

1. Introduction

1.1 Purpose

The purpose of this project is to create an AI-powered system that helps analyze city-level safety information and provides accurate responses to citizen queries related to public services, policies, and civic issues. This system integrates a Large Language Model (LLM) with an interactive Gradio interface, making it accessible for both government authorities and citizens.

1.2 Scope

The system offers two major functionalities:

1. City Analysis: Provides insights into crime index, accident rates, and overall safety assessment for a given city.
2. Citizen Services: Acts as a digital assistant to respond to queries about government policies, civic services, and public information.

The application runs on Google Colab with T4 GPU support to ensure faster model execution and response generation.

1.3 Objectives

Deliver structured analysis about city safety and accident statistics.

Offer reliable information to citizens about government services.

Provide an easy-to-use interface with minimal technical barriers.

Ensure real-time AI-driven responses with a scalable approach.

2. Functional Requirements

2.1 City Analysis Module

Input: City name (e.g., Mumbai, New York, London).

Process:

Generate AI prompt for crime index, accident rates, and safety analysis.

Use the Granite-3.2-2B-Instruct model for response generation.

Output: A detailed text report containing crime statistics, traffic safety, and overall assessment.

2.2 Citizen Services Module

Input: Citizen query related to policies, governance, or civic issues.

Process:

Create a prompt that guides the AI to answer in an official and helpful manner.

Generate a response using the model.

Output: AI-generated government-style response in simple language.

2.3 User Interface Requirements

Must include two tabs:

1. City Analysis Tab – Input city name, get analysis in textbox output.
2. Citizen Services Tab – Input query, get response in textbox output.

Buttons: "Analyze City" and "Get Information".

Outputs should be scrollable and multi-line textboxes for readability.

2.4 System Performance Requirements

Should generate responses within 5–10 seconds on T4 GPU.

Must handle large prompts up to 1024 tokens.

Must avoid hallucinations by ensuring prompt clarity.

3. Non-Functional Requirements

3.1 Usability

The interface must be user-friendly with simple text inputs.

Results should be clear, structured, and easy to read.

3.2 Reliability

System must provide consistent outputs for repeated queries.

Must handle unexpected inputs (e.g., misspelled city names) gracefully.

3.3 Scalability

Designed to integrate with other public data APIs in the future (e.g., live crime databases, traffic statistics).

Should support migration to cloud-based hosting beyond Colab.

3.4 Security

No sensitive citizen data will be collected.

The app only processes text inputs without storing private information.

3.5 Maintainability

Codebase is modular with separated functions for analysis and interaction.

Easy to upgrade the model or integrate new AI backends.

4. System Requirements

4.1 Hardware Requirements

Processor: T4 GPU (for Colab runtime).

RAM: Minimum 12 GB (recommended 16 GB for smooth execution).

Storage: 2 GB available for model and dependencies.

4.2 Software Requirements

Operating System: Runs on Google Colab (Linux backend).

Programming Language: Python 3.10+.

Libraries:

transformers (for model handling).

torch (for tensor operations and GPU support).

gradio (for user interface).

5. Use Case Scenarios

5.1 City Analysis Example

Input: "Mumbai"

Output:

Crime index: medium-high

Accident rates: increasing in traffic-heavy zones

Safety assessment: moderate, with improved police surveillance in some areas

5.2 Citizen Services Example

Input: "How do I apply for a government housing scheme?"

Output:

AI provides step-by-step instructions for applying, including eligibility criteria, required documents, and online portal links.

6. Future Enhancements

Integration with real-time government databases for accurate crime and accident data.

Support for multilingual outputs to reach diverse populations.

Voice input/output support for accessibility.

Expansion to include environmental statistics (pollution, water quality, etc.).

7. Conclusion

The City Analysis & Citizen Services AI system bridges the gap between citizens and government services using AI-driven insights. It not only assists in decision-making for authorities but also provides citizens with quick and reliable information. Running on Google Colab with T4 GPU ensures fast, scalable, and cost-effective deployment.