

# 案例研究

情境檢視與分析

# 案例研究概覽

## 公司背景:

**MicroRetail 微零售有限公司** 總部位於台北市，擁有超過 200 間實體門市，是全台具代表性的零售通路之一。儘管名稱中有「micro」這個字，他們在零售市場的影響力卻不容小覷。近年來，MicroRetail 積極推動電商發展與 OMO ( 虛實整合 ) 策略，希望透過創新技術優化顧客互動體驗與營運效率。然而，目前仍有許多核心系統依賴地端資料中心運作。

# 案例研究概覽

## 現狀與公司未來策略：

資訊長 Daniel Liang 積極推動生成式 AI 的導入，他表示：「我們最怕的不是沒人來逛，而是人太多時 API 被點爆。現在真的需要雲端來分擔流量與運算壓力。」Daniel 相信，AI 的應用不僅可以提升服務韌性，更能帶動整體營運成長。

架構師 Tina 與強大的技術夥伴們，已著手啟動第一階段導入計畫，包括：

- 在官方 App 與網站導入 AI 驅動的**內容審核機制**，過濾不當評論與意見回饋
- 建立一個具備個人化推薦能力的 AI 聊天機器人，能與顧客互動對話，也能適時推薦商品

這項計畫的目標，是在財政年度第 4 季結束前，**將客服平均回應時間縮短 15%**，進一步提升顧客滿意度與服務效率。

面對母親節、年中慶等高峰檔期的巨大流量挑戰，MicroRetail 期望透過 AI 自動化客服與即時互動分析，讓 AI 接手處理大量重複性問題，使客服人員能專注於高價值的溝通與協助，全面升級服務品質。

最終，MicroRetail 的願景是整合來自實體門市、電商平台與行動 App 的顧客數據，打造真正個人化、即時且智慧的顧客體驗。透過 AI 與雲端的強大能量，這個「微小」品牌，正以智慧與數據在市場中閃耀發光。

# 客戶目標與需求

## 一、效能與速度 ( **Performance and Speed** )

- 回應速度需低於 2 秒，提升即時互動體驗。
- 系統須支援高效且即時的顧客互動，不影響網站流暢度。

## 二、個人化體驗 ( **Personalization** )

- 根據顧客偏好與購買紀錄進行推薦，點擊率提升目標為 20%。
- 對話式 AI 機器人須具備個人化推薦能力，強化顧客體驗。

## 三、可靠性與可擴展性 ( **Reliability and Scalability** )

- 系統須具備 99.99% 的可用性與低於 1 秒的延遲。
- 能夠支援大量互動的可擴展性架構。
- 基礎架構需具備容錯與災難復原能力。

## 四、內容審核 ( **Moderation** )

- 能自動審核評論與顧客回饋，避免出現不當語言、垃圾訊息或不適當內容。

# 客戶目標與需求

## 五、安全性與法規遵循 ( **Security and Compliance** )

- 整合私有端點、RBAC、SSO 與資料標註，達成零資料外洩並通過年度資安稽核。
- 符合 GDPR 等隱私法規，並進行季節性稽核。
- 遵循資安最佳實踐確保資料安全。

## 六、治理與成本控管 ( **Governance and Cost Management** )

- 建立治理模型管理 AI 解決方案。
- 成本控管目標為預算偏差控制在 5% 以內。

## 七、技術與基礎建設 ( **Technology and Infrastructure** )

- 支援商用與開源大型語言模型的調用與微調，並能靈活部署以達成商業目標。

## 八、訓練與支援 ( **Training and Support** )

- 團隊對 Azure 不熟悉，需要 Microsoft 合作夥伴協助設計架構、網路與資安配置並提供訓練。

# 客戶問題與疑慮

- 如何確保 AI 審核能有效過濾不當內容（如髒話、垃圾訊息），同時不影響正常評論？
- 這套 AI 解決方案將如何處理客戶資料的安全性與隱私需求？
- 新系統是否會影響我們現有系統的效能，特別是在高流量期間？
- 我們擔心所謂的「幻覺現象」，也就是 AI 聊天機器人可能會捏造產品資訊。我們可以怎麼預防這種情況？
- 要如何衡量 AI 回覆的品質與準確度，確保符合企業標準？
- 顧客回饋將如何被整合，用來持續提升 AI 的準確性？
- 網路與基礎建設需要怎樣更新，才能支援 AI 整合？
- 能否更主動處理資安問題？有沒有更好的威脅偵測與安全監控自動化方式？

# Lab outline

**Level - Easy**

**Exercise one**

Set up Azure AI Foundry

**Exercise two**

Perform prompt engineering

**Level - Medium**

**Exercise Three**

Set up Azure prompt flow

**Exercise four**

Deploy a chatbot to a web app

**Exercise five**

Content moderation

# Exercise one

Set up Azure AI Foundry

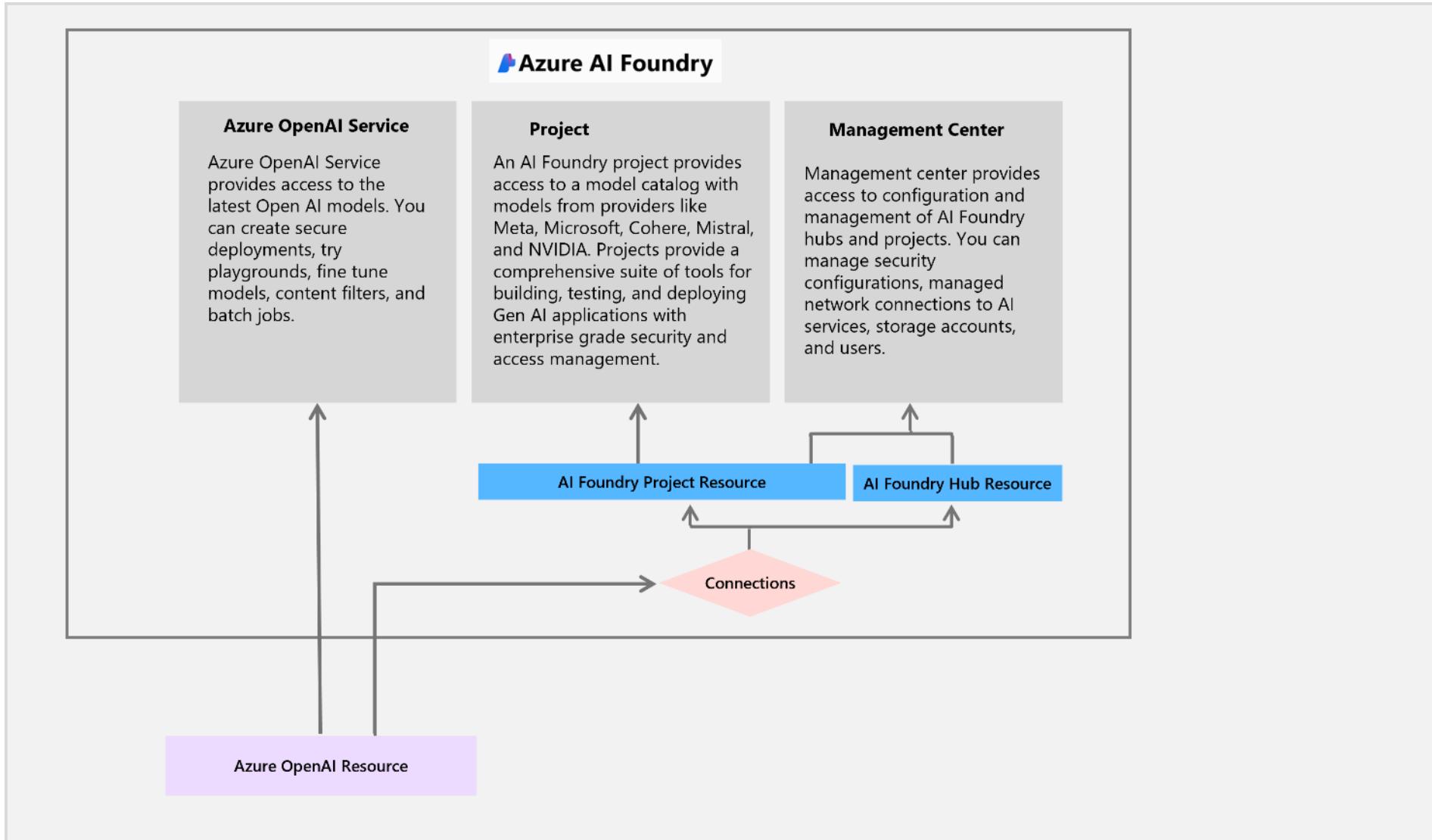
# Introduction: Set up Azure AI Foundry

MicroRetail aims to enhance its user experience with a chatbot, ensuring a safer and more engaging environment for customers. Building a chat bot begins with enabling an AI Foundry, creating a Hub, and managing permissions.

## After completing this exercise, you'll be able to:

-  Set up an Azure AI Foundry hub to manage AI projects and services.
-  Deploy AI models essential for chat-based applications.
-  Create a search service to enhance chatbot capabilities.

# Exercise one concepts



# Task 01: Set up a hub and project in Azure AI Foundry

## Introduction:

To support **MicroRetail**'s initiative for AI-driven customer engagement, the first step is to establish a structured environment where AI projects can be built and managed. By creating a hub in Azure AI Foundry, MicroRetail can centralize AI models, configurations, and data integrations, ensuring scalability and efficiency for future AI implementations.

## Description:

In this task, you'll create a hub within Azure AI Foundry to host your projects and resources. The hub serves as the foundation for deploying AI solutions and integrating AI services that MicroRetail will use to enhance customer engagement. After setting up the hub, you will also create a project within it to organize and manage AI-related workflows effectively.

## Success Criteria:

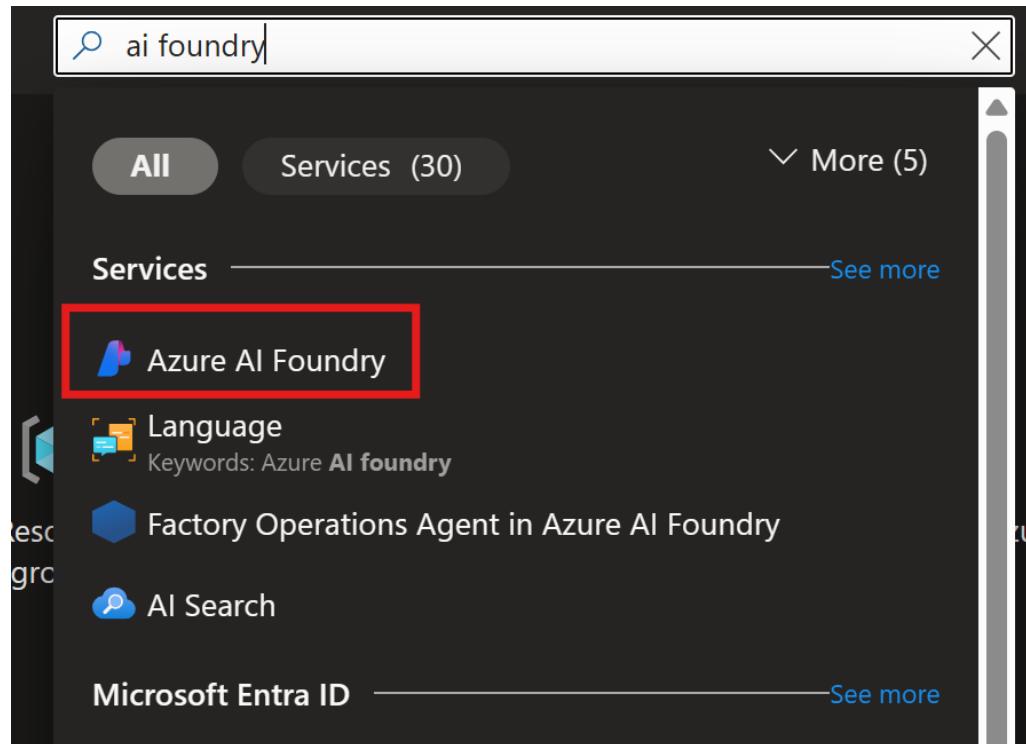
- The Azure AI Foundry hub has been created successfully.
- A new project has been set up within the hub.

# 1-1. Set up a hub and project in Azure AI Foundry

## Step 01: Create hub in Azure AI Foundry

1. Open the Microsoft Edge browser, go to the [Azure Portal site](#), and sign in with your credentials.

2. Once signed in to the portal, select the search bar at the top. Search for **Azure AI** and then select **Azure AI Foundry**.



# 1-1. Set up a hub and project in Azure AI Foundry

## Step 01: Create hub in Azure AI Foundry

3. From Azure AI Foundry, select **+ Create** and then select **Hub**.

The screenshot shows the Microsoft Azure (Preview) interface for AI Foundry. The left sidebar has a red box around the 'AI Hubs' option under 'Use with AI Foundry'. The main area shows a list of resources, with a red box highlighting the 'Hub' item in the 'Project' section. Below it is a table listing eight resources: four Azure AI projects and four Azure AI hubs, all located in East US.

Microsoft Azure (Preview) Search resources, services, and docs (G+) Copilot Home > AI Foundry AI Foundry | AI Hubs Microsoft Non-Production

Search Create Manage view Refresh Export to CSV Open query Assign tags Group by none

**Project**  
Collaborate, organize, and track work to build AI apps.  
[features may be missing. Click here to access the old experience.](#)

**Hub**  
Grouping container for projects. Provides security, connectivity, and compute management.

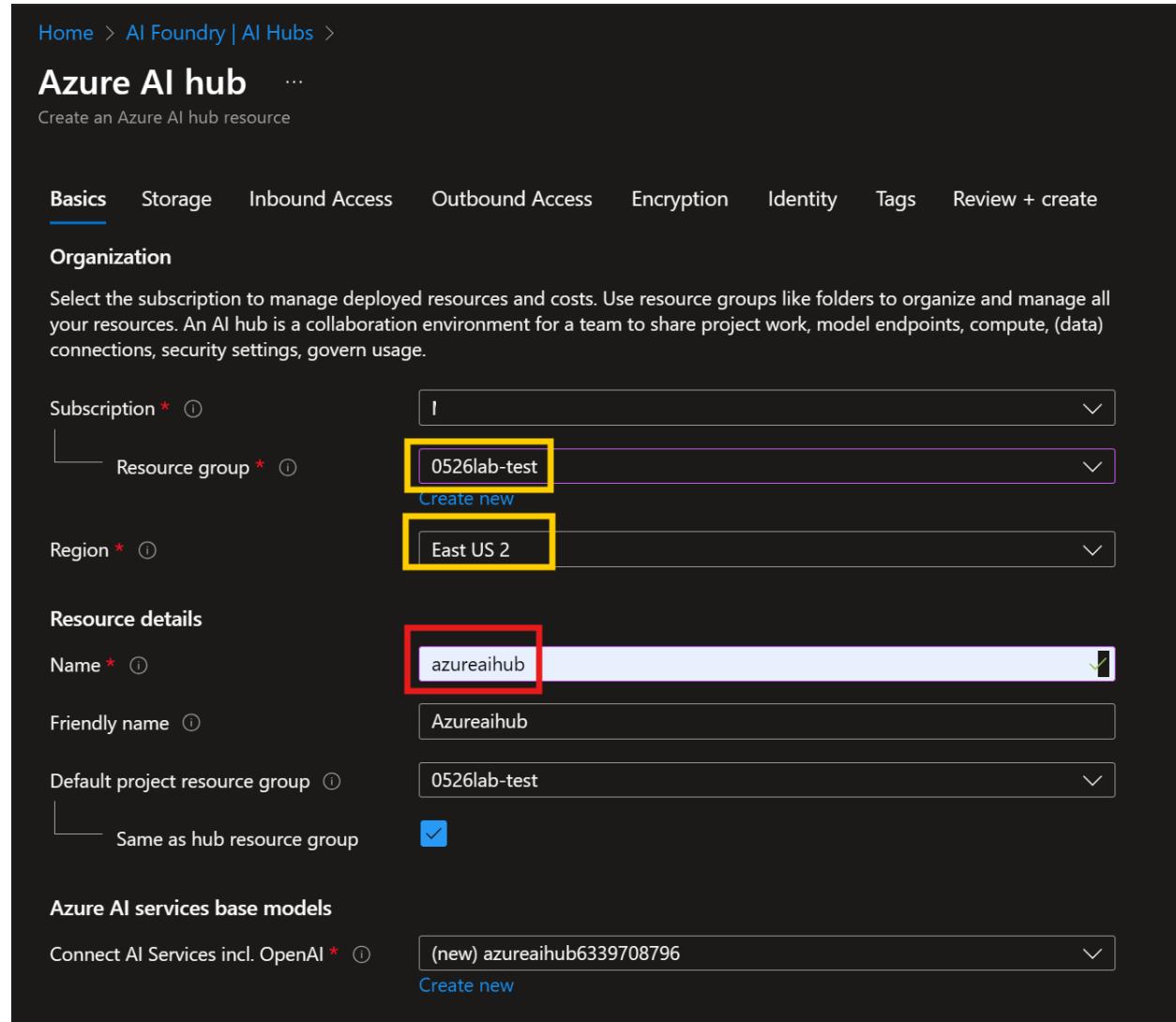
| Group | Type             | Location  | Subscription |
|-------|------------------|-----------|--------------|
|       | Azure AI project | East US   |              |
|       | Azure AI hub     | East US 2 |              |
|       | Azure AI project | East US 2 |              |
|       | Azure AI hub     | East US   |              |
|       | Azure AI hub     | East US   |              |
|       | Azure AI project | East US   |              |
|       | Azure AI hub     | East US   |              |
|       | Azure AI project | East US   |              |

Showing 1 - 8 of 8. Display count: 10 Give feedback

# 1-1. Set up a hub and project in Azure AI Foundry

## Step 01: Create hub in Azure AI Foundry

- On the Azure AI hub page, select your **resource group and region** (these may differ from the screenshot). Name the hub **azureaihub** and leave the default setting to create a new AI service model.



5. Select **Review + create**, then select **Create**.

6. Once the deployment is complete, select **Go to resource**.

# Choice of region

Basics Storage Networking Encryption Identity Tags Review + create

**Organization**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources. An AI hub is a collaboration environment for a team to share project work, model endpoints, compute, (data) connections, security settings, govern usage.

Subscription \* ⓘ

Resource group \* ⓘ RG1  Create new

Region \* ⓘ East US 2

**Resource details**

Name \* ⓘ azureaihub  ✓

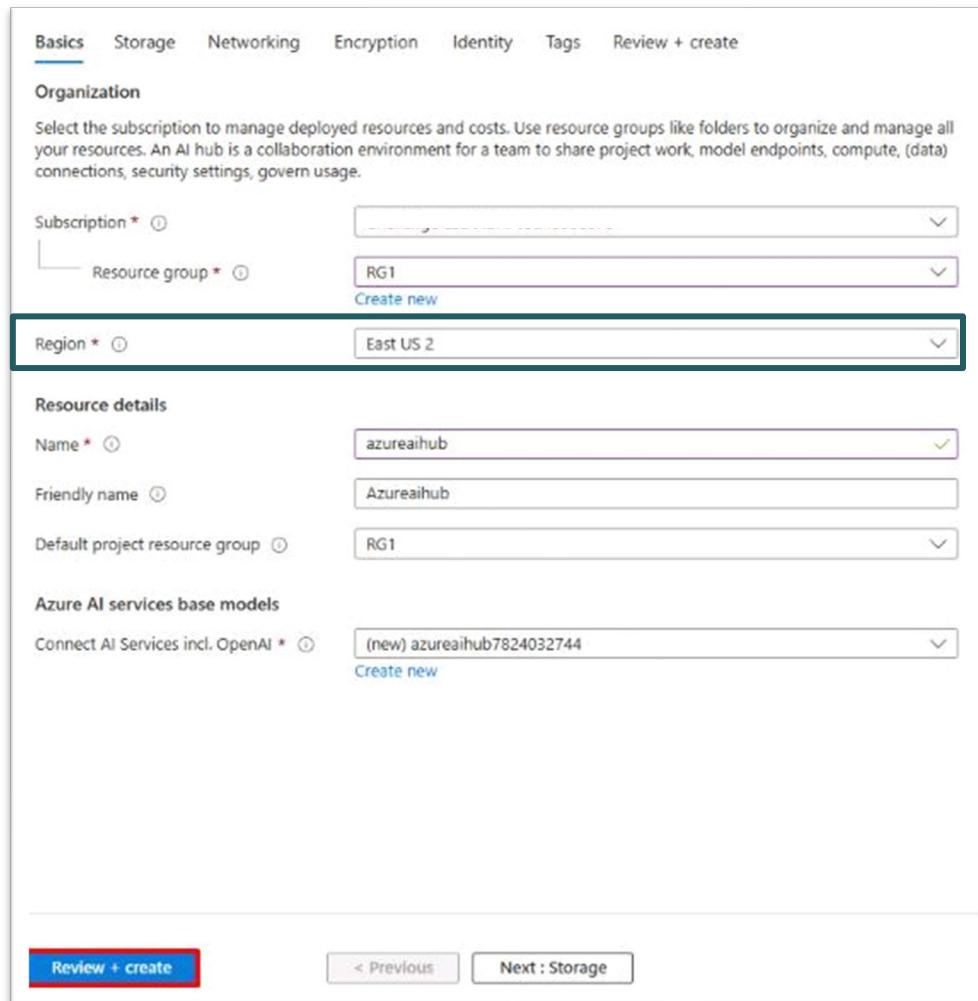
Friendly name ⓘ Azureaihub

Default project resource group ⓘ RG1

**Azure AI services base models**

Connect AI Services incl. OpenAI \* ⓘ (new) azureaihub7824032744  Create new

**Review + create** < Previous Next : Storage



## Important

The availability of models and other resources will depend on the region chosen. Ensure you are using a region that supports Azure AI.

# Needed roles for the hub and project

The hub is the overarching development environment.

- **Role needed:** Owner or Contributor
- **Where:** The associated resource group or an existing hub

Projects exist within a hub and can have different permissions and resources.

**Role needed:** Either the Azure AI Developer or Azure AI Inference Deployment Operator

# 1-1. Set up a hub and project in Azure AI Foundry

## Step 02: Create project in Azure AI Foundry

1. From the **azureaihub** page, select **Launch Azure AI Foundry**.

The screenshot shows the Microsoft Azure (Preview) portal interface. At the top, there's a navigation bar with 'Microsoft Azure (Preview)', a search bar ('Search resources, services, and docs (G+)'), and a 'Copilot' icon. Below the navigation bar, the URL 'Home > Microsoft.MachineLearningServices | Overview >' is visible. On the left, there's a sidebar with various options like 'Overview', 'Activity log', 'Access control (IAM)', etc. The main content area is titled 'azureaihub' and shows an 'Essentials' section with details such as Resource group (0526lab-test), Location (East US), Subscription (Container Registry (edit)), and Key Vault (azureaihub7723947590). Below this, there's a section titled 'Govern the environment for your team in AI Foundry' with a sub-section about Azure AI hub security and a 'Launch Azure AI Foundry' button, which is highlighted with a red box.

Microsoft Azure (Preview)

Search resources, services, and docs (G+)

Copilot

Home > Microsoft.MachineLearningServices | Overview >

azureaihub

Suggest a workload for this ML workspace

Search

Create project

Download config.json

Delete

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Events

Settings

Monitoring

Automation

Support + troubleshooting

Resource group: Project resource group (default)  
0526lab-test

Location: Storage  
East US

Subscription: Container Registry (edit)  
...

Subscription ID: Application Insights (edit)  
...

Key Vault: Provisioning State  
azureaihub7723947590  
Succeeded

Govern the environment for your team in AI Foundry

Your Azure AI hub provides enterprise-grade security, and a collaborative environment to build AI solutions. Centrally audit usage and cost, and set up connections to your company resources that all projects can use. [learn more about the Azure AI Foundry](#)

Launch Azure AI Foundry

# 1-1. Set up a hub and project in Azure AI Foundry

## Step 02: Create project in Azure AI Foundry

- Azure AI Foundry will open in a new tab. Select **+ New project** to create a new project. Enter **project\_AIWorkshop** and select **Create**.

The screenshot shows the Azure AI Foundry Management center interface. On the left, there's a sidebar with options like Management center, All resources, Quota, Hub (azureaihub), Overview, Users, Models + endpoints, Connected resources, and Compute. The Overview tab is selected. In the main area, a hub named "Azureaihub" is displayed. A modal window titled "Name your project" is open, prompting the user to enter a project name. The "Project name" field contains "project\_AIWorkshop". The "Create" button at the bottom of the modal is highlighted with a red box. The overall interface is dark-themed.

