



---

# **Modélisation objet**

**Réalisation d'un système Gestion des suivis  
d'étudiants**

**(Développement de la plateforme InteractionMIAGE)**

**Travail réalisé par : *KECHIR Amine et YEFSAH Rayane***

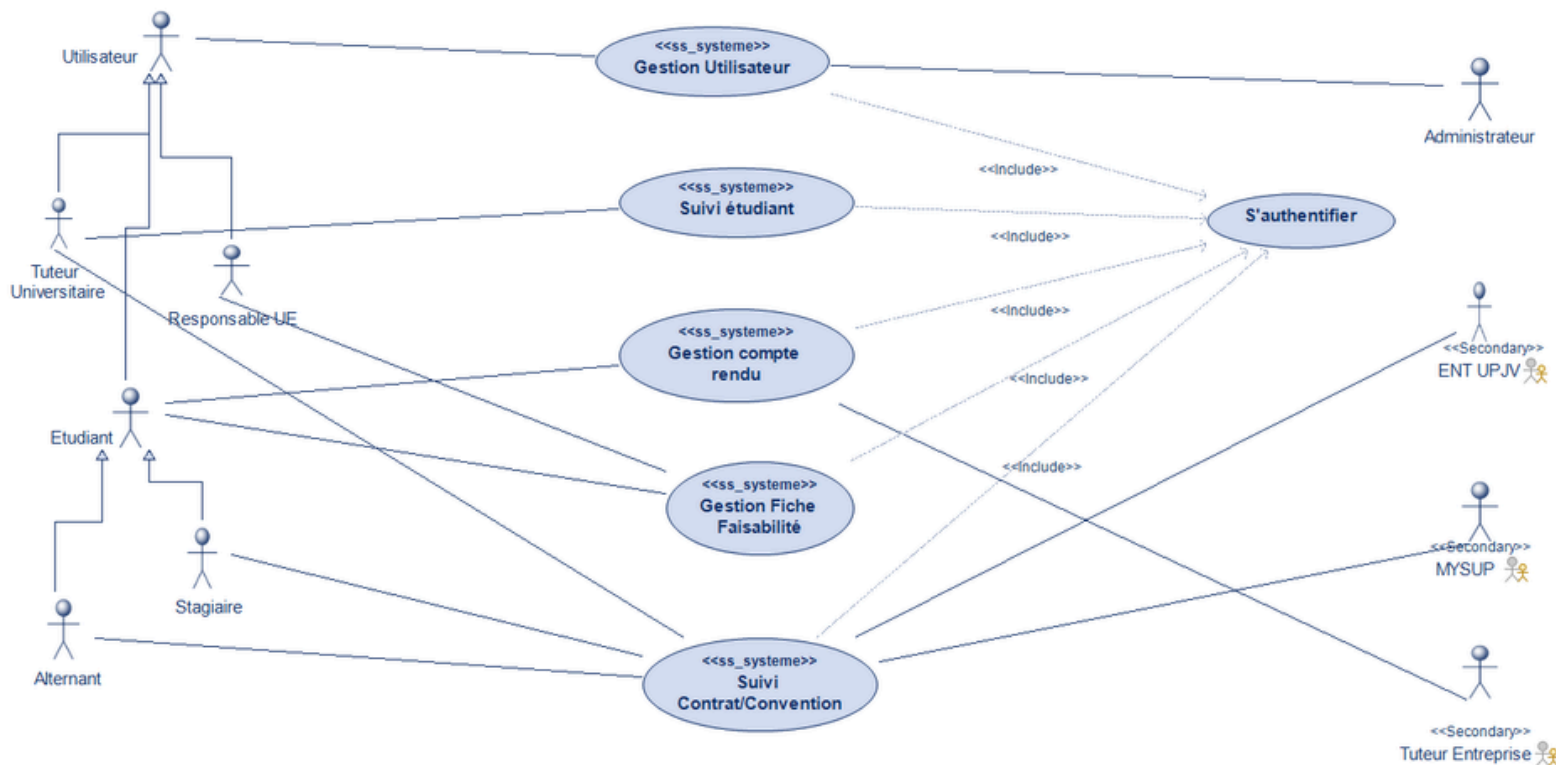
## Introduction :

Dans le cadre de ce projet, nous avons entrepris le développement d'un système visant à optimiser la gestion des étudiants, des tuteurs, et des responsables pédagogiques dans le cadre de stages académiques ou des stage professionnel. Le projet est conçu pour répondre aux besoins des établissements d'enseignement supérieur qui cherchent à structurer et à automatiser les processus de suivi et de validation des activités liées aux stages, tout en garantissant une communication efficace entre les différents acteurs impliqués.

## Identification des sous-systèmes :

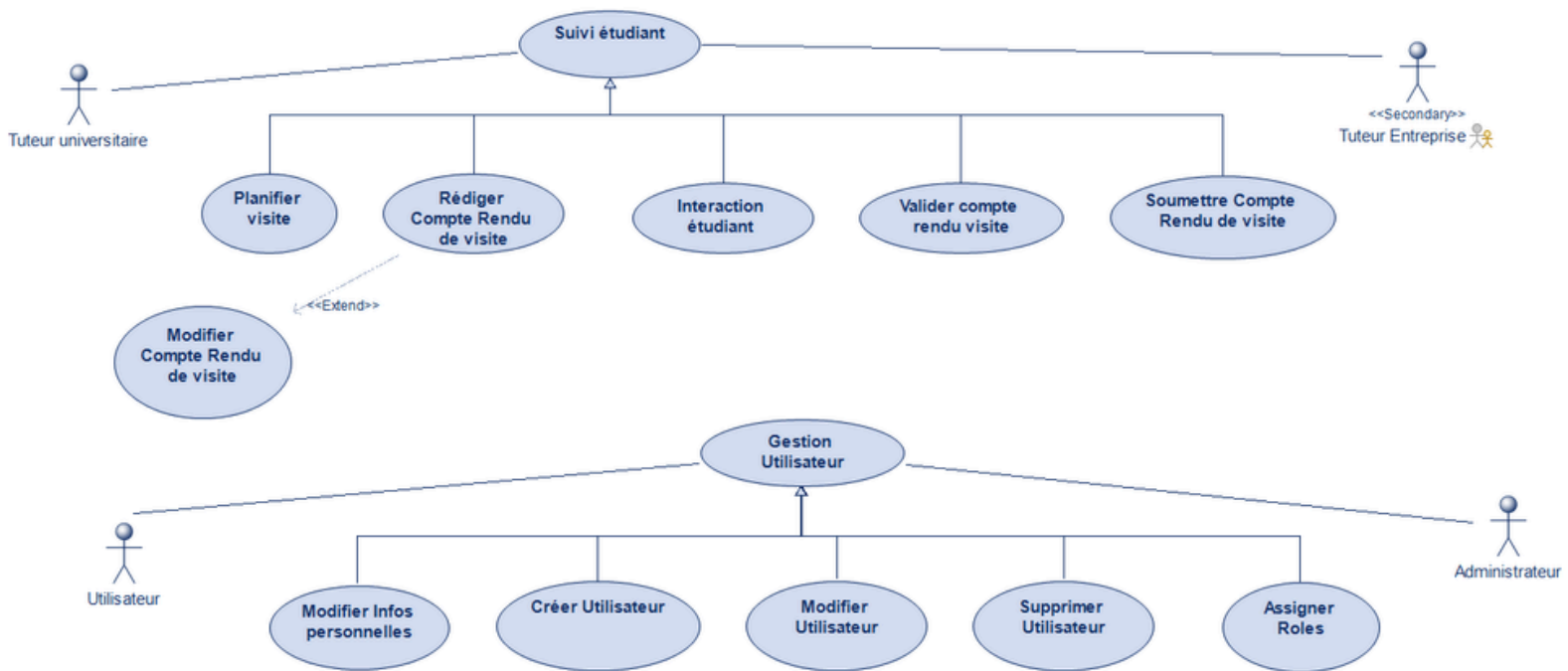
Après l'étude et l'analyse de l'énoncé de projet nous avons pu décomposer notre système en plusieurs sous-systèmes et on les a représentés en diagrammes de cas d'utilisation.

