

# LOG2410 TP2

## Diagramme de concepts et assignation des responsabilités

### Automne 2019

#### 1. Objectifs

L'objectif de ce laboratoire consiste à produire les premiers artéfacts orientés-objet d'une itération initiale de développement logiciel et d'être en mesure de distinguer les éléments conceptuels des éléments logiciels d'une application donnée. Ce travail pratique amènera aussi l'étudiant à se familiariser avec certains concepts fondamentaux du développement logiciel, soit la conception selon les patrons GRASP et selon les principes de réutilisabilité et d'extensibilité.

Dans le cadre de ce laboratoire, deux types de diagrammes orientés-objet seront modélisés à l'aide de l'outil Enterprise Architect : le diagramme de concepts et les diagrammes de classes.

Note : Utilisez le même projet Enterprise Architect LOG2410\_EquiPoly.eapx utilisé au TP1 pour modéliser vos diagrammes.

#### 2. Diagramme de concepts (25 pts)

Le diagramme de concepts est l'étape la plus importante de l'analyse orientée-objet. C'est un glossaire visuel des classes conceptuelles d'un problème donné. Ce diagramme est une source d'inspiration pour la construction des diagrammes de classes. La construction de ce modèle favorise la compréhension du problème et clarifie la terminologie ainsi que le vocabulaire du domaine. Le diagramme de concepts est représenté par un diagramme de classes où les concepts sont présentés par des classes. Les classes conceptuelles sont en relation les unes avec les autres afin de former une description visuelle du domaine d'application.

À partir du modèle fourni au cours de ce laboratoire (Le fichier Modèle.pdf sur Moodle), construisez le diagramme de concepts avec Enterprise Architect.

Le diagramme doit comprendre :

- Les classes conceptuelles ainsi qu'une description sommaire des concepts.

- Les relations et les associations entre les concepts ainsi que leurs multiplicités sources et cibles au besoin.
- Les attributs pertinents.

**Attention :**

- Un diagramme de concepts n'est pas une représentation des entités logicielles d'un système (concepts != classes logicielles).
- Les relations entre les concepts sont purement conceptuelles et ne doivent pas être une représentation d'un flux de données ou de relations logicielles.
- Un attribut de concepts est généralement limité à des types simples : bool, string, date, etc.
- Le(s) diagramme(s) de concepts doi(ven)t se retrouver sous la vue « Analyse »

**3. Diagramme de classes et diagramme d'interaction (25+25 pts)**

Le diagramme de classes et le diagramme d'interaction font, lors de la réalisation des cas d'utilisation, un travail synergique qui aboutit à une spécification logicielle. On vous demande de **choisir un cas d'utilisation**, parmi ceux définis dans la version précédente de votre modèle (au TP1), pour lequel vous allez produire le diagramme de classes et le diagramme d'interaction. Le diagramme d'interaction utilise les classes et les méthodes définies dans le diagramme de classes afin de réaliser le cas d'utilisation choisi.

Chaque classe doit comprendre :

- Les méthodes avec leurs paramètres et les attributs (avec leurs types) pertinents aux cas d'utilisation réalisé.
- Les associations nommées entre les classes.
- La visibilité (aussi appelée navigation) entre les classes indiquées par la direction dans vos associations.

**Attention :**

- Les associations sont aussi de nature logicielle (agrégation, composition, généralisation, etc.).
- Il est possible que les associations conceptuelles et logicielles n'aient pas la même multiplicité.

- Le choix de la nomenclature des éléments logiciels vous revient mais celle-ci doit être uniforme.

**Question :**

Expliquez-en vos propres mots la/les différence(s) entre le diagramme de concepts et le diagramme de classes.

**4. Assignment de responsabilité et patrons GRASP (25 pts)**

Lorsque vous avez ajouté des méthodes dans vos classes, vous avez implicitement choisi d'assigner des responsabilités à ces classes. Identifiez au moins quatre responsabilités que vous avez assignées à ces classes en justifiant votre choix d'assigner ces responsabilités à ces classes en vous référant aux patrons GRASP. En choisissant les responsabilités, assurez-vous de vous référer à au moins quatre patrons GRASP différents. Faites aussi **un diagramme de classes** contenant seulement les classes pertinentes pour démontrer votre conception et qui illustre les quatre patrons GRASP identifiés.

**5. Remise**

Créez un fichier compressé (.zip) nommé LOG2410\_MatriculeA\_MatriculeB\_TP2.zip de votre répertoire de projet et soumettez votre fichier dans l'un des deux comptes Moodle avant la fin de la séance de laboratoire.

Le fichier compressé doit contenir :

- Votre répertoire de projet LOG2410\_EquiPoly avec le modèle Enterprise Architect.
- Le fichier RéponsesTP2.pdf qui contient **le cas d'utilisation choisi** pour produire le diagramme de classes et le diagramme d'interaction, les réponses à la partie 3 et la partie 4. Ce fichier doit aussi contenir l'image de votre nouveau diagramme de classes de la partie 4 du TP.