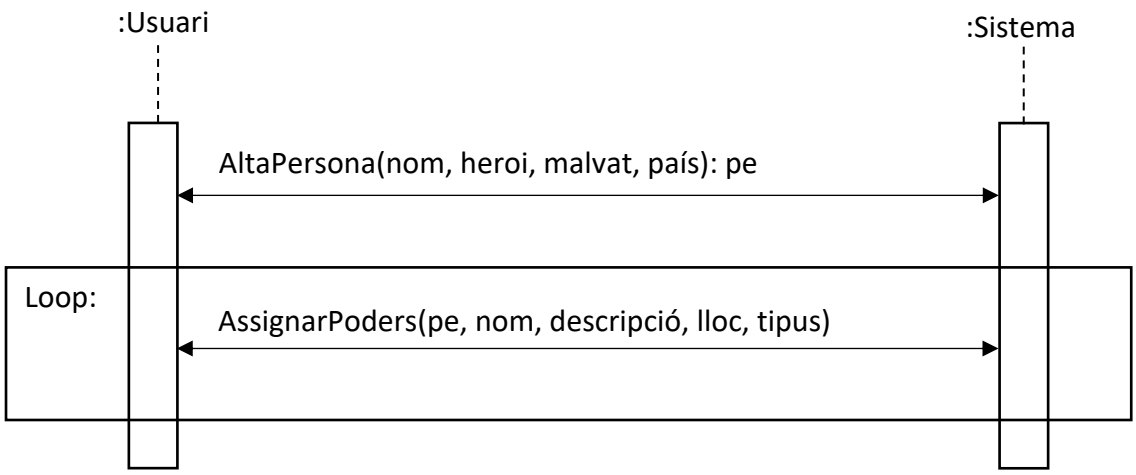
Andreu Orensanz

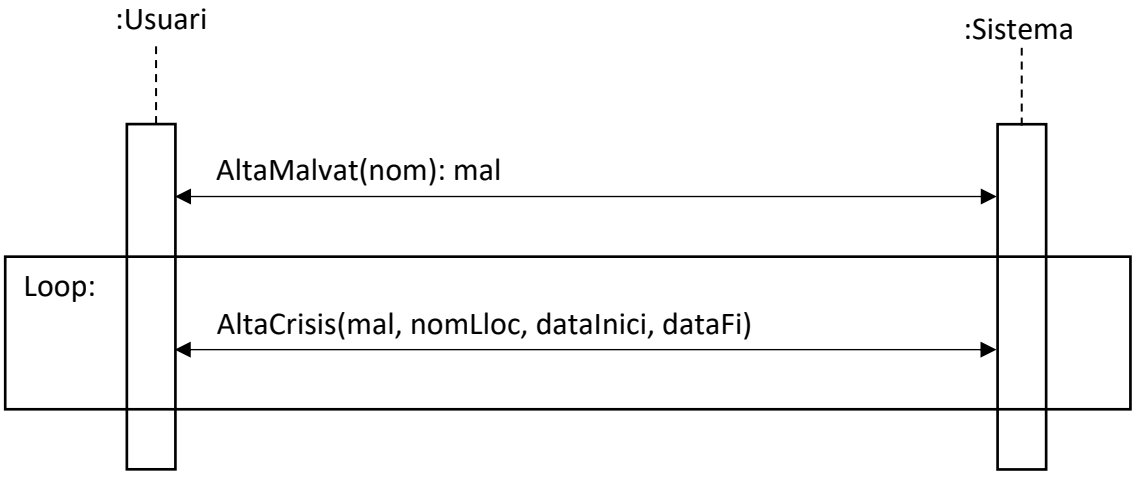
2on control IES QT2122 – Model del Comportament

2.



