

# CULTIVÓ

A

*Theme Based Project Report*

*Submitted in partial fulfilment of the Requirements  
for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

**SHAMAKKAGARI BHARATH REDDY, 1602-20-737-007**

**MALIGIREDDY CHANDRA KIRAN REDDY, 1602-20-737-008**

**MAHANKALI CHARAN RAJ, 1602-20-737-009**



**Department of Information Technology  
Vasavi College of Engineering (Autonomous)**

**ACCREDITED BY NAAC WITH 'A++' GRADE.  
(Affiliated to Osmania University and Approved by AICTE)**

**Ibrahim Bagh, Hyderabad-31 2022**

**VASAVI COLLEGE OF ENGINEERING (AUTONOMOUS)**  
**(AFFILIATED TO OSMANIA UNIVERSITY) HYDERABAD - 500 030**

**Department of Information Technology**



**DECLARATION BY CANDIDATES**

We, **SHAMAKKAGARI BHARATH REDDY, MALIGIREDDY CHANDRA KIRAN REDDY, MAHANKALI CHARAN RAJ**, bearing hall ticket number, **1602-20-737-007, 1602-20-737-008, 1602-20-737-009** hereby declare that the project report entitled "**CULTIVÓ**" Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering in Information Technology**

This is a record of Bonafide work carried out by me and the results embodied in this project report has not been submitted to any other university or institute for the award of any other degree or diploma.

**SHAMAKKAGARI BHARATH REDDY, 1602-20-737-007  
MALIGIREDDY CHANDRA KIRAN REDDY, 1602-20-737-008  
MAHANKALI CHARAN RAJ, 1602-20-737-009**

**VASAVI COLLEGE OF ENGINEERING (AUTONOMOUS)**  
**(AFFILIATED TO OSMANIA UNIVERSITY) HYDERABAD - 500 030**

**Department of Information Technology**



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled "**CULTIVO**" being submitted by **BHARATH REDDY, CHANDRA KIRAN REDDY, CHARAN RAJ** bearing **1602-20-737-007, 1602-20-737-008, 1602-20-737-009**, in partial fulfilment of the requirements for the completion of THEME BASED PROJECT of **Bachelor of Engineering in Information Technology** is a record of bonafide work carried out by them under my guidance.

**Internal Guide**  
G. Swapna Sri

**External Examiner**

**Dr. K Ram Mohan Rao**  
Professor & HOD,  
Dept. of IT.

## **ACKNOWLEDGEMENT**

We extend our sincere thanks to Dr. S. V. Ramana, Principal, Vasavi College of Engineering for his encouragement.

We express our sincere gratitude to Dr. K. Ram Mohan Rao, Professor & Head, Department of Information Technology, Vasavi College of Engineering, for introducing the Theme-Based Project module in our curriculum, and for his suggestions, motivation, and co-operation for the successful completion of our Theme-Based Project.

We also want to thank and convey our gratitude towards our Theme-Based project coordinators C. Sireesha , K. Srinivas Chakravarthy and G. Swapna Sri, for guiding us in understanding the process of project development & giving us timely suggestions at every phase.

We would also like to sincerely thank the project reviewers for their valuable inputs and suggestions

# **Table of Contents**

<b>1 INTRODUCTION .....</b>	<b>6</b>
1.1 OVERVIEW.....	6
1.2 FEATURES.....	6
1.3 SCOPE.....	6
<b>2 TECHNOLOGY .....</b>	<b>7</b>
2.1 SOFTWARE REQUIREMENTS .....	7
2.2 HARDWARE REQUIREMENTS .....	7
<b>3 PROPOSED WORK.....</b>	<b>8</b>
3.1 DESIGN .....	8
3.2 IMPLEMENTATION .....	8
<b>4 RESULTS .....</b>	<b>30</b>
<b>5 OUTCOMES APART FROM CURRICULUM.....</b>	<b>36</b>
<b>6 CONCLUSION .....</b>	<b>36</b>
<b>7 REFERENCES.....</b>	<b>36</b>

# **1 INTRODUCTION**

## **1.1 Overview**

The Main Objective of this Theme Based Project is to develop a website for farmers in local languages, in which instructions and information are provided to the farmers about seasonal and suitable crops, seeds, farming tools & instruments etc.

## **1.2 Features**

- i. User will be able to get all relevant information on specific subjects about Farming and Agriculture around his village/block/district/state.
- ii. Farmers and Consumers can register, and information will be delivered in the form of SMS and audio/video in the language he or she understands.

## **1.3 Scope**

In an effort to help our farmers, the Government needs a website for the farmers in local languages. It should have instructions and information about seasonal and suitable crops, seeds, farming tools & instruments etc.

Now-a-days information is everywhere but language has become a major barrier. So, we tried to overcome that barrier in our website as we tried to put every information that we thought is useful and translated it to almost all regional languages

## **2 Technology**

### **2.1 Software Requirements**

- Windows 7 & newer or MacOS 11 & newer
- Applications Required Pre-Installed: VS-Code, Xampp
- Runtime Environment: Safari or Chrome

