Rapport Itération numéro 3

Identification des membres de l'équipe

Assurez-vous d'utiliser toujours le même compte GitHub pour accéder à ce projet.

Membre 1

- Anna Andersen
- anna.andersen.1@ens.etsmtl.ca
- AQ80740
- aandersen-mal

Membre 2

- Thomas Beetz
- Thomas.beetz.1@ens.etsmtl.ca
- AP59120
- Thomas-Beetz-Knowledge

Membre 3

- Nick-Karl Chao
- nick-karl.chao.1@ens.etsmtl.ca
- AQ18340
- n3chao

Membre 4

- Lucas Schwing
- lucas.schwing.1@ens.etsmtl.ca
- AR77560
- Lukip2603

Membre 5

- Shi Yao Wang
- shi-yao.wang.1@ens.etsmtl.ca
- AQ87340
- syw1-art

Exigences

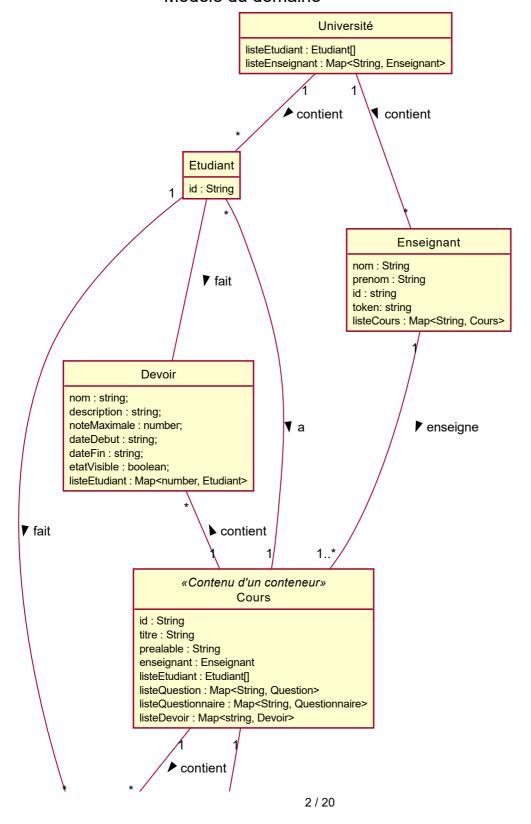
Liste des exigences et personnes responsables de celles-ci.

Exigence	Responsable
CU05a	Anna Andersen, Lucas Schwing, Thomas Beetz

Exigence	Responsable
CU05b	Anna Andersen, Shi Yao Wang, Thomas Beetz
CU05c	Shi Yao Wang, Thomas Beetz, Lucas Schwing
CU05d	Shi Yao Wang, Nick-Karl Chao, Thomas Beetz
CU03a	Anna Andersen, Thomas Beetz, Nick-Karl Chao

Modèle du domaine (MDD)

Modèle du domaine



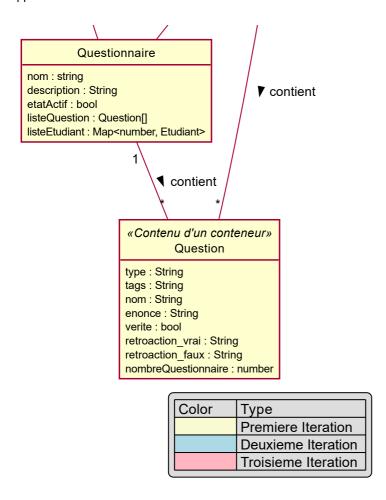
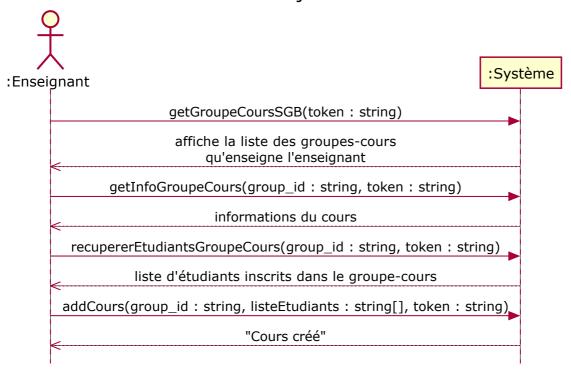


