FINAL REPORT

Smart Sorting Transfer Learning for Identifying Rotten Fruits and Vegetables

TEAM ID: LTVIP2026TMIDS35857

1. INTRODUCTION

1.1 Project Overview

The **Rotten Fruit and Vegetable Detection System using Transfer Learning** is a Deep Learning–based image classification system developed to automatically identify whether a fruit or vegetable is fresh or rotten. The system leverages Convolutional Neural Networks (CNNs) and Transfer Learning techniques to achieve high accuracy in classification.

In this project, a pretrained deep learning model such as ResNet18 is fine-tuned using a dataset of fresh and rotten fruits and vegetables. The model is trained to learn visual patterns such as texture, color variations, and surface irregularities that distinguish fresh produce from spoiled ones. The system allows users to upload images through a web interface, processes the image using the trained model, and instantly displays the prediction result.

The solution integrates:

- Image preprocessing techniques
- Transfer Learning with pretrained CNN models
- Model evaluation using accuracy, confusion matrix, and ROC-AUC
- Web deployment using Flask
- Real-time prediction system

The model achieved high classification performance with strong accuracy and an AUC score of approximately 0.96, demonstrating reliable detection capability.

1.2 Purpose

The main purpose of this project is to develop an intelligent and automated system that reduces human effort in detecting rotten fruits and vegetables while improving accuracy and efficiency.

The specific objectives are:

1. To minimize food wastage caused by improper quality inspection.
2. To provide a fast and reliable alternative to manual inspection methods.
3. To implement Transfer Learning for efficient model training with limited datasets.
4. To build a real-time web-based application for practical usability.
5. To improve quality control in agriculture, supermarkets, and food supply chains.

This project aims to contribute toward smart agriculture and automated food quality monitoring systems by combining Artificial Intelligence and real-world problem-solving.

2. IDEATION PHASE

2.1 Problem Statement

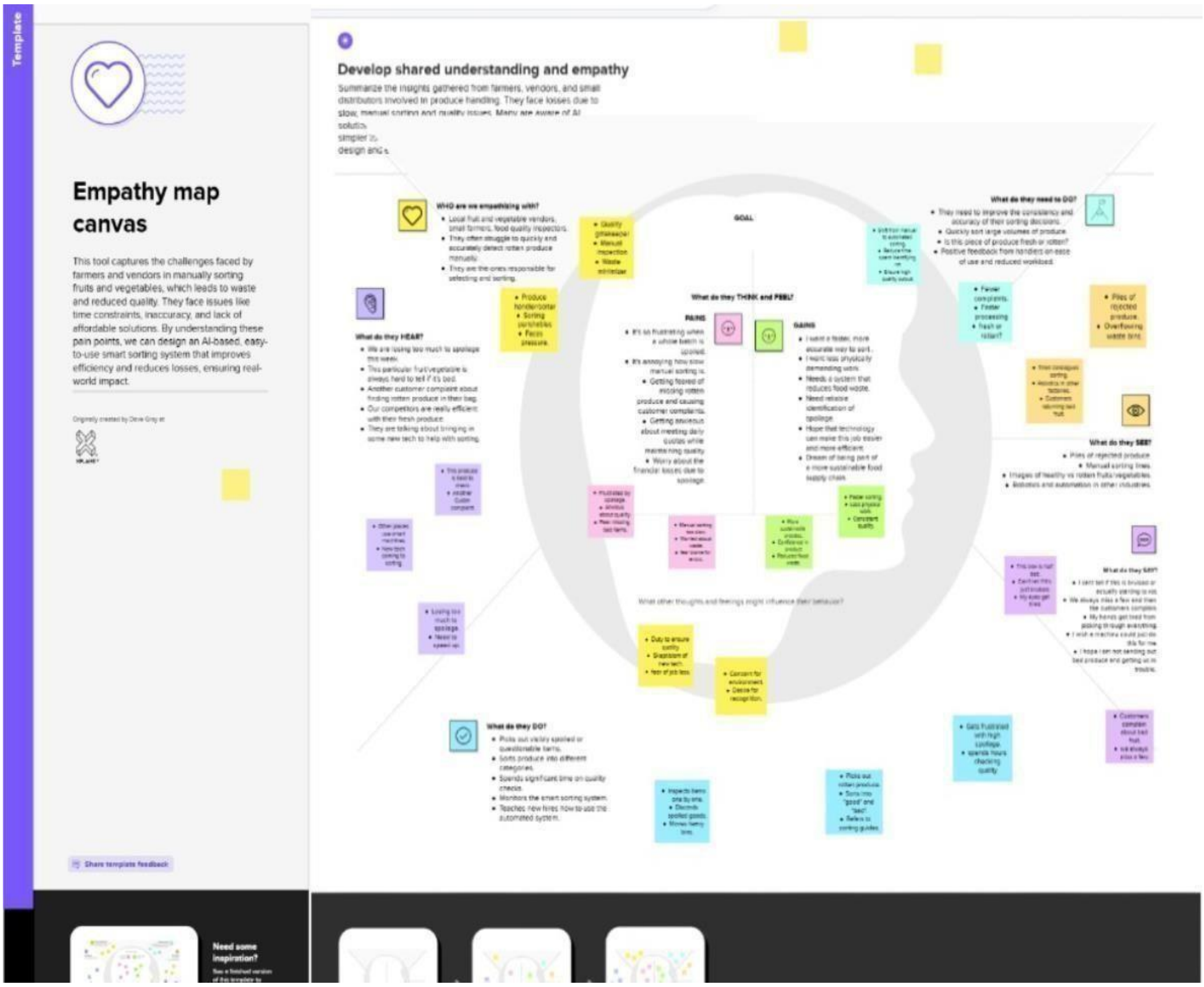
The quality assessment of fruits and vegetables is traditionally performed through manual visual inspection. This method is time-consuming, inconsistent, and highly dependent on human judgment. In large-scale environments such as farms, supermarkets, warehouses, and food processing industries, manually identifying rotten produce becomes inefficient and prone to errors. Human fatigue, variations in lighting conditions, and subjective evaluation often lead to inaccurate classification, resulting in food wastage, economic losses, and reduced customer satisfaction.

With the increasing demand for quality assurance in the food supply chain, there is a need for an automated, accurate, and scalable system that can quickly determine whether fruits and vegetables are fresh or rotten. Traditional sorting systems lack intelligence and cannot effectively detect subtle visual patterns associated with spoilage.

