Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	29 June 2025
Team ID	LTVIP2025TMID41462
Project Name	Smart Sorting : Transfer Learning For Identifying Rotten Fruits And Vegetables
Maximum Marks	

Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1 .	Image Upload Interface	User can upload fruit or vegetable images through a clean, responsive UI
FR-2	Image Pre processing	Automatically resizes and scales input images to the required dimensions (224×224).
FR-3	Classification Model Integration	Predicts one of 4 classes: fresh fruit, fresh vegetable, rotten fruit, rotten vegetable.
FR-4	Result Display	Displays the predicted category along with the uploaded image for user confirmation.
FR-5	Dataset Handling	Balances training and testing dataset from 12 folders into 4 target classes with equal distribution.
FR-6	File Upload & Backend Connectivity	Uses Flask to handle image uploads and run model inference seamlessly in the background.
FR-7	Model Storage & Loading	Saves trained model as `.h5` file and reloads it during real-time prediction.

Non-Functional Requirements:

NFR No.	Non-Functional Requirement	Description
NFR-1	Accuracy	Target model accuracy between 90% to 95% to ensure reliable predictions.
NFR-2	Performance	Prediction response time must be under 2 seconds for a seamless user experience
NFR-3	Scalability	Should support future addition of more classes (e.g., leaf diseases, other produce).
NFR-4	Usability	Intuitive interface accessible to both farmers and sellers with no technical background.
NFR-5	Portability	Flask app must run on any system with Python & TensorFlow installed (cross-platform).
NFR-6	Accessibility	Web app should display well on laptops, tablets, and smartphones (mobile responsive).
NFR-7	Data Integrity	Ensures correct class mapping from 12 folders into 4 classes; no misclassification in datasets.