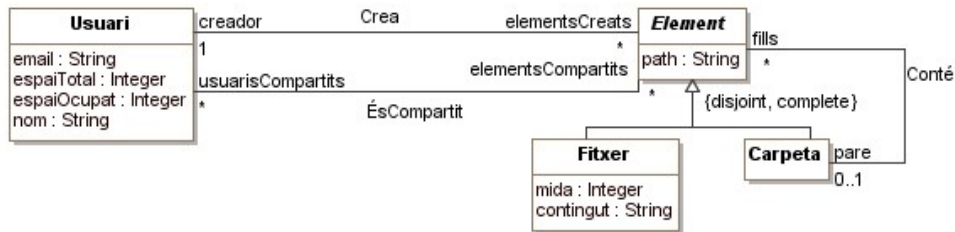


1. [3,5 punts] Volem dissenyar el sistema *ASDrive* per emmagatzemar fitxers i carpetes al núvol de diferents usuaris. Els usuaris s'identifiquen pel seu email i es registra el nom, l'espai total del que disposen en *ASDrive* i l'espai ocupat en MB. Els usuaris poden crear elements, que tenen un path, de dos tipus: carpetes i fitxers. Les carpetes poden contenir elements i els fitxers enregistren la seva mida i contingut. Els usuaris poden compartir els seus elements amb altres usuaris. A continuació disposeu de l'esquema conceptual d'*ASDrive*:



#### R.I. Textuals:

- Claus: (Usuari, email); (Element, Usuari (creador)::email + path)
- L'espai ocupat d'un usuari ha de ser menor o igual al seu espai total.
- L'espai ocupat d'un usuari ha de ser igual al sumatori de les mides dels fitxers creats i els que li han estat compartits.
- Una carpeta no es pot contenir a si mateixa.
- Si un element està compartit amb un conjunt d'usuaris, tots els elements que contingui com a mínim estaran compartits amb el mateix conjunt d'usuaris.
- Un element no es pot compartir amb el seu usuari creador.
- Altres restriccions no rellevants pel problema.

Suposant navegabilitats dobles, es demana el diagrama de seqüència de l'operació *esborraElement* d'*ASDrive*. Indiqueu clarament en el diagrama de seqüència els singleton, les interfaces i les operacions que són abstractes. Es valorarà que l'operació que esborra un element sigui altament reusable.

**context** CapaDomini:: *esborraElement* (emailCreador: String, path: String)

**exc** *elementNoExisteix*: l'element identificat pels paràmetres no existeix.

**exc** *moltesComparticions*: l'element identificat pels paràmetres s'està compartint amb més de 50 usuaris.

**post** *esborraElement*: s'esborra l'element identificat pels paràmetres i l'associació amb el seu creador i amb els usuaris compartits. Si en té, s'esborra l'associació *Conté* entre l'element i el seu pare.

**post** *baixaContingut*: si l'element identificat pels paràmetres és una carpeta, es dona de baixa en cascada tots els elements que conté carpeta.

**post** *actualitzaEspai*: si l'element identificat pels paràmetres és un fitxer, es decrementa l'espai ocupat dels usuaris amb els que s'ha compartit l'element i de l'usuari creador en la *mida* del fitxer.

2. [0,5 punts] (Competència transversal) Explica de forma raonada i breu (màxim 5 línies) si les excepcions de l'operació *esborraElement* de l'exercici 1 podrien passar a ser precondicions i com la pantalla associada al cas d'ús les podria garantir.

3. [6 punts] Després de dissenyar el sistema *ASDrive* es decideix crear dos serveis que enregistren tota la informació disponible al seu esquema conceptual per tal de que puguin ser utilitzats per altres sistemes software a desenvolupar. En concret, els serveis que es proporcionen són: un servei d'usuaris (SvUsu) i un servei d'emmagatzemat (SvEspai). A continuació disposeu d'algunes de les operacions de cada servei:

**context** SvUsu :: *getTotsUsuaris* (): Set (TupleType(email: String, nom: String, espaiTotal: Integer, espaiOcupat: Integer))

**post** es retorna l'email, nom, espaiTotal i espaiOcupat de tots els usuaris que hi ha al servei.

**context** SvUsu :: *getUsuaris* (usus:Set(email:String)): Set (TupleType(email: String, nom: String, espaiTotal: Integer, espaiOcupat: Integer))

**exc** *usuariNoExisteix*: algun dels usuaris indicats al conjunt usus no existeix.

**post** es retorna l'email, nom, espaiTotal i espaiOcupat de tots els usuaris indicats al conjunt usus.

**context** SvUsu :: *getUsuari* (email: String): TupleType(email: String, nom: String, espaiTotal: Integer, espaiOcupat: Integer)

**exc** *usuariNoExisteix*: l'usuari amb l'email *email* no existeix al servei.

**post** es retorna l'email, nom, espaiTotal i espaiOcupat de l'usuari del paràmetre.

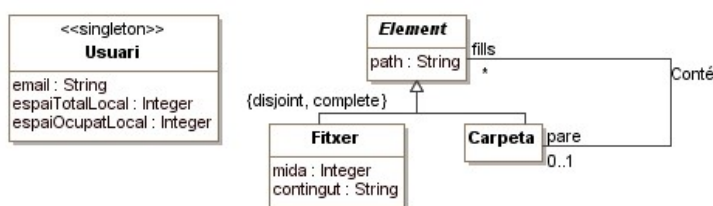
**context** SvEspai :: getInfoBàsicaFitxer (email:String, path:String): TupleType(email:String, path:String, mida:Integer, contingut: String, carpeta[0..1]:TupleType(emailC:String, pathC:String))  
**exc** fitxerNoExisteix: el fitxer identificat als paràmetres no existeix.  
**post** s'obté la informació indicada al retorn de l'operació pel fitxer identificat als paràmetres.

