Já pensou como funciona uma biblioteca digital e seu banco de dados? Essa pesquisa foi feita para responder essas perguntas

Trabalho Banco de Dados

Sobre Biblioteca Digital

André Schwartz Filho, Thiago Nestor Afonso, Matheus Marchese Calabro

Sumário

[Como funciona um banco de dados? 2](#_Toc193182517)

[**M.E.R (Modelo Entidade Relacionamento)** 2](#_Toc193182518)

[**D.E.R** 3](#_Toc193182519)

[**Entidades** 3](#_Toc193182520)

[**Relacionamento** 4](#_Toc193182521)

[**Chave Primária** 5](#_Toc193182522)

[**Chave Estrangeira** 5](#_Toc193182523)

[**Índices** 5](#_Toc193182524)

[**Normalização** 6](#_Toc193182525)

[**Transações e ACID** 7](#_Toc193182526)

[**Tipos de Bancos de Dados** 8](#_Toc193182527)

[**Fontes:** 8](#_Toc193182528)

[**Texto base para criação do banco de dados** 9](#_Toc193182529)

[**Tabelas dos Requisitos dos sistemas e dos Requisitos da empresa** 10](#_Toc193182530)

[**Diagrama do Banco de Dados de uma Biblioteca Virtual (Conceitual)** 0](#_Toc193182531)

[**Diagrama do Banco De Dados de uma Biblioteca Virtual (Logico)** 1](#_Toc193182532)

[**Diagrama do Banco de Dados de uma Biblioteca Virtual (Físico)** 2](#_Toc193182533)

# Como funciona um banco de dados?

## **M.E.R (Modelo Entidade Relacionamento)**

Modelo conceitual do diagrama, usado na engenharia de software, descreve os objetos envolvidos em um negócio em sua totalidade.

Tem como intuito ser facilmente interpretável, descrever bem os objetos (entidades), suas características (atributos) e como elas se

relacionam entre si (relacionamentos).

Em linhas gerais,é um modelo que representa de forma abstrata a estrutura de um banco de dados.

Nem sempre é criado um modelo completo para um sistema, pois poderia resultar em um M.E.R muito extenso e de

difícil interpretação, indo contra a ideia principal do M.E.R. Contudo, nesses casos a melhor solução é dividir a modelagem

em partes menores para simplificar.

## **D.E.R**

Representação gráfica guiada pelo M.E.R. Tem como intuito ser uma forma de visualizar as informações menos

abstrata do que o Modelo, o que auxilia muito no momento de desenvolver o sistema e facilitar a comunicação

daquele que analisa as necessidades do sistema e aquele que propriamente implementa no sistema aquilo que foi modelado

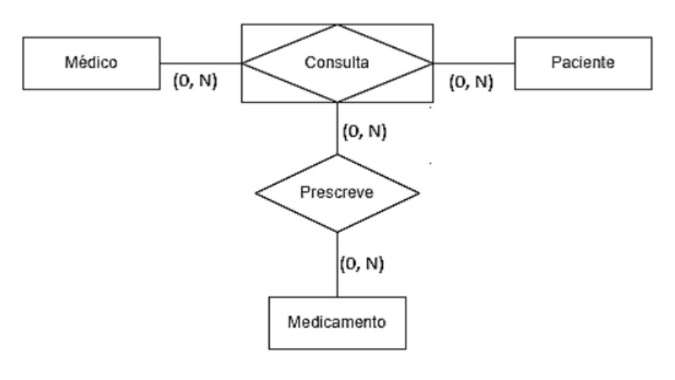


Figura 1: Exemplo de um D.E.R simples

## **Entidades**

Entende-se por entidade, tudo aquilo que se deseja armazenar informações sobre.

Toda a entidade possui atributos, os quais são as informações que caracterizam a entidade.

Exemplo: Entidade - Livro. Atributos: Título, Autor e Data de Lançamento.



Figura 2: Representação de uma entidade

**Tipos de Entidades:**

Entidade Forte: É aquela que possui uma identidade própria e independente. Ela pode ser identificada unicamente por seus próprios atributos, sem depender de outras entidades. Por exemplo, uma entidade "Produto" pode ser identificada por um código exclusivo, sem precisar de uma entidade externa para fazer referência a ela.

