

# Rapport itération 2 - équipe 3

---

LOG210-2021-été-g01

Coéquipiers

- Bédard, Tommy, AQ13400
- Nom, Prénom, Code universel

Chargés de laboratoire

- Valère K. Fami

## Grille de correction

---

► Cliquez ici pour lire les consignes obligatoires

- Tous vos diagrammes doivent être faits avec <https://plantuml.com/fr/>
- Les diagrammes doivent être visibles dans ce rapport
- Supprimer les textes explicatifs du gabarit (sauf ces consignes-ci)
- Vous devez exporter ce fichier en format PDF et l'ajouter dans votre dépôt

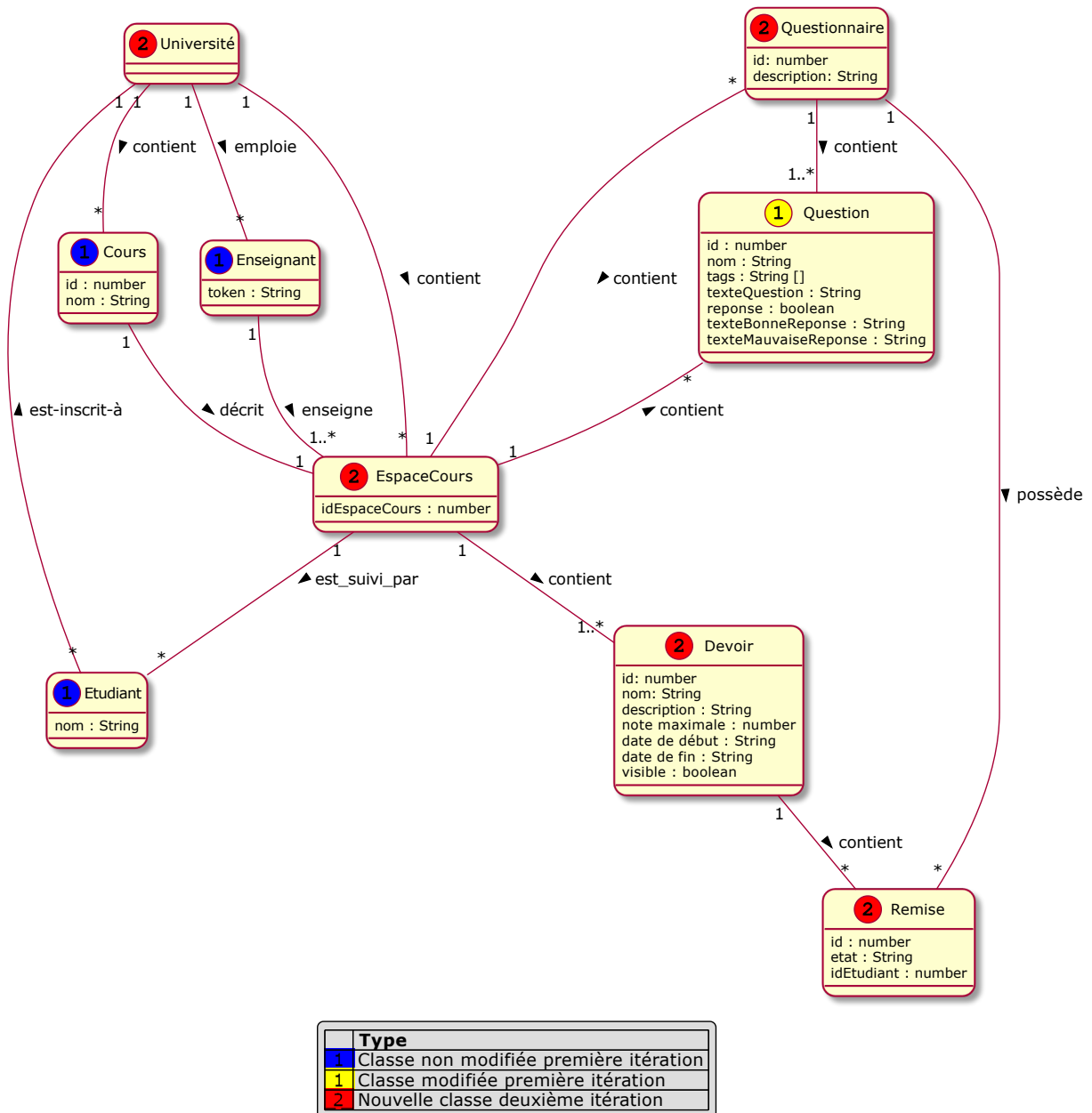
## Introduction

---

## Modèle du domaine

---

### Modèle du domaine (adapté au Cas d'utilisation ajouter et récupérer )



## CU04a - Ajouter devoir

**Acteur principal:** Enseignant

**Préconditions:**

- L'enseignant est authentifié.

