Fertilizers Recommendation System for Disease Prediction

Done by,

Team ID: PNT2022TMID08595

N.C. Sathiyanarayanan (727619BCS034)

L. Nithishkumar (727619BCS038)

V.V. Arulyaswin (727619BCS082)

R. Aadharsh (727619BCS084)

Index

1. INTRODUCTION

1.1 Project Overview

Agriculture is the most crucial one for the human existence. Crop disease prediction helps in enhancing the growth of agricultural crops and booming up the agriculture. Plant disease prediction helps in the detection and recognition of the plant diseases. The images of plants are captured and analyzed for certain symptoms using Computer vision and image processing. By identifying the disease, the deficit nutrients that lead to the disease are found. Based on the available data on fertilizers are recommended for curing the disease and also suggests some effective measures for increasing the crop yield.

1.2 Purpose

The plant diseases may lead to the reduction in the yield of crops. The main objective of the project is to detect the crop diseases effectively and suggesting the suitable fertilizers for diagnosing the disease. The project also aims at guiding the farmers with the proper choice of the fertilizers that are required to counter the deficiency of the nutrients that cause the disease.

2. LITERATURE SURVEY

2.1 Existing Problem

I. Soil Based Fertilizer Recommendation System for crop disease prediction

(**Dr.P.Pandi Selvi, P. Poornima – 2021)**

Agriculture is the heart and life of most Indians. But in recent days, the field was going down due to various natural calamities. In order to overcome the problem, various issues in this field need to be addressed. The soil type, fertilizer recommendation, diseases in plants and leaves. All these features need to be considered. Our proposed system was organized in such a way, to analyse the soil type, diseases in the leaves and finally to recommend the appropriate fertilizer to the farmers, that may be of great help to them. 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. Hence the authors proposed and implemented new fertilizers Recommendation System for crop disease prediction.

II. Study on prognostication of crop diseases using Artificial Intelligence
(B. S. Eleena, Meghana Mangipudi, K. Apoorva - Computer Science - Asian Journal of Research in Computer Science – 2022)

It is universally accepted fact that crop diseases are one of the major threats in agriculture that ultimately result in drastic reduction of food supply. The present project study aims to use artificial intelligence in building a model which is integrated with a user-friendly web application. The web application is created using the Python-based Django framework. This user interface allows the user to choose a crop name and upload an image of a leaf wherein the trained model then begins the process of feature extraction on the image and tries to make an accurate prediction. The final result is displayed to the user confirming whether the crop may be “healthy” or the “diseased “and even the name of the disease that infects the plant will be displayed. The application also suggests a suitable treatment to combat the disease. Thus, the scope of this project study is very scalable as it can be easily be used by amateur gardeners as well as by farmers. The model itself can also be extended to include more plant types along with any new diseases which may arise due to factors like climate change, pest - resistance etc.

III. Machine learning is an emerging research field in crop yield analysis. Yield prediction is a very important issue in agriculture. Any farmer is interested in knowing how much yield he is about to expect. In the past, yield prediction was performed by considering farmer's experience on field and crop. The yield prediction is a major issue that remains to be solved based on available data. Machine learning techniques are the better choice for this purpose. Different Machine learning techniques are used and evaluated in agriculture for estimating the future year's crop production. This paper proposes and implements a system to predict crop yield from previous data. This is achieved by applying machine learning algorithms like Support Vector Machine and Random Forest on agriculture data and recommends fertilizer suitable for every crop. The paper focuses on creation of a prediction model which may be used for future prediction of crop yield. It presents a brief analysis of crop yield prediction using machine learning techniques.

2.2 References

[1]. G. Preethi, P. Rathi, S. M. Sanjula, S. D. Lalitha, B. V. Bindhu, “Agro based crop and fertilizer recommendation system using machine learning”, European Journal of Molecular & Clinical Medicine, 7, 4, 2020, 2043- 2051 https://deepai.org/publication/farmer-s-assistant-a-machine-learningbased-application-for-agricultural-solutions.

[2]. Plant Disease Detection Using Image Processing and Machine Learning Pranesh Kulkarni1 , Atharva Karwande1 , Tejas Kolhe1 , Soham Kamble1 , Akshay Joshi1 , Medha Wyawahare1 1 Department of Electronics and Telecommunication, Vishwakarma Institute of Technology. https://arxiv.org/ftp/arxiv/papers/2106/2106.10698.pdf

[3]. Zhen Nan Liu, et al., (2018), In this paper, authors have compared different machine learning algorithms for calculating, Standardised Precipitation Index (SPI) and SPEI. After data collection, Extreme learning methods, Online sequential extreme learning machine, Selfadaptive evolutionary extreme learning machine. Authors claimed that all three algorithms can be applied successfully on drought forecasting. However, OS-ELM and SADE- ELM performs better than ELM.

[4]. K. G. Liakos et al., (2018) presented a comprehensive review of research dedicated to machine learning applications in agriculture domain. Various parameters on which work was analysed were: crop management, livestock management, water management and soil management. ML models have applied for crop yield prediction and disease detection. ML based

detection can be extracted without the need of fusion of data from other resources. Author claims that farm management systems are evlvong into real artificial intelligent systems, with the ultimate scope of production improvement. Author motivates to use ML for the benefit of agriculture as it is the basic need amongst all other needs for survival.

[5]. S. D. Khirade, A. B. Patil, “Plant Disease Detection Using Image Processing”, 2015 International Conference on Computing Communication Control and Automation, 2015, pp. 768-771, doi: 10.1109/ICCUBEA.2015.153

2.2 Problem Statement Definition

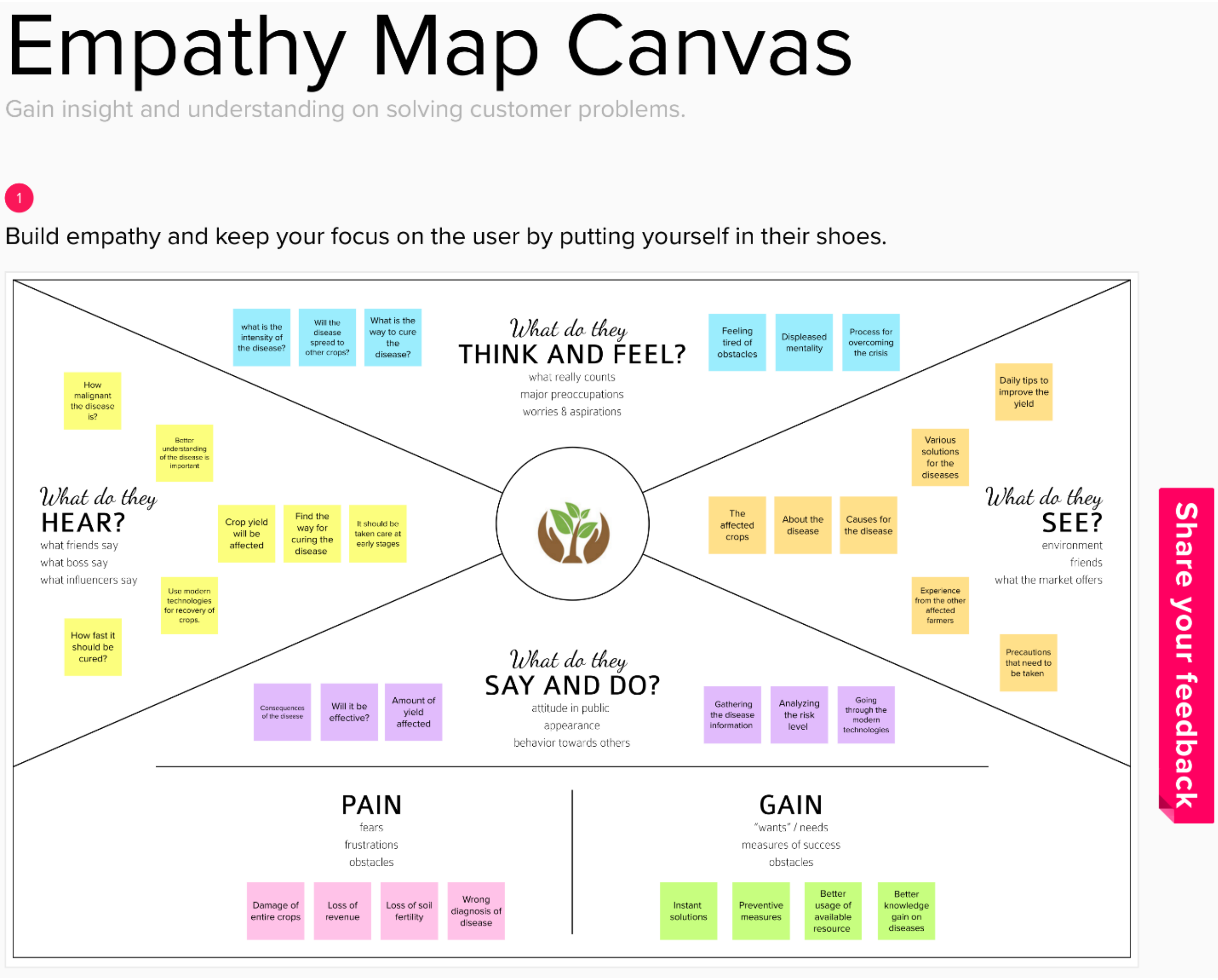
Problem Statement (PS)	**I am**	**I’m trying to**	**But**	**Because**	**Which makes me feel**
PS-1	Farmer	Find the solution for the disease.	I was not able to find the resources.	There was no centralized and organized information about diseases.	Helpless and hopeless.
PS-2	Student	Learn about the diseases and their appropriate fertilizers for curing.	I did not have enough capital for the gain of knowledge.	I was not able to find the free resources.	Clueless and unguided.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

what is the intensity of the disease?
Will the disease spread to other crops?
What is the way to cure the disease?
Feeling tired of obstacles
Displeased mentality
Process for overcoming the crisis

What do they HEAR?
what friends say
what boss say
what influencers say

How malignant the disease is?
Better understanding of the disease is important
Crop yield will be affected
Find the way for curing the disease
It should be taken care at early stages
Use modern technologies for recovery of crops.
How fast it should be cured?

