

ER
Enginyeria de
Requisits

CARÀCTERÍSTIQUES DEL SISTEMA: ESQUEMA CONCEPTUAL

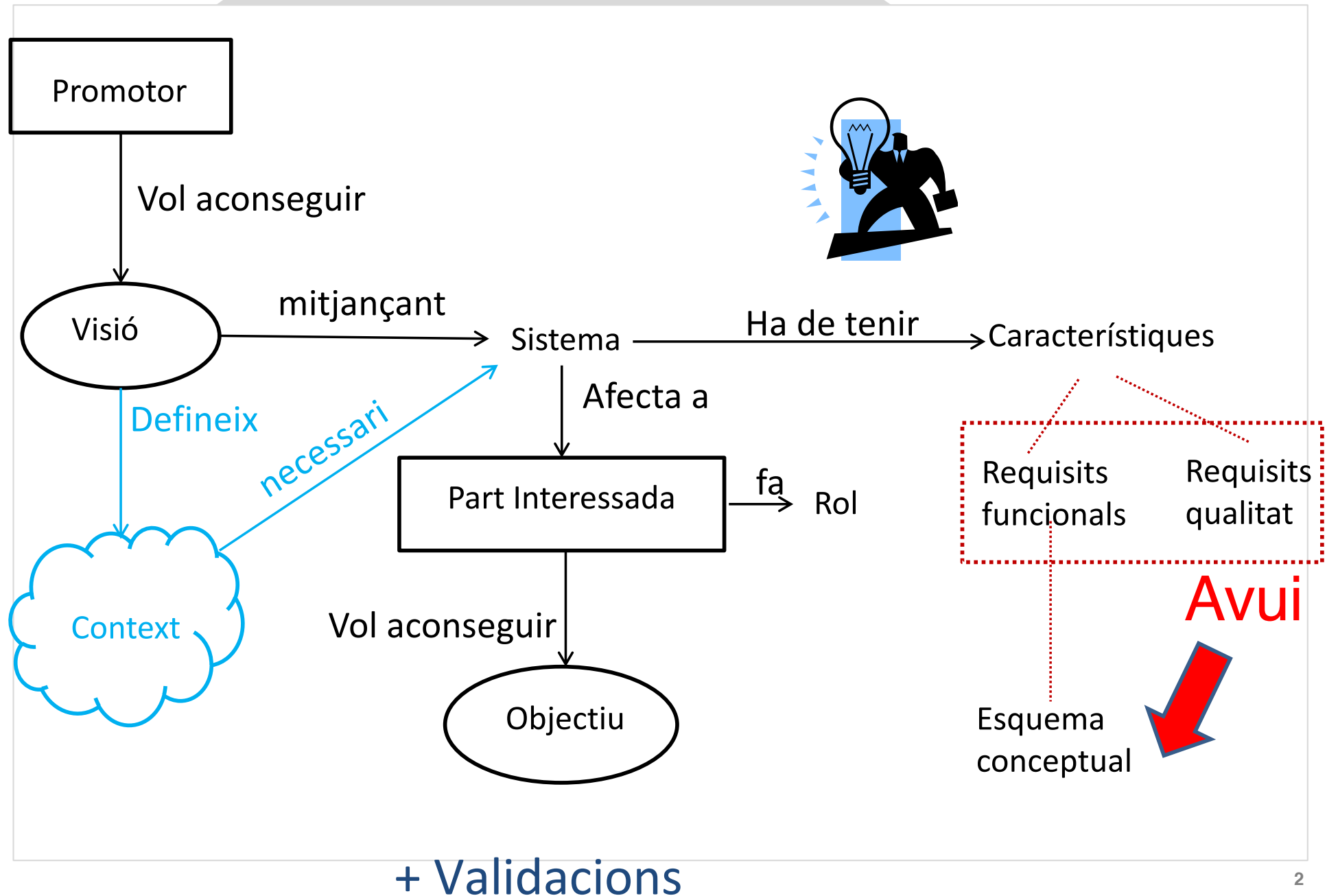
FACULTAT
D'INFORMÀTICA
DE BARCELONA

FIB



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Maria-Ribera Sancho, Ernest Teniente, Sergio Morales



La funcionalitat d'un sistema acostuma a ser **massa gran** per a ser analitzada com una sola unitat.

Característiques

- **Requisits funcionals**
 - Casos d'ús
 - Històries d'usuaris
- **Requisits de qualitat**

Els treballarem mitjançant l'exercici de la setmana que sortirà a la plataforma:

Volere Requirements Specification Template (Edition 17—2015)
Apartats 10 al 17 del document.

Objectiu classe

- Exemple d'Especificació de requisits Funcionals
 1. Esquema conceptual de les dades
 2. Esquema del Comportament
 - a) Esquema del Comportament amb Casos d'Ús+ds+op
 - b) Esquema del Comportament amb Històries d'Usuari
- Solució (publicada després de la classe)

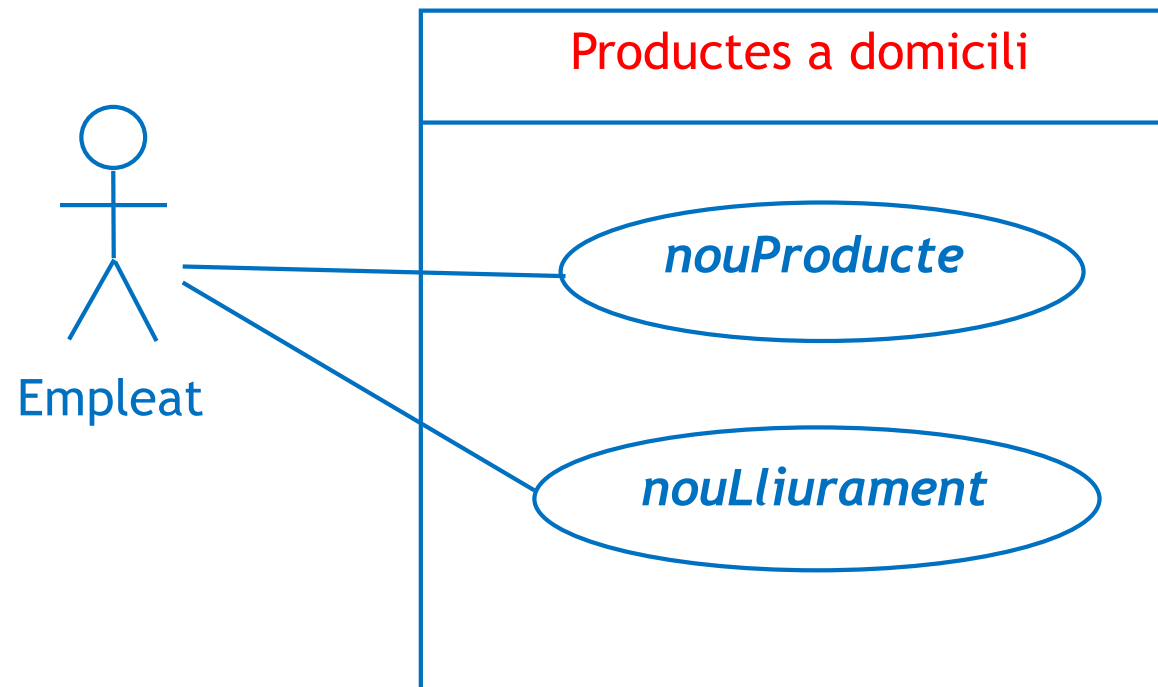


1. Claus externes de les classes no associatives: (Producte, codi); (Comanda, codi); (Client, codi i contrasenya); (Data, data); (Empleat, nom)
2. La data màxima d'una comanda ha de ser posterior o igual a la data de la comanda.
3. Un client no pot fer més de 5 comandes en una mateixa data.
4. La data de lliurament d'una comanda ha de ser posterior o igual a la data de la comanda.
5. Els productes d'un lliurament d'una comanda pertanyen a la seva comanda.
6. La suma de les quantitats dels productes lliurats d'una comanda no és superior a la suma de les quantitats dels productes demanats en la comanda
7. No es pot fer un lliurament d'una comanda reclamada amb decisió="renúncia"
8. La data de reclamació d'una comanda ha de ser posterior a la data màxima de la comanda.

1. /preu d'un lot és la suma de: preu x quantitat dels productes individuals que el componen

Diagrama de casos d'ús:

Considerarem només dos casos d'ús: **nouProducte** i **nouLliurament**.

