PROJECT REPORT

FERTILIZER RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

Submitted by

Team ID :PNT2022TMID39927

U DWARAKESWARAN (511319106009)

M KALAISELVAN(511319106019)

B VINAY(511319106005)

V JAGAN(511319106015)

IN PARTIAL FULFILMENT FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING

IN

ELECTRONIC COMMUNICATION ENGINEERING

KINGSTON ENGINEERING COLLEGE ,VELLORE

FERTILIZER RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX Source Code GitHub & Project Demo Link

1.INTRODUCTION

1.1 OVERVIEW:

Since the past days and in the present days too farmers use their naked eyes to predict the crop disease and what kind of fertilizers required for the problem. But which leads to dangers state of agriculture, because some or most of problem look almost similar which leads the famer to confused state. In order to take a well defined solution for the problem, the famers should have the detailed knowledge about the type of diseases and should have the lot experience to make actual decisions.

To over come the above problem, a application is developed “FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION”, through this application the farmer can easily find out problem and solution of this too. This project contains two dataset such as fruit and vegetable data set. the collected data set is trained and tested with ML and DL. Through Convolution Neural Networks (CNN) techniques provides high accuracy by identifying the diseases on leaves and recommend pesticides for the disease.

1.2 PURPOSE:

This system recommends us the best suitable crop to grow and the composition of fertilizers to be used. And also the system used to predict and provides the solution for the fruits and vegetable diseases.

2.LITERATURE SURVEY.

2.1 EXISTING PROBLEM.

The foremost existing problem in all over world in the field of agriculture is “RECOMMENDATION OF BEST FERTILIZER FOR PARTICULAR CROP “. All through the main application have been developed for this problem which doesn’t provide the accuracies has much expected or it may consist of less number of train and test sets of data.

Image processing techniques in agriculture are broadly extending their applications for disease diagnosis for different crops in an accurate and time efficient way [7], [8]. Sankaran et al.[7] have presented a review of different spectroscopic and imaging techniques currently being used for detecting plant diseases and monitoring plant health. They highlight in this paper (a) fluorescence and hyper-spectral imaging techniques, and (b) different spectroscopic techniques in visible, infrared, fluorescence and multi-spectral bands for plant disease or stress detection. Barbedo et al.[8] discuss different techniques adopted for detecting, quantifying and classifying diseases or pests mostly affecting stems and leaves of a wide range of crops for the visible band from digital images. This survey reveals that one of the challenge of the existing image processing tools for disease detection to be adopted in real time applications is that the algorithms are mainly developed on images which are captured under a particular condition of lighting, with a specific angle of capture and at a particular distance of the image capturing device from the object. In recent years, some research has gone into plant disease identification considering images captured under uncontrolled environment. Sannakki et al.[9] proposed an algorithm for identification of Downy-Mildew and Powdery-Mildew affected portion of the leaves while images considered in the study were captured with complex background. Lloret et al. [10] developed a system for capturing the status of health of the plants in a grape vineyard by analysing the images captured with a number of webcams for detecting and quantifying diseased plants. Wang et al. [11] proposed an image processing and neural network based algorithm to classify between Powdery- and Downy-Mildew diseases of grapes. In this paper, performance of different machine learning approaches like back-propagation neural network, Generalized regression, Probabilistic and Radial Basis Function neural network have been compared. Our aim is to develop an algorithm to detect and classify between different leaf diseases in grapes while the images considered for the study are captured under uncontrolled environment with complex background. Disease considered in this study are Anthracnose, Powdery Mildew and Downy Mildew. The main contribution of this paper is use of Random Forest classification algorithm using GLCM features for separating disease patches from background and classifying between different disease patches.

2.2 REFERENCES:

[1] Semi-automatic leaf disease detection and classification system for soybean culture IET Image Processing, 2018

[2] Cloud Based Automated Irrigation And Plant Leaf Disease Detection System Using An Android Application. International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017.

[3] Ms. Kiran R. Gavhale, Ujwalla Gawande, Plant Leaves Disease detection using Image Processing Techniques, January 2014. https://www.researchgate.net/profile/UjwallaGawande/publication/314436486_An_Overview_of_the _Research_on_Plant_Leaves_Disease_detection_using_Image_Processing_Techniques/links/5d37106 64585153e591a3d20/An-Overviewof-the-Research-on-Plant-Leaves-Diseae detection-using-Image-ProcessingTechniques.pdf

[4] Duan Yan-e, Design of Intelligent Agriculture Management Information System Based on IOT‖, IEEE,4th, Fourth International reference on Intelligent Computation Technology and Automation, 2011 https://ieeexplore.ieee.org/document/5750779

[5] R. Neela, P. Fertilizers Recommendation System For Disease Prediction In Tree Leave International journal of scientific & technology research volume 8, issue 11, november 2019 http://www.ijstr.org/final-print/nov2019/Fertilizers-Recommendation-System-For-Disease-PredictionIn-Tree-Leave.pdf .

[6] Swapnil Jori1, Rutuja Bhalshankar2, Dipali Dhamale3, Sulochana Sonkamble , Healthy Farm: Leaf Disease Estimation and Fertilizer Recommendation System using Machine Learning,International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211

[7] Detection of Leaf Diseases and Classification using Digital Image Processing International Conference on Innovations in Information, Embedded and Communication Systems(ICIIECS), IEEE, 2017.

[8] Shloka Gupta ,Nishit Jain ,Akshay Chopade, Farmer's Assistant: A Machine Learning BasedApplication for Agricultural Solutions.

2.3 PROBLEM STATEMENT DEFINITION:

(i) Find the suitable composition of the fertilizers.

(ii) Find the composition of the recommended fertilizer

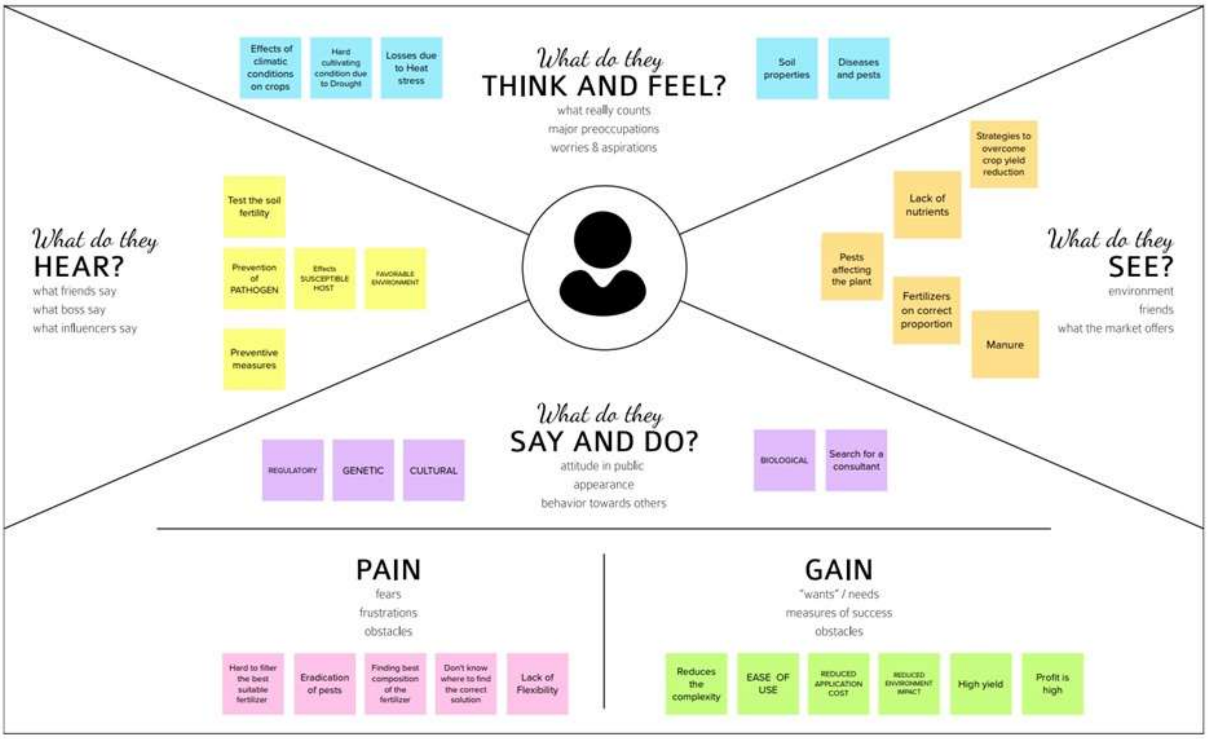
(iii)which is the best suitable plant

The above question arised because of the following reasons, Faced huge loss, Many customers are not having a clear idea, wants to make more profit.

