Solution Requirements

Date	31 January 2025
Team ID	LTVIP2025TMID60812
Project Name	Grain Palette - A Deep Learning Odyssey In Rice Type Classification Through Transfer Learning
Maximum Marks	4 Marks

Solution Requirements

□ 1. Functional Requirements

These are the core features and capabilities that the system must provide.

ID	Requirement
FR1	The system shall allow users to upload rice grain images.
FR2	The system shall preprocess input images (resize, normalize, augment).
FR3	The system shall classify the rice variety using a trained ML model.
FR4	The system shall return the predicted rice type and its confidence score.
FR5	The system shall store classification history for users (optional).
FR6	The system shall display performance metrics (accuracy, confusion matrix).
FR7	The system shall allow batch predictions (multiple images at once).
FR8	The system shall enable users to provide feedback on prediction correctness.
FR9	The system shall export results to reports or dashboards (e.g., Power BI).
FR10	The system shall support user authentication (for premium access, if needed).

□□ 2. Non-Functional Requirements

These define the quality attributes and operational constraints of the system.

ID	Requirement
NFR1	The system shall respond to a single image prediction within 2 seconds.
NFR2	The system shall have at least 90% classification accuracy on the test dataset.
NFR3	The system shall be accessible via web browser or mobile app.
NFR4	The system shall be lightweight enough to run on low-resource devices.
NFR5	The system shall be usable offline with a preloaded model (optional).
NFR6	The system shall handle at least 100 concurrent image predictions.
NFR7	The system shall ensure image data privacy and comply with data standards.
NFR8	The system shall use interpretable AI techniques (e.g., show confidence scores).
NFR9	The system shall be easily retrainable with new rice varieties in future.
NFR10	The user interface shall support multilingual text (e.g., English, Hindi, Tamil).

□□ 3. Technical Requirements

Component	Details
Programming Language	Python (preferred), JavaScript (for front-end if web-based)
Frameworks	TensorFlow/Keras or PyTorch, Flask/Streamlit for UI
ML Models	Transfer Learning with EfficientNetB0, ResNet50, etc.
Deployment	Local system, Cloud (AWS, Azure, or Google Cloud), or Mobile APK
Storage	SQLite or Firebase for storing user data and classification history
Visualization	Power BI, Matplotlib, Seaborn for dashboard and report generation