

CPOA : Projet Arche

Apprentissage C++ et Développement itératif.

Arnaud SCHEFFLER - Victor DARMOIS - Elliot THIS

Informations

Vous pouvez compiler le projet avec `make` pour les versions en console
et avec `make gui` pour les itérations en interface graphique.

Le mot de passe des utilisateurs déjà présents dans l'application est "pass".

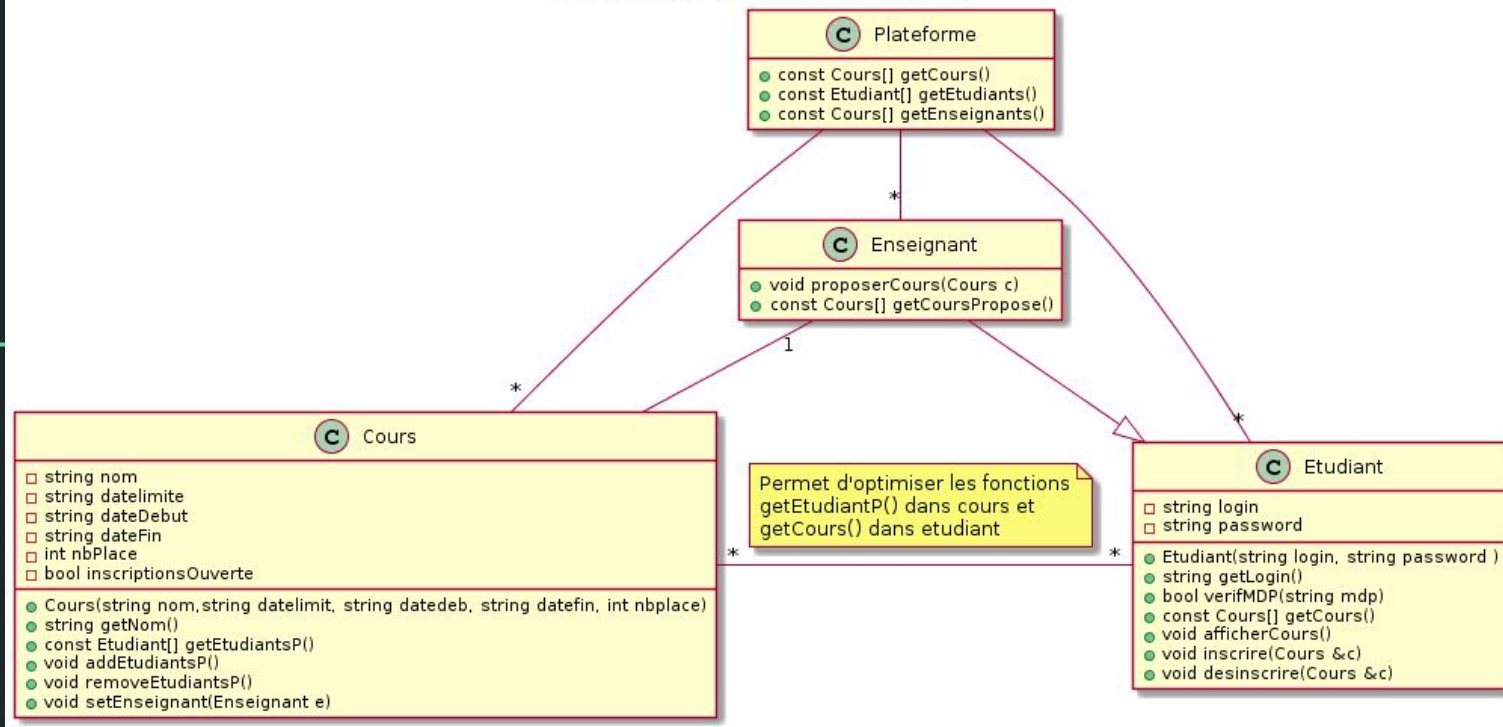
Version 5 : 5 étudiants : a, b, this4u, jo

Version 6 : 2 enseignants en plus Arnaud, Gautier

Version 7 : 2 administrateurs admin, Elliot

Iteration 1 :

Diagramme de classe : Itération 1



Iteration 1 :