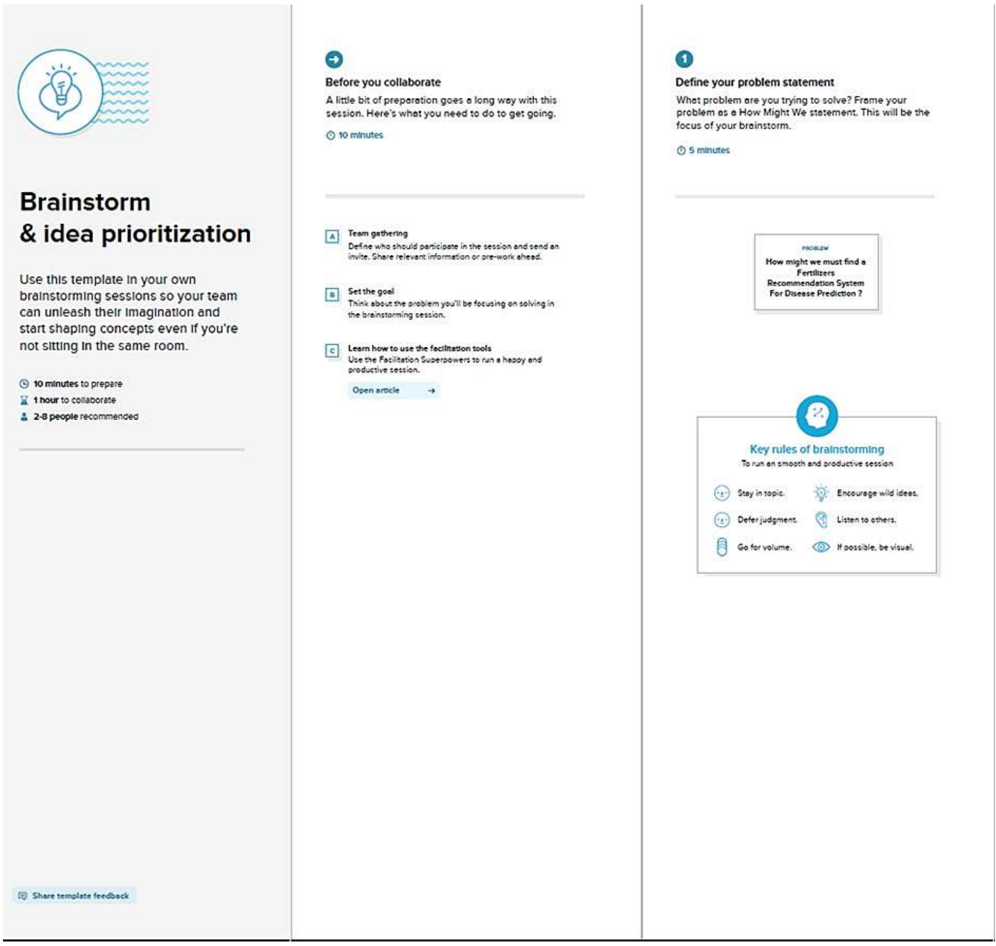
The purposed solution for the problem is, An automated system is introduced to identify different diseases on the plants by checking the symptoms shown on the leaves of the plant. Deep learning techniques are used to identify the diseases and suggest the precautions that can be taken for those diseases.

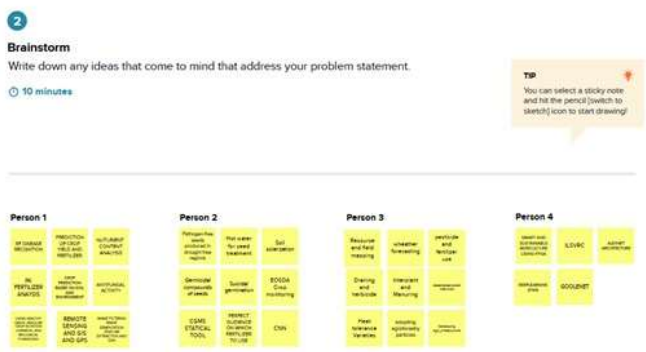
3.IDEATION &PROPOSED SOLUTION.

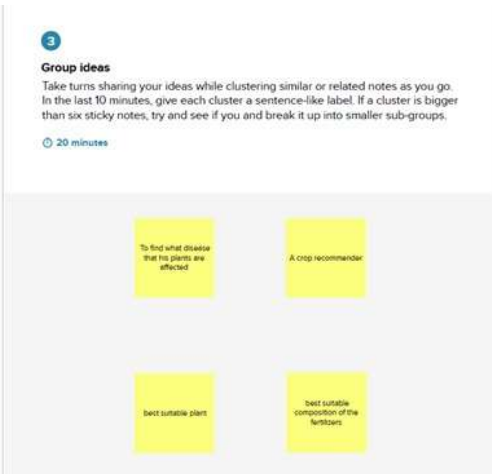
3.1 EMPATHY MAP CANVAS:



3.2 IDEATION &BRAINSTORMING:







4

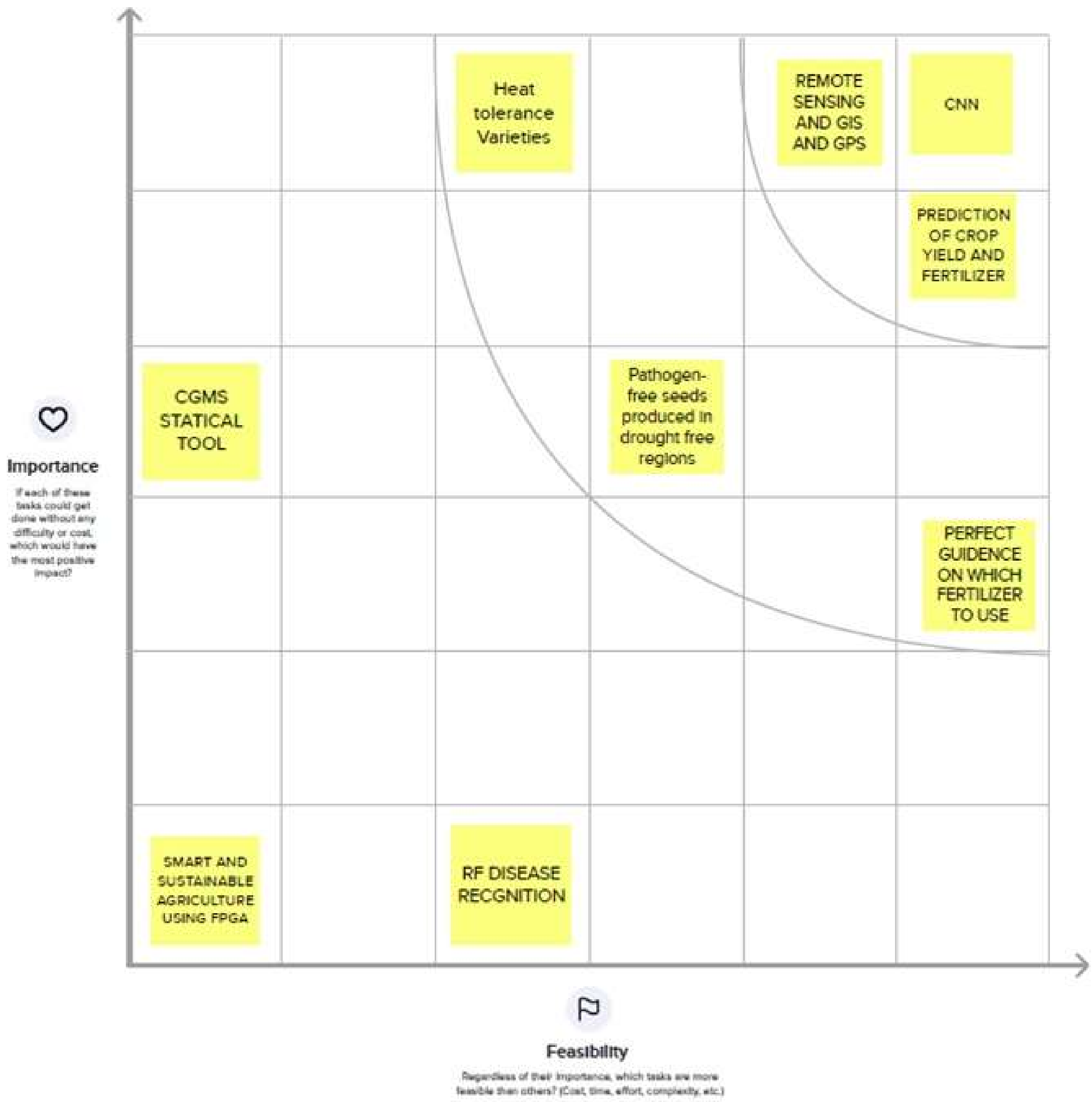
Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.



