Fertilizers Recommendation System for Disease Prediction

PROJECT REPORT

Submitted by

Team ID: PNT2022TMID34560

T. BENISHA (961819205008)

A. ATHINA (961819205007)

R. ASMITHA (961819205006)

L. MONICA BOSCO (961819205016)

In partial fulfilment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

PONJESLY COLLEGE OF ENGINEERING ,

NAGERCOIL.

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 bacterial and fungal diseases. Diseases on plants placed a major constraint on the production and a major threat to food security. Hence, early and accurate identification of plant diseases is essential to ensure high quantity and best quality. In recent years, the number of diseases on plants and the degree of harm caused has increased due to the variation in pathogen varieties, changes in cultivation methods, and inadequate plant protection techniques.

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

1.2 Purpose:

✓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.

2. LITERATURE SURVEY

2.1 Existing problem :

Indumathi proposed a method for leaf disease detection and suggest fertilizers to cure leaf diseases. But the method involves less number of train and test sets which results in poor accuracy. Pandi selvi proposed a simple prediction method for soil-based fertilizer recommendation system for predicted crop diseases. This method gives less accuracy and prediction. Shiva reddy proposed an IoT based system for leaf disease detection and fertilizer recommendation which is based on Machine Learning techniques yields less 80 percentage accuracies.

2.2 References :

- Fertilizers Recommendation System For Disease Prediction In Tree Leave | Semantic Scholar
- Soil Based Fertilizer Recommendation System for Crop Disease Prediction System (ijetajournal.org)
- Leaf Disease Detection and Fertilizer Suggestion | IEEE Conference Publication | IEEE Xplore
- IRJET-V7I1004.pdf

□A nutrient recommendation system for soil fertilization based on evolutionary computation - ScienceDirect

□Fertilizers-Recommendation-System-For-Disease-Prediction-In-Tree-Leave.pdf (ijstr.org)

□2204.11340.pdf (arxiv.org)

□371-376,Tesma405,IJEAST.pdf

□CROFED - Crop and Fertilizer Recommendation and Disease diagnosis system using Machine Learning and Internet of Things. (ijirt.org)

□Prediction of Crop, Fertilizer and Disease Detection for Precision Agriculture by IJRASET – Issuu.

2.3 Problem statement definition :

3. IDEATION AND PROPOSED SOLUTION

3.1 Empathy map canvas :

What do they think and feel ?

Improves quality and quantity. It can reduce a man power. It makes the farmer as smart as possible. Which fertilizer will cure the disease? It has accelerated the agriculture process. Improves productivity and effciency. Early response for the disease. It unlocks a new level of modern agriculture. It has accelerated the agriculture process. Improve productivity and effciency.

What do they hear ?

The cost for using this service is less. Easy and user-friendly. It reduces the complexity of disease prediction. It is far better than traditional analysis technique. This is more helpful in identifying the disease in the crop and make us more profit. Fix our problems from early stages with this application.

What do they see ?

Reducing pest and diseases. Promoting a healthy lifestyle for the farmer. People wih no prior knowledge can access. Multilingual application. Instant solution. Ugration of the industry with this application.

What do they say and do ?

Before applying a new technology farmer should fully understand the risk. It is such a good application for the farmers and it will make them profitable in their business. How can i trust a machine for my business? Can you guarantee the accuracy of this application? can this application responsibility for the losses that happen due to this application? I will try this and compare with actual outcome and predicted one?

