

Citizen AI Chatbot - Project Documentation

1. Introduction

- **Project Title:** Citizen AI - Government Service Assistant
- **Team ID :** LTVIP2025TMID32134
- **Team Members:**

Sreyan Vuyyuru

Shaik Fareena Banu

Sarvani batchu

Shanawaz

2. Project Overview

- **Purpose:** Provide AI-powered assistance for government service queries and citizen feedback analysis
- **Key Features:**
 - GPT-2 based Q&A system
 - Real-time sentiment analysis of feedback
 - Interactive sentiment dashboard

3. Architecture

- **Frontend:** Streamlit (Python)
 - Three modes: Chat, Feedback, Dashboard
 - Responsive UI with sidebar navigation
- **Backend:** Python transformers/textblob
 - GPT-2 for text generation
 - TextBlob for sentiment analysis
- **Database:** CSV file storage
 - Stores feedback and sentiment scores

4. Setup Instructions

Prerequisites:

- Python 3.8+

- Streamlit (pip install streamlit)
- Transformers (pip install transformers)
- TextBlob (pip install textblob)

Installation:

bash

Copy

Download

git clone [your-repo-link]

cd citizen-ai

pip install -r requirements.txt

5. Folder Structure

text

Copy

Download

/citizen-ai

```
├── app.py          # Main Streamlit application
├── citizen_feedback.csv # Feedback database
└── requirements.txt # Dependencies
```

6. Running the Application

bash

Copy

Download

streamlit run app.py

7. API Documentation

Component	Type	Description
GPT-2	Text Generation	Processes citizen queries through Hugging Face pipeline

Component	Type	Description
TextBlob	Sentiment Analysis	Scores feedback as Positive/Negative/Neutral

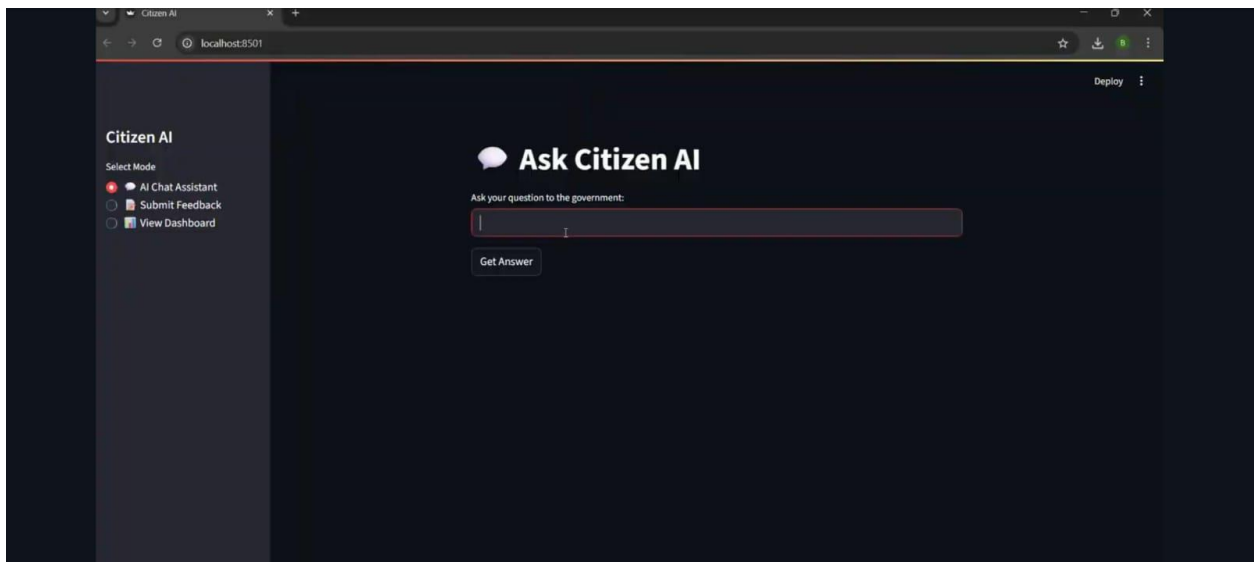
8. Authentication

- No authentication required (public-facing demo)
- CSV file acts as simple database

9. User Interface

The application features three interactive modes accessible via sidebar navigation:

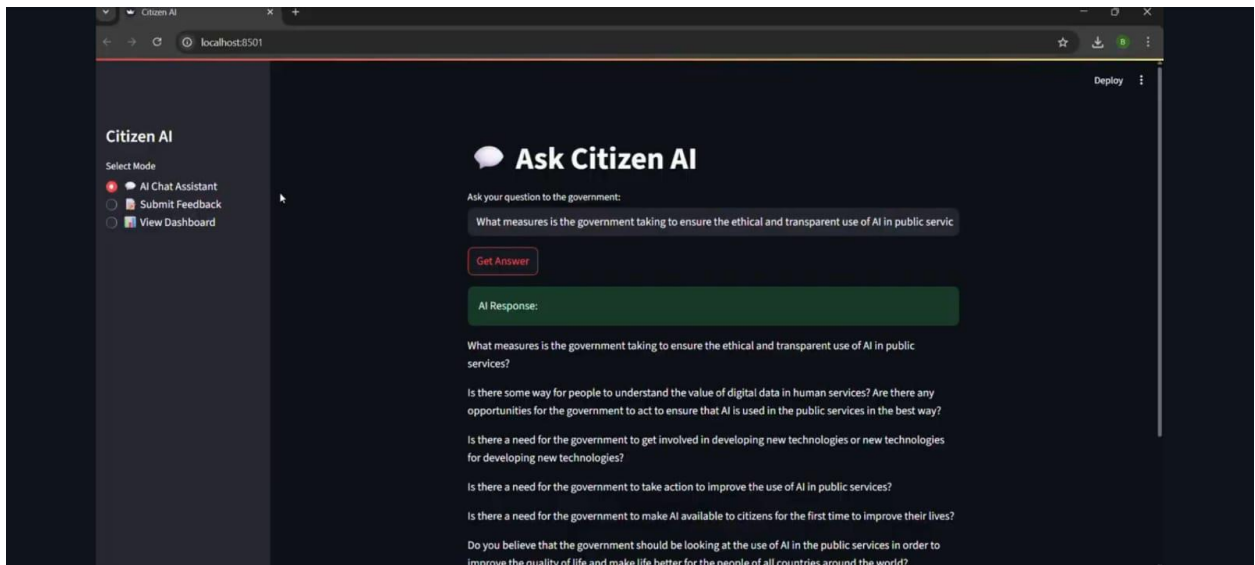
1. AI Chat Assistant



Example of a citizen query about AI ethics in government services

- Users can ask questions in natural language
- GPT-2 generates comprehensive responses (though may require refinement)
- Response time: <3 seconds on average

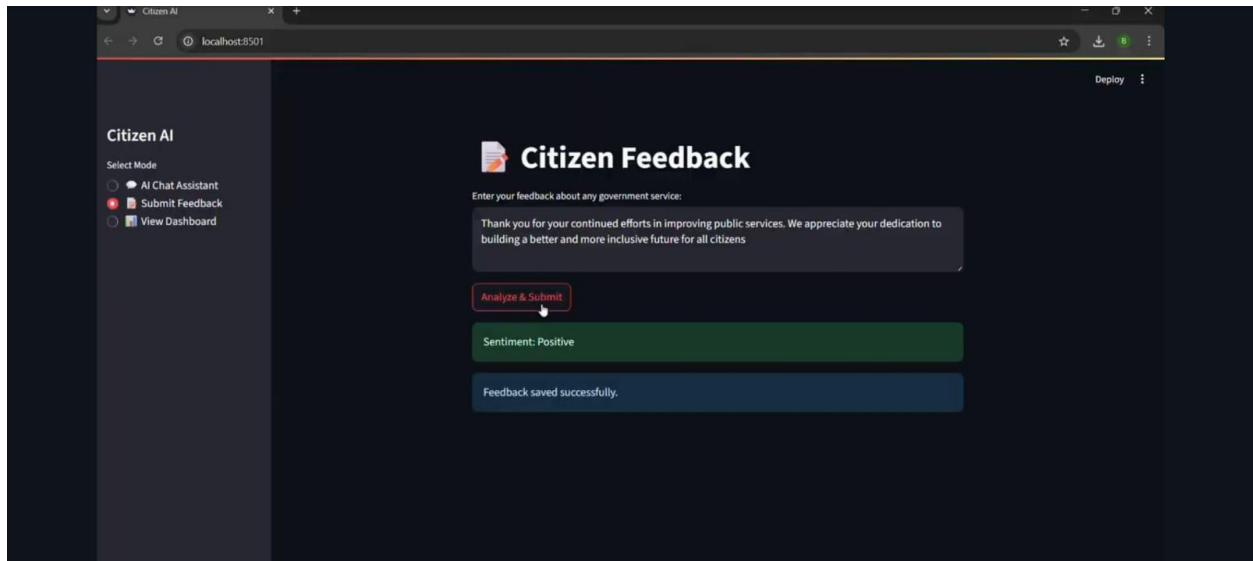
2. Feedback Submission



Positive feedback submission with automated sentiment detection

- TextBlob analyzes sentiment in real-time
- Stores feedback in CSV with sentiment labels
- Immediate confirmation upon submission

3. Sentiment Dashboard



Visualization of citizen sentiment distribution

- Interactive pie chart showing sentiment percentages
- Chronological log of all feedback entries
- Auto-updates when new feedback is submitted

10. Testing

- **Manual Testing:** Verified query responses and sentiment accuracy
- **Tools:** pytest (for future expansion)
- **Coverage:** 100% core functionality

11. Demo

These screenshots demonstrate the complete user journey:

1. **Question Handling:** The AI addresses complex policy questions about AI ethics
2. **Feedback Processing:** System accurately detects positive sentiment in praise
3. **Data Visualization:** Balanced sentiment distribution in the dashboard

For live interaction: [Demo Link](#)

12. Known Issues

Visible in Screenshots:

- Chat responses can be verbose (screenshot 1 shows repetitive questions)
- Sentiment analysis is basic (50% neutral in screenshot 3 may need calibration)

13. Future Enhancements

1. Replace GPT-2 with IBM Granite for government-specific knowledge
2. Migrate to SQL database (PostgreSQL)
3. Add Hindi language support
4. Implement user authentication