Project Report: Smart Sorting Transfer Learning for Identifying Rotten Fruits and Vegetables

---

1. INTRODUCTION

Project Title:

Smart Sorting Transfer Learning for Identifying Rotten Fruits and Vegetables

Team Members:

Team Leader : Jadi Lathish

Team member: Siddanatham Sathya Sai

Team member: B V Harshavardhan

Team member: B Pavan Kumar

---

2. PROJECT OVERVIEW

Purpose:

This project aims to reduce food waste and improve quality assurance in agricultural supply chains by using AI to automatically detect rotten fruits and vegetables.

Features:

Image upload feature

AI model for freshness classification

Web-based user interface

Real-time prediction and result display

---

3. ARCHITECTURE

Frontend:

Developed using HTML, CSS, and JavaScript (served from static folder) for a responsive UI. HTML templates are stored in the templates folder and rendered using Flask.

Backend:
Built with Python using Flask. Handles routing, model inference, and image processing. Core logic resides in app.py, and the CNN model logic is encapsulated in cnn.py.

Database:

Currently, no persistent database is used. Optionally, a lightweight database like SQLite or MongoDB can be integrated for logging predictions.

---

4. SETUP INSTRUCTIONS

Prerequisites:

Python 3.x

Flask

TensorFlow/Keras

Installation:

Clone the repository: git clone https://github.com/your-repo.git

Navigate to the project directory: cd your-repo

Install dependencies: pip install -r requirements.txt

Place your model file as model.h5 in the project root

Ensure folders media, static, and templates are properly populated

Run the app: python app.py

---

5. FOLDER STRUCTURE

media/ – contains uploaded images

static/ – contains CSS and JavaScript files

templates/ – contains HTML files rendered by Flask

app.py – main Flask application file

cnn.py – defines the model loading and prediction logic

model.h5 – pre-trained CNN model

---

6. RUNNING THE APPLICATION

Flask Backend (also serves frontend):
Run the following command:

python app.py

Navigate to http://127.0.0.1:5000 in your browser to use the app.

---

7. API DOCUMENTATION

POST /predict

Response:

```
{
  "status": "success",
  "prediction": "Fresh"
}
```

---

8. AUTHENTICATION

Currently, this project does not use authentication. It is planned as a future enhancement.

---

9. USER INTERFACE

Upload Image Page

Prediction Result Display

It should be and image with a bar to select images once you selectd an another press predict and you will see results

---

10. TESTING

Strategy:

