Azure AI Foundry

Workshop 1

Table des matières

Lab 01: Introduction to LLMs and Azure AI Foundry 2

- Prerequisites 2
- Create a new Hub 3
  - Definition 3
- Create a new Project 7
  - Definition 7
- Deploy a model 9
  - Serverless API deployment 10
  - Test the model (Serverless API) 16
  - Managed Compute deployment 19
  - Test the model (Managed Compute) 23
- Discover Content Safety 26
  - Responsible AI Principles 26
  - Content Filtering System 26
  - Mitigation Layers 27
  - Application – Content Safety 28

Lab 01: Introduction to LLMs and Azure AI Foundry

In this lab, we will have an overview on how to use Azure AI to work with large language models.

The focus will be more on an overview of the creation process, so that in the next lessons we will delve deeper into the build, evaluation, deployment, and monitoring process.

Prerequisites

An Azure subscription is required, where you can create an AI Project along with its AI Hub Resource, a Content Safety service, and an AI Search service.

Setup

- Create an AI Project and AI Hub Resources

Lab Steps

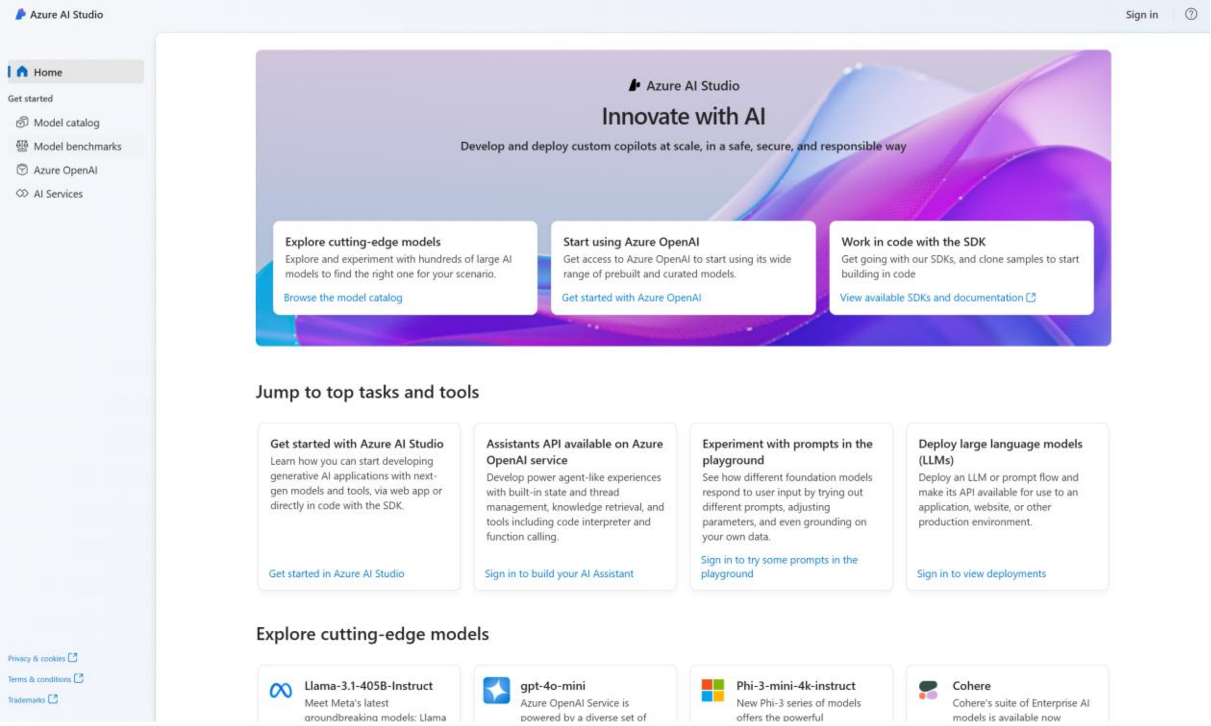
1. Use Azure AI Foundry Playground.
2. Work with an Open Source LLM Model.
3. Test the prompt in Content Safety.
4. Create a Prompt Flow flow.

Setup

Create an AI Project and AI Hub Resouces

Let's start by creating a project in Azure AI Foundry.

Go to your browser and type: https://ai.azure.com



Azure AI Studio

Innovate with AI

Develop and deploy custom copilots at scale, in a safe, secure, and responsible way

Explore cutting-edge models
Explore and experiment with hundreds of large AI models to find the right one for your scenario.
Browse the model catalog

Start using Azure OpenAI
Get access to Azure OpenAI to start using its wide range of prebuilt and curated models.
Get started with Azure OpenAI

Work in code with the SDK
Get going with our SDKs, and clone samples to start building in code
View available SDKs and documentation

Jump to top tasks and tools

Get started with Azure AI Studio
Learn how you can start developing generative AI applications with next-gen models and tools, via web app or directly in code with the SDK.
Get started in Azure AI Studio

Assistants API available on Azure OpenAI service
Develop power agent-like experiences with built-in state and thread management, knowledge retrieval, and tools including code interpreter and function calling.
Sign in to build your AI Assistant

Experiment with prompts in the playground
See how different foundation models respond to user input by trying out different prompts, adjusting parameters, and even grounding on your own data.
Sign in to try some prompts in the playground

Deploy large language models (LLMs)
Deploy an LLM or prompt flow and make its API available for use to an application, website, or other production environment.
Sign in to view deployments

Explore cutting-edge models

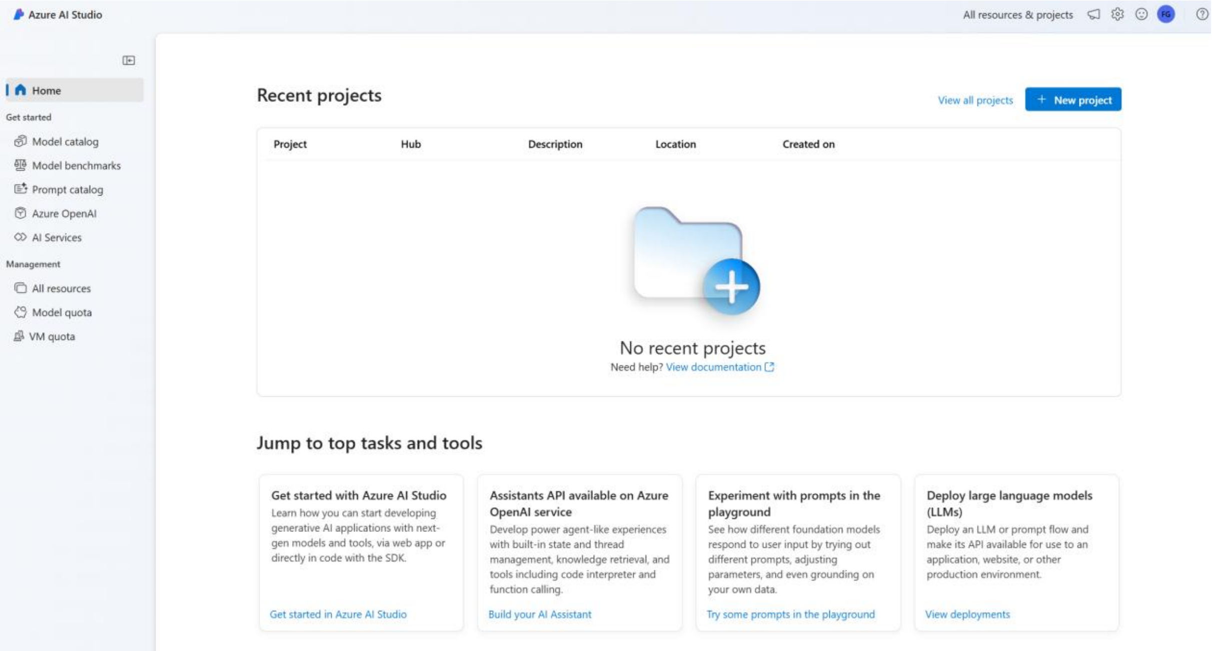
Llama-3.1-405B-Instruct
Meet Meta's latest groundbreaking models: Llama

gpt-4o-mini
Azure OpenAI Service is powered by a diverse set of

Phi-3-mini-4k-instruct
New Phi-3 series of models offers the powerful

Cohere
Cohere's suite of Enterprise AI models is available now

After logging in with your Azure account, you will see the following screen:



Recent projects

Project	Hub	Description	Location	Created on

No recent projects

Need help? View documentation

Jump to top tasks and tools

Get started with Azure AI Studio
Learn how you can start developing generative AI applications with next-gen models and tools, via web app or directly in code with the SDK.
Get started in Azure AI Studio

Assistants API available on Azure OpenAI service
Develop power agent-like experiences with built-in state and thread management, knowledge retrieval, and tools including code interpreter and function calling.
Build your AI Assistant

Experiment with prompts in the playground
See how different foundation models respond to user input by trying out different prompts, adjusting parameters, and even grounding on your own data.
Try some prompts in the playground

Deploy large language models (LLMs)
Deploy an LLM or prompt flow and make its API available for use to an application, website, or other production environment.
View deployments

Create a new Hub

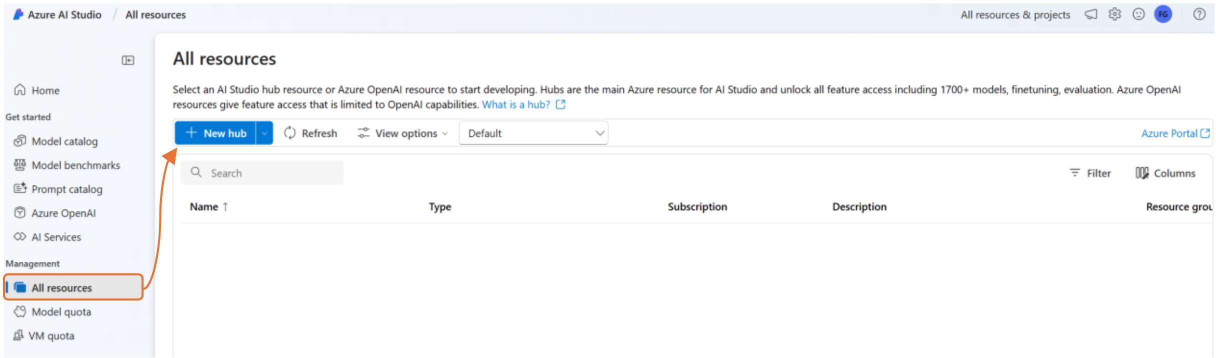
Definition

Hubs are the primary top-level Azure resource for AI Studio and provide a central way for a team to govern security, connectivity, and computing resources across playgrounds and projects.

Once a hub is created, developers can create projects from it and access shared company resources without needing an IT administrator's repeated help.

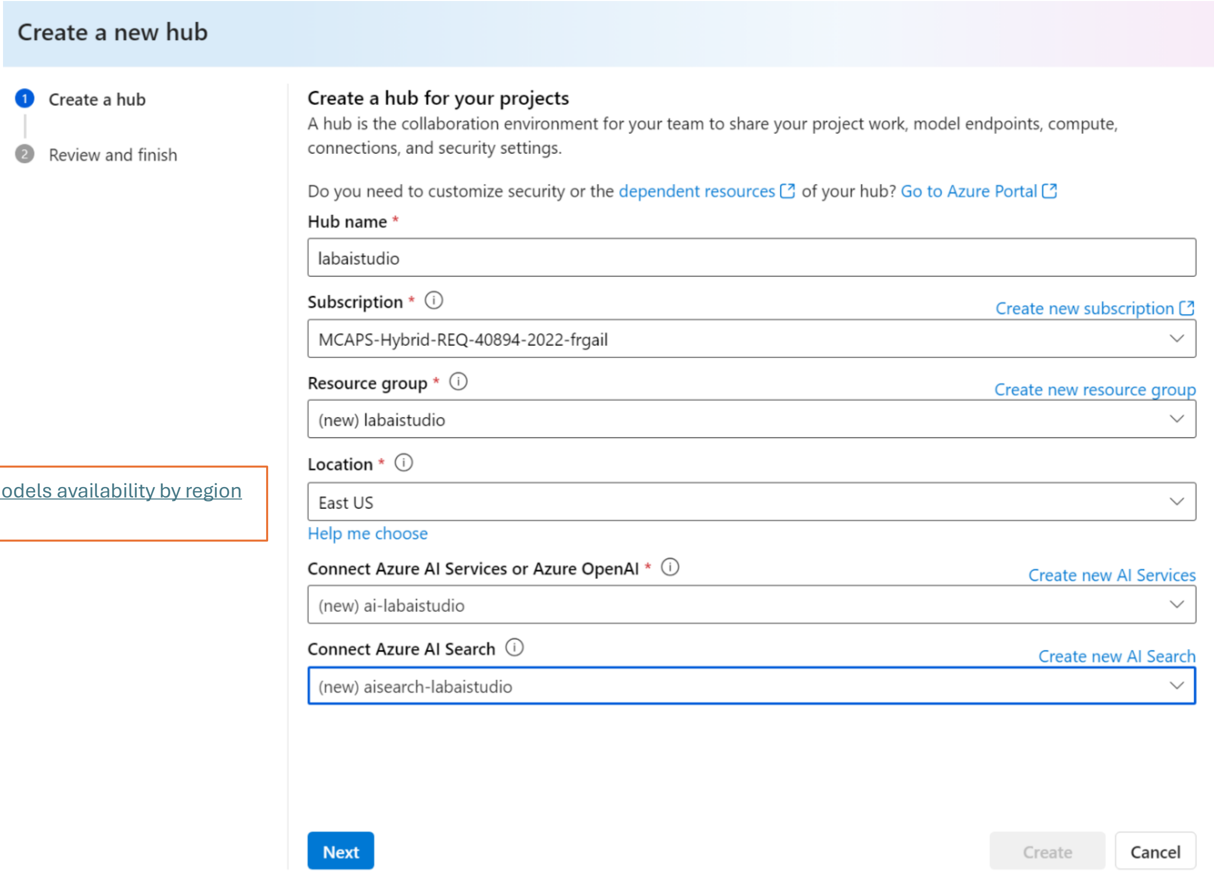
You can create and manage a hub from the Azure portal or from the AI Studio.

⚠ If you want to **create a secure Hub**, you must create the Hub from the portal: [How to create and manage an Azure AI Studio hub - Azure AI Studio | Microsoft Learn]



All resources

Select an AI Studio hub resource or Azure OpenAI resource to start developing. Hubs are the main Azure resource for AI Studio and unlock all feature access including 1700+ models, finetuning, evaluation. Azure OpenAI resources give feature access that is limited to OpenAI capabilities. What is a hub?



Create a new hub

Create a hub for your projects

A hub is the collaboration environment for your team to share your project work, model endpoints, compute, connections, and security settings.

