1.Project Overview:

GreenClassify is a deep learning-based web application that classifies vegetable images using a Convolutional Neural Network (CNN).

The system allows users to:

1. Upload a vegetable image
2. Analyze the image using a trained CNN model
3. Receive prediction results with confidence percentage

The model is trained using TensorFlow/Keras and deployed using Flask.

2.Project Workflow:

System Working Process:

1. User selects an image.
2. Image is uploaded to Flask server.
3. Image is resized to 128x128 pixels.
4. Image is normalized (pixel values scaled between 0–1).
5. Pre-trained CNN model analyzes the image.
6. Model predicts the vegetable class.
7. System displays:
    - Predicted vegetable name
    - Confidence percentage (%)

3.Technologies Used:

Category		Technology
1.	Deep Learning	Tensorflow
2.	Model Building	Keras
3.	Image Processing	NumPy
4.	Frontend	HTML,CSS
5.	ModelType	Convoltional Neural NetWrok

4.Required Libraries:

1.Flask

2.tensorflow

3.numpy

5.Requirements.txt:

1.Flask==2.3.2

2.tensorflow==2.15.0

3.numpy==1.24.3

6.Project Folder Structure:

GREENCLASSIFY/

```
│
├── models/
│   └── vegetable_model.h5
│
├── code/
│   └── Vegetable Images/
│       ├── train/
│       └── validation/
│
├── static/
├── templates/
├── uploads/
│
├── app.py
├── train.py
├── evaluate.py
├── requirements.txt
└── README.md
```

7.Model Architecture:

The CNN architecture used:

1. Conv2D (16 filters)
2. MaxPooling2D
3. Conv2D (32 filters)
4. MaxPooling2D
5. Conv2D (64 filters)
6. MaxPooling2D
7. Flatten
8. Dense (128 neurons)
9. Output Layer (Softmax)

8.Activation Functions:

- ReLU (hidden layers)
- Softmax (output layer)

9.Optimizer:

- Adam (learning rate = 0.0001)

10.Loss Function:

- Categorical Crossentropy

11.Image Preprocessing:

- Resize image to 128x128
- Convert to array
- Normalize pixel values (divide by 255)
- Expand dimensions for model input

12.Model Training Details:

Parameter	Value
1. ImageSize	128*128
2. BatchSize	32
3. Epochs	3
4. Optimizer	Adam
5. Augmentation	Zoom, Shear, Flip

13.Prediction Output:

After image upload:

Predicted Vegetable Name

Confidence Score (e.g., 92.45%)

14.How to Run the Project:

Step 1: Install Dependencies

pip install -r requirements.txt

Step 2: Train the Model (Optional)

python train.py

Step 3: Run Flask App

python app.py

Step 4: Open Browser

http://127.0.0.1:5000

15.Key Features:

- User-friendly interface
- Real-time image classification
- Confidence percentage output
- Deep learning-based approach
- Lightweight CNN model

16.Future Improvements:

- Increase dataset size
- Train with more epochs
- Use Transfer Learning (MobileNet, ResNet)
- Deploy on cloud (Render, AWS, Azure)
- Add multiple image prediction
- Add history tracking

17.Advantages:

- Fast prediction
- Easy to use
- Works on custom uploaded images
- Lightweight deployment

18.Limitations:

- Limited dataset size
- Accuracy depends on training data quality
- Basic CNN (not transfer learning)
- Only vegetable classification

19.Conclusion:

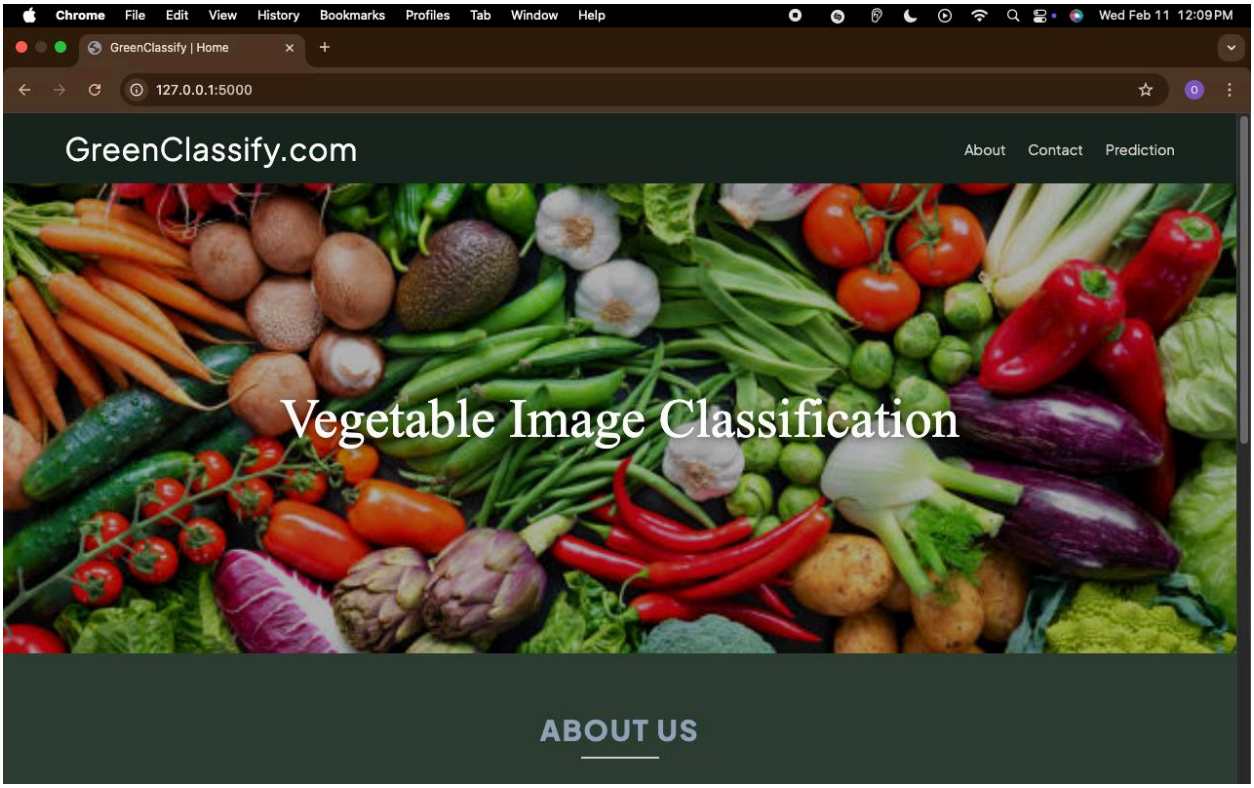
GreenClassify demonstrates how deep learning and web technologies can be combined to build an intelligent vegetable classification system.

