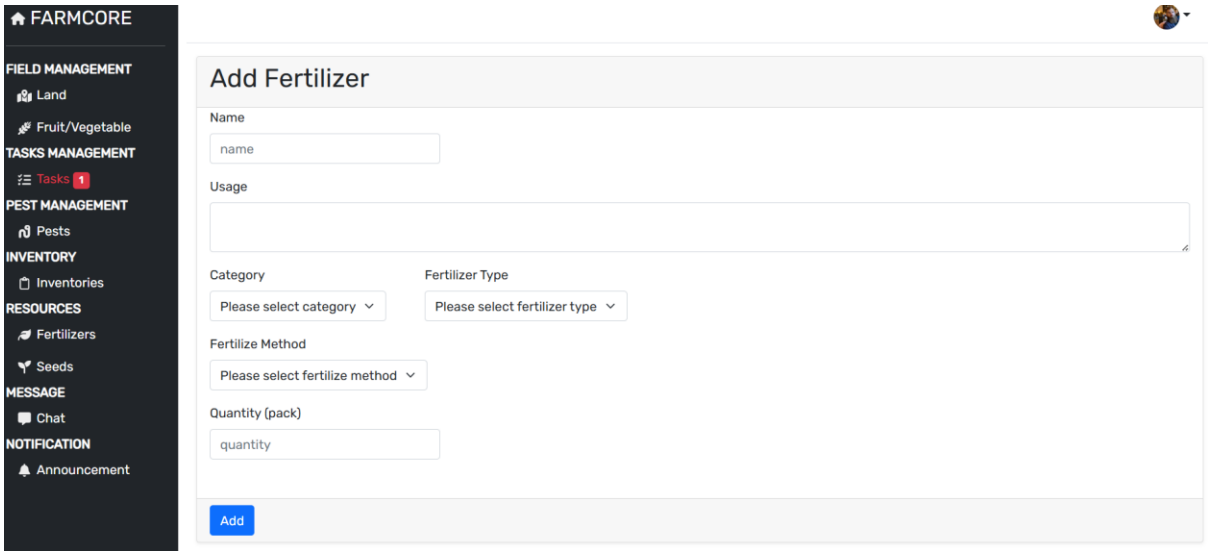


Figure 4.3.34 UID of Add New Fertilizer Page

UID of Update Fertilizer Page

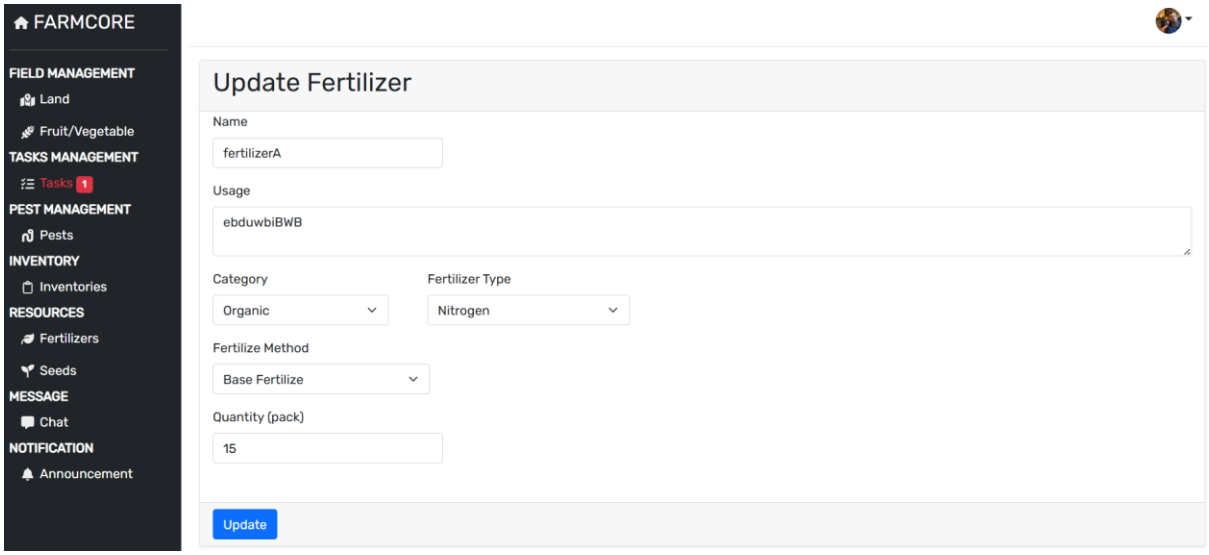


Figure 4.3.35 UID of Update Fertilizer Page

UID of Delete Fertilizer Page

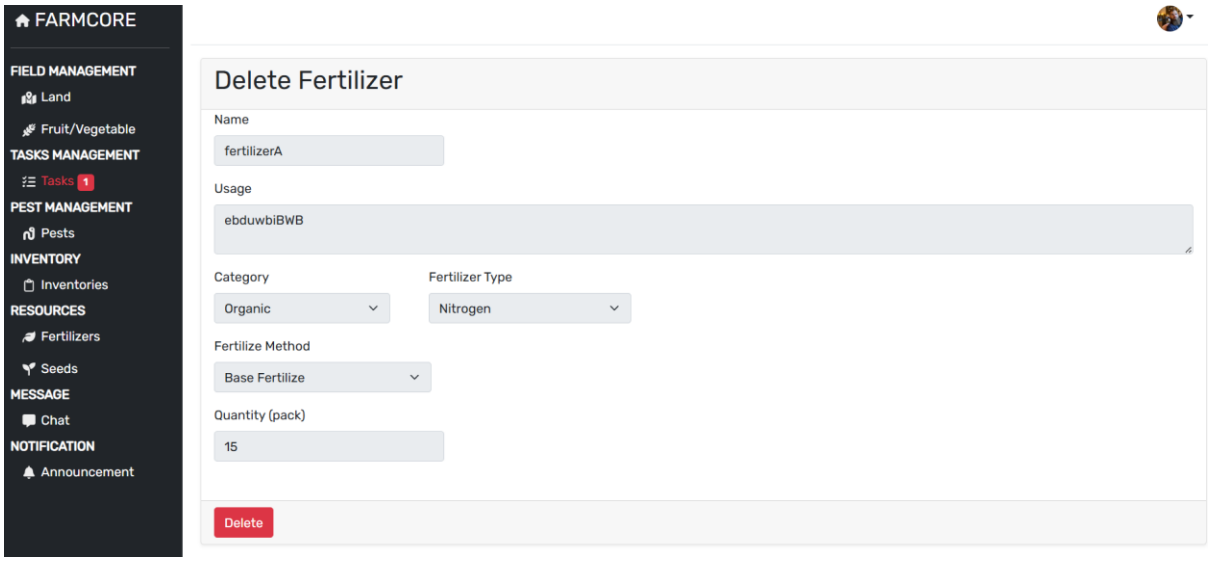


Figure 4.3.36 UID of Delete Fertilizer Page

UID of Replenish Fertilizer Amount (Pack)

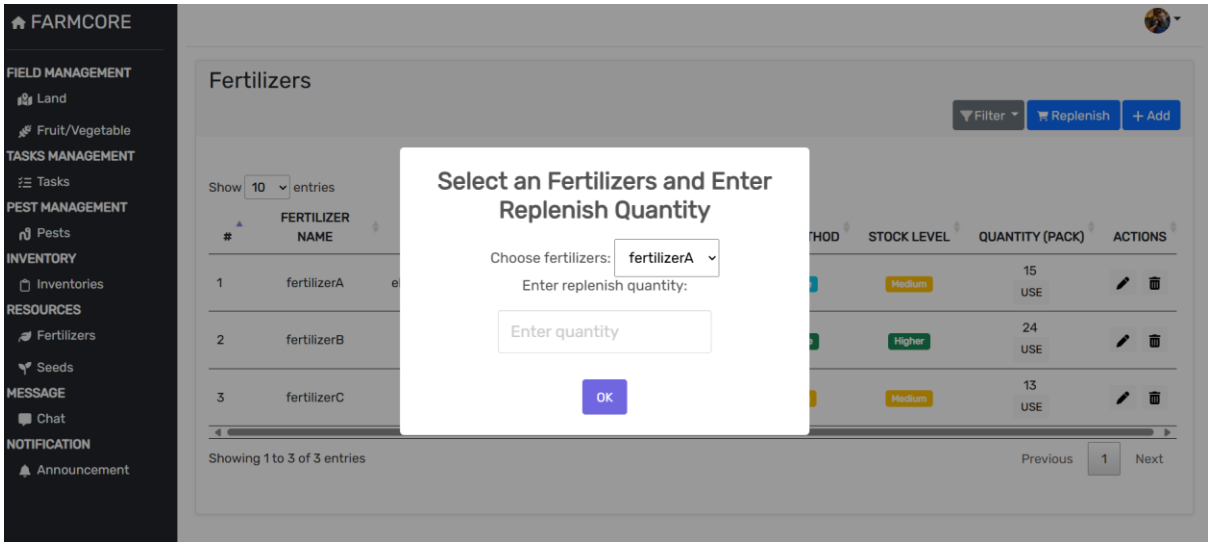


Figure 4.3.37 UID of Replenish Fertilizer Quantity

UID of Deduct Used Fertilizer Amount

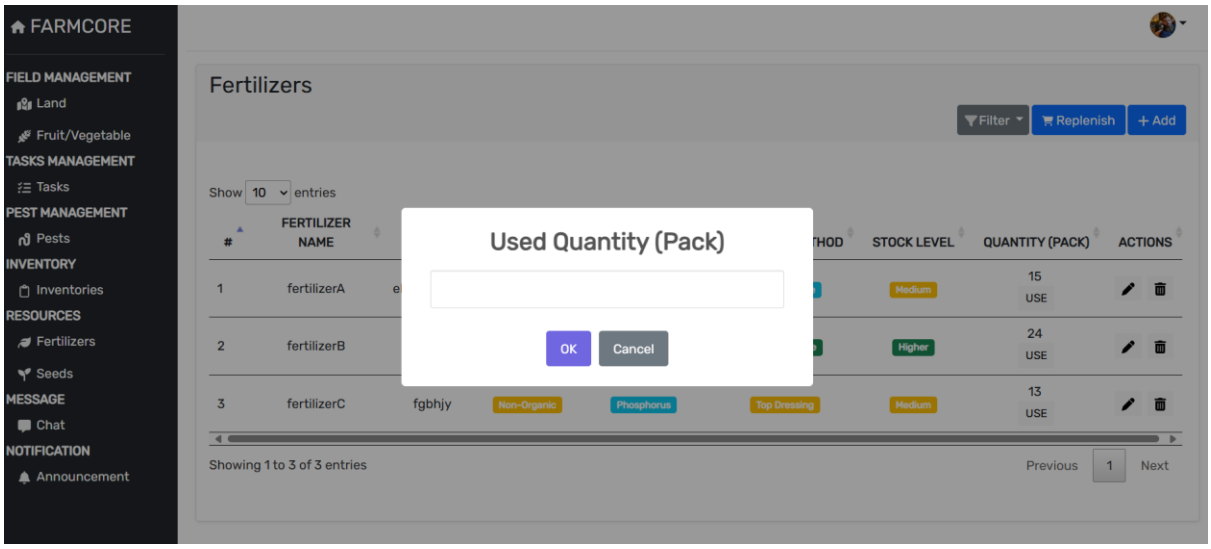


Figure 4.3.38 UID of Deduct Used Fertilizer Quantity

UID of Seeds Page

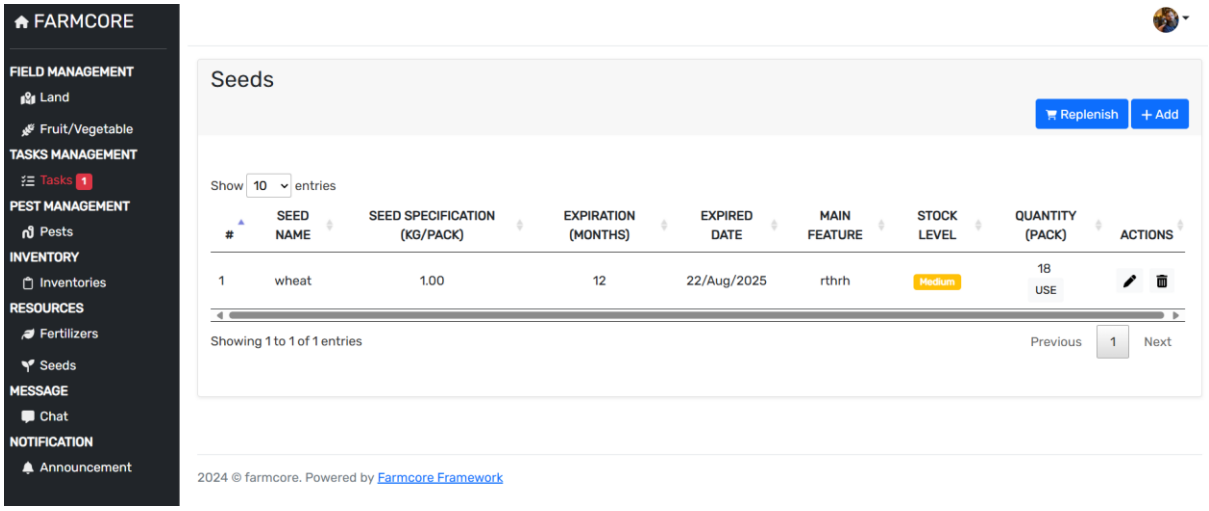


Figure 4.3.49 UID of Seeds Page

UID of Add New Seed Page

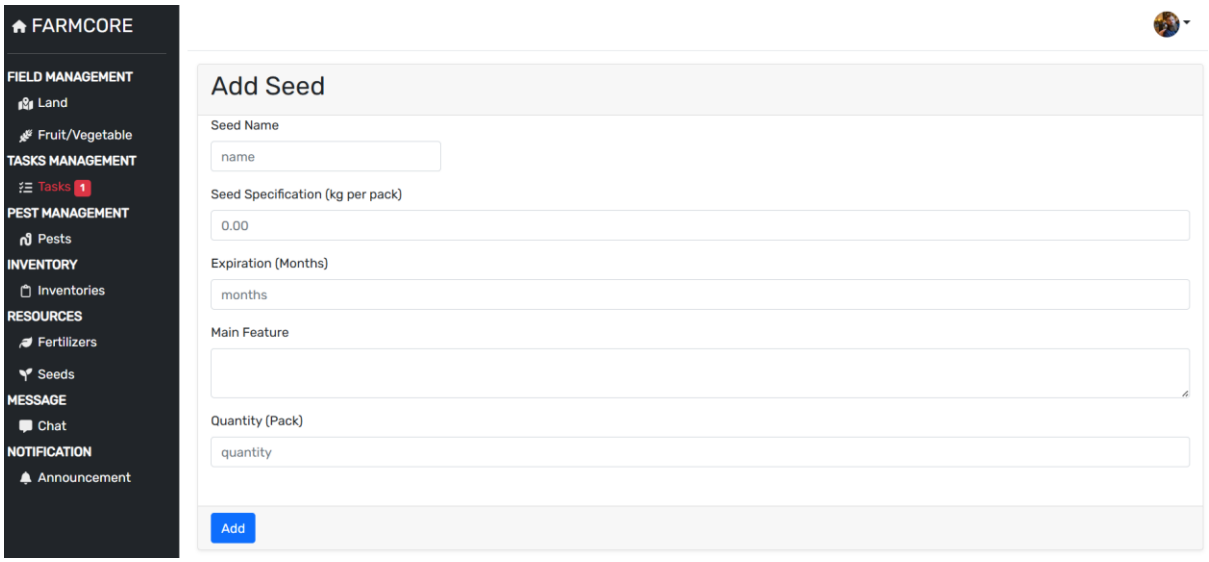


Figure 4.3.40 UID of Add New Seed Page

UID of Update Seed Page

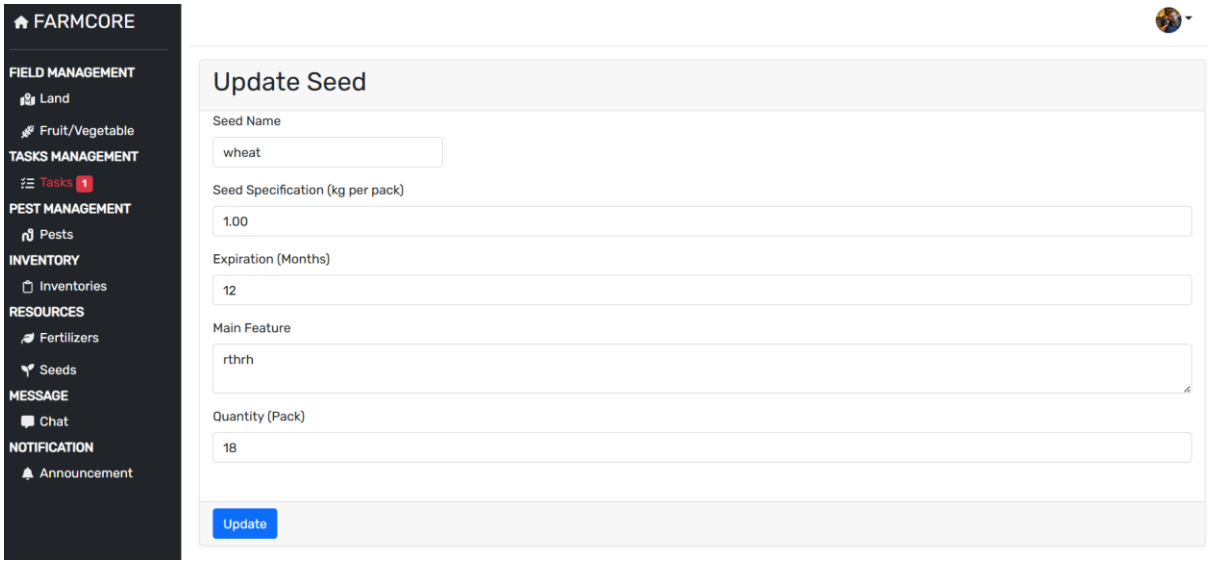


Figure 4.3.41 UID of Update Seed Page

UID of Delete Seed Page

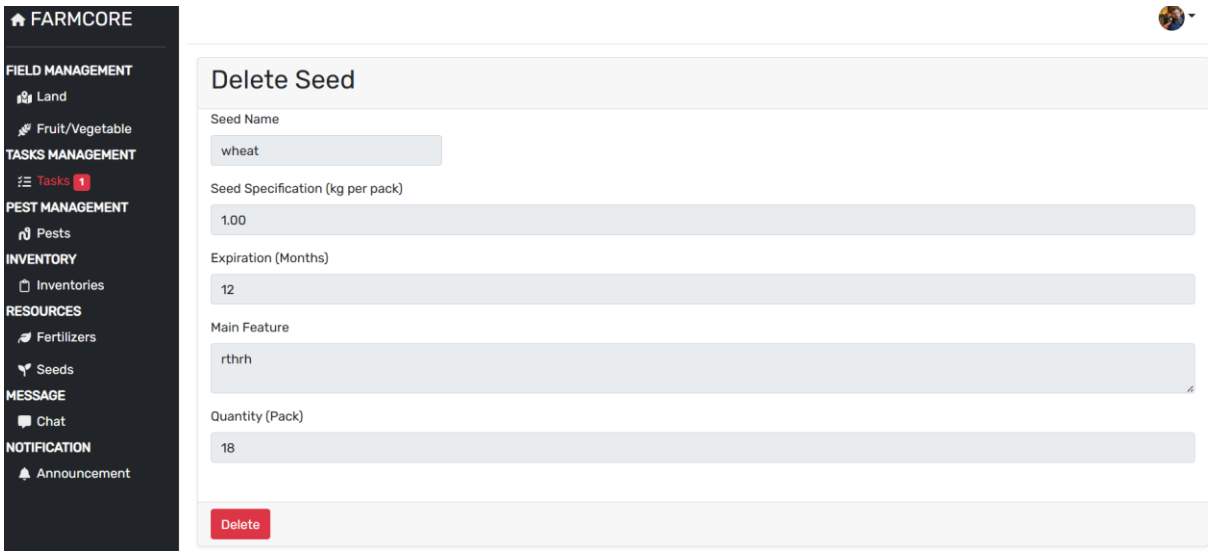


Figure 4.3.42 UID of Delete Seed Page

UID of Replenish Seed Amount (Pack)

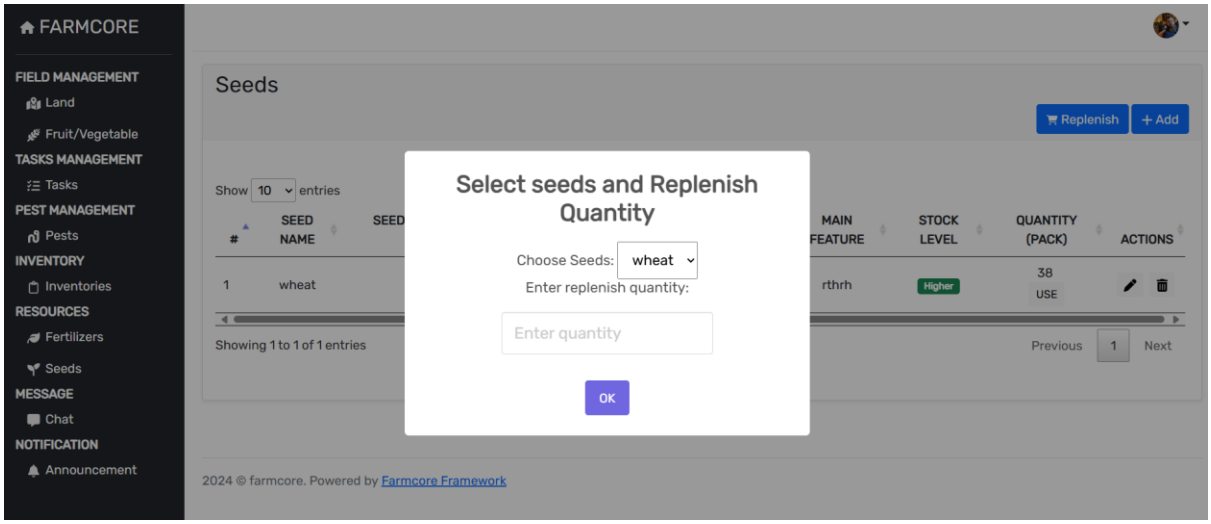


Figure 4.3.43 UID of Replenish Seed Quantity (Pack)

UID of Deduct Used Seed Amount (Pack)

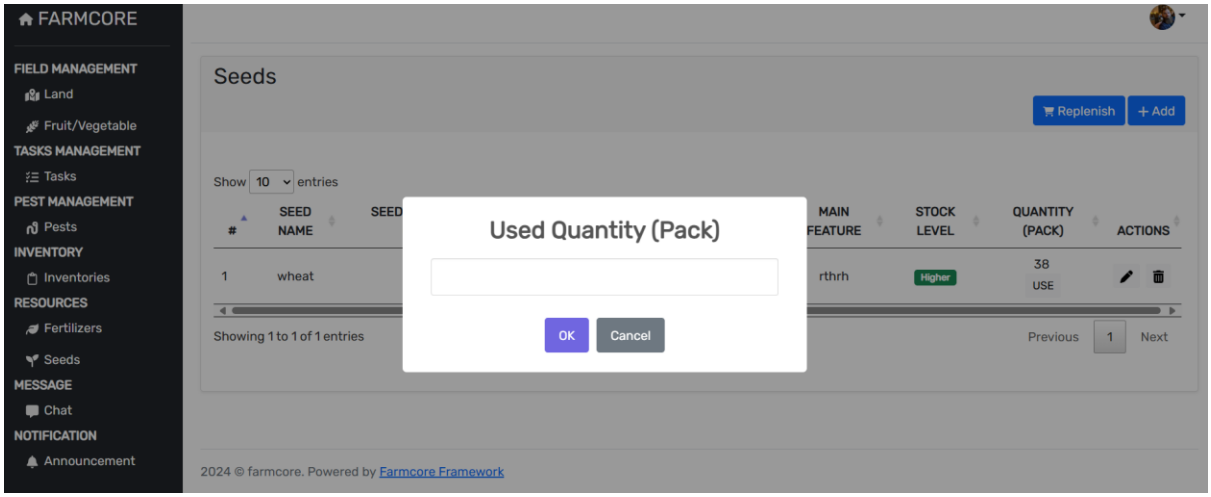


Figure 4.3.44 UID of Deduct Used Seed Quantity (Pack)

UID of Chat Page

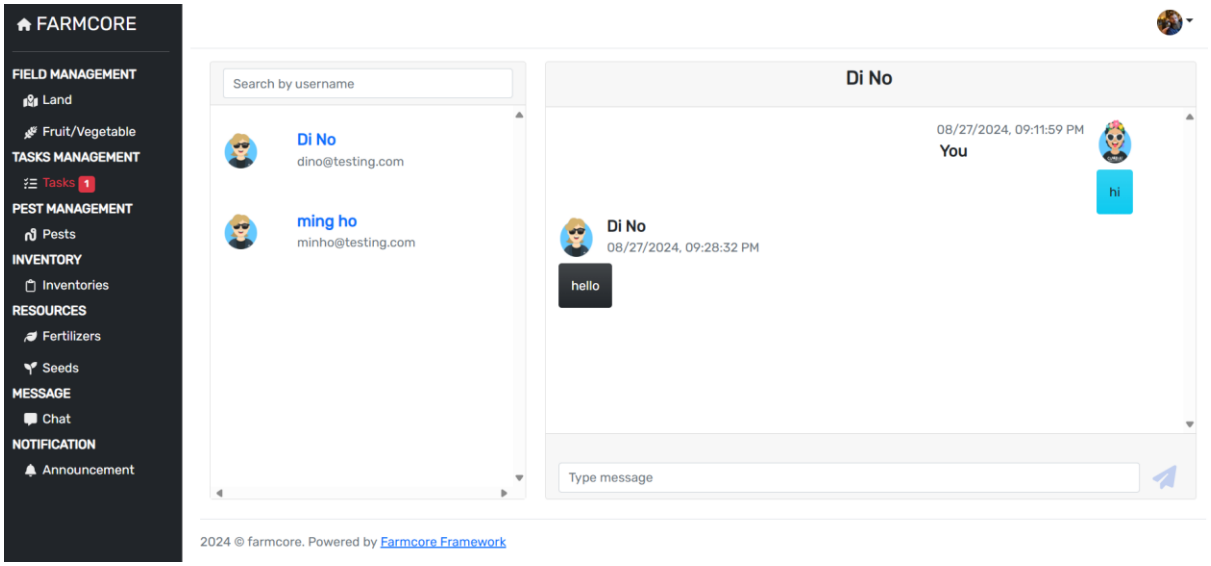


Figure 4.3.45 UID of Chat Page

UID of Announcement Page

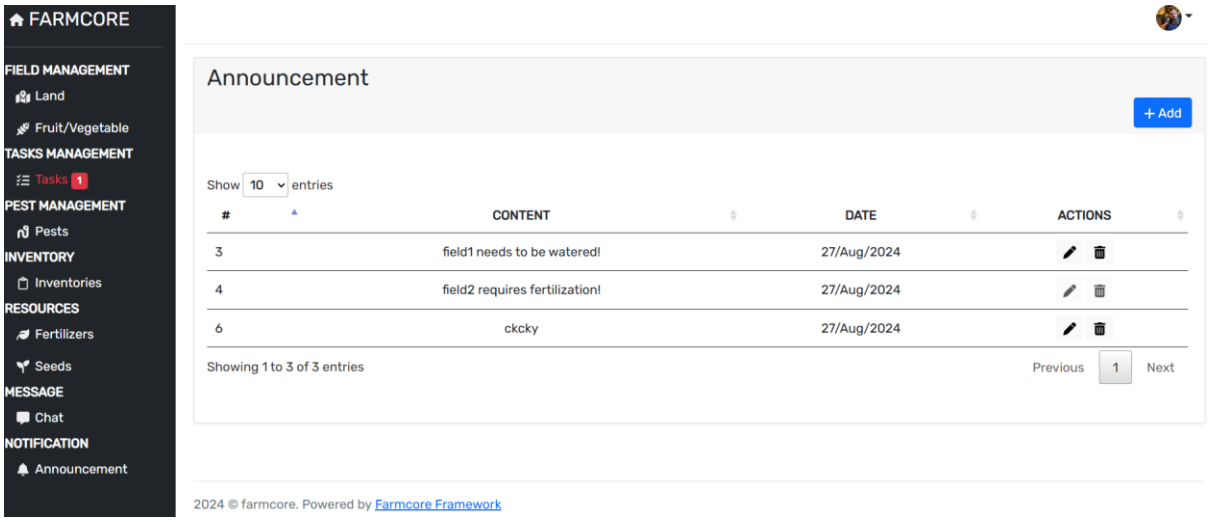


Figure 4.3.46 UID of Announcements Page

UID of Add New Announcement

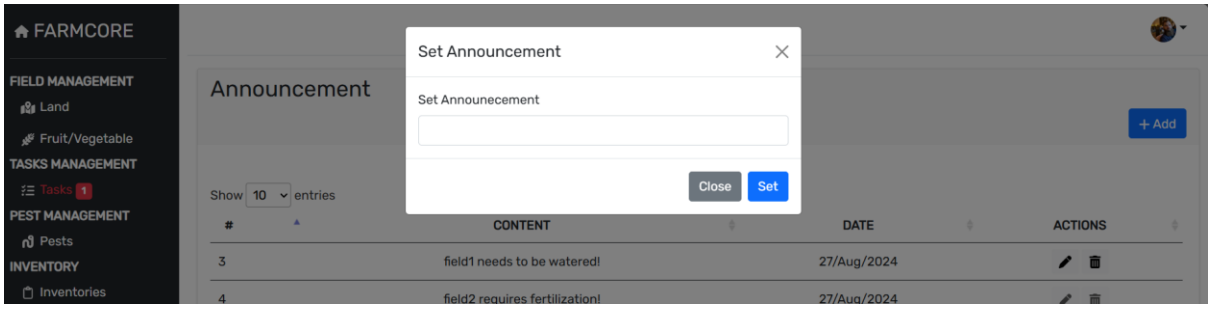


Figure 4.3.47 UID of Add New Announcement

UID of Update Announcement

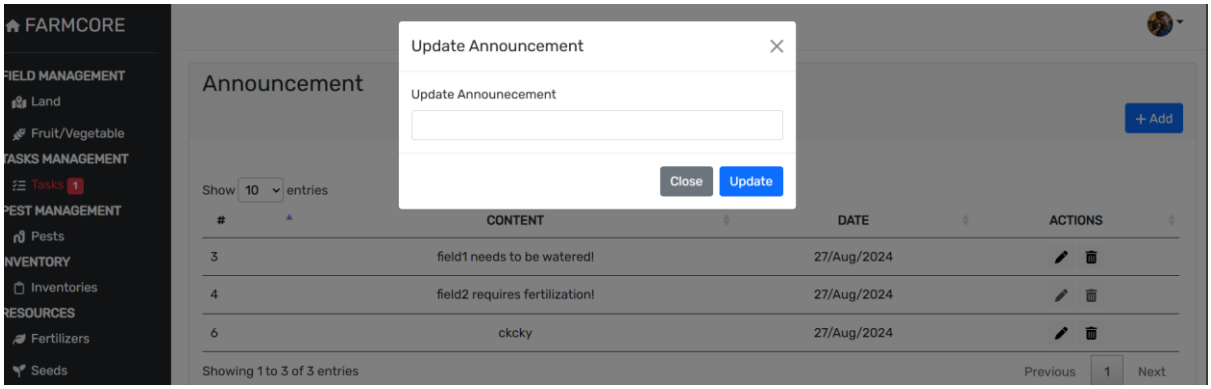


Figure 4.3.48 UID of Update Announcement

UID of Delete Announcement

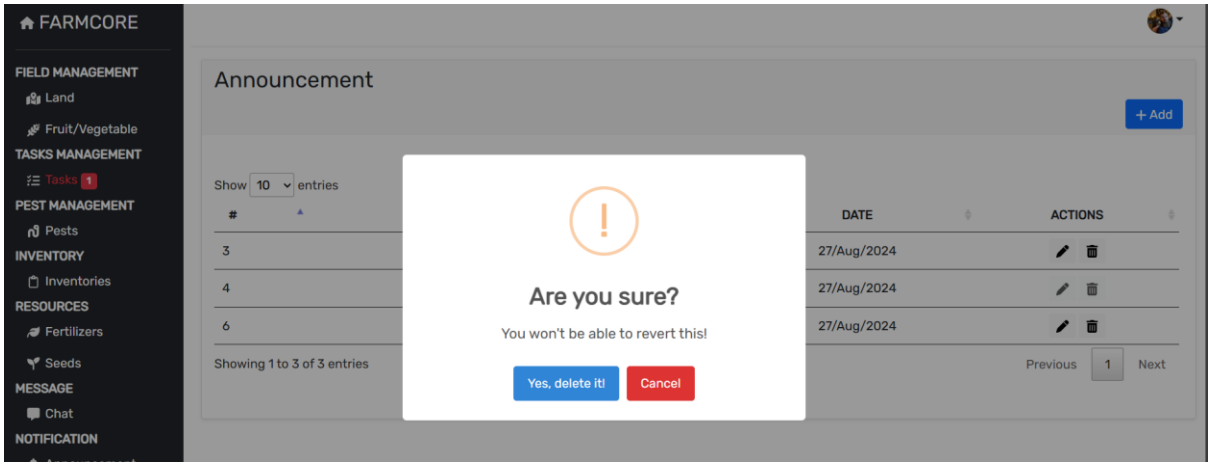


Figure 4.3.49 UID of Delete Announcement

Chapter 5

System Implementation

5.1 Software Setup

5.1.1 Microsoft Visual Studio 2022 Community

Figure 5.1.1 Microsoft Visual Studio 2022 Community

Figure 5.1.1 shows the Microsoft Visual Studio 2022 community which I used to develop my project. It is an integrated development environment (IDE) that is useful for the developer to develop software as it includes facilities of automation build-in tools, source code editor, and debugging tools. Besides, this software supported various programming languages for the developer to develop software. The programming language I used in this software is ASP.NET C# to develop the web application. Before developing the web application, I first downloaded .NET 6 packages in this Microsoft Visual Studio 2022 community to set up the web application environment.

5.1.2 XAMPP

Figure 5.1.2 XAMPP

Figure 5.1.2 above is XAMPP which is a database software for user to store their data. This software is developed by Apache distribution which includes MariaDB, PHP, and Perl. This software has provided 5 modules which are Apache, MySQL, Tomcat, Mercury, and FileZilla. The module I have used in this web application is Apache which is a web server of XAMPP localhost, and MySQL module as a database server to store all the data of the application.

5.2 Setting Configuration

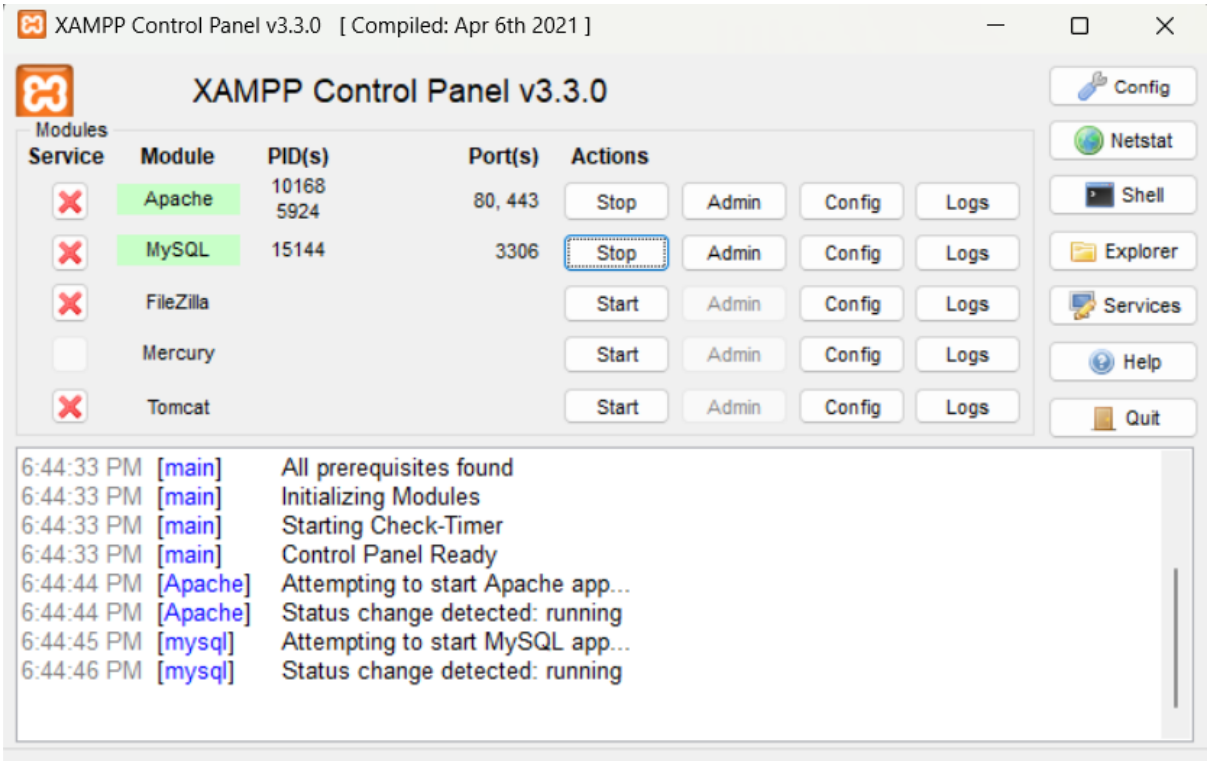


Figure 5.2.0 XAMPP Control Panel

The figure above shows the XAMPP Control Panel used to start the MySQL database server. It is required to start up the Apache web server before accessing the phpMyAdmin server. After starting both servers, the phpMyAdmin server became available to access to manage the database for my web application.

Figure 5.2.1 Database Connection Environment Setup in "appsetting.json"

```
public static string GetConnection()
{
    string? connectionstring = Environment.GetEnvironmentVariable("ConnectionStrings:Default") != null ? Environment.GetEnvironmentVariable("C
    if (connectionstring == null)
    {
        return "";
    }
    else
    {
        return connectionstring;
    }
}
```

Figure 5.2.2 Get Database Connection Environment in "Database.cs"

Figure 5.2.1 and figure 5.2.2 above shows the database configuration setup for the web application. To set up the environment in the web application, it is required to connect the MySQL server and its port in the "appsetting.json" file (figure 5.2.1). By default, the server of MySQL is 127.0.0.1 and the port number is 3306. Then provide the name of the web application

database and define the username of the database. After setting up the database environment in the “appsetting.json” file, it is required to get the database connection environment from the “appsetting.json” file in the “Database.cs” file (figure 5.2.2). This is because the “appsetting.json” is an environment setup and is not directly connected with the web application.

5.3 System Operations

Sign Up an Account

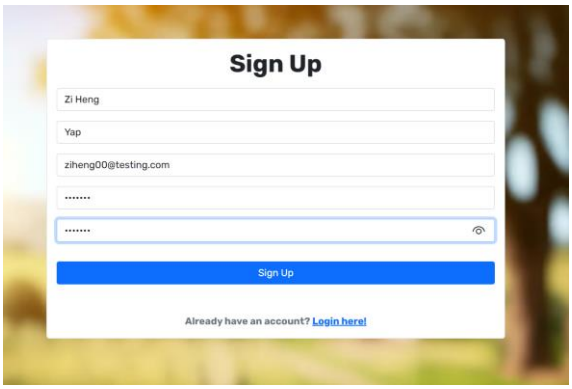


Figure 5.3.1 System Operations of Sign-Up Account

Figure 5.3.1 above shows to sign up a user account for a new user. Once the user opens this web application, the farmer will be firstly redirected to the login page. If the farmer has no account yet, the farmer can click on “Sign Up Now”; then, the farmer will be redirected to this Sign-Up page to register their account shown as in figure 5.3.1. The farmer is required to key in all the necessary information which is the first name, last name, email, password and confirm password. After successfully signing up for an account, the web application will redirect back to the login page.

