PROJECT PROPOSAL

1.1 Abstract

In recent years, the agricultural sector has seen a transformative integration of Artificial Intelligence (AI) technologies. This integration has paved the way for advanced predictive systems designed to mitigate the impact of diseases on crops. The "Agriculture Disease Prediction System Enhanced by Artificial Intelligence" represents a cutting-edge approach to disease management in agriculture. This system leverages AI-powered algorithms, including machine learning and computer vision, to analyse various data sources. By processing this diverse dataset, the system can accurately predict and identify potential disease outbreaks in crops, allowing farmers to take timely preventive measures. The existing agricultural disease prediction system relies on manual observation and periodic field visits by experts to identify crop diseases. The system is inefficient, not automated, and often results in delayed disease detection and response, leading to crop losses and increased pesticide use. Additionally, data is often stored in separate silos, making integration and analysis challenging. This abstract provides a glimpse into the potential of AI-powered agriculture disease prediction systems, highlighting their significance in modern agriculture and their potential to revolutionize farming practices for a more sustainable and productive future.

1.2 Existing System

- **Manual Observation**: Farmers and agricultural experts rely on their experience and knowledge to identify signs and symptoms of diseases in crops. This process is often time-consuming and subjective, leading to potential misdiagnosis.
- **Field Visits:** Agricultural experts may conduct field visits to assess the health of crops and identify disease outbreaks. These visits are periodic and may not always coincide with the onset of diseases.

1.3 Problem Statement

- **Manual and Subjective:** Disease detection relies heavily on manual observation, which can be time-consuming, subjective, and prone to human error. This approach hampers the early identification of diseases.
- **Limited Data Sources:** The system often lacks access to diverse and critical data sources, such as real-time satellite imagery, weather conditions, soil analysis, and historical disease data. This limitation hinders comprehensive disease prediction.

1.4 Proposed System

- Leveraging machine learning algorithms for swift and automated disease identification. Faster response times and reduced dependence on manual observation.
- Incorporating data from diverse sources, including satellite imagery and soil analysis. Comprehensive datasets for accurate and timely predictions.
- Intuitive interfaces for farmers and experts to interact seamlessly with the prediction system. Improved usability and accessibility for stakeholders.

1.5 Software and hardware Requirements

Software Requirements

- **Operating System:** Windows 10, Windows 11.
- **Web Browser:** Chrome, Firefox, Safari, or Edge.
- **Security:** Antivirus or security software.

Hardware Requirements

- **Processor (CPU)**: Dual-core or higher (e.g., Intel Core i3/i5, AMD Ryzen 3/5).
- **RAM:** 8 GB or more.
- **Storage:** 256 GB SSD or higher.
- **Graphics:** Integrated graphics are usually sufficient for basic tasks.
- **Display:** Standard monitor or laptop screen.

1.6 List of Modules

This the Agriculture Disease Prediction System Enhanced by AI contains five main modules they are,

- Upload file.
- Predict.
- Tips.
- About.
    - Tips reference.
    - Dataset reference.
- Admin.
    - Update Dataset

Upload file

- Get the disease plant image and process the disease image.

Predict

- It is processing the image to dataset and give the disease name.

Tips

- Give the tips to the user based on the disease name.

Admin

- This module is used for admin to update the dataset for adding new disease.

About

- This module is used for user to know about the website.

1.7 Summary

The chapter describes Project Proposal Document (Abstract, Existing System, Problem Statement, Proposed System, Software and hardware Requirements, List of Modules) and in next chapter describe Feasibility .

FEASIBILITY STUDY

2.1 Introduction

An important outcome of the preliminary investigation is the confirmation that the system requested is feasible. The Agriculture Disease Prediction System Enhanced by AI is wed application used for disease predict . The existing system process is more workload and increase the work of farmers.

2.2 Feasibility Analysis

An important outcome of the preliminary investigation is the affirmation that system requested is feasible. The three types of feasibility studies which helped me to identity the solutions are stated below:

- Technical Feasibility
- Economical Feasibility
- Operational Feasibility

Technical Feasibility

To identity whether it is technical feasible, technology platform, supporting software and tools ad work area have to be given the first priority. Further, this system requires the duration of three months to complete.

The objectives of new tool are to maximum scalability and security of the business application.

And the feasibility study resulted in ‘yes’ for the following question:

- Does the necessary technology exist to do what is suggested?
- Does the proposed equipment have the technical capacity to hold the data required to use the new tool?
- Can the system be upgraded if developed?

The result ‘yes’ to the above questions and the proposed system requires the duration of six months to complete.

Economic Feasibility

The website owners must buy the software and hardware which are required; no initial cost regarding that is incurred.

Operational Feasibility

The system is developed using the front-end that is used that case coding, as it is more a web application rather than giving much concentration on network and their implementations. Backend requires having capability to store Data.

Thus, economically, and technically the new system tool is feasible. The operational feasibility is to take into consideration of the user's acceptance of the system. Any new system, especially a monitoring system, can result in users going against it since they may feel that it intruders into their privacy. However, properly specifying the merits of system can help overcome the initial rejection.

2.3 Vision Document

Problem Statement

The problem of	Farmers didn't Know disease for their plant.
Affects	Farmer's
The impact of which	In this process, better growth in crops.
A successful solution would be	Farmers enhance the productivity.

Problem Positioning Statement

Name	Represent	Roles
User	Farmers	User helped to know about the application they are getting into.
Admin	To manage the disease prediction Application.	Making ensure the everything is fine or Issues.

2.4 Summary

This chapter concludes with the summary and objectives of the project. The next chapter describes the requirements analysis of the project.

SYSTEM REQUIREMENTS SPECIFICATION (SRS)

3.1 Introduction

This chapter describes the modules that are implemented in the software with UML diagrams and use case designs.

3.2 Modules Description

It is a machine learning technology, which is used to disease Prediction System.

Modules

This the Soil Prediction and Plants Recommendations contains five main modules they are,

- Upload file.
- Predict.
- Tips.
- About.
    - Tips reference.
    - Dataset reference.
- Admin.
    - Update Dataset

3.3 Module Specification:

Upload

- **Purpose:**
  For uploading the image.
- **Responsible person:**
  Users, Admin.
- **Entry Criteria:**
  By clicking the upload button.
- **Input:**
  Image file.
- **Process:**
  Compares the image and trained dataset from database.

- **Output:**

  If the image is like dataset, it shows the result, otherwise shows not found message.
- **Exit Criteria:**

  You can exit by clicking back button.

Tips

- **Purpose:**

  Show recovery tips from the disease.
- **Responsible person:**

  User, Admin.
- **Entry Criteria:**

  By clicking the tips button.
- **Input:**

  Disease name.
- **Process:**

  Compares the disease name from the dataset.
- **Output:**

  If the disease name is like the dataset, it shows the tips , otherwise shows not found message.
- **Exit Criteria:**

  You can exit by clicking back button.

About Us

- **Purpose:**

  For Displaying the Details of the application.
- **Responsible person:**

  User, Admin.
- **Entry Criteria:**

  By clicking the About us button.
- **Input:**

  Feedback.

- **Process:**

  Send feedback to the admin.
- **Output:**

  It displays the details about the application.
- **Exit Criteria:**

  You can exit by clicking back button.

Admin

- **Purpose:**

  To access all control.
- **Responsible person:**

  Admin.
- **Entry Criteria:**

  By clicking the admin button.
- **Input:**

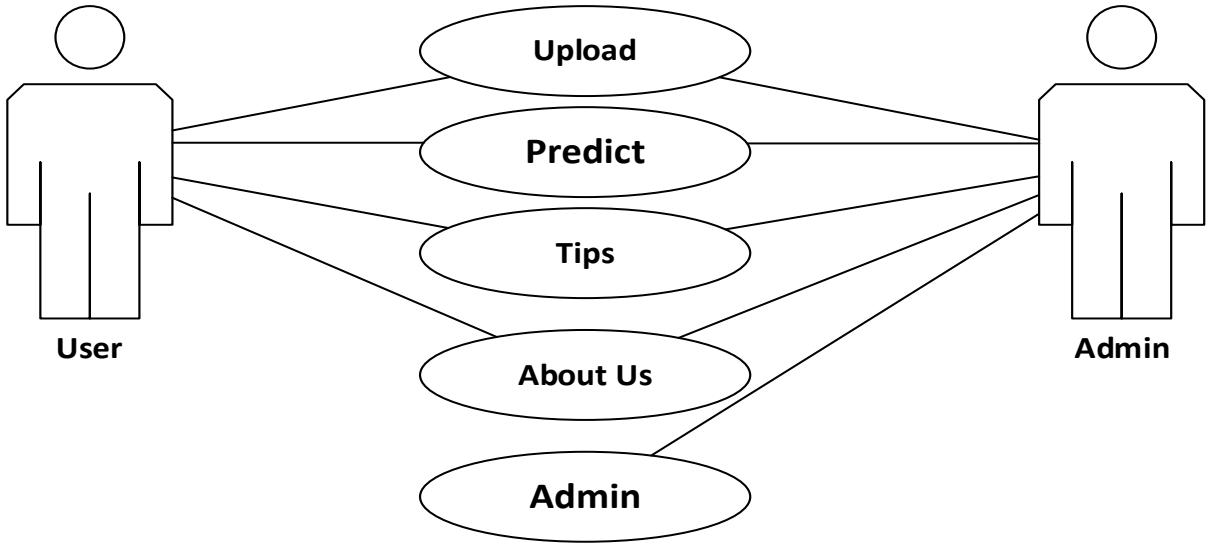
  Username, Password.
- **Process:**

  To add the new disease to the dataset.
- **Output:**

  To update the dataset.
- **Exit Criteria:**

  You can exit by clicking back button

3.4 UML Use Case



USE CASE	DESCRIPTION
Upload	The user and admin can upload the plant disease image.
Predict	The user and admin can predict the disease name.
Tips	The user and admin can view the tips based on the disease name.
About	The user can know about the application.
Admin	The admin can update the dataset.