The model analyzes uploaded images and predicts the vegetable type with a confidence percentage, making it useful for:

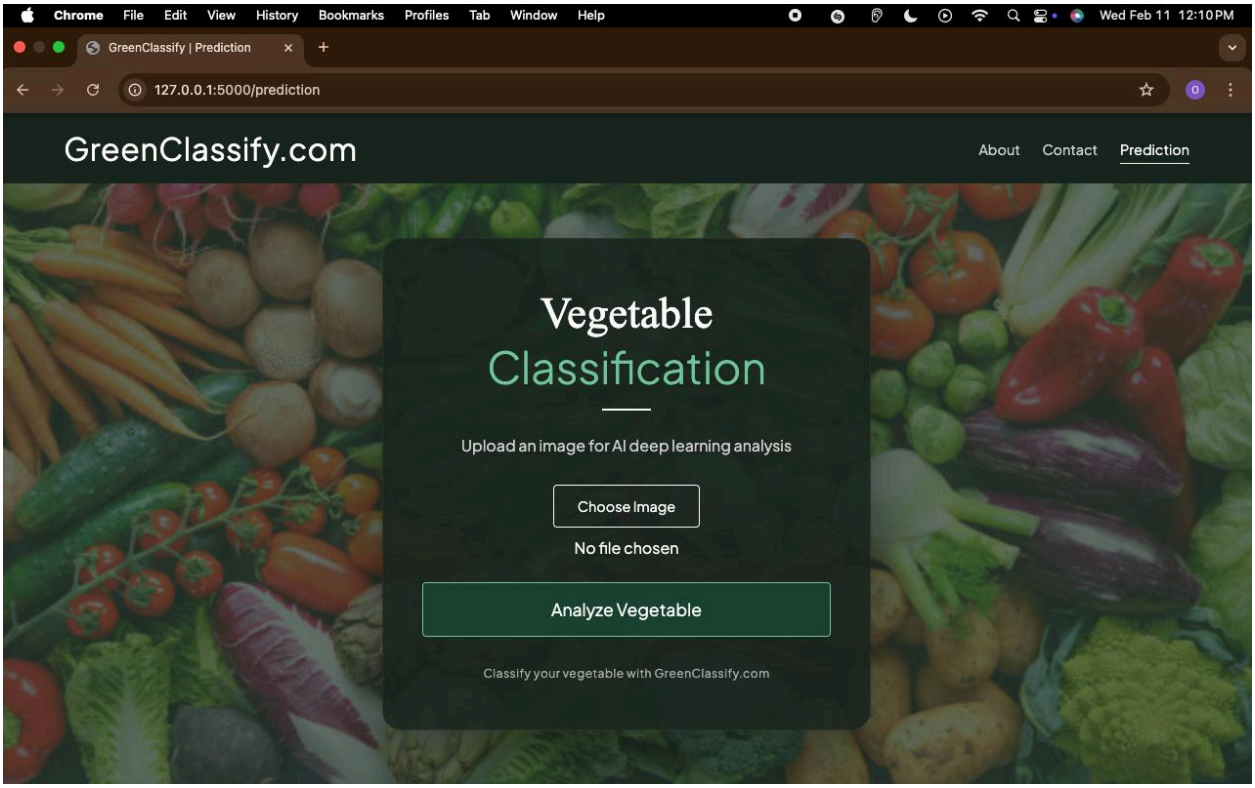
- Agriculture support
- Smart farming
- Retail automation
- Educational purposes

Output Images:

Home page:



Choose Image:



Analyze Image: