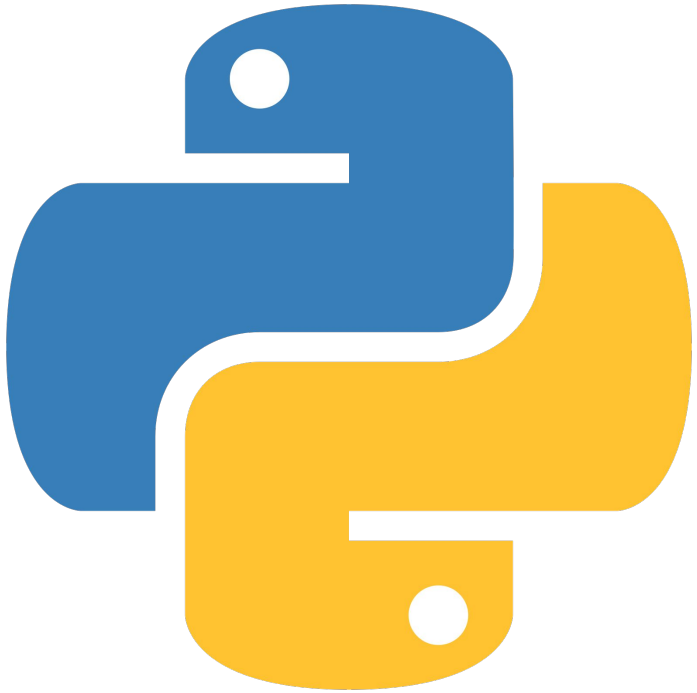
Case Engenharia de Dados

Samuel Tovo

Teste Python

- Para acessar o teste, clique aqui: Teste

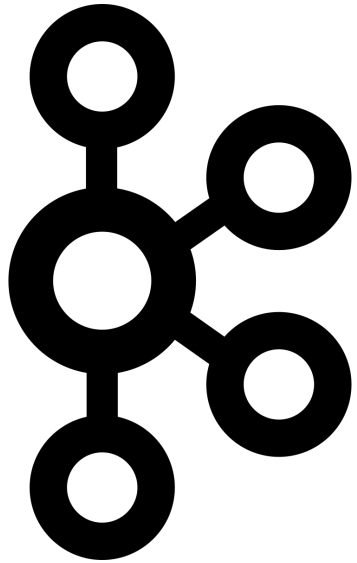


Arquitetura Data Lake

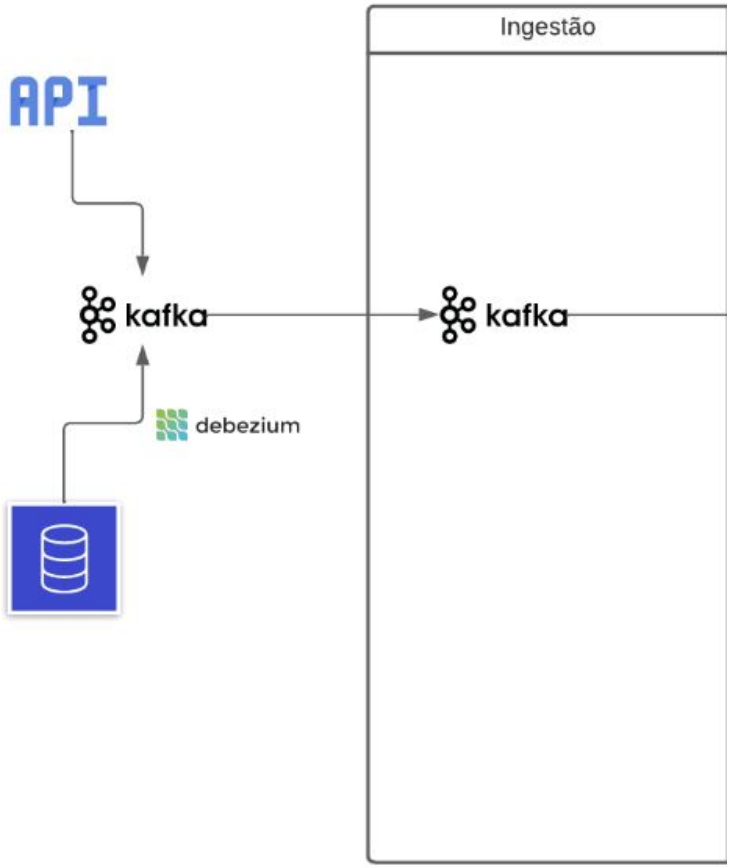
A arquitetura do Data Lake foi principalmente desenvolvida sobre a premissa de um fluxo em Real Time, baseado no Google Cloud Storage com uma camada do Delta Lake.

