Fertilizers Recommendation System For Disease Prediction

TEAM ID:PNT2022TMID26444

TEAM MEMBERS

SRI SURAJ SR - 211719104137

SIVAKUMAR P - 211719104132

SRINESH R - 211719104135

SUNDARESWAR - 211719104143

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

This project is used to test the fruits and vegetables samples and identify the Different diseases. Also ,this project recommends fertilizers for predicted diseases.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

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. Also proposed a simple prediction method for soil based fertilizer recommend ratio system for predicted crop diseases. This method gives less accuracy and prediction. 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

[1]. R Indumathi.; N Saagari.; V Thejuswini.; R Swarnareka.," Leaf Disease Detectionand Fertilizer Suggestion", IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), 29-30 March 2019, DOI: 10.1109/ICSCAN.2019.8878781.

[2]. P. Pandi Selvi, P. Poornima, "Soil Based Fertilizer Recommendation System for Crop Disease Prediction System", International Journal of Engineering Trends and Applications (IJETA) – Volume 8 Issue 2, Mar-Apr 2021 .

[3]. H Shiva reddy, Ganesh hedge, Prof. DR Chinnaya3, "IoT based Leaf Disease

Detection and Fertilizer Recommendation", International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 11, Nov 2019, e-ISSN: 2395-0056.

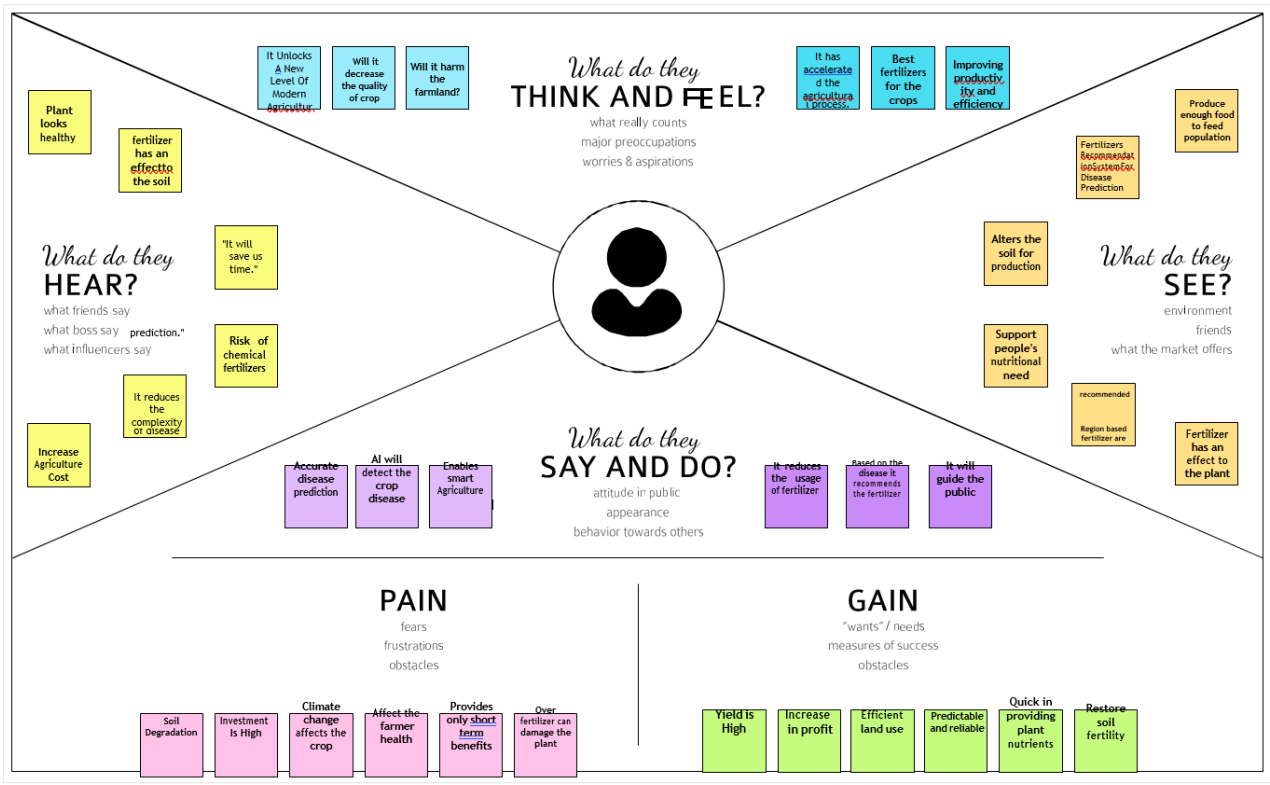
2.3 PROBLEM STATEMENT DEFINITION

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

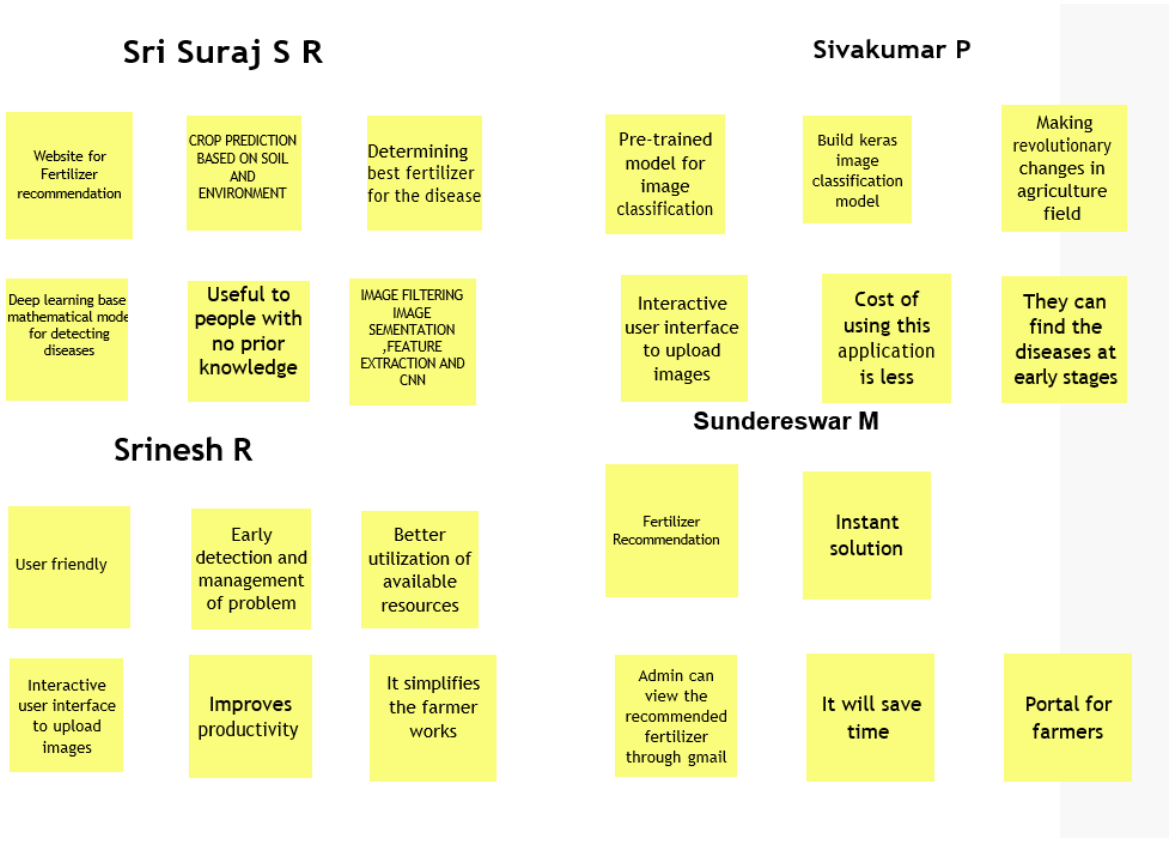
3. IDEATION &PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