Entidade Fraca: Ao contrário da entidade forte, a entidade fraca não pode existir sem estar associada a uma entidade forte. Sua identificação é feita por meio de uma combinação da chave primária da entidade forte com seus próprios atributos. Um exemplo seria uma entidade "Item de Pedido", que depende de um "Pedido" para ser identificada.

Entidade Associativa: Este tipo de entidade é usada para representar um relacionamento entre duas ou mais entidades fortes. Ela não é simplesmente uma ligação entre as entidades, mas sim um objeto que pode ter atributos próprios. Por exemplo, a entidade "Matricula" pode representar o vínculo entre "Aluno" e "Curso", registrando informações sobre o momento da matrícula.

Esses tipos de entidades ajudam a estruturar o banco de dados de maneira mais eficiente, permitindo que a modelagem das relações seja clara e compreensível.

## **Relacionamento**

Em modelagem de banco de dados, relacionamento refere-se à maneira como as entidades se conectam ou interagem entre si. É o laço que une diferentes partes do sistema. Existem três tipos principais de relacionamentos:

Um para Um (1:1): Cada instância de uma entidade está associada a uma única instância de outra. Por exemplo, um "passaporte" é dado a um único "cidadão".

Um para Muitos (1:N): Uma instância de uma entidade está associada a muitas instâncias de outra, mas a segunda entidade se associa a apenas uma instância da primeira. Por exemplo, um "departamento" pode ter muitos "funcionários", mas cada funcionário pertence a um único departamento.

Muitos para Muitos (N:M): Muitas instâncias de uma entidade podem se associar a muitas instâncias de outra. Por exemplo, "alunos" podem se matricular em muitos "cursos", e cada curso pode ter muitos alunos.

**Cardinalidade**

Descreve as regras numéricas desses relacionamentos. Ou seja, ela define quantas vezes uma entidade pode se relacionar com outra. A cardinalidade está diretamente ligada aos tipos de relacionamento, pois ela especifica a quantidade de instâncias que podem ou devem ser associadas.

Cardinalidade 1:1: Uma instância de uma entidade se relaciona com uma única instância de outra.

Cardinalidade 1:N: Uma instância de uma entidade se relaciona com muitas instâncias de outra.

Cardinalidade N:M: Muitas instâncias de uma entidade podem se associar a muitas instâncias de outra.

A diferença entre relacionamento e cardinalidade é simples: relacionamento define o tipo de associação entre as entidades, enquanto a cardinalidade define a quantidade de instâncias que podem se associar dentro desse relacionamento. Em resumo, o relacionamento diz "como" as entidades se conectam, e a cardinalidade diz "quanto" de uma entidade pode se relacionar com a outra.

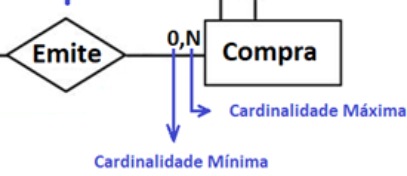


Figura 3: Ilustração de Cardinalidade

## **Chave Primária**

Serve para identificar unicamente cada registro em uma tabela. Ela impede que existam registros duplicados.

Exemplo: Em uma tabela de "Clientes", o ID do Cliente pode ser a chave primária, garantindo que cada cliente tenha um identificador único.

## **Chave Estrangeira**

É um campo em uma tabela que se conecta à chave primária de outra tabela. Ela estabelece relacionamentos entre os dados.

Exemplo: Em uma tabela de "Pedidos", o ID do Cliente pode ser uma chave estrangeira, referenciando o ID do Cliente na tabela "Clientes".

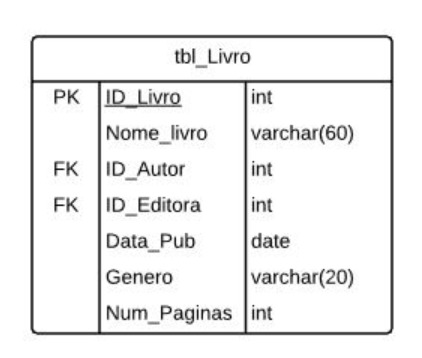


Figura 4: PK (Chave Primária), FK (Chave Estrangeira)

## **Índices**

Os índices em um banco de dados são estruturas que melhoram a velocidade das consultas. Eles funcionam como um catálogo, permitindo que os dados sejam encontrados mais rapidamente sem precisar percorrer toda a tabela.

**Tipos de Índices:**

**Índice Primário**: Criado automaticamente quando uma chave primária é definida. Ele garante que os valores da chave sejam únicos e organizados.