Reset Password

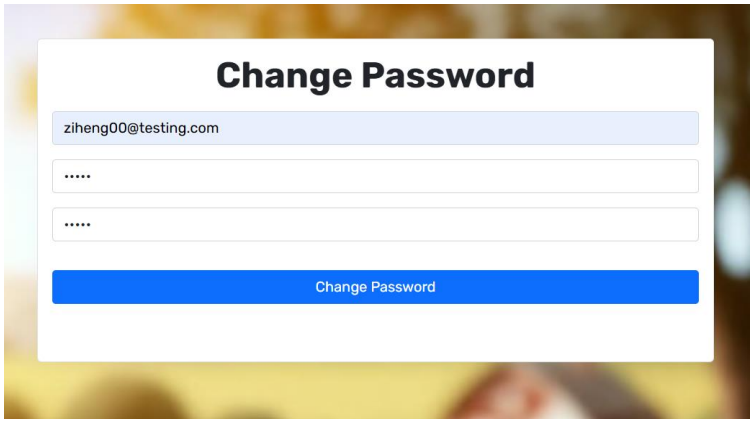


Figure 5.3.2 System Operations of Reset Password

Figure 5.3.2 shows system operations of resetting the password for the farmer who has forgotten their password while logging or wishes to change their password. The farmer can select “Forgot Password”; then, they will be redirected to this reset password page shown in Figure 5.3.2. They are required to enter their email, and new password and confirm new password. After changed, the web application will redirect back to the login page.

Sign In (Login) Page

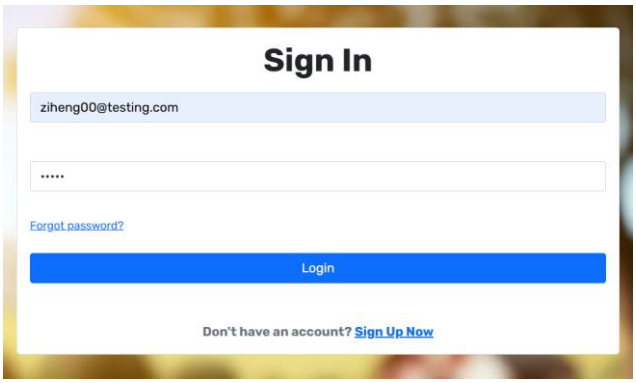


Figure 5.3.3 System Operations of Sign-In (Login) Page

Figure 5.3.3 above shows the system operations of the sign-in page for the farmer's full access to this web application. After entering their email and password correctly, the farmer will successfully access this web application.

Add, Update, Delete Land

Figure 5.3.4 Land Data Page

Figure 5.3.4 shows the land data page and this page includes “add”, “update”, and “delete” actions for the farmer to manage their land. If the farmer wants to add new land, the farmer can click on the “Add” button at the right corner. If the farmer wants to update the land, the farmer can select the “update” button inside the “Actions” column with the pencil icon. If want to

delete, select the “delete” button beside the “update button” with the “rubbish bin” icon. While, if the farmer wants to view the field(s) inside the land, the farmer can select the “Fields” button inside the “Actions” column, they then will be redirected to the “Field” page to view and manage the field that is inside the selected land. The farmer can also filter the list by the field name.

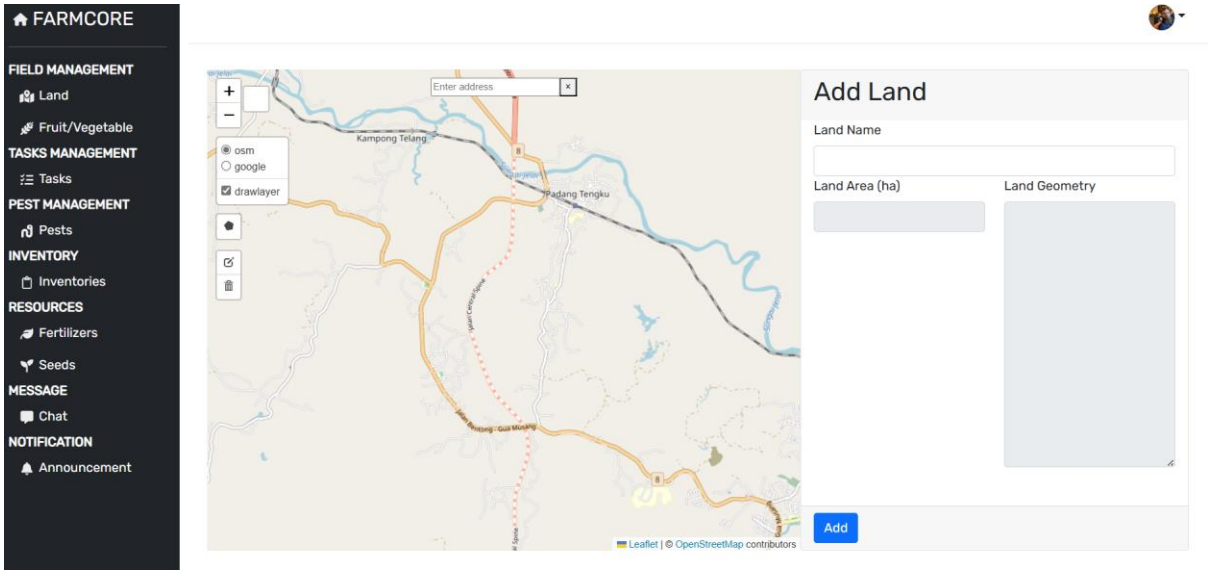


Figure 5.3.5 System Operations of Add New Land

Figure 5.3.5 shows the system operations of adding a new land page. Before plotting land on the map, the farmer is required to specify a location by searching for an address in the “Enter address” column. Otherwise, it will show an error message and ask the farmer to enter an address.

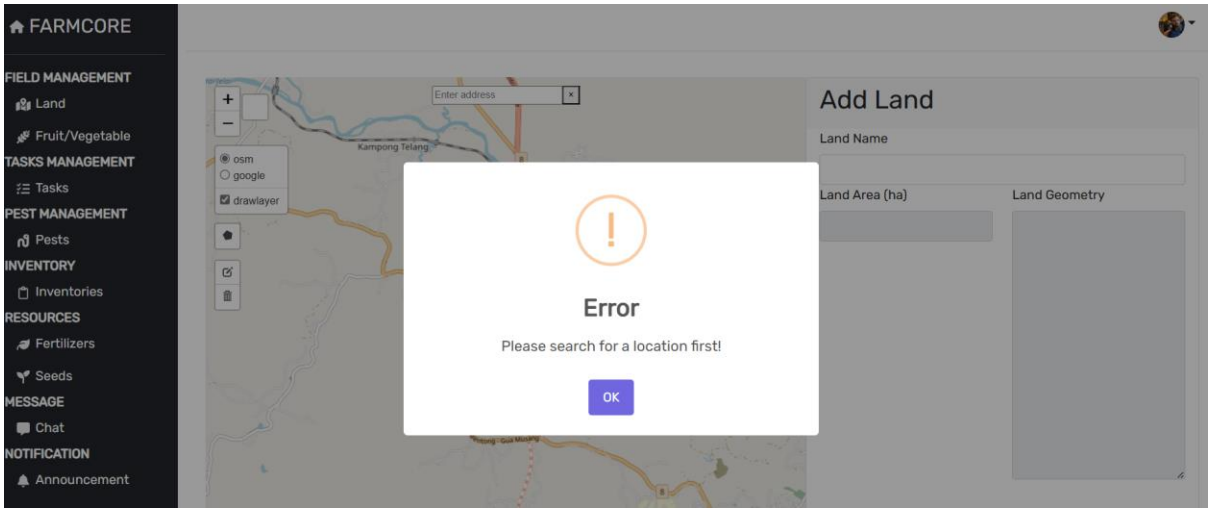


Figure 5.3.6 Error Message without Specify the Location

Figure 5.3.6 shows an error message displayed when the farmer did not search for a location and plotted the land directly on the map. The farmer has been asked to search for a location before the plot.

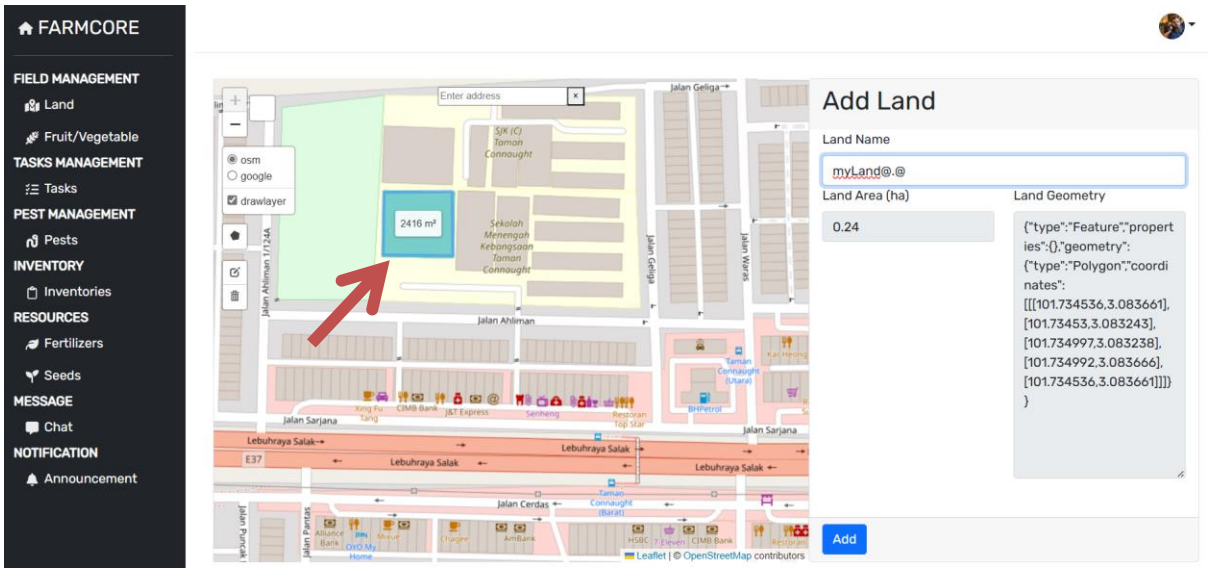


Figure 5.3.7 Plotting Land in Selected Location Successfully

Figure 5.3.7 shows the farmer successfully plotted land after searching for a location. Once the farmer successfully plotted, the area and its geometry will be auto-filled in the column. The farmer is required to provide a name for this land to convenient the other to identify. After clicking on the "Add" button, it will return to the main page of the land module and the newly added land will be successfully shown in the list.

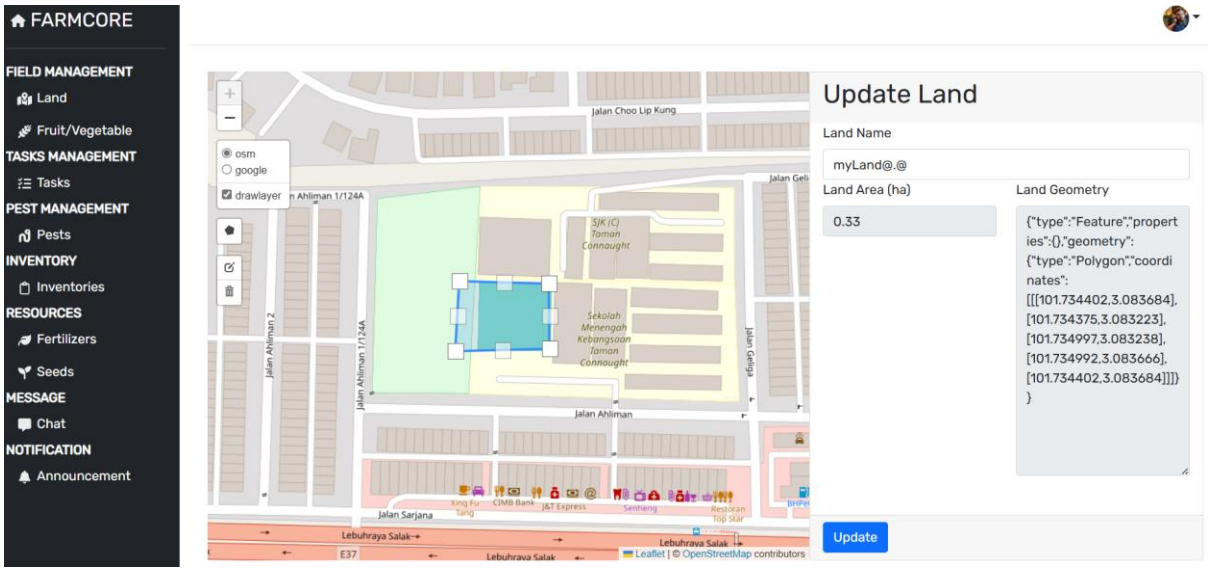


Figure 5.3.8 System Operation of Update Land

Figure 5.3.8 shows the system operations of updating the created land. In this updated land, the farmer is not allowed to search for a location, they can only drag and adjust the area of the

created land in the map; then, the newly adjusted area and geometry will update in the column. The farmer can also rename the land name. After successfully updating, it will be redirected back to the main page of the land module, and the information of the land will be updated.

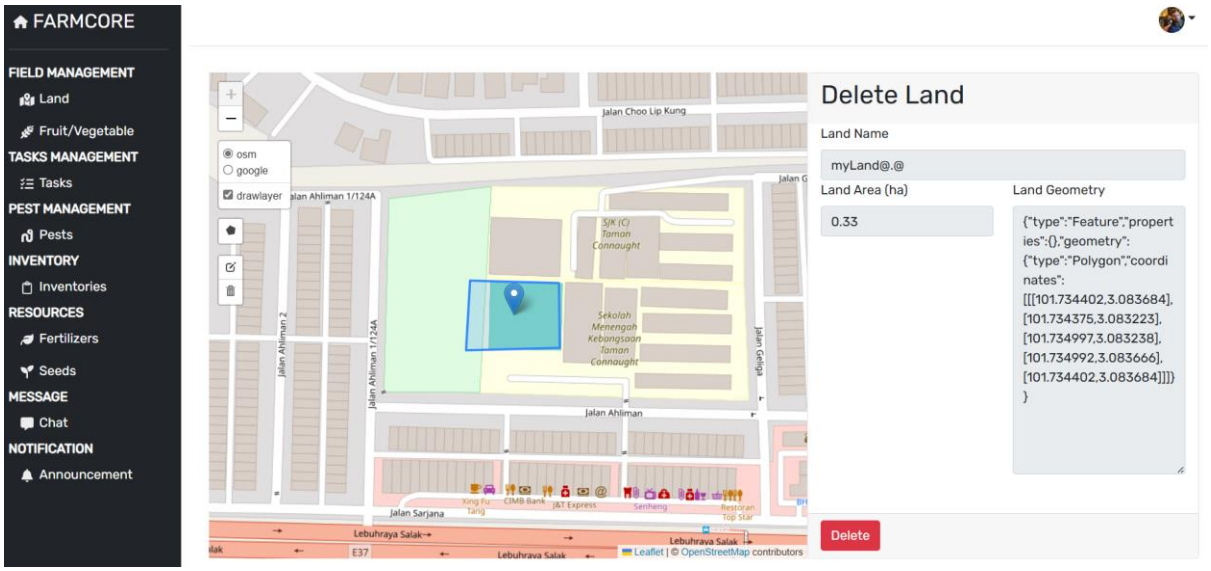


Figure 5.3.9 System Operations of Delete Land

Figure 5.3.9 shows the system operations of deleting the selected land. On this page, the farmer is not allowed to update any data on the land; they only be allowed to review the data of the land.

View, Add, Update, Delete, Field, View Dashboard, Set Field Announcement

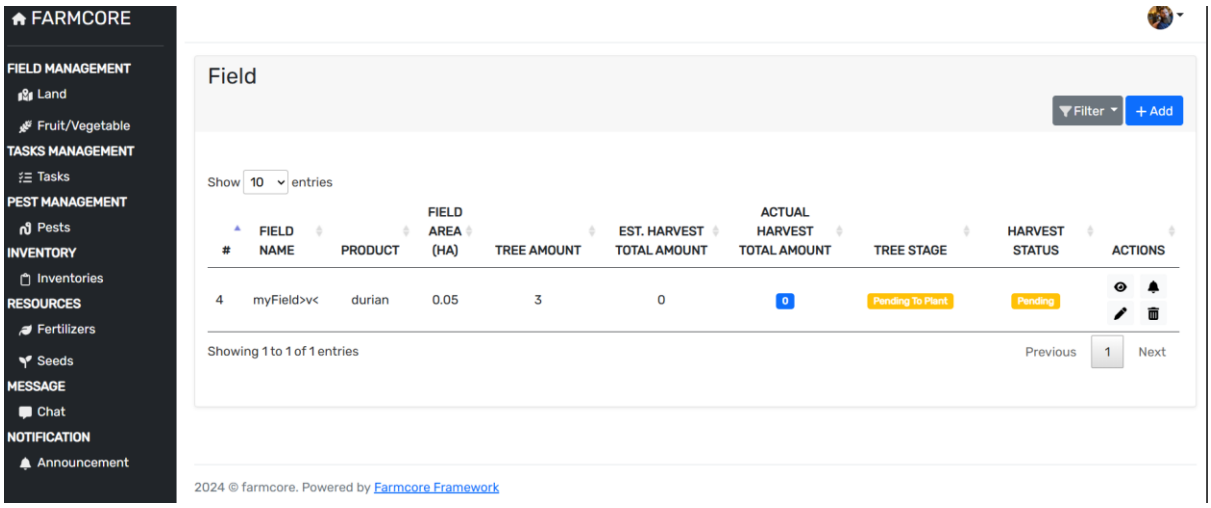


Figure 5.3.10 System Operations of View Field

Figure 5.3.10 above shows the system operations of the view field. The farmer can filter the list by the field name. If they want to add a new field, click on the “Add” button at the top right corner of the card. Inside, the action column, the farmer can choose to select the “view” button to view the dashboard with the “eye” icon, the “announcement” button to make an

announcement related to this field with the “bell icon, select the “update” button to update field information with a pencil icon and select “delete button” to delete the field from the land with the “rubbish bin” icon.

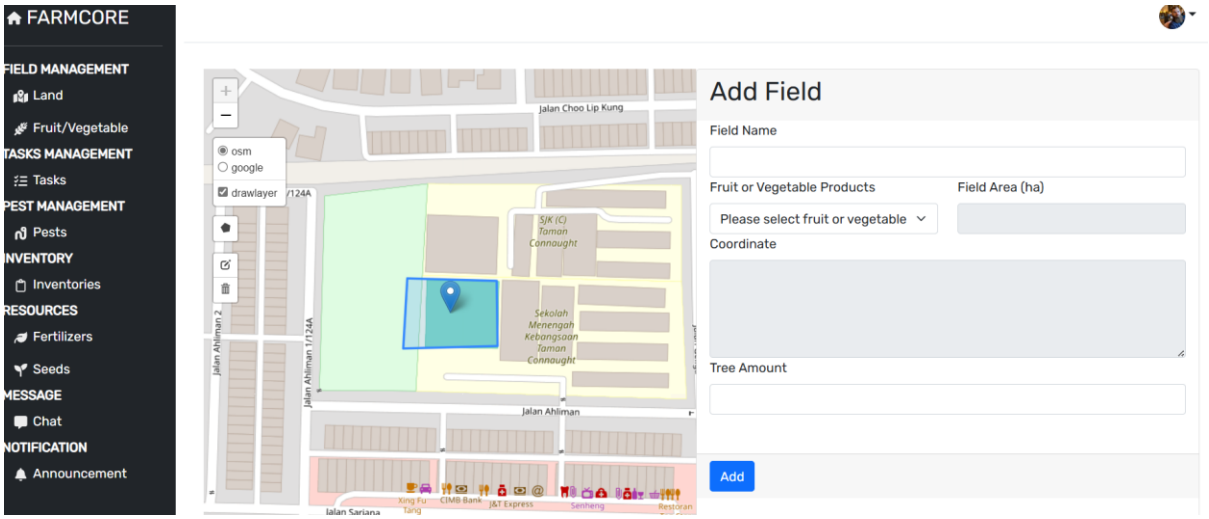


Figure 5.3.11 System Operations of Add Field

Figure 5.3.11 shows the system operations of the add field. In this page, it will point to the land that the farmer chose to plot field inside the land. If the farmer accidentally plotted outside the land area, it will pop out an error message and ask the farmer to plot again as Figure 5.3.12 below.

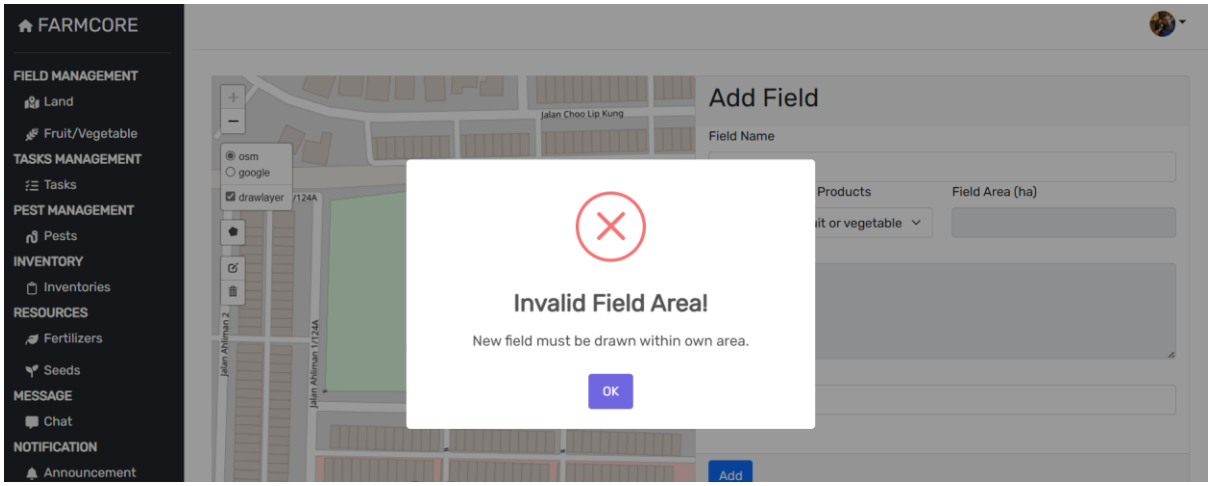


Figure 5.3.12 Error Message of Invalid Plot Area

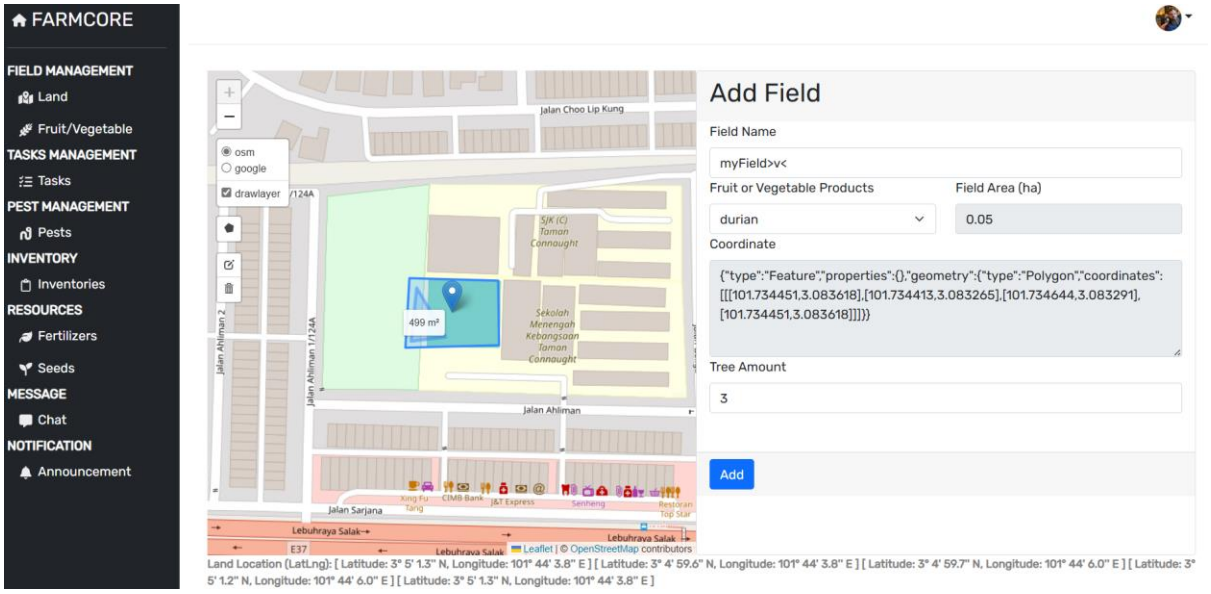


Figure 5.3.13 Valid Plotting Field Area

Figure 5.3.13 shows if the farmer plotted the area within the land, the farmer can continue to fill in the information of the field. After successfully added, it will redirect back to the main page of the field module and it will show all the fields with the newly added one.

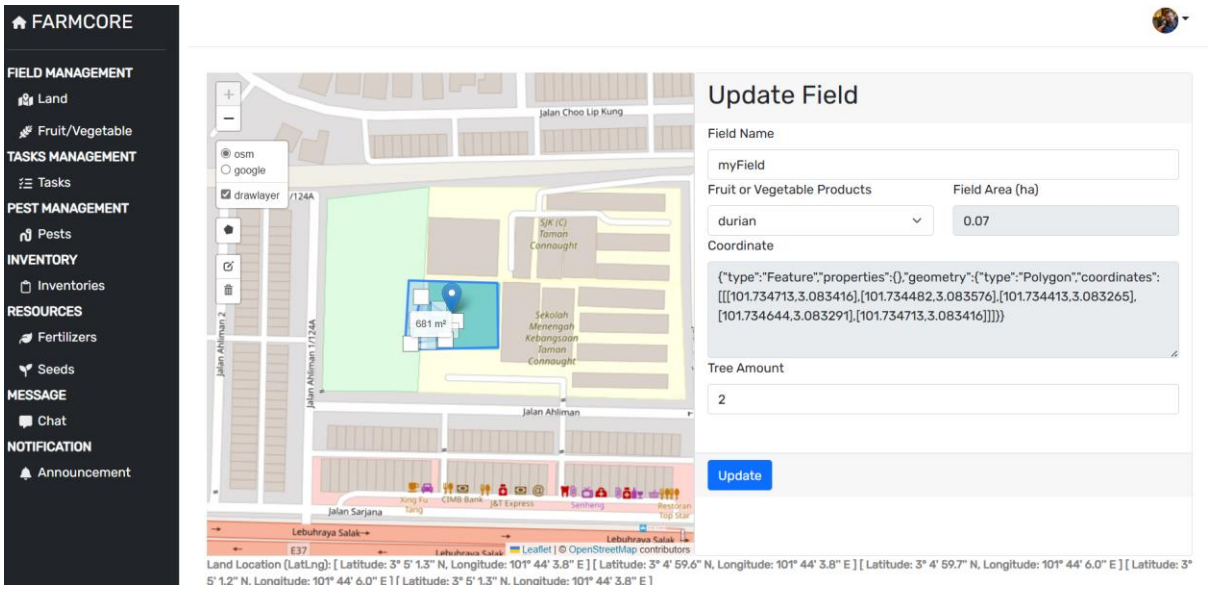


Figure 5.3.14 System Operation of Update field

Figure 5.3.14 shows the system operation of updating the field. In the map only the selected field is allowed to be dragged and re-adjust the area. The marker is only to show the farmer is using the selected land. Same as plotting a new field, if the farmer dragged the field area outside the land boundary, an error message is shown and asked to re-adjust again shown as in figure 5.3.15 below.

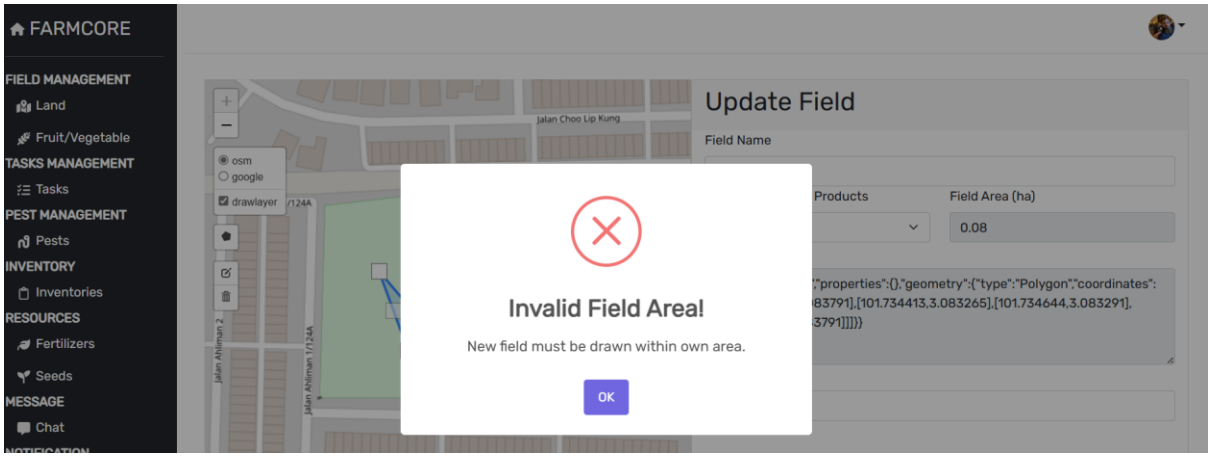


Figure 5.3.15 Error Message of Invalid Drag Field Area in Update Field Page

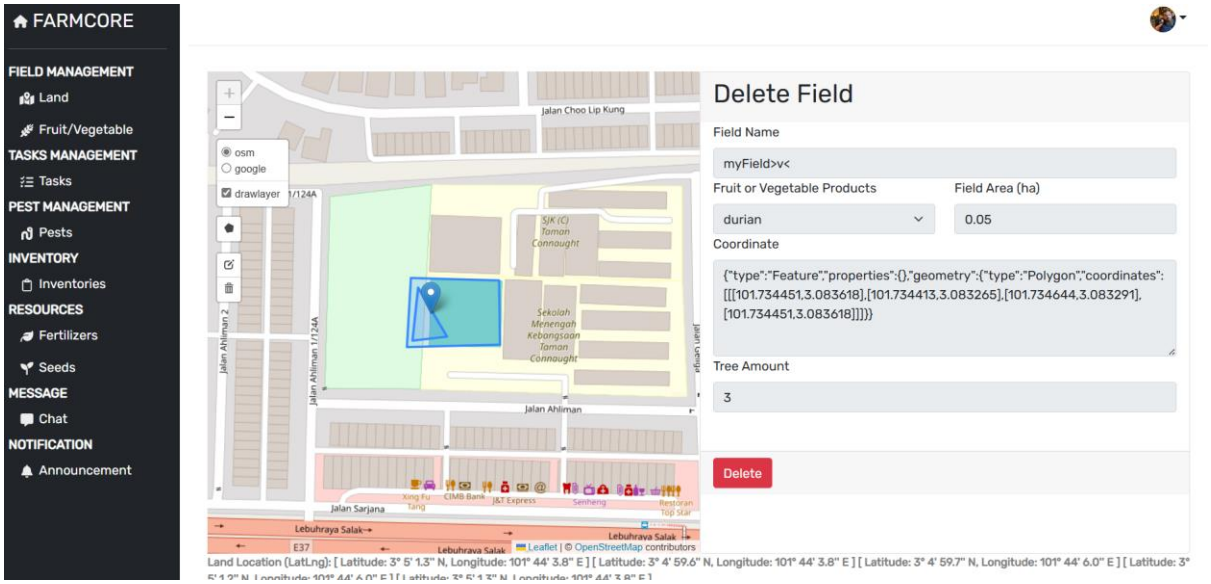


Figure 5.3.16 System Operations of Delete Field

The figure above shows the system operations of deleting the field. The marker will point out the field and display all the information of the field as a review before they confirm to delete it. After successfully deleted, it will redirect back to the main page of field module.

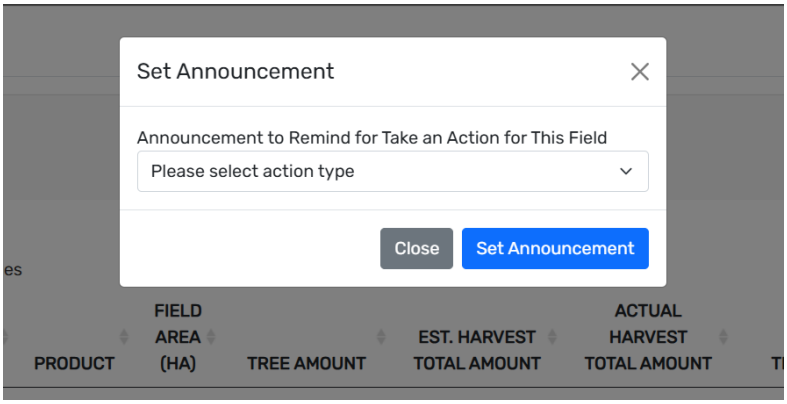


Figure 5.3.17 System Operations of Set Announcement for Selected Field

Figure 5.3.17 above shows the system operations of set announcement for the selected field. The farmer can set the announcement as a reminder to remind the farmer which action can be started for this field. This announcement will be posted on the announcement page.

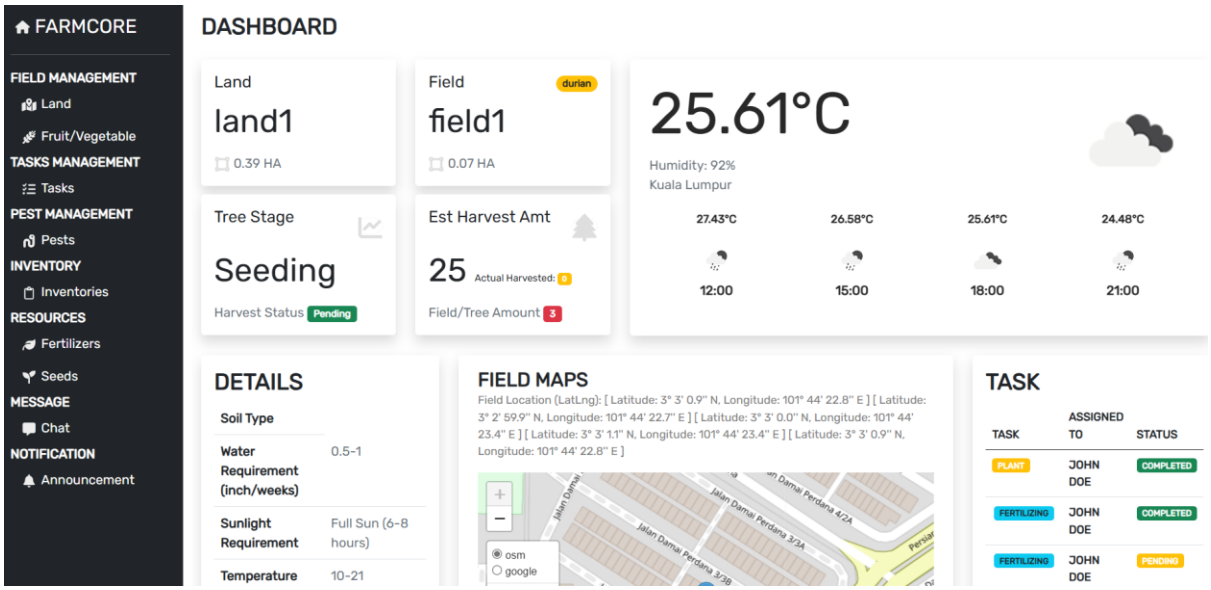


Figure 5.3.18 System Operations of Field Dashboard

Figure 5.3.18 above shows the system operations of the field dashboard. After selecting the "View Dashboard" button from the view field page (figure 5.3.10), it will redirect to this dashboard page for the farmer to view all the information related to the field in detail.

View, Add, Update, Delete Fruit or Vegetable Planting Data

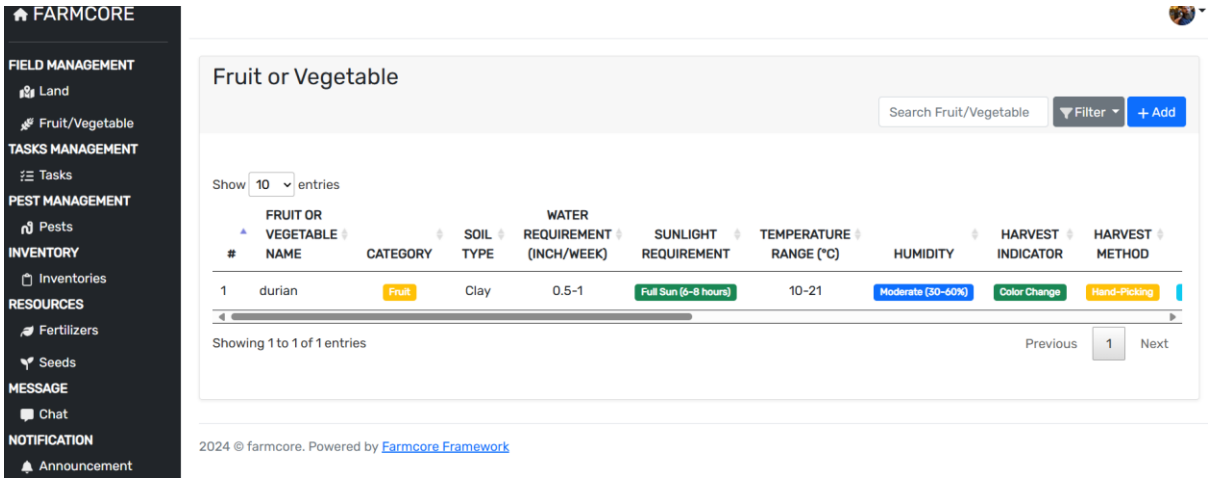


Figure 5.3.19 System Operations of View Fruit or Vegetable Planting Data

Figure 5.3.19 shows the system operations of view fruit or vegetable planting data for the farmer to manage. The farmer can search for the fruit or vegetable in the search column, and filter the list by category either fruit or vegetable. The farmer can also add new planting data by selecting the "Add" button. To update or delete, the farmer has to scroll the table to the end

will show the “Update” and “Delete” buttons in the action column. All of the fruits or vegetables recorded in this table will be a selection for the farmer to select which product needs to be planted in the field.

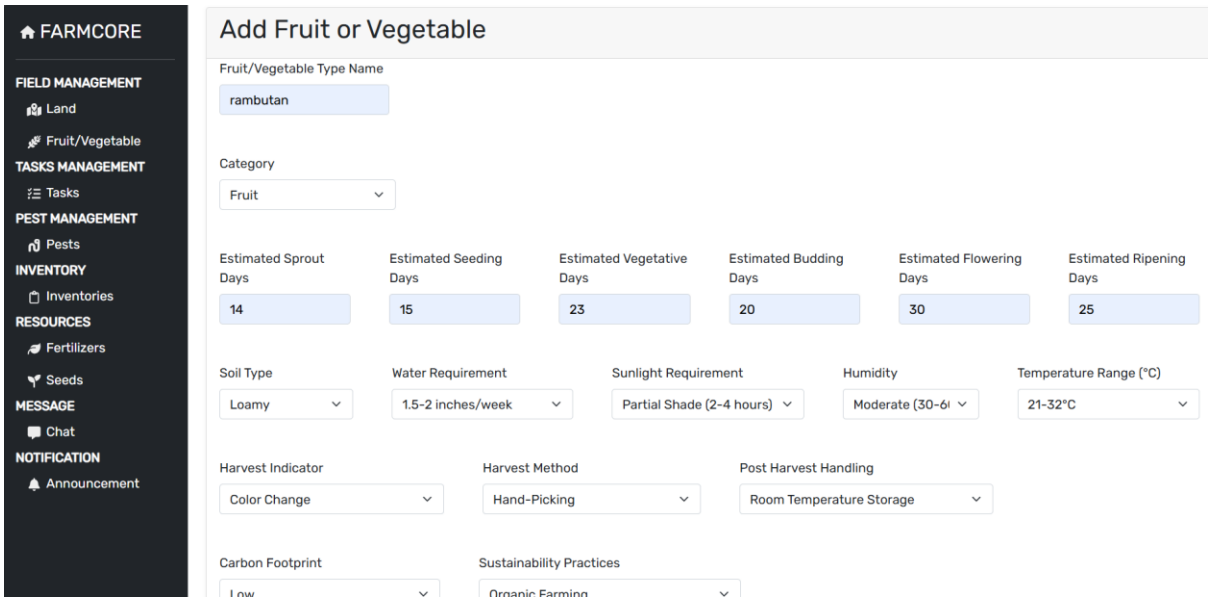


Figure 5.3.20 System Operations of Add New Fruit or Vegetable Planting Data

Figure 5.3.20 shows the system operations of adding new fruit or vegetable planting data. The farmer is required to enter all the data into this form. After entering all data, click on the “Add” button at the bottom of the card. If is successfully added, will redirect back to the main page of the fruit or vegetable planting data module and the newly added product planting details will be included in the list.