Agriculture plays a major role in economic growth and development. As the provider of food it is acornerstone of human existence. As a furnisher of industrial raw materials it is an important contributor to economic activity in other sectors of the economy,For our project, we are getting surveys from farmers to understand what they truly require and desire


What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
It Unlocks A New Level Of Modern Agricultur
Will it decrease the quality of crop
Will it harm the farmland?
It has accelerated the agricultural process.
Best fertilizers for the crops
Improving productivity and efficiency
Plant looks healthy
fertilizer has an effectto the soil
Produce enough food to feed population
Fertilizers Recommendat ionSystemFor Disease Prediction
What do they
HEAR?
what friends say
what boss say
what influencers say
"It will save us time."
Risk of chemical fertilizers
It reduces the complexity or disease
Increase Agriculture Cost
prediction."
Alters the soil for production
Support people's nutritional need
What do they
SEE?
environment
friends
what the market offers
recommended
Region based fertilizer are
Fertilizer has an effect to the plant
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Accurate disease prediction
AI will detect the crop disease
Enables smart Agriculture
It reduces the usage of fertilizer
Based on the disease it recommends the fertilizer
It will guide the public
PAIN
fears
frustrations
obstacles
Soil Degradation
Investment Is High
Climate change affects the crop
Affect the farmer health
Provides only short term benefits
Over fertilizer can damage the plant
GAIN
"wants" / needs
measures of success
obstacles
Yield Is High
Increase in profit
Efficient land use
Predictable and reliable
Quick in providing plant nutrients
Restore soil fertility

3.2 IDEATION & BRAINSTROMING



3.3 PROPOSED SOLUTION

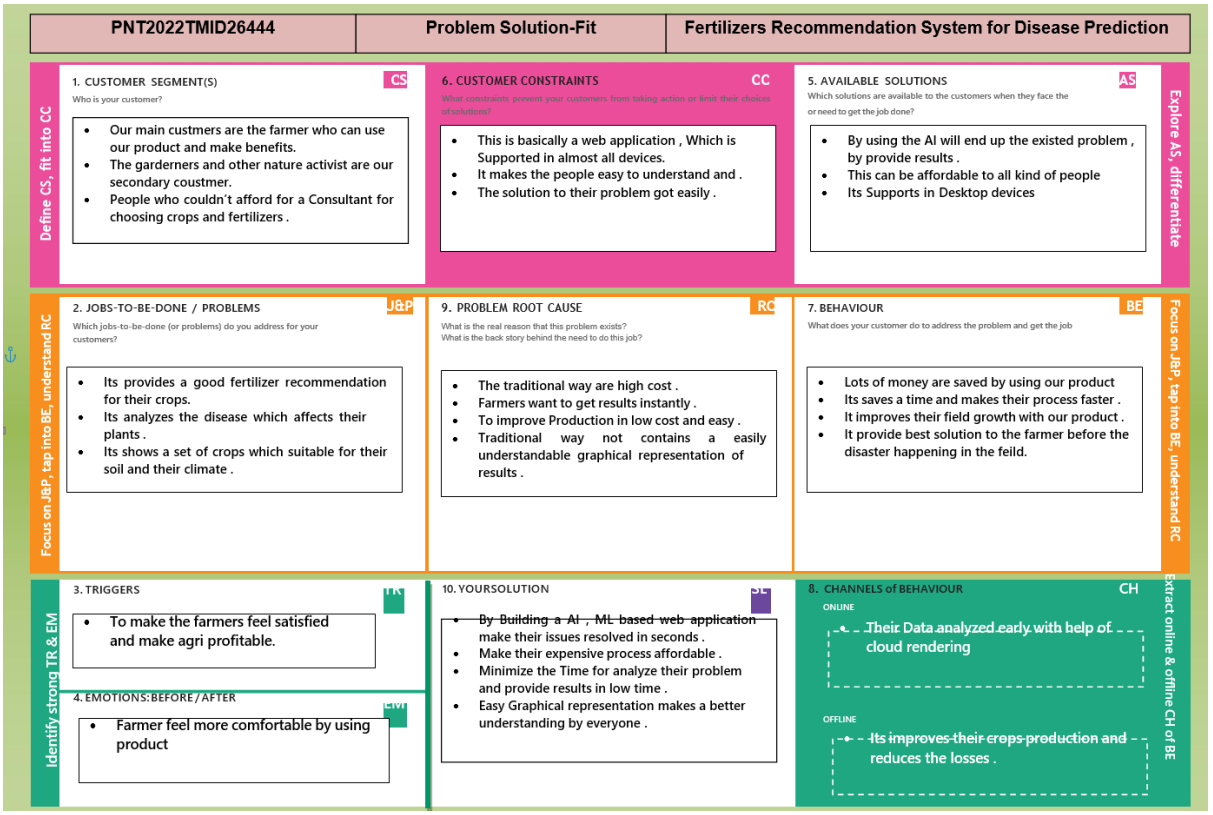
Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No	Parameter	Description
1.	Problem Statement (Problem to besolved)	• Because of a lack of expertise and outdated procedures, farmers are unable to identify crop diseases. • When only a few crops are grown, the soil becomes depleted, and if the crops are injured by diseases,
2.	Idea / Solution description	• If there is a sickness affecting the crop, the user is then presented with a workable cure. • The ability to anticipate agricultural yield based on location and the effective application of algorithms has demonstrated that a higher crop yield is possible.

3.	Novelty / Uniqueness	• Use a smart irrigation system on farms to increase production • Using image processing to detect crop illnesses so that users can order herbicides based on disease photos
4.	Social Impact / Customer Satisfaction	• Using cloud computing to provide complete irrigation data • Effective use of current knowledge via artificial intelligence
5.	Business Model (Revenue Model)	• The suggested method use SVM to categorise tree leaves, detect disease, and provide fertilizer • Support Vector Machines are effective at predicting crop yields.
6.	Scalability of the Solution	• Support Vector Machine Random Forest algorithm

3.4 PROBLEM SOLUTION FIT



4. REQUIRMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement	Sub Requirement
FR-1	User Registration	Registration through form Registration through Gmail Registration through LinkedIn

FR-2	Image Capture	Take image of a leaf Check the leaf is captured undergiven parameters
FR-3	Image Processing	Upload the leaf image Click the predict button
FR-4	Updated Native Language	Languages can be changed according to the user, which heis more understandable with. (Ex: English, Hindi, Tamil)
FR-5	Leaf Prediction	Add the pesticides and fertilizers to be used for an unhealthy leaf
FR-6	Image Description	Show the prescribed fertilizer and description of the diseasefor curing a unhealthy leaf
FR-7	Providing Datasets	Training datasets Testing datasets
FR-8	Adding Datasets	Fruit datasets for fruits Vegetable datasets for vegetables
FR-9	E-mail Notification	Farmers will be received a Email notification about the leaf and its history

4.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional Requirements:

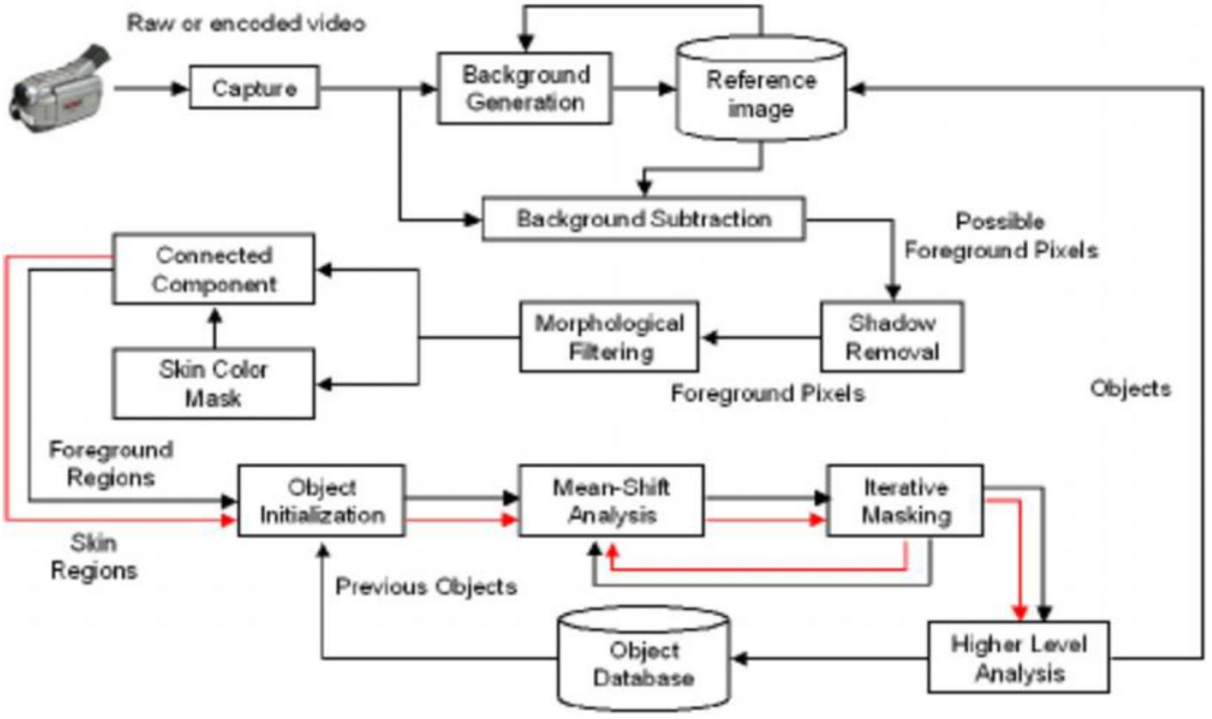
Following are the non-functional requirements of the proposed solution.

z

NFR No.	**Non-Functional Requirement**	**Description**
NFR-1	Usability	Leaf datasets can be used for detection of all kind of leafs Datasets can be reusable Datasets can be prepared according to the leaf
NFR-2	Security	User information and leaf data are secured The algorithms used are more secure
ZZNFR-3	Reliability	The leaf quality is more The datasets and image capturing performs consistently well
NFR-4	Performance	Leaf problem defines once the leafis detected Performs well according to the quality of leaf provides certain cure to it.
NFR-5	Availability	Quality of leaf will be used againfor detection Available and easy access of datasets provided
NFR-6	Scalability	Increase in growth of predicting the results and defining a leaf

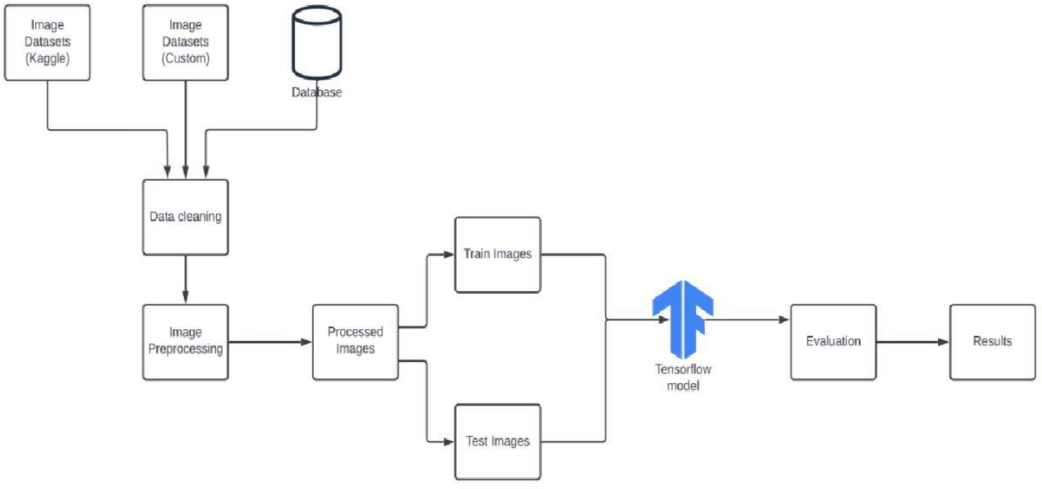
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURE

Plant crop disease is anticipated, and appropriate fertilizer is advised for a higher yield. The diseased plant photos are acquired and preprocessed in comparison to the dataset of diseased plants. The photos are processed using a Deep Learning algorithm, which is subsequently tested. A model is then created based on the evaluations, trained using a variety of inputs, and predictions are presented to the users, aiding in the fertilizer recommendation process. The inclusion of the Convolutional layers in the classification and processing of the images further aids in



5.3 USER STORIES

A digital camera or similar devices are used to take images of different types, and then those are used to identify the affected area in leaves. Then different types of image-processing techniques are applied to them, the process those images, to get different and useful features needed for the purpose of analyzing later-Plant leaf disease identification is especially needed to predict both the quality and quantity of the First segmentation step primarily based on a mild polygonal leaf model is first achieved and later used to guide the evolution of an energetic contour. Combining global shape descriptors given by the polygonal model with local curvature

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

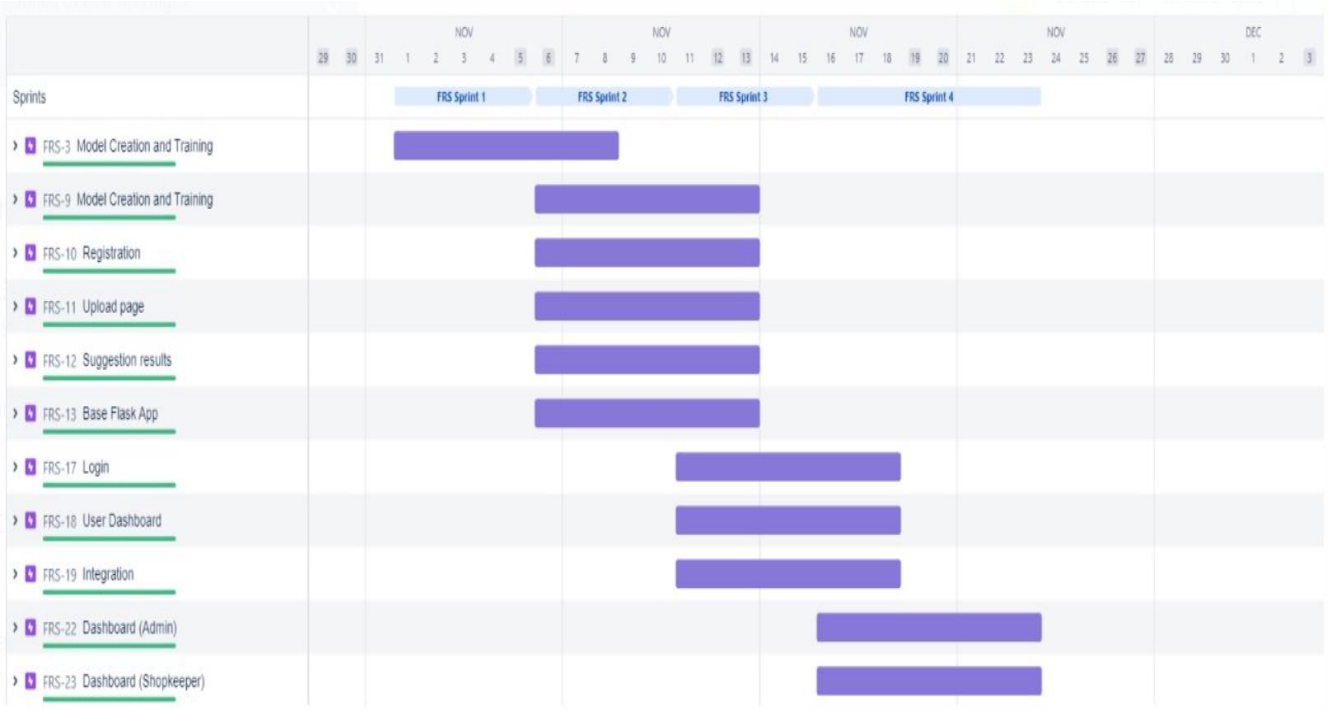
Sprint	Functional Requirement(Epic)	User Story Number	User Story / Task	Story Points (Total)	Priority	Team Members
Sprint-1	Model Creation and Training		Create a model which can classify diseased plant and train the model for accurate results	8	High	SRISURAJ
	Model Creation and Training		Create a model which can classify diseasedplants from the rest of the healthy plants	2	High	SRISURAJ

