Sustainable Smart City Assistant Using IBM Granite LLM

1. INTRODUCTION

1.1 Project Overview

The Sustainable Smart City Assistant is an intelligent modular platform that empowers urban governance, enhances sustainability, and fosters citizen engagement. Built on IBM Watsonx's Granite LLM, FastAPI, and Streamlit, it delivers features such as policy summarization, eco tips, anomaly detection, and predictive analytics—all accessible through a unified dashboard.

1.2 Purpose

To simplify urban decision-making and promote proactive sustainability through AI-powered services, enabling city planners, citizens, and administrators to engage with structured and unstructured urban data more effectively.

2. IDEATION PHASE

2.1 Problem Statement

Cities lack a centralized AI interface that can help analyze urban KPIs, summarize complex policies, and support sustainable decision-making through citizen interaction and intelligent automation.

2.2 Empathy Map Canvas

- **Think & Feel**: Urban planners feel overwhelmed with data; citizens want their voices heard.
- **See**: Fragmented tools, inaccessible policy documents, slow issue response times.
- **Hear**: Complaints about lack of sustainability initiatives.
- **Say & Do**: Citizens raise concerns; administrators seek actionable insights.
- **Pain**: Delayed actions, underutilized data, public dissatisfaction.
- **Gain**: A single assistant that processes data, handles feedback, and offers smart suggestions.

2.3 Brainstorming

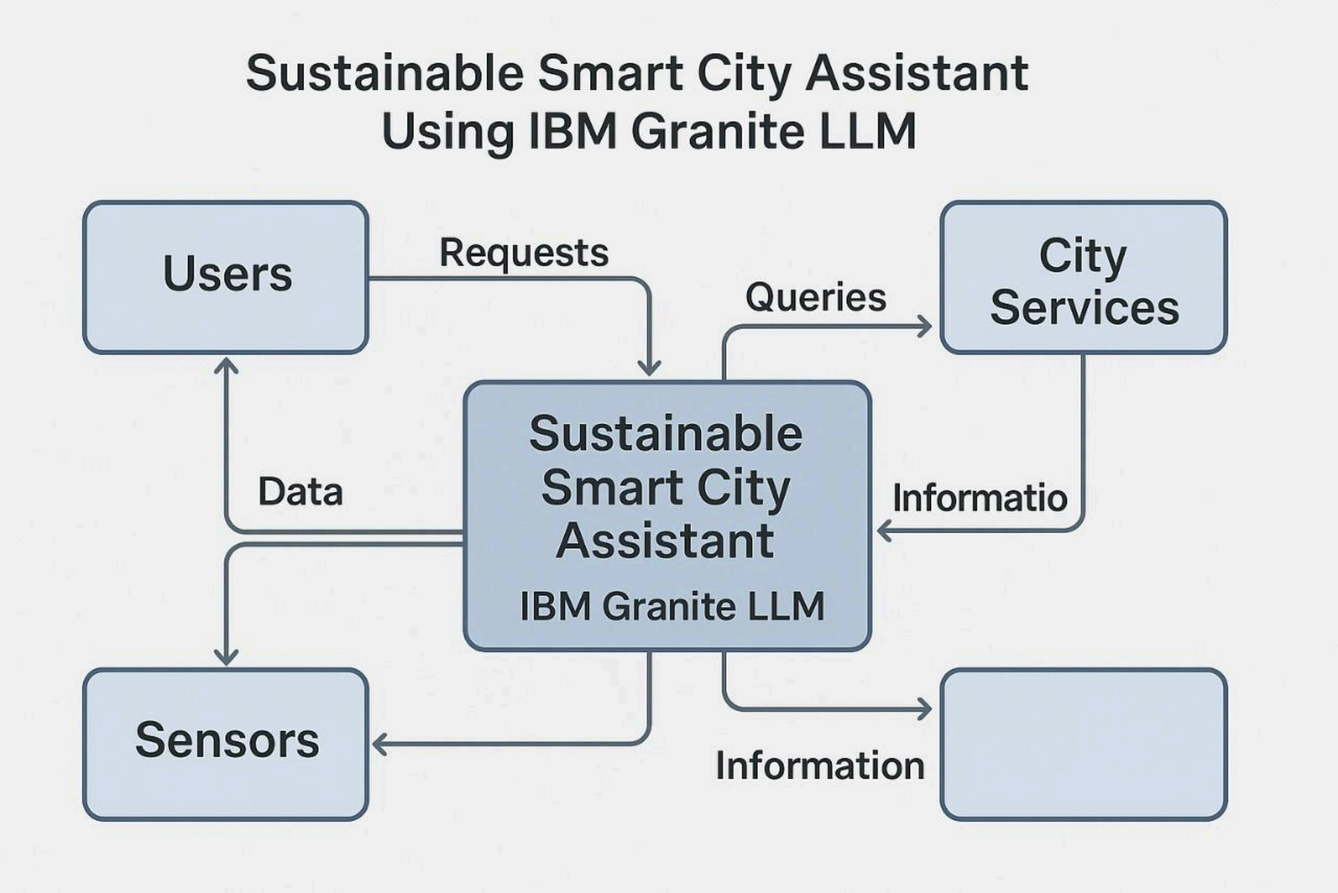
- Use LLMs to process and explain documents
- Deploy KPI forecasting via ML models
- Include anomaly detection for civic utilities
- Create eco-awareness tools for schools
- Design intuitive dashboards using Streamlit

3. REQUIREMENT ANALYSIS

3.1 Solution Requirements

- Modular backend (FastAPI)
- Interactive UI (Streamlit)
- Secure environment setup
- External service integration (Watsonx, Pinecone)

3.2 Data Flow Diagram



3.4 Technology Stack

- IBM Watsonx Granite LLM
- Pinecone Vector Search
- FastAPI, Streamlit
- Python (Pandas, Scikit-learn, dotenv, pydantic)
- Sentence-transformers, Matplotlib

4. PROJECT DESIGN

4.1 Problem-Solution Fit

Bridges the gap between policy complexity and public understanding while giving administrators a data-driven assistant.

4.2 Proposed Solution

An AI assistant that supports summarization, forecasting, anomaly detection, and eco-awareness via a streamlined dashboard and modular backend.

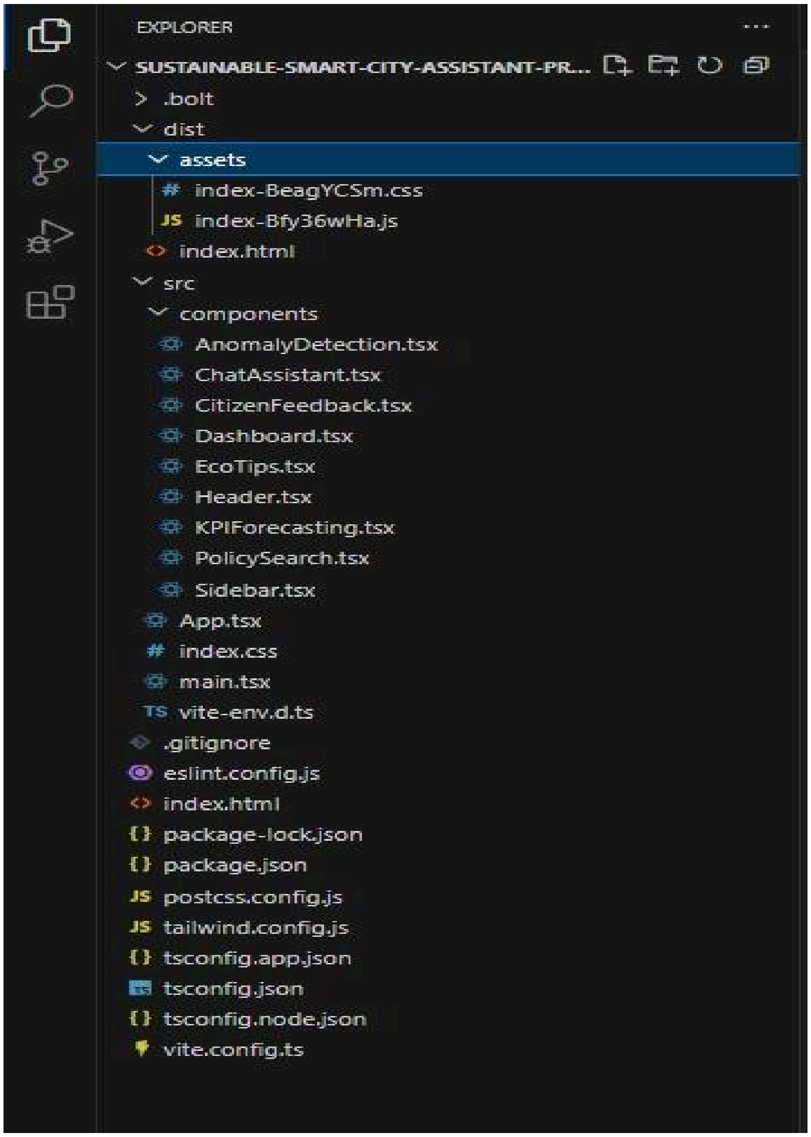
4.3 Solution Architecture

Componentized backend → FastAPI routers Embedding engine → Pinecone + Sentence Transformers LLM Services → Granite LLM UI Dashboard → Streamlit frontend with sidebar navigation

5. PROJECT PLANNING & SCHEDULING

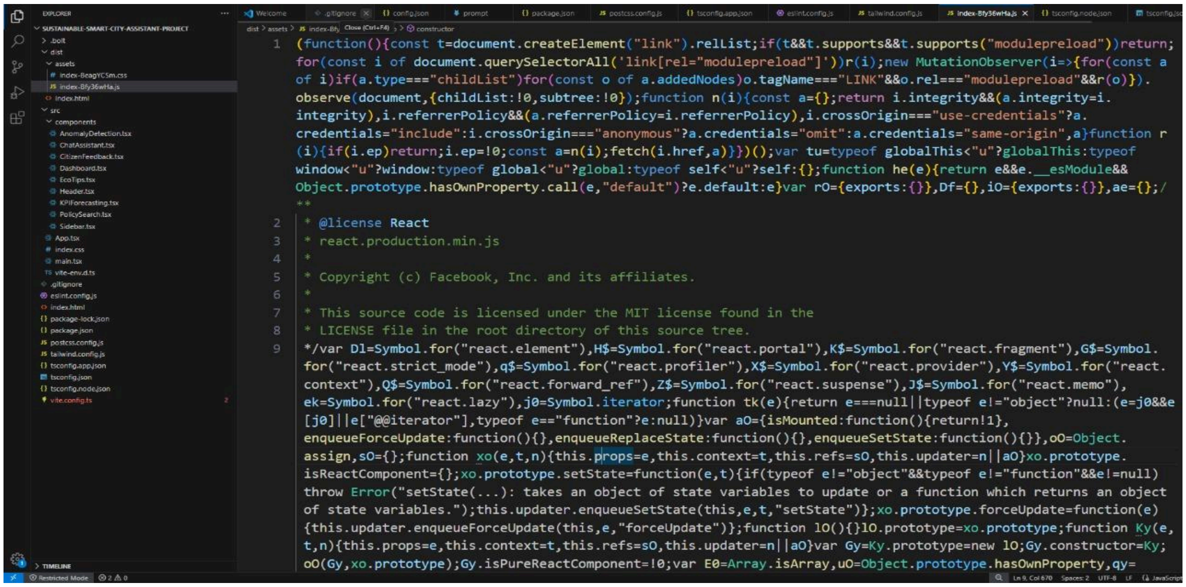
5.1 Project Planning

Phase 1: Initialization & Environment Setup



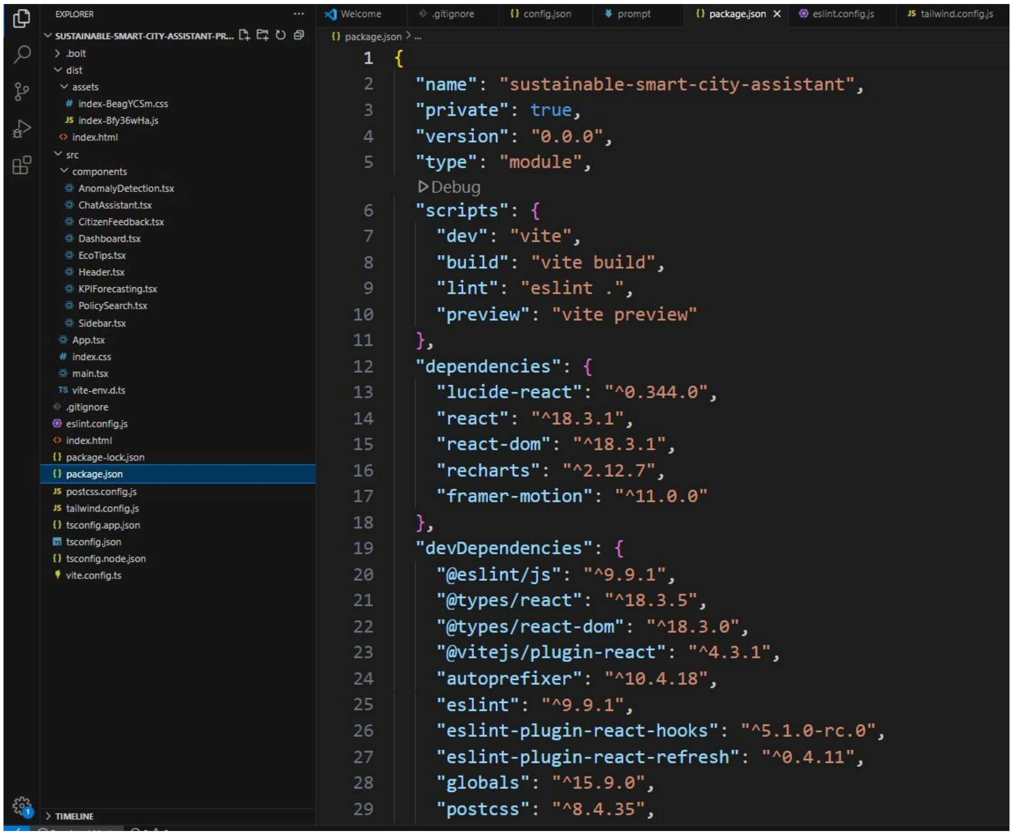
```
EXPLORER                                              ...
∨ SUSTAINABLE-SMART-CITY-ASSISTANT-PR...
  > .bolt
  ∨ dist
    ∨ assets
        index-BeagYCSm.css
        index-Bfy36wHa.js
      index.html
  ∨ src
    ∨ components
        AnomalyDetection.tsx
        ChatAssistant.tsx
        CitizenFeedback.tsx
        Dashboard.tsx
        EcoTips.tsx
        Header.tsx
        KPIForecasting.tsx
        PolicySearch.tsx
        Sidebar.tsx
      App.tsx
      index.css
      main.tsx
      vite-env.d.ts
    .gitignore
    eslint.config.js
    index.html
    package-lock.json
    package.json
    postcss.config.js
    tailwind.config.js
    tsconfig.app.json
    tsconfig.json
    tsconfig.node.json
    vite.config.ts
```

- **Phase 2**: LLM Integration & API Key Management



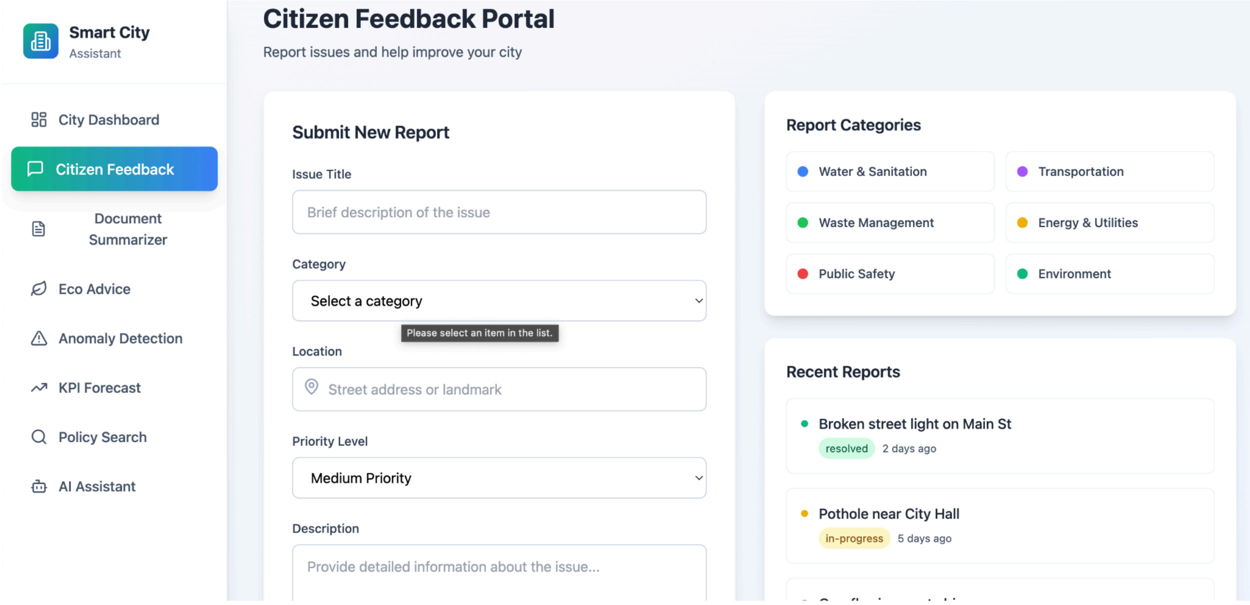
```
(function(){const t=document.createElement("link").relList;if(t&&t.supports&&t.supports("modulepreload"))return;
for(const i of document.querySelectorAll('link[rel="modulepreload"]'))r(i);new MutationObserver(i=>{for(const a
of i)if(a.type==="childList")for(const o of a.addedNodes)o.tagName==="LINK"&&o.rel==="modulepreload"&&r(o)}).
observe(document,{childList:!0,subtree:!0});function n(i){const a={};return i.integrity&&(a.integrity=i.
integrity),i.referrerPolicy&&(a.referrerPolicy=i.referrerPolicy),i.crossOrigin==="use-credentials"?a.
credentials="include":i.crossOrigin==="anonymous"?a.credentials="omit":a.credentials="same-origin",a}function r
(i){if(i.ep)return;i.ep=!0;const a=n(i);fetch(i.href,a)}})();var tu=typeof globalThis<"u"?globalThis:typeof
window<"u"?window:typeof global<"u"?global:typeof self<"u"?self:{};function he(e){return e&&e.__esModule&&
Object.prototype.hasOwnProperty.call(e,"default")?e.default:e}var rO={exports:{}},Df={},iO={exports:{}},ae={};/
**
 * @license React
 * react.production.min.js
 *
 * Copyright (c) Facebook, Inc. and its affiliates.
 *
 * This source code is licensed under the MIT license found in the
 * LICENSE file in the root directory of this source tree.
 */var Dl=Symbol.for("react.element"),H$=Symbol.for("react.portal"),K$=Symbol.for("react.fragment"),G$=Symbol.
 for("react.strict_mode"),q$=Symbol.for("react.profiler"),X$=Symbol.for("react.provider"),Y$=Symbol.for("react.
 context"),Q$=Symbol.for("react.forward_ref"),Z$=Symbol.for("react.suspense"),J$=Symbol.for("react.memo"),
 ek=Symbol.for("react.lazy"),j0=Symbol.iterator;function tk(e){return e===null||typeof e!="object"?null:(e=j0&&e
 [j0]||e["@@iterator"],typeof e=="function"?e:null)}var aO={isMounted:function(){return!1},
 enqueueForceUpdate:function(){},enqueueReplaceState:function(){},enqueueSetState:function(){}},oO=Object.
 assign,sO={};function xo(e,t,n){this.props=e,this.context=t,this.refs=sO,this.updater=n||aO}xo.prototype.
 isReactComponent={};xo.prototype.setState=function(e,t){if(typeof e!="object"&&typeof e!="function"&&e!=null)
 throw Error("setState(...): takes an object of state variables to update or a function which returns an object
 of state variables.");this.updater.enqueueSetState(this,e,t,"setState")};xo.prototype.forceUpdate=function(e)
 {this.updater.enqueueForceUpdate(this,e,"forceUpdate")};function lO(){}lO.prototype=xo.prototype;function Ky(e,
 t,n){this.props=e,this.context=t,this.refs=sO,this.updater=n||aO}var Gy=Ky.prototype=new lO;Gy.constructor=Ky;
 oO(Gy,xo.prototype);Gy.isPureReactComponent=!0;var E0=Array.isArray,uO=Object.prototype.hasOwnProperty,qy=
```