PNT2022TMID39927

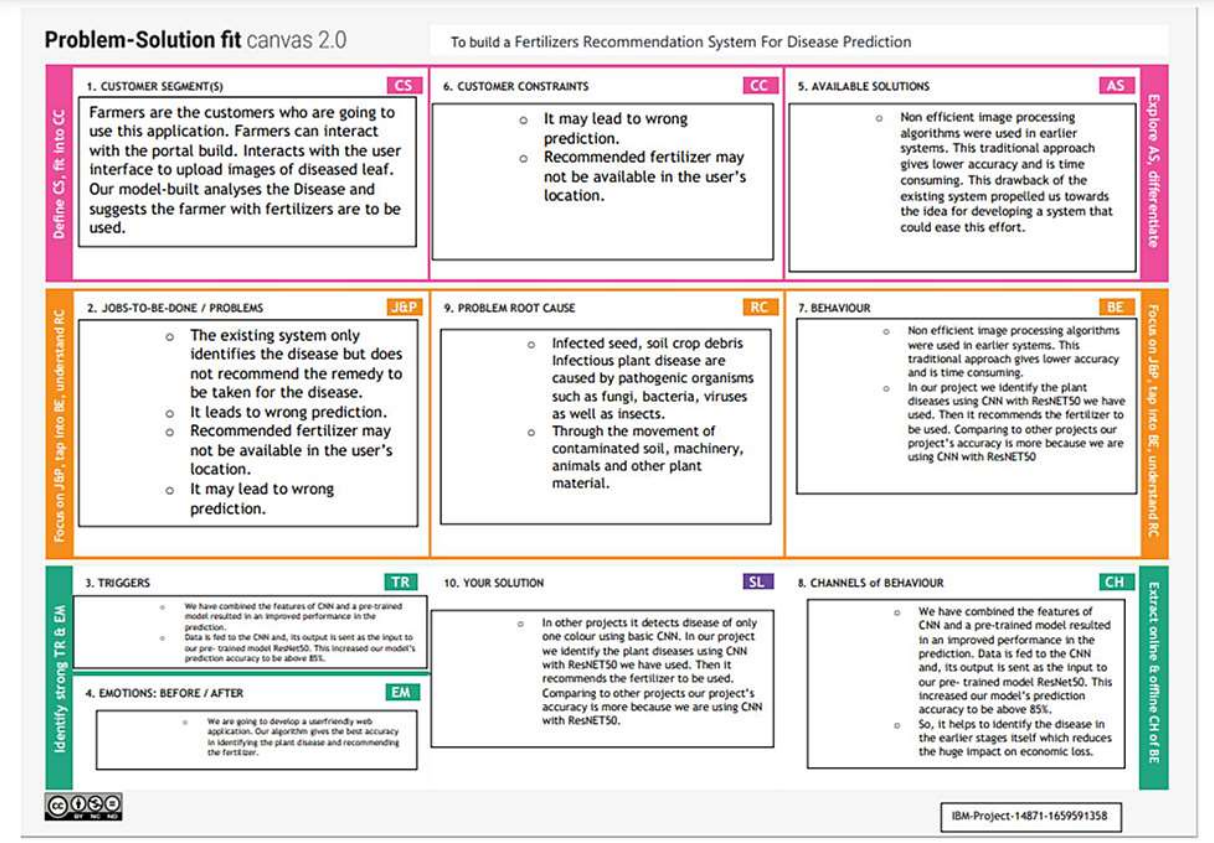
3.3 PROPOSED SOLUTION:

S/NO	Parameter	Description
1	Problem Statement (Problem to be solved)	In Agriculture, Precautions has to be taken to save the plants from disease. Hence this Application is used to predict the disease in plants and crops grown by farmers and Gardeners & provide a way to cure and nurture it.
2	Idea / Solution description	The farmer and Gardener can upload the picture of the infected plant or crop and this system will predict the disease and provide remedies along with the fertilizer recommendation.
3	Novelty / Uniqueness	This application recommends the best fertilizer for the plant and they can consult with the agricultural experts. They can also report the improvement of the plants after using the fertilizer
4	Social Impact / Customer Satisfaction	Helps farmers to get good yield out of the crop. People will get good quality products. This application will satisfy the farmer & Gardener needs. It will be more interactive.
5	Business Model (Revenue Model)	This Application Can be given to farmers in subscription bases with some ideas. Recommending fertilizers and places that has, so they can get commission out of it.
6	Scalability of the Solution	This Application Can be given to farmers in subscription bases with some ideas. Recommending fertilizers and places that has, so they can get commission out of it.

An automated system is introduced to identify different diseases on the plants by checking the symptoms shown on the leaves of the plant. Deep learning techniques are used to identify the diseases and suggest the precautions that can be taken for those diseases.

3.4 PROBLEM SOLUTION FIT:

Machine learning, which is one of the applications of artificial intelligence, is being used to implement the proposed system.



4.REQUIREMENT ANALYSIS:

4.1 FUNCTIONAL REQUIREMENTS:

FR/ NO	PARAMETERS	STORY/SUB-TAKS
FR-1	URL	As per the compiler or IDLE generation
FR-2	DASHBOARD	Its primary inention is to provide information at- a-glance, such as KPIs
FR-3	UPLOAD IMAGE	input through the upload link
FR-5	PROCESSING	using CNN,keras,Tensorflow
FR-6	DESCRIBE	Describe inputs, behaviours and outputs. To enables user to accomplish their task

4.2 NON- FUNCTIONAL REQUIREMENTS:

Following are the non-functional requirements of the proposed solution

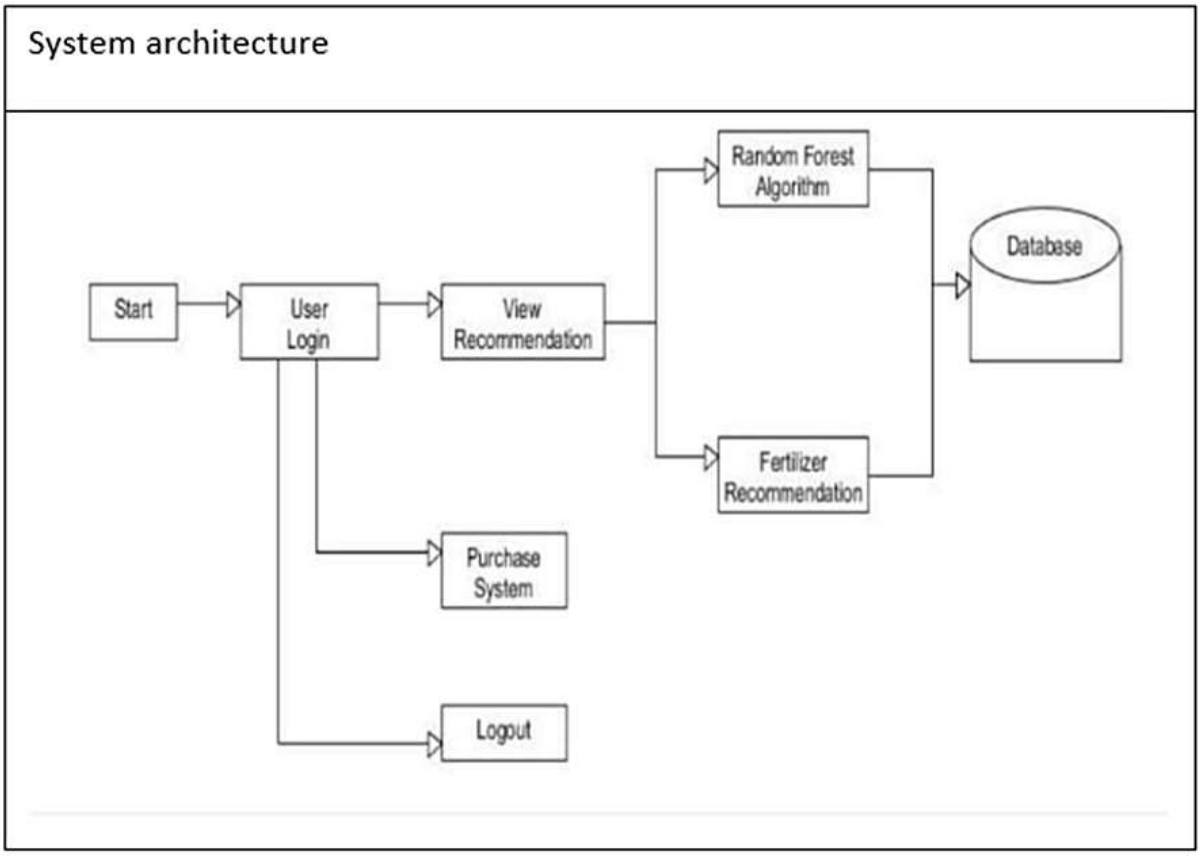
NFR/ N0	PARAMETER	DESCRIPTION
NFR-1	Usability	Clarity, utility, precise, and well formatted
NFR-2	Security	Clarity, utility, precise, and well formatted
NFR-3	Reliability	Avoid anonymous authors
NFR-4	Performance	High speed of response, throughput, execution time, and storage capacity
NFR-5	Availability	High maximum potential uptime and accessibility for content stored on it
NFR-6	Scalabilitymd	Increasing system capacity ,typically through replication, and optimization of system components

