**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

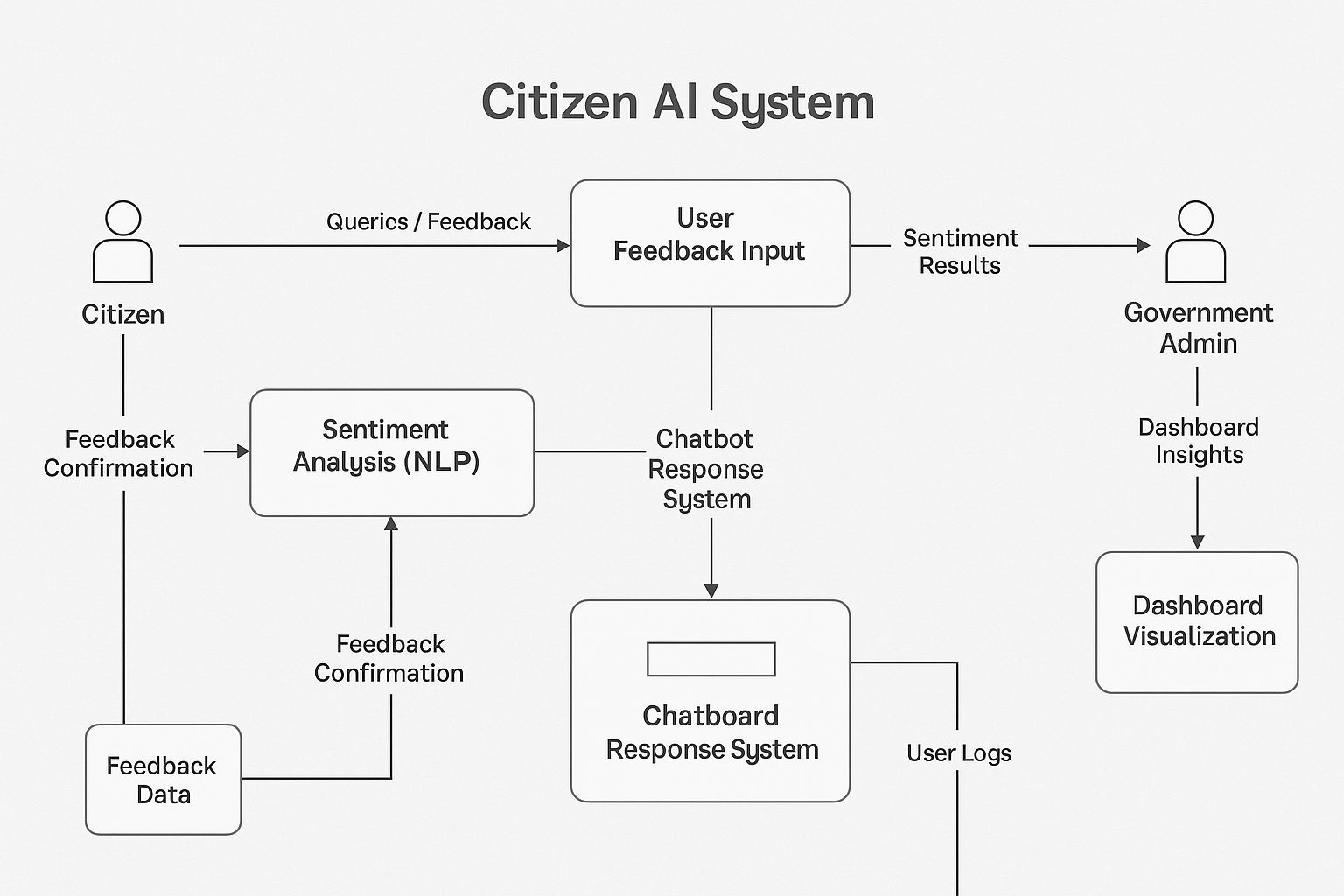
| Date | 31 January 3035 |
| --- | --- |
| Team ID | LTVIP2025TMID59424 |
| Project Name | Citizen AI – Intelligent Citizen Engagement Platform |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)



**Table-1 : Components & Technologies:**

| **S.No** | **Component** | **Description** | **Technology** |
| --- | --- | --- | --- |
| **1** | **User Interface** | **Web UI for citizens and admin dashboard** | **HTML, CSS, JavaScript, Bootstrap, Streamlit** |
| **2** | **Application Logic-1** | **Core application backend and routing logic** | **Python (Flask, FastAPI)** |
| **3** | **Application Logic-2** | **Sentiment Analysis via NLP models** | **TextBlob, IBM Watson NLU, VADER** |
| **4** | **Application Logic-3** | **Chatbot functionality and AI conversation model** | **IBM Watson Assistant, IBM Granite / GPT-3.5** |
| **5** | **Database** | **Stores feedback, sentiment, and user logs** | **SQLite (Local), PostgreSQL (Planned)** |
| **6** | **Cloud Database** | **Cloud-based storage of structured feedback data** | **IBM DB2, IBM Cloudant (Future Integration)** |
| **7** | **File Storage** | **Storing logs and form data** | **Local Filesystem, IBM Block Storage (Future)** |
| **8** | **External API-1** | **To pull weather/location context for contextual responses (optional)** | **IBM Weather API** |
| **9** | **External API-2** | **For verifying user identity during registration (optional)** | **Aadhaar API** |
| **10** | **Machine Learning Model** | **NLP model for sentiment classification and language understanding** | **IBM Granite, OpenAI GPT-3.5, Fine-tuned LLM** |
| **11** | **Infrastructure** | **Deployment for development and production** | **Localhost (Dev), IBM Cloud, Cloud Foundry** |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
| **1** | **Open-Source Frameworks** | **Utilizes open-source libraries and frameworks for full-stack development** | **Flask (Python), Streamlit, SQLite, Chart.js, Bootstrap** |
| **2** | **Security Implementations** | **Basic authentication, input sanitization, protected API keys, secure forms** | **JWT (Planned), SHA-256 (Hashing), HTTPS, OAuth 2.0 (Planned), OWASP Guidelines** |
| **3** | **Scalable Architecture** | **Designed using layered architecture; future-ready for microservices adoption** | **3-Tier Architecture (Frontend - Backend - DB), Docker (Planned), REST API** |
| **4** | **Availability** | **Can be deployed on cloud for 24/7 access; designed for load balancing** | **IBM Cloud, Cloud Foundry, Horizontal Scaling (Planned)** |
| **5** | **Performance** | **Designed for <3 sec response time, low-latency APIs, supports future caching** | **FastAPI (for performance), SQLite, Redis (Planned), Async Processing (Planned)** |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)