

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE SÃO PAULO**

RAUL DIAS PEREIRA

Sistema de Gerenciamento de Pet shop

Banco de dados 2

Paulo Giovani

Análise e Desenvolvimento de Sistemas

**CAMPOS DO
JORDÃO 2024**

RESUMO

O projeto visa revolucionar a gestão de um pet shop através da criação de um sistema de banco de dados relacional. Este sistema cobre desde o registro detalhado dos clientes até informações sobre animais atendidos, produtos e serviços oferecidos. A metodologia adotada inicia-se com uma análise dos requisitos e é seguida pela modelagem dos dados. Estima-se que a eficiência operacional do estabelecimento seja amplamente aprimorada, facilitando desde a gestão de estoque até o atendimento ao cliente. A satisfação dos clientes tende a aumentar, uma vez que o sistema possibilitará um atendimento mais personalizado, dinâmico e eficiente, com acesso rápido a informações relevantes, como data de vacinação e retornos de consultas. Assim, este projeto não visa apenas modernizar o funcionamento do pet shop, mas também elevar a experiência do cliente a um novo patamar.

Palavras-Chave: Banco de Dados Relacional; Pet Shop; Modelagem de dados;

ABSTRACT

The project aims to revolutionize pet shop management through the creation of a relational database system. This system covers everything from detailed client registration to information about attended animals, products, and offered services. The adopted methodology begins with a requirements analysis, followed by data modeling. It is estimated that the establishment's operational efficiency will be greatly enhanced, facilitating everything from inventory management to customer service. Customer satisfaction is expected to increase as the system will enable more personalized, dynamic, and efficient service, with quick access to relevant information such as vaccination dates and consultation returns. Thus, this project not only aims to modernize the pet shop's operations but also to elevate the customer experience to a new level.

Keywords: Relational Database; Pet Shop; Data Modeling;

LISTA DE ILUSTRAÇÕES

FIGURA 1 – Modelo conceitual do banco de dados	9
---	----------

SUMÁRIO

1	INTRODUÇÃO	5
1.1	Objetivos	5
1.2	Justificativa	5
1.3	Aspectos Metodológicos	6
1.4	Aporte Teórico	6
2	METODOLOGIA	7
2.1	Considerações iniciais	7
2.2	Ferramenta utilizada para modelagem	7
2.3	Ferramenta utilizada para criação do modelo físico	7
2.4	Projeto de dados	8
2.5	Coleta das Regras de Negócio	8
3	RESULTADOS OBTIDOS	9
3.1	Modelo conceitual	9
3.2	Regras de negócio	9
3.3	Dicionário de Dados	10
3.3.1	Tabela Cliente	10
3.3.2	Tabela Animal	10
3.3.3	Tabela Produto	10
3.3.4	Tabela Serviço	11
3.3.5	Tabela Produto_Serviço	11
3.3.6	Tabela Profissional	11

3.3.7	Tabela Historico_Servico_____	12
3.4	Modelo Físico_____	12
3.5	Dados Inseridos_____	13
3.5.1	Dados na Tabela Cliente_____	13
3.5.2	Dados na Tabela Animal_____	14
3.5.3	Dados na Tabela Produto_____	14
3.5.4	Dados na Tabela Profissional_____	14
3.5.5	Dados na Tabela Servico_____	14
3.5.6	Dados na Tabela Produto_Servico_____	14
3.5.7	Dados na Tabela Historico_Servico_____	15
3.6	Consultas_____	15
4	CONCLUSÃO _____	24
4.1	Conclusões Finais _____	24
4.2	Sugestões de melhoria _____	24
4.3	Considerações finais _____	25

1. INTRODUÇÃO

Este trabalho tem como objetivo desenvolver um sistema de gerenciamento de banco de dados relacional para um pet shop, justificado pela necessidade de eficiência e confiabilidade no armazenamento e gestão de informações. O sistema visa auxiliar na complexidade das operações diárias e melhorar a relação com os clientes, consequentemente aumentando o faturamento. A metodologia utilizada inclui análise de requisitos e modelagem de dados. O embasamento teórico abrange conceitos de sistemas de banco de dados e modelagem relacional.

1.1 Objetivos

Este trabalho tem como objetivo desenvolver um sistema de gerenciamento de banco de dados relacional para um pet shop, justificado pela necessidade de eficiência e confiabilidade no armazenamento e gestão de informações. Para a consecução deste objetivo, foram estabelecidos os seguintes objetivos específicos:

- Realizar uma investigação sobre os atuais sistemas de gerenciamento de banco de dados relacional utilizados em pet shops e outras pequenas empresas.
- Propor melhorias e adaptações específicas para as necessidades de um pet shop.
- Desenvolver um sistema de gerenciamento de banco de dados relacional específico para um pet shop, com foco na eficiência operacional e na satisfação dos clientes.
- Melhorar a eficiência operacional e a satisfação dos clientes, proporcionando uma gestão mais eficaz.
- Facilitar a gestão de estoque e o atendimento ao cliente com acesso rápido a informações relevantes, otimizando as operações diárias e melhorando a relação com os clientes.

1.2 Justificativa

A escolha de um sistema de banco de dados relacional se justifica pela necessidade de um gerenciamento eficaz das informações do pet shop, que abrange desde o cadastro de clientes e animais até o controle de produtos e serviços oferecidos. Um sistema bem estruturado pode proporcionar um atendimento mais personalizado e eficiente, aumentando a satisfação dos clientes e, por consequência, o faturamento do

estabelecimento.

1.3 Aspectos Metodológicos

A metodologia adotada envolve as seguintes etapas:

- **Análise de Requisitos:** Entendimento das necessidades e expectativas do pet shop em relação ao sistema. Inclui entrevistas com funcionários, pesquisas na internet e questionários para identificar as funcionalidades desejadas e os desafios atuais.
- **Modelagem de Dados:** Criação de diagramas ER (Entidade-Relacionamento) para representar as informações e suas relações. Esta etapa envolve a identificação de entidades (clientes, animais, produtos, serviços) e atributos (nome, endereço, espécie).

