

Analyse Orientée Objet – ING1

TD 11 : Révision

Exercice 1. Etude de cas (1h30)

Une agence de tourisme a besoin de créer un logiciel pour gérer les réservations d'hôtel de ses clients via Internet. Voici les informations concernant le cahier des charges et l'analyse fonctionnelle de logiciel :

- Un client a un nom, un prénom et une adresse. *On n'accepte pas de clients homonymes.*
- Un hôtel a un nom, une adresse, une catégorie (1^{ère} étoile, 2^{ème} étoile, ...). Il est composé de plusieurs chambres. Pour chaque chambre, on connaît le type (simple ou double), le nombre maximum d'adultes et le nombre maximum d'enfants acceptés. Le prix unitaire par nuit d'une chambre varie selon la saison. Chaque saison est caractérisée par la date de début et la date de fin. *Bien sur, il n'y a pas de chevauchement entre les saisons.*
- Afin de faire les réservations sur le site, le client doit s'identifier avec un login et un mot de passe. S'il n'a pas encore un compte, il faut en créer un.
- Pour chercher des hôtels disponibles, le client doit d'abord saisir la ville, la date d'arrivée, la date de départ (*le nombre de nuits sera calculé à partir de ces deux dates*), le nombre d'adultes et le nombre d'enfants accompagnés. Le logiciel va vérifier la disponibilité des hôtels dont l'adresse correspond à la ville saisie, calculer *le prix total en fonction de prix unitaire par chambre* et les propose au client. Le client va réserver un hôtel parmi la liste et on lui attribue un numéro. *Il faut respecter la capacité des chambres associées à une réservation.*
- Une fois la réservation est faite, le client peut procéder au paiement tout de suite, ou décider de revenir plus tard pour payer ou annuler la réservation. Pour ces fonctionnalités du système, il faut bien sûr s'identifier.

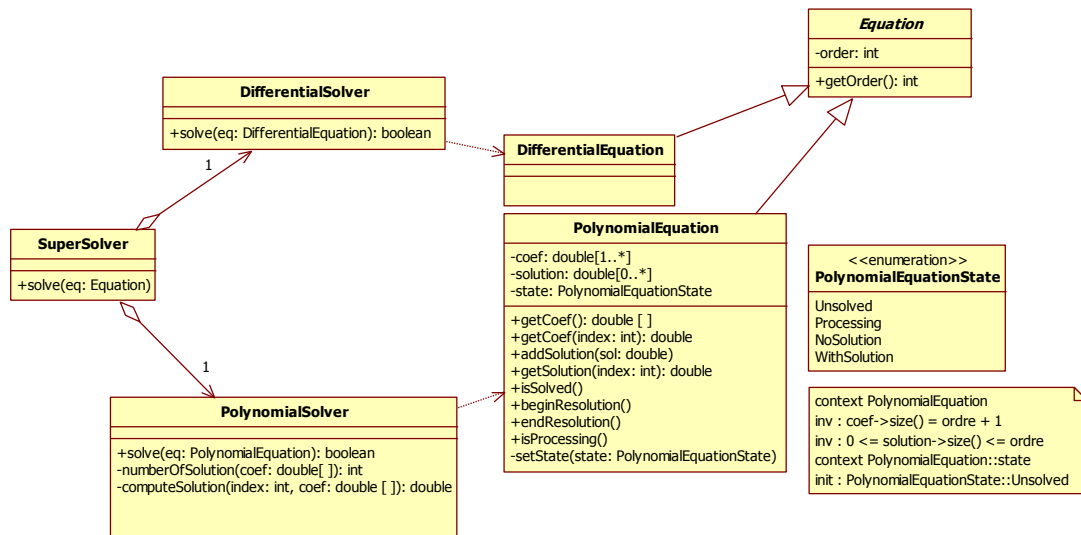
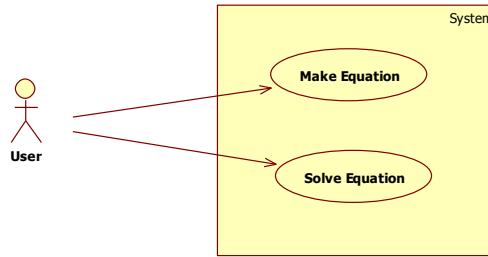
Etablir pour ce cas d'étude les diagrammes UML suivants :

- 1) Diagramme de cas d'utilisation
- 2) Diagramme d'activité pour le cas d'utilisation **Réserver hôtel**
- 3) Diagramme de classes avec les contraintes OCL écrites en italique
- 4) Diagramme de séquence pour le cas d'utilisation **Réserver hôtel**
- 5) Diagramme d'états-transitions pour l'objet **Réservation** avec les différents états : réservée, payée ou annulée.

Essayer d'avoir une cohérence maximale entre les différents diagrammes en complétant vos diagrammes avec tous les attributs, les opérations, les événements, les énumérations, ... nécessaires.

Exercice 2. Diagramme de séquence et diagramme d'états-transitions (1h)

On considère un système de résolution d'équation dont on vous donne le diagramme des cas d'utilisation et le diagramme de classe.



Le scénario du cas d'utilisation Solve Equation est le suivant :

1. L'acteur user donne une équation à résoudre à un super solveur.
2. Suivant le type de l'équation le super solveur délègue la résolution au solveur polynomial ou au solveur différentiel.
3. Le scénario de la résolution différentielle n'est pas précisé. On le désignera uniquement par son nom : Differential Solving.
4. Pour le solveur polynomial :
 - a. Si l'équation est déjà résolue ou en cours de résolution, il s'arrête et renvoie vrai.
 - b. Il récupère l'ordre (i.e. degré) de l'équation polynomiale puis ses coefficients.
 - c. Il calcule le nombre de solutions N en fonction des valeurs des coefficients.
 - d. L'équation passe en mode Processing en signalant le début de sa résolution
 - e. Il calcule les N solutions en fonction des coefficients de l'équation.
 - f. Les solutions sont au fur et à mesure ajoutées à l'équation.
 - g. La résolution étant finie :
 - i. Le solveur signale à l'équation la fin de la résolution et cette dernière passe à l'état NoSolution ou WithSolution selon N.
 - ii. Le solveur renvoie vrai pour indiquer que la résolution est accomplie.

Question 1) Etablir le diagramme de séquence nommé Solving de ce scénario. Il est conseillé de découper ce diagramme en plusieurs sous-parties (une pour chaque solveur par exemple).

Question 2) Etablir le diagramme d'état d'un objet PolynomialEquation.