Vaiu AI Software Developer Internship Assignment

Assignment Overview

Duration: 1-2 days
Submission: GitHub repository link
Objective: Build a functional voice-enabled AI agent with custom integrations

This assignment will help us evaluate your:

- Full-stack development skills (MERN stack)
- Ability to integrate third-party APIs
- Problem-solving and code quality
- Understanding of voice/conversational AI concepts

Use Case: Restaurant Booking Voice Agent

Build an intelligent voice agent that helps users book restaurant tables through natural conversation. The agent should be able to:

1. **Greet the user** and understand their booking intent
2. **Collect booking information** through voice:
   - Number of guests
   - Preferred date and time
   - Cuisine preference (Italian, Chinese, Indian, etc.)
   - Special requests (birthday, anniversary, dietary restrictions)
3. **Fetch real-time weather** for the booking date and suggest indoor/outdoor seating
4. **Confirm booking details** via voice
5. **Store booking** in a database

Technical Requirements

Option 1: Use LiveKit

LiveKit Agents Framework - Perfect for voice agents!

Resources:

- GitHub: https://github.com/livekit-examples
- Docs: https://docs.livekit.io/agents/
- Quick Start: https://docs.livekit.io/agents/quickstart/

Option 2: Alternative Frameworks (If you prefer)

- Vocode: https://github.com/vocodedev/vocode-python
- Pipecat: https://github.com/pipecat-ai/pipecat
- Web Speech API: Browser-based (simpler but less powerful)

Must Have (Core Requirements):

1. Voice Interaction

- Implement speech-to-text (user input)
- Implement text-to-speech (agent responses)
- Handle voice commands naturally

Suggested Libraries:

- Web Speech API (browser-based, free)
- Google Cloud Speech-to-Text / Text-to-Speech
- OpenAI Whisper API (speech-to-text)
- ElevenLabs (text-to-speech)

2. Backend API (Node.js + Express)

Create RESTful endpoints:

```
POST /api/bookings          - Create a new booking

GET  /api/bookings          - Get all bookings

GET  /api/bookings/:id      - Get specific booking

DELETE /api/bookings/:id    - Cancel booking
```

3. Database (MongoDB)

Design a schema for restaurant bookings:

```
{

  bookingId: String,

  customerName: String,

  numberOfGuests: Number,

  bookingDate: Date,

  bookingTime: String,

  cuisinePreference: String,

  specialRequests: String,
  weatherInfo: Object,
  seatingPreference: String, // indoor/outdoor
  status: String, // confirmed, pending, cancelled
  createdAt: Date
}
```

4. Custom Integration: Weather API

Required Feature:

- Integrate a weather API (OpenWeatherMap, WeatherAPI.com, etc.)
- When user provides booking date, fetch weather forecast
- Agent should verbally suggest seating:
  - "The weather looks great on [date]! Would you prefer outdoor seating?"
  - "It might rain on [date]. I'd recommend our cozy indoor area."

Implementation:

```
// Example weather integration
async function getWeatherForDate(date, location) {
  // Fetch weather forecast
  const weather = await weatherAPI.getForecast(date, location);

  // Generate voice response based on weather
  if (weather.condition === 'sunny') {
    return "Perfect weather for outdoor dining!";
  } else if (weather.condition === 'rainy') {
    return "Looks like rain. Indoor seating would be better.";
  }
}
```

What NOT to Do:

Do NOT use random LLM responses for weather information.

Agent MUST fetch real data from YOUR backend API and use that data in responses.

Sample Resources & References

Voice Agent GitHub Repositories (Reference):

1. **Basic Voice Assistant**
    - https://github.com/linto-ai/linto-web-client
    - Simple web-based voice interface
2. **Speech Recognition Examples**
    - https://github.com/webrtc/samples (Web Speech API examples)
    - https://github.com/openai/whisper (Whisper API integration)

3. **Conversational AI Patterns**
    - https://github.com/microsoft/BotFramework-WebChat
    - https://github.com/botpress/botpress
4. **Full-Stack Voice App Example**
    - https://github.com/mozilla/DeepSpeech-examples
    - https://github.com/JustinTRoss/voice-assistant-mern

Helpful Documentation:

- **Web Speech API**: https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API
- **OpenAI Whisper**: https://platform.openai.com/docs/guides/speech-to-text
- **ElevenLabs API**: https://docs.elevenlabs.io/
- **OpenWeatherMap API**: https://openweathermap.org/api
- **Socket.io (for real-time)**: https://socket.io/docs/

Bonus Features (Optional)

Implement any of these to stand out:

1. **Natural Language Processing**
    - Use OpenAI GPT API for intelligent conversation
    - Handle variations in user input ("I want to book a table" vs "Table for 2")
2. **Multi-language Support**
    - Support Hindi + English voice commands
    - Code-switching capability
    -

3. **Calendar Integration**
   - Show available time slots
   - Prevent double bookings
4. **SMS/Email Confirmation**
   - Send booking confirmation via Twilio SMS
   - Email receipt using NodeMailer
5. **Admin Dashboard**
   - View all bookings
   - Analytics (peak hours, popular cuisines)
   - Export reports

Deliverables:

1. Working voice agent (can run locally)
2. Clean code with comments
3. README with setup instructions
4. Screen recording (2-3 min) showing it working

Deliverables Checklist

Before submitting, ensure you have:

- GitHub Repository (public)
  - Clean code structure
  - Comments explaining key functions
  - Include a README that clearly explains your approach.