**Garanties en cas de succès (postconditions):**

- Un nouveau devoir est créé et associé à un cours

**Scénario principal (succès):**

1. L'enseignant commence la création d'un devoir
2. Le système affiche les cours de l'enseignant ainsi que le nombre de devoirs associés à chaque cours
3. L'enseignant sélectionne un cours

4. Le système affiche tous les devoirs associés au cours.
5. L'enseignant crée un nouveau devoir avec une description et un nom, une note maximale, une date de début, une date de fin et un état visible ou non.
6. Le système confirme l'ajout du devoir et affiche tous les devoirs associés au cours.

On répète les étapes 5-6 tant qu'il y a un devoir à ajouter

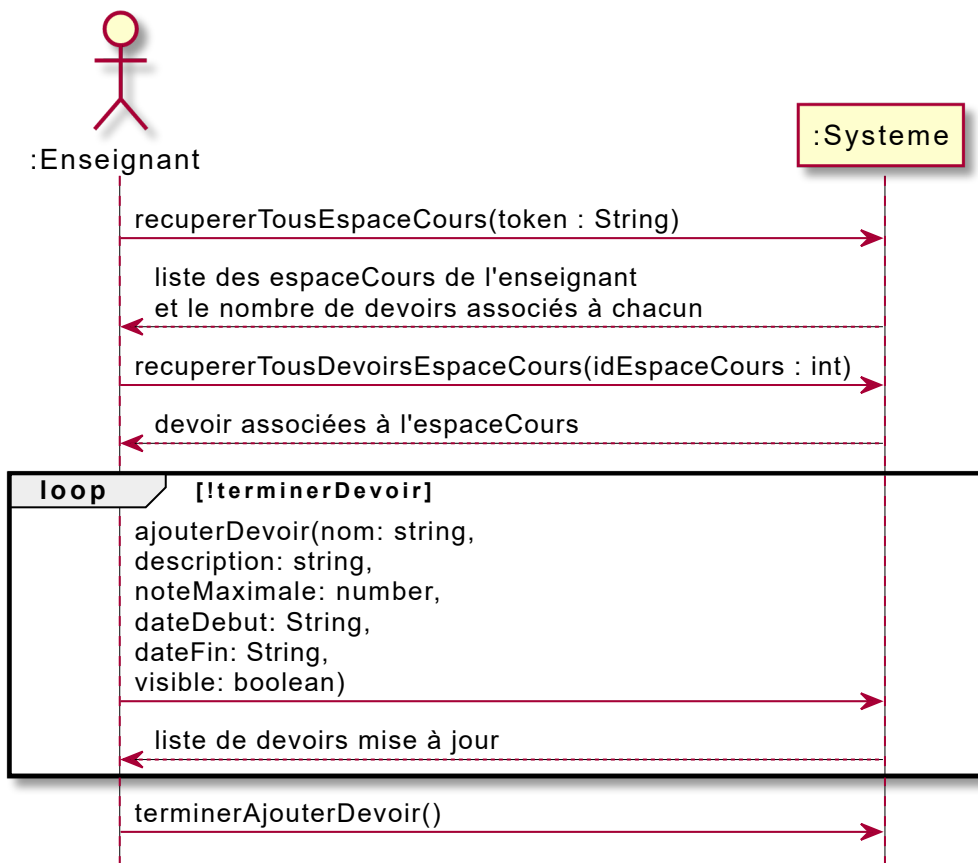
### Extensions (ou scénarios alternatifs):

5a. La date de début est après la date de fin.

