**INTRODUCTION**

* 1. Project Overview

The project **“Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables”** aims to develop an intelligent system that can automatically detect and classify fruits and vegetables as fresh or rotten using machine learning techniques. The system leverages **transfer learning** models trained on image datasets to achieve accurate classification.

The solution is designed to help **reduce manual sorting errors, save time, minimize food wastage, and ensure better quality control** in food processing and retail environments. The project provides an easy-to-use **web interface** where users can upload images and instantly receive classification results.

The system is built using technologies like **Python, TensorFlow, HTML, CSS, Flask, and Kaggle datasets.**

* 1. Purpose

The purpose of this project is to **develop an intelligent sorting system** that can automatically **identify and classify rotten and fresh fruits and vegetables** using image processing and transfer learning techniques.

This system aims to:

* **Improve sorting efficiency**
* **Reduce manual inspection efforts**
* **Minimize food wastage**
* **Ensure better quality control** in food supply chains, especially in supermarkets, warehouses, and local markets.

The solution will provide a **quick, reliable, and user-friendly platform** where users can upload images and receive instant classification results.

2. **IDEATION PHASE**

2.1 Problem Statement

2.2 Empathy Map Canvas

2.3 Brainstorming

**3. REQUIREMENT ANALYSIS**

3.1 Customer Journey map

3.2 Solution Requirement

3.3 Data Flow Diagram

3.4 Technology Stack

**4. PROJECT DESIGN**

4.1 Problem Solution Fit

4.2 Proposed Solution

4.3 Solution Architecture

**5. PROJECT PLANNING & SCHEDULING**

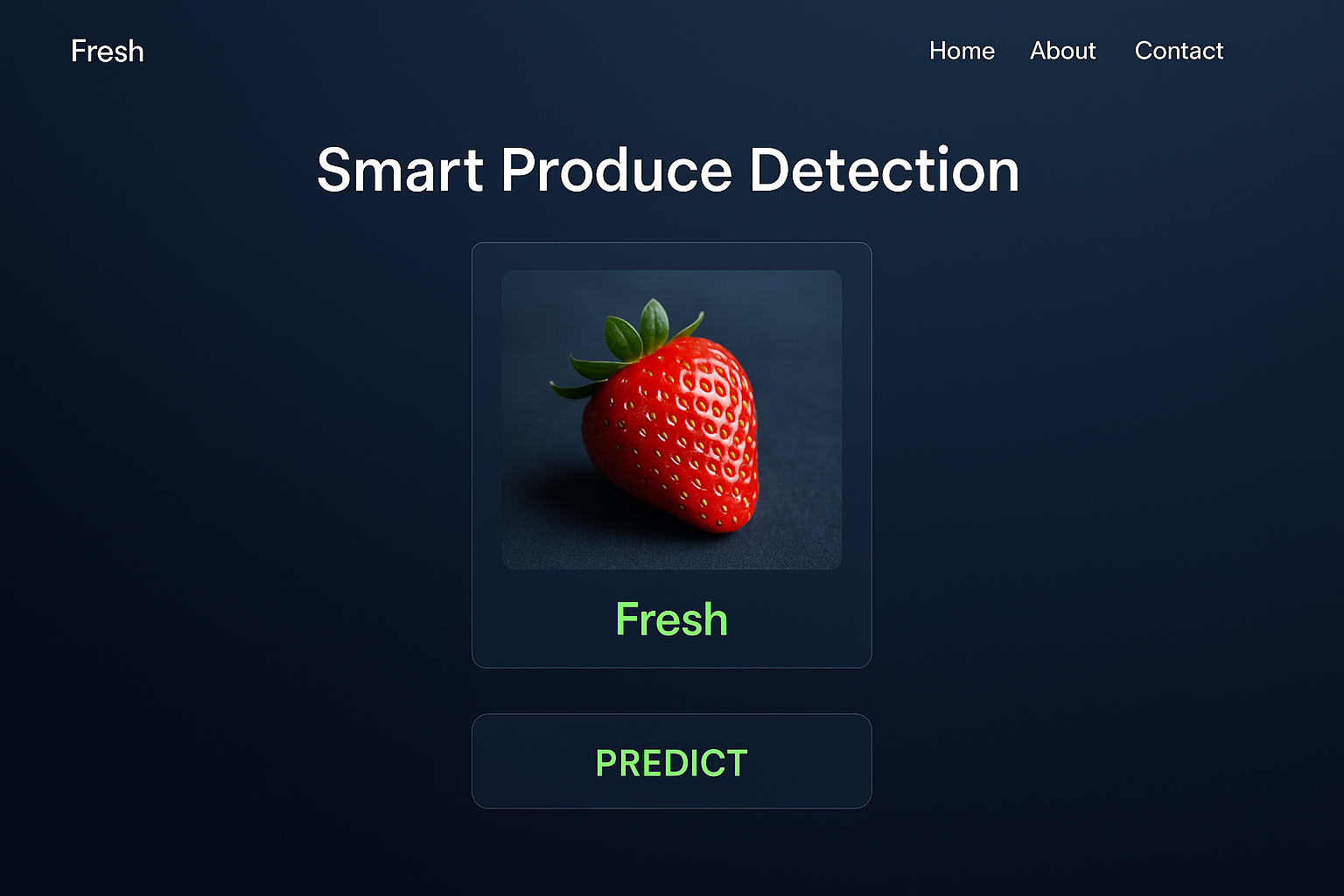
5.1 Project Planning

**6. FUNCTIONAL AND PERFORMANCE TESTING**

6.1 Performance Testing

**7. RESULTS**

7.1 Output Screenshots



**8. ADVANTAGES & DISADVANTAGES**

**Advantages:**

1. Time-Saving: Quickly identifies rotten fruits and vegetables, reducing manual inspection time.
2. Improved Accuracy: Uses machine learning to provide consistent and reliable classification.
3. Minimizes Food Wastage: Helps in early detection of rotten produce to prevent spoilage of fresh items.
4. User-Friendly Interface: Easy for users to upload images and receive results instantly.
5. Scalable: Can be extended to other types of food products in the future.

**Disadvantages:**

1. **Limited Offline Functionality:** System requires internet access to process images through the model.
2. **Dataset Dependency:** Accuracy highly depends on the quality and variety of the training dataset.
3. **Initial Setup Time:** Preprocessing, model training, and deployment require significant initial effort.
4. **Image Quality Sensitivity:** Low-quality or unclear images may lead to incorrect classification.

**9. CONCLUSION**

The project “Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables” successfully demonstrates the use of machine learning and image processing to automate the classification of fruits and vegetables as fresh or rotten.

By implementing transfer learning, the system provides quick, accurate, and reliable results, reducing manual effort and minimizing food wastage. The user-friendly web interface makes the solution accessible to anyone with basic computer knowledge.

This project highlights the potential of artificial intelligence in improving food quality control and efficiency in real-world applications.

**10. FUTURE SCOPE**

Extend to detect fruit ripeness and specific diseases.

Develop a mobile app for easy access.

Add offline processing features.

Integrate with real-time sorting systems.

Expand dataset for higher accuracy.

**11. APPENDIX**

Source Code(if any)

The project is developed using **Python, HTML, CSS, and Flask.**

Code includes model training, image upload, and result display.

Files are available locally and can be shared on request.

Dataset Link

GitHub & Project Demo Link

GitHub link:- <https://github.com/sameershaik16/smart-sorting>

Demo link :- https://drive.google.com/file/d/1E-2SKLkf07DeBCLIdmPswQlzM2LtV32B/view?usp=drivesdk