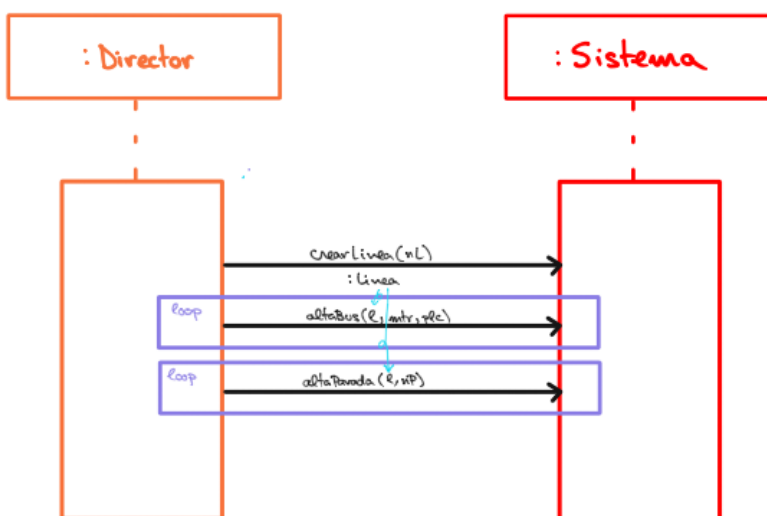




Exercici 4 - TMB

Crear Linia

Quan el director de TMB decideix crear una nova línia d'autobusos li comunica a un empleat d'administració que introdueixi al sistema la nova línia amb totes les dades necessàries. Feu que la interacció necessària per a portar a terme aquesta funcionalitat requereixi més d'un esdeveniment.



```
Context: Sistema::crearLinha (nL: Integer) : Linia
pre:

post:  Linia.allInstances() -> exists ( l |
      l.ocIsNew()
      and l.numero = nL
      and result = l )

Context: Sistema::altaBus (l: Linia, mtr: String, plc: Integer) --LOOP
pre:

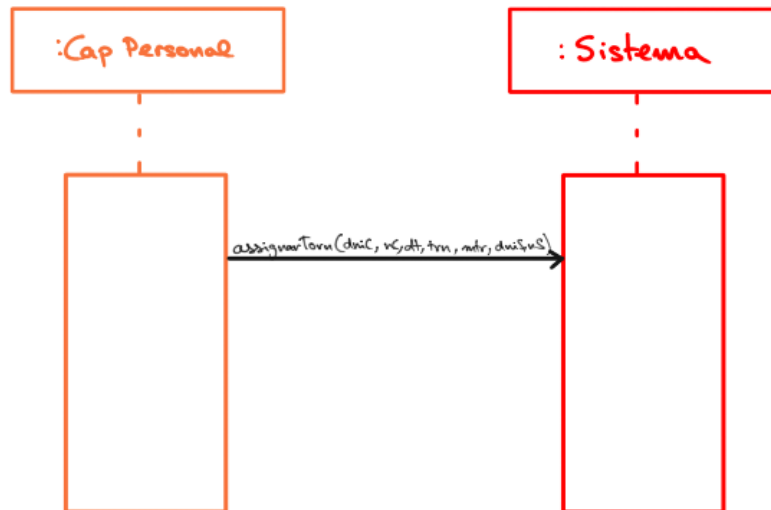
post:  Bus.allInstances() -> exists ( b |
      b.matricula = mtr
      and b.places = plc
      and b.linia = l )

Context: Sistema::altaParada (l: Linia, nP: String) --LOOP
pre:

post:  Parada.allInstances() -> exists ( p |
      p.ocIsNew()
      and p.nom = nP
      and p.linia = l )
```

Assignar Torns

Quan el cap de personal vol assignar un conductor a un torn comunica ell mateix al sistema totes les dades necessàries. Cal tenir en compte que el conductor que s'assigna al torn treballat ha d'existir al sistema, però el conductor suplent pot ser creat en aquest mateix moment, amb tota la informació que requereix.