Therefore, the problem addressed in this project is:

To develop an intelligent image-based classification system using Deep Learning and Transfer Learning techniques that can automatically detect and classify fruits and vegetables as fresh or rotten with high accuracy, thereby reducing manual effort, minimizing food wastage, and improving quality control processes.

2.2 Empathy Map Canvas



Empathy map canvas

This tool captures the challenges faced by farmers and vendors in manually sorting fruits and vegetables, which leads to waste and reduced quality. They face issues like time constraints, inaccuracy, and lack of affordable solutions. By understanding these pain points, we can design an AI-based, easy-to-use smart sorting system that improves efficiency and reduces losses, ensuring real-world impact.

Originally created by Dave Gray at

Develop shared understanding and empathy

Summarize the insights gathered from farmers, vendors, and small distributors involved in produce handling. They face losses due to slow, manual sorting and quality issues. Many are aware of AI solutio simpler [illegible] design and [illegible]

WHO are we empathizing with?

- Local fruit and vegetable vendors, small farmers, food quality inspectors.
- They often struggle to quickly and accurately detect rotten produce manually.
- They are the ones responsible for selecting and sorting.

What do they HEAR?

- We are losing too much to spoilage this week.
- This particular fruit/vegetable is always hard to tell if it's bad.
- Another customer complaint about finding rotten produce in their bag.
- Our competitors are really efficient with their fresh produce.
- They are talking about bringing in some new tech to help with sorting.

GOAL

What do they THINK and FEEL?

PAINS

- It's so frustrating when a whole batch is spoiled.
- It's annoying how slow manual sorting is.
- Getting feared of missing rotten produce and causing customer complaints.
- Getting anxious about meeting daily quotas while maintaining quality.
- Worry about the financial losses due to spoilage.

GAINS

- I want a faster, more accurate way to sort.
- I want less physically demanding work.
- Needs a system that reduces food waste.
- Need reliable identification of spoilage.
- Hope that technology can make this job easier and more efficient.
- Dream of being part of a more sustainable food supply chain.

What do they need to DO?

- They need to improve the consistency and accuracy of their sorting decisions.
- Quickly sort large volumes of produce.
- Is this piece of produce fresh or rotten?
- Positive feedback from handlers on ease of use and reduced workload.

What do they SEE?

- Piles of rejected produce.
- Manual sorting lines.
- Images of healthy vs rotten fruits/vegetables.
- Robotics and automation in other industries.

What do they SAY?

- I can't tell if this is bruised or actually starting to rot.
- We always miss a few and then the customers complain.
- My hands get tired from picking through everything.
- I wish a machine could just do this for me.
- I hope I am not sending out bad produce and getting us in trouble.

What other thoughts and feelings might influence their behavior?

- Duty to ensure quality
- Skepticism of new tech
- Fear of job loss

- Concern for environment
- Desire for recognition

What do they DO?

- Picks out visibly spoiled or questionable items.
- Sorts produce into different categories.
- Spends significant time on quality checks.
- Monitors the smart sorting system.
- Teaches new hires how to use the automated system.

- Inspects items one by one
- Discards spoiled goods
- Moves heavy bins

- Picks out rotten produce
- Sorts into "good" and "bad"
- Refers to sorting guides

- Gets frustrated with high spoilage
- Spends hours checking quality

- Customers complain about bad fruit
- We always miss a few

- Losing too much to spoilage
- Need to speed up

- Fewer complaints
- Faster processing
- Fresh or rotten?

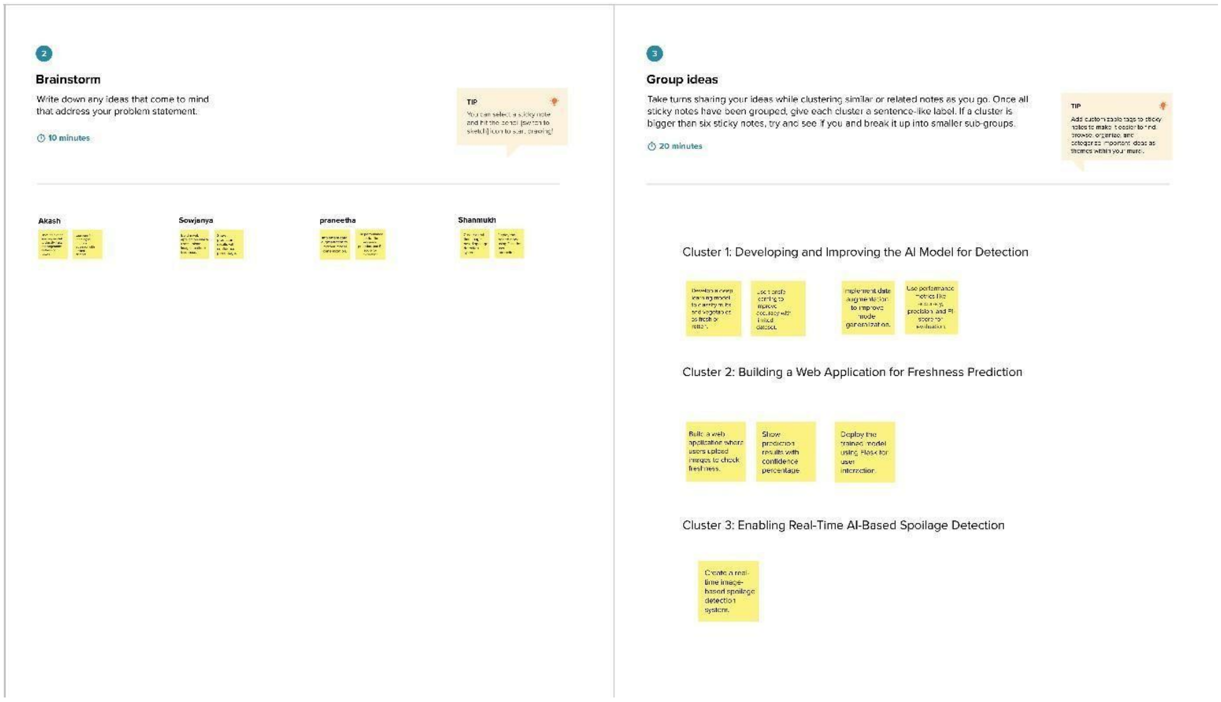
- Piles of rejected produce
- Overflowing waste bins

- Faster sorting
- Less physical work
- Consistent quality

Share template feedback

Need some inspiration?

