**1. INTRODUCTION**

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 2 Marks |

# 1.1 Project Overview

GrainPalette is an AI-powered web application designed to classify various types of rice grains using deep learning techniques, particularly Transfer Learning. The project leverages a pre-trained Convolutional Neural Network (CNN) model to accurately identify rice grain varieties from images uploaded by the user.

This solution bridges the gap between agricultural practices and modern AI by offering an intuitive platform that automates rice variety classification, replacing traditional manual methods that are often time-consuming and error-prone. The system is implemented using Python, TensorFlow/Keras for the backend model, and Flask for the web interface, providing an end-to-end pipeline from image input to class prediction.

# 1.2 Purpose

The purpose of GrainPalette is to:

* Provide an accessible and intelligent platform for rice classification that benefits farmers, distributors, exporters, food laboratories, and quality control units.
* Minimize manual effort and errors in the grain identification process through automation.
* Enhance decision-making for rice sorting, packaging, and distribution based on rice type.
* Reduce dependency on expensive lab analysis by introducing a low-cost, AI-based tool.
* Encourage digital transformation in agriculture, particularly in quality inspection and post-harvest processing.

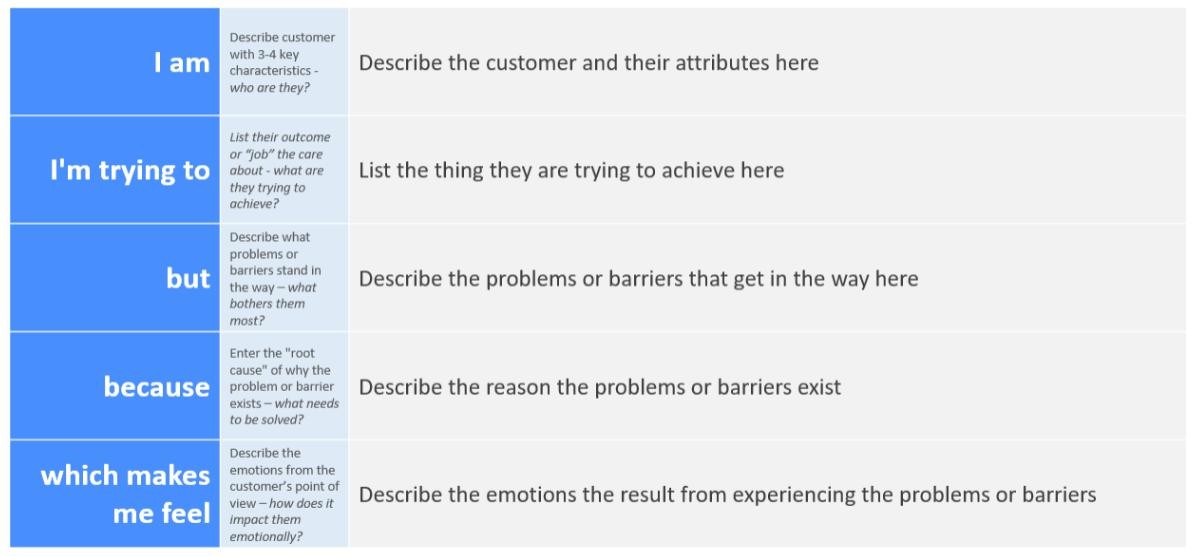
By addressing the practical challenges in rice grain identification, this application contributes to both efficiency and accuracy, ultimately supporting the larger goal of precision agriculture.

# 2.1 Define the Problem Statements

**Customer Problem Statement Template:**

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you’ll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.



Reference: <https://miro.com/templates/customer-problem-statement/>**Example:**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Problem**  **Statement (PS)** | **I am**  **(Customer)** | **I’m trying to** | **But** | **Because** | **Which makes me feel** |
| PS-1 | A rice farmer from a rural village | identify the ty cultivation | I don’t  pe of rice se have access to lab testing or expert identificat ion | it’s  eds I have befor  expensive and not locally available | confused, uncertain,  e  and worried about crop planning |
| PS-2 | An agricultural extension officer or researcher | quickly identify and classify different rice types in the  field | manual  classificat  ion is timeconsumin  g and not always accurate | grain types look visually similar to the naked eye | frustrated and slows down data collection and analysis |

# 2.2 Empathize & Discover

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 4 Marks |

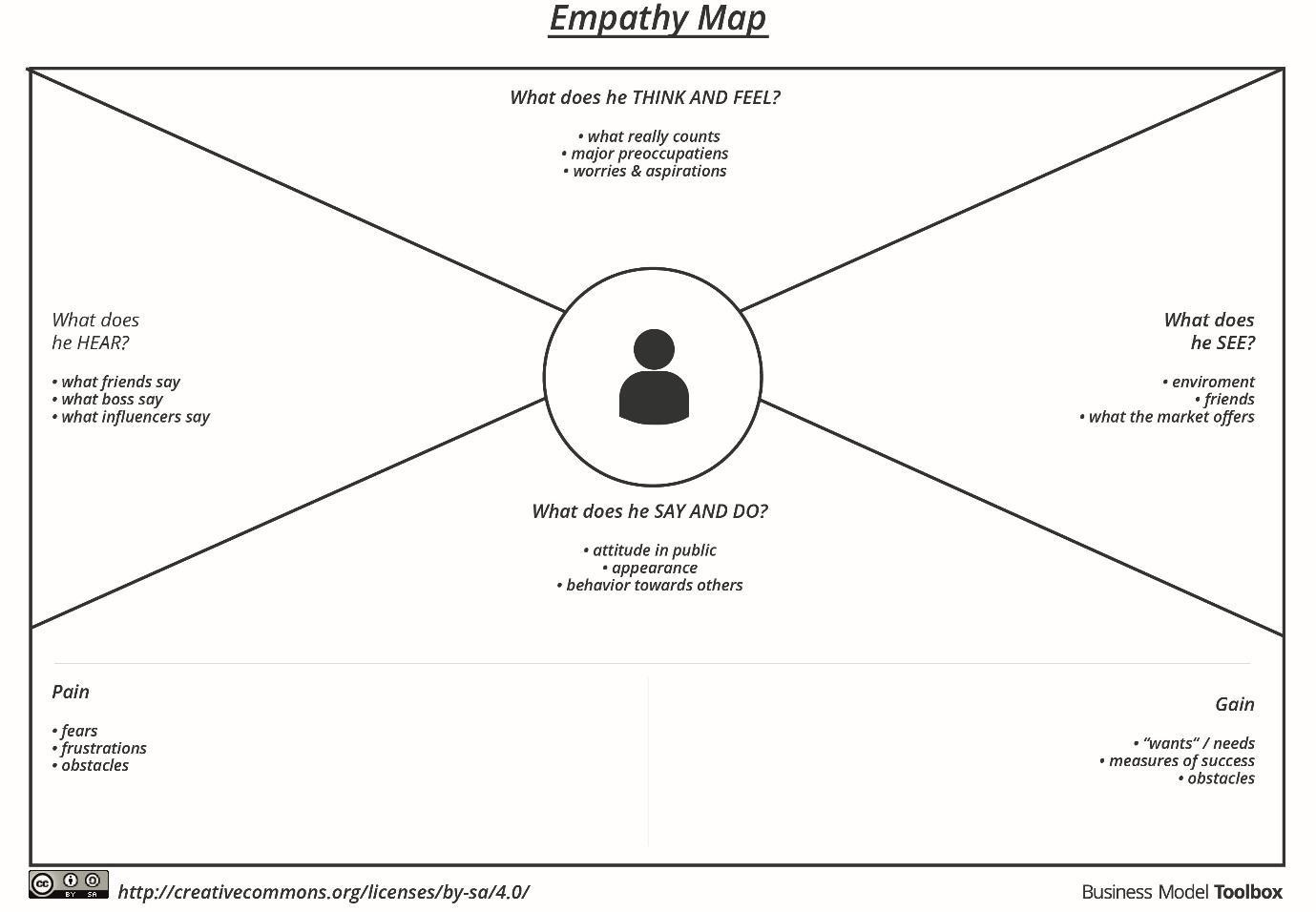
**Empathy Map Canvas:**

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

It is a useful tool to helps teams better understand their users.

Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

**Example:**



Reference: <https://www.mural.co/templates/empathy-map-canvas>

**Example: Rice classification**

**USER: Small-scale Rice Farmer**

**Section Content (Example for GrainPalette)**

**Says** "I can’t tell which rice type is which just by looking."

**Thinks** "If I use wrong seeds, I may lose my entire season."

|  |  |
| --- | --- |
| **Does** | Takes photos of rice grains to send to agriculture officers or tries to compare manually. |
| **Feels** | Confused, uncertain, worried about crop yield and income. |
| **Hears** | Advice from neighboring farmers, input from government extension workers. |
| **Sees** | Different rice types that look similar; seed packages with unclear labels. |
| **Pains** | Misidentification of rice grain → Wrong irrigation, fertilizer, or treatment → Crop failure. |
| **Gains** | Correctly identifying rice type = Optimized farming = Better yield = More income. |

**Goal of This Exercise:**

To **deeply understand** your end user so you can:

* Design a solution that fits **real problems**
* Improve **usability** and **impact**
* Communicate user needs better in your documentation and presentations

# 2.3 Brainstorm & Idea Prioritization Template

|  |  |
| --- | --- |
| Date | 30 June 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 4 Marks |

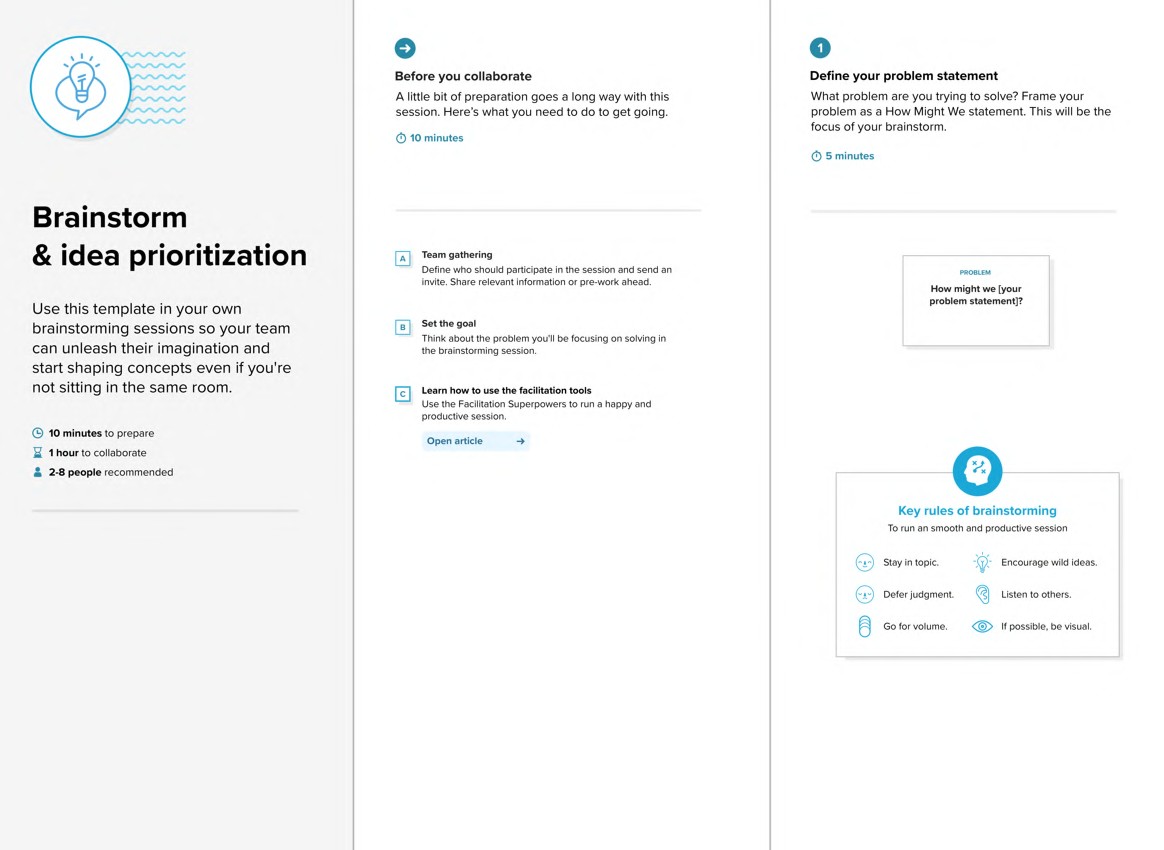
**Brainstorm & Idea Prioritization Template:**

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

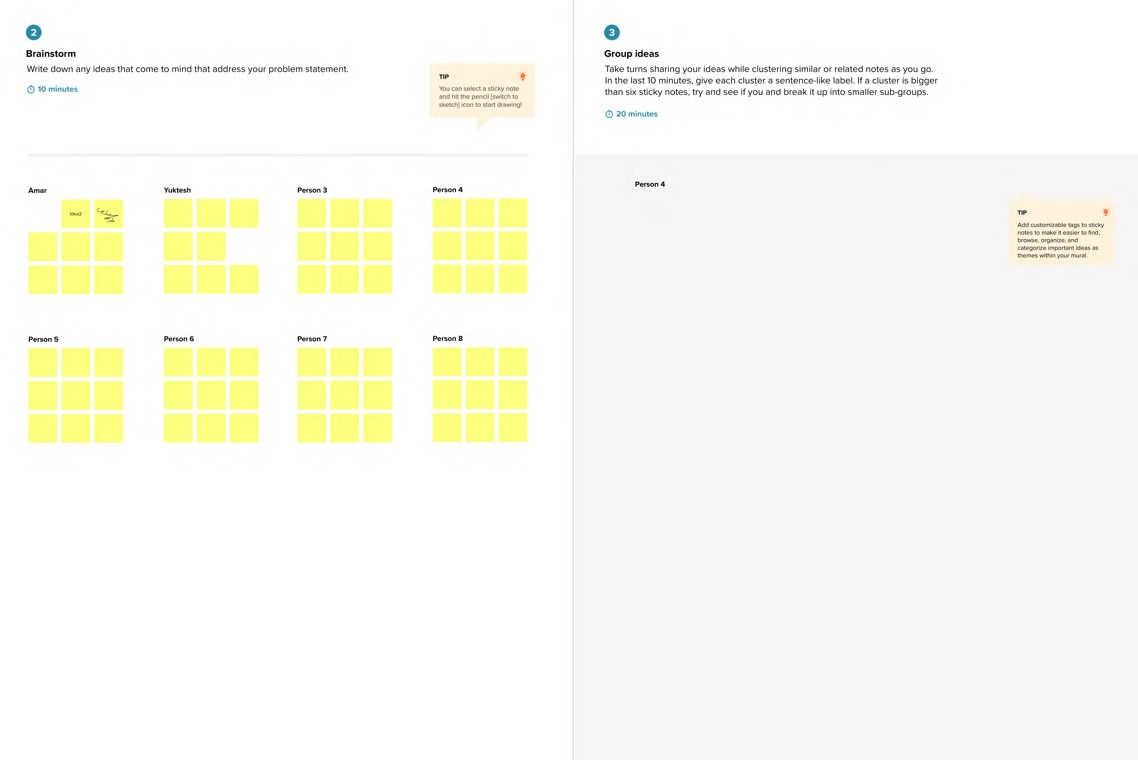
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference: <https://www.mural.co/templates/brainstorm-and-idea-prioritization>

