|  |
| --- |
| 19 décembre 2017 |

|  |
| --- |
| **Dossier d’analyse et d’architecture** |
| Ingénieur Généraliste  SLR F23B  Encadrant : Jacques SIMONIN |

1, rue de l’Institut

75000 PARIS – France

Tél. +33 (0)1 00 00 00 00

Siret : 000 000 000 000 00

APE : 0000 A

[**www.mines-telecom.fr**](http://www.mines-telecom.fr)

Mines ParisTech

Mines Albi

Mines Alès

Mines Douai

Mines Nantes

Mines Saint Étienne

Télécom ParisTech

Télécom Bretagne

Télécom SudParis

Télécom Ecole de Management

Télécom Lille1

Eurecom

Chao CHEN

Junshuai ZHU

Sommaire

[1. Analyse Fonctionnelle 2](#_Toc501458543)

[1.1 Contexte 2](#_Toc501458544)

[1.2 Rôles 2](#_Toc501458545)

[1.3 Cas d’utilisation 3](#_Toc501458546)

[1.3.2 Cas d’utilisation UcCréerProjet 4](#_Toc501458547)

[1.3.3 Cas d’utilisation UcArrêterProjet 6](#_Toc501458548)

[1.3.4 Cas d’utilisation UcChangerContrat 8](#_Toc501458549)

[1.3.5 Cas d’utilisation UcAjouterContributeur 10](#_Toc501458550)

[1.3.6 Cas d’utilisation UcSupprimerContributeur 12](#_Toc501458551)

[1.3.7 Cas d’utilisation UcConsulterfichierProjet 14](#_Toc501458552)

[1.4 Entités participantes aux cas d’utilisation 15](#_Toc501458553)

[2. Architecture Fonctionnelle 17](#_Toc501458554)

[2.1 îlots fonctionnels 17](#_Toc501458555)

[2.2 Dépendance des îlot fonctionnels 18](#_Toc501458556)

[2.3 Données Fonctionnelles 48](#_Toc501458557)

[3. Architecture Applicative 50](#_Toc501458558)

[3.1 Composant applicatives 50](#_Toc501458559)

[3.2 Dépendance des composants applicatifs 51](#_Toc501458560)

[3.3 Données applicatives 81](#_Toc501458561)

# Analyse Fonctionnelle

Ce chapitre présente l'analyse des exigences fonctionnelles du système:

* Le contexte du logiciel définissant le système
* Les rôles interagissant avec le système
* Les cas d’utilisation du système et les scénarios qui les illustrent
* Le modèle d’entité associé aux cas d'utilisation

L'analyse fonctionnelle proposée ici complète une description d'acteur de la direction scientifique et un modèle UML de données.

## Contexte

Figure 1 Contexte

## Rôles

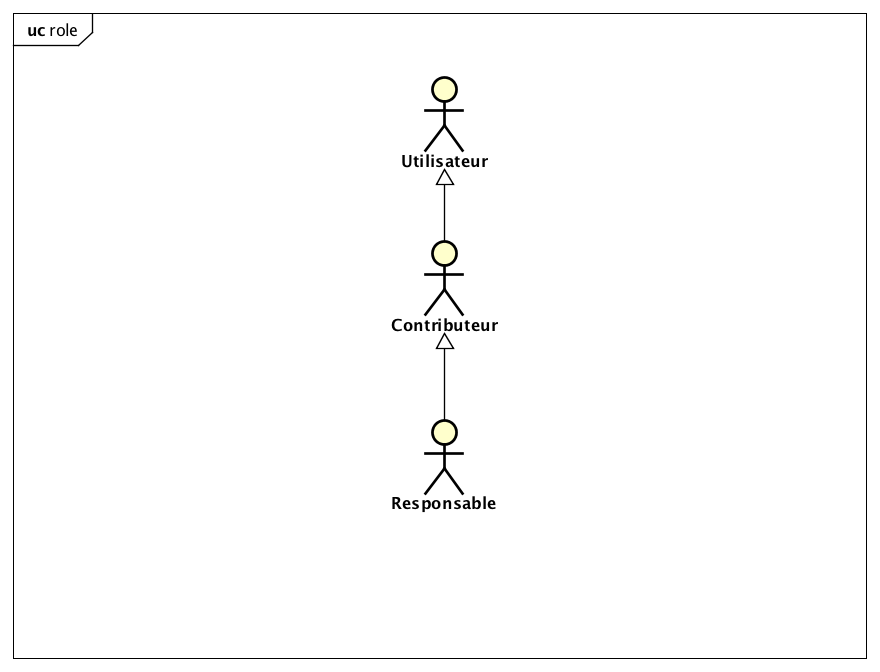
Les rôles sont définis dans le tableau Tableau 1 et modélisés avec UML dans la Figure 2.

Figure 2 Rôle

|  |  |
| --- | --- |
| Nom du rôle | Description du rôle |
| Utilisateur | Il peut être authentifié par son login et son mot de passe. Il a un champ de rôle. |
| Contributeur | Il contribue au projet. Il peut consulter le projet. |
| Responsable | Il est unique. Il gère tous les projets. |

Tableau 1

## Cas d’utilisation

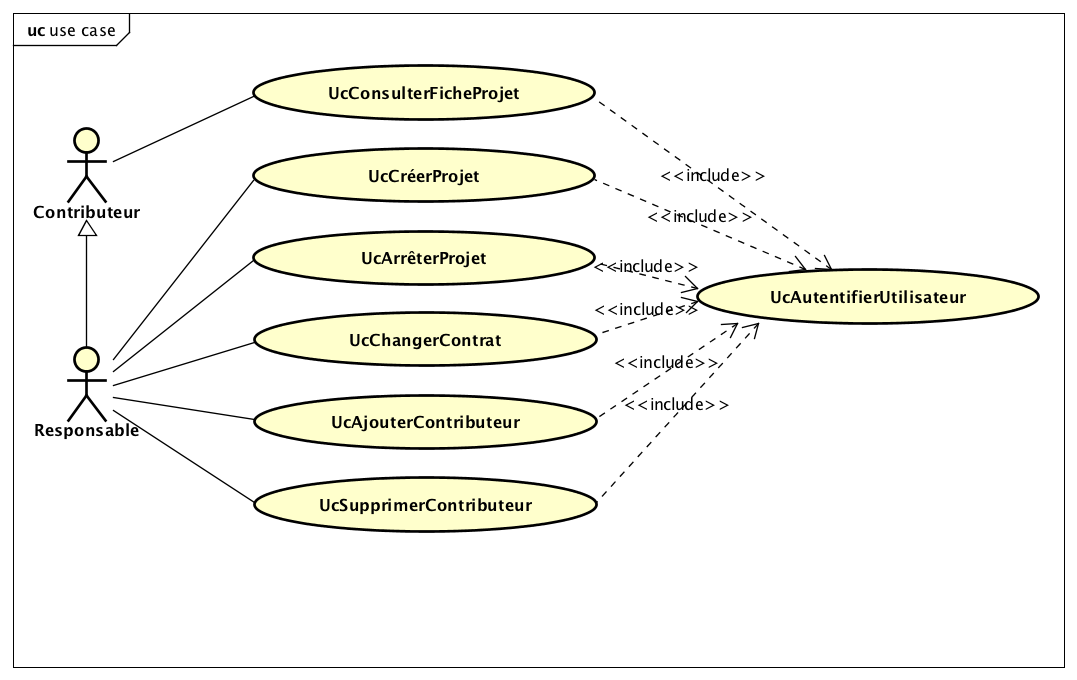
La Figure 3 représente le diagramme UML des cas d'utilisation comprenant les rôles qui interagissant avec cas d’utilisation et leurs relations d'inclusion.

Figure 3 Cas d’utilisation

* + 1. **Cas d’utilisation UcAuthentifierUtilisateur**

**Résumé**

Ce cas d'utilisation permet d’authentifier le rôle d’un utilisateur du système.

**Contexte de déclenchement**

Le contexte est d’un cas d’utilisation secondaire utile à tous les cas d'utilisation de gestion de direction scientifique pour des projet.

**Rôle**

Utilisateur

**Pré-condition**

Aucune

**Description**

L’utilisateur s'authentifie avec son login et son mot de passe.

Le système renvoie son rôle (contributeur, responsable) ou refuse l'accès si l’utilisateur n’est pas authentifié.

**Post-condition**

Le rôle de l’utilisateur est connu.

**Exception**

L’utilisateur n’est pas authentifié.

**Scénarios**

1. Scénario nominal d’authentification d’un utilisateur

|  |
| --- |
| **ScAuthentifierUtilisateur** |
| 1. Contrôle du login et du mot de passe de l’utilisateur 2. Le système renvoi le rôle de l’utilisateur |

1. Scénario Exception d’authentification d’un utilisateur inexistant.

|  |
| --- |
| **ScExceptionAuthentifierUtilisateurInexistant** |
| 1. Contrôle du login et du mot de passe de l’utilisateur 2. Le système refuse l'accès |

### Cas d’utilisation UcCréerProjet

**Résumé**

Ce cas d'utilisation permet de créer un projet.

**Contexte de déclenchement**

Ce cas d’utilisation est exécuté à la demande de responsable.

**Rôle**

Responsable

**Pré-condition**

Aucune

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable alors le cas d’utilisation est abandonné.

Le responsable peut demander à créer un nouveau projet.

Le système vérifie que le projet ayant le même nom n'existe pas déjà. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable doit donner une date de début et un data de fin pour un projet.

Le responsable ajoute des contributeurs avec ses charges dans ce projet.

Le système vérifie que le somme de charge est au moins 3 hommes par an. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable associer un contrat existant sur ce projet.

Le système vérifie l’existence du contrat. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le système enregistre ce projet.

**Post-condition**

Un nouveau projet est créé

**Exception**

Un non-responsable demande à créer un projet

Un projet a été créé précédemment avec le même nom

La création d’un projet sans le donne la quantité de travail suffisante

La création d’un projet sans le donne le contrat existant

**Scénarios**

1. Scénario nominal de création d’un projet

|  |
| --- |
| **ScCréerProjetAvecResponsable** |
| 1. Demande d'créer un projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet par le nom du projet 5. Contrôle négatif de l'existence d’un projet ayant le même titre 6. Le responsable donne la date de début, la date de fin du projet et le contrat 7. Le responsable ajoute des contributeurs avec ses charges dans ce projet 8. Contrôle positif de la somme de charge est au moins 3 hommes par an 9. Contrôle positif de l’existence du contrat. 10. Enregistrement du projet |

2. Scénario d’exception signifiant l'impossibilité de créer un projet utilisant le rôle non-responsable

|  |
| --- |
| **ScCréerProjetAvecNonResponsable** |
| 1. Demande d'créer un projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test négatif de son rôle de responsable 4. Abandon de la création du projet |

1. Scénario d’exception signifiant l'impossibilité de créer un projet déjà existé

|  |
| --- |
| **ScExceptionCréerProjetExistant** |
| 1. Demande d'créer un projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Identification du projet par le nom du projet 4. Contrôle positif de l'existence d’un projet ayant le même titre 5. Abandon de la création du projet |

1. Scénario d’exception signifiant l'impossibilité de créer un projet sans le donne la quantité de travail suffisante

|  |
| --- |
| **ScExceptionCréerProjetQuantitéNonSuffisant** |
| 1. Demande d'créer un projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Identification du projet par le nom du projet 4. Contrôle négatif de l'existence d’un projet ayant le même titre 5. Le responsable ajoute des contributeurs avec ses charges dans ce projet 6. Le système vérifie que le somme de charge est au moins 3 hommes par an 7. Contrôle négatif de la somme de charge est au moins 3 hommes par an 8. Abandon de la création du projet |

1. Scénario d’exception signifiant l'impossibilité de créer un projet sans le donne le contrat

|  |
| --- |
| **ScExceptionCréerProjetSansContratExistant** |
| 1. Demande d'créer un projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Identification du projet par le nom du projet 4. Contrôle négatif de l'existence d’un projet ayant le même titre 5. Le responsable ajoute des contributeurs avec ses charges dans ce projet 6. Le système vérifie que le somme de charge est au moins 3 hommes par an 7. Contrôle positif de la somme de charge est au moins 3 hommes par an 8. Contrôle négatif de l’existence du contrat. 9. Abandon de la création du projet |

### Cas d’utilisation UcArrêterProjet

**Résumé**

Ce cas d’utilisation permet d’arrêter un projet.

**Contexte de déclenchement**

Le cas d’utilisation est exécuté à la demande d’un responsable qu’il souhaite arrêter un projet.

**Rôle**

Responsable

**Pré-condition**

Aucune

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable alors le cas d’utilisation est abandonné.

Le responsable sélectionne un projet dans l’application par le nom du projet.

Le système vérifie que le projet existe. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable modifie la date finale de ce projet.

Le système vérifie que le projet n’est pas déjà arrêté. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le système enregistre la modification.

**Post-condition**

La date finale de projet sélectionnée est modifiée à la date de la modification.

**Exception**

L’utilisateur n’a pas été authentifié comme responsable

Le projet est déjà arrêté.

Le projet n’existe pas.

**Scénarios**

1. Scénario nominal d'arrêt d’un projet

|  |
| --- |
| **ScArrêterProjetCourantAvecResponsable** |
| 1. Demande d'arrêt un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification positive du projet 5. Vérification positive de la validation de date final de projet 6. Modification de la date de fin de projet 7. Enregistrement de la modification |

1. Scénario d’exception signifiant un utilisateur n’ayant pas le rôle de responsable

|  |
| --- |
| **ScExceptionArrêterProjetAvecNonResponsable** |
| 1. Demande d'arrêt un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test négatif de son rôle de responsable 4. Abandon de la modification de la date de fin du projet |

1. Scénario d’exception d'arrêt d’un projet expiré

|  |
| --- |
| **ScExceptionArrêterProjetExpiré** |
| 1. Demande d'arrêt un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification positive du projet 5. Vérification négative de la validation de la date final du projet 6. Abandon de la modification de la date de fin du projet |

1. Scénario d’exception d'arrêt d’un projet inexistant

|  |
| --- |
| **ScExceptionArrêterProjetInexistant** |
| 1. Demande d'arrêt un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification négative du projet 5. Abandon de la modification de la date final du projet |

### Cas d’utilisation UcChangerContrat

**Résumé**

Ce cas d’utilisation permet d’ajouter un contrat sur un projet.

**Contexte de déclenchement**

Le cas d’utilisation est exécuté à la demande d’un responsable qu’il souhaite ajouter un contrat sur un projet.

**Rôle**

Responsable

**Pré-condition**

Aucune

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable alors le cas d’utilisation est abandonné.

Le responsable sélectionne un projet par son nom.

Le système vérifie que le projet existe. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable sélectionne un contrat par son nom.

Le système vérifie que le contrat existe. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable remplace le contrat de ce projet par le nouveau contrat.

Le contrat est identifié par le nom du contrat.

Le système enregistre la modification de contrat sur ce projet.

**Post-condition**

Un contrat est changé sur le projet

**Exception**

Un non-responsable demande à ajouter un contrat.

Le projet n’est pas existant.

Le nouveau contrat n’existe pas.

**Scénarios**

1. Scénario nominal d’ajouter un contrat

|  |
| --- |
| **ScChangerContratAvecResponsable** |
| 1. Demande de changer le contrat sur projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet par le nom du projet 5. Identification du contrat par le nom du contrat 6. Remplacement de contrat sur le projet 7. Enregistrement de la modification |

1. Scénario d’exception signifiant l'impossibilité d’ajouter un contrat utilisant un rôle non-responsable

|  |
| --- |
| **ScExceptionChangerContratAvecNonResponsable** |
| 1. Demande de changer le contrat sur projet dans l'application DS 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test négatif de son rôle de responsable 4. Abandon de la modification d'un contrat |

1. Scénario d’exception de changement du contrat sur un projet inexistant

|  |
| --- |
| **ScExceptionChangerContratSurProjetInexistant** |
| 1. Demande de changer le contrat sur projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification négative du projet 5. Abandon de la modification du contrat sur le projet |

1. Scénario d’exception de changement du contrat inexistant

|  |
| --- |
| **ScExceptionChangerContratInexistant** |
| 1. Demande de changer le contrat sur projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification positive du projet 5. Identification négative du contrat 6. Abandon de la modification du contrat sur le projet |

### Cas d’utilisation UcAjouterContributeur

**Résumé**

Ce cas d’utilisation permet d’ajouter un contributeur sur un projet.

**Contexte de déclenchement**

Le cas d’utilisation est exécuté à la demande d’un responsable qu’il souhaite ajouter un contributeur dans un projet.

**Rôle**

Responsable

**Pré-condition**

Aucune

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable alors le cas d’utilisation est abandonné.

Le responsable peut demander à ajouter un contributeur dans un projet.

Le système vérifie que le projet existe. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable choisit le contributeur et le projet.

Le système vérifie que le contributeur existe dans la base de données. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le système vérifie que le contributeur n'existe pas déjà dans ce projet. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le contributeur est identifié par le nom du contributeur.

Le système enregistre ce contributeur dans le projet.

**Post-condition**

Un contributeur est ajouté

**Exception**

Un non-responsable demande à ajouter un contributeur.

Un contributeur a été ajouté précédemment avec le même nom.

Un contributeur est ajouté dans un projet inexistant.

Un contributeur n’existe pas.

**Scénarios**

1. Scénario nominal d’ajouter contributeur dans un projet

|  |
| --- |
| **ScAjouterContributeurAvecResponsable** |
| 1. Demande d’ajouter un contributeur sur le projet 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet par le nom 5. Identification du contributeur par le nom du contributeur 6. Contrôle négatif de l'existence d’un contributeur existant dans le projet 7. Enregistrement du contributeur |

1. Scénario d’exception signifiant l'impossibilité d’ajouter un contributeur utilisant un rôle non-responsable

|  |
| --- |
| **ScExceptionAjouterContributeurAvecNonResponsable** |
| 1. Demande d’ajouter un contributeur sur le projet 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test négatif de son rôle de responsable 4. Abandon de l'ajout d'un contributeur |

1. Scénario d’exception signifiant l'impossibilité d’ajouter un contributeur déjà lié par le projet.

|  |
| --- |
| **ScExceptionAjouterContributeurDejadansProjet** |
| 1. Demande d’ajouter un contributeur sur le projet 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet par le nom 5. Identification du contributeur par le nom du contributeur 6. Contrôle positif de l'existence d’un contributeur existant dans le projet 7. Abandon de l'ajout d'un contributeur |

1. Scénario d’exception d’ajouter un contributeur dans un projet inexistant

|  |
| --- |
| **ScExceptionAjouterContributeurDansProjetInexistant** |
| 1. Demande d'ajouter un contributeur sur un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification négative du projet 5. Abandon de l’ajoute du contributeur sur le projet |

1. Scénario d’exception d’ajouter un contributeur Inexistant

|  |
| --- |
| **ScExceptionAjouterContributeurInexistant** |
| 1. Demande d’ajouter un contributeur sur le projet 2. Authentification d’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification négative du contributeur par le nom du contributeur 5. Abandon de l'ajout d'un contributeur |

### Cas d’utilisation UcSupprimerContributeur

**Résumé**

Ce cas d’utilisation permet de supprimer un contributeur sur un projet.

**Contexte de déclenchement**

Le cas d’utilisation est exécuté à la demande d’un responsable qu’il souhaite ajouter un contributeur dans un projet.

**Rôle**

Responsable

**Pré-condition**

Le contributeur sélectionné travaille sur le projet sélectionné.

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable alors le cas d’utilisation est abandonné.

Le responsable sélectionne un projet par son nom.

Le système vérifie que le projet existe. Si ce n'est pas le cas, le cas d’utilisation est abandonné.

Le responsable supprime un contributeur sur le projet.

Le système vérifie que ce contributeur existe dans ce projet. Si ce contributeur n'existe pas, alors le cas d’utilisation est abandonné.

Le système vérifie que le somme de charge est au moins 3 hommes par an. Si le somme de charge est moins que 3 hommes par an, alors le cas d’utilisation est abandonné.

Le système enregistre la suppression dans le projet.

**Post-condition**

Le contributeur sélectionné est supprimé sur le projet.

**Exception**

L'utilisateur n’a pas été authentifié comme responsable.

Le projet sélectionné n'existe pas.

Le contributeur sélectionné n'existe pas dans le projet.

Le somme de charge est moins que 3 hommes par an après avoir supprimé le contributeur.

**Scénarios**

1. Scénario nominal de supprimer un contributeur sur un projet.

|  |
| --- |
| **ScSupprimerContributeurAvecResponsable** |
| 1. Demande de supprimer un contributeur sur un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet 5. Identification du contributeur 6. Pré-Suppression d’un contributeur sur ce projet 7. Vérification de la somme de charge est au moins que 3 hommes par an 8. Enregistrement de la suppression |

1. Scénario d’exception de supprimer un contributeur dans un projet inexistant

|  |
| --- |
| **ScExceptionSupprimerContributeurDansProjetInexistant** |
| 1. Demande de supprimer un contributeur de projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification négative du projet 5. Abandon de la suppression du contributeur sur le projet |

1. Scénario d’exception de supprimer un contributeur inexistant

|  |
| --- |
| **ScExceptionSupprimerContributeurDansInexistant** |
| 1. Demande de supprimer un contributeur de projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet 5. Identification négative du contributeur 6. Abandon de la suppression du contributeur sur le projet |

1. Scénario d’exception signifiant un utilisateur n’ayant pas le rôle de responsable

|  |
| --- |
| **ScExceptionSupprimerContributeurAvecNonResponsable** |
| 1. Demande de supprimer un contributeur sur un projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test négatif de son rôle de responsable 4. Abandon de la suppression de contributeur sur le projet |

1. Scénario d’exception de la somme de charge insuffisant

|  |
| --- |
| **ScExceptionChargeInsuffisant** |
| 1. Demande d'arrêt de projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable 4. Identification du projet 5. Identification négative du contributeur 6. Pré-Suppression un contributeur sur ce projet 7. Vérification négative de la somme de charge est moins que 3 hommes par an 8. Abandon de la modification de la date de fin de projet |

### Cas d’utilisation UcConsulterFichierProjet

**Résumé**

Ce cas d’utilisation permet de consulter le fichier du projet.

**Contexte de déclenchement**

Le cas d’utilisation est exécuté à la demande d’un Contributeur qu’il souhait consulter le fichier d'un projet.

**Rôle**

Contributeur

**Pré-condition**

Aucune

**Description**

L’utilisateur s’authentifie son login et son mot de passe (cf. UcAuthentifierUtilisatuer).

S’il n’a pas de rôle de responsable ou de rôle contributeur alors le cas d’utilisation est abandonné.

Le contributeur peut consulter l’information un projet.

Le système vérifier l'existence du projet, si le projet existe, le système le renvoie les informations correspond au projet. Sinon, le cas d’utilisation est abandonné.

**Post-condition**

L’information du fichier est renvoyée.

**Exception**

Le projet n’existe pas.

**Scénarios**

1. Scénario nominal de consulter un projet.

|  |
| --- |
| **ScConsulterfichierProjet** |
| 1. Demande d'arrêt de projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable ou contributeur 4. Identification du projet par le nom du projet 5. Contrôle positif de l'existence d’un projet 6. Renvoie tous les informations correspondent au projet |

1. Scénario d’exception signifiant l'impossibilité de consulter un projet qui n'existe pas.

|  |
| --- |
| **ScExceptionConsulterfichierProjetNonExistant** |
| 1. Demande d'arrêt de projet dans l'application DS 2. Authentification de l’utilisateur (cf. UcAuthentifierUtilisatuer) 3. Test positif de son rôle de responsable ou contributeur 4. Identification du projet par le nom du projet 5. Contrôle négatif de l'existence d’un projet 6. Abandon de la consultation du projet |

## Entités participantes aux cas d’utilisation

Les entités participantes aux cas d'utilisation sont définies dans le Tableau 2 et modélisées avec leur relation, conformément à UML dans la Figure 4.

|  |  |
| --- | --- |
| Nom de l’entité | Définition de l’entité |
| Utilisateur | Une personne ayant un accès au système d’un rôle. Il est identifiée par son login et son mot de passe |
| Contributeur | Une personne travaille sur des projets. Il peut consulter le fichier du projet. Il est identifié comme contributeur |
| Responsable | Une personne gère tous les projets dans ce système. Il est identifié comme responsable |
| Charge | La relation entre les contributeurs et un projet, ayant la quantité de charge sur un projet |
| Projet | Projet ayant les informations de dates, de contributeurs, de contrat relatifs |
| Contrat | Règle de projet |

Tableau 2

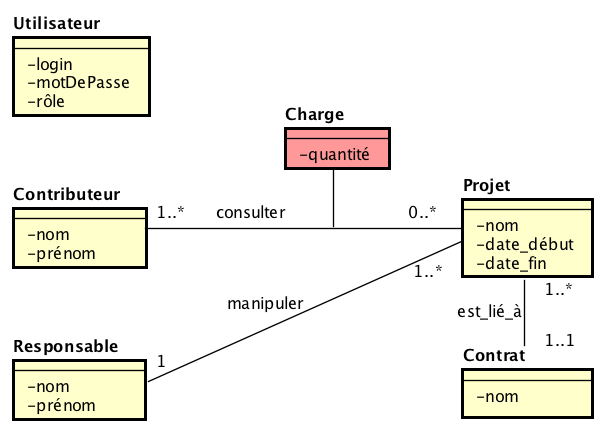


Figure 4 Diagramme des entités participantes aux cas d’utilisation

# Architecture Fonctionnelle

## îlots fonctionnels

La conception des îlots fonctionnels du système est déduite des entités participant aux cas d’utilisation (Tableau 3) et de leurs relations (Tableau 4). Elle aurait pu aussi déduite des cas d'utilisation.

|  |  |  |
| --- | --- | --- |
| Entité participante | Attribut | Îlot fonctionnel |
| Contributeur | -nom | IF Gestion Personne |
| -prénom |
| Responsable | -nom | IF Gestion Personne |
| -prénom |
| Projet | -data\_debut | IF Gestion Projet |
| -date\_fin |
| -nom |
| Contrat | -nom | IF Gestion Contrat |
| Utilisateur | -login | IF Gestion Authentification |
| -password |
| -role |

Tableau 3

|  |  |  |
| --- | --- | --- |
| Relation entre entités  participantes | Entités reliées | îlot fonctionnel |
| est\_lié\_à | Projet | IF Gestion Contrat |
| Contrat |
| manipule | Responsable | IF Gestion Projet |
| Project |
| consulte | Contributeur | IF Gestion Projet |
| Projet |

