A

Mini Project

On

PLANT DISEASE DETECTION AND PESTICIDES RECOMMENDATION

Submitted

in partial fulfillment of the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

K. RAJA SULAKSHANA (207R1A0585)

CH. MOHANA LAHARI (207R1A0599)

Under the Guidance of

Dr. D.T.V. Dharmajee Rao

Professor of CSE & Dean Academics

DEPARTMENT OF COMPUTER SCIENCEAND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, NewDelhi, Recognized under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

2020-2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR
GROUP OF INSTITUTIONS
EXPLORE TO INVENT

CERTIFICATE

This is to certify that the project entitled **"PLANT DISEASE DETECTION AND PESTICIDES RECOMMENDATION"** being submitted by **K. RAJA SULAKSHANA (207R1A0585), CH. MOHANA LAHARI (207R1A0599)** in partial fulfillment of the requirements for the award of the degree of B.Tech. in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under my guidance and supervision during the year 2023-24.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. D.T.V. Dharmajee Rao
(Professor of CSE & Dean Academics)
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
Head of the Department

EXTERNAL EXAMINER

Submitted for viva voice Examination held on____________________`_____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Dr. D.T.V. Dharmajee Rao**, Dean Academics and Professor of CSE for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **G.Vinesh Shanker, Dr. J. Narasimharao, Ms. Shilpa, & Dr. K. Maheswari** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju**, Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy**, Director for being cooperative throughout the course of this project. We also express our sincere gratitude to **Sri. Ch. Gopal Reddy**, Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

K. RAJA SULAKSHANA (207R1A0585)

CH. MOHANA LAHARI (207R1A0599)

ABSTRACT

Plant diseases are a major threat to farmers, consumers, and the global economy. In India alone, 35% of field crops are lost to pathogens and pests causing losses to farmers. Indiscriminate use of pesticides is also a serious health concern as many are toxic and biomagnified. These adverse effects can be avoided by early disease detection, crop surveillance and targeted treatments. Most diseases are diagnosed by agricultural experts by examining external symptoms. However, farmers have limited access to experts. Our project is the first integrated and collaborative platform for automated disease diagnosis, tracking and forecasting. Farmers can instantly and accurately identify diseases and get solutions with a mobile app by photographing affected plant parts. Real-time diagnosis is enabled using the latest Artificial Intelligence (AI) algorithms for Cloud-based image processing. The AI model continuously learns from user uploaded images and expert suggestions to enhance its accuracy. Farmers can also interact with local experts through the platform. For preventive measures, disease density maps with spread forecasting are rendered from a Cloud based repository of geo-tagged images and micro-climactic factors. A web interface allows experts to perform disease analytics with geographical visualizations. In our experiments, the AI model (CNN) was trained with large disease datasets, created with plant images self-collected from many farms over 7 months. Test images were diagnosed using the automated CNN model and the results were validated by plant pathologists. Over 95% disease identification accuracy was achieved. Our solution is a novel, scalable and accessible tool for disease management of diverse agricultural crop plants and can be deployed as a Cloud based service for farmers and experts for ecologically sustainable crop production

LIST OF FIGURES/TABLES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

This project is titled “Plant Disease Identification Using CNN”,it involves developing a Cloud Based AI architecture, it uses convolutional neural network to identify common diseases in a variety of crops and horticultural plants, providing real-time diagnosis through user uploaded images. The System will focus on accuracy, user-friendliness and scalability.

1.2 PROJECT PURPOSE

The purpose of your plant disease detection project is to empower farmers and growers with a reliable and accessible tool that uses computer vision and machine learning to swiftly and accurately identify plant diseases. By doing so, the project aims to enhance agricultural productivity, reduce crop losses, and contribute to food security by enabling timely disease management and intervention.

1.3 PROJECT FEATURES

The main features of this project are that this model classifies the It addresses the problem of learning hierarchical representations with a single algorithm or a few algorithms and has mainly beaten records in image recognition, natural language processing, semantic segmentation and many other real world scenarios. There are different deep learning approaches like Convolutional Neural Network(CNN), Stacked Autoencoder, and Deep Belief Network (DBN). CNN mostly used algorithms in image and face recognition.

2. SYSTEM ANALYSIS

2. SYSTEMANALYSIS

SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst hasa firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

The goal of this project is to develop a system that can accurately detect and classify diseases in plants using computer vision and machine learning techniques. Plant diseases can have a significant impact on crop yield and quality, leading to economic losses and food security concerns. Early and accurate detection of these diseases is essential for effective disease management and crop protection.

