**NAME:** GOMATHAM JOSHIKA SIRI CHANDANA

**AICTE Student ID:** STU67dc275d630e31742481245

**AICTE Internship ID**: INTERNSHIP\_174099535967c57b1f336c3

**Edunet Foundation, in collaboration with AICTE & Shell-**

**on Green Skills using AI technologies.**

**Plant Disease Detection System for Sustainable Agriculture**

**Problem Statement**

Design and implement a Convolutional Neural Network (CNN)-based system for automatic detection and classification of plant diseases using leaf images from crops such as apple, cherry, grape, and corn. The model should distinguish between healthy and diseased leaves and accurately identify specific types of diseases. This solution aims to support sustainable agriculture by facilitating early diagnosis, minimizing crop loss, and promoting timely and targeted disease management practices.

# Plant Disease Detection System Pipeline

## Step 1: Data Collection & Loading

- Collect images of plant diseases and organize them into categories (e.g., healthy, diseased)  
- Divide the dataset into three folders:  
 - Train: contains images used for training the model  
 - Test: contains images used for evaluating the final performance of the model  
 - Valid: contains images used for tuning the model's hyperparameters  
- Each folder contains subfolders for different categories of plant diseases

## Step 2: Model Setup

- Use the training set to train a machine learning model  
- Use the validation set to tune the model's hyperparameters and prevent overfitting  
- Use the test set to evaluate the final performance of the model

## Step 3: ZIP & Upload

- Zip the dataset folder containing the train, test, and valid sets  
- Upload the zipped dataset to Google Drive  
- Mount the Google Drive in Google Colab and unzip the dataset  
- Load the dataset using Python code in Google Colab

## Step 4: Image Processing & Augmentation

- Resize all images to a uniform size (e.g., 128x128) to ensure consistency  
- Apply image augmentation techniques to increase the diversity of the training data, such as:  
 - Flipping (horizontal and vertical)  
 - Rotation (random angles)  
 - Color jittering (brightness, contrast, saturation)  
 - Other transformations (e.g., scaling, cropping)

## Step 5: CNN Model

- Use a Convolutional Neural Network (CNN) architecture to train on the processed data  
- CNNs are suitable for image classification tasks due to their ability to extract features from images  
- Train the model using the training set and evaluate its performance on the validation set

## Step 6: Test/Evaluate

- Use the test set to evaluate the final performance of the model  
- Metrics for evaluation may include:  
 - Accuracy  
 - Precision  
 - Recall  
 - F1-score  
 - Confusion matrix

## Step 7: Deployment

- Wrap the trained model in a Python script for deployment or operation  
- The script can be used to:  
 - Load new images  
 - Preprocess the images  
 - Make predictions using the trained model  
 - Output the predicted class label or probability distribution

By following the Plant Disease Detection System Pipeline, we develop a robust system using deep learning techniques. This system enables accurate disease detection in plants, providing valuable insights for farmers, researchers, and agricultural professionals.