# Management layers in Azure AI Foundry

AI Foundry targets three different management needs:

Provide **AI developers and business stakeholders** with a SaaS-like self-serve experience, to allow for rapid AI experimentation

---

Provide **team leads** with central configuration and governance for managing capacity, spend, shareable assets for their team

---

Provide a compliant, yet non-repetitive or duplicate setup by **IT security** using templates

AI Foundry: projects

**Customize** in projects

AI Foundry: hub

**Share** connections, compute, base models

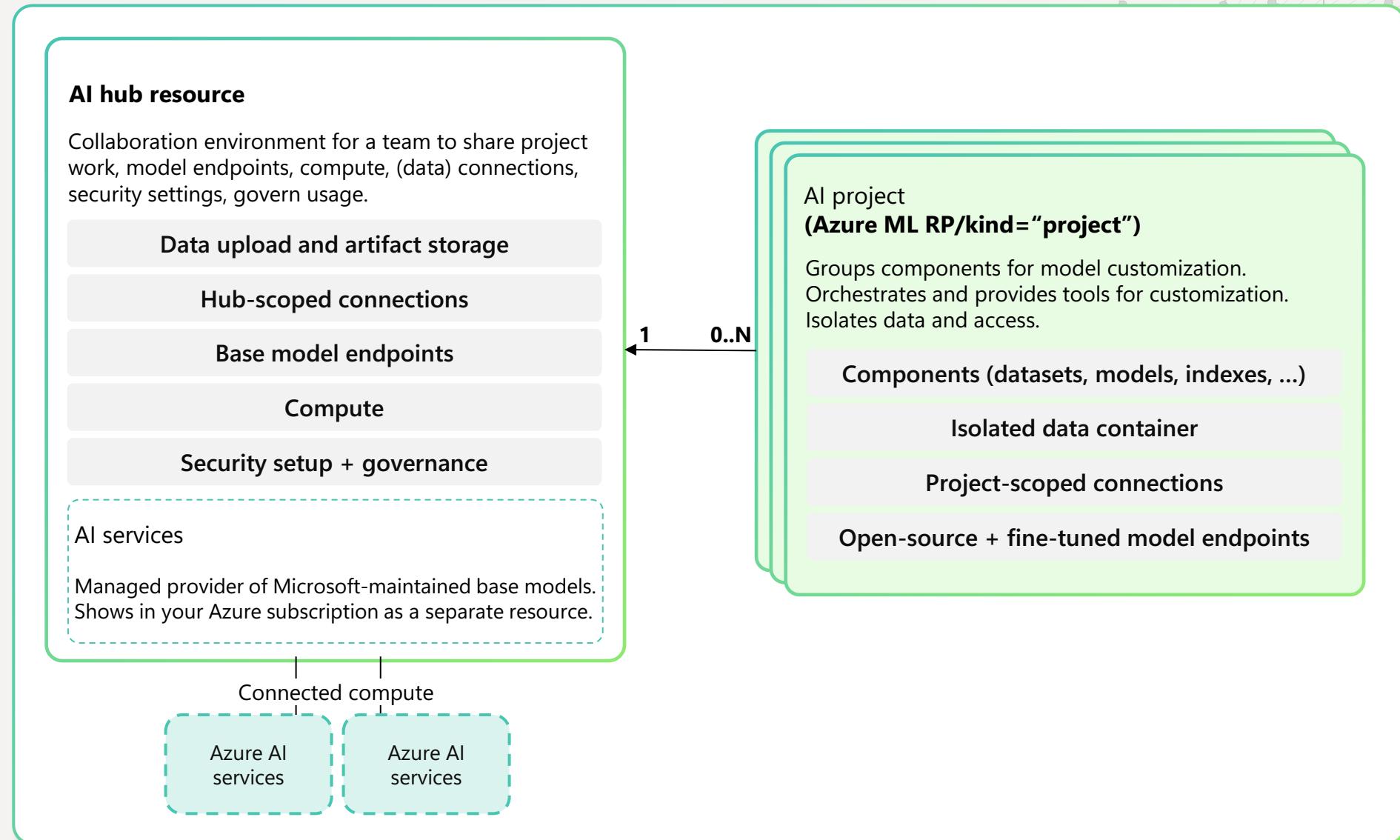
**Govern** quota and usage

Azure portal

Platform setup

**Govern** security

# Azure AI Foundry Hubs

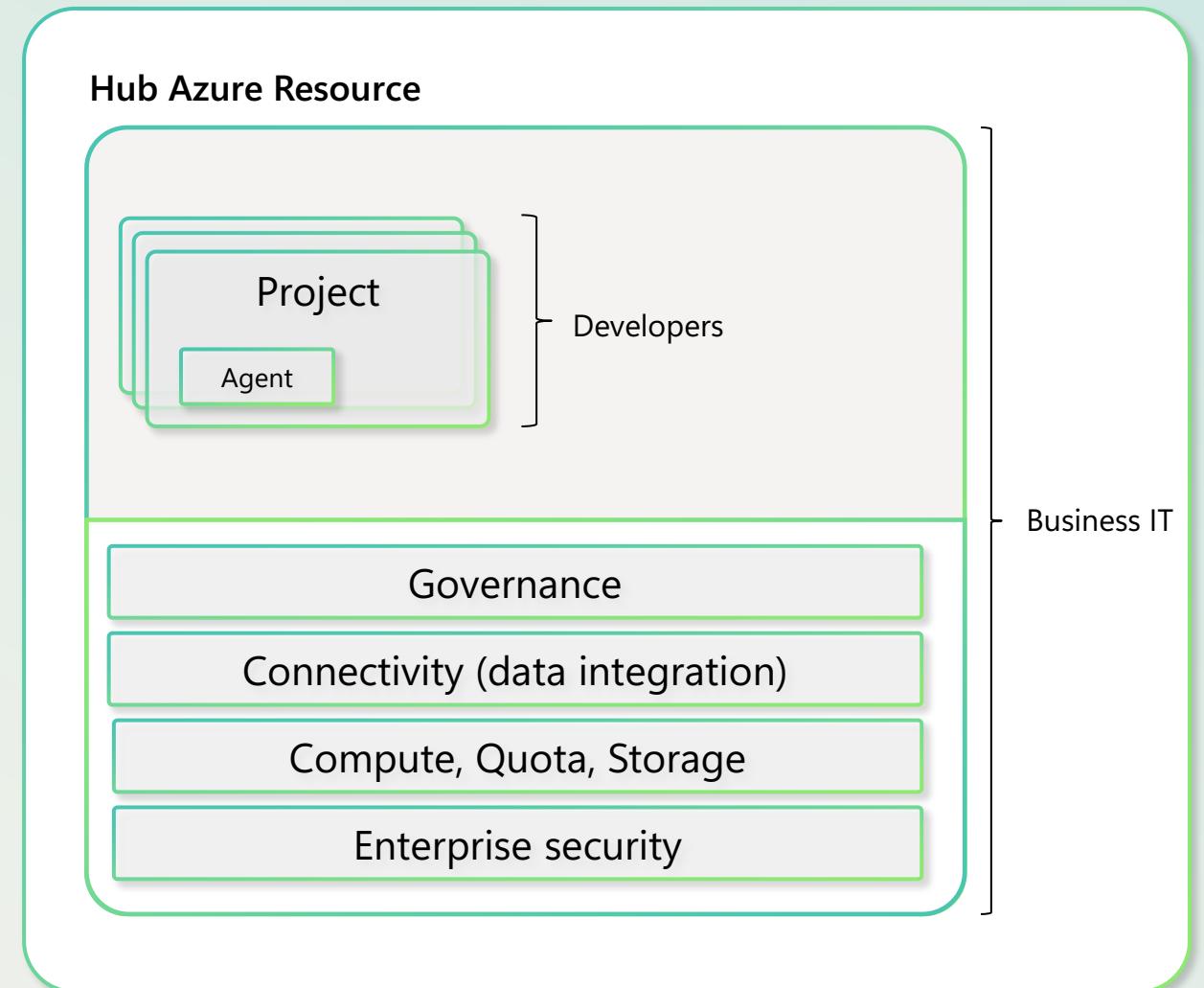


# Enterprise setup in AI Foundry

Seeks to balance a self-serve authoring experience for developers in AI Foundry, with granular platform controls for IT using Azure Portal/IaC templates

- **Hub:** configures Azure infrastructure, enterprise security, storage, connectivity for data integration. Groups projects for a team
- **Projects:** Provides a place for developers to collaborate and organize their work for customizing AI models. Container for data upload and access control

Tenant, Management Groups, Subscriptions



# Location of search service

Create a search service ...

Basics Scale Networking Tags Review + create

Project details

Subscription \*

Resource Group \*  RG1 [Create new](#)

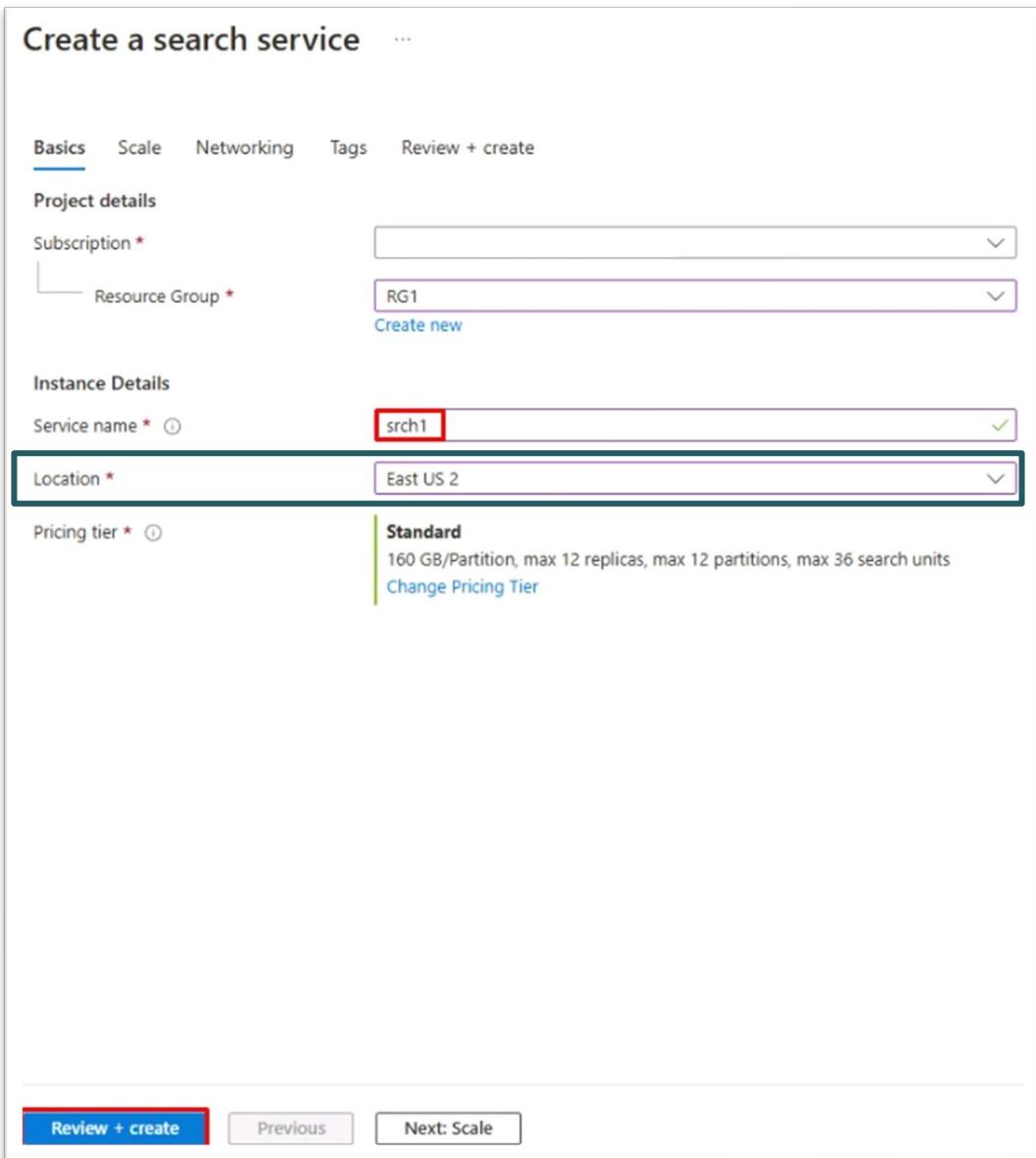
Instance Details

Service name \*  srch1

Location \*  East US 2

Pricing tier \*  Standard  
160 GB/Partition, max 12 replicas, max 12 partitions, max 36 search units  
[Change Pricing Tier](#)

[Review + create](#) [Previous](#) [Next: Scale](#)



## Important

Ensure the **Location** is set to the same location where the hub was created earlier.

# Task 02: Create AI model deployments

## Introduction:

To enable AI-driven customer service and content moderation, MicroRetail requires robust AI models that can process language effectively. By deploying models like gpt-4o-mini for conversation handling and text-embedding-ada-002 for data representation, MicroRetail can power its AI chatbot and optimize search results based on user queries.

## Description:

In this task, you'll deploy the **gpt-4o-mini** and **text-embedding-ada-002** models to your project. These models are necessary for creating an AI-assisted chat application.

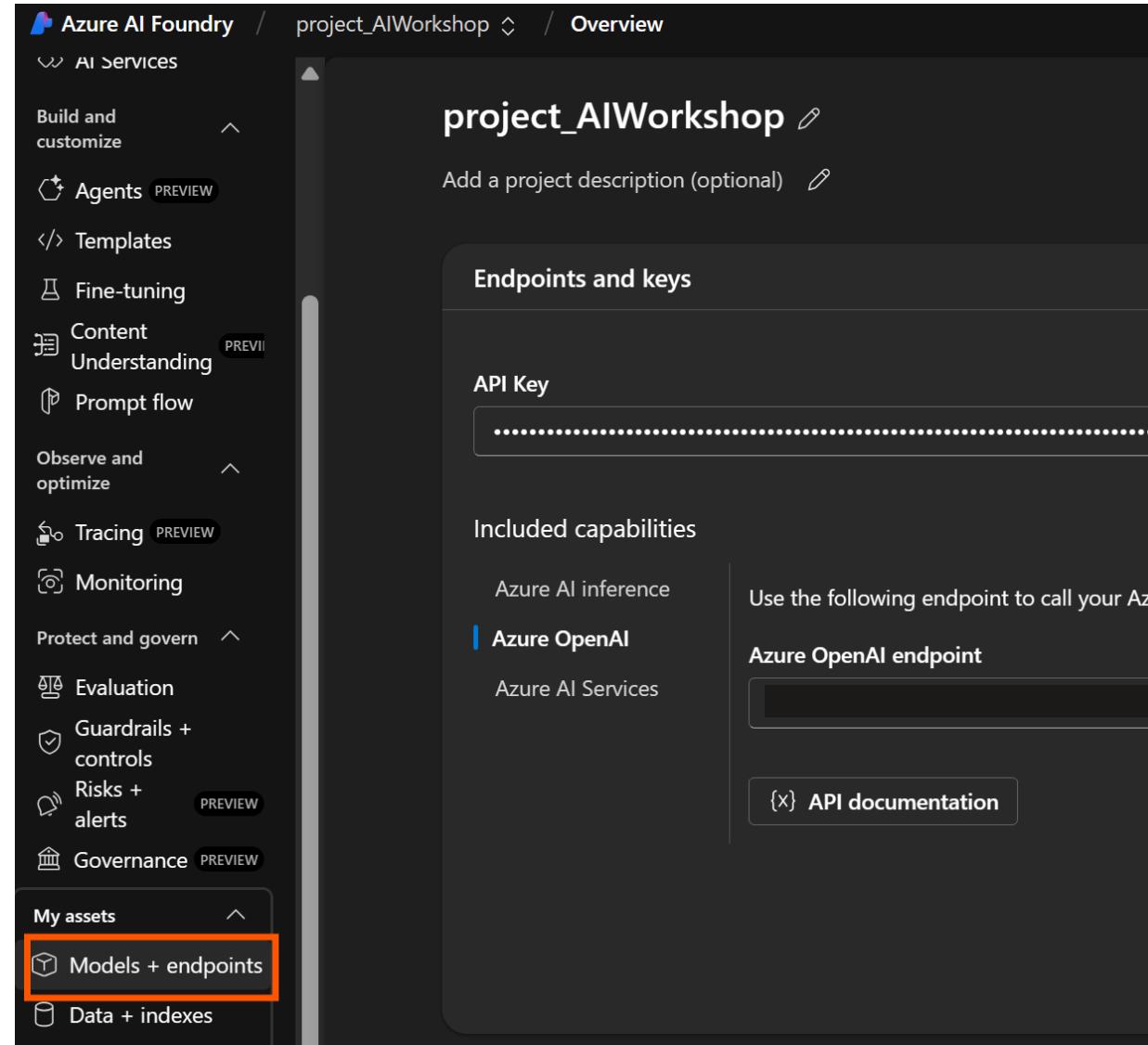
## Success Criteria:

- The **gpt-4o-mini** model has been deployed.
- The **text-embedding-ada-002** model has been deployed.

## 1-2. Create AI model deployments

### Step 01: Deploy gpt-4o-mini and text-embedding-ada-002 models

1. From the **Overview** tab on the **project\_AIWorkshop** page, select **Models + endpoints** from the left menu.



The screenshot shows the Azure AI Foundry interface. The top navigation bar includes the Azure AI Foundry logo, the project name "project\_AIWorkshop", and the "Overview" tab. The left sidebar contains several sections: "Build and customize" (Agents, Templates, Fine-tuning, Content Understanding, Prompt flow), "Observe and optimize" (Tracing), "Protect and govern" (Evaluation, Guardrails + controls, Risks + alerts, Governance), and "My assets" (Models + endpoints, Data + indexes). The "Models + endpoints" option is highlighted with an orange border. The main content area displays the "project\_AIWorkshop" title, a placeholder for "Add a project description (optional)", and a section titled "Endpoints and keys" which shows an "API Key" field containing a redacted string. Below this is a "Included capabilities" section listing "Azure AI inference", "Azure OpenAI" (which is selected and highlighted in blue), and "Azure AI Services". A note says "Use the following endpoint to call your Azure OpenAI service" followed by an "Azure OpenAI endpoint" input field and a link to "API documentation".

## 1-2. Create AI model deployments

### Step 01: Deploy gpt-4o-mini and text-embedding-ada-002 models

2. From the **Manage deployments of your models and services** page, select **+ Deploy model** and then select **Deploy base model**.
3. From the **Select a model** page, select **gpt-4o-mini**, then select **Confirm**.

The screenshot shows a 'Select a model' dialog box. On the left, a search bar contains 'gpt-4o-mini'. Below it is a list of models:

- gpt-4o-mini-tts (Text to speech)
- gpt-4o-mini-transcribe (Speech to text)
- gpt-4o-mini-audio-preview (Audio generation)
- gpt-4o-mini-realtime-prev... (Audio generation)
- gpt-4o-mini (Chat completion)** (highlighted with an orange border)
- ruslandev-llama-3-8b-gpt... (Text generation)
- abdou-arabert-mini-algeri...

On the right, details for the selected 'gpt-4o-mini' model are shown:

**gpt-4o-mini**  
Task: Chat completion

GPT-4o mini enables a broad range of tasks with its low cost and latency, such as applications that chain or parallelize multiple model calls (e.g., calling multiple APIs), pass a large volume of context to the model (e.g., full code base or conversation history), or interact with customers through fast, real-time text responses (e.g., customer support chatbots).

Today, GPT-4o mini supports text and vision in the API, with support for text, image, video and audio inputs and outputs coming in the future. The model has a context window of 128K tokens and knowledge up to October 2023. Thanks to the improved tokenizer shared with GPT-4o, handling non-English text is now even more cost effective.

GPT-4o mini surpasses GPT-3.5 Turbo and other small models on academic benchmarks across both textual intelligence and multimodal reasoning, and supports the same range of languages as GPT-4o. It also demonstrates strong performance in function calling, which can enable developers to build applications that fetch data or take actions with external systems, and improved long-context performance compared to GPT-3.5 Turbo.

At the bottom are 'Confirm' and 'Cancel' buttons, with 'Confirm' highlighted by an orange border.

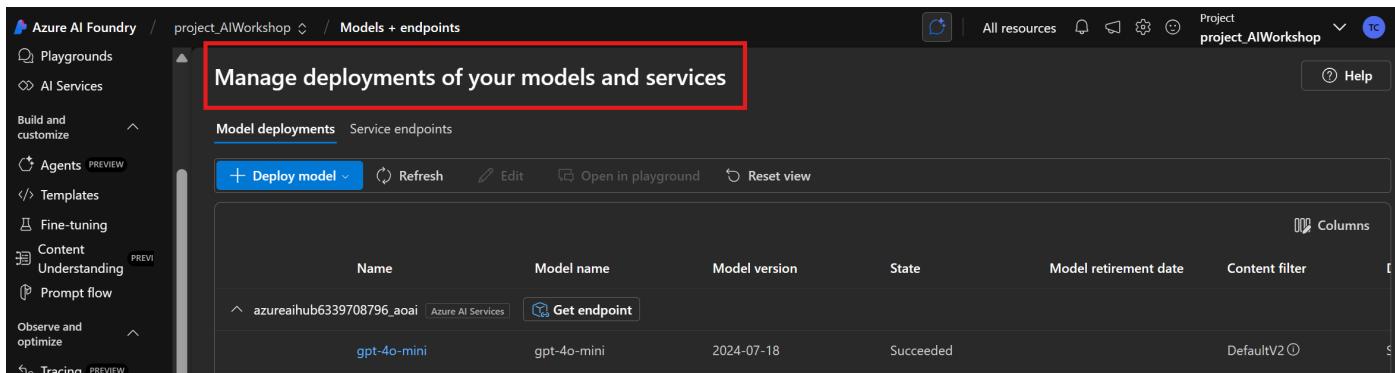
## 1-2. Create AI model deployments

### Task 01: Deploy gpt-4o-mini and text-embedding-ada-002 models

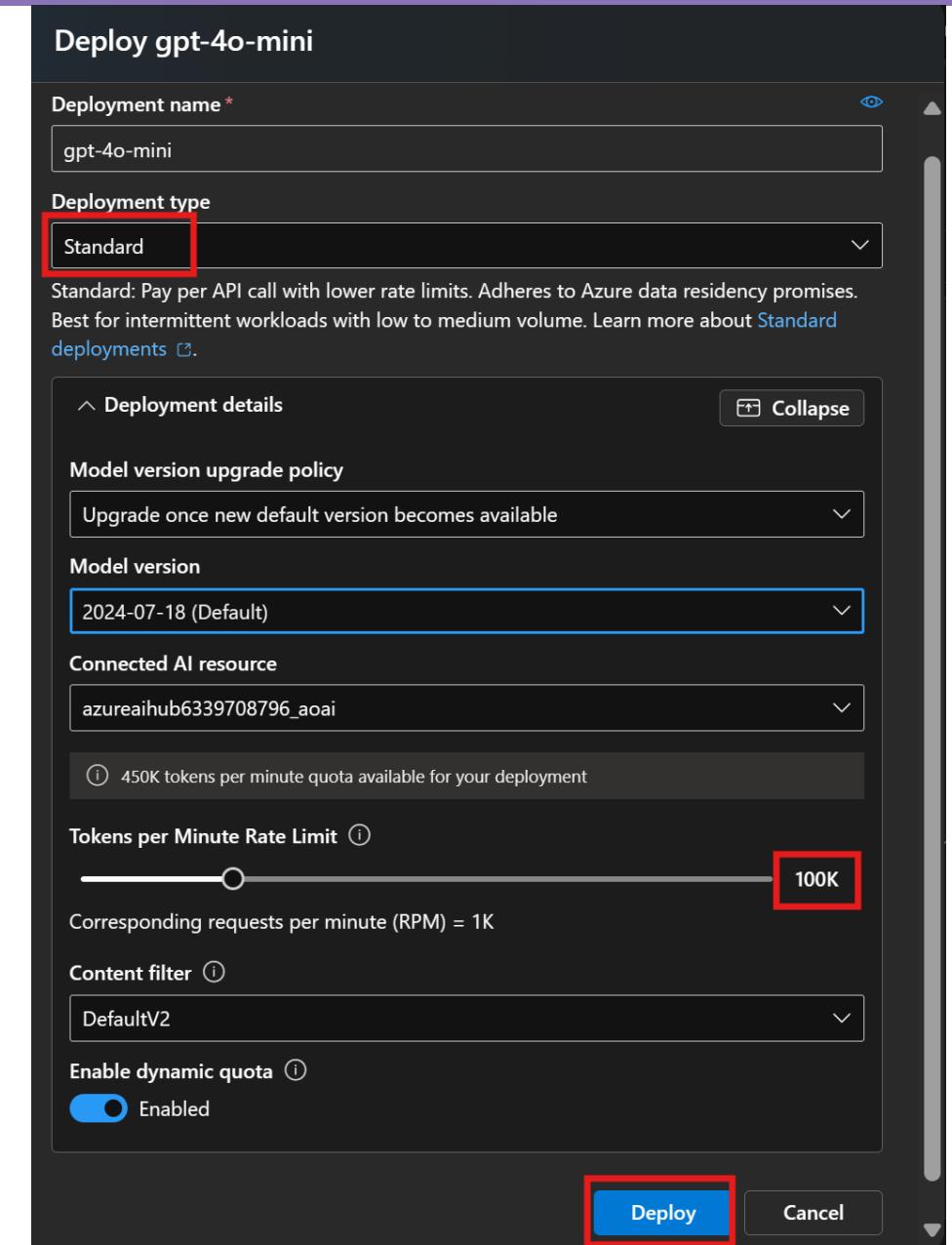
4. On the **Deploy model gpt-4o-mini** page, set the **Deployment type** to **Standard**.

5. Move the **Tokens per Minute Rate Limit** slider to around **100K**. Leave all other settings as default and select **Deploy**.

6. Once deployed, select the back button to return to the **Manage deployments of your models and services** page and deploy another base model.



The screenshot shows the Azure AI Foundry interface. The top navigation bar includes 'Azure AI Foundry / project\_AIWorkshop / Models + endpoints'. Below this, a main header reads 'Manage deployments of your models and services' with a red box highlighting it. Underneath, there are two tabs: 'Model deployments' (selected) and 'Service endpoints'. A 'Deploy model' button is visible. The main content area displays a table of model deployments, showing one entry: 'gpt-4o-mini' (Model name), 'gpt-4o-mini' (Model version), '2024-07-18' (Model retirement date), 'Succeeded' (State), and 'DefaultV2' (Content filter). A 'Get endpoint' button is also present.



The screenshot shows the 'Deploy gpt-4o-mini' configuration page. The 'Deployment name' field contains 'gpt-4o-mini'. The 'Deployment type' dropdown is set to 'Standard', which is highlighted with a red box. Below it, a description states: 'Standard: Pay per API call with lower rate limits. Adheres to Azure data residency promises. Best for intermittent workloads with low to medium volume.' The 'Model version' dropdown is set to '2024-07-18 (Default)'. The 'Connected AI resource' dropdown is set to 'azureaihub6339708796\_aoai'. A note indicates '450K tokens per minute quota available for your deployment'. The 'Tokens per Minute Rate Limit' slider is set to '100K', which is also highlighted with a red box. Below the slider, it says 'Corresponding requests per minute (RPM) = 1K'. The 'Content filter' dropdown is set to 'DefaultV2'. The 'Enable dynamic quota' toggle switch is turned 'Enabled'. At the bottom, a large blue 'Deploy' button is highlighted with a red box, and a 'Cancel' button is to its right.

## 1-2. Create AI model deployments

### Step 01: Deploy gpt-4o-mini and text-embedding-ada-002 models

7. For this one, select **text-embedding-ada-002**, then select **Confirm**.

The screenshot shows the 'Select a model' interface. A search bar at the top contains the text 'text-embedding-ada-002'. Below it, a list of models is displayed. The 'text-embedding-ada-002' model is highlighted with a red box around its thumbnail and name. The model details page is open, showing the following information:

- Task:** Embeddings
- Description: text-embedding-ada-002 outperforms all the earlier embedding models on text search, code search, and sentence similarity tasks and gets comparable performance on text classification. Embeddings are numerical representations of concepts converted to number sequences, which make it easy for computers to understand the relationships between those concepts.
- Note: this model can be deployed for inference, specifically for embeddings, but cannot be finetuned.
- Model variation:** text-embedding-ada-002 is part of gpt-3 model family.
- Learn more at <https://learn.microsoft.com/azure/cognitive-services/openai/concepts/models#embeddings-models>

At the bottom right of the model details page, there are 'Confirm' and 'Cancel' buttons, with 'Confirm' also highlighted by a red box.

The screenshot shows the deployment configuration page for the 'text-embedding-ada-002' model. The 'Deployment name' field is set to 'text-embedding-ada-002'. The 'Deployment type' dropdown is set to 'Standard', which is also highlighted by a red box. The 'Deployment details' section includes:

- Model version upgrade policy:** Opt out of automatic model version upgrades. A note indicates that automatic version update is not available for embedding and finetuned models.
- Model version:** 2 (Default)
- Connected AI resource:** azureaihub6339708796\_aoai
- Tokens per Minute Rate Limit:** A slider is set to 120K, which is also highlighted by a red box. Below the slider, it says 'Corresponding requests per minute (RPM) = 720'.
- Content filter:** DefaultV2
- Enable dynamic quota:** Enabled

At the bottom right, there are 'Deploy' and 'Cancel' buttons.

8. Ensure the **Deployment type** is set to **Standard**, then select **Deploy**.

# Task 03: Create a search service and add data connections

## **Introduction:**

MicroRetail's chatbot must provide accurate and context-aware responses. To achieve this, an AI-powered search service is required. By implementing Azure AI Search, MicroRetail can index and retrieve relevant information from a structured data repository, ensuring customers receive the most relevant responses based on their queries.

## **Description:**

In this task, you'll create an Azure AI Search service resource and connect it to your Azure AI project. The AI Search service enhances chatbot functionality by allowing it to pull relevant information from customer reviews, product catalogs, and historical interactions.

## **Success Criteria:**

- The Azure AI Search service has been created.
- The Azure AI Search service has been connected to the Azure AI project.

