Azure OpenAI Virtual Assistant

OUTLINE DOCUMENTATION

LAST UPDATED
23 May 2024

Contents

Acronyms and Abbreviations

The following abbreviations are commonly used throughout this document.

Table 1, Acronyms and Abbreviations

Abridged	Meaning
ACL	Access Control Lists
API	Application Programming Interface
AI	Artificial Intelligence
C5	C5 Alliance
CSS	Cascading Style Sheets
CI	Channel Islands
DB	Database
GPT	Generative Pre-trained Transformer
LLM	Large Language Model
Q&A	Question and Answer
UI	User Interface
UX	User Experience

1 Overview

The following sections outline the overall architecture of the GPT powered virtual assistant, or 'chatbot,' that C5 Alliance have deployed within your Microsoft Azure tenant. Detail will be given on the Azure resources involved, permissions required to interface with and make changes to the assistant, and the general process-flow that the solution follows.

As displayed in the diagram in section 2, the solution is comprised of two distinct processes: the knowledge-base build process and main Q&A process.

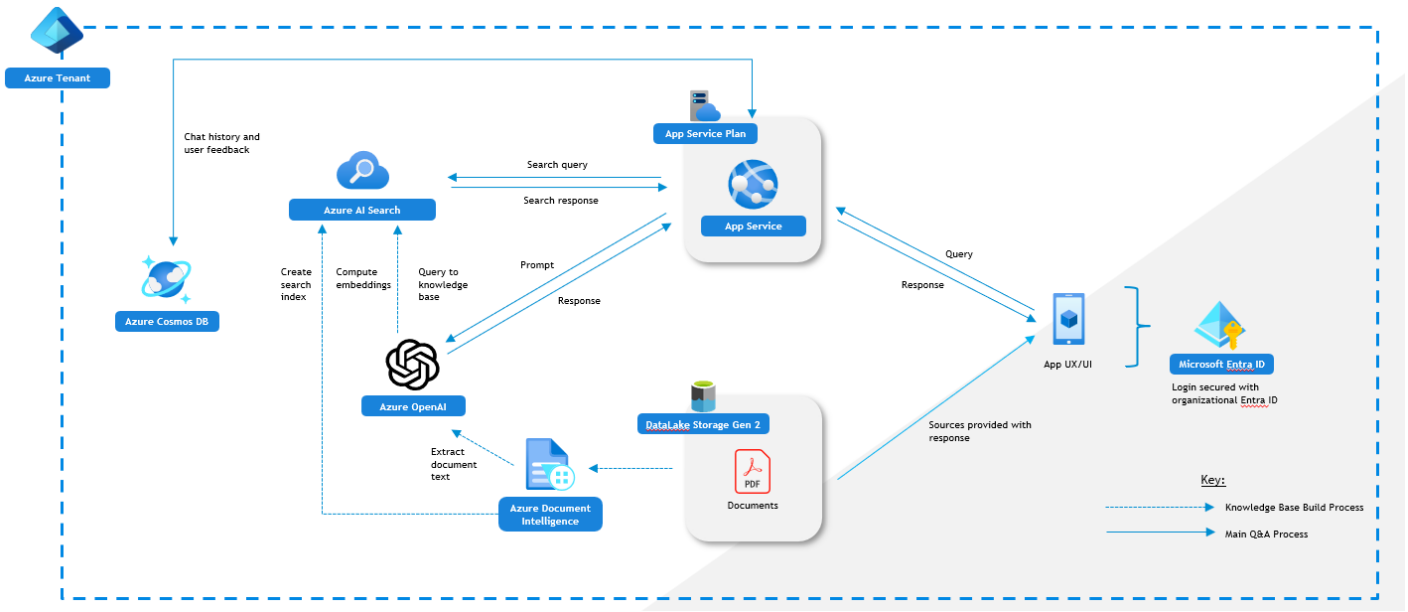
- Knowledge-base building is the process of document preparation and ingestion to create an AI search index that the main Q&A process can query against. It is this process that tailors the answers given by the virtual assistant to the relevant context within your organisation.
- In contrast, the main Q&A process is the normal use-case by which the virtual assistant operates – using OpenAI's GPT models to provide natural-language, informed answers to user queries using content it can access via the search index.

Security is grouped broadly into three categories: App-level, Assistant-level and Document-level.

- App-level security is controlled by organisational Microsoft Entra ID, meaning someone external to your organisation cannot get past the login page
- Assistant-level security is assigned in the Azure Cosmos DB, giving an administrator user the ability to grant access to an assistant to subsets of users, rather than the entire organisation. This can be in the form of individual user IDs or the IDs of entire Azure AD groups.
- Document-level security is assigned within the storage account using access control lists (ACLs), and can be determined at a container, directory or blob-level. This ensures that users can only get answers from the assistant within the context of the documents that they themselves are permitted to access.

2 High Level Architecture

A visual representation of the high-level architecture of the solution is provided below. This will also be included as an appendix to this document.



