Project Report

1. INTRODUCTION:

1.1 Project Overview:

The Citizen AI Engagement Platform is a cutting-edge solution designed to foster meaningful interactions between citizens and government institutions. By harnessing the power of AI, this platform enables personalized communication, streamlines feedback mechanisms, and promotes inclusive decision-making processes.

1.2 Purpose:

The purpose of this platform is to:

1. Enhance Citizen Participation: Encourage active engagement and participation from citizens in governance processes.

2. Improve Transparency: Provide clear and accessible information, ensuring citizens are informed and empowered.

3. Foster Trust: Build trust between citizens and government institutions through responsive and effective communication.

4. Drive Inclusive Decision-Making: Ensure that diverse perspectives and needs are considered in policy-making processes.

2. IDEATION PHASE:

2.1 Problem Statement:

Define the Problem Statements

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Customer Problem Statement Template:

Create a problem statement to understand your citizen's point of view. The Citizen Problem Statement Template helps civic teams and government bodies focus on what truly matters to create public services and digital experiences that people trust and engage with.

A well-defined citizen problem statement enables your team to identify the right solutions for the real challenges that citizens face. Through this approach, you build empathy with citizens and gain a deeper understanding of how they interact with and perceive public services enabled by the Citizen AI Platform.

Example:

2.2 Empathy Map Canvas:

Ideation Phase

Empathize & Discover

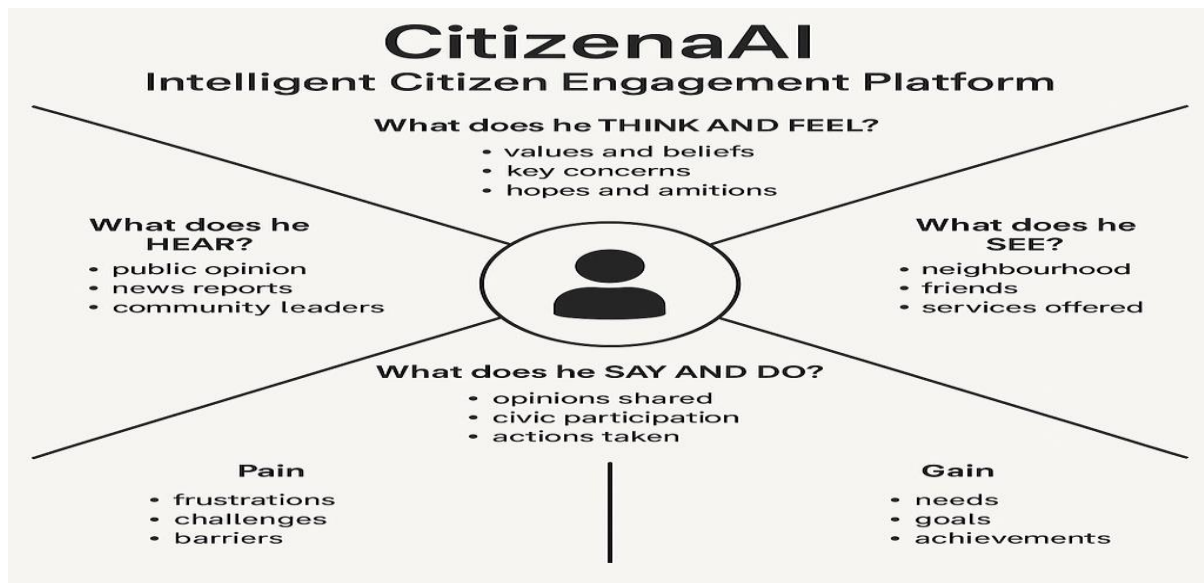| Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - intelligent citizen engagement platform
Maximum Marks	

Empathy Map Canvas:

An Empathy Map is a clear, concise visual tool designed to capture insights about citizens' behaviors, feelings, and needs. In the context of Citizen AI, it enables public service teams and

AI developers to deeply understand the people they serve—their lived experiences, goals, and frustrations.