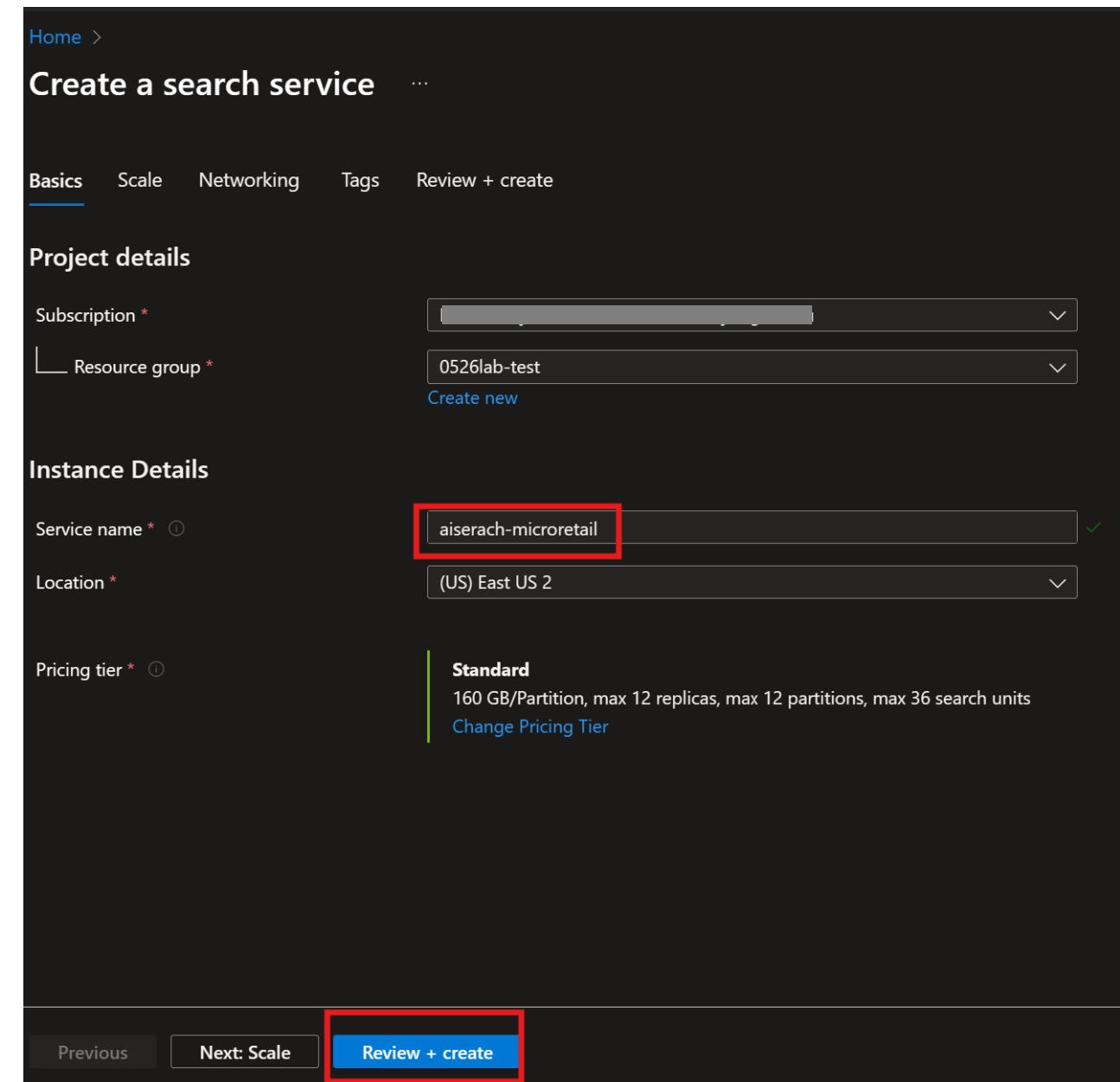
# 1-3. Create a search service and add data connections

## Step 01: Create the azureaiss search service

1. To create a search service, return to the Azure portal tab.
2. In the search bar at the top, search for **search**, then select **AI Search**.
3. From the **AI Search** page, select **Create search service**.
4. On the **Create a search service** page, select your resource group and location (these may differ from the screenshot).

Name the search service **aiserach-microretail**, select **Review + create**, then select **Create**.

\*\* Ensure the **Location** is set to the same location where the hub was created earlier.



# 1-3. Create a search service and add data connections

(Optional)

## Step 01: Create the azureaiss search service

5. Return to the tab with Azure AI Overview . On the right of the Overview page, select **Open in management center**.

Azure AI Foundry / project\_aiworkshop / Overview

project\_AIWorkshop

Add a project description (optional)

**Endpoints and keys** [View all endpoints](#)

AI Services resource

API Key

Included capabilities

- Azure AI inference
- Azure OpenAI**
- Azure AI Services

Use the following endpoint to call your Azure OpenAI models:

Azure OpenAI endpoint

[API documentation](#)

**Project details**

Project connection string

Subscription

Subscription ID

Location

Manage project settings

- Add users
- View quota
- Connect resources
- Track costs

**Open in management center**

# 1-3. Create a search service and add data connections

(Optional)

## Step 01: Create the azureaiss search service

6. From the **Management center** for **project\_AIWorkshop**, under **Connected resources**, select **+ New connection**.

The screenshot shows the Azure Management Center interface for the project\_AIWorkshop project. The left sidebar lists various project components like Overview, Users, Models + endpoints, and Connected resources. The main area displays the 'Connected resources' section, which currently contains four entries. A red box highlights the '+ New connection' button at the bottom of this section.

| Name                | Type         |
|---------------------|--------------|
| azur [REDACTED] i   | Azure OpenAI |
| azur [REDACTED]     | AI Services  |
| ai-yt [REDACTED] ai | Azure OpenAI |
| ai-yt [REDACTED]    | AI Services  |

**+ New connection**

## Step 01: Create the azureaiss search service

7. On the **Add a connection to external assets** page, select **Azure AI Search**, then select **Add connection** next to the **aiserach-microretail** service.

Connect an Azure AI Search resource

← Back to select an asset type

Browse resources  Enter manually

Search for a resource

Displaying (3) resources

|   |                                |                |
|---|--------------------------------|----------------|
| Name<br>aiserach-microretail                            | Resource group<br>0526lab-test | Add connection |
| Location<br>eastus2                                     | Sku<br>standard                |                |
| Subscription<br>MCAPS-Hybrid-REQ-109107-2025-yunghuichu | Semantic search<br>free        |                |

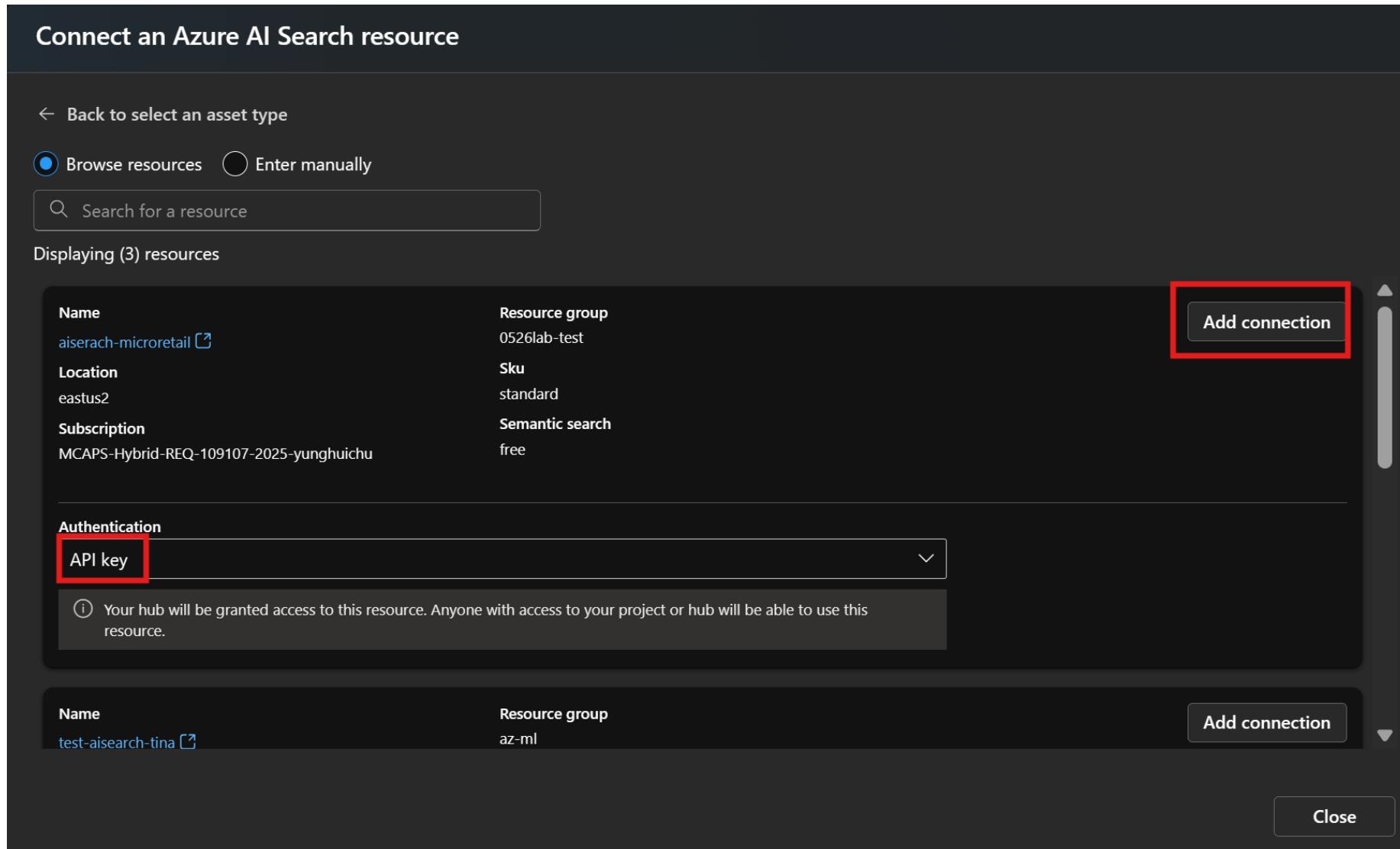
**Authentication**

API key

Your hub will be granted access to this resource. Anyone with access to your project or hub will be able to use this resource.

|                            |                         |                |
|----------------------------|-------------------------|----------------|
| Name<br>test-aisearch-tina | Resource group<br>az-ml | Add connection |
|----------------------------|-------------------------|----------------|

Close



# Exercise two

Perform prompt engineering

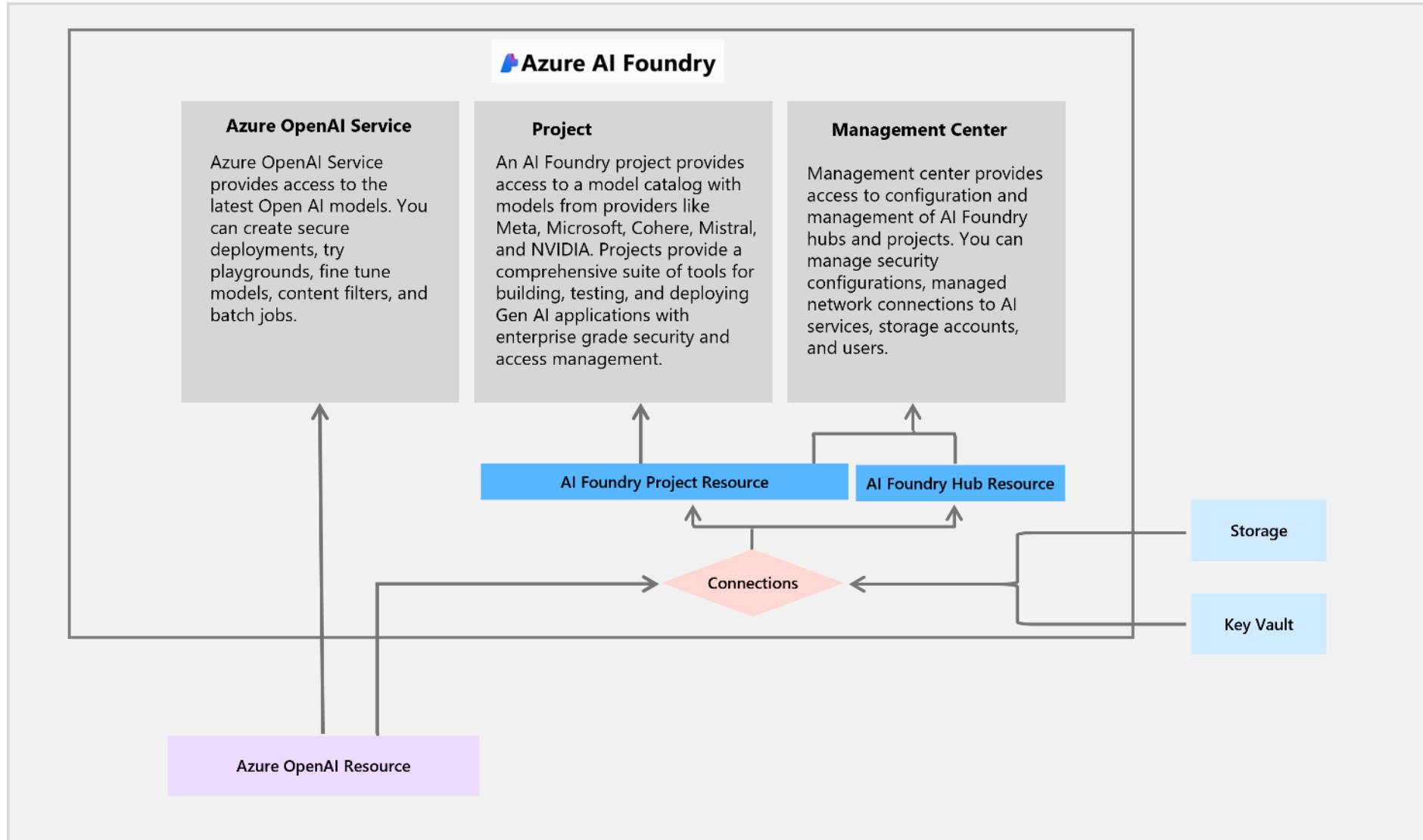
# Introduction: Perform prompt engineering

In addition to providing a safe and engaging environment MicroRetail aims to provide a personalized online shopping experience for its customers. To achieve this, MicroRetail will introduce its own data and vector stores in the chat playground.

**After completing this exercise, you'll be able to, within Azure AI Foundry:**

-  Manage compute resources for AI workflows.
-  Add data to AI projects and create a search index.
-  Use the Azure AI Playground to interact with indexed data.

# Exercise two architecture



# Task 01: Manage compute at the hub level

## Introduction:

To ensure smooth execution of AI workflows, **MicroRetail** must provision appropriate compute resources. Managing compute at the hub level allows for efficient processing of AI models and prompt flows, supporting the company's AI-driven chatbot and customer interaction initiatives.

## Description:

In this task, you'll create a compute instance to use in running prompt flows. This ensures the necessary processing power is available to execute AI workloads effectively.

## Success Criteria:

- The compute instance has been created and is running.

## 2-1. Manage compute at the hub level

### Step 01: Create the compute instance resource in hub

- From the **Azure AI Foundry Management center**, select **Compute** under the **Hub** section from the left menu.

The screenshot shows the Azure AI Foundry Management center interface. On the left, there's a navigation sidebar with sections like 'Management center', 'Hub (azureaihub)', 'Project (project\_AIWorkshop)', and 'Compute'. The 'Compute' section is highlighted with a red box. The main area displays a project named 'project\_AIWorkshop'. Under 'Models + endpoints', there are two entries: 'gpt-4o-mini' and 'text-embedding-ada-002'. Under 'Connected resources', there are two entries: 'azureaihub6339708796\_aoui' and 'azureaihub6339708796'. At the bottom left, there's a 'Go to project' button.

#### IMPORTANT

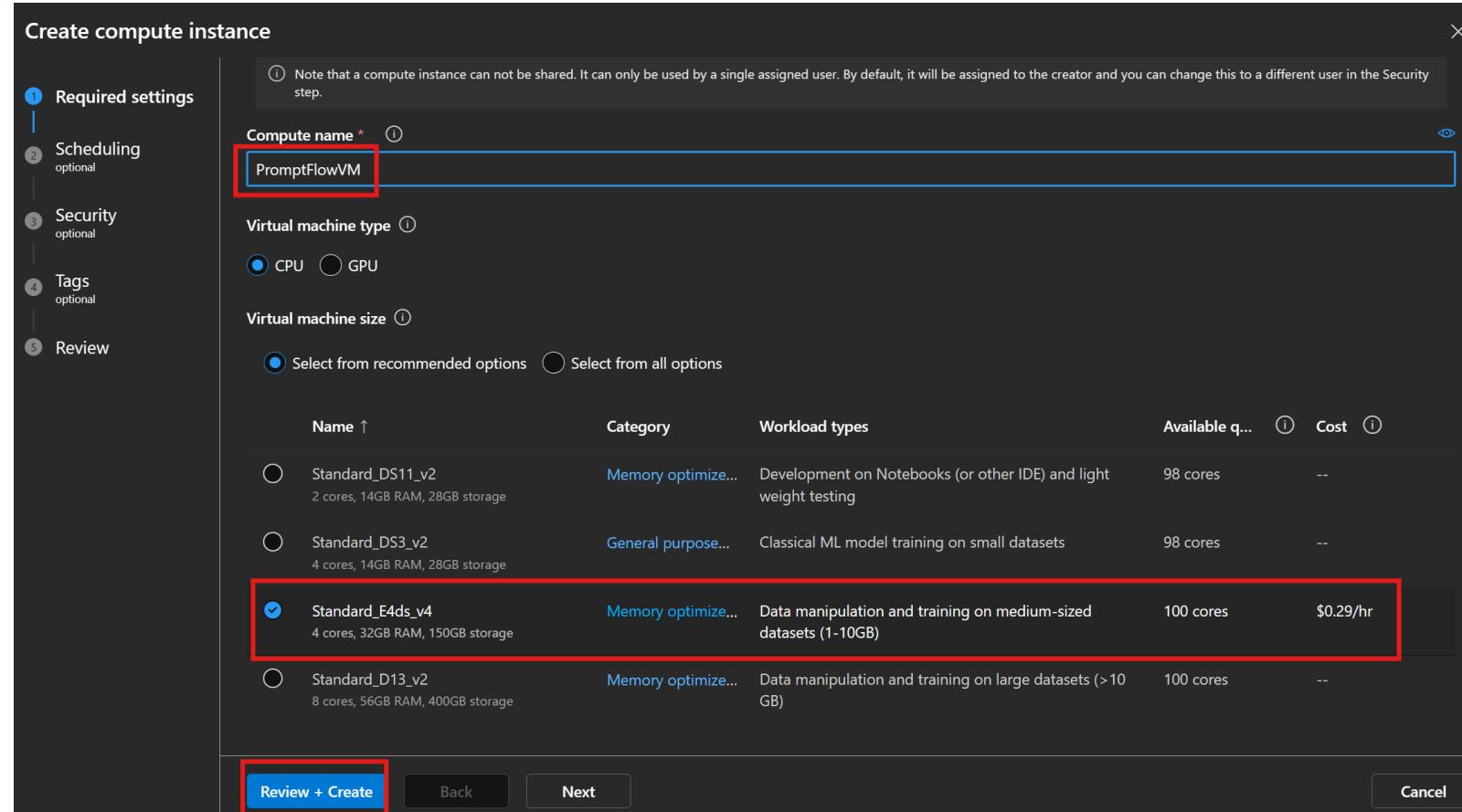
A compute instance is a virtual machine that is used to run prompt flows within projects. It's created at the hub level and accessible by any projects with proper permissions.

- From the **Manage compute resources in this hub** page, select **+ New**.

## 2-1. Manage compute at the hub level

### Step 01: Create the compute instance resource in hub

3. Name the compute instance **PromptFlowVM**, leave the other settings as default, then select **Review + Create**.



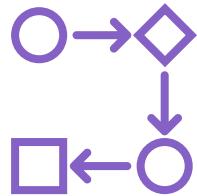
4. Once the compute instance is created, it should show a status of **Running**.

#### IMPORTANT

The compute instance can be used for various different tasks such as creating an index, building an open source model, or executing complex prompt flows.

# Compute instance

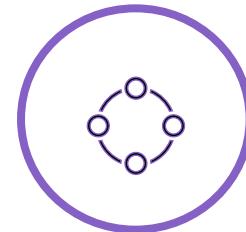
## What it is:



A virtual machine used to run prompt flows within projects



Accessible by any projects with proper permissions



Created at the hub level

You will use the compute instance later to run prompt flows.

# Task 02: Add data and create a search index

## Introduction:

To enable personalized AI interactions, **MicroRetail** needs to integrate structured data that enhances chatbot responses. By indexing this data, the AI model can retrieve and process relevant customer and product information efficiently.

## Description:

In this task, you'll add data to your project and index it for retrieval. This allows the chatbot to access relevant product details and improve customer interaction quality.

## Success Criteria:

- **products.xlsx** has been created.
- **products.xlsx** has been indexed successfully.

## 2-1. Add data and create a search index

### Step 01: Download products.xlsx file for use in Playground

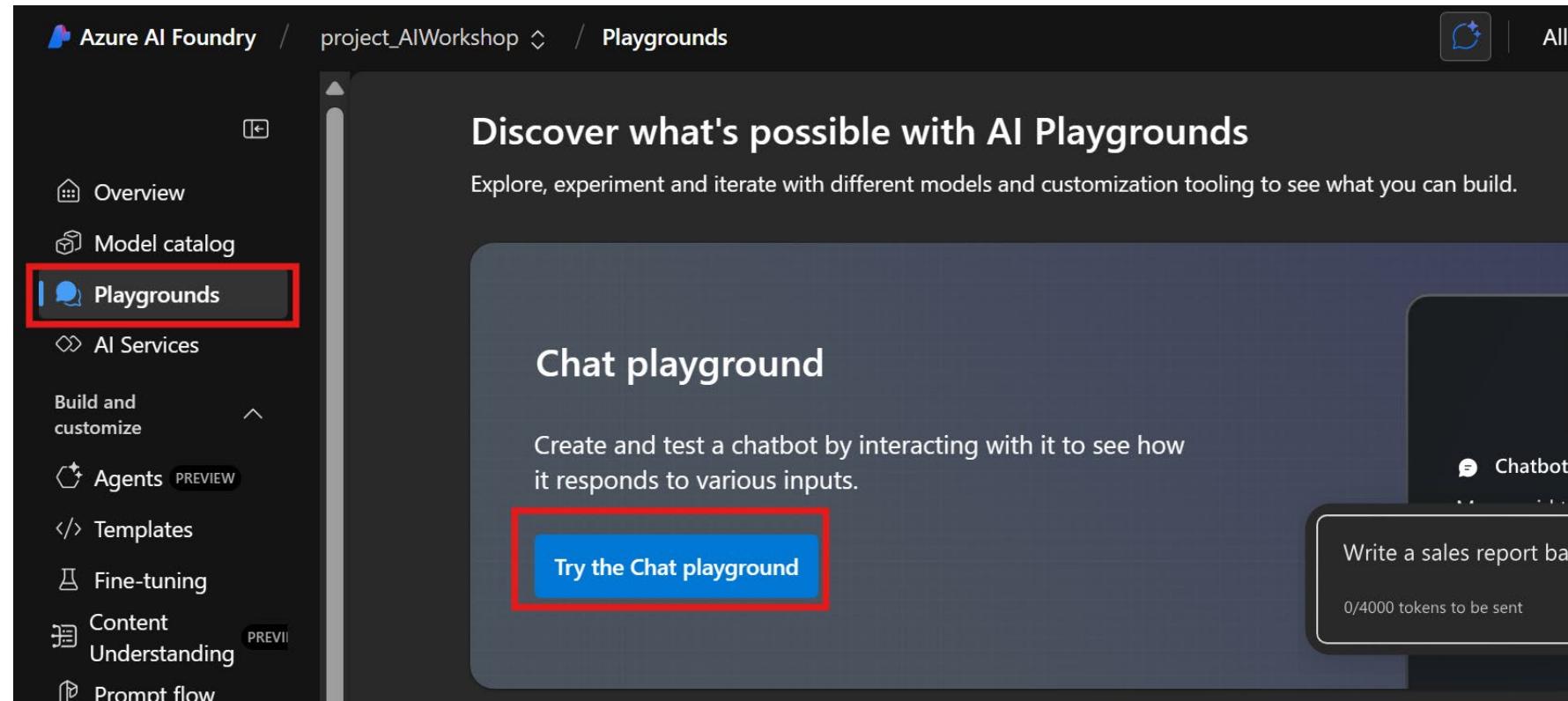
1. Download the **products.xlsx** file [HERE](#).

| id | name                       | price | category        | brand         | description  |
|----|----------------------------|-------|-----------------|---------------|--|
| 1  | TrailMaster X4 Tent        | 250   | Tents           | OutdoorLiving | Unveiling the TrailMaster X4 Tent from OutdoorLiving, your |
| 2  | Adventurer Pro Backpack    | 90    | Backpacks       | HikeMate      | Venture into the wilderness with the HikeMate's Adventurer |
| 3  | Summit Breeze Jacket       | 120   | Hiking Clothing | MountainStyle | Discover the joy of hiking with MountainStyle's Summit Bre |
| 4  | TrekReady Hiking Boots     | 140   | Hiking Footwear | TrekReady     | Introducing the TrekReady Hiking Boots - stepping up your  |
| 5  | BaseCamp Folding Table     | 60    | Camping Tables  | CampBuddy     | CampBuddy's BaseCamp Folding Table is an adventurer's      |
| 6  | EcoFire Camping Stove      | 80    | Camping Stoves  | EcoFire       | Introducing EcoFire's Camping Stove, your ultimate compa   |
| 7  | CozyNights Sleeping Bag    | 100   | Sleeping Bags   | CozyNights    | Embrace the great outdoors in any season with the lightwe  |
| 8  | Alpine Explorer Tent       | 350   | Tents           | AlpineGear    | Welcome to the joy of camping with the Alpine Explorer Te  |
| 9  | SummitClimber Backpack     | 120   | Backpacks       | HikeMate      | Adventure waits for no one! Introducing the HikeMate Sum   |
| 10 | TrailBlaze Hiking Pants    | 75    | Hiking Clothing | MountainStyle | Meet the TrailBlaze Hiking Pants from MountainStyle, the s |
| 11 | TrailWalker Hiking Shoes   | 110   | Hiking Footwear | TrekReady     | Meet the TrekReady TrailWalker Hiking Shoes, the ideal co  |
| 12 | TrekMaster Camping Chair   | 50    | Camping Tables  | CampBuddy     | Gravitate towards comfort with the TrekMaster Camping C    |
| 13 | PowerBurner Camping Stove  | 100   | Camping Stoves  | PowerBurner   | Unleash your inner explorer with the PowerBurner Dual Bu   |
| 14 | MountainDream Sleeping Bag | 130   | Sleeping Bags   | MountainDream | Meet the MountainDream Sleeping Bag: your new must-ha      |
| 15 | SkyView 2-Person Tent      | 200   | Tents           | OutdoorLiving | Introducing the OutdoorLiving SkyView 2-Person Tent, a pe  |
| 16 | TrailLite Daypack          | 60    | Backpacks       | HikeMate      | Step up your hiking game with HikeMate's TrailLite Daypac  |
| 17 | RainGuard Hiking Jacket    | 110   | Hiking Clothing | MountainStyle | Introducing the MountainStyle RainGuard Hiking Jacket - th |
| 18 | TrekStar Hiking Sandals    | 70    | Hiking Footwear | TrekReady     | Meet the TrekStar Hiking Sandals from TrekReady - the ult  |
| 19 | Adventure Dining Table     | 90    | Camping Tables  | CampBuddy     | Discover the joy of outdoor adventures with the CampBudd   |
| 20 | CompactCook Camping Stove  | 60    | Camping Stoves  | CompactCook   | Step into the great outdoors with the CompactCook Campi    |

## 2-1. Add data and create a search index

### Step 02: Create vector index for gpt-4o-mini model

1. Return to your browser tab with Azure AI Foundry. From the **Management center for project\_AIWorkshop**, select **Go to project** at the bottom of the left menu.
2. From the left menu of the **project\_AIWorkshop** page, select **Playgrounds**.
3. At the top of the page, select **Try the Chat playground**.



