# Sustainable Smart City Assistant

DATE

TEAM ID  
NM2025TMID00605

PROJECT NAME  
SUSTAINABLE SMART CITY ASSISTANT USING IBM GRANITE LLM

MAXIMUM MARKS

## Introduction

The Sustainable Smart City Assistant is an AI-powered solution that provides actionable eco-friendly living tips and summarizes complex policy documents to enhance community engagement and promote sustainable practices. Leveraging IBM Granite LLM for advanced natural language processing, the assistant simplifies sustainability efforts by helping users make informed decisions on environmental conservation and policy interpretation.

## Key Features

* Eco Tips Generator: Provides actionable, context-aware eco-friendly tips based on user-provided keywords.
* Policy Summarization: Summarizes uploaded PDF policy documents or pasted text, highlighting essential points.
* PDF Parsing: Extracts and processes content from PDF documents for easy summarization.
* Gradio Interface: Offers a simple, interactive interface for eco tips and policy analysis.

## Disclaimer

The Smart City Assistant is an informational tool designed to promote awareness and sustainability efforts. It should not replace professional expertise or official policy documentation.

## Getting Started

### Prerequisites

Python 3.x (recommended version: 3.8 or higher)  
PyTorch (with CUDA support for GPU acceleration, optional)  
Gradio (for building the web interface)  
Transformers library (for AI model loading)  
PyPDF2 (for extracting text from PDFs)

### Installation Steps

Install dependencies using:  
  
pip install torch gradio transformers PyPDF2

## How It Works

### Model Loading

The assistant uses IBM’s Granite language model for generating context-aware responses.  
  
model\_name = "ibm-granite/granite-3.2-2b-instruct"  
tokenizer = AutoTokenizer.from\_pretrained(model\_name)  
model = AutoModelForCausalLM.from\_pretrained(model\_name, ... )

### Response Generation

The generate\_response() function processes user prompts to generate AI responses. The output is controlled by parameters like max\_length and temperature for balanced creativity and precision.

### Eco Tips Generator

The eco\_tips\_generator() function provides actionable sustainability recommendations based on user-provided keywords.  
  
Input: Keywords (e.g., 'plastic waste', 'solar energy')  
Output: Eco-friendly tips and practices.

### Policy Summarization

The policy\_summarization() function summarizes uploaded PDF policy documents or pasted text, offering key points, provisions, and implications.

### PDF Extraction

The extract\_text\_from\_pdf() function uses PyPDF2 to extract and process text from uploaded PDF files.

## Gradio Interface

The interface consists of two tabs:  
1. Eco Tips Generator: Users provide keywords to receive actionable sustainability suggestions.  
2. Policy Summarization: Users upload PDFs or paste policy text for concise summaries.

## Conclusion

The Sustainable Smart City Assistant simplifies sustainability planning and enhances policy comprehension. It empowers communities, organizations, and individuals to embrace greener practices and understand policies better.