Smart Sorting Project Documentation

# Title: Smart Sorting – Transfer Learning for Identifying Rotten Fruits and Vegetables

## 1. Introduction

The Smart Sorting project aims to automate the classification of fruits and vegetables into fresh or rotten categories using deep learning. This system is useful in agricultural industries, warehouses, and retail stores to maintain food quality and reduce waste. We used transfer learning to improve accuracy with minimal training time.

## 2. Objectives

- To build a model that accurately detects fresh or rotten produce.  
- To deploy the model using a web-based interface for real-time use.  
- To create a reliable, user-friendly system for food quality assessment.

## 3. Technologies Used

- Python: Programming language  
- TensorFlow & Keras: Deep learning model  
- Transfer Learning: Leverage pre-trained models  
- Google Colab: Model training with GPU  
- Flask: Web app deployment  
- HTML/CSS: Frontend interface  
- Google Drive: Dataset storage and access

## 4. Dataset Details

- Total Images: 32,769  
- Number of Classes: 16 (e.g., freshbanana, rottenbanana, etc.)  
- Train/Test Split: Images divided into separate `train` and `test` folders.  
- Image Format: JPG/PNG

## 5. Project Workflow

1. Dataset Preparation  
 - Collected and organized images into labeled folders.  
2. Model Building (in Google Colab)  
 - Used ImageDataGenerator to preprocess images.  
 - Applied CNN (transfer learning) for feature extraction.  
 - Trained the model using training and validation sets.  
3. Model Saving  
 - Saved model as `fruit\_classifier.h5`  
 - Saved class label mapping in `class\_indices.json`  
4. Web App Development (Flask)  
 - Built a Flask app with an HTML form for image upload.  
 - Integrated the trained model to predict uploaded images.  
 - Displayed class name (e.g., "rottenbanana") as output.

## 6. Results

- Model Accuracy: ~95% on validation data  
- Correct Prediction Mapping: Ensured using saved `class\_indices.json`  
- Flask App: Successfully predicts uploaded image class with correct label

## 7. Web Application Features

- Upload an image of a fruit or vegetable  
- Instantly see if it is fresh or rotten  
- Easy-to-use frontend with clean UI

## 8. Future Improvements

- Add more classes for additional fruits and vegetables  
- Deploy on Raspberry Pi for edge-based sorting  
- Integrate hardware sensors and conveyor systems for automation

## 9. Conclusion

The Smart Sorting system combines deep learning and web technologies to provide a simple yet powerful tool for food quality analysis. It automates a critical part of the food industry and opens doors for advanced smart farming applications.