- **Phase 3**: Backend Modular API Development



```
{
  "name": "sustainable-smart-city-assistant",
  "private": true,
  "version": "0.0.0",
  "type": "module",
  "scripts": {
    "dev": "vite",
    "build": "vite build",
    "lint": "eslint .",
    "preview": "vite preview"
  },
  "dependencies": {
    "lucide-react": "^0.344.0",
    "react": "^18.3.1",
    "react-dom": "^18.3.1",
    "recharts": "^2.12.7",
    "framer-motion": "^11.0.0"
  },
  "devDependencies": {
    "@eslint/js": "^9.9.1",
    "@types/react": "^18.3.5",
    "@types/react-dom": "^18.3.0",
    "@vitejs/plugin-react": "^4.3.1",
    "autoprefixer": "^10.4.18",
    "eslint": "^9.9.1",
    "eslint-plugin-react-hooks": "^5.1.0-rc.0",
    "eslint-plugin-react-refresh": "^0.4.11",
    "globals": "^15.9.0",
    "postcss": "^8.4.35",
```

- **Phase 4**: Streamlit UI Creation



Citizen Feedback Portal

Report issues and help improve your city

Submit New Report

Issue Title: Brief description of the issue

Category: Select a category

Please select an item in the list.

Location: Street address or landmark

Priority Level: Medium Priority

Description: Provide detailed information about the issue...