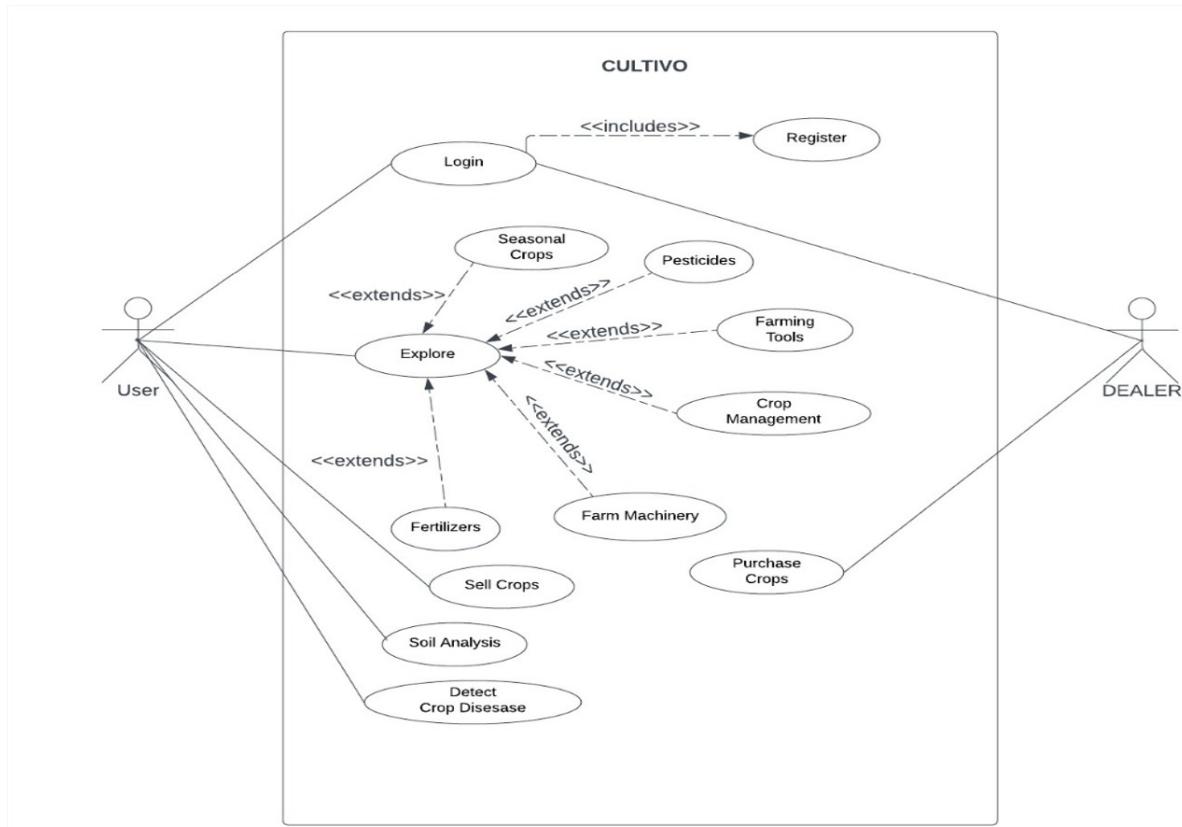
### **2.2 Hardware Requirements**

- x86 64-bit CPU Processor
- 4GB RAM
- 5 GB of free disk space

## 3 PROPOSED WORK

### 3.1 Design:

#### Use-Case Diagram:



### 3.2 Implementation:

#### Module-wise Code:

##### Main.py:

```
from flask import Flask, render_template, request, redirect
import mysql.connector

app=Flask(__name__)

conn=mysql.connector.connect(host="localhost",user="root",password="",database="db")

cursor=conn.cursor()
```

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

def home():
    return render_template('Home.html')

@app.route('/FM')
def FM():
    return render_template('FM.html')

@app.route('/CM')
def CM():
    return render_template('CM.html')

@app.route('/register')
def register():
    return render_template('register.html')

@app.route('/news')
def news():
    return render_template('news.html')

@app.route('/logout')
def logout():
    return redirect('/')

@app.route('/State')
def State():
    return redirect('State.html')

@app.route('/explore')
def explore():
    return render_template('Explore.html')

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

```

def cm():

    return render_template('CM.html')

@app.route('/home1')

def home1():

    return render_template('Home1.html')

@app.route('/statet')

def statet():

    return render_template('Telangana1.html')

@app.route('/login_validation', methods=['POST'])

def login_validation():

    email=request.form.get('email')

    password=request.form.get('password')

    cursor.execute("""SELECT * FROM `users` WHERE `email` LIKE '{}' AND `password` LIKE '{}'"""
                   .format(email,password))

    users=cursor.fetchall()

    if len(users)>0:

        return redirect('/home1')

    else:

        return redirect('/')

@app.route('/add_user',methods=['POST'])

def add_user():

    name=request.form.get('uname')

    mobileno=request.form.get('umobile')

    email=request.form.get('uemail')

    password=request.form.get('upassword')

```

```

        cursor.execute("""INSERT INTO `users`  

        (`name`, `umobile`, `email`, `password`) VALUES  

        ('{}', '{}', '{}', '{}')""".format(name,mobileno,email,password  

        ))  

        conn.commit()  

        return redirect('/home')  

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

```

Here we are importing necessary modules:

- Flask is the main class of the Flask module and is used to create a Flask application.
- render\_template is used to render HTML templates.
- request is used to access incoming request data.
- redirect is used to redirect the user to a different URL.
- mysql.connector is a module for connecting to a MySQL database.

The mysql.connector.connect() function is used to establish a connection to the MySQL database. You need to provide the host, user, password, and database name.

The cursor() method returns a cursor object to execute SQL queries on the database.

The @app.route() decorator is used to define routes for different URLs. In this example, routes for '/', '/FM', '/CM', '/register', '/news', '/logout', '/State', '/explore', '/cm', '/home1', and '/statet' are defined.

## Home.html:

```

<!DOCTYPE html>

<html lang="en">

<head>

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

</head>

```

```

<body>

    <!-- <div class="WebContainer"> -->

        <button onclick="topFunction()" id="myBtn" title="Go to top">&#8593</button>

        <div class="main">

            <div class="navbar">

                <div class="icon">

                    <h2 class="logo">Cultivó</h2>

                </div>

                <div class="menu">

                    <ul>

                        <li><a href="#main">HOME</a></li>

                        <li><a href="/news" target="_blank">NEWS</a></li>

                            <!-- <li><a href="textchatbot" target="_blank"></a></li> -->

                        <li><a href="#about">ABOUT</a></li>

                        <li><a href="#contact">CONTACT</a></li>

                            <!-- <li><a href="textchatbot" target="_blank">CHAT</a></li>

                        <li><a href="chatbot" target="_blank">VOICE CHAT</a></li> -->

                    </ul>

                </div>

                <div class = "search" id="google_translate_element">

                </div>

            <script type="text/javascript">

```

```

        function googleTranslateElementInit() {
            new google.translate.TranslateElement(
                {pageLanguage: 'en'},
                'google_translate_element'
            );
        }
    
```

</script>

<script type="text/javascript"

src="https://translate.google.com/translate\_a/element.js?cb=googleTranslateElementInit">

</script>

</div>

<div class="content">

<h1><p>A Farmer-friendly</p><span>Website</span><br><p> Regional Languages</p></h1><br>

<!-- <button class="cn"><a href="/explore">Explore</a></button> -->

<form class="form" method="post" action="/login\_validation">

<h2>Sign In</h2>

<input type="email" class="form-control" name="email" placeholder="Enter Email">

<input type="password" class="form-control" name="password" placeholder="Enter Password">