Diagramme de séquence système (DSS)

Un seul DSS sera choisi et corrigé par l'auxiliaire d'enseignement

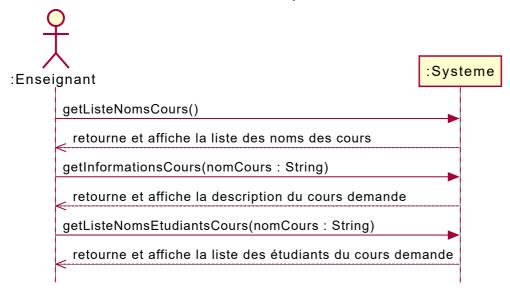
CU01a

DSS CU01a - Ajouter cours

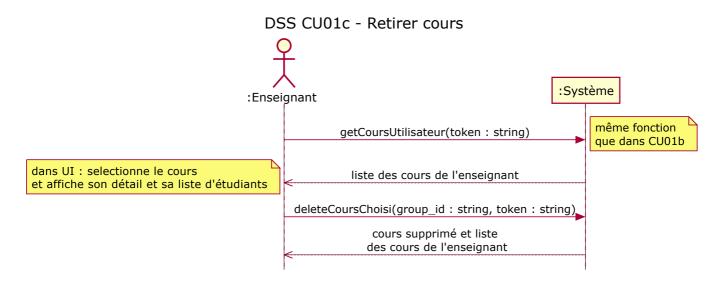


CU01b

DSS CU01b - Récupérer cours

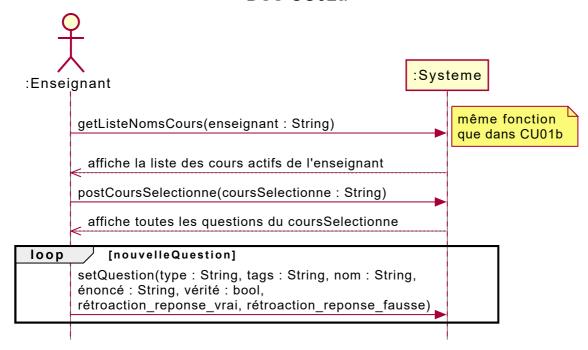


CU01c



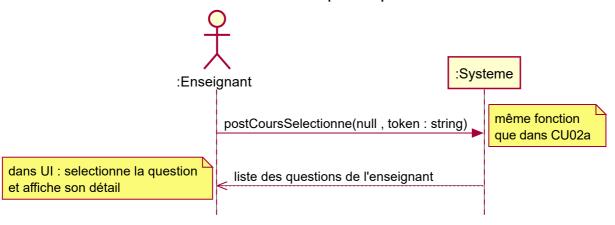
CU02a

DSS CU02a



CU02b

DSS CU02b - Récupérer question



CU02c

DSS CU02c - Modifier question

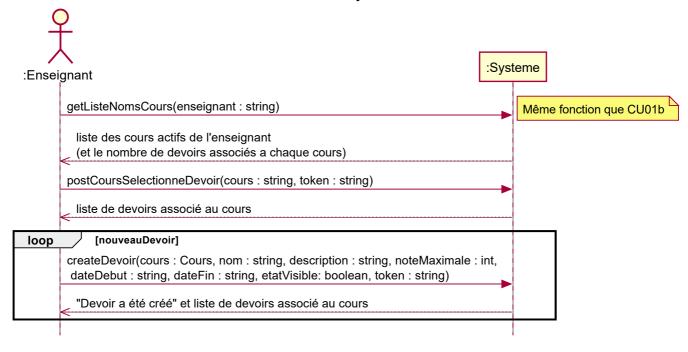


CU02d

:Enseignant :Systeme postCoursSelectionne(null , token : string) dans UI : selectionne la question et affiche son détail supprimerQuestion(nomQuestion : string, token : string) confirmation de la suppression

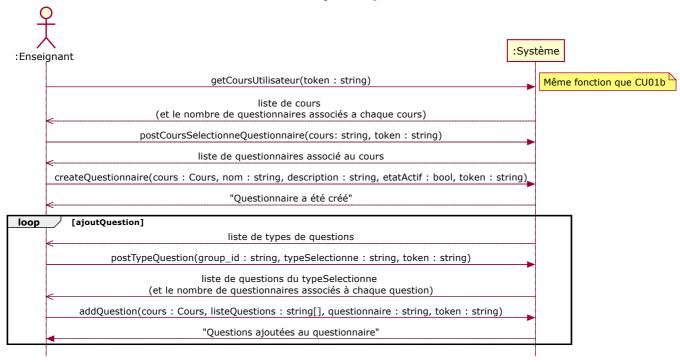
CU03a

DSS CU03a - Ajouter Devoir



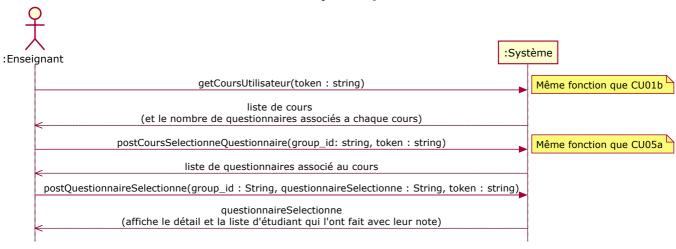
CU05a

DSS CU05a - Ajouter Questionnaire



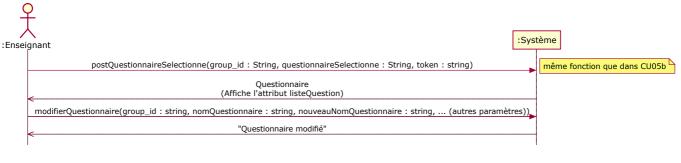
CU05b

DSS CU05b - Ajouter Questionnaire



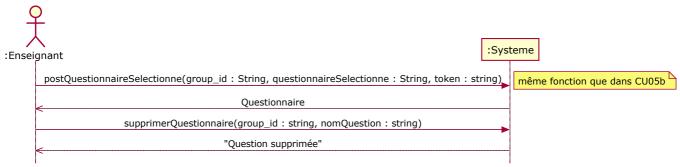
CU05c

DSS CU05c - Modifier Questionnaire



CU05d

DSS CU05d - Supprimer Question



Contrats

Si vous avez choisi un cas d'utilisation nécessitant un contrat, il faut le mettre dans cette section. Note: même s'il y a plusieurs contrats, un seul contrat sera choisi et corrigé par l'auxiliaire d'enseignement

CU01a

Opération getGroupeCours(enseignant : Enseignant)

Postconditions:

Aucune

Opération postGroupeCoursChoisi(groupeCoursChoisi : String)

Postconditions:

Aucune

Opération: setCours(groupeCoursChoisi)

Postconditions:

- une nouvelle instance c de Cours a été créée
- c a été associée a l'enseignant enseignant
- c a été associé aux groupes d'étudiants inscrits etudiant[]

CU01b

getListeNomsCours()

Postconditions:

Aucune

getInformationsCours(nomCours: String)

Postconditions:

Aucune

getListeNomsEtudiantsCours(nomCours : String)

Postconditions:

Aucune



Opération deleteGroupeCoursChoisi(groupeCoursChoisi : String)

Postconditions Le cours n'existe plus dans le système SGA



Opération getListCoursActif(enseignant : Enseignant)

Postconditions:

Aucune

Opération postCoursSelectionne (coursSelectionne : String)

Post conditions:

Aucune

Opération: setQuestion(type : String, tags : String, nom : String,énoncé : String, vérité : bool,rétroaction_reponse_vrai, rétroaction_reponse_fausse)

Postconditions:

- une nouvelle instance q de Question a été créée
- q a été associée au cours



Opération getListeQuestion(enseignant : String)

Postconditions:

Aucune

Opération postQuestionSelectionne(questionSelectionne : String)

Postconditions:

Aucune



Opération getQuestion(nomCours : string, nomQuestion : string)

Postconditions:

Aucune

Opération modifierQuestion(nomCours : string, nomQuestion : string, nouveauNomQuestion : string, ... (autres paramètres))

Postconditions : -La question a été modifiée dans la banque pour le cours



CU02d

Opération getQuestion(nomCours : string, nomQuestion : string)

Postconditions:

Aucune

Opération supprimerQuestion(nomCours : string, nomQuestion : string)

Postconditions:

• La question a été supprimée de la banque pour le cours



CU03a

Opération getListeNomsCours(enseignant : string)

Postconditions -Aucune

Opération postCoursSelectionneDevoir(cours : string, token : string)

Postconditions - Aucune

Opération createDevoir(cours : Cours, nom : string, description : string, noteMaximale : int,\n dateDebut : string, dateFin : string, etatVisible: boolean, token : string)

Postconditions -Une instance d de Devoir a été créé -Le Cours cours est associé au Devoir d



CU05a

Opération getCoursUtilisateur(token : string)

Postconditions

Aucune

Opération postCoursSelectionneQuestionnaire(cours: string, token : string)

Postconditions

Aucune

Opération postQuestionnaireSelectionne(group_id : String, questionnaireSelectionne : String, token : string)

Postconditions

Aucune



Opération postCoursSelectionneQuestionnaire(cours: string, token: string)

Postconditions

Aucune

Opération getCoursUtilisateur(token : string)

Postconditions

Aucune

Opération postQuestionnaireSelectionne(group_id : String, questionnaireSelectionne : String, token : string)

Postconditions

Aucune



Opération postQuestionnaireSelectionne(group_id : String, questionnaireSelectionne : String, token : string)

Postconditions

Aucune

Opération modifierQuestionnaire (group_id : string, nomQuestionnaire : string, nouveauNomQuestionnaire : string, ... (autres paramètres))

Postconditions:

Aucune



Opération postQuestionnaireSelectionne(group_id : String, questionnaireSelectionne : String, token : string)

Postconditions

Aucune

Opération supprimerQuestionnaire(group_id : string, nomQuestion : string)

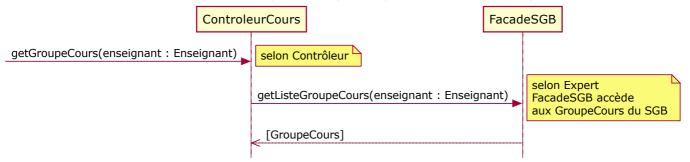
Postconditions - L'instance q de Questionnaire (dont le nomQuestion est donné en paramètre) est supprimé

Réalisation de cas d'utilisation (RDCU)

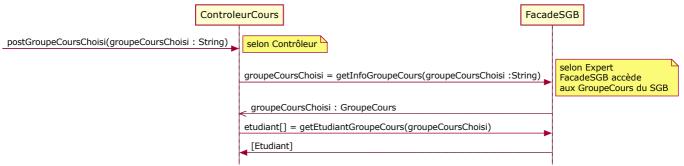
Chaque cas d'utilisation nécessite une RDCU. Note: une seule RDCU sera choisie et corrigée par l'auxiliaire d'enseignement