Voici le diagramme de cas d'utilisation général qui représente l'ensemble des sous-systèmes



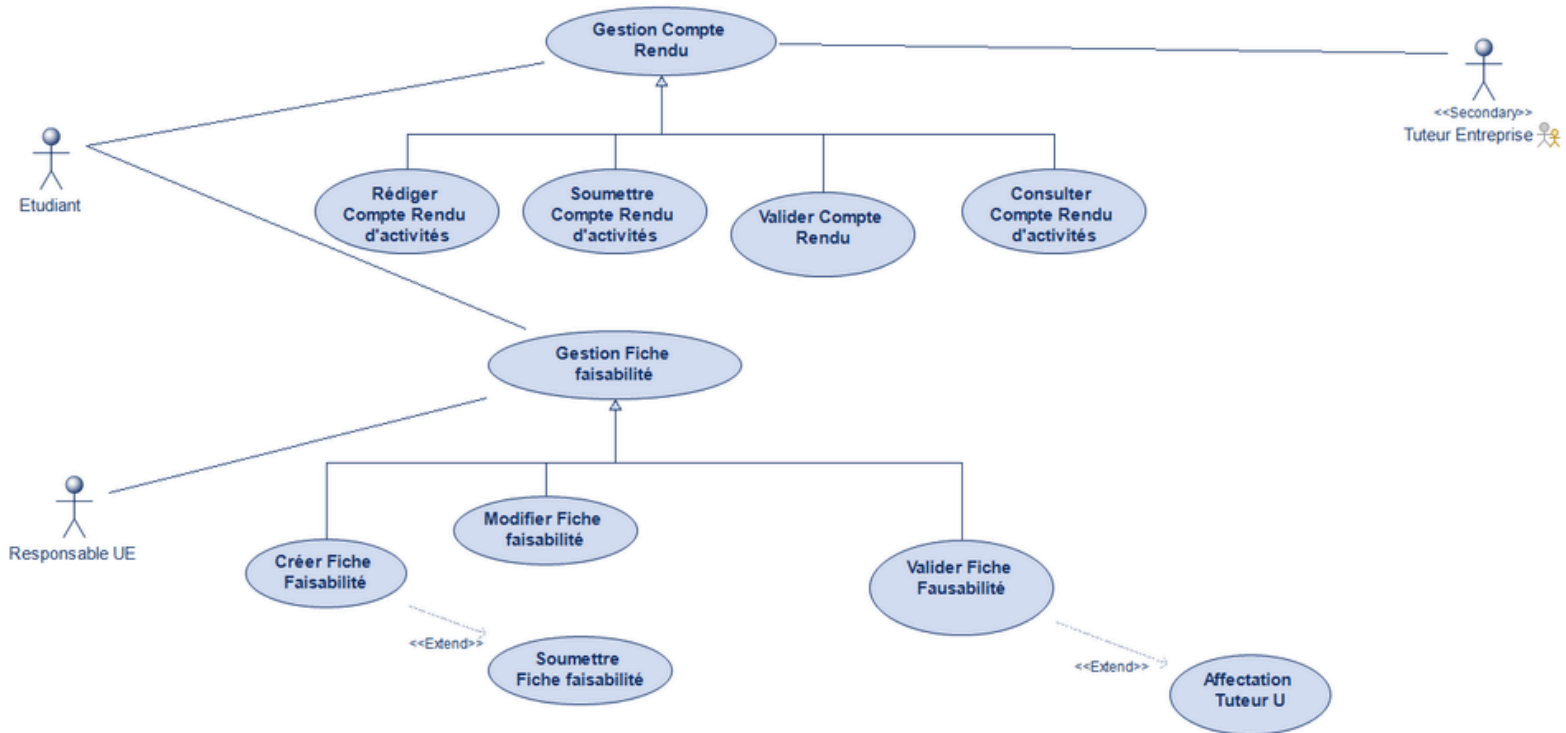
Et voici les diagrammes de cas d'utilisation détaillé de chaque sous-système :

- **Suivi des étudiant :** Ce sous-système permet aux acteurs pédagogiques, comme le tuteur universitaire et le tuteur entreprise, d'assurer un suivi efficace des étudiants en entreprise. Il inclut des fonctionnalités telles que la planification de visites, la rédaction et validation de comptes rendus, l'interaction avec les étudiants, et la soumission de rapports. Il garantit une traçabilité et une collaboration entre les différents intervenants..



