

Examen

| | |
|--|-------------------|
| Année Universitaire : 2016 - 2017 | Date : 13/01/2017 |
| Filière : Ingénieur | |
| Semestre : S5 | |
| Période : P1 & P2 | Durée : 1h30min |
| Module : M.5.1 : Intégration des Applications d'Entreprise | |
| Elément de Module : M.5.1.1 : Intégration des Services et des objets | |
| Professeur : M. Nassar | |

Documents non autorisés

Questions de cours (7 points)

- Les réponses aux questions suivantes doivent être concises.
1. Qu'est-ce que le référentiel d'interfaces dans CORBA ? (1 pt)
 2. Le Service de Nommage (Naming Service) est-il un objet accessible à distance ou un objet local (un objet possédant un IOR ou non) ? (1 pt)
 3. Quels sont les deux éléments dans Corba qui permettent l'interopérabilité. (1 pt)
 4. Qu'est-ce qu'une architecture logicielle de type SOA et quels sont ses avantages ? (1 pt)
 5. Y a-t-il des liens entre les notions d'architecture SOA et de processus métier ? Si oui lesquels ? (1 pt)
 6. Qu'est-ce qu'un service web ? (1 pt)
 7. Lorsqu'un client souhaite utiliser un service web, faut-il obligatoirement qu'il possède la description WSDL du service ? Si oui, pourquoi ? Si non, quelles sont les alternatives ? (1 pt)

Exercice (13 pts)

On considère un parking dont on souhaite automatiser la gestion en l'administrant à distance avec un middleware CORBA. Le parking peut accueillir au maximum 50 véhicules. Une barrière gère les entrées et les sorties de véhicules ; elle délivre un ticket à l'entrée d'un véhicule et elle contrôle le ticket fourni par l'automobiliste à la sortie. Un automate de paiement permet sur présentation du ticket, de régler le montant correspondant au temps de stationnement. L'automobiliste règle le montant du stationnement à l'automate avant de franchir la barrière pour sortir. La barrière, l'automate et le parking sont représentés chacun par un objet CORBA.

L'interface *BarriereIf* de la barrière fournit deux méthodes :

- *entrer* : correspond à l'entrée d'un véhicule dans le parking. Retourne vrai si le véhicule peut rentrer (si place libre), faux sinon (et dans ce cas le véhicule ne peut pas entrer). Cette méthode fournit en paramètre de sortie une structure *Ticket* comprenant un numéro de ticket de type entier et une heure d'entrée sous la forme d'une chaîne de caractères.

- **sortir** : correspond à la sortie d'un véhicule. Retourne vrai si l'automobiliste a réglé le montant du stationnement, faux sinon (et dans ce cas le véhicule ne peut pas sortir). Cette méthode prend en paramètre d'entrée une structure Ticket.

L'interface **AutomateItf** de l'automate fournit deux méthodes :

- **payer** : correspond au paiement en fonction de la durée de stationnement. Retourne vrai si le montant fourni par l'automobiliste est suffisant, faux sinon. Cette méthode prend en paramètre d'entrée une structure Ticket et un montant de type réel double. Cette méthode fournit également en sortie une valeur de type réel double représentant la monnaie à rendre à l'utilisateur.

- **cestpaye** : à partir d'une structure Ticket fournie en entrée, retourne vrai si le montant du ticket a été réglé, faux sinon.

L'interface **ParkingItf** du parking fournit trois méthodes :

- **nbPlacesLibres** : retourne un entier indiquant le nombre de places libres dans le parking.

- **entreeVehicule** : signale l'entrée d'un véhicule sur le parking. Ne prend aucun paramètre, ne retourne rien.

- **sortieVehicule** : signale la sortie d'un véhicule du parking. Ne prend aucun paramètre, ne retourne rien.

1. Définir la structure Ticket et les interfaces IDL correspondant à ces trois objets CORBA dans un module ParkingPkg. (2 pts)
2. Lorsqu'un automobiliste se présente à la Barrière, l'objet CORBA barrière interroge l'objet Parking avant de laisser entrer l'automobiliste. Pourquoi ? Quelle méthode invoque-t-il ? (2 pts)
3. A l'issue de cela, il se peut que l'objet Barrière invoque la méthode **entreeVehicule** de l'objet Parking. Cette méthode n'a pas de paramètres et a pourtant un rôle essentiel. Lequel ? (2,5 pts)
4. Décrire le scénario de la sortie d'un véhicule du parking : quel(s) objet(s) appelle(nt) quelle(s) méthode(s) de quel(s) autre(s) objet(s) et pourquoi ? En particulier, on s'attachera à fournir un scénario de fonctionnement permettant à l'objet Barrière d'autoriser ou non la sortie du véhicule. (2,5 pts)
5. Donner le code Java de la classe ParkingImpl implantant l'interface ParkingItf. (2 pts)
6. Même question pour l'interface BarriereItf. (2 pts)

Bonne chance