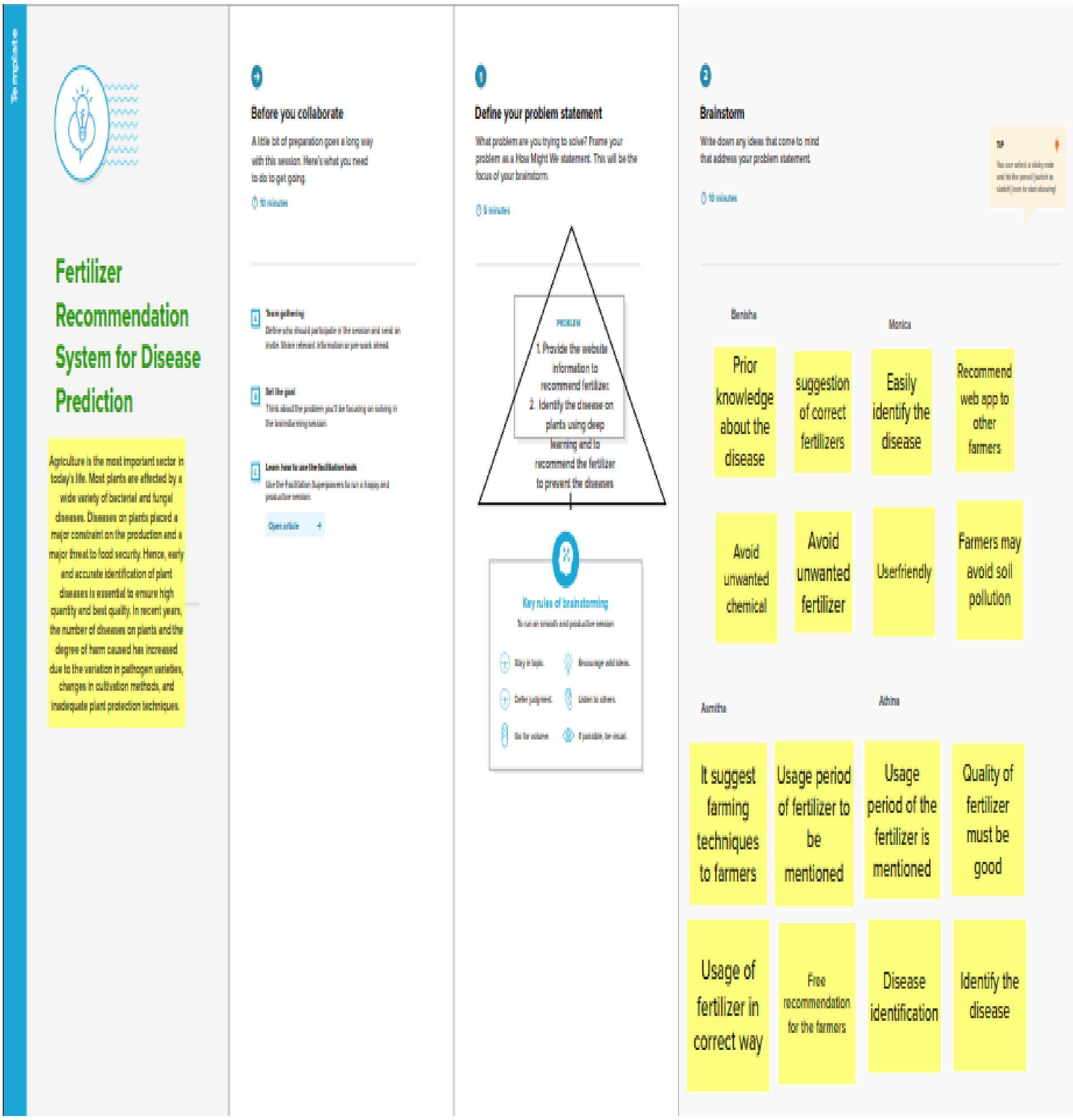
Pain

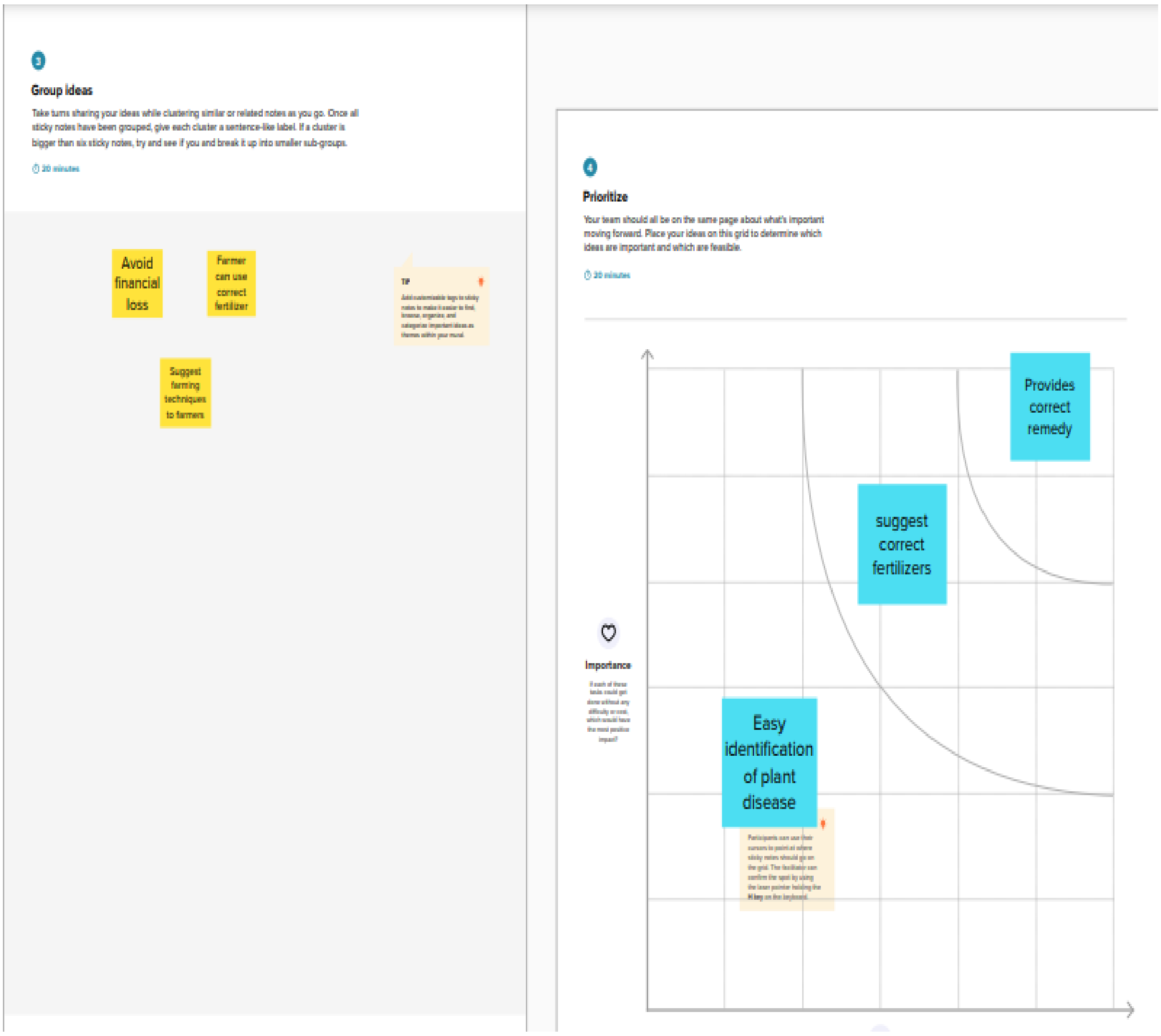
It may lead to the wrong prediction. Not accurate at all the time due to lack of data. Crop prediction accuracy, disease and correct fertilizer recommendations. A small error in the algorithm or data results in a large amount of loss. Is recommended fertilizer available in the user's location?

Gain

Early detection and management of problems. Self-working environment. Better utilization of available resources. One of the most efcient and rapid methods of disease prediction. Improves productivity.