2.2 EXISTING SYSTEM

In India alone, 35% of field crops are lost to pathogens and pests causing losses to farmers. Indiscriminate use of pesticides is also a serious health concern as many are toxic and biomagnified. These adverse effects can be avoided by early disease detection, crop surveillance and targeted treatments. Most diseases are diagnosed by agricultural experts by examining external symptoms. However, farmers have limited access to experts.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

1) Less accuracy
2) Low Efficiency
3) Indiscriminate use of pesticides is also a serious health concern as many are toxic and biomagnified

2.3 PROPOSED SYSTEM

In this project author using convolution neural network as artificial intelligence to train all plant diseases images and then upon uploading new images CNN will predict plant disease available in uploaded images. For storing CNN train model and images author is using cloud services. so, using Al author predicting plant disease and cloud is used to store data. In this Project author using smart phone to upload image but designing android application will take extra cost and time so we build it as python web application. Using this web application CNN model will get trained and user can upload images and then application will apply CNN model on uploaded images to predict diseases. If this web application deployed on real web server then it will extract users location from request object and can display those location in map.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

1)High accuracy

2)High efficiency

3)Accurately identify diseases and get solutions with a mobile app by photographing affected plant parts

2.4 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

2.4.1 ECONOMICAL FEASIBILITY:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

2.4.2 TECHNICAL FEASIBILITY:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 SOCIAL FEASIBILITY:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.5 HARDWARE& SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS

- **System** : 11th Gen Intel Core
- **Hard Disk** : 8 GB
- **Floppy Drive** : 1.44 Mb
- **Monitor** : 14' Colour Monitor
- **Mouse** : Optical Mouse
- **Ram** : 512 Mb

2.5.2 SOFTWARE REQUIREMENTS

- **Operating system** : Windows 10 and Above
- **Coding Language** : Python
- **Front-End** : Python
- **Designing** : Html, CSS, JavaScript
- **Data Base** : MySQL

3. ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final prediction.

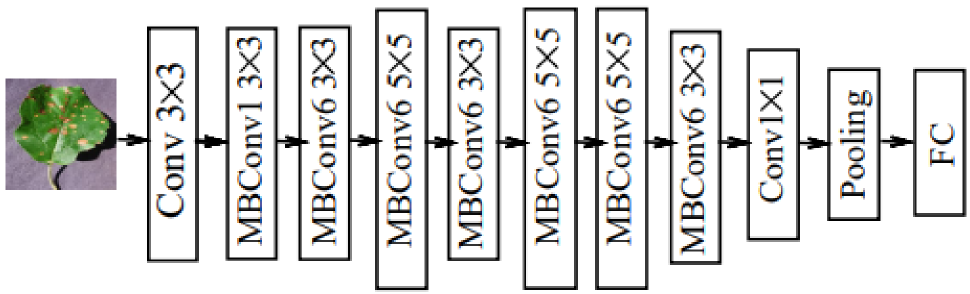


Figure 3.1 : Project Architecture of Plant Disease Detection

3.2 DESCRIPTION

The typical architecture of a CNN involves stacking multiple convolutional and pooling layers to progressively extract hierarchical features from the input data, followed by one or more fully connected layers for making predictions. CNNs have demonstrated outstanding performance in various computer vision tasks and have been adapted for applications in other domains, such as natural language processing and speech recognition, by using techniques like 1D convolutions and attention mechanisms.

3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model. A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of usersthe system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

