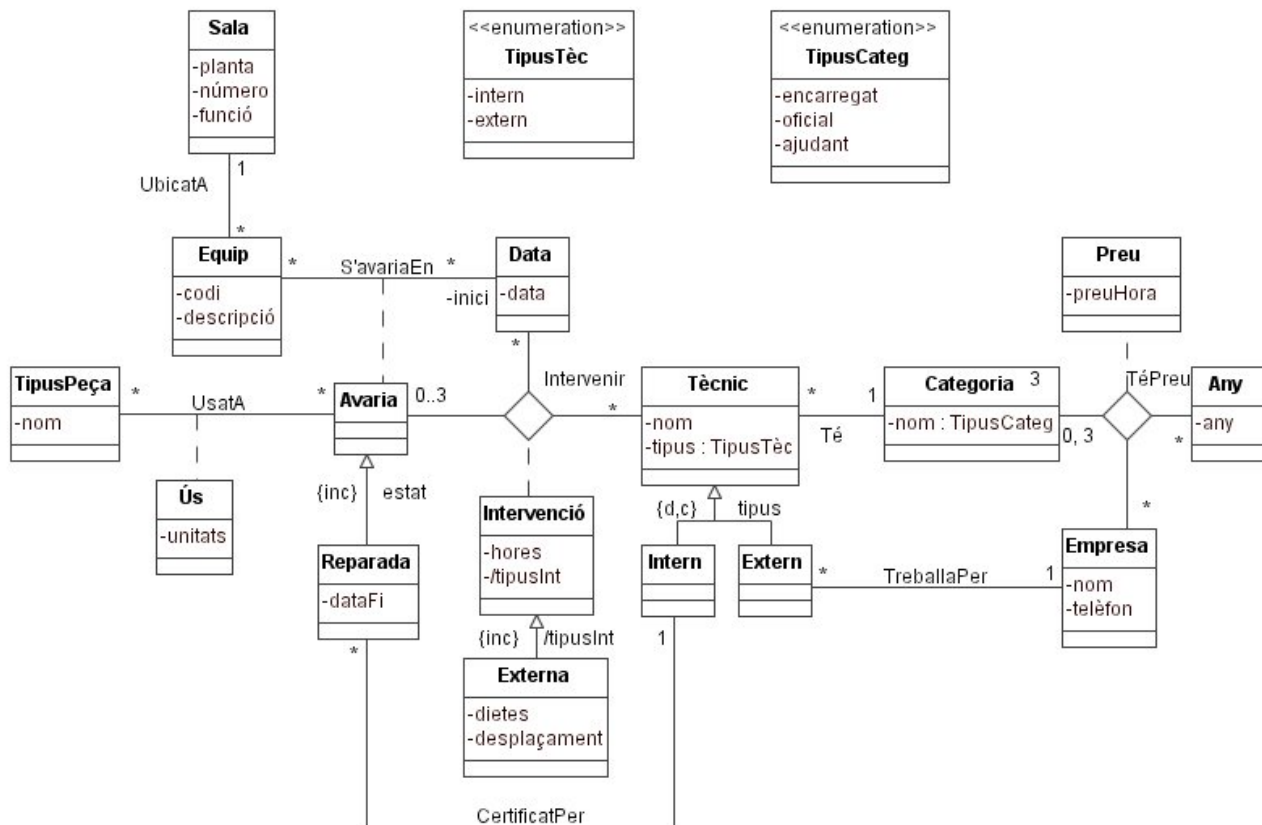


1r Problema – Model Conceptual de les Dades (3,5 punts)



Restriccions textuais:

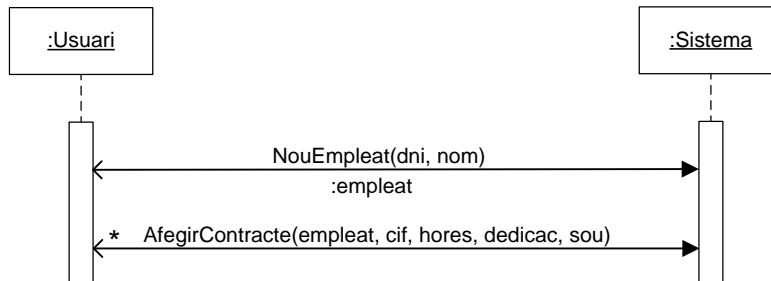
- 1- Claus externes: (Equip: codi), (Sala: planta, número), (Tècnic: nom), (Categoria: nom), (Empresa: nom), (Any: any), (Data: data), (TipusPeça: nom).
- 2- La data de fi d'una avaria reparada és posterior a la seva data d'inici.
- 3- Un equip no pot tenir dues avaries que se solapin temporalment (opcional).
- 4- La data d'una intervenció ha d'estar entre la data d'inici i la data de fi de la seva avaria.
- 5- La suma d'hores de les intervencions d'un tècnic en una data no pot ser superior a 8.
- 6- Tots els tècnics externs que intervenen en una mateixa avaria han de pertànyer a una única empresa subcontractada.

Regla de derivació:

- 1- El tipusInt d'una intervenció és 'Externa' si la intervenció correspon a un tècnic de tipus extern.

2on Problema – Model del Comportament (3,5 punts)

Diagrama de seqüència: AltaEmpleatSenseFormació



Contractes de les operacions

Operació NouEmpleat(dni:String, nom:String): Empleat

Semàntica: Es crea una nova instància d'empleat sense formació amb els paràmetres que se li passen.

