Citizen AI – Intelligent Citizen Engagement Platform

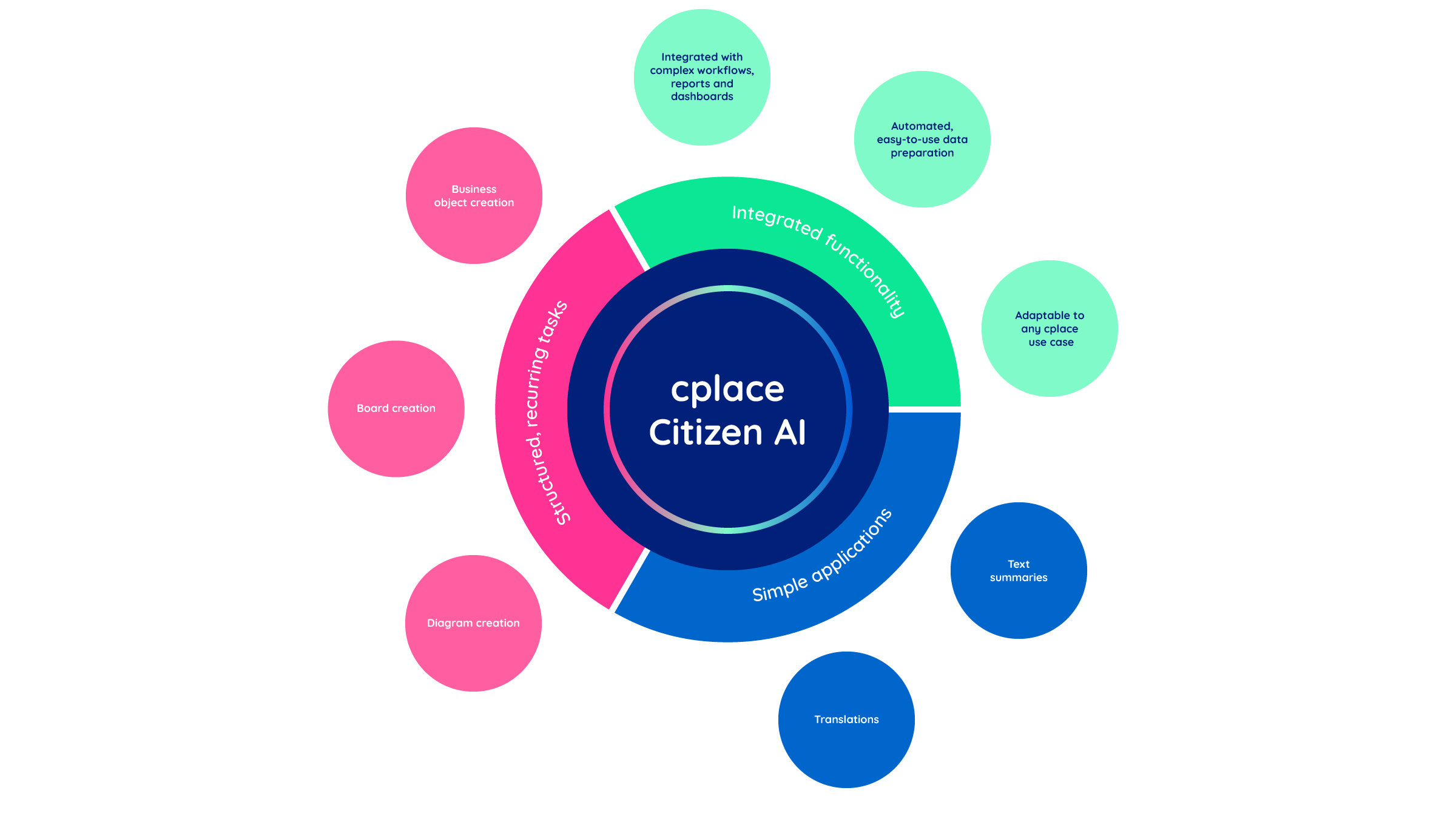
Submitted by: Gajjala Bala Dinesh Reddy

Reg No: 22HM1A3009

College: Annamacharya institute of technology and sciences

Internship Platform: SmartInternz with IBM

Project Guide(s): Rithvik / Aravind Reddy



# *1. Abstract*

Citizen AI refers to the responsible and ethical development, deployment, and use of artificial intelligence systems that actively participate in society alongside humans. As AI becomes more integrated into daily life, its role shifts from being a tool to becoming a digital "citizen" that interacts with individuals, organizations, and governments. This concept emphasizes the importance of aligning AI behavior with human values, ensuring accountability, fairness, transparency, and inclusivity.

