

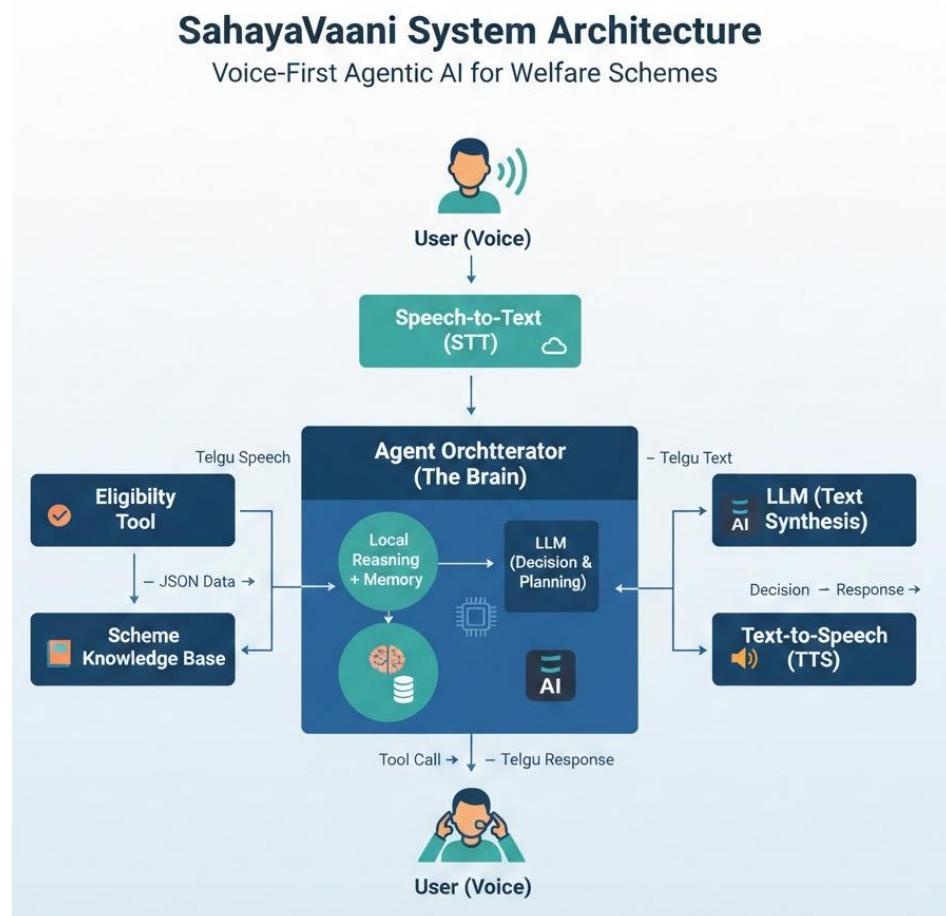
Architecture Document – SahaayaVaani

This document describes the **system architecture**, **agent lifecycle**, **decision flow**, **memory handling**, and **prompting strategy** used in the SahaayaVaani voice-first welfare scheme assistant.

1. High-Level System Architecture

SahaayaVaani is a **voice-driven, agentic system** that processes user speech, reasons over extracted information, uses tools to make decisions, and responds using speech.

High-Level Flow

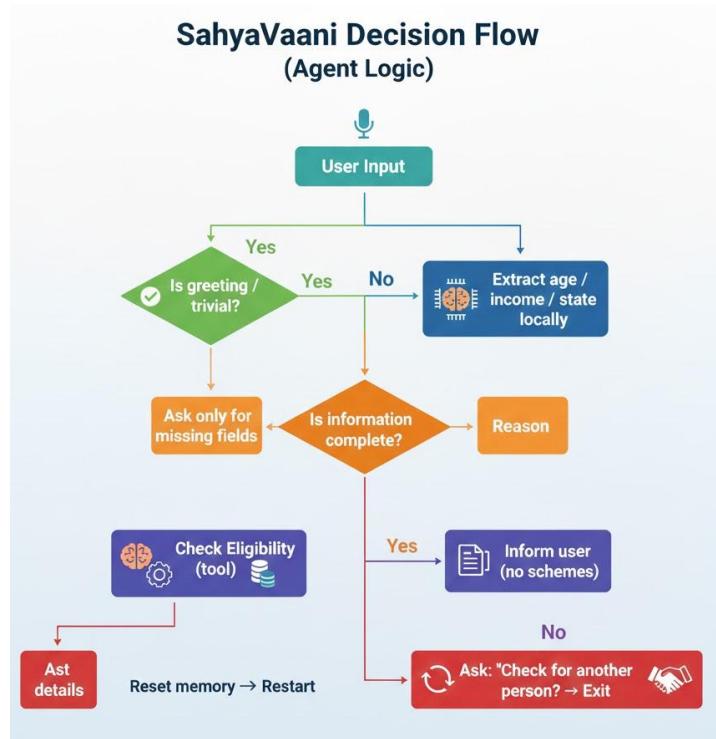


2. Agent Lifecycle

The agent follows a **multi-stage lifecycle** instead of a single request–response flow.