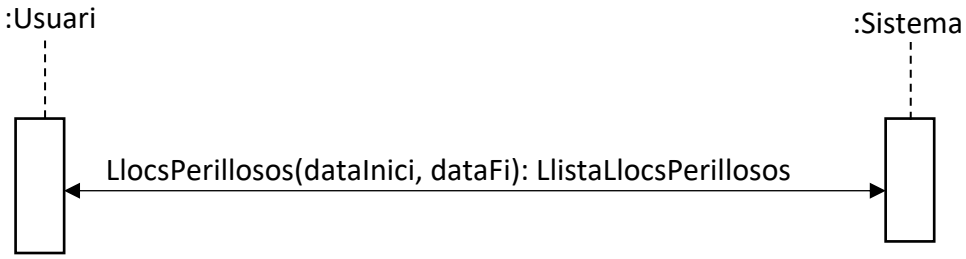
Alta de Persona amb Poders

```
context Sistema::AltaPersona(nom: String, heroi: Boolean, malvat: Boolean, pais: String): pe
pre: Malvat.allInstances() -> select(m | m.PoderDePersona -> excludesAll(m.PoderDePersona.oclIsTypeOf(Innat))) -> size >= 10
post: Persona.allInstances() -> Exists(p | p.oclIsNew() and
                                           p.nom = nom and
                                           if heroi then p.oclIsTypeOf(Heroi) endif and
                                           if malvat then p.oclIsTypeOf(Malvat) endif and
                                           p.pais = pais and
                                           result = p)

context Sistema::AssignarPoders(pe: Persona, nom: String, descripcio: String, lloc: String, tipus: String)
pre:
post: if not Poder.allInstances()@pre -> Exists(pod | pod.nom = nom) then
      Poder.allInstances() -> Exists(pod | pod.oclIsNew() and
                                           pod.nom = nom and
                                           pod.descripcio = descripcio) endif and
      PoderDePersona.allInstances() -> Exists(pdp | pdp.oclIsNew() and
                                                    pdp.Persona = pe and
                                                    pdp.Poder.nom = nom and
                                                    if tipus = 'Adquirit' then pdp.oclIsTypeOf(Adquirit) and
                                                                               pdp.oclAsType(Adquirit).Lloc.nom = lloc
                                                    else pdp.oclIsTypeOf(Innat))
```



Alta de Crisis d'un Malvat

```
context Sistema::AfegeixMalvat(nom: String):mal
pre: Malvat.allInstances() -> Exists(m | m.nom = nom)
post: Malvat.allInstances() -> Exists(m | m.nom = nom and result = m)  -- No cal crear Malvat ja que hem donat
                                                                       -- d'alta Persona a l'anterior funcionalitat

context Sistema::AltaCrisis(mal: Malvat, nomLloc: String, dataInici: Data, dataFi: Data)
pre: -- No cal verificar que el lloc existeix perquè pot no existir-ne cap segons l'UML
post: Crisi.allInstances() -> Exists(c | c.oclIsNew() and
                                         c.Malvat = mal and
                                         c.lloc.nom = nomLloc and
                                         c.inici.data = dataInici and
                                         c.data_fi = dataFi)
```



Llocs Perillosos

```
context Sistema::LlocsPerillosos(dataInici: Date, dataFi: Date): Set(TupleType(nomLloc: String,
                                                                               nomMalvat: Set(String))
pre: Persona.allInstances() -> Exists(per | per.oclIsTypeOf(Heroi) and
                                            per.oclIsTypeOf(Malvat) and
                                            per.PoderDePersona.oclIsTypeOf(Adquirit) and
                                            per.Lloc -> includes('Barcelona'))
body: let llocs: Set(Lloc) = Lloc.allInstances() -> select(ll | ll.Crisi -> size() > 5 and
                                                                ll.inici.data >= dataInici and
                                                                ll.Crisi.data_fi <= dataFi and
                                                                ll.Crisi.Heroi -> size() < 3)
      in result = llocs -> collect(ll | Tuple {
                                          nomLloc = ll.nom
                                          nomMalvat = ll.Adquirit.ensenyant -> select(e | e.oclIsTypeOf(Malvat).nom) -> asSet()
                                        })
```