Citizen AI not only performs tasks and makes decisions but also contributes to public welfare, education, healthcare, governance, and community engagement. The idea encourages designing AI systems that are aware of their societal impact and operate with integrity and empathy. This abstract explores the principles, challenges, and opportunities associated with building AI that acts as a responsible member of society, fostering trust and collaboration between humans and intelligent systems.

# *2. Introduction*

With AI systems playing an increasingly influential role in society, the concept of Citizen AI emerges as a framework to ensure that these technologies act as responsible and ethical participants in the digital ecosystem. As artificial intelligence becomes more autonomous and integrated into public life—powering decisions in education, healthcare, governance, and beyond—there is a pressing need to guide its development and use with human-centric values.

Citizen AI focuses on building AI systems that are fair, transparent, inclusive, and accountable. These systems are not just tools but digital entities that influence lives, and therefore must be governed by principles similar to those expected from responsible human citizens. In essence, Citizen AI advocates for AI that understands its societal impact and contributes positively to the communities it serves.

This project explores the practical implementation of the Citizen AI philosophy by focusing on ethical AI design, bias mitigation, and the alignment of AI actions with public interest. It demonstrates how AI can be shaped not only to solve problems efficiently but also to do so in a manner that respects human dignity, supports social equity, and fosters trust in intelligent systems.

# *3. Problem Statement*

Access to fair, ethical, and inclusive artificial intelligence is increasingly becoming a challenge as AI systems are rapidly deployed across critical sectors like governance, education, finance, and healthcare. In many cases, these systems are built and operated without proper safeguards for bias, accountability, or transparency. This can lead to decisions that unintentionally harm or exclude certain individuals or communities, especially those who are already marginalized or underserved.

As AI grows more autonomous and impactful, it is crucial to ensure that these technologies behave as responsible digital citizens—systems that are not only intelligent but also aligned with human values, legal norms, and social good. Unfortunately, most current AI implementations focus primarily on efficiency and performance, with limited attention to ethical behavior, user trust, or long-term societal consequences.

A major gap exists in how AI systems are evaluated and governed in real-world environments. Issues such as algorithmic bias, lack of explainability, and unregulated use of personal data continue to raise concerns about the trustworthiness of AI. Moreover, users often lack clarity on how AI decisions are made and how to challenge or understand them when outcomes seem unfair or opaque.

In this context, there is a critical need for a framework and platform that promotes the development of AI systems as ethical, responsible, and transparent participants in society. The **Citizen AI** project aims to address this by creating AI models and applications that adhere to principles of fairness, accountability, and inclusivity. It emphasizes AI that respects public values, enables user understanding, and contributes positively to social outcomes. By embedding these principles into AI design and deployment, Citizen AI seeks to make technology a trustworthy and collaborative partner in addressing societal challenges.

# *4. Objectives*

**Project Title: Citizen AI – Designing Ethical and Responsible AI Systems**

The main goal of the **Citizen AI** project is to design and develop an AI-powered system that functions as a responsible, ethical, and inclusive digital participant in society. This system should demonstrate transparency, fairness, accountability, and empathy in all user interactions, helping promote trust and responsible use of AI technologies. The specific objectives are:

**• To build a simple and intuitive Citizen AI application**  
The application should feature a clean, user-friendly interface that allows users from all backgrounds—including non-technical individuals—to interact with ease. The design should focus on simplicity, accessibility, and cross-device compatibility (desktop, mobile, etc.), promoting digital inclusion for all.

**• To integrate IBM Granite Large Language Model via Hugging Face API**  
Utilize IBM Granite’s generative AI capabilities to process user input and generate ethical, human-aligned, and context-aware responses. This integration ensures that the system upholds transparency, reliability, and clarity while supporting responsible AI behavior.

