from tensorflow.keras.preprocessing.image import load\_img, img\_to\_array import numpy as np

# **Define your class labels (adjust as per your model)**

CLASS\_NAMES = ['Healthy', 'Leaf Spot', 'Blight', 'Rust']

def preprocess(file): """ Preprocess uploaded image for model prediction. - Resize to match model input - Normalize pixel values - Expand dimensions for batch input """ img = load\_img(file, target\_size=(224, 224)) # adjust if your model uses a different size img\_array = img\_to\_array(img) / 255.0 # normalize to [0, 1] return np.expand\_dims(img\_array, axis=0) # shape: (1, 224, 224, 3)

def decode\_prediction(pred): """ Convert model output to human-readable label and confidence. """ class\_index = np.argmax(pred) confidence = float(np.max(pred)) label = CLASS\_NAMES[class\_index] return { 'label': label, 'confidence': round(confidence \* 100, 2) # percentage format }