• Sustainable Smart City Assistant

Project Documentation

Team Members

Anandhi E — Team Leader

Lingaisha M— Team Member

Anusiya E — Team Member

Arthi D — Team Member

Yamini D — Team Member

Academic Year: 2025

Department of Computer Science & Engineering

College Project Report

1. Introduction

The concept of a Smart City has evolved as a response to rapid urbanization, growing populations, and the urgent need for sustainable development. Modem cities face challenges such as resource scarcity, environmental degradation, traffic congestion, and waste management. To address these issues, govemments and researchers are turning to Artificial Intelligence (Al), Internet of Things (IOT), and data analytics for innovative solutions. The Sustainable Smart City Assistant is designed as a platform that empowers both citizens and policymakers. Citizens benefit from eco-friendly lifestyle guidance, while policymakers gain clear and simplified insights from policy documents. This creates a bridge between governance and community, ensuring that sustainability goals are met collaboratively.

2. Project Overview

Purpose:

The purpose of this assistant is to provide Al-powered support for sustainable development in cities. It achieves this by helping optimize essential resources such as energy, water, and waste, while guiding citizens toward environmentally responsible behavior. For city officials, it acts as a decision-making partner by summarizing complex policy documents, forecasting future trends, and detecting anomalies in urban data. Ultimately, this assistant contributes to greener, smarter, and more resilient cities.

Features:

- Conversational Interface: Provides natural language interaction for citizens and policymakers.
- Policy Summarization: Converts lengthy government documents into concise, actionable summaries.
- Resource Forecasting: Estimates future usage of energy, water, and waste.

- Eco-Tip Generator: Delivers personalized daily suggestions for sustainable living.
- Citizen Feedback Loop: Engages the community by collecting input for planning.
- KPI Forecasting: Supports officials with projections of key performance indicators.
- Anomaly Detection: Alerts stakeholders about unusual trends in city data.
- Multimodal Input: Accepts text, PDFs, and CSVs for broader usability.
- User-Friendly Interface: Gradio-based design with intuitive navigation.

3. System Architecture

The architecture of the Sustainable Smart City Assistant consists of multiple layers working together to ensure smooth functionality:

- Frontend: Built with Gradio for interactive UI, providing tabs for Eco Tips, Policy Summarization, and Policy Q&A.;
- Backend: Powered by FastAPI, enabling fast and scalable communication between modules.
- LLM Integration: IBM Granite LLM handles natural language processing for summaries and Q&A.;
- Data Handling: PDFs and CSVs are processed for extracting text and forecasting data.
- Forecasting Modules: Implemented using Scikit-learn for predictive analytics on resources.
- Semantic Search: (Optional) Pinecone is used for vector search on uploaded documents.

4. Setup Instructions

To run the Sustainable Smart City Assistant, the following prerequisites and steps are required:

1. Python 3.9 or higher must be installed.
2. Clone the project repository from the source control system.
3. Install required dependencies from requirements.txt.
4. Configure environment variables including API keys in a .env file.

5 Start the FastAPI backend server.

6 Launch the Gradio-based web UI.

7 Upload policy documents or CSV datasets and start interacting with the system.

5. Folder Structure

The project is organized into modular components for scalability:
app/ — Contains FastAPI backend logic.
applapi/ — API routes for chat, feedback, and documents.

- ui/ — Gradio UI components for web interface.
- granite_llm.py — Handles IBM Granite LLM interactions.
- document_embedder.py — Converts policy documents into embeddings.
  kpi_file_forecaster.py — Forecasts energy and water usage.
- anomaly_file_checker.py — Detects anomalies in KPI datasets.
- report_generator.py — Builds AI-generated sustainability reports.

6. Running the Application

Once installed, the application can be run by launching the backend and UI components. The assistant provides the following workflow:

- Launch FastAPI backend and Gradio frontend.
- Navigate through the tabs: Eco Tips, Policy Summarization, and Q&A.;
- Upload PDFs or enter text input for processing.
- Receive eco-friendly tips, summarized policies, and direct answers to queries.
- Download reports in text or PDF format for further use.

7. API Documentation

The backend provides several REST APIs for interaction:

- POST /chat/ask — Processes user queries and retums AI-generated answers.
- POST [upload-doc — Uploads and embeds documents.
- GET [search-docs — Retrieves semantically similar policy documents.
- GET /get-eco-tips — Provides sustainability tips on selected topics.
- POST /submit-feedback — Collects and stores citizen feedback.

8. Authentication

The demonstration version runs in an open environment, but secure deployments will include the following:

- Token-based authentication using JWT.
- OAuth2 with IBM Cloud credentials.
- Role-based access control (citizen, admin, researcher). • Session tracking for personalized experiences.

9. Testing

Testing ensures the reliability of the system across multiple scenarios:

- Unit Testing for utility functions and prompt handling.
- API Testing via Swagger UI and Postman.
- Manual Testing of file uploads and interactive responses.
- Edge Case Handling for malformed inputs and large documents.

10. Conclusion & Future Enhancements

The Sustainable Smart City Assistant is a step toward building eco-conscious urban environments. It not only supports citizens with lifestyle guidance but also empowers policymakers with tools to analyze, summarize, and forecast urban policies. Future enhancements include:

- Integration with IOT devices for real-time monitoring.
- Mobile application development.
- Support for multiple languages.
- Advanced predictive analytics for long-term planning.

This project demonstrates how Al-driven platforms can foster community engagement, efficient resource usage, and sustainable urban development.