

# INTRODUCTION

---

“Irrigreat” defines in its slogan, Farm = AI + Crop + Fertilizer + Pesticide, taking care of soil’s health. Irrigreat honors the Indian farmer’s love, hard work and character. Farmers help to feed a nation whose population is nearly 1.4 billion, however the productivity of farms is threatened by various natural factors that ruin the crops and farmer’s livelihood. Irrigreat is a small initiative enhancing agriculture, making smart decisions to consider the demographics of the field, the factors affecting the crop, as well as how to keep the farm healthy for a super awesome yield. This will be implemented in the form of a website providing features of Crop Recommendation, Fertilizer Recommendation and Pesticide Recommendation based on site specific parameters.

## 1.1 Project Overview

Irrigreat is a website made for farmers to help them with crop recommendations based on values of N, P, K, temperature, rainfall, relative humidity and pH. Generally, soil gets degraded and productivity is reduced if the right crop isn’t chosen, but Irrigreat makes it really easy by using the ML model to make the real time prediction. Second feature is Fertilizer Prediction. If the farmer opts not to change the crop as per land, he can go with the same crop but use fertilizer which will be recommended by Irrigreat based on N, P, K and crop values. Lastly, a very useful feature implemented is Pesticide Recommendation. Pests are a huge threat but they can be stopped. Farmers simply need to upload a picture which clearly shows the pest and Irrigreat will identify the pest with the help of DL Model which is CNN and recommend the corresponding pesticide along with required dosage to get rid of pests and protect the crop. If the farmer already knows about the pest, then he/she can select the pest and corresponding pesticide will be recommended. Generally it’s seen that tests for soil are done by Indian government and results come within a few days but farmers really don’t know much on what to do next, so Irrigreat is sort of their next step. A simple, intuitive website will really help farmers to easily know the whereabouts of crops, thus helping every possible bit which Irrigreat can. Hence, the three modules: Crop, Fertilizer, Pesticide really comes handy and a boon for farmers.

## **1.2 Problem Definition**

Irrigreat aims to help Indian farmers and reduce their hardship. The problems faced by Indian farmers are defined as follows:

1. Productivity needs to be increased so that farmers can get more pay from the same piece of land without degrading soil.
2. Indian farmers aren't able to choose the right crop based on their soil requirements depending upon factors like N, P, K, temperature, humidity, rainfall, pH.
3. Farmers are generally unaware about the organic fertilizers or standard fertilizers to use as per soil requirements.
4. Due to inadequate and imbalanced fertilization, soil degradation is occurring, which leads to nutrient mining and the development of second-generation problems in nutrient management.
5. According to a study by the Associated Chambers of Commerce and Industry of India, annual crop losses due to pests amount to Rs. 50,000 crore.

## **1.3 Problem Objectives**

Corresponding to problems cited above, following are objectives that "Irrigreat" is trying to solve:

1. To implement precision agriculture (A modern farming technique that uses research data of soil characteristics, soil types, crop yield data collection and suggests the farmers the right crop based on their site specific parameters to reduce the wrong choice on a crop and increase in productivity).
2. To solve the problem by proposing a recommendation system through an ensemble model with majority voting technique crop for the site specific parameters with high accuracy and efficiency.
3. To recommend organic fertilizer on the basis of N, P, K values and crop.
4. To recognize the pest and recommend particular pesticide available in India as per ISO standards (ISO 9001, ISO 14001, ISO 17025).
5. To design a web application for achieving above objectives.

## **1.4 Need Analysis**

Agriculture is one of the biggest sources of earning for Indians. Though the Indian farmers really work hard in their fields, their productivity is threatened by natural factors. The fact cannot be changed that natural factors are uncontrollable so the best way is to take most out of the field despite the natural factors. One of the major problems is soil degradation which can be prevented by growing the crop that is the most suitable as per the land. But even if the farmer chooses to grow a particular type of crop, then the appropriate dosage of fertilizers would help. Another major problem is the pest which can be only treated through suitable pesticides. This will help farmers. Various tests are conducted by the government in India which check the contents of soil but farmers are unaware of what to do with the results of the soil test. Hence Irrigreat makes use of all the values of the tests and helps the farmers with crop recommendation, fertilizer recommendation and pesticide recommendation.

## **1.5 Project Outcomes and Deliverables**

The idea is to create a **web application** which has three major modules namely

### **1. Crop Recommendation**

- a. The user will input values of N, P, K, temperature (in °C), relative humidity (in %), rainfall (in mm) and pH.
- b. Output: Prescribed Crop

### **2. Fertilizer Recommendation**

- a. The user will input values of N, P, K and crop.
- b. Output: Corresponding organic fertilizer suggestions.

### **3. Pesticide Recommendation**

- a. The user will upload the picture which clearly shows the pest.
- b. Alternatively, the user can select the pest if the user already knows the pest.
- c. Output: Identified pest along with recommended pesticides available in India as per ISO standards (ISO 9001, ISO 14001, ISO 17025).

## 1.6 Assumptions and Constraints

Irrigreat follows some of the assumptions and constraints (shown in Table 1), if they are followed then Irrigreat would give best results.

Table 1: Assumptions and Constraints

S.No.	Assumptions and Constraints
1	Irrigreat supports 22 crops: apple, banana, blackgram, chickpea, coconut, coffee, cotton, grapes, jute, kidney beans, lentil, maize, mango, mothbeans, mungbean, muskmelon, orange, papaya, pigeon peas, pomegranate, rice, watermelon. Hence the user will get results which best suit the land but only from these 22 crops.
2	The system supports 10 pests: aphids, armyworm, beetle, bollworm, earthworm, grasshopper, mites, mosquito, sawfly and stem borer, which is a constraint.
3	The user can opt for uploading image of the pest or manual selection of the pest: In case of first choice, any other picture of pest uploaded (apart from 10 supported pests) will display the result which is close resemblance with pests supported and in the latter case, the user can only make a selection among 10 pests.
4	The user must have the picture which clearly shows the pest. (In case the user opts to upload picture)
5	The user must be connected to the internet so as to access the web application.
6	The user must enter realistic values for getting the best result. (Though the invalid values are not accepted)
7	The maximum file size in case of image upload is 2 GB and maximum dimensions as per Webp format are: 16383 x 16383

## 1.7 Standards

Irrigreat uses following standards:

1. Fertilizer - Organic Fertilizers
2. Pesticide Recommendation (Quality Management) - ISO 9001
3. Pesticide Recommendation (Environmental Management) - ISO 14001
4. Pesticide Recommendation (Testing and Calibration) - ISO 17025
5. Pest Image Upload: Webp
6. Web 2.0 : Internet technology to create online applications that behave dynamically.

## **1.8 Novelty of Work**

The works done in this field of agriculture are disintegrated and no single platform provides all such facilities of crop recommendation, fertilizer recommendation and pesticide recommendation altogether. “Irrigreat” is one stop solution to all the problems of the farmers and the feedback system really helps to improve and adapt according to the needs of the farmers. The work in the field of pests is only limited to pest detection but “Irrigreat” extends the idea of pest identification to pesticide recommendation as per the corresponding pest identified which is a practical use of pest detection. Along with that, the dataset is more customized w.r.t. Indian farms which are unique in themselves. The fertilizers recommended are natural fertilizers and the pesticides are as per ISO standards.

# REQUIREMENT ANALYSIS

---

The following section discusses expectations from “Irrigreat”. Software requirements consist of functional as well as non functional requirements which are discussed below. Also, this section talks about literature survey.

## 2.1 Literature Survey

Agriculture is a major source of livelihood in India and Indian farmers put in their heart and soul to feed people. Farmers generally deal with crops, fertilizers, pests and pesticides. Hence, Irrigreat aims to serve Indian farmers via all three modules of Crop Recommendation, Fertilizer Recommendation and Pesticide Recommendation. Crop recommendation has been an area which is explored a lot, but all of the systems vary on the basis of parameters that are fed into the ML model. Most of the ML models use Random Forest, some use Decision Tree, while others use Ensemble methods via Majority Voting Mechanism. Fertilizer Recommendation doesn't work much in the area of AI. Main reason can be disintegrated data, but Irrigreat collected all of the data from various sources and integrated it to have a well formed dataset. A dictionary based solution is implemented in Irrigreat. Thirdly, Pesticide Recommendation is not at all touched area, researchers have just restricted it to Pest Detection only, but Irrigreat extends the idea of identification of pest, along with a dictionary based solution for the corresponding pesticide, available in India. Irrigreat uses ISO 9001, ISO 14001 and ISO 17025 standards for pesticide recommendation. Most of the pesticides are taken from biostadt site which is a really popular site for farmers but the problem is that search isn't easy there and maximum pesticides recommended aren't available in India. Following is discussed about various research papers pertaining to services offered by Irrigreat.

(Rajak et al. 951-952) i.e. paper [\[1\]](#) talks about crop prediction using various learners like SVM used as a classifier, Naive Bayes, Multilayer perceptron (ANN) and lastly Random Forest. The parameters used for crop prediction are: pH, depth, water holding capacity, drainage, erosion.

The rule below demonstrates an example of the proposed recommendation system.

IF ph is mild alkaline

AND depth is above 90

AND water holding capacity is LOW

AND drainage is moderate

AND erosion is LOW

THEN PADDY

**(Dighe et al. 476-480) i.e. paper [2]** reviewed CHAID, KNN, K-means, Decision Tree, Neural Network, Naïve Bayes, C4.5, LAD, IBK and SVM algorithms and generated rules for recommendation system. Considering various factors like pH level of soil, month of cultivation, weather in the region, temperature, type of soil, etc. factors were considered to select maximum likely crops for plantation.

**(Mokarrama and Arefin) i.e. paper [3]** discussed Location Detection, Data analysis and storage, Similar location detection and Recommendation generation module. Physiographic database, Thermal zone database, Crop growing period database, crop production rate database and seasonal crop database were used to get the final crop.

**(Gadge and Sandhya) i.e. paper [4]** talks about Attribute selection, Multiple Linear Regression, Decision Tree using ID3, SVM, Neural Networks, C4.5, K-means and KNN. The proposed system consists of firstly Selection of agricultural field then Selection of crop previously planted, it takes input from user, preprocesses it, then in backend there is attribute selection followed by classification algorithm on data and then crop is recommended.

**(Mishra et al.) i.e. paper [5]** uses J48, LAD Tree, LWL, IBK algorithm, firstly WEKA tool is used, LAD tree showed the lowest accuracy, though pruning the tree can minimize the errors, IBK gave good accuracy.

Table 2 performs a compared study of all research papers for Crop Recommendation.

Table 2: Comparative Study for Crop Recommendation

Sr. No.	Name of Researcher, Year of Publication	Paper Title	Parameters Used/ Database Used	Methodology Adopted/ Modules Used
1	Rajak et al. , 2017	Crop Recommendation System to Maximize Crop Yield using Machine Learning Technique.	pH, depth, water holding capacity, drainage, erosion.	SVM used as a classifier, Naive Bayes, Multilayer perceptron (ANN) and Random Forest.
2	Dighe et al, 2018	Crop Recommendation System for Precision Agriculture.	pH level of soil, month of cultivation, weather in the region, temperature, type of soil	CHAID, KNN, K-means, Decision Tree, Neural Network, Naïve Bayes, C4.5, LAD, IBK and SVM
3	Mokarrama and Arefin, 2017	RSF: A Recommendation System for Farmers.	Physiographic database, Thermal zone database, Crop growing period database, crop production rate database and seasonal crop database.	Location Detection, Data analysis and storage, Similar location detection and Recommendation generation.
4	Gadge and Sandhya, 2017	A study on various data mining techniques for crop yield prediction.	Agricultural field, Crop previously planted.	Attribute selection, Multiple Linear Regression, Decision Tree using ID3, SVM, Neural Networks, C4.5, K-means and KNN
5	Mishra et al. , 2018	Use of data mining in crop yield prediction	Not mentioned	J48, LAD Tree, LWL, IBK algorithm.

**(Wu et al.) i.e paper [6]** collects a large-scale dataset for insect pest identification called IP102, which contains over 75, 000 photographs of 102 insects. In comparison to previous datasets, the IP102 complies with a number of features of insect pest distribution in real-world settings . Meanwhile, they use the dataset to test certain cutting-edge recognition techniques. The findings show that existing handcrafted feature methods and deep feature methods are insufficient for pest identification.

**(Kasinathan et al.) i.e paper [7]** Using a machine learning and insect pest detection algorithm, various insect datasets were identified and detected, and the results were correlated. ANN, SVM, KNN, Naive Bayes, and the CNN model were used to test classification accuracy between different machine learning techniques. According to the findings, the CNN model has the best classification precision of 91.5 percent and 90 percent for 9 and 24 insect groups, respectively, from the Wang and Xie datasets.

**(Ding and Taylor) i.e paper [8]** describes a tool for automatically tracking pests using photographs from traps. A sliding window-based detection pipeline is proposed, in which a convolutional neural network is applied to image patches at various locations to decrease the success of possessing a particular pest type. To generate the final detections, image patches are filtered using non-maximum suppression and thresholding based on their positions and related confidences.

**(TÜRKOĞLU and HANBAY) i.e paper [9]** For the identification of plant diseases and pests, the different effects are compared of deep feature extraction and transfer learning. Deep features for tunings layers of these deep models were extracted. The results of the obtained deep features were then determined using SVM, ELM, and KNN classifiers. Deep models were then fine-tuned using pictures of plant disease and pests. In comparison to conventional approaches, deep learning models achieved better outcomes, according to the evaluation results. The findings of deep feature extraction surpassed those of transfer learning.

**(Selvaraj et al.) i.e paper [10]** Using a transfer learning approach, massive datasets of expert pre-screened banana disease and pest symptom/damage images were gathered and used to model

three distinct convolutional neural network (CNN) architectures for detection. The DCNN proved to be a reliable and simple-to-implement technique for detecting digital banana disease and pests. Deep transfer learning (DTL) uses a pre-trained disease recognition algorithm to create a network that can make reliable predictions, and it was discovered that ResNet50 and InceptionV2 based models worked better than MobileNetV1.

Table 3 performs a compared study of all research papers for Pest Detection.

Table 3: Comparative Study for Pest Detection

Sr. No.	Name of Researcher, Year of Publication	Paper Title	Parameters Used/ Database Used	Methodology Adopted/ Modules Used
6	Wu et al., 2019	A Large-Scale Benchmark Dataset for Insect Pest Recognition	Pest identification database IP102	handcrafted feature methods and deep feature methods
7	Kasinathan et al., 2020	Insect classification and detection in field crops using modern machine learning techniques	Different field crops	ANN, SVM, KNN, Naive Bayes, and the CNN model
8	Ding and Taylor, 2016	Automatic moth detection from trap images for pest management	commercial codling moth dataset	CNN
9	TÜRKOĞLU and HANBAY, 2018	Plant disease and pest detection using deep learning-based features	Plant diseases and pest database	SVM, ELM, and KNN classifiers
10	Selvaraj et al., 2019	AI-powered banana diseases and pest detection	Datasets of expert pre-screened banana disease and pest	Distinct convolutional neural network, Deep transfer learning algorithm, ResNet50 model, InceptionV2 model and MobileNetV1 model

## **2.2 Functional Requirements**

“Irrigreat” has three different modules namely: Crop Recommendation, Fertilizer Recommendation and Pesticide Recommendation. So this section will define functional requirements for all the modules separately.

- 1. Crop Recommendation:** The system will recommend the crop as per site specific parameters entered by the user.
- 2. Fertilizer Recommendation:** The system will recommend the organic fertilizers as per the values entered by the user.
- 3. Pesticide Recommendation**
  - 3.1 Uploading the image:** The user will upload the image which clearly shows the pest.
  - 3.2 Manual selection of pest:** The user can choose to select the pest (alternative to uploading an image).
  - 3.3 Pest Identification:** The website will identify the pest.
  - 3.4 Pesticide Recommendation:** Based on the pest identified, the corresponding pesticide (as per ISO 9001, ISO 14001, ISO 17025 standards) will be recommended.

## **2.3 Non-Functional Requirements**

To judge the operation of “Irrigreat”, rather than specific behavior, there are certain non functional requirements given below which really plays role in Irrigreat’s usability, success, and effectiveness.

### **2.3.1 Performance Requirements**

The website has a feature of crop recommendation. The performance measuring variable for this module will be accuracy scores since the crop is displayed based on the pickle file created from the ML model. The desired accuracy score is  $\geq 90\%$ . For the other feature which is Fertilizer Recommendation, the performance metric is based on the effectiveness of organic solutions. Thirdly, the module of Pesticide Recommendation is based on Pest Identification and corresponding Pesticide Recommendation. The pests are identified by DL model, so training and testing accuracy as well as loss are performance metrics. The desired training and testing accuracy must be  $\geq 90\%$  and the pesticides must be as per ISO standards (ISO 9001, ISO 14001, ISO 17025).

### **2.3.2 Safety and Security Requirements**

So as to ensure safety and security, one time sign up is allowed and during sign up only three basic entries are to be made: email id, username, password. Hence, no personal information is asked for, which implies that there won’t be any safety or security concerns. Also, the password will be stored in hashed format (SHA256). Post signup only, one can login into the system after authentication and avail any service. For login, the user requires username and password. Minimum password length is 8 characters.

### **2.3.3 User friendliness**

The interface is quite simple, easy to use and intuitive.

### **2.3.4 Compatibility**

The website is compatible with all popular browsers (Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, Opera) and can be opened on PC, laptops, mobiles but to get the best

experience use Google Chrome and laptop/PC (if using mobile, the user can use Moto G4, Samsung phones, iPhone 5, 6, 7, 8, Plus, X, iPad, iPad Pro, Surface Duo, Galaxy Fold).

### **2.3.5 Scalability**

The system can be scaled to 100K+ users, 10+ pests, 22+ crops and more pesticides, fertilizers.

## **2.4 Cost Analysis**

Irrigreat” doesn’t involve any hardware. All the datasets are custom made by gathering the data from verified resources, Hence no paid dataset is used. The user must have the internet access to avail the services of Irrigreat and account on the website. This makes it really economical for Indian farmers.

## **2.5 Risk Analysis**

The crop recommendation module of Irrigreat recommends the crop on the basis of site specific parameters, which means environment factors are taken into account but not the economical aspects. Hence economic profitability is at risk. The crop performance could also be affected by the weather. Low rainfall or drought may lead to low yields while heavy rain could damage the crops. Not only crop but also the care of the crop decides the yields. Unmeasured dosage of pesticide could ruin the yield. The uncertainty factors, how the farmers grow the crop, the process carried out while using natural fertilizers and the pesticide dosage could ruin or increase the productivity.

## METHODOLOGY ADOPTED

---

The following section discusses how “Irrigreat” can be implemented.

### 3.1 Proposed Solution

“Irrigreat” has three different modules. Methodology for all the modules will be discussed one by one.

#### 3.1.1 Crop Recommendation

This module can be implemented in four steps as discussed below and shown in Figure 1:

##### Step 1: Data Acquisition

Dataset can be acquired from kaggle. Click [here](#) to have a look at the dataset.

##### Step 2: Values Input

Users are expected to input the site specific parameters like: N, P, K (all of them in %), temperature (in °C), relative humidity (in %), rainfall (in mm) and pH.

##### Step 3: ML Model Training and creating .pkl file

Recommendation system is based on the ensemble model with majority voting technique. The constituent models are:

1. SVM
2. Random Forest
3. Naive Bayes
4. kNN

After the model is trained, a .pkl file is created.

##### Step 4: Crop Recommendation

.pkl file is loaded to recommend the crop based on input.

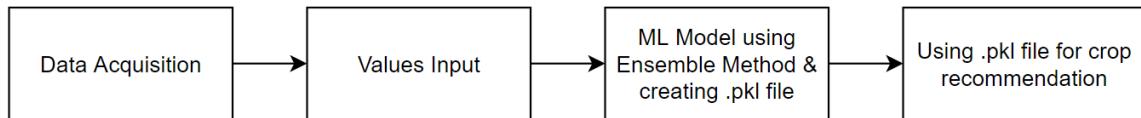


Figure 1: Methodology for Crop Recommendation

### **3.1.2 Fertilizer Recommendation**

This module can be implemented in four steps as discussed below and shown in Figure 2:

#### **Step 1: Data Acquisition**

Dataset will be created manually after collecting data from verified sources listed below:

1. The Fertilizer Association of India
2. Indian Institute of Water Management
3. Kaggle

The columns of the dataset are: N, P, K (all of them in %) and crop.

#### **Step 2: Values Input**

Users are expected to input the site specific parameters like: N, P, K (all of them in %), and crop (select from list - only 22 crops supported).

#### **Step 3: Difference between desired and actual**

Difference is calculated between desired value of N, P, K as per crop and the farm's actual value, based on it there are 3 outcomes possible for all three nutrients:

1. High
2. Low
3. Upto the mark

#### **Step 4: Fertilizer Recommendation**

Based on the outcomes from the above step, a dictionary based solution (organic fertilizers) will be displayed.

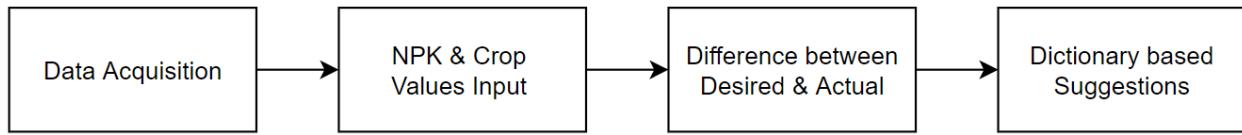


Figure 2: Methodology for Fertilizer Recommendation

### 3.1.3 Pesticide Recommendation

This module can be implemented in four steps as discussed below and shown in Figure 3:

#### Step 1: Data Acquisition

Dataset will be created by scraping images from Google via automatic script using Selenium and Chrome Driver. Along with that, pest labels will be provided as well.

