



SMARTBRIDGE

Let's Bridge the Gap

## Generative AI Project Using IBM Cloud Documentation

Project Title: Sustainable Smart City Assistant Using IBMGranite LLM

Team Member:

- **Kanna Nithin Ashok Kumar – (Team Lead - Backend Structure & Routing)**  
Led the backend architecture and handled API routing across modules. Ensured clean file structure and stable data flow.
- **Allaparthi Akshaya – (Backend Development & Integration)**  
Developed the core backend using FastAPI and integrated IBM Granite APIs. Ensured smooth connectivity between all modules.
- **Anumala Vishnu Vardhan– (Prompt Design & Model Testing)**  
Designed effective prompts and tested model outputs. Helped fine-tune responses for accuracy and relevance.
- **Battula Leela Krishna – (Module Coordination & Feature Improvement)**  
Managed module interaction and I/O flow. Suggested feature improvements to enhance overall user experience.

---

### 1. Introduction

The Sustainable Smart City Assistant is a generative AI-based platform designed to empower citizens with intelligent urban services. It provides eco-advice, monitors real-time city health data, summarizes documents, collects citizen feedback, and supports conversational interaction using LLMs.

---

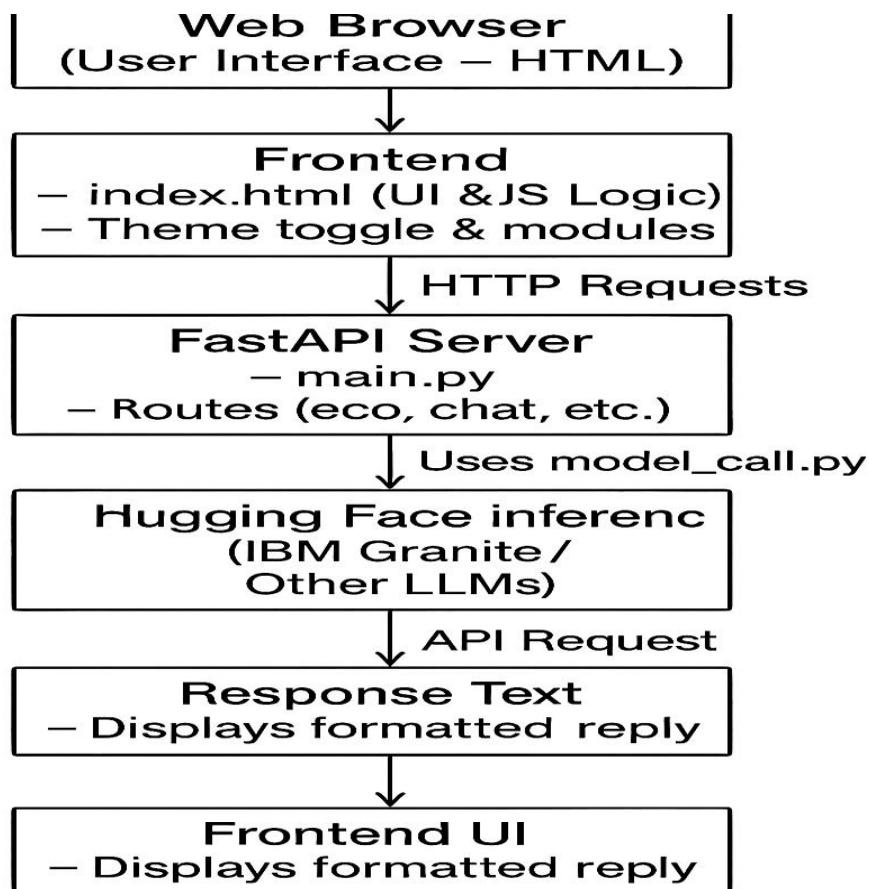
### 2. Project Overview

- **Objective:**  
Build an AI assistant to facilitate seamless communication between citizens and smart city services.
- **Features:**
  - Eco Advice Generator
  - City Health Dashboard (Air Quality, Noise, etc.)
  - Chat Assistant for Public Queries

- Document Summarization
  - AI-powered Feedback Collector
- 

### 3. Architecture

- Frontend:  
HTML, CSS, JS (Jinja2 Templates via FastAPI)
- Backend:  
FastAPI-based API service integrating Hugging Face models through secured .env configuration.
- AI Models:  
Used models from Hugging Face like distilgpt2, flan-t5-small, etc. for various tasks.