1.4 Aporte Teórico

O embasamento teórico do projeto inclui:

- **Conceitos de Sistemas de Banco de Dados:** Definições, tipos de bancos de dados, vantagens e desvantagens.
- **Modelagem Relacional:** Técnicas e metodologias para a criação de modelos de dados relacionais, incluindo normalização e desnormalização de dados.

2 . METODOLOGIA

2.1 Considerações Iniciais

O projeto inicia-se com uma análise detalhada das necessidades do pet shop. Esta etapa envolve a realização de entrevistas com funcionários para entender os principais desafios enfrentados e as funcionalidades desejadas no novo sistema. Questionários também foram utilizados para coletar dados quantitativos sobre as operações do pet shop.

2.2 Ferramenta Utilizada para a Criação do Modelo Conceitual

Para a modelagem dos dados, foi utilizada a ferramenta draw.io, que oferece suporte para a criação de diagramas ER (Entidade-Relacionamento) e facilita a visualização das relações entre as tabelas. A notação utilizada foi a pé de galinha, também chamada de Notação de Crow's Foot, conhecida por sua clareza na representação de entidades e relacionamentos.

2.3 Ferramenta Utilizada para a criação do Modelo Físico

Para a elaboração do modelo físico do banco de dados, foi utilizado o Microsoft SQL Server 2022 em conjunto com o SQL Server Management Studio (SSMS) 20, uma interface gráfica robusta e amplamente adotada para gerenciar e interagir com bancos de dados relacionais. Essa combinação de ferramentas oferece suporte a transações, alta disponibilidade e integração com diversas tecnologias, tornando-a ideal para projetos acadêmicos e profissionais.

Requisitos para Instalação:

- **Sistema Operacional:** Windows 10 ou superior, Windows Server 2016 ou superior (64 bits).
- **Processador:** Pelo menos 1,4 GHz com arquitetura x64.
- **Memória RAM:** Mínimo de 2 GB, recomendável 4 GB ou mais.
- **Espaço em Disco:** Aproximadamente 6 GB para instalação básica.
- **.NET Framework:** Versão 4.8 ou superior.
- **Outros:** Conexão com a internet para download do instalador e atualizações.

O Microsoft SQL Server 2022 foi configurado para hospedar o banco de dados físico do projeto, enquanto o SSMS 20 foi utilizado para a criação das tabelas, definição das chaves primárias e estrangeiras, bem como para o gerenciamento das consultas e manipulação dos dados. Essa abordagem garantiu conformidade com os requisitos do sistema proposto e facilitou o controle da integridade dos dados.

2.4 Projeto de Dados

A descrição do projeto de dados incluiu a identificação das entidades principais, como Clientes, Animais, Produtos e Serviços, e a definição de suas respectivas tabelas no banco de dados. Cada entidade foi modelada com atributos específicos, como nome, endereço, data de nascimento, etc., no caso de clientes, e nome, espécie, raça, idade, etc., no caso de animais.

2.5 Coleta das Regras de Negócio

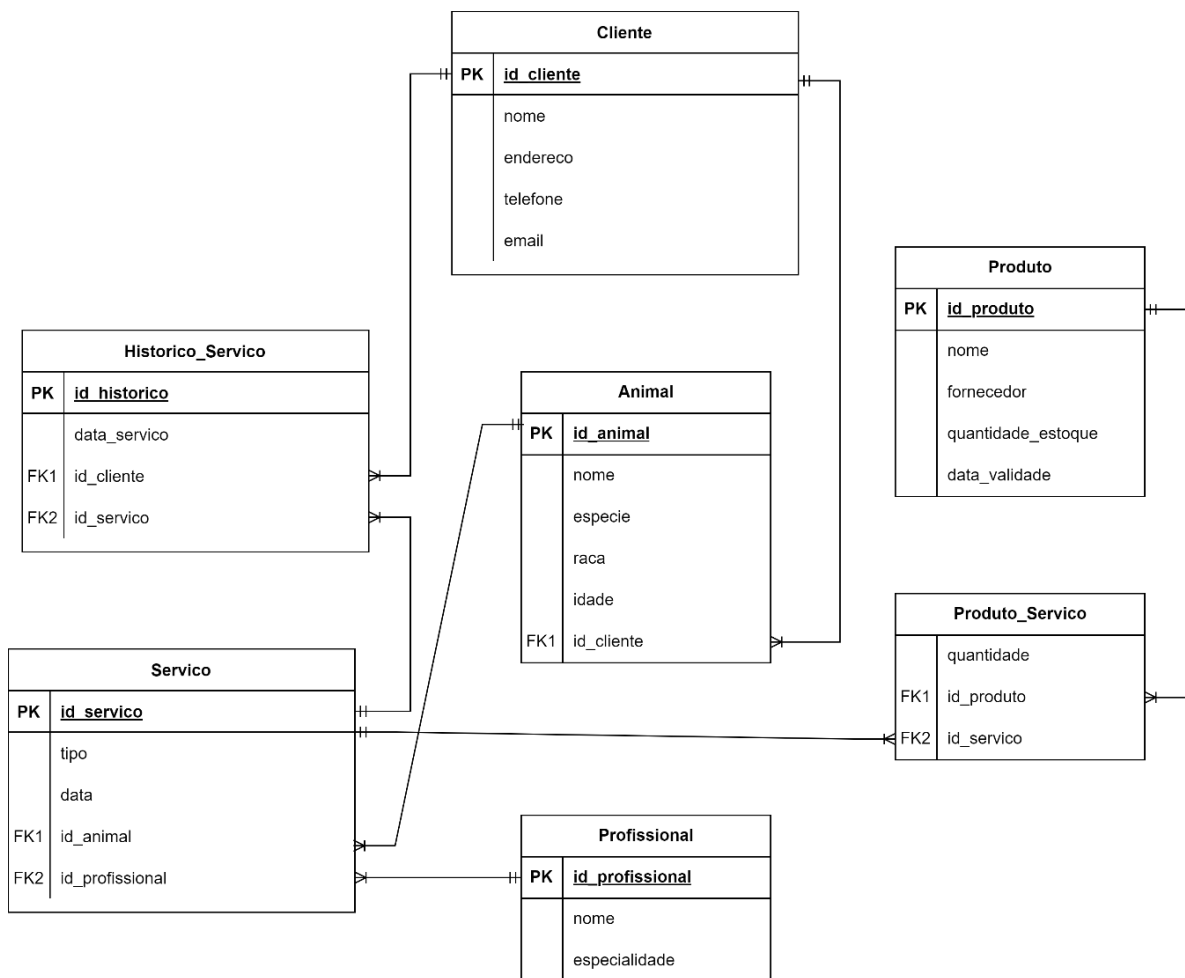
As regras de negócio foram coletadas através de entrevistas com os funcionários do pet shop e análise dos processos atuais de operação. Estas regras incluem:

- **Registro e atualização de informações de clientes e animais:** Cada cliente deve ter um registro único com informações de contato e histórico de serviços. Cada animal deve estar associado a um cliente e conter informações sobre espécie, raça, idade e nome.
- **Controle de estoque de produtos:** Deve haver um controle detalhado de estoque, incluindo informações sobre fornecedores, datas de validade e quantidades disponíveis.
- **Agendamento e registro de serviços prestados:** Deve ser possível agendar e registrar serviços prestados, incluindo datas, tipos de serviços e profissional responsável.

3. RESULTADOS OBTIDOS

3.1 Modelo Conceitual

O modelo conceitual desenvolvido para o sistema de banco de dados do pet shop é apresentado abaixo:



3.2 Regras de Negócio:

- **Clientes:** Cada cliente deve ter um registro único com informações de contato e histórico de serviços.
- **Animais:** Cada animal deve estar associado a um cliente, com informações sobre espécie, raça e histórico de atendimentos.
- **Produtos:** Controle de estoque detalhado, com informações sobre fornecedores e datas de validade.

- **Serviços:** Registro de serviços prestados, com datas, tipos de serviços e profissional responsável.

3.3 Dicionário de Dados.

3.3.1 Tabela Cliente

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_cliente	INT	-	Identificador único do cliente	Chave primária, IDENTITY(1,1)
nome	VARCHAR	100	Nome completo do cliente	Não nulo
endereço	VARCHAR	150	Endereço residencial do cliente	Não nulo
telefone	VARCHAR	15	Número de telefone do cliente	Não nulo
email	VARCHAR	100	E-mail do cliente	Não nulo

3.3.2 Tabela Animal

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_animal	INT	-	Identificador único do animal	Chave primária, IDENTITY(1,1)
nome	VARCHAR	50	Nome do animal	Não nulo
especie	VARCHAR	50	Espécie do animal	Não nulo
raca	VARCHAR	50	Raça do animal	Não nulo
idade	INT	-	Idade do animal	Nulo permitido
id_cliente	INT	-	Referência ao cliente responsável	Chave estrangeira

3.3.3 Tabela Produto

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_produto	INT	-	Identificador único do produto	Chave primária, IDENTITY(1,1)

nome	VARCHAR	100	Nome do produto	Não nulo
fornecedor	VARCHAR	100	Nome do fornecedor	Não nulo
quantidade_estoque	INT	-	Quantidade disponível no estoque	Não nulo
data_validade	INT	-	Data de validade do produto	Não nulo

3.3.4 Tabela Servico

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_servico	INT	-	Identificador único do serviço	Chave primária, IDENTITY(1,1)
tipo	VARCHAR	100	Tipo de serviço prestado	Não nulo
data	DATE	-	Data de realização do serviço	Não nulo
id_animal	INT	-	Referência ao animal atendido	Chave estrangeira
id_profissional	INT	-	Referência ao profissional responsável	Chave estrangeira

3.3.5 Tabela Produto_Servico

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_produto_servico	INT	-	Identificador único da relação	Chave primária, IDENTITY(1,1)
id_produto	INT	-	Referência ao produto utilizado	Chave estrangeira
id_servico	INT	-	Referência ao serviço relacionado	Chave estrangeira
quantidade	INT	-	Quantidade de produto utilizada	Não nulo

3.3.6 Tabela Profissional

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_profissional	INT	-	Identificador único do profissional	Chave primária, IDENTITY(1,1)
nome	VARCHAR	100	Nome do profissional	Não nulo

especialidade	VARCHAR	100	Especialidade do profissional	Não nulo
---------------	---------	-----	-------------------------------	----------

3.3.7 Tabela Historico_Servico

Nome da Coluna	Tipo de Dados	Tamanho	Descrição	Restrições
id_historico	INT	-	Identificador único do histórico	Chave primária, IDENTITY(1,1)
data_servico	DATE	-	Data do serviço realizado	Não nulo
id_cliente	INT	-	Referência ao cliente atendido	Chave estrangeira
id_servico	INT	-	Referência ao serviço realizado	Chave estrangeira

3.4 Modelo Físico

Abaixo, apresenta-se o código para a criação do banco de dados *PetShopDB* e das tabelas que compõem seu modelo físico. Cada tabela possui comentários que descrevem sua função no sistema e seus relacionamentos com outras tabelas.

```
CREATE DATABASE PetShopDB;
GO
USE PetShopDB;
GO
```

