FERTILIZER RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

Team Id : PNT2022TMID29439
Domain : Artificial Intelligence

Department of Computer Science and Engineering

Submitted by,

Team Leader : Hemadharshini G(510419104032)

Team Member1 : Hemalatha R (510419104033)

Team Member2 : Gnanasoundariya S(510419104028)

Team Member3 : Sandhiya G(510419104058)

ARUNAI ENGINEERING COLLEGE
Velu nagar, Tiruvannamalai

Project Report Format

1. **INTRODUCTION**
   - 1.1 Project Overview
   - 1.2 Purpose
2. **LITERATURE SURVEY**
   - 2.1 Existing problem
   - 2.2 References
   - 2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   - 3.1 Empathy Map Canvas
   - 3.2 Ideation & Brainstorming
   - 3.3 Proposed Solution
   - 3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   - 4.1 Functional requirement
   - 4.2 Non-Functional requirements
5. **PROJECT DESIGN**
   - 5.1 Data Flow Diagrams
   - 5.2 Solution & Technical Architecture
   - 5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
   - 6.1 Sprint Planning & Estimation
   - 6.2 Sprint Delivery Schedule
   - 6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   - 7.1 Feature 1
   - 7.2 Feature 2
   - 7.3 Database Schema (if Applicable)
8. **TESTING**
   - 8.1 Test Cases
   - 8.2 User Acceptance Testing
9. **RESULTS**
   - 9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

1.INTRODUCTION

1.1 Project Overview

Agriculture is the most important sector in today's life. Most plants are affected by a wide variety of 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.

1.2 Purpose

- In agricultural aspects, if the plant is affected by leaf disease then it reduces the growth of the agricultural level.
- Finding the leaf disease is an important role of agriculture preservation.
- The disease-based similarity measure is used for fertilizer recommendation.
- Detection and recognition of plant diseases using machine learning are very efficient in providing symptoms of identifying diseases at its earliest.
- Agricultural productivity is something on which economy highly depends. This is the one of the reasons that disease detection in plants plays an important role in agriculture field, as having disease in plants are quite natural.

2.LITERATURE SURVEY

2.1 Existing problem

- 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.
- 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.
- The Indian agriculture is facing a number of hurdles because of the change of climate, water pollution/ shortages, lack of fertilizers, old methods/technologies, different plant's diseases and many more.
- 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.

2.2 References

[1] Sue Han, CheeSeng Chan, Paul Wilkin, and Paolo Remagnino, "Deep-plant: Plant identification with convolutional neural networks", In Image Processing INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 8, ISSUE 11, NOVEMBER 2019 ISSN 2277-8616 3346 IJSTR©2019 www.ijstr.org (ICIP), 2015 IEEE International Conference on, pp. 452-456, IEEE, 2015.

[2] Zeiler, Matthew D., and Rob Fergus, "Visualizing and understanding convolutional networks", In European conference on computer vision, pp. 818-833. Springer, Cham, 2014.

[3] Vijayashree .T and Gopal .A, "Authentication of Leaf Image Using Image Processing Technique", ARPN Journal of Engineering and Applied Sciences 10, no. 9, pp: 4287-4291, 2015.

[4] Satnam Singh and Manjit Singh Bhamrah, "Leaf identification using feature extraction and neural network", IOSR Journal of Electronics and Communication Engineering 5, pp: 134-140, 2015.

[5] Reyes Angie .K, Juan C. Caicedo, and Jorge E. Camargo, "Fine-tuning Deep Convolutional Networks for Plant Recognition", In CLEF (Working Notes), 2015.

2.3 Problem Statement Definition

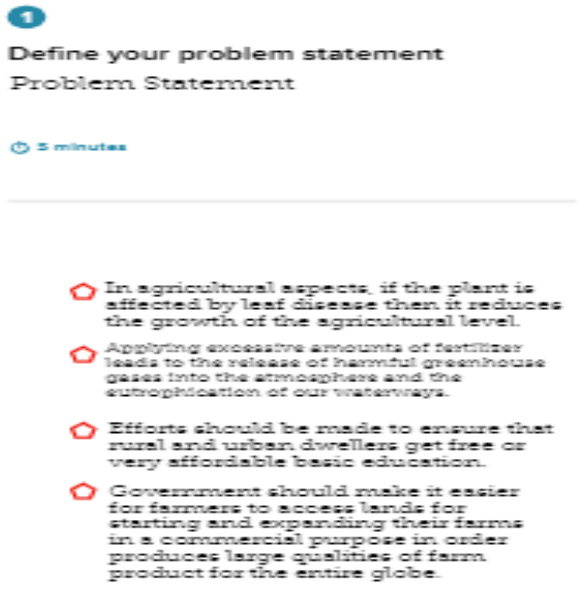
- In agricultural aspects, if the plant is affected by leaf disease then it reduces the growth of the agricultural level.
- Applying excessive amounts of fertilizer leads to the release of harmful greenhouse gases into the atmosphere and the eutrophication of our waterways.
- Efforts should be made to ensure that rural and urban dwellers get free or very affordable basic education.
- Government should make it easier for farmers to access lands for starting and expanding their farms in a commercial purpose in order produces large qualities of farm product for the entire globe.
- For the fertilizer recommendation application, the user can input the soil data and the type of crop they are growing, and the application will predict what the soil lacks or has excess of and will recommend improvements.
- Most of the Indian population depends on agriculture for their livelihood. Agriculture gives an opportunity of employment to the village people to develop a country like India on large scale and give a push in the economic sector.