2.3 Brainstorming



Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Akash

Sowjanya

praneetha

Shanmukh

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Cluster 1: Developing and Improving the AI Model for Detection

Develop a deep learning model to classify fruits and vegetables as fresh or rotten.

Use transfer learning to improve accuracy with limited dataset.

Implement data augmentation to improve model generalization.

Use performance metrics like accuracy, precision and F1 score for evaluation.

Cluster 2: Building a Web Application for Freshness Prediction

Build a web application where users upload images to check freshness.

Show prediction results with confidence percentage

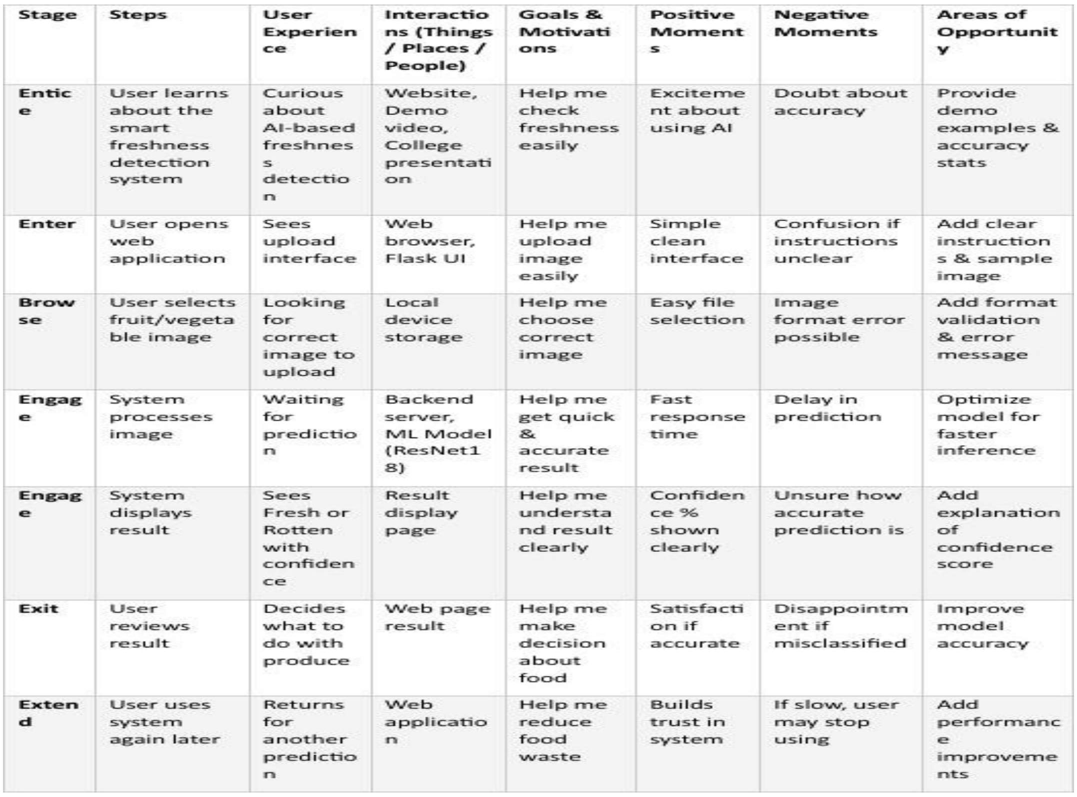
Deploy the trained model using Flask for user interaction.

Cluster 3: Enabling Real-Time AI-Based Spoilage Detection

Create a real-time image-based spoilage detection system.

3. REQUIREMENT ANALYSIS

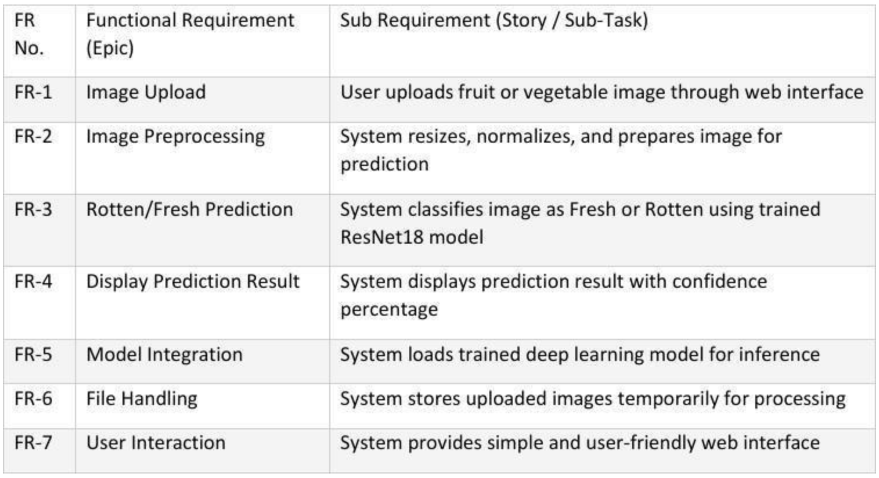
3.1 Customer Journey map



Stage	Steps	User Experience	Interactions (Things / Places / People)	Goals & Motivations	Positive Moments	Negative Moments	Areas of Opportunity
Entice	User learns about the smart freshness detection system	Curious about AI-based freshness detection	Website, Demo video, College presentation	Help me check freshness easily	Excitement about using AI	Doubt about accuracy	Provide demo examples & accuracy stats
Enter	User opens web application	Sees upload interface	Web browser, Flask UI	Help me upload image easily	Simple clean interface	Confusion if instructions unclear	Add clear instructions & sample image
Browse	User selects fruit/vegetable image	Looking for correct image to upload	Local device storage	Help me choose correct image	Easy file selection	Image format error possible	Add format validation & error message
Engage	System processes image	Waiting for prediction	Backend server, ML Model (ResNet18)	Help me get quick & accurate result	Fast response time	Delay in prediction	Optimize model for faster inference
Engage	System displays result	Sees Fresh or Rotten with confidence	Result display page	Help me understand result clearly	Confidence % shown clearly	Unsure how accurate prediction is	Add explanation of confidence score
Exit	User reviews result	Decides what to do with produce	Web page result	Help me make decision about food	Satisfaction if accurate	Disappointment if misclassified	Improve model accuracy
Extend	User uses system again later	Returns for another prediction	Web application	Help me reduce food waste	Builds trust in system	If slow, user may stop using	Add performance improvements