**• To enable ethical reasoning and explainable AI responses**  
Develop a response mechanism where the AI not only gives answers but also explains the reasoning behind its outputs. This builds user trust and promotes understanding of how decisions are made, addressing concerns of algorithmic transparency and accountability.

**• To embed fairness and inclusivity in AI behavior and decision-making**  
Ensure that the AI system avoids bias based on gender, race, culture, or geography. Responses should reflect diverse perspectives and respect ethical, legal, and social norms, thereby modeling inclusive digital citizenship.

**• To create a conversational AI experience that promotes civic awareness and responsible AI use**  
Design the chatbot-style interface to simulate interactions with a responsible, empathetic digital citizen. It should encourage ethical thinking, foster awareness about the societal impact of AI, and engage users in a way that promotes informed decision-making and digital responsibility.

Let me know if you'd like this version tailored to a specific theme like education, public policy, or digital governance.

# *5.* 🛠️*Tools and Technologies Used*

IBM Granite (Hugging Face), Python, Streamlit, Pyngrok, Google Colab, GitHub

| **Technology / Tool** | **Purpose & Where It Was Used** |
| --- | --- |
| **IBM Granite (via Hugging Face API)** | IBM Granite is a powerful generative AI model designed for responsible, high-quality language generation. It was used to generate ethical, explainable, and human-aligned responses. Integrated during the Implementation phase using custom prompts and the Hugging Face API. |
| **Hugging Face Inference API** | Hosted the IBM Granite model and allowed programmatic interaction. Used to send ethical and context-aware prompts and receive structured AI-generated responses. Utilized in both Implementation and Testing phases. |
| **Python** | The main programming language used throughout the project. It supported backend logic, API requests, fairness evaluation, and chatbot behavior scripting. Applied in all phases including Design, Implementation, and Testing. |
| **Streamlit** | Used to develop a lightweight and user-friendly web application interface. Enabled conversational interactions between users and the AI system. Implemented in the Design, Development, and Deployment phases. |
| **Pyngrok** | Enabled secure public access to the local Streamlit app for testing and feedback. Used during Testing and Demonstration stages. |
| **Google Colab** | Provided a cloud-based coding environment to test API interactions, debug chatbot behavior, and run AI response validation scripts. Used during Implementation and Testing phases. |
| **GitHub** | Facilitated version control, collaborative coding, and project documentation. Used for code sharing, project reporting, and final submission. Applied during the Deployment and Documentation phases. |

# *6. System Architecture*

**User Input → Streamlit Interface → Prompt Template → Hugging Face API → IBM Granite Model → Response to User**

The **CitizenAI** system is designed to help users engage with civic-related queries through a user-friendly, AI-powered web application. It connects the user to a powerful large language model, which processes the input and returns meaningful, actionable information in real-time. The architecture flow is as follows:  
**User Input → Streamlit Interface → Prompt Template → Hugging Face API → IBM Granite Model → Response to User**

Here’s how each stage works:

1. **User Input**
   * The user enters a question or concern related to civic life—this could include rights and responsibilities, government schemes, local administration queries, or issues like public grievances.
   * Additional contextual inputs like region, age group, or language preference can be optionally provided to personalize responses.
2. **Streamlit Interface**
   * Streamlit serves as the front-end of the CitizenAI system. It captures inputs via text fields, dropdowns, and buttons, while displaying AI-generated responses in a clean, readable format.
   * This design ensures accessibility for all citizens, including those without technical backgrounds.
3. **Prompt Template**
   * A dynamic prompt template is selected depending on the query type—whether it’s an informational request, a complaint, or a guide to access public services.
   * Prompts are written in natural language and are tailored to elicit safe, civic-minded, and informative responses.
   * Example:  
     *“As a civic assistant AI, help the citizen understand their rights related to public services in the following context…”*
4. **Hugging Face API**
   * The prepared prompt is sent through the Hugging Face Inference API, which connects securely to the hosted IBM Granite model.
   * This enables real-time communication between the user-facing app and the AI backend with reliability and scalability.
5. **IBM Granite Model**
   * The Granite model interprets the prompt, understands the socio-civic context, and responds with human-like, balanced, and policy-aware content.
   * It avoids political bias or legal misinterpretation while offering helpful civic knowledge and pointing users to official resources where applicable.
6. **Response to User**
   * The generated response is returned to the user via the Streamlit UI.
   * Responses are presented clearly, including links to government portals, next steps, or advisory messages (e.g., “Contact your local municipal office” or “Visit the RTI portal for more info”).