CU01a

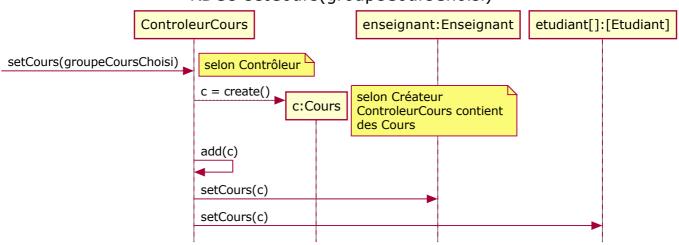
RDCU getGroupeCours(enseignant : Enseignant)



RDCU postGroupeCoursChoisi(groupeCoursChoisi: String)

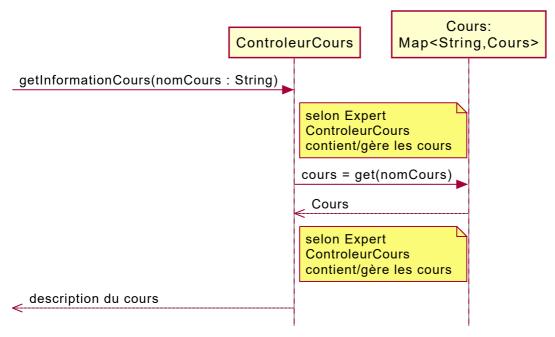


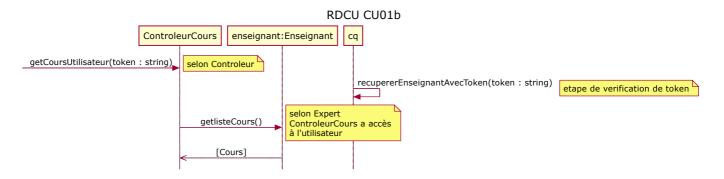
RDCU setCours(groupeCoursChoisi)