#### 4. Setup Instructions

- Prerequisites:  
Python 3.8+, Hugging Face account & API Token
- Installation:

```
git clone https://github.com/akshaya615/smart-city-assistant.git
```

```
cd smart-city-assistant
```

```
pip install -r requirements.txt
```

- .env File Example:

```
HUGGINGFACE_TOKEN=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
```

```
MODEL_ECO=https://api-inference.huggingface.co/models/mistralai/Mixtral-8x7B-Instruct-v0.1
```

```
MODEL_FEEDBACK=https://api-inference.huggingface.co/models/HuggingFaceH4/zephyr-7b-beta
```

```
MODEL_CHAT=https://api-inference.huggingface.co/models/HuggingFaceH4/zephyr-7b-beta
```

```
MODEL_HEALTH=https://api-inference.huggingface.co/models/mistralai/Mixtral-8x7B-Instruct-v0.1
```

```
MODEL_SUMMARY=https://api-inference.huggingface.co/models/sshleifer/distilbart-cnn-12-6
```

---

#### 5. Folder Structure

```
smart-city-assistant/
```

```
|   └── main.py
```

```
|   └── templates/
```

```
|       └── index.html
```

```
|   └── static/
```

```
|       └── smartcity2.jpeg
```

```
└── app/
```



SMARTBRIDGE

Let's Bridge the Gap

```
|   └── routes/
|   |   └── eco.py
|   |   └── chat.py
|   |   └── feedback.py
|   |   └── city_health.py
|   |   └── summarizer.py
|   └── utils/
|       └── model_call.py
└── requirements.txt
└── .env
└── README.md
```

---

## 6. Run the App

Start the local server:

```
uvicorn main:app --reload
```

Open browser at:

<http://localhost:8000>

---

## 7. API Overview

- POST to /eco\_advice, /chat\_assistant, /city\_health, etc.
- All routes forward to Hugging Face model with prompt:

```
{ "inputs": "<your query here>" }
```

- Headers:

```
{ "Authorization": "Bearer hf_your_token" }
```

---



## 8. Authentication

Token-based access via Hugging Face API Key stored in .env. No frontend login implemented yet.

---

## 9. UI/UX Features

- Responsive cards for modules
  - Toggle Theme (Dark/Light)
  - Animated popups for each module
  - Chat history shown inline
  - Suggestions listed for guidance
  - Scrollable formatted responses
- 

## 10. Testing

- API model response validation
  - UI tested on desktop and mobile
  - Error fallback and empty input handling
- 