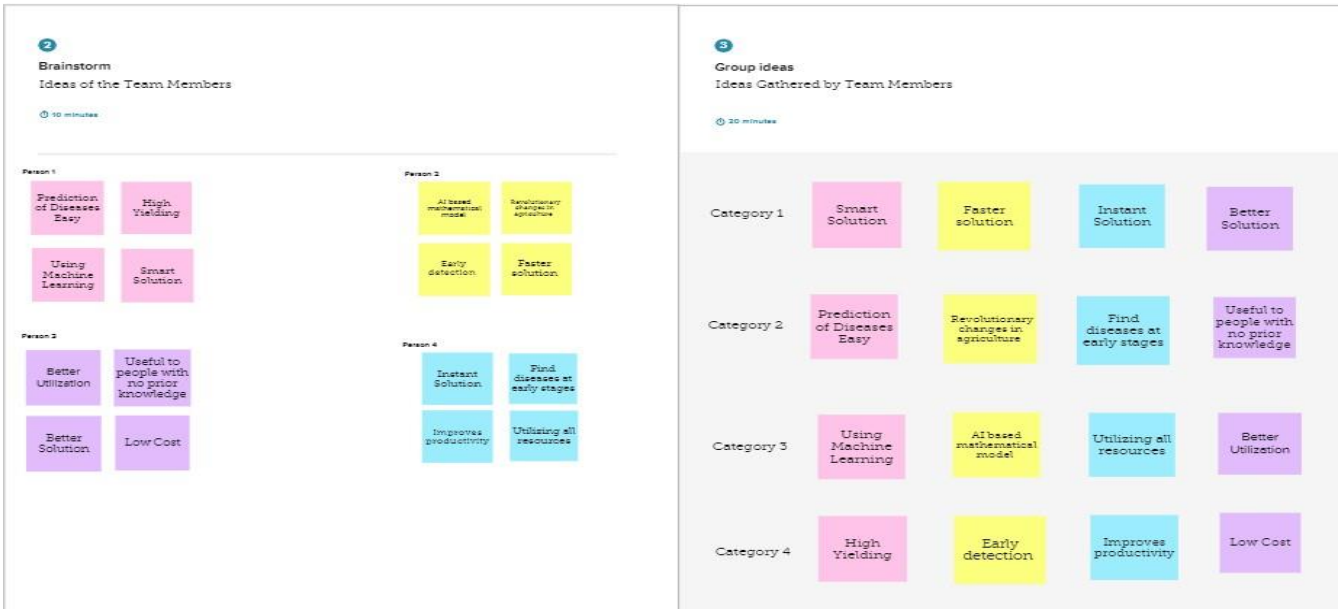
.

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming




2
Brainstorm
Ideas of the Team Members
10 minutes
Person 1
Prediction of Diseases Easy
High Yielding
Using Machine Learning
Smart Solution
Person 2
AI based mathematical model
Revolutionary changes in agriculture
Early detection
Faster solution
Person 3
Better Utilization
Useful to people with no prior knowledge
Better Solution
Low Cost
Person 4
Instant Solution
Find diseases at early stages
Improves productivity
Utilizing all resources
3
Group ideas
Ideas Gathered by Team Members
20 minutes
Category 1
Smart Solution
Faster solution
Instant Solution
Better Solution
Category 2
Prediction of Diseases Easy
Revolutionary changes in agriculture
Find diseases at early stages
Useful to people with no prior knowledge
Category 3
Using Machine Learning
AI based mathematical model
Utilizing all resources
Better Utilization
Category 4
High Yielding
Early detection
Improves productivity
Low Cost


4
Prioritize
Prioritize levels
20 minutes
Prediction of Diseases Easy
Improves productivity
AI based mathematical model
Useful to people with no prior knowledge
Better Solution

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	➢ Agriculture-Importance sector in Today's life. In Agricultural aspects if plant affected by disease then it reduce the level of Agriculture. ➢ Disease on plants-Major constraint on production and major threat to food security. ➢ Fertilizers- Farmers does not find the availability of where the Fertilizers stock is currently available. Application- In agricultural aspects, there is no smart way to predict the plant disease and problem solving way by the farmers.
2.	Idea / Solution description	➢ Disease- Effect on less amount of Agriculture production and compromises with quality. ➢ Fertilizers- By using the smart application they can easily track the agro services. ➢ Application- Used to solve a specific plant problem.
3.	Novelty / Uniqueness	➢ Simple to understand ➢ Smooth to use ➢ Effortless ➢ User friendly
4.	Social Impact / Customer Satisfaction	➢ Easy to predict the disease before the crop dead. Searching of nearby fertilizers shop is easy. ➢ Allow crops to grow bigger, faster and to produce more food.
5.	Business Model (Revenue Model)	➢ Predicting the fertilizers, Analyzing the disease in a tap makes the life of farmers 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.
6.	Scalability of the Solution	➢ High yield response ➢ Reliable ➢ Cost effective

3.4 Problem Solution fit

Define CS, fit into CC

1. CUSTOMER SEGMENT CS

FARMERS

6. CUSTOMER CONSTRAINTS CC

- Mobile phone
- Network connection
- No cost
- Basic knowledge
- Free search
- User friendly

5. AVAILABLE SOLUTIONS AS

- To recommend the fertilizer for the current state of the soil.
- By calculating the pH and moisture content in soil.
- Soil analysis to increase the crop yield.
- Based on the soil report, Fertilizer will be recommended to the user.

Explore AS, differentiate

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

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

- Agriculture-Importance sector in Today's life. In Agricultural aspects if plant affected by disease then it reduce the level of Agriculture.
- Disease on plants-Major constraint on production and major threat to food security.
- Fertilizers- Farmers does not find the availability of where the Fertilizers stock is currently available.
- Application- In agricultural aspects, there is no smart way to predict the plant disease and problem solving way by the farmers.

9. PROBLEM ROOT CAUSE RC

- Significant increases in algae harm water quality.
- Increases food resources and habitats.
- Limit visibility and alter plant growth.

7. BEHAVIOUR BE

- Soil parameters and temperature.
- Leaf wetness.
- Weather stations.
- Relative humidity.
- Due to fungus.
- Determine risky periods.

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

3. TRIGGERS TR

Seeing neighbour farmer using smart application and get inspired.

