



# Visual Search RAG System with GenAIOps

A comprehensive **GenAIOps (LLMOps)** solution for an e-commerce Visual Search system. This project allows users to upload an image of a product and find similar items in a catalog using a "Caption-then-Embed" strategy powered by GPT-4o and Azure AI Search.

It features a robust **Governance & Guardrails** layer and a unified **Monitoring** system.



## Key Features

- **Multimodal Search:** Uses GPT-4o to generate detailed visual descriptions of images, enabling semantic search using text embeddings.
  - **Vector Store Agnostic:** Switch between **Azure AI Search** (Production) and **ChromaDB** (Local/Dev) via configuration.
  - **Governance & Guardrails:**
    - **Safety Validator:** Detects prompt injection, jailbreaks, and unsafe content (using Azure Content Safety).
    - **Compliance Checker:** Detects and redacts PII (GDPR/HIPAA compliance).
    - **Hallucination Detection:** Checks responses for uncertain language.
- 



## Prerequisites

- **Python 3.10+**
  - **Azure Subscription** with:
    - **Azure OpenAI Service** (Deployments: gpt-4o, text-embedding-3-small)
    - **Azure AI Search** (Optional if using ChromaDB)
    - **Azure Content Safety** (Optional but recommended)
    - **Azure Monitor** (Optional)
  - **Docker** (for containerization)
-

# Setup & Installation

## 1. Clone the Repository (optional if you're not using GitHub)

None

```
git clone <your-repo-url>
cd week_14_llmops
```

## 2. Create a Virtual Environment

None

```
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate
```

## 3. Install Dependencies

None

```
pip install -r requirements.txt
```

## 4. Configure Environment

- Copy the sample configuration file:

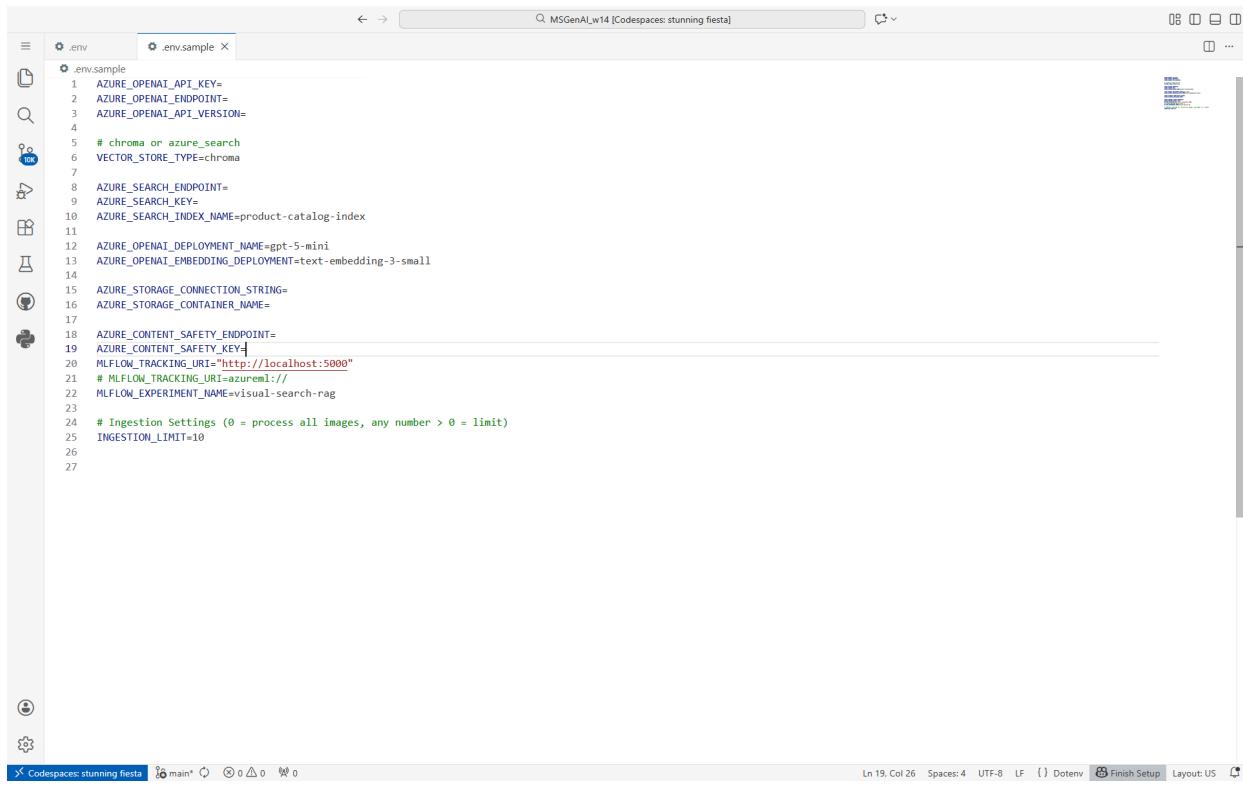
None

```
cp .env.sample .env
```

- Open .env and configure:

- VECTOR\_STORE\_TYPE: Set to chroma (local) or azure\_search (cloud).
- AZURE\_OPENAI\_\*: Add your keys and endpoints.
- MLFLOW\_TRACKING\_URI: Set tracking URI (default http://localhost:5000 for local).

**NOTE:** We recommend **gpt-3.5-turbo-0301** and **text-embedding-ada-002** models for LLM and Embedding model deployments respectively.



```
AZURE_OPENAI_API_KEY=
AZURE_OPENAI_ENDPOINT=
AZURE_OPENAI_API_VERSION=
# chroma or azure_search
VECTOR_STORE_TYPE=chroma
AZURE_SEARCH_ENDPOINT=
AZURE_SEARCH_KEY=
AZURE_SEARCH_INDEX_NAME=product-catalog-index
AZURE_OPENAI_DEPLOYMENT_NAME=pt-5-mini
AZURE_OPENAI_EMBEDDING_DEPLOYMENT=text-embedding-3-small
AZURE_STORAGE_CONNECTION_STRING=
AZURE_STORAGE_CONTAINER_NAME=
AZURE_CONTENT_SAFETY_ENDPOINT=
AZURE_CONTENT_SAFETY_KEY|
MLFLOW_TRACKING_URI="http://localhost:5000"
# MLFLOW_TRACKING_URI=azureml://
MLFLOW_EXPERIMENT_NAME=visual-search-rag
# Ingestion Settings (0 = process all images, any number > 0 = limit)
INGESTION_LIMIT=10
```



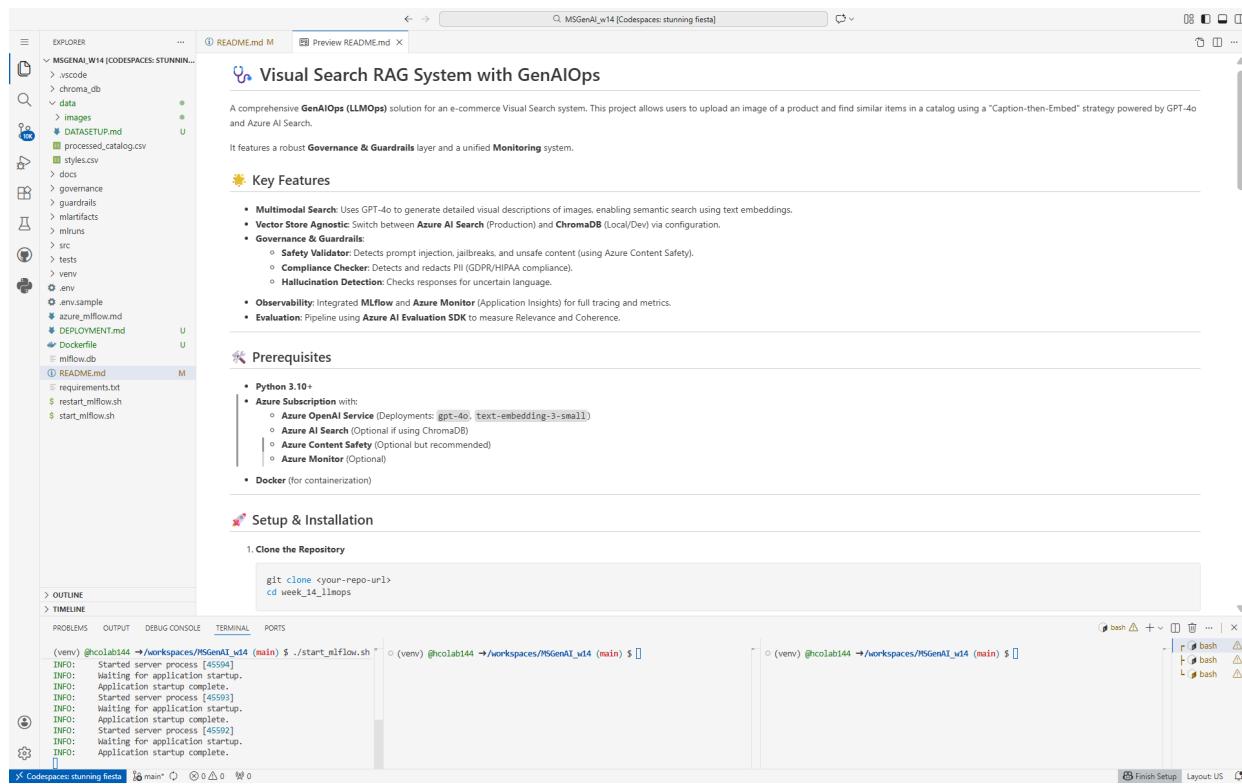
## Usage Guide - The LLMOps Pipeline

### 1. Start Monitoring (MLflow)

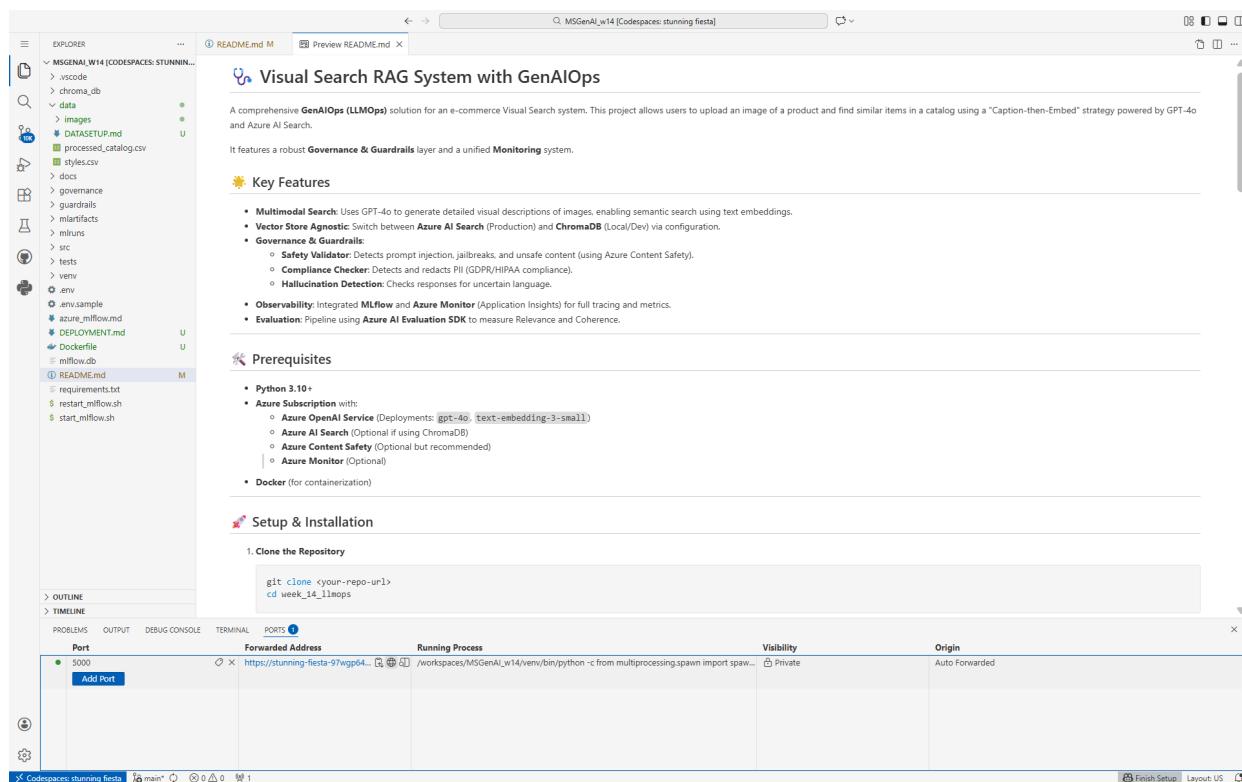
Start the local MLflow server to track all experiments and traces.

None

`./start_mlflow.sh`



- The dashboard can be viewed at <http://localhost:5000>



The MLFlow Dashboard opens in a new tab.

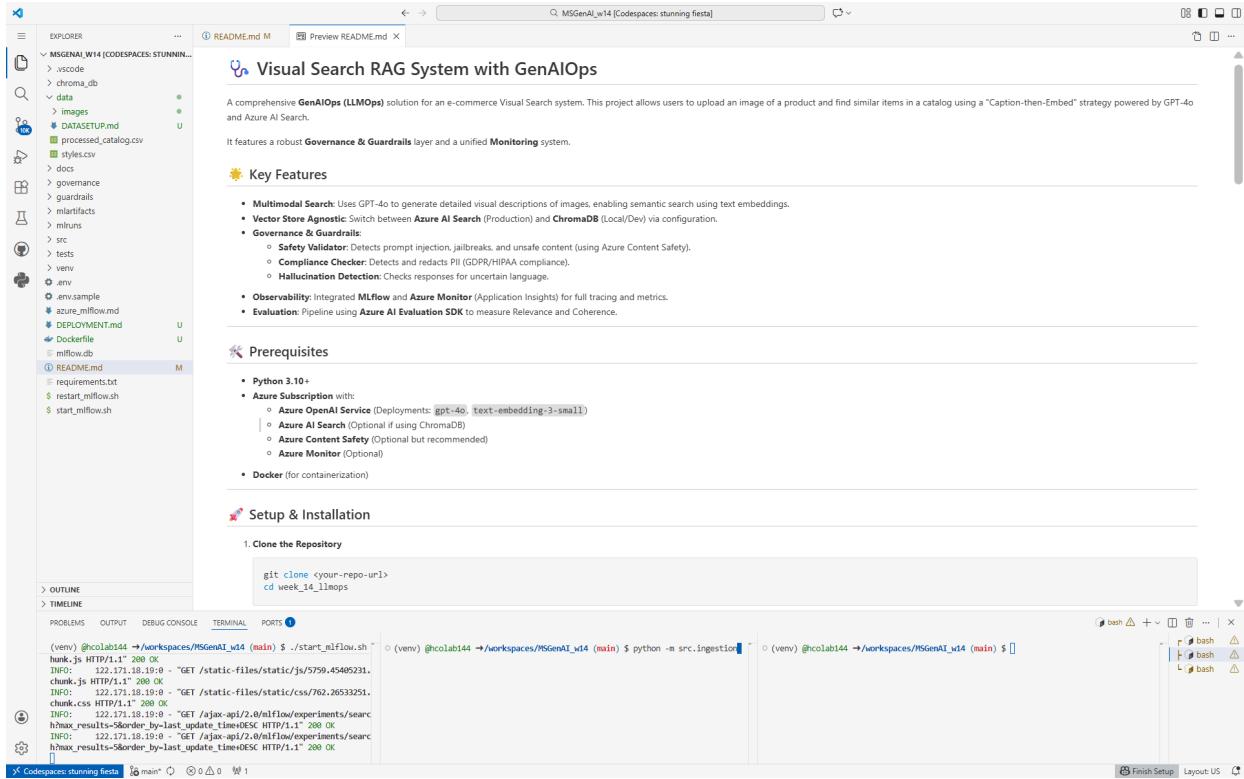
The screenshot shows the MLflow 3.5.0 dashboard. The top navigation bar includes the MLflow logo, version 3.5.0, a gear icon, GitHub, and Docs links. On the left, a sidebar has a '+ New' button and links for Home, Experiments, Models, and Prompts. The main content area starts with a 'Welcome to MLflow' header and a 'Get started' section with four cards: 'Log traces' (Trace LLM applications for debugging and monitoring), 'Run evaluation' (Iterate on quality with offline evaluations and comparisons), 'Train models' (Track experiments, parameters, and metrics throughout training), and 'Register prompts' (Manage prompt updates and collaborate across teams). Below this is an 'Experiments' section with a table showing one experiment named 'Default' created on 02/03/2026 at 05:21:06 PM. The final section is 'Discover new features' with three cards: 'MLflow MCP server' (Connect your coding assistants and AI applications to MLflow and automatically analyze your experiments and traces), 'Optimize prompts' (Access the state-of-the-art prompt optimization algorithms such as MIPROv2, GEPA, through MLflow Prompt Registry), and 'Agents-as-a-judge' (Leverage agents as a judge to perform deep trace analysis and improve your evaluation accuracy). A 'View all' link is located in the top right of the experiments and discoveries sections.

## 2. Data Ingestion

Populate the vector index (Chroma or Azure) with the product catalog.

None

```
# Ingest data into Vector Store
python -m src.ingestion
```



**NOTE:** The ingestion process typically takes a few minutes depending on the number of images. In this demo, we've used 10 images by defining the environment variable in .env file

None

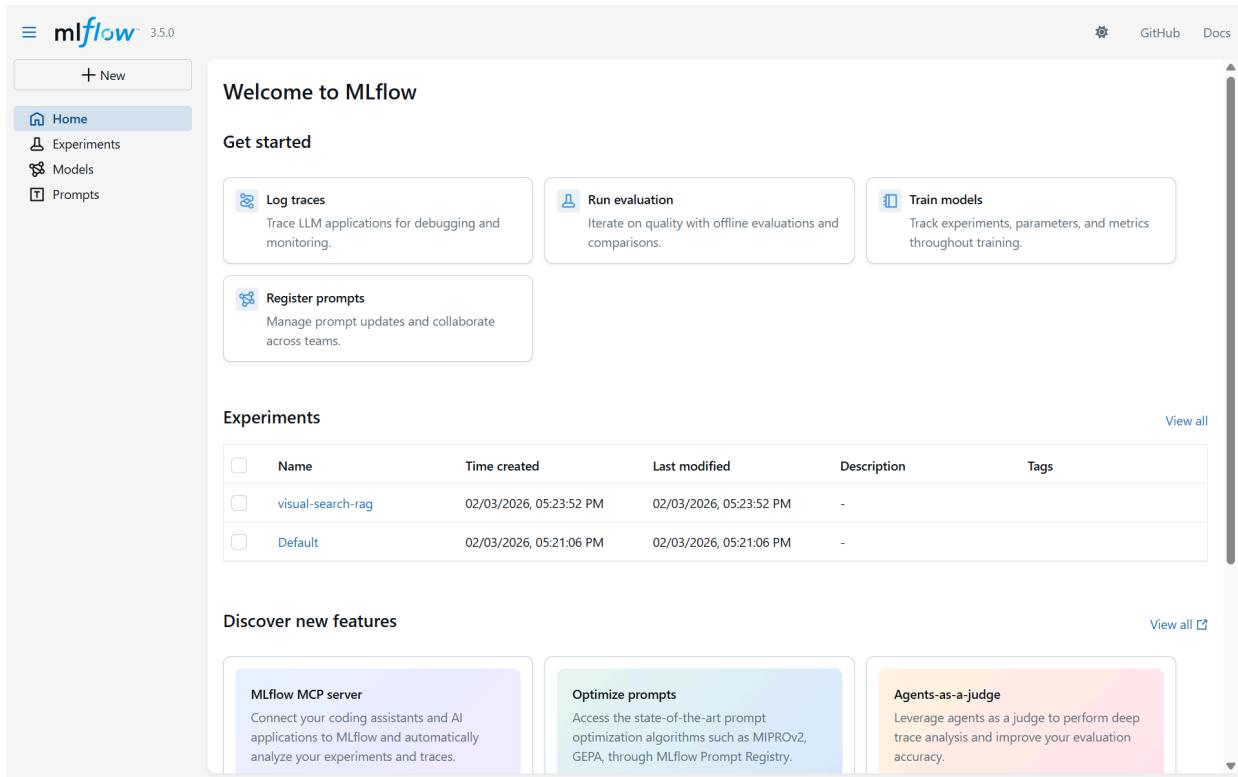
# Ingestion Settings (0 = process all images, any number > 0 = limit)

INGESTION\_LIMIT=10

The screenshot shows the Microsoft Visual Studio Code interface. The Explorer sidebar on the left displays a file tree for a workspace named 'MSGGenAI\_w14'. The terminal at the bottom shows command-line output related to MLflow experiments. The activity bar at the bottom right contains icons for Python, bash, and other tools.

```
(venv) @hcolab144 ~>/workspaces/MSGGenAI_w14 (main) $ python -m src.ingestion
Initialized ChromDB (LangChain) at /workspaces/MSGGenAI_w14/chroma_db
Forcing 20876 products. Starting ingestion...
Processing first product (via DEPLOYMENT_LIMET env variable).
2025/02/03 11:53:52 INFO mlflow.tracking.fluent: Experiment with name 'visual-search-rag' does not exist. Creating a new experiment.
[0]
| 0/10 [00:00<-, ?it/s]
└ View run generate_description at: http://localhost:5000/#/experiments/536852411889588039/runs/5a074660b244d83e540247637a00a78
└ View experiment at: http://localhost:5000/#/experiments/536852411889588039
```

You can track the experiments from the MLflow dashboard.



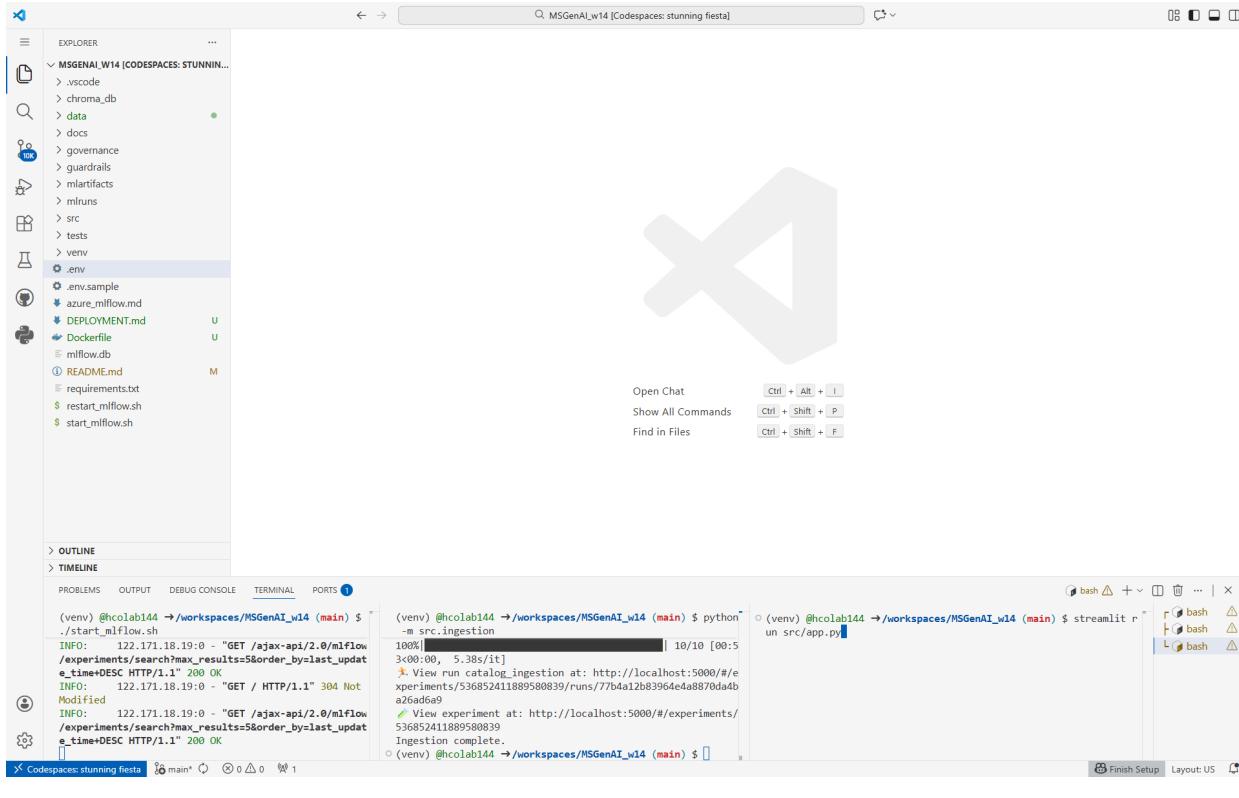
### 3. Running the Application

Launch the Streamlit interface. This handles the **RAG flow** with **Governance** checks.