Report Categories

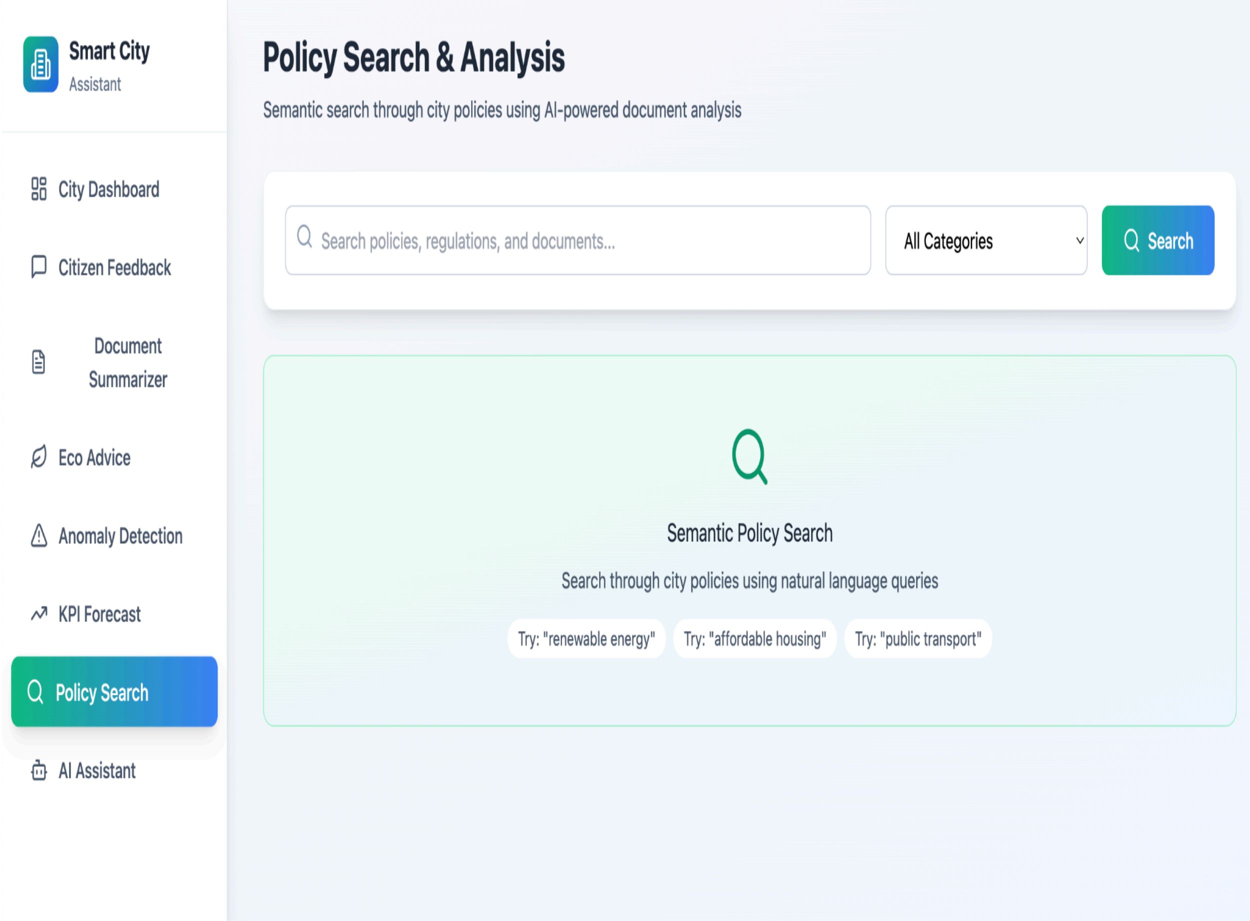
● Water & Sanitation
● Transportation
● Waste Management
● Energy & Utilities
● Public Safety
● Environment

Recent Reports

● Broken street light on Main St — resolved — 2 days ago

● Pothole near City Hall — in-progress — 5 days ago

- **Phase 5**: Embedding + Semantic Search



Policy Search & Analysis

Semantic search through city policies using AI-powered document analysis

Search policies, regulations, and documents...

