

# Watsonx Orchestrate Chat with Agents

API REST para comunicação com IBM Watson Orchestrate

IBM CLOUD

POWERED

 FASTAPI

FRAMEWORK

 PYTHON

3.8+

## Índice

- [Visão Geral](#)
- [Endpoints Disponíveis](#)
- [Como Obter as Credenciais](#)
- [Configuração do Projeto](#)
- [Documentação da API](#)
- [Variáveis de Ambiente](#)
- [Notas Importantes](#)

## Visão Geral

Este asset fornece um **serviço HTTP baseado em FastAPI** que permite a comunicação com o **IBM Watson Orchestrate (WxO)** por meio de três endpoints principais:

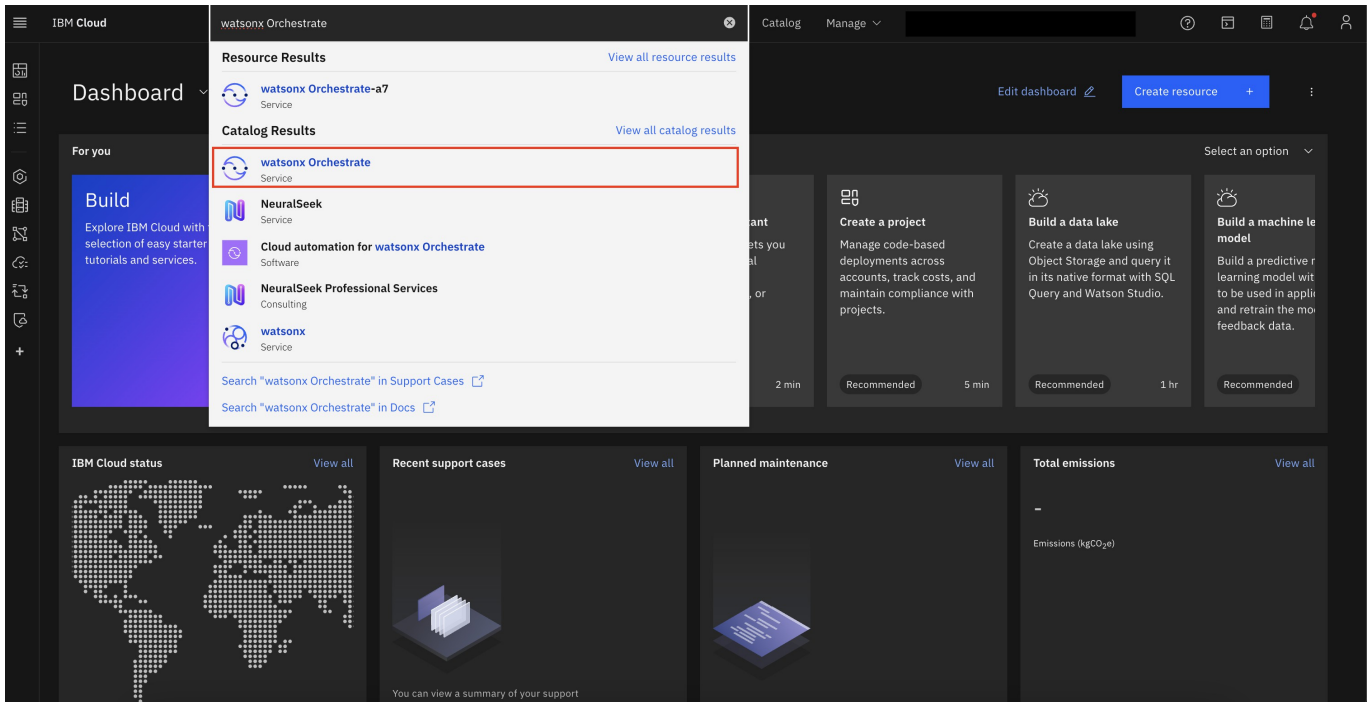
Endpoint	Descrição
<a href="#">/list_agents</a>	Retorna uma lista de agentes disponíveis em uma determinada instância de watsonx Orchestrate
<a href="#">/get_response</a>	Retorna uma única resposta de um agente WxO
<a href="#">/stream_response</a>	Transmite a resposta do agente em tempo real usando Server-Sent Events (SSE)

## Links Úteis

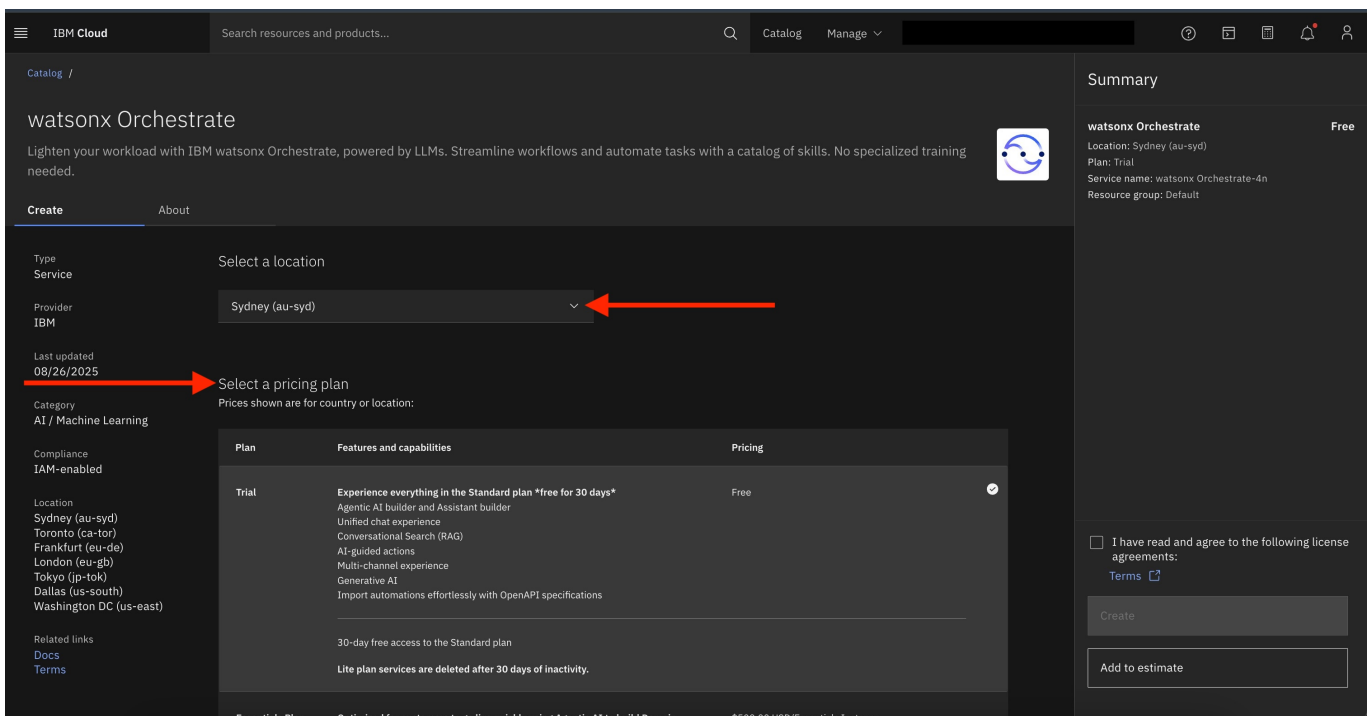
- **Código Fonte:** [GitHub Repository](#)
- **Endpoints Implementados:** [Routes Directory](#)

## Criar uma Instância do Watsonx Orchestrate

Após fazer login na **IBM Cloud**, no campo de pesquisa digite **watsonx Orchestrate**.