4. EMOTIONS: BEFORE / AFTER EM

Before:

- Spraying wrong fertilizers leads to heavy plant damage.
- During climatic change, farmers does not earn proper yield.

After:

- By using smart applications we get better knowledge of fertilizers.
- Easy to predict the disease.

10. YOUR SOLUTION SL

- ✓ Disease- Effect on less amount of Agriculture production and compromises with quality.
- ✓ Fertilizers- By using the smart application they can easily track the agro services.
- ✓ Application- Used to solve a specific plant problem.

8. CHANNELS of BEHAVIOUR CH

ONLINE

- Farmers take the picture of affected plant and send it by using online applications
- Online platform shows the nearby fertilizer shop.
- Predicting the type of fertilizer to be used depending on the crop type and soil.

OFFLINE

- By the help of applications they have to use the fertilizers in correct manner
- Harvesting crops on time.

4. REQUIREMENT ANALYSIS

4.1 Functional requirements

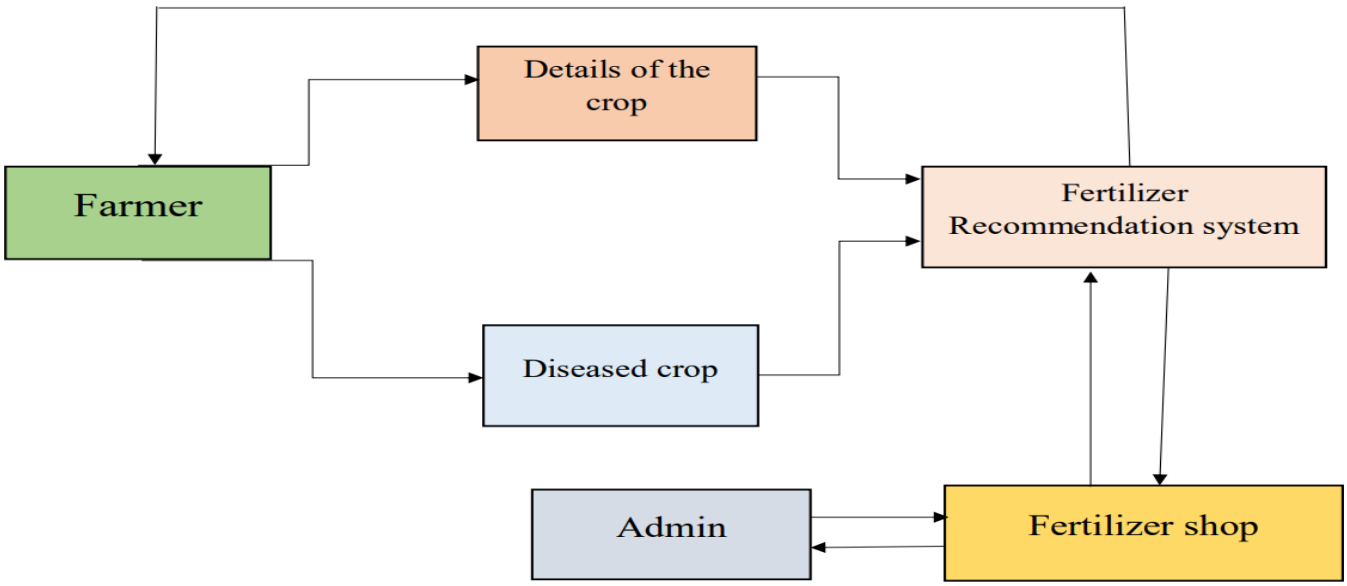
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Confirmation	Confirmation via Email
FR-3	User Profile	Filling the profile page after logging in.
FR-4	Uploading Dataset(Leaf)	Images of the leaves are to be uploaded.
FR-5	Requesting solution	Uploaded images is compared with the pre-defined Model andsolution is generated.
FR-6	Downloading Solution	The solution in pdf format which contains the recommendations offertilizers and the possible diseases.

4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system allows the user to perform the tasks easily and efficiently and effectively.
NFR-2	Security	Assuring all data inside the system or its part will be protected against malware attacks or unauthorized access.
NFR-3	Reliability	The website does not recover from failure quickly,ittakes time as the application is running in single server.
NFR-4	Performance	Response Time and Net Processing Time is fast.
NFR-5	Availability	The system will be available up to 95% of the time.
NFR-6	Scalability	The website is scalable.