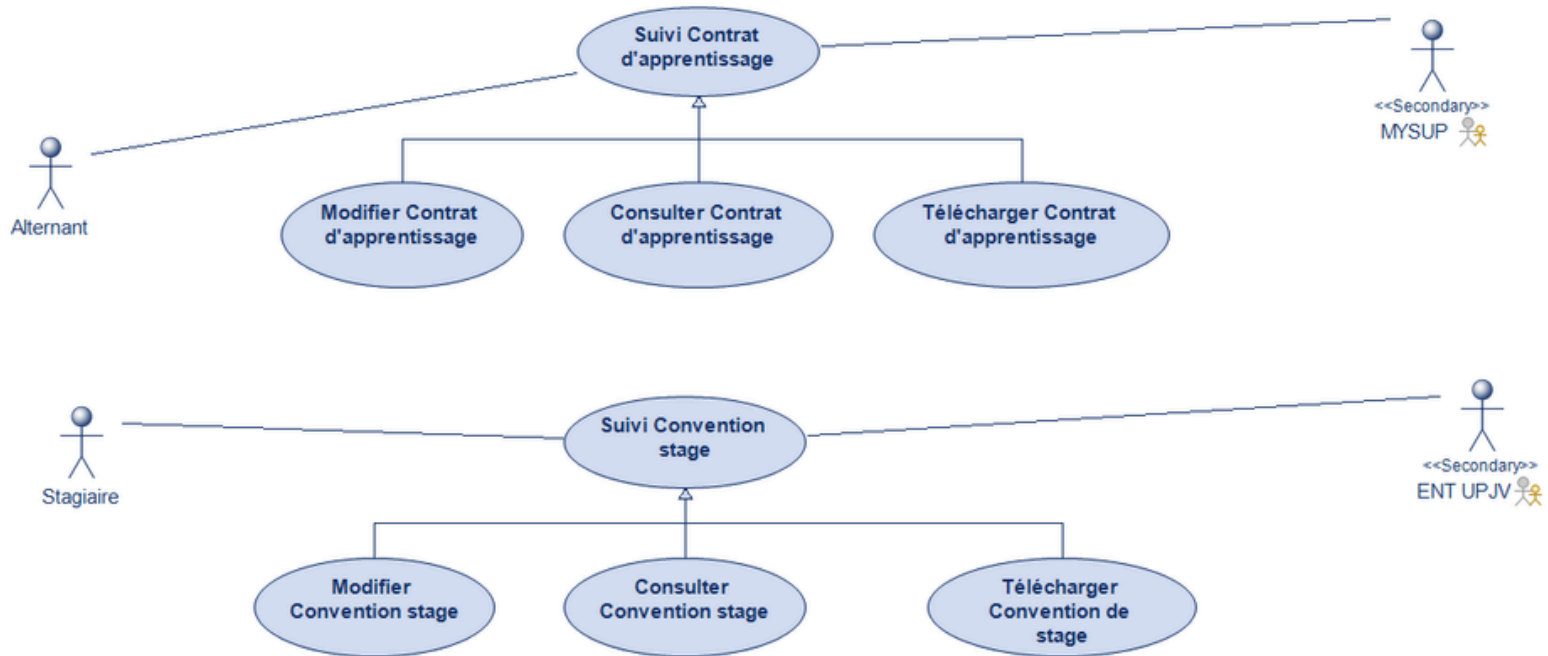
- **Gestion utilisateur:** Ce sous-système se concentre sur l'administration des utilisateurs de la plateforme. Il permet à l'administrateur de créer, modifier, supprimer des utilisateurs et d'assigner des rôles, tandis que les utilisateurs peuvent mettre à jour leurs informations personnelles. Il assure ainsi une gestion sécurisée et organisée des accès et droits sur le système.

- **Gestion Compte rendu:** Ce sous-système permet à l'étudiant de rédiger et soumettre des comptes rendus d'activités, lesquels peuvent être consultés et validés par le tuteur universitaire ou le tuteur entreprise. Il assure un suivi précis et structuré des activités réalisées en entreprise, facilitant la communication entre les différents acteurs.



- **Gestion Fiche Faisabilité:** Ce sous-système est utilisé par le responsable d'unité d'enseignement (UE) et l'étudiant pour créer, modifier et soumettre des fiches de faisabilité. Ces fiches doivent être validées par le responsable de l'UE avant l'affectation du tuteur universitaire. Ce processus garantit que les missions proposées par l'entreprise respectent les exigences académiques et pédagogiques.

- • **Suivi Contrat d'apprentissage:** Ce sous-système permet la gestion complète et efficace du contrat d'apprentissage. Il permet à l'alternant de gérer de manière autonome son document contractuel tout au long de sa formation, tout en offrant à l'alternant la possibilité de modifier, consulter et télécharger son contrat sous la supervision de MYSUP.

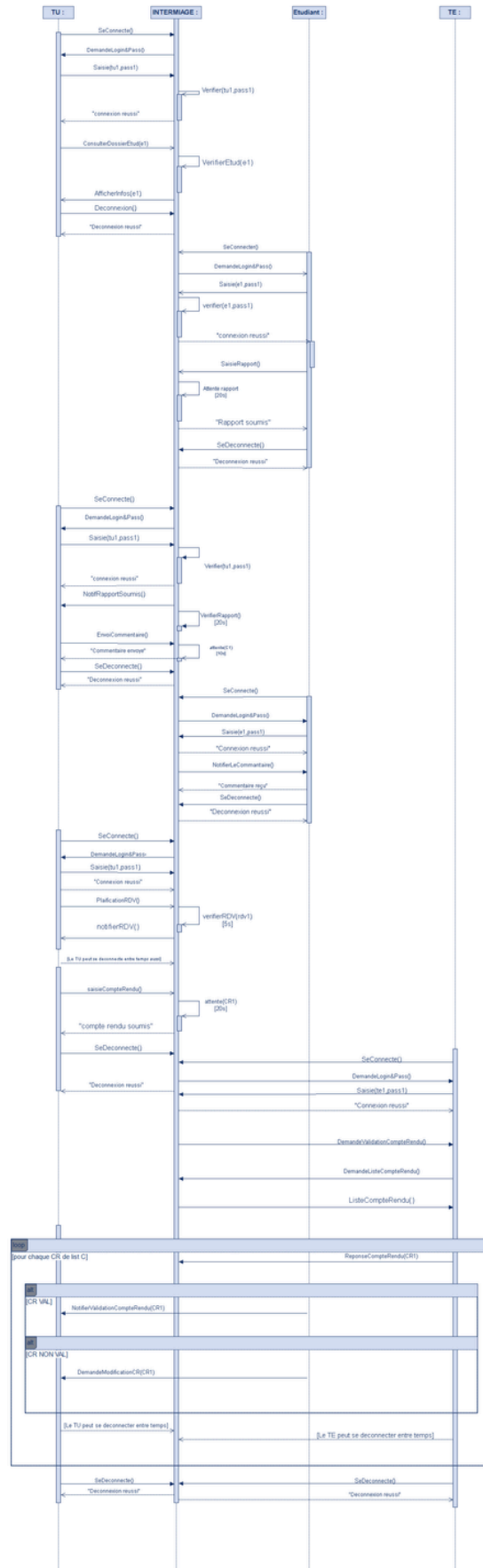


- **Suivi Convention stage:** Ce sous-système est conçu pour faciliter la gestion complète des conventions de stage. Il permet au stagiaire de gérer son document de manière autonome tout au long de sa période de stage, sous la supervision de l'ENT UPJV.

