Build and modernize AI apps – Call Center

# Key Technologies

* Azure AI Services
* Azure SQL Database
* Azure CosmosDB
* Azure Static Web Apps
* Azure App Services
* Azure Communication Services

# Main Goals

The primary objective is to feature a Call Center scenario to enhance an existing custom Customer Service Agent client application. The solution is a virtual assistant that delivers context-sensitive cues to the agent based on the discussion happening in real-time on the call. The custom client application is enhanced to leverage the speech-to-text capabilities within Azure AI Services and the integration of Azure OpenAI using a ChatGPT model augmented with private internal organizational data.

# Overview

Contoso Suites is a luxury hotel chain owning properties in seven countries, including the United States, Costa Rica, French Guiana, and several Caribbean islands. Contoso Suites takes pride in providing a personal touch by having their PSTN calls answered by a real human being. Their VP of Customer Service, Olivia Wilson, has read up on large language models like ChatGPT and is interested in using this sort of technology to serve as a virtual assistant to assist call center agents by providing context-aware cues based on the discussion on the call as well as integrating with in-house systems to automate common functions such as account lookup, retrieving customer history, and booking a reservation. Furthermore, the virtual assistant should be able to respond to natural language questioning from the agent and provide meaningful responses. Finally, a great deal of time is currently being spent by agents summarizing the call, it would be nice if this could be automated.

Key contacts:

* Olivia Wilson, VP of Customer Service
* Wesley Brooks, VP of Engineering
* Kiana Anderson, Application Architect
* Omar Bennett, Data Scientist

Current state:

* Internal systems, each provides an API for retrieving and storing data. Customer service agents currently access the web front end of each system as needed while on calls with customers.
  + CIS – customer information system containing contact information, call history with summary, and reviews left.
  + BOOKINGS - Reservation system containing room availability information, providing booking functionality, and the ability to retrieve booking history.
  + FAQ – knowledge base of frequently asked questions along with answers.
* Customer Service Agent Web application front end
  + React
  + Running as an Azure Static Web App
* Customer Service Agent REST API back end
  + ASP.NET MVC using .NET Framework 6
  + Running in Azure App Service
* Application database
  + Azure SQL Database, 4 vCPU, 100 GB of data
  + Contains information on each call, including agent information, customer information, time spent, and call transcript.
  + Azure Cosmos DB to store Chat History

Trying to accomplish:

* Interpret real-time audio as prompts into a custom large language model that will serve customer service agents with appropriate cues to assist in the discussion.
* Expose a chat-based interface for customer service agents to ask the large language model questions and receive context-aware responses.
* Watch for prompts that indicate a common action, such as account lookup or checking room availability. Automate these actions through integration with the CIS or BOOKING systems APIs and return appropriate scripts for the call center agent to read to the customer.
* Use the large language model to generate a call summary and send it to the CIS system for storage. This is currently accomplished manually by the agent after the call.
* During and after the call, the large language model needs to be able to answer questions about the call, such as what the general subject of the call is, and if the call is compliant (the customer was informed the call is being recorded).

# Customer needs

* Existing developers are versed in C#, JavaScript, and Python languages. The solution should be implemented using one or more of these languages.
* Contoso Suites would like to enhance the large language model responses by providing private internal data.
* The solution needs to respond promptly due to the real-time nature of a customer call.
* The solution must ensure the agent doesn’t need to leave the Customer Service Agent Web interface to access data from an external system.
* Upon completion of a customer call, the conversation should be summarized automatically and sent to the CIS application.
* The large language model needs to be able to answer natural language questions about the call.

# Customer objections

* Contoso Suites has an extensive collection of frequently asked questions and their answers in their internal FAQ system. Large language models do not have this information as it is private. Can we combine public and private information into the same model easily?
* Contoso Suites already has automated common actions using APIs implemented atop various internal systems. These actions are things like customer lookup and room availability checks. The solution must identify when these actions are needed based on the incoming call transcript, execute the action, and display the return values as a formatted script for the Customer Service Agent within the Customer Service Agent Web application. Is this possible? Currently, Customer Service Agents access multiple web applications to retrieve information.
* Contoso Suites would like to leverage historical transcripts to improve the customer service agent cues that the large language model delivers to the agents in the web application. How can this be accomplished?

# Current state

A diagram of a customer service agent web application architecture.


Complex scenario hands-on-lab

# Description:

Contoso Suites wants to integrate Azure OpenAI functionality into their Customer Service Agent Web application. They wish to use real-time incoming call audio as prompts for a custom large language model enhanced with the internal FAQ system knowledge base data. The model should provide timely cues to the Customer Service Agent based on the customer's questions and be on the lookout for prompts that indicate a common action, such as account lookup or checking room availability, and execute those actions accordingly. During the call, the agent needs to be able ask natural language questions about the call and receive context-aware responses. Upon completion of a call, the details of the call should be summarized by the large language model and sent to the CIS system.

At the end of this lab, you will be better able to customize ChatGPT responses by augmenting the model with private internal data and automating the execution of external function calls to perform actions such as data retrieval. Furthermore, you will learn to integrate real-time speech-to-text functionality using Azure AI Services Speech service and use this data as prompts into the customized large language model. Finally, leverage the text summarization capabilities of Azure OpenAI to summarize the call transcript quickly.

