

Passo a Passo da Criação do Projeto

Autora: Iara Cristina Mescua Castro

Email: iaramescuacastro@gmail.com

Neste projeto irei usar créditos do Azure pela conta de estudante, com disponibilidade de 200 dólares de créditos no primeiro mês.

Página inicial >

Azure subscription 1 Assinatura

Pesquisar

Cancelar assinatura Renomear Alterar diretório Comentários

Visão geral

⚠ Seu crédito gratuito de US\$ 200,00 restante expira em 29 dias. Faça a atualização para manter a conta. →

Fundamentos

| | | | |
|------------------------|-----------------------------|----------------------------|------------------------|
| ID da Assinatura | : [REDACTED] | Nome da assinatura | : Azure subscription 1 |
| Diretório | : [REDACTED] (Proprietário) | Minha função | : Proprietário |
| Status | : Ativo | Plano | : Plano do Azure |
| Grupo de gerenciamento | : [REDACTED] | Classificação de Segurança | : Não disponível |

1. Crie um **Grupo de Recursos** no Painel do Azure. Meu grupo de recursos teve o nome “rg-azure-frontier-girls-challenge”.

Página inicial > Resource Manager

Resource Manager | Grupos de Recursos

Pesquisar

+ Criar Gerenciar a exibição Renovar Exportar para CSV Consulta aberta Atribuir marcas

Você está exibindo uma nova versão da experiência de navegação. Clique aqui para acessar a experiência antiga.

Filtrar por qualquer caixa Assinatura é igual a tudo Localização é igual a tudo + Adicionar filtro

2. Crie um **Recurso**: Entre no seu grupo de recursos e crie um recurso.

rg-azure-frontier-girls-challenge Grupo de recursos

Pesquisar

+ Criar Gerenciar a exibição Excluir o grupo de recursos Renovar Exportar para CSV Consulta aberta Atribuir marcas Mover Agrupar por nenhum Exibição JSON

Visão geral Fundamentos

| | | | |
|--------------------|------------------------|--------------|----------------|
| Assinatura (mover) | : Azure subscription 1 | Implantações | : 2 Êxito |
| ID da Assinatura | : [REDACTED] | Localização | : Brazil South |
| Rótulos (editar) | : Adicionar marcas | | |

Recursos Recomendações

Filtrar por qualquer caixa Tipo é igual a tudo Localização é igual a tudo + Adicionar filtro

3. No marketplace selecione **Azure Foundry AI** e configure-o de acordo com suas necessidades. No meu agente selecionei a localização “EAST US 2” para maior facilidade de acesso aos modelos.

Página inicial > Resource Manager | Grupos de Recursos > rg-azure-frontier-girls-challenge > Marketplace

Introdução Provedores de Serviço Pesquisar com IA Gerenciamento Marketplace Privado Gerenciamento da Oferta Privada Meu Marketplace Favoritos Minhas soluções Criado recentemente Planos privados Categorias

Azure Foundry

Preço: Todo Sistema Operacional: Todo Tipo de Publicador: Todo Tipo de Produto: Todo Nome do editor: All

Qualificado somente para o benefício do Azure Somente os serviços do Azure

Exibindo 1 a 20 de 38 resultados para ‘Azure Foundry’. Limpar pesquisa

Azure AI Foundry Microsoft Azure Service Design, customize, and manage AI apps and agents at scale.

QuSmart GENESIS Governance Agent - AI QuSmart.AI Azure Application QuSmart GENESIS Governance Agent creates a single, immutable AI Foundry model/project control layer.

Mosaic's LLM Foundry on Debian11 Apps4Rent LLC Virtual Machine Mosaic's LLM Foundry is an open-source toolkit for training and deploying large language models (LLMs) on Azure. Inicia em R\$ 0,366/hora

OMS Cloud Foundry monitoring Solution Microsoft Log Analytics OMS Cloud Foundry monitoring Solution for Azure Log Analytics(OMS) to monitor your Cloud Foundry

LTMindtree BlueVerse Foundry LTMindtree Limited Container (Kubernetes App) BlueVerse Foundry

Mosaic's LLM Foundry on Ubuntu20 Apps4Rent LLC Virtual Machine Mosaic's LLM Foundry is an open-source toolkit for training and deploying large language models (LLMs) on Azure. Inicia em R\$ 0,366/hora

4. Ao voltar para a tela de **Grupo de Recursos** e clicando no recurso criado, podemos acessar o **Portal do Azure Foundry**. Meu recurso teve o nome “b4388-mi0pgeii-eastus2”.

