Artizence Systems LLP Technical Assessment for AI Developer / Full stack Engineer

---

🎙️ Project Overview: Real-Time AI Voice Orchestration System

A modular, web-based platform that allows users to create, customize, and deploy AI voice agents with low-latency, human-like interaction.

🏗️ System Architecture & Logic

- Orchestration Layer: Powered by Qwen 1.5B. It acts as the "Controller," handling intent classification, routing conversation flows, and making logic-based decisions.
- Conversational Layer: Utilizes lightweight LLMs (LLaMA-1B or Liquid-1.5B) to generate rapid text responses, ensuring the conversation feels fluid and natural.
- Voice Processing (STT/TTS): Integrated via high-performance APIs like Deepgram, Cartesia, or fal.ai to achieve sub-second latency for speech-to-text and text-to-speech.

💻 Technical Stack

Component	Technology
Backend (Core)	Python with Django REST Framework (Auth & Data persistence).
Backend (Streaming)	FastAPI for high-concurrency, real-time WebSocket handling.

Frontend	Next.js / React.js (Production) or Streamlit (Prototyping).
AI Agents code	Langchain , Langgraph
Models/APIs	External API consumption for scalability and modularity.
Testing	Postman for API documentation and workflow validation.

✨ Key Features

- Custom Agent Builder: Users can define unique agent personas via a custom system prompt and name.
- Real-Time Streaming: Direct browser-based voice calls using optimized streaming protocols.
- Modular Intelligence: A clean separation between the Orchestrator (logic) and the Responder (personality).
- Production-Ready UX: Simple, one-click interface to initiate calls and manage agent sessions.

Submission

- **Invite the repo code along with postman collection to akshat0098**
- **Timeline 5-7days from the date of given (7th feb 2026)**