-- Tabela Cliente: Armazena informações dos clientes do PetShop.

```
CREATE TABLE Cliente (
    id_cliente INT IDENTITY(1,1) PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    endereco VARCHAR(150) NOT NULL,
    telefone VARCHAR(15) NOT NULL,
    email VARCHAR(100) NOT NULL
);
```

-- Tabela Animal: Registra os animais de estimação e associa cada um a um cliente.

```
CREATE TABLE Animal (
    id_animal INT IDENTITY(1,1) PRIMARY KEY,
    nome VARCHAR(50) NOT NULL,
    especie VARCHAR(50) NOT NULL,
    raca VARCHAR(50) NOT NULL,
    idade INT NULL,
    id_cliente INT NOT NULL,
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)
);
```

-- Tabela Produto: Gerencia os produtos disponíveis no PetShop.

```
CREATE TABLE Produto (
    id_produto INT IDENTITY(1,1) PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    fornecedor VARCHAR(100) NOT NULL,
    quantidade_estoque INT NOT NULL,
    data_validade DATE NOT NULL
);
```

```

);

-- Tabela Profissional: Registra os profissionais que realizam serviços no PetShop.
CREATE TABLE Profissional (
    id_profissional INT IDENTITY(1,1) PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    especialidade VARCHAR(100) NOT NULL
);

-- Tabela Servico: Registra os serviços realizados e associa um animal a um profissional.
CREATE TABLE Servico (
    id_servico INT IDENTITY(1,1) PRIMARY KEY,
    tipo VARCHAR(100) NOT NULL,
    data DATE NOT NULL,
    id_animal INT NOT NULL,
    id_profissional INT NOT NULL,
    FOREIGN KEY (id_animal) REFERENCES Animal(id_animal),
    FOREIGN KEY (id_profissional) REFERENCES Profissional(id_profissional)
);

-- Tabela Produto_Servico: Relaciona os produtos utilizados em serviços realizados.
CREATE TABLE Produto_Servico (
    id_produto_servico INT IDENTITY(1,1) PRIMARY KEY,
    id_produto INT NOT NULL,
    id_servico INT NOT NULL,
    quantidade INT NOT NULL,
    FOREIGN KEY (id_produto) REFERENCES Produto(id_produto),
    FOREIGN KEY (id_servico) REFERENCES Servico(id_servico)
);

-- Tabela Historico_Servico: Registra o histórico dos serviços realizados para cada cliente.
CREATE TABLE Historico_Servico (
    id_historico INT IDENTITY(1,1) PRIMARY KEY,
    data_servico DATE NOT NULL,
    id_cliente INT NOT NULL,
    id_servico INT NOT NULL,
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente),
    FOREIGN KEY (id_servico) REFERENCES Servico(id_servico)
);
GO

```

3.5 Dados Inseridos

O banco de dados foi populado com dados que foram inseridos com o intuito de simular um cenário realista de funcionamento do PetShop, abrangendo diferentes tipos de profissionais, serviços, produtos e o relacionamento entre clientes, animais e seus atendimentos. Essas inserções permitem verificar a consistência do modelo físico e validar as relações entre as tabelas.

3.5.1 Dados na Tabela Cliente

Os registros na tabela *Cliente* incluem informações básicas de cinco clientes fictícios.

```

INSERT INTO Cliente (nome, endereco, telefone, email) VALUES
('João Silva', 'Rua das Flores, 123', '11999999999', 'joao@gmail.com'),
('Maria Oliveira', 'Av. Paulista, 456', '11888888888', 'maria@gmail.com'),
('Carlos Santos', 'Rua Verde, 789', '11777777777', 'carlos@gmail.com'),
('Ana Costa', 'Av. Azul, 321', '11666666666', 'ana@gmail.com'),
('Fernanda Lima', 'Rua Rosa, 654', '11555555555', 'fernanda@gmail.com');

```

3.5.2 Dados na Tabela Animal

Os animais cadastrados estão associados aos seus respectivos proprietários.

```
INSERT INTO Animal (nome, especie, raca, idade, id_cliente) VALUES
('Rex', 'Cachorro', 'Labrador', 5, 1),
('Mimi', 'Gato', 'Persa', 3, 2),
('Thor', 'Cachorro', 'Husky Siberiano', 4, 3),
('Luna', 'Gato', 'Siamês', 2, 4),
('Bidu', 'Cachorro', 'Beagle', 6, 5);
```

3.5.3 Dados na Tabela Produto

Foram inseridos produtos comuns a um Petshop, cada um com quantidade em estoque e validade.

```
INSERT INTO Produto (nome, fornecedor, quantidade_estoque, data_validade) VALUES
('Ração para cães', 'Fornecedor A', 50, '2024-12-31'),
('Brinquedo para gatos', 'Fornecedor B', 30, '2025-01-01'),
('Shampoo para cães', 'Fornecedor C', 40, '2025-06-30'),
('Coleira antipulgas', 'Fornecedor D', 20, '2025-01-15'),
('Ração para gatos', 'Fornecedor A', 60, '2025-08-20');
```

3.5.4 Dados na Tabela Profissional

A tabela *Profissional* foi populada com registros de cinco profissionais, cada um com seu nome e especialidade. Estes profissionais desempenham funções específicas no Petshop, como veterinários e tosadores.

```
INSERT INTO Profissional (nome, especialidade) VALUES
('Dr. Carlos', 'Veterinário'),
('Ana Paula', 'Tosadora'),
('Mariana Souza', 'Veterinária'),
('Pedro Henrique', 'Tosador'),
('Cláudia Rocha', 'Veterinária');
```