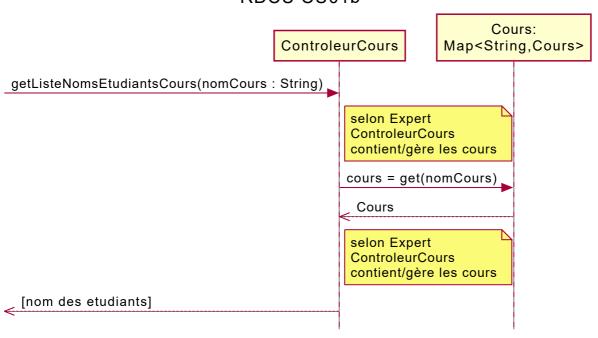
CU01b

RDCU CU01b

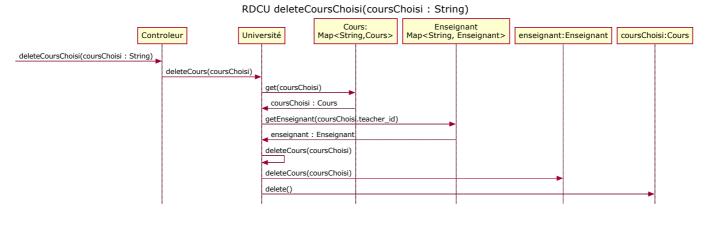




RDCU CU01b

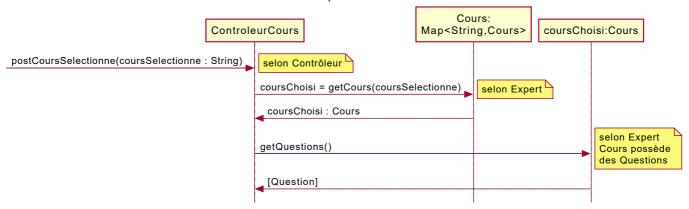


CU01c

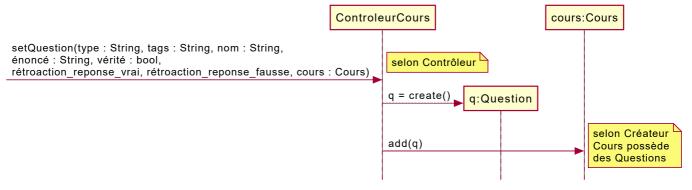


CU02a

RDCU CU02a - postCoursSelectionne

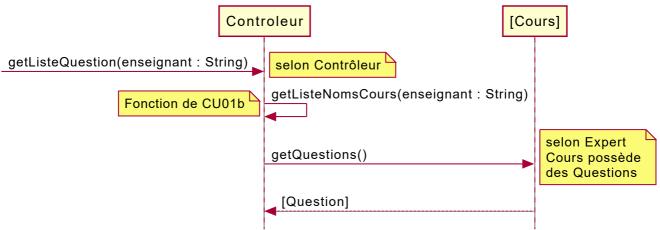


RDCU CU02a - setQuestion

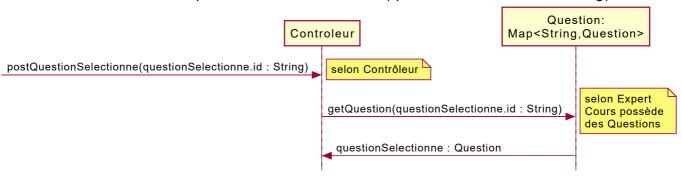


CU02b

RDCU CU02b - getListeQuestion(enseignant : String)

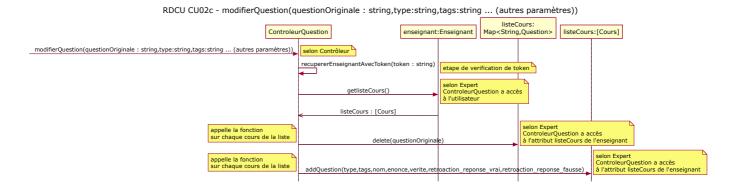


RDCU CU02b - postQuestionSelectionne(questionSelectionne : String)

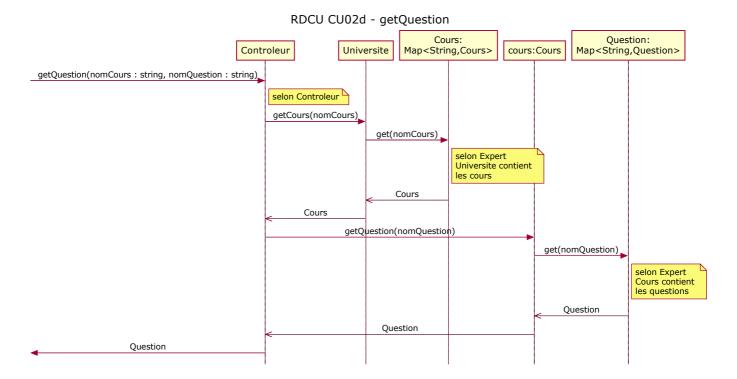


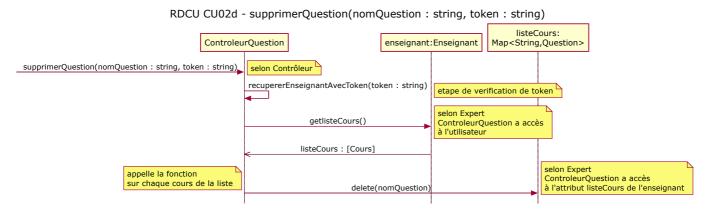
CU02c

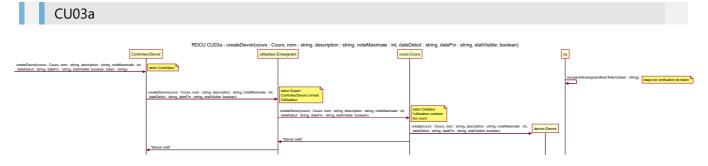
RDCU CU02c - getQuestion Cours: Map<String,Cours> Question: Map<String,Question> Universite Controleur cours:Cours getQuestion(nomCours : string, nomQuestion : string) selon Controleur getCours(nomCours) get(nomCours) selon Expert Universite contient les cours Cours getQuestion(nomQuestion) get(nomQuestion) selon Expert Cours contient les questions Question Question Question

