

Citizen AI – Project Documentation

1. Introduction

- Project Title: Citizen AI
- Team Members:
 - - Ansar Ali
 - - Buvanesh
 - - Shatrock
 - - Ashwin Kumar

2. Project Overview

Purpose: Citizen AI is an intelligent citizen engagement platform designed to enhance interaction between governments and the public. Using AI and real-time data, the assistant provides city safety insights, government service query resolution, and sentiment analysis of citizen feedback.

Features:

- Conversational Interface – Natural language interaction for citizens.
- City Safety Analysis – Summarized crime and accident statistics.
- Sentiment Analysis – Classifies citizen feedback.
- Citizen Feedback Dashboard – Visual analytics of sentiment distribution.
- Gradio UI – Interactive multi-tab dashboard.

3. Architecture

- Frontend: Built with Gradio (interactive UI).
- Backend: Python + Transformers + Torch.
- LLM Integration: IBM Granite model.
- Visualization: Matplotlib for charts.

4. Setup Instructions

- Python 3.9+ required.
- Install dependencies: transformers, torch, gradio, textblob, matplotlib.
- Run: python app.py
- Access Gradio dashboard in browser.

5. Folder Structure

```
project/ ├── app.py # Main script ├── requirements.txt # Dependencies ├── models/ # Model configs ├── ui/ # Gradio layouts ├── utils/ # Utility functions
```

6. Running the Application

- Launch Gradio app.
- Use City Analysis, Citizen Assistant, Sentiment Analysis, and Dashboard tabs.

7. API Documentation

- POST /city-analysis – Returns AI-generated safety report.
- POST /citizen-assist – Handles citizen queries.
- POST /analyse-sentiment – Performs sentiment analysis.

8. Authentication

Currently open for demo use. Future enhancements include token-based authentication, role-based access (citizens, officials, admins).

9. User Interface

Tabbed layout for each feature, dashboard visualization of sentiment, and clean accessible design for non-technical users.

10. Testing

- Unit Testing – for AI response functions.
- Manual Testing – UI flows.
- Edge Cases – large queries, empty input.

11. Known Issues

- AI sometimes repeats prompt text.
- Model slower on CPU.

12. Future Enhancements

- Integration with real-time city data APIs.
- Multi-language support.
- Role-based authentication.
- Advanced analytics and forecasting.