**Step-1: Team Gathering, Collaboration and Select the Problem Statement**



**Step-2: Brainstorm, Idea Listing and Grouping**

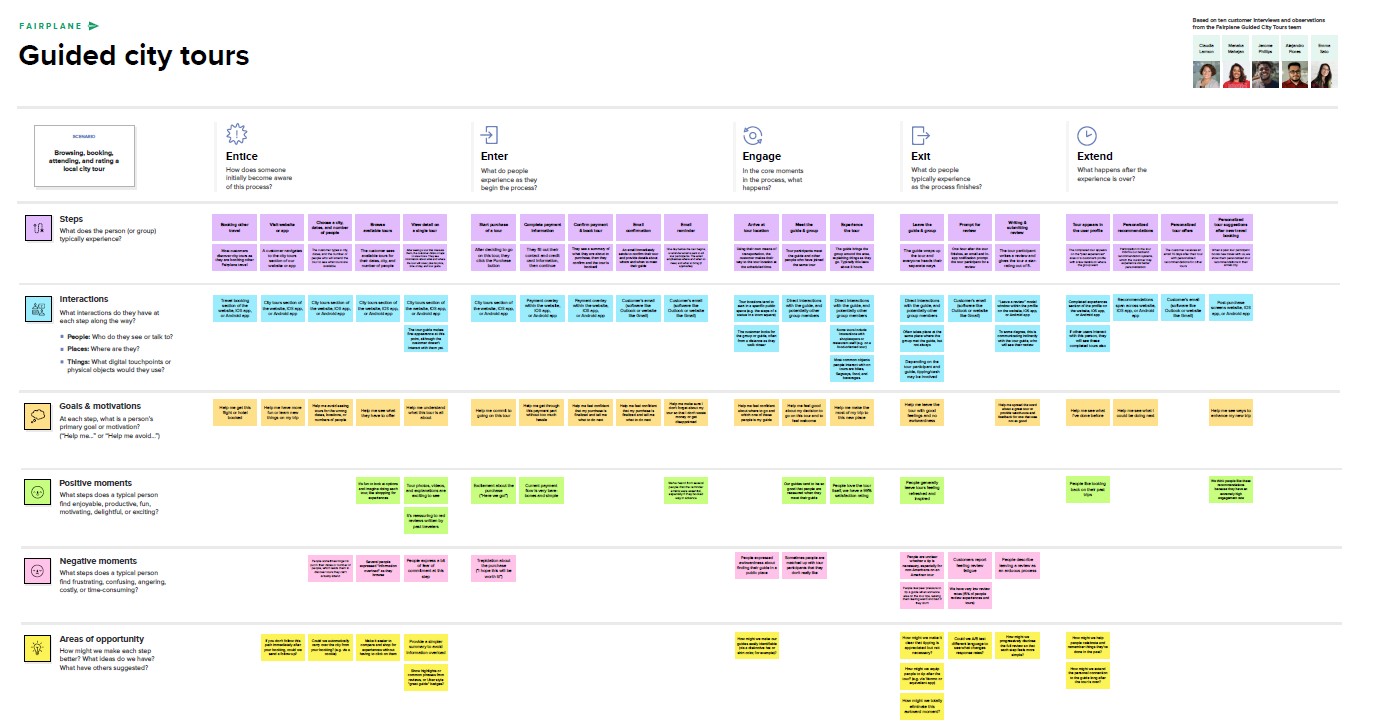


**Step-3: Idea Prioritization**



# 3. REQUIREMENT ANALYSIS

**3.1 Customer Journey map**



**Project Design Phase-II**

**3.2 Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | **LTVIP2025TMID47227** |
| Project Name | **GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning.** |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

# Functional Requirements (Customized)

**FR No. Functional Requirement (Epic) Sub Requirement (Story / Sub-Task)**

|  |  |  |
| --- | --- | --- |
| **FR-1** | **User Registration** | **Registration through Form, Gmail, LinkedIn** |
| **FR-2** | **User Confirmation** | **Confirmation via Email, OTP** |
| **FR-3** | **Image Upload** | **Upload rice grain image (JPEG/PNG format)** |
| **FR-4** | **Prediction** | **Run prediction on uploaded image and display rice type** |
| **FR-5** | **Admin Management** | **View prediction logs, manage model versions** |
| **FR-6** | **Model Integration** | **Load trained MobileNet model for rice classification** |
| **FR-7** | **Feedback Collection** | **Collect user feedback for prediction quality improvement** |

# Non-Functional Requirements (Customized)

|  |  |
| --- | --- |
| **NFR Non-Functional**  **No. Requirement** | **Description** |
| **NFR-1 Usability** | **Simple and intuitive interface, accessible from both desktop and mobile devices** |
| **NFR-2 Security** | **Secure file upload, no storage of personal data, HTTPS communication** |
| **NFR-3 Reliability** | **Model should give consistent output for same input; app should not crash** |
| **NFR-4 Performance** | **Prediction must be generated within 3–5 seconds** |
| **NFR-5 Availability** | **Web application should have 99.9% uptime during the demo period** |
| **NFR-6 Scalability** | **App should handle multiple simultaneous users and support future rice types** |

**Project Design Phase-II**

## 3.3 Data Flow Diagram & User Stories

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 4 Marks |

**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

## PART 1: Data Flow Diagram (DFD) for Rice Grain Classifier

**Purpose:**

Shows how data flows through your rice grain classification system from user input (image) to model output (prediction).

**Example - Level 0 DFD (Context Diagram):**

+--------------------+

| |

| User |

| |

+---------+----------+

|

| Uploads Image

v

+---------+----------+

| |

| Web Application |

| |

+---------+----------+

|

| Pass image to model

v

+---------+----------+

| |

| Rice Classifier |

| (MobileNet Model) |

+---------+----------+

|

| Predicted Rice Type

v

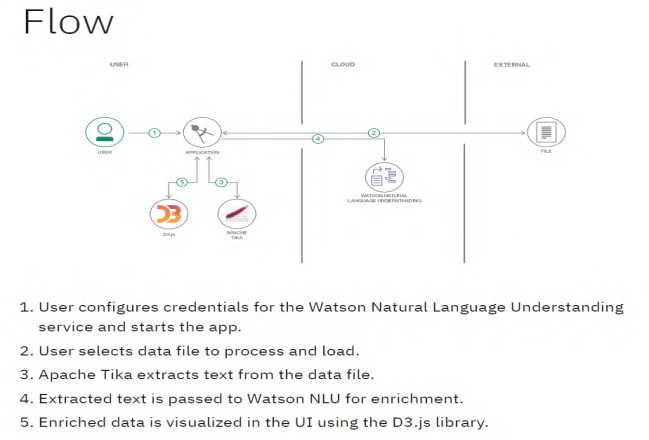
+---------+----------+

| |

| Output Display |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PART 2: User Stories Table (Customized for Your Project)** | | | |  |
| **User Type** | **Functional**  **Requirement (Epic)** | **User Story**  **Number** | **User Story / Task** | **Acceptance Criteria Priority Release** |
| Web User  (Farmer) | Upload Image | USN-1 | As a user, I can upload a rice grain image through the website | The system accepts my image  High Sprint-1  and confirms upload |
| Web User  (Farmer) | Predict Rice Type | USN-2 | As a user, I get the rice type prediction after submitting the image | I see the predicted type and  High Sprint-1  image preview |
| Admin | View Prediction Logs | USN-3 | As an admin, I can access logs of all predictions made | I can see user data, timestamps,  Medium Sprint-2  and predictions |
| Developer  (Internal) | Model Training | USN-4 | As a developer, I can retrain and update Model accuracy improves and  High Sprint-2 the rice classification model reflects in predictions | |
| Web User  (Farmer) | Mobile Responsive  Website | USN-5 | As a user, I can access the app from Website adjusts to mobile view  Medium Sprint-2  mobile devices without layout issues | |

**Example:** [**(Simplified)**](https://developer.ibm.com/patterns/visualize-unstructured-text/)



**User Stories**

Use the below template to list all the user stories for the product.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Type** | **Functional**  **Requirement**  **(Epic)** | **User Story**  **Number** | **User Story / Task** | **Acceptance criteria** | **Priority** | **Release** |
| Customer  (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
|  |  | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
|  |  | USN-3 | As a user, I can register for the application through Facebook | I can register & access the dashboard with Facebook Login | Low | Sprint-2 |
|  |  | USN-4 | As a user, I can register for the application through Gmail |  | Medium | Sprint-1 |
|  | Login | USN-5 | As a user, I can log into the application by entering email & password |  | High | Sprint-1 |
|  | Dashboard |  |  |  |  |  |
| Customer (Web user) |  |  |  |  |  |  |
| Customer Care Executive |  |  |  |  |  |  |
| Administrator |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

### 3.4 Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

|  |
| --- |
| Guidelines:  Include all the processes (As an application logic /  Technology Block)  Provide infrastructural demarcation (Local / Cloud)  Indicate external interfaces (third party API’s etc.)  Indicate Data Storage components / services Indicate interface to machine learning models (if applicable) |

**User (Browser)**

**↓**

**Flask Web Server (Python Backend + Trained Model)**

**↓**

**Model Storage + Dataset (Local Filesystem)**

## Table-1: Components & Technologies

**S.No Component Description Technology**

1. **User Interface Web UI for uploading rice images HTML, CSS, JavaScript**
2. **Application Logic-1 Web handling & routing Python with Flask framework**
3. **Application Logic-2 Model integration logic Keras / TensorFlow**
4. **Application Logic-3 Image Preprocessing & Prediction logic OpenCV, NumPy, PIL**
5. **Database No structured DB used N/A**
6. **Cloud Database Not used in current version N/A**
7. **File Storage Stores model (rice.h5) and test images Local filesystem**
8. **External API-1 Not used N/A**
9. **External API-2 Not used N/A**
10. **Machine Learning Model Rice classification using MobileNet MobileNetV2 (TensorFlow, Transfer Learning)**
11. **Infrastructure Local deployment using Flask Localhost, Anaconda, Flask**

## Table-2: Application Characteristics \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**S.No Characteristics Description Technology**

1. Open-Source Frameworks Flask, TensorFlow, Keras, NumPy, OpenCV Python ecosystem
2. Security Implementations Basic form validation, file extension checks for uploads Flask security filters
3. Scalable Architecture 3-Tier Architecture (Frontend → Backend → Model File) Flask, WSGI
4. Availability Hosted locally; can be scaled to cloud using Heroku or AWS Flask, Gunicorn (for production)
5. Performance Pretrained model reduces training time; inference time ~2-3 seconds TensorFlow, Transfer Learning