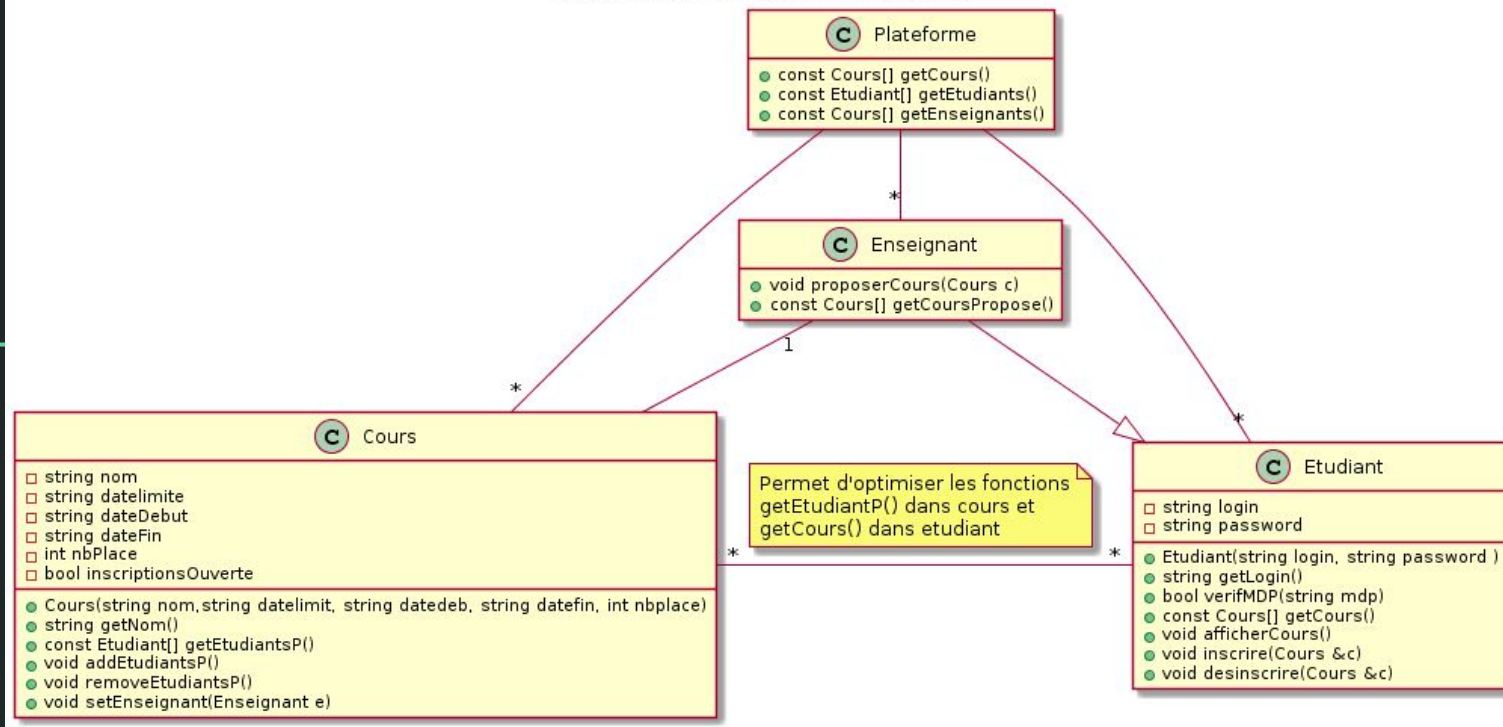
Notes

En accord avec le rendu de conception, nous avons implémenté 3 classes interdépendantes (Étudiant, Enseignant, Cours) et la classe Plateforme qui est l'entrée dans notre modèle de donnée (DP d'interface).

Nous avons aussi fait une classe de test en console pour tester en dur les implémentations de nos classes.

Iteration 2 :

Diagramme de classe : Itération 1



Iterations 2a-2b :

Notes

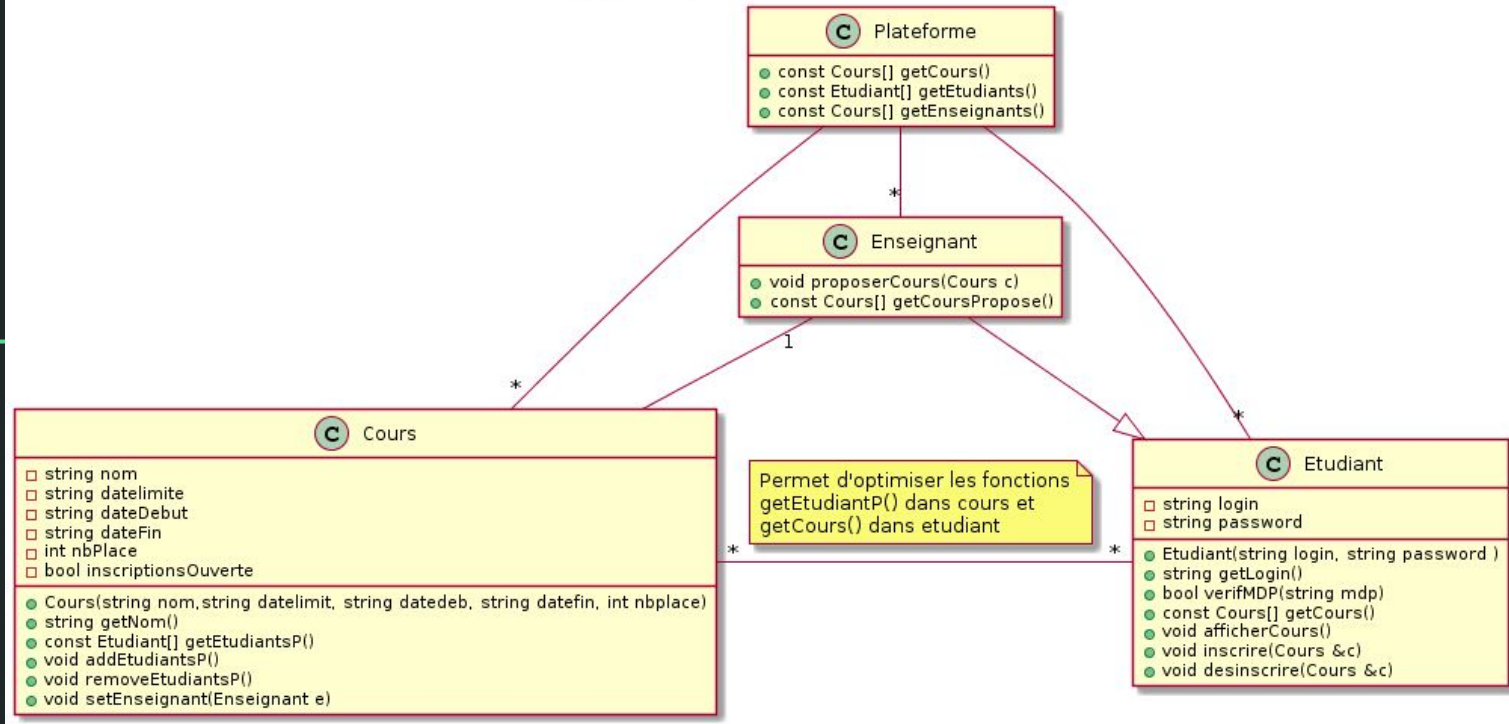
Par rapport à l'itération 1, on a ajouté un menu dans la console pour pouvoir faire des tests sur les méthodes sans les appeler les unes après les autres. Cependant au vu du gros changement du menu du fait de l'utilisation de la plateforme, on a découpé l'itération en deux parties :

- 2a menu testant les méthodes de Enseignant/Étudiant/Cours sans la plate-forme.
- 2b menu testant les classes au travers de la plate-forme.

Le diagramme de classe ne change pas.

Iteration 3 :

