Fertilizers Recommendation System for Disease Prediction

PROJECT REPORT

Submitted by,

Team ID: PNT2022TMID33383

S.Dharani (922519104030)
R.Dharshini(922519104032)
M.Nandhini(922519104101)
N.Narmatha(922519104103)

In partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

DEPARTMENT OF COMPUTER SCIENCEAND ENGINEERING

V.S.B. ENGINEERING COLLEGE, KARUR

V.S.B ENGINEERING COLLEGE, KARUR

(Approved by AICTE & Affiliated by Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report "**Fertilizer Recommendation System for Disease Prediction"** is the Bonafide record work done by **Ms DHARANI S** (**922519104030**), **Ms DHARSHINI R** (**922519104032**) , **Ms NANDHINI** (**922519104101)**, and **Ms NARMATHA N** (**922519104103**) for **IBM- NALAIYATHIRAN** in **VII** semester of **B.E.,** degree course **in Computer Science and Engineering** branch during the academic year of 2022 – 2023.

Staff-In Charge
MANIKANDAN.T

Evaluator
Gunasekaran P

Head of the Department
Mr. Anbumani P

ACKNOWLEDGEMENT

First and foremost, we express my thanks to our parents for providing us a very nice environment for doing this mini project. We wish to express our sincere thanks to our founder and Chairman **Shri.V.S.BALSAMY** for his endeavour in educating us in this premier institution.

We wish to express our appreciation and gratefulness to our principal, **Dr.V.NIRMAL KANNAN** and vice principal **Mr.T.S.KIRUBASANKAR** for their encouragement and sincere guidance.

We are grateful to our head of the department **Mr.P.ANBUMANI** and our IBM Nalaiyathiran project coordinator **Mr.P.GUNASEKARAN** Department of Computer Science and Engineering for their valuable support.

We express our indebtedness to the supervisor of our IBM Nalaiyathiran project, **Mr.P.GUNASEKARAN** Assistant Professor, Department of Computer Science and Engineering, for guidance throughout the course of our project.

Our sincere thanks to all the teaching staff of V.S.B Engineering College and our friends for their help in the successful completion of this IBM Nalaiyathiran project work. Finally, we bow before God, the Almighty who always had a better plan for us. We give our praise and glory to Almighty God for successful completion of this IBM Nalaiyathiran.

TABLE OF FIGURES

1.INTRODUCTION

Detection and recognition of plant diseases using machine learning are very efficient in providing symptoms of identifying diseases at its earliest. Plant pathologists can analyse 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, the 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 depending on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves.

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.

1.1PROJECT OVERVIEW

Overview In this project, two datasets name fruit dataset and vegetable dataset are collected. The collected datasets are trained and tested with deep learning neural network named Convolutional Neural Networks (CNN). First, the fruit dataset is trained and then tested with CNN. It has 6 classes and all the classes are trained and tested. Second, the vegetable dataset is trained and tested. The software used for training and testing of datasets is Python. All the Python codes are first written in Jupyter notebook supplied along with Anaconda Python and then the codes are tested in IBM cloud. Finally, a web-based framework is designed

with help Flask a Python library. There are 2 html files are created in templates folder along with their associated files in static folder. The Python program 'app.py' used to interface with these two webpages is written in Spyder-Anaconda python and tested.

- Pre-process the images.
- Applying the CNN algorithm to the dataset.
- How deep neural networks detect the disease.
- You will be able to know how to find the accuracy of the model.
- You will be able to build web applications using the Flask framework.

Create a system for predicting crops according to soil details, predicting fertilizers according to soil and crop details, and detecting diseases in the plant. The objective of our system is to help farmers because it's difficult to grow interventions. Each control approach is then accompanied with the aid of using danger nodes representing unsure occasions (i.e., 'disorder free' or 'dead', with a view to having possibilities connected to them. Finally, endpoints of DTs are represented with the aid of using a terminal node (triangle) on the proper of the tree. The final results measures (e.g., software

value) are usually connected to those endpoints. Costs, however, are connected to occasions in the tree, in addition to endpoints. The anticipated values (expenses and effectiveness) related to every department are expected with the aid of using 'averaging out' and 'folding back' the tree from proper to left.

1.2 PURPOSE

This project is used to test the fruits and vegetables samples and identify the different diseases. Also, this project recommends fertilizers for predicted diseases. In day-to-day life, Agriculture is the most important sector. Most plants are affected by a wide variety of bacterial and fungal diseases. Farmers face several challenges when growing crops like uncertain irrigation, poor soil quality, etc. Especially in India, a major fraction of farmers does not have the knowledge to select appropriate crops and fertilizers. 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 depending on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves Moreover, crop failure due to disease causes a significant loss to the farmers, as well as the consumers. While there have been recent developments in the automated detection of these diseases using Machine Learning techniques, the utilization of Deep Learning has not been fully explored. Additionally, such models are not easy to use because of the high-quality data used in their training, lack of computational power, and poor generalizability of the models. To this end, we create an open- source easy-to-use web application to address some of these issues which may help improve crop production. In particular, we support crop recommendation, fertilizer recommendation, plant disease prediction, and an

interactive news-feed. In addition, we also use interpretability techniques in an attempt to explain the prediction made by our disease detection model.

2.LITERATURE SURVEY

Numerous articles have been reviewed and their conclusions are summarized in this section. This section presents documents that were studied before and during project development. The documents provided a better understanding of existing solutions, how algorithms could be optimized and how selection could be facilitated algorithms on the basis of their performance.

2.1 EXISTING PROBLEM

The author says that in India, the largest source of subsistence is agriculture and its federated sectors. In rural regions, there are about 82% of small and marginal farmers, and 70% of rural households depend primarily on agriculture only. The proposed system recommends the suitable crops for the lands with varied soil nutrients. The appropriate fertilizers that are suitable for specific soil nutrient and crop sown are also recommended. Plant physiology can be damaged due to fungal, viral or bacterial diseases. Plants affected from the above pathogens are detected. Random forest classifier gives an accuracy of 98% for recommendation system, and PyTorch neural network gives an accuracy of 99.2% for disease prediction [1]. The Author states that Agriculture is the main aspect for the economic development of a country. Agriculture is the heart and life of most Indians. The soil type, fertilizer recommendation, diseases in plants and leaves. Plant disease, especially on leaves, is one of the major factors that reduce the yield in both quality and quantity of the food crops. Finding the leaf disease is an important role to preserve agriculture. Smart analysis and Comprehensive prediction model in agriculture helps the farmer to yield right crop at the right time. The main benefits of the proposed system are as follows: Yield right crop at the right time, Balancing the crop production, control plant disease, Economic growth, and planning to reduce the crop scarcity. Hence to Detect and recognize the plant diseases and to recommend fertilizer it is necessary to provide symptoms in identifying the disease at its earliest [2]. The Author claims that Agriculture is the mainstay of a rising economy in India.Traditionally farmers followed ancestral farming patterns and norms. However, a single farmer cannot be expected to take into account all innumerable factors that contribute to crop growth. A single misguided or imprudent decision by the farmer can have undesirable ramifications. With the advancements in various domains, intelligent agricultural system is needed for upliftment of Indian economy. The collaboration of recommender system with machine learning will lead to Intelligent Agriculture System that helps the farmer community in their decision making of farm management and agribusiness activities such as

a. Predicting agriculture commodity market price before cultivation,
b. Determining best cultivars to plant
c. Determine optimum cultivation date

d. Evaluate demand and supply risk

e. Investment Prioritizing. It also helps farmer to perform the activities like crop management including applications on yield prediction, disease detection, weed detection, crop quality, and growth prediction etc. This chapter describes the case study on "Crop Disease Detection and Yield prediction". The study includes identification of crop condition, disease detection, prediction about specific crop and recommendation using machine learning algorithms. It gives an idea about how recommender system is used in agriculture for disease detection and prediction [3].

2.2 REFERENCE

[1] Suriya Krishnaan, K., Kumar, L. C., & Vignesh, R. (2022). Recommendation System for Agriculture Using Machine Learning and Deep Learning. In Inventive Systems and Control (pp. 625-635). Springer, Singapore.

[2] Selvi, P. P., & Poornima, P. Soil Based Fertilizer Recommendation System for Crop Disease Prediction System.

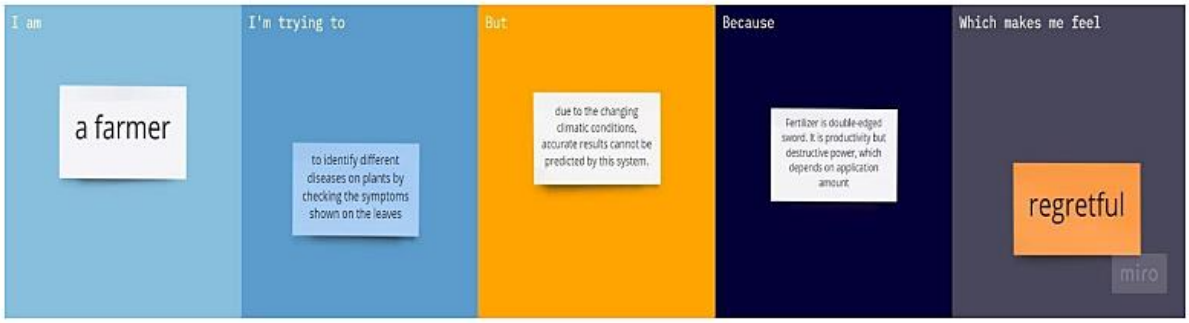
[3] Akulwar, P. (2020). A recommended system for crop disease detection and yield prediction using machine learning approach. Recommender System with Machine Learning and Artificial Intelligence: Practical Tools and Applications in Medical, Agricultural and Other Industries, 141-163.

2.3 PROBLEM STATEMENT DEFINITION

In this project work, a deep learning based neural network is used to train the collected datasets and test the same. The deep learning based neural network is CNN which gives more than 90% classification accuracies. By increasing Image Preprocessing Image dataset collection Image dataset training Build & Save Mode Predict the test dataset the more number of dense layers and by modifying hyperparameters such as number of epochs, batch size, the accuracy rate can be increased to 95% to 98%.

Farmers' conventional methods of agricultural cultivation are ineffective. It does not make proper use of all available resources. Farmers are unable to detect crop diseases due to a lack of knowledge and old practices, which often result in soil nutrient deterioration and exhaustion. As a result, crop failure occurs. Growing only certain crops depletes the soil, and if the crops are harmed by illnesses, farmers are uninformed of how to recover such crops. Food needs cannot be met until and unless efficient resource management and use is implemented.

Suggesting the best compost for each specific harvest is likewise a difficult undertaking. Furthermore, the other and most significant issue is the point at which a plant gets found out by heterogeneous sicknesses that impact on less measure of horticulture creation and compromises with quality too.



I am	Farmers are the customers working in the cultivation land. People who Grow Crops and facing Issues of Plant Disease.
I'm trying to	An automated system is introduced to identify different diseases on plants by checking the symptoms shown on the leaves of the plant and recommendation of the fertilizers by image processing.
But,	Protecting crops in organic farming is not an easy task. Plant disease, especially on leaves, is one of the major factors of reductions in both quality and quantity of the food crops. Fertilizer Recommendation is not taken into consideration. Due to the changing climatic conditions, accurate results cannot be predicted by this system.
Because	Fertilizer is double-edged sword. It is productivity but destructive power, which depends on application amount. Having some reasonable knowledge of application of fertilizer can avoid unnecessary waste in the purchase of fertilizers and increase the yield of crops.
Which makes me feel	These things make feel that it can't protect my crops and get loss in my investment. So, I wish to get expert knowledge to save my field.

3. IDEATION & PROPOSED SOLUTION

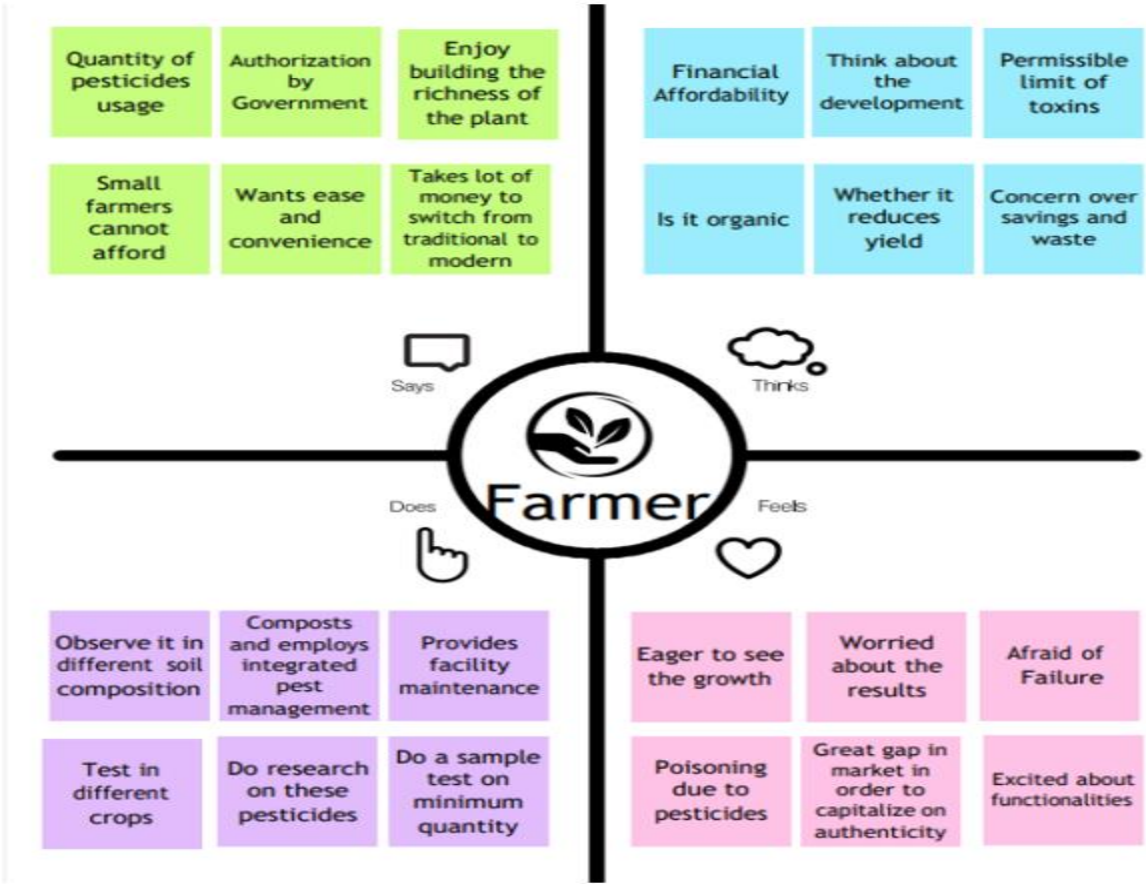
3.1 EMPATHY MAP CANVAS

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment.

Agriculture is the main aspect of the economic development of a country. Agriculture is the heart and life of most Indians. By understanding their feelings and problems, we can create a better product and contribute to their lives. For our project, we are getting surveys from farmers to understand what they truly require and desire.

