

Projet C++

Session repassage septembre 2023

Enoncé :

Développer une application desktop en utilisant le **framework QT** et le **langage C++** qui permet de gérer une **garderie scolaire** via les modules ci-dessous :

Gestion des **Enseignants**

Gestion des **Elèves**

Gestion des **Etablissements**

Gestion des **Activités**

Chaque étudiant est demandé de choisir deux modules et de les implémenter.

Il est à noter qu'une entité est caractérisée par :

- Des fonctionnalités **de base** : Qui sont obligatoires et correspondent, dans le cadre de ce projet, aux CRUDs (Ajouter, Afficher, Modifier et Supprimer).
- Des fonctionnalités **avancées** : Qui représentent dans le cadre de ce projet les métiers.

Travail demandé :

Chaque étudiant doit :

- 1- Elaborer le diagramme de classes relatif à ses modules.
- 2- Elaborer les diagrammes de cas d'utilisation relatifs à ses modules.
- 3- Elaborer les MCD, MLD et MPD requis.
- 4- Implémenter les 2 CRUDs relatifs aux modules choisis.
- 5- Utiliser Git et GitHub pour la gestion des différentes versions de son application.
- 6- Implémenter les trois métiers suivants pour chaque module :
 - « recherche » et « tri » multicritères avec **au minimum 3 critères.**
 - Statistique en liaison avec le module en question comportant obligatoirement des graphiques qui seront **mis à jour** dès qu'il y aura un **changement** au niveau de la **base de données.**
 - Génération d'un exécutable.
- 7- Implémenter deux autres métiers utiles aux choix.

NB :

- La génération d'un document doit se limiter à un seul type (PDF ou word ou Excel) uniquement.
- La génération d'un PDF ne doit pas se limiter à un imprime écran. Le document doit être personnalisé.
- Chaque métier doit être **utile** dans le contexte du module.
- Il faut respecter les contraintes d'intégrité (Clé primaire/ étrangère ...)
- Utiliser une **seule** connexion à la BD.
- Utiliser obligatoirement l'SGBD **oracle**.
- Utiliser les requêtes **préparées**.
- Proposer des **GUI** ergonomiques : opter plutôt pour les « **containers** », tels que Tab Widget, Stacked widget et minimiser les « dialog ».
- Effectuer les **contrôles de saisie** nécessaires dans le **code C++** et dans la **base de données**.
- Implémenter les requêtes **dans les classes purement C++** relatives aux entités gérées par l'application.
- **Aucune requête ne doit être implémentée derrière un bouton.**

Bonne chance