1. Le Système signale l'erreur et rejette la saisie.

## DSS CU04a - Ajouter devoir

### DSS Ajouter Devoir



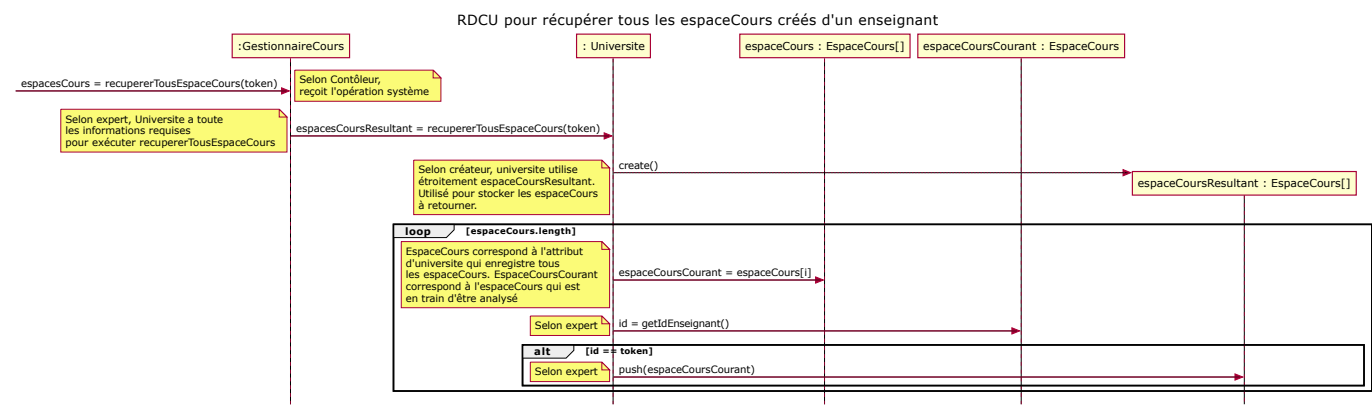
`recupererTousEspaceCours(token : String)`

### Contrat d'opération

*PostCondition*

- Aucun

### RDCU



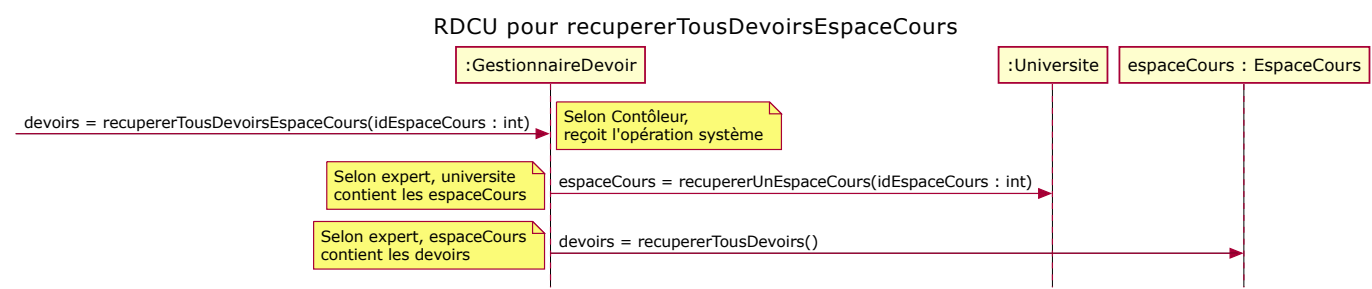
recupererTousDevoirsEspaceCours(idEspaceCours : number)

Contrat d'opération

PostCondition

- Aucune.