In addition to being essential to the economy, agriculture is seen as the foundation of the economic system in developing nations. In India, the agriculture and related sectors employ 54.6% of the total labour force. We are building a method based on a survey of home farmers and farmers to increase agriculture efficiency and alleviate challenges faced by both farmers and those engaged in this activity

EMPATHY MAP



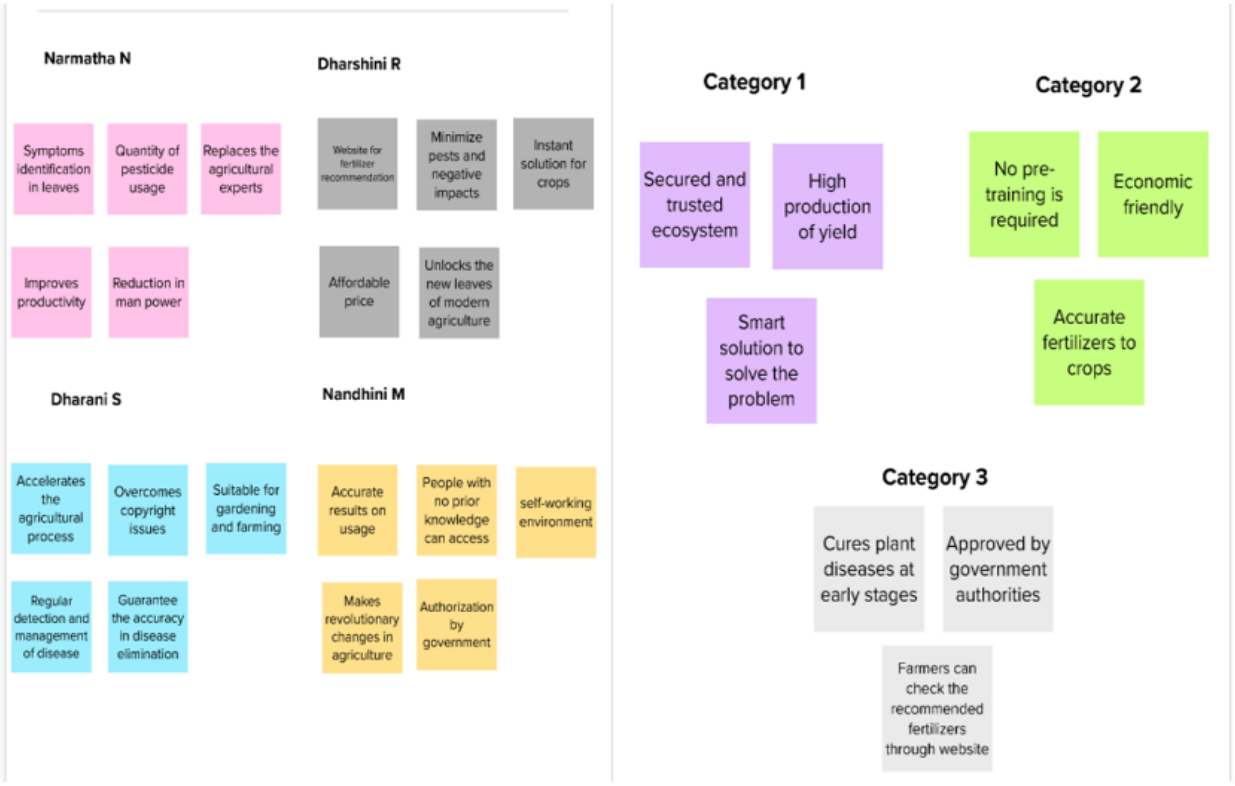
3.2 IDEATION & BRAINSTROMING

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.

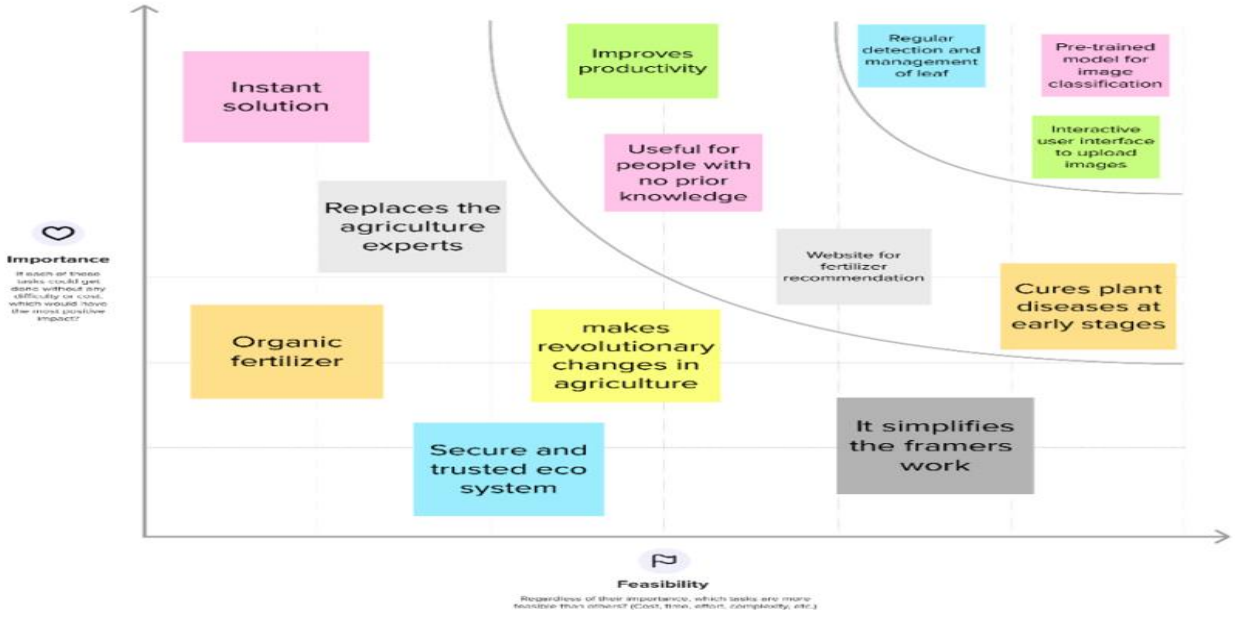
Brainstorming is one of the primary methods employed during the Ideation stage of a typical Design Thinking process. Brainstorming is a great way to generate many ideas by leveraging the collective thinking of the group, engaging with each other, listening, and building on other ideas.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



3.3 PROPOSED SOLUTION

In this project work, a deep learning based neural network is used to train thecollected datasets and test the same. The deep learning based neural network is CNN which gives more than 90% classification accuracies. By increasing the more number of dense layers and by modifying hyperparameters such as number ofepochs, batch size, the accuracy rate can be increased to 95% to 98%.

S.No.	Parameter	Description
1.	Problem Statement (Problem tobe solved)	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 productionand a major threat to food security. Hence, early andaccurate identification of plant diseases is essentialto 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. Our project is proposed to build an automated system which 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.
2.	Idea / Solution description	The user needs a way to get relevant fertilizer based on the disease symptoms shown in the leaves. Protecting crops in organic farming is not an easy task. Plant disease, especially on leaves, isone of the major factors of reductions in both quality and quantity of the food crops. Fertilizer Recommendation is not taken into consideration. So, it is necessary to develop crop yield predictionand fertilizer recommendation system which predicts crop yield and recommend fertilizer for selected crop based on different datasets like fertilizer data,location data andcrop yield data.

3.	Novelty / Uniqueness	Image data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset. Recommend the fertilizer for affected leaves based on severity level. Fertilizers may be organic or inorganic. Admin can store the fertilizers based on disease categorization with severity levels. The field of agriculture is in a great threat this includes the diseases that attack the plant leaves. Our system finds the area of the leaf that has been affected and also the disease that attacked the leaves. A system that automatically detects leaf disease with the help of image processing is being developed. This system does few image pre-processing techniques like image acquisition, image segmentation, feature extraction and classification.
4.	Social Impact / Customer Satisfaction	User friendly application, the customers are satisfied as we use different methods include different fundamental processes like segmentation, feature extraction and classification and so on. Mostly, the prediction and diagnosis of leaf diseases are depending on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves, so the farmers get high yield.
5.	Business Model (Revenue Model)	Making the business model strong leads to an early identification of disease and the fertilizer is recommended. Crop Yield Prediction can be done using crop yield data, nutrients. Fertilizers can support rolling out beneficial improvements in cultivating by expanding crop yield. Farmers can diminish costs per unit of creation and increment the edge of return over absolute expense by expanding paces of use of fertilizer on chief money and feed crops. This not simply offers huge benefits for farmer jobs and food security, however also conveys natural benefits by diminishing our solicitations for farmland.

6.	Scalability of the Solution	This model reflects different approaches to recognize the illnesses in crops. This model focuses on detecting the plant disease and providing suitable fertilizer which is used for accurate treatment of diseases on crops to provide high quantity and improved quality of products.

3.4 PROBLEM SOLUTION FIT

1. CUSTOMER SEGMENT(S)

Farmers are the customers who are unable to identify fertilizers needed for the affected plants.

2. JOBS-TO-BE-DONE / PROBLEMS

Using AI Technology diagnose and identify the symptoms of disease in plants regularly and provide the right solution at right time.

3. TRIGGERS

Advertising the fertilizer for disease prediction through the social media. Conducting the awareness program and displaying the types of plant diseases and their remedies.

4. EMOTIONS: BEFORE / AFTER

Through traditional farming, farmers do not predict the future but in AI farming farmers can predict the future and get high yield.

5. AVAILABLE SOLUTIONS

First, we should identify the disease of the leaf and it causes. The measurements of the fertilizer should be suggested based on the severity of the disease and based on the soil.

6. CUSTOMER CONSTRAINTS

Cost of Fertilizer is high. Using of not suitable fertilizer can harm the plant and environment.

7. BEHAVIOUR

By addressing the Problem, they can get the high yield by utilizing the correct amount of fertilizer according to the leaf disease and the surrounding factors.

8. CHANNELS of BEHAVIOUR

8.1 ONLINE With the help of the helpline number farmer can get clarify about the queries.

8.2 OFFLINE By consulting with expert people farmers can get help.

9. PROBLEM ROOT CAUSE

There is no proper guidance for the farmer to prevent the leaf from dreadful diseases.

10. YOUR SOLUTION

The customers are unable to identify the disease in plants and feel fear to utilize the fertilizer by think that can harm their environment and fails to yield high. To overcome the farmers problem we use Artificial Intelligence to predict the problem automatically and report the problem. So that we can prevent the disease at early stages and by suggesting the required amount of fertilizes which leads to high yield.

4. REQUIREMENT ANALYSIS

There are two types of requirement analysis, namely

1. Functional Requirements
2. Non-functional Requirements

4.1 FUNCTIONAL REQUIREMENTS

A functional requirement defines a system or its component. A non-functional requirement defines the quality attribute of a software system. It specifies "What should the software system do?" It places constraints on "How should the software system fulfil the functional requirements?"

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through Website
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP Confirmation via Message
FR-3	User Requirements	Get the requirements for the farmers' plants. Capture the image of the leaf And check the parameter of the captured image.
FR-4	User establishment	Establish the good recommendation of fertilizers using the requirements. Upload the image for the prediction and predict the disease in leaf.
FR 5	User review	Suggesting the best fertilizer for the disease.

4.2 NON-FUNCTIONAL REQUIREMENTS

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Predicting the fertilizers, Analyzing the disease in a tap makes the life of farmers easy with minimal subscriptions. It detects many diseases in cropsand recommends appropriate fertilizersto help them recover. It gives farmers vital information about farming techniques to assist them to enhance crop productivity. Datasets of all the leaf is used to detecting the disease that present in the leaf.
NFR-2	**Security**	Armed with sensors and other monitoring devices, the present farmers can oversee field conditions without going to the homestead. The information belongs to the user and leaves are secured highly.
NFR-3	**Reliability**	For constant admittance to in-handle information, network unwavering quality is critical. With regards to maintainable agribusiness, this nullifies the motivation behind accuracy brilliant cultivating, which depends on the most cutting-edge information and constant natural observing. The leaf quality is important for the predicting the disease in leaf.
NFR-4	**Performance**	The performance is based on the quality of the leaf used for disease prediction. It mainly provides better performance so it helps in high productivity and ensures the attack of disease.
NFR-5	**Availability**	It is available for all users to predict the disease in the plant. This assists ranchers with working on the nature of their items as well as the accessibility, bring down their expenses, and increment the client experience.

NFR-6	**Scalability**	Increasing the prediction of the disease in the leaf. The versatility of a framework to expand the limit, for instance.

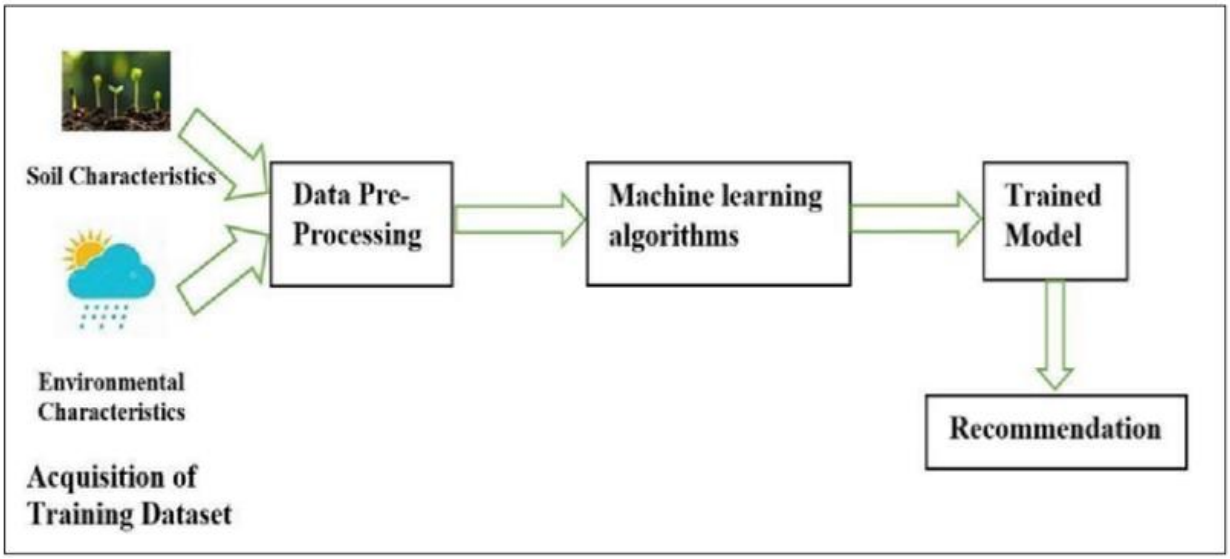
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

A 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.

Example: (Simplified)

DIAGRAM



5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

SOLUTION ARCHITECTURE:

A solution architecture (SA) is **an architectural description of a specific solution**. SAs combine guidance from different enterprise architecture view points(business, information and technical), as well as from the enterprise solution architecture (ESA).

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Predicting the fertilizers, Analysing the disease in a tap makes the life of farmers easy with minimal subscriptions would provide an acceptable return for the organization. This action adds a lot of value to the company and the business in society.

- The device is created primarily for farmers. It detects many diseases in crops and recommends appropriate fertilizers to help them recover. It gives farmers vital information about farming techniques to assist them to enhance crop productivity.

- Providing a fertilizer recommendation system to enrich the soil and improve land productivity and system is evaluated by using appropriate timing and accuracy measures.

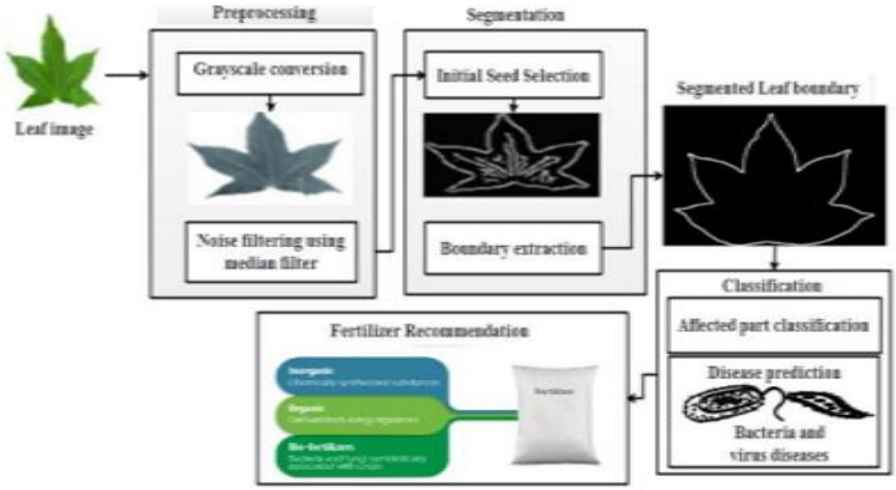
- Analyse data on symptoms, disease types, and medical treatments to provide the best solution for treating diseases. By providing the construction of a recommendation system that facilitates the identification of pest and the selection of suitable treatments.