#### Step 2: Data Cleaning and Data Augmentation

The data collected from Google needs to be cleaned manually to get rid of non-useful content e.g: In case of scraping images of pest named “beetle” there are also few images of “car called beetle”. Later on, the dataset needs to be augmented so as to increase variability.

#### Step 3: DL Model Creation

This involves model configuration, training configuration and model evaluation. Later on, .h5 file will be created to store the model.

#### Step 4: Pest Identification and corresponding Pesticide Recommendation

.h5 model will be loaded to identify the pest, later on based on the result, corresponding pesticide will be recommended based on dictionary based solution.

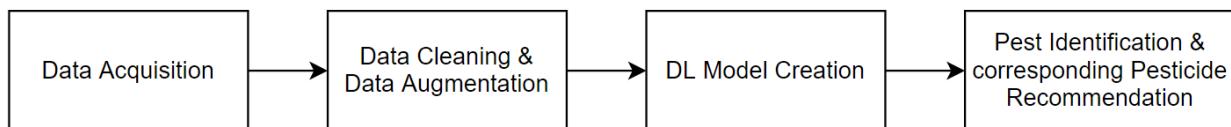


Figure 3: Methodology for Pesticide Recommendation

### 3.2 Work Breakdown Structure

“Irrigreat” has three different modules as shown in Figure 4. All the work breakdown structures are made based on project objectives and deliverables.

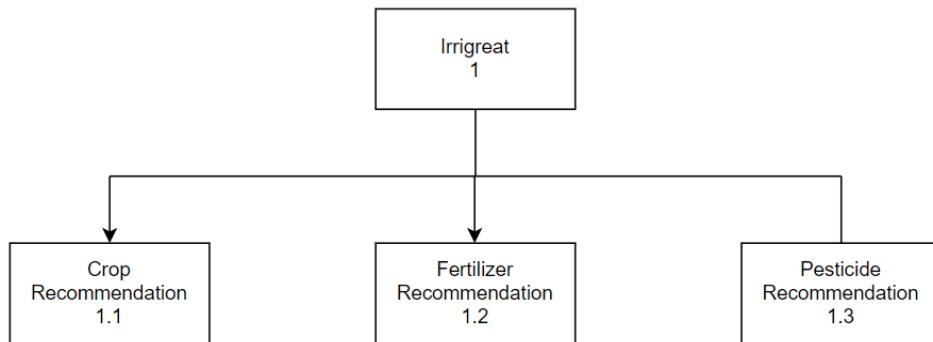


Figure 4: Irrigreat WBS

Figure 5 shows in detail, the submodule “Crop Recommendation” and further their submodules.

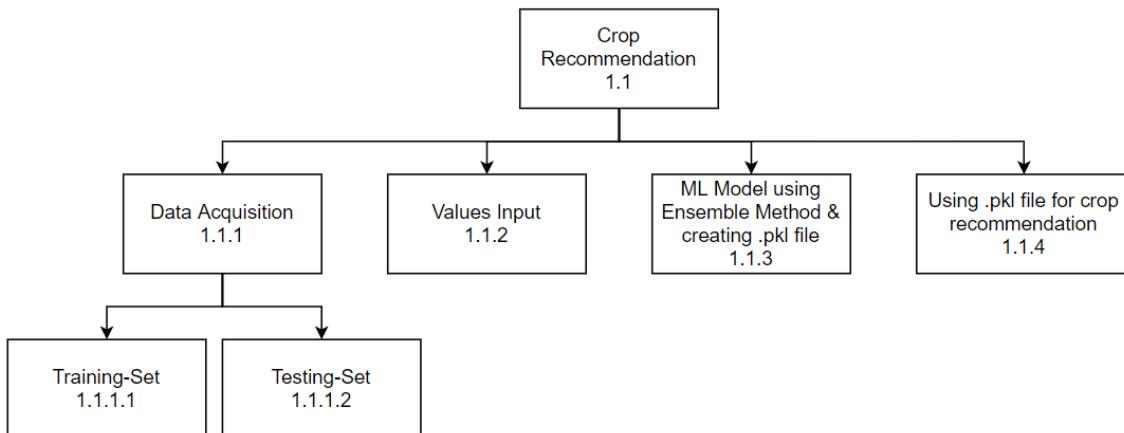


Figure 5: Crop Recommendation WBS

The second submodule is Fertilizer Recommendation, it's submodules are shown in Figure 6 while for the third and last submodule which is Pesticide Recommendation, its submodules are shown in Figure 7.

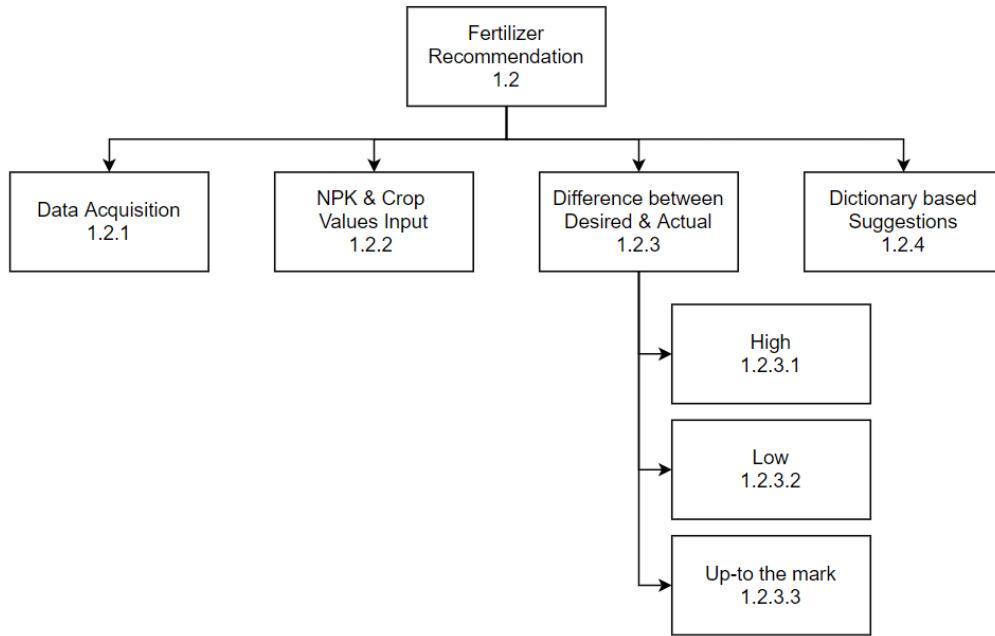


Figure 6: Fertilizer Recommendation WBS

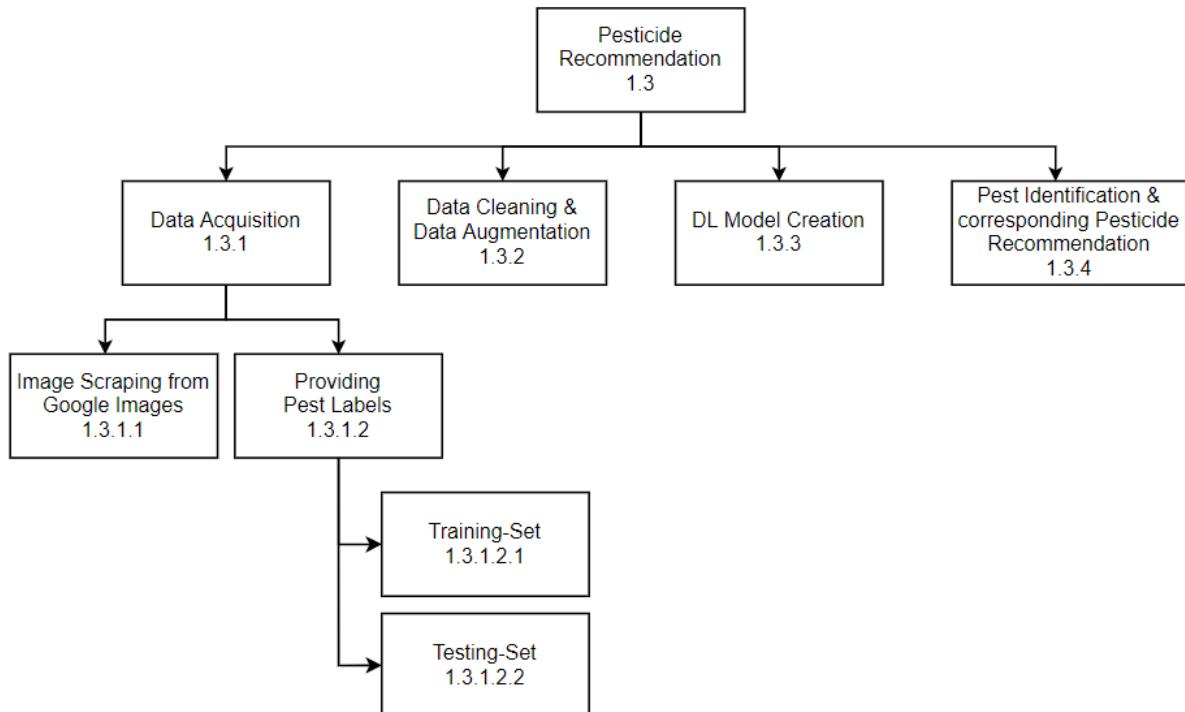


Figure 7: Pesticide Recommendation WBS

### 3.3 Gantt Chart

Following Gantt Chart (Figure 8) shows tentative project timeline, start date and end date of project and along with task dependencies. The tasks and subtasks can be identified through Gantt Chart Table only. To just see the table and not the timeline, refer to Table 4. Total Slack and Free Slack, Resource Assignment can also be added for more details. This Gantt Chart is as per 10th May, 2021.

Table 4: Gantt Chart Table

		Name	Duration	Work	Free Slack	Total Slack	Start	Finish	Actual Finish	Predecessors
1		Irrigreat	124.333 days?	592.5 hours	0 days	0 days	19/3/21 8:00 AM	28/5/21 2:00 PM		
2	✓	Research on project	7 days?	57 hours	57 days	57 days	19/3/21 8:00 AM	23/3/21 2:00 PM	23/3/21 2:00 PM	
3	✓	Studying Research papers	7 days?	21 hours	-7 days	-7 days	19/3/21 8:00 AM	23/3/21 2:00 PM	23/3/21 2:00 PM	
4	✓	Requirement Analysis	5 days?	15 hours	2 days	59 days	19/3/21 8:00 AM	22/3/21 4:00 PM	22/3/21 4:00 PM	
5	✓	Study on crops, fertilizers, pests and pesticides	7 days?	21 hours	0 days	57 days	19/3/21 8:00 AM	23/3/21 2:00 PM	23/3/21 2:00 PM	
6	✓	Research Summary	0 days?	0 hours	57 days	57 days	23/3/21 2:00 PM	23/3/21 2:00 PM	23/3/21 2:00 PM	3,4,5
7		Project Documentation	24 days?	72 hours	36.333 days	36.333 days	27/4/21 8:00 AM	10/5/21 8:00 AM		
8		Analysis Modelling, Design Modelling etc	24 days?	72 hours	0 days	36.333 days	27/4/21 8:00 AM	7/5/21 5:00 PM		
9		Project Report	0 days?	0 hours	36.333 days	36.333 days	10/5/21 8:00 AM	10/5/21 8:00 AM		8
10	✓	Crop Recommendation	9 days?	27.5 hours	14 days	14 days	27/4/21 8:00 AM	30/4/21 11:00 AM	30/4/21 11:00 AM 2	
11	✓	Data Acquisition	4 days?	12 hours	0 days	14 days	27/4/21 8:00 AM	28/4/21 1:00 PM	28/4/21 1:00 PM	
12	✓	Values Input	0.167 days?	0.5 hours	6.833 days	20.833 days	27/4/21 8:00 AM	27/4/21 8:30 AM	27/4/21 8:30 AM	
13	✓	ML Model using Ensemble method & creating .pkl file	3 days?	9 hours	0 days	14 days	28/4/21 12:00 PM	29/4/21 2:00 PM	29/4/21 2:00 PM	11
14	✓	Using .pkl file for crop recommendation	2 days?	6 hours	0 days	14 days	29/4/21 2:00 PM	30/4/21 11:00 AM	30/4/21 11:00 AM	12,13
15	✓	Crop Recommendation Module	0 days?	0 hours	14 days	14 days	30/4/21 11:00 AM	30/4/21 11:00 AM	30/4/21 11:00 AM	14
16	✓	Fertilizer Recommendation	10.167 days?	31 hours	12.833 days	12.833 days	27/4/21 8:00 AM	30/4/21 3:30 PM	30/4/21 3:30 PM	2
17	✓	Data Acquisition	3 days?	9 hours	0 days	12.833 days	27/4/21 8:00 AM	28/4/21 9:00 AM	28/4/21 9:00 AM	
18	✓	Values Input	0.167 days?	0.5 hours	2.833 days	15.667 days	27/4/21 8:00 AM	27/4/21 8:30 AM	27/4/21 8:30 AM	
19	✓	Difference between desired and actual	0.167 days?	0.5 hours	0 days	12.833 days	28/4/21 9:00 AM	28/4/21 9:30 AM	28/4/21 9:30 AM	17,18
20	✓	Dictionary based solutions	7 days?	21 hours	0 days	12.833 days	28/4/21 9:30 AM	30/4/21 3:30 PM	30/4/21 3:30 PM	19
21	✓	Fertilizer Recommendation Module	0 days?	0 hours	12.833 days	12.833 days	30/4/21 3:30 PM	30/4/21 3:30 PM	30/4/21 3:30 PM	20
22	✓	Pesticide Recommendation	23 days?	69 hours	0 days	0 days	27/4/21 8:00 AM	7/5/21 2:00 PM	7/5/21 2:00 PM	2
23	✓	Data Acquisition	4 days?	12 hours	0 days	0 days	27/4/21 8:00 AM	28/4/21 1:00 PM	28/4/21 1:00 PM	
24	✓	Data Cleaning and Data Augmentation	3 days?	9 hours	0 days	0 days	28/4/21 12:00 PM	29/4/21 2:00 PM	29/4/21 2:00 PM	23
25	✓	DL Model Training	8 days?	24 hours	0 days	0 days	29/4/21 2:00 PM	4/5/21 2:00 PM	4/5/21 2:00 PM	24
26	✓	Pest Detection & corresponding Pesticide recommendation	8 days?	24 hours	0 days	0 days	4/5/21 2:00 PM	7/5/21 2:00 PM	7/5/21 2:00 PM	25
27	✓	Pesticide Recommendation Module	0 days?	0 hours	0 days	0 days	7/5/21 2:00 PM	7/5/21 2:00 PM	7/5/21 2:00 PM	26
28		End to End Deployment	37.333 days?	336 hours	0 days	0 days	7/5/21 2:00 PM	28/5/21 2:00 PM		10:16:22
29		Flask Coding	37.333 days?	112 hours	0 days	0 days	7/5/21 2:00 PM	28/5/21 2:00 PM		
30		Front end design	37.333 days?	112 hours	0 days	0 days	7/5/21 2:00 PM	28/5/21 2:00 PM		
31		Integrating all three modules	37.333 days?	112 hours	0 days	0 days	7/5/21 2:00 PM	28/5/21 2:00 PM		
32		Deployment	37.333 days?	0 hours	0 days	0 days	7/5/21 2:00 PM	28/5/21 2:00 PM		
33		Web Application	0 days?	0 hours	0 days	0 days	28/5/21 2:00 PM	28/5/21 2:00 PM		29,30,31,32

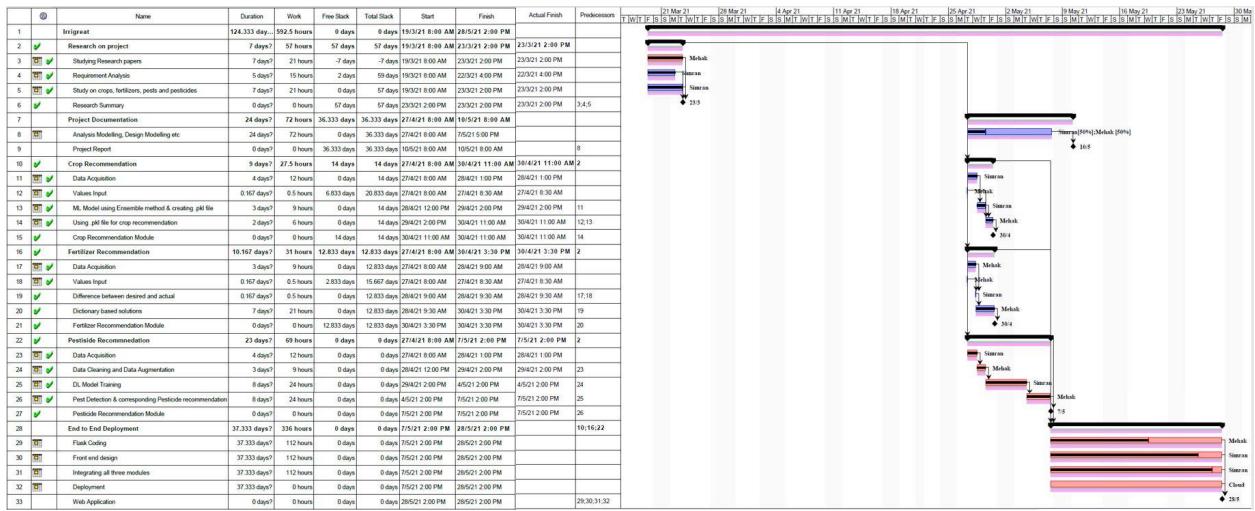


Figure 8: Gantt Chart for Irrigreat

## Identification of Tasks and Subtasks

Following are listed the tasks and subtasks for the system "Irrigreat". The Figure for the same (major tasks only) is shown in Figure 9.

### 1. Research on project

- a. Studying Research Papers
- b. Requirement Analysis
- c. Study on crops, fertilizers, pests, pesticides

### 2. Project Documentation

- a. Analysis Modelling, Design Modelling etc.

### 3. Crop Recommendation

- a. Data Acquisition
- b. Values Input
- c. ML model using ensemble method and creating. pkl file
- d. Using .pkl file for crop recommendation

### 4. Fertilizer Recommendation

- a. Data Acquisition
- b. Values Input
- c. Difference between desired and actual
- d. Dictionary based solution

## 5. Pesticide Recommendation

- a. Data Acquisition
- b. Data Cleaning and Data Augmentation
- c. DL model training
- d. Pest Identification and corresponding Pesticide Recommendation

## 6. End to End Deployment

- a. Flask Coding
- b. Front End Design
- c. Integrating all three modules
- d. Deployment

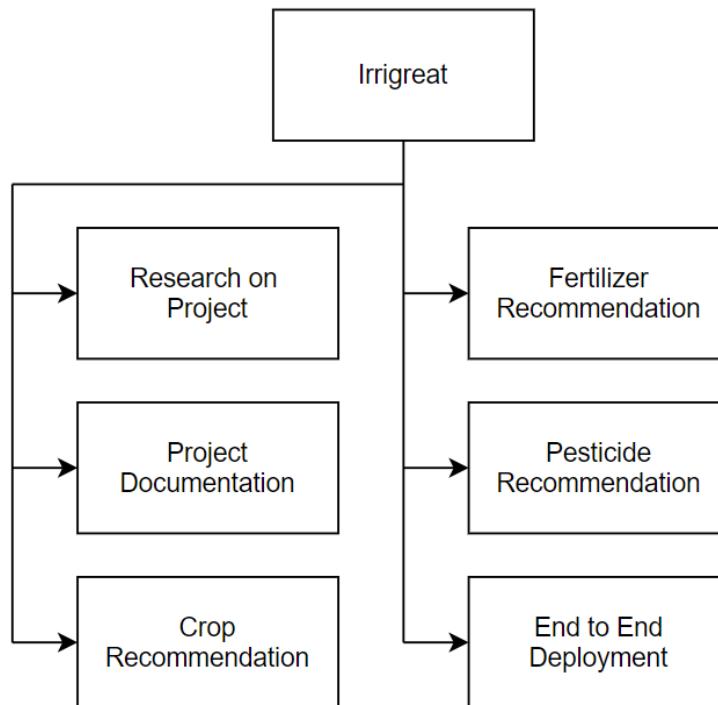


Figure 9: Major Tasks for Irrigreat

### **3.4 Tools and Technology Used**

Following is the list of tools and technology used while making the “Irrigreat”:

1. numpy
  - a. working with arrays
2. pandas
  - a. working with csv files
3. flask
  - a. app routing
  - b. web application
4. heroku
  - a. cloud deployment
5. pickle
  - a. saving ML model
6. pymongo
  - a. databases for irrigreat users
  - b. databases for irrigreat feedback
7. neural networks (keras, tensorflow, CNN)
  - a. for classification and training
8. ssl and passlib
  - a. for hashing of password for storage
9. PyCharm
  - a. python offline coding
10. OS
  - a. for manipulating files
11. matplotlib.pyplot
  - a. plotting graphs for training and testing accuracy
  - b. plotting graphs for training and testing loss
12. h5
  - a. storing DL model
13. sklearn
  - a. classifiers

## **DESIGN SPECIFICATIONS**

---

The following section discusses the end product “Irrigreat” and particular specifications, how “Irrigreat” performs in usual and unusual scenarios. All types of user inputs are handled and following points provide insight on workflow of the system as well. It also defines the actors, pre conditions, post conditions and other tidbits related to Irrigreat. The design specifications make the production of the final product really faster and easier.

### **4.1 Analysis Diagrams**

Analysis diagrams capture the system behavior and tell how the system will behave in different scenarios. In this report, use case diagram, use case template, activity diagram, workflow diagram and various other diagrams are made so as to design the system before the coding starts.

#### 4.1.1 Workflow Diagram

Workflow for Irrigreat is displayed in Figure 10, 11, 12, 13 which provides an insight on all the three modules and the control flow which takes place as per the selected module. To see the overall diagram, click [here](#) or refer to Figure 14.