(Les deux sous-systèmes ont une structure similaire, centrée sur le type particulier de contrat apprentissage ou stage. En outre, ils ont des fonctionnalités similaires adoptées au contexte, ce qui signifie qu'un seul approché est pris en compte pour assurer la gestion correcte des deux formes de pratique. Comme conséquence, la documentation administrée est gérée de manière standardisée et efficace car les pré-requis font partie l'un ou l'autre du contrat que signent un étudiant alternant ou un stagiaire. En outre, les outils de supervision officielle MYSUP et ENT UPJV sont inclus dans cette structure à chaque étape.)

# Représentation des fonctionnalités en diagramme d'interaction :

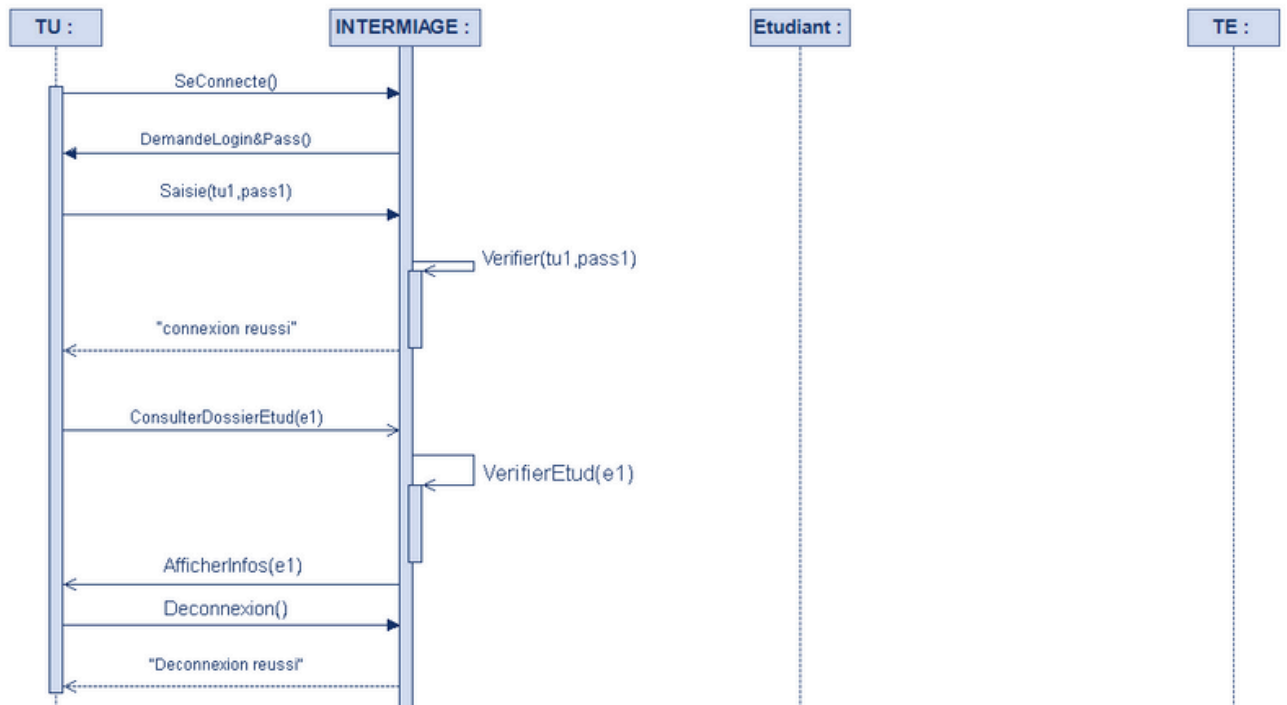
Nous avons élaboré un diagramme de séquence afin de clarifier et expliquer les interactions entre les différentes parties du notre système pour réaliser un certain traitement. Cela nous a facilité la transition vers la tâche de codage. Nous l'avons ensuite décomposé pour mieux expliquer les interactions entre les parties concernées.



## Décomposition des fonctionnalités en diagramme d'interaction :

**Connexion tuteur universitaire:** Ce diagramme illustre un scénario où un tuteur universitaire se connecte au système intermédiaire (INTERMIAGE), consulte les informations d'un étudiant, puis se déconnecte. Le diagramme montre les interactions principales entre le TU, le système INTERMIAGE et les vérifications effectuées. Il comprend notamment les opérations suivantes :

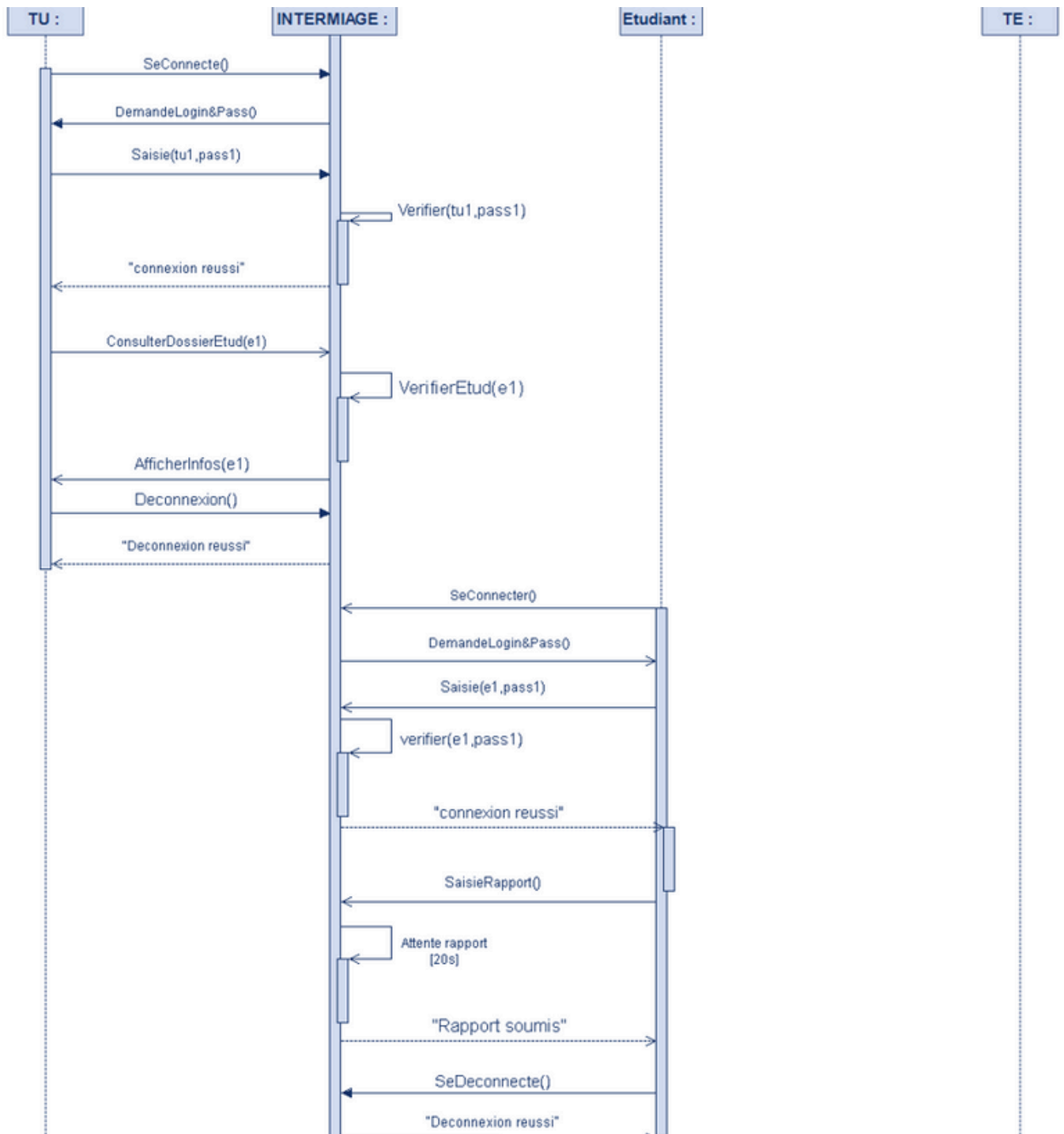
- Connexion
- Consultation des Informations Étudiant
- Déconnexion