## 2-1. Add data and create a search index

### Step 02: Create vector index for gpt-4o-mini model

- From the **Chat playground** page, ensure the **Deployment** is set to the **gpt-4o-mini** model, then select **+ Add a new data source** under the **Add your data** dropdown menu.

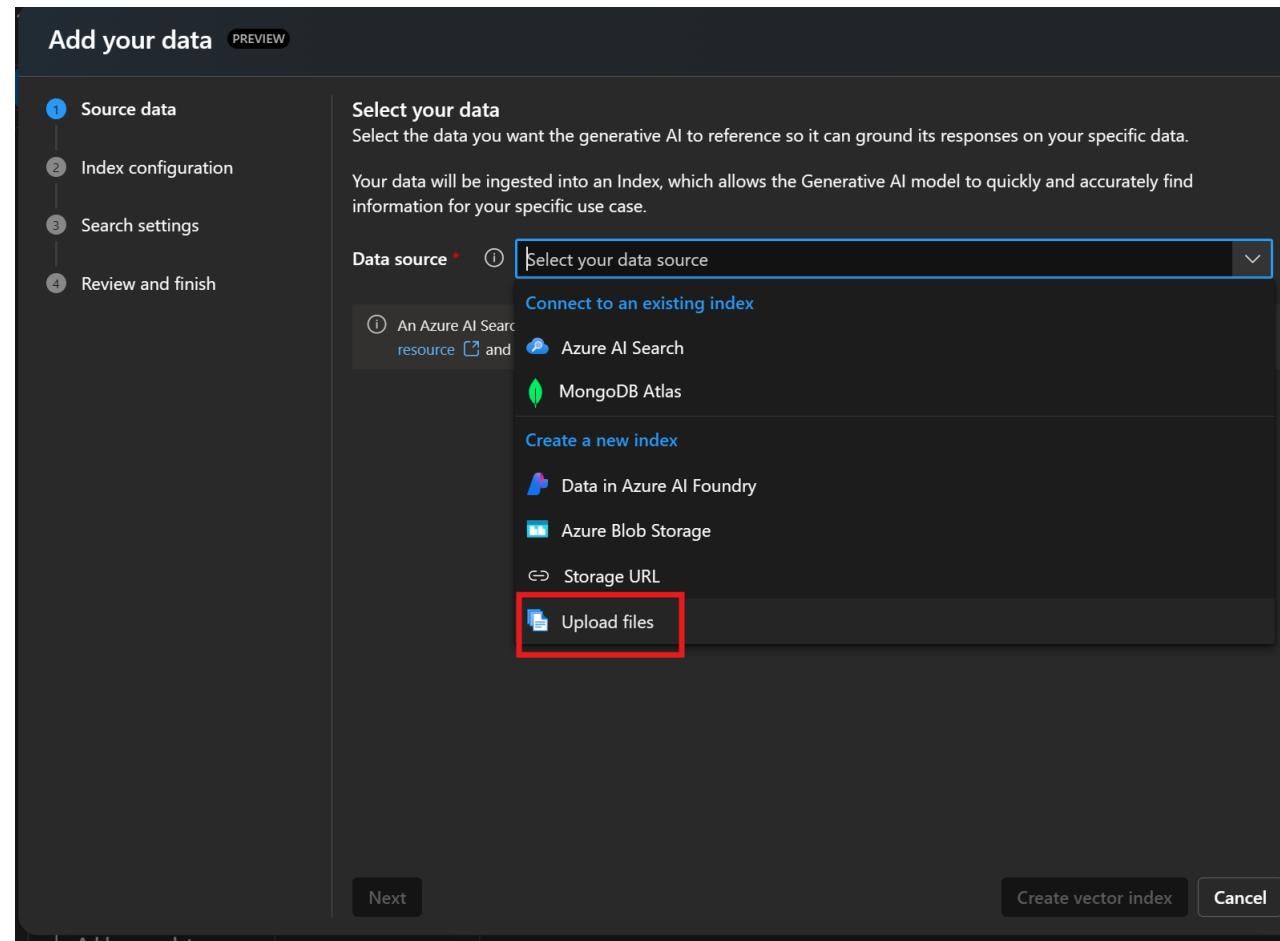
The screenshot shows the Azure AI Foundry interface with the following details:

- Project:** project\_AIWorkshop
- Playground:** Chat playground
- Setup Section:**
  - Deployment:** gpt-4o-mini (version:2024-07-18) (highlighted with a red box)
  - Give the model instructions and context:** You are an AI assistant that helps people find information.
  - Buttons:** Apply changes, Generate system prompt, + Add section
- Add your data Section:**
  - Sub-section:** Ask questions about your own data. The data remains stored in the data source you designate. [Learn more about how your data is protected.](#)
  - Select available project index:** No project index
  - Buttons:** + Add a new data source (highlighted with a red box)
- Chat history:** Placeholder text: Start with a sample prompt. Options include: Historical fiction, Recipe creation, Poetry generation.
- Bottom:** Type user query here. (Shift + Enter for new line)
- Footer:** 11/128000 tokens to be sent, a file icon, and a send button.

## 2-1. Add data and create a search index

### Step 02: Create vector index for gpt-4o-mini model

5. From the **Source data** tab on the **Add your data** page, select the **Data source** dropdown menu, then select **Upload files**.
6. Select the **Upload** dropdown menu, then select **Upload files**.
7. Select the **products.xlsx** file, then select **Open**. Select **Next** once it has been uploaded.



### Method 2: 資料於AI Foundry管理

The screenshot shows the Azure AI Foundry interface with the 'Data + indexes' section selected. The left sidebar includes 'Playgrounds', 'AI Services', 'Build and customize', 'Agents', 'Templates', 'Fine-tuning', 'Content Understanding', 'Prompt flow', 'Observe and optimize', 'Tracing', 'Monitoring', 'Protect and govern', 'Evaluation', 'Guardrails + controls', 'Risks + alerts', and 'Governance'. The 'Data + indexes' section is highlighted with a red box. Within this section, there are tabs for 'Data files' (which is selected) and 'Indexes'. A '+ New data' button is highlighted with a red box. Below it, a table lists a single data asset: 'product\_for\_index' (Version 1). There are 'Search', 'Archive', and 'Show archived data' buttons at the top of the table.

## 2-1. Add data and create a search index

### Step 02: Create vector index for gpt-4o-mini model

8. From the **Index configuration** tab, select the **Select Azure AI Search service** dropdown menu ,then select **aiserach-microretail**.
9. Enter **product-vector-index** for the **Vector index** name, then select **Next**.

Add your data PREVIEW

Source data

Index configuration

Search settings

Review and finish

Index settings

Configure your index

Index storage \*

Azure AI Search

Select Azure AI Search service \*

aiserachmicroretail

Create a new Azure AI Search resource

Vector index \*

product-vector-index

Schedule updates \*

One time indexing (no scheduled updates)

Virtual machine \*

Auto select  Select from recommended options  Select from all options

Selecting a virtual machine will incur additional costs.

Back Next Create vector index Cancel

Add your data PREVIEW

Connect an existing resource

Browse resources  Enter manually

Search for a resource

Displaying (3) resources

| Name                 | Resource group |
|----------------------|----------------|
| aiserach-microretail | 0526lab-test   |
| Location             | eastus2        |
| Subscription         | standard       |
| Semantic search free |                |

Add connection

Authentication

API key

Your hub will be granted access to this resource. Anyone with access to your project or hub will be able to use this resource.

## 2-1. Add data and create a search index

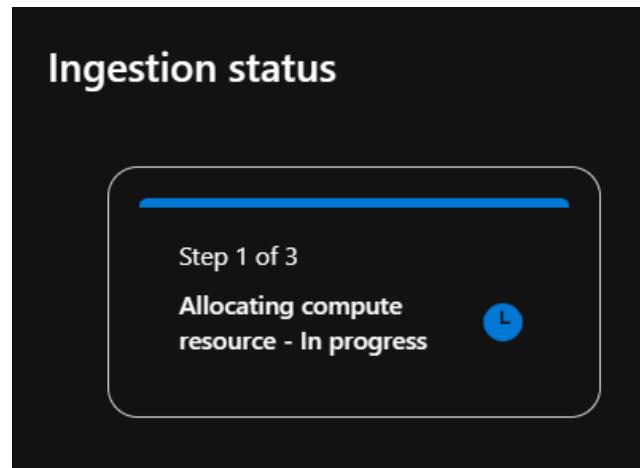
### Step 02: Create vector index for gpt-4o-mini model

10. From the **Search settings** tab, leave the default settings, then select **Next**.

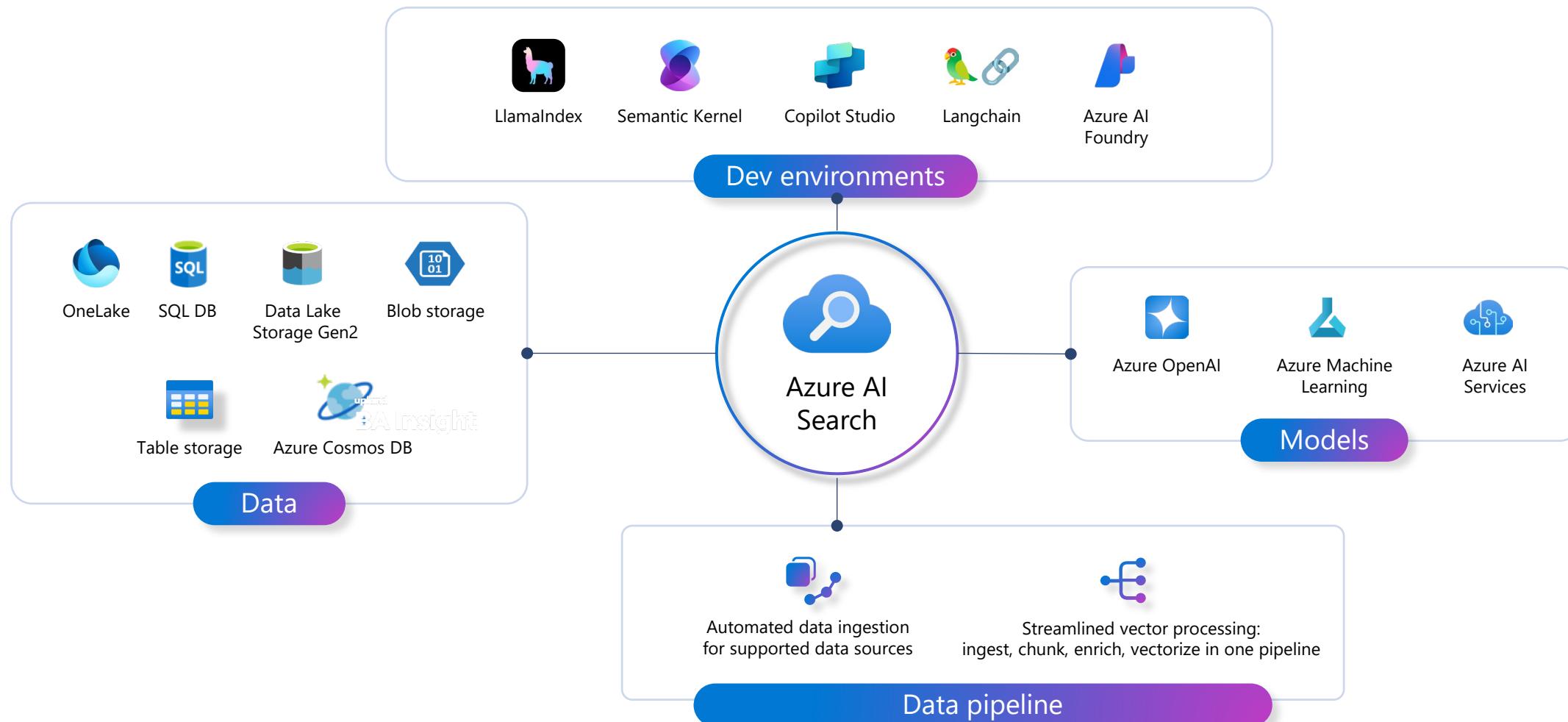
11. From the **Review and finish** tab, select **Create vector index**.

Note:

Wait for the index to be created. This should take about 3-5 minutes.  
The Ingestion status will show you the progress.



# Seamless integrations for your GenAI deployments



# Task 03: Use Playground to chat with your data

## Introduction:

With indexed data in place, **MicroRetail** can leverage the Azure AI Playground to test and refine chatbot interactions. The Playground enables real-time testing, ensuring the chatbot returns relevant responses based on customer inquiries.

## Description:

In this task, you'll use the Azure AI Playground to chat with your indexed data. This will allow you to validate response accuracy and optimize chatbot interactions.

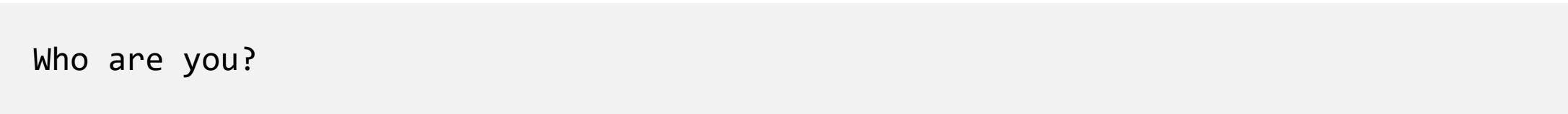
## Success Criteria:

- The chat returns output based on the indexed data when queried.

## 2-1. Use Playground to chat with your data

### Step 01: Use the system message in the Playground to test responses

- Once the data has been indexed, select the chat box on the right of the playground and enter the following query:



- Select Enter and take note of the response. This is referencing the information provided in the system message, shown to the left.

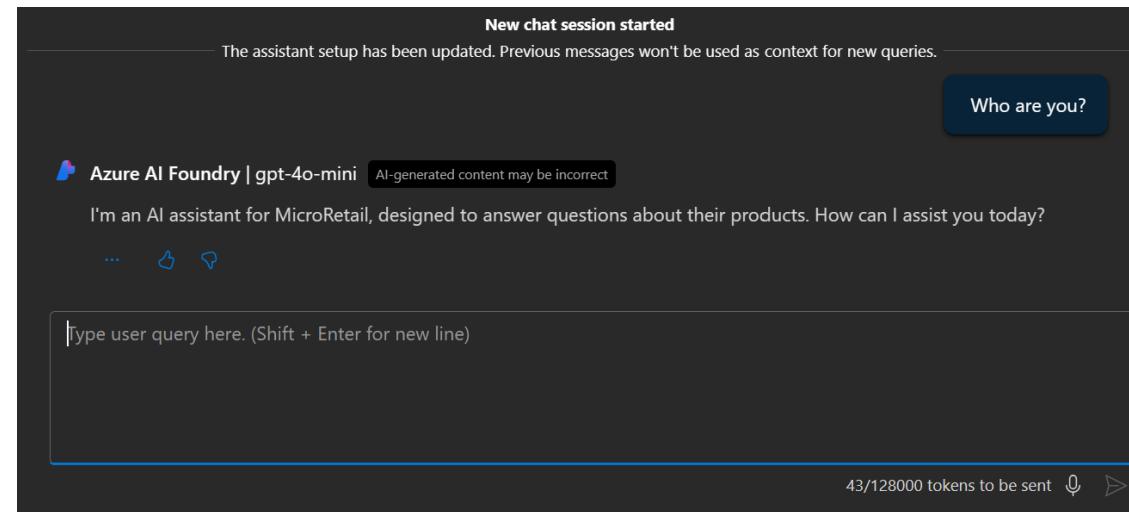
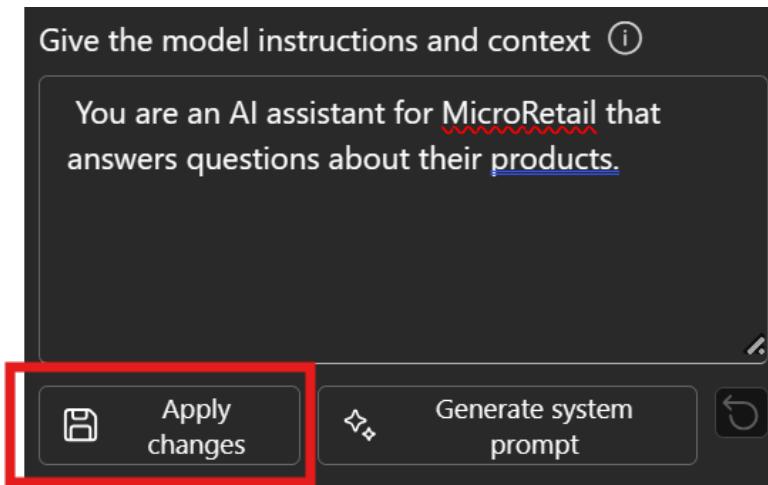
A screenshot of the Chat playground interface. On the left, the "Deployment" section shows "gpt-4o-mini (version:2024-07-18)". Below it, the "Give the model instructions and context" section contains the text: "You are an AI assistant that helps people find information.". On the right, the "Chat history" section shows a message from "Azure AI Foundry | gpt-4o-mini" with the text: "I am an AI assistant designed to help you find information and answer your questions using retrieved documents. How can I assist you today?". There are three small icons below the message: three dots, a thumbs up, and a thumbs down. At the bottom, there are buttons for "Apply changes" and "Generate system prompt".

## 2-1. Use Playground to chat with your data

### Step 01: Use the system message in the Playground to test responses

3. Change the system message to the following, then select **Apply changes**:

You are an AI assistant for MicroRetail that answers questions about their products.



4. Run the **Who are you?** query again and note the updated response.

The system message can provide more than just context for the purpose of the tool, it can also be used to influence how the model behaves and responds to queries.

## 2-1. Use Playground to chat with your data

### Step 01: Use the system message in the Playground to test responses

5. Next, run the following query:

Show me a list of products.

The response should return a list of products pulled from the **products.xlsx** file.

A screenshot of the Azure AI Foundry playground interface. The top bar shows "Chat history" and three icons: a gear, a speech bubble, and a refresh arrow. Below the input field, there's a status bar with the AI logo, "Azure AI Foundry | gpt-4o-mini", and a note "AI-generated content may be incorrect". The main area displays a list of products. The first item is "1. TrailMaster X4 Tent" with details: Price: \$250, Category: Tents, Brand: OutdoorLiving, Description: A spacious tent for four, made from durable polyester with water-resistant construction and multiple features for comfort and convenience. The second item is "2. Adventurer Pro Backpack" with details: Price: \$90, Category: Backpacks, Brand: HikeMate, Description: Ergonomically designed with a 40L capacity, featuring multiple compartments and hydration system compatibility. The third item is "3. Summit Breeze Jacket" with details: Price: \$120, Category: Hiking Clothing, Brand: MountainStyle. At the bottom, a text input field says "Type user query here. (Shift + Enter for new line)".

Chat history

Show me a list of products.

Azure AI Foundry | gpt-4o-mini AI-generated content may be incorrect

Here's a list of products available:

- 1. TrailMaster X4 Tent**
  - Price: \$250
  - Category: Tents
  - Brand: OutdoorLiving
  - Description: A spacious tent for four, made from durable polyester with water-resistant construction and multiple features for comfort and convenience.
- 2. Adventurer Pro Backpack**
  - Price: \$90
  - Category: Backpacks
  - Brand: HikeMate
  - Description: Ergonomically designed with a 40L capacity, featuring multiple compartments and hydration system compatibility.
- 3. Summit Breeze Jacket**
  - Price: \$120
  - Category: Hiking Clothing
  - Brand: MountainStyle

Type user query here. (Shift + Enter for new line)

## 2-1. Use Playground to chat with your data

### Step 01: Use the system message in the Playground to test responses

6. Next, run the following query to narrow down the products to what we're looking for:

Can you recommend a good camping chair?

The response should reference the TrekMaster Camping Chair and provide information about it.

A screenshot of the Azure AI Foundry playground interface. At the top, a blue header bar contains the question "Can you recommend a good camping chair?". Below this, the AI's response is shown in a dark gray box. The AI identifies the "TrekMaster Camping Chair" from CampBuddy and lists its features:

- **Price:** \$50
- **Description:** This chair boasts sturdy construction using high-quality materials for durability. It is lightweight and portable, making it ideal for camping, picnics, or sporting events. The ergonomic design includes an adjustable recline, padded seat and backrest, integrated cup holder, and side pockets for added convenience. It's weather-resistant, easy to clean, and comes with a carry bag for easy transport<sup>1</sup>.

If you're looking for comfort and convenience, the TrekMaster Camping Chair would be a great choice for your outdoor adventures!

✓ 1 references

1 products.xlsx - Part 1

...

The chat history keeps track of the entire conversation with the chatbot. In the previous question, we didn't specify what item we wanted the price for, but the chat was able to use context from chat history to properly answer the query.

# System message and chat history

## System message

- provides more than just context for the purpose of the tool, it can also be used to influence how the model behaves and responds to queries.

## Chat history

- Keeps track of the entire conversation with the chatbot.  
The chat is able to use context from chat history to properly answer the query.

# Exercise three

Set up Azure Prompt Flow

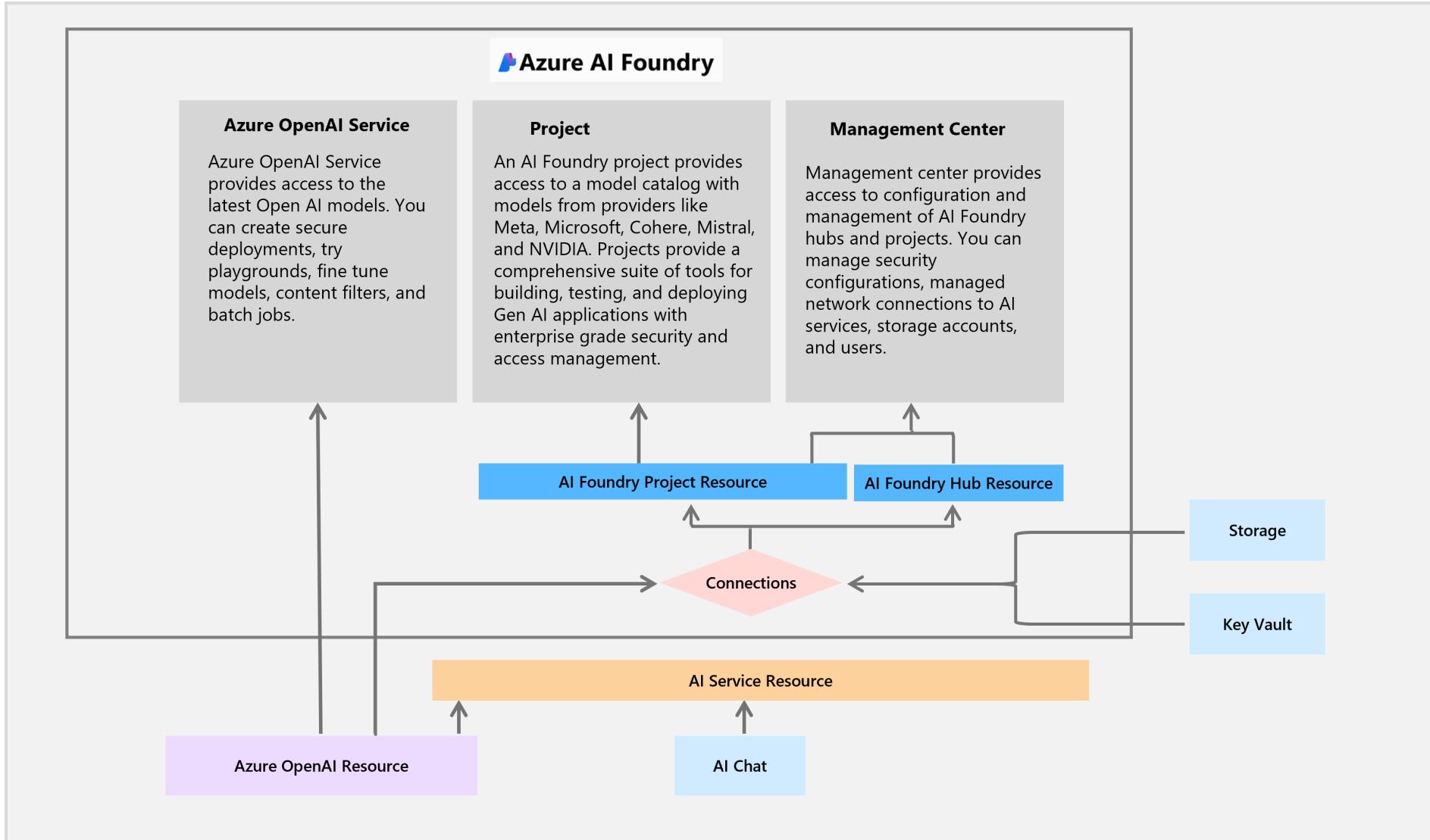
# Introduction: Set up Azure Prompt Flow

MicroRetail aims to leverage AI-driven solutions to reduce customer service response times and proactively address common issues through real-time analysis. By adhering to established best practices in Prompt Flow and content safety, MicroRetail is well-positioned to achieve these objectives.

## After completing this exercise, you'll be able to:

-  Create and manage AI-driven chat flows.
-  Test and refine chat interactions for accuracy and efficiency.
-  Evaluate chatbot responses based on predefined criteria.

# Exercise three architecture



# Task 01: Create a chat flow with chat history

## Introduction:

To enhance customer support and engagement, **MicroRetail** is implementing chat history tracking within its AI chatbot. Maintaining chat history allows for more context-aware interactions, improving user experience and reducing repetitive queries.

## Description:

In this task, you'll create a chat flow using your existing playground configuration as a template. This structured approach ensures consistency in chatbot interactions and enables tracking of previous conversations for better response accuracy.

## Success Criteria:

- The chat flow has been created successfully.

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

1. Return to the tab with Azure portal.
2. In the search bar at the top, enter **storage** and then select **Storage accounts**.
3. Select the storage account, then select **Security + networking > Networking** from the left menu.
4. For **Public network access**, select **Enabled from all networks**, then select **Save**.

The screenshot shows the Azure Storage account settings for 'azureaihub3976591542'. The 'Networking' tab is selected in the left sidebar. On the right, under 'Public network access', the radio button for 'Enabled from all networks' is selected and highlighted with a red box. A 'Save' button is also highlighted with a red box. A status message indicates that firewall settings will remain in effect for up to a minute after saving.

Home > Storage center | Storage accounts (Blobs) > azureaihub3976591542

azureaihub3976591542 | Networking

Storage account

Search

Events

Storage browser

Storage Mover

Partner solutions

Resource visualizer

Data storage

Security + networking

Networking

Front Door and CDN

Access keys

Shared access signature

Encryption

Microsoft Defender for Cloud

Data management

Firewalls and virtual networks

Private endpoint connections

Custom domain

Save

Discard

Refresh

Give feedback

Firewall settings restricting access to storage services will remain in effect for up to a minute after saving update.

Public network access

Enabled from all networks

Enabled from selected virtual networks and IP addresses

Disabled

All networks, including the internet, can access this storage account. [Learn more](#)

Network Routing

Determine how you would like to route your traffic as it travels from its source to an Azure endpoint. Microsoft most customers.

Routing preference \*

Microsoft network routing

Internet routing

Publish route-specific endpoints

Microsoft network routing

Internet routing

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

5. Select **Settings > Configuration** from the left menu.
6. For **allow storage account key access**, select **Enabled**, then select **Save**.