- This depicts some promising results to present enhanced methods and tools for creating fully automated pest identification including the extraction with detection.

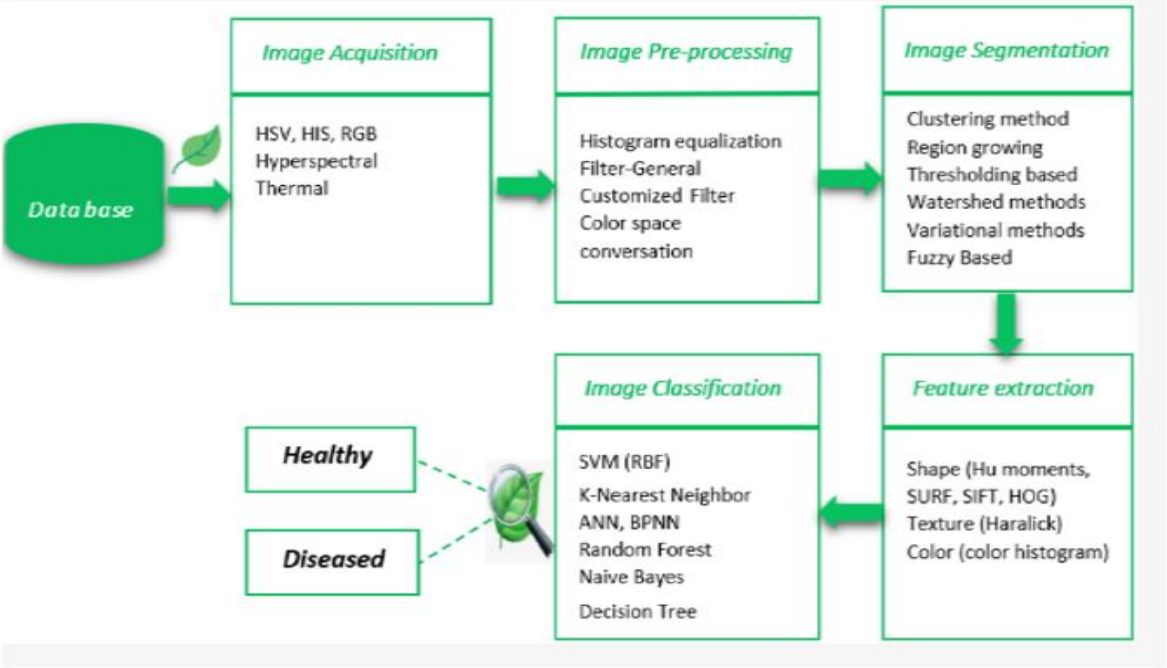
- Plants nowadays are affected by many diseases such as they cause devastating economic, social and ecological losses and many more. Hence, it is most important to identify plants disease in an accurate and timely way. Plant diseases can be extensively grouped by the idea of their essential causal operator, either irresistible or non-infectious.

Solution Architecture Diagram:

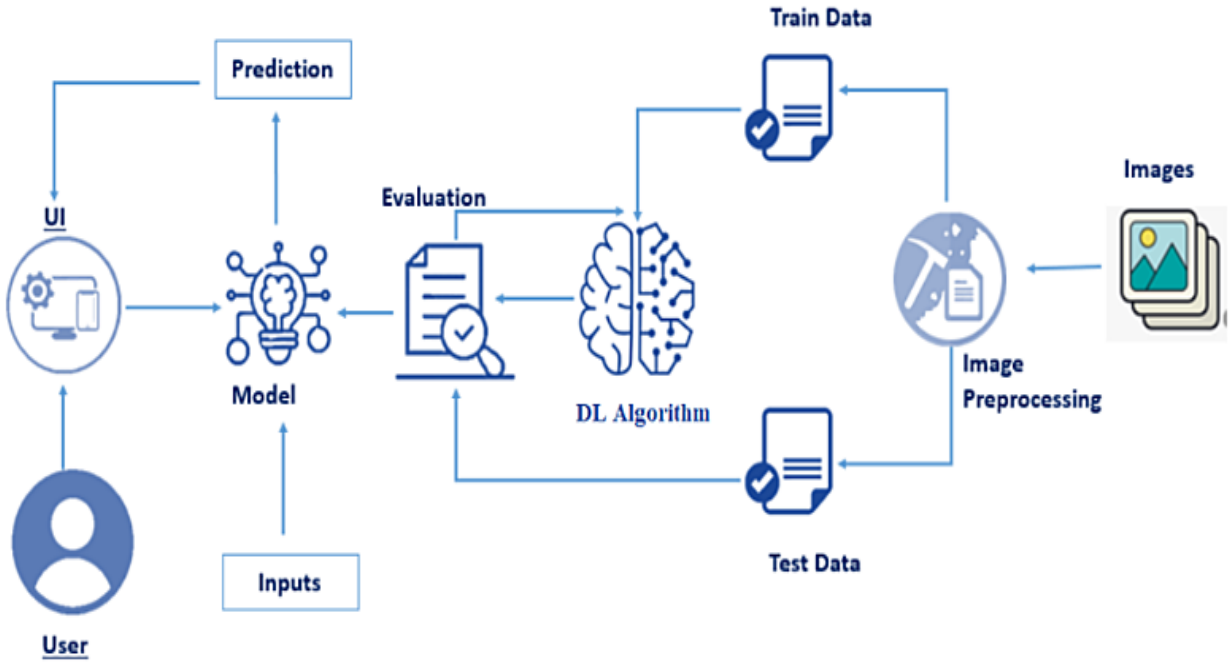
Fertilizers Recommendation System for Disease Prediction



Different approaches for the identification of leaf diseases



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 by entering my email, phone number, date of birth, password and confirm password	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation messagein my email once I have registered or OTP will be sent	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register through Gmail	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can log in by entering email & password	The system can authenticate me and I cantrust it.	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password.	The system can remember me and save the data.	High	Sprint-1
	Dashboard	USN-6	Enter the password and mail ID to login the dashboard.	To view the updates and what are the changes are to be done.	Medium	Sprint-1

Customer (Web user)	Forgot password	USN-1	Suppose a user forgot password by clicking forgot password and OTP send to my number or mail.	By entering the OTP sent via phone number or email.	High	Sprint -1
Customer Care Executive	Professional responsible.	USN-1	As a customer care executive, I'm the responsible for communicating the how's and why's regarding service exceptions within a company.	Answering phones, responding to customer questions and assisting with customer issues.	High	Sprint -1
Administrator	Login	USN-1	As an admin I can login by using mail id and password.	Preparing, organizing and storing information in paper and digital form.	High	Sprint -1
	Data collection	USN-2	As an admin,I can upload the data set to train the device.	Dealing with queries on thephone and byemail. Arranging post and deliveries	High	Sprint -1

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Milestone Name	Milestone Number	Description	Mandatory	Optional
Project Objectives	M-001	We will be able to learn to prepare dataset, image processing, working with CNN layers, read images using Open CV and CNN for computer vision AI	Yes	-
Project Flow	M-002	A project management process flowchart is a graphical aid, designedto visualize the sequence of steps to be followed throughout the project management process	Yes	
Pre-Requisites	M-003	To complete this project we should have known following project such as Keras, Tensor flow, Python, Anaconda, Open CV, Flask, Scikit-learn etc...	Yes	
Prior Knowledge	M-004	One should have knowledge on the Supervised Learning ,CNN and Regression Classification and Clustering, ANN	Yes	
Data collection	M-005	We can collect dataset from different open sources like kaggle.com, UCI machine learning etc	Yes	
Image Preprocessing	M-006	Importing the Image Data Generator libraries, Define Parameters/Arguments for Image Data Generator class, Applying Image Data Generator Functionality to train set and test set.	Yes	
Model Building	M-007	Importing the model building libraries, Initializing the model, Adding CNN layers, Adding Dense layers, Configuring the learning Process, Train the model, Save the model, Predictions.	Yes	
Video Analysis	M-008	Open cv for video processing, creating an account in twilio service andsending alert message	Yes	
Train CNN model	M-009	Register for IBM Cloud and train Image Classification Model	Yes	
Ideation Phase	M-010	Prepare Literature Survey on the selected Project and Information Gathering, empathy map and ideation	Yes	
Project Design Phase-I	M-011	Prepare Proposed solution , problem-solution fit and Solution Architecture	Yes	
Project Design Phase-II	M-012	Prepare Customer journey ,functional requirements, Data flow diagram andTechnology Architecture	Yes	
Project Planning Phase	M-013	Prepare Milestone list , Activity list and Sprint Delivery Plan	Yes	
Project Development Phase	M-014	Project Development delivery of Sprint 1, Sprint 2, Sprint 3, Sprint 4	Yes	

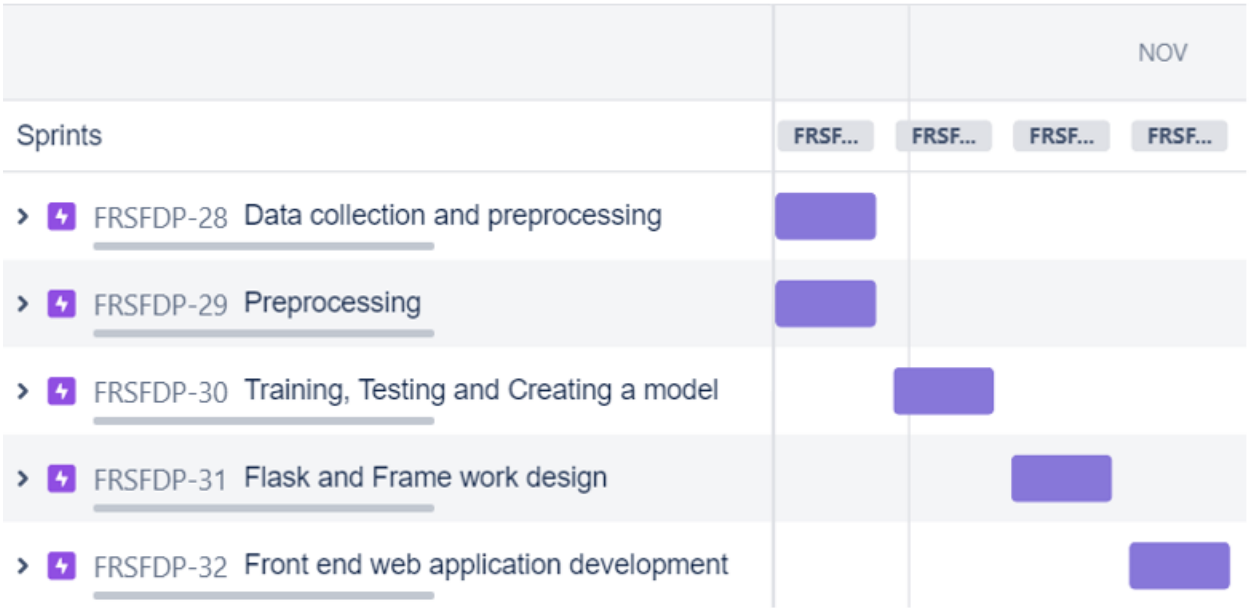
6.2 SPRINT DELIVERY SCHECDULE

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Member
Sprint-1	Data collection and preprocessing	USN-1	Collecting plant disease dataset	2	Low	Narmatha N
Sprint-1		USN-2	Labelling the dataset according to class	3	Medium	Nandhini M
Sprint-1		USN-3	38 types of plant diseases is labeled accordingly	2	Medium	Dharshini R
Sprint-1		USN-4	Data set Will contain both healthy and diseased data	1	Low	Dharani S
Sprint-1	Preprocessing	USN-5	To prepare raw data in a format that the network can accept	2	High	Nandhini M
Sprint-1		USN-7	Shear range image will be distorted along an axis, mostly rectify the perception angle	3	High	Dharshini R Narmatha N
Sprint-1		USN-8	Zoom Augment will randomly zoom the image adds new pixels for the image	3	High	Nandhini M Dharani S
Sprint-1		USA-9	Flipping the entire pixels of an image horizontally	3	High	Dharshini R Dharani S
Sprint-2	Training, Testing and Creating model	USN-10	Start initiating the model	3	Medium	Nandhini M Dharshini R
Sprint-2		USN-11	Adding different layers of CNN (convolution, pooling dense, flatten)	2	Medium	Dharani S
Sprint-2		USN-12	Creating/compiling with Adam optimizer	1	Medium	Narmatha N
Sprint-2		USN-13	Kera's Categorical Cross Entropy Loss Function for multi-class classification	2	Medium	Dharshini R

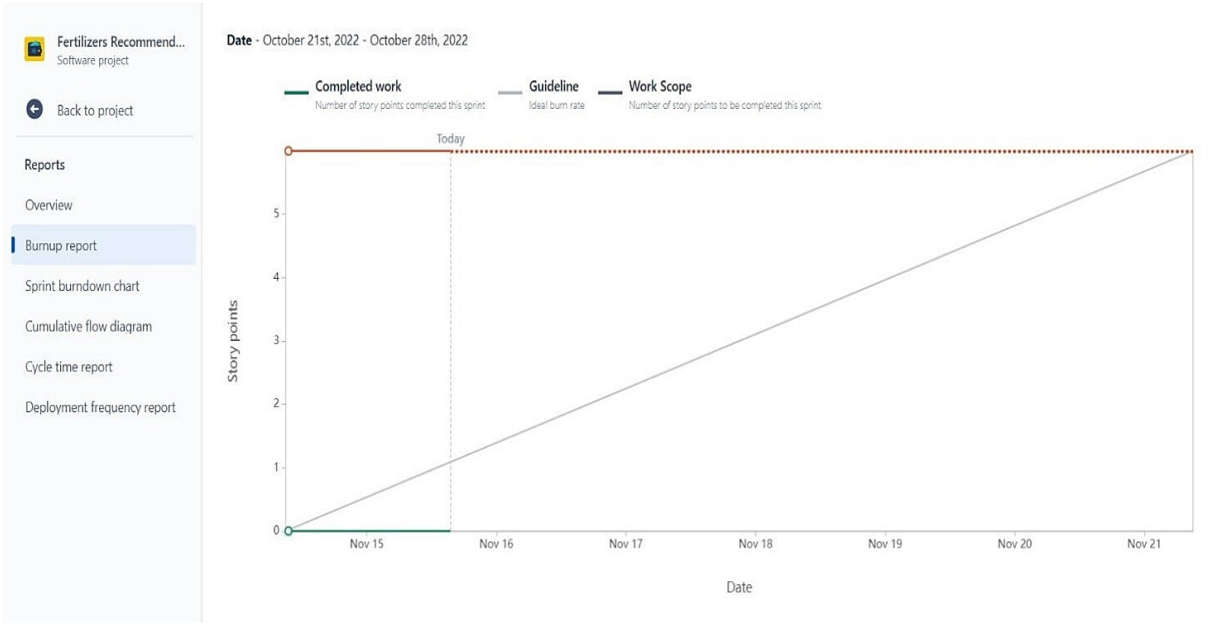
Sprint-2		USN-14	creating metrics	2	Medium	Nandhini M
Sprint-2		USN-15	train the data with 20epoch	3	High	Narmatha N Dharshini R
Sprint-2		USN-16	testing the model	5	High	Dharani S Dharshini R Nandhini M
Sprint-2		USN-17	save the model	2	Medium	Narmatha N
Sprint-3	Flask and Frame work design	USN-18	Creating backend framework with flask	8	High	Dharani S Dharshini R Nandhini M
Sprint-3		USN-19	importing the model file	3	Medium	Dharshini R
Sprint-3		USN-20	Create route to link HTML Routes and View Functions in Flask Framework index file	5	High	Nandhini M Dharani S
Sprint-3		USN-21	Server Startup, requests and services in a loop	4	Medium	Narmatha N Dharshini R
Sprint-4	Front end web application development	USN-22	creating a html template with CSS file	8	High	Narmatha N Dharani S Dharshini R Nandhini M
Sprint-4		USN-23	user can import diseased plant leaf in web page	2	Medium	Narmatha N Dharani S Dharshini R Nandhini M
Sprint-4		USN-24	predicting what is the type of disease occurred for the given input	2	Medium	Dharshini R Dharani S
Sprint-4		USN-25	User can classify as healthy or diseased	2	Medium	Narmatha N Nandhini M
Sprint-4		USN-26	if plant has disease, then suggest fertilizer and pesticides	3	Medium	Dharshini R Nandhini M
Sprint-4		USN-27	alert the admin about the prediction with the Gmail	3	Medium	Narmatha N