<button type="submit" class="bttn btn-primary btn-block btn-lg" value="Login"><a>Login</a></button>

<p class="link">Don't have an account<br>

<a href="/register">Sign-up</a> here</a></p>

```

        </form>

        </div>

        </div>

    </div>

    <!-- <div id="crops">

        <a href="cbmain.js"></a>

    </div> -->

<div class="container1" id="about">

        <h4 style="color:rgb(5, 5, 5); font-size:2.13em; font-family: fantasy;">About</h4>

        <p>Farming has been the backbone of our country. More than 42 percent in India are employed directly or farming related activities. Empowering our farmers will effectively help our economy. This will have instructions and information about seasonal and suitable crops, seeds, farming tools & instruments etc.</p>

    </div>

</div>

<div class="bg-img" id="contact">

    <form action="/action_page" class="container">

        <h1>Contact</h1>

        <label for="name"><b>Name</b></label>

        <input type="text" placeholder="Enter name" name="name" required>

        <label for="mobile number"><b>Mobile Number</b></label>

```

```

        <input type="text" placeholder="Enter Mobile
number" name="mobile number" required>

        <label for="sug"><b>Queries</b></label>

        <input type="text" placeholder="Enter queries"
name="sug" required>

        <button type="submit" class="btn">Submit</button>

    </form>

</div>

<script>

    // Get the button

    let mybutton = document.getElementById("myBtn");

    // When the user scrolls down 20px from the top of
    // the document, show the button

    window.onscroll = function() {scrollFunction()};

    function scrollFunction() {

        if (document.body.scrollTop > 20 ||

document.documentElement.scrollTop > 20) {

            mybutton.style.display = "block";

        } else {

            mybutton.style.display = "none";

        }

    }

    // When the user clicks on the button, scroll to the
    // top of the document

    function topFunction() {

        document.body.scrollTop = 0;

        document.documentElement.scrollTop = 0;
    }

```

```

        }

    </script>

</body>

</html>

```

## **Explore.html:**

```

<!DOCTYPE html>

<html lang="en">

<head>

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

</head>

<body>

    <div class="content">

        <div class="split left">

            <div class="centered">

                <!-- <div id="leftpnl" > -->

                <div class="top">

                    <!-- <h1 class="h1 heading_icon_agri_horti" style="padding: 0px 0px 5px 5px !important;">Agriculture Corner</h1> -->

                </div>

                <div class= "container">

                    <h2 class="h1" style="color: rgb(246, 242, 242);">Agriculture Corner </h2>

                    <p>&nbsp;</p>

                    <p>&nbsp;</p>

                <!-- <div class="box">

```

```

<a href="FM.html">Farm Machinery</a>

        <div class="popup"
onclick="myFunction()">Farm Machinery

                <span class="popuptext"
id="myPopup"><button class="cn"><a
href="Home.html">Tractor</a></button><br>Seeder<br>Combin
e</span>

        </div>

    </div> -->

    <div class="box">

        <a href="/cm">Crop Management</a>

        <!-- <div class="popup"
onclick="myFunction()">Crop Management

                <span class="popuptext"
id="myPopup"><button class="cn"><a
href="Home.html">Tractor</a></button><br>Seeder<br>Combin
e</span>

        </div> -->

    </div>

    <div class="box">

        <a
href="https://diseaseidentification.anvil.app"
target="_blank">Disease Detection</a>

        <!-- <div class="popup"
onclick="myFunction()">Farm Tools

                <span class="popuptext"
id="myPopup"><button class="cn"><a href="Home.html">Grape
hoe</a></button><br>Seeder<br>Combine</span>

        </div> -->

    </div>

    <div class="box">

        <a href="">Crop Monitoring</a>

```

```

        <!-- <div class="popup"
onclick="myFunction()">Fertilizers

                <span class="popuptext"
id="myPopup"><button class="cn"><a
href="Home.html">Tractor</a></button><br>Seeder<br>Combin
e</span>

            </div> -->

        </div>

    </div>

    </div>

<div class="box1">

    <div class="split right">

        <div class="container1">

            <h2 class="h1">Click on a State to navigate
</h2>

        <div class="centered">

            <div class="middle">

                <table style="width: 100%; height: 434px;
clear: both; margin: 0; padding: 0;" border="0">

                    <tbody>

                        <tr>

                            <td>

                                <!-- <div class="transition">
-->

                                <!-- <p align="right"
style="padding: 0px 0px 0px 50px"> -->

