1. Problem statement:

AI-Powered Digital Call Center Using Autonomous AI Agents

Traditional contact centers are expensive, slow to scale, and heavily dependent on human agents. Enterprises need AI-first, agent-driven digital call centers that can handle voice, chat, email, and messaging channels with minimal human intervention while ensuring quality, compliance, and escalation.

2. Brief End to End Solution:

Here the designed system just starts by capturing the user's voice as the input, which is then processed by the Large Language Model (LLM) using the predefined data to understand the user's intent.

The LLM then converts the voice input into a processed clear text like format and then is sent to the chat processing model, where the sentimental analysis is performed, the conversation that the user made is maintained and the summary and feedback of the conversation is generated. These insights would help in understanding the user's behavior based on the sentimental analysis and improve the response quality.

For the voice based interaction, the processed text is converted to speech using a Text-to-Speech (TTS) engine. Then the audio of the speech is then analyzed to generate transcriptions, measure the call duration and then this would help in the extraction of analytics like metadata after the call.

All reports, issues, and feedback collected from both chat and audio processing are then given as the input into the RAG layer, which enhances the LLM. This creates the feedback driven loop that improves the systems accuracy, intelligence and user experience.

3. Unique Approach

- **Back-to-Back User Agent via Kamailio :**

  This allows for deep packet-level control. By terminating the RTP stream at the media server and splitting it into a PCM Audio Stream for the ASR, you gain lower latency than standard web-socket relays. It allows for "audio injection" directly into an active SIP session, making the AI feel like a native participant in the phone call rather than an overlay. This gives the lowest latency, Accuracy, Private hosted and customisable.

- **The Multi-Track Analytics Engine :**

  Most systems just record a call, our solution treats the call as three distinct data streams (Voice, Chat, and Audio) processed in parallel.
  The solution has a dedicated Data Analytics block that performs "Sentiment Analysis" and "Context of the conversation" summaries post-call. This transforms a simple voice bot into a business intelligence tool. The "Sentiment Score" and provide immediate feedback to the admin dashboard without manual review.

- **Edge-Cloud Hybrid Deployment:**

  The solution "Hosted Server" vs. "Local Server" (Llama, Kokoro TTS, ngrok) setup is highly innovative for privacy and cost management. Can process sensitive data locally to maintain privacy while using high-fidelity cloud models for the user-facing voice, optimizing both performance and cost.

4. Target Users and use cases:

a. Primary Target Users / Customer Segment

The solution is built for the Enterprise and SMB (Small to Medium Business) Customer Service segment. Specifically:

D2C Retail & E-commerce: Businesses requiring automated handling of order tracking, delivery logistics, and FAQ resolution.

VoIP Service Providers: Organizations looking to integrate advanced "Voice-as-a-Service" (VaaS) layers into their existing SIP-based infrastructure.

Corporate Helpdesks: Internal departments in large organizations that need to automate routine employee inquiries regarding ID cards, benefits, or IT support.

b. Main Use Cases :

Automated Logistics & Order Management:

The system acts as an intelligent agent for order-related queries. When a customer calls, the system identifies them via their phone number, queries the SQL Transaction database via RAG, and provides real-time status updates.

- **Example Scenario:** A user calls to ask "Where is my grocery delivery?". The system fetches the current PCM stream, transcribes it, queries the delivery status, and responds with a natural voice using **TTS**, injecting the audio directly into the RTP stream.

High-EQ Support & Escalation:

The system is designed to handle complex customer support scenarios where emotional intelligence is required.

- **Example Scenario:** If a caller expresses frustration, the Sentiment Analysis engine flags the call in real-time. The system can then prioritize the call on the Admin Dashboard for a human agent to "Engage" or "Playback," ensuring that sensitive issues are resolved with a human touch while routine data collection is handled by the AI.

c. User Environment Assumptions :

To ensure the performance of this real-time AI solution, the following assumptions are made regarding the environment:

Telephony Compatibility: The caller is assumed to be using standard PSTN or SIP-compliant devices. My system handles Jitter and DTMF on the Media Server level to maintain call quality.

End-User Proficiency: No technical skills are required from the caller; the interface is entirely natural language. Admins are assumed to have basic proficiency with web-based dashboards for monitoring KPIs and Reports.