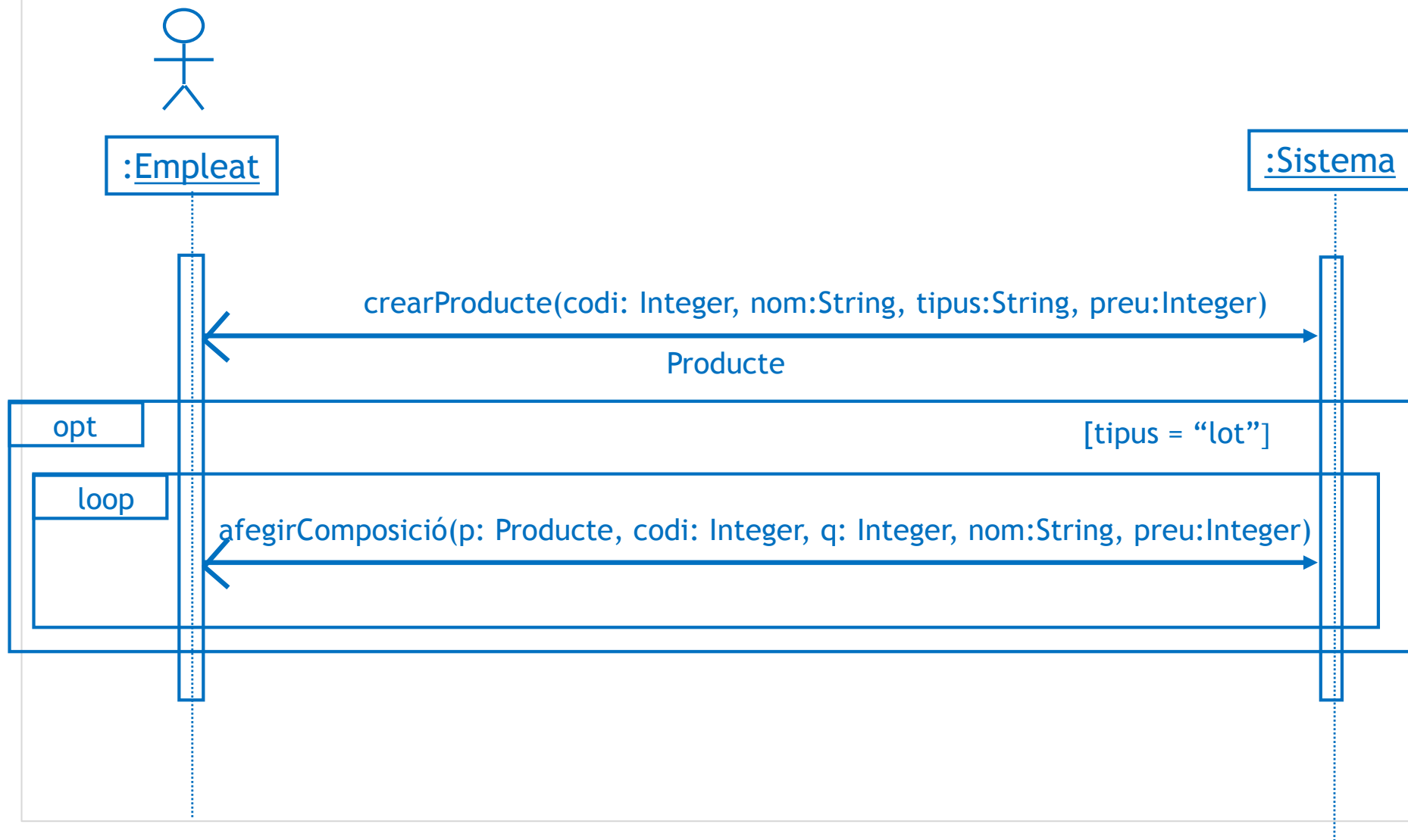


Especificació (*brief style*) dels casos d'ús:

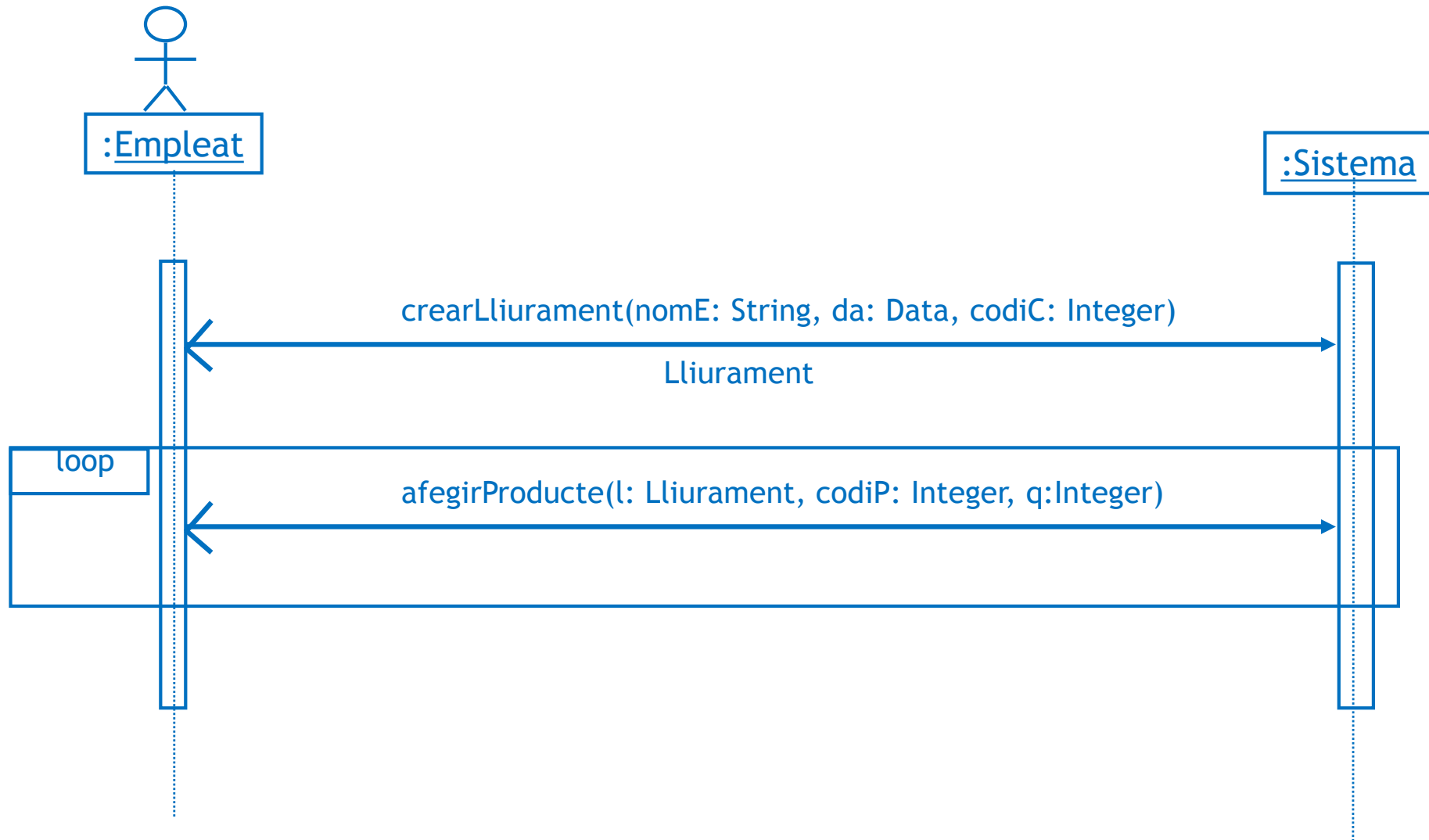
nouProducte. Quan un empleat registrat al sistema vol crear un producte ha d'indicar el codi del producte i tota la informació necessària per a donar-lo d'alta. Si el producte que es crea és un lot, aleshores també caldrà indicar tota la informació dels productes individuals que el componen. Si un producte individual d'un lot no existeix en el moment de crear el lot, aleshores també caldrà crear aquest producte individual.

nouLliurament. Quan un empleat registrat al sistema decideix fer un nou lliurament indica totes les dades necessàries per identificar el lliurament i tota la informació necessària dels productes que conté. Un lliurament només es pot fer si tots els productes que conté ja existeixen en el moment de fer el lliurament. Un lliurament no té perquè fer referència a tota una comanda sencera.

