

# Rapport Itération numéro 2

---

## Identification des membres de l'équipe

Veuillez éditer ce fichier afin de fournir les informations nécessaires à votre évaluation.

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

### Membre 1

- Philippe Bolduc
- philippe.bolduc.3@ens.etsmtl.ca
- AP85660
- PhilippeBolduc-hub

### Membre 2

- Ishak Megatli
- ishak.megatli.1@ens.etsmtl.ca
- AS96830
- IshakMegatli1

### Membre 3

- Daniel Atik
- daniel.atik.1@ens.etsmtl.ca
- AT56880
- DanielAtik1

### Membre 4

- Marc-Sheldon Bazelais
- marc-sheldon.bazelais.1@ens.etsmtl.ca
- AQ00910
- msbazelais

## Exigences

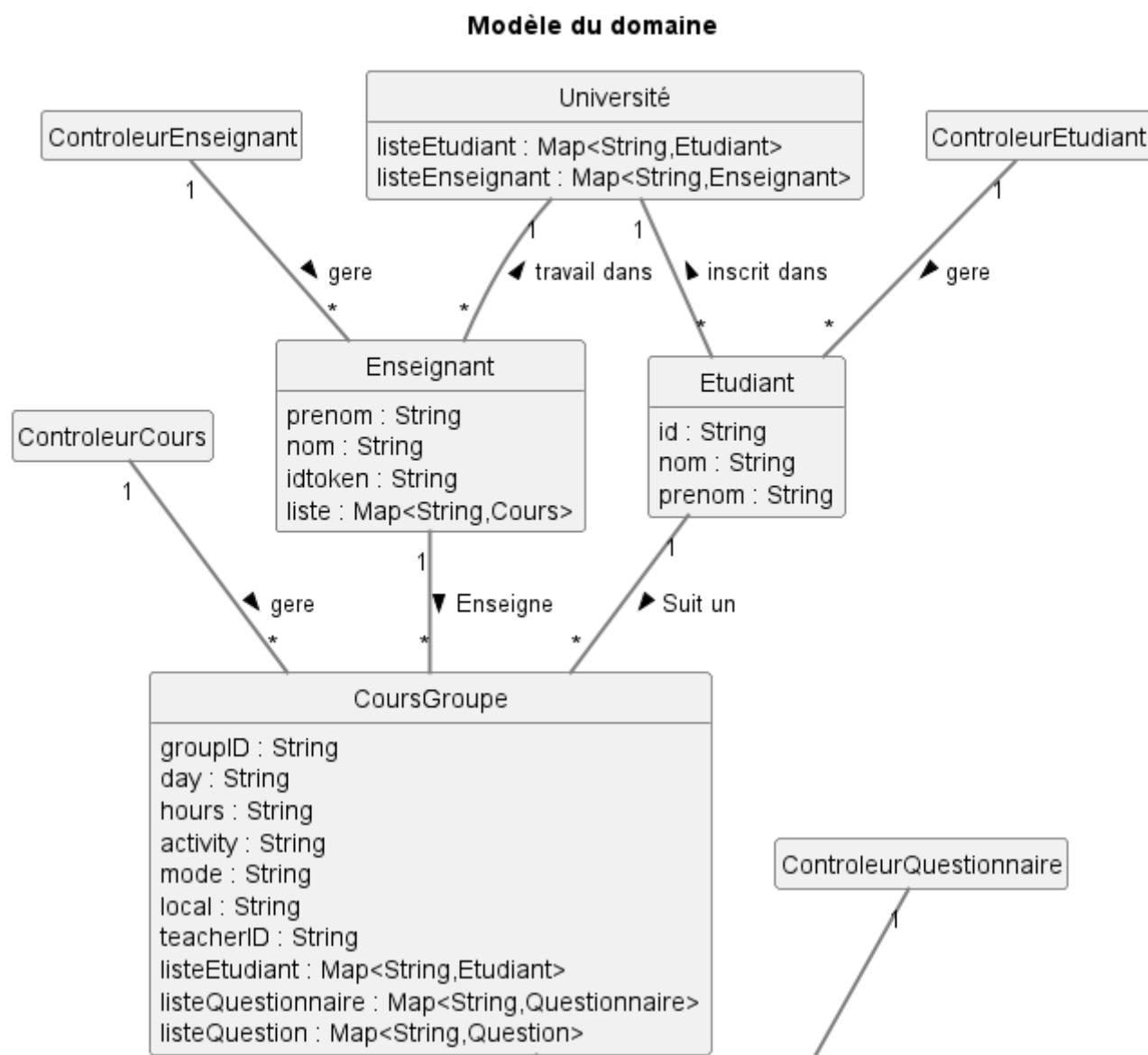
Liste des exigences et personnes responsables de celles-ci.