```

```

<map name="Map"
id="Map1">

    <area
href="State.html?id=2" alt="ANDHRA PRADESH" shape="poly"
coords="154,384,165,385,180,380,191,374,201,369,214,371,2
22,363,225,354,243,347,256,337,265,329,280,331,274,343,26
3,352,254,359,245,366,239,370,239,377,230,379,221,385,217
,387,210,395,206,406,211,417,211,426,201,431,188,435,182,
439,174,443,177,434,170,427,169,421,162,423,153,422,149,4
19,148,410,146,401,150,394">

    <area href="/statet"
target="_blank" alt="TELANGANA" shape="poly"
coords="158,382,159,366,160,351,159,340,166,331,177,320,1
90,323,201,324,202,338,210,339,217,342,222,353,224,361,21
9,371,210,372,200,373,193,377,188,382,182,385,168,389,163
,384">

    <area
href="State.aspx?SCode=24" alt="JHARKHAND" shape="poly"
coords="264, 232, 276, 233, 281, 237, 288, 234, 299, 231,
304, 230, 309, 234, 317, 230, 320, 221, 327, 219, 328,
224, 324, 233, 319, 239, 309, 244, 306, 247, 301, 248,
300, 254, 307, 255, 311, 257, 314, 265, 314, 273, 305,
270, 303, 276, 291, 275, 290, 269, 280, 271, 273, 268,
275, 261, 271, 253, 269, 246, 263, 241, 261, 238">

    <area
href="State.aspx?SCode=25" alt="UTTARAKHAND" shape="poly"
coords="167, 127, 169, 120, 176, 113, 180, 110, 186, 110,
191, 110, 195, 117, 200, 118, 207, 120, 207, 124, 214,
126, 218, 129, 217, 133, 212, 137, 210, 142, 208, 147,
206, 154, 203, 159, 200, 158, 196, 157, 194, 155, 188,
152, 186, 146, 180, 144, 179, 139, 176, 141, 171, 144,
168, 136">

    <area
href="State.aspx?SCode=31" alt="A_AND_N_ISLANDS"
shape="poly" coords="407, 378, 401, 395, 398, 400, 396,
406, 394, 410, 393, 418, 393, 425, 389, 433, 385, 435,
389, 451, 394, 454, 397, 469, 402, 481, 412, 499, 422,
515, 429, 516, 422, 504, 421, 495, 421, 486, 410, 480,
403, 471, 400, 454, 403, 434, 414, 428, 419, 421, 422,
400, 417, 388">

    <area
href="State.aspx?SCode=32" alt="ARUNACHAL_PRADESH"
shape="poly" coords="389, 178, 386, 173, 381, 171, 380,
164, 384, 166, 390, 165, 394, 163, 397, 157, 404, 154,
404, 149, 411, 147, 413, 145, 418, 139, 424, 134, 434,
```

```
137, 442, 131, 448, 132, 448, 137, 452, 138, 453, 147,
464, 150, 464, 158, 461, 163, 464, 169, 457, 167, 443,
179, 434, 177, 437, 171, 445, 167, 444, 161, 444, 154,
422, 164, 414, 172, 409, 176">

    <area
    href="State.aspx?SCode=34" alt="DADRA_AND_NAGAR_HAVELI"
    shape="poly" coords="82, 308, 87, 304, 93, 308, 88, 311,
    83, 310">

    <area
    href="State.aspx?SCode=36" alt="Daman & Diu"
    shape="poly" coords="53, 299, 54, 294, 59, 295, 57, 297,
    54, 298">

    <area
    href="State.aspx?SCode=33" alt="CHANDIGARH"
    coords="151,118,156,116,158,121,154,126,152,121">

</map>

<!-- -->

</td>

</tr>

</tbody>

</table>

</div>

</div>
```

```

// When the user clicks on <div>, open the popup

function myFunction() {

    var popup = document.getElementById("myPopup");

    popup.classList.toggle("show");
}

</script>

</body>

</html>

```

## **Telangana.html:**

```

<html>

    <head>

        <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/Telangana1.css') }}>

    </head>

    <body style="background-image: url(img/about.jpg); width: 300px; height: 200px;">

        <div class="split left">

            <h2 style="text-align: center;"><b>Hover over a district to view details</b></h2>

            <map name="imagemap">

                <area shape="poly"
coords="574,184,209,10,198,54,185,104,215,126,249,109,281
,116,280,94,305,66,287,37,244,23,214,13" title="Adilabad(City of Cotton) - The major crops of the Adilabad district are Jowar, Paddy, Cotton, Wheat, Maize, Chilli, Sugarcane, and Soybean. Other major crops are Soybeans, Pulses like Red Gram and Bengal Gram, and Grains. Fruits, vegetables, and spices are grown on an additional 92,900 hectares. More than half of this land is on Mangoes and the rest is Oranges, Tomatoes, Peppers,">

```

Eggplants, Cabbage, Okra, Turmeric, and Coriander"  
href="">

<area shape="poly"  
coords="106,238,89,237,76,270,166,313,178,297,226,304,233  
,258,163,246,153,256,135,252,112,242" title="Kamareddy -  
The major crops grown in the area are Paddy, Sugarcane,  
and Maize." href="">

<area shape="poly"  
coords="254,227,238,256,239,285,306,287,311,269,302,240,2  
61,224" title="Sirsilla" href="">

<area shape="poly"  
coords="312,220,316,274,331,285,381,302,395,291,403,266,3  
88,266,349,255,331,223,314,220" title="Karimnagar"  
href="">

<area shape="poly"  
coords="335,214,355,249,388,264,426,239,437,236,441,215,4  
20,218,376,190,346,206" title="Peddapally" href="">

<area shape="poly"  
coords="61,510,76,518,104,495,102,529,91,537,103,545,104,  
575,98,595,76,611,24,593,50,577,57,560,58,538,60,517"  
title="Narayananpet" href="">

<area shape="poly"  
coords="109,503,131,513,150,503,192,508,202,550,185,545,1  
70,550,166,566,147,565,142,585,114,599,107,592,113,567,10  
7,537,108,505" title="Mahabubnagar" href="">

<area shape="poly"  
coords="83,612,64,624,66,679,124,684,156,698,171,693,135,  
654,99,613" title="Jogulamba Gadwal - This district is  
most famous for its Cotton production in the state. The  
Cotton grown here, especially in the Gattu Mandal, is  
classified as a high-quality cotton seed due to its soil  
condition, water, and rainfall." href="">

<area shape="poly"  
coords="151,566,148,582,119,604,104,601,91,608,133,638,17  
3,686,187,684,172,664,173,625,171,617,192,608,161,590,162  
,575" title="Wanaparthy" href="">

<area shape="poly"  
coords="172,554,165,589,194,604,191,615,181,625,178,657,1  
89,674,206,656,239,656,259,665,279,652,292,637,309,629,33  
9,631,341,616,290,580,250,581,252,555,261,544,228,532,219  
,540,209,540,205,558,187,553" title="Nagarkurnool"  
href="">

```

<area shape="poly"
coords="383,448,349,443,339,457,310,456,306,478,286,489,2
65,507,273,524,266,555,258,577,294,573,341,607,350,577,41
8,555,385,448" title="Nalgonda" href="">

<area shape="poly"
coords="376,424,401,479,415,519,422,551,450,551,459,563,4
83,531,470,520,480,506,450,473,435,435,380,416"
title="Suryapet" href="">

<area shape="poly"
coords="445,450,456,459,458,477,481,502,505,499,519,504,5
32,529,544,529,565,540,564,528,564,514,538,506,534,491,55
0,490,568,474,581,484,609,494,614,450,586,454,556,450,536
,434,537,424,509,413,500,436,484,444,470,438,449,447"
title="Khammam" href="">

</map>

</div>

<div class="box">

<h2>Crops Of Telangana</h2>

<p>

    Telangana state has considered agriculture as its primary goal to improve farmer community wellness, educate on latest technical farming knowledge, train framers to boost the agricultural production and productivity.

</p>

<p>&nbsp;</p>

<div class="table">

<table>

<tr>

<th>S.No.</th>

<th>Climate Zone</th>

<th>Kharif Season Crop</th>

<th>Rabi Season Crop</th>