Creating truly impactful AI-driven civic solutions requires deep empathy with citizens. The Empathy Map helps teams step into the shoes of the citizen to uncover the real, often unmet, needs behind their interactions with public systems.

Example:

Example: citizen Engagement :
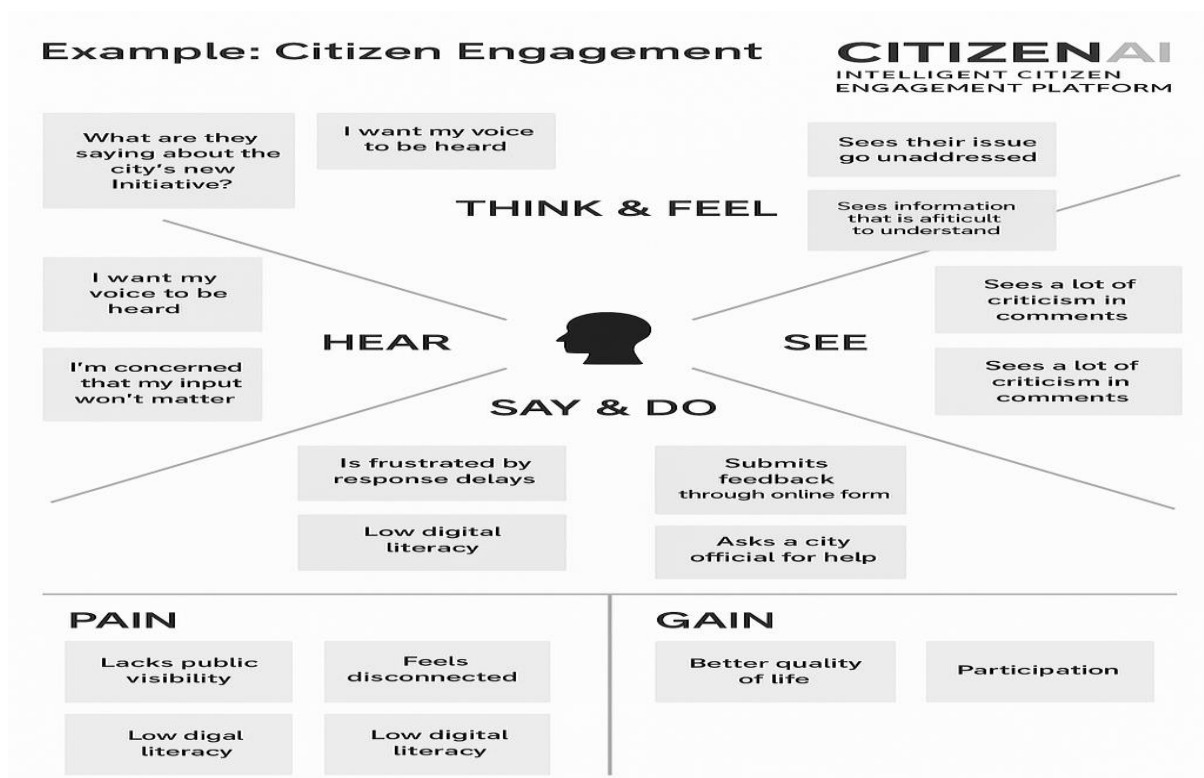
2.3 Brainstorming:

Ideation Phase

Brainstorm & Idea Prioritization Template:

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Brainstorm & Idea Prioritization Template:

Citizen AI Intelligent Citizen Engagement Platform: Idea Generation & Prioritization Just as traditional brainstorming fosters an open environment for creative problem-solving, a Citizen AI Intelligent Citizen Engagement Platform provides a dynamic space where every citizen can actively participate in shaping their community. This platform encourages the free flow of ideas, prioritizing the volume of diverse input over initial perceived value. Out-of-the-box suggestions are welcomed and, through the power of AI, can be built upon and refined. The platform facilitates collaboration, allowing citizens and AI to work together to develop a rich array of creative solutions for civic challenges.

Step-1: Team Gathering, Collaboration and Select the Problem Statement:

Step-2: Brainstorm, Idea Listing and Grouping:

Step-3:Idea-Prioritization:
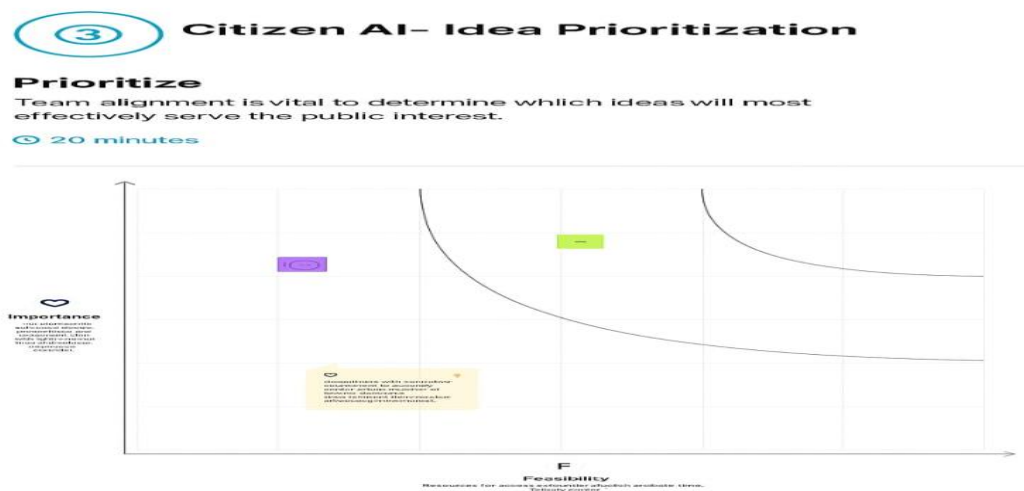
3. REQUIREMENT ANALYSIS:

3.1 Customer Journey Map:
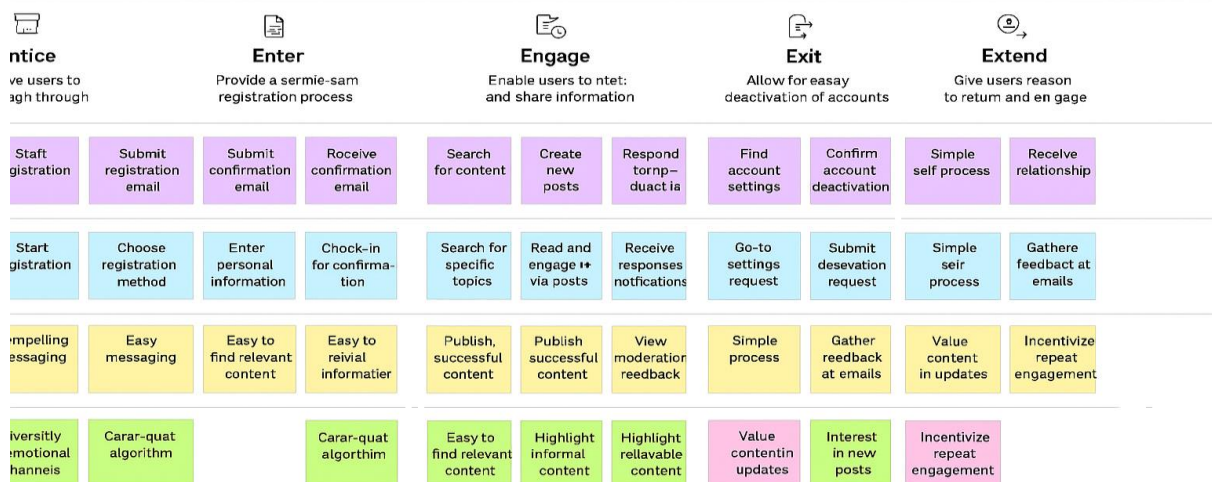
3.2 Solution Requirement:

Project Design Phase-II

Solution Requirements (Functional & Non-functional)