Página inicial > rg-azure-frontier-girls-challenge >

b4388-mi0pgeii-eastus2 Fábrica de IA do Azure

Pesquisar Go to Azure AI Foundry portal Excluir

Visão geral

Fundamentos

| | | | |
|---------------------------|--|-------------|---------------|
| Grupo de recursos (mover) | : rg-azure-frontier-girls-challenge | API Kind | : AI Services |
| Assinatura (mover) | : Azure subscription 1 | Localização | : eastus2 |
| ID da Assinatura | : [REDACTED] | Status | : Succeeded |
| Pontos de Extremidade | : Clique aqui para exibir os pontos de extremidade | | |
| Marcações (editar) | : Adicionar marcas | | |

Log de atividade IAM (Controle de acesso) Marcações Diagnosticar e resolver problemas Visualizador de recursos

Gerenciamento de Recursos Security Monitoramento Automação Ajuda

Build apps and agents in Azure AI Foundry
Unlock advanced AI capabilities, seamlessly integrate with your systems, and accelerate AI application development with Azure AI Foundry.

Go to Azure AI Foundry portal

5. Ao entrar no **Painel do Azure AI Foundry**, fiz o deploy de um modelo pela seção “Models + endpoints”, onde selecionei o gpt-4o-mini após clicar em **Deploy Model**. Selecionei o gpt-4o-mini não apenas por consumir menos créditos por token em comparação aos modelos 4.1 e 5.0 do gpt, mas também por oferecer um excelente equilíbrio entre custo, velocidade e qualidade de análise. Ele é otimizado para tarefas de compreensão e sumarização, entregando respostas consistentes e contextuais. Além disso, possui baixa latência, permitindo que o agente processe listas extensas de comentários de forma rápida. Quando for escolher seu modelo, verifique se ele está disponível na região do seu recurso.

Azure AI Foundry / b4388-mi0pgeii-eastus2_project / Models + endpoints

Model deployments

| Name | Model name | Model version | State | Model retirement date | Content filter | Deployment type | Fine-tur |
|-------------|-------------|---------------|-----------|-----------------------|----------------|-----------------|----------|
| gpt-4o-mini | gpt-4o-mini | 2024-07-18 | Succeeded | Feb 26, 2026 9:00 PM | DefaultV2 | Global Standard | |

Model deployments Service endpoints

+ Deploy model Refresh Edit Open in playground Reset view

Unsupported browser: This site is supported and works best with the latest version of the new Microsoft Edge (Edge legacy is not fully supported). Google Chrome, Mozilla Firefox or Apple Safari. The site may not work correctly on other versions or browsers.

Help

Overview Model catalog Playgrounds Build and customize Agents Templates Fine-tuning Content Understanding Observe and optimize Tracing Monitoring Protect and govern Evaluation Guardrails + controls Risks + alerts Governance Azure OpenAI Stored completions Batch jobs My assets Models + endpoints More

6. Configure o agente. Na seção **Agents**, nomeei meu agente de **FeedbackAgent** e configurei sua instrução como “Resuma as impressões gerais dos comentários em um parágrafo conciso, destacando os pontos positivos, os elogios comuns e as críticas recorrentes. Se houver pontos negativos significativos, mencione-os objetivamente”. Também adicionei uma descrição opcional para fins de organização.

The screenshot shows the Azure AI Foundry interface. On the left, a sidebar lists various sections like Overview, Model catalog, Playgrounds, Build and customize, Agents (which is selected and highlighted with a red box), Templates, Fine-tuning, Content, Understanding, Observe and optimize, Tracing, Monitoring, Protect and govern, Evaluation, Guardrails + controls, Risks + alerts, Governance, Azure OpenAI, Stored completions, and Batch jobs. The main area is titled 'Create and debug your agents' and shows a table of 'My agents'. One agent, 'FeedbackAgent', is listed with columns for Name, ID, Model (gpt-4o-mini), and Created (Nov 15, 2024). To the right, a 'Setup' panel shows the 'Agent ID' (asst_...), 'Agent name' (FeedbackAgent), and 'Deployment' (gpt-4o-mini (version:2024-07-18)). A large red box highlights the 'Instructions' section, which contains the following text:

Resuma as impressões gerais dos comentários em um parágrafo conciso, destacando os pontos positivos, os elogios comuns e as críticas recorrentes. Se houver pontos negativos significativos, mencione-os objetivamente.

7. De preferência, teste se o agente está coerente em **Try in playground**. Em um chat, enviei 3 simulações de feedback de um evento separados por aspas e espaço, onde o agente deveria retornar um resumo das impressões dos participantes conforme a instrução dada. Note que a resposta do agente está de acordo com o **Input** e a **Instrução**. Esse agente pode economizar tempo de organizadores que precisam analisar opiniões e feedbacks de inúmeros usuários de uma única vez.