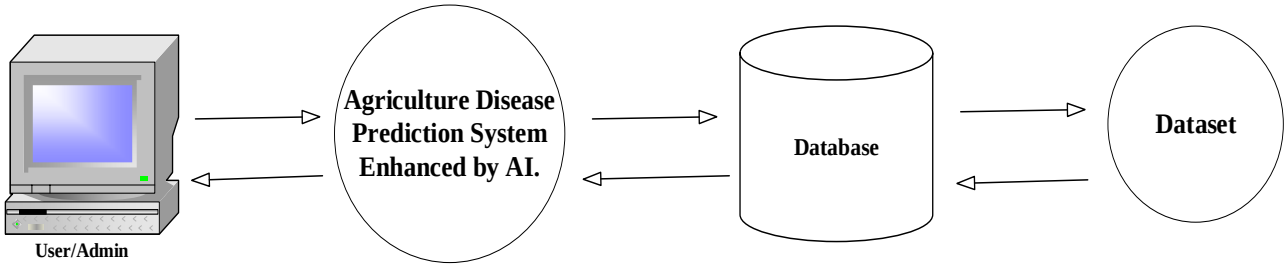
3.5 Summary

This session is described about System Analysis and design introduces the module description, and use case diagram

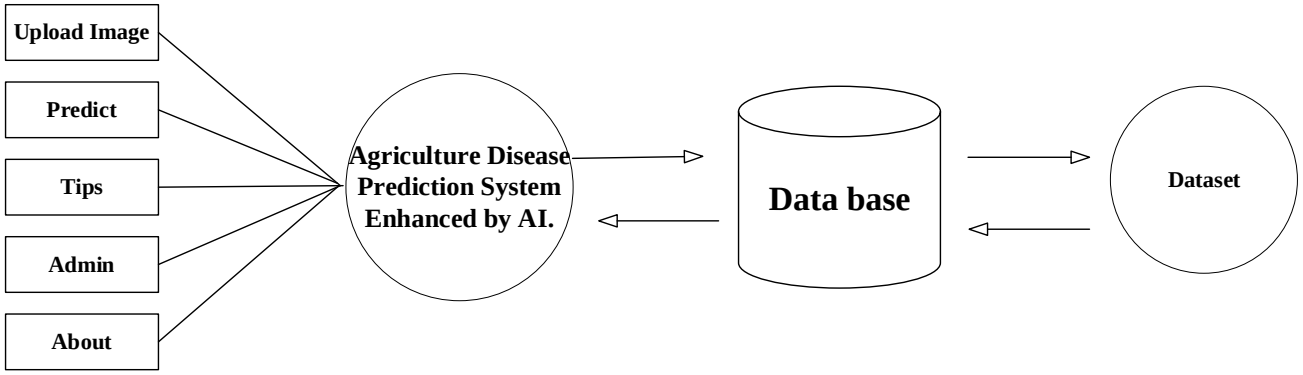
SOFTWARE ANALYSIS AND DESIGN

4.1 Data Flow Diagram

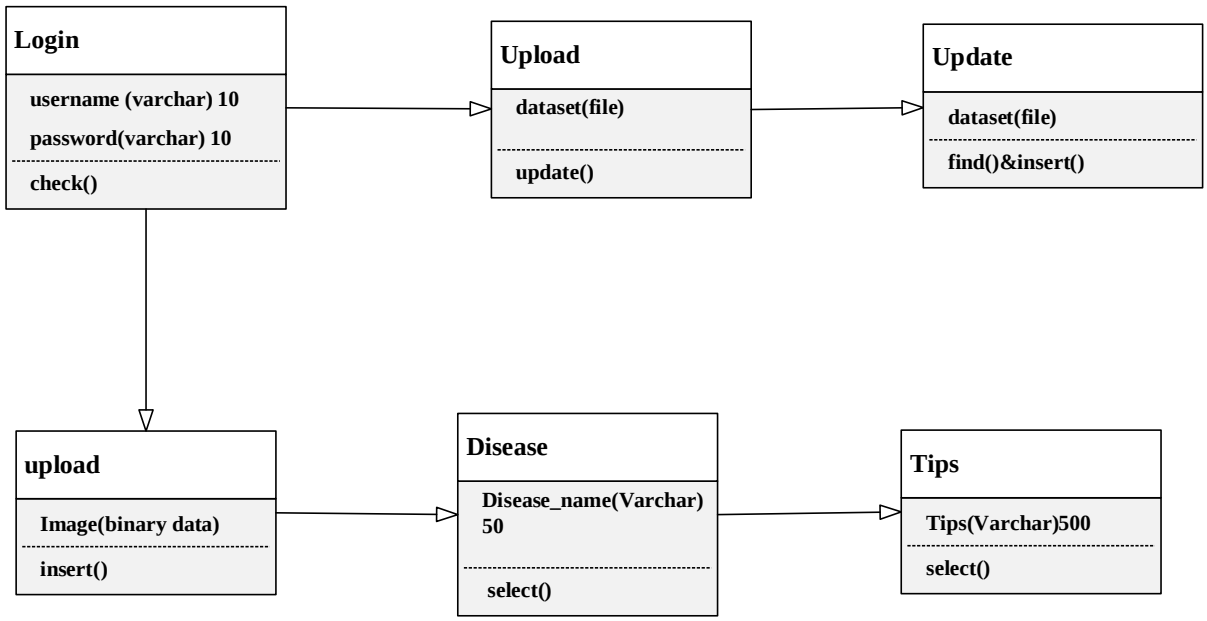
Level 0



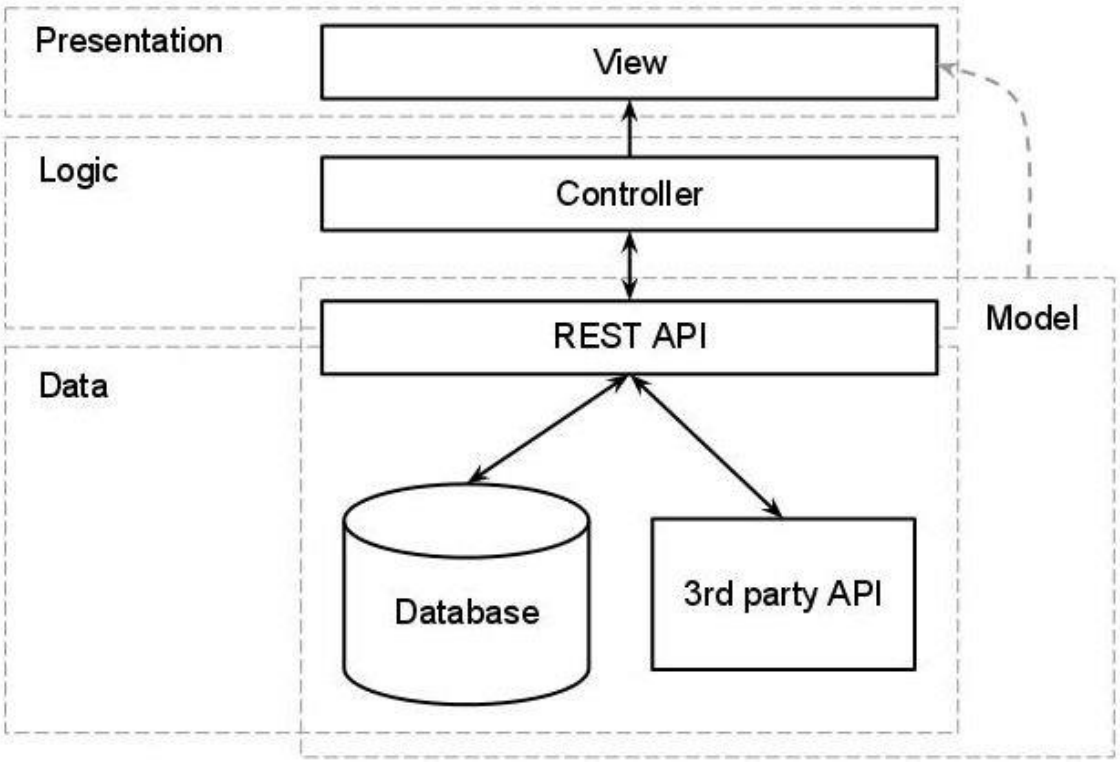
Level 1



4.2 Class Diagram



4.3 Architecture Diagram



4.4 Table Design

Admin

Column Name	Data type	Length	Key
Username	Varchar	10	Primary key/foreign key
Password	Varchar	8	-