6.3REPORTS FROM JIRA

RoadMap:

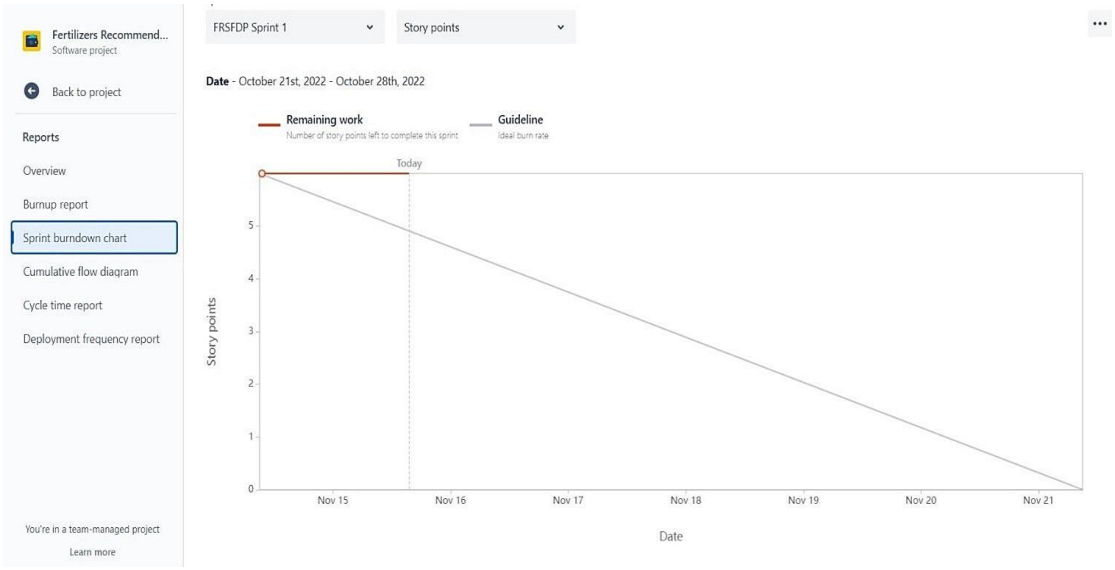


Burn up Chart:



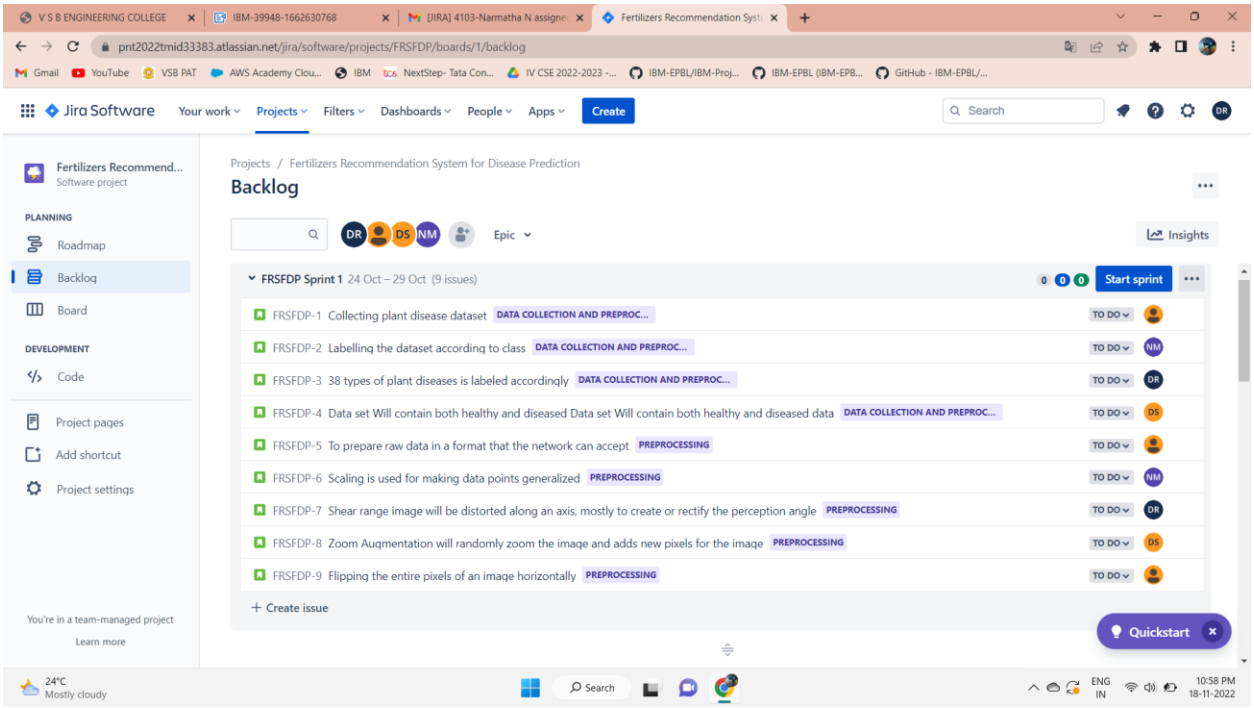
Burn down Chart:

A burndown chart **shows the amount of work that has been completed in anepic or sprint, and the total work remaining**. Burndown charts are used to predict your team's likelihood of completing their work in the time available.

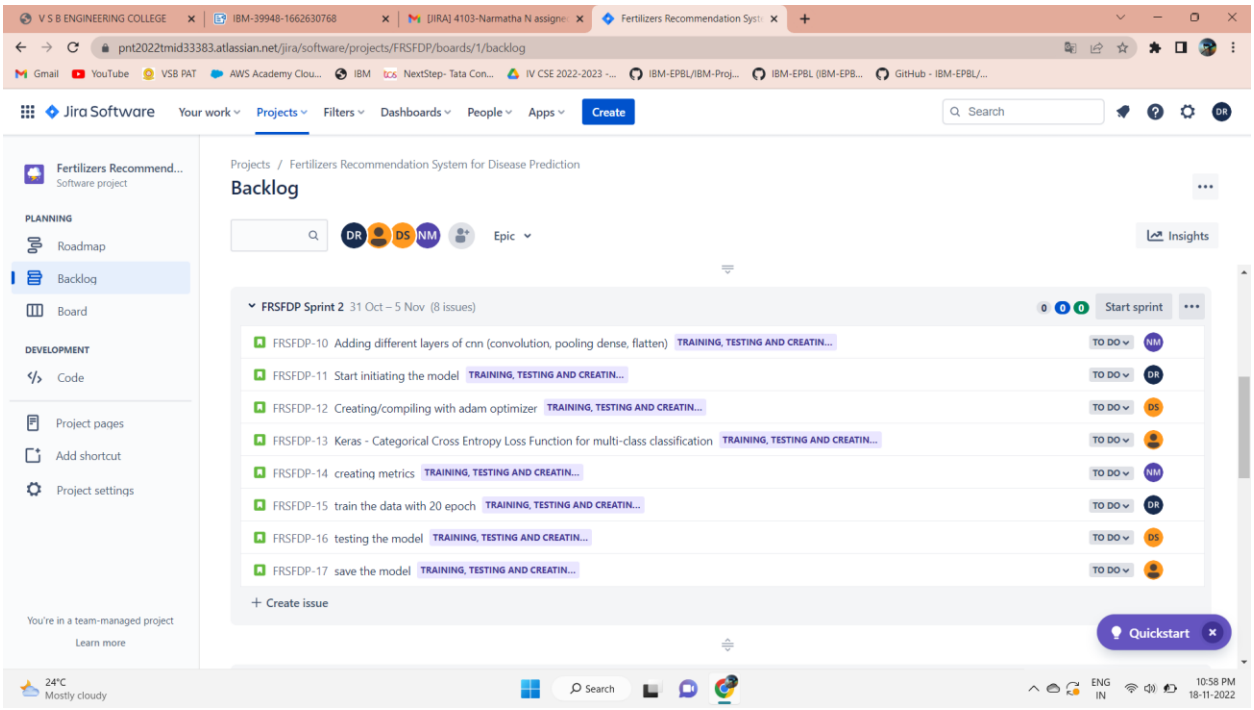


BACKLOG:

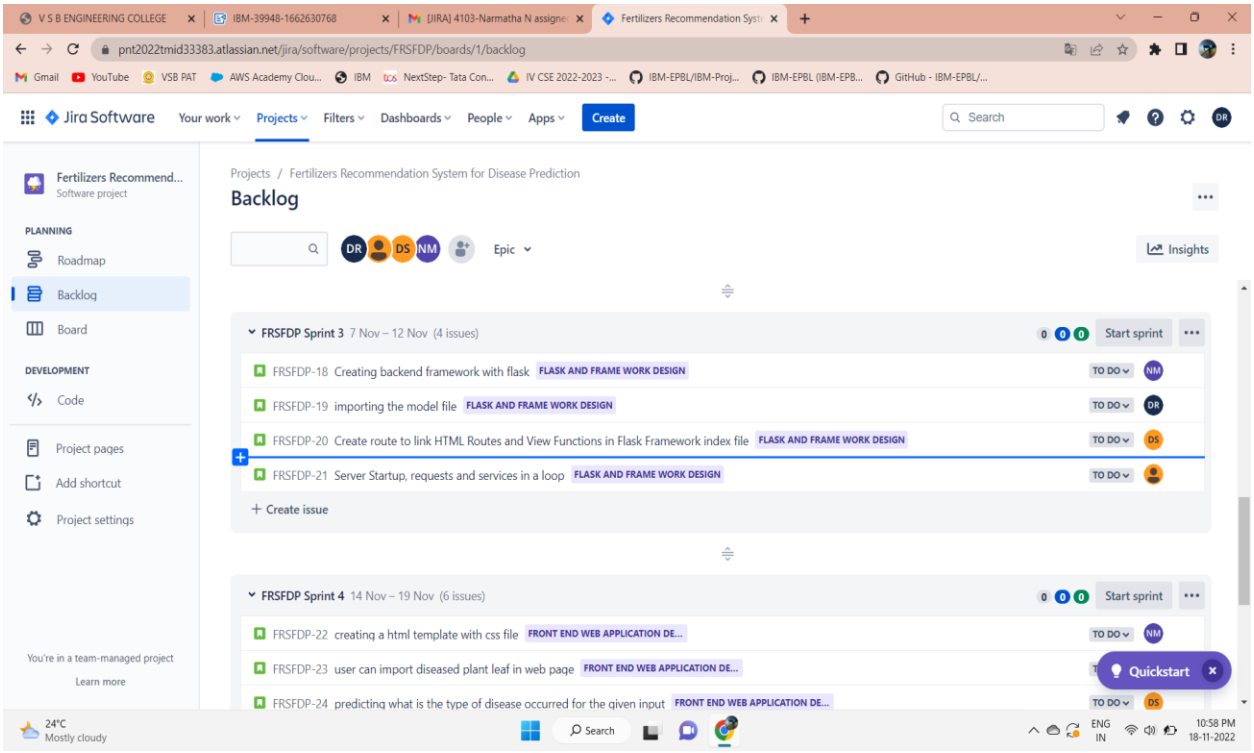
SPRINT 1:



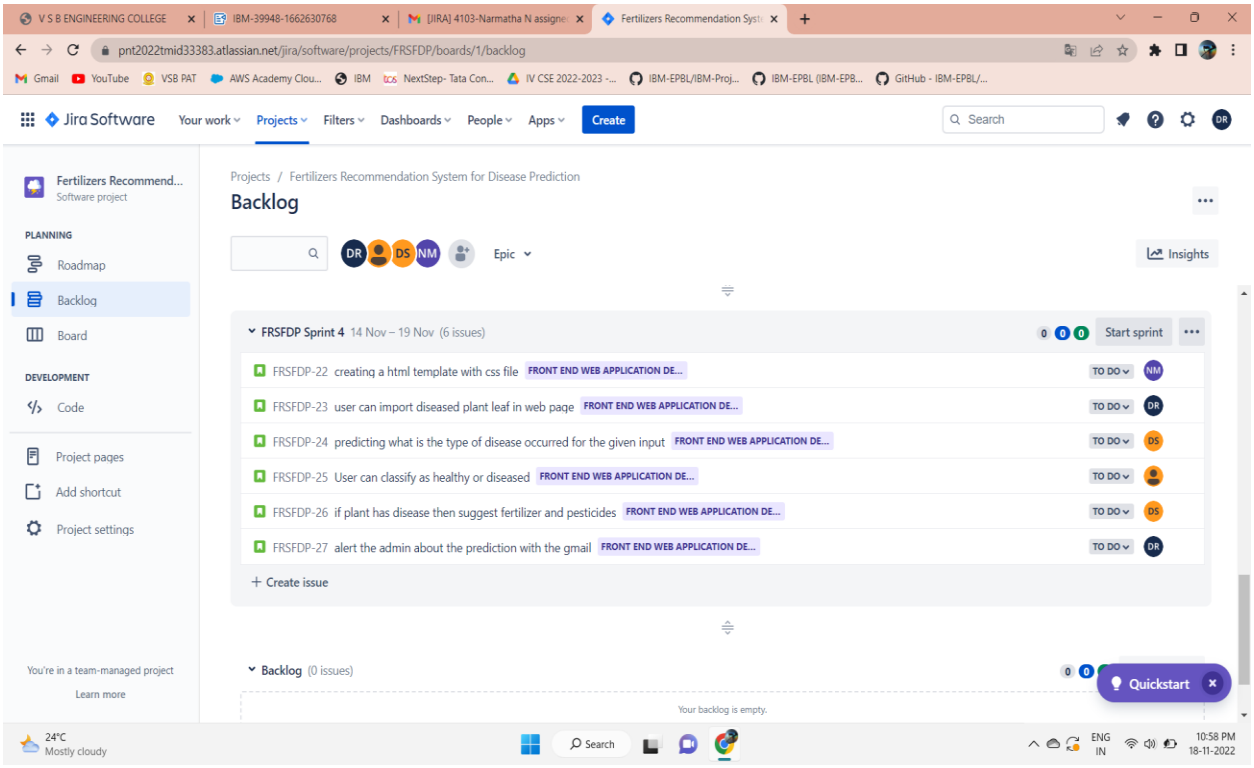
SPRINT 2



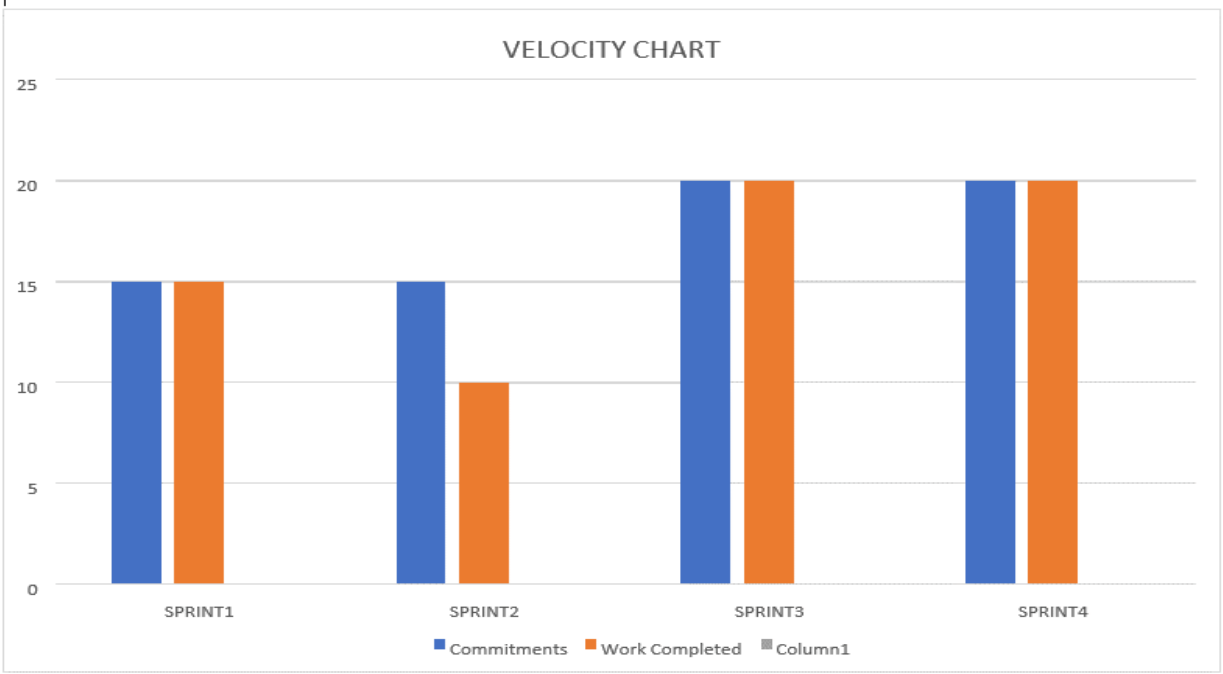
SPRINT 3



SPRINT 4



Velocity Chart:



7.CODING & SOLUTION:

7.1 Feature 1:

In this project, we have created a page where the farmers can upload the pictures of the leaves. The page will have separate design for fruit and vegetable leaves. The concerned leaves are upload in the page. After uploading the leaves, if the leaves are not affected by any diseases, it will indicate that the plant is in good condition and the farmers can follow the same techniques. Suppose if the leaves are affected by certain bacterial, fungal or any other diseases, it will predict the type of disease and recommend the suitable fertilizer to plants, so that the disease affected plants get cured and will give more yield to the farmers.

7.2 Feature 2:

The trained machine learning model can predict the output from an image that is uploaded, and the nutrition facts are also displayed on the same page. The model's accuracy was determined to be 95%, and when it was trained on the IBM cloud, it reached 100%. 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.

7.3 Database Schema:

To design the web pages of the project, HTML is used. HTML is used structuring the webpage and CSS (Cascading Style Sheets) are used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes. For the backend development, python code is used for implementing the web page and developing the project.

8.TESTING

A test case has components that describe input, action and an expected response, in order to determine if a feature of an application is working correctly. A test case is a set of instructions on “HOW” to validate a particular test objective/target, which when followed will tell us if the expected behaviour’s of the system is satisfied or not.

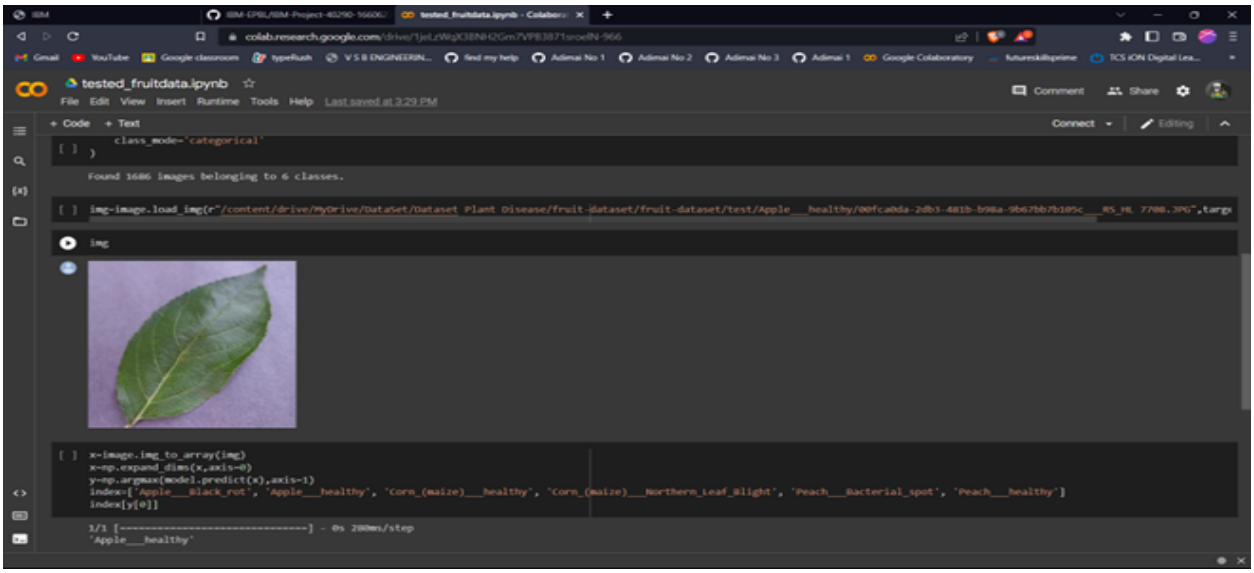
Characteristics of a good test case:

- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

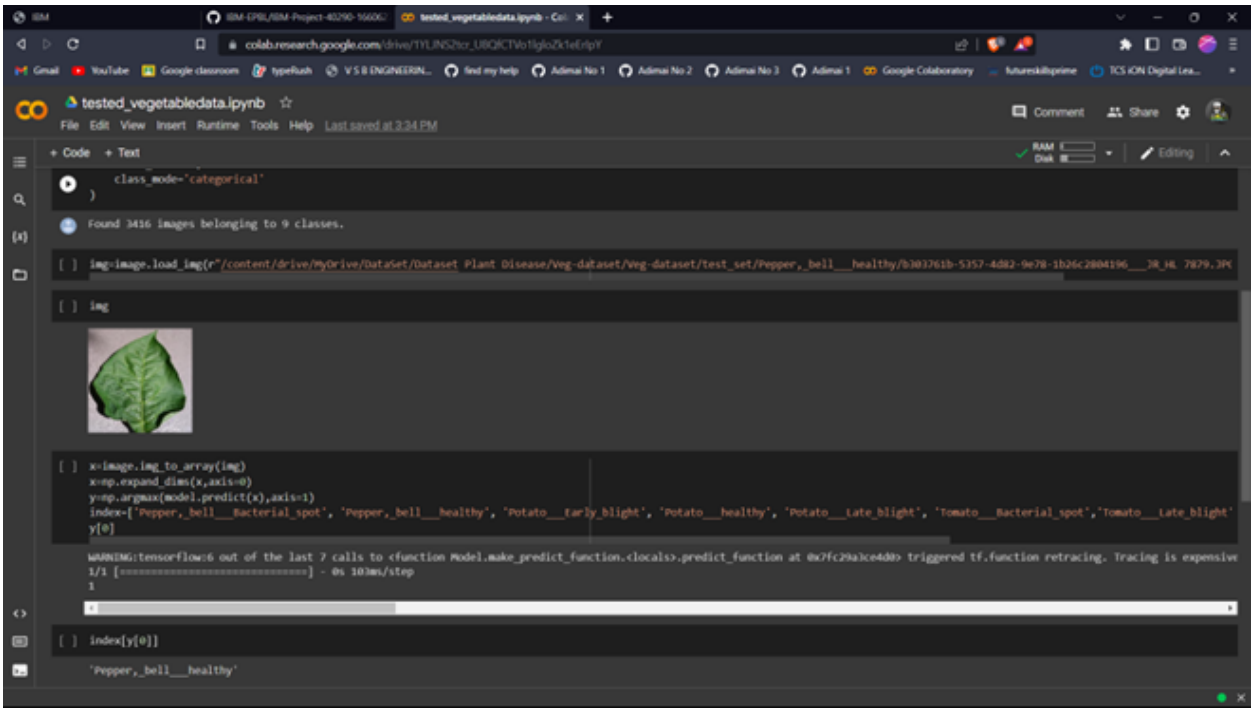
S.NO	Scenario	Input	Expected Output	Actual Output
1	Home Page	Predict	Introduction to the predict page	Predict page
2	Predict Page	Selection	Selecting the Leaf images	Leaf images selected successfully
3	Predict Page	Predicting	Predicting leaf healthy and giving precautions	Healthy of a Leaf predicted and given precautions successfully

8.1 Test Cases:

Tested Fruit Data:



Tested Vegetable Data:



8.2 User Acceptance Testing:

Defect Analysis:

This report shows the number of resolved or closed bugs at severity level, and how they were resolved. The purpose of this document is to briefly explain the test coverage and open issues ofthe [Fertilizer Recommendation system for plant disease prediction] project at the time of the release to User Acceptance Testing (UAT).

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Sub Total
Yellow Leaves	10	4	5	15	34
Blights	1	5	2	4	12
Fruit rots	3	1	0	2	6

Leaf spots	9	2	4	18	33
Mosaic leaf pattern	3	9	6	6	24
Fruit Spots	3	1	5	1	10
Leaves misshapen	0	7	2	1	10
Totals	29	29	24	47	129

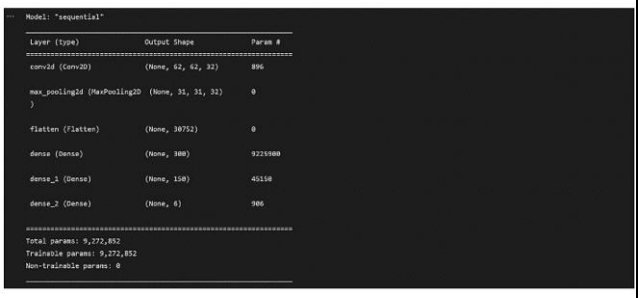
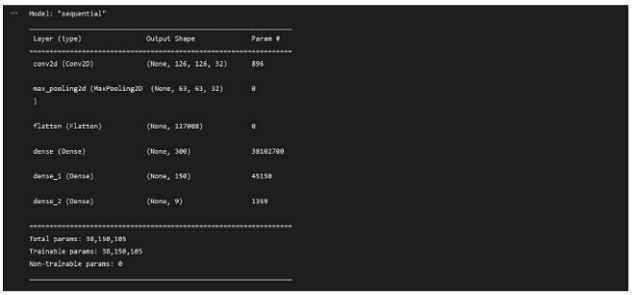
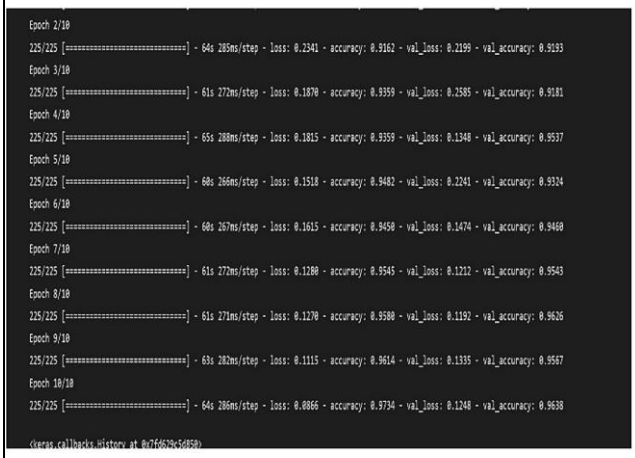
Test Case Analysis:

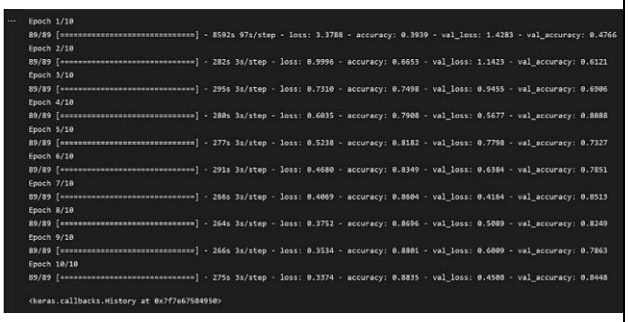
This report shows the number of test cases that have passed, failed, and untested.

Section	Total Cases	Not Tested	Fail	Pass
Yellow Leaves	20	0	0	20
Blights	43	0	0	43
Fruit rots	9	0	0	9
Leaf spots	5	0	0	5
Mosaic leaf pattern	19	0	0	19
Fruit Spots	2	0	0	2
Leaves misshapen	4	0	0	4

