



## Alta de Persona amb Poders:

```
1 context: Sistema::AltaPersona(nomPe:String, paisPe:String, esH:bool, esM:bool):Persona
2
3 pre: Malvat.allInstances() -> select(
4   m | m.poderdepersona.oclIsTypeOf(Innat) -> isEmpty()
5   ).size() >= 10
6
7 post: Persona.allInstances() -> exists(
8   p | p.nom = nomPe and p.pais = paisPe and p.oclIsNew()
9   and if (esH) then p.oclIsTypeOf(Heroi) endif
10  and if (esM) then p.oclIsTypeOf(Malvat) endif
11  and result = p
12  )
13
14
15 context: Sistema::TePoder(personaPod:Persona, nomPo:String, descPo:String, esInnat:bool, nomLloc:String)
16
17 pre:
18
19 post:
20   personaPod.poder.nom -> includes(nomPo)
21   and if not Poder.allInstances()@pre -> exists (po | po.nom = nomPo) then
22     Poder.allInstances() -> exists(po2 |
23       po2.oclIsNew() and
24       po2.nom = nomPo and
25       po2.descripció = descPo)
26     endif
27   and PoderDePersona.allInstances() -> exists(pdp |
28     pdp.oclIsNew() and
29     pdp.poderosa.nom = personaPod.nom and
30     pdp.poder.nom = nomPo and
31     pdp.poder.descripció = descPo
32     if esInnat then
33       pdp.oclIsTypeOf(Innat)
34     else
35       pdp.oclIsTypeOf(Adquirir) and
36       pdp.oclAsType(Adquirir).lloc.nom = nomLloc
37     endif
38   )
```

## Alta de Crisi d'un Malvat

```
1 context: Sistema::EscullMalvat(nomM:String):Malvat
2
3 pre: Malvat.allInstances() -> exists(m | m.nom = nomM)
4
5 post: Malvat.allInstances() -> select(m | m.nom = nomM and result = m);
6
7
8 context: Sistema::AltaCrisi(organitzador:Malvat, dataI:Date, llocC:String, dataF:Date)
9
10 pre:
11
12 post: Crisi.allInstances() -> exists (
13   c | c.data_fi = dataF and c.inici.data = dataI and c.lloc.nom = llocC and
14   c.malvat.nom = organitzador.nom and c.oclIsNew()
15 )
```

## Consulta de Llocs Perillosos

```
1 context: Sistema::ConsultaLlocsPerillosos(dataI:Date, dataF:Date)
2   :Set(
3     tupletype(
4       nomLloc:String,
5       malvats:Set(String)
6     )
7   )
8
9 pre: Persona.allInstances() -> exists(p |
10   p.oclIsTypeOf(Malvat) and
11   p.oclIsTypeOf(Heroi) and
12   p.poderdepersona -> exists (pp |
13     pp.oclIsTypeOf(Adquirir) and
14     pp.oclAsType(Adquirir).lloc.nom = "Barcelona"
15   )
16 )
17
18 body:
19   let llistaLlocs: Set(Lloc) = Lloc.allInstances()->select(
20     l | l.crisi -> size() > 5 and l.inici.data >= dataI and l.crisi.data_fi <= dataF and l.crisi.heroi -> size() < 3
21   )
22   in result = llistaLlocs -> collect (
23     ll | tuple {
24       nomLloc = ll.nom,
25       malvats = ll.adquirir.poderosa.oclAsType(Malvat).nom;
26     }
27   )
28
```