## Décomposition des fonctionnalités en diagramme d'interaction :

**Connexion et Soumission d'un Rapport (Étudiant):** Ce diagramme illustre un scénario dans lequel un étudiant interagit avec le système intermédiaire (INTERMIAGE) pour soumettre un rapport. Il met en évidence les principales étapes impliquées dans le processus, de la connexion au système jusqu'à la déconnexion, en passant par la soumission du rapport. Les interactions clés entre l'étudiant et le système sont représentées, incluant des délais pour le traitement du rapport. Ce processus comprend trois phases principales :

- Connexion
- Soumission de Rapport
- Déconnexion

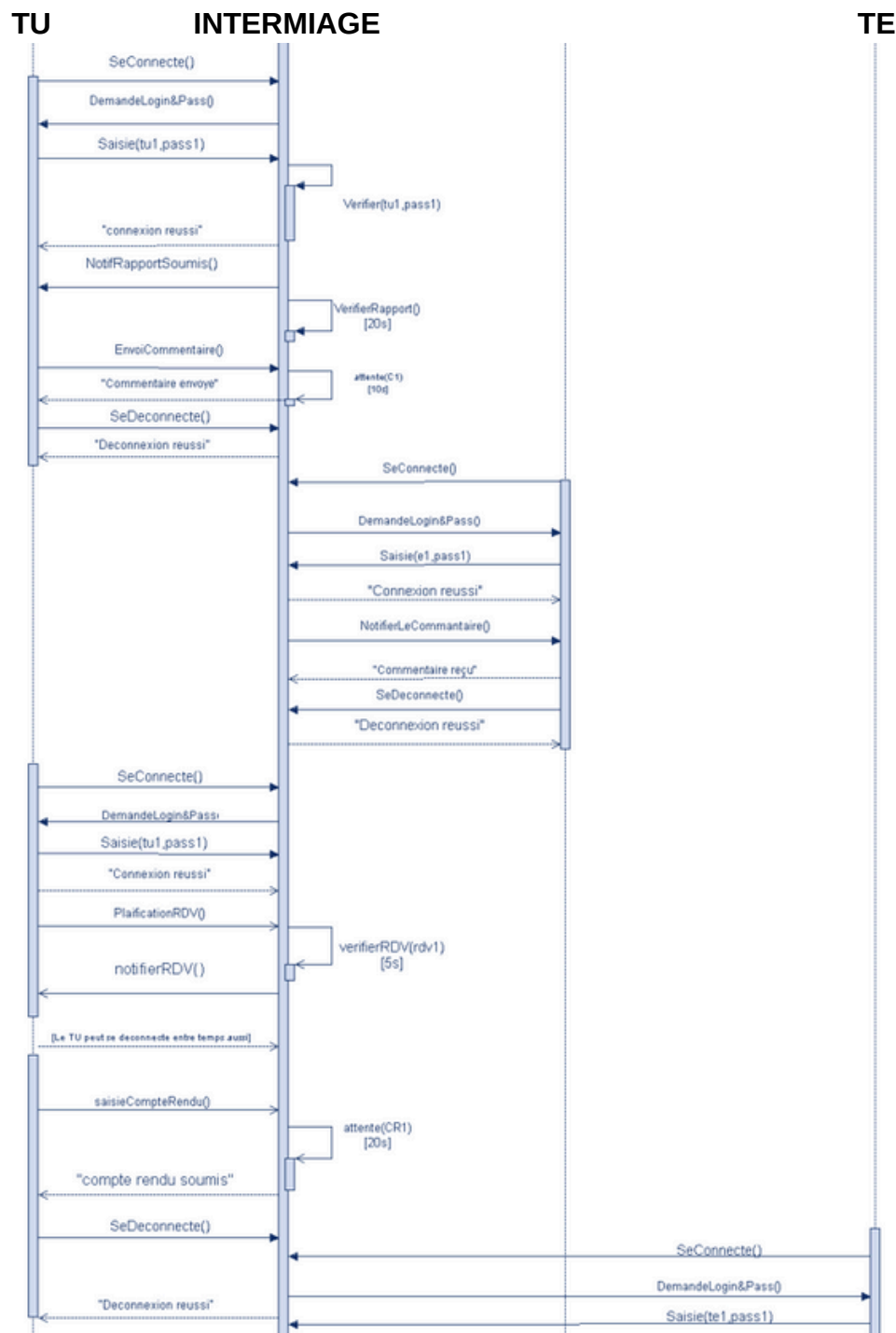




## Décomposition des fonctionnalités en diagramme d'interaction :

**Connexion, Planification et Validation des Comptes Rendus de visites (Tuteur universitaire):** Ce diagramme illustre les interactions entre le Tuteur Universitaire (TU) et le système intermédiaire (INTERMAGE) pour la gestion des comptes rendus de visites et la planification des rendez-vous. Il met en avant les principales étapes suivantes :