**Índice Secundário**: Criado manualmente para otimizar consultas frequentes em colunas específicas.

**Índice Clusterizado**: Ordena fisicamente os dados na tabela conforme o índice, sendo útil para pesquisas sequenciais.

**Índice Não Clusterizado**: Mantém uma estrutura separada para os índices, permitindo múltiplos índices em uma tabela.

**Exemplo**: Um índice em uma coluna "CPF" na tabela de clientes pode acelerar buscas específicas por um determinado CPF.

## **Normalização**

A normalização é o processo de organizar os dados de um banco para reduzir redundância e melhorar a integridade. Ela divide grandes tabelas em tabelas menores e estabelece relacionamentos entre elas.

**Formas Normais:**

1ª Forma Normal (1FN): Elimina grupos repetitivos, garantindo que cada coluna armazene valores atômicos (indivisíveis).

2ª Forma Normal (2FN): Elimina dependências parciais, ou seja, todos os atributos devem depender inteiramente da chave primária.

3ª Forma Normal (3FN): Remove dependências transitivas, garantindo que os atributos dependam diretamente da chave primária e não de outros atributos.

Exemplo: Uma tabela contendo nome do cliente e vários telefones pode ser dividida em "Clientes" e "Telefones", relacionando-as para evitar duplicação.

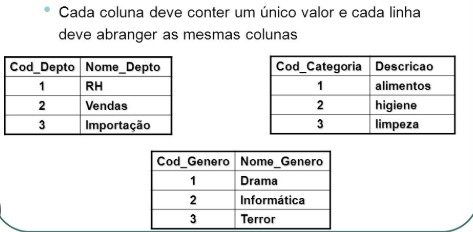


Figura 5: Exemplo de Normalização

## **Transações e ACID**

As transações no banco de dados garantem que um conjunto de operações seja executado corretamente. O princípio ACID assegura:

Confiabilidade: A transação é concluída por inteiro ou revertida completamente em caso de falha.

Consistência: Garante que o banco permaneça em um estado válido antes e depois da transação.

Isolamento: Transações concorrentes não interferem umas nas outras.

Durabilidade: Após a confirmação da transação, os dados são armazenados permanentemente.

Exemplo: Em um sistema bancário, ao transferir dinheiro entre contas, ambas as alterações (débito e crédito) devem ocorrer ou nenhuma delas será salva.

## **Tipos de Bancos de Dados**

Relacional: Usa tabelas e SQL para manipular dados. Exemplo: MySQL, PostgreSQL.

Não Relacional (NoSQL): Usa modelos flexíveis como documentos e chave-valor. Exemplo: MongoDB.

Hierárquico: Organiza os dados em estrutura de árvore. Exemplo: IBM IMS.

Orientado a Objetos: Trata os dados como objetos, útil para aplicações complexas. Exemplo: db4o.

Cada tipo de banco de dados é adequado para diferentes necessidades, desde aplicações empresariais até sistemas em tempo real.

## **Fontes:**

DevMedia - MER e DER: Modelagem de Bancos de Dados

W3Schools - Database Normalization

IBM - ACID Transactions

Microsoft Learn - Fundamentos de Banco de Dados

Oracle - Introdução a Banco de Dados

# **Texto base para criação do banco de dados**

Precisamos de um sistema de banco de dados para a nossa biblioteca digital.

Especificações de software: um sistema de registro de compra, um sistema de identificação e login, um sistema de aluguel de livro, um sistema de histórico, um sistema de recomendação.

**Requisitos do sistema:**

**No sistema de compras é necessário:** 1. uma ferramenta que peça os dados da pessoa como CPF, CEP, Nome e etc…;

2. uma ferramenta de cadastro da forma de pagamento;

3. uma ferramenta de verificação em duas etapas;

4. uma ferramenta de confirmação de compra.

**No sistema de identificação e login é necessário:**

1. uma ferramenta que peça os dados de login como E-mail, senha e outros;

2. uma ferramenta de cadastro caso o usuário não tenha um cadastro efetivo;

3. uma ferramenta de verificação de senha, pode ser em 2 etapas ou não;

4. uma ferramenta de verificação por dispositivo.

**No sistema de aluguel de livro é necessário:**

1. uma ferramenta de login;

2. uma ferramenta de cadastro de compra;

3. uma ferramenta de prazo.

