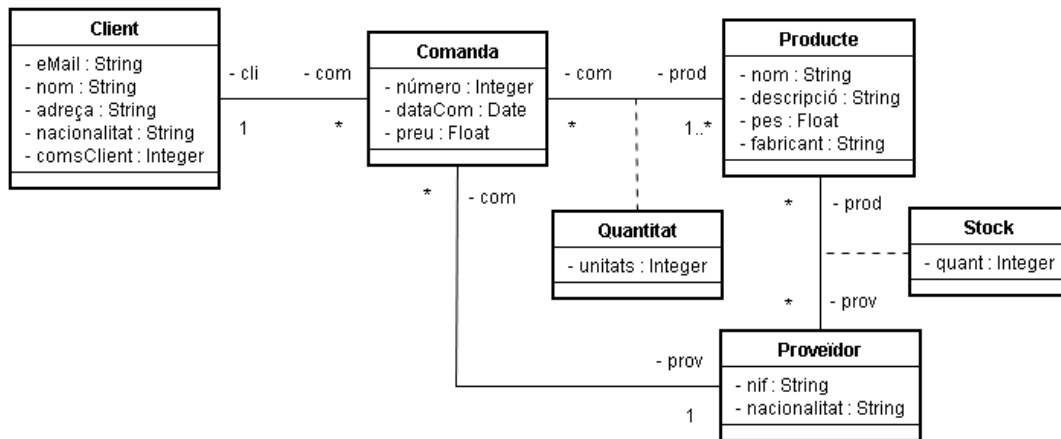


[2 punts] **Problema 5.** La companyia de venda de productes per internet Orinoc vol informatitzar la informació sobre les seves comandes segons el diagrama de classes adjunt. Les comandes fetes pels clients consten d'un nombre determinat de productes dels que pot comprar-se diverses unitats. Las comandes són servides per un únic proveïdor. Els proveïdors tenen determinat estoc dels diversos productes.



Restriccions textuais:

RT1. Claus: (Client, email); (Comanda, número); (Producte, nom); (Proveïdor, nif)
 ... altres restriccions no rellevants al problema

Suposeu que el sistema s'organitza com una arquitectura SOA amb els serveis següents:

- SvClient: manté la informació personal dels clients. En concret, aquest servei emmagatzema l'email, el nom, l'adreça i la nacionalitat dels clients. La interfície d'aquest servei la podeu determinar vosaltres.
- Serveis dels proveïdors: cada proveïdor *Prov* ofereix un servei *SvProv*. Aquest servei gestiona els productes que distribueix el proveïdor amb el seu estoc. El servei ofereix les operacions següents d'entrada, però vosaltres podeu afegir-ne:

```

context SvProv::hiHaEstoc?(nomP: String, q: Integer): Bool
post mira-estoc: retorna cert si el proveïdor té al menys q unitats del producte
    nomP, i fals si no les té (incloent el cas que no distribueixi el producte)

context SvProv::encarrega(lp: Set(String+(Integer)) -- nom producte + quantitat
exc prod-incorrecetes: el proveïdor no té prou estoc d'algun dels productes de lp (o
    no serveix algun d'ells)
post encarrega-estoc: descompta les unitats corresponents de tots els productes de lp

context SvProv::infoProd(nomP: String): String+Float+String
exc no-té-prod: el proveïdor no distribueix el producte nomP
post obté-info: retorna la descripció, el pes i el fabricant del producte nomP
  
```

[1 punt] (a) Mostreu com queda el model conceptual de dades del sistema per a Orinoc.

[1 punt] (b) Feu el diagrama de seqüència de l'operació següent de la capa de domini:

```

context CapaDeDomini::ferComanda(nc: Integer)
exc comanda-incorreceta: la comanda nc no existeix, o bé ja té proveïdor assignat
exc no-hi-ha-estoc: no hi ha prou estoc en cap proveïdor per satisfer tota la comanda
post enregistra-proveïdor: associa la comanda amb un proveïdor qualsevol dels que té
    prou estoc com per satisfer la demanda
post decrementa-estoc: actualitza l'estoc del proveïdor per cada producte de la comanda
post actualitza-client: incrementa l'atribut comsClient del client de la comanda
  
```

Useu Domain Model, Data Mapper i controlador transacció. Declareu (amb el seu contracte) les operacions que hàgiu afegir sobre els serveis, si és el cas. Podeu usar l'operació:

```

context Comanda::obtéProductes(): Set(String+Integer)
post retorna la llista de noms de productes, amb la seva quantitat, de la comanda self
  
```