3.2 Ideation and brainstorming





3.3 Proposed solution

So, we have built Web Application where:

1. Farmer interact with the portal build.
2. Interact with the user interface to upload images of diseased leaf.

3. Our model build analyses the disease and suggests the farmer with fertiizers are to be used.

➢ Detection and recognition of plant diseases using machine learning are very efficient in providing symptoms of identifying diseases at its earliest.

➢ It recommends the fertilizer for affected leaves based on severity level.

➢ This web application makes the farmers to take right decision in selecting the fertilizer for crop disease such that agricultural sector will be developed by innovative idea.

3.4 Problem solution fit

Farmer Are The First Customer For This Application. Farmer Can Easily Use This Application And Get Suggestion For Fertilizer To Used Correctly. Availability of good networks. Capturing the image in a required pixels to get a accurate prediction of disease in the plant. People are judge the disease in plants by Identifying through the change of leaf's quality This application focuses on helping for the farmer who needs a better recommendation of fertilizer on the infected plants . Identifying the disease is one of the biggest problem here. Various disease on the plants can lead to reducing the quality and quantity of the crops productivity. The insects on the plants can spread the disease . Seeing their crops are being infected by disease and facing huge loss in quantity and quality. Directly Farmer can easily identify the disease by the application and they don't need any extra knowledge on the disease prediction. Indirectly, Farmer can be able to get resul through online immediately. Before losing self-confidence,distress After gaining self-confidence relief Using the fertilizer is one the solution for the disease in the plants. Our Application use the image of the infected plant by identifying the disease and suggest the good fertilizer for the disease. Basic knowledge on the plant and fertilizer. People try to identify the disease by the quality of the leaf's.

4. REQUIREMENT ANALYSIS

4.1 Functional requirements

User registration - Registration through form Registration through Gmail.

User confirmation - Confirmation via OTP Confirmation via Email.

Capturing image - Capture the image of the leaf And check the parameter of the captured image .

Image processing - Upload the image for the prediction of the disease in the leaf.

Leaf identification - Identify the leaf and predict the disease in leaf.

Image description - Suggesting the best fertilizer for the disease .

4.2 Non-functional requirements

Usability - Datasets of all the leaf is used to detecting the disease that present in the leaf.

Security - The information belongs to the user and leaf are secured highly.

Reliability - The leaf quality is important for the predicting the disease in leaf.

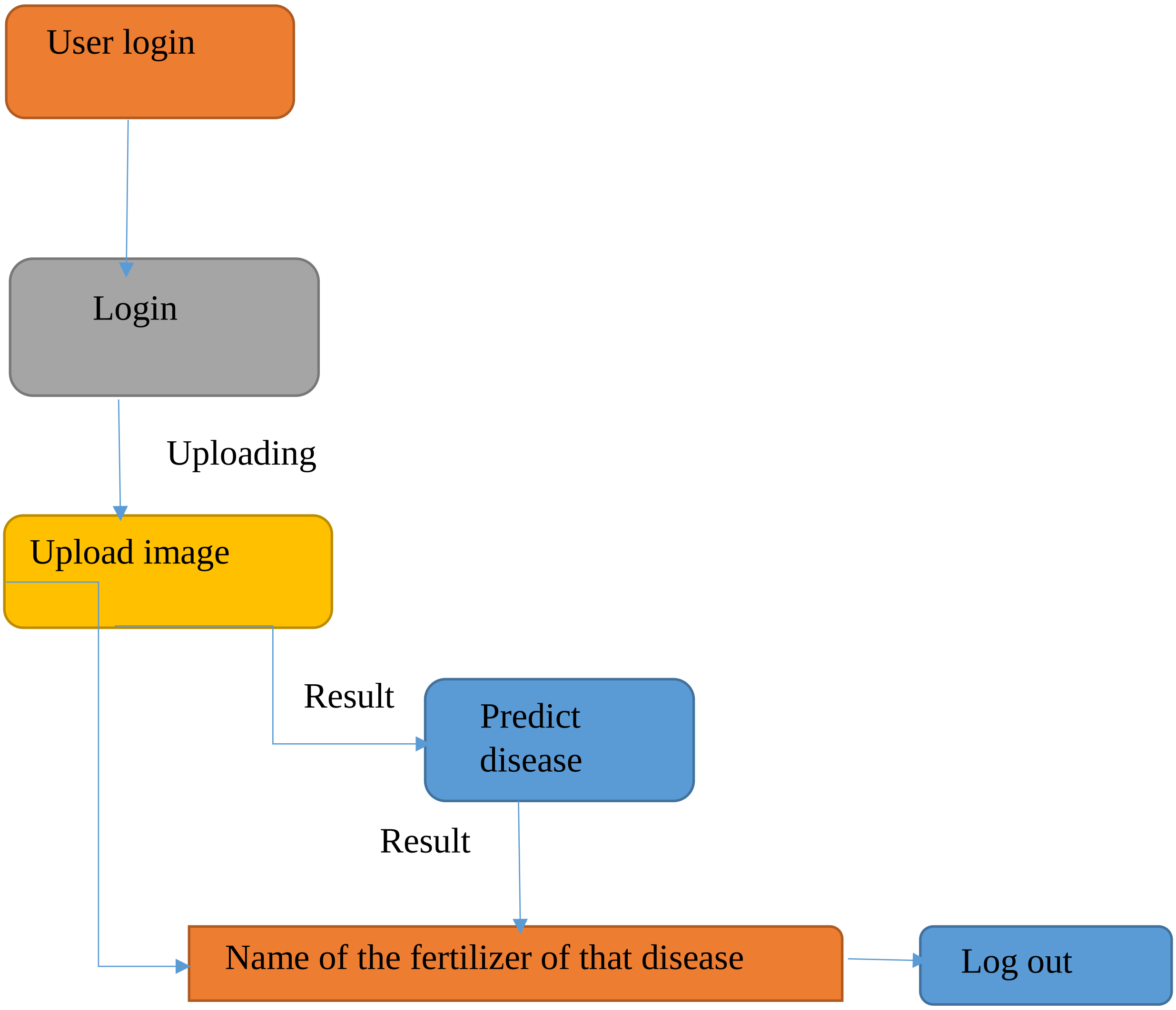
Performance - The performance is based on the quality of the leaf used for disease prediction.

