

## Projeto CRM de Leads

---

### 1. Introdução

Este documento apresenta o projeto de um banco de dados para gerenciamento de leads, permitindo o registro de interações (e-mail, WhatsApp e ligação), o acompanhamento de estágios e a geração de relatórios que dão suporte a decisões de vendas.

---

### 2. Objetivo

Criar um modelo de dados simples e eficiente que cubra:

- Cadastro de leads (contendo nome, e-mail, telefone).
  - Registro de interações, associando cada ação a um usuário e canal.
  - Evolução de cada lead por estágios pré-definidos (Novo → Contato Realizado → ... → Fechado).
  - Estrutura flexível para relatórios analíticos e automações de follow-up.
- 

### 3. Situação Problemática

Atualmente não existe centralização das informações de contato e histórico de interações. Isso gera:

- Perda de follow-up por ausência de alerta de prazos.
  - Dificuldade em medir produtividade dos vendedores.
  - Retrabalho e inconsistência de dados.
- 

### 4. Modelo de Negócio

Usuários do sistema:

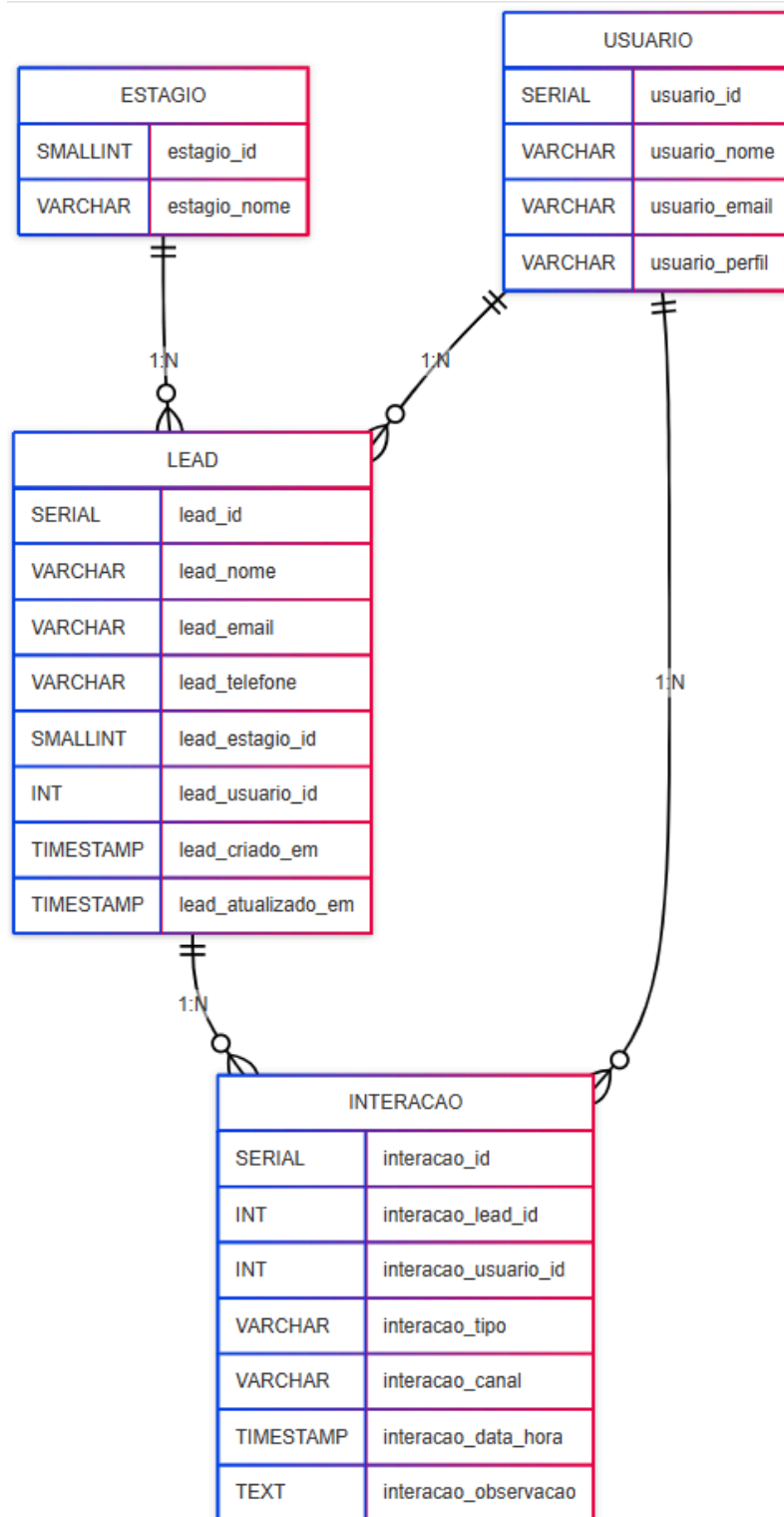
- **Vendedores:** registram e atualizam status dos leads.