Tableau 4

## Dépendance des îlot fonctionnels

La dépendance des îlots fonctionnels répertoriés précédemment est conçu à partir des scénarios spécifiés en analyse fonctionnelle. C'est une approche dynamique de l'architecture fonctionnelle. La traçabilité des scénarios dans les diagrammes de séquences fonctionnelles est reportée dans le Tableau 5:

|  |  |
| --- | --- |
| Scénario | Diagramme de séquences fonctionnelles |
| ScAuthentifierUtilisateur | Figure 5 |
| ScExceptionAuthentifierUtilisateurInexistant | Figure 6 |
| ScCréerProjetAvecResponsable | Figure 7 |
| ScExceptionCréerProjetAvecNonResponsable | Figure 8 |
| ScExceptionCréerProjetExistant | Figure 9 |
| ScExceptionCréerProjetQuantitéNonSuffisant | Figure 10 |
| ScExceptionCréerProjeSansContratExistant | Figure 11 |
| ScArrêterProjetCourantAvecResponsable | Figure 12 |
| ScExceptionArrêterProjetAvecNonResponsable | Figure 13 |
| ScExceptionArrêterProjetExpiré | Figure 14 |
| ScExceptionArrêterProjetInexistant | Figure 15 |
| ScChangerContratAvecResponsable | Figure 16 |
| ScExceptionChangerContratAvecNonResponsable | Figure 17 |
| ScExceptionChangerContratSurProjetInexistant | Figure 18 |
| ScExceptionChangerContratInexistant | Figure 19 |
| ScAjouterContributeurAvecResponsable | Figure 20 |
| ScExceptionAjouterContributeurAvecNonResponsable | Figure 21 |
| ScExceptionAjouterContributeurDansProjetInexistant | Figure 22 |
| ScExceptionAjouterContributeurDejadansProjet | Figure 23 |
| ScExceptionAjouterContributeurInexistant | Figure 24 |
| ScSupprimerContributeurAvecResponsable | Figure 25 |
| ScExceptionSupprimerContributeurAvecNonResponsable | Figure 26 |
| ScExceptionSupprimerContributeurDansProjetInexistant | Figure 27 |
| ScExceptionSupprimerContributeurInexistant | Figure 28 |
| ScExceptionSupprimerContributeurChargeInsuffisant | Figure 29 |
| ScConsulterfichierProjet | Figure 30 |
| ScExceptionConsulterfichierProjetNonExistant | Figure 31 |

Tableau 5

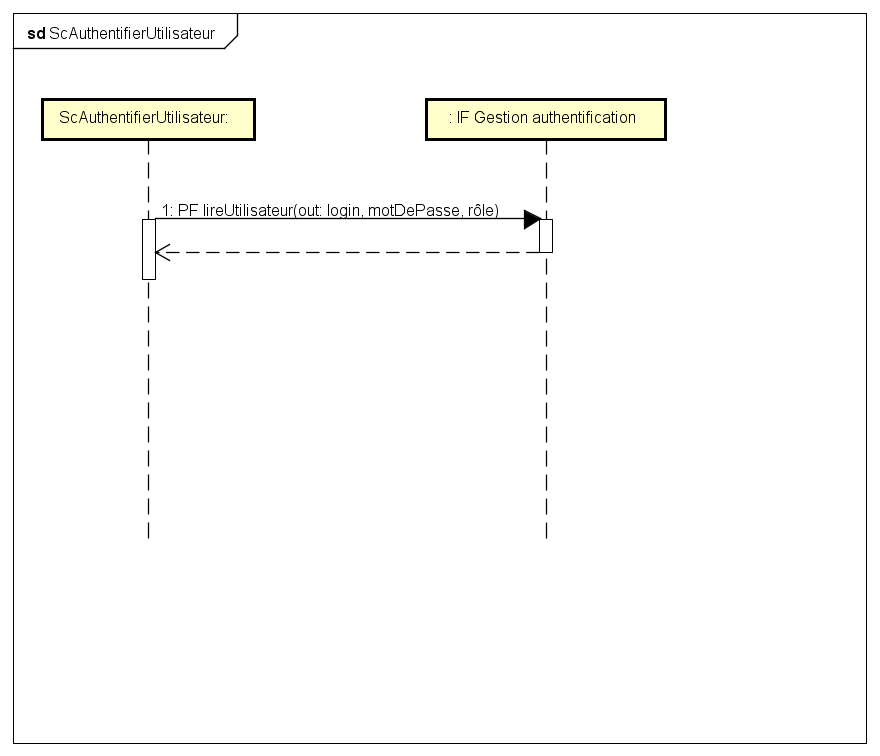


Figure 5 Diagramme de séquences fonctionnelles du scénario ScAuthentifierUtilisateur

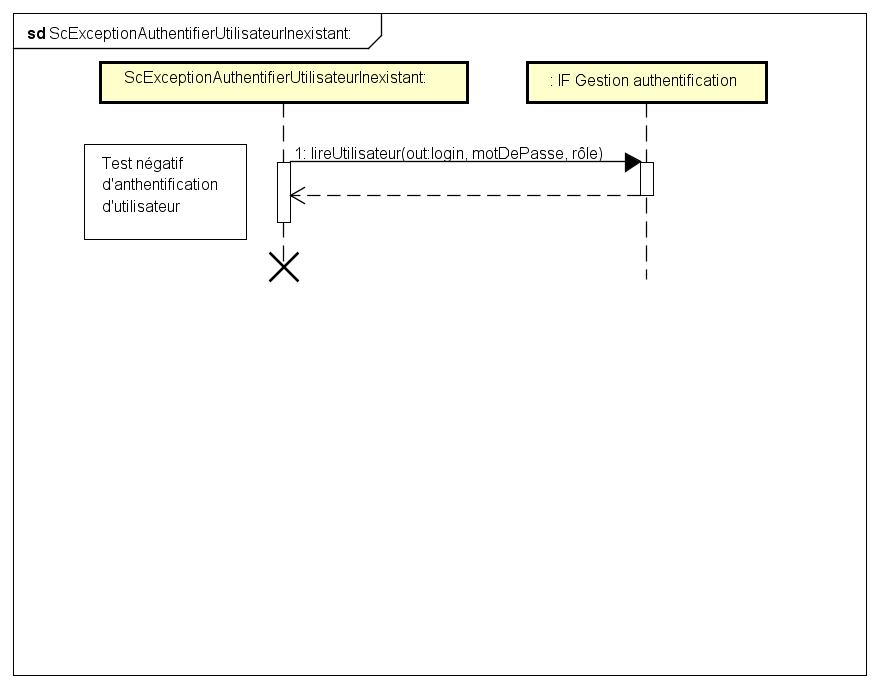


Figure 6 Diagramme de séquences fonctionnelles du scénario ScExceptionAuthentifierUtilisateurInexistant

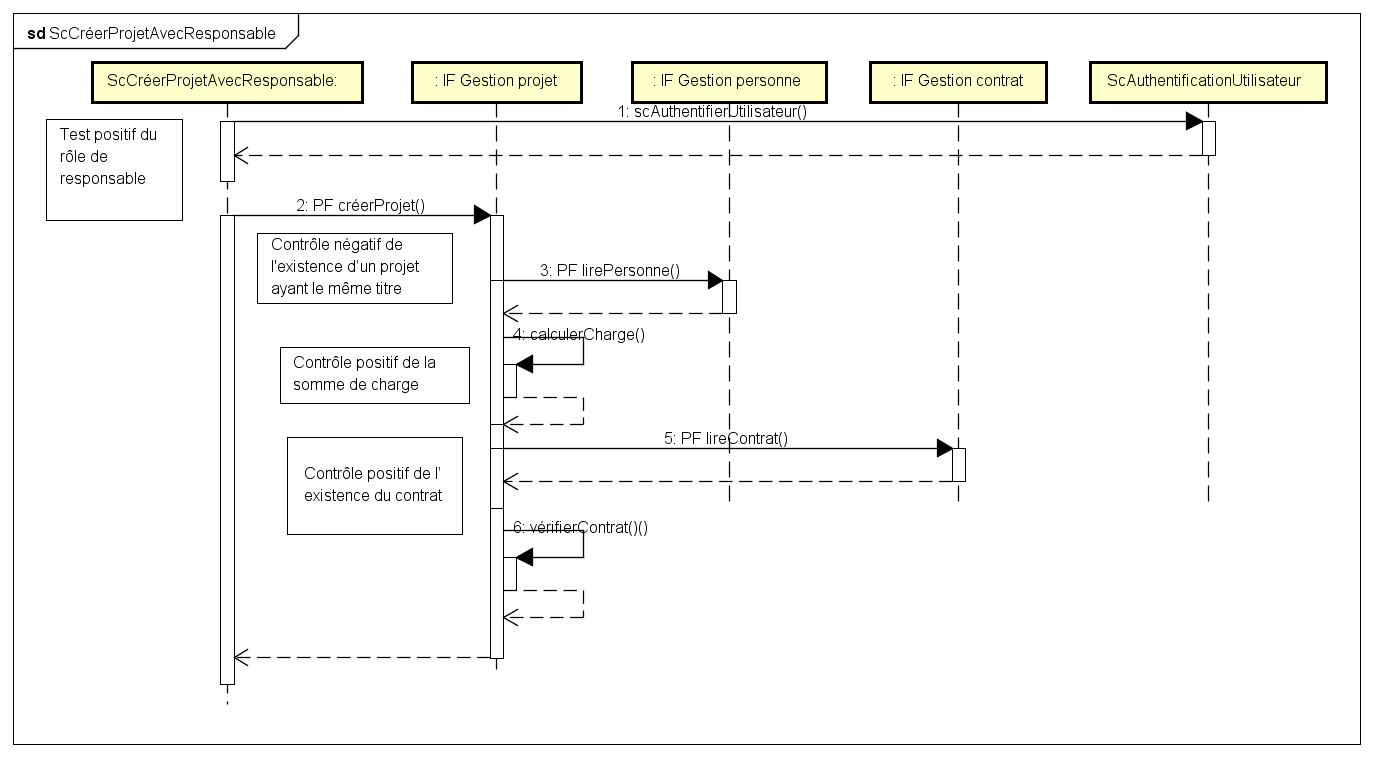
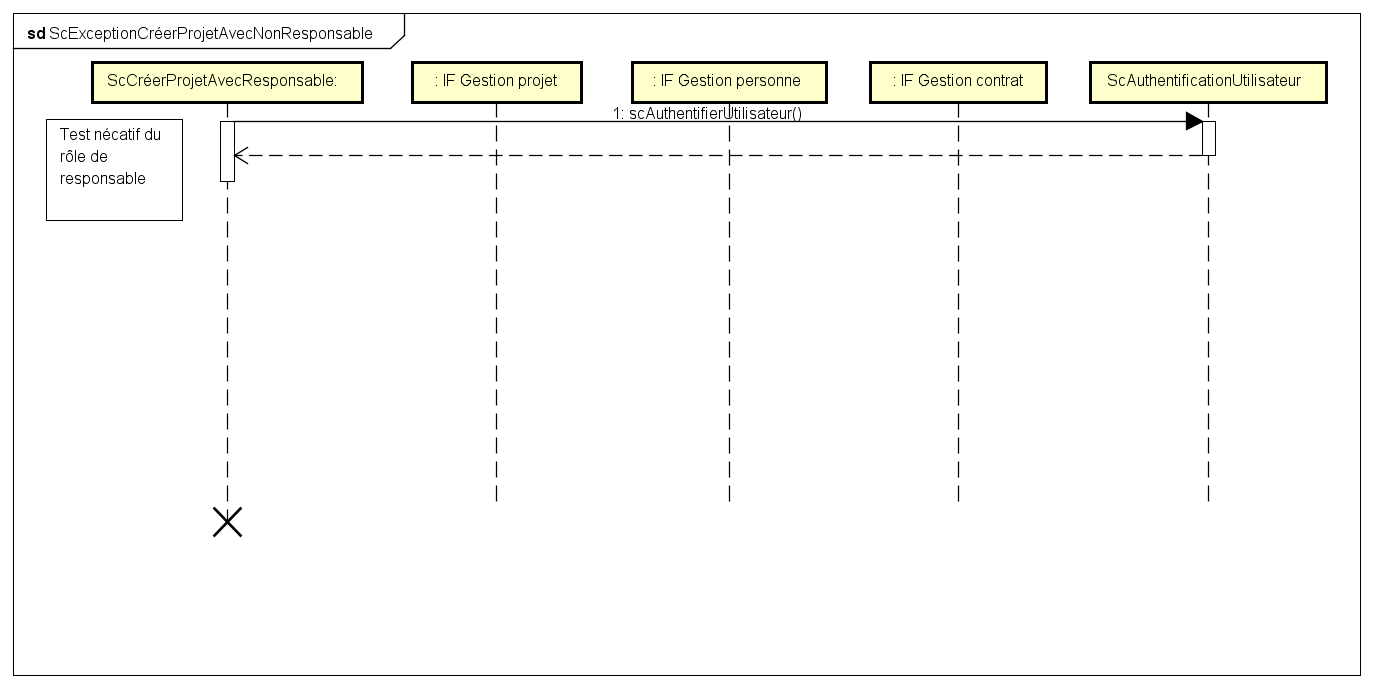
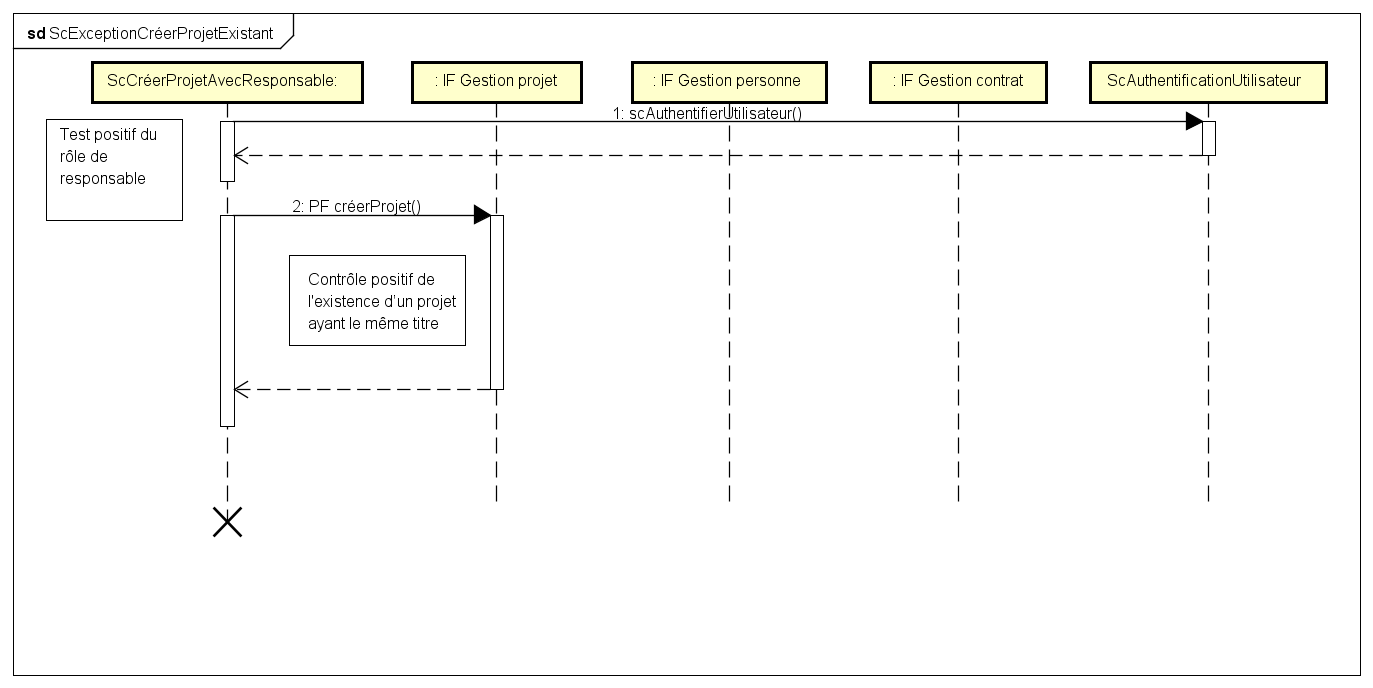
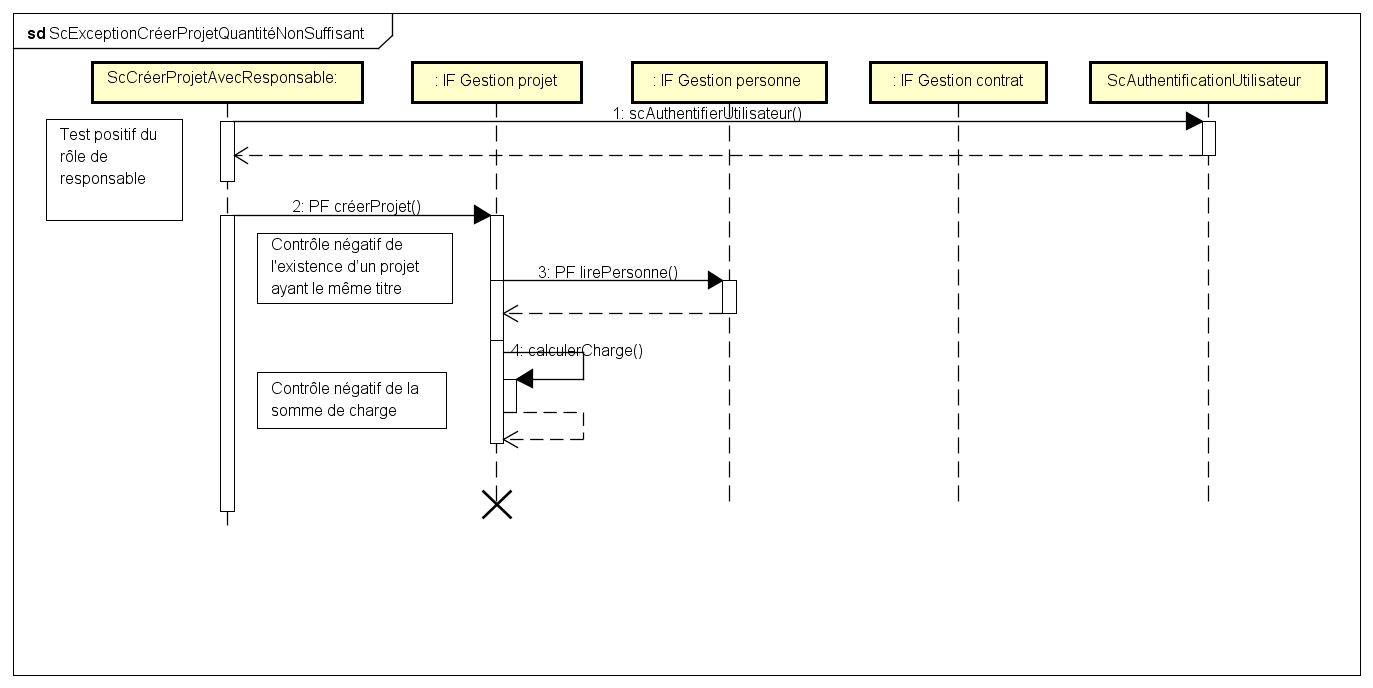
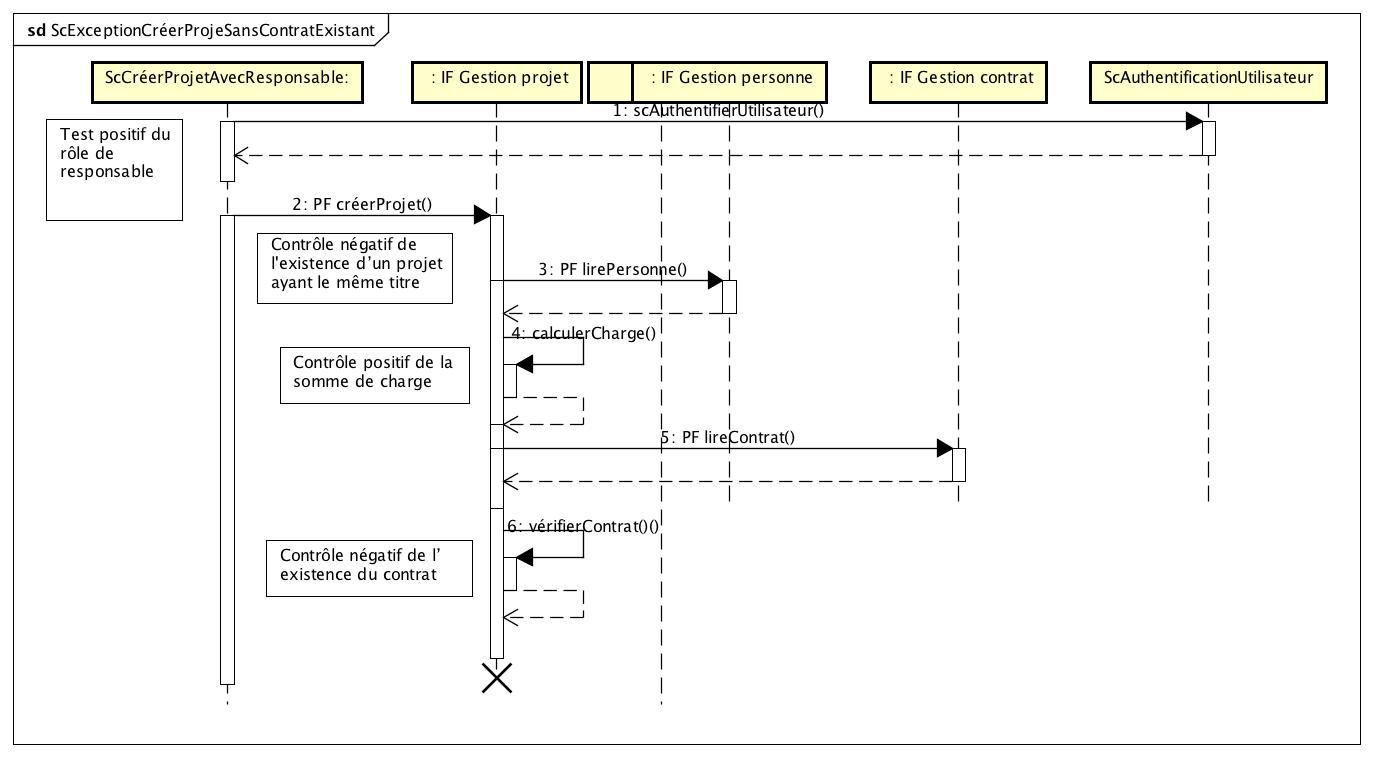
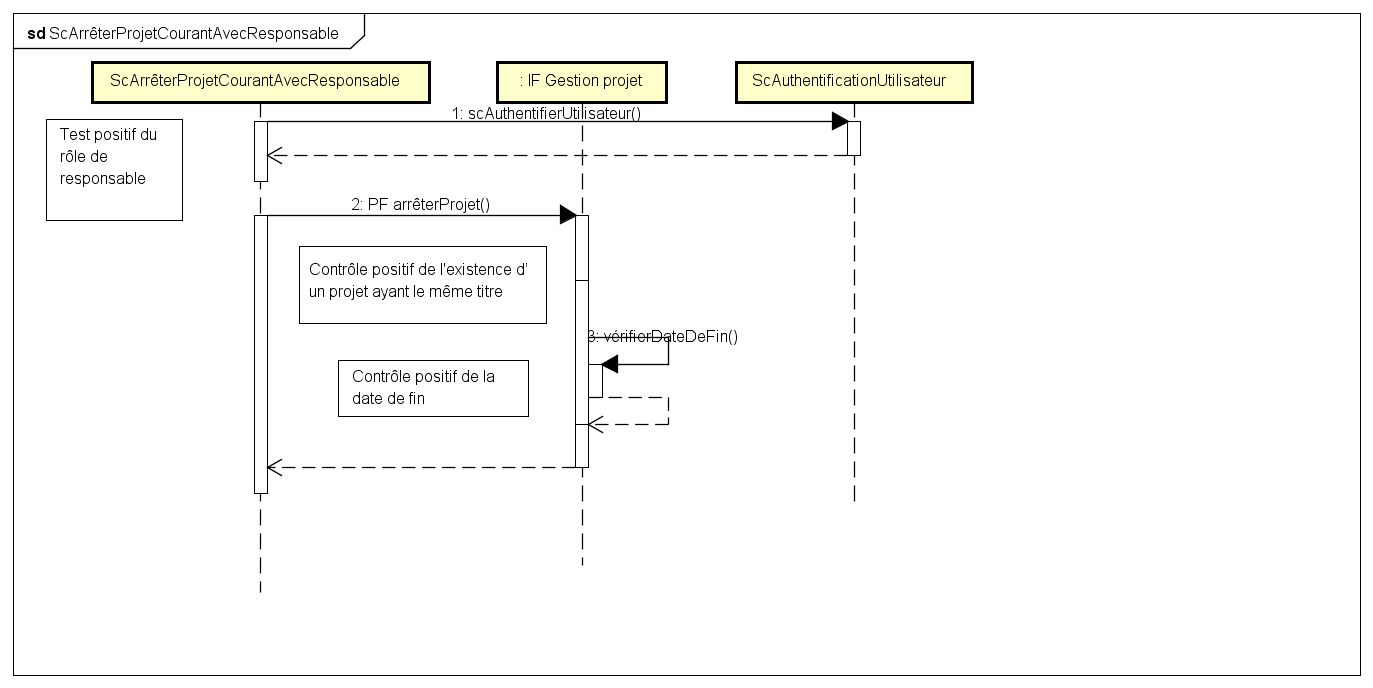
Figure 7 Diagramme de séquences fonctionnelles du scénario ScCréerProjetAvecResponsable

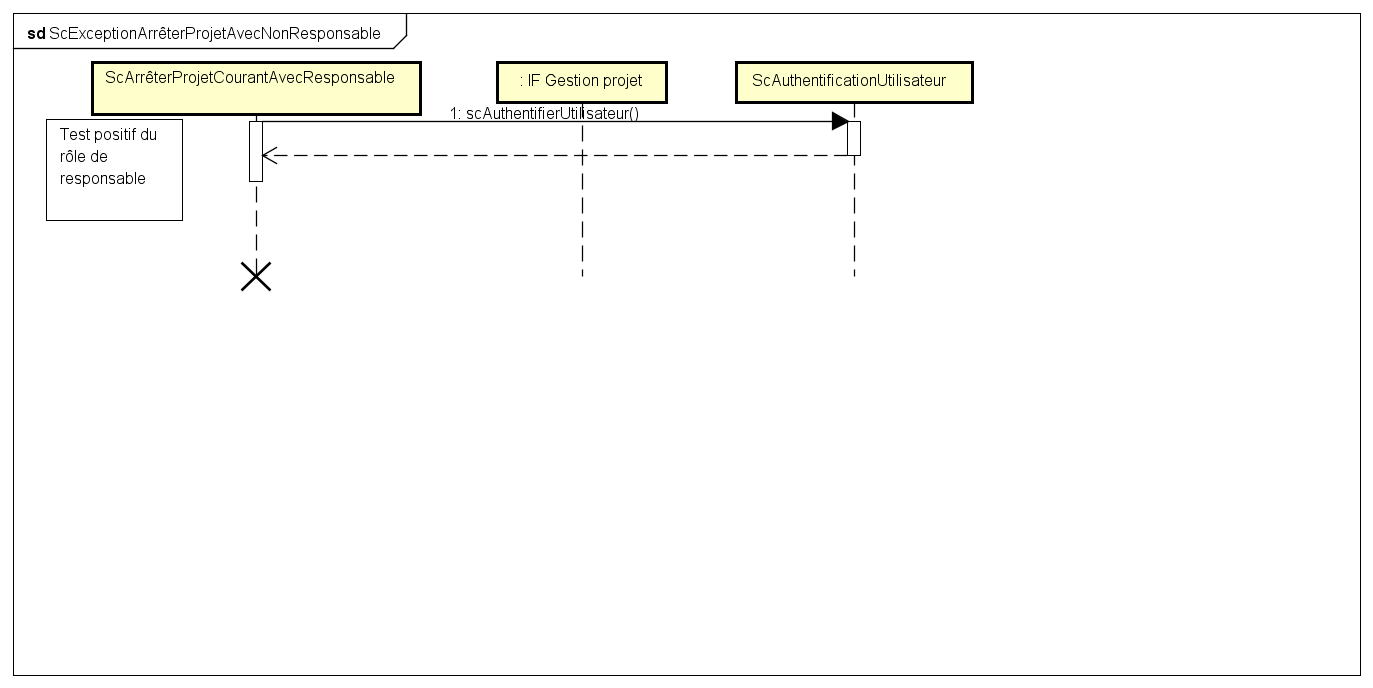
Figure 8 Diagramme de séquences fonctionnelles du scénario ScExceptionCréerProjetAvecNonResponsable