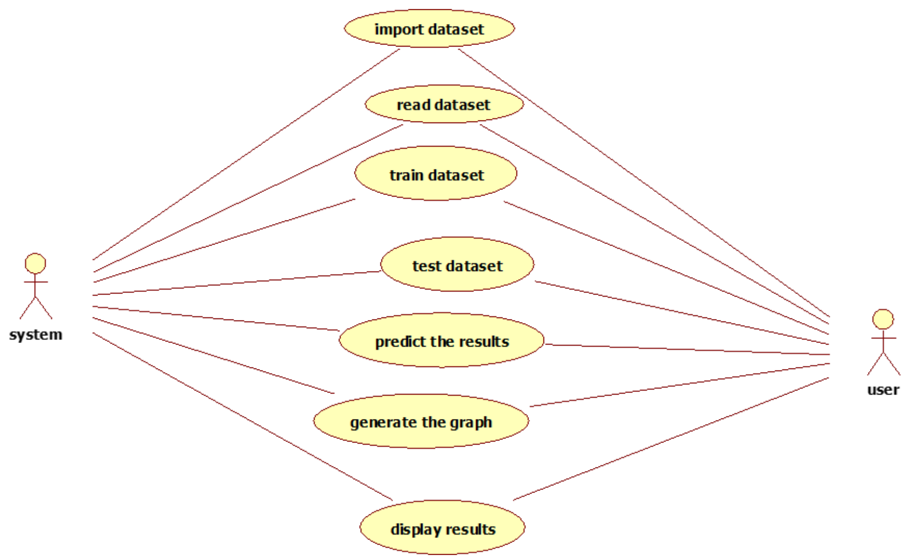


Figure 3.2: Use Case Diagram for Plant Disease Identification and Pesticide Recommendation

3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations(or methods), and the relationships among objects.

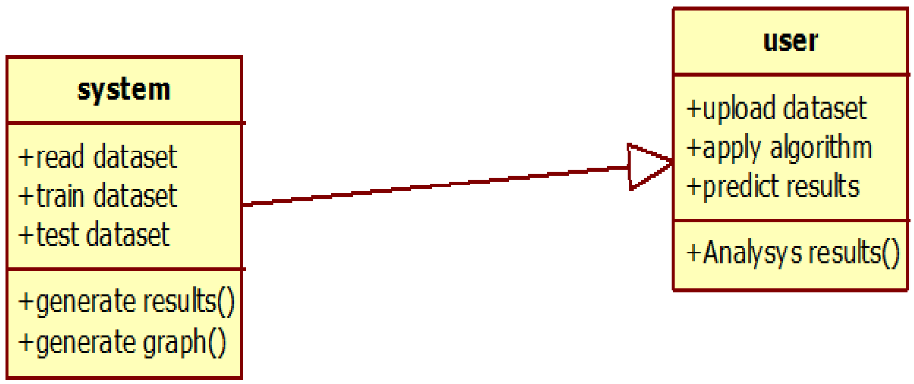


Figure 3.3:Class Diagram for Plant Disease Identification and Pesticide Recommendation

3.5 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

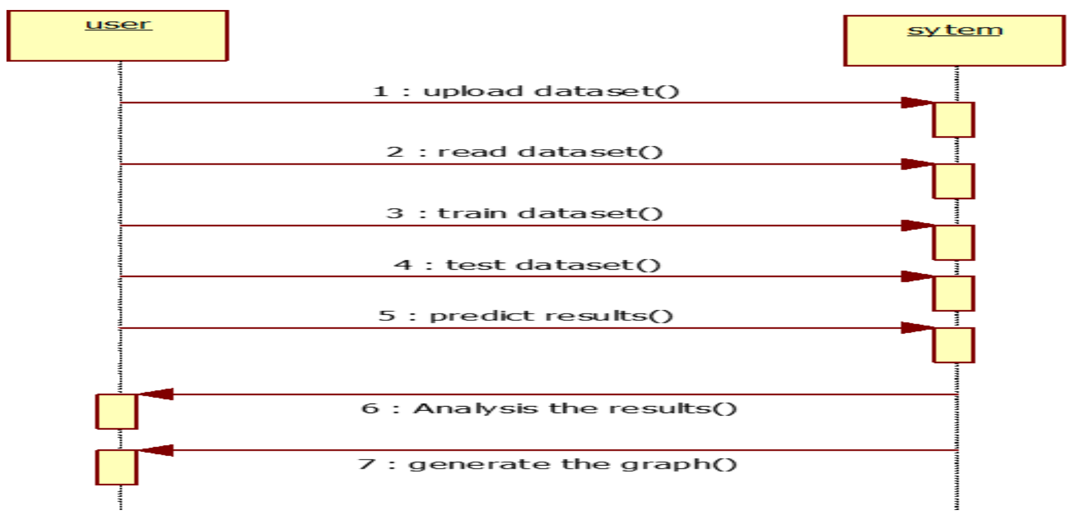


Figure 3.4:Sequence Diagram for Plant Disease Identification and Pesticide Recommendation

3.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more data stores.

