Artificial Intelligence and Machine Learning

Project Report

Semester-IV (Batch 2022)

IndiGro: Predict. Prevent. Prosper.

Supervised By:

Ms. Shagun Sharma

Submitted By:

Japneet Kaur	2210990433	G-10
Japneet Singh	2210990434	G-10
Jasneet Arora	2210990451	G-10

Department of Computer Science and Engineering
Chitkara University Institute of Engineering & Technology,
Chitkara University, Punjab

Abstract

Plant diseases represent a significant threat to global food security, with the potential to cause substantial losses in agricultural productivity and economic revenue. Timely and accurate detection of these diseases is crucial for implementing effective disease management strategies and ensuring sustainable crop production. In recent years, deep learning methodologies, particularly Convolutional Neural Networks (CNNs), have emerged as powerful tools for automating disease detection processes. CNNs have demonstrated remarkable capabilities in analyzing complex visual data, making them well-suited for tasks such as image classification and object detection.

This project proposes a comprehensive approach to plant disease detection by leveraging CNNs. The primary objective is to develop a robust and efficient model capable of accurately identifying various types of plant diseases from images of leaves. The proposed model architecture is designed to effectively learn discriminative features from input images, enabling it to distinguish between healthy and diseased plants with high accuracy. By harnessing the power of deep learning, the system aims to streamline the disease detection process, enabling farmers and agricultural practitioners to detect and address diseases in their crops more effectively.

In addition to disease detection, the project integrates a chatbot developed using Streamlit and the Gemini API into the system. This chatbot serves as a valuable resource for users by providing assistance in locating potential cures or treatments for the detected diseases. The chatbot can analyze user queries and provide relevant information on disease management strategies, recommended treatments, and preventive measures. This integration not only streamlines the disease management process but also empowers users with valuable insights and resources to combat plant diseases effectively and sustainably.

TABLE OF CONTENTS

1. Introduction

1.1 Introduction of Plant Disease Detection

The most important sector of our economy is agriculture. Various types of diseases damage the plant leaves and affect the production of crops; therefore, leaf disease detection is important. Regular maintenance of plant leaves results in profitability in agricultural products. Farmers do not have expertise in leaf diseases, thus leading to a lack of production. Leaf disease detection is crucial because profit and loss depend on production. Therefore, deep learning techniques are used here to detect diseases in plant leaves. This dataset contains thirty-eight types of leaf diseases, with 35,406 leaf images used to train the model. All images are resized to 128 x 128, and they are divided into three parts: training, testing and validation datasets.

1.Apple Scab 2.Grape Esca 3.Corn Leaf Spot

Fig. 1. Leaves with Disease Part

In Fig. 1, we can see vegetable and fruit leaves such as apple, grape and corn with diseased parts. This disease can be easily detected using deep learning techniques. The disease is detected using a convolutional neural network (CNN).

To enhance user experience and streamline disease management, this project integrates a chatbot into the system, providing real-time guidance and equipping users with invaluable resources to combat plant diseases efficiently and sustainably.

1.2 Applications

- Biological research
- Plant leaf disease detection also useful in agriculture institute.
- Some plant leaf disease detection automatic techniques are beneficial for large work of monitoring in farm of crops disease detection.

1.3 Objectives

This project aims to focus on plant leaf disease detection utilizing Convolutional Neural Networks (CNN). CNN technology is employed to analyze both healthy and diseased plant leaves, enhancing agricultural productivity and profitability by enabling early disease identification. Additionally, the integration of a chatbot into the system aims to augment user experience by providing valuable assistance in managing plant diseases effectively. Developed using Streamlit and the Gemini API, the chatbot offers guidance on potential cures or treatments for detected diseases, while also delivering information on disease management strategies, recommended treatments, and preventive measures in response to user queries. This integration streamlines the disease management process, equipping users with valuable insights and resources to combat plant diseases efficiently and sustainably.

1.4 Motivation

The motivation behind this project is to combat leaf diseases in agriculture, crucial for preserving crop yield and profitability. By accurately identifying and recognizing these diseases, we can prevent significant reductions in crop yield, ensuring agricultural productivity. Additionally, this initiative benefits agricultural institutes and biological research by advancing our understanding of plant diseases and fostering the development of effective management strategies. Moreover, the integration of a chatbot into the system aims to enhance user experience by providing real-time guidance on disease management, streamlining the process and equipping users with valuable resources to combat plant diseases efficiently and sustainably.

2. Problem Definition

Agriculture, vital for economic sustenance, faces significant challenges from leaf diseases, hindering crop productivity. Lack of efficient detection methods undermines profitability and sustainability, with labor-intensive conventional approaches prone to inaccuracies. Farmers often lack expertise, leading to yield losses. This project aims to develop an automated solution using deep learning techniques, particularly CNNs, to swiftly and accurately detect leaf diseases, empowering farmers with actionable insights for crop health optimization. Additionally, integrating a chatbot with Streamlit and Gemini API offers users access to potential cures or treatments, streamlining disease management and enhancing agricultural sustainability.

3. Proposed Design and Methodology

3.1 Methodology

The main aim is to design a system which is efficient and which provide disease name. For that purpose, we use two phases: 1st is training phase and 2nd is testing phase. In 1st phase: Image acquisition, Image Pre-processing and CNN based training. In 2nd phase: Image acquisition, Image Pre-processing, Classification and disease identification.

