

Atividade Final

Arquitetura Cloud

Pipeline de Dados

Streaming de Dados





PetAdota: Escalando a Plataforma de Adoção

A PetAdota está crescendo rapidamente e precisa de uma infraestrutura escalável, segura e automatizada para suportar a crescente demanda de usuários que buscam adotar cães e gatos.



Dog API: <https://dog.ceo>



Cat API: <https://api.thecatapi.com>



Objetivos do Projeto:



✓ Criar e gerenciar infraestrutura na nuvem de forma automatizada.



✓ Provisionar máquinas virtuais, bancos de dados e buckets para armazenar imagens e dados dos animais disponíveis para adoção.



✓ Configurar servidores automaticamente para garantir alta disponibilidade.



✓ Implementar um pipeline de dados e um sistema de streaming para processar interações de usuários em tempo real.



Ferramentas e Tecnologias



GitHub – Versionamento do código.



Terraform – Provisionamento de infraestrutura.



Ansible – Automação de servidores.



GCP – Hospedagem da infraestrutura.



S3 – Armazenamento de imagens.



Kafka – Streaming de eventos de interação.



Apache Spark Streaming – Processamento de dados em tempo real.



Airflow – Automação e monitoramento contínuo.




MySQL – Armazenamento dos dados dos animais disponíveis.

IMPORTANTE

OBS: Os seus códigos devem ser entregues no colab, porém é imprescindível enviar a evidência que este projeto foi publicado no seu github. Sem isso a nota será considerada apenas metade do informado no título da questão.


Provisionamento de Infraestrutura Automatizada (3)

 **Objetivo:** Criar e gerenciar a infraestrutura da PetAdota de forma automatizada usando **Terraform** e **Ansible**.

Tarefas do Exercício:

- ✓ **Criar um bucket no GCP** para armazenar imagens dos animais.
- ✓ **Provisionar uma máquina virtual Linux** para hospedar o banco de dados.
- ✓ **Instalar e configurar um banco MySQL** na VM automaticamente via **Ansible**.
- ✓ Criar o banco de dados **Pet_Adota** e a tabela **animais_disponiveis** com as colunas:
 - id (identificador único)
 - raca (raça do animal)
 - url (URL da imagem armazenada no S3)
 - tipo (cachorro ou gato)
- ✓ Criar um **usuário de consulta** que pode ler os dados mas não alterar.
- ✓ **Validar a estrutura do banco** acessando via MySQL console.

Pipeline de Imagens da API para o Banco (3)

 **Objetivo:** Construir um pipeline automatizado para **baixar imagens de animais, armazenar no S3 e registrar no banco MySQL.**

Tarefas do Exercício:

- ✓ **Consultar as APIs de cães e gatos** (Dog API e Cat API).
- ✓ **Baixar as imagens** e nomeá-las com um contador por raça (labrador_01.jpg).
- ✓ **Salvar as imagens no S3**, organizando por **raça e tipo**:
 - s3://pet-adota-imagens/cachorro/labrador/labrador_01.jpg
 - s3://pet-adota-imagens/gato/persa/persa_01.jpg
- ✓ **Registrar os metadados no banco MySQL** (Pet_Adota > animais_disponiveis).
- ✓ **Criar um DAG no Airflow** para rodar esse pipeline a cada 30 segundos.



Processamento de Interações de Usuários em Tempo Real (4)

 **Objetivo:** Criar um sistema de **monitoramento de tendências de adoção** em tempo real, utilizando **Kafka e Spark Streaming**.

Tarefas do Exercício:

✓ **Simular eventos de interação** de usuários clicando nos perfis de animais no site.

✓ **Publicar eventos no Kafka**, contendo:

- usuario_id (quem interagiu)
- raca (raça do animal visualizado)
- tipo (cachorro ou gato)
- timestamp (horário da interação)

✓ **Processar os eventos com Spark Streaming**, identificando tendências:





- Animais mais procurados no último minuto.
- Horário de pico de visualizações.

✓ **Salvar estatísticas no banco MySQL** para análise futura.


✓ Criar um **DAG no Airflow** para visualizar as tendências **a cada 30 segundos**

Material de apoio



1 CI/CD – Terraform + Ansible

-  Terraform
-  Ansible
-  Google Cloud Compute Engine
-  Google Cloud Storage (Buckets)




2 Pipeline de Dados – API de Imagens + S3 + MySQL

-  Dog API (Cachorros)
-  Cat API (Gatos)
-  [AWS S3](#)
-  [MySQL](#)
-  [Apache Airflow](#)

3 Streaming de Dados – Kafka + Spark Streaming

-  [Apache Kafka](#)
-  [Apache Spark Streaming](#)
-  [MySQL - Real-Time Processing](#)
-  [Airflow para Streaming](#)

Outros Recursos Úteis

-  Python Requests (Para acessar APIs)
-  Boto3 (SDK AWS para S3)
-  SQLAlchemy (Conexão com MySQL)

The background is a solid light pink color. It features several large, white, organic, wavy shapes that resemble liquid droplets or clouds. There are four white circles of different sizes: one in the upper left, one in the upper right, one in the lower right, and one in the lower left. The word "Obrigado" is centered in the white space.

Obrigado