Citizen AI – Intelligent Citizen Engagement Platform

Project Title: Citizen AI – Intelligent Citizen Engagement Platform

- **Team ID:** LTVIP2025TMID21064
- **Team Size:** 4 members
- **Team Leader:** Batchula Sri Teja Naga Sai
- **Team Members:** Anagani Yogananda, Alluri Pranathi Sai, Annavarapu Phaneedra

INTRODUCTION:

1.1 Project Overview

- Citizen AI is a generative AI–based citizen engagement platform built using Hugging Face's IBM Granite LLM, FastAPI, and cloud-based technologies. The platform aims to enhance civic participation and service delivery by providing:
- Real-time civic query responses
- Grievance redressal through smart assistant
- Sentiment analysis for public mood tracking
- Dynamic feedback dashboard
- Personalized civic suggestions and AI insights

1.2 Purpose

- The platform enables:
- AI-powered responses to user queries on government services and policies
- Reporting and tracking civic complaints
- Real-time monitoring of citizen sentiment
- Display of community-wide analytics for policymakers
- Delivery of personalized suggestions and feedback loops

2. IDEATION PHASE:

2.1 Problem Statement

- Government agencies struggle with slow grievance redressal, poor data integration, and lack of intelligent interfaces to support citizens. Current systems are reactive, fragmented, and don't provide personalized civic services or public sentiment insight. A centralized, AI-powered citizen assistant can address these challenges.

2.2 Empathy Map Canvas

- Users: Citizens, Local Municipal Staff, Civic Planners
- Needs: Smart complaint system, real-time feedback, local updates
- Pain Points: Delay in issue resolution, low system trust, no civic transparency

2.3 Brainstorming (with UI-Based Features)

- Civic Query Assistant (Chat):
- Citizens can ask queries on city rules or services
- Uses IBM Granite LLM via Hugging Face API for answers
- Chat history saved for future use
- Complaint Registration System:
- Text-based input form for raising complaints
- Auto-tagging of category (e.g., roads, water, waste)
- Complaint ID and status tracking available
- Sentiment Analyzer:
- Analyzes public feedback using sentiment analysis
- Displays trends (positive vs negative opinions)
- Helps administrators understand citizen satisfaction
- Real-time Dashboard:
- Displays complaint statistics, sentiment charts, and user feedback
- Updated every 15 seconds using Chart.js
- Offers actionable insights for policy planning
- Contextual Suggestions:
- Based on user input or location, provides eco-tips or civic guidance
- Example: "You can raise a complaint with the local council via this link"
- PDF Policy Reader (Future Scope):
- Upload policy documents to extract summaries (planned feature)

3. REQUIREMENT ANALYSIS:

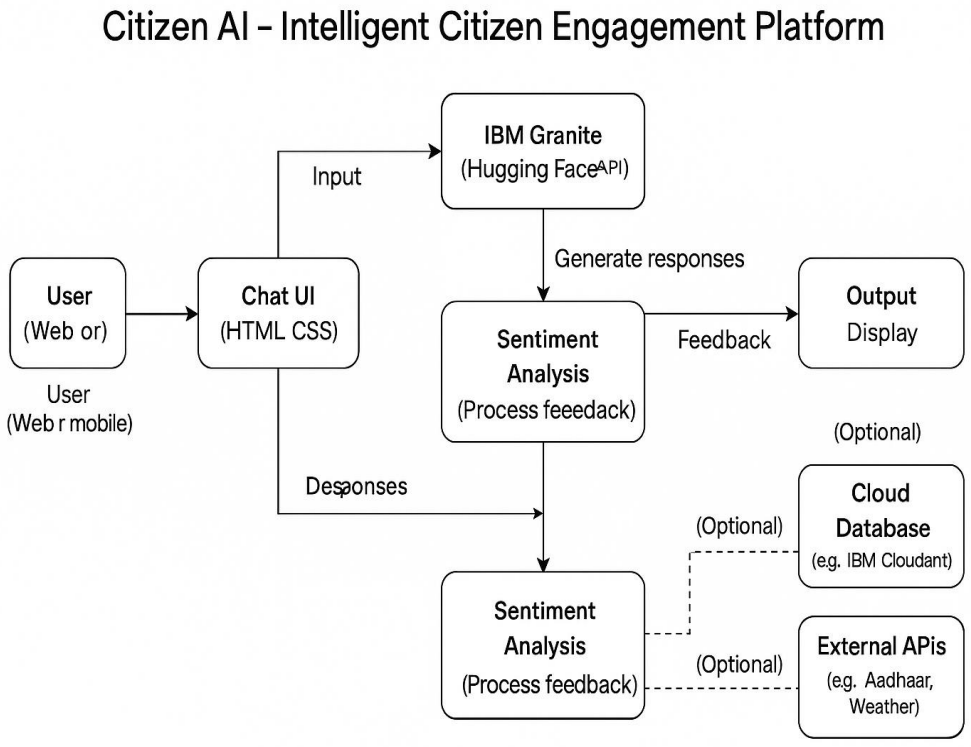
3.1Customer Journey:

- Enters a complaint or question
- AI model (IBM Granite) processes and returns the output
- Dashboard updates with sentiment/feedback
- Data is stored for analytics
- User visits the web portal

3.2 Requirements

- FastAPI for backend services
- Hugging Face integration with IBM Granite model
- Google Colab for model testing and analysis
- Frontend: HTML, CSS, JS, Chart.js
- JSON/CSV for local storage

3.2Data Flow Diagram



3.3Tech Stack Layer Tools

Layer	Tools
Frontend	HTML, CSS, JS, Chart.js
Backend	FastAPI
AI Model	Hugging Face (IBM Granite 3.3-2B)
Sentiment Engine	Python + Transformers
Storage	JSON or NoSQL

4. PROJECT DESIGN

4.1 Problem–Solution Fit:

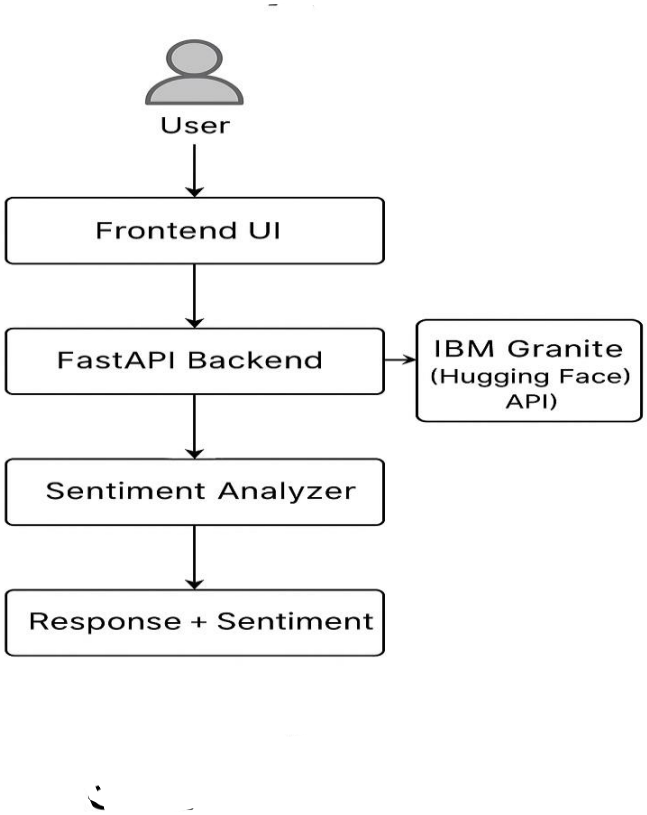
- Citizen AI solves civic engagement issues by:
- Providing real-time chatbot assistance
- Automating complaint logging
- Analyzing user sentiment and feedback
- Visualizing data on a live dashboard

4.2 Proposed Solutions:

- Chat Assistant (AskMe AI)
- Complaint Registration & Auto-Categorization
- Sentiment Analysis
- Real-Time Analytics Dashboard
- User Feedback Loop
- Contextual Tips

