**Project Design Phase**

**Solution Architecture**

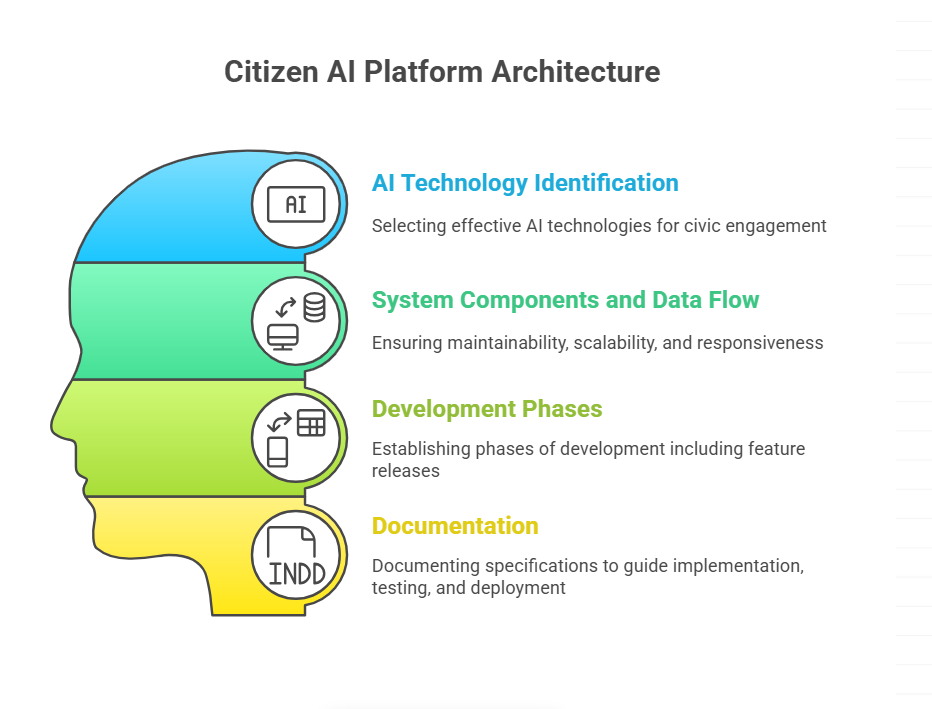
|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID31802 |
| Project Name | Citizen AI |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

* Find the best tech solution to solve existing business problems.
* Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
* Define features, development phases, and solution requirements.
* Provide specifications according to which the solution is defined, managed, and delivered.

**Example - Solution Architecture Diagram:**



**Architecture Components**

The Citizen AI solution is comprised of the following core modules:

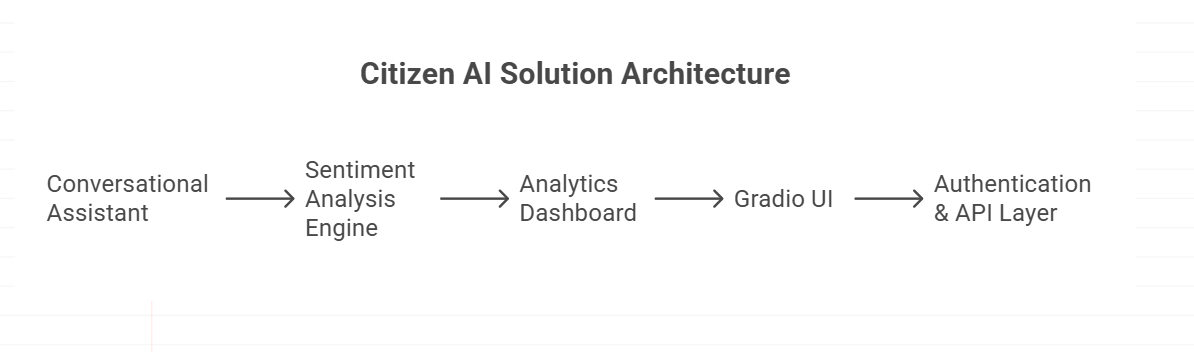
- Conversational Assistant: Built using IBM Granite models for real-time, context-aware interactions.

