7è problema d'IES-QT2122

Alta d'Inscripció

: Participant → :Sistema

AltaInscripció(dni, nomTaller, preu)
: Inscripció

Alta de Reserva

: Organitzador → : Sistema

AltaReserva(CodiSala, data, hora, nomTaller)
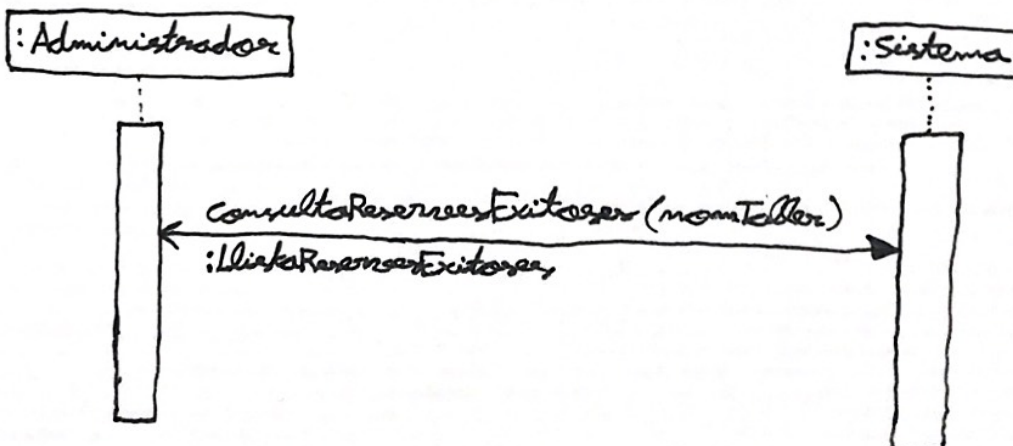: Res

ReqRecurs (Res, CodiRecurs)
: RecursReservas

Consulta Reserves Exitoses

:Administrador → :Sistema
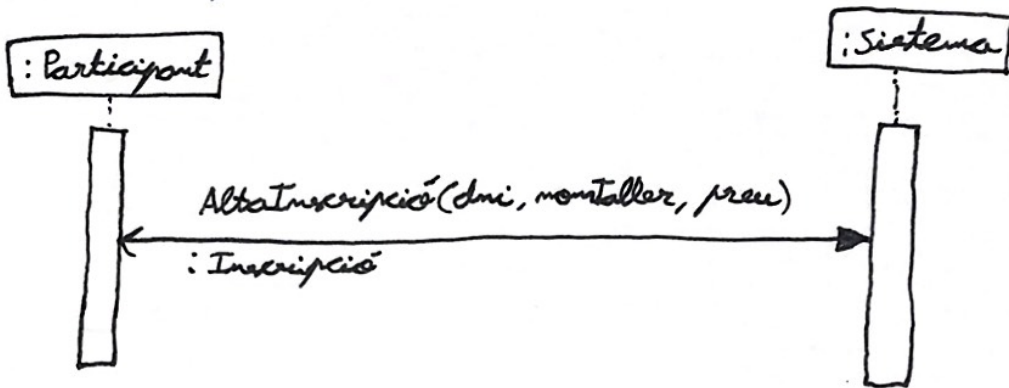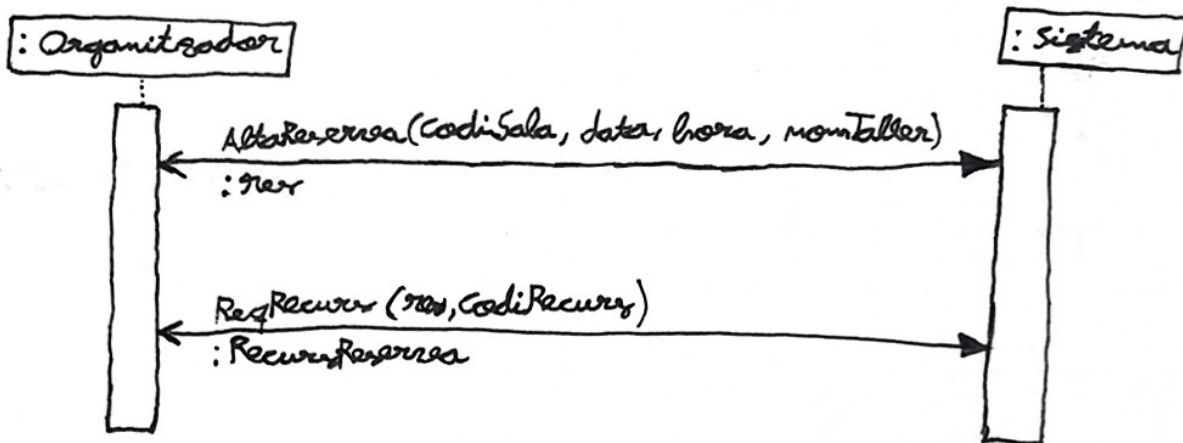
ConsultaReservesExitoses (nomTaller)
:LlistaReservesExitoses

```
1    --Alta d'Inscripció
2    context Sistema::AltaInscripcio(dni: String, nomTaller: String, price: TPreu):Inscripcio
3    pre: Participant.allInstances() -> (p | p.dni = dni) and
4    |    Taller.allInstances() -> (t | t.nom = nomTaller)
5    post: Inscripcio.allInstances() -> Exists(i | i.oclIsNew() and
6                                                 i.Participant.dni = dni and
7                                                 i.Taller.nom = nomTaller and
8                                                 if i.tipusIns == InscripcioDePagament then
9                                                 i.avaluacioPreu = price and i.pagada = false endif
10                                                and result = i)

12   --Alta de Reserva
13   context Sistema::AltaReserva(codiSala: String, data: Date, hora: Time, nomTaller: String):res
14   pre: Sala.allInstances() -> Exists(s | s.codi = codiSala)
15   |    DataHora.allInstances() -> Exists(dh | dh.data = data and dh.hora = hora)
16   |    Taller.allInstances() -> Exists(t | t.nom = nomTaller)
17   post: Reserva.allInstances() -> Exists(r | r.oclIsNew() and
18                                             r.Sala.codi = codiSala and
19                                             r.Taller.nom = nomTaller and
20                                             r.DataHora.data = data and
21                                             r.DataHora.hora = hora and
22                                             result = r)
23
24   context Sistema::ReqRecurs(res: Reserva, codiRecurs: String):RecursReserva
25   pre: Reserva.allInstances() -> Exists(r | r = res)
26   |    Recurs.allInstances() -> Exists(rec | rec.codi = codiRecurs)
27   post: Reserva.allInstances() -> Exists(r | r.oclIsNew() and
28                                             r.Recurs.codi -> includes(codiRecurs) and
29                                             r = RecursReserva and
30                                             result = r)
31

32   --Consulta Reserves Exitoses
33   context Sistema::consultaReservesExitoses(nomTaller: String):
34   |    Set(tupletype(codiSala: String, data: Date, hora: Time, correus: String))LlistaReservesExitoses
35   pre: Taller.allInstances() -> Exists(t | t.nom = nomTaller) and
36   |    Inscripcio.allInstances() -> select(i | if i.tipusIns = "InscripcioDePagament" then
37                                             i.oclAsType(InscripcioDePagament).pagada = true and
38                                             i.oclAsType(InscripcioDePagament).avaluacioPreu = TPreu::barat endif) -> size() > 3
39   body: let resultat: Set(Reserva) = Reserva.allInstances() -> select(r | r.Taller.nom = nomTaller and
40                                                             r.Valoracio -> select(v | v.puntuacio = TPuntuacio::5))
41                                                                           -> size() > 5
42      in result = resultat -> collect(r | Tuple {
43                                             codiSala = r.Sala.codi,
44                                             data = r.DataHora.data,
45                                             hora = r.DataHora.hora,
46                                             correus = r.Taller.Participant.correuE
47                                             })
```