KAMARAJ COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

PROJECT REPORT

FERTILIZER RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

COURSE CODE

HX8001

TEAM ID : PNT2022TMID12393

SUBMITTED BY

1. **PANDI RAJA P - 920419205029 (TEAM LEADER)**
2. **VENKATESH PRABHU R - 920419205046 (TEAM MEMBER)**
3. **SANTHOSH G - 920419205037 (TEAM MEMBER)**
4. **DHANASEKARA PANDIAN G - 920419205009 (TEAM MEMBER)**
5. **ANANTHAN C P - 920419205003 (TEAM MEMBER)**

INDEX

Ex.No	Title	Page no.
1	INTRODUCTION	
1.1	Project Overview	4
1.2	Purpose	4
2	LITERATURE SURVEY	
2.1	Existing problem	5
2.2	References	7
2.3	Problem Statement Definition	7
3	**IDEATION AND PROPOSED SOLUTION**	
3.1	Empathy map and Canvas	8
3.2	Ideation & Brainstorming	9
3.3	Proposed Solution	12
3.4	Problem Solution fit	13
4	**REQUIREMENT ANALYSIS**	
4.1	Functional requirement	14
4.2	Non-Functional requirement	14
5	**PROJECT DESIGN**	
5.1	Data Flow Diagram	15
5.2	Solution & Technical Architecture	16
5.3	User Stories	17
6	**PROJECT PLANNING & SCHEDULING**	
6.1	Sprint Planning & Estimation	19
6.2	Sprint Delivery Schedule	21
6.3	Reports from JIRA	22
7	**CODING AND SOLUTIONING**	
7.1	Feature 1	25
7.2	Feature 2	38
8	**TESTING**	
8.1	Test Cases	45
8.2	User Acceptance Testing	45
9	**RESULTS**	
9.1	Performance Metrics	47
10	**ADVANTAGES & DISADVANTAGES**	49

11	**CONCLUSION**	50
12	**FUTURE SCOPE**	51
13	**APPENDIX**	
13.1	Source Code	52
13.2	GitHub & Project Demo Link	74

1. INTRODUCTION

1.1 Project Overview

Plant disease prediction helps in the detection and recognition of the plant diseases. The images of plants are captured and analyzed for certain symptoms using Computer vision and image processing. By identifying the disease, the deficit nutrients that lead to the disease are found. Based on the available data on fertilizers, the necessary nutrient rich fertilizers are recommended.

1.2 Purpose

The plant diseases may lead to abnormal functionalities which may end up with the death of the plant.The project aims at recognizing the symptoms at the early stages. The project also aims at guiding the farmers with the proper choice of the fertilizers that are required to counter the deficiency of the nutrients that cause the disease.

2. LITERATURE SURVEY

2.1 Existing problem

S. No	**Title**	**Technique**	**Links**
1	Soil Based Fertilizer Recommendation System for Crop Disease Prediction System– P.Pandi Selvi,P.Poornima	Long or Short Term Memory Algorithm	http://www.ijetajournal.org/vloume-8/issue-2/IJETA-V812P1
2	IOT based Crop Recommendation,Crop Disease Prediction and Its Solution – Rani Holambe, Pooja Patil, Padmaja PawarHrushikesh Joshi,Saurabh Salunkhe		https://arxiv.org/pdf/2204.11340
3	Farmer's Assistant:A Machine Learning Based Application for Agricultural Solutions-Shloka Gupta,Aparna Bhonde,Akshay Chopade,Nishit Jain	Image Analysis,Deep Learning,Machine Learning	https://www.irjet.net/archives/V7/i10/IRJET-V7l1004
4	R. Neela,P. Fertilizers Recommendation System ForDisease Prediction In Tree Leave International journal of scientific & technology research volume 8, issue 11, November 2019	Adding a CNN(Convolutional neural network) and SVM(Support VectorMachine)	http://www.ijstr.org/final-print/nov2019/

5	Plant Disease Detection Using Image Processing and Machine Learning	Random Forestclassifier, a combination of multiple decision trees is used where each tree is trained by using different subsets of the whole dataset to reduce the over fitting and improves the accuracy of the classifier.	https://arxiv.org/abs/2106.10698
6	Fertilizers Recommendation System for Disease Prediction in TreeLeaves	Support Vector Machine (SVM) algorithm classifies the leaf image asnormal or affected. And it is used to identify a function Fx which obtain the hyper-plane.	https://www.semanticscholar.org/paper/Fertilizers-Recommendatio Disease-In-Neela-Nithya/495379d3ef2b461fabd2de8d0605c16

2.2 References

1. R. Neela, P. Fertilizers Recommendation System For DiseasePrediction In Tree Leave International journal of scientific & technology research volume 8, issue 11, november2019 http://www.ijstr.org/final-print/nov2019/Fertilizers-Recommendation-System-For-Disease-Prediction In-Tree-Leave.pdf .

2. Swapnil Jori1, Rutuja Bhalshankar2, Dipali Dhamale3, Sulochana Sonkamble , Healthy Farm: Leaf Disease Estimation and Fertilizer Recommendation System using Machine Learning,International Journalof All Research Education and Scientific Methods(IJARESM), ISSN: 2455-6211

3. Detection of Leaf Diseases and Classification using Digital Image Processing International Conference on Innovations in Information, Embedded and Communication Systems(ICIIECS), IEEE, 2017.

4. Shloka Gupta ,Nishit Jain ,Akshay Chopade, Farmer's Assistant: A MachineLearning BasedApplication for Agricultural Solution

2.3 Problem Statement Definition

Identify the diseases in plants and suggest required fertilizer to prevent disease and produce good crops. This project aims at providing a system to support the cultivators in choosing the right fertilizers for their plants to counter the deficiency of nutrients that cause various infections and diseases. The below blocks define the problems faced by the different users and the solutions that are provided by the system.