Upload

Column Name	Data type	Length	Key
Dataset	Blob	50	-

Update

Column Name	Data type	Length	Key
Dataset	Blob	50	-

Upload

Column Name	Data type	Length	Key
Image	Blob	50	-

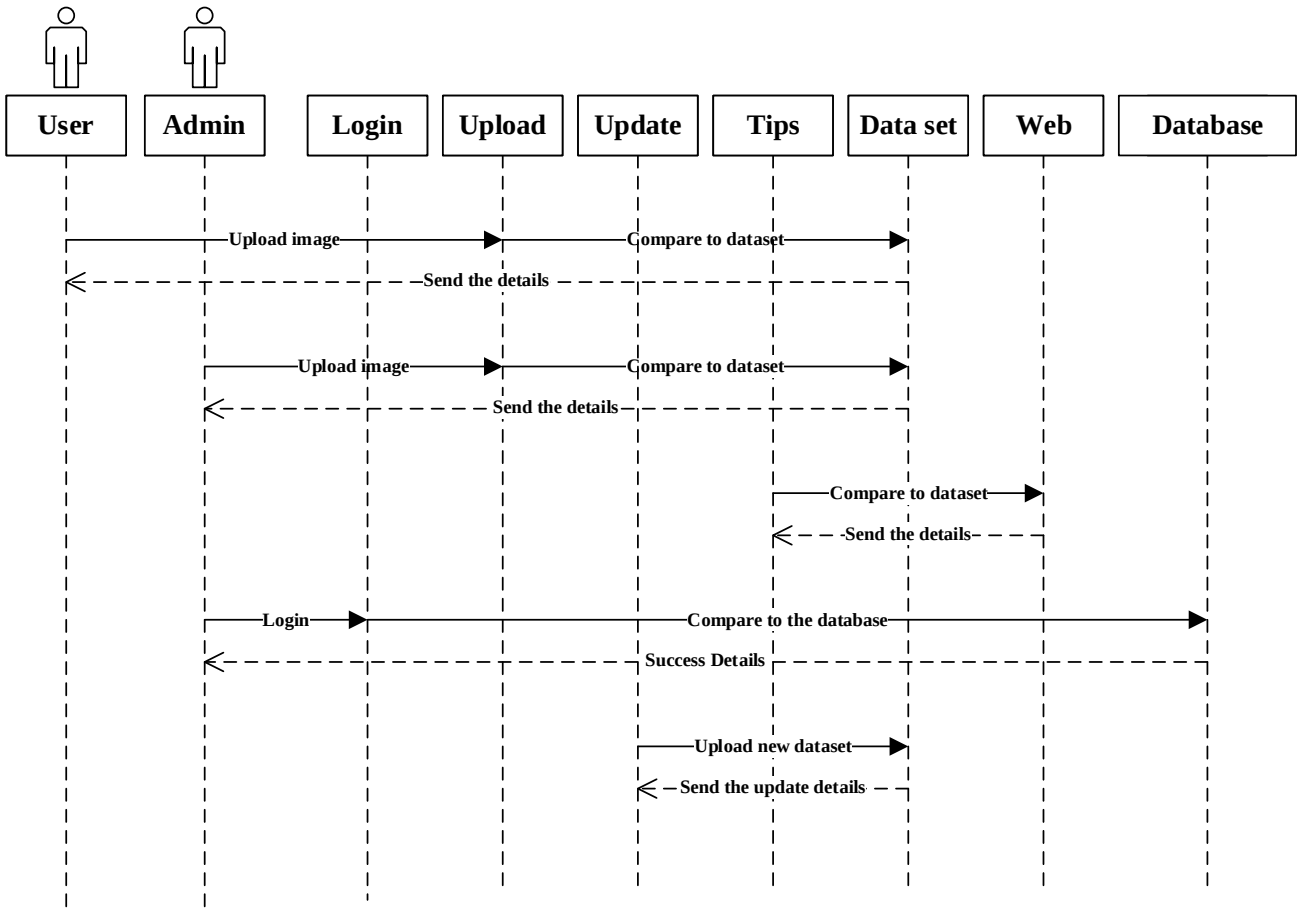
Disease

Column Name	Data type	Length	Key
Disease Name	Varchar	50	-

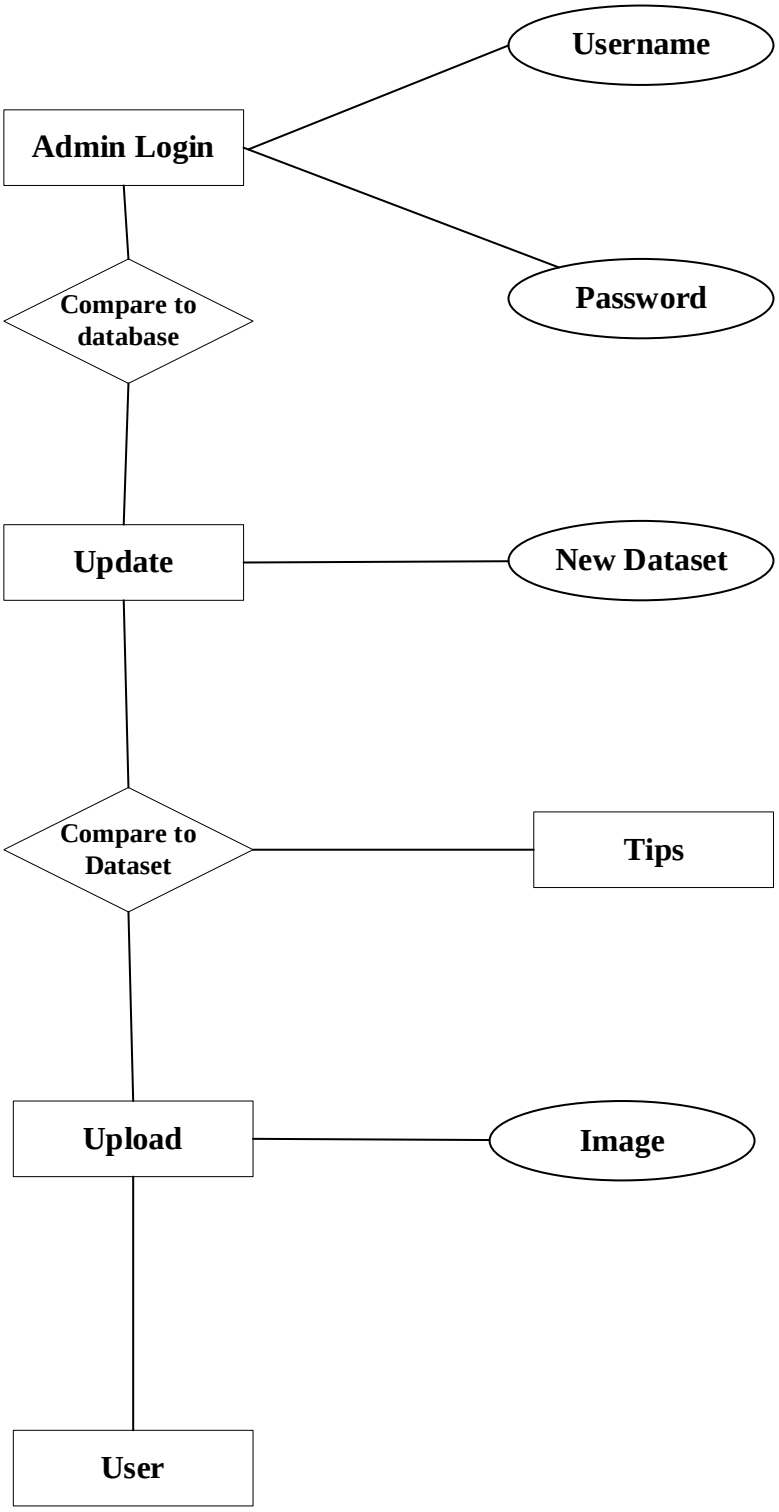
Tips

Column Name	Data type	Length	Key
Tips	Varchar	500	-

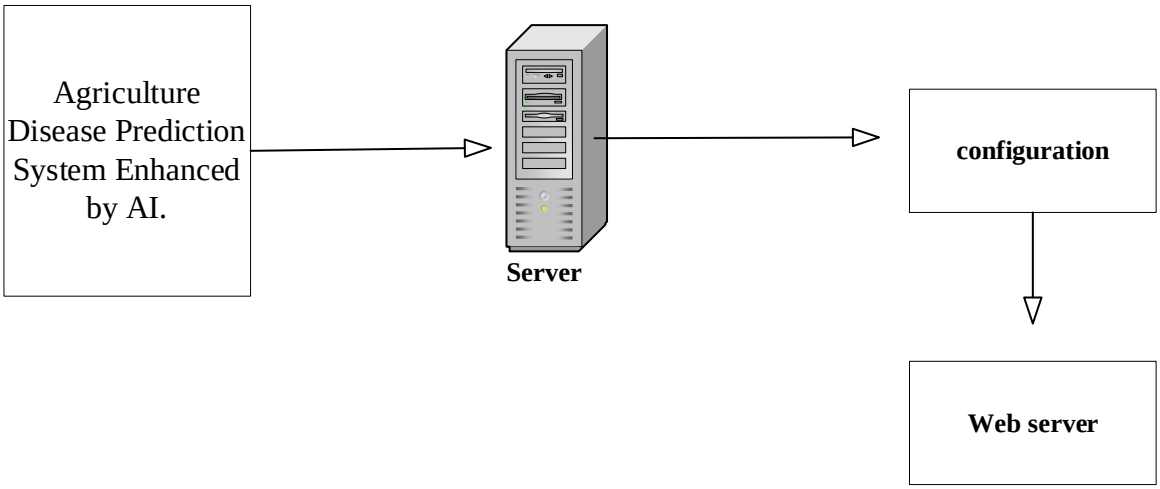
4.5 Sequence Diagram



4.6 E-R Diagram



4.7 Deployment Diagram



4.8 Summary

This session is described about software analysis and design, data flow diagram, class diagram, activity diagram, E-R diagram, and table design.

TEST CASE DESIGN

5.1 Introduction

Testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. It includes preventing bugs, reducing deployment costs and improving performance.

5.2 Types of testing

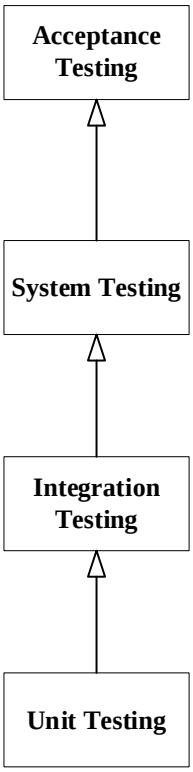
Manual Testing

Manual testing is a type of software testing in which test cases are executed manually by a tester without using any automated tools. The purpose of Manual Testing is to identify the bugs, issues, and defects in the software application.

Automatic Testing

Automated testing is a technique where software tests are executed automatically by a software tool or script, which compares the actual test results with the expected results.

5.3 Levels of Testing



Unit Testing

Unit testing is a type of software testing that focuses on individual units or components of a software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements.

Test No	**Module Name**	**Test Case**	**Expected Input**	**Expected Output**	**Test status**
1	Login	Checks whether the username and password correct or not.	Username	The username must contain only alphabets and digits.	Successful
			Password	The password should contain at least one special character, digits, uppercase letter, and a lowercase letter.	Successful
			Username and Password	If username and password match with the data stored in the DB. Or If the Username and password doesn't match.	Successful Unsuccessful

2	Upload	Allow the user to upload the image	Disease affected image.	If the image is valid, it displays the disease name to the upload image. Or If the image is invalid it alerts not found.	Successful Unsuccessful
3	Tips	Allow the user to find the disease related tips	Disease name	If the disease name related to the dataset Or If the disease name doesn't find the dataset it alerts not found.	Successful Unsuccessful

Integration Testing

Integration testing is a software testing technique that tests the interaction between different software modules or components. It is conducted after unit testing and before system testing.

Test No	**Module Name**	**Test Case**	**Expected Input**	**Expected Output**	**Test status**
1	Login & Upload	Checks whether the email and password and get the image from user	Username Password and image	If the username and password is in database, it receives and compare image to dataset. Or If the username and password is not in the database, it doesn't receive and image not in the dataset alert it.	Successful Unsuccessful

System Testing

System testing is a type of software testing that evaluates the overall functionality and performance of a complete and fully integrated software solution. It is conducted after integration testing and before acceptance testing. The purpose of system testing is to ensure that the software meets the customer's requirements and specifications, and to identify any defects or issues that may arise when the software is deployed to the end users.

Acceptance Testing

Acceptance testing is a type of software testing that is performed to determine whether the software meets the customer's requirements and specifications. It is conducted as a formal testing process based on user requirements and function processing, and it determines whether the software is conforming to specified requirements and user requirements or not .

5.4 Summary

The chapter describes Test design case and in next chapter describe Conclusion.

CONCLUSION

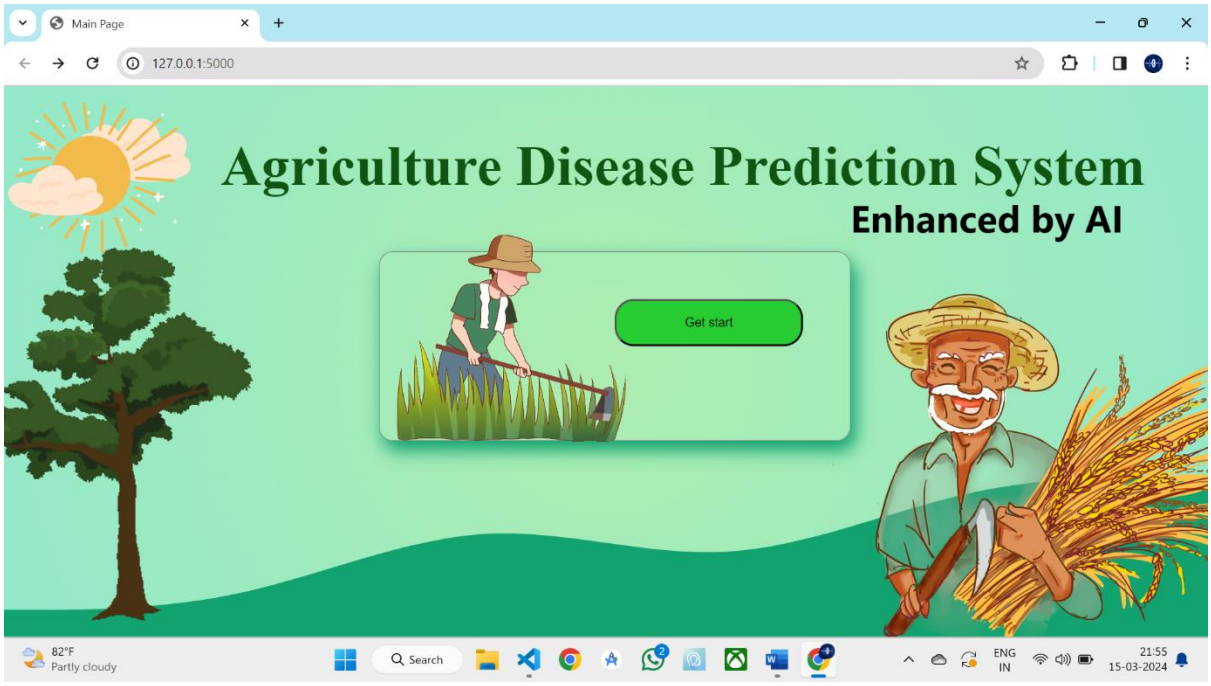
The enhanced Agriculture Disease Prediction System utilizes machine learning algorithms to accurately predict crop diseases, thereby assisting farmers in timely disease management and crop protection. By integrating data on environmental factors, crop varieties, the system provides proactive recommendations to farmers, reducing crop losses and promoting sustainable agricultural practices. Through rigorous testing and validation, the project demonstrates its efficacy in improving crop health monitoring and empowering farmers with actionable insights for informed decision-making, ultimately contributing to enhanced crop productivity and food security.

