

Sustainable Smart City Assistant

Project Documentation

1.Introduction

- Project title : **Sustainable Smart City Assistant** using **IBM Granite LLM**
- Team member : Mohammed Dilshad
- Team member : Parthaboina Sindhu
- Team member : Pattapu Chandrika Padmavathi
- Team member : Pilli Meri Gnana Kusumanjali

2.project overview

- Purpose :

The purpose of a Sustainable Smart City Assistant is to empower cities and their residents to thrive in a more eco-conscious and connected urban environment. By leveraging AI and real-time data, the assistant helps optimize essential resources like energy, water, and waste, while also guiding sustainable behaviors among citizens through personalized tips and services. For city officials, it serves as a decision-making partner—offering clear insights, forecasting tools, and summarizations of complex policies to support strategic planning. Ultimately, this assistant bridges technology, governance, and community engagement to foster greener cities that are more efficient, inclusive, and resilient.

- Features:

Conversational Interface

Key Point: Natural language interaction

Functionality: Allows citizens and officials to ask questions, get updates, and receive guidance in plain language

Policy Summarization

Key Point: Simplified policy understanding

Functionality: Converts lengthy government documents into concise, actionable summaries.

Resource Forecasting

Key Point: Predictive analytics

Functionality: Estimates future energy, water, and waste usage using historical and real-time data.

Eco-Tip Generator

Key Point: Personalized sustainability advice

Functionality: Recommends daily actions to reduce environmental impact based on user behavior.

Citizen Feedback Loop

Key Point: Community engagement

Functionality: Collects and analyzes public input to inform city planning and service improvements.

KPI Forecasting

Key Point: Strategic planning support

Functionality: Projects key performance indicators to help officials track progress and plan ahead.

Anomaly Detection

Key Point: Early warning system

Functionality: Identifies unusual patterns in sensor or usage data to flag potential issues.

Multimodal Input Support

Key Point: Flexible data handling

Functionality: Accepts text, PDFs, and CSVs for document analysis and forecasting.

Streamlit or Gradio UI

Key Point: User-friendly interface

Functionality: Provides an intuitive dashboard for both citizens and city officials to interact with the assistant.

3. Architecture

Frontend (Stream lit):

The frontend is built with Streamlit, offering an interactive web UI with multiple pages including dashboards, file uploads, chat interface, feedback forms, and report viewers. Navigation is handled through a sidebar using the streamlit-option-menu library. Each page is modularized for scalability.

Backend (Fast API):

Fast API serves as the backend REST framework that powers API endpoints for document processing, chat interactions, eco tip generation, report creation, and vector embedding. It is optimized for asynchronous performance and easy Swagger integration.

LLM Integration (IBM Watsonx Granite):

Granite LLM models from IBM Watsonx are used for natural language understanding and generation. Prompts are carefully designed to generate summaries, sustainability tips, and reports.

Vector Search (Pinecone):

Uploaded policy documents are embedded using Sentence Transformers and stored in Pinecone. Semantic search is implemented using cosine similarity to allow users to search documents using natural language queries.

ML Modules (Forecasting and Anomaly Detection):

Lightweight ML models are used for forecasting and anomaly detection using Scikit-learn. Time-series data is parsed, modeled, and visualized using pandas and matplotlib.

4. Setup Instructions

Prerequisites:

- Python 3.9 or later
- pip and virtual environment tools
- API keys for IBM Watsonx and Pinecone
- Internet access to access cloud services

Installation Process:

- Clone the repository
- Install dependencies from requirements.txt
- Create a .env file and configure credentials
- Run the backend server using Fast API
- Launch the frontend via Stream lit
- Upload data and interact with the modules

5. Folder Structure

app/ – Contains all Fast API backend logic including routers, models, and integration modules.

app/api/ – Subdirectory for modular API routes like chat, feedback, report, and document vectorization.

ui/ – Contains frontend components for Stream lit pages, card layouts, and form UIs.

smart_dashboard.py – Entry script for launching the main Stream lit dashboard.

granite_llm.py – Handles all communication with IBM Watsonx Granite model including summarization and chat.

document_embedder.py – Converts documents to embeddings and stores in Pinecone.

kpi_file_forecaster.py – Forecasts future energy/water trends using regression.

anomaly_file_checker.py – Flags unusual values in uploaded KPI data.

report_generator.py – Constructs AI-generated sustainability reports.