Pre:

Post: emp.ocllsNew() and emp.ocllsTypeOf(Empleat) and emp.dni=dni and emp.nom=nom

Sortida: emp

Operació AfegirContracte(empleat:Empleat, cif:String, hores:Integer, dedicac:TipDedicació, sou:Integer)

Semàntica: Es crea una nova instància de EnPràctiques entre el'empleat i l'empresa.

Pre:

Empresa.allInstances()->exists(e| e.cif=cif) and

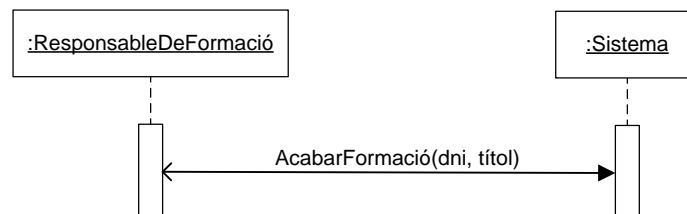
not (Contracte.allInstances() -> exists(c| c.empleat = empleat and c.empresa.cif = cif)

Post:

ep.ocllsNew()and ep.ocllsTypeOf(EnPràctiques) and ep.empleat=empleat and ep.empresa.cif=cif and

ep.hores-set=hores and ep.dedicació=dedicac and ep.sou=sou

Diagrama de seqüència: AcabarFormació



Contracte de l'operació

Operació AcabarFormació(dni:String, títol:String)

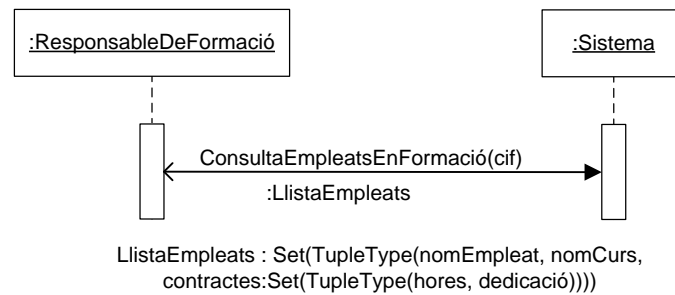
Semàntica: La instància de l'empleat en-formació amb dni que se li passa com a paràmetre, passa a ser instància d'empleat format amb el títol que se li passa com a paràmetre i tot els seus contractes passen a ser indefinits

Pre:

En-Formació.allInstances()->exists (e| e.dni = dni)

Post: Empleat.allInstances()->exists(e|e.dni=dni and e.ocllsTypeOf(Format) and not (e.ocllsTypeOf(En-Formació)) and e.ocllsType(Format).títol=títol and e.contracte->forAll(c| c.ocllsTypeOf(Indefinit) and not (c.ocllsTypeOf(EnPràctiques))))

3. ConsultaEmpleatsEnFormació:



Operació LlistaEmpleats(cif:String):

Set(TupleType(nomEmpleat:String, nomCurs:String, contractes:Set(TupleType(hores, dedicació))))

Semàntica: Llista els empleats en formació d'una empresa

Pre:

Empresa.allInstances()->exists(e| e.cif=cif)

Post:

Sortida:

let emplEnForm:Set(En-formació) =

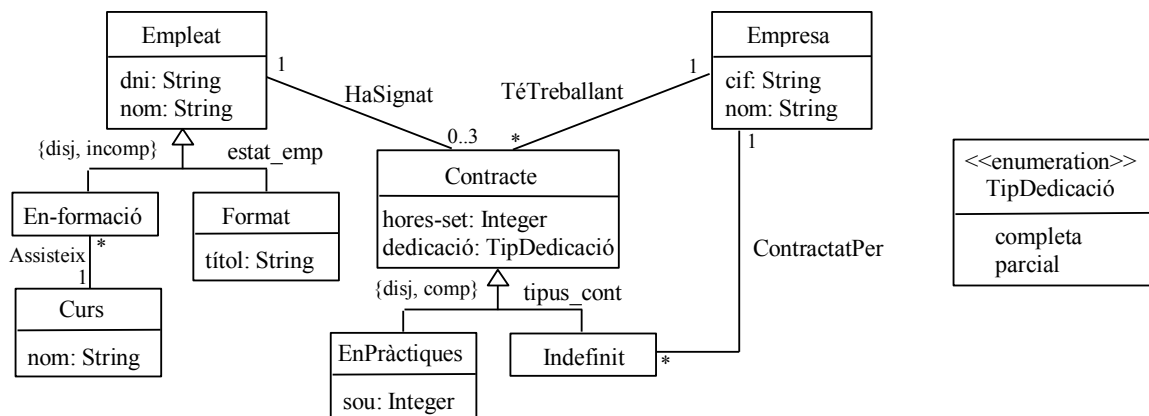
En-formació.allInstances()->

select (ef| ef.empresa.cif ->includes(cif) and ef.contractes.oclAsType(EnPràctiques)->

select(pr | pr.sou>1000)->size())>2)) in

result = emplEnForm->collect(e| Tuple{nomEmpleat=e.nom, nomCurs=e.curs.nom, contractes=(e.contracte ->collect(co| Tuple{hores=co.hores-set, dedicació=co.dedicació})))})

NORMALITZACIÓ



Restriccions d'integritat textuais (afegides):

- No hi pot haver dos contractes amb els mateixos empleat i empresa.

Contracte AcabarFormació:

Afegir a la precondition:

- L'empleat no pot treballar més de 60 hores setmanals (5a rest. textual).
- Per cada contracte de l'empleat, l'empresa no pot passar a tenir més del 50% de contractes indefinis (6a rest. textual).

Afegir a la postcondició:

- Per cada contracte que s'ha passat de EnPràctiques a Indefinit, es crea l'associació ContractatPer amb l'empresa amb la qual té el contracte.