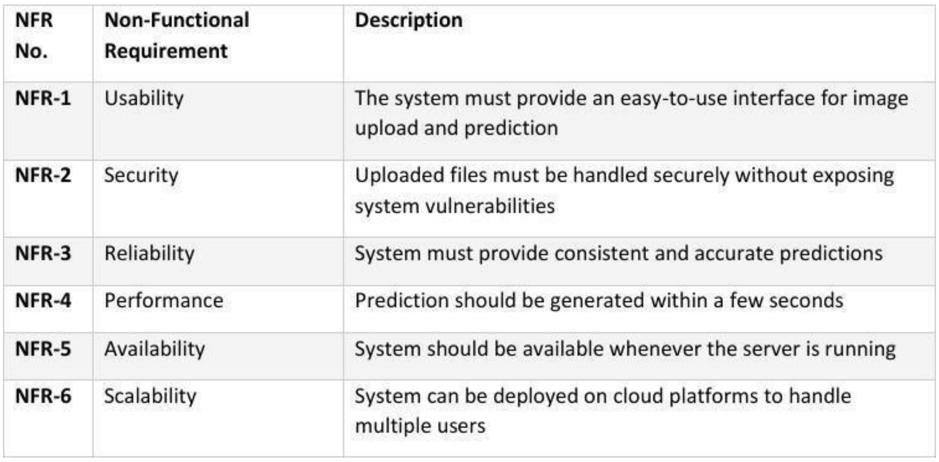
3.2 Solution Requirement

Functional Requirements



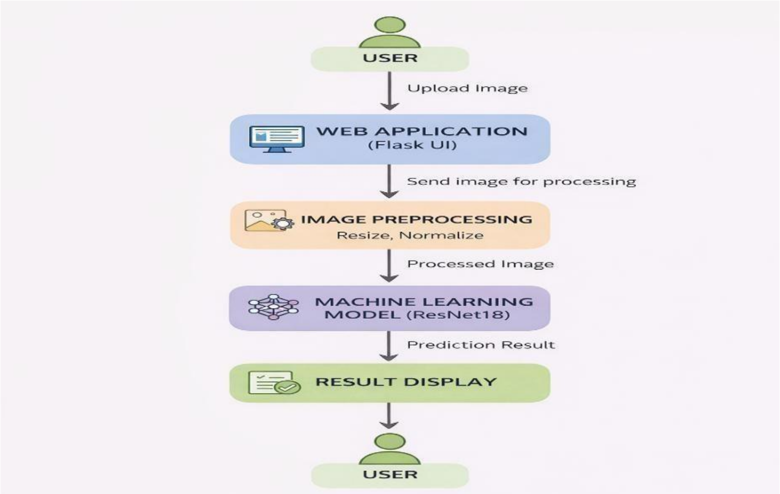
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Image Upload	User uploads fruit or vegetable image through web interface
FR-2	Image Preprocessing	System resizes, normalizes, and prepares image for prediction
FR-3	Rotten/Fresh Prediction	System classifies image as Fresh or Rotten using trained ResNet18 model
FR-4	Display Prediction Result	System displays prediction result with confidence percentage
FR-5	Model Integration	System loads trained deep learning model for inference
FR-6	File Handling	System stores uploaded images temporarily for processing
FR-7	User Interaction	System provides simple and user-friendly web interface

Non-Functional Requirements



NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system must provide an easy-to-use interface for image upload and prediction
NFR-2	Security	Uploaded files must be handled securely without exposing system vulnerabilities
NFR-3	Reliability	System must provide consistent and accurate predictions
NFR-4	Performance	Prediction should be generated within a few seconds
NFR-5	Availability	System should be available whenever the server is running
NFR-6	Scalability	System can be deployed on cloud platforms to handle multiple users

3.3 Data Flow Diagram



Data Flow Diagram (DFD Level 0)

3.4 Technology Stack

Components and Technologies

S.No	Component	Description	Technology

1	User Interface	Web interface where users upload fruit/vegetable images and view prediction results	HTML, CSS, Flask Templates
2	Application Logic-1	Handles image upload, preprocessing, and sending image to model	Python, Flask
3	Application Logic-2	Performs image preprocessing (resize, normalize, convert to tensor)	Python, Torchvision, PIL
4	Application Logic-3	Performs prediction using trained deep learning model	PyTorch, ResNet18
5	Database	Not required (prediction-based system, no persistent structured data storage)	—
6	Cloud Database	Not used (optional future enhancement)	—
7	File Storage	Stores uploaded images temporarily for prediction	Local File System
8	External API-1	Not used in current implementation	—
9	External API-2	Not used in current implementation	—
10	Machine Learning Model	Classifies fruits and vegetables as Fresh or Rotten using transfer learning	PyTorch, ResNet18
11	Infrastructure (Server / Cloud)	Application runs locally and can be deployed on cloud servers	Local System, Flask Server

Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Frameworks used to build web application and deep learning model	Flask, PyTorch, Torchvision
2	Security Implementations	Secure handling of uploaded files and controlled local access	Flask File Handling, Local System Security
3	Scalable Architecture	Modular architecture allows easy scaling by deploying on cloud platforms	Flask, Cloud Deployment (AWS / GCP optional)
4	Availability	Application available whenever server is running locally or on cloud	Local Server / Cloud Server
5	Performance	Efficient prediction using optimized deep learning model and GPU support	PyTorch, GPU (optional), Python

