



## **Projeto Lógico de Banco de Dados**

Luis Felipe Ramalho Carvalho

Universidade Federal de Sergipe - Prof. André Britto

# 1. Introdução

Este trabalho tem como objetivo a elaboração completa de um **projeto lógico de banco de dados relacional**, a partir de um **Diagrama Entidade-Relacionamento (DER)** previamente definido no projeto conceitual. O projeto foi desenvolvido conforme as regras e boas práticas discutidas em sala de aula, contemplando aspectos de modelagem conceitual, lógica e implementação física.

O banco de dados modelado tem como finalidade dar suporte a um sistema de gestão de empresas, usuários, pesquisas, créditos e leads, garantindo integridade referencial, consistência dos dados e aderência ao modelo relacional. Ao final do processo, o banco de dados é implementado no **PostgreSQL**, utilizando o serviço **Amazon RDS (Relational Database Service)**.

---

## 2. Projeto Conceitual

O projeto conceitual foi desenvolvido utilizando a notação de DER apresentada em aula. Nesse nível de abstração, foram identificadas as principais entidades do sistema, seus atributos e os relacionamentos existentes entre elas.

### 2.1 Entidades Principais

- **Empresa**: representa as empresas cadastradas no sistema.
- **Usuário**: representa os usuários do sistema.
- **Usuário\_Comum**, **Admin\_da\_Empresa** e **Admin\_do\_Sistema**: especializações da entidade Usuário.
- **Pesquisa**: representa buscas realizadas por usuários comuns.
- **Lead**: representa possíveis clientes retornados pelas pesquisas.
- **Resultado\_de\_pesquisa**: entidade associativa responsável por resolver o relacionamento entre Pesquisa e Lead.
- **Crédito**: representa créditos disponíveis para a realização de pesquisas.
- **Pagamento**: representa pagamentos realizados por empresas.
- **Integração**: representa integrações externas vinculadas às empresas.
- **Log\_de\_acesso**: registra acessos realizados pelos usuários.

### 2.2 Relacionamentos

- Uma **Empresa** possui vários **Usuários** (1:N).
- Um **Usuário** pertence a exatamente uma **Empresa**.
- Um **Usuário\_Comum** realiza várias **Pesquisas** (1:N).
- Uma **Pesquisa** utiliza um **Crédito**.
- O relacionamento entre **Pesquisa** e **Lead** é do tipo **N:N**, resolvido pela entidade associativa **Resultado\_de\_pesquisa**.
- Uma **Empresa** pode possuir várias **Integrações**, **Pagamentos** e **Créditos**.

O DER conceitual foi validado para garantir que todos os relacionamentos estivessem corretamente definidos e que não houvesse ambiguidades estruturais.

---

### 3. Projeto Lógico – Modelo Relacional

A partir do DER conceitual, foi realizado o mapeamento para o modelo relacional, seguindo rigorosamente as regras de transformação vistas em aula.

#### 3.1 Mapeamento das Entidades

Cada entidade forte do DER foi transformada em uma tabela no modelo relacional, com a definição explícita de:

- Chave primária (PRIMARY KEY)
- Tipos de dados adequados
- Restrições de unicidade (UNIQUE)
- Restrições de nulidade (NOT NULL quando aplicável)

#### 3.2 Especialização de Usuário

A especialização da entidade **Usuário** foi implementada utilizando tabelas separadas:

- Usuario\_Comum
- Admin\_da\_Empresa
- Admin\_do\_Sistema

Cada uma dessas tabelas possui como chave primária a mesma chave da tabela Usuario, funcionando também como chave estrangeira, caracterizando uma especialização total e disjunta.

#### 3.3 Relacionamento N:N – Pesquisa e Lead

O relacionamento muitos-para-muitos entre **Pesquisa** e **Lead** foi corretamente resolvido por meio da tabela **Resultado\_de\_pesquisa**, que possui:

- Chave primária composta (idPesquisa, idLead)
- Chaves estrangeiras referenciando Pesquisa e Lead
- Atributos próprios do relacionamento, como ordem, relevância e data\_de\_associacao

Essa abordagem garante aderência ao modelo relacional e preserva a semântica do DER original.

---

## 4. Geração do Script SQL

O script SQL foi inicialmente gerado pelo MySQL Workbench a partir do modelo lógico e posteriormente ajustado para ser compatível com o **PostgreSQL**.

### 4.1 Ajustes Realizados

- Remoção de instruções específicas do MySQL (ENGINE, VISIBLE/INVISIBLE)
- Uso de tipos compatíveis com PostgreSQL (TIMESTAMP, DECIMAL)
- Definição explícita de chaves primárias e estrangeiras
- Inclusão de ações referenciais (ON DELETE / ON UPDATE)
- Garantia de integridade referencial entre todas as tabelas

O script final cria todas as estruturas do banco de dados de forma consistente e pode ser executado diretamente no PostgreSQL.

---

## 5. Implementação no AWS RDS

Após a geração do script SQL, o banco de dados foi implementado utilizando o serviço **Amazon RDS**, com as seguintes configurações:

- Engine: PostgreSQL
- Região: conforme configuração da conta AWS
- Criação de usuário:
  - Usuário: professor
  - Senha: professor

O script SQL foi executado com sucesso no banco criado, resultando na criação de todas as tabelas e restrições previstas no projeto lógico.

O link do RDS com o banco de dados criado é fornecido juntamente com este trabalho, conforme solicitado no enunciado.

---

## 6. Considerações Finais

Este projeto permitiu aplicar de forma prática os conceitos de modelagem de dados abordados na disciplina, passando por todas as etapas fundamentais: projeto conceitual, projeto lógico e implementação física.

A utilização do Amazon RDS proporcionou uma experiência realista de implantação de banco de dados em ambiente de nuvem, reforçando a importância da correta definição de restrições e integridade referencial.

O modelo desenvolvido atende aos requisitos propostos, está devidamente normalizado e pronto para ser utilizado por aplicações que demandem gerenciamento de empresas, usuários, pesquisas e leads de forma consistente e segura.