

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

**Leandro Torres Mocelin**

**SISTEMA INTEGRADO PARA GERENCIAMENTO DE ANI-  
MAIS DE ESTIMAÇÃO**

**CAMPOS DO JORDÃO**  
**2024**

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

**Leandro Torres Mocelin**

Entrega parcial de trabalho final apresentado  
ao Instituto Federal de São Paulo (IFSP-  
CJO), em cumprimento a exigência da disci-  
plina de Banco de Dados 2, do curso de Anali-  
se e Desenvolvimento de Sistema (ADS).

**Me. Paulo Giovani de Faria Zeferine**

**CAMPOS DO JORDÃO**

**2024**

## RESUMO

Este projeto visa criar um Sistema Integrado para Gerenciamento de Animais de Estimação, com o propósito de simplificar a vida dos proprietários de pets na organização de informações cruciais sobre seus companheiros. O objetivo principal é fornecer uma solução abrangente, englobando o histórico médico, a localização de serviços veterinários e pet shops, proporcionando uma experiência mais eficiente e conectada para os amantes de animais. Na busca por atender às crescentes demandas dos cuidadores de animais, a metodologia adotada inicia-se com uma análise aprofundada das necessidades e expectativas dos usuários. A ferramenta escolhida para a modelagem é o SQL, utilizando a notação UML para desenvolver um modelo conceitual robusto. Os requisitos do sistema serão minuciosamente detalhados, e a coleta de regras de negócio será realizada por meio de entrevistas com veterinários e proprietários de animais de estimação. Os resultados esperados abrangem a criação de um sistema eficiente que ofereça aos usuários uma maneira organizada e acessível de gerenciar o histórico médico de seus animais de estimação. Além disso, a integração de funcionalidades para localização de serviços veterinários e pet shops proporcionará maior comodidade e praticidade. Concluindo, a proposta deste Sistema Integrado busca preencher uma lacuna na gestão de informações relacionadas aos animais de estimação, oferecendo uma solução completa e inovadora. Ao consolidar o histórico médico e facilitar o acesso a serviços especializados, espera-se contribuir significativamente para a experiência dos cuidadores, promovendo uma abordagem mais eficaz e conectada ao cuidado com seus companheiros de quatro patas.

**Palavras-Chave:** Sistema; Gerenciamento; Animais de estimação; SQL; UML; Pet shops; Veterinários.

## ABSTRACT

This project aims to create an Integrated System for Pet Management, with the purpose of simplifying the lives of pet owners in organizing crucial information about their companions. The primary objective is to provide a comprehensive solution, encompassing medical history, the location of veterinary services, and pet shops, offering a more efficient and connected experience for animal lovers. To meet the growing demands of pet caregivers, the adopted methodology begins with a thorough analysis of user needs and expectations.

SQL is chosen as the modeling tool, utilizing UML notation to develop a robust conceptual model. The system requirements will be meticulously detailed, and business rules will be collected through interviews with veterinarians and pet owners. The expected results include the creation of an efficient system that offers users an organized and accessible way to manage the medical history of their pets. Additionally, the integration of features for locating veterinary services and pet shops will provide greater convenience and practicality.

In conclusion, the proposal of this Integrated System seeks to fill a gap in the management of information related to pets, offering a comprehensive and innovative solution. By consolidating medical history and facilitating access to specialized services, the aim is to contribute significantly to the caregivers' experience, promoting a more effective and connected approach to the care of their four-legged companions.

**Keywords:** System; Management; Pets; SQL; UML; Pet shops; Veterinarians.

## LISTA DE ILUSTRAÇÕES

<b>FIGURA 1 – Modelo Conceitual</b>	17
<b>FIGURA 2 – Modelo Lógico</b>	23
<b>FIGURA 3 – Resultado 1</b>	29
<b>FIGURA 4 – Resultado 2</b>	30
<b>FIGURA 5 – Resultado 3</b>	30
<b>FIGURA 6 – Resultado 4</b>	31
<b>FIGURA 7 – Resultado 5</b>	31
<b>FIGURA 8 – Resultado 6</b>	32
<b>FIGURA 9 – Resultado 7</b>	32
<b>FIGURA 10 – Resultado 8</b>	33
<b>FIGURA 11 – Resultado 9</b>	33
<b>FIGURA 12 – Resultado 10</b>	34
<b>FIGURA 13 – Resultado 11</b>	34
<b>FIGURA 14 – Resultado 12</b>	35
<b>FIGURA 15 – Resultado 13</b>	35
<b>FIGURA 16 – Resultado 14</b>	36
<b>FIGURA 17 – Resultado 15</b>	36
<b>FIGURA 18 – Resultado 16</b>	37
<b>FIGURA 19 – Resultado 17</b>	37
<b>FIGURA 20 – Resultado 18</b>	38
<b>FIGURA 21 – Resultado 19</b>	38
<b>FIGURA 22 – Resultado 20</b>	39

<b>FIGURA 23 – Resultado 21</b>	39
<b>FIGURA 24 – Resultado 22</b>	40
<b>FIGURA 25 – Resultado 23</b>	40
<b>FIGURA 26 – Resultado 24</b>	41
<b>FIGURA 27 – Resultado 25</b>	41
<b>FIGURA 28 – Resultado 26</b>	42
<b>FIGURA 29 – Resultado 27</b>	42
<b>FIGURA 30 – Resultado 28</b>	43
<b>FIGURA 31 – Resultado 29</b>	43
<b>FIGURA 32 – Resultado 30</b>	44

## **LISTA DE QUADROS**

**QUADRO 1 –** Dicionário de Dados

18

## LISTA DE ALGORITMOS

<b>ALGORITMO 1 – Modelo Físico em SQL</b>	24
<b>ALGORITMO 2 – Consulta 1</b>	29
<b>ALGORITMO 3 – Consulta 2</b>	30
<b>ALGORITMO 4 – Consulta 3</b>	30
<b>ALGORITMO 5 – Consulta 4</b>	30
<b>ALGORITMO 6 – Consulta 5</b>	31
<b>ALGORITMO 7 – Consulta 6</b>	32
<b>ALGORITMO 8 – Consulta 7</b>	32
<b>ALGORITMO 9 – Consulta 8</b>	32
<b>ALGORITMO 10 – Consulta 9</b>	32
<b>ALGORITMO 11 – Consulta 10</b>	33
<b>ALGORITMO 12 – Consulta 11</b>	33
<b>ALGORITMO 13 – Consulta 12</b>	34
<b>ALGORITMO 14 – Consulta 13</b>	34
<b>ALGORITMO 15 – Consulta 14</b>	35
<b>ALGORITMO 16 – Consulta 15</b>	35
<b>ALGORITMO 17 – Consulta 16</b>	36
<b>ALGORITMO 18 – Consulta 17</b>	36
<b>ALGORITMO 19 – Consulta 18</b>	37
<b>ALGORITMO 20 – Consulta 19</b>	37
<b>ALGORITMO 21 – Consulta 20</b>	38
<b>ALGORITMO 22 – Consulta 21</b>	38



<b>ALGORITMO 23 – Consulta 22</b>	39
<b>ALGORITMO 24 – Consulta 23</b>	39
<b>ALGORITMO 25 – Consulta 24</b>	40
<b>ALGORITMO 26 – Consulta 25</b>	40
<b>ALGORITMO 27 – Consulta 26</b>	41
<b>ALGORITMO 28 – Consulta 27</b>	41
<b>ALGORITMO 29 – Consulta 28</b>	42
<b>ALGORITMO 30 – Consulta 29</b>	42
<b>ALGORITMO 31 – Consulta 30</b>	43