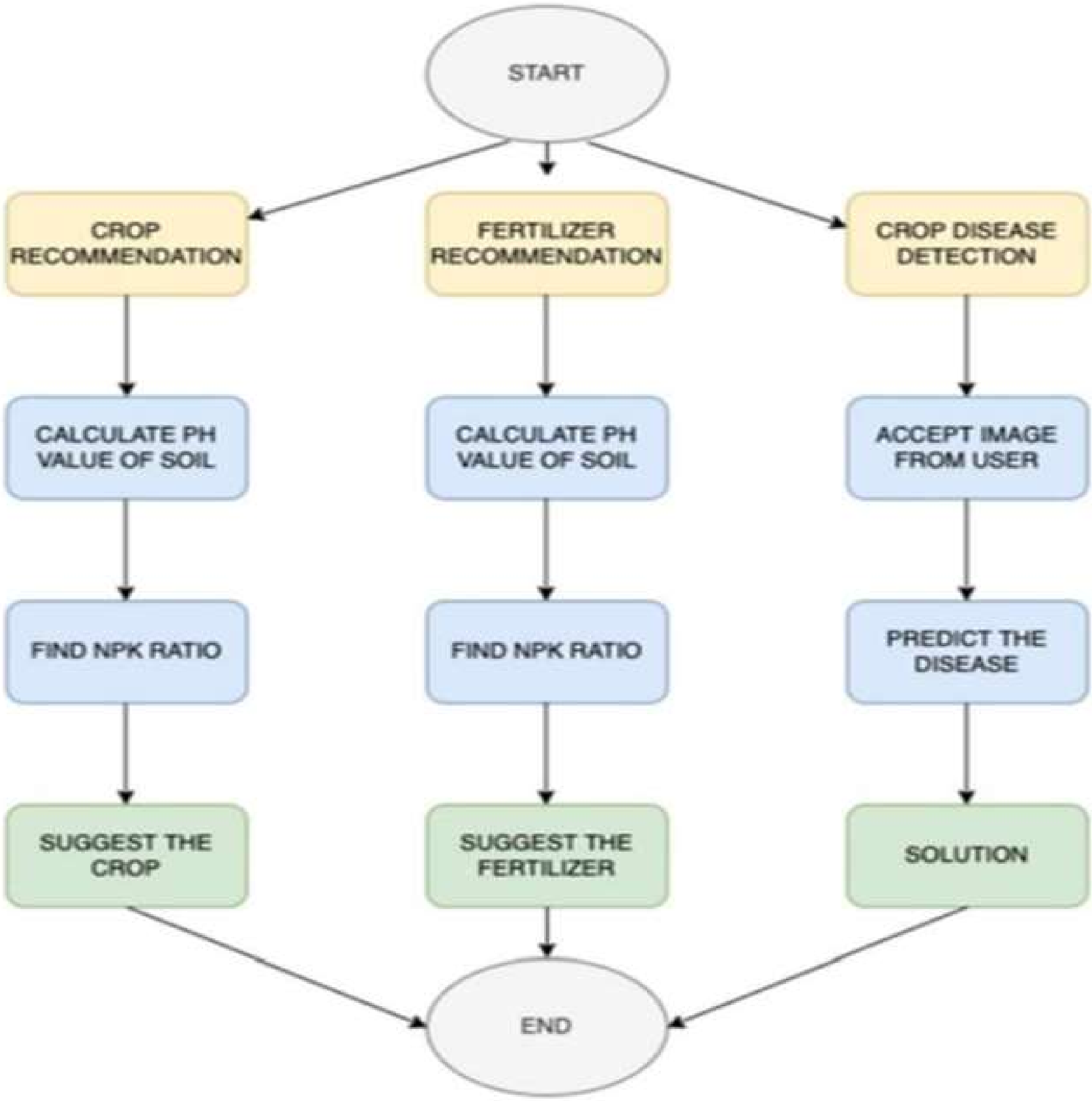
5.PROJECT DESIGN:

5.1 DATA FLOW DIAGRAM:

Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Flow Diagrams:



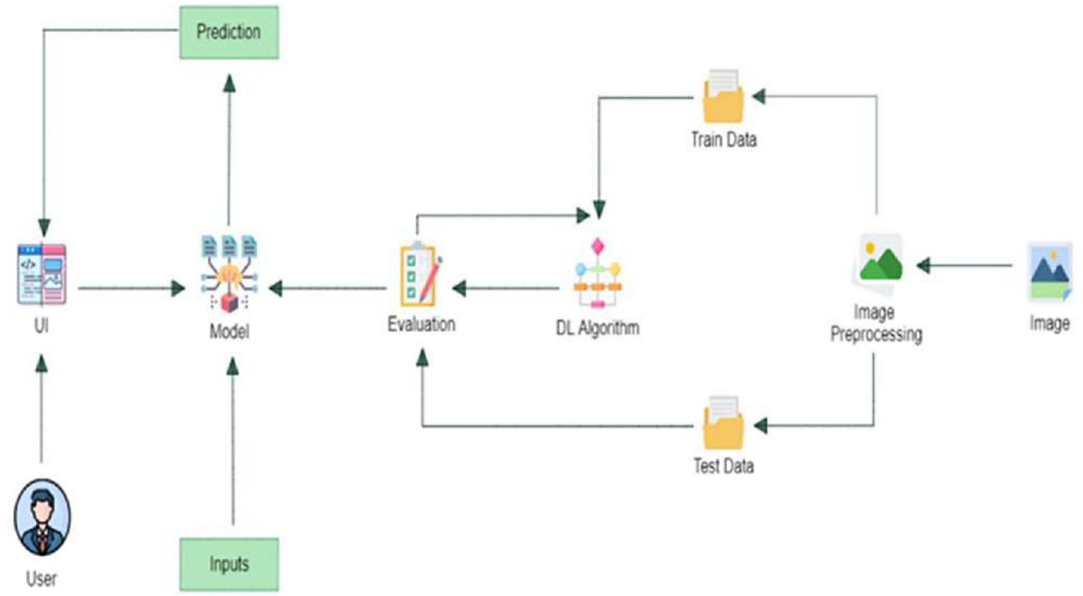

START
CROP RECOMMENDATION
FERTILIZER RECOMMENDATION
CROP DISEASE DETECTION
CALCULATE PH VALUE OF SOIL
CALCULATE PH VALUE OF SOIL
ACCEPT IMAGE FROM USER
FIND NPK RATIO
FIND NPK RATIO
PREDICT THE DISEASE
SUGGEST THE CROP
SUGGEST THE FERTILIZER
SOLUTION
END

5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

In our framework we have proposed procedure that is separated into varies stage as appeared in FIG 5.2.