5. Architecture and technical design

a. High-Level Architecture Description:

The architecture is divided into four primary layers:

- **Telephony & Signaling Layer:** The Kamailio is used as a SIP Proxy to handle high-concurrency signaling and load balancing. This layer manages the initial SIP INVITE/SDP exchange.
- **Media Processing Layer:** The core of media handling is built on Asterisk. It is configured as the layer as a B2BUA (Back-to-Back User Agent**)** to terminate the RTP stream. It performs critical tasks like Jitter Buffer management, DTMF detection, and most importantly splitting the audio into a PCM stream for real-time processing.
- **Orchestration Layer :** This is a high-performance backend that manages the real-time pipeline:

  Streaming ASR: Transcribes the PCM stream into text with sub-second latency.

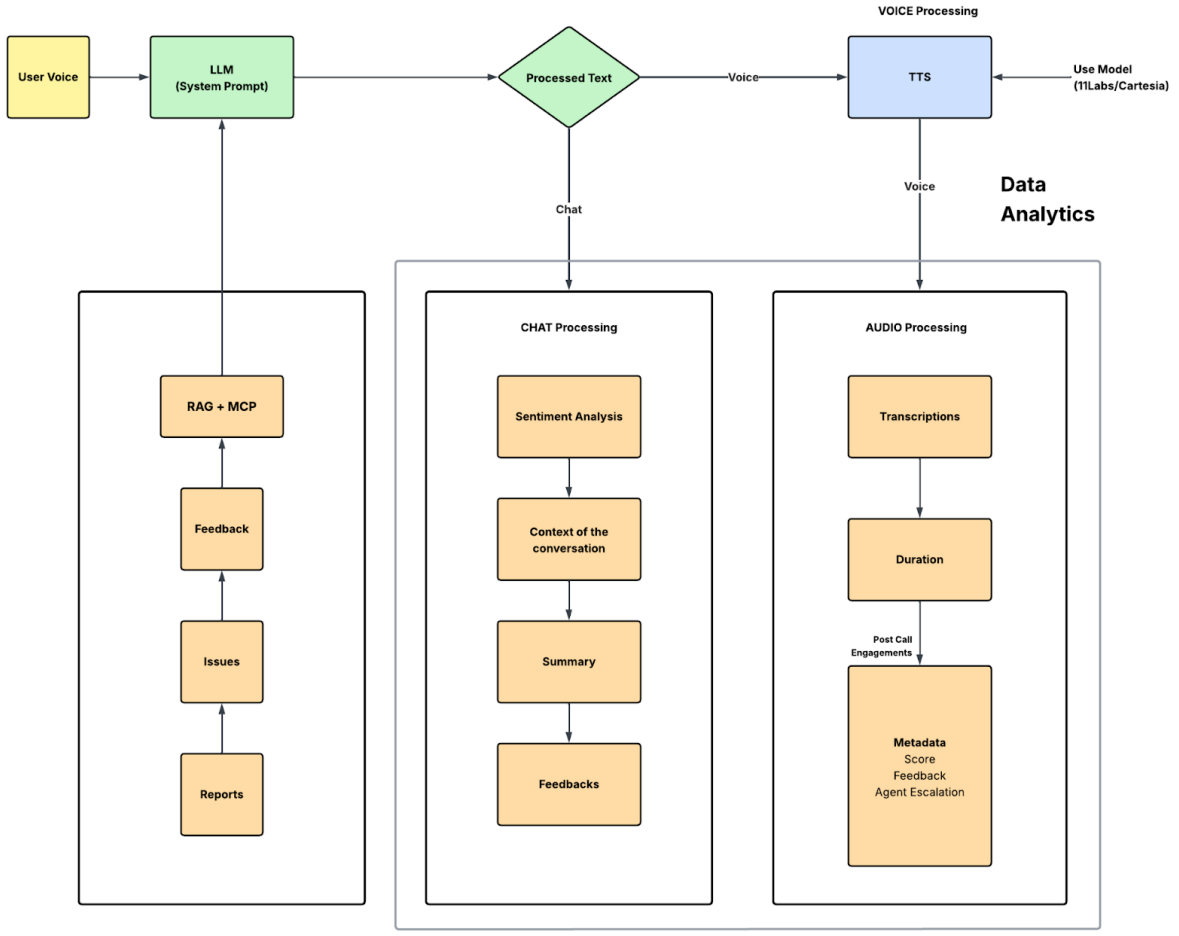
  Agentic LLM: Uses RAG (Retrieval-Augmented Generation) and the Model Context Protocol (MCP) to fetch real-time data from my SQL databases (e.g., Online Grocery Store transactions) and provides a high-EQ response.

  TTS Synthesis: Converts the generated text back into high-fidelity speech.

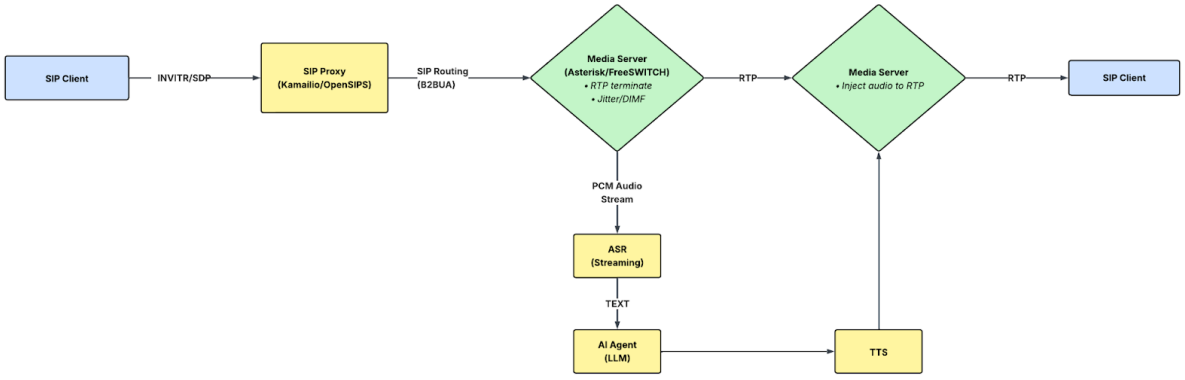
b. Technical Stack:

Languages - Python, HTML, CSS, Javascript.
Base Model - fixie-ai/ultravox-v0.5-llama-3-2-1b
pipeline - Text to speech model(kaetesia/elevenlabs)
Cloud Hosting - Vulte/AHS/Replicate AI/Digital Ocean/ ketlb/Linode

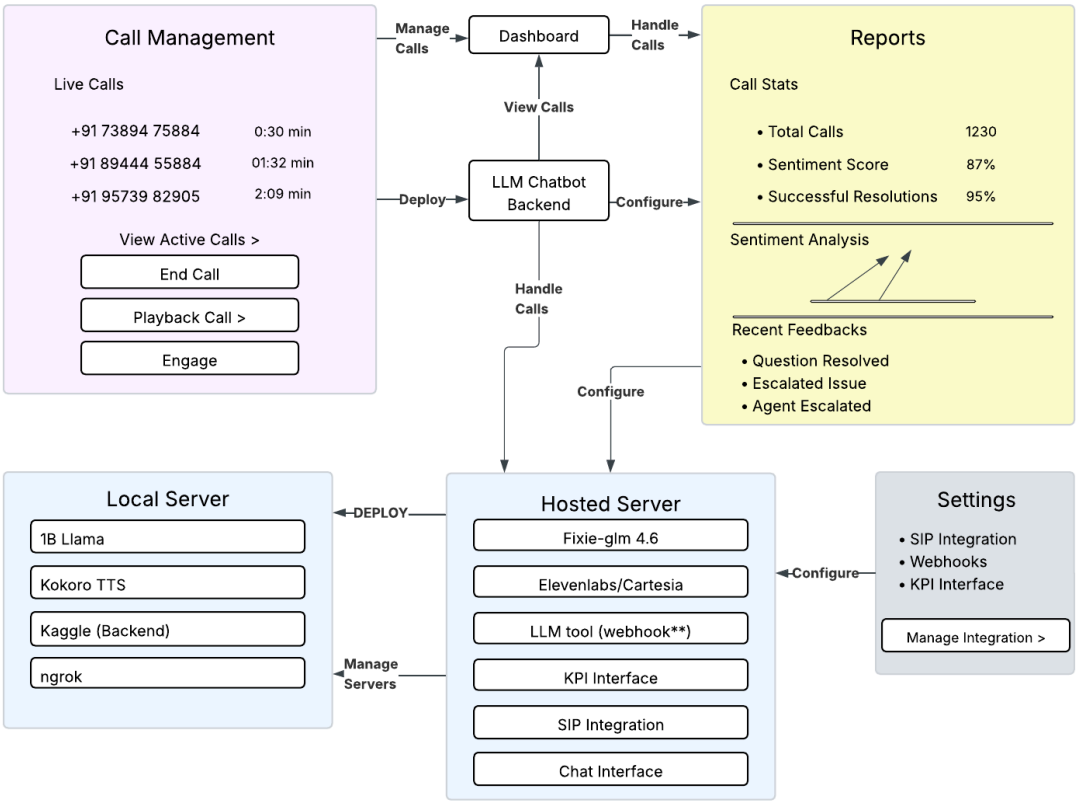
Main Flow :



SIP Flow:



Final Flow :



6. Implementation details

a. Current Implementation Status

The project has reached the end-to-end flow stage. The system is fully integrated, enabling a seamless transition from the initial telephony signaling to the final AI-generated audio response. The current status encompasses the following:

Integrated Signaling & Media: The SIP Proxy and Media Server (Asterisk/FreeSWITCH) successfully establish calls, terminate RTP streams, and bridge audio to the AI backend.

