User Acceptance Testing (UAT) Template

Date	19/05/2025-30/6/2025
Team ID	LTVIP2025TMID39191
Project Name	GrainPalette - A Deep Learning Odyssey In Rice Type Classification Through Transfer Learning
Maximum Marks	

Project Overview

- **Project Name:** Rice Type Identification using AI
- **Project Description:**
  This AI project helps farmers and agriculture professionals identify different rice types through image classification. Built with CNN and MobileNetv4, users upload images to receive predictions, enabling better decisions on cultivation and resource management.
- **Project Version:** v1.0
- **Testing Period:** 22 JUNE 2025 to 27 JUNE 2025

Testing Scope

Features and Functionalities to be Tested:

- Image upload interface
- Model prediction on uploaded rice image
- Display of predicted rice type with confidence
- Responsive interface

User Stories / Requirements to be Tested:

- As a farmer, I want to identify the rice type by uploading an image
- As a researcher, I want quick identification in the field
- As a home gardener, I want to explore rice variety

Testing Environment

- **URL/Location:** https://github.com/masthan07/Rice_classification

- **Credentials (if required):** Not Required

Test Cases:

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Pass/Fail
TC-001	Upload image of rice grain	1. Open app 2. Upload a rice grain image 3. Click Submit	The predicted rice type is displayed with confidence	Prediction displayed correctly	Pass
TC-002	Upload an invalid file format	1. Open app 2. Upload a non-image file (e.g., .txt) 3. Click Submit	Error message indicating invalid file type	Error shown as expected	Pass

Bug Tracking:

Bug ID	Bug Description	Steps to reproduce	Severity	Status	Additional feedback
BG-001	No message when no image is uploaded	1. Open app 2. Click submit	Low	Closed	Validation adde
BG-002	Minor UI overlap on mobile view	1. Open app on small screen device	Medium	In Progress	May need responsive CSS tweaks

Sign-off:

Tester Name: Rufus raj dandangi

Date: 27 JUNE 2025

Notes:

- Ensure that all test cases cover both positive and negative scenarios.
- Encourage testers to provide detailed feedback, including any suggestions for improvement.
- Bug tracking should include details such as severity, status, and steps to reproduce.