PROJECT REPORT

FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

IBM-PROJECT-ID : IBM-PROJECT-23753-1659927643

TEAM ID : PNT2022TMID10715

TEAM MEMBERS : 4 MEMBER

TEAM LEAD : UMASANKARI M

(42111910210107)

M1 LEAD : VIDHYA S

(42111910210111)

M2 LEAD : VINITHINI M

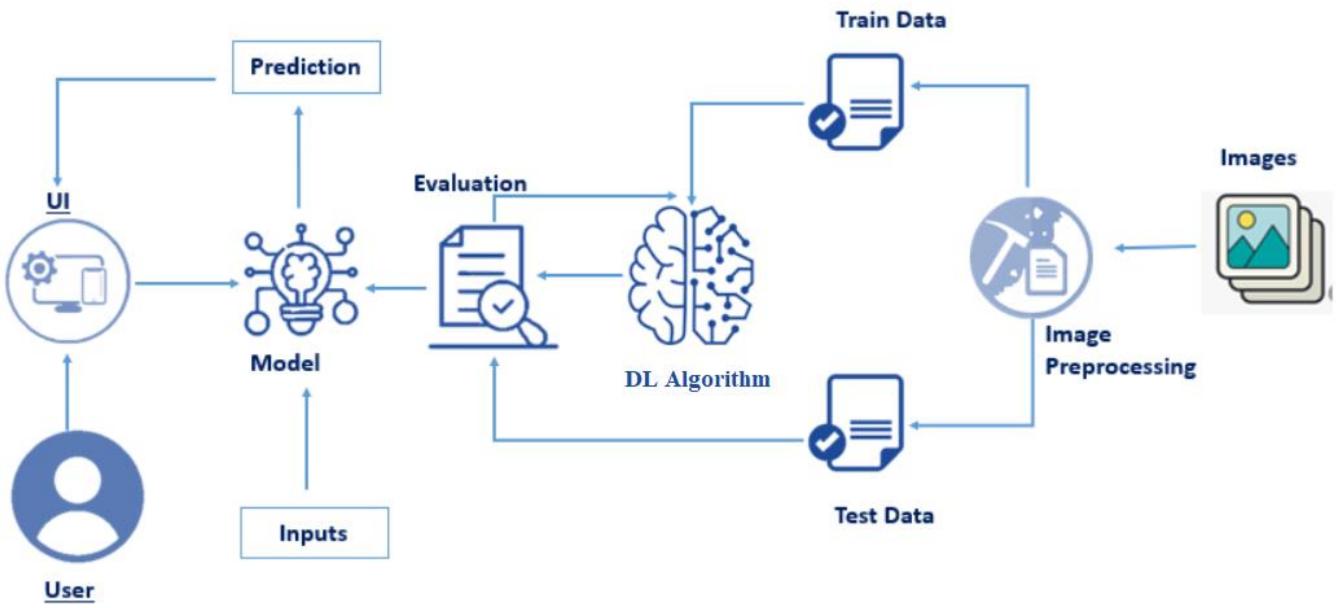
(42111910210116)

M3 LEAD : YUVASRI S

(42111910210120)

1. INTRODUCTION

Agriculture is the most important sector in today's life. Most plants are affected by a wide variety of bacterial and fungal diseases. Diseases on plants placed a major constraint on the production and a major threat to food security. Hence, early and accurate identification of plant diseases is essential to ensure high quantity and best quality. In recent years, the number of diseases on plants and the degree of harm caused has increased due to the variation in pathogen varieties, changes in cultivation methods, and inadequate plant protection techniques. An automated system is introduced to identify different diseases on 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.



1.1 PROJECT OVERVIEW

Overview This project collects two datasets, namely the vegetable dataset and the fruit dataset. These datasets are then trained with deep learning neural networks called Convolutional Neural Networks (CNN) and tested. The first step is to train the fruit dataset and then test it with CNN. There are 6 classes, and all classes have been trained and tested. The second step is to train and test the vegetable dataset. Python is the software that is used to train and test datasets. All Python code is first written in Anaconda Python's Jupyter notebook. Then the code is tested in IBM cloud. Finaly, a web-based framework has been created with Flask, a Python library. The templates folder contains 2 html files. They are also associated with static folder. Spyder-Anaconda Python is used to interface with these webpages. It has been tested.

1.2 PURPOSE

This project is used for testing the vegetables and fruits samples to identify diseases. This project also recommends fertilizers to prevent prediced diseases.

2. LITERATURE SURVEY

[1] Totally 54% of India's land area is deemed arable, making it the world's largest agrarian economy. Soil infertility owing to over fertilization, as well as a lack of access and awareness of contemporary agricultural practices, are the different factors that contribute to low agricultural production. The main purpose of this research work is to develop a machine learning-based recommendation system to increase agricultural productivity. A variety of datasets were used in this study to design and develop advanced models to estimate the crop, recommend fertiliser, and identify plant disease. An algorithm called MobileNet uses an image of a leaf to identify the disease present in a plant. The XGBoost model predicts a suitable crop based on the local soil nutrients and rainfall. Random Forest [RF] model was used to propose fertilizer and develop ideas for improving soil fertility depending on nutrients present in the soil. When compared to other approaches, the proposed model delivers a high level of accuracy.

[2] Detection and recognition of plant diseases using machine learning are very efficient in providing symptoms of identifying diseases at its earliest. Plant pathologists can analyze the digital images using digital image processing for diagnosis of plant diseases. Application of computer vision and image processing strategies simply assist farmers in all of the regions of agriculture. Generally, plant diseases are caused by the abnormal physiological functionalities of plants. Therefore, the characteristic symptoms are generated based on the differentiation between normal physiological functionalities and abnormal physiological functionalities of the plants. Mostly, the plant leaf diseases are caused by Pathogens which are positioned on the stems of the plants. These different symptoms and diseases of leaves are predicted by different methods in image processing. These different methods include different fundamental processes like segmentation, feature extraction and classification and so on. Mostly, the prediction and diagnosis of leaf diseases are dependent on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves.

[3] This method used datasets to find diseased and healthy plant leaves. we introduced a deep convolutional neural network to identify crop series and diseases that may not be present in the plant tissue. The model trained on the test set has an accuracy of 99.35%. This process is enabled by deep learning, machine learning and digital epidemiologyA neural network associate’s images of diseased plants and crops as a pair. A neural network node is a mathematical function that receives numerical inputs from input edges and provides numerical outputs as output edges. We analyze 54,306 images of plant leaves that have been assigned a variance of 38 class labels. We resize the images to 256x256 pixels and perform both model optimization and prediction on these reduced images.

Advantages: this system identifies the diseases that may not be present in the plant issue.

[4] Data mining is a rising studies area in crop yield analyis Yield predicition is a complete essential problem in agriculture. Any farmer is interested in knowing how muchyield he is about to expect also, it will end-user-helpful to farmers for indicting which fertilizers to be used as well as knowing the crop diseases all at one place. The project comes with a model to be precise and accurate in predicting crop, fertilizers, Crop disease and deliver the end-user with proper recommendations about the required fertilizer ratio based on atmospheric and soil parameters of the land which enhance to increase the crop yield and increase farmer revenue.

[5] In this paper, the disease classification was initially performed by the International Center for Tropical Agriculture (CIAT) with banana images as input, which was transferred to the primary processing technique. A hybrid segmentation called the generalized variation fuzzy mean sum (TGVFCMS) was used to segment the affected leaf area. After segmentation, the data is passed to CNN for final review classification. It has a database of more than 18,000 real photos of bananas in the CIAT image gallery. The dataset includes dry/aged leaves (DOL), HP and 700 balanced images of 5 major diseases such as banana Fusarium wilt of Banana (FWB), Black Sigatoka (BS), Xanthoma wilt or bacterial banana wilt (BBW) , Yellow Sigatoka (YS) and banana pustulosis.

[6] This proposed system explains about the water needs of plants vary from place to place due to changes in soil content, texture, climatic factors, and more. In addition to water requirements, plant diseases can also cause plants not to grow properly. In this article, we proposed a new intelligent irrigation system that can automatically control irrigation using an Android mobile application. In addition, photos of plant leaves are captured and sent to the cloud server. This is further processed and compared with images of diseased plant leaves in the cloud database. Based on the comparison, a list of suspected plant diseases is displayed to the user via an Android mobile application.

[7] India is an agrarian nation. But creating a profitable yield for the farmer in each crop cycle is becoming a major challenge on various factors. Picking the reasonable fertilizer for the land and yield is an important and basic part of agriculture. Deciding the supplement levels in soil utilizing lab hardware can be restrictively costly, particularly in developing nations. The current frameworks on deciding soil nutrient substance and proposal for fertilizer isn't sufficiently proficient efficient enough. This paper introduces a compelling technique for estimation of nutrient dimension in soil and suggestion for appropriate fertilizer. The proposed methodologies comprise of four stages: soil analysis, data pre-processing, data analysis and Recommendation. The soil sample is analyzed using an IoT based device utilizing NPK sensor with two electrodes are set to calculate collect the NPK ratio of the soil nutrient and for pre-processing, the data gathered from sensors are figured into correct dataset and machine learning algorithm is utilized to recognize the reasonable fertilizer. This venture is extremely valuable to farmer to pick the right fertilizer toward the start of product cycle and amplify the yield.

[8] This paper presents a methodology for classifying three major leaf diseases of banana using local textural characteristics. Disease-affected regions are identified using image enhancement and color segmentation. The segmented image is transformed into one transform domain using three Image transforms (DWT, DTCWT, and ranklet transforms). Feature vectors are extracted from transform-domain images using LBP and its variants (ELBP, MeanELBP, and MedianELBP). Experimental results showed the best classification performance of ELBP features extracted from the DTCWT domain (accuracy 95.4%, accuracy 93.2%, sensitivity 93.0%, Fscore 93.0%, and specificity 96.4%).

2.1 EXISTING PROBLEM

In most of the existing methods, the process of finding the soil type, identifying the leaf disease and preferring the fertilizer were all carried out manually. The method was prone to various disadvantages. Even when the framework was digitalized, it has certain problems as, predicting a diverse fertilizer for a soil type, certain files regarding the leaf disease or soil type or fertilizer International Journal of Engineering Trends and Applications may not be updated. In other situation the system may not provide the needed support. Hence in order to overcome some of these issues, the authors proposed a new approach.

2.2 REFERENCES

[1] D.N.V.S.L.S indra, M.Sobhana, A.H.L.Swaroop, V.PhaniKumar. "A Machine Learning Based New Recommendation System To The Farmer".6th International Conference on Intelligent Computing and Control Systems (ICICCS) May 2022

[2] R. Neela; P. Nithya " Fertilizers Recommendation System For Disease Prediction In Tree Leave " 2019

[3] S. Sankaran, A. Mishra, R. Ehsani, and C. Davis, "A review of advanced techniques for detecting plant diseases, " Computers and Electronics in Agriculture. 2010

[4] SanidhyaPurohit, DeepSanghani, NamanSenjaliya, Prof.Anuradha Kapoor " Agro-Farm-Crop,Fertilizer & Disease Prediction" 2022

[5] V. Gokula Krishnan1, J. Deepa, Pinagadi Venkateswara Rao, V. Divya , S. Kaviarasan " An automated segmentation and classification model for banana leaf disease detection ".2022

[6] Ranjith, Saheer Anas, Ibrahim Badhusha, O.T. Zaheema, K.Faseela, Minnuja Shelly " Cloud based automated irrigation and plant leaf disease detection system using an android application" 2017

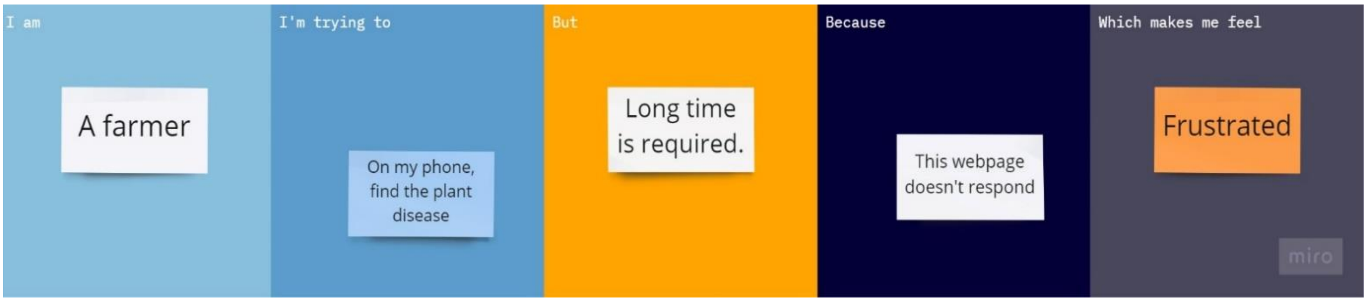
[7] Mayuri Pawar, Geetha Chillarage " Soil Toxicity Prediction and Recommendation System Using Data Mining in Precision Agriculture" 2018

[8] Deepthy Mathew, C. Sathish Kumar, K. Anita Cherian " Foliar fungal disease classification in banana plants using elliptical local binary pattern on multiresolution dual tree complex wavelet transform domain " 2020

2.3 PROBLEM STATEMENT DEFINITION

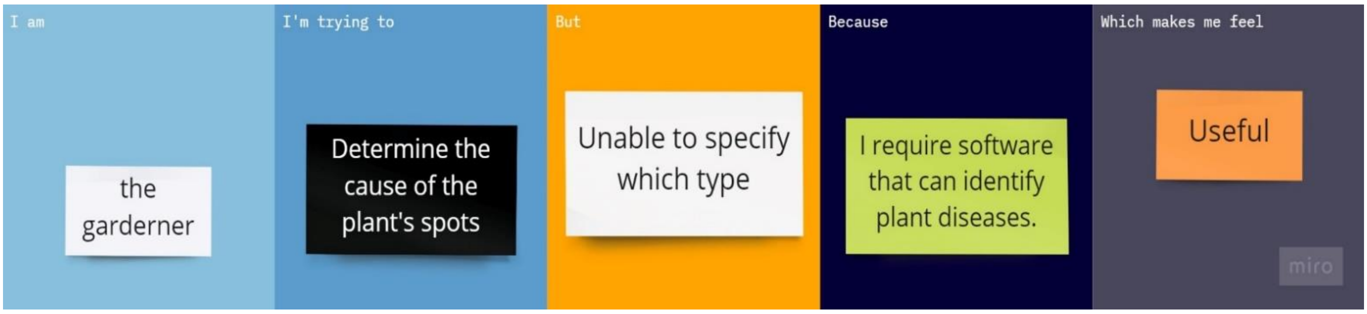
The primary driver of national development is agriculture. Many individuals work in agriculture, which is directly tied to agricultural products. One of the main causes of decreases in the quality and quantity of food crops, particularly on leaves, is plant disease. The growth of the agricultural level is slowed down in agricultural elements if the plant has leaf disease. It is crucial for you to find the leaf illness. Preservation of agriculture After pre-processing with a median filter, guided active contour method segmentation is used to identify the leaf illness.

Problem Statement 1:



Problem Statement 2:

Problem Statement 3:



Problem Statement (PS)	I am (Customer)	I’m trying to	But	Because	Which makes me feel
PS 1	A Farmer	On my phone find the plant disease	Long time is required	Th webpage doesn’t respond	Frustrated
PS 2	Student	Develop healthy plants	Disease always has an impact on it	I’m knowledgeable about it	Fear of fertiliser usage errors
PS 3	The gardener	Determine the cause of the plant’ s spots	Unable to specify which type	I require software that can identify plant diseases	Useful

3. IDEATION & PROPOSED SOLUTION

The proposed approach was organized in such a manner, that it is universal to all the users in the world.

3.1 EMPATHY MAP CANVAS

3.2 PROPOSED SOLUTION

Problem Statement

Plant diseases significantly reduced crop productivity and posed a serious danger to farmers' livelihoods and food security. Thus, to ensure that crops are produced in large quantities and at the highest possible standards, it is crucial to identify plant illnesses early and accurately and to use the appropriate nutrients.

Idea / Solution description

As part of our project, we want to create a web application that will help farmers foresee crop infestations. New technologies like machine learning and deep learning algorithms can forecast it. Along with forecasting, we also intended to advise on the appropriate pesticide to use and the recommended dosage in order to save the crop from degeneration. The CNN algorithm is then used to form a neural network for predicting crop disease after the train and test picture datasets have been preprocessed. To provide the farmers with an interface, a web application built with Flask is developed.

Novelty / Uniqueness

The software advises both organic and inorganic fertilisers, and farmers can place online orders for the fertilisers that are readily accessible

Social Impact / Customer Satisfaction.

The objective of this study is to more accurately predict crop disease early on and stop additional crop loss. In order to apply fertilisers more effectively, the location of the disease's affected area is also discovered. so that Farmers might exercise caution and preventive measures

Business Model (Revenue Model)

helps farmers produce food goods in a good way and cuts down on production losses earlier. Crop yield, crop efficiency, and agricultural product output will all rise with the suggested approach. Agriculture output will expand significantly, and profit will be increased.

Scalability of the Solution

The proposed structure for precision agriculture enables the application of a flexible methodology that may be modified for various types of crops. Based on the taught data, the software will anticipate an accurate answer

3.3 PROBLEM SOLUTION FIT

Customer Segments

Customers who will utilise this application are armers. Farmers can communicate with the portal construction. utilises the user interface to upload pictures of the sick leaf. Our disease analysis algorithm recommends using farmers who have access to fertilisers

Jobs-To-Be-Done / Problems

A disease that affects plants could result in decreased crop production and a decline in the production of agricultural goods.

Triggers

Being informed about plant diseases and taking preventative measures to minimise issues

Emotions: Before / After

Before:

Is there a way to get help?

After:

User-friendly, easy to browse, with available 24/7 support/help choices

Available Solutions

The algorithms utilised in early systems for image processing were ineffective. By conducting research using books, e-books, websites, etc. By gathering data from people and coming to a conclusion.

Customer Constraints

Anxiety:

When the consumer was still unsure of how to utilise the fertiliser, they started to become nervous.

Mysteries:

They might have called it a mystery because they couldn't solve it.

Behaviour

Searches for the top disease prediction and fertiliser advice applications. A situation of this nature arises when the farmer is ignorant of the sickness.

Problem Root Cause

The type of disease affecting the crops is unknown to the farmers. The farmer must consult an expert to produce

Your Solution

To determine what type of disease is affecting the plants and how to minimise its effects, deep learning algorithms are used. It also suggests using fertiliser to treat those illnesses.

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

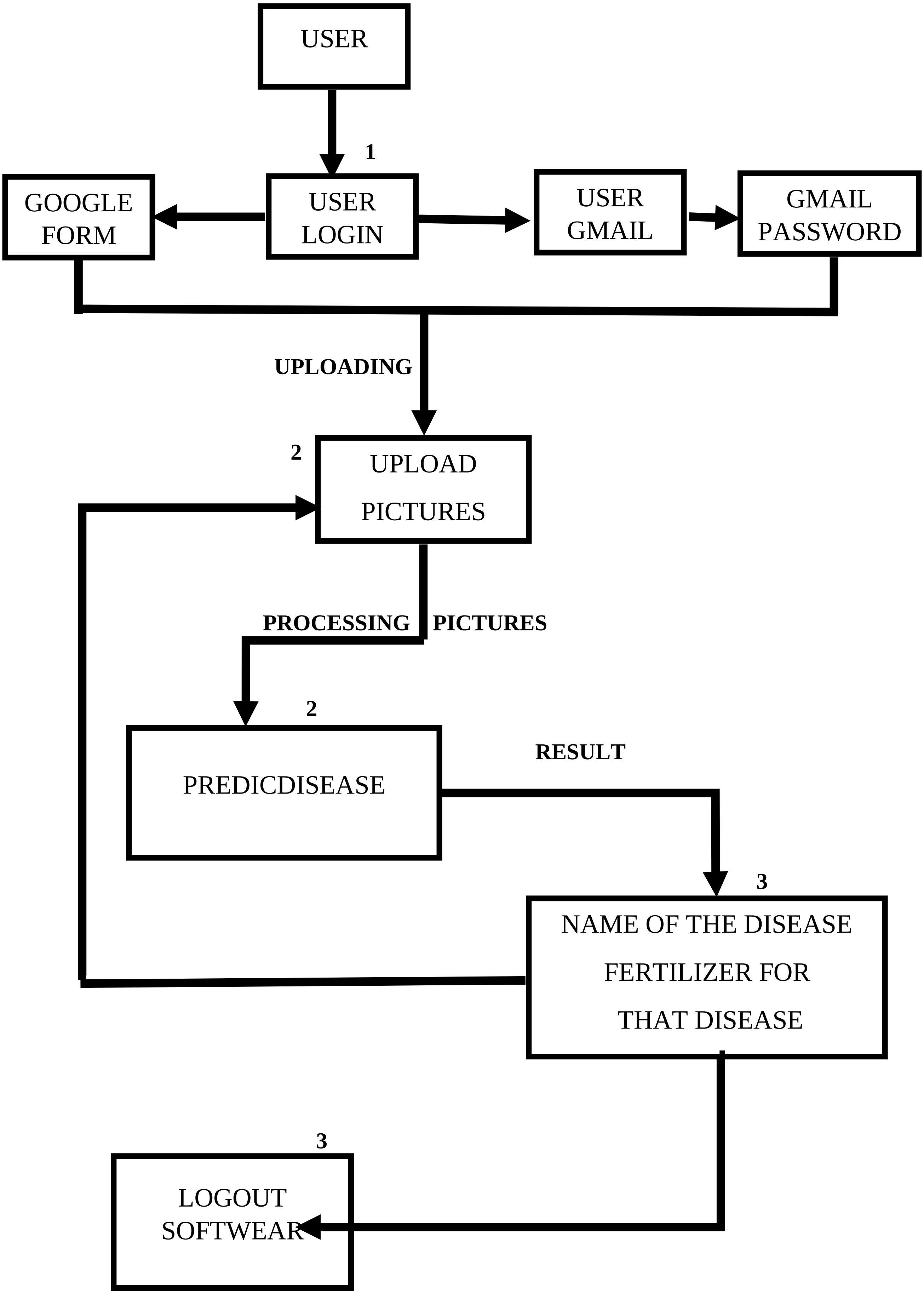
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
1	User Registration	Registration through Form Registration through Gmail
2	User Confirmation	Confirmation via OTP Confirmation via Email
3	User work	Take photo Upload from gallery
4	Image processing	Upload the image for the leaf disease prediction.
5	Prediction result	The disease's official name fertiliser for that disease
6	Final part	Logout Start user work again

4.2 NON-FUNCTIONAL REQUIREMENTS

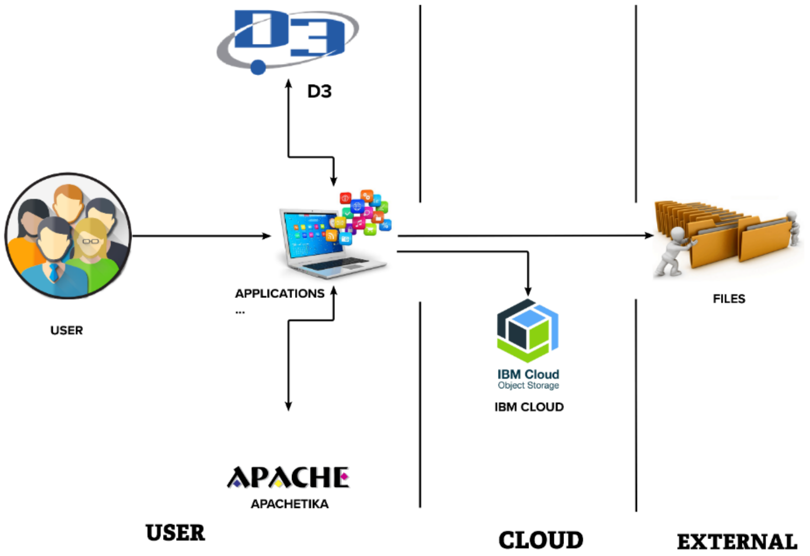
Following are the non-functional requirements of the proposed solution.

FR NO.	Non-functional requirement	Description
1	Usability	The standard of a user's interaction with the system.
2	Security	In order for software to continue to operate as intended, it must be protected from malicious attacks and other hacking dangers.
3	Reliability	Failure-free detection of a computer program for plant disease
4	Performance	The performance is based on the quality of the leaf used for disease prediction
5	Availability	Everyone has access to the ability to predict plant disease.
6	Scalability	The measure of a system's ability to increase or decrease in performance and cost in response to changes in application and system processing demands.

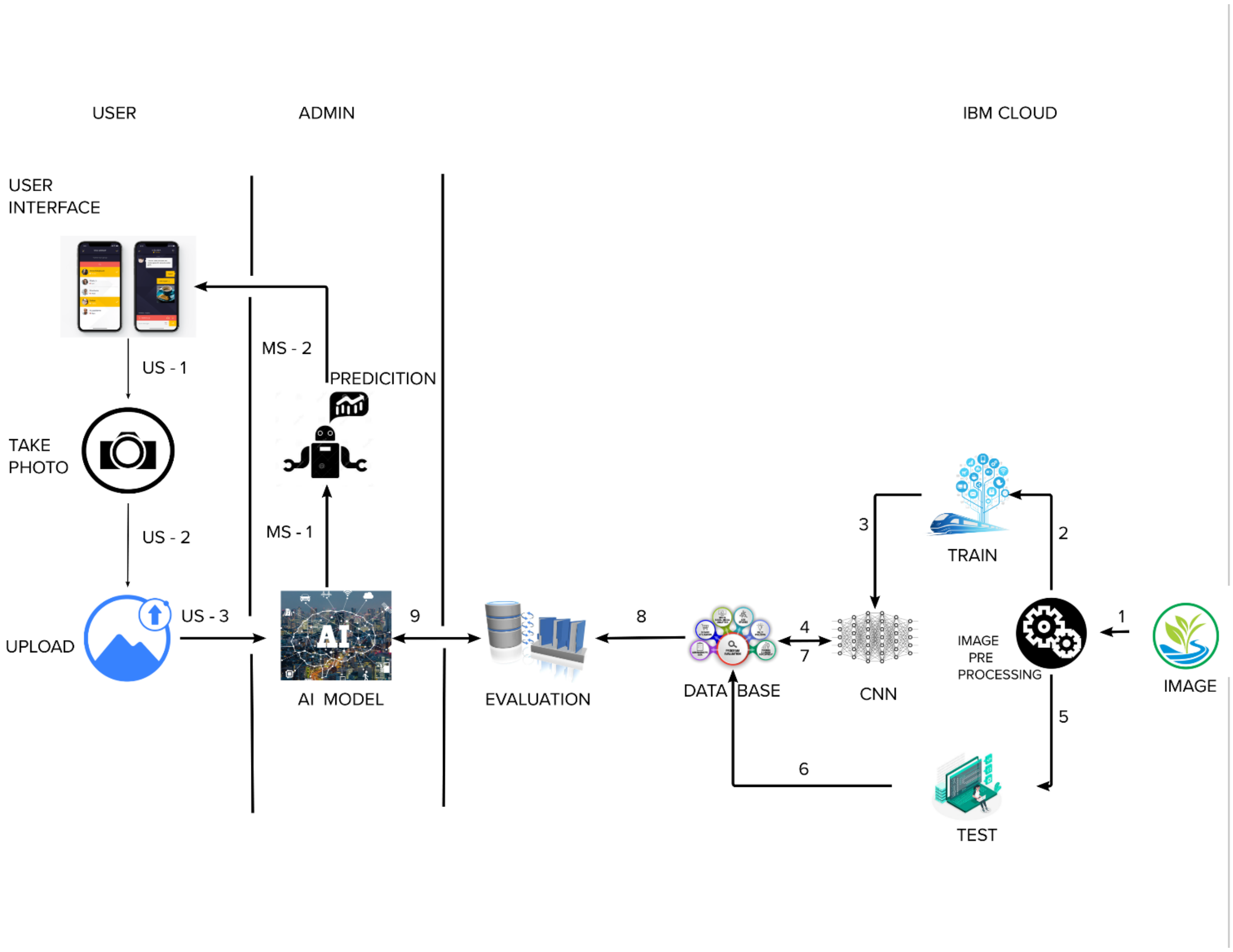
5. PROJECT DESIGN



5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email as a user, password, and confirming my password.	I can access my profile / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can Access the Application via Gmail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can use the Application for Disease Prediction	High	Sprint-1
Customer (Web user)	Registration	USN-6	As a Web user, I can register with a User ID on the System	I can able to access the app as a website	High	Sprint-1
Customer Care Executive	Customer Support	USN-7	As a Supporter, I can Understand exactly how customer use the product	I can create Guidelines and Practices for Customer	Low	Sprint-2
Administrator	Analyst	USN-8	As a Admin, I can Update many dataset about the Plant Diseases	I can able to store large Amount of Data	High	Sprint-1
Customer Purpose	Prediction	USN-9	It uses AI to identify the Plants Disease within the Captured photos and Live View of Prediction	I can Predict Disease of the Plant	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

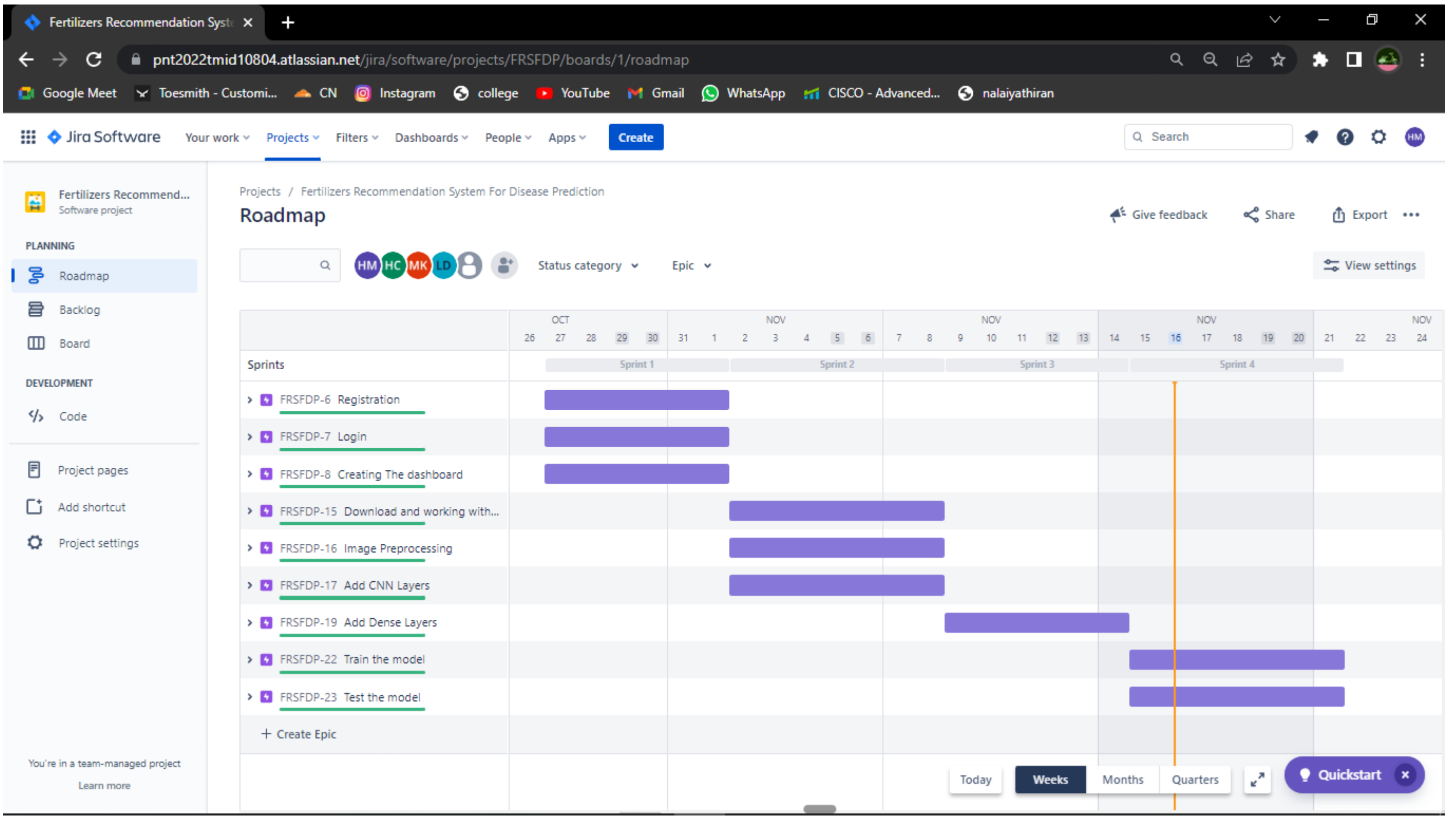
6.1 SPRINT PLANNING & ESTIMATION

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	20	6 Days	29 Oct 2022	03 Nov 2022	20	03 Oct 2022
Sprint-2	20	6 Days	04 Nov 2022	10 Nov 2022	20	10 Nov 2022
Sprint-3	20	6 Days	11 Nov 2022	16 Nov 2022	20	26 Nov 2022
Sprint-4	20	6 Days	17 Nov 2022	23 Nov 2022	20	23 Nov 2022

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Vinothini. M Umasankar.E
		USN-2	As a user, I will receive confirmation email once I have registered for the application	2	High	Vidhya. S Vinothini. M Yuvasri. S
		USN-3	As a user, I can register for the application through Gmail	2	Medium	Vinothini. M Yuvasri. S
	Login	USN-4	As a user, I can log into the application by entering email & password	4	High	Yuvasri. S Vinothini. M Umasankari. E
	Creating The dashboard	USN-5	Create the Dashboard to Interacts with the user interface to upload images	10	High	Vinothini. M Vidhya. S Umasankari. E Yuvasri. S
Sprint-2	Download and working with the dataset	USN-6	To work on the given dataset, Download and Understand the Dataset.	2	Medium	Umasankari. E Vidhya. S
			Load the dataset	2	Medium	Vinothini. M Umasankari. E
	Image Preprocessing		Format images in the dataset before they are used by model training	4	Medium	Yuvasri. S Vinothini. M Vidhya. S
			Import the Libraries	4	Medium	Vinothini. M Umasankari. E
			Initializing the model	4	Medium	Vinothini. M Vidhya. S
	Add CNN Layers	USN-7	Adding three layers for CNN • Convolution layer • Pooling layer • Flattening layer	4	High	Vinothini. M Vidhya. S Umasankari. E Yuvasri. S
Sprint-3	Add Dense Layers	USN-8	Add a hidden layer and output layer	20	High	Vinothini. M Vidhya. S
Sprint-4	Train the model	USN-9	Using the dataset, Train the model for disease prediction to recommend the fertilizer	10	Medium	Vinothini. M Umasankari. E Yuvasri. S
	Test the model	USN-10	Test the model with different data	10	High	Vidhya. S Vinothini. M

6.3 REPORTS FROM JIRA



7. CODING & SOLUTIONING

7.1 FEATURE 1

Image Enhancement - Image Histogram

Import required packages necessary for image processing.

InÂ [1]:

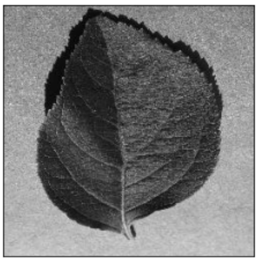
```
import cv2
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

InÂ [2]:

```
img = cv2.imread('/content/0a285c8b-1c31-48d4-89f2-af8b9edc36f6___RS_HL 5759.JPG', 0)
plt.imshow(img, cmap='gray'), plt.grid(False)
plt.xticks([]), plt.yticks([])
```

Out[2]:

```
(([], <a list of 0 Text major ticklabel objects>),
 ([], <a list of 0 Text major ticklabel objects>))
```



1-Histogram of an image

cv2.calcHist(images, channels, mask, histSize, ranges[, hist[, accumulate]])

7.2 FEATURE 2

5380 lines (5380 sloc) | 662 KB

Raw Blame

In [1]:
```
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)
```

In [2]:
```
from google.colab import files
```

In [3]:
```
uploaded = files.upload()
```

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving fruit-dataset.zip to fruit-dataset.zip
```

In [4]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

In [5]:
```
for fn in uploaded.keys():
  print('User uploaded file "{name}" with length {length} bytes'.format(
      name=fn, length=len(uploaded[fn])))
```

```
User uploaded file "fruit-dataset.zip" with length 98553840 bytes
```

In [6]:
```
!unzip drive/"My Drive"/images.zip
```

```
unzip:  cannot find or open drive/My Drive/images.zip, drive/My Drive/images.zip.zip or drive/My Drive/images.zip.ZIP.
```

In [7]:
```
!unzip /content/fruit-dataset.zip
```

```
Streaming output truncated to the last 5000 lines.
```

7.3 DATABASE SCHEMA

```
 flatten_1 (Flatten)         (None, 127008)            0

=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0
_________________________________________________________________
```

In [81]:
```
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
```

In [82]:
```
model.add(Dense(9,activation='softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
```

Out[82]: 475

In [76]:
```
1238/24
```

Out[76]: 51.583333333333336

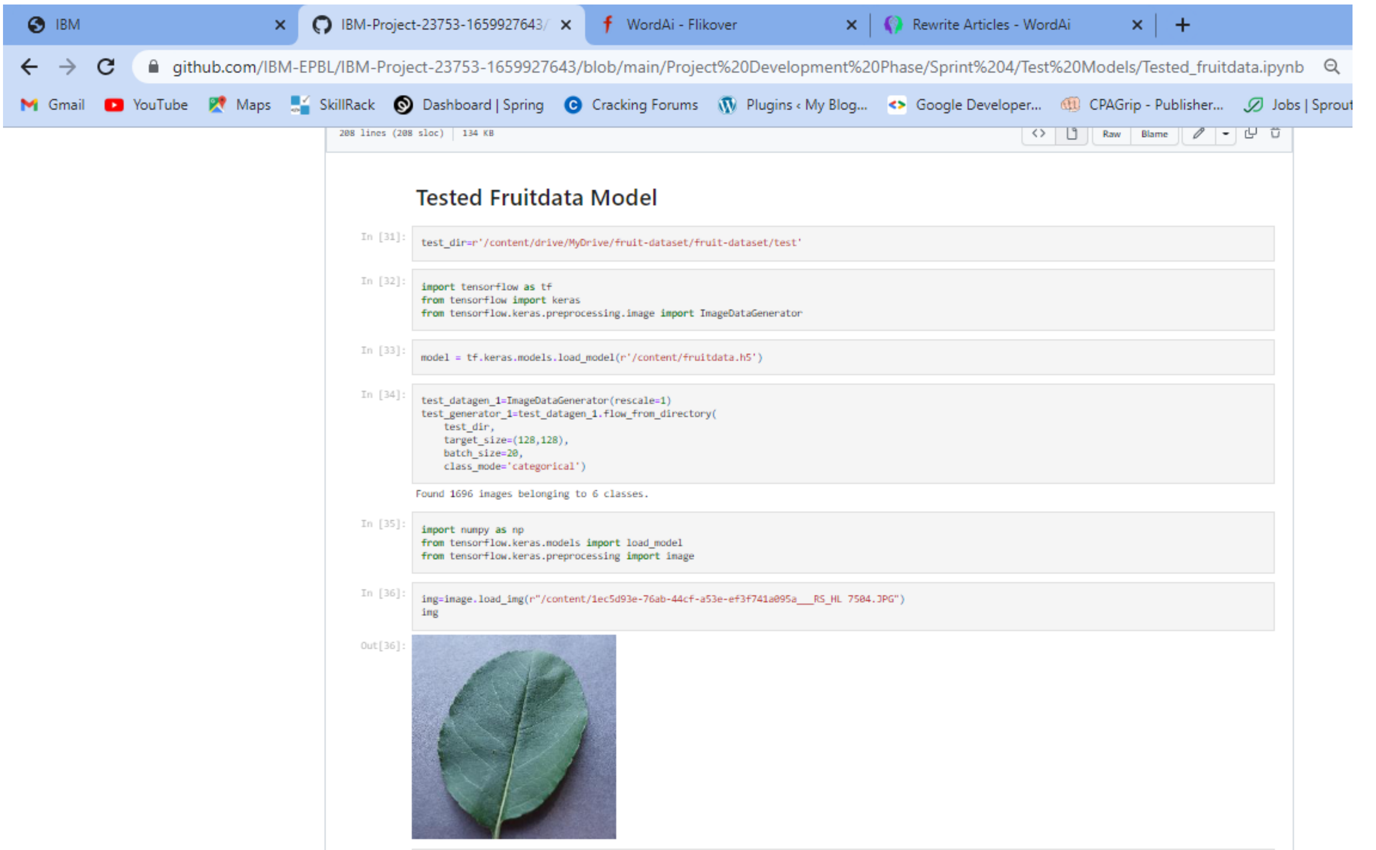
In [83]:
```
model.fit(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
```

```
Epoch 1/10
475/475 [==============================] - 337s 707ms/step - loss: 2.0829 - accuracy: 0.1790 - val_loss: 2.0840 - val_accuracy: 0.1953
Epoch 2/10
475/475 [==============================] - 334s 703ms/step - loss: 2.0765 - accuracy: 0.1815 - val_loss: 2.0986 - val_accuracy: 0.1232
Epoch 3/10
475/475 [==============================] - 332s 699ms/step - loss: 2.0769 - accuracy: 0.1831 - val_loss: 2.0859 - val_accuracy: 0.1754
Epoch 4/10
475/475 [==============================] - 333s 701ms/step - loss: 2.0762 - accuracy: 0.1829 - val_loss: 2.0844 - val_accuracy: 0.1953
Epoch 5/10
475/475 [==============================] - 333s 701ms/step - loss: 2.0763 - accuracy: 0.1855 - val_loss: 2.0847 - val_accuracy: 0.1953
Epoch 6/10
475/475 [==============================] - 331s 697ms/step - loss: 2.0759 - accuracy: 0.1843 - val_loss: 2.0835 - val_accuracy: 0.1754
Epoch 7/10
475/475 [==============================] - 332s 699ms/step - loss: 2.0752 - accuracy: 0.1797 - val_loss: 2.0838 - val_accuracy: 0.1953
```