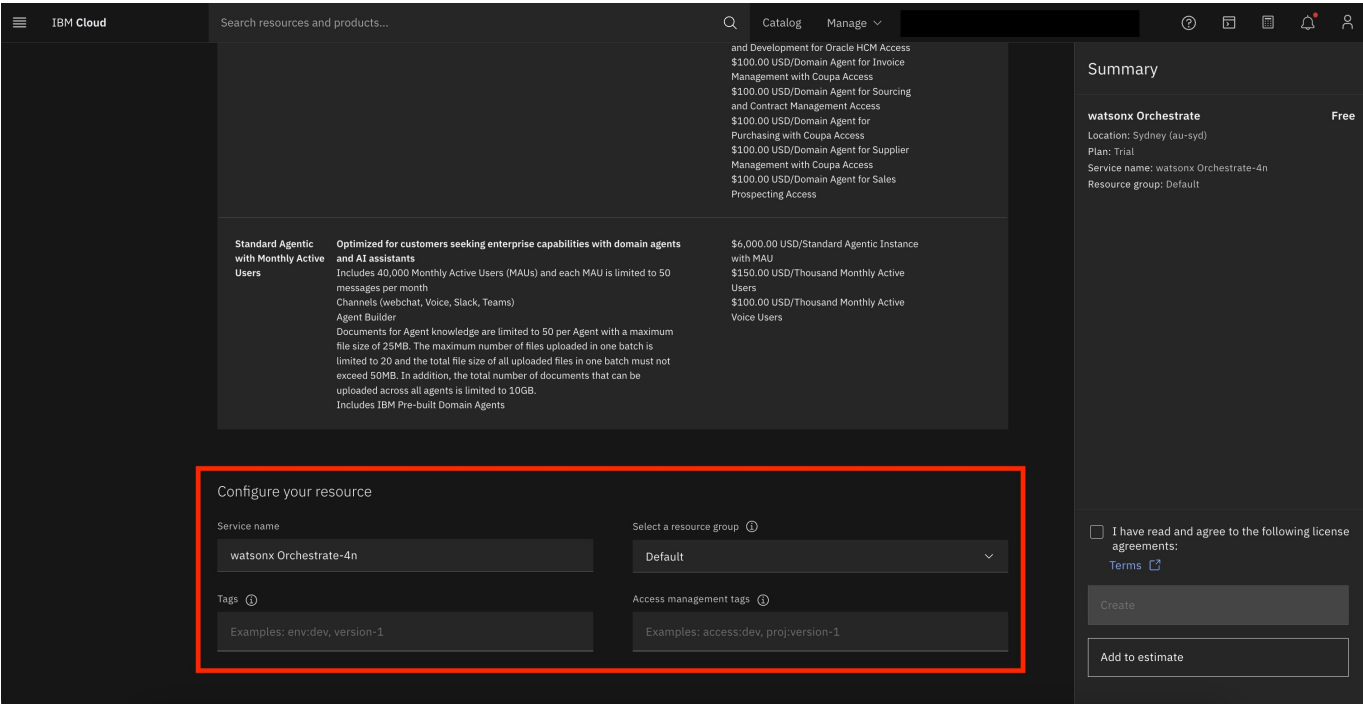


Escolha a **location** de sua instância e o **plano** que lhe atende melhor

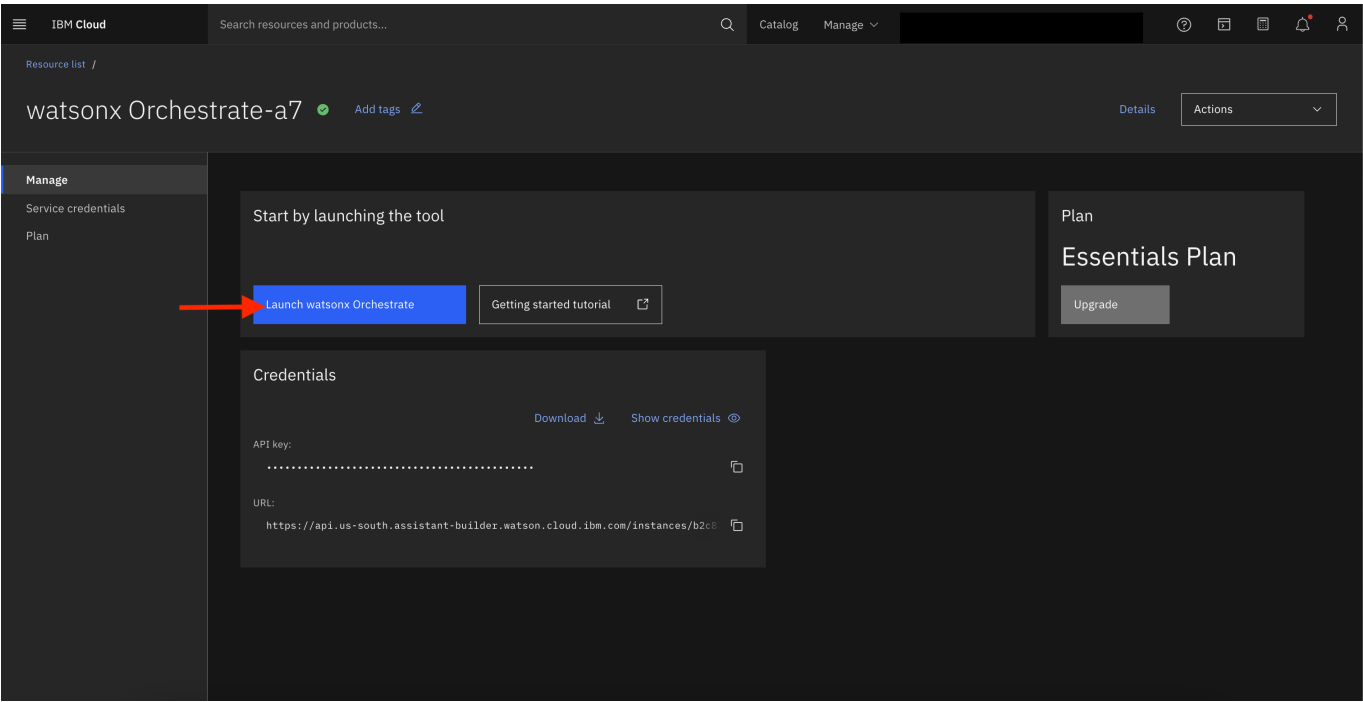


Por fim, digite o **nome do serviço**, opcionalmente **tags** e **tags de acionamento**, e selecione um **resource group**.

Aceite os termos da IBM Cloud ao lado direito da tela e clique em **Create**.



Com sua instância criada, clique em **Launch watsonx Orchestrate**.

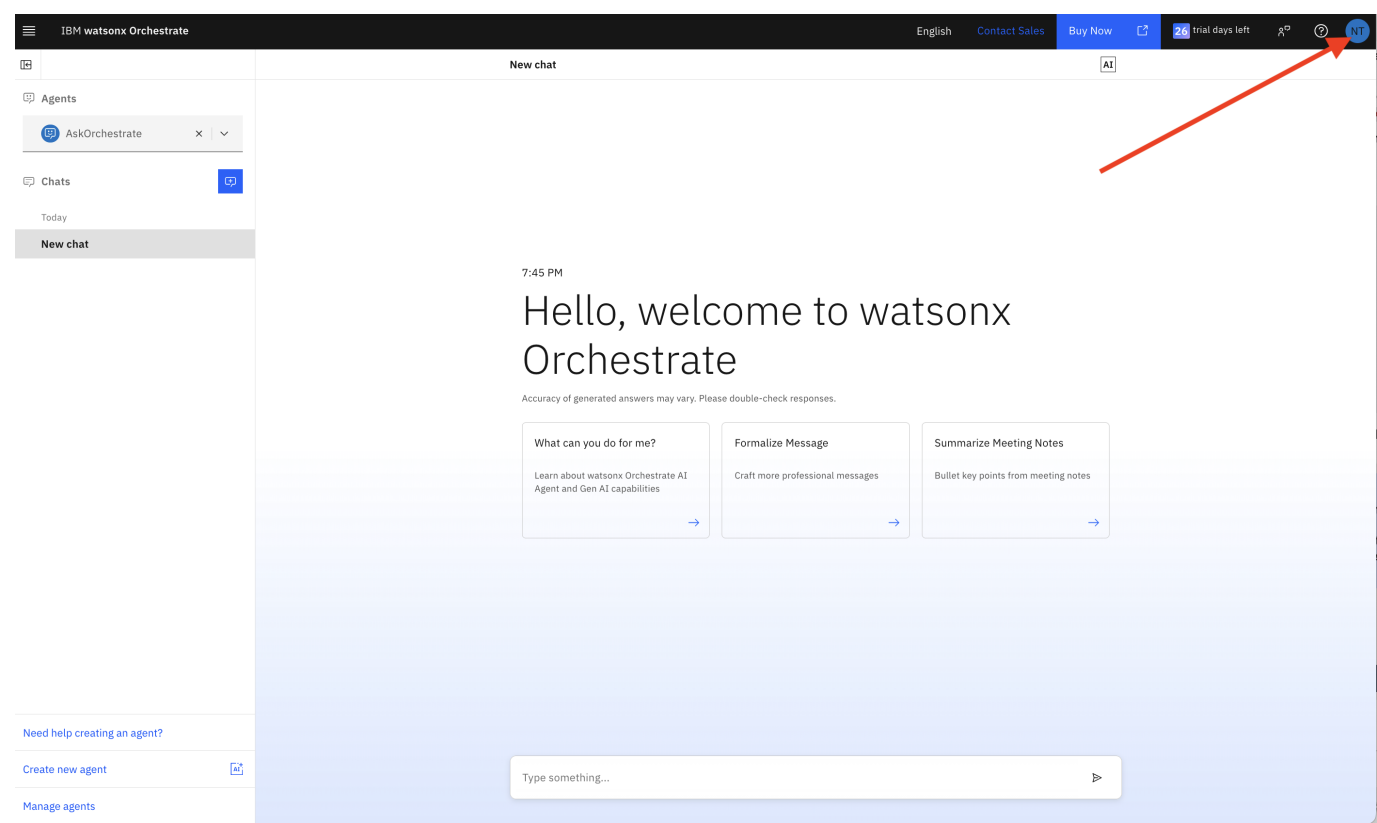


Para se comunicar com o Watsonx Orchestrate via API, você precisará obter duas credenciais essenciais:

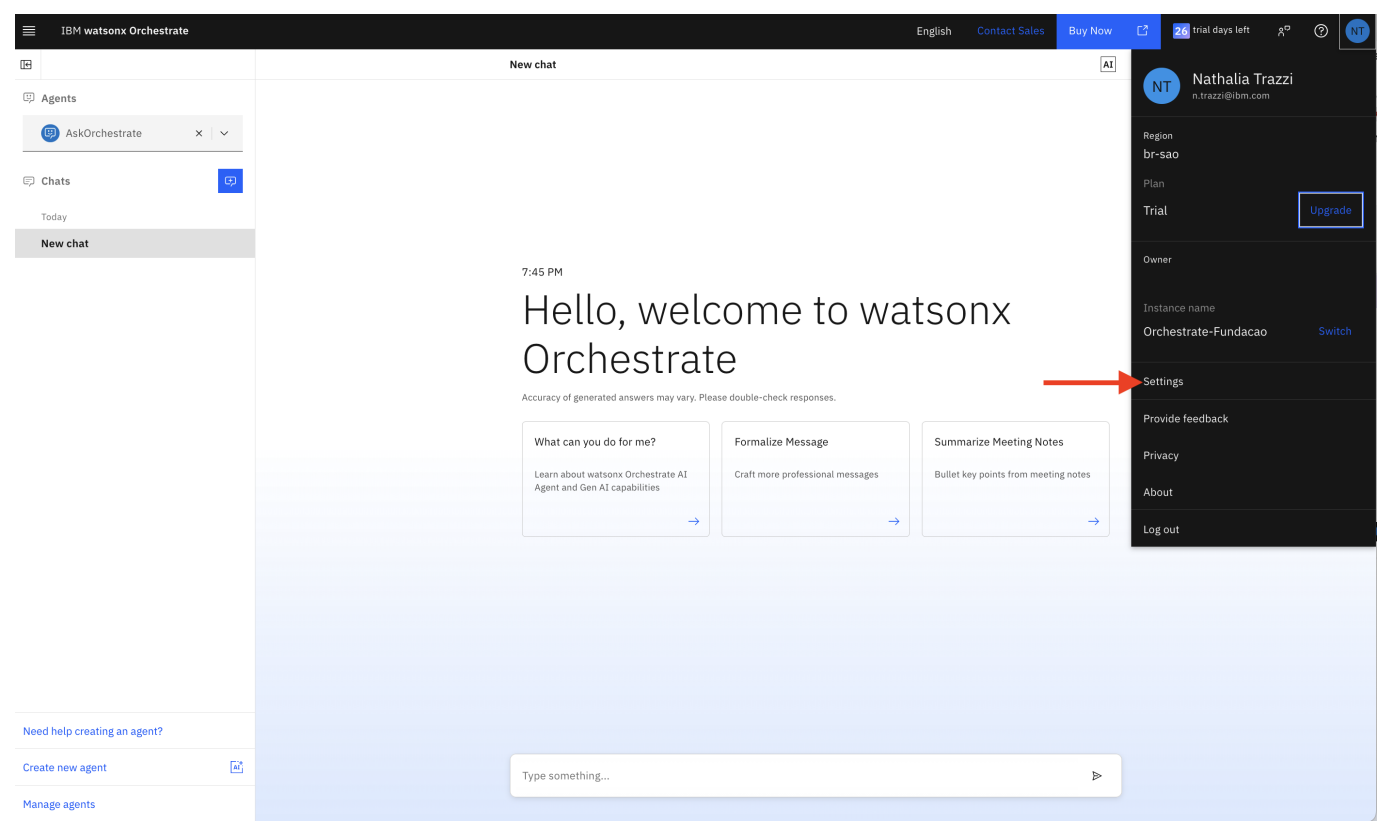
- IAM\_API\_KEY
- WXO\_INSTANCE\_ID

## Como obter as credenciais do watsonx Orchestrate

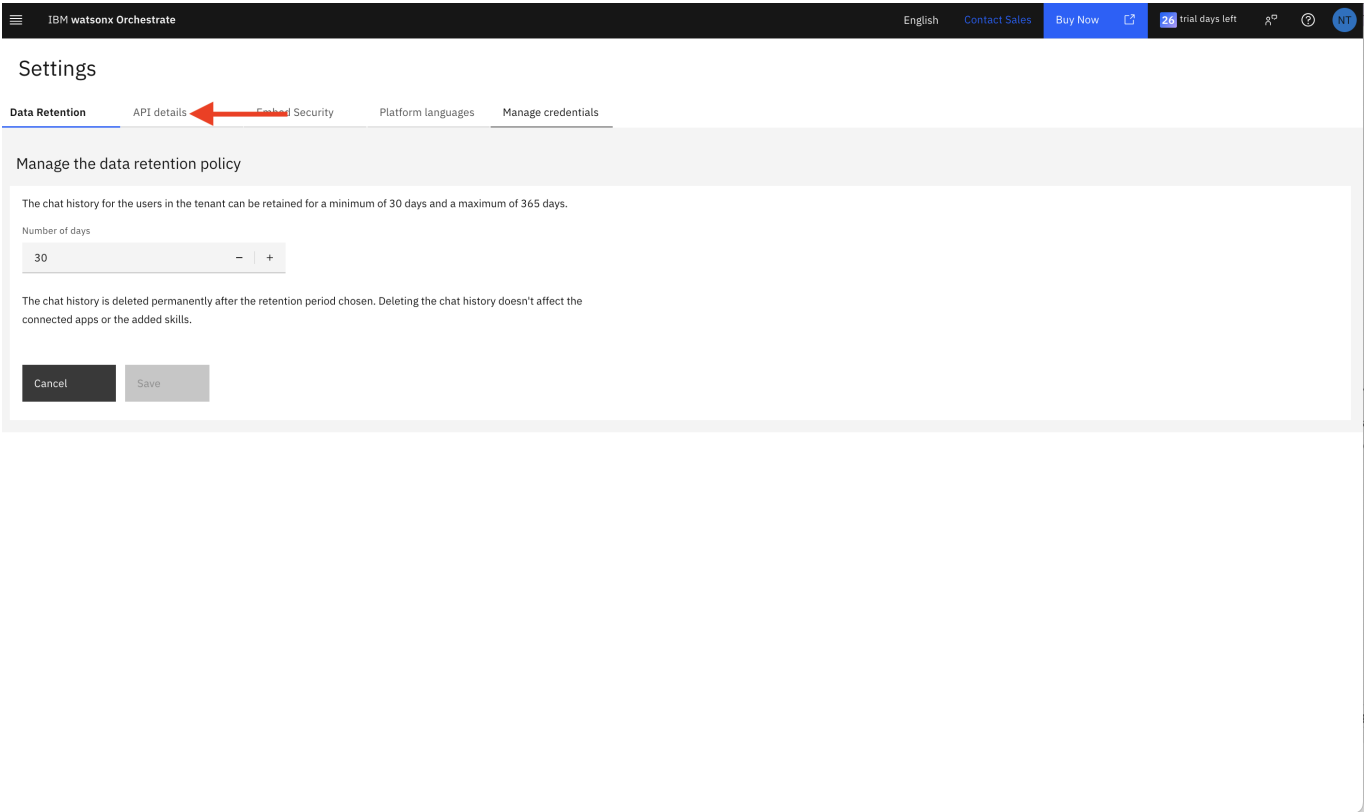
No ícone da sigla de seu nome e sobrenome ao lado direito da interface



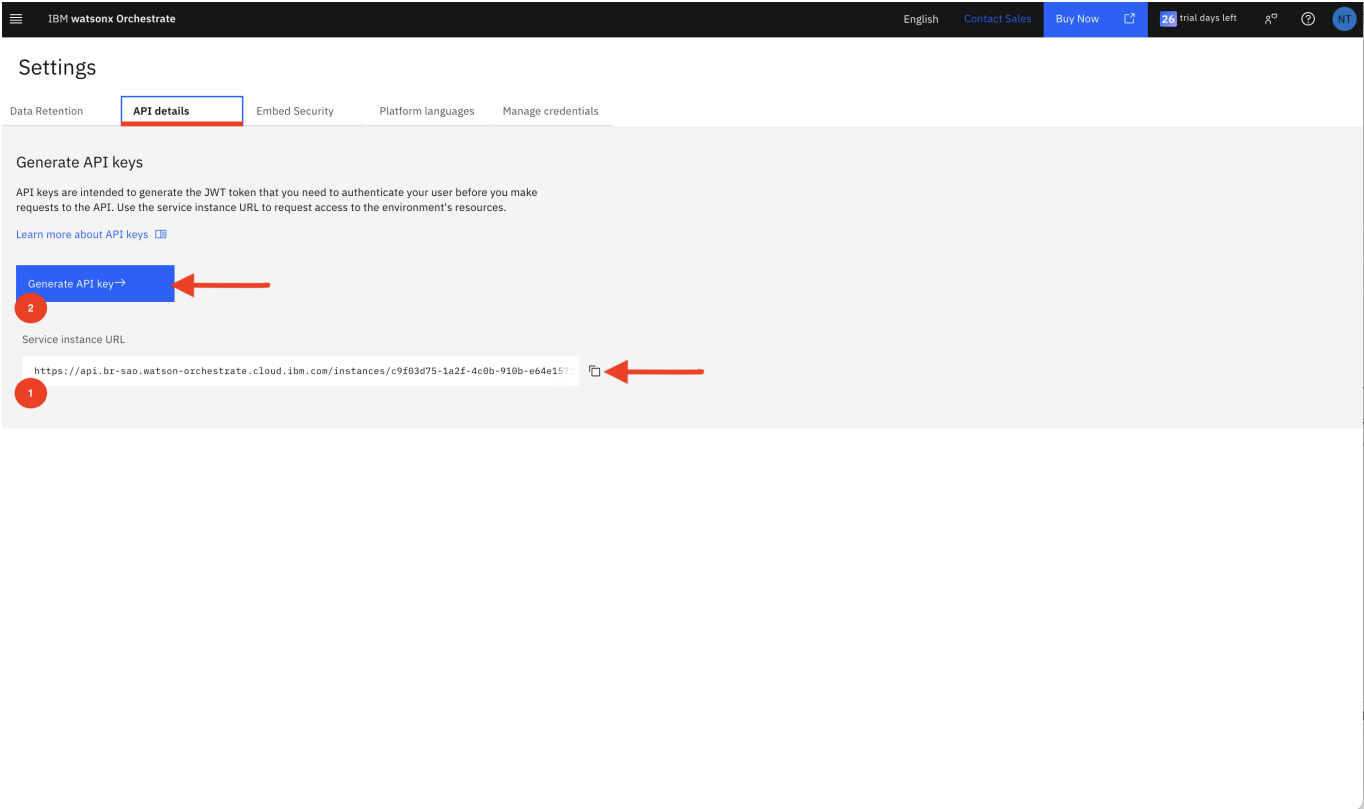
Clique em **Settings**.



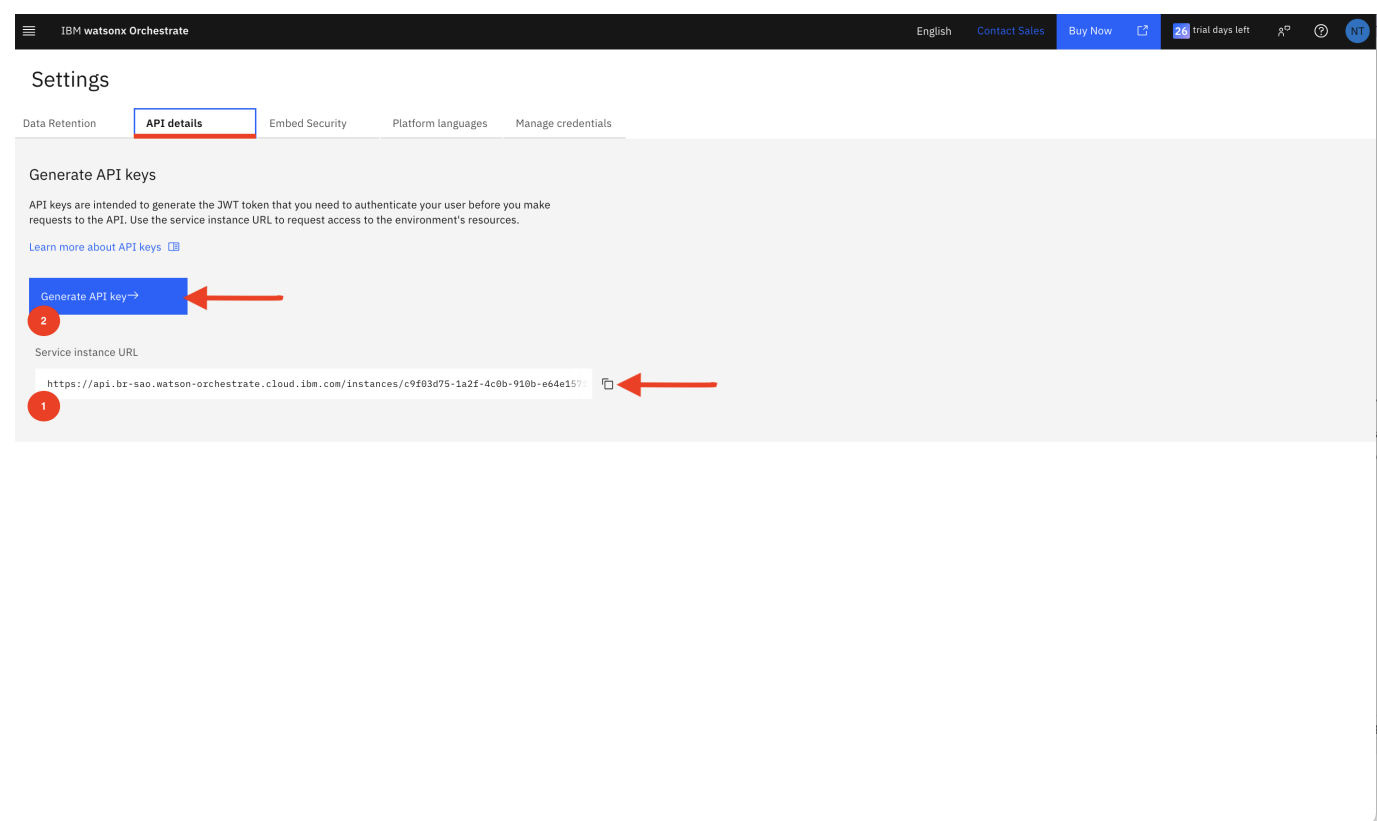
Na página de configuração e entre na aba **API details**



1. Copie o ID de sua instância

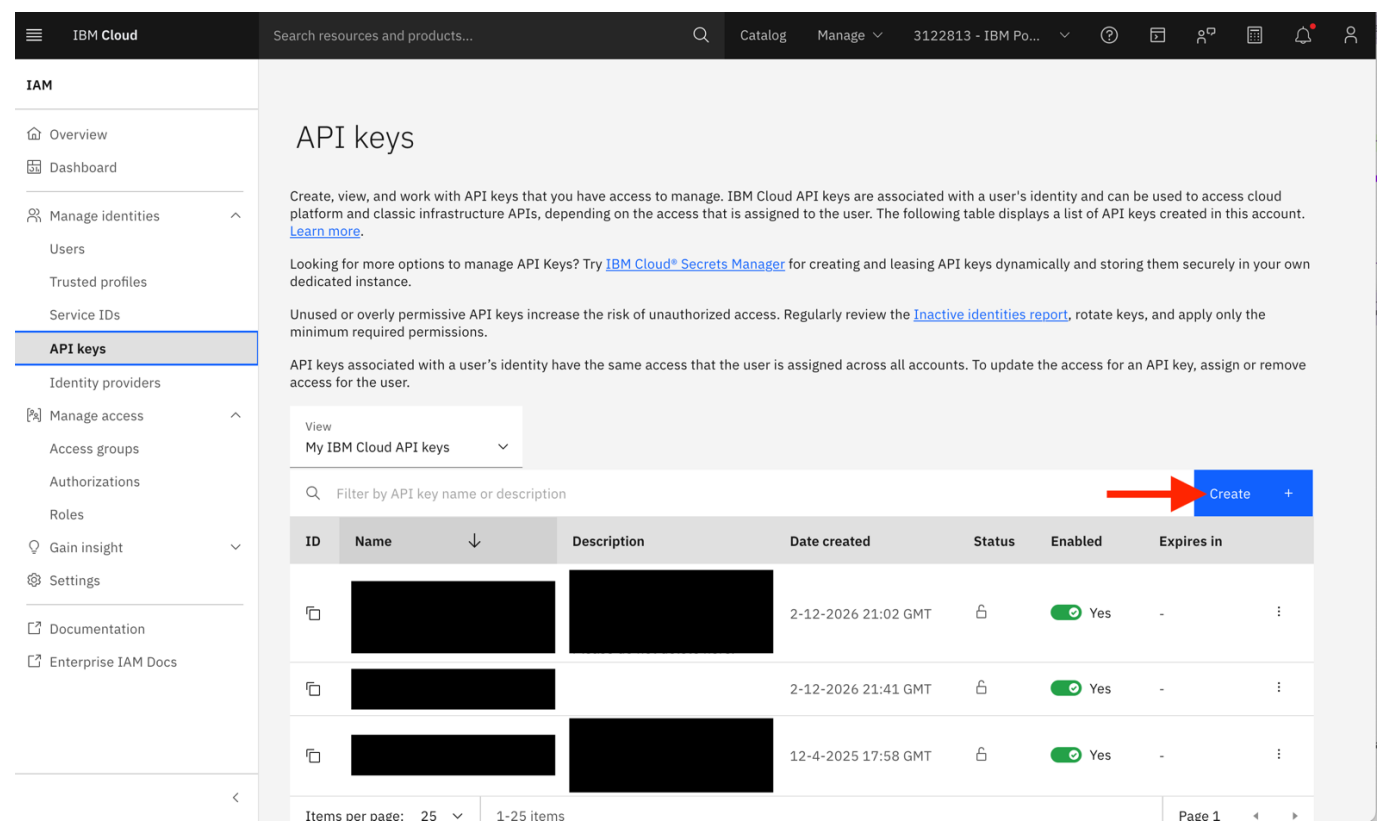


2. Em seguida, clique em **Generate API Key**



Na próxima página, clique em **Generate API key**.

Você será redirecionado para o catálogo onde você cria, visualiza e trabalha com chaves de API da IBM Cloud.



### O que é uma API Key?

Uma API Key é um segredo que permite que usuários, aplicações ou serviços acessem a IBM Cloud de forma autenticada. Ela funciona como uma "senha técnica" usada em CLIs, automações,

pipelines ou integrações.

## Configuração da API Key

### Name (Nome)

Dê um nome descritivo para a chave, como:

- `apikey-wxo`
- `local-development-key`

**Dica:** Use nomes que indiquem o propósito, responsável ou ambiente.

### Description (Opcional)

Aqui você pode adicionar uma descrição da sua chave, ajudando na identificação dela.

### Lead Action (Ação em Caso de Vazamento)

Opção	Descrição
<input type="radio"/> <b>Disable the leaked key</b> (recomendado)	A chave é automaticamente desativada. Boa opção para produção.
<input type="radio"/> <b>Delete the leaked key</b>	A chave é deletada permanentemente.
<input type="radio"/> <b>Nothing</b>	Nenhuma ação é tomada. Não recomendado.

### API Key Expiration (Expiração)

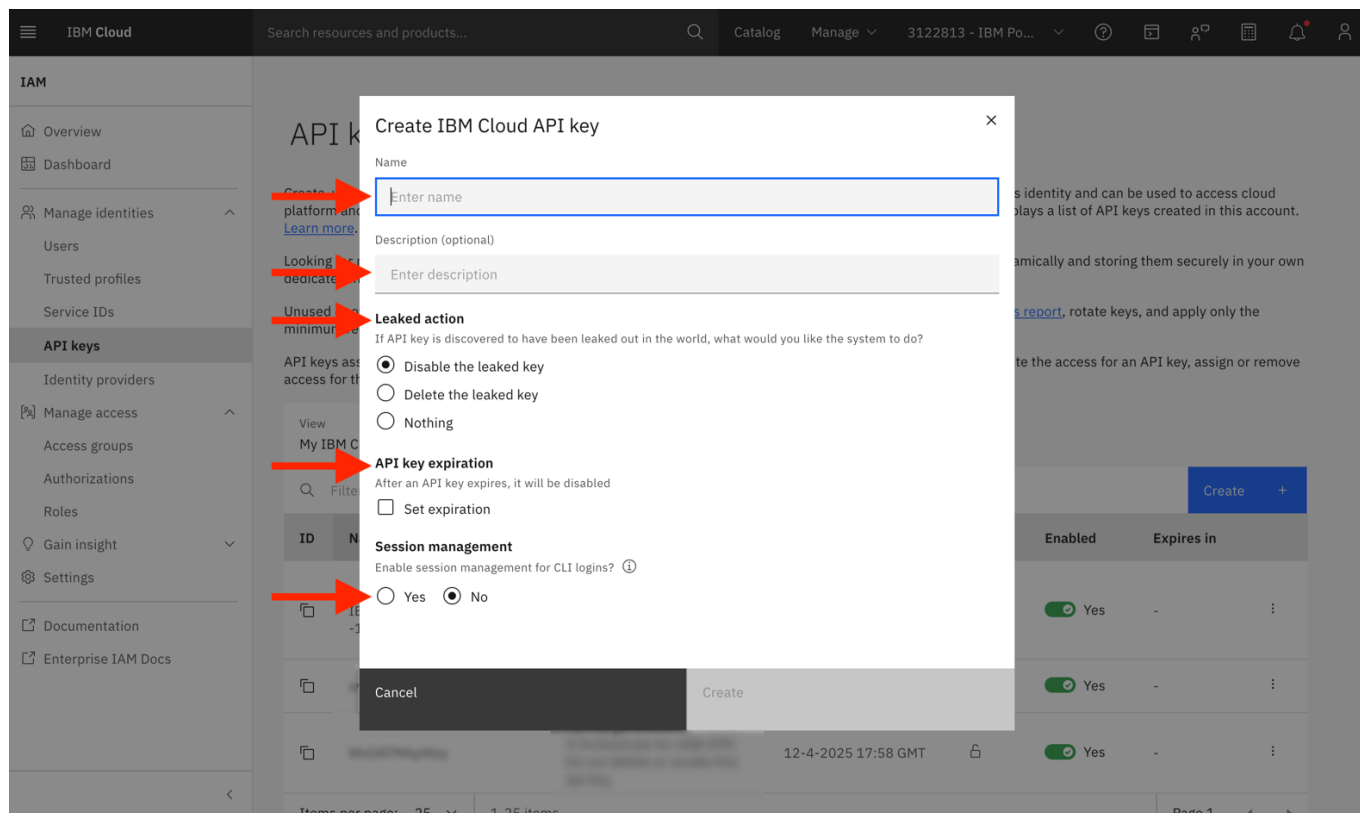
Você pode escolher se a chave deve expirar automaticamente.

