Especie

- _id: String (final)
- _nome: String (final)

+ getanimais (): Animal[]
+ getid (): String
+ getnome (): String
+ getvacina (): Vacina []
+ getveterenario (): Veterenario []

Vacina

- _id: String (final)
- _nome: String (final)

+ getnome (): String
+ getid (): String

Habitat

- _id: String (final)
- _nome: String (final)
- _area: double

+ populacao (h: Habitat): int
+ adequacao (a: Animal, h: Habitat): int
+ getanimais (): Animal []
+ getarvores (): Arvore []
+ getnome (): String
+ getarea (): double
+ gettratador (): Tratador

Registo

- _veterenario: Veterenario
- _animal: Animal
- _historico: Registo []

+ ordenar (r: Registo): Void
+ getveterenario (): Veterenario
+ getanimal (): Animal
+ gethistorico (): Registo []

EstadoSaude

- _historico: List<String>

+ tamanho (e: Especie): int
+ tamanho nomes (e1: Especie, e2:Especie): int
+ caracteres comuns (e1: Especie, e2: Especie): int
+ dano (e1: Especie, e2: Especie): int

Tratador

- _area: double

---

nivelSatisfacao (t: Tratador): double
populacao (h: Habitat): int
+ getarea (): double
+ trabalho nohabitat (h: Habitat): double
+ numerodetratadoresdohabitat (h:habitat): int

Veterenario

nivelSatisfacao (v: Veterenario): double
populacao (e: Especie): int
+ nveterenarios (e: Especie): int

→ Diogo Pinto , nº 103976

→ Tomás Santos , nº 104 111

Grupo nº 65

TURNO : L08

Caduca

-dificuldadeLimpeza : int

---

esforcosazonal () : double

Perene

-dificuldadeLimpeza : int

---

esforcosazonal () : double

HotelManager

+ gethotel() : Hotel
+ getfilename() : String

<<enumeration>>
Estacao

+ Primavera
+ Veroo
+ Outono
+ Inverno

<<enumeration>>
Influencia

+ Positivo
+ Neutro
+ Negativo

<<enumeration>>
Dano

+ Normal
+ Confusoo
+ Acidente
+ Erro

→ Diogo Pinto, nº 103976

→ Tomás Santos, nº 104111

Grupo : nº 65

TURNO : LO2

Hotel

+ getanimais () : Animal []
+ registaanimal (id1:String, nome:String, id2:string, id3:String):Void
+ transferiranimal (id: string, h: Habitat) : void
+ calculasatisfacao (id : string) : double
+ get funcionarios () : Funcionario []
+ registafuncionario (id:string, nome:String, tipo:Funcionario) :void
+ atribuirresponsabilidade (id1: string, id2:string): void
+ retiraresponsabilidade (id1:String, id2:string) : void
+ gethabitats ()! Habitat []
+ registahabitat (id:string, nome:String, area:double): void
+ alteraarea (id: string, nova : int) : void
+ alterainfluencia(id1: string, id2:String, influencia: Influencia):void
+ planta (id1:String, id2:string, nome:String, idade:int, dificuldade:int, tipo:Arvore): void
+ getarvore (id:String) : Arvore []
+ getvacinas () : Vacina []
+ registavacina (id1:String, nome:String, id2: List<string>):void
+ vacina (id1:string, id2:String, id3:String) : void
+ listavacinacoes () : Registo
+ consultaanimaishabitat (id: String):Animal[]
+ consultavacinacoesanimal (id:string): Registo
+ consultaatosveterenario (id:string) : Registo
+ consultavacinaserradas () : Estadosaude
+ avancaestacao () : int
+ calculasatisfacaoanimais () : double
+ calcula satisfacao funcionarios () : double
+ calcula satisfacaototal () : double

→ Diogo Pinto, nº 103970   Grupo : nº 65

→ Tomás santos, nº 104111   TURNO : L08

→ Diogo Pinto, nº 103976

→ Tomás Santos, nº 104111

Grupo: nº 65

Turma: L08

Registo

Registo de administração

Vacina do registo

Vacina

Especie Obloio

Vacina para a especie

Especie ANIMAL

Especie *

Animais Pertencentes *

Animal *

Animais presentes *

EstadoSaude

Dano

Vacina

Vacine aplicada

Habitat Animal

Habitat *

Habitats atribuidos *

Tratador do Habitat

Influência

Habitat arvore

Hotel *

Estação anal

Estacao

Estação Nascimento arvore

Arvore *

Arvores Pertencentes *

Perene

Caduca

Tratador

Funcionario

Veterinario

Especie *

Hotel Manager