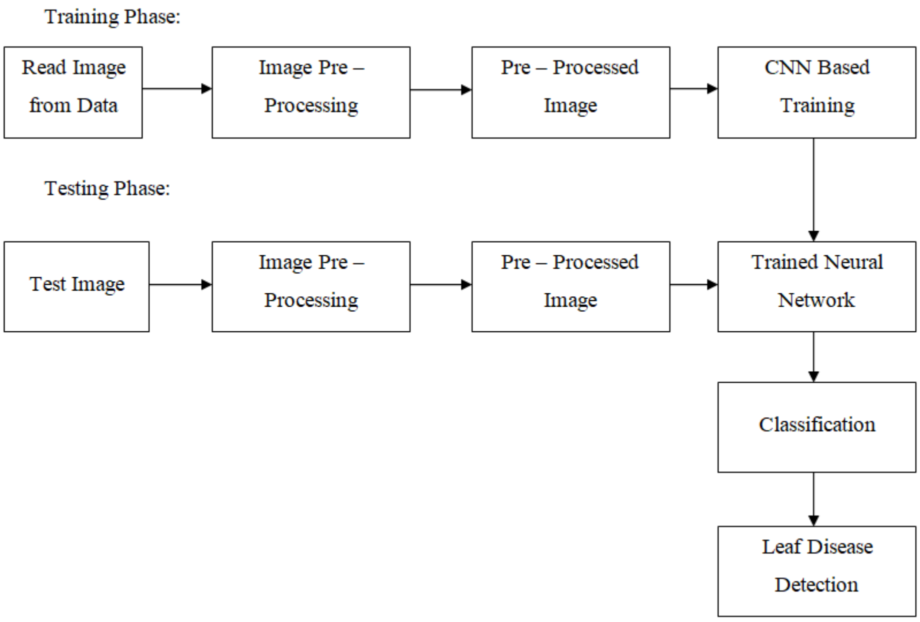

Fig. 2. Work Flow

Using Streamlit and the Gemini API, the chatbot can assist in disease detection and subsequently search for potential cures or treatments for the identified illness, streamlining the process and providing valuable assistance to users in need.

3.2 Tools and Technologies

3.2.1 Python

Python's simplicity and versatility drive its use in web development, scientific computing, AI, and automation, aided by rich libraries.

3.2.2 NumPy

Python library for numerical computing with arrays. Offers efficient operations, linear algebra, random simulations, and integration with C/Fortran.

3.2.2 Pandas

Pandas, a Python library, streamlines data manipulation and analysis with DataFrames, widely adopted in data science and finance, integrating seamlessly with NumPy and Matplotlib. Enhances analysis and visualization.

3.2.3 Matplotlib

Matplotlib, a Python library, crafts static, animated, interactive visualizations with extensive customization for research, data analysis, and presentations, integrated with NumPy and pandas.

3.2.4 Seaborn

Seaborn, a Python visualization library, enhances Matplotlib with high-level tools for creating appealing statistical graphics. Simplifying plot creation, it's ideal for exploring complex datasets, integrating seamlessly with Pandas DataFrames.

3.2.5 Scikit – Learn

Scikit-learn, a Python library for machine learning, provides intuitive tools for classification, regression, and clustering, simplifying workflows with preprocessing and evaluation. Widely favored for efficiency and community support.

3.2.6 TensorFlow

TensorFlow, an open-source machine learning framework, facilitates building complex neural network architectures with CPU and GPU computing support. Its flexibility, scalability, and extensive documentation make it popular across machine learning domains.

3.2.7 Keras

Keras, an open-source deep learning framework, offers a high-level API for neural network building and training, simplifying prototyping and experimentation. Supports TensorFlow, Theano, enabling flexibility for research and industry.

3.2.8 OpenCV

OpenCV is an open-source computer vision and machine learning library, offering tools for image and video analysis, object recognition, and more. Supporting multiple languages, it's widely used across diverse fields.

3.2.9 Streamlit

Streamlit, a Python library, facilitates interactive web app development for data science and machine learning. It enables Python scripts to generate interactive visualizations and dashboards in the browser.

4. Results

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 128, 128, 32)	896
conv2d_1 (Conv2D)	(None, 126, 126, 32)	9,248
max_pooling2d (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_2 (Conv2D)	(None, 63, 63, 64)	18,496
conv2d_3 (Conv2D)	(None, 61, 61, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_4 (Conv2D)	(None, 30, 30, 128)	73,856
conv2d_5 (Conv2D)	(None, 28, 28, 128)	147,584
max_pooling2d_2 (MaxPooling2D)	(None, 14, 14, 128)	0
conv2d_6 (Conv2D)	(None, 14, 14, 256)	295,168
conv2d_7 (Conv2D)	(None, 12, 12, 256)	590,080
max_pooling2d_3 (MaxPooling2D)	(None, 6, 6, 256)	0
conv2d_8 (Conv2D)	(None, 6, 6, 512)	1,180,160
conv2d_9 (Conv2D)	(None, 4, 4, 512)	2,359,808
max_pooling2d_4 (MaxPooling2D)	(None, 2, 2, 512)	0
dropout (Dropout)	(None, 2, 2, 512)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 1500)	3,073,500
dropout_1 (Dropout)	(None, 1500)	0
dense_1 (Dense)	(None, 38)	57,038