The screenshot shows the Azure Storage account configuration page for 'azureaihub3976591542'. The left sidebar shows navigation options like Data storage, Security + networking, and Configuration, with Configuration selected. The main area displays account settings: Account kind (StorageV2), Performance (Standard selected), Secure transfer required (Enabled), Allow Blob anonymous access (Disabled), and the 'Allow storage account key access' setting, which is highlighted with a red box and has 'Enabled' selected. A second red box highlights the 'Save' button at the top right of the configuration pane.

Home > Storage center | Storage accounts (Blobs) > azureaihub3976591542

## azureaihub3976591542 | Configuration

Storage account

Search

Save Discard Refresh Give feedback

The cost of your storage account depends on the usage and the options you choose below. [Learn more](#)

Account kind: StorageV2 (general purpose v2)

Performance: Standard

Secure transfer required: Enabled

Allow Blob anonymous access: Disabled

Allow storage account key access: Enabled

Data storage

Security + networking

- Networking
- Front Door and CDN
- Access keys
- Shared access signature
- Encryption
- Microsoft Defender for Cloud

Data management

Settings

Configuration

Data Lake Gen2 upgrade

Resource sharing (CORS)

# 3-1. Create a chat flow with chat history

## Step 01: Create prompt flow using Playground as a template

7. Return to the tab with **Chat playground** and select **Prompt flow** from the left menu.
8. Select **+ Create**, then select **Upload** under **Upload from local**.
9. Download the **chatflow.zip** file [HERE](#).
10. On the **Upload from local** pane, select **Zip file**, then select **Browse** and upload the **chatflow.zip** file.
11. Name the folder **chatflow\_sample**, set the **Select flow type** field to **Chat flow**, and select **Upload**.

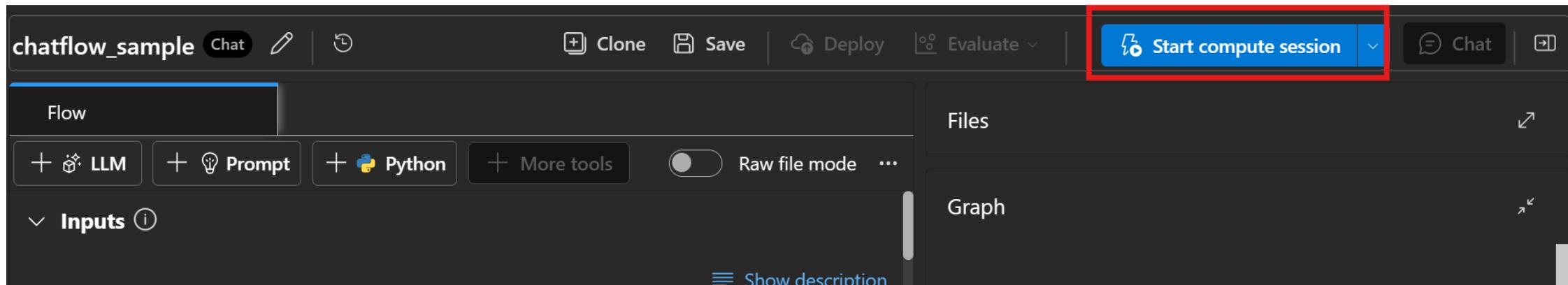
The screenshot shows the 'Create a new flow' section with three options: Standard flow, Chat flow (selected), and Evaluation flow. Below this is the 'Explore gallery' section with various flow templates like Multi-Round Q&A on Your Data, QnA Groundedness Evaluation, etc. At the bottom is the 'Import' section with 'Import from file share' and 'Upload from local' (button highlighted with a red box).

The screenshot shows the 'Upload from local' dialog box. It includes fields for 'Upload from' (radio buttons for 'Folder' and 'Zip file' - the latter is highlighted with a red box), 'chatflow.zip' (file path highlighted with a red box), 'Location to store flow' (set to 'Users/yunhuichu/promptflow'), 'Folder name' ('chatflow\_sample' highlighted with a red box), 'Select flow type' ('Chat flow' highlighted with a red box), and 'Upload' (button highlighted with a red box) and 'Cancel' buttons at the bottom.

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

12. Once the chat flow is created, select **Start compute session** in the top right.



The prompt flow page contains two main panes. On the left is the flow pane, which allows you to add and configure new flows with LLMs, prompts, and various Python tools. The right pane contains a graph, allowing you to easily visualize the flow of different nodes. The graph will update dynamically as updates are made in the flow pane.

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

13. In the left pane, find the **rewriteIntent** node and connect it to your hub. Ensure the deployment is set to **gpt-4o-mini** and the response it set to **text**.
14. Next, find the **generateReply** node and configure the same settings.

The image shows two nodes in the Azure AI Playground:

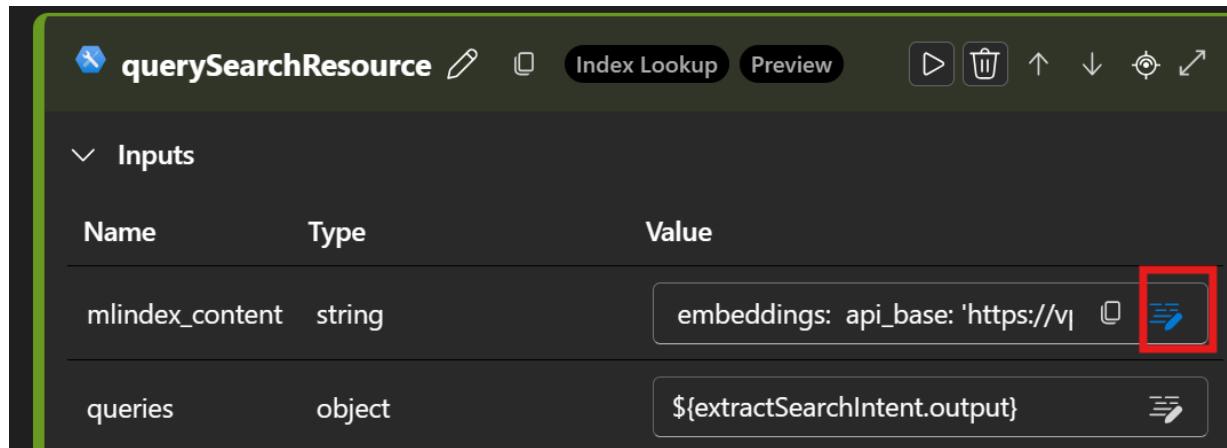
- rewriteIntent** node configuration:
  - Connection: azureaihub6339708796\_aoui
  - deployment\_name: gpt-4o-mini
  - response\_format: {"type": "text"}
- generateReply** node configuration:
  - Connection: azureaihub6339708796\_aoui
  - deployment\_name: gpt-4o-mini
  - response\_format: {"type": "text"}

\*\*Ensure the compute instance has started before continuing.

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

15. Next, find the **querySearchResource** node and select the edit icon next to **mlindex\_content**.



### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

16. Set the **acs\_index\_connection** and set it to the **aiserach-microretail** index. Ensure the remaining fields match the screenshot below and select **Save**.

Generate

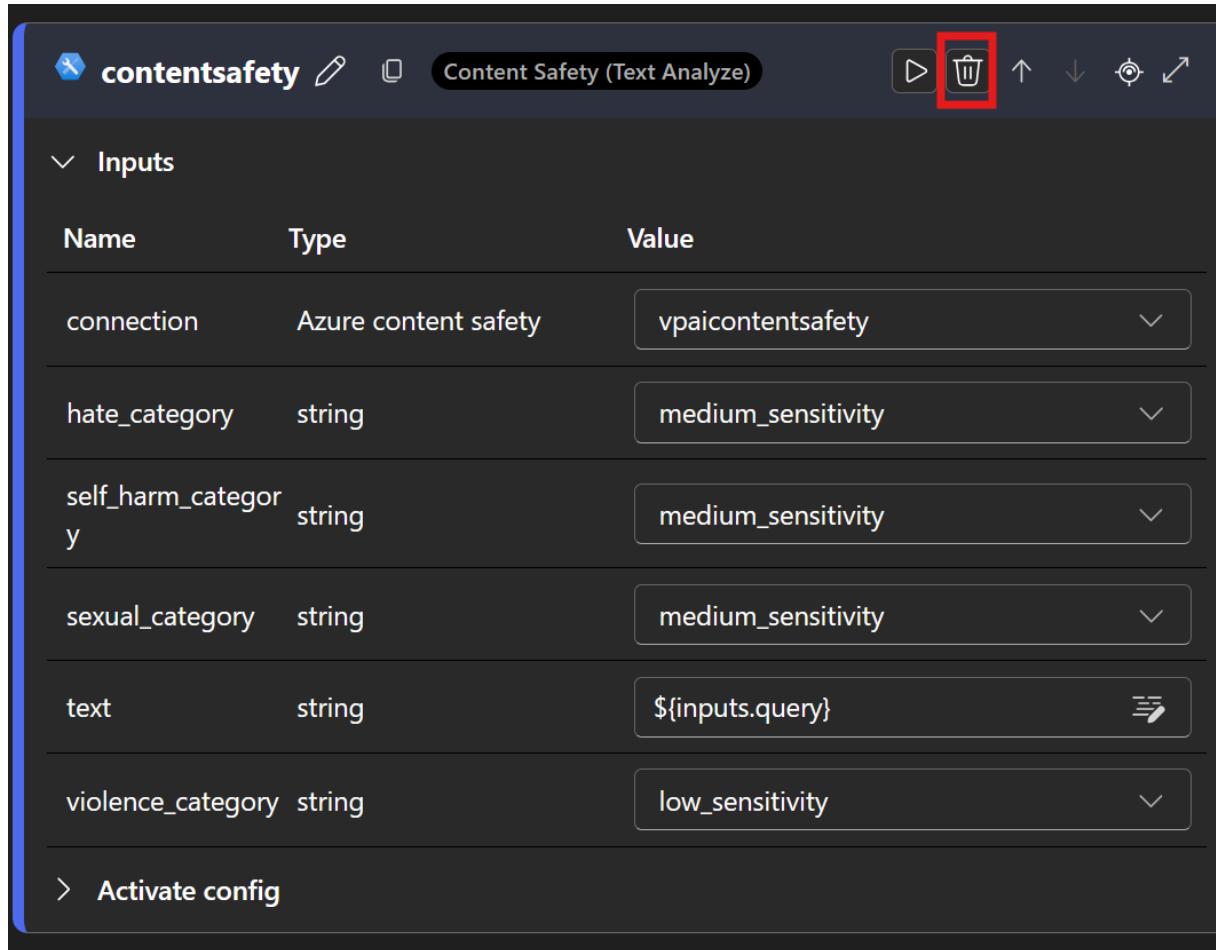
| Name                          | Type            | Value                     |
|-------------------------------|-----------------|---------------------------|
| index_type                    | string          | Azure AI Search           |
| acs_index_connection          | Azure AI Search | aiserachmicroretail       |
| acs_index_name                | string          | product-vector-index      |
| acs_content_field             | string          | content                   |
| acs_embedding_field           | string          | contentVector             |
| acs_metadata_field            | string          | meta_json_string          |
| semantic_configuration        | string          | azureml-default           |
| embedding_type                | string          | Azure OpenAI              |
| aoai_embedding_connectio<br>n | Azure OpenAI    | azureaihub4149210246_aoai |
| embedding_deployment          | string          | text-embedding-ada-002    |

**Save** **Cancel**

### 3-1. Create a chat flow with chat history

#### Step 01: Create prompt flow using Playground as a template

17. Find the **contentsafety** node at the bottom and delete it for now. We'll add this in a later step.



The screenshot shows the configuration interface for the 'contentsafety' node. At the top, there's a title bar with the node name 'contentsafety', a pencil icon for editing, a copy icon, and a 'Content Safety (Text Analyze)' button. To the right of the title bar are icons for moving, deleting, and other node operations. A red box highlights the delete icon (a trash can). Below the title bar, a section titled 'Inputs' is expanded, showing a table of input parameters:

| Name               | Type                 | Value              |
|--------------------|----------------------|--------------------|
| connection         | Azure content safety | vpacontentsafety   |
| hate_category      | string               | medium_sensitivity |
| self_harm_category | string               | medium_sensitivity |
| sexual_category    | string               | medium_sensitivity |
| text               | string               | \${inputs.query}   |
| violence_category  | string               | low_sensitivity    |

At the bottom left of the configuration area, there's a link labeled '> Activate config'.

# Prompt Flow Types: Standard vs. Chat

| Feature                  | Standard Flow                               | Chat Flow  |
|--------------------------|---|--|
| <b>Purpose</b>           | General-purpose LLM applications            | Designed for chat-based applications   |
| <b>Input/Output</b>      | Single-turn input/output                    | Multi-turn with <code>chat_history</code> , <code>chat_input</code> , and <code>chat_output</code> |
| <b>UI Experience</b>     | Traditional form-based input                | Chat-like interface for development and testing  |
| <b>Use Case Examples</b> | Document summarization, classification, Q&A | Virtual assistants, customer support bots, multi-turn Q&A  |
| <b>Debugging</b>         | Standard logs and outputs                   | Simulates real-time chat interactions for debugging  |

# Notes on Prompt Flow in Chat playground

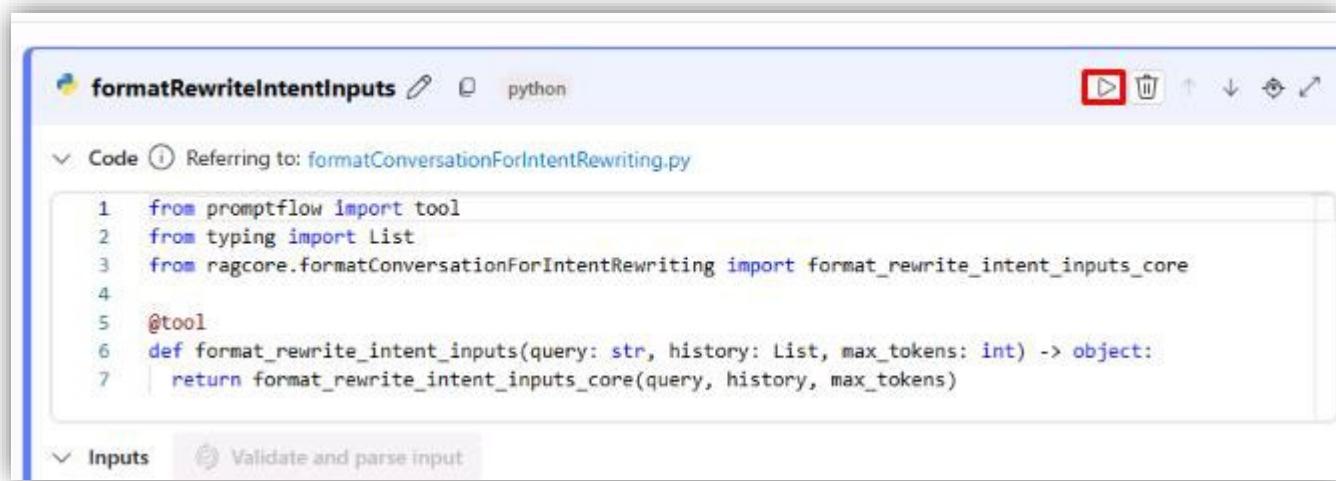
## Efficiency

- Deploying a chat flow from the playground using its configuration as a template is quicker and requires less configuration than creating the prompt flow manually.

## Two panes

- Flow pane (left) - add and configure new flows with LLMs, prompts, and various Python tools.
- Graph (right) - easily visualize the flow of different nodes.

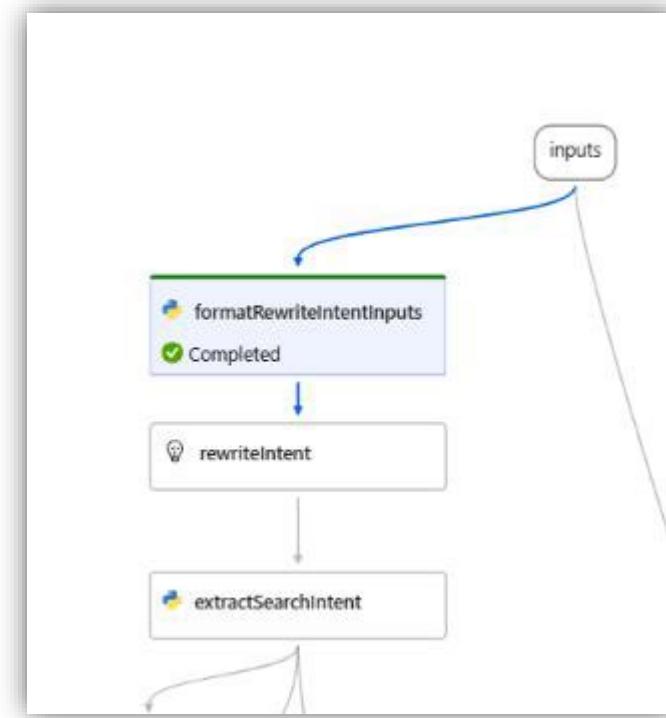
# Location of search service



A screenshot of a code editor window titled "formatRewriteIntentInputs". The code is written in Python and defines a function that takes a query, history, and max\_tokens as input and returns an object. The code editor interface includes tabs for "Code" and "Inputs", and buttons for running, deleting, and navigating.

```
1 from promptflow import tool
2 from typing import List
3 from ragcore.formatConversationForIntentRewriting import format_rewrite_intent_inputs_core
4
5 @tool
6 def format_rewrite_intent_inputs(query: str, history: List, max_tokens: int) -> object:
7     return format_rewrite_intent_inputs_core(query, history, max_tokens)
```

Test a single flow at a time and see the status of the test in the graph



# Testing Prompt Flow

## Outputs

- Use the Outputs dropdown menu to test the connection of your flow  
As multiple items in a flow link together, ensuring the correct input and output values for each is important.
- Check the outputs of the different flows to get a better idea of how information is being linked and passed between them.

# Task 02: Test the chat flow

## Introduction:

To ensure optimal performance, **MicroRetail** needs to test its chat flow before deployment. Testing allows the identification of errors, inconsistencies, and areas for improvement, ensuring a seamless user experience..

## Description:

In this task, you'll test various elements of the chat flow to ensure it's ready for deployment. This includes verifying input/output accuracy, response latency, and conversation coherence.

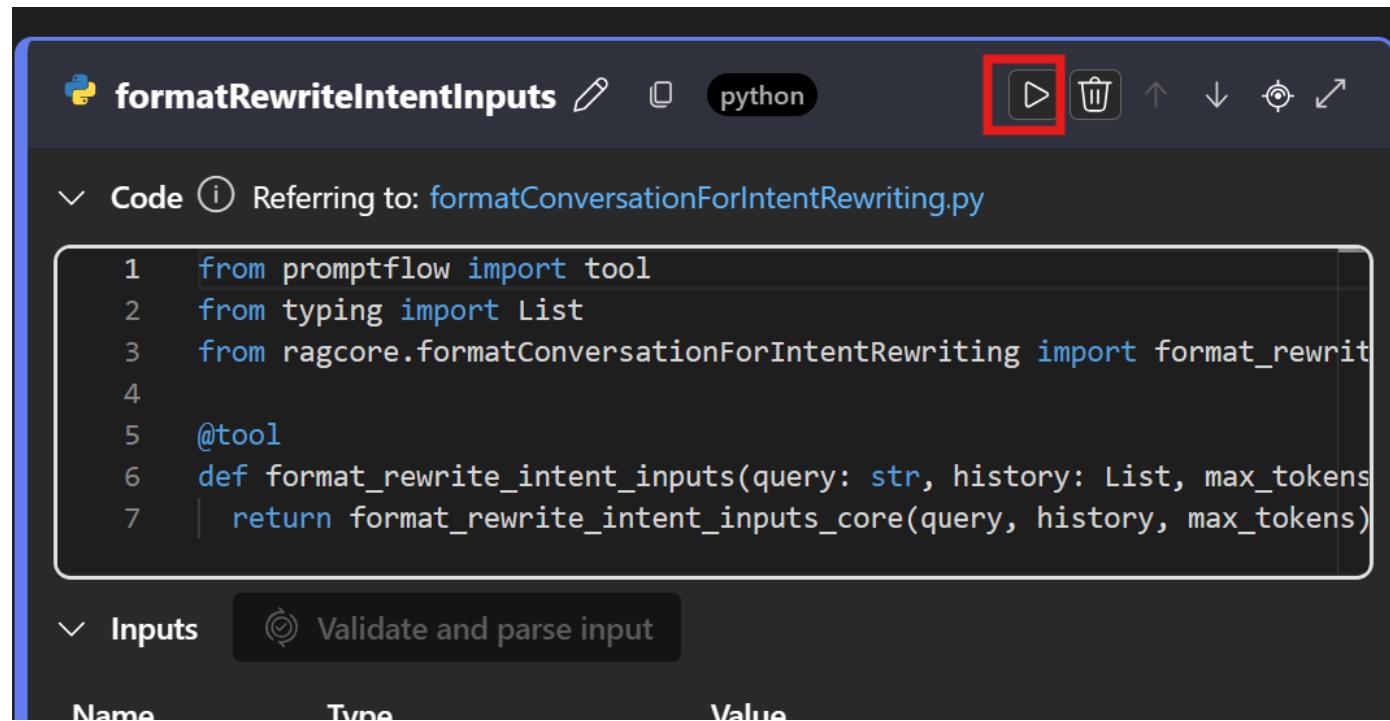
## Success Criteria:

- The chat flow is working properly.

## 3-2. Test the chat flow

### Step 01: Test the individual items using the flow pane

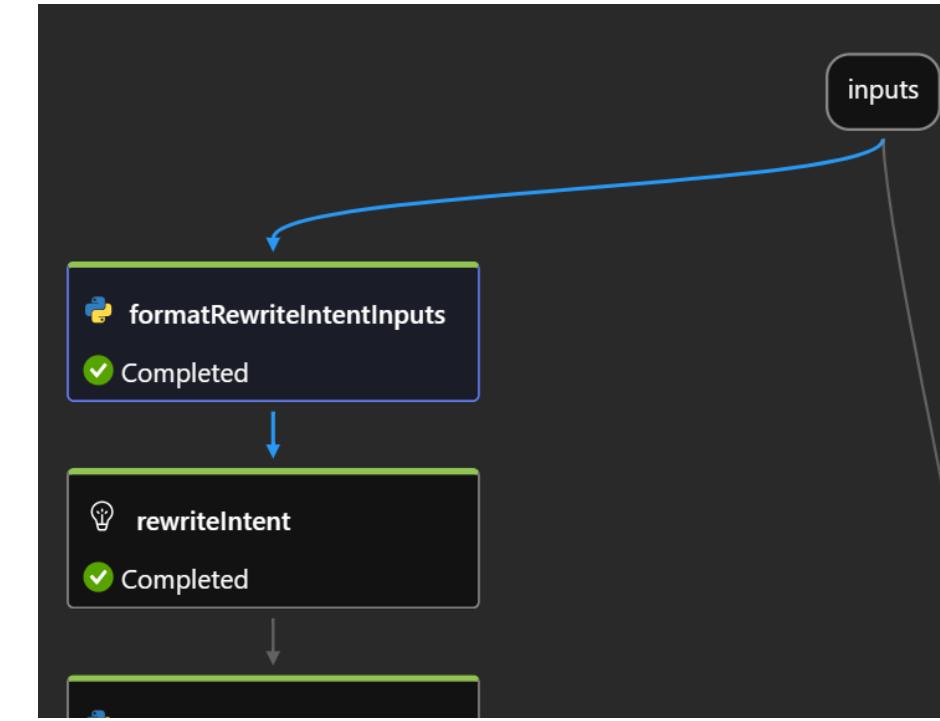
- Once the compute session is running, scroll to the first item in the flow pane on the left, **formatRewriteIntentInputs** and select the play button in the upper right.



The screenshot shows the 'formatRewriteIntentInputs' item in the flow pane. The code editor displays a Python script:

```
1 from promptflow import tool
2 from typing import List
3 from ragcore.formatConversationForIntentRewriting import format_rewrite
4
5 @tool
6 def format_rewrite_intent_inputs(query: str, history: List, max_tokens: int) ->
7     return format_rewrite_intent_inputs_core(query, history, max_tokens)
```

The toolbar at the top includes a play button (highlighted with a red box), a trash can, and other icons. Below the code editor, there's an 'Inputs' section with a 'Validate and parse input' button. At the bottom, there's a table with columns 'Name', 'Type', and 'Value'.

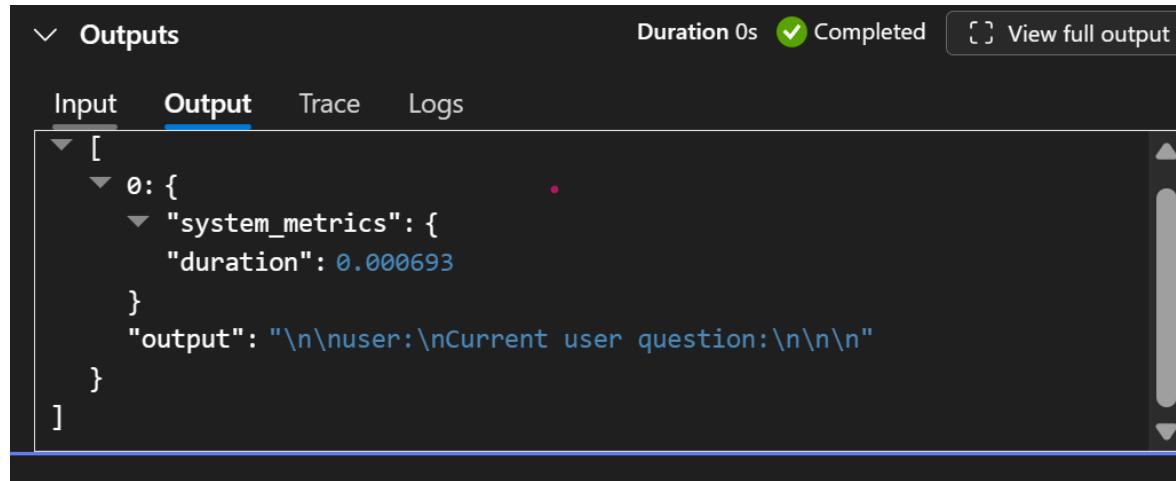


This will allow you to test a single flow at a time. This is useful when building a new flow and connecting the individual elements. The graph to the right will also show the status of the test.

## 3-2. Test the chat flow

### Step 01: Test the individual items using the flow pane

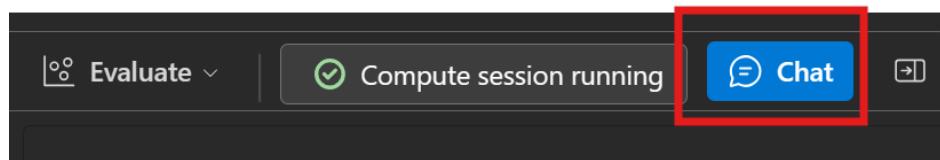
2. Once the test has completed, select the **Outputs** dropdown at the bottom of the **formatRewriteIntentInputs** flow.