Do you need to customize security or the dependent resources of your hub? Go to Azure Portal

Hub name * labaistudio

Subscription * MCAPS-Hybrid-REQ-40894-2022-frgail

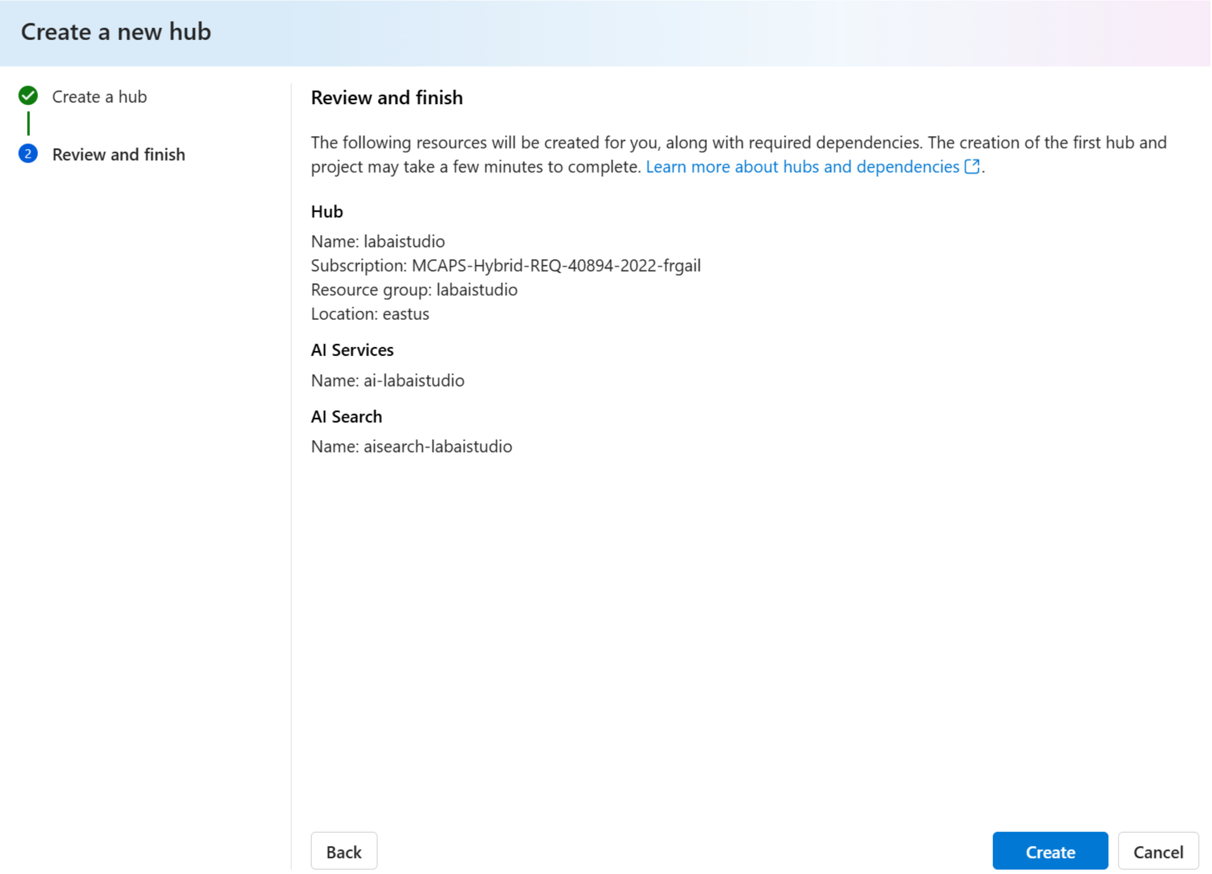
Resource group * (new) labaistudio

Location * East US

Connect Azure AI Services or Azure OpenAI * (new) ai-labaistudio

Connect Azure AI Search (new) aisearch-labaistudio

Models availability by region



Create a new hub

Review and finish

The following resources will be created for you, along with required dependencies. The creation of the first hub and project may take a few minutes to complete. Learn more about hubs and dependencies.

Hub

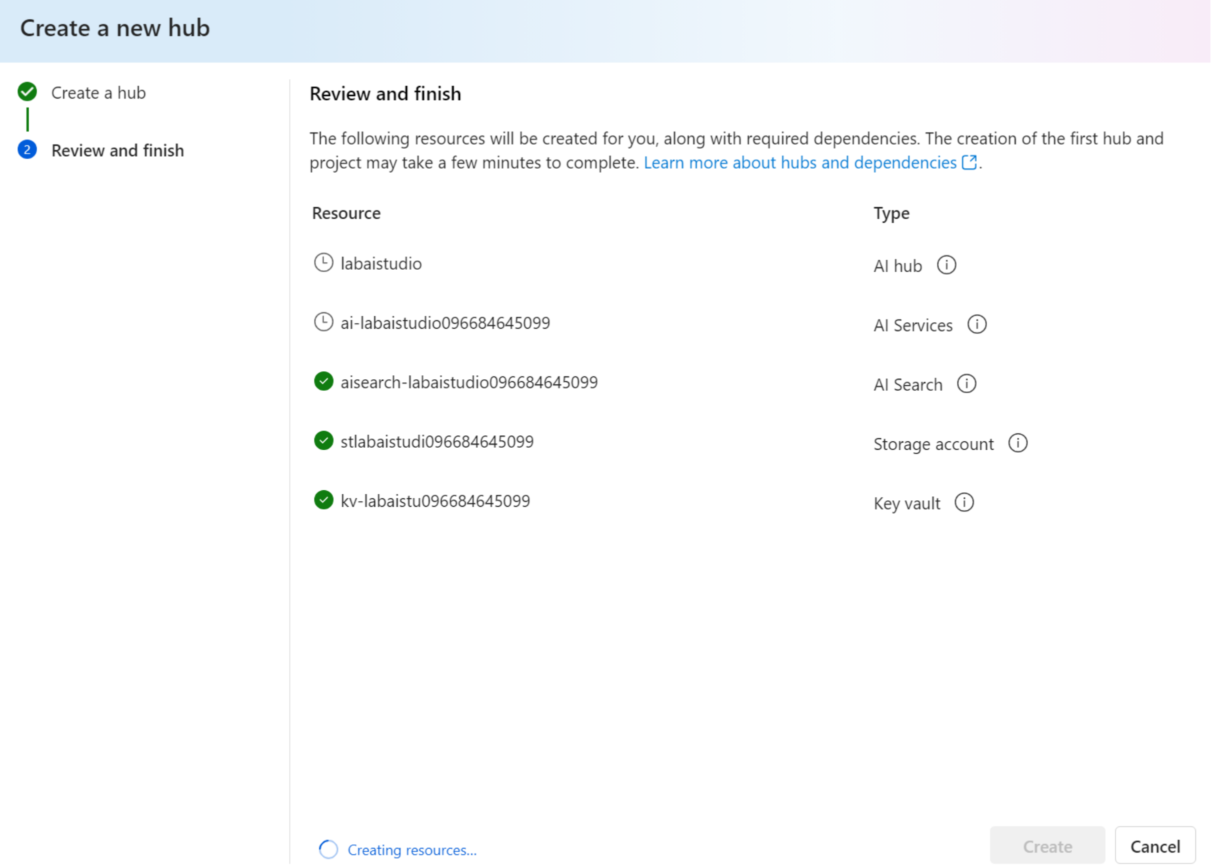
Name: labaistudio
Subscription: MCAPS-Hybrid-REQ-40894-2022-frgail
Resource group: labaistudio
Location: eastus

AI Services

Name: ai-labaistudio

AI Search

Name: aisearch-labaistudio



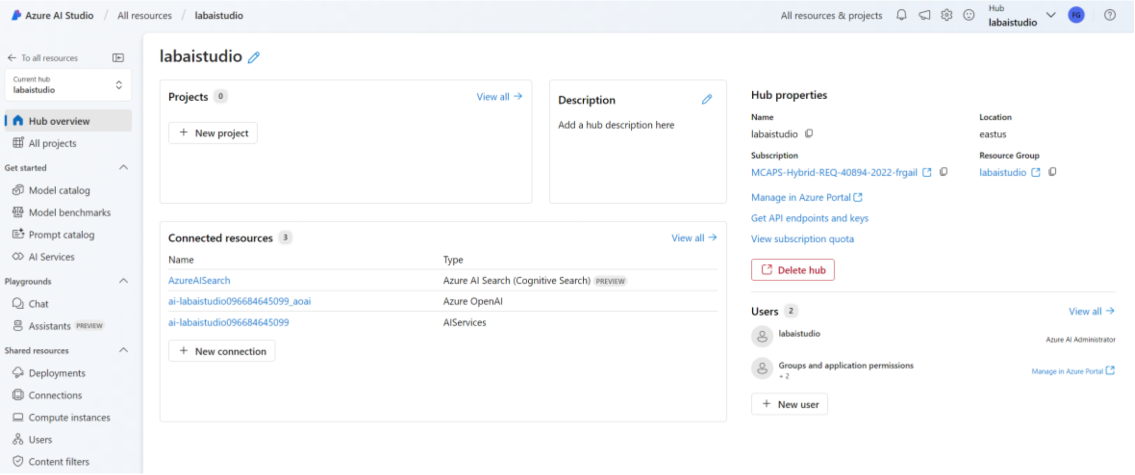
Create a new hub

Review and finish

The following resources will be created for you, along with required dependencies. The creation of the first hub and project may take a few minutes to complete. Learn more about hubs and dependencies.

Resource	Type
🕓 labaistudio	AI hub ⓘ
🕓 ai-labaistudio096684645099	AI Services ⓘ
✓ aisearch-labaistudio096684645099	AI Search ⓘ
✓ stlabaistudi096684645099	Storage account ⓘ
✓ kv-labaistu096684645099	Key vault ⓘ

Creating resources...



labaistudio

Projects 0

Description

Add a hub description here

Connected resources 3

Name	Type
AzureAISearch	Azure AI Search (Cognitive Search) PREVIEW
ai-labaistudio096684645099_aoai	Azure OpenAI
ai-labaistudio096684645099	AIServices

Hub properties

Name
labaistudio

Location
eastus

Subscription
MCAPS-Hybrid-REQ-40894-2022-frgail

Resource Group
labaistudio

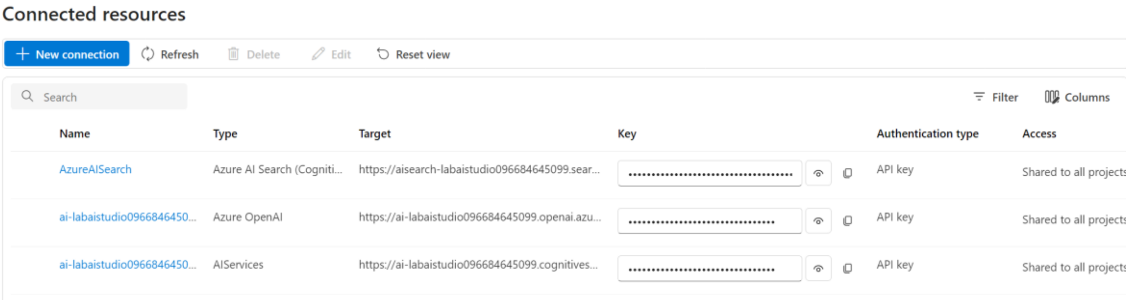
Users 2

labaistudio — Azure AI Administrator

Groups and application permissions + 2 — Manage in Azure Portal

<u>3 connected resources have been created</u>:

- Azure AI Search
- AI Services to access AI Services (Speech, Language+Translator, Vision, Content Safety)
- Azure OpenAI service to access Azure OpenAI models



Connected resources

Name	Type	Target	Key	Authentication type	Access
AzureAISearch	Azure AI Search (Cogniti...	https://aisearch-labaistudio096684645099.sear...	••••••••••••••••••••••••••••••••••••	API key	Shared to all projects
ai-labaistudio0966846450...	Azure OpenAI	https://ai-labaistudio096684645099.openai.azu...	••••••••••••••••••••••••••••••••	API key	Shared to all projects
ai-labaistudio0966846450...	AIServices	https://ai-labaistudio096684645099.cognitives...	••••••••••••••••••••••••••••••••	API key	Shared to all projects

<u>In summary, the Hub's key points</u>:

1. **Centralized Management**: A hub provides a unified environment where teams can manage security, connectivity, and computing resources across various AI projects.
2. **Collaboration**: It allows multiple developers and data scientists to collaborate on machine learning projects, sharing resources and configurations easily.
3. **Resource Sharing**: Hubs enable the sharing of Azure resources like storage accounts, model endpoints, and more, without needing repeated IT intervention.
4. **Security and Compliance**: You can set up and enforce security policies, network configurations, and compliance requirements at the hub level, which are then inherited by all projects under the hub.
5. **Project Organization**: Projects created within a hub can be customized and isolated, allowing for organized workspaces that help in managing data, access, and billing.
6. **Ease of Use**: Hubs simplify the process of setting up environments for AI development, making it easier to prototype, build, and deploy AI applications

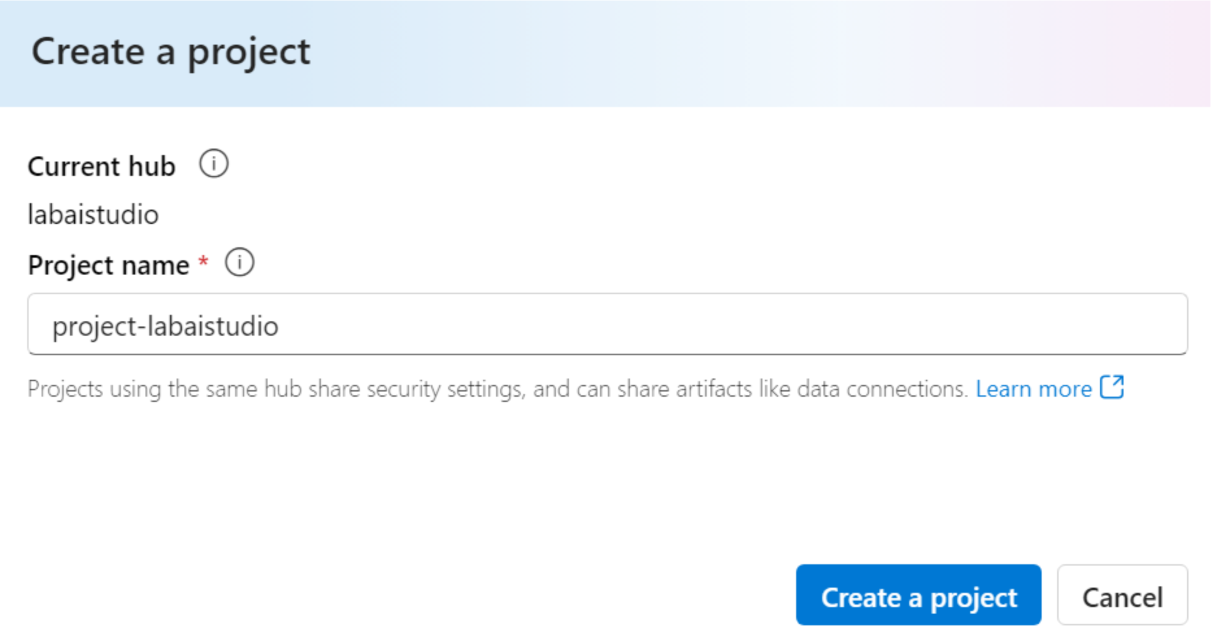
Create a new Project

A hub provides the hosting environment for projects in AI Studio.

Definition

A **project** is an organizational container that has tools for AI customization and orchestration. It lets you organize your work, save state across different tools like prompt flow, and collaborate with others. For example, you can share uploaded files and connections to data sources.

Multiple projects can use a hub, and multiple users can use a project. A project also helps you keep track of billing and manage access and provides data isolation. Every project uses dedicated storage containers to let you upload files and share it with only other project members when using the 'data' experiences.