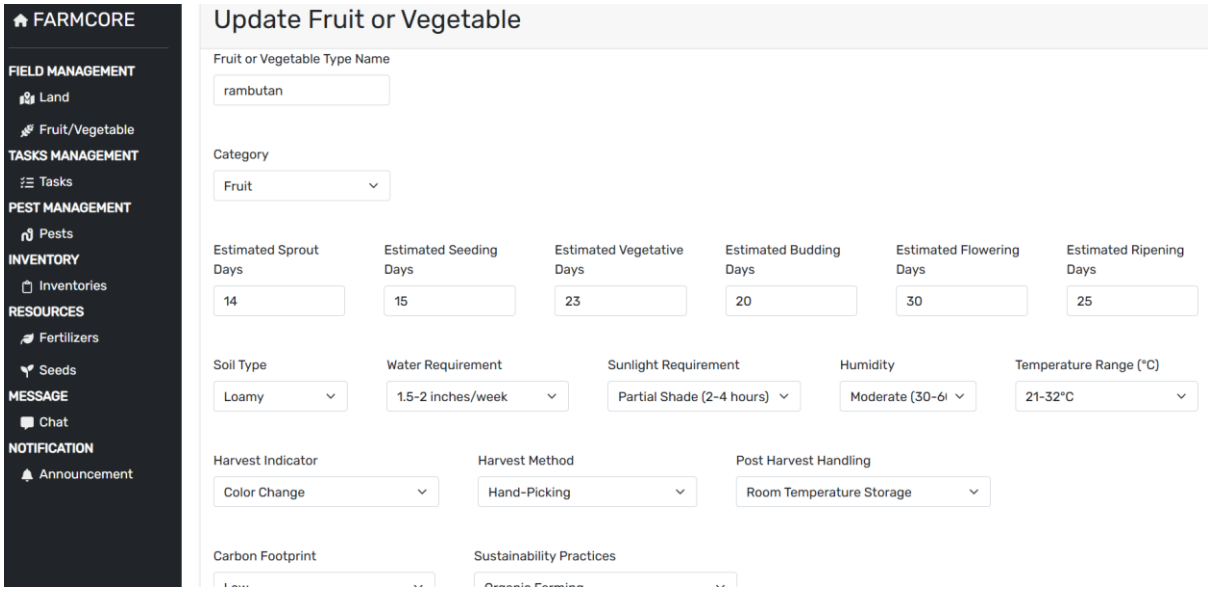


Figure 5.3.21 System Operations of Update Fruit or Vegetable Planting Data

Figure 5.3.21 shows the system operations of updating fruit or vegetable planting data. When the farmer selects to update the particular fruit or vegetable planting data, it will display all the last updated data that was stored by the farmer previously. The farmer can change any data and click on the "Update" button at the bottom of the card. Once updated successfully, it will be redirected back to the main page of the fruit or vegetable planting data module and the data will be updated.

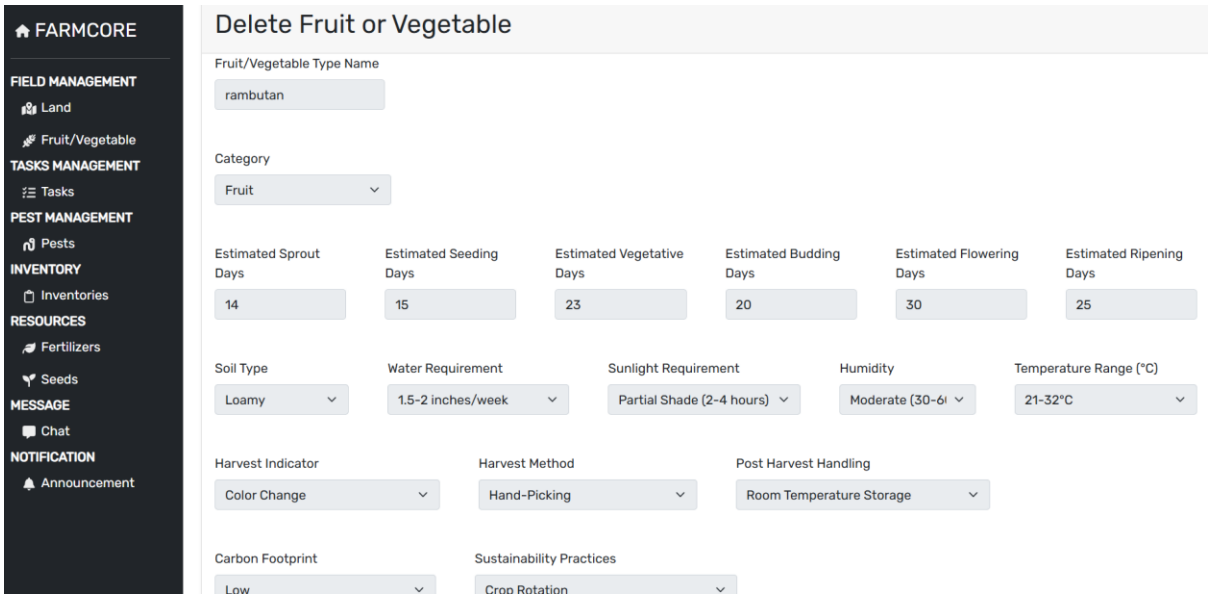


Figure 5.3.22 System Operations of Delete Fruit or Vegetable Planting Data

Figure 5.3.22 above shows the system operations of deleting fruit or vegetable planting data. Similar to figure 5.3.21 select the fruit or vegetable product wanted to delete it from the list, select the "Delete" button in the "Action" column. Once the farmer is redirected to Figure 5.3.22, the farmer can only review the data before confirming to delete and is not allowed to edit any data. After clicking the "Delete" button at the bottom of the card, it will redirect back to the main page of the fruit or vegetable planting data and the deleted product will no longer show in the list.

View, Add, Update, Delete Task, View Dashboard

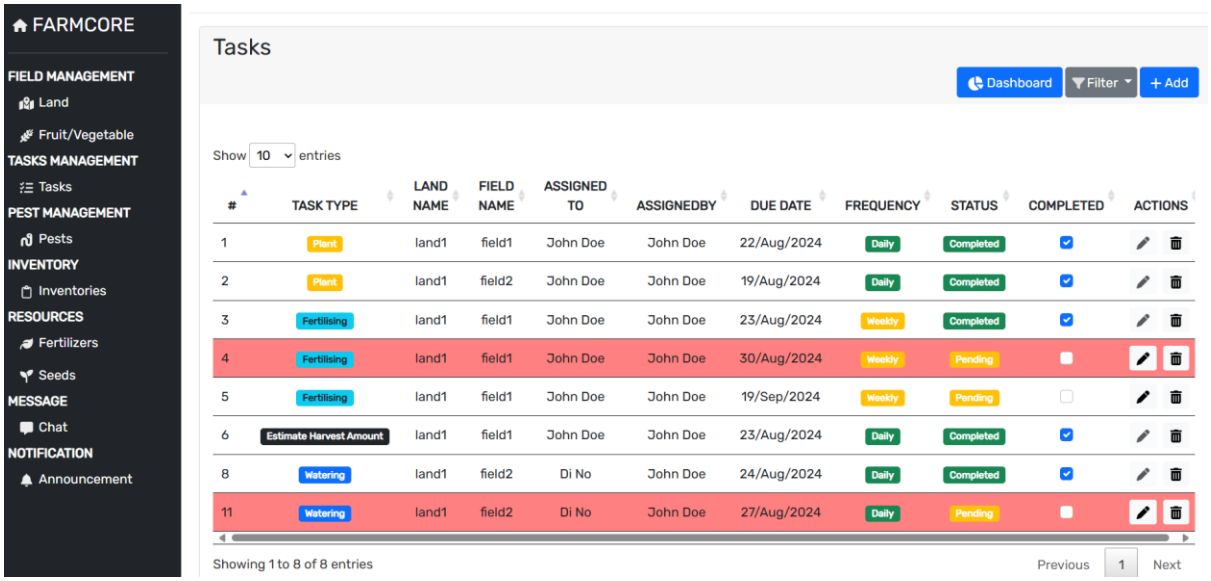


Figure 5.3.23 System Operations of View Task Page

Figure 5.3.23 above shows the system operations of the view task page. The farmer can check all the tasks that have been assigned and their status. If the web application finds the task is overdue, it will highlight it. Besides, every time the farmer logs in or returns to the homepage, it will pop out an alert message to ask the farmer to complete it as soon as possible. The farmer can add new tasks by selecting the "Add" button at the right upper corner of the card. To update or delete, the farmer has to select the "Update" or "Delete" button in the "Actions" column. If want to view the overall task analysis, the farmer can select the "Dashboard" button beside the filter button. The farmer can filter the list by status, field, or frequency. If the farmer completes the task and checks on the checkbox in the completed column, the status will change to "Completed".

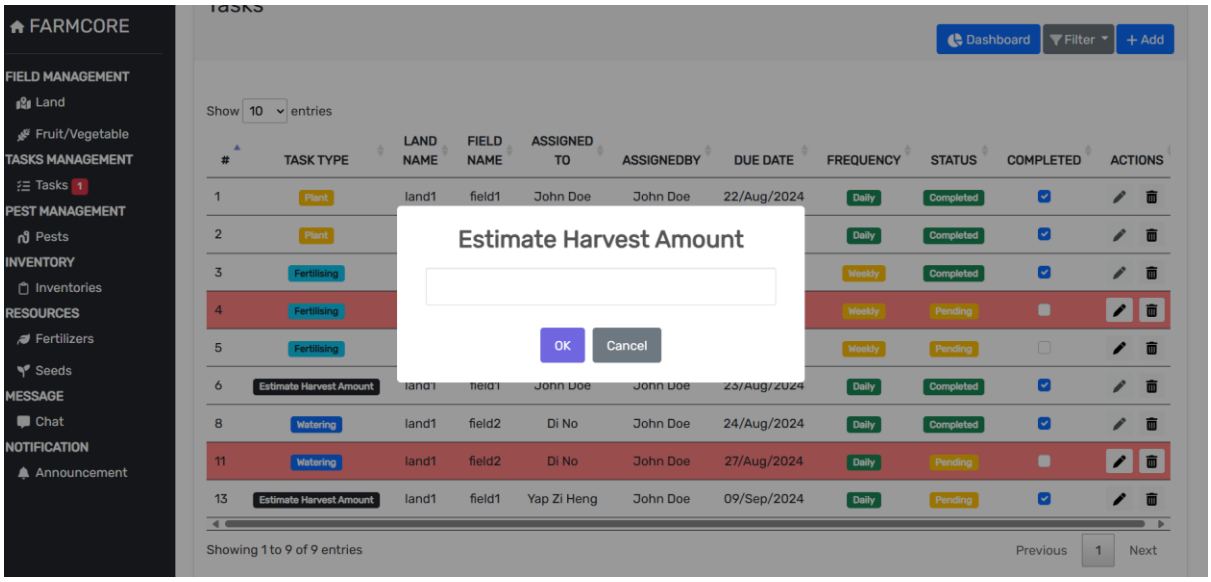


Figure 5.3.24 Update Completion Status and Enter Estimate Harvest Amount

Figure 5.3.24 above shows that the farmer asked to enter the estimated harvest amount of the products currently cultivated in the field before completely updating the status of the task. This modal will be opened when the task type is “Estimate Harvest Amount” After entering the amount, the web application will recalculate the total estimated amount that will be entered into the inventory as well as determine the amount of the products that can be harvested by which month and displayed in a line chart at the homepage. The same goes for the “Harvesting” task type, which will also open a modal for the farmer to enter the total amount that has been harvested and update the quantity in the inventory. For the other task, the type will not show the modal and just update the task completion status, but for the “planting” task type, once the checkbox is checked, the web application will start to calculate and determine the product’s growth stage.

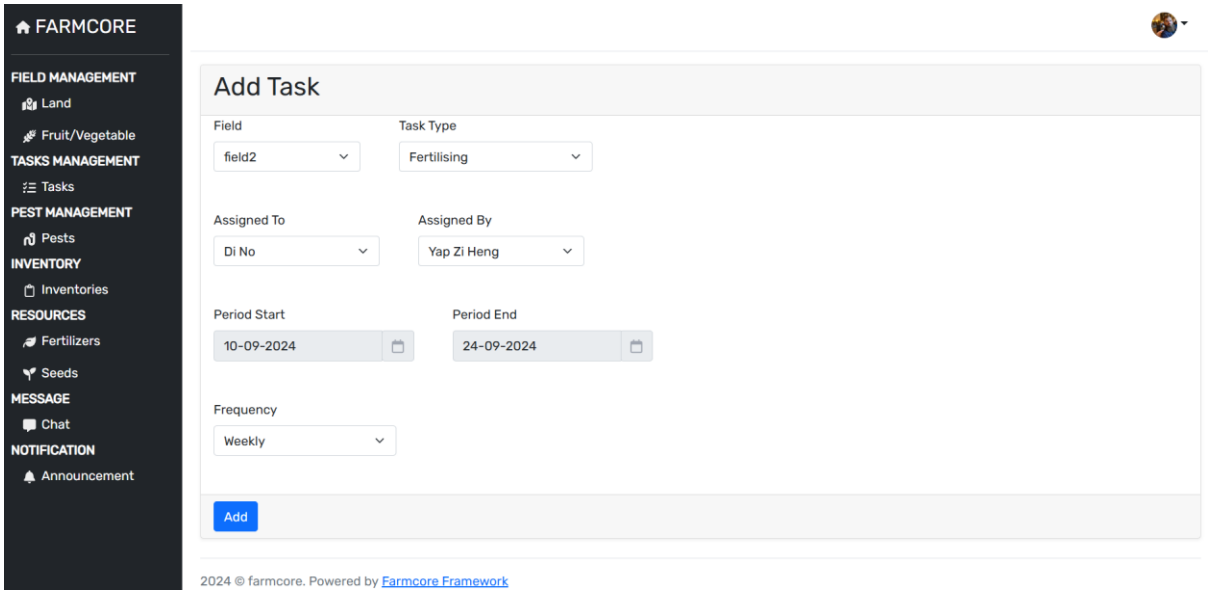


Figure 5.3.25 System Operations of Add Tasks

Figure 3.3.25 shows the system operations of add tasks. The farmer is required to select a field, task type, an assignee, an assigner, select task period for the assignee, and the task frequency. Once the “Add” button, the web application will calculate the due date of the task. By using this figure as an example, the period of the task assigned to the assignee is from 10th September 2024 to 24th September 2024 which is around 3 weeks and the frequency of the task is weekly; thus, this web application will add 3 pending tasks for this fertilizing task for the assignee with a different due date. After successfully added, it will redirect back to the main page of the task module. The result is shown in the figure below:

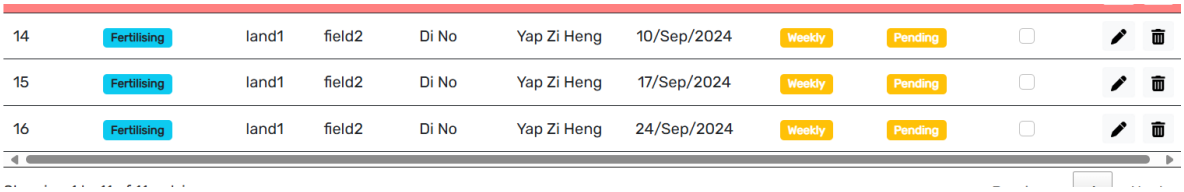


Figure 5.3.25.1 Output of Add Tasks

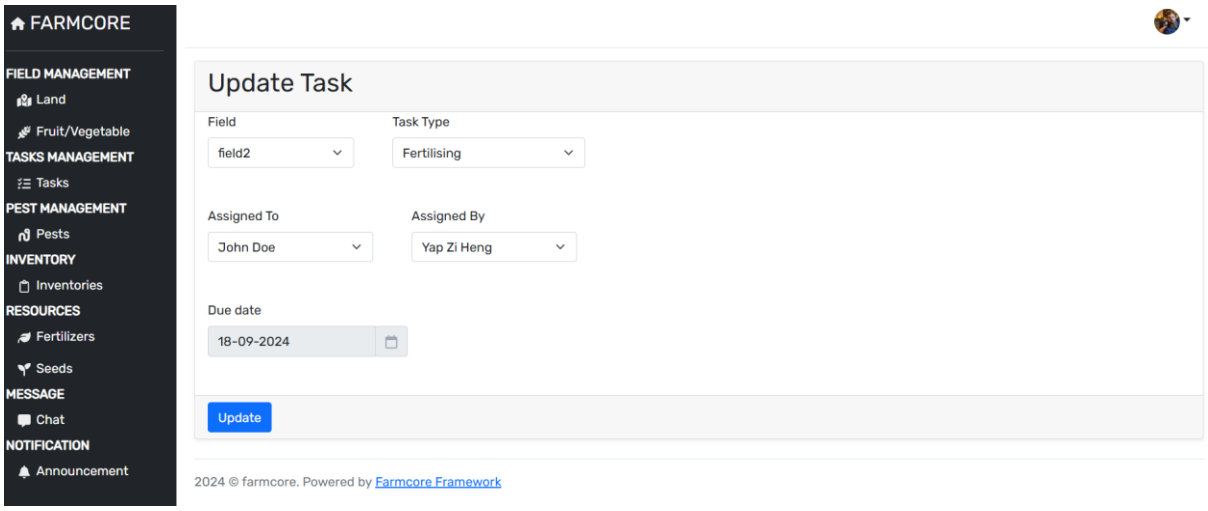


Figure 5.3.26 System Operations of Update Task

Figure 5.3.26 shows the system operations of the update task. The farmer can update the task that is pending to complete by clicking on the "Update" button in the "Action" column. Different from Figure 5.3.26 have the task frequency and task period, this update task does not have the task frequency selection and task period. The farmer can only choose to change the field, task type, assignee, assigner, or due date of the task. After being updated successfully will redirect back to the main page of the task module.

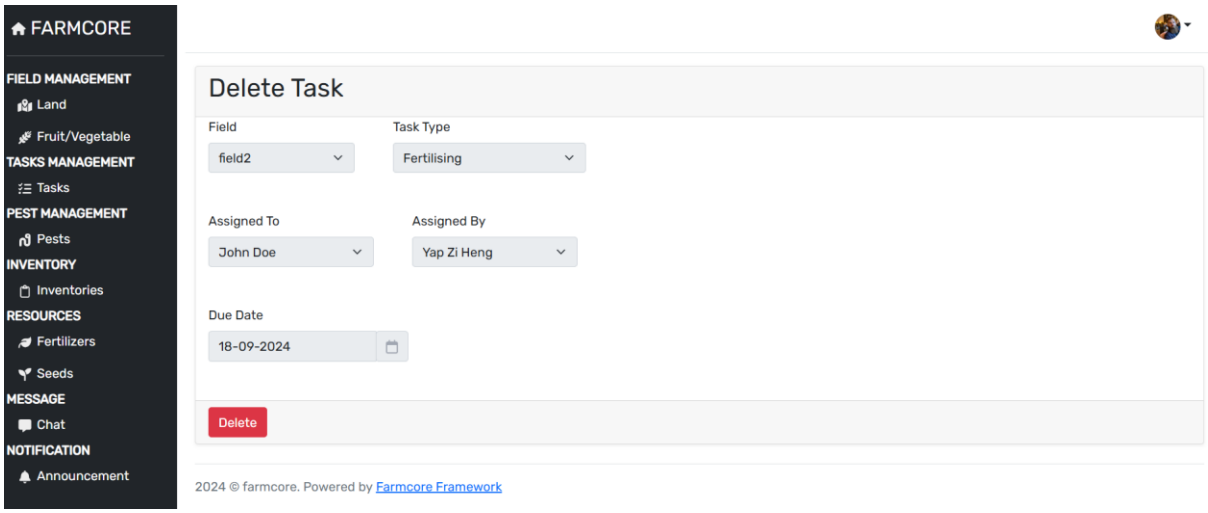


Figure 5.3.27 System Operations of Delete Task

Figure 5.3.27 shows the system operations of the delete task. The farmer selects the task wish to delete by clicking on the "Delete" button in the "Action" column. Then the farmer will

redirect to this delete task page to review the task information before confirming to delete it. After successfully deleted, it will redirect back to the task page and the deleted task has been removed.

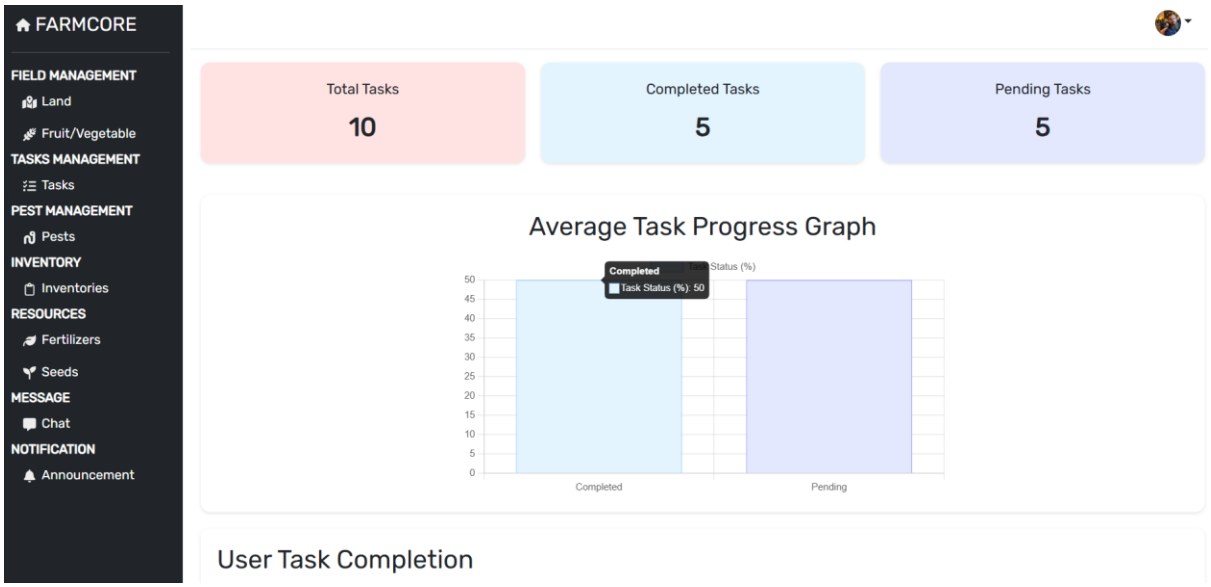


Figure 5.3.28 System Operations of Task Dashboard

Figure 5.3.28 above shows the system operations of the task dashboard. The farmer can access this dashboard by clicking on the "Dashboard" button on the task page (figure 5.3.22). The farmer can view the visualized task analysis in this dashboard. This dashboard included total tasks, completion status, farmer's progression, task type distribution, and frequency of task distribution.

View, Add, Update, Delete Pests Issue

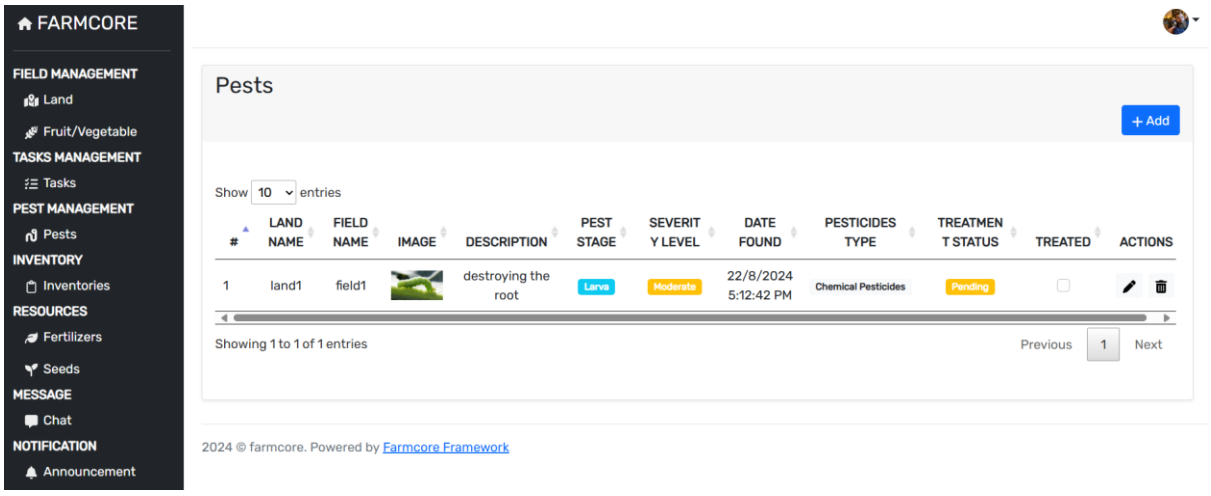


Figure 5.3.29 System Operations of View Pest

Figure 5.3.29 above shows the system operations of view pests. The farmer can manage the issue in this module. The farmer can check on the checkbox in the "Treated" column if the

issue is treated and the status will change to “Treated”. The farmer can also either add, update, or delete the issue. Clicking on the “Add” button, will redirect to the “Add Pest” page; clicking on the “Update” button in the “Action” column will redirect to the “Update Pest” page; clicking on the “Delete” button in “Action” column will redirect to “Delete Pest” page.

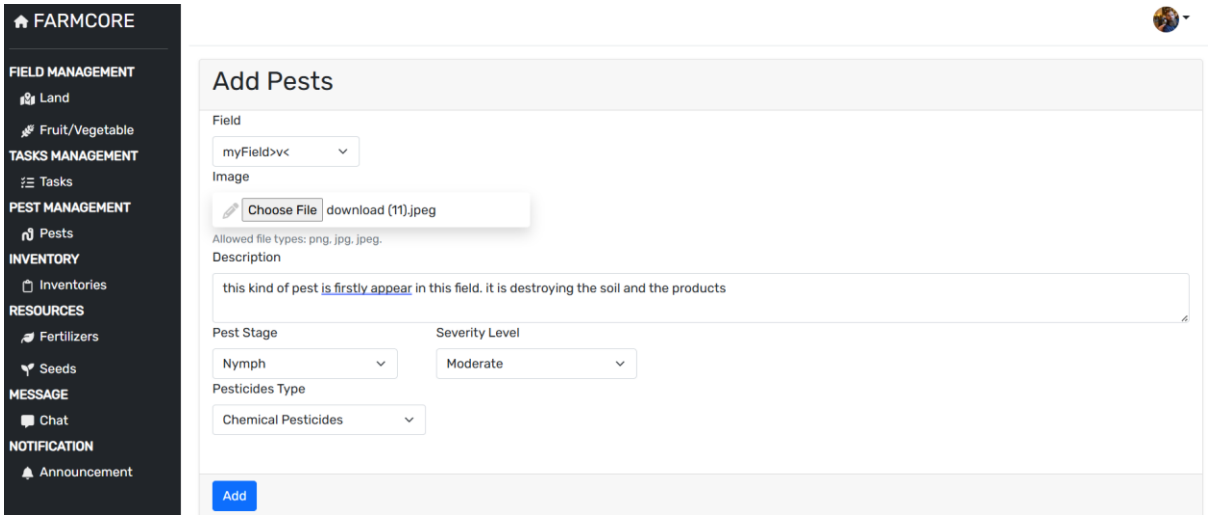


Figure 5.3.30 System Operations of Add Pests

Figure 5.3.30 above shows the system operations of add pests. The farmer has to capture the pests and upload them. The farmer needs to select the field, provide a description, the current stage of the pest, estimate its severity level to the field, and what pesticide type needs to be used to handle the issue. After being added successfully, it will return to the main page of the pests module (figure 5.3.29), and the newly added issue is shown.

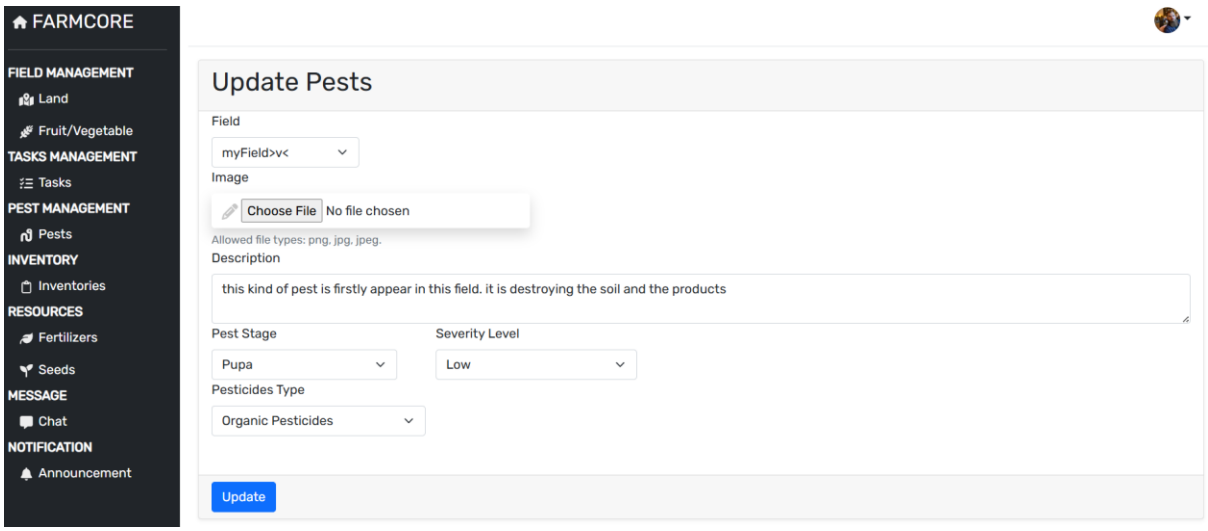


Figure 5.3.31 System Operation of Update Pests

Figure 5.3.31 above shows the system operations of updated pests. The farmer can select which issue they want to update it. When redirected to this update pest page, all the last updated information on the pest issue will be auto-filled into every column except for the image. If there

is no update for the image, the image column can remain empty on this update page. However, the other information must be not empty. After successfully updating, it will redirect to the main page of the pests module (figure 5.3.29) and the newly updated data is shown.

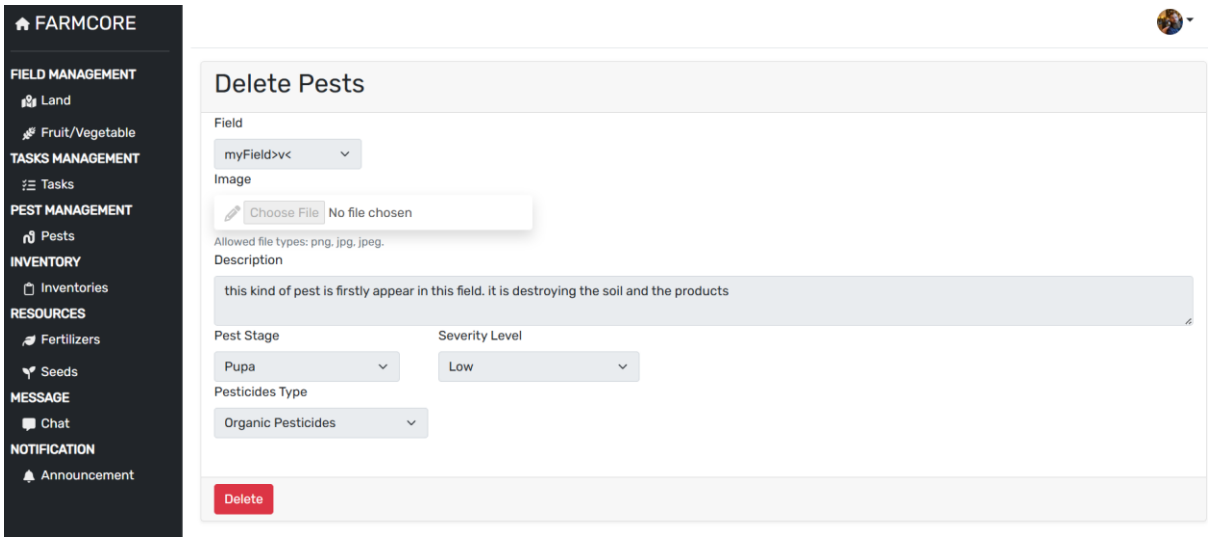


Figure 5.3.32 System Operations of Delete Pests

Figure 5.3.32 above shows the system operations of deleting pests. Similar to updating pests, the last update of pest information will auto-fill into every column. However, the farmer is not allowed to edit the information. The farmer can only review all the information before deleting it. After deletion, it will redirect to the main page of the pests module and the deleted pest's issue has been removed permanently.

View, Add, Update, Delete, and Export Inventory

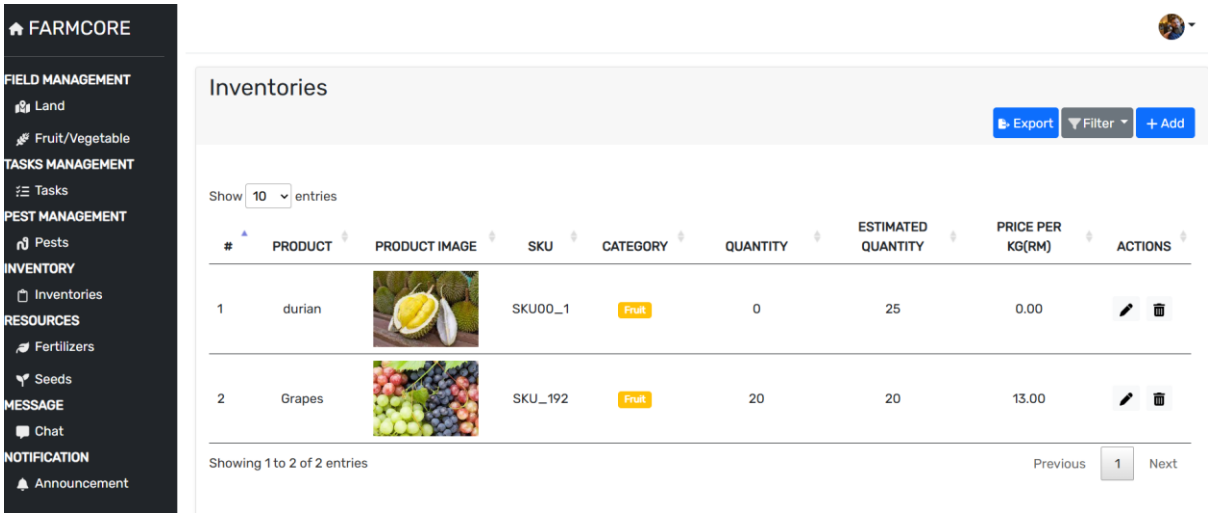


Figure 5.3.33 System Operations of View Inventories

Figure 5.3.33 above shows the system operation of inventory. On this page, the farmer can view the product information such as product name, image, SKU number, category, quantity and estimated quantity, and price. The farmer can export the inventory data in a CSV file. The

farmer can filter this inventory by the product category either vegetable category or fruit category. The farmer has to click on the “Export” button at the card header and it will open a modal and ask the farmer to select the category that needs to be exported in the CSV file as shown in the figure 5.3.34 below. The farmer can also select the “Add” button to add a new product to this inventory, select the “Update” button to update the information about the product, or select the “Delete” button to delete the data. Both of the “Update” and “Delete” buttons are in the “Actions” column.

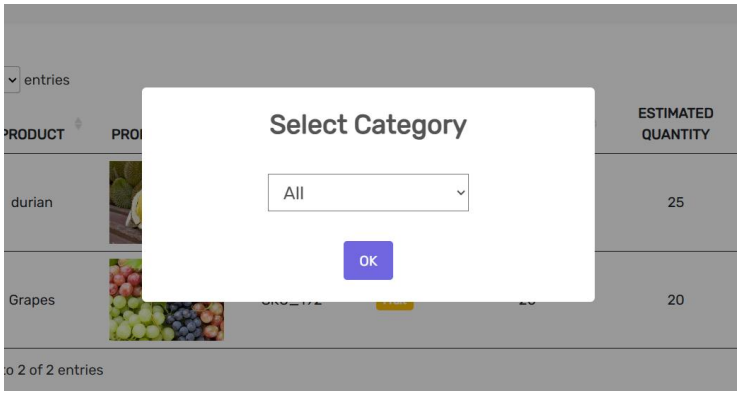


Figure 5.3.34 System Operations of Export Inventory

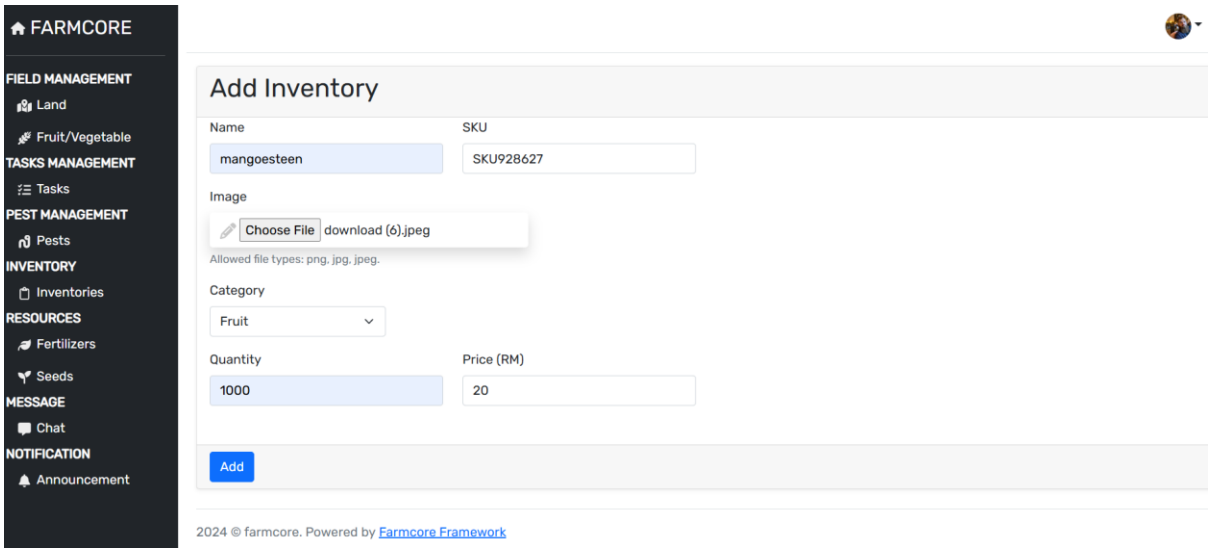


Figure 5.3.35 System Operations of Add Inventory

Figure 5.3.35 above shows the system operations of add inventory. The farmer can enter new agricultural products manually. The farmer is required to enter all the data of new agricultural products such as name, SKU number, image, category, quantity, and price. After clicking on the “Add” button, the new agricultural products will be added to the inventory. It will then redirect back to the main page of inventory module (figure 5.3.33) and newly added products will be shown.

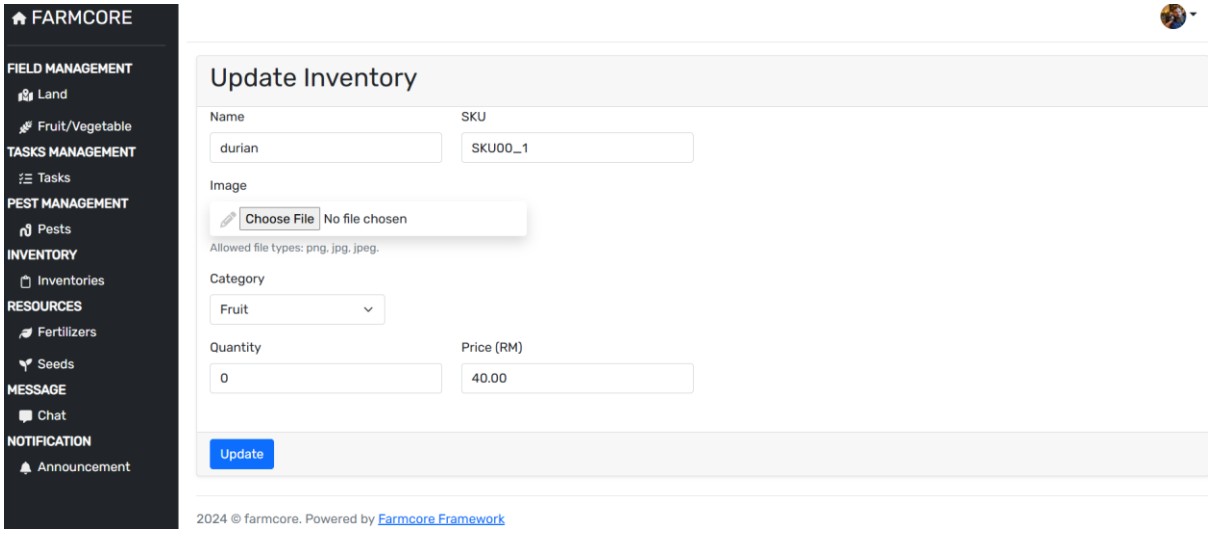


Figure 5.3.36 System Operations of Update Inventory

Figure 5.3.36 above shows the system operations of updated inventory. The farmer can select any of the agricultural products that wants to update. After redirecting to this updated inventory page, the information is filled, and the farmer is allowed to change the information. If the farmer does not want to update the image of the product, the farmer can keep the image input column empty. However, the other input column must be filled. After successfully updating, it will redirect back to the main page of the inventory module (figure 5.3.33), and the product data has been updated in the list.