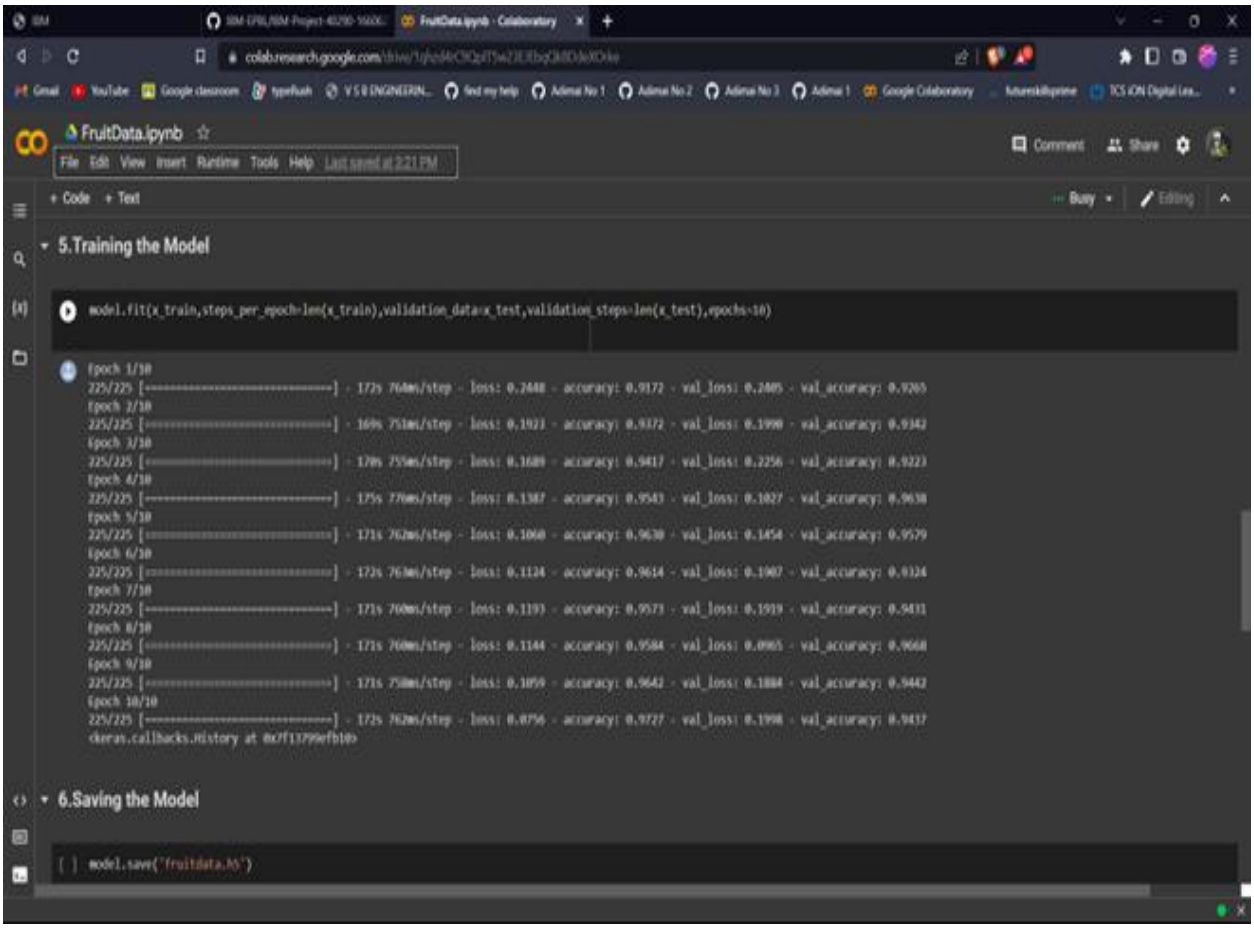
9.RESULT

9.1 PERFORMANCE METRICS

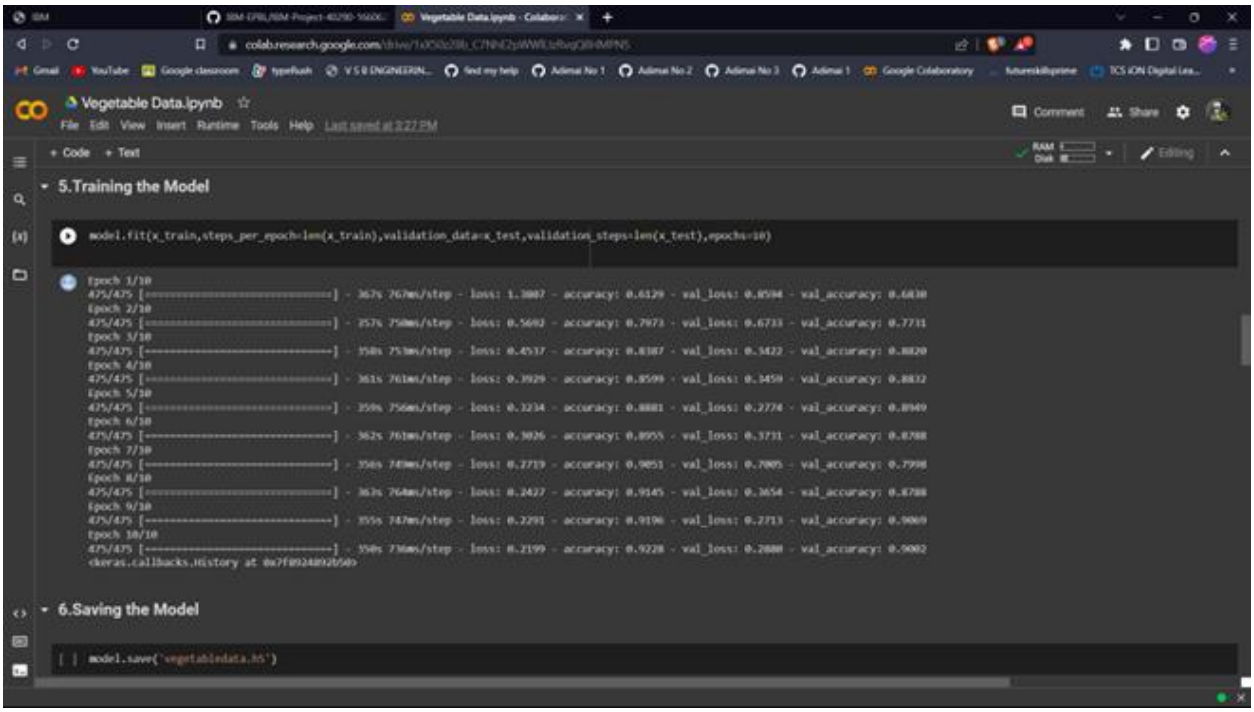
S. No.	Parameter	Values	Screenshot
1.	Model Summary ofFruit	Training the dataset of Vegetable images by usingthe CNN models to predict the disease of the given leaves.	
2.	Model Summary for Vegetable	Training the dataset of Vegetable images by usingthe CNN models to predict the disease of the given leaves.	
4.	Accuracy forFruit	Training Accuracy - 0.9734 Validation Accuracy - 0.9638	

4.	Accuracy for Vegetable	Training Accuracy - 0.8835 Validation Accuracy - 0.8448	

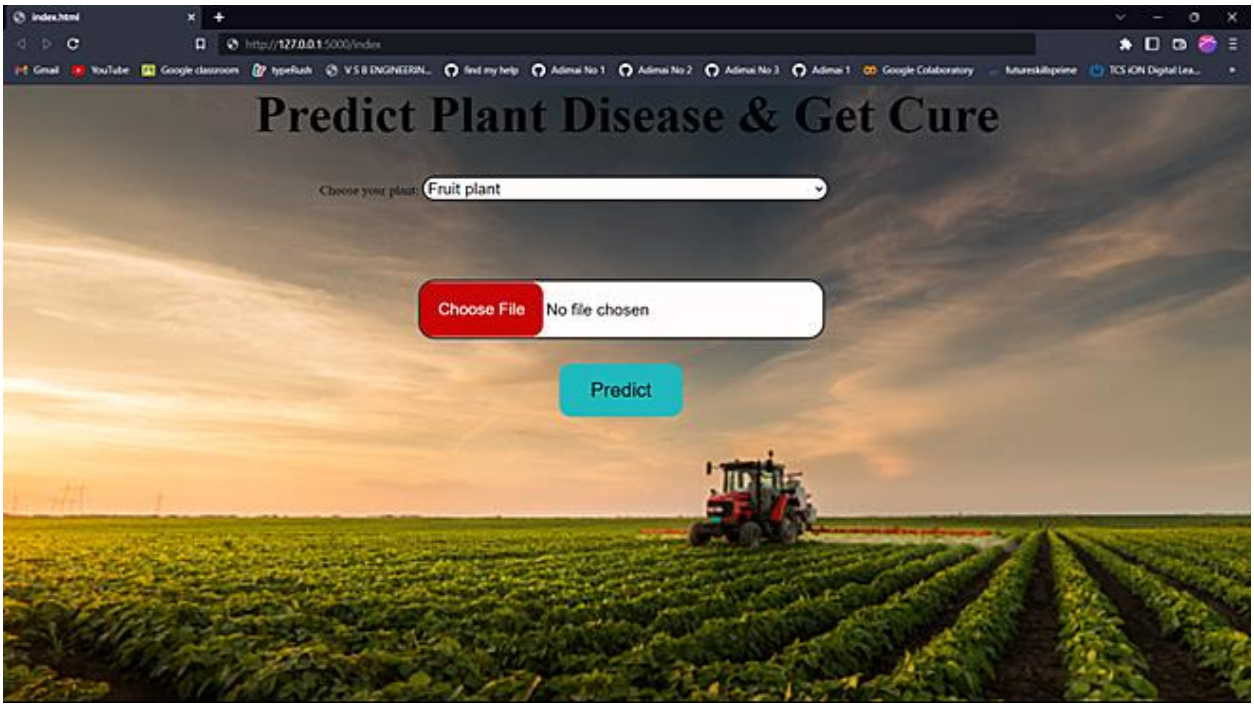
Training Fruits Dataset:



Training Vegetable Dataset:



Login Page



Predicting the Fruit Leaf:

Predicting the Vegetable Leaf:

10. ADVANTAGES AND 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.
- The suggested model yields extremely high classification accuracy
- It can train and test on very large datasets.
- It can resize very high-quality images within 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.
- The proposed model is computationally expensive to train and test.
- The neural network architecture used in this project work is highly complex.

11. CONCLUSION:

The model proposed here involves image classification of fruit datasets and vegetable datasets. The following points are observed during model testing and training:

- The accuracy of classification increased by increasing the number of epochs.
- For different batch sizes, different classification accuracies are obtained.
- The accuracies are increased by increasing more convolution layers.
- The accuracy of classification also increased by varying dense layers.
- Different accuracies are obtained by varying the size of kernel used in the convolution layer output.
- Accuracies are different while varying the size of the train and test datasets.

The model here involves classifying images from datasets of fruits and vegetables. The number of epochs was increased to boost categorization accuracy. Different classification accuracies are obtained for different batch sizes. The accuracies are increased by adding more convolution layers. The accuracy of classification is also increased by adjusting the number of dense layers. The accuracies are different while varying the size of the train and test datasets.

12. FUTURE SCOPE:

The model that is being provided in this project work can be expanded to recognize images. Using python to exe software, the complete model may be turned into application software. With the aid of the OpenCV Python package, real-time image categorization, picture recognition, and video processing are all made feasible. This project's work can be expanded to include security applications including face, iris, and figure print recognition. The system successfully interprets various Diseases and is also capable of providing fertilizers suggestion for the respective disease. Furthermore, this system can be made more robust by incorporating more image dataset with wider variations like morethan one leaf in a single image. An App could also be developed for the project whichcould 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. This project is limited to just one crop for now but in the future more crops and even flowers dataset can be added so that it is help full for every agriculturalneed. Newer models can also be added and tried with time which may result in betteraccuracy and would make the model even faster.

13. APPENDIX:

Source Code:

1_Histogram.ipynb

```
import cv2
import matplotlib.pyplot as plt
import numpy as np
img = cv2.imread('/content/drive/MyDrive/0a285c8b-1c31-48d4-89f2-
af8b9edc36f6___RS_HL 5759.JPG', 0)
plt.imshow(img, cmap='gray'), plt.grid(False)
plt.xticks([]), plt.yticks([])
hist = cv2.calcHist([img],[0],None,[50],[0,256])

# different methods for displaying a histogram
plt.bar(range(50), hist.ravel())
plt.title('Histogram of the airplane image')
plt.xlabel('Gray values')
plt.ylabel('Frequency')
# Another method
hist,bins = np.histogram(img.ravel(),256,[0,256])
plt.plot(hist)
# Let's read two other images
```

```
high = cv2.imread('/content/drive/MyDrive/aa04db6d-645f-4e8a-88dc-
c9f9396f0800___RS_HL 1975.jpg')
low = cv2.imread('/content/drive/MyDrive/0a285c8b-1c31-48d4-89f2-
af8b9edc36f6___RS_HL 5759.JPG')
# show images
plt.subplot(121), plt.imshow(high)
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(122), plt.imshow(low)
plt.grid(False), plt.xticks([]), plt.yticks([])
plt.show()
# Calculate histogram of both images for the last channel.
# Channels can differ from 0 to 2.
hist_high = cv2.calcHist([high],[2],None,[256],[0,256])
hist_low = cv2.calcHist([low],[2],None,[256],[0,256])

# Plot histograms
plt.subplot(121)
plt.plot(hist_high)

plt.subplot(122)
plt.plot(hist_low)

plt.show()
cdf_low = hist_low.cumsum()
cdf_high = hist_high.cumsum()

# plot cumulative histograms
plt.subplot(221), plt.plot(cdf_high), plt. title('cdf of bright image')
plt.subplot(222), plt.plot(hist_high, 'k'), plt. title('pdf of bright image')

plt.subplot(223), plt.plot(cdf_low), plt. title('cdf of dark image')
plt.subplot(224), plt.plot(hist_low, 'k'), plt. title('pdf of dark image')

# adjust the placement of subplots
plt.subplots_adjust(bottom=2, right=0.8, top=3)

plt.show()
low_gray = cv2.cvtColor(low, cv2.COLOR_BGR2GRAY)
high_gray = cv2.cvtColor(high, cv2.COLOR_BGR2GRAY)
# show images and their histograms
```

```
plt.subplot(221), plt.imshow(high_gray, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])
plt.subplot(223), plt.plot(cv2.calcHist([high_gray],[0],None,[256],[0,256]))

plt.subplot(222), plt.imshow(low_gray, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])
plt.subplot(224), plt.plot(cv2.calcHist([low_gray],[0],None,[256],[0,256]))

plt.show()
# Define a function to easily handle manipulation.
def manip_image(image, alpha, beta):

  new_image = np.zeros(image.shape, image.dtype)

  for y in range(image.shape[0]):
    for x in range(image.shape[1]):
        new_image[y,x] = np.clip(alpha*image[y,x] + beta, 0, 255)

  return new_image
# Test on the image
bright = manip_image(img, 1, 30)
dark = manip_image(img, 1, -30)

# Compare the results
plt.figure()
plt.subplot(231), plt.imshow(dark, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(232), plt.imshow(img, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])


plt.subplot(233),plt.imshow(bright, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(234)
plt.plot(cv2.calcHist([dark],[0],None,[256],[0,256])), plt.ylim((0, 1750))

plt.subplot(235)
plt.plot(cv2.calcHist([img],[0],None,[256],[0,256]))

plt.subplot(236)
```

```
plt.plot(cv2.calcHist([bright],[0],None,[256],[0,256]))
# Test on the dark image
l_bright = manip_image(low_gray, 1, 150)
l_dark = manip_image(low_gray, 1, -25)

# Compare the results
plt.figure()
plt.subplot(231), plt.imshow(l_dark, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(232), plt.imshow(low_gray, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])


plt.subplot(233),plt.imshow(l_bright, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(234)
plt.plot(cv2.calcHist([l_dark],[0],None,[256],[0,256])), plt.ylim((0, 1100))

plt.subplot(235)
plt.plot(cv2.calcHist([low_gray],[0],None,[256],[0,256])), plt.ylim((0, 1100))

plt.subplot(236)
plt.plot(cv2.calcHist([l_bright],[0],None,[256],[0,256])), plt.ylim((0, 1100))
# Test on the image
increase_contrast = manip_image(img, 1.35, 0)
decrease_contrast = manip_image(img, 0.35, 0)

# Compare the results
plt.figure()
plt.subplot(231), plt.imshow(decrease_contrast, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(232), plt.imshow(img, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])


plt.subplot(233),plt.imshow(increase_contrast, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(234)
```

```
plt.bar(range(256),
        cv2.calcHist([decrease_contrast],[0],None,[256],[0,256]).ravel())
plt.ylim((0, 1750))

plt.subplot(235)
plt.bar(range(256),
        cv2.calcHist([img],[0],None,[256],[0,256]).ravel())
plt.ylim((0, 1750))

plt.subplot(236)
plt.bar(range(256),
        cv2.calcHist([increase_contrast],[0],None,[256],[0,256]).ravel())
plt.ylim((0, 1750))
img_eq = cv2.equalizeHist(img)

grid = plt.GridSpec(3, 4, wspace=0.4, hspace=0.3)

plt.subplot(grid[:2, :2])
plt.imshow(img, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(grid[:2, 2:])
plt.imshow(img_eq, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(grid[2, :2])
plt.bar(range(256),
        cv2.calcHist([img],[0],None,[256],[0,256]).ravel())

plt.subplot(grid[2, 2:])
plt.bar(range(256),
        cv2.calcHist([img_eq],[0],None,[256],[0,256]).ravel())
clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(16, 16))
img_cl = clahe.apply(img)

grid = plt.GridSpec(3, 4, wspace=0.4, hspace=0.3)

plt.subplot(grid[:2, :2])
plt.imshow(img, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(grid[:2, 2:])
```

```
plt.imshow(img_cl, cmap='gray')
plt.grid(False), plt.xticks([]), plt.yticks([])

plt.subplot(grid[2, :2])
plt.bar(range(256),
        cv2.calcHist([img],[0],None,[256],[0,256]).ravel())

plt.subplot(grid[2, 2:])
plt.bar(range(256),
        cv2.calcHist([img_cl],[0],None,[256],[0,256]).ravel())
```

2_Spatial_Filtering.ipynb

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
ref = cv2.imread('/content/drive/MyDrive/0a285c8b-1c31-48d4-89f2-
af8b9edc36f6___RS_HL 5759.JPG')
plt.imshow(ref), plt.grid(False)

