Voice Bot Architecture Document

Michael Smith

Overview

This project implements an automated voice bot that simulates a patient calling Pretty Good AI's test line to evaluate conversational quality and identify bugs. The system places outbound calls using Twilio's Voice API and interacts with the AI agent in real time through webhook endpoints served by a FastAPI application. Each conversation is recorded, transcribed using OpenAI Whisper, and stored as a structured transcript. The bot generates patient responses dynamically using a large language model, allowing realistic and varied test scenarios.

System Architecture

The architecture follows a simple event-driven webhook model. When a call is initiated, Twilio triggers the `/voice` endpoint hosted via FastAPI and exposed publicly through ngrok. Incoming audio recordings are downloaded securely using Twilio authentication credentials and transcribed locally using Whisper. The transcribed agent utterance is then passed to a scenario-driven LLM module, which generates the next patient response based on the conversation state. Conversation state like the turn count, scenario, and transcript history is stored in memory per call session to prevent infinite loops.

When the call ends, a `/call_status` webhook finalizes logging and saves a complete transcript and full-call recording. This design intentionally prioritizes clarity and reliability over complexity. The system uses minimal infrastructure components, local state management, and explicit webhook handling to ensure reproducibility, transparency, and easy debugging. Architectural choices were guided by the assessment's emphasis on working code, real calls, and clear reasoning rather than production-scale deployment.