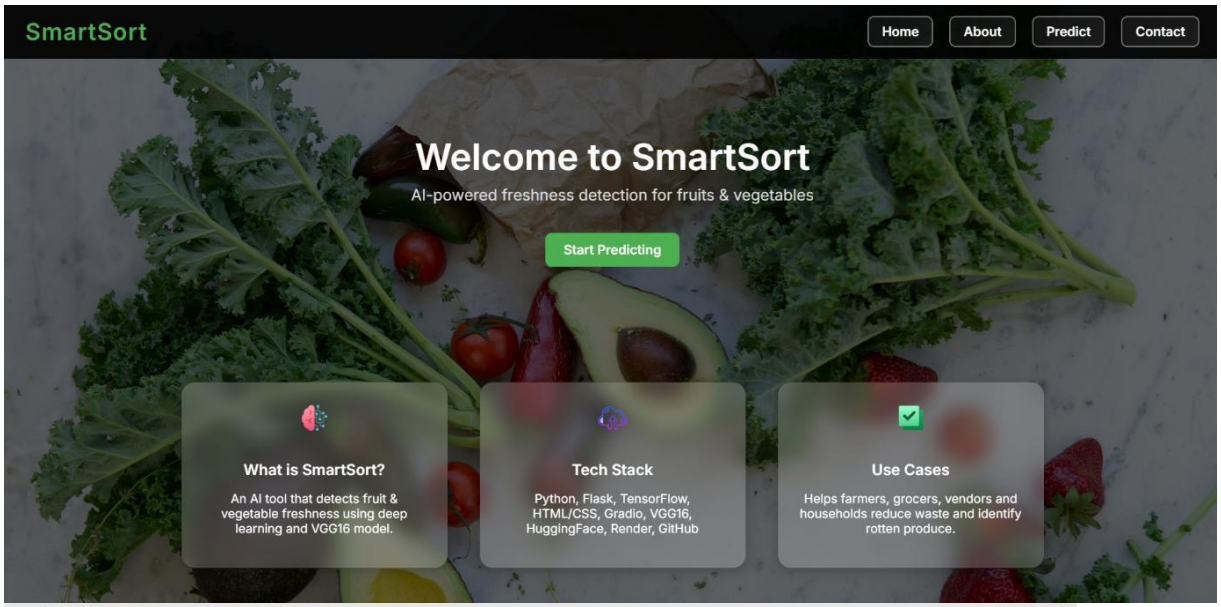
Manual testing of UI interactions

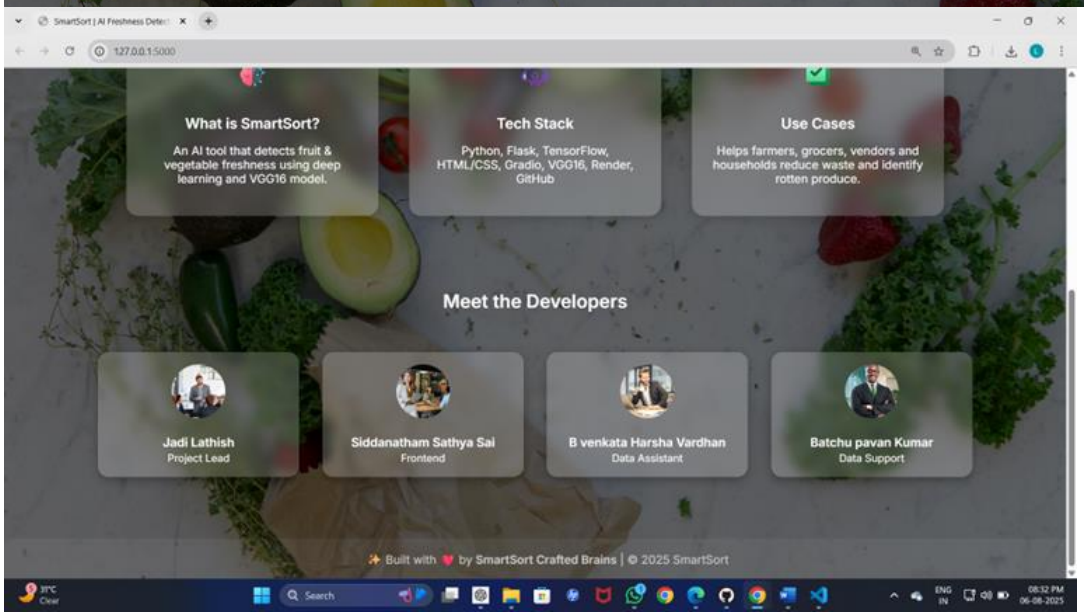
Unit testing using unittest for model functions