The five phases are as per the following:

i. Dataset collection
ii. pre-processing(noise removal)
iii. feature extraction
iv. applied various machine learning algorithm
v. recommendation system
vi. recommended crop



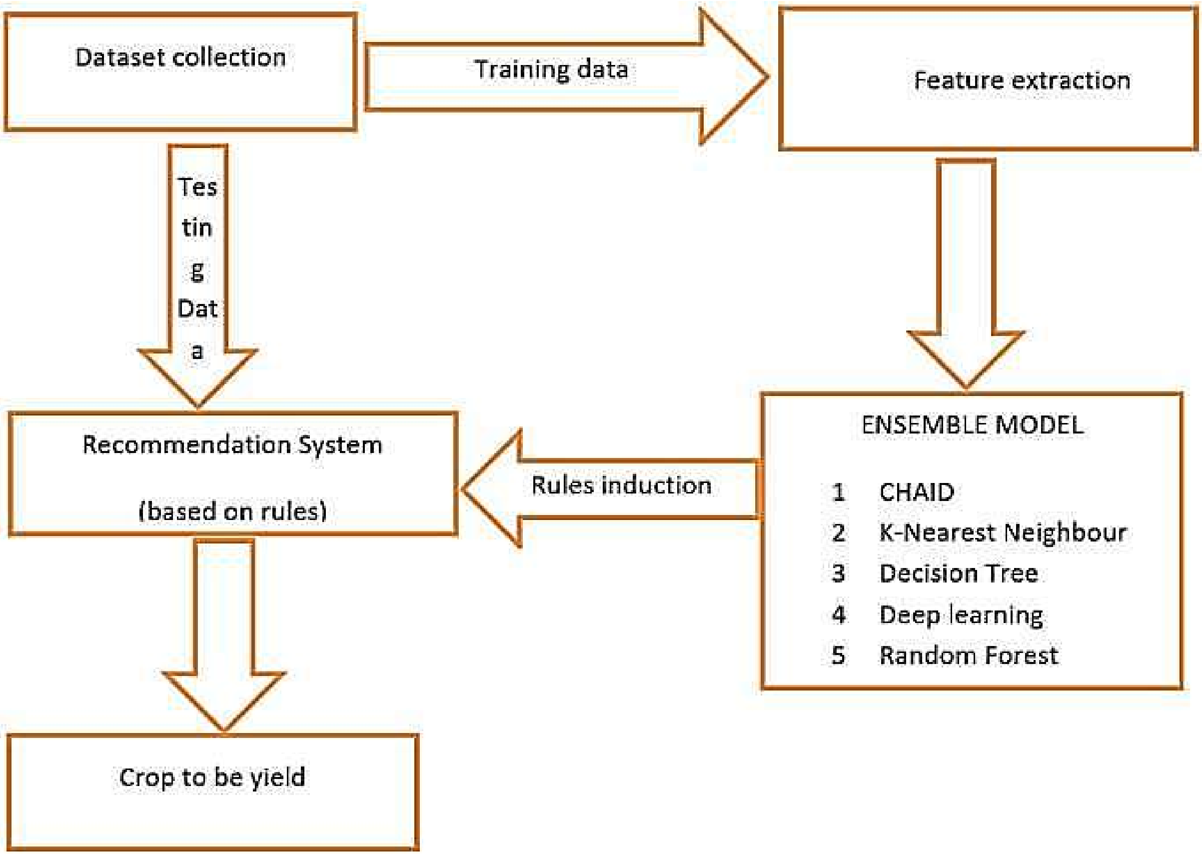


FIG.5.2.TECHNICAL ARCHITECTURE.

5.3 USER STORIES.

Use the below template to list all the user stories for the product.

S/N0	Functional Requirement	User Story Number	User Story Number	Acceptance criteria	Priority	SPRINT
1	URL	USN-1	This is to navigate the user to its user interface		High	Sprint-1
2	Dashboard	USN-2	Its primary inention is to provide information at- a-glance, such as KPIs		High	Sprint-1
3	User input	USN-3	As a user, I can upload the image(input)	I can receive confirmation input through the upload link	High	Sprint-1
4	Handling customer complaints	USN-4	Managing a team of representative offering customer support. Overseeing the customer service process. Resolving the customer complaint brought to your attention. Creating policies and procedures Planning the training and standardization of service delivery		medium	Sprint-2
5	describe	USN-5	Describe inputs, behaviours and outputs. To enables user to accomplish their task	I can understand the system behaviour under specific condition	medium	Sprint-2

6.PROJECT PLANNING &SCHEDULING:

6.1 SPRINT PLANNING &SCHEDULING:

SPRINT	FUNCTIONAL REQUIREMENT(Epic)	USER STORY NUMBER	USER STORY/TASK	STORY POINTS	PRIORITY	TEAM MEMBERS
Sprint-1	Modelling Phase	USN-1	As a customer I can understand the farmers problem. Because country side farmers face many problems such as finding the actual disease is quite difficult.	3	MedIum	DWARAKESWARAN U
Sprint-1		USN-2	Data Collection - Collect the sample images of disease affected leaves of different kind of varieties and unpredictable disease affected leaves.	2	Medium	KALAISELVAN M
Sprint-1		USN-3	Image Preprocessing - Preprocess the collected disease affected images such as rotating to grayscale, calling.	3	Low	B VINAY
Sprint-1		USN-4	Train and test the collected dataset and to measure the accuracy of the dataset.	4	Medium	JAGAN V
Sprint-2		USN-5	Model building - Create a CNN model for the image segmentation	5	High	B VINAY

Sprint-2		USN-6	Cnn model evaluation - Evaluating the cnn model to check the accuracy and precision.	3	High	KALAISELVAN M
Sprint-2	Development Phase	USN-7	SVM algorithm - Use of svm is classifies the images and give 95% accuracy.	5	High	DWARAKESWARAN U
Sprint-2		USN-8	Database creation for each dataset classes.	3	Medium	JAGAN V
Sprint-2		USN-9	User database creation for the user details.	2	Low	DWARAKESWARAN U
Sprint-2		USN-10	Description Page - It contains the details of predicting criteria and user guides.	3	Medium	KALAISELVAN M
Sprint-3		USN-11	Login Page - Login the user with phone number and email id.	2	Low	B VINAY
Sprint-3		USN-12	IAM - Access via OTP or SSH key protection.	3	Medium	JAGAN V
Sprint-3		USN-13	Dashboard and Input page creation - Contains user profiles and predicting accuracy. Input page we can able to feed the input images	2	Low	B VINAY
Sprint-3		USN-14	Prediction page - Show the prediction based on the user input	2	Low	JAGAN V
Sprint-4		USN-15	Model Load – API creation using flask	4	Medium	DWARAKESWARAN U
Sprint-4	Deployment Phase	USN-16	Connecting User interface and backend API calls	5	High	JAGAN V

Sprint-4		USN-17	Deploy the application using IBM cloud	5	High	DWARAKE SWARAN U
Sprint-4	Testing Phase	USN-18	Test the application function to be working with high accuracy and low latency with reliable.	5	High	KALAISELV AN M
Sprint-4	Testing Phase	USN-19	Testing the application as a user all user interfaces will be working properly with check the prediction accuracy	5	High	B VINAY

6.2 <u>SPRINT DELIVERY SCHEDULE</u>

SPRINT	TOTAL STORY POINTS	DURATION	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	12	6 Days	24 Oct 2022	29 Oct 2022	12	28 Oct 2022
Sprint-2	21	4 Days	30 Oct 2022	02 Nov 2022	21	01 Nov 2022
Sprint-3	09	5 Days	03 Nov 2022	07 Nov 2022	09	07 Nov 2022
Sprint-4	24	5 Days	08 Nov 2022	12 Nov 2022	24	10 Nov 2022

6.3 <u>REPORT FROM JIRA</u>

VELOCITY:

Sprint 1 average velocity:

Average Velocity = 12 / 6 = 2

Sprint 2 average velocity:

Average Velocity = 21 / 4 = 5.2

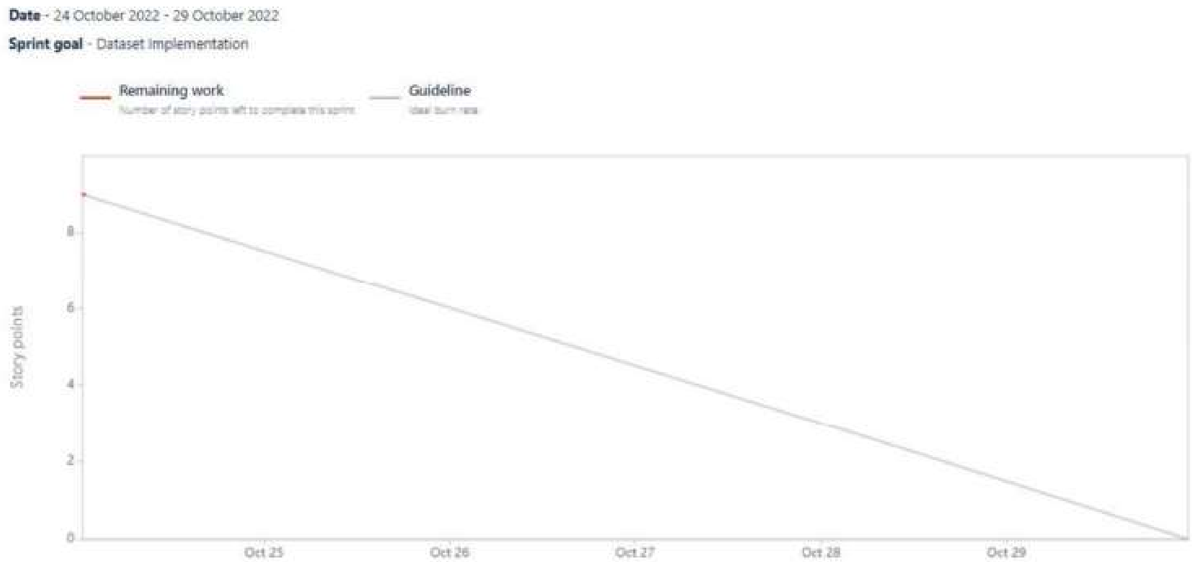
Sprint 3 average velocity:

Average Velocity = 09 / 5 = 1.8

Sprint 4 average velocity:

Average Velocity = 24 / 5 = 4.8

Burndown Chart:



7.CODING&SOLUTION

7.1 FEATURE 1

BACKEND:

- ✓ Creating the models for fruits and vegtable leaves.
- ✓ save each model in(.h5) formate.
- ✓ all the models are integrated using app.py file

7.2 FEATURE 2

FRONTEND:

There are 10 HTML files

- ✓ index.html is the main file
- ✓ pg-2 shows the prediction and fertilizer recommendation
- ✓ app.py is used to connect the frontend and backend files

DATASET:

FRUIT:



VEGETABLE:

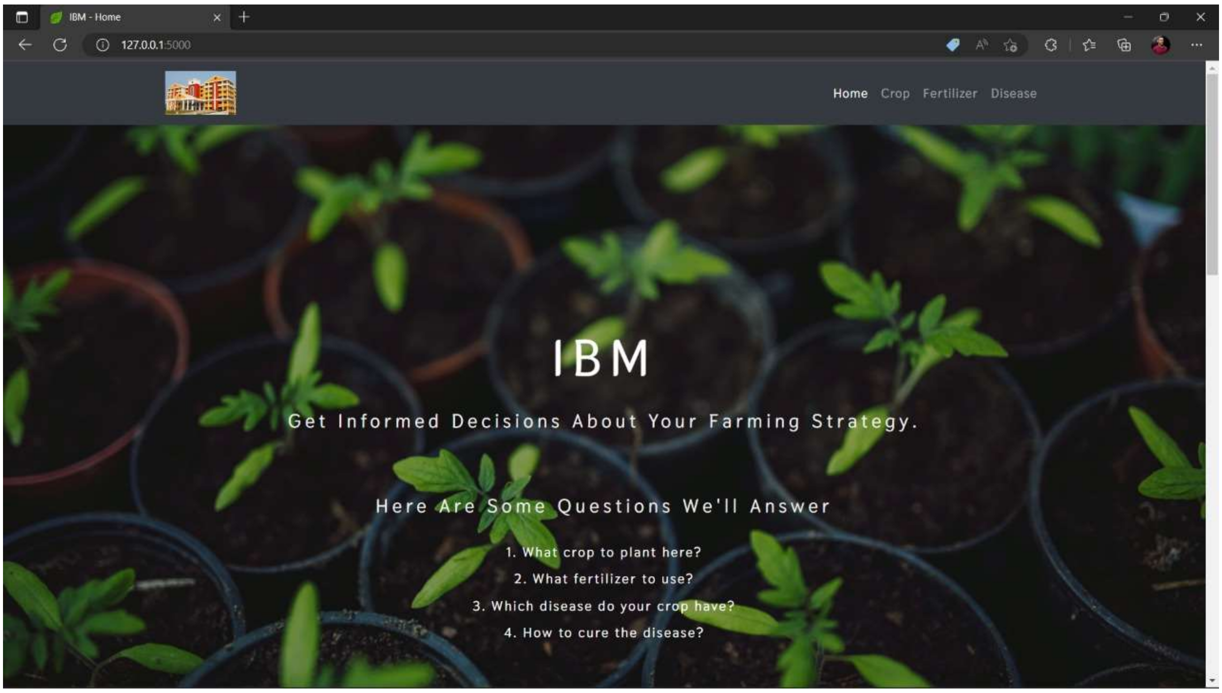


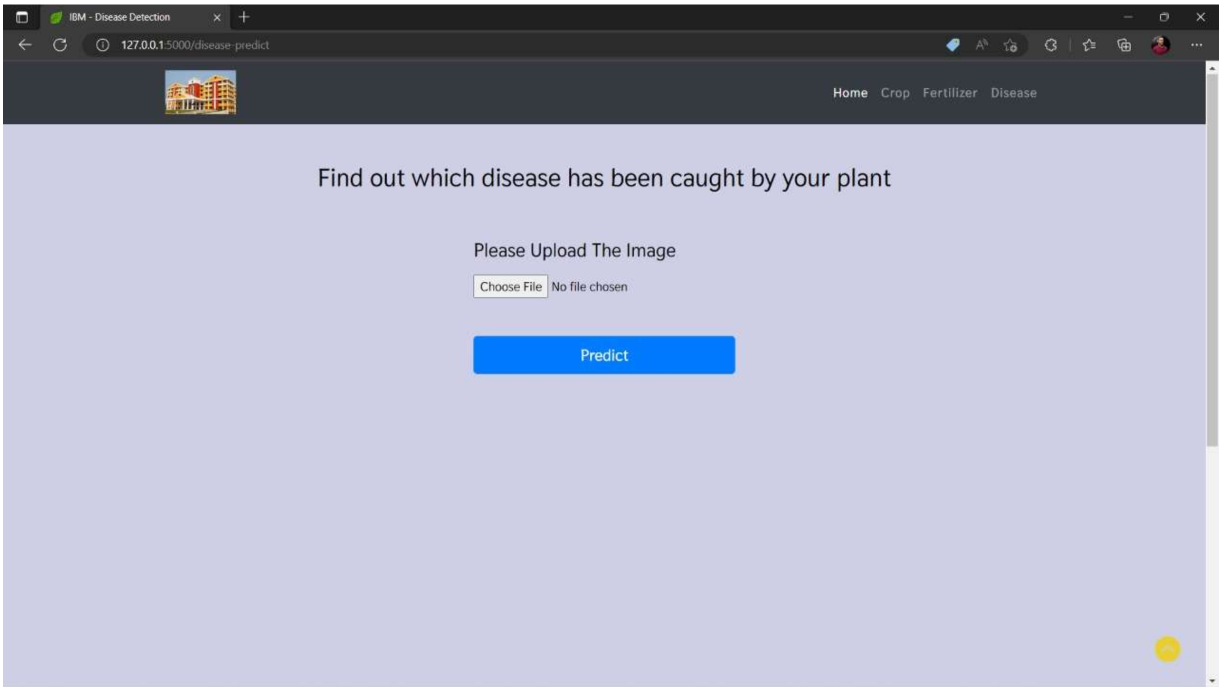
GDrive link:

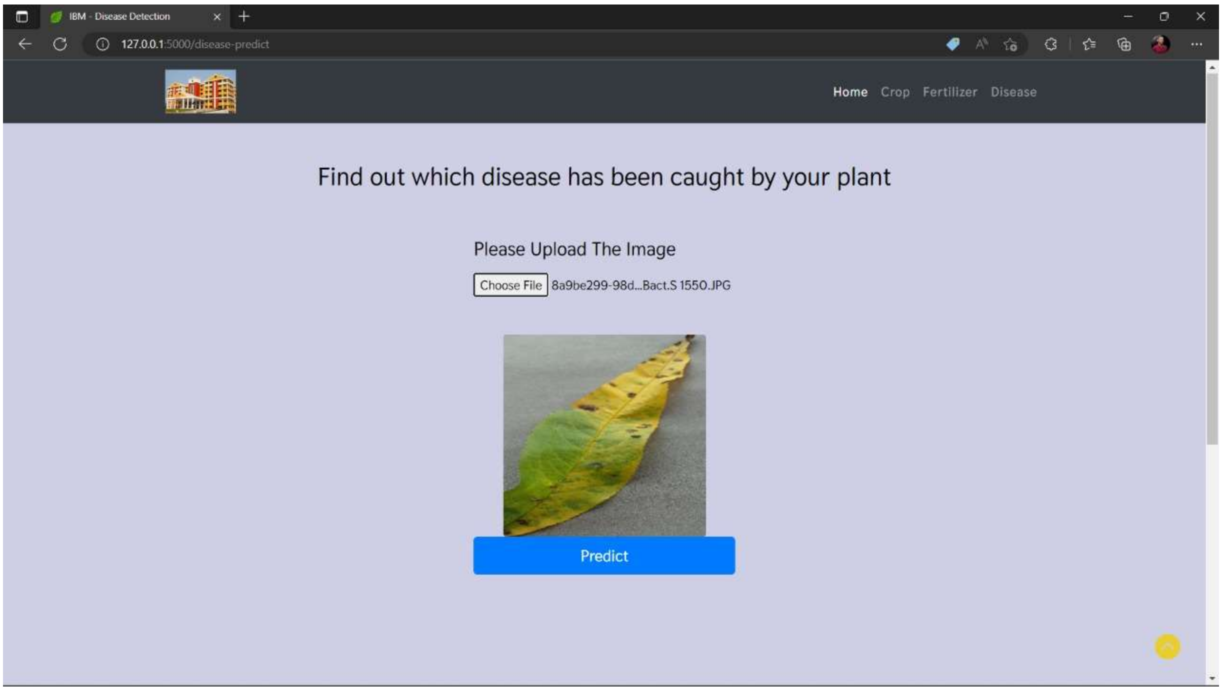
https://drive.google.com/file/d/1fxs7ptI6zh7NTbCOZARKZ7AmYKjnprrY/view?usp=sharing

