

# Candidate Assignment: Multilingual Voice Bot (Hindi + English)

## Objective

Build a **working prototype** of a Voice Bot where:

1. user calls / starts a voice session
2. bot validates user (OTP or DB-based)
3. user asks multiple **multilingual (Hindi+English)** questions (including code-mixed)
4. bot answers smoothly with **human-like voice behavior** (filler words + natural pacing)

## What you must implement (minimum scope)

### A) Voice Conversation Core

- **Real-time voice interaction**
- **Hindi + English support**
- **Code-mixed inputs** like:
  - "Hi, mera policy status check kar do"
  - "OTP aaya nahi, can you resend?"

### B) Validation Mechanism (choose ONE)

#### Option 1: OTP-based

- Bot asks for mobile number
- Generates OTP (mocked is fine)
- User speaks OTP
- Bot validates and proceeds

#### Option 2: Database-based

- Simple in-memory DB (JSON / SQLite) with 5 dummy users
- Bot asks for (mobile + DOB / last 4 digits)
- Validates and proceeds

### C) Multilingual Q&A

- At least **10 FAQs** (mix of Hindi/English)
- Q&A can be hardcoded or retrieved from a JSON file
- Must handle:
  - repeated question
  - user changing language mid-question
  - user interrupting the bot (barge-in)

# Feature Evaluation Requirements (what you'll be judged on)

## 1) Low Latency (must show measurement)

- Measure:
  - **User stops speaking → bot starts speaking**
- Print logs like:
  - ASR\_end\_time, LLM\_start\_time, TTS\_start\_time, Audio\_first\_byte\_time
- Provide a small report:
  - Average latency, P95 latency (even if rough)

## 2) Barge-in (must demonstrate)

- When bot is speaking, user interrupts with:
  - "Wait wait, ek minute..." / "Stop—my question is different"
- Bot must:
  - **stop TTS**
  - **listen immediately**
  - respond to the interruption

## 3) Background Voice Cancellation (basic demo acceptable)

- Candidate must demonstrate that the bot still works when:
  - background audio is present (TV noise / fan / cafe noise)
- Acceptable implementations:
  - noise suppression setting in pipeline, or
  - VAD tuning + denoise module
- Must provide **one demo recording** with noise on.

## 4) Voice Printing (speaker verification)

- Implement a **voiceprint enrollment + verification** step.
- Flow:
  - Enroll: "Please repeat this phrase: 'Mera naam verify kijiye'"
  - Verify later: user repeats a short phrase
- If verification fails:
  - fallback to OTP (or ask security question)

(This can be a lightweight speaker embedding approach; doesn't need to be enterprise-grade.)

## 5) Filler Words + Human-like behaviors

Bot voice should include:

- Natural micro-pauses
- Fillers occasionally:
  - English: "Hmm...", "Let me check..."
  - Hindi: "Haan...", "Ek second..."
- But should not overdo it.
- Must follow rules:
  - Fillers ONLY when "thinking / fetching"
  - Never while reading OTP or sensitive data

This assignment is designed to evaluate your ability to build a **real-time, human-like multilingual Voice AI system**. We are not expecting a production-grade solution, but a **working prototype** that demonstrates engineering clarity, conversational design, and system thinking.

Please ensure your implementation is **self-contained** and runnable on a local machine or cloud environment. You do **not** need to send recorded demos in advance.

Instead, **you will be required to demonstrate the working solution live during the interview**, including the validation flow, multilingual conversations, barge-in behavior, and latency measurement.

If any assumptions are made, document them clearly.

We look forward to seeing your solution and discussing your approach during the interview.