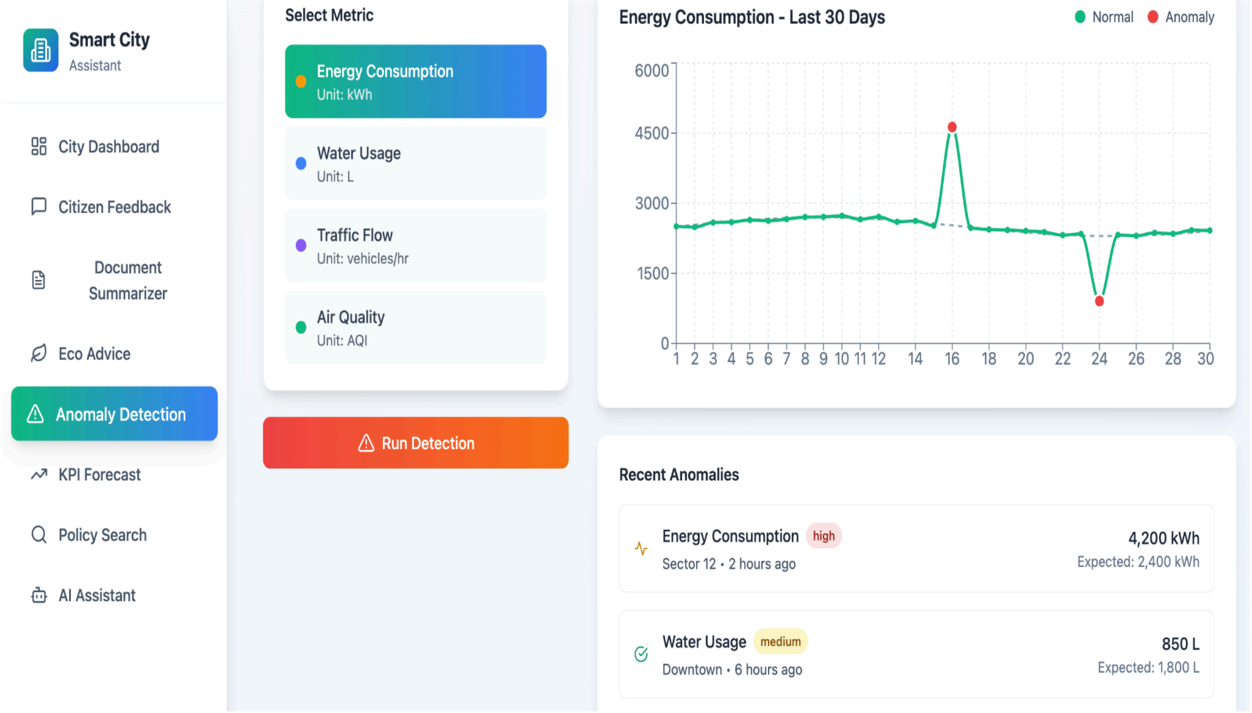
All Categories

Semantic Policy Search

Search through city policies using natural language queries

Try: "renewable energy" Try: "affordable housing" Try: "public transport"

- **Phase 6**: Forecasting + Anomaly Detection



Select Metric

- Energy Consumption — Unit: kWh
- Water Usage — Unit: L
- Traffic Flow — Unit: vehicles/hr
- Air Quality — Unit: AQI

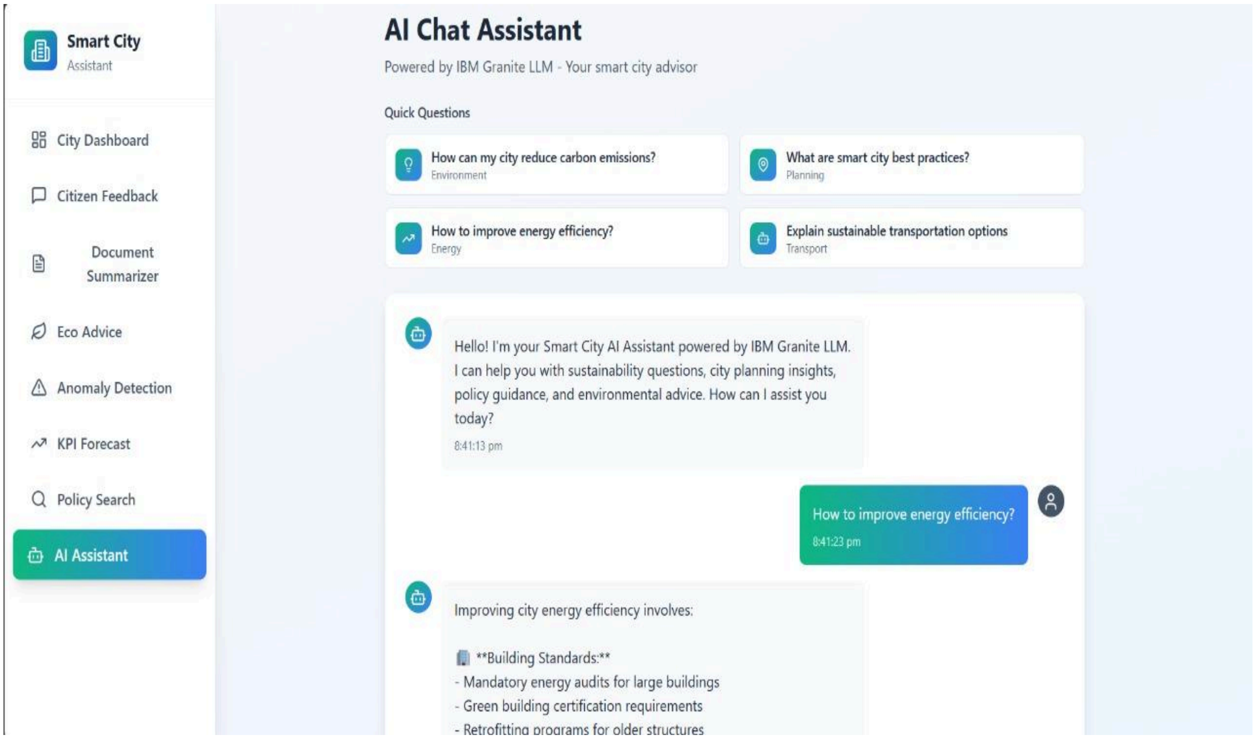
Energy Consumption - Last 30 Days

Recent Anomalies

Energy Consumption high — Sector 12 • 2 hours ago — 4,200 kWh — Expected: 2,400 kWh

Water Usage medium — Downtown • 6 hours ago — 850 L — Expected: 1,800 L

- **Phase 7**: Report Generation + Chat Assistant



AI Chat Assistant

Powered by IBM Granite LLM - Your smart city advisor

Quick Questions

- How can my city reduce carbon emissions? — Environment
- What are smart city best practices? — Planning
- How to improve energy efficiency? — Energy
- Explain sustainable transportation options — Transport

Hello! I'm your Smart City AI Assistant powered by IBM Granite LLM. I can help you with sustainability questions, city planning insights, policy guidance, and environmental advice. How can I assist you today?