- **Supervisores:** acompanham métricas, distribuem carteira e validam movimentações.

Fluxo básico:

1. Recebimento/Importação de um lead → estágio “Novo”.
2. Registro de cada interação → e-mail, WhatsApp ou ligação.
3. Atualização automática/manual do estágio conforme regras de negócio.
4. Supervisão e geração de relatórios para tomada de decisão.

## 5. Diagrama Entidade-Relacionamento



- **ESTAGIO 1:N LEAD**

Cada lead possui exatamente um estágio; um estágio pode agrupar vários leads.

- **USUARIO 1:N LEAD**  
Um usuário (vendedor ou supervisor) é responsável por vários leads.
  - **LEAD 1:N INTERACAO**  
Um lead pode ter diversas interações registradas.
  - **USUARIO 1:N INTERACAO**  
Cada interação é registrada por um único usuário.
- 

## 6. Listagem de Tabelas

- **estagio**
  - estagio\_id (SMALLINT, PK, NOT NULL)
  - estagio\_nome (VARCHAR(50), UNIQUE, NOT NULL)
- **usuario**
  - usuario\_id (INT, PK, AUTO\_INCREMENT, NOT NULL)
  - usuario\_nome (VARCHAR(100), NOT NULL)
  - usuario\_email (VARCHAR(100), UNIQUE, NOT NULL)
  - usuario\_perfil (VARCHAR(20), NOT NULL)
- **lead**
  - lead\_id (INT, PK, AUTO\_INCREMENT, NOT NULL)
  - lead\_nome (VARCHAR(100), NOT NULL)
  - lead\_email (VARCHAR(100), UNIQUE, NOT NULL)
  - lead\_telefone (VARCHAR(20), NULL)
  - lead\_estagio\_id (SMALLINT, FK→estagio.estagio\_id, NOT NULL)
  - lead\_usuario\_id (INT, FK→usuario.usuario\_id, NOT NULL)
  - lead\_criado\_em (DATETIME, NOT NULL, DEFAULT CURRENT\_TIMESTAMP)

- lead\_atualizado\_em (DATETIME, NULL)
  - **interacao**
    - interacao\_id (INT, PK, AUTO\_INCREMENT, NOT NULL)
    - interacao\_lead\_id (INT, FK→lead.lead\_id, NOT NULL)
    - interacao\_usuario\_id (INT, FK→usuario.usuario\_id, NOT NULL)
    - interacao\_tipo (VARCHAR(20), NOT NULL)
    - interacao\_canal (VARCHAR(20), NOT NULL)
    - interacao\_data\_hora (DATETIME, NOT NULL, DEFAULT CURRENT\_TIMESTAMP)
    - interacao\_observacao (TEXT, NULL)
- 

## 7. Link para o Script SQL

[https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/create\\_crm\\_leads.sql](https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/create_crm_leads.sql)

---

## 8. Lista de Views

### 8.1 view\_leads\_por\_estagio

- **Descrição:** retorna, para cada estágio, a quantidade total de leads atualmente posicionados nele.
- **Objetivo:** fornecer um panorama do funil de vendas em tempo real.
- **Tabelas envolvidas:**
  - lead (lead\_estagio\_id)
  - estagio (estagio\_nome)

### 8.2 view\_interacoes\_por\_lead

- **Descrição:** lista todas as interações registradas para cada lead (tipo, canal,

data/hora e usuário).

