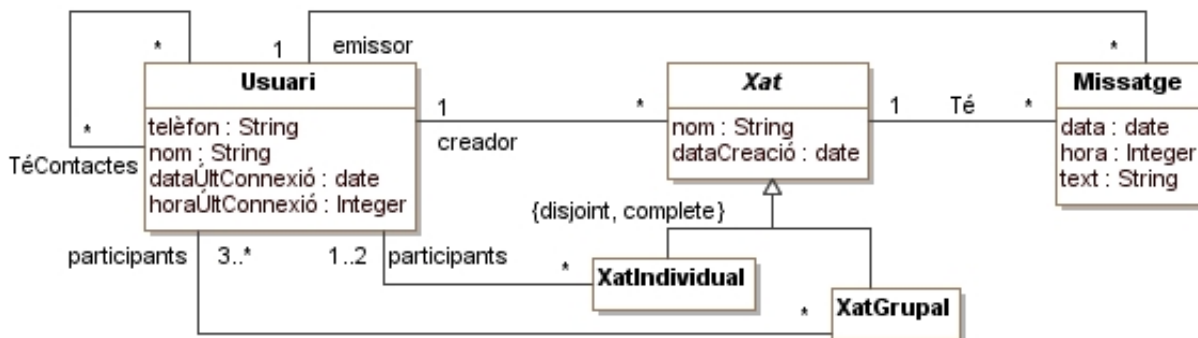


Volem dissenyar un sistema software de missatgeria. Aquest software tindrà usuaris i els seus contactes. Cada usuari podrà crear un nombre indeterminat de xats. Els xats podran ser xats individuals i grupals. Els xats individuals tindran com a molt dos participants, el creador i un altre participant i els grupals tindran un mínim de 3 participants. Els xats tindran els missatges que envien els seus emissors (que han de ser participants del xat). A continuació disposeu d'un fragment de l'esquema conceptual del sistema a dissenyar:

Esquema conceptual de l'especificació:



R.I. Textuals:

- Claus: (Usuari, telèfon); (Xat, Usuari::telèfon + nom); (Missatge, Xat::nom +Usuari::telèfon + data + hora);
- El creador d'un xat ha de ser un dels seus participants.
- L'emissor d'un missatge d'un xat ha de ser un dels participants del xat.
- La data d'un missatge ha de ser posterior a la data de la creació del xat.
- Els participants dels xats d'un usuari han de ser contactes de l'usuari del xat i ell mateix.
- Altres restriccions no rellevants pel problema

Volem dissenyar dues operacions de la capa de domini. L'operació *EliminarParticipantXat* elimina un dels participants del xat. A continuació disposeu del contracte de l'operació:

context CapaDomini :: EliminarParticipantXat (nomXat: String, usuariCreador: String, usuariParticipant: String)

✱ **exc** *xatNoExisteix*: El xat identificat per *nomXat* i *usuariCreador* no existeix.

xat **exc** *noParticipa*: L'usuari *usuariParticipant* no és un participant del xat.

xat **exc** *usuariAmbMissatges*: L'*usuariParticipant* és emissor de missatges al xat.

xat **exc** *usuariCreador*: L'*usuariParticipant* és el creador del xat.

usuari **exc** *usuariAmbPocsXats*: L'*usuariParticipant* participa en menys de 3 xats.

xat **exc** *xatAmbParticipantsMínims*: El xat té el nombre de participants mínim (3 si és grupal i 1 si és individual).

post *eliminaParticipant*: S'elimina el participant del xat.

El sistema de missatgeria permetrà configurar part de la informació que mostrarà als seus usuaris juntament amb els seus xats. En concret, inicialment oferirà dues opcions de configuració: *ÚltimMissatge* i *ÚltimaConnexió*. La primera permetrà mostrar la data de l'últim missatge rebut per l'usuari en qualsevol dels seus xats i la segona la data de l'última connexió de l'usuari. Cada usuari haurà d'indicar la configuració que desitja quan es doni d'alta en el sistema i la podrà canviar sempre que vulgui en temps d'execució. Per poder mostrar la informació relacionada amb aquesta configuració hem de dissenyar l'operació *ObtéInfoConfiguració*. A continuació disposeu del contracte de l'operació:

context CapaDomini :: ObtenirInfoConfiguració (usuari: String): date

exc *usuariNoExisteix*: L'usuari identificat per *usuari* no existeix.

exc *usuariSenseXats*: La configuració de l'usuari és *ÚltimMissatge* i no té cap xat o en té però sense missatges.

post *result*= Si la configuració de l'usuari és *ÚltimMissatge* retorna la data de l'últim missatge rebut en qualsevol dels seus xats creats. Si la configuració és *ÚltimaConnexió* retorna la data de l'última connexió de l'usuari.

Nota:

- Podeu suposar l'existència de les operacions:
 - `Xat:getDataÚltimMissatge():date` que retorna la data més recent d'entre totes les dates dels missatges del xat.
 - `comparaDates (data1:date, data2:date): date` que retorna la data més recent entre les dues proporcionades als paràmetres. Aquesta operació pot ser invocada des d'on la necessiteu.
 - `iniDate():date` que retorna una data anterior a la data de posada en marxa del sistema. Aquesta operació pot ser invocada des d'on la necessiteu.
- 1. **[5 punts]** Diagrama de seqüència de l'operació *EliminarParticipantXat* de la capa de domini. Indiqueu clarament en el diagrama de seqüència els singleton, les interfaces i les operacions que són abstractes. Supposeu que totes les navegabilitats són dobles excepte les de les associacions entre els diferents tipus de xat i els seus participants. Aquestes navegabilitats van del tipus de xat cap a l'usuari.
- 2. **[4 punts]** Diagrama de seqüència de l'operació *ObtéInfoConfiguració* de la capa de domini. Indiqueu clarament en el diagrama de seqüència els singleton, les interfaces i les operacions que són abstractes.
- 3. **[1 punt] (Competència transversal)** Per a **tots** els patrons de disseny que heu utilitzat en el disseny de les dues operacions anteriors, expliqueu de forma raonada i breu (màxim 3 línies per patró) el motiu pel qual els heu utilitzat.