3. IDEATION & PROPOSED SOLUTION

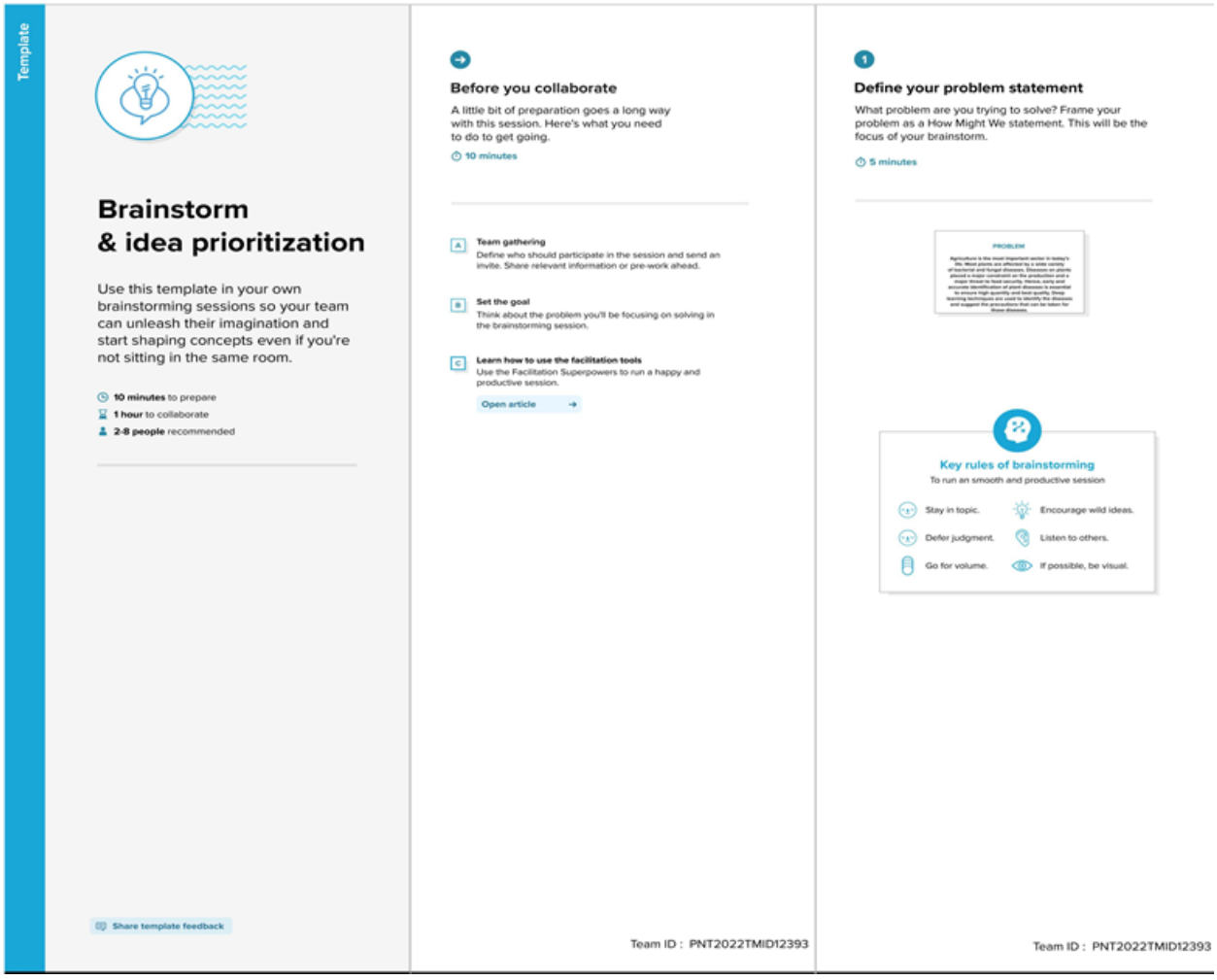
3.1 Empathy Map Canvas

Agriculture is the main aspect of the economic development of a country. Agriculture is the heart and life of most Indians.By understanding their feelings and problems, we can create a better product and contribute to their lives. For our project, we are getting surveys from farmers to understand what they truly require and desire.

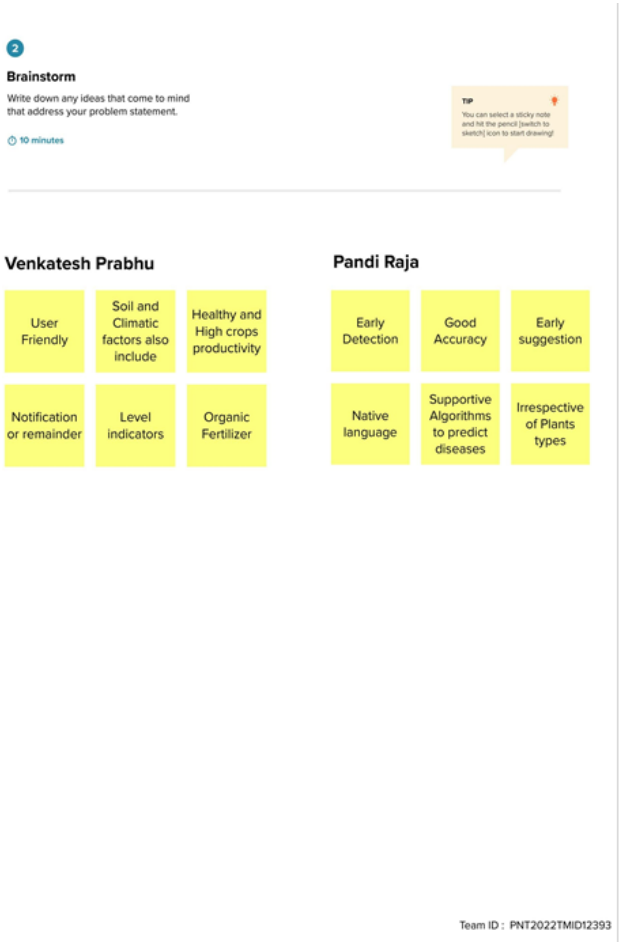
3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a richamount of creative solutions.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

