

**EIC0023 - Base de Dados**

## **Gestão de um Jardim Zoológico**

**Mestrado Integrado em Engenharia Informática e Computação**

**Prof: Carla Alexandra Teixeira Lopes**

**Porto, 06 de Abril de 2014**

**Elementos do Grupo:**

Joana Lopes Beleza – [ei12029@fe.up.pt](mailto:ei12029@fe.up.pt)

Mário André Ferreira Macedo – [ei12105@fe.up.pt](mailto:ei12105@fe.up.pt)

Pedro Miguel Martins de Lemos da Cunha Faria – [ei12097@fe.up.pt](mailto:ei12097@fe.up.pt)

## **Gestão de um Jardim Zoológico**

Com esta Base de Dados pretendemos fazer um inventario dos animais de um Jardim Zoológico, assim como dos tratamentos aos quais estiveram sujeitos e as condições das suas jaulas.

A nossa classe central Animal indica o nome, idade e peso respectivo de um determinado animal e associa-o a uma e só uma espécie.

Assim como acontece num ordinário Jardim Zoológico, pode existir um ou mais tratadores associados a um ou mais animais estando estes distribuídos por Jaulas, de diferentes tipos, que necessitam de limpezas periódicas.

As limpezas são realizadas numa data e podem envolver diferentes materiais sobre os quais é necessário saber a sua quantidade.

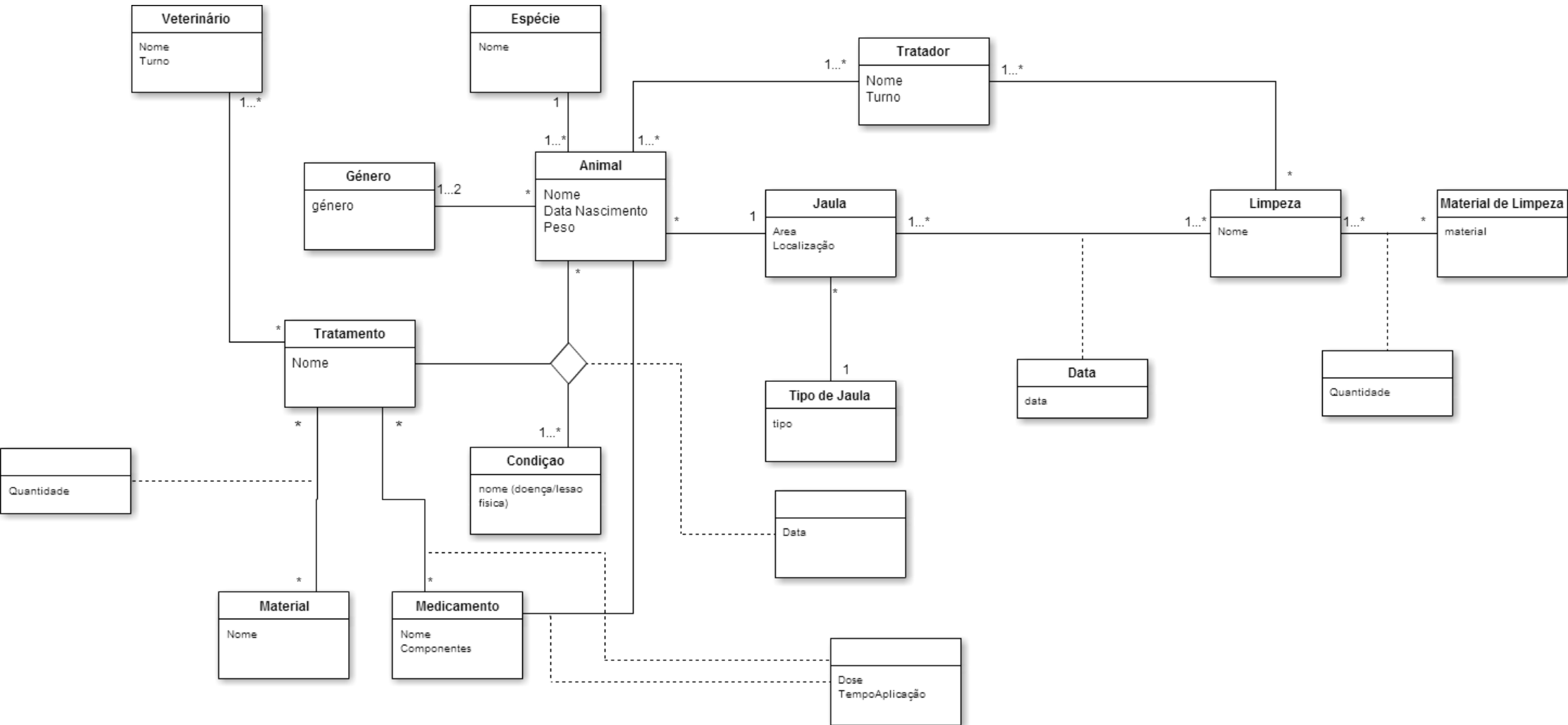
Por outro lado, o animal pode necessitar de tratamento devido a lesões físicas ou devido a doenças. Em qualquer um dos casos, o tratamento será acompanhado por um ou mais veterinários, podendo estes fazer vários tratamentos em simultâneo. O mesmo animal poderá necessitar de diferentes tratamentos em simultâneo, podendo os mesmos ser feitos pelo mesmo veterinário ou por veterinários diferentes. Para além disso, o tratamento associado a si uma data.