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points (Total)	Priority	Team Members
Sprint-2	Model Creation and Training		Create a model which can classify diseasedplants from the healthy plant	6	High	SIVAKUMAR SRINESH
	Registration	USN-1	As a user, I can register by entering my email, password, and confirming my password or viaOAuth API	3	Medium	SRINESH
	Upload page	USN-2	As a user, I will be redirected to a page where Ican upload my pictures of crops	4	High	SIVAKUMAR
	Suggestion results	USN-3	As a user, I can view the results and then obtainthe suggestions provided by the ML model	4	High	SRINESH SIVAKUMAR
	Base Flask App		A base Flask web app must be created as aninterface for the ML model	2	High	SIVAKUMAR SRINESH
Sprint-3	Login	USN-4	As a user/admin/shopkeeper, I can log into theapplication by entering email & password	2	High	SRISURAJ
	User Dashboard	USN-5	As a user, I can view the previous results andhistory	3	Medium	SUNDARESWAR
	Integration		Integrate Flask, CNN model with MySql DB	5	Medium	SRISURAJ SUNDARESWAR
Sprint-4	Dashboard (Admin)	USN-6	As an admin, I can view other user details anduploads for other purposes	2	Medium	SIVAKUMAR SRISURAJ
	Dashboard (Shopkeeper)	USN-7	As a shopkeeper, I can enter fertilizer productsand then update the details if any	2	Low	SIVAKUMAR SRISURAJ

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	10	5 Days	01 NOV 2022	05 NOV 2022	10	05 NOV 2022
Sprint-2	15	5 Days	06 NOV 2022	10 Nov 2022	15	10 Nov 2022
Sprint-3	15	4 Days	11 Nov 2022	14 Nov 2022	15	15 Nov 2022
Sprint-4	12	4 Days	16 Nov 2022	19 Nov 2022	10	20 Nov 2022

6.3 REPORTS FROM JIRA



7. CODING & SOLUTIONING

7.1 FEATURE 1

The application's registration page is created. User registration is carried out if the user hasn't already done so. Enough work was put into making this process seamless. If the user has registered, he can now log in directly. Email address, name, and password were required for registration. The code to link it to the backend was successful, and this data is stored in Firebase.

7.2 FEATURE 2

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

7.3 DATABASE SCHEMA

The Firebase platform was used. A mechanism for storing and retrieving data that is modelled in ways other than the tabular relations used in relational databases is provided by the Firebase database (NoSQL).

8. TESTING

8.1 TEST CASES

The test cases include invalid email and unrecognizable images. For the image part, a text file or other format files were uploaded as a corner case.

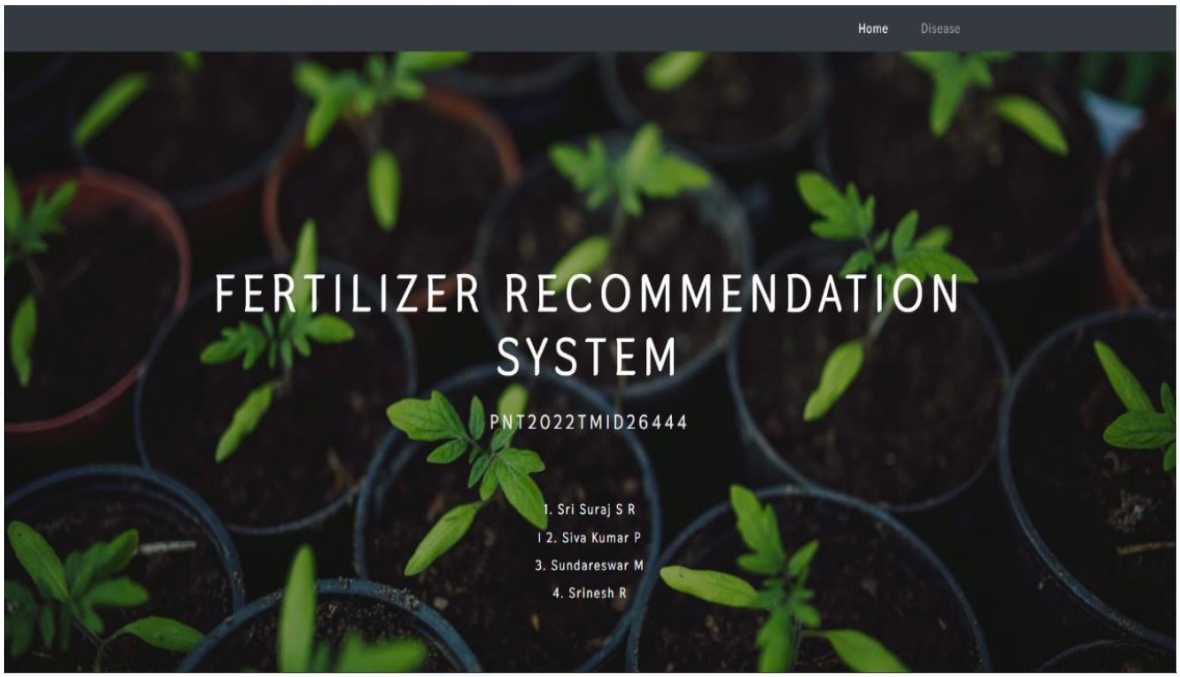
8.2 USER ACCEPTANCE TESTING

10 users of the test application were able to discover the nutritional data for the fruit image they supplied. This combined and tested both the registration and prediction modules, which showed to provide accurate results.