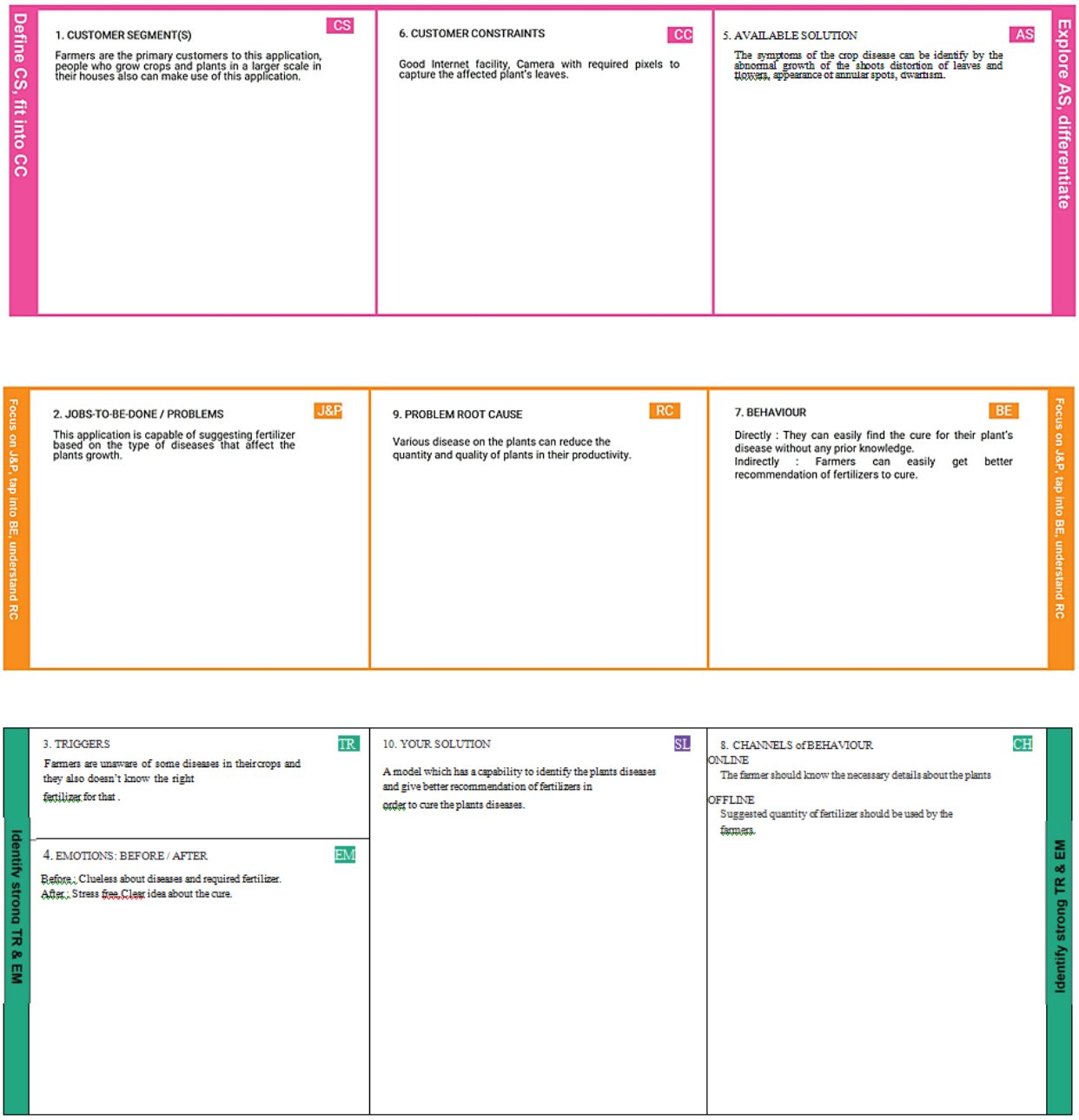
20 minutes

Team ID : PNT2022TMID12393

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	Identify the diseases in plants and suggestrequired fertilizer to prevent disease and produce good crops
2.	Idea / Solution description	Early identification of diseases by using the datasets having various type of diseases andrecommended fertilizer to prevent.
3.	Novelty / Uniqueness	Not only suggestion but also it notify thetime,quantity of fertilizer to be added.
4.	Social Impact/ Customer Satisfaction	This early prediction will reduce soil erosionand the growth of unhealthy plants.It also increase the productivity.
5.	Business Model (Revenue Model)	The application will recommend to the farmersin subscription basis.
6.	Scalability of the Solution	Our Future enhancement is to introduce an e-commerce facility in this application which help Farmersto sell their crops online withoutany mediator.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1. Functional requirement

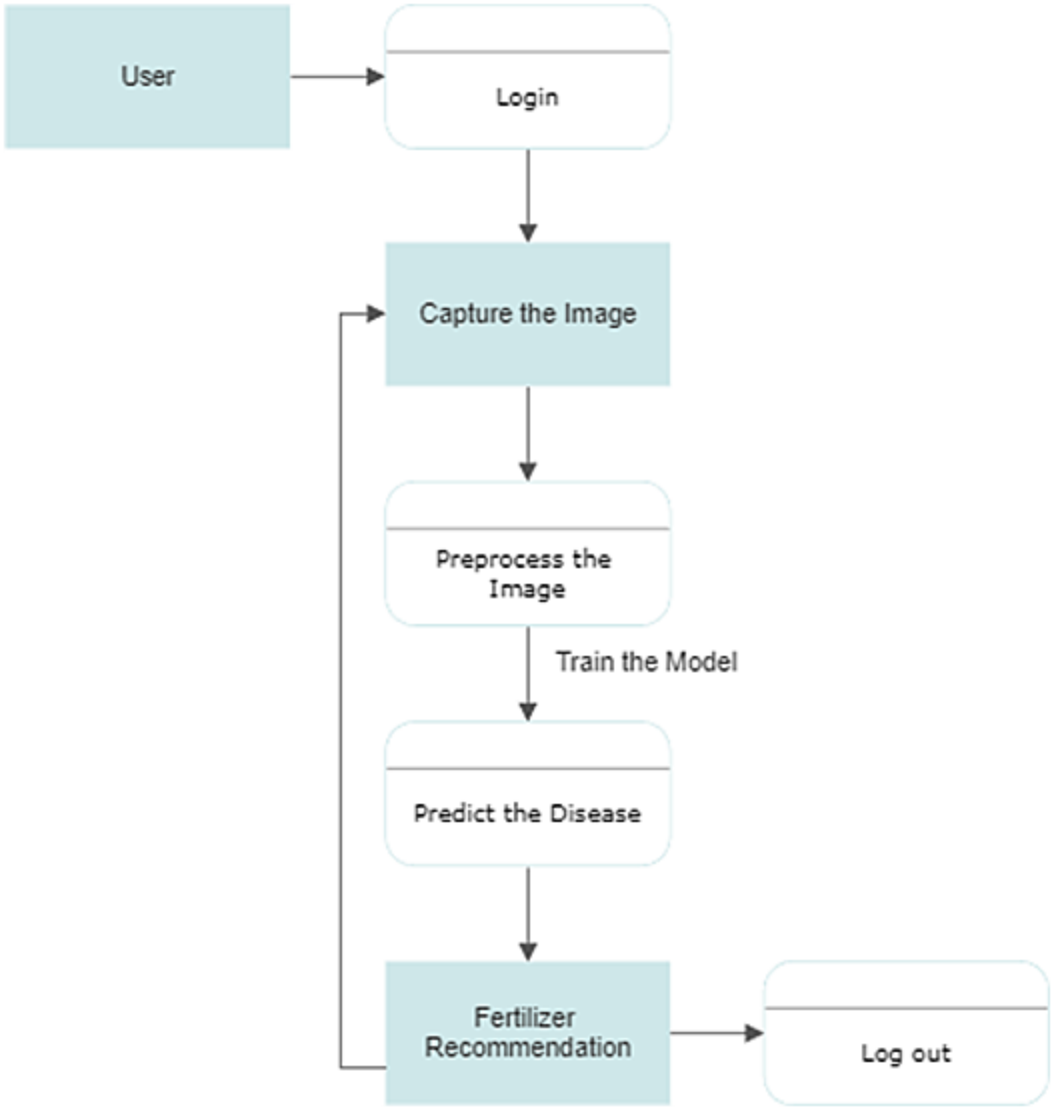
FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	User Registration	Registration through Form Registration throughGmail
FR-2	User Confirmation	Confirmation via EmailConfirmation via TFA
FR-3	Image Capturing	Capture the leaves image and check for affected areas
FR-4	Image Processing	Process the image to predict the disease using variousAlgorithms.
FR-5	Disease Prediction	Identify the diseases based on the captured images.
FR-6	Suggest fertilizer for the plants	Suggest fertilizer based on the predicted disease toincrease the Productivity