Ingestão

- Dentro da camada de ingestão será usada, para os bancos de dados relacionais, o Debezium, que enviará as mudanças dos bancos para o Kafka. Para dados de logs e mensageria, será conectada diretamente ao Apache Kafka com as APIs.
- Para conectar ao Google Cloud Storage, será usado o Kafka Connect.



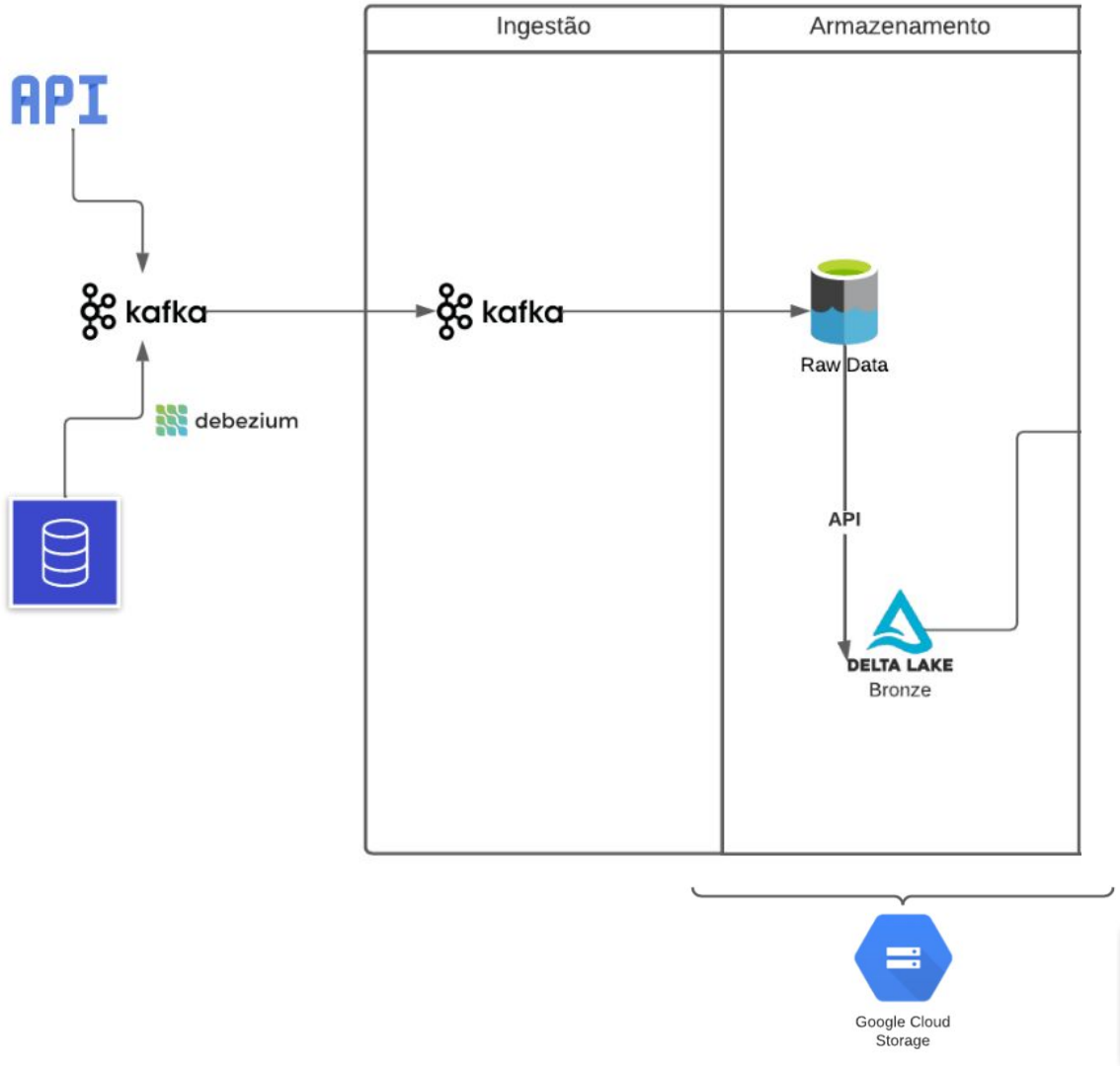
Ingestão



Armazenamento

- Para o armazenamento, será separado em 2 camadas: bronze (em Armazenamento) e silver (em Processamento). Aqui será utilizado o Google Cloud Storage. Junto a isso, será utilizado também o Delta Lake para adicionar uma camada mais inteligente de metadados, para governança e manutenção da qualidade de dados, além de trazer o melhor do data lake e do data warehouse, unificando tudo em um único local, o que facilita o acesso de dados para projetos de Data Science.
    - A camada bronze será alimentada pelo Kafka, sendo ela uma camada "raw", ou seja, sem processamento algum, essa camada é necessária para que não se perca nenhum dado no processo;
        - Com isso, os dados serão conectados ao Delta Lake com o Delta Lake API;