The screenshot shows the 'Agents playground' interface. At the top, there's a header with '← Agents playground' and a 'Help' link. Below it, a toolbar has buttons for 'New agent', 'View code', 'Delete', 'Create trigger', and a 'PREVIEW' button. The main area has tabs for 'New thread' and 'Thread logs', with a current count of 248t. A red box highlights a user query in a text input field:

"""
 O evento foi ótimo, gostei de como o calendário foi organizado. As palestras tiveram alta qualidade.
 """
 """
 O evento foi interessante, profissionais bem capacitados e carismáticos. Porém houve um atraso para começar.
 """
 """
 O evento foi bom, gostaria de mais tempo para tirar dúvidas com os palestrantes, mas foi divertido!
 """

Below the query, a red box highlights the generated response summary:

As impressões gerais dos comentários sobre o evento são bastante positivas, destacando a organização do calendário e a qualidade das palestras, com elogios à capacitação e carisma dos profissionais envolvidos. No entanto, algumas críticas recorrentes incluem o atraso no início do evento e a falta de tempo suficiente para interagir com os palestrantes. Apesar dessas questões, a experiência foi considerada divertida e enriquecedora.

At the bottom, there's a 'Type user query here. (Shift + Enter for new line)' input field and a 'View Run Info' button.

8. Selecione seus dados. Minha ideia principal era integrar o agente à respostas de um formulário. Então pensando no **Microsoft Forms**, isso pode ser feito rapidamente clicando em **Exibir respostas** e **Abrir Resultados no Excel**. Criei uma simulação de formulário abaixo para representar a situação quando você é proprietário no formulário.

The top screenshot shows the Microsoft Forms interface for a survey titled "Pesquisa de comentários pós-evento". The "Exibir respostas" button is highlighted with a red box. The bottom screenshot shows the "Visão Geral das Respostas" (General Response View) dashboard, which includes a summary card with 8 responses, average time of 01:10, and duration of 0 days. It also displays a satisfaction rating chart with a mean of 4.38 and a 5-star classification. The "Pesquisa de comentários p..." link in the "Insights e ações" section is highlighted with a red box.

Mas o agente também pode ser útil para outros tipos de formulário, feedbacks e comentários desde que estejam armazenados em um arquivo de texto. O código também poderia ser adaptado para receber outros tipos de arquivo além de CSV.

9. Faça a integração com Python ou a linguagem que for realizar sua automação. Criei o arquivo app.py com o aplicativo que gerasse a interface. Ela recebe um CSV na variável upload_file, o separador separator e o encoder encoding. A interface gera erros caso o usuário não preencha um dos campos corretamente e avisa o usuário caso a leitura do CSV falhe.

The screenshot shows the "Feedback Analyzer with Azure Foundry Agent" application interface. On the left, there is configuration for "AGENT_ID" and "PROJECT_ENDPOINT", with a note: "If you prefer, set AGENT_ID and PROJECT_ENDPOINT as environment variables instead." Below this is a "Quick run" section with the command "python -m streamlit run app.py". The main area is titled "Feedback Analyzer with Azure Foundry Agent" and contains fields for "Upload your feedback CSV" (with a "Drag and drop file here" button and a "Browse files" button), "CSV Separator" (set to ;), "File Encoding" (set to windows-1252), and a note: "Please upload a CSV file to start. Default parameters are meant for Microsoft Forms exports."

É uma interface simples em streamlit que pode ser aberta localmente desde que o usuário tenha a biblioteca (e as outras dependências em “requirements.txt”) com o comando: `streamlit run app.py`. A imagem acima mostra a tela inicial e a imagem abaixo como ela fica ao selecionar um arquivo.

Azure Agent Configuration

AGENT_ID

PROJECT_ENDPOINT

If you prefer, set `AGENT_ID` and `PROJECT_ENDPOINT` as environment variables instead.

Quick run: `python -m streamlit run app.py`

Feedback Analyzer with Azure Foundry Agent

Upload your feedback CSV

Drag and drop file here
Limit 200MB per file • CSV

Browse files

pesquisade_comentarios_pos_evento.csv 1.7KB

CSV Separator: ; File Encoding: windows-1252

Preview of uploaded data (first 5 rows):

| | Id | Hora de inicio | Hora de conclusão | Email | Nome | No geral, qual é o seu nível de satisfação com o evento? | Esta é a primeira vez que você participa de eventos como este? | Qual é a probabilidade de vc |
|---|----|------------------|-------------------|-----------|------|--|--|------------------------------|
| 0 | 1 | 15/11/2025 17:20 | 15/11/2025 17:23 | anonymous | None | 5 | Sim | |
| 1 | 2 | 15/11/2025 17:23 | 15/11/2025 17:24 | anonymous | None | 5 | Não | |
| 2 | 3 | 15/11/2025 17:24 | 15/11/2025 17:24 | anonymous | None | 3 | Não | |
| 3 | 4 | 15/11/2025 17:24 | 15/11/2025 17:25 | anonymous | None | 4 | Não | |
| 4 | 5 | 15/11/2025 17:25 | 15/11/2025 17:26 | anonymous | None | 4 | Não | |

Select column with feedback text
O que você mais gostou no evento?

Send to model

Quando selecionamos um arquivo, a interface mostra uma prévia dele em uma tabela para o usuário selecionar com segurança a coluna desejada. No caso do Microsoft Forms exportado para CSV, basta selecionar a pergunta na qual queremos obter todas as suas respostas. Para mais detalhes do código da interface veja o arquivo `app.py` no repositório.

10. Desenvolva uma função que use a API do Azure AI Foundry para chamar o seu agente. Criei a função `run_agent()` no arquivo `azure_agent_worker.py` para criar uma thread (baseado no **Endpoint do Projeto**) através das classes `azure.ai.agents` e `azure.identity` onde selecione meu agente através do seu **ID** e envio o conteúdo a ser analisado pelo parâmetro “content”. Veja que antes o Input passa pela função `build_message()` que separa os comentários com aspas e espaço, uma abordagem eficiente de Prompt Engineering para ajudar o modelo a diferenciar um feedback do outro.

```

azure_agent_worker.py M X
azure_agent_worker.py > ...
1  import os
2  from typing import Optional
3  import pandas as pd
4  from azure.ai.agents import AgentsClient
5  from azure.identity import DefaultAzureCredential
6  from dotenv import load_dotenv
7
8  load_dotenv()
9
10 def build_message(data: list[str]) -> str:
11     quotes = '""'
12     separator = f"\n{n{quotes}}\n\n{n{quotes}}\n"
13     return quotes + "\n" + separator.join(data) + "\n" + quotes
14
15
16 def get_response(client: AgentsClient, thread_id: str) -> str:
17     # Messages are listed from newest to oldest
18     for message in client.messages.list(thread_id):
19         if message.role == "assistant":
20             return message.content[0].text.value
21
22     raise RuntimeError("Could not find the assistant response message.")
23
24
25 def run_agent(
26     data: list[str],
27     agent_id: Optional[str] = None,
28     project_endpoint: Optional[str] = None,
29     temperature: float = 0.8,
30 ) -> str:
31     """Run the Azure Foundry agent with the provided list of texts and return its response.
32
33     This function reads `AGENT_ID` and `PROJECT_ENDPOINT` from the environment if not provided.
34     Raises ValueError when required configuration is missing, RuntimeError on auth or API failures.
35     """
36
37     # Load env again in case Streamlit changed it or user provided in UI
38     load_dotenv(override=True)
39     AGENT_ID = agent_id or os.getenv("AGENT_ID")
40     PROJECT_ENDPOINT = project_endpoint or os.getenv("PROJECT_ENDPOINT")
41
42     if not AGENT_ID or not PROJECT_ENDPOINT:
43         raise ValueError("Missing AGENT_ID or PROJECT_ENDPOINT environment variables.")
44
45     try:
46         credential = DefaultAzureCredential()
47         client = AgentsClient(PROJECT_ENDPOINT, credential)
48     except Exception as exc:
49         raise RuntimeError(f"Authentication failed: {exc}")
50
51     message = build_message(data)
52
53     try:
54         response = client.create_thread_and_process_run(
55             agent_id=AGENT_ID,
56             thread={"messages": [{"role": "user", "content": message}]},
57             temperature=temperature,
58             polling_interval=1,
59         )
60     except Exception as exc:
61         raise RuntimeError(f"Failed to send request to agent: {exc}")
62
63     try:
64         return get_response(client, response.thread_id)
65     except Exception as exc:
66         raise RuntimeError(f"Failed to get agent response: {exc}")

```

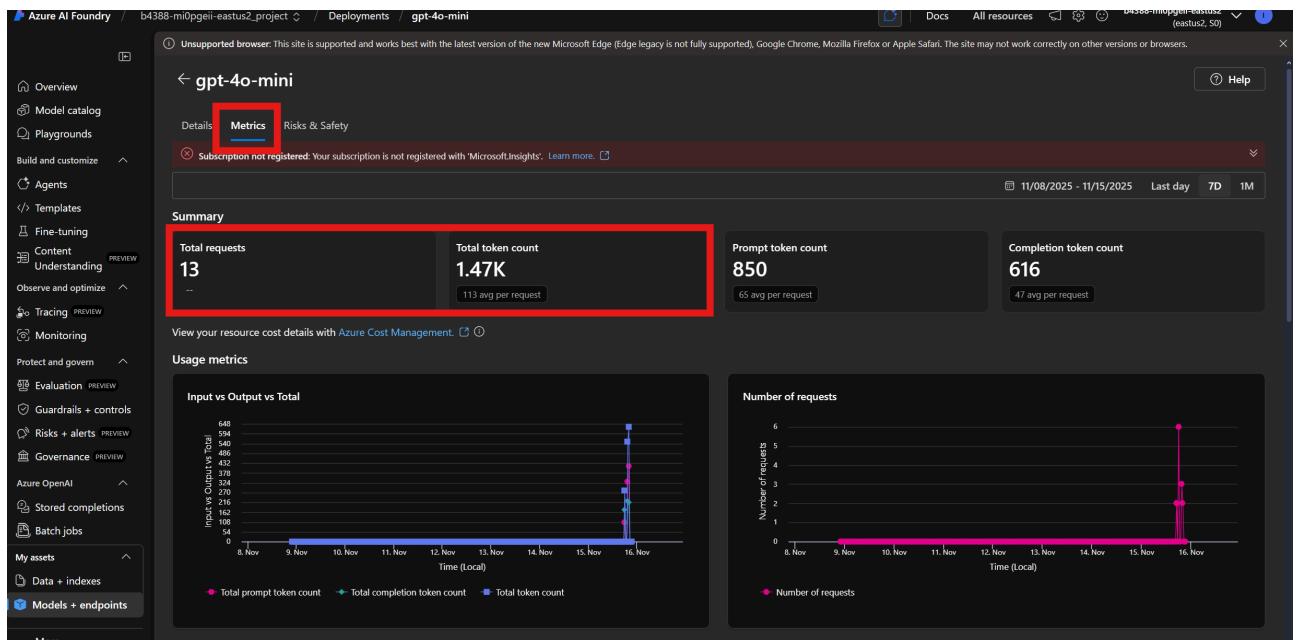
11. Integre a interface com o agente. Continuando o código da interface, caso o botão **Run model** seja pressionado, obteremos o **ID do Agente** e o **EndPoint do Projeto** que foram preenchidos nos campos da Barra lateral **Azure Agent Configuration**. Caso estejam vazios, serão utilizados as variáveis em `.env`. Depois chamaremos a função `run_agent()` para receber todos os comentários selecionados e retornar a resposta do modelo contida na variável `result`. Exemplo de uso está no `README`.

```
if run_button:
    try:
        texts = df[selected_col].dropna().astype(str).tolist()
        if len(texts) == 0:
            st.error("Selected column is empty. Choose another column with text values.")
            st.stop()
    except Exception as e:
        st.error(f"Failed to extract column data: {e}")
        st.stop()

    # Use sidebar values if provided, otherwise rely on environment variables
    agent_id = agent_id_input.strip() or None
    project_endpoint = project_endpoint_input.strip() or None

    with st.spinner("Sending data to the model and waiting for response..."):
        try:
            result = run_agent(texts, agent_id=agent_id, project_endpoint=project_endpoint)
            st.success("Model returned a response:")
            st.code(result, "markdown", wrap_lines=True)
        except Exception as e:
            st.error(e)
```

12. Usei a subseção **Metrics** na seção **Models + endpoints** para acompanhar as requests de testes e tokens gastos pelo meu modelo durante testes. É importante não fazer requisições em massa para não consumir créditos desnecessariamente.



Limitações e Futuras Expansões

Atualmente, o projeto está limitado à análise de comentários textuais extraídos de arquivos CSV, e depende da qualidade do modelo de linguagem selecionado, que pode apresentar limitações na identificação de sarcasmo, ironia ou contextos muito específicos. Futuramente, o sistema pode ser expandido para aceitar outros formatos de entrada (como integrações diretas com Google Forms ou Microsoft Forms via API), gerar relatórios detalhados com os gráficos mais relevantes e realizar análises por categorias ou segmentos, processar grandes volumes de dados de forma escalável e permitir feedbacks individuais automatizados para cada participante quando há poucos comentários. Também é possível integrar análise de sentimentos multimodal (texto + áudio ou imagem) para situações específicas.