

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

Date	19 February 2026
Team ID	LTVIP2026TMIDS83873
Project Name	Smart-Sorting-Transfer-Learning-forIdentifying Rotten-Fruits-and-Vegetables
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

Functional Requirements

FR No.	Sub Requirement (Story / Sub-Task)
FR-1	The user shall be able to upload an image of a fruit or vegetable through the web interface.
FR2	The system shall preprocess the uploaded image (resize to 224x224, normalization, formatting) before prediction.
FR-3	The system shall classify the image as either Fresh or Rotten using the pre-trained VGG16 transfer learning model .
FR-4	The system shall provide prediction results in real-time (within a few seconds).
FR-5	Uploaded images shall be temporarily stored in the static/uploads folder for rendering results.
FR-6	The system shall allow users to upload different fruit and vegetable categories (28 supported categories).

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

NFR-1	The system should provide a clean, simple, and responsive user interface using HTML and CSS.
NFR-2	The system should deliver predictions with an average response time of less than 3 seconds .
NFR-3	The trained model should maintain a minimum validation accuracy of 90% (Current: ~92%).
NFR-4	The application should be deployable on public cloud platforms such as Render .
NFR-5	The system should ensure high availability with minimal downtime (target 99.9% uptime when deployed).
NFR-6	The application must be compatible with modern web browsers (Chrome, Edge, Firefox).
NFR-7	The system should securely handle uploaded images without permanently storing user data.

System Requirements

S.No	Component	Details
SR-1	Programming Language	Python 3.10+

NFR No.	Description
---------	-------------

SR-2 Framework	Flask
SR-3 Model Format	.h5 (Keras / TensorFlow SavedModel)
SR-4 Libraries Used	TensorFlow, Keras, NumPy, Flask
SR-5 Frontend	HTML, CSS
SR-6 Deployment	Render (via GitHub repo)
SR-7 RAM/Memory (for local runs)	Minimum 4GB recommended