Availability - It is available for all user to predict the disease in the plant.

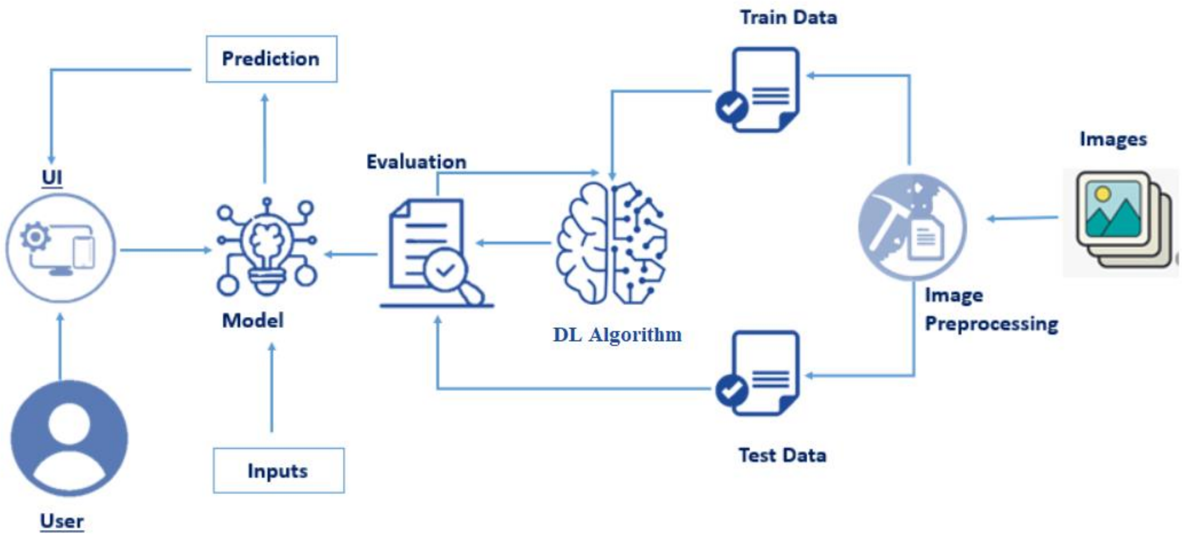
Scalability - Increasing the prediction of the disease in the leaf.

5. PROJECT DESIGN

5.1 Dataflow diagrams



5.2 Solution and technical architecture.



5.3 User stories

1. User Interface How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. HTML, CSS, JavaScript / Angular Js / React Js etc.

2. Application Logic-1 Logic for a process in the application Python

3. Application Logic-2 Logic for a process in the application IBM Watson STT service

4. Application Logic-3 Logic for a process in the application IBM Watson Assistant

5. Database Data Type, Configurations etc. MySQL, NoSQL, etc.

6. Cloud Database Database Service on Cloud IBM DB2, IBM Cloudant etc.
7. File Storage File storage requirements IBM Block Storage or Other Storage Service or Local Filesystem

8. External API-1 Purpose of External API used in the application IBM Weather API, etc.

9. External API-2 Purpose of External API used in the application Aadhar API, etc.

10. Machine Learning Model Purpose of Machine Learning Model Object Recognition Model, etc.

11. Infrastructure (Server / Cloud) Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration : Local, Cloud Foundry, Kubernetes, etc.

6. PROJECT PLANNNING AND SCHEDULING

6.1 Sprint planning and estimation

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**S P**
Sprint-1	Download dataset and Image Preprocessing.	USN-1	As a user, I can download dataset and I can retrieve useful information about the images.	
Sprint-2	Model Building for Fruit Disease Prediction.	USN-2	As a user, I can able to predict fruit disease using this model.	
Sprint-2	Model Building for Vegetable Disease Prediction.	USN-3	As a user, I can able to predict vegetable disease using this model.	
Sprint-3	Application Building.	USN-4	As a user, I can see a web page for Fertilizers Recommendation System for Disease Prediction	

Sprint-4	Train The Model on IBM Cloud.	USN-5	As a user, I can save the information about Fertilizers and crops on IBM cloud	

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Pe Comple on Plannec Date)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20

7. CODING AND SOLUTIONING

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"></scri
pt>
    <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"></scr
ipt>
<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;
}