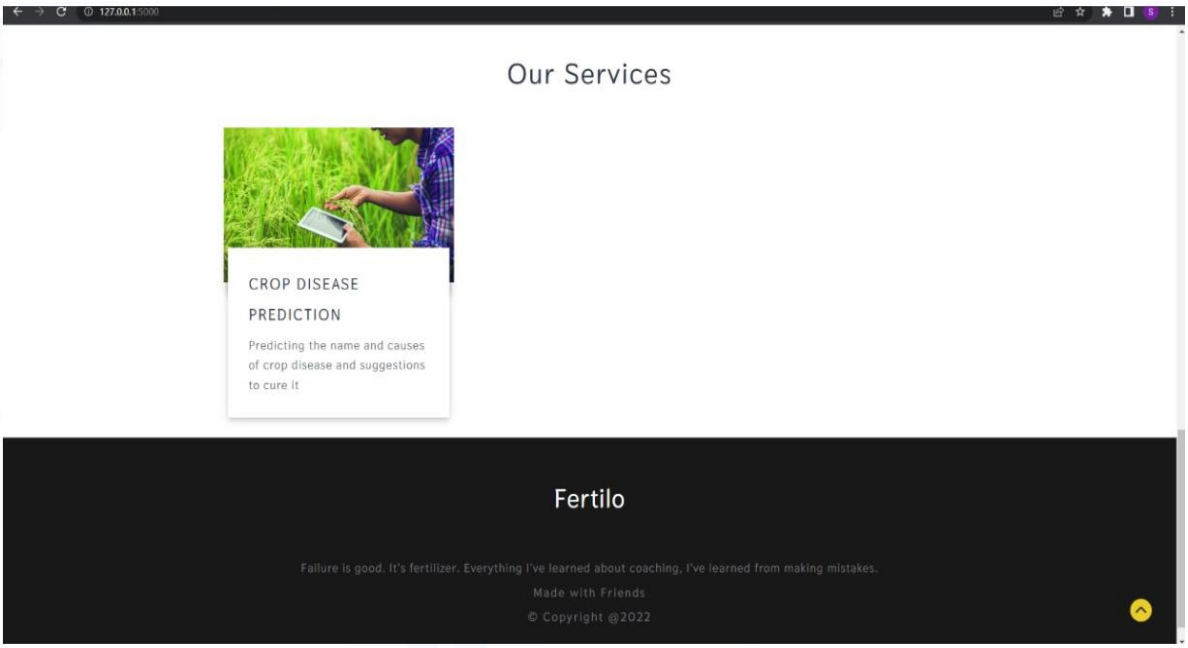
9. RESULTS

9.1 PERFORMANCE METRICS

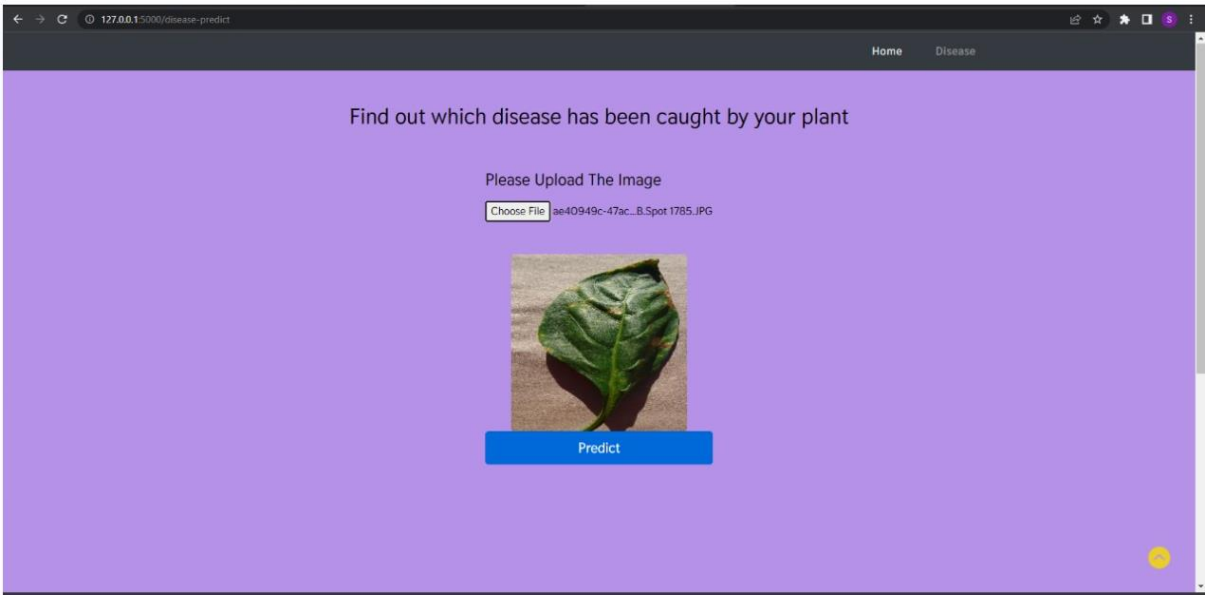
INTRODUCTION PAGE :-



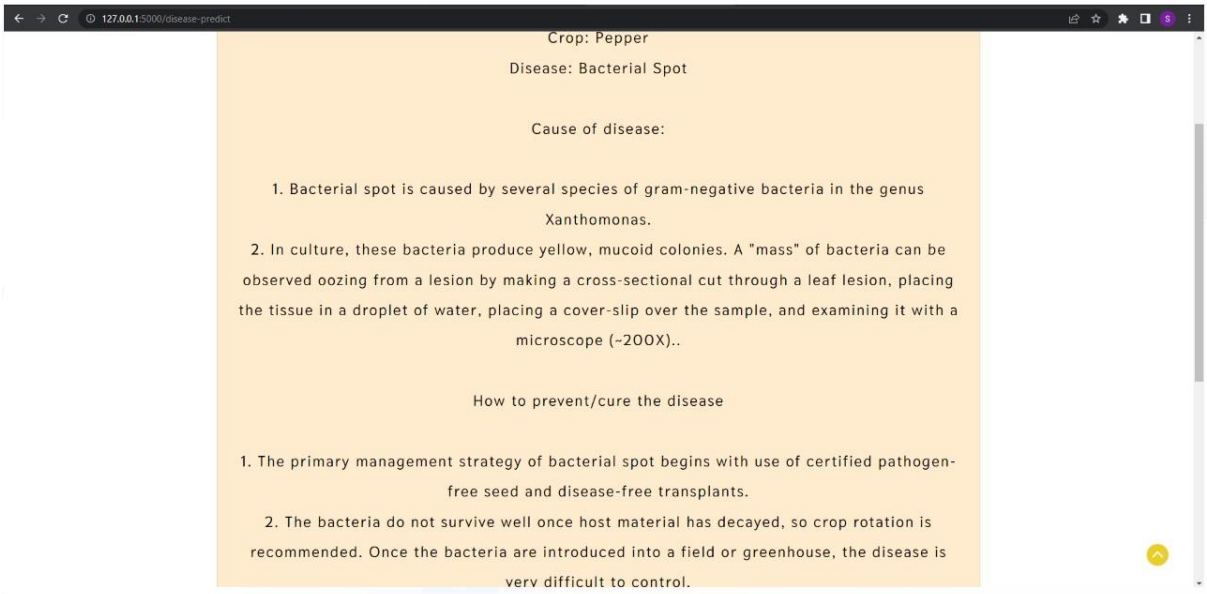
SELECTION BOX :-



UPLOADING IMAGE:-



RESULT DISPLAY:-



10. ADVANTAGES &DISADVANTAGES

ADVANTAGES:

- The suggested model yields extremely high classification accuracy
- It can train and test on very large datasets.
- It can resize very high-quality images within itself.

DISADVANTAGES:

- The proposed model is computationally expensive to train and test.
- The neural network architecture used in this project work is highly complex.

11. CONCLUSION

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

12. FUTURE SCOPE

The model that is being provided in this project work can be expanded to recognise images. Using python to exe software, the complete model may be turned into application software. With the aid of the OpenCV Python package, real-time image categorization, picture recognition, and video processing are all made feasible. This project's work can be expanded to include security applications including face, iris, and figure print recognition.