4.2 Non-Functional requirements

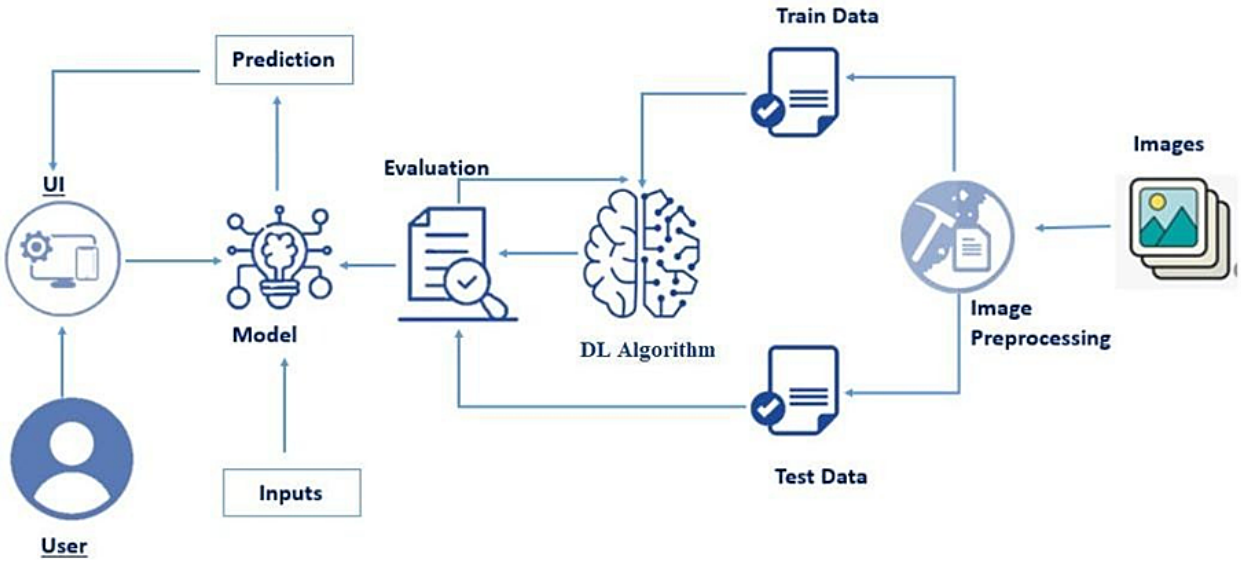
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Datasets of all the leaf is used to detecting thedisease that present in the leaf.
NFR-2	**Security**	The information belongs to the user and leaf aresecured highly.
NFR-3	**Reliability**	The leaf quality is important for the predictingthe disease in leaf.
NFR-4	**Performance**	The performance is based on the quality of theleaf used for disease prediction
NFR-5	**Availability**	It is available for all user to predict the disease in the plant
NFR-6	**Scalability**	Increasing the prediction of the disease in theleaf

5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



5.3 User Stories

UserType	Functional Requirement (Epic)	User Story Number	UserStory / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application byentering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email onceI have registered for the application	I can receive confirmation email& click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Gmail	I can use my Gmail account to access the application	Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email &password	I can make use of the application to predict the plant's diseases	High	Sprint-1
Customer (Web user)	Registration	USN-5	As a web userI can sign up on the system with the customer ID	I can access the application like Website	High	Sprint-1
Customer Care Executive	Customer Support	USN-6	As a supporter, I can see how user friendly theapplication is to the customers.	I can develop customer guidelines and practices	Medium	Sprint-2

Administrator	Analyst	USN-7	As an admin, I can make the application up to date(ie. BothDiseases and Fertilizers).	I can manage the large dataset efficiently	High	Sprint-1
Customer Purpose	Prediction	USN-8	It uses a Machine Learning algorithm to identify the plant's disease with the help of captured images and suggests the requiredfertilizer.	I can predict the plants Disease	High	Sprint-2

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	FunctionalRequirement (Epic)	User Story Number	User Story / Task	Story Points (Total)	Priority	Team Members
Sprint 2	Registration	USN-1	As a user,I can register by entering my email, password, and confirming my password or viaOAuth API	3	Medium	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Upload page	USN-2	As a user, I will be redirected to a page where I can upload my pictures of crops	4	High	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Suggestion results	USN-3	As a user, I can view the results and then obtain the suggestions provided by the ML model	4	High	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Base Flask App		A base Flask web app must be created as an interface for the ML model	2	High	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan

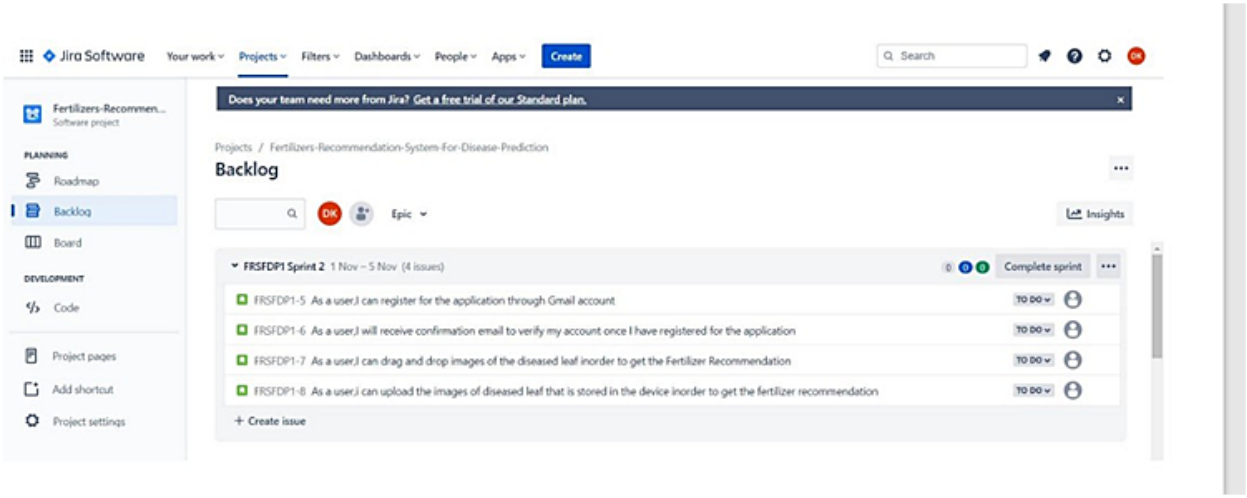
Sprint-3	Login	USN-4	As a user/admin/shopkeeper, I can log into the application by entering email & password	2	High	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	User Dashboard	USN-5	As a user, I can view the previous results andhistory	3	Medium	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Integration		Integrate Flask,CNN model withCloudant DB	5	Medium	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Containerization		Containerize Flaskapp using Docker	2	Low	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
Sprint-4	Dashboard (Admin)	USN-6	As an admin, I can view other user details and uploads for other purposes	2	Medium	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan

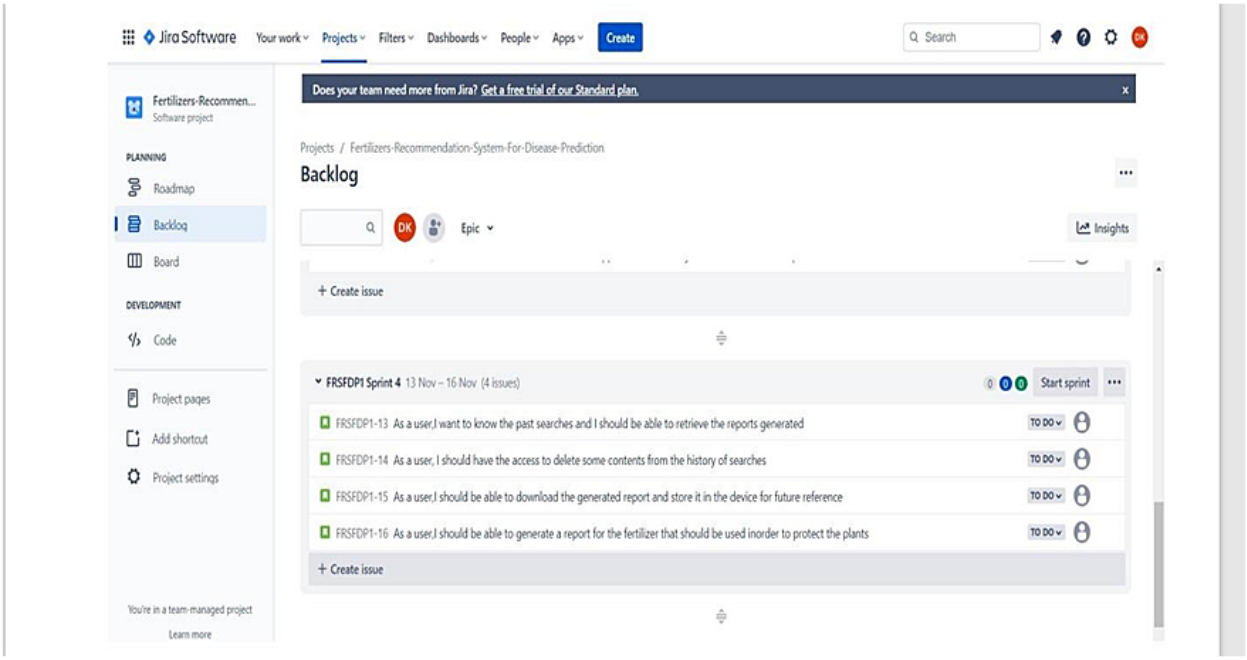
	Dashboard (Shopkeeper)	USN-7	As a shopkeeper, I can Enterfertilizer products and then update the details if any	2	Low	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan
	Containerization		Create and deploy Helm charts using DockerImage made before	2	Low	R. Venkatesh Prabhu, G. Santhosh, P. Pandi Raja, G. Dhanasekara Pandian, C. P. Ananthan