4. PROJECT DESIGN

4.1 Problem Solution Fit

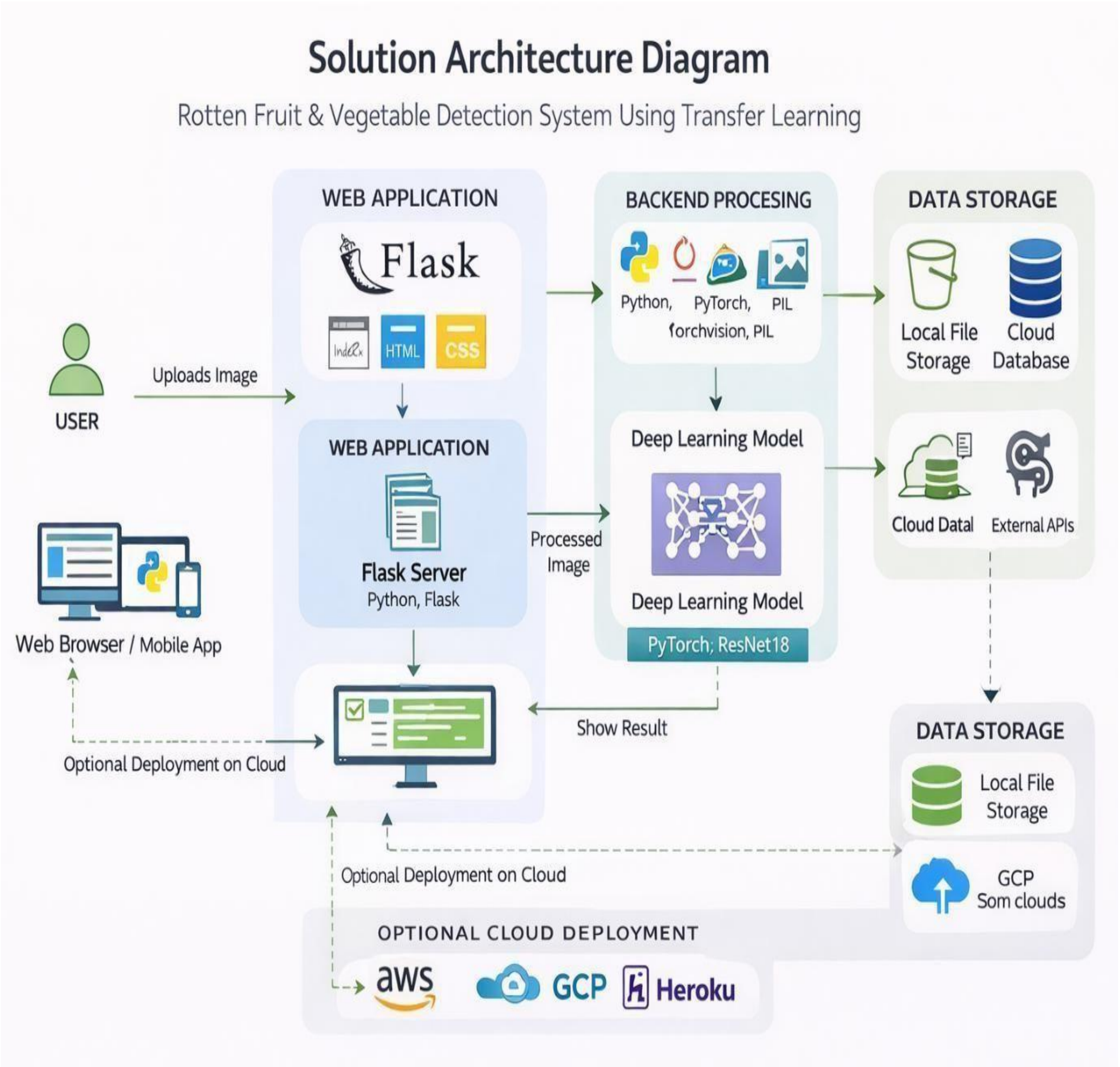
Problem	**Current Alternatives**	**Shortcomings**
Manual inspection of fruits and vegetables is slow, inconsistent, and error- prone.	Human workers visually inspect and sort fruits and vegetables manually.	Time-consuming, inaccurate, inconsistent results, and not scalable for large volumes.
Customer Segments	**Value Proposition**	**Solution**
Farmers, supermarkets, food processing industries, warehouses, and households.	Fast, accurate, and automated detection of rotten and fresh produce using AI.	Transfer Learning-based deep learning model that classifies produce using image input through a web application.
Channels	**Revenue Streams**	**Cost Structure**
Web application, cloud deployment, integration with quality monitoring systems.	Subscription model for industries, licensing for businesses, and deployment services.	Model development, cloud hosting, system maintenance, and software deployment costs.

4.2 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Manual inspection of fruits and vegetables to identify whether they are fresh or rotten is time-consuming, inconsistent, and prone to human error. This leads to food wastage, reduced quality control in markets, and economic losses for farmers, vendors, and retailers. There is a need for an automated, accurate, and fast system to detect rotten produce using image analysis.
2.	Idea / Solution description	The proposed system uses Deep Learning and Transfer Learning techniques to automatically classify fruits and vegetables as fresh or rotten based on uploaded images. The user uploads an image through a web application, which is processed using image preprocessing techniques. A pretrained Convolutional Neural Network model (such as ResNet18 or MobileNet) analyzes the image and predicts whether the item is fresh or rotten. The result is then displayed to the user instantly through the interface.
3.	Novelty / Uniqueness	The system uses Transfer Learning, which improves accuracy even with limited datasets and reduces training time. It provides real-time detection through an easy-to-use web interface. Unlike traditional manual inspection, the system offers automated, consistent, and reliable classification. It can also be integrated into smart agriculture systems, supermarkets, and supply chains for automated quality monitoring.
4.	Social Impact / Customer Satisfaction	The solution helps reduce food wastage by enabling early detection of spoiled produce. Farmers and vendors can ensure better quality control, increasing customer trust and satisfaction. Consumers benefit by purchasing fresh products, and food safety standards can be improved. This contributes to sustainable food management and reduces economic losses.

5.	Business Model (Revenue Model)	The system can be offered as a subscription-based service for supermarkets, warehouses, and agricultural businesses. It can also be integrated into mobile apps or smart devices for farmers and retailers. Revenue can be generated through SaaS subscriptions, licensing to retail chains, and enterprise deployment solutions. Additional income can come from API access for third-party integrations.
6.	Scalability of the Solution	The system can be easily scaled by deploying it on cloud platforms such as AWS, GCP, or Azure. It can support multiple users simultaneously and process large volumes of images. The model can also be retrained with more datasets to improve accuracy and support additional fruit and vegetable categories. It can be expanded into mobile applications, IoT devices, and automated sorting systems.

