

# Sistema de Gestão de Campanhas de Vacinação

## Projeto da Disciplina de Fundamentos de Banco de Dados

Maria Eduarda Rodrigues Bastos e Melchisedec Nogueira Martins Neto

Junho de 2025

## 1. Introdução

O presente projeto tem como objetivo o desenvolvimento de um banco de dados para um sistema de gestão de campanhas de vacinação. A aplicação permitirá o controle de vacinas, lotes, locais de vacinação, públicos-alvo e registro de doses aplicadas. Além disso, serão gerados relatórios gerenciais e funcionalidades específicas para acompanhar a cobertura vacinal.

## 2. Requisitos Funcionais

- **REQ 001 - Cadastro de Vacinas e Lotes:** Inserir, editar, remover e listar vacinas e seus lotes. **Dados:** Nome da vacina, fabricante, data de validade do lote, quantidade em estoque. **Restrições:** Nome da vacina e número do lote não podem se repetir.
- **REQ 002 - Gestão de Locais de Vacinação:** Cadastrar, atualizar, excluir e consultar locais de vacinação. **Dados:** Nome, endereço (Rua, Número, Bairro, Cidade), capacidade de atendimento. **Restrições:** Nome do local não pode se repetir.
- **REQ 003 - Registro de Doses Aplicadas:** Registrar aplicação de doses. **Dados:** CPF do vacinado, vacina aplicada, lote, data de aplicação, local de vacinação, profissional aplicador. **Restrições:** CPF do vacinado deve ser único por aplicação.
- **REQ 004 - Relatório de Estoque de Vacinas:** Listar a quantidade de cada vacina em estoque, com data de validade. **Dados:** Nome da vacina, quantidade, validade.
- **REQ 005 - Definição de Público-Alvo:** Cadastrar e gerenciar grupos específicos. **Dados:** Nome do público-alvo, critérios de elegibilidade. **Restrições:** Nome não pode se repetir.
- **REQ 006 - Relatório de Cobertura Vacinal por Público-Alvo:** Gerar um relatório com a quantidade de doses aplicadas para um público-alvo em um período específico. **Dados:** Público-alvo, data inicial, data final, vacina (opcional).

- **REQ 007 - Notificação de Próximas Doses:** Notificar indivíduos sobre a próxima dose. **Dados:** CPF, data da última dose, vacina, intervalo recomendado. **Restrições:** Apenas para vacinas com múltiplas doses.
- **REQ 008 - Consulta de Histórico Vacinal Individual:** Permitir consulta ao histórico de vacinas por CPF. **Dados:** CPF do indivíduo.

### 3. Modelo ER/EER

#### Descrição Textual

O modelo de dados inclui as seguintes entidades:

- **Vacina:** ID\_Vacina (PK), Nome, Fabricante.
- **Lote:** ID\_Lote (PK), Data de validade, Quantidade em estoque, ID\_Vacina (FK).
- **LocalVacinaçao:** ID\_Local (PK), Nome, Endereço (composto: Rua, Número, Bairro, Cidade), Capacidade.
- **Pessoa (Herança):** CPF (PK), Nome, Data de nascimento.
  - **PublicoAlvoEspecifico (Herança):** Nome do público-alvo, Critérios de elegibilidade.
- **DoseAplicada:** ID\_Dose (PK), CPF (FK), ID\_Vacina (FK), ID\_Lote (FK), Data de aplicação, ID\_Local (FK), Profissional aplicador.

#### Relacionamentos:

- Uma Vacina pode ter vários Lotes (1:N).
- Uma Pessoa pode ter várias DosesAplicadas (1:N).
- Uma DoseAplicada ocorre em um único LocalVacinaçao (N:1).
- Herança: Pessoa  $\rightarrow$  PublicoAlvoEspecifico.

### 4. Esquema Relacional

- Vacina (ID\_Vacina PK, Nome, Fabricante)
- Lote (ID\_Lote PK, DataValidade, Quantidade, ID\_Vacina FK)
- LocalVacinaçao (ID\_Local PK, Nome, Rua, Número, Bairro, Cidade, Capacidade)
- Pessoa (CPF PK, Nome, DataNascimento)
- PublicoAlvoEspecifico (NomePublicoAlvo PK, Critérios, CPF FK)
- DoseAplicada (ID\_Dose PK, CPF FK, ID\_Vacina FK, ID\_Lote FK, DataAplicacao, ID\_Local FK, Profissional)

## 5. Divisão de Tarefas

Atividade	Descrição	Responsável
Requisitos Funcionais (REQ 001 a REQ 004)	Definir e documentar os requisitos iniciais	Melchisedec Nogueira
Requisitos Funcionais (REQ 005 a REQ 008)	Definir e documentar os requisitos adicionais	Maria Eduarda Bastos
Modelo ER/EER	Criar o diagrama e a descrição textual	Melchisedec Nogueira
Esquema Relacional	Elaborar o modelo relacional e o diagrama	Melchisedec Nogueira
Documentação Final	Redação do relatório em LaTeX e criação dos slides	Maria Eduarda Bastos
Gravação da Apresentação	Participação na apresentação em vídeo	Ambos

## 6. Tela de Gestão de Vacinas e Fabricantes (Responsável: Melchisedec Nogueira Martins Neto)

Esta tela é dedicada ao gerenciamento das informações sobre vacinas e seus respectivos fabricantes. Ela oferece uma interface para visualizar, adicionar, atualizar e remover dados essenciais para o controle de estoque e tipos de vacinas.