What do they SEE?
environment
friends
what the market offers

Daily tips to improve the yield
Various solutions for the diseases
The affected crops
About the disease
Causes for the disease
Experience from the other affected farmers
Precautions that need to be taken

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

Consequences of the diseese
Will it be effective?
Amount of yield affected
Gathering the disease information
Analyzing the risk level
Going through the modern technologies

PAIN
fears
frustrations
obstacles

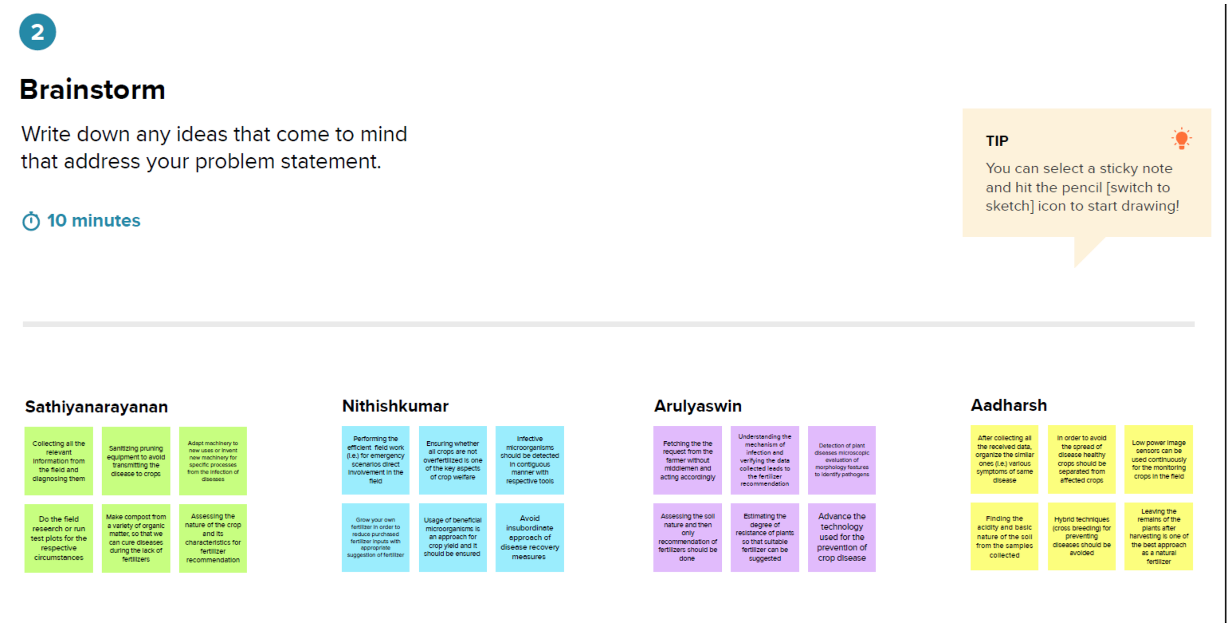
Damage of entire crops
Loss of revenue
Loss of soil fertility
Wrong diagnosis of disease

GAIN
"wants" / needs
measures of success
obstacles

Instant solutions
Preventive measures
Better usage of available resource
Better knowledge gain on diseases

3.2 Ideation & Brainstorming

Ideation and Brainstorming are performed to generate ideas and solutions. Brainstorming is a group activity unlike ideation.



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Sathiyanarayanan

- Collecting all the relevant information from the field and diagnosing them
- Sanitizing pruning equipment to avoid transmitting the disease to crops
- Adapt machinery to new uses or invent new machinery for specific processes from the infection of diseases
- Do the field research or run test plots for the respective circumstances
- Make compost from a variety of organic matter, so that we can cure diseases during the lack of fertilizers
- Assessing the nature of the crop and its characteristics for fertilizer recommendation

Nithishkumar

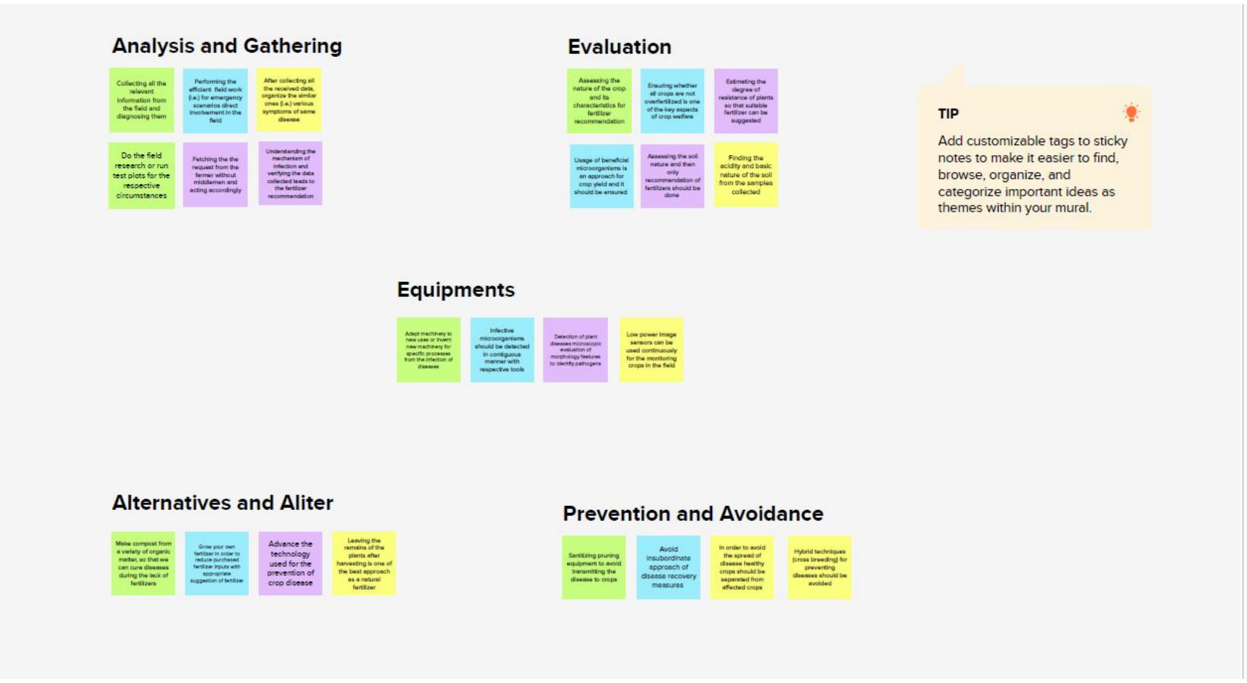
- Performing the efficient field work (i.e.) for emergency scenarios direct involvement in the field
- Ensuring whether all crops are not overfertilized is one of the key aspects of crop welfare
- Infective microorganisms should be detected in contiguous manner with respective tools
- Grow your own fertilizer in order to reduce purchased fertilizer inputs with appropriate suggestion of fertilizer
- Usage of beneficial microorganisms is an approach for crop yield and it should be ensured
- Avoid insubordinate approach of disease recovery measures

Arulyaswin

- Fetching the the request from the farmer without middlemen and acting accordingly
- Understanding the mechanism of infection and verifying the data collected leads to the fertilizer recommendation
- Detection of plant diseases microscopic evaluation of morphology features to identify pathogens
- Assessing the soil nature and then only recommendation of fertilizers should be done
- Estimating the degree of resistance of plants so that suitable fertilizer can be suggested
- Advance the technology used for the prevention of crop disease

Aadharsh

- After collecting all the received data, organize the similar ones (i.e.) various symptoms of same disease
- In order to avoid the spread of disease healthy crops should be separated from affected crops
- Low power image sensors can be used continuously for the monitoring crops in the field
- Finding the acidity and basic nature of the soil from the samples collected
- Hybrid techniques (cross breeding) for preventing diseases should be avoided
- Leaving the remains of the plants after harvesting is one of the best approach as a natural fertilizer



Analysis and Gathering

- Collecting all the relevant information from the field and diagnosing them
- Performing the efficient field work (i.e.) for emergency scenarios direct involvement in the field
- After collecting all the received data, organize the similar ones (i.e.) various symptoms of same disease
- Do the field research or run test plots for the respective circumstances
- Fetching the the request from the farmer without middlemen and acting accordingly
- Understanding the mechanism of infection and verifying the data collected leads to the fertilizer recommendation

Evaluation

- Assessing the nature of the crop and its characteristics for fertilizer recommendation
- Ensuring whether all crops are not overfertilized is one of the key aspects of crop welfare
- Estimating the degree of resistance of plants so that suitable fertilizer can be suggested
- Usage of beneficial microorganisms is an approach for crop yield and it should be ensured
- Assessing the soil nature and then only recommendation of fertilizers should be done
- Finding the acidity and basic nature of the soil from the samples collected

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Equipments

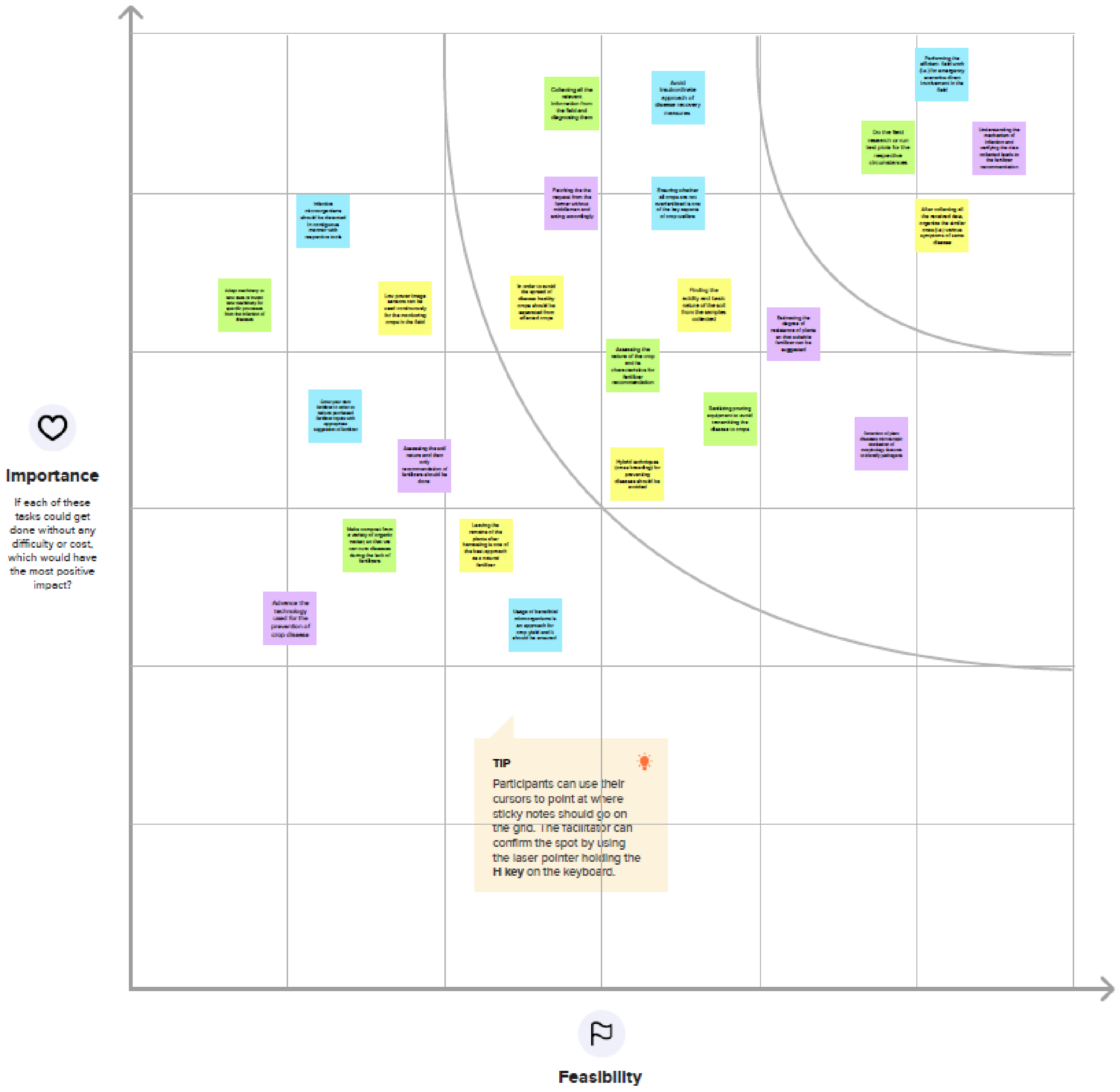
- Adapt machinery to new uses or invent new machinery for specific processes from the infection of diseases
- Infective microorganisms should be detected in contiguous manner with respective tools
- Detection of plant diseases microscopic evaluation of morphology features to identify pathogens
- Low power image sensors can be used continuously for the monitoring crops in the field