None

```
streamlit run src/app.py
```

- **Governance in Action:** Try uploading an unsafe image or simulating a prompt injection to see the GovernanceGate block it.
- **Monitoring:** Check MLflow traces for every search request.



# Integrating MLflow with Azure Machine Learning (AML)

## A Comprehensive Step-by-Step Guide

This guide explains how to configure your Python applications to use Azure Machine Learning as a centralized backend for **MLflow Tracking**. This setup allows you to manage experiment logs, metrics, models, and GenAI traces in a unified, cloud-based environment.

---

## 1. Prerequisites

Before starting, ensure you have the following Azure resources provisioned:

- **Azure Machine Learning Workspace:** The central hub for your ML operations.
- **Azure Resource Group:** The container holding your AML workspace.

- **Python Environment:** Python 3.8+ is recommended.

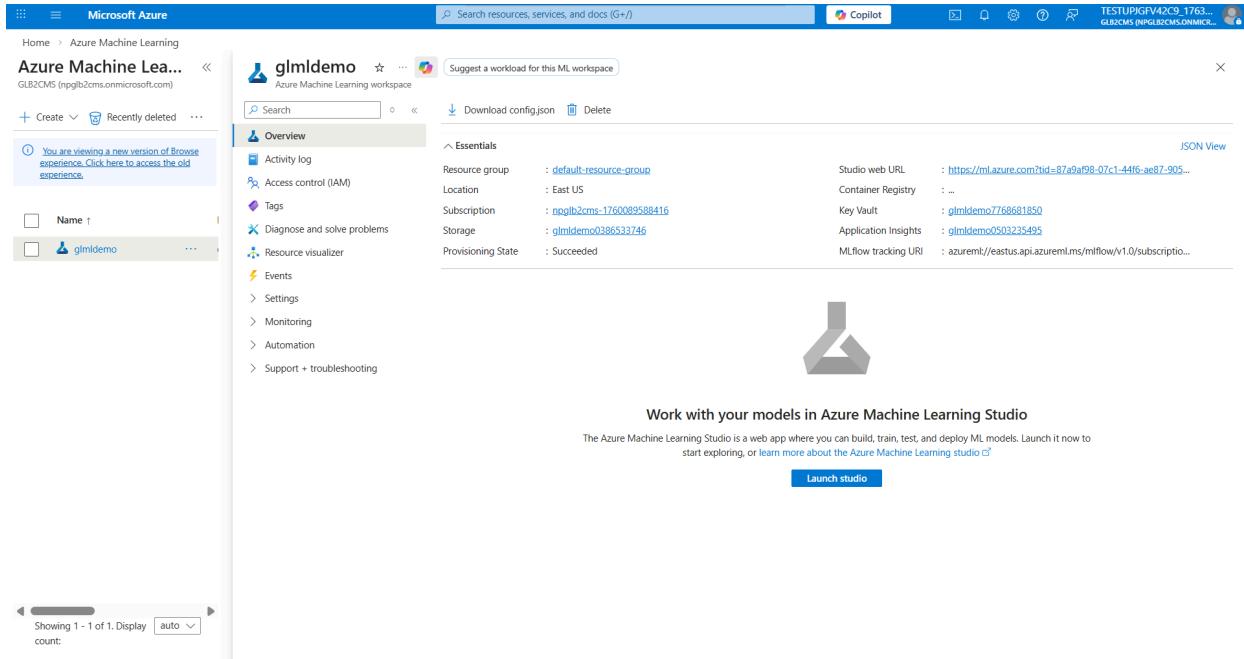
## 2. Retrieve the MLflow Tracking URI

To point MLflow to Azure, you must obtain the workspace-specific Tracking URI.

### Option A: Via Azure Portal (UI)

1. Navigate to the Azure Portal.
2. Open your **Azure Machine Learning Workspace**.
3. On the **Overview** page, locate the **MLflow tracking URI** in the top-right section (you may need to click "JSON View" or "See more" if it is hidden).
4. Copy the URI. It follows this format:

`azureml://<region>.api.azureml.ms/mlflow/v1.0/subscriptions/<sub-id>/resourceGroups/<rg-name>/providers/Microsoft.MachineLearningServices/workspaces/<workspace-name>`



The screenshot shows the Azure Machine Learning workspace overview page for 'glmldemo'. The 'MLflow tracking URI' is displayed as a JSON key-value pair under the 'Essentials' section:

	:	Value
MLflow tracking URI	:	<code>azureml://eastus.api.azureml.ms/mlflow/v1.0/subscriptions/0503235495/providers/Microsoft.MachineLearningServices/workspaces/glmldemo</code>

## Environment Configuration

Your application uses the MLFLOW\_TRACKING\_URI environment variable to determine where to send data.

## For Local Development

Update your .env file or export the variable in your terminal to set tracking URI to localhost.

For production deployment (when deployed using Azure App Services), use the [Azure ML Tracking URI](#).

```
None  
# Local tracking (disabled)  
# MLFLOW_TRACKING_URI="http://localhost:5000"  
  
# Azure ML Tracking  
MLFLOW_TRACKING_URI="azureml://eastus.api.azureml.ms/mlflow/v1.0/..."
```

---

## Implementation and Verification

Update the .env file with the MLflow tracking URI

```
# MLFLOW_TRACKING_URI="http://localhost:5000"  
MLFLOW_TRACKING_URI=azureml://eastus.api.azureml.ms/mlflow/v1.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resourc  
MLFLOW_EXPERIMENT_NAME=visual-search-rag  
  
# Ingestion Settings (0 = process all images, any number > 0 = limit)  
INGESTION_LIMIT=10
```

---

## Authentication and Security

Azure Machine Learning requires authentication. By using azure-identity, you can avoid hardcoding credentials.

## For Local Development

Simply authenticate via the Azure CLI. The library will use your developer identity:

None

`az login`

## For Production Application (Passwordless Login)

For production deployment, we'll use Managed Identity to ensure secure, credential-free communication happens between the RAG application and Azure Machine Learning Workspace. We'll define roles that will allow for the communication to happen between these Azure services.

**NOTE:** This section assumes that you've created **Azure App Service**. You may revisit this section when deployment to Azure App Service is complete.

- Enable Identity:
  - Go to **App Service > Settings > Identity**.

The screenshot shows the Azure portal interface for managing an app service. The left sidebar lists various settings like Deployment, Environment variables, Configuration, Instances, Authentication, and Identity. The 'Identity' section is currently selected. On the right, under the 'System assigned' tab, there is a 'Status' switch that is currently set to 'Off'. A detailed description below the switch explains that a system assigned managed identity is restricted to one per resource and is tied to the lifecycle of the resource. It also mentions that you can grant permissions to the managed identity by using Azure role-based access control (Azure RBAC) and that the managed identity is authenticated with Microsoft Entra ID, so you don't have to store any credentials in code. At the bottom of the page, there are buttons for 'Save', 'Discard', 'Refresh', 'Troubleshoot', and 'Got feedback?'

- Set Status to **On** (System assigned) and click **Save**.

The screenshot shows the Azure portal interface for managing a Web App named 'visualragapp'. The left sidebar navigation includes Home, Microsoft Defender for Cloud, Events (preview), Log stream, Resource visualizer, Deployment (Deployment slots, Deployment Center), Settings (Environment variables, Configuration (preview), Instances, Authentication, Identity), Backups, Custom domains, Certificates, Networking, Webjobs, Service Connector, Locks, Performance (Load Testing), App Service plan (App Service plan, Scale up, Scale out), Development Tools, API, Monitoring, Automation, and Support + troubleshooting. The 'Identity' section is currently selected. The main content area displays the 'System assigned' tab, which is highlighted in blue. It shows the status is 'On' and the object (principal) ID is '592d0a042-dca0-4ea8-94bc-ebe990a12ad1'. A note at the bottom states: 'This resource is registered with Microsoft Entra ID. The managed identity can be configured to allow access to other resources. Be careful when making changes to the access settings for the managed identity because it can result in failures.' There are Save, Discard, Refresh, Troubleshoot, and Got feedback? buttons at the top of the form.

- **Grant Permissions:**

- **Navigate to your Azure Machine Learning Workspace.**
  - **Go to Access control (IAM) > Add > Add role assignment.**

You are viewing a new version of Browse experience. Click here to access the old experience.

Name: glmldemo

Check access | Role assignments | Roles | Deny assignments | Classic administrators

**My access**

View my level of access to this resource.

**Check access**

Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

**Check access**

**Grant access to this resource**

Grant access to resources by assigning a role. [Learn more](#)

**Add role assignment**

**View access to this resource**

View the role assignments that grant access to this and other resources. [Learn more](#)

**View**

**View deny assignments**

View the role assignments that have been denied access to specific actions at this scope. [Learn more](#)

**View**

- Role: **AzureML Data Scientist** (allows logging and reading experiments).

Role Members Conditions Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Copilot can help pick a role

Job function roles Privileged administrator roles

Grant access to Azure resources based on job function, such as the ability to create virtual machines.

Name	Description	Type	Category	Details
AzureML Data Scientist	Can perform all actions within an Azure Machine Learning workspace, except for creating or deleting compute resources and modifying th...	BuiltInRole	None	<a href="#">View</a>

Showing 1 - 1 of 1 results.

Review + assign | Previous | Next | Feedback

- Assign access to: Managed Identity.

The screenshot shows the 'Add role assignment' interface in Microsoft Azure. The 'Members' tab is active. The 'Selected role' is 'AzureML Data Scientist'. The 'Assign access to' section has 'Managed identity' selected. The 'Members' table is empty. A 'Description' field is present. At the bottom, there are navigation buttons: 'Review + assign', 'Previous', 'Next', and 'Feedback'.

- Member: Select your specific **App Service instance**.

Microsoft Azure

Home > Azure Machine Learning > glimdemo | Access control (IAM)

Add role assignment ...

Role Members Conditions Review + assign

Selected role AzureML Data Scientist

Assign access to  User, group, or service principal  Managed identity

Members [+ Select members](#)

Name	Object ID	Type
No members selected		

Description

Review + assign Previous Next

Select managed identities

Some results might be hidden due to your ABAC condition.

Subscription \* npglb2cms-1760089588416

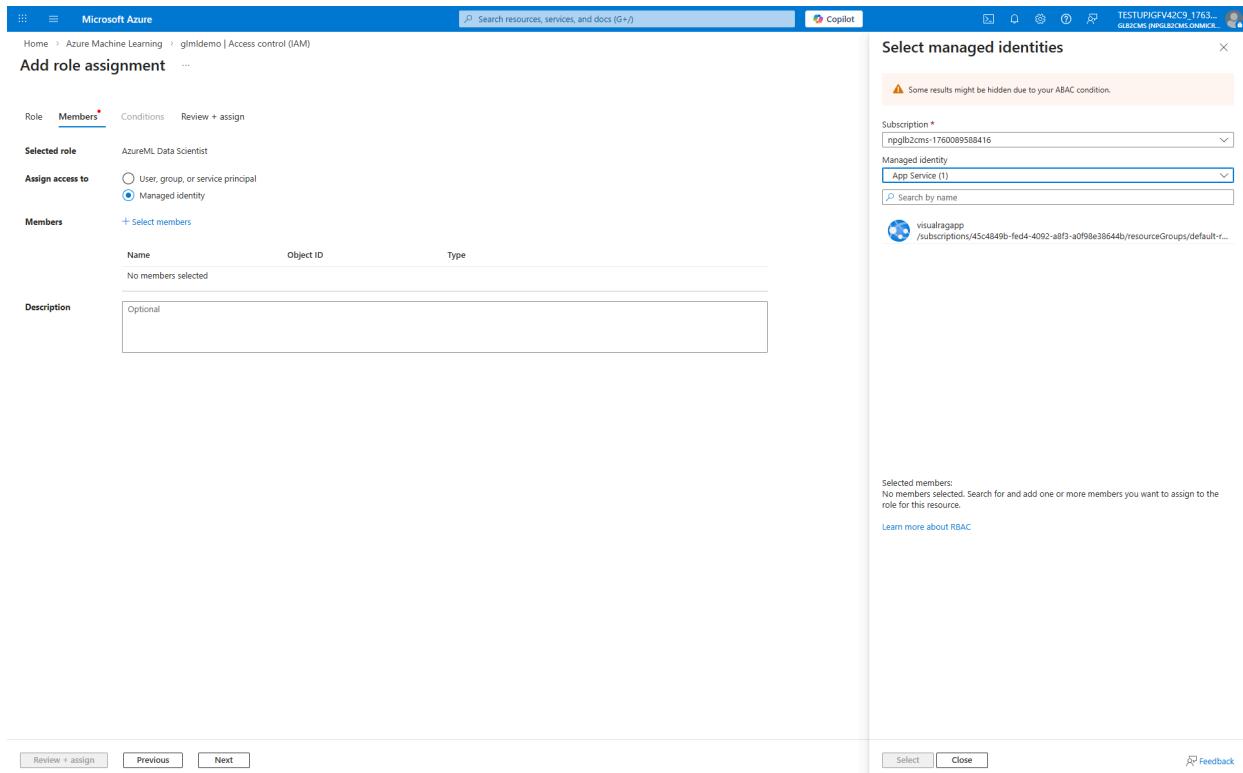
Managed identity App Service (1)

Search by name

Selected members:  
No members selected. Search for and add one or more members you want to assign to the role for this resource.