</style>
</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>
```

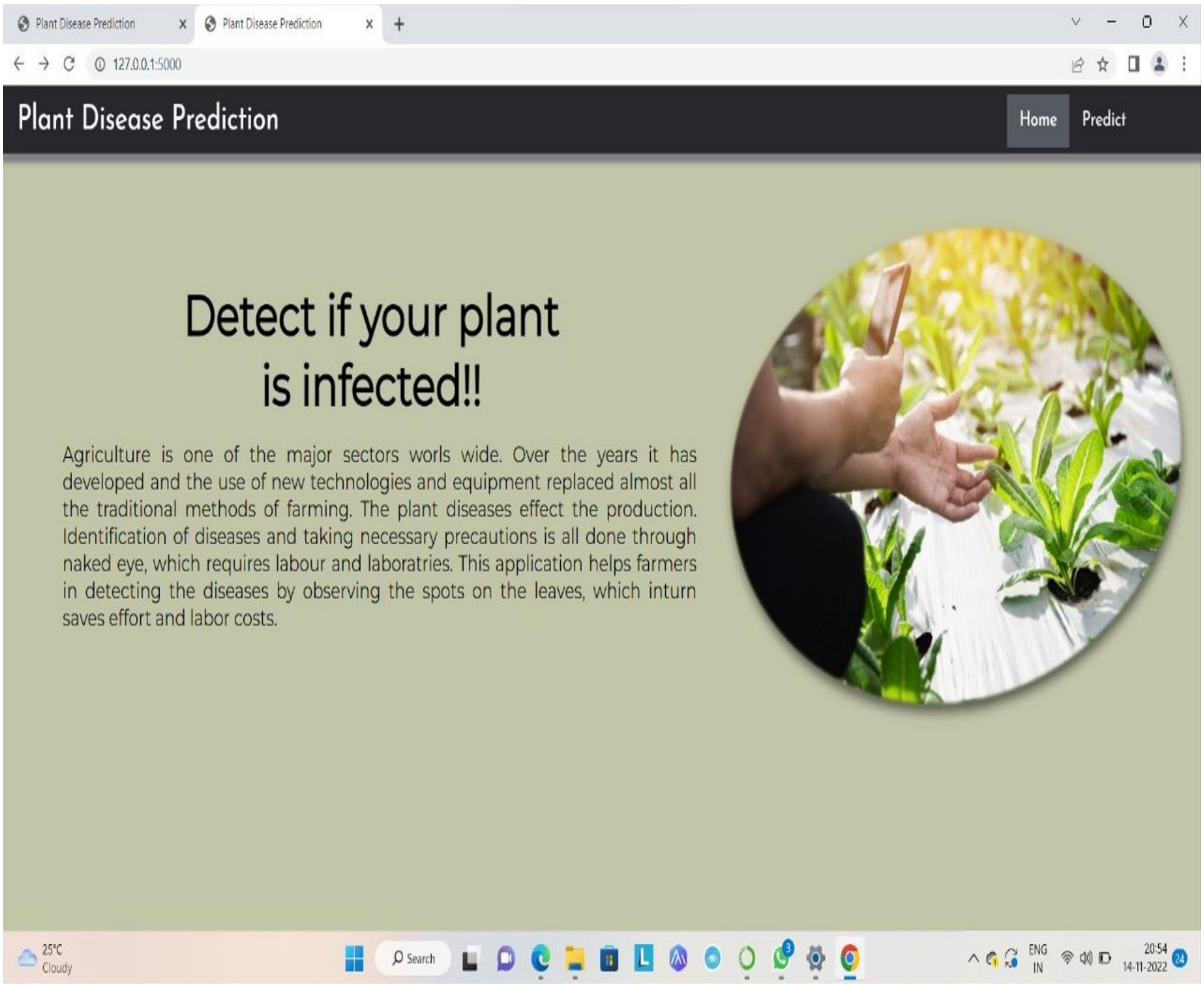
```
                             <div>
                                    <button type="button" class="btn
btn-info btn-lg " id="btn-predict" style="background:
#28272c;">Predict!</button>
                             </div>
                      </div>