Alternatives and Aliter

- Make compost from a variety of organic matter, so that we can cure diseases during the lack of fertilizers
- Grow your own fertilizer in order to reduce purchased fertilizer inputs with appropriate suggestion of fertilizer
- Advance the technology used for the prevention of crop disease
- Leaving the remains of the plants after harvesting is one of the best approach as a natural fertilizer

Prevention and Avoidance

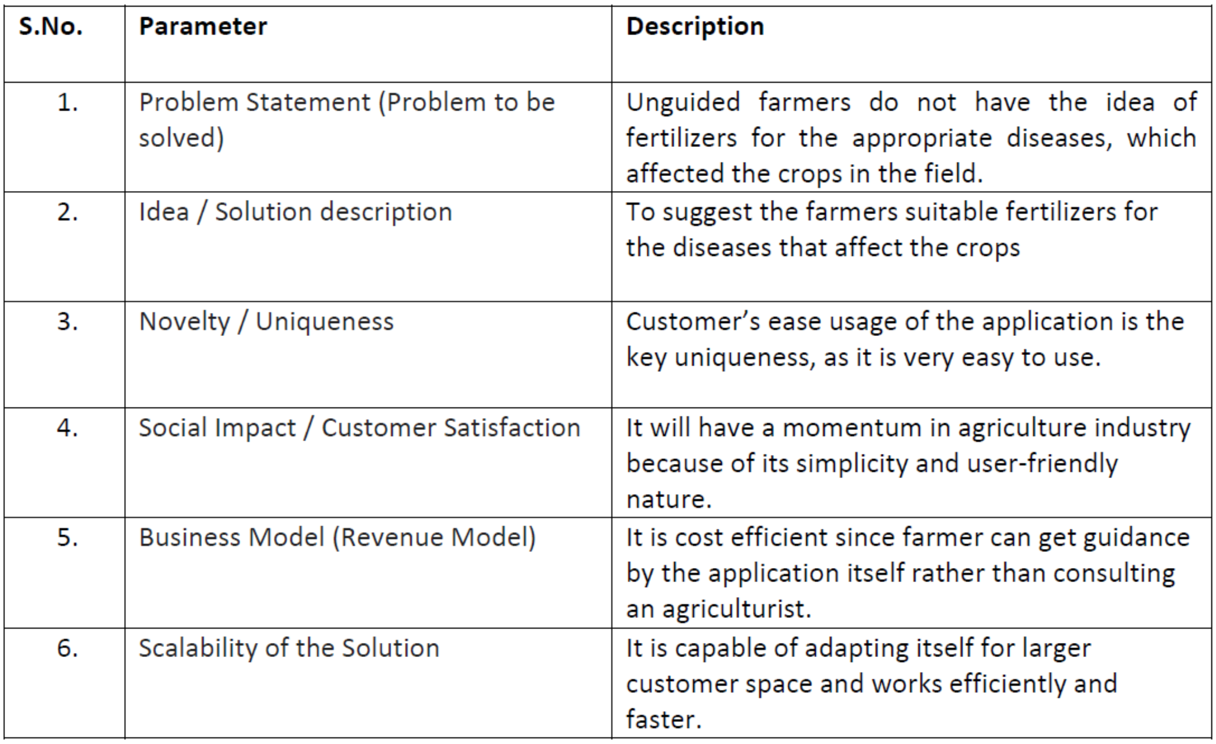
- Sanitizing pruning equipment to avoid transmitting the disease to crops
- Avoid insubordinate approach of disease recovery measures
- In order to avoid the spread of disease healthy crops should be separated from affected crops
- Hybrid techniques (cross breeding) for preventing diseases should be avoided



TIP

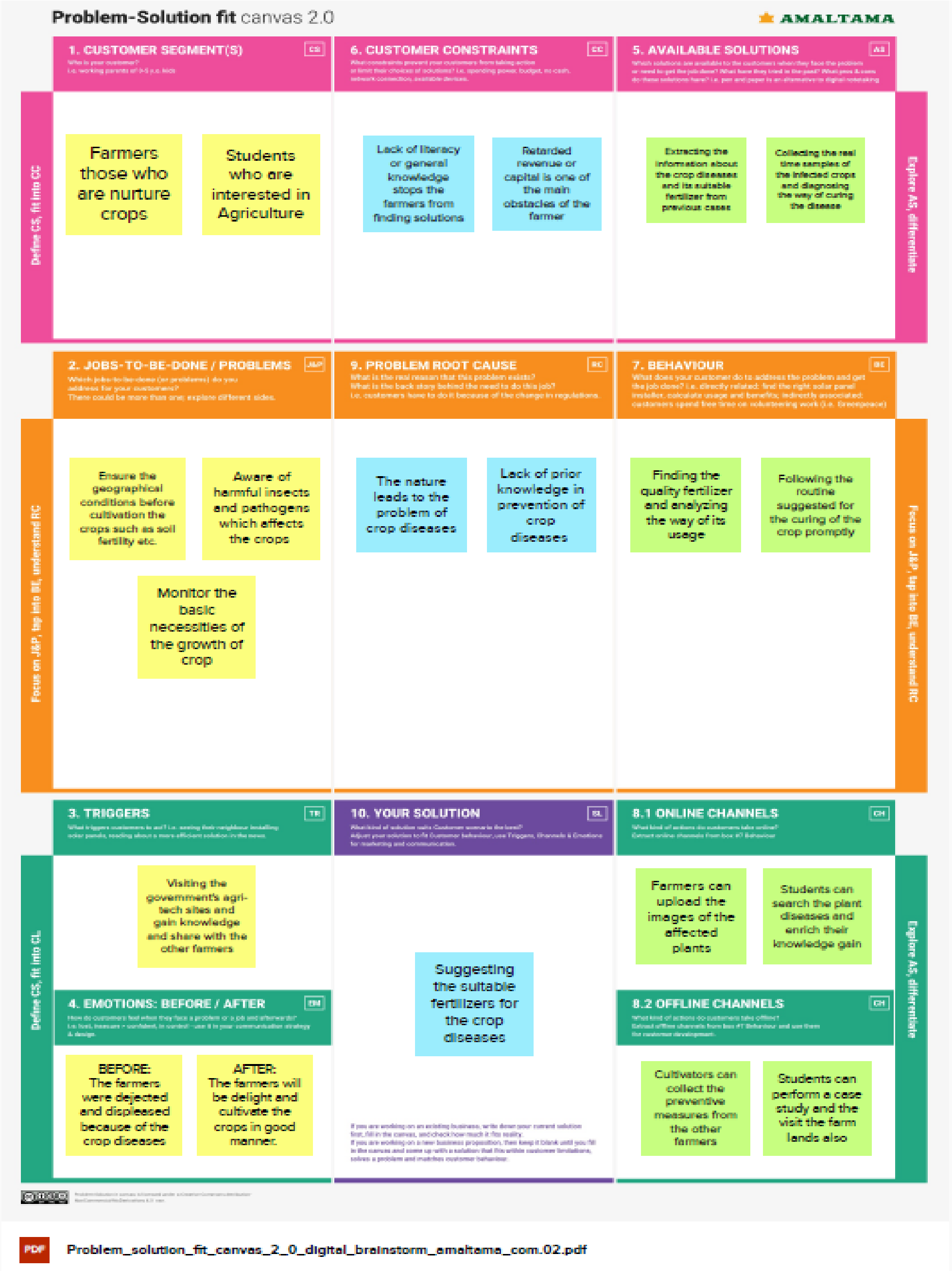
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

3.3 Proposed Solution



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Unguided farmers do not have the idea of fertilizers for the appropriate diseases, which affected the crops in the field.
2.	Idea / Solution description	To suggest the farmers suitable fertilizers for the diseases that affect the crops
3.	Novelty / Uniqueness	Customer's ease usage of the application is the key uniqueness, as it is very easy to use.
4.	Social Impact / Customer Satisfaction	It will have a momentum in agriculture industry because of its simplicity and user-friendly nature.
5.	Business Model (Revenue Model)	It is cost efficient since farmer can get guidance by the application itself rather than consulting an agriculturist.
6.	Scalability of the Solution	It is capable of adapting itself for larger customer space and works efficiently and faster.

3.4 Problem Solution Fit



Problem-Solution fit canvas 2.0

AMALTAMA

1. CUSTOMER SEGMENT(S) CS

Who is your customer? i.e. working parents of 0-5 y.o. kids

Farmers those who are nurture crops

Students who are interested in Agriculture

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Lack of literacy or general knowledge stops the farmers from finding solutions

Retarded revenue or capital is one of the main obstacles of the farmer

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Extracting the information about the crop diseases and its suitable fertilizer from previous cases

Collecting the real time samples of the infected crops and diagnosing the way of curing the disease

Define CS, fit into CC

Explore AS, differentiate

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

Ensure the geographical conditions before cultivation the crops such as soil fertility etc.

Aware of harmful insects and pathogens which affects the crops

Monitor the basic necessities of the growth of crop

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations.

The nature leads to the problem of crop diseases

Lack of prior knowledge in prevention of crop diseases

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Finding the quality fertilizer and analyzing the way of its usage

Following the routine suggested for the curing of the crop promptly

Focus on J&P, tap into BE, understand RC

Focus on J&P, tap into BE, understand RC

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Visiting the government's agri-tech sites and gain knowledge and share with the other farmers

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

BEFORE: The farmers were dejected and displeased because of the crop diseases

AFTER: The farmers will be delight and cultivate the crops in good manner.

10. YOUR SOLUTION SL

What kind of solution suits Customer scenario the best? Adjust your solution to fit Customer behaviour, use Triggers, Channels & Emotions for marketing and communication.

Suggesting the suitable fertilizers for the crop diseases

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

8.1 ONLINE CHANNELS CH

What kind of actions do customers take online? Extract online channels from box #7 Behaviour

Farmers can upload the images of the affected plants

Students can search the plant diseases and enrich their knowledge gain

8.2 OFFLINE CHANNELS CH

What kind of actions do customers take offline? Extract offline channels from box #7 Behaviour and use them for customer development.

