

**POSTO VETERINÁRIO**

**Bases de Dados**

**Turma 1 Grupo 05**

João Leite / up201705312@fe.up.pt

João Campos / up201704982@fe.up.pt

Márcia Teixeira / up201706065@fe.up.pt

17 de março de 2019

**Índice**

[Introdução 4](#_Toc5979219)

[1. Contexto 5](#_Toc5979220)

[2. Diagrama UML Inicial 6](#_Toc5979221)

[3. Diagrama UML Revisto 7](#_Toc5979222)

[4. Classes 8](#_Toc5979224)

[4.1. Posto 8](#_Toc5979225)

[4.2. Pessoa 8](#_Toc5979226)

[4.2.1. Funcionário 8](#_Toc5979227)

[4.2.1.1. Médico 8](#_Toc5979228)

[4.2.1.2. Outro 8](#_Toc5979229)

[4.2.2. Cliente 9](#_Toc5979230)

[4.3. Especialidade 9](#_Toc5979231)

[4.4. Animal 9](#_Toc5979232)

[4.5. Problema 9](#_Toc5979233)

[4.6. Espécie 9](#_Toc5979234)

[4.7. Tratamento 10](#_Toc5979235)

[4.8. Consulta 10](#_Toc5979236)

[4.9. Horário Atendimento 10](#_Toc5979237)

[5. Esquema relacional 11](#_Toc5979238)

[6. Análise de Dependências Funcionais e Formas Normais 12](#_Toc5979239)

[6.1. Relação Posto 12](#_Toc5979240)

[6.2. Relação Pessoa 12](#_Toc5979241)

[6.3. Relação Cliente 12](#_Toc5979242)

[6.4. Relação Funcionario 12](#_Toc5979243)

[6.5. Relação Medico 12](#_Toc5979244)

[6.6. Relação Outro 13](#_Toc5979245)

[6.7. Relação Especialidade 13](#_Toc5979246)

[6.8. Relação HorarioAtendimento 13](#_Toc5979247)

[6.9. Relação Animal 13](#_Toc5979248)

[6.10. Relação Especie 13](#_Toc5979249)

[6.11. Relação Problema 14](#_Toc5979250)

[6.12. Relação HistoricoProblemas 14](#_Toc5979251)

[6.13. Relação PossiveisProblemas 14](#_Toc5979252)

[6.14. Relação Tratamento 14](#_Toc5979253)

[6.15. Relação TratamentosRecomendados 14](#_Toc5979254)

[6.16. Relação Consulta 14](#_Toc5979255)

[6.17. Relação TratamentosConsulta 15](#_Toc5979256)

# 

# Introdução

Este trabalho, realizado para a unidade curricular de Bases de Dados, tem como objetivo avaliar a capacidade dos estudantes criarem uma base de dados, tendo esta segunda entrega como objetivo o melhoramento do diagrama UML, o mapeamento do modelo anteriormente criado para um esquema relacional e implementação deste esquema numa base de dados SQLite. Este relatório tem como objetivo descrever o modo como a base de dados pretendida estará organizada.

# Contexto

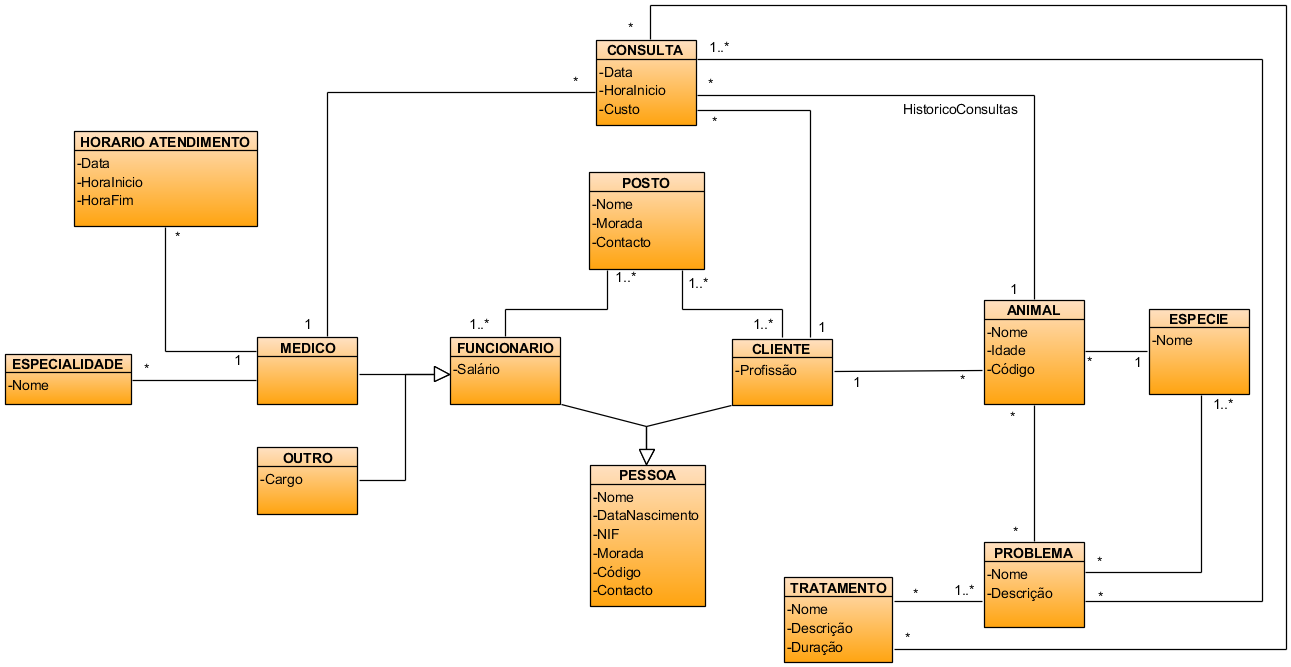
Este trabalho consistirá em fazer uma base de dados para um posto veterinário. Cada posto terá um nome, uma morada e um contacto.

Ao posto estarão associadas pessoas, caracterizadas por um nome, um NIF, um código identificador, uma morada, a sua data de nascimento e um contacto. As pessoas podem ser divididas em dois tipos, funcionários ou clientes, sendo que os clientes terão também a sua profissão e os funcionários terão o seu salário, sendo também divididos em dois tipos, médicos ou “outro”. O “outro” será identificado pelo seu cargo, já o médico terá associadas uma especialidade e um horário de atendimento - que guarda a data e horas de início e fim de atendimento para essa data.

Os clientes terão um conjunto de animais, caracterizados por um nome, a sua idade e um código identificativo, assim como uma espécie associada. Aos animais estarão associados diversos problemas, com um determinado nome e descrição. Estes problemas terão diferentes tratamentos possíveis.

O posto terá ainda consultas, com uma certa data, hora e custo, associadas a um determinado médico, animal e cliente (que pode ou não ser o dono do animal) e pode estar associada a um conjunto de problemas abordados e tratamentos recomendados.

# Diagrama UML Inicial



Acima encontra-se o diagrama UML criado pelo nosso grupo na primeira fase do trabalho, tendo em conta uma ideia geral de um posto veterinário.

Este UML foi depois revisto, tendo sido resolvidos alguns problemas do mesmo.

# Diagrama UML Revisto

# 

# Classes

# Posto

Um objeto desta classe representa um posto veterinário, identificado pelo seu nome e morada, tendo também um contacto.

**Restrições:** O contacto deve ser um número com 9 algarismos.

# Pessoa

Uma pessoa é, no contexto do posto veterinário, identificada por um código. Uma pessoa é também caracterizada pelo seu nome, data de nascimento, morada, NIF e contacto.

**Restrições:** O contacto deve ser um número com 9 algarismos e o código é constituído por uma letra seguida por 6 números.

# Funcionário

Um funcionário é um tipo de pessoa que trabalha no posto veterinário, tendo, para além dos atributos de pessoa, um salário.

**Restrições:** A letra do código identificativo dos funcionários é ‘F’.

# Médico

Um objeto desta classe representa um médico veterinário, que é um tipo de funcionário, tendo os mesmos atributos.

# Outro

Esta classe representa “outros” funcionários, isto é, funcionários que não são médicos. Este tipo de funcionários tem como atributo o cargo que exerce (para além dos atributos de funcionário). Este cargo não pode ser “médico” ou “médico veterinário”, já que estes fazem parte da classe referida acima.

