Project Report

# CleanTech: Transforming Waste Management with Transfer Learning

Team Name: Trash Transformers

Team Members: Tejasree, Srikanth, Chandana, Jaswanth

## Phase 1: Brainstorming & Ideation

### Objective

To develop an AI-powered waste classification system using transfer learning that improves sorting accuracy, reduces manual effort in solid waste management, and supports applications such as smart recycling, automated segregation, and environmentally sustainable waste handling.

### Problem Statement

Municipal solid waste classification is still heavily reliant on manual processes, which are time-consuming and error-prone. This leads to inefficient recycling and increased landfill usage.  
  
CleanTech utilizes transfer learning to classify waste types from images accurately. By integrating AI into the waste management system, it enhances classification speed, precision, and supports smarter environmental decision-making.

### Proposed Solution

An AI-powered application that uses transfer learning to accurately classify waste images in real-time into categories like biodegradable, recyclable, and trash. It helps municipalities and citizens by providing quick, reliable waste segregation support.

### Target Users

- Municipal waste departments and sanitation workers who need fast and accurate waste sorting.  
- Recycling centers aiming to automate the waste segregation process.  
- Citizens who want to contribute to smart waste disposal through user-friendly applications.  
- Educational institutions promoting eco-awareness and sustainability practices.

### Expected Outcome

- A functional AI-powered application that accurately classifies waste images using transfer learning.  
- Helps reduce manual sorting errors, improves recycling rates, and supports smart city waste management systems.

## Phase 2: Requirement Analysis

### Objective

Define the technical and functional requirements for the CleanTech waste classification system.

### Technical Requirements

- Programming Language: Python  
- Python Packages: NumPy, Pandas, Scikit-learn, Matplotlib, SciPy, Seaborn, TensorFlow, Flask  
- Frameworks: Flask for web integration, TensorFlow for deep learning  
- Pre-trained Model: VGG16 (used for transfer learning)  
- Development Tools: Command Line (pip install)

### Functional Requirements

- Ability to upload waste images through a web interface.  
- Classify waste into categories like biodegradable, recyclable, and trash using a trained model.  
- Display classification results along with prediction confidence.  
- Provide an easy-to-use interface for municipal staff, students, and the general public.

### Constraints & Challenges

- Handling imbalanced datasets for different waste categories.  
- Managing blurry or unclear waste images that may affect accuracy.  
- Optimizing model performance for quick prediction and smooth deployment.  
- Ensuring the interface works well on both desktop and mobile devices.

## ****Phase 3: Project Design****

### ****Objective****

Develop the architecture and user flow of the CleanTech waste classification system.

### ****System Architecture****

* The user uploads a waste image through the web interface (UI).
* The image is passed to the trained deep learning model.
* The model (based on a pre-trained CNN like VGG16) predicts the type of waste.
* The prediction with confidence score is shown in the UI.
* Initially, the model is trained on preprocessed images split into training/testing sets.
* The training set fine-tunes the transfer learning model; testing set is used for evaluation.

### ****User Flow****

1. User opens the web app
2. Uploads an image of waste
3. The model processes the image and predicts the waste class
4. The predicted class and confidence score are displayed
5. User can repeat with more images or use results to guide disposal

## ****Phase 4: Project Planning****

### ****Objective****

Break down development tasks for efficient and timely completion.

### ****Sprint Planning****

#### ****Sprint 1 – Setup & Preparation (Day 1)**** 🔴 High Priority

* Set up the environment using Anaconda Navigator
* Install required packages (TensorFlow, Flask, etc.)
* Collect and preprocess the waste image dataset

#### ****Sprint 2 – Model Development & Integration (Day 2)**** 🔴 High Priority

* Build and train the waste classification model using transfer learning (e.g., VGG16)
* Integrate the model into the Flask or Streamlit web interface

#### ****Sprint 3 – Testing, Deployment & Presentation (Day 2)**** 🟡 Medium / 🟢 Low

* Test app functionality, fix bugs, improve UI responsiveness
* Finalize deployment and prepare demo/presentation materials

## ****Phase 5: Project Development****

### ****Objective****

Implement the core features of CleanTech using transfer learning for waste classification.

### ****Technology Stack****

* **Frontend**: HTML (via Flask templates) or Streamlit
* **Backend**: Flask Framework
* **Deep Learning**: TensorFlow with VGG16
* **Language**: Python

### ****Development Process****

* Built and trained a waste classification model using VGG16
* Preprocessed a dataset of labeled waste images
* Integrated the trained model into a web interface
* Created a UI for uploading images and displaying prediction results

### ****Challenges & Fixes****

* **Overfitting on specific classes** → Solved with data augmentation and dropout
* **Slow inference due to model size** → Optimized and saved in .h5 format
* **Image quality variation** → Applied resizing and normalization during preprocessing