## LISTA DE SIGLAS

<b>IFSP</b>	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo
<b>CJO</b>	Campos do Jordão
<b>UML</b>	Unified Modeling Language
<b>SQL</b>	Structured Query Language
<b>ADS</b>	Análise e Desenvolvimento de Sistema
<b>IBGE</b>	Instituto Brasileiro de Geografia e Estatística

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
<b>1.1</b>	<b>Objetivos</b>	<b>12</b>
<b>1.2</b>	<b>Justificativa</b>	<b>12</b>
<b>1.3</b>	<b>Aspectos Metodológicos</b>	<b>13</b>
<b>1.4</b>	<b>Aporte Teórico</b>	<b>13</b>
<b>2</b>	<b>METODOLOGIA</b>	<b>13</b>
<b>2.1</b>	<b>Considerações Iniciais</b>	<b>14</b>
<b>2.2</b>	<b>Requisitos</b>	<b>14</b>
<b>2.3</b>	<b>Ferramentas de Modelagem</b>	<b>15</b>
<b>2.4</b>	<b>Descrição do Projeto de Dados</b>	<b>15</b>
<b>2.5</b>	<b>Regras de Negócio</b>	<b>15</b>
<b>3</b>	<b>RESULTADOS OBTIDOS</b>	<b>16</b>
<b>3.1</b>	<b>Avaliação e Entrevistas</b>	<b>16</b>
<b>3.2</b>	<b>Modelo Conceitual</b>	<b>17</b>
<b>3.2.1</b>	<b>DICIONÁRIO DE DADOS</b>	<b>18</b>
<b>3.3</b>	<b>Modelo Lógico</b>	<b>23</b>
<b>3.4</b>	<b>Modelo Físico</b>	<b>23</b>
<b>3.5</b>	<b>Inserção dos Dados</b>	<b>28</b>
<b>3.5.1</b>	<b>EXEMPLOS DE CONSULTAS</b>	<b>29</b>
<b>4</b>	<b>CONCLUSÃO</b>	<b>41</b>
	<b>REFERÊNCIAS</b>	<b>42</b>
	<b>GLOSSÁRIO</b>	<b>43</b>

# **1 INTRODUÇÃO**

O número de animais de estimação no Brasil vem crescendo nos últimos anos. Segundo dados do Instituto Brasileiro de Geografia e Estatística (IBGE), em 2021, havia cerca de 142 milhões de animais de estimação no Brasil, sendo 54,2 milhões de cães e 39,3 milhões de gatos.

Com o aumento do número de animais de estimação, cresce também a demanda por serviços e produtos para pets. Nesse contexto, o desenvolvimento de sistemas para o gerenciamento de animais de estimação é uma oportunidade para atender a essa demanda.

O sistema proposto neste projeto atenderá às seguintes necessidades dos donos de animais de estimação:

Gerenciamento de informações pessoais do pet, incluindo nome, raça, idade, sexo, etc.

Gerenciamento do histórico médico do pet, incluindo vacinas, consultas, exames e medicamentos.

Localização de pet shops e veterinários próximos.

## **1.1 Objetivos**

O principal objetivo deste projeto é criar um sistema abrangente que ofereça uma gestão eficaz do histórico médico de animais de estimação, incluindo funcionalidades para localização de serviços veterinários e pet shops. Almejamos fornecer aos usuários uma ferramenta organizada e acessível, contribuindo para a otimização do cuidado e aprimoramento da experiência dos cuidadores de animais.

## **1.2 Justificativa**

Com o crescente número de animais de estimação no Brasil, há uma lacuna na gestão de informações relacionadas a esses companheiros. Este sistema busca preencher essa lacuna, oferecendo uma solução inovadora e completa para atender

às demandas dos cuidadores. Ao consolidar o histórico médico e facilitar o acesso a serviços especializados, a justificativa reside na contribuição significativa para uma abordagem mais eficaz e conectada ao cuidado com os animais de estimação.

### **1.3 Aspectos Metodológicos**

A metodologia adotada inicia-se com uma análise aprofundada das necessidades e expectativas dos usuários, utilizando entrevistas com veterinários e proprietários de animais de estimação. A ferramenta escolhida para a modelagem é o SQL, e a notação UML será empregada no desenvolvimento de um modelo conceitual robusto. A eficiência na busca e recuperação de informações será considerada no projeto de dados, assegurando a qualidade e a relevância do sistema proposto.

### **1.4 Aporte Teórico**

O aporte teórico deste projeto baseia-se em conceitos fundamentais de Banco de Dados, utilizando SQL para modelagem e manipulação eficaz de dados. Além disso, a notação UML será aplicada para desenvolver um modelo conceitual sólido. Referências de autores especializados em sistemas de informação e banco de dados, como Carlos Alberto Heuser e Peter Rob, serão exploradas para embasar teoricamente o desenvolvimento do Sistema Integrado para Gerenciamento de Animais de Estimação.

## **2 METODOLOGIA**

A metodologia adotada inicia com a análise das necessidades e expectativas dos usuários, orientando o desenvolvimento do sistema. A ferramenta escolhida para a modelagem é o SQL, utilizando a notação UML para a elaboração do modelo conceitual. Os requisitos do sistema serão detalhados, e a coleta de regras de negócio será realizada por meio de entrevistas com veterinários e donos de animais de estimação.

O projeto de dados será elaborado considerando a eficiência na busca e recuperação de informações.

## **2.1 Considerações Iniciais**

A condução deste trabalho iniciou-se com uma análise aprofundada das necessidades e expectativas dos usuários, estabelecendo considerações iniciais que fundamentaram o desenvolvimento do projeto. Nessa fase, foram identificadas lacunas na gestão de informações relacionadas a animais de estimação, proporcionando insights cruciais para a definição de objetivos claros e alinhados às demandas reais dos cuidadores de pets.

## **2.2 Requisitos**

A próxima etapa crucial no desenvolvimento do Sistema Integrado envolve a definição precisa dos requisitos. Esses requisitos podem ser categorizados em duas vertentes essenciais:

### **Requisitos Funcionais:**

- Registro e organização do histórico médico dos animais.
- Funcionalidades de busca e localização de serviços veterinários e pet shops.

### **Requisitos Não Funcionais:**

- Desempenho ágil e responsivo do sistema.
- Garantia de segurança e proteção dos dados sensíveis dos animais e proprietários.
- Interface de usuário intuitiva para facilitar a usabilidade.

Ao detalhar esses requisitos, o objetivo é estabelecer critérios claros que orientarão o desenvolvimento do sistema, garantindo que atenda efetivamente às necessidades dos usuários e ofereça uma solução integrada e eficaz para o gerenciamento de animais de estimação. Essa abordagem visa não apenas facilitar a vida dos proprietários de pets, mas também promover uma experiência mais completa e conectada no

cuidado com seus companheiros de quatro patas.

### **2.3 Ferramentas de Modelagem**

Para a etapa de modelagem, foi selecionada a ferramenta SQL (Structured Query Language), uma linguagem universal para manipulação de bancos de dados. A escolha do SQL possibilitou a criação de um modelo de dados robusto, capaz de armazenar e recuperar informações de maneira eficiente. Além disso, a notação UML (Unified Modeling Language) foi empregada para a elaboração do modelo conceitual, oferecendo uma representação visual clara e compreensível da arquitetura do sistema.

### **2.4 Descrição do Projeto de Dados**

O projeto de dados foi meticulosamente elaborado, considerando a eficiência na busca e recuperação de informações. Tabelas foram definidas para armazenar dados essenciais, como histórico médico, informações pessoais dos animais de estimação, dados de veterinários e petshops. A definição cuidadosa das relações entre as tabelas garantiu a integridade e a coesão do banco de dados, possibilitando consultas eficazes.

