Generative AI with IBM Cloud

Project Documentation format

---

1. Introduction

- **Project Title**: *Citizen AI – Intelligent Citizen Engagement Platform*

- **Team Members**:

  - Shaik Mohammad Rida Ahalam

  - Chadalavada Hima Sai Saranya

  - Tenali Ravindhra

  - Chanda Rohith Babu

- Team ID: LTVIP2025TMID60447

---

2. Project Overview

- **Purpose**:
  The CitizenAI project is designed to empower citizens to express their feedback, concerns, and sentiments about civic issues through a conversational AI platform. It leverages natural language understanding to classify and store feedback for analysis by authorities.

- **Features**:

  - Chat interface with IBM Granite AI model integration

  - Sentiment analysis (positive, negative, neutral)

  - Feedback classification (suggestion, complaint, inquiry, etc.)

  - Real-time dashboard for analytics

  - Admin panel to manage submissions

---

3. Architecture

- **Frontend**:
  Built using React.js, with components for chat, dashboard, login/signup, and user feedback forms. Styled using Tailwind CSS and uses Axios for API communication.

- **Backend**:
Developed using Node.js and Express.js. It exposes REST APIs for user authentication, AI inference, and feedback storage.

- **Database**:
MongoDB is used to store user data, chat history, sentiments, and categorized feedback. Mongoose is used for schema modeling.

---

4. Setup Instructions

- **Prerequisites**:

  - Node.js v18+

  - MongoDB (local or Atlas)

  - Python 3.11 (for IBM Granite integration)

  - IBM Granite model setup

  - Flask (if AI model is served separately)

- **Installation**:

- git clone https://github.com/your-repo/citizenai.git

- cd citizenai

- npm install

- cd client

- npm install

- cd ..

- npm run dev

Set up a .env file with:

MONGO_URI=your_mongo_uri

JWT_SECRET=your_jwt_secret

---

5. Folder Structure

- **Client**:

- /client

- /src

- /components

- /pages

- /services

- App.js

- index.js

- **Server**:

- /server

- /routes

- /models

- /controllers

- /middleware

- index.js

---

6. Running the Application

- **Frontend**:

- cd client

- npm start

- **Backend**:

- cd server

- npm start

---

7. API Documentation

- **Endpoints**:

  - POST /api/auth/login – User login

  - POST /api/auth/register – User signup

  - POST /api/chat – Send prompt to AI and receive response

  - GET /api/dashboard – Fetch analytics for admin

   o POST /api/feedback – Submit categorized feedback

- **Parameters & Responses**:
  Include JSON structure for request body and sample responses in Swagger or Postman documentation.

