Project Title: Conversational IVR Modernization Framework – Group 1

Name: Rhitam Mondal

Use Case: Patient Feedback IVR for Hospitals

Problem Statement:

This project aims to modernize traditional hospital IVR (Interactive Voice Response) systems built on VoiceXML (VXML) by integrating them with modern Conversational AI platforms such as ACS and BAP Services. The method will minimize rework and allow these systems to offer conversational interfaces by extending and reusing legacy components. The strategy focuses on lowering the technical complexity of switching from legacy IVRs to AI-driven workflows while also boosting usability and user experience.

Introduction:

Traditional hospital IVR systems are primarily built on menu-driven VoiceXML architectures, where patients must navigate through multiple "press 1, press 2" options to complete simple tasks such as giving feedback. While these systems were effective in the past, they often create frustration, especially for elderly patients, people with limited technical literacy or those under stress after medical visits. Remembering keypad options, listening to long menus and navigating complex call trees can lead to call drop-offs and poor feedback collection rates.

Conversational AI offers a more natural and human-like interaction model. Instead of pressing keys, patients can simply speak in their own words, such as "I want to give feedback about my appointment" or "The waiting time was too long."

Additionally, modern conversational systems can understand intent, handle flexible responses and guide users dynamically, unlike rigid legacy IVR menus. By integrating Conversational AI with existing VXML-based systems, hospitals can enhance user experience, increase feedback participation and modernize their communication infrastructure without completely replacing their legacy systems.

Technical Backlogs in Legacy IVR Systems:

Legacy IVR systems built on VoiceXML are primarily designed around fixed, menu-driven interactions. These systems rely heavily on DTMF inputs and predefined call flows, which makes them time consuming for users. Patients must listen to long prompts and remember keypad options, often navigating through multiple layers of menus to complete even simple tasks. This rigid structure increases call duration, leads to user frustration and results in higher call abandonment rates.

From a technical perspective, legacy IVRs are difficult to modify and scale. Any change in business logic, menu structure or feedback process requires updates to VXML scripts and extensive testing, which slows down deployment cycles. Integration with modern APIs or cloud-based services is also complex, as many legacy systems use outdated protocols and tightly coupled architectures. This creates compatibility issues when trying to add new features such as speech recognition, analytics or personalization.

Another major backlog is limited data intelligence. Traditional IVRs only capture structured inputs like key presses, which restricts the depth of feedback collected. They lack natural language understanding, making it impossible to capture detailed patient experiences. These technical limitations, combined with maintenance overhead and scalability challenges, make legacy IVR systems inefficient, motivating the transition to a conversational AI–based solution.

Existing and Proposed Systems:

Existing System:

- Menu-driven VXML IVR
- DTMF-based navigation
- Limited speech support
- High user effort

Proposed System:

- Conversational AI-driven IVR
- Natural language voice interaction
- Integration using ACS and BAP
- Middleware connecting legacy backend systems

Key Components Enabling a Versatile Conversational IVR:

To overcome the limitations of legacy IVR systems, the proposed solution introduces modern conversational technologies such as Azure Communication Services (ACS) and the Bot Application Platform (BAP). These components transform the rigid, menu-driven structure of traditional IVRs into a flexible, intelligent and user-friendly conversational system.

ACS acts as the communication backbone of the system. It manages incoming calls, handles call control and provides real-time speech-to-text and text-to-speech capabilities. This allows patients to interact using natural voice input instead of keypad selections.

BAP uses natural language understanding to interpret user intent, manage dialogue flow and generate appropriate responses. Unlike legacy IVR systems that follow fixed scripts, BAP enables dynamic conversations that adapt to the user's responses. It can handle varied phrases, clarify ambiguous inputs and guide patients smoothly through the feedback process.

Together, ACS and BAP create a scalable, flexible and intelligent IVR environment, addressing the rigidity, limited input methods and poor user experience associated with traditional VXML-based systems.

Technical Incorporation of ACS and BAP in the Conversational IVR System:

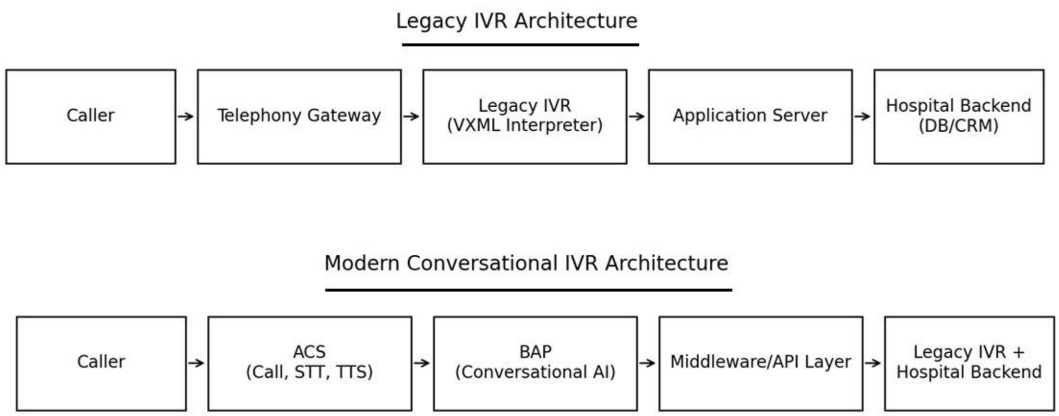
In the proposed conversational IVR architecture, Azure Communication Services (ACS) and the Bot Application Platform (BAP) are integrated through a middleware layer that bridges modern AI services with the existing legacy IVR and backend systems. This approach ensures that conversational capabilities can be introduced without rewriting the entire legacy infrastructure.

ACS functions as the telephony and media processing layer. When a patient places a call, ACS handles call initiation, session management and audio streaming. It converts the patient's speech into text using speech-to-text (STT) services and forwards the recognized text to the conversational engine. ACS also converts the system's textual responses into natural-sounding speech using text-to-speech (TTS) before playing them back to the caller.

BAP operates as the conversational intelligence layer. It receives the transcribed user input from ACS, performs natural language understanding to detect user intent and determines the appropriate dialogue flow. Based on the identified intent, BAP sends structured requests to the middleware layer.

The middleware acts as an integration bridge. It maps conversational intents to existing VXML flows or backend APIs, executes the required business logic and returns responses to BAP. The response then travels back through ACS, which delivers it to the caller as synthesized speech, completing the conversational loop.

Architectures:



Conclusion:

The conversational AI–based IVR system offers significant advantages over traditional legacy IVR systems for both hospitals and patients. By incorporating modern components such as ACS and the BAP, the proposed system transforms rigid, menu-driven interactions into natural, voice-based conversations. Patients no longer have to explore complicated menus or remember keypad selections. Rather, they are free to communicate, which makes the system more usable by older users, people with visual impairments and others who are not very tech-savvy.

For hospitals, this approach enables the collection of richer and more detailed patient feedback. Conversational AI can capture open-ended responses, identify sentiments and detect recurring issues such as long waiting times, staff behavior or service delays. This helps hospital management gain deeper insights into patient experiences and take corrective actions more effectively.

From a technical standpoint, the integration of ACS, BAP and a middleware layer ensures scalability, flexibility and easier system upgrades without replacing the entire legacy infrastructure. The system becomes easier to maintain, faster to adapt to new requirements and more cost-efficient in the long run.