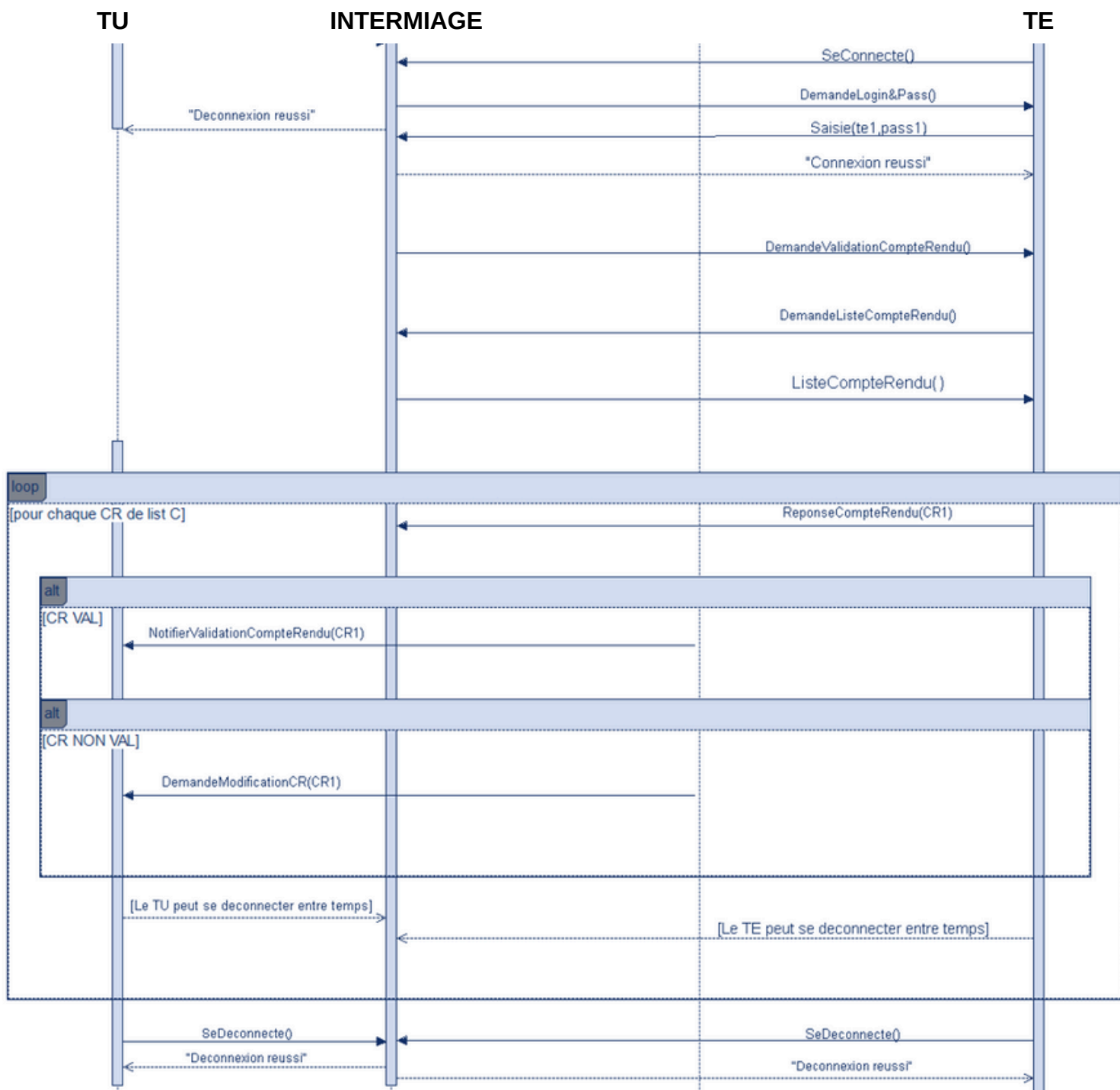
- **Connexion au système :** Le TU saisit ses identifiants, qui sont vérifiés pour un accès sécurisé.
- **Gestion des rendez-vous :** Le TU planifie des rendez-vous avec les étudiants et reçoit des notifications de confirmation.
- **Gestion des comptes rendus de visites :** Le TU reçoit les comptes rendus soumis, les valide ou ajoute des commentaires pour modification.
- **Déconnexion**



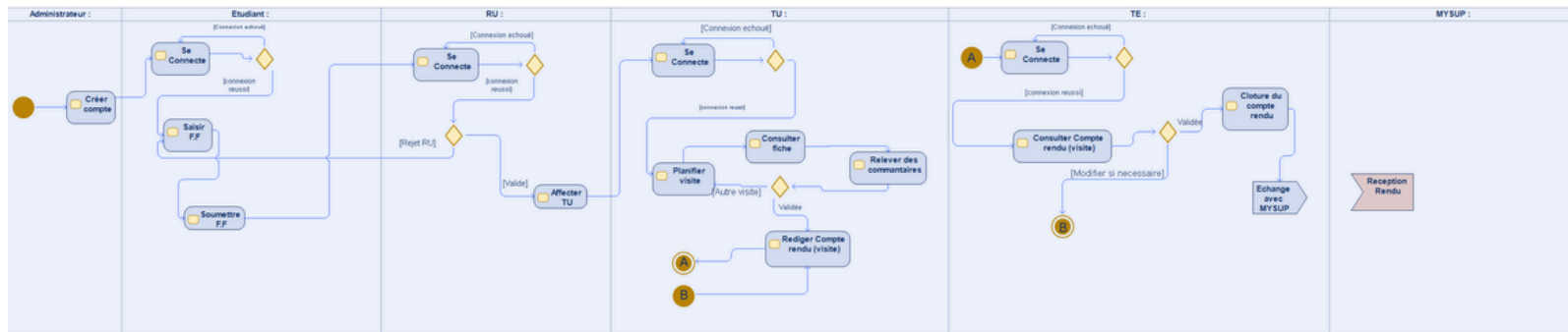
## Décomposition des fonctionnalités en diagramme d'interaction :

**Connexion et Validation des Comptes Rendus (Tuteur Entreprise):** Ce diagramme décrit le processus dans lequel un tuteur d'entreprise (TE) interagit avec le système intermédiaire (INTERMAGE) pour valider ou demander la modification de comptes rendus soumis. Le diagramme met en évidence les interactions structurées entre le TE et le système, en prenant en compte les principales étapes impliquées dans le processus, de la connexion au système jusqu'à la déconnexion, en passant par la validation des comptes rendus. en intégrant des mécanismes de boucle (loop) et d'alternatives (alt) pour traiter plusieurs comptes rendus. Le processus se divise en trois phases principales :

- connexion,
- validation des comptes rendus
- déconnexion.