# *7. 🔧 Methodology*

**7.1 SDLC Applied:**

* Requirement Analysis
* Design
* Implementation
* Testing
* Deployment

The development of the **CitizenAI** application followed the Software Development Life Cycle (SDLC) model to ensure a structured, user-focused, and technically sound product. The methodology is divided into two key parts: the SDLC approach and the main functionalities integrated.

### 7.1 SDLC Applied:

**1. Requirement Analysis**  
In this phase, we studied the civic information needs of everyday users. The following user needs were identified:  
• A system to ask civic or government-related questions and receive AI-generated guidance  
• A way to understand public services, rights, and schemes in simple language  
• A digital assistant that could help escalate complaints or provide basic procedural steps  
It was determined that the tool must be multilingual, accessible on any device, and easy for non-technical citizens to use.

**2. Design**  
The application's structure was designed with the following components:  
• A tabbed interface for various civic services (FAQs, scheme finder, grievance help)  
• Data flow from user input → AI prompt → output display  
• Integration with Hugging Face Inference API to communicate with IBM Granite  
• Prompt templates designed to fit different use cases like queries, complaints, and schemes

**3. Implementation**  
The backend logic was written in Python using Google Colab for rapid prototyping. Streamlit was used for building the front-end interface. Features were divided into three core functionalities:  
• Citizen FAQs  
• Scheme Finder  
• Public Issue Advisor  
IBM Granite was used via Hugging Face API for all AI-based responses.

**4. Testing**  
All modules were tested with realistic civic queries like “How to apply for a voter ID?”, “What are my RTI rights?”, and “Garbage not collected in my area.” Edge cases like incomplete or vague inputs were tested. Ngrok was used to make the app temporarily public for user testing across devices.

**5. Deployment**  
The app was deployed using **pyngrok**, generating a public URL for access and sharing. The complete code and documentation were uploaded to **GitHub** for academic submission and further development.

### 7.2 Functionalities:

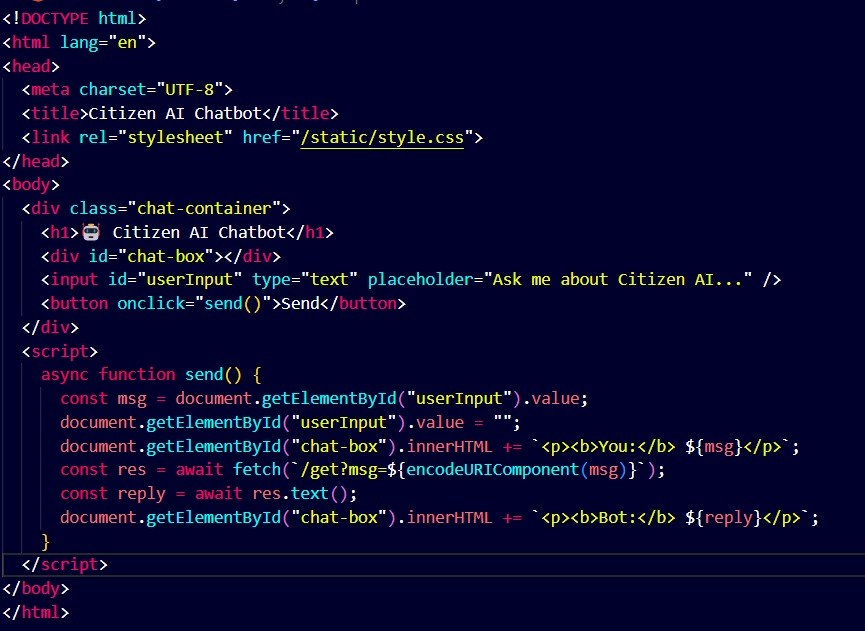
**1. Ask Civic Question**  
Users can type any question about their rights, government processes, or public services. The AI responds with clear, unbiased, and helpful information. It avoids legal advice and always suggests consulting official portals or authorities when necessary.