### Objetivo Principal

Controlar o cadastro de fabricantes e vacinas disponíveis no sistema.

### Funcionalidades Detalhadas

- **Consulta e Filtragem de Fabricantes:**

- Exibe uma tabela com todos os fabricantes cadastrados (ID e Nome).
- Permite filtrar a lista de fabricantes por nome, atualizando a tabela dinamicamente.

- **Inclusão de Fabricante:**

- Campo para inserir o nome de um novo fabricante.
- Botão “Adicionar Fabricante” que registra o novo fabricante no banco de dados.

- **Remoção de Fabricante:**

- Campo para informar o ID do fabricante a ser removido.
- Botão “Remover Fabricante” que exclui o registro do fabricante do banco. O sistema valida se há vacinas associadas, prevenindo a remoção se existirem dependências.

- **Consulta e Filtragem de Vacinas:**

- Exibe uma tabela com todas as vacinas cadastradas (ID, Nome, Tipo e Fabricante).
- Permite filtrar a lista de vacinas por nome, tipo e nome do fabricante, atualizando a tabela em tempo real.
- **Inclusão de Vacina:**
  - Campos para inserir o nome e o tipo da nova vacina.
  - Um seletor (dropdown) para escolher o fabricante associado à vacina, populado dinamicamente com os fabricantes cadastrados.
  - Botão “Adicionar Vacina” que registra a nova vacina no banco de dados.
- **Edição de Vacina:**
  - Campo para informar o ID da vacina a ser editada.
  - Campos opcionais para o novo nome, novo tipo e um seletor para o novo fabricante.
  - Botão “Atualizar Vacina” que modifica os dados da vacina no banco.
- **Remoção de Vacina:**
  - Campo para informar o ID da vacina a ser removida.
  - Botão “Remover Vacina” que exclui o registro da vacina do banco. O sistema valida se há aplicações de vacina associadas, prevenindo a remoção se existirem dependências.

## 7. Tela de Gestão Técnica e Integração de Dados (Responsável: Maria Eduarda Rodrigues Bastos)

Esta parte do projeto foi responsável por implementar a estrutura técnica do sistema, incluindo o servidor backend, a integração com o banco de dados e o desenvolvimento de telas que interagem diretamente com as rotas e o armazenamento de informações.

### Objetivo Principal

Garantir a comunicação eficiente entre o banco de dados e as interfaces de usuário, implementando regras de negócio, estruturação das rotas e persistência de dados com segurança e integridade.

### Funcionalidades Desenvolvidas

- **Backend com FastAPI:**
  - Estruturação de um servidor FastAPI que expõe rotas RESTful para operações de CRUD nas entidades principais (*Vacina*, *Fabricante*, *Paciente*, *Aplicação*).
  - Utilização de SQLAlchemy como ORM para integração com o banco de dados relacional (SQLite).

- Implementação de rotas como:
  - \* `GET /usuarios/` – Retorna lista de pacientes;
  - \* `GET /vacinas/` – Retorna lista de vacinas e fabricantes;
  - \* `POST /registro/` – Registra uma nova aplicação de vacina.
- **Banco de Dados:**
  - Desenvolvimento do script `vacinas.db` responsável por:
    - \* Criar as tabelas baseadas no modelo relacional do projeto;
    - \* Inserir dados iniciais de teste (pacientes, vacinas, fabricantes);
    - \* Gerar o arquivo `vacinas.db` para ser consumido pelo backend.
- **Telas e Consumo de API:**
  - Criação de interfaces simples em `notebook` (`front_vacinas`) que simulam a interação do usuário com o sistema.
  - Consumo das rotas via biblioteca `requests`, permitindo ações como registro de aplicação ou consulta de pacientes, de forma integrada ao banco de dados.

## Fluxo de Execução do Sistema

1. **Inicialização do Banco:** executar o notebook `vacinas.db.ipynb` para criar e popular o banco de dados.
2. **Execução do Servidor:** iniciar o backend com o arquivo `back_vacinas.py` ou notebook equivalente.
3. **Consumo das Rotas:** utilizar o frontend no notebook `front_vacinas.ipynb` para simular a interação com o sistema.

## 8. Conclusão

O desenvolvimento deste projeto permitiu aplicar, de forma integrada, os principais conceitos de banco de dados relacional e sua utilização em um sistema completo. A partir da modelagem conceitual e relacional, implementamos um banco funcional com múltiplas entidades inter-relacionadas e com regras de integridade.

Além da modelagem, foi possível evoluir para a construção de um ambiente funcional que simula o uso real de um sistema de vacinação, com backend implementado utilizando a biblioteca FastAPI, banco de dados persistente com SQLAlchemy e SQLite, além de uma interface de interação via Jupyter Notebook com consumo de rotas REST.

A divisão de tarefas permitiu uma atuação complementar entre os membros do grupo: enquanto um focou no levantamento e detalhamento das funcionalidades visuais e de cadastro (como vacinas e fabricantes), o outro foi responsável pela estruturação da base técnica que viabiliza o funcionamento do sistema, incluindo banco de dados, servidor e integração com as interfaces.

O projeto também proporcionou experiência com ferramentas modernas de desenvolvimento, organização em módulos e execução prática de um ciclo completo de criação de banco de dados e aplicação web.