



Concours d'Accès Au Doctorat Informatique

Epreuve : Génie Logiciel & Systèmes d'Information
(Documents non autorisés)

Exercice 1 (9pts)

Une entreprise souhaite modéliser avec UML le processus de formation de ses employés afin d'informatiser certaines tâches.

Le processus de formation est initialisé quand le responsable formation reçoit une demande de formation d'un employé. Cet employé peut éventuellement consulter le catalogue des formations offertes par les organismes agréés par l'entreprise. Cette demande est instruite par le responsable qui transmet son accord ou son refus à l'employé.

En cas d'accord, le responsable cherche la formation adéquate dans le catalogues des formations agréées qu'il tient à jour. Il informe l'employé du contenu de la formation et lui soumet la liste des prochaines sessions prévues. Lorsque l'employé a fait son choix, le responsable formation l'inscrit à la session retenue auprès de l'organisme de formation concerné.

En cas d'empêchement, l'employé doit avertir au plus vite le responsable formation pour que celui-ci demande l'annulation de l'inscription. A la fin de la formation, l'employé transmet une appréciation sur le stage suivi et un document attestant sa présence. Le responsable formation contrôle la facture envoyée par l'organisme de formation.

a) Donnez le diagramme des cas d'utilisation.

b) Donnez le schéma des classes de cette application, incluant toutes les classes que l'on peut déduire de l'énoncé, ainsi que les associations entre classes avec leurs cardinalités.

c) Donnez le diagramme de séquence associé à la demande initiale de l'employé décrite dans le deuxième paragraphe de l'énoncé ; assurer la cohérence de votre réponse avec la question précédente.

Exercice 2 (11pts)

Un centre hospitalier régional souhaite développer une nouvelle application pour la gestion des visites et des séjours d'hospitalisation de patients dans ses services.

Tout premier passage d'un patient au sein de l'hôpital, qu'il s'agisse d'une visite ou d'une hospitalisation, implique son inscription. Cette inscription consiste à lui affecter un numéro qui permettra de l'identifier lors de toute nouvelle visite ou hospitalisation ultérieure. Tout patient est également caractérisé par ses nom, prénom, coordonnées postale et téléphonique.

Lorsqu'il s'agit d'une simple visite, l'accueil de l'hôpital note le médecin que le patient vient consulter, ainsi que la date et l'heure de la visite. Une fois la visite terminée, le médecin doit reporter dans le dossier patient l'ensemble des actes pratiqués sur le patient, ainsi qu'un compte rendu pour chaque acte. Par exemple, un patient venant consulter le Dr XX pour une douleur persistante à l'épaule pourra être amené à effectuer une radio auprès du docteur YY (ce qui donnera lieu à deux visites et à un acte par visite).

Lorsqu'il s'agit d'une hospitalisation, l'accueil de l'hôpital note la date d'entrée du patient, la chambre dans laquelle le patient est hospitalisé, ainsi que le motif du séjour à l'hôpital. Une fois l'hospitalisation terminée, l'accueil enregistre la date de sortie du patient. Il s'avère que pour des raisons évidentes de gestion, lors de l'arrivée du patient, l'accueil enregistre une date prévisionnelle de sortie. C'est cette même date qui est modifiée lors de la sortie effective du patient. Tout au long de l'hospitalisation, le personnel soignant reporte également l'ensemble des actes à pratiquer sur le patient, le médecin qui doit pratiquer chaque acte, ainsi que la date et l'heure de l'acte. Une fois l'acte terminé sur le patient, le médecin ayant pratiqué cet acte doit enregistrer son compte rendu dans le dossier patient.

Tout acte est caractérisé par un code, un libellé, une durée prévisionnelle et un coût. Chaque acte correspond à une spécialité précise, elle-même caractérisé par un code et un libellé. Lorsqu'il s'agit d'une hospitalisation, cette durée représente la durée minimale de séjour d'un patient pour cet acte. Les services sont répertoriés avec un code et un nom. Chaque service comprend et gère un certain nombre de chambres. Chaque chambre est caractérisée par un numéro unique (une chambre n'est gérée que par un seul service) et un nombre de lits. On pourra ainsi connaître le nombre de lits dont dispose chaque service, la liste des médecins affectés à chaque service (soit à titre principal, soit pour des visites). Chaque médecin est répertorié avec ses code, nom, prénom, téléphone, numéro de bureau, ses spécialités (un médecin possède au moins une spécialité), le service auquel il est affecté, ainsi que son titre (par exemple, praticien hospitalier, chef de clinique, ...).

1. Donnez le diagramme de cas d'utilisation propres à ce système.
2. Donnez les scénarios majeurs des cas d'utilisation proposés.
3. Donnez le diagramme de classe global de l'application.
4. Donnez un diagrammes d'états-transitions de la classe (du diagramme proposé en 3) ayant une dynamique la plus importante dans cet gestion.