4.3 Architecture:

- All interactions handled through HTML/JS frontend
- FastAPI backend integrates with Hugging Face model via API
- Google Colab used for prompt testing and model inference
- Live data fed to frontend dashboard



4.4 Key Configuration Files

.env

HUGGINGFACE_API_KEY=your_key

HUGGINGFACE_MODEL_ID=ibm-granite/granite-3.3-2b-instruct

FRONTEND_PORT=8501

API_HOST=127.0.0.1

DEBUG=Tr

5. PROJECT PLANNING & SCHEDULING PHASE:

- Followed a Sprint-based Agile workflow for flexible iteration and progress tracking
- Conducted daily sync meetings to coordinate team efforts and resolve blockers
- Adopted Gantt-style planning to visualize phases, timelines, and dependencies

Phase-wise Tasks

- Ideation (2 days): Define features, user needs, system flow
- Backend Dev (5 days): Build FastAPI endpoints for chat, complaints, sentiment
- LLM Setup (2 days): Integrate IBM Granite via Hugging Face using Google Colab
- Frontend UI (4 days): Create HTML/CSS layout, dashboard with Chart.js
- Testing (2 days): Validate API, model, and dashboard performance
- Documentation (1 day): Prepare architecture, reports, and diagrams
- Deployment (1 day): Run backend on local/Colab environment
- Buffer (1 day): Final fixes and improvements

Roles Involved

- FastAPI Developer: Backend API development and routing
- LLM Engineer: Hugging Face model integration and prompt tuning
- Frontend Developer: UI design using HTML, CSS, and JavaScript
- Tester: Test APIs, model output, and dashboard updates
- Project Lead: Coordinates team, tasks, and documentation

6. FUNCTIONAL & PERFORMANCE TESTING

- Civic query speed tested for latency
- Complaint categorization auto-tagging accuracy
- Dashboard updates tested with live JSON
- Sentiment analyzer checked with mock feedback
- AskMe AI tested for contextual responses
- Multi-module interaction tested concurrently
- Error resilience validated with invalid inputs