### **2.5 Coleta de Regras de Negócio**

A coleta das regras de negócio foi realizada por meio de entrevistas com veterinários e proprietários de animais de estimação. Essa abordagem prática permitiu a identificação de requisitos específicos do domínio, fundamentais para o desenvolvimento de funcionalidades que atendessem às reais necessidades dos usuários. O diálogo direto com os proprietários de animais contribuiu significativamente para a definição precisa de requisitos e o refinamento contínuo do sistema.

### 3 RESULTADOS OBTIDOS

Todo início de (sub)seção deve ter um texto antes de iniciar suas (sub)subseções. Por exemplo, “Nesta seção serão apresentados os resultados deste trabalho e uma discussão sobre eles.”.

#### 3.1 Avaliação e Entrevistas

A fase de avaliação do Sistema Integrado para Gerenciamento de Animais de Estimação foi enriquecida pela participação ativa dos usuários por meio de entrevistas.

##### **Entrevista com Proprietários de Animais de Estimação:**

**Pergunta 1:** Como é o seu atual processo de organização das informações relacionadas ao seu animal de estimação?

**Respostas:** Os proprietários destacaram desafios na gestão de dados dispersos, como histórico médico e informações sobre serviços veterinários e pet shops.

**Pergunta 2:** Que funcionalidades você considera cruciais para um sistema que visa auxiliar no cuidado com animais de estimação?

**Respostas:** Os proprietários falaram de armazenamento seguro do histórico médico, lembretes para vacinações e consultas, além da facilidade em localizar serviços veterinários e pet shops próximos.

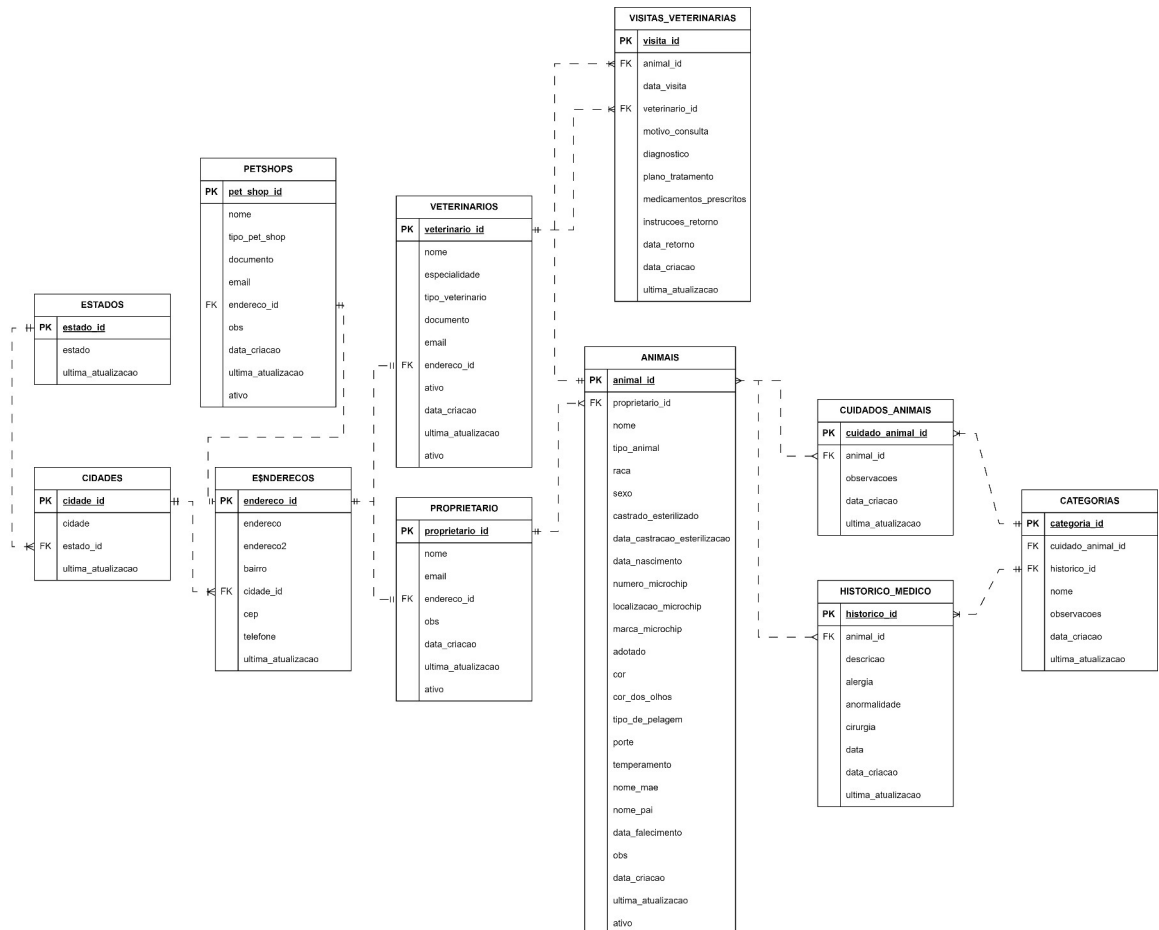
**Pergunta 3:** Como você percebe a importância da comunicação entre clínicas veterinárias e pet shops para o bem-estar do seu animal de estimação?

**Respostas:** Os proprietários destacaram a relevância da comunicação entre prestadores de serviços para garantir um cuidado mais integral, abordando aspectos de saúde, alimentação e outros serviços essenciais para seus animais de estimação.



### 3.2 Modelo Conceitual

O Modelo Conceitual do sistema gerenciamento de animais de estimação foi feito no software *DRAWN.io* com a notação ER (Entidade-Relacional).



**Figura 1** – Modelo Conceitual (Acervo pessoal)

### 3.2.1 Dicionário de Dados

TABELA	ATRIBUTO	CONTEÚDO	NECESSÁRIO	PK / FK	TABELA REFERENCIADA
<u>ESTADOS</u>	estado_id	Identificador do Estado	Sim	PK	
	estado	Nome do Estado	Sim		
	ultima_atualizacao	Última atualização	Não		
<u>CIDADES</u>	cidade_id	Identificador da Cidade	Sim	PK	
	estado_id	Identificador do Estado	Sim	FK	ESTADOS
	ultima_atualizacao	Última atualização	Não		
<u>ENDERECOS</u>	endereco_id	Identificador do Endereço	Sim	PK	
	endereco	Endereço	Sim		
	endereco2	Complemento do Endereço	Não		
	bairro	Bairro	Não		
	cidade_id	Identificador da Cidade	Sim	FK	CIDADES
	cep	CEP	Sim		
	telefone	Telefone	Sim		
	ultima_atualizacao	Última atualização	Não		
<u>PROPRIETARIOS</u>	proprietario_id	Identificador do Proprietário	Sim	PK	
	nome	Nome do Proprietário	Sim		
	email	Email do Proprietário	Sim		
	endereco_id	Identificador do Endereço	Sim	FK	ENDERECOS
	obs	Observações	Não		
	data_criacao	Data de Cria-	Não		

		ção			
	ultima_atualiza- cao	Última atuali- zação	Não		
	ativo	Ativo	Sim		
<u>VETERINARI- OS</u>	veterinario_id	Identificador do Veteriná- rio	Sim	PK	
	nome	Nome do Ve- terinário	Sim		
	especialidade	Especialida- de	Não		
	tipo_veterinario	Tipo do Vete- rinário (pj, pf)	Sim		
	documento	Documento (CPF, CNPJ)	Sim		
	email	Email do Ve- terinário	Não		
	observacoes	Observações	Não		
	data_criacao	Data de Cria- ção	Não		
	ultima_atualiza- cao	Última atuali- zação	Não		
	ativo	Ativo	Sim		
	endereco_id	Identificador do Endereço	Sim	FK	ENDERECOS
<u>PETSHOPS</u>	pet_shop_id	Identificador do Pet Shop	Sim	PK	
	nome	Nome do Pet Shop	Sim		
	tipo_pet_shop	Tipo do Pet Shop (pj, pf)	Sim		
	documento	Documento (CPF, CNPJ)	Sim		
	email	Email do Pet Shop	Não		
	observacoes	Observações	Não		
	data_criacao	Data de Cria- ção	Não		
	ultima_atualiza- cao	Última atuali- zação	Não		

	ativo	Ativo	Sim		
	endereco_id	Identificador do Endereço	Sim	FK	ENDERECOS
<u>ANIMAIS</u>	animal_id	Identificador do Animal	Sim	PK	
	proprietario_id	Identificador do Proprietário	Sim	FK	PROPRIETARIOS
	nome	Nome do Animal	Sim		
	tipo_animal	Tipo do Animal (cachorro, gato)	Sim		
	raca	Raça do Animal	Não		
	sexo	Sexo do Animal (macho, fêmea)	Não		
	castrado_esterilizado	Castrado/Esterilizado (sim, não)	Não		
	data_castracao_esterilizacao	Data de Castração/Esterilização	Não		
	data_nascimento	Data de Nascimento	Não		
	numero_microchip	Número do Microchip	Não		
	localizacao_microchip	Localização do Microchip	Não		
	marca_microchip	Marca do Microchip	Não		
	adotado	Adotado (sim, não)	Não		
	cor	Cor do Animal	Não		
	cor_dos_olhos	Cor dos Olhos do Animal	Não		
	tipo_de_pelagem	Tipo de Pelagem	Não		
	porte	Porte do Ani-	Não		

		mal			
	temperamento	Temperamento do Animal	Não		
	nome_mae	Nome da Mãe do Animal	Não		
	nome_pai	Nome do Pai do Animal	NãoNão		
	data_falecimento	Data de Falecimento	Não		
	observacoes	Observações	Não		
	data_criacao	Data de Criação	Não		
	ultima_atualizacao	Última atualização	Não		
	ativo	Ativo	Sim		
<u>CUIDADOS_A NIMAIS</u>	cuidado_animal_id	Identificador do Cuidado Animal	Sim	PK	
	animal_id	Identificador do Animal	Sim	FK	ANIMAIS
	observacoes	Observações	Não		
	data_criacao	Data de Criação	Não		
	ultima_atualizacao	Última atualização	Não		
<u>CATEGORIAS</u>	categoria_id	Identificador da Categoria	Sim	PK	
	cuidado_animal_id	Identificador do Cuidado Animal	Sim	FK	CUIDADOS_ANIMAIS
	historico_id	Identificador do Histórico Médico	Sim	FK	HISTORICO_MEDICO
	nome	Nome da Categoria	Sim		
	observacoes	Observações	Não		
	data_criacao	Data de Criação	Não		
	ultima_atualizacao	Última atualização	Não		

	cao	zação			
<u>HISTORICO MEDICO</u>	historico_id	Identificador do Histórico Médico	Sim	PK	
	animal_id	Identificador do Animal	Sim	FK	ANIMAIS
	descricao	Descrição do Histórico	Não		
	alergia	Alergia	Não		
	anormalidade	Anormalidade	Não		
	cirurgia	Cirurgia	Não		
	data	Data da Cirurgia	Não		
	data_criacao	Data de Criação	Não		
	ultima_atualizacao	Última atualização	Não		
<u>VISITAS VETERINARIAS</u>	visita_id	Identificador da Visita Veterinária	Sim	PK	
	animal_id	Identificador do Animal	Sim	FK	ANIMAIS
	data_visita	Data da Visita	Sim		
	veterinario_id	Identificador do Veterinário	Sim	FK	VETERINARIOS
	motivo_consulta	Motivo da Consulta	Sim		
	diagnostico	Diagnóstico	Não		
	plano_tratamento	Plano de Tratamento	Não		
	medicamentos_prescritos	Medicamentos Prescritos	Não		
	instrucoes_retorno	Instruções para Retorno	Não		
	data_retorno	Data de Retorno	Sim		

**Quadro 1** – Dicionário de Dados

### 3.3 Modelo Lógico

O Modelo Lógico do sistema de farmácia foi feito pelo software de gerenciamento de banco de dados MySQL Workbench 8.0 CE.

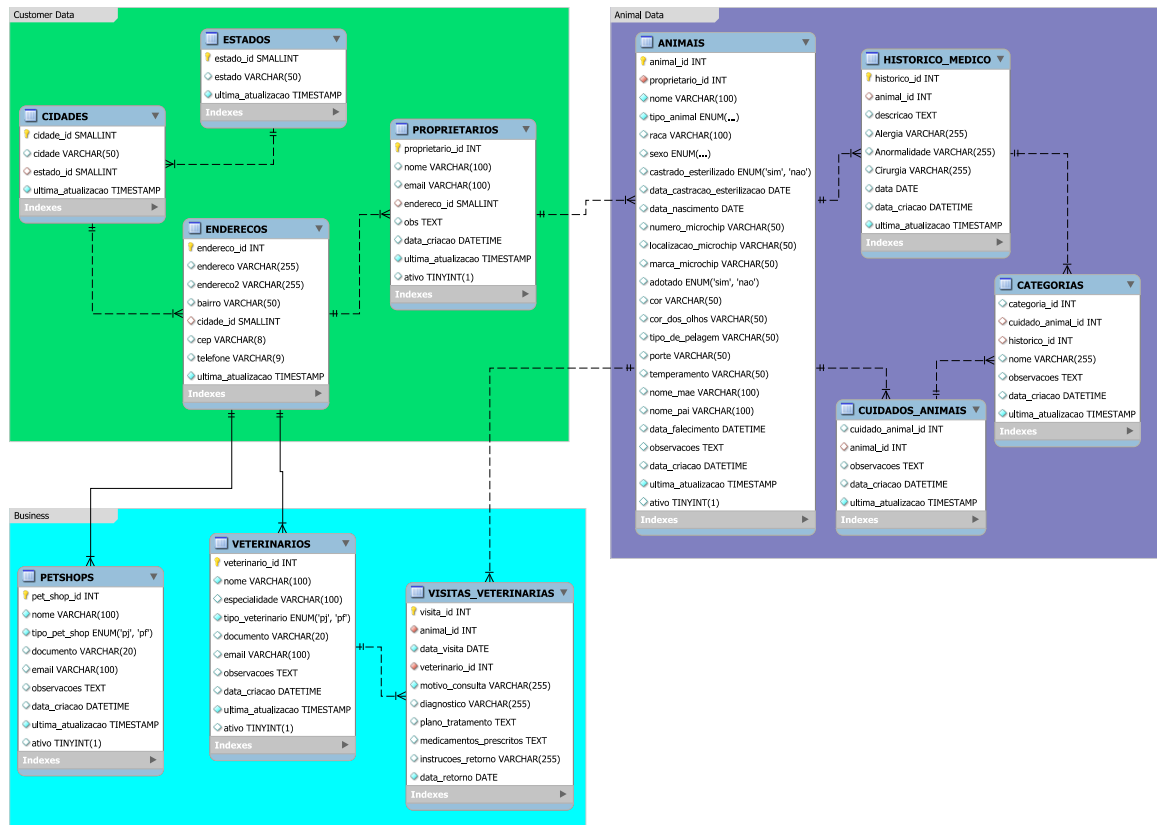


Figura 2 – Modelo Lógico (Acervo pessoal)