Cultivators can collect the preventive measures from the other farmers

Students can perform a case study and the visit the farm lands also

Define CS, fit into CL

Explore AS, differentiate

Problem-Solution fit canvas is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 Inter

PDF Problem_solution_fit_canvas_2_0_digital_brainstorm_amaltama_com.02.pdf

4. REQUIREMENT ANALYSIS

4.1 Functional Requirement

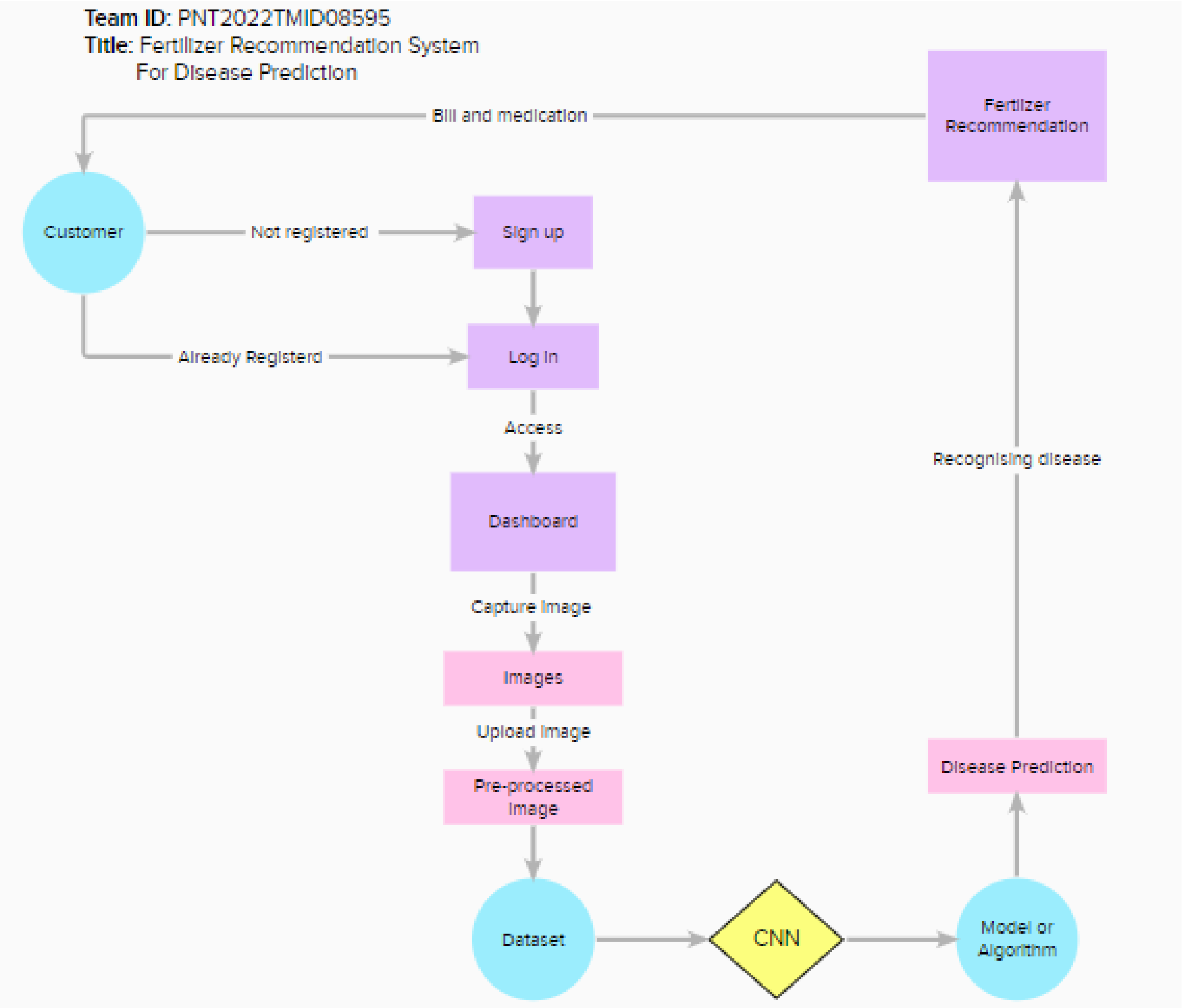
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Mobile Number Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Capturing Image	Capture the pictures of the infected leaves and upload it for the recognition of the disease
FR-4	Image Processing	The user uploads the images and those images will be processed and prediction of disease will be done
FR-5	Disease prediction	Using the images and trained datasets, the model will identify the accurate crop disease
FR-6	Fertilizer Recommendation	Using the inputs and trained datasets, suitable fertilizers will be suggested for the diseases

4.2 Non-Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Considering the level of knowledge possessed by the users of this system, it is developed as more user friendly and ease of use is the key feature.
NFR-2	**Security**	The subsystem should provide a high level of security and integrity of the data, only authorized personnel can gain access with valid password and username to view user's page
NFR-3	**Reliability**	The system provides a relatively high degree of consistency and executes its intended functions adequately without failure
NFR-4	**Performance**	On focussing its functionality, the system shows an efficient performance and never lets down its consistent state
NFR-5	**Availability**	The system will be available for all the time for all the users with its related features
NFR-6	**Scalability**	The system is even more efficient for large number of inputs and user scale.

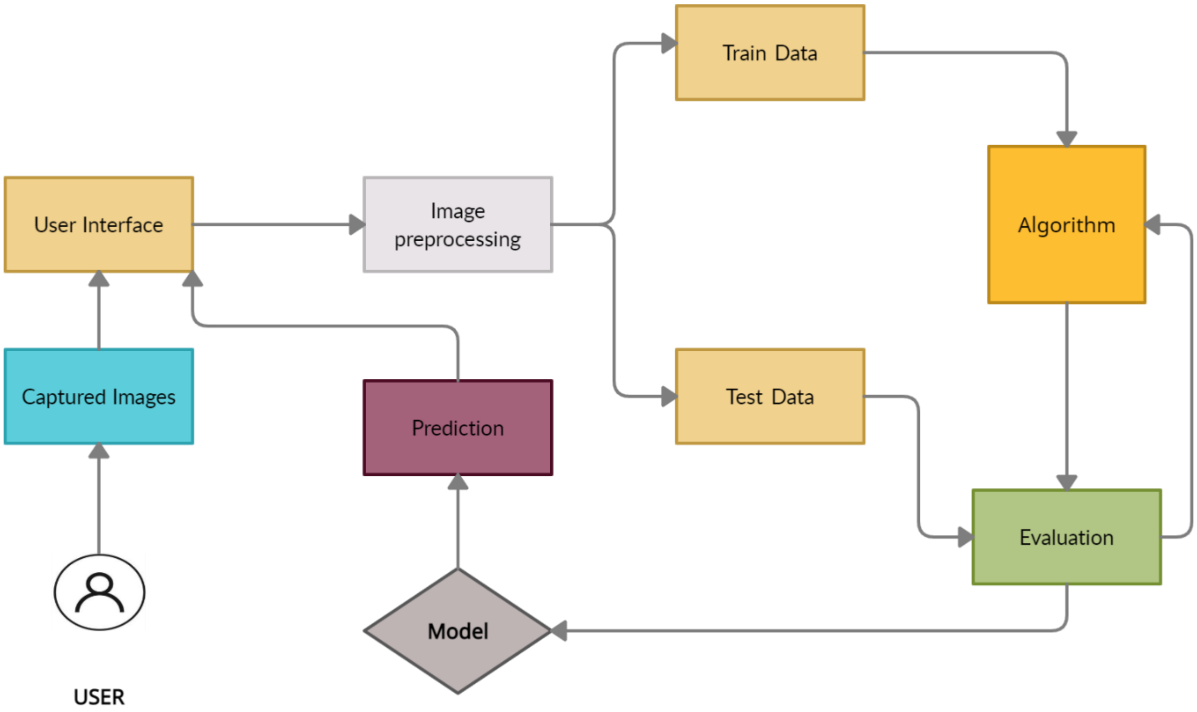
5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture

Solution architecture is the process of developing solutions based on predefined processes, guidelines and best practices with the objective that the developed solution fits within the enterprise architecture in terms of information architecture, system portfolios, integration requirements, etc.



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
	Login	USN-2	As a user, I can enter the application by entering the certain registered email and password	I should enter valid login credentials	High	Sprint-1
	Dashboard	USN-3	Once getting logged in, I can use the dashboard board menu and capture images and Upload images and update data	I should have dashboard access	High	Sprint-2
Customer (Web user)	Registration	USN-4	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-3
	Login	USN-5	As a user, I can enter the application by entering the certain registered email and password	I should enter valid login credentials	High	Sprint-3
	Dashboard	USN-6	Once getting logged in, I can use the dashboard board menu and capture images and Upload images and update data	I should have dashboard access	High	Sprint-4
	Dataset	USN-7	As a user, I want to know the available categories of flora and prevention of diseases	Variety of flora and its recovery methods	Low	Sprint-1
	Recommendation of Fertilizer	USN-8	As a user, after uploading the images, I will be suggested suitable fertilizers	I should upload disease affected crop images	High	Sprint-4
Administrator	Login	USN-9	As an administrator, I can enter the application by entering the email and password	I will have all the access to perform functionalities	High	Sprint-4
	Access and Privileges	USN-10	As an administrator, I will have access to perform all the functions and can restrict users	User information should be provided	High	Sprint-4
	Updates and Modification	USN-11	As an administrator, I can update the data and records of the application	All the data and records should be provided	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

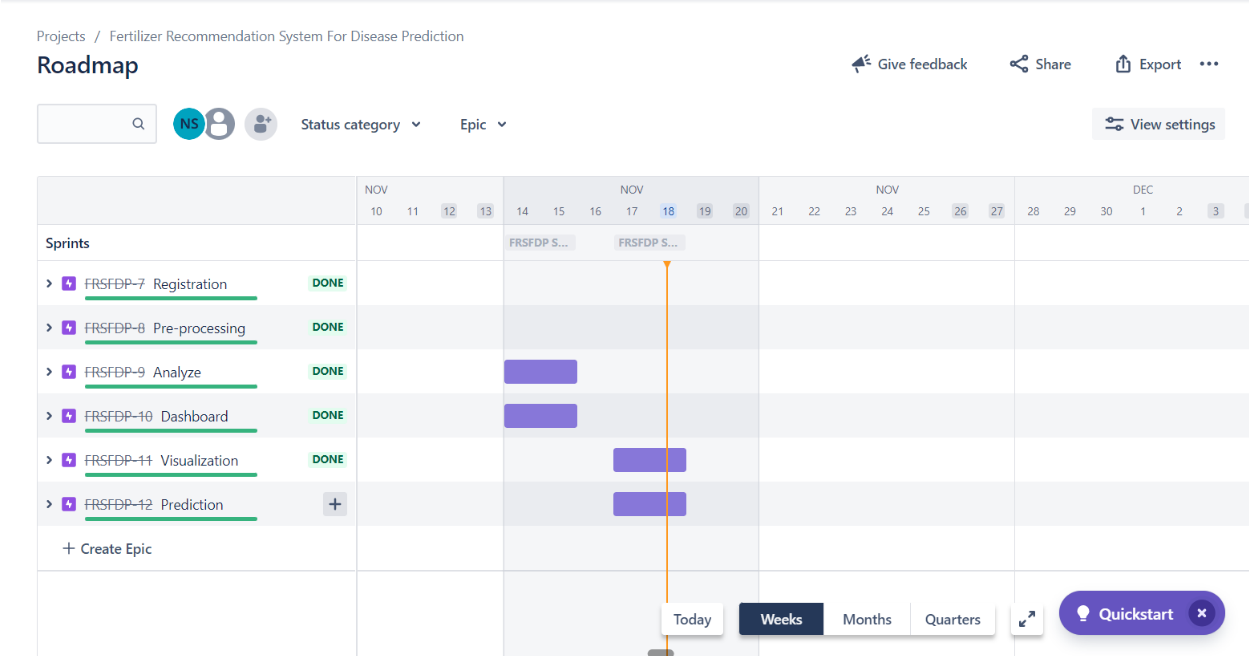
Sprint	Functional Requirement (Epic)	User Number Story	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application using email and password.	5	High	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh
Sprint-1	Pre-processing	USN-2	All the data that are collected is cleaned and uploaded in the database or IBM cloud.	8	Medium	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh
Sprint-2	Analyze	USN-3	The uploaded data are completely analyzed and making of predictions are done.	8	Medium	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh
Sprint-3	Dashboard	USN-4	Once getting logged in, I can use dashboard menu and capture images and upload images and update.	5	Medium	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh
Sprint-4	Visualization	USN-5	I can visualize the list of available diseases and their suitable fertilizer in the list.	5	High	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh
Sprint-4	Prediction	USN-6	We can Protect the crops from the diseases by predicting suitable fertilizers.	5	High	Sathiyanarayanan, Nithishkumar, Arulyaswin, Aadharsh