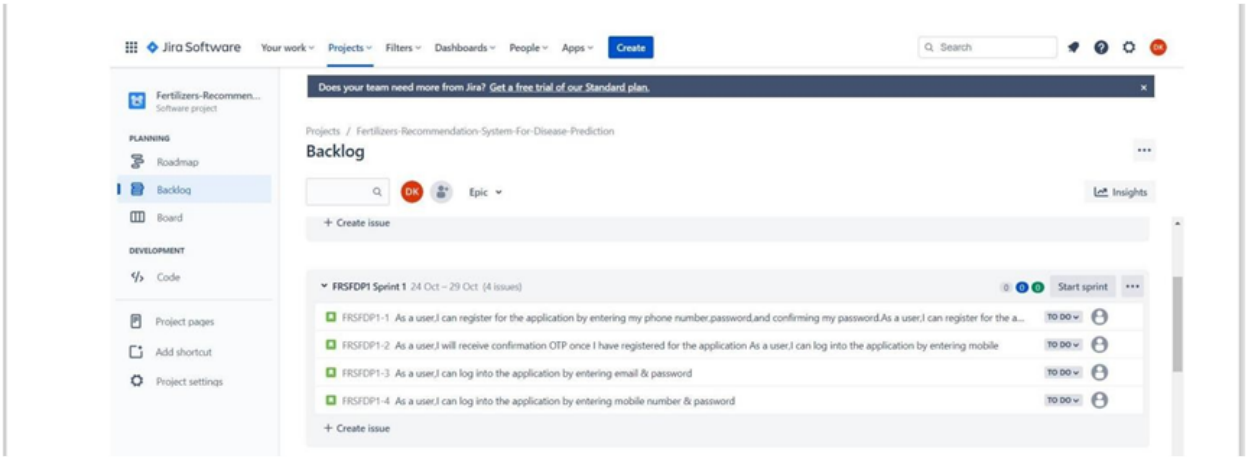
6.2 Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration**	**Sprint StartDate**	**Sprint End Date(Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date(actual)**
Sprint-1	10	6 Days	24 Oct 2022	29 Oct 2022	10	30 Oct 2022
Sprint-2	15	6 Days	31 Oct 2022	05 Nov 2022	15	06 Nov 2022
Sprint-3	15	6 Days	07 Nov 2022	12 Nov 2022	15	13 Nov 2022
Sprint-4	12	6 Days	14 Nov 2022	19 Nov 2022	10	20 Nov 2022

6.3 Reports from JIRA

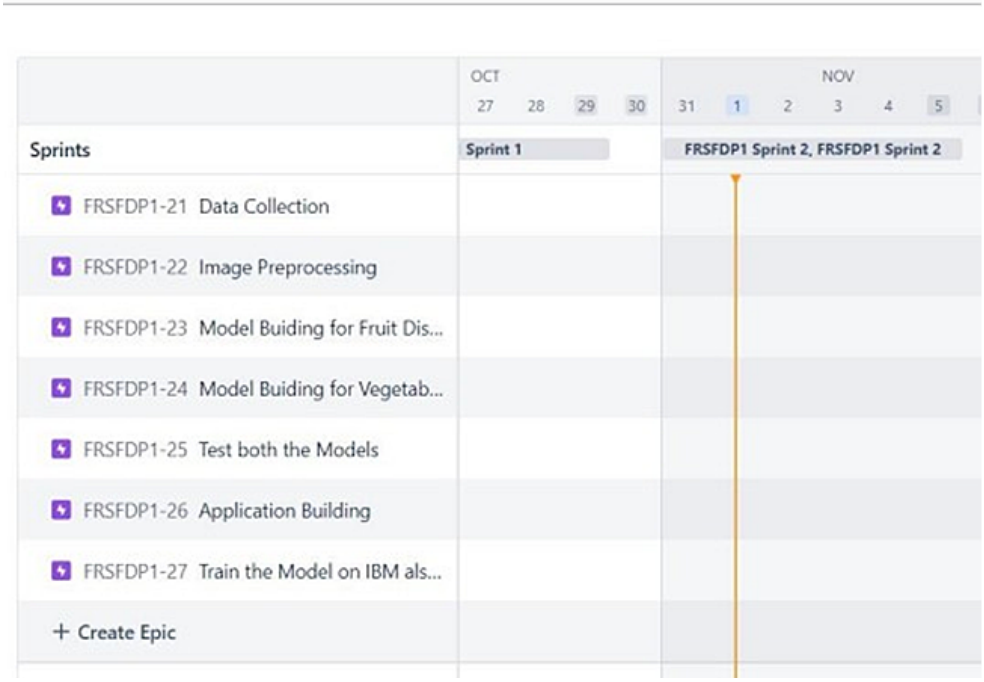






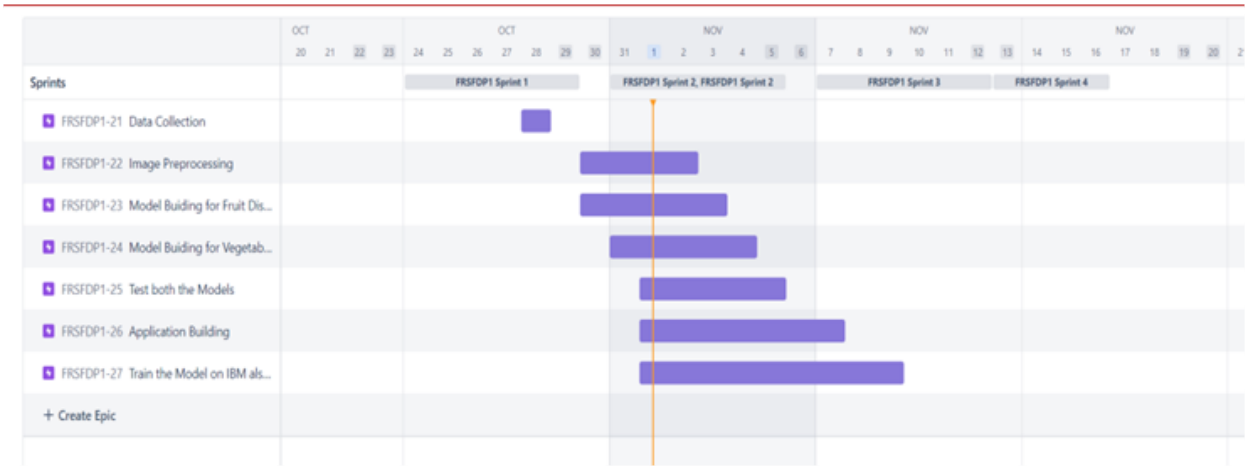
Board:

A board reflects your team's process, tracking the status of work. The columns on the board represent the status of your team's issues. The visual representation of the work helps in discussing and tracking of the progress of the project from start to finish.



