**Model Performance Test**

|  |  |
| --- | --- |
| Date | 1 july 2025 |
| Team ID | - |
| Project Name | Project -cleantech |
| Maximum Marks |  |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Parameter** | **Values** | **Screenshot** |
|  | Model Summary | **-** |  |
|  | Accuracy | Training Accuracy -   Validation Accuracy - |  |
| 3. | Fine Tunning Result( if Done) | Validation Accuracy - |  |

Project Title: CleanTech: Transforming Waste Management with Transfer Learning  
  
Overview:  
This project aims to revolutionize waste management through advanced deep learning techniques. The core idea is to use transfer learning with a pre-trained VGG16 model to classify waste into three categories: biodegradable, recyclable, and trash. The model is integrated into a Flask-based web application that allows real-time prediction using uploaded waste images.  
  
Dataset Collection:  
The dataset used in this project includes three categories—biodegradable, recyclable, and trash. The images were sourced from Kaggle. You can access the dataset from the following link:  
Link: Dataset  
  
After downloading the dataset, unzip it and convert it into a structured format using pandas.  
  
Project Structure:  
Create the Project folder which contains files as shown below:  
- app.py (Flask script)  
- templates/ (HTML files)  
- static/ (image uploads, CSS, etc.)  
- Vgg16.h5 (pre-trained model file)  
  
Architecture:  
The architecture uses transfer learning. We have fine-tuned a VGG16 model on our dataset. Once trained, the model is saved as 'Vgg16.h5' for integration into a Flask web app.  
  
Activities  
  
Activity 1.1: Importing the Libraries  
Import all necessary libraries like TensorFlow, Keras, NumPy, Pandas, Matplotlib, OS, etc.  
  
Activity 1.2: Read the Dataset  
Use pandas to read image labels and organize image directories accordingly. Visualize sample images to understand the distribution.  
  
Data Visualization:  
Use random sampling to display example images from each class. The Python code uses IPython display, random, and os modules to show and verify correct classification of:  
- Biodegradable images  
- Recyclable images  
- Trash images  
  
Data Augmentation:  
To address the limited dataset size, data augmentation techniques such as rotation, flipping, zooming, and brightness adjustment were initially considered. However, in this implementation, the dataset already includes cropped and prepared images, so further augmentation was skipped.  
  
Impact:  
- Improved classification accuracy of waste types.  
- Real-time prediction using a web interface.  
- Supports smart cities in achieving better waste segregation.