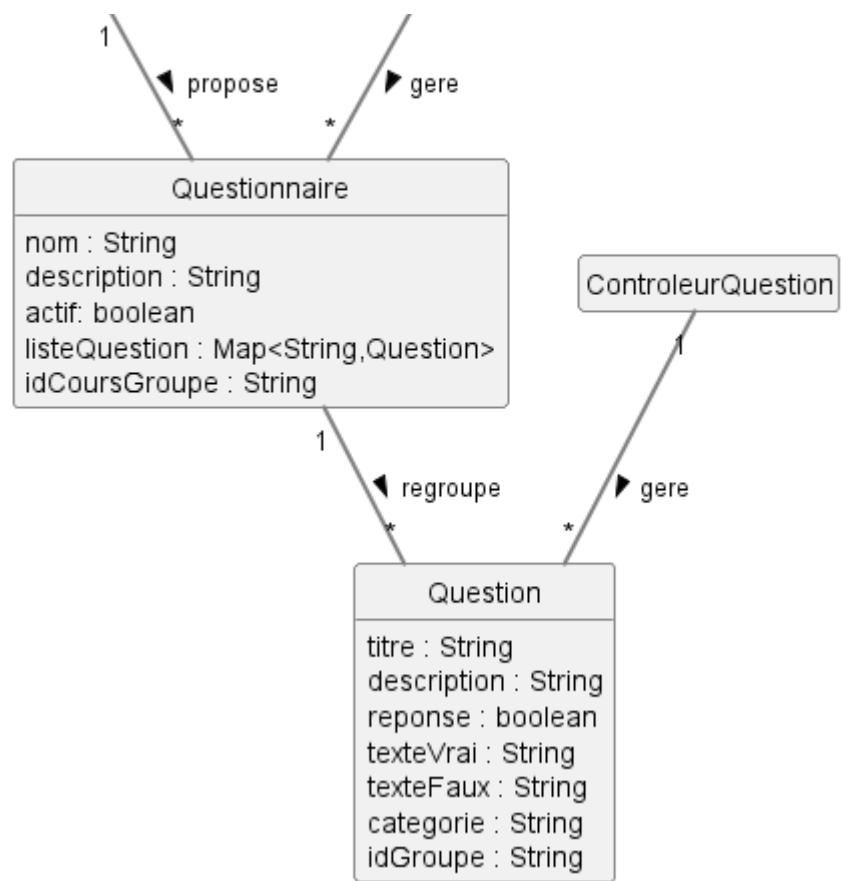
Exigence	Responsable
CU05a conception	Philippe / Marc-Sheldon
CU05a analyse	Philippe / Marc-Sheldon
CU05a révision de modeles	Philippe Bolduc
CU05a implémentation	Marc-Sheldon

Exigence	Responsable
CU05a tests	Marc-Sheldon
CU05b conception	Daniel / Philippe
CU05b analyse	Daniel / Philippe
CU05b révision de modeles	Daniel / Philippe
CU05b implémentation	Ishak / Daniel
CU05b tests	Ishak / Daniel

## Modèle du domaine (MDD)

Le MDD est cumulatif : vous devez y ajouter des éléments à chaque itération (ou corriger les erreurs), selon la portée (et votre meilleure compréhension du problème) visée par votre solution. Utilisez une légende dans le MDD pour indiquer la couleur de chaque itération afin de faire ressortir les changements (ce n'est pas toujours possible pour les associations et les attributs). Voir les stéréotypes personnalisés : <https://plantuml.com/fr/class-diagram> et [comment faire une légende avec couleurs en PlantUML](#).