### 3.4 Modelo Físico

O Modelo Físico foi criado no software de gerenciamento de banco de dados MySQL Workbench 8.0 CE usando como base o Modelo Lógico, e modificado em outro software de desenvolvimento Visual Studio Code. O Modelo Físico foi feito na linguagem SQL (Structured Query Language, em português Linguagem de Consulta Estruturada). A seguir, o código em SQL para a criação do banco de dados do sistema gerenciamento de animais de estimação.

```
-- Criação do Banco de Dados
CREATE DATABASE IF NOT EXISTS PetManagementSystem;
USE PetManagementSystem;
```

```
-- Tabela Estados
CREATE TABLE IF NOT EXISTS ESTADOS (
    estado_id SMALLINT AUTO_INCREMENT PRIMARY KEY,
    estado VARCHAR(50),
    ultima_atualizacao TIMESTAMP
);
```

```
-- Tabela Cidades
CREATE TABLE IF NOT EXISTS CIDADES (
    cidade_id SMALLINT AUTO_INCREMENT PRIMARY KEY,
    cidade VARCHAR(50),
    estado_id SMALLINT,
    ultima_atualizacao TIMESTAMP,
    FOREIGN KEY (estado_id)
    REFERENCES ESTADOS(estado_id)
);
```

```
-- Tabela Enderecos
CREATE TABLE IF NOT EXISTS ENDERECOS (
    endereco_id INT AUTO_INCREMENT PRIMARY KEY,
    endereco VARCHAR(255),
    endereco2 VARCHAR(255),
    bairro VARCHAR(50),
    cidade_id SMALLINT,
    cep VARCHAR(8),
    telefone VARCHAR(9),
    ultima_atualizacao TIMESTAMP,
    FOREIGN KEY (cidade_id)
    REFERENCES CIDADES(cidade_id)
);
```

```
-- Tabela Proprietários
CREATE TABLE IF NOT EXISTS PROPRIETARIOS (
    proprietario_id INT AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(100),
    email VARCHAR(100),
    endereco_id SMALLINT,
    obs TEXT,
    data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,
    ultima_atualizacao TIMESTAMP,
    ativo TINYINT(1) DEFAULT 1,
    FOREIGN KEY (endereco_id)
    REFERENCES ENDERECOS(endereco_id)
);
```



-- Tabela Veterinários

```
CREATE TABLE IF NOT EXISTS VETERINARIOS (  
  veterinario_id INT AUTO_INCREMENT PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL,  
  especialidade VARCHAR(100),  
  tipo_veterinario ENUM('pj', 'pf') NOT NULL,  
  documento VARCHAR(20),  
  email VARCHAR(100),  
  observacoes TEXT,  
  data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,  
  ultima_atualizacao TIMESTAMP,  
  ativo TINYINT(1) DEFAULT 1,  
  FOREIGN KEY (endereco_id)  
  REFERENCES ENDERECOS(endereco_id),  
  CONSTRAINT chk_tipo_veterinario CHECK (tipo_veterinario IN ('pj', 'pf')),  
  CONSTRAINT chk_documento CHECK (  
    (tipo_veterinario = 'pf' AND documento = 'CPF') OR  
    (tipo_veterinario = 'pj' AND documento = 'CNPJ')  
  )  
);
```

-- Tabela Pet Shops

```
CREATE TABLE IF NOT EXISTS PETSHOPS (  
  pet_shop_id INT AUTO_INCREMENT PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL,  
  tipo_pet_shop ENUM('pj', 'pf') NOT NULL,  
  documento VARCHAR(20),  
  email VARCHAR(100),  
  observacoes TEXT,  
  data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,  
  ultima_atualizacao TIMESTAMP,  
  ativo TINYINT(1) DEFAULT 1,  
  FOREIGN KEY (endereco_id)  
  REFERENCES ENDERECOS(endereco_id),  
  CONSTRAINT chk_tipo_pet_shop CHECK (tipo_pet_shop IN ('pj', 'pf')),  
  CONSTRAINT chk_documento_pet_shop CHECK (  
    (tipo_pet_shop = 'pf' AND documento = 'CPF') OR  
    (tipo_pet_shop = 'pj' AND documento = 'CNPJ')  
  )  
);
```

-- Tabela Animais

```
CREATE TABLE IF NOT EXISTS ANIMAIS (  
  animal_id INT AUTO_INCREMENT PRIMARY KEY,  
  proprietario_id INT NOT NULL,  
  nome VARCHAR(100) NOT NULL,  
  tipo_animal ENUM('cachorro', 'gato') NOT NULL,  
  raca VARCHAR(100),  
  sexo ENUM('macho', 'femea'),  
  castrado_esterilizado ENUM('sim', 'nao'),  
  data_castracao_esterilizacao DATE,  
  data_nascimento DATE,  
  numero_microchip VARCHAR(50),  
  localizacao_microchip VARCHAR(50),  
  marca_microchip VARCHAR(50),  
  adotado ENUM('sim', 'nao'),  
  cor VARCHAR(50),  
  cor_dos_olhos VARCHAR(50),  
  tipo_de_pelagem VARCHAR(50),  
  porte VARCHAR(50),  
  temperamento VARCHAR(50),  
  nome_mae VARCHAR(100),  
  nome_pai VARCHAR(100),  
  data_falecimento DATETIME,  
  observacoes TEXT,  
  data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,  
  ultima_atualizacao TIMESTAMP,  
  ativo TINYINT(1) DEFAULT 1,  
  FOREIGN KEY (proprietario_id)  
  REFERENCES PROPRIETARIOS(proprietario_id),  
  CONSTRAINT chk_tipo_animal CHECK (tipo_animal IN ('cachorro', 'gato')),  
  CONSTRAINT chk_sexo CHECK (sexo IN ('macho', 'femea')),  
  CONSTRAINT chk_castrado_esterilizado CHECK (castrado_esterilizado IN ('sim',  
'nao'))),  
  CONSTRAINT chk_adotado CHECK (adotado IN ('sim', 'nao'))  
);
```

-- Tabela Cuidados Basicos

```
CREATE TABLE IF NOT EXISTS CUIDADOS_ANIMAIS (  
  cuidado_animal_id INT,  
  animal_id INT,  
  observacoes TEXT,  
  data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,  
  ultima_atualizacao TIMESTAMP,  
  FOREIGN KEY (animal_id)  
  REFERENCES ANIMAIS(animal_id)  
);
```

```

-- Tabela Categorias
CREATE TABLE IF NOT EXISTS CATEGORIAS (
    categoria_id INT,
    cuidado_animal_id INT,
    historico_id INT,
    nome VARCHAR(255),
    observacoes TEXT,
    data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,
    ultima_atualizacao TIMESTAMP,
    FOREIGN KEY (cuidado_animal_id)
    REFERENCES CUIDADOS_ANIMAIS(cuidado_animal_id),
    FOREIGN KEY (historico_id)
    REFERENCES HISTORICO_MEDICO(historico_id)
);

-- Tabela Histórico Médico
CREATE TABLE IF NOT EXISTS HISTORICO_MEDICO (
    historico_id INT AUTO_INCREMENT PRIMARY KEY,
    animal_id INT,
    descricao TEXT,
    Alergia VARCHAR(255),
    Anormalidade VARCHAR(255),
    Cirurgia VARCHAR(255),
    data DATE,
    data_criacao DATETIME DEFAULT CURRENT_TIMESTAMP,
    ultima_atualizacao TIMESTAMP,
    FOREIGN KEY (animal_id)
    REFERENCES ANIMAIS(animal_id)
);

-- Tabela Visitas Veterinárias
CREATE TABLE IF NOT EXISTS VISITAS_VETERINARIAS (
    visita_id INT AUTO_INCREMENT PRIMARY KEY,
    animal_id INT NOT NULL,
    data_visita DATE NOT NULL,
    veterinario_id INT NOT NULL,
    motivo_consulta VARCHAR(255) NOT NULL,
    diagnostico VARCHAR(255),
    plano_tratamento TEXT,
    medicamentos_prescritos TEXT,
    instrucoes_retorno VARCHAR(255),
    data_retorno DATE NOT NULL,
    FOREIGN KEY (animal_id)
    REFERENCES ANIMAIS(animal_id),
    FOREIGN KEY (veterinario_id)
    REFERENCES VETERINARIOS(veterinario_id)
);