6. Running the Application

To start the project:

- Launch the FastAPI server to expose backend endpoints.
- Run the Streamlit dashboard to access the web interface.
- Navigate through pages via the sidebar.
- Upload documents or CSVs, interact with the chat assistant, and view outputs like reports, summaries, and predictions.

- All interactions are real-time and use backend APIs to dynamically update the frontend.

Frontend (Stream lit):

The frontend is built with Stream lit, offering an interactive web UI with multiple pages including dashboards, file uploads, chat interface, feedback forms, and report viewers. Navigation is handled through a sidebar using the stream lit-option-menu library. Each page is modularized for scalability.

Backend (Fast API):

Fast API serves as the backend REST framework that powers API endpoints for document processing, chat interactions, eco tip generation, report creation, and vector embedding. It is optimized for asynchronous performance and easy Swagger integration.

7. API Documentation

Backend APIs available include:

POST /chat/ask – Accepts a user query and responds with an AI-generated message

POST /upload-doc – Uploads and embeds documents in Pinecone

GET /search-docs – Returns semantically similar policies to the input query

GET /get-eco-tips – Provides sustainability tips for selected topics like energy, water, or waste

POST /submit-feedback – Stores citizen feedback for later review or analytics

Each endpoint is tested and documented in Swagger UI for quick inspection and trial during development.

8. Authentication

each endpoint is tested and documented in Swagger UI for quick inspection and trial during development.

This version of the project runs in an open environment for demonstration. However, secure deployments can integrate:

- Token-based authentication (JWT or API keys)
 - OAuth2 with IBM Cloud credentials
 - Role-based access (admin, citizen, researcher)
 - Planned enhancements include user sessions and history tracking.⁸
- Authentication

9. User Interface

The interface is minimalist and functional, focusing on accessibility for non-technical users. It includes:

Sidebar with navigation

KPI visualizations with summary cards

Tabbed layouts for chat, eco tips, and forecasting

Real-time form handling

PDF report download capability

The design prioritizes clarity, speed, and user guidance with help texts and intuitive flows.

10. Testing

Testing was done in multiple phases:

Unit Testing: For prompt engineering functions and utility scripts

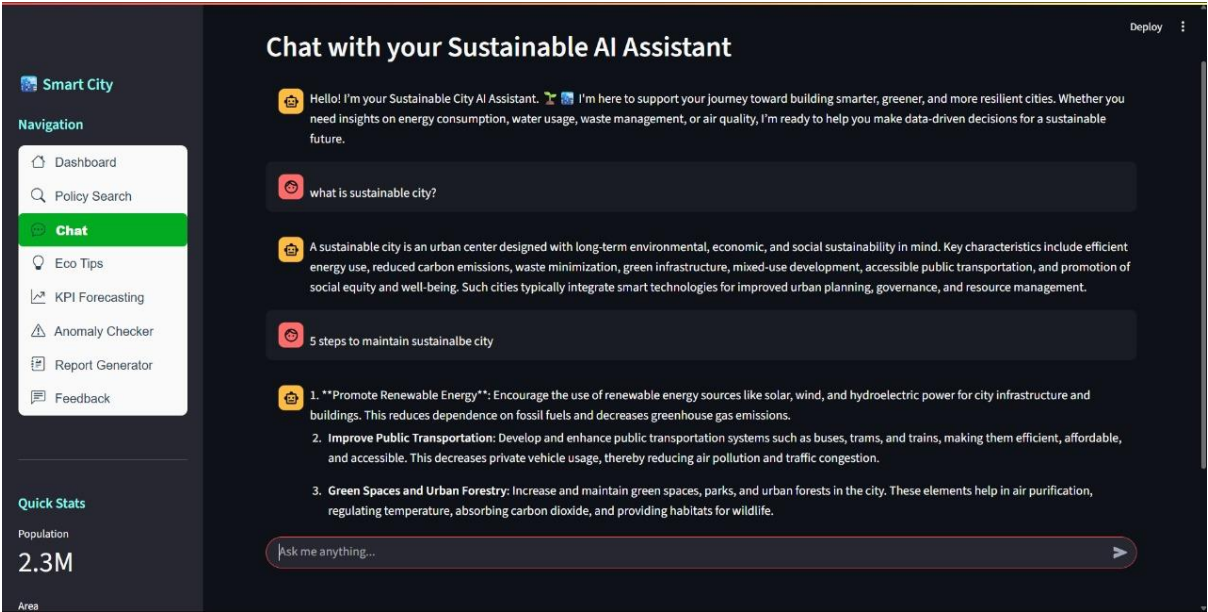
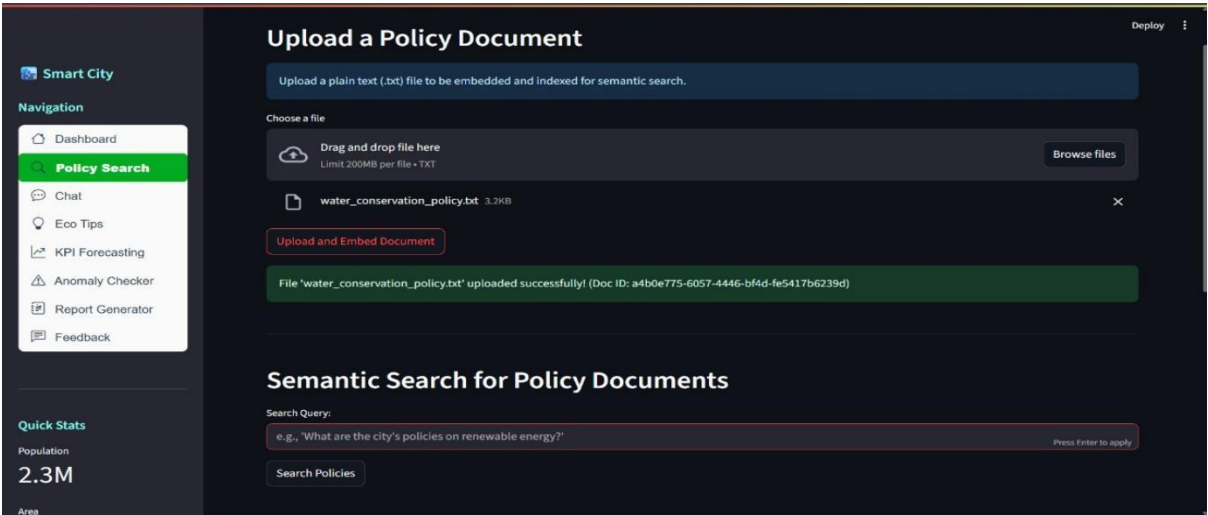
API Testing: Via Swagger UI, Postman, and test scripts

Manual Testing: For file uploads, chat responses, and output consistency

Edge Case Handling: Malformed inputs, large files, invalid API keys

Each function was validated to ensure reliability in both offline and API-connected modes.

11.screen shots



Smart City

Navigation

- Dashboard
- Policy Search
- Chat
- Eco Tips
- KPI Forecasting
- Anomaly Checker
- Report Generator
- Feedback

Quick Stats

Population

2.3M

Area

Resources

5. **Smart Building and Energy-Efficient Technologies:** Encourage the construction and renovation of buildings to be energy-efficient. Smart building technology, like automated lighting and HVAC systems, can optimize resource consumption based on occupancy and environmental conditions, further reducing a city's ecological footprint.

These steps, when consistently applied and adapted to a city's unique characteristics and resources, can help maintain a sustainable urban environment.

what is covid 19?

I'm here to help with questions about sustainable cities, smart city governance, and urban living. Please ask something related to these topics!

suggest me some tips to maintain good health

I'm here to help with questions about sustainable cities, smart city governance, and urban living. Please ask something related to these topics!

give me some educational tips.

I'm here to help with questions about sustainable cities, smart city governance, and urban living. Please ask something related to these topics!

ask me anything...