3.5.5 Dados na Tabela Servico

Os registros na tabela *Servico* associam os serviços realizados aos animais atendidos e aos profissionais responsáveis. Cada registro descreve o tipo de serviço, a data em que foi realizado, o animal atendido e o profissional responsável.

```
INSERT INTO Servico (tipo, data, id_animal, id_profissional) VALUES
('Consulta veterinária', '2024-11-28', 1, 1),
('Banho e Tosa', '2024-11-29', 2, 2),
('Vacinação', '2024-12-01', 3, 1),
('Consulta veterinária', '2024-12-05', 4, 1),
('Banho e Tosa', '2024-12-10', 5, 2);
```

3.5.6 Dados na Tabela Produto_Servico

Os registros da tabela *Produto_Servico* indicam quais produtos foram utilizados em cada serviço e em qual quantidade. Esta tabela funciona como uma relação entre os produtos cadastrados e os serviços realizados.

```
INSERT INTO Produto_Servico (id_produto, id_servico, quantidade) VALUES
(1, 1, 1),
(2, 2, 2),
(3, 3, 1),
(4, 4, 1),
(5, 5, 1);
```


3.5.7 Dados na Tabela Historico_Servico

A tabela *Historico_Servico* registra o histórico dos serviços realizados para cada cliente, associando a data, o cliente e o serviço prestado.

```
INSERT INTO Historico_Servico (data_servico, id_cliente, id_servico) VALUES
('2024-11-28', 1, 1),
('2024-11-29', 2, 2),
('2024-12-01', 3, 3),
('2024-12-05', 4, 4),
('2024-12-10', 5, 5);
```

3.6 Consultas

1. Listar todos os clientes que possuem pelo menos um animal cadastrado.

Descrição: Exibe os clientes que possuem animais registrados na tabela Animal.

Consulta:

```
SELECT DISTINCT c.nome AS Nome_Cliente, c.telefone, c.email
FROM Cliente c
JOIN Animal a ON c.id_cliente = a.id_cliente;
```

Resultado:

	Nome_Cliente	telefone	email
1	Ana Costa	11666666666	ana@gmail.com
2	Carlos Santos	11777777777	carlos@gmail.com
3	Fernanda Lima	11555555555	femanda@gmail.com
4	João Silva	11999999999	joao@gmail.com
5	Maria Oliveira	11888888888	maria@gmail.com

2. Listar os profissionais que realizaram serviços nos últimos 30 dias.

Descrição: Filtra os profissionais com serviços executados recentemente.

Consulta:

```
SELECT DISTINCT p.nome AS Nome_Profissional, p.especialidade
FROM Profissional p
JOIN Servico s ON p.id_profissional = s.id_profissional
WHERE s.data >= DATEADD(DAY, -30, GETDATE());
```

Resultado:

	Nome_Profissional	especialidade
1	Ana Paula	Tosadora
2	Dr. Carlos	Veterinário

3. Quantidade de animais por cliente.

Descrição: Retorna a quantidade de animais cadastrados por cliente.

Consulta:

```
SELECT c.nome AS Nome_Cliente, COUNT(a.id_animal) AS Quantidade_Animalis
FROM Cliente c
LEFT JOIN Animal a ON c.id_cliente = a.id_cliente
GROUP BY c.nome
```

ORDER BY Quantidade_Animalis **DESC**;

Resultado:

	Nome_Cliente	Quantidade_Animalis
1	Ana Costa	1
2	Carlos Santos	1
3	Fernanda Lima	1
4	João Silva	1
5	Maria Oliveira	1

4. Lista de serviços realizados por espécie.

Descrição: Exibe a quantidade de serviços realizados para cada espécie de animal.

Consulta:

```
SELECT a.especie, COUNT(s.id_servico) AS Quantidade_Servicos
FROM Animal a
JOIN Servico s ON a.id_animal = s.id_animal
```

GROUP BY a.especie;

Resultado:

	especie	Quantidade_Servicos
1	Cachorro	3
2	Gato	2

5. Produtos mais utilizados em serviços.

Descrição: Mostra os produtos mais empregados em serviços, ordenados pela quantidade utilizada.

Consulta:

```
SELECT p.nome AS Nome_Produto, SUM(ps.quantidade) AS Quantidade_Utilizada
FROM Produto p
JOIN Produto_Servico ps ON p.id_produto = ps.id_produto
GROUP BY p.nome
ORDER BY Quantidade_Utilizada DESC;
```

Resultado:

	Nome_Produto	Quantidade_Utilizada
1	Brinquedo para gatos	2
2	Coleira antipulgas	1
3	Ração para cães	1
4	Ração para gatos	1
5	Shampoo para cães	1

6. Identificar os clientes que utilizaram mais serviços.

Descrição: Lista os clientes com maior histórico de serviços realizados.

Consulta:

```
SELECT c.nome AS Nome_Cliente, COUNT(hs.id_historico) AS Quantidade_Servicos
FROM Cliente c
JOIN Historico_Servico hs ON c.id_cliente = hs.id_cliente
GROUP BY c.nome
ORDER BY Quantidade_Servicos DESC;
```

Resultado:

	Nome_Cliente	Quantidade_Servicos
1	Ana Costa	1
2	Carlos Santos	1
3	Fernanda Lima	1
4	João Silva	1
5	Maria Oliveira	1

7. Serviços realizados em um intervalo de datas.

Descrição: Lista todos os serviços executados entre duas datas específicas.

Consulta:

```
SELECT s.tipo AS Tipo_Servico, s.data AS Data_Servico, a.nome AS Nome_Animal
FROM Servico s
JOIN Animal a ON s.id_animal = a.id_animal
WHERE s.data BETWEEN '2024-11-01' AND '2024-11-30';
```

Resultado:

	Tipo_Servico	Data_Servico	Nome_Animal
1	Consulta veterinária	2024-11-28	Rex
2	Banho e Tosa	2024-11-29	Mimi

8. Estoque de produtos com validade próxima.

Descrição: Exibe produtos cujo prazo de validade está se aproximando (nos próximos 60 dias).

