### ****CleanTech — Technology Stack and Architecture****

**Date**:1 july 2025  
**Team ID**: LTVIP2025TMID38158

**Project Name**: CleanTech  
**Maximum Marks**: 4 Marks

### ****Technical Architecture****

CleanTech is an AI-powered waste classification application that uses transfer learning with the VGG16 model. The system classifies images into **biodegradable**, **recyclable**, or **trash** using a Flask-based web interface.

#### Architecture Elements:

**Frontend**: HTML/CSS for uploading and displaying predictions

**Backend**: Flask server in Python handles model logic

**Model**: Pre-trained VGG16 using Keras/TensorFlow

**Storage**: Local file storage for uploaded images

**Deployment**: Runs on localhost (can be containerized for cloud)

### 📋 ****Table-1: Components & Technologies****

| **Component** | **Description** | **Technology** |
| --- | --- | --- |
| User Interface | Image upload & result display | HTML, CSS, JavaScript |
| Application Logic-1 | Upload & handle input | Python, Flask |
| Application Logic-2 | Predict using VGG16 model | TensorFlow, Keras |
| Database | Not applicable | — |
| Cloud Database | Not used | — |
| File Storage | Store uploaded files | Local Filesystem |
| External API-1 | Not used | — |
| External API-2 | Not used | — |
| Machine Learning Model | Waste classification | VGG16 (Transfer Learning) |
| Infrastructure | Hosted locally via Flask | Localhost, Python environment |

### 📋 ****Table-2: Application Characteristics****

| **Characteristic** | **Description** | **Technology** |
| --- | --- | --- |
| Open-Source Frameworks | Flask, Pandas, TensorFlow, Keras | Python Libraries |
| Security Implementations | File type/size validation, basic auth possible | Flask Middleware |
| Scalable Architecture | Container-ready with Docker or cloud migration | Docker (Optional) |
| Availability | Deployable on cloud with load balancing | AWS EC2 / GCP |
| Performance | Optimized for small workloads | No CDN, Local Cache |