De modo a efectuar o tratamento, o veterinário poderá utilizar alguns materiais, medicamentos ou até mesmo ambos, conforme a situação do animal:

Os medicamentos são identificados através do nome e da sua composição, sendo administrados em doses, de acordo com o peso do animal em questão e durante períodos de aplicação que podem variar.

Os materiais utilizados no tratamento de um animal são identificados através de um nome, podendo ser utilizadas varias unidades do mesmo material, por exemplo, compressas.

# UML



## Modelo Relacional

- Animal(idAnimal, Nome, Data de nascimento, Peso, Nome->Espécie, gênero->Gênero, idJaula->Jaula, nome->Condição, idTratador->Tratador)
- Espécie(Nome)
- Gênero(gênero)
- Jaula(idJaula, área, localização, tipo->Tipo de Jaula, idLimpeza->Limpeza)
- Tipo de Jaula(tipo)
- Limpeza(idLimpeza, nome, idTratador->Tratador, nome->Material de Limpeza)
- DataLimpeza(idJaula->Jaula, idLimpeza->Limpeza, data)
- Material de Limpeza(material)
- QuantMaterialLimpeza(idLimpeza->Limpeza, nome->MaterialdeLimpeza, quantidade)
- Tratador(idTratador, nome, turno)
- Veterinário(idVeterinario, nome, turno, idTratamento->Tratamento)
- Condição(idAnimal->Animal, idTratamento->Tratamento, nome)
- Tratamento(idTratamento, nome->Condição, idAnimal->Animal, nome)
- DataTratamento(idTratamento->Tratamento, nome->Condição, idAnimal->Animal, data)
- Medicamento(idMedicamento, idTratamento->Tratamento, nome, componentes)
- Dosagem(dose, tempo de aplicação, peso->Animal, idMedicamento->Medicamento, idTratamento->Tratamento)
- Material(idTratamento->Tratamento, nome)
- QuantidadeMaterial(idTratamento->Tratamento, nome->Material, quantidade)

## Criação de Tabelas Sqlite

.mode columns

.headers on

.nullvalue NULL

pragma foreign\_key=on;

drop table if exists Animal;

create table Animal(idAnimal integer PRIMARY KEY, Nome text, DataDeNascimento text, Peso integer, NomeEspecie text REFERENCES Especie(Nome), genero text REFERENCES Genero(genero), idJaula integer REFERENCES Jaula(idJaula), condicao text REFERENCES Condicao(nome), idTratador integer REFERENCES Tratador(idTratador));

drop table if exists Especie;

create table Especie(Nome text PRIMARY KEY);

drop table if exists Genero;

create table Genero(genero text PRIMARY KEY);

drop table if exists Jaula;

create table Jaula(idJaula integer PRIMARY KEY, area integer, localizacao text, tipo text REFERENCES TipoDeJaula(tipo), idLimpeza integer REFERENCES Limpeza(idLimpeza));

drop table if exists TipoDeJaula;

create table TipoDeJaula(tipo text PRIMARY KEY);

drop table if exists Limpeza;

create table Limpeza(idLimpeza integer PRIMARY KEY, nome text, idTratador integer REFERENCES Tratador(idTratador), material text REFERENCES MaterialDeLimpeza(material));

drop table if exists DataLimpeza;

create table DataLimpeza(idJaula integer REFERENCES Jaula(idJaula), idLimpeza integer REFERENCES Limpeza(idLimpeza), dataLimpeza text);

drop table if exists MaterialDeLimpeza;

create table MaterialDeLimpeza(material text PRIMARY KEY);

drop table if exists QuantMaterialLimpeza;