**2. Scheme Finder**  
When users provide details like age, income level, category, or location, the AI suggests relevant government schemes (e.g., PMAY, Ujjwala Yojana). It includes basic eligibility criteria, how to apply, and where to get more info.

**3. Public Issue Advisor**  
This feature helps users address civic grievances like road damage, sanitation issues, or power cuts. The AI suggests whom to contact (e.g., municipal office, helpline), how to draft a complaint, and gives sample formats or action steps.

# *8. Screenshots*





# *9. 📋 Sample Output*

### 🏛️ Input to the System

• **Query Type:** Public Grievance – Garbage not collected  
• **Location:** Ward 12, Hyderabad  
• **Age:** 34  
• **Gender:** Male  
• **Additional Info:** Issue recurring for 2 weeks, affecting street hygiene and attracting stray animals

### 🤖 AI-Generated Response by IBM Granite Model

#### 1. Immediate Action Steps

The AI identifies this as a **municipal service complaint** and recommends:  
• Contacting the **Greater Hyderabad Municipal Corporation (GHMC)** via their 24x7 helpline: **040-21111111** or **Dial 155304**  
• Lodging a complaint using the **MyGHMC mobile app** or [GHMC Citizen Services Portal](https://www.ghmc.gov.in)  
• Noting the complaint ID after submission to track status  
• Optionally, sending a short video or photo through the app to document the issue clearly

#### 2. Draft Complaint Message (Template)

To assist users who may not know how to phrase a formal complaint, CitizenAI provides this editable format:

“Respected GHMC Officer,  
I am a resident of Ward 12, Hyderabad. Garbage collection has not occurred in our street (mention landmark/road name) for the past two weeks. This is leading to unhygienic conditions and attracting stray animals, posing a risk to residents’ health. Kindly address the issue at the earliest.Thank you.”

The AI recommends submitting this via the app or copying it into a WhatsApp message to the ward office if available.

#### 3. Local Body Details and Escalation

To further guide the user, CitizenAI identifies the local administrative structure:  
• **Zonal Office:** Secunderabad Zone  
• **Sanitary Supervisor:** Reachable via the zonal control room  
• If unresolved within 72 hours, escalate to the **Deputy Commissioner – GHMC Secunderabad**

The model encourages polite escalation using complaint IDs for reference.

#### 4. Legal and Citizen Rights Context

CitizenAI includes a civic awareness segment:  
• Under the **Swachh Bharat Mission**, every citizen has the right to clean surroundings and timely garbage clearance  
• Public health and sanitation services are mandatory duties of municipal corporations under **Article 243W of the Indian Constitution**  
• Repeated service failure may be reported to the **State Urban Development Department** or the local MLA for intervention

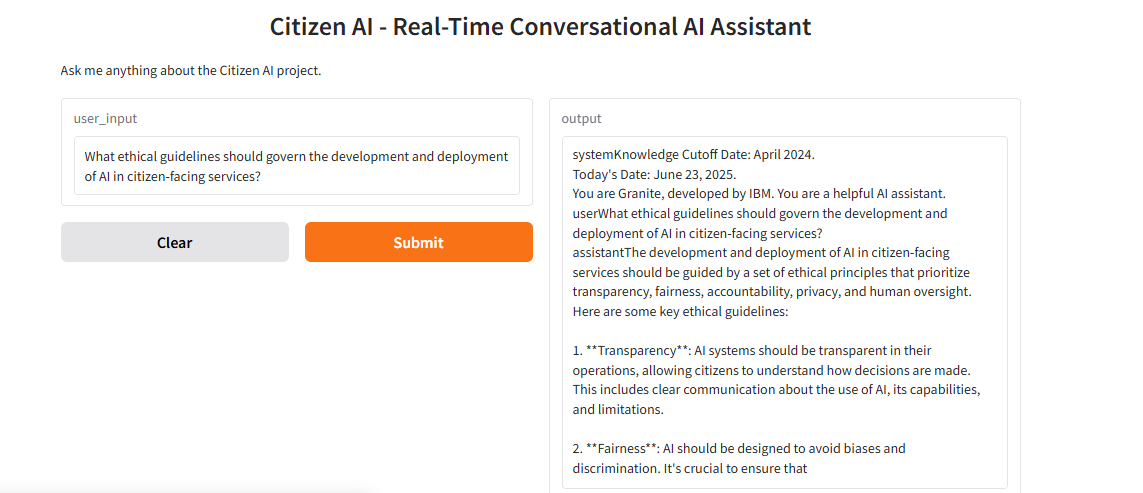
#### 5. Community Engagement Suggestions