```

```
</tr>

<tr>

    <td>1</td>

    <td>Northern Telangana Zone</td>

    <td>Rice, Cotton, Maize, Soybean, Red gram</td>

    <td>Rice, Maize, Jowar</td>

</tr>

<tr>

    <td>2</td>

    <td>Central Telangana Zone</td>

    <td>Rice, Cotton, Maize</td>

    <td>Rice, Maize</td>

</tr>

<tr>

    <td>3</td>

    <td>Southern Telangana Zone</td>

    <td>Rice, Cotton, Maize</td>

    <td>Rice, Maize</td>

</tr>

</table>

</div>

</div>

</body>

</html>
```

## MangoDisease.ipynb:

```
!pip install gdown  
  
!gdown --id 1F3C7qwo16TnvdTP3dRys1s90BV_uMONk  
  
!unzip lemondetect.zip  
  
!mv /mangodetection /content/  
  
#Importing necessary modules  
  
import tensorflow as tf  
  
import IPython.display as display  
  
import PIL  
  
import PIL.Image  
  
import numpy as np  
  
import matplotlib.pyplot as plt  
  
import os  
  
import time  
  
import pathlib  
  
from tensorflow.keras.models import Sequential  
  
from tensorflow.keras.layers import Dense, Conv2D,  
Flatten, Dropout, MaxPooling2D  
  
import matplotlib.pyplot as plt  
  
from tensorflow.keras.callbacks import ReduceLROnPlateau,  
EarlyStopping, ModelCheckpoint  
  
from tensorflow.keras.models import load_model  
  
from tensorflow.keras.preprocessing.image import  
ImageDataGenerator  
  
from tensorflow.keras import layers  
  
from tensorflow.keras.callbacks import ReduceLROnPlateau,  
EarlyStopping, ModelCheckpoint
```

```

#Setting up directories and counting images

data_dir = pathlib.Path('/content/mangodetection/train')

test_dir = pathlib.Path('/content/mangodetection/test')

train_count = len(list(data_dir.glob('*/*.JPG')))

test_count = len(list(test_dir.glob('*/*.JPG')))

#xSetting up image preprocessing and data generators:

batch_size = 32

img_height = 180

img_width = 180

#Generate More Data by Generator

image_gen_train = ImageDataGenerator(rescale =
1./255,validation_split=0.2,rotation_range =
45,width_shift_range=.15,height_shift_range
=.15,horizontal_flip=True,zoom_range=0.5)

#Shuffle All Data

train_data_gen =
image_gen_train.flow_from_directory(batch_size=batch_size
,subset = 'training', directory=
data_dir,shuffle=True,target_size=(img_height,
img_width))

#Divide Data for Validation

validation_data_gen =
image_gen_train.flow_from_directory(batch_size=batch_size
,subset = 'validation', directory=
data_dir,shuffle=True,target_size=(img_height,
img_width))

CLASS_NAMES = np.array([item.name for item in
data_dir.glob('*') if item.name != "LICENSE.txt"])

num_classes = 2

#Defining the model architecture

model = Sequential([

```

```

        Conv2D(16, 3, padding='same',
activation='relu',input_shape=(img_height, img_width
,3)),

        MaxPooling2D(),

        Conv2D(32, 3, padding='same', activation='relu'),

        MaxPooling2D(),

        Conv2D(64, 3, padding='same', activation='relu'),

        MaxPooling2D(),

        Dropout(0.2),

        Flatten(),

        Dense(128, activation='relu'),

        Dense(num_classes)

    )

model.compile(optimizer='adam',loss=tf.keras.losses.BinaryCrossentropy(from_logits=True),metrics=['accuracy'])

model.summary()

#Loading a pre-trained model

model_mango =
load_model('/content/mangodetection/model.h5')

#Performing inference on test images

image_gen_test = ImageDataGenerator(rescale = 1./255)

test_data_gen =
image_gen_test.flow_from_directory(batch_size=32,directory=
test_dir,shuffle=True,target_size=(img_height,
img_width))

start_time = time.time()

predictions = model_mango.predict(test_data_gen)

for i in range(len(predictions)):

    if(predictions[i][0]<0):

```

```

        print("Healthy
"+str(abs(predictions[i][0]))+"%")

    else:

        print("Diseased      "+str(100-
abs(predictions[i][0]))+"%")

print("Total Time: ", time.time()-start_time)

print("Average Time Per Image: ", (time.time()-
start_time)/len(predictions))

!pip install anvil-uplink

#Setting up Anvil server and defining a callable function

import anvil.server

anvil.server.connect("SNO5N6UNOGXMICYSINU4DIEJ-
AFBDS2UOAU7ZPMEU")

import anvil.media

import numpy as np

from tensorflow.keras.utils import load_img

from tensorflow.keras.utils import save_img

from tensorflow.keras.utils import img_to_array

import cv2

%cd /content

!mkdir tst

%cd tst

!mkdir test

%cd /content

@anvil.server.callable

def imgload(file):

    model_mango =
load_model('/content/mangodetection/model.h5')