8:41:13 pm

How to improve energy efficiency?

8:41:23 pm

Improving city energy efficiency involves:

🏢 **Building Standards:**
- Mandatory energy audits for large buildings
- Green building certification requirements
- Retrofitting programs for older structures

- **Phase 8**: Final Integration & Testing

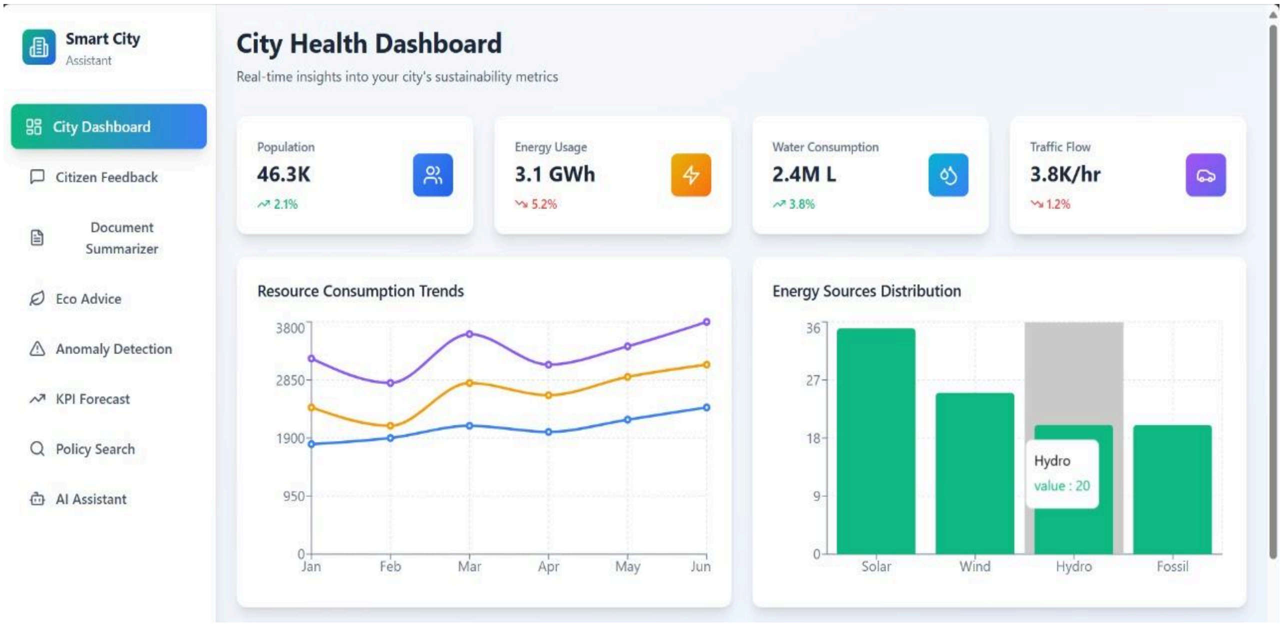
6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- All API endpoints tested using Swagger UI
- Stress-tested document summarization and semantic search
- Load-tested KPI forecasting and anomaly modules with large CSVs

7. RESULTS

7.1 Output Screenshots



City Health Dashboard

Real-time insights into your city's sustainability metrics

Population 46.3K 2.1%

Energy Usage 3.1 GWh 5.2%

Water Consumption 2.4M L 3.8%

Traffic Flow 3.8K/hr 1.2%

Hydro value : 20

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Modular and scalable
- Citizen-centric design
- Real-time forecasting and anomaly detection
- Leverages trusted LLM (Granite) and semantic search (Pinecone)

Disadvantages:

- API keys dependency
- Initial setup complexity
- Limited offline usability

9. CONCLUSION

The Sustainable Smart City Assistant demonstrates how AI can integrate urban data, promote transparency, and foster responsible civic participation. It sets a foundation for data-driven smart governance and scalable public services.

10. FUTURE SCOPE

- Voice-enabled assistant for visually impaired users
- Integration with IoT sensors for real-time KPI updates
- Multilingual chat support
- Deployment on cloud platforms for wider reach

11. APPENDIX

GitHub_Repository_Link:https://github.com/SANDEEPREDDY2205/Sustainable-Smart-City-Assistant-Using-IBM-Granite-LLM

GitHub/Project Demo:https://fastidious-tapioca-eee015.netlify.app