Smart City

Navigation

- Dashboard
- Policy Search
- Chat
- Eco Tips
- KPI Forecasting
- Anomaly Checker
- Report Generator
- Feedback

Quick Stats

Population

2.3M

Area

Get Personalized Eco-Friendly Tips

Enter a topic:

water

Get Eco Tip

Eco Tip for 'Water':

1. Implement Rainwater Harvesting: Encourage buildings, both residential and commercial, to install rainwater harvesting systems. This captured rainwater can be used for non-potable purposes like irrigation, toilet flushing, and even car washing. City-wide initiatives, such as offering subsidies or tax benefits for installing such systems, can significantly increase participation.

2. Establish Decentralized Water Treatment and Reuse: Deepen community involvement in local water management by setting up small-scale, decentralized wastewater treatment and reuse systems. These could be implemented in neighborhoods or districts, treating wastewater for non-potable uses like irrigation of public spaces, street cleaning, or even cooling towers for industrial processes. This not only reduces strain on centralized infrastructure but also fosters local sustainability consciousness and resilience.

Smart City

Navigation

- Dashboard
- Policy Search
- Chat
- Eco Tips
- KPI Forecasting
- Anomaly Checker
- Report Generator
- Feedback

Quick Stats

Population

2.3M

Area

Analysis Results

Date	Value
Mon 25	1200
Fri 29	1300
2024	1400
Fri 05	1500
Tue 09	1600
Sat 13	1700
Wed 17	1800
Jan 21	1900
Thu 25	2000
Mon 29	3500
February	200
Mon 05	1000

Found 2 potential anomalies.

	date	value	is_anomaly	anomaly_type
0	2024-01-29	3500	<input checked="" type="checkbox"/>	High anomaly (value much higher than expected)
1	2024-01-31	200	<input checked="" type="checkbox"/>	Low anomaly (value much lower than expected)

See Statistical Details



Smart City Dashboard

localhost:8501

Smart City

Navigation

Dashboard

Policy Search

Chat

Eco Tips

KPI Forecasting

Anomaly Checker

Report Generator

Feedback

Quick Stats

Population

2.3M

Area

Sustainability Report for Sunport - Year 2024

Deploy

1. Air Quality

- Air Quality Index (AQI): 60

The AQI of 60 indicates that the air quality in Sunport is considered "Moderate," falling within the range that is acceptable and does not pose significant health risks to most individuals. Maintaining this level highlights effective regulatory measures and pollution controls.

Implication: Continued monitoring and enforcement of emission standards are crucial to sustain this air quality status. Any deviation might suggest a need for stricter enforcement or innovative air-purifying initiatives, like expanding green spaces or promoting zero-emission vehicles.

2. Energy Usage

- Total Energy Consumption (MWh): 14,000
- Renewable Energy (MWh): 8,000

Sunport demonstrates a commendable shift towards renewable energy sources, with 57% (8,000 MWh) of its total energy consumption coming from renewables in 2024.

Insight: The high share of renewables (57%) signifies progress in the city's energy transition strategy. However, there remains room for growth, particularly focusing on increasing the renewable energy fraction to approach or exceed 100% to meet sustainability goals fully.

