

# Conectando o MongoDB Atlas ao Django

Neste guia você realizará o passo a passo para conectar o python ao Atlas. Este último é uma plataforma em nuvem que oferece, como serviço, armazenamento de dados não-relacionais num banco MongoDB.


---

## Criando uma conta no MongoDB Atlas


Entre neste [link](#) para se registrar. Você pode fazer isso utilizando sua conta do Google (que é a forma mais rápida), ou preenchendo o formulário disponível.

**Get started free**

No credit card required

 Sign up with Google

or



8 characters minimum

☐ I agree to the [terms of service](#) and [privacy policy](#).

**Get started free**

Already have an account? [Sign in](#).


# Criando o banco de acordo com o tipo de serviço

Após registrado, você deverá escolher o tipo de serviço. Opte pela opção gratuita, como mostrado a seguir:

## Deploy a cloud database

Experience the best of MongoDB on AWS, Azure, and Google Cloud. Choose a deployment option to get started.

PREVIEW

 **Serverless**


For serverless applications that aren't critical with variable traffic. Minimal configuration required.

- ✓ Pay only for the operations you run
- ✓ Resources scale seamlessly to meet your workload
- ✓ Always-on security and backups

Create

Starting at  
**\$0.30/1M reads**

ADVANCED

 **Dedicated**


For production applications with sophisticated workload requirements. Advanced configuration controls.

- ✓ Network isolation and fine-grained access controls
- ✓ On-demand performance advice
- ✓ Multi-region and multi-cloud options available

Create

Starting at  
**\$0.08/hr\***  
\*estimated cost \$56.94/month

FREE

 **Shared**

For learning and exploring MongoDB in a cloud environment. Basic configuration options.

- ✓ No credit card required to start
- ✓ Explore with sample datasets
- ✓ Upgrade to dedicated clusters for full functionality

Create

Starting at  
**FREE**

[I'll do this later](#)

[Advanced Configuration Options](#)

Agora escolha o provedor e sua região. Somente a AWS e o Google Cloud possuem servidores no Brasil, portanto opte por um dos dois. Depois disso é só clicar em "Create Cluster".

## Cloud Provider & Region

AWS, Sao Paulo (sa-east-1) ▾





★ Recommended region ⓘ  Paid tier region ⓘ

### NORTH AMERICA

 N. Virginia (us-east-1) ★



 Oregon (us-west-2) ★

 Ohio (us-east-2) ★ 

 N. California (us-west-1) 

 Montreal (ca-central-1) 

### SOUTH AMERICA

 Sao Paulo (sa-east-1) 

### EUROPE

 Frankfurt (eu-central-1) ★

 Stockholm (eu-north-1) ★

 Ireland (eu-west-1) ★

 Paris (eu-west-3) ★


 London (eu-west-2) ★ 

 Milan (eu-south-1) ★ 


### MIDDLE EAST

 Bahrain (me-south-1) ★

### AFRICA


 Cape Town (af-south-1) ★


### AUSTRALIA


 Sydney (ap-southeast-2) ★


### ASIA

 Singapore (ap-southeast-1) ★

 Seoul (ap-northeast-2)

 Hong Kong (ap-east-1) ★

 Mumbai (ap-south-1)

 Tokyo (ap-northeast-1) ★

 Osaka (ap-northeast-3) ★ 

**FREE**

Free forever! Your MO cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

[Back](#)

Create Cluster

Caso apareça a tela abaixo, é só clicar em "skip".



Get \$10 in credits for every person  
you invite to your workspace

#### Invite users

You'll receive \$10 in credits for every user you invite who signs up and verifies their email. You can redeem these credits by applying it to any organization where you have admin privileges.

