Sustainable Smart City Assistant — Project Documentation

Project Phases Template

Project Title:
Sustainable Smart City Assistant

Team Name:
EcoTech Minds

Team Members:
P V S L Deepthi

Phase-1: Brainstorming & Ideation

Objective:
Identify the problem statement, define the purpose and impact of the project.

Key Points:

1. Problem Statement:
Urban citizens often struggle to access clear, timely, and localized information about sustainable resources, public services, and environmental metrics in their city.

2. Proposed Solution:
Develop a public-facing web application powered by IBM Granite 2b-Instructor and FastAPI with AI-driven services.

3. Target Users:
Urban citizens, Environmental enthusiasts, City planners, Policy researchers.

4. Expected Outcome:
A modular, scalable assistant empowering users for eco-friendly decisions and city services access.

Phase-2: Requirement Analysis

Objective:
Define technical and functional requirements.

Key Points:

1. Technical Requirements:
Python, FastAPI, IBM Granite, APIs, SQLite/PostgreSQL, Render/IBM Cloud.

2. Functional Requirements:
Chatbot, Service finder, Dashboard, Alerts, Transparency hub.

3. Constraints & Challenges:
API limits, real-time data retrieval, data privacy compliance, hosting constraints.

Phase-3: Project Design

Objective:
Create the architecture and user flow.

Key Points:

1. System Architecture Diagram:
[User] → [Frontend] → [Backend] → [AI Model/APIs/Database]

2. User Flow:
Homepage → Chatbot/Features → API Processing → Result Display.

3. UI/UX Considerations:
Eco-themed layout, Map navigation, Card dashboards, Chatbot icon, Data panel.

Phase-4: Project Planning (Agile Methodologies)

Objective:
Break down tasks using Agile.

Key Points:

1. Sprint Planning:
5 sprints for modules.

2. Task Allocation:
Members assigned per module.

3. Timeline & Milestones:
Weekly module deadlines.

Phase-5: Project Development

Objective:
Code and integrate components.

Key Points:

1. Technology Stack Used:
Python, FastAPI, IBM Granite, SQLite/PostgreSQL, APIs, Render.

2. Development Process:
Route setup, API integration, Frontend building, Real-time calls, Privacy controls.

3. Challenges & Fixes:
API errors, map lags, AI latency, privacy UX fixes.

Phase-6: Functional & Performance Testing

Objective:
Ensure the project works as expected.

Key Points:

1. Test Cases Executed:
Chatbot, Service finder, Dashboard, Alerts, Document Q&A, Data privacy tests.

2. Bug Fixes & Improvements:
Map API caching, voice input fallback, payload optimizations.

3. Final Validation:
Project meets requirements.

4. Deployment:
Demo hosted on Render (link to be added).