Diagrames de Seqüència: nouProducte



Diagrames de Seqüència: nouLliurament



Contractes de les operacions en OCL: **nouProducte**

context:

Sistema::crearProd(codi: Integer, nom:String, tipus:String, preu:Integer):
Producte

post:

Producte.allInstances()->exists(p | p.ocllsNew() and p.codi=codi and p.nom=nom
and
if tipus="Lot" then p.ocllsTypeOf(Lot) and result=p
else p.ocllsTypeOf(Individual) and p.oclAsType(Individual).preu=preu endif)

context:

Sistema::afegirComposició(p: Producte, codi:Integer, q:Integer, nom:String,
preu:Integer)

pre:

post:

if not Individual.allInstances()@pre -> exists(ili.codi = codi) then
Individual.allInstances()->exists(I | i.ocllsNew()and i.codi=codi and i.nom=nom
and i.preu=preu) endif
and Composició.allInstances()->exists(c| c.ocllsNew() and c.quant=q and
c.individual.codi=codi and c.lot=p)

Contractes de les operacions en OCL: **nouLliurament**

context:

Sistema::crearLliurament(nomE: String, da: Data, codiC: Integer): Lliurament

pre:

Comanda.allInstances()->exists(c | c.codi = codiC
and Empleat.allInstances()->exists(e | e.nom = nomE)

post:

Lliurament.allInstances()->exists(l | l.ocllsNew() and l.data.data = da and
l.empleat.nom = nomE and l.comanda.codi = codiC and result=l)

context:

Sistema::afegirProd(l: Lliurament, codiP: Integer, q: Integer)

pre:

Producte.allInstances()->exists(p | p.codi = codiP)

post:

ProdLliurat.allInstances()->exists(pl | pl.ocllsNew() and pl.quant=q and
pl.lliurament=l and pl.producte.codi=codiP)

Les Histories d'Usuari segueixen el format:

*As a <type of user>, I want <some goal>
so that <some reason>.*

As a **bank account holder**
I want **to be informed if my monthly balance**
is projected to go to zero or below
so that **I can arrange for an overdraft**

Hi ha tantes històries d'usuari com Casos d'ús

*As a <type of user>, I want <some goal>
so that <some reason>.*

Feature: nouProducte

Com a Empleat autènticat al sistema, **jo vull** poder crear un nou producte **de manera que** jo pugui augmentar l'oferta de l'empresa

Criteris d'acceptació:

- Ha de poder crear un producte (individual o lot) sempre que vulgui
- No podrà crear un nou producte si ja n'hi ha un altre que tingui el mateix codi
- Ha de poder assignar un producte individual ja existent a un lot sempre que vulgui

*As a <type of user>, I want <some goal>
so that <some reason>.*

Feature: nouLliurament

Criteris d'acceptació

Els Escenaris són **Criteris d'Acceptació**. Segueixen el format:

Scenario: Title

Given <context>and<some more context>

When <event>

Then <outcome>and <another outcome>

- Hi ha tants escenaris de cada història d'usuari com criteris d'acceptació té.
- Els escenaris han de considerar també els casos en que es violen les restriccions d'integritat.
- Un escenari seria un camí d'execució determinat (dels molts possibles) del contracte d'una operació.

Escenari: alta de producte individual

Donat un empleat autènticat que vol donar d'alta un producte individual

Quan introdueix un codi de producte inexistent, un nom i un preu de producte

Aleshores es crea una instància de producte individual amb aquests atributs

Escenari: alta de producte existent

Donat un empleat autènticat que vol donar d'alta un producte individual o lot

Quan introdueix un codi de producte existent

Aleshores el sistema retorna l'excepció "producte ja existeix"

Escenari: alta de producte lot

Donat un empleat autènticat que vol donar d'alta un producte lot

Quan introdueix un codi de producte inexistent i un conjunt de productes individuals amb la seva quantitat de composició

Aleshores es crea una instància de producte lot al que se li associen els productes individuals amb la quantitat corresponent

Escenari afegir producte a lot

Donat un empleat autènticat que vol afegir una composició del lot

Quan entra un producte lot existent, un producte individual existent i una quantitat

Aleshores es crea una instància de composició entre el lot i el producte individual amb la quantitat corresponent

Escenari: composició insuficient d'un producte lot

Donat un empleat autènticat que dóna per acabada la creació d'un producte lot

Quan el nombre de productes que componen el lot és inferior a dos

Aleshores el sistema retorna l'excepció "composició de lot insuficient"

Escenari: lliurament ...

Escenari: lliurament ...

Escenari: lliurament ...

Exercici complet

- Feu la versió Històries d'Usuari+Escenaris de l'exemple que hem presentat en versió clàssica
- Marqueu (subratllat o ressaltat) les referències a conceptes de l'esquema conceptual de les dades que apareguin en els escenaris.
- Indiqueu els punts forts i els punts febles que, al vostre criteri té cada enfocament.