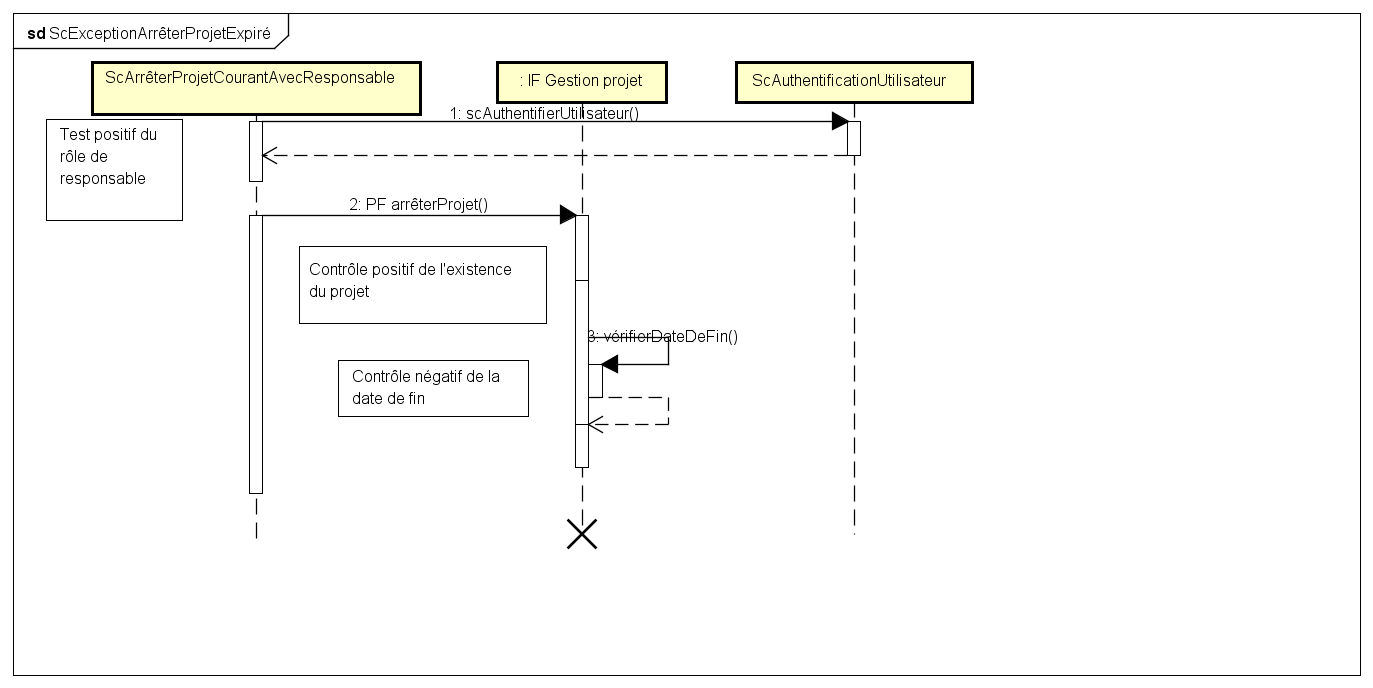
Figure 9 Diagramme de séquences fonctionnelles du scénario ScExceptionCréerProjetExistant

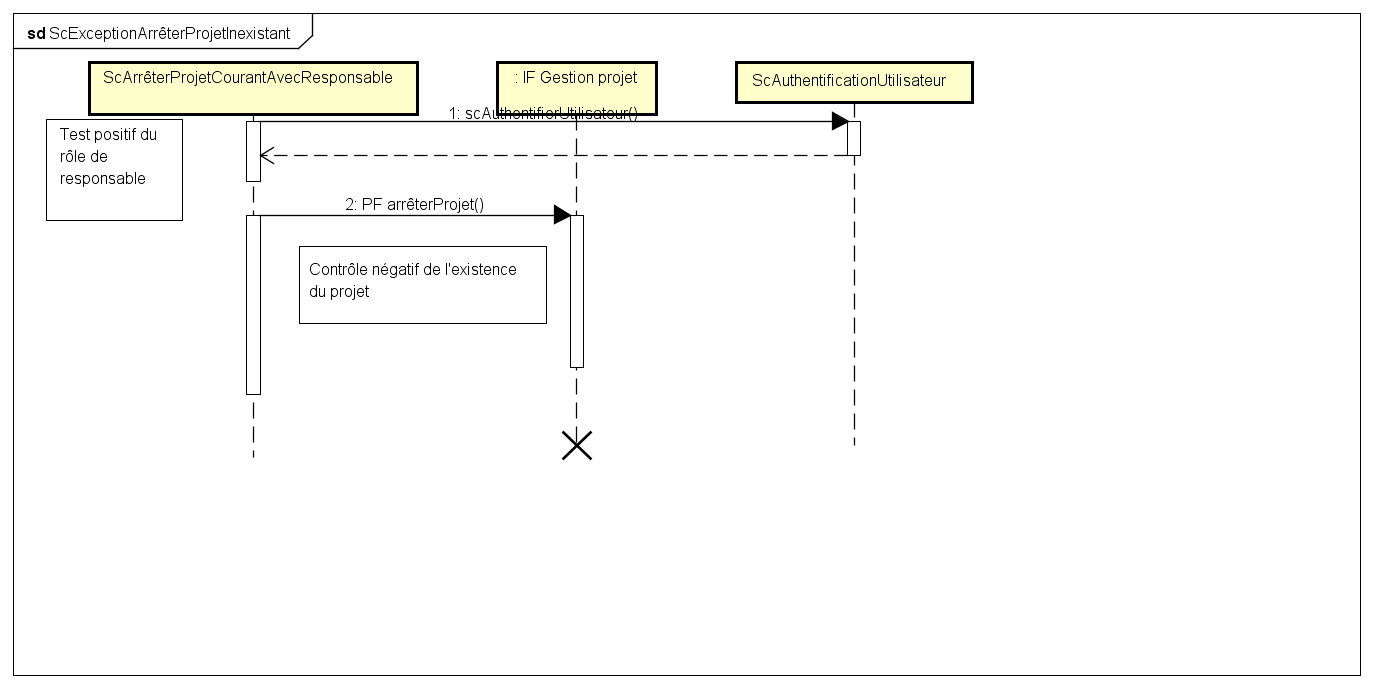
Figure 10 Diagramme de séquences fonctionnelles du scénario ScExceptionCréerProjetQuantitéNonSuffisante

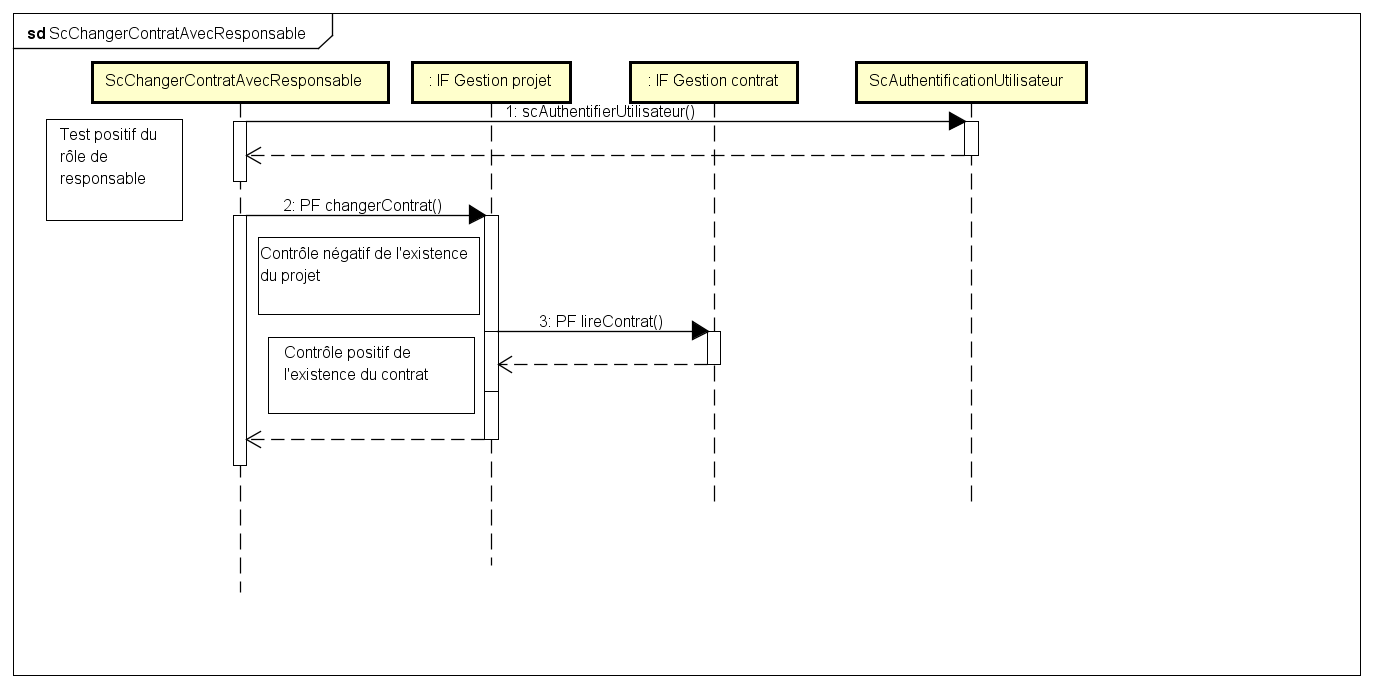
Figure 11 Diagramme de séquences fonctionnelles du scénario ScExceptionCréerProjeSansContratExistant

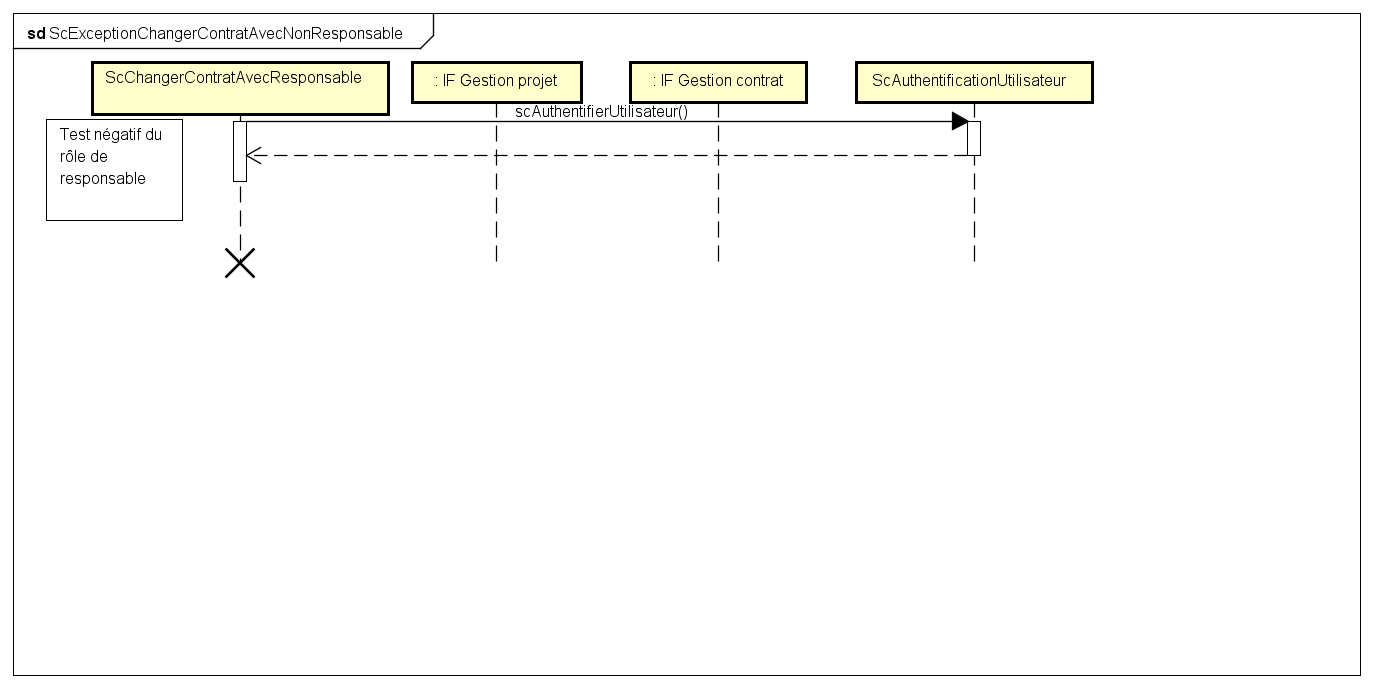
Figure 12 Diagramme de séquences fonctionnelles du scénario ScArrêterProjetCourrantAvecResponsable

Figure 13 Diagramme de séquences fonctionnelles du scénario ScExceptionArrêterProjetAvecNonResponsable

Figure 14 Diagramme de séquences fonctionnelles du scénario ScExceptionArrêterProjetExpiré

Figure 15 Diagramme de séquences fonctionnelles du scénario ScExceptionArrêterProjetAvecNonInexistant

Figure 16 Diagramme de séquences fonctionnelles du scénario ScChangerContratAvecResponsable

Figure 17 Diagramme de séquences fonctionnelles du scénario ScExceptionChangerContratAvecNonResponsable

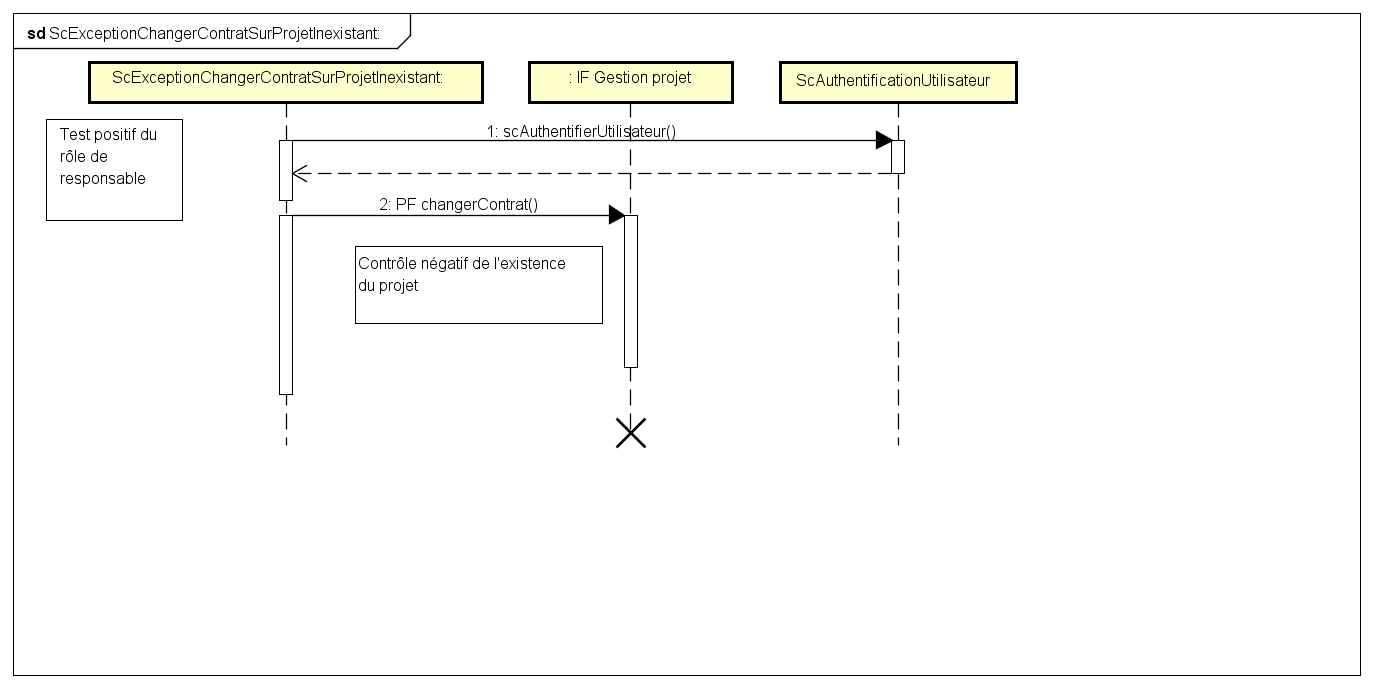
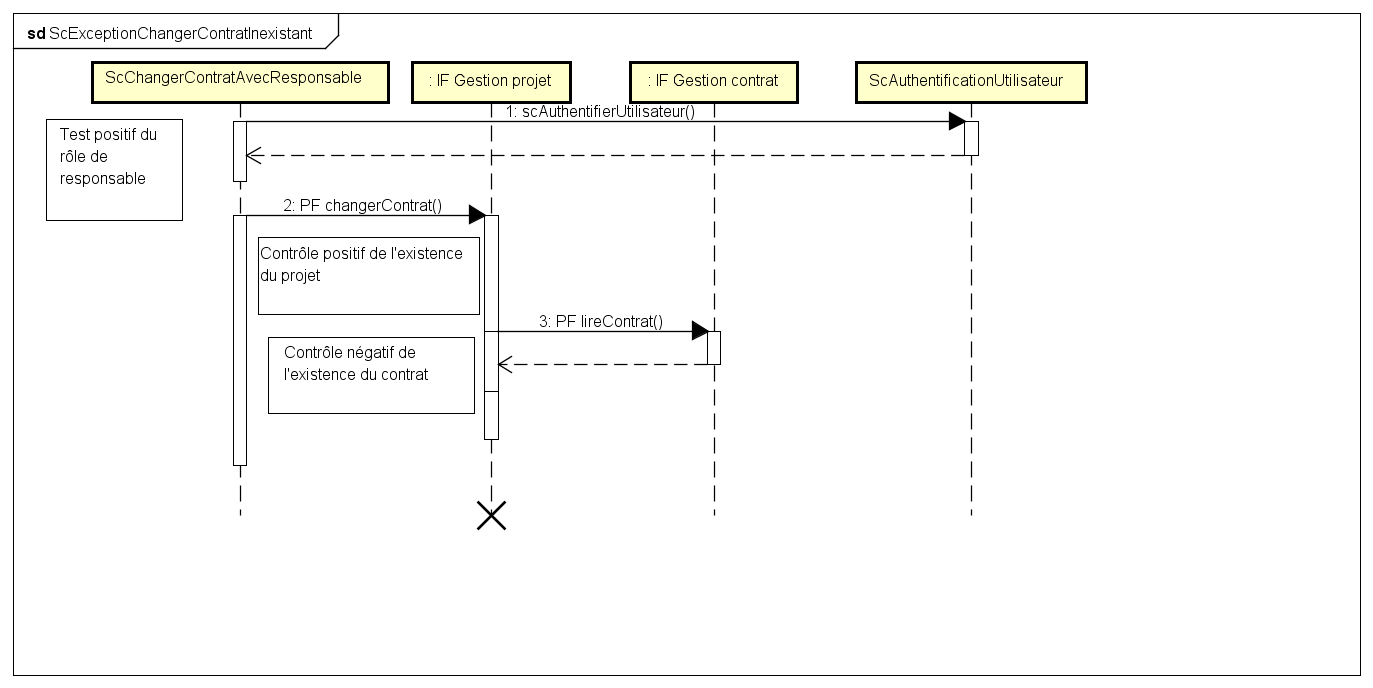
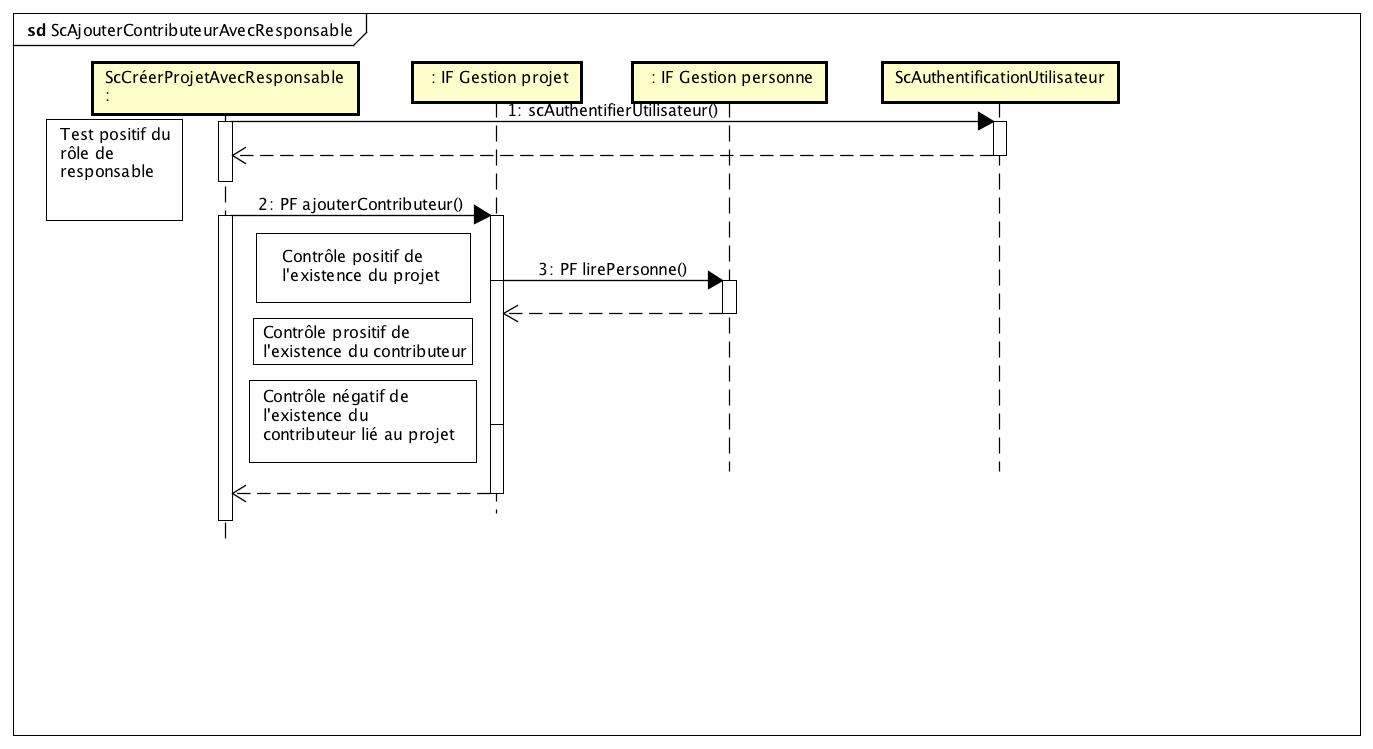
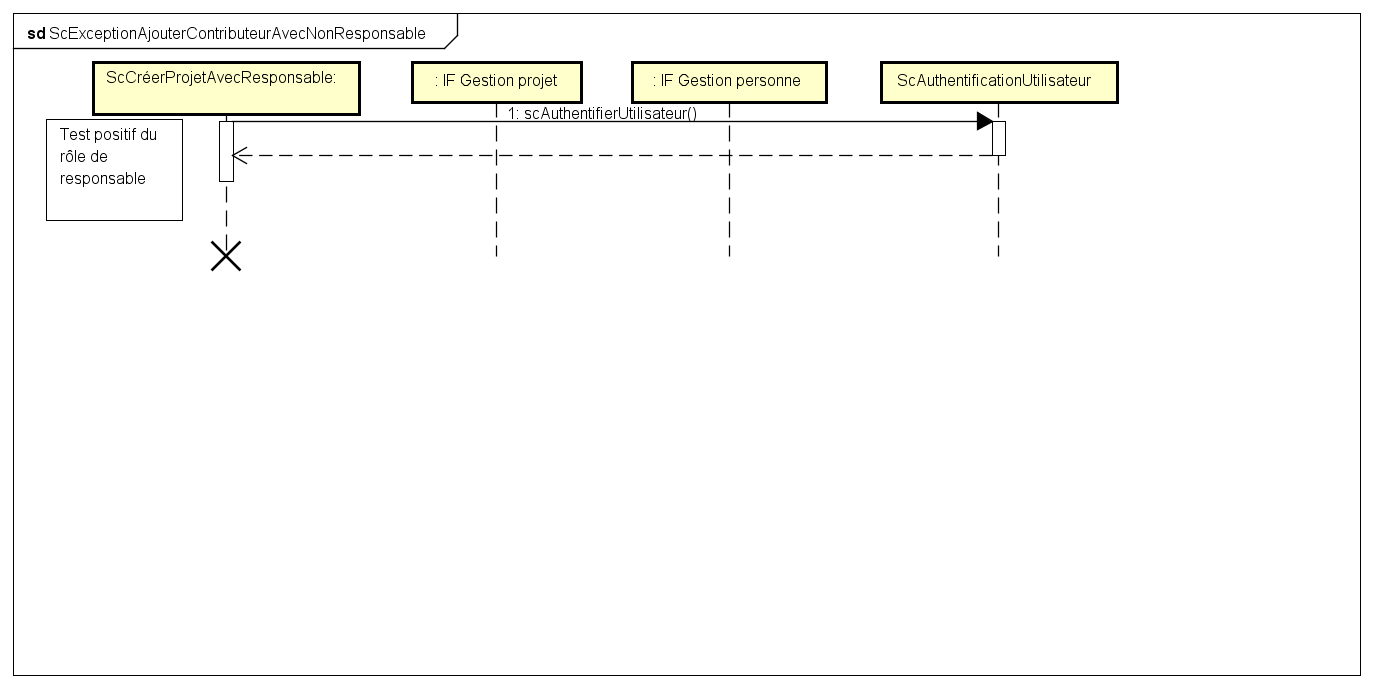
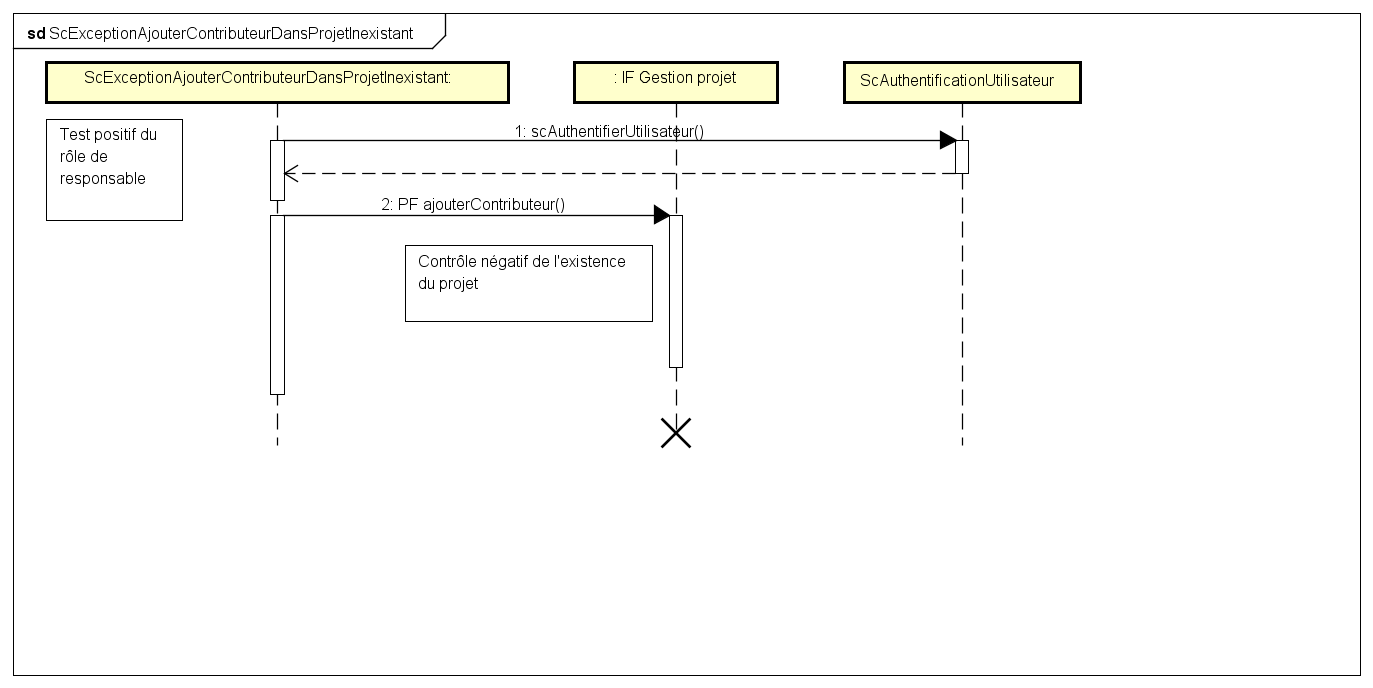