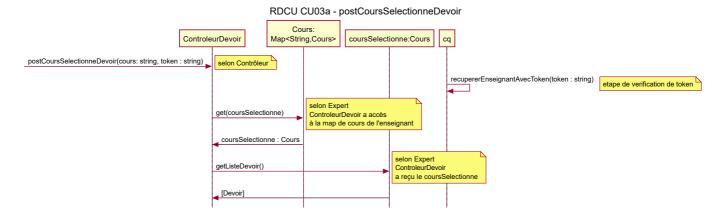


CU02d



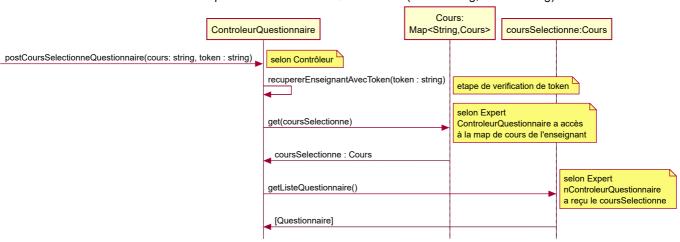


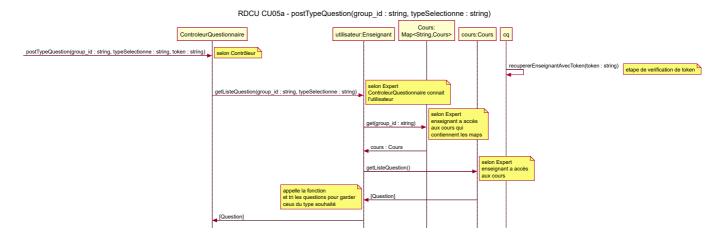




CU05a

RDCU CU05a - postCoursSelectionneQuestionnaire(cours: string, token: string)