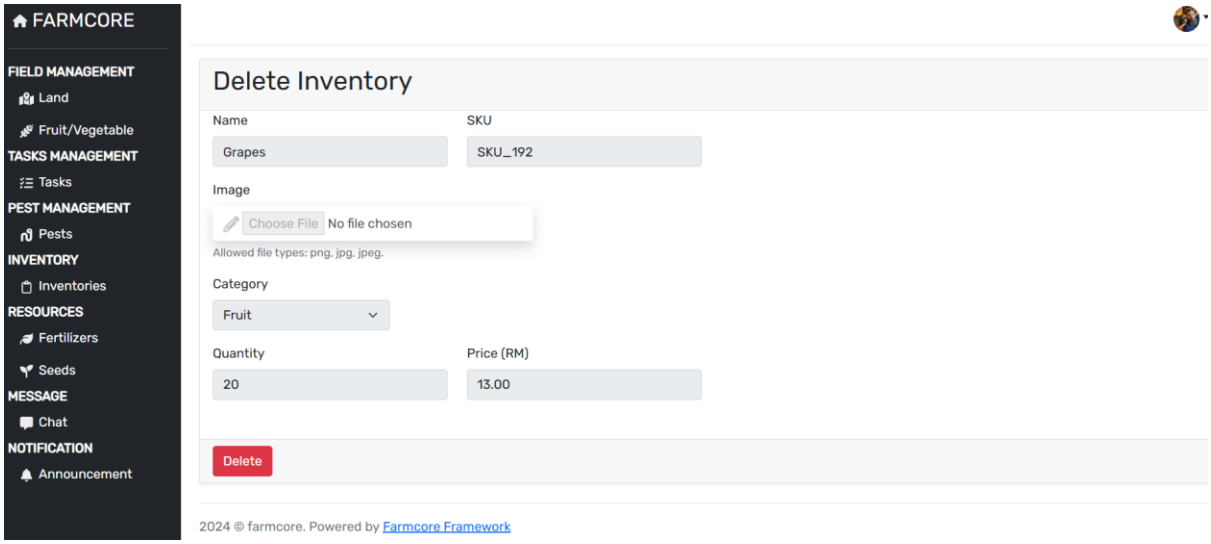


Figure 5.3.37 System Operations of Delete Inventory

Figure 5.3.37 shows the system operations of deleting inventory. After selecting the agricultural product to delete, it will redirect to this deleted page. The farmer can only review the product data before deleting it. Once clicked on the “Delete” button at the bottom, it will

redirect back to the main page of the inventory module (figure 5.3.33) and the delete product has been removed.

View, Add, Update, Delete, Deduct, Replenish Fertilizers Resource

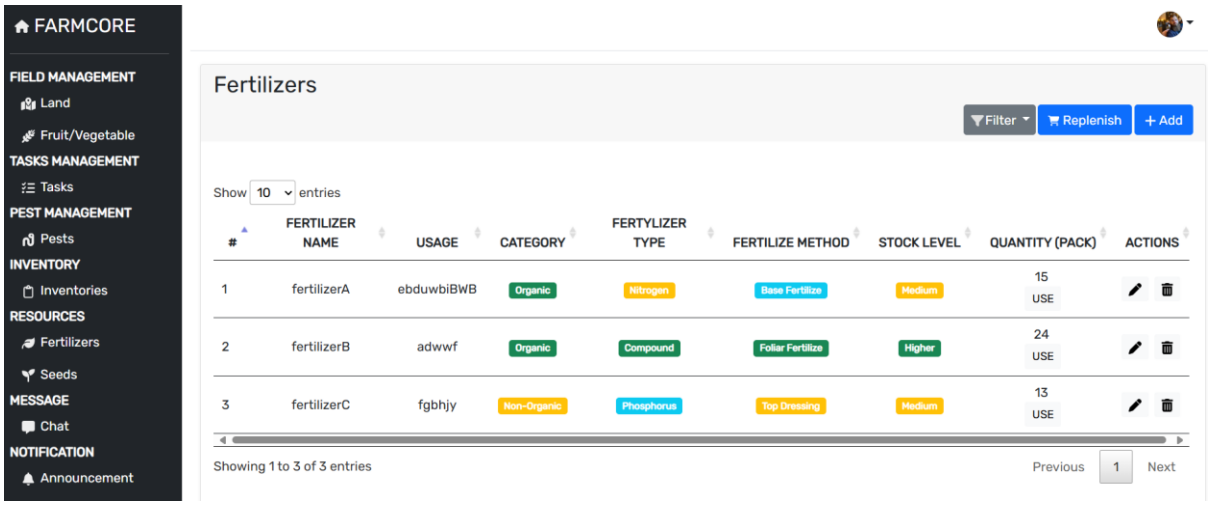


Figure 5.3.38 System Operations of Fertilizers Resource

Figure 5.3.38 above shows the system operations of fertilizers. On this page, the farmer can filter the list by applying a filter by category either organic or non-organic. The farmer can click on the "Add" button to add new fertilizers; then, it will redirect to add new fertilizers page. If want to update the fertilizer or fertilizer, the farmer can click on the "Update" or "Delete" button in the "Action" column and redirect to the update or delete page. To replenish the fertilizers, the farmer can click on the "Replenish" button and a modal will open to replenish the quantity; while to deduct the amount, the farmer can click on the "USE" button in the quantity column and a modal will open out to use to deduct the fertilizers quantity of pack.

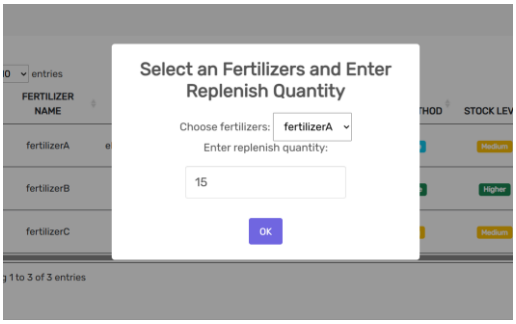


Figure 5.3.39 System Operations of Replenish Fertilizers

Figure 5.3.39 shows the system operations of replenishing fertilizers. The farmer selects which fertilizers have lower stock and enters the quantity needed to replenish. Then click on the "OK" button, and the quantity of the fertilizers will be increased.

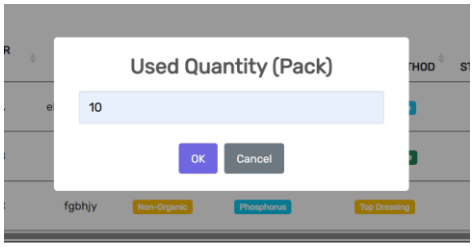


Figure 5.3.40 System Operations of Deduct Fertilizers Quantity

Figure 5.3.40 above shows the system operations of deducting fertilizers quantity. When clicked on the "USE" button, this modal will open and ask the farmer to enter the used fertilizers quantity. After clicking on the "OK" button, the quantity of the fertilizers will be deducted.

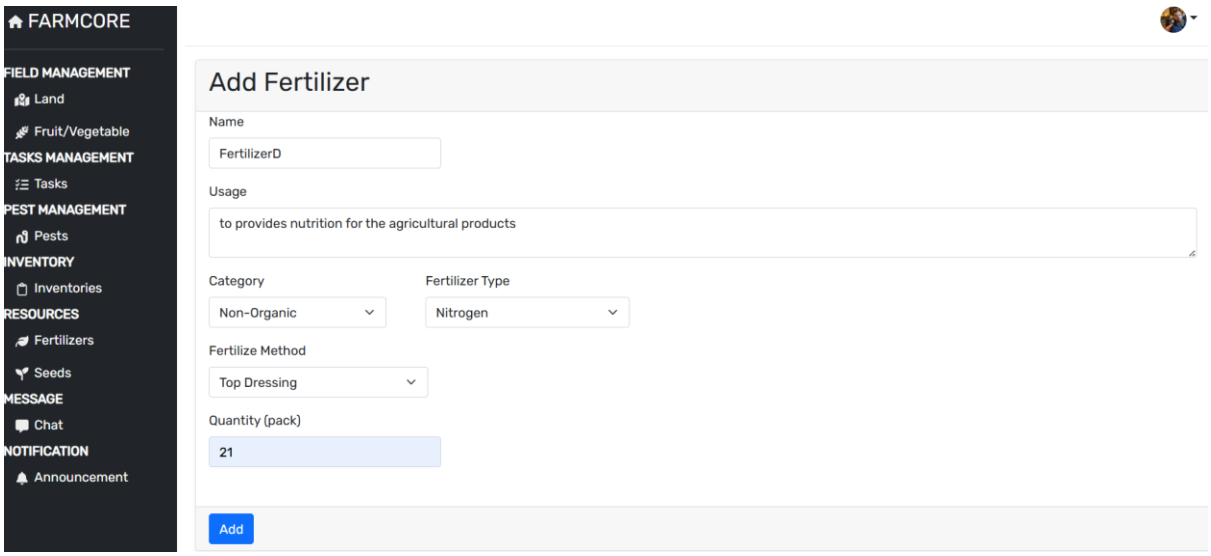


Figure 5.3.41 System Operations of Add New Fertilizer

Figure 5.3.41 above shows the system operations of adding new fertilizer into the resources. The farmer is required to fill up all the input columns before clicking on the "Add" button at the bottom. Once the farmer has confirmed to add and clicked on the "Add" button, it will redirect back to the main page of the fertilizer module (figure 5.3.38), and the newly added fertilizers are shown in the list.

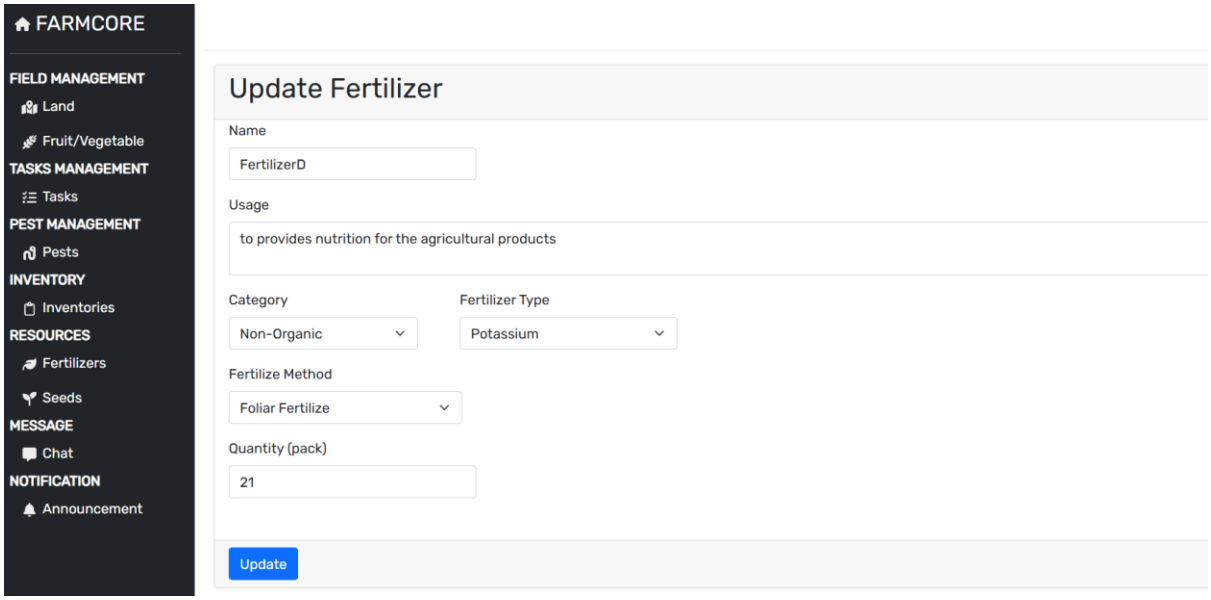


Figure 5.3.42 System Operations of Update Fertilizer

Figure 5.3.42 above shows the system operations of updated fertilizer. Once a fertilizer to updated, it will redirect to this update fertilizer page and all the last updates of fertilizer information have been auto-filled in every input column. The farmer can edit the information. After clicking on the "Update" button, it will be redirected back to the main page of the fertilizer module (figure 5.3.38) and the information of the fertilizers has been updated.

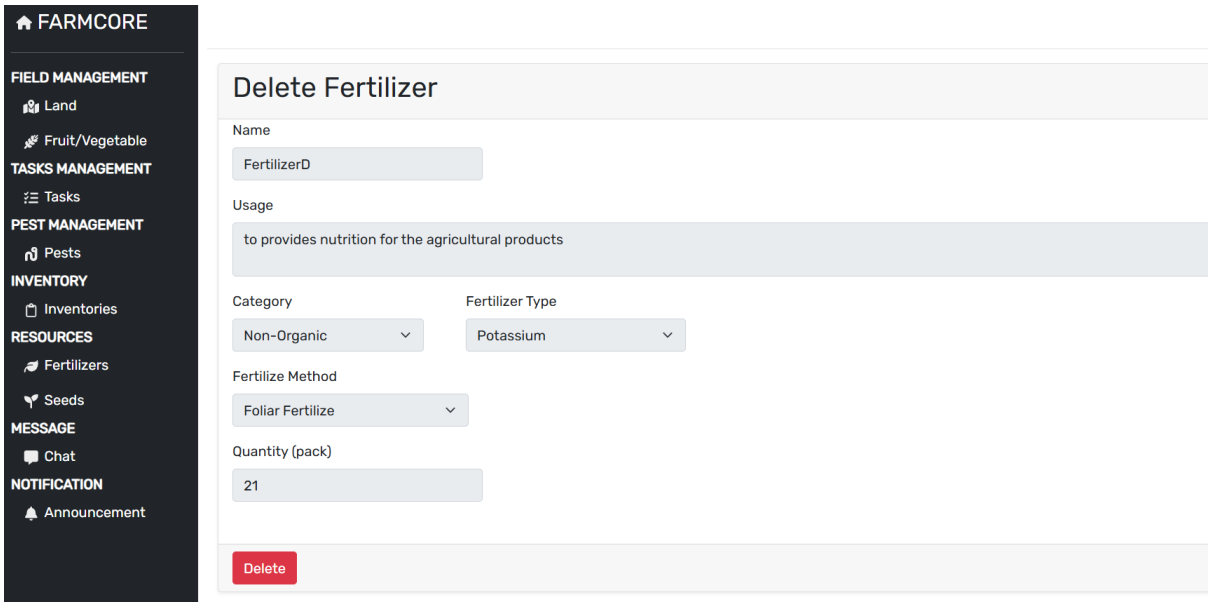


Figure 5.3.43 System Operations of Delete Fertilizers

Figure 5.3.43 above shows the system operations of deleting fertilizers. After selecting to delete the fertilizer, it will redirect to the delete fertilizer page. The farmer is not allowed to edit the information and can only review the information before confirm delete it. After deleting

this fertilizer, it will be redirected back to the main page of the fertilizer module (figure 5.3.38) and the fertilizers have been removed permanently.

View, Add, Update, Delete, Replenish, and Deduct Seeds Resource

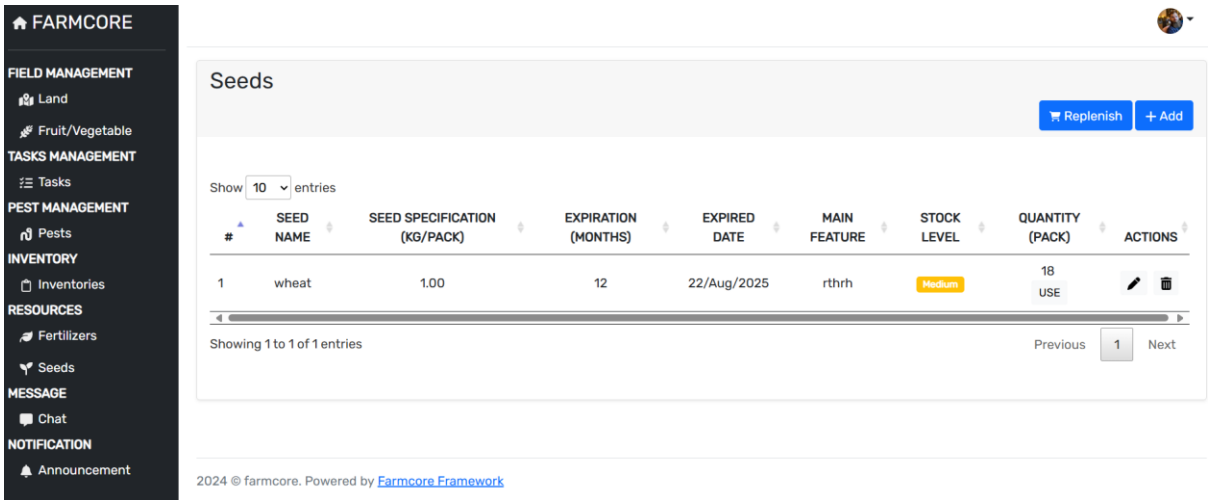


Figure 5.3.44 System Operations of Seeds Resource

Figure 5.3.44 above shows the system operations of seed resources. On this page, the farmer can conduct the replenishment and deduct the quantity of seeds. These two will open out a modal and ask the farmer to enter the quantity of seeds wants to replenish or deduct. The farmer is also allowed to add new seeds by clicking on the "Add" button at the top right corner of the card, and update or delete the seeds by clicking on the "Update" or "Delete" button in the "Action" column.

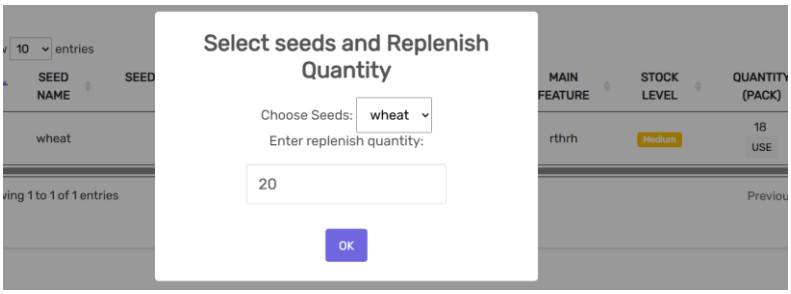


Figure 5.3.45 System Operations of Replenish Seeds

Figure 5.3.45 shows the system operations of replenishing seed quantity into the resource. The farmer will be asked to select which seeds wants to replenish and replenish the quantity. After clicking the "OK" button, the quantity of the seeds will be successfully replenished.

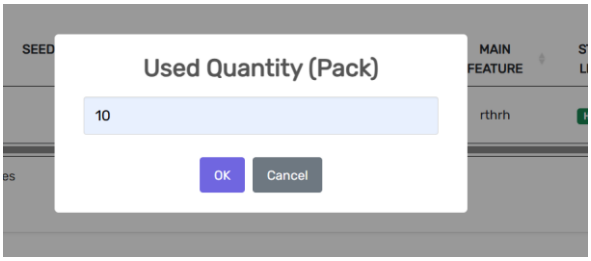


Figure 5.3.46 System Operations of Deduct Seeds Quantity

Figure 5.3.46 above shows the system operations of deducting the quantity of seeds from the resources. This modal will be opened and asked to enter the deduct quantity of the seeds after clicking on the "USE" button in the "quantity" column. By selecting the "OK" button, the modal will close and the quantity will be successfully deducted from the resource.

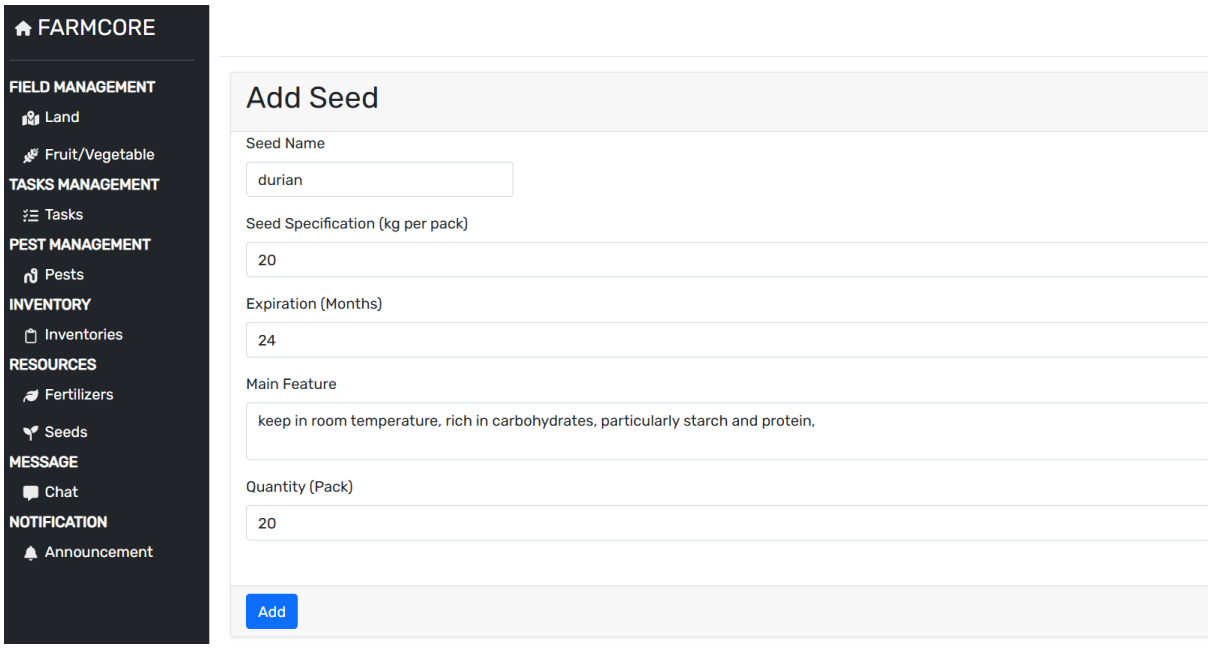


Figure 5.3.47 System Operations of Add Seed Resource

Figure 5.3.47 above shows the system operations of adding new seed resources. After clicking the "Add" button, the farmer will be redirected to this Add New Seed Resource page. To add new seed resources, the farmer must enter all the information about the seeds before confirming to add it to the seed resource. By clicking on the "Add" button after entering all the required information, it will redirect back to the main page of the seed module (figure 5.3.44) and the newly added seeds will be shown in the table.

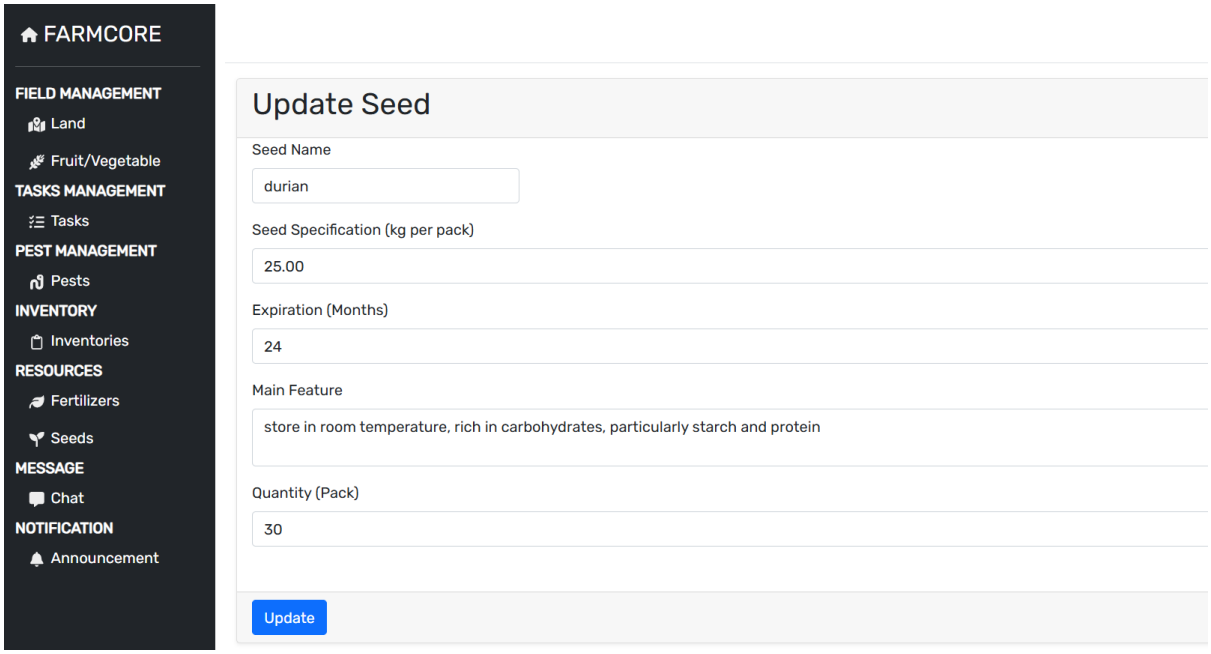


Figure 5.3.48 System Operations of Update Seed Resource

Figure 5.3.48 above shows the system operations of updating seed resources if the farmer wants to update it. After redirecting to this page, all the last updated information on the seeds is filled in every column automatically and the farmer can edit any column that needs to update. After editing and clicking the "Update" button, it will redirect back to the main page of the seeds module (figure 5.3.44) and the information on the particular seeds has also been updated in the list.

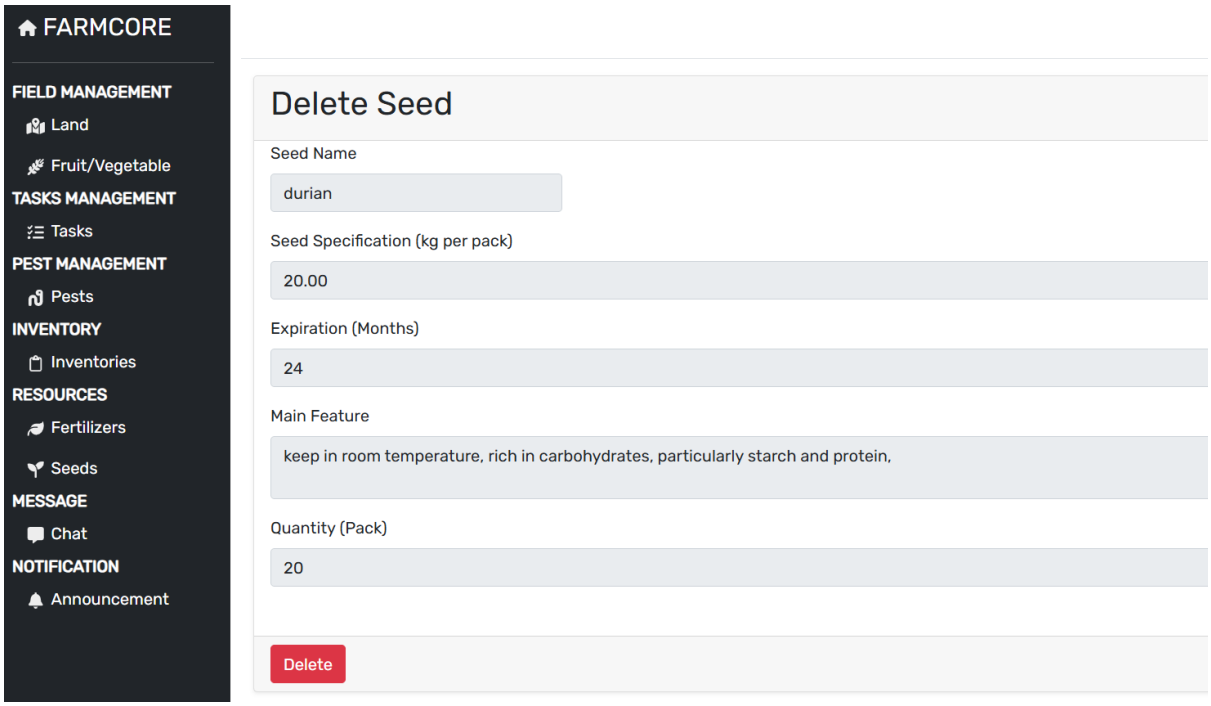


Figure 5.3.49 System Operations of Delete Seed Resource

Figure 5.3.49 shows the system operation of deleting seed resources. If the farmer selects to delete the particular seed; then, will redirect to this page. The farmer is not allowed to edit and can only review the information before confirming to delete it. After confirming deletion by clicking on the "Delete" button, it will redirect back to the main page of seeds module (figure 5.3.44) and the deleted seed has been removed.

Chat

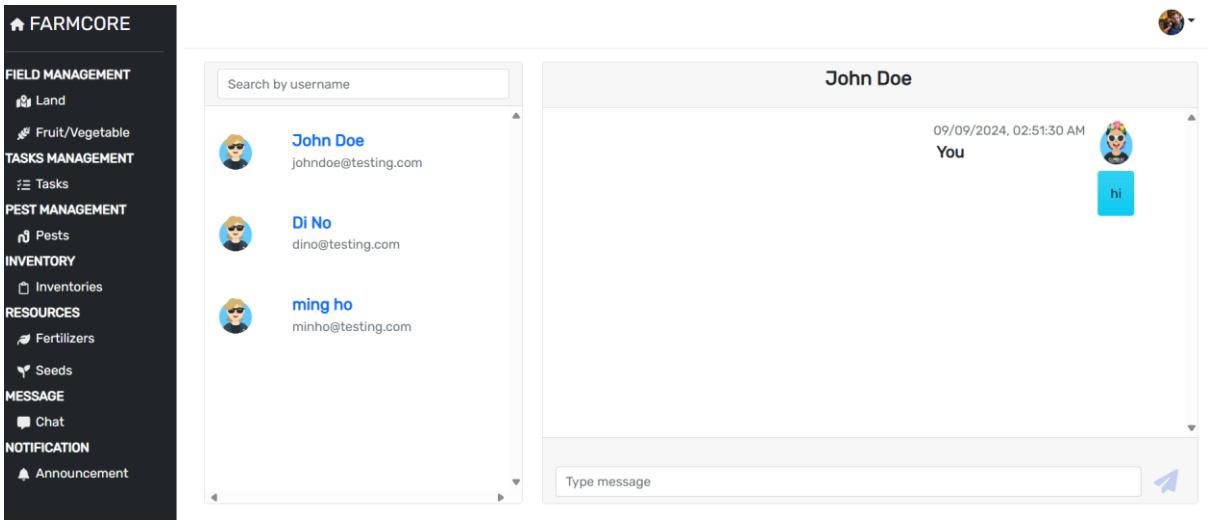


Figure 5.3.50 System Operations of Chat

Figure 5.3.50 shows the system operations of chat. The farmer can select any of the farmers that he or she wants to chat with from the user list besides the conversation card. The farmer can also search for a farmer by username.

View, Add, Update, Delete Announcement

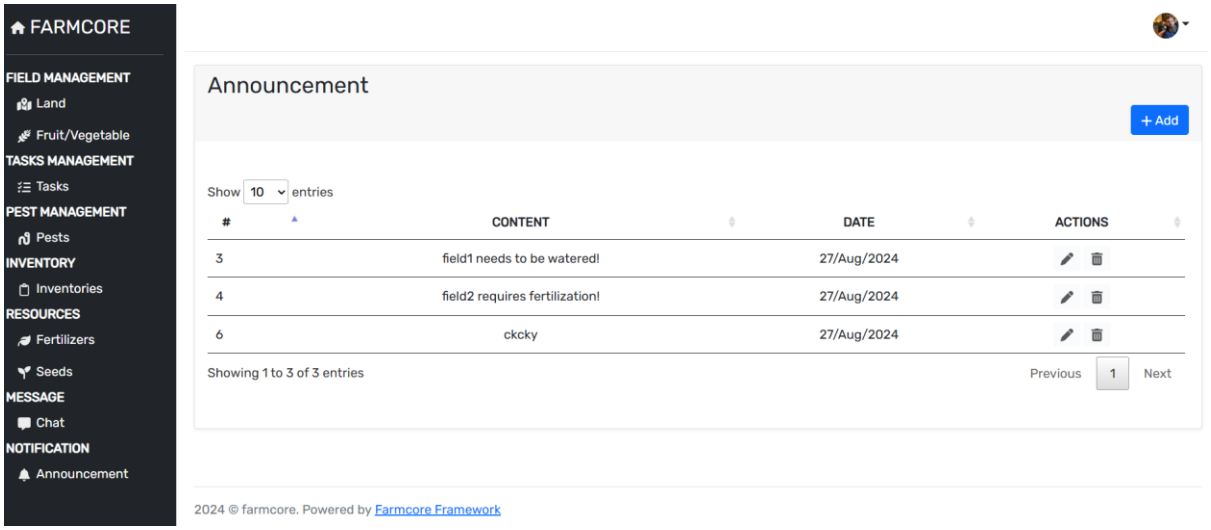


Figure 5.3.51 System Operations of Announcement

Figure 5.3.51 above shows the system operations of the announcement created by the farmers. The farmer can click on the "Add" button to create an announcement. The farmer can

also edit and delete the announcement that was created by himself or herself. The farmer can click on the “Update” or “Delete” button to conduct the action.

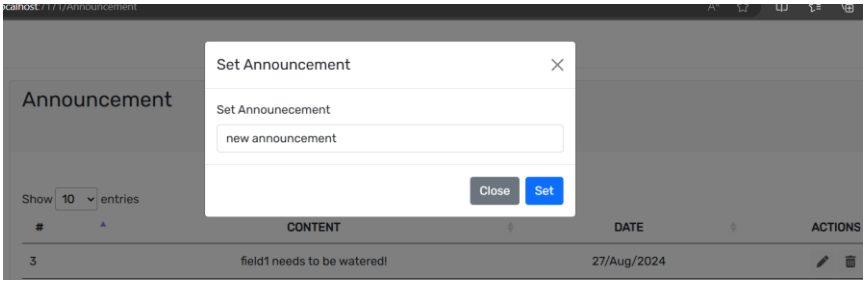


Figure 5.3.52 System Operations of Set Announcement

Figure 5.3.52 above shows the system operations of set announcements for the others. When the farmer clicks on the “Add” button in Figure 5.3.51, the modal to add an announcement is opened. The farmer will be asked to enter the content of the announcement. After clicking on the “Set” button of the modal, the modal will close and the announcement will be posted successfully.

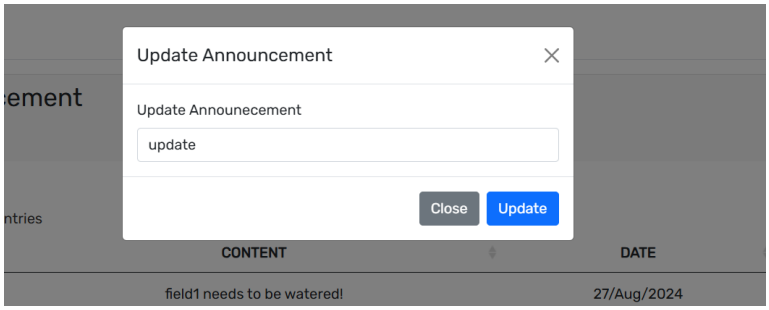


Figure 5.3.53 System Operations of Update Announcement

Figure 5.3.53 shows the system operations of updating the announcement. The farmer can choose to update the particular announcement by clicking on the “Update” button in Figure 5.3.51. After clicking, this modal is opened and will ask to enter the new announcement content. Once clicked on the “Update” button, the announcement is successfully updated and the modal is closed.

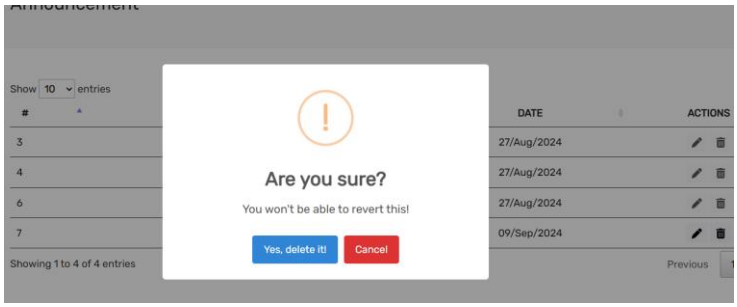


Figure 5.3.54 System Operations of Delete Announcement

Figure 5.3.54 above shows the system operations of the delete announcement. Once the farmer clicks on the "Delete" button of the particular announcement in Figure 5.3.51, it will open a modal and ask for delete confirmation. If the farmer clicks on "Yes, delete it", the modal will close, and the particular announcement will be deleted and no longer able to view.

5.4 Implementation and Issues Challenges

The challenge I have faced during the development of this web application is the corruption of the database server. This issue will occur anytime while developing and operating the web application. This causes me to be unable to log in or manage the data from the database due to the server being down. Thus, it forced me to close up the XAMPP application for several days and restart it. However, this method is not suggested to solve the issue as it caused me to delay the progress. Sometimes, it may not be useful to solve the issue which means required to delete the whole XAMPP application from my device and reinstall it. Since deleting and reinstalling, all the stored data has been lost; therefore, I have to create all the tables for the database again.

Another challenge I have faced during the development process is part of the source code sometimes might suddenly not work as usual. When facing this issue, I have to remove the particular source code and rewrite the source code again. Otherwise, the issue will not be solved and cause the application to unable to be function properly.

5.5 Concluding Remark

In short, this chapter has covered the software setup, setting configuration, system operations as well as the implementation and issue challenges. The software setup has included the software tools I have used to develop the whole web application. The setting configuration is to connect the web application with the MySQL database server to operate CRUD operations. The system operation is a kind of user manual to guide the farmer to integrate this web application into their farming business operations. Last, the implementation and issue challenges have described the difficulties I have met while developing this web application.

Chapter 6

System Evaluation

6.1 Use Case Testing

User Sign Up

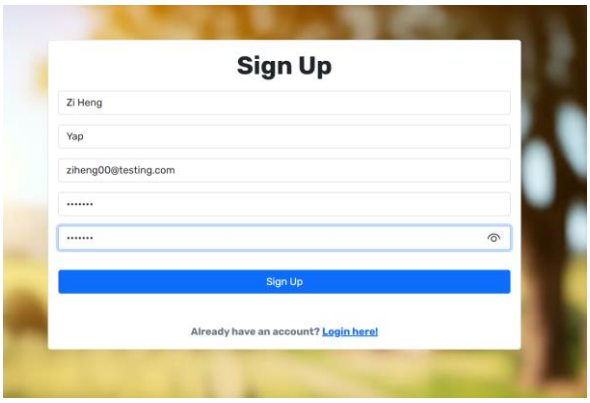


Figure 6.1.1 Input of Sign Up

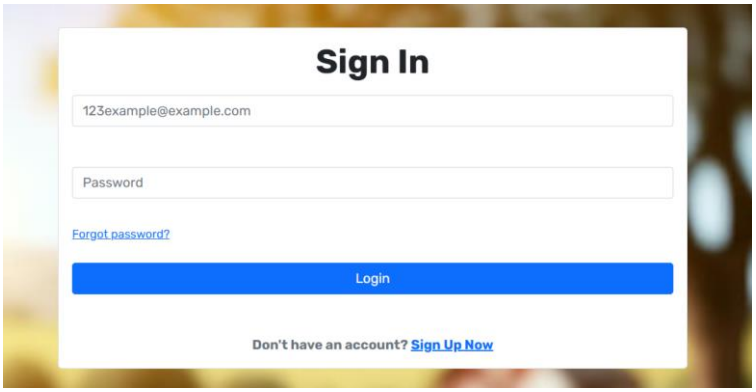


Figure 6.1.2 Output After Sign Up

Figure 6.1.1 above shows the input of registering for an account while Figure 6.1.2 shows the output after signing up for an account. Once the farmer successfully signs up for an account and information is saved in the database correctly, it will auto-redirect back to the sign-in page. From the figure above, it has shown that the account registration is successfully conducted and redirected back to the sign-in page. Thus, the result of user sign-up case testing is PASS.