---

8. Authentication

- JWT-based authentication

- Login credentials generate a token stored in localStorage

- Middleware protects private routes

- Roles (user/admin) are stored in user schema

---

9. User Interface

- Built with React and Tailwind

- Responsive design for mobile and desktop

- Chat UI mimics a conversational interface

- Dashboard includes charts, recent feedback, and sentiment analysis
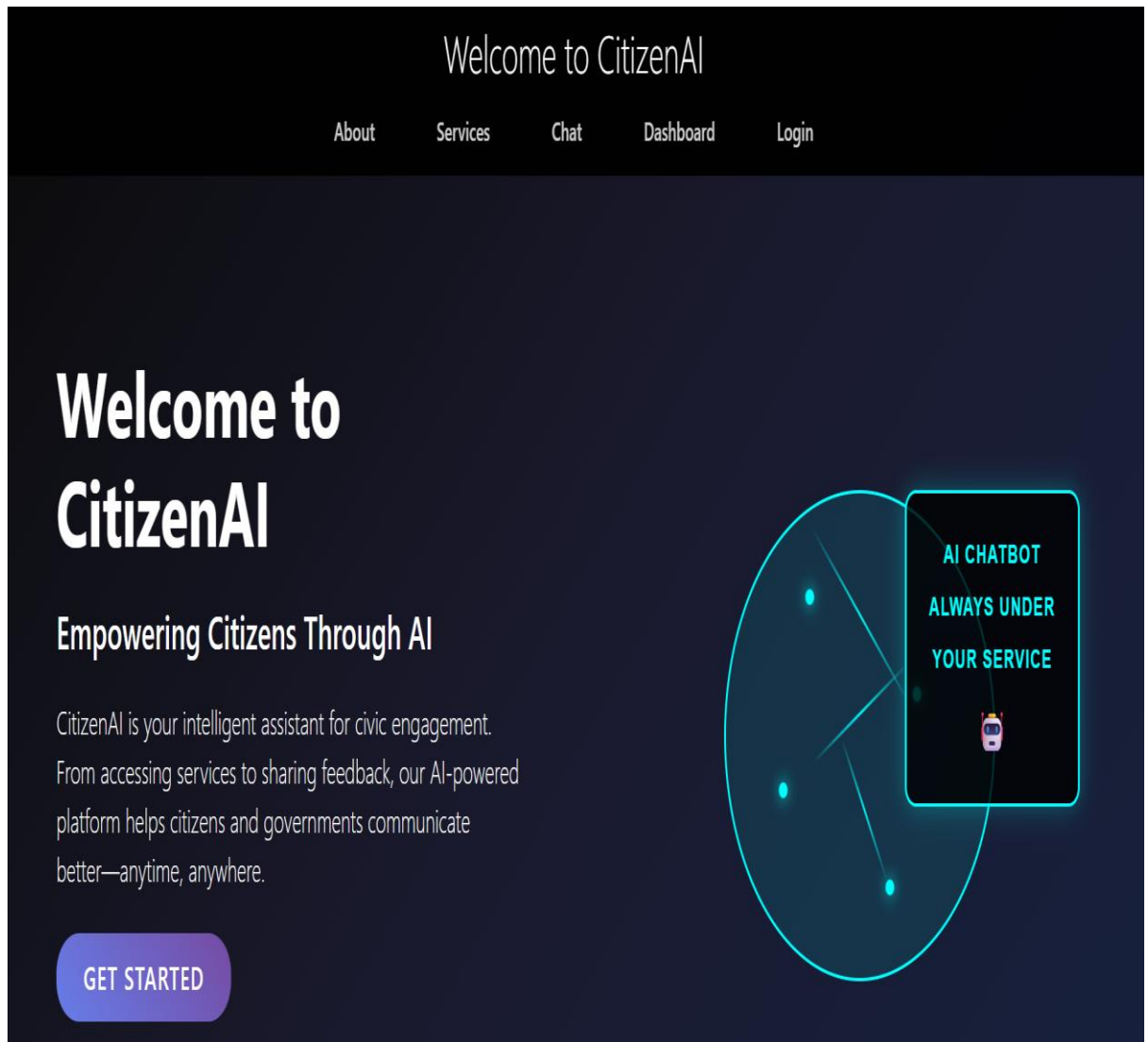
- Login and signup pages with validation

---

10. Testing

- **Tools**:

   o Jest for unit tests

   o Postman for API testing

   o React Testing Library for frontend components

- **Strategy**:

   o Test critical paths like authentication, feedback submission, and dashboard updates.

---

11. Screenshots or Demo

- Add screenshots of:

- o Chat interface

- o Admin dashboard

- o Sentiment report

- o

Welcome to CitizenAI

🤖 AI Chat Assistant
Powered by IBM Granite AI Model - Secure Session Active

🤖

AI Assistant:

Hello admin! I'm your AI assistant for government services. How can I help you today?

Just now

You:

How do I apply for a ration card?

08:47 AM

👤

🤖

AI Assistant:

For ration card applications, you need: 1) Address proof 2) Identity proof 3) Income certificate. You can apply online at your state's food department portal.

08:47 AM

Ask about government services...   ➤

💡 Try asking about: ration cards, pension schemes, licenses, permits, tax filing

-

💬 **Submit Feedback**

Your Feedback

Share your experience with government services...

Our AI will analyze the sentiment of your feedback.