Opção	Descrição
<input type="radio"/> <b>Set expiration</b>	Permite escolher uma data futura. -> Ideal para acessos temporários, integrações de curto prazo, etc. Quando expirar, a chave é automaticamente desabilitada.
<input type="radio"/> <b>Sem expiração</b> (default)	A chave não expira.

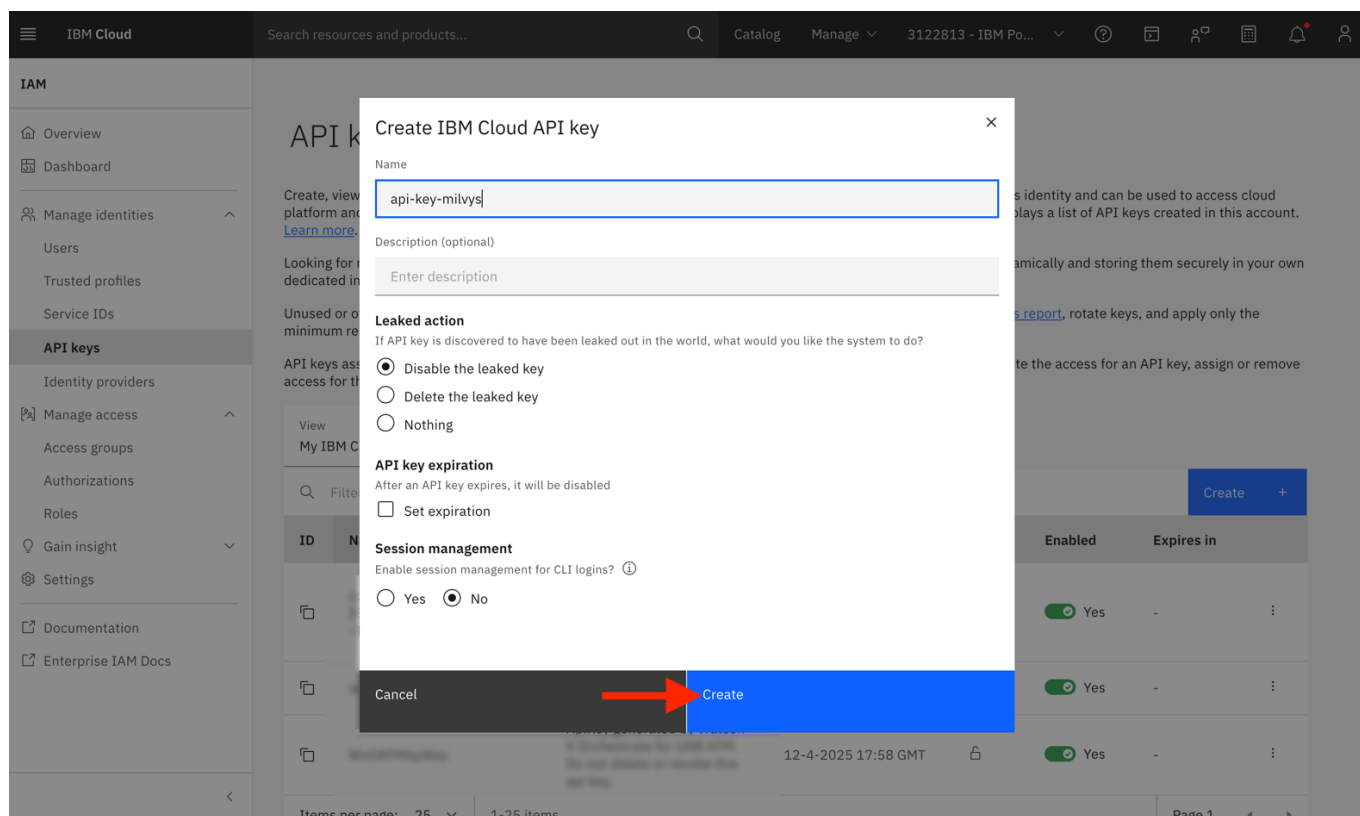
### Session Management (Gerenciamento de Sessão)

Define se esta chave pode ser usada para gerenciamento de sessões no IBM Cloud CLI.

Opção	Descrição
<input type="radio"/> <b>Yes</b>	Permite que a chave seja usada para fazer login no CLI com <code>ibmcloud login --apikey</code>
<input type="radio"/> <b>No</b>	A API Key não poderá ser usada para login em sessões CLI. Use quando desejar restringir a chave apenas para chamadas específicas de API.