Change Password

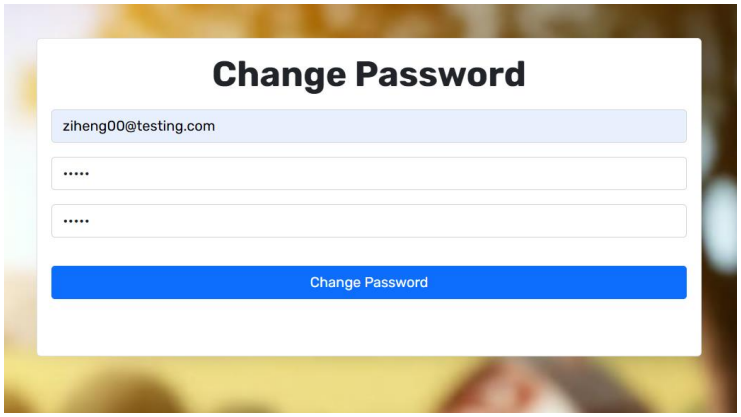


Figure 6.1.3 Input of Change Password

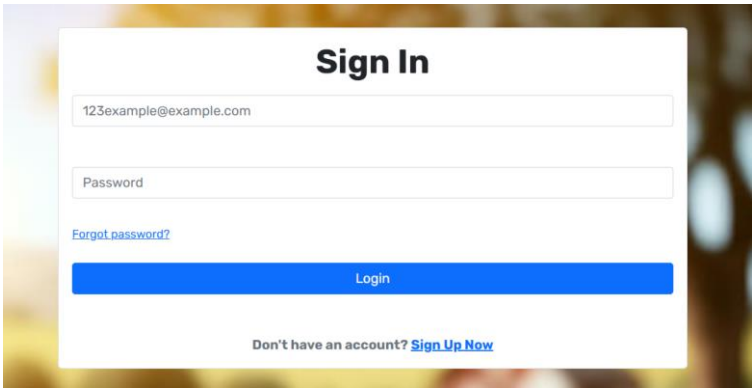


Figure 6.1.4 Output of Successfully Changed Password

Figure 6.1.3 shows the input of changing the password while Figure 6.1.4 shows the output after changing the password successfully. After the password has successfully changed, it will automatically redirect back to the sign-in page. Figure 6.1.3 and figure 6.1.4 above have shown that the password has successfully changed and successfully redirected back to the sign-in page. Hence, the result of the change password case testing is PASS.

Sign In

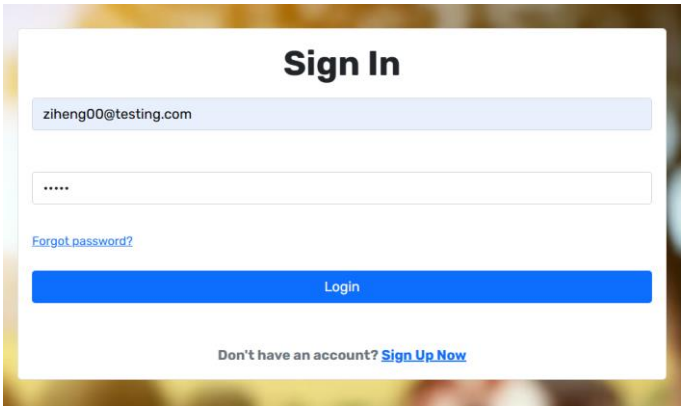


Figure 6.1.5 Input of Sign-In

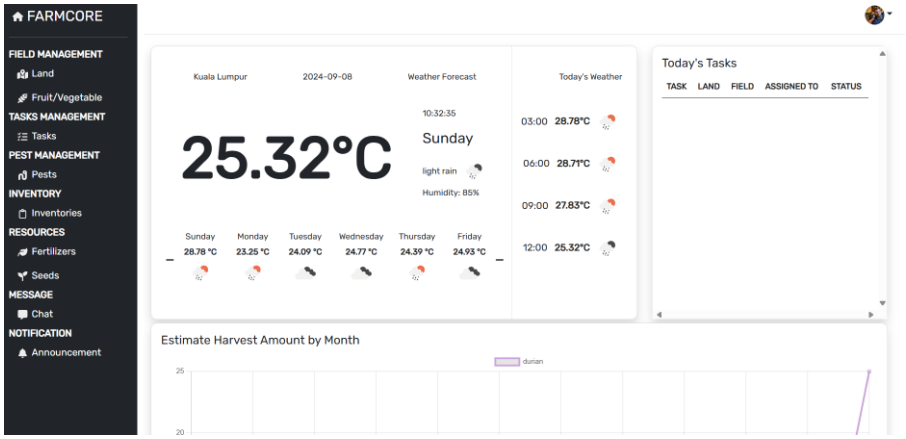


Figure 6.1.6 Output of Successfully Sign-In

Figure 6.1.5 above shows the input of signing in to the web application and figure 6.1.6 above shows the output of successfully signing in to the web application. After entering the valid email and password, it will auto-redirect to the homepage of the web application. The two figures above show that after successfully signing in to the web application, it has successfully redirected to the homepage of the web application. Thus, the result of sign-in case testing is PASS.

Land

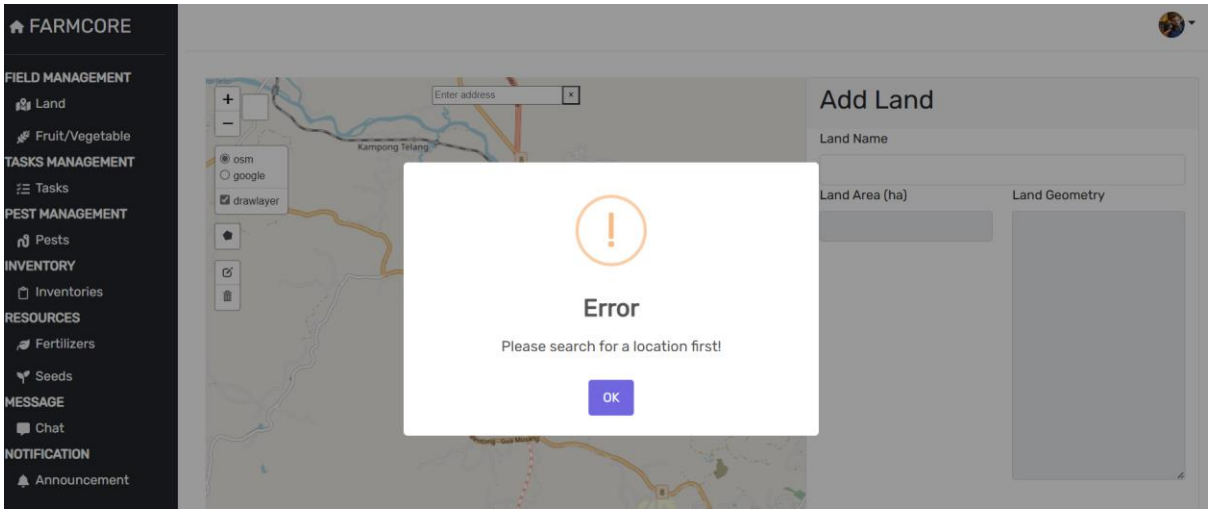


Figure 6.1.7 Output Without Input Searching Address

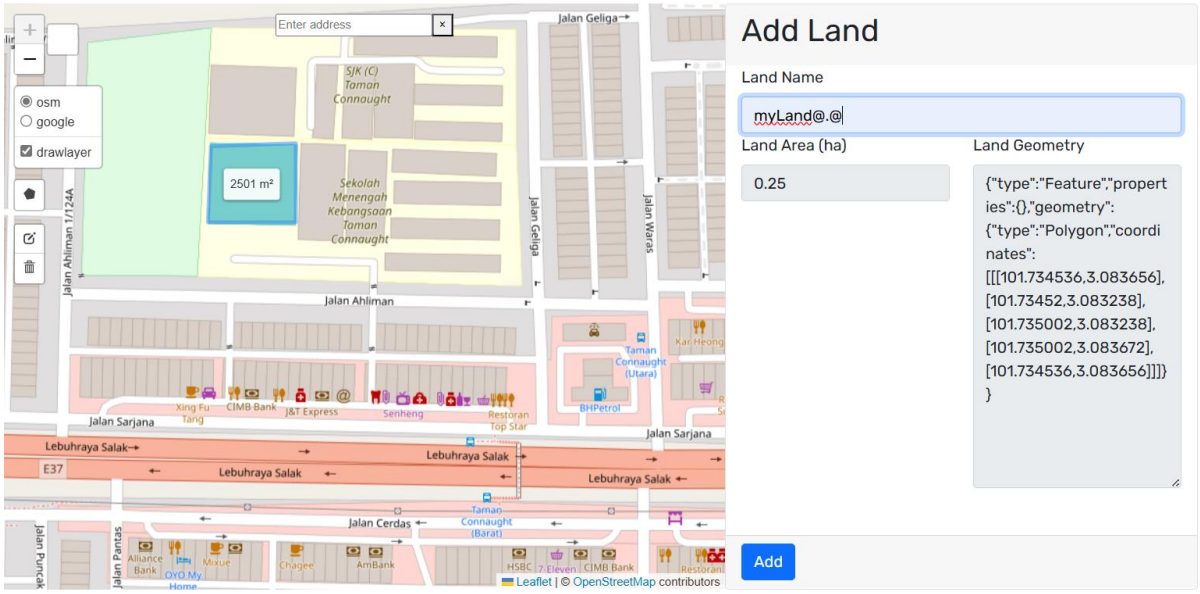


Figure 6.1.8 Output of Inputted Address and Input of Add Land

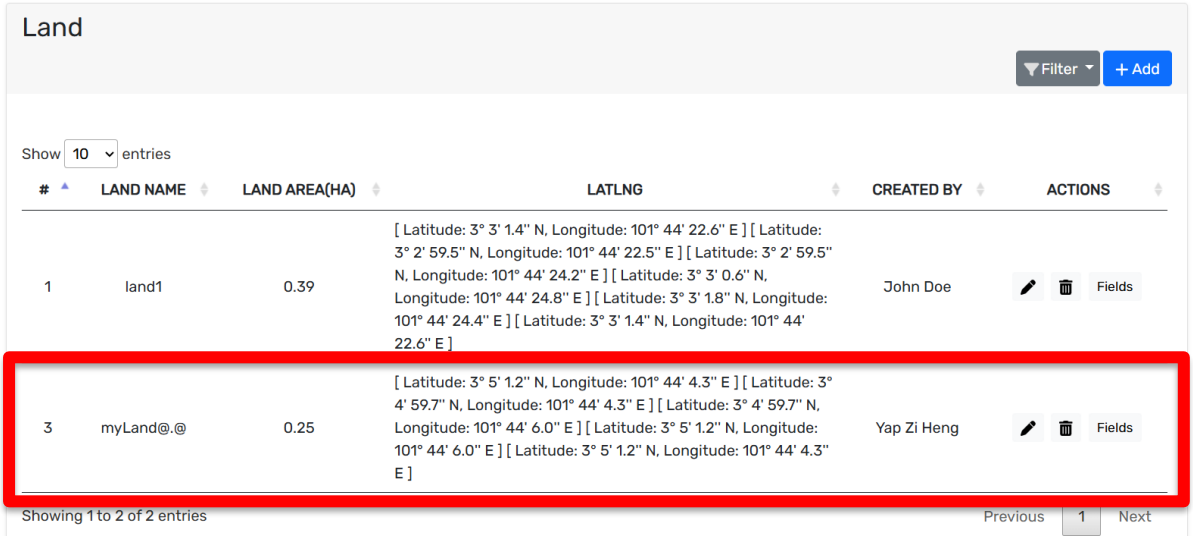


Figure 6.1.9 Output of Successfully Added Land

Figure 6.1.7 shows the output of plotting land without searching for an address. If the farmer does not search for an address, it will show an error message. If the farmer has entered and searched for an address; then the farmer can continue to plot the land as Figure 6.1.8 above. After successfully adding new land, it then will redirect back to the main page of the land module and the new land information has been added to the list now as Figure 6.1.9. Thus, the result of adding new land case testing is PASS.

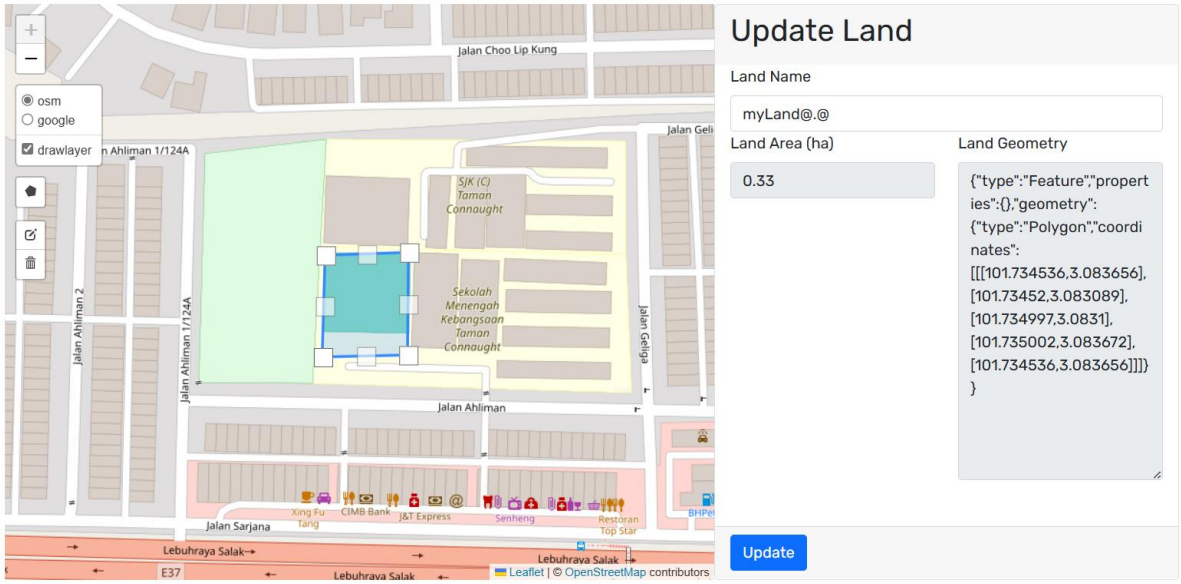


Figure 6.1.10 Input of Update Land

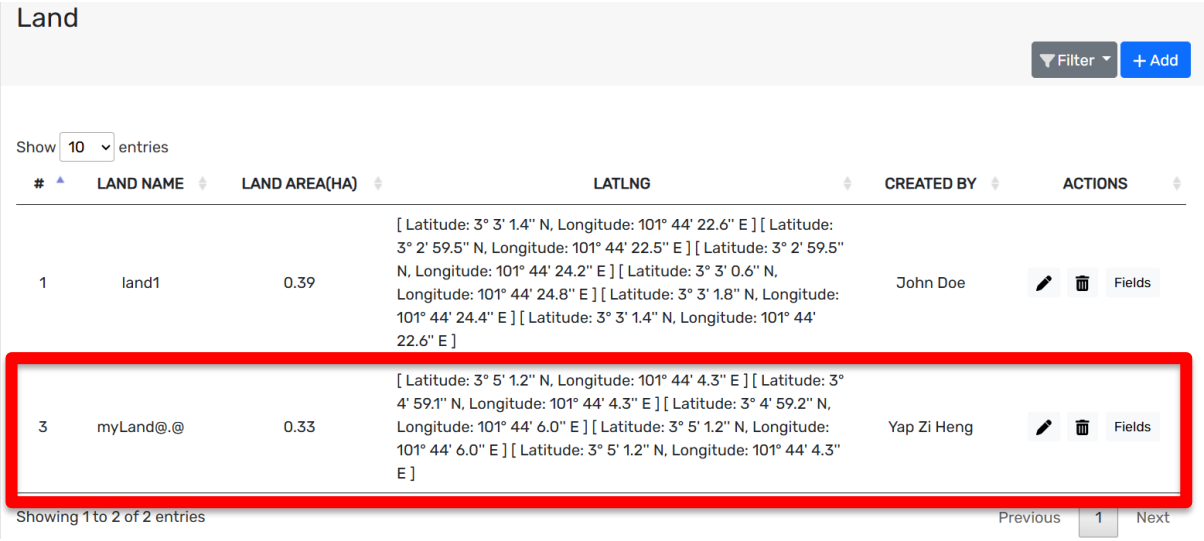


Figure 6.1.11 Output of Update Land

Figure 6.1.10 above shows the input of updating the land. After the land is updated, it will redirect back to the main page of the land module and it shows that the information has been updated as shown in Figure 6.1.11 above. Hence, the result of updated land case testing is PASS.

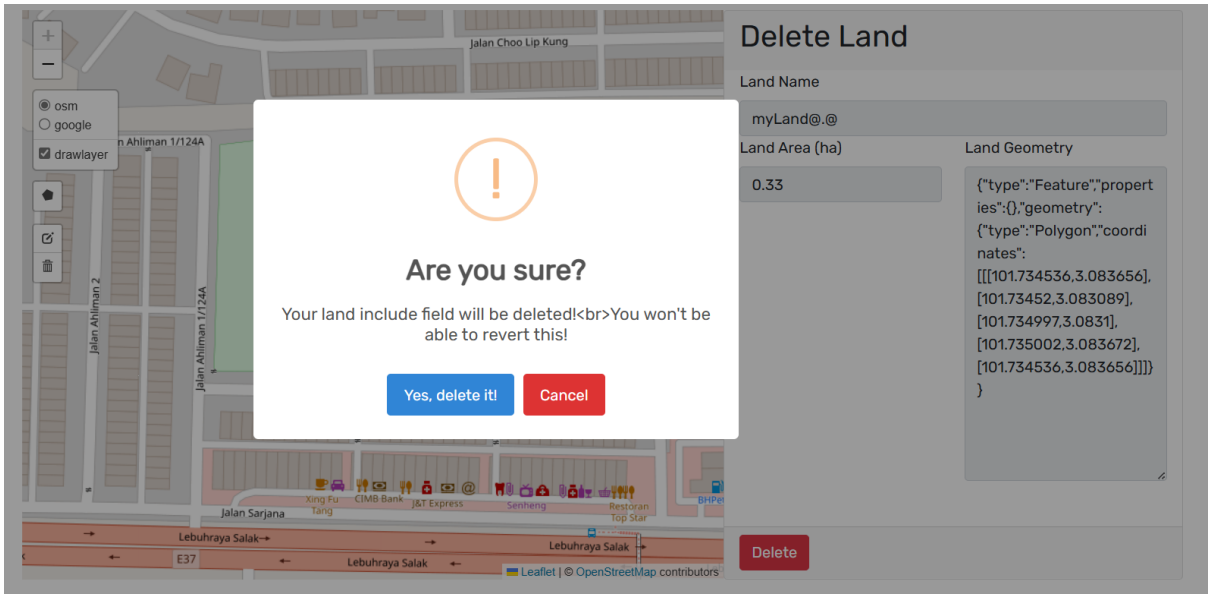


Figure 6.1.12 Input of Delete Land

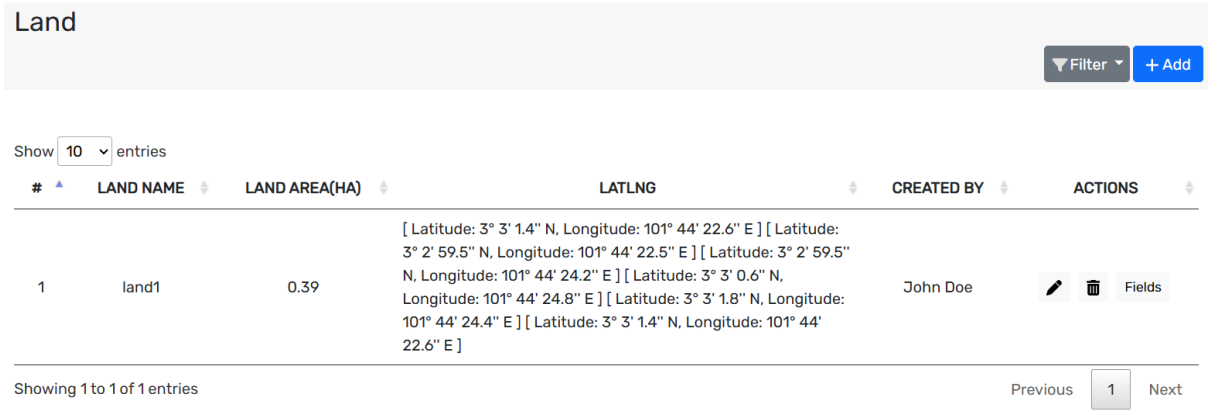


Figure 6.1.13 Output of Delete Land

Figure 6.1.12 above shows the input of deleted land. When the delete button is clicked, it will show an alert message to remind the farmer the land might include a field. This alert message will always be shown every time the farmer wants to delete land regardless of the land whether including the field. After being confirmed to delete, it will redirect back to the main page of the land module and the deleted land has been removed from the table as shown in Figure 6.1.13. Hence, the result of delete land case testing is PASS.

Field

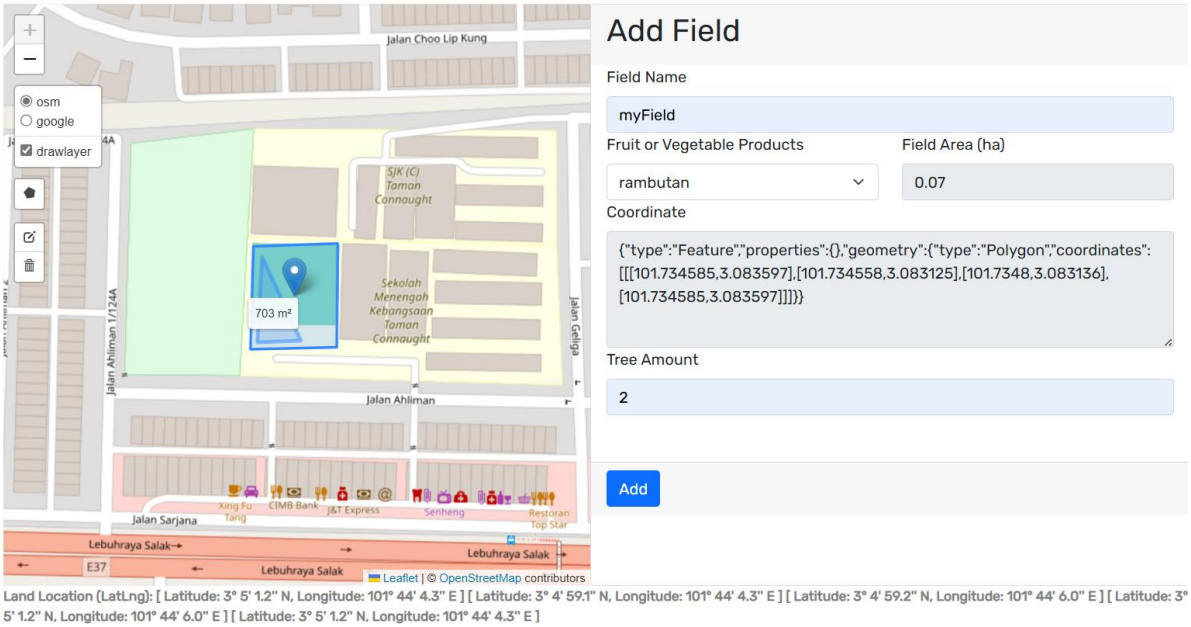


Figure 6.1.14 Input of Add New Field to Land

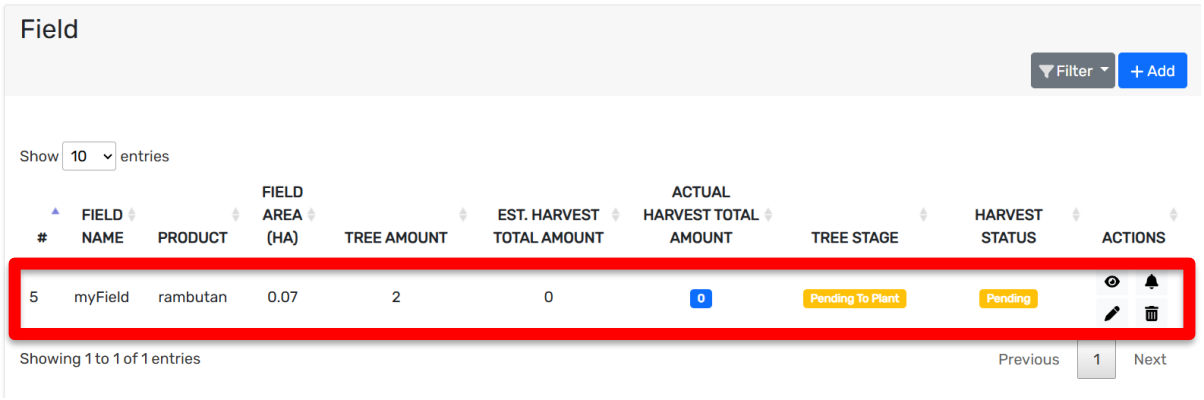


Figure 6.1.15 Output of Successfully Added New Field

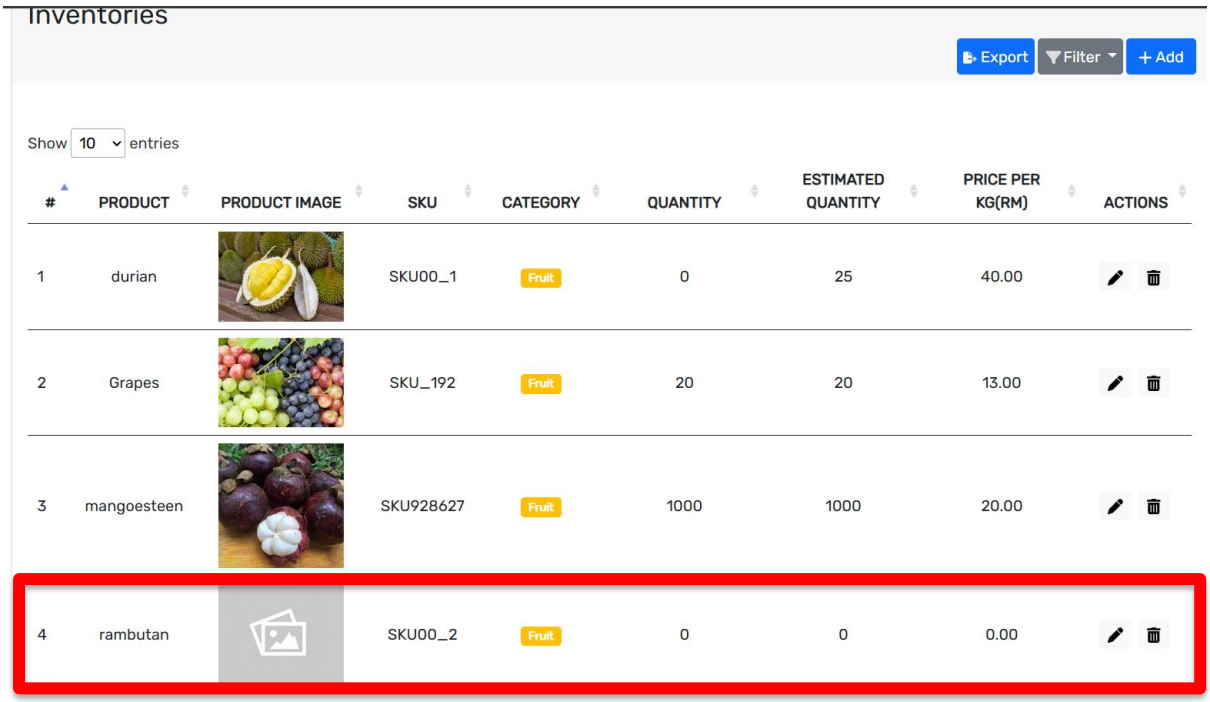


Figure 6.1.16 Output of Automatically Add Product in Inventory

Figure 6.1.14 shows the input of adding a new field in the land. After plotting a field within the land and entering the necessary information, it will successfully redirect back to the main page of the field module and the new field will be shown now in figure 6.1.15. Since initially the "rambutan" fruit is not been stored in inventory, the web application will automatically add it to the inventory as Figure 6.1.16. Otherwise, if the web application found is already stored, it will just remain the same. Hence, the result of adding a new field case is PASS.

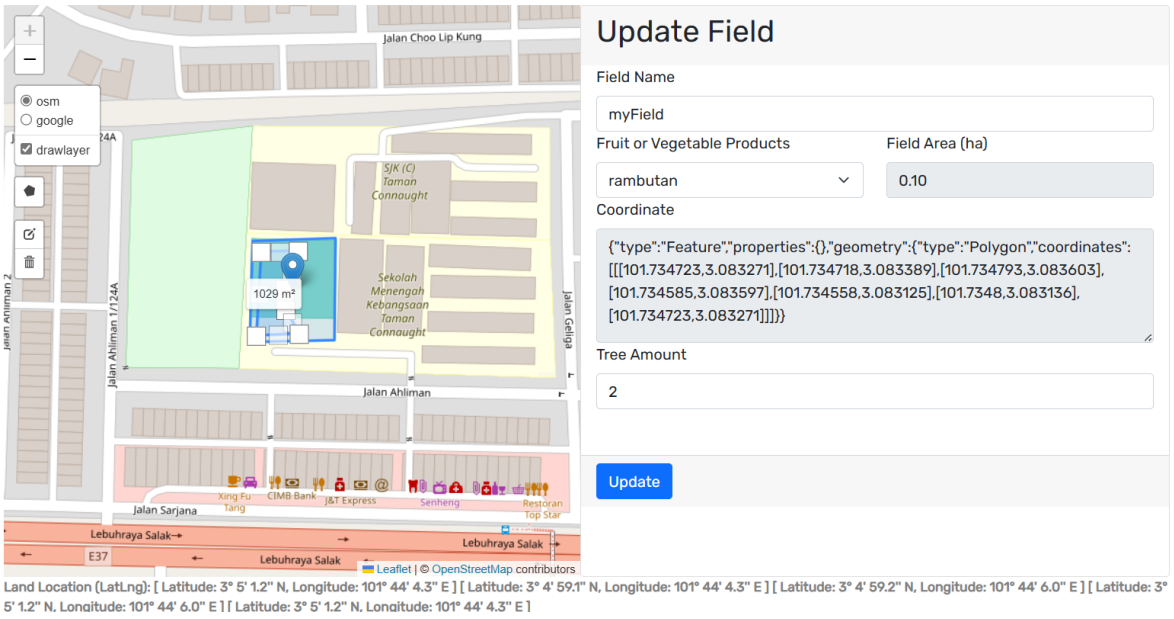


Figure 6.1.17 Input of Update Field

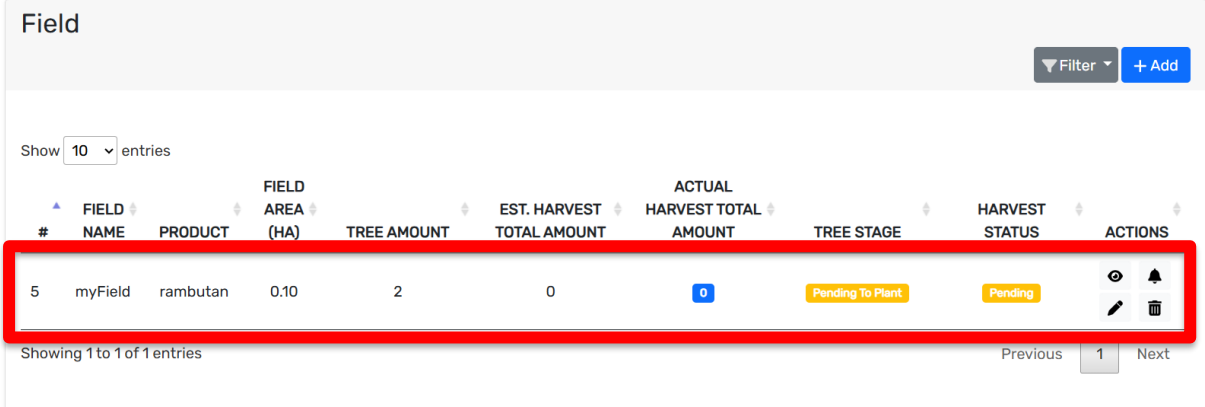


Figure 6.1.18 Output of Successful Update Field

Figure 6.1.17 shows the input of the update field while Figure 6.1.18 shows the output after successful update the field. After the farmer updates the field information as in Figure 6.1.18 and update it, it will redirect back to the main page of the field module, and the information is updated. While in figure 6.1.17 shows the product is not updated; thus, the inventory will remain unchanged as in Figure 6.1.16. Therefore, the result of the update field is PASS.

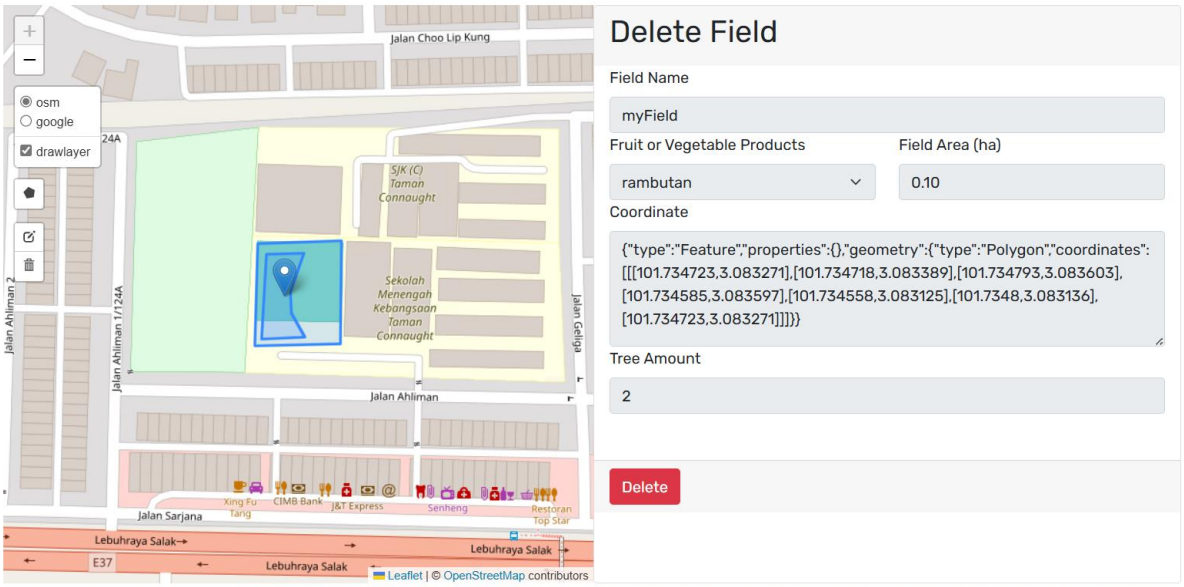


Figure 6.1.19 Input of Delete Field

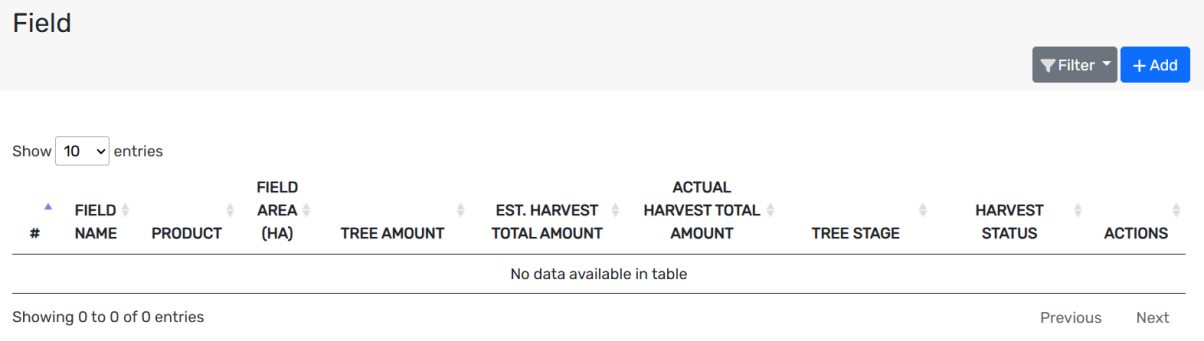


Figure 6.1.20 Output of Successfully Delete Field

Figure 6.1.19 above shows the input of the delete field while Figure 6.1.20 shows the output of the delete field. After the farmer reviews and confirms to delete it, it will be redirected back to the main page of the field module, and the field will be deleted as shown in Figure 6.1.20. Thus, the result of the delete field case is PASS.

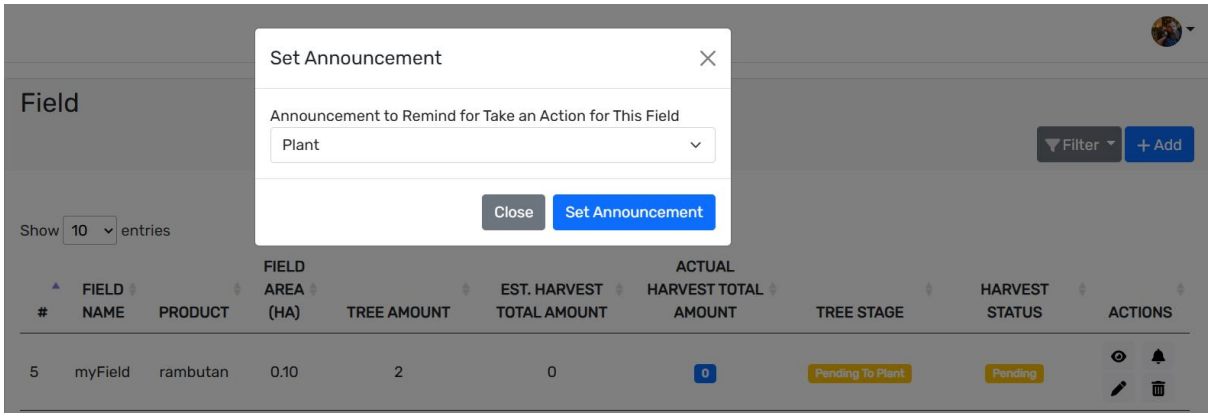


Figure 6.1.21 Input to Make Announcement for Field

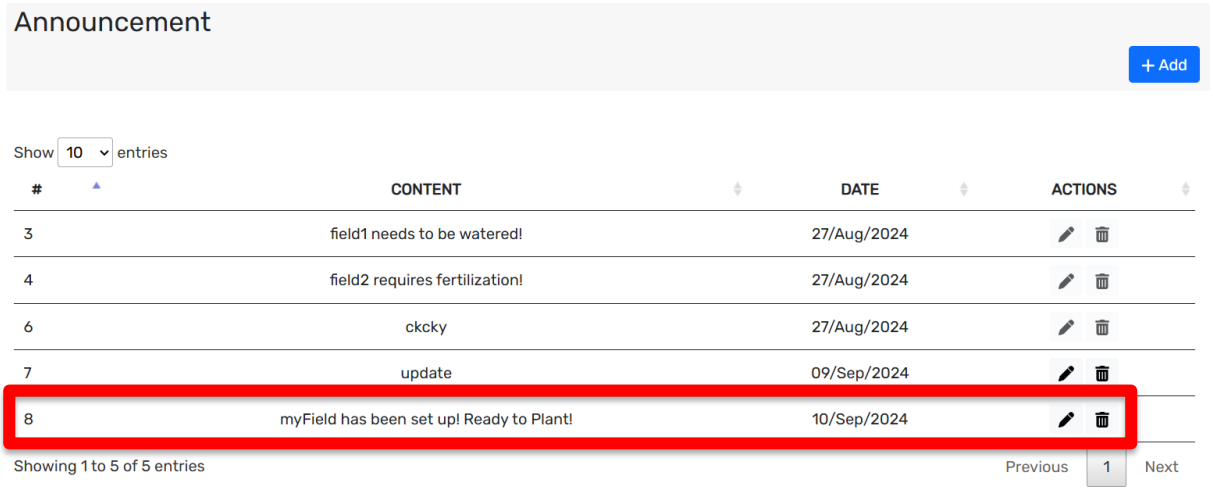


Figure 6.1.22 Output of After Make Announcement for Field

Figure 6.1.21 shows the input of announcing the particular field. After the farmer selects the action that needs to be taken and confirms to make the announcement, the announcement will be posted on the main page of the announcement module as shown in Figure 6.1.22 above. In short, the result of the announcement of field case testing is PASS.