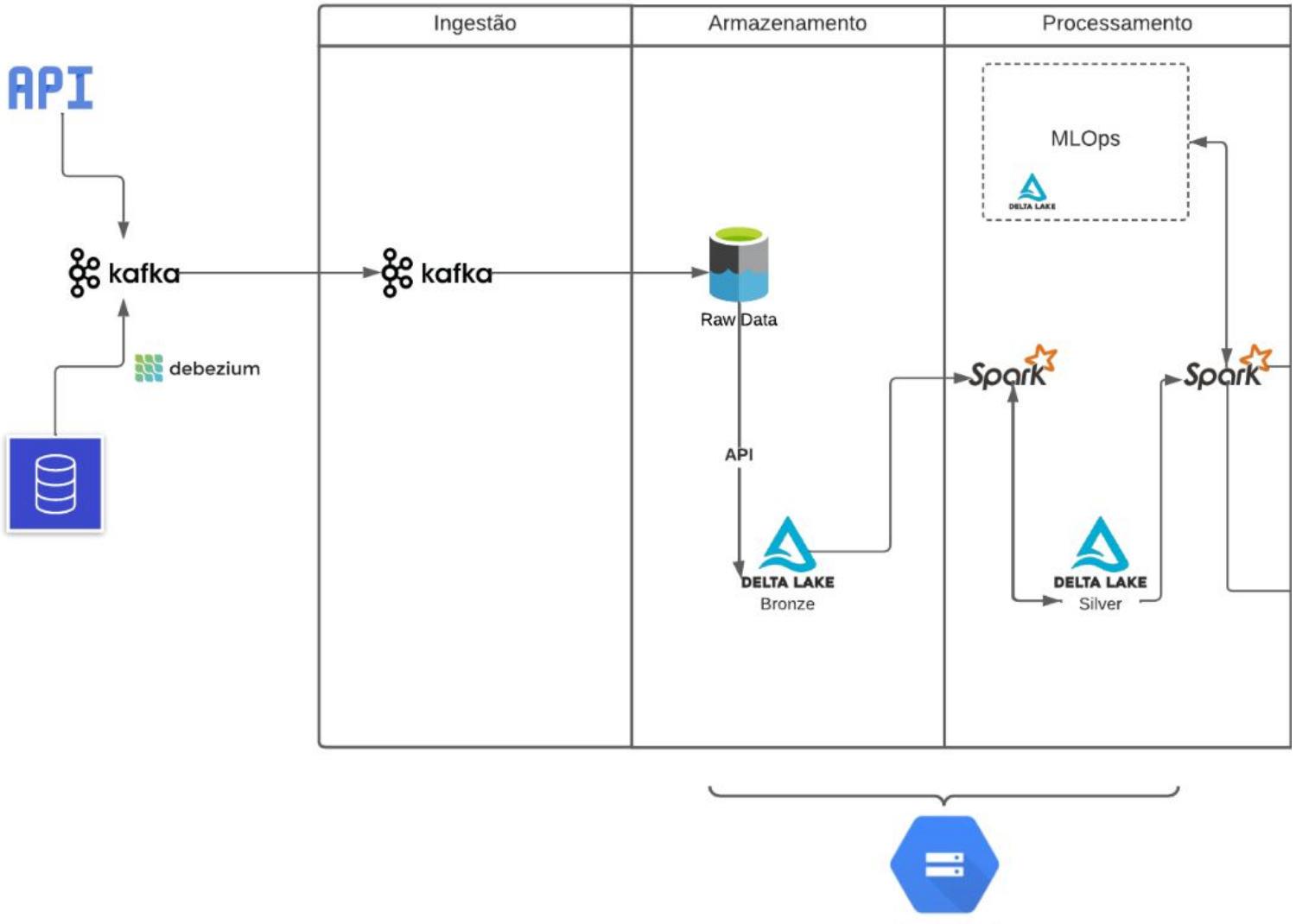
Armazenamento



Processamento e Treinamento de Modelos

- Processamento dos Dados
  - A camada silver será o processamento dos dados, utilizando o Spark Streaming, com o propósito de disponibilizar dados de alta qualidade para diversos tipos de aplicação em real time, como modelos de machine learning e ter tabelas disponíveis para uso posterior do time de business intelligence;
- Treinamento de modelos
  - A camada silver do Delta Lake permite uso de MLOps pelo time de Data Science, já que o Delta Lake foi desenhado para funcionar muito bem com frameworks de machine learning.

