Al-powered Voice Sales Agent

Project Overview:

This project focuses on building an AI-Powered Voice Sales Agent that can conduct automated voice conversations with potential customers to promote online courses. The system is designed to simulate a real phone call — capturing voice input from the user, interpreting it using AI and responding back with audio output all in real-time.

How It Works:

Voice Input:

The user (mock caller) speaks into the microphone. The input is captured as audio.

• Speech-to-Text (STT):

The audio is transcribed to text using OpenAl Whisper, known for its high accuracy even in noisy environments.

Language Processing (LLM):

The transcribed text is sent to GPT-4 (via OpenAl API) which acts as the brain of the agent. Based on the conversation stage and customer input, the model generates an appropriate response — including course recommendations or objection handling.

Conversation Memory:

Each user input and GPT-generated response is stored in the conversation history. This history is used in every interaction to maintain context and ensure the agent responds naturally and coherently across multiple turns.

• Text-to-Speech (TTS):

The GPT-generated response is converted back to voice using gTTS (Google Text-to-Speech).

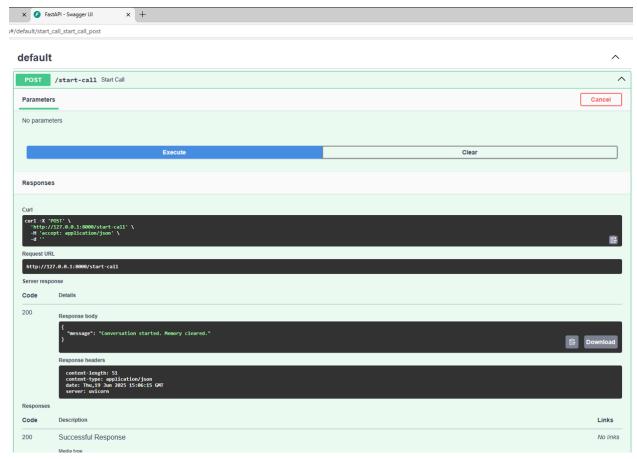
Audio Output:

The response is played back to the user as audio, completing the voice-based interaction loop.

Backend API (FastAPI):

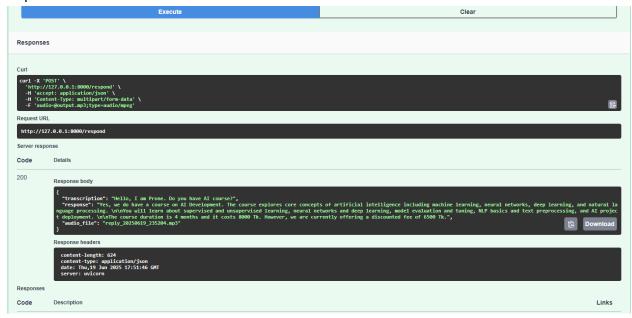
Three key endpoints have been developed:

- POST /start-call
 - Initializes a new conversation session.



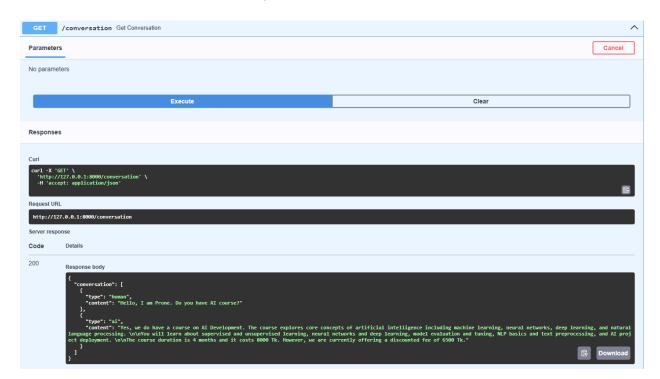
- POST /respond/
 - Accepts audio input, processes it through STT \rightarrow LLM \rightarrow TTS and returns an audio

response.



GET /conversation

- Retrieves the full conversation history..



Integration Components:

• LLM Integration:

Used OpenAI GPT-4 API known for its strong reasoning and context retention capabilities.

• Course Database:

A local courses json file is used to simulate a course database. The LLM can search through this structured data to offer targeted course suggestions.

Vector Search:

Implemented FAISS to support similarity-based retrieval from the course database — enhancing relevance in course pitching.

Conversation Flow Design:

❖ Introduction:

Bot welcomes the user and introduces the company and purpose of the call.

Qualification:

Ask a few questions to gauge user interest, background or career goals.

❖ Course Pitch:

Based on responses, the bot selects a relevant course and provides a short pitch with features and benefits.

Objection Handling:

Addresses common user concerns logically and politely, offering solutions or reassurances.

❖ Wrap-up:

Offers the user a follow-up or next steps (mocked).

Tech Stack Summary:

Language: PythonLLM: OpenAI GPT-4

Speech to text : OpenAl Whisper

Text to speech : gTTS (Google Text-to-Speech)

❖ Vector Database : FAISS

Framework : Langchain, FastAPI