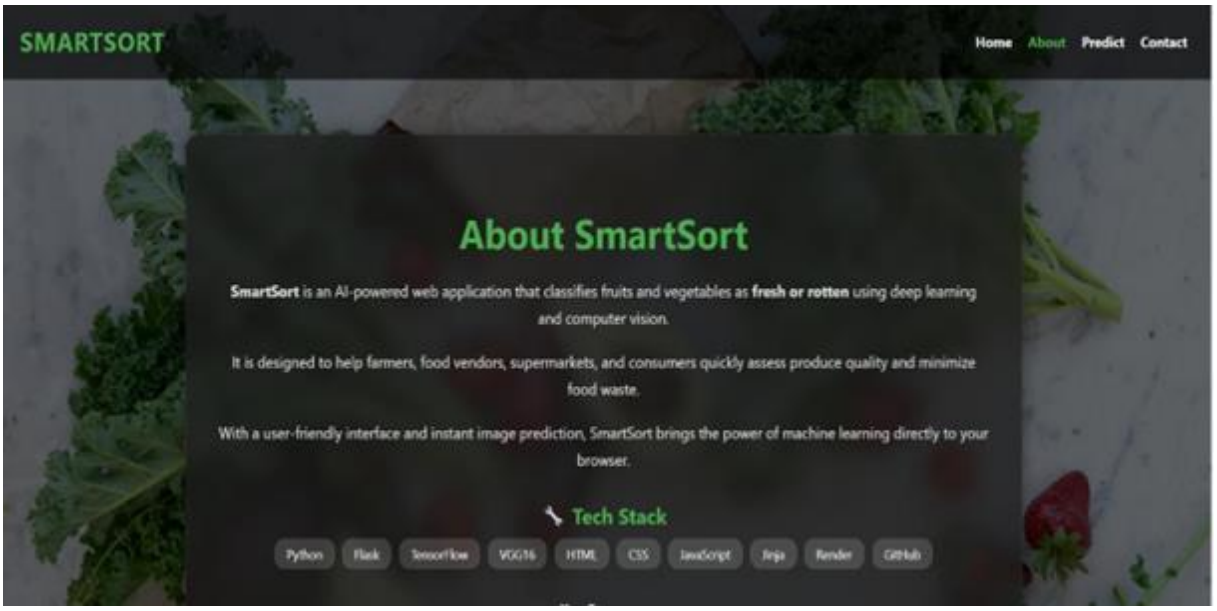
Accuracy testing using test datasets

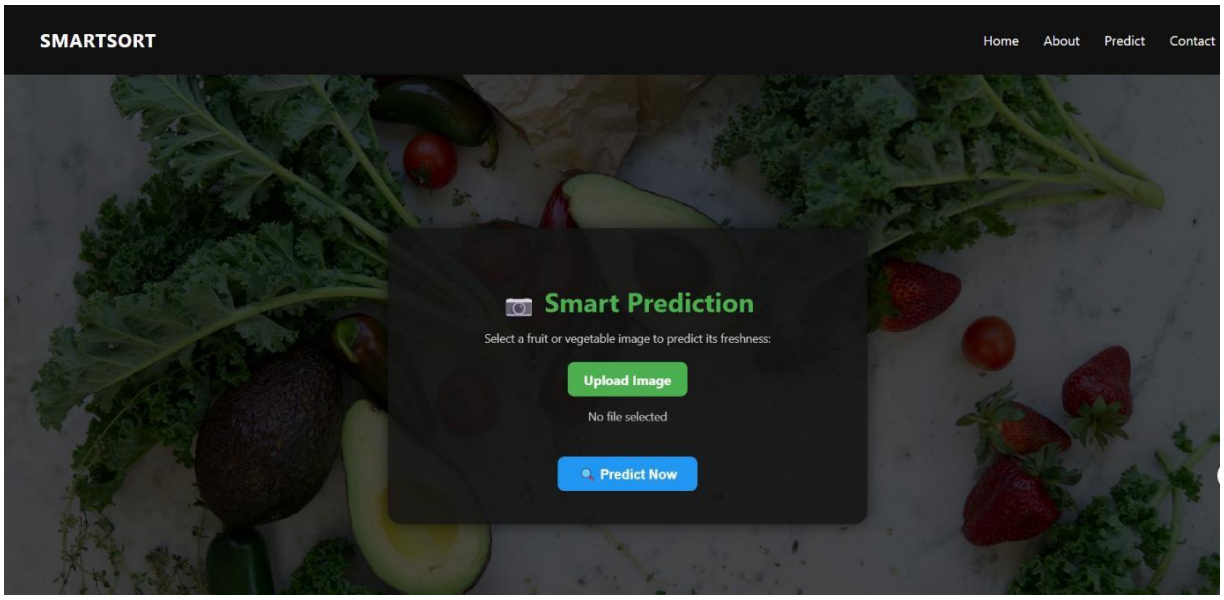
---

11. SCREENSHOTS OR DEMO

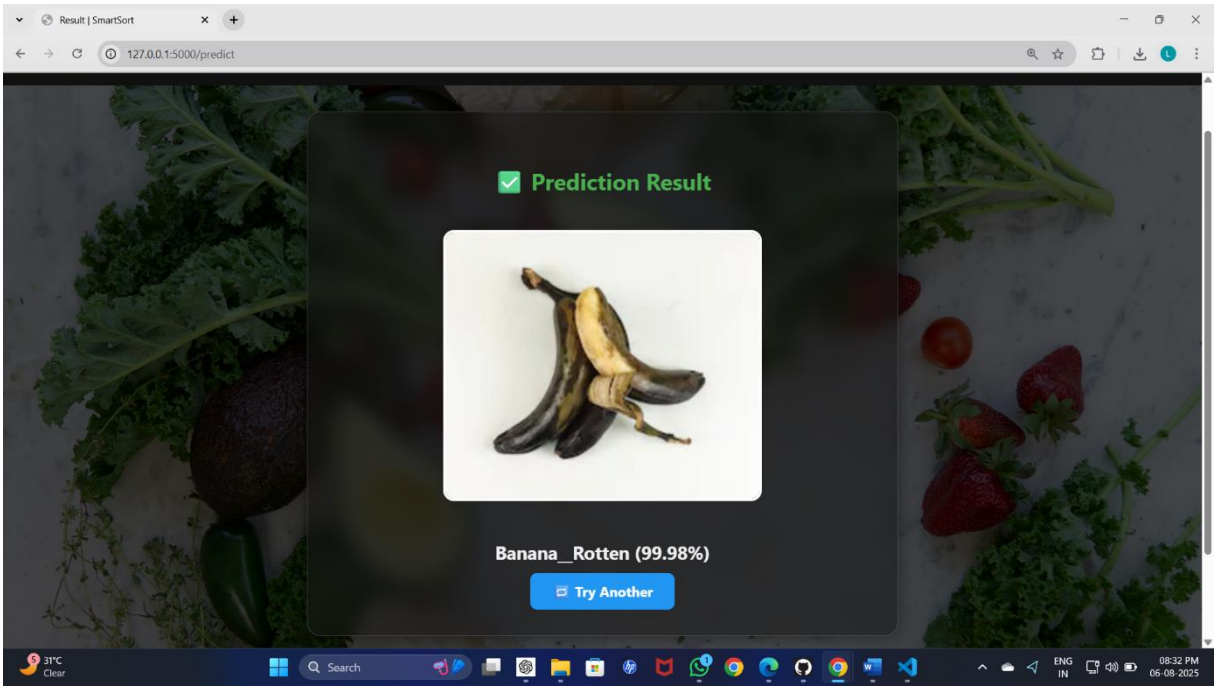


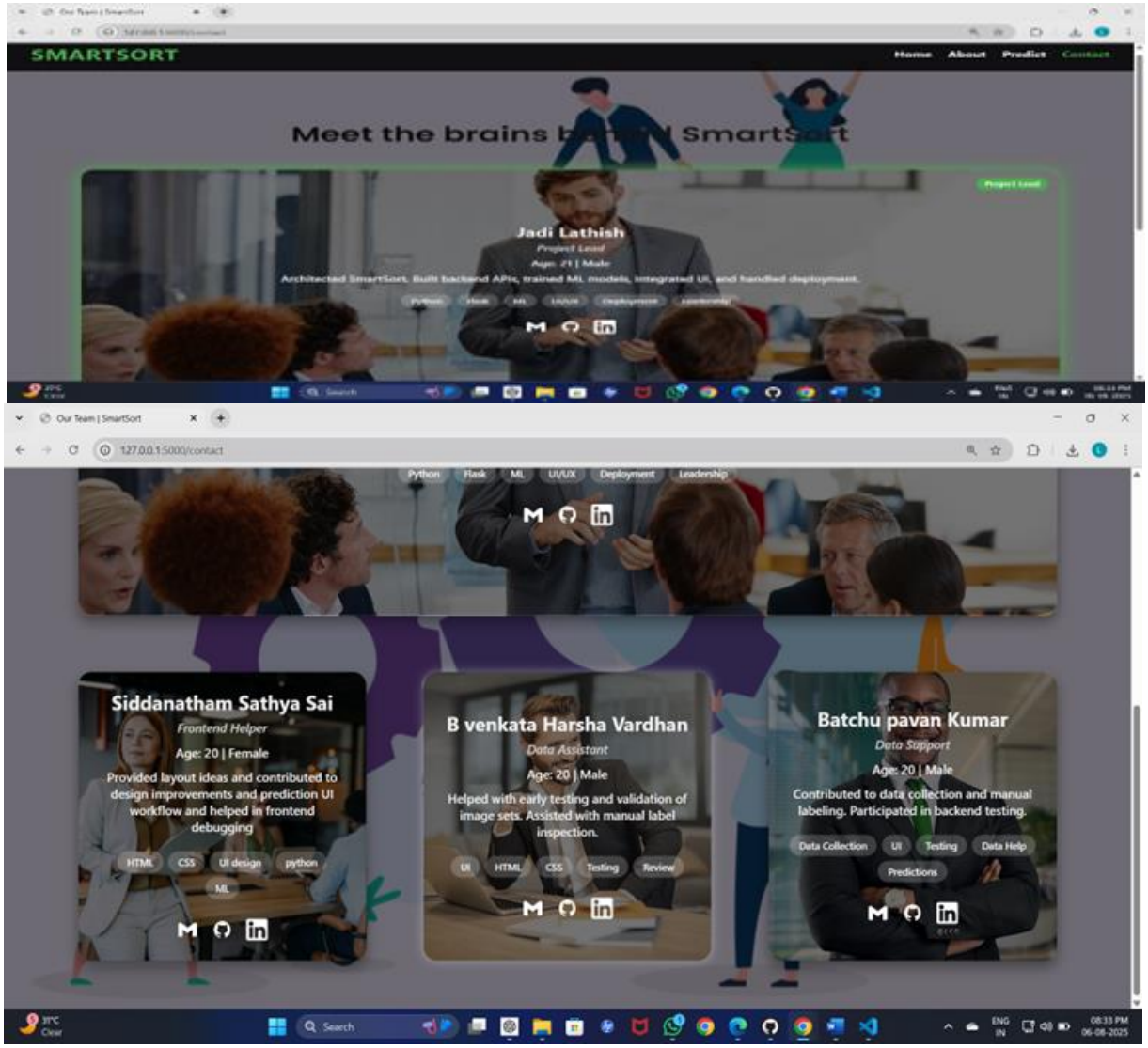



SMARTSORT
Home
About
Predict
Contact
About SmartSort
SmartSort is an AI-powered web application that classifies fruits and vegetables as fresh or rotten using deep learning and computer vision.
It is designed to help farmers, food vendors, supermarkets, and consumers quickly assess produce quality and minimize food waste.
With a user-friendly interface and instant image prediction, SmartSort brings the power of machine learning directly to your browser.
Tech Stack
Python
Flask
TensorFlow
VGG16
HTML
CSS
JavaScript
Jinja
Render
GitHub


SMARTSORT
Home
About
Predict
Contact
Smart Prediction
Select a fruit or vegetable image to predict its freshness:
Upload Image
No file selected
Predict Now


Prediction Result
Apple__Rotten (92.20%)
Try Another


Result | SmartSort
127.0.0.1:5000/predict
Prediction Result
Banana__Rotten (99.98%)
Try Another
31°C
Clear
Search
ENG
IN
08:32 PM
06-08-2025



12. KNOWN ISSUES

- Image quality affects prediction accuracy
- Limited to trained categories (only trained fruits/vegetables)
- No user login system yet

---

13. FUTURE ENHANCEMENTS

- Add user authentication system
- Enable drag-and-drop upload
- Train model on more fruits/vegetables
- Develop mobile app version
