**Model Performance Test**

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

## **Test Scenarios & Results**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Scenario (What to test)** | **Test Steps (How to test)** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| **FT-01** | Text Input Validation (e.g., topic, job title) | Enter valid and invalid text in input fields | Valid inputs accepted, errors for invalid inputs | Valid inputs accepted, errors for invalid inputs | p |
| **FT-02** | Number Input Validation (e.g., word count, size, rooms) | Enter numbers within and outside the valid range | Accepts valid values, shows error for out-of-range | Accepts valid values, shows error for out-of-range | p |
| **FT-03** | Content Generation (e.g., blog, resume, design idea) | Provide complete inputs and click "Generate" | Correct content is generated based on input | Correct content is generated based on input | p |
| **FT-04** | API Connection Check | Check if API key is correct and model responds | API responds successfully | API responds successfully | p |
| **PT-01** | Response Time Test | Use a timer to check content generation time | Should be under 3 seconds | Should be under 3 seconds | p |
| **PT-02** | API Speed Test | Send multiple API calls at the same time | API should not slow down | API should not slow down | p |
| **PT-03** | File Upload Load Test (e.g., PDFs) | Upload multiple PDFs and check processing | Should work smoothly without crashing | Should work smoothly without crashing | p |

Project Title: CleanTech: Transforming Waste Management with Transfer Learning  
  
Overview:  
This project uses deep learning with transfer learning techniques (VGG16) to classify waste images into three categories: biodegradable, recyclable, and trash. It is integrated with a Flask-based web application for real-time predictions.  
  
Collect the Dataset:  
There are many popular open sources for collecting the data, such as kaggle.com and UCI repository. In this project, we have used images of three waste types: biodegradable, recyclable, and trash, downloaded from Kaggle.  
  
Link: Dataset  
  
Once downloaded, unzip and load the data using pandas for exploration and visualization.  
  
Activity 1.1: Importing the Libraries:  
Essential libraries include TensorFlow, Keras, pandas, numpy, matplotlib, os, and random.  
  
Activity 1.2: Read the Dataset:  
Load the data into a pandas DataFrame. The dataset might be in formats like .csv, .txt, .json, or .zip.  
  
Data Visualization:  
Randomly display images from each class using IPython’s Image module to verify proper labeling and understand the data visually.  
  
Example Predictions:  
- Biodegradable: The model correctly predicted the image as biodegradable.  
- Recyclable: The model correctly predicted the image as recyclable.  
- Trash: The model correctly predicted the image as trash.  
  
Data Augmentation:  
Data augmentation (flipping, rotating, scaling, etc.) improves model robustness. Although in this case, augmentation was skipped as the dataset was pre-processed, the training time increased slightly, but accuracy was not compromised.  
  
Project Structure:  
- app.py (Flask backend)  
- templates/ (HTML files)  
- static/ (images, styles)  
- Vgg16.h5 (saved model)  
  
Architecture:  
The application uses a fine-tuned VGG16 model saved in H5 format. This model is loaded in Flask for real-time predictions via a web interface.  
  
Conclusion:  
The CleanTech project offers an innovative way to assist in smart waste management using AI. It classifies waste efficiently and integrates seamlessly with a user-friendly web app.