Fig. 3. Sequential Model Summary

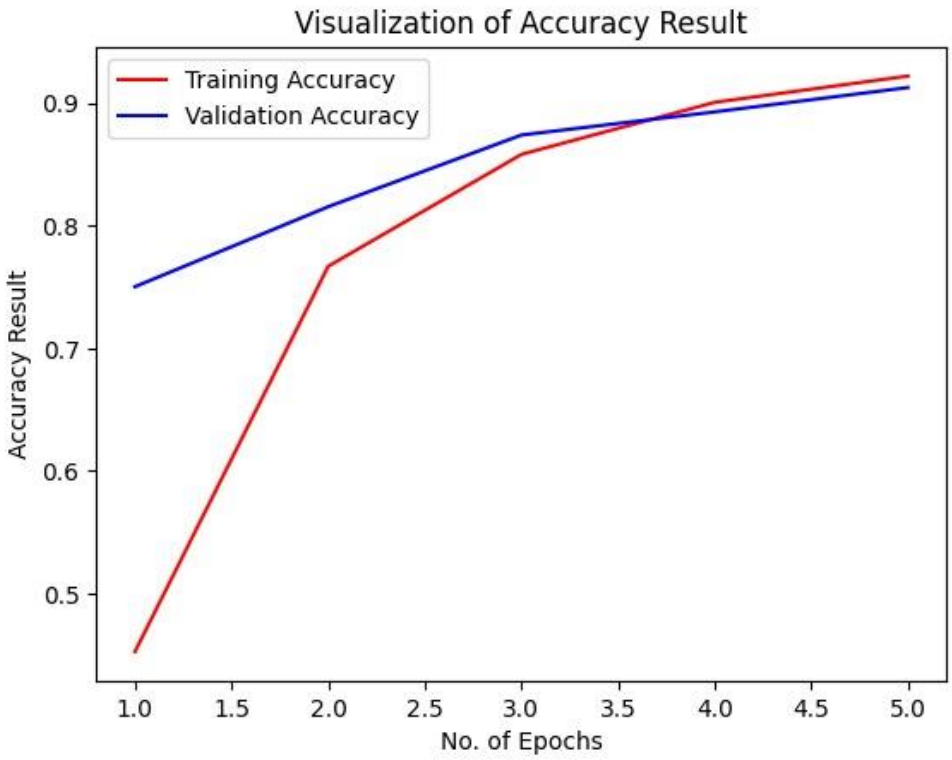

Fig. 4. Visualization of Accuracy Result

	precision	recall	f1-score	support
Apple___Apple_scab	0.94	0.80	0.87	504
Apple___Black_rot	0.96	0.96	0.96	497
Apple___Cedar_apple_rust	0.95	0.93	0.94	440
Apple___healthy	0.90	0.91	0.91	502
Blueberry___healthy	0.93	0.92	0.93	454
Cherry_(including_sour)___Powdery_mildew	0.99	0.92	0.95	421
Cherry_(including_sour)___healthy	0.93	0.95	0.94	456
Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot	0.92	0.84	0.88	410
Corn_(maize)___Common_rust_	0.97	0.98	0.97	477
Corn_(maize)___Northern_Leaf_Blight	0.87	0.96	0.92	477
Corn_(maize)___healthy	0.99	1.00	0.99	465
Grape___Black_rot	0.92	0.95	0.94	472
Grape___Esca_(Black_Measles)	0.97	0.94	0.96	480
Grape___Leaf_blight_(Isariopsis_Leaf_Spot)	0.98	0.98	0.98	430
Grape___healthy	0.98	0.96	0.97	423
Orange___Haunglongbing_(Citrus_greening)	0.91	0.98	0.94	503
Peach___Bacterial_spot	0.92	0.86	0.89	459
Peach___healthy	0.98	0.96	0.97	432
Pepper,_bell___Bacterial_spot	0.85	0.86	0.86	478
Pepper,_bell___healthy	0.99	0.76	0.86	497
Potato___Early_blight	0.96	0.89	0.92	485
Potato___Late_blight	0.68	0.98	0.81	485
Potato___healthy	0.94	0.94	0.94	456
...				
accuracy			0.91	17572
macro avg	0.92	0.91	0.91	17572
weighted avg	0.92	0.91	0.91	17572

Fig. 5. Classification Report

```
['Apple___Apple_scab',
 'Apple___Black_rot',
 'Apple___Cedar_apple_rust',
 'Apple___healthy',
 'Blueberry___healthy',
 'Cherry_(including_sour)___Powdery_mildew',
 'Cherry_(including_sour)___healthy',
 'Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot',
 'Corn_(maize)___Common_rust_',
 'Corn_(maize)___Northern_Leaf_Blight',
 'Corn_(maize)___healthy',
 'Grape___Black_rot',
 'Grape___Esca_(Black_Measles)',
 'Grape___Leaf_blight_(Isariopsis_Leaf_Spot)',
 'Grape___healthy',
 'Orange___Haunglongbing_(Citrus_greening)',
 'Peach___Bacterial_spot',
 'Peach___healthy',
 'Pepper,_bell___Bacterial_spot',
 'Pepper,_bell___healthy',
 'Potato___Early_blight',
 'Potato___Late_blight',
 'Potato___healthy',
 'Raspberry___healthy',
 'Soybean___healthy',
...
 'Tomato___Spider_mites Two-spotted_spider_mite',
 'Tomato___Target_Spot',
 'Tomato___Tomato_Yellow_Leaf_Curl_Virus',
 'Tomato___Tomato_mosaic_virus',
 'Tomato___healthy']
```