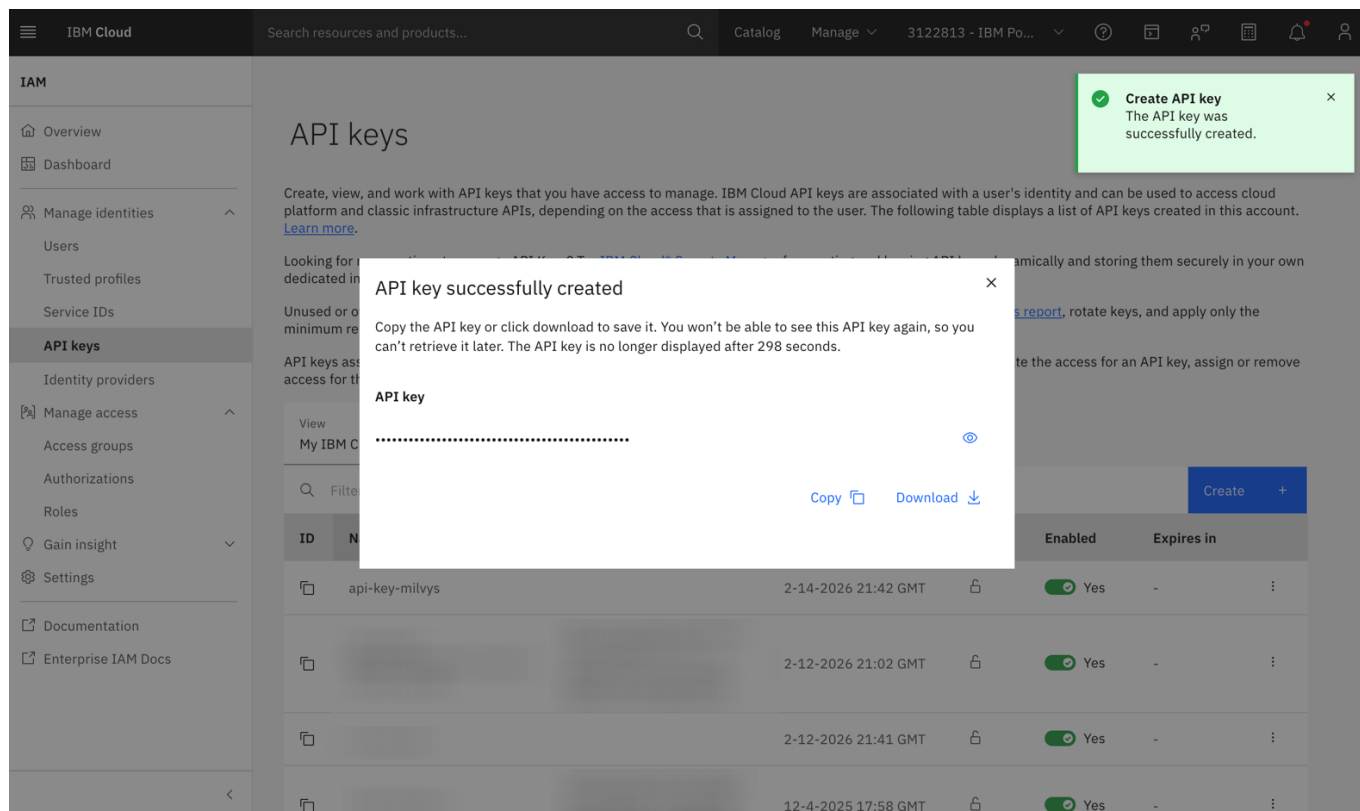


Após preencher tudo, clique em **Create**.



**IMPORTANTE:** Aproveite esse momento para copiar ou fazer o download sua API Key. A chave ficará disponível por alguns segundos. Após esse período, não será mais possível visualizá-la novamente.





Após obter as credenciais, você já possui as duas informações necessárias para a execução da interação via backend com o chat do seu agent (watsonx Orchestrate).

### 1. Navegue até o diretório do projeto:

```
cd ./source/rest_api/
```

### 2. Configure as variáveis de ambiente:

- Altere o nome do arquivo `.env_example` para `.env`
- Preencha com suas credenciais:

```
IAM_API_KEY=sua-api-key-aqui  
WX0_INSTANCE_ID=seu-instance-id-aqui
```

### 3. Instale as dependências:

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

### 4. Execute o servidor:

```
python app.py
```

---

## Gerenciamento de Conversas (Thread ID)

**Importante:** Para manter o contexto da conversa entre múltiplas mensagens:

- **Primeira mensagem:** Faça sua primeira chamada para `/get_response` ou `/stream_response` **sem fornecer** um `thread_id` (deixe vazio ou omita da requisição). A resposta incluirá um valor `thread_id`.
- **Mensagens subsequentes:** Para continuar a conversa no mesmo contexto, inclua o `thread_id` retornado como parâmetro em todas as requisições subsequentes.

Isso permite que o agente mantenha o contexto entre as mensagens.

---

### POST `/get_response`

Envia uma mensagem para um agente e recebe uma resposta única.

**Requisição** (parâmetros de query):

```
POST /get_response?message=Hello&agent_id=your-agent-id[&thread_id=optional-thread-id]
```

#### Parâmetros:

- `message` (obrigatório): A mensagem a ser enviada
- `agent_id` (obrigatório): ID do agente
- `thread_id` (opcional): ID da thread para continuar uma conversa

#### Observações:

- Para a primeira mensagem, **não inclua** `thread_id`
- Para mensagens seguintes, defina `thread_id` com o valor retornado na resposta anterior

#### Resposta:

```
{
  "response": "Agent response here",
  "thread_id": "thread-id-value"
}
```

---

### POST `/stream_response`

Envia uma mensagem para um agente WxO e recebe a resposta de forma contínua (SSE).

**Requisição** (parâmetros de query):

---

```
POST /stream_response?message=Hello&agent_id=your-agent-id[&thread_id=optional-thread-id]
```

### Parâmetros:

- **message** (obrigatório): A mensagem a ser enviada
- **agent\_id** (obrigatório): ID do agente
- **thread\_id** (opcional): ID da thread para continuar uma conversa

### Observações:

- Para a primeira mensagem, **não inclua thread\_id**
- Para mensagens seguintes, defina **thread\_id** com o valor retornado na resposta anterior

### Resposta:

```
Content-Type: text/event-stream  
  
data: {"chunk": "Response", "thread_id": "thread-id-value"}  
  
data: {"chunk": " chunk", "thread_id": "thread-id-value"}  
  
data: {"chunk": " here", "thread_id": "thread-id-value"}
```

Cada bloco de dados do evento é enviado em tempo real conforme o agente processa a resposta.

---

### Variáveis de Ambiente

Variável	Descrição
<b>IAM_API_KEY</b>	Chave de API do IBM Cloud IAM
<b>WXO_INSTANCE_ID</b>	ID da instância do Watsonx Orchestrate

---

### Notas Importantes

#### Atenção:

- Certifique-se de que seu **ID do agente** seja válido
- Sua **chave IAM** deve ter permissões suficientes para acessar a instância
- O endpoint **/stream\_response** usa **streaming (SSE)** para interagir com o agente watsonx Orchestrate processando respostas em tempo real

---

Agora você está pronto para integrar seu aplicativo com o IBM Watson Orchestrate.

**Desenvolvido por Nathalia Trazzi**