**Project Design Phase**

| **Field** | **Details** |
| --- | --- |
| **Date** | **16-06-2025** |
| **Team ID** | **LTVIP2025TMID35102** |
| **Project Name** | **Smart Sorting: Detecting Rotten Fruits with Transfer Learning** |
| **Maximum Marks** | **2 Marks** |

**Solution Fit – Smart Sorting: Detecting Rotten Fruits with Transfer Learning**

The Problem–Solution Fit phase ensures that the machine learning model effectively addresses the real-world challenge of detecting rotten fruits—a problem faced by food suppliers, retailers, and warehouse managers. For the project *Smart Sorting: Detecting Rotten Fruits with Transfer Learning*, this phase validates that the deep learning-based classification system can enhance sorting accuracy, reduce manual effort, and minimize waste.

This phase ensures:

* Alignment of model predictions with quality inspection standards
* Understanding of user needs in sorting workflows and loss prevention
* Improvement in decision accuracy and sorting speed
* Early validation of prediction outputs before deployment at scale

**1. Target Customer Segments**

* Identifying the stakeholders using the fruit classification model is key to tailoring the deployment and UI:

| **Customer Type** | **Description** |
| --- | --- |
| Warehouse Managers | Require automation to separate rotten fruits before packaging or delivery |
| Food Retailers | Need consistent quality to prevent customer complaints and reduce waste |
| Sorting Line Operators | Use visual or software-based classification tools to streamline processes |
| Distributors & Exporters | Want fast, reliable sorting to maintain shipment quality standards |

**2. Problem Statement (As-Is Situation)**

Fruit inspection processes are often manual, inefficient, and lack consistency. The absence of scalable, automated tools leads to spoilage going undetected and significant revenue losses.

**Challenges Identified:**

* Human error in sorting due to fatigue or lighting conditions
* No standardized system for fruit quality classification
* Delayed shipment due to manual checking
* Wastage due to overlooked spoilage during visual inspection
* No real-time interface to aid workers in prediction

**Key Problems:**

* Inconsistent inspection standards
* High dependency on trained workers
* Inability to scale sorting in high-volume warehouses

**3. Current Workaround (Before ML Solution)**

| Existing Practice | Limitation |
| --- | --- |
| Manual sorting by workers | Inconsistent and prone to human error |
| Physical quality tags | Time-consuming and can be misplaced |
| Visual scanning under lights | Not reliable across different environments |
| Use of color-based sensors | Limited ability to detect early-stage rot or texture defects |

**4. Proposed Solution (To-Be State)**

The proposed solution—Smart Sorting with Transfer Learning—uses a pre-trained deep learning model (VGG16) fine-tuned for classifying fruit images as *fresh* or *rotten*. Integrated with a Flask-based web interface, the model allows users to upload fruit images and get real-time predictions.

Core Features Include:

* Pre-trained transfer learning model (VGG16) for high accuracy
* Flask web app to upload and classify fruit images
* Real-time prediction with confidence score display
* Output page with clear labels and visual feedback
* Exportable prediction logs for batch inspections
* Scalable architecture to integrate into conveyor or camera systems
* Support for additional fruits with further dataset training

**5. How the Solution Solves the Problem**

| Problem | ML/Flask Feature That Solves It |
| --- | --- |
| Manual sorting is slow and inaccurate | Automated image classification using VGG16 model |
| Lack of real-time visual inspection | Flask web interface delivers instant predictions on uploaded images |
| No consistency in labeling | Model provides objective, repeatable output |
| Missed spoilage during inspection | Trained on various stages of rot to detect subtle spoilage |
| No digital record of inspection | Logs and exportable results support traceability and QA |

**6. Solution Adoption Channels**

* Flask Web App: For standalone manual image inspection
* Camera-Based Integration: Attach camera on sorting line for real-time scanning
* API Service: REST API version for integration with mobile/IoT tools
* Backend Data Storage: Logs can be stored for audits or analytics

**7. Solution Validation**

The Smart Sorting system was validated through:

* Model Accuracy Testing: Precision, recall, and F1-score evaluated on test set
* UI Testing: Web interface tested by non-technical users for usability and clarity
* Speed Checks: Prediction completed within 1-2 seconds per image
* Edge Case Testing: Tested on images with lighting and background variations

All validations confirm that the solution supports efficient, consistent, and real-time classification of fruits.

| Benefit | How Smart Sorting Achieves It |
| --- | --- |
| Real-time quality control | Predictions in seconds with clear, actionable results |
| Reduced sorting effort | Automates visual inspection and minimizes manual workload |
| Minimized food waste | Early rot detection prevents spoilage from reaching customers |
| Deployment-ready | Can be extended for conveyor cameras or mobile apps |
| Supports scale-up | Easily expandable to new fruit types with model retraining |