**No sistema de histórico é necessário:**

1. uma ferramenta de registro de compras, aluguéis e livros já lidos.

**No sistema de recomendações é necessário:**

1. uma ferramenta de análise de histórico;

2. um sistema de análise de preferências;

3. uma ferramenta de exposição das recomendações.

**Regras de negócio:**

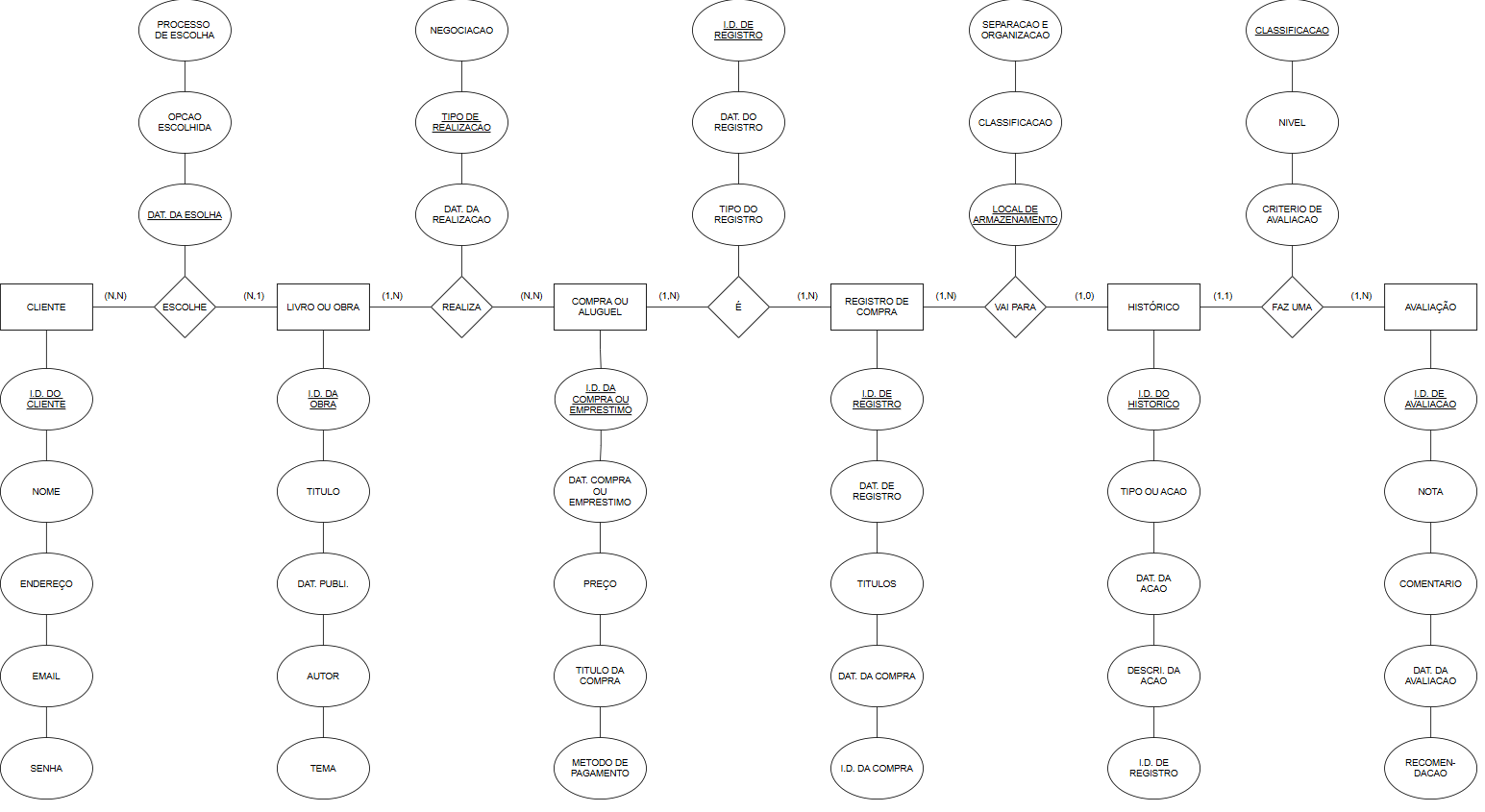
1. As cores tema do banco deve ser branco e vermelho; 2. Os diálogos com o usuário devem ser simples e diretos; 3. A logo da empresa deve estar em todas as telas.

