

Proposta de Estrutura de Banco de Dados para o Food Control App

Autor: Manus AI Data: 21 de Dezembro de 2025

Este documento apresenta a proposta de modelagem de dados para o aplicativo Food Control App, com foco na implementação no Supabase (PostgreSQL). A estrutura foi concebida a partir da análise detalhada das páginas de entrada de dados ("Cadastro Geral", "Insumos" e "Fichas Técnicas"), garantindo a integridade referencial e a escalabilidade do sistema.

A arquitetura de dados é dividida em três categorias principais: **Tabelas de Dados Auxiliares** (para padronização de cadastros), **Tabelas de Dados Principais** (para o core do negócio) e **Tabelas de Relacionamento** (para modelar as composições).

1. Tabelas de Dados Auxiliares (Lookup Tables)

As tabelas auxiliares são essenciais para manter a consistência dos dados e facilitar a manutenção, armazenando as listas de opções utilizadas nos campos de seleção do aplicativo. O uso de `UUID` como chave primária (`PK`) é recomendado para garantir a unicidade global dos registros.

Tabela	Coluna	Tipo de Dado (PostgreSQL)	Descrição
categorias_insumos	id	uuid (PK)	Identificador único da categoria de insumo.
	nome	varchar(100)	Nome da categoria (e.g., Bebidas, Confeitaria).
categorias_sinteticas	id	uuid (PK)	Identificador único da categoria sintética.
	nome	varchar(100)	Nome da categoria sintética (e.g., Insumo de produção pronto).
setores_responsaveis	id	uuid (PK)	Identificador único do setor responsável.

	nome	varchar(100)	Nome do setor (e.g., Brigadeiro, Chocolate).
especialidades	id	uuid (PK)	Identificador único da especialidade.
	nome	varchar(100)	Nome da especialidade (e.g., Bem-Casado, Bolos).
unidades_medida	id	uuid (PK)	Identificador único da unidade de medida.
	sigla	varchar(10)	Sigla da unidade (e.g., g, ml, kg, L, un).
	descricao	varchar(50)	Descrição completa da unidade.
dificuldades	id	uuid (PK)	Identificador único do nível de dificuldade.
	nome	varchar(50)	Nível de dificuldade (e.g., Fácil, Médio, Difícil).

2. Tabelas de Dados Principais

2.1. Tabela insumos (Ingredientes)

Esta tabela centraliza o cadastro de todos os insumos, sejam eles matérias-primas ou produtos base (Fichas Técnicas que se tornam insumos). O custo real por unidade de medida (g/ml/un) será um campo calculado, derivado dos dados de compra e do fator de correção.

Coluna	Tipo de Dado (PostgreSQL)	Chave/Relacionamento	Descrição
id	uuid (PK)		Identificador único do insumo.

nome_padronizado	varchar(255)		Nome padrão do insumo (usado na FT).
descricao_produto	varchar(255)		Descrição detalhada do produto (marca, embalagem).
categoria_id	uuid (FK)	categorias_insumos.id	Categoria do insumo.
categoria_sintetica_id	uuid (FK)	categorias_sinteticas.id	Categoria sintética do insumo.
custo_compra	numeric		Custo total da compra (R\$).
quantidade_compra	numeric		Quantidade comprada.
unidade_compra_id	uuid (FK)	unidades_medida.id	Unidade de medida da compra (e.g., kg, L, un).
peso_unidade	numeric		Peso/Volume da unidade comprada.
unidade_peso_id	uuid (FK)	unidades_medida.id	Unidade de medida do peso/volume (e.g., g, ml).
fator_correcao	numeric		Fator de correção (Peso Bruto / Peso Líquido).
fornecedor	varchar(255)		Nome do fornecedor.
data_cadastro	timestamp with time zone		Data de criação do registro.

2.2. Tabela `fichas_tecnicas` (Receitas)

Armazena as informações de identificação e metadados de cada Ficha Técnica de Produção.