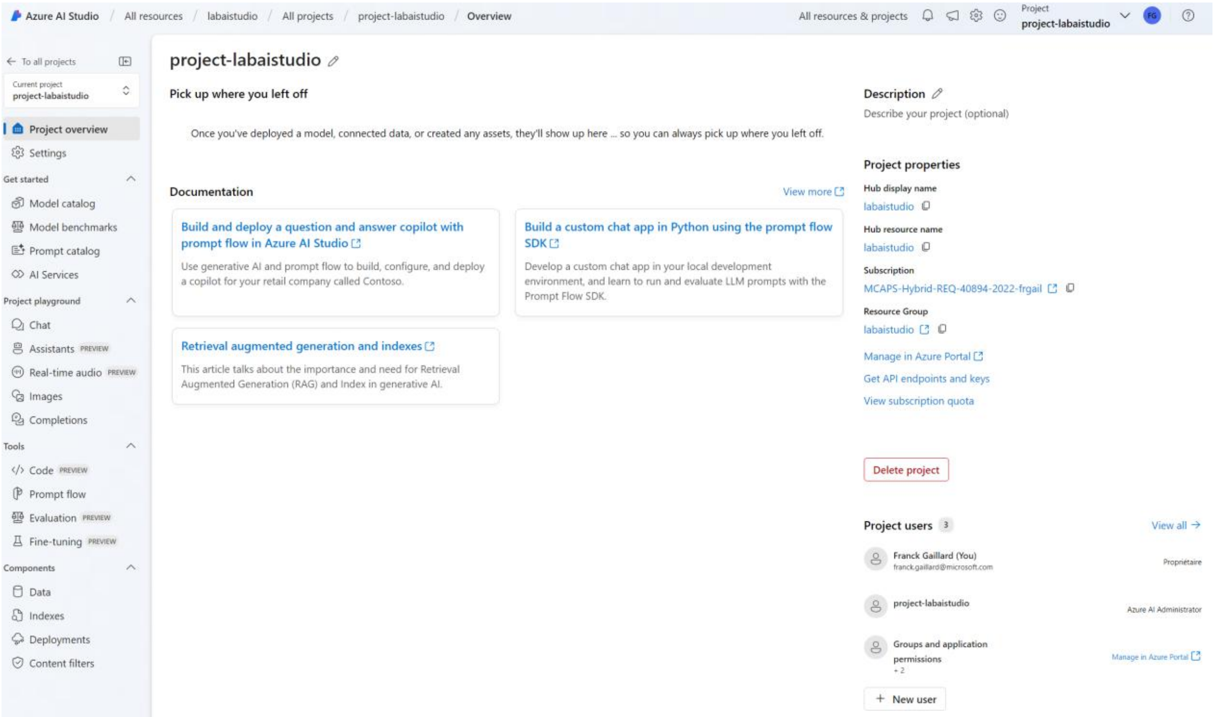


Create a project

Current hub
labaistudio

Project name *
project-labaistudio

Projects using the same hub share security settings, and can share artifacts like data connections. Learn more



project-labaistudio

Pick up where you left off

Once you've deployed a model, connected data, or created any assets, they'll show up here ... so you can always pick up where you left off.

Documentation

Build and deploy a question and answer copilot with prompt flow in Azure AI Studio

Use generative AI and prompt flow to build, configure, and deploy a copilot for your retail company called Contoso.

Build a custom chat app in Python using the prompt flow SDK

Develop a custom chat app in your local development environment, and learn to run and evaluate LLM prompts with the Prompt Flow SDK.

Retrieval augmented generation and indexes

This article talks about the importance and need for Retrieval Augmented Generation (RAG) and Index in generative AI.

Description

Describe your project (optional)

Project properties

Hub display name
labaistudio

Hub resource name
labaistudio

Subscription
MCAPS-Hybrid-REQ-40894-2022-frgail

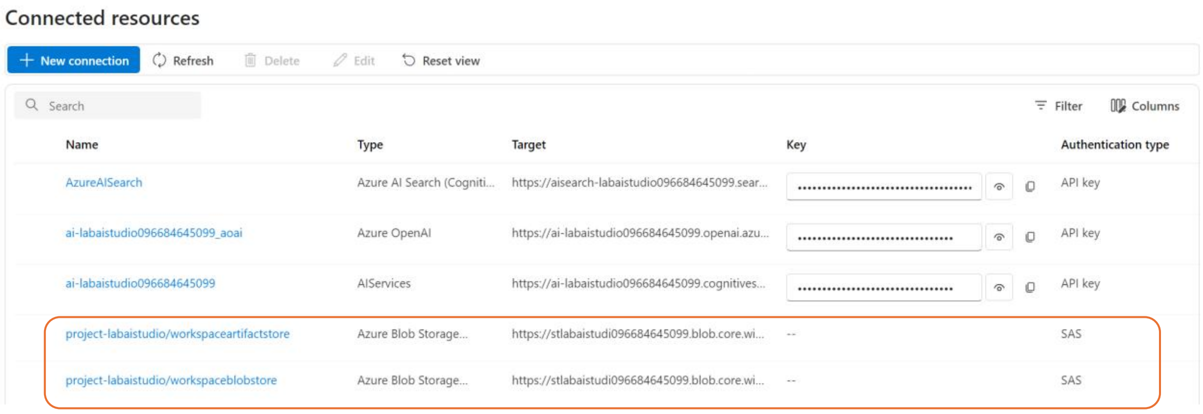
Resource Group
labaistudio

Manage in Azure Portal
Get API endpoints and keys
View subscription quota

Project users 3

Franck Gaillard (You) franck.gaillard@microsoft.com	Propriétaire
project-labaistudio	Azure AI Administrator
Groups and application permissions + 2	Manage in Azure Portal

When you click on the settings tab and view the Connected Resources:



Name	Type	Target	Key	Authentication type
AzureAISearch	Azure AI Search (Cogniti...	https://aisearch-labaistudio096684645099.sear...	••••••••••••••••••••••••••••••••••••	API key
ai-labaistudio096684645099_aoai	Azure OpenAI	https://ai-labaistudio096684645099.openai.azu...	••••••••••••••••••••••••••••••••	API key
ai-labaistudio096684645099	AIServices	https://ai-labaistudio096684645099.cognitives...	••••••••••••••••••••••••••••••••	API key
project-labaistudio/workspaceartifactstore	Azure Blob Storage...	https://stlabaistudi096684645099.blob.core.wi...	--	SAS
project-labaistudio/workspaceblobstore	Azure Blob Storage...	https://stlabaistudi096684645099.blob.core.wi...	--	SAS

Workspace Artifact Store:

- Primarily used to store various artifacts related to your AI projects, such as datasets, models, logs, and other files.
- Each project has its own dedicated storage containers within the workspace artifact store, maintaining data isolation and security.

Workspace Blob Store :

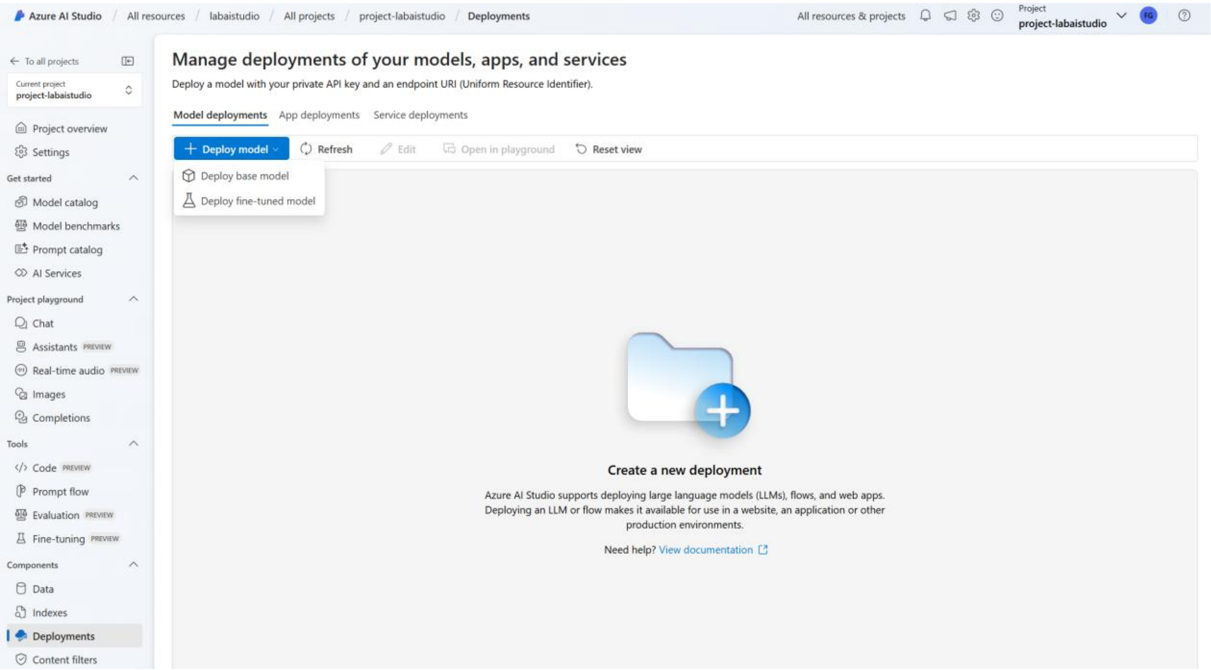
- Acts as the default blob storage for the workspace, used for general data storage needs.

  - Typically used for storing large amounts of unstructured data, such as text, images, and binary data.

Deploy a model

After creating your AI Project, the first step is to create a deployment of an Azure OpenAI model so you can start experimenting with the prompts you will use in your application.

To do this, access your newly created project, select the **Deployments** option, and click on "**Deploy model**", "**Deploy base model**":



Manage deployments of your models, apps, and services

Deploy a model with your private API key and an endpoint URI (Uniform Resource Identifier).

Create a new deployment

Azure AI Studio supports deploying large language models (LLMs), flows, and web apps. Deploying an LLM or flow makes it available for use in a website, an application or other production environments.

Need help? View documentation

⚠ In Azure AI Studio, there are **two primary types of model deployment**:

1. Serverless API

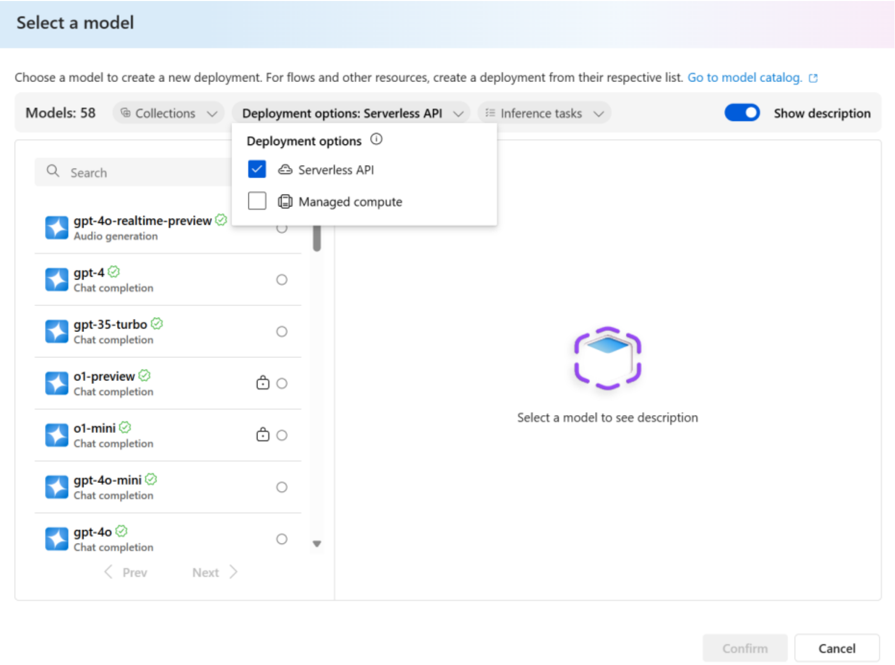
- **Description**: This deployment type allows you to deploy your model as a service without managing the underlying infrastructure.
- **Billing**: You are billed based on the number of tokens processed (pay-as-you-go).
- **Scalability**: Automatically scales to handle varying loads.
- **Use Case**: Ideal for applications where you need a flexible, cost-effective solution that can handle unpredictable traffic.

2. Managed Compute

- **Description**: This deployment type involves hosting your model on dedicated virtual machines within your Azure subscription.
- **Billing**: You are billed for the virtual machine core hours used.
- **Control**: Provides more control over the infrastructure, including the ability to configure the number of instances and manage capacity.
- **Use Case**: Suitable for scenarios requiring consistent performance, higher control over the environment, and potentially lower costs for predictable workloads.

Now, we're going to test these 2 deployments.

Serverless API deployment



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.

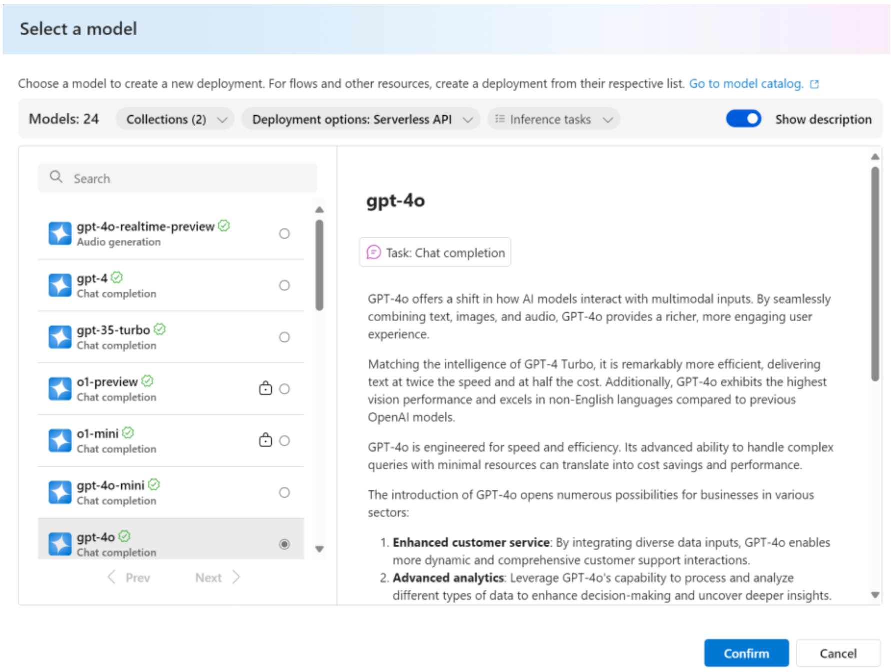
Models: 58

Deployment options: Serverless API

☑ Serverless API

☐ Managed compute

You select the model **gpt-4o** and you click on **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.

gpt-4o-realtime-preview Audio generation	○
gpt-4 Chat completion	○
gpt-35-turbo Chat completion	○
o1-preview Chat completion	○
o1-mini Chat completion	○
gpt-4o-mini Chat completion	○
gpt-4o Chat completion	●

gpt-4o

Task: Chat completion

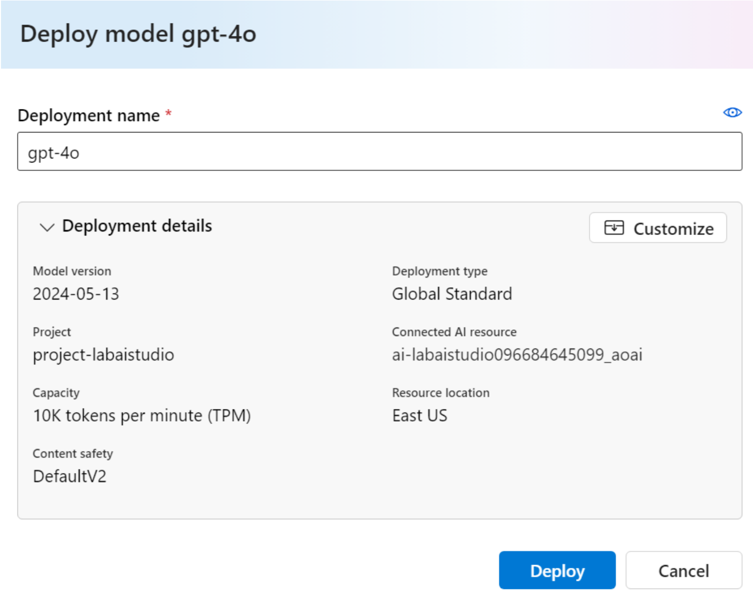
GPT-4o offers a shift in how AI models interact with multimodal inputs. By seamlessly combining text, images, and audio, GPT-4o provides a richer, more engaging user experience.

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-4o 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.