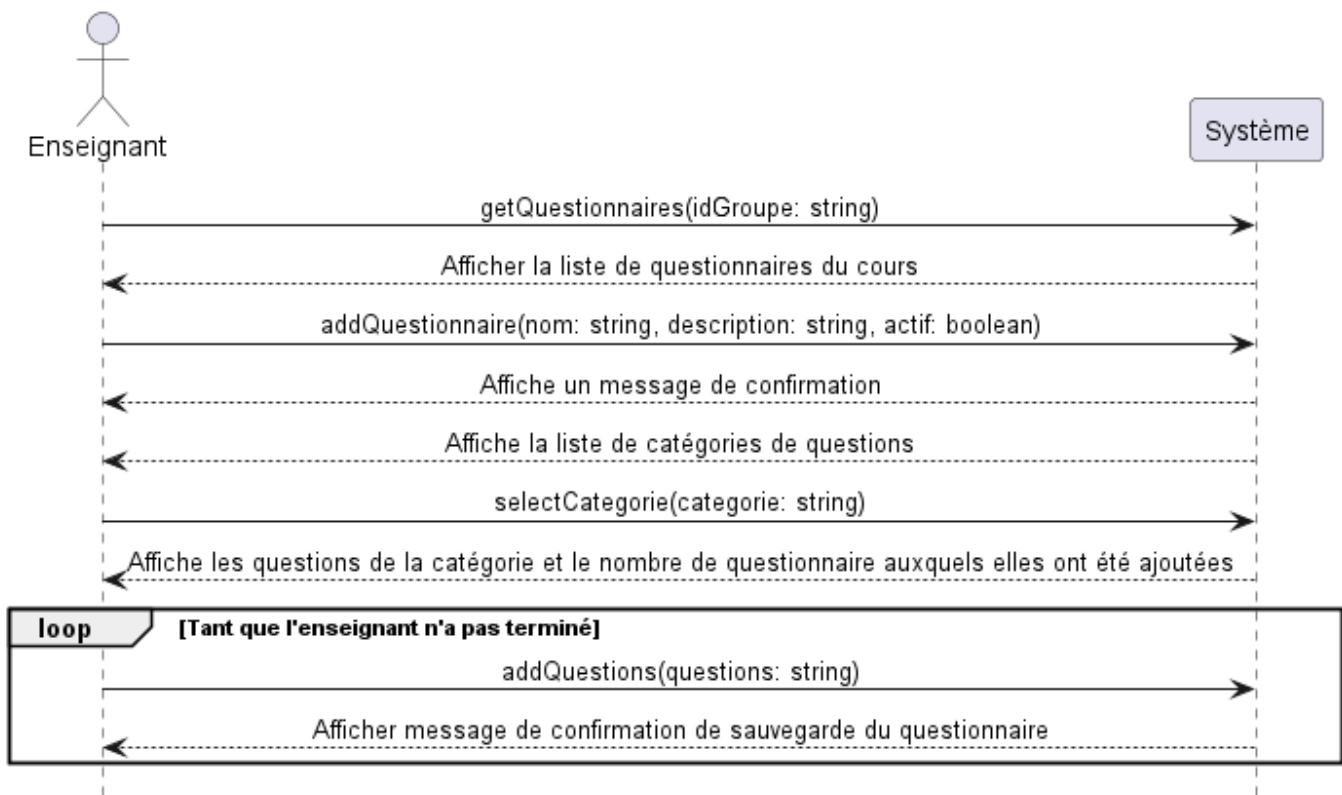




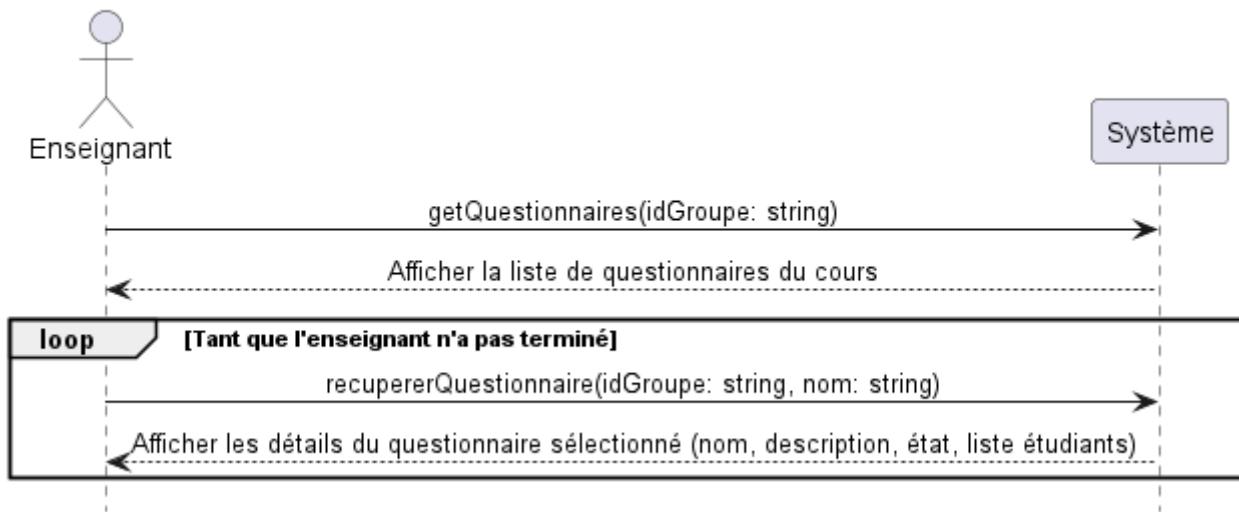
## Diagramme de séquence système (DSS)

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

### DSS : CU05a - Ajouter questionnaire



### DSS : CU05b - Afficher questionnaire



## 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  
Note: il n'est pas nécessaire de mettre les préconditions mais je vous suggère fortement de les ajouter dans votre rapport.

CU05a1 Opération : addQuestionnaire()

Post Conditions :

-Un nouveau questionnaire a été créé -Le questionnaire courant a été associé au cours courant

## CU05a2 / CU05b1 Opération : getQuestionnaire()

Post Conditions :