7.RESULTS

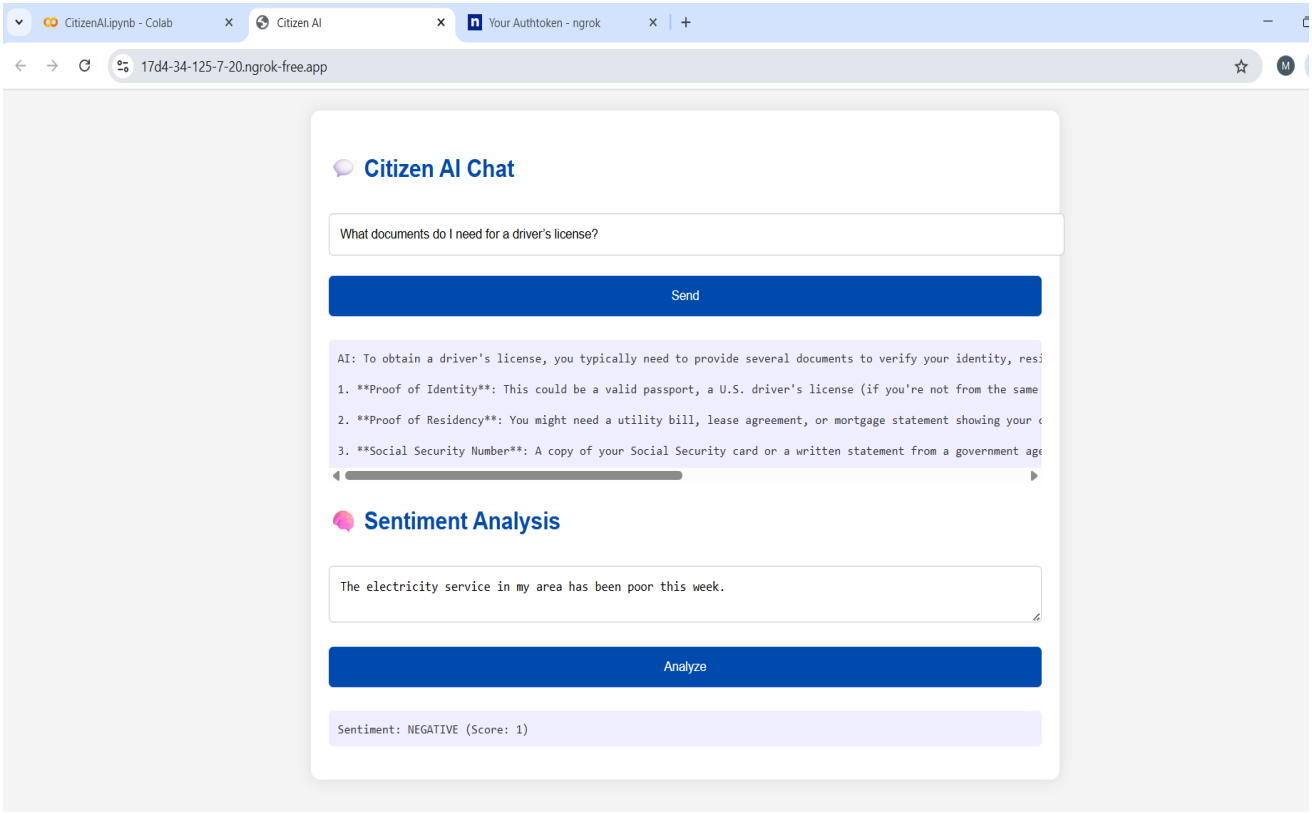


Fig: DashBoard

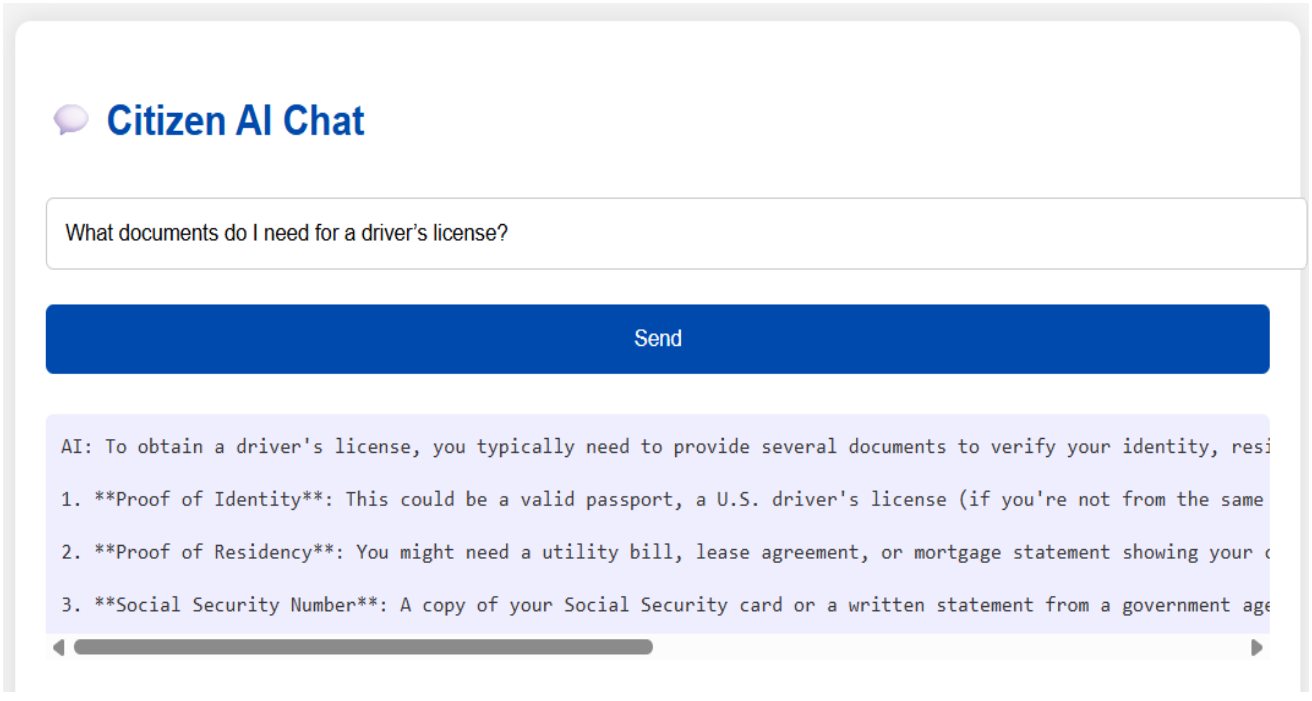


Fig:Chat Assistant

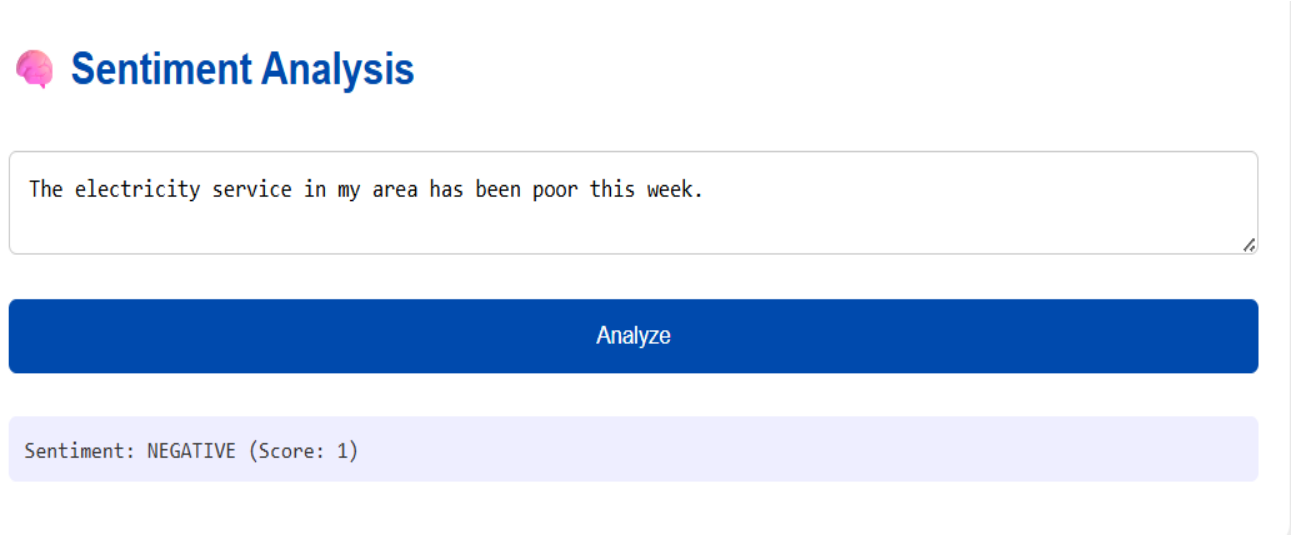


Fig:Citizen Feedback

8. ADVANTAGES & DISADVANTAGES

Advantages

- Personalized AI-driven support
- Simple UI for complaint and query
- Real-time analytics and sentiment tracking
- Works with open-source APIs

Disadvantages

- Requires stable internet access
- Limited offline functionality
- Hugging Face model integration needs API cost consideration

---

9. CONCLUSION

Citizen AI bridges the gap between the government and citizens by offering intelligent, fast, and responsive civic engagement features. It ensures transparency, simplifies complaint logging, and empowers the public through an AI-powered assistant.

---

10. FUTURE SCOPE

- Voice-based assistant
- Mobile app for field usage
- Multilingual support
- IoT integration with real-time sensors
- GIS-based complaint heatmaps
- Smart alert system (SMS/email)
- Admin analytics and exportable reports
- e-Governance API integration

11. APPENDIX:

- **GitHub:** https://github.com/Pranathi-Sai-Alluri/Citizen-AI
- **Demo Link:** *https://drive.google.com/file/d/1pdjzdwuWuIndoIWtHoXMVLJHUQ078GR-/view?usp=sharing*