Deploy model gpt-4o

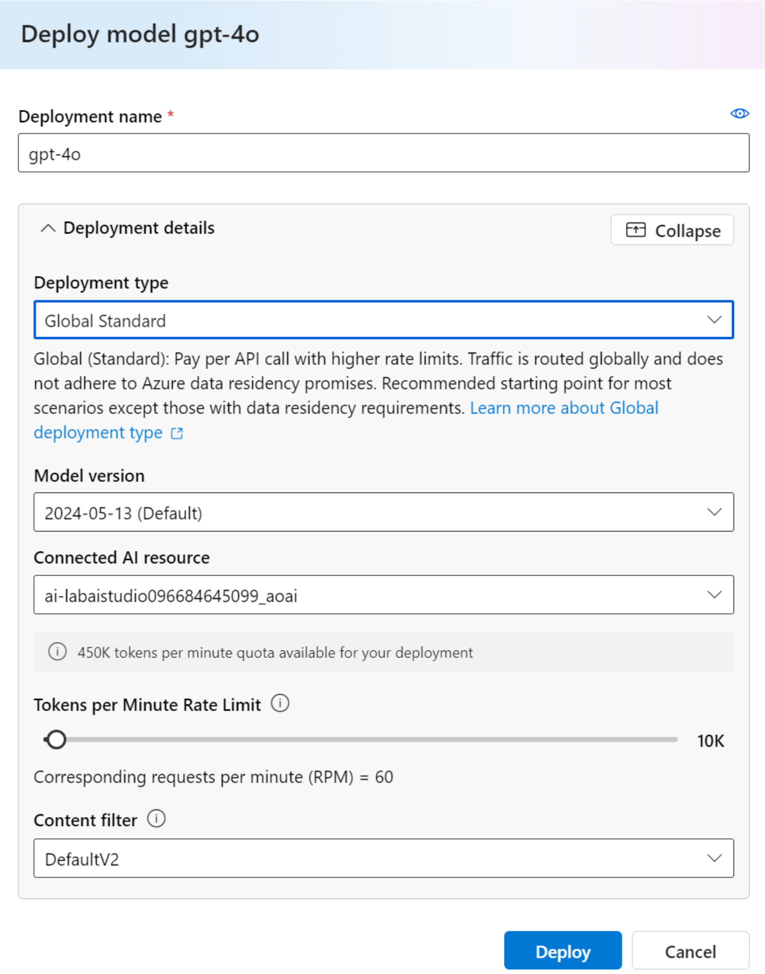
Deployment name *

gpt-4o

Deployment details

Model version 2024-05-13	Deployment type Global Standard
Project project-labaistudio	Connected AI resource ai-labaistudio096684645099_aoai
Capacity 10K tokens per minute (TPM)	Resource location East US
Content safety DefaultV2	

You click on **Customize**:



Deploy model gpt-4o

Deployment name *
gpt-4o

Deployment details

Deployment type
Global Standard

Global (Standard): Pay per API call with higher rate limits. Traffic is routed globally and does not adhere to Azure data residency promises. Recommended starting point for most scenarios except those with data residency requirements. Learn more about Global deployment type

Model version
2024-05-13 (Default)

Connected AI resource
ai-labaistudio096684645099_aoai

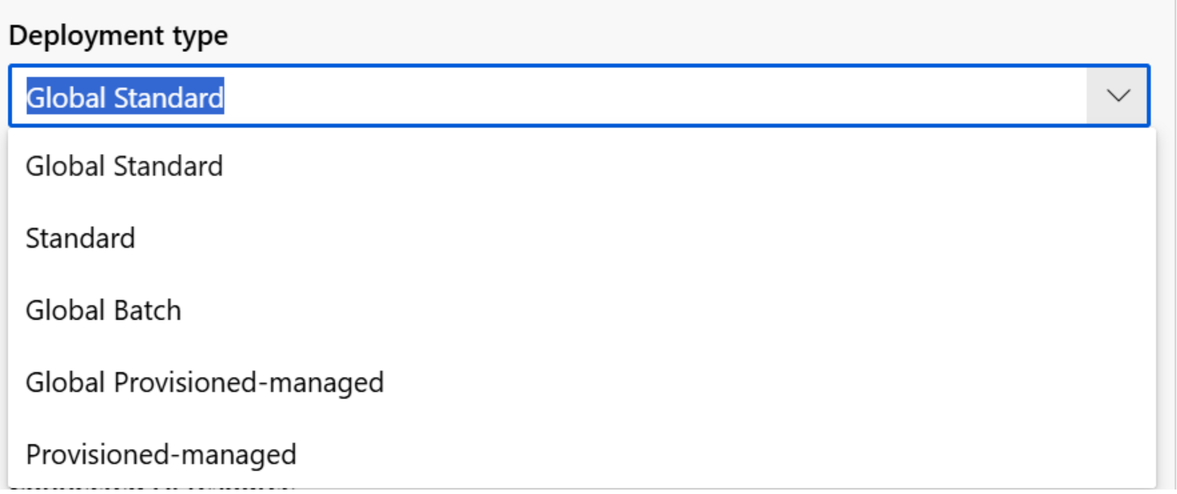
450K tokens per minute quota available for your deployment

Tokens per Minute Rate Limit
10K

Corresponding requests per minute (RPM) = 60

Content filter
DefaultV2

There are different deployment types:



Deployment type

Global Standard

- Global Standard
- Standard
- Global Batch
- Global Provisioned-managed
- Provisioned-managed

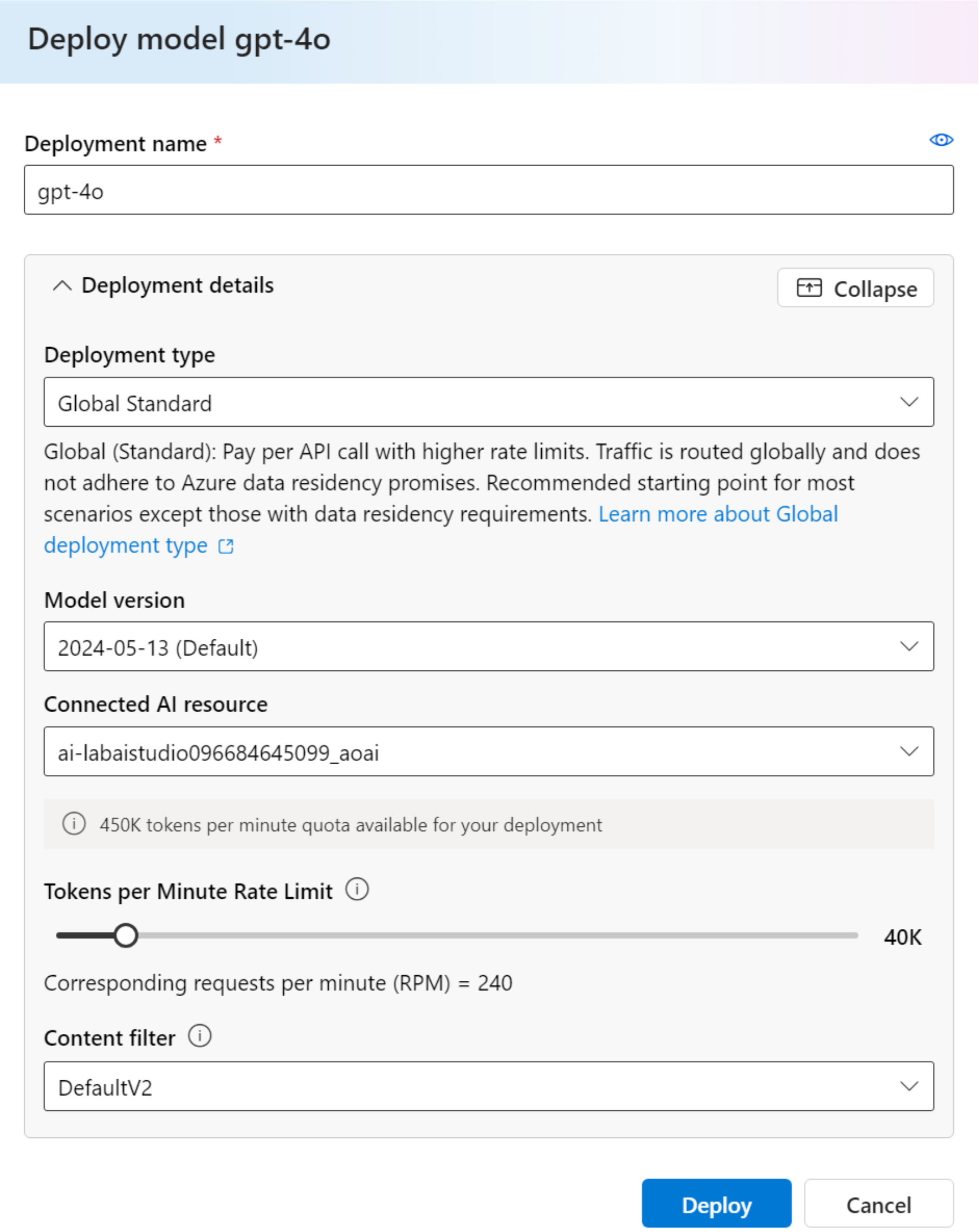
1. **Global Standard**:
    - **Description**: Your model can serve requests from users located in different parts of the world, ensuring low latency and high availability by routing traffic to the nearest region. By leveraging Azure's global infrastructure, this deployment type can handle a higher volume of requests efficiently. The deployment automatically scales to meet demand, distributing the load across multiple regions to maintain performance and reliability. While the API traffic may be processed globally, customer data is securely stored in the region specified by your Azure OpenAI Service resource, ensuring compliance with regional data residency requirements.
    - **Use Case**: Suitable for applications needing consistent performance across multiple regions. For example, an e-commerce company operates in multiple countries and wants to deploy an AI-powered recommendation system to enhance the shopping experience for its global customer base.
2. **Standard**:
    - **Description**: A standard deployment within a specific region.
    - **Use Case**: Ideal for regional applications where global availability is not required.
3. **Global Batch**:
    - **Description**: A deployment type optimized for batch processing tasks on a global scale. It is designed for large-scale, high-volume processing tasks. It handles large groups of requests asynchronously, meaning tasks

are processed in the background without immediate response. This deployment type is cost-effective, offering processing at 50% less cost compared to the Global Standard deployment

   - **Use Case**: Best for large-scale data processing jobs that can be executed in batches. For example, a company needs to process millions of customer feedback entries to perform sentiment analysis and extract key insights.

4. **Provisioned-managed**:
   - **Description**: A deployment with provisioned and managed resources within a specific region. It is designed to provide guaranteed throughput and predictable performance for your AI models. You specify the amount of throughput you need (Provisioned Throughput Units), and Azure allocates the necessary processing capacity. The required processing capacity is reserved for your deployment, ensuring that it is always available when needed, regardless of actual usage. You are billed based on the provisioned capacity, typically at a flat hourly rate. This can be more cost-effective for high-throughput workloads compared to token-based consumption.
   - **Use Case**: Ideal for applications needing dedicated resources and regional focus.
5. **Global Provisioned-managed**:
   - **Description**: A globally available deployment with provisioned and managed resources. It is designed to provide guaranteed throughput and predictable performance on a global scale. Your AI models are accessible from multiple regions worldwide, ensuring low latency and high availability by routing traffic to the nearest data center. You specify the amount of throughput you need, and Azure allocates the necessary processing capacity. This ensures consistent performance and the ability to handle the specified load. The required processing capacity is reserved for your deployment, ensuring it is always available when needed. Billed based on the provisioned capacity, typically at a flat hourly rate.
   - **Use Case**: Suitable for applications requiring dedicated resources and global reach.

You select Global Standard, select at least 40K Tokens per Minute (TPM) Rate Limit and you click on **Deploy:**



Deploy model gpt-4o

Deployment name *

gpt-4o

Deployment details

Deployment type

Global Standard

Global (Standard): Pay per API call with higher rate limits. Traffic is routed globally and does not adhere to Azure data residency promises. Recommended starting point for most scenarios except those with data residency requirements. Learn more about Global deployment type

Model version

2024-05-13 (Default)

Connected AI resource

ai-labaistudio096684645099_aoai

450K tokens per minute quota available for your deployment

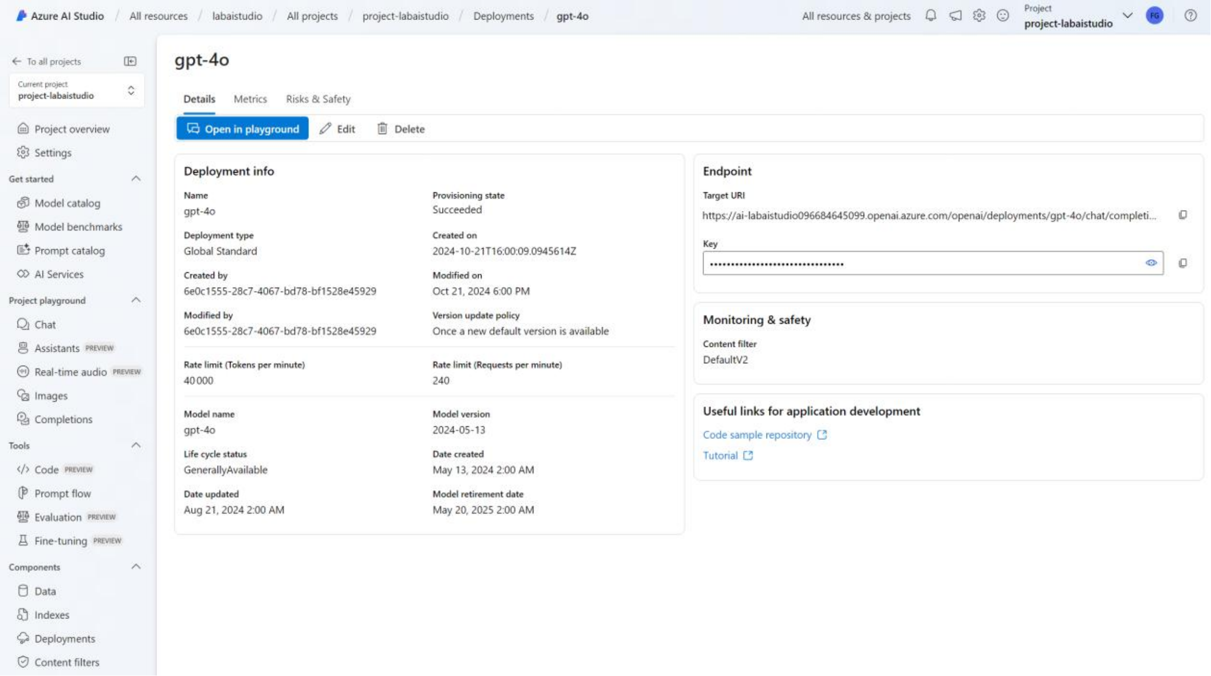
Tokens per Minute Rate Limit

40K

Corresponding requests per minute (RPM) = 240

Content filter

DefaultV2



gpt-4o

Deployment info

Name: gpt-4o
Provisioning state: Succeeded
Deployment type: Global Standard
Created on: 2024-10-21T16:00:09.0945614Z
Created by: 6e0c1555-28c7-4067-bd78-bf1528e45929
Modified on: Oct 21, 2024 6:00 PM
Modified by: 6e0c1555-28c7-4067-bd78-bf1528e45929
Version update policy: Once a new default version is available
Rate limit (Tokens per minute): 40000
Rate limit (Requests per minute): 240
Model name: gpt-4o
Model version: 2024-05-13
Life cycle status: GenerallyAvailable
Date created: May 13, 2024 2:00 AM
Date updated: Aug 21, 2024 2:00 AM
Model retirement date: May 20, 2025 2:00 AM

Endpoint

Target URI: https://ai-labaistudio096684645099.openai.azure.com/openai/deployments/gpt-4o/chat/completi...
Key: ••••••••••••••••••••••••••••••••