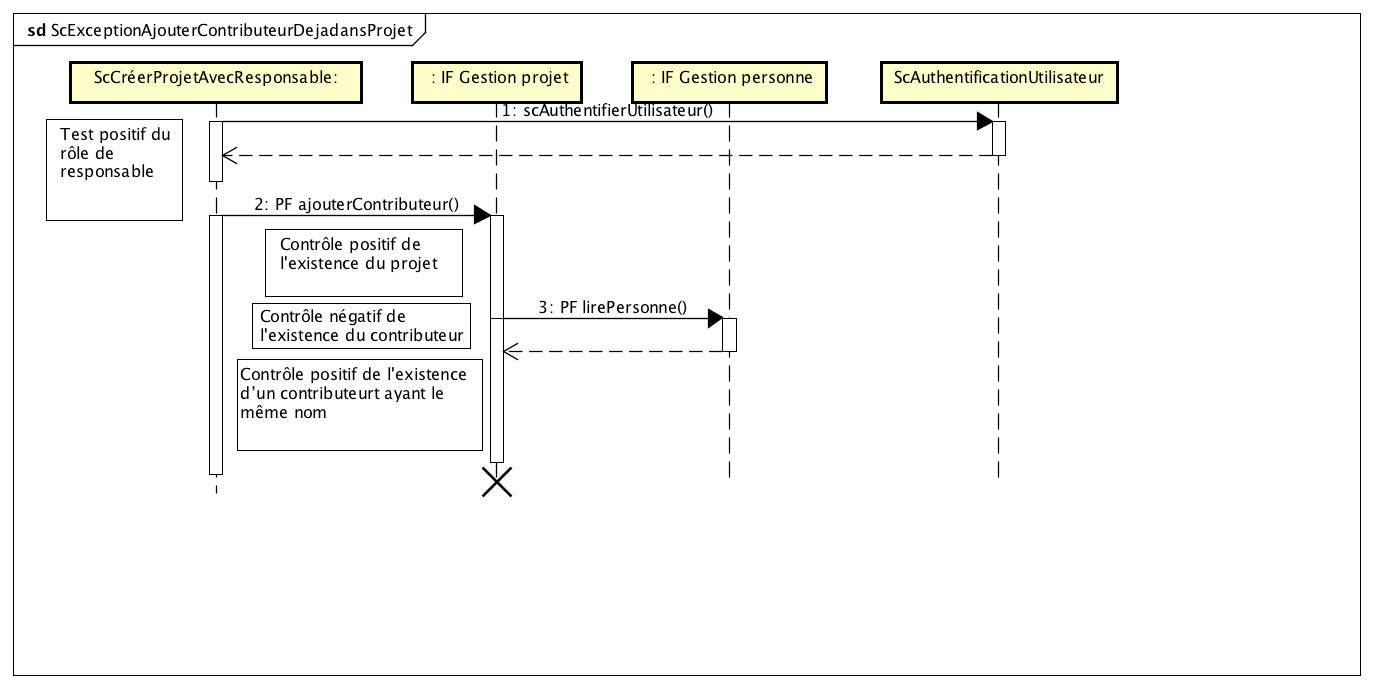
Figure 18 Diagramme de séquences fonctionnelles du scénario ScExceptionChangerContratSurProjetInexistant

Figure 19 Diagramme de séquences fonctionnelles du scénario ScExceptionChangerContratInexistant

Figure 20 Diagramme de séquences fonctionnelles du scénario ScAjouterContributeurAvecResponsable

Figure 21 Diagramme de séquences fonctionnelles du scénario ScExceptionAjouterContributeurAvecNonResponsable

Figure 22 Diagramme de séquences fonctionnelles du scénario ScExceptionAjouterContributeurDansProjetInexistant

Figure 23 Diagramme de séquences fonctionnelles du scénario ScExceptionAjouterContributeurDejadansProjet

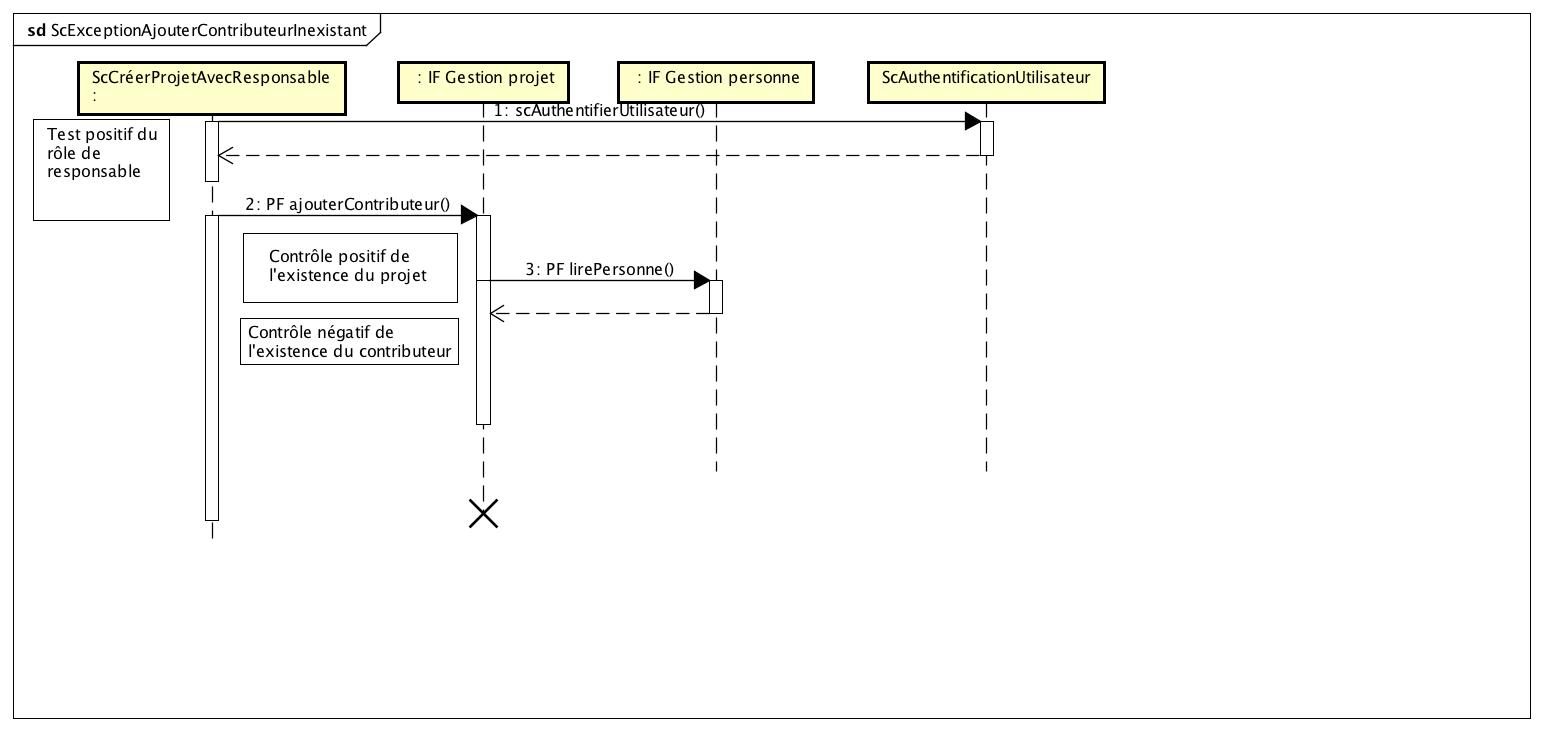
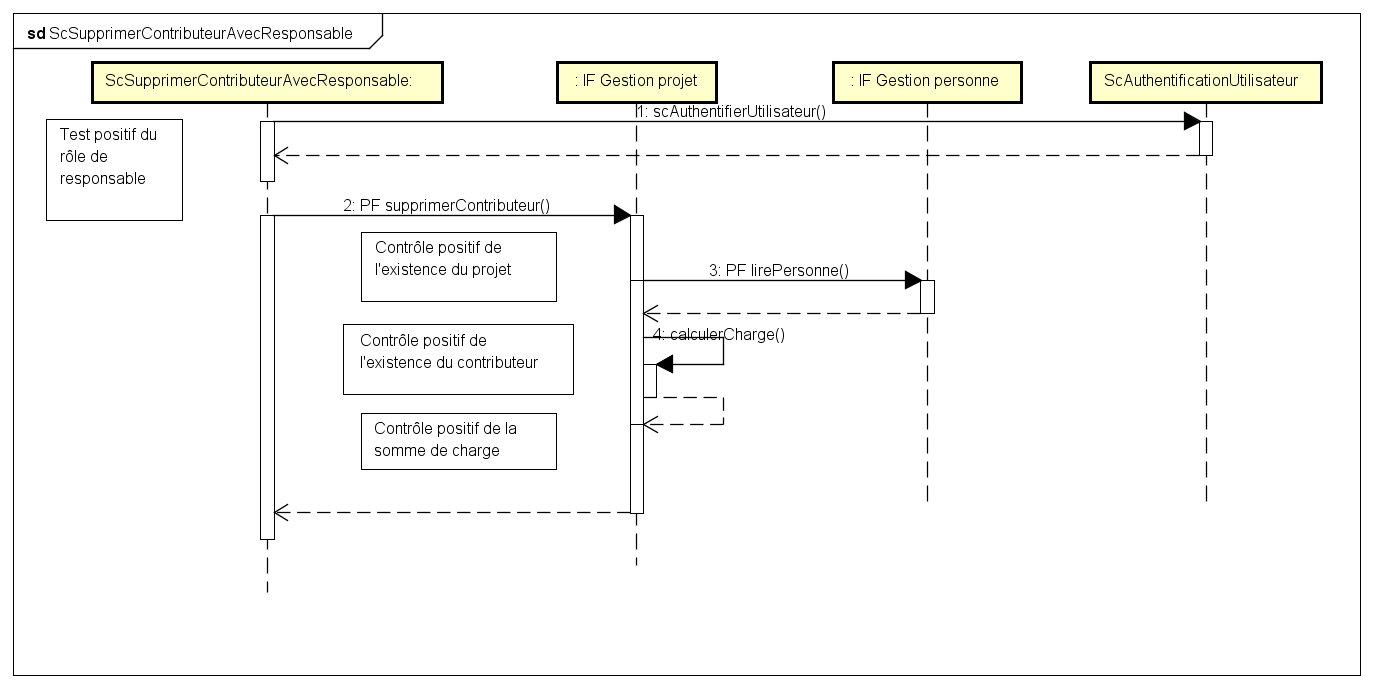


Figure 24 Diagramme de séquences fonctionnelles du scénario ScExceptionAjouterContributeurInexistant

Figure 25 Diagramme de séquences fonctionnelles du scénario ScSupprimerContributeurAvecResponsable

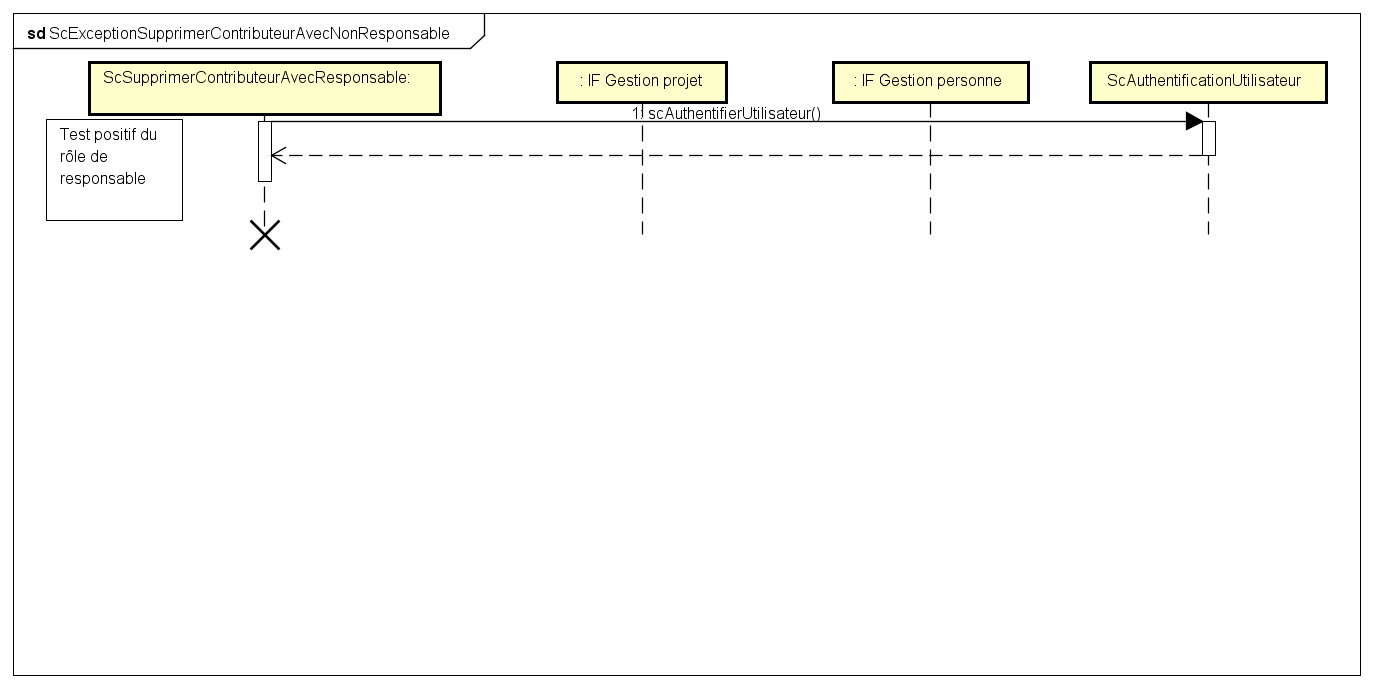


Figure 26 Diagramme de séquences fonctionnelles du scénario ScExceptionSupprimerContributeurAvecNonResponsable

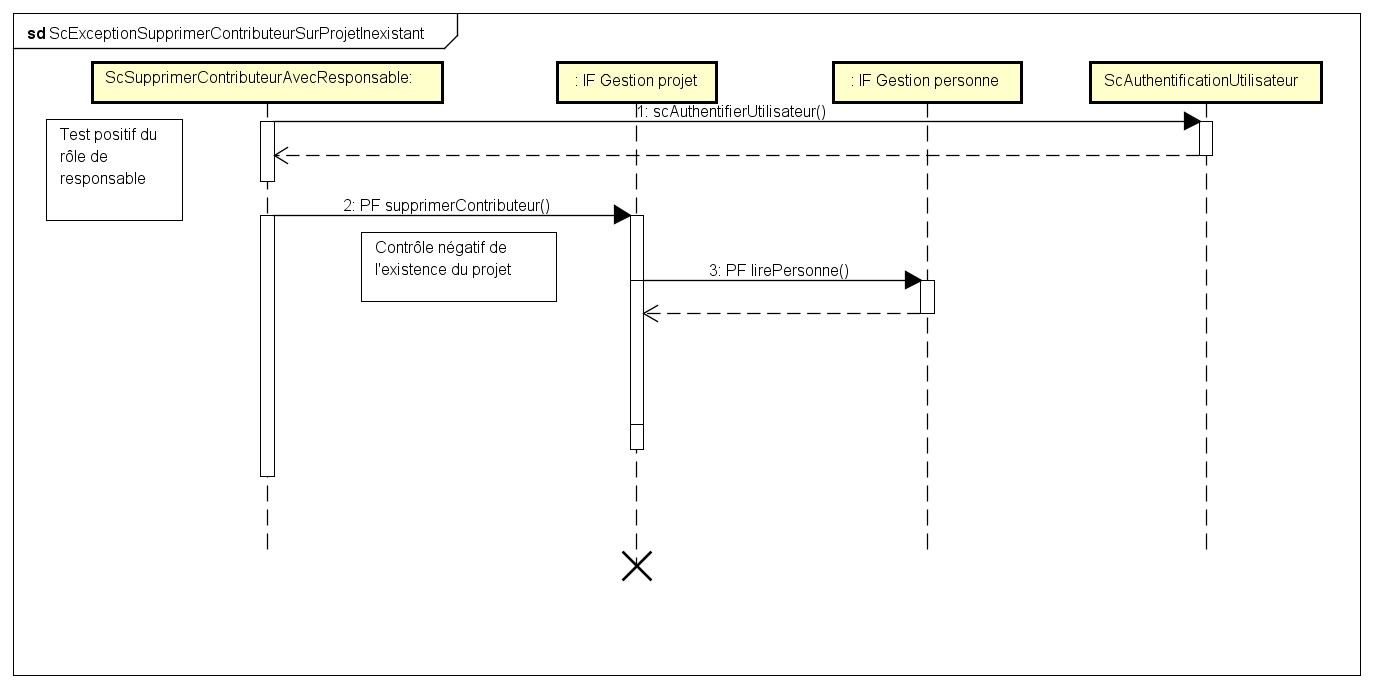
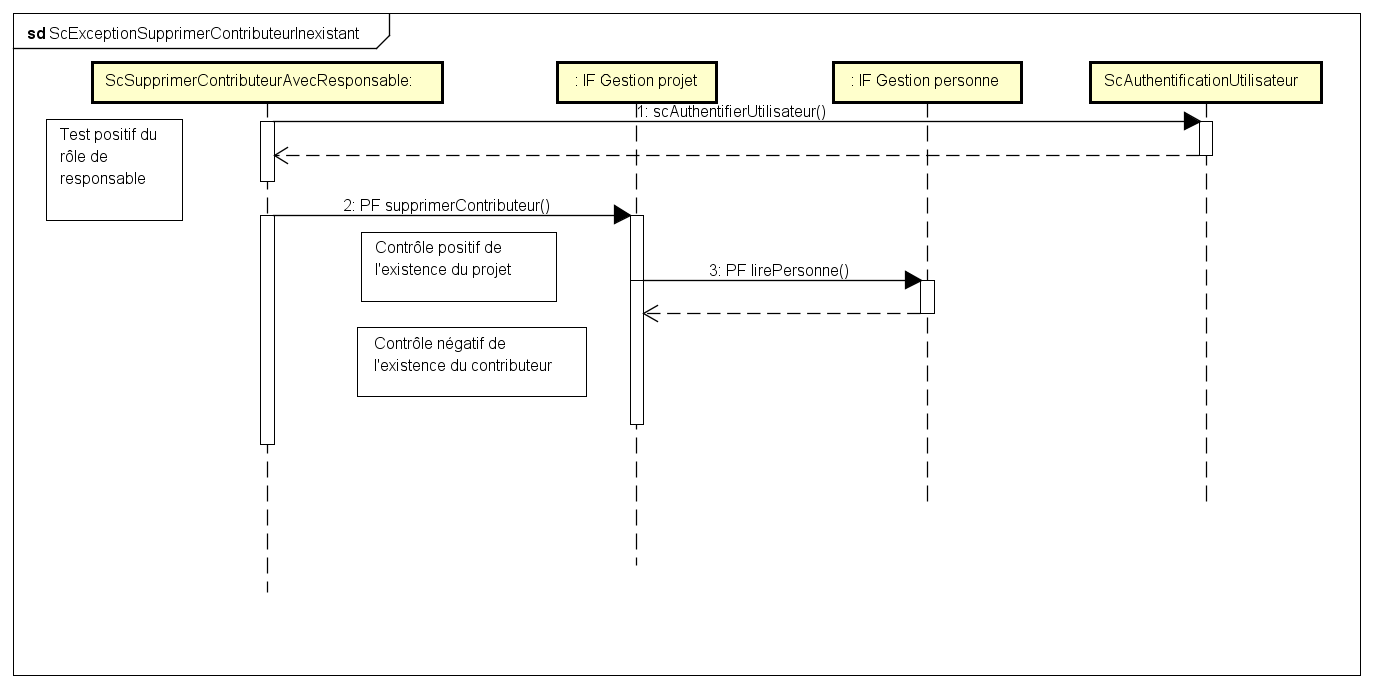
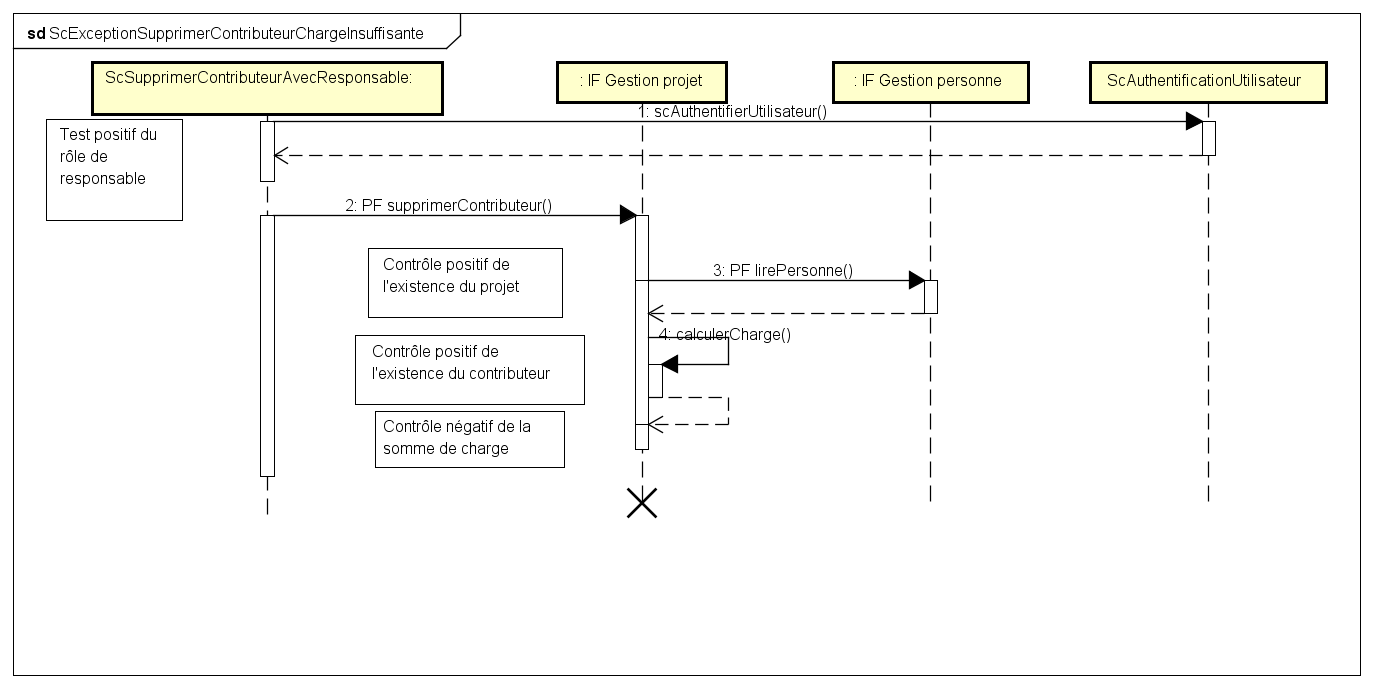
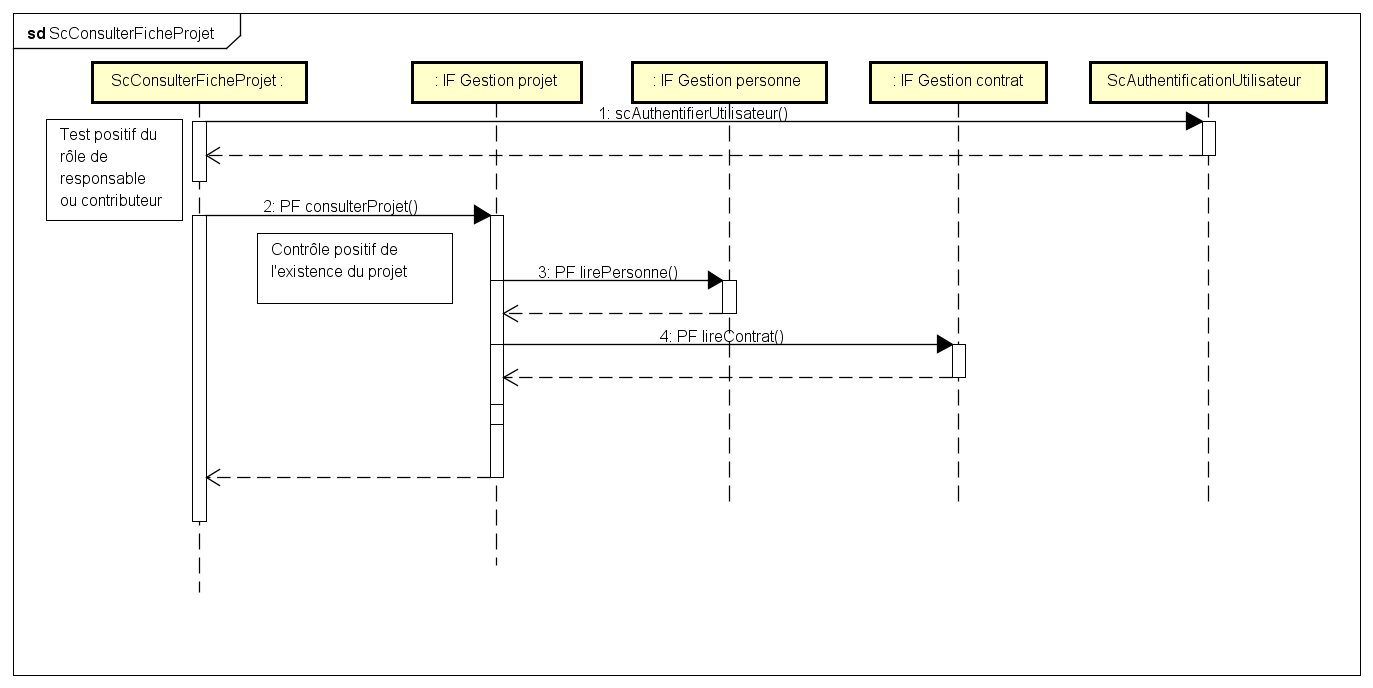
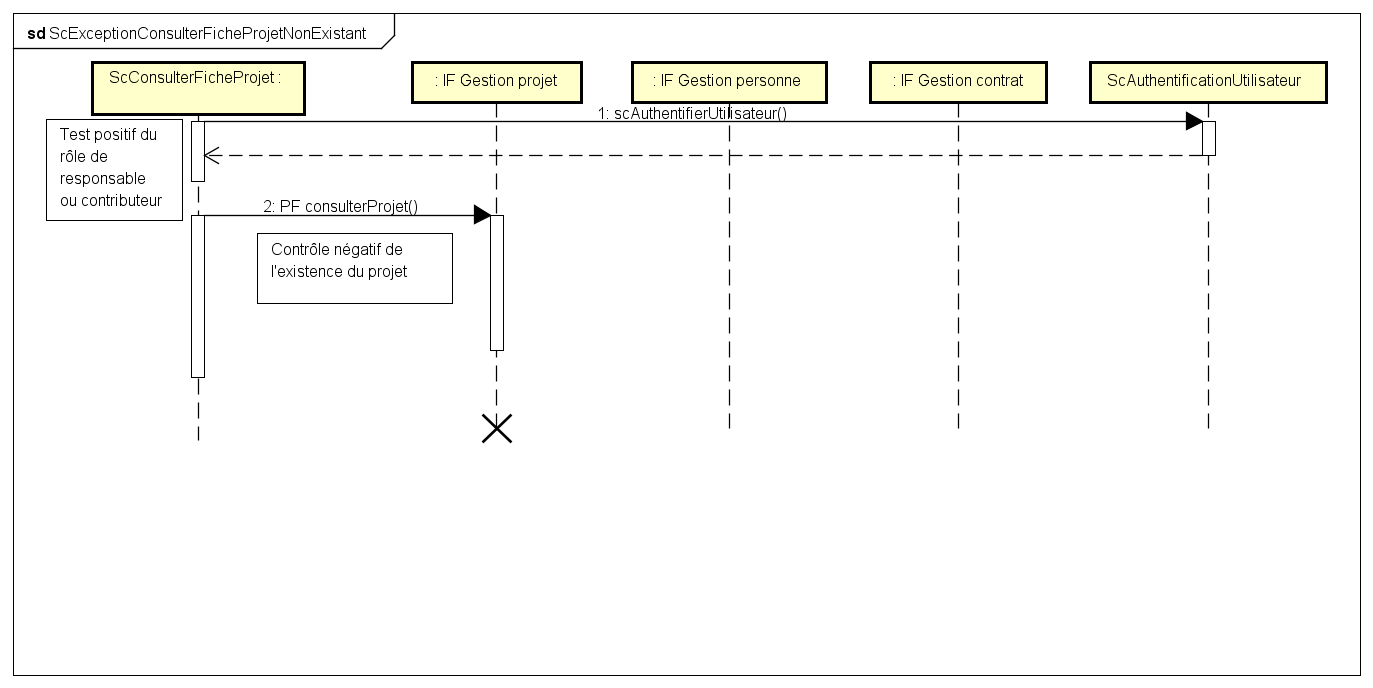
Figure 27 Diagramme de séquences fonctionnelles du scénario ScExceptionSupprimerContributeurSurProjetNonExistant

