



ESCOLA SUPERIOR DE ADMINISTRAÇÃO, MARKETING E
COMUNICAÇÃO

IGOR GABRIEL LOPES DE LIMA

DOCUMENTAÇÃO TÉCNICA

Banco de dados: Listas de campos e componentes do banco de dados

UBERLANDIA, MG

2025

	BANCO DE DADOS I		Código: 01-2025	
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas 09	Revisão: 01

Sumário

INTRODUÇÃO	3
ESTRUTURA DO BANCO DE DADOS.....	3
NORMAS E REGRAS	3
MER	4
Entidades.....	4
Descrição de Entidades	4
Relacionamentos.....	4
DER – DIAGRAMA DE ENTIDADE - RELACIONAMENTOS	5
DLD – DIAGRAMA LOGICO DE DADOS	5
DICIONARIO DE DADOS	5
Entidades.....	6
DOCUMENTAÇÃO BANCO DE DADOS	6
DDL	7
DML	7
VIEWS	7
TRIGGERS.....	8
INDICES.....	8
OUTRAS CONSULTAS SQL.....	8
CONCLUSAO	9

	BANCO DE DADOS I		Código: 01-2025
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas: 09
			Revisão: 01

Título: Listas de campos e componentes do banco de dados.		
Sistema: PostgreSQL	Relator: Igor Gabriel Lopes de Lima	Data: 22/05/2025
Objetivos: Descrever de forma clara e organizada a estrutura, os relacionamentos e os principais componentes do banco de dados desenvolvido.		

INTRODUÇÃO

Este documento tem como objetivo registrar o processo de criação de um banco de dados desenvolvido durante o período de estudo, com foco no aprendizado e aplicação de conceitos fundamentais de modelagem e gerenciamento de dados. Através deste projeto, buscou-se compreender etapas importantes como a definição de requisitos, elaboração do modelo conceitual e lógico, normalização de tabelas e implementação do banco em um sistema gerenciador de banco de dados (SGBD).

A prática da construção de um banco de dados permite consolidar conhecimentos teóricos adquiridos em sala de aula, além de desenvolver habilidades técnicas essenciais para a atuação profissional na área de Sistemas de Informação. Ao longo deste relatório, serão apresentados os objetivos do projeto, as decisões de modelagem adotadas, a estrutura do banco criado e as principais funcionalidades implementadas.

ESTRUTURA DO BANCO DE DADOS


Nesta seção, são apresentados os principais componentes envolvidos na modelagem e estruturação do banco de dados desenvolvido ao longo do estudo. Inicialmente, são descritas as normas e regras adotadas para garantir padronização e integridade nas definições. Em seguida, é apresentado o Modelo Entidade-Relacionamento (MER), que compreende a identificação das entidades, seus atributos e os relacionamentos entre elas, acompanhado da descrição detalhada de cada elemento.

O Diagrama Entidade-Relacionamento (DER) fornece uma representação visual dessa estrutura conceitual, facilitando a compreensão do funcionamento do sistema. Na sequência, o Diagrama Lógico de Dados (DLD) traduz essa modelagem para um formato mais próximo do banco de dados real, já considerando tipos de dados e chaves.

Por fim, o Dicionário de Dados documenta detalhadamente todas as entidades, atributos e relacionamentos definidos, servindo como guia técnico para a implementação e manutenção do banco de dados. Esta estrutura proporciona uma base sólida para o desenvolvimento do sistema e assegura a organização e a consistência das informações.

NORMAS E REGRAS

Diretrizes aplicadas durante todas as etapas do projeto para garantir organização, consistência e integridade das informações. Com base nas três primeiras formas normais (1FN,

	BANCO DE DADOS I DOCUMENTO DE ESPECIFICAÇÃO DE DADOS	Código: 01-2025	
		Folhas: 09	Revisão: 01

2FN e 3FN), essas normas visam eliminar redundâncias, evitar dependências inadequadas entre dados e facilitar a manutenção do sistema. Além disso, incluem padrões de nomenclatura, definição de chaves primárias e estrangeiras, e aplicação de regras de integridade referencial.

MER

O Modelo Entidade-Relacionamento (MER) é uma técnica conceitual utilizada no desenvolvimento de bancos de dados para representar entidades, seus atributos e os relacionamentos entre elas. Ele permite compreender de forma visual a estrutura e a organização dos dados, além de como as entidades interagem dentro de um sistema de informação. A partir do MER, são elaborados os Diagramas Entidade-Relacionamento (DER), que representam graficamente essa estrutura conceitual.

Entidades

- USUARIO
- CONTA
- CATEGORIA
- TRANSAÇÃO
- ORÇAMENTO
- META_FINANCEIRA
- PARCELA
- LEMBRETE_FINANCEIRO

Descrição de Entidades


- USUÁRIO (id_usuario, nome, email, senha);
- CONTA (id_conta, id_usuario, nome_banco, tipo_conta, saldo_inicial);
- CATEGORIA (id_categoria, nome, tipo);
- TRANSAÇÃO (id_transacao, id_conta, id_categoria, data, valor, descricao, tipo);
- ORÇAMENTO (id_orcamento, id_usuario, id_categoria, mes, ano, valor_limite);
- META_FINANCEIRA (id_meta, id_usuario, nome, valor_objetivo, data_limite);
- PARCELA (id_transacao, n_parcela, valor_parcela, data_vencimento);
- LEMBRETE_FINANCEIRO (id_lembrete PK, id_usuario FK, titulo, descricao, data_lembrete, recorrente, frequencia);

Relacionamentos

USUARIO – CONTA: Um usuário pode ter várias contas. Cada conta pertence a um único usuário. Cardinalidade: (1: n)

USUARIO – ORÇAMENTO: Um usuário pode ter vários orçamentos. Cada orçamento pertence a um único usuário. Cardinalidade: (1: n)

USUARIO – META_FINANCEIRA: Um usuário pode definir várias metas financeiras. Cada meta pertence a um único usuário. Cardinalidade: (1: n)

	BANCO DE DADOS I		Código: 01-2025	
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas 09	Revisão: 01

USUARIO – LEMBRETE_FINANCEIRO: Um usuário cria vários lembretes; cada lembrete pertence a um único usuário. Cardinalidade: (1: n)

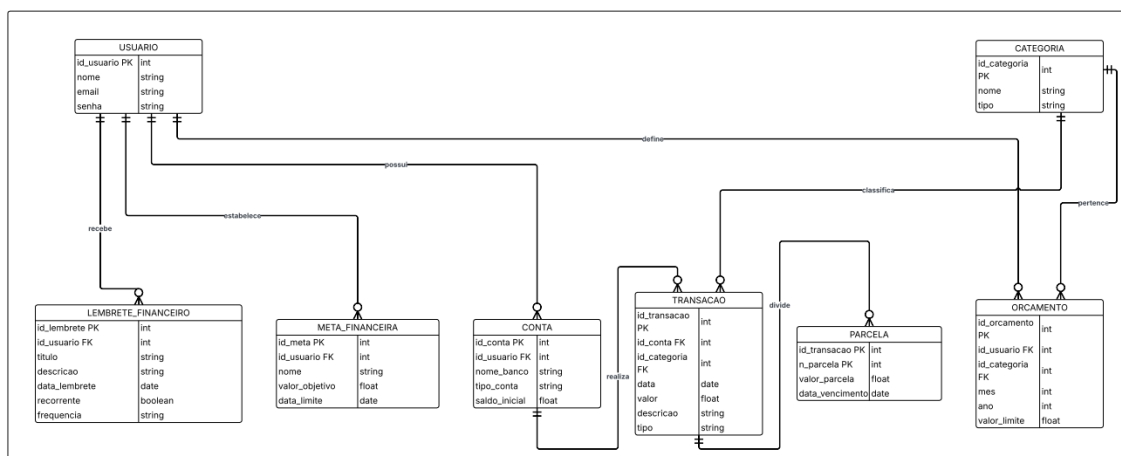
CONTA – TRANSAÇÃO: Uma conta pode ter várias transações. Cada transação está vinculada a uma única conta. Cardinalidade: (1: n)

CATEGORIA – TRANSAÇÃO: Uma categoria pode ser associada a várias transações. Cada transação pertence a uma única categoria. Cardinalidade: (1: n)

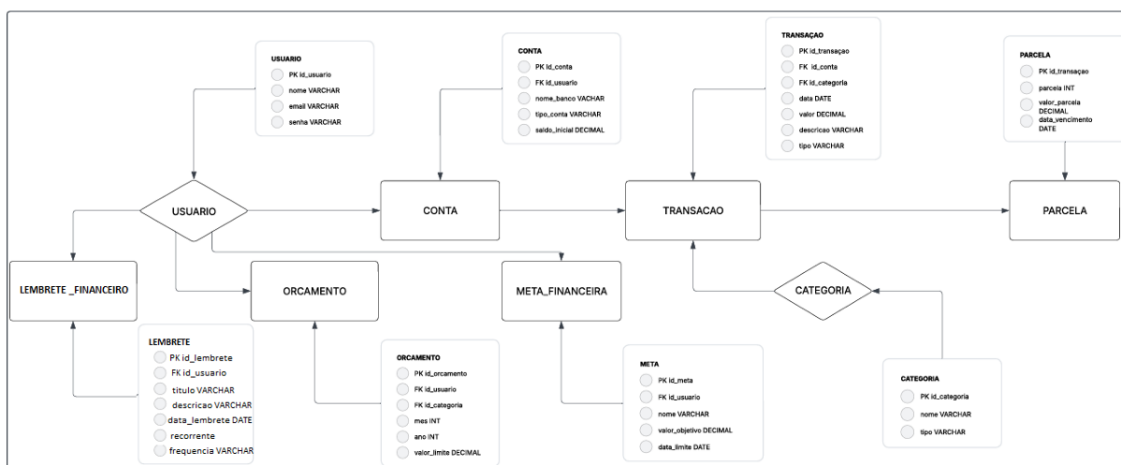
CATEGORIA – ORÇAMENTO: Uma categoria pode ser usada em vários orçamentos. Cada orçamento se refere a uma única categoria. Cardinalidade: (1: n)

TRANSAÇÃO – PARCELA: Uma transação pode ter várias parcelas. Cada parcela pertence a uma única transação. Cardinalidade: (1: n) | (Chave composta: id_transacao + n_parcela)

DER – DIAGRAMA DE ENTIDADE - RELACIONAMENTOS



DLD – DIAGRAMA LOGICO DE DADOS



DICIONARIO DE DADOS

Um recurso fundamental no gerenciamento de bancos de dados. Trata-se de um documento ou repositório estruturado que descreve de forma detalhada os elementos do banco, como tabelas, campos, tipos de dados, relacionamentos e regras de negócio associadas. Ele funciona

ESAMC	BANCO DE DADOS I	Código: 01-2025	
		Folhas 09	Revisão: 01

DOCUMENTO DE ESPECIFICAÇÃO DE DADOS

como uma fonte oficial de referência para desenvolvedores, analistas e demais stakeholders, promovendo a compreensão, padronização e consistência das informações em todo o sistema. Ao fornecer uma visão clara da estrutura e do significado dos dados, o Dicionário de Dados facilita a manutenção, integração e o uso eficiente das informações no ambiente organizacional.

Entidades

Tabelas das propriedades de entidades, atributos, dados e descrições de funcionalidades.