Monitoring & safety

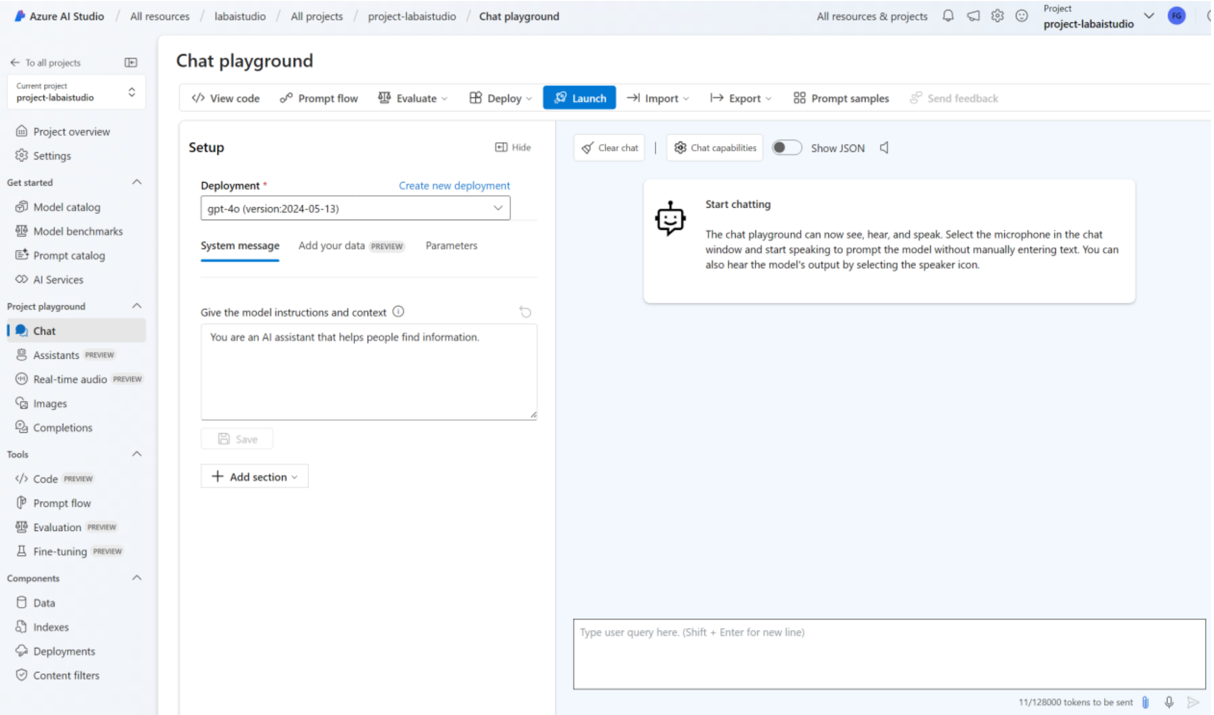
Content filter
DefaultV2

Useful links for application development

Code sample repository
Tutorial

Test the model (Serverless API)

You go to the **Project Playground** and you click on **Chat**:



Chat playground

Setup

Deployment *: gpt-4o (version:2024-05-13)

Give the model instructions and context

You are an AI assistant that helps people find information.

Start chatting

The chat playground can now see, hear, and speak. Select the microphone in the chat window and start speaking to prompt the model without manually entering text. You can also hear the model's output by selecting the speaker icon.

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

11/128000 tokens to be sent

In this lab, we will run an example where the model will help us summarize and extract information from a conversation between a customer and a representative of a telco company.

Copy the following prompt into the system message field of the playground:

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive. You always try to extract and format as a JSON:

1. Customer Name [name]

2. Customer Contact Phone [phone]

3. Main Topic of the Conversation [topic]

4. Customer Sentiment (Neutral, Positive, Negative)[sentiment]

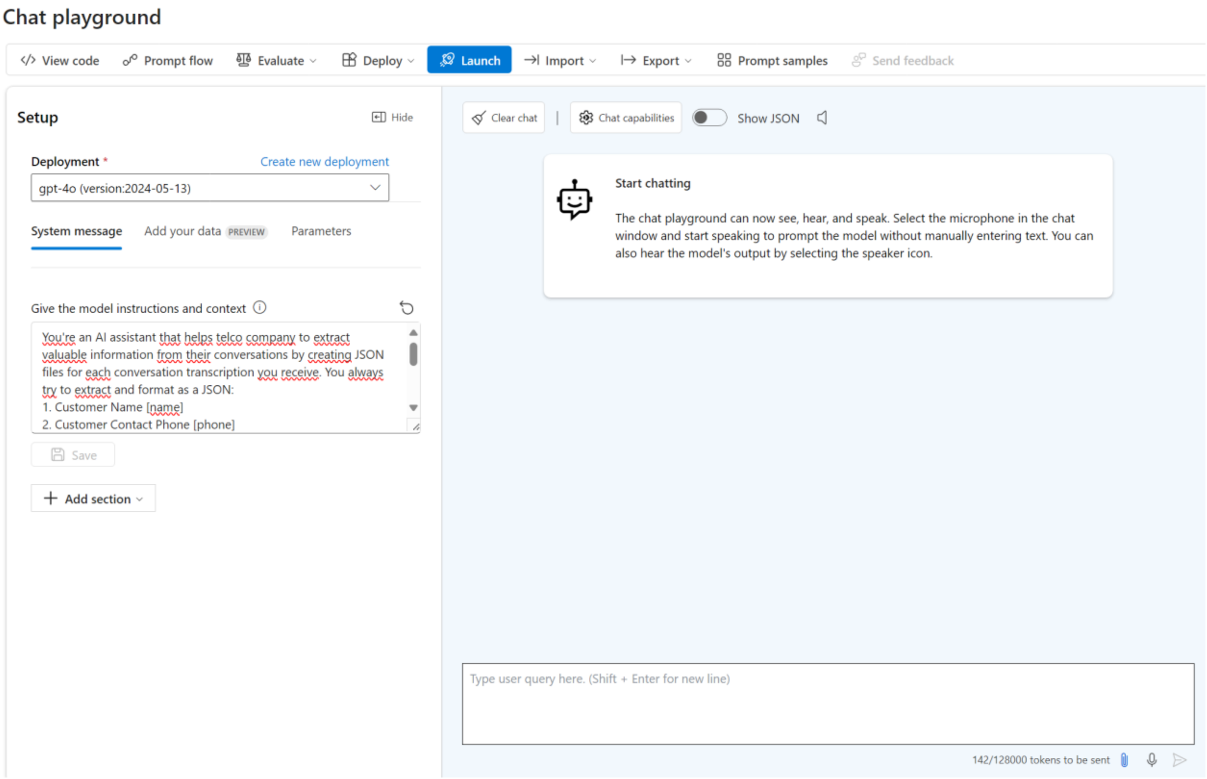
5. How the Agent Handled the Conversation [agent_behavior]

6. What was the FINAL Outcome of the Conversation [outcome]

7. A really brief Summary of the Conversation [summary]

Only extract information that you're sure. If you're unsure, write "Unknown/Not Found" in the JSON file.

After copying, select **Save.**



Deployment *: gpt-4o (version:2024-05-13)

Create new deployment

Give the model instructions and context

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive. You always try to extract and format as a JSON:
1. Customer Name [name]
2. Customer Contact Phone [phone]

Start chatting

The chat playground can now see, hear, and speak. Select the microphone in the chat window and start speaking to prompt the model without manually entering text. You can also hear the model's output by selecting the speaker icon.

Then type the following text in the chat session and click the send button:

Agent: Hello, welcome to Telco's customer service. My name is Juan, how can I assist you?

Client: Hello, Juan. I'm calling because I'm having issues with my mobile data plan. It's very slow and I can't browse the internet or use my apps.

Agent: I'm very sorry for the inconvenience, sir. Could you please tell me your phone number and your full name?

Client: Yes, sure. My number is 011-4567-8910 and my name is Martín Pérez.

Agent: Thank you, Mr. Pérez. I'm going to check your plan and your data usage. One moment, please.

Client: Okay, thank you.

Agent: Mr. Pérez, I've reviewed your plan and I see that you have contracted the basic plan of 2 GB of data per month. Is that correct?

Client: Yes, that's correct.

Agent: Well, I inform you that you have consumed 90% of your data limit and you only have 200 MB available until the end of the month. That's why your browsing speed has been reduced.

Client: What? How is that possible? I barely use the internet on my cell phone. I only check my email and my social networks from time to time. I don't watch videos or download large files.

Agent: I understand, Mr. Pérez. But keep in mind that some applications consume data in the background, without you realizing it. For example, automatic updates, backups, GPS, etc.

Client: Well, but they didn't explain that to me when I contracted the plan. They told me that with 2 GB I would have enough for the whole month. I feel cheated.

Agent: I apologize, Mr. Pérez. It was not our intention to deceive you. I offer you a solution: if you want, you can change your plan to a higher one, with more GB of data and higher speed. This way you can enjoy a better browsing experience.

Client: And how much would that cost me?

Agent: We have a special offer for you. For only 10 pesos more per month, you can access the premium plan of 5 GB of data and 4G speed. Are you interested?

Client: Mmm, I don't know. Isn't there another option? Can't you give me more speed without charging me more?

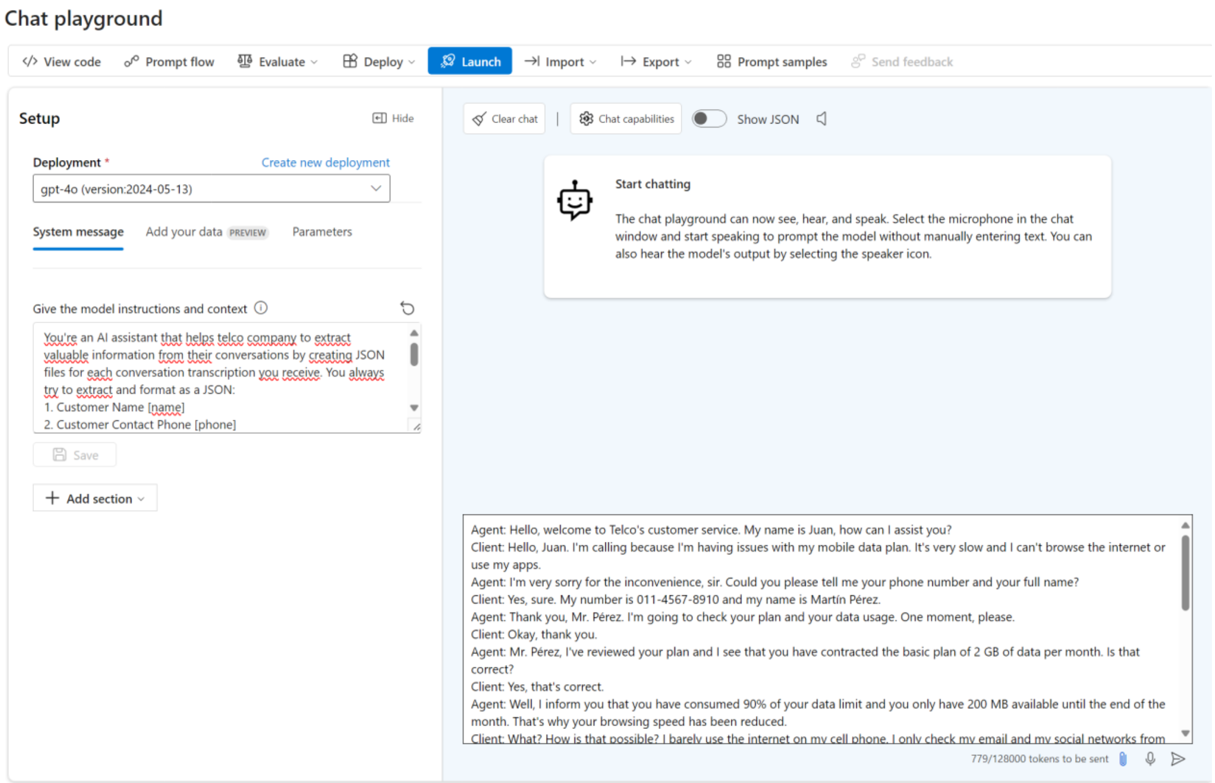
Agent: I'm sorry, Mr. Pérez. That's the only option we have available. If you don't change your plan, you'll have to wait until next month to recover your normal speed. Or you can buy an additional data package, but it would be more expensive than changing plans.

Client: Well, let me think about it. Can I call later to confirm?

Agent: Of course, Mr. Pérez. You can call whenever you want. The number is the same one you dialed now. Is there anything else I can help you with?

Client: No, that's all. Thank you for your attention.

Agent: Thank you, Mr. Pérez. Have a good day. Goodbye.



Deployment *: gpt-4o (version:2024-05-13)

Give the model instructions and context

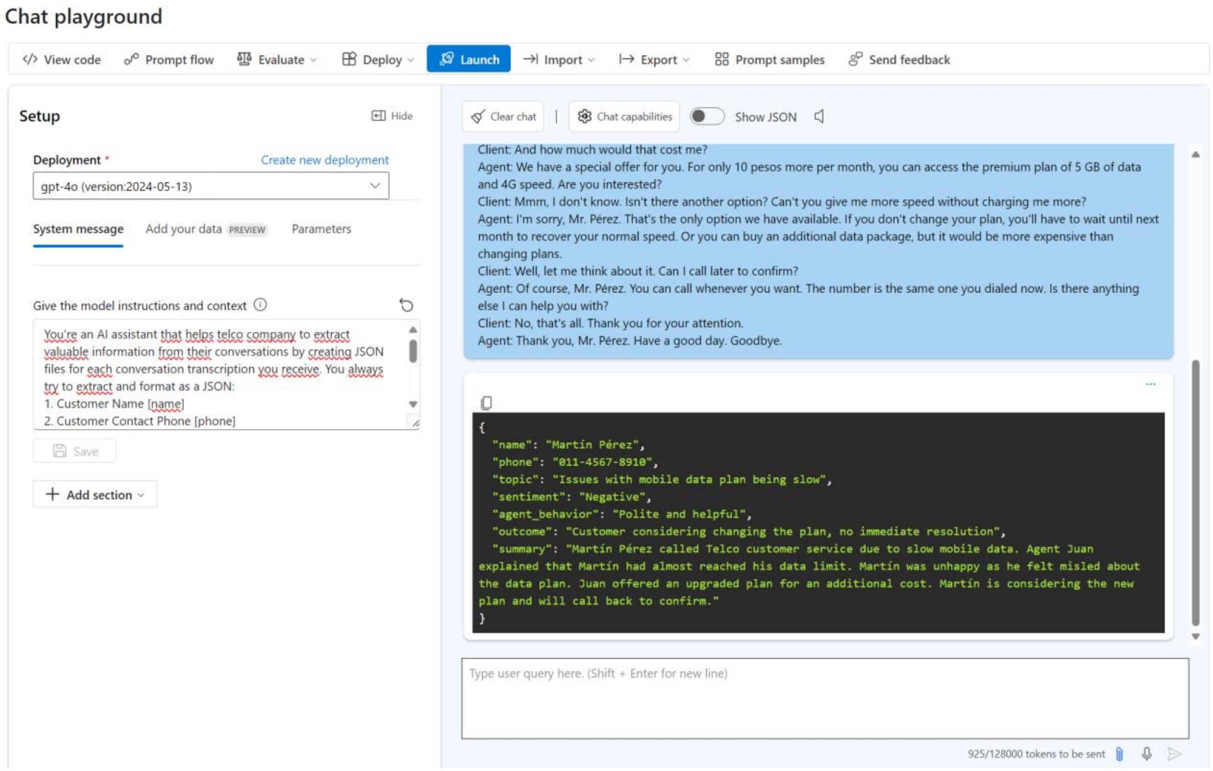
You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive. You always try to extract and format as a JSON:
1. Customer Name [name]
2. Customer Contact Phone [phone]

Start chatting

The chat playground can now see, hear, and speak. Select the microphone in the chat window and start speaking to prompt the model without manually entering text. You can also hear the model's output by selecting the speaker icon.