# Lab outline:

* Create a Customer Service Agent virtual assistant with Azure AI Services and Azure OpenAI Services
  + [Abstract and learning objectives](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#abstract-and-learning-objectives)
  + [Overview](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#overview)
  + [Solution architecture](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#solution-architecture)
  + [Requirements](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#requirements)
  + [Before the hands-on lab](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#before-the-hands-on-lab)
  + Exercise 1: Enhance the ChatGPT model by adding internal FAQ data
    - Task 1: Prepare an FAQ dataset to augment the ChatGPT model
    - Task 2: Add the FAQ data to Azure OpenAI
    - Task 3: Test FAQ completions using the ChatGPT playground
  + [Exercise](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#exercise-1-sql-database-migration) 2: Implement Function Calling to integrate with external APIs
    - [Task 1:](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#task-1-install-data-migration-assistant) Implement the customer account information lookup function
    - Task 2: Define the function for the Azure OpenAI Chat Completions API
    - [Task 3:](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#task-4-migrate-database-to-azure-sql-mi) Test function calling using the ChatGPT playground
  + [Exercise](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#exercise-3-azure-arc-enable-on-premises-vm) 3: Implement audio transcription using Azure AI Services Speech Service
    - Task 1: Implement real-time speech-to-text in the Customer Service Agent Web application
    - Task 2: Test speech-to-text using an audio file (optional)
    - Task 3: Test speech-to-text using a microphone (optional)
  + Exercise 4: Provide live audio transcription to the ChatGPT model
    - Task 1: Integrate ChatGPT with the Customer Service Agent Web application
    - Task 2: Test the FAQ completions in the web application
    - Task 3: Test the prompt to execute the customer account information lookup function
    - Task 4: Determine if the current call is compliant using natural language prompts
    - Task 5: Determine the main subject of the call using custom entity extraction
  + Exercise 5: Generate a call summary and send it to the CIS system API
    - Task 1: Leverage the Completions API to generate a call summary
    - Task 2: Send the call summary to the CIS system API for storage
  + [After the hands-on lab](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#after-the-hands-on-lab)
    - [Task 1: Delete the resource group to remove the lab environment](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#task-1-delete-resource-group-to-remove-the-lab-environment)

Simple scenario click-through-lab/demo

# Description:

Contoso Suites would like to enhance ChatGPT responses by augmenting the model with private FAQ data. They would also like to automate common actions, such as account information lookup in an external system, without ever having to leave the chat interface. Contoso Suites would also like to automate the generation of a call summary using the call transcript. Summarization should include main subject extraction as well as if the call is in compliance with the recording policy.

At the end of this lab/demo, you will be better able to explain to your customers how to enhance ChatGPT model responses by augmenting with customer data, automating data retrieval from external systems using function calling, and leveraging the text summarization feature of the Completions API.

# Lab/demo outline:

* Enable customer success by augmenting a ChatGPT model with customer data and integrating external system data retrieval with function calling.
  + [Abstract and learning objectives](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#abstract-and-learning-objectives)
  + [Overview](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#overview)
  + [Solution architecture](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#solution-architecture)
  + [Requirements](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#requirements)
  + Exercise 1: Enhance the ChatGPT model by adding internal FAQ data
    - Task 1: Prepare an FAQ dataset to augment the ChatGPT model
    - Task 2: Add the FAQ data to Azure OpenAI
    - Task 3: Test FAQ completions using the ChatGPT playground
  + [Exercise](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#exercise-1-sql-database-migration) 2: Implement Function Calling to integrate with external APIs
    - [Task 1:](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#task-1-install-data-migration-assistant) Implement the customer account information lookup function
    - Task 2: Define the function for the Azure OpenAI Chat Completions API
    - [Task 3:](https://github.com/microsoft/MCW-Building-the-business-migration-case-with-Windows-Server-and-SQL-Server/blob/main/Hands-on%20lab/HOL%20step-by-step%20-%20Building%20the%20business%20migration%20case%20with%20Windows%20Server%20and%20SQL%20Server.md#task-4-migrate-database-to-azure-sql-mi) Test function calling using the ChatGPT playground
  + Exercise 3: Generate a call summary and send it to the CIS system API
    - Task 1: Leverage the Completions API to generate a call summary using the ChatGPT playground
    - Task 2: Determine if the current call is compliant using natural language prompts
    - Task 3: Determine the main subject of the call using custom entity extraction

Helpful References:

[Use your own data with Azure OpenAI Service - Azure OpenAI | Microsoft Learn](https://learn.microsoft.com/en-us/azure/ai-services/openai/use-your-data-quickstart?tabs=command-line&pivots=programming-language-studio)

[How to use function calling with Azure OpenAI Service - Azure OpenAI Service | Microsoft Learn](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/function-calling)

[Azure OpenAI speech to speech chat - Speech service - Azure AI services | Microsoft Learn](https://learn.microsoft.com/en-us/azure/ai-services/speech-service/openai-speech?context=%2Fazure%2Fcognitive-services%2Fopenai%2Fcontext%2Fcontext&tabs=windows&pivots=programming-language-csharp)