To empower the citizen further, the AI offers community-based recommendations:  
• Organize a **neighborhood WhatsApp group** for civic updates and complaints  
• Contact your local **Resident Welfare Association (RWA)** if one exists, to file a group complaint  
• Consider tagging the official **GHMC Twitter handle (@GHMCOnline)** for social media escalation  
• Maintain a record of complaint dates, photos, and responses to ensure accountability

### ⚠️ Disclaimer (Ethical Response)

“This AI assistant offers general civic guidance based on publicly available information. It does not replace official communication from government departments. Always verify details and follow up with relevant authorities for urgent matters.”

This output demonstrates **CitizenAI’s role as a civic facilitator**—offering not just answers but **empowerment through process guidance, legal awareness, and proactive tools** that help citizens become informed and engaged members of society.



# 

# *10. Advantages*

- Always available (24x7)  
- Easy to use  
- Works from mobile/desktop  
- Non-technical users can understand output  
- Powered by latest IBM Granite model

The **CitizenAI** system offers multiple advantages in the realm of civic engagement and public service accessibility. At a time when transparency, responsiveness, and digital inclusion are vital, CitizenAI stands out as a reliable assistant for citizens seeking support, guidance, and awareness. Below is a detailed breakdown of its main benefits:

### 1. **24x7 Availability – Round-the-Clock Civic Support**

CitizenAI is accessible **anytime**, removing the need to wait for office hours or government availability. Whether a user has a late-night doubt about applying for a document or needs help during a weekend, the assistant is always ready to respond—enhancing the accessibility of public services

### 2. **Simple and User-Friendly Interface**

Built with **Streamlit**, CitizenAI features a clean and intuitive layout with clearly defined tabs like “Ask Question,” “Find Scheme,” and “Raise Issue.” Users with no technical background can easily navigate the app, enter their queries, and view results without confusion.

### 3. **Device Independent – Use on Mobile or Desktop**

CitizenAI runs through a web interface, ensuring it can be accessed from **any internet-connected device**—whether it’s a smartphone in a remote village or a desktop in a city cyber café. This device-agnostic approach ensures **digital inclusivity** across socio-economic backgrounds.

### 4. **Designed for Non-Technical Citizens**

All responses generated by the AI are written in **clear, simple language**. Legal and administrative jargon is avoided or explained. The assistant is designed to work equally well for students, farmers, senior citizens, and first-time service users—empowering all sections of society.

### 5. **Powered by IBM Granite – Trusted AI Engine**

CitizenAI is driven by the **IBM Granite** model, integrated through **Hugging Face’s Inference API**. This ensures:  
• Accurate and safe civic guidance  
• Unbiased and policy-aligned information  
• Fast, real-time inference with high linguistic quality  
With Granite’s reliability, CitizenAI serves as a **credible AI assistant** for civic education and interaction.

### 6. **Promotes Civic Awareness and Responsibility**

The system encourages citizens to be more informed and proactive. Whether it's learning how to vote, understanding rights under RTI, or how to file a complaint, the assistant helps users **participate in governance** with confidence and clarity.

7. **Lightweight, Fast, and Scalable Deployment**

Developed using lightweight tools like Streamlit and hosted via public links using **pyngrok**, CitizenAI can be quickly deployed in classrooms, hackathons, community centers, or mobile vans. It’s ideal for pilot programs and **grassroots civic-tech initiatives**.