Consulta:

```
SELECT nome, fornecedor, data_validade, quantidade_estoque
FROM Produto
WHERE data_validade <= DATEADD(DAY, 60, GETDATE());
```

Resultado:

	nome	fornecedor	data_validade	quantidade_estoque
1	Ração para cães	Fornecedor A	2024-12-31	50
2	Brinquedo para gatos	Fornecedor B	2025-01-01	30
3	Coleira antipulgas	Fornecedor D	2025-01-15	20

9. Profissionais que participaram de serviços na ultima semana.

Descrição: Listar os profissionais que participaram de serviços nos últimos 7 dias.

Consulta:

```
SELECT DISTINCT p.nome AS Nome_Profissional, p.especialidade
FROM Profissional p
JOIN Servico s ON p.id_profissional = s.id_profissional
WHERE s.data >= DATEADD(DAY, -7, GETDATE());
```

Resultado:

	Nome_Profissional	especialidade
1	Ana Paula	Tosadora
2	Dr. Carlos	Veterinário

10. Animais mais atendidos.

Descrição: Mostra os animais que passaram por mais serviços.

Consulta:

```
SELECT a.nome AS Nome_Animal, COUNT(s.id_servico) AS Quantidade_Atendimentos
```

FROM Animal a
 JOIN Servico s ON a.id_animal = s.id_animal
 GROUP BY a.nome
 ORDER BY Quantidade_Atendimentos DESC;

Resultado:

	Nome_Animal	Quantidade_Atendimentos
1	Bidu	1
2	Luna	1
3	Mimi	1
4	Rex	1
5	Thor	1

11. Quantidade de serviços por profissional.

Descrição: Retorna a quantidade de serviços realizados por cada profissional.

Consulta:

SELECT p.nome AS Nome_Profissional, COUNT(s.id_servico) AS Total_Servicos
 FROM Profissional p
 LEFT JOIN Servico s ON p.id_profissional = s.id_profissional
 GROUP BY p.nome
 ORDER BY Total_Servicos DESC;

Resultado:

	Nome_Profissional	Total_Servicos
1	Dr. Carlos	3
2	Ana Paula	2
3	Cláudia Rocha	0
4	Mariana Souza	0
5	Pedro Henrique	0

12. Clientes com serviços agendados para o próximo mês.

Descrição: Lista os clientes que possuem serviços marcados nos próximos 30 dias.

Consulta:

SELECT DISTINCT c.nome AS Nome_Cliente, c.telefone, s.data AS Data_Servico
 FROM Cliente c
 JOIN Animal a ON c.id_cliente = a.id_cliente
 JOIN Servico s ON a.id_animal = s.id_animal
 WHERE s.data BETWEEN GETDATE() AND DATEADD(DAY, 30, GETDATE());

Resultado:

	Nome_Cliente	telefone	Data_Servico
1	Ana Costa	11666666666	2024-12-05
2	Carlos Santos	11777777777	2024-12-01
3	Fernanda Lima	11555555555	2024-12-10

13. Detalhes de serviços de uma espécie específica.

Descrição: Mostra os serviços realizados para uma espécie específica, como "Cachorros".

Consulta:

SELECT s.tipo AS Tipo_Servico, s.data AS Data_Servico, a.nome AS Nome_Animal
 FROM Servico s
 JOIN Animal a ON s.id_animal = a.id_animal
 WHERE a.especie = 'Cachorro';

Resultado:

	Tipo_Servico	Data_Servico	Nome_Animal
1	Consulta veterinária	2024-11-28	Rex
2	Vacinação	2024-12-01	Thor
3	Banho e Tosa	2024-12-10	Bidu

14. Animais mais velhos cadastrados no sistema.

Descrição: Exibe os animais com as idades mais avançadas.

Consulta:

```
SELECT nome, especie, raca, idade
FROM Animal
ORDER BY idade DESC
```

Resultado:

	nome	especie	raca	idade
1	Bidu	Cachorro	Beagle	6
2	Rex	Cachorro	Labrador	5
3	Thor	Cachorro	Husky Siberiano	4
4	Mimi	Gato	Persa	3
5	Luna	Gato	Siamês	2

15. Estoque total agrupado por fornecedor.

Descrição: Calcula o total de produtos em estoque para cada fornecedor.

Consulta:

```
SELECT fornecedor, SUM(quantidade_estoque) AS Total_Em_Estoque
FROM Produto
GROUP BY fornecedor;
```

Resultado:

	fornecedor	Total_Em_Estoque
1	Fornecedor A	110
2	Fornecedor B	30
3	Fornecedor C	40
4	Fornecedor D	20

16. Relatório de serviços por mês.

Descrição: Mostra a quantidade de serviços realizados em cada mês do ano atual.

Consulta:

```
SELECT MONTH(s.data) AS Mes, COUNT(s.id_servico) AS Quantidade_Servicos
FROM Servico s
WHERE YEAR(s.data) = YEAR(GETDATE())
GROUP BY MONTH(s.data)
ORDER BY Mes;
```

Resultado:

	Mes	Quantidade_Servicos
1	11	2
2	12	3

17. Relacionar clientes e serviços realizados.

Descrição: Lista os serviços realizados por cada cliente.

Consulta:

```
SELECT c.nome AS Nome_Cliente, s.tipo AS Tipo_Servico, s.data AS Data_Servico
FROM Cliente c
```

JOIN Animal a ON c.id_cliente = a.id_cliente
 JOIN Servico s ON a.id_animal = s.id_animal;

Resultado:

	Nome_Cliente	Tipo_Servico	Data_Servico
1	João Silva	Consulta veterinária	2024-11-28
2	Maria Oliveira	Banho e Tosa	2024-11-29
3	Carlos Santos	Vacinação	2024-12-01
4	Ana Costa	Consulta veterinária	2024-12-05
5	Fernanda Lima	Banho e Tosa	2024-12-10

18. Produtos em estoque abaixo de um limite.

Descrição: Exibe produtos cujo estoque é menor que 25 unidades.

