1. INTRODUCTION

1.1 Project Overview

Rice Type Detection is a machine learning-based system designed to identify different types of rice grains from images. This helps in improving quality control in the agriculture and food industry.

1.2 Purpose

The purpose of the project is to provide a fast, accurate, and user-friendly web-based tool that classifies rice types from uploaded images, using deep learning and computer vision techniques.

---

2. IDEATION PHASE

2.1 Problem Statement

Automate the identification of rice types from images to assist in quality control and classification in the agriculture and food industry.

2.2 Empathy Map Canvas

- **Users**: Food industry professionals, rice distributors, researchers.
- **Needs**: Fast and reliable rice classification.
- **Pains**: Manual classification is time-consuming and error-prone.
- **Gains**: Automation improves efficiency and accuracy.

2.3 Brainstorming

- Use CNN models for classification.
- Build a web interface for uploading and predicting.
- Ensure the system works for common rice types (e.g., Basmati, Jasmine, Brown).

---

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. User opens the web app.
2. Uploads an image of rice grains.
3. System processes the image.

4. Result is displayed (e.g., "Basmati").

3.2 Solution Requirement

- **Technical**: Python, Flask, PyTorch
- **Functional**:
  - Upload rice grain images
  - Classify the rice type
  - Display prediction result
- **Constraints**:
  - Large dataset handling
  - High accuracy needed

3.3 Data Flow Diagram

(Insert DFD image showing: User -> Upload Image -> ML Model -> Classification -> Display Result)

3.4 Technology Stack

- **Backend**: Python, Flask
- **ML Libraries**: PyTorch, Torchvision, NumPy, Pillow
- **Frontend**: HTML, CSS

---

4. PROJECT DESIGN

4.1 Problem Solution Fit

The solution aligns with the problem of manual rice classification by providing an automated, scalable tool.

4.2 Proposed Solution

A deep learning model integrated into a web application where users can upload images to detect rice type.

4.3 Solution Architecture

(Insert diagram showing: User Interface → Web Server → CNN Model → Output Display)

---

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- **Sprint 1**: Data preparation
- **Sprint 2**: Model training
- **Sprint 3**: Web development
- **Sprint 4**: Testing and Integration

---

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- **Test Case 1**: Basmati Rice → Expected: Basmati → Actual: Basmati ✅
- **Test Case 2**: Jasmine Rice → Expected: Jasmine → Actual: Jasmine ✅
- **Test Case 3**: Brown Rice → Expected: Brown → Actual: Brown ✅
- **Bug Fixes**:
  - Improved model prediction accuracy
  - Fixed image upload issues

---

7. RESULTS

7.1 Output Screenshots

(Insert screenshots showing uploaded images and predicted rice types)

---

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Fast and automated classification
- Easy-to-use web interface
- Scalable and cloud-deployable

Disadvantages:

- May not recognize unseen rice types
- Depends on image quality

---

9. CONCLUSION

This project demonstrates how machine learning can automate rice classification, offering a real-world application for agriculture and food industries.

---

10. FUTURE SCOPE

- Expand to detect more rice types
- Improve accuracy with larger datasets
- Mobile app version
- Add multilingual support

---

11. APPENDIX

- **Source Code**: https://github.com/sandya2611/Rice-Type-Dectection/tree/main/Project%20Files
- **Dataset Link**: https://www.kaggle.com/datasets/muratkokludataset/rice-image-dataset
- **GitHub & Project Demo Link**:
  - GitHub: https://github.com/sandya2611/Rice-Type-Dectection
  - Demo Video:

    ---

    https://github.com/sandya2611/Rice-Type-Dectection/tree/main/Video%20Demo