



FACULDADE DONADUZZI  
2º PERÍODO CIÊNCIA DE DADOS E INTELIGÊNCIA ARTIFICIAL

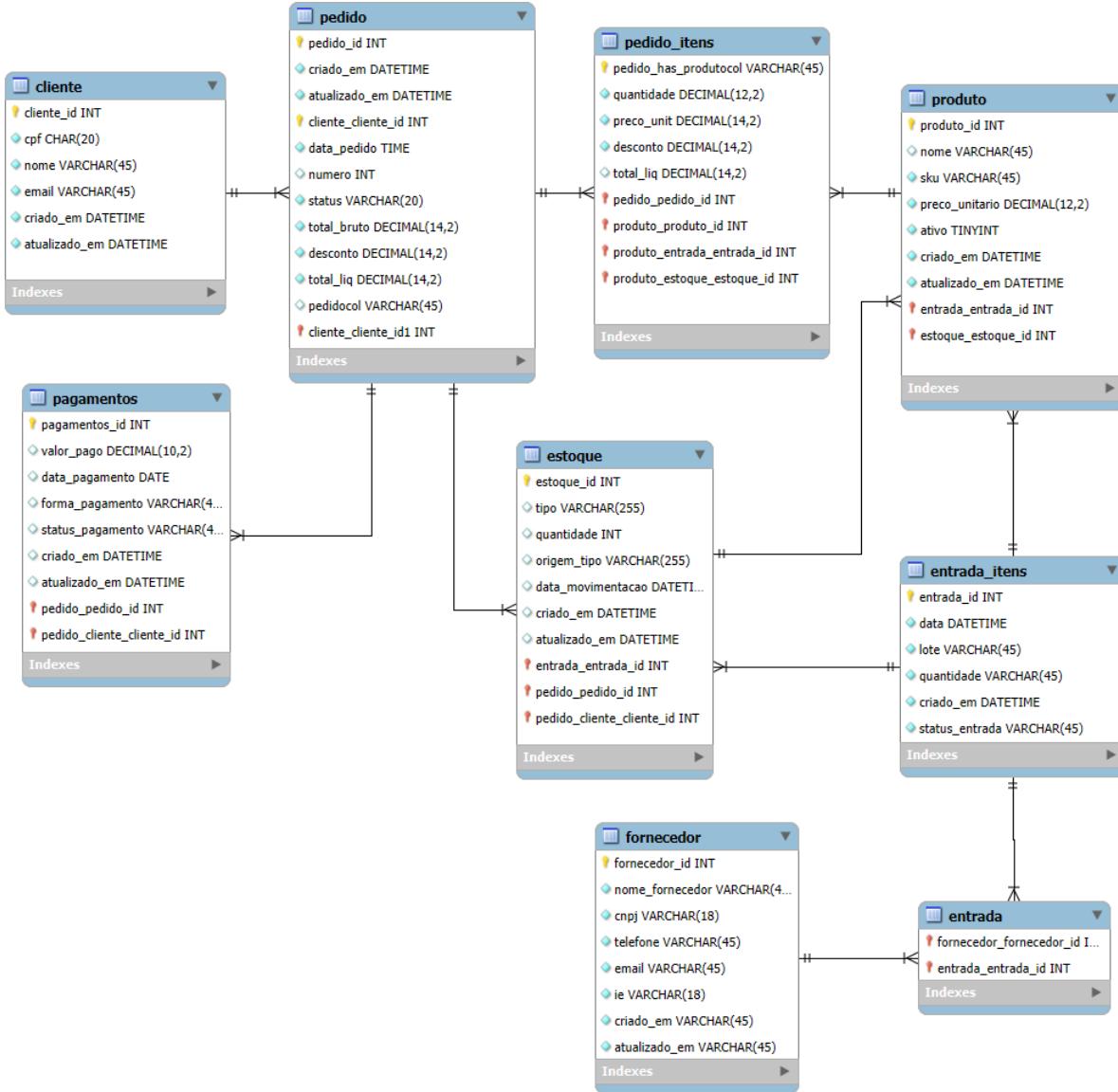
Miguel Mantoan Castellani  
Kaique Geska  
Vitor Sauer

**Armazenamento e Otimização de  
Banco de Dados**

Grupo 06  
Toledo, 2025

# Relatório de Justificativa do Modelo de Banco de Dados

## Modelo EER



## 1. Justificativa das Chaves (PKs e FKs)

As chaves primárias (PKs) e chaves estrangeiras (FKs) são a espinha dorsal do modelo relacional, garantindo a unicidade e a integridade dos dados.