Consulta:

SELECT nome, quantidade_estoque, fornecedor
 FROM Produto
 WHERE quantidade_estoque < 25;

Resultado:

	nome	quantidade_estoque	fornecedor
1	Coleira antipulgas	20	Fornecedor D

19. Profissionais com maior número de atendimentos.

Descrição: Mostra os profissionais que mais realizaram serviços, ordenados pelo total.

Consulta:

SELECT p.nome AS Nome_Profissional, COUNT(s.id_servico) AS Total_Atendimentos
 FROM Profissional p
 JOIN Servico s ON p.id_profissional = s.id_profissional
 GROUP BY p.nome
 ORDER BY Total_Atendimentos DESC;

Resultado:

	Nome_Profissional	Total_Atendimentos
1	Dr. Carlos	3
2	Ana Paula	2

20. Gasto estimado por cliente com serviços e produtos.

Descrição: Calcula o total estimado que cada cliente gastou em serviços e produtos.

Consulta:

SELECT c.nome AS Nome_Cliente, SUM(ps.quantidade * 50) AS Gasto_Total
 FROM Cliente c
 JOIN Animal a ON c.id_cliente = a.id_cliente
 JOIN Servico s ON a.id_animal = s.id_animal
 JOIN Produto_Servico ps ON s.id_servico = ps.id_servico
 GROUP BY c.nome;

Resultado:

	Nome_Cliente	Gasto_Total
1	Ana Costa	50
2	Carlos Santos	50
3	Fernanda Lima	50
4	João Silva	50
5	Maria Oliveira	100

21. Serviços realizados em finais de semana.

Descrição: Lista os serviços realizados nos sábados e domingos.

Consulta:

```
SELECT s.tipo AS Tipo_Servico, s.data AS Data_Servico, p.nome AS Nome_Profissional
FROM Servico s
JOIN Profissional p ON s.id_profissional = p.id_profissional
WHERE DATEPART(WEEKDAY, s.data) IN (1, 7);
```

Resultado:

	Tipo_Servico	Data_Servico	Nome_Profissional
1	Vacinação	2024-12-01	Dr. Carlos

22. Clientes que usaram produtos específicos nos serviços.

Descrição: Identifica quais clientes usaram um produto específico, como "Ração para cães".

Consulta:

```
SELECT DISTINCT c.nome AS Nome_Cliente, p.nome AS Nome_Produto
FROM Cliente c
JOIN Animal a ON c.id_cliente = a.id_cliente
JOIN Servico s ON a.id_animal = s.id_animal
JOIN Produto_Servico ps ON s.id_servico = ps.id_servico
JOIN Produto p ON ps.id_produto = p.id_produto
WHERE p.nome = 'Ração para cães';
```

Resultado:

	Nome_Cliente	Nome_Produto
1	João Silva	Ração para cães

23. Dias mais movimentados com serviços.

Descrição: Descobre os dias com maior número de serviços realizados.

Consulta:

```
SELECT s.data AS Data_Servico, COUNT(s.id_servico) AS Total_Servicos
FROM Servico s
GROUP BY s.data
ORDER BY Total_Servicos DESC;
```

Resultado:

	Data_Servico	Total_Servicos
1	2024-11-28	1
2	2024-11-29	1
3	2024-12-01	1
4	2024-12-05	1
5	2024-12-10	1

24. Lista de produtos com estoque acima de um valor.

Descrição: Retorna produtos cujo estoque excede 30 unidades.

Consulta:

```
SELECT nome, fornecedor, quantidade_estoque
FROM Produto
WHERE quantidade_estoque > 30;
```

Resultado:

	nome	fornecedor	quantidade_estoque
1	Ração para cães	Fornecedor A	50
2	Shampoo para cães	Fornecedor C	40
3	Ração para gatos	Fornecedor A	60

25. Serviços realizados por cada profissional.

Descrição: Listar os serviços realizados por cada profissional e o número total de serviços executados.

Consulta:

```
SELECT p.nome AS Nome_Profissional, COUNT(s.id_servico) AS Total_Servicos
FROM Profissional p
JOIN Servico s ON p.id_profissional = s.id_profissional
GROUP BY p.nome
ORDER BY Total_Servicos DESC;
```

Resultado:

	Nome_Profissional	Total_Servicos
1	Dr. Carlos	3
2	Ana Paula	2

26. Serviços que utilizaram mais de um produto.

Descrição: Identifica os serviços que utilizaram vários produtos.

Consulta:

```
SELECT s.tipo AS Tipo_Servico, COUNT(ps.id_produto) AS Produtos_Utilizados
FROM Servico s
JOIN Produto_Servico ps ON s.id_servico = ps.id_servico
GROUP BY s.tipo
HAVING COUNT(ps.id_produto) > 1;
```

Resultado:

	Tipo_Servico	Produtos_Utilizados
1	Banho e Tosa	2
2	Consulta veterinária	2

27. Quantidade total de serviços por tipo.

Descrição: Calcula o total de serviços realizados para cada tipo.

Consulta:

```
SELECT s.tipo AS Tipo_Servico, COUNT(s.id_servico) AS Total_Realizados
FROM Servico s
GROUP BY s.tipo;
```

Resultado:

	Tipo_Servico	Total_Realizados
1	Banho e Tosa	2
2	Consulta veterinária	2
3	Vacinação	1

28. Produtos mais utilizados em serviços de "Vacinação".

Descrição: Lista os produtos utilizados especificamente em serviços de vacinação.