*Promotion limited to 10 unique users or \$100 in credits.*

#### Email Address

#### Roles

Project Owner ▼

Project Owner ▼

Project Owner ▼

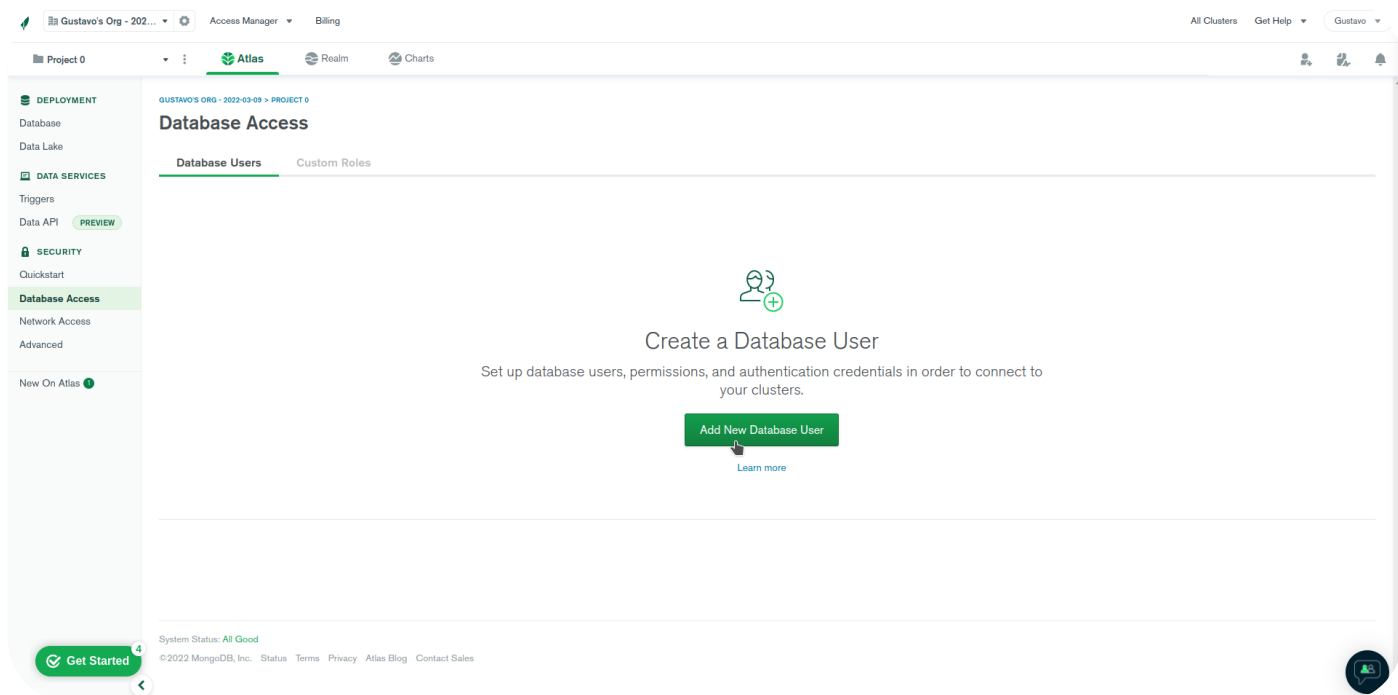
+ Add another user

[Skip](#)

Send and Continue


## Adicionando um usuário para seu banco

Após criado o cluster (que por padrão virá com o nome Cluster0), você terá que adicionar pelo menos um usuário para ter acesso a seu banco de dados. Para isso, selecione a opção "Database Access" à esquerda. Em seguida clique em "Add New Database User".



Crie um nome de usuario e uma senha para ele. Você pode gerar uma automaticamente clicando em "Autogenerate Secure Password". Siga as configurações definidas na imagem abaixo e clique em "Add User".

# Add New Database User

Create a database user to grant an application or user, access to databases and collections in your clusters in this Atlas project. Granular access control can be configured with default privileges or custom roles. You can grant access to an Atlas project or organization using the corresponding [Access Manager](#) 

## Authentication Method

Password

Certificate


AWS IAM  
(MongoDB 4.4 and up)


MongoDB uses [SCRAM](#) as its default authentication method.

## Password Authentication

usuario\_gustavo

SHOW

 Autogenerate Secure Password

 Copy

## Database User Privileges

Configure role based access control by assigning database users a mix of one built-in role, multiple custom roles, and multiple specific privileges. A user will gain access to all actions within the roles assigned to them, not just the actions those roles share in common. **You must choose at least one role or privilege.** [Learn more about roles.](#)

### Built-in Role

1 SELECTED



Select one [built-in role](#) for this user.

Read and write to any database

### Custom Roles



Select your [pre-defined custom role\(s\)](#). Create a custom role in the [Custom Roles](#)  tab.

### Specific Privileges



Select multiple privileges and what database and collection they are associated with.  
Leaving collection blank will grant this role for all collections in the database.

## Restrict Access to Specific Clusters/Data Lakes

Enable to specify the resources this user can access. By default, all resources in this project are accessible.



## Temporary User

This user is temporary and will be deleted after your specified duration of 6 hours, 1 day, or 1 week.

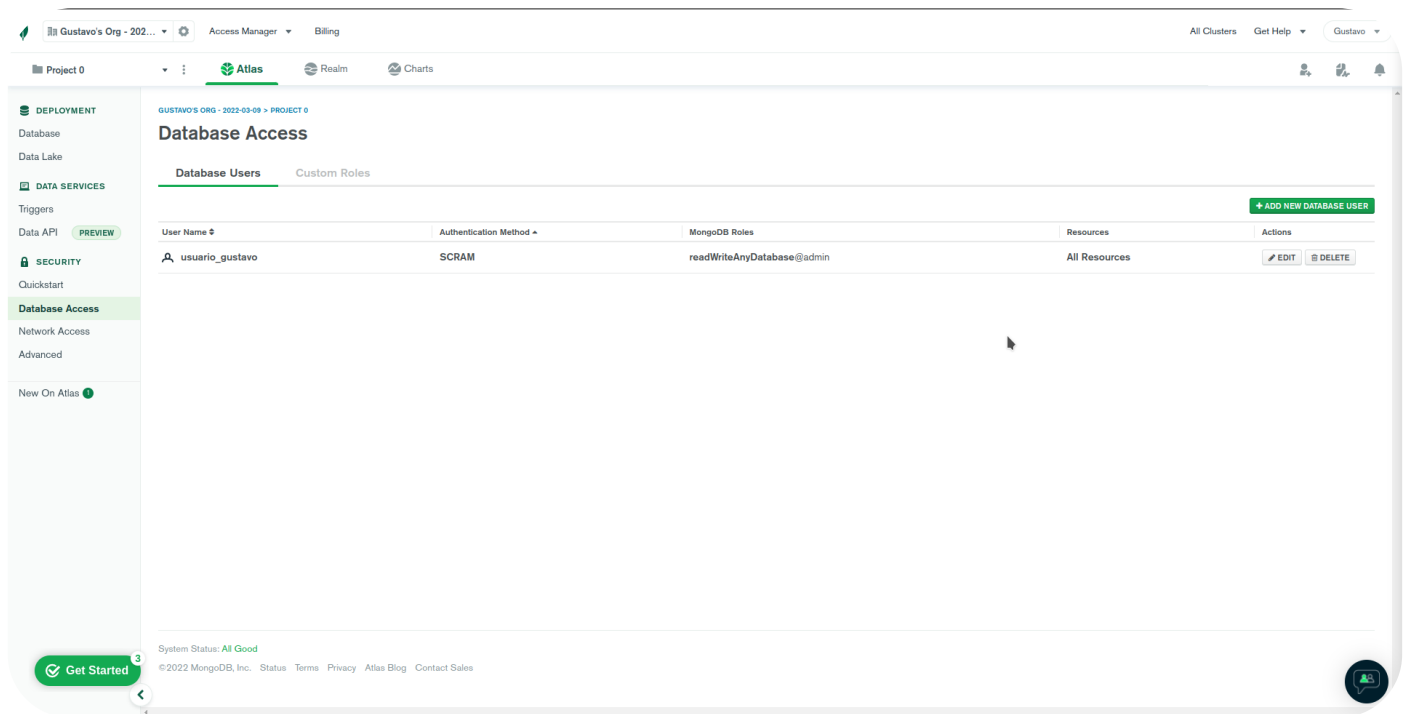


Cancel

Add User



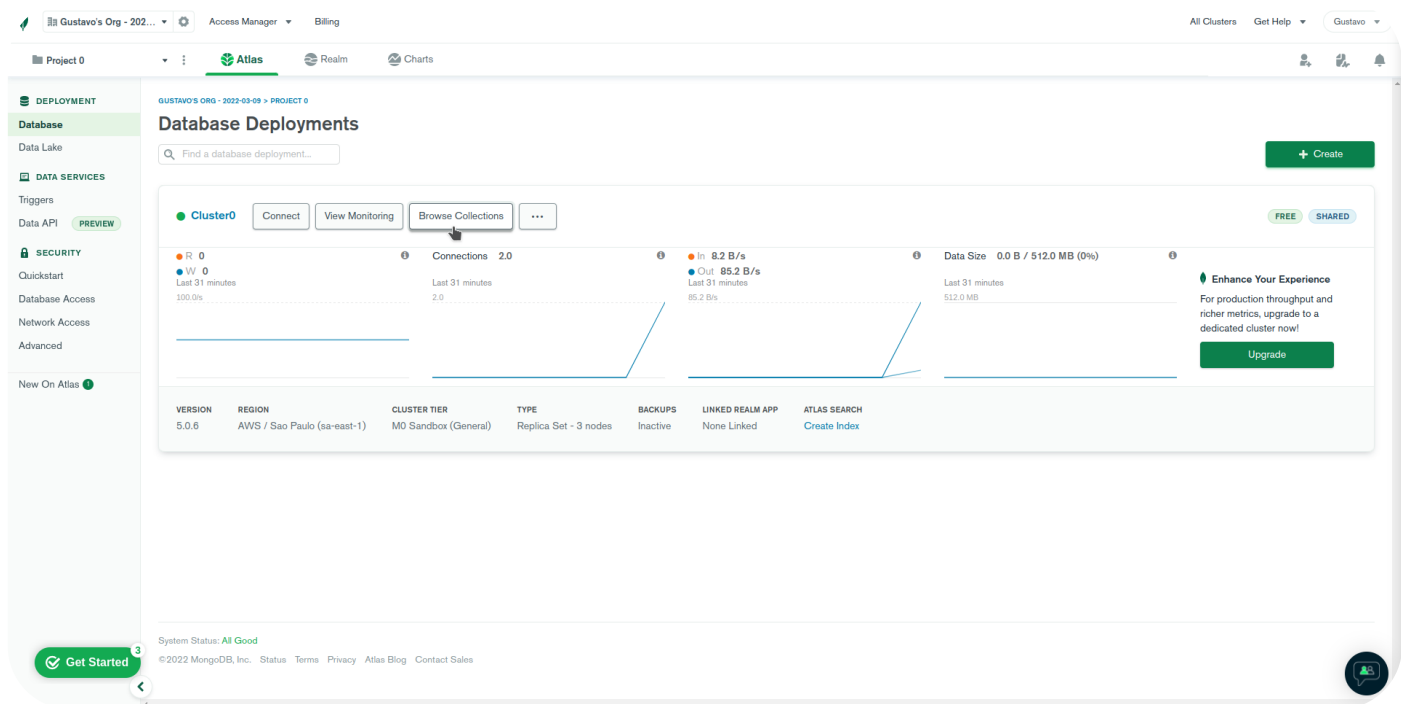
Confira se seu usuário foi criado:



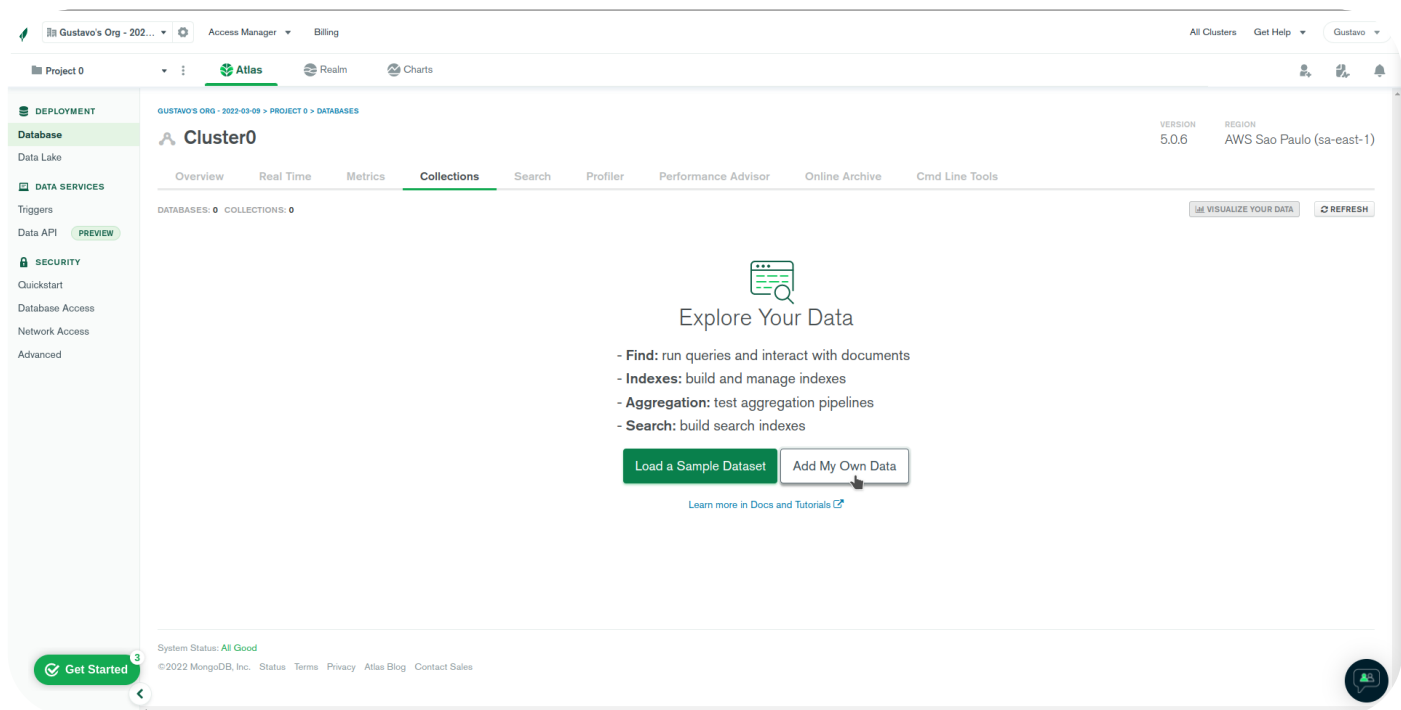
# Criando seu primeiro banco de dados e coleção de documentos em um cluster

Acesse seu cluster selecionando a aba "Database", à esquerda. Depois clique em "Browse Collections":





Clique em "Add My Own Data":



Para fins didáticos, defina seu "Database name" e sua "Collection name" conforme a imagem abaixo. Em seguida, clique em "Create":

# Create Database

Database name ?

meu\_db

Collection name ?