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	13	6 Days	24 Oct 2022	29 Oct 2022	13	31 Oct 2022
Sprint-2	8	6 Days	31 Oct 2022	05 Nov 2022	8	14 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	17 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	18 Nov 2022

6.3 Reports From JIRA

Roadmap



Projects / Fertilizer Recommendation System For Disease Prediction

Roadmap

Sprints

~~FRSFDP-7~~ Registration DONE

~~FRSFDP-8~~ Pre-processing DONE

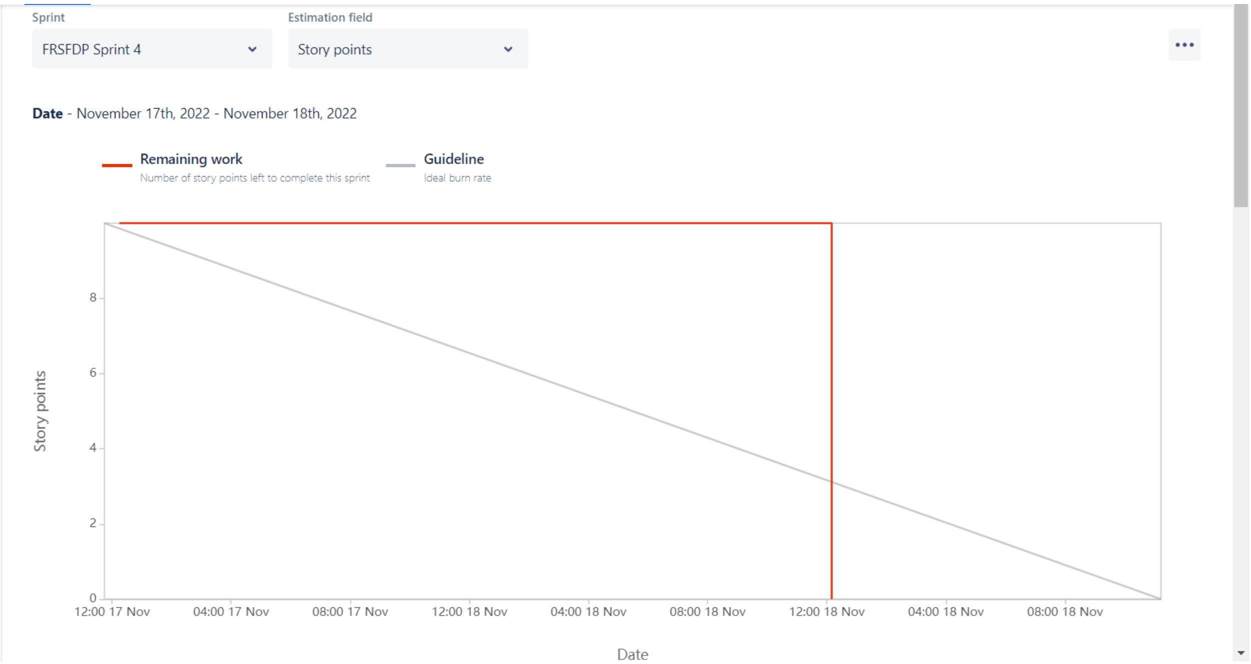
~~FRSFDP-9~~ Analyze DONE

~~FRSFDP-10~~ Dashboard DONE

~~FRSFDP-11~~ Visualization DONE

~~FRSFDP-12~~ Prediction

Burndown Report

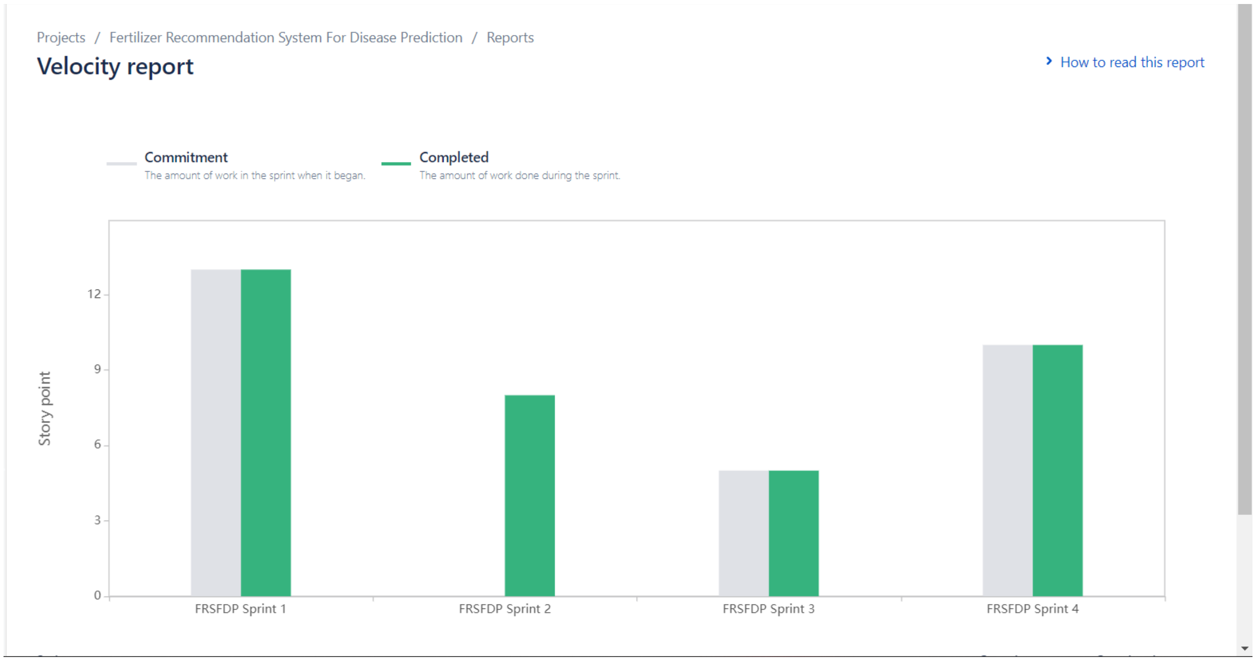


Sprint
FRSFDP Sprint 4

Estimation field
Story points

Date - November 17th, 2022 - November 18th, 2022

Velocity Report



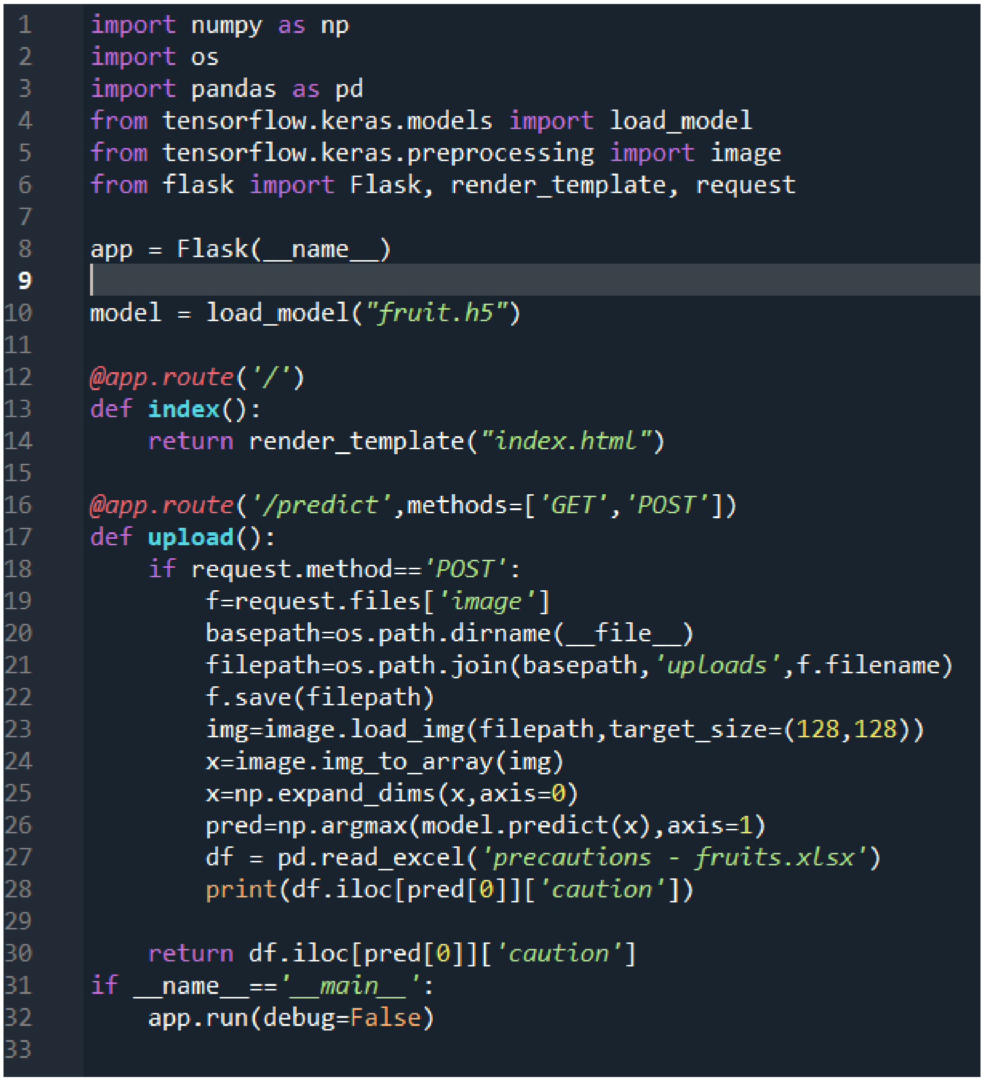
Projects / Fertilizer Recommendation System For Disease Prediction / Reports