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirem	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3		
FR-4		
FR-6		

Non-functional Requirements:

Following are the non-functional requiremernts of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Useriity
NFR-2	Security	Reliabllity

3.3 Data Flow Diagram:

<div align="center">

Project Design Phase-II

Data Flow Diagram & User Stories

</div>

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	
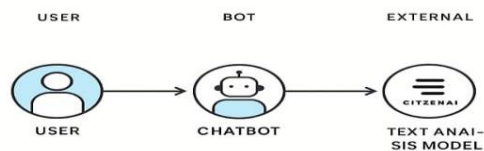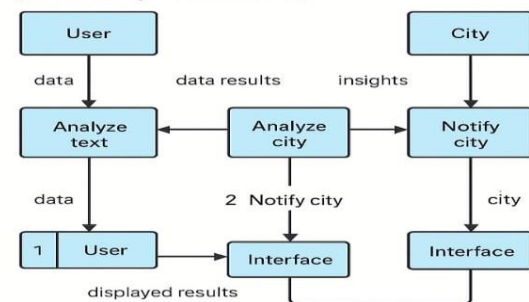
Citizen AI Intelligent Engagement Platform User Stories

Use the below template to list he user stories for the
user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Login	USN-1	As a user, I can register for the enail cation by entering my emall password, and confirming my password	I can access my account / dashboard	High	Sprint-1
	Login	USN-2	As a user, I will receive confirmation email once I have registered for the abplication through email	I can receive confirmation email & click link	High	Sprint-1
Customer Care Executive Administrator	Ldy	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	High	Sprint-1
Dashboard	USN-4	USN-4	As a user, I can log into the application by entering email & password	I can access my account / dashboard	High	Sprint-1
Customer	USN-5	USN-5	As an admin, I can view the	_	_	

3.4 Technology Stack:

<p style="text-align:center">Project Design Phase-II</p>

<p style="text-align:center">Technology Stack (Architecture & Stack)</p>

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Technical Architecture:
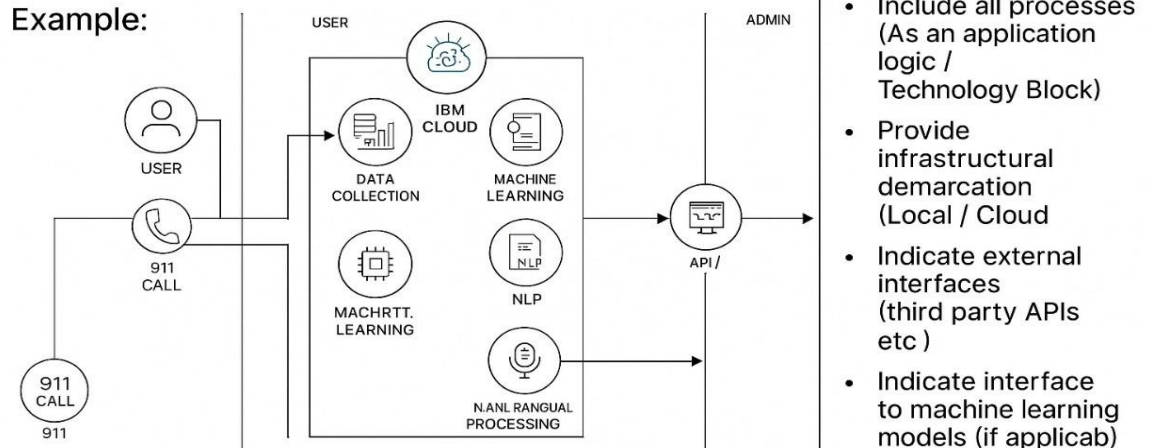

