

Problema06

Alta Animal

```
//////////
// Alta Animal
//////////
context:
  Sistema::AltaPersona(nomPersona : String, nomPoblacio : String) : Persona
pre:
  // Res
post:
  // S'hagi creat i no sigui "Veterinari"
  Persona.allInstances()->exists(p |
    // Assegurar-se dels paràmetres
    p.nom = nomPersona AND p.poblacio = nomPoblacio
    // Assegurar-se que sigui nou
    AND p.oclIsNew() AND result = p
    // Assegurar-se que sigui "Persona" i no "Veterinari"
    AND p.oclIsTypeOf(Persona)
  )

context:
  Sistema::AltaAnimal(
    persona_i : Persona,
    nomEspecie : String,
    nomAnimal : String, sexeAnimal : TSexe, esAgressiu : Boolean) : Animal
pre:
  // La especie ha d'existir (PQ. multiplicitat 1..1 i és obligatori)
  Especie.allInstances()->exists(e | e.nom = nomEspecie)
  // La persona existeix perquè l'hem fet abans
post:
  Animal.allInstances()->exists(a |
    // Assegurar-se dels paràmetres
    a.nom = nomAnimal AND a.sexe = sexeAnimal
    // Assegurar-se de les relacions
    AND a.persona.nom = persona_i.nom AND a.especie.nom = nomEspecie
    // Assegurar-se que sigui nou
    AND a.oclIsNew() AND result = a
    // Assegurar-se que si es "Gos" tingui l'agressivitat
    AND (a.especie.nom = "Gos") implies(a.oclIsTypeOf(Gos) AND a.agressiu = esAgressiu)
  )
```

Alta Operació Urgent

```
//////////
// Alta Operació Urgent
//////////
context:
  Sistema::AltaOperacio(
    veterinariEncarregat : Veterinari,
    animalOperat : Animal,
    dataOperacio : Data,
    horaIniciOperacio : Time, motiuOperacio : String) : Operació

pre:
  // 1. Veterinari Encarregat ha de ser especialista
  veterinariEncarregat.especie.nom = animalOperat.especie.nom
  // 2. Veterinari Encarregat no pot haver fet més de 2 aquell dia
  AND Operació.allInstances()->select(o |
    o.veterinari.nom = veterinariEncarregat.nom
    AND o.data.data = dataOperacio
  ) < 2

post:
  Operació.allInstances()->exists(o |
    // Assegurar-se dels paràmetres
    o.horaInici = horaIniciOperacio
    AND o.motiu = motiuOperacio
    // Assegurar-se de les relacions
    AND o.veterinari.nom = veterinariEncarregat.nom
    AND o.animal.nom = animalOperat
    AND o.data.data = dataOperacio
    // Assegurar-se que sigui nou
    AND o.oclIsNew() AND result = o
  )

context:
  Sistema::AltaOperacioUrgent(
    operacioRealitzada : Operació,
    nomMedicament : String
    esExitosa : Boolean, horaFiOperacio : Time
  )

pre:
  // Assegurar-se que Finalitzada (Aixo realment no se on ficar-ho)
  operacioRealitzada.oclIsTypeOf(OperacióFinalitzada)
  // El medicament ha d'exsitir (Ho fico perquè 1..* --> Obligatori?)
  AND Medicament.allInstances()->exists(m | m.nom = nomMedicament)
  // Comprovar interval
  AND horaFiOperacio > operacioRealitzada.hora

post:
  Operacio.allInstances()->exists(o |
    // Assgurar-se dels paràmetres
    o.exitosa = esExitosa AND o.horaFi = horaFiOperacio
    // Assegurar-se de les relacions
    o.medicament.nom = nomMedicament
    // Assegurar-se que sigui nou
    o.oclIsNew() AND result = o
  )
```

Consulta Gossos Perillosos no Esterilitzats

```
////////////////////////////////////
// Consulta Gossos Perillosos no Esterilitzats
////////////////////////////////////
context:
  Sistema::obteLlistaGossosPerillNoEst(nomPoblacio : String) : Set (
    | TupleType(nomGos : String, nomPropietari : String, esVeterinari))

pre:
  // Visqui min. 1 persona i sigui propietaria de min. 1 animal
  Persona.allInstances()->select(p |
    | p.poblacio = nomPoblacio
    | AND p.animal->Num() >= 1
  ) >=1

body:
  result = Animal.allInstances()->select(a |
    | // És Gos + Mascle + Agressiu
    | a.oclIsTypeOf(Gos) AND a.sexe = TSexe::mascle AND a.agressiu = true
    | // Viu a la població
    | AND a.persona->select(p | p.poblacio = nomPoblacio)
    | // NO SÉ COM FER PER SABER Operacions
    | // - [ ] Gos no esterilitzat
    | // - [ ] Gos no té esterilització programada
  ) -> collect(a2 |
    | tuple {
    |   nomGos: a2.nom,
    |   nomPropietari: a2.persona.nom,
    |   esVeterinari: // NO sé com fer-ho
    | }
  )
)
```