Agricultural Product (Fruit/Vegetable) Planting Data

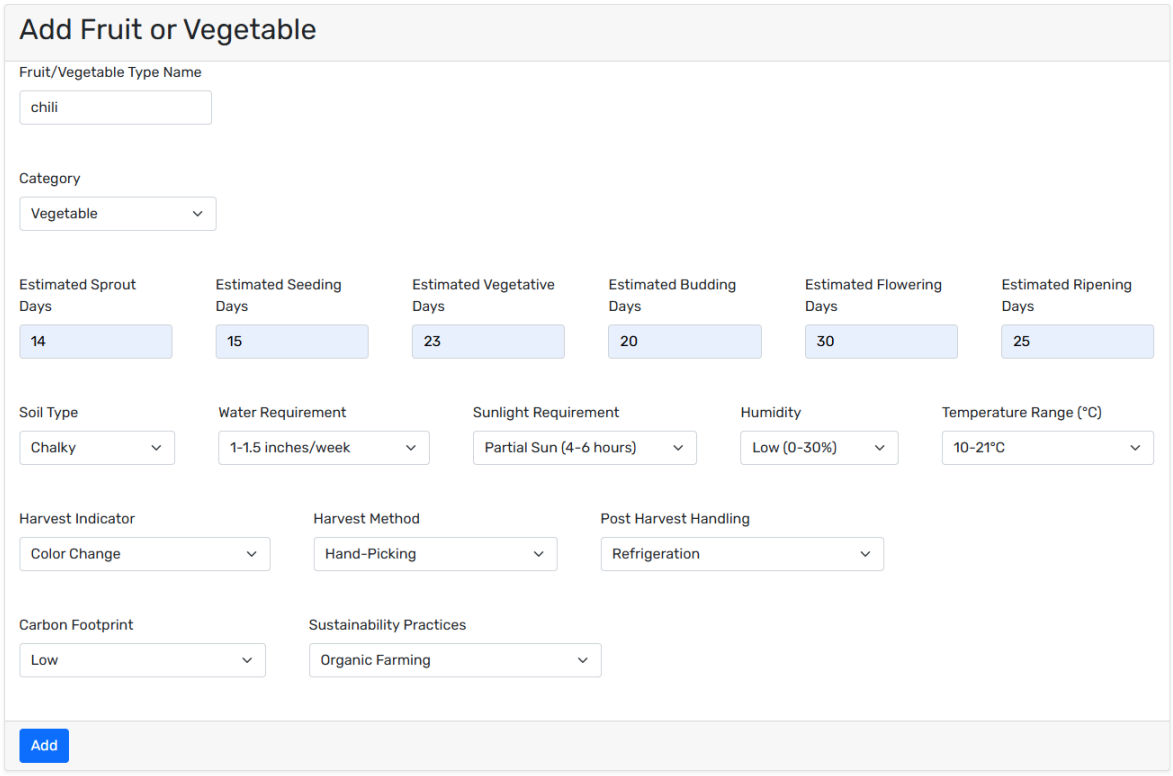


Figure 6.1.23 Input of Add New Agricultural Product Planting Data

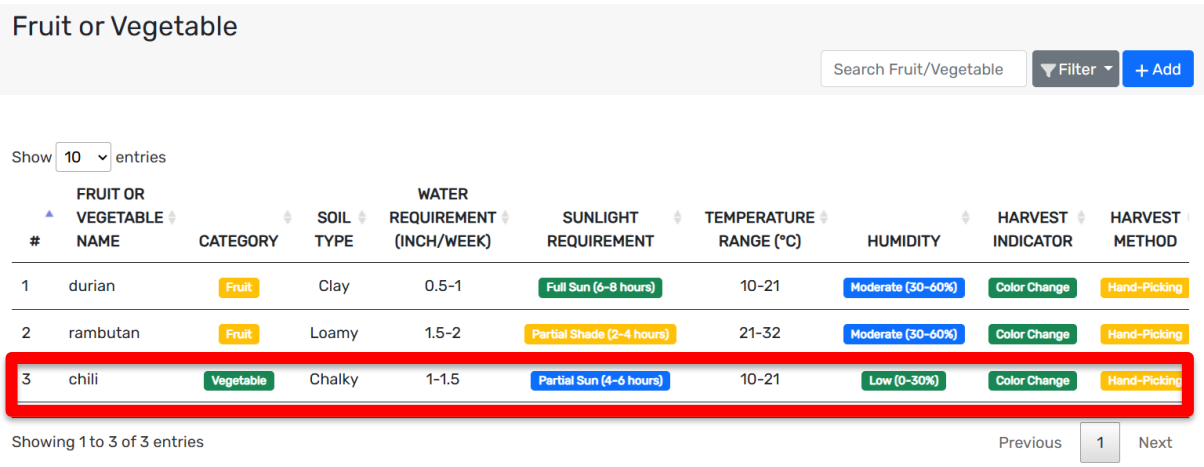


Figure 6.1.24 Output of Successfully Added New Agricultural Product Planting Data

Figure 6.1.23 above shows the input of adding new fruit or vegetable planting data. After entering all necessary information and successfully adding, it will redirect back to the main page of this fruit or vegetable planting data module and the new data has been added to the list as shown in Figure 6.1.24.

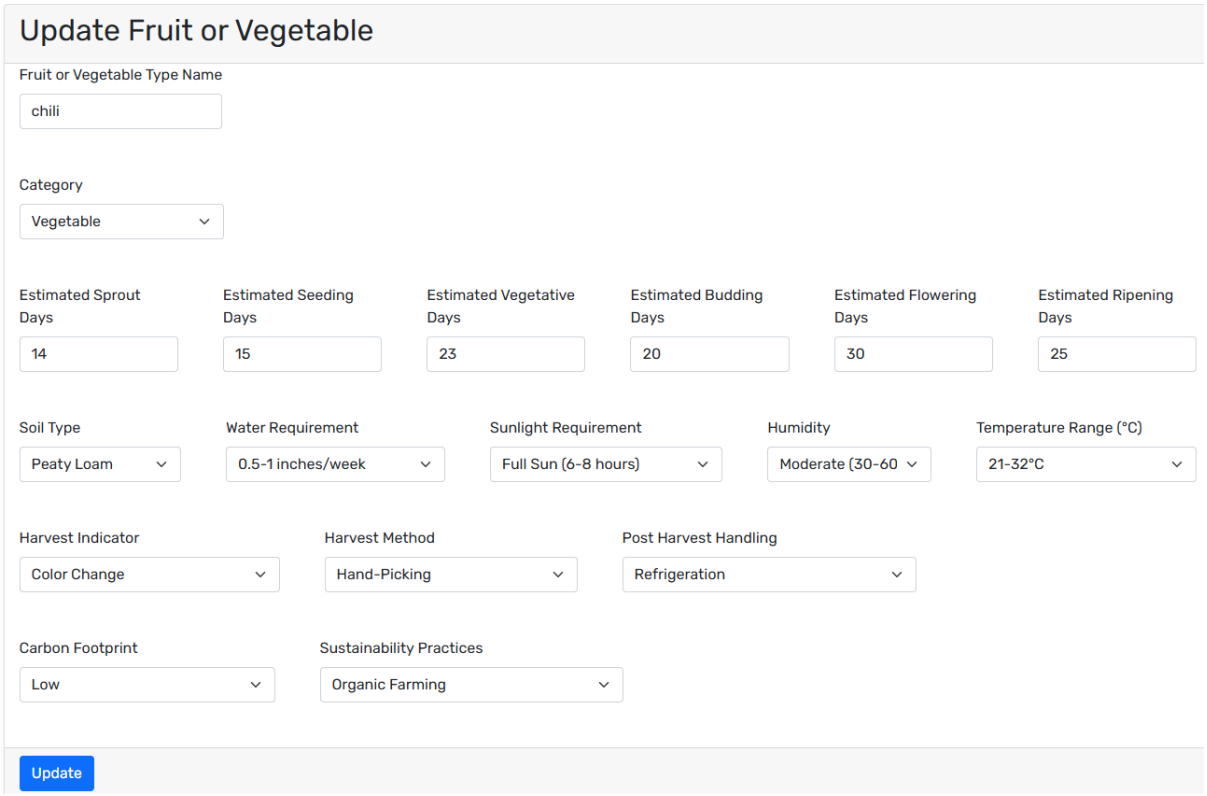


Figure 6.1.25 Input of Update Existing Agricultural Product Planting Data

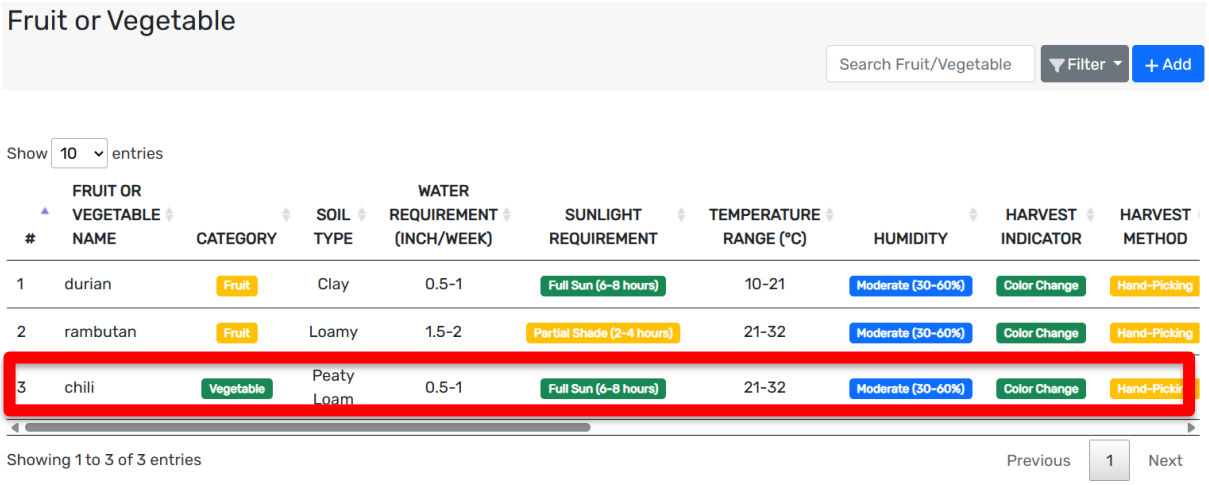


Figure 6.1.26 Output of Successfully Updated Existing Agricultural Product Planting Data

Figure 6.1.25 above shows the input of updated fruit or vegetable planting data. After the information is updated successfully, it will redirect back to the main page of this fruit or vegetable planting data module and the information on the particular product is updated shows as Figure 6.1.26. To conclude, the result of updating the existing agricultural product planting data case is PASS.

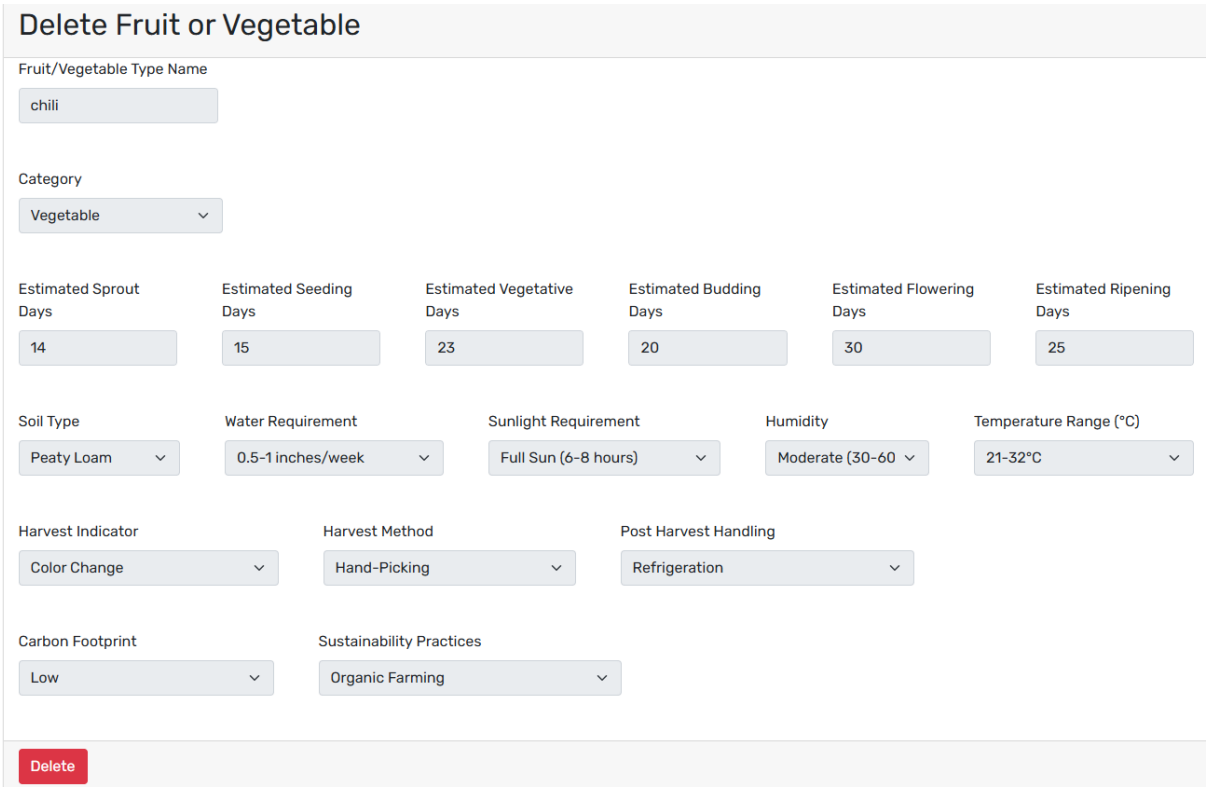


Figure 6.1.27 Input of Delete Existing Agricultural Product Planting Data

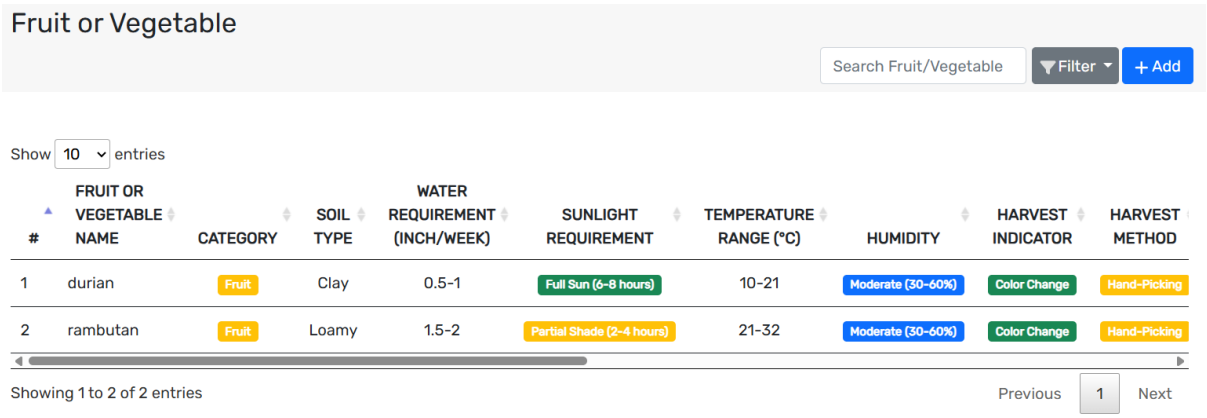


Figure 6.1.28 Output of Successfully Deleted Existing Agricultural Product Planting Data

Figure 6.1.27 above shows the input of deleted existing agricultural product planting data. After being reviewed and successfully deleted, it will redirect back to the main page of this fruit or vegetable planting data module and the information on the particular product is deleted shown as in Figure 6.1.28. To conclude, the result of deleting existing agricultural product planting data case is PASS.

Tasks

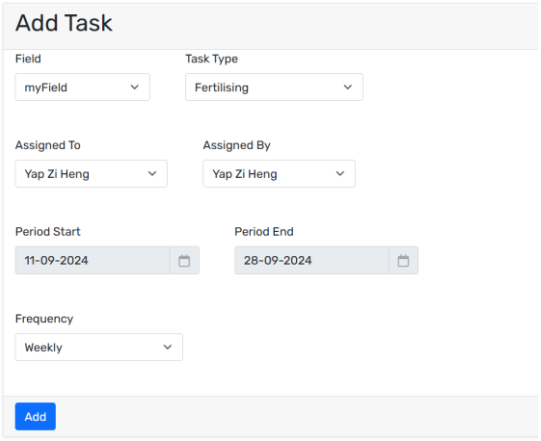


Figure 6.1.29 Input of Add Task

Figure 6.1.30 Output of Successfully Add New Task

Figure 6.1.29 shows the input of adding a new task. This figure, it has shown that the frequency assigned for this task is “Weekly” and the period for this task is approximately 3 weeks; hence, the total number of tasks that will be assigned is 3. While from the figure 6.1.30 above shows the 3 tasks that have been successfully added with the different calculated due dates within the period of the task. Besides, the notification has also been sent to the assignee as shown in Figure 6.1.30 above. In short, the result of adding new task case testing is PASS.

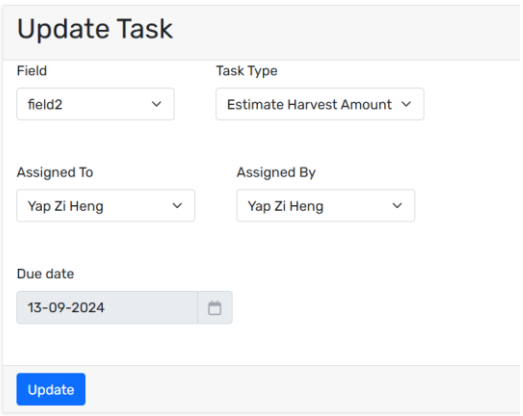


Figure 6.1.31 Input of Update Task

17	Plant	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Daily	Pending		
18	Fertilising	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Weekly	Pending		
19	Estimate Harvest Amount	land1	field2	Yap Zi Heng	Yap Zi Heng	13/Sep/2024	Weekly	Pending		
20	Fertilising	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	25/Sep/2024	Weekly	Pending		

Figure 6.1.32 Output of Successfully Update Task

Figure 6.1.31 above shows the input of updating the pending task. After editing the particular task and successfully updating, it will redirect to the main page of the task and the particular task has been updated shown as in Figure 6.1.3.32. Thus, the result of update task case testing is PASS.

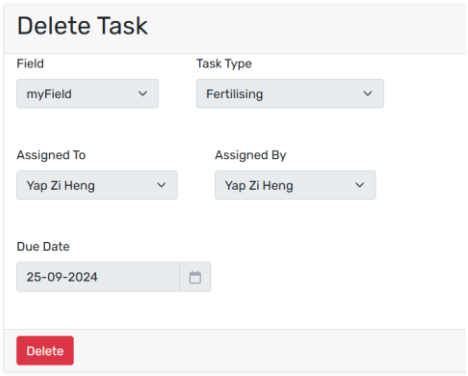
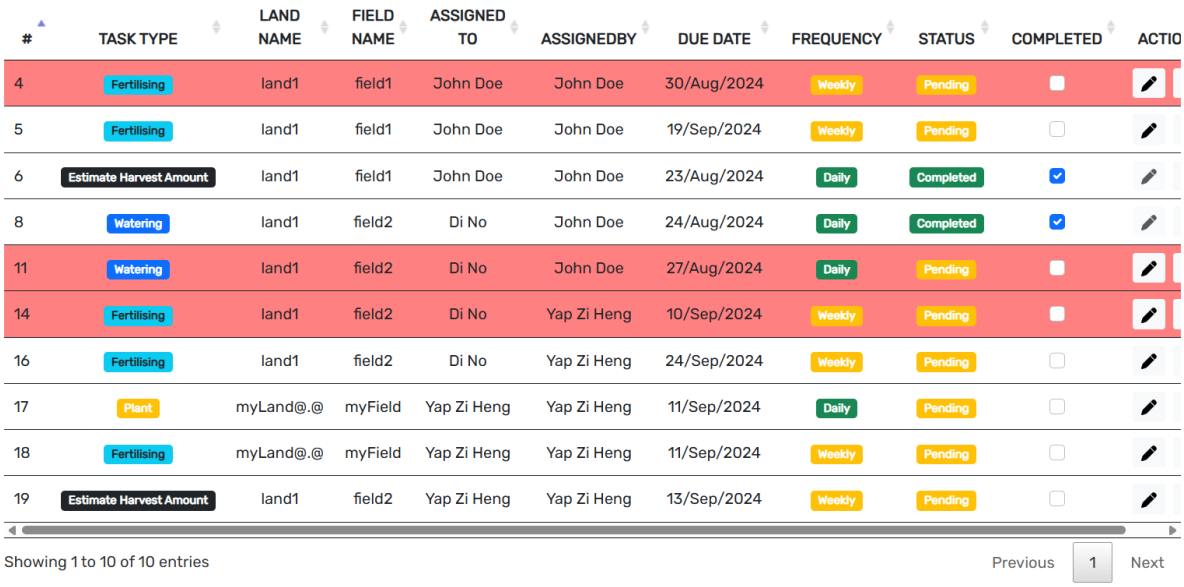


Figure 6.1.33 Input of Delete Task



#	TASK TYPE	LAND NAME	FIELD NAME	ASSIGNED TO	ASSIGNEDBY	DUE DATE	FREQUENCY	STATUS	COMPLETED	ACTIO
4	Fertilising	land1	field1	John Doe	John Doe	30/Aug/2024	Weekly	Pending		
5	Fertilising	land1	field1	John Doe	John Doe	19/Sep/2024	Weekly	Pending		
6	Estimate Harvest Amount	land1	field1	John Doe	John Doe	23/Aug/2024	Daily	Completed		
8	Watering	land1	field2	Di No	John Doe	24/Aug/2024	Daily	Completed		
11	Watering	land1	field2	Di No	John Doe	27/Aug/2024	Daily	Pending		
14	Fertilising	land1	field2	Di No	Yap Zi Heng	10/Sep/2024	Weekly	Pending		
16	Fertilising	land1	field2	Di No	Yap Zi Heng	24/Sep/2024	Weekly	Pending		
17	Plant	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Daily	Pending		
18	Fertilising	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Weekly	Pending		
19	Estimate Harvest Amount	land1	field2	Yap Zi Heng	Yap Zi Heng	13/Sep/2024	Weekly	Pending		

Showing 1 to 10 of 10 entries Previous 1 Next

Figure 6.1.34 Output of Successfully Delete Task

Figure 6.1.33 shows the input of deleting a task. Once completed review the task and confirm delete is a success, it will redirect back to the main page of the task module and the

deleted task has been removed as shown in Figure 6.1.34 above. Therefore, the result of delete task case testing is PASS.

#	TASK TYPE	LAND NAME	FIELD NAME	ASSIGNED TO	ASSIGNEDBY	DUE DATE	FREQUENCY	STATUS	COMPLETED	ACTIO
4	Fertilising	land1	field1	John Doe	John Doe	30/Aug/2024	Weekly	Pending	☐	✎
5	Fertilising	land1	field1	John Doe	John Doe	19/Sep/2024	Weekly	Pending	☐	✎
6	Estimate Harvest Amount	land1	field1	John Doe	John Doe	23/Aug/2024	Daily	Completed	☑	✎
8	Watering	land1	field2	Di No	John Doe	24/Aug/2024	Daily	Completed	☑	✎
11	Watering	land1	field2	Di No	John Doe	27/Aug/2024	Daily	Pending	☐	✎
14	Fertilising	land1	field2	Di No	Yap Zi Heng	10/Sep/2024	Weekly	Pending	☐	✎
16	Fertilising	land1	field2	Di No	Yap Zi Heng	24/Sep/2024	Weekly	Pending	☐	✎
17	Plant	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Daily	Pending	☐	✎
18	Fertilising	myLand@.@	myField	Yap Zi Heng	Yap Zi Heng	11/Sep/2024	Weekly	Pending	☐	✎

Figure 6.1.35 Input of Update Task Completion Status

Figure 6.1.36 Output of Update Task Completion Status

Figure 6.1.35 above shows the input of the updated task completion status. Once the checkbox in the status column is checked, the task status will change from “Pending” to “Completed” and the notification will be removed. The result is shown in the figure 6.1.36. Hence, the result of the update task completion status is PASS.

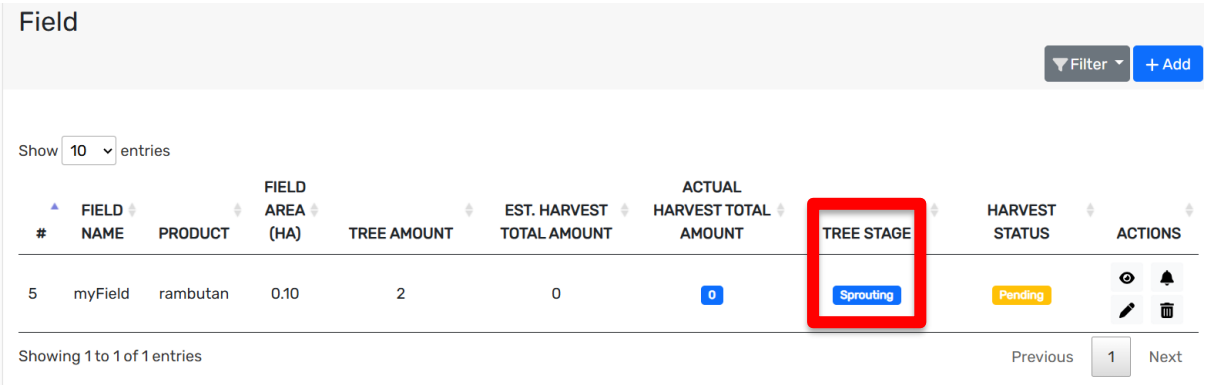


Figure 6.1.37 Output of After Completed Planting Task

Figure 6.1.37 shows the output of after completed planting task type. Once the planting task has been completed, this web application will start to calculate the growth stage of the

particular agricultural product. This figure shows that the result of the update task completion status case for the planting task type is PASS.

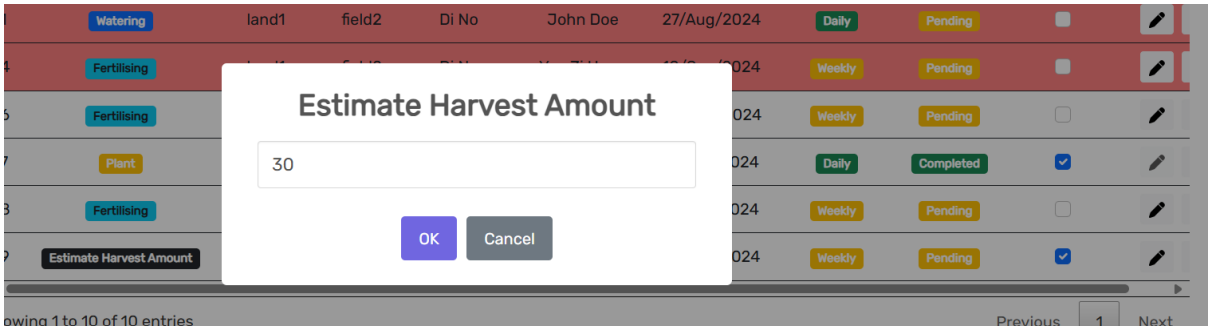


Figure 6.1.38 Input of Update Task Status for Estimate Harvest Amount Task

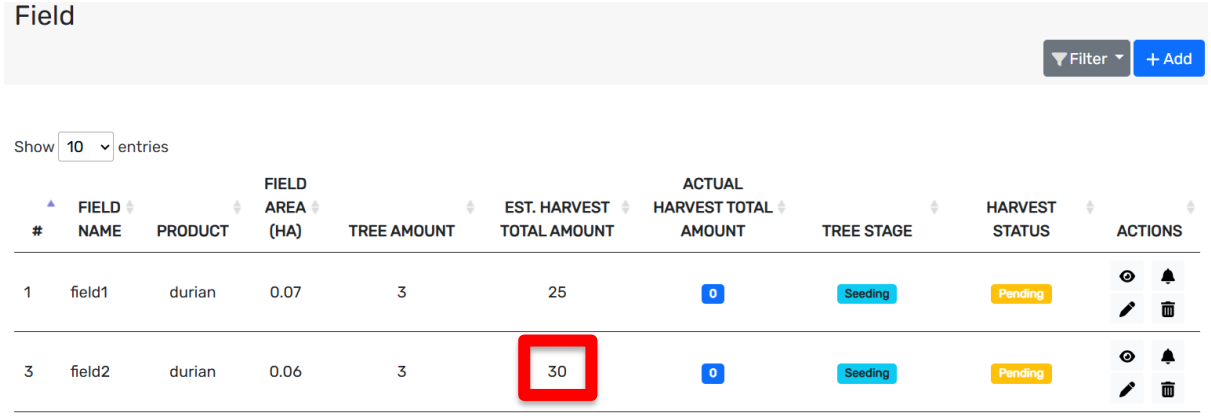


Figure 6.1.39 Output of Successfully Updated Task Status for Estimate Harvest Amount Task in Field

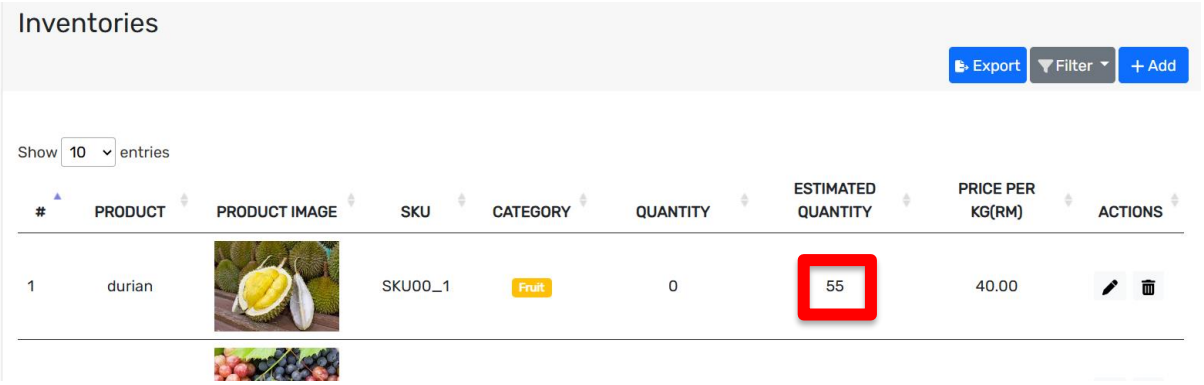


Figure 6.1.40 Output of Successfully Updated Task Status for Estimate Harvest Amount Task in Inventory

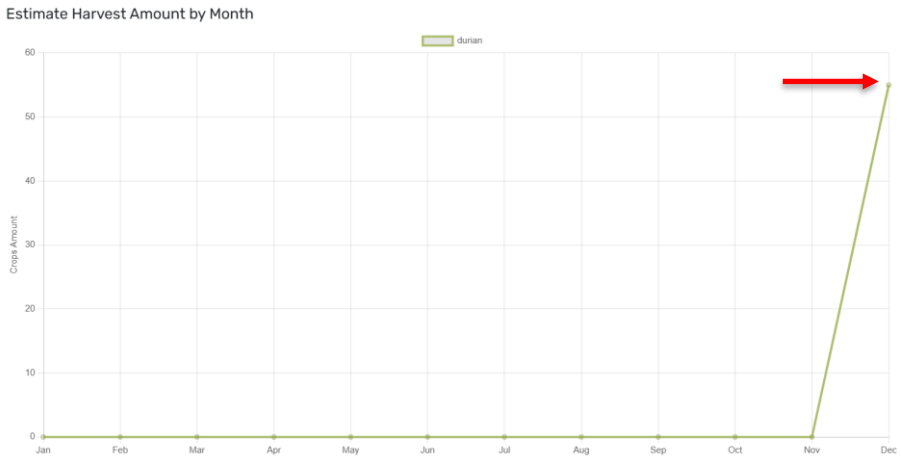


Figure 6.1.41 Output of Updated Line Chart After Completed Estimate Harvest Amount Task

Figure 6.1.39, figure 6.1.40, and Figure 6.1.41 above show the output of the completed estimate of the harvest amount for each cultivating agricultural product. After the farmer enters the estimated amount as shown in Figure 6.1.38, it will automatically update these 3 figures. Figure 6.1.39, shows the estimated harvest amount column has been updated. While for the figure 6.1.40 shows the estimated quantity of the particular products has been recalculated correctly lastly figure 6.1.41 shows that the particular products are estimated to be harvested in December and the quantity will be 55. The reason for the estimated harvested quantity is 55 in December as there are the same agricultural products in another field also estimated to be harvested in December. Hence, the result of the updated task completion status case for estimating task type is PASS.

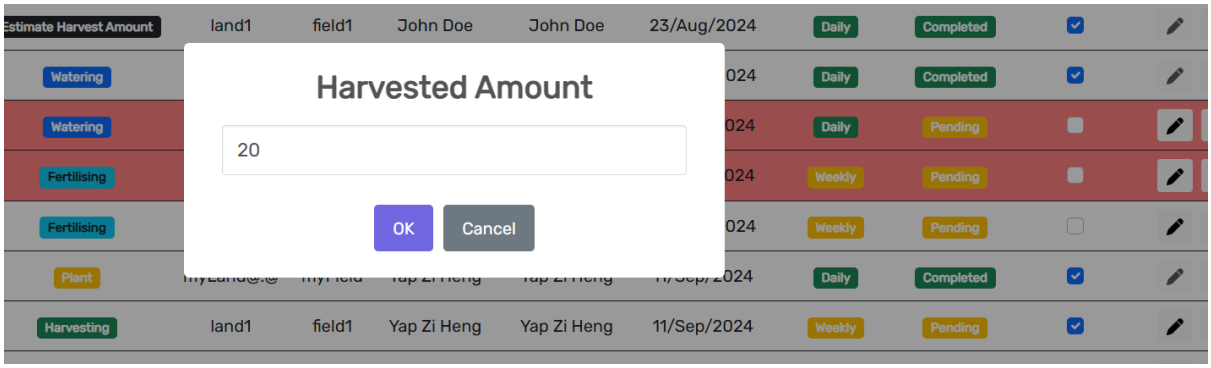


Figure 6.1.42 Input of Update Task Status for Harvesting Task Type

#	FIELD NAME	PRODUCT	FIELD AREA (HA)	TREE AMOUNT	EST. HARVEST TOTAL AMOUNT	ACTUAL HARVEST TOTAL AMOUNT	TREE STAGE	HARVEST STATUS	ACTIONS
1	field1	durian	0.07	3	25	20	Riped	Harvested	

Figure 6.1.43 Output of Update Task Status for Harvesting Task Type in Field

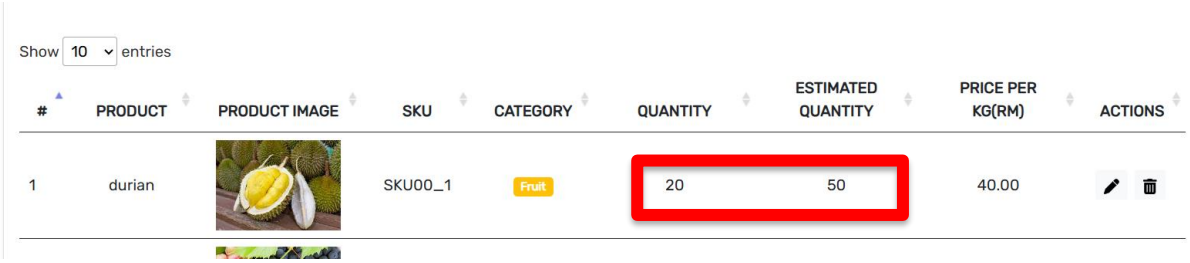


Figure 6.1.44 Output of Update Task Status for Harvesting Task Type in Inventory

Figure 6.1.42 shows the input of the update completion task status for the harvesting task type. After successfully entering the harvest amount, the task status is changed to "Completed". At the same time, the actual harvest amount has been updated in the particular field, and the harvest status changed from "Pending" to "Harvested" as shown in Figure 6.1.43. In Figure 6.1.43, the actual quantity stored in inventory has been added and the estimated quantity has been recalculated. The reason the estimated quantity is shown as 50 not 30 is due to it included the actual quantity. This is because it can allow the farmer to estimate the total quantity that will be stored in the inventory. Hence, the result of the updated task completion status case for the harvesting task type is PASS.

Pest

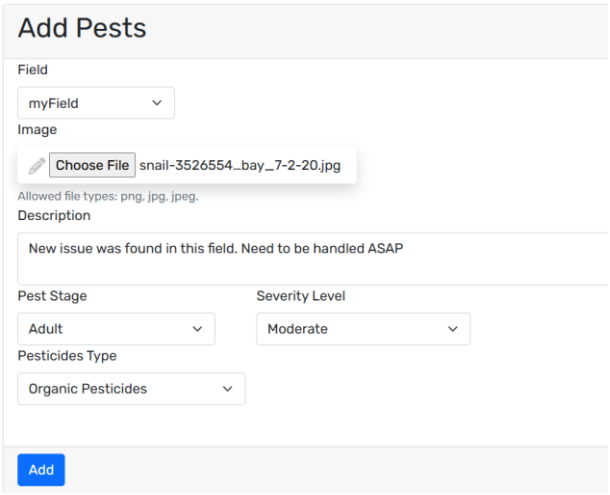


Figure 6.1.45 Input of Add New Pest Issue

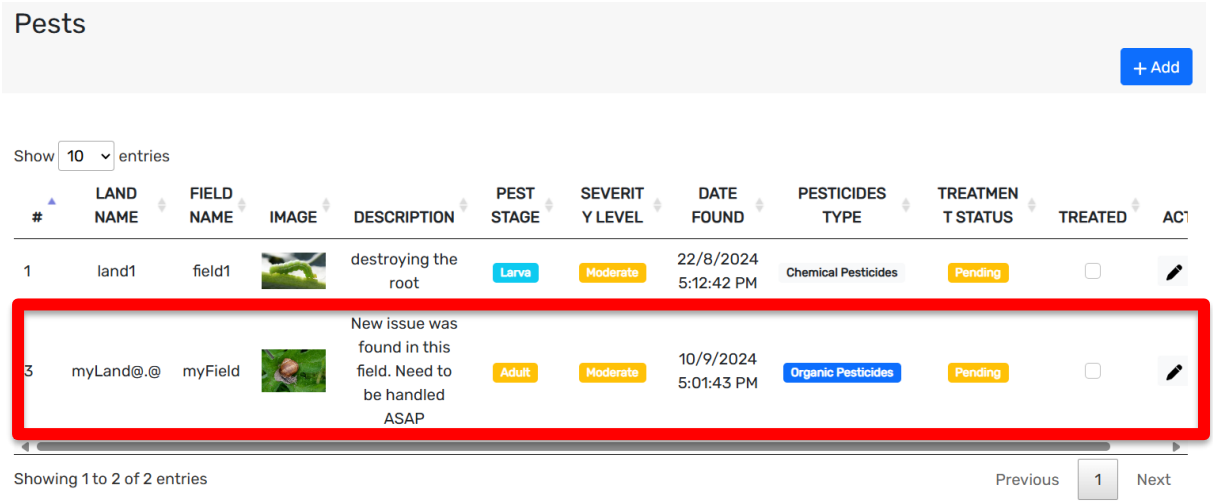


Figure 6.1.46 Output of Successful Add New Pests Issue

Figure 6.1.45 shows the input of new pests found in the particular field. After the farmer enters the necessary input and successfully adds, it will redirect to the main page of the pest module as shown in Figure 6.1.46. From this figure, the new issue has been added and listed in the table. Hence, the result of adding new pest issue case testing is PASS.