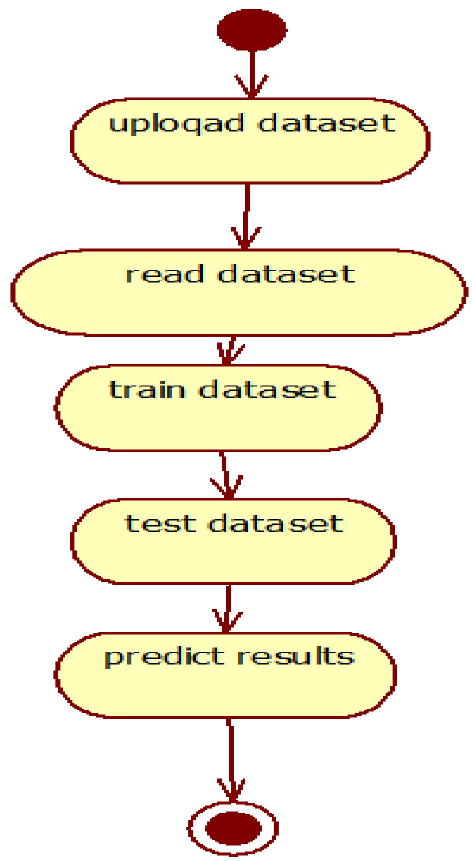


Figure 3.5: Activity Diagram for Plant Disease Identification and Pesticide Recommendation

4.IMPLEMENTATION

4.1 SAMPLE CODE

```
import os
import sys

if__name__== '_main_':
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'PlantDisease.settings')
    try:
        from django.core.management import execute_from_command_line
     except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)



import pymysql
print(pymysql._file_)
pymysql.install_as_MySQLdb()



import os


# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
```

```
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(_file_)))


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'tfa4qqytfcxtc@y9y3abkep@^qu9#3o)e+#n*#+j@^2_a=k@-1'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'PlantDiseaseApp'
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
```

```
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'PlantDisease.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [
                os.path.join('D:/MINI
PROJECT/B2_PlantDisease/PlantDisease/PlantDiseasep', 'templates'),
            ],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'PlantDisease.wsgi.application'


# Database
# https://docs.djangoproject.com/en/2.1/ref/settings/#databases
```

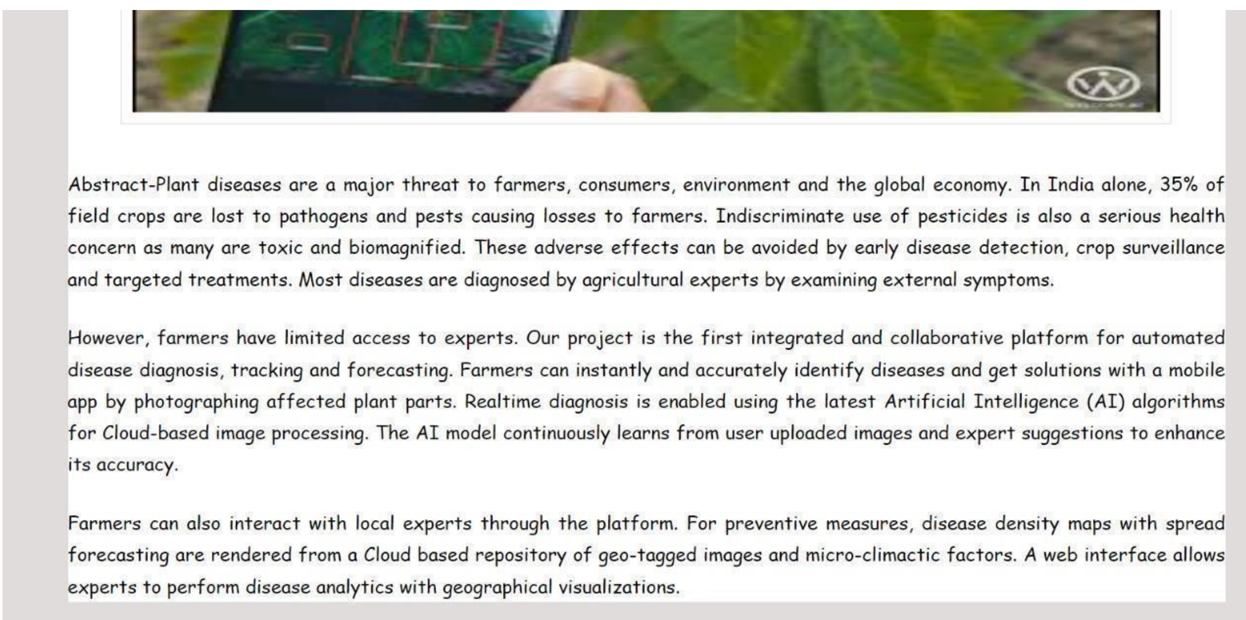
```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'PlantDiseaseDB',
        'HOST': '127.0.0.1',
        'PORT': '3306',
        'USER': 'root',
        'PASSWORD': '',
            'OPTIONS': {
          'autocommit': True,
        },
    }
}


# Password validation
# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
```

```
    },
]


# Internationalization
# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.1/howto/static-files/

STATIC_URL
 from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('PlantDiseaseApp.urls')),
]
```

```
import os

from django.core.wsgi import get_wsgi_application

os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'PlantDisease.settings')

application = get_wsgi_application()
```