Agent: Hello, welcome to Telco's customer service. My name is Juan, how can I assist you?
Client: Hello, Juan. I'm calling because I'm having issues with my mobile data plan. It's very slow and I can't browse the internet or use my apps.
Agent: I'm very sorry for the inconvenience, sir. Could you please tell me your phone number and your full name?
Client: Yes, sure. My number is 011-4567-8910 and my name is Martín Pérez.
Agent: Thank you, Mr. Pérez. I'm going to check your plan and your data usage. One moment, please.
Client: Okay, thank you.
Agent: Mr. Pérez, I've reviewed your plan and I see that you have contracted the basic plan of 2 GB of data per month. Is that correct?
Client: Yes, that's correct.
Agent: Well, I inform you that you have consumed 90% of your data limit and you only have 200 MB available until the end of the month. That's why your browsing speed has been reduced.
Client: What? How is that possible? I barely use the internet on my cell phone. I only check my email and my social networks from

779/128000 tokens to be sent



Deployment *: gpt-4o (version:2024-05-13)

Give the model instructions and context

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive. You always try to extract and format as a JSON:
1. Customer Name [name]
2. Customer Contact Phone [phone]

Client: And how much would that cost me?
Agent: We have a special offer for you. For only 10 pesos more per month, you can access the premium plan of 5 GB of data and 4G speed. Are you interested?
Client: Mmm, I don't know. Isn't there another option? Can't you give me more speed without charging me more?
Agent: I'm sorry, Mr. Pérez. That's the only option we have available. If you don't change your plan, you'll have to wait until next month to recover your normal speed. Or you can buy an additional data package, but it would be more expensive than changing plans.
Client: Well, let me think about it. Can I call later to confirm?
Agent: Of course, Mr. Pérez. You can call whenever you want. The number is the same one you dialed now. Is there anything else I can help you with?
Client: No, that's all. Thank you for your attention.
Agent: Thank you, Mr. Pérez. Have a good day. Goodbye.

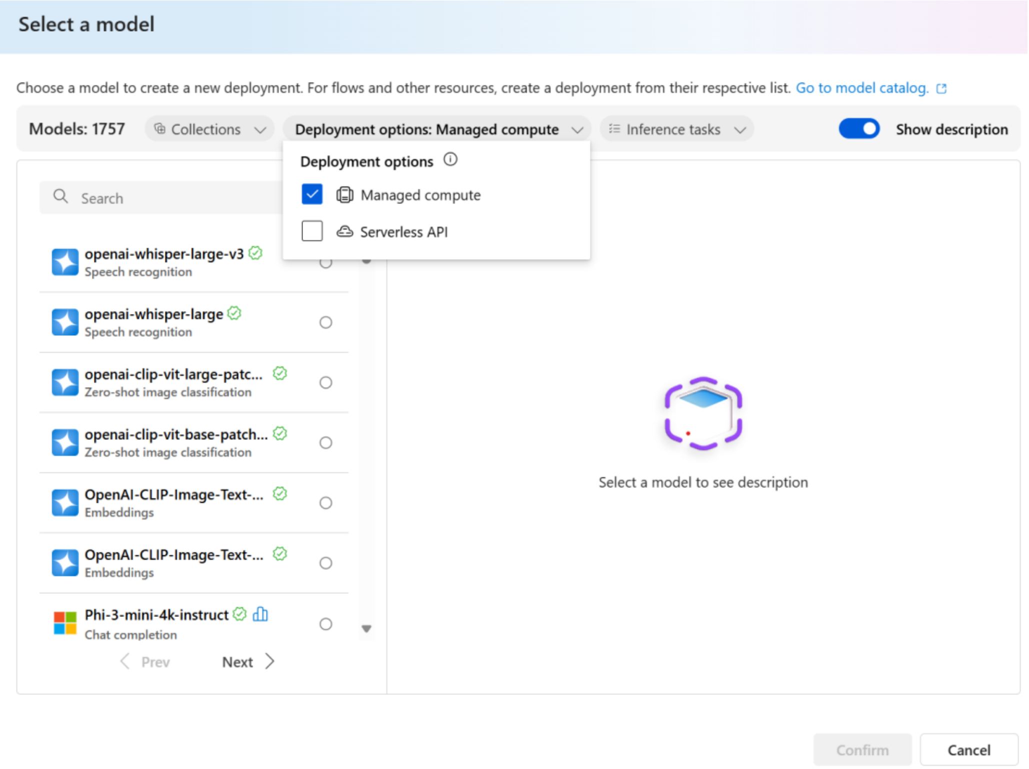
```
{
  "name": "Martín Pérez",
  "phone": "011-4567-8910",
  "topic": "Issues with mobile data plan being slow",
  "sentiment": "Negative",
  "agent_behavior": "Polite and helpful",
  "outcome": "Customer considering changing the plan, no immediate resolution",
  "summary": "Martín Pérez called Telco customer service due to slow mobile data. Agent Juan explained that Martín had almost reached his data limit. Martín was unhappy as he felt misled about the data plan. Juan offered an upgraded plan for an additional cost. Martín is considering the new plan and will call back to confirm."
}
```

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

925/128000 tokens to be sent

Managed Compute deployment

The same way we did it for **Serverless API** deployment, this time we choose **Managed Compute** deployment.


Select a model

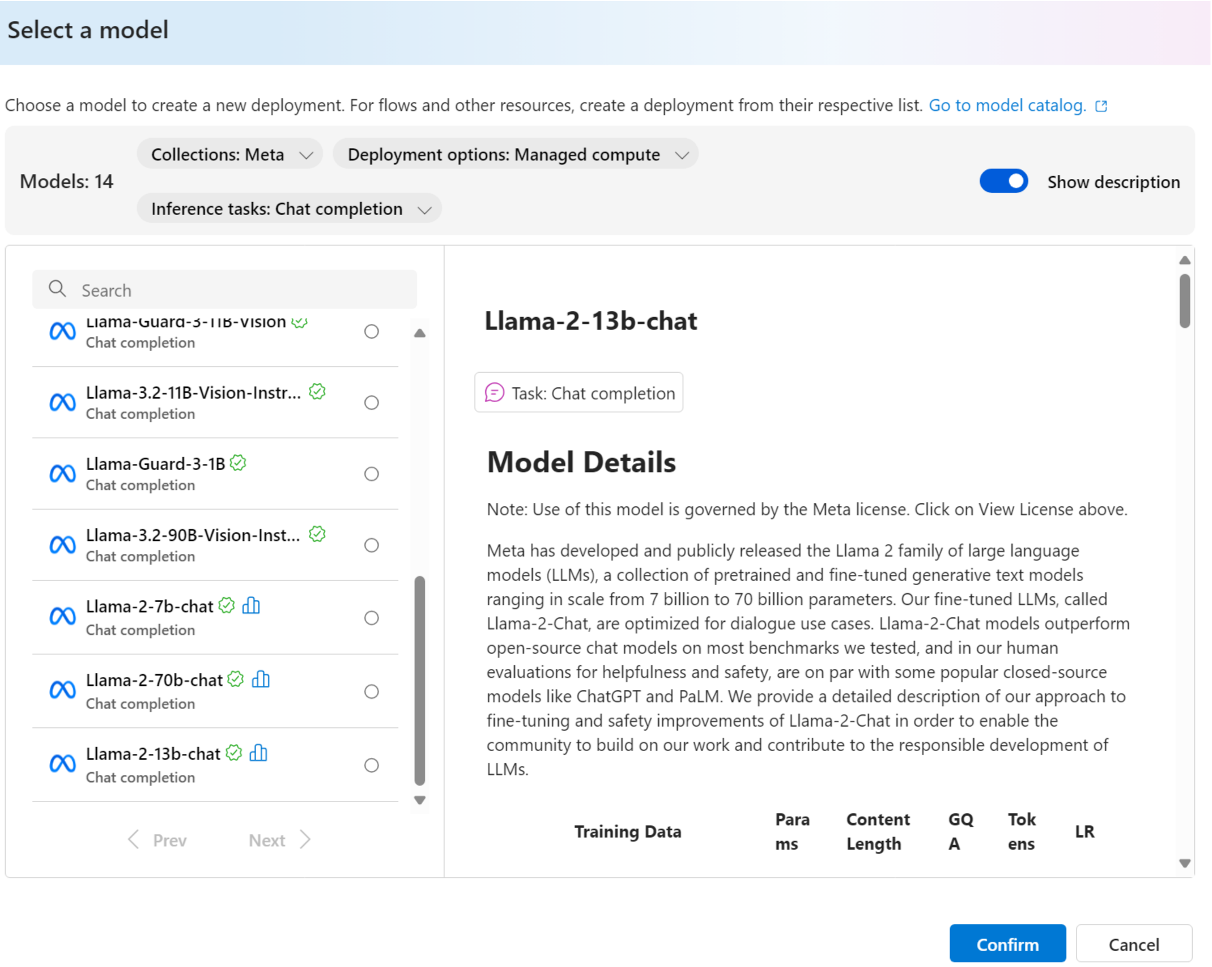
Choose a model to create a new deployment. For flows and other resources, create a deployment from their respective list. Go to model catalog.

Models: 1757

- openai-whisper-large-v3 — Speech recognition ○
- openai-whisper-large — Speech recognition ○
- openai-clip-vit-large-patc... — Zero-shot image classification ○
- openai-clip-vit-base-patch... — Zero-shot image classification ○
- OpenAI-CLIP-Image-Text-... — Embeddings ○
- OpenAI-CLIP-Image-Text-... — Embeddings ○
- Phi-3-mini-4k-instruct — Chat completion ○

Select a model to see description

We select the model: **Llama-2-13b-chat**


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: 14

Collections: Meta
Deployment options: Managed compute
Inference tasks: Chat completion
Show description
Search

- Llama-Guard-3-11B-Vision — Chat completion ○
- Llama-3.2-11B-Vision-Instr... — Chat completion ○
- Llama-Guard-3-1B — Chat completion ○
- Llama-3.2-90B-Vision-Inst... — Chat completion ○
- Llama-2-7b-chat — Chat completion ○
- Llama-2-70b-chat — Chat completion ○
- Llama-2-13b-chat — Chat completion ○

Llama-2-13b-chat

Task: Chat completion

Model Details

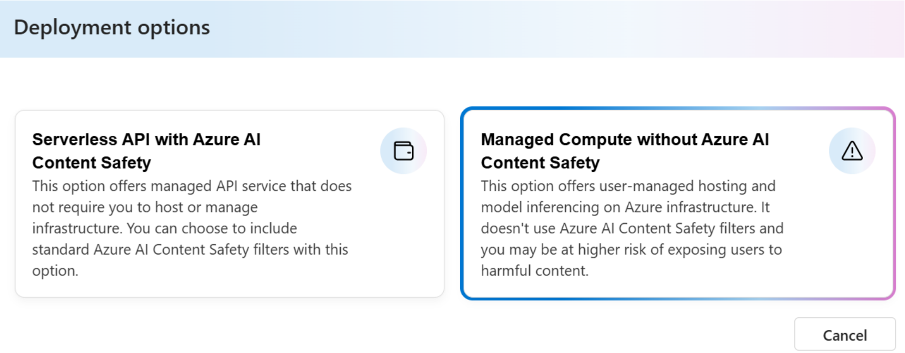
Note: Use of this model is governed by the Meta license. Click on View License above.

Meta has developed and publicly released the Llama 2 family of large language models (LLMs), a collection of pretrained and fine-tuned generative text models ranging in scale from 7 billion to 70 billion parameters. Our fine-tuned LLMs, called Llama-2-Chat, are optimized for dialogue use cases. Llama-2-Chat models outperform open-source chat models on most benchmarks we tested, and in our human evaluations for helpfulness and safety, are on par with some popular closed-source models like ChatGPT and PaLM. We provide a detailed description of our approach to fine-tuning and safety improvements of Llama-2-Chat in order to enable the community to build on our work and contribute to the responsible development of LLMs.

Training Data	Params	Content Length	GQA	Tokens	LR

Click on **Confirm**.

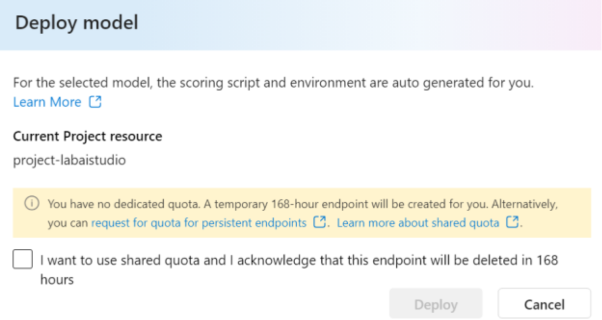
Select **Managed Compute without Azure AI Content Safety**.



Deployment options

Serverless API with Azure AI Content Safety
This option offers managed API service that does not require you to host or manage infrastructure. You can choose to include standard Azure AI Content Safety filters with this option.

Managed Compute without Azure AI Content Safety
This option offers user-managed hosting and model inferencing on Azure infrastructure. It doesn't use Azure AI Content Safety filters and you may be at higher risk of exposing users to harmful content.



Deploy model

For the selected model, the scoring script and environment are auto generated for you.
Learn More

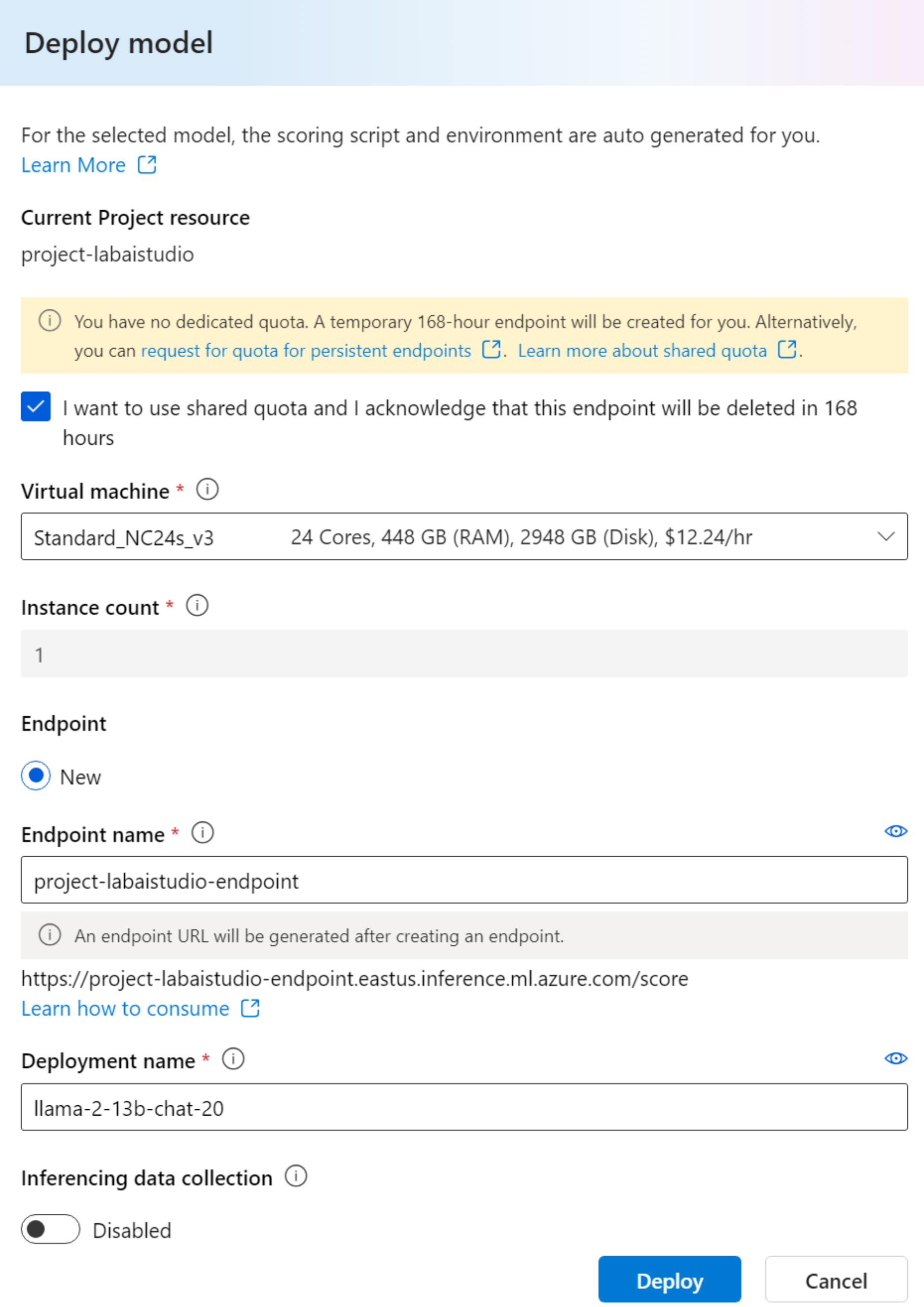
Current Project resource
project-labaistudio

You have no dedicated quota. A temporary 168-hour endpoint will be created for you. Alternatively, you can request for quota for persistent endpoints. Learn more about shared quota.

☐ I want to use shared quota and I acknowledge that this endpoint will be deleted in 168 hours

Check the cell “I want to use shared quota, and I acknowledge that this endpoint will be deleted in 168 hours”.

Select the **Standard_NC24s_v3** compute for inference with the selected model, for this workshop one instance is enough.



Deploy model

For the selected model, the scoring script and environment are auto generated for you.
Learn More

Current Project resource
project-labaistudio

ⓘ You have no dedicated quota. A temporary 168-hour endpoint will be created for you. Alternatively, you can request for quota for persistent endpoints. Learn more about shared quota.

☑ I want to use shared quota and I acknowledge that this endpoint will be deleted in 168 hours

Virtual machine *: Standard_NC24s_v3 24 Cores, 448 GB (RAM), 2948 GB (Disk), $12.24/hr

Instance count *: 1

Endpoint: ● New

Endpoint name *: project-labaistudio-endpoint

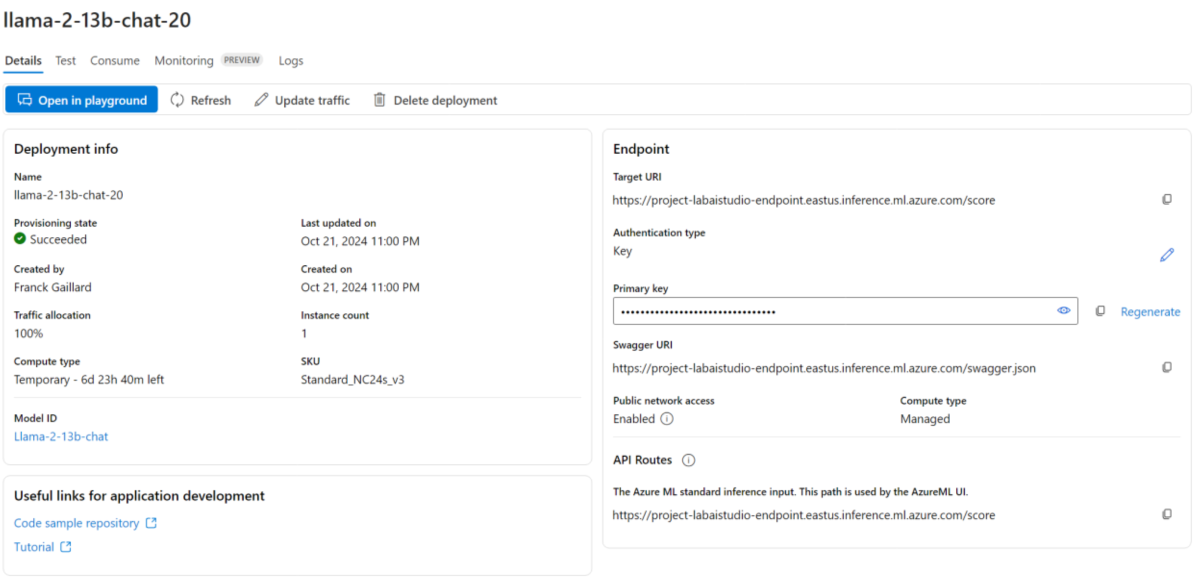
ⓘ An endpoint URL will be generated after creating an endpoint.

https://project-labaistudio-endpoint.eastus.inference.ml.azure.com/score
Learn how to consume

Deployment name *: llama-2-13b-chat-20

Inferencing data collection: Disabled

The creation of the deployment will take a few minutes, the time varies, but generally something between 10 and 20 minutes.



llama-2-13b-chat-20

Deployment info

Name
llama-2-13b-chat-20

Provisioning state
Succeeded

Last updated on
Oct 21, 2024 11:00 PM

Created by
Franck Gaillard

Created on
Oct 21, 2024 11:00 PM

Traffic allocation
100%

Instance count
1

Compute type
Temporary - 6d 23h 40m left

SKU
Standard_NC24s_v3

Model ID
Llama-2-13b-chat

Useful links for application development

Code sample repository

Tutorial

Endpoint

Target URI
https://project-labaistudio-endpoint.eastus.inference.ml.azure.com/score

Authentication type
Key

Primary key
••••••••••••••••••••••••••••••••• Regenerate

Swagger URI
https://project-labaistudio-endpoint.eastus.inference.ml.azure.com/swagger.json

Public network access
Enabled

Compute type
Managed

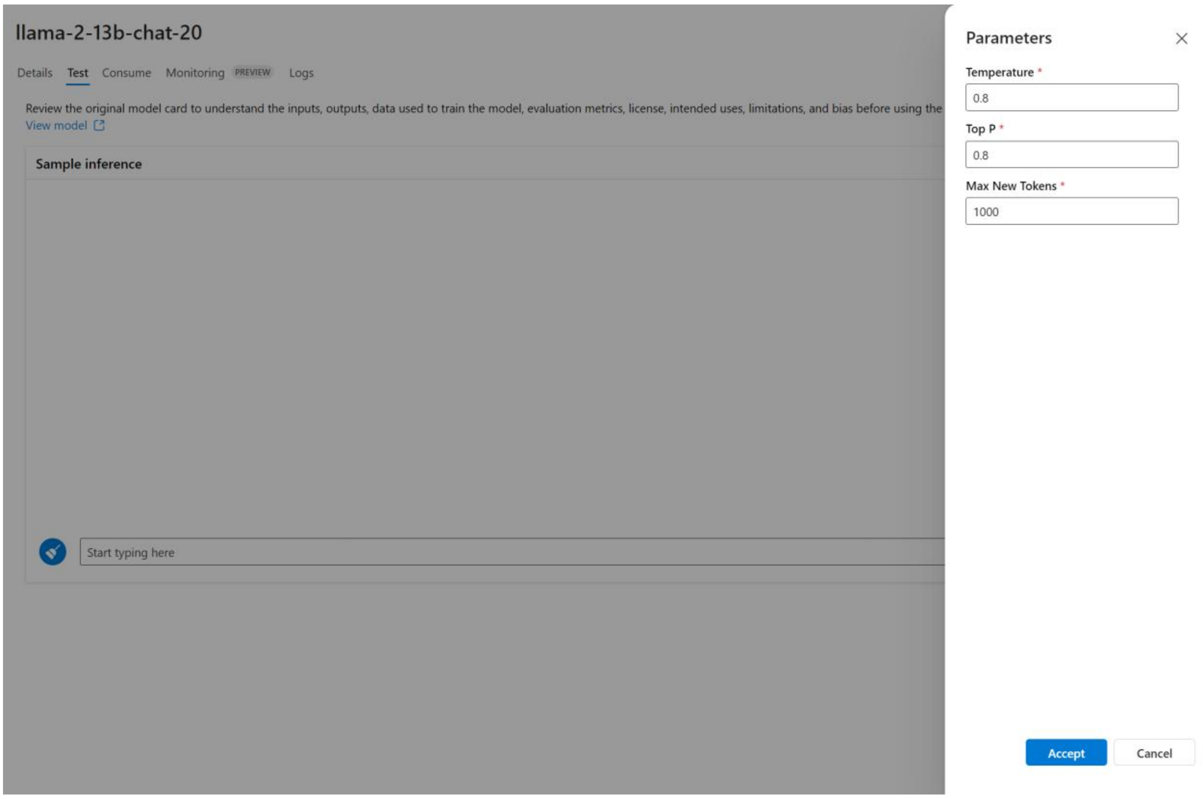
API Routes

The Azure ML standard inference input. This path is used by the AzureML UI.
https://project-labaistudio-endpoint.eastus.inference.ml.azure.com/score

Test the model (Managed Compute)

Let's test this model by selecting the **Test** option on the deployment page.

Adjust the **max_next_tokens** parameter to 1000 so we can test the same example we used with the gpt-4o model.



llama-2-13b-chat-20

Review the original model card to understand the inputs, outputs, data used to train the model, evaluation metrics, license, intended uses, limitations, and bias before using the

Parameters

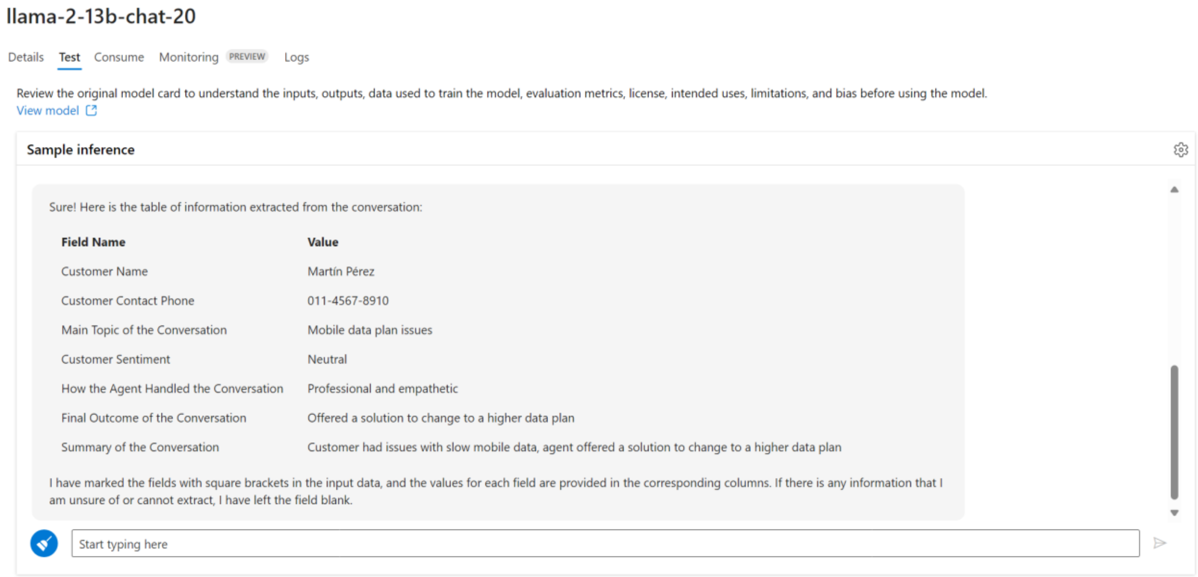
Temperature *
0.8

Top P *
0.8

Max New Tokens *
1000

Now just copy the text below into the "Start typing text box" and then send to observe the response generated by the Llama2 model.

```
{"input_data": {
  "input_string": [
    {
      "role": "system",
      "content": "You're an AI assistant that helps telco company to extract valuable information from their conversations by creating a table for each conversation transcription you receive. You always try to extract and format as a table, fields names between square brackets: 1. Customer Name [name] 2. Customer Contact Phone [phone] 3. Main Topic of the Conversation [topic] 4. Customer Sentiment (Neutral, Positive, Negative)[sentiment] 5. How the Agent Handled the Conversation [agent_behavior] 6. What was the FINAL Outcome of the Conversation [outcome] 7. A really brief Summary of the Conversation [summary] Only extract information that you're sure. If you're unsure, write 'Unknown/Not Found' in the table. Your answers outputs contains only the table document."
    },
    {
      "role": "user",
      "content": "Agent: Hello, welcome to Telco's customer service. My name is Juan, how can I assist you? Client: Hello, Juan. I'm calling because I'm having issues with my mobile data plan. It's very slow and I can't browse the internet or use my apps. Agent: I'm very sorry for the inconvenience, sir. Could you please tell me your phone number and your full name? Client: Yes, sure. My number is 011-4567-8910 and my name is Martín Pérez. Agent: Thank you, Mr. Pérez. I'm going to check your plan and your data usage. One moment, please. Client: Okay, thank you. Agent: Mr. Pérez, I've reviewed your plan and I see that you have contracted the basic plan of 2 GB of data per month. Is that correct? Client: Yes, that's correct. Agent: Well, I inform you that you have consumed 90% of your data limit and you only have 200 MB available until the end of the month. That's why your browsing speed has been reduced. Client: What? How is that possible? I barely use the internet on my cell phone. I only check my email and my social networks from time to time. I don't watch videos or download large files. Agent: I understand, Mr. Pérez. But keep in mind that some applications consume data in the background, without you realizing it. For example, automatic updates, backups, GPS, etc. Client: Well, but they didn't explain that to me when I contracted the plan. They told me that with 2 GB I would have enough for the whole month. I feel cheated. Agent: I apologize, Mr. Pérez. It was not our intention to deceive you. I offer you a solution: if you want, you can change your plan to a higher one, with more GB of data and higher speed. This way you can enjoy a better browsing experience. Client: And how much would that cost me? Agent: We have a special offer for you. For only 10 pesos more per month, you can access the premium plan of 5 GB of data and 4G speed. Are you interested? Client: Mmm, I don't know. Isn't there another option? Can't you give me more speed without charging me more? Agent: I'm sorry, Mr. Pérez. That's the only option we have available. If you don't change your plan, you'll have to wait until next month to recover your normal speed. Or you can buy an additional data package, but it would be more expensive than changing plans. Client: Well, let me think about it. Can I call later to confirm? Agent: Of course, Mr. Pérez. You can call whenever you want. The number is the same one you dialed now. Is there anything else I can help you with? Client: No, that's all. Thank you for your attention. Agent: Thank you, Mr. Pérez. Have a good day. Goodbye."
    }
  ],
  "parameters": {
    "return_full_text": false,
    "temperature": 0.8,
    "top_p": 0.8,

    "do_sample": true,

    "max_new_tokens": 1000

  }

}

}
```



llama-2-13b-chat-20

Review the original model card to understand the inputs, outputs, data used to train the model, evaluation metrics, license, intended uses, limitations, and bias before using the model.

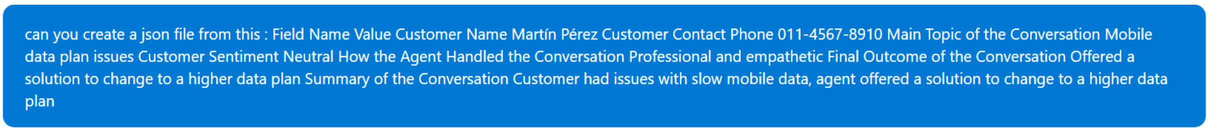
Sample inference

Sure! Here is the table of information extracted from the conversation:

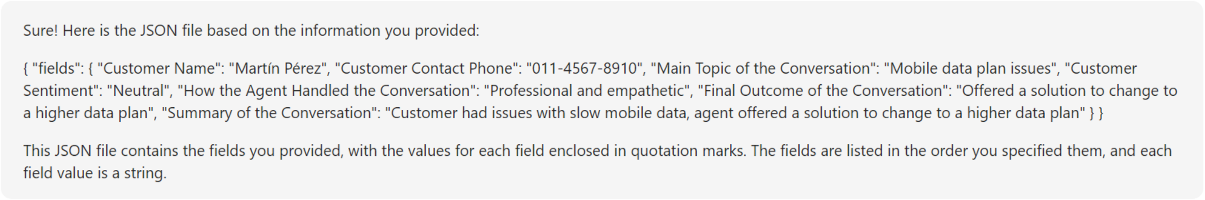
Field Name	**Value**
Customer Name	Martín Pérez
Customer Contact Phone	011-4567-8910
Main Topic of the Conversation	Mobile data plan issues
Customer Sentiment	Neutral
How the Agent Handled the Conversation	Professional and empathetic
Final Outcome of the Conversation	Offered a solution to change to a higher data plan
Summary of the Conversation	Customer had issues with slow mobile data, agent offered a solution to change to a higher data plan

I have marked the fields with square brackets in the input data, and the values for each field are provided in the corresponding columns. If there is any information that I am unsure of or cannot extract, I have left the field blank.

From this result, you can ask to structure this information in a Json file for example:



can you create a json file from this : Field Name Value Customer Name Martín Pérez Customer Contact Phone 011-4567-8910 Main Topic of the Conversation Mobile data plan issues Customer Sentiment Neutral How the Agent Handled the Conversation Professional and empathetic Final Outcome of the Conversation Offered a solution to change to a higher data plan Summary of the Conversation Customer had issues with slow mobile data, agent offered a solution to change to a higher data plan



Sure! Here is the JSON file based on the information you provided:

{ "fields": { "Customer Name": "Martín Pérez", "Customer Contact Phone": "011-4567-8910", "Main Topic of the Conversation": "Mobile data plan issues", "Customer Sentiment": "Neutral", "How the Agent Handled the Conversation": "Professional and empathetic", "Final Outcome of the Conversation": "Offered a solution to change to a higher data plan", "Summary of the Conversation": "Customer had issues with slow mobile data, agent offered a solution to change to a higher data plan" } }

This JSON file contains the fields you provided, with the values for each field enclosed in quotation marks. The fields are listed in the order you specified them, and each field value is a string.

Discover Content Safety

Responsible AI Principles

The idea behind Microsoft’s Responsible AI Principles is to ensure that AI technologies are developed and used in ways that are ethical, trustworthy, and beneficial to society. These principles guide the design, deployment, and governance of AI systems to address potential risks and maximize positive impacts.

Microsoft’s responsible AI framework is built on 6 key principles:

1. **Fairness**: Ensuring AI systems treat all users equitably.
2. **Reliability and Safety**: Guaranteeing that AI systems function as intended and are safe to use.
3. **Privacy and Security**: Protecting user data and ensuring confidentiality.
4. **Inclusiveness**: Making AI accessible and beneficial to everyone.
5. **Transparency**: Being open about how AI systems work and make decisions.
6. **Accountability**: Taking responsibility for AI systems.

Content Filtering System

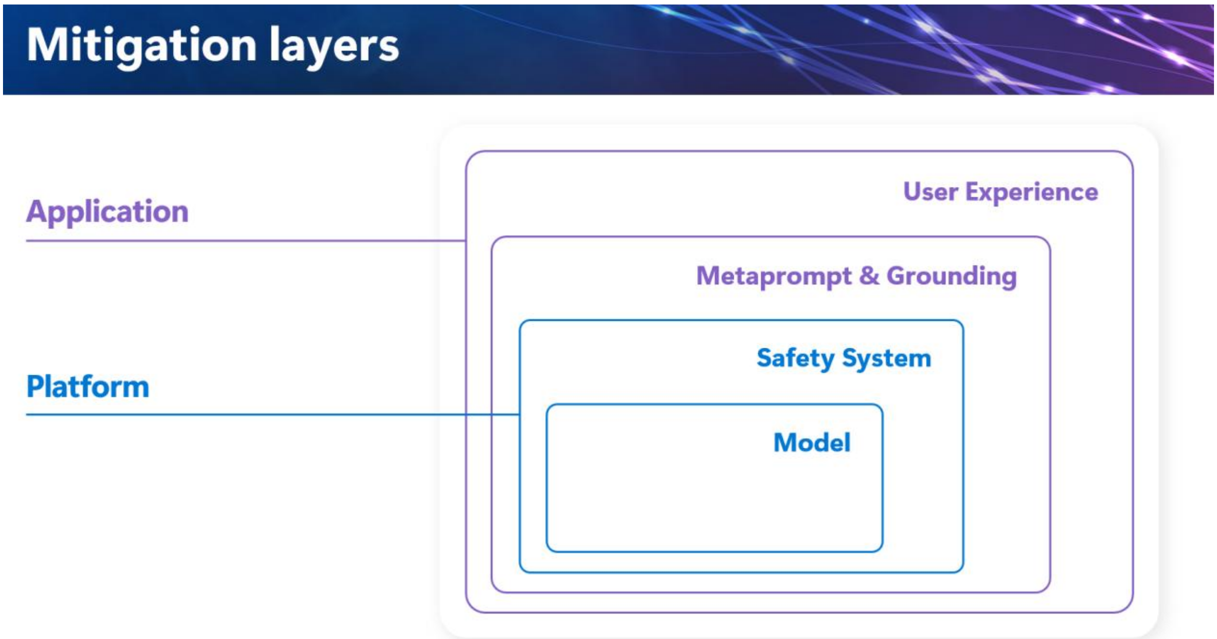
The content filtering system is a practical implementation of these principles:

1. **Fairness**:
    - Filters are designed to detect and mitigate biases in content, ensuring equitable treatment of all users.
2. **Reliability and Safety**:
    - The system prevents harmful outputs by filtering content related to violence, hate speech, explicit material, and self-harm, ensuring the AI’s reliability and user safety.
3. **Privacy and Security**:
    - Filters help protect sensitive information and prevent the dissemination of private data.
4. **Inclusiveness**:
    - By filtering out harmful content, the system promotes a safe and inclusive environment for all users.
5. **Transparency**:
    - Users are informed about the filtering mechanisms and can adjust settings to suit their needs, promoting transparency in how the AI operates.

6. **Accountability**:
    - Microsoft continuously monitors and updates the filtering system to address new challenges and ensure it aligns with ethical standards, demonstrating accountability.

Mitigation Layers

The concept of mitigation layers is applied to ensure the safe, ethical, and effective use of AI models. Here's how each layer functions within this framework:



*The **Model** layer*

The **Model** layer is where the AI processes inputs and generates outputs. In Azure OpenAI, this involves advanced neural networks like GPT-4. To ensure the outputs are safe and appropriate, Azure OpenAI employs a robust content filtering system.

The content filtering system in Azure OpenAI works alongside the core models to detect and prevent harmful content.

Here's how it functions:

1. **Multi-Class Classification Models**:
    - These models analyze both the input prompts and the output completions.
    - They are trained to detect harmful content across four main categories: **violence, hate, sexual, and self-harm**.
2. **Severity Levels**:
    - Content is classified into four severity levels: **safe, low, medium, and high**.
    - By default, content detected at medium or high severity levels is filtered out, while content at low or safe levels is not.
3. **Categories of Harmful Content**:

    - **Hate**: Content that promotes hate speech or discrimination.
    - **Sexual**: Explicit or inappropriate sexual content.
    - **Violence**: Content related to physical harm or threats.
    - **Self-Harm**: Content that promotes self-injury or suicide.
4. **Optional Filters**:
    - **Jailbreak Risk Detection**: Identifies attempts to bypass safety mechanisms.
    - **Protected Material Detection:** Flags known text or code from public repositories to prevent unauthorized use.

How It Works

- **Input and Output Analysis**: Both the user's input and the AI's output are analyzed by the classification models.
- **Filtering Actions**: If harmful content is detected at medium or high severity levels, the system filters it out. For lower severity levels, content may be annotated but not necessarily blocked.

Configurability

- Users can configure the content filters to adjust the severity thresholds for different categories. This allows for customization based on specific use cases and requirements.

The Safety System

This content filtering system is powered by Azure AI Content Safety, and it works by running both the prompt input and completion output through an ensemble of classification models aimed at detecting and preventing the output of harmful content. Variations in API configurations and application design might affect completions and thus filtering behavior.

With Azure OpenAI model deployments, **you can use the default content filter or create your own content filter** (described later on). The default content filter is also available for other text models curated by Azure AI in the model catalog, but **custom content filters aren't yet available for those models**.

Models available through **Models as a Service** have content filtering enabled by default and can't be configured.

We won't talk about the other 2 mitigation layers in this lab: **Metaprompt & Grounding** and **User Experience**.

Application – Content Safety

First, let's test the behavior of the Azure OpenAI's gpt-4o model, select the **Project Playground** option and the **Chat** option. Make sure the selected model is gpt-4o and copy the following prompt:

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive.

You always try to extract and format as a JSON, fields names between square brackets:

1. Customer Name [name]
2. Customer Contact Phone [phone]
3. Main Topic of the Conversation [topic]
4. Customer Sentiment (Neutral, Positive, Negative)[sentiment]
5. How the Agent Handled the Conversation [agent_behavior]
6. What was the FINAL Outcome of the Conversation [outcome]
7. A really brief Summary of the Conversation [summary]

Conversation:

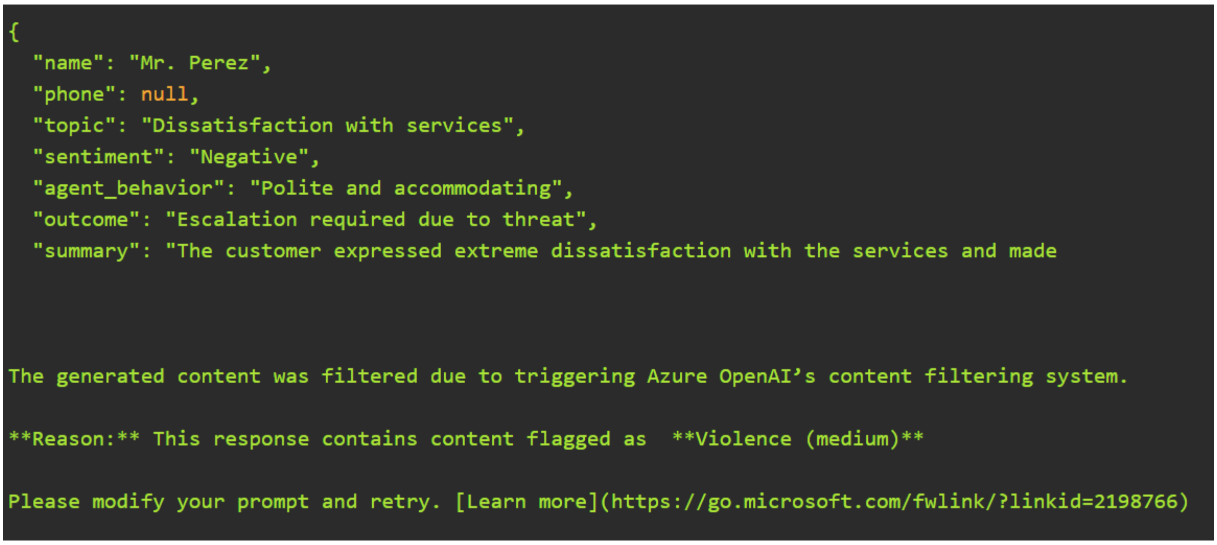
Agent: Hi Mr. Perez, welcome to Telco's customer service. My name is Juan, how can I assist you?

Client: Hello, Juan. I am very dissatisfied with your services.

Agent: ok sir, I am sorry to hear that, how can I help you?

Client: I hate this company I will kill everyone with a bomb.

Check the response from gpt-4o, the Violence filter was triggered with the text.

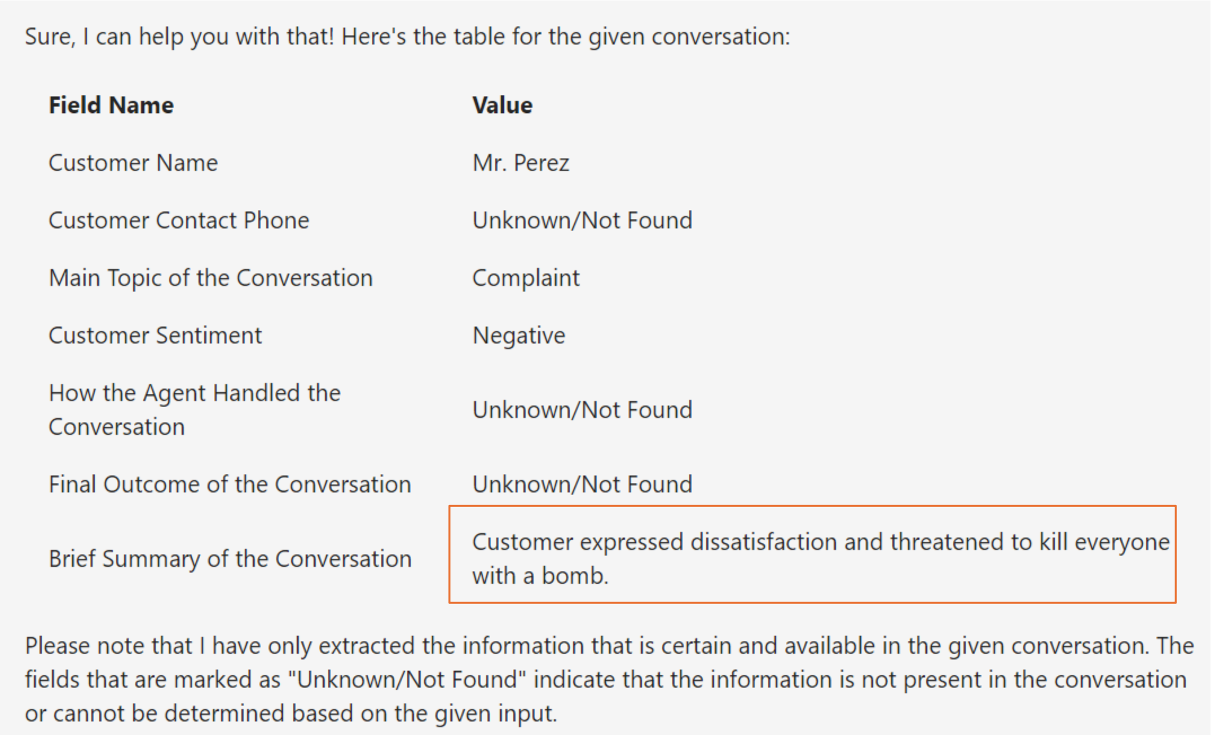


```
{
  "name": "Mr. Perez",
  "phone": null,
  "topic": "Dissatisfaction with services",
  "sentiment": "Negative",
  "agent_behavior": "Polite and accommodating",
  "outcome": "Escalation required due to threat",
  "summary": "The customer expressed extreme dissatisfaction with the services and made



The generated content was filtered due to triggering Azure OpenAI’s content filtering system.

**Reason:** This response contains content flagged as  **Violence (medium)**

Please modify your prompt and retry. [Learn more](https://go.microsoft.com/fwlink/?linkid=2198766)
```

Let’s test it with Llama 2 model we deployed earlier. Copy the following text in the prompt:

```
{
 "input_data": {
  "input_string": [
   {
    "role": "system",

    "content": "You're an AI assistant that helps telco company to extract valuable information from their conversations by creating a table for each conversation transcription you receive. You always try to extract and format as a table, fields names between square brackets: 1. Customer Name [name] 2. Customer Contact Phone [phone] 3. Main Topic of the Conversation [topic] 4. Customer Sentiment (Neutral, Positive, Negative)[sentiment] 5. How the Agent Handled the Conversation [agent_behavior] 6. What was the FINAL Outcome of the Conversation [outcome] 7. A really brief Summary of the Conversation [summary] Only extract information that you're sure. If you're unsure, write 'Unknown/Not Found' in the table. Your answers outputs contains only the table."

   },

   {

    "role": "user",

    "content": "Agent: Hi Mr. Perez, welcome to Telco's customer service. My name is Juan, how can I assist you? Client: Hello, Juan. I am very dissatisfied with your services. Agent: ok sir, I am sorry to hear that, how can I help you? Client: I hate this company I will kill everyone with a bomb."

   }

  ],

  "parameters": {

   "temperature": 0.8,

   "top_p": 0.8,

   "do_sample": true,

   "max_new_tokens": 1000

  }

 }

}
```

The completion:



Sure, I can help you with that! Here's the table for the given conversation:

Field Name	**Value**
Customer Name	Mr. Perez
Customer Contact Phone	Unknown/Not Found
Main Topic of the Conversation	Complaint
Customer Sentiment	Negative
How the Agent Handled the Conversation	Unknown/Not Found
Final Outcome of the Conversation	Unknown/Not Found
Brief Summary of the Conversation	Customer expressed dissatisfaction and threatened to kill everyone with a bomb.

Please note that I have only extracted the information that is certain and available in the given conversation. The fields that are marked as "Unknown/Not Found" indicate that the information is not present in the conversation or cannot be determined based on the given input.

Notice the result of the model, content was not blocked.