- La liste de questionnaires du groupe courant a été affichés

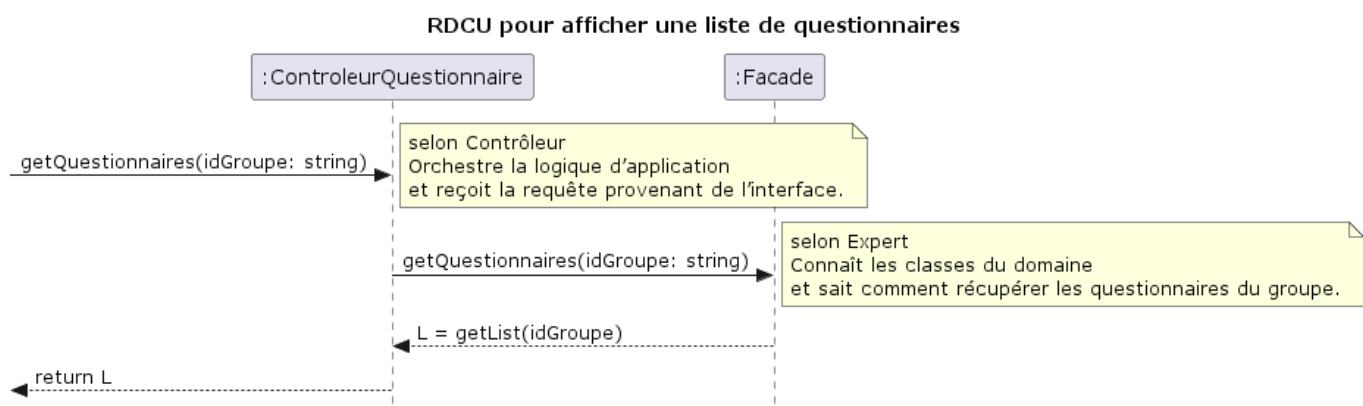
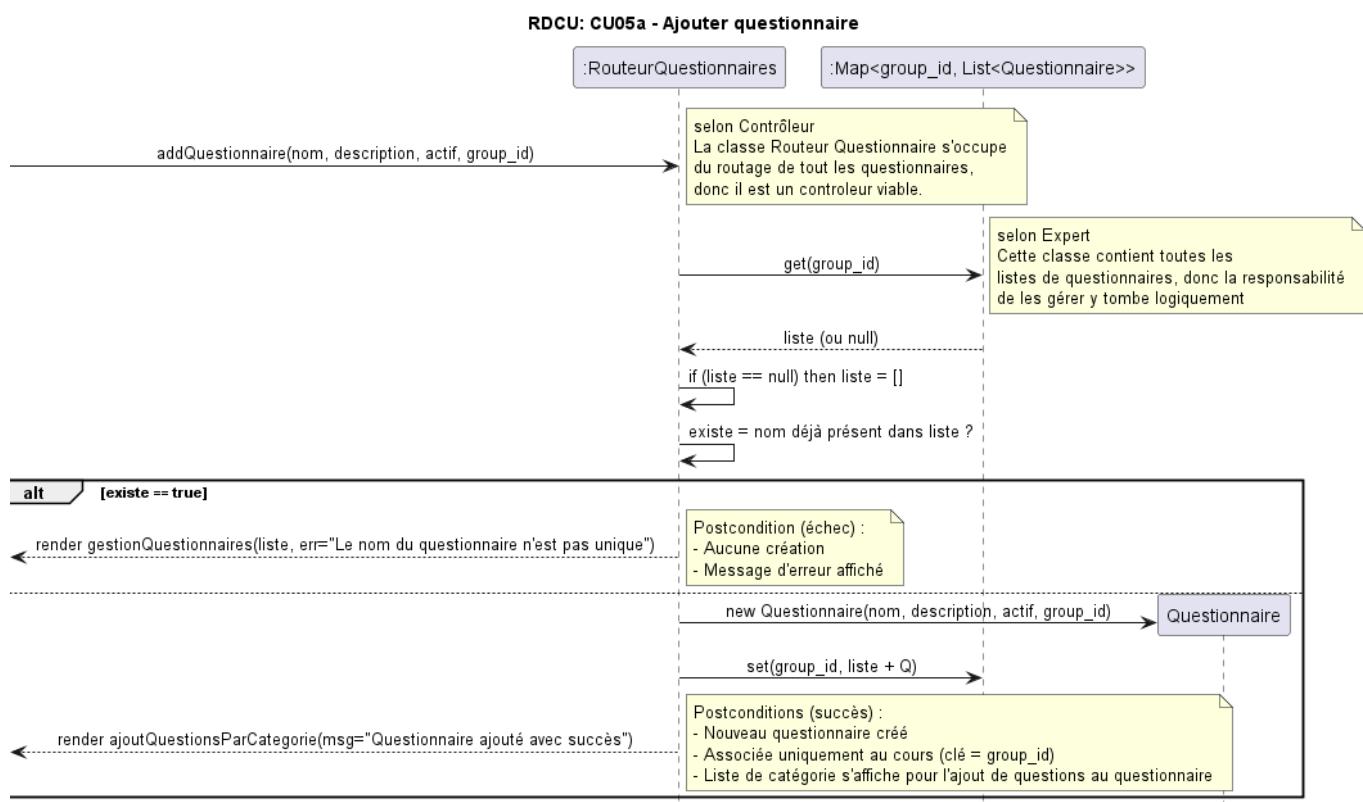
## CU05b2 Opération : recupererQuestionnaire()