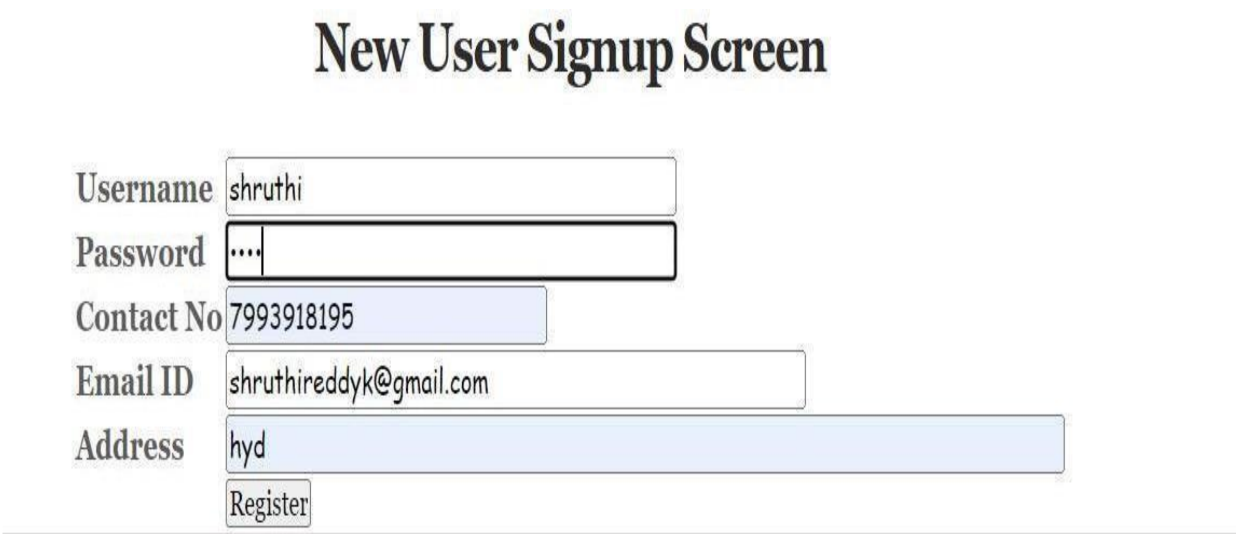
5. SCREENSHOTS

Screenshot 5.1: AI and Cloud Based platform for Plant Disease Identification.