```

### Algoritmo 1 – Modelo Físico em SQL

### 3.5 Inserção dos Dados

Após a criação do banco de dados “PetManagementSystem”, foram realizadas 11 inserções diretas nas tabelas utilizando instruções SQL. As inserções abrangeram todas as tabelas do banco de dados, garantindo que todas as relações e dados necessários estivessem presentes.

Os dados inseridos são os seguintes:

- **ESTADOS:** Inserção de cinco estados diferentes.
- **CIDADES:** Inserção de cinco cidades diferentes, cada uma associada a um estado.
- **ENDERECOS:** Inserção de cinco endereços diferentes, cada um associado a uma cidade.
- **PROPRIETARIOS:** Inserção de cinco proprietários diferentes, cada um associado a um endereço.
- **VETERINARIOS:** Inserção de cinco veterinários diferentes, cada um associado a um endereço.
- **PETSHOPS:** Inserção de cinco pet shops diferentes, cada um associado a um endereço.
- **ANIMAIS:** Inserção de cinco animais diferentes, cada um associado a um proprietário.
- **CUIDADOS\_ANIMAIS:** Inserção de cuidados básicos realizados para cinco animais diferentes.
- **HISTORICO\_MEDICO:** Inserção de históricos médicos para cinco animais diferentes.
- **CATEGORIAS:** Inserção de cinco categorias diferentes, associadas a cuidados básicos e históricos médicos.
- **VISITAS\_VETERINARIAS:** Inserção de cinco visitas veterinárias diferentes, cada uma associada a um animal e um veterinário.

Com esses dados inseridos, é possível realizar consultas SQL complexas, que os SGBD (sistemas de gerenciamento de banco de dados) compreendem e executam para manipular e recuperar informações do banco de dados. Essas consultas podem fornecer insights detalhados sobre os diferentes aspectos do sistema de gerenciamento de pets.

### 3.5.1 EXEMPLOS DE CONSULTAS

Nessa seção serão apresentadas 30 consultas possíveis nesse banco de dados criado anteriormente neste trabalho. Cada consulta terá uma descrição, o seu código em SQL e o resultado da busca.

1. Listar todos os estados e suas cidades associadas:

```
SELECT e.estado, c.cidade
FROM ESTADOS e
JOIN CIDADES c ON e.estado_id = c.estado_id;
```

**Algoritmo 2 – Consulta 1**

	estado	cidade
►	São Paulo	São Paulo
	São Paulo	São Paulo
	Rio de Janeiro	Rio de Janeiro
	Rio de Janeiro	Rio de Janeiro
	Minas Gerais	Belo Horizonte
	Minas Gerais	Belo Horizonte
	Bahia	Salvador
	Bahia	Salvador
	Paraná	Curitiba
	Paraná	Curitiba

**Figura 3 – Resultado 1 (Acervo pessoal)**

2. Listar todos os proprietários e seus respectivos endereços:

```
SELECT p.nome, e.endereco, e.bairro, c.cidade, es.estado
FROM PROPRIETARIOS p
JOIN ENDEREÇOS e ON p.endereco_id = e.endereco_id
JOIN CIDADES c ON e.cidade_id = c.cidade_id
JOIN ESTADOS es ON c.estado_id = es.estado_id;
```

**Algoritmo 3 – Consulta 2**

	nome	endereço	bairro	cidade	estado
►	João Silva	Rua A, 123	Centro	São Paulo	São Paulo
	Maria Oliveira	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Carlos Souza	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Ana Costa	Av. D, 101	Pituba	Salvador	Bahia
	Pedro Lima	Rua E, 202	Água Verde	Curitiba	Paraná
	João Silva	Rua A, 123	Centro	São Paulo	São Paulo
	Maria Oliveira	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Carlos Souza	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Ana Costa	Av. D, 101	Pituba	Salvador	Bahia
	Pedro Lima	Rua E, 202	Água Verde	Curitiba	Paraná

**Figura 4** – Resultado 2 (Acervo pessoal)

3. Listar todos os veterinários ativos:

```
SELECT nome, especialidade, tipo_veterinario, documento
FROM VETERINARIOS
WHERE ativo = 1;
```

**Algoritmo 4** – Consulta 3

	nome	especialidade	tipo_veterinario	documento
►	Dra. Fernanda Martins	Cardiologia	pf	CPF
	Dr. Ricardo Alves	Dermatologia	pf	CPF
	Dra. Luisa Ramos	Ortopedia	pf	CPF
	Vet Saúde LTDA	Clínica Geral	pj	CNPJ
	Dr. Marcos Pinto	Cirurgia	pf	CPF
	Dra. Fernanda Martins	Cardiologia	pf	CPF
	Dr. Ricardo Alves	Dermatologia	pf	CPF
	Dra. Luisa Ramos	Ortopedia	pf	CPF
	Vet Saúde LTDA	Clínica Geral	pj	CNPJ
	Dr. Marcos Pinto	Cirurgia	pf	CPF

**Figura 5** – Resultado 3 (Acervo pessoal)

4. Listar todos os animais e seus proprietários:

```
SELECT a.nome AS nome_animal, a.tipo_animal, p.nome AS nome_proprietario
FROM ANIMAIS a
JOIN PROPRIETARIOS p ON a.proprietario_id = p.proprietario_id;
```

**Algoritmo 5** – Consulta 4

	nome_animal	tipo_animal	nome_proprietario
▶	Rex	cachorro	João Silva
	Mia	gato	Maria Oliveira
	Buddy	cachorro	Carlos Souza
	Luna	gato	Ana Costa
	Thor	cachorro	Pedro Lima
	Rex	cachorro	João Silva
	Mia	gato	Maria Oliveira
	Buddy	cachorro	Carlos Souza
	Luna	gato	Ana Costa
	Thor	cachorro	Pedro Lima

**Figura 6** – Resultado 4 (Acervo pessoal)

5. Listar todos os pet shops com seus endereços:

```
SELECT ps.nome, e.endereco, e.bairro, c.cidade, es.estado
FROM PETSHOPS ps
JOIN ENDEREÇOS e ON ps.endereco_id = e.endereco_id
JOIN CIDADES c ON e.cidade_id = c.cidade_id
JOIN ESTADOS es ON c.estado_id = es.estado_id;
```

**Algoritmo 6** – Consulta 5

	nome	endereco	bairro	cidade	estado
▶	Pet Love	Rua A, 123	Centro	São Paulo	São Paulo
	Pet Store	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Pet Center	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Pet Point	Av. D, 101	Pituba	Salvador	Bahia
	Pet Care	Rua E, 202	Água Verde	Curitiba	Paraná
	Pet Love	Rua A, 123	Centro	São Paulo	São Paulo
	Pet Store	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Pet Center	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Pet Point	Av. D, 101	Pituba	Salvador	Bahia
	Pet Care	Rua E, 202	Água Verde	Curitiba	Paraná