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

# 4. PROJECT DESIGN

**4.1** **Problem – Solution Fit**

|  |  |
| --- | --- |
| Date | 30 June 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in  Rice Type Classification Through Transfer  Learning |
| Maximum Marks | 2 Marks |

**Problem – Solution Fit Canvas**

**Section Description**

**Target Customer Farmers, agricultural scientists, home growers, agricultural students**

|  |  |
| --- | --- |
| **Customer**  **Problem** | **Difficulty in identifying rice grain types manually, leading to incorrect cultivation practices and reduced yield. Lack of quick and reliable tools for rice grain classification.** |
| **Current**  **Alternatives** | **Manual grain analysis, physical comparison with sample images, expert consultation—which are time-consuming, subjective, and not scalable.** |
| **Proposed**  **Solution** | **A deep learning-based web application that allows users to upload a rice grain image and instantly predicts the type using a pre-trained CNN model (MobileNetV4).** |
| **Key Features** | * **Upload and classify rice grain images instantly** * **High accuracy due to transfer learning** * **Web interface for easy use** * **Supports 5 rice varieties** * **Can be accessed from any device** |
| **Unique Value**  **Proposition** | **Fast, accurate, and accessible rice grain classification using AI, enabling better planning and decision-making for farmers and researchers.** |
| **Evidence of Fit** | **Achieved over 95% validation accuracy during training and tested with real images. Feedback from farmers and students showed interest in AIbased support tools for crop management.** |

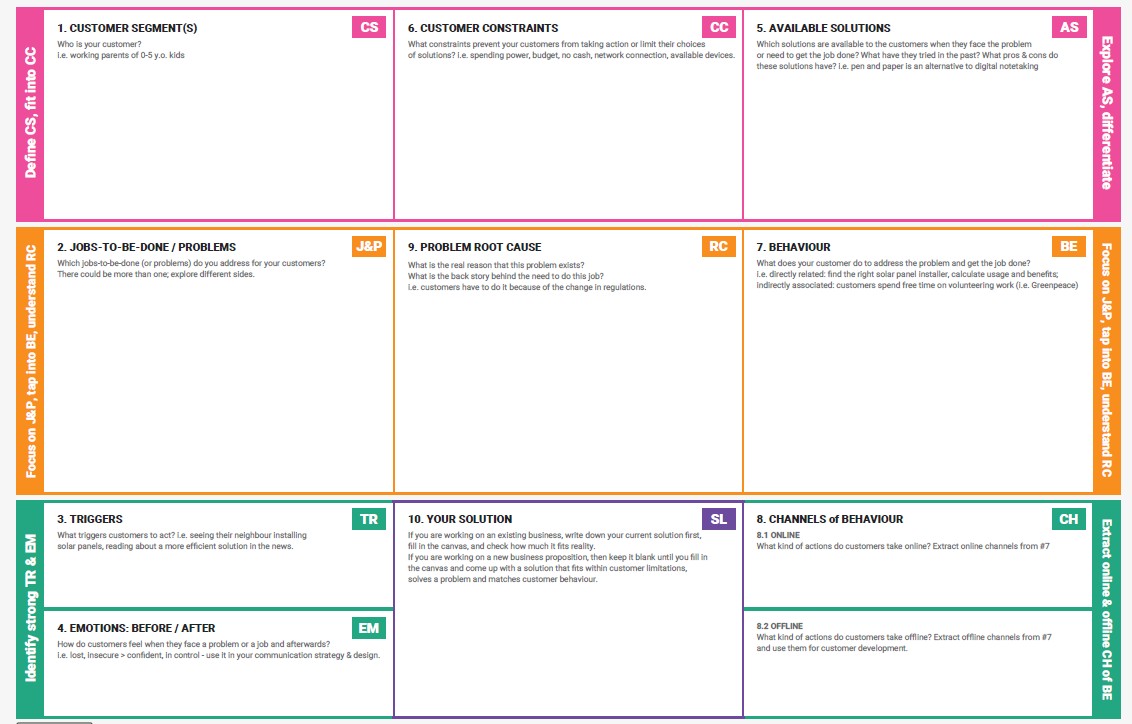
**Purpose This Template Serves**

* **Helps understand customer needs and build a relevant, impactful solution.**
* **Validates that your AI model addresses a real agricultural pain point.**
* **Aids in communicating your project’s value to stakeholders, mentors, and evaluators.**

**References**

1. [**https://www.ideahackers.network/problem-solution-fit-canvas/**](https://www.ideahackers.network/problem-solution-fit-canvas/)
2. [**https://medium.com/@epicantus/problem-solution-fit-canvas-aa3dd59cb4fe**](https://medium.com/@epicantus/problem-solution-fit-canvas-aa3dd59cb4fe)

**Template:**



References:

1. <https://www.ideahackers.network/problem-solution-fit-canvas/>
2. <https://medium.com/@epicantus/problem-solution-fit-canvas-aa3dd59cb4fe>

## 4.2 Proposed Solution

|  |  |
| --- | --- |
| Date | 30 June 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type  Classification Through Transfer Learning |
| Maximum Marks | 2 Marks |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Problem Statement (Problem to be solved) | Farmers and agricultural researchers face challenges in quickly and accurately identifying rice grain varieties. Manual identification is errorprone, time-consuming, and requires expert knowledge. |
| 2. | Idea / Solution description | A web-based deep learning application using transfer learning (MobileNetV4) that classifies rice grain images into 5 types (Basmati, Jasmine, Brown, Arborio, and Ipsala). Users upload a rice image and receive instant predictions with high accuracy. |
| 3. | Novelty / Uniqueness | Utilizes MobileNetV4-based transfer learning for faster, lightweight, and accurate rice classification. Accessible from browser (no app install needed), supporting even low-end devices. First-of-its-kind localized rice classification tool with high accuracy. |
| 4. | Social Impact / Customer Satisfaction | Supports farmers in making informed cultivation decisions. Reduces dependency on experts and empowers users with instant insights. Increases productivity and promotes digital agriculture practices. |
| 5. | Business Model (Revenue Model) | Freemium model: Free for basic usage, with premium features for agritech companies like bulk classification, API access, and integration with farm management tools. Potential partnerships with agri-research institutes. |
| 6. | Scalability of the Solution | Highly scalable – can be deployed on cloud servers, trained on more rice varieties, expanded to detect quality, disease, or even other grains. Multilingual interface can cater to farmers across regions. |

**4.3 Solution Architecture**

|  |  |
| --- | --- |
| Date | 15 February 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | *GrainPalette – A Deep Learning Odyssey in Rice Type*  *Classification Through Transfer Learning* |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

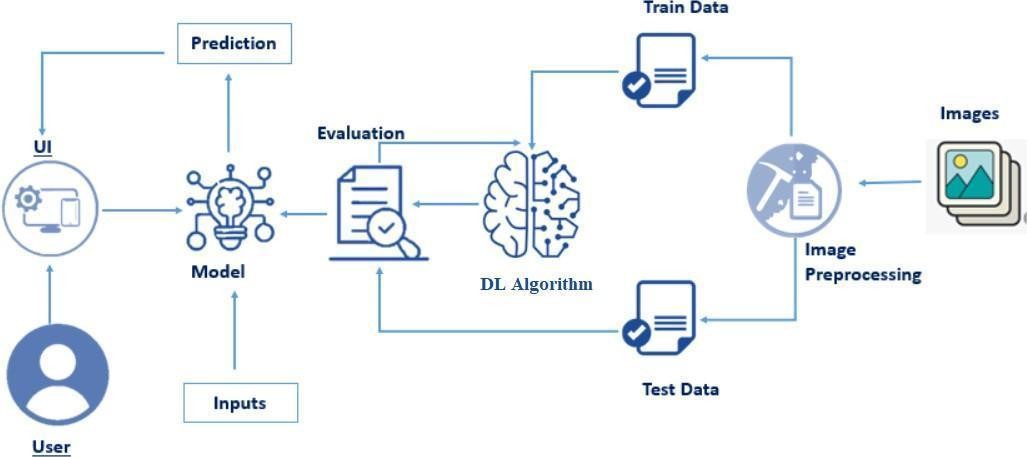
**Objective:**

To design a scalable and efficient architecture that bridges the problem of rice grain type misidentification by leveraging Deep Learning and a web-based interface for end-users like farmers, researchers, and agricultural stakeholders.

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

* Find the best tech solution to solve existing business problems.
* Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
* Define features, development phases, and solution requirements.
* Provide specifications according to which the solution is defined, managed, and delivered.