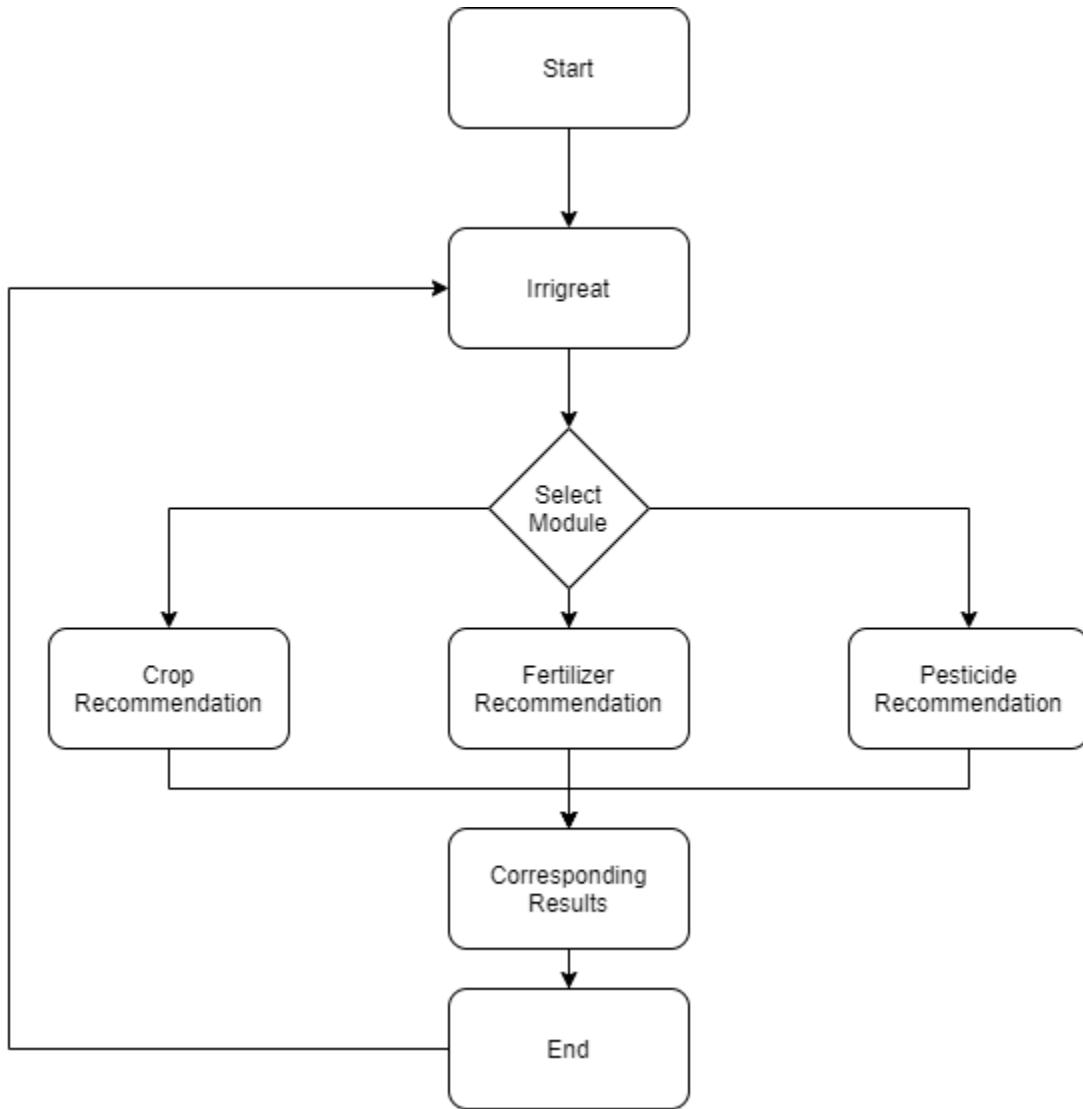


Figure 10: Irrigreat Workflow Diagram

Following displays the workflow diagram for Crop Recommendation.

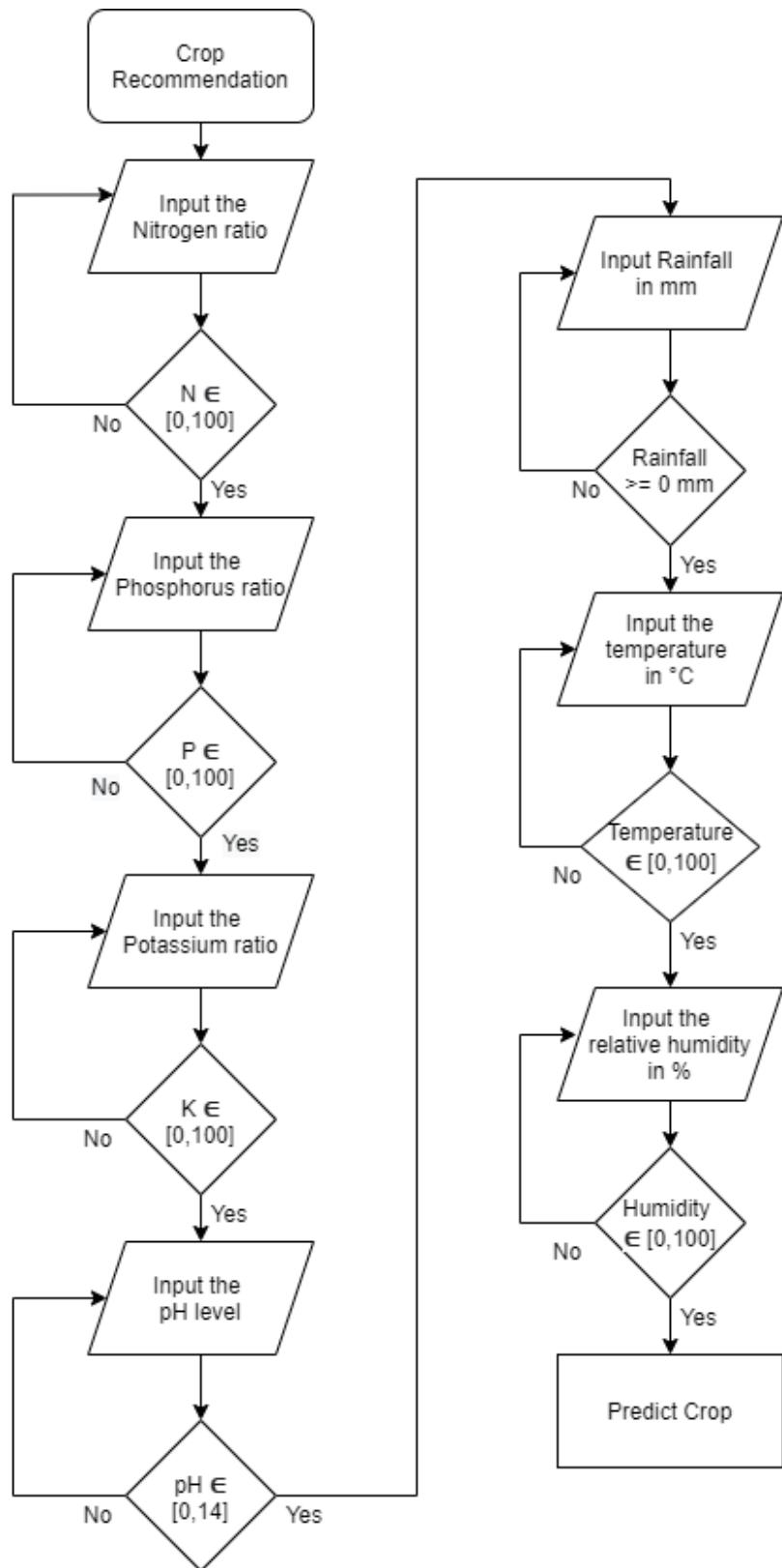


Figure 11: Crop Recommendation Workflow Diagram

Following displays the workflow diagram for Fertilizer Recommendation.

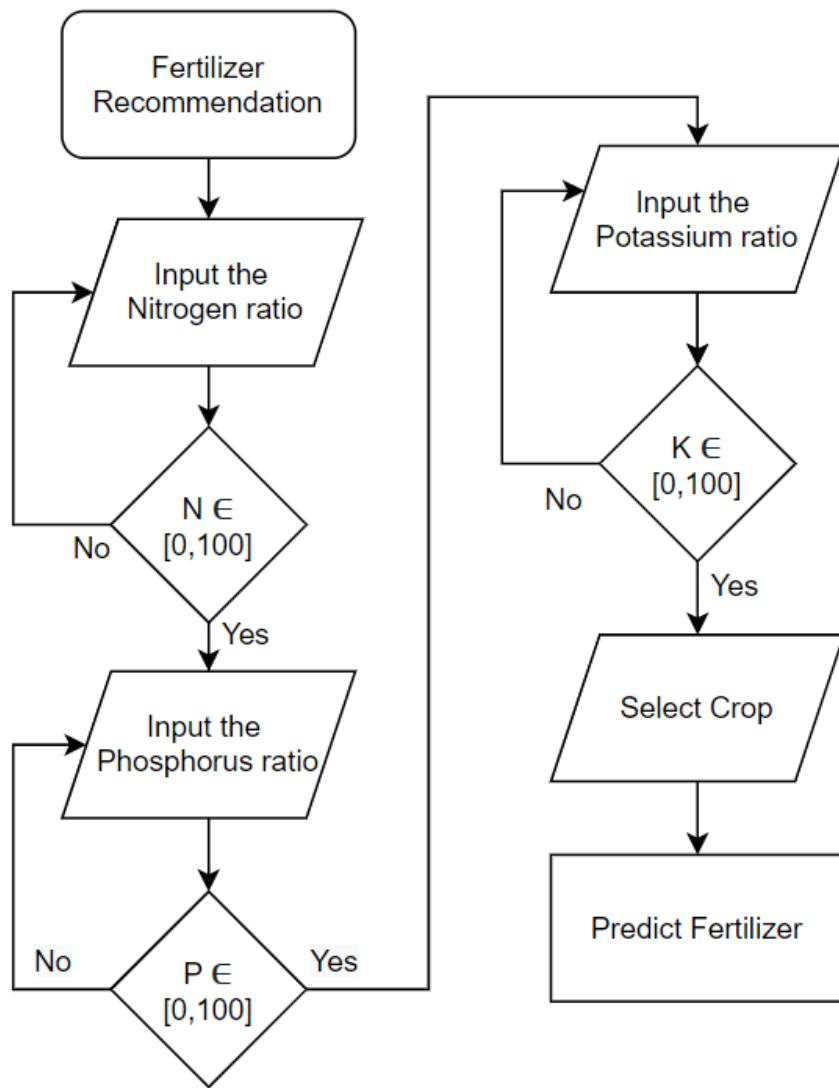


Figure 12: Fertilizer Recommendation Workflow Diagram

Following displays the workflow diagram for Pesticide Recommendation.

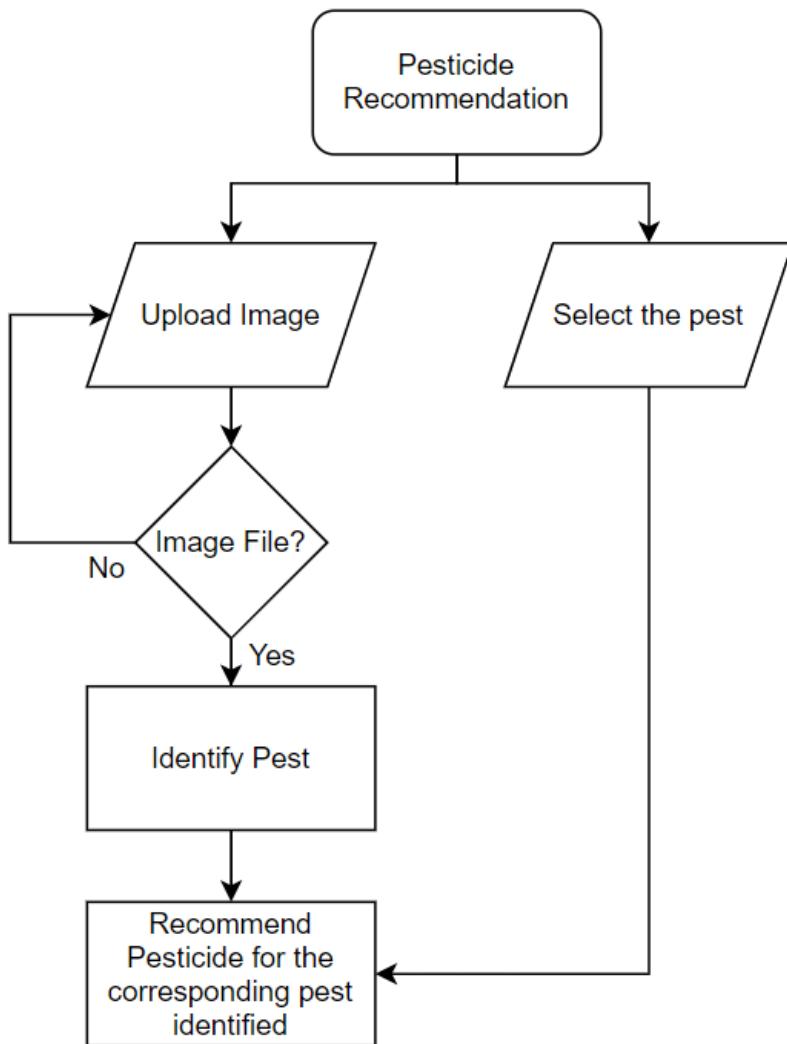


Figure 13: Pesticide Recommendation Workflow Diagram

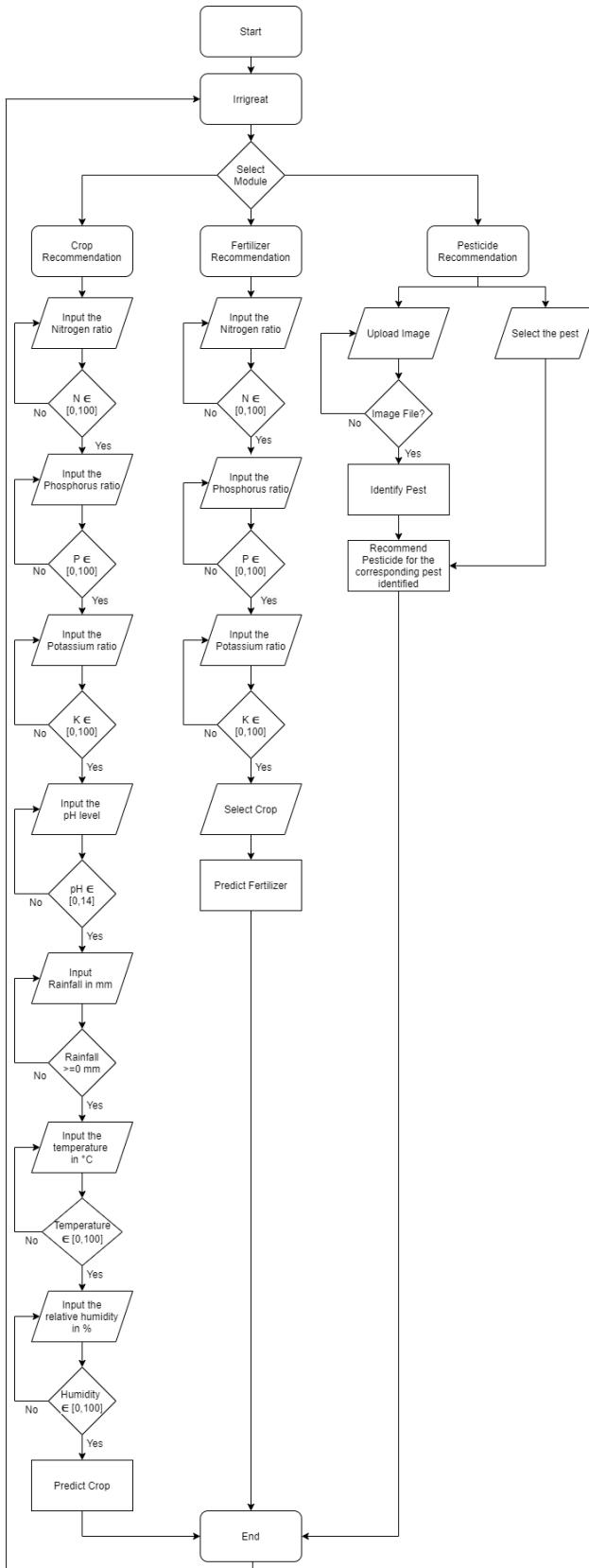


Figure 14: Overall Workflow Diagram

#### 4.1.2 Activity Diagram

To portray the workflow of “Irrigreat” from start till the end, activity diagram of overall system is shown in Figure 15 and for insight of all the three refer Figure 16, 17, 18. To see the overall diagram, click [here](#) or refer to Figure 19.

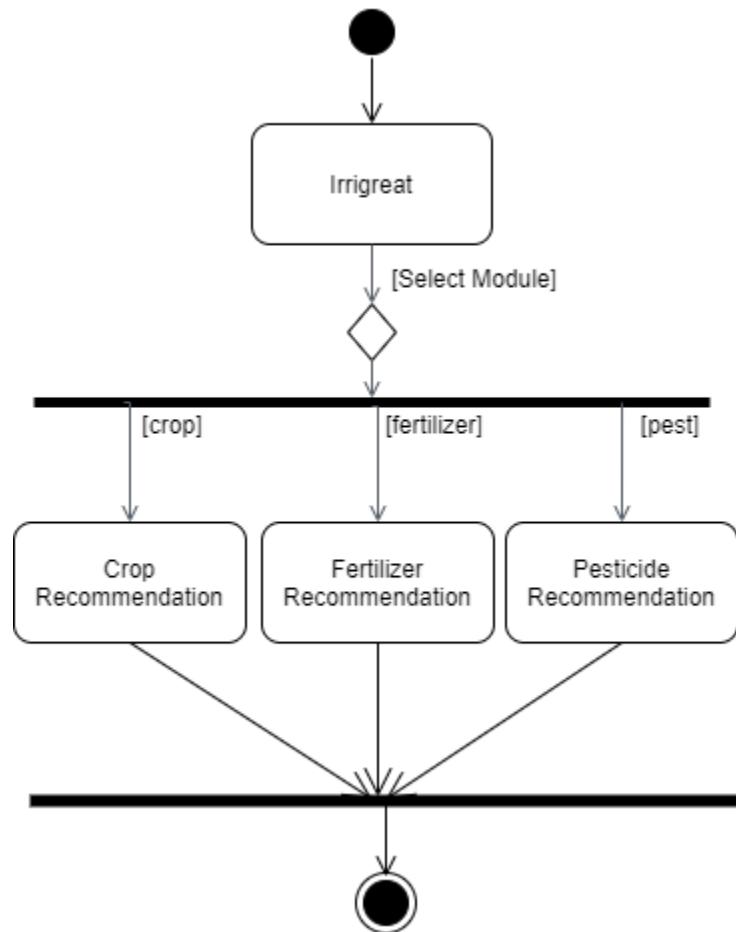


Figure 15: Irriggreat Activity Diagram

Following displays the activity diagram for Crop Recommendation.

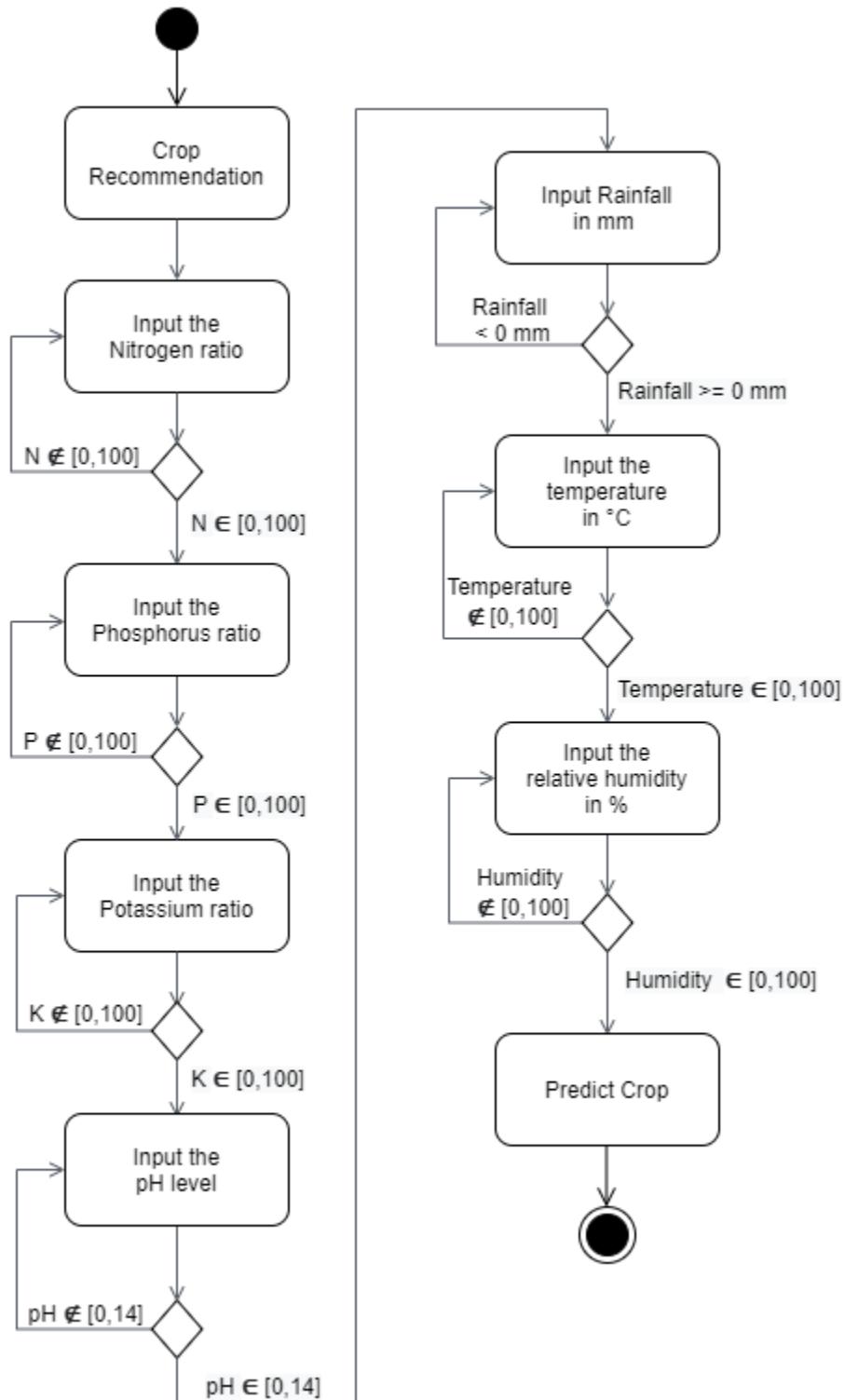


Figure 16: Crop Recommendation Activity Diagram

Following displays the activity diagram for Fertilizer Recommendation.