Figure 28 Diagramme de séquences fonctionnelles du scénario ScExceptionSupprimerContributeurNonExistant

Figure 29 Diagramme de séquences fonctionnelles du scénario ScExceptionSupprimerContributeurChargeInsuffisant

Figure 30 Diagramme de séquences fonctionnelles du scénario ScConsulterFichier

Figure 31 Diagramme de séquences fonctionnelles du scénario ScExceptionConsulterFichierProjetNonExistant

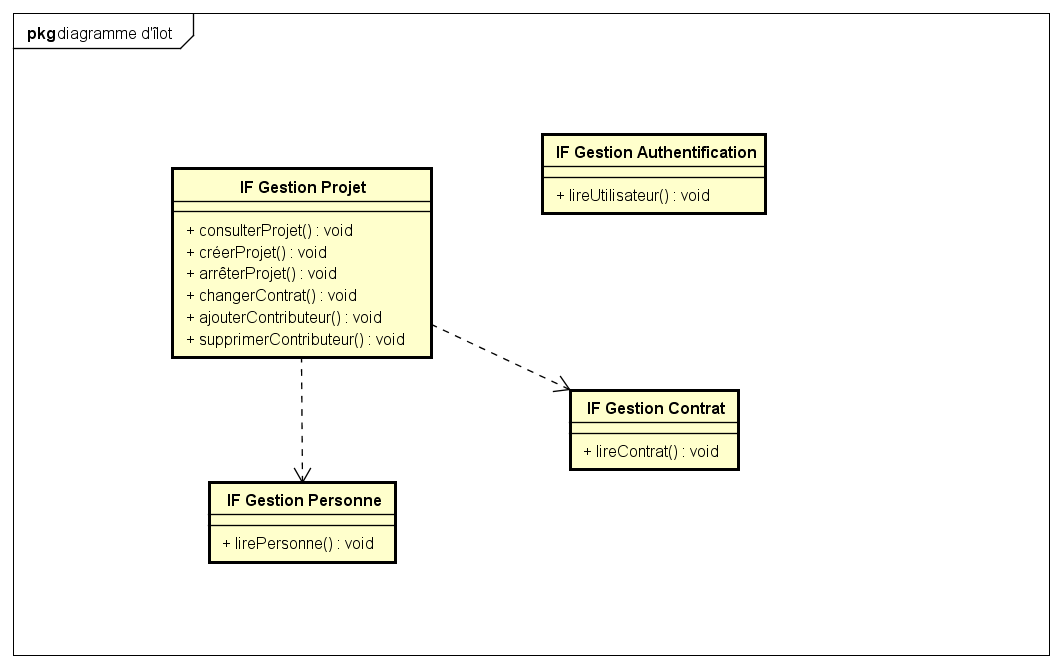
Le modèle des îlots fonctionnels est représenté par un diagramme de classes UML stéréotypées “îlot fonctionnel” dans la figure ci-dessous. Les dépendances entre îlots sont déduites des diagrammes de séquences fonctionnelles.

Figure 32 Modèle des îlots fonctionnels

## Données Fonctionnelles

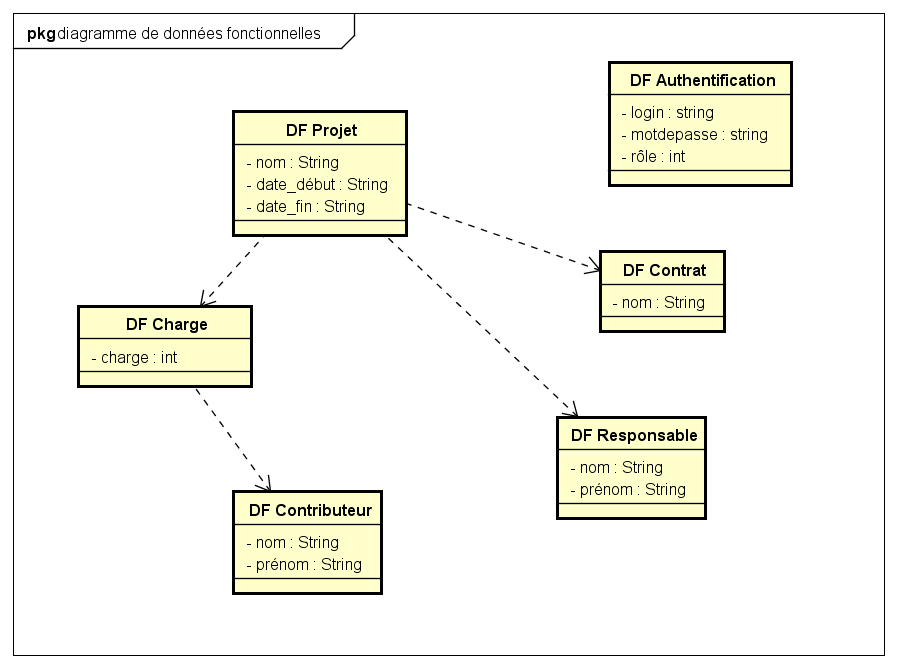
Le modèle des données fonctionnelles est représenté par un diagramme de class UML stéréotypes “donnée fonctionnelle” dans la figure ci-dessous. Chaque donnée et chaque dépendance entre donnée sont cohérentes avec le modèle des îlots fonctionnels.

Figure 33 Modèle des données fonctionnelles

Le Tableau 6 représenté le lien entre les îlots fonctionnels et les données fonctionnelles. Un îlot fonctionnel peut fournir une donnée fonctionnelle(F) ou utiliser une donnée fonctionnelle(U).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Donnée fonctionnelle**  **Îlot fonctionnelle** | DF Authentification | DF Projet | DF Contributeur | DF Responsable | DF Contrat | DF Charge |
| IF Gestion Personne |  |  | U | U |  |  |
| IF Gestion Projet |  | U/F |  |  |  | U/F |
| IF Gestion Contrat |  |  |  |  | U/F |  |
| IF Gestion Authentification | U |  |  |  |  |  |

Tableau 6

La traçabilité des entité participantes aux cas d’utilisation est représentés dans le Tableau 7 :

|  |  |  |
| --- | --- | --- |
| Entité participante | Attribut | Donnée fonctionnelle |
| Utilisateur | nom | DF Authentification |
| prénom |
| rôle |
| Projet | nom | DF Projet |
| date\_début |
| date\_fin |
| Contributeur | nom | DF Contributeur |
| prénom |
| Responsable | nom | DF Responsable |
| prénom |
| Contrat | nom | DF Contrat |
| Charge | quantity | DF Charge |

Tableau 7

# Architecture Applicative

## Composant applicatives

Les composants applicatifs du logiciel sont décrits au niveau logiciel dans le Tableau 8.

|  |  |  |
| --- | --- | --- |
| **Îlot fonctionnel** | **Parcelle fonctionnelle** | **Logiciel** |
| IF Gestion Authentification | lireUtilisateur | Direction scientifique |
| IF Gestion Projet | créerProjet |
| consulterProjet |
| arrêterProjet |
| changerContrat |
| ajouterContributeur |
| supprimerContributeur |
| IF Gestion Personne | lirePersonne |
| IF Gestion Contrat | lireContrat |

Tableau 8

La décomposition par protection sur chaque couche est décrite dans le Tableau 9.

|  |  |  |
| --- | --- | --- |
| **Logiciel** | **Couche** | **Composant applicatif** |
| Direction Scientifique | Présentation | CA Test Direction Scientifique |
| Service | CA Authentifier Utilisateur |
| CA Créer Projet |
| CA Arrêter Projet |
| CA Changer Contrat |
| CA Ajouter Contributeur |
| CA Supprimer Contributeur |
| CA Consulter Fichier Projet |
| Données | CA Authentification |
| CA Projet |
| CA Contributeur |
| CA JoinProjetContributeur |
| CA Contrat |
| CA Responsable |

Tableau 9

## Dépendance des composants applicatifs

Les dépendances des composants applicatifs répertoriés précédemment sont conçues à partir des scénarios spécifiés en analyse fonctionnelle et des diagrammes de séquences conçu en architecture fonctionnelle qui sont à réaliser. C'est une approche dynamique de l'architecture applicatives. La traçabilité des scénarios dans les diagrammes de séquences applicatives est reportée dans le Tableau 10.

|  |  |
| --- | --- |
| **Scénario** | **Diagramme de séquences applicatives** |
| ScAuthentifierUtilisateur | Figure 34 |
| ScExceptionAuthentifierUtilisateurInexistant | Figure 35 |
| ScCréerProjetAvecResponsable | Figure 36 |
| ScExceptionCréerProjetAvecNonResponsable | Figure 37 |
| ScExceptionCréerProjetExistant | Figure 38 |
| ScExceptionCréerProjetQuantitéNonSuffisant | Figure 39 |
| ScExceptionCréerProjeSansContratExistant | Figure 40 |
| ScArrêterProjetCourantAvecResponsable | Figure 41 |
| ScExceptionArrêterProjetAvecNonResponsable | Figure 42 |
| ScExceptionArrêterProjetExpiré | Figure 43 |
| ScExceptionArrêterProjetInexistant | Figure 44 |
| ScChangerContratAvecResponsable | Figure 45 |
| ScExceptionChangerContratAvecNonResponsable | Figure 46 |
| ScExceptionChangerContratSurProjetInexistant | Figure 47 |
| ScExceptionChangerContratInexistant | Figure 48 |
| ScAjouterContributeurAvecResponsable | Figure 49 |
| ScExceptionAjouterContributeurAvecNonResponsable | Figure 50 |
| ScExceptionAjouterContributeurDansProjetInexistant | Figure 51 |
| ScExceptionAjouterContributeurDejadansProjet | Figure 52 |
| ScExceptionAjouterContributeurInexistant | Figure 53 |
| ScSupprimerContributeurAvecResponsable | Figure 54 |
| ScExceptionSupprimerContributeurAvecNonResponsable | Figure 55 |
| ScExceptionSupprimerContributeurDansProjetInexistant | Figure 56 |
| ScExceptionSupprimerContributeurInexistant | Figure 57 |
| ScExceptionChargeInsuffisant | Figure 58 |
| ScConsulterfichierProjet | Figure 59 |
| ScExceptionConsulterfichierProjetNonExistant | Figure 60 |

Tableau 10

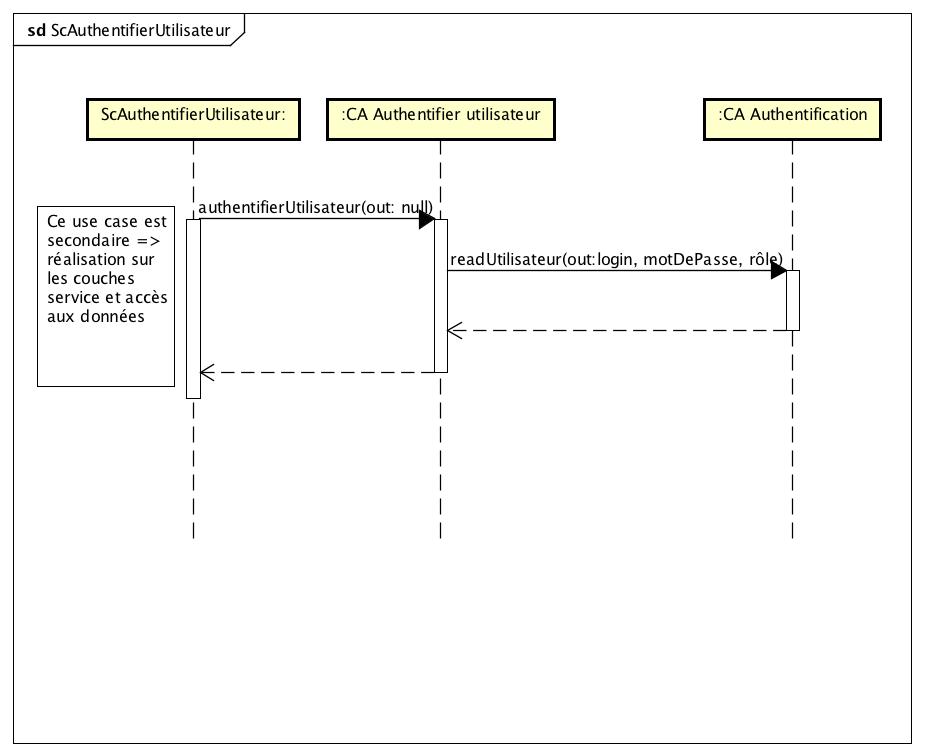


Figure 34 Diagramme de séquences applicatives du scénario ScAuthentifierUtilisateur

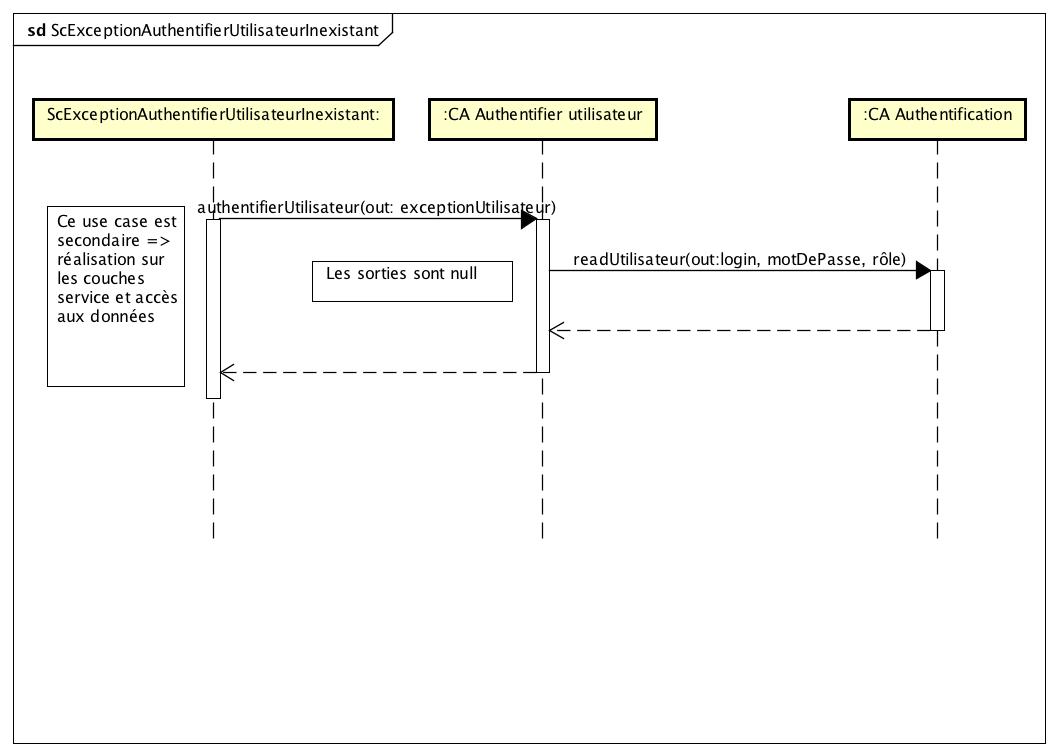
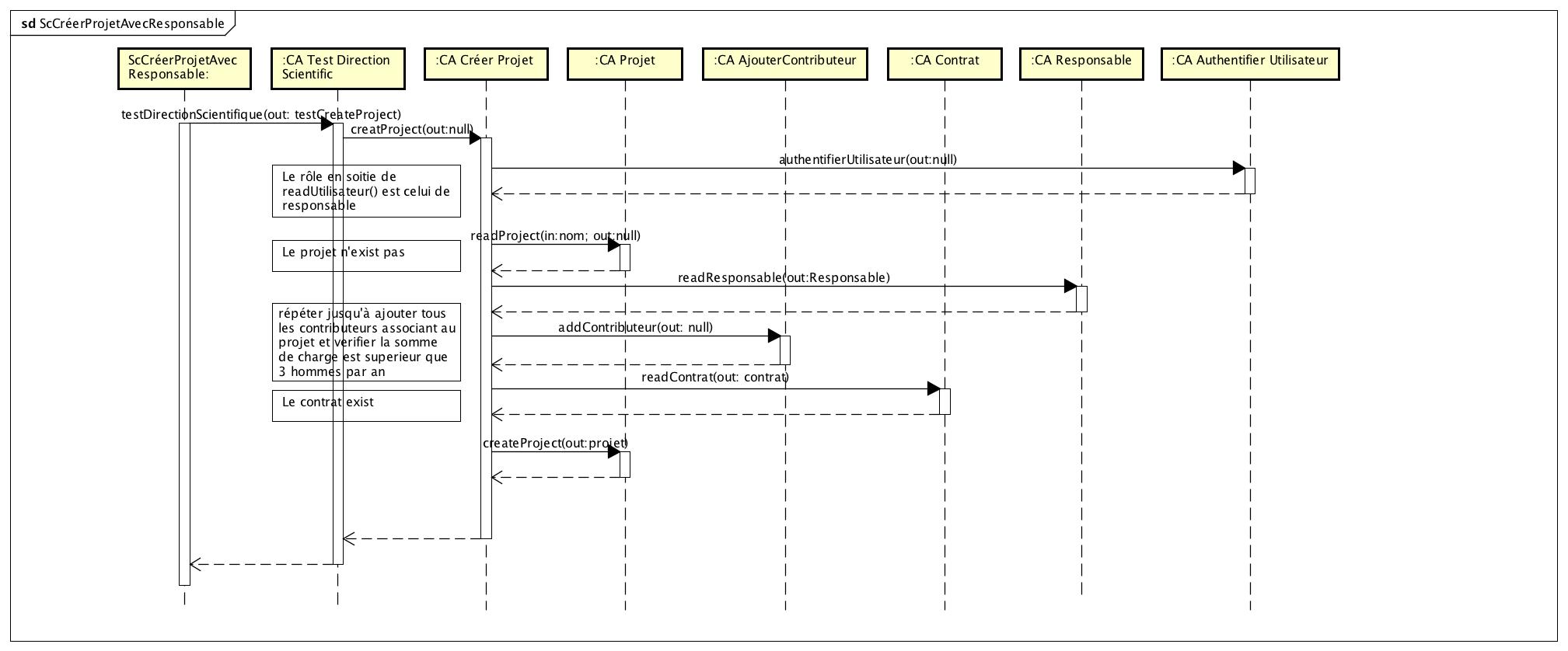


Figure 35 Diagramme de séquences applicatives du scénario ScExceptionAuthentifierUtilisateurInexistant

Figure 36 Diagramme de séquences applicatives du scénario ScCréerProjetAvecResponsable



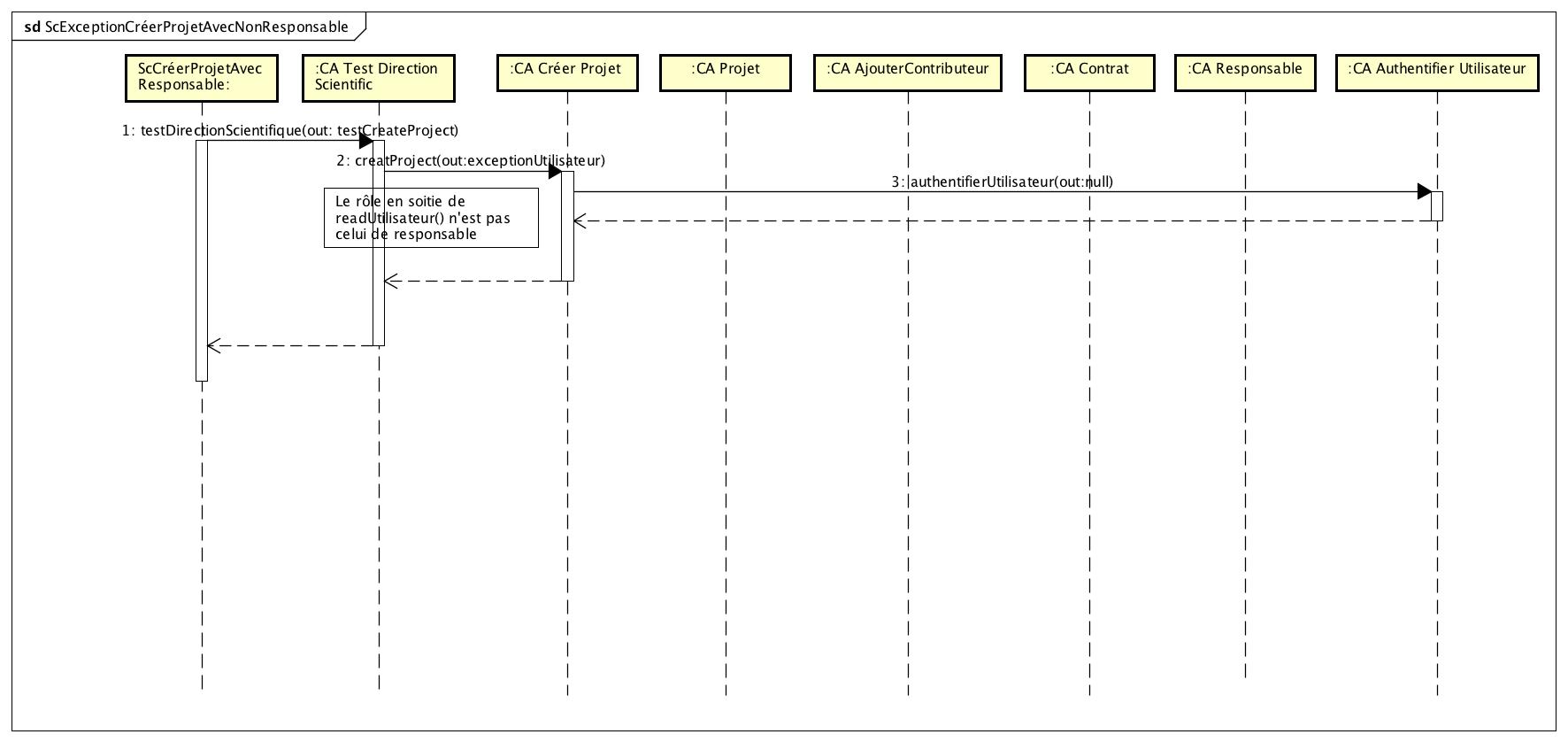


Figure 37 Diagramme de séquences applicatives du scénario ScExceptionCréerProjetAvecNonResponsable