CU05b

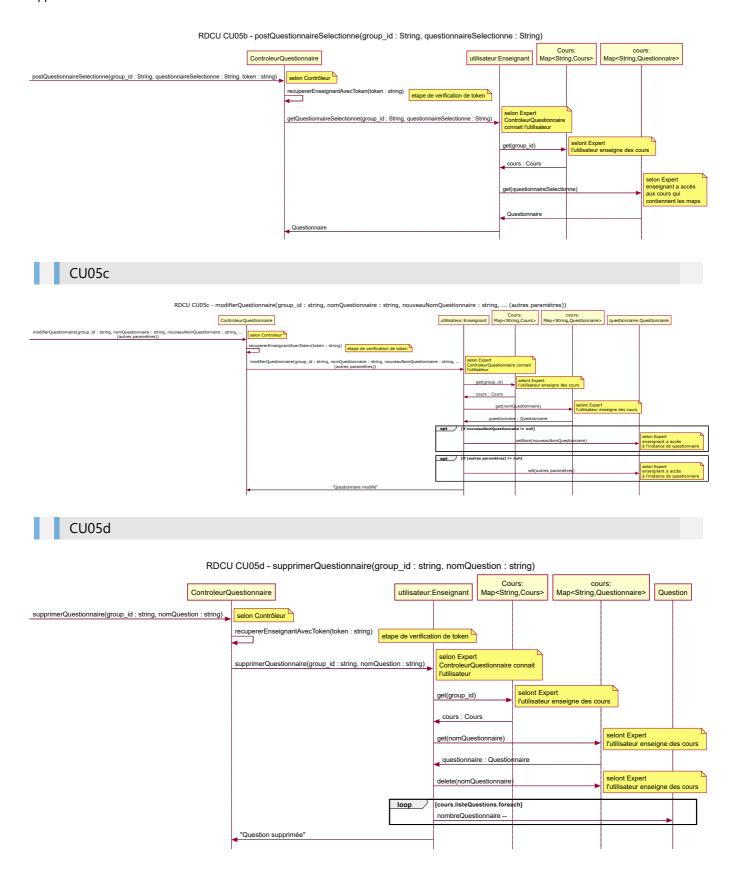
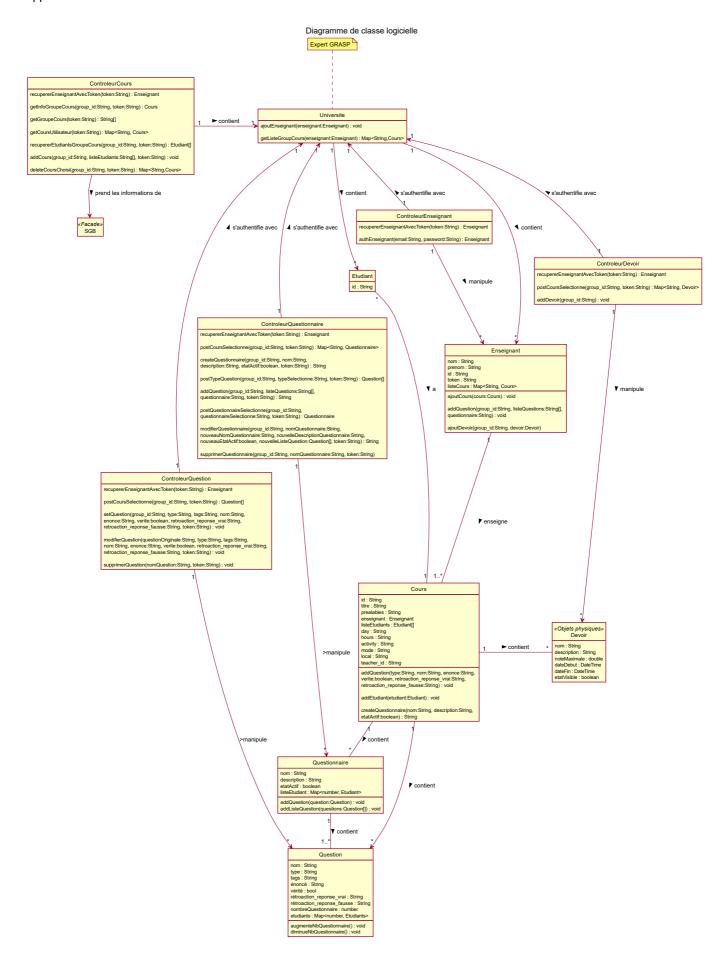


Diagramme de classe logicielle (DCL)

Facultatif, mais fortement suggéré Ce diagramme vous aidera à planifier l'ordre d'implémentation des classes. Très utile lorsqu'on utilise TDD.



Vérification finale

- Vous avez un seul MDD
 - ∘ ✓ Vous avez mis un verbe à chaque association

- Chaque association a une multiplicité
- Vous avez un DSS par cas d'utilisation

 - Chaque opération synchrone a un retour d'opération
 - L'utilisation d'une boucle (LOOP) est justifiée par les exigences
- Vous avez autant de contrats que d'opérations système (pour les cas d'utilisation nécessitant des contrats)
 - Les postconditions des contrats sont écrites au passé
- Vous avez autant de RDCU que d'opérations système
 - Chaque décision de conception (affectation de responsabilité) est identifiée et surtout **justifiée** (par un GRASP ou autre heuristique)
 - o ✓ Votre code source (implémentation) est cohérent avec la RDCU (ce n'est pas juste un diagramme)
- Vous avez un seul diagramme de classes
- Vous avez remis la version PDF de ce document dans votre répertoire
- Vous avez regardé cette petite présentation pour l'architecture en couche et avez appliqué ces concepts