The screenshot shows the 'Outputs' pane of a tool. At the top, there are tabs for 'Input', 'Output', 'Trace', and 'Logs'. The 'Output' tab is selected, indicated by a blue underline. Below the tabs, the status bar shows 'Duration 0s' and 'Completed' with a green checkmark. There is also a 'View full output' button. The main area displays a JSON-like structure representing the flow's output:

```
[  
  0: {  
    "system_metrics": {  
      "duration": 0.000693  
    },  
    "output": "\n\nuser:\nCurrent user question:\n\n"  
  }  
]
```

3. To test the entire flow, select the **Chat** button in the upper right. This will bring up a chat window over the flow visualization graph



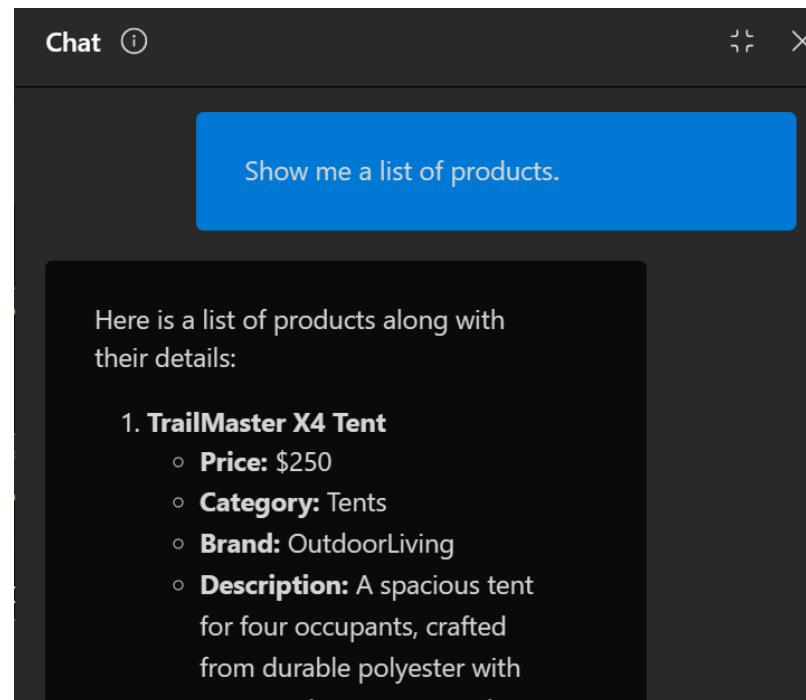
## 3-2. Test the chat flow

### Step 01: Test the individual items using the flow pane

4. In the chat, run the following query to test the overall functionality of the flow:

Show me a list of products.

The response should show a list of products, like in the playground.



Each of the items in the flow pane will now have output data relevant to an actual test. You can check the outputs of the different flows to get a better idea of how information is being linked and passed between them.

5. Close the chat window with the **X** in the upper right to return to the flow visualization graph. Note that all flow items show as **Completed** as a result of running the entire flow.

# Task 03: Evaluate the chat flow

## Introduction:

Evaluating chatbot interactions ensures that responses align with **MicroRetail's** customer service goals. Using predefined criteria, Adatum can measure and refine chatbot accuracy, relevance, and coherence.

## Description:

In this task, you'll assess chatbot performance using key evaluation metrics. By setting up automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity, **MicroRetail** can ensure that the chatbot maintains high-quality interactions and meets user expectations.

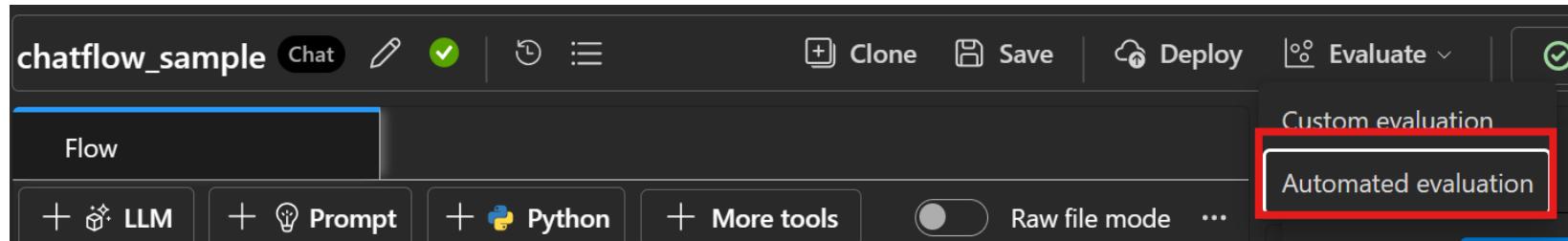
## Success Criteria:

- The flow evaluation is completed successfully.

### 3-3. Evaluate the chat flow

#### Step 01: Set up the automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity.

1. Download the **eval.jsonl** file [HERE](#).
2. On the **chatflow\_sample** page, select **Evaluate** and then select **Automated evaluation**.



### 3-3. Evaluate the chat flow

#### Step 01: Set up the automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity.

3. On the **Basic information** tab, set the **Evaluation name** to **eval\_aiworkshop** and select **Next**.
4. On the **Configure test data** tab, select **+ Add your dataset**.
5. Select **Upload file** and select the **eval.jsonl** file.
6. Once the data loads, ensure the **chat\_history** value is set to  **\${data.chat\_history}** and the **query** value is set to  **\${data.query}**, then select **Next**.

The screenshot shows the 'Configure test data' step of the AI Workshop setup wizard. It includes sections for adding datasets, previewing dataset rows, mapping dataset columns to prompt flow inputs, and navigation buttons.

**Add your dataset:** A red box highlights the 'Upload a file' button, which is also highlighted with a blue border. Below it is a 'Drag and drop CSV or JSONL file here' area and an 'Upload file' button.

**Preview of top 3 rows from your dataset:**

| query                             | ground_truth                       | response                           | chat_history | context                          |
|-----------------------------------|------------------------------------|------------------------------------|--------------|----------------------------------|
| Which tent is the most waterpr... | The Alpine Explorer Tent has th... | The Alpine Explorer Tent is kno... |              | When choosing a tent for hars... |
| Which camping table holds the...  | The Adventure Dining Table ha...   | The Adventure Dining Table hol...  |              | A camping table should be cho... |
| How much do the TrailWalker ...   | The TrailWalker Hiking Shoes a...  | The TrailWalker Hiking Shoes ar... |              | When shopping for hiking shoe... |

**Dataset mapping for prompt flow \*** (i)

| Flow      | Input        | Type   | Dataset column  |
|-----------|--------------|--------|---|
| chatflow1 | chat_history | list   | \${data.chat_history} <input type="button" value="edit"/> |
|           | query        | string | \${data.query} <input type="button" value="edit"/>        |

**Buttons:** 'Back' (disabled), 'Next' (highlighted with a red box and blue border), and 'Cancel'.

### 3-3. Evaluate the chat flow

#### Step 01: Set up the automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity.

7. On the **Select metrics** tab, select the checkboxes for **Groundedness, Relevance, Coherence, Fluency, and Similarity**.
8. Select your connection from the **Connection** dropdown menu. The **gpt-4o-mini** model should be automatically selected.

The screenshot shows the 'Select metrics' step of a wizard. On the left, a vertical navigation bar lists four steps: 'Basic information' (done), 'Configure test data' (done), 'Select metrics' (in progress, indicated by a blue circle), and 'Review and finish'. The main area is titled 'Select metrics' with a sub-section 'AI quality (AI Assisted)'. This section contains five cards, each with a checked checkbox and a description:

- Groundedness**: Measures how well the generative AI application's generated answers align with information from the input source.
- Relevance**: Measures the extent to which the generative AI application's generated responses are pertinent and directly related to the given questions.
- Coherence**: Measures how well the generative AI application can produce output that flows smoothly, reads naturally, and resembles human-like language.
- Fluency**: Measures the language proficiency of a generative AI application's predicted answer.
- Similarity**: Measures the similarity between a source data (ground truth) sentence and the generated response by a generative AI application.

Below this section, there are three dropdown menus:

- Connection \***: azureaihub1777048799\_aoai
- Provider**: AzureOpenAI
- Deployment name/Model \***: gpt-4o-mini

### 3-3. Evaluate the chat flow

#### Step 01: Set up the automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity.

9. Scroll to the bottom and ensure the data mapping is correct, then select **Next**:

How does your dataset map to your evaluation input? \*

| Name         | Description   | Type   | Data source           |
|--------------|---|--------|-----------------------|
| context      | The source that response is generated with respect to               | string | \${data.context}      |
| response     | The response to question generated by the model as answer           | string | \${data.response}     |
| query        | A query seeking specific information                                | string | \${data.query}        |
| ground_truth | The response to question generated by user/human as the true answer | string | \${data.ground_truth} |

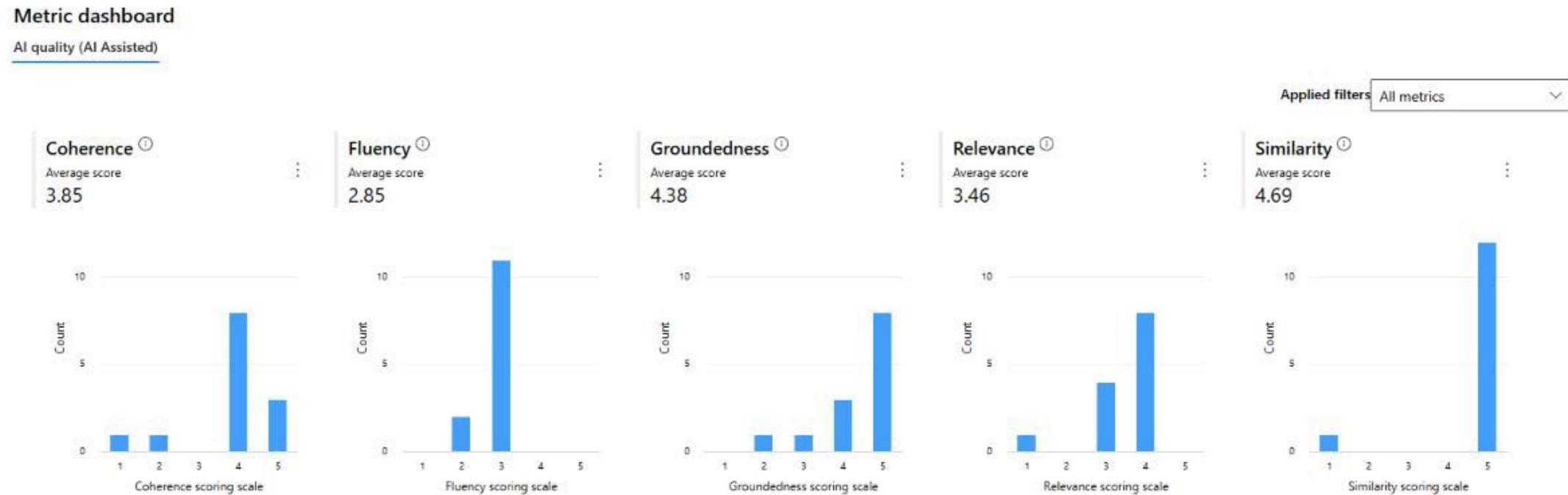
**Back** **Next** **Cancel**

### 3-3. Evaluate the chat flow

#### Step 01: Set up the automatic evaluation for Groundedness, Relevance, Coherence, Fluency, and Similarity.

10. Select **Submit** and wait for the evaluation to finish.

Once the evaluation is finished, you'll see scores for the metrics that were selected. These scores are based on the chat flow's response to the test data. You can scroll down to see more detailed information on the metric scores and the reasoning behind those scores.



# Models and metrics

## Models

- Advanced models are normally used for evaluation.
- However for this evaluation, we'll use the compact gpt-4o-mini model that our chat flow uses

## Metrics

- Scores based on chat flow's response to test data.
- More detail is available on the metric scores and the reasoning behind them.

# Task 04: Introduction to Azure AI Agent Service

## Introduction:

Azure is introducing the Azure AI Agent Service to help developers create secure, stateful AI agents that automate complex business processes. This service integrates models, tools, and data sources, making it easier for businesses to deploy autonomous agents for a range of tasks like scheduling, market research, customer service, and code management.

The service leverages a wide array of tools, including Azure Logic Apps, Azure Functions, OpenAPI 3.0, and more. It also provides secure data grounding via Bing, SharePoint, and Azure Blob, ensuring agents have accurate, up-to-date information. With multi-modal support and a flexible model selection, including OpenAI and other top-tier providers, developers can optimize their agents for specific tasks.

Azure AI Agent Service is built for enterprise use, offering features like secure data handling, compliance, and monitoring. It's designed for scalability and reliability, with built-in performance tracking and content filters to ensure safe, high-quality outputs. Additionally, the service supports multi-agent orchestration for more complex workflows.

This platform is designed to simplify agent creation, allowing businesses to rapidly develop, deploy, and manage AI agents to improve efficiency and productivity.

## Azure AI Foundry – Agent Service

### Built-in enterprise readiness

BYO-file storage

BYO-search index

OBO Authorization Support

Enhanced Observability

### Extensive Ecosystem of Tools

#### Knowledge

Microsoft Fabric

SharePoint

Bing Search

Azure AI Search

Your own licensed data 

Files (local or Azure Blob)

File Search

Code Interpreter

#### Actions

 Azure Logic Apps

 OpenAPI 3.0 Specified Tools

 Azure Functions

### Model Catalog

 Azure OpenAI Service  
(GPT-4o, GPT-4o mini)

### Models-as-a-Service

 Llama 3.1-405B-Instruct

 Mistral Large

 Cohere-Command-R-Plus

# Task 04: Introduction to Azure AI Agent Service

## **Description:**

In this task, you'll learn how to set up and deploy the Azure AI Agent Service. By following the steps, you will understand how to integrate various tools and data sources to create secure, stateful AI agents that can automate complex business processes.

## **Success Criteria:**

- The Azure AI Agent Service is set up and deployed successfully.
- The AI agents created are able to automate tasks like scheduling, market research, customer service, and code management.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

1. From the **project\_AIWorkshop** page, select **Models + endpoints** from the left menu.
2. Select **+ Deploy model**, then select **Deploy base model**.
3. Select the **gpt-4o** model, then select **Confirm**.

Select a model

Choose a model to create a new deployment. For flows and other resources, create a deployment from their respective list. [Go to model catalog](#)

Models 30    Collections    Deployment options    Inference tasks    Show description

Search bar: 4o

| Model Name                   | Type             | Status |
|------------------------------|------------------|--------|
| gpt-4o-mini-realtime-preview | Audio generation | ...    |
| gpt-4o                       | Chat completion  | ...    |
| gpt-4o-mini                  | Chat completion  | ...    |
| gpt-4o-audio-preview         | Audio generation | ...    |
| gpt-4o-realtime-preview      | Audio generation | ...    |
| ruslandev-llama-3-8b-gpt...  | Text generation  | ...    |
| mistral-medium-2505          | ...              | ...    |

Task: Chat completion

**gpt-4o**

gpt-4o offers a shift in how AI models interact with multimodal inputs. By seamlessly combining text, images, and audio, gpt-4o provides a richer, more engaging user experience.

Matching the intelligence of gpt-4 turbo, it is remarkably more efficient, delivering text at twice the speed and at half the cost. Additionally, GPT-4 exhibits the highest vision performance and excels in non-English languages compared to previous OpenAI models.

gpt-4o is engineered for speed and efficiency. Its advanced ability to handle complex queries with minimal resources can translate into cost savings and performance.

The introduction of gpt-4o opens numerous possibilities for businesses in various sectors:

1. **Enhanced customer service:** By integrating diverse data inputs, gpt-4o enables more dynamic and comprehensive customer support interactions.
2. **Advanced analytics:** Leverage gpt-4o's capability to process and analyze different types of data to enhance decision-making and uncover deeper insights.

Confirm    Cancel

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

4. Set the **Deployment type** to **Standard**, set the **Tokens per Minute Rate Limit** to **150K**, then select **Deploy**.

Deploy gpt-4o

Deployment name \*  
gpt-4o

Deployment type  
**Standard**  

Standard: Pay per API call with lower rate limits. Adheres to Azure data residency promises.  
Best for intermittent workloads with low to medium volume. Learn more about Standard deployments [\(opens in new tab\)](#).

Deployment details Collapse

Model version upgrade policy  
Upgrade once new default version becomes available

Model version  
**2024-08-06 (Default)**  

Connected AI resource  
azureaihub4149210246\_aoai

(i) 150K tokens per minute quota available for your deployment

Tokens per Minute Rate Limit (i)  
 150K

Corresponding requests per minute (RPM) = 900

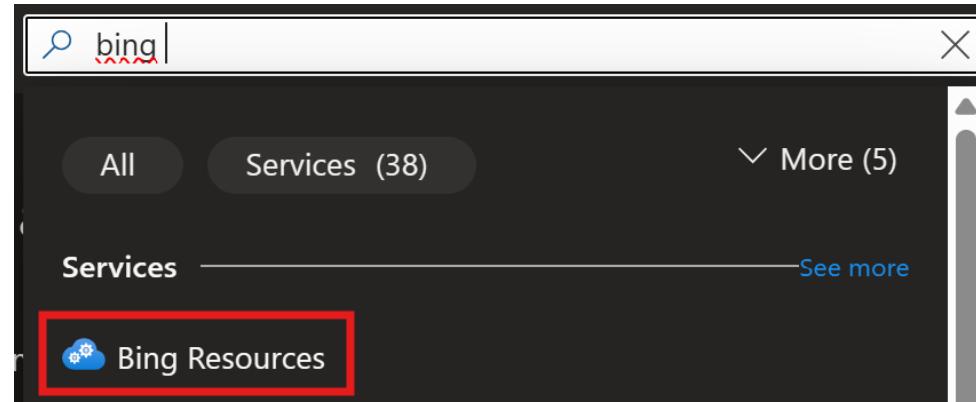
Content filter (i)  
DefaultV2

Enable dynamic quota (i)  
 Enabled

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

5. Switch to the tab with Azure portal.
6. In the search bar at the top, search for **bing** and then select **Bing Resources**.



## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

7. From the **Bing Resources** page, select **+ Add**, then select **+ Grounding with Bing Search**.
8. On the **Create a Grounding with Bing Search resource** page, select your resource group and pricing tier. Give it a name of **bingsrch-AIWorkshop** and select **Review + Create**, then select **Create**.

Home > Bing Resources >

### Create a Grounding with Bing Search resource

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Resource group \* ⓘ  Create new

Instance details

Name \*  ✓

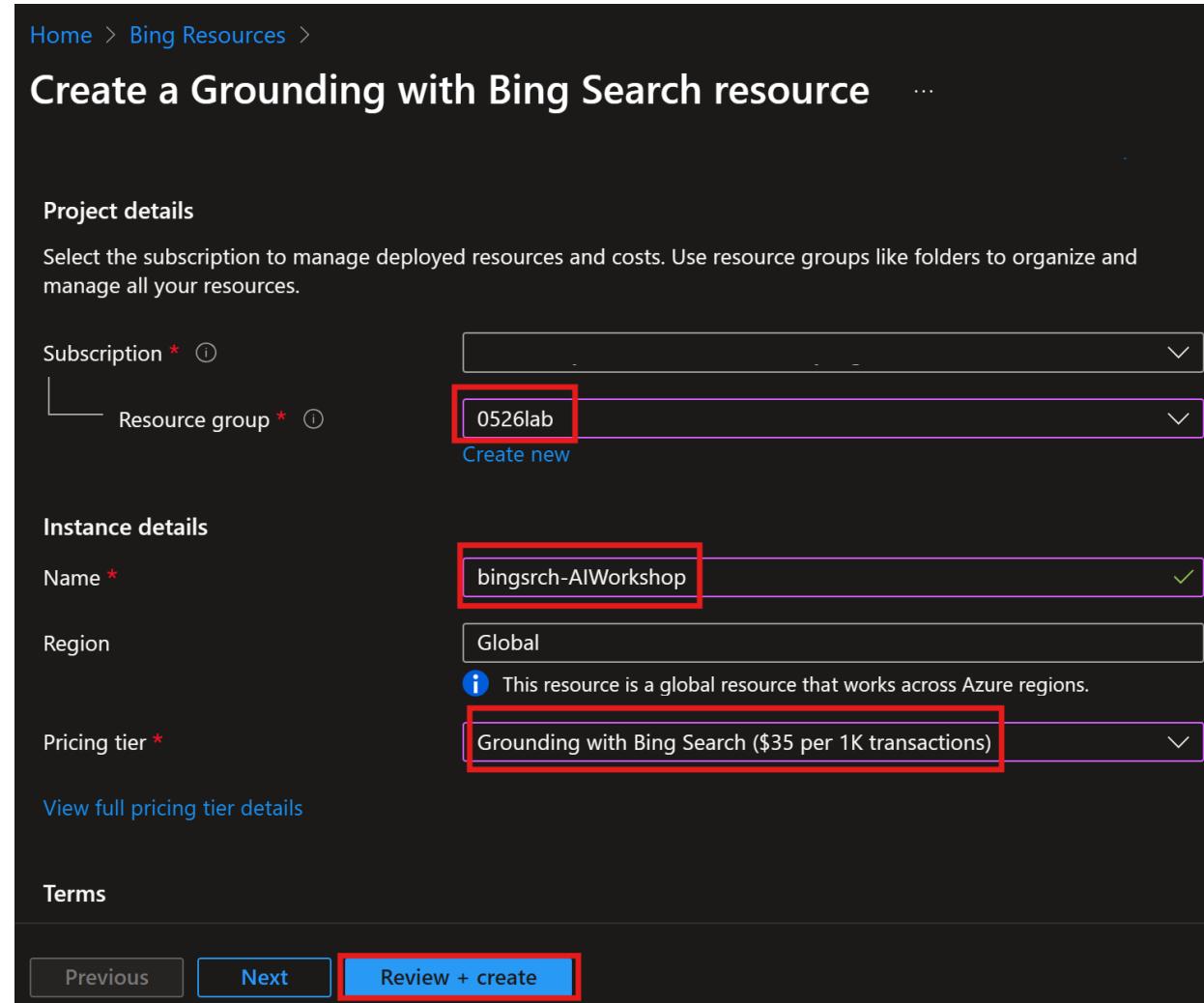
Region   
 ⓘ This resource is a global resource that works across Azure regions.

Pricing tier \*  ✓

[View full pricing tier details](#)

Terms

[Previous](#) [Next](#) [Review + create](#)



## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

9. Return to the tab with the Azure AI model deployment and select **Agents** from the left menu.
10. Under **Select an Azure OpenAI Service resource**, select your hub and select **Let's go**.

The screenshot shows the Azure AI Foundry interface. The top navigation bar includes the project name "project\_AIWorkshop0526" and a "Agents" section. The left sidebar has sections like "Overview", "Model catalog", "Playgrounds", "AI Services", "Build and customize", "Agents PREVIEW" (which is selected), "Templates", "Fine-tuning", "Content Understanding PREVIEW", "Prompt flow", "Observe and optimize", "Tracing PREVIEW", "Monitoring", "Protect and govern", "Evaluation", and "Guardrails + controls". The main content area is titled "Foundry Agent Service" and features the text "Fast, secure enterprise agents for any business process". A red box highlights the "Select an Azure OpenAI resource" dropdown, which contains the value "azureaihub4149210246\_aoai". Below the dropdown is a note: "Agents use multi-tenant search and storage resources fully managed by Microsoft. If you want to use your own resources, see here". At the bottom is a blue "Let's go" button.

Agents PREVIEW

Project  
project\_AIWorkshop0526

Overview

Model catalog

Playgrounds

AI Services

Build and customize

Agents PREVIEW

Templates

Fine-tuning

Content Understanding PREVIEW

Prompt flow

Observe and optimize

Tracing PREVIEW

Monitoring

Protect and govern

Evaluation

Guardrails + controls

Foundry Agent Service

Fast, secure enterprise agents for any business process

Select an Azure OpenAI resource \*

azureaihub4149210246\_aoai

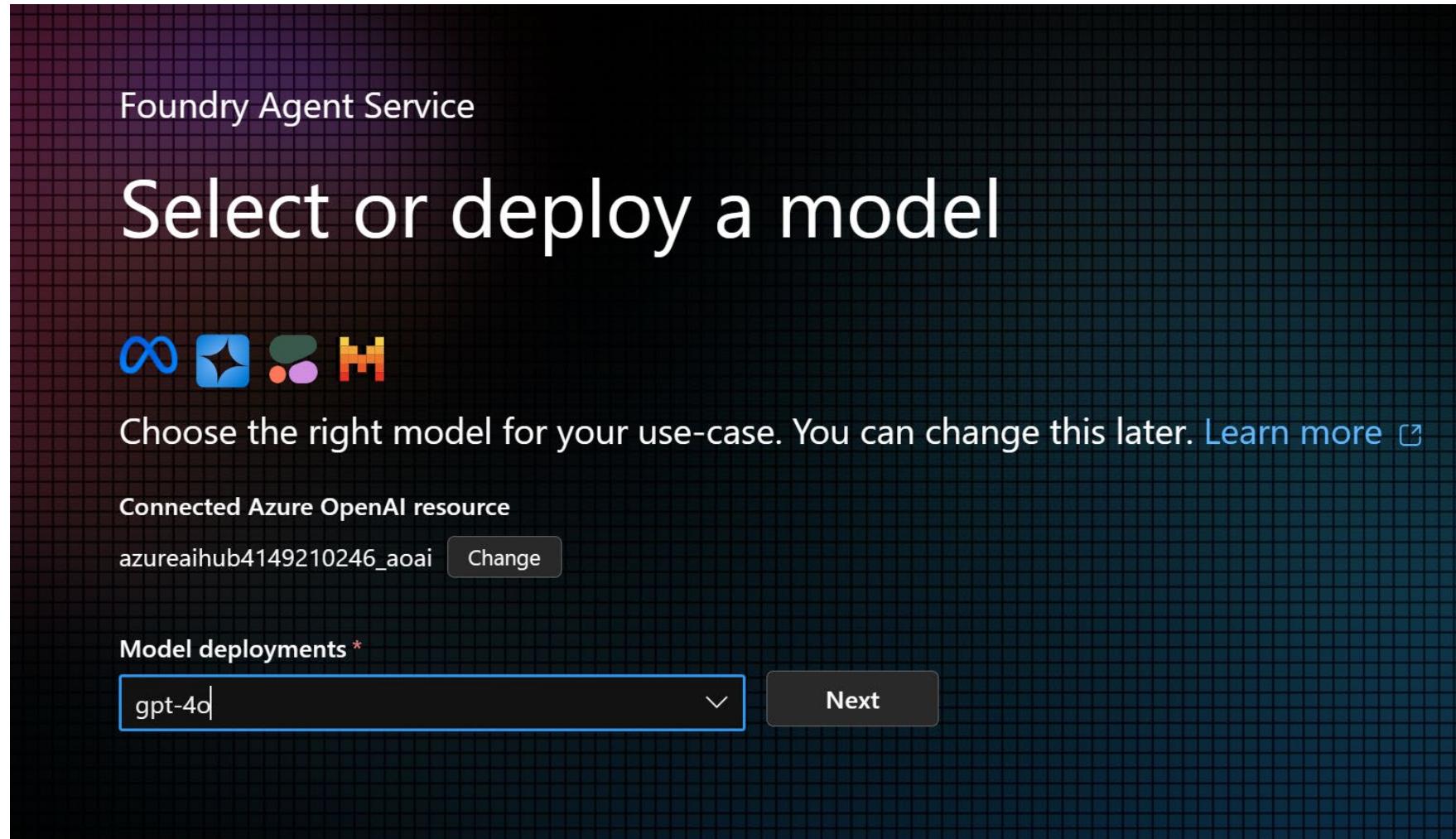
Agents use multi-tenant search and storage resources fully managed by Microsoft. If you want to use your own resources, [see here](#).

Let's go

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

11. Under **Select or deploy a model**, select **gpt-4o** then select **Next**.



## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

12. Select the agent to open the **Setup** pane.

13. In the **Instructions** field of the **Setup** pane, enter the following:

**Understand User Query:**

Analyze the user's query to identify if it requires real-time information (e.g., weather, date, news).