Fig. 6. Training Set Class Names

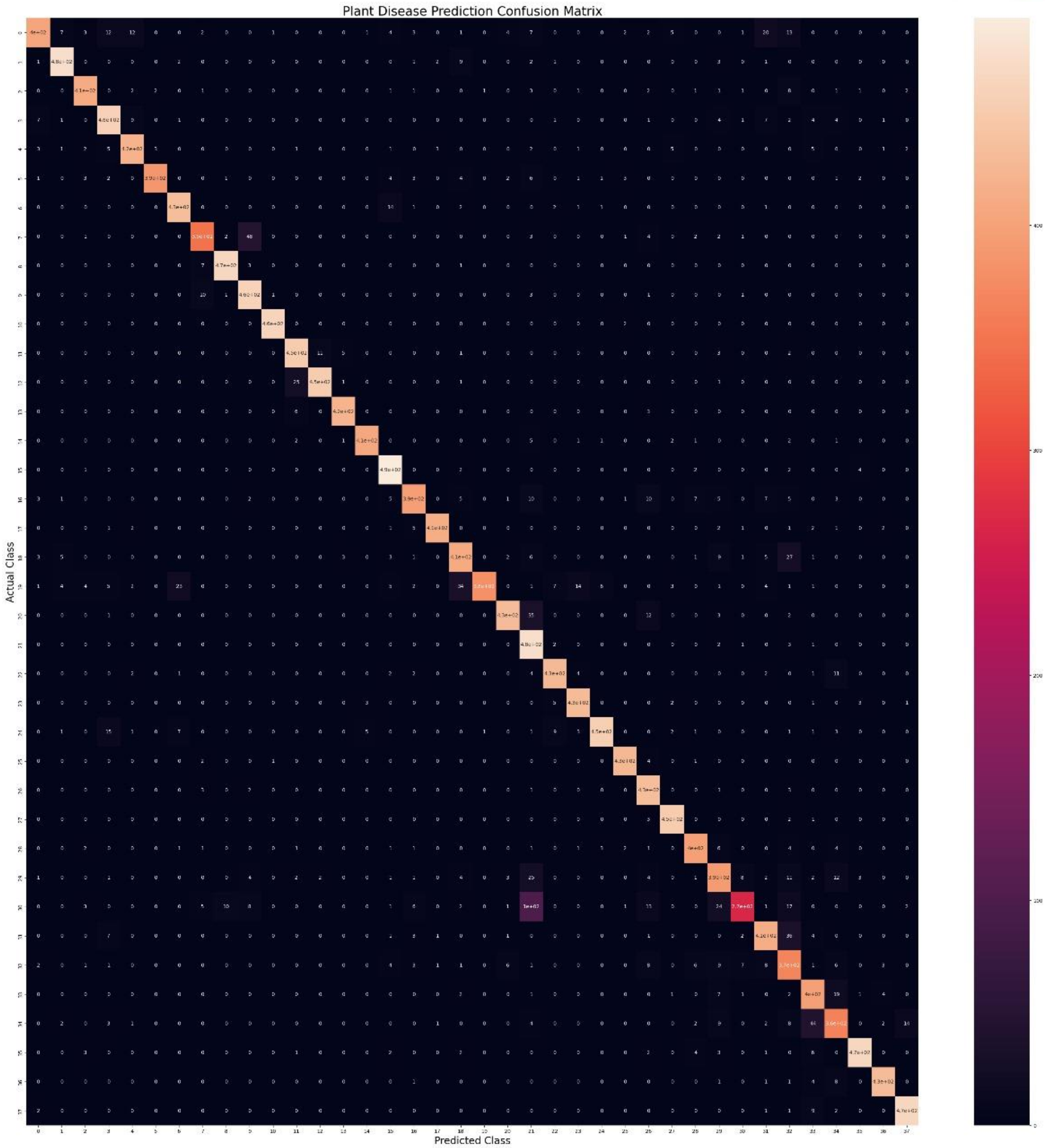

Fig. 7. Plant Disease Prediction Confusion Matrix

```python
# Displaying image
plt.imshow(img)
plt.title('Test Image')
plt.xticks([])
plt.yticks([])
plt.show()
```