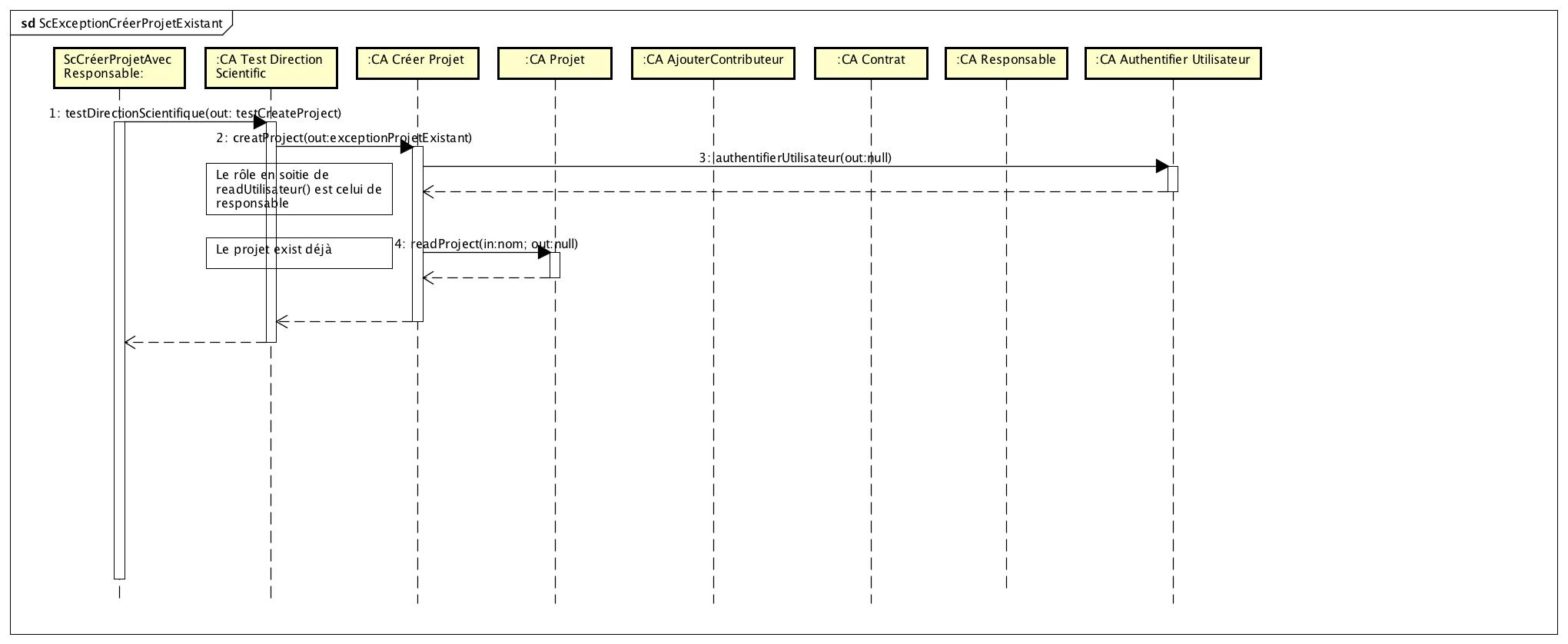


Figure 38 Diagramme de séquences applicatives du scénario ScExceptionCréerProjetExistant

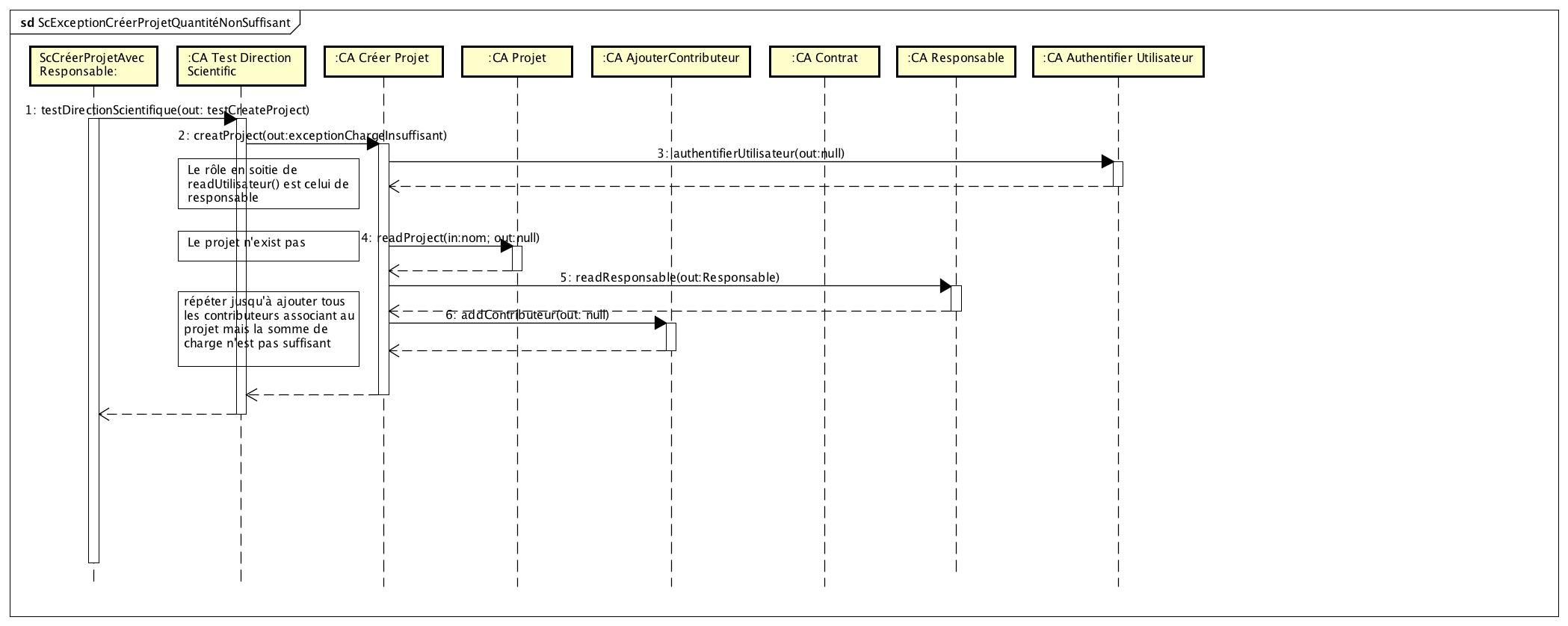


Figure 39 Diagramme de séquences applicatives du scénario ScExceptionCréerProjetQuantitéNonSuffisant

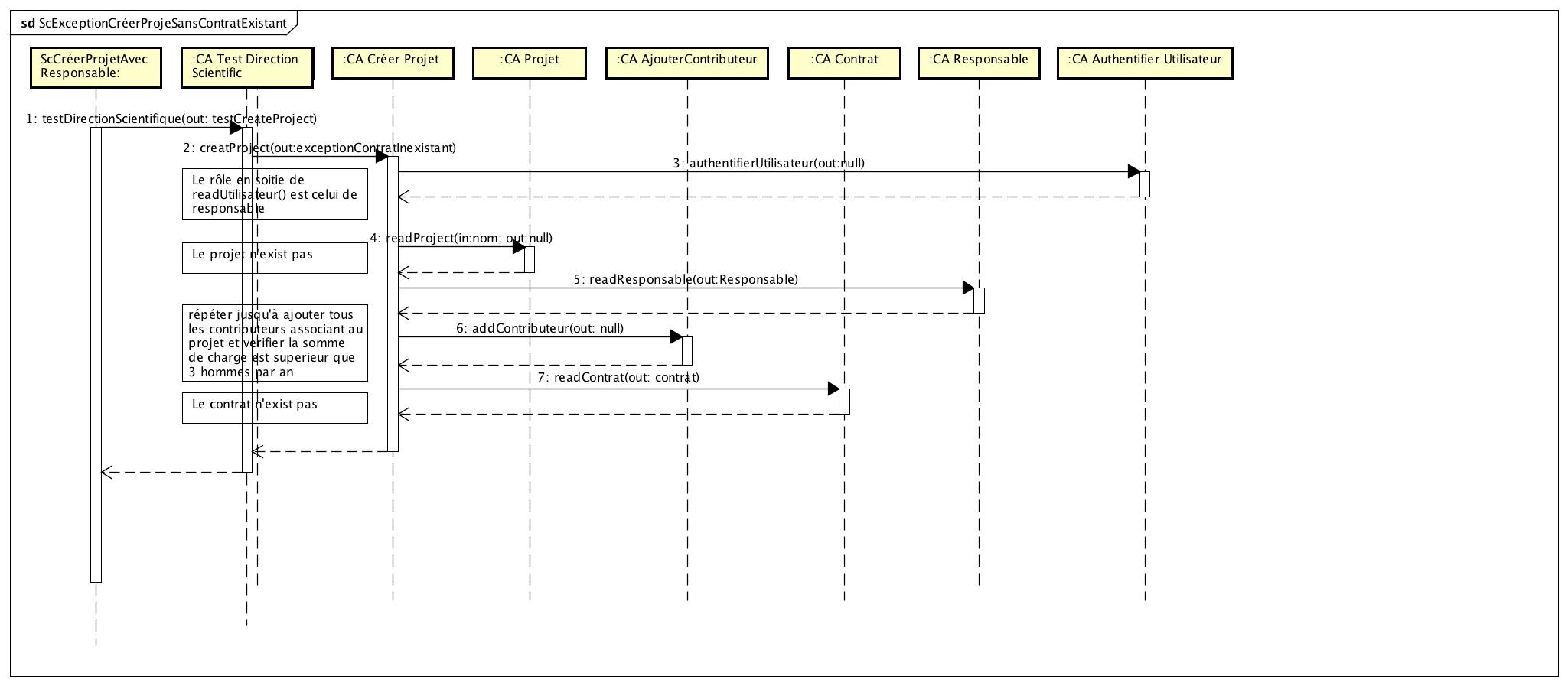


Figure 40 Diagramme de séquences applicatives du scénario ScExceptionCréerProjeSansContratExistant

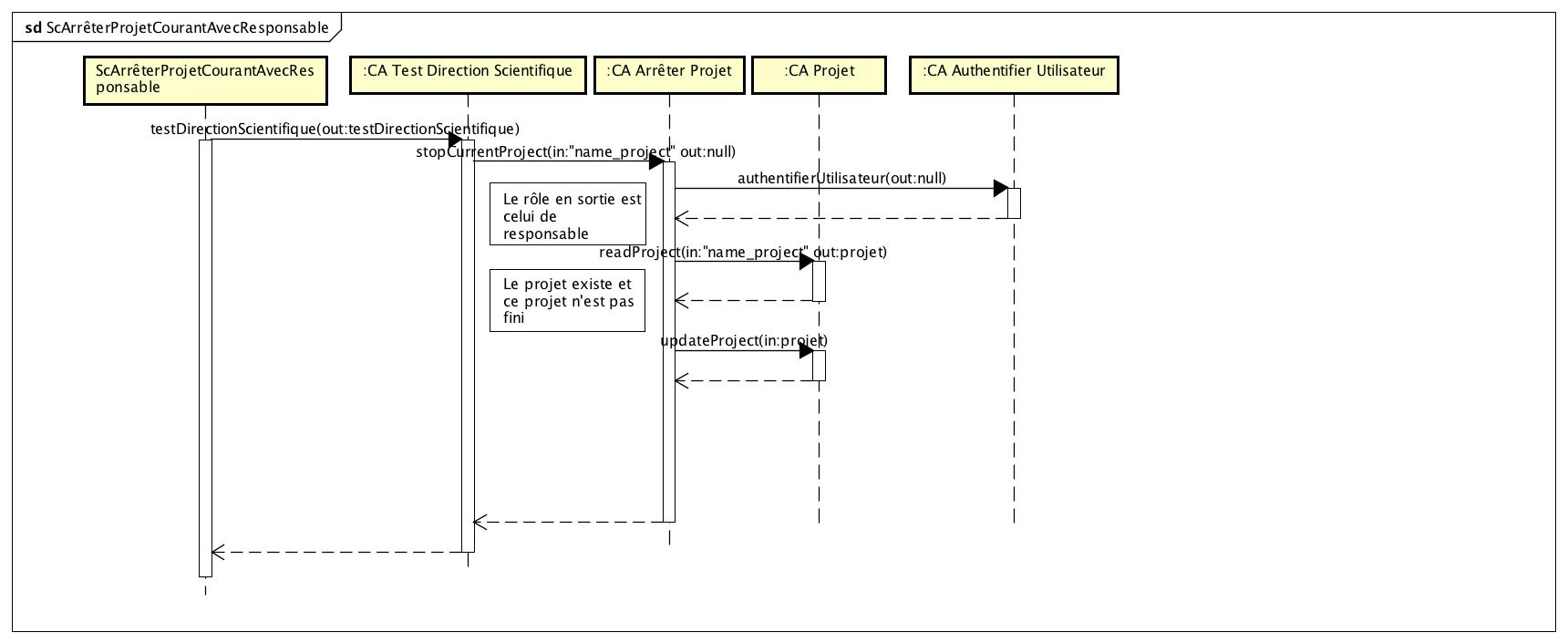


Figure 41 Diagramme de séquences applicatives du scénario ScArrêterProjetCourantAvecResponsable

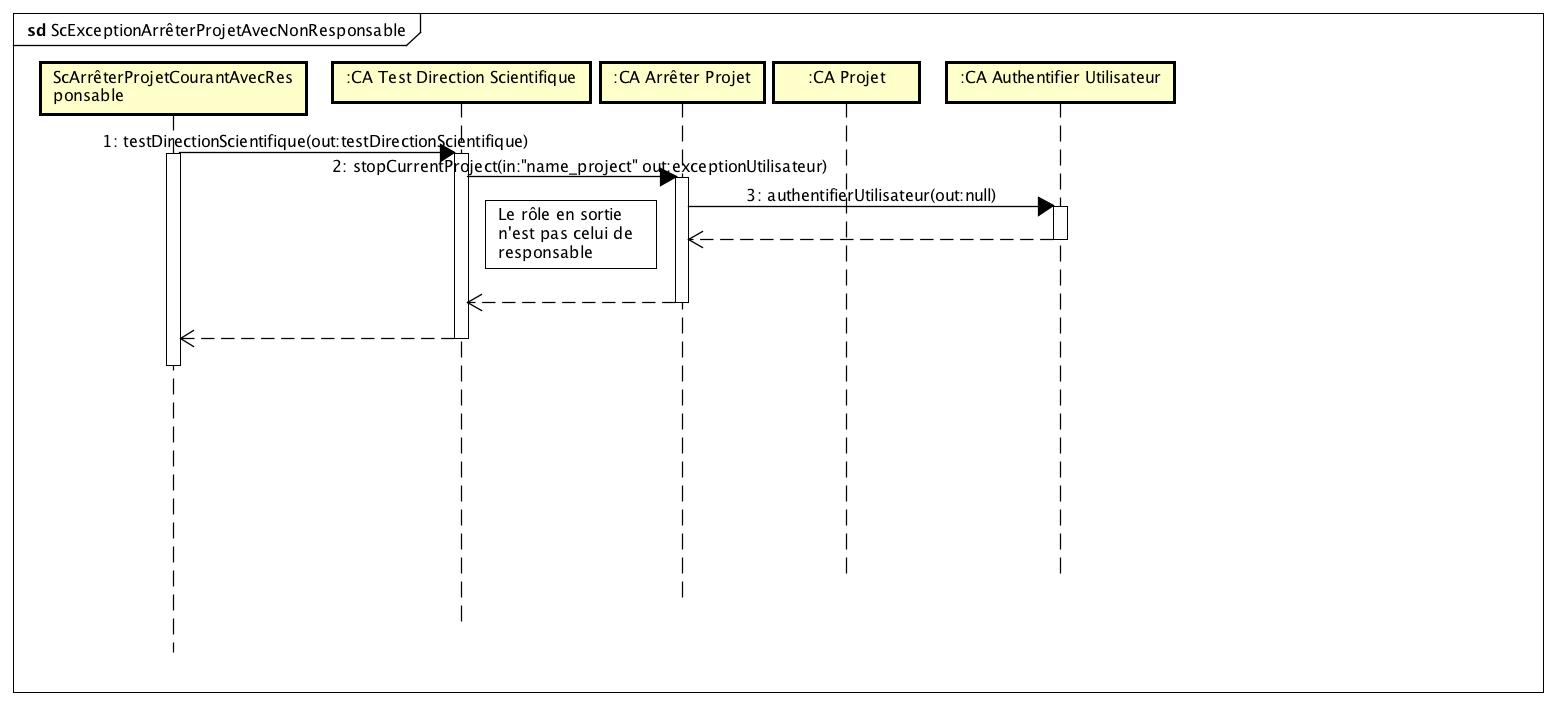


Figure 42 Diagramme de séquences applicatives du scénario ScArrêterProjetCourantAvecNonResponsable

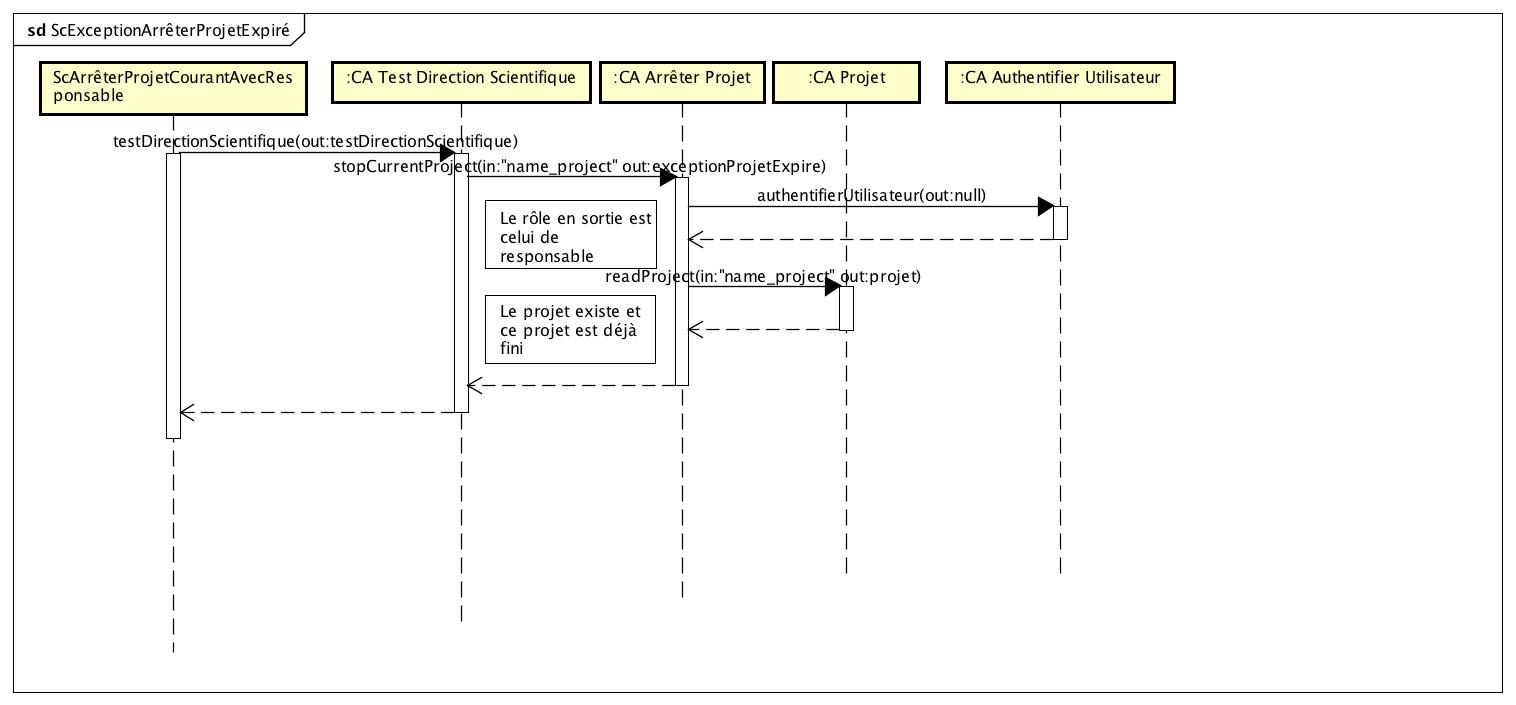


Figure 43 Diagramme de séquences applicatives du scénario ScExceptionArrêterProjetExpiré

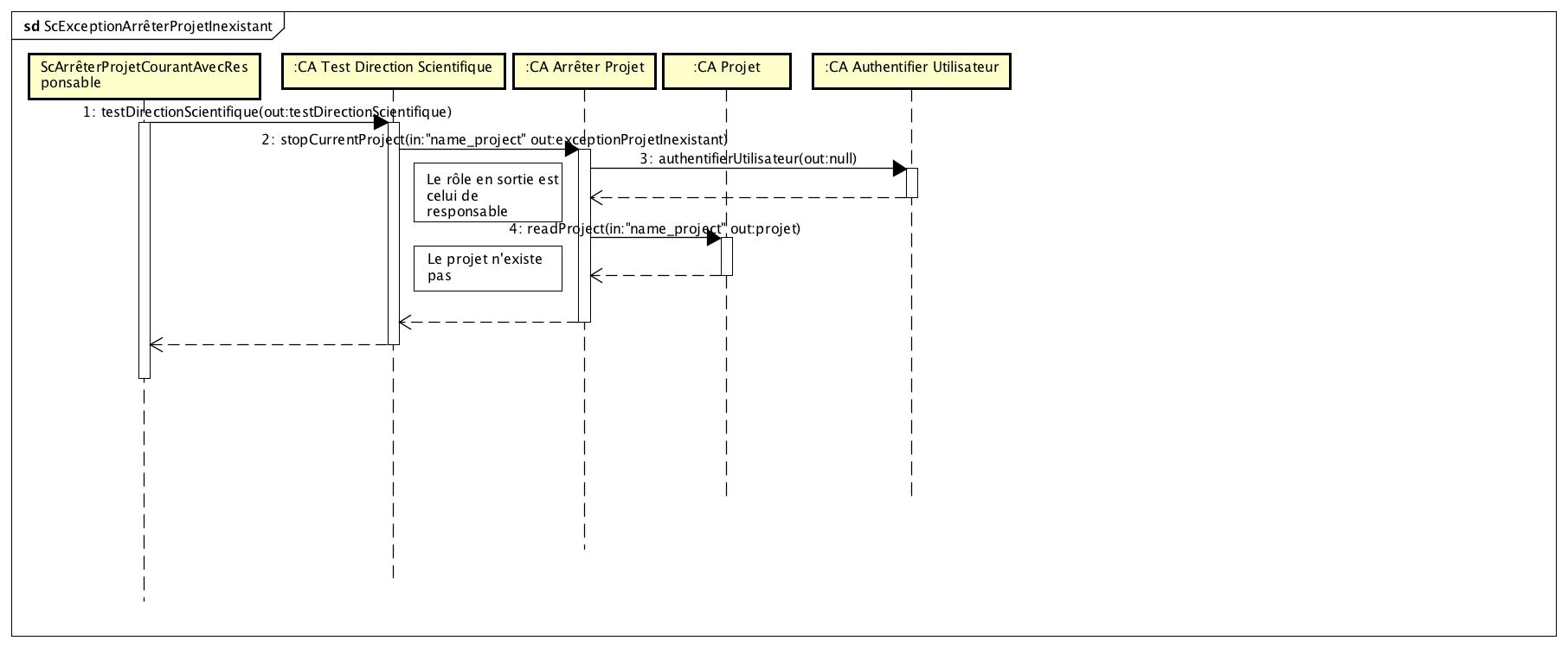


Figure 44 Diagramme de séquences applicatives du scénario ScExceptionArrêterProjetInexistant

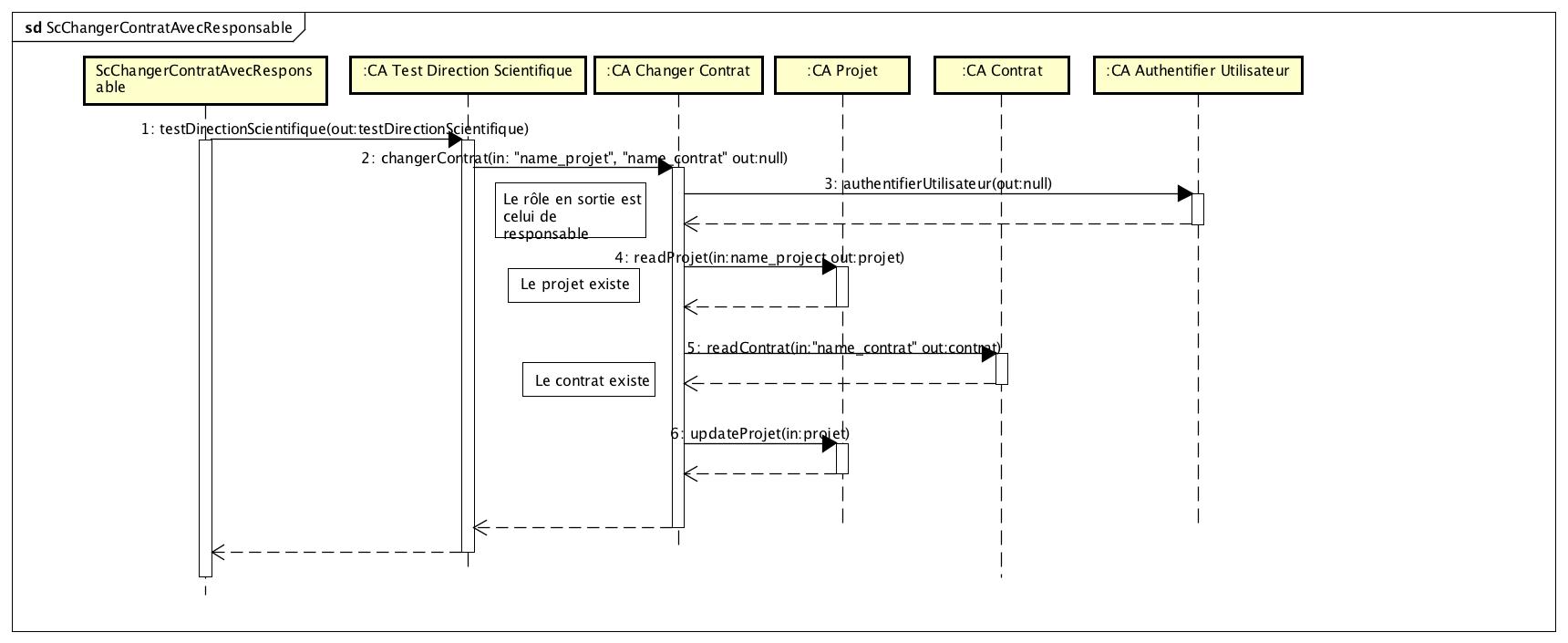


Figure 45 Diagramme de séquences applicatives du scénario ScChangerContratAvecResponsable

