



Universidade Federal da Paraíba
Centro de Informática
Banco de Dados I
Professor Marcelo Iury

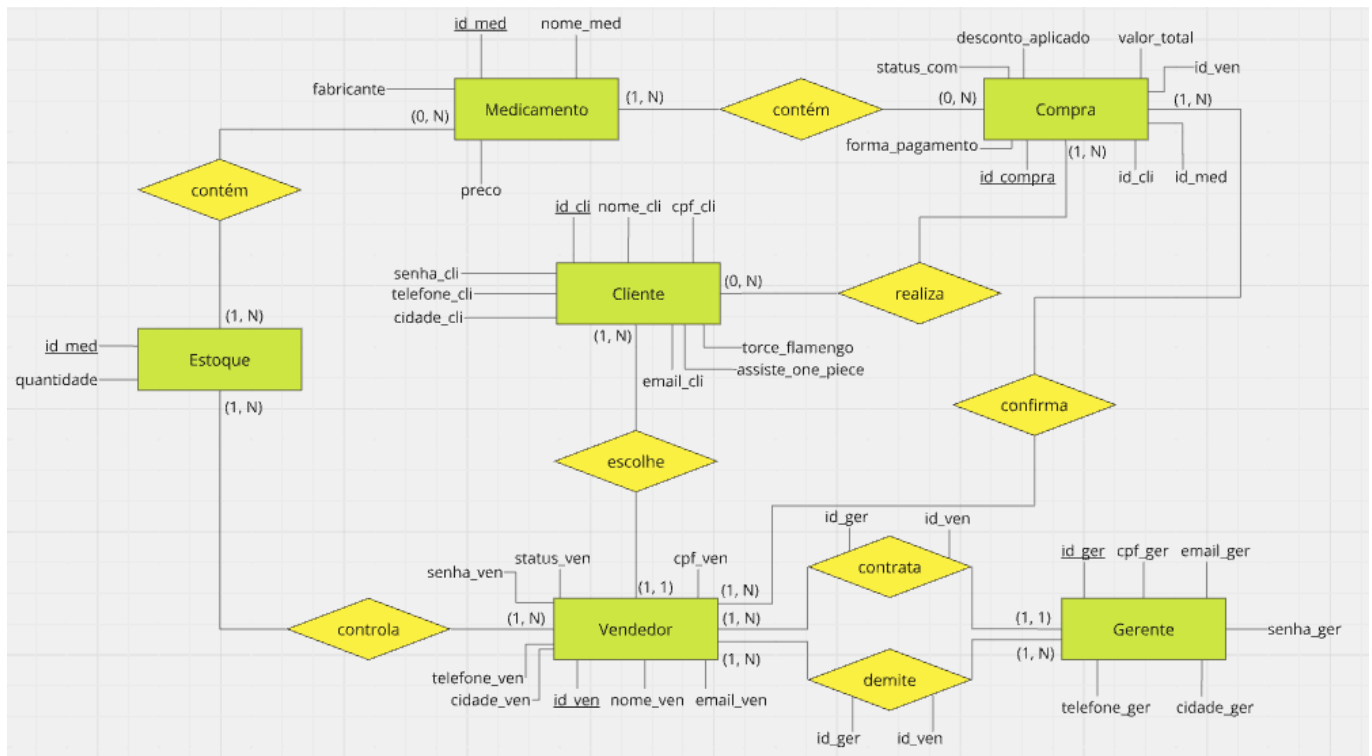


Documentação do banco de dados de uma farmácia

João Victor Oliveira de Lima - 20190027718 (Turma 1)
Luiz Gustavo Oliveira de Souza - 20230146030 (Turma 2)
Thaís Melquíades Macêdo - 20190021376 (Turma 2)

João Pessoa
19 de outubro de 2024

Documentação Modelo ER



Medicamento:

A tabela obedece à segunda e terceira formas normais, pois cada medicamento possui uma chave única (id_med), e todos os atributos dependem unicamente dessa chave.

Estoque:

A tabela foi criada separadamente para relacionar Medicamento com sua quantidade disponível. Isso evita que a quantidade seja duplicada em várias linhas da tabela Medicamento, obedecendo às regras de normalização. A relação entre Medicamento e Estoque é 1:N, já que um medicamento pode estar em apenas um estoque por vez, mas há muitos medicamentos em estoque.

Vendedor e Gerente:

O modelo ER dessas entidades foi convertido diretamente para as tabelas Vendedor e Gerente no modelo relacional. O relacionamento entre Gerente e Vendedor é expresso nas relações contrata e demite, onde o gerente responsável por contratar ou demitir o vendedor pode ser referenciado, mas esses atributos foram mantidos diretamente em Vendedor, já que o relacionamento segue a segunda e a terceira formas normais.

Cliente:

A tabela Cliente foi criada com atributos próprios, como nome, CPF, e informações de contato, além de atributos específicos para aplicar descontos, como "torce_flamengo" e

"assiste_one_piece". A relação Cliente-Compra foi feita referenciando a chave estrangeira id_cli na tabela Compra.