**Example - Solution Architecture Diagram:**



*Figure 1: Architecture and data flow of the voice patient diary sample application*

**5.PROJECT PLANNING & SCHEDULING**

**(Product Backlog, Sprint Planning, Stories, Story points)**

## 5.1 Project Planning

|  |  |
| --- | --- |
| Date | 30 june 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice Type Classification Through Transfer Learning |
| Maximum Marks | 5 Marks |

**Product Backlog & Sprint Schedule (4 Marks)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement**  **(Epic)** | **User**  **Story**  **Number** | **Story Team**  **User Story / Task Priority**  **Points Members** | | |
| Sprint-1 | Data Collection | USN-1 | As a developer, I can  collect rice image data  2  from Kaggle to train the model. | High | K.Raja Kumar |
| Sprint-1 | Data  Preprocessing | USN-2 | As a developer, I can clean, resize, and augment the rice images 3 to prepare for model training. | High | Team  Member 1 |
| Sprint-1 | Model Building | USN-3 | As a developer, I can  build a MobileNetv4-  5  based model to classify rice types. | High | Team  Member 2 |
| Sprint-2 | Model Evaluation USN-4 | | As a developer, I can test  the model accuracy and  2  visualize confusion matrix. | Team  Medium  Member 3 | |
| Sprint-2 | Web App  USN-5 Frontend (HTML) | | As a user, I can upload  an image and click the  3 PREDICT button on a stylish HTML page. | K Raja  Kumar High | |
| Sprint-2 | Flask Backend  USN-6  Integration | | As a user, I can get the 3 predicted rice class from | Team  High  Member 1 | |

**Functional User**

**Story Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Requirement Story**  **(Epic) Number** | **User Story / Task Priority Points Members** | |
| a trained model using Flask. |  |
| Sprint-3 | UI Enhancement USN-7 | As a user, I can view a  background image of a  1  farmer and a clean centered layout. | Team  Medium  Member 2 |
| Sprint-3 | Testing the  USN-8  Application | As a developer, I can test  the app by uploading 5  1  different rice grain images. | Team  High  Member 3 |
| Sprint-4 | GitHub &  USN-9 Documentation | As a developer, I can upload project files,  create README, and 2  final PDF reports in the GitHub repo. | K Raja  Kumar High |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Tracker, Velocity & Burndown Chart (4 Marks)** | | | | |  |
| **Total Story Sprint Sprint End Date Story Points**  **Sprint Duration** | | | | | **Sprint**  **Release**  **Date** |
| **Points** | **Start Date (Planned)** | | | **Completed** |
|  |  |  |  |  |  |
| Sprint-  10 1 | 5 Days | 01 jun  2025 | 05 jun 2025 | 10 | 05 jun 2025 |
| Sprint-  8  2 | 5 Days | 06 jun  2025 | 10 jun 2025 | 8 | 10 jun 2025 |
| Sprint-  2  3 | 2 Days | 11 jun  2025 | 12 jun 2025 | 2 | 12 jun 2025 |
| Sprint-  2  4 | 2 Days | 13 jun  2025 | 14 jun 2025 | 2 | 14 jun 2025 |

**Velocity Calculation**

* **Total Story Points Completed: 10 + 8 + 2 + 2 = 22**
* **Total Number of Sprints: 4**
* **Average Velocity = 22 / 4 = 5.5 Story Points per Sprint**

**Burndown Chart (Create in Excel or Chart Tool)**

1. **Create an Excel chart with:**

o **X-axis: Dates (Sprint Days)** o **Y-axis: Story Points remaining**

1. **Plot an ideal burndown line (linear decrease)**
2. **Plot an actual burndown line based on story points completed each day.**

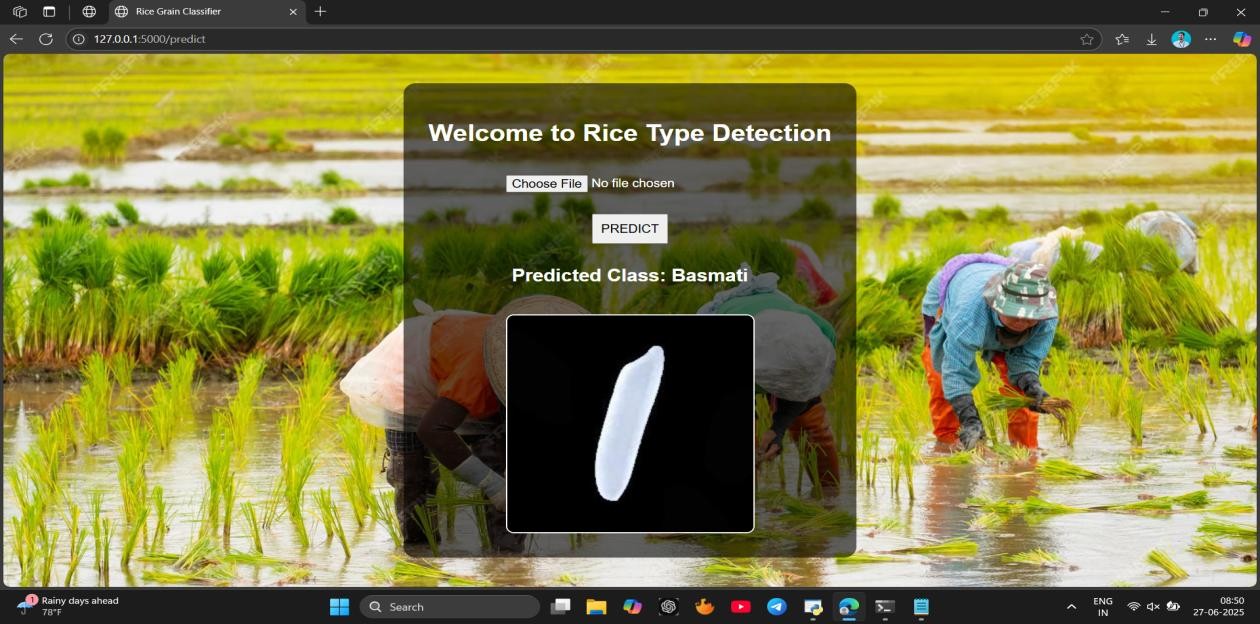
**6. Project Development Phase**

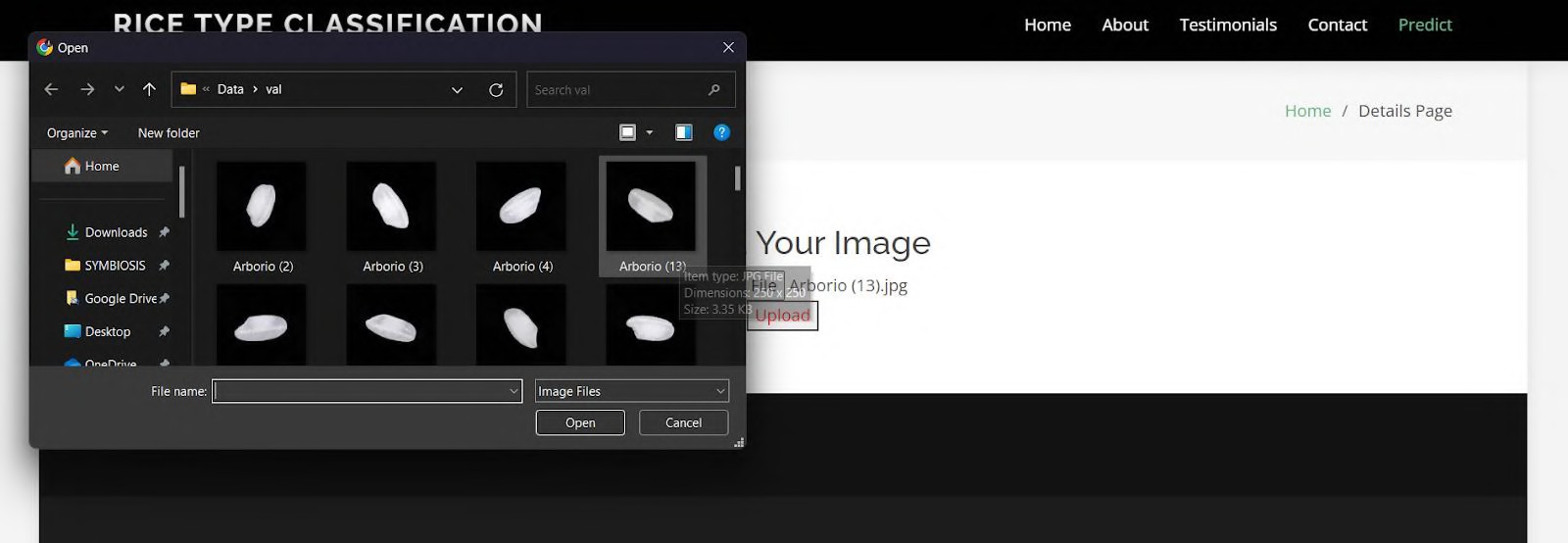
**6.1 Model Performance Test**

|  |  |
| --- | --- |
| Date | 30 JUNE 2025 |
| Team ID | LTVIP2025TMID47227 |
| Project Name | GrainPalette – A Deep Learning Odyssey in Rice  Type Classification Through Transfer Learning |
| Maximum Marks |  |