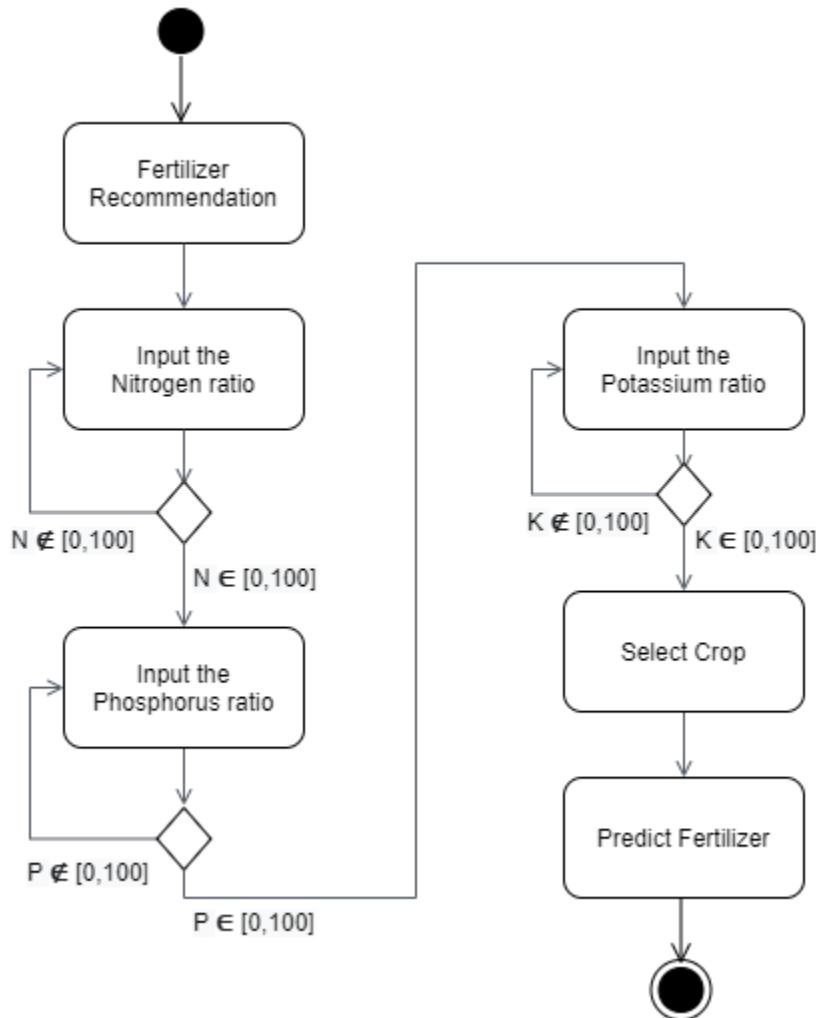


Figure 17: Fertilizer Recommendation Activity Diagram

Following displays the activity diagram for Pesticide Recommendation.

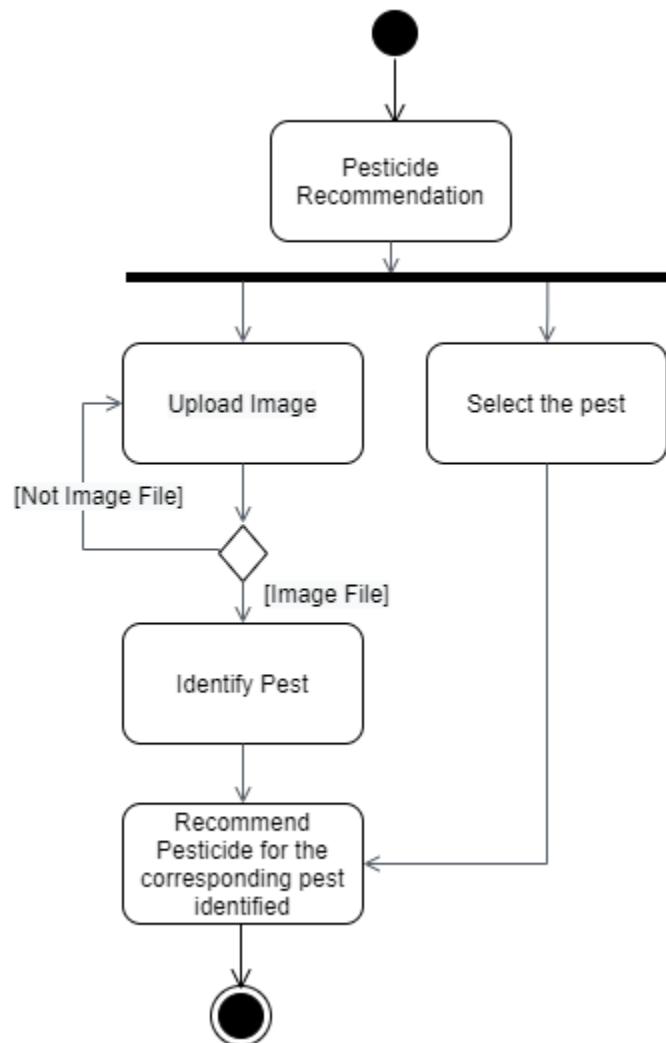


Figure 18: Pesticide Recommendation Activity Diagram

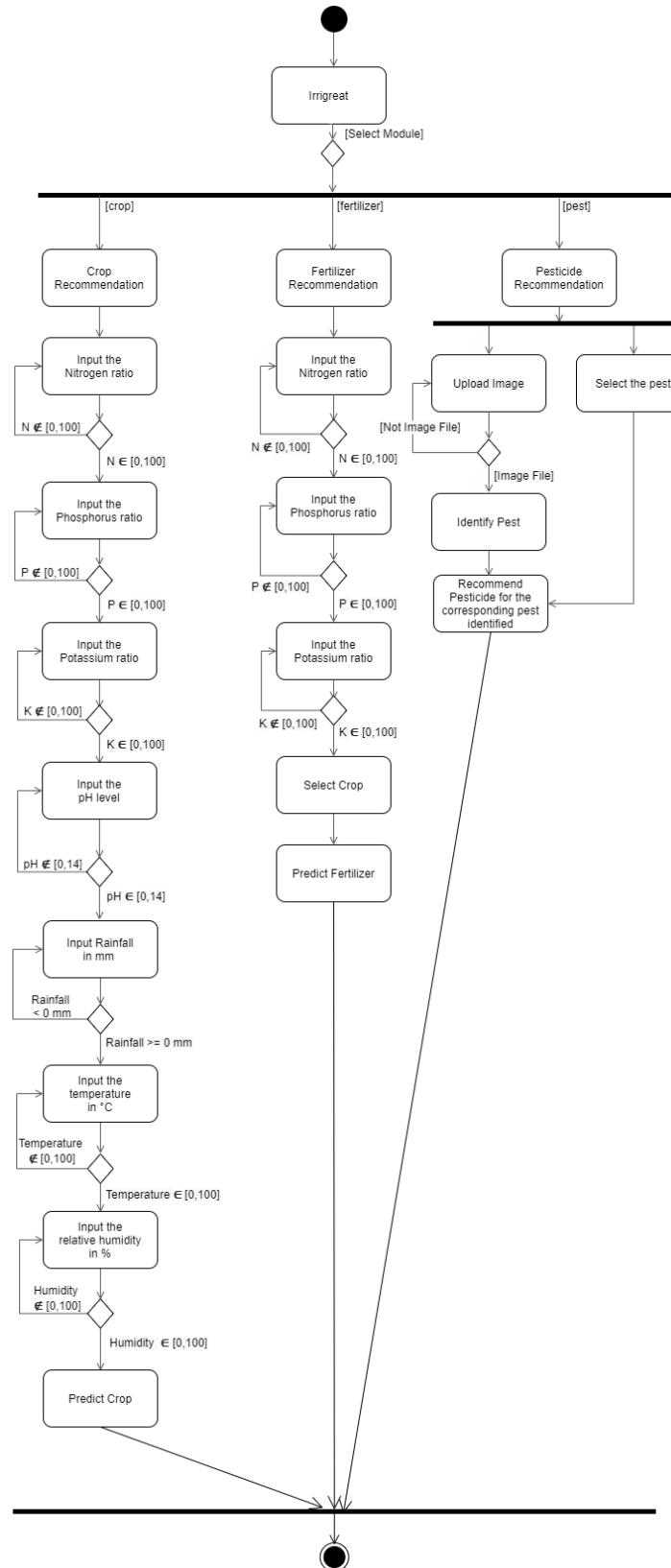


Figure 19: Overall Activity Diagram

#### 4.1.3 Use Case Diagram

Since the system is quite simple and the aim was to make it as user friendly as possible, keeping in view the farmer, so the actor here is only one which is the user, the farmer. Further the use cases, communication link, system boundary and use case relationships are shown in Figure 20 as given below.

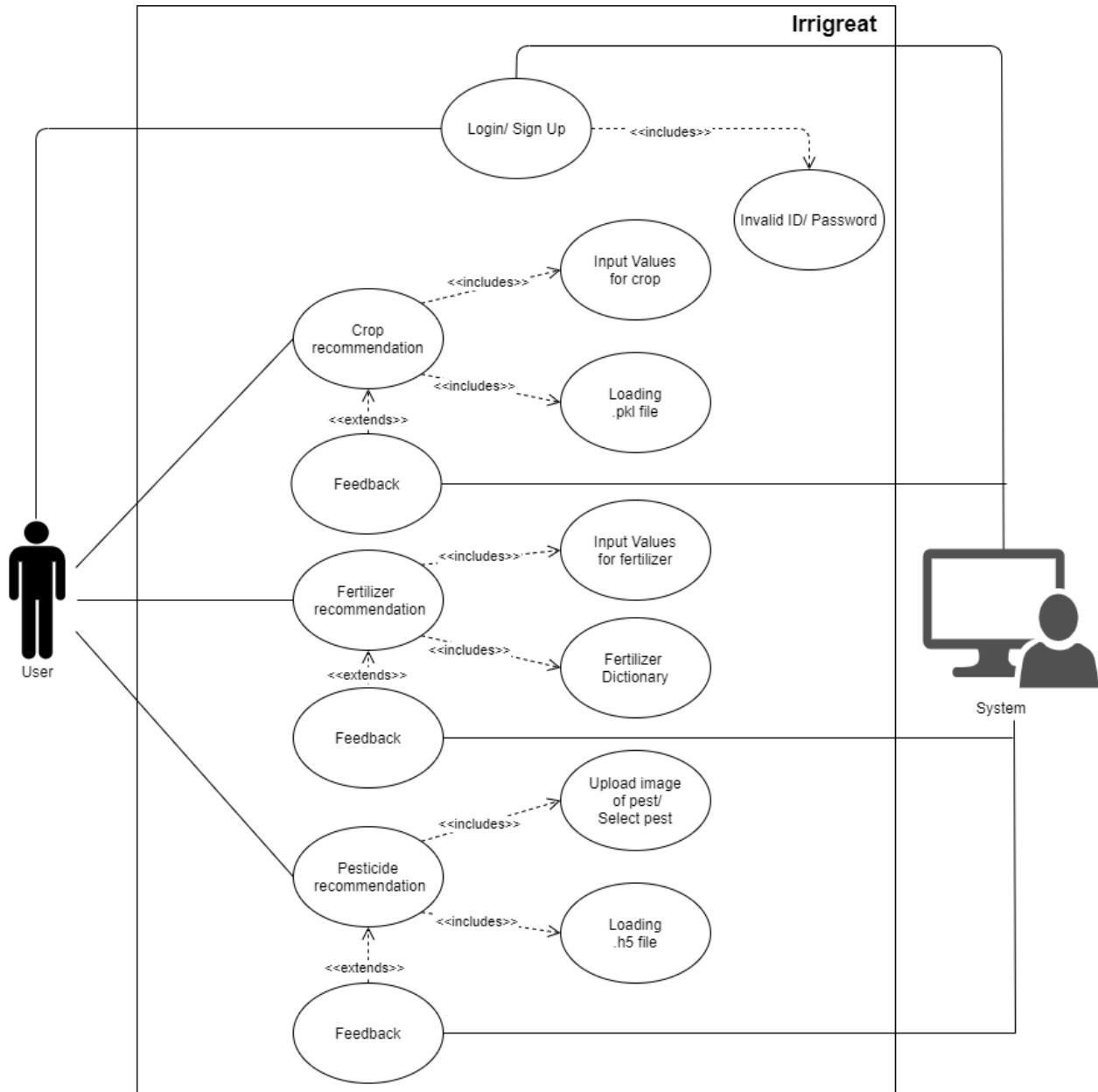


Figure 20: Irrigreat Use Case Diagram

#### 4.1.4 Use Case Template

Use Case Template will serve as the starting point of creating “Irrigreat”, the use case template for the same is shown in Table 5, 6 and 7. It ensures that the system being developed meets the needs of the end user or not.

Table 5: Use Case Template for Crop Recommendation

<b>Use Case Title</b>	Irrigreat 1.1
<b>Use Case Abbreviation</b>	Irrigreat 1.1
<b>Description</b>	Irrigreat Crop Recommendation module takes in input for N, P, K, temperature, humidity, rainfall and pH and outputs the crop that can be grown on the land.
<b>Goal</b>	The system must output the crop as per site specific parameters so that soil degradation does not occur and that crop must be grown on the land that suits as per land and soil conditions. Thus, farmers can get informed advice on which crop is best for their farm.
<b>Pre Conditions</b>	<ol style="list-style-type: none"> <li>1. The user must be connected to the internet so as to access the website.</li> <li>2. The user must have signed up once and logged in to the system so as to avail any service.</li> <li>3. The user must have realistic values which will be input to the system.</li> </ol>
<b>Basic Course</b>	<ol style="list-style-type: none"> <li>1. The user will login into the system.</li> <li>2. The user will select the crop recommendation service.</li> <li>3. The user will enter the realistic N, P, K values (0 to 100).</li> <li>4. pH level (0 to 14) is entered by the user.</li> <li>5. Rainfall in the area (in mm) is entered (<math>\geq 0</math>).</li> <li>6. Temperature is entered in <math>^{\circ}\text{C}</math> (0 to 100).</li> <li>7. Relative humidity is entered by the user in % (0 to 100).</li> <li>8. The user clicks on the “Recommend” button.</li> <li>9. Crop is recommended on the basis of site specific parameters.</li> </ol>
<b>Alternate Flow</b>	<p>At step 1 if login fails</p> <ol style="list-style-type: none"> <li>1. Error messages will be displayed to enter the valid credentials and the login page will be displayed again.</li> </ol> <p>At step 3 if values are inappropriate</p>

	<ol style="list-style-type: none"> <li>1. If the user enters the value which is &lt; 0, an error message will be displayed to enter the value (<math>\geq 0</math>).</li> <li>2. Or if the user enters the value which is &gt; 100, an error message will be displayed to enter the value (<math>\leq 100</math>).</li> </ol> <p>At step 4 if pH level is filled incorrectly</p> <ol style="list-style-type: none"> <li>1. If the user enters the value which is &lt; 0, an error message will be displayed to enter the value (<math>\geq 0</math>).</li> <li>2. Or if the user enters the value which is &gt; 14, an error message will be displayed to enter the value (<math>\leq 14</math>).</li> </ol> <p>At step 5 if rainfall will not filled correctly (in mm)</p> <ol style="list-style-type: none"> <li>1. Error messages will be displayed to enter <math>\geq 0</math>.</li> </ol> <p>At step 6 if the temperature(in °C) will not appropriate</p> <ol style="list-style-type: none"> <li>1. If the user enters the temperature which is &lt; 0, an error message will be displayed to enter the temperature (<math>\geq 0</math>).</li> <li>2. Or if the user enters the temperature which is &gt; 100, an error message will be displayed to enter the temperature (<math>\leq 100</math>).</li> </ol> <p>At step 7 if relative humidity in % will filled incorrectly</p> <ol style="list-style-type: none"> <li>1. If the user enters the humidity value which is &lt; 0, an error message will be displayed to enter the value (<math>\geq 0</math>).</li> <li>2. Or if the user enters the humidity value which is &gt; 100, an error message will be displayed to enter the value (<math>\leq 100</math>).</li> </ol>
<b>Post Conditions</b>	<ol style="list-style-type: none"> <li>1. The predicted crop will be shown on screen which is suitable to their farm.</li> <li>2. The user can provide optional feedback.</li> </ol>
<b>Actors</b>	<ol style="list-style-type: none"> <li>1. User</li> <li>2. System</li> </ol>
<b>Use Cases</b>	<ol style="list-style-type: none"> <li>1. Crop Recommendation</li> <li>2. Input Values for crop</li> <li>3. Loading .pkl file</li> <li>4. Feedback</li> </ol>
<b>Includes</b>	<ol style="list-style-type: none"> <li>1. Input Values for crop</li> <li>2. Loading .pkl file</li> </ol>
<b>Extends</b>	Feedback

<b>Modification History</b>	18th April, 2021
<b>Author</b>	Simran, Mehak Puri

Table 6: Use Case Template for Fertilizer Recommendation

<b>Use Case Title</b>	Irrigreat 1.2
<b>Use Case Abbreviation</b>	Irrigreat 1.2
<b>Description</b>	Irrigreat Fertilizer Recommendation module takes in input for N, P, K, and crop and outputs the organic fertilizer that can be used.
<b>Goal</b>	The system outputs organic fertilizer so as to get a good yield from the same piece of land and same crop. The fertilizer makes sure that the soil contents are appropriate as per desired crop. Thus, farmers can get informed advice on fertilizers.
<b>Pre Conditions</b>	<ol style="list-style-type: none"> <li>1. The user must be connected to the internet so as to access the website.</li> <li>2. The user must have signed up once and logged in to the system so as to avail any service.</li> <li>3. The user must have realistic values which will be input to the system.</li> </ol>
<b>Basic Course</b>	<ol style="list-style-type: none"> <li>1. The user will login into the system.</li> <li>2. The user will select the fertilizer recommendation service.</li> <li>3. The user will enter the realistic N, P, K values (0 to 100).</li> <li>4. The user will select the crop or otherwise by default the first crop will be selected.</li> <li>5. The user clicks on the “Recommend” button.</li> <li>6. Organic Fertilizer is recommended on the basis of site specific parameters.</li> </ol>
<b>Alternate Flow</b>	<p>At step 1 if login fails</p> <ol style="list-style-type: none"> <li>1. Error messages will be displayed to enter the valid credentials and the login page will be displayed again.</li> </ol> <p>At step 3 if values are inappropriate</p> <ol style="list-style-type: none"> <li>1. If the user enters the value which is &lt; 0, an error message will be displayed to enter the value (<math>\geq 0</math>).</li> </ol>

	<p>2. Or if the user enters the value which is &gt; 100, an error message will be displayed to enter the value (&lt;= 100).</p>
<b>Post Conditions</b>	<ol style="list-style-type: none"> <li>1. According to N, P, K values and crop selected to grow, differences will be shown between desired and farm values</li> <li>2. Suggestions will be given for organic fertilizer.</li> <li>3. The user can provide optional feedback.</li> </ol>
<b>Actors</b>	<ol style="list-style-type: none"> <li>1. User</li> <li>2. System</li> </ol>
<b>Use Cases</b>	<ol style="list-style-type: none"> <li>1. Fertilizer Recommendation</li> <li>2. Input Values for fertilizer</li> <li>3. Fertilizer Dictionary</li> <li>4. Feedback</li> </ol>
<b>Includes</b>	<ol style="list-style-type: none"> <li>1. Input Values for fertilizer</li> <li>2. Fertilizer Dictionary</li> </ol>
<b>Extends</b>	Feedback
<b>Modification History</b>	18th April, 2021
<b>Author</b>	Simran, Mehak Puri

Table 7: Use Case Template for Pesticide Recommendation

<b>Use Case Title</b>	Irrigreat 1.3
<b>Use Case Abbreviation</b>	Irrigreat 1.3
<b>Description</b>	Pesticide Recommendation module takes a picture of the pest or manually selects the pest by the user as input and outputs the identified pest along with pesticide (as per ISO 9001, ISO 14001, ISO 17025 standards) that can be used to get rid of that pest.
<b>Goal</b>	Pests are a threat to the farm and farmers' hard work goes futile because of pests. But if they are controlled in the initial stage then they cause no harm, thus this module identifies the pest and recommends pesticide which can be used to

	get rid of unwanted pests. Along with that, it also tells the required dosage of pesticide.
<b>Pre Conditions</b>	<ol style="list-style-type: none"> <li>1. The user must be connected to the internet so as to access the website.</li> <li>2. The user must have signed up once and logged in to the system so as to avail any service.</li> <li>3. The user must have a picture that clearly shows the pest.</li> </ol>
<b>Basic Course</b>	<ol style="list-style-type: none"> <li>1. The user will login into the system.</li> <li>2. The user will select the pesticide recommendation service.</li> <li>3. The user will upload the file to identify the pest.</li> <li>4. The user clicks on the “Recommend” button.</li> <li>5. Pesticide (as per ISO 9001, ISO 14001, ISO 17025 standards) along with dosage will be shown according to the identified pest.</li> </ol>
<b>Alternate Flow</b>	<p>At step 1 if login fails</p> <ol style="list-style-type: none"> <li>1. Error messages will be displayed to enter the valid credentials and the login page will be displayed again.</li> </ol> <p>At step 3 if the chosen file is not image file</p> <ol style="list-style-type: none"> <li>1. The error message will be displayed that file format is not appropriate and again ask to upload the image file.</li> </ol> <p>At step 3</p> <ol style="list-style-type: none"> <li>1. The user can manually select the pest.</li> </ol>
<b>Post Conditions</b>	<ol style="list-style-type: none"> <li>1. Identify the pest from the uploaded pest image.</li> <li>2. Pesticide (as per ISO 9001, ISO 14001, ISO 17025 standards) will be shown along with its dosage according to the identified pest.</li> <li>3. The user can provide optional feedback.</li> </ol>
<b>Actors</b>	<ol style="list-style-type: none"> <li>1. User</li> <li>2. System</li> </ol>
<b>Use Cases</b>	<ol style="list-style-type: none"> <li>1. Pesticide Recommendation</li> <li>2. Upload Image of pest/ Select pest</li> <li>3. Loading .h5 file</li> <li>4. Feedback</li> </ol>
<b>Includes</b>	<ol style="list-style-type: none"> <li>1. Upload Image of pest</li> <li>2. Loading .h5 file</li> </ol>

<b>Extends</b>	Feedback
<b>Modification History</b>	18th April, 2021
<b>Author</b>	Simran, Mehak Puri

#### 4.1.5 Sequence Diagram

Figure 21 depicts interaction between the objects in a sequential order. There are 8 objects named User, Irrigreat\_Users Database, Crop recommendation, Crop recommendation Database, Fertilizer recommendation, Fertilizer Dictionary, Pesticide Recommendation and Pest Identification. The diagram below shows the order in which the interaction between the objects takes place.