Processamento e Treinamento de Modelos

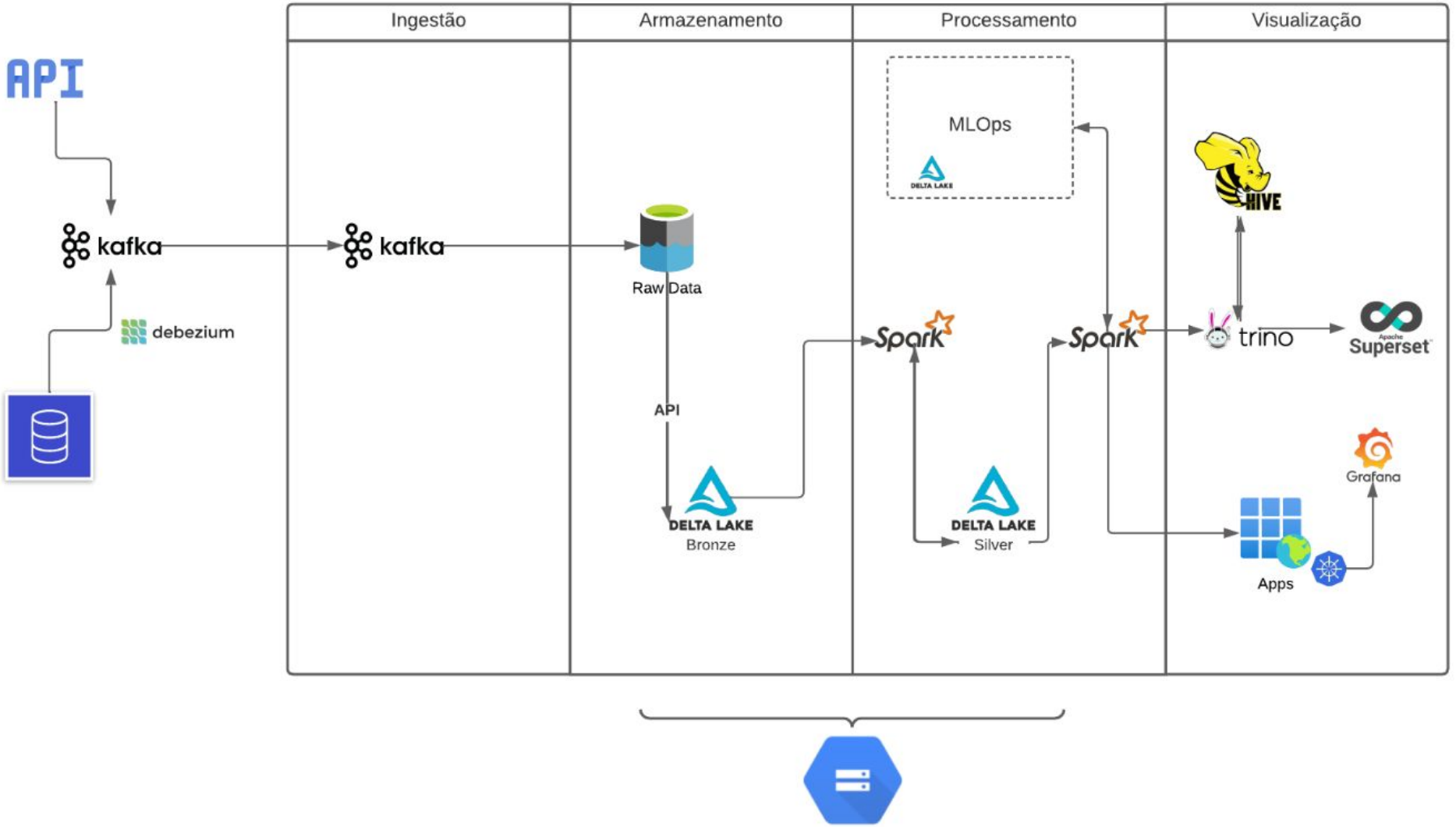


Data Analytics, Aplicações e Visualização

- Data Analytics e Aplicações
  - Trino para analytics e criações de relatórios para a empresa. Junto a ele, para que os metadados criados sejam guardados, o Apache Hive entraria para isso. Alem disso, o Trino permite analises em real time.
  - Aplicações da empresa utilizando Python e Java direto do Delta Lake, com o metadado e logs das aplicações sendo adicionadas a tabelas, utilizando, por exemplo, pods Kubernetes para execução.

- Visualização
  - Superset, recebendo novas tabelas de analytics e data science do Trino.
  - Monitoramento:No Grafana, com os metadados das aplicações, é possível observar o uso de CPU e memória para possíveis melhorias, conseguindo um monitoramento eficiente. Além disso, é possível configurar alertas baseados em condições dos dados, com isso, pode notificar o time de Data Engineering quando esses eventos ou anomalias acontecem.

Data Analytics, Aplicações e Visualização



Data Governance e Data Quality

- Governança - É importante frisar que com o Delta Lake Metadata API se torna mais fácil definir e categorizar as tabelas do framework, além de facilitar o acesso ao metadado que fica armazenado em tabelas do próprio Delta Lake.

- Qualidade - Com o Delta Lake é possível automatizar a checagem de schema, e a própria limpeza do dado, como retirar duplicatas, campos vazios ou qualquer outro problema com a qualidade dos dados.

Pontos fracos e fortes

- Fracos:
  - O uso do kafka traz mais complexidade e custos num longo prazo, partindo da premissa de um crescimento dos dados da empresa, o Kafka necessita de muito poder computacional em uma grande escala de dados
  - O uso do Delta Lake pode trazer uma camada mais de complexidade para o fluxo de dados. Além disto, apesar das grandes vantagens de seus logs para o metadata, ira ocupar ainda mais espaço da cloud, agregando custo ao processo. Alem disso, quando adicionada a camada, se adiciona um pouco mais de latência ao processo, quando comparado ao uma solução apenas de Streaming.