RDCU



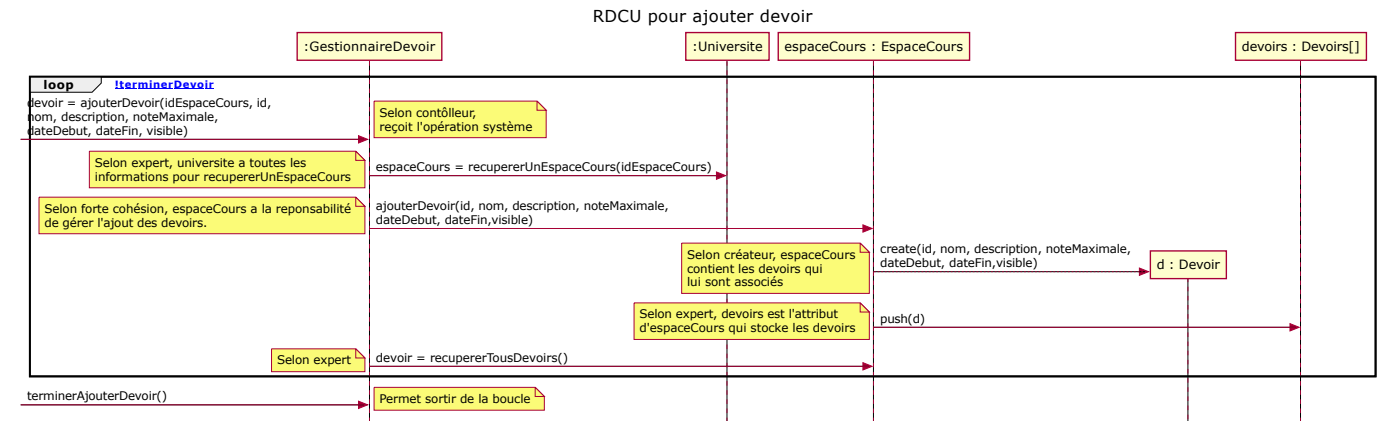
ajouterDevoir(nom: string, description: string, noteMaximale: number, dateDebut: String, dateFin: String, visible: boolean)

Contrat d'opération

PostCondition

- Un nouveau devoir d a été créé
- Les attributs de d ont été initialisés
- d a été associé à un espaceCours sur la base de idEspaceCours

RDCU



## CU04b - Récupérer devoir

**Acteur principal:** Enseignant

**Préconditions:**

- L'enseignant est authentifié.

**Garanties en cas de succès (postconditions):**

- Aucune

**Scénario principal (succès):**

1. L'enseignant commence la récupération d'un devoir
2. Le système affiche les cours de l'enseignant ainsi que le nombre de devoirs associés à chaque cours
3. L'enseignant sélectionne un cours
4. Le système affiche tous les devoirs associés au cours.
5. L'enseignant sélectionne un devoir
6. Le système affiche le détail du devoir
7. Le système affiche la liste des étudiants ayant fait le devoir ainsi que la note leur étant associée.

On répète les étapes 5 à 7 tant que l'enseignant n'a pas terminé.