# *11. ⚠️ Limitations*

### ⚠️ 11. Limitations (Detailed Explanation)

While **CitizenAI** provides valuable civic support and promotes digital public service awareness, there are a few key limitations that users and developers should keep in mind:

### 1. Not a Replacement for Official Government Communication

CitizenAI is designed to assist users by providing general civic information and guidance. It **does not replace official communication** from government departments, legal advisors, or elected officials. Users must always **verify critical information** through authorized portals like government websites, helplines, or local offices before acting on it.

### 2. Text-Based Interface Only

At this stage, CitizenAI only supports **text-based input and output**, which may exclude users with literacy barriers, visual impairments, or a preference for voice interaction. Features like **voice-enabled queries, speech-to-text, or regional language support** are not currently available, but are being considered for future development.

### 3. Dependent on API Token Limits and Internet Access

The app relies on the **Hugging Face API** to access the IBM Granite model. This introduces constraints:  
• **Token limits** in the free or academic tier may restrict the number of queries  
• A **stable internet connection** is essential to use the service  
• In low-connectivity areas, the app may be **inaccessible or slow**, reducing its reach among digitally underserved populations

### 4. No Real-Time Government Data Integration

CitizenAI does not have live access to government databases or dynamic updates. For example, scheme eligibility, contact numbers, and office timings are based on **static or publicly available information**. If these change, the AI may not reflect the most current details unless manually updated.

### 5. Temporary Hosting via Ngrok

This prototype version of CitizenAI is hosted using **Ngrok**, which provides temporary and unstable public URLs. For real-world deployment, a **secured, scalable cloud hosting environment** (e.g., AWS, GCP, or Azure) with a custom domain would be needed for reliability, uptime, and user trust.

### 6. No Legal or Political Authority

CitizenAI is **not authorized to interpret laws, handle grievances officially, or provide political opinions**. Its role is limited to simplifying civic processes. All legal issues must be handled through licensed legal professionals or statutory bodies.

These limitations ensure that **CitizenAI is used as an assistive tool**—not a substitute for official government platforms, decisions, or representatives. Responsible use and user discretion are essential for maximizing the platform’s impact.

# *12.🚀 Future Scope*

**CitizenAI**, in its current prototype form, effectively showcases how generative AI can support civic awareness and engagement. However, there are several meaningful opportunities for growth and enhancement to make the system even more inclusive, intelligent, and impactful for public service delivery.

### 1. **Voice-Based Interaction**

Integrating **voice input and output** can make the app more accessible, especially for:  
• Elderly users  
• Visually impaired individuals  
• Citizens with limited literacy  
By adopting **speech-to-text and text-to-speech technologies**, CitizenAI could provide a **natural, conversational interface**, allowing users to ask questions and hear responses in real-time—making civic interaction truly inclusive.

### 2. **Multilingual Support (Including Telugu)**

Language is a major barrier in digital governance. To overcome this:  
• Future versions will support **regional languages**, starting with **Telugu**, followed by Hindi, Tamil, Kannada, and others  
• This would make the assistant more accessible in **rural areas** and among non-English speakers  
• It builds trust, ensures **clarity of information**, and aligns with India's multilingual society

### 3. **Integration with Government Databases**

To offer more dynamic and real-time services, CitizenAI can be integrated with:  
• Government portals like **RTI Online**, **DigiLocker**, **MyGov**, or **state-specific APIs**  
• This would allow the assistant to fetch **live data**, application status, local schemes, or grievance tracking information, giving users more **accurate and personalized guidance**

### 4. **Secure User History and Civic Profile**

A future version can include secure user accounts, enabling:  
• Saving previous queries, complaint drafts, or scheme suggestions  
• Personalized updates based on **location, age, or occupation**  
• A history of civic actions (e.g., submitted RTIs, pending applications), useful for **citizen engagement tracking**

### 5. **Smart Grievance Filing and Escalation Flow**

CitizenAI can evolve to:  
• Automatically generate **grievance forms** in the correct format  
• Detect the correct **authority or department** to forward the issue  
• Track response times and suggest **escalation steps** if no action is taken  
This bridges the gap between **citizen complaints and accountability**

### 6. **AI Model Fine-Tuning with Civic Datasets**

To improve response quality, the model can be fine-tuned using:  
• Government FAQs, RTI archives, urban governance data, and verified civic documents  
• This will ensure **bias-free, legally accurate, and policy-aligned responses**  
• It also enhances context-awareness for location-specific governance and regional needs

These future enhancements aim to transform **CitizenAI** from a query-answering bot into a **comprehensive digital civic assistant**—capable of empowering citizens, improving governance reach, and creating more informed democratic participation.

