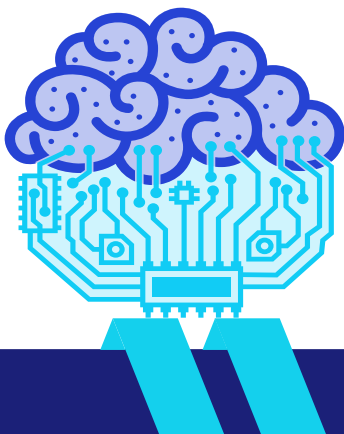
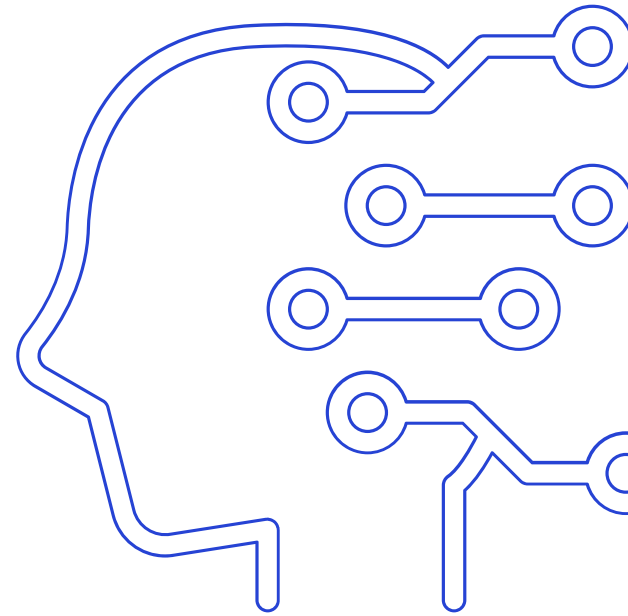


Project Showcase: Livekit Voice Agent – Dynamic Conversational AI

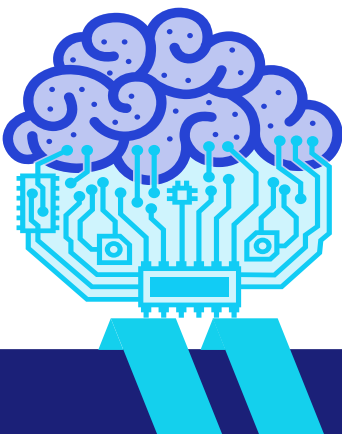
Project Showcase: Livekit Voice Agent – Dynamic Conversational AI

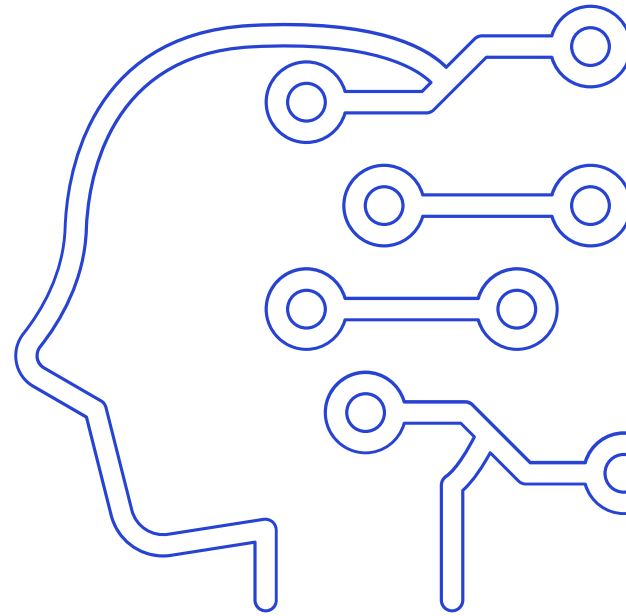




Project Summary

- Engineered a Livekit Voice Agent, a dynamic, real-time conversational AI system designed for complex, flow-based customer support.
- This is a robust, multi-voice solution that establishes a new standard for intelligent and flexible support infrastructure.





Code & Demo

- GitHub: <https://github.com/gspec-1/voice-agent-livekit-affan>
- Demo: <https://voice-agent-livekit-affan.vercel.app/>

A screenshot of a web interface for a voice assistant. The interface has a purple header with a robot icon, the text 'Voice Assistant', and a status indicator 'Ready to help' with 'Intent: Offline'. Below this, there's a list of session details: Session ID, Status, and Features. The main section is titled 'Connect to Voice Assistant' with a subtitle 'Select your bot and voice to start a conversation'. It contains three input fields: 'Bot Name' (with a placeholder 'e.g., voice-1, dustin-gpt, sales-bot'), 'Org Name' (with a placeholder 'e.g., alive5stage0, your-org-name'), and 'Agent Voice' (a dropdown menu showing 'Jacqueline - Reassuring Agent'). A 'Join Voice Chat' button is at the bottom.