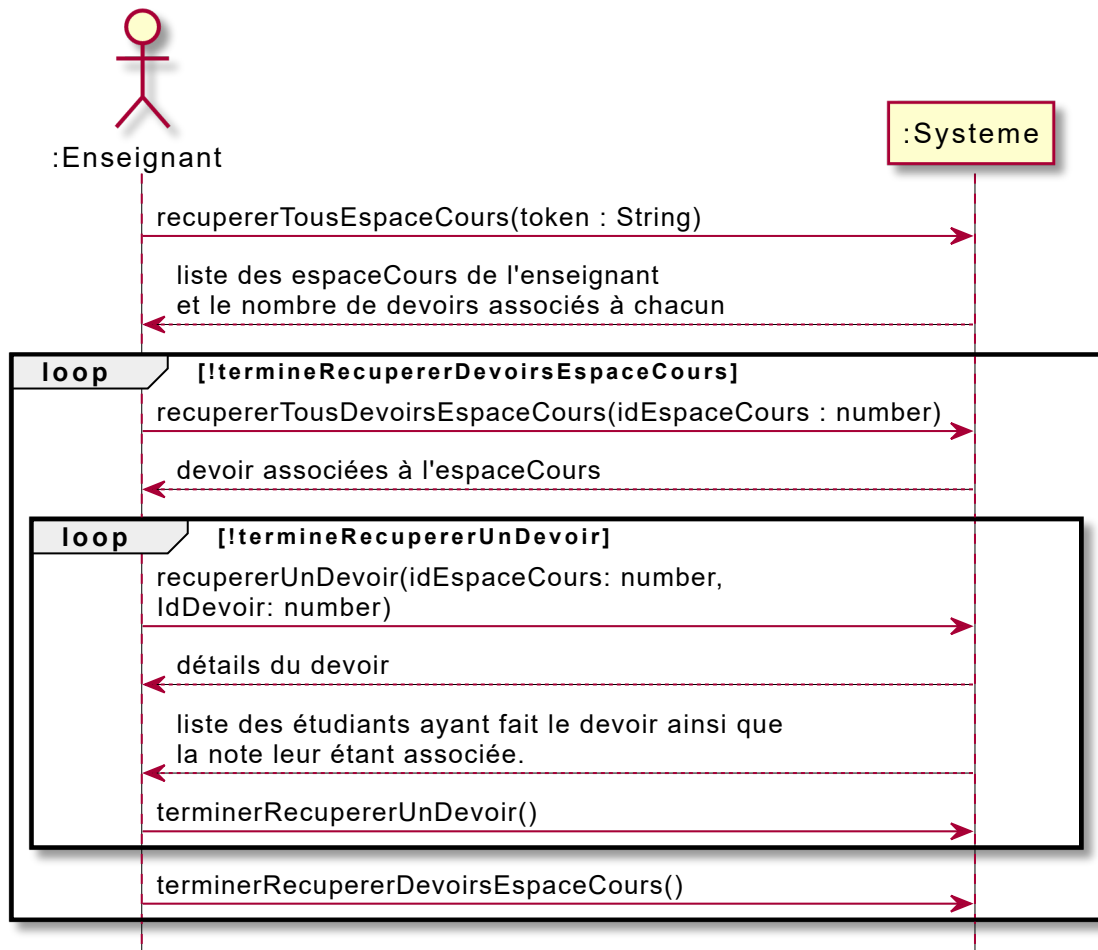
On répète les étapes 3 à 7 tant que l'enseignant n'a pas terminé.

**Extensions (ou scénarios alternatifs):**

- 7a. Le système affiche les étudiants par ordre alphabétique.
- 7b. Le système affiche les étudiants par ordre croissant de la note.

## DSS CU04b - Récupérer devoir

## DSS recupererDevoir



`recupererTousEspaceCours(token : String)`

`recupererTousEspaceCours`

`recupererTousDevoirsEspaceCours(idEspaceCours : number)`

`recupererTousDevoirsEspaceCours`

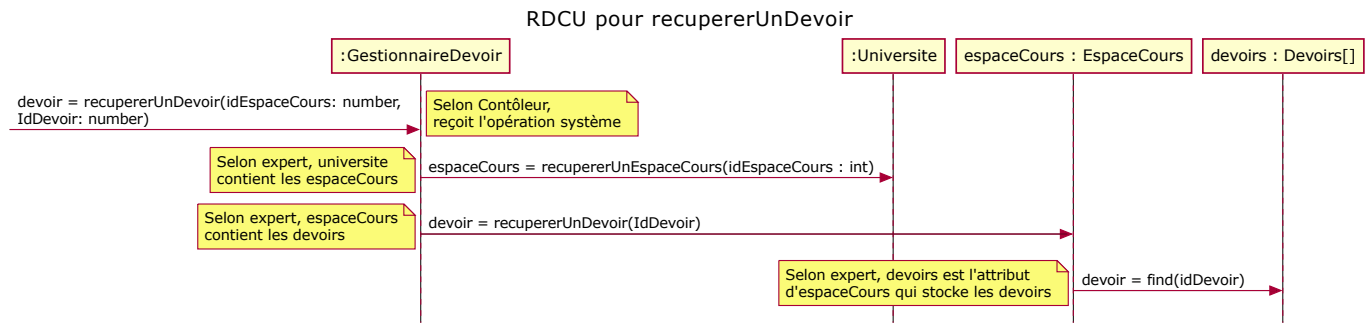
`recupererUnDevoir(idEspaceCours: number, IdDevoir: number)`

### Contrat d'opération

*PostCondition*

- Aucune.

### RDCU



## CU04c - Modifier devoir

**Acteur principal:** Enseignant

**Préconditions:**

- L'enseignant est authentifié.