**Use Bing Search Tool for Real-Time Data:**

If the query involves up-to-date information, use the Bing Search tool to retrieve relevant data.

**Craft a Clear, Concise Response:**

Extract the relevant information (e.g., temperature, news) and provide the answer in a simple and direct way.

**Ask for Clarification if Needed:**

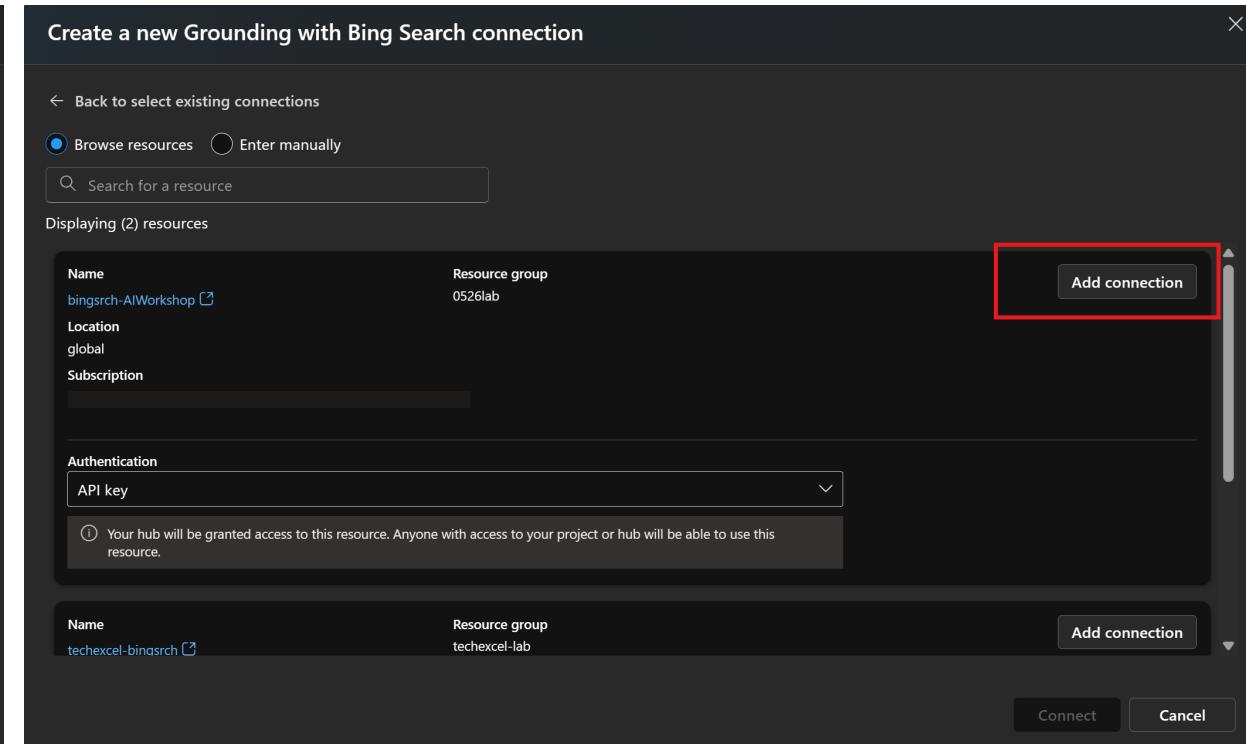
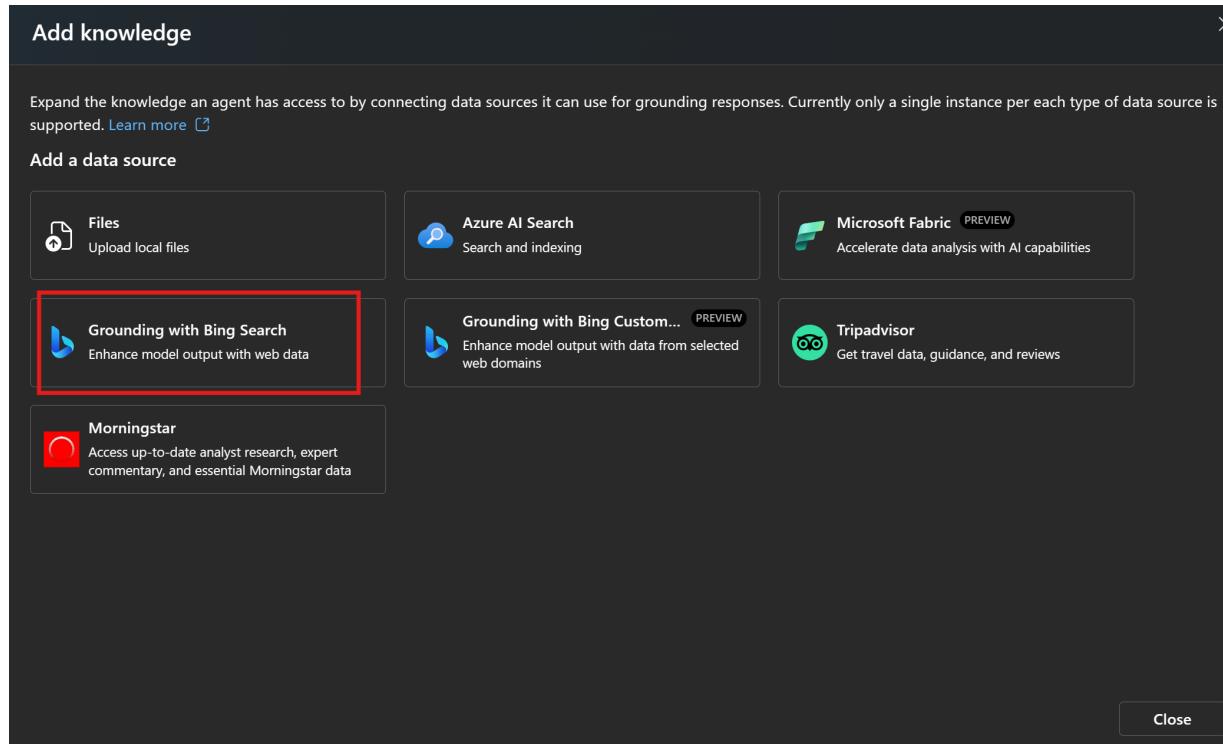
If the query is vague or missing details (e.g., location for weather), ask the user for more information.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

14. Under **Knowledge** in the **Setup** pane, select **+ Add**, then select **Grounding with Bing Search**.

15. Select the **+ Create connection** connection, then select **Add connection** next to the **bingsrch** resource.



The **Knowledge** section allows you to specify the source of information for the agent. In this case, we're using the **Grounding with Bing Search** service to retrieve up-to-date information from external sources, beyond our local dataset. You can also set the source to a pre-existing search index or local data. If you want the agent to be able to retrieve both local and web-based results, you can add a connection for each.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

16. Under **Actions** in the **Setup** pane, select **+ Add**, then select **Code interpreter**.

17. On the **Add code interpreter action** page, select **Select local files** and then select the **products.xlsx** file created earlier.

18. Select **Upload and Save**.

**Add code interpreter action**

Use code interpreter to read and interpret information from datasets, generate code, and create graphs and charts using your data. Supports up to 20 files.  
File types supported: .c, .cs, .cpp, .doc, .docx, .html, java, json, .md, .pdf, .php, .pptx, .py, .rb, .tex, .txt, .css, js, .sh, .ts, .csv, jpeg, jpg, .gif, .png, .tar, .xlsx, .xml, .zip  
(<200MB, UTF-8 BOM text file) [Learn more about code interpreter](#)

| Name | Status | Error | Size | File type | Uploaded |
|------|--------|-------|------|-----------|----------|
|      |        |       |      |           |          |



Add local files or skip for now

You can return and add files later, in the meantime code interpreter will be running and ready to help.

**Select local files**

Save Cancel

**Add code interpreter action**

Use code interpreter to read and interpret information from datasets, generate code, and create graphs and charts using your data. Supports up to 20 files.  
File types supported: .c, .cs, .cpp, .doc, .docx, .html, java, json, .md, .pdf, .php, .pptx, .py, .rb, .tex, .txt, .css, js, .sh, .ts, .csv, jpeg, jpg, .gif, .png, .tar, .xlsx, .xml, .zip  
(<200MB, UTF-8 BOM text file) [Learn more about code interpreter](#)

Uploaded 1/20

| Name          | Status  | Error   | Size     | File type | Uploaded            |
|---------------|---|---|----------|-----------|---------------------|
| products.xlsx |  |  | 18.21 KB | Local     | May 25, 2025 9:2... |

< Prev Next >

**Select local files**

**Upload and save** Cancel

The **Actions** section allows you to specify additional tasks for the agent beyond simple data retrieval. The **Code interpreter** tool can be used for tasks like performing calculations or creating visualizations from your data.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

19. From the upper right of the **Setup** pane, select **Try in playground**.

The screenshot shows the Azure AI Agent Service interface. On the left, the 'Agents' list displays a single agent named 'Agent121' with ID 'asst\_Ov02SP6jqA53czS8E7BziTUu'. On the right, the 'Setup' pane is open, containing fields for 'Agent id' (set to 'asst\_Ov02SP6jqA53czS8E7BziTUu'), 'Agent name' (set to 'Agent121'), and 'Azure OpenAI resource connection' (set to 'azureaihub4149210246\_aoai'). A red box highlights the 'Try in playground' button at the top right of the Setup pane.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

20. In the **Agents playground** chat, enter **What is the weather like in New York?**

The screenshot shows a dark-themed chat interface. At the top, a user message box contains the text "What is the weather like in New York?". Below it, an AI response box displays the current weather in New York: "Currently, in New York, the temperature is around 73°F (23°C), and it's mostly sunny. Winds are coming from the west at about 10 mph. It feels comfortable with a moderate humidity level<sup>1</sup>." Below the response are two cards: one for "current weather in New York" and another for "New York, NY Current Weather | Acc...". At the bottom of the screen, there is a message input field with the placeholder "Type user query here. (Shift + Enter for new line)". A note at the bottom left states: "Messages in the Agents playground are visible to anyone with access to this resource and using the API." On the right side of the screen, there are icons for a trash bin, a dropdown menu, a plus sign, and a right-pointing arrow.

The gpt-4o model doesn't have direct access to the current date. However, by using the Bing Search tool, the agent can retrieve up-to-date information for time-sensitive queries.

## 3-4. Introduction to Azure AI Agent Service

### Step 01: Set up Azure AI Agent Service

21. In the chat, enter **What is the average price of the products in the xlsx file?**

The screenshot shows a dark-themed chat interface. At the top, a user message bubble contains the text "What is the average price of the products in the xlsx file?". Below it, an AI response bubble starts with a thinking face icon and says "The average price of the products in the Excel file is \$118.25.". A large, semi-transparent rectangular box covers the bottom half of the screen, containing three lines of code interpreter output:

```
code_interpreter (import pandas as pd # Load the uploaded Excel file to examine its structure file_path = '/mnt/data/assistant-4XRxWjskaBUfXR1CpcnLhV' data = pd.Ex...)

code_interpreter (# Load the data from the sheet named 'products' df = data.parse('products') #
Display the first few rows to understand its structure df.head())

code_interpreter (# Calculate the average price of the products average_price = df['price'].mean())
```

The code interpreter tool allows for more complex queries about your data. In this case, we used it to retrieve the average price of the products in our data set.

# Knowledge section

Choose an existing Grounding with Bing Search connection

← Back to select knowledge type

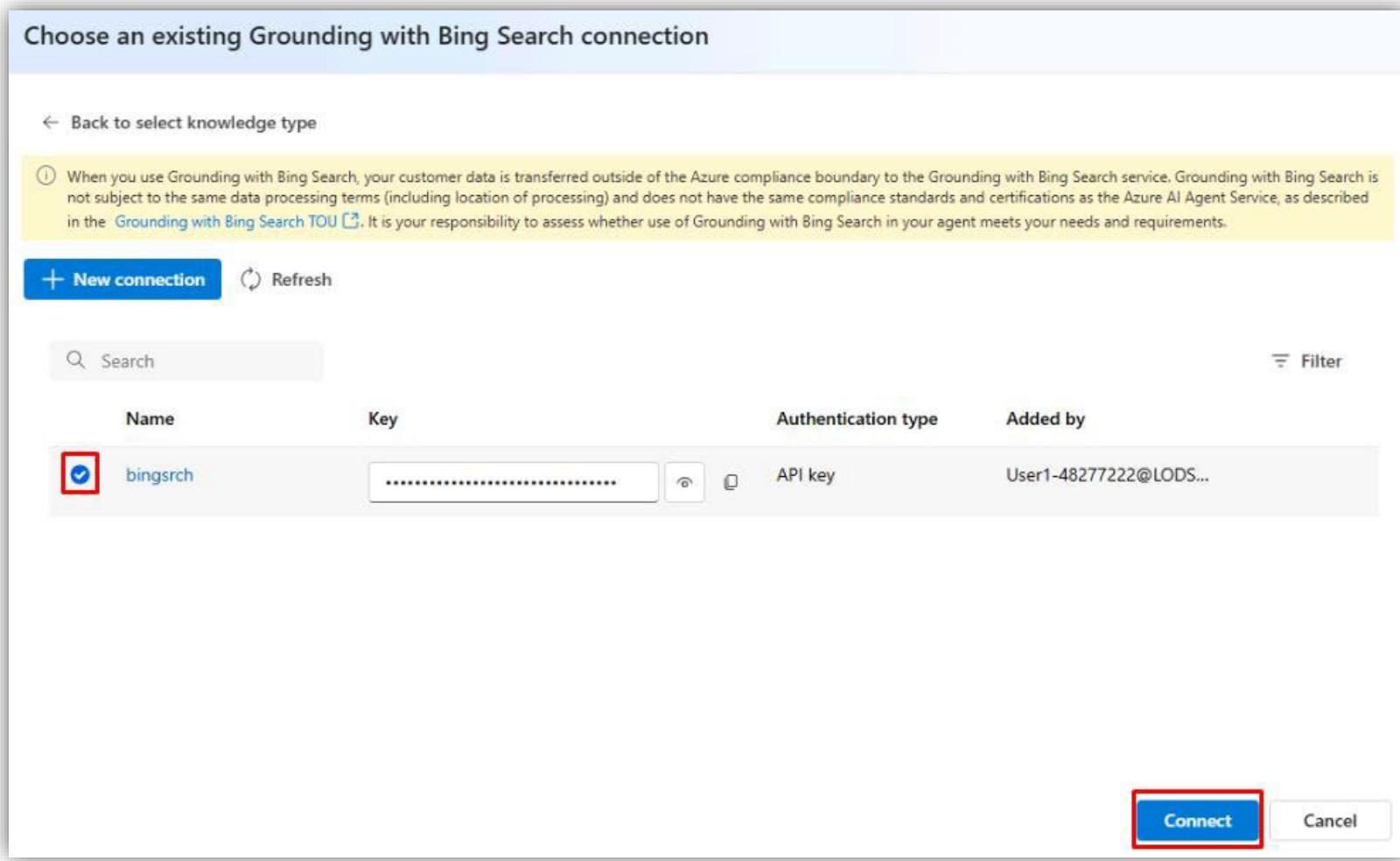
When you use Grounding with Bing Search, your customer data is transferred outside of the Azure compliance boundary to the Grounding with Bing Search service. Grounding with Bing Search is not subject to the same data processing terms (including location of processing) and does not have the same compliance standards and certifications as the Azure AI Agent Service, as described in the [Grounding with Bing Search TOU](#). It is your responsibility to assess whether use of Grounding with Bing Search in your agent meets your needs and requirements.

+ New connection   Refresh

| Name     | Key   | Authentication type | Added by               |
|----------|-------|---------------------|------------------------|
| bingsrch | ..... | API key             | User1-48277222@LODS... |

Search   Filter

Connect   Cancel



Allows you to specify the source of information for the agent.

You can set the connection to:

- Grounding with Bing Search service to retrieve up-to-date information from external sources
- A pre-existing search index or local data.
- Separate connections for local and web-based results to allow agent to retrieve from both.

# Actions section

Allows you to specify additional tasks for the agent beyond simple data retrieval

Add code interpreter action

Use code interpreter to read and interpret information from datasets, generate code, and create graphs and charts using your data. Supports up to 20 files.  
File types supported: .c, .cs, .cpp, .doc, .docx, .html, .java, .json, .md, .pdf, .php, .pptx, .py, .rb, .tex, .txt, .css, .js, .sh, .ts, .csv, jpeg, jpg, gif, png, tar, .xlsx, .xml, .zip  
(<200MB, UTF-8 BOM text file) [Learn more about code interpreter](#)

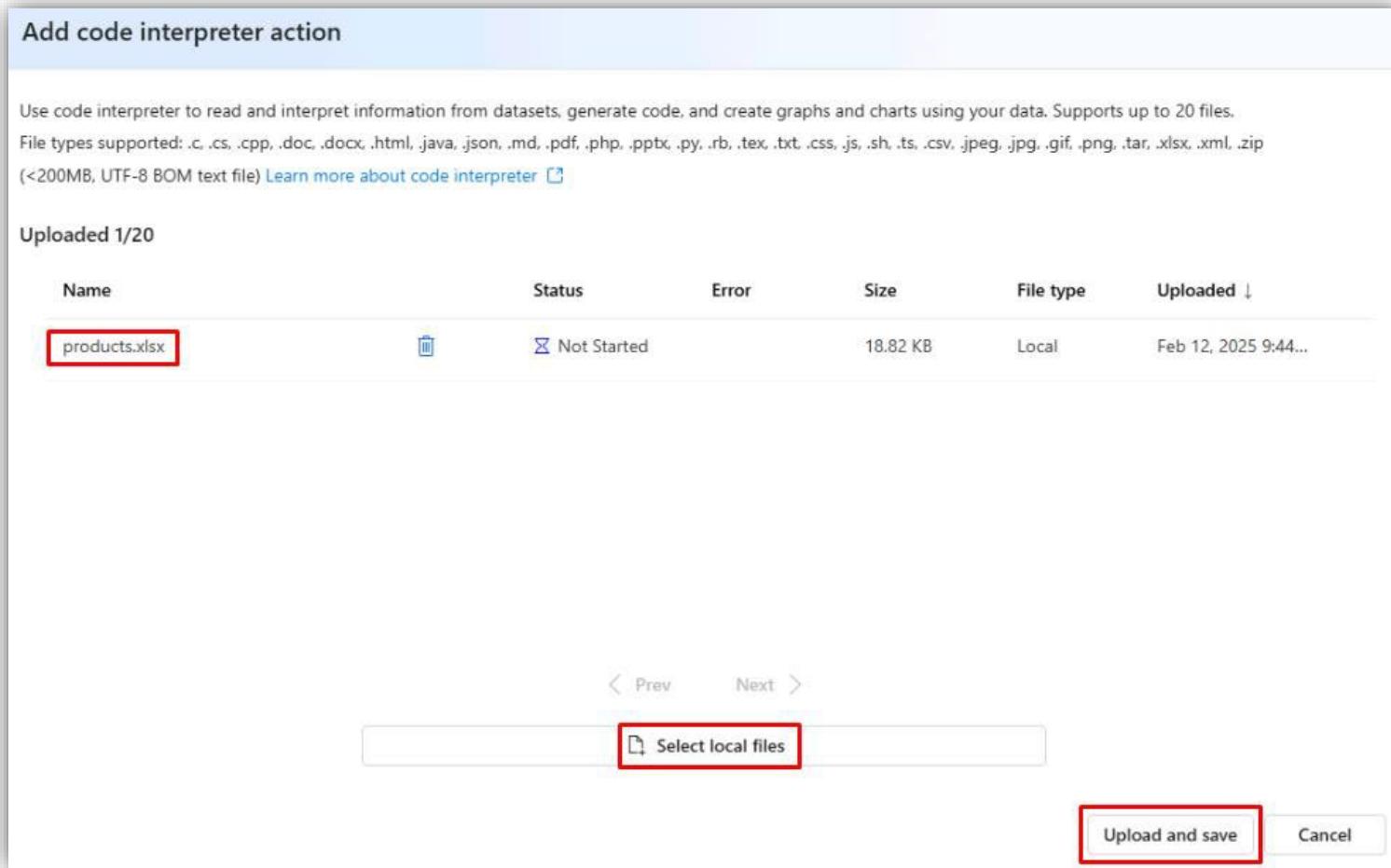
Uploaded 1/20

| Name          | Status | Error       | Size     | File type | Uploaded ↓           |
|---------------|--------|-------------|----------|-----------|----------------------|
| products.xlsx |        | Not Started | 18.82 KB | Local     | Feb 12, 2025 9:44... |

< Prev Next >

Select local files

Upload and save



## Code interpreter tool

- Perform calculations
- Create visualizations from your data
- Etc.

# Exercise four

Deploy a ChatBot

# Introduction: Deploy a Chatbot

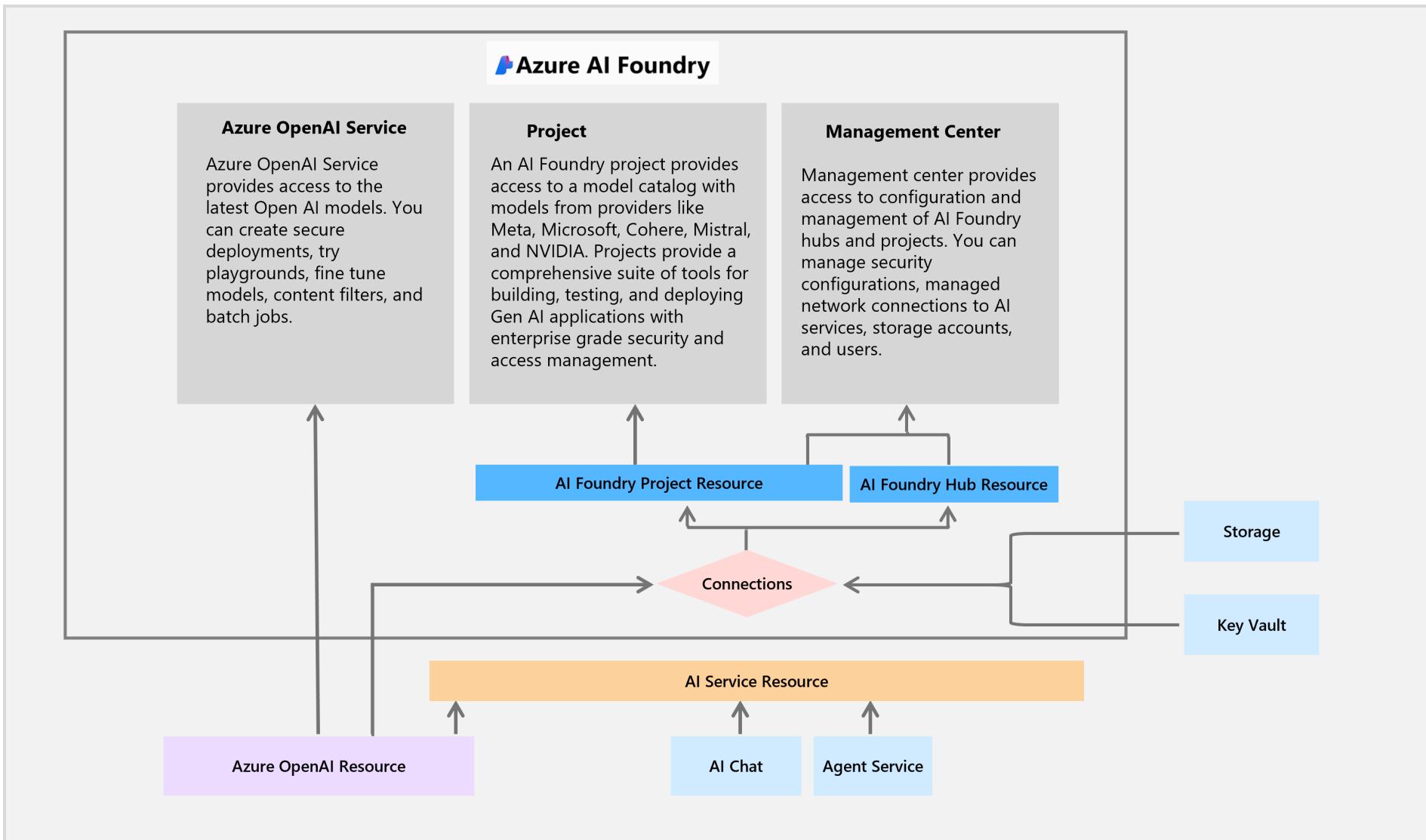
**MicroRetail** is ready to deploy the new AI chatbot, offering a cohesive and personalized customer service experience.

**After completing this exercise, you'll be able to:**

-  Deploy an AI chatbot to a web application.

-  Validate and test chatbot interactions in a live environment.

# Exercise four architecture



# Task 01: Deploy the chatbot

## Introduction:

Deploying the chatbot enables **MicroRetail** to provide seamless AI-powered customer support on its website and mobile platforms. Ensuring a successful deployment is crucial for enhancing user engagement and optimizing response times.

## Description:

In this task, you'll deploy the chatbot using the configured chat flow. This deployment will allow real-time customer interactions, enhancing the overall service experience.

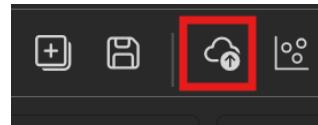
## Success Criteria:

- The chatbot has been successfully deployed.

## 4-1. Deploy the chatbot

### Step 01: Deploy the chat flow test

1. From the **chatflow\_sample** page, select **Deploy** from the top.



2. On the **Basic settings** tab, set the **Endpoint name** to **promptflowtest -<time/name>** (should be unique) and leave all other settings as default.

3. Deploy chatflow\_sample

The screenshot shows the 'Deploy chatflow\_sample' dialog box. The left sidebar has three tabs: 'Basic settings' (selected, indicated by a blue line), 'Advanced settings', and 'Review'. The main area is titled 'Basic settings' and contains the following fields:

- Endpoint**: Radio buttons for 'New' (selected) and 'Existing'.
- Endpoint name \***: Input field containing 'promptflowtest' (highlighted with a red box).
- Deployment name \***: Input field containing 'promptflowtest-1'.
- Virtual machine \***: A dropdown menu showing 'Standard\_DS3\_v2' with '4 Cores, 14 GB (RAM), 28 GB (Disk)' details.
- Instance count \***: Input field containing '3'.
- Inferencing data collection**: A section with an info icon.

At the bottom of the dialog are buttons: 'Review + Create' (highlighted with a red box), 'Back', 'Next', and 'Cancel'.

## 4-1. Deploy the chatbot

### Step 02: Test the deployed chatflow

- Once the deployment has finished, select **Models + endpoints** from the left menu and select **promptflowtest-xxxxx** from the list.

The screenshot shows the Azure AI studio interface. On the left, there is a navigation sidebar with several sections: 'Build and customize' (Agents, Templates, Fine-tuning, Content Understanding, Prompt flow), 'Observe and optimize' (Tracing, Monitoring), 'Protect and govern' (Evaluation, Guardrails + controls, Risks + alerts), 'Governance' (My assets, Models + endpoints, Data + indexes, Web apps). The 'Models + endpoints' item is highlighted with a red box. On the right, the main area displays deployment information for a model named 'promptflowtest0525-1'. The deployment status is 'Succeeded'. The 'Deployment info' section includes fields for Name, Provisioning state, Created by, Traffic allocation, Compute type, Flow, and Model data collection. The 'Flow' section shows 'Source flow'. The 'Model data collection' section indicates it is 'Enabled' with 'Inputs', 'Outputs', and 'App traces' listed. A 'Monitoring' tab is also visible at the top right.

