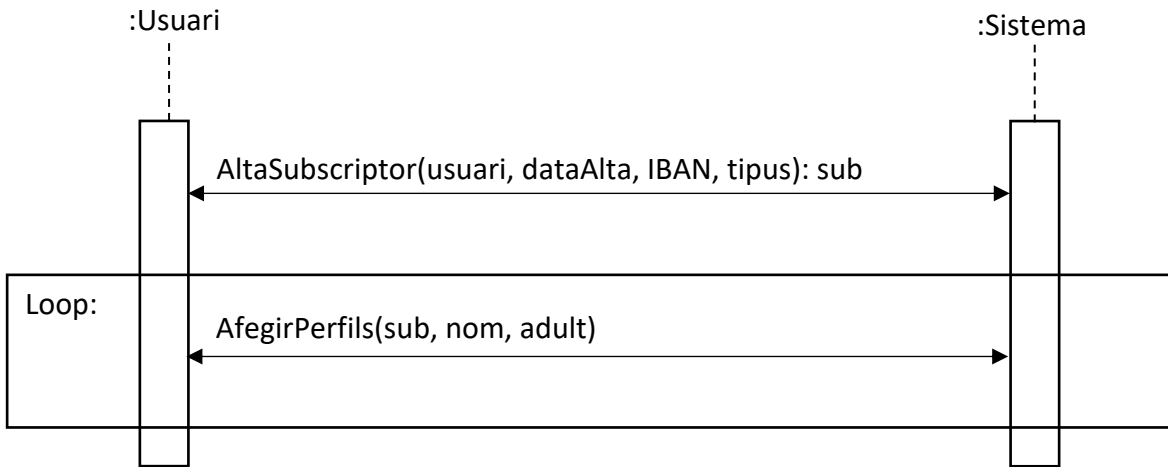


## 2on control IES QT2122 – Model del Comportament

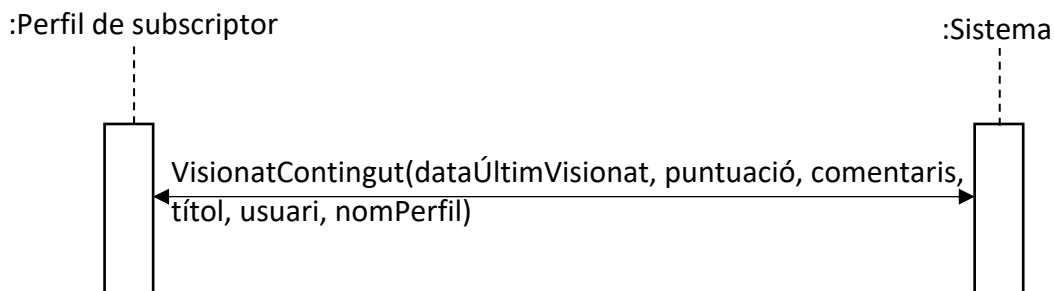
3.



Alta de Subscriptor

```
context Sistema::AltaSubscriptor(usuari: String, dataAlta: Date, IBAN: String, tipus: String):subs
pre:
post: Subscriptor.allInstances() -> Exists(s | s.ocIsNew() and
|                                     s.usuari = usuari and
|                                     s.dataAlta = dataAlta and
|                                     s.IBAN = IBAN and
|                                     if tipus = 'premium' then s.ocIsTypeOf(SubscriptorPremium)
|                                     endif and result = s)

context Sistema::AfegirPerfil(subs: Subscriptor, nom: String, adult: Boolean)
pre:
post: Perfil.allInstances() -> Exists(p | p.ocIsNew() and
|                                     p.nom = nom and
|                                     p.adult = adult and
|                                     p.Subscriptor = sub)
```

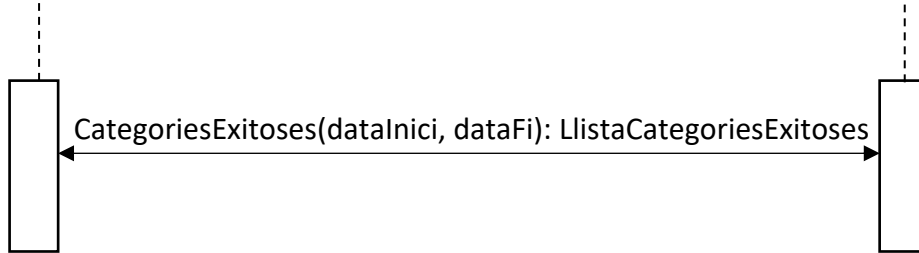


## Visionat de Contingut

```
context Sistema::VisionatContingut(dataUltimVisionat: Date, puntuacio: TPuntuacio, comentaris: String, titol: String, usuari: String, nomPerfil: String)
pre: Contingut.allInstances() -> Exists(c | c.titol = titol) and
    Perfil.allInstances() -> Exists(p | p.nom = nomPerfil and
        p.Subscriptor.usuari = usuari)
post: if not Perfil.allInstances()@pre -> Exists(p | p.Contingut.titol = titol and --No existeix i ho creem
    p.nom = nomPerfil and
    p.Subscriptor.usuari = usuari)
    then Perfil.allInstances() -> Exists(p | p.ocllsNew() and
        p.ContingutVisionat.dataUltimVisionat = dataUltimVisionat and
        p.ContingutVisionat.puntuacio = puntuacio and
        p.ContingutVisionat.comentaris = comentaris and
        p.Contingut.titol = titol and
        p.Subscriptor.usuari = usuari and
        p.nom = nomPerfil)
    else Perfil.allInstances() -> Exists(p | p.ContingutVisionat.dataUltimVisionat = dataUltimVisionat and
        p.ContingutVisionat.puntuacio = puntuacio and
        p.ContingutVisionat.comentaris = comentaris and
        p.Contingut.titol = titol and
        p.Subscriptor.usuari = usuari and
        p.nom = nomPerfil) endif
```

:Administrador

:Sistema



## Categories Exitoses

```
context Sistema::CategoriesExitoses(dataInici: Date, dataFi: Date): Set(TupleType(titol: String,
    nomsCategories: Set(String),
    preu: Float))
pre: ContingutDescarregat.allInstances() -> select(c | c.Data.date >= dataInici and
    c.Data.date <= dataFi) -> size() >= 3
body: let continguts: Set(ContingutDescarregat) = ContingutDescarregat.allInstances() -> select(c | c.Data.date >= dataInici and
    c.Data.date <= dataFi and
    c -> size() > 5)
    in result = continguts -> collect(cd | Tuple {
        titol = cd.ContingutPremium.Contingut.titol
        nomsCategories = cd.ContingutPremium.Categoria.nom -> asSet()
        preu = cd.ContingutPremium.preu * cd -> size()
    })
```