**Garanties en cas de succès (postconditions):**

- Un devoir est modifié

**Scénario principal (succès):**

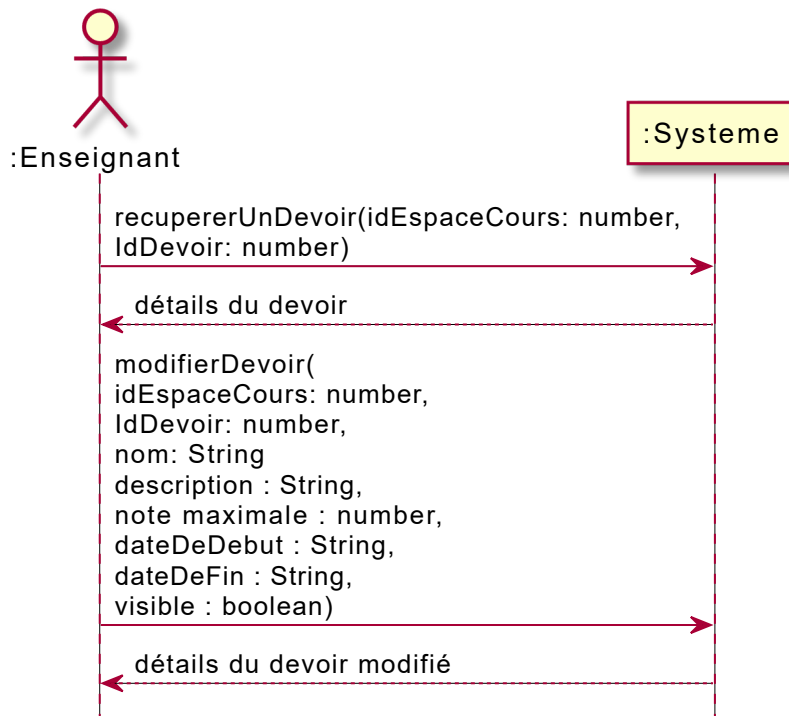
1. L'enseignant commence la modification d'un devoir
2. Le système affiche les valeurs actuelles du devoir à modifier.
3. L'enseignant effectue les modifications du devoir et les sauvegarde.
4. Le système affiche tout le devoir modifié

**Extensions (ou scénarios alternatifs):**

- 1a. Un devoir ne peut pas être modifié si des étudiants ont déjà commencé à réaliser celui-ci.