✈ **Submit Feedback**

Quick examples:

Positive | Neutral | Negative

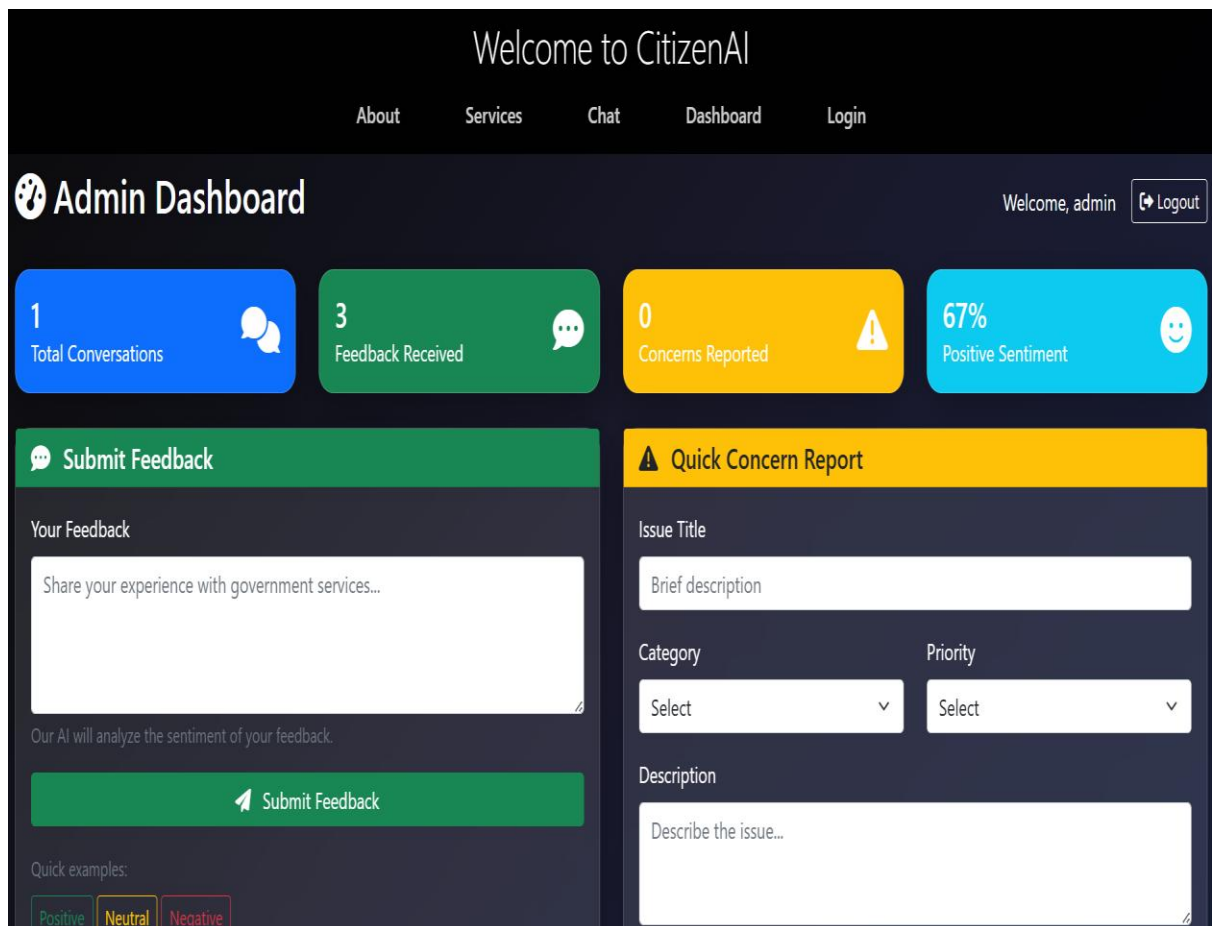**Recent Feedback**

Feedback	Sentiment	Date
for getting driving license it is taking more time...	Neutral	2025-06-30

---

12. Known Issues

- IBM Granite model latency under heavy load

- Limited language support in current version

- Mobile responsiveness can be improved

---

13. Future Enhancements

- Add speech-to-text input for voice-based feedback

- Multi-language support

- Role-based dashboard views (citizen/admin/officer)

- Notification system for new feedback alerts

- Offline storage and sync