All of the necessary resources have been deployed from code using Bicep scripts, and are situated in a dedicated resource group. The following sections will provide detailed information on each. The full list of resources is:

- App Service Resource
- App Service Plan
- Azure Cosmos DB
- Azure DataLake Storage Gen 2
- Azure Document Intelligence
- Azure OpenAI
- Azure AI Search

3 Resources

3.1.1 App Service

The virtual assistant is accessed through a web application hosted on an Azure App Service Plan. The application features a Python backend (3.11) utilising the Quart web framework, and a TypeScript frontend built with the ReactJS framework. Additionally, Tailwind CSS is utilized for efficient and responsive styling. The backend interacts with the below resources through their associated REST APIs, often through the use of relevant Microsoft SDK's.

3.1.2 App Service Plan

The app is hosted on a Linux app service plan with two instances for high availability. The pricing plan is P0v3, allowing for production-level workloads.

3.1.3 Azure AI Search

Azure AI Search is a cloud-based service designed for indexing and searching source material. It is used within the solution in tandem with Azure OpenAI to create and host an index of the documents to create a **knowledge-base**, enabling efficient query searches for information contained in the documents. This index is created from the output of the document intelligence service, as detailed in section 3.1.7. The exact fields that comprise the index will vary according to the exact requirements of your solution, but they will always include:

- A unique identifier of each section
- The plain text of the section
- The section's embedding (as discussed in 3.1.7)
- Document-level security for the section

3.1.4 Azure Cosmos DB

Cosmos DB is a managed, distributed database service utilising the NoSQL paradigm. It is commonly used in application development due to its scalability and low latency. Within the solution there are two containers – *assistants* and *conversations*. *Assistants* contains the configuration settings of the individual assistants (if there are multiple) including:

- Title
- Example questions (displayed on the web app home page)
- Assistant-level security

Conversations contains the full conversation history of all users with the assistants including:

- Conversation ID
- Message ID
- User ID
- Timestamp
- Assistant
- Question / Answer Content

3.1.5 Azure DataLake Storage Gen 2

The Azure DataLake Storage Gen 2, hereinafter referred to as the 'storage account,' is used to store the documents that make up the knowledge base. Within the storage account, there will be a container containing all directories related to the solution. Within those directories there is a 'documents' directory, within which you'll find a directory specific to each assistant (if there are multiple). Within these are the source documents that the assistant has access to, and that make up the knowledge-base that the assistant pulls from to answer questions. It also forms the repository from which the sources are provided alongside the generated response.

As previously mentioned, document-level security can be assigned at a directory or blob-level, meaning that subsets of users can be given access to only certain documents in the knowledge-base. This ensures that a particular user cannot get an answer from the assistant containing information that they themselves are not privy to. This is achieved through the management of ACLs. In a similar manner to assistant-level security, individual users or entire AD groups can be added to the ACL of a directory or blob. This information is then included as part of the search index. Thus, when document-level security is updated, or a new document is added, a re-index of the knowledge-base must be undertaken for this to reflect in the web app.

3.1.6 Azure Document Intelligence

Azure Document Intelligence, formally Form Recognizer, is an Azure AI service for automated extraction of raw data from documents. It is used within the solution as part of the knowledge-base build process. PDF documents uploaded to the storage account are passed to the document intelligence service, where the plain text is extracted. In this step, supplementary text is added to any visual or diagram that may require it. This text is then separated into logical 'sections', the exact length and composition of which has been determined during the development phase.

3.1.7 Azure OpenAI

Azure OpenAI Service provides access to OpenAI's state-of-the-art LLM offering, including GPT-3.5, GPT-4 and GPT-4o. These models are incredibly powerful for tasks involving natural language processing and semantic understanding. Within the solution, there may be one or several deployments of the OpenAI service, providing access to different models within the OpenAI ecosystem. This service is integral to the solution, and is used in several different ways:

1. During the knowledge-base build process, the plain text sections produced using the document intelligence service are passed to OpenAI's embeddings API, which computes a vector representation of the text. This vector embedding is then included, alongside the text itself, in the search index, enabling much more sophisticated semantic searching.
2. During the main Q&A experience, the user's question is reduced to an optimised key-word search query using the completions API. A semantic version of the query is also generated in the form of a vector representation using the embeddings API once again. These queries are then passed to the Azure AI Search Service, where both simple searching (based on the key-word query) and semantic searching (based on the embedding) are executed against the index, returning several results from the knowledge base.
3. To formulate the final answer, the question, AI Search results and the assistant's system message (an engineered statement that defines the way the assistant acts and interacts) are passed to the completions API, where the GPT model is leveraged to produce a probable, semantically-accurate and coherent response.

FOR MORE INFORMATION:

Luke Hand

+44 1534 633733

luke.hand@c5alliance.com

Published in the Channel Islands.

C5 Alliance Limited,
Windward House,
La Route de la Libération,
St Helier,
Jersey,
JE2 3BQ

www.c5alliance.com