Velocity report

› How to read this report

7. CODING & SOLUTIONING

7.1 Python – app_feretilizer.py



```python
import numpy as np
import os
import pandas as pd
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from flask import Flask, render_template, request

app = Flask(__name__)

model = load_model("fruit.h5")

@app.route('/')
def index():
    return render_template("index.html")

@app.route('/predict',methods=['GET','POST'])
def upload():
    if request.method=='POST':
        f=request.files['image']
        basepath=os.path.dirname(__file__)
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)
        img=image.load_img(filepath,target_size=(128,128))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)
        pred=np.argmax(model.predict(x),axis=1)
        df = pd.read_excel('precautions - fruits.xlsx')
        print(df.iloc[pred[0]]['caution'])

    return df.iloc[pred[0]]['caution']
if __name__=='__main__':
    app.run(debug=False)

```

7.2 Index.html

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

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Fertilizer Recommented System and Disease Prediction</title>
    <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet">
    <script
src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script
src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
    <link href="{{ url_for('static', filename='css/main.css') }}"
rel="stylesheet">
```

```
  <style>

        .bg-dark {
            background-color: #7CFC00!important;
        }
        #result {
            color: #000f;
        }
  body
{
    background-image: url("https://wallpaperaccess.com/full/327148.jpg");
    background-size: cover;
}

  </style>
</head>

<body>

    <nav class="navbar navbar-dark bg-dark">
        <div class="container">
            <a class="navbar-brand" href="#"><center><b>CROP DISEASE
PREDICTION</b></center></a>
        </div>
    </nav>
    <div class="container">
        <div id="content" style="margin-top:2em">
        <div class="container">
          <div class="row">
              <div class="col-sm-6 bd">
                <h3>Plant Disease Prediction: </h3>
                <br>
                <p>Agriculture is one of the major sectors world wide over the
years it has developed and the use of new technologies and equipment replaced
almost all the traditional methods of farming . the plant diseases affect the
production , Identification of disesases and taking necessary precautions are
done through naked eye , which requires labour and labratries . This
application helps farmers in detecting thr diseases by observing the spots and
the leaves , which in turn saves effort and labour costs .</p>
                     <img src="http://www.geocities.ws/senthilirtt/First.jpg"
height="50%",width="20%">
              </div>
              <div class="col-sm-6">
                     <div>
                            <h4>Upload Image Here To Identify the Plant
Disease</h4>
              <form action = "http://localhost:5000/" id="upload-file"
method="post" enctype="multipart/form-data">
                     <label for="imageUpload" class="upload-label">
                            Choose...
                     </label>
                     <input type="file" name="image" id="imageUpload"
accept=".png, .jpg, .jpeg">
              </form>


              <div class="image-section" style="display:none;">
                     <div class="img-preview">
```

```
                           <div id="imagePreview">
                           </div>
                     </div>
                     <div>
                           <button type="button" class="btn btn-info btn-lg "
id="btn-predict">Predict!</button>
                     </div>
              </div>

              <div class="loader" style="display:none;"></div>

              <h3>
                     <span id="result"> </span>
              </h3>

        </div>
              </div>

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

<footer>
    <script src="{{ url_for('static', filename='js/main.js') }}"
type="text/javascript"></script>
</footer>

</html>
```

7.3 main.css

```
.img-preview {
    width: 256px;
    height: 256px;
    position: relative;
    border: 5px solid #F8F8F8;
    box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);
    margin-top: 1em;
    margin-bottom: 1em;
}

.img-preview>div {
    width: 100%;
    height: 100%;
    background-size: 256px 256px;
    background-repeat: no-repeat;
    background-position: center;
}

input[type="file"] {
    display: none;
}

.upload-label{
    display: inline-block;
    padding: 12px 30px;
    background: #39D2B4;
    color: #fff;
    font-size: 1em;
    transition: all .4s;
    cursor: pointer;
}

.upload-label:hover{
    background: #34495E;
    color: #39D2B4;
}

.loader {
    border: 8px solid #f3f3f3; /* Light grey */
    border-top: 8px solid #3498db; /* Blue */
    border-radius: 50%;
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

@keyframes spin {
    0% { transform: rotate(0deg); }
    100% { transform: rotate(360deg); }
}
```

7.4 main.js

```
$(document).ready(function () {
    // Init
    $('.image-section').hide();
    $('.loader').hide();
    $('#result').hide();

    // Upload Preview
    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();
            reader.onload = function (e) {
                $('#imagePreview').css('background-image', 'url(' +
e.target.result + ')');
                $('#imagePreview').hide();
                $('#imagePreview').fadeIn(650);
            }
            reader.readAsDataURL(input.files[0]);
        }
    }
    $("#imageUpload").change(function () {
        $('.image-section').show();
        $('#btn-predict').show();
        $('#result').text('');
        $('#result').hide();
        readURL(this);
    });

    // Predict
    $('#btn-predict').click(function () {
        var form_data = new FormData($('#upload-file')[0]);

        // Show loading animation
```

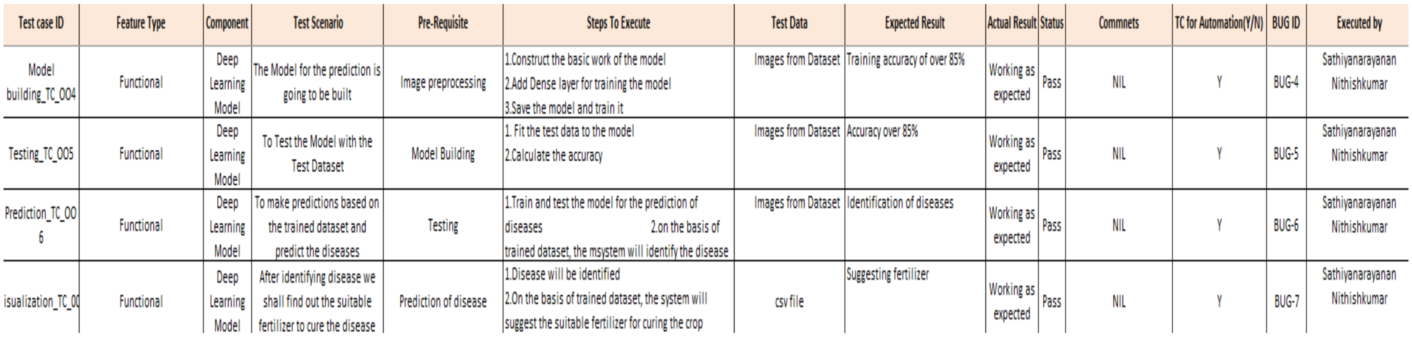
```
            $(this).hide();
            $('.loader').show();

            // Make prediction by calling api /predict
            $.ajax({
                type: 'POST',
                url: '/predict',
                data: form_data,
                contentType: false,
                cache: false,
                processData: false,
                async: true,
                success: function (data) {
                    // Get and display the result
                    $('.loader').hide();
                    $('#result').fadeIn(600);
                    $('#result').text(' Result:  ' + data);
                    console.log('Success!');
                },
            });
        });

    });
```

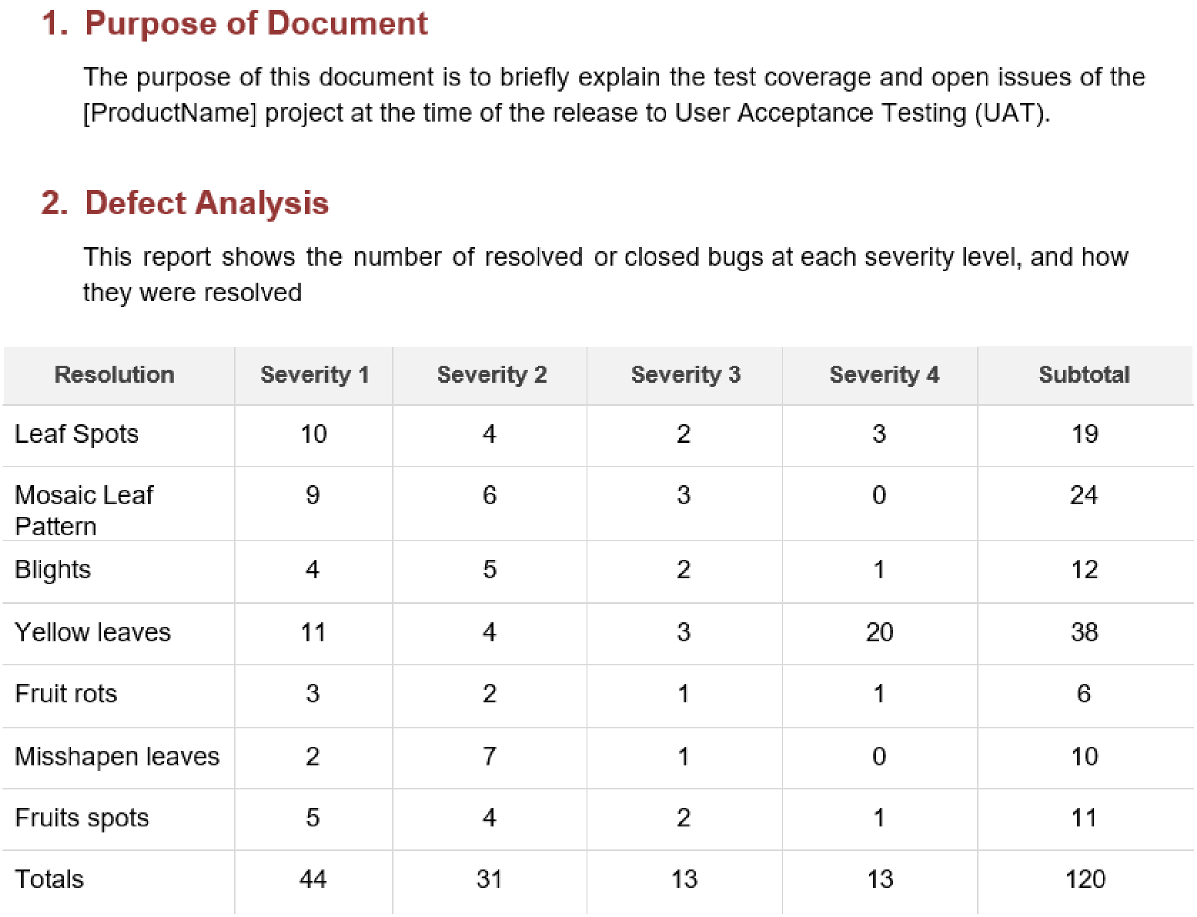
8. TESTING

8.1 Test Cases



Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed by
Model building_TC_004	Functional	Deep Learning Model	The Model for the prediction is going to be built	Image preprocessing	1.Construct the basic work of the model 2.Add Dense layer for training the model 3.Save the model and train it	Images from Dataset	Training accuracy of over 85%	Working as expected	Pass	NIL	Y	BUG-4	Sathiyanarayanan Nithishkumar
Testing_TC_005	Functional	Deep Learning Model	To Test the Model with the Test Dataset	Model Building	1. Fit the test data to the model 2.Calculate the accuracy	Images from Dataset	Accuracy over 85%	Working as expected	Pass	NIL	Y	BUG-5	Sathiyanarayanan Nithishkumar
Prediction_TC_006	Functional	Deep Learning Model	To make predictions based on the trained dataset and predict the diseases	Testing	1.Train and test the model for the prediction of diseases 2.on the basis of trained dataset, the msystem will identify the disease	Images from Dataset	Identification of diseases	Working as expected	Pass	NIL	Y	BUG-6	Sathiyanarayanan Nithishkumar
isualization_TC_00	Functional	Deep Learning Model	After identifying disease we shall find out the suitable fertilizer to cure the disease	Prediction of disease	1.Disease will be identified 2.On the basis of trained dataset, the system will suggest the suitable fertilizer for curing the crop	csv file	Suggesting fertilizer	Working as expected	Pass	NIL	Y	BUG-7	Sathiyanarayanan Nithishkumar

8.2 User Acceptance Testing



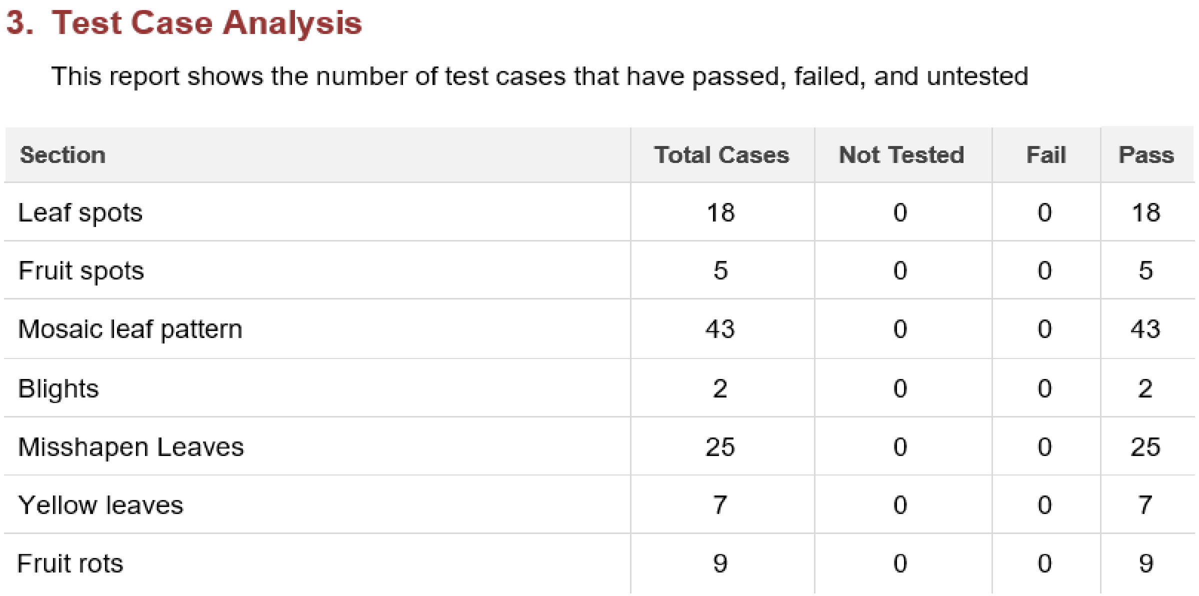
1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
Leaf Spots	10	4	2	3	19
Mosaic Leaf Pattern	9	6	3	0	24
Blights	4	5	2	1	12
Yellow leaves	11	4	3	20	38
Fruit rots	3	2	1	1	6
Misshapen leaves	2	7	1	0	10
Fruits spots	5	4	2	1	11
Totals	44	31	13	13	120



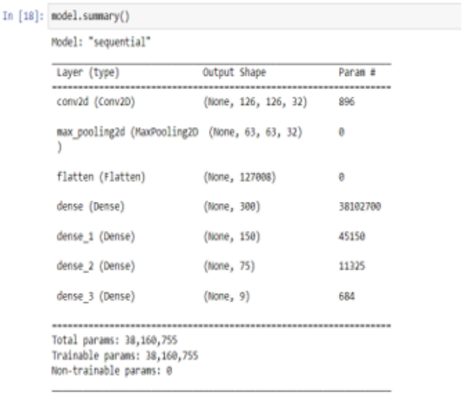
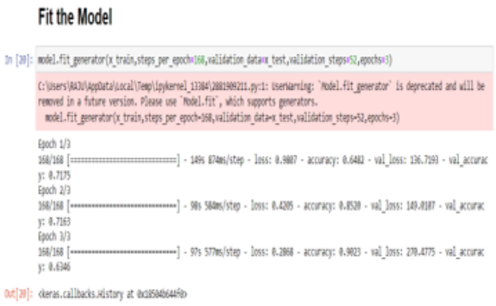
3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Leaf spots	18	0	0	18
Fruit spots	5	0	0	5
Mosaic leaf pattern	43	0	0	43
Blights	2	0	0	2
Misshapen Leaves	25	0	0	25
Yellow leaves	7	0	0	7
Fruit rots	9	0	0	9

9. RESULTS

9.1 Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 38,160,755 Trainable params: 38,160,755 Non-Trainable params: 0	
2.	Accuracy	Training Accuracy – 97.55 Validation Accuracy – 96.45	

```
In [18]: model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 126, 126, 32)      896

 max_pooling2d (MaxPooling2D  (None, 63, 63, 32)       0
 )

 flatten (Flatten)           (None, 127008)            0

 dense (Dense)               (None, 300)               38102700

 dense_1 (Dense)             (None, 150)               45150

 dense_2 (Dense)             (None, 75)                11325

 dense_3 (Dense)             (None, 9)                 684

=================================================================
Total params: 38,160,755
Trainable params: 38,160,755
Non-trainable params: 0
_________________________________________________________________
```

Fit the Model

```
In [20]: model.fit_generator(x_train,steps_per_epoch=168,validation_data=x_test,validation_steps=52,epochs=3)

C:\Users\RAJU\AppData\Local\Temp\ipykernel_13384\2881909211.py:1: UserWarning: `Model.fit_generator` is deprecated and will be
removed in a future version. Please use `Model.fit`, which supports generators.
  model.fit_generator(x_train,steps_per_epoch=168,validation_data=x_test,validation_steps=52,epochs=3)

Epoch 1/3
168/168 [==============================] - 149s 874ms/step - loss: 0.9807 - accuracy: 0.6482 - val_loss: 136.7193 - val_accurac
y: 0.7175
Epoch 2/3
168/168 [==============================] - 98s 584ms/step - loss: 0.4205 - accuracy: 0.8520 - val_loss: 149.0187 - val_accurac
y: 0.7163
Epoch 3/3
168/168 [==============================] - 97s 577ms/step - loss: 0.2868 - accuracy: 0.9023 - val_loss: 270.4775 - val_accurac
y: 0.6346

Out[20]: <keras.callbacks.History at 0x18504b644f0>
```

10. ADVANTAGES & DISADVANTAGES

Advantages

- Early detection of plant diseases.
- Proper fertilizer recommendation to prevent or cure the plant infection or disease.
- No need to consult any specialists.
- Fully automated system.

Disadvantages

- Requires training the system with large dataset.
- Works only on the pretrained diseases.
- When a plant is infected with multiple diseases the system may not predict all the diseases due to the mixed symptoms.
- Requires a good device connected to the internet.

11. CONCLUSION

The procedure starts from collecting the images used for training, testing and validation to image preprocessing and augmentation and finally comparison of different pretrained models over their accuracy. Finally, at the end, our model detects and distinguishes between a healthy plant and different diseases and provides suitable remedies so as to cure the disease. This paper proposed and developed a system which uses plant leaf images to detect different types of disease in tomato crops, and also provides appropriate fertilizer suggestions.

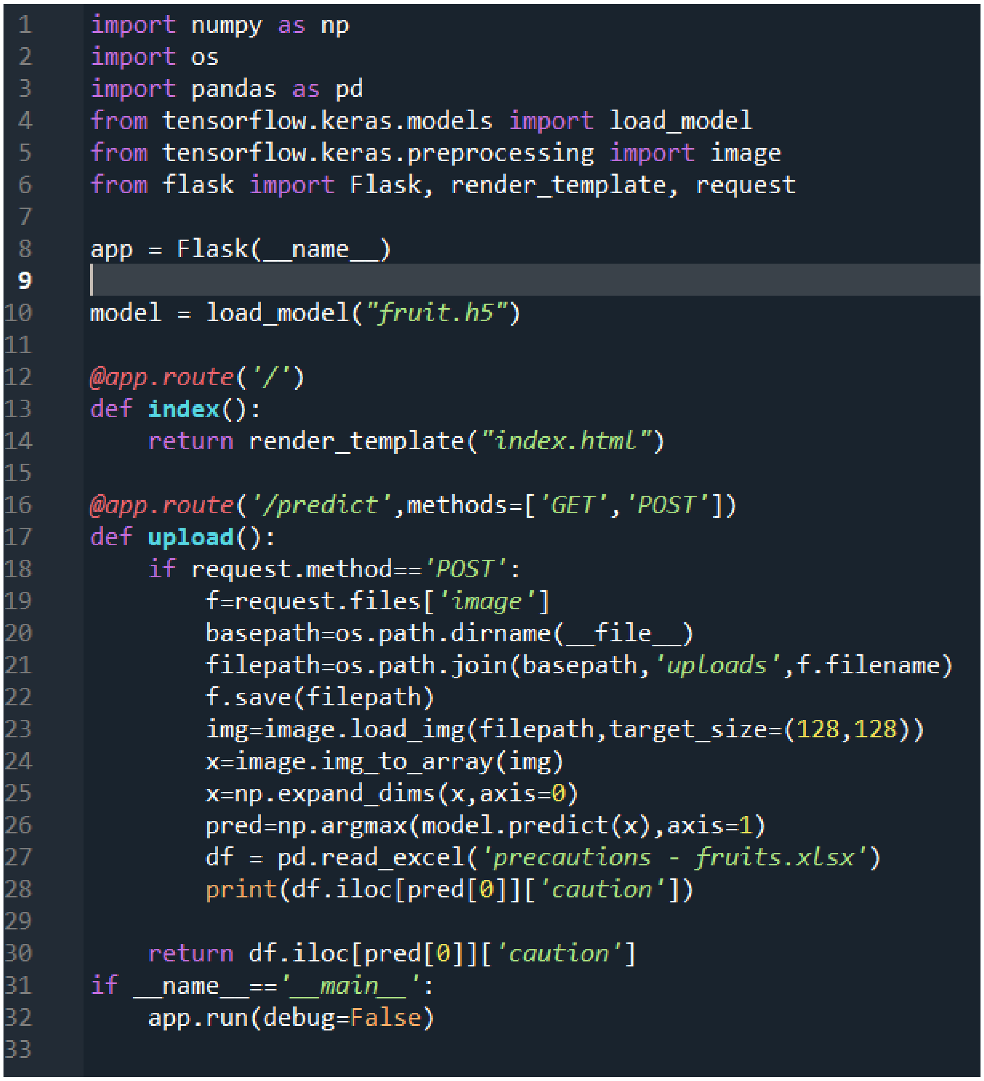
12. FUTURE SCOPE

Furthermore, this system can be made more robust by incorporating more image dataset with wider variations like more than one leaf in a single image. An App could also be developed for the project which could make the work of the farmers easier. They could directly upload image on the app and it would tell the disease and the cure then and there. This would reduce the time and efforts. 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 helpful for every agricultural need. Newer models can also be added and tried with time which may result in better accuracy and would make the model even faster.

13. APPENDIX

Source Code

Python – app_feretilizer.py



