Citizen AI – Intelligent Citizen Engagement Platform

Project Documentation

1.INTRODUCTION

. PROJECT title : Citizen AI – Intelligent Citizen Engagement Platform

. Team Member : SANTHIYA A

. Team Member : SANGEETHA A

. Team Member : SANGEETHA R

. Team Member : ROOPASRI M

2.Project overview

purpose :

Citizen AI is designed to strengthen the relationship between government bodies and citizens by using Artificial Intelligence for transparent, efficient, and real-time engagement. The platform helps citizens access policies, provide feedback, get eco-friendly lifestyle tips, and view government resource usage forecasts.

Features :

1. Policy Summarization – AI-powered summarization of government policies for easy understanding.

2. Resource Forecasting – Predictive analytics to forecast usage of public resources.

3. Eco-Tip Generator – Daily personalized environmental tips for citizens.

4. Citizen Feedback Loop – Mechanism to collect, analyze, and act on citizen feedback.

5. KPI Forecasting – Monitoring and predicting key performance indicators for governance.

6. Secure Authentication – Role-based access for citizens, admins, and policymakers.

7. User-Friendly Interface – Mobile-first, multilingual support.

3. Architecture

Frontend: React / Next.js (Responsive UI)

Backend: Node.js / Express

Database: MongoDB / PostgreSQL

AI Models: NLP for policy summarization, ML models for forecasting

Hosting: Cloud-based (AWS / Azure / GCP)

APIs: RESTful APIs with JSON responses

Authentication: JWT-based authentication and OAuth2 for third-party integrations

4. Setup Instructions

1. Clone the repository:

git clone : <https://github.com/santhiya40/ai1>

2. Install dependencies:

npm install

3. Set up environment variables in .env file:

DB\_URI=<your\_database\_url>

JWT\_SECRET=<your\_secret\_key>

AI\_API\_KEY=<your\_ai\_service\_key>

4. Run database migratinpm run migrate

5. Start the development server:

Npm run dev

5. Folder Structure

Citizen-AI/

├── backend/              # Backend (Node.js, Express)

│   ├── models/           # Database models

│   ├── routes/           # API routes

│   ├── controllers/      # Business logic

│   ├── middleware/       # Authentication, logging, etc.

│   └── utils/            # Helper functions

├── frontend/             # Frontend (React/Next.js)

│   ├── components/       # Reusable UI components

│   ├── pages/            # Main pages

│   ├── services/         # API calls

│   └── assets/           # Images, icons

├── docs/                 # Documentation

├── tests/                # Testing files

├── .env                  # Environment variables

└── package.json

---

6. Running the Application

Development Mode:

npm run dev

Production Build:

npm run build

npm start

---

7. API Documentation

Example: Get Policy Summary

Endpoint: /api/policy/summary/:id

Method: GET

Response:

{

  "policy\_id": "12345",

  "title": "Clean Energy Policy",

  "summary": "This policy aims to promote renewable energy adoption by..."

}

Example: Submit Feedback

Endpoint: /api/feedback

Method: POST

Request Body:

{

  "user\_id": "u789",

  "feedback": "The waste management system needs improvement."

}

Response:

{

  "status": "success",

  "message": "Feedback submitted successfully."

}

---

8. Authentication

JWT-based authentication for secure user sessions.

Role-based access control (Admin, Citizen, Policy Maker).

OAuth2 integration for social logins (Google, Facebook).

---

9. User Interface

Dashboard: Personalized insights for citizens.

Policy Center: Browse, read, and get summaries of policies.

Eco-Tips: Daily AI-generated environmental tips.

Feedback Portal: Submit and track feedback.

Analytics View: Charts and forecasts for transparency.