## DSS CU04b - Modifier devoir

## DSS modifierDevoir



`recupererUnDevoir(idEspaceCours: number, IdDevoir: number)`

RécupérerUnDevoir

`modifierDevoir(idEspaceCours: number, IdDevoir: number)`

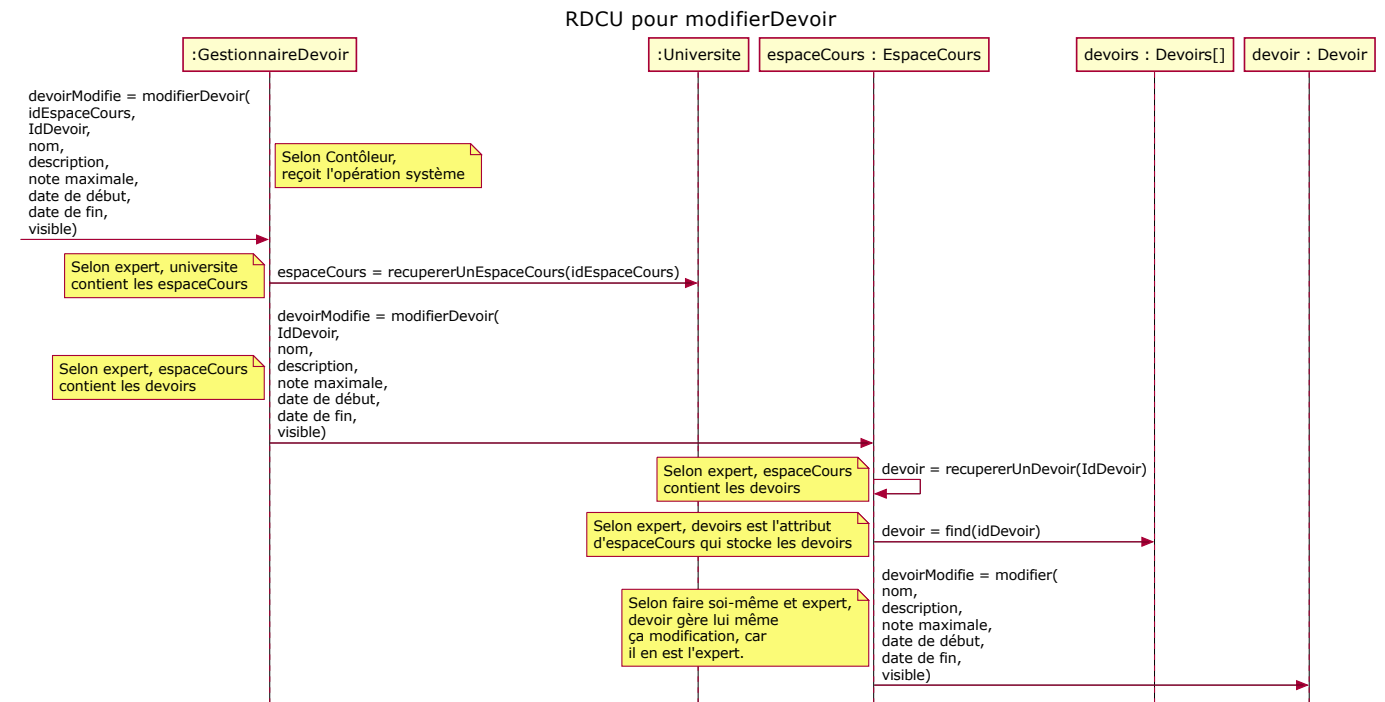
### Contrat d'opération

*PostCondition*

- Les attributs du devoir d ont été modifié

### RDCU





## CU04d - Supprimer devoir

**Acteur principal:** Enseignant

**Préconditions:**

- L'enseignant est authentifié.

**Garanties en cas de succès (postconditions):**

- Un devoir a été supprimé pour le cours.

**Scénario principal (succès):**

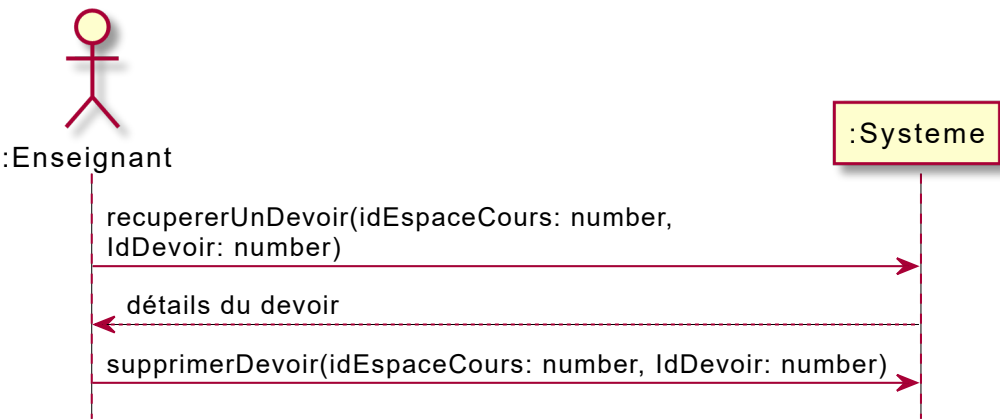
1. L'enseignant commence la suppression d'un devoir
2. Le système affiche les valeurs du devoir à supprimer.
3. L'enseignant confirme la suppression du devoir

**Extensions (ou scénarios alternatifs):**

- 2a. Le système désactive la possibilité de suppression tant que le devoir a été utilisé par des étudiants.

## DSS CU04b - Supprimer devoir

DSS supprimerDevoir



recupererUnDevoir(idEspaceCours: number, IdDevoir: number)

RécupérerUnDevoir

supprimerDevoir(idEspaceCours: number, IdDevoir: number)

Contrat d'opération

PostCondition

- L'instance d de Devoir a été supprimée

RDCU