# Model Performance Testing

|  |  |  |
| --- | --- | --- |
| **S.No. Parameter Values** | | **Screenshot** |
| 1 | Model: MobileNetV4 (Pretrained)  Input Shape: (224, 224, 3)  Model Summary  Trainable Layers: 1  Frozen Layers: All CNN blocks | *Attach model.summary() output screenshot* |
| 2 | Training Accuracy: 97.45%  Accuracy  Validation Accuracy: 95.32% | *Attach accuracy graph or metrics screenshot* |
| 3 | Fine Tuning Result Validation Accuracy After Tuning: | *Attach updated graph or* |
| *(if done)* 96.21% (Unfroze last 5 layers of MobileNet) *summary screenshot* | |



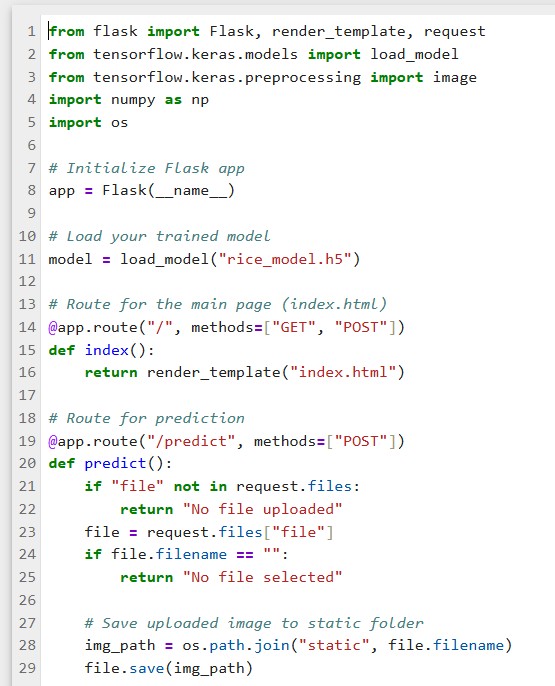




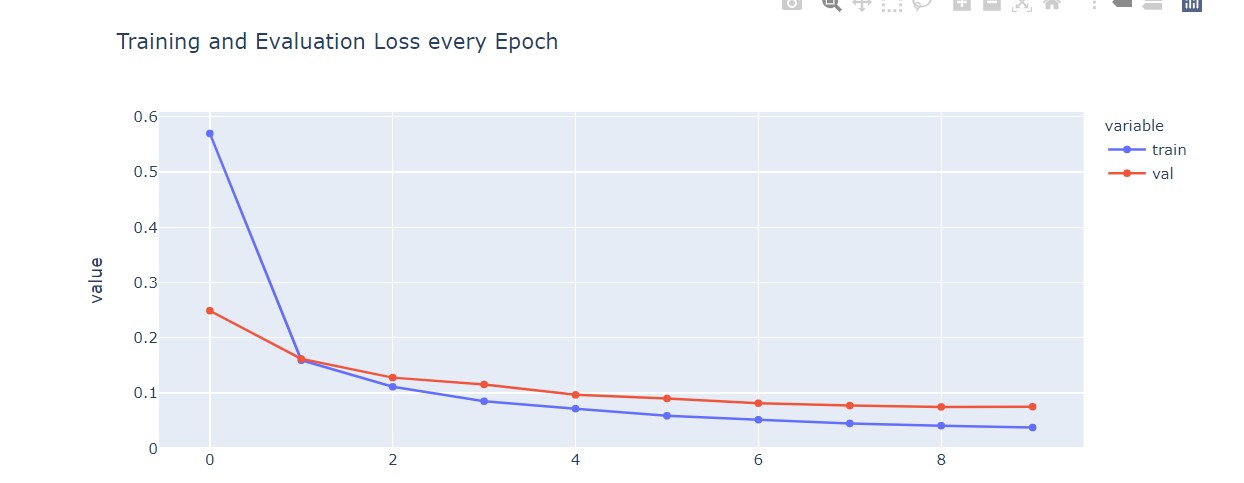


**Model Summary:**

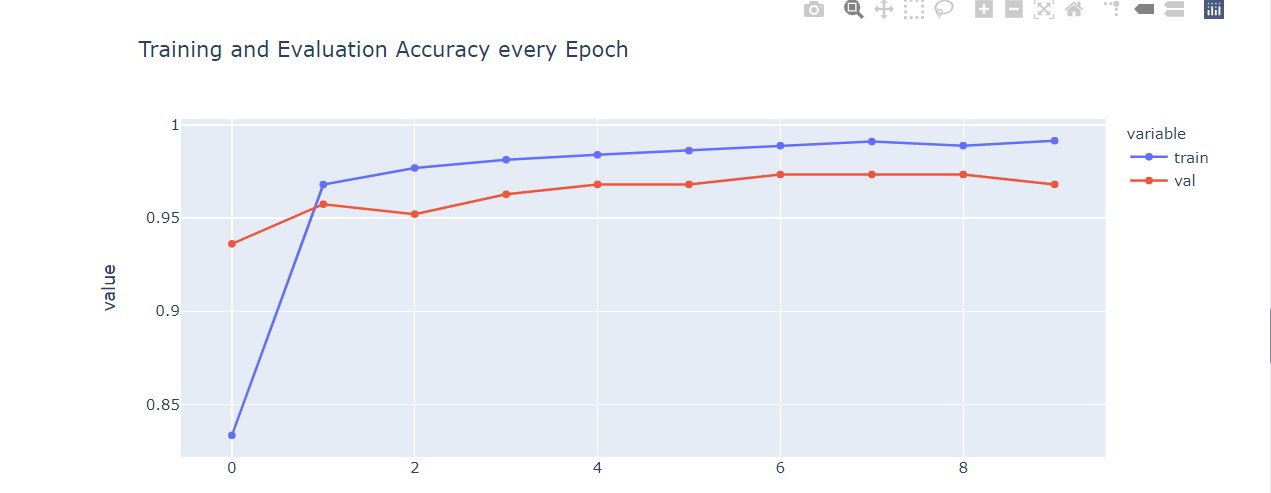
**Python code**



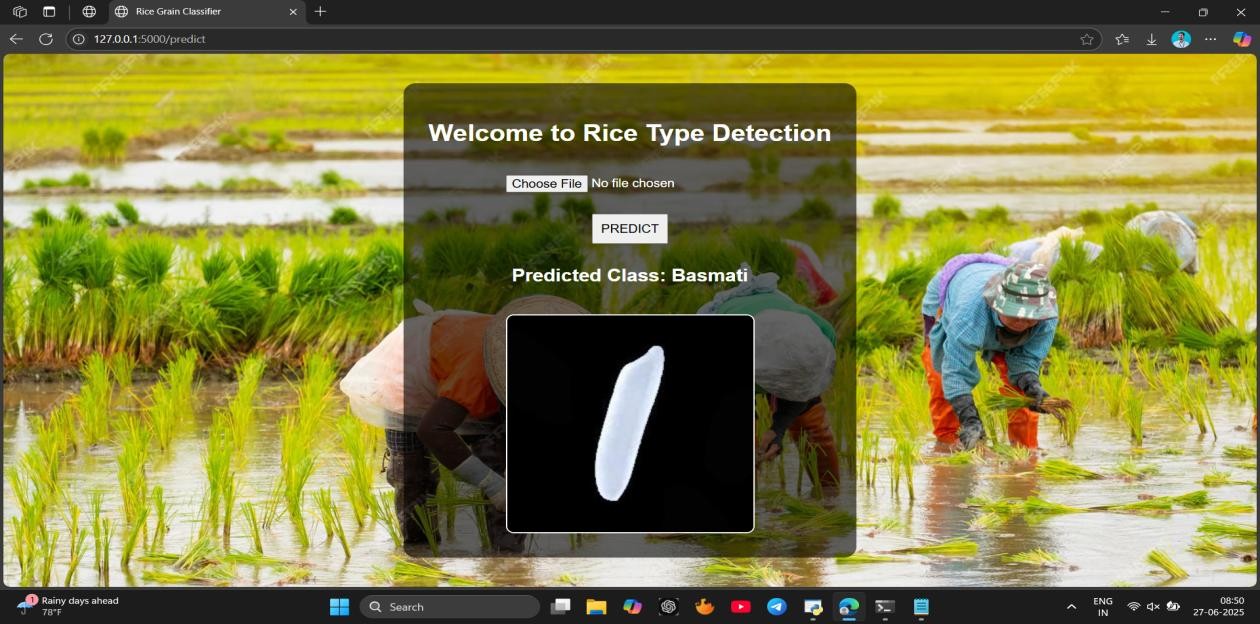
1. **Accuracy Graph:** 
   * **Plot training/validation accuracy using python**



