

INFO-H-302 : Analyse et conception par objets
Prof. E. Zimányi
Examen de première session 2008-2009

Remarques préliminaires

- N’oubliez pas d’inscrire vos noms, prénoms et numéro d’ordre sur chacune des feuilles de réponse.
- Vous disposez de 3 heures.
- Vous pouvez utiliser : un bic, un crayon.
- L’examen comporte 5 questions.

Question 1 – Étude de cas [7 points]

On vous demande de concevoir une application permettant la réservation de billets de concert pour l’ensemble des salles de Belgique. Les places de chaque salle sont réparties en catégories. Le prix d’un billet dépend de l’artiste, de la date et de la catégorie.

Une réservation se déroule comme suit : (i) l’utilisateur se connecte au système, (ii) il choisit le concert, (iii) le système lui montre les places encore disponibles dans chaque catégorie, (iv) l’utilisateur choisit la place, (v) le paiement est effectué.

On vous demande de :

1. Lister les cas d’utilisation de cette application.
2. Détailler le cas d’utilisation ’effectuer réservation’ en utilisant le modèle vu au cours pour les cas d’utilisation détaillés.
3. Établir un modèle conceptuel pour ce domaine, décrivant les classes significatives et les relations qui les lient, et le représenter en UML.
4. Illustrer sous forme de diagramme de communication la réaction du système lorsque deux utilisateurs tentent d’acheter au même moment la dernière place disponible d’un concert.

Question 2 – Patron de conception [4 points]

Présenter de manière complète le patron de conception *Observateur* à l'aide d'un diagramme de classe et un diagramme de communication.

Question 3 – Cohésion [3 points]

(a) Définir la notion de *cohésion*.

(b) Expliquer le lien avec la conception de logiciel : pourquoi une bonne cohésion est-elle importante dans le cadre de la conception de logiciel ? Que faire lorsque la cohésion n'est pas bonne ?

Question 4 – Diagramme d'états [3 points]

(a) Expliquer à l'aide d'un exemple bien choisi la notion *d'états concurrents* dans un diagramme d'état.

(b) Quelle est l'utilité de cette notion ? Peut-on s'en passer sans perte d'expressivité ? Expliquer.

Question 5 – Composition [3 points]

(a) Définir la notion *d'agrégation forte (composition)*.

(b) En Java (ou C++), comment assurer qu'une association soit une agrégation forte (composition) ? Autrement dit, comment conserver cette relation de composition lorsque ces classes sont utilisées par un autre code que vous ne contrôlez pas. Justifier exhaustivement.

BON TRAVAIL !