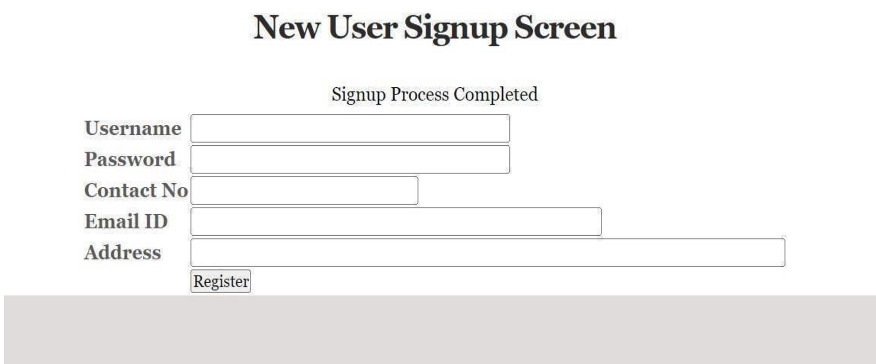


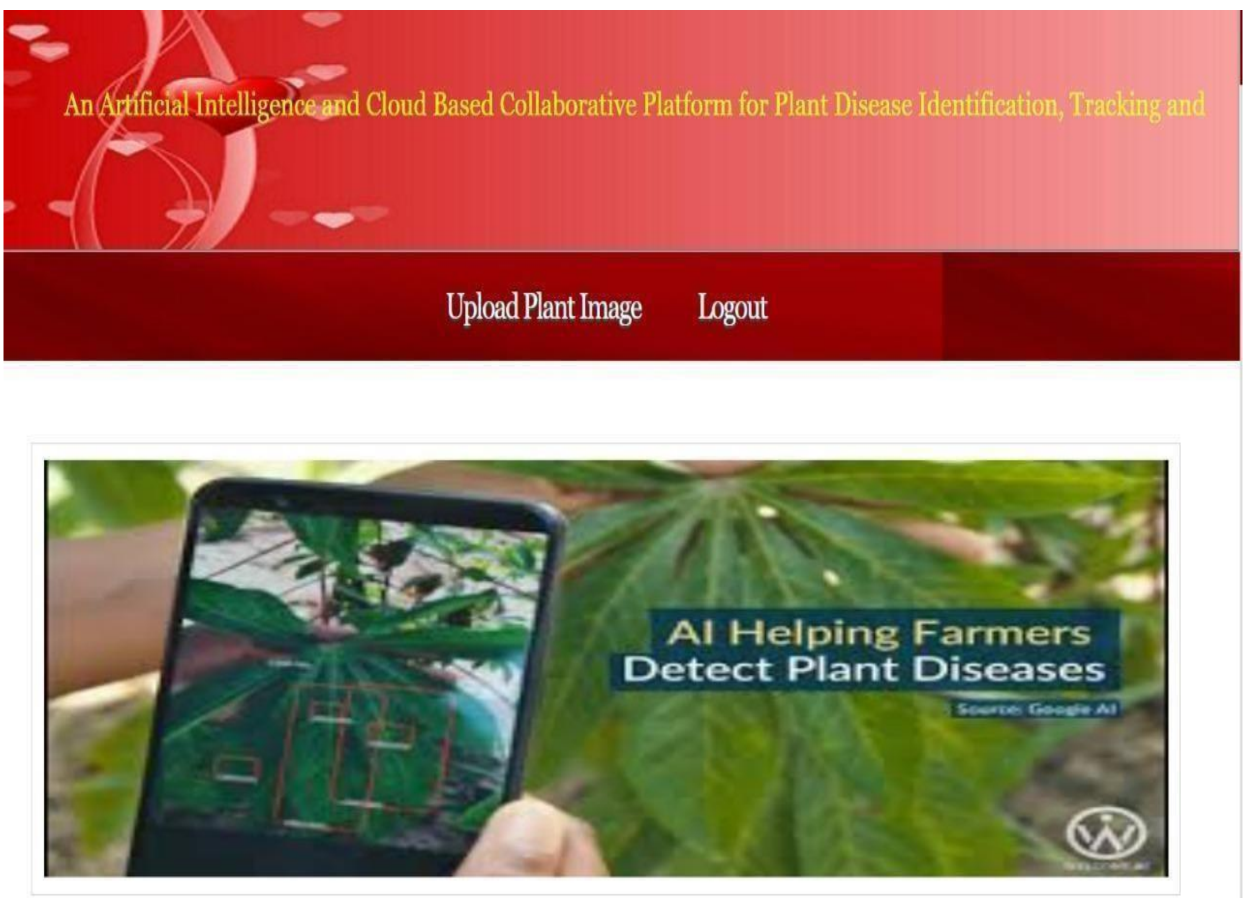
Screenshot 5.2: Abstract displayed in the User Interface.

Screenshot 5.3: New User Sign Up Screen.