Roadmap:

A roadmap offers quick and easy planning that helps teams better manage their dependencies and track progress on the big picture in real-time.



7.CODING & SOLUTIONING

7.1 Feature 1

home.html

```
<!DOCTYPE html>

<html >

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <title> Plant Disease Prediction</title>

  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>

<style>

.header {

                        top:0;

                        margin:0px;

                        left: 0px;
```

```
                right: 0px;
                position: fixed;
                background-color: #4BDB21;
                color: white;
                box-shadow: 0px 8px 4px grey;
                overflow: hidden;
                padding-left:20px;
                font-family: 'Josefin Sans';
                font-size: 2vw;
                width: 100%;
                height:8%;
                text-align: center;
            }
            .topnav {
  overflow: hidden;
  background-color: #333;
}

.topnav-right a {
  float: left;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
```

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

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

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

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

body {

  background-color:#ffffff;
  background-repeat: no-repeat;
```

```
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #28272c;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  border-radius: 12px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
```

```
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

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

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
```

```
  background-color: #f44336;
}

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

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

.container {
  padding: 16px;
}

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

/* Change styles for span and cancel button on extra small screens */
```

```
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
          margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;
}
.login{
          margin:80px;
          box-sizing: content-box;
  width: 84%;
  height: 420px;
```

```
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}

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

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

/* Caption text */
.text {
  color: #f2f2f2;
```

```
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active {
  background-color: #717171;
}

/* Fading animation */
```

```
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-key frames fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
</style>
</head>
```

```
<body style="font-family:'Times New Roman', Times, serif;background-color:white;">


<div class="header">

      <div style="width:50%;float:left;font-size:2vw;text-align:left;color:white; padding-top:1%">Fertilizer Recommendation System</div>

  <div class="topnav-right"style="padding-top:0.5%;">

    <a class="active" href="{{ url_for('home')}}">Home</a>

    <a href="{{ url_for('prediction')}}">Predict</a>

  </div>

</div>


<div style="background-color:#ffffff;">

<div style="width:60%;float:left;">

<div style="font-size:50px;font-family:Montserrat;padding-left:20px;text-align:left;padding-top:10%;">

<b>Agriculture<br></b></div><br>

<div style="font-size:20px;font-family:Montserrat;padding-left:70px;padding-right:30px;text-align:justify;">Agriculture, with its allied sectors, is unquestionably the largest livelihood provider in India, more so in the vast rural areas. It also contributes a significant figure to the Gross Domestic Product (GDP). Sustainable agriculture, in terms of food security, rural employment, and environmentally sustainable technologies such as soil conservation, sustainable natural resource management and biodiversity protection, are essential for holistic rural development. Indian agriculture and allied activities have witnessed a green revolution, a white revolution, a yellow revolution and a blue revolution..Agriculture can help reduce poverty, raise incomes and improve food security for 80% of the world's poor, who live in rural areas and work mainly in farming. The World Bank Group is a leading financier of agriculture.</div><br><br>

</div>
```

```
</div>

<div style="width:40%;float:right;"><br><br>

<i mg src="{{url_for('static',filename='images/12456.png')}}" style="max-height:100%;max-width:100%;">


</div>

</div>


<div class="home">


<br>


</div>


<script>

var slideIndex = 0;

showSlides();


function showSlides() {

  var i;

  var slides = document.getElementsByClassName("mySlides");

  var dots = document.getElementsByClassName("dot");

  for (i = 0; i < slides.length; i++) {
```

```
    slides[i].style.display = "none";
  }
  slideIndex++;
  if (slideIndex > slides.length) {slideIndex = 1}
  for (i = 0; i < dots.length; i++) {
    dots[i].className = dots[i].className.replace(" active", "");
  }
  slides[slideIndex-1].style.display = "block";
  dots[slideIndex-1].className += " active";
  setTimeout(showSlides, 2000); // Change image every 2 seconds
}
</script>
</body>
</html>
```

7.2 Feature 2

predict.html

```
<!DOCTYPE html>
<html >

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Get your Recommendations</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
<link href="{{ url_for('static', filename='css/final.css') }}" rel="stylesheet">
<style>
.header {
                        top:0;
                        margin:0px;
                        left: 0px;
                        right: 0px;
```

```
                position: fixed;
                background-color: #4BDB21;
                color: white;
                box-shadow: 0px 8px 4px grey;
                overflow: hidden;
                padding-left:20px;
                font-family: 'Josefin Sans';
                font-size: 2vw;
                width: 100%;
                height:8%;
                text-align: center;
            }
            .topnav {
  overflow: hidden;
  background-color: #333;
}

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

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

```
}

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

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

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

  background-color:#ffffff;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }

.login{
        margin-top:100px;
}

.container {
  margin-top:40px;
```

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

</style>
</head>

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

<div class="header">
      <div      style="width:50%;float:left;font-size:2vw;text-align:left;color:white;      padding-
```

```
top:1%">Get your Recommendations Here!</div>
  <div class="topnav-right" style="padding-top:0.5%;">
  </div>
</div>
<div class="container">
      <div id="content" style="margin-top:2em">
                <div class="container">
                  <div class="row">
                        <div class="col-sm-6 bd" >

                          <br>
                              <img            src="{{url_for('static',filename='images/789.jpg')}}"
style="height:450px;width:550px"class="img-rounded" alt="Gesture">
                        </div>
                        <div class="col-sm-6">
                              <div>
                                    <h4>Drop in the image to get the prediction  </h4>
                        <form        action        =        ""        id="upload-file"        method="post"
enctype="multipart/form-data">
                              <select name="plant">

                                          <option     value="select"     selected>Select     plant
type</option>
                                      <option value="fruit">Fruit</option>
                                      <option value="vegetable">Vegetable</option>
                </select><br>
                              <label            for="imageUpload"            class="upload-label"
style="background: #28272c;">
                                    Choose...
```

```
                    </label>
                    <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
                </form>

                <div class="image-section" style="display:none;">
                    <div class="img-preview">
                        <div id="imagePreview">
                        </div>
                    </div>
                    <div>
                        <button type="button" class="btn btn-info btn-lg " id="btn-predict" style="background: #28272c;">Predict!</button>
                    </div>
                </div>

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

                <h3>
                    <span id="result" style="font-size:17px; "> </span>
                </h3>