### Chaves Primárias (PKs)

Todas as tabelas foram definidas com uma chave primária numérica, seguindo a boa prática de usar chaves substitutas (surrogate keys). O tipo de dados escolhido foi BIGSERIAL (que cria uma sequência e usa BIGINT). Este tipo garante que o sistema possa escalar para um grande volume de registros sem esgotar o limite do tipo INTEGER. O padrão de nomenclatura adotado foi o descriptivo, como cliente\_id e pedidos\_id, em vez de apenas id.

Esta escolha melhora a clareza e a legibilidade em consultas com múltiplas junções, evitando ambiguidades.

#### *Chaves Estrangeiras (FKs)*

As FKs foram implementadas para garantir a Integridade Referencial, assegurando que os dados dependentes sempre apontem para registros válidos e existentes.

- Pedidos: A coluna cliente\_id referencia a tabela clientes, garantindo que todo pedido seja feito por um cliente cadastrado.
- Itens do Pedido: As colunas pedido\_id e produto\_id referenciam, respectivamente, as tabelas pedidos e produtos, garantindo que o item pertença a um pedido existente e se refira a um produto válido.
- Estoque: A coluna produto\_id referencia a tabela produtos, garantindo que a movimentação de estoque seja para um produto existente.
- Pagamentos: A coluna pedido\_id referencia a tabela pedidos, garantindo que todo pagamento esteja vinculado a um pedido existente.

### **2. Justificativa do Comportamento ON DELETE**

O comportamento ON DELETE nas FKs foi definido de forma estratégica para proteger o histórico de dados e, ao mesmo tempo, garantir a limpeza de registros dependentes.

#### *ON DELETE CASCADE*

O CASCADE foi utilizado nos relacionamentos de forte dependência, onde a exclusão da linha principal implica que a linha dependente não tem mais utilidade.

- Pedidos para Clientes: Se um cliente for excluído (por exemplo, por solicitação de privacidade), todos os seus pedidos, e consequentemente os itens, pagamentos e movimentações de estoque associadas, devem ser excluídos automaticamente.
- Itens e Pagamentos para Pedidos: Um item de pedido ou um registro de pagamento não faz sentido sem seu pedido pai. A exclusão do pedido leva à exclusão automática de seus itens e pagamentos.

#### *ON DELETE RESTRICT*

O RESTRICT foi utilizado no relacionamento de Itens do Pedido para Produtos.

- Justificativa Crucial para o Histórico: Esta regra impede que um produto seja excluído da tabela produtos se houver qualquer item de pedido (histórico de vendas) que o refcrcie. Isso protege a integridade do histórico de vendas. Para descontinuar um produto, a coluna ativo deve ser usada, mantendo o registro do produto para fins históricos.

### **3. Justificativa das CHECK Constraints**

As CHECK constraints foram utilizadas para impor as Regras de Negócio (RNs) e garantir a validade de domínio dos dados, prevenindo a inserção de valores logicamente incorretos.

- *Valores Não Negativos*: Constraints foram aplicadas em produtos.preco\_unitario, pedido\_itens.quantidade, e pagamentos.valor\_pago para garantir que os valores sejam

sempre maiores ou iguais a zero (ou estritamente maiores que zero, no caso da quantidade).

- *Domínio de Status:* A constraint na coluna pedidos.status limita os valores aceitos para 'ABERTO', 'PAGO', ou 'CANCELADO', garantindo a conformidade com a Regra de Negócio.

#### **4. Justificativa dos Índices**

Os índices foram criados para otimizar o desempenho das consultas e para impor a unicidade, conforme os requisitos.

- *Unicidade:* Índices UNIQUE foram criados implicitamente nas colunas clientes.email e produtos.sku para garantir que cada registro seja único, conforme a Regra de Negócio.
- *Otimização de Consultas:* Índices explícitos foram criados nas Chaves Estrangeiras mais consultadas (idx\_pedidos\_cliente, idx\_itens\_produto, idx\_estoque\_produto, idx\_pagamentos\_pedido). Estes índices aceleraram a busca por dados relacionados, como encontrar todos os pedidos de um cliente ou todas as movimentações de estoque de um produto.

#### **5. Normalização (3FN)**

O modelo atende à Terceira Forma Normal (3FN), o que é o requisito mínimo para um esquema consistente

- *Primeira Forma Normal (1FN):* É atendida, pois não há grupos repetitivos ou atributos multivalorados; cada coluna contém valores atômicos.
- *Segunda Forma Normal (2FN):* É atendida, pois todas as colunas não-chave dependem da chave primária inteira.
- *Terceira Forma Normal (3FN):* É atendida, pois não há dependências transitivas. Por exemplo, o preço unitário do produto no momento da venda é armazenado na tabela pedido\_itens (preco\_unit) e não na tabela pedidos. Isso garante que o preço registrado no momento da venda seja preservado, e que as colunas não-chave não dependam de outras colunas não-chave, minimizando a redundância e protegendo a integridade histórica dos dados.