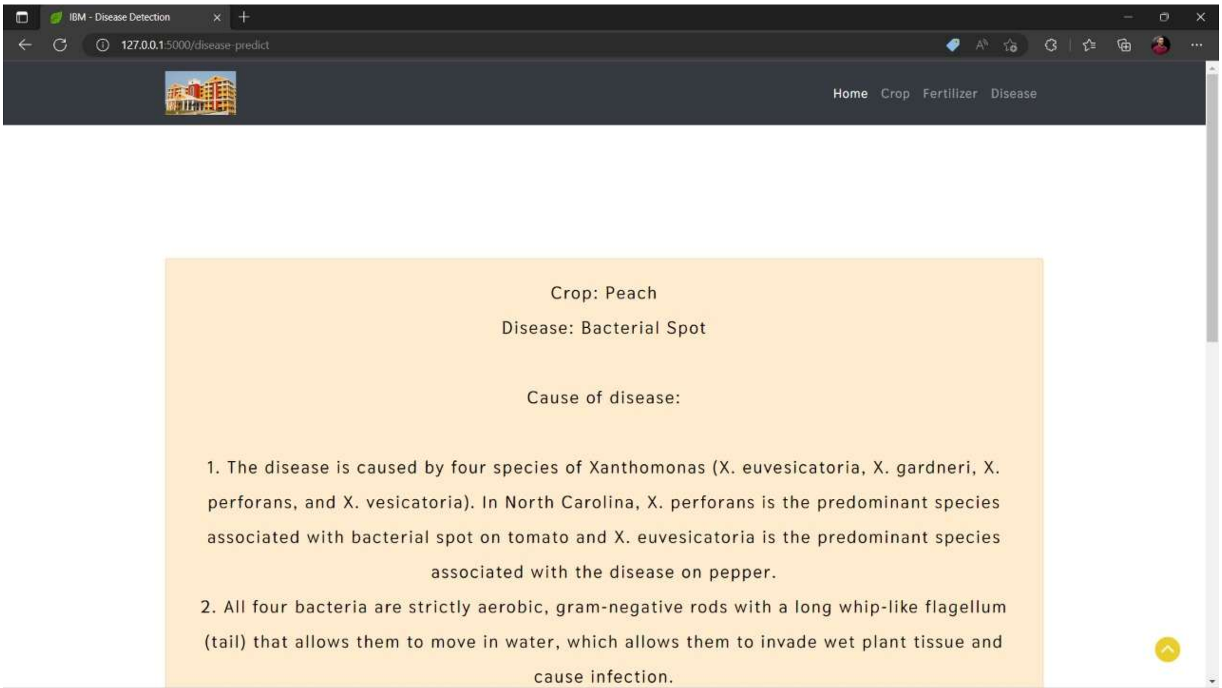
8.TESTING:

This help us to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).






IBM - Disease Detection
127.0.0.1:5000/disease-predict
Home
Crop
Fertilizer
Disease
Find out which disease has been caught by your plant
Please Upload The Image
Choose File
8a9be299-98d...Bact.S 1550.JPG
Predict


IBM - Disease Detection
127.0.0.1:5000/disease-predict
Home
Crop
Fertilizer
Disease
Crop: Peach
Disease: Bacterial Spot
Cause of disease:
1. The disease is caused by four species of Xanthomonas (X. euvesicatoria, X. gardneri, X. perforans, and X. vesicatoria). In North Carolina, X. perforans is the predominant species associated with bacterial spot on tomato and X. euvesicatoria is the predominant species associated with the disease on pepper.
2. All four bacteria are strictly aerobic, gram-negative rods with a long whip-like flagellum (tail) that allows them to move in water, which allows them to invade wet plant tissue and cause infection.

8.1 TESTCASE

DEFECT ANALYSIS

RESOLUTION	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

8.2TEST CASE ANALYSIS:

This report shows the number of test cases that have passed, failed, and untested

SECTION	TOTAL CASES	NOT TESTED	FAIL	PASS
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2 USER ACCEPTNCE

Different types of image datasets are provided, and then those are used to identify the affected area in leaves. Then different types of image-processing techniques are applied to them. The process those images, to get different and useful features needed for the purpose of analyzing later-Plant leaf disease identification is especially needed to predict both the quality and quantity of the First segmentation step primarily based on a mild polygonal leaf model is first achieved and later used to guide the evolution of an energetic contour.

Combining global shape descriptors given by the polygonal model with local curvature based features, the leaves are then classified overleaf datasets. In this research work introduce a method designed to deal with the obstacles raised by such complex images, for simple and plant leaves. A first segmentation step based on graph-cut approach is first performed and later used to guide the evolution of leaf boundaries, and implement classification algorithm to classify the diseases and recommend the fertilizers to affected leaves.

9.RESULT

9.1 PERFORMANCE METRICS

The accuracy for the fruit leaf disease prediction is upto 95.91% and accuracy for the vegetable leaf disease prediction is upto 93.74% .Also,it will recommends the suitable fertilizers for the disease.

FRUIT:

```
Epoch 10/10
225/225 [==============================] - 619s 3s/step - loss: 0.0852 - accuracy: 0.9708 - val_loss: 0.1422 - val_accuracy: 0.9591
```

VEGETABLE:

```
Epoch 10/10
475/475 [==============================] - 441s 928ms/step - loss: 0.1693 - accuracy: 0.9407 - val_loss: 0.1717 - val_accuracy: 0.9374
```

10.ADVANTAGES &DISADVANTAGES

ADVANTAGES:

The proposed model here produces very high accuracy of classification
Very large dataset can also be trained and tested
Images of very high can be resized within the proposed itself.

DISADVANTAGES:

For training and testing the proposed model require very high computational time.
The neural network architecture used in this project work has high complexity

11.CONCLUSION

Deep learning methods are used to train the model, which aids in making appropriate disease decisions. To contain infected diseases, the farmer is advised to use pesticides as a cure.

In the future, the proposed scheme could be expanded to provide additional facilities such as nearby government markets, pesticide price lists, and a nearby open market, among others.

This paper presents a review of various disease classification strategies for plant disease detection and providing a solution by recommending fertilizers for the diseases.

Different approaches and models of Deep Learning methods were explored and used in this project so that it can detect and classify plant diseases correctly through image processing of leaves of the plants.

The procedure starts from collecting the images used for training, testing and validation to image preprocessing and augmentation and finally comparison of different pretrained models over their accuracy. Finally, at the end , our model detects and distinguishes between a healthy plant and different diseases and provides suitable remedies so as to cure the disease.

This paper proposed and developed a system which uses plant leaf images to detect different types of disease in fruit and vegetable,it provides appropriate fertilizer suggestions.

12.FUTURE SCOPE

The prediction of crop yield based on location and proper implementation of algorithms have proved that the higher crop yield can be achieved.

From above work we conclude that for soil classification Random Forest is good with accuracy 86.35% compare to Support Vector Machine. For crop yield prediction Support Vector Machine is good with accuracy 99.47% compare to Random Forest algorithm.

The work can be extended further to add following functionality. Mobile application can be build to help farmers by uploading image of farms. Crop diseases detection using image processing in which user get pesticides based on disease images. Implement Smart Irrigation System for farms to get higher yield.

The system successfully interprets various Diseases and is also capable of providing fertilizers suggestion for the respective disease. Further more, this system can be made more robust by incorporating more image dataset with wider variations like more than one leaf in a single image.

An App could also be developed for the project which could make the work of the farmers easier.They could directly upload image on the app and it would tell the disease and the cure then and there.This would reduce the time and efforts.

