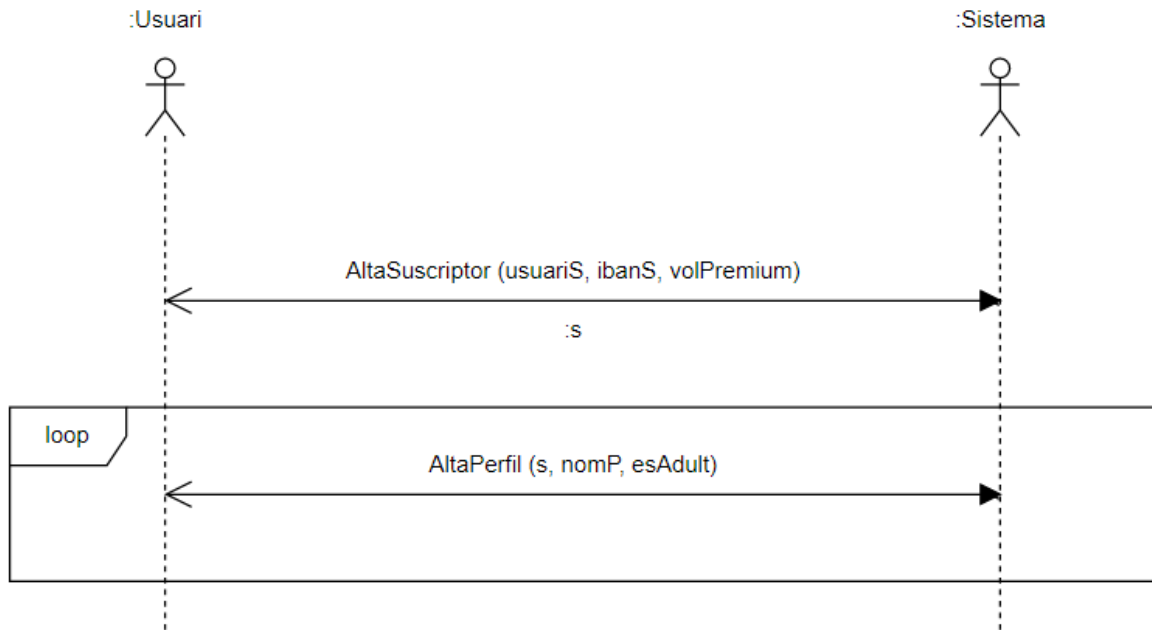


PROBLEMA 3 CONTROL 2:

->Alta de Subscriptor:



context: Sistema::AltaSubscriptor (usuariS : String, ibanS : String,
volPremium : Boolean):Subscriptor

pre:

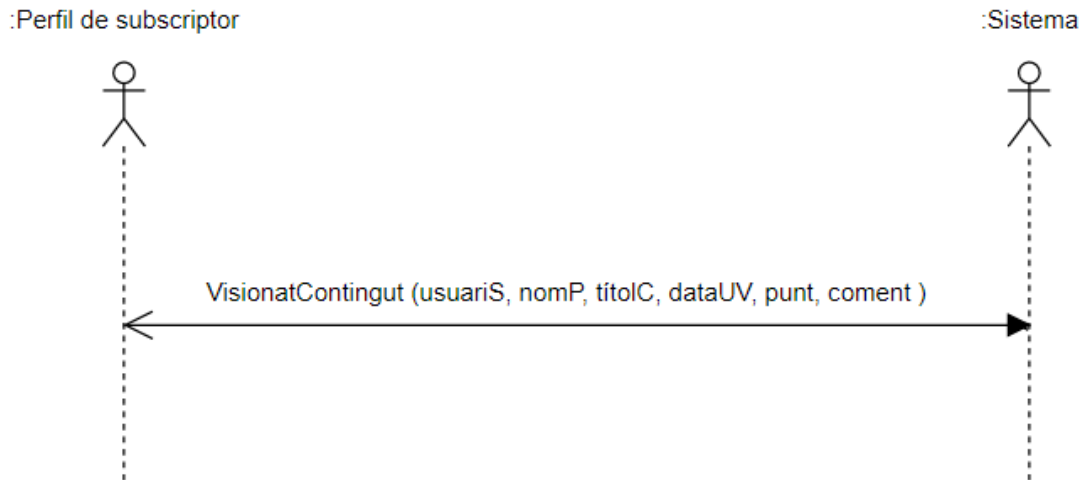
post: Subscriptor.allInstances()->exists(s | s.ocllsNew() and s.usuari = usuariS and
s.iban = ibanS and s.dataAlta = today()) and
if (volPremium)
then s.ocllsTypeOf(SubscriptorPremium)
endif
and result = s)

context: Sistema::AltaPerfil (s : Subscriptor, nomP : String, esAdult : Boolean)

pre:

post: Perfil.allInstances()->exists(p | p.ocllsNew() and p.nom = nomP and p.adult = esAdult
and p.subscriptor = s)

->Visionat de Contingut:



context: Sistema::VisionatContingut (usuariS : String, nomP : String, titolC: String,
dataUV : Date, punt : TPuntuació, coment : String)

pre: Subscriptor.allInstances()->exists(s | s.usuari = usuariS and s.perfil.nom = nomP) and
Contingut.allInstances()->exists(c | c.titol = titolC)

post: if (not (ContingutVisionat.allInstances())@pre->exists(cv | cv.contingut.titol = titolC and
cv.perfil.nom = nomP and cv.perfil.subscriptor.usuari = usuariS)))
then ContingutVisionat.allInstances()->exists(cv | cv.ocllsNew() and
cv.dataÚltimVisionat = dataUV and cv.puntuació = punt and cv.comentaris = coment
and cv.perfil.nom = nomP and cv.perfil.subscriptor.usuari = usuariS and
cv.contingut.titol = titolC)
else ContingutVisionat.allInstances()->exists(cv | cv.dataÚltimVisionat = dataUV and
cv.puntuació = punt and cv.comentaris = coment and cv.perfil.nom = nomP and
cv.perfil.subscriptor.usuari = usuariS and cv.contingut.titol = titolC)

->Categories Exitoses:



context: Sistema::CategoriesExistoses (dataIni : Date, dataFi : Date):

Set(TupleType(títolC : String, nomCat : Set(String), preuD : Float))

pre: ContingutDescarregat.allInstances()->select(cd | cd.data.date >= dataIni and
cd.data.date <= dataFi)->size() > 2

body: let cont : Set(ContingutPremium) =

ContingutPremium.allInstances()->select(cp | cp.contingutDescarregat->

select (cd | cd.data.date >= dataIni and cd.data.date <= dataFi)->size() > 5)

in

result = cont->collect(cp | Tuple {títolC = cp.títol, nomCat = cp.categoria.nom,

preuD = cp.preu*cp.contingutDescarregat->size()})