            </div>
                    </div>

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

```
    </div>
</body>

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

8. TESTING

8.1 Test Cases

Test cases are a set of actions performed on a system to determine if it satisfies software requirements and functions correctly as it claimed to perform

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
HomePage_TC_OD01	Functional	Home Page	Verify user is able to see the home page or not.		1.Enter URL and click go 2.verify whether the user is able to see the home page.	Enter URL and click go	User able to see the home page	Working as expected	Pass	Nil	N	_	Bharathkannan R
HomePage_TC_OD02	UI	Home Page	Verify the UI elements in Home Page		1.Enter URL and click go 2.Verify the UI elements in Home Page.	Enter URL and click go	Application should show below UI elements: Home Tab & Predict Tab	Working as expected	pass	Nil	N	_	Gokul S
PredictPage_TC_OD03	Functional	Predict page	Verify user is able to redirect to predict page or not.		1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not.	Click the predict button in home page	User should navigate to Predict page	Working as expected	pass	Nil	N	_	Balabaskaran
PredictPage_TC_OD04	UI	Predict page	Verify the UI elements in Predict Page		1.Enter URL and click go 2.Verify the UI elements in Predict Page.	Click the predict button and redirect to predict page	Application should show below UI elements: Dropdown List , Upload file Button, Predict button.	Working as expected	pass	Nil	N	_	Gokul , Bharathkannan
PredictPage_TC_OD05	Functional	Predict page	Verify user is able to select the dropdown value or not.		1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not.	Fruit or Vegetable	Application should shows user to choose fruit or vegetable option in dropdown list.	Working as expected	pass	Nil	N	_	Pachaiappan
PredictPage_TC_OD06	Functional	Predict page	Verify user is able to upload the image or not.		1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not. 5.Verify user is able to upload the images or not	Images to be Uploaded	Application should shows the uploaded image.	Working as expected	pass	Nil	N	_	Jayaprakash
PredictPage_TC_OD07	Functional	Predict page	Verify whether the image is predicted correctly or not		1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not. 5.Verify user is able to upload the images or not 6. Verify whether the image is predicted correctly or not	Click the Predict Button	Application shows the predicted output	Working as expected	pass	Nil	N	_	Bharathkannan , Gokul

8.2 User Acceptance Testing

Defect and Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	1	0	1
Duplicate	1	3	2	2	8
External	2	3	0	0	5
Fixed	4	4	4	4	16
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	7	10	7	7	31

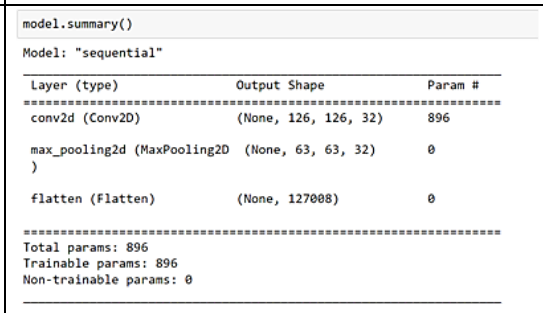
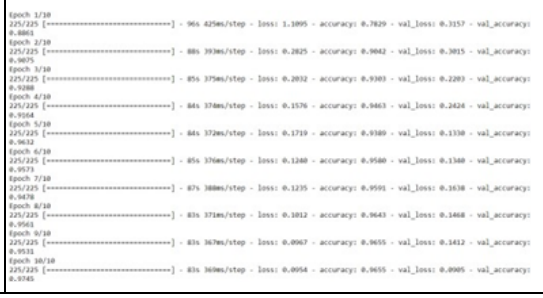
Test case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	1	0	0	1
Client Application	1	0	0	1
Security	1	0	0	1
Outsource Shipping	1	0	0	1
Exception Reporting	1	0	0	1
Final Report Output	1	0	0	1
Version Control	1	0	0	1

9. RESULTS

9.1 Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params:896 Trainable params: 896 Non-trainable params: 0	
2.	Accuracy	Training Accuracy – 96.55 Validation Accuracy – 97.45	

Model Summary

```
model.summary()
```

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

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

 flatten (Flatten)           (None, 127008)            0

=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0
_________________________________________________________________
```

Accuracy

```
model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
```

```
Epoch 1/10
225/225 [==============================] - 96s 425ms/step - loss: 1.1095 - accuracy: 0.7829 - val_loss: 0.3157 - val_accuracy:
0.8861
Epoch 2/10
225/225 [==============================] - 88s 393ms/step - loss: 0.2825 - accuracy: 0.9042 - val_loss: 0.3015 - val_accuracy:
0.9075
Epoch 3/10
225/225 [==============================] - 85s 375ms/step - loss: 0.2032 - accuracy: 0.9303 - val_loss: 0.2203 - val_accuracy:
0.9288
Epoch 4/10
225/225 [==============================] - 84s 374ms/step - loss: 0.1576 - accuracy: 0.9463 - val_loss: 0.2424 - val_accuracy:
0.9164
Epoch 5/10
225/225 [==============================] - 84s 372ms/step - loss: 0.1719 - accuracy: 0.9389 - val_loss: 0.1330 - val_accuracy:
0.9632
Epoch 6/10
225/225 [==============================] - 85s 376ms/step - loss: 0.1240 - accuracy: 0.9580 - val_loss: 0.1340 - val_accuracy:
0.9573
Epoch 7/10
225/225 [==============================] - 87s 388ms/step - loss: 0.1235 - accuracy: 0.9591 - val_loss: 0.1638 - val_accuracy:
0.9478
Epoch 8/10
225/225 [==============================] - 83s 371ms/step - loss: 0.1012 - accuracy: 0.9643 - val_loss: 0.1468 - val_accuracy:
0.9561
Epoch 9/10
225/225 [==============================] - 83s 367ms/step - loss: 0.0967 - accuracy: 0.9655 - val_loss: 0.1412 - val_accuracy:
0.9531
Epoch 10/10
225/225 [==============================] - 83s 369ms/step - loss: 0.0954 - accuracy: 0.9655 - val_loss: 0.0905 - val_accuracy:
0.9745
```

10. ADVANTAGES & DISADVANTAGES

Advantages:

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

Disadvantages:

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

11. CONCLUSION

Hence a system that takes in images as user input, analyses those for certain symptoms and identifies the disease, recommends the fertilizer to counter the deficiency of the nutrients is built and deployed.

12. FUTURE SCOPE

The system must be trained with numerous images of plant disease symptoms. In case of presence of multiple diseases, suitable classification must be done to predict each disease accurately and recommend separate fertilizers as a solution to each deficiency or infection.

13. APPENDIX

13.1 Source Code

app.py

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

app = Flask(__name__,template_folder="../templates")

#load both the vegetable and fruit models
#model = load_model("F:\Project\Model\\vegetable.h5")
model=load_model("F:\Project\Model\\fruit.h5")

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

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

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

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

        # Save the file to ./uploads
        base path = os.path.dirname(__file__)
        file_path = os.path.join(
            basepath, 'uploads', secure_filename(f.filename))
        f.save(file_path)
        img = image.load_img(file_path, target_size=(128, 128))
        x = image.img_to_array(img)
        x = np.expand_dims(x, axis=0)
        plant=request.form['plant']
        print(plant)
        if(plant=="vegetable"):
            preds = model.predict(x)
            preds=np.argmax(preds)
            print(preds)
            df=pd.read_excel('precautions-veg.xlsx')
            print(df.iloc[preds]['caution'])
        else:
            preds = model.predict(x)
            preds=np.argmax(preds)
            df=pd.read_excel('precautions-fruits.xlsx')
            print(df.iloc[preds]['caution'])
```

```
        return df.iloc[preds]['caution']
if __name__ == "__main__":
    app.run(debug=False)
```

home.html