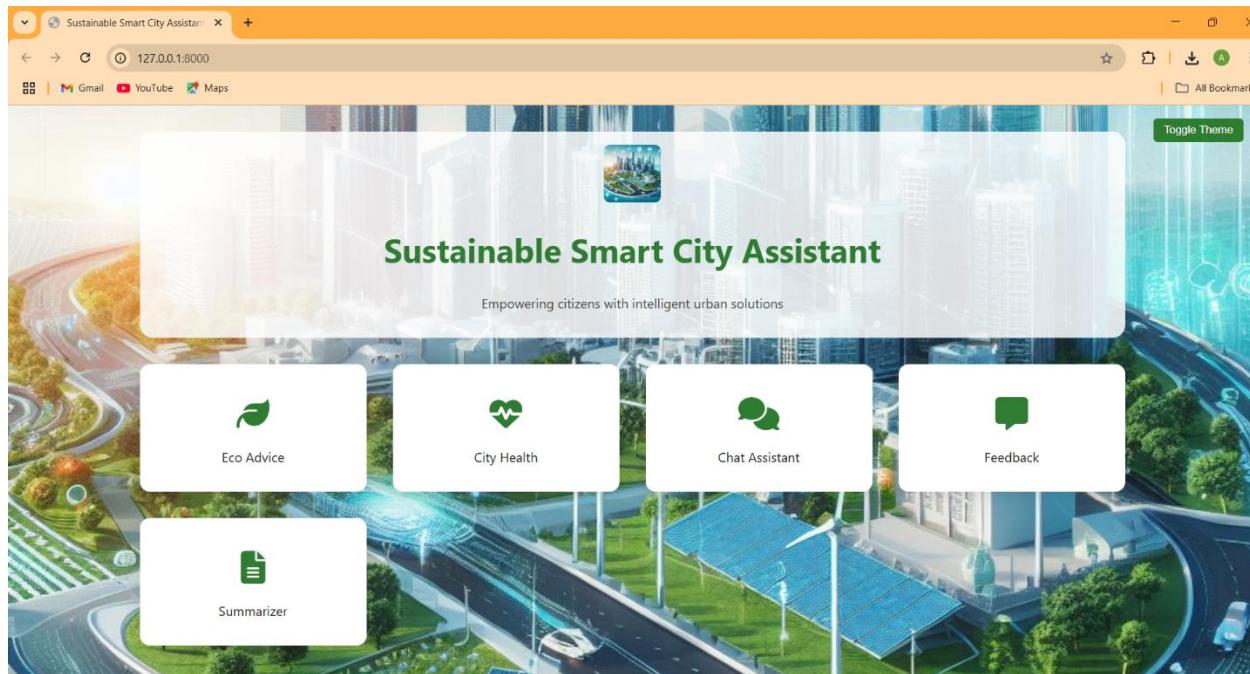
## 11. Demo Video

Input Codes:



**SMARTBRIDGE**  
Let's Bridge the Gap

## Output Codes:





SMARTBRIDGE  
Let's Bridge the Gap

The screenshot displays the SmartBridge AI interface with four main panels:

- City Health:** A form asking "Is noise pollution rising?" with a "Submit" button. Below it, a "Suggestions:" section lists:
  - Current AQI in my area
  - Is noise pollution rising?
  - Safe water zones todayA "Assistant:" section provides information about noise pollution, mentioning a study from 1996 and its impact on health.
- Chat Assistant:** A form asking "How to report garbage issue?" with a "Submit" button. Below it, a "Suggestions:" section lists:
  - Where is nearest EV charging station?
  - How to report garbage issue?
  - City smart card applicationA "Assistant:" section details the steps to report a garbage issue via the city's mobile app.
- Eco Advice:** A form asking "tips for water conservation" with a "Submit" button. Below it, a "Suggestions:" section lists:
  - Best ways to save energy at home
  - Time to replace plastic cups
  - Tips for water conservationA "Assistant:" section provides tips for water conservation, such as encouraging showers instead of baths and turning off the faucet while soaping.
- Summarizer:** A form asking "Paste document to summarize..." with a "Submit" button. Below it, a "Suggestions:" section lists:
  - Your In today's rapidly evolving world, technology plays a vital role in transforming societies and improving the quality of life. From smart cities to artificial intelligence, the integration of innovative solutions is helping address pressing global challenges. As we continue to embrace these advancements, it becomes essential to ensure that these technologies are developed responsibly, inclusively, and sustainably to benefit all members of society.A "Assistant:" section summarizes the provided text, noting the role of technology in transforming societies.

## 12. Known Issues

- Rate limits apply to free Hugging Face usage
- Some model responses are too verbose
- No persistent user state

## 13. Future Improvements

- Add user login and profile history
- Use real time sensors for city data
- Multilingual support
- Deployment to Render or Hugging Face Spaces