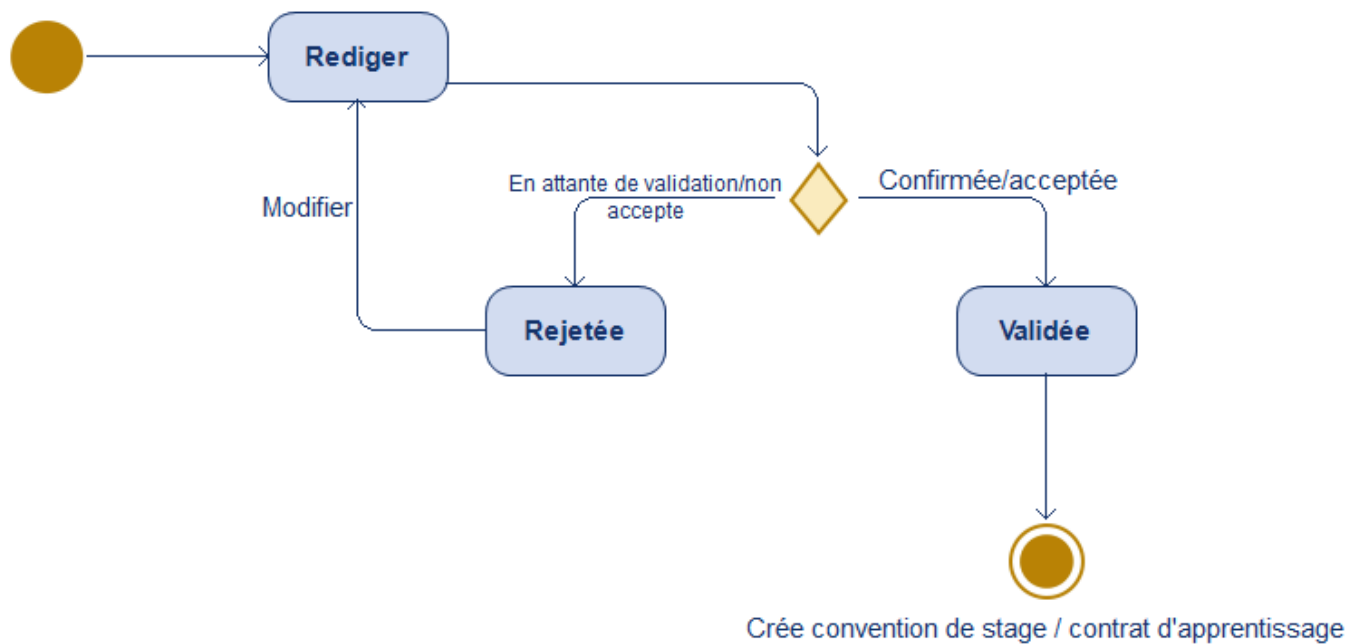
- **Interactions entre l'étudiant et tuteurs(universitaire et entreprise)** : Pour les interactions entre ses différents utilisateurs on l'a modélisé par un diagramme d'activité. Celui-ci permet de suivre de manière séquentielle les différentes étapes allant de la connexions passant par les actions associés à chaque utilisateurs jusqu'à la déconnexion



### Flux du processus :

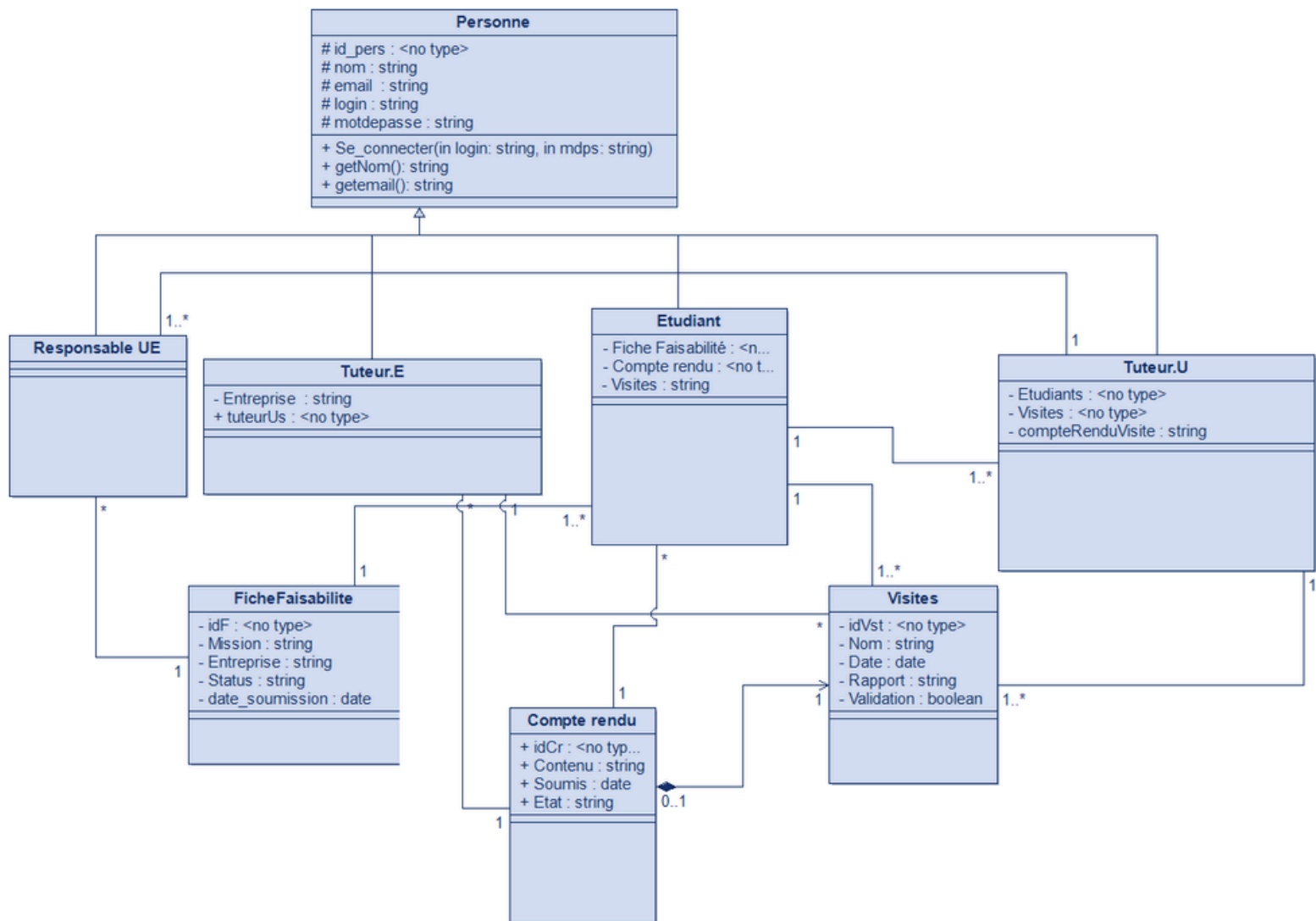
Administrateur	Etudiant:	Le RU (Responsable UE)	Le TU (Tuteur Universitaire) :	Le TE (Tuteur Entreprise) :
<ul style="list-style-type: none"> <li>Crée des comptes utilisateurs</li> </ul>	Se connecte Saisie et soumis une fiches de faisabilités	: <ul style="list-style-type: none"> <li>Se connecte</li> <li>Valide ou refuse les fiches de faisabilité</li> <li>Attribue les TU</li> </ul>	<ul style="list-style-type: none"> <li>Se connecte</li> <li>Consulte les fiches</li> <li>Planifie les visites</li> <li>Rédige les comptes rendus de visite</li> <li>Relève les commentaires</li> </ul>	<ul style="list-style-type: none"> <li>Se connecte</li> <li>Consulte les comptes rendus de visite</li> <li>Valide les compte rendus de visite</li> <li>Peut modifier si nécessaire</li> </ul>