# while learning how to perform spatial filtering,
# you can also note how to apply different python commands.
plt.title('The original image')
plt.xticks([])
plt.yticks([])
plt.show()
# Defining a kernel using numpy.
kernel_5 = np.ones((5,5),np.float32)/25
kernel_3 = np.ones((3,3),np.float32)/9

# Convolves an image with the kernel.
# -1 means that the center of the kernel is located on the center pixel.
# compare two kernel sizes.
filtered_5 = cv2.filter2D(ref,-1,kernel_5)
filtered_3 = cv2.filter2D(ref,-1,kernel_3)

# plot the results in two subplots.
fig=plt.figure(figsize=(14, 14), dpi= 80, facecolor='w', edgecolor='k')

plt.subplot(121), plt.imshow(filtered_3), plt.title('3-by-3 filter')
```

```
plt.grid(False)
plt.xticks([])
plt.yticks([])

plt.subplot(122), plt.imshow(filtered_5), plt.title('5-by-5 filter')
plt.grid(False)
plt.xticks([])
plt.yticks([])

plt.show()
# you can check the docs for further information.
blurred = cv2.blur(ref, (5, 5), -1)

plt.imshow(blurred), plt.grid(False), plt.xticks([]), plt.yticks([]), plt.show()
top = 10; bottom = 5; left = 20; right = 5
const = 100
img2 = cv2.copyMakeBorder(ref, top, bottom, left, right,
                          cv2.BORDER_WRAP)
img3 = cv2.copyMakeBorder(ref, top, bottom, left, right,
                          cv2.BORDER_REFLECT)
img4 = cv2.copyMakeBorder(ref, top, bottom, left, right,
                          cv2.BORDER_REPLICATE)
img5 = cv2.copyMakeBorder(ref, top, bottom, left, right,
                          cv2.BORDER_CONSTANT, const)

# Display the images
fig=plt.figure(figsize=(14, 14), dpi= 80, facecolor='w', edgecolor='k')

plt.subplot(221), plt.imshow(img2), plt.grid(False)
plt.xticks([]), plt.yticks([]), plt.title('wrap')

plt.subplot(222), plt.imshow(img3), plt.grid(False)
plt.xticks([]), plt.yticks([]), plt.title('reflect')

plt.subplot(223), plt.imshow(img4), plt.grid(False)
plt.xticks([]), plt.yticks([]), plt.title('replicate')

plt.subplot(224), plt.imshow(img5), plt.grid(False)
plt.xticks([]), plt.yticks([]), plt.title('constant')

plt.show()
```

```
# vertical gradient kernel
# define a random kernel
vertical_gd = np.array([[1, 0, -1], [1, 0, -1], [1, 0, -1]])

# apply it.
filter_v = cv2.filter2D(ref[:, :, 2], -1, vertical_gd)

# show in a different colormap.
plt.imshow(filter_v, cmap='gray'), plt.grid(False)
plt.xticks([]), plt.yticks([])
s_and_p = np.random.rand(ref.shape[0], ref.shape[1])

# if we consider 5% salt and pepper noise, we'd like to have
# 2.5% salt and 2.5% pepper. thus:
salt = s_and_p > .975
pepper = s_and_p < .025

# in order to add some noise, we should turn off black (pepper) locations and
# turn on white (white) locations.
channel_2 = np.atleast_1d(ref[:, :, 1])
noisy = np.zeros_like(channel_2)

for i in range(channel_2.shape[0]*channel_2.shape[1]):
  if salt.ravel()[i] == 1:
    noisy.ravel()[i] = 255
  elif pepper.ravel()[i] == 1:
    noisy.ravel()[i] = 0
  else:
    noisy.ravel()[i] = channel_2.ravel()[i]

# apply median filter with size 3
Med = cv2.medianBlur(noisy, 3)

# Display the results
fig=plt.figure(figsize=(14, 14), dpi= 80, facecolor='w', edgecolor='k')
plt.subplot(121), plt.xticks([]), plt.yticks([])
plt.imshow(noisy, cmap='gray'), plt.grid(False)
plt.subplot(122), plt.xticks([]), plt.yticks([])
plt.imshow(Med, cmap='gray'), plt.grid(False)
plt.show()
# Creating random normal (gaussian) noise with pre-defined mean and std.
```

```
# The noisy image should be the size of the reference image.
mean = 0
sigma = 20.0
gauss_noise = np.random.normal(mean, sigma, (ref.shape[0], ref.shape[1]))

# Convert RGB image to Grayscale image using cvtColor()
gray = cv2.cvtColor(ref, cv2.COLOR_BGR2GRAY)

# Add gaussian noise to the image
g_noisy = gray + gauss_noise # Gaussian noisy image

# Showing gray image, noise image, and noisy image
fig=plt.figure(figsize=(14, 14), dpi= 80, facecolor='w', edgecolor='k')
plt.subplot(131), plt.xticks([]), plt.yticks([])
plt.imshow(gray, cmap='gray'), plt.grid(False)
plt.subplot(132), plt.xticks([]), plt.yticks([])
plt.imshow(gauss_noise, cmap='gray'), plt.grid(False)
plt.subplot(133), plt.xticks([]), plt.yticks([])
plt.imshow(g_noisy, cmap='gray'), plt.grid(False)
g_filtered = cv2.GaussianBlur(g_noisy, (3, 3), 20, 20)

# Display the result
plt.imshow(g_filtered, cmap='gray'), plt.grid(False)
plt.xticks([]), plt.yticks([])
# Create a single gaussian kernel
g_kernel = cv2.getGaussianKernel(3, 20)
print(g_kernel)

# Apply two separate kernels over the image.
g_filtered_2 = cv2.sepFilter2D(g_noisy, -1, g_kernel, g_kernel)

# Displaying the results.
fig=plt.figure(figsize=(14, 14), dpi= 80, facecolor='w', edgecolor='k')
plt.subplot(121), plt.xticks([]), plt.yticks([]), plt.title('first method')
plt.imshow(g_filtered, cmap='gray'), plt.grid(False)
plt.subplot(122), plt.xticks([]), plt.yticks([]), plt.title('second method')
plt.imshow(g_filtered_2, cmap='gray'), plt.grid(False)
```

3_Intensity_Transformations.ipynb

```
from skimage.io import imread
import matplotlib.pyplot as plt
import numpy as np
live = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/0b37761a-de32-47ee-a3a4-e138b97ef542___JR_FrgE.S 2908.JPG')
mask = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/00e909aa-e3ae-4558-9961-336bb0f35db3___JR_FrgE.S 8593.JPG')

plt.figure(figsize=(10, 10))
plt.subplot(121), plt.imshow(live, cmap='gray')
plt.subplot(122), plt.imshow(mask, cmap='gray')
plt.show()
plt.figure(figsize=(10, 10))
plt.subplot(121), plt.imshow(live, cmap='gray')
plt.subplot(122), plt.imshow(live - 20, cmap='gray')
plt.show()
plt.figure(figsize=(10, 10))
plt.subplot(131), plt.imshow(mask - live, cmap='gray')
plt.subplot(132), plt.imshow(-(mask - live + 128), cmap='gray')
plt.subplot(133), plt.imshow(mask - live + 128, cmap='gray')
plt.show()
shaded = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/0b37761a-de32-47ee-a3a4-e138b97ef542___JR_FrgE.S 2908.JPG')
shading = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/00e909aa-e3ae-4558-9961-336bb0f35db3___JR_FrgE.S 8593.JPG')

plt.figure(figsize=(10, 10))
plt.subplot(121), plt.imshow(shaded, cmap='gray')
plt.subplot(122), plt.imshow(shading, cmap='gray')
plt.show()
plt.figure(figsize=(10, 10))
plt.subplot(121), plt.imshow(np.multiply(shaded, 1/shading), cmap='gray')
plt.subplot(122), plt.imshow(shaded, cmap='gray')
plt.show()
```

```
# Test on the X-ray dental image
xray = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/00e909aa-e3ae-4558-9961-336bb0f35db3___JR_FrgE.S 8593.JPG')
mask_xray = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Peach___Bacterial_spot/00ddc106-692e-4c67-b2e8-569c924caf49___Rutg._Bact.S 1228.JPG')
plt.figure(figsize=(10, 10))
plt.subplot(121), plt.imshow(xray, cmap='gray')
plt.subplot(122), plt.imshow(mask_xray, cmap='gray')
plt.show()
plt.figure()
plt.imshow(np.multiply(xray, mask_xray/255), cmap='gray')
plt.show()
# Test on another image
scan = imread('/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Corn_(maize)___Northern_Leaf_Blight/0a62fe5a-22db-42e2-bca0-53a8dcfd8129___RS_NLB 0810.JPG')
print(scan.shape)
# Showing the body scan image
plt.figure(figsize=(7, 7))
plt.imshow(scan, cmap='gray')
plt.show()
plt.figure(figsize=(10, 10))
plt.subplot(211), plt.imshow(xray, cmap='gray')
plt.subplot(212), plt.plot(np.histogram(xray, bins=256)[0])
plt.show()
```

Image Preprocessing.ipynb

```
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)
x_train = train_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test',target_size = (128,128), batch_size = 32, class_mode = 'categorical')
```

```
x_test =
test_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/train',target_size = (128,128), batch_size = 32, class_mode = 'categorical')
x_train =
train_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/Veg-dataset/Veg-dataset/test_set',target_size = (128,128), batch_size = 32, class_mode = 'categorical')
x_test =
test_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/Veg-dataset/Veg-dataset/test_set',target_size = (128,128), batch_size = 32, class_mode = 'categorical')
```

TRAINING THE MODELS

FruitData.ipynb

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)
x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/train",target_size=(128,128),
                                          class_mode='categorical',batch_size=24)
x_test=test_datagen.flow_from_directory(r"/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test",target_size=(128,128),
                                          class_mode='categorical',batch_size=24)
model=Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()
32*(3*3*3+1)
model.add(Dense(300,activation='relu'))
```

```
model.add(Dense(150,activation='relu'))
model.add(Dense(6,activation='softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accurac
y'])
len(x_train)
model.fit(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_
steps=len(x_test),epochs=10)
model.save('fruitdata.h5')
```

Vegetable Data.ipynb

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_fli
p=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)
x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/DataSet/Dat
aset Plant Disease/Veg-dataset/Veg-dataset/train_set",target_size=(128,128),
                                          class_mode='categorical',batch_size=24)
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Datase
t Plant Disease/Veg-dataset/Veg-dataset/test_set',target_size=(128,128),
                                          class_mode='categorical',batch_size=24)
model=Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(9,activation='softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accurac
y'])
model.fit(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_
steps=len(x_test),epochs=10)
model.save('vegetabledata.h5')
```

TESTING THE MODELS

Tested_fruitdat.ipynb

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
test_dir=r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test'
model = tf.keras.models.load_model(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruitdata.h5')
test_datagen_1=ImageDataGenerator(rescale=1)
test_generator_1=test_datagen_1.flow_from_directory(
    test_dir,
    target_size=(128,128),
    batch_size=20,
    class_mode='categorical'
)
img=image.load_img(r"/content/drive/MyDrive/DataSet/Dataset Plant Disease/fruit-dataset/fruit-dataset/test/Apple___healthy/00fca0da-2db3-481b-b98a-9b67bb7b105c___RS_HL 7708.JPG",target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['Apple___Black_rot', 'Apple___healthy', 'Corn_(maize)___healthy', 'Corn_(maize)___Northern_Leaf_Blight', 'Peach___Bacterial_spot', 'Peach___healthy']
index[y[0]]
model.evaluate(test_generator_1,steps=50)
```

Tested_vegetabledata.ipynb

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import numpy as np
from tensorflow.keras.models import load_model
```

```
from tensorflow.keras.preprocessing import image
test_dir=r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/Veg-dataset/Veg-dataset/test_set'
model = tf.keras.models.load_model(r'/content/drive/MyDrive/DataSet/Dataset Plant Disease/Veg-dataset/vegetabledata.h5')
test_datagen_1=ImageDataGenerator(rescale=1)
test_generator_1=test_datagen_1.flow_from_directory(
    test_dir,
    target_size=(128,128),
    batch_size=20,
    class_mode='categorical'
)
img=image.load_img(r"/content/drive/MyDrive/DataSet/Dataset Plant Disease/Veg-dataset/Veg-dataset/test_set/Pepper,_bell___healthy/b303761b-5357-4d82-9e78-1b26c2804196___JR_HL 7879.JPG",target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['Pepper,_bell___Bacterial_spot', 'Pepper,_bell___healthy', 'Potato___Early_blight', 'Potato___healthy', 'Potato___Late_blight', 'Tomato___Bacterial_spot','Tomato___Late_blight','Tomato___Leaf_Mold','Tomato___Septoria_leaf_spot']
y[0]
index[y[0]]
```

APPLICATION BUILDING

app.py

```
#Import necessary libraries
from flask import Flask, render_template, request

import numpy as np
import os

from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
```

```
filepath =
'C:/Users/Anandh/AppData/Local/Programs/Python/Python38/Tomato_Leaf_Disea
se_Prediction/fruitdata.h5'
model = load_model(filepath)
print(model)

print("Model Loaded Successfully")

def pred_tomato_dieas(plant):
  test_image = load_img(plant, target_size = (128, 128)) # load image
  print("@@ Got Image for prediction")

  test_image = img_to_array(test_image)/255 # convert image to np array and
normalize
  test_image = np.expand_dims(test_image, axis = 0) # change dimention 3D to 4D

  result = model.predict(test_image) # predict diseased palnt or not
  print('@@ Raw result = ', result)

  pred = np.argmax(result, axis=1)
  print(pred)
  if plant=="fruit":
    if pred==0:
       return "Apple__Black_rot", 'Apple__Black_rot.html'

    elif pred==1:
       return "Apple__healthy", 'Apple__healthy.html'

    elif pred==2:
       return "Corn_(maize)__healthy", 'Corn_(maize)__healthy.html'

    elif pred==3:
       return "Corn_(maize)__Northern_Leaf_Blight",
'Corn_(maize)__Northern_Leaf_Blight.html'

    elif pred==4:
       return "Peach__Bacterial_spot", 'Peach__Bacterial_spot.html'

    elif pred==5:
       return "Peach__healthy", 'Peach__healthy.html'
  elif plant=="Vegetable":
    if pred==0:
```

```
        return "Pepper,_bell__Bacterial_spot", 'Pepper,_bell__Bacterial_spot.html'

    elif pred==1:
        return "Pepper,_bell__healthy", 'Pepper,_bell__healthy.html'

    elif pred==2:
        return "Potato__Early_blight", 'Potato__Early_blight.html'

    elif pred==3:
        return "Potato__healthy", 'Potato__healthy.html'

    elif pred==4:
        return "Potato__Late_blight", 'Potato__Late_blight.html'

    elif pred==5:
        return "Tomato__Bacterial_spot", 'Tomato__Bacterial_spot.html'

    elif pred==6:
        return "Tomato__Late_blight" , 'Tomato__Late_blight.html'

    elif pred==7:
        return "Tomato__Leaf_Mold" , 'Tomato__Leaf_Mold.html'

    elif pred==8:
        return "Tomato__Septoria_leaf_spot" , 'Tomato__Septoria_leaf_spot.html'



# Create flask instance
app = Flask(__name__)

# render index.html page
@app.route("/", methods=['GET', 'POST'])
def home():
        return render_template('index.html')


# get input image from client then predict class and render respective .html page
for solution
@app.route("/predict", methods = ['GET','POST'])
def predict():
```

```
    if request.method == 'POST':
        file = request.files['image'] # fet input
        filename = file.filename
        print("@@ Input posted = ", filename)

        file_path =
os.path.join('C:/Users/Madhuri/AppData/Local/Programs/Python/Python38/Tomat
o_Leaf_Disease_Prediction/static/upload/', filename)
        file.save(file_path)

        print("@@ Predicting class.....")
        pred, output_page = pred_tomato_dieas(tomato_plant=file_path)

        return render_template(output_page, pred_output = pred, user_image =
file_path)