```python
import numpy as np
import os
import pandas as pd
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from flask import Flask, render_template, request

app = Flask(__name__)

model = load_model("fruit.h5")

@app.route('/')
def index():
    return render_template("index.html")

@app.route('/predict',methods=['GET','POST'])
def upload():
    if request.method=='POST':
        f=request.files['image']
        basepath=os.path.dirname(__file__)
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)
        img=image.load_img(filepath,target_size=(128,128))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)
        pred=np.argmax(model.predict(x),axis=1)
        df = pd.read_excel('precautions - fruits.xlsx')
        print(df.iloc[pred[0]]['caution'])

    return df.iloc[pred[0]]['caution']
if __name__=='__main__':
    app.run(debug=False)

```

index.html

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

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Fertilizer Recommented System and Disease Prediction</title>
    <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet">
    <script
src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script
src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
    <link href="{{ url_for('static', filename='css/main.css') }}"
rel="stylesheet">
 <style>

        .bg-dark {
            background-color: #7CFC00!important;
        }
        #result {
            color: #000f;
        }
 body
{
    background-image: url("https://wallpaperaccess.com/full/327148.jpg");
    background-size: cover;
}

 </style>
</head>

<body>

    <nav class="navbar navbar-dark bg-dark">
        <div class="container">
            <a class="navbar-brand" href="#"><center><b>CROP DISEASE
PREDICTION</b></center></a>
        </div>
    </nav>
    <div class="container">
        <div id="content" style="margin-top:2em">
        <div class="container">
          <div class="row">
              <div class="col-sm-6 bd">
                <h3>Plant Disease Prediction: </h3>
                <br>
                <p>Agriculture is one of the major sectors world wide over the
years it has developed and the use of new technologies and equipment replaced
almost all the traditional methods of farming . the plant diseases affect the
production , Identification of disesases and taking necessary precautions are
done through naked eye , which requires labour and labratries . This
application helps farmers in detecting thr diseases by observing the spots and
the leaves , which in turn saves effort and labour costs .</p>
                     <img src="http://www.geocities.ws/senthilirtt/First.jpg"
height="50%",width="20%">
              </div>
              <div class="col-sm-6">
                     <div>
                           <h4>Upload Image Here To Identify the Plant
Disease</h4>
              <form action = "http://localhost:5000/" id="upload-file"
method="post" enctype="multipart/form-data">
                     <label for="imageUpload" class="upload-label">
                           Choose...
                     </label>
                     <input type="file" name="image" id="imageUpload"
accept=".png, .jpg, .jpeg">
              </form>
              <div class="image-section" style="display:none;">
                     <div class="img-preview">
                           <div id="imagePreview">
                           </div>
                     </div>
                     <div>
                           <button type="button" class="btn btn-info btn-lg "
id="btn-predict">Predict!</button>
                     </div>
              </div>

              <div class="loader" style="display:none;"></div>

              <h3>
                     <span id="result"> </span>
              </h3>

        </div>
              </div>

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

<footer>
    <script src="{{ url_for('static', filename='js/main.js') }}"
type="text/javascript"></script>
</footer>

</html>
```

main.css

```
.img-preview {
    width: 256px;
    height: 256px;
    position: relative;
    border: 5px solid #F8F8F8;
    box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);
    margin-top: 1em;
    margin-bottom: 1em;
}

.img-preview>div {
    width: 100%;
    height: 100%;
    background-size: 256px 256px;
    background-repeat: no-repeat;
    background-position: center;
}

input[type="file"] {
    display: none;
}

.upload-label{
    display: inline-block;
    padding: 12px 30px;
    background: #39D2B4;
    color: #fff;
    font-size: 1em;
    transition: all .4s;
    cursor: pointer;
}

.upload-label:hover{
    background: #34495E;
    color: #39D2B4;
}

.loader {
    border: 8px solid #f3f3f3; /* Light grey */
    border-top: 8px solid #3498db; /* Blue */
    border-radius: 50%;
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

@keyframes spin {
    0% { transform: rotate(0deg); }
    100% { transform: rotate(360deg); }
}
```

main.js

```
 $(document).ready(function () {
     // Init
     $('.image-section').hide();
     $('.loader').hide();
     $('#result').hide();

     // Upload Preview
     function readURL(input) {
         if (input.files && input.files[0]) {
             var reader = new FileReader();
             reader.onload = function (e) {
                 $('#imagePreview').css('background-image', 'url(' +
 e.target.result + ')');
                 $('#imagePreview').hide();
                 $('#imagePreview').fadeIn(650);
             }
             reader.readAsDataURL(input.files[0]);
         }
     }
     $("#imageUpload").change(function () {
         $('.image-section').show();
         $('#btn-predict').show();
         $('#result').text('');
         $('#result').hide();
```

```
        readURL(this);
    });

    // Predict
    $('#btn-predict').click(function () {
        var form_data = new FormData($('#upload-file')[0]);

        // Show loading animation
        $(this).hide();
        $('.loader').show();

        // Make prediction by calling api /predict
        $.ajax({
            type: 'POST',
            url: '/predict',
            data: form_data,
            contentType: false,
            cache: false,
            processData: false,
            async: true,
            success: function (data) {
                // Get and display the result
                $('.loader').hide();
                $('#result').fadeIn(600);
                $('#result').text(' Result:  ' + data);
                console.log('Success!');
            },
        });
    });

});
```

Fruit Model

Image Pre-processing

```
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'C:\Users\RAJU\OneDrive\Desktop\Sathiya\IBM\Dataset
Plant Disease\fruit-dataset\fruit-
dataset\train',target_size=(128,128),batch_size=32,class_mode='categorical')
x_test=test_datagen.flow_from_directory(r'C:\Users\RAJU\OneDrive\Desktop\Sathiya\IBM\Datas
et Plant Disease\fruit-dataset\fruit-dataset\train',target_size=(128,128),
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPool2D,Flatten
model=Sequential()
```

Adding CNN Layers

```
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
x_train.class_indices
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(40, kernel_initializer='uniform',activation='relu'))
model.add(Dense(20, kernel_initializer='random_uniform',activation='relu'))
model.add(Dense(6,activation='softmax', kernel_initializer='random_uniform'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
5384/32
```

Fitting The Model

```
model.fit_generator(x_train,steps_per_epoch=168,validation_data=x_test,validation_steps=52,epochs=3)
model.save("fruit.h5")
ls
from keras.preprocessing import image
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
import numpy as np
model = load_model("fruit.h5")
```

Testing Model

```
import keras.utils as image
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/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)
pred = np.argmax(model.predict(x),axis=1)
pred
```

```
index
=['Apple___Black_rot','Apple___healthy','Corn_(maize)___Northern_Leaf_Blight','Corn_(maize)_
__healthy','Peach___Bacterial_spot','Peach___healthy']
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant
Disease/fruit-dataset/fruit-dataset/test/Apple___Black_rot/0f3d45f4-e121-42cd-a5b6-
be2f866a0574___JR_FrgE.S 2870.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant
Disease/fruit-dataset/fruit-dataset/test/Corn_(maize)___Northern_Leaf_Blight/00a14441-7a62-
4034-bc40-b196aeab2785___RS_NLB 3932.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant
Disease/fruit-dataset/fruit-dataset/test/Corn_(maize)___healthy/0a68ef5a-027c-41ae-b227-
159dae77d3dd___R.S_HL 7969 copy.jpg',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant
Disease/fruit-dataset/fruit-dataset/test/Peach___Bacterial_spot/00ddc106-692e-4c67-b2e8-
569c924caf49___Rutg._Bact.S 1228.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
```

```
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant
Disease/fruit-dataset/fruit-dataset/test/Peach___healthy/1a07ce54-f4fd-41cf-b088-
144f6bf71859___Rutg._HL 3543.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
```

Vegetable Model

Image Pre-processing

```
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'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset
Plant Disease/veg-dataset/veg-
dataset/train_set',target_size=(128,128),batch_size=16,class_mode='categorical')
x_test=test_datagen.flow_from_directory(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Datas
et Plant Disease/veg-dataset/veg-dataset/test_set',target_size=(128,128),
                                   batch_size=16,class_mode='categorical')
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPool2D,Flatten
model=Sequential()
```

Adding CNN Layers

```
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
x_train.class_indices
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(300, kernel_initializer='uniform',activation='relu'))
model.add(Dense(150, kernel_initializer='uniform',activation='relu'))
model.add(Dense(75, kernel_initializer='uniform',activation='relu'))
model.add(Dense(9,activation='softmax', kernel_initializer='uniform'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
11386/16
```

Fitting The Model

```
model.fit_generator(x_train,steps_per_epoch=89,validation_data=x_test,validation_steps=27,epochs=20)
model.save("vegetable.h5")
ls
model.summary()
```

Test The Model

```
from keras.preprocessing import image
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
import numpy as np
model = load_model("vegetable.h5"
index=['Pepper,_bell___Bacterial_spot','Pepper,_bell___healthy','Potato___Early_blight','Potato___Late_blight','Potato___healthy','Tomato___Bacterial_spot','Tomato___Late_blight','Tomato___Leaf_Mold','Tomato___Septoria_leaf_spot']
import keras.utils as image
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Pepper,_bell___Bacterial_spot/ad921dec-e88f-41d8-9455-0880c69063fc___NREC_B.Spot 9216.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Pepper,_bell___healthy/b45d62a2-3de1-411b-8f88-ab52195b6dda___JR_HL 7639.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
```

```
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Potato___Early_blight/b6220993-c51f-48fa-bee9-fb5cb89cc0c4___RS_Early.B 7467.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Potato___Late_blight/d20fef22-7016-4115-9930-a66b32c718c6___RS_LB 4417.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Potato___healthy/ff700844-68ad-4e99-8427-58a39c07f817___RS_HL 1860.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Tomato___Bacterial_spot/b433dcf3-c57d-4f5e-9116-5aaeecbaef01___GCREC_Bact.Sp 3715.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Tomato___Late_blight/b19055d9-6a5d-4b73-90a5-3b3685ac344a___GHLB2 Leaf 8729.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
```

```
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Tomato___Leaf_Mold/d8f27502-b864-4da3-9dd5-b7eccf2e8cbc___Crnl_L.Mold 6708.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
img = image.load_img(r'C:/Users/RAJU/OneDrive/Desktop/Sathiya/IBM/Dataset Plant Disease/veg-dataset/veg-dataset/test_set/Tomato___Septoria_leaf_spot/c3233f4f-a6f5-4bfb-a6bf-bd5e5d045f7a___JR_Sept.L.S 2728.JPG',target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)
print('the given image belogs to=',index[pred[0]])
```

GitHub Link

GitHub - IBM-EPBL/IBM-Project-7873-1658901481: Fertilizers ...

Project Demo Link

https://drive.google.com/file/d/1awEGqXDaPWLYEUZWmDd0pbzQXSd44iVS/view?usp=sharing