- **Objetivo:** oferecer um histórico completo de contato por lead.
- **Tabelas envolvidas:**
  - interacao (interacao\_tipo, interacao\_canal, interacao\_data\_hora, interacao\_observacao, interacao\_lead\_id, interacao\_usuario\_id)
  - lead (lead\_nome)
  - usuario (usuario\_nome)

### 8.3 view\_leads\_sem\_interacao

- **Descrição:** identifica todos os leads que ainda não possuem nenhuma interação registrada.
- **Objetivo:** sinalizar leads que precisam de contato inicial.
- **Tabelas envolvidas:**
  - lead
  - interacao

### 8.4 view\_atividade\_por\_usuario

- **Descrição:** agrupa o total de interações realizadas por cada usuário, com opção de filtro por período.
- **Objetivo:** medir a produtividade de vendedores/supervisores.
- **Tabelas envolvidas:**
  - interacao (interacao\_usuario\_id, interacao\_data\_hora)
  - usuario (usuario\_nome)

---

## 9. Lista de Funções

### 9.1 fn\_tempo\_em\_estagio(p\_lead\_id INT)

- **Descrição:** calcula o número de dias desde que o lead foi criado (início no

estágio atual).

- **Objetivo:** medir a velocidade de passagem dos leads pelo funil.
- **Tabelas/Colunas envolvidas:**
  - lead.lead\_criado\_em
  - lead.lead\_id

## 9.2 fn\_total\_interacoes(p\_lead\_id INT)

- **Descrição:** retorna o total de interações registradas para um lead.
- **Objetivo:** avaliar o engajamento de follow-up.
- **Tabelas/Colunas envolvidas:**
  - interacao.interacao\_lead\_id

## 9.3 fn\_percentual\_conversao(p\_estagio\_origem INT, p\_estagio\_destino INT)

- **Descrição:** calcula a taxa de conversão de leads de um estágio para outro.
  - **Objetivo:** monitorar eficiência entre fases do funil.
  - **Tabelas/Colunas envolvidas:**
    - lead.lead\_estagio\_id
- 

# 10. Lista de Stored Procedures

## 10.1 sp\_inserir\_dados\_teste()

- **Descrição:** popula as tabelas com dados de exemplo em estagio, usuario, lead e interacao.
- **Objetivo:** facilitar testes e validações, criando um cenário inicial de dados.
- **Tabelas envolvidas:** estagio, usuario, lead, interacao

## 10.2 sp\_atualizar\_estagios\_por\_interacao(p\_lead\_id INT)

- **Descrição:** atualiza o campo lead\_estagio\_id de um lead com base no tipo de



sua última interação.

- **Objetivo:** automatizar a evolução do lead pelo funil de vendas conforme o contato registrado.
- **Tabelas envolvidas:** interacao, lead

### 10.3 sp\_gerar\_relatorio\_pipeline()

- **Descrição:** gera um result set com contagem de leads por estágio e percentual de cada estágio sobre o total.
  - **Objetivo:** fornecer um relatório sintético para dashboards e análise do pipeline de vendas.
  - **Tabelas envolvidas:** lead, estagio
- 

## 11. Lista de Triggers

### 11.1 trg\_atualiza\_timestamp\_lead

- **Evento:** AFTER INSERT em interacao
- **Descrição:** ao inserir uma nova interação, atualiza lead\_atualizado\_em na tabela lead.
- **Objetivo:** manter registro da última atividade do lead sempre atualizado.

### 11.2 trg\_validar\_transicao\_estagio

- **Evento:** BEFORE UPDATE em lead
  - **Descrição:** impede que o lead mude de estágio de forma não sequencial.
  - **Objetivo:** reforçar as regras de negócio do funil de vendas.
- 

## 12. Inserção / Importação de Dados

### 12.1 Insert Manual de Dados de Teste

- Utilize a stored procedure sp\_inserir\_dados\_teste() para povoar rapidamente todas as tabelas com dados de exemplo.

## 12.2 Importação via CSV

- Arquivos:
    1. data/leads.csv (colunas: lead\_nome, lead\_email, lead\_telefone, lead\_estagio\_id, lead\_usuario\_id)
    2. data/interacoes.csv (colunas: interacao\_lead\_id, interacao\_usuario\_id, interacao\_tipo, interacao\_canal, interacao\_data\_hora, interacao\_observacao)
  - **Passo a passo (MySQL Workbench):**
    1. **File → Table Data Import Wizard**
    2. Selecione data/leads.csv, mapeie as colunas para a tabela lead e finalize.
    3. Repita para data/interacoes.csv na tabela interacao.
- 

## 13. Links para Scripts SQL

### **create\_objects.sql:**

[https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/create\\_objects.sql](https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/create_objects.sql)

### **insert\_data.sql:**

[https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/insert\\_data.sql](https://raw.githubusercontent.com/matheusmurias/projeto-crm-leads/main/sql/insert_data.sql)