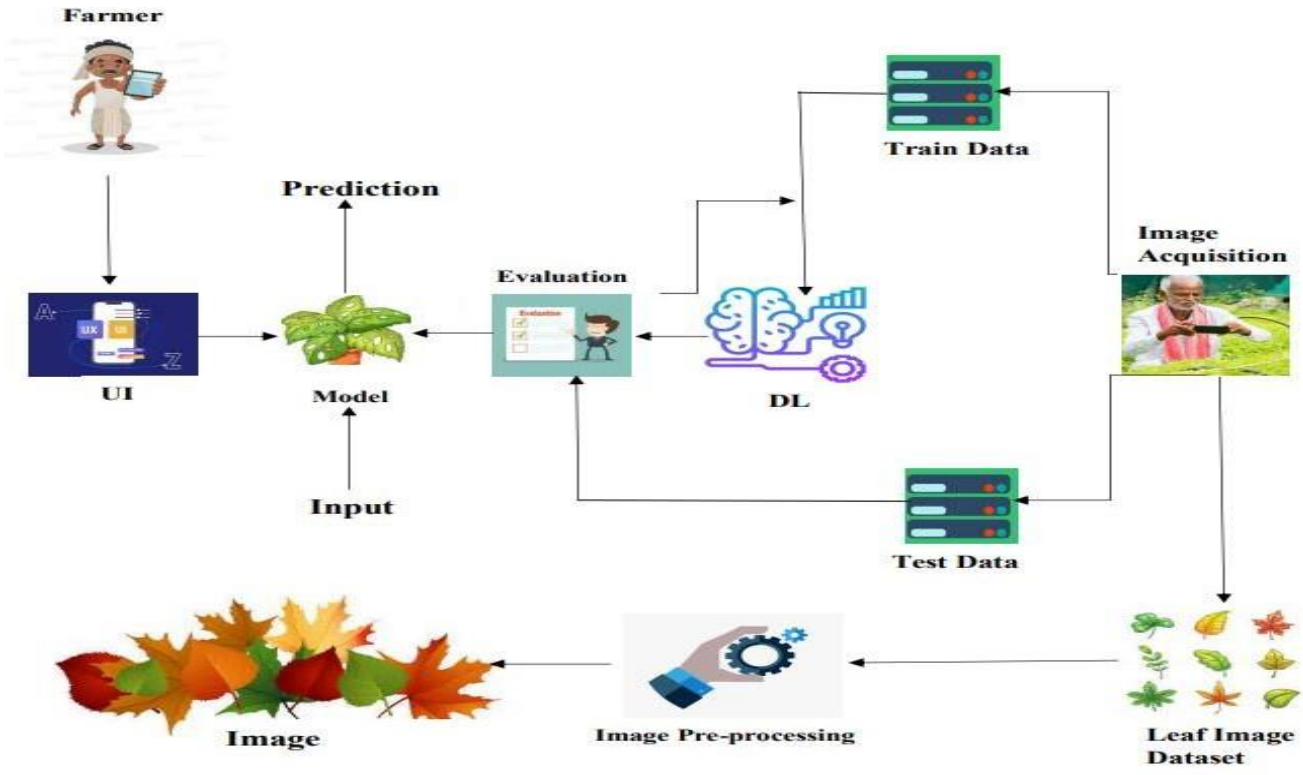
5. PROJECT DESIGN

5.1 Data Flow Diagrams

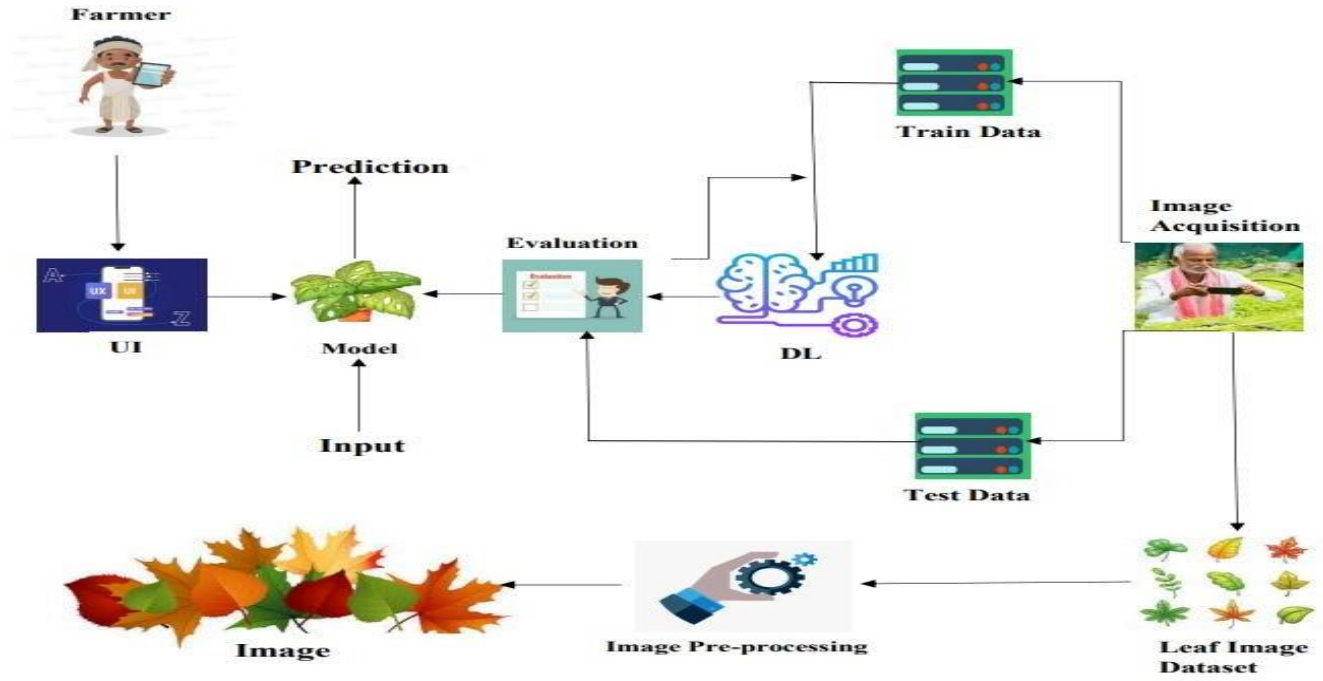


5.2 Solution & Technical Architecture

Solution Architecture



Technical Architecture



5.3User 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 log into the application by entering email & password	I can receive confirmation email& click confirm	High	Sprint-1
	Dashboard	USN-3	As a user, I can view the page of theapplication where I can upload my images and Fertilizers should be recommended.	I can Access myaccount	High	Sprint-2
Customer (Web user)	Registration	USN-4	As a user, I can open the website link by touching alink.	I can register using password,username.	High	Sprint-3
	Login	USN-5	As a user, I can login to my web dashboard withinthe login credentials.	I can login using myuser credentials	High	Sprint-3