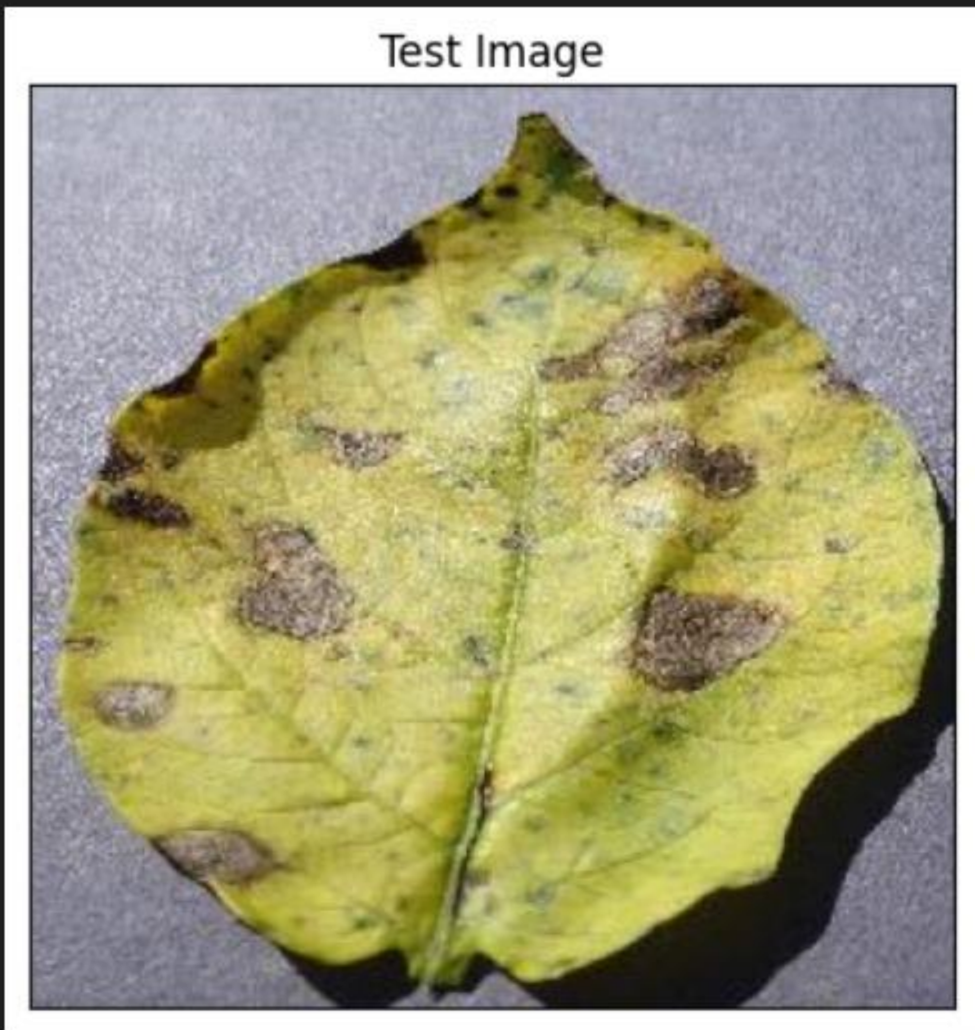

Fig. 8. Displaying Test Image

Displaying Result of Disease Prediction

```
model_prediction = class_name[result_index]
plt.imshow(img)
plt.title(f"Disease Name: {model_prediction}")
plt.xticks([])
plt.yticks([])
plt.show()
```