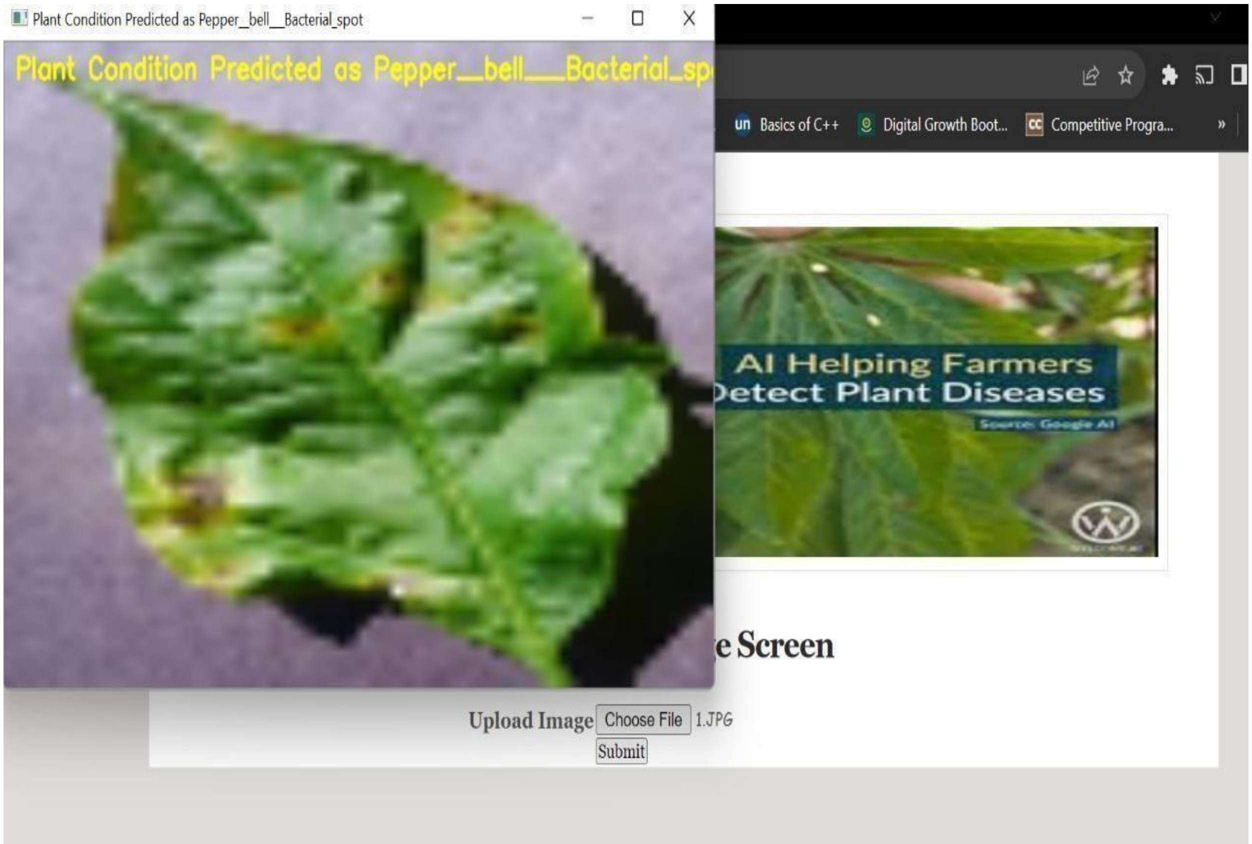


Screenshot 5.4: Filled New User Sign Up Screen.





Screenshot 5.5: Screen with Upload Plant Image and Logout options.



Screenshot 5.6: Screen Displaying Image with Disease Name.

```
Select C:\WINDOWS\system32\cmd.exe
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\Raja Sulakshana\AppData\Roaming\Python\Python37\site-packages\tensorboard\compat\tensorflow_
es.py:550: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a futur
 of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
System check identified no issues (0 silenced).

You have 15 unapplied migration(s). Your project may not work properly until you apply the migrations
s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 11, 2023 - 22:26:51
Django version 2.1.7, using settings 'PlantDisease.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