13. APPENDIX

13.1 SAMPLE SOURCE CODE:-

Index.html

```
<section class="banner_w3lspvt" id="home">
    <div class="csslider infinity" id="slider1">

                <div class="banner-top">
                    <div class="overlay">
                        <div class="container">
                            <div class="w3layouts-banner-info text-center">
                                <h3 class="text-wh">Fertilizer Recommendation
System</h3>
                                <h4 class="text-wh mx-auto my-
4"><b><strong>PNT2022TMID26444</strong></b></h4>
                                <br>
                                <h4 class="text-wh mx-auto my-4"><strong>
</strong></h4>
                                <p class="text-li mx-auto mt-2">
                                    1. Sri Suraj S R <br>I
                                    2. Siva Kumar P<br>
                                    3. Sundareswar M<br>
                                    4. Srinesh R</p>
                            </div>
                        </div>
                    </div>
                </div>
    </div>
</section>
<!-- //banner -->

<!-- core values -->
<section class="core-value py-5">
    <div class="container py-md-4">
        <h3 class="heading mb-sm-5 mb-4 text-center"> About Us</h3>
        <div class="row core-grids">
            <div class="col-lg-6 core-left">
                <img src="{{ url_for('static', filename='images/core.jpg') }}"
class="img-fluid" alt="" />
            </div>
            <div class="col-lg-6 core-right">
                <h3 class="mt-4">Improving Agriculture, Improving Lives,
Cultivating Crops To Make Farmers Increase
                    Profit.</h3>
                <p class="mt-3">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</p>
            </div>
        </div>
    </div>
</section>
<!-- //core values -->

<!-- Products & Services -->
<section class="blog py-5">
    <div class="container py-md-5">
        <h3 class="heading mb-sm-5 mb-4 text-center"> Our Services</h3>
        <div class="ro blog-grids">
            <div class="col-lg-4 col-md-6 blog-left mb-lg-0 mb-sm-5 pb-lg-0
pb-5">

                <a href="">

                    <div class="blog-info">

                        <h4></h4>

                        <p class="mt-2"> </p>
                    </div>
                </a>
            </div>




            <div class="col-lg-4 col-md-6 blog-middle mb-lg-0 mb-sm-5 pb-lg-0
pb-md-5">

                <a href="{{ url_for('fertilizer_recommendation') }}">
                    <div class="blog-info">
                        <h4></h4>

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

```
            <div class="col-lg-4 col-md-6 blog-right mt-lg-0 mt-5 pt-lg-0 pt-
md-5">

                <centre><img src="{{ url_for('static',
filename='images/s4.jpg') }}" class="img-fluid" alt=""></centre>

                <!-- <img src="images/s4.jpg" class="img-fliud" alt="" /> -->
                <a href="{{ url_for('disease_prediction') }}">
                    <div class="blog-info">
                        <h4>Crop Disease Prediction</h4>
                        <p class="mt-2">Predicting the name and causes of crop
disease and suggestions to cure it</p>
                    </div>
                </a>
            </div>
        </div>
    </div>
</section>
<!-- //Products & Services -->


<!-- Creating custom grid and hover effect
<section>
<div class="col-lg-3 col-md-4 col-sm-6 col-xs-12">
    <div class="hovereffect">
        <img class="img-responsive" src="images/s2.jpg" alt="">
        <div class="overlay">
           <h2>Hover effect 1</h2>
           <a class="info" href="#">link here</a>
        </div>
    </div>
</div> -->

</html>
```

Crop.html

```
<section class="banner_w3lspvt" id="home">
    <div class="csslider infinity" id="slider1">

                <div class="banner-top">
                    <div class="overlay">
                        <div class="container">
                            <div class="w3layouts-banner-info text-center">
                                <h3 class="text-wh">Fertilizer Recommendation
System</h3>
                                <h4 class="text-wh mx-auto my-
4"><b><strong>PNT2022TMID26444</strong></b></h4>
                                <br>
                                <h4 class="text-wh mx-auto my-4"><strong>
</strong></h4>
                                <p class="text-li mx-auto mt-2">
                                    1. Sri Suraj S R <br>I
                                    2. Siva Kumar P<br>
                                    3. Sundareswar M<br>
                                    4. Srinesh R</p>
                            </div>
                        </div>
                    </div>
                </div>
    </div>
</section>
<!-- //banner -->

<!-- core values -->
<section class="core-value py-5">
    <div class="container py-md-4">
        <h3 class="heading mb-sm-5 mb-4 text-center"> About Us</h3>
        <div class="row core-grids">
            <div class="col-lg-6 core-left">
                <img src="{{ url_for('static', filename='images/core.jpg') }}"
class="img-fluid" alt="" />
            </div>
            <div class="col-lg-6 core-right">
                <h3 class="mt-4">Improving Agriculture, Improving Lives,
Cultivating Crops To Make Farmers Increase
                    Profit.</h3>
                <p class="mt-3">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</p>
            </div>
        </div>
    </div>
</section>
<!-- //core values -->

<!-- Products & Services -->
<section class="blog py-5">
    <div class="container py-md-5">
        <h3 class="heading mb-sm-5 mb-4 text-center"> Our Services</h3>
        <div class="ro blog-grids">
            <div class="col-lg-4 col-md-6 blog-left mb-lg-0 mb-sm-5 pb-lg-0
pb-5">

                <a href="">

                    <div class="blog-info">

                        <h4></h4>

                        <p class="mt-2"> </p>
                    </div>
                </a>
            </div>




            <div class="col-lg-4 col-md-6 blog-middle mb-lg-0 mb-sm-5 pb-lg-0
pb-md-5">

                <a href="{{ url_for('fertilizer_recommendation') }}">
                    <div class="blog-info">
                        <h4></h4>

                    </div>
                </a>
            </div>
            <div class="col-lg-4 col-md-6 blog-right mt-lg-0 mt-5 pt-lg-0 pt-
md-5">
```

```
                <centre><img src="{{ url_for('static',
filename='images/s4.jpg') }}" class="img-fluid" alt=""></centre>

                <!-- <img src="images/s4.jpg" class="img-fliud" alt="" /> -->
                <a href="{{ url_for('disease_prediction') }}">
                    <div class="blog-info">
                        <h4>Crop Disease Prediction</h4>
                        <p class="mt-2">Predicting the name and causes of crop
disease and suggestions to cure it</p>
                    </div>
                </a>
            </div>
        </div>
    </div>
</section>
<!-- //Products & Services -->


<!-- Creating custom grid and hover effect
<section>
<div class="col-lg-3 col-md-4 col-sm-6 col-xs-12">
    <div class="hovereffect">
        <img class="img-responsive" src="images/s2.jpg" alt="">
        <div class="overlay">
           <h2>Hover effect 1</h2>
           <a class="info" href="#">link here</a>
        </div>
    </div>
</div> -->

</html>
```

Crop-result.html

```
<style>
  .mt-0 {
  margin-top: 50 !important;
}
</style>


<div class="container py-2 mx-auto my-50 h-10 " style="margin: 12rem;">
  <div class="row">
      <div class="col-sm py-2 py-md-3">
          <div class="card card-body" style="justify-content: center;">
               <h1 class="text-center"><b>You should grow <i>{{ prediction }}
</i>in your farm</b></h1>
          </div>
      </div>
  </div>
</div>
```

Disease.html

```
<style>
  html body {
    background-color: rgb(182, 145, 232);
  }
</style>
<br /><br />
  <h2 style="text-align: center; margin: 0px; color: black">
    <b>Find out which disease has been caught by your plant</b>
  </h2>
  <br />
  <br>

<div style="
    width: 350px;
    height: 50rem;
    margin: 0px auto;
    color: black;
    border-radius: 25px;
    padding: 10px 10px;
    font-weight: bold;
  ">