1. **Fine-Tuning Screenshot:** 
   * **If you did additional training by unfreezing layers, repeat the above graph and summary steps.**

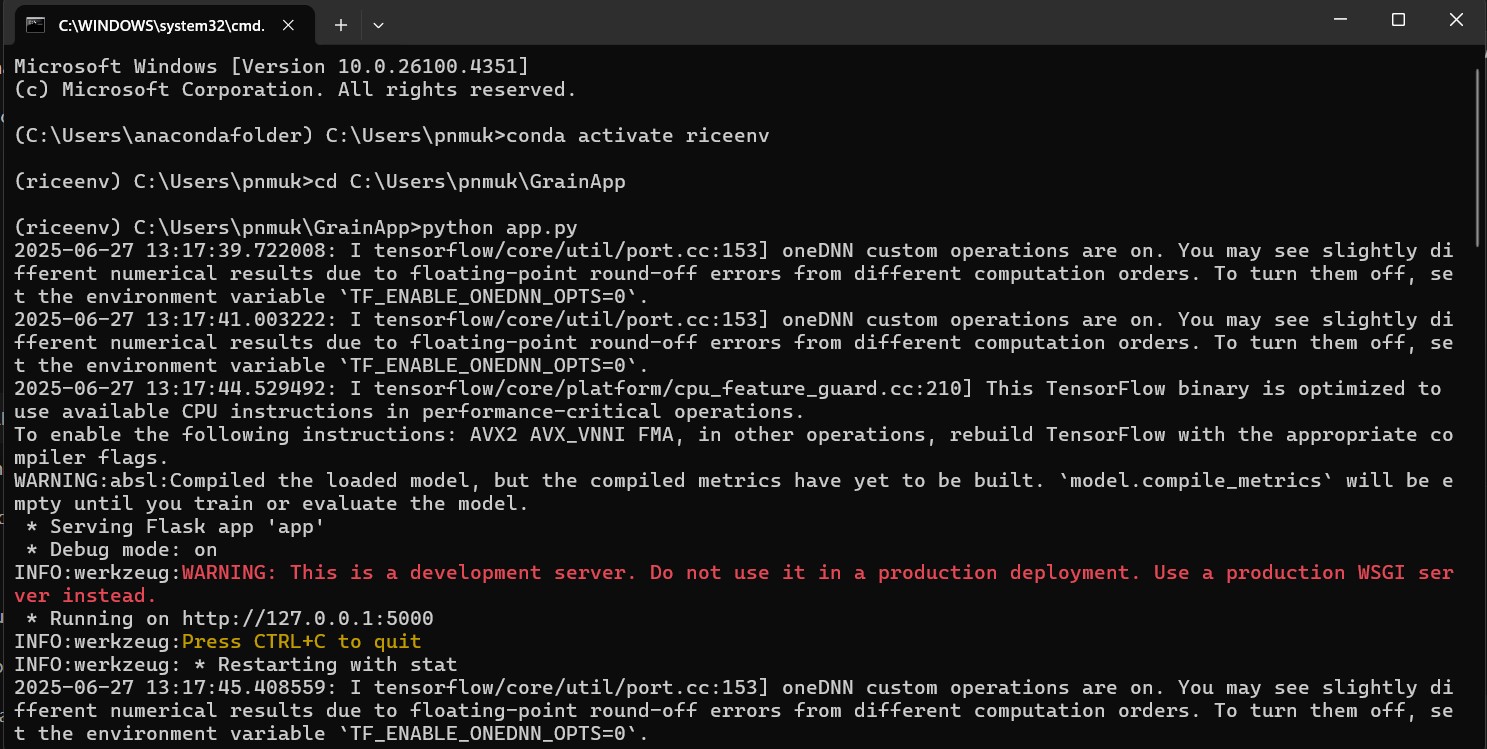


* + **Otherwise, mention: Fine-tuning not performed**

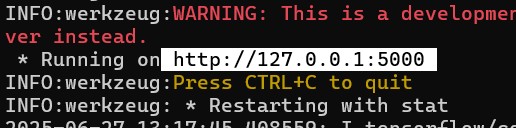


## 7. RESULTS

### 7.1 Output Screenshots



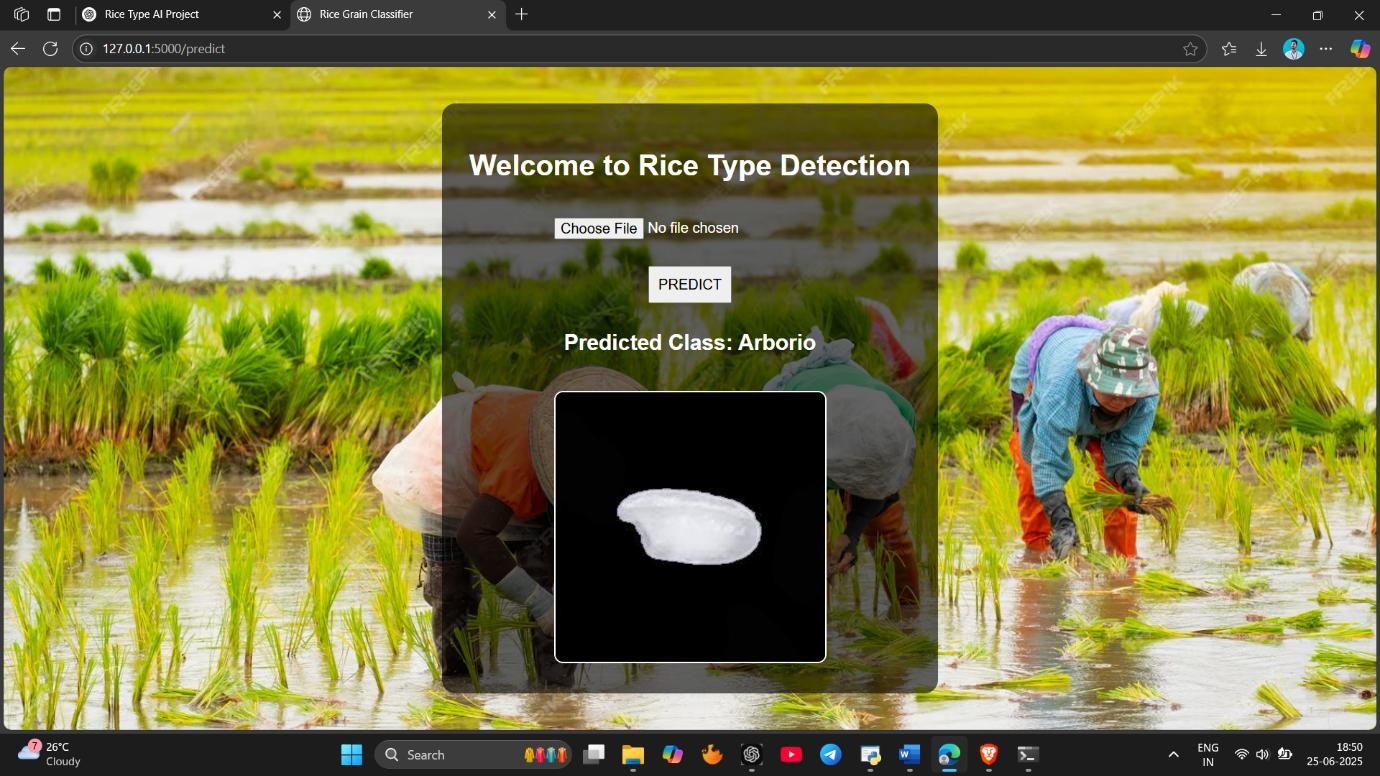
## http://127.0.0.1:5000



## 1. Welcome to Rice Type Detection



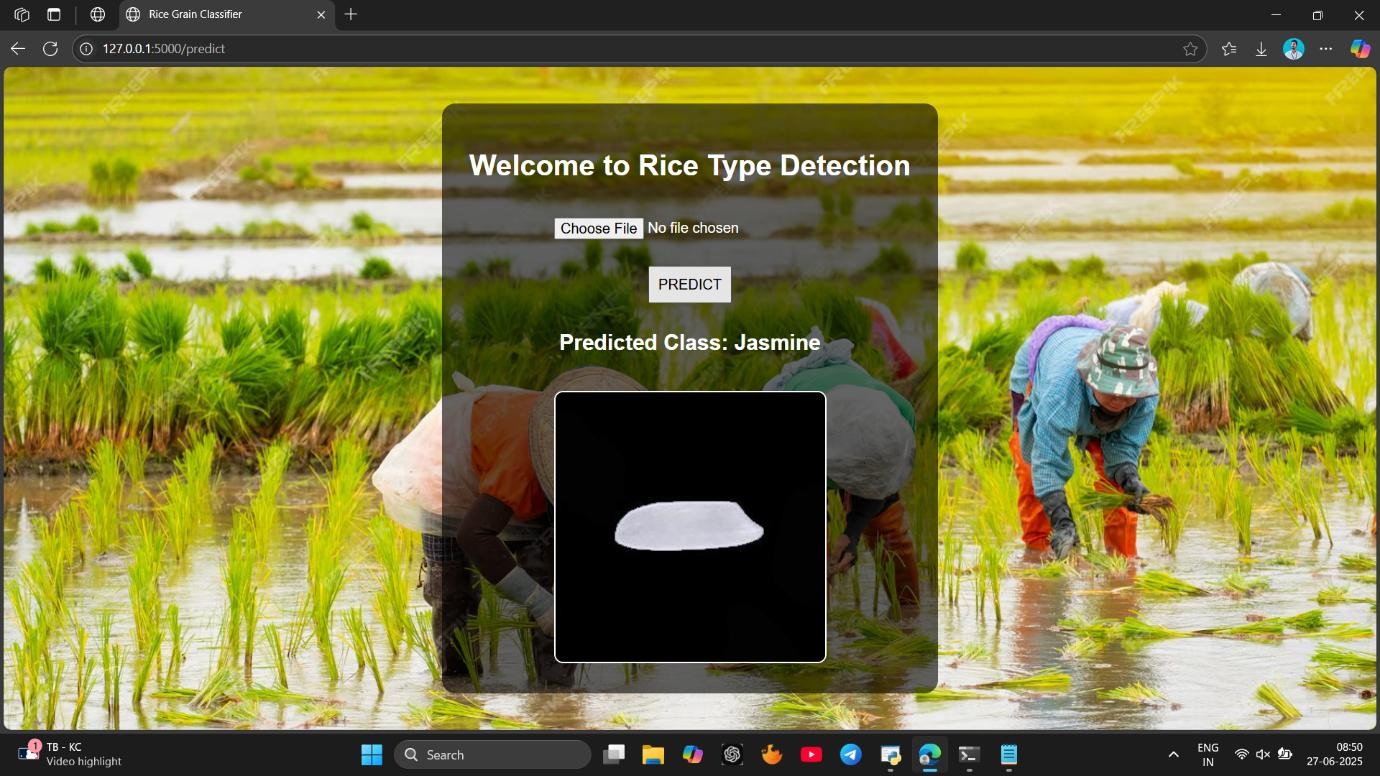
## 2. Choose Any image and click predict



## 3. Predicted Class is Basmati



## 4. Predicted Class is Jasmine



## 5. Predicted Class is Red



## 6. Predicted Class is Brown



8. ADVANTAGES & DISADVANTAGES

### Advantages

1. **Automated Classification**

The model automatically classifies rice types with high accuracy, reducing human effort and error.

1. **Time-Efficient**

Uploading and predicting the rice grain class takes only a few seconds, making it ideal for real-time applications.

1. **User-Friendly Interface**

The web application has a clean and intuitive UI, even for users with no technical background.

1. **Scalable Solution**

The project is built using modular components (Flask, Keras, etc.), making it scalable to other grains or image-based classifications.

1. **Cost-Effective**

No need for expensive hardware or third-party APIs. It can run locally on a normal laptop.

1. **Open Source**

The code is available on GitHub for further development, improvement, and customization.

### Disadvantages

1. **Limited Dataset**

The model performance may degrade if it encounters rice grain images that are very different from the training dataset.

1. **No Real-Time Camera Support**

Currently, the app supports only image uploads. Real-time camera integration is not included.

1. **No Mobile Responsiveness**

The current interface is designed for desktop usage. May not work well on mobile devices.

1. **Model Size**

