Project Design Phase
Solution Architecture

Date	28 June 2025
Team ID	LTVIP2025TMID60547
Project Name	**Sustainable Smart City Assistant Using IBM Granite LLM**
Maximum Marks	4 Marks

Solution Architecture:

1. Frontend Layer (User Interface)

- Built using **Streamlit** for an intuitive dashboard experience.
- Components include:
  - smart_dashboard.py for centralized UI
  - feedback_form.py, eco_tips.py, chat_assistant.py for module-based interaction
- Features include dynamic KPI cards, toast notifications, sidebar navigation, and eco-themed design.

2. API Layer (Backend Services)

- Powered by **FastAPI**, serving modular endpoints such as:
  - /chat → powered by Watsonx Granite LLM
  - /feedback → logs and categorizes citizen reports
  - /summarize and /search → document summarization and semantic search via Pinecone
  - /forecast-kpi and /detect-anomaly → for KPI-based analytics

3. AI & ML Services

- **IBM Watsonx Granite LLM**: Handles chat, policy summarization, eco tips, and report generation.
- **ML Models** (Scikit-learn): Linear Regression for KPI forecasting; statistical techniques for anomaly detection.
- **Vector Embedding**: Sentence-transformers convert documents into vectors for semantic search via Pinecone.

4. Data Layer

- **Pinecone Vector Database**: Stores vectorized policy documents for similarity-based retrieval.
- **File Ingestion**: CSV/JSON/Text for KPI data and document uploads.

- **Environment Config**: Managed via .env and pydantic, protecting API credentials.

5. Process Flow

1. **User Input** via Streamlit: prompts, file uploads, and feedback forms.
2. **API Request** routed via FastAPI endpoints.
3. **LLM or ML Model Execution**: Watsonx generates responses; ML forecasts or detects anomalies.
4. **Storage or Retrieval**: Pinecone and file loaders fetch or store data as needed.
5. **Response Rendering**: AI-generated results displayed back in the Streamlit interface.

Example - Solution Architecture Diagram:

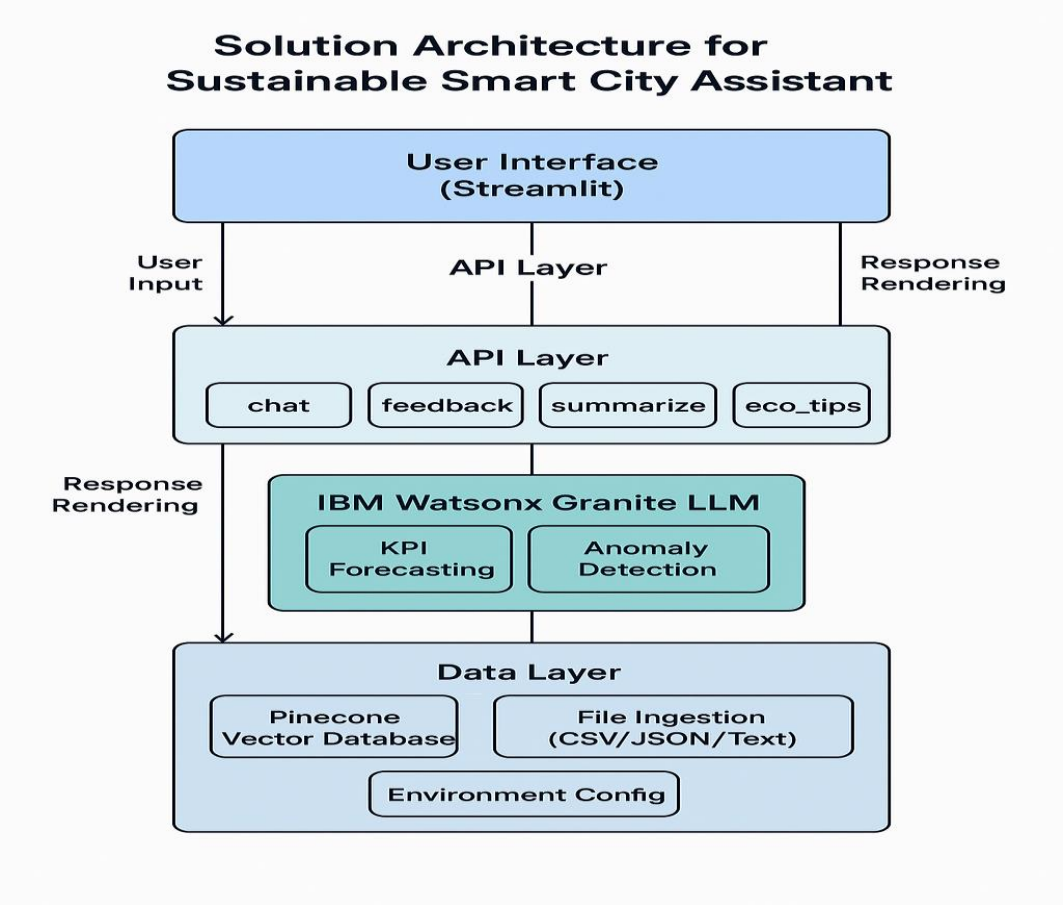

Figure 1: Architecture for Sustainable Smart City Assistant using IBM Granite LLM