  <form class="form-signin" method=post enctype=multipart/form-data>
```

```
    <h2 class="h4 mb-3 font-weight-normal"><b>Please Upload The Image</b></h2>
    <input type="file" name="file" class="form-control-file" id="inputfile"
onchange="preview_image(event)" style="font-weight: bold;">
    <br>
    <br>
    <img id="output-image" class="rounded mx-auto d-block" />
    <button class="btn btn-lg btn-primary btn-block" type="submit"
style="font-weight: bold;">Predict</button>


  </form>
</div>

<script type="text/javascript">
  function preview_image(event) {
    var reader = new FileReader();
    reader.onload = function () {
      var output = document.getElementById('output-image')
      output.src = reader.result;
    }
    reader.readAsDataURL(event.target.files[0]);
  }
</script>

</div>
```

Disease-result.html

```
<div class="container py-2 mx-auto my-50 h-10 " style="margin: 9rem;">
  <div class="row">
      <div class="col-sm py-2 py-md-3">
          <div class="card card-body" style="justify-content: center;
background-color:blanchedalmond">
            <p class="text-center" style="color: black; font-size: 22px;">{{
prediction }}</p>
          </div>
      </div>
  </div>
</div>
```

Fertilizer.html

```
<style>
  html body {
    background-color: rgb(206, 206, 228);
  }
</style>
<!--Form Section-->
<br /><br />
<h2 style="text-align: center; margin: 0px; color: black">
  <b>Get informed advice on fertilizer based on soil</b>
</h2>
<br />

<div
  style="
    width: 350px;
    height: 40rem;
    margin: 0px auto;
    color: black;
    border-radius: 25px;
    padding: 10px 10px;
  "
>
  <form method="POST" action="{{ url_for('fert_recommend') }}">
    <div class="form-group">
      <label for="Nitrogen" style="font-size: 17px"><b>Nitrogen</b></label>
      <input
        type="number"
        class="form-control"
        id="Nitrogen"
        name="nitrogen"
        placeholder="Enter the value (example:50)"
        style="font-weight: bold"
        required
      />
    </div>
    <div class="form-group">
      <label for="Phosphorous" style="font-size: 17px"
        ><b>Phosphorous</b></label
      >
      <input
        type="number"
        class="form-control"
        id="Phosphorous"
        name="phosphorous"
        placeholder="Enter the value (example:50)"
        style="font-weight: bold"
        required
```

```
    />
  </div>

  <div class="form-group">
    <label for="Pottasium" style="font-size: 17px"><b>Pottasium</b></label>
    <input
      type="number"
      class="form-control"
      id="Pottasium"
      name="pottasium"
      placeholder="Enter the value (example:50)"
      style="font-weight: bold"
      required
    />
  </div>
  <div class="form-group">
    <label for="crop" style="font-size: 17px"
      ><b>Crop you want to grow</b></label
    >
    <select
      name="cropname"
      class="form-control"
      id="crop"
      placeholder="Select a crop"
      style="font-weight: bold"
      required
    >
      <option selected>Select crop</option>
      <option>rice</option>
      <option>maize</option>
      <option>chickpea</option>
      <option>kidneybeans</option>
      <option>pigeonpeas</option>
      <option>mothbeans</option>
      <option>mungbean</option>
      <option>blackgram</option>
      <option>lentil</option>
      <option>pomegranate</option>
      <option>banana</option>
      <option>mango</option>
      <option>grapes</option>
      <option>watermelon</option>
      <option>muskmelon</option>
      <option>apple</option>
      <option>orange</option>
      <option>papaya</option>
      <option>coconut</option>
      <option>cotton</option>
```

```
        <option>jute</option>
        <option>coffee</option>
      </select>
    </div>

    <div class="d-flex justify-content-center">
      <button
        type="submit"
        class="btn btn-info"
        style="
          color: black;
          font-weight: bold;
          width: 130px;
          height: 50px;
          border-radius: 12px;
          font-size: 21px;
        "
      >
        Predict
      </button>
    </div>
  </form>
</div>
```

Fertilizer-result.html

```
<div class="container py-2 mx-auto my-50 h-10 " style="margin: 9rem;">
  <div class="row">
      <div class="col-sm py-2 py-md-3">
          <div class="card card-body" style="justify-content: center;
background-color:blanchedalmond">
              <p class="text-center" style="color: black; font-size:
20px;">{{ recommendation }}</p>
          </div>
      </div>
  </div>
</div>
```

Layout.html

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

<head>
    <title>{{ title }}</title>
    <link rel="shortcut icon" href="{{ url_for('static',
filename='images/favicon.ico') }}"/>

    <!-- for-mobile-apps -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta charset="utf-8">
    <meta name="keywords" content="Agro Harvest Responsive web template,
Bootstrap Web Templates, Flat Web Templates, Android Compatible web template,
Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyEricsson, Motorola web design" />

    <style>
        html {
            font-size: 1rem;
        }

        @media (min-width: 576px) {
            html {
                font-size: 1.25rem;
            }
        }

        @media (min-width: 768px) {
            html {
                font-size: 1.5rem;
            }
        }

        @media (min-width: 992px) {
            html {
                font-size: 1.75rem;
            }
        }

        @media (min-width: 1200px) {
            html {
                font-size: 2rem;
            }

            html {
                font-size: 1rem;
            }
```

```
h1 {
    font-size: 1.2rem;
}

h2 {
    font-size: 1.1rem;
}

@media (min-width: 768px) {
    html {
        font-size: 1.1rem;
    }

    h1 {
        font-size: 1.3rem;
    }

    h2 {
        font-size: 1.2rem;
    }
}

@media (min-width: 991px) {
    html {
        font-size: 1.2rem;
    }

    h1 {
        font-size: 1.5rem;
    }

    h2 {
        font-size: 1.4rem;
    }
}

@media (min-width: 1200px) {
    html {
        font-size: 1.2rem;
    }

    h1 {
        font-size: 1.7rem;
    }

    h2 {
        font-size: 1.6rem;
```

```
                }
            }

        }
    </style>
    <script>
        addEventListener("load", function () {
            setTimeout(hideURLbar, 0);
        }, false);

        function hideURLbar() {
            window.scrollTo(0, 1);
        }

    </script>

    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
        integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
        crossorigin="anonymous"></script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js
"
        integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
        crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
        integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
        crossorigin="anonymous"></script>
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
        integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
        crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
        integrity="sha384-
Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
        crossorigin="anonymous"></script>
    </body>
    <!-- css files -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css
"
        integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
```

```
    <link href="{{ url_for('static', filename='css/bootstrap.css') }}"
rel='stylesheet' type='text/css' />
    <!-- bootstrap css -->
    <link href="{{ url_for('static', filename='css/style.css') }}"
rel='stylesheet' type='text/css' />
    <!-- custom css -->
    <link href="{{ url_for('static', filename='css/font-awesome.min.css') }}"
rel="stylesheet"><!-- fontawesome css -->
    <!-- //css files -->
    <!-- <link rel="icon" type="image/png" href="{{ url_for('static',
filename='images/favicon.png?') }}"> -->


    <script type="text/JavaScript" src="{{ url_for('static',
filename='scripts/cities.js') }}"></script>


    <!-- google fonts -->
    <link
href="//fonts.googleapis.com/css?family=Thasadith:400,400i,700,700i&amp;subset
=latin-ext,thai,vietnamese"
        rel="stylesheet">
    <!-- //google fonts -->

    <style>
        header {
            background-color: rgba(30, 30, 30, 1);
            margin-top: 0rem;
            display: block;

        }
    </style>
</head>

<body>

    <!-- Navigation -->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark static-top"
style="background-color: #1C00ff00;">
        <div class="container">
            <a class="navbar-brand" href="{{ url_for('home') }}">

            </a>
            <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarResponsive"
                aria-controls="navbarResponsive" aria-expanded="false" aria-
label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
```

```
            <div class="collapse navbar-collapse" id="navbarResponsive">
                <ul class="navbar-nav ml-auto">
                    <li class="nav-item active">
                        <a class="nav-link" href="{{ url_for('home') }}">Home
                            <span class="sr-only">(current)</span>
                        </a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="{{ url_for('crop_recommend')
}}"></a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="{{
url_for('fertilizer_recommendation') }}"></a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="{{
url_for('disease_prediction') }}">Disease</a>
                    </li>
                </ul>
            </div>
        </div>
    </nav>



    {% block body %} {% endblock %}

    <!-- footer -->
    <footer class="text-center py-5">
        <div class="container py-md-3">
            <!-- logo -->
            <h2 class="logo2 text-center">
                <a href="{{ url_for('home') }}">
                    Fertilo
                </a>
            </h2>
            <!-- //logo -->
            <!-- address -->
            <div class="contact-left-footer mt-4">




                <!-- <a href="community.html">Community</a> -->
                </p>
            </div>
```

```
            <div class="w3l-copy text-center">
                <p class="text-da"><br>Failure is good. It's fertilizer.
Everything I've learned about coaching, I've learned from making mistakes.
</p>
            </div>
            <p class="homelogo">
            <p>Made with Friends</p>
            <p>&copy; Copyright  @2022 </p>

        </div>
    </footer>
    <!-- //footer -->

    <!-- move top icon -->
    <a href="#home" class="move-top text-center"></a>
    <!-- //move top icon -->
</body>

</html>
```