**Restrições:** O cargo não pode ser “médico (veterinário)”, já que estes funcionários fazem parte da classe Médico.

# Cliente

Um cliente é um tipo de pessoa que é cliente do posto veterinário. Para além dos atributos de pessoa, tem também como atributo a sua profissão.

**Restrições:** A letra do código identificativo dos clientes é ‘C’.

# Especialidade

Esta classe representa a especialidade de um médico do posto, tendo como atributo apenas o nome da especialidade.

# Animal

Um animal é identificado, no contexto da clínica, por um código. Tem também como atributos o seu nome e idade, e está associado à sua espécie e ao seu dono (cliente do posto). Pode também ter associados vários problemas, que representam o histórico de problemas (problemas passados e presentes) do animal.

**Restrições:** O código é constituído pela letra ‘A’ seguida por 6 números.

# Problema

Um objeto desta classe representa um problema de saúde que pode afetar os animais, sendo caracterizado pelo seu nome e descrição. Um problema pode ter vários tratamentos associados, que representam os tratamentos que podem ser usados para tratar o problema.

**Restrições:** Sempre que um animal apresenta um problema numa consulta, este problema deve poder afetar a sua espécie (estar associado à mesma).

# Espécie

Esta classe representa a espécie de um animal, sendo identificada pelo seu nome. Uma espécie pode estar associada a vários problemas, que representam os problemas que podem afetar animais daquela espécie.

# Tratamento

Esta classe representa um tratamento que pode ser recomendado para tratar certos problemas, tendo como atributos o seu nome, uma descrição, e a sua duração.

**Restrições:** Sempre que, numa consulta, é recomendado um tratamento, este deve ser adequado a um dos problemas que o animal apresentou nessa consulta.

# Consulta

Esta classe representa uma consulta dada por um médico veterinário a um animal, tendo como atributos a data da consulta, a sua hora de início e o seu custo. A uma consulta estão associados um médico (que dá a consulta), um animal (paciente) e um cliente (que paga a consulta). Podem também estar associados vários tratamentos que tenham sido feitos ou prescritos pelo médico na consulta.

**Restrições:** A hora de início da consulta não pode ser anterior à hora de início de atividade constante no horário de atendimento do médico que dá a consulta para aquela data, nem igual ou posterior à hora de fim de atendimento. A hora de início não pode também coincidir com a hora de início de outra consulta do mesmo médico naquela data.

# Horário Atendimento

Esta classe representa o horário de atendimento de um médico para um determinado dia, tendo como atributos a data a que se refere, a hora em que se inicia o atendimento e a hora em que este termina.

**Restrições:** A hora de fim de atendimento deve ser posterior à hora de início de atendimento.

# Esquema relacional

Posto (nome, morada, contacto)

Pessoa (codigo, nome, dataNasc, NIF, morada, contacto)

Cliente (codigoCliente-> Pessoa, nome, dataNasc, NIF, morada, contacto, profissao)

Funcionario (codigoFuncionario-> Pessoa, nome, dataNasc, NIF, morada, contacto, salario)

Medico (codigoMedico-> Funcionario, nome, dataNasc, NIF, morada, contacto, salario, nomeEspecialidade->Especialidade)

Outro (codigoFuncionario-> Funcionario, nome, dataNasc, NIF, morada, contacto, salario, cargo)

Especialidade (nomeEsp)

HorarioAtendimento (codigoMedico-> Medico, data, horaInicio, horaFim)

Animal (codigoAnimal, codigoCliente->Cliente, idade, nome, nomeEspecie->Especie)

Especie (nomeEspecie)

Problema (nomeProblema, descricao)

HistoricoProblemas (codigoAnimal-> Animal, nomeProblema-> Problema)

PossiveisProblemas (nomeEspecie-> Especie, nomeProblema-> Problema)

Tratamento (nomeTratamento, descricao, duracao)

TratamentosRecomendados (nomeProblema-> Problema, nomeTratamento-> Tratamento)

Consulta (data, codigoAnimal-> Animal, codigoMedico-> Medico, codigo-> Cliente, custo)

TratamentosConsulta (data-> Consulta, codigoAnimal-> Animal-> Consulta, nomeTratamento-> Tratamento)

# Análise de Dependências Funcionais e Formas Normais

## Relação Posto

nome, morada-> contacto

Esta relação está na 3NF e na BCNF, pois {nome, morada} é superkey da relação.