REFERENCES

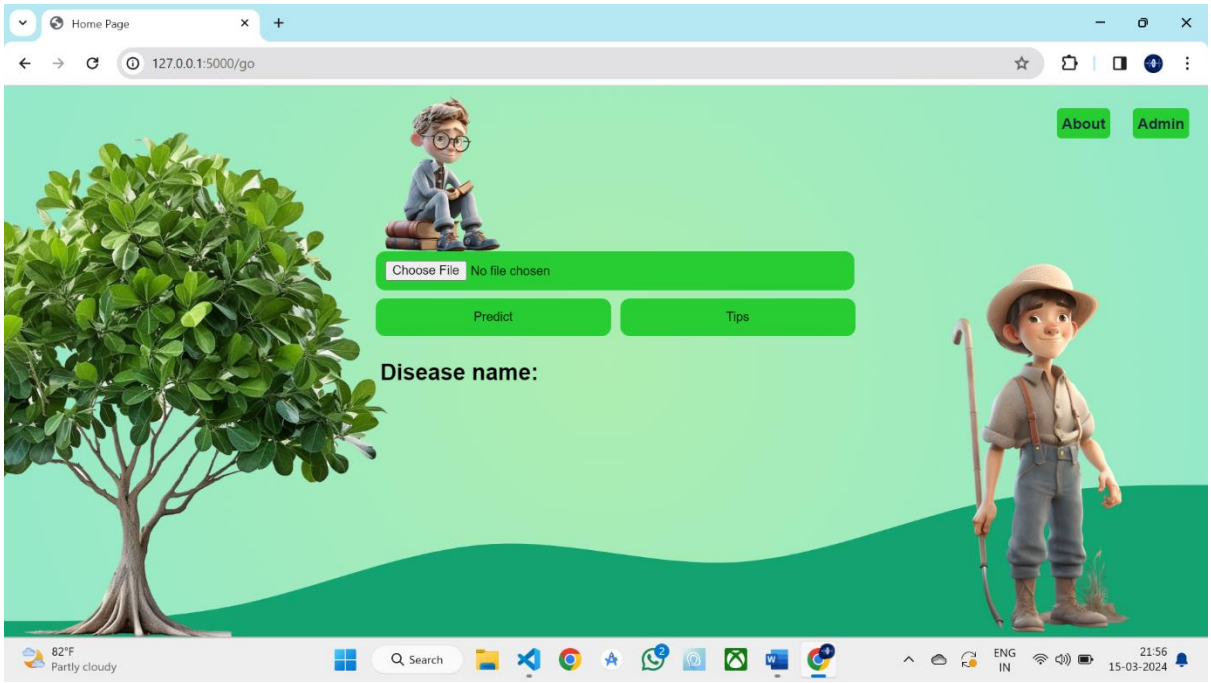
- https://www.educative.io/answers/what-is-morphological-image-processing
- https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset
- https://www.w3schools.com/python/
- https://stackoverflow.blog/2021/07/14/getting-started-with-python/
- https://www.w3schools.com/python/python_ml_getting_started.asp