Entidade	Atributo	Propriedades do atributo	Tipo de dado	Tamanho	Descrição
USUARIO	id_usuario	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único do usuário.
USUARIO	nome	NOT NULL	CHARACTER	100	Nome completo do usuário.
USUARIO	email	UNIQUE NOT NULL	CHARACTER	100	Endereço de e-mail único do usuário.
USUARIO	senha	NOT NULL	CHARACTER	100	Senha utilizada para login no sistema.
CONTA	id_conta	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único da conta.
CONTA	id_usuario	NOT NULL	INTEGER	-	Identificador do usuário proprietário da conta.
CONTA	nome_banco		CHARACTER	100	Nome da instituição bancária.
CONTA	tipo_conta		CHARACTER	50	Tipo da conta (ex: corrente, poupança).
CONTA	saldo_inicial		NUMERIC	12,2	Saldo inicial da conta ao ser cadastrada.
CATEGORIA	id_categoria	PRIMARY KEY NOT NULL	INT	-	Identificador único da categoria.
CATEGORIA	nome	NOT NULL	CHARACTER	100	Nome da categoria.
CATEGORIA	tipo	NOT NULL	CHARACTER	10	Tipo da categoria: 'entrada' ou 'saida'.
TRANSAÇÃO	id_transacao	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único da transação.
TRANSAÇÃO	id_conta	NOT NULL	INTEGER	-	Conta relacionada à transação.
TRANSAÇÃO	id_categoria	NOT NULL	INTEGER	-	Categoria associada à transação.
TRANSAÇÃO	data	NOT NULL	DATE	-	Data em que a transação ocorreu.
TRANSAÇÃO	valor	NOT NULL	NUMERIC	12,2	Valor monetário da transação.
TRANSAÇÃO	descricao		TEXT	-	Descrição detalhada da transação.
TRANSAÇÃO	tipo	NOT NULL	CHARACTER	10	Tipo da transação: 'entrada' ou 'saida'.
ORÇAMENTO	id_orcamento	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único do orçamento.
ORÇAMENTO	id_usuario	NOT NULL	INTEGER	-	Usuário ao qual o orçamento pertence.
ORÇAMENTO	id_categoria	NOT NULL	INTEGER	-	Categoria de despesa ou receita relacionada.
ORÇAMENTO	mes		INTEGER	-	Mês de referência do orçamento.
ORÇAMENTO	ano		INTEGER	-	Ano de referência do orçamento.
ORÇAMENTO	valor		INTEGER	-	Valor planejado para a categoria (pode ser opcional).
ORÇAMENTO	limite		NUMERIC	12,2	Limite financeiro definido para a categoria.
META_FINANCEIRA	id_meta	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único da meta financeira.
META_FINANCEIRA	id_usuario	NOT NULL	INTEGER	-	Usuário associado à meta.
META_FINANCEIRA	nome	NOT NULL	CHARACTER	100	Nome ou título da meta.
META_FINANCEIRA	valor_objetivo		NUMERIC	12,2	Valor monetário que se deseja atingir.
META_FINANCEIRA	data_limite		DATE	-	Data limite para alcançar o objetivo.
PARCELA	id_transacao	PRIMARY KEY NOT NULL	INTEGER	-	Identificador da transação associada.
PARCELA	n_parcela	PRIMARY KEY NOT NULL	INTEGER	-	Número da parcela.
PARCELA	valor_parcela		NUMERIC	12,2	Valor da parcela individual.
PARCELA	data_vencimento		DATE	-	Data de vencimento da parcela.
LEMBRETE_FINANCEIRO	id_lembrete	PRIMARY KEY NOT NULL	INTEGER	-	Identificador único do lembrete.
LEMBRETE_FINANCEIRO	id_usuario	NOT NULL	INTEGER	-	Usuário responsável pelo lembrete.
LEMBRETE_FINANCEIRO	titulo	NOT NULL	CHARACTER	100	Título do lembrete.
LEMBRETE_FINANCEIRO	descricao		TEXT	-	Descrição detalhada do lembrete.
LEMBRETE_FINANCEIRO	data_lembrete	NOT NULL	DATE	-	Data em que o lembrete deve ser exibido.
LEMBRETE_FINANCEIRO	recorrente		BOOLEAN	-	Indica se o lembrete é recorrente.
LEMBRETE_FINANCEIRO	frequencia		CHARACTER	20	Frequência da repetição (ex: diária, semanal, mensal).

DOCUMENTAÇÃO BANCO DE DADOS

A criação do banco de dados realizada por meio da linguagem **SQL (Structured Query Language)**, padrão para definição, manipulação e controle de dados em sistemas relacionais. O uso do SQL permite a criação precisa de tabelas, definição de tipos de dados, aplicação de restrições, definição de chaves primárias e estrangeiras, além da implementação de regras de integridade e relacionamentos entre entidades.

	BANCO DE DADOS I		Código: 01-2025
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas: 09

DDL

Com base no arquivo **Script DDL.sql**, foi estruturado um banco de dados relacional voltado ao controle financeiro pessoal, utilizando comandos DDL (Data Definition Language). O script define tabelas como Usuario, Conta, Categoria, Transacao, Orcamento, MetaFinanceira, Parcela e Lembrete_Financeiro, cada uma com suas respectivas chaves primárias e estrangeiras. Foram aplicadas restrições como NOT NULL, UNIQUE, CHECK e ON DELETE CASCADE, garantindo a integridade dos dados e a consistência dos relacionamentos entre entidades. Destaca-se o uso do tipo SERIAL para auto incremento de IDs e NUMERIC(12,2) para valores monetários com precisão. O modelo também contempla aspectos como recorrência de lembretes, metas com prazos e controle de parcelas de transações.

DML


Com base no arquivo **Script DML.sql**, foram utilizados comandos da linguagem DML (Data Manipulation Language), especialmente o INSERT, para povoar as tabelas do banco de dados com dados iniciais. Esses comandos inserem registros em tabelas como Usuario, Conta, Categoria, Transacao, Orcamento, MetaFinanceira, Parcela e Lembrete_Financeiro, fornecendo exemplos práticos de uso e permitindo testes imediatos no sistema. Os dados inseridos incluem usuários fictícios com suas contas bancárias, categorias de receitas e despesas, lançamentos financeiros com diferentes datas e valores, orçamentos mensais, metas com prazos definidos, além de lembretes financeiros com e sem recorrência. As instruções respeitam as restrições definidas no modelo, como integridade referencial e valores válidos em campos controlados por CHECK. Esse script complementa a estrutura criada anteriormente no arquivo DDL, simulando um ambiente realista de uso do sistema e demonstrando como os dados se relacionam de forma integrada.

VIEWS

Com base no arquivo **views.sql**, foram criadas três views que facilitam a análise de dados financeiros no banco de dados. Essas visualizações agregam informações relevantes a partir das transações registradas, permitindo consultas mais simples e eficientes.

A view total_receitas_despesas_por_categoria apresenta o total de valores agrupados por categoria e tipo (receita ou despesa), organizando os resultados de forma decrescente por valor dentro de cada tipo. Já a view saldo_mensal_por_conta resume, para cada conta e mês, o total de receitas, despesas e o saldo mensal resultante, com base no tipo de cada categoria associada às transações. Essa visualização é útil para acompanhar a evolução financeira de cada conta ao longo do tempo. Por fim, a view despesas_por_mes_e_categoria mostra o total de despesas mensais por categoria, ordenando os resultados do mês mais recente para o mais antigo, priorizando as categorias com maiores gastos.

Essas views otimizam o acesso a informações consolidadas e são ferramentas valiosas para relatórios, painéis e decisões financeiras no sistema, evitando a repetição de consultas complexas diretamente sobre as tabelas base.

	BANCO DE DADOS I		Código: 01-2025
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas 09 Revisão: 01

TRIGGERS

Com base no arquivo **triggers.sql**, foram criados gatilhos (triggers) e funções em PL/pgSQL que automatizam regras de negócio e asseguram a integridade dos dados no banco. A primeira função, `fn_valor_positivo`, garante que o valor de uma transação sempre seja positivo, convertendo automaticamente valores negativos no momento da inserção ou atualização — essa lógica é aplicada pela trigger `trg_valor_positivo` na tabela `Transacao`.

Outra funcionalidade implementada é a atualização automática do campo `ultima_atualizacao` na tabela `Conta`, sempre que houver uma modificação nos dados dessa tabela. Isso é feito pela função `fn_update_ultima_atualizacao`, associada à trigger `trg_update_ultima_atualizacao`, que registra a data e hora da última alteração.

Por fim, a função `fn_prevent_categoria_delete`, ligada à trigger `trg_prevent_categoria_delete`, impede que uma categoria seja excluída caso haja transações associadas a ela. Essa medida protege a consistência do histórico financeiro, evitando a perda de vínculos importantes entre os dados.

Esses mecanismos tornam o banco mais seguro e inteligente, reduzindo erros operacionais e reforçando a integridade dos dados de forma automática.

INDICES

Índice simples na tabela Transação para o campo data

```
CREATE INDEX idx_transacao_data ON Transacao(data);
```

Justificativa: Esse índice acelera consultas que filtram ou ordenam transações por data, muito comum para relatórios mensais, anuais ou por período.