Coluna	Tipo de Dado (PostgreSQL)	Chave/Relacionamento	Descrição
--------	---------------------------	----------------------	-----------

<code>id</code>	<code>uuid</code> (PK)		Identificador único da ficha técnica.
<code>nome_receita</code>	<code>varchar(255)</code>		Nome da receita.
<code>tipo_produto</code>	<code>varchar(50)</code>	<code>ENUM ('Final', 'Base')</code>	Tipo de produto (Final ou Base).
<code>setor_responsavel_id</code>	<code>uuid</code> (FK)	<code>setores_responsaveis.id</code>	Setor responsável pela receita.
<code>especialidade_id</code>	<code>uuid</code> (FK)	<code>especialidades.id</code>	Especialidade da receita.
<code>codigo_id</code>	<code>varchar(50)</code>	(Unique)	Código/ID da receita (e.g., BRI-0001).
<code>data_criacao</code>	<code>date</code>		Data de criação da ficha.
<code>dificuldade_id</code>	<code>uuid</code> (FK)	<code>dificuldades.id</code>	Nível de dificuldade.
<code>tempo_preparo</code>	<code>varchar(50)</code>		Tempo de preparo (e.g., 45 min).
<code>e_insumo</code>	<code>boolean</code>		Indica se a receita também é um insumo.
<code>rendimento_kg</code>	<code>numeric</code>		Rendimento final da receita em kg.
<code>observacoes</code>	<code>text</code>		Campo para observações ou dicas de uso da receita.

3. Tabela de Relacionamento (M:N)

3.1. Tabela `ft_ingredientes` (Composição da Ficha Técnica)

Esta tabela resolve o relacionamento de muitos-para-muitos entre Fichas Técnicas e Insumos, permitindo que uma receita seja composta por múltiplos insumos e que um insumo seja utilizado em múltiplas receitas.

Coluna	Tipo de Dado (PostgreSQL)	Chave/Relacionamento	Descrição
ft_id	uuid (PK, FK)	fichas_tecnicas.id	ID da Ficha Técnica.
insumo_id	uuid (PK, FK)	insumos.id	ID do Insumo (ingrediente).
quantidade_utilizada	numeric		Quantidade do insumo usada na receita.
unidade_utilizada_id	uuid (FK)	unidades_medida.id	Unidade de medida da quantidade utilizada.
dica_uso	varchar(255)		Dica de uso opcional para o ingrediente.

4. Estratégias de Implementação no Supabase

Para otimizar a performance e a lógica de negócio no Supabase, sugere-se a aplicação das seguintes estratégias:

- Cálculos de Custo:** Utilizar **Funções de Banco de Dados (Stored Procedures)** ou **Views Materializadas** para calcular o custo real dos insumos e o custo total das receitas. Isso desonera o aplicativo e garante que a lógica de cálculo seja centralizada e eficiente.
- Integração FT-Insumo:** A lógica de que "uma FT pode virar um insumo" pode ser implementada através de uma **View** que combine os registros da tabela `insumos` com as Fichas Técnicas onde o campo `e_insumo` é `TRUE`. Esta View pode ser consultada pelo aplicativo ao listar os insumos disponíveis para composição de outras Fichas Técnicas.
- Segurança (RLS):** Implementar o **Row Level Security (RLS)** do Supabase em todas as tabelas para garantir que apenas usuários autenticados e autorizados possam acessar e modificar os dados.
- Validação de Dados:** Utilizar as restrições de banco de dados (e.g., `NOT NULL`, `CHECK` constraints) para garantir a integridade dos dados, como a obrigatoriedade de campos essenciais e a validação de valores numéricos.

Esta estrutura relacional atende integralmente aos requisitos de cadastro e relacionamento do Food Control App, fornecendo uma base sólida para o desenvolvimento da aplicação.

com o Supabase.