Vous pouvez poser les hypothèses que vous jugez nécessaires dans le but de répondre aux problèmes posés. Il sera tenu compte de la clarté des réponses.

Concours d'Accès Au Doctorat Informatique

Epreuve : Base de Données Avancées
(Documents non autorisés)

Exercice 1(4 pts) :

Soit un schéma relationnel de bases de données contenant les relations suivantes :

Bureau(NumBureau, NumTelephone, Taille) avec

F1 = { NumBureau → NumTelephone, Taille; NumTelephone → NumBureau; }

Occupant(NumBureau, PersonnelID) avec F2 = { NumBureau → PersonnelID }

Materiel(NumBureau, NumPC) avec F3 = { NumPC → NumBureau }

1. Les contraintes ci-dessous sont-elles vérifiées par ce schéma de bases de données? Si la réponse est positive, expliquez pourquoi. Si la réponse est négative, indiquez quelle(s) dépendance(s) fonctionnelle(s) il faut ajouter/supprimer ou modifier pour que la contrainte soit vérifiée.

(a) "Un bureau peut contenir plusieurs postes téléphoniques."

(b) "Il y a une et une seule personne par bureau."

(c) "Un bureau contient un seul PC."

2. A partir de l'ensemble de dépendances fonctionnelles initiales données dans l'énoncé, indiquez quelles sont les clés candidates possibles de chaque relation.

Exercice 2(6 pts) :

1. Les instructions SQL3 suivantes sont-elles correctes ? Si une instruction n'est pas correcte, dites laquelle et expliquez pourquoi.

```
create type A as object (C number);
create type B under A (D number);
create table T of A;
insert into T valuesc(B(10,20));
select x.D from T x;
```

2. Soit le schéma objet suivant :

```
CREATE TYPE ModeleRoue AS OBJECT ( nom VARCHAR2(10) );
CREATE TYPE ModeleMoteur AS OBJECT (nom VARCHAR2(10) );
CREATE TYPE ModeleVoiture AS OBJECT ( nom VARCHAR2(50) );
```

```
CREATE TABLE EnsModeleVoiture OF ModeleVoiture;
CREATE TABLE EnsModeleMoteur OF ModeleMoteur;
CREATE TABLE EnsModeleRoue OF ModeleRoue;
```

```
CREATE TYPE Roue AS OBJECT (
    numero NUMBER(20),
    modele REF ModeleRoue,
    marque VARCHAR2(50)
);
CREATE TYPE Moteur AS OBJECT (
    numero NUMBER(20),
    modele REF ModeleMoteur,
    marque VARCHAR2(50)
);
```



```
CREATE TYPE LesRoues AS TABLE OF Roue;
CREATE TYPE Voiture AS OBJECT (
    numero NUMBER(20),
    modele REF ModeleVoiture,
    roues LesRoues,
    mot Moteur
);
```

```
CREATE TABLE EnsVoiture OF Voiture
NESTED TABLE roues STORE AS tabroue;
```

Requêtes :

Donner le code SQL des requêtes suivantes :

1. Les numéros de voitures dont le moteur est de type "CDI" et qui sont de type "Renault".
2. Le garagiste a fait des erreurs de saisie et il a associé le même moteur à plusieurs voitures. Trouver les couples de voitures qui possèdent le même moteur
3. Les numéros des voitures qui possèdent la roue numéro 444
4. Le nombre de voitures par modèle de moteur

Exercice 3(10 pts):

Soit donné un document planning.xml validé par la DTD suivante :

```
<!ELEMENT PLAN ((ETUDIANT|ENSEIGNANT)+, COURS+, INSCR+)>
<!ELEMENT ETUDIANT (NOM, COORD)>
<!ATTLIST ETUDIANT NO ID>
<!ELEMENT INSCR #PCDATA>
<!ATTLIST INSCR ETUDIANT IDREF COURS IDREF>
<!ELEMENT ENSEIGNANT (NOM, EMAIL)>
<!ATTLIST ENSEIGNANT NO ID>
<!ELEMENT COURS (INTIT,SALLE)>
<!ATTLIST COURS NO ID RESP IDREF>
<!ELEMENT COORD (ADR, EMAIL)>
<!ELEMENT NOM #PCDATA>
<!ELEMENT SALLE #PCDATA>
<!ELEMENT ADR #PCDATA>
```

Formulez les requêtes suivantes en XQuery / XPath 2.0 :

1. Les noms et les adresses email de tous les étudiants.
2. L'adresse email de l'étudiant avec le numéro 24367.
3. Les intitulés des cours suivis par l'étudiant 24367.
4. Les noms et les numéros des responsables du cours avec le numéro NI244.