Índice simples na tabela Transacao para o campo id_categoria

```
CREATE INDEX idx_transacao_id_categoria ON Transacao(id_categoria);
```

Justificativa: Melhora a performance de consultas que agrupam ou filtram transações por categoria (exemplo: total de despesas por categoria).


Índice composto na tabela Transacao para os campos id_conta e data

```
CREATE INDEX idx_transacao_conta_data ON Transacao(id_conta, data);
```

Justificativa: Este índice é útil para consultas que buscam transações por conta em um intervalo de datas, como no caso da view de saldo mensal por conta. O índice composto permite que o banco filtre rapidamente pela conta e depois ordene/filtre por data, tornando a consulta mais eficiente.

OUTRAS CONSULTAS SQL

Com base no arquivo **Consultas.sql**, foi elaborada uma série de instruções `SELECT` com foco em análises financeiras detalhadas, utilizando filtros, agregações e junções entre múltiplas tabelas. As consultas permitem extrair informações relevantes para o usuário, como os totais de

	BANCO DE DADOS I		Código: 01-2025	
	DOCUMENTO DE ESPECIFICAÇÃO DE DADOS		Folhas 09	Revisão: 01

despesas por categoria em determinado mês e ano, o saldo atualizado das contas, o comparativo entre orçamento planejado e gastos reais, e a identificação de usuários inativos nos últimos 30 dias.

Outras consultas trazem rankings, como o das maiores despesas por usuário em 2025, e projeções de metas financeiras com base em receitas acumuladas. Também há comandos para monitoramento de parcelas com vencimentos futuros, estatísticas mensais de transações agrupadas por tipo, e a contagem de movimentações de alto valor por conta e data. Além disso, é feita uma análise da média diária de despesas dos últimos sete dias.

Essas instruções demonstram o uso avançado de funções como SUM, AVG, COUNT, COALESCE e DATEADD, além de cláusulas como GROUP BY, HAVING, ORDER BY, LIMIT e subconsultas correlacionadas. O conjunto de consultas oferece uma base sólida para geração de relatórios, painéis analíticos e tomada de decisão baseada em dados.

CONCLUSÃO

O projeto de banco de dados descrito neste documento atingiu plenamente o objetivo pedagógico de aplicar, de forma integrada, os principais conceitos de modelagem, implementação e governança de dados aprendidos ao longo do curso. A partir de requisitos claramente definidos, foi possível construir um Modelo Entidade-Relacionamento consistente, normalizado até a 3FN, refletindo entidades-chave como USUÁRIO, CONTA, CATEGORIA e TRANSAÇÃO, bem como relacionamentos e restrições que preservam a integridade referencial do sistema.

A tradução desse modelo para scripts DDL resultou em uma estrutura relacional robusta, reforçada por chaves primárias, estrangeiras, checks e gatilhos que automatizam regras de negócio críticas (por exemplo, impedir exclusão de categorias utilizadas ou garantir valores positivos em transações). Em complemento, os comandos DML providenciaram um conjunto representativo de dados, essencial para testes funcionais e para a validação das consultas analíticas propostas.

A criação de views e índices demonstrou preocupação com desempenho e usabilidade: as views resumem indicadores financeiros relevantes, enquanto os índices reduzem a latência de consultas frequentes por data, categoria e conta. Ademais, as diversas consultas SQL ilustram como o banco atende a cenários reais de gestão financeira, desde o cálculo de saldos e orçamentos até projeções de metas e ranking de despesas, evidenciando o potencial do modelo para gerar informação de valor para usuários finais.

Em síntese, o trabalho consolidou competências técnicas essenciais — da concepção conceitual à otimização de consultas — e produziu um repositório de dados coerente, escalável e seguro. Futuras evoluções podem incluir mecanismos de auditoria mais granulares, políticas de segurança avançadas e integrações com ferramentas de visualização para ampliar ainda mais o apoio à tomada de decisão. Entretanto, como se apresenta, o banco já constitui uma base sólida e confiável para aplicações de controle financeiro pessoal, comprovando a eficácia do processo metodológico adotado.