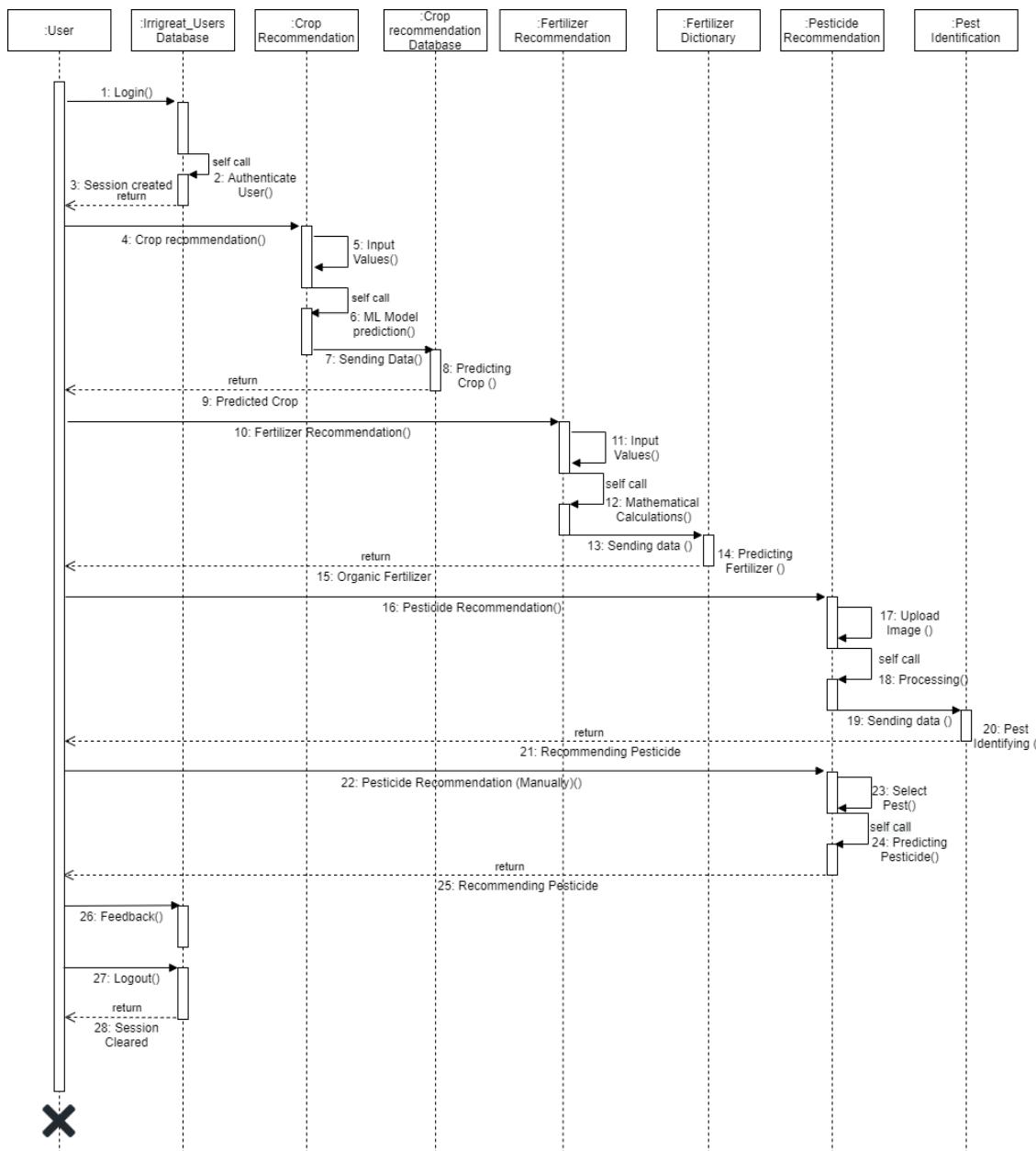


Figure 21: Sequence Diagram

#### 4.1.6 Class Diagram

Following Class Diagram (Figure 22) shows various classes, attributes and functions.

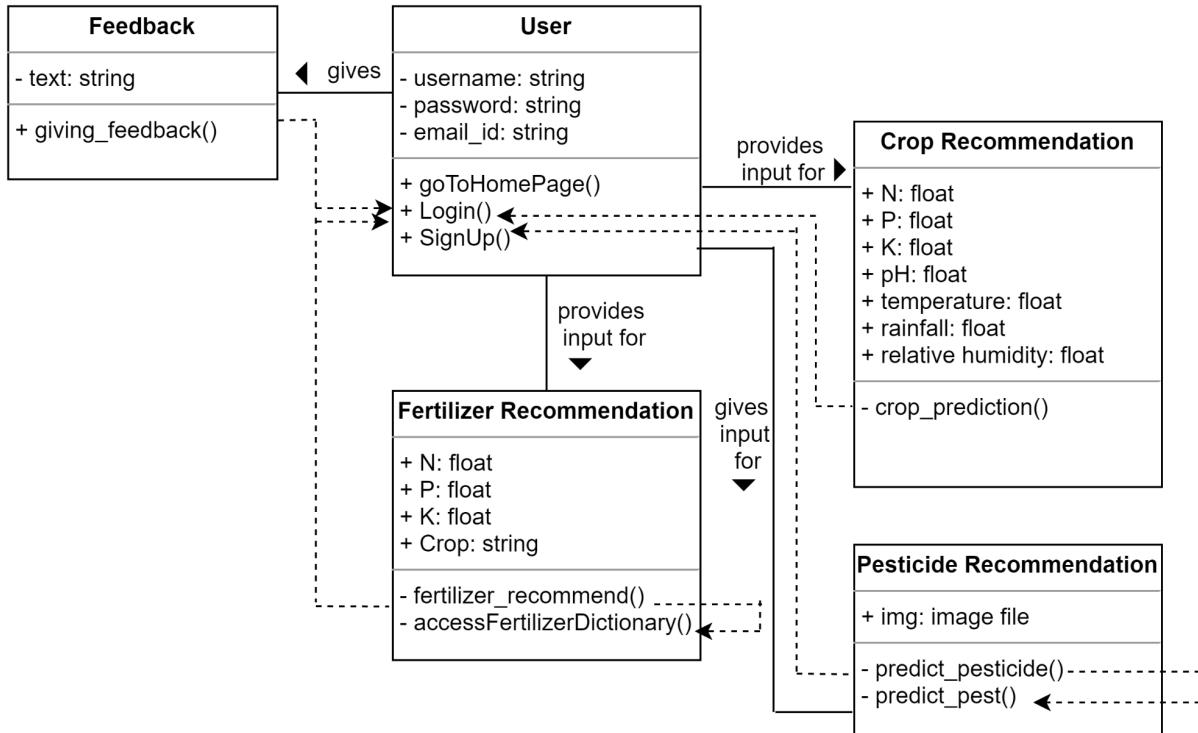


Figure 22: Class Diagram

## 4.2 System Architecture

Figure 23 shows the layered architecture of the product, constituting client, server, business layer, persistence layer, database.

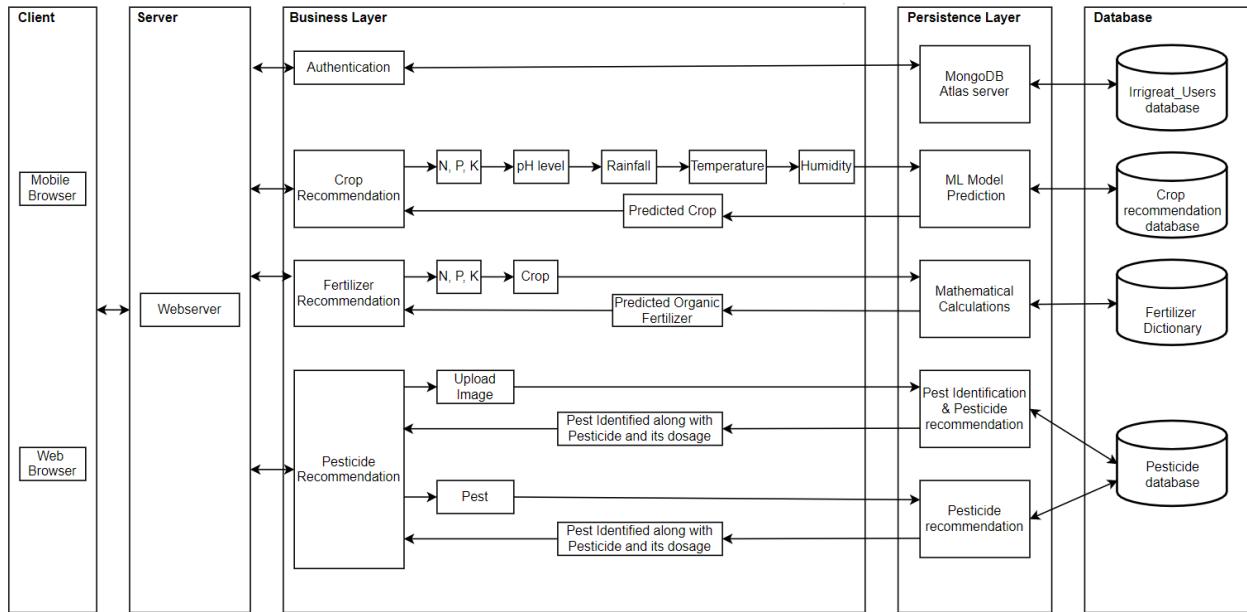


Figure 23: Layered Architecture of the product

## **4.3 Design Level Diagrams**

Design Level Diagrams include Component design, Data Design given in section 4.3.1 and 4.3.2 respectively.

### **4.3.1 Component Design**

Component design is the definition and design of components after the architectural design phase. It defines the data structures, algorithms, interface characteristics and communication mechanisms allocated to each component for the system development.

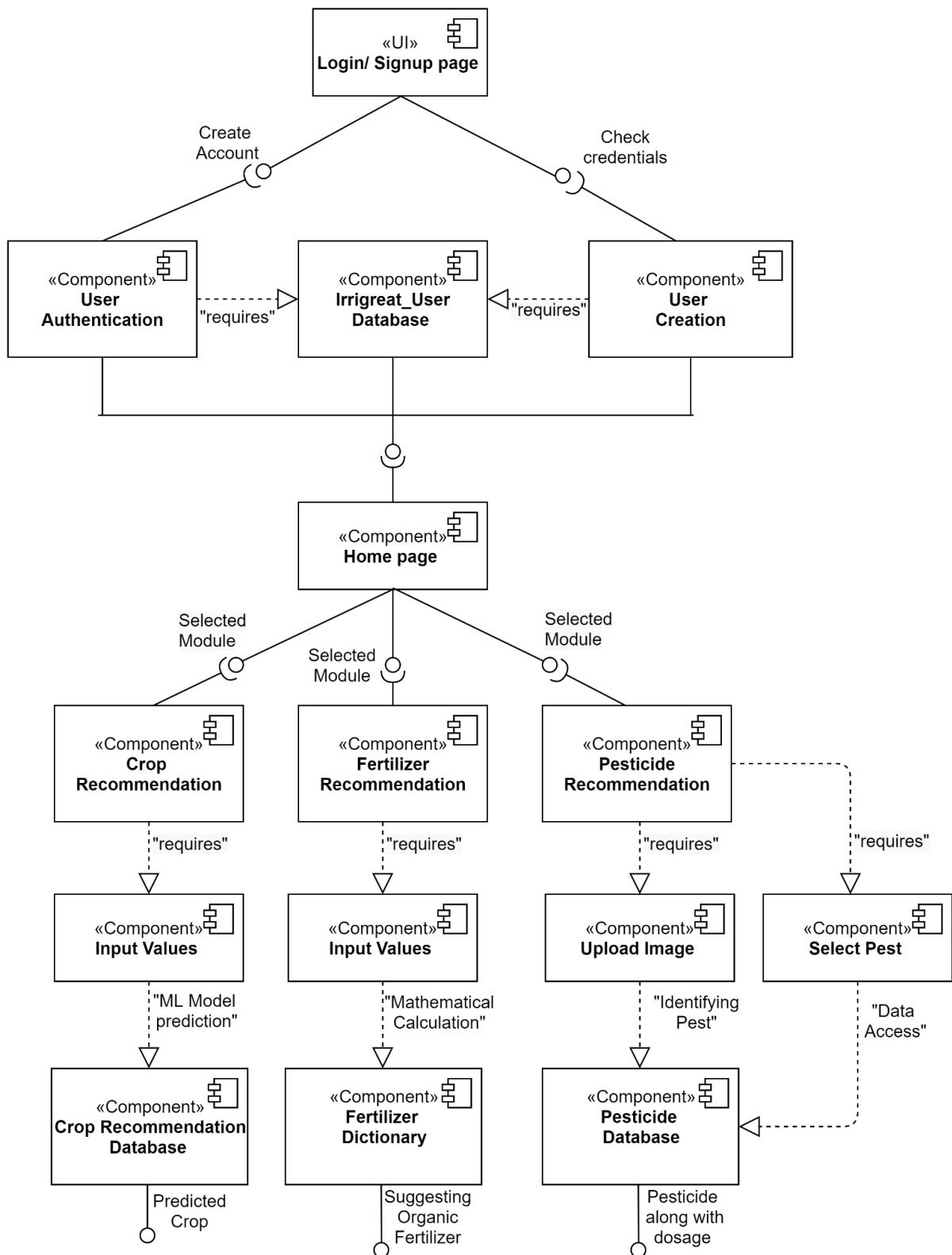


Figure 24: Component Design

### **4.3.2 Data Design (ER Diagram)**

Irrigreat uses databases so as to achieve its objectives, the ER diagram presents databases and relation among them. The ER Diagram for the system is shown in Figure 25. The ER diagram is made for the following modules of “Irrigreat”:

1. Login and SignUp
2. Crop Recommendation
3. Fertilizer Recommendation
4. Pesticide Recommendation (Uploading Image)
5. Feedback

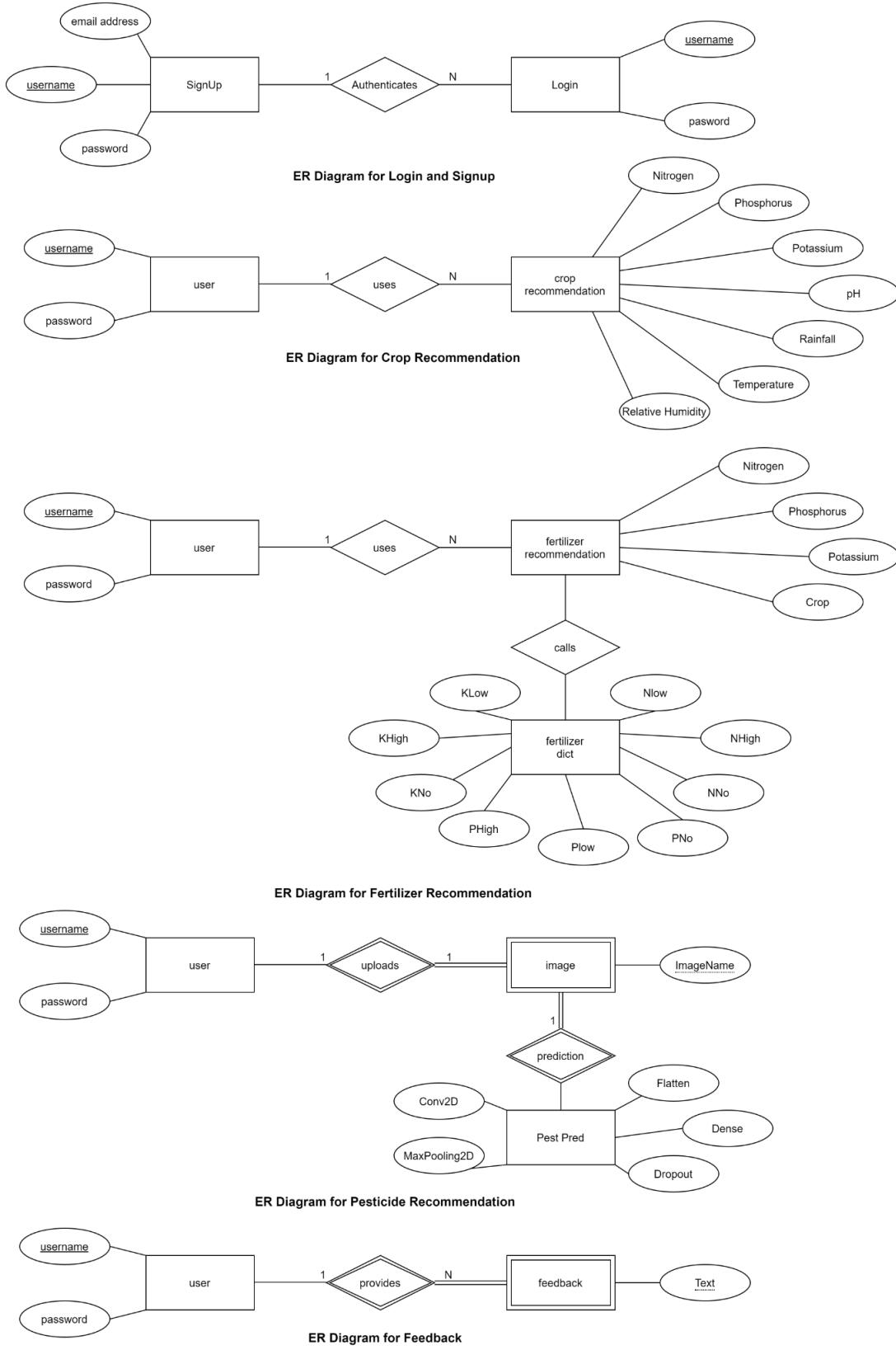


Figure 25: ER Diagram for Irrigreat

# **IMPLEMENTATION AND EXPERIMENTAL RESULTS**

---

This section deals with discussion of implementation and experimentation with regards to the project. It also mentions all the test plans including the features to be tested, the test cases and discusses the inference drawn from the results. This doesn't end here, this section will also discuss algorithms used, system screenshots and in the end validate project objectives.

## **5.1 Experimental Setup**

Irrigreat has majorly 3 modules namely: Crop Recommendation, Fertilizer Recommendation, Pesticide Recommendation. Following will discuss the experimental setup for all the three modules.

1. For all the services, the user needs to have an account created on the website, post that user can login any no. of times to avail any of the three services. During signup, the user needs to have an email address. Along with that, the user will give any username and password (minimum 8 characters and maximum 20 characters). For one email address, the user can have only one account and the username must be unique, hence not be taken. During signup, the user just needs to enter username and password.
2. Post successful authentication, the user can avail any of the 3 services.
3. For the first module, which is crop recommendation, the user needs to fill in the values for N, P, K (all in ratio), temperature (in °C), relative humidity (in %), rainfall (in mm) and pH. After that, the user will be recommended the most suitable crop as per the land.
4. For the second module which is Fertilizer Recommendation, the user needs to have values for N, P, K and select the crop, based on that natural fertilizers will be recommended as per deficiency or surplus of nutrients.
5. For the third module, the user can choose to select the pest manually if the user knows about the pest and directly pesticide would be recommended, otherwise the user can choose to upload the picture that clearly shows the pest, thereby pest will be identified in the backend and corresponding pesticide would be recommended.
6. Post all the activities, the user can give optional feedback. Later on, he/she can again go to "Home" and look for services or logout.

## **5.2 Experimental Data**

Irrigreat is an agriculture based project, hence data plays a very crucial role here. Along with that, data needs to be from authentic resources.