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset collection	USN-1	Collect the dataset from the user	6	High	Hemadharshini.G, Hemalatha.R
	Image Preprocessing	USN-2	Process the images which can be collected from the user	4	High	Gnanasoundariya.S, Sandhiya.G
Sprint-2	Model Building for fruit disease prediction	USN-4	Create a model which can classify diseased fruit plants from given images.I also need to test the model and deploy it on IBM Cloud	4	High	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Fruit Dataset	USN-5	Datasets with fruits	2	Low	Hemadharshini.G ,Hemalatha.R,
	Model Building for vegetable disease prediction	USN-6	Create a model which can classify diseased vegetable plants from given images and train on IBM Cloud	3	Medium	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Vegetable Dataset	USN-7	Datasets with vegetables	4	High	Hemadharshini.G, Hemalatha.R,
Sprint-3	Test both models	USN-8	Test the both models using the datasets	2	High	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Application buildings	USN-9	Build the application	2	High	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Registration	USN-10	As a user/admin/shopkeeper,I can log into the application by entering email & password	3	Medium	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Redirect the page	USN-11	Page can be redirected to Another page	5	Medium	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Upload Image	USN-12	Where we can upload the diseased fruit images	2	Low	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
	Recommended Result	USN-13	As an admin,I can view other user details and uploads for other purposes	4	Medium	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G
Sprint-4	Train the models on IBM cloud	USN-14	Created model can be tested by IBM cloud	10	High	Hemadharshini.G, Hemalatha.R, Gnanasoundariya.S Sandhiya.G

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	10	31 Oct 2022
Sprint-2	13	6 Days	31 Oct 2022	05 Nov 2022	13	06 Nov 2022
Sprint-3	18	6 Days	07 Nov 2022	12 Nov 2022	18	13 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

6.3 Reports from JIRA

7.CODING & SOLUTIONING

7.1 Feature Code 1

home.html

```
<!DOCTYPE html>
<html >

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Plant Disease Prediction</title>
<link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
<style>
.header {
                        top:0;
                        margin:0px;
                        left: 0px;
                        right: 0px;
                        position: fixed;
                        background-color: #28272c;
                        color: white;
                        box-shadow: 0px 8px 4px grey;
                        overflow: hidden;
                        padding-left:20px;
                        font-family: 'Josefin Sans';
                        font-size: 2vw;
                        width: 100%;
                        height:8%;
                        text-align: center;
                }
                .topnav {
  overflow: hidden;
  background-color: #333;
}

.topnav-right a {
  float: left;
  color: #f2f2f2;
  text-align: center;
```

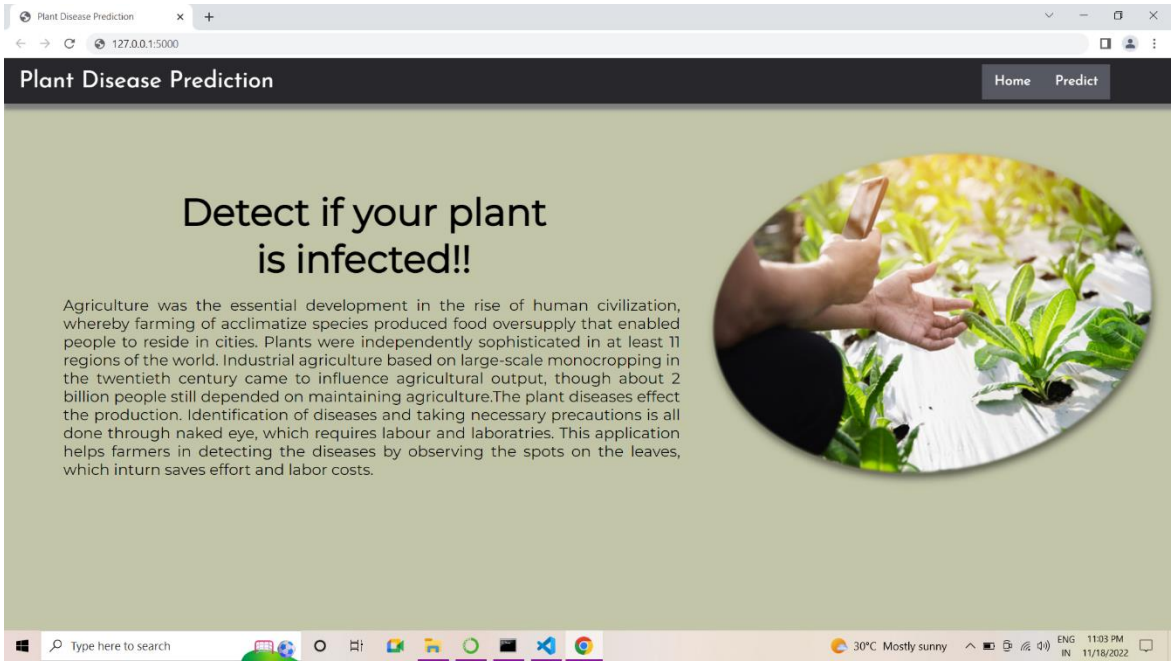
```
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #ddd;
  color: black;
}

.topnav-right a.active {
  background-color: #565961;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}

body {

  background-color:#ffffff;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #28272c;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  border-radius: 12px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}
input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
```

```
}
button {
  background-color: #28272c;
  color: white;
  padding: 14px 20px;
  margin-bottom:8px;
  border: none;
  cursor: pointer;
  width: 15%;
  border-radius:4px;

}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
```

```
  }
}

.home{
        margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
        margin:80px;
        box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
box-sizing: content-box;
height: 400px;
margin:20px;
border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
```

```
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}
.active {
  background-color: #717171;
}
/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}
@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}
@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}
/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
</style>
</head>
<body style="font-family:'Times New Roman', Times, serif;background-color:#C2C5A8;">
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:white; padding-top:1%">Plant Disease Prediction</div>
  <div class="topnav-right"style="padding-top:0.5%;">
    <a class="active" href="{{ url_for('home')}}">Home</a>
    <a href="{{ url_for('prediction')}}">Predict</a>
  </div>
</div>
<div style="background-color:#ffffff;">
<div style="width:60%;float:left;">
<div style="font-size:50px;font-family:Montserrat;padding-left:20px;text-align:center;padding-top:10%;">
<b>Detect if your plant<br> is infected!!</b></div><br>
<div style="font-size:20px;font-family:Montserrat;padding-left:70px;padding-right:30px;text-align:justify;">Agriculture is one of the major sectors worls 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 effect the production. Identification of diseases and taking necessary precautions is all done through naked eye, which requires labour and laboratries. This application helps farmers in detecting the diseases by observing the spots on the leaves, which inturn saves effort and labor costs.</div><br><br>
```

```
</div>
</div>
<div style="width:40%;float:right;"><br><br>
<img src="{{url_for('static',filename='images/12456.png')}}" style="max-height:100%;max-
width:100%;">
</div>
</div>
<div class="home">
<br>
</div>
<script>
var slideIndex = 0;
showSlides();
function showSlides() {
  var i;
  var slides = document.getElementsByClassName("mySlides");
  var dots = document.getElementsByClassName("dot");
  for (i = 0; i < slides.length; i++) {
    slides[i].style.display = "none";
  }
  slideIndex++;
  if (slideIndex > slides.length) {slideIndex = 1}
  for (i = 0; i < dots.length; i++) {
    dots[i].className = dots[i].className.replace(" active", "");
  }
  slides[slideIndex-1].style.display = "block";
  dots[slideIndex-1].className += " active";
  setTimeout(showSlides, 2000); // Change image every 2 seconds
}
</script>
</body>
</html>
```