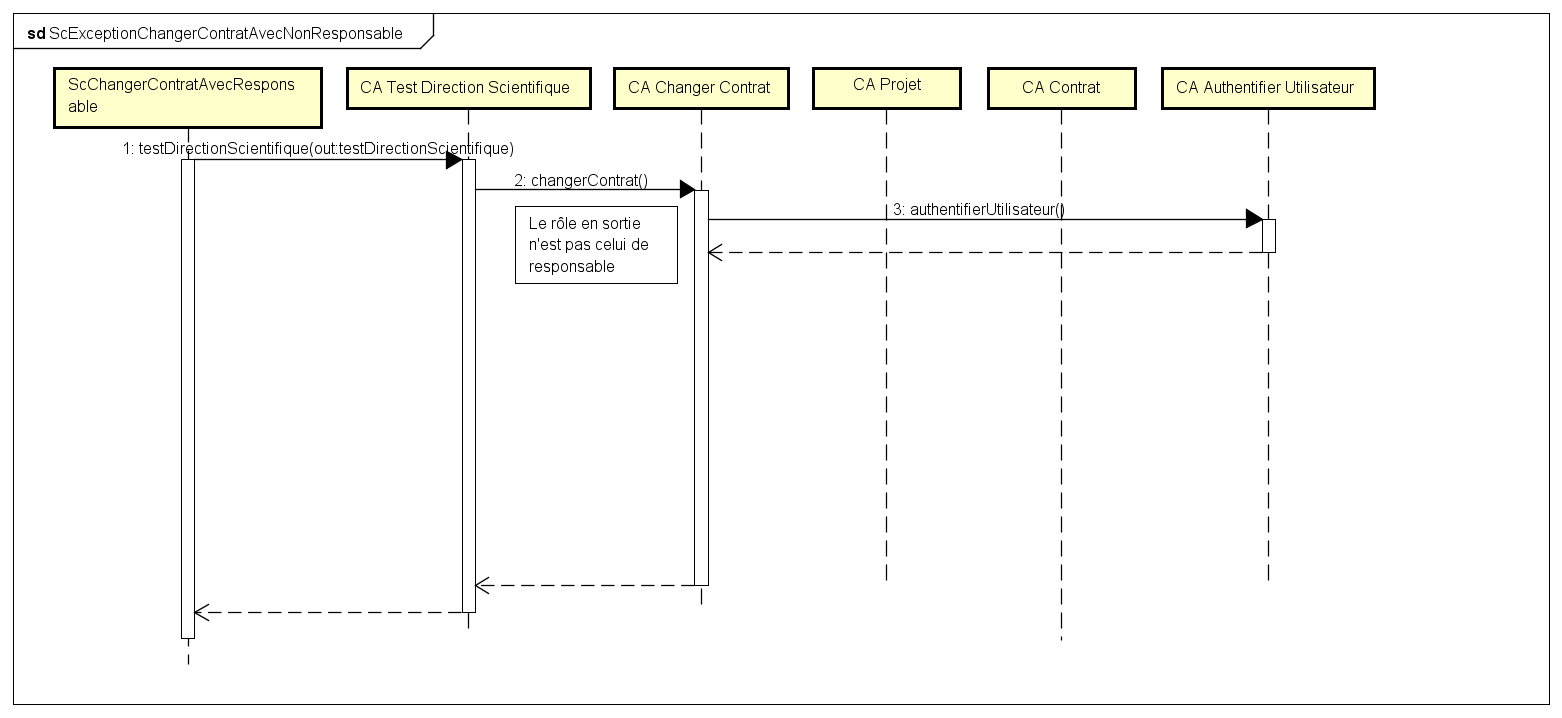


Figure 46 Diagramme de séquences applicatives du scénario ScExceptionChangerContratAvecNonResponsable

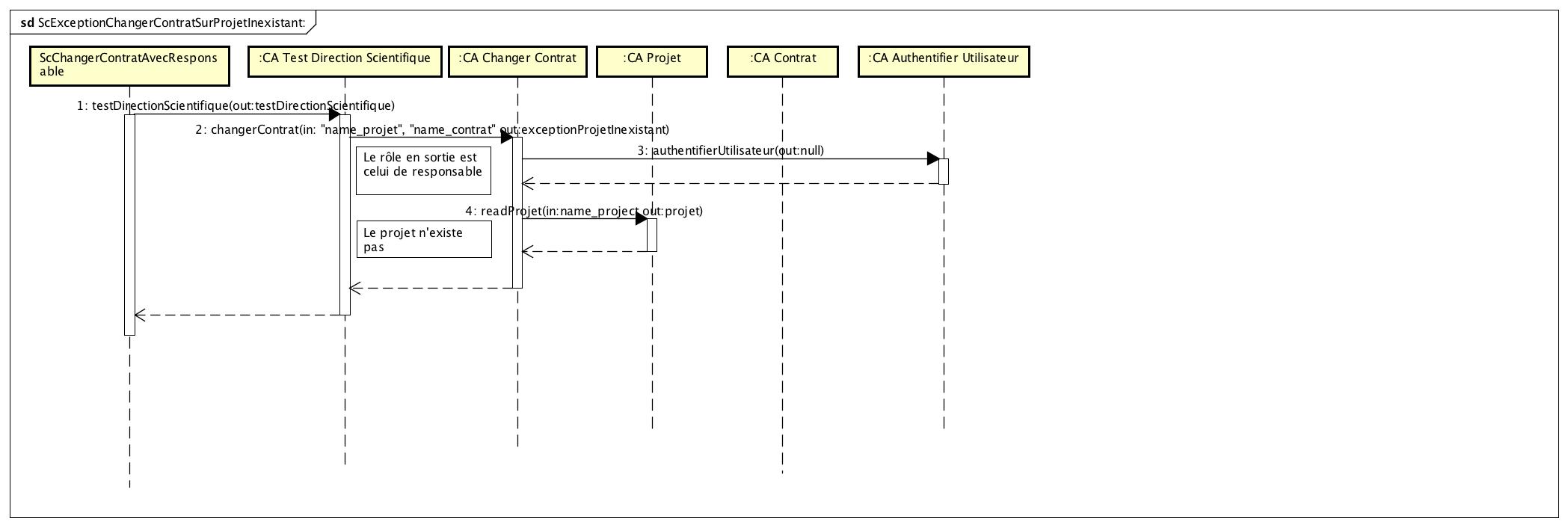


Figure 47 Diagramme de séquences applicatives du scénario ScExceptionChangerContratSurProjetInexistant

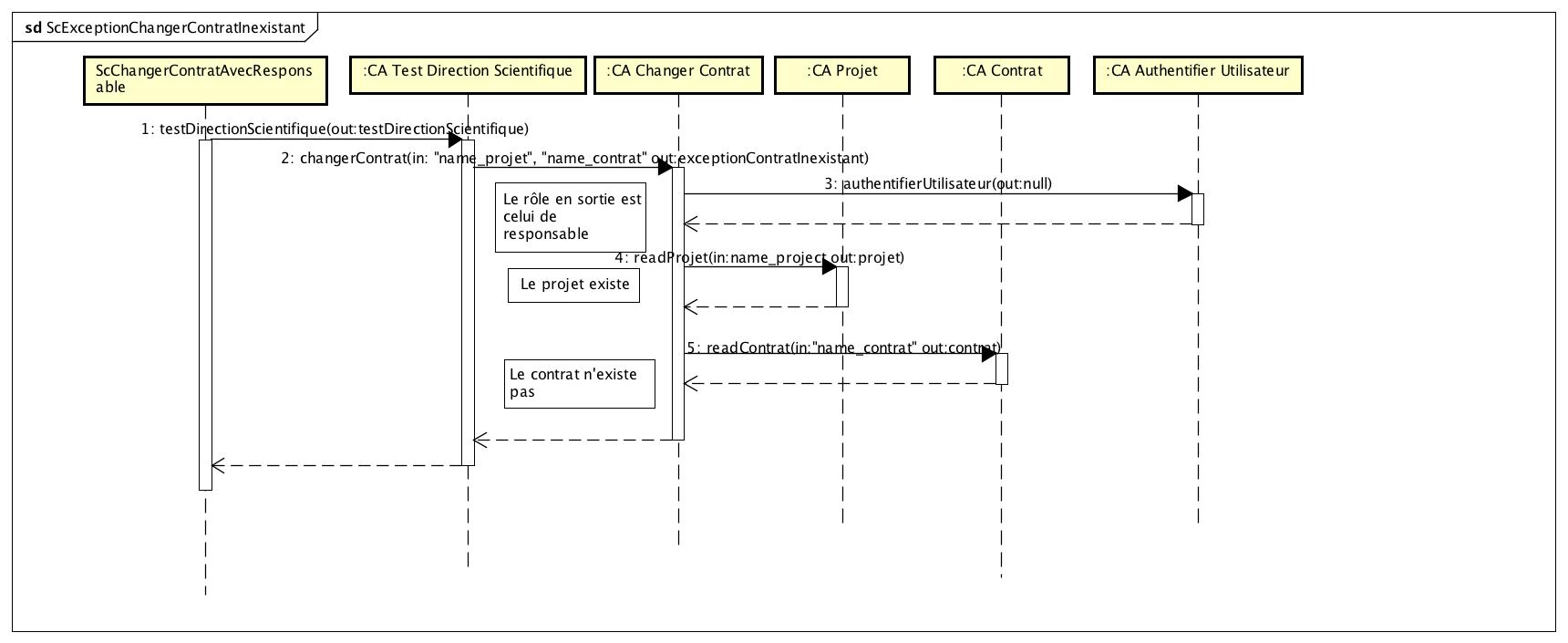


Figure 48 Diagramme de séquences applicatives du scénario ScExceptionChangerContratInexistant

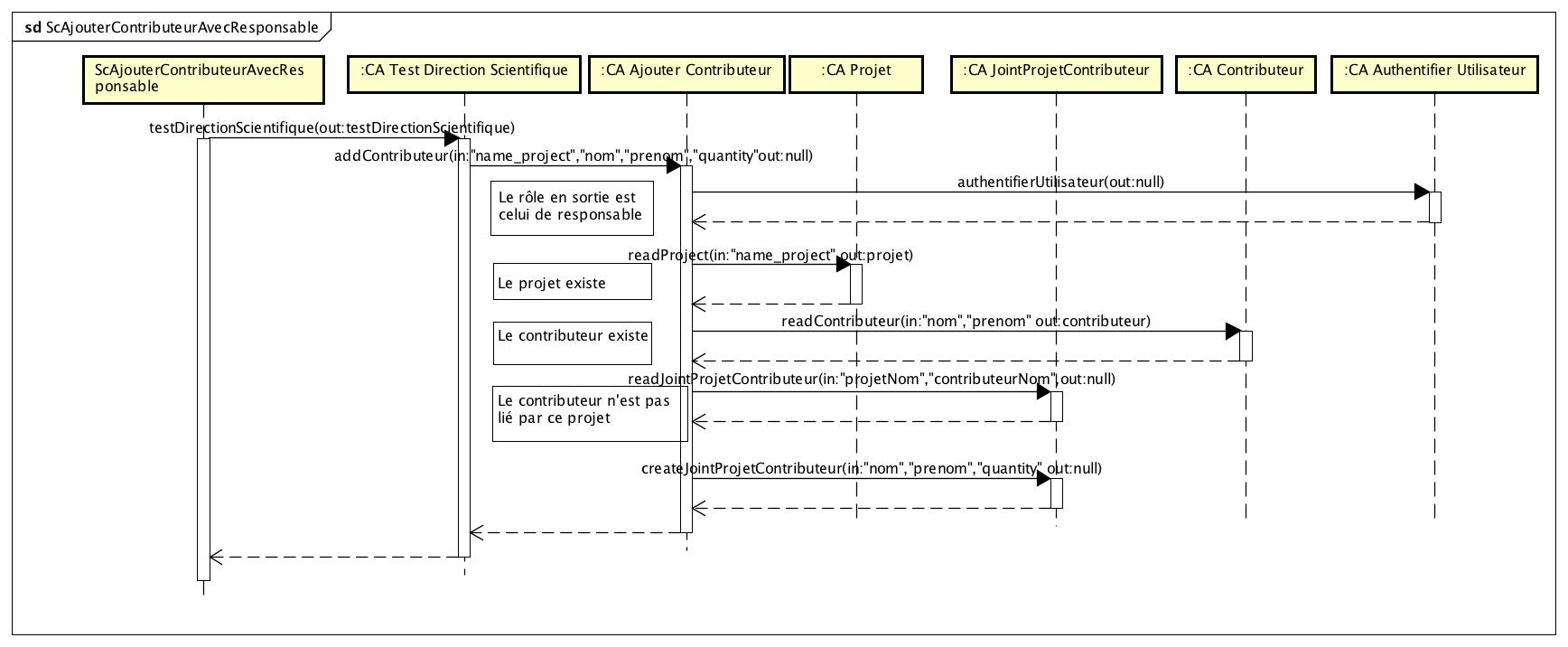


Figure 49 Diagramme de séquences applicatives du scénario ScAjouterContributeurAvecResponsable

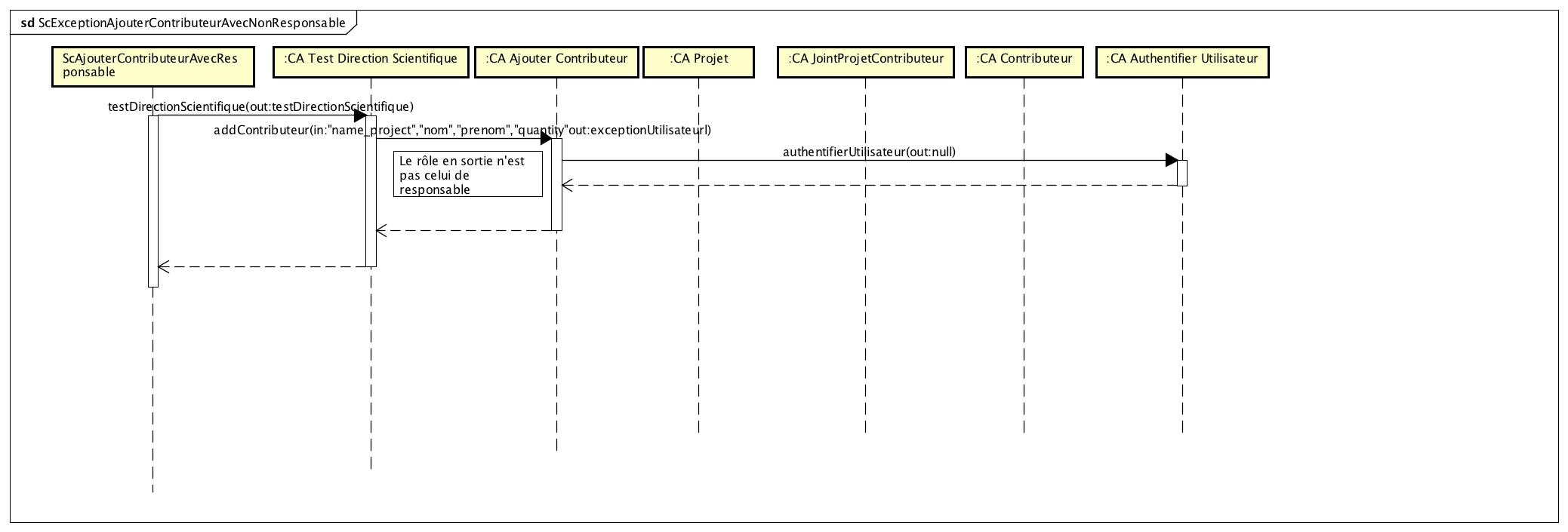


Figure 50 Diagramme de séquences applicatives du scénario ScAjouterContributeurAvecNonResponsable

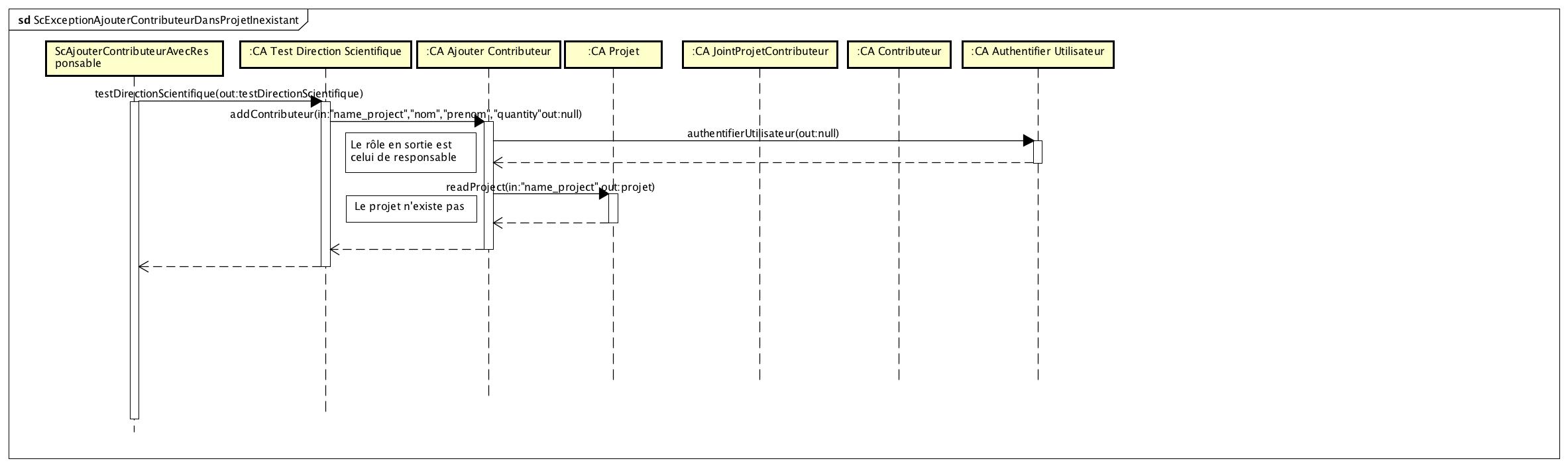


Figure 51 Diagramme de séquences applicatives du scénario ScExceptionAjouterContributeurDansProjetInexistant

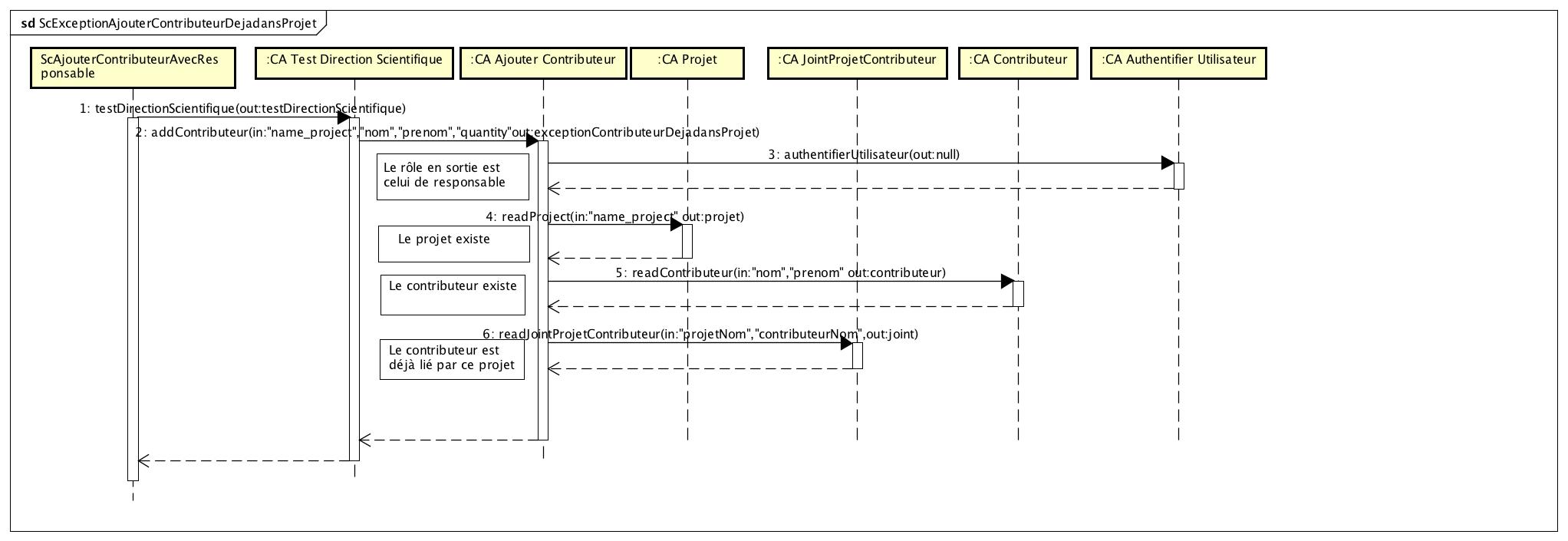


Figure 52 Diagramme de séquences applicatives du scénario ScExceptionAjouterContributeurDejadansProjet

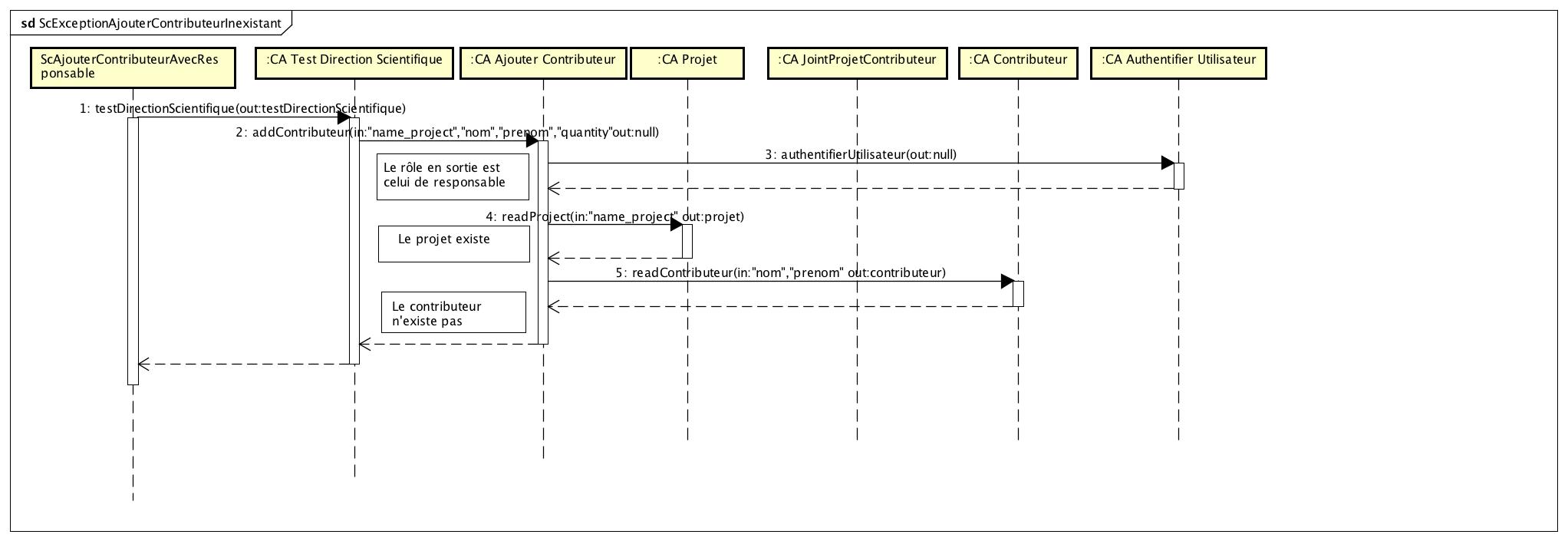


Figure 53 Diagramme de séquences applicatives du scénario ScExceptionAjouterContributeurInexistant

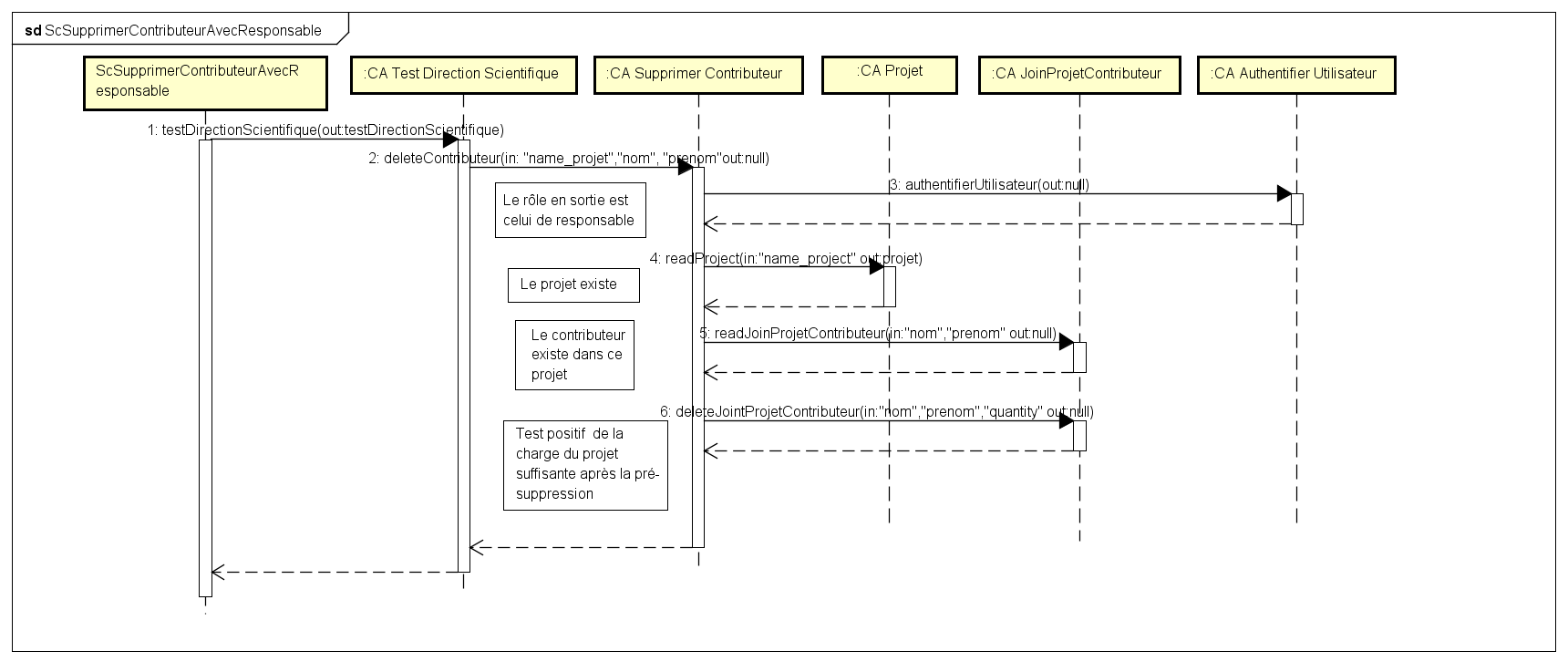


Figure 54 Diagramme de séquences applicatives du scénario ScSupprimerContributeurAvecResponsable