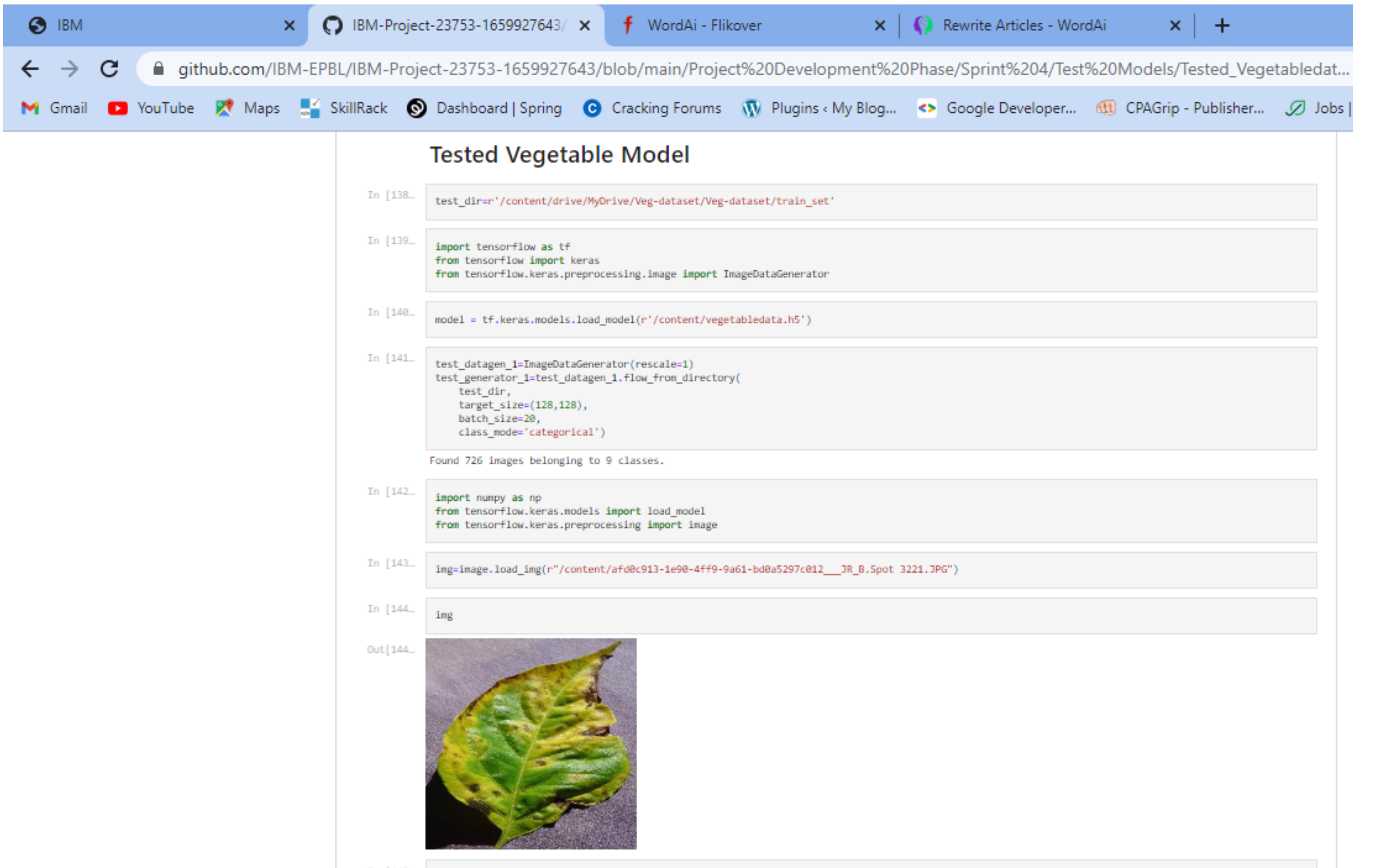
8. TESTING

8.1 TEST CASES

Fruit Case

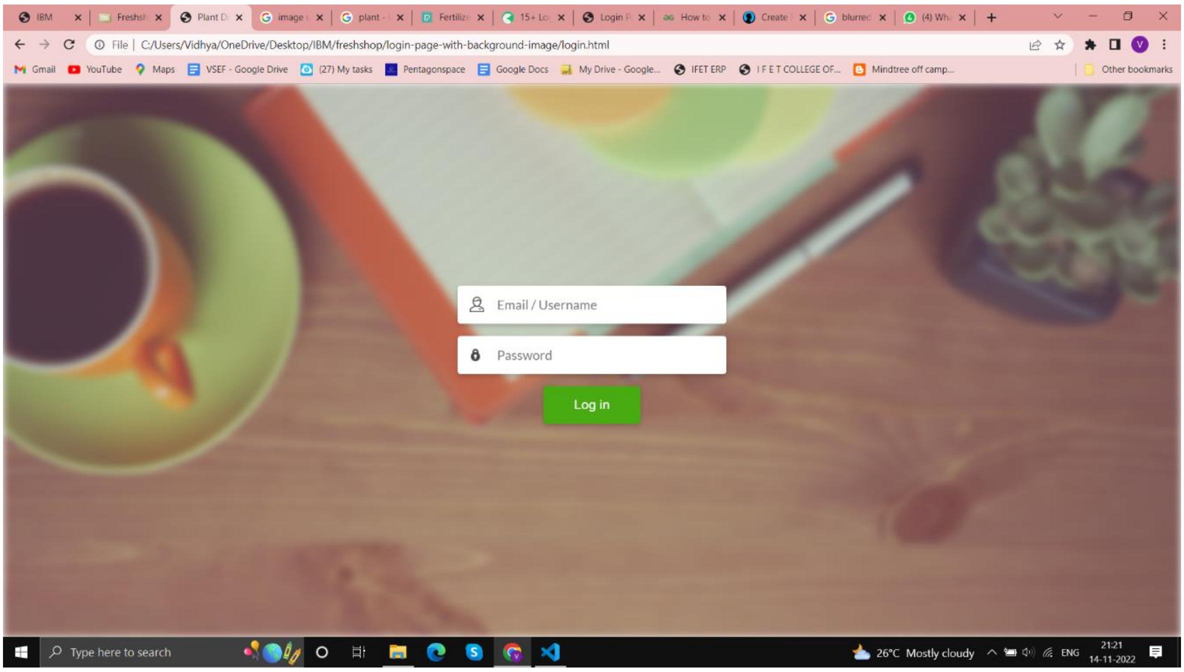


Vegetable Case

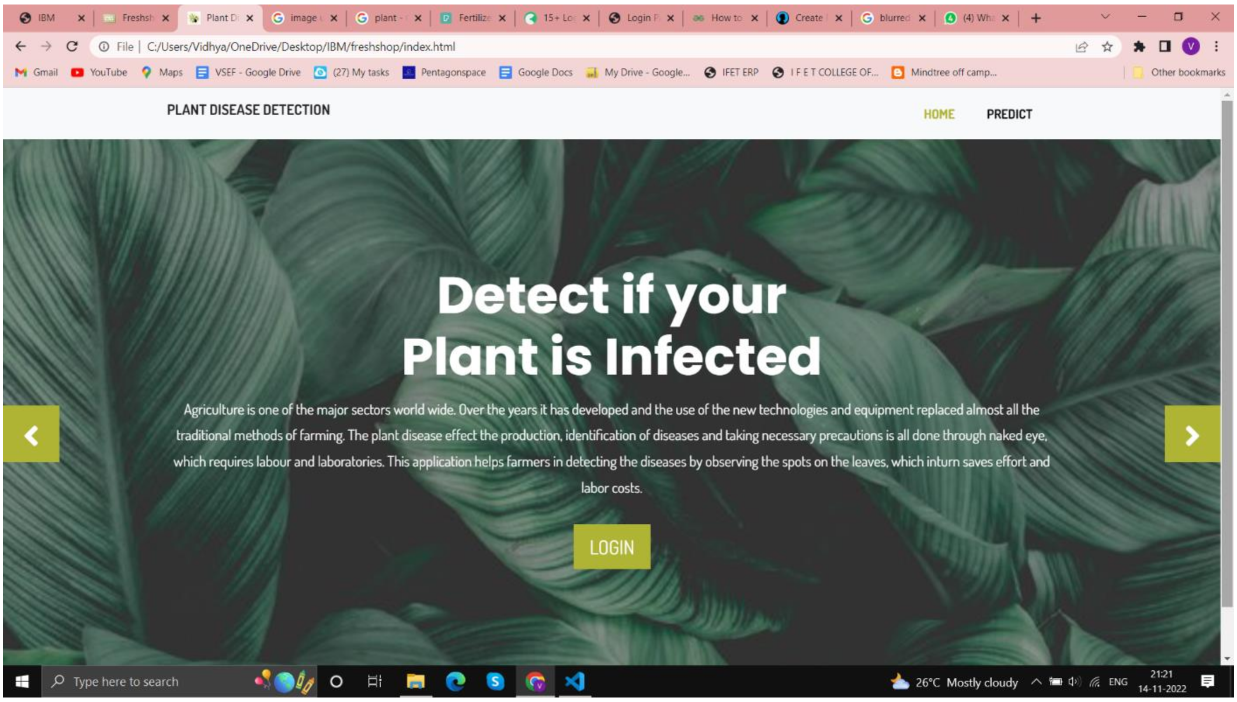


9. RESULTS

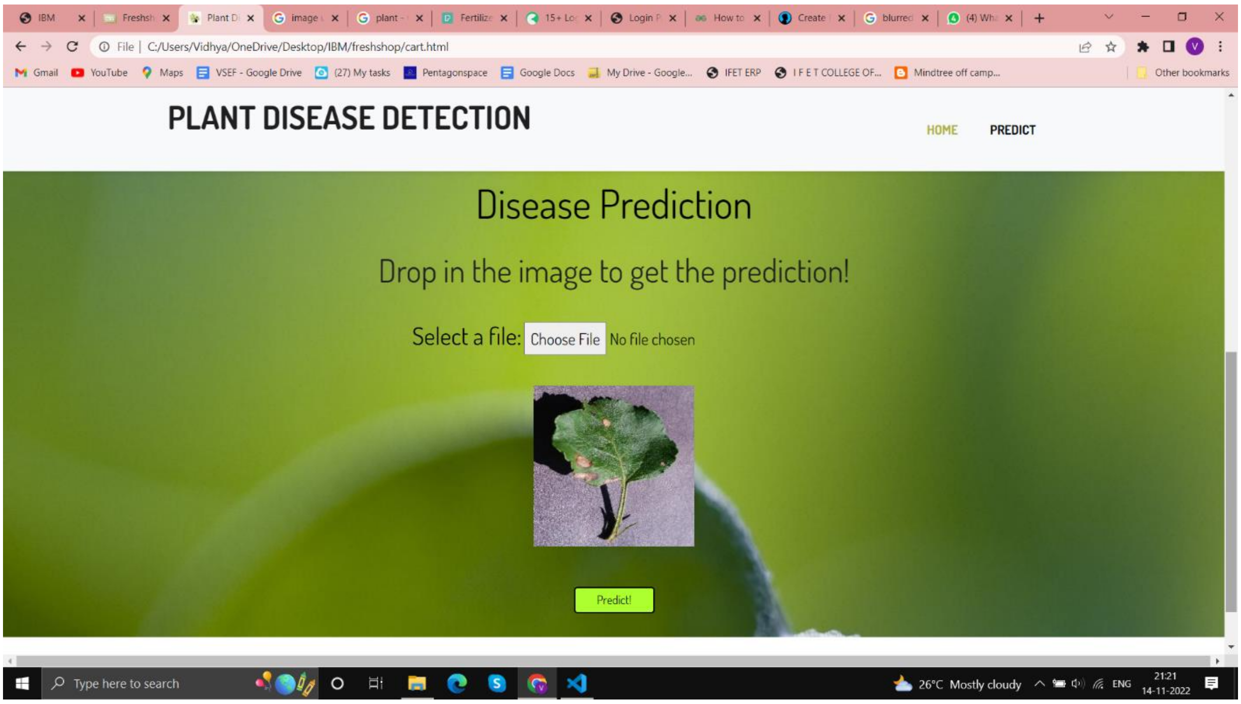
Login:



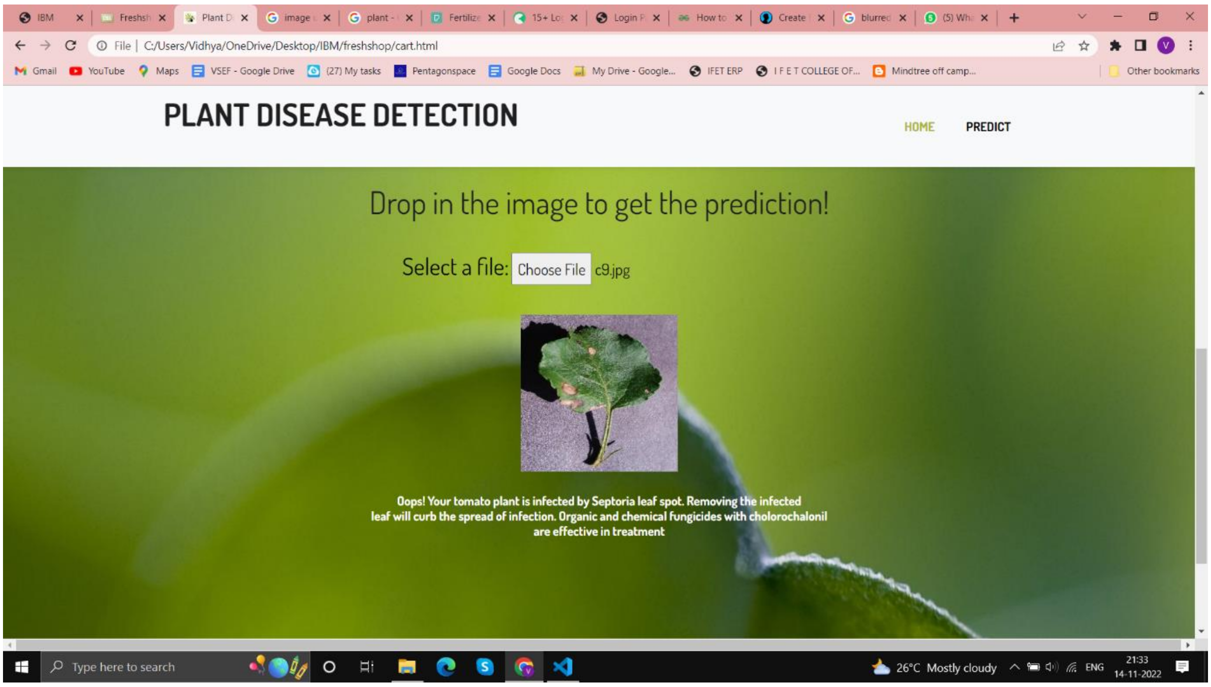
Description Login:



Disease Leaf:



Fertilizer:



10. ADVANTAGES & DISADVANTAGES

Advantages:

- The proposed model here produces very high accuracy of classification.
- Very large datasets 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 requires very high computational time.
- The neural network architecture used in this project work has high complexity.

11. CONCLUSION

This model involves image classification of vegetable datasets and fruit datasets. During model training and testing, the following observations were made: Classification accuracy increased with increasing the number of epochs. Different batch sizes yield different classification accuracy. By adding more convolution layers, the accuracy of classification can be increased. Variable density layers can also improve classification accuracy. Variation in the kernel size used to produce the convolution layer output can result in different accuracy levels. The size of the test and train datasets can have an impact on the accuracy.

12. FUTURE SCOPE

Image recognition can be applied to the model proposed in this project work. The whole model can be converted into application software with python-to-exe software. OpenCV python library allows for real-time image classification, recognition, and video processing. This project can be used for security applications like face recognition, iris recognition, and figure print recognition.

13. APPENDIX

SOURCE CODE

```
import requests