Diagramme de classe : Itération 1



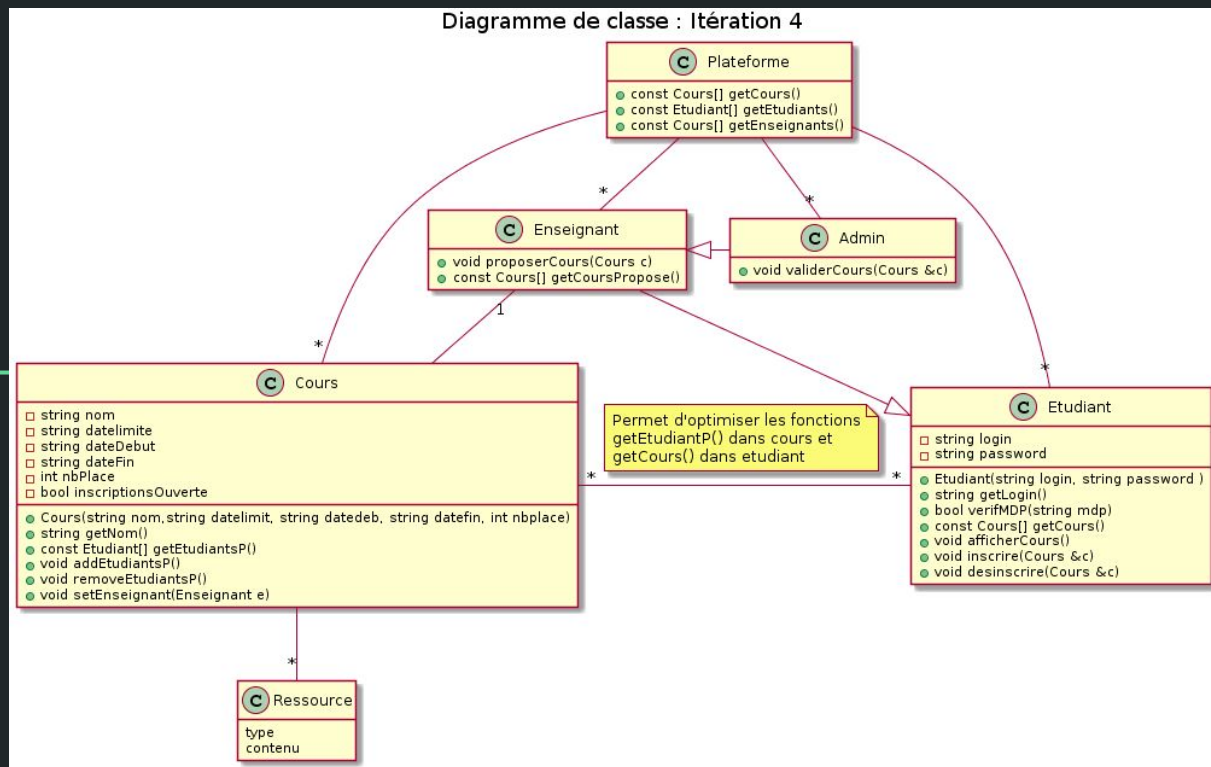
Iteration 3 :

Notes

Par rapport à l'itération 2, on a implémenté la classe Admin qui hérite de Enseignant (héritant elle même de Étudiant) ainsi que ses tests dans notre menu console. La gestion de la file principale et d'attente a été ajouté aux Cours. Un administrateur peut désormais changer l'état d'un cours en changeant un booléen.

Le diagramme de classe est mis à jour.

Iteration 4 :



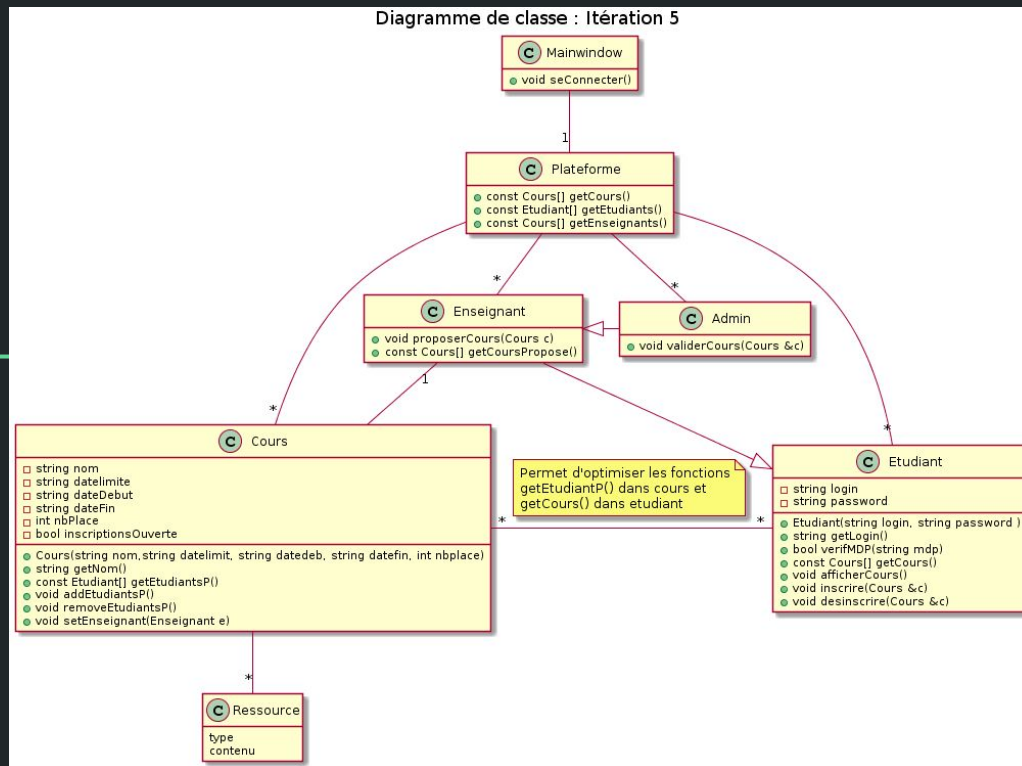
Iteration 4 :