3. Water Usage

- Total Water Usage (Gallons): 9,000,000

Smart City Dashboard

localhost:8501

Smart City

Navigation

Dashboard

Policy Search

Chat

Eco Tips

KPI Forecasting

Anomaly Checker

Report Generator

Feedback

Quick Stats

Population

2.3M

Area

Get Personalized Eco-Friendly Tips

Deploy

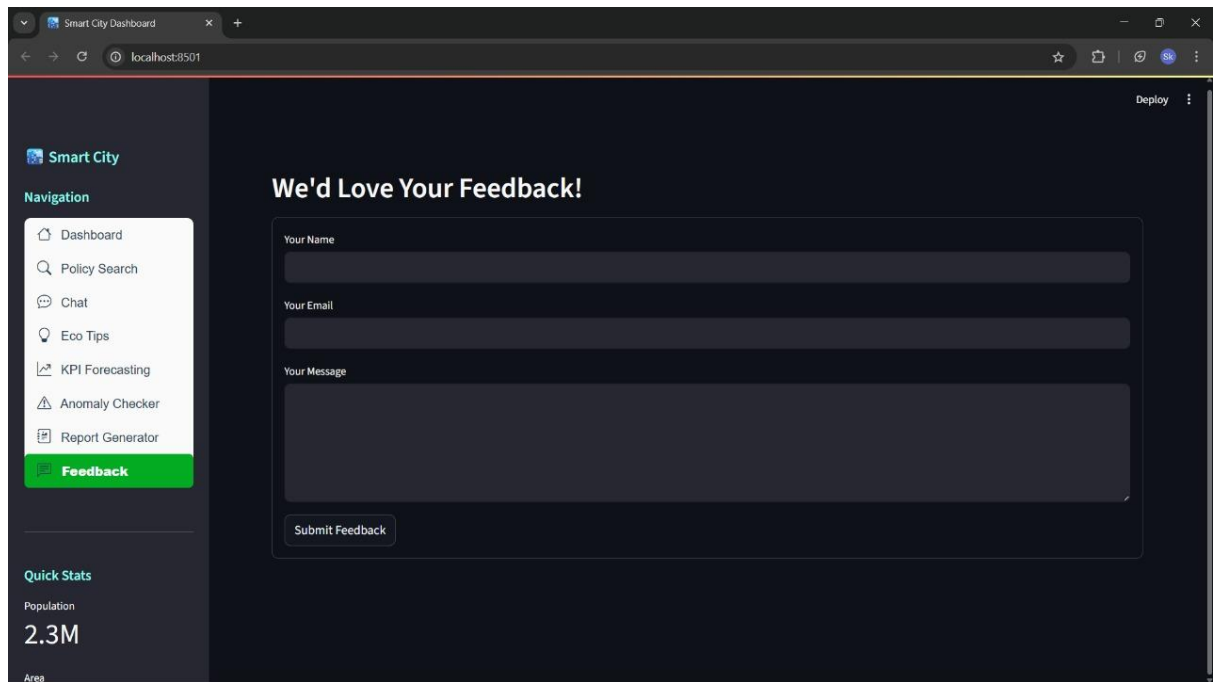
Enter a topic:

water

Get Eco Tip

Eco Tip for 'Water':

- Implement Rainwater Harvesting: Encourage buildings, both residential and commercial, to install rainwater harvesting systems. This captured rainwater can be used for non-potable purposes like irrigation, toilet flushing, and even car washing. City-wide initiatives, such as offering subsidies or tax benefits for installing such systems, can significantly increase participation.
- Establish Decentralized Water Treatment and Reuse: Deepen community involvement in local water management by setting up small-scale, decentralized wastewater treatment and reuse systems. These could be implemented in neighborhoods or districts, treating wastewater for non-potable uses like irrigation of public spaces, street cleaning, or even cooling towers for industrial processes. This not only reduces strain on centralized infrastructure but also fosters local sustainability consciousness and resilience.



12. Known Issues

High latency if Watsonx API quota is reached

Pinecone index size may hit limit in free tier

No persistent database for feedback collection

No built-in data validation for file types and schema mismatch

Doesn't support multiple concurrent users in demo version

These issues are known and documented for future patching.

13. Future enhancement

- **Adaptive Learning Models**

Functionality: Continuously improve recommendations and predictions by learning from user behavior, seasonal trends, and policy changes.

- **Multilingual & Cultural Adaptation**

Functionality: Support diverse populations by offering localized content, language options, and culturally relevant eco-tips.

- **Real-Time IoT Integration**

Functionality: Seamlessly connect with smart meters, traffic systems, and environmental sensors for live decision-making.

- **Automated Policy Drafting**
Functionality: Assist city officials in drafting sustainability policies using AI-generated templates and insights.
- **Climate Resilience Modeling**
Functionality: Simulate the impact of climate scenarios on infrastructure and suggest adaptive strategies.
- **Participatory Budgeting Tools**
Functionality: Enable citizens to vote on sustainability projects and track funding transparency.
- **Explainable AI (XAI)**
Functionality: Provide clear justifications for AI decisions to build trust among citizens and officials.
- **Satellite Data Integration**
Functionality: Use satellite imagery for urban heat mapping, green cover analysis, and disaster monitoring.
- Integrate real-time IoT sensors and GIS maps
- Expand language support
- Use Docker and GitHub Actions for deployment
- Add analytics dashboard for city officials
- Store and visualize trends over months/years
- Allow policy document OCR and PDF parsing
- This project is scalable and ready for real-world extension, making it a strong foundation for smart city innovation.