Consulta:

```
SELECT p.nome AS Nome_Produto, SUM(ps.quantidade) AS Quantidade_Utilizada
FROM Produto p
JOIN Produto_Servico ps ON p.id_produto = ps.id_produto
JOIN Servico s ON ps.id_servico = s.id_servico
WHERE s.tipo = 'Vacinação'
GROUP BY p.nome;
```

Resultado:

	Nome_Produto	Quantidade_Utilizada
1	Shampoo para cães	1

29. Serviços utilizados pelos clientes.

Descrição: Listar os clientes e os tipos de serviço que eles utilizaram (sem duplicar o tipo por cliente).

Consulta:

```
SELECT DISTINCT c.nome AS Nome_Cliente, s.tipo AS Tipo_Servico
FROM Cliente c
JOIN Animal a ON c.id_cliente = a.id_cliente
JOIN Servico s ON a.id_animal = s.id_animal
ORDER BY c.nome, s.tipo;
```

Resultado:

	Nome_Cliente	Tipo_Servico
1	Ana Costa	Consulta veterinária
2	Carlos Santos	Vacinação
3	Fernanda Lima	Banho e Tosa
4	João Silva	Consulta veterinária
5	Maria Oliveira	Banho e Tosa

30. Serviços realizados em intervalo de data.

Descrição: Identificar os serviços realizados em um intervalo específico de datas, agrupados por tipo de serviço.

Consulta:

```
SELECT s.tipo AS Tipo_Servico, COUNT(s.id_servico) AS Total_Servicos
FROM Servico s
WHERE s.data BETWEEN '2024-11-01' AND '2024-12-31'
GROUP BY s.tipo
ORDER BY Total_Servicos DESC;
```

Resultado:

	Tipo_Servico	Total_Servicos
1	Banho e Tosa	2
2	Consulta veterinária	2
3	Vacinação	1

4 CONCLUSÃO

4.1 Conclusões finais

O desenvolvimento deste projeto de sistema para o gerenciamento de um pet shop, focado na criação de um banco de dados relacional, foi concluído com sucesso. O projeto cumpriu sua missão de propor e modelar um sistema capaz de gerenciar eficientemente informações essenciais, desde o registro de clientes até o controle de estoque e a prestação de serviços. Com a transição de um modelo conceitual para um modelo físico, o sistema agora está apto a ser implementado em um ambiente real, utilizando tecnologias de banco de dados como o Microsoft SQL Server 2022.

A análise detalhada dos requisitos e a modelagem dos dados proporcionaram as bases para a criação de um sistema robusto que pode melhorar a eficiência operacional e a experiência do cliente no pet shop. Além disso, a implementação do modelo físico garante que o banco de dados será funcional e pronto para lidar com grandes volumes de dados, proporcionando alta disponibilidade e integração com outras tecnologias. Este projeto não apenas demonstrou a viabilidade de um sistema de banco de dados relacional para gerenciar as operações de um pet shop, mas também apresentou um modelo prático que pode ser utilizado diretamente na operação de um pet shop real.

4.2 Sugestões de melhoria

Apesar dos avanços alcançados, o projeto ainda possui potencial para evoluções futuras. Algumas sugestões para possíveis melhorias são:

- **Integração com Sistemas de Pagamento e Faturamento:** Implementar uma funcionalidade de processamento de pagamentos para que o pet shop possa registrar automaticamente transações financeiras, como o pagamento de serviços ou produtos adquiridos. A integração com gateways de pagamento ou sistemas de faturamento permitiria que o sistema fosse ainda mais completo,

automatizando processos financeiros e evitando erros manuais.

- **Aprimoramento da Funcionalidade de Cadastro de Animais:** Ampliar o cadastro de animais para incluir mais informações detalhadas, como histórico de saúde, alergias, histórico de serviços realizados e outras observações específicas. Isso permitiria um atendimento mais personalizado e seguro, principalmente em casos de necessidades especiais dos animais.
- **Adoção de Funcionalidade de Agenda para Profissionais:** Integrar um sistema de agendamento para os profissionais (veterinários, tosadores, etc.), permitindo que os clientes agendem serviços diretamente no sistema. Isso pode incluir notificações automáticas para os clientes e profissionais, além de melhorar o planejamento das atividades no pet shop.

4.3 Considerações finais

Este projeto atingiu seu objetivo de desenvolver um sistema de gerenciamento de banco de dados relacional para um pet shop, com a transição bem-sucedida de um modelo teórico para um modelo físico. A partir deste ponto, o sistema se encontra em um estágio que pode ser aplicado em um cenário real, proporcionando melhorias significativas na gestão e no atendimento ao cliente. As diretrizes e princípios definidos neste trabalho, agora com um modelo físico pronto para ser implementado, apresentam um modelo prático de como sistemas de bancos de dados relacionais podem ser utilizados no gerenciamento de pet shops, com potencial para otimizar operações e garantir a eficiência do serviço prestado.

REFERÊNCIAS

Exemplo de diagrama ER de banco de dados (pé de galinha). Lucidchart.

Disponível em: <https://www.lucidchart.com/pages/pt/modelos/exemplo-de-diagrama-er-de-banco-de-dados-pe-de-galinha>. Acesso em: 30 nov.2024.

Símbolos e notação de diagramas entidade-relacionamento. Lucidchart.

Disponível em: <https://www.lucidchart.com/pages/pt/simbolos-de-diagramas-entidade-relacionamento>. Acesso em: 30 nov. 2024.

Tutorial de criação e estruturação de banco de dados. Lucidchart. Disponível em:

<https://www.lucidchart.com/pages/pt/tutorial-de-criacao-e-estruturacao-de-banco-de-dados>. Acesso em: 30 nov. 2024.

Hardware and software requirements for installing SQL Server 2022.

MICROSOFT. Disponível em:

<https://learn.microsoft.com/pt-br/sql/sql-server/install/hardware-and-software-requirements-for-installing-sql-server-2022?view=sql-server-ver16>. Acesso em: 30 nov. 2024.