```
<!DOCTYPE html>
<html >

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title> Plant Disease Prediction</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
<style>
.header {
                        top:0;
                        margin:0px;
                        left: 0px;
                        right: 0px;
                        position: fixed;
                        background-color: #4BDB21;
```

```
                color: white;
                box-shadow: 0px 8px 4px grey;
                overflow: hidden;
                padding-left:20px;
                font-family: 'Josefin Sans';
                font-size: 2vw;
                width: 100%;
                height:8%;
                text-align: center;
            }
            .topnav {
  overflow: hidden;
  background-color: #333;
}

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

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

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

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

body {

  background-color:#ffffff;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #28272c;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  border-radius: 12px;
}
.button:hover {
```

```
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

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

button:hover {
  opacity: 0.8;
}

.cancelbtn {
```

```
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

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

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

.container {
  padding: 16px;
}

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

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

```
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
          margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;
}
.login{
          margin:80px;
          box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}
```

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

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

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
```

```
}

.active {
  background-color: #717171;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
</style>
```

```
</head>

<body style="font-family:'Times New Roman', Times, serif;background-color:white;">

<div class="header">
      <div      style="width:50%;float:left;font-size:2vw;text-align:left;color:white;      padding-top:1%">Fertilizer Recommendation System</div>
  <div class="topnav-right"style="padding-top:0.5%;">
    <a class="active" href="{{ url_for('home')}}">Home</a>
    <a href="{{ url_for('prediction')}}">Predict</a>
  </div>
</div>

<div style="background-color:#ffffff;">
<div style="width:60%;float:left;">
<div      style="font-size:50px;font-family:Montserrat;padding-left:20px;text-align:left;padding-top:10%;">
<b>Agriculture<br></b></div><br>
<div      style="font-size:20px;font-family:Montserrat;padding-left:70px;padding-right:30px;text-align:justify;">Agriculture, with its allied sectors, is unquestionably the largest livelihood provider in India, more so in the vast rural areas. It also contributes a significant figure to the Gross Domestic Product (GDP). Sustainable agriculture, in terms of food security, rural employment, and environmentally sustainable technologies such as soil conservation, sustainable natural resource management and biodiversity protection, are essential for holistic rural development. Indian agriculture and allied activities have witnessed a green revolution, a white revolution, a yellow revolution and a blue revolution..Agriculture can help reduce poverty, raise incomes and improve food security for 80% of the world's poor, who live in rural areas and work mainly in farming. The World Bank Group is a leading financier of agriculture.</div><br><br>
</div>
```

```
</div>
<div style="width:40%;float:right;"><br><br>
<img    src="{{url_for('static',filename='images/12456.png')}}"    style="max-height:100%;max-
width:100%;">

</div>
</div>

<div class="home">

<br>

</div>

<script>
var slideIndex = 0;
showSlides();

function showSlides() {
  var i;
  var slides = document.getElementsByClassName("mySlides");
  var dots = document.getElementsByClassName("dot");
  for (i = 0; i < slides.length; i++) {
    slides[i].style.display = "none";
  }
  slideIndex++;
  if (slideIndex > slides.length) {slideIndex = 1}
  for (i = 0; i < dots.length; i++) {
    dots[i].className = dots[i].className.replace(" active", "");
```

```
  }
  slides[slideIndex-1].style.display = "block";
  dots[slideIndex-1].className += " active";
  setTimeout(showSlides, 2000); // Change image every 2 seconds
}
</script>
</body>
</html>
```

predict.html

```
<!DOCTYPE html>
<html >

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Get your Recommendations</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
<link href="{{ url_for('static', filename='css/final.css') }}" rel="stylesheet">
<style>
.header {
                        top:0;
                        margin:0px;
                        left: 0px;
                        right: 0px;
                        position: fixed;
                        background-color: #4BDB21;
                        color: white;
                        box-shadow: 0px 8px 4px grey;
                        overflow: hidden;
                        padding-left:20px;
                        font-family: 'Josefin Sans';
                        font-size: 2vw;
                        width: 100%;
                        height:8%;
                        text-align: center;
                }
                .topnav {
  overflow: hidden;
  background-color: #333;
}

.topnav-right a {
  float: left;
```

```
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

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

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

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

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

  background-color:#ffffff;
  background-repeat: no-repeat;
```

```
  background-size:cover;
  background-position: 0px 0px;
  }

.login{
        margin-top:100px;
}

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

```
        transition: box-shadow .5s ease;
}

</style>
</head>

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

<div class="header">
      <div       style="width:50%;float:left;font-size:2vw;text-align:left;color:white;       padding-top:1%">Get your Recommendations Here!</div>
  <div class="topnav-right" style="padding-top:0.5%;">
  </div>
</div>
<div class="container">
      <div id="content" style="margin-top:2em">
                  <div class="container">
                    <div class="row">
                          <div class="col-sm-6 bd" >

                            <br>
                                <img              src="{{url_for('static',filename='images/789.jpg')}}" style="height:450px;width:550px"class="img-rounded" alt="Gesture">
                          </div>
                          <div class="col-sm-6">
                                <div>
                                      <h4>Drop in the image to get the prediction  </h4>
                          <form       action       =       ""       id="upload-file"       method="post" enctype="multipart/form-data">
```

```
                    <select name="plant">

                                <option value="select" selected>Select plant type</option>
                                <option value="fruit">Fruit</option>
                                <option value="vegetable">Vegetable</option>
                    </select><br>
                    <label for="imageUpload" class="upload-label" style="background: #28272c;">
                        Choose...
                    </label>
                    <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
                </form>


                <div class="image-section" style="display:none;">
                    <div class="img-preview">
                        <div id="imagePreview">
                        </div>
                    </div>
                    <div>
                        <button type="button" class="btn btn-info btn-lg " id="btn-predict" style="background: #28272c;">Predict!</button>
                    </div>
                </div>

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

```
                        <h3>
                                <span id="result" style="font-size:17px; "> </span>
                        </h3>

                </div>
                        </div>

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

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

final.css

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

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

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

.upload-label{
    display: inline-block;
    padding: 12px 30px;
    background: #28272c;
    color: #fff;
    font-size: 1em;
    transition: all .4s;
    cursor: pointer;
}

.upload-label:hover{
    background: #C2C5A8;
    color: #39D2B4;
}

.loader {
    border: 8px solid #f3f3f3; /* Light grey */
```

```
    border-top: 8px solid #28272c; /* Blue */
    border-radius: 50%;
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

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

main.js

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

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

```
    }
}
$("#imageUpload").change(function () {
    $('.image-section').show();
    $('#btn-predict').show();
    $('#result').text('');
    $('#result').hide();
    readURL(this);
});

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

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

    // Make prediction by calling api /predict
    $.ajax({
        type: 'POST',
        url: '/predict',
        data: form_data,
        contentType: false,
        cache: false,
        processData: false,
        async: true,
        success: function (data) {
            // Get and display the result
```

```
            $('.loader').hide();
            $('#result').fadeIn(600);
            $('#result').text('Prediction: '+data);
            console.log('Success!');
        },
    });
  });

});
```

13.2 Github Link

https://github.com/IBM-EPBL/IBM-Project-20350-1659717820

13.3 Project Demo Link

https://drive.google.com/file/d/1Ia2Z6c7soUCEJE4w7O3cPUKXzuXmwAWE/view?usp=sharing