## Relação Pessoa

codigo-> nome, dataNasc, NIF, morada, contacto  
NIF-> codigo, nome, dataNasc, morada, contactonome, dataNasc, morada, contacto -> codigo, NIF

Esta relação está na 3NF e na BCNF, pois {codigo}, {NIF} e {nome, dataNasc, morada, contacto} são superkeys da relação.

## Relação Cliente

codigoCliente-> nome, dataNasc, NIF, morada, contacto, profissao  
NIF-> codigoCliente, nome, dataNasc, morada, contacto, profissaonome, dataNasc, morada, contacto -> codigoCliente, NIF, profissão

Esta relação está na 3NF e na BCNF, pois {codigoCliente}, {NIF} e {nome, dataNasc, morada, contacto} são superkeys da relação.

## Relação Funcionario

codigoFuncionario-> nome, dataNasc, NIF, morada, contacto, salario  
NIF-> codigoFuncionario, nome, dataNasc, morada, contacto, salarionome, dataNasc, morada, contacto -> codigoFuncionario, NIF, salario

Esta relação está na 3NF e na BCNF, pois {codigoFuncionario}, {NIF} e {nome, dataNasc, morada, contacto} são superkeys da relação.

## Relação Medico

codigoMedico-> nome, dataNasc, NIF, morada, contacto, salario, nomeEspecialidade  
NIF-> codigoMedico, nome, dataNasc, morada, contacto, salario, nomeEspecialidadenome, dataNasc, morada, contacto -> codigoMedico, NIF, salario, nomeEspecialidade

Esta relação está na 3NF e na BCNF, pois {codigoMedico}, {NIF} e {nome, dataNasc, morada, contacto} são superkeys da relação.

## Relação Outro

codigoFuncionario-> nome, dataNasc, NIF, morada, contacto, salario, cargo  
NIF-> codigoFuncionario, nome, dataNasc, morada, contacto, salario, cargonome, dataNasc, morada, contacto -> codigoFuncionario, NIF, salario, cargo

Esta relação está na 3NF e na BCNF, pois {codigoFuncionario}, {NIF} e {nome, dataNasc, morada, contacto} são superkeys da relação.

## Relação Especialidade

nomeEsp -> nomeEsp

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na 3NF e na BCNF pois não existem Dependências Funcionais não-triviais.

## Relação HorarioAtendimento

codigoMedico, data-> horaInicio, horaFim

Esta relação está na 3NF e na BCNF, pois {codigoMedico, data} é superkey da relação.

## Relação Animal

codigoAnimal-> codigoCliente, idade, nome, nomeEspecie

Esta relação está na 3NF e na BCNF, pois {codigoAnimal} é superkey da relação.

## Relação Especie

nomeEspecie -> nomeEspecie;

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na 3NF e na BCNF, pois não existem Dependências Funcionais não-triviais.

## Relação Problema

nomeProblema-> descrição

Esta relação está na 3NF e na BCNF, pois {nomeProblema} é superkey da relação.

## Relação HistoricoProblemas

codigoAnimal, nomeProblema -> codigoAnimal, nomeProblema

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na 3NF e na BCNF, pois não existem Dependências Funcionais não-triviais.

## Relação PossiveisProblemas

nomeEspecie, nomeProblema -> nomeEspecie, nomeProblema

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na 3NF e na BCNF, pois não existem Dependências Funcionais não-triviais.

## Relação Tratamento

nomeTratamento-> descricao, duração

Esta relação está na 3NF e na BCNF, pois {nomeTratamento} é superkey da relação.

## Relação TratamentosRecomendados

nomeProblema, nomeTratamento -> nomeProblema, nomeTratamento

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na BCNF, pois não existem Dependências Funcionais não-triviais.

## Relação Consulta

data, codigoAnimal -> codigoMedico, codigoCliente, horaInicio, custo

data, codigoMedico, horaInicio -> codigoAnimal, codigoCliente, custo

Esta relação está na 3NF e na BCNF, pois {data, codigoAnimal} e {data, codigoMedico, horaInicio} são superkeys da relação.

## Relação TratamentosConsulta

data, codigoAnimal, nomeTratamento -> data, codigoAnimal, nomeTratamento

(O conjunto de todos os atributos é a Primary Key da relação, não há Dependências Funcionais não-triviais).

Esta relação está na 3NF e na BCNF, pois não existem Dependências Funcionais não-triviais.