Learn more about RBAC

Select Close Feedback



Microsoft Azure

Home > Azure Machine Learning > glimdemo | Access control (IAM)

Add role assignment ...

Role Members Conditions Review + assign

Selected role AzureML Data Scientist

Assign access to  User, group, or service principal  Managed identity

Members [+ Select members](#)

Name	Object ID	Type
No members selected		

Description

Review + assign Previous Next

Select managed identities

Some results might be hidden due to your ABAC condition.

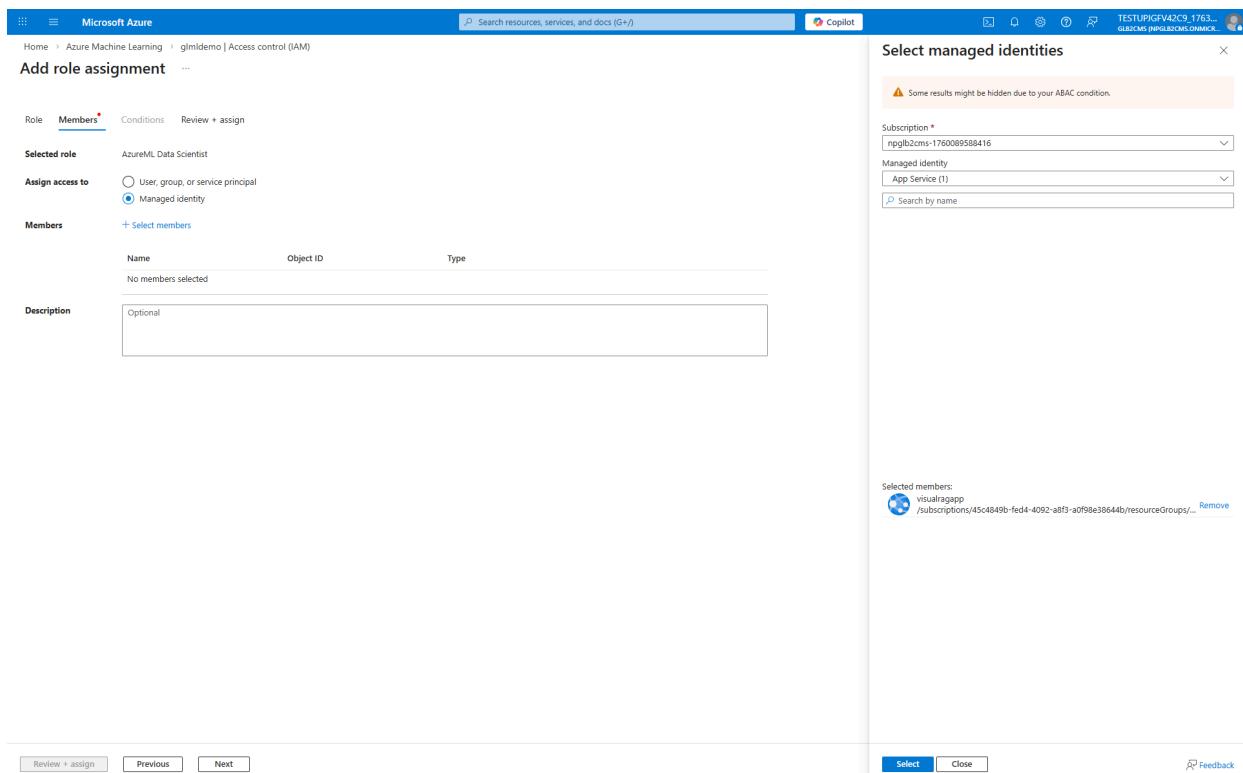
Subscription \* npglb2cms-1760089588416

Managed identity App Service (1)

Search by name

Selected members:  
 visualragapp /subscriptions/45c4849b-fed4-4092-a8f3-a0f90e30644b/resourceGroups/... Remove

Select Close Feedback



## Click on Next

The screenshot shows the 'Add role assignment' interface in the Microsoft Azure portal. The 'Members' tab is active. A single member, 'visualragapp', is listed under the 'Members' section. The 'Selected role' is set to 'AzureML Data Scientist'. The 'Assign access to' field is set to 'Managed identity'. The 'Description' field is optional and empty. At the bottom, there are navigation buttons for 'Previous' and 'Next'.

Microsoft Azure

Home > Azure Machine Learning > glimldemo | Access control (IAM)

Add role assignment ...

Role Members Conditions Review + assign

Selected role AzureML Data Scientist

Assign access to  User, group, or service principal  Managed identity

Members [+ Select members](#)

Name	Object ID	Type
visualragapp	532da042-dca0-4ea8-94bc-e1fe990a12a...	App Service

Description

Previous Next

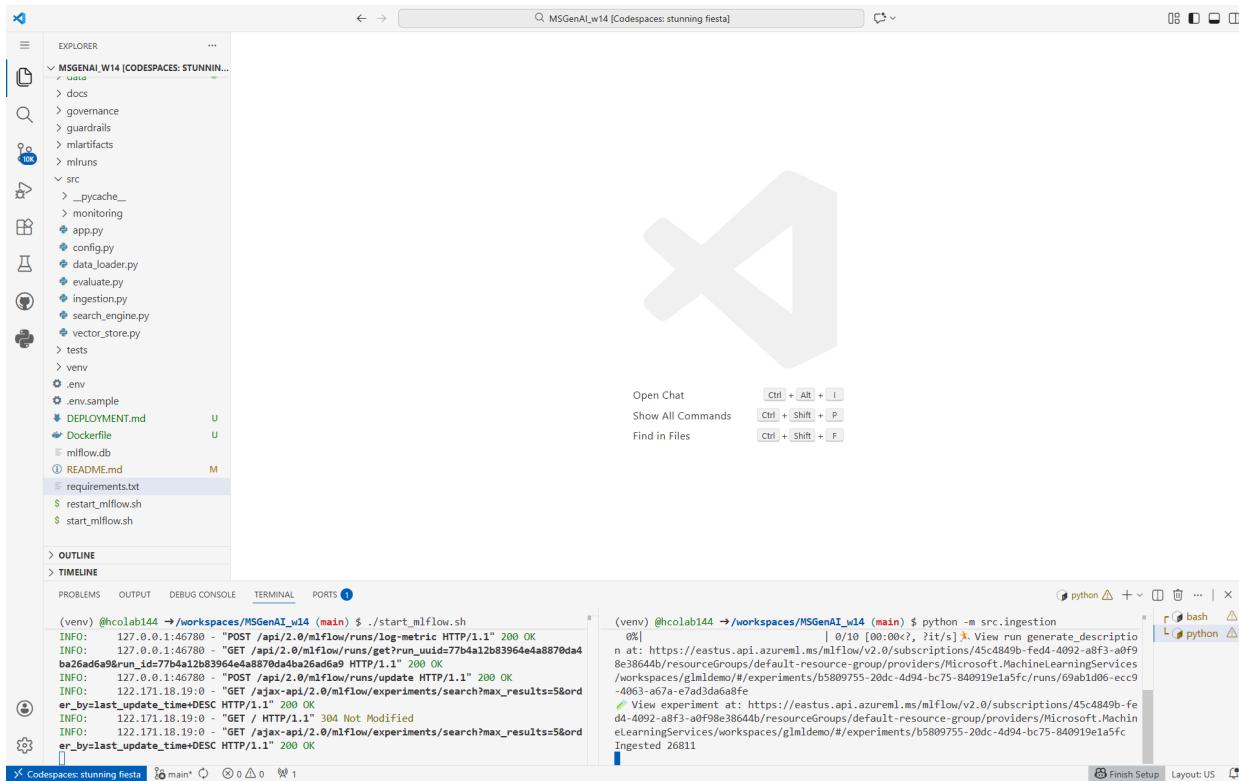
https://portal.azure.com/#

## Click on Review + assign

The screenshot shows the Microsoft Azure 'Add role assignment' interface. At the top, there's a navigation bar with 'Microsoft Azure', a search bar, and various icons. Below it, the path 'Home > Azure Machine Learning > glmlidemo | Access control (IAM)' is visible. The main title is 'Add role assignment ...'. A horizontal navigation bar below the title includes 'Role', 'Members', 'Conditions', and 'Review + assign' (which is highlighted in blue). The 'Role' section shows 'AzureML Data Scientist'. The 'Scope' section shows the subscription ID. The 'Members' section contains a table with one row: 'visualdragapp' (Name), '532da042-dca0-4ea8-94bc-efe990a12ad1' (Object ID), and 'App Service' (Type). The 'Description' section is empty. At the bottom, there are buttons for 'Review + assign' (highlighted in blue), 'Previous', 'Next', and 'Feedback'.

## Tracing via Azure ML Workspace

Run your script: python src/[ingestion.py](#).



## Open Azure Machine Learning Studio.

The screenshot shows the Azure Machine Learning workspace 'glmldemo'. The left sidebar has a 'Jobs' tab selected. The main area displays the workspace overview with a summary card showing a green status icon, 0 datasets, 0 models, and 0 pipelines. Below this, there's a section titled 'Work with your models in Azure Machine Learning Studio' with a 'Launch studio' button.

**Overview**

**Essentials**

	:	
Resource group	:	default-resource-group
Location	:	East US
Subscription	:	ngl2cms-1760089588416
Storage	:	glmldemo0386533746
Provisioning State	:	Succeeded

**Activity log**

**Tags**

**Diagnose and solve problems**

**Resource visualizer**

**Events**

**Settings**

**Monitoring**

**Automation**

**Support + troubleshooting**

**Work with your models in Azure Machine Learning Studio**

The Azure Machine Learning Studio is a web app where you can build, train, test, and deploy ML models. Launch it now to start exploring, or learn more about the Azure Machine Learning studio [\[?\]](#)

**Launch studio**

Navigate to the **Jobs** tab in the left sidebar

Microsoft Foundry | Azure Machine Learning

GLB2CMS > glimdemo > Jobs

## Jobs

All experiments All jobs All schedules

Refresh Archive experiment Reset view View archived experiments

Experiment	Latest job	Last submitted	Created
visual-search-rag	catalog_ingestion	Feb 3, 2026 5:33 PM	Feb 3, 2026 4:59 PM

Columns

https://ml.azure.com/runs?wsid=/subscriptions/45c4849b-fed4-4092-a8f3-a...

< Prev Next > 20/Page

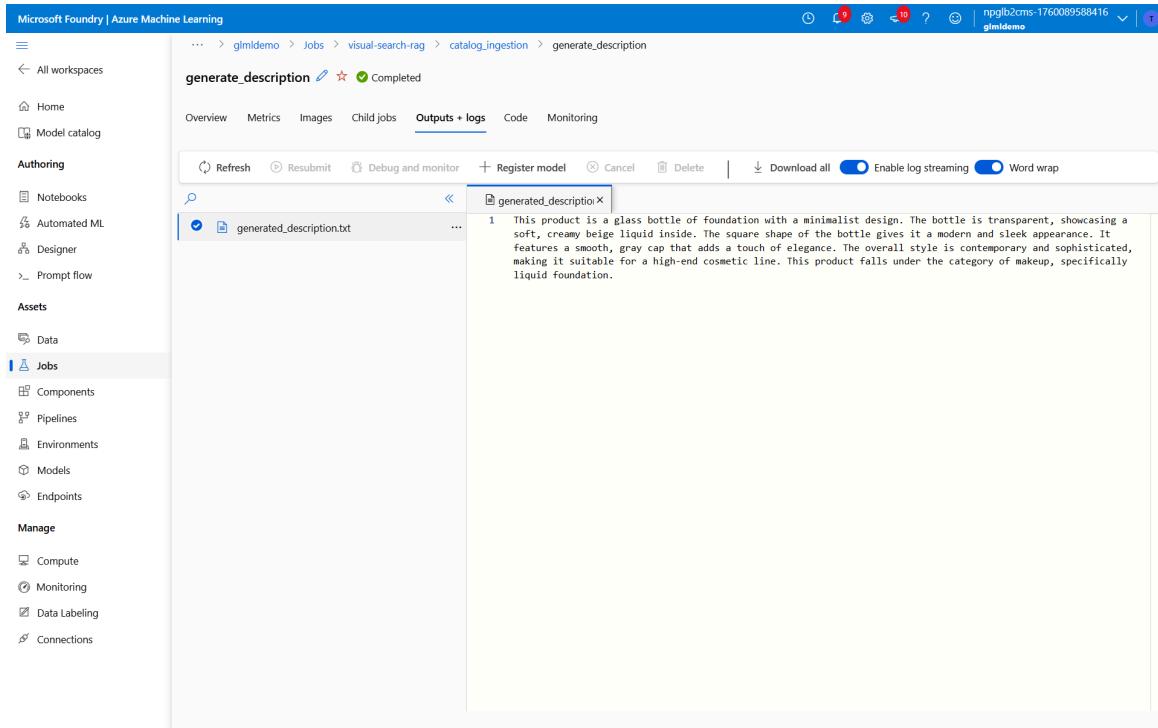
Select your experiment (e.g., visual-search-rag).