# For local system & cloud
if __name__ == "__main__":
    app.run(threaded=False,port=8080)
```

HTML AND CSS CODE

index.html

```
<html>
<head>
<style>

* {
        margin: 0px;
        padding: 0px;
        box-sizing: border-box;


    }



    form {
        display: flex;
        height: 85vh;
```

```
    justify-content: center;
    align-items: center;
    margin-top: -150px;
    width: 60%;
    text-align: center;
                    margin-left:300px;

}


.details h2 {

    position: relative;
    top: 100px;
    margin: auto;
    color: rgb(18, 231, 231);
    font-size: 3rem;
}

label:hover {
    transform: scale(1.03);
}

.details h2 {
    /* margin-bottom: 300px; */
    position: relative;
    top: 100px;
    margin: auto;
    color: rgb(18, 231, 231);
    font-size: 3rem;

}



.details h1 {
    color: white;
    padding: 20px;
    border-radius: 15px;
    background-color: rgb(8, 8, 8);
}
```

```
.upload {

    font-size: 20px;
    background-color: rgb(255, 252, 252);
    border-radius: 20px;
    outline: none;
    width: 500px;
    color: rgb(0, 0, 0);
    border: 3px solid rgb(45, 47, 49);
}

.lload {
    position: relative;
    top: 40px; left: 390px;
    font-size: 20px;
    background-color: rgb(255, 252, 252);
    border-radius: 20px;
    outline: none;
    width: 500px;
    color: rgb(0, 0, 0);
    border: 3px solid rgb(45, 47, 49);
}

.txt {
    position: relative;
    top: 40px; left: 390px;
}

::-webkit-file-upload-button {
    color: rgb(255, 252, 252);
    padding: 20px;
    border: 2px solid rgb(201, 6, 6);
    background-color: rgb(201, 6, 6);
    border-radius: 15px;

}

::-webkit-file-upload-button:hover {
    border-radius: 20px;
```

```
        border: 2px solid rgb(177, 174, 174);


    }



    input[type="submit"] {
        position: absolute;
        margin-top: 200px;
        padding: 15px 35px;
        background-color: rgb(31, 185, 190);
        border-radius: 15px;
        color: black;
        font-size: 1.5rem;
        border: 4px solid rgb(31, 185, 190);
    }
</style>
</head>
<body style="background-image: url('Background.jpeg');">
        <h1 style="text-align:center;font-size:4rem;">Predict Plant Disease & Get
Cure</h1>
<section>



  <label for="cars" class = "txt" >Choose your plant:</label>
  <select name="cars" id="cars" class="lload">
    <option value="Fruit">Fruit plant</option>
    <option value="Vegetable">vegetable plant</option>
  </select>
        <form action="/predict" method="post" enctype="multipart/form-data"
onsubmit="showloading()">

            <br>
            <input type="file" name="image" class="upload">
                        <br>
                        <br>
            <input type="submit" value="Predict">

        </form>
    </div>
```

```
    </section>

</body>
</html>
```

Pepper_bell_bacterial_spot.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
```

```
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>
```

```
                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                     <p style="text-align: center;position: relative;top: 0px;font-size:
15.5px;">
                      Control of bacterial spot on greenhouse transplants is an essential
step for preventing the spread of the leaf spot bacteria in the field. Transplants
should be inspected regularly to identify symptomatic seedlings. Transplants with
symptoms may be removed and destroyed or treated with streptomycin, if detected
at the very early stage of disease development. It should be noted that strains of
leaf spot bacteria resistant to streptomycin may arise with multiple applications of
streptomycin

                    </p>
                    </div>

                </div>
            </div>
          </div>
        </div>


    </div>
</body>

</html>
```

pepper_bell_healthy.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
```

```
    integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

  <title>PLANT DISEASE PREDICTION</title>

  <style>
    * {
      margin: 0px;
      padding: 0px;
      box-sizing: border-box;

    }

    .border img {
      border-radius: 15px;
      border: 2px solid black;
    }

  </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

  <div>
    <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
  </div>

 <br>
 <br>


  <div class="container my-2">
    <div class="row mb-5">

      <div class="col-sm" style="margin-bottom: 23px;">
```

```
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>

               <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                  <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                    <p style="text-align: center;position: relative;top: 50px;font-size:
22px;">
                      This leaf is <span style="color: green;"><b>healthy</b></span>.
                  </p>
                  </div>

               </div>
            </div>
         </div>
      </div>
```

```
    </div>
</body>

</html>
```

Potato_early_blight.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
```

```
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>
```

```
                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center; font-size: 15.5px;">
                        Early blight can be minimized by maintaining optimum growing
conditions, including proper fertilization, irrigation, and management of other
pests. Grow later maturing, longer season varieties. Fungicide application is
justified only when the disease is initiated early enough to cause economic loss.
                    </p>
                    </div>

                </div>
              </div>
          </div>
       </div>


   </div>
</body>

</html>
```

Potato_healthy.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
```

```
        margin: 0px;
        padding: 0px;
        box-sizing: border-box;

    }

    .border img {
        border-radius: 15px;
        border: 2px solid black;
    }

  </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

  <div>
    <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
  </div>

  <br>
  <br>


  <div class="container my-2">
    <div class="row mb-5">

      <div class="col-sm" style="margin-bottom: 23px;">
        <span class="border border-primary">
          <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


        </span>
      </div>
```

```
        <div class="col-sm">

            <div>
                <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                    class="text-center mb-5 content-h1 rounded">
                    {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center;position: relative;top: 50px;font-size:
22px;">
                        This leaf is <span style="color: green;"><b>healthy</b></span>.
                    </p>
                    </div>

                </div>
            </div>
        </div>
      </div>


   </div>
</body>

</html>
```

Potato_late_blight.html

```
<!DOCTYPE html>
```

```
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
         margin: 0px;
         padding: 0px;
         box-sizing: border-box;

      }

      .border img {
         border-radius: 15px;
         border: 2px solid black;
      }

   </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>
```

```
  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>

               <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                  <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                     <p style="text-align: center; font-size: 15.5px;">
                      Late blight is controlled by eliminating cull piles and volunteer
potatoes, using proper harvesting and storage practices, and applying fungicides
when necessary. Air drainage to facilitate the drying of foliage each day is
```

important. Under marginal conditions, overhead sprinkler irrigation can favor late blight; in Tule Lake under solid set sprinklers, conditions conducive to late blight development are enhanced by day time irrigation but not night time irrigation.

```
                    </p>
                    </div>

                </div>
            </div>
        </div>
    </div>


</body>

</html>
```

Tomato_bacterial_spot.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }
```

```
        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-field-with-sprayer-at-spring-picture-id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

    <div>
        <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
    </div>

   <br>
   <br>


    <div class="container my-2">
        <div class="row mb-5">

            <div class="col-sm" style="margin-bottom: 23px;">
                <span class="border border-primary">
                    <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


                </span>
            </div>

            <div class="col-sm">

                <div>
                    <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
```

```
                class="text-center mb-5 content-h1 rounded">
                {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
            </div>

            <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                  <p style="text-align: center; font-size: 15.5px;">
                    Plant pathogen-free seed or transplants to prevent the
introduction of bacterial spot pathogens on contaminated seed or seedlings.  If a
clean seed source is not available or you suspect that your seed is contaminated,
soak seeds in water at 122°F for 25 min. to kill the pathogens.
                </p>
                </div>

            </div>
          </div>
        </div>
      </div>


   </div>
</body>

</html>
```

Tomato_late_blight.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

    <div>
        <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
    </div>

   <br>
   <br>
```

```
<div class="container my-2">
    <div class="row mb-5">

        <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
                <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
        </div>

        <div class="col-sm">

            <div>
                <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                    class="text-center mb-5 content-h1 rounded">
                    {{pred_output}} </h1>
            </div>
            <div class="details">
                                        <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center; font-size: 15.5px;">
                        Continue weekly spray applications to protect plants from further
infection. Severely infected plants can be rogued and either buried or burned.
Avoid composting diseased plants
                    </p>
                    </div>

                </div>
```

```
            </div>
          </div>
        </div>


    </div>
</body>

</html>
```

Tomato_leaf_mold.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>
```

```
<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
```

```
                <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
            </div>

            <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                  <p style="text-align: center; font-size: 15.5px;position:relative;top:
50px;">
                    Maintain night temps higher than outside temperatures.

                </p>
                </div>

            </div>
          </div>
        </div>
      </div>


  </div>
</body>

</html>
```

Tomato_septoria_leaf_spot.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

  <title>PLANT DISEASE PREDICTION</title>
```

```
    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

    <div>
        <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
    </div>

   <br>
   <br>


    <div class="container my-2">
        <div class="row mb-5">

            <div class="col-sm" style="margin-bottom: 23px;">
                <span class="border border-primary">
                    <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">
```

```
            </span>
        </div>

        <div class="col-sm">

            <div>
                <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                    class="text-center mb-5 content-h1 rounded">
                    {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center; font-size: 15.5px;">
                        Improve air circulation around the plants. If the plants can still be
handled without breaking them, stake or cage the plants to raise them off the
ground and promote faster drying of the foliage.
                    </p>
                    </div>

                </div>
            </div>
        </div>
    </div>


   </div>
</body>

</html>
```

Apple_Black_Rot.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
         margin: 0px;
         padding: 0px;
         box-sizing: border-box;

      }

      .border img {
         border-radius: 15px;
         border: 2px solid black;
      }

   </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">
```

```
   <div>
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>

               <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
```

```
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center;">
                        Carefully prune and dispose of dead wood. This should be an
important component of both current-season and long-range management.
                        Prune and remove cankers; properly dispose of prunings by burial
or burning.
                        Remove all mummified fruit.
                        Control fire blight by pruning out infected wood or controlling
insect vectors
                    </p>
                    </div>

                </div>
            </div>
        </div>
    </div>


  </div>
</body>

</html>
```

Apple_Healthy.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
    integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

  <title>PLANT DISEASE PREDICTION</title>

  <style>
```

```
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

    <div>
        <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
    </div>

   <br>
   <br>


    <div class="container my-2">
        <div class="row mb-5">

            <div class="col-sm" style="margin-bottom: 23px;">
                <span class="border border-primary">
                    <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


                </span>
```

```
        </div>

        <div class="col-sm">

            <div>
                <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                    class="text-center mb-5 content-h1 rounded">
                    {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center;position: relative;top: 50px;font-size:
22px;">
                        This leaf is <span style="color: green;"><b>healthy</b></span>.
                    </p>
                    </div>

                </div>
            </div>
        </div>
      </div>


   </div>
</body>

</html>
```

Corn(maize)_Healthy.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
         margin: 0px;
         padding: 0px;
         box-sizing: border-box;

      }

      .border img {
         border-radius: 15px;
         border: 2px solid black;
      }

   </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
```

```
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>

               <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                  <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
```

```
                    <p style="text-align: center;position: relative;top: 50px;font-size:
22px;">
                      This leaf is <span style="color: green;"><b>healthy</b></span>.
                    </p>
                    </div>

                </div>
              </div>
            </div>
        </div>

    </div>
</body>

</html>
```

Corn(maize)_northern_leaf_blight.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
        integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

    <title>PLANT DISEASE PREDICTION</title>

    <style>
        * {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;

        }

        .border img {
```

```
            border-radius: 15px;
            border: 2px solid black;
        }

    </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-field-with-sprayer-at-spring-picture-id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

    <div>
        <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
    </div>

   <br>
   <br>


    <div class="container my-2">
        <div class="row mb-5">

            <div class="col-sm" style="margin-bottom: 23px;">
                <span class="border border-primary">
                    <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


                </span>
            </div>

            <div class="col-sm">

                <div>
                    <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                        class="text-center mb-5 content-h1 rounded">
```

```
                {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                     <p style="text-align: center;position: relative;top: 0px;font-size:
18px;">
                       First, choose corn varieties or hybrids that are resistant or at least
have moderate resistance to northern corn leaf blight. When you grow corn, make
sure it does not stay wet for long periods of time. The fungus that causes this
infection needs between six and 18 hours of leaf wetness to develop. Plant corn
with enough space for airflow and water in the morning so leaves can dry
throughout the day.

                    </p>
                    </div>

                </div>
            </div>
          </div>
        </div>


    </div>
</body>

</html>
```

Peach_bacterial_spot.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
         margin: 0px;
         padding: 0px;
         box-sizing: border-box;

      }

      .border img {
         border-radius: 15px;
         border: 2px solid black;
      }

   </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

   <div>
```

```
      <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
   </div>

  <br>
  <br>


   <div class="container my-2">
      <div class="row mb-5">

         <div class="col-sm" style="margin-bottom: 23px;">
            <span class="border border-primary">
               <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


            </span>
         </div>

         <div class="col-sm">

            <div>
               <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                  class="text-center mb-5 content-h1 rounded">
                  {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                  <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
               </div>

               <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                  <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
```

The best strategy is to use cultivars with better bacterial spot resistance. This is especially true for orchards in sandy sites prone to the disease. Varieties developed by breeding programs in wet, sandy regions will tend to have better bacterial spot resistance than those developed for dry regions such as California. A variety with moderate resistance to bacterial spot may do well in a site sheltered from wind but have unacceptable symptoms in a bacterial spot-prone site or if planted with highly susceptible varieties.

</body>

</html>

Peach_healthy.html

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.1/css/bootstrap.min.css"
      integrity="sha384-
VCmXjywReHh4PwowAiWNagnWcLhlEJLA5buUprzK8rxFgeH0kww/aWY76T
fkUoSX" crossorigin="anonymous">

   <title>PLANT DISEASE PREDICTION</title>

   <style>
      * {
         margin: 0px;
```

```
        padding: 0px;
        box-sizing: border-box;

    }

    .border img {
        border-radius: 15px;
        border: 2px solid black;
    }

  </style>
</head>

<body style="background-image:
url('https://media.istockphoto.com/photos/tractor-spraying-pesticides-on-soybean-
field-with-sprayer-at-spring-picture-
id966855552?b=1&k=20&m=966855552&s=170667a&w=0&h=xtZdO1-
cST47qsG2rGkhCpyC7poG3kGcTrYVC3djIG0=');background-size:
cover;background-attachment: fixed;">

  <div>
    <h1 style="text-align:left;font-size: 50px;padding: 20px;color:
white;">Predict Disease And Get Cure</h1>
  </div>

  <br>
  <br>


  <div class="container my-2">
    <div class="row mb-5">

      <div class="col-sm" style="margin-bottom: 23px;">
        <span class="border border-primary">
          <img src="{{url_for('static',filename =
'images/Tomato___Leaf_Mold.JPG' )}}" alt="Image Not Found"
style="width:500px;height:500px;">


        </span>
      </div>
```

```
        <div class="col-sm">

            <div>
                <h1 style="padding: 15px; background-color:rgb(95, 231, 95); color:
white;"
                    class="text-center mb-5 content-h1 rounded">
                    {{pred_output}} </h1>
            </div>
            <div class="details">
                                    <div class="box" style="background-color:
aliceblue;width: 200px;border-radius: 10px;position: relative;left: 200px;top:
72px;">
                    <p style="text-align: center;font-size: 30px;color:
black"><b>Treatment</b></p>
                </div>

                <div class="box-sol" style="background-color: white;height:
200px;width: 600px;border-radius: 10px;position: relative;top: 50px;">
                    <div class="mypara" style="height: 150;width: 550px; position:
relative;left: 20px;top: 30px;">
                      <p style="text-align: center;position: relative;top: 50px;font-size:
22px;">
                        This leaf is <span style="color: green;"><b>healthy</b></span>.
                    </p>
                    </div>

                </div>
            </div>
        </div>
      </div>


   </div>
</body>

</html>
```

Git Hub Link:

https://github.com/IBM-EPBL/IBM-Project-39948-1660571875

Project Demo Link:

https://drive.google.com/file/d/1HT-UC4nfK1coGsUiLsmkjWddG0bU7uuA/view?usp=share_link