### **1. Data Source for Crop Recommendation**

- a. Dataset has labels: N, P, K (all in ratio), temperature (in °C), relative humidity (in %), rainfall (in mm) and pH.
- b. Data Source: Kaggle

### **2. Data Source for Fertilizer Recommendation**

- a. Required N, P, K for crops dataset
  - i. Dataset labels: N, P, K, crop
  - ii. Data Source: The Fertilizer Association of India (FAI), Indian Institute of Water Management, Kaggle
- b. Fertilizer Dictionary (Source: Google)

### **3. Data Source for Pesticide Recommendation**

- a. Type of dataset: Image Dataset
- b. No. of training images: 300 images per pest
- c. No. of testing images: 50 images per pest
- d. Data Source: Automatic script to scrape images of pest from Google through Selenium and Chrome Driver
- e. Pesticide Dataset: From biostadt website

### **4. Data Cleaning**

- a. It needs to be done on pest dataset (training images). For instance, in scraping images for beetles (a pest), images also had a beetle car.

### **5. Data Augmentation**

- a. Data Augmentation is to be performed in training images of the pest, so as to increase the variability.

### 5.3 Algorithms Used

Irrigreat is an agriculture based project, hence data plays a very crucial role here. Along with that, data needs to be from authentic resources. The algorithms used must result in good accuracy.

Firstly, given below is an algorithm for crop recommendation.

```
crop_recommendation_model_path = 'Crop_Recommendation.pkl'  
crop_recommendation_model = pickle.load(open(crop_recommendation_model_path, 'rb'))  
  
@ app.route('/crop_prediction', methods=['POST'])  
def crop_prediction():  
    if request.method == 'POST':  
        N = float(request.form['nitrogen'])  
        P = float(request.form['phosphorous'])  
        K = float(request.form['potassium'])  
        ph = float(request.form['ph'])  
        rainfall = float(request.form['rainfall'])  
        temperature = float(request.form['temperature'])  
        humidity = float(request.form['humidity'])  
        data = np.array([[N, P, K, temperature, humidity, ph, rainfall]])  
        my_prediction = crop_recommendation_model.predict(data)  
        final_prediction = my_prediction[0]  
        return render_template('crop-result.html', prediction=final_prediction, pred =  
'img/crop/'+final_prediction+'.jpg')
```

Now, as it can be seen that algorithm is using a .pkl file, so following is the algorithm that creates this .pkl file, basically the algorithm used is Ensemble Techniques through Majority Voting Mechanism. The learners are: SVM, kNN, Naive Bayes, Random Forest.

```

from sklearn.model_selection import train_test_split
import pandas as pd
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import RandomForestClassifier, VotingClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
from sklearn import model_selection

crop = pd.read_csv('Data/crop_recommendation.csv')
X = crop.iloc[:, :-1].values
Y = crop.iloc[:, -1].values

X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.15)

models = []
models.append(('SVC', SVC(gamma ='auto', probability = True)))
models.append(('svm1', SVC(probability=True, kernel='poly', degree=1)))
models.append(('svm2', SVC(probability=True, kernel='poly', degree=2)))
models.append(('svm3', SVC(probability=True, kernel='poly', degree=3)))
models.append(('svm4', SVC(probability=True, kernel='poly', degree=4)))
models.append(('svm5', SVC(probability=True, kernel='poly', degree=5)))
models.append(('rf',RandomForestClassifier(n_estimators = 21)))
models.append(('gnb',GaussianNB()))
models.append(('knn1', KNeighborsClassifier(n_neighbors=1)))
models.append(('knn3', KNeighborsClassifier(n_neighbors=3)))
models.append(('knn5', KNeighborsClassifier(n_neighbors=5)))
models.append(('knn7', KNeighborsClassifier(n_neighbors=7)))
models.append(('knn9', KNeighborsClassifier(n_neighbors=9)))

```

```

vot_soft = VotingClassifier(estimators=models, voting='soft')
vot_soft.fit(X_train, y_train)
y_pred = vot_soft.predict(X_test)

scores = model_selection.cross_val_score(vot_soft, X_test, y_test, cv=5, scoring='accuracy')
print("Accuracy: ", scores.mean())

score = accuracy_score(y_test, y_pred)
print("Voting Score %d" % score)

import pickle
pkl_filename = 'Crop_Recommendation.pkl'
Model_pkl = open(pkl_filename, 'wb')
pickle.dump(vot_soft, Model_pkl)
Model_pkl.close()

```

Second module is Fertilizer Recommendation, there isn't any specific algorithm for this module. It is a simple mathematical calculation and a fertilizer dictionary.

```

def fertilizer_recommend():
    crop_name = str(request.form['cropname'])
    N_filled = float(request.form['nitrogen'])
    P_filled = float(request.form['phosphorous'])
    K_filled = float(request.form['potassium'])

    df = pd.read_csv('Data/Crop_NPK.csv')

    N_desired = df[df['Crop'] == crop_name]['N'].iloc[0]
    P_desired = df[df['Crop'] == crop_name]['P'].iloc[0]
    K_desired = df[df['Crop'] == crop_name]['K'].iloc[0]

```

```
n = N_desired - N_filled  
p = P_desired - P_filled  
k = K_desired - K_filled
```

```
if n < 0:
```

```
    key1 = "NHigh"
```

```
elif n > 0:
```

```
    key1 = "Nlow"
```

```
else:
```

```
    key1 = "NNo"
```

```
if p < 0:
```

```
    key2 = "PHigh"
```

```
elif p > 0:
```

```
    key2 = "Plow"
```

```
else:
```

```
    key2 = "PNo"
```

```
if k < 0:
```

```
    key3 = "KHigh"
```

```
elif k > 0:
```

```
    key3 = "Klow"
```

```
else:
```

```
    key3 = "KNo"
```

```
abs_n = abs(n)
```

```
abs_p = abs(p)
```

```
abs_k = abs(k)
```

```

response1 = Markup(str(fertilizer_dict[key1]))
response2 = Markup(str(fertilizer_dict[key2]))
response3 = Markup(str(fertilizer_dict[key3]))
return render_template('Fertilizer-Result.html', recommendation1=response1,
                      recommendation2=response2, recommendation3=response3,
                      diff_n = abs_n, diff_p = abs_p, diff_k = abs_k)

```

The fertilizer dictionary used above is given below. The dictionary shown below is not a complete dictionary but just for Nitrogen to give an insight.

```

fertilizer_dict = {
    'NHigh': """The N value of soil is high and might give rise to weeds.  

    <br/><br/> Please consider the following suggestions:  

    <p align="justify">1. <i>Manure</i> – adding manure is one of the simplest ways to amend your soil with nitrogen. Be careful as there are various types of manures with varying degrees of nitrogen.</p>
    <p align="justify">2. <i>Coffee grinds</i> – use your morning addiction to feed your gardening habit! Coffee grinds are considered a green compost material which is rich in nitrogen. Once the grounds break down, your soil will be fed with delicious, delicious nitrogen. An added benefit to including coffee grounds to your soil is while it will compost, it will also help provide increased drainage to your soil.</p>
    <p align="justify">3. <i>Plant nitrogen fixing plants</i> – planting vegetables that are in Fabaceae family like peas, beans and soybeans have the ability to increase nitrogen in your soil.</p>
    <p align="justify">4. Plant ‘green manure’ crops like cabbage, corn and brocolli.</p>
    <p align="justify">5. <i>Use mulch (wet grass) while growing crops</i> - Mulch can also include sawdust and scrap soft woods. </p><hr style = "height:2px; background-color:#c79c60;">"""
}

```

'Nlow': """<b style = "color:#c79c60;">The N value of your soil is low.</b>

<br/><br/> Please consider the following suggestions:

<p align="justify"> 1. <i>Add sawdust or fine woodchips to your soil</i> – the carbon in the sawdust/woodchips love nitrogen and will help absorb and soak up and excess nitrogen.</p>

<p align="justify">2. <i>Plant heavy nitrogen feeding plants</i> – tomatoes, corn, broccoli, cabbage and spinach are examples of plants that thrive off nitrogen and will suck the nitrogen dry.</p>

<p align="justify">3. <i>Water</i> – soaking your soil with water will help leach the nitrogen deeper into your soil, effectively leaving less for your plants to use.</p>

<p align="justify">4. <i>Sugar</i> – In limited studies, it was shown that adding sugar to your soil can help potentially reduce the amount of nitrogen in your soil. Sugar is partially composed of carbon, an element which attracts and soaks up the nitrogen in the soil. This is similar concept to adding sawdust/woodchips which are high in carbon content.</p>

<p align="justify">5. Add composted manure to the soil.</p>

<p align="justify">6. Plant Nitrogen fixing plants like peas or beans.</p>

<p align="justify">7. <i>Use NPK fertilizers with high N value.</i></p>

<p align="justify">8. <i>Do nothing</i> – It may seem counter-intuitive, but if you already have plants that are producing lots of foliage, it may be best to let them continue to absorb all the nitrogen to amend the soil for your next crops.</p><hr style = "height:2px; background-color:#c79c60;">""",

'NNo': """<b style = "color:#c79c60;">The N value of your soil is up to the mark.</b><hr style = "height:2px; background-color:#c79c60;">"""

}

Next, the third module is for pesticide recommendation. There are two options for that. One is to manually select the pest and following pesticides will be recommended. So, as such there is no algorithm for that. It's a dictionary based solution. For the second option which is uploading the image of the pest, there needs to be an algorithm that identifies the pest and recommends the corresponding pesticide. The algorithm is as follows:

```

classifier = load_model('Trained_model.h5')
classifier._make_predict_function()

def pred_pest(pest):
    try:
        test_image = image.load_img(pest, target_size=(64, 64))
        test_image = image.img_to_array(test_image)
        test_image = np.expand_dims(test_image, axis=0)
        print(test_image.shape)
        result = classifier.predict_classes(test_image)
        return result
    except:
        return 'x'

def predict():
    if request.method == 'POST':
        file = request.files['image'] #fetch input
        filename = file.filename
        try:
            file_path = os.path.join('static/user uploaded',filename)
            file.save(file_path)

            pred = pred_pest(pest=file_path)
            if pred == 'x':
                return render_template('unapifile.html')
            if pred[0] == 0:
                pest_identified = 'aphids'
            elif pred[0] == 1:
                pest_identified = 'armyworm'
            elif pred[0] == 2:

```

```

pest_identified = 'beetle'
elif pred[0] == 3:
    pest_identified = 'bollworm'
elif pred[0] == 4:
    pest_identified = 'earthworm'
elif pred[0] == 5:
    pest_identified = 'grasshopper'
elif pred[0] == 6:
    pest_identified = 'mites'
elif pred[0] == 7:
    pest_identified = 'mosquito'
elif pred[0] == 8:
    pest_identified = 'sawfly'
elif pred[0] == 9:
    pest_identified = 'stemborer'

return render_template(pest_identified + ".html", pred=pest_identified)
except:
    return render_template("uploadpicture.html")

```

The algorithm for model training that is used in the above algorithm is given below.

```

# Part 1 - Building the CNN
# importing the Keras libraries and packages
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten, Dense, Dropout
from keras import optimizers

```

```

# Initializing the CNN
classifier = Sequential()

# Step 1 - Convolution Layer
classifier.add(Convolution2D(32, 3, 3, input_shape = (64, 64, 3), activation = 'relu'))

#step 2 - Pooling
classifier.add(MaxPooling2D(pool_size =(2,2)))

# Adding second and third convolution layer
classifier.add(Convolution2D(32, 3, 3, activation = 'relu'))
classifier.add(MaxPooling2D(pool_size =(2,2)))
classifier.add(Convolution2D(64, 3, 3, activation = 'relu'))
classifier.add(MaxPooling2D(pool_size =(2,2)))

#Step 3 - Flattening
classifier.add(Flatten())

#Step 4 - Full Connection
classifier.add(Dense(256, activation = 'relu'))
classifier.add(Dropout(0.5))
classifier.add(Dense(10, activation = 'softmax'))

#Compiling The CNN
classifier.compile(
    optimizer = 'adam',
    loss = 'categorical_crossentropy',
    metrics = ['accuracy'])

#Part 2 Fittting the CNN to the image

```

```

from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./255)

training_set = train_datagen.flow_from_directory(
    'Data/train',
    target_size=(64, 64),
    batch_size=32,
    class_mode='categorical')

test_set = test_datagen.flow_from_directory(
    'Data/test',
    target_size=(64, 64),
    batch_size=32,
    class_mode='categorical')

model = classifier.fit_generator(
    training_set,
    steps_per_epoch=100,
    epochs=100,
    validation_data = test_set,
    validation_steps = 6500
)

#Saving the model
import h5py
classifier.save('Trained_Model.h5')

```

That was all about modules, for login and signup, MongoDB Atlas was used, here algorithm is used only to store password in hashed format (SHA256), so as to keep it safe, even the password matching during login is not done by unhashing stored password and comparing it with input password, rather input password is hashed and then matched with stored password.

```
from passlib.context import CryptContext

context = CryptContext(
    schemes=["pbkdf2_sha256"],
    default="pbkdf2_sha256",
    pbkdf2_sha256__default_rounds=50000
)
```

## 5.4 System Screenshots

Just like Indian farmers put in their heart and soul to grow crops, similarly we also made an attempt to put our efforts to help them. The screenshots for the system are given as under. First, when the user opens the website then the landing page appears (Figure 26), from here the user has 3 functionalities: Home, Login, SignUp.

From “Home” the user can know what “Irrigreat” is about, it’s services and the team. If the user thinks that Irrigreat would be suitable for him/her and he/she wants to avail the service then the user can SignUp and create an account (as shown in Figure 27), three simple details are asked for: Email Address, Username and Password. Email Address must be unique which means that with one email address, the user can have only one account. The username must be unique and not be taken. The password has a minimum length of 8 characters and maximum length of 20 characters. If the user exceeds length or sets a password of length less than 8 characters then user is shown about the constraint.

To see the running project, click [here](#).

*Innigreat*

HOME LOGIN SIGNUP

## — Welcome To — IRRIGREAT

Farm = AI + Crop + Fertilizer + Pesticide. We take care of your soil's health. We at Irrigreat, honor the Indian Farmer's love, hard work and character. They help to feed a nation whose population is nearly 1.4 billion. However their productivity is threatened by some natural factors that can ruin their crops and their livelihoods. Irrigreat is a small initiative enhancing agriculture, make smart decisions to consider the demographics of your field, the factors affecting your crop, as well as how to keep them healthy for a super awesome yield.

*Our Services*

*Crop*  
**Recommendation**

Which crop to grow, that is ideally suited to conditions? Make an informed decision before starting the cultivation of crops.

*Fertilizer*  
**Recommendation**

Which fertilizer to choose for the cultivated soil and prescribed crop for optimum production? Provide your crops, the essential nutrients.

*Pesticide*  
**Recommendation**

Pests are a big threat. Use Irrigreat to know which pesticide to use to control pests before they cause potential damage to your crop?

---

**OUR TEAM**

SIMRAN  
MEHAK PURI

**Few words for our Indian Farmers**

For this one farmer the worries are over, lie down and rest your head. Your time has been and struggles enough, put the tractor in the shed.

Years were not easy, many downright hard, but your faith in God transcended. Put away your tools and sleep in peace. The fences have all been mended.

You raised a fine family, worked the land well and always followed the Son. Hang up your shovel inside of the barn; your work here on earth is done.

A faith few possess led your journey through life, often a jagged and stony way. The sun is setting, the cattle are all bedded, and here now is the end of your day.

Your love of God's soil has passed on to your kin; the stories flow like fine wine. Wash off your work boots in the puddle left by blessed rain one final time.

You always believed that the good Lord would provide and He always had somehow. Take off your gloves and put them down, no more sweat and worry for you now.

Your labor is done, your home now is heaven; no more must you wait, your legacy lives on, your love of the land, and we will close the gate.

**Connect With Us**

Simran

Mehak

Copyright © 2021 Simran & Mehak

Figure 26: Landing Page (before login)

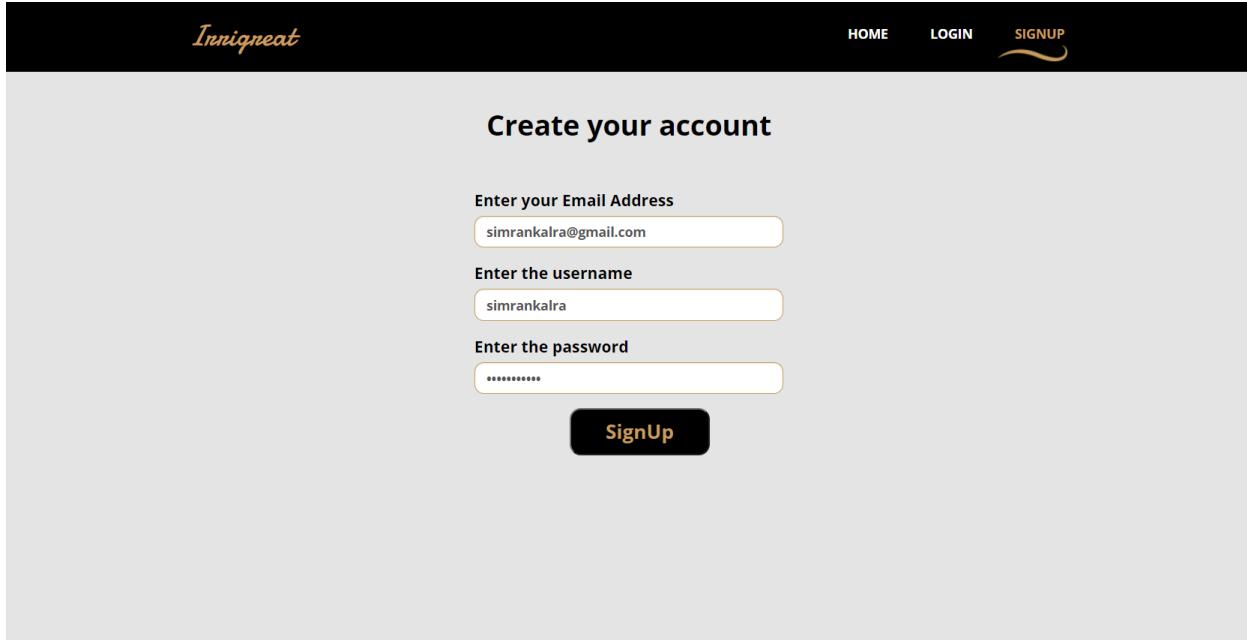


Figure 27: SignUp Page

After SignUp, the user can log into the system by filling username and password as shown in Figure 28 below.

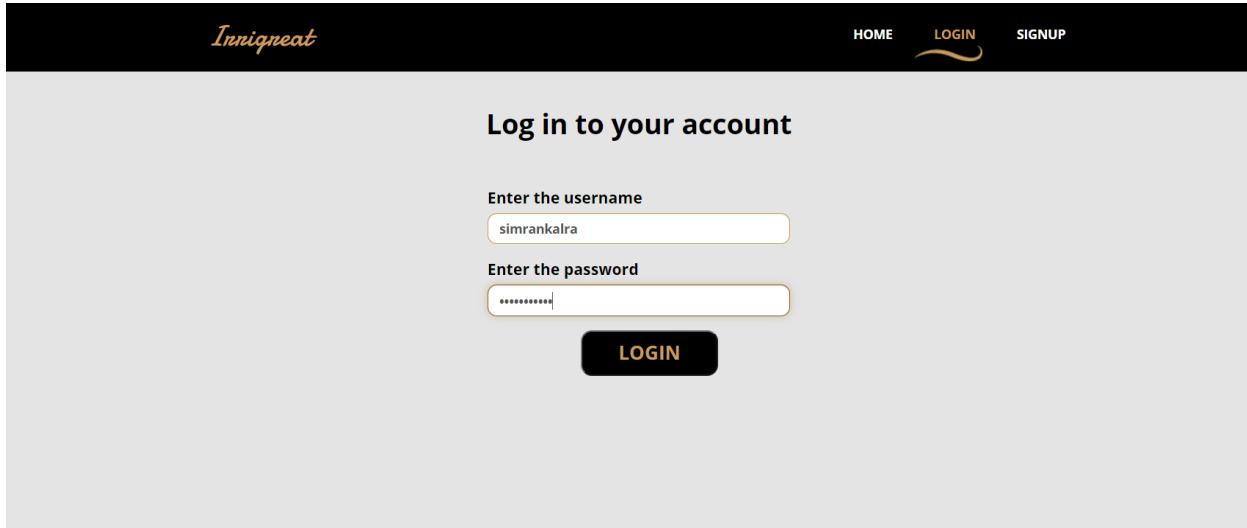


Figure 28: Login Page

Post successful login, the user gets to see a different view of the landing page and now the user can avail all the facilities of Irrigreat as shown in Figure 29.

*Innigreat*

HOME CROP FERTILIZER PESTICIDE FEEDBACK LOGOUT

Welcome simrankalra

## — Welcome To — IRRIGREAT

Farm + AI + Crop + Fertilizer + Pesticide. We take of your soil's health. We at Irrigreat, honor the Indian Farmer's love, hard work and character. They help to feed a nation whose population is nearly 1.4 billion. However their productivity is threatened by some natural factors that can ruin their crops and their livelihoods. Irrigreat is a small initiative enhancing agriculture, make smart decisions to consider the demographics of your field, the factors affecting your crop, as well as how to keep them healthy for a super awesome yield.

### Our Services

#### Crop Recommendation

Which crop to grow, that is ideally suited to conditions? Make an informed decision before starting the cultivation of crops.

[Proceed Now](#)

#### Fertilizer Recommendation

Which fertilizer to choose for the cultivated soil and prescribed crop for optimum production? Provide your crops, the essential nutrients.

[Proceed Now](#)

#### Pesticide Recommendation

Pests are a big threat. Use Irrigreat to know which pesticide to use to control pests before they cause potential damage to your crop?

[Proceed Now](#)

**OUR TEAM**

SIMRAN  
MEHAK PURI

**Few words for our Indian Farmers**

For this one farmer the worries are over, lie down and rest your head. Your time has been and struggles enough, put the tractor in the shed.

