**Project Title: CITIZEN AI : Intelligent Citizen Engagement Platform**

**Project Documentation**

1. Introduction

Project Title:\* Citizen AI: Intelligent Citizen Engagement Platform

Team Members:

1. Dhanush. S

2. Sachin krishna. S

3. Aathi Narayanan. K. V

4. Sanjay. P. S

2. Project Overview

Purpose:

The purpose of the Citizen AI platform is to strengthen democratic governance and improve citizen-government interaction through AI-powered tools. By using natural language processing and data-driven insights, the platform empowers citizens to access services, understand policies, and provide feedback in real-time. For government officials, it serves as a decision-support assistant—summarizing public sentiment, forecasting needs, and helping to design inclusive policies. Ultimately, Citizen AI creates a bridge between citizens and government for smarter, transparent, and more responsive governance.

Features:

Conversational Interface

Key Point: AI-driven chatbot

Functionality: Citizens ask questions about schemes, services, or issues and get instant, natural-language responses.

Policy Summarization

Key Point: Simplified understanding

Functionality: Converts lengthy policy or scheme documents into short, actionable summaries.

Citizen Feedback Loop

Key Point: Public participation

Functionality: Collects and analyzes citizen input to improve governance and services.

Predictive Analytics

Key Point: Future forecasting

Functionality: Anticipates resource demand, service usage, or citizen concerns using past and real-time data.

Multilingual Support

Key Point: Inclusivity

Functionality: Supports queries in multiple Indian languages to reach all sections of society.

Smart Alerts & Notifications

Key Point: Real-time updates

Functionality: Provides reminders about deadlines, scheme eligibility, or civic emergencies.

3. Architecture

Frontend (Web / Mobile App):

Interactive UI with chatbot, dashboards, complaint forms, and service tracking.

Backend (FastAPI / Node.js):

Handles citizen queries, feedback storage, and AI model integration.

LLM Integration (OpenAI / Watsonx):

Natural language understanding for queries and summarization.

Database (SQL / NoSQL):

Stores citizen requests, government policies, and analytics data.

Analytics & ML Models:

Forecasting service demand, analyzing citizen sentiment, detecting anomalies in service delivery.

4. Setup Instructions

Prerequisites:

Python 3.9 or Node.js

API keys for AI model integration

Cloud/Server environment

Database setup

Installation:

1. Clone repository

2. Install dependencies

3. Configure environment variables

4. Run backend server

5. Launch frontend dashboard

6. Connect database and AI APIs

5. Folder Structure

app/\* – Backend logic

app/api/\* – APIs for chat, feedback, and reports

ui/\* – Frontend dashboard and citizen portal

analytics/\* – Forecasting and sentiment analysis modules

report\\_generator/\* – Generates AI-based citizen service reports

6. Running the Application

1. Start backend server

2. Open citizen portal (web/mobile)

3. Citizens interact via chatbot or forms

4. Officials view reports, analytics, and feedback dashboards

7. API Documentation

POST /ask-query\* – Citizen submits a question, AI responds

POST /upload-policy\* – Upload government documents for summarization

GET /feedback\* – Retrieve aggregated citizen feedback

GET /alerts\* – Fetch notifications for schemes/events

8. Authentication

Role-based access (citizen, officer, admin)

Secure login with OTP or Aadhaar-linked identity

Token-based authentication for API calls

9. User Interface

Citizen-facing chatbot & feedback forms

Government dashboard with analytics, sentiment reports, and heatmaps of issues

Multilingual text and voice input

10. Testing

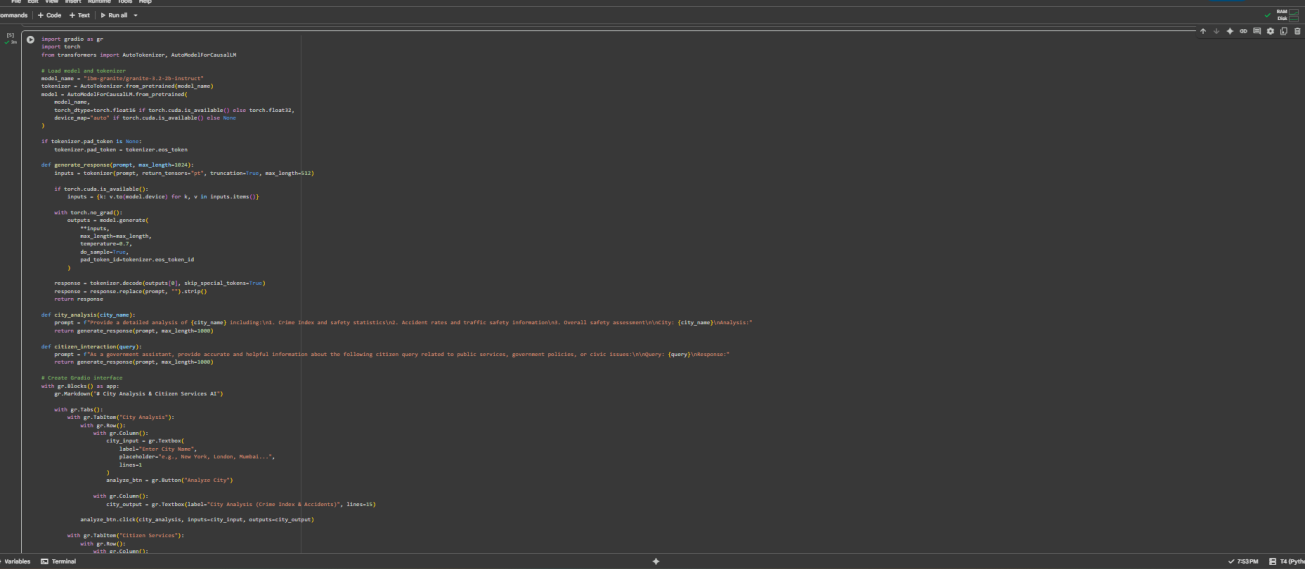
Unit Testing:\* Chatbot responses, summarization accuracy

