

# **Qualidade, Governança e Arquitetura de Dados**

## **Problemas de qualidade de dados identificados**

Problema 1: Inconsistência entre a data de registro e a primeira transação.

Há um grande volume de clientes que possuem data de registro maior que a data da primeira transação. Isso significa que o cliente realizou uma compra antes mesmo de estar registrado no sistema.

Exemplo:

Customer ID First	Registration Date	Transaction Date	Diferença
1	18/02/2024	16/11/2022	15 meses

Problema 2: Clientes sem transações

Existe em torno de 1000 clientes na base que não possuem nenhum tipo de transação. Isso pode significar que os clientes estão inativos ou há uma sujeira na base.

Duplicação de números de telefone

Existem cerca de 274 clientes que compartilham o mesmo número de telefone. Isso pode impactar em campanhas de marketing se isso tratar de um erro de registro.

# **Proposta de Arquitetura de Integração e Governança de Dados**

## **CAMADA DE INGESTÃO**

Origem da conexão dos dados, seja dados de um CRM/ERP, e-commerce ou arquivos brutos

## **CAMADA DE PROCESSAMENTO**

Ferramenta de ETL para fazer a ingestão de dados para uma base de dados. Podemos trabalhar com ferramentas low-code como data factory ou linguagens python, spark dentro outros para fazer as conexões e extrações dos dados de outros ambientes, seja através do databricks, Microsoft fabric ou outra plataforma.

## **CAMADA DE ARMAZENAMENTO**

Inserção em um datawarehouse ou datalake (Microsoft Fabric, SQL Server, BigQuery, etc..).

Aqui podemos trabalhar com camadas. Dados brutos (bronze), dados limpos (silver) e dados agregados (gold).

## **CAMADA DE GOVERNANÇA**

Plataformas que ofereçam uma governança dos dados, como por exemplo o Microsoft Perview. Neste caso vamos conseguir ter um catálogo dos dados, rastreamento da origem dos dados, dependência entre os dados, gerenciamento de acessos, entre outros.

## Dicionário de Dados

Tabela “Customers”

Campo	Tipo	Nulo?	Descrição	Validação	Exemplo
customer_id	INTEGER	Não	Identificador do cliente	PK> 0	1
age	INTEGER	Não	Idade do cliente	18 ≤ age ≤ 120	35
gender	VARCHAR(10 )	Não	Gênero do cliente	Feminino, Masculino ou Outros	‘Feminino’
city	VARCHAR(10 )	Não	cidade	Não nulo	‘São Paulo’
registration_date	Date	Não	Data de registro do cliente	<= data de hoje	2021-01-15
phone_number	VARCHAR(20 )	Não	Número de telefone	15 dígitos	‘11987654321’

Tabela “Produtos”

Campo	Tipo	Nulo?	Descrição	Validação	Exemplo
product_id	INTEGER	Não	Categoria do produto	PK> 0	1
category	VARCHAR(50)	Não	Idade do cliente	‘Moda’, ‘Esportes’, ‘Casa’, ‘Eletrônicos’, ‘Livros’	‘Moda’
price	DECIMAL(10,2 )	Não	Preço unitário do produto	R\$ > 0, ≤ 100000	99.99
partnership	VARCHAR(50)	Não	Parceiro responsável pelo produto	Não nulo	‘Partner_A’

Tabela “Transactions”

Campo	Tipo	Nulo?	Descrição	Validação	Exemplo
transaction_id	BIGINT	Não	Identificador único da transação	PK>0	1000001
customer_id	INTEGER	Não	ID do cliente que realizou a compra	FK - customers.customer_id	1
product_id	INTEGER	Não	ID do produto comprado	FK - products.product_id	1
transaction_date	TIMESTAMP	Não	Data e hora da transação	Não futuro, $\geq$ que 01/01/2021	2023-06-15 14:30:45
quantity	INTEGER	Não	Quantidade de unidades compradas	$1 \leq \text{qtde} \leq 1000$	5
price	DECIMAL(, )	Não	Preço unitário no momento da compra	>0	99.99
total_value	DECIMAL(, )	Não	Valor total da transação (quantity $\times$ price)	$> 0, \leq 1000000$	299.97