Notes

Par rapport à l'itération 3, on a implémenté la classe Ressource et les méthodes associées dans la console pour tester.

Le diagramme de classe est mis à jour.

Iteration 5 :



Iteration 5 :

Notes

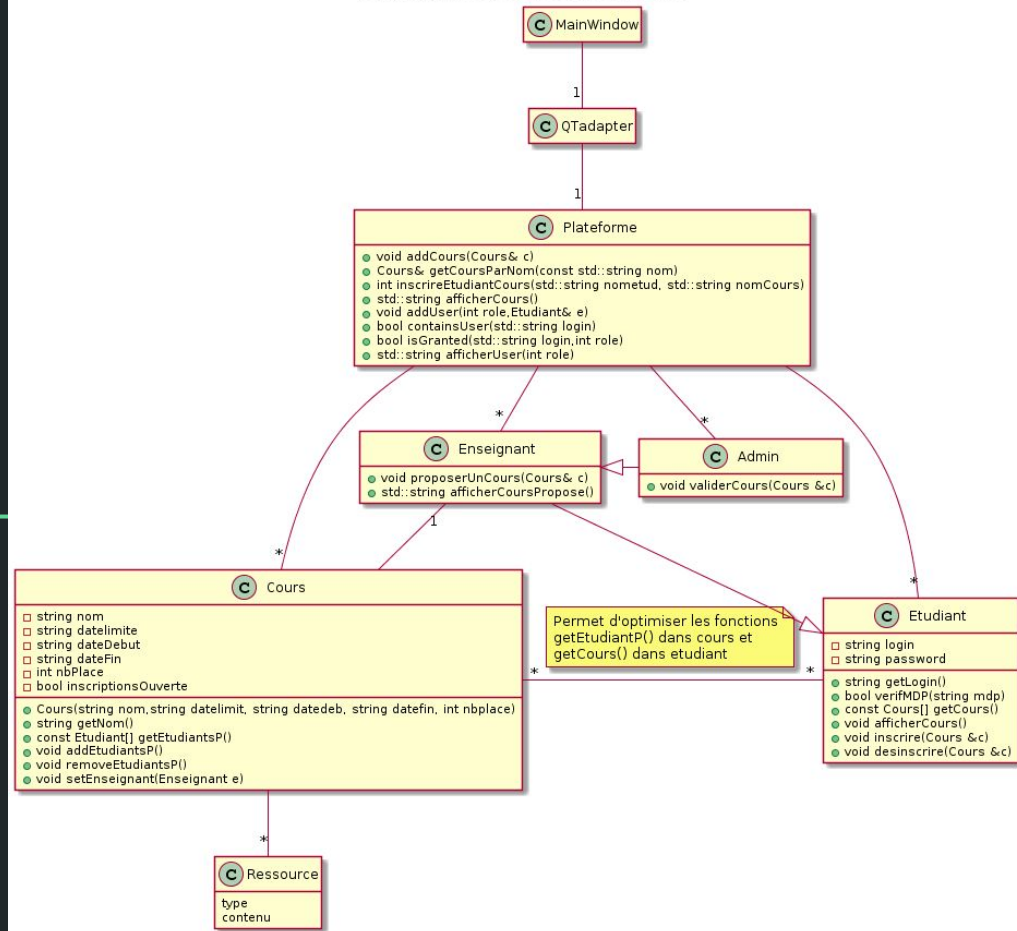
Par rapport à l'itération 4, on a commencé l'implémentation de l'interface graphique avec Qt.

Il est possible de se connecter, de sélectionner un des cours auquel on participe et de voir les listes d'inscrits et les ressources.

Le diagramme de classe a été mis à jour.

Iteration 6 :

Diagramme de classe : Itération 6



Iteration 6 :

Notes

Par rapport à l'itération 5, on a continué l'implémentation de l'interface graphique avec Qt.