13.APPENDIX

13.1 SOURCE CODE

```
from flask import Flask, render_template, request, Markup
import numpy as np
import pandas as pd
from utils.disease import disease_dic
from utils.fertilizer import fertilizer_dic
import requests
import config
import pickle
import io
import torch
from torchvision import transforms
from PIL import Image
from utils.model import ResNet9
disease_classes = ['Apple___Apple_scab',
                   'Apple___Black_rot',
                   'Apple___Cedar_apple_rust',
                   'Apple___healthy',
                   'Blueberry___healthy',
                   'Cherry_(including_sour)___Powdery_mildew',
                   'Cherry_(including_sour)___healthy',
                   'Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot',
                   'Corn_(maize)___Common_rust_',
                   'Corn_(maize)___Northern_Leaf_Blight',
                   'Corn_(maize)___healthy',
                   'Grape___Black_rot',
                   'Grape___Esca_(Black_Measles)',
```

```
                'Grape___Leaf_blight_(Isariopsis_Leaf_Spot)',
                'Grape___healthy',
                'Orange___Haunglongbing_(Citrus_greening)',
                'Peach___Bacterial_spot',
                'Peach___healthy',
                'Pepper,_bell___Bacterial_spot',
                'Pepper,_bell___healthy',
                'Potato___Early_blight',
                'Potato___Late_blight',
                'Potato___healthy',
                'Raspberry___healthy',
                'Soybean___healthy',
                'Squash___Powdery_mildew',
                'Strawberry___Leaf_scorch',
                'Strawberry___healthy',
                'Tomato___Bacterial_spot',
                'Tomato___Early_blight',
                'Tomato___Late_blight',
                'Tomato___Leaf_Mold',
                'Tomato___Septoria_leaf_spot',
                'Tomato___Spider_mites Two-spotted_spider_mite',
                'Tomato___Target_Spot',
                'Tomato___Tomato_Yellow_Leaf_Curl_Virus',
                'Tomato___Tomato_mosaic_virus',
                'Tomato___healthy']
disease_model_path = 'models/plant_disease_model.pth'
disease_model = ResNet9(3, len(disease_classes))
disease_model.load_state_dict(torch.load(
    disease_model_path, map_location=torch.device('cpu')))
```

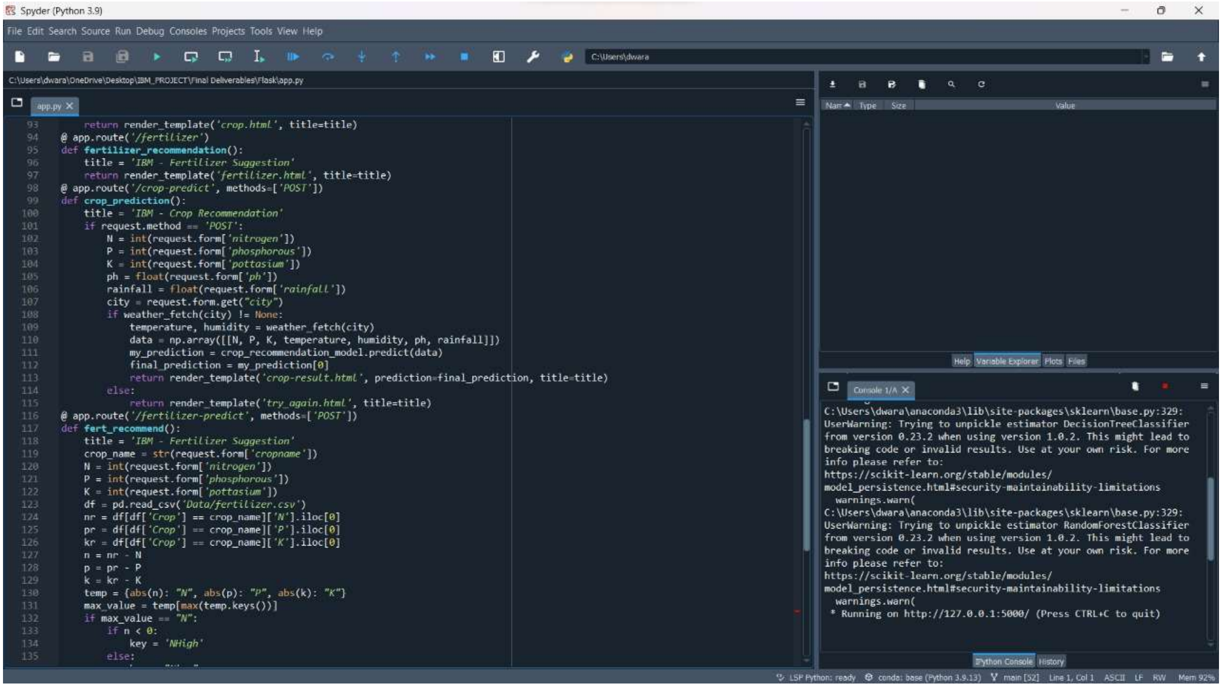
```
disease_model.eval()
crop_recommendation_model_path = 'models/RandomForest.pkl'
crop_recommendation_model = pickle.load(
_  open(crop_recommendation_model_path, 'rb'))
def weather_fetch(city_name):
    api_key = config.weather_api_key
    base_url = "http://api.openweathermap.org/data/2.5/weather?"
    complete_url = base_url + "appid=" + api_key + "&q=" + city_name
    response = requests.get(complete_url)
    x = response.json()
    if x["cod"] != "404":
        y = x["main"]
        temperature = round((y["temp"] - 273.15), 2)
        humidity = y["humidity"]
        return temperature, humidity
    else:
        return None
def predict_image(img, model=disease_model):
    transform = transforms.Compose([
        transforms.Resize(256),
        transforms.ToTensor(),
    ])
    image = Image.open(io.BytesIO(img))
    img_t = transform(image)
    img_u = torch.unsqueeze(img_t, 0)
    yb = model(img_u)
    _, preds = torch.max(yb, dim=1)
    prediction = disease_classes[preds[0].item()]
    return prediction
```

```
app = Flask(__name__)
@ app.route('/')
def home():
    title = 'IBM - Home'
    return render_template('index.html', title=title)
@ app.route('/crop-recommend')
def crop_recommend():
    title = 'IBM - Crop Recommendation'
    return render_template('crop.html', title=title)
@ app.route('/fertilizer')
def fertilizer_recommendation():
    title = 'IBM - Fertilizer Suggestion'
    return render_template('fertilizer.html', title=title)
@ app.route('/crop-predict', methods=['POST'])
def crop_prediction():
    title = 'IBM - Crop Recommendation'
    if request.method == 'POST':
        N = int(request.form['nitrogen'])
        P = int(request.form['phosphorous'])
        K = int(request.form['pottasium'])
        ph = float(request.form['ph'])
        rainfall = float(request.form['rainfall'])
        city = request.form.get("city")
        if weather_fetch(city) != None:
            temperature, humidity = weather_fetch(city)
            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, title=title)
```

```
        else:
            return render_template('try_again.html', title=title)
@ app.route('/fertilizer-predict', methods=['POST'])
def fert_recommend():
    title = 'IBM - Fertilizer Suggestion'
__crop_name = str(request.form['cropname'])
    N = int(request.form['nitrogen'])
    P = int(request.form['phosphorous'])
    K = int(request.form['pottasium'])
    df = pd.read_csv('Data/fertilizer.csv')
    nr = df[df['Crop'] == crop_name]['N'].iloc[0]
    pr = df[df['Crop'] == crop_name]['P'].iloc[0]
    kr = df[df['Crop'] == crop_name]['K'].iloc[0]
    n = nr - N
    p = pr - P
    k = kr - K
    temp = {abs(n): "N", abs(p): "P", abs(k): "K"}
    max_value = temp[max(temp.keys())]
    if max_value == "N":
        if n < 0:
            key = 'NHigh'
        else:
            key = "Nlow"
    elif max_value == "P":
        if p < 0:
            key = 'PHigh'
        else:
            key = "Plow"
    else:
```

```
        if k < 0:
            key = 'KHigh'
        else:
            key = "Klow"
    response = Markup(str(fertilizer_dic[key]))
    return render_template('fertilizer-result.html', recommendation=response, title=title)
@app.route('/disease-predict', methods=['GET', 'POST'])
def disease_prediction():
    title = 'IBM - Disease Detection'
    if request.method == 'POST':
        if 'file' not in request.files:
            return redirect(request.url)
        file = request.files.get('file')
        if not file:
            return render_template('disease.html', title=title)
        try:
            img = file.read()
            prediction = predict_image(img)
            prediction = Markup(str(disease_dic[prediction]))
            return render_template('disease-result.html', prediction=prediction, title=title)
        except:
            pass
    return render_template('disease.html', title=title)
if __name__ == '__main__':
    app.run(debug=False)
```

<u>13.2 SCREENSHOT</u>



GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-14871-1659591358

YOUTUBE LINK : https://www.youtube.com/embed/JXq8l4zWS_M