



DAS

Universidade Federal de Santa Catarina
Departamento de Automação e Sistemas
Engenharia de Controle e Automação
Fundamento de Sistemas de Banco de Dados

Modelagem do Banco de Dados de um Pet Shop

Alunos:

Gabriel Martins de Souza

Guilherme dos Santos

Henrique Roos Baltezan

Victor Soares Stefenon

Professor:

Ronaldo dos Santos Mello

Florianópolis

Maio de 2021

1 Texto de Requisitos

Um pet shop vende produtos. Esses produtos possuem um código do produto, quantidade em estoque, nome, preço e o fornecedor. O banco de dados também guarda o nome, o CNPJ, o endereço, o telefone, o e-mail e o tipo de produto oferecido por esse fornecedor. Esses produtos são fornecidos a um determinado custo e em uma quantidade. Além disso, os produtos podem ou não ser especificados como ração, medicamento, casinha ou roupa. Quando for ração, deve conter as raças e idade recomendada do animal. Quando for medicamento, deve conter via de aplicação, indicações e porte recomendado. Quando for casinha, deve conter o porte recomendado do animal, o formato da casinha e a cor. Quando for roupa, deve conter o tamanho, a cor e o tipo de roupa. Quando um produto é vendido, guarda-se a informação da data da venda, bem como a quantidade de cada produto vendido. O pet shop também oferece serviços, que podem ser consultas, banhos, tosas, ou banhos e tosas. Cada serviço tem um preço e deve estar associado a um pet, um funcionário, uma data, uma hora e pode ser avulso ou fazer parte de um plano. No caso de uma consulta, é necessário que o funcionário seja um veterinário. O pet shop oferece o serviço de buscar e levar o pet até a residência do dono após ou antes dos serviços. Os funcionários possuem um CPF, um nome, uma data de nascimento, um telefone, um endereço, um e-mail, um salário e um cargo. Quando o funcionário for um veterinário, também deve conter o CRMV do veterinário. Os pets possuem um dono, um nome, uma idade, problemas de saúde, peso e uma raça. Cada raça tem uma determinada expectativa de vida. Os donos possuem um CPF, um nome, uma data de nascimento, um telefone, um endereço e um e-mail. Além disso, o pet shop oferece diferentes planos, todos mensais e que devem ser pagos até o dia de vencimento. O preço desse plano varia de acordo com o número de pets que o dono insere no plano, tendo uma redução de custo por pet até um máximo de 3 pets. Existe um plano que oferece apenas banhos semanais (Básico). Outro oferece, além dos banhos, uma tosa mensal (Médio) e um outro plano oferece, além dos banhos e da tosa, uma consulta mensal (Premium). Portanto, existem 9 planos: Básico para 1, para 2 ou para 3 ou mais pets, Médio para 1 pet, para 2 ou para 3 ou mais pets e Premium para 1, para 2 ou para 3 ou mais pets.

