

TP Outils informatiques pour la communication : diagrammes de classes

Au cours de ce TP, nous chercherons à modéliser des diagrammes de classes UML. Nous utiliserons le logiciel Dia pour les représenter. Il s'agit d'un éditeur graphique de diagrammes, dans différents langages dont UML. L'utilisation en est assez intuitive ; pour réaliser un diagramme UML, il faut choisir « UML » dans la liste déroulante présentant les langages (menu de gauche), puis glisser-déposer les éléments de la palette de gauche vers le diagramme.

Le diagramme est sauvegardé dans un format propre à Dia, dans un fichier portant l'extension « .dia ». On peut ensuite exporter ces diagrammes dans une grande variété de format. L'export étant loin d'être parfait, le mieux dans le cadre d'un rapport est d'utiliser l'export au format « PNG (anti-crênelé) » (ou antialiased en anglais), en gardant la taille par défaut proposée par Dia. On peut alors inclure les diagrammes dans un fichier L^AT_EX.

Quelques conseils pour Dia :

- Les noms, attributs, méthodes des classes sont à renseigner dans la fenêtre obtenue par un double clic sur la classe.
- Les multiplicités, types de flèches, type de traits des relations sont configurables de même.
- Les extrémités des relations s'accrochent aux classes (via les points d'ancrage visibles sur les côtés des rectangles représentant ces dernières).
- Le type de retour d'une méthode (si utile), est spécifié dans le champ « type » de la définition de la méthode.

La notion importante du TP est la conception de diagrammes. En conséquence, si vous connaissez et préférez un autre logiciel de représentation de diagrammes de classes, vous pouvez utiliser celui-ci.

Exercice 1 Banque

1. On souhaite concevoir un système informatique modélisant une banque incluant des informations sur des comptes et des clients. L'information relative à un client consiste en son nom, son prénom, son adresse, son téléphone, ainsi que son numéro de sécurité sociale. Les comptes ont un numéro, un solde, un type (courant, épargne, etc.). Il faut en outre pouvoir déterminer à tout moment quels sont les comptes d'un client donné ainsi que le client propriétaire d'un compte.

Donner un diagramme de classes UML, représentant les classes avec leurs attributs et méthodes. Pour les associations, donnez un nom et les multiplicités (inutile de donner les rôles). Discutez les multiplicités possibles des associations.

2. Modifier la conception de la question précédente en tenant compte des informations supplémentaires suivantes :
 - Pour chaque compte, il peut y avoir plusieurs titulaires.
 - Un client peut posséder plusieurs comptes mais il est considéré comme client s'il a au moins un compte.
 - Les adresses sont composées d'un numéro, d'un nom de rue, d'une ville, et d'un code postal. Un client peut avoir plusieurs adresses.
 - Pour chaque adresse, il peut y avoir plusieurs numéros de téléphone fixe et on connaît leur type (dégrouper ou non dégroupé). Ceci implique qu'il est nécessaire de conserver pour chaque adresse quels numéros lui sont associés. En outre, chaque client peut avoir un numéro de téléphone portable (on n'en stockera qu'un seul au plus) dont il est le seul propriétaire.

Exercice 2 Équipes

1. Donner une représentation UML pour la gestion d'un système gérant de l'information relative à des équipes, des joueurs et leurs fans. On souhaite un diagramme de classes avec les attributs mais pas de méthodes. Pour les associations, donnez un nom et les multiplicités, mais pas de rôles. On suppose que :
 - chaque équipe a un nom, un capitaine et une couleur de maillot
 - chaque joueur a un nom, un âge et un numéro de portable
 - chaque fan a un nom, un âge, des équipes favorites et des joueurs favoris
2. On souhaite représenter l'historique des joueurs jouant dans les équipes. Ajouter une classe époque (avec des attributs de date de début et de fin) permettant cette représentation.