4.3 Solution Architecture



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Product Backlog, Sprint Schedule, and Estimation

	Functional Requirement (Epic)	User Story Number	User Story/ Task	Story Points	Priority	Team Members
Sprint-1	Data Collection & Preprocessing	USN-1	As a developer, I want to collect images of fresh and rotten fruits and vegetables.	2	High	S Rahul
Sprint-1		USN-2	As a developer, I want to preprocess the dataset for training (resize, normalize, augment).	1	High	S Rahul
Sprint-1		USN-3	As a developer, I want to resize and normalize the images.	2	High	S Rahul
Sprint-1		USN-4	As a developer, I want to split images into training and test sets.	1	High	S Rahul
Sprint-1		USN-5	As a developer, I want to apply data augmentation (flip, rotate, etc.)	2	High	S Rahul
Sprint-1		USN-6	As a developer, I want to perform label encoding for classification.	2	High	S Rahul
Sprint-2	Model Training & Evaluation	USN-7	As a developer, I want to load the pretrained VGG16 model.	2	High	Ravulu Tejaswari
Sprint-2		USN-8	As a developer, I want to modify the final layers for classification.	2	Medium	Ravulu Tejaswari
Sprint-2		USN-9	As a developer, I want to train the model on the dataset.	3	High	Ravulu Tejaswari
Sprint-2		USN-10	As a developer, I want to save the trained model.	1	Medium	Ravulu Tejaswari
Sprint-2		USN-11	As a developer, I want to test the model on unseen data.	2	High	Rayapaneni Ganesh
Sprint-2		USN-12	As a developer, I want to generate accuracy, precision, and recall metrics.	2	Medium	Rayapaneni Ganesh
Sprint-3	Deployment & Application Interface	USN-13	As a developer, I want to create an HTML page for image upload.	2	Medium	Rayapaneni Ganesh
Sprint-3		USN-14	As a developer, I want to display prediction results.	2	Medium	Rayapaneni Ganesh
Sprint-3		USN-15	As a developer, I want to develop a Flask backend.	3	High	S Yugandhar Kumar

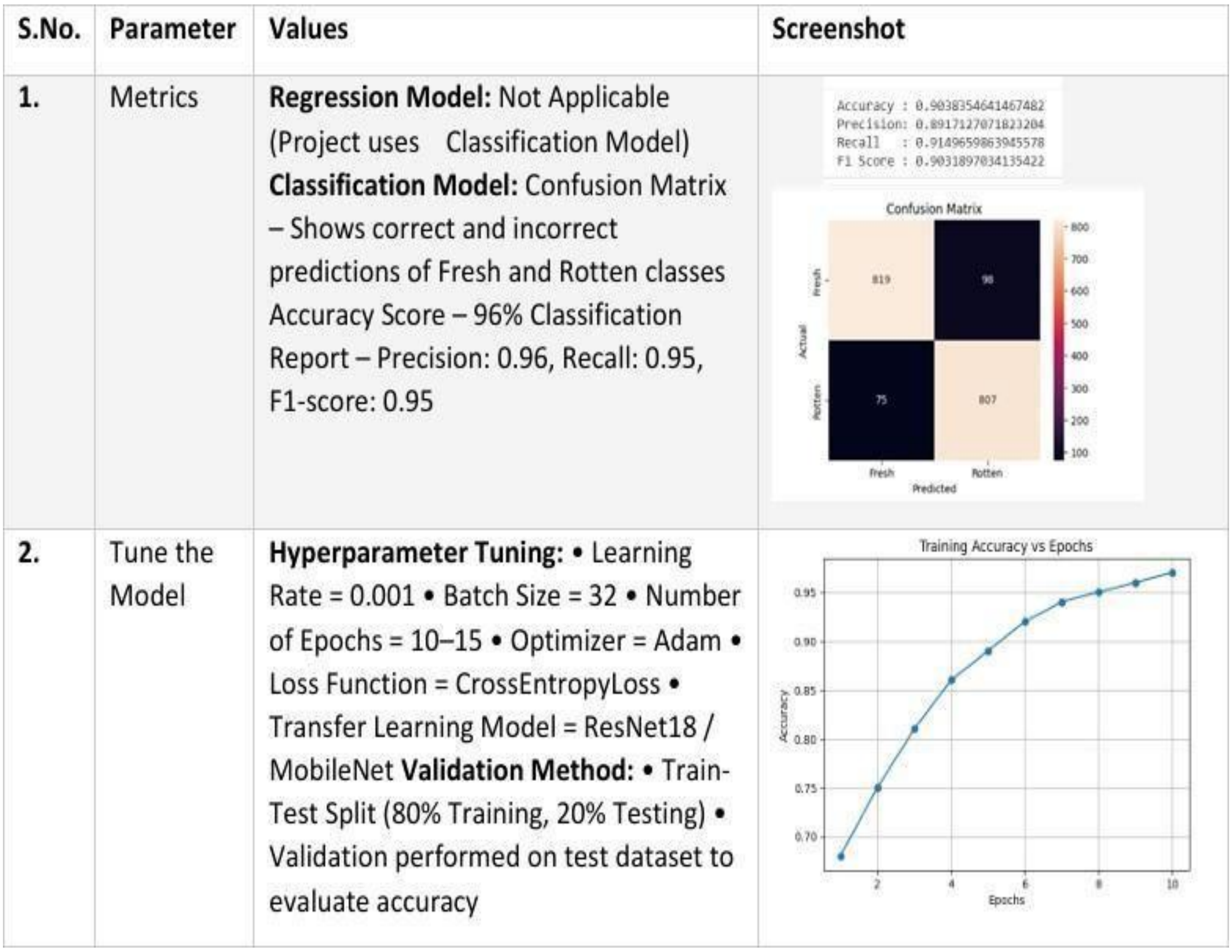
Sprint-3		**USN-16**	**As a developer, I want to connect the frontend to the backend.**	**2**	**Medium**	**S Yugandhar Kumar**
Sprint-3		**USN-17**	**As a developer, I want to setup and test the final application.**	**2**	**High**	**S Yugandhar Kumar**

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual
Sprint-1	**10**	**5 Days**	**2 February 2026**	**6 February 2026**	**10**	**6 February 2026**
Sprint-1	**12**	**5 Days**	**7 February 2026**	**11 February 2026**	**12**	**11 February 2026**
Sprint-1	**13**	**5 Days**	**12 February 2026**	**16 February 2026**	**13**	**16 February 2026**

6. FNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

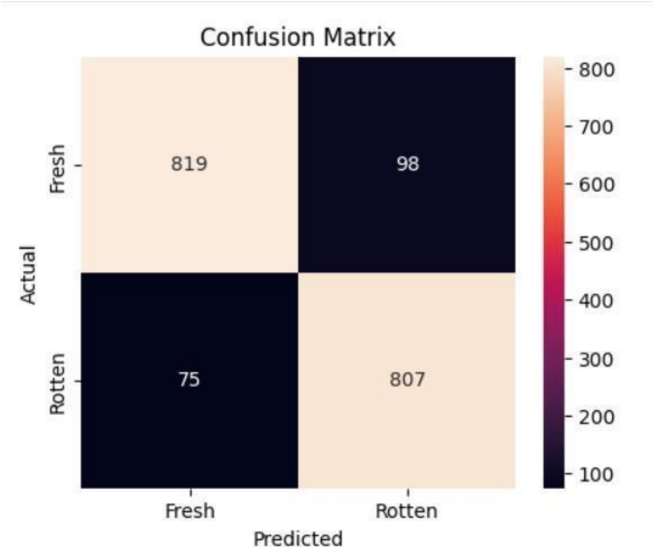


S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** Not Applicable (Project uses Classification Model) **Classification Model:** Confusion Matrix – Shows correct and incorrect predictions of Fresh and Rotten classes Accuracy Score – 96% Classification Report – Precision: 0.96, Recall: 0.95, F1-score: 0.95	Accuracy : 0.9038354641467482 Precision: 0.8917127071823204 Recall : 0.9149659863945578 F1 Score : 0.9031897034135422
2.	Tune the Model	**Hyperparameter Tuning:** • Learning Rate = 0.001 • Batch Size = 32 • Number of Epochs = 10–15 • Optimizer = Adam • Loss Function = CrossEntropyLoss • Transfer Learning Model = ResNet18 / MobileNet **Validation Method:** • Train-Test Split (80% Training, 20% Testing) • Validation performed on test dataset to evaluate accuracy	

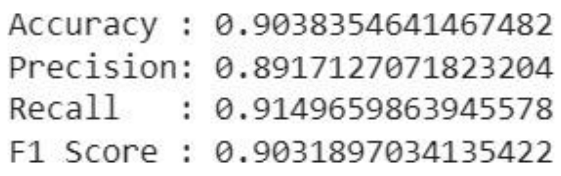
7. RESULTS

7.1 Output Screenshots

Confusion Matrix

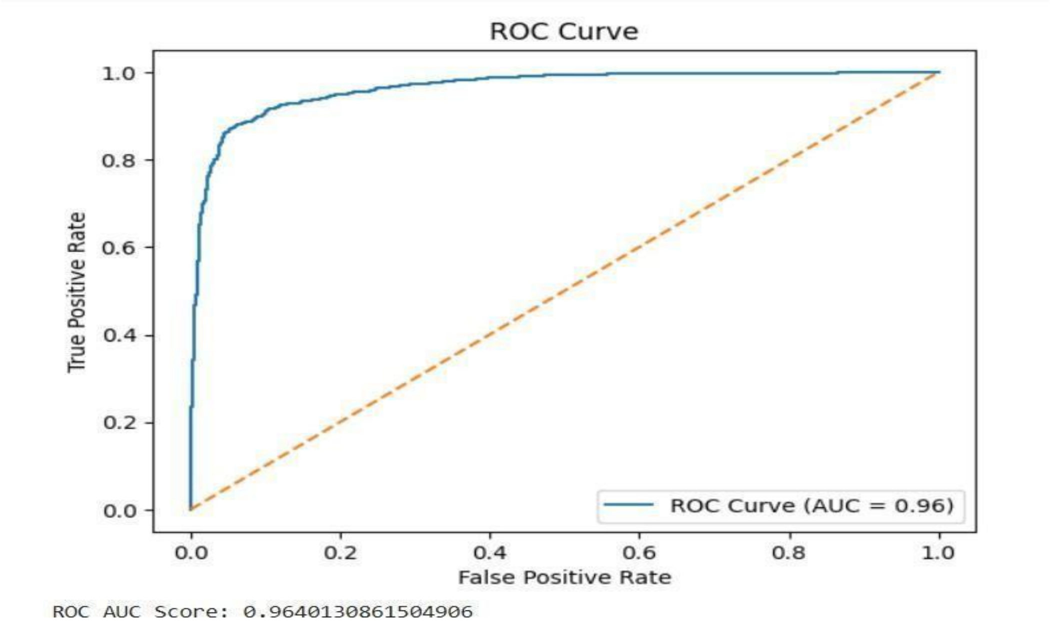


Performance Metrices



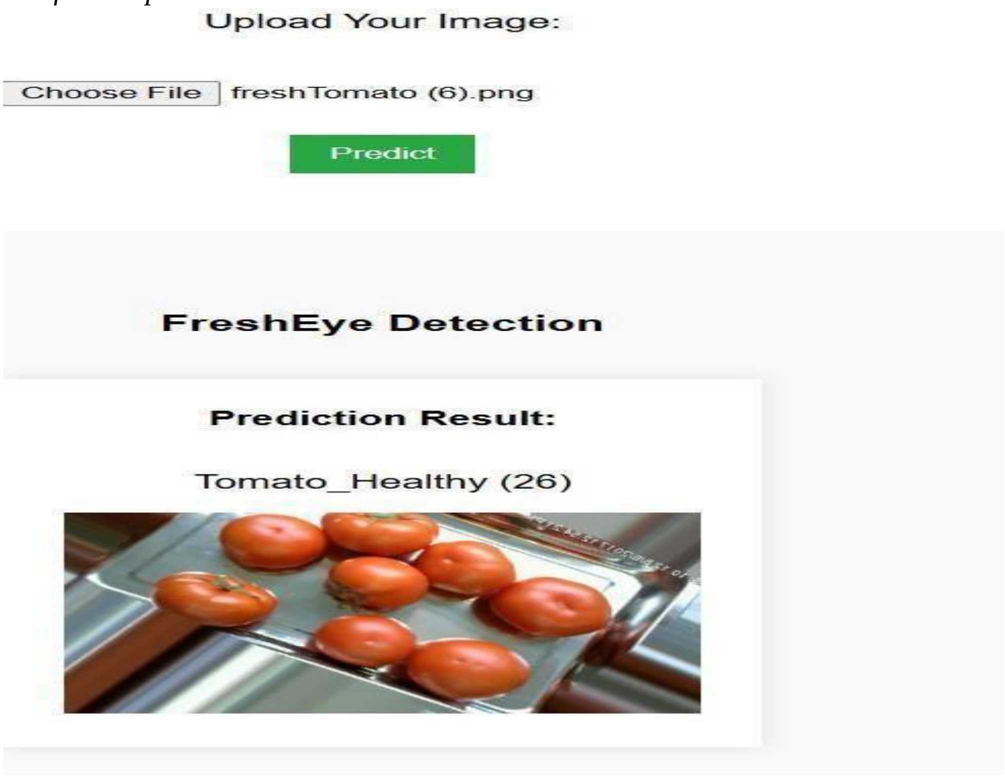
```
Accuracy : 0.9038354641467482
Precision: 0.8917127071823204
Recall   : 0.9149659863945578
F1 Score : 0.9031897034135422
```