2 Modelagem Conceitual

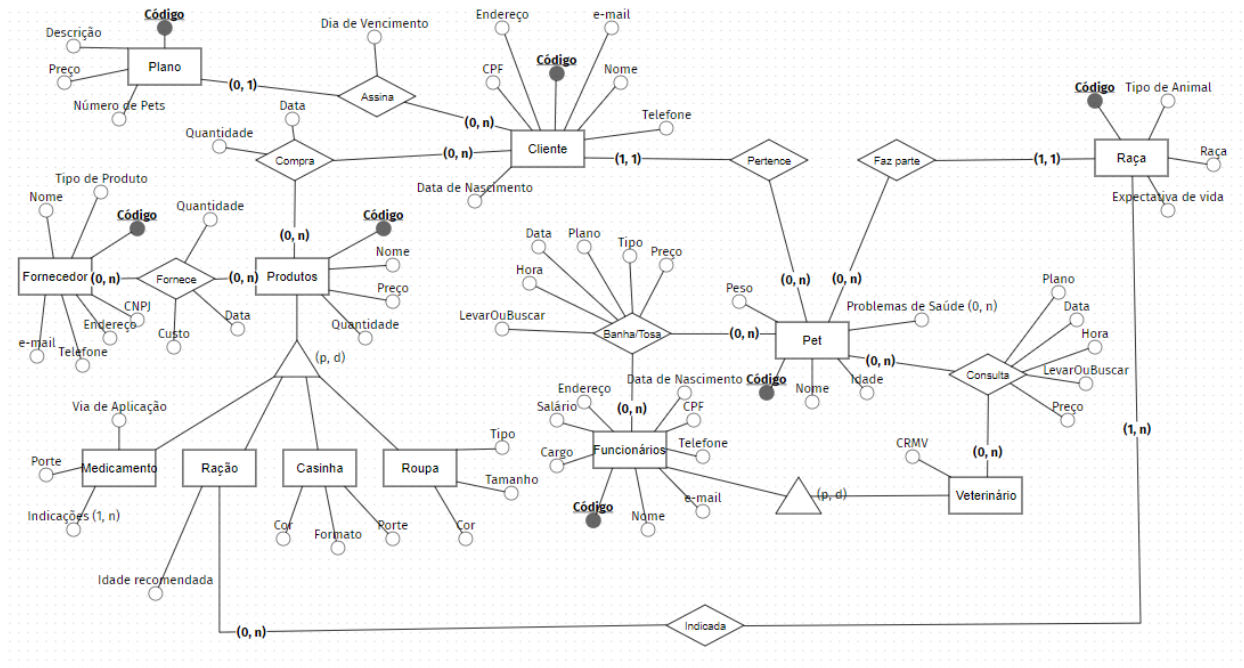


Figura 1: Modelagem Conceitual

3 Modelagem Lógica

As tabelas para a modelagem lógica, com seus respectivos atributos, podem ser vistas abaixo:

Obs.: As chaves primárias foram denotadas com o sublinhado e as chaves estrangeiras com o sobrelinhado.

- Fornecedor(CodForn, Nome, CNPJ, TipoProduto, Endereço, Email, Telefone)

Tabela Fornecedor: Fornecedores dos produtos da PetShop, contendo nome, CNPJ, endereço, e-mail e telefone como informações básicas e o tipo do produto que pode ser um dos seguintes tipos: Medicamento, Ração, Casinha, Roupa, entre outros. A tabela também possui o código do fornecedor como chave primária, representada pelo CodForn.

- ComprasFornecedor(NroPedidoForn, CodForn, CodProd, Quantidade, Custo, Data)

Tabela ComprasFornecedor: Produtos, dos fornecedores, comprados pela PetShop, contendo a data da compra, bem como a quantidade e o custo de cada produto. A tabela também possui o número do pedido (esse número foi inserido para permitir que o PetShop possa comprar novamente um mesmo produto do mesmo fornecedor) como chave primária e os códigos do fornecedor e produto como chaves primárias e estrangeiras (provenientes das tabelas Fornecedor e Produtos, respectivamente).

- Produtos(CodProd, Nome, Preço, Quantidade)

Tabela Produtos: Produtos que estão no estoque da Petshop, contendo o nome, preço e quantidade de cada produto. A tabela também possui o código do produto como chave primária.

- Compras(NroPedido, CodProd, CodCliente, QtdProduto, DataCompra)

Tabela Compras: Compras, realizadas na PetShop, por clientes da PetShop, contendo a quantidade do produto. A tabela também possui o número do pedido (esse número foi inserido para permitir que o mesmo cliente possa repetir compras do mesmo produto) como chave primária, além do código do cliente e código do produto como chaves primárias e estrangeiras (provenientes das tabelas Clientes e Produtos, respectivamente).

- Cliente(CodCliente, Nome, CPF, Email, Endereco, Telefone, DataNascimento, CodPlano, VencimentoPlano)

Tabela Cliente: Cadastro dos clientes da loja contendo nome, CPF, e-mail, endereço, telefone, data de nascimento e vencimento do plano como informações do cliente. A tabela também possui o código do cliente como chave primária e código do plano como estrangeira, proveniente da tabela Planos. A coluna referente ao vencimento do plano poderia ter sido substituída por uma tabela representando o relacionamento *Assina* (vista na Figura 1), porém, neste caso, ficaríamos com uma tabela com apenas um atributo além de suas chaves primárias (código do plano e do cliente). Portanto, com essa escolha, evitou-se a criação desta nova tabela.

- Pet(CodPet, Nome, CodCliente, CodR, Peso, Idade)

Tabela Pet: Cadastro de cada Pet na PetShop contendo seu nome, peso e idade. A tabela também possui o código do Pet como chave primária e o código do cliente e da sua raça como chaves estrangeiras provenientes das tabelas Cliente e Raça, respectivamente.

- ProblemasSaude(CodPet, Problema)

Tabela ProblemasSaude: Problemas que cada Pet tem, contendo a descrição dos problemas do Pet como chave primária e o código do Pet como chave primária e estrangeira, proveniente da tabela Pet. A necessidade da criação dessa tabela surge do fato de problemas de saúde ser um atributo multivalorado da entidade Pet no modelo Conceitual.