- **gestion des fiches de faisabilité** : La modélisation du pour la “gestion des fiches de faisabilité” sur la plateforme à l'aide d'un diagramme d'état permet de décrire les différents états qui représente un processus de rédaction de la fiche de faisabilité et de validation ou de refus d'une fiche de faisabilité par un responsable UE afin de créer une convention de stage ou d'un contrat d'apprentissage.

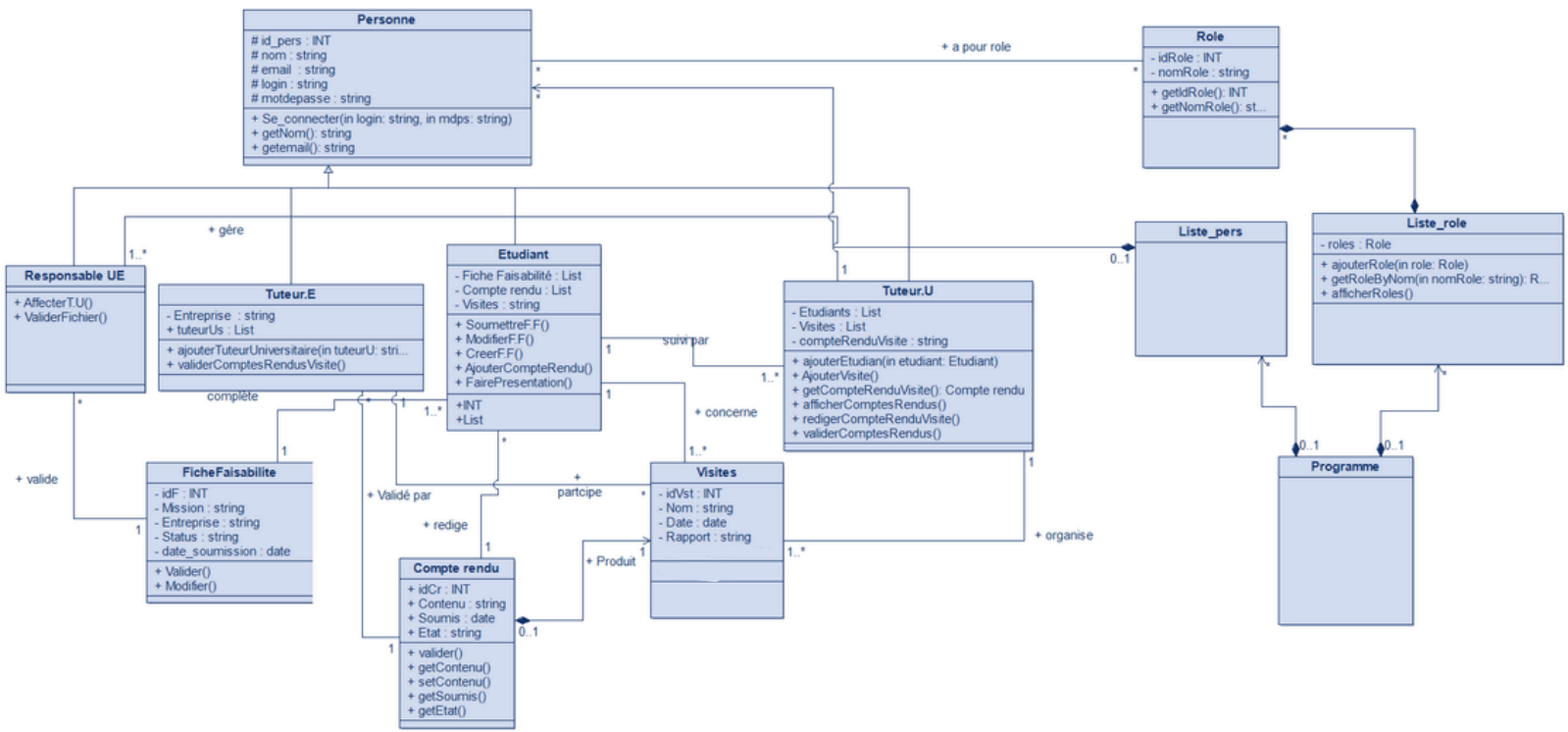


# La représentation de système par un diagramme de classe :

**Diagramme de classe général :** Ce schéma illustre simplement les composants clés de notre système, décrivant leurs caractéristiques et les relations entre eux. Cette vue d'ensemble nous a aidés à saisir la structure générale avant de passer à la phase de développement.



**Diagramme de classe détaillé :** Dans ce schéma, nous avons enrichi la représentation des entités de notre système en y intégrant des méthodes spécifiques et en ajoutant quelques classes pour le bon fonctionnement du programme. Ce niveau de détail supplémentaire a été introduit pour guider plus précisément le processus de développement, en définissant les opérations spécifiques et les interactions nécessaires au bon fonctionnement de notre application.



# Conclusion:

Notre projet n'a pas été exempt de défis, et nous avons dû faire face à plusieurs obstacles au cours de son développement. Cependant, chaque difficulté rencontrée a été une occasion d'apprentissage et d'amélioration. Nous avons pris le temps d'explorer en profondeur les concepts de modélisation orientée objet et de perfectionner notre maîtrise des outils de conception UML, tels que les diagrammes d'état, d'activités et de classes.

Durant la phase de développement, nous avons dû surmonter des complexités liées à la structuration des interactions entre les tuteurs et les étudiants, ainsi qu'à l'intégration des fonctionnalités spécifiques à l'UE « Projet Professionnel ». En Java, le défi a été de traduire avec précision nos diagrammes en code fonctionnel tout en respectant les principes de modularité et de réutilisabilité.

Malgré ces défis, nous avons réussi à créer une base solide pour la plateforme InteractionMIAGE. Ce programme gère efficacement les interactions entre les tuteurs et les étudiants tout en s'intégrant avec les outils existants, répondant ainsi aux besoins identifiés. Ce projet nous a permis de développer nos compétences techniques et méthodologiques, tout en nous préparant à relever des défis similaires dans des contextes professionnels.