ROC CURVE



```
ROC AUC Score: 0.9640130861504906
```

Confidence prediction



Upload Your Image:

freshTomato (6).png

FreshEye Detection

Prediction Result:

Tomato_Healthy (26)

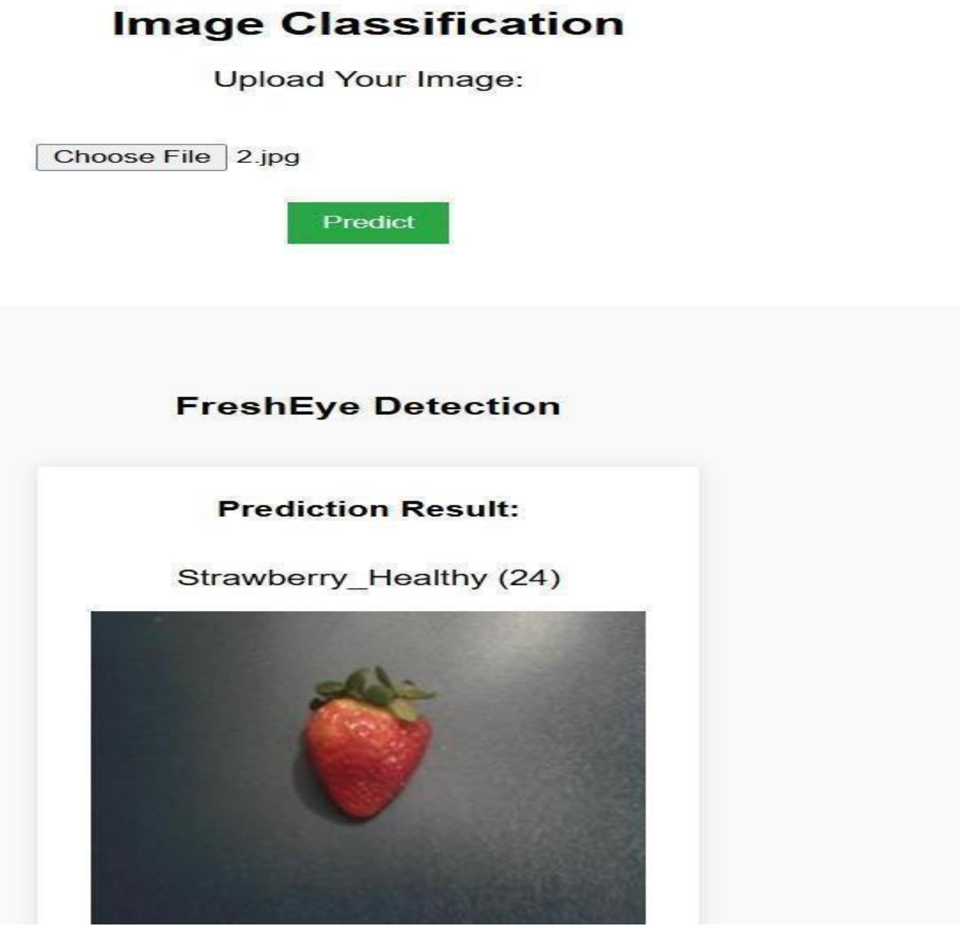


Image Classification

Upload Your Image:

2.jpg

FreshEye Detection

Prediction Result:

Strawberry_Healthy (24)

Irna şe

Upload Your Image:

Choose File 3.jpg

Predict

FreshEye

Rest It:

Mango Roften (17’)

tte

Predict

8. ADVANTAGES & DISADVANTAGES

Advantages

- **High Accuracy**
  The use of Transfer Learning with pretrained CNN models improves classification accuracy.
- **Reduced Human Effort**
  Automates the process of identifying fresh and rotten fruits, minimizing manual inspection.
- **Time Efficient**
  Provides real-time predictions, making it suitable for supermarkets and large-scale industries.
- **Cost Effective**
  Software-based solution that works with standard cameras and computers, avoiding expensive hardware.
- **Scalable System**
  Can be deployed in farms, warehouses, retail stores, and food processing industries.
- **Reduction in Food Wastage**
  Early detection of rotten produce helps in preventing spoilage spread and economic losses.
- **Consistent Performance**
  Unlike humans, the model does not suffer from fatigue or inconsistency.

Disadvantages

- **Dependent on Image Quality**
  Poor lighting or low-resolution images may affect prediction accuracy.

- **Limited to Trained Categories**
  The model can classify only the fruits and vegetables included in the training dataset.

- **Initial Training Time**
  Model training requires computational resources and time.

- **Dataset Dependency**
  Performance depends on the quality and diversity of the dataset.

9. CONCLUSION

The Rotten Fruit and Vegetable Detection System using Transfer Learning successfully demonstrates the application of Deep Learning in solving real-world agricultural and food quality problems. The developed model accurately classifies fruits and vegetables as fresh or rotten based on image inputs.

By leveraging pretrained CNN architectures, the system achieves high performance while reducing training time and computational requirements. The implementation of a web-based interface ensures practical usability and real-time predictions.

This project contributes to reducing food wastage, improving quality control, and enhancing efficiency in agricultural supply chains. Overall, the system proves to be a reliable, scalable, and intelligent solution for automated food quality inspection.

10. FUTURE SCOPE

- **Multi-Class Classification**
  Extend the system to classify multiple categories of fruits and vegetables separately.
- **Mobile Application Development**
  Develop an Android/iOS app for real-time detection using smartphone cameras.
- **IoT Integration**
  Integrate with smart sorting machines and conveyor belt systems for automated industrial use.
- **Internal Quality Detection**
  Incorporate advanced imaging techniques such as hyperspectral imaging to detect internal spoilage.
- **Cloud-Based Deployment**
  Deploy the system on cloud platforms for large-scale commercial usage.
- **Real-Time Video Detection**
  Enhance the system to process live video streams instead of static images.
- **Integration with Supply Chain Management**
  Connect with inventory systems to automatically track and remove spoiled items.

11. APPENDIX Source Code (if any) Dataset Link, GitHub & Project Demo Link

GitHub link: https://github.com/rahul777-code

Dataset Link : https://www.kaggle.com/datasets/muhammad0subhan/fruit-and-vegetable-disease-healthy-vs-rotten