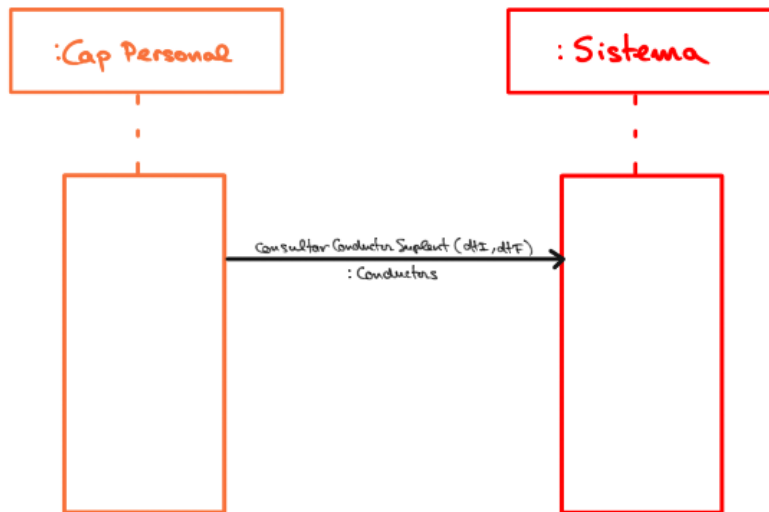


```
Context: Sistema::assignarTorn ( dniC: String, nC: String, dt: Data, trn: TipusTorn, mtr: String, dniS: String, nS: String )
pre:      Conductor.allInstances() -> exists ( c | c.dni = dniC )
        and Torn.allInstances() -> exists ( t | t.torn = trn )

post:     TornTreballat.allInstances() -> exists ( t |
        t.ocIsNew()
        and t.conductor.dni = dniC
        and t.conductor.nom = nC
        and t.data.data = dt
        and t.torn.torn = trn
        and t.bus.matricula = mtr
        and if ( t.torn.torn = TipusTorn::nit ) then
            if (Conductor.allInstances()@pre -> exists ( c | c.dni = dniS ) ) then
                t.AsTypeOf(TornNocturn).suplent.dni = dniS
                and t.AsTypeOf(TornNocturn).suplent.nom = nS
            else
                Conductor.allInstances() -> exists ( c |
                    c.ocIsNew()
                    and c.nom = nS
                    and c.dni = dniS
                    and t.AsTypeOf(TornNocturn).suplent = c)
            endif
        endif
    endif
```

Consultar Conductors Suplents

Finalment, quan el cap de personal vol obtenir un llistat dels conductors que han estat suplents en un cert període ho demana ell mateix al sistema. El llistat mostra, per cada conductor que ha estat suplent més d'una vegada dins el període, el seu dni i el número de totes les línies a les que està assignat. Aquest llistat només es pot demanar si existeix algun conductor que hagi estat suplent però no hagi estat assignat a cap torn treballat.



```
Context: Sistema::consultarConductorSuplent (dtI: data, dtF: data) : conductors : Set ( TupleType ( dniC: String, l: Set(Linia)))
pre: TornTreballat.allInstances() -> exists ( t |
    t.AsTypeOf(TornNocturn).suplent -> forAll(TornTreballat.conductor != t))

body: let cond : Set(Conductor) = TornNocturn.allInstances() -> select ( t |
    t.data.data <= dtF
    and t.data.data >= dtI
    and t.suplent.TornNocturn -> size() > 1 )

    in result = cond -> collect ( r |
        Tuple { dniC = r.suplent.dni,
                l = r.suplent.linia})
```



Ignasi Fibla Figuerola < ignasi.fibla@estudiantat.upc.edu >

Gerard Madrid Miró < gerard.madrid@estudiantat.upc.edu >