Years were not easy, many downright hard, but your faith in God transcended. Put away your tools and sleep in peace. The fences have all been mended.

You raised a fine family, worked the land well and always followed the Son. Hang up your shovel inside of the barn; your work here on earth is done.

A faith few possess led your journey through life, often a jagged and stormy way. The sun is setting, the cattle are all bedded, and here now is the end of your day.

Your love of God's soil has passed on to your kin; the stories flow like fine wine. Wash off your work boots in the puddle left by blessed rain one final time.

You always believed that the good Lord would provide and He always had somehow, Take off your gloves and put them down, no more sweat and worry for you now.

Your labor is done, your home now is heaven; no more must you wait, Your legacy lives on, your love of the land, and we will close the gate.

Copyright © 2021 Simran & Mehak

Figure 29: Landing page after login

Say that the user wants to avail the Crop Recommendation service then he/she can fill the values of N, P, K, pH, rainfall, temperature and relative humidity in the units specified to know about the crop that they must grow in their farm. See Figure 30 for more details.

**Find out the most suitable crop to grow in your farm**

Nitrogen (ratio)  
90

Phosphorous (ratio)  
42

Potassium (ratio)  
43

pH level  
6.5

Rainfall (in mm)  
202.9

Temperature (in °C)  
20.88

Relative Humidity (in %)  
82

**Recommend**

Figure 30: Crop Recommendation Module (Input the values)

Now, after the user pressed the “Recommend” button then the result will be shown on the screen (Figure 31), here in this case, it recommends “rice”. Hence this is most suitable to the soil as per current weather conditions and soil conditions.

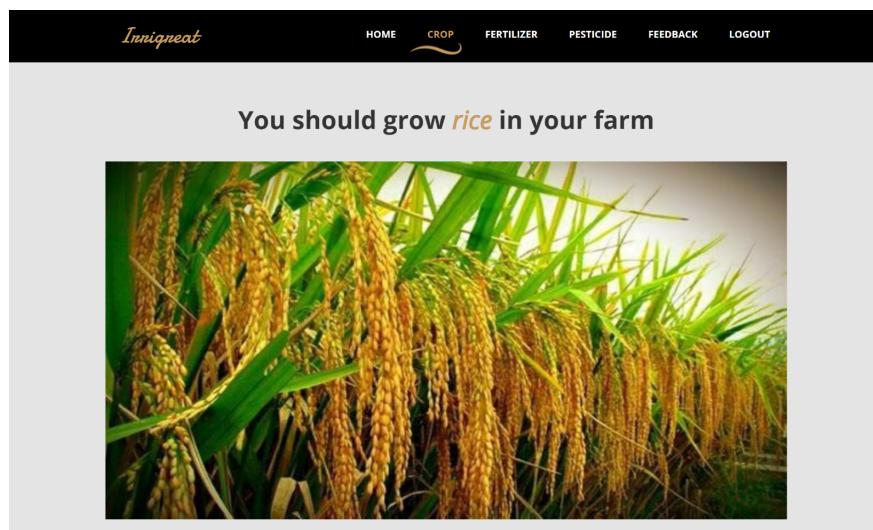


Figure 31: Crop Recommendation Module (Result)

Similarly the user can avail the “Fertilizer Recommendation” Service by filling the values for N, P, K and crop (Figure 32). Post that the user will know about the status of the soil and will tell the difference between the desired value of nutrients and the user's farm's nutrients and then “Irrigreat” will give informed advice on organic fertilizers to use as per the current condition of soil. See Figure 33 for reference.

The screenshot shows a web interface for the "Fertilizer Recommendation" module. At the top, there is a navigation bar with links for HOME, CROP, FERTILIZER (which is highlighted in yellow), PESTICIDE, FEEDBACK, and LOGOUT. Below the navigation bar, the title "Get informed advice on fertilizer based on soil" is displayed. The form consists of four input fields for nutrient ratios: "Nitrogen (ratio)" with value 74, "Phosphorous (ratio)" with value 46, and "Potassium (ratio)" with value 88. There is also a dropdown menu for "Crop you want to grow" set to "lentil". A large black button labeled "Recommend" is positioned below the input fields.

Figure 32: Fertilizer Recommendation Module (Input the values)

Difference between desired value of N and your farm's N value is 54.0

The N value of soil is high and might give rise to weeds.

Please consider the following suggestions:

1. *Manure* – adding manure is one of the simplest ways to amend your soil with nitrogen. Be careful as there are various types of manures with varying degrees of nitrogen.
2. *Coffee grinds* – use your morning addiction to feed your gardening habit! Coffee grinds are considered a green compost material which is rich in nitrogen. Once the grounds break down, your soil will be fed with delicious, delicious nitrogen. An added benefit to including coffee grounds to your soil is while it will compost, it will also help provide increased drainage to your soil.
3. *Plant nitrogen fixing plants* – planting vegetables that are in Fabaceae family like peas, beans and soybeans have the ability to increase nitrogen in your soil.
4. Plant 'green manure' crops like cabbage, corn and broccoli.
5. *Use mulch (wet grass) while growing crops* - Mulch can also include sawdust and scrap soft woods.

---

Difference between desired value of P and your farm's P value is 6.0

The P value of your soil is high.

Please consider the following suggestions:

1. *Avoid adding manure* – manure contains many key nutrients for your soil but typically including high levels of phosphorous. Limiting the addition of manure will help reduce phosphorus being added.
2. *Use only phosphorus-free fertilizer* – if you can limit the amount of phosphorous added to your soil, you can let the plants use the existing phosphorus while still providing other key nutrients such as Nitrogen and Potassium. Find a fertilizer with numbers such as 10-0-10, where the zero represents no phosphorous.
3. *Water your soil* – soaking your soil liberally will aid in driving phosphorous out of the soil. This is recommended as a last ditch effort.
4. Plant nitrogen fixing vegetables to increase nitrogen without increasing phosphorous (like beans and peas).
5. Use crop rotations to decrease high phosphorous levels.

---

Difference between desired value of K and your farm's K value is 48.0

The K value of your soil is high.

Please consider the following suggestions:

1. *Loosen the soil* deeply with a shovel, and water thoroughly to dissolve water-soluble potassium. Allow the soil to fully dry, and repeat digging and watering the soil two or three more times.
2. *Sift through the soil*, and remove as many rocks as possible, using a soil sifter. Minerals occurring in rocks such as mica and feldspar slowly release potassium into the soil slowly through weathering.
3. Stop applying potassium-rich commercial fertilizer. Apply only commercial fertilizer that has a '0' in the final number field. Commercial fertilizers use a three number system for measuring levels of nitrogen, phosphorous and potassium. The last number stands for potassium. Another option is to stop using commercial fertilizers all together and to begin using only organic matter to enrich the soil.
4. Mix crushed eggshells, crushed seashells, wood ash or soft rock phosphate to the soil to add calcium. Mix in up to 10 percent of organic compost to help amend and balance the soil.
5. Use NPK fertilizers with low K levels and organic fertilizers since they have low NPK values.
6. Grow a cover crop of legumes that will fix nitrogen in the soil. This practice will meet the soil's needs for nitrogen without increasing phosphorus or potassium.

---

Figure 33: Fertilizer Recommendation Module (Result)

Figure 34 shows the third module “Pesticide Recommendation” where the user can either upload an image of the pest or select the pest. If the user is informed of the pest that resides in his/her farm, then he/she can simply select the pest (Figure 37) and get the result for recommended pesticides (Figure 38), but if the user is unaware of the pests then he/she can upload the picture that clearly shows the pest (Figure 35) and then “Irrigreat” would identify the pest and recommended the pesticides accordingly (Figure 36). The user must take care that the picture must not be blur, since it may lead to wrong identification of the pest.

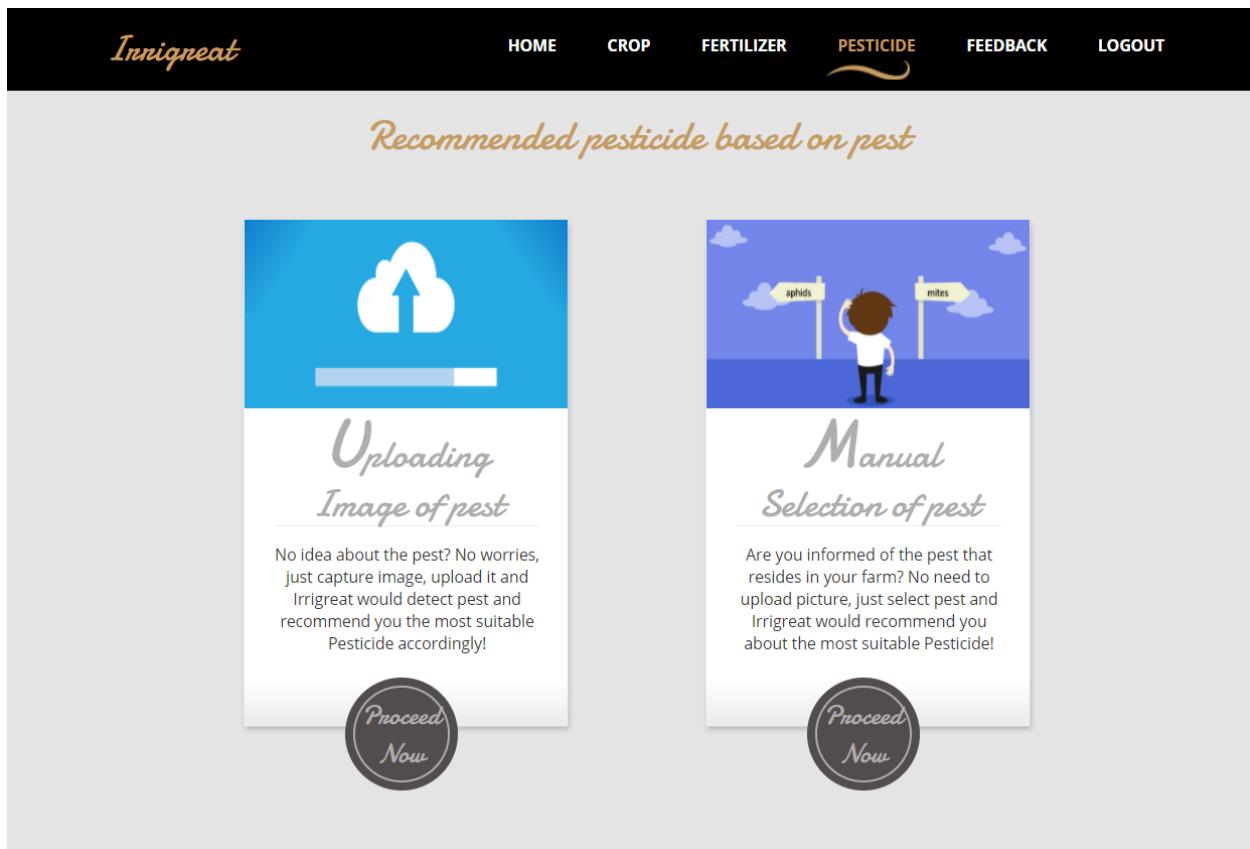


Figure 34: Pesticide Recommendation Module

## Recommended pesticide based on pest

Please upload the picture which clearly shows the pest,  
so that we can recommend you, the pesticide accordingly!

Choose File jpg\_12.jpg

**Recommend**

Figure 35: Pesticide Recommendation Module (Uploading Image)

**Identified Pest: bollworm**

### Recommended Products



Dose: 160-280 ml/Ha



Dose: 220 gm/Ha



Dose: 1000 ml/Ha



Dose: 600-100 ml/Ha

Figure 36: Pesticide Recommendation Module (Result after uploading image)

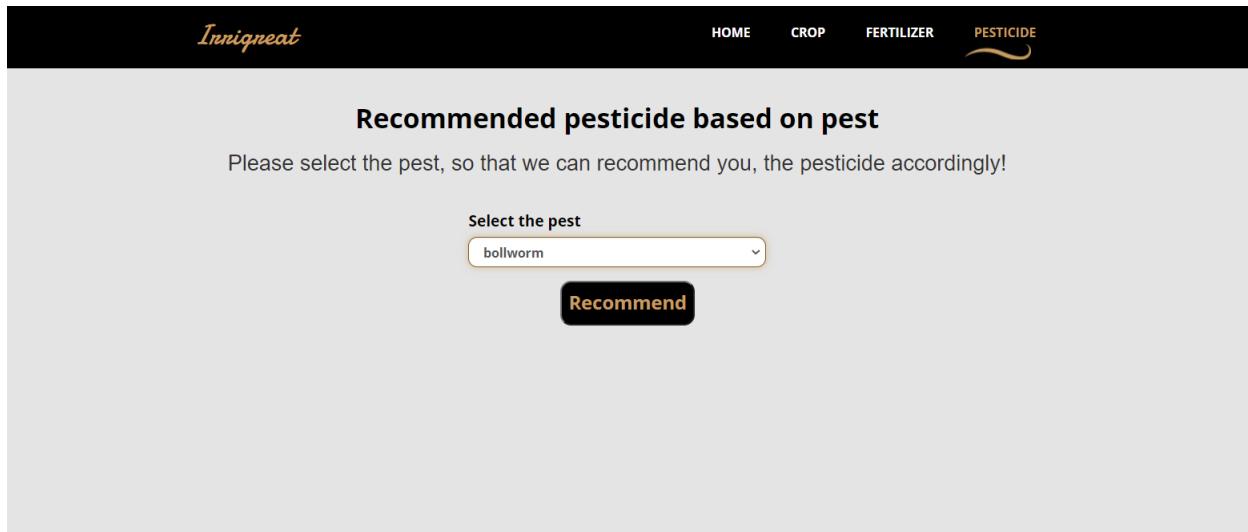


Figure 37: Pesticide Recommendation Module (Manual Selection)

A screenshot of the same web-based pesticide recommendation system. The navigation bar remains the same. The main content area now displays the message "Identified Pest: bollworm". Below this, a section titled "Recommended Products" shows four different pesticide containers in a grid. The first three products are shown in a row, and the fourth is shown below them. Each product has its name and recommended dose listed below it.

Recommended Products			
Dose: 160-280 ml/Ha	Dose: 220 gm/Ha	Dose: 1000 ml/Ha	Dose: 600-100 ml/Ha

Figure 38: Pesticide Recommendation Module (Results after Manual Selection)

Post any activity, the user can give feedback so that the team of “Irrigreat” could improve and provide better facilities. This is totally optional. Figure 39 shows the valuable feedback given by an user.

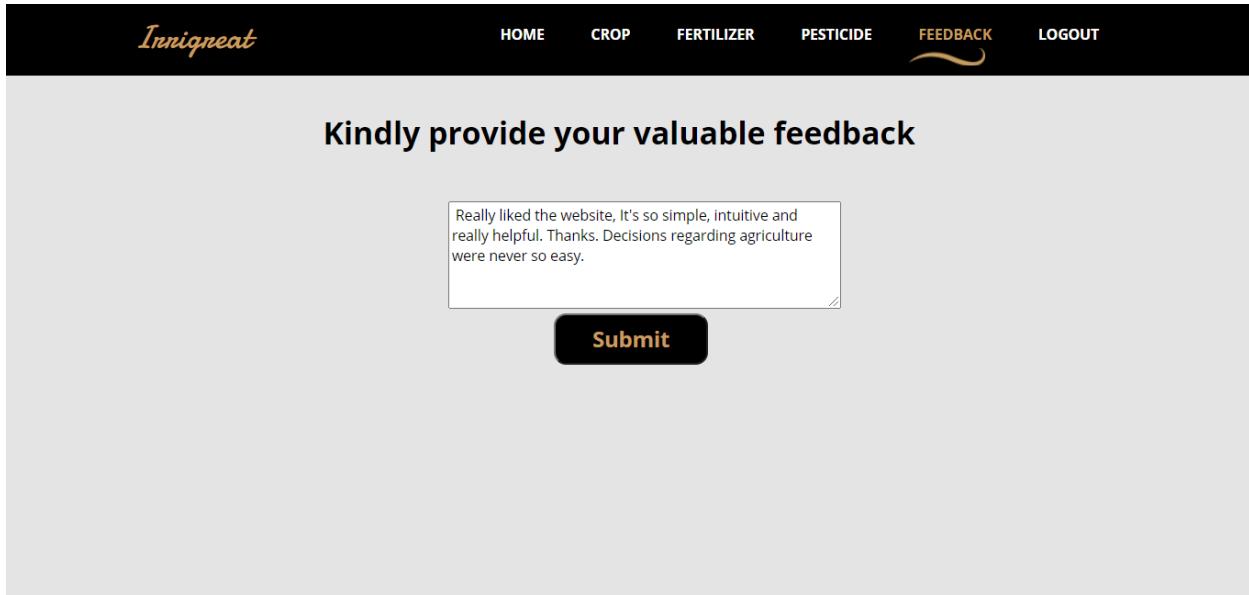


Figure 39: Feedback

## 5.5 Testing Process

A project is successful in real terms if it passes all the tests and if not all the tests then at least it must be able to handle the cases: base cases, corner cases with good accuracy. Some errors can be neglected depending upon application. Irrigreat also underwent a rigorous series of tests and how it performed is given below.

First module to be tested is SignUp. The test cases are given below in Table 8.

Table 8: Test Cases for SignUp

Sr. No.	Test Case	Expected Output	Actual Output	Pass
1	Email Address already exists	This email address is already registered	This email address is already registered	Yes
2	Username already exists	This username is already taken	This username is already taken	Yes
3	Password has length less than 8 characters	Please lengthen this text to 8 characters or more	Please lengthen this text to 8 characters or more	Yes
4	Unique email address, username and password length $\geq 8$ characters	Account created successfully	Account created successfully	Yes

Second module to be tested is login. The test cases are given below in Table 9.

Table 9: Test Cases for Login

Sr. No.	Test Case	Expected Output	Actual Output	Pass
1	Wrong Username	Invalid Credentials	Invalid Credentials	Yes
2	Wrong Password	Invalid Credentials	Invalid Credentials	Yes
3	Correct Username and password	User is logged in	User is logged in	Yes

Third module to be tested is Crop Recommendation. The test cases are given below in Table 10.

Table 10: Test Cases for Crop Recommendation

Sr. No.	Test Case	Expected Output	Actual Output	Pass
1	N value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
2	N value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
3	P value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
4	P value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
5	K value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
6	K value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
7	pH > 14	Value must be less than or equal to 14	Value must be less than or equal to 14	Yes
8	pH < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
9	Rainfall < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
10	Temperature < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
11	Temperature > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
12	Relative Humidity < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
13	Relative Humidity > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
14	Any field empty	Please fill out this field	Please fill out this field	Yes
15	N = 90, P = 42, K = 43, pH = 6.5, rainfall = 202.9, temperature = 20.88, relative humidity = 82	rice	rice	Yes
16	N = 3, P = 18, K = 31, pH = 6.39, rainfall = 91.09, temperature = 31.65, relative humidity = 48.2	mango	mango	Yes

Fourth module to be tested is Fertilizer Recommendation. The test cases are given below in Table 11.

Table 11: Test Cases for Fertilizer Recommendation

Sr. No.	Test Case	Expected Output	Actual Output	Pass
1	N value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
2	N value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
3	P value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
4	P value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
5	K value < 0	Value must be greater than or equal to 0	Value must be greater than or equal to 0	Yes
6	K value > 100	Value must be less than or equal to 100	Value must be less than or equal to 100	Yes
7	Any field empty	Please fill out this field	Please fill out this field	Yes
8	No crop selected	Selected Crop: apple	Selected Crop: apple	Yes
9	N = 45, P = 45, K = 78, crop = maize	Nlow, Plow, KHigh and measures	Nlow, Plow, KHigh and measures	Yes

Fifth module to be tested is Pesticide Recommendation. The test cases are given below in Table 12.

Table 12: Test Cases for Pesticide Recommendation

Sr. No.	Test Case	Expected Output	Actual Output	Pass
1	Manual Selection of pest: grasshopper	Recommended products: Malathion (Dose 750 g/L), Perfek (Dose 2.5-3.5 Tbsp / 16 L)	Recommended products: Malathion (Dose 750 g/L), Perfek (Dose 2.5-3.5 Tbsp / 16 L)	Yes
2	File uploaded is not image file	File format is not appropriate. Kindly upload an image file.	File format is not appropriate. Kindly upload an image file.	Yes
3	No file uploaded and pressed “Recommend” button.	No file uploaded! Kindly upload an image file.	No file uploaded! Kindly upload an image file.	Yes
4	Uploaded Image is of aphids	Identified pest: aphids	Identified pest: aphids	Yes
5	Uploaded Image is of armyworm	Identified pest: armyworm	Identified pest: aphids	No
6	Uploaded Image is of beetle	Identified pest: beetle	Identified pest: beetle	Yes
7	Uploaded Image is of sawfly	Identified pest: sawfly	Identified pest: sawfly	Yes
8	Uploaded image is blur image of sawfly	Identified pest: sawfly	Identified pest: stem borer	No

## 5.6 Results and Discussion

The proposed model for “Irrigreat” can be judged in various aspects. Firstly, for the crop recommendation, since ML model is used to predict the crop which would be best suited as per site specific parameters, so here accuracy score helps to tell about how effective the solution is. Ensemble model using majority voting technique was used. The learners are: Naive Bayes, kNN, SVM and Random Forest. Accuracy Score came out to be 96.44%. The desired accuracy was  $\geq 90\%$ , but ML model is able to achieve 96.44%, hence it's appreciable. Since, Fertilizer Recommendation is simply a dictionary based solution, so it is based on research performed by the team members. Last module is Pesticide Recommendation. If the user chooses to upload an image, then pesticide would be recommended post identification of the pest and pests are identified through the DL model which is CNN. Here, the performance metric is training and validation accuracy, training and validation loss which are shown below.