Try again.html

```
<div class="container py-2 mx-auto my-50 h-10 text-center" style="margin:
9rem;">
  <div class="row">
    <div class="col-sm py-2 py-md-3">
      <div class="card card-body" style="justify-content: center; background-
color:blanchedalmond">
        <h1 class="text-center" style="color: black; font-size:
20px;"><b>Sorry we couldn't process your request
            currently. <br> Please try again</b></h1>


        <a href="{{ url_for('home') }}">
          <button type="submit" class="btn btn-info text-center" style="
        color: black;
        font-weight: bold;
        margin: 1rem;">
            Try again
          </button>
        </a>
      </div>
    </div>
  </div>
</div>
```

Fertilizer.csv

,Crop,N,P,K,pH,soil_moisture

0,rice,80,40,40,5.5,30

3,maize,80,40,20,5.5,50

5,chickpea,40,60,80,5.5,60

12,kidneybeans,20,60,20,5.5,45

13,pigeonpeas,20,60,20,5.5,45

14,mothbeans,20,40,20,5.5,30

15,mungbean,20,40,20,5.5,80

18,blackgram,40,60,20,5,60

24,lentil,20,60,20,5.5,90

60,pomegranate,20,10,40,5.5,30

61,banana,100,75,50,6.5,40

62,mango,20,20,30,5,15

63,grapes,20,125,200,4,60

66,watermelon,100,10,50,5.5,70

67,muskmelon,100,10,50,5.5,30

69,apple,20,125,200,6.5,50

74,orange,20,10,10,4,60

75,papaya,50,50,50,6,20

88,coconut,20,10,30,5,45

93,cotton,120,40,20,5.5,70

94,jute,80,40,40,5.5,20

95,coffee,100,20,30,5.5,20

Model.py

```
import torch
import torch.nn as nn
import torch.nn.functional as F


def ConvBlock(in_channels, out_channels, pool=False):
    layers = [nn.Conv2d(in_channels, out_channels, kernel_size=3, padding=1),
              nn.BatchNorm2d(out_channels),
              nn.ReLU(inplace=True)]
    if pool:
        layers.append(nn.MaxPool2d(4))
    return nn.Sequential(*layers)


# Model Architecture
class ResNet9(nn.Module):
    def __init__(self, in_channels, num_diseases):
        super().__init__()

        self.conv1 = ConvBlock(in_channels, 64)
        self.conv2 = ConvBlock(64, 128, pool=True) # out_dim : 128 x 64 x 64
        self.res1 = nn.Sequential(ConvBlock(128, 128), ConvBlock(128, 128))

        self.conv3 = ConvBlock(128, 256, pool=True) # out_dim : 256 x 16 x 16
        self.conv4 = ConvBlock(256, 512, pool=True) # out_dim : 512 x 4 x 44
        self.res2 = nn.Sequential(ConvBlock(512, 512), ConvBlock(512, 512))

        self.classifier = nn.Sequential(nn.MaxPool2d(4),
                                        nn.Flatten(),
                                        nn.Linear(512, num_diseases))

    def forward(self, xb): # xb is the loaded batch
        out = self.conv1(xb)
        out = self.conv2(out)
        out = self.res1(out) + out
        out = self.conv3(out)
        out = self.conv4(out)
        out = self.res2(out) + out
        out = self.classifier(out)
        return out
```

REQUIREMENTS

argon2-cffi==20.1.0

astroid==2.4.2

async-generator==1.10

attrs==20.3.0

backcall==0.2.0

bleach==3.2.1

certifi==2020.12.5

cffi==1.14.4

chardet==3.0.4

click==7.1.2

cycler==0.10.0

dataclasses==0.6

decorator==4.4.2

defusedxml==0.6.0

docopt==0.6.2

entrypoints==0.3

Flask==1.1.2

future==0.18.2

gunicorn==20.0.4

idna==2.10

ipykernel==5.3.4

ipython==7.19.0

ipython-genutils==0.2.0

ipywidgets==7.5.1

isort==5.6.4

itsdangerous==1.1.0

jedi==0.17.2

Jinja2==2.11.2

joblib==0.17.0

jsonschema==3.2.0

jupyter==1.0.0

jupyter-client==6.1.7

jupyter-console==6.2.0

jupyter-core==4.7.0

jupyterlab-pygments==0.1.2

kiwisolver==1.3.1

lab==6.2

lazy-object-proxy==1.4.3

MarkupSafe==1.1.1

matplotlib==3.3.3

mccabe==0.6.1

mistune==0.8.4

nbclient==0.5.1

nbconvert==6.0.7

nbformat==5.0.8

nest-asyncio==1.4.3

notebook==6.1.5

numpy==1.19.4

packaging==20.4

pandas==1.1.4

pandocfilters==1.4.3

parso==0.7.1

pexpect==4.8.0

pickleshare==0.7.5

Pillow==8.0.1

pipreqs==0.4.10

prometheus-client==0.9.0

prompt-toolkit==3.0.8

ptyprocess==0.6.0

pycodestyle==2.6.0

pycparser==2.20

pyflakes==2.2.0

Pygments==2.7.2

pylint==2.6.0

pyparsing==2.4.7

pyrsistent==0.17.3

python-dateutil==2.8.1

pytz==2020.4

pyzmq==20.0.0

qtconsole==5.0.1

QtPy==1.9.0

requests==2.25.0

scipy==1.5.4

toml==0.10.2

torch==1.7.0+cpu

torchaudio==0.7.0

torchvision==0.8.1+cpu

tornado==6.1

traitlets==5.0.5

txt2tags==3.7

typing-extensions==3.7.4.3

urllib3==1.26.2

wcwidth==0.2.5

webencodings==0.5.1

Werkzeug==1.0.1

widgetsnbextension==3.5.1

wrapt==1.12.1

xgboost==1.2.1

yarg==0.1.9

13.2 GITHUB

https://github.com/IBM-EPBL/IBM-Project-22504-1659853298

13.3 PROJECT DEMO LINK

video link-https://youtu.be/BL7W1mFJfSw