from tensorflow.keras.preprocessing import image

from tensorflow.keras.models import load_model

import numpy as np

import pandas as pd

import tensorflow as tf

from flask import Flask, request,render_template,redirect,url_for
```

```
import os
from werkzeug.utils import secure_filename
from tensorflow.python.keras.backend import set_session

app = Flask(__name__)
model=load_model("fruit.h5")
model1=load_model("veg.h5")

@app.route('/')
def home():
    return render_template('home.html')
@app.route('/prediction')
def prediction():
    return render_template('predict.html')
@app.route('/predict',methods=['POST'])
def predict():
    if( request.method=='POST'):
        f = request.files['image']
        basepath=os.path.dirname(__file__)
        file_path=os.path.join(basepath,'uploads',secure_filename(f.filename))
        f.save(file_path)
        img=image.load_img(file_path,target_size=(128,128))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)
        plant=request.form['plant']
        print(plant)
        if(plant=='vegetable'):
            preds =model1.predict(x)
            classes=np.argmax(preds,axis=1)
            print(classes)
```

```
            df=pd.read_excel('precautions - veg.xlsx')
            print(df.iloc[classes[0]]['caution'])
        else:
            preds =model.predict(x)
            classes=np.argmax(preds,axis=1)
            df=pd.read_excel('precautions - fruits.xlsx')
            print(df.iloc[classes[0]]['caution'])
        return df.iloc[classes[0]]['caution']
if __name__=="__main__":
    app.run(debug=False
```

GITHUB & PROJECT DEMO LINK

Github Link:

https://github.com/IBM-EPBL/IBM-Project-23753-1659927643

Project Demo Link:

https://1drv.ms/v/s!AhqjGPy5BGCNhHcZfnGXyFGcR5yV?e=k EZgHe