CitizenAI Intelligent Citizen Engagement Platform

Guidelines:
- Include all processes (As an application logic / Technology Block)
- Provide infrastructural demarcation (Local / Cloud
- Indicate external interfaces (third party APIs etc)
- Indicate interface to machine learning models (if applicab)

Table 1:

Citizen AI Intelligent Engagement Platform

S.No	Component	Description	Technology
1	User Interface	How user ineracts with application	User web App, Chatbot etc.
2	Application Logic1	Logic for a process in application	Java / Python
3	Application Logic2	Logic for a process in application	IBM Watson STT service
4	Application Logic3	Logic for a process in application	IBM Watson Assistant
5	External Database	Data Type, Configurations etc.	SQLite
6	Cloud Database	Database Service on Cloud	Database Service on Cloud
7	File Storage	File storage requirements	IBM Block Storage or Other Storage Service
8	External API-1	Purpose of External API e.g. Unique Identification	Aadhar API
9	External API-2	Purpose of External API e.g. Object Recognition	Object Recognition API
10	Machine Learning Model	Purpose of Machine Learning Model	ML framework, Models etc.
11	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Cloud Server Configuration:	Local, Cloud Foundry, Kubernetes etc.

4. PROJECT DESIGN:

4.1 Problem Solution Fit:

Project Design Phase

Problem – Solution Fit Template

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI-Intelligent Citizen Engagement Platform
Maximum Marks	

Problem-Solution Fit Template:

The Problem-Solution Fit is a powerful tool that ensures you've identified a genuine pain point for your customers and crafted a solution that truly addresses their needs. This template empowers entrepreneurs, innovators, and marketers to:

- Solve with Empathy: Tackle complex problems in a way that resonates with your customers' behaviors and preferences.

- Accelerate Success: Leverage existing habits and channels to boost solution adoption and achieve faster results.

- Communicate Effectively: Craft compelling messaging and triggers that speak directly to your target audience.

- Build Trust: Identify the right problem-behavior fit and deliver solutions that alleviate frustrations, urgent needs, or costly issues.

- Understand and Improve: Gain a deep understanding of your target group's situation and create meaningful improvements that make a lasting impact

Templet:

Citizen AI Intelligent Citizen Engagement Platform

CA AI DRIVEN AI ASSISTANT CA	C CITIZENS	D DATA SOURCES
An autonomous AI agent that interacts with citizens in a person alized and human-like manner	Individuals within communities, seexking assistance, information, or engagement	Open government data and APIs that provids real-time information relevant to citizen ne-eds
J JOBS-TO-BE-DONE/ NEEDS	**R RESPONSE CONTEXT**	**B BEHAVIOR**
Understand and address needs. issues, and concerns of citizens in a proactive and efficient	Crafting contextual, accurate, and policy-compliant responses based on available data ard previous interactions	Engages with citizens in conversational and empathetic way to provide support and information
T TRIGGERS	**S INTELLIGENT RESPONSE**	**CH CHANNELS OF BEHAVIOR**
A citizen reaches out with a question, issue, or request through a designated digital channel	The AI assistant provides information, resolves issues or escalates to human agents if needed	Text/SMS, mobile apps, web portals, and other accessible channels where citizens can engage with AI assistant

AMALTAMA

4.2 Proposed Solution:

Project Design Phase

Proposed Solution Template

Date	27 June 2025

≡ CITIZEN AI
intelligent citizen engageent platform

Proposed Solution Template:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	
2.	Idea / Solution description	
3.	Novelty / Uniquenesss	
4.	Social Impact / Customer Satisfaction	
5.	Business Model (Revenue Model)	

Proposed Solution Template:Project team shall fill the following information in the proposed solution template.