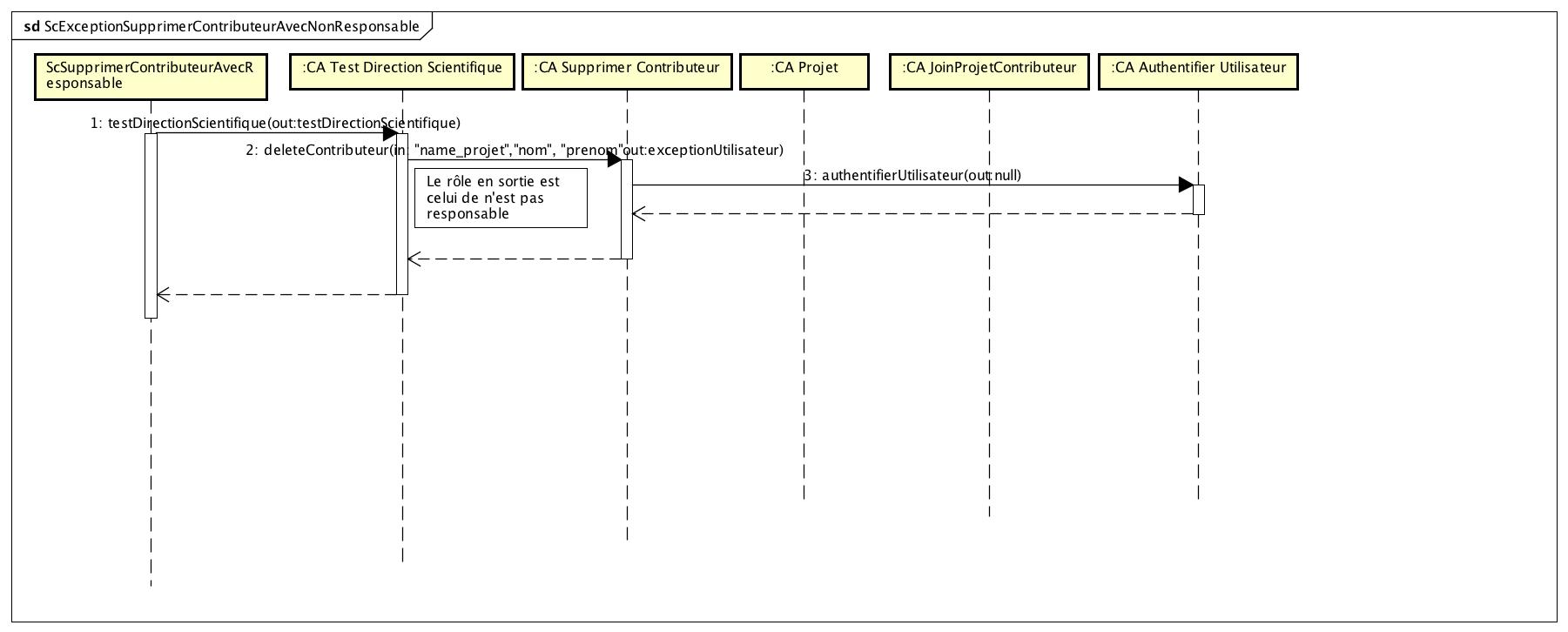


Figure 55 Diagramme de séquences applicatives du scénario ScExceptionSupprimerContributeurAvecNonResponsable

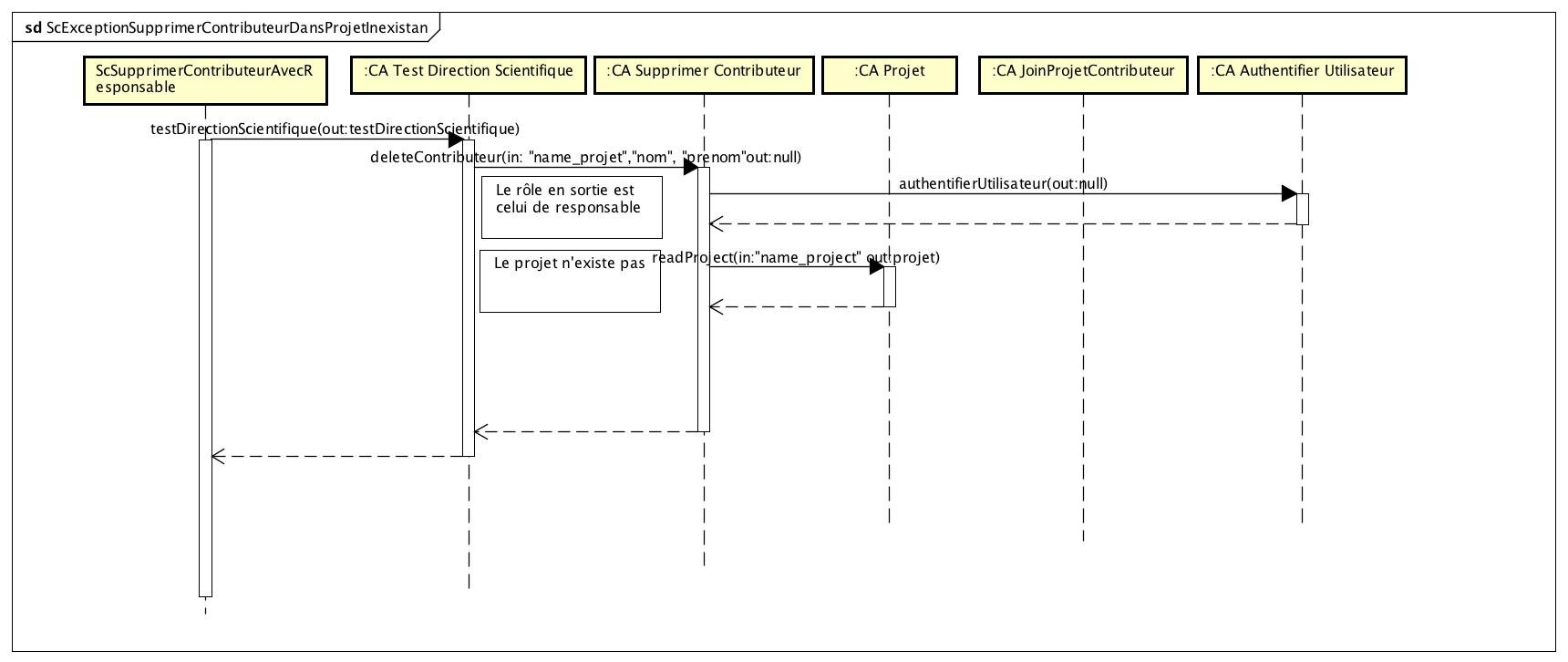
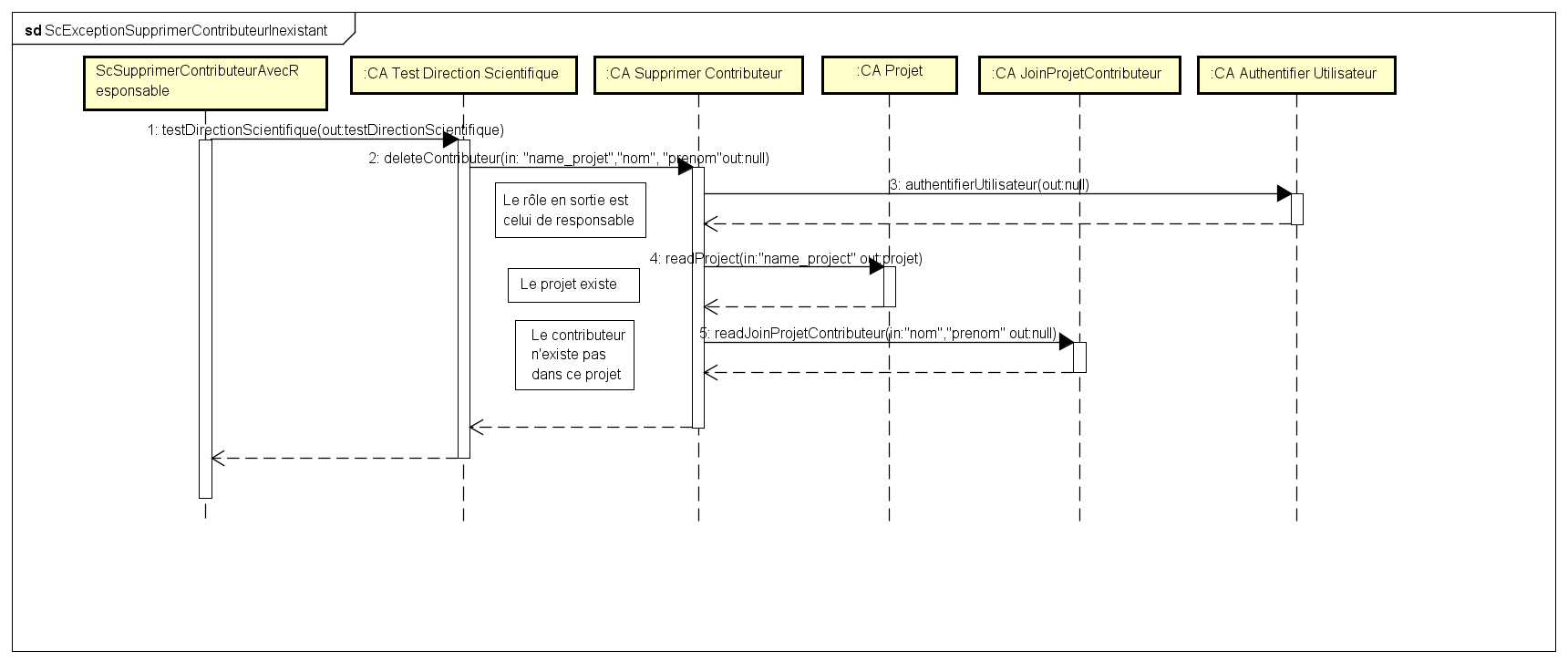


Figure 56 Diagramme de séquences applicatives du scénario ScExceptionSupprimerContributeurDansProjetInexistant

Figure 57 ScExceptionSupprimerContributeurInexistant

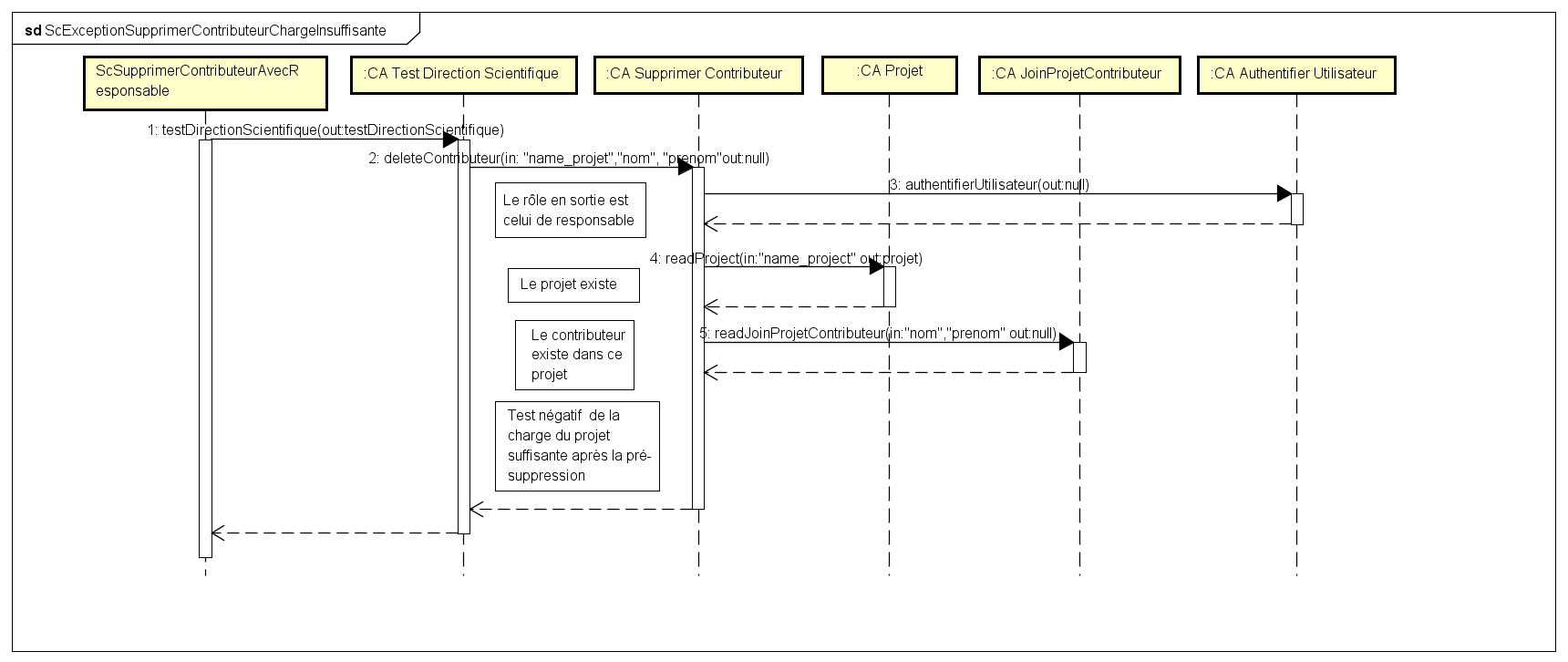


Figure 58 Diagramme de séquences applicatives du scénario ScExceptionSupprimerContributeurChargeInsuffisant

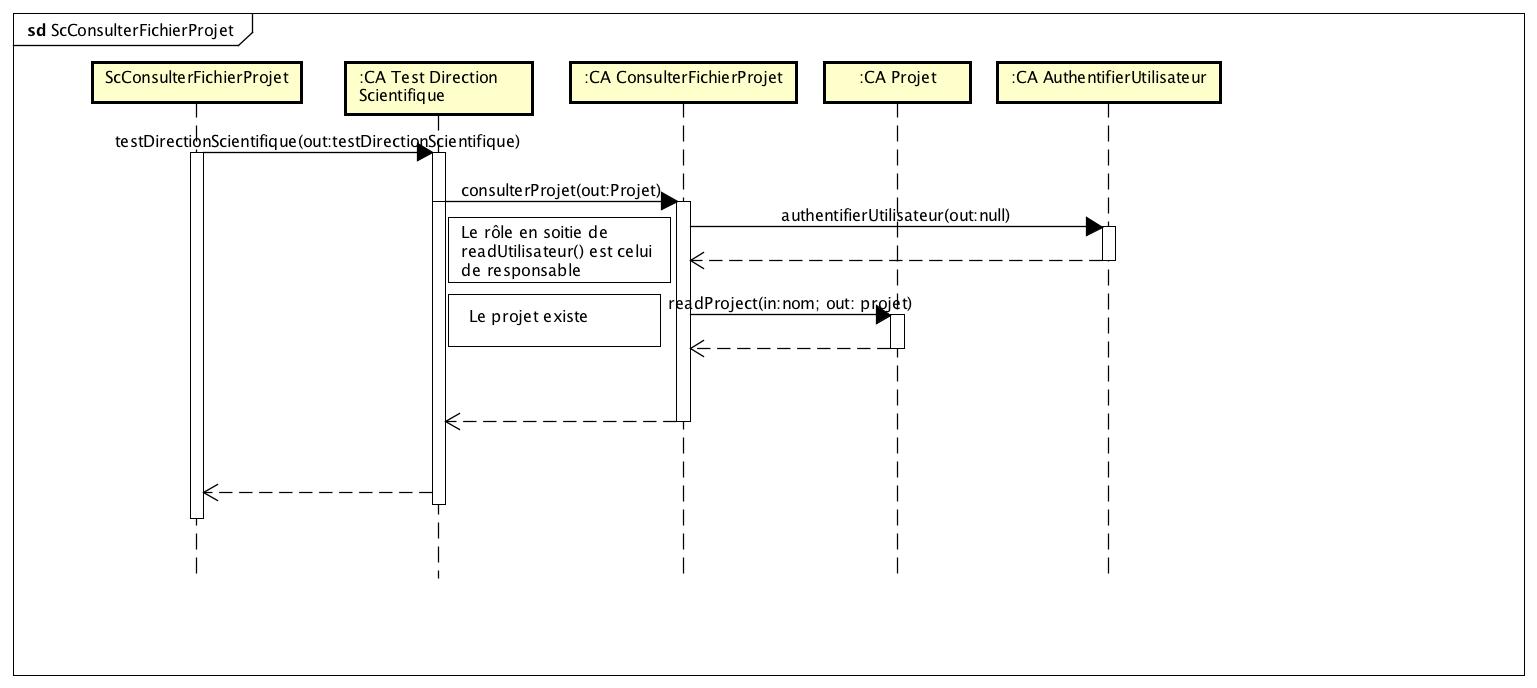


Figure 59 Diagramme de séquences applicatives du scénario ScConsulterfichierProjet

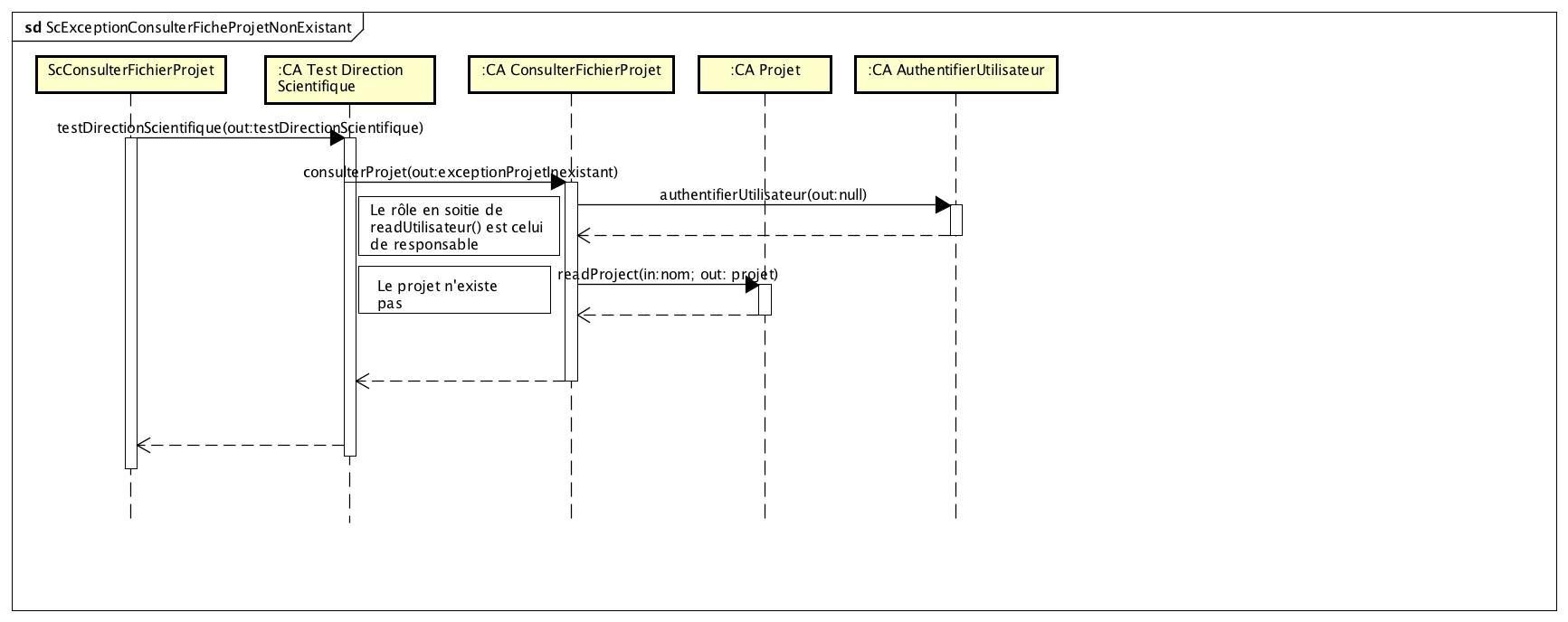


Figure 60 Diagramme de séquences applicatives du scénario ScConsulterfichierProjetNonExistant

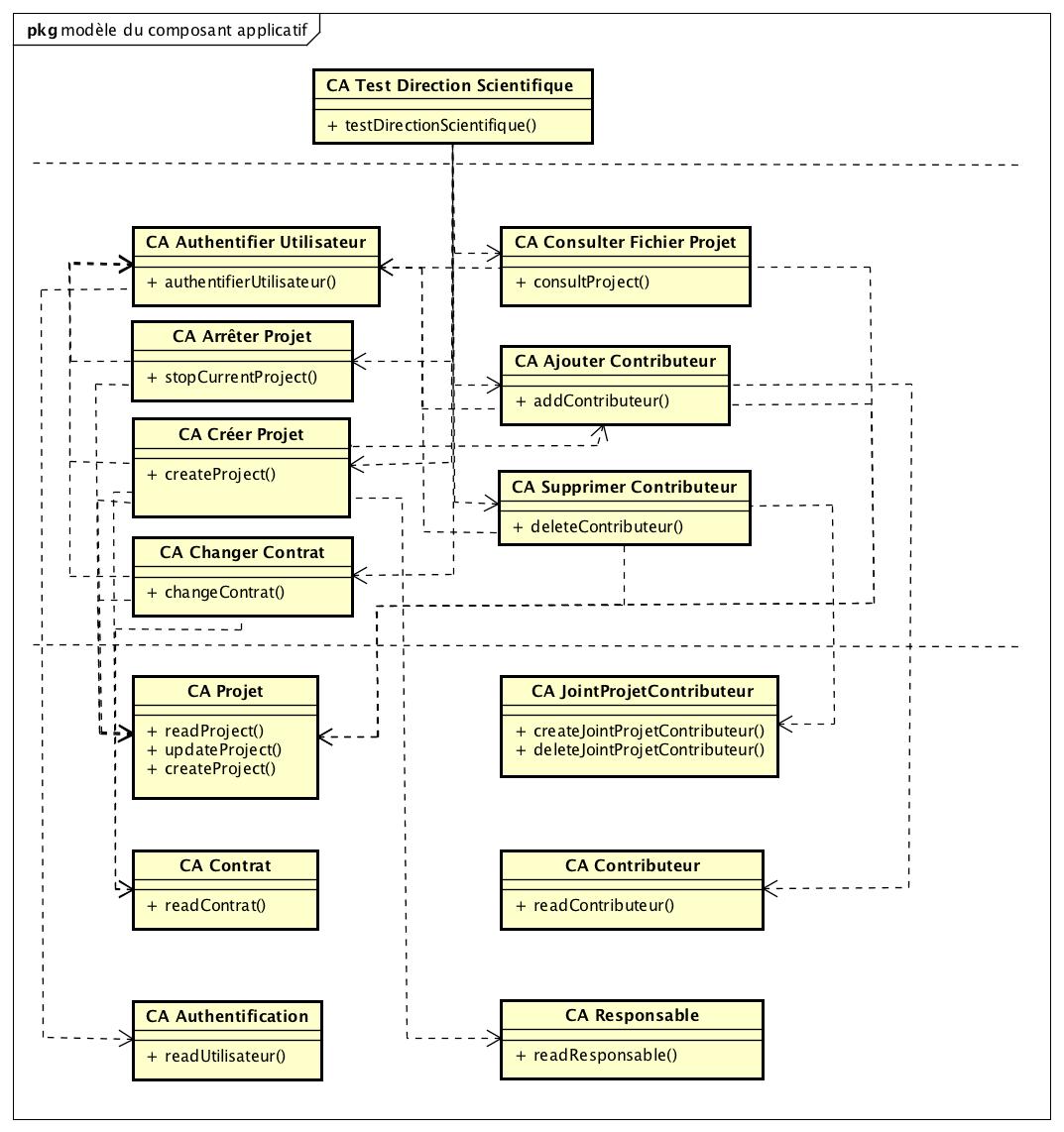
Le modèle des composants applicatifs est représenté par un diagramme de classes UML stéréotypées “composant applicatifs - ‘couche’” dans la Figure 61. Les dépendances entre composants sont déduites des diagrammes de séquence applicatives.

Figure 61 Modèle des composants applicatifs

## Données applicatives

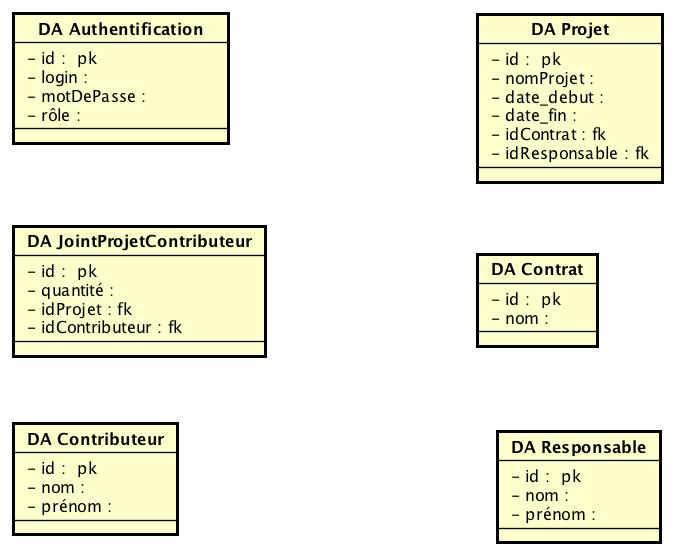
Le modèle des données applicatives est représenté par un diagramme de classes UML stéréotypées “donnée applicative” dans la figure 60. Chaque donnée et chaque dépendance entre données sont cohérentes avec le modèle des composants applicatifs.

Figure 62 Modèle des données applicatives

Le Tableau 11 représente le lien entre les composants applicatifs d’accès aux données et les données applicatives. Un composant applicatif peut fournir une donnée applicative (F) ou utiliser une donnée applicative au sens objet (U). La règle d’urbanisme étant qu’une donnée applicative ne peut être fournie que par un seul composant applicatif d’accès aux données.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Donnée applicative**  **Composant applicatif** | DA Authentification | DA Projet | DA Contributeur | DA Responsable | DA Contrat | DA JointProjetContribu-teur |
| CA Authentification | F |  |  |  |  |  |
| CA Projet |  | F |  | U | U |  |
| CA Contributeur |  |  | F |  |  |  |
| CA Responsable |  |  |  | F |  |  |
| CA Contrat |  |  |  |  | F |  |
| CA JointProjetContributeur |  | U | U |  |  | F |

Tableau 11

La traçabilité des données fonctionnelles est représentée dans le Tableau 12.

|  |  |  |
| --- | --- | --- |
| **Donnée fonctionnelle** | **Attribut** | **Donnée applicative** |
| DF Authentification | nom | DA Authentification |
| prénom |
| rôle |
| DF Projet | nom | DA Projet |
| date\_début |
| date\_fin |
| DF Contributeur | nom | DA Contributeur |
| prénom |
| DF Responsable | nom | DA Responsable |
| prénom |
| DF Contrat | nom | DA Contrat |
| DF Charge | quantity | DA JointProjetContributeur |

Tableau 12

Des dépendances des données fonctionnelles réalisant des associations entre entités participantes ayant pour cardinalités \*-\* dans le Tableau 13 :

|  |  |
| --- | --- |
| Dépendance entre données fonctionnelles | Donnée applicative |
| DF Projet vers DF Contributeur | DA JointProjetContributeur |

Tableau 13

|  |  |
| --- | --- |
|  | Technopôle Brest-Iroise CS 83818 29238 Brest Cedex 3 France  +33 (0)2 29 00 11 11  **www.telecom-bretagne.eu** |
| INSTITUT-MINES-TELECOM_Logo_Buro_petit | |

|  |  |
| --- | --- |
|  |  |
|  | |