```
Pepper__bell___Bacterial_spot: rust can be controlled by spraying plants with a copper solution (0.5 to 2.0 oz/ gallon of water) at least four times between late August and late October.
Pepper__bell___healthy: Spray with "captan or sulfur" while disease in early stages.
Potato___Early_blight: For best control, spray liquid copper soap early, two weeks before symptoms normally appear. Alternatively, begin applications when disease first appears, and repeat at 7 to10 day intervals up to blossom drop.
Potato___healthy: Monogenic and polygenic resistance can act together to reduce the severity of NCLB. Crop rotation and tillage practices may be helpful in some cases.
Potato___Late_blight: Spray "mancozeb 2.5 g/I" as soon as pustules appear.
Tomato_Bacterial_spot: Combination of "azoxystrobin and propiconazole, prothioconazole and trifloxystrobin.
Tomato_Early_blight: Use "copper fungicides, or copper plus mancozeb". And growing healthy plants from seed free from infection.
Tomato_healthy: Combination of copper sulfate and lime.
Tomato_Late_blight: Use a copper-based fungicide (mix 2 ounces of fungicide with a gallon of water) every 6 or 7 days following a watering.
Tomato_Leaf_Mold: Use chlorothalonil, menab, mancozeb and copper formulations.
Tomato_Septoria_leaf_spot: "copper and potassium bicarbonate" will help contain the fungal disease and keep it from spreading.
Tomato_Spider_mites_Two_spotted_spider_mite: # spiromesifen (Oberon 2SC): Group 23, mainly affects immature stages # insecticidal soap (M-Pede) # neem oil (Trilogy)
Tomato__Target_Spot: use chlorothalonil, copper oxychloride or mancozeb.
Tomato__Tomato_mosaic_virus: Soak seeds in a 10% solution of trisodium phosphate (Na3PO4) for at least 15 minutes.
Tomato__Tomato_Yellow_Leaf_Curl_Virus: Insecticides of the family of the pyrethroids used as soil drenches or spray during the seeding stage can reduce the population of whiteflies.
Potato___Early_blight: Trichoderma harzianum and Pseudomonas fluorescens are effective against the disease
Potato___Late_blight: spray of "mancozeb at 0.25%" followed by cymoxanil + mancozeb
Grape___Black_rot: Buscalid 25.2%WG and Pyraclostrobin 12.8%WG (or) fosetyl 80.0%WG
Grape___Esca_(Black_Measles): Chlorantranilprole +lambda cyhalortrin(0.5ml/I) or cypermethrin(1ml/l)
Grape___Leaf_blight_(Isariopsis_Leaf_Spot): Ferrous sulphate Fe19%
```

Screenshot 5.7: Screen Displaying Disease Names.

6. TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application . It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit testsensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.3 TEST CASES

6.3.1 CLASSIFICATION

S.NO	Test Case	Excepted Result	Result	Remarks(IF Fails)
1.	User Register	If User registration successfully.	Pass	If already user email exist then it fails.
2.	User Login	If Username and password is correct then it will getting valid page.	Pass	Un Register Users will not logged in.
3.	User View User	Show our dataset	Pass	If Data set Not Available fail.
4.	View Fast History Results	The Four Alarm Score Should be Displayed.	Pass	The Four Alarm Score Not Displaying fail
5.	User Prediction	Display Review with true results	Pass	Results not True Fail
6.	Show Detection Process	Display Detection process	Pass	Results Not True Fail
7.	Show Eye Blink Process	Display Eye Blink Process	Pass	If Results not Displayed Fail.
8.	Admin login	Admin can login with his login credential. If success he get his home page	Pass	Invalid login details will not allowed here
9.	Admin can activate the register users	Admin can activate the register user id	Pass	If user id not found then it won’t login
10.	Results	For our Four models the accuracy and F1 Score	Pass	If Accuracy And F1 Score Not Displayed fail

7. CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

In conclusion, our project on plant disease identification has successfully demonstrated the potential of machine learning and image recognition techniques to accurately detect and diagnose various plant diseases. Through extensive data collection, model training, and testing, we have achieved promising results in terms of disease detection accuracy. This technology holds great promise for farmers and agricultural professionals, as it can help in early disease detection, reducing crop loss, and improving overall crop yield. However, there is room for further refinement and optimization of the models, and future research should focus on scalability and real-world deployment. Overall, our project underscores the importance of leveraging technology to address critical challenges in agriculture and enhance food security.

7.2 FUTURE SCOPE

Real-Time Monitoring: Integrating real-time disease monitoring using IoT devices and sensors can enhance your project's practicality. This would allow farmers to continuously monitor their crops and receive immediate alerts when diseases are detected.

Data Collection and Sharing: Establish a platform for farmers to share disease data and images. This collaborative approach can help create a broader dataset, improve model accuracy, and promote knowledge sharing among agricultural communities.

Localization: Adapt your model for different regions and climates. Plant diseases can vary depending on geographical location, so customizing your system for specific regions can be highly beneficial.

Education and Training: Develop training programs and resources to educate farmers and agricultural workers on how to use your system effectively. Empowering them with the knowledge and skills to identify and manage plant diseases is essential for long-term success.

Machine Learning Advancements: Keep up with advancements in machine learning and computer vision technologies. Incorporate state-of-the-art algorithms and techniques to improve the accuracy and efficiency of disease identification.

Environmental Monitoring: Consider incorporating environmental factors such as weather conditions, soil quality, and pest populations into your system. These factors can influence disease outbreaks and crop health.

8. BIBLIOGRAPHY

8.BIBLIOGRAPHY

8.1 REFERENCES

[1] Lee SH, Chan CS, Mayo SJ, Remagnino P. How deep learning extracts and learns leaf features for plant classification. Pattern Recogn. 2017;71:1–13.

[2] Kumar S, Singh SK. Occluded thermal face recognition using bag of CNN(BoCNN). IEEE Signal Process Lett. 2020;27:975–9.

[3] Kumar S, Kaur R. Plant disease detection using image processing—a review. Int J Comput Appl. 2015;124(2):6–9.

8.2 GITHUB LINK

https://github.com/Sulakshana3007/Plant-Disease-Detection.git