- Sentiment Analysis Engine: Integrated using Hugging Face models to classify feedback.

- Analytics Dashboard: Built using Matplotlib and Pandas for visualizing sentiment trends.

- Gradio UI: Facilitates user interaction through an intuitive two-tab interface for chat and dashboard.

- Authentication & API Layer: Secures API keys and enables communication with external services.



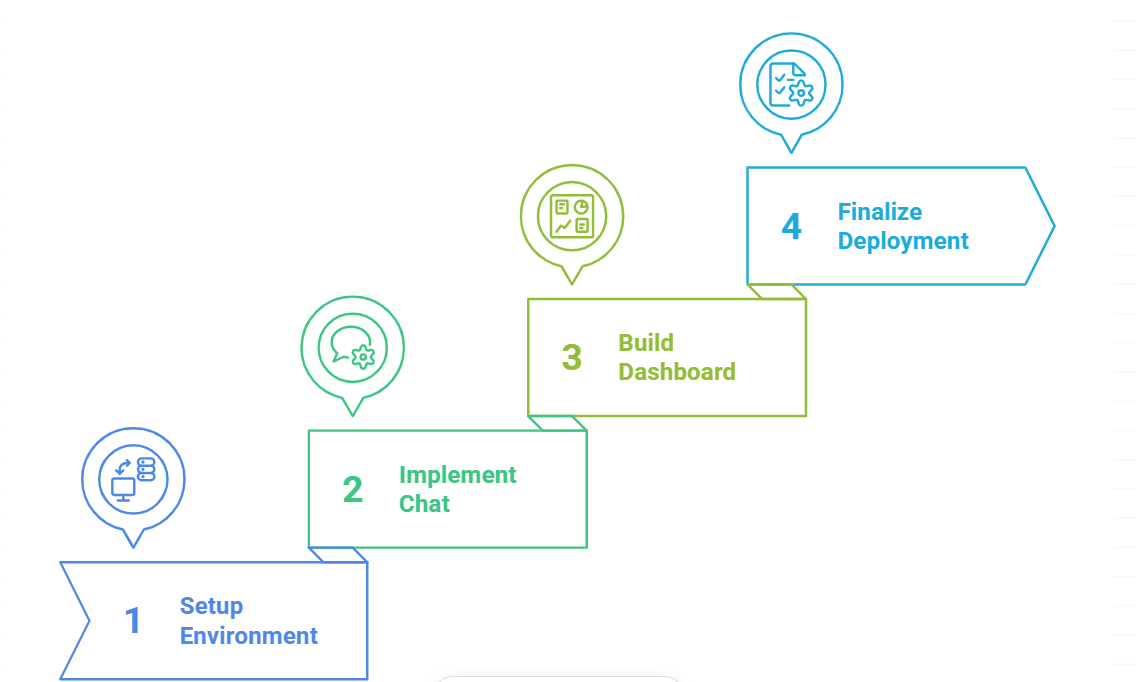
**Development Phases**

- Phase 1: Setup environment, integrate Granite and Hugging Face models.

- Phase 2: Implement real-time chat functionality and feedback classification.

- Phase 3: Build dashboard and plotting tools with CSV/JSON downloads.

- Phase 4: Conduct performance and UAT testing, finalize deployment settings.

****

**Data Flow (Described Architecture)**

1. Citizen submits a query → sent to IBM Granite for response generation.  
2. Citizen submits feedback → passed to sentiment model → logged to CSV/JSON.  
3. Dashboard component reads data → generates pie chart and sentiment trend.  
4. User can download processed feedback logs.  
5. System checks API health and gracefully handles errors or missing tokens.