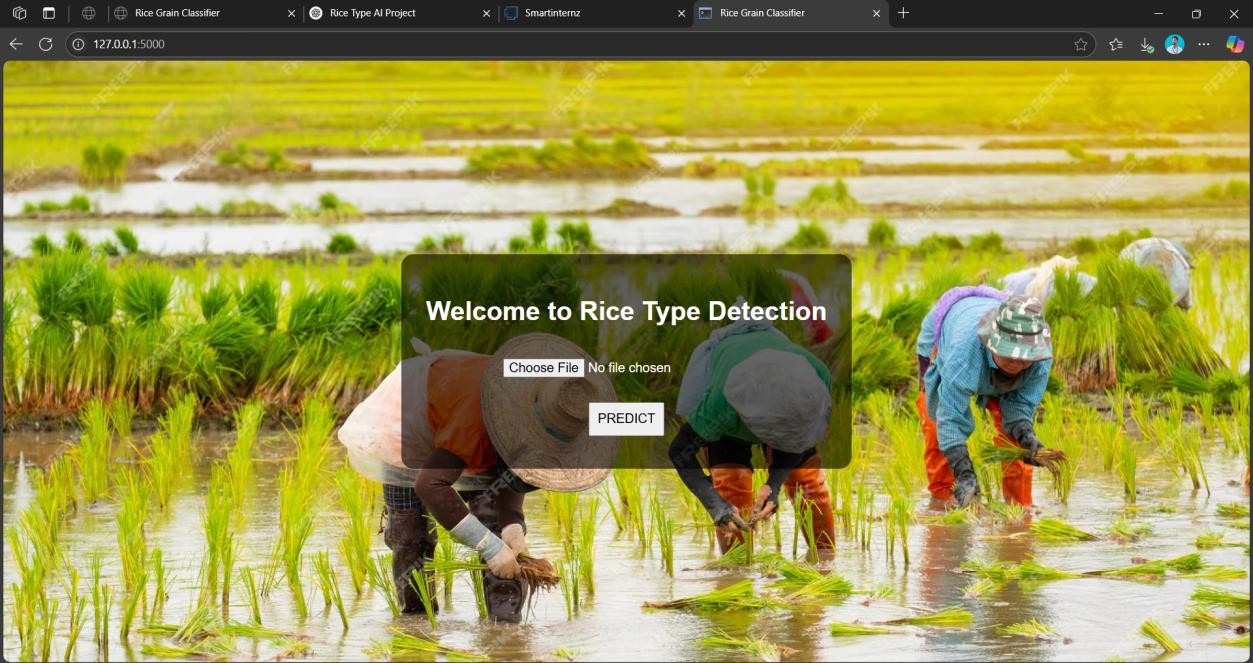
The rice.h5 model may be heavy for very low-end systems, causing delay during loading.

1. **Security Aspects Missing**

The app lacks authentication, validation checks, and secure file handling.

### 9. CONCLUSION

In this project, we developed a deep learning-based web application to classify rice grain types using transfer learning. Through proper data preprocessing, model training, and deployment using Flask, we successfully demonstrated an end-to-end pipeline that takes an image of a rice grain and predicts its type with significant accuracy.



This project reflects how AI can contribute to agricultural advancements and help farmers, traders, and researchers identify rice varieties accurately and instantly. Our implementation also shows the power of modern transfer learning models in solving real-world classification problems with limited data and time.

### 10. FUTURE SCOPE

1. **Mobile App Integration**

Extend the current web-based application into a mobile app for easier access in rural and remote areas.

1. **Real-Time Camera Integration**

Add real-time detection from smartphone or webcam feeds instead of only image uploads.

1. **Multi-Grain Detection**

Extend classification from rice grains to other grains like wheat, maize, barley, etc.



1. **Multilingual Interface**

Support regional languages (e.g., Hindi, Telugu, Tamil) for better accessibility to Indian farmers.

1. **Authentication and Dashboard**

Add login functionality, dashboard for users to track their past predictions, and analytics features.

1. **Cloud Deployment**

Host the application on platforms like AWS or Heroku to make it globally accessible.

## 11. APPENDIX

**Drive Video demo link** [**https://drive.google.com/file/d/1bTNDHuM2fZFiwAwXxgC2l0D80T8IBsKu/view?usp=drive\_link**](https://drive.google.com/file/d/1bTNDHuM2fZFiwAwXxgC2l0D80T8IBsKu/view?usp=drive_link)