The screenshot shows the Microsoft Foundry Azure Machine Learning interface. The left sidebar navigation includes sections for Notebooks, Automated ML, Designer, Prompt flow, Components, Pipelines, Environments, Models, Endpoints, Compute, Monitoring, Data Labeling, and Connections. The main content area displays a list of jobs for the experiment 'visual-search-rag'. The list includes columns for Display name (I visualized), Parent job name, Status, Created on, Duration, Created by, and Compute target. There are 10 entries, all of which are completed. The URL at the bottom of the page is: https://ml.azure.com/experiments/d/b5809755-20dc-4d94-bc75-840919e1a5fc/runs/d12e2885-0fa9-49ed-893f-9aa08104c519?wsid=/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourcegroups/default-resource-group/...

You will see your runs, metrics, and traces listed there.

The screenshot shows the details of a specific job named 'generate\_description'. The left sidebar is identical to the previous screenshot. The main content area shows the job details under the 'Overview' tab. The 'Properties' section includes fields for Status (Completed), Created on (Feb 3, 2026 5:34 PM), Start time (Feb 3, 2026 5:34 PM), Duration (10.01s), Compute duration (10.01s), Compute target (None), and Name (4d56b4df-577a-4873-a6a8-57bc95b69b43). The 'Tags' section contains two tags: 'mlflow.parentRunId: d12e2885-0fa9-49ed-893f-9aa08104c519' and 'mlflow.runName: generate\_description'. The 'Params' section lists 'model: gpt-4o' and 'task: image\_captioning'. The 'Metrics' section indicates 'No data'. The 'Description' section has a placeholder 'Click edit icon to add a description'. The URL at the bottom of the page is: https://ml.azure.com/experiments/d/b5809755-20dc-4d94-bc75-840919e1a5fc/runs/d12e2885-0fa9-49ed-893f-9aa08104c519?wsid=/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourcegroups/default-resource-group/...

The Outputs + logs tab provides the Inputs - Output pair for each LLM Chain.



# Streamlit Application

To run the streamlit application, run the following command

None

```
streamlit run src/app.py
```

The screenshot shows a VS Code interface with a large X logo in the center. The Explorer sidebar on the left lists files and folders, including `src`, `app.py`, `config.py`, and `streamlit run src/app.py`. The terminal at the bottom shows command-line logs related to MLflow experiments.

```
(venv) @colab144 ~/workspaces/MSGenAI_v14 (main) $ .. /start_mlflow.sh
INFO: 127.0.0.1:46780 - "POST /api/2.0/mlflow/runs/get?run_uuid=77b4a12b83964ea870da4ba26add9a9&run_id=774a12b83964ea8870da25ad6a9 HTTP/1.1" 200 OK
INFO: 127.0.0.1:46780 - "GET /api/2.0/mlflow/runs/get?run_uuid=77b4a12b83964ea870da4ba26add9a9&run_id=774a12b83964ea8870da25ad6a9 HTTP/1.1" 200 OK
INFO: 127.0.0.1:46780 - "POST /api/2.0/mlflow/runs/update HTTP/1.1" 200 OK
INFO: 122.171.18.19:0 - "GET /ajax-api/2.0/mlflow/experiments/search?max_results=5&order_by=last_update_time+DESC HTTP/1.1" 200 OK
INFO: 122.171.18.19:0 - "GET /HTTP/1.1" 304 Not Modified
INFO: 122.171.18.19:0 - "GET /ajax-api/2.0/mlflow/experiments/search?max_results=5&order_by=last_update_time+DESC HTTP/1.1" 200 OK
```

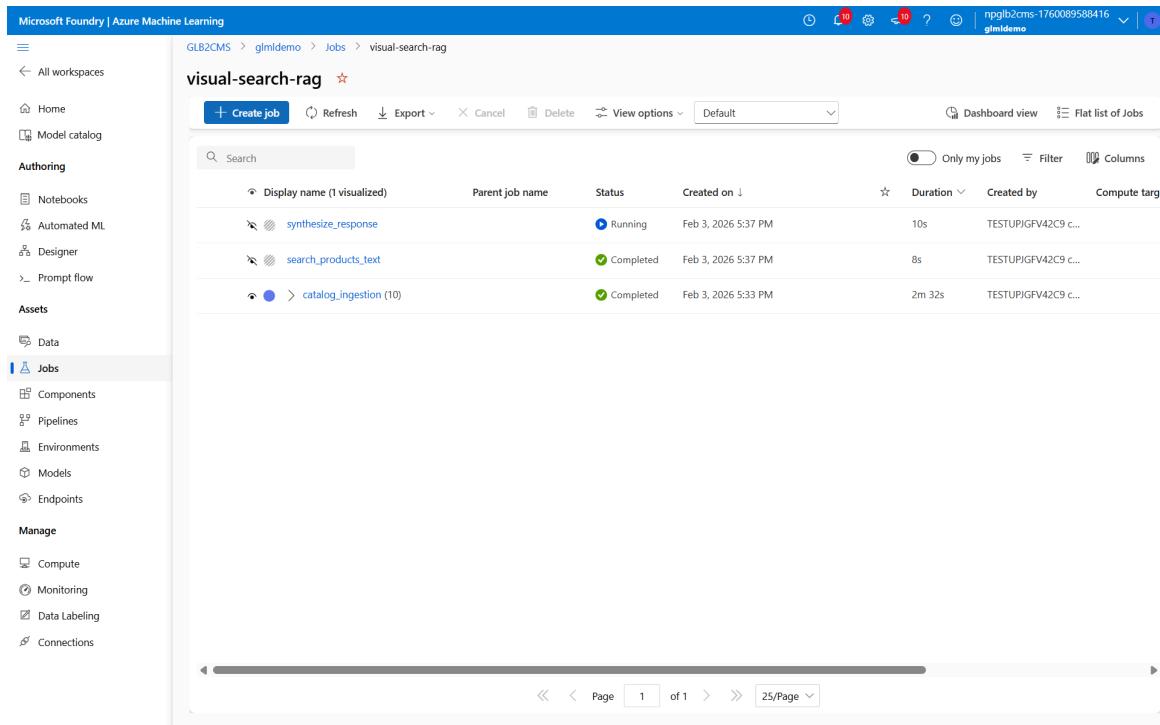
**The streamlit application shows the results for the user query. “Blue casual shirt”**

The screenshot shows a search interface with a search bar containing "Blue casual shirt". Below the search bar, it says "Search completed in 10.21s" and "Found 5 matches". The results are listed under "Top Matches", showing a dark gray short-sleeve shirt with a button-down collar. Product details include:

- 1. 12189
- Product Description: Men's Short-Sleeve Shirt
- Color: Dark gray with subtle variations in tone.
- Pattern: Solid with a smooth texture, featuring a classic, understated look.
- Material: Lightweight cotton blend, providing breathability and comfort.
- Style: Casual short-sleeve design with a button-down collar, ideal for relaxed or semi-formal occasions.
- Details: Includes two front chest pockets with flaps, adding functionality and style.
- Fit: Tailored fit, designed to provide a modern silhouette without being too tight.
- Category: Men's apparel, specifically categorized as a casual shirt.

This shirt is versatile and can be paired with jeans or chinos for a polished yet casual appearance.

## The tracking server keeps track of the LLM calls



The screenshot shows the Microsoft Foundry Azure Machine Learning interface. The left sidebar navigation includes: All workspaces, Home, Model catalog, Authoring (Notebooks, Automated ML, Designer, Prompt flow), Assets (Data), and Jobs (selected). The main content area displays a list of jobs under the project 'visual-search-rag'. The table has columns: Display name (1 visualized), Parent job name, Status, Created on, Duration, Created by, and Compute targ. Three rows are listed:

Display name (1 visualized)	Parent job name	Status	Created on	Duration	Created by	Compute targ
synthesize_response		Running	Feb 3, 2026 5:37 PM	10s	TESTUPJGFV42C9 c...	
search_products_text		Completed	Feb 3, 2026 5:37 PM	8s	TESTUPJGFV42C9 c...	
catalog_ingestion (10)		Completed	Feb 3, 2026 5:33 PM	2m 32s	TESTUPJGFV42C9 c...	

Pagination at the bottom indicates 1 of 1 page.

**Note:** There is occasionally a 1–2 minute propagation delay for logs and GenAI traces to appear in the Azure ML Studio UI compared to a local MLflow dashboard.



## Governance & Guardrails

The system uses a layered safety approach located in governance/ and guardrails/.

- **Governance Gate** (governance.governance\_gate): Orchestrates checks.
- **Safety Validator**: Blocks harmful content.
- **Compliance Checker**: Redacts PII.

See Governance README for details.



## Monitoring

Unified monitoring via monitoring/.

- **Logger:** Structured JSON logging.
- **Metrics:** Token usage, latency, cost.
- **Traces:** Distributed tracing via OpenTelemetry.

See Monitoring README for details.

---

## Build Docker Image

Select the **Docker Image** and click on **Build Image**

```
FROM python:3.10-slim-buster
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
# Expose port for Streamlit
EXPOSE 8501
# Command to run the application
CMD ["streamlit", "run", "src/app.py", "--server.port=8501", "--server.address=0.0.0.0"]
```

Enter the Image Tag and click Enter.

MSGENAI\_W14 [CODESPACES: STUNNING...]

```

Dockerfile U x visual_rag:latest
FROM python:3.10-slim-buster
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
# Expose port for Streamlit
EXPOSE 8501
# Command to run the application
CMD ["streamlit", "run", "src/app.py", "--server.port=8501", "--server.address=0.0.0.0"]

```

DEPLOYMENT.md

Dockerfile

README.md

requirements.txt

restart\_mlflow.sh

start\_mlflow.sh

OUTLINE

TIMELINE

Codespaces: stunning fiesta

Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Dockerfile Finish Setup Layout: US

MSGENAI\_W14 [CODESPACES: STUNNING...]

```

Dockerfile U x
FROM python:3.10-slim-buster
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
# Expose port for Streamlit
EXPOSE 8501
# Command to run the application
CMD ["streamlit", "run", "src/app.py", "--server.port=8501", "--server.address=0.0.0.0"]