- Raca(CodR, TipoAnimal, Raca, ExpectativaVida)

Tabela Raca: Raças de cada pet, com o tipo do animal (cachorro, gato, papagaio, etc.), a raça deste animal e sua expectativa de vida. A PetShop só trabalha com a divisão de raças para cachorros e gatos. A raça dos demais animais é tratada como sendo igual ao seu próprio tipo.

- Consulta(CodPet, CRMV, Data, Hora, Preco, Plano, LevarOuBuscar)

Tabela Consulta: Registro histórico de consultas realizadas no PetShop, com o preço da consulta, o plano do cliente que a contratou e se o PetShop precisa levar e/ou buscar o pet (pode assumir os valores "Levar", "Buscar", "Levar e Buscar" ou nulo, quando nenhuma das ações é necessária e o próprio dono as faz). A chave primária foi definida como a composição do código do pet, o CRMV do veterinário, a data e a hora da consulta porque um mesmo veterinário pode atender um mesmo pet diversas vezes, mas não de maneira simultânea.

- BanhoTosa(CodPet, CodF, Data, Hora, Tipo, Preco, Plano, LevarOuBuscar)

Tabela BanhoTosa: Registro histórico de serviços de banho e/ou tosa realizados pelo PetShop. Contém a data e hora da realização do serviço, o tipo (banho, tosa ou banho e tosa), o preço do serviço, o plano do cliente que contratou o serviço e se o PetShop precisa levar e/ou buscar o pet, que funciona

da mesma forma que na tabela *Consulta*. A chave primária foi definida dessa forma pelo mesmo motivo da tabela *Consulta*.

- Funcionarios(CodF, Nome, Endereco, DataNascimento, Email, CPF, telefone, cargo, salario, CRMV)

Tabela Funcionarios: Cadastro dos funcionários empregados pela loja, contendo nome, endereço, data de nascimento, e-mail, CPF, telefone, cargo, salário e, caso o funcionário seja veterinário, o seu CRMV. Salienta-se que a inclusão da especialização veterinário, do modelo conceitual, na tabela funcionários foi feita com o intuito de evitar a criação de uma nova tabela que conteria apenas o código de funcionário e o CRMV do veterinário.

- Racao(CodProd, idade)

Tabela Racao: Especificação dos produtos do tipo "Racao", com a idade indicada para cada ração. Esta tabela e as tabelas das demais especificações de produtos foram criadas por possuírem diversos atributos diferentes. Dessa forma, caso todos os atributos fossem incluídos na tabela *Produtos*, haveria um grande número de dados nulos.

- RacaoIndicada(CodProd, CodR)

Tabela RacaoIndicada: Relação entre a tabela *Raca* e a tabela *Ração*. Essa tabela existe pelo fato de que, no modelo conceitual, a relação é dada de muitos pra muitos.

- Medicamentos(CodProd, Porte, ViaAplicacao)

Tabela Medicamentos: Especificação dos produtos do tipo "Medicamento", com informação do porte indicado e da via de aplicação. A tabela foi criada pela mesma razão da tabela *Racao*.

- Indicacoes(CodProd, Indicacao)

Tabela Indicacoes: Relação de problemas de saúde para os quais cada medicamento é indicado. A tabela foi criada porque um medicamento pode ser indicado para diversos problemas, bem como um mesmo problema pode ter vários medicamentos indicados (atributo multivalorado no modelo conceitual).

- Casinha(CodProd, Cor, Formato, Porte)

Tabela Casinha: Especificação dos produtos do tipo "Casinha", com cor, formato da casa e porte indicado. A tabela foi criada pela mesma razão da tabela *Racao*.

- Roupa(CodProd, Cor, Tamanho, Tipo)

Tabela Roupas: Especificação dos produtos do tipo "Roupa", com a cor, o tamanho e o tipo destas. A tabela foi criada pela mesma razão da tabela *Racao*.

- Planos(CodPlano, Descricao, Preco)

Tabela Planos: Planos disponíveis para os clientes do PetShop, com a descrição do plano (Básico, Médio ou Premium e número de pets) e o preço do plano. Devido a isso o atributo número de pets, presente no modelo conceitual (Figura 1), foi retirada para evitar redundância. Salienta-se que a tabela *Planos* poderia ter sido incluída em *Clientes*, contudo, haveria um grande número de campos nulos e redundância.