3. Decision Flow (Agent Logic) : The agent does **not** follow a fixed script. Instead, it adapts based on conversation state and available information.



4. Memory Architecture

The system maintains **session-level memory** to store user information across turns.

Memory Contents

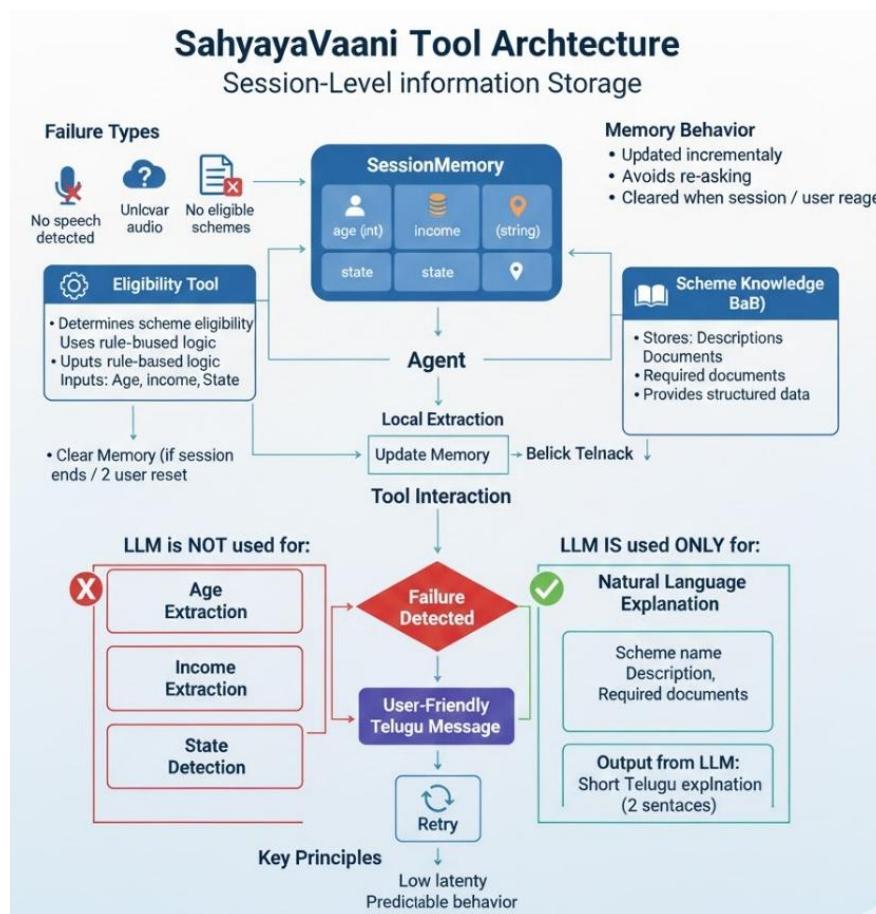
Session Memory

- age
- income
- state

Memory Behaviour

- Updated incrementally after each user input
- Used to avoid re-asking already provided details
- Cleared when:
 - A conversation ends
 - User chooses to check for another person

Memory Flow Diagram



5. Tool Architecture

The agent uses **multiple tools**, each with a clearly defined responsibility.

Tools Used

1. Eligibility Engine

- Determines scheme eligibility based on:
 - Age
 - Income
 - State
- Uses rule-based logic

2. Scheme Knowledge Base

- Stores:
 - Scheme descriptions
 - Required documents
- Provides structured data for explanation



6. Prompting Strategy

The system uses **minimal and controlled prompting**.

Key Principles

- LLM is **not** used for:
 - Age extraction
 - Income extraction
 - State detection

- LLM is used **only once**, for:
 - Natural language explanation of results

Prompt Scope

Input to LLM:

- Scheme name
- Description
- Required documents

Output from LLM:

- Short Telugu explanation (2 sentences max)

This approach ensures:

- Low latency
- Quota safety
- Predictable behavior

7. Failure Handling Architecture

The system includes explicit handling for failure scenarios.

Failure Types

- No speech detected
- Noise / unclear audio
- Missing user information

- No eligible schemes