8.1 Screenshots

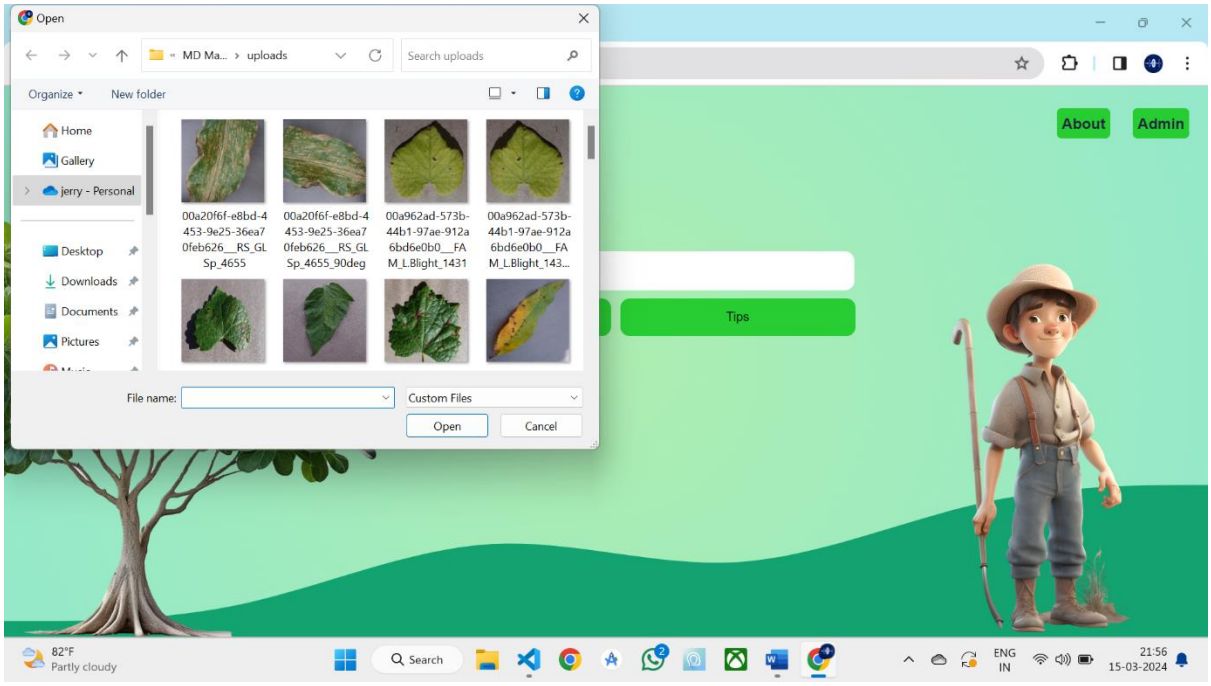
Index Page



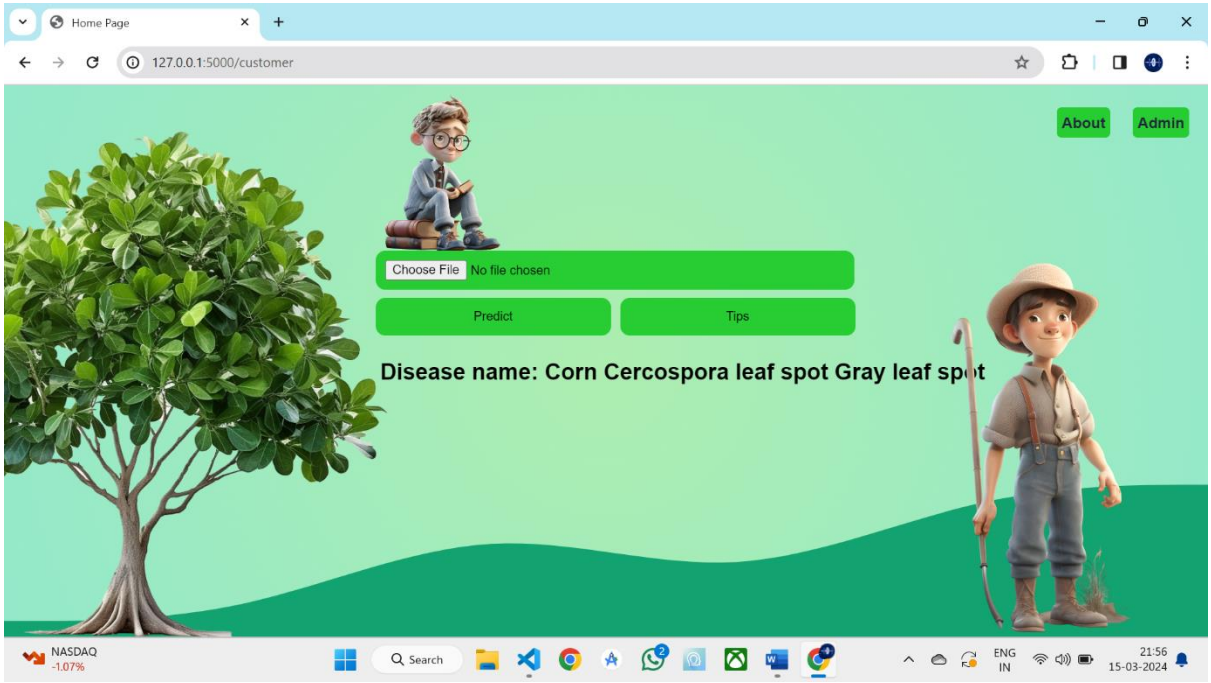
Home Page



File Upload Page

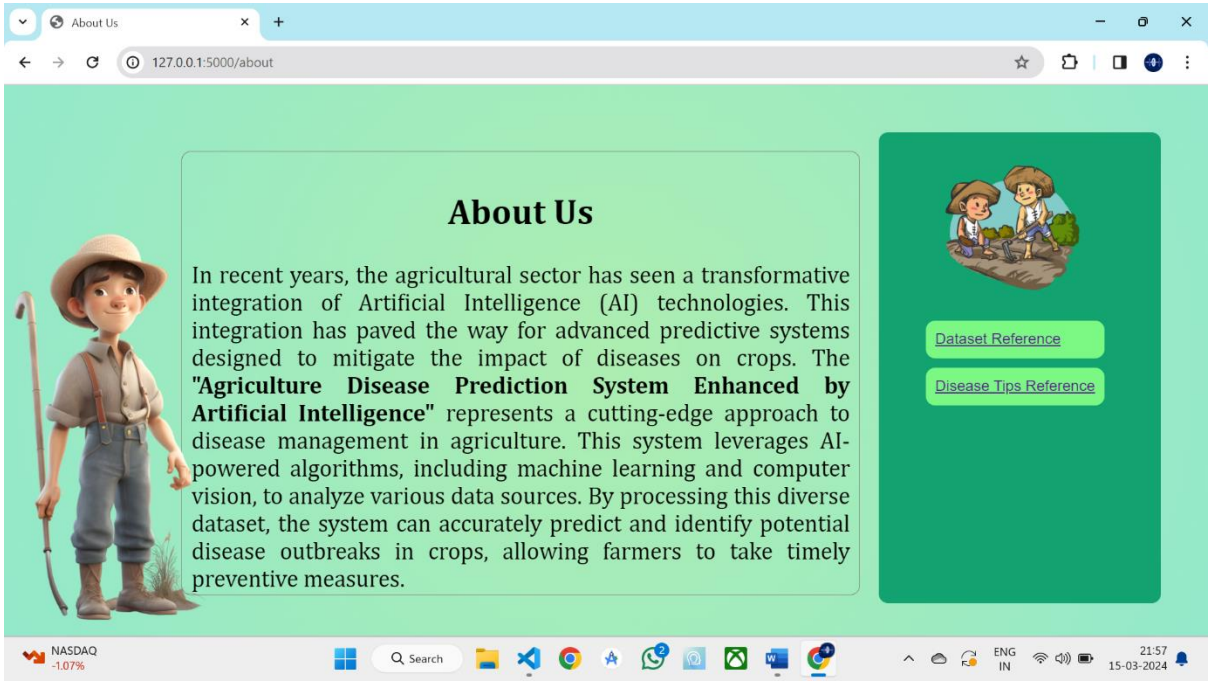


Predict Page

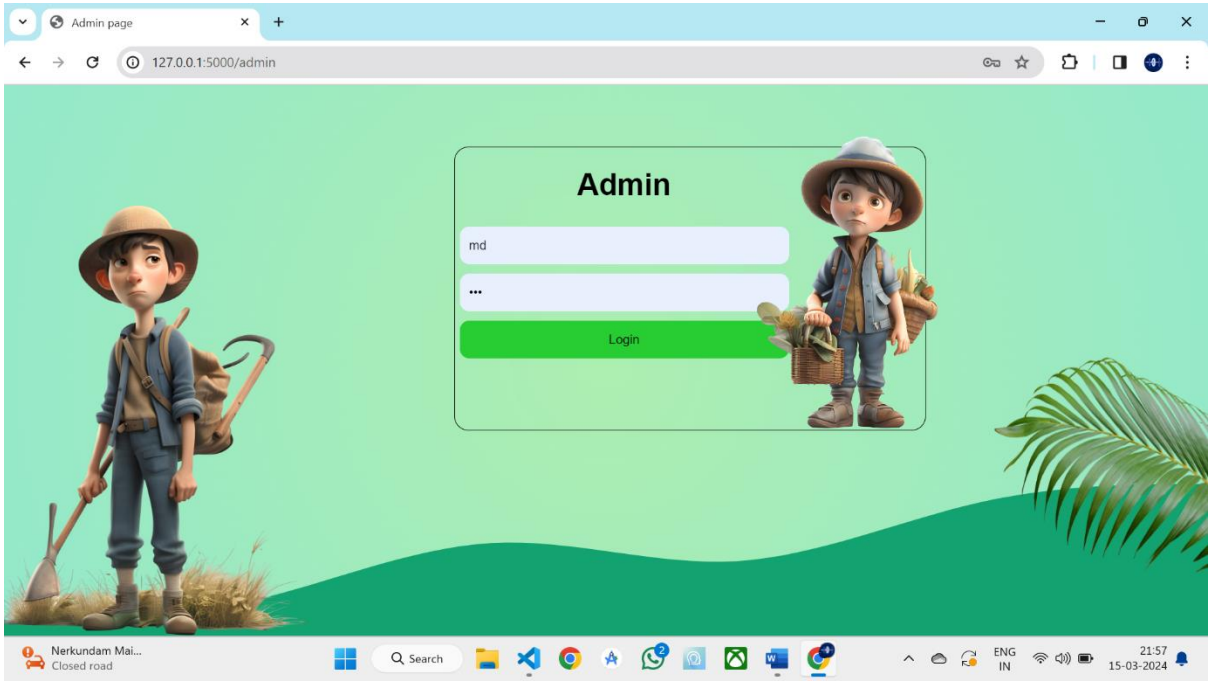


Tips Page

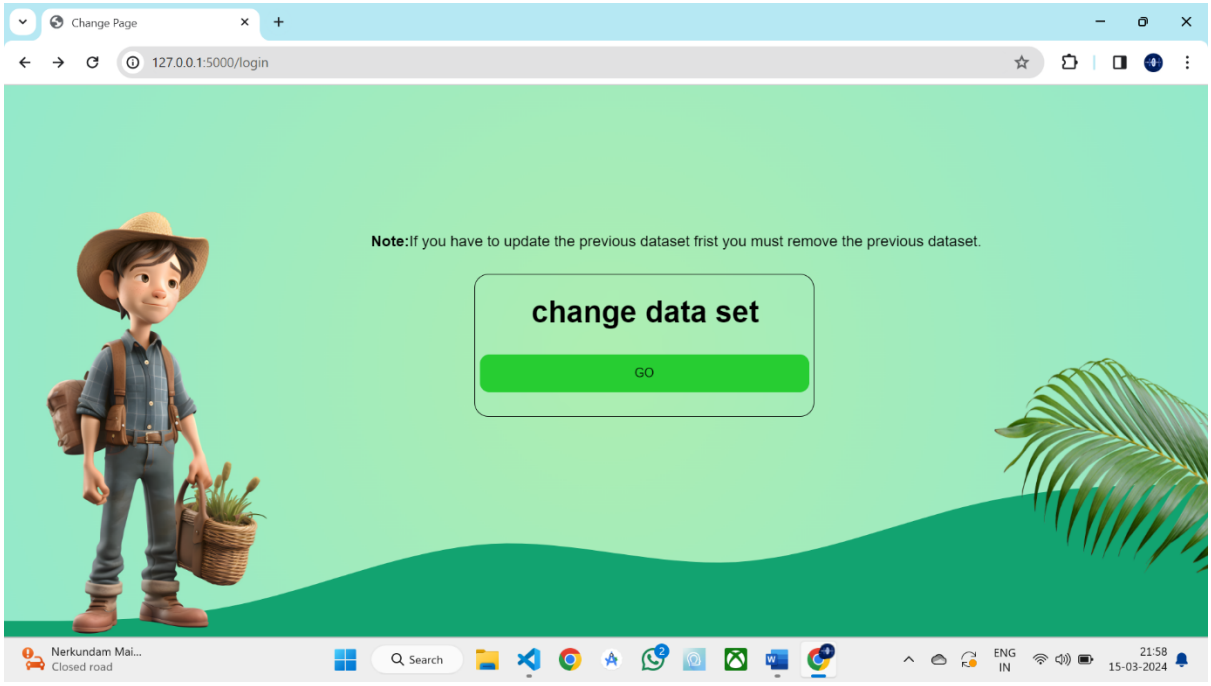
About Us Page



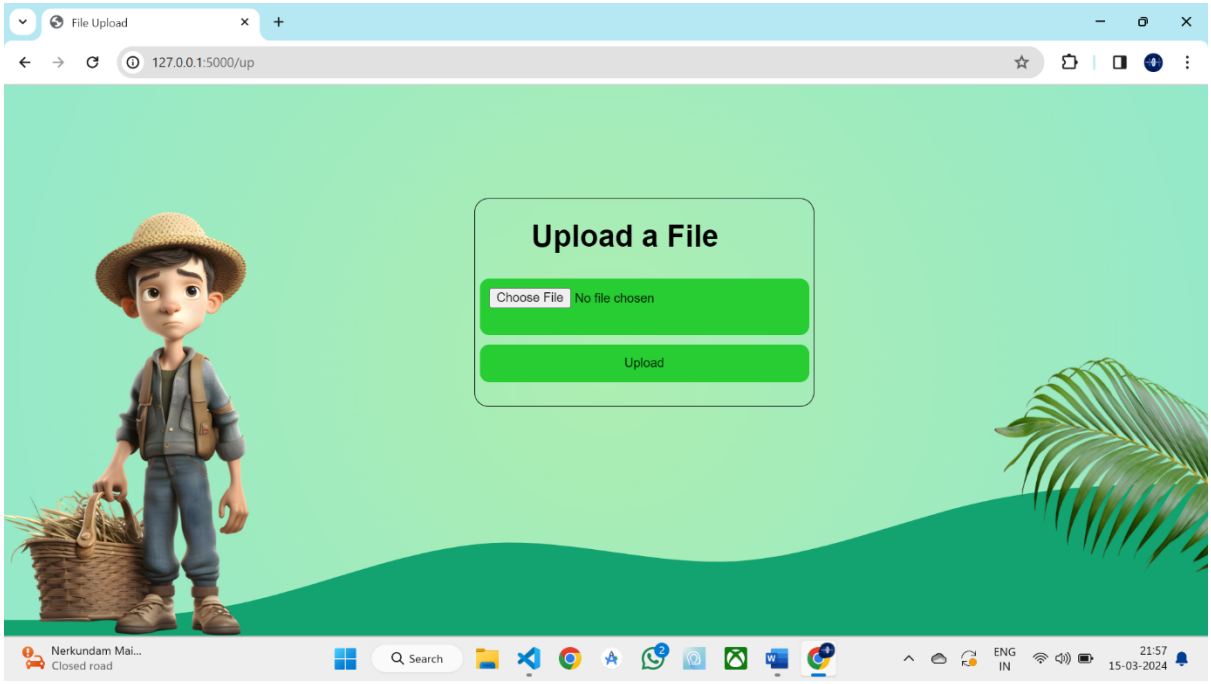
Admin Login Page



Dataset Change Page



Update Dataset Page



8.2 Source Code

Index.html

```
<!doctype html>
<html lang="en">
<head>
<title>Main Page</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<style>
body{
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1)
100%);
background-attachment: fixed;
background-position: center;
```

```
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
margin: 0;
padding: 0;
}
.button{
position: absolute;
top: 50px;
left:250px;
width: 200px;
height: 50px;
background-color:#27cd32;
border-radius: 20px;
}
.box1 {
position: absolute;
border: 1px solid gray;
top:30%;
right: 380px;
height: 200px;
width: 500px;
border-radius: 15px;
box-shadow: 5px 10px 18px #12a370;
}
.button:hover {
background-color: #ffffff;
}
.redpg{
position: relative;
```

```
top:300px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
right: -50px;
bottom: -40px;
}
.img2{
position: relative;
height: 270px;
width: 270px;
bottom: 30px;
left: 1px;
}
.img3{
position: absolute;
height: 200px;
width: 200px;
top: 10px;
left: 1px;
}
.img4{
position: absolute;
height: 450px;
width: 450px;
bottom: -10px;
left: -90px;
}
```

```
.h1{
position: absolute;
top: 10px;
left: 18%;
font-family: 'fantasy';
font-size:60px ;
color: hsl(122, 69%, 20%);
}
.h2{
position: absolute;
top: 85px;
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
right: 90px;
font-size: 40px;
}
</style>
</head>
<body>
<div class="redpg">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1"
d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>
</svg></div>
<h1 class="h1"><b>Agriculture Disease Prediction System </b></h1>
<h1 class="h2">Enhanced by AI</h1>
<div class="box1">
<img src="/static/img1.png" alt="img" class="img2">
<form action="/go" method="POST">
<input type="submit" value="Get start" class="button">
```

```
</form>
</div>
<img src="/static/img2.gif" alt="img" class="img3">
<img src="/static/img1.gif" alt="img" class="img4">
<img src="/static/former.png" alt="img" class="img">
</body>
</html>
```

About.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>About Us</title>
<style>
body {
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
.boxleft{
position: absolute;
height:500px;
width: 300px;
```

