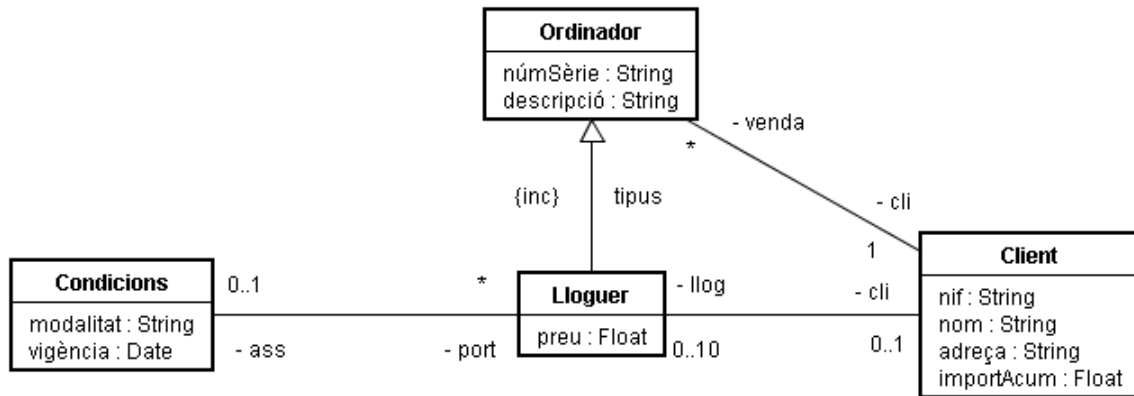


Disseny Distribuït - Ordinadors

Una tenda informàtica d'ordinadors de segona mà manté un estoc d'ordinadors. Els ordinadors els adquireixen a clients (associació entre Ordinador i Client). Alguns dels ordinadors són per llogar a clients en unes condicions determinades. El model conceptual de dades següent resumeix aquesta situació:



Restriccions textuais:

RT1: Identificadors: (Ordinador → numSèrie), (Client → nif), (Condicions → modalitat)

Es disposa del contracte següent de la capa de domini del sistema:

context CapaDeDomini::lloga(numS: String, nifC: String, modC: String)
pre existeixen les condicions de modalitat *modC*
exc ordinador-no-existeix: no existeix cap ordinador de número de sèrie *numS*
exc client-no-existeix: no existeix cap client amb nif *nifC*
exc lloguer-no-existeix: l'ordinador *numS* no és de lloguer
exc ja-llogat: l'ordinador *numS* està actualment llogat (els rols *cli* i *ass* tenen valors no nuls)
exc massa-lloguers: el client *nifC* ja té 10 ordinadors llogats
post nou-lloguer: enregistra que l'ordinador *numS* es lloga al client *nifC* amb condicions *modC*
post actualitza-import: actualitza l'import acumulat del client *nifC* amb el preu de lloguer

context CapaDeDomini::ordsEspecials(): Set(String + String+String)
post retorna el número de sèrie, descripció i nom de llogater dels ordinadors llogats en modalitat "Especial" per aquells clients que tenen un import acumulat més gran 1000 Euros i que encara no han fet cap venda

Es demana que dissenyeu aquestes operació usant Domain Model i Data Mapper sense operacions de consulta a la Capa de Gestió de Dades i controlador Transacció a la Capa de Domini. L'arquitectura ha de ser orientada a serveis considerant els serveis següents:

- Servei SvClients: l'arquitecte del software decideix segregar la informació sobre clients en un servei independent per ser usat en altres sistemes. Per cada client, tindrem el nif, el nom i l'adreça. Noteu que el servei no emmagatzema ni l'import acumulat ni les vendes.
- Servei SvCatàleg: l'arquitecte del software disposa d'un servei que manté el catàleg d'ordinadors de l'empresa, és a dir, el número de sèrie i la descripció. La informació

sobre si l'ordinador és de lloguer forma part del nostre sistema. El servei ofereix només les operacions:

<p>context SvCatàleg::nouOrdinador(númS: String, descr: String) exc ordinador-existeix: existeix l'ordinador de número de sèrie <i>númS</i> post crea un nou ordinador amb número de sèrie <i>númS</i> i descripció <i>descr</i></p> <p>context SvCatàleg::obtéDescr(númS: String): String exc ordinador-no-existeix: no existeix cap ordinador de número de sèrie <i>númS</i> post retorna la descripció de l'ordinador</p>
--

Es demana:

- 1) Diagrama de classes de disseny del sistema d'informació de la tenda informàtica, incloent-hi els atributs (però no les operacions) i també les classes façana dels serveis. Indiqueu-ne els acoblaments de les transaccions i navegabilitats finals del vostre disseny.
- 2) Diagrama de seqüència de les operacions *CapaDeDomini::lloga* i *CapaDeDomini::ordsEspecials* i de totes les auxiliars que hi apareguin, tret de les operacions que quedin assignades als serveis.
- 3) Contracte de les operacions del servei SvClients que hàgiu necessitat en el problema.
- 4) Si poguéssiu modificar el servei SvCatàleg, afegiríeu alguna operació nova? Per què? En cas afirmatiu, escriviu-ne el contracte.

Com a criteris de disseny, sobretot volem minimitzar el nombre d'invocacions remotes mantenint els criteris habituals (canviabilitat, portabilitat, reusabilitat). També es vol minimitzar en la mesura dels possibles la redundància de les dades.