**Figura 7** – Resultado 5 (Acervo pessoal)

6. Listar todas as visitas veterinárias de um animal específico:

```
SELECT v.data_visita, v.motivo_consulta, v.diagnostico
FROM VISITAS_VETERINARIAS v
JOIN ANIMAIS a ON v.animal_id = a.animal_id
WHERE a.nome = 'Nome do Animal';
```

**Algoritmo 7** – Consulta 6

	data_visita	motivo_consulta	diagnostico
▶	2023-04-10	Consulta de rotina	Saudável
	2023-04-10	Consulta de rotina	Saudável

**Figura 8** – Resultado 6 (Acervo pessoal)

7. Listar todos os cuidados básicos de um animal específico:

```
SELECT c.observacoes
FROM CUIDADOS_ANIMAIS c
JOIN ANIMAIS a ON c.animal_id = a.animal_id
WHERE a.nome = 'Nome do Animal';
```

**Algoritmo 8** – Consulta 7

	observacoes
▶	Banho e tosa mensal
	Banho e tosa mensal

**Figura 9** – Resultado 7 (Acervo pessoal)

8. Listar todos os históricos médicos de um animal específico:

```
SELECT h.descricao, h.Alergia, h.Anormalidade, h.Cirurgia
FROM HISTORICO_MEDICO h
JOIN ANIMAIS a ON h.animal_id = a.animal_id
WHERE a.nome = 'Nome do Animal';
```

**Algoritmo 9** – Consulta 8

	descricao	Alergia	Anormalidade	Cirurgia
▶	Alergia a pulgas	Pulgas	NULL	NULL
	Alergia a pulgas	Pulgas	NULL	NULL

**Figura 10** – Resultado 8 (Acervo pessoal)

9. Listar todas as categorias associadas a um cuidado básico específico:

```
SELECT ca.nome, ca.observacoes
FROM CATEGORIAS ca
JOIN CUIDADOS_ANIMAIS cu ON ca.cuidado_animal_id = cu.cuidado_animal_id
WHERE cu.observacoes = 'Observação do Cuidado';
```

**Algoritmo 10** – Consulta 9



	nome	observacoes
►	Cuidados Gerais	Cuidados regulares
	Cuidados Gerais	Cuidados regulares

**Figura 11** – Resultado 9 (Acervo pessoal)

10. Listar todas as categorias associadas a um histórico médico específico:

```
SELECT ca.nome, ca.observacoes
FROM CATEGORIAS ca
JOIN HISTORICO_MEDICO h ON ca.historico_id = h.historico_id
WHERE h.descricao = 'Descrição do Histórico';
```

**Algoritmo 11** – Consulta 10

	nome	observacoes
►	Cirurgias	Histórico de cirurgias
	Cirurgias	Histórico de cirurgias

11. Listar todos os animais castrados:

```
SELECT nome, tipo_animal, raca, data_castracao_esterilizacao
FROM ANIMAIS
WHERE castrado_esterilizado = 'sim';
```

**Algoritmo 12** – Consulta 11

	nome	tipo_animal	raca	data_castracao_esterilizacao
►	Rex	cachorro	Labrador	2020-01-15
	Buddy	cachorro	Poodle	2019-11-25
	Luna	gato	Persa	2022-05-14
	Rex	cachorro	Labrador	2020-01-15
	Buddy	cachorro	Poodle	2019-11-25
	Luna	gato	Persa	2022-05-14

12. Listar todos os animais adotados:

```
SELECT nome, tipo_animal, raca
FROM ANIMAIS
WHERE adotado = 'sim';
```

**Algoritmo 13** – Consulta 12

	nome	tipo_animal	raca
►	Rex	cachorro	Labrador
	Buddy	cachorro	Poodle
	Thor	cachorro	Bulldog
	Rex	cachorro	Labrador
	Buddy	cachorro	Poodle
	Thor	cachorro	Bulldog

13. Listar todos os animais falecidos:

```
SELECT nome, tipo_animal, data_falecimento
FROM ANIMAIS
WHERE data_falecimento IS NOT NULL;
```

#### Algoritmo 14 – Consulta 13

	nome	tipo_animal	data_falecimento
--	------	-------------	------------------

14. Listar todos os animais de uma determinada raça:

```
SELECT nome, tipo_animal, raca
FROM ANIMAIS
WHERE raca = 'Raça Específica';
```

#### Algoritmo 15 – Consulta 14

	nome	tipo_animal	raca
►	Rex	cachorro	Labrador
	Rex	cachorro	Labrador

15. Listar todos os animais de um determinado proprietário:

```
SELECT a.nome, a.tipo_animal, a.raca
FROM ANIMAIS a
JOIN PROPRIETARIOS p ON a.proprietario_id = p.proprietario_id
WHERE p.nome = 'Nome do Proprietário';
```

#### Algoritmo 16 – Consulta 15

	nome	tipo_animal	raca
▶	Buddy	cachorro	Poodle
	Buddy	cachorro	Poodle

16. Listar todos os animais com microchip:

```
SELECT nome, tipo_animal, numero_microchip
FROM ANIMAIS
WHERE numero_microchip IS NOT NULL;
```

#### Algoritmo 17 – Consulta 16

	nome	tipo_animal	numero_microchip
▶	Rex	cachorro	1234567890
	Buddy	cachorro	0987654321
	Luna	gato	1122334455
	Rex	cachorro	1234567890
	Buddy	cachorro	0987654321
	Luna	gato	1122334455

17. Listar todos os proprietários que possuem mais de um animal:

```
SELECT p.nome, COUNT(a.animal_id) AS numero_animais
FROM PROPRIETARIOS p
JOIN ANIMAIS a ON p.proprietario_id = a.proprietario_id
GROUP BY p.nome
HAVING COUNT(a.animal_id) > 1;
```

#### Algoritmo 18 – Consulta 17

	nome	numero_animais
▶	João Silva	2
	Maria Oliveira	2
	Carlos Souza	2
	Ana Costa	2
	Pedro Lima	2

18. Listar todos os veterinários e o número de visitas realizadas:

```
SELECT v.nome, COUNT(vv.visita_id) AS numero_visitas
FROM VETERINARIOS v
JOIN VISITAS_VETERINARIAS vv ON v.veterinario_id = vv.veterinario_id
GROUP BY v.nome;
```

#### Algoritmo 19 – Consulta 18

	nome	numero_visitas
▶	Dra. Fernanda Martins	2
	Dr. Ricardo Alves	2
	Dra. Luisa Ramos	2
	Vet Saúde LTDA	2
	Dr. Marcos Pinto	2

19. Listar todos os cuidados básicos realizados no último mês:

```
SELECT ca.observacoes
FROM CUIDADOS_ANIMAIS ca
WHERE ca.data_criacao >= DATE_SUB(CURRENT_DATE, INTERVAL 1 MONTH);
```

#### Algoritmo 20 – Consulta 19

	observacoes
▶	Banho e tosa mensal
	Vacinação anual
	Dieta especial
	Consulta trimestral
	Exames periódicos
	Banho e tosa mensal
	Vacinação anual
	Dieta especial
	Consulta trimestral
	Exames periódicos

20. Listar todos os históricos médicos registrados no último ano:

```
SELECT h.descricao, h.data
FROM HISTORICO_MEDICO h
WHERE h.data >= DATE_SUB(CURRENT_DATE, INTERVAL 1 YEAR);
```