```
right: 50px;
border-radius: 10px;
top: 50px;
background-color: #12a370;
}
.boxcenter{
position: absolute;
height: 450px;
width: 700px;
right: 370px;
border-radius: 10px;
top: 70px;
border: 1px solid gray;
backdrop-filter: blur(10px);
background: transparent;
color: rgb(0, 0, 0);
box-shadow: 10px;
text-align: justify;
padding: 10px;
font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;
font-size: 25px;
}
.button1{
position: absolute;
border-radius: 10px;
background-color: #7cf985;
height: 20px;
width: 170px;
padding: 10px;
top: 200px;
```

```
left: 50px;
}
.button2{
position: absolute;
border-radius: 10px;
background-color: #7cf985;
height: 20px;
width: 170px;
padding: 10px;
top: 250px;
left: 50px;
}
.button3{
position: absolute;
border-radius: 10px;
background-color: #7cf985;
height: 40px;
width: 190px;
padding: 10px;
top: 300px;
left: 50px;
}
.img{
width: 150px;
height: 150px;
border-radius: 150px;
top: 25px;
left: 65px;
position: absolute;
}
```

```
.img4{
position: absolute;
height: 450px;
width: 450px;
bottom: -1px;
left: -115px;
}
</style>
</head>
<body>
<div class="boxleft">
<img src="/static/img2.png" alt="image" class="img">
<a href="https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset" class="button1" >Dataset Reference</a><br>
<a href="https://extension.umn.edu/" class="button2" >Disease Tips Reference</a>
</div>
<div class="boxcenter">
<h2><center>About Us</center></h2>
<p>In recent years, the agricultural sector has seen a transformative integration of Artificial Intelligence (AI) technologies.
This integration has paved the way for advanced predictive systems designed to mitigate the impact of diseases on crops.
The <b>"Agriculture Disease Prediction System Enhanced by Artificial Intelligence" </b>represents a cutting-edge approach to disease management in agriculture.
This system leverages AI-powered algorithms, including machine learning and computer vision, to analyze various data sources. By processing this diverse dataset,
the system can accurately predict and identify potential disease outbreaks in crops, allowing farmers to take timely preventive measures.  </p>
</div>
<img src="/static/img13.png" alt="img" class="img4">
</body>
</html>
```

Admin.html

```
<html>
<head>
<title>Admin page</title>
<style>
body{
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
.button {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 350px;
height: 40px;
border-radius: 10px;
margin: 5px;
}
.box1 {
position: absolute;
border: 1px solid rgb(0, 0, 0);
backdrop-filter: blur(10px);
background: transparent;
```

```
top:65px;
right: 300px;
height: 300px;
width: 500px;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
left: -100px;
bottom: -10px;
}
.img2{
position: absolute;
height: 340px;
width: 340px;
right: -100px;
bottom: -10px
}
.img3{
```

```
position: absolute;
height: 340px;
width: 340px;
right: -100px;
bottom: -10px;
}
.h1{
margin-left: 130px;
}
</style>
</head>
<body>
<div class="box1">
<h1 class="h1">Admin</h1>
{% if error_message %}
<p style="color: red;">{{ error_message }}</p>
{% endif %}
<form action="/login" method="post">
<input type="text" name="user" required placeholder=" Enter the UserName" class="button"><br>
<input type="password" name="pass" required placeholder="Enter the Password" class="button"><br>
<input type="submit" value="Login" class="button">
</form>
<img src="/static/img8.png" alt="img" class="img2">
</div>
<div class="redpg">
<img src="/static/img6.png" alt="img" class="img">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1" d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,
```

320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>

</svg>

<img src="/static/img12.png" alt="img" class="img2">

</body>

</html>

Adup.html

<html>

<head>

<title>Change Page</title>

<style>

body{

font-family: Arial, sans-serif;

background: rgb(174,238,177);

background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);;

background-attachment: fixed;

background-position: center;

background-repeat: no-repeat;

background-size: cover;

overflow: hidden;

}

.button {

padding: 10px 10px;

background-color:#27cd32;

color: #000000;

border: none;

width: 350px;

height: 40px;

border-radius: 10px;

margin: 5px;

```
}
.box1 {
position: absolute;
border: 1px solid rgb(0, 0, 0);
backdrop-filter: blur(10px);
background: transparent;
top:200px;
left: 500px;
height: 150px;
width: 360px;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
left: -100px;
bottom: -10px;
}
.img2{
position: absolute;
```

```
height: 340px;
width: 340px;
right: -100px;
bottom: -10px;
}
.h1{
margin-left: 60px;
}
.p2{
position: absolute;
text-align: justify;
left: 390px;
top: 140px;
}
</style>
</head>
<body>
<p class="p2"><b>Note:</b>If you have to update the previous dataset frist you must remove the previous dataset.</p>
<div class="box1">
<h1 class="h1">change data set</h1>
<form action="/up" method="post">
<input type="submit" value="GO" class="button">
</form>
</div>
<div class="redpg">
<img src="/static/img7.png" alt="img" class="img">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1"
d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,
```

320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>

</svg>

<img src="/static/img12.png" alt="img" class="img2">

</body>

</html>

No.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>No MAtch Page</title>

<style>

body{

font-family: Arial, sans-serif;

background: rgb(174,238,177);

background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);

background-attachment: fixed;

background-position: center;

background-repeat: no-repeat;

background-size: cover;

}

.button{

position: absolute;

top: 40%;

left: 30%;

width: 200px;

height: 50px;

background-color:#5cea66;

border-radius: 20px;

}

```
.box1 {
position: absolute;
border: 1px solid gray;
backdrop-filter: blur(10px);
background: transparent;
top:30%;
right: 380px;
height: 200px;
width: 500px;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;

}
</style>
</head>
<body>
<div class="box1">
<h1>No Machase</h1>
</div>
</body>
</html>
```

Removed.html

```
<html>
<head>
<title>Removed Page</title>
<style>
body{
```

```
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
.button {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 350px;
height: 40px;
border-radius: 10px;
margin: 5px;
}
.box1 {
position: absolute;
border: 1px solid rgb(0, 0, 0);
backdrop-filter: blur(10px);
background: transparent;
top:150px;
left: 500px;
height: 200px;
width: 360px;
border-radius: 15px;
}
```

```
.button:hover {
background-color: #ffffff;
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
left: -100px;
bottom: -10px;
}
.img2{
position: absolute;
height: 340px;
width: 340px;
right: -100px;
bottom: -10px;
}
.h1{
margin-left: 60px;
}
.p2{
position: absolute;
text-align: justify;
left: 390px;
```

```
top: 140px;
}
</style>
</head>
<body>
<div class="box1">
<h1 class="h1">Removed Successfully </h1>
<form action="/back2" method="post">
<input type="submit" value="Back" class="button">
</form>
</div>
<div class="redpg">
<img src="/static/img6.png" alt="img" class="img">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1" d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>
</svg></div>
<img src="/static/img12.png" alt="img" class="img2">
</body>
</html>
```

Success.html

```
<html>
<head>
<title>Upladed page</title>
<style>
body{
font-family: Arial, sans-serif;
background: rgb(174,238,177);
```

```
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
.button {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 350px;
height: 40px;
border-radius: 10px;
margin: 5px;
}
.box1 {
position: absolute;
border: 1px solid rgb(0, 0, 0);
backdrop-filter: blur(10px);
background: transparent;
top:120px;
left: 500px;
height: 220px;
width: 360px;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;
```

```
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
left: -100px;
bottom: -10px;
}
.img2{
position: absolute;
height: 340px;
width: 340px;
right: -100px;
bottom: -10px;
}
.h1{
margin-left: 60px;
}
</style>
</head>
<body>
<div class="box1">
<h1 class="h1">Successfully updated</h1>
<form action="/success" method="post">
```

<input type="submit" value="Back" class="button">

</form>

<img src="/static/img10.png" alt="img" class="img">

<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1" d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>

</svg>

<img src="/static/img12.png" alt="img" class="img2">

</body>

</html>

Tip.html

<!DOCTYPE html>

<head>

<title>Tips Page</title>

<link rel="stylesheet" type="text/css" href="static/css/imsge.css">

<style>

body {

 font-family: Arial, sans-serif;

background: rgb(174,238,177);

background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);

background-attachment: fixed;

background-position: center;

background-repeat: no-repeat;

background-size: cover;

overflow: hidden;

}

```
.container{
width: 600px;
height: 300px;
padding: 60px;
position: relative ;
border-radius: 10px;
text-align: left;
margin-left: 75px;
display: flex;
flex-direction: column;
justify-content: center;
font-family: 'Times New Roman', Times, serif;
font-size: 20px;
left: 150px;
top: 20px;
border: 1.5px solid rgb(8, 8, 8);
backdrop-filter: blur(10px);
background: transparent;
}
.img{
position: relative;
margin-top: 10px;
top: 100px;
width: 50px;
height: 50px;
border-radius: 10px;
margin-left: 900px;
}
.button {
padding: 10px 10px;
```

```
background-color: #27cd32;
color: #030303;
border: none;
width: 150px;
height: 20px;
border-radius: 10px;
margin: 5px;
cursor: pointer;
position: relative;
left: 59px;
top: 130px;
}
.button2 {
display: inline-block;
padding: 10px 10px;
background-color:   #27cd32;
color: #000000;
border: none;
width: 150px;
border-radius: 10px;
margin: 5px;
cursor: pointer;
position: relative;
left: 59px;
top: 140px;
}
.container2{
position: absolute;
height: 355px;
width: 290px;
```

```
right: 70px;
top: 130px;
border: 1px solid gray;
backdrop-filter: blur(15px);
background: transparent;
background-color:  #12a370;
align-items: center;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;
}
.button2:hover {
background-color: #ffffff;
}
.h1{
position: absolute;
top: 10px;
}
.clo{
color: rgb(0, 0, 0);
}
.co1{
position: absolute;
 top: -100px;
}
.img{
position: absolute;
height: 250px;
width: 250px;
```

```
margin-left: 30px;
margin-top: 60px;
}
.img2{
position: absolute;
height: 400px;
width: 400px;
left: -40px;
top: 190px;
}
</style>
</head>
<body>
<center><h1>{{name}}</h1></center>
<div class="container">
<ul>
{% for sentence in hello.split('.') %}
{% if sentence %}
<li>{{ sentence.strip() }}</li>
{% endif %}
{% endfor %}
</ul>
</div>
<img src="/static/img3.png" alt="img" class="img2">
<div class="container2">
<div class="co1">
<div class="button">
<a href={{medi}}><color class="clo">Medicine</color></a><br><br>
</div>
<div class="button2">
```

```
<a href={{more}}><color class="clo">More About Tips</color></a>
</div>
</div>
<img src="/static/img5.png" alt="img" class="img">
</div>
</body>
</html>
<!DOCTYPE html>
```

Upload.html

```
<!DOCTYPE html>
<html>
<head>
<title>Home Page</title>
<meta charset="UTF-8">
<link rel="stylesheet" href="{{url_for('static',filename='css/bootstrap.min.css')}}">
<link rel="stylesheet" href="{{url_for('static',filename='myfont/css/all.min.css')}}">
<script src="{{url_for('static',filename='js/jquery.min.js')}}">
</script>
<style>
body {
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
```

