

# TP1 - Utilisation d'outils pour le GL

Ce TP est à effectuer en binôme (quelques monômes acceptés). A rendre pour le 25 septembre 23h59. Par email à votre encadrant de TP.

## I – Diagramme de Gantt : planner et Ganttproject

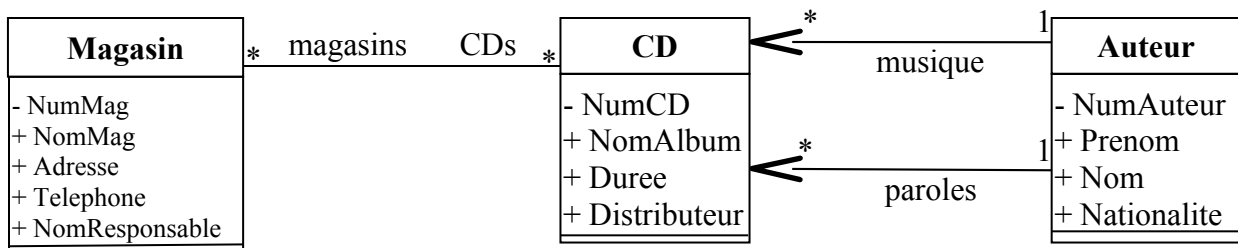
Durée 30 minutes - à finir chez vous.

1. Avec *Planner*, créez un planning sur une semaine (5 jours) pour un groupe de 3 étudiant. Quelle est la résolution/granularité temporelle la plus fine ?
2. Insérez-le dans un document Openoffice, puis exportez-le en PDF et vérifiez que le résultat produit est lisible.
3. Téléchargez *Ganttproject* et exécutez-le. Faites la même chose qu'avec *Planner*.
4. Rédigez un tableau comparatif succinct des deux outils et indiquez lequel a votre préférence.
5. Vous pouvez également télécharger et tester *MS Project* (Windows-only).

## II - Modélisation UML : Umbrello, ArgoUML et Poseidon

Durée : 1h - à finir chez vous.

1. Représenter le schéma suivant à l'aide d'*Umbrello*. Il est facile à prendre en main. Il existe des alternatives open-source (dia, gaphor, bouml, argoUML), open-source Windows-only (StarUML), ou encore des versions d'essai de logiciels commerciaux (Poseidon). Certains ne font que du dessin (Violet UML editor), d'autres peuvent générer du code à partir d'un diagramme de classe.



2. Ajouter quelques méthodes manquantes dans la modélisation.
3. Générer le code **C++** correspondant. Il sera produit dans `$HOME/uml-generated-code`. Attention, *Umbrello* génère un code qui, par défaut, ne compile pas (dans les options de génération, remplacez par exemple `string` par `std::string`)
4. Ajouter le code manquant afin d'obtenir une application et compiler (`g++ -c *.cpp`). Attention, n'écrasez-pas votre code modifié en ré-exécutant la génération de code !!
5. Une version compilable, exécutable mais incomplète des fichiers se trouve sur ma page Web (répertoire `magasin`). Vous pouvez vous en servir comme point de départ. Modifiez ensuite le `main.cpp` pour tester votre code.
6. En ligne de commande, exécutez *Poseidon\_for\_UML* puis *ArgoUML*. Comparez ces deux logiciels avec *Umbrello*.

7. Passez en revue différents outils UML. Jetez un œil à <http://open-tube.com/top-5-open-source-uml-tools/> ainsi qu'à la page Wikipedia consacrée aux outils UML. Testez au moins l'un d'entre eux.

### III – Utilisation d'Openoffice.org writer (ou de LibreOffice)

Durée 30 minutes - à finir chez vous.

1. Le but est de créer trois documents distincts (appelés par exemple : *Introduction*, *Spécifications*, et *Planification*) et de les insérer dans un document-maître (appelé *Rapport final*). Les styles du document maître seront appliqués automatiquement à tous les sous-documents.  
Pour la documentation, voir :  
[http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer\\_Guide/Methode\\_recommandee](http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer_Guide/Methode_recommandee)
2. Faites-en sorte que sur chaque page de vos documents (interne) apparaisse un en-tête d'une ligne (titre du document, numéro de version) et un pied de page (numérotation). Chaque document doit de plus posséder sa propre page de garde (où figure le titre du document, le nom des auteurs et l'historique des versions)
3. Votre document maître doit avoir une page de garde, une *table des matières*. Celle-ci doit être générée automatiquement grâce à la numérotation automatique de vos titres.
4. Sachez aussi générer un *index* en fin d'ouvrage.
5. Pour finir, apprenez à créer une *feuille de style* et à sauvegarder un *modèle* de document (voir  
[http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer\\_Guide/Introduction\\_styles](http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer_Guide/Introduction_styles) et  
[http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer\\_Guide/Travail\\_styles](http://wiki.services.openoffice.org/wiki/FR/Documentation/Writer_Guide/Travail_styles)).
6. Exportez le résultat en PDF.