4 Criação do banco de dados em SQL

A construção das tabelas em SQL pode ser vista nos códigos a seguir:

Obs.: A definição de colunas não-nulas e únicas que não são chaves foi feita de acordo com a interpretação do problema por parte dos integrantes do grupo.

4.1 Tabelas

```
CREATE TABLE Fornecedor (CodForn int NOT NULL UNIQUE,  
                           Nome varchar(40) NOT NULL,  
                           CNPJ numeric(14) NOT NULL UNIQUE,  
                           TipoProduto varchar(20) NOT NULL check(TipoProduto in ('medica-  
mentos', 'casinha', 'racaio', 'roupa', 'outros')),  
                           Endereco varchar(100),  
                           Email varchar(40),  
                           Telefone numeric(12) NOT NULL,  
                           PRIMARY KEY (CodForn))
```

```
CREATE TABLE ComprasFornecedor(NroPedidoForn int NOT NULL,  
                                 CodForn int NOT NULL,  
                                 CodProd int NOT NULL,  
                                 Quantidade int NOT NULL check(Quantidade > 0),  
                                 Custo numeric(6,2) NOT NULL check(Custo > 0),  
                                 PRIMARY KEY (NroPedidoForn, CodForn, CodProd),  
                                 FOREIGN KEY (CodForn) REFERENCES Fornecedor,  
                                 FOREIGN KEY (CodProd) REFERENCES Produtos)
```

```
CREATE TABLE Produtos (CodProd int NOT NULL UNIQUE,  
                         Nome varchar(40) NOT NULL,  
                         Preco numeric(6,2) NOT NULL, check(Preco > 0)  
                         Quantidade int check(Quantidade > 0),  
                         PRIMARY KEY (CodProd))
```

```
CREATE TABLE Compras (NroPedido int NOT NULL,  
                        CodProd int NOT NULL,  
                        CodCliente int NOT NULL,  
                        QtdProduto int NOT NULL check(QtdProduto > 0),  
                        DataCompra date,  
                        PRIMARY KEY (NroPedido, CodProd, CodCliente),  
                        FOREIGN KEY (CodCliente) REFERENCES Cliente,  
                        FOREIGN KEY (CodProd) REFERENCES Produtos)
```

```

CREATE TABLE Cliente (CodCliente int NOT NULL UNIQUE,
                        Nome varchar(40) NOT NULL,
                        CPF numeric(11) NOT NULL UNIQUE,
                        Email varchar(40),
                        Endereco varchar(100),
                        Telefone numeric(12) NOT NULL,
                        DataNascimento date,
                        VencimentoPlano date,
                        CodPlano int,
                        PRIMARY KEY (CodCliente),
                        FOREIGN KEY (CodPlano) REFERENCES Planos)

CREATE TABLE Pet (CodPet int NOT NULL UNIQUE,
                  Nome varchar(20) NOT NULL,
                  CodCliente int NOT NULL,
                  CodR int NOT NULL,
                  Peso int check(Peso > 0),
                  Idade int check(idade > 0),
                  PRIMARY KEY (CodPet),
                  FOREIGN KEY (CodCliente) REFERENCES Cliente,
                  FOREIGN KEY (CodR) REFERENCES Raca)

CREATE TABLE ProblemaSaude (CodPet int NOT NULL,
                             Problema varchar(20) NOT NULL,
                             PRIMARY KEY (CodPet, Problema),
                             FOREIGN KEY (CodPet) REFERENCES Pet)

CREATE TABLE Raca (CodR int NOT NULL UNIQUE,
                    TipoAnimal varchar(20) NOT NULL,
                    Raca varchar(30) NOT NULL,
                    ExpectativaVida int NOT NULL check(ExpectativaVida > 0),
                    PRIMARY KEY (CodR))

CREATE TABLE Consulta (CodPet int NOT NULL,
                        CRMV numeric(5) NOT NULL,
                        Data date NOT NULL,
                        Hora time NOT NULL,
                        CodPlano int,
                        Preco numeric(6,2) NOT NULL check(Preco > 0),
                        LevarOuBuscar varchar(15) check(LevarOuBuscar in ('Levar', 'Buscar',
                        'Levar e Buscar')),
                        PRIMARY KEY (CodPet, CRMV, Data, Hora),
                        FOREIGN KEY (CodPet) REFERENCES Pet,
                        FOREIGN KEY (CRMV) REFERENCES Funcionarios,
                        FOREIGN KEY (CodPlano) REFERENCES Planos)