```
.top-right {
position: absolute;
top: 10px;
right: 10px;
}
nav {
text-align: center;
margin-top: 20px;
}
nav a {
text-decoration: none;
color: #24243e;
background-color: #27cd32;
padding: 7px 5px;
margin: 10px;
border-radius: 5px;
font-weight: bold;
}
nav a:hover {
background-color: #ffffff;
color: #000000;
}
.button1 {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 489px;
border-radius: 10px;
margin: 5px;
```

```
}
.button {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 250px;
height: 40px;
border-radius: 10px;
margin: 5px;
}
.button:hover {
background-color: #ffffff;
}
.button1:hover {
background-color: #ffffff;
}
.b1{
position: absolute;
}
.b2{
position: absolute;
margin-top: 50px;
}
.b3{
position: absolute;
margin-top: 50px;
margin-left: 260px;
}
.label {
```

```
color: black;
font-size: 1.5em;
font-weight: bold;
}
.l1{
position: absolute;
margin-top: 120px;
margin-left: 10px;
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 430px;
width: 430px;
right: -20px;
top: -140px;
}
.img2{
position: absolute;
height: 185px;
width: 185px;
left: 370px;
top: 2px;
}
.img3{
```

```
position: absolute;
height: 655px;
width: 655px;
left: -200px;
bottom: -60px;
}
.con{
position: absolute;
height: 200px;
width: 700px;
top: 170px;
left: 390px;
}
</style>
</head>
<body>
<nav class="top-right">
<a href="/about">About</a>
<a href="/admin">Admin</a>
</nav>
<img src="/static/img11.png" alt="img" class="img2">
<div class="con">
<form method="POST" enctype="multipart/form-data" action="/customer">
<div class="b1">
<input id="button1" type="file" name="file" accept=".jpg, .jpeg, .png" class="button1">
</div>
<div class="b2">
<input id="button2" type="submit" value="Predict" class="button">
</div>
</form>
```

```
<div class="l1">
<div class="label">
Disease name: {{data}}
</div>
</div>
<div class="b3">
<form method="POST" action="/tips">
<input id="button2" type="submit" value="Tips" class="button">
</form>
</div>
</div>
<div class="redpg">
<img src="/static/img13.png" alt="img" class="img">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1" d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>
</svg>
</div>
<img src="/static/img4.png" alt="img" class="img3">
</body>
</html>
```

Upload2.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>File Upload</title>
```

```
<style>
body{
font-family: Arial, sans-serif;
background: rgb(174,238,177);
background: radial-gradient(circle, rgba(174,238,177,1) 0%, rgba(148,233,203,1) 100%);
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
overflow: hidden;
}
.button {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 350px;
height: 40px;
border-radius: 10px;
margin: 5px;
}
.button2 {
padding: 10px 10px;
background-color:#27cd32;
color: #000000;
border: none;
width: 330px;
height: 40px;
border-radius: 10px;
margin: 5px;
```

```
}
.box1 {
position: absolute;
border: 1px solid rgb(0, 0, 0);
backdrop-filter: blur(10px);
background: transparent;
top:120px;
left: 500px;
height: 220px;
width: 360px;
border-radius: 15px;
}
.button:hover {
background-color: #ffffff;
}
.button2:hover {
background-color: #ffffff;
}
.redpg{
position: relative;
top:300px;
width: 1300px;
right: 10px;
}
.img{
position: absolute;
height: 500px;
width: 500px;
left: -70px;
bottom: -10px;
```

```
}
.img2{
position: absolute;
height: 340px;
width: 340px;
right: -100px;
bottom: -10px;
}
.h1{
margin-left: 60px;
}
</style>
</head>
<body>
<div class="box1">
<h1 class="h1">Upload a File</h1>
{% if existing_file %}
<p>You already have a file uploaded: {{ existing_file }}. <form action="/remove" method="post"><button type="submit" class="button">Remove</button></form></p>
{% else %}
<form action="/upload2" method="post" enctype="multipart/form-data">
<input type="file" name="file" class="button2">
<button type="submit" class="button">Upload</button>
</form>
{% endif %}
</div>
<div class="redpg">
<img src="/static/img9.png" alt="img" class="img">
<svg xmlns="http://www.w3.org/2000/svg"  viewBox="0 0 1440 320"><path fill="#12a370" fill-opacity="1" d="M0,288L48,288C96,288,192,288,288,266.7C384,245,480,203,576,197.3C672,192,768,224,864,218.7C960,213,1056,171,1152,149.3C1248,128,1344,128,1392,128L1440,128L1440,
```

320L1392,320C1344,320,1248,320,1152,320C1056,320,960,320,864,320C768,320,672,320,576,320C480,320,384,320,288,320C192,320,96,320,48,320L0,320Z"></path>

</svg>

<img src="/static/img12.png" alt="img" class="img2">

</body>

</html>

App.py

```
import os

from flask import Flask, request, render_template

from predict import predict_image_class

from werkzeug.utils import secure_filename

from flask import Flask, render_template,request,flash,redirect,url_for,session

import sqlite3

from tips2 import scrape_quick_facts

app = Flask(__name__)

app.config['UPLOAD_FOLDER'] = 'uploads'

app.config['ALLOWED_EXTENSIONS'] = {'jpg', 'jpeg', 'png'}

val = None

def allowed_file(filename):

return '.' in filename and filename.rsplit('.', 1)[1].lower() in
app.config['ALLOWED_EXTENSIONS']

app.secret_key="123"

con=sqlite3.connect("database.db")

con.execute("create table if not exists customer(pid integer primary key,name text,address
text,contact integer,mail text)")

con.close()

@app.route('/')

def index():

return render_template('index.html')
```

```
@app.route('/go',methods=["GET","POST"])
def go():
return render_template('upload.html')
#image proccess
@app.route('/customer',methods=["GET","POST"])
def upload_image():
if request.method == 'POST':
if 'file' not in request.files:
return "No file part"
file = request.files['file']
if file.filename == '':
n="No selected file"
return render_template('upload.html',data=n)
if file and allowed_file(file.filename):
global filename
filename = secure_filename(file.filename)
filepath = os.path.join(app.config['UPLOAD_FOLDER'], filename)
file.save(filepath)
global val
val= predict_image_class(filepath)
#return f"name {val}"
return render_template('upload.html',data=val)
return render_template('upload.html')
#get tips and medicine
@app.route('/tips', methods=['GET', 'POST'])
def tips():
global val
hi=val

if hi =='Apple Black rot':
```

target_url = "https://extension.umn.edu/plant-diseases/cedar-apple-rust"

m1="https://www.indiamart.com/proddetail/fungi-organic-fungicide-23064943791.html?pos=1&kwd=apple%20black%20rot%20fungicide&tags=A||||0|Price|product|pfSt|selam|TS|type=attr=1|attrS|br"

mo1="https://extension.umn.edu/plant-diseases/cedar-apple-rust"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=mo1)

elif hi== 'Apple_scab':

target_url = "https://extension.umn.edu/plant-diseases/apple-scab"

mo1="https://extension.umn.edu/plant-diseases/apple-scab"
m1="https://dir.indiamart.com/search.mp?ss=apple+scab+fungicide&mcatid=4883&catid=12&src=as-popular%257Ckwd%253Dapple%2520scab%257Cpos%253D2%257Ccat%253D12%257Cmcat%253D4883%257Ckwd_len%253D10%257Ckwd_cnt%253D2%7C&prdsrc=1&stype=attr=1|attrS&res=RC3&com-cf=nl&qu=to&qry_typ=P"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=mo1)

elif hi== 'Apple Cedar apple rust':

target_url = "https://extension.umn.edu/plant-diseases/cedar-apple-rust"

mo1 = "https://extension.umn.edu/plant-diseases/cedar-apple-rust"

m1="https://www.indiamart.com/proddetail/sulphur-80-wdg-fungicide-2852058008130.html?pos=1&kwd=apple%20scab%20fungicide&tags=||||1292.7979|Price|product|pfSt|selam|TS|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=mo1)

elif hi== 'Apple healthy':

target_url = "https://extension.umn.edu/fruit/growing-apples"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Blueberry healthy':

target_url = "https://extension.umn.edu/fruit/growing-blueberries-home-garden"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

```
elif hi== 'Cherry healthy':

target_url = "https://extension.umn.edu/plant-diseases/brown-rot-stone-fruit"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Cherry Powdery mildew':

target_url = "https://extension.umn.edu/plant-diseases/powdery-mildew-trees-and-shrubs"

m1="https://www.indiamart.com/proddetail/basf-acrisio-fungicide-2852928185230.html?pos=2&kwd=cherry%20powdery%20mildew%20fungicide&tags=A||||141.88246|Price|product|pfSt|selam|TS"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Corn Cercospora leaf spot Gray leaf spot':

target_url = "https://extension.umn.edu/corn-pest-management/corn-ear-rots-and-mycotoxins"

m1="https://www.indiamart.com/proddetail/bayer-nativo-fungicide-2852888121962.html?pos=1&kwd=corn%20circospora%20leaf%20spot%20gray%20leaf%20spotfungicide&tags=||||141.88246|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Corn Common rust ':

target_url = "https://extension.umn.edu/corn-pest-management/corn-ear-rots-and-mycotoxins"

m1="https://www.indiamart.com/proddetail/krilaxyl-fungicide-2850376276733.html?pos=2&kwd=corn%20common%20rust%20fungicide&tags=A||||1517.1705|Price|product|pfSt|selam|MDC"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Corn healthy':

target_url = "https://extension.umn.edu/vegetables/growing-sweet-corn"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)
```

elif hi== 'Corn Northern Leaf Blight':

target_url = "https://extension.umn.edu/corn-pest-management/corn-ear-rots-and-mycotoxins"

m1="https://www.indiamart.com/proddetail/roko-fungicide-powder-2853220848062.html?pos=2&kwd=corn%20northern%20leaf%20blight%20fungicide&tags=BA||||44.23123|Price|product|pfSt|selam|MDC|type=attr=1|attrS|br