```

```
with anvil.media.TempFile(file) as filename:

    img = load_img(filename)

    img_array = img_to_array(img)

    save_img('/content/tst/test/temp.png',img_array)

    print("Processed Incoming Image")

    image_gen_test = ImageDataGenerator(rescale = 1./255)

    test_data_gen =
image_gen_test.flow_from_directory(batch_size=1,directory
= "/content/tst",shuffle=True,target_size=(img_height,
img_width))

    predictions = model_mango.predict(test_data_gen)

    print(test_data_gen,predictions)

    li=''

    if(predictions[0][0]<0):

        li='Healthy'

    else:

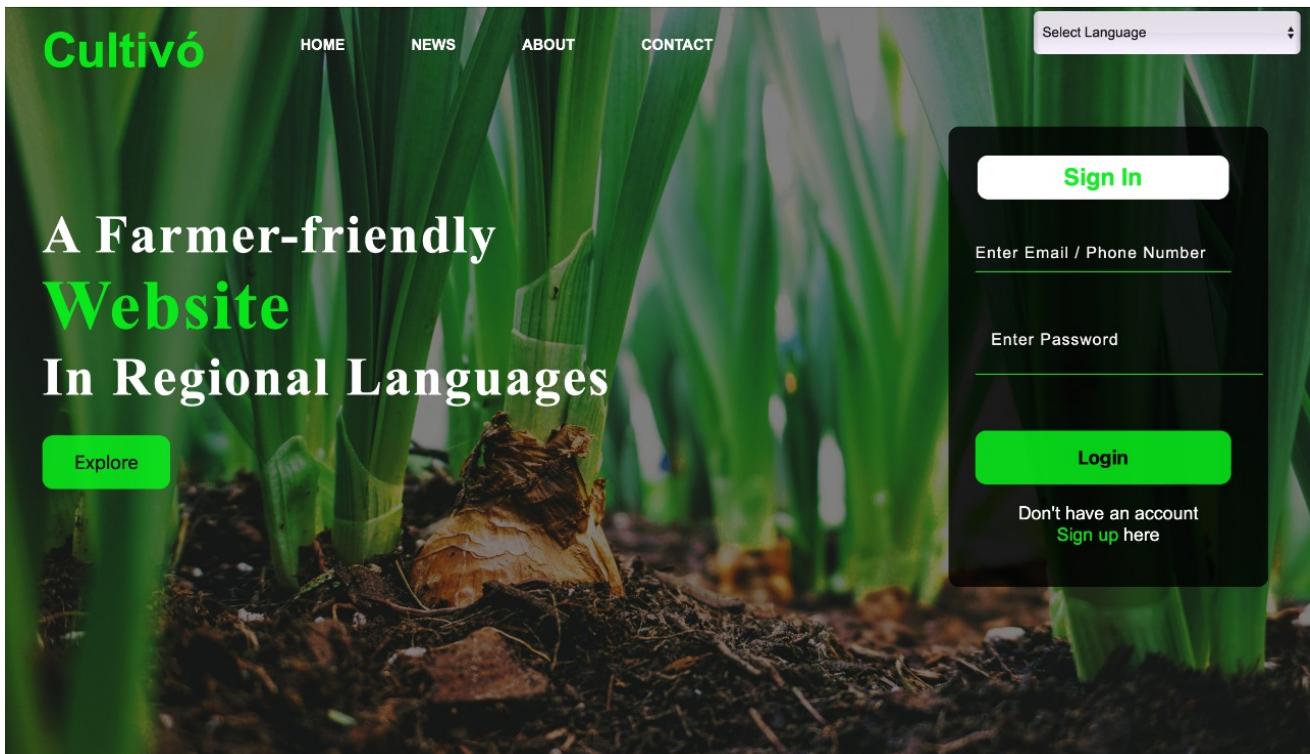
        li="Diseased"

    return(li)

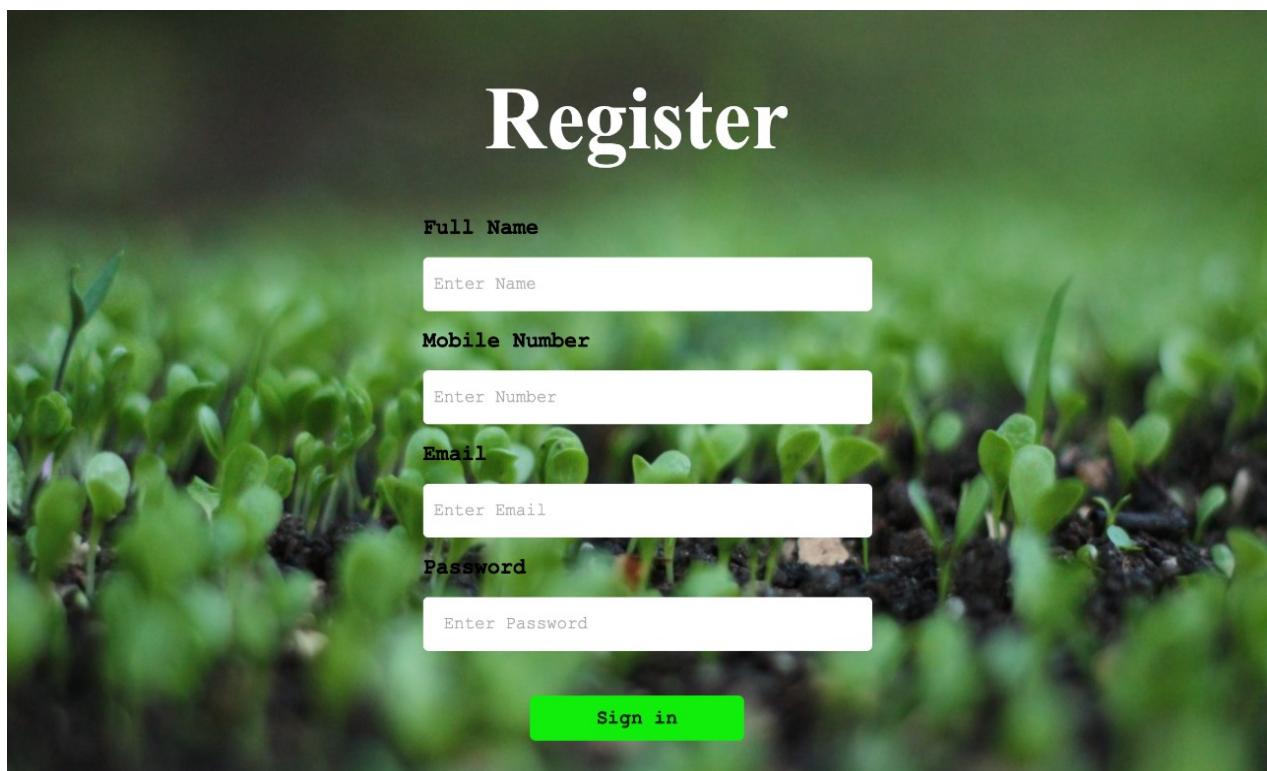
anvil.server.wait_forever()
```