```

DEPLOYMENT.md

Dockerfile

README.md

requirements.txt

restart\_mlflow.sh

start\_mlflow.sh

OUTLINE

TIMELINE

TERMINAL

```

Executing task: docker build --pull --rm -f Dockerfile -t 'visual_rag:latest' .
[1/5] FROM docker.io/library/python:3.10-slim-buster@sha256:37aa74c2d001f99b14828450d003c55f821c90f225fdfdd80c5180fcc77b3f
=> sha256:0903bceec9721c4ee87e188b263a0a392a1b2e2813c7a0ed0f9c4416194734 2.10MB / 11.50MB
=> sha256:37aa274cd001f90b14828450d003c55f821c90f225fdfdd80c5180fcfa77b3f 9888 / 9888
=> sha256:66f714f81d6522d6ad2156abee2535307c2be24a96781f4823c60422dfffa3c2 1.37KB / 1.37KB
=> extracting sha256:80b1b88d57765c08c602668755a3f6dc437b6ce15a17e4857139efc964f3
=> sha256:66f714f81d6522d6ad2156abee2535307c2be24a96781f4823c60422dfffa3c2 1.37KB / 1.37KB
=> sha256:b2697057e25fe078e5f17bd522b6a013fc3c2a041d34d2e4771e0f3dd3c3f 242B / 242B
=> [internal] load build context
=> transferring context: 5.94MB

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

bash

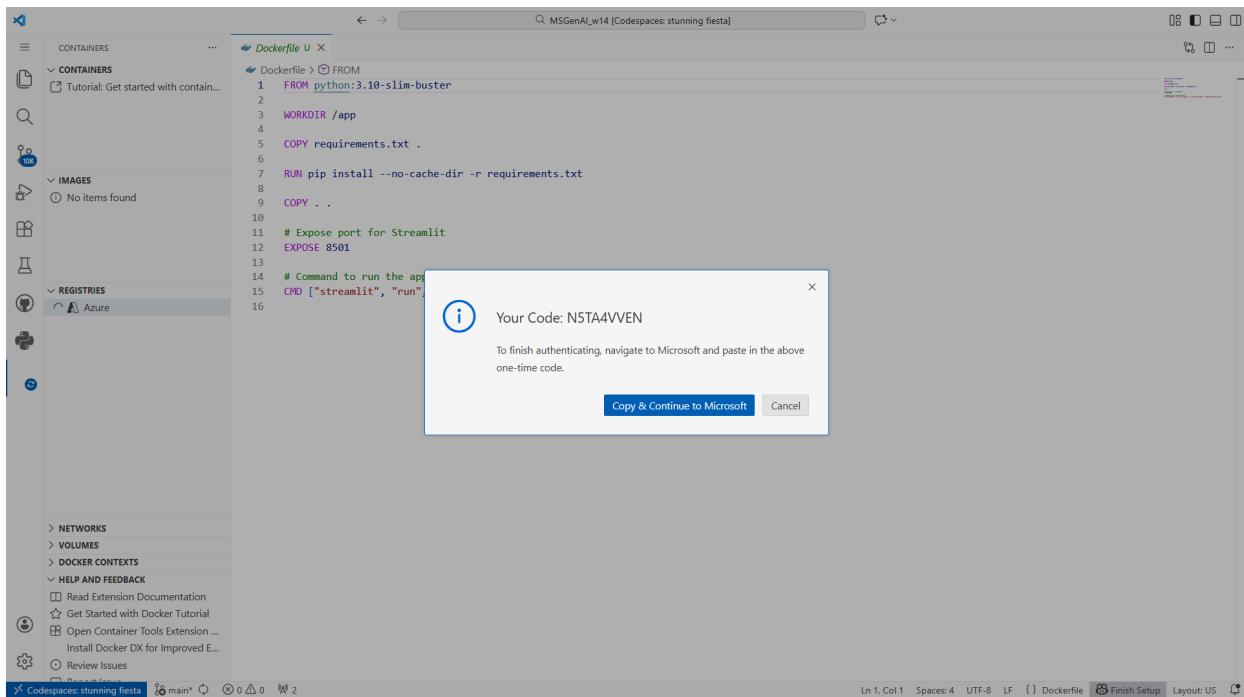
python

Dockerfile

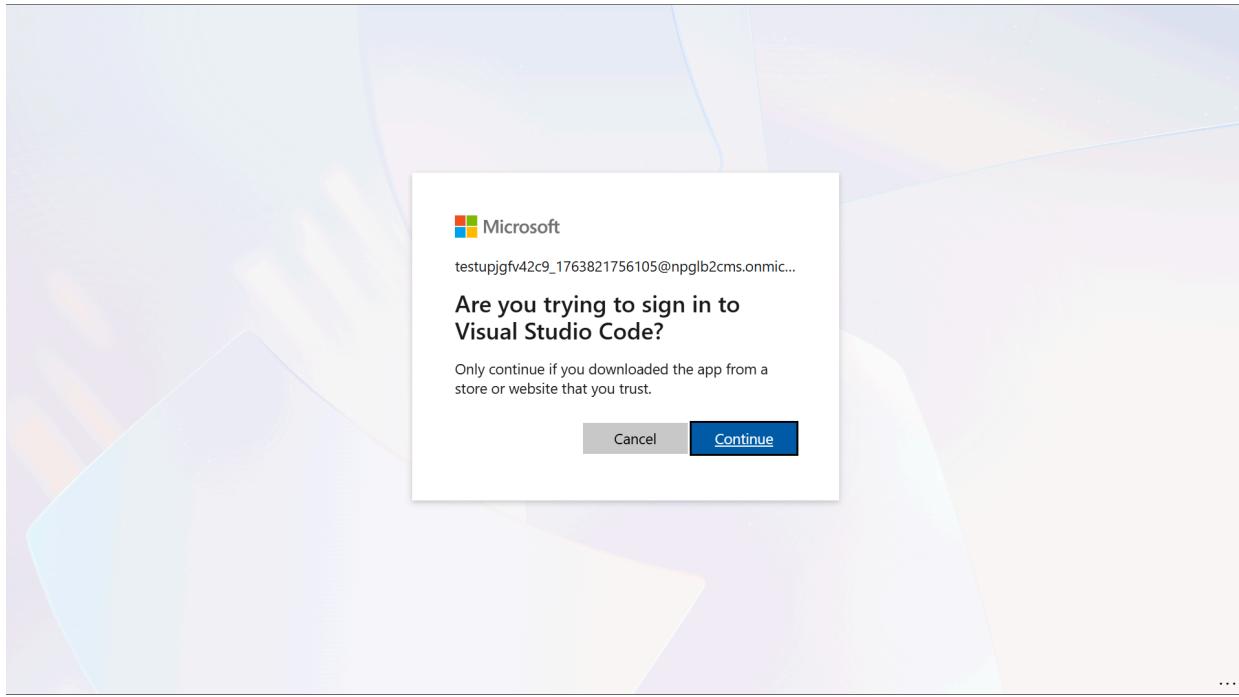
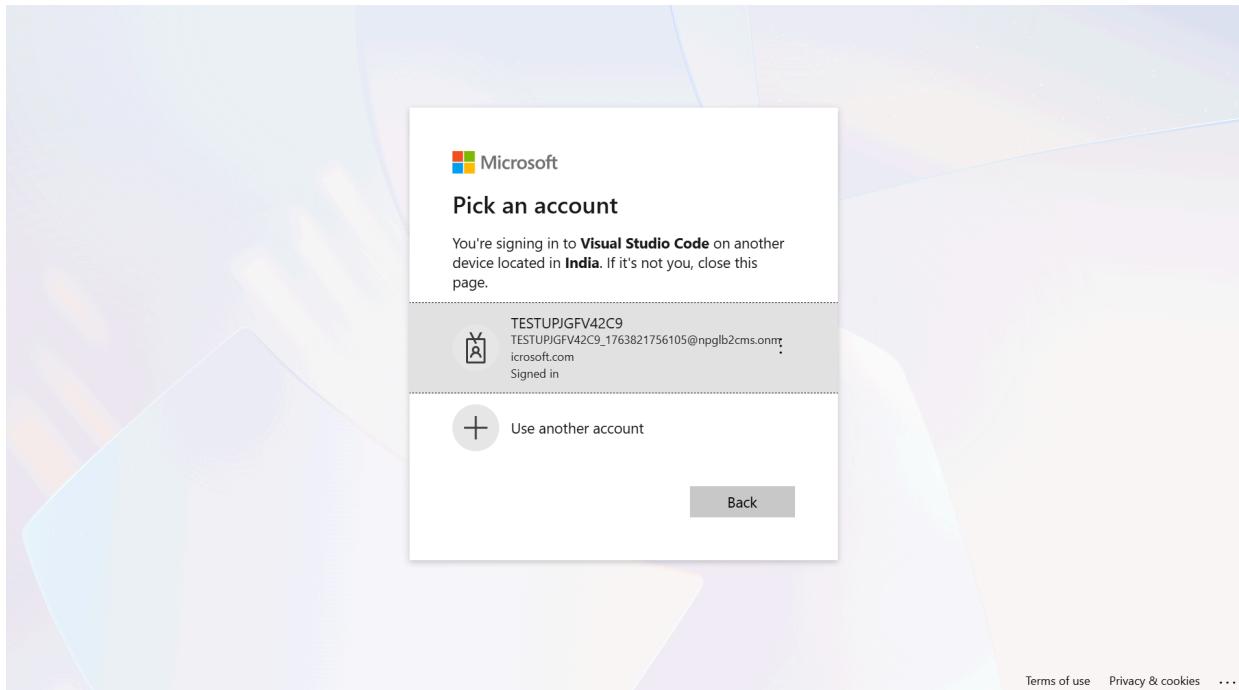
Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Dockerfile Finish Setup Layout: US

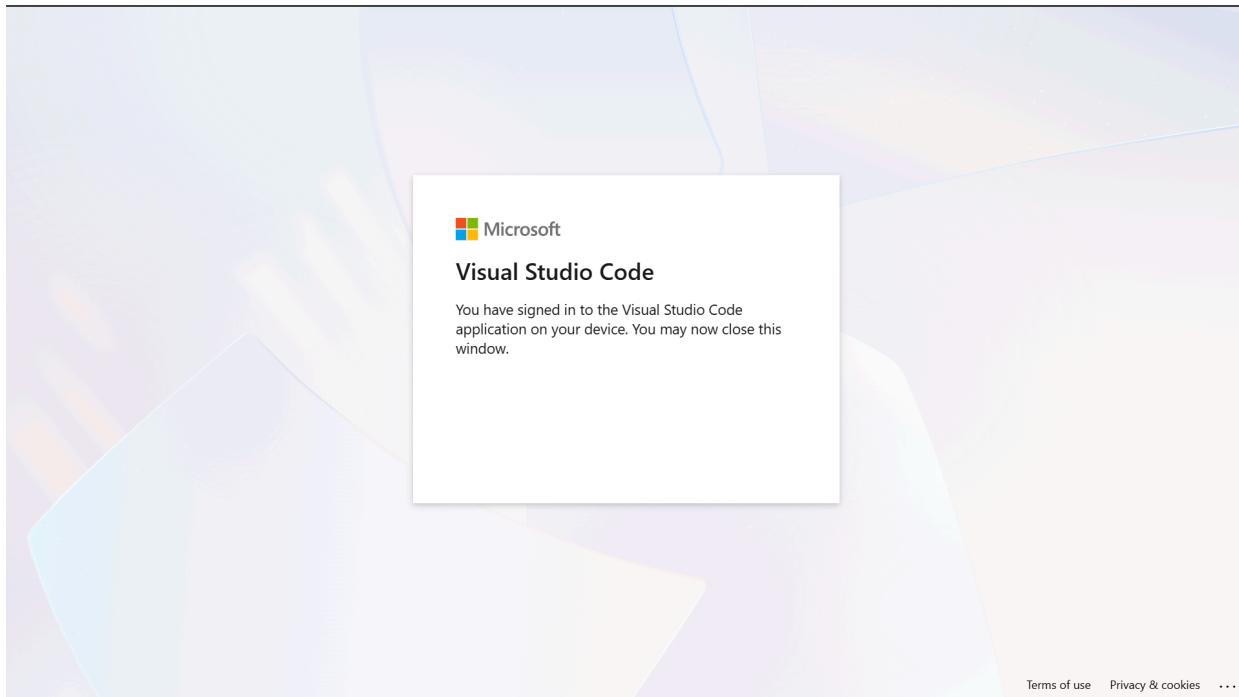
Once the Docker image is built, navigate to the **Container Tools** extension in VS Code and click on **Azure**.

This will prompt you to login using your **Azure account**.



Login via your Azure account.

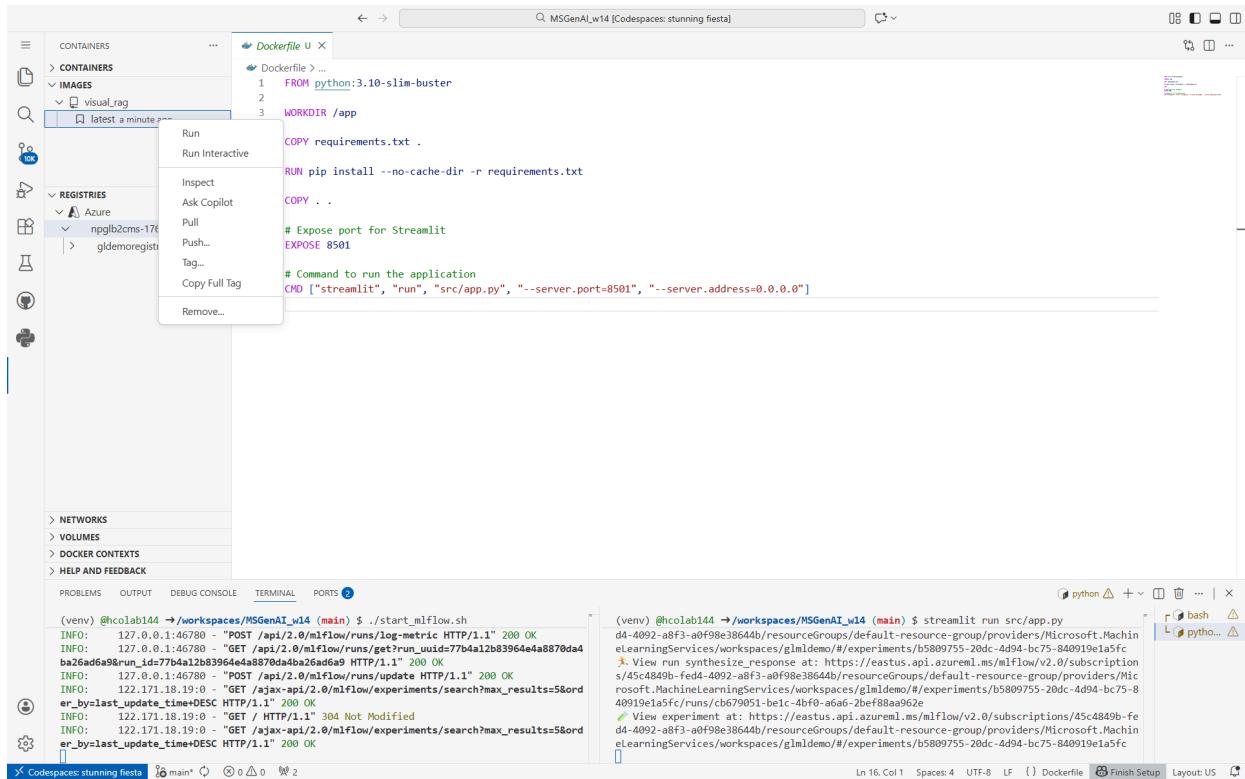


A screenshot of the Azure Container Registry Dockerfile editor in Visual Studio Code. The interface shows a sidebar with "CONTAINERS", "IMAGES", "REGISTRIES", and other navigation options. The main area displays a Dockerfile with the following content:

```
FROM python:3.10-slim-buster
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt
COPY . .
# Expose port for Streamlit
EXPOSE 8501
# Command to run the application
CMD ["streamlit", "run", "src/app.py", "--server.port=8501", "--server.address=0.0.0.0"]
```

The terminal tab at the bottom shows command-line output related to the Dockerfile and Streamlit.

Click on the Docker Image and click on Push



**NOTE:** The above steps assume that you've already created the Azure Container Registry. The steps for creating a registry can be found in Week 9 and Week 13 MLS sections

## Azure App Service

**NOTE:** Once the Docker image has been pushed to the ACR, the next step is to deploy the Docker image using a service such as **Azure App Services**.

You may refer to the MLS content of Week 9, Week 13 for the step-by-step instructions on how to create an Azure App Service and deploy the application using Docker Image via Azure Container Registry (ACR)

1. Navigate to your **App Service** in the Azure Portal.
2. Go to **Settings > Configuration** (or **Environment variables** in newer interfaces).
3. Click **+ Add** to create a new setting.

For instance to add MLFLOW tracking URI, click Add and set the following:

- **Name:** MLFLOW\_TRACKING\_URI
- **Value:** <Your Azure ML Tracking URI>

4. Click **Save**.

Enter all the environment variables required for the application and click **Apply**

**NOTE:** Make sure the variable **VECTOR\_STORE\_TYPE** is **azure\_search**

Name	Value	Deployment slot setting	Source	Delete
APPLICATIONINSIGHTS_CONNECTION_STRING	InstrumentationKey=d00f9f97-9765-4933-a9a-b14ad8e04569;ingestionEndpoint=https://swedencentral-0.in.applicationinsights.azure.com/		App Service	
ApplicationInsightsAgent_EXTENSION_VERSION			App Service	
AZURE_CONTENT_SAFETY_ENDPOINT	https://gl-content-safety15.cognitiveservices.azure.com/		App Service	
AZURE_CONTENT_SAFETY_KEY			App Service	
AZURE_OPENAI_APIKEY			App Service	
AZURE_OPENAI_APIVERSION	2023-01-01-preview		App Service	
AZURE_OPENAI_DEPLOYMENT_NAME	gpt-4o-mini		App Service	
AZURE_OPENAI_EMBEDDING_DEPLOYMENT	text-embedding-3-small		App Service	
AZURE_OPENAI_ENDPOINT	https://hubproject1847643180.openai.azure.com/		App Service	
AZURE_SEARCH_ENDPOINT	https://rasearchfree16.search.windows.net		App Service	
AZURE_SEARCH_INDEX_NAME	product-catalog-index		App Service	
AZURE_SEARCH_KEY	0112QH9tbOZ53LwsUhY2fSmJ3GBbY5WRVSjWRgYQAzSeCTAIDV		App Service	
AZURE_STORAGE_CONNECTION_STRING	DefaultEndpointsProtocol=https;AccountName=ragdocs15;AccountKey=2bRuJPIMsgVWVs+rPqhgsv7SCz/tICp3F0ogreAK		App Service	
AZURE_STORAGE_CONTAINER_NAME	ragdocs15		App Service	
INGESTION_LIMIT	10		App Service	
MLFLOW_EXPERIMENT_NAME	visual-search-rag		App Service	
MLFLOW_TRACKING_URI	azurerm://eastus.azureml.ms/mlflow/v1.0/subscriptions/45c4849b-fed4-4092-abf3-a0f98e38644b/resourceGroups/default		App Service	
VECTOR_STORE_TYPE	azure_search		App Service	

Save the changes and click **Confirm**.

Click on the URL defined in the Default domain.

## NOTE:

For production purposes, we will ingest the images using Azure AI search.

Update the .env file to **azure\_search**

```
None  
.env file  
# Vector Store Selection  
# Options: "chroma" (Local) or "azure_search" (Production)  
VECTOR_STORE_TYPE=azure_search
```

Run the following command to populate the search index in Azure AI Search before searching in the deployed application.