```

```

CREATE TABLE Racao (CodProd int NOT NULL,
                     IdadeRecomendada int NOT NULL check( IdadeRecomendada > 0),
                     PRIMARY KEY (CodProd),
                     FOREIGN KEY (CodProd) REFERENCES Produtos)

CREATE TABLE RacaoIndicada (CodProd int NOT NULL,
                             CodR int NOT NULL,
                             PRIMARY KEY (CodProd, CodR),
                             FOREIGN KEY (CodProd) REFERENCES Produtos,
                             FOREIGN KEY (CodR) REFERENCES Raca)

CREATE TABLE Medicamentos (CodProd int NOT NULL,
                             Porte varchar(10) NOT NULL check(Porte in ('pequeno', 'medio',
                             'grande')),
                             ViaAplicacao varchar(20) NOT NULL,
                             PRIMARY KEY (CodProd),
                             FOREIGN KEY (CodProd) REFERENCES Produtos)

CREATE TABLE Indicacoes (CodProd int NOT NULL,
                           Indicacao varchar(25) NOT NULL,
                           PRIMARY KEY (CodProd, Indicacao),
                           FOREIGN KEY (CodProd) REFERENCES Produtos)

CREATE TABLE Casinha (CodProd int NOT NULL,
                       Cor varchar(20) NOT NULL,
                       Formato varchar(20) NOT NULL,
                       Porte varchar(20) NOT NULL check(Porte in ('pequeno', 'medio', 'grande')),
                       PRIMARY KEY (CodProd),
                       FOREIGN KEY (CodProd) REFERENCES Produtos)

CREATE TABLE Roupa (CodProd int NOT NULL,
                     Cor varchar(20) NOT NULL,
                     Tamanho varchar(2) NOT NULL check(Tamanho in ('PP', 'P', 'M', 'G',
                     'GG', 'U')),
                     Tipo varchar(20) NOT NULL check(Tipo in ('sapato', 'coleira', 'agasalho',
                     'gravata', 'laco', 'chapeu', 'outras')),
                     PRIMARY KEY (CodProd),
                     FOREIGN KEY (CodProd) REFERENCES Produtos)

```



```

CREATE TABLE BanhoTosa (CodPet int NOT NULL,
                          CodF int NOT NULL,
                          CodPlano int,
                          Data date NOT NULL,
                          Hora time NOT NULL,
                          Tipo varchar(40) NOT NULL check(Tipo in ('Banho', 'Tosa', 'Banho e
                          Tosa')),
                          Preco numeric(6,2) NOT NULL check(Preco > 0),
                          LevarOuBuscar varchar(15) check(LevarOuBuscar in ('Levar', 'Buscar',
                          'Levar e Buscar')),
                          PRIMARY KEY (CodPet, CodF, Data, Hora),
                          FOREIGN KEY (CodPet) REFERENCES Pet,
                          FOREIGN KEY (CodF) REFERENCES Funcionarios
                          FOREIGN KEY (CodPlano) REFERENCES Planos)

CREATE TABLE Funcionarios (CodF int NOT NULL UNIQUE,
                             Nome varchar(40) NOT NULL,
                             CPF numeric(11) NOT NULL UNIQUE,
                             Email varchar(40),
                             Endereco varchar(100), NOT NULL
                             Telefone numeric(12), NOT NULL
                             DataNascimento date,
                             Cargo varchar(20) NOT NULL,
                             Salario numeric(7,2) NOT NULL check(Salario > 0),
                             CRMV numeric(5) UNIQUE,
                             PRIMARY KEY (CodF))

CREATE TABLE Planos (CodPlano int NOT NULL UNIQUE,
                      Descricao varchar(25) NOT NULL check(Descricao in ('Basico 1 pet', 'Basico
                      2 pets', 'Basico 3 ou mais pets', 'Medio 1 pet', 'Medio 2 pets', 'Medio 3 ou
                      mais pets', 'Premium 1 pet', 'Premium 2 pets', 'Premium 3 ou mais pets')),
                      Preco numeric(6,2) check(Preco > 0),
                      PRIMARY KEY (CodPlano))

```

4.2 Rules

Para garantir que o todo veterinário deva ser cadastrado com seu CRMV é necessário que seja criado uma rule.

```

CREATE RULE CRMVdoVet AS ON INSERT TO Funcionarios
IF NEW.tipo = 'veterinario' AND NEW.CRMV IS NULL THEN
DO INSTEAD NOTHING

```