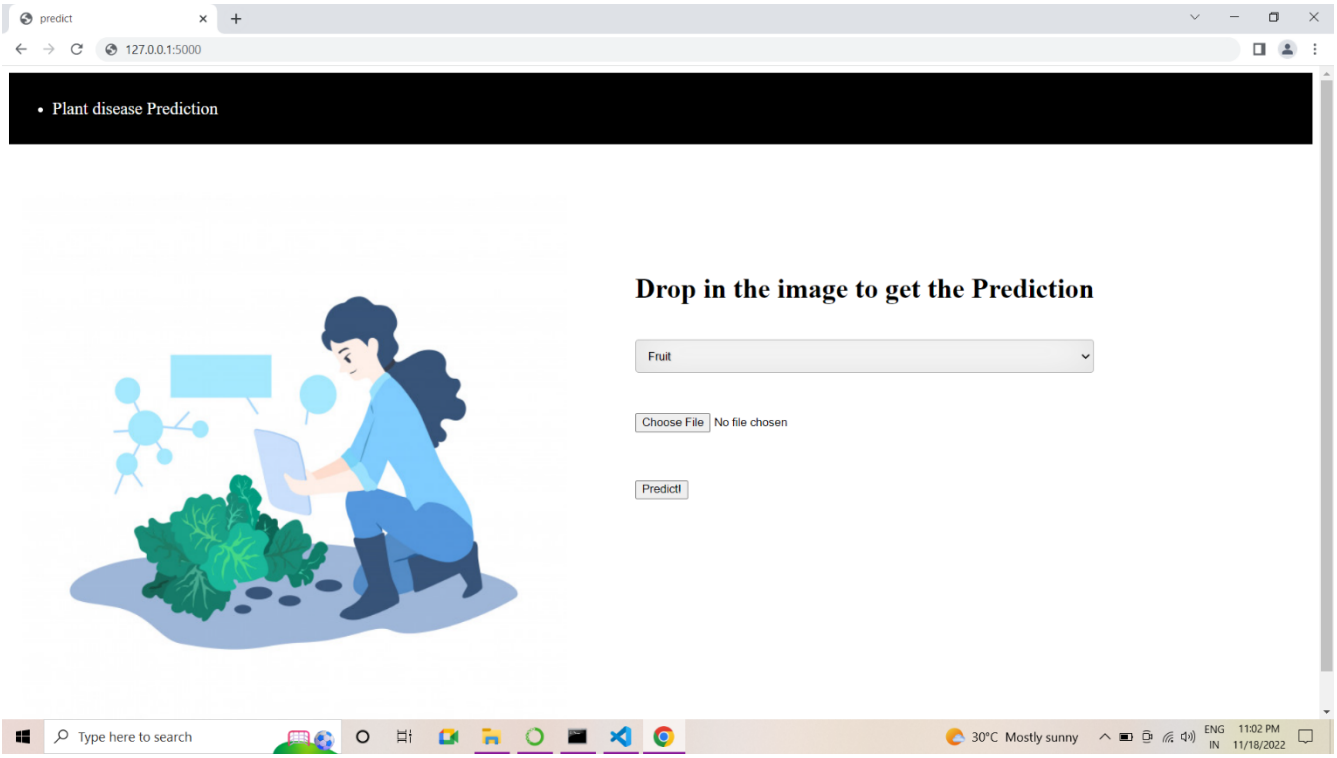


Predict.html

```
<!DOCTYPE html>
<html >

<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Plant Disease Prediction</title>
<link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<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
src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
<link href="{{ url_for('static', filename='css/final.css') }}" rel="stylesheet">
<style>
.header {
                        top:0;
                        margin:0px;
                        left: 0px;
                        right: 0px;
                        position: fixed;
                        background-color: #28272c;
                        color: white;
                        box-shadow: 0px 8px 4px grey;
                        overflow: hidden;
                        padding-left:20px;
                        font-family: 'Josefin Sans';
                        font-size: 2vw;
                        width: 100%;
                        height:8%;
                        text-align: center;
                }
                .topnav {
  overflow: hidden;
  background-color: #333;
}

.topnav-right a {
  float: left;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
```

```
  background-color: #ddd;
  color: black;
}

.topnav-right a.active {
  background-color: #565961;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}

.login{
margin-top:-70px;
}
body {

  background-color:#ffffff;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
.login{
        margin-top:100px;
}

.container {
  margin-top:40px;
  padding: 16px;
}
select {
        width: 100%;
        margin-bottom: 10px;
        background: rgba(255,255,255,255);
        border: none;
        outline: none;
        padding: 10px;
        font-size: 13px;
        color: #000000;
        text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
        border: 1px solid rgba(0,0,0,0.3);
        border-radius: 4px;
        box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
        -webkit-transition: box-shadow .5s ease;
        -moz-transition: box-shadow .5s ease;
        -o-transition: box-shadow .5s ease;
        -ms-transition: box-shadow .5s ease;
        transition: box-shadow .5s ease;
}

</stvle>
</head>
<body style="font-family:Montserrat;overflow:scroll;">

<div class="header">
```

```
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:white; padding-top:1%">Plant Disease Prediction</div>
  <div class="topnav-right" style="padding-top:0.5%;">
  </div>
</div>
<div class="container">
      <div id="content" style="margin-top:2em">
                <div class="container">
                  <div class="row">
                        <div class="col-sm-6 bd" >
                          <br>
                              <img src="{{url_for('static',filename='images/789.jpg')}}" style="height:450px;width:550px"class="img-rounded" alt="Gesture">
                        </div>
                        <div class="col-sm-6">
                              <div>
                                    <h4>Drop in the image to get the prediction  </h4>
                        <form action = "" id="upload-file" method="post" enctype="multipart/form-data">
                              <select name="plant">
                                    <option value="select" selected>Select plant type</option>
                                    <option value="fruit">Fruit</option>
                                    <option value="vegetable">Vegetable</option>
                </select><br>
                              <label for="imageUpload" class="upload-label" style="background: #28272c;">
                                    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" style="background: #28272c;">Predict!</button>
                              </div>
                        </div>
                        <div class="loader" style="display:none;"></div>
                        <h3>
                              <span id="result" style="font-size:17px; "> </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>
```