---

10. Testing

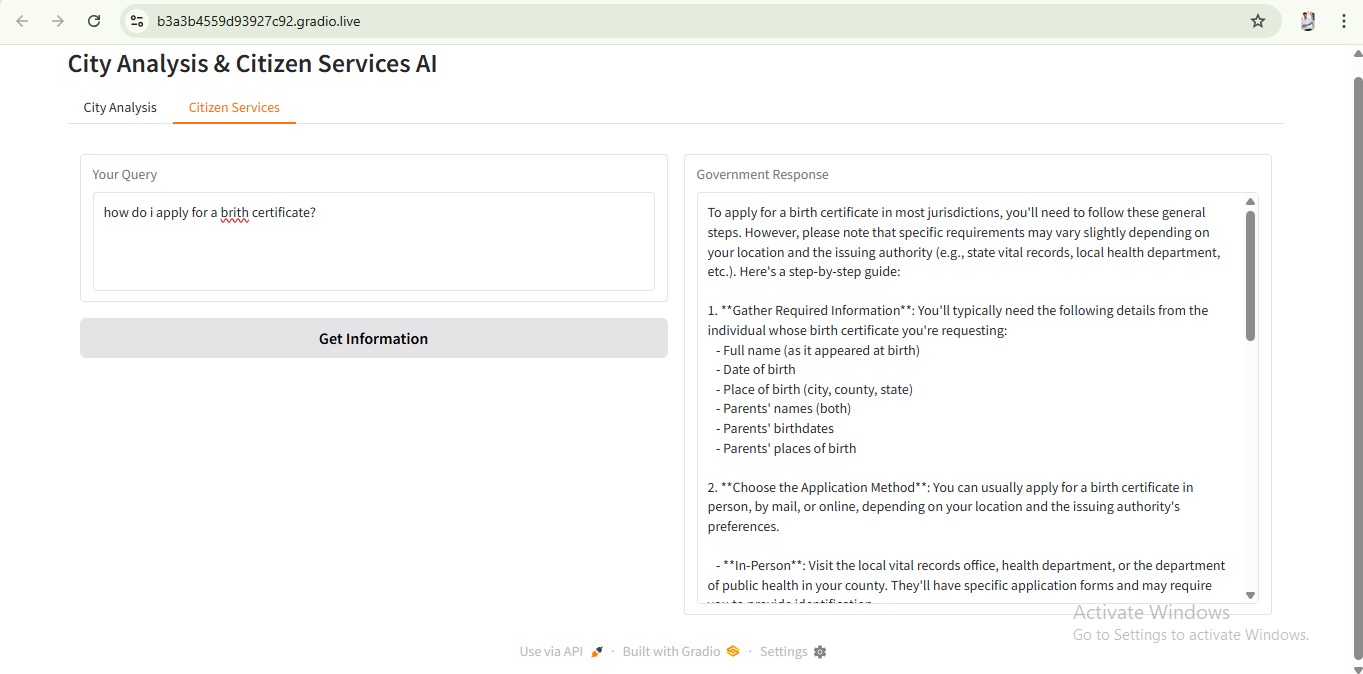
Unit Tests: Jest / Mocha for backend functions.

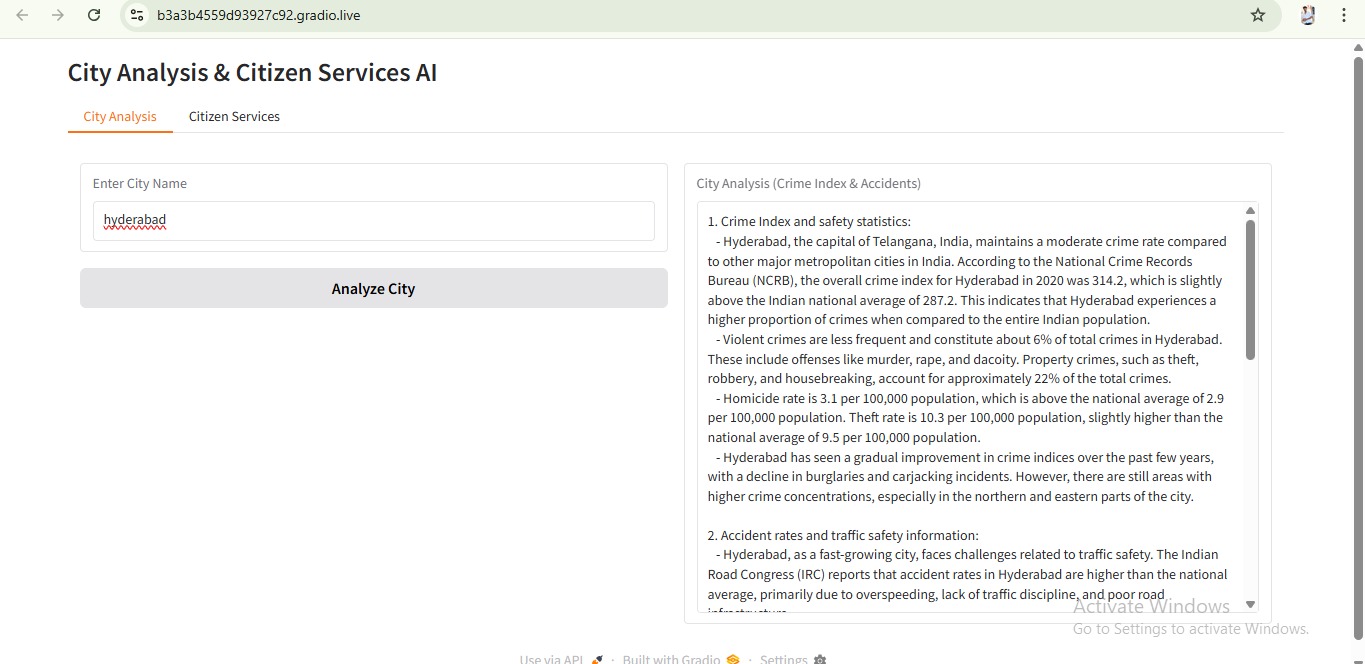
Integration Tests: Supertest for API testing.

UI Testing: Cypress for end-to-end testing.

CI/CD: Automated testing using GitHub Actions.

11. Screen shots :



 12.Conclusion :

Citizen AI – Intelligent Citizen Engagement Platform serves as a transformative solution to bridge the gap between citizens and governing bodies through intelligent, data-driven engagement. By integrating features such as policy summarization, resource forecasting, eco-tip generation, KPI prediction, and citizen feedback loops, the platform empowers both decision-makers and communities to collaborate more effectively.