4.3 Solution Architecture:

Project Design Phase
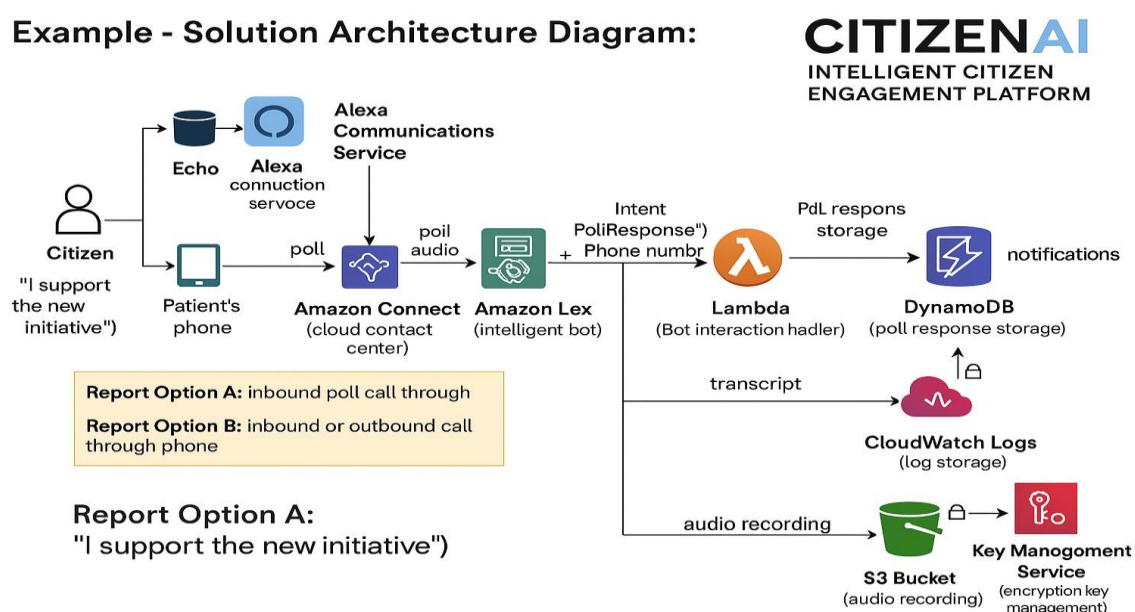
Solution Architecture

Proposed Solution Template:

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Solution Architecture:

Solution architecture for Citizen AI – Intelligent Citizen Engagement Platform is a structured and strategic process that connects civic challenges with innovative AI-driven technological solutions. It acts as a bridge between citizen needs and smart governance systems, ensuring effective public engagement and service delivery.

- Identify the Optimal AI-driven Solution

- Design a Transparent and Scalable System

- Detail Functionalities and Development Roadmap

- Establish Technical and Operational Standards

Example - Solution Architecture Diagram:

5. PROJECT PLANNING & SCHEDULING:

5.1 Project Planning:

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	27 June 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen Ai - intelligent citizen engagement platform
Maximum Marks	

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

CitizenAI Intelligent Citizen Engagement Platform

Sprint	Functional Requirement	User Story / Task	Story Points	Priority	Team Members
Sprint–1	Registration	As a user, I can register for the application by entering my email, password, and confirming my pavord.	2	High	
Sprint–2	USN–2	As a user, I will receive confirmation email once I have registered for the application.	3	High	
Sprint–2	USN–3	As a user, I can register for the application through Facebook.	2	High	
Sprint–1	Login	As a user, I can register for the application through Gmail.	2	Medium	
	Dashboard	As a user, can login	—	—	

6. FUNCTIONAL AND PERFORMANCE TESTING:

6.1 Performance Testing:

Functional & Performance Testing Templat

Date	27 june 2025
Team ID	LTVIP2025TMID59932
Project Name	Citizen AI - Intelligent Citizen Engagement Platform
Maximum Marks	

Test Scenarios & Results:

Test Case ID	Scenario (What to test)	Test Steps (How to test)	Expected Result	Actual Result	Pass/Fail
FT-01	Text Input Validation (e.g., topic, job title)	Enter valid and invalid text in input fields	Valid inputs accepted, errors for invalid inputs		
FT-02	Number Input Validation (e.g., word count, size, rooms)	Enter numbers within and outside the valid range	Accepts valid values, shows error for out-of-range		
FT-03	Content Generation (e.g., blog, resume, design idea)	Provide complete inputs and click "Generate"	Correct content is generated based on input		
FT-04	API Connection Check	Check if API key is correct and model responds	API responds successfully		
PT-01	Response Time Test	Use a timer to check content generation time	Should be under 3 seconds		
PT-02	API Speed Test	Send multiple API calls at the same time	API should not slow down		
PT-03	File Upload Load Test (e.g., PDFs)	Upload multiple PDFs and check processing	Should work smoothly without crashing		

7. RESULTS:

7.1 Output Screenshots:

8. ADVANTAGES & DISADVANTAGES:

Advantages:

1. Enhanced Citizen Engagement: Personalized interactions and streamlined feedback mechanisms encourage active participation.

2. Improved Transparency: Clear and accessible information empowers citizens and fosters trust.

3. Data-Driven Decision Making: AI-driven insights inform policy decisions, ensuring more effective governance.

4. Increased Efficiency: Automated processes and AI-powered tools reduce administrative burdens.

Disadvantages:

1. Digital Divide: Limited access to technology may exclude certain citizen groups.

2. Data Privacy Concerns: Collection and analysis of citizen data raise privacy and security concerns.

3. Bias in AI Decision Making: AI algorithms may perpetuate existing biases if not properly designed and trained.

4. Dependence on Technology: Over-reliance on AI may lead to decreased human interaction and empathy.

5. Technical Issues: Platform malfunctions or technical glitches may undermine citizen trust and engagement.

9. CONCLUSION:

The platform enhances citizen participation, transparency, and inclusivity in governance through AI-driven technologies. While offering benefits like improved engagement and efficiency, it requires careful consideration of challenges like data privacy and inclusivity. A promising solution for modernizing governance.

10. FUTURE SCOPE:

The platform's future scope includes:

1. Expansion to More Cities/Regions

2. Integration with Emerging Technologies (AR, VR, Blockchain)

3. Enhanced AI Capabilities for Personalized Services

4. Increased Focus on Data Analytics and Insights

5. Collaboration with Other Smart City Initiatives

11. APPENDIX:

Source Code :

```
!pip install transformers torch gradio accelerate bitsandbytes
# Imports
import gradio as gr
import torch
from transformers import AutoTokenizer, AutoModelForCausalLM, pipeline


# 🎲 CitizenAI Core Class
class CitizenAI:
    def _init_(self):
        self.model_name = "ibm-granite/granite-3.3-2b-instruct"
        self.tokenizer = None
        self.model = None
        self.pipeline = None
        self.load_model()
def load_model(self):
    try:
        print("🔁 Loading AI model...")
        self.tokenizer = AutoTokenizer.from_pretrained(self.model_name, trust_remote_code=True)
    self.pipeline = pipeline(
            "text-generation",
            model=self.model,
            tokenizer=self.tokenizer,
            max_length=1024,
            temperature=0.7,
            do_sample=True,
```

```python
            pad_token_id=self.tokenizer.eos_token_id
        )
        print("✅ AI model loaded.")
    except Exception as e:
        print(f"❌ Error: {e}")
        print("⚠ Falling back to DialoGPT-medium...")
        fallback_model = "microsoft/DialoGPT-medium"
        self.tokenizer = AutoTokenizer.from_pretrained(fallback_model)
        self.model = AutoModelForCausalLM.from_pretrained(fallback_model)
        self.pipeline = pipeline(
            "text-generation",
            model=self.model,
            tokenizer=self.tokenizer,
            max_length=1024,
            temperature=0.7,
            do_sample=True,
            pad_token_id=self.tokenizer.eos_token_id
        )
        print("✅ Fallback model loaded.")

def handle_citizen_query(self, query):
    prompt = f"You are an intelligent citizen engagement assistant. The user has the following query:\n\n{query}\n\nProvide a clear, helpful, civic-centered response:"

    response = self.pipeline(prompt)
    result = response[0]['generated_text'].split("response:")[-1].strip()
    return result

def get_emergency_contacts(self, city):
```

```python
        prompt = f"You are an AI civic assistant. Provide a list of emergency contact
numbers (police, fire, ambulance, disaster helpline) for the city: {city}."

        response = self.pipeline(prompt)

        result = response[0]['generated_text'].split("helpline:")[-1].strip()

        return result

    def get_city_analysis(self, city):

        prompt = f"You are an AI civic analyst. Provide an overall analysis for the city:
{city}, covering recent events, safety, infrastructure, and public services."

        response = self.pipeline(prompt)

        result = response[0]['generated_text'].strip()

        return result

# 📃 Gradio Interface Builder

def create_gradio_interface():

    with gr.Blocks(title="Citizen AI - Intelligent Engagement Platform") as app:

        gr.HTML("<h1 style='text-align:center;'>🏙 Citizen AI</h1><p style='text-
align:center;'>AI-powered Intelligent Citizen Engagement Platform</p>")

with gr.Tabs():

            # 🗣 Citizen Query Assistant Tab

            with gr.Tab("🗣 Ask Citizen AI"):

                citizen_query_input = gr.Textbox(label="Enter your query", lines=6)

                query_btn = gr.Button("Get AI Response")

                query_output = gr.Textbox(label="Citizen AI Response", lines=12)

# 🆘 Emergency Contact Finder Tab

            with gr.Tab("🆘 Emergency Contacts"):

                city_input = gr.Textbox(label="Enter your city name", lines=1)

                contact_btn = gr.Button("Get Emergency Contacts")
```

```python
        contact_output = gr.Textbox(label="Emergency Contact Information",
lines=10)


        # 📊 City Analysis Tab

        with gr.Tab("📊 City Analysis"):

            analysis_city_input = gr.Textbox(label="Enter your city name", lines=1)

            analysis_btn = gr.Button("Get City Analysis")

            analysis_output = gr.Textbox(label="City Analysis Report", lines=12)
    # Event Handlers

        # These handlers will now be able to access citizen_ai as it's initialized
outside the _main_ block

        def handle_query(query):

            if not query.strip():

                return "🚫 Please enter a query."

            result = citizen_ai.handle_citizen_query(query)
    return result
    def handle_contacts(city):

            if not city.strip():

                return "🚫 Please enter a city name."

            result = citizen_ai.get_emergency_contacts(city)

            return result
    def handle_city_analysis(city):

            if not city.strip():

                return "🚫 Please enter a city name."

            result = citizen_ai.get_city_analysis(city)

            return result
```

```python
    # Bind Buttons

    query_btn.click(fn=handle_query,inputs=citizen_query_input,
outputs=query_output)

    contact_btn.click(fn=handle_contacts,inputs=city_input,
outputs=contact_output)

    analysis_btn.click(fn=handle_city_analysis,inputs=analysis_city_input,
outputs=analysis_output)

gr.HTML("<p style='text-align:center; color:gray;'>⚙ Powered by IBM Granite AI
| CitizenAI Platform</p>")

return app

# 🚀 Run Application

# Move the initialization outside the _main_ block

print("🚀 CitizenAI Initializing...")

citizen_ai = CitizenAI()

iface = create_gradio_interface()

print("🌐 Launching CitizenAI with public link...")

iface.launch(share=True)
```

GitHub : https://github.com/nagamallika1/Citizen-AI

Project-Demo-Link:
**https://drive.google.com/file/d/1k5SKnhgAXUtyO7rwM4E6WFBNp8HT1cEt/
view?usp=drivesdk**