	ACCURACY	LOSS
<b>TRAINING</b>	0.9699	0.0712
<b>VALIDATION</b>	0.9520	0.4681

The graphs for the same could be seen in Figure 40 and 41. Figure 43 represents model accuracy vs epochs. Figure 44 represents model loss vs epochs.

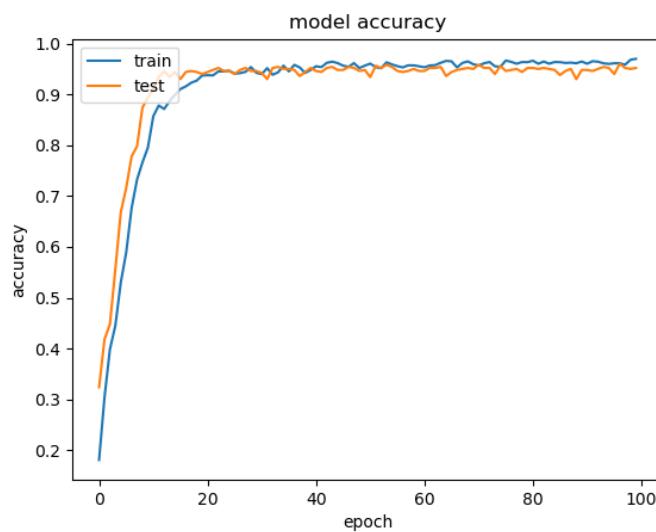


Figure 40: Model Accuracy vs Epochs

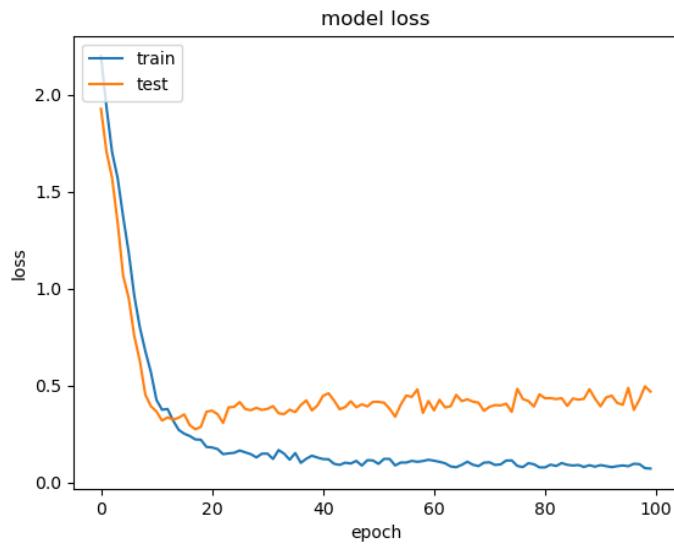


Figure 41: Model Loss vs Epochs

For pest identification, DL model is able to perform quite well but for some cases, DL model identifies “armyworm” as aphids due to close resemblance. Also, the system is not able to perform well with blur images, hence the user must upload the pictures that clearly show the pest.

## **5.7 Validation of Objectives**

Table 13 shows the status (successful/unsuccessful) of the objectives for “Irrigreat”.

Table 13: Validation of Objectives

S.No.	Objectives	Status
1	To implement precision agriculture	Successful
2	To solve the problem by proposing a recommendation system through an ensemble model with majority voting technique crop for the site specific parameters with high accuracy and efficiency.	Successful
3	To recommend organic fertilizer on the basis of N, P, K values and crop.	Successful
4	To recognize the pest and recommend particular pesticide available in India as per ISO standards (ISO 9001, ISO 14001, ISO 17025).	Successful
5	To design a web application for achieving above objectives.	Successful

## **CONCLUSION AND FUTURE SCOPE**

---

In this section conclusion and future scope with regards to this project is discussed.

### **6.1 Work accomplished w.r.t. Objectives**

The work accomplished w.r.t. objectives is given below.

#### **1. To implement precision agriculture**

- a. Using research data of soil characteristics, soil types, crop yield data collection “Irrigreat” suggests the farmers the right crop based on their site specific parameters to reduce the wrong choice on a crop and increase in productivity.

#### **2. To solve the problem by proposing a recommendation system through an ensemble model with majority voting technique crop for the site specific parameters with high accuracy and efficiency.**

- a. Learners used are: kNN, Random Forest, SVM, Naive Bayes
- b. Accuracy achieved: 96.44%

#### **3. To recommend organic fertilizer on the basis of N, P, K values and crop.**

- a. Fertilizer Dictionary is created for natural fertilizers as per the labels: NHigh, Nlow, NNo, PHigh, Plow, PNo, KHigh, Klow, KNo.

#### **4. To recognize the pest and recommend particular pesticide available in India as per ISO standards.**

- a. ISO 9001, ISO 14001, ISO 17025 are followed for pesticide recommendation

#### **5. To design a web application for achieving above objectives.**

- a. Website is designed and deployed.

## **6.2 Conclusion**

India's farmers are hard at work. They help to feed a nation whose population is nearly 1.4 billion. However their productivity is threatened by some natural factors that can ruin their crops and their livelihoods.

So, this solution (Irrigreat) will benefit farmers to maximize productivity in agriculture, reduce soil degradation in cultivated fields, and have informed advice on organic fertilizers/ other fertilizers and also know about the right crop by considering various attributes. This would provide a comprehensive prediction and hence benefit both farmers and the environment. Not only this, but pest control would also be a major issue to be solved via this project.

## **6.3 Environmental/ Economic/ Social Benefits**

Irrigreat suggests the crops based on soil characteristics, thereby preventing soil degradation which saves the environment. Natural fertilizers also benefit the environment. Pesticides that are recommended are as per ISO standards. Social benefits include that it will be helping that section of India to feed the nation of 1.4 billion, which means Indian farmers. Economic benefits are abundant because availing services of Irrigreat just requires the user to have an account on the website which is absolutely free.

## **6.4 Reflections**

The whole journey of building “Irrigreat” has been a valuable experience, starting with the discovery of possible opportunities to think of the idea to the phase where the same idea was actually deployed. The team gained insight into the field of software development and now in the future, members shall feel more confident in the process of project development. Furthermore, it was learnt how to analyze the existing frameworks and perform literature surveys and utilize that analysis to identify the problem statement, research gaps and come up with the solution ideas. It was a learning of how to incorporate and take care of the user requirements. It was the time when the importance of documentation was realized and what are techniques involved in being organized about it. One of the takeaways was how to manage the resources in an efficient manner and most importantly to use common sense and build a viable and efficient model, but the best takeaway was development of analytical skills while working in the team and discussing

each point of the assigned task in detail. The whole project helped us in exploring the skills as a computer engineer and improved confidence levels, ability to work under pressure and helped in learning project management techniques. It aided the members to be familiarized with the working and delivering of projects and how to build an entire product from just an idea.

## 6.5 Future Scope

“Irrigreat” is not limited to current usage, it can be extended to many features as discussed below”:

1. Irrigreat currently supports 22 crops that are apple, banana, blackgram, chickpea, coconut, coffee, cotton, grapes, jute, kidney beans, lentil, maize, mango, mothbeans, mungbean, muskmelon, orange, papaya, pigeon peas, pomegranate, rice, watermelon. Later on, the admin can add other crops. Moreover in the future, fertilizers can also be added accordingly. The training was done on 10 pests: aphids, armyworm, beetle, bollworm, earthworm, grasshopper, mites, mosquito, sawfly and stem borer and with this pesticides are suggested. In future, training can be done on more pests and more pesticides can also be added according to the pests.
2. In Crop Recommendation, values are manually entered by user of temperature, humidity, rainfall. Admin can also use some weather API to fetch the real time parameters by the city and state.
3. In Pesticide Recommendation, the uploaded image should be clear for correct results, otherwise with a blur image, the system sometimes gives wrong results so, further filters can be used to obtain better results. Also the system can use better DL models.
4. In future pesticide code can be integrated with drone code so that it can take live pictures of pests and by email or by mobile the farmers would be notified about the pest along with the pesticides.

## PROJECT METRICS

---

This section talks about various project metrics.

### 7.1 Challenges Faced

Every new learning and trying unexplored areas always come with challenges. The performance of “Irrigreat” is entirely dependent on datasets, ML model and DL model. Crop recommendation dataset is easily available on kaggle. Creating a fertilizer dictionary was a challenge. The dataset for exact N, P, K values for the crops was not available at one place, so it was quite challenging to browse through various websites while also keeping in mind that the sources must be authentic and verified. Another challenge was related to pest dataset. No open source dataset is available so the dataset was custom made by writing an automatic script that scrape the images from google using selenium and chrome driver. Along with this finding the pesticides that were as per the ISO standards also a daunting task. Since the model used CNN algorithm, setting hyperparameters was another task. With a lot of hits and trials, some thumb rules, hyperparameters were tuned to get the best out of the algorithm. The model suffered from overfitting which was tackled by the Dropout method. Later on, the next problem that everyone faced was coordination among team members during covid-19 time period when all were at their places, anyhow work from home was well-versed due to technology like Team Viewer to help get access of PC where the main project is made, Zoom and Google Meet for sharing and discussion of work progress. Despite challenges, it was really amazing and quite a learning journey starting from scratch and then making things work. Team is the core part which makes everything work.

## 7.2 Relevant Subjects

The Capstone Project requires the knowledge of multiple subjects. Some of them are direct learnings from courses (taught in Institute), and some are the skills to be learnt by the developers themselves. Table 14 provides information about the relevant subjects used to successfully complete the project.

Table 24: Subject Code and Subject Name

Subject Code	Subject Name	Description
UML501	Machine Learning	To get better insight about how “crop prediction” problems can be looked at and hence, solved.
UCS742	Deep Learning	To apply the most suitable deep learning algorithm (CNN) for pest identification, and also knowledge of optimizers, loss functions.
UCS503	Software Engineering	Software Development Lifecycle, Preparation of SRS, Working as per Scrum model, Shaping functional and non-functional requirements, Understanding and communication of ideas through UML Diagrams, Creating Design level and analysis level diagrams and most importantly the system architecture.
UCS806	Ethical Hacking	To store the passwords in hashed format.

### **7.3 Interdisciplinary Knowledge Sharing**

From doing this project, the members gained knowledge with regards to computer engineering aspects like machine learning, deep learning, web development, databases and also about some aspects of agriculture. Learning was also with regards to target users- Indian farmers and also got some knowledge about the technicalities involved in growing crops. Major revelation was about their everyday problems, how they face such challenges and what new ways have been brought into their lives through modernization.

### **7.4 Peer Assessment Matrix**

Table 15 gives a view on Peer Assessment Score and Table 16 provides Student information w.r.t. Student Codes.

Table 15: Peer Assessment Marks

		<b>Evaluation of</b>	
		<b>S1</b>	<b>S2</b>
<b>Evaluation By</b>	<b>S1</b>	5	5
	<b>S2</b>	5	5

Table 16: Student Information

<b>Student Code</b>	<b>Student Roll No.</b>	<b>Student Name</b>
S1	101703545	Simran
S2	101883015	Mehak Puri

## 7.5 Role Playing and Work Schedule

Although all members have equally contributed to the project, the major portion of the work was done by dividing them among team members. Table 17 shows the individual roles and contributions of the team members (screenshot taken from ProjectLibre).

Table 17: Individual Roles and Contributions

	Name	Work
1	Simran	353.5 hours
	<i>Data Acquisition</i>	12 hours
	<i>DL Model Training</i>	24 hours
	<i>Data Acquisition</i>	12 hours
	<i>Study on crops, fertilizers, pests and pesticides</i>	21 hours
	<i>Difference between desired and actual</i>	0.5 hours
	<i>Requirement Analysis</i>	15 hours
	<i>Front end design</i>	112 hours
	<i>Analysis Modelling, Design Modelling etc</i>	36 hours
	<i>ML Model using Ensemble method &amp; creating .pkd file</i>	9 hours
2	Mehak	239 hours
	<i>Data Acquisition</i>	9 hours
	<i>Analysis Modelling, Design Modelling etc</i>	36 hours
	<i>Pest Detection &amp; corresponding Pesticide recommendation</i>	24 hours
	<i>Values Input</i>	0.5 hours
	<i>Studying Research papers</i>	21 hours
	<i>Values Input</i>	0.5 hours
	<i>Data Cleaning and Data Augmentation</i>	9 hours
	<i>Using .pkd file for crop recommendation</i>	6 hours
	<i>Dictionary based solutions</i>	21 hours
	<i>Flask Coding</i>	112 hours

## 7.6 Student Outcomes Description and Performance Indicators (A-K Mapping)

Table 18 provides insight of AK Mapping of various concepts as shown above.

Table 18: AK Mapping of various concepts

SO	Description	Outcome
A1	Applying mathematical concepts to obtain analytical and numerical solutions.	Used mathematical concepts like calculating the difference between desired and actual value for N, P, K for a particular crop.
A2	Applying basic principles of science towards solving engineering problems.	Used the principles that N value of soil helps the plant to green up, P value helps the plants to produce blooms and K value promotes all round well being.
A3	Applying engineering techniques for solving computing problems.	Used ML model for crop prediction and DL model for pest prediction.
B1	Identify the constraints, assumptions and models for the problems.	Picture of the pest must be clear and the user must have internet access so as to avail the service of Irrigreat.
B2	Use appropriate methods, tools and techniques for data collection.	Proper research on agriculture was done in order to prepare an accurate dataset.
B3	Analyze and interpret results with respect to assumptions, constraints and theory.	Made the model portable and affordable.
C1	Design software system to address desired needs in different problem domains.	Helped in specialisation and better understanding.
C2	Can understand scope and constraints	Provided the farmers one stop solution for

	such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.	crops, fertilizers and pesticides.
<b>D1</b>	Fulfill assigned responsibility in multidisciplinary teams.	Learned the virtue of teamwork, multitasking and punctuality.
<b>D2</b>	Can play different roles as a team player.	Increases efficiency and interdisciplinary knowledge.
<b>E1</b>	Identify engineering problems.	Learned to think logically, critically and practically.
<b>E2</b>	Develop appropriate models to formulate solutions.	Built several prototypes to achieve the final one.
<b>E3</b>	Use analytical and computational methods to obtain solutions.	Used sklearn, CNN, ssl, passlib, keras, tensorflow and various other modules.
<b>F1</b>	Showcase professional responsibility while interacting with peers and professional communities.	Boosted confidence, listening and speaking skills.
<b>F2</b>	Able to evaluate the ethical dimensions of a problem.	Reviewed stats in order to come up to this final project.
<b>G1</b>	Produce a variety of documents such as laboratory or project reports using appropriate formats.	Taking notes and maintaining a weekly diary helped in reviewing work from time to time.
<b>G2</b>	Deliver well-organized and effective oral presentation.	Helped the panel to understand and give suggestions.
<b>H1</b>	Aware of the environmental and societal impact of engineering solutions.	Less consumption of power, helping the farmers out.

<b>H2</b>	Examine economic tradeoffs in computing systems.	Using high computational yet economical components was the main objective.
<b>I1</b>	Able to explore and utilize resources to enhance self-learning.	Reading literature surveys of different products, analyzing their drawbacks and removing them from our model was the goal.
<b>I2</b>	Recognize the importance of life-long learning.	Practicality always increases the learnability than reading theory.
<b>J1</b>	Comprehend the importance of contemporary issues.	Understanding the importance and responsibility to work in this field in order to work for Indian farmers.
<b>K1</b>	Writing code in different programming languages.	Helped in learning different programming languages, their advantages over others.
<b>K2</b>	Apply different data structures and algorithmic techniques.	Helped in optimization.
<b>K3</b>	Use software tools necessary for computer engineering domain.	Understanding working to increase the accuracy and speed rather than doing it manually.

## **7.7 Brief Analytical Assessment**

### **Q1. What sources of information did your team explore to arrive at the list of possible Project Problems?**

**Ans:** The different sources used were first going through research papers and taking out keywords. Then, the team discussed the approach with the mentor Dr. Seema Bawa. The team studied agriculture to understand the field in a detailed manner. The team explored already available solutions and their drawbacks. Then after seeing the pros and cons of different modules, the project was finalized.

### **Q2. What analytical, computational and/or experimental methods did your project team use to obtain solutions to the problems in the project?**

**Ans:** The analytical methods to reach solutions were taking the lead generation survey, performing literature survey to analyse existing frameworks and identifying research gaps and studying existing solutions. Design diagrams and UML diagrams along with software requirement specification were equally very important to think through the solution ideas and make necessary changes. Experimental methods like prototyping and then its analysis, checking efficiency of the product though testing also helped in reaching the final solution.

### **Q3. Did the project demand demonstration of knowledge of fundamentals scientific and/or engineering principles? If yes, how did you apply?**

**Ans:** Irrigreat used basic principles of science and engineering. The team used various algorithms for crop prediction, pest prediction and password storing. Further algorithms like CNN were used to train the model to identify the pest.

### **Q4. How did your team share responsibility and communicate the information of schedule with others in a team to coordinate design and manufacturing dependencies?**

**Ans:** Our team had good communication throughout the project. We had one official whatsapp group with our mentor and one was of all the team members. Our team leader used to assign tasks to all group members and we were asked to finish it well in time. Whatever problems came

across were discussed and if we were not able to solve it on our own then we used to ask our mentor. Also, there were regular meetings with ma'am and among ourselves.

**Q5. What resources did you use to learn new materials not taught in class for the course of the project?**

**Ans:** We used various resources such as the internet, books in the library. We went through a lot of research papers, understanding of various products already made in this field, and understanding their drawbacks. We also consulted our mentor whenever in doubt. This project wouldn't have been possible without the guidance of our mentor.

**Q6. Does the project make you appreciate the need to solve problems in real life using engineering and could the project development make you proficient with software development tools and environments?**

**Ans:** This capstone project taught us a lot of things. Not only the working of the project is important but also it teaches us how the documentation should go side by side which keeps us in track of what part is left, what is done. Also, it taught us how real-life problems are much different than the theory one. We learn better when we are doing things and seeing it rather than reading from the book. We learnt how to use various new languages, software and tools.

## APPENDIX A: REFERENCES

---

- [1] Rajak, Rohit Kumar, et al. “Crop Recommendation System to Maximize Crop Yield using Machine Learning Technique.” *International Research Journal of Engineering and Technology (IRJET)*, vol. 04, no. 12, 2017, pp. 951-952. *IRJET*, <https://www.irjet.net/archives/V4/i12/IRJET-V4I12179.pdf>.
- [2] Dighe, Deepti, et al. “Crop Recommendation System for Precision Agriculture.” *IRJET*, vol. 05, no. 11, 2018, pp. 476-480. *IRJET*, <https://www.irjet.net/archives/V5/i11/IRJET-V5I1190.pdf>.
- [3] Mokarrama, Miftahul Jannat, and Mohammad Shamsul Arefin. “RSF: A Recommendation System for Farmers.” *Region 10 Humanitarian Technology Conference*, vol. 2, no. 17, 2017, [https://www.researchgate.net/publication/323203384\\_RSF\\_A\\_recommendation\\_system\\_for\\_farmers](https://www.researchgate.net/publication/323203384_RSF_A_recommendation_system_for_farmers).
- [4] Gadge, Yogesh, and Sandhya. “A study on various data mining techniques for crop yield prediction.” *IEEE Xplore*, 2017. *IEEE Xplore*, <https://ieeexplore.ieee.org/document/8284541>.
- [5] Mishra, Shruti, et al. *Use of data mining in crop yield prediction*. 2018. ResearchGate, [https://www.researchgate.net/publication/326073480\\_Use\\_of\\_data\\_mining\\_in\\_crop\\_yield\\_prediction](https://www.researchgate.net/publication/326073480_Use_of_data_mining_in_crop_yield_prediction).
- [6] Wu, Xiaoping, et al. *A Large-Scale Benchmark Dataset for Insect Pest Recognition*. 2019. *IEEE Xplore*,

[https://openaccess.thecvf.com/content\\_CVPR\\_2019/papers/Wu\\_IP102\\_A\\_Large-Scale\\_Benchmark\\_Dataset\\_for\\_Insect\\_Pest\\_Recognition\\_CVPR\\_2019\\_paper.pdf](https://openaccess.thecvf.com/content_CVPR_2019/papers/Wu_IP102_A_Large-Scale_Benchmark_Dataset_for_Insect_Pest_Recognition_CVPR_2019_paper.pdf).

[7] Kasinathan, Thenmozhi, et al. *Insect classification and detection in field crops using modern machine learning techniques*. 2020. *Science Direct*,

<https://www.sciencedirect.com/science/article/pii/S2214317320302067>.

[8] Ding, Weiguang, and Graham Taylor. *Automatic moth detection from trap images for pest management*. 2016. *Science Direct*,

[https://www.sciencedirect.com/science/article/pii/S0168169916300266?casa\\_token=EM\\_2nL2kQswAAAAA:POy8LOCK6U1FOou5RxJujTtBAhO5ofZxktJ4jmUBcDryDZHXBgvrNS5dJi6xwsu9vZVnAULFoh](https://www.sciencedirect.com/science/article/pii/S0168169916300266?casa_token=EM_2nL2kQswAAAAA:POy8LOCK6U1FOou5RxJujTtBAhO5ofZxktJ4jmUBcDryDZHXBgvrNS5dJi6xwsu9vZVnAULFoh).

[9] TÜRKOĞLU, Muammer, and Davut HANBAY. *Plant disease and pest detection using deep learning-based features*. 2018. *Turkish Journal of Electrical Engineering & Computer Sciences*, <https://journals.tubitak.gov.tr/elektrik/issues/elk-19-27-3/elk-27-3-6-1809-181.pdf>.

[10] Selvaraj, Michael Gomez, et al. *AI-powered banana diseases and pest detection*. 2019.

*Plant Methods*,

[https://plantmethods.biomedcentral.com/articles/10.1186/s13007-019-0475-z?utm\\_source=dlvr.it&utm\\_medium=twitter](https://plantmethods.biomedcentral.com/articles/10.1186/s13007-019-0475-z?utm_source=dlvr.it&utm_medium=twitter).