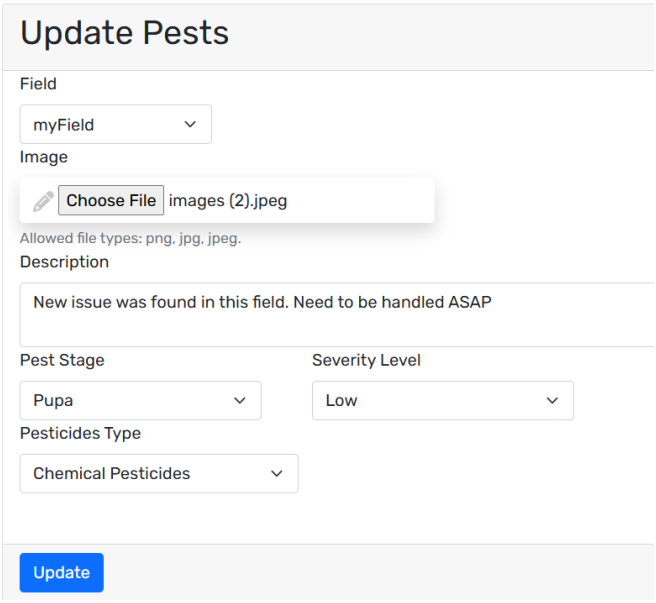


Figure 6.1.47 Input of Update Selected Untreated Pest Issue

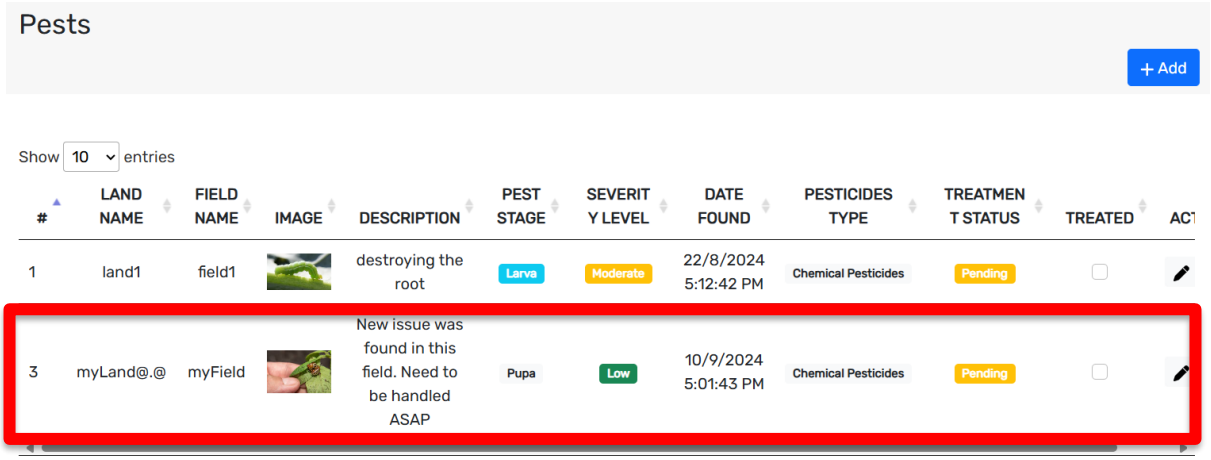


Figure 6.1.48 Output of Successfully Update Selected Untreated Pest Issue

Figure 6.1.47 above shows the input of the updated existing issue. After editing the input and clicking on the update button, it was successfully updated and redirected back to the main page of the pest module. After being redirected, the information on the particular issue has been updated in the list as shown in Figure 6.1.48. Thus, the result of updating existing issue case testing is PASS.

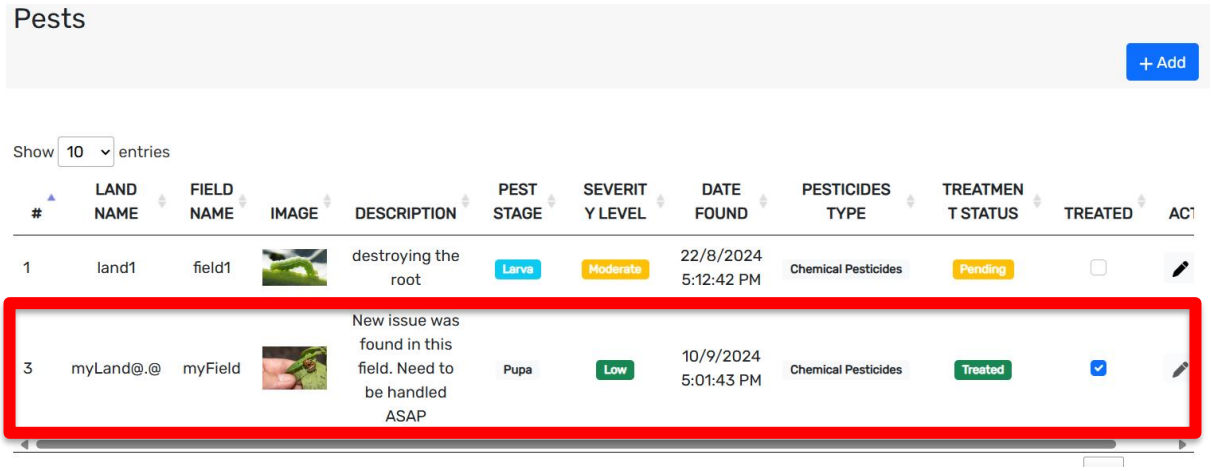


Figure 6.1.49 Input of Update Pest Treatment Status

Figure 6.1.49 above shows the input of updated pest treatment status. Once the farmer solves the issue and checks the checkbox in the "TREATED" column, it will update the treatment status from "Pending" to "Treated". Thus, the result of the updated pest treatment status case testing is PASS.

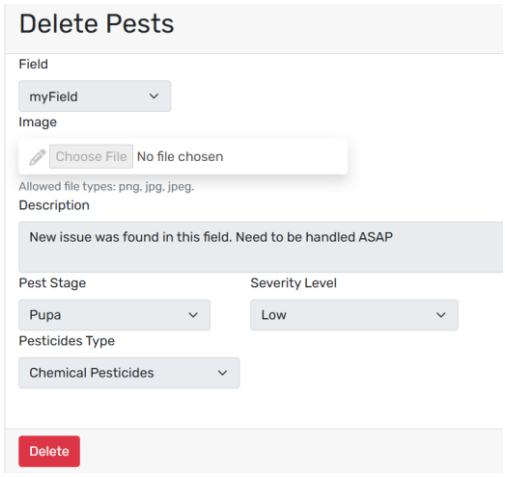


Figure 6.1.50 Input of Delete Selected Pest Issue

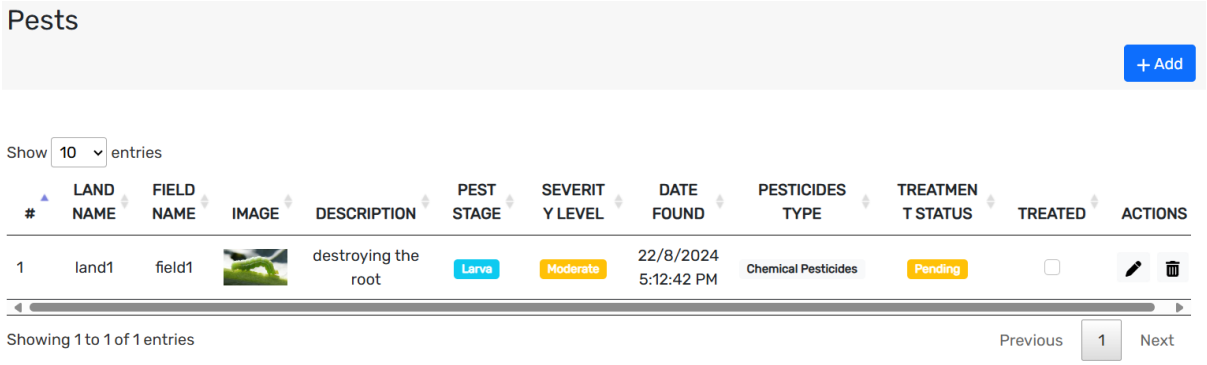


Figure 6.1.51 Output of Successful Delete Selected Pest Issue

Figure 6.1.50 above shows the input of the deleted selected pest issue. After reviewing the issue and confirming to delete it, it will fully delete it from the database and redirect back to the main page of the pest module. In the page, the deleted issue has been removed from the list as shown in Figure 6.1.51. In short, the result of deleting selected pest issue case testing is PASS".

Inventory

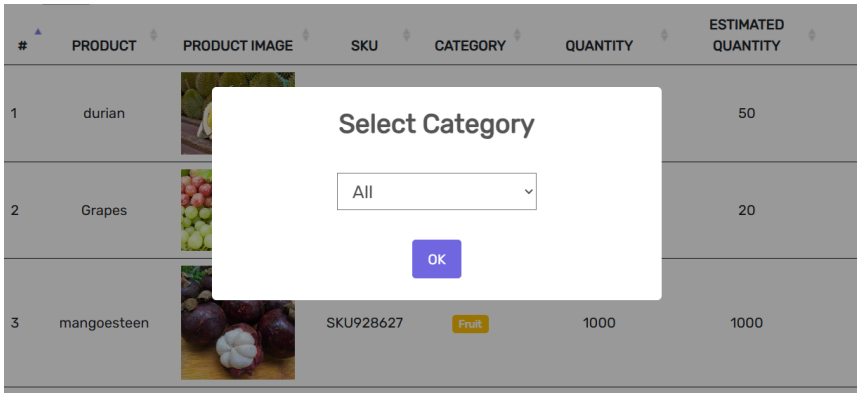


Figure 6.1.52 Input of Export Inventory Data in CSV File

	A	B	C	D	E
1	product	productsku	category	quantity	price
2	durian	SKU00_1	fruit	20	40
3	Grapes	SKU_192	fruit	20	13
4	mangoeste	SKU92862	fruit	1000	20
5	rambutan	SKU00_2	fruit	0	0

Figure 6.1.53 Output of Exported Inventory Data

Figure 6.1.52 shows the input of export inventory data into a CSV file. The farmer can select to download the data by the agricultural product category. Figure 6.1.52 shows the selected category is "All" which all agricultural product data in the inventory will be exported. After successfully downloaded, the inventory data has been exported as shown in Figure 6.1.53. Hence, the result of export inventory data case testing is PASS.

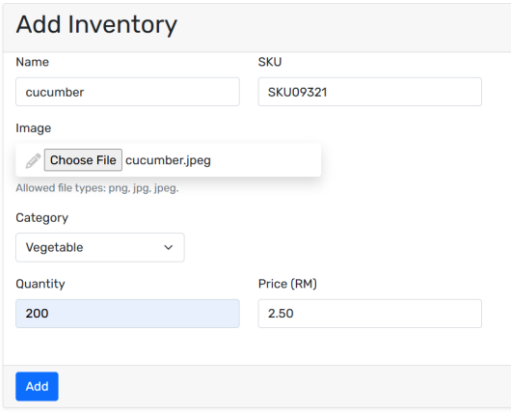


Figure 6.1.54 Add New Agricultural Product in Inventory

#	PRODUCT	PRODUCT IMAGE	SKU	CATEGORY	QUANTITY	ESTIMATED QUANTITY	PRICE PER KG(RM)	ACTIONS
1	durian		SKU00_1	Fruit	20	50	40.00	
2	Grapes		SKU_192	Fruit	20	20	13.00	
3	mangoesteen		SKU928627	Fruit	1000	1000	20.00	
4	rambutan		SKU00_2	Fruit	0	0	0.00	
5	cucumber		SKU09321	Vegetable	200	200	2.50	

Figure 6.1.55 Output of Successfully Added New Agricultural Product in Inventory

Figure 6.1.54 shows the input of adding new agricultural products to the inventory. After entering the necessary data of the agricultural product and confirming to add the product, it will redirect back to the main page of the inventory module and the newly added product will be shown in the inventory list as shown in Figure 6.1.55. Hence, the result of updating new agricultural products in inventory case testing is PASS.

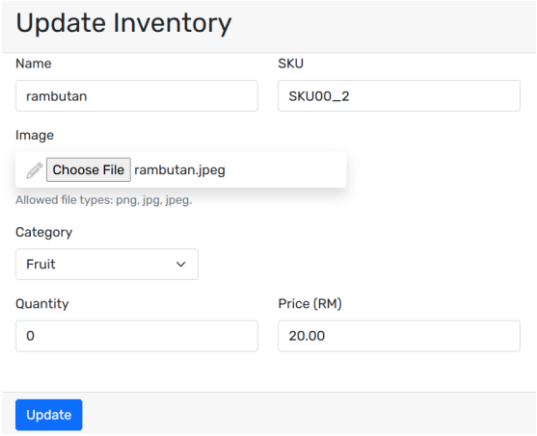


Figure 6.1.56 Input of Update Existing Agricultural Products in Inventory

#	PRODUCT	PRODUCT IMAGE	SKU	CATEGORY	QUANTITY	ESTIMATED QUANTITY	PRICE PER KG(RM)	ACTIONS
1	durian		SKU00_1	Fruit	20	50	40.00	
2	Grapes		SKU_192	Fruit	20	20	13.00	
3	mangoesteen		SKU928627	Fruit	1000	1000	20.00	
4	rambutan		SKU00_2	Fruit	0	0	20.00	
5	cucumber		SKU09321	Vegetable	200	200	2.50	

Figure 6.1.57 Output of Successfully Update Existing Agricultural Products in Inventory

Figure 6.1.56 above shows the input of updated existing agricultural products in inventory. Selecting the product wish to update and edit the information of the product; then, confirm to update it. After successfully updating the product, it will be redirected back to the main page of inventory. On the main page, the particular product has been successfully updated and shown in the list as shown in Figure 6.1.57. To conclude, this updated existing agricultural product in inventory case testing is PASS.

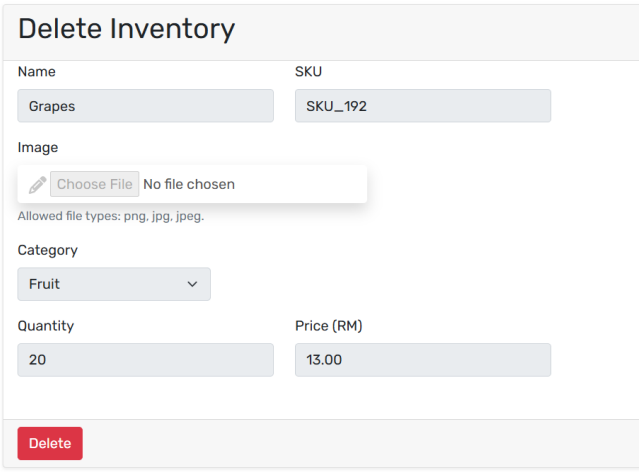


Figure 6.1.58 Input of Delete Existing Agricultural Products in Inventory

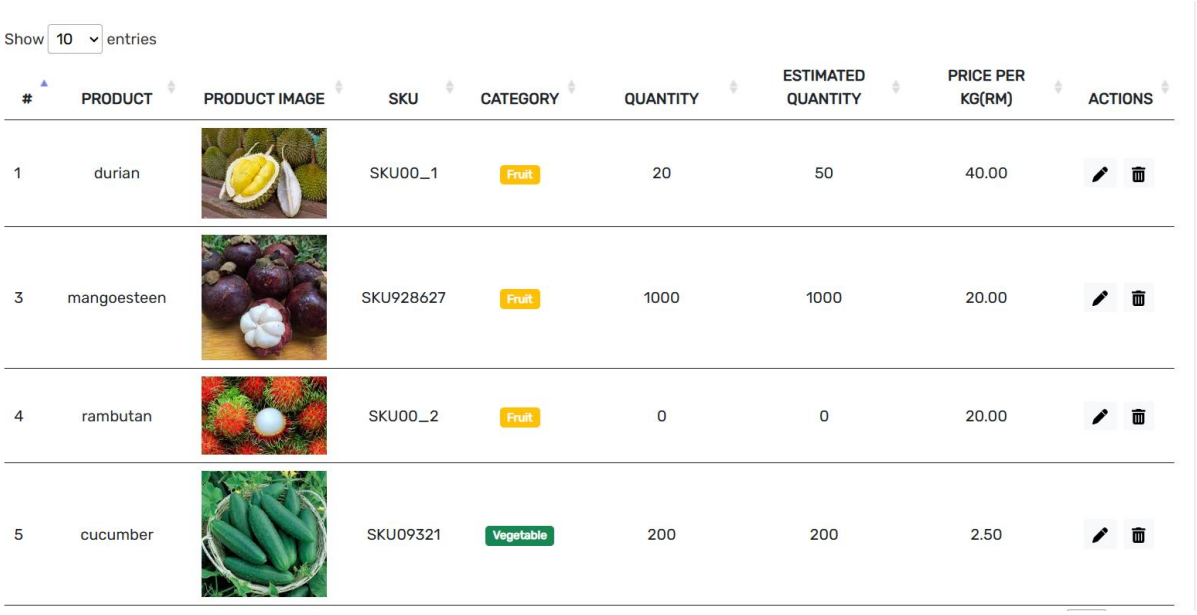


Figure 6.1.59 Output of Successfully Delete Existing Agricultural Products in Inventory

Figure 6.1.58 above shows the input of deleted existing agricultural products in inventory. After reviewing the product and confirming to delete it, it will redirect back to the main page of the inventory module. After being redirected, the deleted product no longer existed in the list as shown in Figure 6.1.59 above. Hence, the result of deleting existing agricultural product case testing is PASS.

Fertilizer Resources

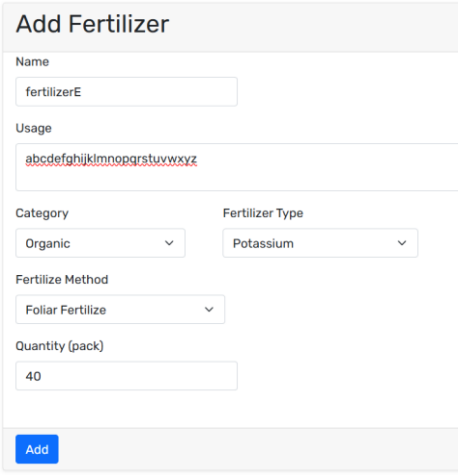


Figure 6.1.60 Input of Add New Fertilizers

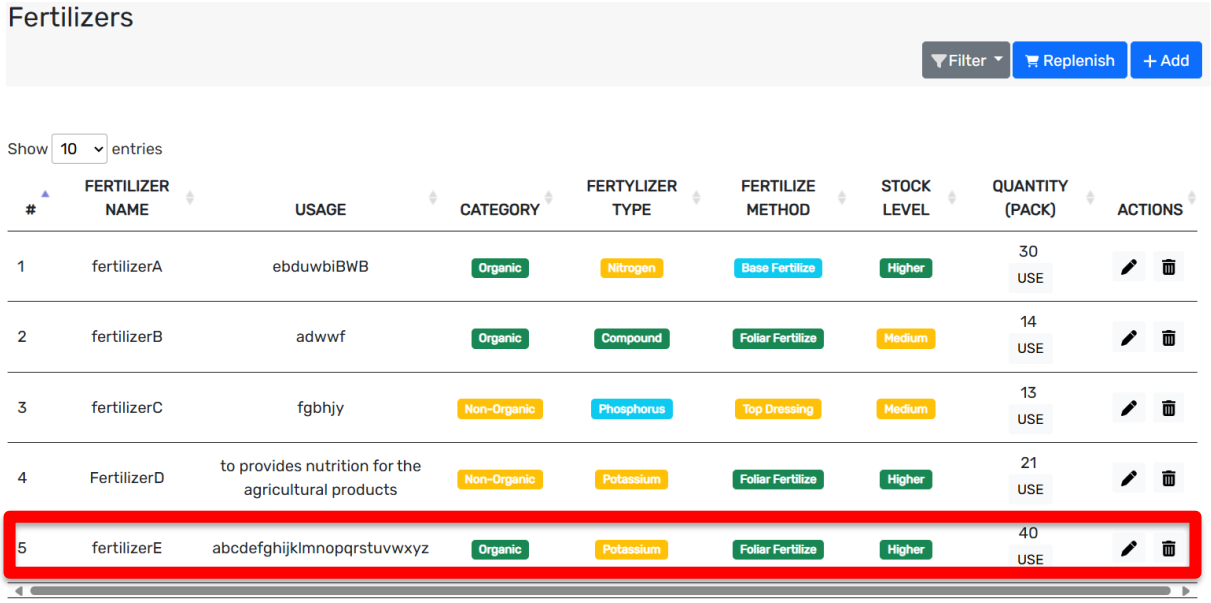


Figure 6.1.61 Output of Successfully Add New Fertilizers

Figure 6.1.60 above shows the input of adding new fertilizer resources. After entering all the necessary details of the fertilizers and confirming to add, it will redirect back to the main page of the fertilizers module. In this page, it will show the newly added fertilizers are successfully stored and listed in the table as shown in Figure 6.1.61 above. Hence, the result of adding new fertilizer case testing is PASS.

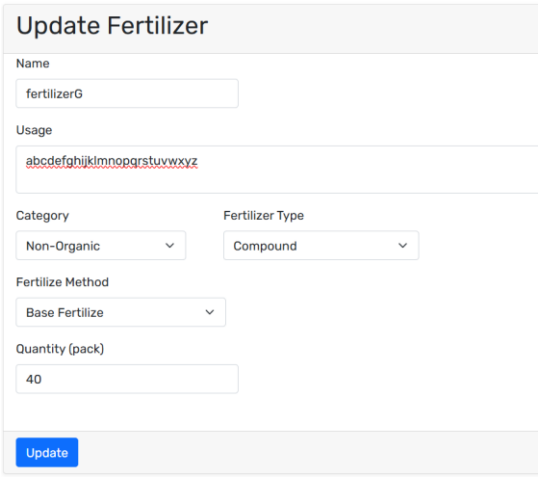


Figure 6.1.62 Input of Update Existing Fertilizers

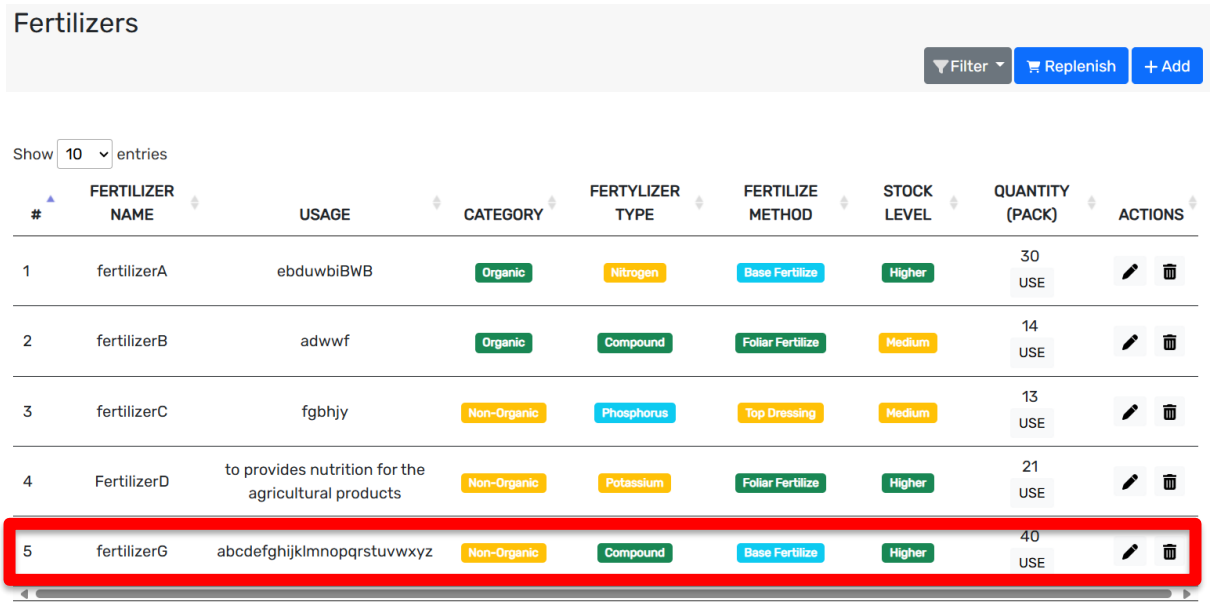


Figure 6.1.63 Output of Successfully Update Existing Fertilizers

Figure 6.1.62 above shows the input of updated existing fertilizers. After editing the particular fertilizers and confirming the update, it will be redirected back to the main page of the fertilizers module and the particular fertilizers have been successfully updated in the list as shown in Figure 6.1.63 above. Thus, the result of updated existing fertilizers case testing is PASS.

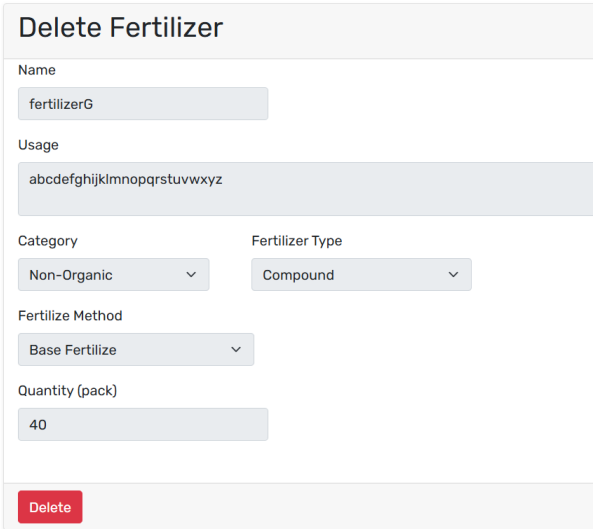


Figure 6.1.64 Input of Delete Existing Fertilizer

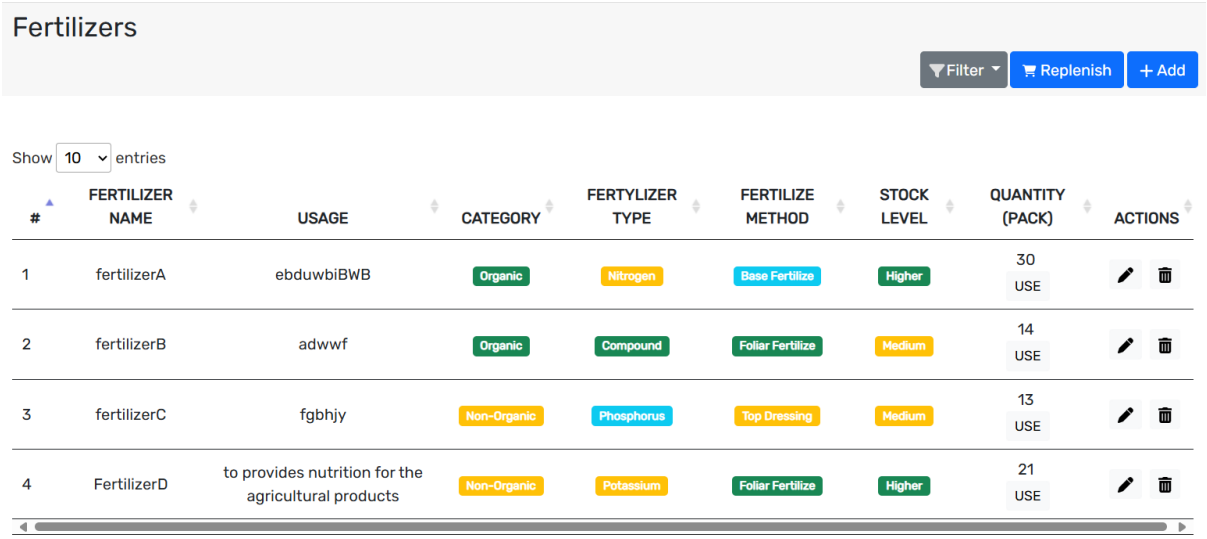


Figure 6.1.65 Output of Successfully Delete Existing Fertilizer

Figure 6.1.64 above shows the input of deleting existing fertilizer. After reviewing the information on the particular fertilizer and confirming to delete it, it will redirect back to the main page of the fertilizer module. Once redirected back, the deleted fertilizer no longer shows in this list as shows in Figure 6.1.65. Hence, the result of deleting existing fertilizer case testing is PASS.

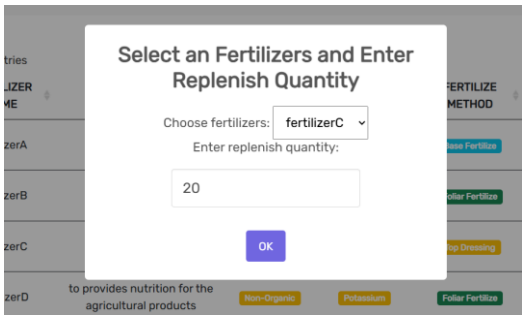


Figure 6.1.66 Input of Replenish Fertilizer

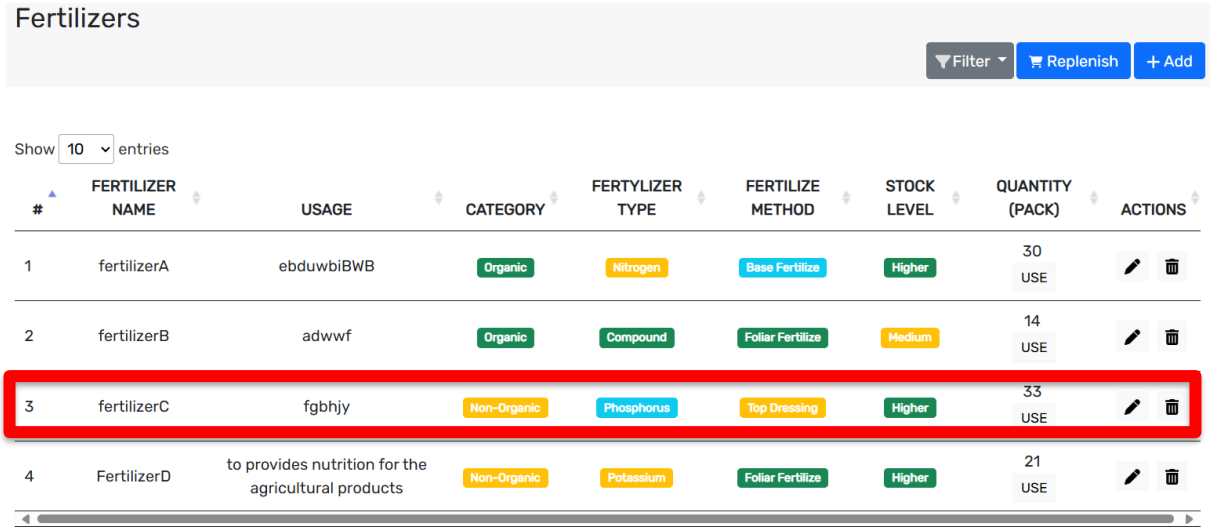


Figure 6.1.67 Output of Successfully Replenish Fertilizer

Figure 6.1.66 above shows the input of replenishing the particular fertilizer. When the farmer clicks the "Replenish" button, the replenish modal is opened. The farmer is required to select a fertilizer and enter a quantity. After being confirmed to replenish, the modal is closed and the quantity of the fertilizer has increased as shown in Figure 6.1.67 above. Therefore, the result of replenished fertilizer case testing is PASS.

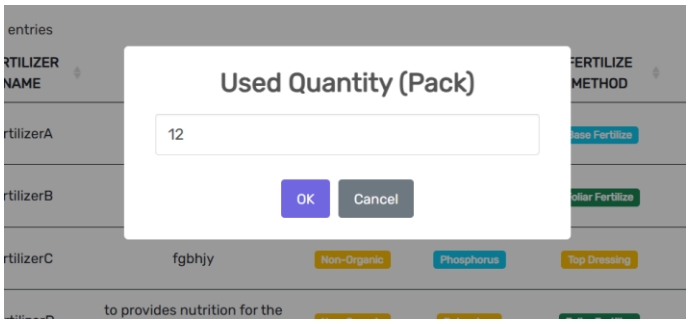


Figure 6.1.68 Input of Deduct Fertilizer Quantity

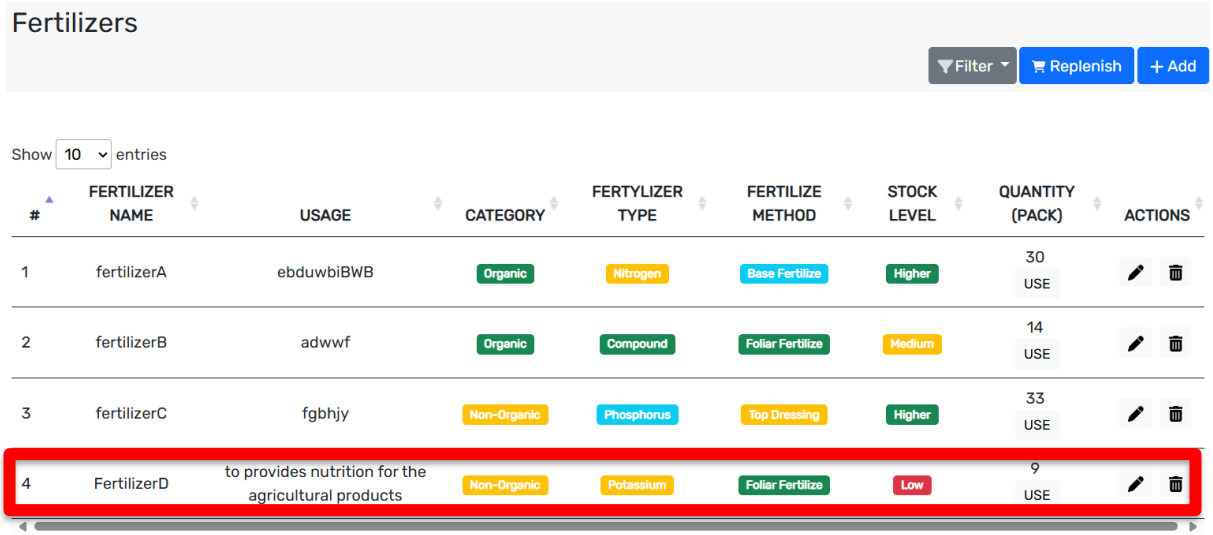


Figure 6.1.69 Output of Successfully Deduct Fertilizer Quantity

Figure 6.1.68 shows the input of deducting fertilizer quantity. When the farmer clicks on the "USE" button for the particular fertilizer row, it will open a modal, and the farmer is required to enter the used amount. After confirming to deduct the quantity, the modal will close and the quantity is updated as shown in Figure 6.1.69 above. Hence, the result of deduct fertilizer quantity case testing is PASS.

Seeds

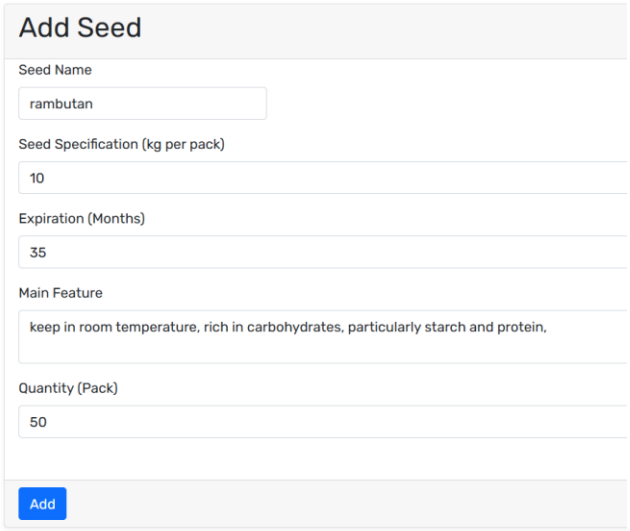


Figure 6.1.70 Input of Add New Seeds

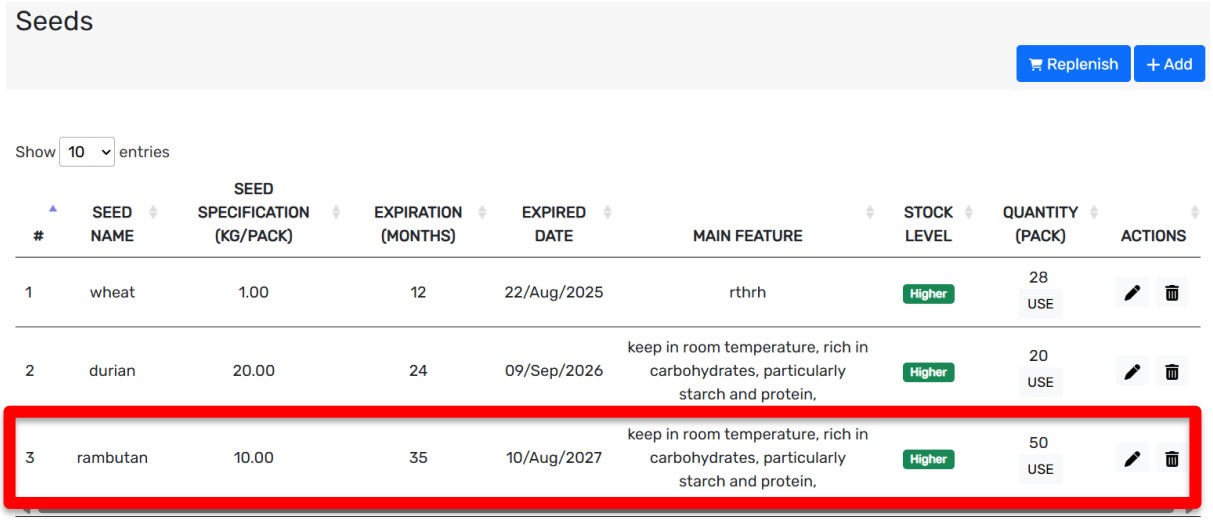


Figure 6.1.71 Output of Successful Add New Seeds

Figure 6.1.70 shows the input of adding new seeds into the resource. The farmer is required to input all necessary information about the seeds into each column. After confirming to add it to the resources, it will redirect back to the main page of the seeds module, and the newly added seeds are successfully added and shown in the list as shown in figure 6.1.71 above. Hence, the result of adding new seed case testing is PASS.

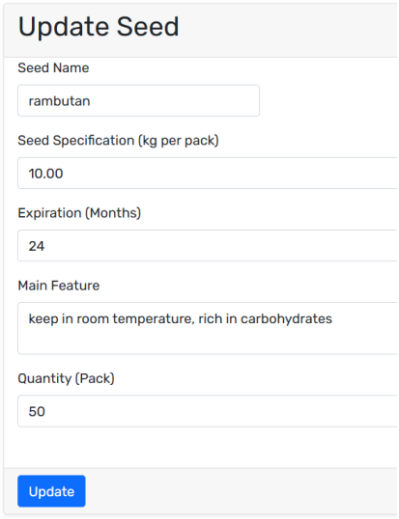


Figure 6.1.72 Input of Update Existing Seed

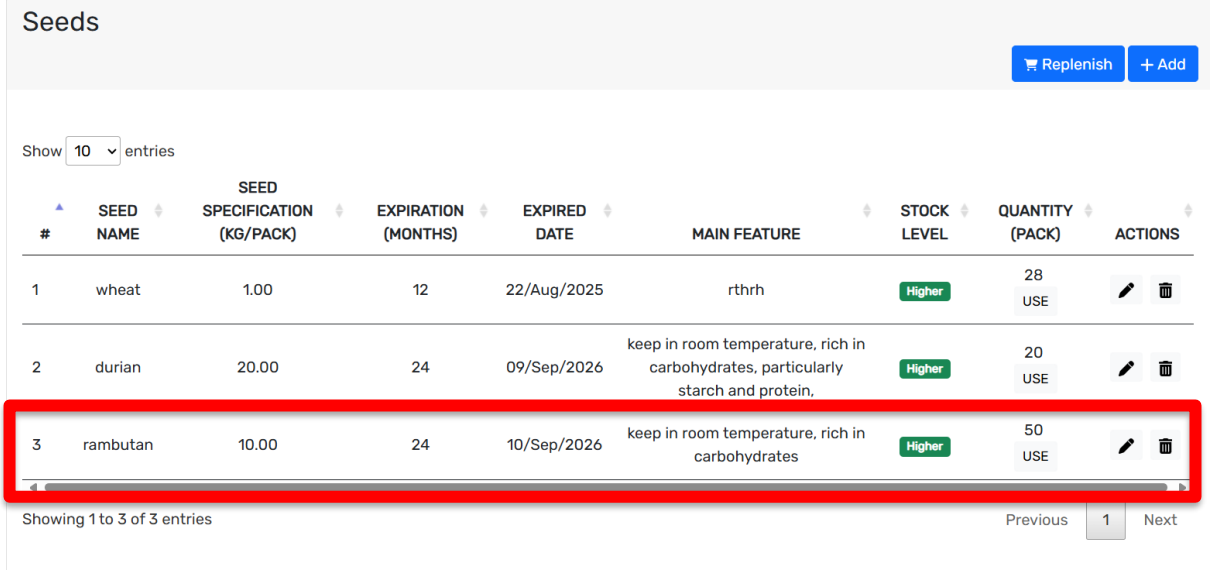


Figure 6.1.73 Output of Successfully Update Existing Seed

Figure 6.1.72 above shows the input of updated existing seeds in the resources. The farmer reviewed and edited the particular seed information. After editing and confirming the update, it will redirect back to the main page of the seeds module and the particular seed information in the list has been updated as shown in Figure 6.1.73 above. Thus, the result of updating existing seed case testing is PASS.