Compra:

O relacionamento entre Cliente, Vendedor e Medicamento foi implementado diretamente na tabela Compra, com a inclusão das chaves estrangeiras id_cli, id_ven, e id_med. A decomposição dessas informações em uma única tabela facilita o controle e a normalização das operações de compra. O atributo forma_pagamento reflete os diferentes métodos de pagamento, e o atributo desconto_aplicado foi adicionado para aplicar os descontos conforme as condições do cliente. A tabela Compra obedece à terceira forma normal, pois todos os atributos são dependentes diretamente da chave primária id_compra.

Modelo Relacional

- Medicamento (id_med, nome_med, fabricante, preco)
- Estoque (id_med, quantidade) *id_med referencia Medicamento:id_med*
- Vendedor (id_ven, nome_ven, cpf_ven, email_ven, senha_ven, status_ven, telefone_ven, cidade_ven)
- Gerente (id_ger, nome_ger, cpf_ger, email_ger, senha_ger, telefone_ger, cidade_ger)
- Cliente (id_cli, nome_cli, cpf_cli, email_cli, senha_cli, telefone_cli, cidade_cli, torce_flamengo, assiste_one_piece)
- Compra (id_compra, id_cli, id_ven, id_med, forma_pagamento, status_com, desconto_aplicado, valor_total) *id_cli referencia Cliente:id_cli, id_ven referencia Vendedor:id_ven, id_med referencia Medicamento:id_med*

Documentação DDL, DML e DQL

Criação Medicamento:

```
CREATE TABLE IF NOT EXISTS `medicamento` (  
  `id_med` int NOT NULL AUTO_INCREMENT,  
  `nome_med` varchar(50) DEFAULT NULL,  
  `fabricante` varchar(50) DEFAULT NULL,  
  `estoque` int DEFAULT NULL,  
  `preco` decimal(10,2) DEFAULT NULL,  
  PRIMARY KEY (`id_med`),  
  UNIQUE KEY `idMedicamento_UNIQUE` (`id_med`)  
);
```

Criação Cliente:

```
CREATE TABLE IF NOT EXISTS `cliente` (  
  `id_cli` int NOT NULL AUTO_INCREMENT,  
  `cpf_cli` varchar(11) NOT NULL,  
  `nome_cli` varchar(255) NOT NULL,
```

```

`email_cli` varchar(255) NOT NULL,
`senha_cli` varchar(255) NOT NULL,
`telefone_cli` varchar(20) DEFAULT NULL,
`cidade_cli` varchar(100) NOT NULL,
`torce_flamengo` tinyint(1) NOT NULL,
`assiste_one_piece` tinyint(1) NOT NULL,
PRIMARY KEY (`id_cli`),
UNIQUE KEY `email_cli` (`email_cli`),
UNIQUE KEY `cpf_cli_UNIQUE` (`cpf_cli`),
UNIQUE KEY `id_cli_UNIQUE` (`id_cli`)
);

```

Criação Vendedor:

```

CREATE TABLE IF NOT EXISTS `vendedor` (
  `id_ven` int NOT NULL AUTO_INCREMENT,
  `cpf_ven` varchar(11) NOT NULL,
  `nome_ven` varchar(255) NOT NULL,
  `email_ven` varchar(255) NOT NULL,
  `senha_ven` varchar(255) NOT NULL,
  `telefone_ven` varchar(20) DEFAULT NULL,
  `cidade_ven` varchar(100) NOT NULL,
  PRIMARY KEY (`id_ven`),
  UNIQUE KEY `email_ven` (`email_ven`),
  UNIQUE KEY `cpf_ven_UNIQUE` (`cpf_ven`),
  UNIQUE KEY `id_ven_UNIQUE` (`id_ven`)
);

```

Criação Compra:

```

CREATE TABLE IF NOT EXISTS `compra` (
  `id_compra` INT AUTO_INCREMENT PRIMARY KEY,
  `id_cli` INT,
  `id_ven` INT,
  `data_compra` TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  `id_med` INT,
  `desconto_aplicado` DECIMAL(10, 2),
  `valor_total` DECIMAL(10,2),
  `forma_pagamento` tinyint(1),
  `status_pagamento` tinyint(1),
  FOREIGN KEY (`id_cli`) REFERENCES cliente(`id_cli`),
  FOREIGN KEY (`id_ven`) REFERENCES vendedor(`id_ven`),
  FOREIGN KEY (`id_med`) REFERENCES medicamento(`id_med`)
);

```

Criação Gerente:

```
CREATE TABLE IF NOT EXISTS `gerente` (  
  `id_ger` int NOT NULL AUTO_INCREMENT,  
  `cpf_ger` varchar(11) NOT NULL,  
  `nome_ger` varchar(255) NOT NULL,  
  `email_ger` varchar(255) NOT NULL,  
  `senha_ger` varchar(255) NOT NULL,  
  `cidade_ger` varchar(100) NOT NULL,  
  `telefone_ger` varchar(20) DEFAULT NULL,  
  PRIMARY KEY (`id_ger`),  
  UNIQUE KEY `id_ger_UNIQUE` (`id_ger`),  
  UNIQUE KEY `cpf_ger_UNIQUE` (`cpf_ger`),  
  UNIQUE KEY `email_ger_UNIQUE` (`email_ger`)  
);
```