# **Tabelas dos Requisitos dos sistemas e dos Requisitos da empresa**

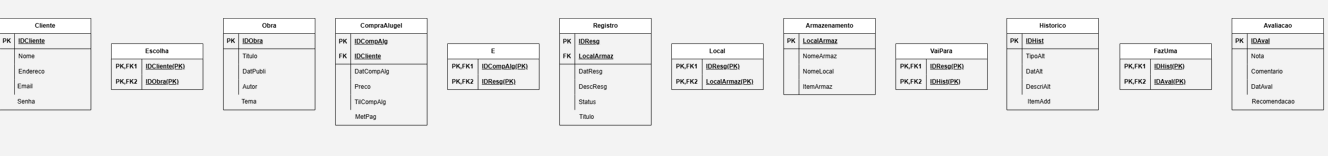
A tabela e usada para melhor entendimento dos Requisitos do Sistema é as Regras da Empresa, isso favorece muito a criação do Diagrama e da uma outra perspectiva. A logo da empresa deve estar em todas as telas.

|  |  |
| --- | --- |
| **Sistema** | **Requisitos** |
| **Sistema de Compras** | Ferramenta que peça os dados do usuário (CPF, CEP, Nome, etc.).  Ferramenta de cadastro da forma de pagamento.  Ferramenta de verificação em duas etapas.  Ferramenta de confirmação de compra. |
| **Sistema de Identificação e Login** | Ferramenta que peça os dados de login (E-mail, senha, etc.).  Ferramenta de cadastro para novos usuários.  Ferramenta de verificação de senha (com ou sem duas etapas).  Ferramenta de verificação por dispositivo. |
| **Sistema de Aluguel de Livro** | Ferramenta de login.  Ferramenta de cadastro de compra.  Ferramenta de prazo. |
| **Sistema de Histórico** | Ferramenta de registro de compras, aluguéis e livros já lidos. |
| **Sistema de Recomendações** | Ferramenta de análise de histórico.  Sistema de análise de preferências.  Ferramenta de exposição das recomendações. |
| **Regras** | **Descrição** |
| **1** | As cores do tema do banco devem ser branco é vermelho. |
| **2** | Os diálogos com o usuário devem ser simples e diretos. |
| **3** | A logo da empresa deve estar em todas as telas. |

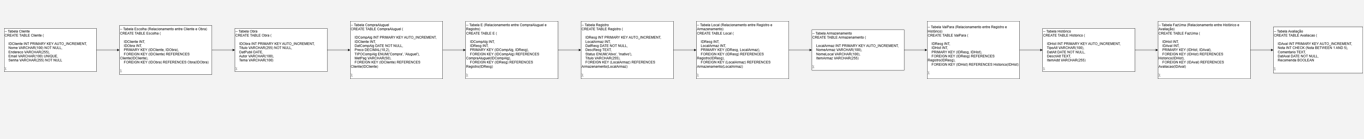
# **Diagrama do Banco de Dados de uma Biblioteca Virtual (Conceitual)**



# **Diagrama do Banco De Dados de uma Biblioteca Virtual (Logico)**



# **Diagrama do Banco de Dados de uma Biblioteca Virtual (Físico)**



# **Binário Eletrônico**

|  |  |
| --- | --- |
| Siglas | Significado |
| D.E.R é M.E.R | D.E.R:  - (Diagrama Entidade-Relacionamento)  M.E.R:  - (Modelo Entidade-Relacionamento) |
| P.K é F.K | P.K:  - Primary Key  F.K:  - Foreign Key |
| ACID | Atomicidade, Consistência, Isolamento e Durabilidade |
| Id | Código de identificação |
| Dat | Data |
| Comp | Compra |
| Alg | Aluguel |
| Til | Titulo |
| Met | Método |
| Pag | Pagamento |
| Resg | Registro |
| Desc | Descrição |
| Armaz | Armazenamento |
| Alt | Alteração |
| Add | Adicionado |
| His | Histórico |
| Aval | Avaliação |
| Int | Valor Inteiro |
| Enum | Enumeração |

Tabelas

|  |  |
| --- | --- |
| Tabela 01 | Requisitos da Empresa |
| Tabela 02 | Regras de Negocio |

Diagramas

|  |  |
| --- | --- |
| Diagrama 01 | Conceitual |
| Diagrama 02 | Lógico |
| Diagrama 03 | Físico |