create table QuantMaterialLimpeza(idLimpeza integer REFERENCES Limpeza(idLimpeza), nome text REFERENCES MaterialdeLimpeza(nome), quantidade integer, PRIMARY KEY(idLimpeza, nome));

drop table if exists Tratador;

create table Tratador(idTratador integer PRIMARY KEY, nome text, turno text);

drop table if exists Veterinario;

create table Veterinario(idVeterinario integer PRIMARY KEY, idTratamento integer REFERENCES Tratamento(idTratamento), nome text, turno text);

drop table if exists Condicao;

create table Condicao(nome text PRIMARY KEY, idAnimal integer REFERENCES Animal(idAnimal), idTratamento integer REFERENCES Tratamento(idTratamento));

drop table if exists Tratamento;

create table Tratamento(nome text REFERENCES Condicao(nome), idAnimal text REFERENCES Animal(idAnimal), idTratamento integer PRIMARY KEY, nome text);

drop table if exists DataTratamento;

create table DataTratamento(idTratamento integer REFERENCES Tratamento(idTratamento), nome text REFERENCES Condicao(nome), idAnimal integer REFERENCES Animal(idAnimal), data text, PRIMARY KEY(idTratamento, nome, idAnimal));

drop table if exists Medicamento;

create table Medicamento(idMedicamento integer PRIMARY KEY, idTratamento integer REFERENCES Tratamento(idTratamento), nome text, componentes text);

drop table if exists Dosagem;

```
create table Dosagem(dose integer, tempoDeAplicação integer, peso
integer REFERENCES Animal(peso), idMedicamento integer REFERENCES
Medicamento(idMedicamento), idTratamento integer REFERENCES
Tratamento(idTratamento));
```

```
drop table if exists Material;
```

```
create table Material(idTratamento integer REFERENCES
Tratamento(idTratamento), nome text PRIMARY KEY);
```

```
drop table if exists QuantidadeMaterial;
```

```
create table QuantidadeMaterial(idTratamento integer REFERENCES
Tratamento(idTratamento), nome text REFERENCES Material(nome),
quantidade integer, PRIMARY KEY(idTratamento, nome));
```

### **Inserção de dados nas tabelas Sqlite**

```
insert into Animal(idAnimal, Nome, DataDeNascimento, Peso,
NomeEspecie, genero, idJaula, condicao, idTratador)
values(01,'Tobias','12out2004',0.030,'Amphiprion
akallopisos','m',1,'saudavel',1);
```

```
insert into Especie(Nome) values('Amphiprion akallopisos');
```

```
insert into Especie(Nome) values('Papio anubis');
```

```
insert into Genero(genero) values('m');
```

```
insert into Genero(genero) values('f');
```

```
insert into Jaula(idJaula, area, localizacao, tipo, idLimpeza)
values(1,1.5,'Oeste','Aquario',1);
```

```
insert into Jaula(idJaula, area, localizacao, tipo, idLimpeza)
values(2,50,'Este','Terrario',2);
```

```
insert into TipoDeJaula(tipo) values('Aquario');
```

```
insert into TipoDeJaula(tipo) values('Terrario');
```

```
insert into Limpeza(idLimpeza, nome, idTratador,material)
values(1,'Algas',1,'Espatula');
```

```
insert into Limpeza(idLimpeza, nome, idTratador,material)
values(2,'Folhas',2,'Vassoura');
```

```
insert into DataLimpeza(idJaula, idLimpeza,dataLimpeza)
values(1,1,'20out2012');
```

```
insert into MaterialDeLimpeza(material) values('Espatula');
```

```
insert into MaterialDeLimpeza(material) values('Vassoura');
```

```
insert into DataLimpeza(idJaula, idLimpeza,dataLimpeza)
values(2,2,'3nov2012');
```

```
insert into Medicamento(0, idTratamento(0), "flumazenil", "flumazenil");
```

```
insert into Medicamento(1, idTratamento(1), "etorfina", "etorfina");
```

```
insert into QuantMaterialLimpeza(idLimpeza, nome, quantidade)
values(2,'Vassoura',2)
```

```
insert into QuantMaterialLimpeza(idLimpeza, nome, quantidade)
values(1,'Espatula',1)
```

```
insert into Tratador(idTratador, nome, turno) values(1,'Jorge','primeiro');
```

```
insert into Tratador(idTratador, nome,turno)
values(2,'Antonio','primeiro');
```

```
insert into Veterinario(idVeterinario, idTratamento, nome, turno)
values(1,1,'Custodio','primeiro');
```