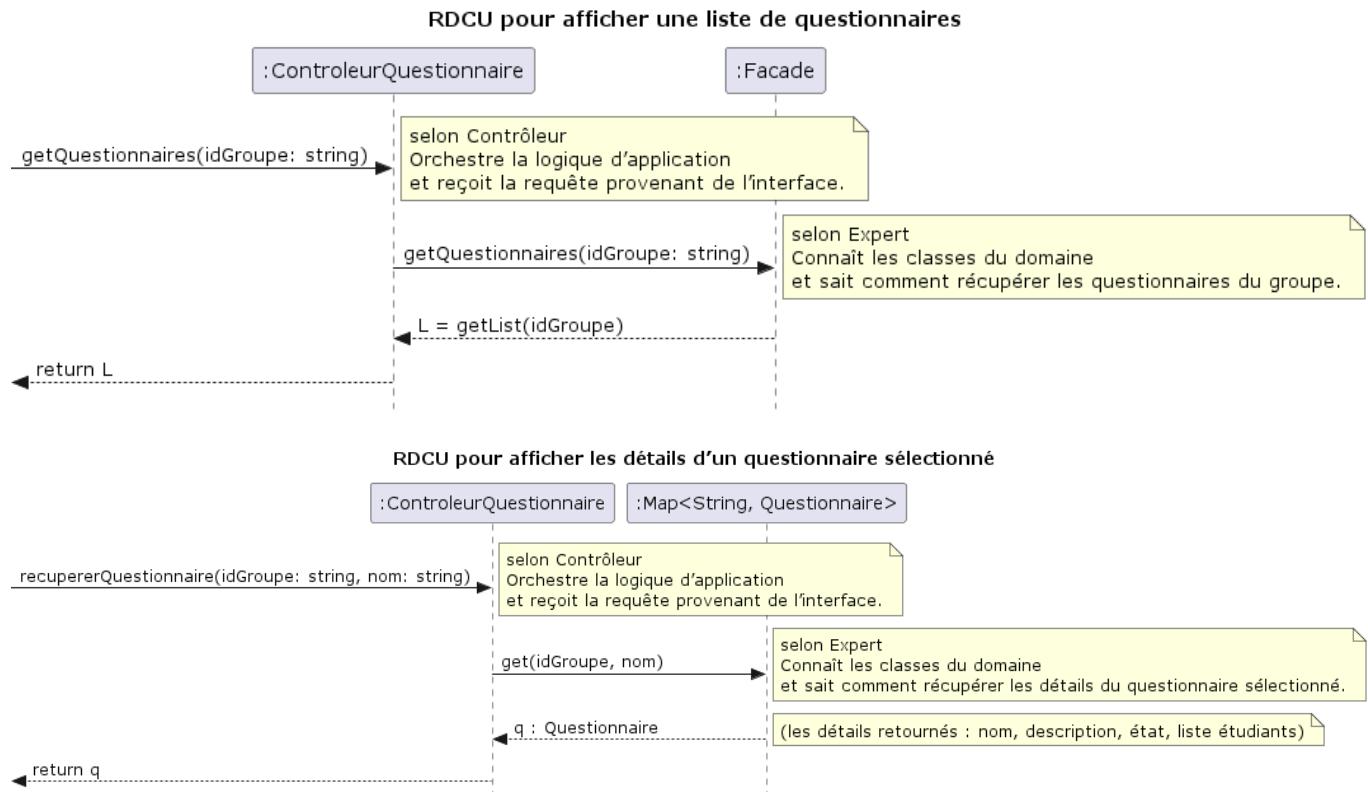
Post Conditions :

- Les attributs du questionnaire courant ont été affichés

## 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 Suivez les directives d'implémentation dans le fichier README.md pour vous faciliter la tâche d'implémentation.





\*\*\* Nous n'avons pas de RDCU pour les opérations systèmes selectCategories() car c'est simplement une sélection sur l'interface

## Diagramme de classe logiciel (DCL)

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

## Diagramme de classe TPLANT

- Générer un diagramme de classe avec l'outil TPLANT et commenter celui-ci par rapport à votre MDD.
- <https://www.npmjs.com/package/tplant>

## Retour sur la correction du rapport précédent

Démontrer avec des preuves à l'appui que vous avez réglé les problèmes identifiés dans le rapport de l'itération précédente.

## 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 DSS a un titre
  - 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)
  - 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