                      <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>
```

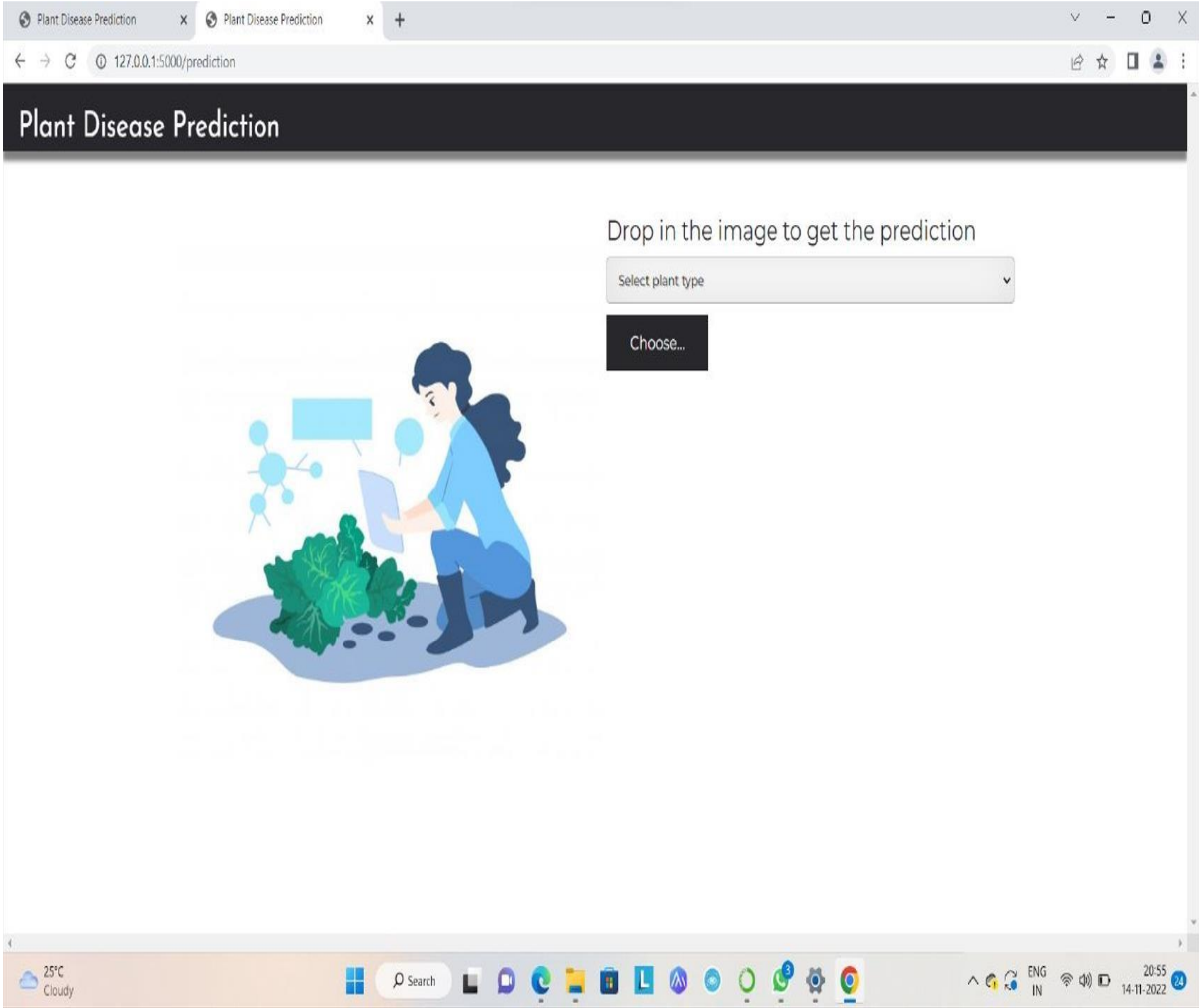
APP.PY

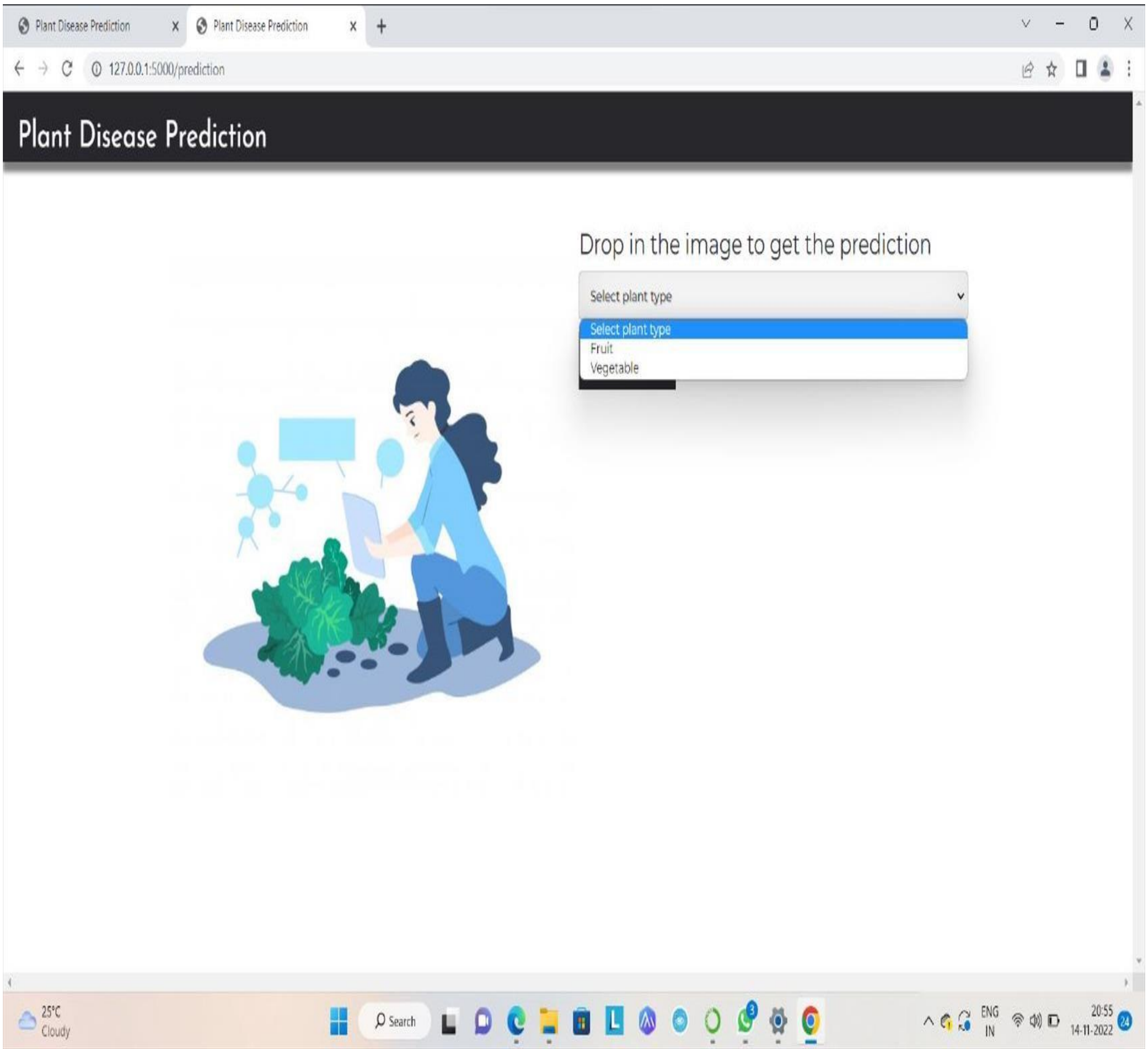
```
import requests
from tensorflow.keras.preprocessing import image
from tensorflow.keras.models import load_model
import numpy as np
import pandas as pd
import tensorflow as tf
from flask import Flask, request, render_template, redirect, url_for
import os
from werkzeug.utils import secure_filename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