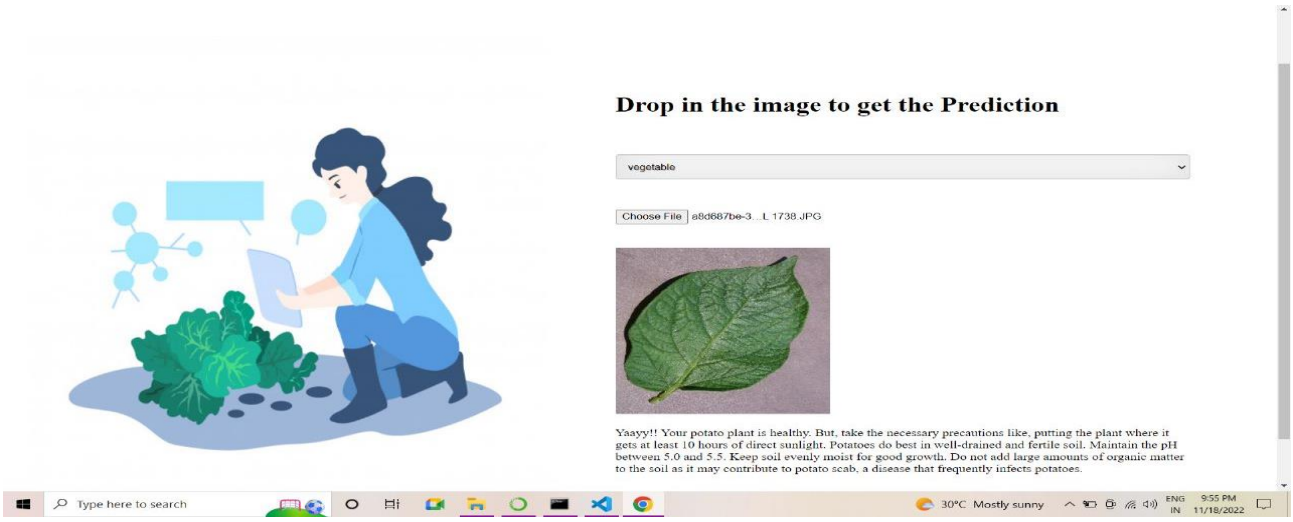
7.2 FEATURE CODE

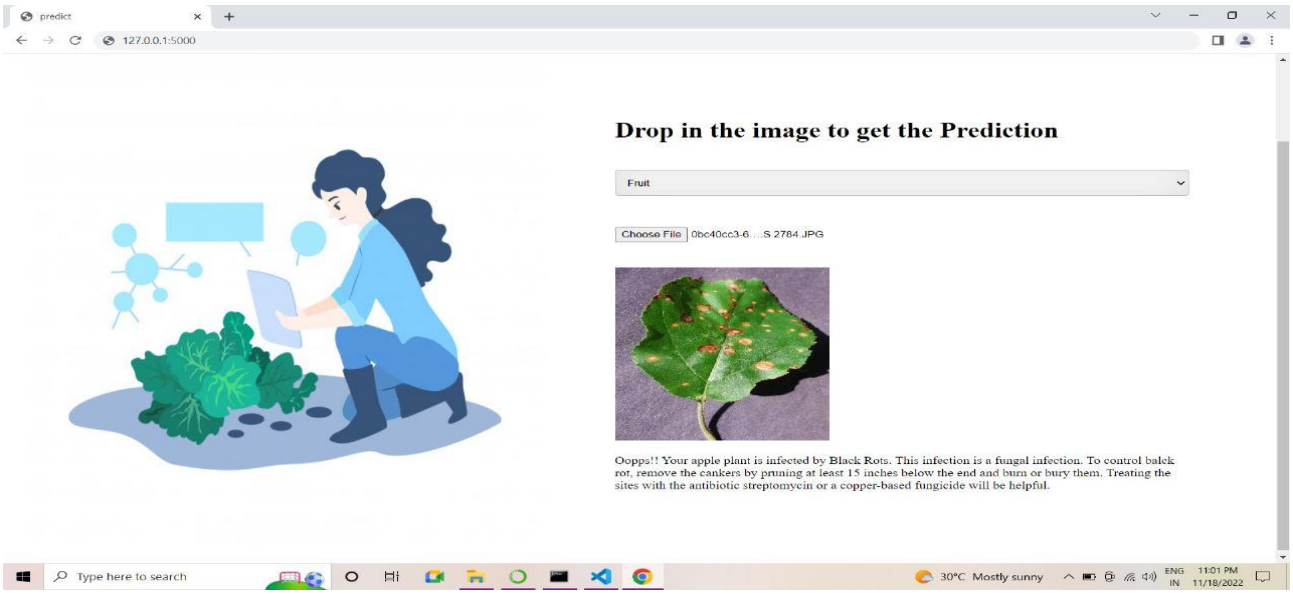
app.py

```
import requests
from tensorflow.keras.preprocessing import image
from tensorflow.keras.models import load_model
import numpy as np
import pandas as pd
import tensorflow as tf
from flask import Flask, request, render_template, redirect, url_for
import os
from werkzeug.utils import secure_filename
from tensorflow.python.keras.backend import set_session

app = Flask(__name__)

#load both the vegetable and fruit models
model = load_model("vegetable.h5")
model1=load_model("fruit.h5")

#home page
@app.route('/')
def home():
    return render_template('home.html')

#prediction page
@app.route('/prediction')
```

```
def prediction():
    return render_template('predict.html')

@app.route('/predict',methods=['POST'])
def predict():

    if request.method == 'POST':
        # Get the file from post request
        f = request.files['image']

        # Save the file to ./uploads
        basepath = os.path.dirname(__file__)
        file_path = os.path.join(
            basepath, 'uploads', secure_filename(f.filename))
        f.save(file_path)
        img = image.load_img(file_path, target_size=(128, 128))

        x = image.img_to_array(img)
        x = np.expand_dims(x, axis=0)

        plant=request.form['plant']
        print(plant)
        if(plant=="vegetable"):
            preds = model.predict(x)
            preds=np.argmax(preds)
            print(preds)
            df=pd.read_excel('precautions - veg.xlsx')
            print(df.iloc[preds]['caution'])
        else:
            preds = model1.predict(x)
            preds=np.argmax(preds)
            df=pd.read_excel('precautions - fruits.xlsx')
            print(df.iloc[preds]['caution'])


        return df.iloc[preds]['caution']

if __name__ == "__main__":
    app.run(debug=False)
```

Output





8.TESTING

8.1 Test Cases

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

This reports hows the number of resolve do closed bugs at each severity level, and how they were resolved

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
Leaf spots	10	4	2	3	19
Mosaic leaf pattern	9	6	3	6	24

Misshapen leaves	2	7	0	1	10
Yellow leaves	11	4	3	20	38
Fruit rots	3	2	1	0	6
Fruit spots	5	3	1	1	10
Blights	4	5	2	1	12
Totals	44	31	13	32	119

8.2 User Acceptance Testing

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

Section	TotalCases	Not Tested	Fail	Pass
Leaf spots	17	0	0	17
Mosaic leaf pattern	51	0	0	51
Misshapen leaves	20	0	0	20
Yellow leaves	7	0	0	7
Fruit rots	9	0	0	9
Fruit spots	4	0	0	4
Blights	2	0	0	2

9. RESULTS

9.1 Performance Metrics

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values

1.	Model Summary	To evaluate object detection models like R-CNN and velocity. It detect the plant disease and give the prediction
2.	Accuracy	Training Accuracy – 89% Validation Accuracy -95%

10. ADVANTAGES & DISADVANTAGES

Advantages:

➤ Agricultural productivity is somethin on which economy highly depends.

