

Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables

Team Name:

“The Innovators”

Team Members:

1. Ysaswini Bandi
2. Yarram Veera Durga Dineshini
3. Yasam Gopalakrishna
4. Yarla Shasanth

Phase 1: Brainstorming & Ideation

Problem Statement: Usually, the classification of fresh and rotten fruits is carried by humans is not effectual for the fruit farmers. Human beings will become tired after doing the same task multiple times,

Proposed Solution: the project proposes an approach to reduce human efforts, reduce the cost and time for production by identifying the defects in the fruits in the agricultural industry

.Target Users: Agricultural farmers and agricultural researchers.

Expected Outcome: The project aims to automate the classification of fruits and vegetables into fresh or rotten categories using deep learning.

Phase 2: Requirement Analysis

Technical Requirements:

- Python
- TensorFlow & Keras
- Transfer Learning
- Google Colab
- Flask
- HTML/CSS
- Google Drive

Functional Requirements:

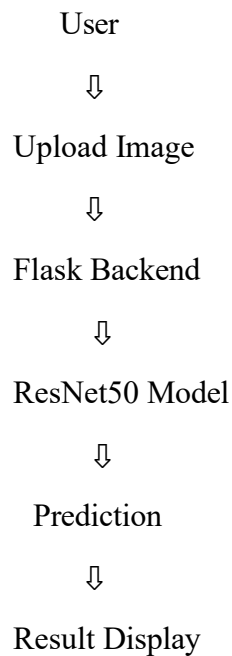
- Upload image
- Integrated the trained model to predict uploaded images.
- Display result

Constraints & Challenges:

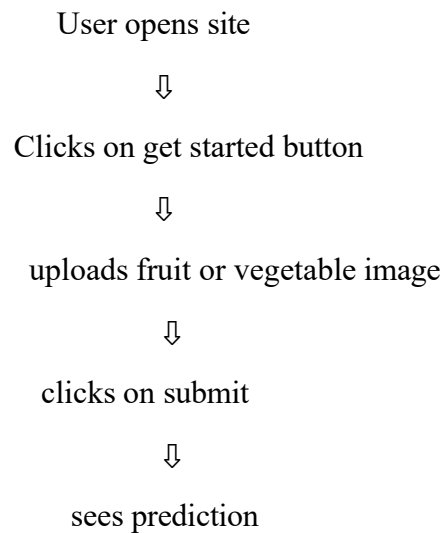
- Model accuracy depends on dataset quality

Phase 3: Project Design

System Architecture Diagram:



User Flow:



UI/UX Considerations:

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

Phase 4: Project Planning (Agile Methodologies)

- **Sprint Planning:**
 - Week 1: Dataset collection & preprocessing
 - Week 2: Model training and tuning
 - Week 3: Flask integration
 - Week 4: Frontend + Testing + Deployment
- **Task Allocation:**
 - Member A: Model training
 - Member B: Flask backend
 - Member C: Frontend UI
 - Member D: Documentation & Testing

- **Timeline & Milestones:**

- Milestone 1: Dataset ready (Week 1)
- Milestone 2: Model trained (Week 2)
- Milestone 3: Web integration (Week 3)
- Milestone 4: Testing + Report (Week 4)

Phase 5: Project Development

Technology Stack Used: Python, Flask, TensorFlow/Keras, ResNet50, HTML/CSS, Chatgpt

Development Process:

- Trained ResNet50 on poultry dataset
- Created app.py with prediction route
- HTML templates for UI
- Uploaded image saved and pre-processed
- Model predicts and result shown on predict.html

Challenges & Fixes:

- ResNet50 needed image shape fixing → solved with `img_to_array` & resizing
- File not saving correctly → fixed with `os.path.join()`
- Styling issues fixed via HTML template updates

Phase 6: Functional & Performance Testing

Test Cases Executed:

- Uploaded valid/invalid image formats
- Checked correct predictions for known test images
- UI responsiveness and error handling

Bug Fixes & Improvements:

- Fixed image upload not found bug
- Added image preview and file name display
- Improved styling for better UX

Final Validation:

- Project provide a simple yet powerful tool for food quality analysis
- Easy for non-technical users