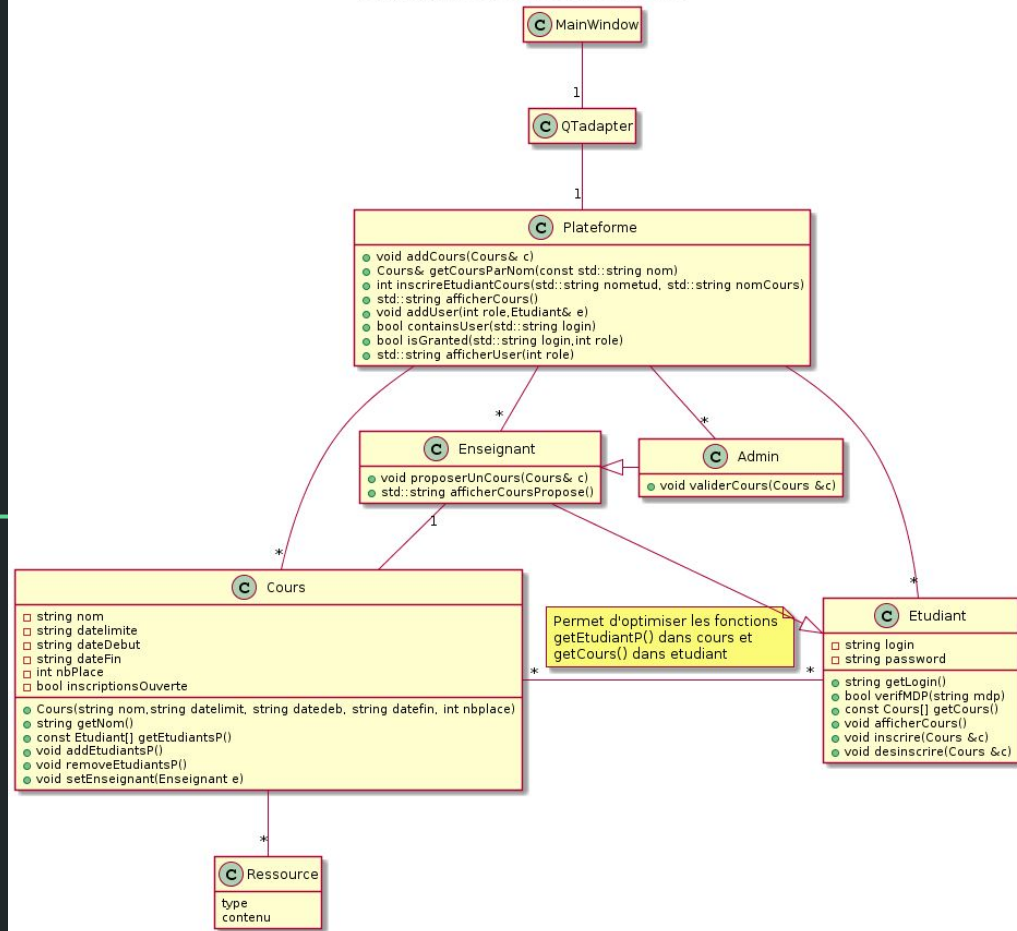
On affiche les différentes informations relatives à un Étudiant et à un Cours. Le cas de l'inscription d'un Étudiant à un cours a été ajouté. Le cas de désinscription d'un Étudiant à un cours a été ajouté. On affiche le nombre de places et l'enseignant du cours sélectionné. La liste d'attente et principale est mise à jour dynamiquement. Il a aussi été corrigé un bug lorsque l'on désinscrit un étudiant d'un cours et qu'il est le dernier de la liste principale.

Nous avons par ailleurs ajouté une classe qtadapter (Un DP Adapter) qui fait le lien entre les types propres QT et notre modèle.

Le diagramme de classe à été modifié.

Iteration 7 :

Diagramme de classe : Itération 6



Iteration 7 :

Notes

Par rapport à l'itération 6, nous avons continué l'implémentation de l'interface graphique avec Qt :

Il est désormais possible de s'inscrire en spécifiant son type de compte. Un administrateur peut valider un cours pour le rendre visible pour tout le monde. Un enseignant à sa liste de cours et peut en proposer de nouveaux qui devront être accepté par un admin. Un étudiant doit être inscrit au cours pour voir ses ressources. L'inscription/désinscription d'un étudiant à été rendu plus logique.

Le diagramme de classe n'as pas changé.