➤ To make economic decisions about disease treatments for control.

➤ The plant disease can be identified at early stage or the initial stage.

➤ Identifying the leaf disease and preferring the fertilizer were all carried out manually.

Disadvantages:

➤ They harm the micro be present in soil.

➤ They reduces oil fertility.

Defected plants may not contain primary nutrients like nitrogen, phosphorous or potassium.

11.CONCLUSION

➤ This paper presents the survey on different diseases classification techniques used for plant leaf disease detection and an algorithm for image segmentation technique that can be used for automatic detection as well as classification of plant leaf diseases later.

➤ With very less computational efforts the optimum results were obtained, which also shows the efficiency of proposed algorithm in recognition and classification of the leafdiseases.

➤ Another advantage of using this method is that the plant diseases can be identified at early stage or the initial stage.

➤ To improve recognition rate in classification process Artificial Neural Network algorithm can be used.

12. FUTURE SCOPE

- This project is implementing the proposed algorithm with the existing public datasets.Also,various segmentation algorithms can be implemented to improve accuracy.
- The proposed algorithm can be modified further to identify the disease that affects the various plant organs such as stems and fruits.
- It will be implemented in many languages.
- In this project, we can able to do both the fertilizer recommendation and crop recommendation.
- It can able to predict the near by fertilizer shop.

14.APPENDIX

Source Code :

```
 from tensorflow.keras.preprocessing.image import ImageDataGenerator

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"C:\Users\maris_q3mm6nk\Desktop\FILES\data_for_ibm\Fertilizers_Recommendation_ System_For_Disease_ Prediction\Dataset 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'C:\Users\maris_q3mm6nk\Desktop\FILES\data_for_ibm\Fertilizers_Recommendation_ System_For_Disease_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\test_set',target_size=(128,128),
                                          class_mode='categorical',batch_size=24)

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten

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=['accuracy'])

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')

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

model=load_model('vegetabledata.h5')

img=image.load_img(r"path")

img

x=image.img_to_array(img)

img=image.load_img(r"image path",target_size=(128,128))
img

x=image.img_to_array(img)

x

x=np.expand_dims(x,axis=0)

x

y=np.argmax(model.predict(x),axis=1)
1/1 [==============================] - 0s 89ms/step
```

Github & Project Demo Link

Github : https://github.com/IBM-EPBL/IBM-Project-50972-1660959841

Demo Link : https://drive.google.com/file/d/1MrtFBJM41dRfF5_3dka7rFW7nwawODpc/view?usp=sharing