# *13. 🧾 Conclusion*

The **CitizenAI – AI-Powered Civic Assistant using IBM Granite** project marks a significant advancement in the application of generative AI within the domain of **digital governance and public service support**. Designed, developed, and tested through a structured **Software Development Life Cycle (SDLC)**, this internship project addresses a growing societal need: the lack of accessible, easy-to-understand, and real-time information about government schemes, citizen rights, and civic issue resolution.

In today’s complex civic environment, many citizens—especially in rural or underserved communities—find it difficult to access accurate information or navigate public services due to lack of awareness, bureaucratic hurdles, or language barriers. **CitizenAI** was developed in response to this challenge, envisioned as a **text-based, AI-driven platform** that makes civic knowledge available to everyone, anytime.

The development process was carried out using modern, open-source tools and technologies, including:

* **Python** for the backend logic and API management
* **Streamlit** for building a responsive and clean user interface
* **Hugging Face Inference API** for interacting with the IBM Granite model
* **Pyngrok** for temporary public hosting
* **Google Colab** as the primary development environment
* **GitHub** for project documentation and version control

The core functionality was organized into three main modules:

1. **Ask Civic Question** – lets users inquire about public rights, government procedures, and official documentation in a conversational format.
2. **Scheme Finder** – suggests central or state government schemes based on user inputs like age, income level, category, and region.
3. **Public Issue Advisor** – guides users on how to raise complaints and resolve local civic problems (e.g., sanitation, roads, electricity) effectively and responsibly.

Each module was tested to ensure **clarity, ethical accuracy, and non-biased outputs**. The AI is not positioned as a legal or official authority but serves as an educational and supportive tool that complements official resources.

The application has proven to be:

* **Accessible** – works on both mobile and desktop devices via a simple browser link
* **User-Friendly** – with intuitive navigation and simplified output
* **Reliable** – backed by IBM Granite’s powerful and responsible natural language capabilities
* **Scalable** – ideal for integration with voice support, live government APIs, and regional language interfaces

We also identified some current **limitations**, such as reliance on internet access, token-based usage limits via Hugging Face API, and lack of voice or multilingual interaction. These limitations have been noted as future development opportunities.

The future scope for CitizenAI includes:

* **Voice-based interaction** for low-literacy or differently-abled users
* **Multilingual support** (beginning with Telugu) to break language barriers
* **Integration with official portals** for real-time updates and applications
* **User profile system** to track past queries and provide personalized civic assistance

In conclusion, **CitizenAI successfully fulfills its goal** of democratizing civic knowledge using responsible AI. It demonstrates the power of **generative AI combined with ethical design** in empowering citizens, encouraging civic participation, and simplifying government interactions. As a working prototype, CitizenAI stands as a strong foundation for future expansion, with the potential to transform how citizens access and engage with their government.

This project has been an enriching learning journey in terms of both **technical skills** and **social problem-solving**, and presents a ready-to-scale solution in the growing field of civic-tech innovation.

# *14. References*

1. **IBM Granite on Hugging Face**  
   <https://huggingface.co/ibm>  
   Used as the foundational large language model to handle user questions, provide scheme recommendations, and generate guidance for civic issues in natural language.
2. **Streamlit Documentation**  
   <https://docs.streamlit.io/>  
   Referred for building an interactive, responsive user interface with multiple modules including form inputs, user prompts, and result displays.
3. **Hugging Face API Guide**  
   <https://huggingface.co/docs/api-inference>  
   Helped understand how to send formatted prompts to the IBM Granite model, manage token limits, and handle real-time API responses.
4. **Pyngrok**  
   <https://github.com/alexdlaird/pyngrok>  
   Used for temporarily deploying the Streamlit app with a public-facing URL for demonstration and real-time testing.
5. **SmartInternz Internship Portal**  
   Platform used for project planning, daily submissions, mentor communication, and final project reporting as part of the structured internship program.