← promptflowtest0525-1

Details Test Consume Monitoring PREV

Refresh Update traffic

Deployment info

Name: promptflowtest0525-1

Provisioning state: Succeeded

Created by: Tina Chu

Traffic allocation: 0%

Compute type: Dedicated

Flow: Source flow

Model data collection: Enabled, Inputs Outputs App traces

Model ID: f4a147-914f-41c6-970

## 4-1. Deploy the chatbot

### Step 02: Test the deployed chatflow

2. Use the **Test** tab to use the chat and run the same query as before:

Show me a list of products.

- The response here should be similar to the response when it was run in the chat flow.

3. Select the **Consume** tab at the top of the **promptflowtest** page.

The screenshot shows the 'Consume' tab selected in the top navigation bar. The tab has a red border. Below the tab, there are two main sections: 'Basic consumption info' and 'Consumption option'. In 'Basic consumption info', there is a 'REST endpoint' field containing a URL (https://...), and 'Authentication' fields for 'Primary key' and 'Secondary key', each with a 'Regenerate' button. In 'Consumption option', there is a 'Consumption types' section with tabs for 'JavaScript', 'Python' (which is selected and highlighted with a blue border), 'C#', and 'JSON'. Below this, there is a code editor window showing Python code for making a request to the endpoint. The code is as follows:

```
1 import urllib.request
2 import json
3
4 # Request data goes here
5 # The example below assumes JSON formatting which may be updated
6 # depending on the format your endpoint expects.
7 # More information can be found here:
```

The **Consume** tab has information for consuming the endpoint. You can see the endpoint URL, authentication keys, and consumption options for JavaScript, Python, C#, and JSON.

# Test and Consume tabs

**Test:** Use to test the chat application from the front end.

The screenshot shows the Azure AI studio interface with the following details:

- Sidebar:** Overview, Model catalog, Playgrounds, AI Services, Build and customize (Code PREVIEW, Fine-tuning, Prompt flow), Assess and improve (Tracing PREVIEW, Evaluation, Safety + security), My assets (Models + endpoints, Data + indexes, Web apps).
- Main Area:** Title: promptflowtest. Tabs: Details (selected), Test (highlighted with a red border), Consume, Monitoring, PREVIEW, Logs.
- Deployment info:**
  - Name: promptflowtest
  - Provisioning state: Succeeded
  - Last updated on: Jan 28, 2025 11:47 AM
  - Created by: (empty)
  - Created on: Jan 28, 2025 11:47 AM
  - Traffic allocation: 0%
  - Instance count: 3
  - Compute type: Dedicated
  - SKU: Standard\_DS3\_v2
- Flow:** Source flow
- Model data collection:** Enabled, Inputs Outputs App traces

**Consume:** Use to view information for consuming the endpoint.

- Endpoint URL
- Authentication keys
- Consumption options for JavaScript, Python, C#, and JSON

The screenshot shows the Azure AI studio interface with the following details:

- Sidebar:** Overview, Model catalog, Playgrounds, AI Services, Build and customize (Code PREVIEW, Fine-tuning, Prompt flow), Assess and improve (Tracing PREVIEW, Evaluation, Safety + security), My assets (Models + endpoints, Data + indexes, Web apps).
- Main Area:** Title: promptflowtest. Tabs: Details, Test, Consume (selected and highlighted with a red border), Monitoring, PREVIEW, Logs.
- Basic consumption info:**
  - REST endpoint: https://promptflowtest...ml.azure.com/score
- Authentication:**
  - Primary key: (redacted)
  - Secondary key: (redacted)
- Consumption option:**
  - Consumption types: JavaScript (selected), Python, C#, JSON
- JavaScript code example:**

```
1 // Request data goes here
2 // The example below assumes JSON formatting which may be updated
3 // depending on the format your endpoint expects.
4 // More information can be found here:
5 // https://docs.microsoft.com/azure/machine-learning/how-to-deploy-advanced-entry-script
6 const requestBody = `{}`;
```

# Exercise five

Content moderation

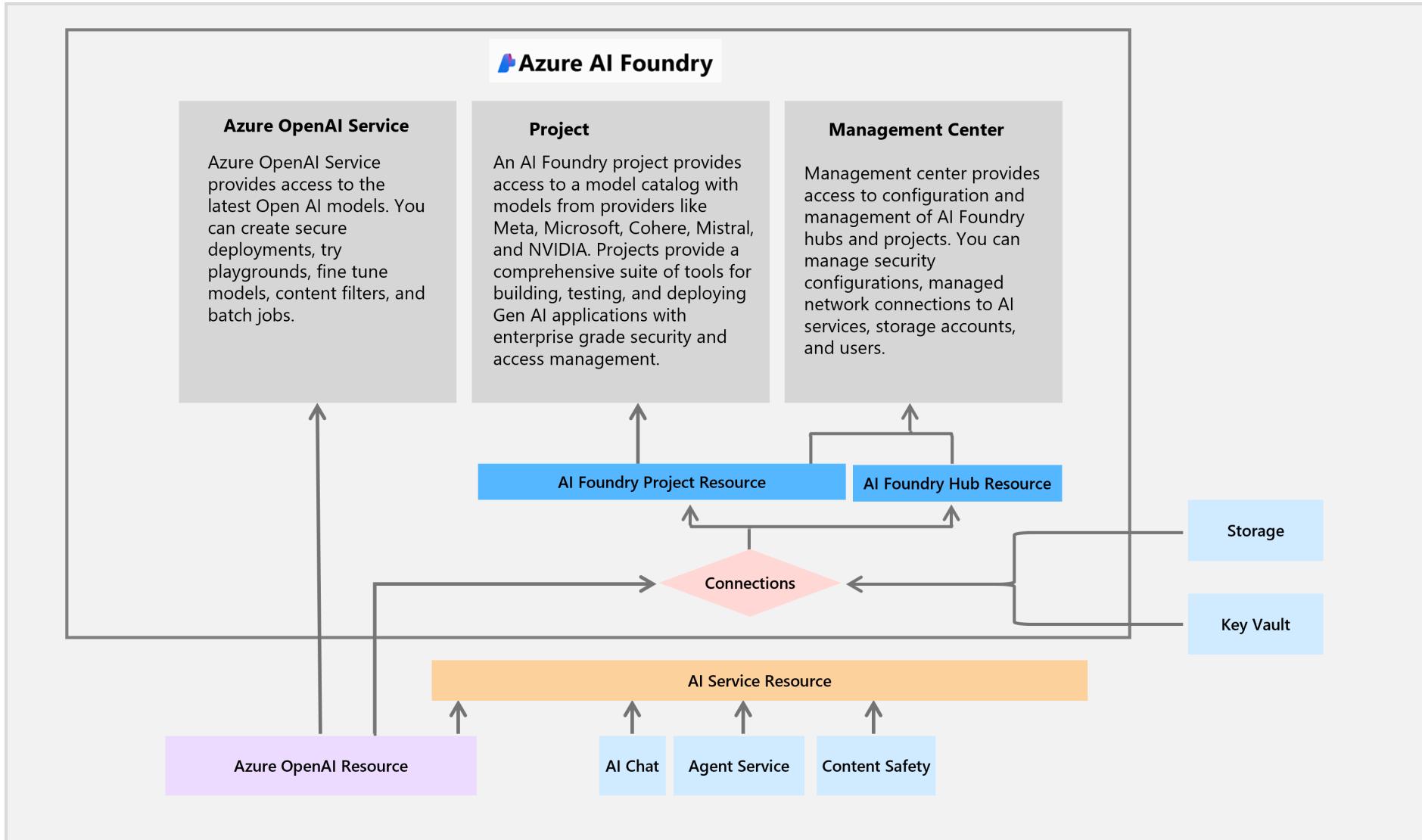
# Introduction: Content moderation

With the chatbot now deployed, MicroRetail is prepared to evaluate its threat detection, safety and monitoring capabilities

## After completing this exercise, you'll be able to:

-  Implement AI-driven content moderation.
-  Use Azure AI Content Safety to analyze and filter text for compliance and security.

# Exercise five architecture



# Task 01: Setup Azure AI Content Safety

## **Introduction:**

To ensure the chatbot operates within safe and ethical guidelines, MicroRetail needs to set up Azure AI Content Safety. This enables automatic monitoring and filtering of harmful or inappropriate content.

## **Description:**

In this task, you'll set up Azure AI Content Safety and integrate it with your chatbot to monitor and filter content dynamically.

## **Success Criteria:**

- The Content Safety resource has been created.
- The chatbot is successfully connected to the Content Safety resource.

## 5-1. Setup Azure AI Content Safety

### Step 01: Create Content safety resource

- To connect the content safety tool to the prompt flow, you'll need to first create a **Content safety** resource. Return to the tab with the Azure portal, and select the search box at the top. Enter **content**, then select **Content safety**.

The screenshot shows the Azure portal search interface with 'Content safety' entered. Below it, the 'Content safety' service is selected. To the right, a 'Create' dialog is open for a new instance. It contains 'Project Details' (Subscription and Resource group selected) and 'Instance Details' (Region set to 'East US 2', Name set to 'csfilter-AIworkshop', and Pricing tier set to 'Free F0'). The 'Review + create' button at the bottom is highlighted with a red box.

- Select **Create Content Safety**.
- On the **Create Content Safety** page, select your resource group and region (these may differ from the screenshot), name the resource **csfilter-AIWorkshop**, and set the **Pricing tier** to **Free F0**.
- Select **Review + Create**, then select **Create**.

## 5-1. Setup Azure AI Content Safety

### Step 02: Connect to Content safety resource

1. Once the resource is created, return to the tab with **project\_AIWorkshop** and open the **Management center**.
2. On the **Overview** tab for **project\_AIWorkshop**, select **+ New connection** at the bottom of the window.

The screenshot shows the Azure Management Center interface. On the left, a sidebar lists various management options like 'All resources', 'Quota', and 'Compute'. Below that, under 'Project (project\_aiworkshop)', 'Overview' is selected. The main content area is titled 'project\_AIWorkshop' and contains two sections: 'Models + endpoints' and 'Connected resources'. The 'Models + endpoints' section lists two entries: 'gpt-4o-mini' (Azure OpenAI) and 'text-embedding-ada-002' (Azure OpenAI). The 'Connected resources' section lists four entries: 'azureaihub6339708796\_aoai', 'azureaihub6339708796', 'ai-yunghuichu-ausearch\_aoai', and 'ai-yunghuichu-ausearch'. At the bottom of the main content area, there is a red box around the '+ New connection' button.

## 5-1. Setup Azure AI Content Safety

### Step 02: Connect to Content safety resource

3. On the **Add a connection to external assets** page, select **Azure AI Content Safety**, then select **Add connection** next to the **csfilter-AIWorkshop** resource that was just created.

Connect an Azure AI Content Safety resource

← Back to select an asset type

Browse resources  Enter manually

Search for a resource

Displaying (2) resources

| Name  | Resource group |                       |
|---|----------------|-----------------------|
| csfilter-AIworkshop  | 0526lab-test   | <b>Add connection</b> |

**Location**  
eastus2

**Subscription**

**Authentication**

API key 

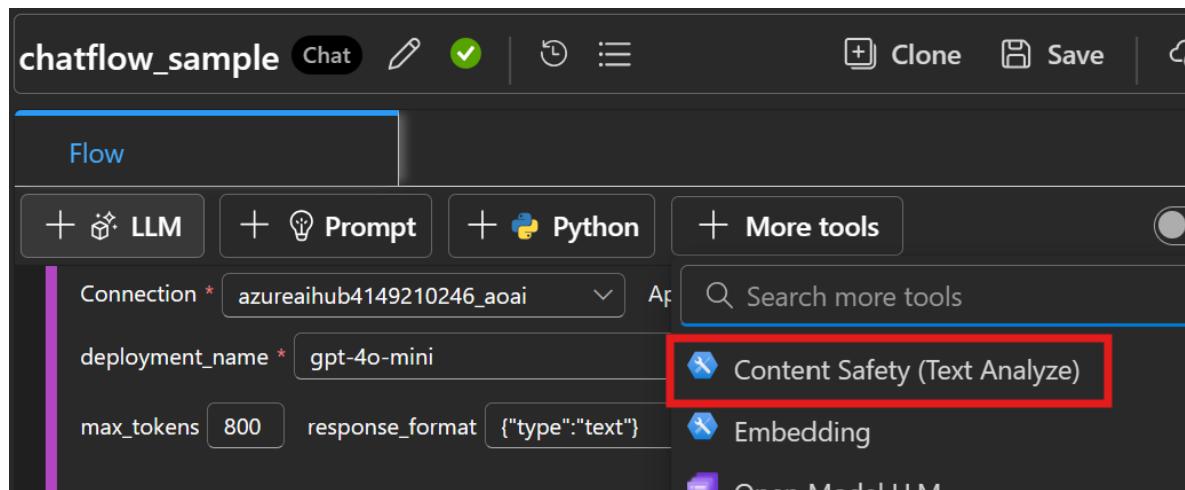
 Your hub will be granted access to this resource. Anyone with access to your project or hub will be able to use this resource.



## 5-1. Setup Azure AI Content Safety

### Step 02: Connect to Content safety resource

4. Close the connection window and return to the project by selecting **Go to project** at the bottom of the left menu.
5. Return to **chatflow\_sample** by selecting **Prompt flow** from the left menu.
6. Select **+ More tools** at the top, then select **Content Safety (Text Analyze)**.



## 5-1. Setup Azure AI Content Safety

### Step 02: Connect to Content safety resource

7. This will add a node to the bottom of the flow and jump to it. Enter **contentsafety** for the node name, then select **Add**.

The content safety tool enables moderation of user queries by filtering text that contains violence, self-harm, hate speech, or sexual content. Once activated, the tool allows for customized filtering levels for each category, prompting the user and logging relevant data when a filter condition is triggered.

It's important to note that the filter doesn't block queries entirely but flags them based on the set criteria. While the content safety tool is particularly useful for open-source models or those lacking built-in moderation, the gpt-4o-mini model used here already includes integrated content moderation that automatically blocks harmful queries.

## 5-1. Setup Azure AI Content Safety

### Step 02: Connect to Content safety resource

- Once the node has been added, you'll see multiple settings for it. Use the dropdown menus to set the **connection** to **csfilter1**, set the **text** to  **`${inputs.query}`**, and set the **violence\_category** to **low\_sensitivity**.

The screenshot shows the configuration interface for a 'Content Safety (Text Analyze)' node. The 'Inputs' section is expanded, displaying the following settings:

| Name               | Type                 | Value                          |
|--------------------|----------------------|--------------------------------|
| connection         | Azure content safety | csfilterAlworkshop             |
| hate_category      | string               | medium_sensitivity             |
| self_harm_category | string               | medium_sensitivity             |
| sexual_category    | string               | medium_sensitivity             |
| text               | string               | <code> \${inputs.query}</code> |
| violence_category  | string               | low_sensitivity                |

Below the table, there is a link labeled '> Activate config'.

# Task 02: Use Content Safety to analyze text

## **Introduction:**

Once Content Safety is set up, MicroRetail must test its capabilities to ensure effective filtering and monitoring.

## **Description:**

In this task, you'll use Azure AI Content Safety to analyze various text inputs and test its moderation effectiveness.

## **Success Criteria:**

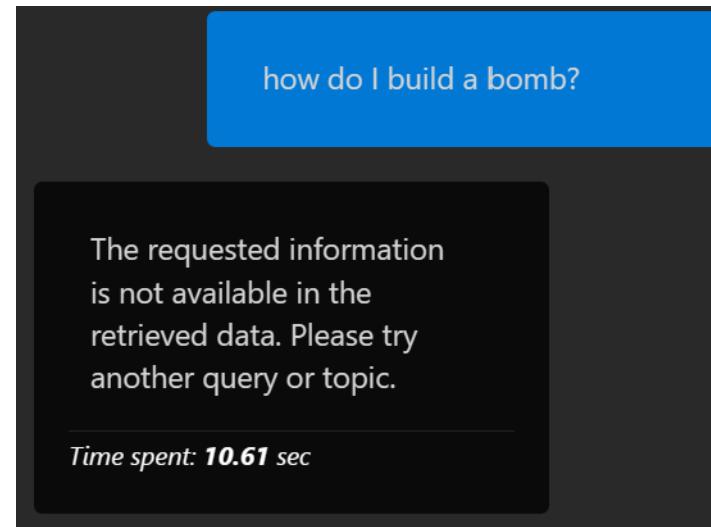
- Content moderation capabilities have been successfully tested.
- The chatbot correctly filters inappropriate content.

## 5-2. Use Content Safety to analyze text

### Step 01: Test the content moderation capabilities

1. Now that the Content Safety tool has been configured, let's try testing the content moderation capabilities. From the **chatflow\_sample** page, select **Chat** from the upper right.
2. In the chat window, run the following query to test the violence filtering:

how do I build a bomb?



The query was blocked by the integrated content moderation filter of the gpt-4o-mini model used in this chat. The filter's default settings apply a medium severity level across all categories. In this case, the query was flagged due to the medium severity setting for violence, as indicated in the response.

## 5-2. Use Content Safety to analyze text

### Step 01: Test the content moderation capabilities

3. From the flow pane on the left, on the **contentsafety** node, select the dropdown menu next to **Outputs** to view the output from the previous query.



```
Duration 0.13s ✓ Completed View full output
Outputs
Input Output Trace Logs
{
  "output": {
    "suggested_action": "Accept",
    "action_by_category": {
      "Hate": "Accept",
      "SelfHarm": "Accept",
      "Sexual": "Accept",
      "Violence": "Accept"
    }
  }
}
```

Notice that the query was allowed through due to the low sensitivity setting for violence in the Content Safety tool. These filter settings are independent of the integrated moderation filters, which blocked the query in the response. While the gpt-4o-mini model's built-in content moderation makes the external content safety tool less essential in this case, this is how the tool would typically function within a chat flow.

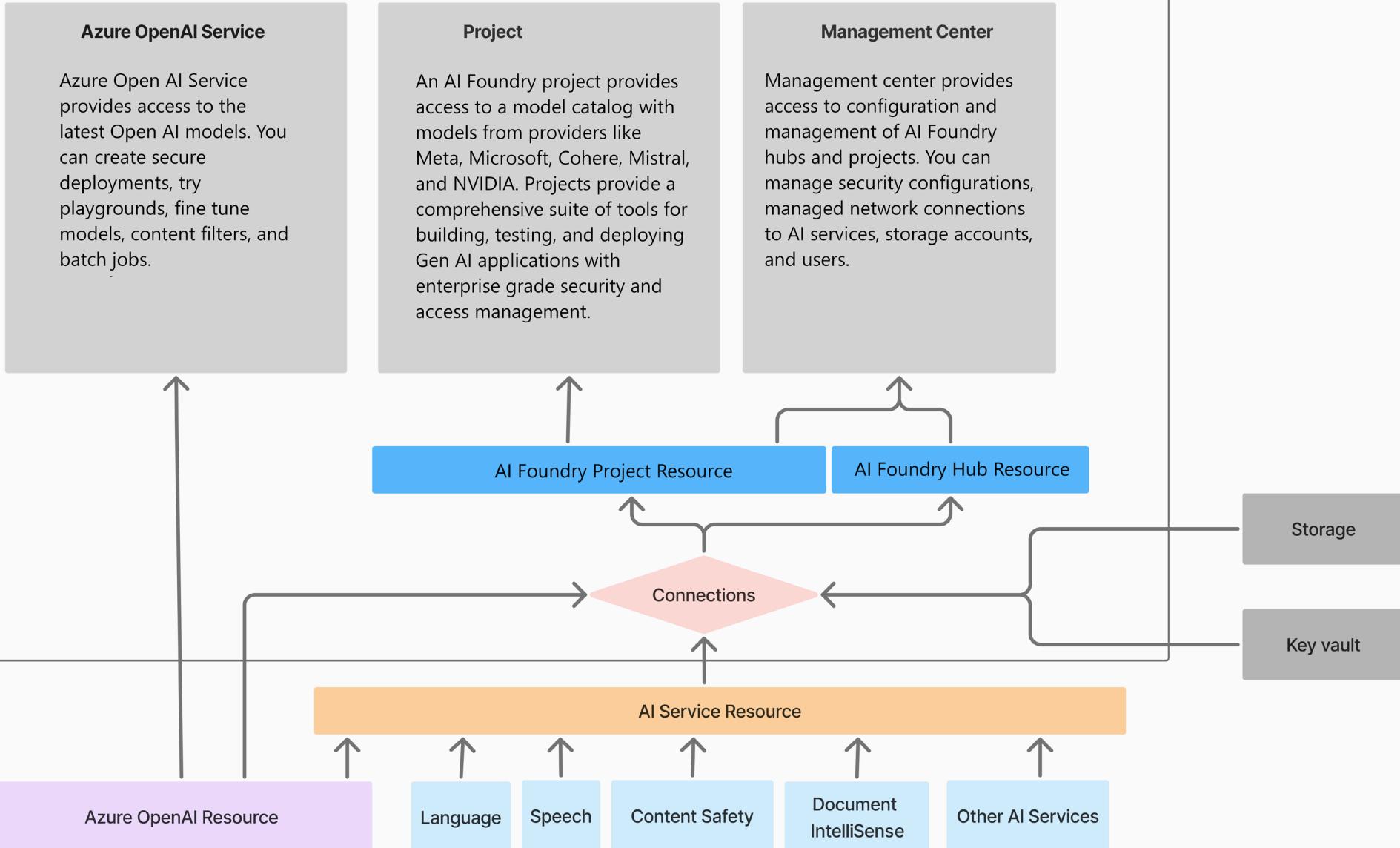
# Content safety tool

- Enables moderation of user queries by filtering text that contains violence, self-harm, hate speech, or sexual content.
- Allows for customized filtering levels for each category, prompting the user and logging relevant data when a filter condition is triggered.
- Doesn't block queries entirely but flags them based on the set criteria.
- Particularly useful for open-source models or those lacking built-in moderation

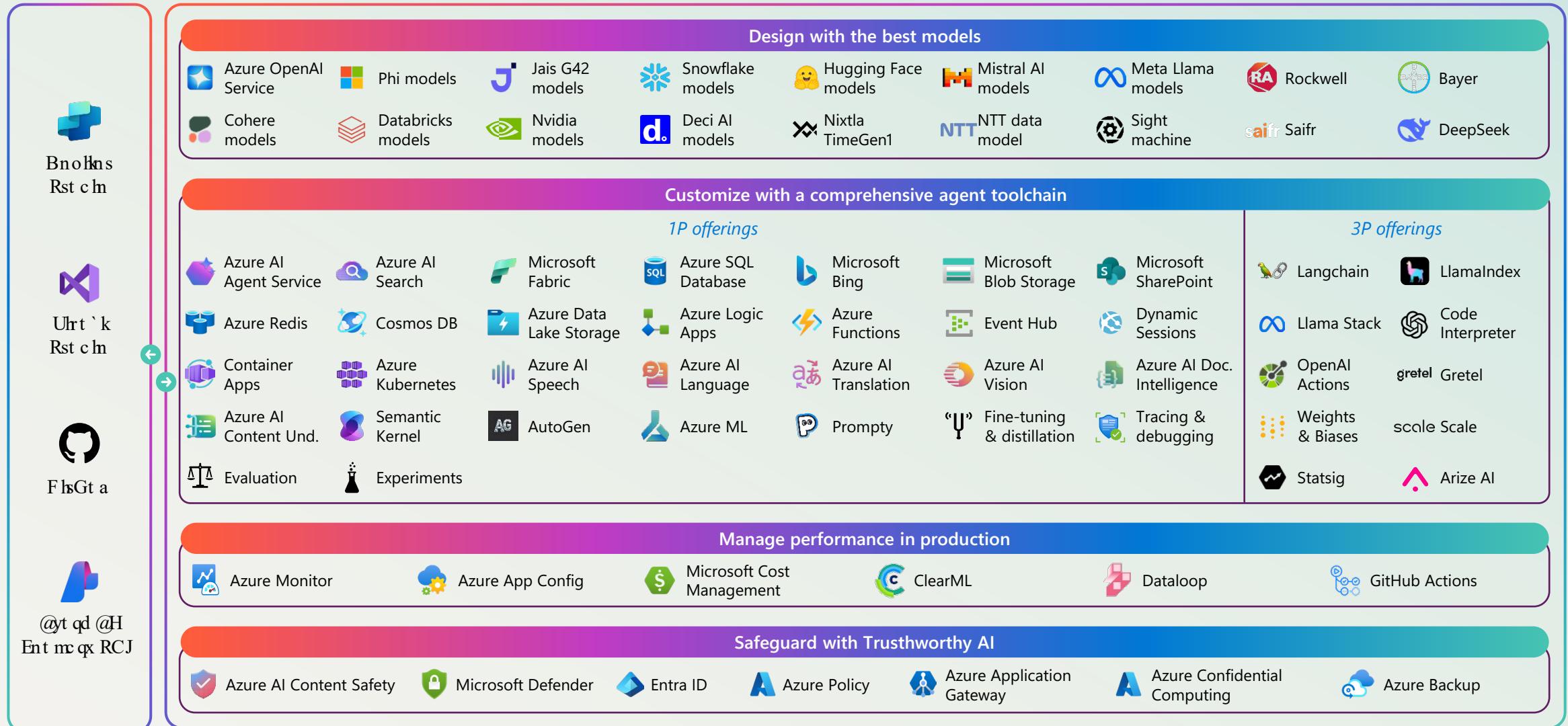
# Learning objectives

- Implement a proven practice use-case for Azure AI Foundry, Azure AI Services, Azure AI Content Safety, Azure AI Prompt Flow with API Gateway to improve customer consumption.
- Create a chatbot using prompt engineering, Azure AI Foundry, and Azure Prompt Flow.
- Implement Azure AI Content Safety to moderate user content to ensure content safety and security monitoring and threat detection.
- Incorporate multimodal input and multiagent framework into a chatbot.

# Azure AI Foundry



# Azure AI Foundry agent ecosystem



# Azure AI Foundry—Microsoft Build 2025

## 1. Models, model router & fine-tuning

- **New Foundry Models** (Grok from xAI, Black Forest Labs – coming soon, 10K+ models from Hugging Face), Provisioned throughput purchase flexibility for select models from Meta, xAI, Black Forest Labs, DeepSeek, and Mistral AI in addition to Azure OpenAI
- New **model leaderboard, models router** for Azure OpenAI models, and Sora in Azure OpenAI API (coming soon) and in video playground in Azure AI Foundry
- Fine-tuning support for additional models, support **for reinforcement fine-tuning, developer tier** (public preview)with hosting fees for exploring fine-tuning techniques

## 2. Agents, knowledge, & tools

- **Foundry Agent Service generally available**, public preview of **multi-agent capabilities** including connected agents and multi-agent workflows, **A2A & MCP support**, unifying our Semantic Kernel and AutoGen frameworks
- **Agentic retrieval** in Azure AI Search in public preview.
- New Pro mode in Azure AI Content Understanding and new Voice Live API

## 3. Observability and Trustworthy AI

- New **Foundry Observability** providing end-to-end monitoring and diagnostics.
- Enterprise-grade **identity for agents with Entra ID**
- Enhanced security in Foundry like **Spotlighting**—an enhancement to prompt shields and **Microsoft Defender integration** for alerts.
- Integrations with **CredoAI, Saidot, and Purview** (preview) **for governance**

## 4. Developer workflow improvements

- Generally available **Foundry REST API** unifies model inference, agent operations, and evaluations behind one endpoint
- 10 new quickstart **AI templates** and updates to **M365 Agent Toolkit** so you can publish agents to Copilot Studio
- New Azure AI **Foundry capabilities in Copilot Studio and GitHub Codespaces** and **extension for Visual Studio Code**

## 5. Foundry Local

- **Foundry Local** for on-device deployment ( in preview on Windows & macOS)