Voice Assistant Ready to help Offline

Intent: Offline

- Session ID: session_1759253919815_q9tm46
- Status: Ready to connect to Alive5 Support
- Features: Dynamic Intent Detection, Real-time Analytics

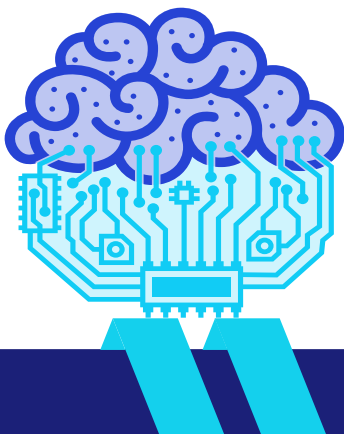
Connect to Voice Assistant
Select your bot and voice to start a conversation

Bot Name:
Enter the name of the bot you want to connect to

Org Name:
Leave empty to use default configuration

Agent Voice:
Choose the voice for your AI assistant

Join Voice Chat

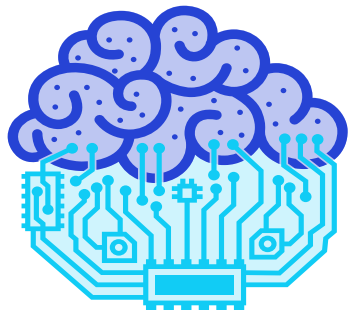


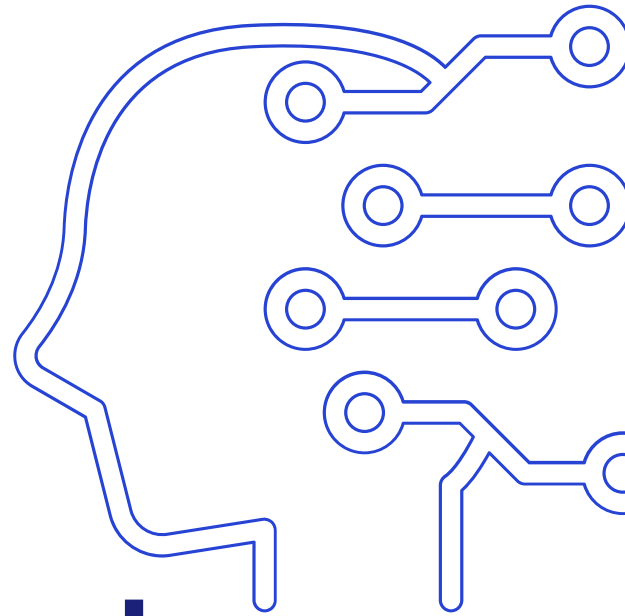


AKENO TECH

Key Technical Achievements

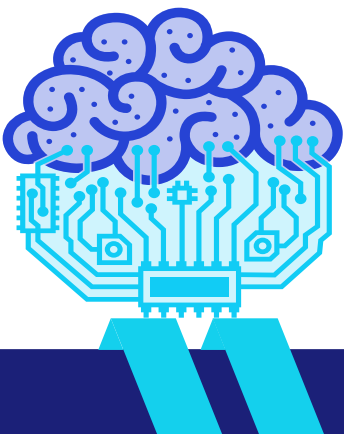
- Dynamic Flow Architecture: JSON-driven conversational logic from external site (Alive5).
- Intelligent Intent Routing: Dynamic AI-powered intent detection and execution.
- Resilient Fallback Mechanism: Queries Alive5 FAQ bot (Bedrock API).
- High-Fidelity Communication: Livekit (RTC), Deepgram (STT), Cartesia (TTS).
- Structured Interface: Secure, modular bot flow setup.





Skills Demonstrated

- Real-time Communication (RTC) development with Livekit.
- Integration of AI services: OpenAI, Deepgram, Cartesia, AWS Bedrock.
- Dynamic, configuration-driven application architectures.
- API development and complex data flow management.





Conclusion

This project demonstrates expertise in delivering smart, engineered solutions for scalable, high-performance conversational AI.