## 4 RESULT

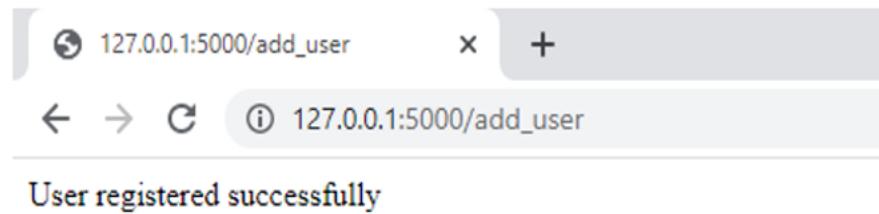
After running the code, first a webpage is displayed:



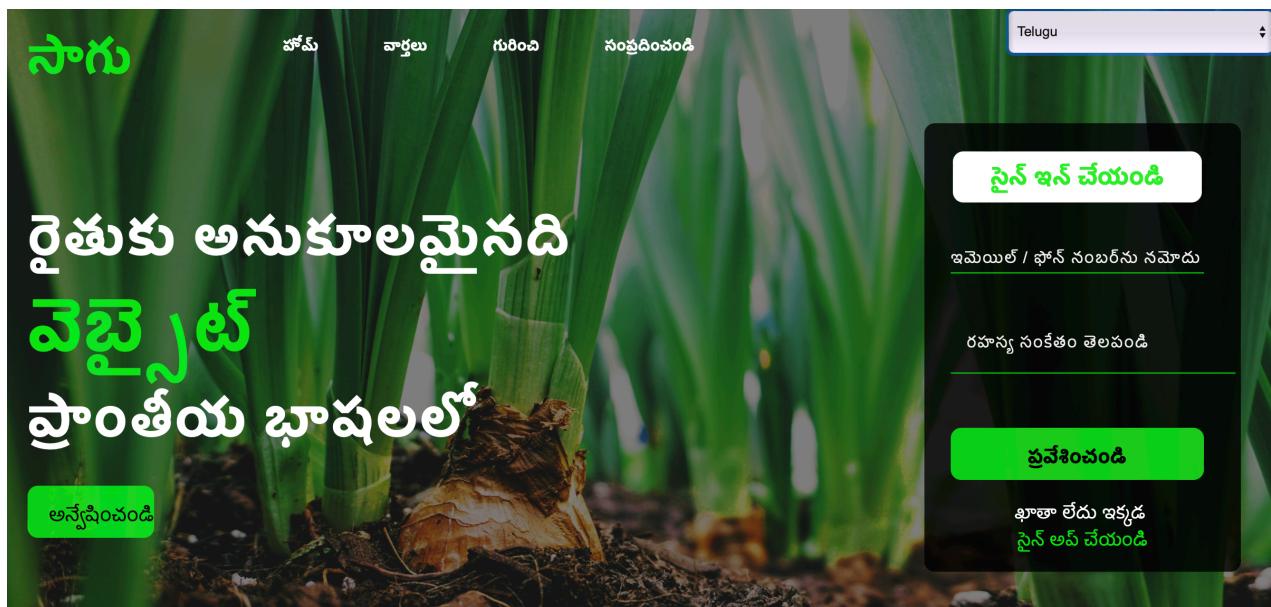
Then there is a login and signup. If someone is new to website, then there is signup page opened when clicked on Signup.



Now After registering, user can login from home page.



Now user can also change the language of the website by just selecting the language.



Now clicking on the news tab, we get :

## Cultivo News



[Read in Kothagudem Sugarcane growers to lobby against inclusion of sugar mill owners in new Karnataka cabinet](#)



Telangana govt prepares crop plan to promote cotton, red gram

[Read More](#)



Agronomic practices helped rejuvenate inundated crops: Telangana

[Read More](#)



India to extend rice export curbs to ensure domestic price stability, supply

[Read More](#)



Maharashtra Budget: Farmers can insure crops at just Rs 1 :Report

[Read More](#)



New Delhi: Govt sets foodgrain production target of 332 MT for 2023-24

[Read More](#)



Agritech can solve farm sector challenges: UNCDF-NITI Aayog Report

[Read More](#)



High-yielding paddy developed in 7 years using speed-breed technique

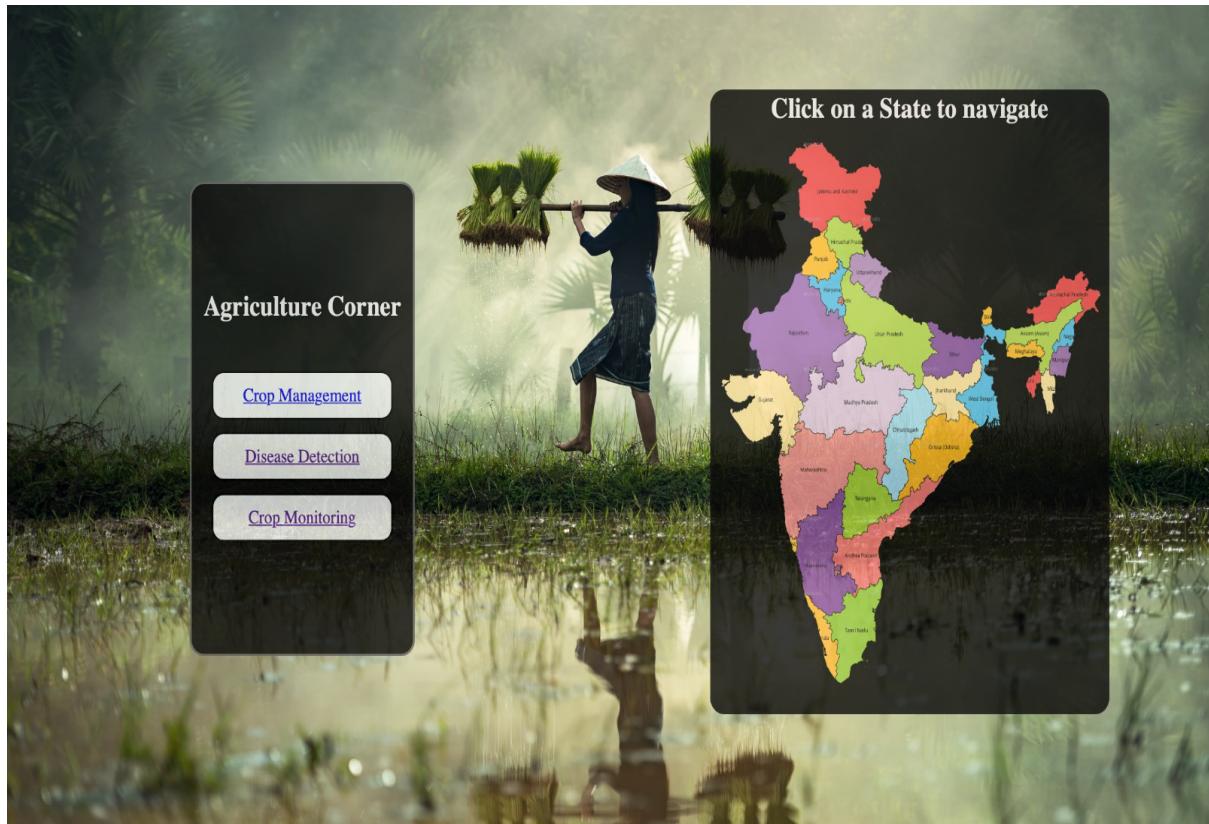
[Read More](#)