```
None  
python -m src.ingestion
```

```
(venv) → /workspaces/MSGenAI_w14 (main) $ python -m src.ingestion  
Starting ingestion process...  
Loading catalog from data/processed_catalog.csv  
Initialized Azure AI Search (LangChain) for index 'product-catalog-index'  
Found 20876 products. Starting ingestion...  
Processing first 10 items (set via INGESTION_LIMIT env variable).  
 0%|██████████| 0/10 [00:00<?, ?it/s] ⏳ View run generate_description at: https://eastus.api.azureml.ms/mlflow/v2.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resource-group/providers/Microsoft.MachineLearningServices/workspaces/g1mldemo/#/experiments/b5809755-20dc-4d94-bc75-840919e1a5fc/runs/64ab9236-20ba-468c-91db-0466985ade7e  
astus.api.azureml.ms/mlflow/v2.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resource-group/providers/Microsoft.MachineLearningServices/workspaces/g1mldemo/#/experiments/b5809755-20dc-4d94-bc75-840919e1a5fc  
Ingested 26811  
10%|██████████| 1/10 [00:18<02:42, 18.08s/it] ⏳ View run generate_description at: https://eastus.api.azureml.ms/mlflow/v2.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resource-group/providers/Microsoft.MachineLearningServices/workspaces/g1mldemo/#/experiments/b5809755-20dc-4d94-bc75-840919e1a5fc/runs/3db66fb7-e7cd-4c31-b4e9-bdc9a7664d70  
astus.api.azureml.ms/mlflow/v2.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resource-group/providers/Microsoft.MachineLearningServices/workspaces/g1mldemo/#/experiments/b5809755-20dc-4d94-bc75-840919e1a5fc  
Ingested 12189  
20%|██████████| 2/10 [00:32<02:08, 16.03s/it] ⏳ View run generate_description at: https://eastus.api.azureml.ms/mlflow/v2.0/subscriptions/45c4849b-fed4-4092-a8f3-a0f98e38644b/resourceGroups/default-resource-group/providers/Microsoft.MachineLearningServices/workspaces/g1mldemo/#/experiments/b5809755-20dc-4d94-bc75-840919e1a5fc/runs/338e344a-c171-4437-b4a1-9155fa6cb544
```

In the deployed application, enter the search query and press **Search**.

The search results are displayed in the sections below.

Visual Product Search

Upload an image to find similar products in the catalog.

Search Products

Enter a description or upload an image to find similar products.

blue pants

Search

Drag and drop file here  
Limit 200MB per file - JPG, PNG, JPEG

Browse files

Search completed in 8.40s

Found 5 matches.

See Query Understanding

Top Matches

1. 26811

Product Description: Men's Dress Pants

- Color: The pants feature a muted olive green hue, offering a versatile and sophisticated look suitable for various occasions.
- Material: Crafted from a lightweight, breathable fabric blend, these pants provide comfort and ease of movement, making them ideal for both formal and casual settings.
- Pattern: The fabric is solid without any visible patterns, ensuring a clean and polished appearance.
- Style: Designed in a classic straight-leg cut, these pants offer a tailored fit that flatters the silhouette while allowing for comfortable wear.
- Category: Suitable for business casual or formal attire, these dress pants can be paired with a button-up shirt or blazer for a complete look.

Overall, these pants combine style and functionality, making them a staple addition to any wardrobe.

> Metadata

The tracking server in ML Studio automatically populates the **jobs** and the **corresponding results**.

The screenshot shows the Microsoft Foundry Azure Machine Learning interface. On the left, there's a sidebar with navigation links like 'All workspaces', 'Home', 'Model catalog', 'Authoring' (Notebooks, Automated ML, Designer, Prompt flow), 'Assets' (Data, Components, Pipelines, Environments, Models, Endpoints), 'Jobs' (selected), 'Manage' (Compute, Monitoring, Data Labeling, Connections), and 'Compute' (Compute, Monitoring, Data Labeling, Connections). The main area is titled 'visual-search-rag' and shows a list of completed jobs. The columns include 'Display name (4 visualized)', 'Parent job name', 'Status', 'Created on', 'Duration', 'Created by', and 'Compute targ'. The jobs listed are: 'synthesize\_response' (Completed, Feb 3, 2026 6:28 PM, 7s, Service Principal), 'search\_products\_text' (Completed, Feb 3, 2026 6:28 PM, 3s, Service Principal), 'catalog\_ingestion (10)' (Completed, Feb 3, 2026 6:15 PM, 2m 46s, TESTUPJGFV42C9 c...), 'synthesize\_response' (Completed, Feb 3, 2026 5:37 PM, 13s, TESTUPJGFV42C9 c...), 'search\_products\_text' (Completed, Feb 3, 2026 5:37 PM, 8s, TESTUPJGFV42C9 c...), and another 'catalog\_ingestion (10)' (Completed, Feb 3, 2026 5:33 PM, 2m 32s, TESTUPJGFV42C9 c...). At the bottom, there's a page navigation bar showing 'Page 1 of 1' and '25/Page'.

## Monitoring

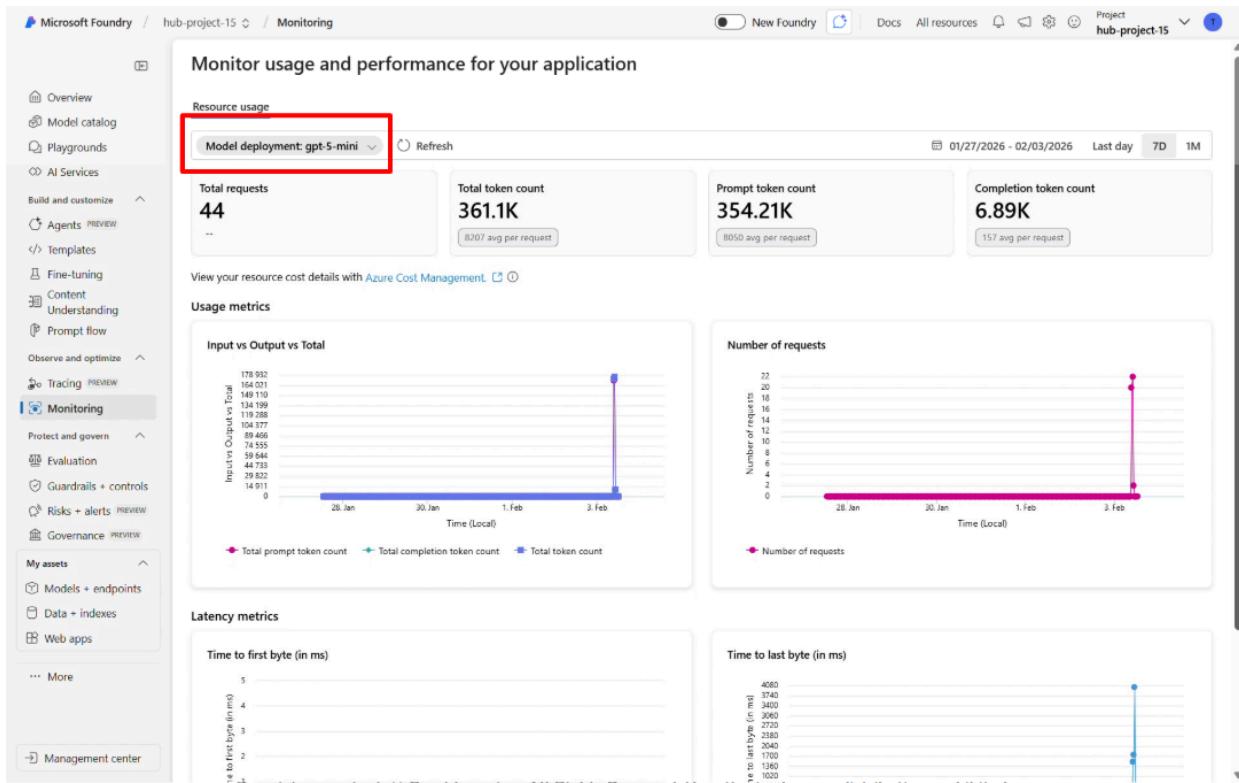
Monitoring of LLM applications provides the following benefits:

- Detection of "Model Drift" and "Knowledge Decay"
- Economic and Performance Visibility (Cost & Latency)
- Regulatory Compliance and Auditability
- Closing the "Feedback Loop" (Continuous Improvement)
- Safety and Guardrail Enforcement

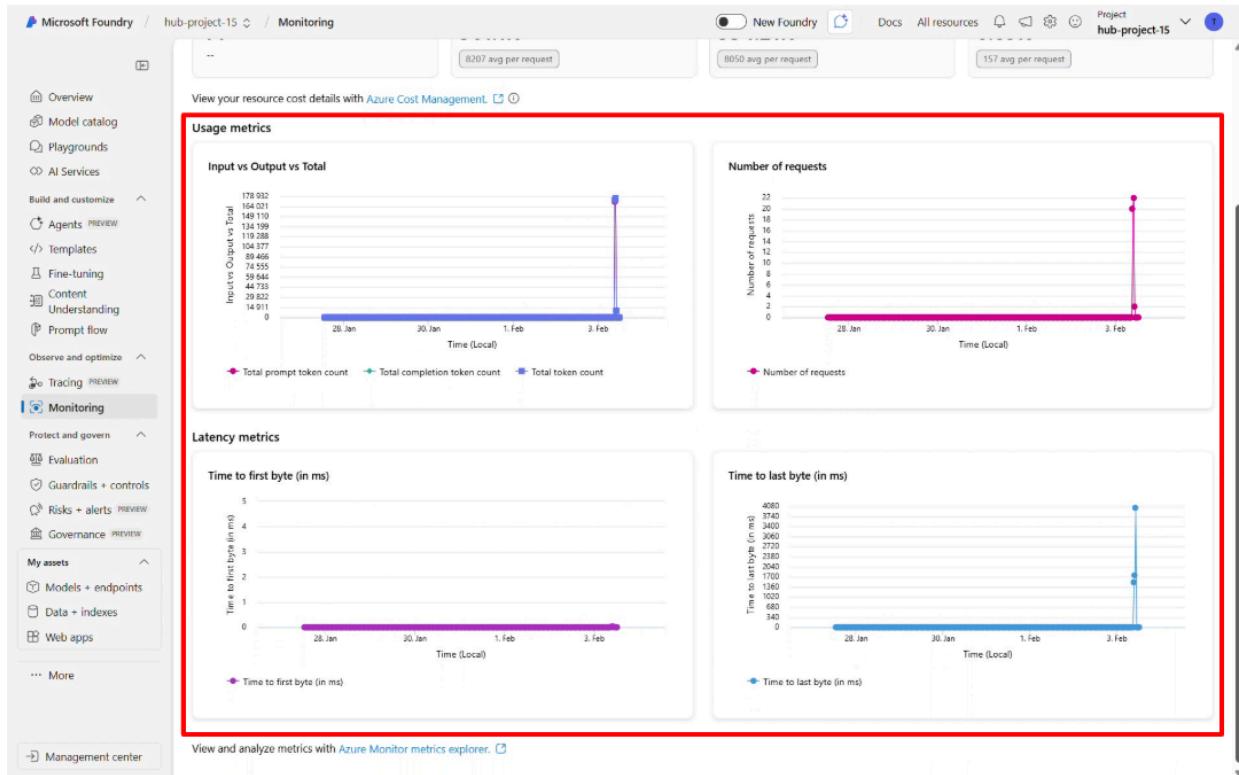
To ensure the RAG application remains a reliable, enterprise-grade asset, we implement a multi-layered monitoring strategy using **Azure App Services**, **Azure AI Search**, **Azure ML Workspace (or MLflow)** and **Azure AI Foundry**

### Azure Foundry Service

- Metrics that can be monitored in the Azure Foundry Service include:
- Request/Response Latency: Measure metrics such as Time to First Token (user-perceived latency), Time per Output Token (the "reading speed" of the AI), Total Request Latency (The end-to-end round trip.)
- Number of Requests: The total number of requests addressed by the model



The Usage metrics for the model are shown as illustrated (Total Tokens, Number of Requests, Latency metrics)



## Azure AI Search

Metrics that can be monitored in the Azure Foundry Service include:

- **Search Latency & Throughput:** Track the time taken to retrieve PDF chunks to ensure the "R" in RAG does not become a bottleneck.
- **Index Health & Freshness:** Monitor document ingestion success rates and ensure the vector store reflects the most recent insurance policy updates.
- **Query Performance:** Analyze "Zero-Result" queries to identify gaps in the knowledge base or issues with the embedding model.
- **Throttling Detection:** Monitor Search Unit (SU) utilization to preemptively scale during high-traffic periods (e.g., catastrophe season).

**Revolutionary retrieval with Azure AI Search**

Don't know where to start? Here are some options from directly within the portal

**Build your knowledge base**

Turn your data into an agentic knowledge base. Test grounded AI answers in the chat playground. [Learn more](#)

**Connect your data**

Start here to import your data. Learn how to quickly connect to your data to build your first search index. [Learn more](#)

**Monitor and scale**

Tools that allow you to monitor your system and scale for optimal performance. Adjust replicas and partitions as needed. [Learn more](#)

**Get started** Properties Usage Monitoring

Resource group (move) : default-resource-group Url : https://aisearchfree15.search.windows.net

Location (move) : East US 2 Pricing tier : Free

Subscription (move) : npqlbz2cmr-1760089588416 Replicas : 1 (No SLA)

Subscription ID : 45c4049b-fed4-4092-a8f3-a0f90e38644b Partitions : 1

Status : Running Search units : 1

Date created : Feb 3, 2026, 11:39:01 AM Compute type : Default

Tags (edit) : Add tags

JSON View

Add index Import data Import data (new) Search explorer Upgrade Refresh Delete Move

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Resource visualizer Agentic retrieval Search management Indexes Indexers Data sources Aliases Skillsets Settings Monitoring Alerts Metrics Diagnostic settings Logs Automation Help

Add or remove favorites by pressing **Ctrl+Shift+F**

**Navigate to Metrics in Monitoring.**

The screenshot shows the Microsoft Azure Metrics dashboard for a search service named 'aisearchfree15'. On the left, there's a navigation sidebar with various options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Agentic retrieval, Search management, Indexes, Data sources, Aliases, Skillsets, Debug sessions, Settings, Monitoring, Alerts, and Metrics (which is selected). The main area has a chart titled 'aisearchfree15 | Metrics' with a Y-axis from 0 to 100 and an X-axis from 6 PM to 6 AM UTC+05:30. At the top of the chart area, there's a 'Chart Title' input field, a '+ Add metric' dropdown menu (highlighted with a red box), and several other buttons like 'Add filter', 'Apply splitting', 'Line chart', 'Drill into Logs', 'New alert rule', 'Save to dashboard', and a three-dot ellipsis. A tooltip box is open over the 'Select a metric above to see data appear on this chart or learn more below:' section, containing three items: 'Filter + Split', 'Plot multiple metrics', and 'Build custom dashboards'.

Select the appropriate metric to be monitored

This screenshot is similar to the previous one but focuses on the 'Select metric' dropdown in the '+ Add metric' section. The dropdown is expanded, showing a list of metrics: 'Document processed count' (selected and highlighted with a red box), 'Search Latency', 'Search queries per second', 'Skill execution invocation count', and 'Throttled search queries percentage'. The rest of the dashboard interface is identical to the first screenshot.

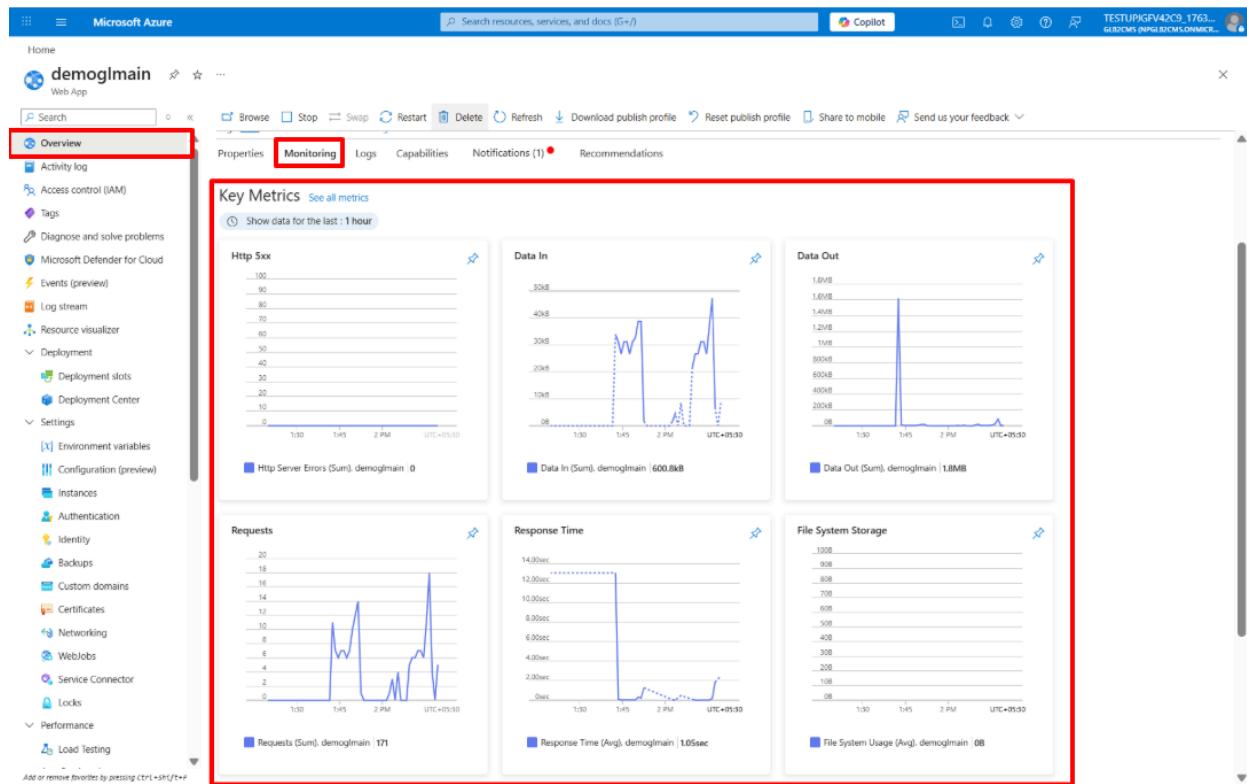
Add or remove favorites by pressing **Ctrl+F** or **Shift+F**.

[Give Feedback](#)

## Azure App Services

Metrics that can be monitored in the Azure Foundry Service include:

- **Request/Response Latency:** Measure end-to-end "Time to First Token" (TTFT) to ensure claims adjusters receive insights in under 2 seconds.
- **Resource Utilization:** Monitor CPU and Memory "spikes" during complex PDF parsing or heavy LLM orchestration tasks to optimize App Service Plan costs.
- **Auto-scaling Triggers:** Set alerts based on concurrency limits to ensure the system scales horizontally during peak claim filing hours.



Go to **Monitoring -> Metrics** in Azure App Services for detailed overview