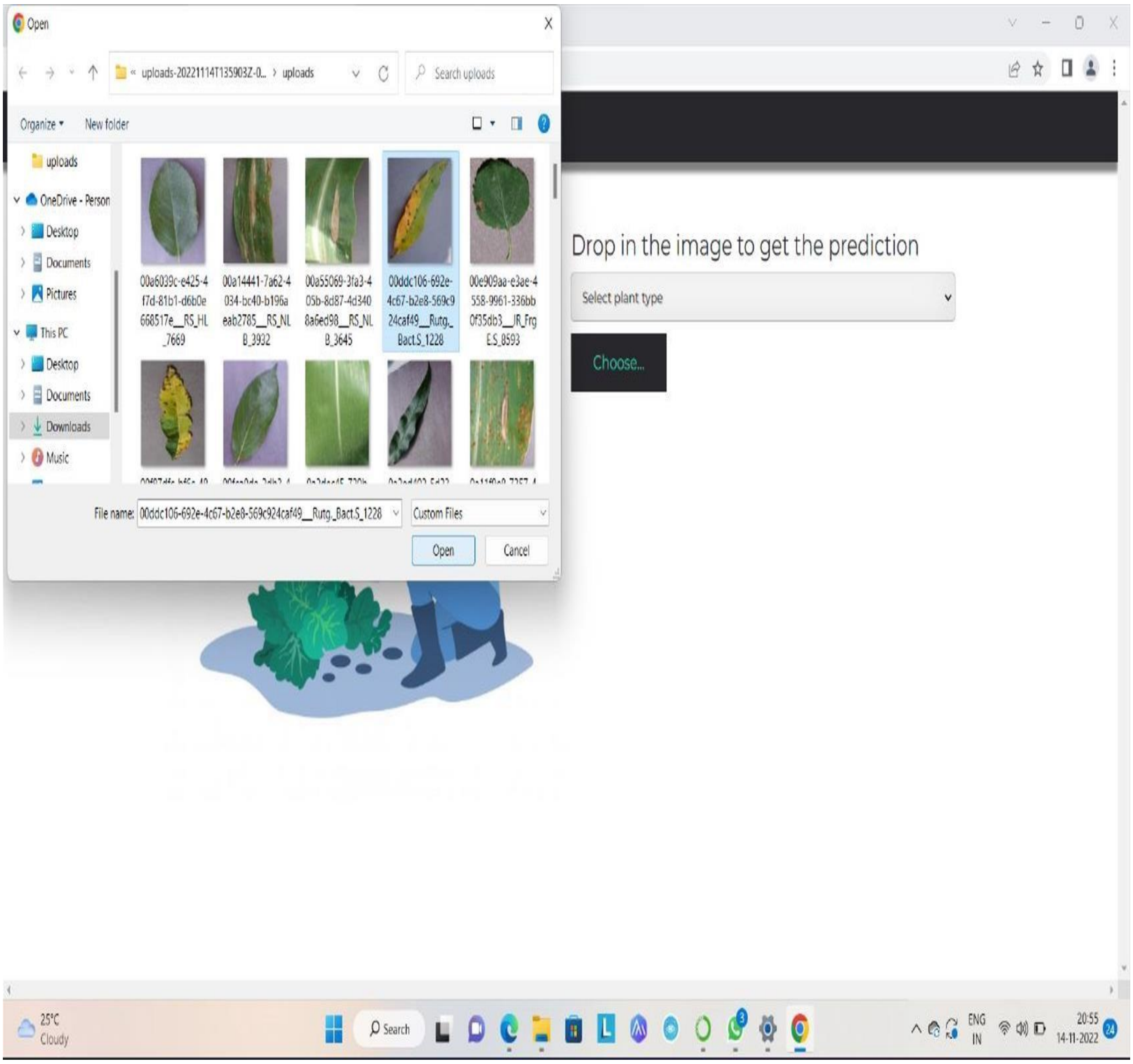
```
@app.route('/predict',methods=['POST'])
def predict():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['image']

        # 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)
s
```