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Grape Black rot':

target_url = "https://extension.umn.edu/fruit/growing-grapes-home-garden"

m1="https://www.indiamart.com/proddetail/multiplex-black-out-bactericide-2853467666648.html?pos=1&kwd=grape%20black%20rot%20fungicide&tags=A||||141.88246|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Grape Esca':

target_url = "https://extension.umn.edu/fruit/growing-grapes-home-garden"

m1="https://www.indiamart.com/proddetail/roko-fungicide-powder-2853220848062.html?pos=1&kwd=grape%20esca%20fungicide&tags=||||44.23123|Price|product|pfSt|selam|MDC|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Grape healthy':

target_url = "https://extension.umn.edu/fruit/growing-grapes-home-garden"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Grape Leaf blight':

target_url = "https://extension.umn.edu/plant-diseases/leaf-spot-diseases-trees-and-shrubs"

m1="https://www.indiamart.com/proddetail/power-plant-virosol-2851223659812.html?pos=4&kwd=grape%20leaf%20blight%20fungicide&tags=BA||||228.01602||product||selam|MDC|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Orange Haunglongbing':

target_url = "https://extension.umn.edu/identify-invasive-species/orange-hawkweed"

m1="https://www.indiamart.com/proddetail/antinematocide-sp-fungicides-24197640897.html?pos=5&kwd=orange%20haunglong%20fungicide&tags=BA||||917.8211|Price|product||selam|MDC|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Peach Bacterial spot':

target_url = "https://extension.umn.edu/plant-diseases/leaf-spot-diseases-trees-and-shrubs"

m1="https://www.indiamart.com/proddetail/organic-anti-bacterial-fertilizer-16671888962.html?pos=7&kwd=peach%20bacterial%20spot%20fungicide&tags=A||||1542.9468||product|pfSt|selam|TS"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Peach healthy':

target_url = "https://extension.umn.edu/fruit/growing-stone-fruits-home-garden"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Pepper bell Bacterial spot':

target_url = "https://extension.umn.edu/disease-management/bacterial-spot-tomato-and-pepper"

m1="https://www.indiamart.com/proddetail/bacto-nill-crop-protectant-bactericide-16687809830.html?pos=7&kwd=pepper%20bell%20bacterial%20spot%20fungicide&tags=A||||808.72754|Price|product|pfSt|selam|TS"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Pepper bell healthy':

target_url = "https://extension.umn.edu/vegetables/growing-peppers"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Potato Early blight':

target_url = "https://extension.umn.edu/disease-management/early-blight-tomato-and-potato"

m1="https://www.indiamart.com/proddetail/monceren-pencycuron-22-9-sc-1-ltr-2850887159555.html?pos=5&kwd=potato%20early%20blight%20fungicide&tags=A|||1883.668|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Potato healthy':

target_url = "https://extension.umn.edu/vegetables/growing-potatoes"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Potato Late blight':

target_url = "https://extension.umn.edu/disease-management/late-blight"

m1="https://www.indiamart.com/proddetail/click-late-blight-downey-mildew-control-biofungicide-26733930933.html?pos=4&kwd=potato%20late%20blight%20fungicide&tags=||||640.0239|Price|product||selam|NA|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Raspberry healthy':

target_url = "https://extension.umn.edu/fruit/growing-raspberries-home-garden"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Soybean healthy':

target_url = "https://extension.umn.edu/soybean-pest-management/soybean-cyst-nematode-management-guide"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Squash Powdery mildew':

target_url = "https://extension.umn.edu/disease-management/downy-mildew-cucurbits"

m1="https://www.indiamart.com/proddetail/fungicide-powder-2853267128373.html?pos=5&kwd=squash%20powder%20mildew%20fungicide&tags=A||||1292.7979|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Strawberry healthy':

target_url = "https://extension.umn.edu/fruit/growing-strawberries-home-garden"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Strawberry Leaf scorch':

target_url = "https://extension.umn.edu/strawberry-farming/strawberry-diseases-minnesota"

m1="https://www.indiamart.com/proddetail/bio-fungicides-23085627288.html?pos=6&kwd=strawberry%20leaf%20scotch%20fungicide&tags=||||1006.8655|Price|product|pfSt|selam|TS|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Bacterial spot':

target_url = "https://extension.umn.edu/disease-management/bacterial-canker-tomato"

m1="https://www.indiamart.com/proddetail/anti-bacterial-anti-fungal-3255502330.html?pos=5&kwd=tomato%20bacterial%20spot%20fungicides&tags=A||||952.8169|Price|product||selam|MDC|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Early blight':

target_url = "https://extension.umn.edu/disease-management/early-blight-tomato-and-potato"

m1="https://www.indiamart.com/proddetail/blightbuster-matelaxy-8-mancozeb-64-wp-2851614720633.html?pos=3&kwd=tomato%20early%20blight%20fungicides&tags=||||1688.8075|Price|product||selam|MDC"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

```
elif hi== 'Tomato healthy':

target_url = "https://extension.umn.edu/vegetables/growing-tomatoes"

hi=scrape_quick_facts(target_url)

return render_template('tip.html',hello=hi,name=val,image=filename,more=target_url)

elif hi== 'Tomato Late blight':

target_url = "https://extension.umn.edu/disease-management/late-blight"

m1="https://www.indiamart.com/proddetail/sulphur-80-wdg-
24658877791.html?pos=3&kwd=tomato%20late%20blight%20fungicides&tags=BB||||0|Price
|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Leaf Mold':

target_url = "https://extension.umn.edu/disease-management/tomato-leaf-mold"

m1="https://www.indiamart.com/proddetail/bannari-leaf-guard-licheniformis-
21450379130.html?pos=1&kwd=tomato%20leaf%20mold%20fungicides&tags=||||135.49039
|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Septoria leaf spot':

target_url = "https://extension.umn.edu/plant-diseases/tomato-leaf-spot-diseases"

m1="https://www.indiamart.com/proddetail/dise-nill-plant-growth-promoter-and-
biofungicide-
14971067048.html?pos=1&kwd=tomato%20septoria%20leaf%20spot%20fungicides&tags=||
||44.541527||product|pfSt|selam|TS|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=mo1,more=target_url)

elif hi== 'Tomato Spider mites':

target_url = "https://extension.umn.edu/yard-and-garden-insects/spider-mites"

m1="https://www.indiamart.com/proddetail/daksh-systemic-organic-fungicides-
24821699655.html?pos=1&kwd=tomato%20spider%20mites%20fungicides&tags=||||141.882
46|Price|product||selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)
```

```
return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Target Spot':

target_url = "https://extension.umn.edu/disease-management/early-blight-tomato-and-potato"

m1="https://www.indiamart.com/proddetail/bayer-antracol-fungicide-
23376359862.html?pos=4&kwd=tomato%20target%20spot%20fungicides&tags=BA||||141.8
8246||product||selam|MDC|type=attr=1|attrS"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato mosaic virus':

target_url = "https://extension.umn.edu/disease-management/tomato-viruses"

m1="https://www.indiamart.com/proddetail/plant-virus-controller-
21809276648.html?pos=2&kwd=tomato%20mosaic%20virus%20fungicides&tags=A||||1265.
247|Price|product|pfSt|selam|TS|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== 'Tomato Yellow Leaf Curl Virus':

target_url = "https://extension.umn.edu/plant-diseases/viruses-backyard-fruit"

m1="https://www.indiamart.com/proddetail/metarhizium-anisopliae-pesticides-
23885192448.html?pos=3&kwd=tomato%20yellow%20leaf%20curl%20virus%20fungicides
&tags=A||||1089.0875|Price|product|pfSt|selam|SSnp|type=attr=1|attrS|attrMtch=1"

hi=scrape_quick_facts(target_url)

return
render_template('tip.html',hello=hi,name=val,image=filename,medi=m1,more=target_url)

elif hi== '':

return render_template('no.html')

else:

return render_template('no.html')

@app.route('/about')

def about():

return render_template('about.html')

@app.route('/admin')
```

```
def admin():
return render_template('admin.html')
#admin login
@app.route('/login',methods=["GET","POST"])
def login():
if request.method=='POST':
name=request.form['user']
password=request.form['pass']
con=sqlite3.connect("database.db")
con.row_factory=sqlite3.Row
cur=con.cursor()
cur.execute("select * from customer where name=? and mail=?",(name,password))
data=cur.fetchone()
if data:
return render_template('adup.html')
else:
error_message = 'Invalid username or password. Please try again.'
return render_template('admin.html', error_message=error_message)
return redirect(url_for("admin"))
#admin upload
UPLOAD_FOLDER1 = 'admin'
app.config['UPLOAD_FOLDER1'] = UPLOAD_FOLDER1
def get_uploaded_file():
files = os.listdir(app.config['UPLOAD_FOLDER1'])
if files:
return files[0]
else:
return None
@app.route('/success', methods=['POST'])
def about1():
```

```
return render_template('adup.html')
@app.route('/back2', methods=['POST'])
def about2():
return render_template('upload2.html')
@app.route('/up', methods=['POST'])
def upload_form1():
existing_file = get_uploaded_file()
return render_template('upload2.html', existing_file=existing_file)

@app.route('/upload2', methods=['POST'])
def upload_file():
existing_file = get_uploaded_file()
if existing_file:
return "You already have a file uploaded."
else:
uploaded_file = request.files['file']
if uploaded_file.filename != '':
# Save the file to the upload folder
uploaded_file.save(os.path.join(app.config['UPLOAD_FOLDER1'], uploaded_file.filename))
return render_template('success.html')
@app.route('/remove', methods=['POST'])
def remove_file():
existing_file = get_uploaded_file()
if existing_file:
file_path = os.path.join(app.config['UPLOAD_FOLDER1'], existing_file)
if os.path.exists(file_path):
os.remove(file_path)
return render_template('removed.html')
#main
if __name__ == '__main__':
```

app.run(debug=True)

Predict.py

```
import tensorflow as tf
from tensorflow.keras.preprocessing import image
import numpy as np
model = tf.keras.models.load_model('models/plant.h5')
# Function to preprocess the user-uploaded image
def preprocess_image(user_image_path):
img = image.load_img(user_image_path, target_size=(224,224,3))
img = image.img_to_array(img)
img = np.expand_dims(img, axis=0)
img = img / 255.0
return img
# Function to predict the class of the user-uploaded image
def predict_image_class(user_image_path):
img = preprocess_image(user_image_path)
predictions = model.predict(img)
predicted_class_index = np.argmax(predictions)
class_labels = ['Apple_scab', 'Apple Black rot', 'Apple Cedar apple rust',
'Apple healthy','Blueberry healthy','Cherry healthy',
'Cherry Powdery mildew','Corn Cercospora leaf spot Gray leaf spot','Corn Common rust ',
'Corn healthy','Corn Northern Leaf Blight','Grape Black rot',
'Grape Esca','Grape Leaf blight','Grape healthy',
'Orange Haunglongbing','Peach Bacterial spot'
,'Peach healthy','Pepper bell Bacterial spot','Pepper bell healthy',
'Potato Early blight','Potato healthy','Potato Late blight',
'Raspberry healthy','Soybean healthy','Squash Powdery mildew'
,'Strawberry healthy','Strawberry Leaf scorch','Tomato Bacterial spot',
```

```
'Tomato Early blight','Tomato healthy','Tomato Late blight',
'Tomato Leaf Mold','Tomato Septoria leaf spot','Tomato Spider mites',
'Tomato Target Spot','Tomato mosaic virus','Tomato Yellow Leaf Curl Virus']
predicted_class = class_labels[predicted_class_index]
#confidence = predictions[0][predicted_class_index]
return predicted_class
# Example usage
#user_uploaded_image_path = 'strowberry.jpeg'
#predicted_class, confidence = predict_image_class(user_uploaded_image_path)
#print(f"Predicted Class: {predicted_class} with confidence: {confidence:.2f}")
```

Tips2.py

```
import requests
from bs4 import BeautifulSoup
def scrape_quick_facts(url):
response = requests.get(url)
# Check if the request was successful (status code 200)
if response.status_code == 200:
soup = BeautifulSoup(response.content, 'html.parser')
quick_facts_section = soup.find('div', class_='quick-facts')
if quick_facts_section:
quick_facts_lines = quick_facts_section.text.strip()
return quick_facts_lines
else:
print("Quick facts section not found on the page.")
else:
print(f"Failed to retrieve the page. Status code: {response.status_code}")
```