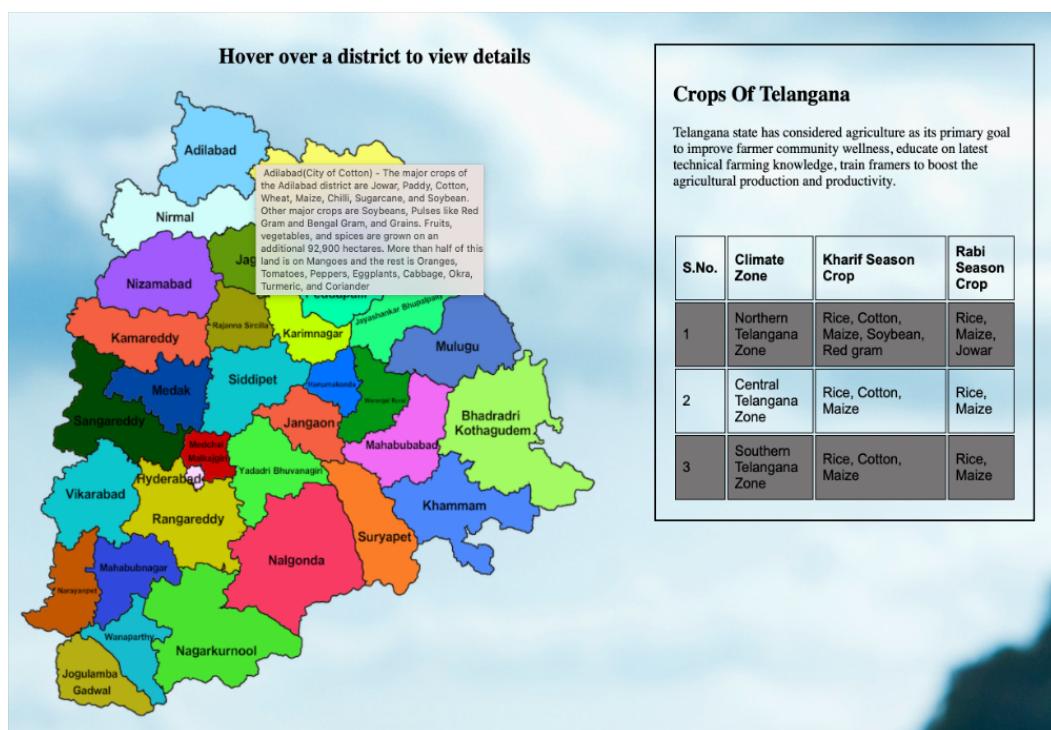
Telangana to soon launch agriculture data exchange, tech to aid agri

[Read More](#)

Now clicking on Explore, we get a lot of our main features:



Now Clicking on Telangana state on the India, we get information about Telangana's crops.



Then Coming back and clicking on Crop Management, we get :

The screenshot shows a dashboard titled "Crop Management". At the top, there is a green header bar with the title. Below it, there is a list of agricultural activities, each represented by a dark grey box with a white label and a plus sign on the right side:

- Soil Preparation
- Sowing Of Seeds
- Addition Of Manures And Fertilizers
- Irrigation
- Protection from Weeds
- Harvesting
- Storage

The background of the dashboard features a photograph of a field with tall grass or crops.

Clicking on Disease Detection, we get :

The screenshot shows a web application titled "Mango Plant Disease Identification". At the top left, it says "Built with Anvil" and at the top right, it says "Build web apps for free with Anvil". The main title is "Mango Plant Disease Identification". Below the title, there is a message: "Please Run Follow the Colab Notebook to run the Backend Server <https://colab.research.google.com/drive/1sRTVwbX3lVU1826SCLnkPVzH1lwFhnYE?usp=sharing>". Below this message is a large input area with a placeholder "Upload" and a blue "UPLOAD" button.

Now, we have to upload the picture of the Leaf and the algorithm will tell whether the leaf is diseased or not.

Built with  anvil

Build web apps for free with Anvil

## Mango Plant Disease Identification

Please Run Follow the Colab Notebook to run the Backend Server <https://colab.research.google.com/drive/1sRTVwbX3lvU1826SCLnkPVzH1lwFhnYE?usp=sharing>

 1 FILE SELECTED



Diseased

Built with  anvil

Build web apps for free with Anvil

## Mango Plant Disease Identification

Please Run Follow the Colab Notebook to run the Backend Server <https://colab.research.google.com/drive/1sRTVwbX3lvU1826SCLnkPVzH1lwFhnYE?usp=sharing>

 1 FILE SELECTED



Healthy

## **5 OUTCOMES**

- > We learned Languages like HTML and also explored different kinds of libraries and modules in Flask and got to know their implementation and working.
- > We have learnt how to build a website using HTML CSS and Flask.
- > We have used modules like TensorFlow.keras, Pandas for Disease Detection.
- > We have also got to know about modules like sql.connector and various components present in those modules.
- > We have implemented GUI using HTML CSS and Backend using Xampp. By doing so, we have learnt so much about building a Website and error handling while doing so.

## **6 CONCLUSION**

By end of this project, we were able to build a language friendly website for farmers and created an environment where farmers can sell their harvest and to search for other requirements like natural alternatives for Chemical pesticides

## **7 FUTURE SCOPE**

- To suggest natural and chemical pesticides in eradicating the disease detected by the Algorithm.
- To develop a mobile application for the same website
- To suggest the nearest farmer to the users based on their present location to buy the required goods.