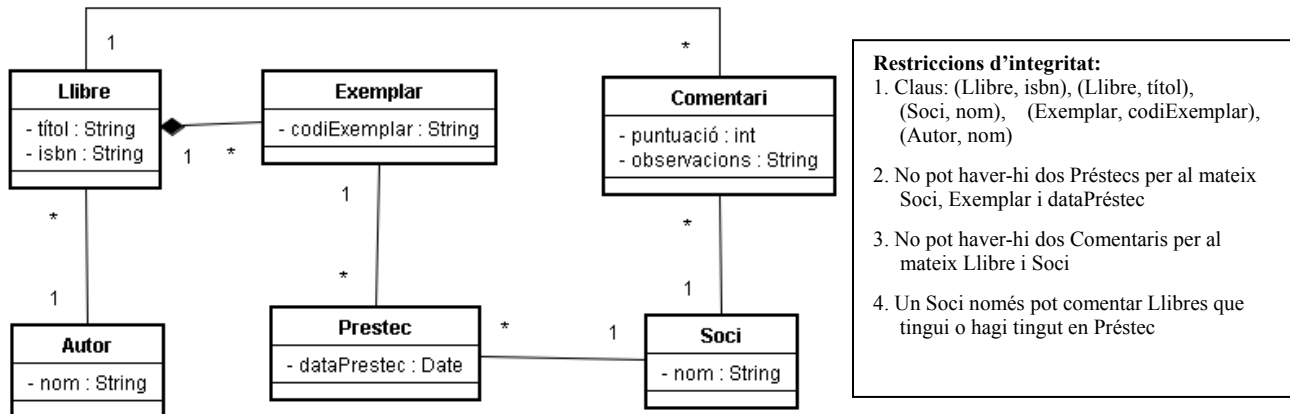


Disseny Distribuït – Biblioteques i exemplars.

Una Biblioteca vol oferir un servei de comentaris de llibres de manera que els socis de la biblioteca puguin puntuar i fer comentaris sobre els llibres que han demanat en préstec. El model conceptual (simplificat) al que han arribat és el següent:



Es disposa de dues operacions de la capa de domini amb els següents contractes:

context CapaDeDomini::mostraComentaris(títol: String):
TupleType (isbn: String, nomAutor: String,
coments: Set(TupleType(punt: int, obs: String, nomSoci: String, dataPrestec: Date))
pre: llibre-existeix: Existeix un llibre l amb títol = 'títol'
post: result = una tupla que conté el isbn del llibre, el nom de l'autor i, per cadascun dels
comentaris del llibre, la puntuació, les observacions, el nom del soci i la data en
que es va fer el darrer préstec del llibre a aquest soci

context CapaDeDomini::ferComentari(isbn: String, nomSoci: String, punt: int, obs: String)
pre: llibre-existeix: Existeix un llibre l amb isbn = 'isbn'
pre: soci-existeix: Existeix un soci s amb nom = 'nomSoci'
exc: llibre-no-prestat: no hi ha cap préstec del llibre l al soci s
exc: comentari-existeix: el soci s ja ha entrat un comentari per al llibre l
post: comentari-fet: s'ha creat una instància de comentari amb puntuació 'punt' i observacions
'obs' associada al soci s i al llibre l

Es vol, però, fer servir una arquitectura orientada a serveis on hauríem d'aprofitar dos serveis prèviament existents:

- SvCatàleg: Gestiona l' informació sobre els llibres i autors. La interfície ofereix dos contractes i NO PODEU AFEGIR-NE DE NOUS:

context SvCatàleg::dadesLlibre(títol: String): TupleType(isbn: String, títol: String)
exc: llibre-no-existeix: no existeix cap llibre l amb aquest títol
post: result.isbn = l.isbn and result.títol = l.títol

context SvCatàleg::autorLlibre(isbn: String): String
exc: llibre-no-existeix: no existeix cap llibre l amb aquest isbn
post: result = l.autor.nom

- SvPréstec: Gestiona la informació sobre préstecs, exemplars i socis. La interfície ofereix els contractes següents I PODEU AFEGIR-HI D'ALTRES:

context SvPréstec::nouSoci(nomS: String)
exc: soci-ja-existeix: ja existeix un soci amb nomS
post: afegeix un soci amb nomS

context SvPréstec::nouExemplar(isbnL: String, codiE: String)
exc: exemplar-ja-existeix: ja existeix un exemplar amb codiE
post: afegeix un exemplar codiE per al llibre que té isbn igual a isbnL

context SvPréstec::nouPréstec(nomS: String, codiE: String, dataP: Date)
exc: soci-no-existeix: no existeix un soci amb nomS
exc: exemplar-no-existeix: no existeix un exemplar amb codiE
exc: préstec-ja-existeix: ja existeix un préstec de l'exemplar codiE al soci nomS en dataP
post: afegeix un préstec de l'exemplar codiE al soci nomS en dataP

Es demana que dissenyeu el sistema de comentaris (només la capa de domini) fent servir Domain Model i Data Mapper amb operacions de consulta a la capa de gestió de dades i controlador transacció a la capa de domini. Concretament, es demana:

- Diagrama de classes del disseny del sistema d'informació de comentaris incloent-hi els atributs (però no les operacions) i també les classes façana dels serveis (aquestes sí amb les operacions). Idiqueu-ne els acoblaments i navegabilitats finals del vostre disseny. Supposeu que la navegabilitat entre Llibre i Comentari és doble.
- Diagrama de seqüència de la operació CapaDeDomini::mostraComentaris i de totes les auxiliars que hi apareguin, tret de les operacions que quedin assignades als serveis
- Diagrama de seqüència de la operació CapaDeDomini::ferComentari i de totes les auxiliars que hi apareguin, tret de les operacions que quedin assignades als serveis
- Contractes de les noves operacions del servei SvPréstec.
- Si poguéssiu modificar el servei SvCatàleg, afegiríeu alguna operació nova o canviariéu alguna de les ja existents? Per què? En cas afirmatiu, escriviu els seus contractes i a més, actualitzeu convenientment el diagrama de l'apartat a (no cal repetir classes transaccions ni façanes de servei).

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