flask\_mongo\_collection

## Additional Preferences

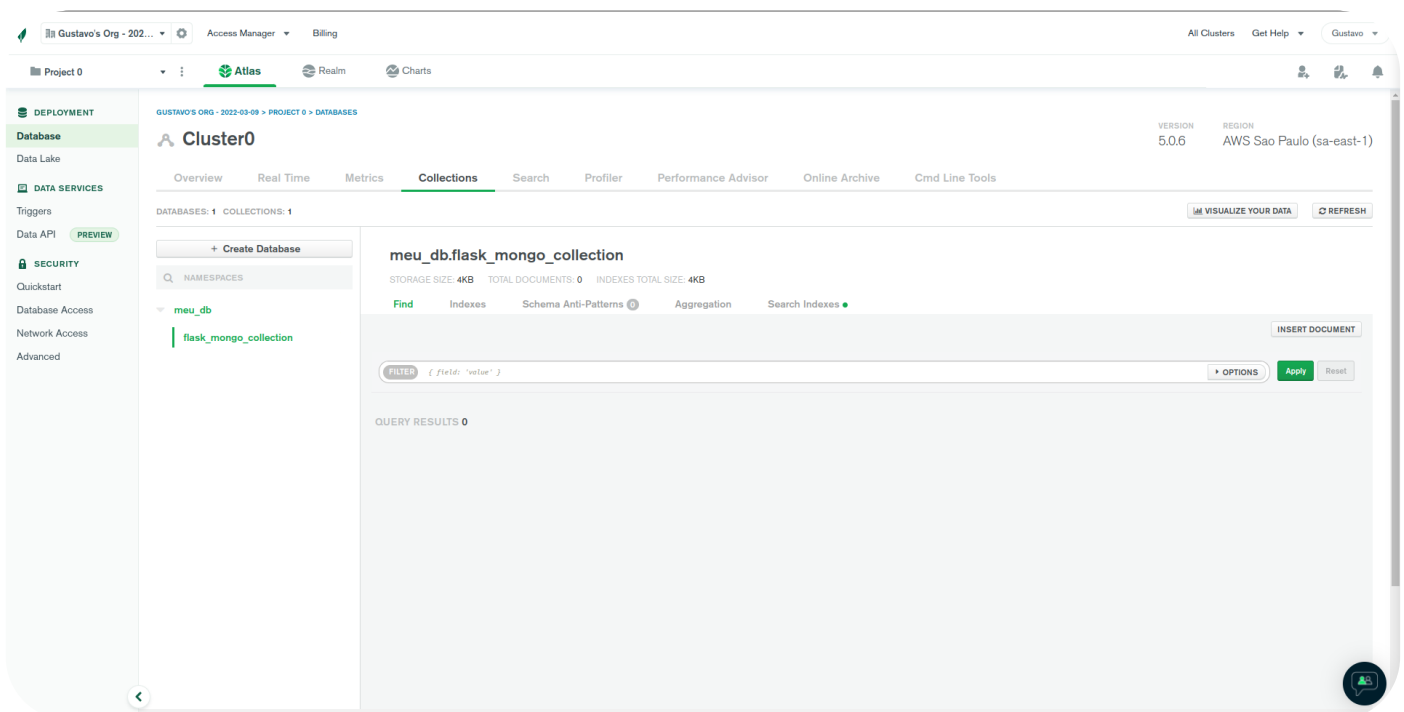
☐ Capped Collection i

☐ Time Series Collection i

Cancel

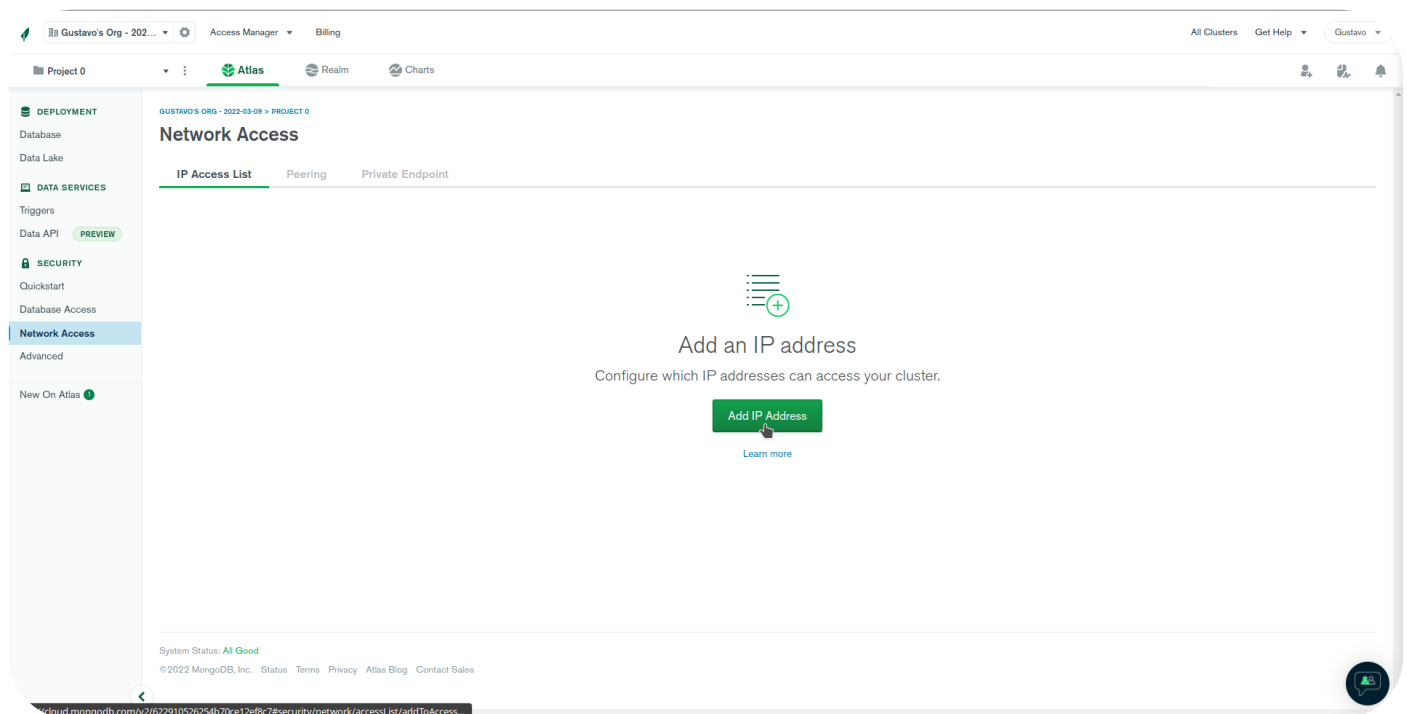
Create

Todos seus bancos e coleções presentes em seu cluster, estarão listados conforme a tela a seguir:



# Configurando os IPs que poderão acessar seu cluster

Vá na aba "Network Access" à esquerda e clique em "Add IP Address":



Adicione seu IP, clicando em "ADD CURRENT IP ADDRESS", para que sua máquina tenha acesso ao cluster. Ou clique em "ALLOW ACCESS FROM ANYWHERE" para permitir que qualquer dispositivo se conecte a seu cluster. Frizando que você pode adicionar um intervalo de IPs manualmente ao preencher o input de "Access List Entry". Depois é só clicar em "Confirm":



## Add IP Access List Entry

Atlas only allows client connections to a cluster from entries in the project's IP Access List. Each entry should either be a single IP address or a CIDR-notated range of addresses. [Learn more.](#)

ADD CURRENT IP ADDRESS

ALLOW ACCESS FROM ANYWHERE

Access List Entry:

0.0.0.0/0

Comment:

Optional comment describing this entry



This entry is temporary and will be deleted in

6 hours

Cancel

Confirm

# Gerando as credenciais para acessar seu cluster

Clique na aba "Database", à esquerda. Depois, em seu cluster, clique em "Connect":

The screenshot displays the MongoDB Atlas 'Database Deployments' page. The left sidebar shows the 'Database' tab selected. The main content area features a table with one cluster, 'Cluster0'. The 'Connect' button is highlighted with a mouse cursor. The cluster details show it is a 'M0 Sandbox (General)' type, with a 'Replica Set - 3 nodes' configuration. The 'Connections' column shows 5.0 connections. The 'Data Size' column shows 4.0 KB / 512.0 MB (0%). The 'Status' column shows 'All Good'.

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED REALM APP	ATLAS SEARCH
5.0.6	AWS / Sao Paulo (sa-east-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	<a href="#">Create Index</a>

## Selecione "Connect your application":

Connect to Cluster0

✓ Setup connection security

Choose a connection method

Connect

Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.



### Connect with the MongoDB Shell

Interact with your cluster using MongoDB's interactive Javascript interface



### Connect your application

Connect your application to your cluster using MongoDB's native drivers



### Connect using MongoDB Compass

Explore, modify, and visualize your data with MongoDB's GUI



Go Back

Close

Selecione o driver e sua versão de acordo com a imagem abaixo. Depois, copie o endereço de acesso (credenciais) gerado:

✓ Setup connection security

✓ Choose a connection method

Connect

1 Select your driver and version

DRIVER

Python

VERSION

3.6 or later

2 Add your connection string into your application code

☐ Include full driver code example

```
mongodb+srv://usuario_gustavo:
<password>@cluster0.vn984.mongodb.net/myFirstDatabase?retryWrites=true&w=majority
```



Replace **<password>** with the password for the **usuario\_gustavo** user. Replace **myFirstDatabase** with the name of the database that connections will use by default. Ensure any option params are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

## Conectando o django ao Atlas

- Instale o driver **django** com `pip install django`. Ele fará com que o django permita o uso do MongoDB como banco de dados
- Com o **django**, já vem instalado o **pymongo** em sua versão mais recente (4.1.1).
- Para evitar incompatibilidades com as migrations, devemos desinstalar essa versão do pymongo com `pip uninstall pymongo`

- Em seguida, rode `pip install pymongo==3.12.1` para instalar o pymongo na sua versão 3.12.1

Em **settings.py**, configure as credenciais de conexão ao banco de dados:

Copiar para área de transferência

```
DATABASES = {  
    'default': {  
        'ENGINE': 'djongo',  
        'NAME': 'meu_db',  
        'HOST': "mongodb+srv://usuario_gustavo:ZkvR",  
        'USER': 'usuario_gustavo',  
        'PASSWORD': 'ZkvRlGPyaJ6B87',  
    }  
}
```

Pronto! Agora é só rodar as migrations e sua aplicação estará pronta para uso.

## Referências!