API Testing:\* Swagger/Postman verification

Manual Testing:\* Citizen query flows, complaint logging

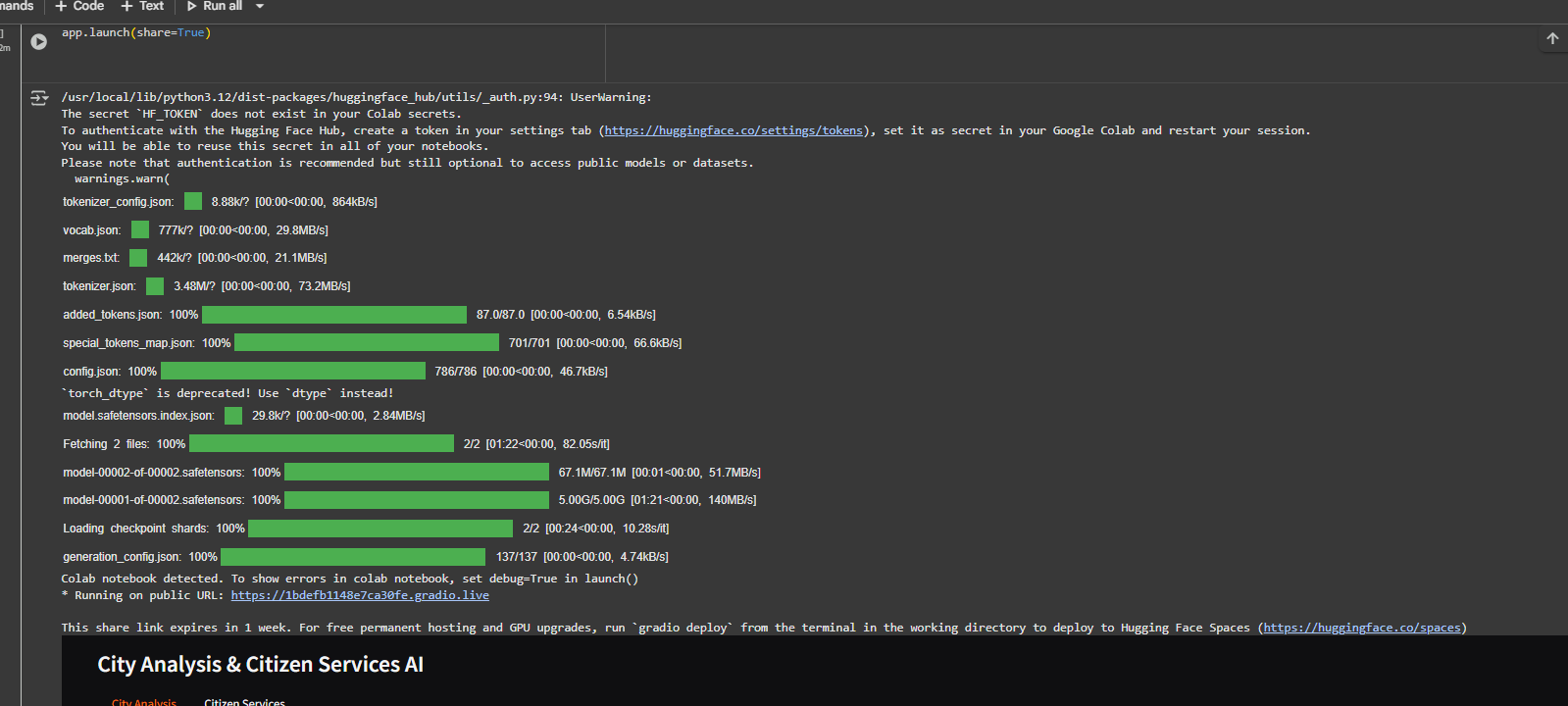
Edge Cases:\* Wrong inputs, unsupported languages

11. Screenshots





OUTPUT:



12. Known Issues

Limited offline support

Data privacy risks if not encrypted

­­­­­­­­­­­13. Future Enhancements

Blockchain for secure citizen data storage

IoT integration for smart city alerts

Voice-based assistants for rural areas

AI bias detection for fair governance