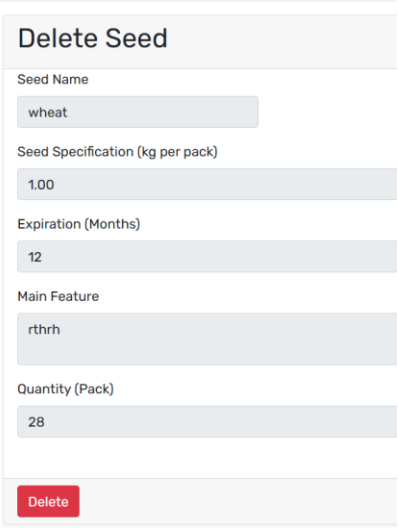


Figure 6.1.74 Input of Delete Existing Seed

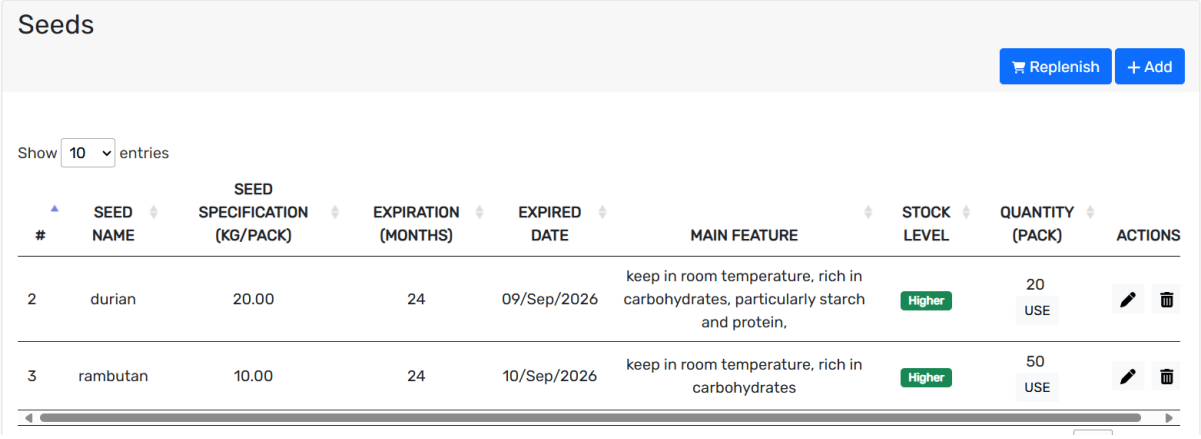


Figure 6.1.75 Output of Successfully Delete Existing Seed

Figure 6.1.74 above shows the input of deleting a particular seed. After the farmer reviews and confirms to delete, it will redirect back to the main page of the seed module. The deleted seed has been successfully deleted from the seed resource list as shown in figure 6.1.75 above. Thus, the result of deleting existing seed case testing is PASS.

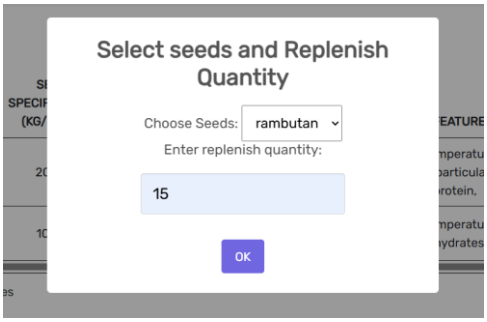


Figure 6.1.76 Input of Replenish Seed Quantity

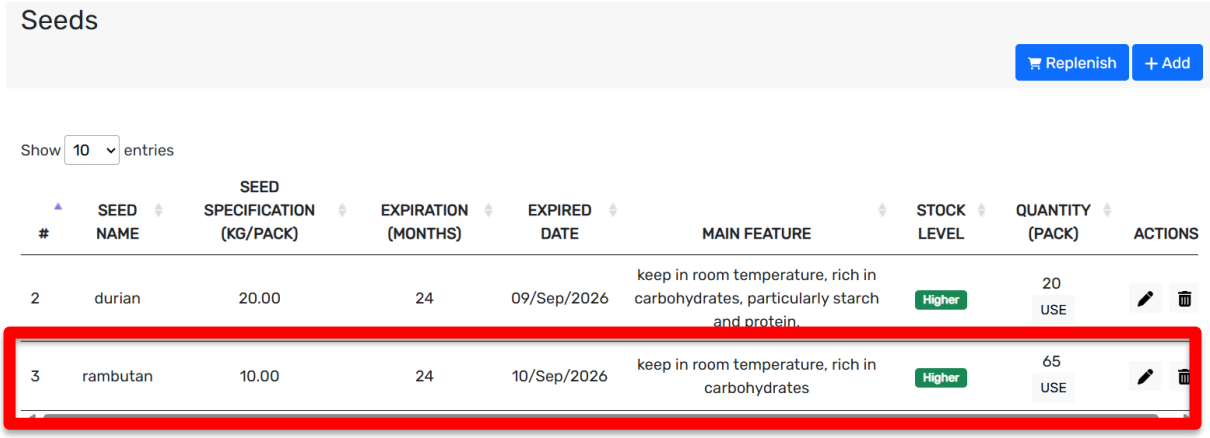


Figure 6.1.77 Output of Successfully Replenish Seed Quantity

Figure 7.1.76 shows the input of replenishing a particular seed quantity. When the "Replenish" button is clicked, a modal is opened and the farmer needs to select a seed and enter the replenished quantity. After confirming to replenish, the modal is closed and the quantity of

the particular is updated as shown in figure 6.1.77 above. So, the result of replenishing seed quantity is PASS.

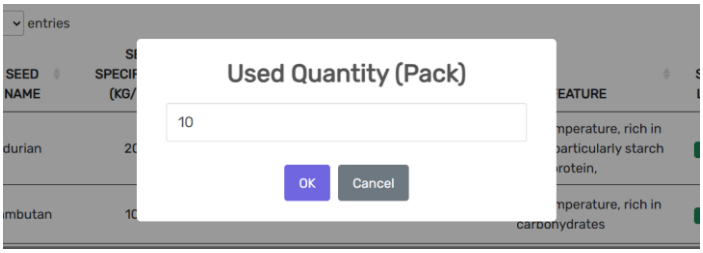


Figure 6.1.78 Input of Deduct Seed Quantity

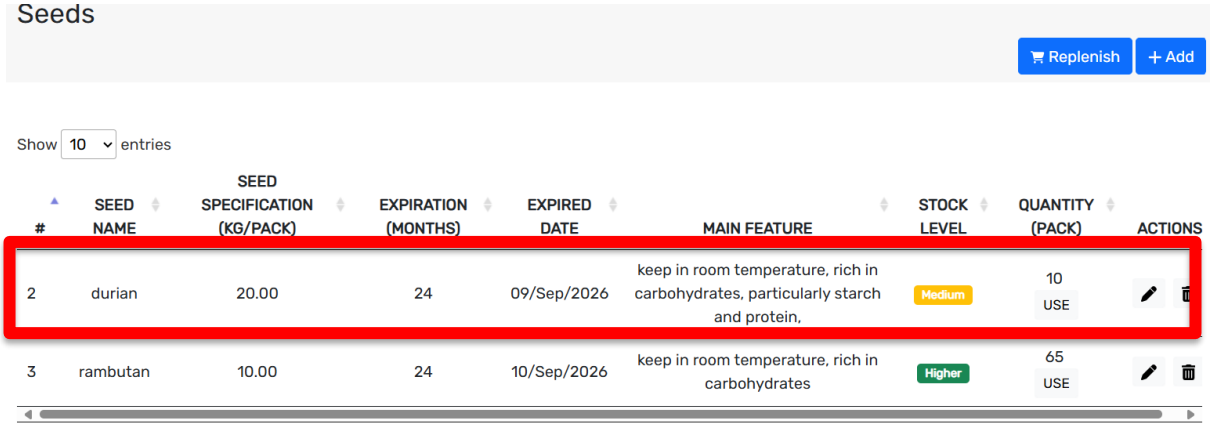


Figure 6.1.79 Output of Successfully Deduct Seed Quantity

Figure 6.1.78 above shows the input of deducting a particular seed quantity. The farmer clicks on the "USE" button for the particular seed; then, a modal is opened and enters the quantity to be deducted. After confirming to deduct, the modal is closed and the quantity of the particular seed is deducted successfully as shown in Figure 6.1.79 above. Thus, the result of deduct seed case testing is PASS.

Announcement

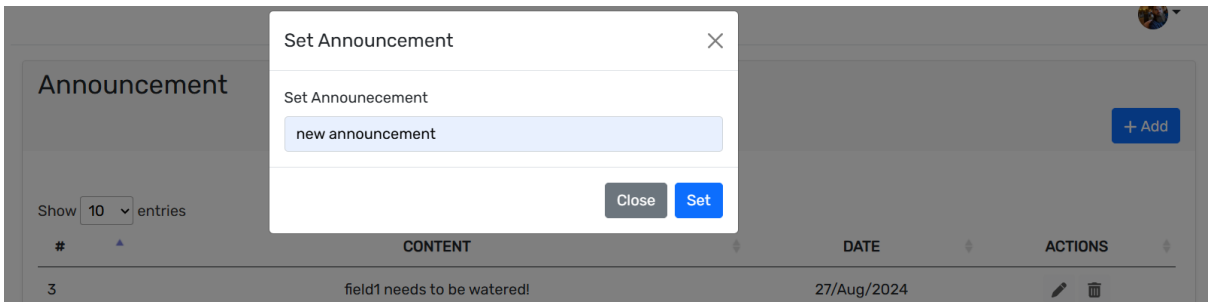


Figure 6.1.80 Input of Make New Announcement

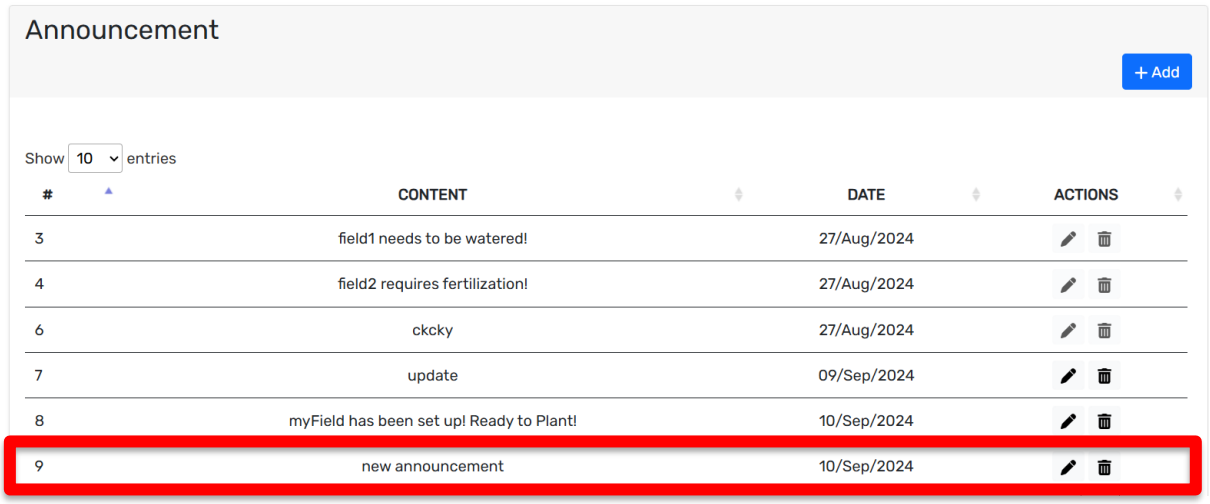


Figure 6.1.81 Output of Successfully Add New Announcement

Figure 6.1.80 shows the input of making a new announcement. When the farmer clicks on the "Add" button, a modal is opened and asked to enter the content of the new announcement. After entering and confirming to set the announcement, the modal is closed and the announcement is posted as shown in Figure 6.1.81 above. Thus, the result of make announcement case testing is PASS.

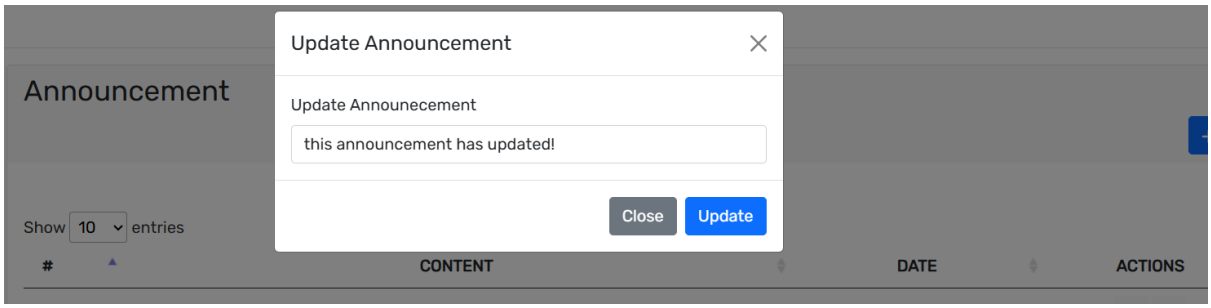


Figure 6.1.82 Input of Update Announcement Content

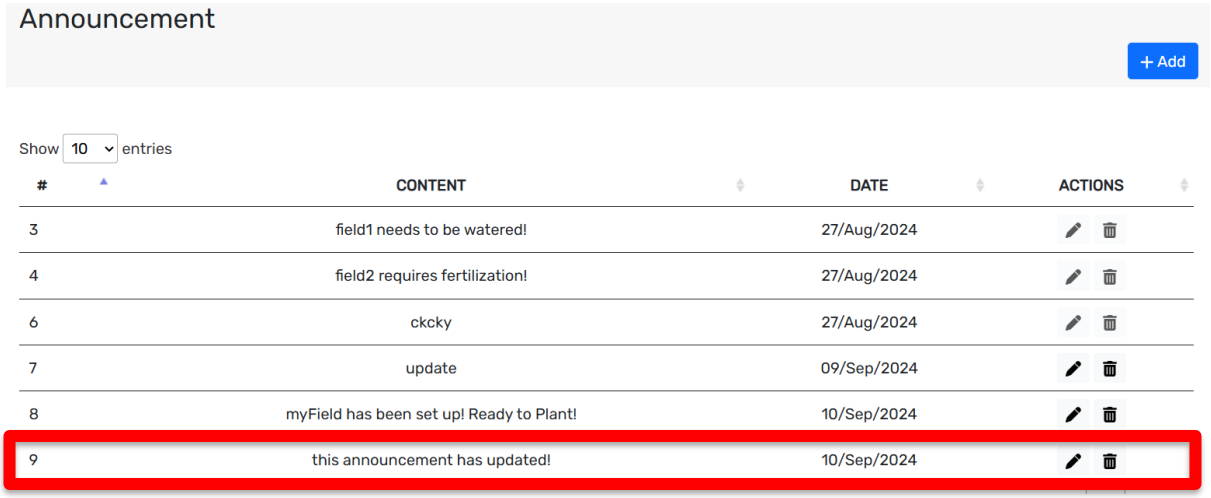


Figure 6.1.83 Output of Successfully Update Announcement

Figure 6.1.82 shows the input of the update of the posted announcement. However, the farmer can only update the announcement that is created by himself or herself. By clicking on the "Update" button, a modal is opened and the farmer has to enter the updated announcement. After confirming to update the announcement, the modal will close and the particular announcement content will be updated as shown in Figure 6.1.83 above. Thus, the result of update announcement content case testing is PASS.

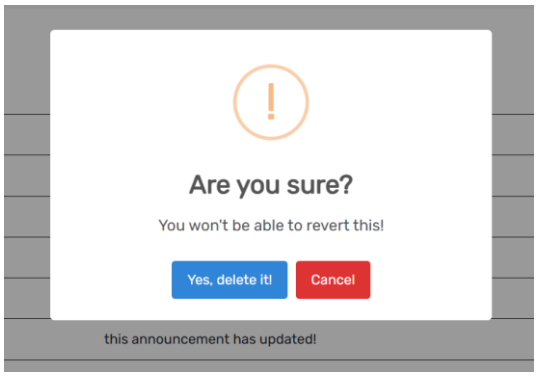


Figure 6.1.84 Input of Delete Announcement

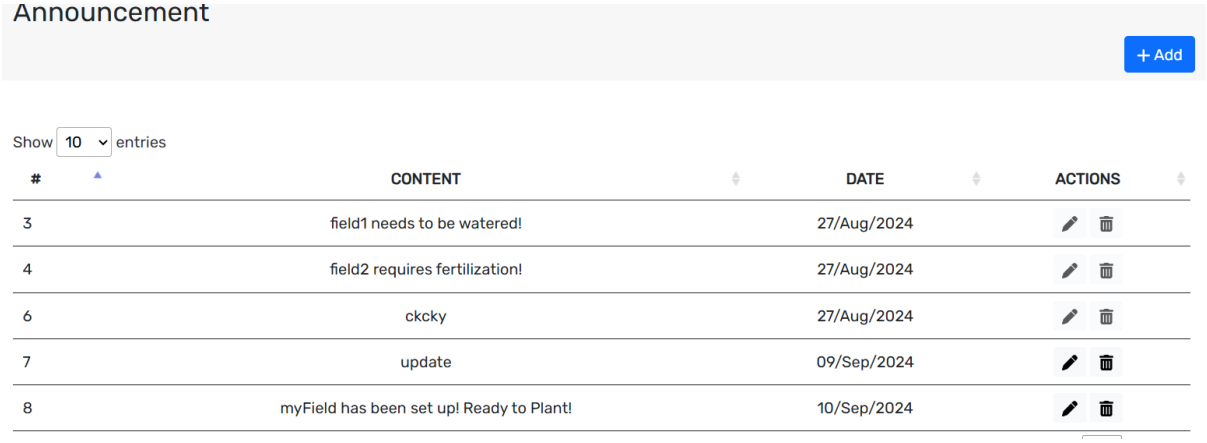


Figure 6.1.85 Output of Successfully Delete Announcement

Figure 6.1.84 above shows the input of the delete announcement. Once the farmer clicks on the "Delete" button for the particular announcement made by himself or herself, it will show this delete confirmation message. After confirming to delete, the modal is closed and the announcement is successfully deleted as shown in Figure 6.1.85. Hence, the result of the delete announcement case testing is PASS.

Chat

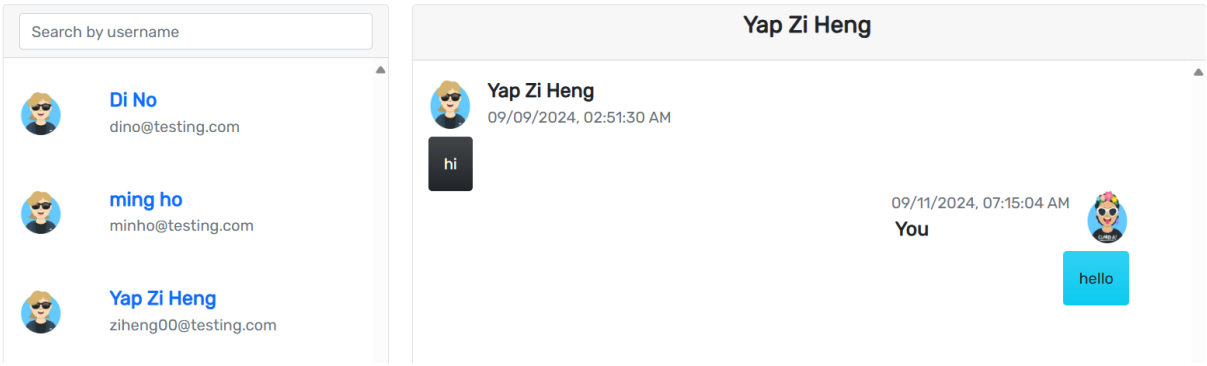


Figure 6.1.86 Input of Send Message to Others Farmer

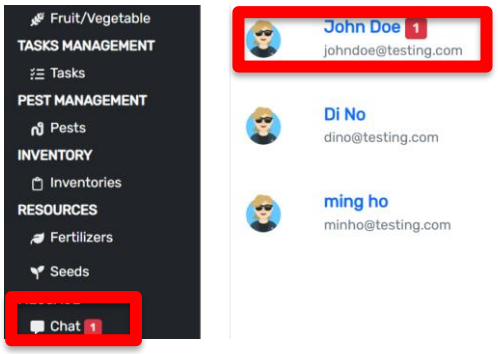


Figure 6.1.87 Output of Send Message Notification to Farmer

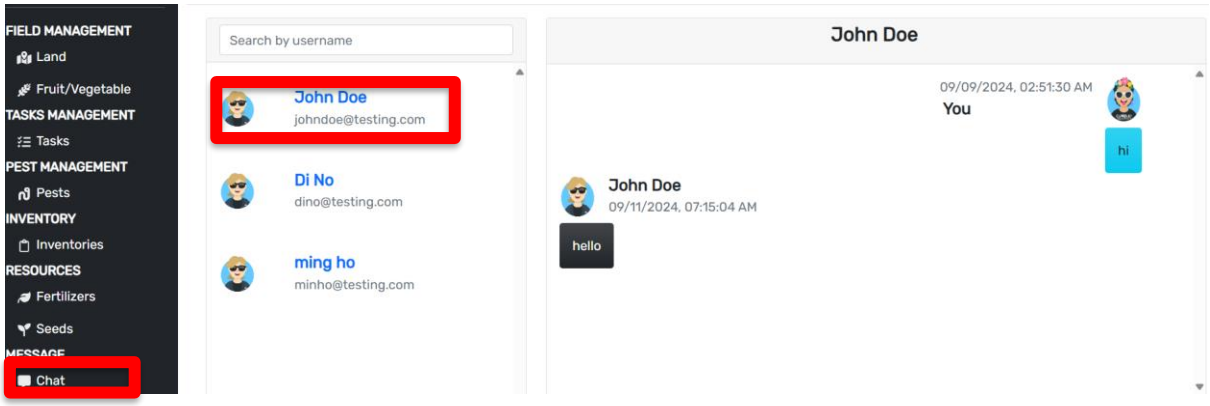


Figure 6.1.88 Output of Update Message Status

Figure 6.1.86 above shows the input of sending messages to other farmers. I have logged in as “John Doe” to send a message to “Yap Zi Heng”. In Figure 6.1.87 above, I have logged in to the “Yap Zi Heng” account and it shows that he or she has successfully received the message notifications. Once the receiver, “Yap Zi Heng” has opened the “John Doe” chat, the message status has been changed from “unread” to “read” which the notification in the user list has disappeared, and the count of total unread messages in the sidebar has been deducted as shown in figure 6.1.88. Thus, the result of send message case testing is PASS.

6.2 Objective Evaluations

Overall, this web application project has included three objectives which are inventory estimation, automatically assigning several tasks at once based on task frequency and period of the task, and determining the growth stage of the agricultural products automatically. These 3 objectives have been successfully achieved.

The first objective is inventory estimation which the estimated total harvest amount of the agricultural products has been able to calculated accurately in the inventory. This estimation has also been visualized and displayed in a line graph which shows the various agricultural products estimated to be harvested in which month. Thus, the farmer can determine the amount of the products that are required to plant to ensure the quantity of the products is always sufficient for the farmer to supply and deliver to market.

The second objective is to assign several tasks at once based on task frequency and period of task. This objective is successfully achieved in this project. The due date for every task is calculated correctly and the notifications are successfully sent to the assignee. This objective can prevent the farmer forget to assign tasks on time and last minute which may affect the growth of the cultivating agricultural product.

The last objective of this project is to determine the growth stage of agricultural products automatically. This objective has saved the farmer's time in visiting the field and observing the growth stage of the agricultural products. Besides, this objective can also allow the farmer to acknowledge that cultivating agricultural products is moving to the next stage on time. Thus, this objective has benefits to the farmer to plan or modify the decision planning on the farming activity for the cultivating products.

This project also included several modules that are not related to the objectives which are pest management, announcement, and, seeds and fertilizers resources. However, these modules can improve the efficiency of the daily farming operations. These modules are useful for the farmer to guarantee their cropping cultivation as these can ensure the nutrition and growth of the agricultural product cultivations.

In short, by implementing this project, the farmer can improve their crop cultivations as well as ensure that agriculture can be supplied to the market on time and is in fresh condition.

6.3 Concluding Remarks

To conclude, this chapter has covered use case testing and objective evaluations of this web application project. From the use case testing, the sub-flows of each case can operate smoothly and the expected outcome is successfully performed in this web application. For objective

evaluations, I have summarized the 3 objectives and the other modules that have been implemented in this web application development. These summarizations have shown that able to improve the efficiency of daily farming operations and agricultural products cultivated to supply all fresh agricultural products into the market to prevent food shortage and ensure food security.

Chapter 7
Conclusion

7.1 Conclusion

In conclusion, this agriculture product supply chain management is a web application able to improve farming production and management efficiency. It can be used to solve the problem of the lack of inventory estimation for predicting the total stock amount in the inventory, the well-planned task of farming activities to cultivate agricultural products, and the calculation to determine the growth stage of agricultural products. These problems may cause the poor performance in the farming operations.

The motivation of this project is to tackle the issues the farmer may face when using the farming management system or application. The approaches of this project to meet the motivation include integrating an inventory estimation function, implementing automatic calculation of task due date based on selected timeline and frequency in task management, and implementing automatic calculation of cycle growth stage of agricultural products.

This web application development project will include user authentication, field management, inventory management, task management, pest management, resource management, chat, and announcement. These allow the farmers to collaborate with other farmers to manage the farming operation smoothly as every farmer can overview the other's cultivation activity, assigning tasks for each other as well as managing the issue and farming resources. These are useful for farmers to communicate with others and monitor the farming operations and process.

In this project, I have proposed motivational solutions by providing various functions for the farmers to solve the issues in the similar existing farming management system. It can enhance farmers' productivity in the process of managing the farming operation.

7.2 Recommendations

Even though this web application has provided various benefits for the farmer to operate their farming operation; however, it may only be useful for the farmer who is operating a small farming business. To develop it for the farmers that running bigger farming operations and businesses, it is suggested to integrate various big data analytics, artificial intelligence (AI), and the Internet of Things (IoT). By integrating AI and IoT in this web application, it is

convenient for the farmer to observe the soil condition in detail as big data analytics has been applied with the IoT to analyse the soil.

Moreover, the farmer can implement the IoT for automatic watering, harvesting, and harvest quantity estimation of the agricultural products in their operations and integrate with this web application to monitor these actions on the remote side. This is useful for the farmers to reduce their workload.

Last but not least, it is also recommended to integrate machine learning into this web application. This machine learning is useful for the farmer to observe the trends of market demands of the various agricultural products. Besides, it also helps the farmers to determine the current market price of agricultural products. This is convenient for them to set the prices of each agricultural product to supply into the market.

REFERENCES

[1] M. Alex, "SUPPLY CHAIN MANAGEMENT," p. 50, Sep. 2022, doi: https://doi.org/10.13140/RG.2.2.16353.38241.

[2] Y. Handayati, T. M. Simatupang, and T. Perdana, "Agri-food supply chain coordination: the state-of-the-art and recent developments," *Logistics Research*, vol. 8, no. 1, Oct. 2015, doi: https://doi.org/10.1007/s12159-015-0125-4.

[3] A. Studoc, "Supply Chain Management In Agriculture: 7 Importance," Jan. 09, 2024. https://www.agristudoc.com/supply-chain-management-in-agriculture/

[4] T. oboloo Team, "Inventory Estimation and Procurement: How to Equip Yourself for Accurate," *oboloo*, May 06, 2023. https://oboloo.com/blog/inventory-estimation-and-procurement-how-to-equip-yourself-for-accurate-results/ (accessed Apr. 07, 2024).

[5] F. Admin, "Components - 365FarmNet," *365FarmNet*, Oct. 06, 2023. https://www.365farmnet.com/en/products/components/

[6] "iAgri Farm Management Agricultural Software," *www.iagri.com*. https://www.iagri.com/

[7] "Financial Management: The Key to Farm-Firm Business Management Risk Management." Accessed: Apr. 12, 2024. [Online]. Available: https://agecoext.tamu.edu/wp-content/uploads/2013/10/rm5-10.pdf

[8] "Conservis | Farm Management Software | Agriculture Apps | Farming Apps," *conservis.ag*. https://conservis.ag/

[9] R. M. Pujahari, S. P. Yadav, and R. Khan, "Intelligent farming system through weather forecast support and crop production," *Application of Machine Learning in Agriculture*, pp. 113–130, 2022, doi: https://doi.org/10.1016/b978-0-323-90550-3.00009-6.

[10] V. Cherlinka, "Stages of plant growth and their role in agriculture," *EOS Data Analytics*, Jan. 01, 1970. https://eos.com/blog/stages-of-plant-growth/

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 2**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Review back preliminary work of FYP 1
- Modified the web application based on received feedback
- Solve issue encountered in FYP 1

2. WORK TO BE DONE

- Add new information regarding the planting data for various products
- Modify the objective
- Research more information related on agricultural web application

3. PROBLEMS ENCOUNTERED

- Failed to integrate with "geolocation" in the add new land module to search for a location.
- Some source code that the Microsoft visual studio failed to process but the code was running smoothly in previous.

4. SELF EVALUATION OF THE PROGRESS

- Need to understand more farming business and operation knowledge
- Lacking of technical knowledge to achieve some of the feature properly

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 4**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Added new information regarding the planting data for various products
- Modified the previous objectives
- Researched more information related on agricultural web application

2. WORK TO BE DONE

- Achieve new modified objectives
- Convert to features or module from the new researched info

3. PROBLEMS ENCOUNTERED

- Tried to ask the farmers to get more information, but did not receive any replies.
- Faced some technical issue from the Microsoft Visual Studio
- Some of the data calculation is not calculated accurate

4. SELF EVALUATION OF THE PROGRESS

- Slightly low efficiency
- Careless on the calculation and retrieving data from database
- Need to familiarise technical issue either from database or Microsoft Visual Studio.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 6**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Drafted out the prototype of new modified objective
- New module, resources coded into this project
- Added new features in existing modules of this project

2. WORK TO BE DONE

- Continue achieve the final outcome of the new modified objective
- Try to find out unique feature can be added into this project
- Modify the homepage design by adding more important info

3. PROBLEMS ENCOUNTERED

- Unfamiliar on writing the code for exporting data into various type of file
- Database server crashed in the middle of developing this project and forced to close the database server for at least 1 day.

4. SELF EVALUATION OF THE PROGRESS

- Required to learn more on writing the code for exporting data into various file in different ways to successfully export the file from the web application
- Need to understand various problem might encountered when using XAMPP database server to find out the better solutions

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 8**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Completed code for new modified objective.
- Completed modify the homepage design and new important information is added
- Found new feature did not include in existing system and prototyped
- Completed coding of export data into CSV file

2. WORK TO BE DONE

- Add new dashboard in existing modules
- Modify the notification function in task module
- Code for new feature in farm field and resource module
- Code for chat module

3. PROBLEMS ENCOUNTERED

- Crashed of XAMPP database server again in the middle of development
- Unable imported the database backup file of this project after reinstalled XAMPP in my laptop
- Spend longer time to think of new feature can include in this project.

4. SELF EVALUATION OF THE PROGRESS

- Need to always ensuring to backup the correct database file.
- Need to improve the efficiency on developing for this project

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 10**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Added new dashboard in existing modules
- Modified the notification function in task module including overdue task notification
- Coded for new feature in farm field and resource module
- Coded for chat module

2. WORK TO BE DONE

- Conduct system testing for this project
- Debug the issue found in project
- Preparing for FYP 2 Report

3. PROBLEMS ENCOUNTERED

- Some visualization data in dashboard found not display accurately

4. SELF EVALUATION OF THE PROGRESS

- Improved the efficiency on developing project
- Lacking on retrieving data for calculation from the database when the SQL query was complexed.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: T3Y3	**Study week no.: 12**
Student Name & ID: Yap Zi Xin 20ACB05249	
Supervisor: Ts. Tan Teik Boon	
Project Title: Agriculture Product Supply Chain Management Application	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Conducted system testing of this project
- Debugged issue found in system testing
- Received the guidance on writing FYP 2 report

2. WORK TO BE DONE

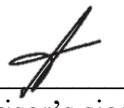
- Complete FYP 2 report
- Prepare for FYP 2 presentation
- Improve the web application before FYP 2 presentation

3. PROBLEMS ENCOUNTERED

- -

4. SELF EVALUATION OF THE PROGRESS

- Enhanced the efficiency in this progress, and successfully debugged issue found in system testing process.



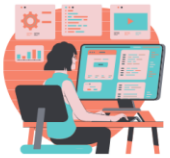
Supervisor's signature

Student's signature

POSTER

AGRICULTURE PRODUCT SUPPLY CHAIN MANAGEMENT APPLICATION

1 Introduction



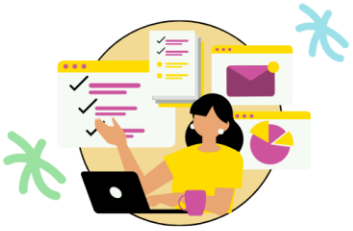
It is a web application for the farmers to enhance the efficiency on operating and managing the farm.

3 Discussion

Shortage of agriculture products is a serious issue around the globe due to the inefficiency of managing the farm. By integrating with an application able to manage the farming operations efficiently and systematically.

2 Method

- Implementing Agile Methodology to develop this web application from planning phase, design phase, development phase, testing phase until review phase.
- Using ASP.NET C# to develop this web application.
- Storing all the farming data in MySQL database.

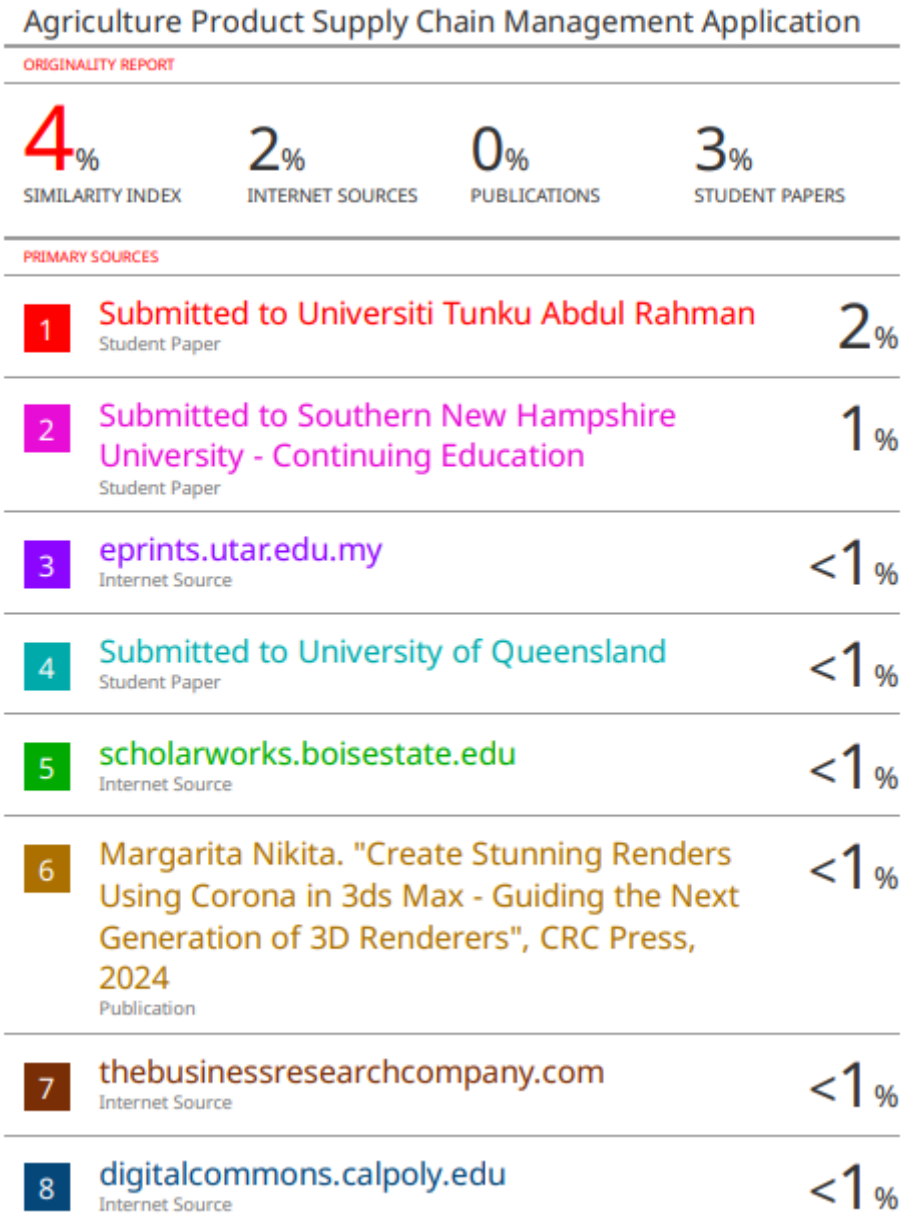


4 Conclusion

This project able to improve the efficiency of managing a farm and maximise farming performance. It allowing the farmers to ensure the agriculture product able to be supplied into the market on time and it is in fresh condition.

Developed By Yap Zi Xin
Supervisor: Ts. Tan Teik Boon

PLAGIARISM CHECK RESULT



Agriculture Product Supply Chain Management Application

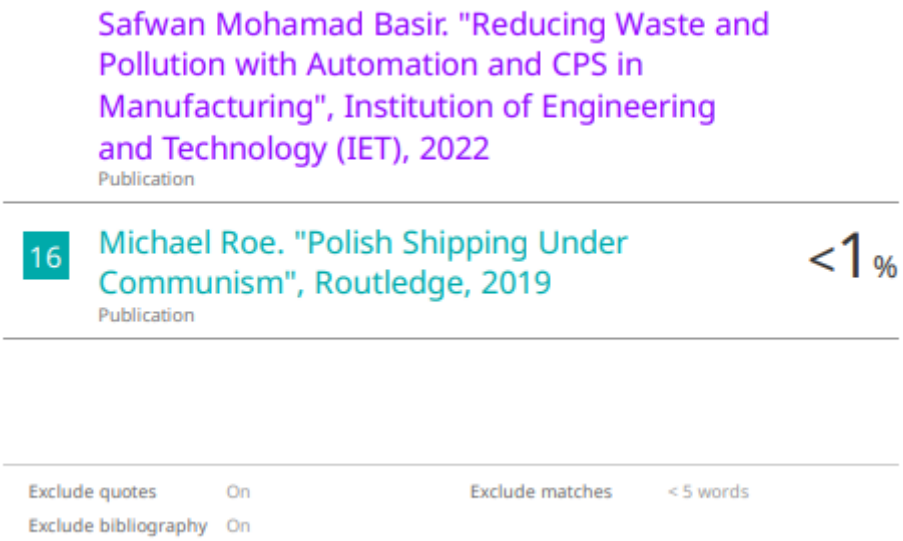
ORIGINALITY REPORT

4%	2%	0%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Universiti Tunku Abdul Rahman Student Paper	2%
2	Submitted to Southern New Hampshire University - Continuing Education Student Paper	1%
3	eprints.utar.edu.my Internet Source	<1%
4	Submitted to University of Queensland Student Paper	<1%
5	scholarworks.boisestate.edu Internet Source	<1%
6	Margarita Nikita. "Create Stunning Renders Using Corona in 3ds Max - Guiding the Next Generation of 3D Renderers", CRC Press, 2024 Publication	<1%
7	thebusinessresearchcompany.com Internet Source	<1%
8	digitalcommons.calpoly.edu Internet Source	<1%

9	Andrew Oplas, Maria Himaya Rabago, Charlene Louise Tormes, Cherry Lyn Sta. Romana, Ralph Laviste. "Aeon: A Smart Medicine Delivery and Inventory System for Cebu City Government's Long Life Medical Assistance Program", 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM), 2018 Publication	<1 %
10	Submitted to UOW Malaysia KDU University College Sdn. Bhd Student Paper	<1 %
11	Submitted to University of Greenwich Student Paper	<1 %
12	blueshift.tistory.com Internet Source	<1 %
13	dt.athabascau.ca Internet Source	<1 %
14	Masahiko Higashino, Shin Okamoto. "Response Control and Seismic Isolation of Buildings", Taylor & Francis, 2006 Publication	<1 %
15	Shuh Huey Ho, Nurul Izzatul Akma Katim, Mohd Sabirin Rahmat, Muhammad Sufyan	<1 %


Safwan Mohamad Basir. "Reducing Waste and Pollution with Automation and CPS in Manufacturing", Institution of Engineering and Technology (IET), 2022
Publication
16
Michael Roe. "Polish Shipping Under Communism", Routledge, 2019
Publication
<1%
Exclude quotes On
Exclude bibliography On
Exclude matches < 5 words

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Yap Zi Xin
ID Number(s)	20ACB05249
Programme / Course	Bachelor of Information System (Honours) Business Information System
Title of Final Year Project	Agriculture Product Supply Chain Management Application

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 4 % **Similarity by source** Internet Sources: 2 % Publications: 0 % Student Papers: 3 %	
Number of individual sources listed of more than 3% similarity: 0	

Parameters of originality required and limits approved by UTAR are as Follows:

(i) Overall similarity index is 20% and below, and

(ii) Matching of individual sources listed must be less than 3% each, and

(iii) Matching texts in continuous block must not exceed 8 words

Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Tan Teik Boon

Date: 13/09/2024

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB05249
Student Name	Yap Zi Xin
Supervisor Name	Ts. Tan Teik Boon

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

_____ _____

(Signature of Student)
Date: 11st September 2024