#### Algoritmo 21 – Consulta 20



	nome	descricao	data
►	Buddy	Problema digestivo	2023-01-15
	Buddy	Problema digestivo	2023-01-15
	Luna	Vacinação contra raiva	2022-09-01
	Luna	Vacinação contra raiva	2022-09-01
	Mia	Cirurgia de esterilização	2022-03-20
	Mia	Cirurgia de esterilização	2022-03-20
	Thor	Fratura na pata	2021-12-05
	Thor	Fratura na pata	2021-12-05
	Rex	Alergia a pulgas	2021-05-10
	Rex	Alergia a pulgas	2021-05-10

23. Listar todos os animais de um determinado tipo (cachorro ou gato):

```
SELECT nome, raca, sexo
FROM ANIMAIS
WHERE tipo_animal = 'cachorro';
```

#### Algoritmo 24 – Consulta 23

	nome	raca	sexo
►	Rex	Labrador	macho
	Buddy	Poodle	macho
	Thor	Bulldog	macho

24. Listar todos os animais e suas cores:

```
SELECT nome, tipo_animal, cor
FROM ANIMAIS;
```

#### Algoritmo 25 – Consulta 24

	nome	tipo_animal	cor
►	Rex	cachorro	Preto
	Mia	gato	Branco
	Buddy	cachorro	Branco
	Luna	gato	Cinza
	Thor	cachorro	Marrom

25. Listar todos os veterinários com uma determinada especialidade:

```
SELECT nome, tipo_veterinario, documento
FROM VETERINARIOS
WHERE especialidade = 'Especialidade Específica';
```

#### Algoritmo 26 – Consulta25

	nome	tipo_veterinario	documento
▶	Dra. Fernanda Martins	pf	CPF

26. Listar todos os proprietários e seus animais adotados:

```
SELECT p.nome, a.nome, a.tipo_animal
FROM PROPRIETARIOS p
JOIN ANIMAIS a ON p.proprietario_id = a.proprietario_id
WHERE a.adotado = 'sim';
```

#### Algoritmo 27 – Consulta 26

	nome	nome	tipo_animal
▶	João Silva	Rex	cachorro
	Carlos Souza	Buddy	cachorro
	Pedro Lima	Thor	cachorro

27. Listar todos os pet shops ativos:

```
SELECT nome, tipo_pet_shop, documento
FROM PETSHOPS
WHERE ativo = 1;
```

#### Algoritmo 28 – Consulta 27

	nome	tipo_pet_shop	documento
▶	Pet Love	pj	CNPJ
	Pet Store	pj	CNPJ
	Pet Center	pj	CNPJ
	Pet Point	pj	CNPJ
	Pet Care	pj	CNPJ

28. Listar todos os veterinários e seus endereços:

```
SELECT v.nome, e.endereco, e.bairro, c.cidade, es.estado
FROM VETERINARIOS v
JOIN ENDEREÇOS e ON v.endereco_id = e.endereco_id
JOIN CIDADES c ON e.cidade_id = c.cidade_id
JOIN ESTADOS es ON c.estado_id = es.estado_id;
```

#### Algoritmo 29 – Consulta 28

	nome	endereco	bairro	cidade	estado
►	Dra. Fernanda Martins	Rua A, 123	Centro	São Paulo	São Paulo
	Dr. Ricardo Alves	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Dra. Luisa Ramos	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Vet Saúde LTDA	Av. D, 101	Pituba	Salvador	Bahia
	Dr. Marcos Pinto	Rua E, 202	Água Verde	Curitiba	Paraná

29. Listar todos os pet shops e seus endereços:

```
SELECT ps.nome, e.endereco, e.bairro, c.cidade, es.estado
FROM PETSHOPS ps
JOIN ENDEREÇOS e ON ps.endereco_id = e.endereco_id
JOIN CIDADES c ON e.cidade_id = c.cidade_id
JOIN ESTADOS es ON c.estado_id = es.estado_id;
```

#### Algoritmo 30 – Consulta 29

	nome	endereco	bairro	cidade	estado
►	Pet Love	Rua A, 123	Centro	São Paulo	São Paulo
	Pet Store	Av. B, 456	Zona Sul	Rio de Janeiro	Rio de Janeiro
	Pet Center	Rua C, 789	Bairro Novo	Belo Horizonte	Minas Gerais
	Pet Point	Av. D, 101	Pituba	Salvador	Bahia
	Pet Care	Rua E, 202	Água Verde	Curitiba	Paraná

30. Listar todos os pet shops e seus endereços:

```
SELECT p.nome, COUNT(a.animal_id) AS numero_animais
FROM PROPRIETARIOS p
JOIN ANIMAIS a ON p.proprietario_id = a.proprietario_id
GROUP BY p.nome;
```

#### Algoritmo 31 – Consulta 30



	nome	numero_animais
▶	João Silva	2
	Maria Oliveira	2
	Carlos Souza	2
	Ana Costa	2
	Pedro Lima	2

Essas consultas permitem explorar diversas informações presentes na base de dados, desde dados de animais até detalhes de proprietários, veterinários, pet shops, e eventos médicos e cuidados.

#### 4 CONCLUSÃO

A conclusão deste projeto reforçará a importância da integração de informações no contexto do cuidado com animais de estimação. Ao criar um Sistema Integrado para Gerenciamento, demonstramos que é possível preencher lacunas na gestão de dados relacionados a animais de estimação, oferecendo uma solução inovadora e completa. A consolidação do histórico médico e a facilitação do acesso a serviços especializados contribuem para uma abordagem mais eficaz e conectada ao cuidado com os animais de estimação. A conclusão não apenas recapitulará os principais resultados, mas também delineará perspectivas futuras, considerando possíveis expansões e aprimoramentos do sistema.

## REFERÊNCIAS

GUEDES, Gilleanes. **UML2: Uma Abordagem Pratica**. 3. ed. São Paulo: Novatec, 2018.

HEUSER, Carlos Alberto. **Projeto de Banco de Dados: Volume 4**. 6. ed. São Paulo: Atlas, 2009.

ROB, Peter; CORONEL Carlos. **Sistema de Banco de Dados: PROJETO, IMPLEMENTAÇÃO e GERENCIAMENTO** 8. ed. São Paulo: Cengage Learning, 2014

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS – ABNT. **NBR 6023**: informação / referências / elaboração. Rio de Janeiro, 2002.

\_\_\_\_\_. **NBR 6024**: numeração progressiva das seções de um documento. Rio de Janeiro, 1989.

**Obs.:** Leis Federais tem sua entrada por BRASIL; Leis Estaduais e Municipais tem a sua entrada pelo nome do Estado e Município, respectivamente. No caso de São Paulo e Rio de Janeiro, como capital e estado tem o mesmo nome, usa-se SÃO PAULO para leis Estaduais e SÃO PAULO (SP) para leis Municipais.

## GLOSSÁRIO

**Sistema Integrado:** Um sistema integrado liga todos os softwares de negócios usados em uma organização. A vantagem disto é que ele elimina silos e fornece uma visão unificada das informações internas. Basicamente, é uma tecnologia que “amarra” os setores e seus processos.

**UML:** Um acrônimo para a expressão “*Unified Modeling Language*” ou simplesmente “*Linguagem de Modelagem Unificada*”, o UML é uma linguagem de notação para uso em projetos de sistema.

**SQL:** “*Structured Query Language*”, ou Linguagem Estruturada de Consultas, em português. Mas não é só “uma” linguagem: é a linguagem padrão universal, usada por cientistas de dados para manipular bancos de dados por meio dos SGBD’s.

**SGBD:** é o conjunto de programas de computador responsáveis pelo gerenciamento de bases de dados. O principal objetivo é retirar da aplicação cliente a responsabilidade de gerenciar o acesso, manipulação e organização dos dados.