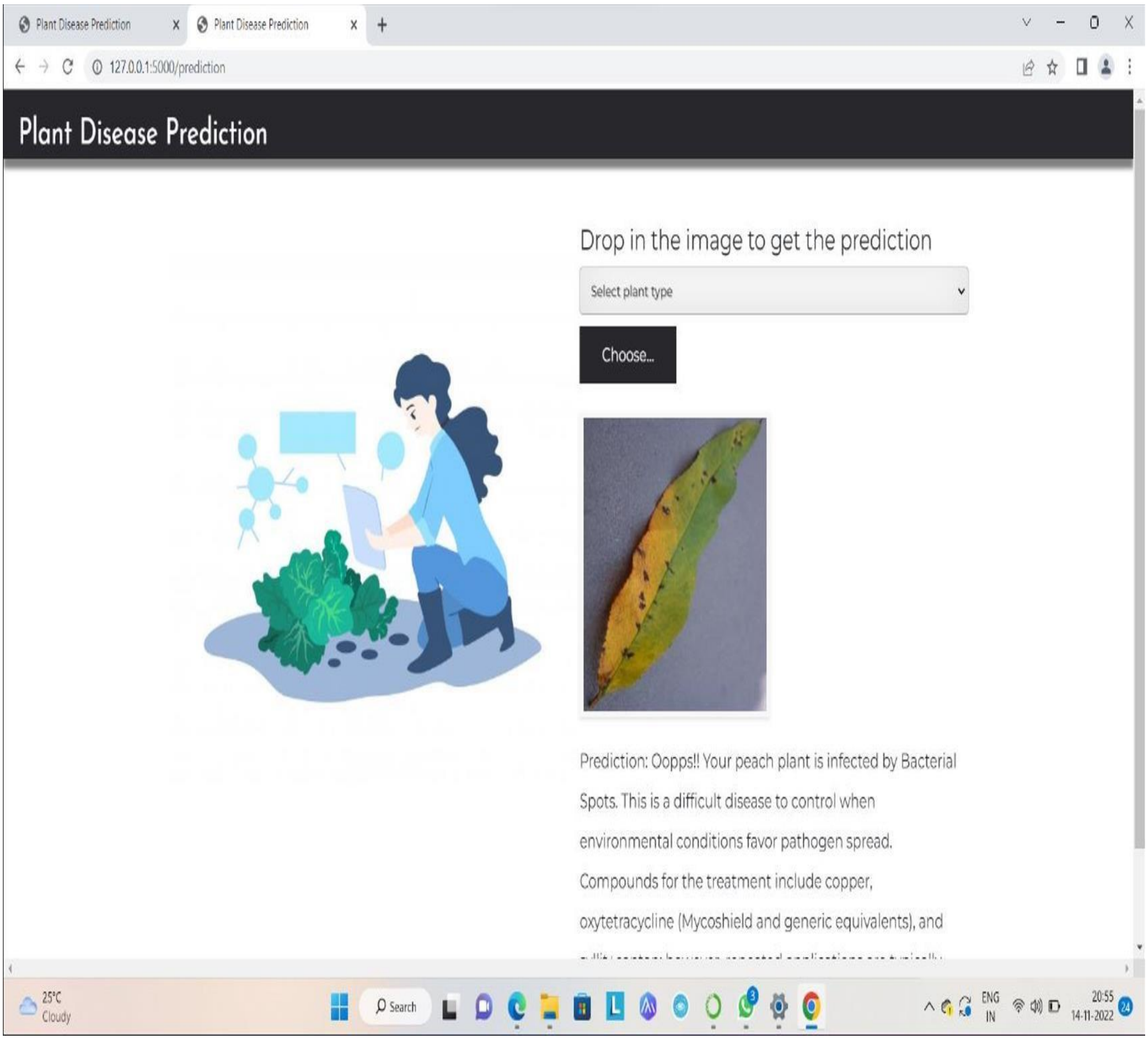
SOLUTION


Plant Disease Prediction
Plant Disease Prediction
127.0.0.1:5000
Plant Disease Prediction
Home
Predict
Detect if your plant
is infected!!
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.
25°C
Cloudy
Search
ENG
IN
20:54
14-11-2022


Plant Disease Prediction
Plant Disease Prediction
127.0.0.1:5000/prediction
Plant Disease Prediction
Drop in the image to get the prediction
Select plant type
Choose...
25°C
Cloudy
Search
ENG
IN
20:55
14-11-2022


Plant Disease Prediction
Plant Disease Prediction
127.0.0.1:5000/prediction
Plant Disease Prediction
Drop in the image to get the prediction
Select plant type
Select plant type
Fruit
Vegetable
25°C
Cloudy
Search
ENG
IN
20:55
14-11-2022


Open
uploads-20221114T135903Z-0... › uploads
Search uploads
Organize
New folder
uploads
OneDrive - Person
Desktop
Documents
Pictures
This PC
Desktop
Documents
Downloads
Music
00a6039c-e425-4f7d-81b1-d6b0e668517e___RS_HL_7669
00a14441-7a62-4034-bc40-b196aeab2785___RS_NLB_3932
00a55069-3fa3-405b-8d87-4d3408a6ed98___RS_NLB_3645
00ddc106-692e-4c67-b2e8-569c924caf49___Rutg._Bact.S_1228
00e909aa-e3ae-4558-9961-336bb0f35db3___JR_FrgE.S_8593
File name: 00ddc106-692e-4c67-b2e8-569c924caf49___Rutg._Bact.S_1228
Custom Files
Open
Cancel
Drop in the image to get the prediction
Select plant type
Choose...
25°C
Cloudy
Search
ENG
IN
20:55
14-11-2022


Plant Disease Prediction
Plant Disease Prediction
127.0.0.1:5000/prediction
Plant Disease Prediction
Drop in the image to get the prediction
Select plant type
Choose...
Prediction: Oopps!! Your peach plant is infected by Bacterial
Spots. This is a difficult disease to control when
environmental conditions favor pathogen spread.
Compounds for the treatment include copper,
oxytetracycline (Mycoshield and generic equivalents), and
25°C
Cloudy
Search
ENG
IN
20:55
14-11-2022

8. ADVANTAGES AND DISADVANTAGES

8.1 Advantages

• The proposed model could predict the disease just from the image of a part icular plant.

• Easy to use UI.

• Model has some good accuracy in detecting the plant just by taking the input(leaf).

8.2 Disadvantages

● For training and testing, the proposed model requires very high computational time.

● The neural network architecture used in this project work has high complexity.

9.CONCLUSION

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

● The accuracy of classification increased by increasing the number of epochs.

● For different batch sizes, different classification accuracies are obtained.

● The accuracies are increased by increasing more convolution layers.

● The accuracy of classification also increased by varying dense layers.

● Different accuracies are obtained by varying the size of kernel used in the convolution layer output.

● Accuracies are different while varying the size of the train and test datasets.

10. FUTURE SCOPE

The proposed model in this project work can be extended to image recognition. The entire model can be converted to application software using python to exe software. The real time image classification, image recognition and video processing are possible with help OpenCV python library. This project work can be extended for security applications such as figure print recognition, iris recognition and face recognition.

11. APPENDIX

ABOUT THE SOFTWARE

FEATURES OF HTML

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.

• Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.

• As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display. Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

FEAURES OF JUPYTER

The Jupyter Notebook is an open-source web application; that enables data scientists to create and share documents that include live code, equations, computational output, visualizations, multimedia resources, and explanatory text. Using only 'shift + enter' keys, your code in Jupyter Notebook gets executed, which helps to identify whether the code works or not. Because of this, it has become one of the day-to-day tools used by data scientists as it is easy to explore and plot data.

Alright, now you may be wondering why hype so much for a tool for documentation and coding purposes, because Jupyter Notebook handles not 1 but 40 programming languages, including Scala, R, Python, and Julia.

12.PROJECT DEMO LINK

https://youtu.be/ar0Nvsin_lA