  - O Spark Streaming, utiliza micro batch, portanto, quando comparado a outras ferramentas de processamento em streaming acaba tendo um pouco mais de latencia.
  - O Trino é uma ferramenta de query em memória, o que significa que ao processar uma grande quantidade de dados, será necessario alocar mais memória para que não haver problemas.

Pontos fracos e fortes

- Fortes:
  - Dentro da solução, sendo ela de real-time, o Kafka se torna uma ferramenta excelente para lidar com muitos producers, com escalabilidade e flexibilidade para integração.

  - O Delta Lake adiciona muitas vantagens ao uso de um data lake, a camada de metadata auxilia todo o processamento dos dados, como por exemplo, com sua execução do Schema, sempre garantindo a consistencia dos dados, possui trasações ACID (onde o dado é indivisível, consistente, idenpendente e durável), além disso possui time travel, garantindo a visualização de tabelas do passado. Como um todo a arquitetura gira em torno do Delta Lake, pois ele facilita o acesso e controle dos dados e tem uma boa flexibilidade para o Spark usado nele.

  - O Spark Streaming possui grande relação com Spark, o que integra a ele muitos modulos e bibliotecas, e com isso se torna uma opção mais acessivel para manutenção no dia-a-dia. Além disso, tem grande escalabilidade e tolerância a falha.

  - O Trino aqui entra com o objetivo de usar sua alta performance, junto ao Delta Lake, como uma ultima camada de dados, ao invés de usar uma Camada "Gold" dentro do Data Lake, para facilitar o uso de SQL para equipes como Data Analytics, criação de tabelas finais para dos times de Data Science, e relatórios para serem observados posteriormente no SuperSet

Obrigado!

Samuel Gatti Tovo

+55 16 981176429

samueltovo@outlook.com