**context** SvEspai :: getInfoCompletaFitxer (email:String, path:String): TupleType(email:String, path:String, mida:Integer, contingut: String, usuarisCompartits: Set (emailUC:String), carpeta[0..1]:TupleType(emailC:String, pathC:String))  
**exc** fitxerNoExisteix: el fitxer identificat als paràmetres no existeix.  
**post** s'obté la informació indicada al retorn de l'operació pel fitxer identificat als paràmetres.

**context** SvEspai :: getUsuarisCompartitsFitxer (email:String, path:String): Set (emailUC:String)  
**exc** fitxerNoExisteix: el fitxer identificat als paràmetres no existeix.  
**post** s'obté l'email dels usuaris als que se'ls ha compartit (*usuarisCompartits*) el fitxer identificat als paràmetres.

**context** SvEspai :: nouFitxer (email:String, path:String, mida:Integer, contingut: String, carpeta[0..1]:TupleType(email:String, path:String))  
**exc** fitxerExisteix: el fitxer identificat als paràmetres existeix.  
**post** es crea un fitxer i s'assigna tota la informació indicada als paràmetres.

Ara dissenyarem un nou sistema software *ASBackup* que cada usuari es podrà instal·lar de forma local i que li permetrà crear una còpia local dels elements que ha creat a *ASDrive* i mantenir-los sincronitzats (no cal que us preocupeu de com es fa aquesta sincronització). En concret, la informació que aquest sistema enregistrarà serà l'email de l'usuari, l'espai total local i l'espai ocupat local. El nou sistema permetrà enregistrar aquells elements creats per l'usuari (no els que s'han compartit amb ell) que l'usuari vulgui emmagatzemar de forma local (poden ser un subconjunt dels creats en *ASDrive*). A continuació disposeu de l'esquema conceptual d'*ASBackup* que és el que haureu de fer servir en aquest exercici (podeu comprovar que hi ha una alta redundància de les dades amb l'esquema conceptual d'*ASDrive* ja que l'objectiu d'*ASBackup* és mantenir còpies locals dels elements creats a *ASDrive*):



#### R.I. Textuals:

- Claus: (Usuari, email); (Element, path)
- L'espai ocupat local d'un usuari ha de ser menor o igual al seu espai total local.
- L'espai ocupat local d'un usuari ha de ser igual al sumatori de les mides de tots els fitxers que té al sistema *ASBackup*.
- Una carpeta no es pot contenir a si mateixa.
- Altres restriccions no rellevants pel problema.

Suposant que les navegabilitats són dobles, es demana el diagrama de seqüència de l'operació *getContingutFitxerCreat* del sistema *ASBackup*. Indiqueu clarament en el diagrama de seqüència els singleton, les interfaces i les operacions que són abstractes. **Com a criteris de disseny volem minimitzar el nombre d'invocacions remotes mantenint els criteris habituals de reusabilitat i canviabilitat.**

**context** CapaDomini:: getContingutFitxerCreat (path: String): TupleType(contingut:String, usuaris: Set(nom:String))  
**pre** fitxerExisteix: el fitxer identificat per l'usuari indicat a la classe singleton (Usuari) i el path indicat al paràmetre hi és al sistema *ASDrive*.  
**exc** noHiHaEspaiLocal: el fitxer identificat pel paràmetre no hi és al sistema *ASBackup* i la mida del fitxer (emmagatzemada a *ASDrive*) és superior a l'espai lliure local.  
**post** getContingutASBackup: si el fitxer identificat pel paràmetre hi és al sistema *ASBackup* es retorna el contingut del fitxer d'aquest sistema i el nom de l'usuari creador del fitxer (al conjunt *usuaris* de la tupla).  
**post** getContingutASDriveICòpia: si el fitxer identificat pel paràmetre no hi és al sistema *ASBackup* es retorna el contingut del fitxer que hi ha al sistema *ASDrive*. A més, es crea el fitxer al sistema *ASBackup* amb les dades obtingudes del sistema *ASDrive* i s'incrementa l'espai ocupat local amb la mida del fitxer. Addicionalment, es retorna el nom dels usuaris (al conjunt *usuaris* de la tupla) amb els que s'ha compartit el fitxer (rol *usuarisCompartits* d'*ASDrive*). **Nota: Si el fitxer té una carpeta com a pare, no cal que feu l'associació amb la carpeta. La gestió de les carpetes locals les fan altres casos d'ús.**