Real-time Processing Pipeline: The streaming ASR, LLM reasoning (via RAG/MCP), and TTS synthesis are fully operational, maintaining a synchronized data flow. **Functional Analytics:** Post-call data is automatically captured and visualized on the Admin Dashboard, providing real-time updates on call stats and sentiment.

b. Data Utilization and Processing

The system utilizes a combination of synthetic datasets and structured transactional data to simulate a production environment.

Data Sources:

- **Transactional Data:** For the Online Grocery Store use case, a relational database is used consisting of Product, Transactions, and Admin tables. These contain synthetic records of inventory, customer IDs, and purchase histories to test the RAG layer's retrieval accuracy.
- **Conversational Data:** Sample voice recordings and text transcripts are used to calibrate the Sentiment Analysis and ASR models, ensuring they recognize domain-specific terminology (e.g., specific grocery items or technical support terms).

7. User experience and workflow

The user experience is built on a Dual-Interface model, combining a natural voice interaction with a visual Admin Dashboard. This allows for real-time monitoring and manual intervention when necessary.
Initiation: The caller dials the system via a standard SIP client or phone line.

Visual Tracking: Simultaneously, the call appears on the Admin Dashboard, displaying the caller's ID and connection status.

Dynamic Interaction: As the user speaks, the dashboard provides a live transcript and real-time Sentiment Analysis scores.

Resolution & Data Display: The system retrieves information (e.g., from the Online Grocery Store database) and presents the data both vocally to the user and visually on the dashboard for the administrator.

Completion & Review: After the call, the UI populates a summary report, including the call duration, resolution status, and the generated transcript for future reference.

8. Challenges and Limitations

a. Technical Challenges Faced

VRAM & Hardware Orchestration:
The most complex technical hurdle faced was the transition from API-dependent models to a self-hosted, hybrid infrastructure. Hosting large-scale models like Llama 3 and Kokoro TTS locally introduced several critical engineering challenges: VRAM & Hardware Orchestration: the VRAM footprint was a primary challenge; for instance, Llama 3 8B typically requires ~16GB-20GB of VRAM in FP16. To fit this alongside a real-time TTS engine on consumer or mid-grade enterprise hardware.

Local-to-Cloud Tunneling Latency:
Using ngrok to bridge the local servers with the hosted SIP proxy (Kamailio) introduced an extra network hop. Had to tune the RTP Jitter Buffer on the Media Server to handle the slight variability in packet arrival times caused by this tunneling, ensuring the audio remained fluid and synchronized.

9. Future enhancements and roadmap

Advanced Local Model Optimization:

To further reduce dependency on external APIs and improve data privacy, the roadmap includes the transition toward specialized, fine-tuned local models.

- **Domain-Specific Fine-Tuning:** The intent is to fine-tune Llama 3 or similar open-source models on specific industry datasets (e.g., Retail/Grocery or IT Support) to improve the accuracy of RAG-based retrievals.
- **Speculative Decoding:** Implementing speculative decoding techniques will be explored to accelerate token generation, aiming to push the "Time to First Token" (TTFT) below 300ms.
- **Local TTS Advancements:** The system will transition to fully local, high-fidelity synthesis using models like Kokoro-82M or Piper, optimized for low-latency execution on edge hardware.

Architectural Scaling and Reliability

The next phase of technical development focuses on transforming the current single-instance prototype into a high-availability cluster.

- **Distributed Inference Clusters:** Using vLLM or TGI (Text Generation Inference), the system will be designed to scale across multiple GPU nodes to support hundreds of concurrent calls.
- **Native WebRTC Integration:** While currently utilizing SIP/RTP, the system will be expanded to include native WebRTC support. This will allow the AI voice agent to be embedded directly into web browsers and mobile applications without requiring a traditional telephony interconnect.

10. Team member details

Name	Email	Phone	Company
Aadarsh Nagrikar	aadarsh.n@tcs.com	7620216605	TCS
Praveen Arumugam	praveen.arumugam1@tcs.com	7904272100	TCS
Hariom Shivhare	hariom.shivhare@tcs.com	8767008518	TCS