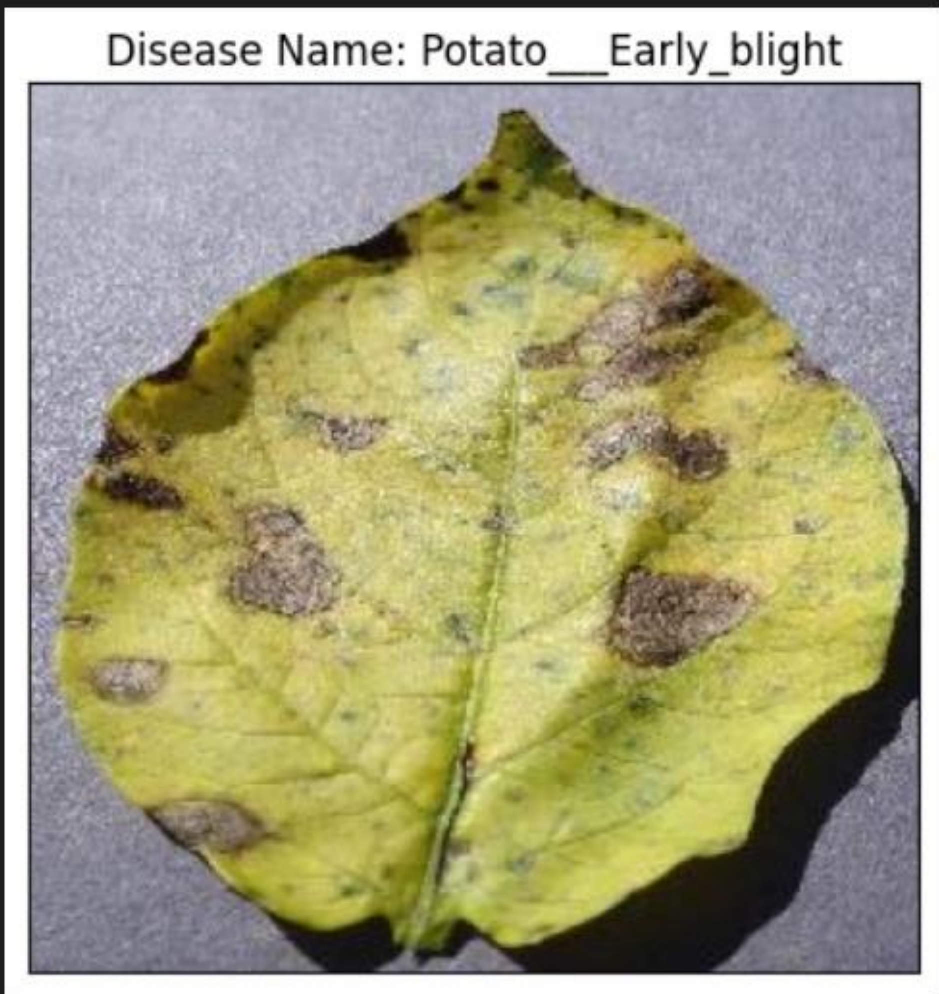

Fig. 9. Displaying Result of Disease Prediction

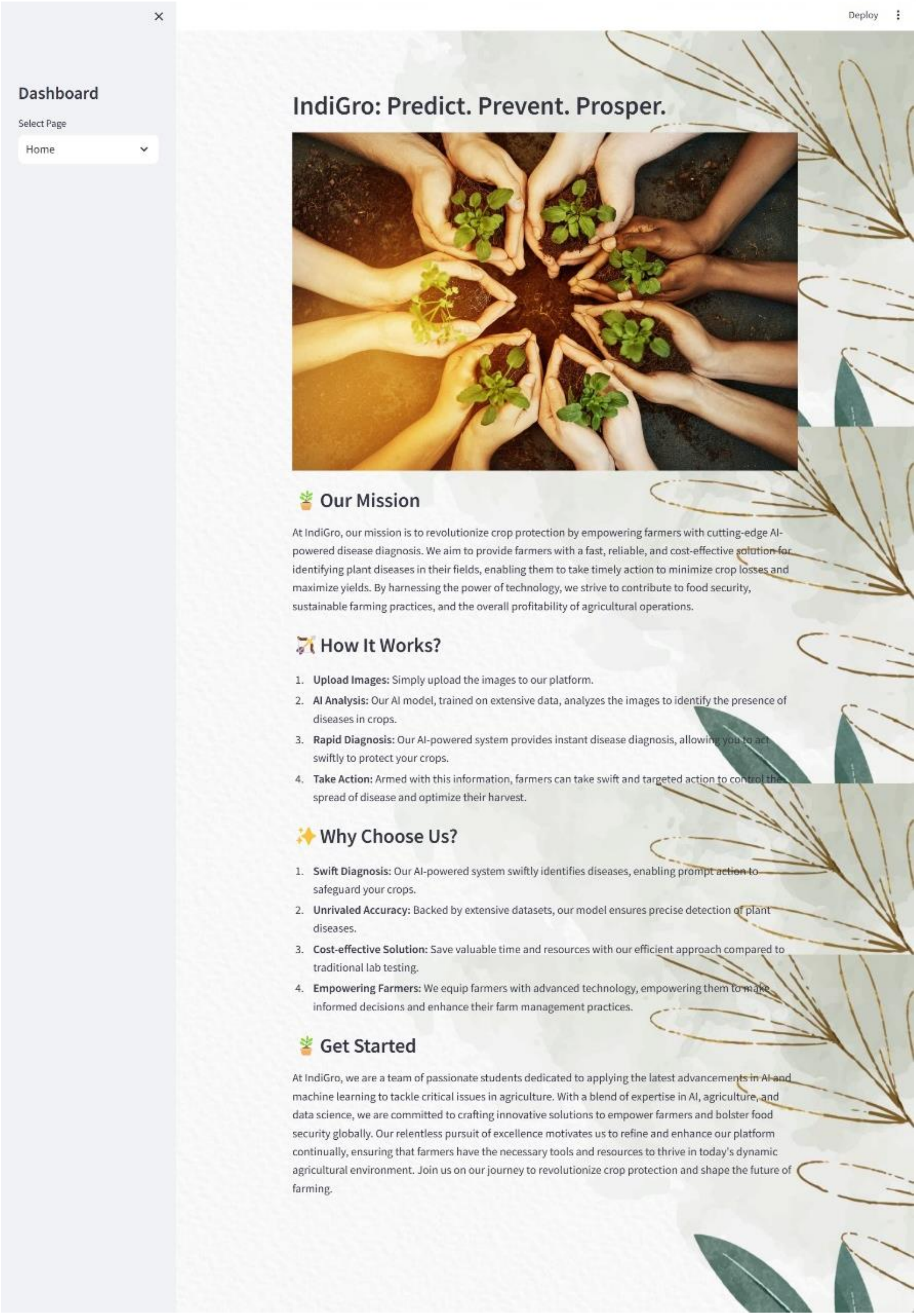

Fig. 10. Home Page

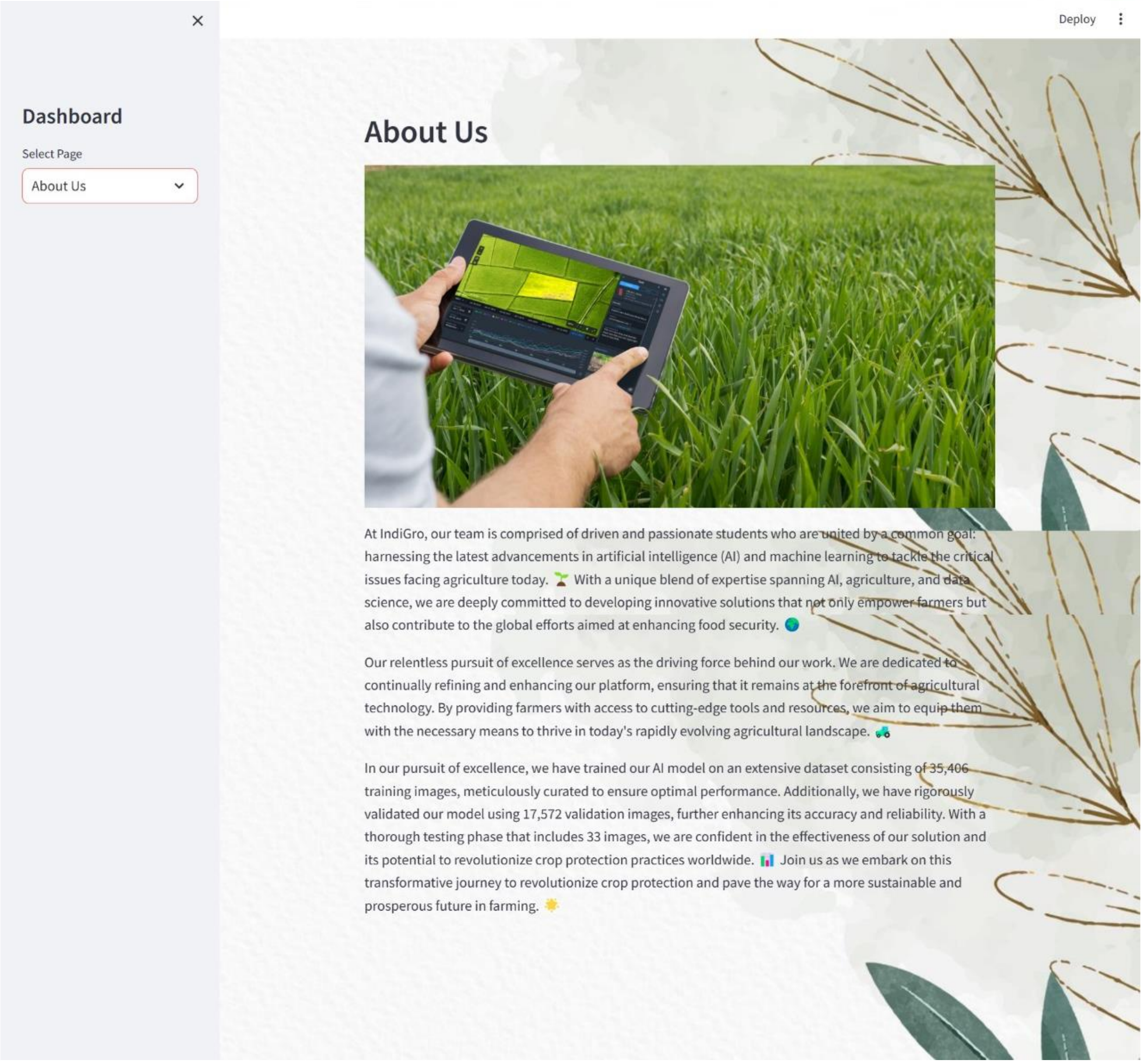

Fig. 11. About Us Page

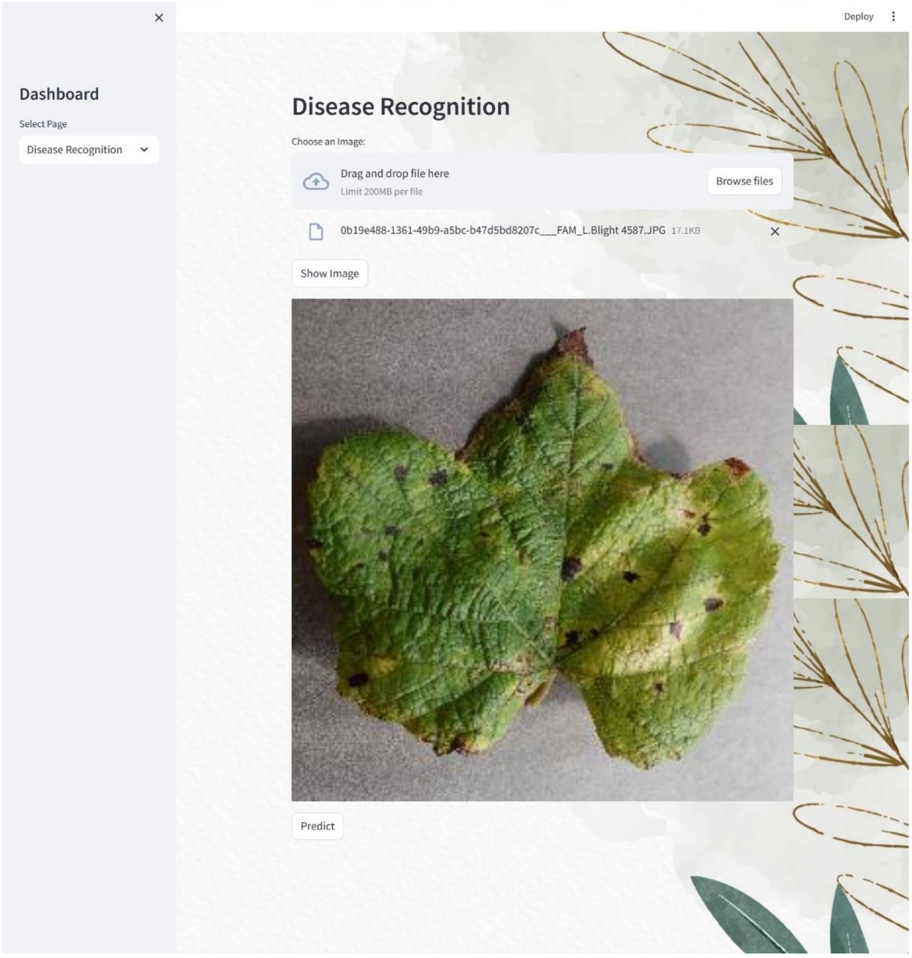

Fig.12. Disease Recognition

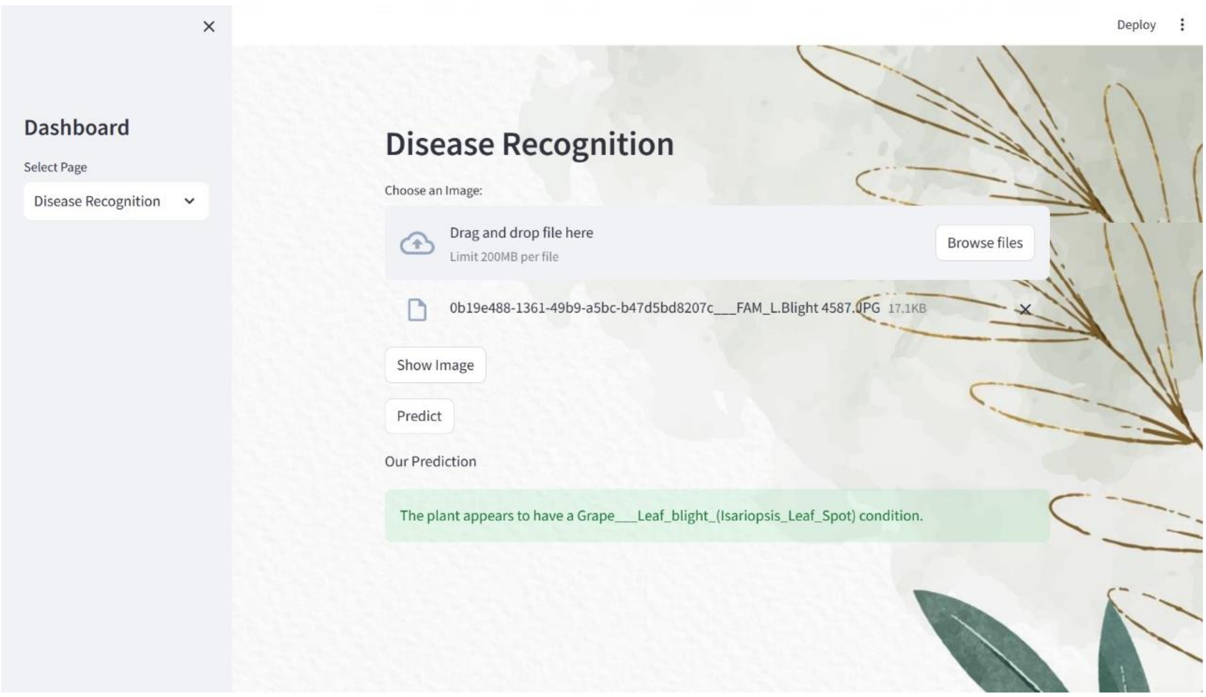

Fig. 13. Disease Prediction

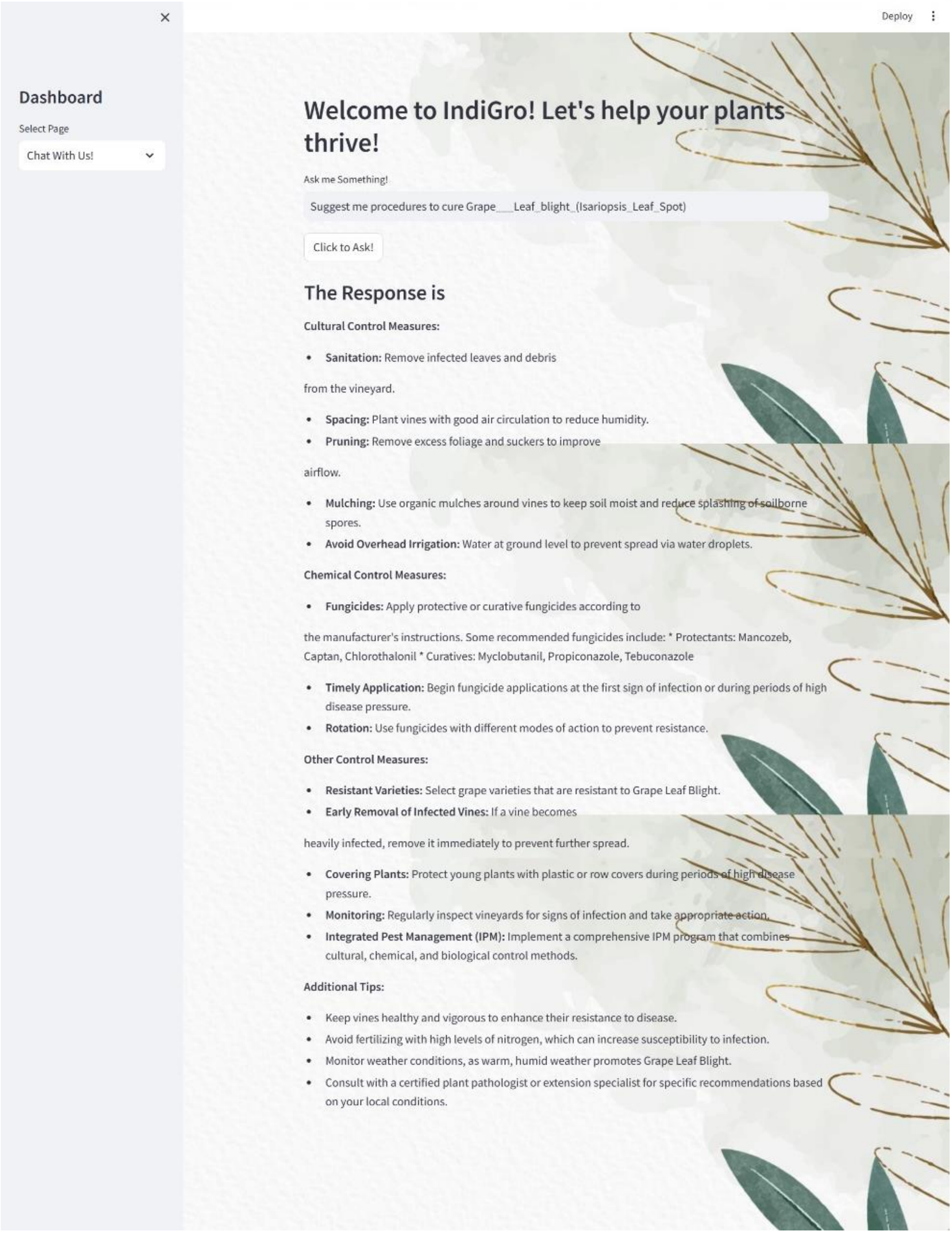

Fig. 14. Chat With Us!

5. References

5.1 Websites

- YouTube: https://www.youtube.com/
- Python Docs: https://www.python.org/
- Pandas Docs: https://pandas.pydata.org/
- Matplotlib Docs: https://matplotlib.org/
- Seaborn Docs: https://seaborn.pydata.org/
- Scikit-Learn Docs: https://scikit-learn.org/
- TensorFlow Docs: https://www.tensorflow.org/
- Keras Docs: https://keras.io/
- OpenCV Docs: https://opencv.org/
- Stremlit Docs: https://streamlit.io/
- GitHub: https://github.com/

5.2 Research Papers

- Leaf Disease Detection Using Machine Learning: https://www.researchgate.net/publication/344282301_Leaf_Disease_Detection_Using_Machine_Learning
- Plant Disease Detection Using ML: https://www.scribd.com/document/518098023/Project-Report

5.3 Dataset

- New Plant Diseases Dataset: https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset