

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université A/Mira de Béjaia



Faculté des Sciences Exactes

Département d'Informatique

Mémoire de fin d'études

Présenté pour obtenir le diplôme de Master en Informatique

Option : Génie logiciel

Thème

*Conception et réalisation d'une application web
pour le SI de gestion du Conseil de Discipline du
Département Informatique.*

Mémoire soutenu par:

Mr: Mersel Azouaou

Soutenu devant le jury composé de :

Examineur : Mr. MOKTEFI Mohand

MAA

U.A.M B

Examineur : Mr. kacimi

MAA

U.A.M B

Encadreur : Mr. Abderrahmane Sider

MAA

U.A.M B

Année Universitaire 2020/2021

Remerciement

En premier, je tiens à remercier le bon Dieu le tout puissant de m'avoir donné la santé, le courage, la volonté et la patience pour mener à terme ce modeste travail.

Je tiens à remercier mes chers parents qui m'ont soutenu et encouragé durant toute ma vie et pendant mon cursus d'études.

Mes remerciements les plus chaleureux vont à monsieur Sider Abderrahmane pour sa disponibilité et ces très précieux conseils ainsi que ces remarques qui m'ont permis d'améliorer la qualité de ce travail.

Je tiens à exprimer toute ma grande gratitude aux membres du jury d'avoir accepté de juger ce travail.

Mes vifs remerciements s'adressent également à tous mes enseignants de la faculté des sciences exactes de l'université A/MIRA Béjaia pour la formation qu'ils ont eu le soin de m'apporter le long de mon cursus universitaire.

Dédicace

*C'est avec un énorme plaisir, un cœur ouvert et une immense joie que je
dédie mon travail*

*À mes chers parents qui m'ont soutenu et prodigué de précieux conseils
tout au long de ma vie.*

À mon cher frère Massinissa et à mon adorable sœur Ines.

*À mon grand père paternel Djoudi et ma grand mère maternelle
Louina qui ont partagé avec moi tout les moments d'émotion lors de la
réalisation de ce travail, que dieu puisse leur accordé longue vie.*

*À mon encadreur monsieur Sider Adberahmane pour son soutien, ses
orientations et ses encouragements.*

*À mes enseignants dont j'ai eu l'honneur et la chance de bénéficier de
leur connaissances et leur compétences, ainsi que leur précieux conseils.*

À toute ma famille, mes proches.

À mes amis et mes camarades.

À tout ceux qui me donnent de l'amour et de la vivacité.

Azouaou Mersel

Table de matière

Table de matière	I
Liste Des Abréviations	IV
Liste des Figures	V
Liste des tableaux	V
Introduction générale	1
Généralités.....	2
I.2 Généralité sur les système d'information(SI)	2
I.2.1 Définition	2
I.2.2 Représentation des Systèmes d'informations	2
I.2.3 Les fonctions d'un système d'information.....	3
I.3 UML (Unified Modeling Language).....	3
I.3.1 Définition	3
I.3.2 Historique de l'UML	3
I.3.3 Les diagrammes de modélisation UML	4
Conclusion.....	6
Analyse et Spécification des besoins.....	7
II.1 Introduction.....	7
II.2 Spécification des besoins	7
II.2.1 Présentation de l'organisme d'accueil.....	7
II.2.1.1 Les formations offertes par le département.....	7
II.2.1.2 Personnel d'enseignement de département informatique	7
II.2.2 Conseils de Discipline Universitaires	8
II.2.2.1 Définition	8
II.2.2.2 Création d'un Conseil de Discipline	8
II.2.2.3 Les membres d'un Conseil de Discipline	8
II.2.2.4 Différents Conseils de Disciplines existant a l'université	8
II.2.2.5 Les type d'infractions	9
II.2.2.6 Sanctions pour infractions présumées	9
II.2.3 La description du contexte du projet	10
II.2.4 Problématique.....	10
II.2.5 Solutions.....	10
II.2.6 Les objectifs de mon application.....	11



II.3	Analyse des besoins.....	11
II.3.1	L'identification des besoins	11
II.3.1.1	Les besoins fonctionnels	11
II.3.1.2	Les besoins non fonctionnels	11
II.3.2	L'identification des acteurs.....	11
II.3.2.1	Les acteurs principaux.....	12
II.3.3	L'identification des cas d'utilisations de l'application	12
II.3.4	Le diagramme des séquences	21
	Conclusion.....	27
	Étude conceptuelle.....	28
III.1	introduction.....	28
III.2	Diagramme d'interaction.....	28
III.3	Diagramme des classes	34
III.3.1	Dictionnaire de données	35
III.4	Le modèle relationnel	35
III.4.1	Règles de passage au modèle relationnel:	35
III.4.2	Modèle relationnel.....	36
	Conclusion.....	36
	Réalisation.....	37
IV.4.	Introduction.....	37
IV.2	Environnement et outils de développement.....	37
IV.2.1	Outils de développement.....	37
IV.2.1.1	Laravel.....	37
IV.2.1.2	Livewire.....	37
IV.2.1.3	Visual Studio code.....	37
IV.2.1.4	PHP.....	37
IV.2.1.5	Tailwind CSS.....	38
IV.2.1.6	Astah UML.....	38
IV.2.1.7	MySQL	38
IV.2.1.8	XAMPP.....	38
IV.3	Ergonomie.....	38
IV.3.1	Logo	38
IV.3.2	Interfaces de l'application	39
IV.3.2.1	Interface d'authentification	39
IV.3.2.2	Page d'Accueil.....	40



IV.3.2.3	Page de gestion des utilisateurs.....	40
IV.3.2.4	Page de gestion des étudiants.....	41
IV.3.2.5	Capture d'écran d'une annonce.....	41
	Conclusion Générale	42
	Bibliographie.....	43

Liste Des Abréviations

UML : Unified Modeling Language.

S.I : Système D'information.

VSC: Visual Studio Code.

PHP: Pre Hypertext Preprocessor.

BDD: Base De données.

SQL: Structured Query Language.

IDE: Environnement de Développement intégré

OMT: Object Modeling Technique

OMG: Object Management Group

SGBDR: Système de Gestion de Base de Données Relationnelles

Liste des Figures

I.1:	Représentation d'un Système d'Information.....	2
I.2:	Différents diagrammes UML.....	6
II.1:	Diagramme de Cas d'Utilisation globale.....	13
II.2:	Diagramme de Cas d'Utilisation gérer les rapports.....	14
II.3:	Diagramme de Cas d'Utilisation gérer les sanctions de conseil discipline.....	15
II.4:	Diagramme de Cas d'Utilisation gérer les réunions de conseil discipline.....	16
II.5:	Diagramme de Cas d'Utilisation gérer les utilisateurs.....	17
II.6:	Diagramme de Cas d'Utilisation gérer les annonces.....	18
II.7:	Diagramme de Cas d'Utilisation gérer les infraction.....	19
II.8:	Diagramme de Cas d'Utilisation gérer les membres de conseil discipline.....	20
II.9:	Diagramme séquence système liée au cas d'utilisation "S'authentifier".....	21
II.10:	Diagramme séquence système liée au cas d'utilisation "Ajouter un étudiant".....	22
II.11:	Diagramme séquence système liée au cas d'utilisation "Modifier un étudiant".....	23
II.12:	Diagramme séquence système liée au cas d'utilisation "Supprimer un étudiant".....	24
II.13:	Diagramme séquence système liée au cas d'utilisation " Consulter un étudiant".....	25
II.14:	Diagramme séquence système liée au cas d'utilisation "Imprimer un rapport".....	26
III.1:	Diagramme d'interaction "authentification d'un utilisateur".....	29
III.2:	Diagramme d'interaction "ajouter un rapport".....	30
III.3:	Diagramme d'interaction "modifier un rapport".....	31
III.4:	Diagramme d'interaction "supprimer un rapport".....	32
III.5:	Diagramme d'interaction" imprimer un rapport".....	33
III.6:	Diagramme des classes	34
IV.1:	Logo de l'université de Bejaia	38
IV.2:	L'interface d'authentification.....	39
IV.3:	Page D'accueil.....	40
IV.4:	Page gestion des utilisateurs.....	40
IV.5:	Page de gestion les étudiants.....	41



Liste des tableaux

Tableau 2.1: Description textuelle du cas d'utilisation « ajouter un rapport».....	14
Tableau 2.2: Description textuelle du cas d'utilisation « modifier une sanction».....	15
Tableau 2.3: Description textuelle du cas d'utilisation «Programmer une réunion».....	16
Tableau 2.4: Description textuelle du cas d'utilisation « ajouter un rapport».....	17
Tableau 2.5: Description textuelle du cas d'utilisation «Supprimer une annonce».....	18
Tableau 2.6: Description textuelle du cas d'utilisation «Supprimer une infractions».....	19
Tableau 2.7: Description textuelle du cas d'utilisation «Supprimer un membre».....	20
Tableau 3.1: Représentatif du dictionnaire de données.....	35

Introduction générale

De nos jours, une des tendances les plus en vue et qui concerne tout les secteurs de développement est l'informatisation.

Depuis l'apparition de l'informatique et son introduction dans le monde économique, les entreprises et entités publiques aspirent à optimiser et à rendre fiable la gestion de leurs structures internes.

Avant l'invention de l'ordinateur, on enregistrait toutes les informations manuellement sur des supports en papier ce qui engendrait beaucoup de problèmes tel que la perte de temps dans la recherche de ces informations ou la dégradation de ces dernières.

Ainsi, jusqu'à présent, l'ordinateur reste le moyen le plus sûr pour le traitement et la sauvegarde de l'information. Cette invention a permis d'informatiser les systèmes de données des entreprises, qui est la partie essentielle dans leur développement d'aujourd'hui.

La gestion automatisée peut être un élément capital de rentabilisation d'une entreprise dans la réalisation du processus de traitement de données. Ce traitement automatisé pourra permettre essentiellement l'allègement du travail de l'homme et fournir les résultats précis, opportuns et nécessaires à la prise de décisions. L'objectif de mon projet présenté dans ce rapport est la conception et la réalisation d'une application web de gestion du conseil de discipline du département informatique de la faculté des sciences exactes (université Abderrahmane mira Bejaia).

Pour la réalisation de ce projet, j'ai opté pour le langage de modélisation UML (Unifier Modeling Language), qui propose une architecture satisfaisante de modélisation orientée objet.

Pour l'implémentation, j'ai utilisé le langage laravel livewire.

Après avoir présenté les outils et la méthode adoptée, je vais maintenant exposer le plan du mémoire qui se subdivise en quatre principaux chapitres.

Dans le premier chapitre, je présente un état de l'art sur les systèmes d'informations, l'outil de modélisation UML.

Le second chapitre présente l'étude de l'existant et expose la problématique et les objectif visés.

Le troisième chapitre présente l'analyse conceptuelle de la solution proposée.

Dans le dernier chapitre, je vais détailler la réalisation de la solution proposée, qui présente la mise en place du nouveau système a l'aide du langage laravel livewire.

je finirai enfin par une conclusion générale.



I.1 Introduction

Dans ce chapitre je propose d'apporter des précisions quant aux termes et concepts qui seront utilisés dans ce présent travail en les circonscrivant dans le contexte de leur utilisation tant au plan général que celui particulier. Pour cela je vais commencer par une présentation des systèmes d'information(SI) d'une manière générale, puis je parlerai sur le langage de modélisation UML (Unified Modeling Language) que je vais utiliser pour modéliser les schémas de mon application.

I.2 Généralité sur les système d'information(SI)

Pour l'entreprise, les systèmes d'information qui permettent de traiter, d'acheminer et de sécuriser des données représentent un enjeu important, à la fois stratégique, économique, technique et humain. En facilitant la prise de décision, un système d'information bien conçu peut permettre d'obtenir des avantages concurrentiels importants et durables.

I.2.1 Définition

Le système d'information (SI) est un élément central d'une entreprise ou d'une organisation. Il permet aux différents acteurs de véhiculer des informations et de communiquer grâce à un ensemble de ressources matérielles, humaines et logicielles. Un SI permet de créer, collecter, stocker, traiter, modifier des informations sous divers formats.[1]

I.2.2 Représentation des Systèmes d'informations

Un système d'information d'une organisation peut être représenté comme deux triangles ou les problèmes et les besoins de l'organisation sont transformés aux données et traitements à l'aide des technologies de l'information. (Figure I.1)



Figure 1.2: Représentation d'un Système d'Information

I.2.3 Les fonctions d'un système d'information

Il existe donc 4 fonctions principales d'un SI :

- **Collecter** : c'est à partir de là que naît la donnée, qu'on acquière les informations provenant de l'environnement interne ou externe à l'entreprise.
- **Stocker** : dès que l'information est acquise, le système d'information la conserve. Elle doit pouvoir être disponible et doit pouvoir être conservée dans le temps.
- **Transformer/traiter** : cette phase permet de transformer l'information et choisir le support adapté pour traiter l'information. Ici ont construit de nouvelles informations en modifiant le fond ou la forme
- **Diffuser** : le SI transmet ensuite l'information dans son environnement interne ou externe.[1]

I.3 UML (Unified Modeling Language)

Le développement de n'importe quel système d'information nécessite une démarche très importante dans le cycle de vie d'un logiciel, la modélisation en pratique est importante dans le développement de logiciels, dans mon projet j'ai choisi le langage de modélisation UML que je vais exposer dans ce chapitre.

I.3.1 Définition

Le langage UML (Unified Modeling Language) est constitué de diagrammes intégrés utilisés par les développeurs informatiques pour la représentation visuelle des objets, des états et des processus dans un logiciel ou un système. Le langage de modélisation peut servir de modèle pour un projet et garantir une architecture d'information structurée ; il peut également aider les développeurs à présenter leur description d'un système d'une manière compréhensible pour les spécialistes externes. UML est principalement utilisé dans le développement de logiciels orientés objet. Les améliorations apportées à la norme dans la version 2.0 la rendent également adaptée à la représentation des processus de gestion.[2]

I.3.2 Historique de l'UML

Au début des années 90, une cinquantaine de méthodes d'analyse et de conception objet qui existaient, seulement trois d'entre se sont nettement détachées. En effet, la volonté de converger vers une méthode unifiée était déjà bien réelle et c'est pour cette raison que les méthodes OMT, BOOCH et OOSE se sont démarquées des autres.

OMT (Object Modeling Technique) de James Rumbaugh et BOOCH de Grady Booch ont été les deux méthodes les plus diffusées en France durant les années 90. Par ailleurs, OOSE de Ivar Jacobson s'est aussi imposée dans le monde objet pour la partie formalisation des besoins.

Pour aller plus loin dans le rapprochement, James Rumbaugh et Grady Booch se sont retrouvés au sein de la société Rational Software et ont été ensuite rejoints par Ivar Jacobson en se donnant comme objectif de fusionner leur méthode et créer UML (Unified Methode Language).

Il est important de noter que contrairement à ce qui avait été envisagé au départ, le processus de développement a été sorti du champ couvert par le projet de norme.

UML est donc une norme du langage de modélisation objet qui a été publiée, dans sa première version, en novembre 1997 par l'OMG (Object Management Group), instance de normalisation internationale du domaine de l'objet. En quelques années, UML s'est imposée comme standard à utiliser en tant que langage de modélisation objet.[3]

I.3.3 Les diagrammes de modélisation UML

Les diagrammes sont dépendant hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Il en existe quatorze depuis UML 2.3, Ces diagrammes sont regroupés dans deux grands ensembles(Figure I.3).

A. Les diagrammes de structurels(statiques)

Ces diagrammes, représentent l'aspect statique d'un système (classes, objets, composants...):[4]

- Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que leurs relations. Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux aspects temporels et dynamiques.
- Le diagramme d'objets, dans le langage de modélisation de donnée UML, permet de représenter les instances des classes, c'est-à-dire des objets. Comme le diagramme de classes, il exprime les relations qui existent entre les objets, mais aussi l'état des objets, ce qui permet d'exprimer des contextes d'exécution. En ce sens, ce diagramme est moins général que le diagramme de classes.
- Le diagramme de composant : Ce diagramme représente les différents constituants du logiciel au niveau de l'implémentation d'un système.
- Le diagramme de déploiement : Ce diagramme décrit l'architecture technique d'un système avec une vue centrée sur la répartition des composants dans la configuration d'exploitation.
- Le diagramme de paquetage : Ce diagramme donne une vue d'ensemble du système structuré en paquetage. Chaque paquetage représente un ensemble homogène d'éléments du système (classes, composants...).
- Le diagramme de structure composite : Ce diagramme permet de décrire la structure interne d'un ensemble complexe composé par exemple de classes ou d'objets et de composants techniques. Ce diagramme met aussi l'accent sur les liens entre les sous-ensembles qui collaborent.
- Le diagramme de profils et un diagramme de structure permettant l'utilisation de profils pour un métamodèle donné. Apparu avec UML 2.2, ce diagramme fournit une représentation des concepts utilisés dans la définition des profils (packages, stéréotypes, application de profils).[8]

B. Les diagrammes de comportement

Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs. Sept diagrammes sont proposés par UML :[4]

- Diagramme des cas d'utilisation : Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un des diagrammes les plus structurants dans l'analyse d'un système.

- Diagramme d'état-transition : Ce diagramme montre les différents états des objets en réaction aux événements.

- Diagramme d'activités : Ce diagramme donne une vision des enchaînements des activités propres à une opération ou à un cas d'utilisation. Il permet aussi de représenter les flots de contrôle et les flots de données.

- Diagramme de séquence : Ce diagramme permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets.

- Diagramme de communication : Ce diagramme est une autre représentation des scénarios des cas d'utilisation qui met plus l'accent sur les objets et les messages échangés.

- Diagramme global d'interaction : Ce diagramme fournit une vue générale des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d'activités.

- Diagramme de temps : Ce diagramme permet de représenter les états et les interactions d'objets dans un contexte où le temps a une forte influence sur le comportement du système à gérer.

- Diagramme global d'interaction ou diagramme d'interactivité est un diagramme UML version 2.0 utilisé pour rendre compte de l'organisation spatiale des participants à l'interaction.

➤ Figure I.2: représente les diagrammes UML existé.

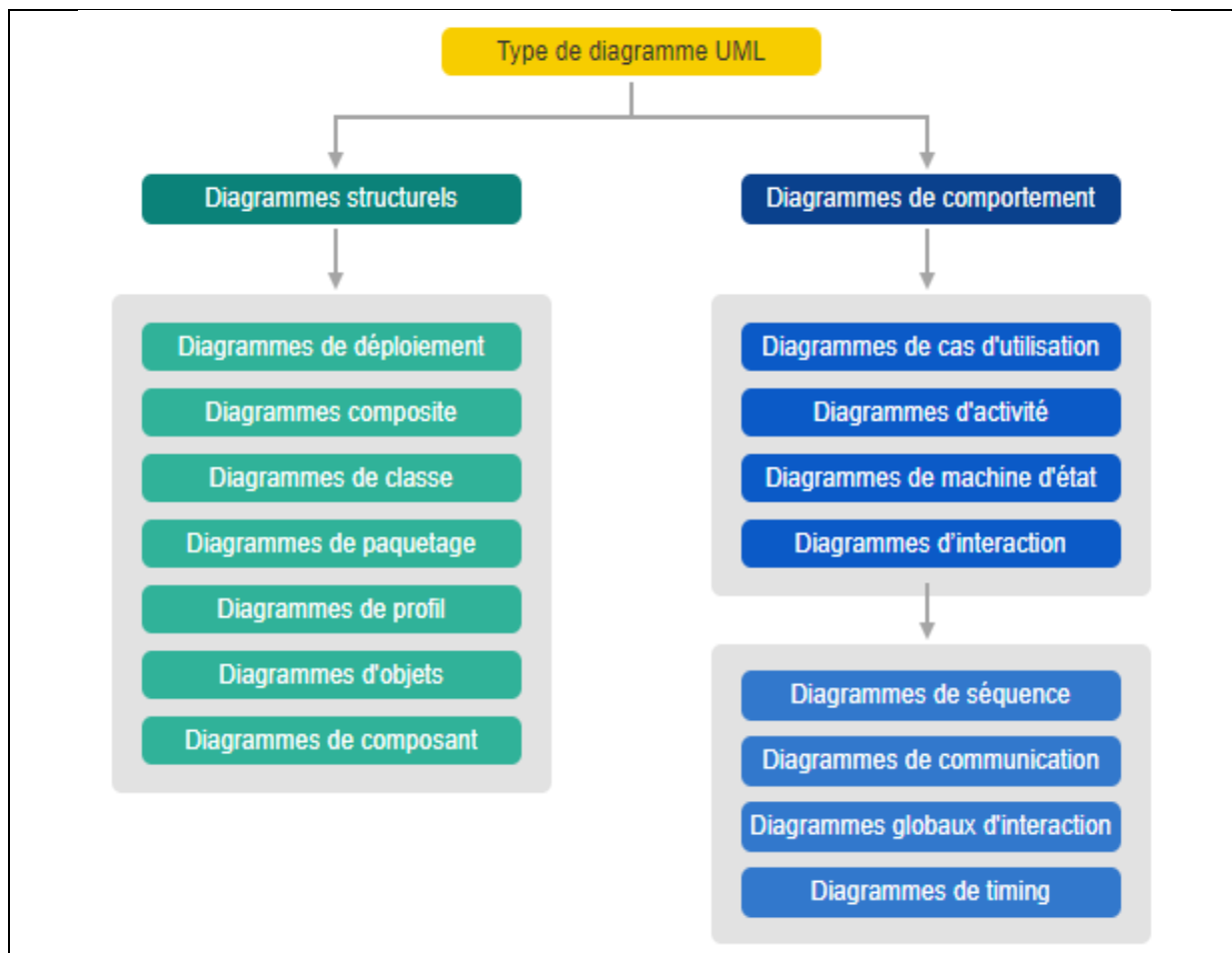


Figure 1.3: Différents diagramme UML

Conclusion

A l'issu de ce chapitre, j'ai réussi à développer une idée générale sur le concept de SI qui est un élément central d'une entreprise ou d'une organisation, et de présenter le langage de modélisation UML, qui me sont indispensables à la réalisation de mon projet.

II.1 Introduction

Cette phase consiste à comprendre le contexte du système. Il s'agit de déterminer les fonctionnalités et les acteurs les plus pertinents de cette application, puis je termine cette phase par la présentation des diagrammes des cas d'utilisations et de séquences de mon application.

Mon objectif dans cette étape est donc d'exprimer les besoins attendus du futur système à développer.

II.2 Spécification des besoins

II.2.1 Présentation de l'organisme d'accueil

Il s'agit du département d'informatique des sciences exactes de l'université de Bejaia qui a été créé en fin d'année 2002. Le but principal du département est de donner aux étudiants des bases solides en informatique, mais aussi d'offrir à ceux qui le souhaitent l'envie et la possibilité de se spécialiser dans ce domaine. Les cours proposés couvrent aussi bien des aspects fondamentaux essentiels et des enseignements d'approfondissement aux frontières du domaine.

Le département d'informatique est doté d'un laboratoire de recherche en Informatique médicale (LIMED) dont l'objectif est de créer et de mettre en place des équipes de recherche compétitives dans différents domaines de l'informatique.

II.2.1.1 Les formations offertes par le département

Le département informatique offre plusieurs formations dans de différentes spécialités de l'informatique et pour les différents niveaux qui sont cités ci-dessous :

- ❖ Deux licences académiques en Informatique :
 - Licence à recrutement National (L1+L2+L3).
 - Licence générale pour les étudiants ayant réussi la première année MI (L2+L3)
- ❖ Deux Masters académiques :
 - Réseaux et systèmes distribués (Resyd).
 - Intelligence Artificielle (IA)
- ❖ Deux Masters professionnels :
 - Administration et Sécurité des réseaux (ASR).
 - Génie Logiciel (GL).
- ❖ Doctorat en Informatique :

Le département propose une formation doctorale en trois spécialités:

 - Intelligence Artificielle
 - Génie Logiciel
 - Réseaux et Systèmes Distribués

II.2.1.2 Personnel d'enseignement de département informatique

Les étudiants du département d'informatique sont encadrés par un personnel enseignant spécialisés et compétents qui se compose de:

- Professeurs : 05
- Maîtres de conférences « A » : 08
- Maîtres de conférences « B » : 18
- Maîtres assistants « A » : 30
- Maîtres assistants « B » : 05

II.2.2 Conseils de Discipline Universitaires

II.2.2.1 Définition

C'est une réunion d'enseignants destinée à examiner le comportement inapproprié d'un étudiant.[5]

II.2.2.2 Création d'un Conseil de Discipline

Il est créé au sein de chaque établissement d'enseignement supérieur par la décision de chef de l'établissement:

- Un conseil de Discipline de l'établissement.
- Un conseil de Discipline par structure de l'établissement: département ou faculté.

II.2.2.3 Les membres d'un Conseil de Discipline

Le conseil de Discipline est composé:

- De cinq membres titulaires et cinq membres suppléants, élus par et parmi les enseignants de la structure concernée.
- D'un représentant des étudiants titulaire et un suppléant, élus par et parmi les étudiants de la structure concernée.
- Il est présidé par le premier responsable de la structure concernée ou son représentant.

II.2.2.4 Différents Conseils de Disciplines existant à l'université

Les conseils de discipline sont hiérarchisés et organisés de la sorte:

- Le conseil de discipline de Département est compétent pour toutes les infractions du premier degré.
- le conseil de discipline de la Faculté est compétent pour les infractions de premier et deuxième degrés dont la sanction d'exclusion n'excède pas une année universitaire.
- Le conseil de discipline de l'Université est compétent pour toutes les infractions du deuxième degré.

➤ Remarque

Le conseil de discipline de l'Université est compétent pour l'étude des recours.[5]

II.2.2.5 Les type d'infractions:

Dans les universités il existe deux type d'infraction:

A. Infractions du 1er degré

Sont des infractions qui sont traiter au niveau de conseil de discipline de Département ou au niveau de conseil de discipline de la Faculté comme :

- Toute tentative de fraude, fraude établie ou fraude préméditée établie à un examen.
- Tout refus d'obtempérer à des directives émanant de l'administration, du personnel enseignant-chercheur ou de sécurité.
- Toute demande non fondée de double correction.

B. Infractions du 2ème degré

Sont des infractions qui sont traiter au niveau de conseil de discipline de la Faculté ou au niveau de conseil de discipline de l'Université comme:

- Les récidives des infractions du 1er degré
- L'entrave à la bonne marche de l'établissement, le désordre organisé, la violence, les menaces et voies de faits de toute nature.
- La détention de tout moyen avec l'intention établie de porter atteinte à l'intégrité physique du personnel enseignant-chercheur, du personnel administratif, technique et de service, et des étudiants.
- Le faux et usage de faux, la falsification et la substitution de documents pédagogiques et administratifs.
- L'usurpation d'identité.
- La diffamation à l'égard de l'ensemble du personnel universitaire et des étudiants,
- Les actions délibérées de perturbation et de désordre caractérisés portant atteinte au bon déroulement des activités pédagogiques telles que les entraves aux enseignements et aux examens ou leur boycott, le regroupement perturbateur...,
- Le vol, l'abus de confiance et le détournement de biens de l'établissement, des enseignants et des étudiants.
- La détérioration délibérée des biens de l'établissement : matériels, mobiliers et accessoires.
- Les insultes et propos irrévérencieux à l'égard de l'ensemble du personnel, enseignants-chercheurs, personnel administratif, technique et de service, et des étudiants.
- Le refus d'obtempérer à un contrôle réglementaire dans l'enceinte de l'établissement.

II.2.2.6 Sanctions pour infractions présumées

- ✓ Les sanctions applicables aux infractions du premier degré sont fixées comme suit :
 - Avertissement.
 - Avertissement écrit versé au dossier pédagogique de l'étudiant.
 - Blâme versé au dossier pédagogique de l'étudiant.
- ✓ Les sanctions applicables aux infractions du deuxième degré sont fixées comme suit :

- Exclusion pour un semestre ou une année
- Exclusion pour deux ans
- Exclusion définitive.[6]

II.2.3 La description du contexte du projet

Dans mon étude je m'intéresse au service gestion de Conseil de Discipline qui se trouve au sein du département informatique, il est créé par la décision du chef de département, il est composé de cinq membres titulaires et cinq membres suppléants élus par et parmi les enseignants de la structure concernée, d'un représentant d'étudiants titulaire et un suppléant, élus par et parmi les étudiants de la structure concernée. et il est présidé par le premier responsable de la structure concernée ou son représentant.

Mon sujet d'étude s'intitule : Conception et réalisation d'une application web pour la gestion du conseil discipline pour le département informatique qui lui permettra de:

- Créer un dossier disciplinaire pour chaque étudiant
- Stocker des dossier disciplinaire dans des archivage électronique
- Chercher un dossier disciplinaire d'un étudiant
- Envoyer des convocation a des étudiants
- Envoyer des emails aux membres de conseil discipline

Ce thème a été proposé et choisi dans le but d'essayer de résoudre les problèmes et les difficultés que rencontre quotidiennement le personnel du service.

II.2.4 Problématique

Vu l'importance des tâches qui s'effectuent quotidiennement au niveau de département, sont fastidieuses et complexes demeure à l'état manuel.

Aux cours des interviews, que nous avons réalisés avec le responsable, nous sommes arrivés à confirmer l'existence de ces défaillances qu'on résume comme suit :

- Difficulté d'organiser des réunions.
- Difficulté à convoquer les membres du Conseil.
- Risque de perte des informations concernant l'état disciplinaire des étudiants.
- Masse d'informations importantes traitées manuellement.
- Possibilité d'erreur dans le remplissage des différents documents et registres manuellement.
- Le manque de souplesse dans le travail.
- perte de temps .
- etc.

II.2.5 Solutions

Implémenter une base de données qui gardera la trace de toutes les informations concernant les étudiants du département, les enseignants et membres du conseil.

Établir une application efficace répondant aux besoins immédiats du département, ergonomique et facile à utiliser.

II.2.6 Les objectifs de mon application

L'application de la gestion de conseil discipline a pour objectif de:

- A. Améliorer la gestion des informations en les centralisant.
- B. Constituer une banque de données.
- C. Accéder à l'historique des étudiants précédents, leur état disciplinaire...
- D. Identifier les règles de gestion du SI de conseil discipline .
- E. Etablir les documents qualité relatifs à la gestion du dossier.
- F. Améliorer la disponibilité du dossier et des informations qu'il contient, y compris en cas d'urgence.

II.3 Analyse des besoins

II.3.1 L'identification des besoins

UP distingue deux types de besoins :

- Les besoins fonctionnels qui conduisent à l'élaboration des cas d'utilisation.
- Les besoins techniques (non fonctionnels) qui aboutissent à la rédaction d'une des exigences du système pour sa réalisation et son bon fonctionnement.

II.3.1.1 Les besoins fonctionnels

- Avoir une base de données pour le stockage des étudiants, des enseignants, des infractions avec leur sanctions, et des décisions des membres du conseil.
- Manipulation et mise à jour de la base de données.
- Chaque utilisateur possède un login et un mot de passe unique pour accéder à cette application.
- L'administrateur peut ajouter, supprimer, modifier ses données, gérer les utilisateur, les décisions du conseil de discipline, les réunions du conseil de discipline, les rapports etc. Tandis que l'enseignant peut gérer les rapports, et l'étudiant n'a que le droit de les consulter.
- Avoir un historique des étudiants concernant leurs état disciplinaire.

II.3.1.2 Les besoins non fonctionnels

- L'application doit permettre de gérer les accès des utilisateurs selon un privilège et un état d'activation de chaque compte.
- Il faut garantir la sécurité d'accès à l'espace administrateur afin de protéger les données personnelles des utilisateurs..
- L'interface de cette application doit être ergonomique, conviviale et voire même apte à aider l'utilisateur à mieux gérer son espace de travail.

II.3.2 L'identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié.

Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données.[7]

II.3.2.1 Les acteurs principaux

Les acteurs principaux sont:

1. Les administrateurs:

Le rôle des administrateurs est de :

- Gérer les rapports.
- Gérer les sanctions du conseil de discipline.
- Gérer les membres du conseil de discipline.
- Gérer les réunions du conseil de discipline.
- Gérer les utilisateurs.
- Gérer les annonces.
- Gérer les infractions.
- Gérer les Sessions.
- Consulter les informations.

2. Les enseignants:

Le rôle de l'enseignant est limité:

- Gérer les rapports.
- Consulter les informations.

3. Les étudiants:

Le rôle des étudiants est limité:

- Consulter les informations.

II.3.3 L'identification des cas d'utilisations de l'application

Les principales fonctionnalités de l'application à concevoir sont régies autour des besoins des utilisateurs. Elles sont illustrées dans le diagramme de cas d'utilisation suivant :

1. Le diagramme de cas d'utilisation globale

La figure II.1 sise ci-dessous présente le digramme globale des cas d'utilisation de mon application

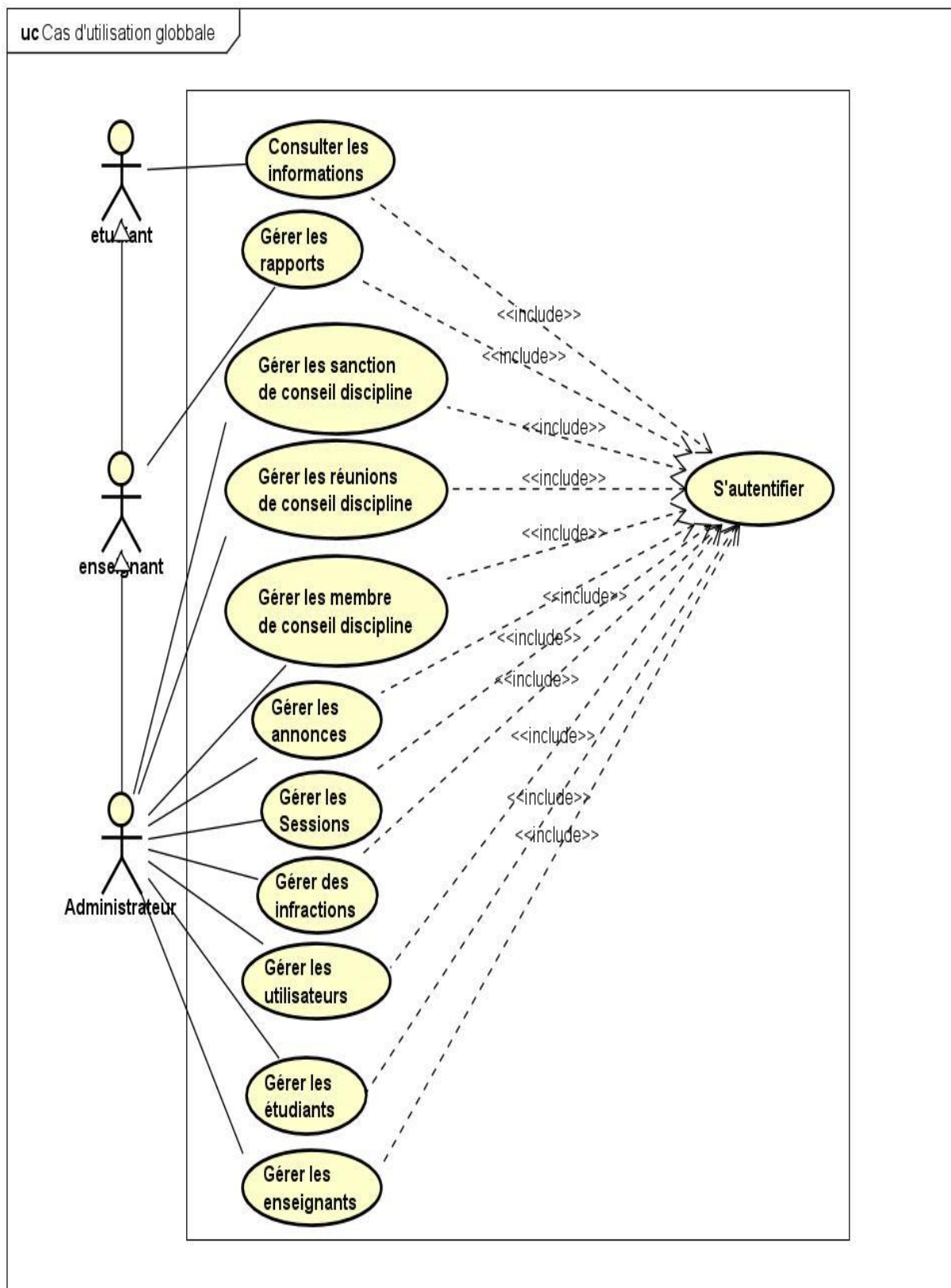


Figure II.1: Diagramme de Cas d'Utilisation globale

A l'issue de l'expression des besoins à l'aide du diagramme des cas d'utilisation globale, dans ce qui suit, nous détaillons chacun des cas d'utilisations présente dans les figure(II.2) jusqu'à la figure (II.8), en donnant sa description textuelle.

2. Le cas d'utilisation «gérer les rapports»

La figure II.2 présente le raffinement du cas d'utilisation intitulé «gérer les rapports».

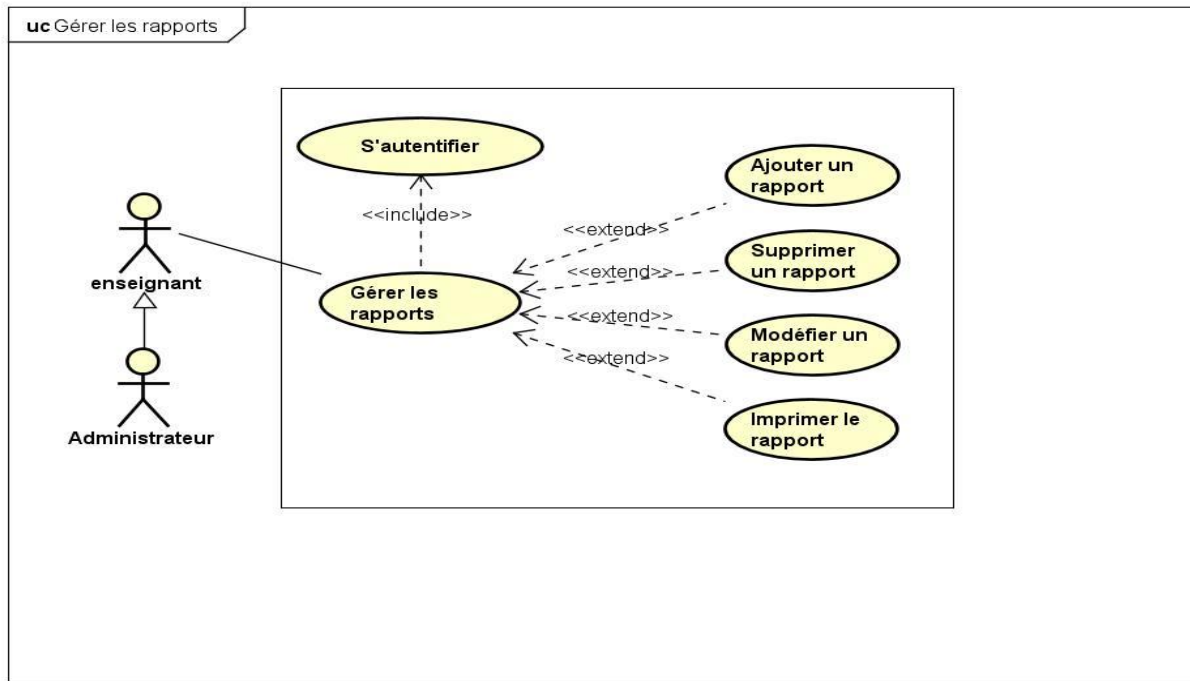


Figure II.2: Diagramme de Cas d'Utilisation gérer les rapports

A.2 Description du cas d'utilisation «Ajouter un rapport»

Le tableau 2.1 présente la description textuelle du cas d'utilisation « ajouter un rapport»

Titre : Ajouter un rapports
Acteur : Administrateur, enseignant
Pré-Condition : S’authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'utilisateur accède au menu rapport. 2. Le système affiche l’interface de rapport. 3. L'utilisateur choisit d'ajouter un rapport. 4. L'utilisateur doit remplir les informations nécessaires et valider. 5. Le système enregistrer le rapport et affiche un message de succès. 6. l'utilisateur doit imprime le rapport.
Scénario alternatif : <ol style="list-style-type: none"> 4. Si les informations de rempli par l'utilisateur sont incomplètes, le système affiche un message d’erreur. 5. Reprise de l’étape 2 du scénario nominal.
Poste-Condition : un rapport est ajouté.

3. Le cas d'utilisation «gérer les sanctions de conseil discipline»

La figure II.3 présente le raffinement du cas d'utilisation intitulé «gérer les sanctions de conseil discipline».

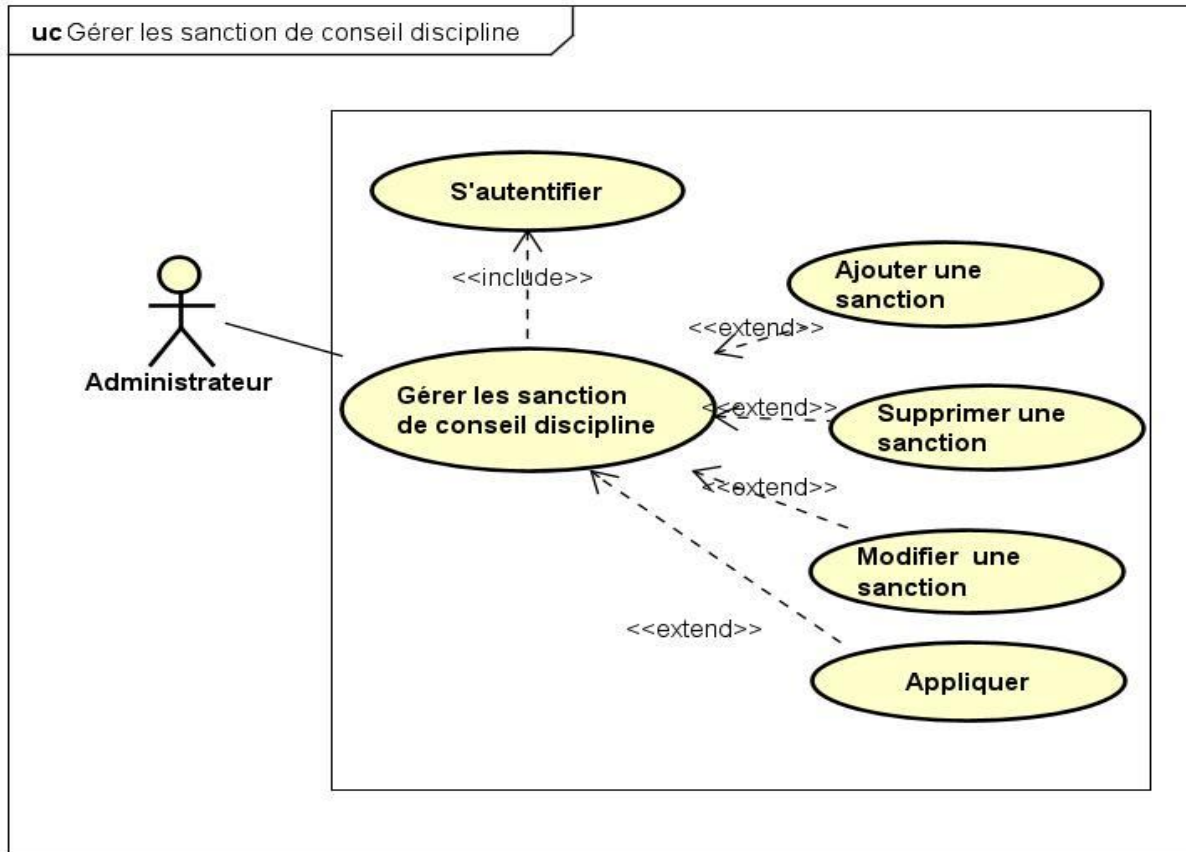


Figure II.3: Diagramme de Cas d'Utilisation gérer les sanctions de conseil discipline

B.2 Description du cas d'utilisation «modifier une sanction»

Le tableau 2.2 présente la description textuelle du cas d'utilisation « modifier une sanction»

Titre : modifier une sanction
Acteur : Administrateur
Pré-Condition : S\'authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L' administrateur choisit la sanction à modifier. 2. Le système affiche le formulaire de modification. 3. Il modifie la sanction et valide. 4. Le système met à jour les informations dans la base de données. 5. Le système actualise la liste des sanctions.
Scénario alternatif : <ol style="list-style-type: none"> 4. Si les informations de remplis par l' administrateur sont incomplètes, le système affiche un message d'erreur. 5. Reprise de l'étape 2 du scénario nominal.
Poste-Condition : une sanction est modifier.

4. Le cas d'utilisation «gérer les réunions de conseil discipline»

La figure II.4 présente le raffinement du cas d'utilisation intitulé «gérer les réunions de conseil discipline».

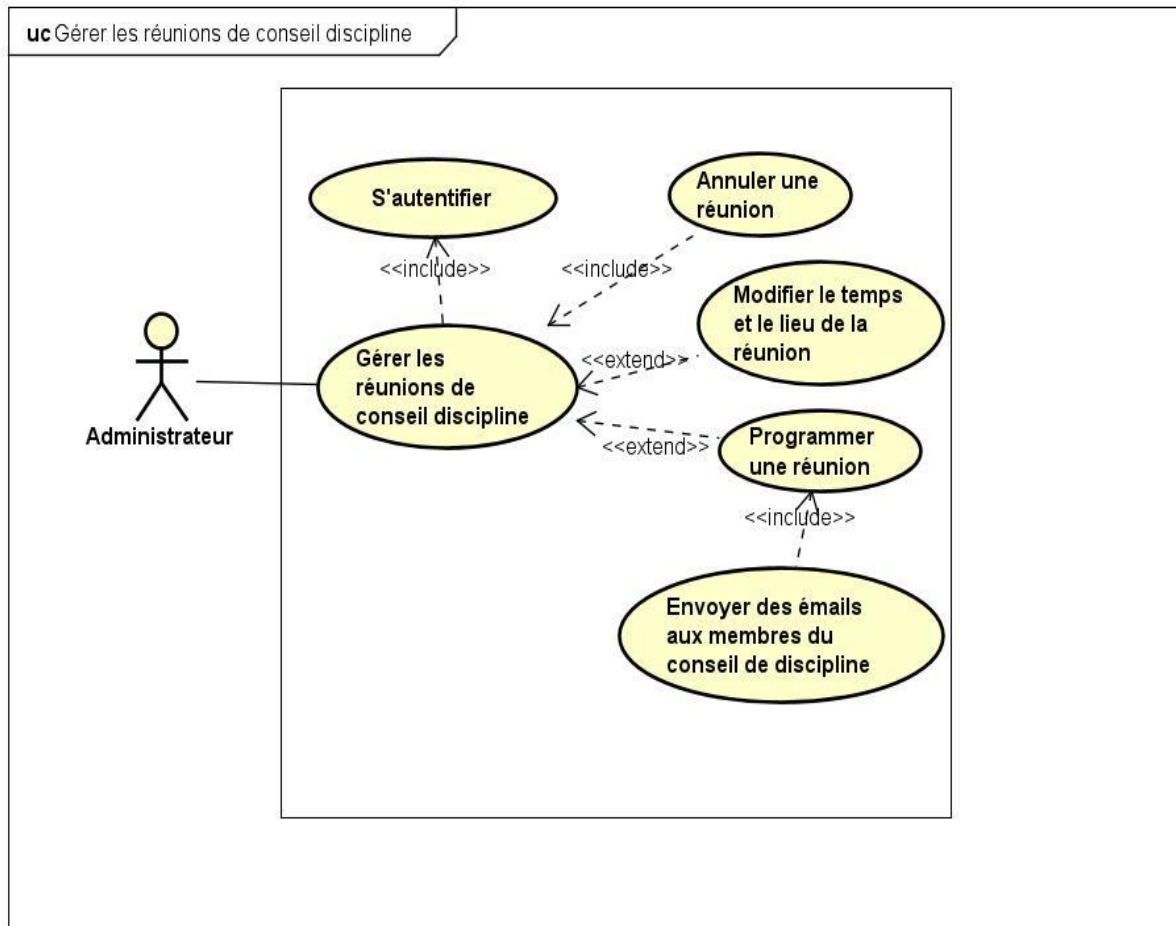


Figure II.4: Diagramme de Cas d'Utilisation gérer les réunions de conseil discipline

C.2 Description du cas d'utilisation «Programmer une réunion»

Le tableau 2.3 présente la description textuelle du cas d'utilisation «Programmer une réunion»

Titre : Programmer une réunion
Acteur : Administrateur
Pré-Condition : S\'authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'administrateur accède au menu réunion. 2. Le système affiche l'interface de réunion. 3. L'administrateur choisit le temps de la réunion et il a valide. 4. Le système enregistrer la réunion et affiche un message de succès. 5. Le système envoie un email aux membre de conseil discipline.
Scénario alternatif : <ol style="list-style-type: none"> 1. Si les informations de rempli par l'utilisateur sont incomplètes, le système affiche un message d'erreur. 2. Reprise de l'étape 2 du scénario nominal.
Poste-Condition : une réunion est Programmé.

5. Le cas d'utilisation «gérer les utilisateurs»

La figure II.5 présente le raffinement du cas d'utilisation intitulé «gérer les utilisateurs».

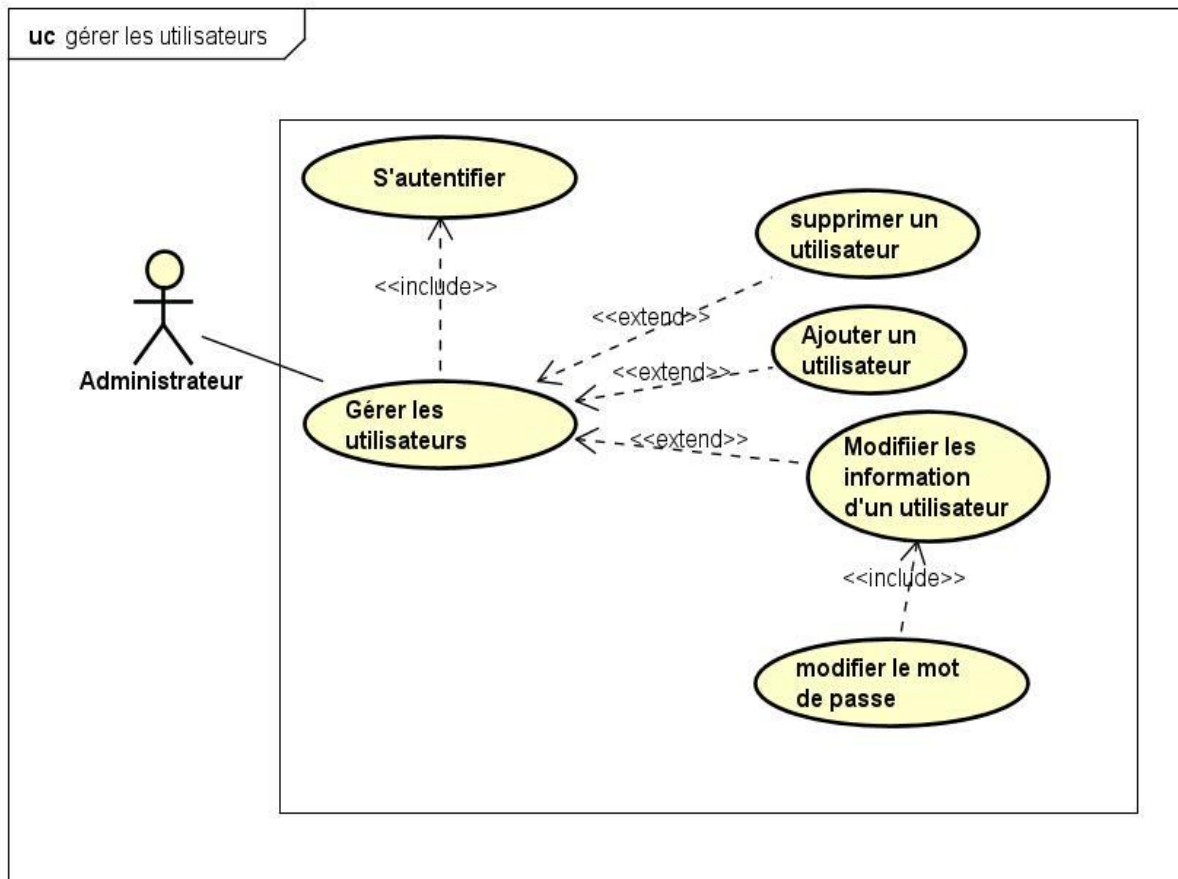


Figure II.5: Diagramme de Cas d'Utilisation gérer les utilisateurs

D.2 Description du cas d'utilisation «Ajouter un utilisateurs»

Le tableau 2.4 présente la description textuelle du cas d'utilisation « ajouter un rapport»

Titre : Ajouter un rapports
Acteur : Administrateur
Pré-Condition : S'authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'administrateur choisit d'ajouter un utilisateur. 2. Le système affiche le formulaire à remplir. 3. L'administrateur rempli et valide le formulaire. 4. Le système ajoute les informations dans la base de donnée. 5. Le système actualise la liste des utilisateurs.
Scénario alternatif : <ol style="list-style-type: none"> 3. Si les informations de rempli par l'utilisateur sont incomplètes, le système affiche un message d'erreur. 4. Reprise de l'étape 2 du scénario nominal.
Poste-Condition : un utilisateur est ajouté.

6. Le cas d'utilisation «gérer les annonces»

La figure II.6 présente le raffinement du cas d'utilisation intitulé «gérer les annonces».

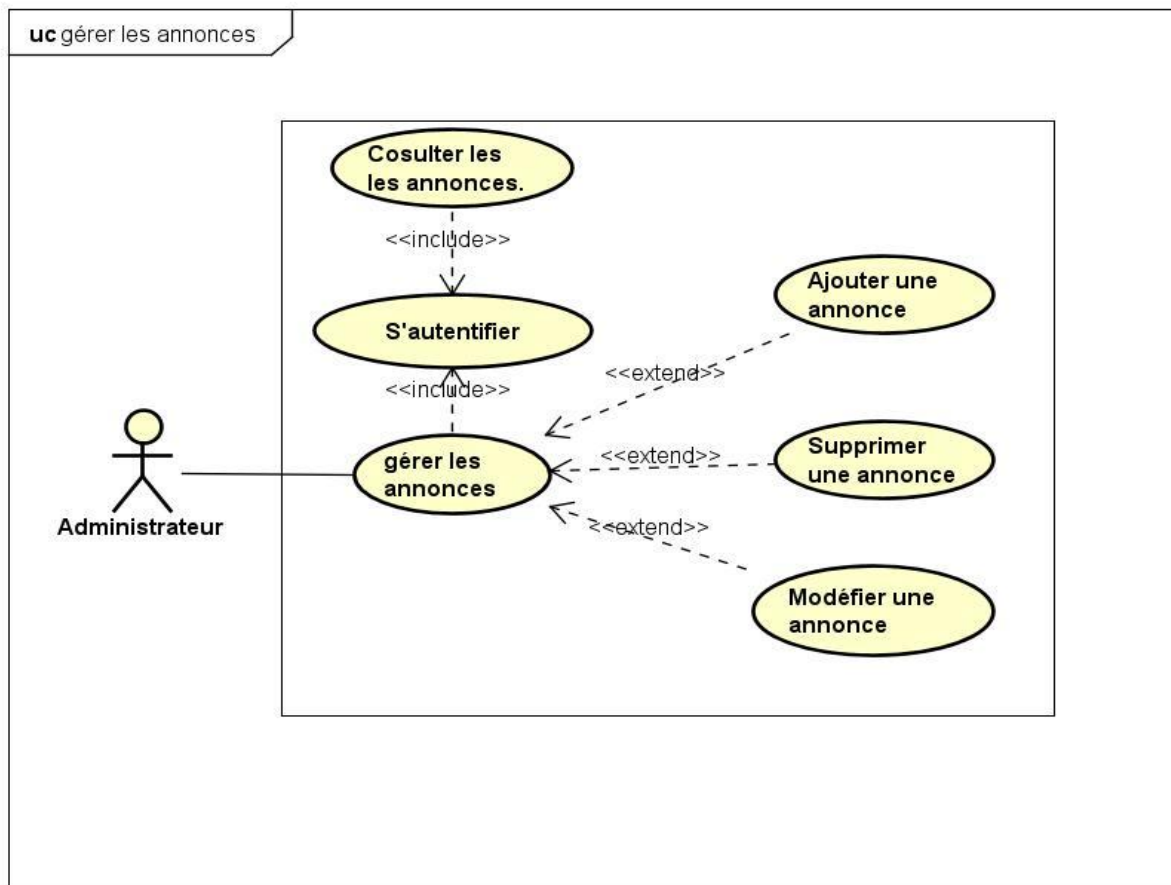


Figure II.6: Diagramme de Cas d'Utilisation gérer les annonces

E.2 Description du cas d'utilisation «Supprimer une annonce»

Le tableau 2.5 présente la description textuelle du cas d'utilisation «Supprimer une annonce».

Titre : Supprimer une annonce
Acteur : Administrateur
Pré-Condition : S'authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'administrateur accéder au menu annonce 2. le système affiche le menu annonces 3. L'administrateur choisit l'annonce à supprimer. 4. Le système demande une confirmation. 5. L'administrateur confirme ou annule la suppression. 6. Le système supprime l'annonce de la base de donnée.
Scénario alternatif : <ol style="list-style-type: none"> 5. Si l'annonce choisit par l'utilisateur n'existe pas, le système affiche un message d'erreur. 6. Reprise de l'étape 2 du scénario nominal.
Poste-Condition : une annonce est supprimer

7. Le cas d'utilisation «gérer les infractions»

La figure II.7 présente le raffinement du cas d'utilisation intitulé «gérer les infractions».

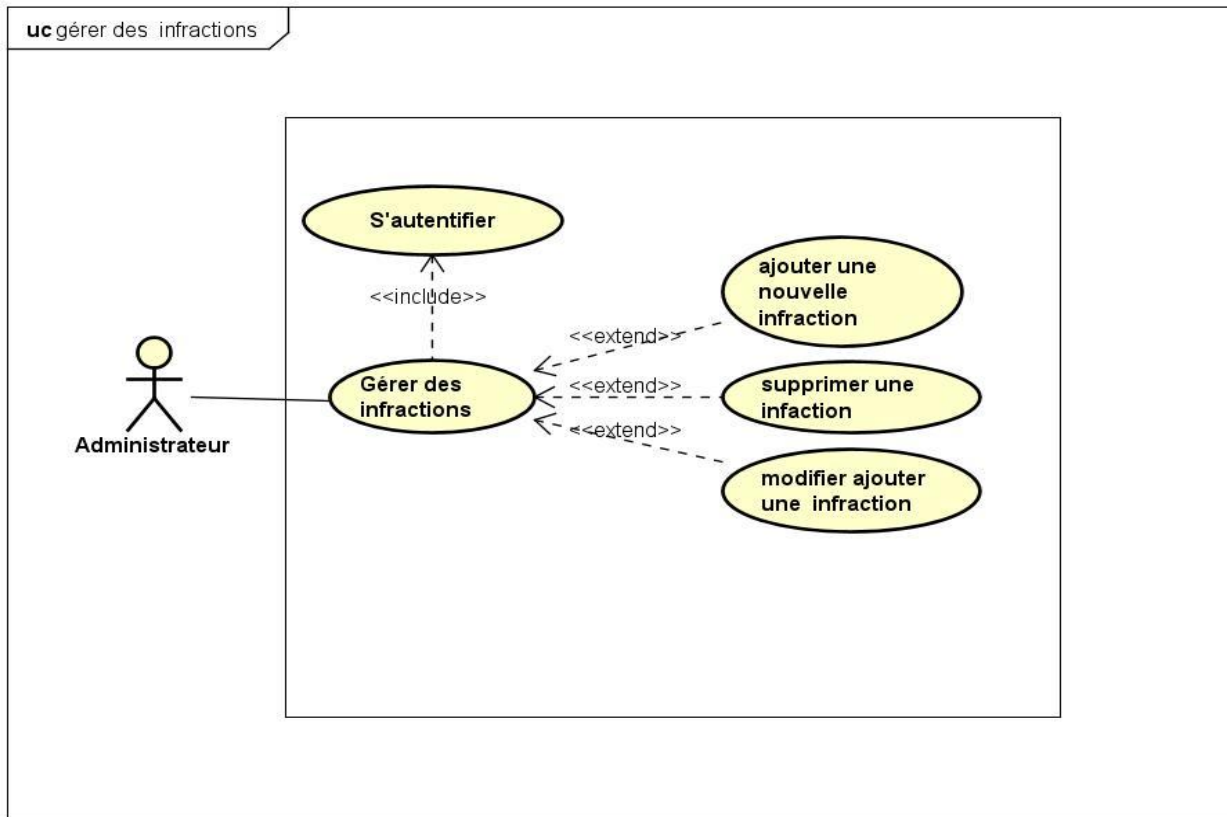


Figure II.7: Diagramme de Cas d'Utilisation gérer les infractions

F.2 Description du cas d'utilisation «Supprimer une infractions»

Le tableau 2.6 présente la description textuelle du cas d'utilisation «Supprimer une infractions».

Titre : Supprimer une infractions
Acteur : Administrateur
Pré-Condition : S'authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'administrateur accéder au menu infractions. 2. le système affiche le menu infractions. 3. L'administrateur choisit l' infraction à supprimer. 4. Le système demande une confirmation. 5. L'administrateur confirme ou annule la suppression. 6. Le système supprime l infraction de la base de données.. 7. Le système mettre a jour la liste des infractions
Scénario alternatif : <ol style="list-style-type: none"> 1. Si l annonce choisit par l'utilisateur n'existe pas, le système affiche un message d'erreur. 2. Reprise de l'étape 2 du scénario nominal.
Poste-Condition : une infraction est supprimer

8. Le cas d'utilisation gérer les membres de conseil discipline»

La figure II.8 présente le raffinement du cas d'utilisation intitulé «gérer les membres de conseil discipline».

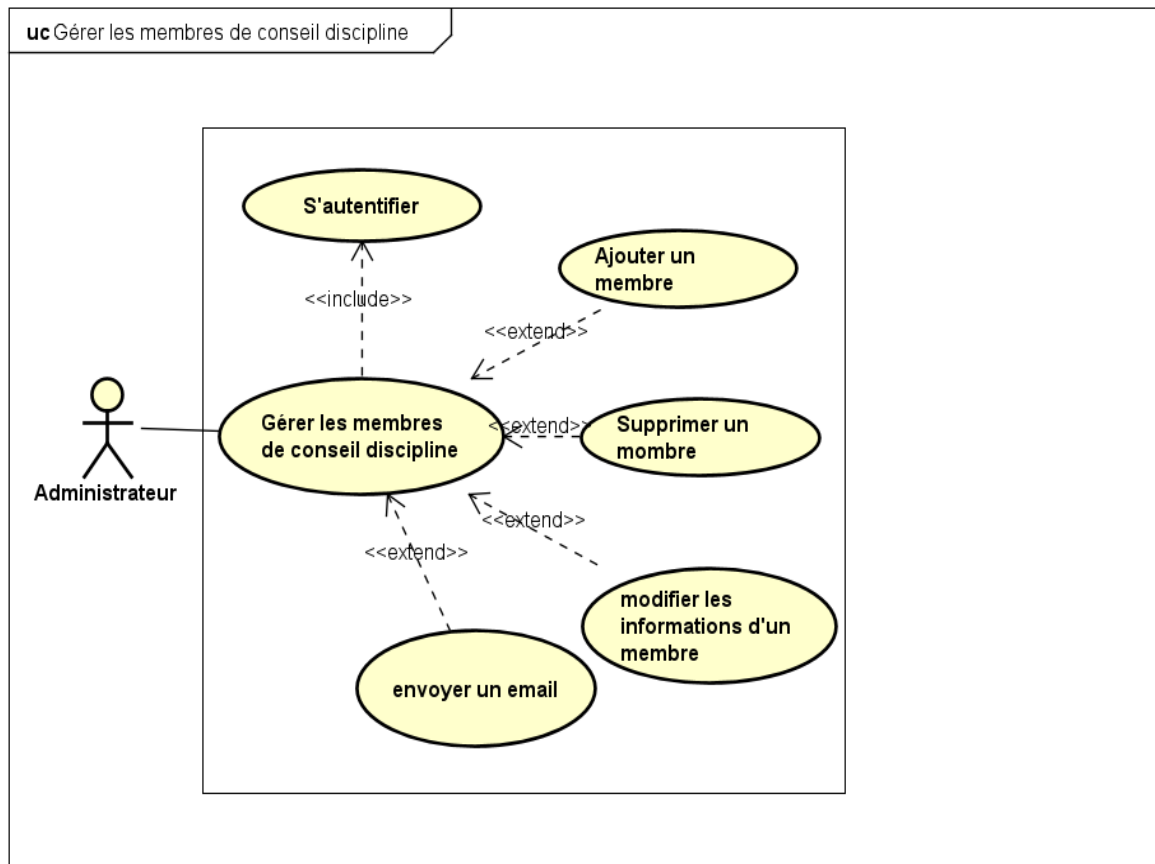


Figure II.8: Diagramme de Cas d’Utilisation gérer les membres de conseil discipline.

G.2 Description du cas d’utilisation «Supprimer un membre»

Le tableau 2.7 présente la description textuelle du cas d’utilisation «Supprimer un membre».

Titre : Supprimer un membre
Acteur : Administrateur
Pré-Condition : S’authentifier
Scénario Nominale : <ol style="list-style-type: none"> 1. L'administrateur accéder au menu membre 2. le système affiche le menu membre 3. L'administrateur choisit membre à supprimer. 4. Le système demande une confirmation. 5. L'administrateur confirme ou annule la suppression. 6. Le système supprime le membre dans la base de données. 7. Le système mettre a jour la liste des membre.
Scénario alternatif : <ol style="list-style-type: none"> 7. Si il y'a un Bug, le système affiche un message d’erreur. 8. Reprise de l’étape 2 du scénario nominal.
Poste-Condition : un membre est supprimer

II.3.4 Le diagramme des séquences

Les opérateurs d'interaction que j'ai utilisés dans les diagrammes de séquences sont :

- Référence (ref) : cet opérateur désigne que le fragment fait référence a un cas vue précédemment.
- Alternative(Alt) :cet opérateur désigne que le fragment composé représente un choix de comportement. Une opérande d'interaction au maximum sera choisie. L'opérande choisi doit avoir une expression de garde implicite ou explicite qui a la valeur "true" à ce point de l'interaction.
- Loop : cet opérateur désigne que le fragment composé représente une boucle. L'opérande "loop" sera répétée plusieurs fois.

Les figures (II.9) jusqu'à (II.14) présenterons respectivement les diagrammes des séquences.

A. Diagramme de séquence s'authentifier

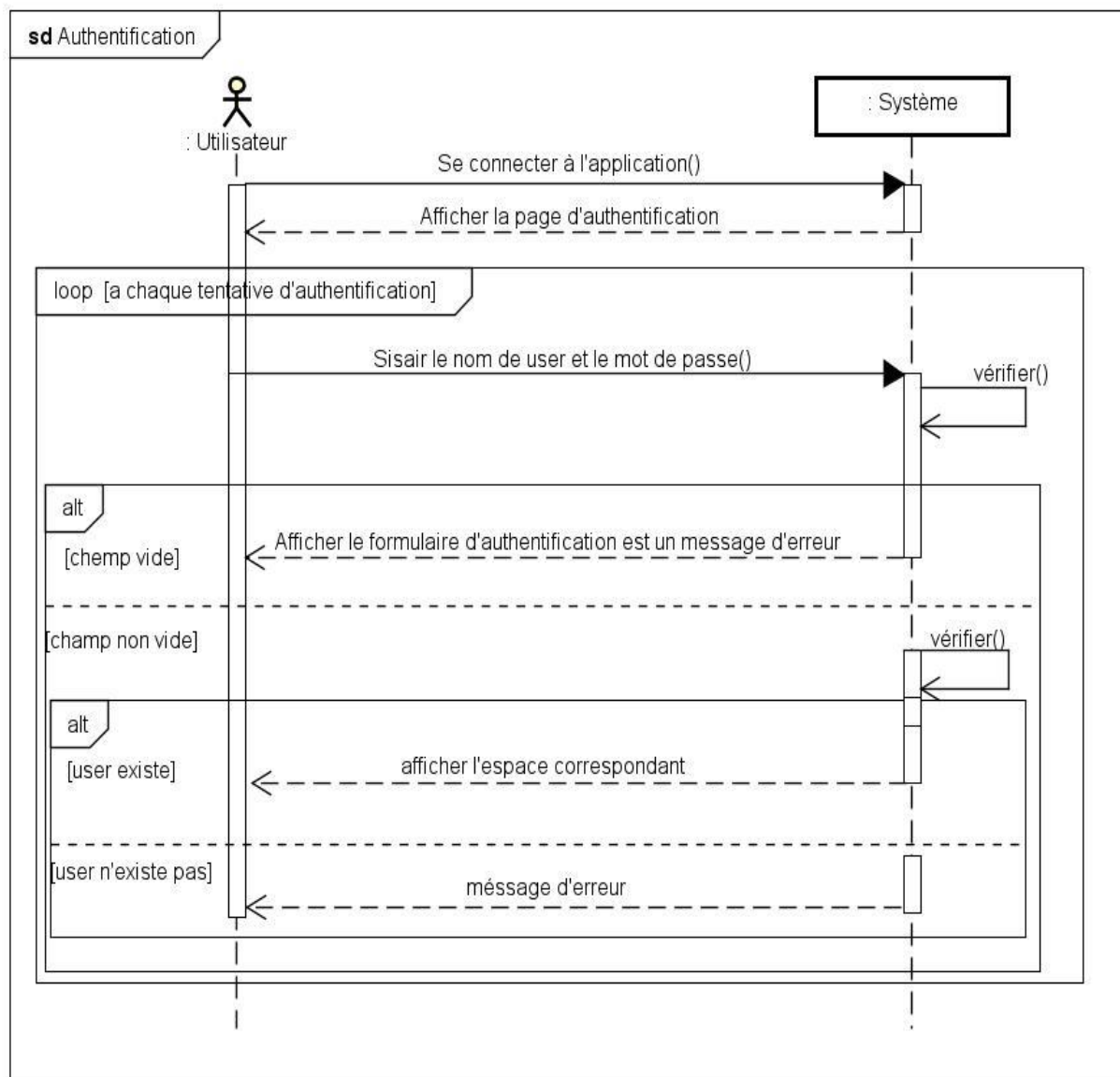


Figure II.9: Diagramme séquence système liée au cas d'utilisation "S'authentifier"

B. Diagramme de séquence ajouter un dossier étudiant

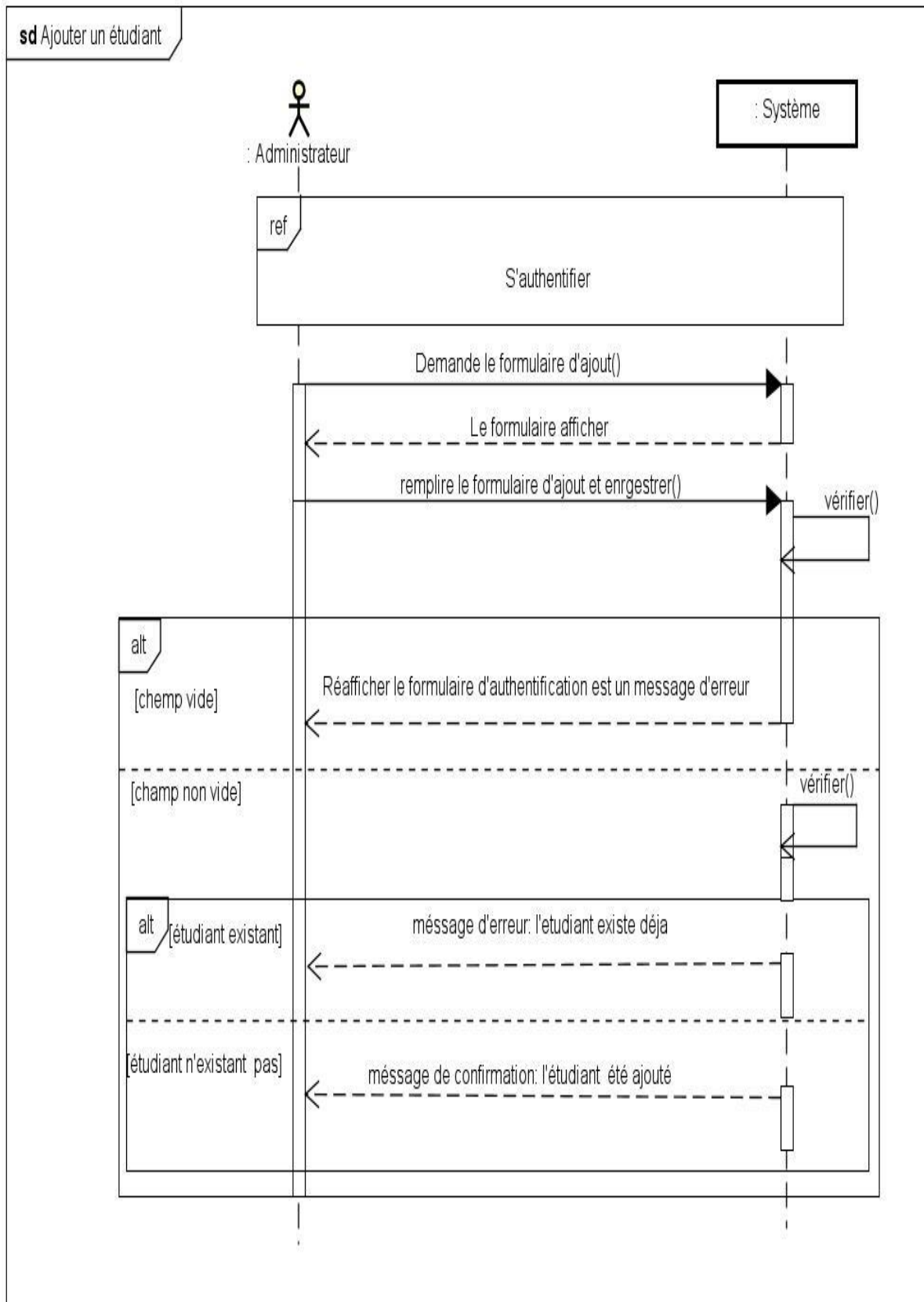


Figure II.10: Diagramme séquence système liée au cas d'utilisation "Ajouter un étudiant"

C. Diagramme de séquence modifier un dossier étudiant

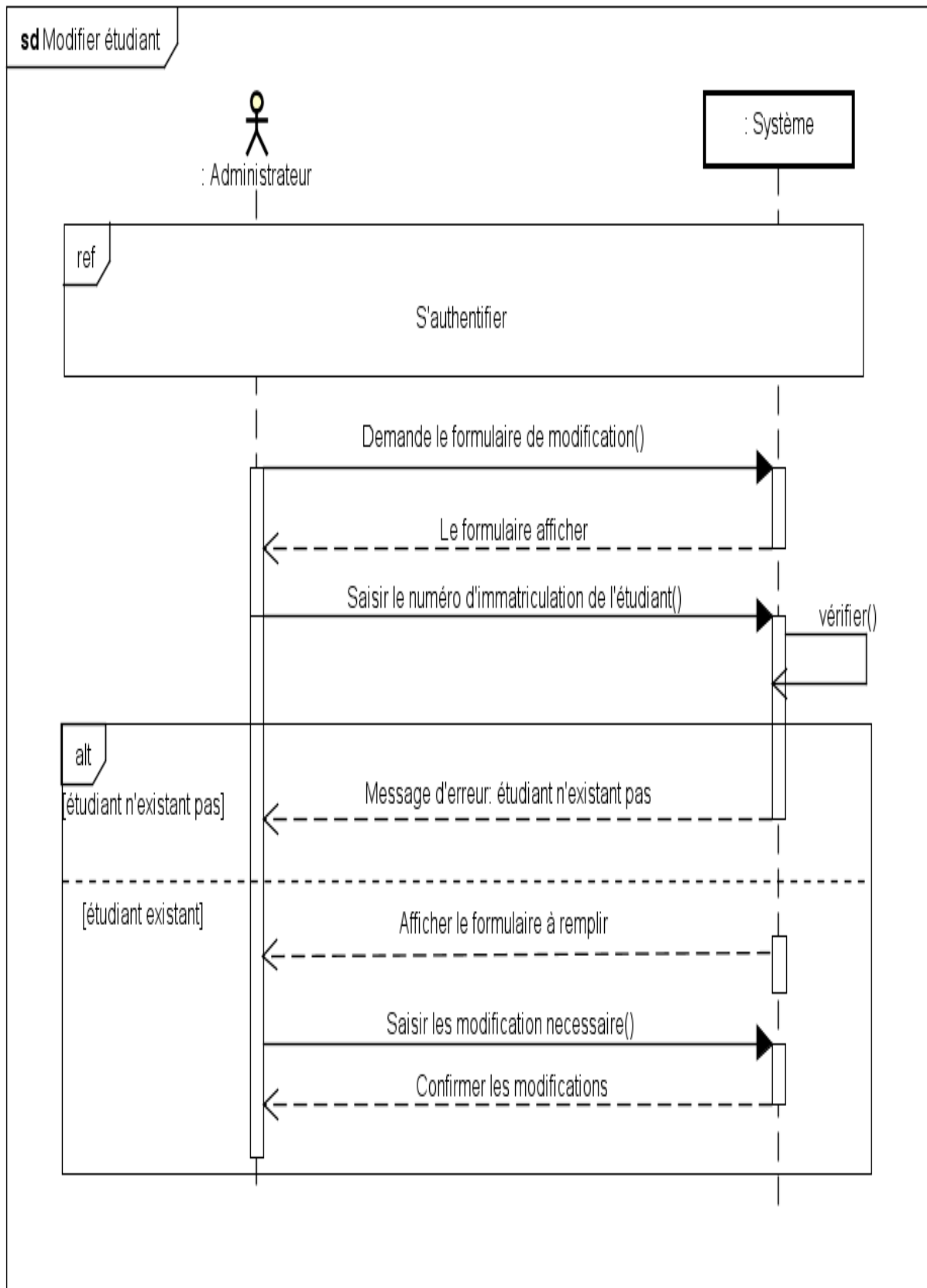


Figure II.11: Diagramme séquence système liée au cas d'utilisation "Modifier un étudiant"

D. Diagramme de séquence supprimer un dossier étudiant

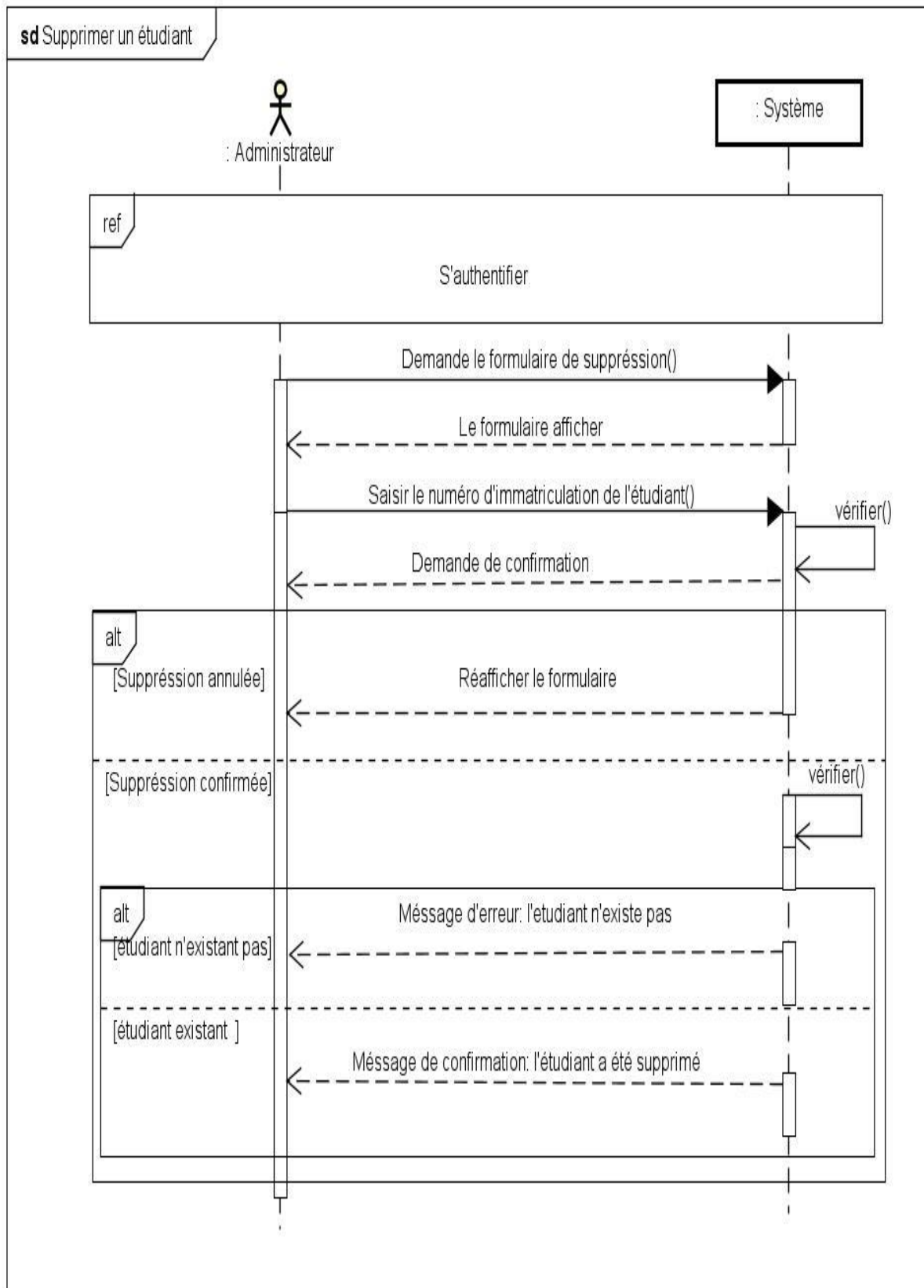


Figure II.12: Diagramme séquence système liée au cas d'utilisation "Supprimer un étudiant"

E. Diagramme de séquence consulter un étudiant

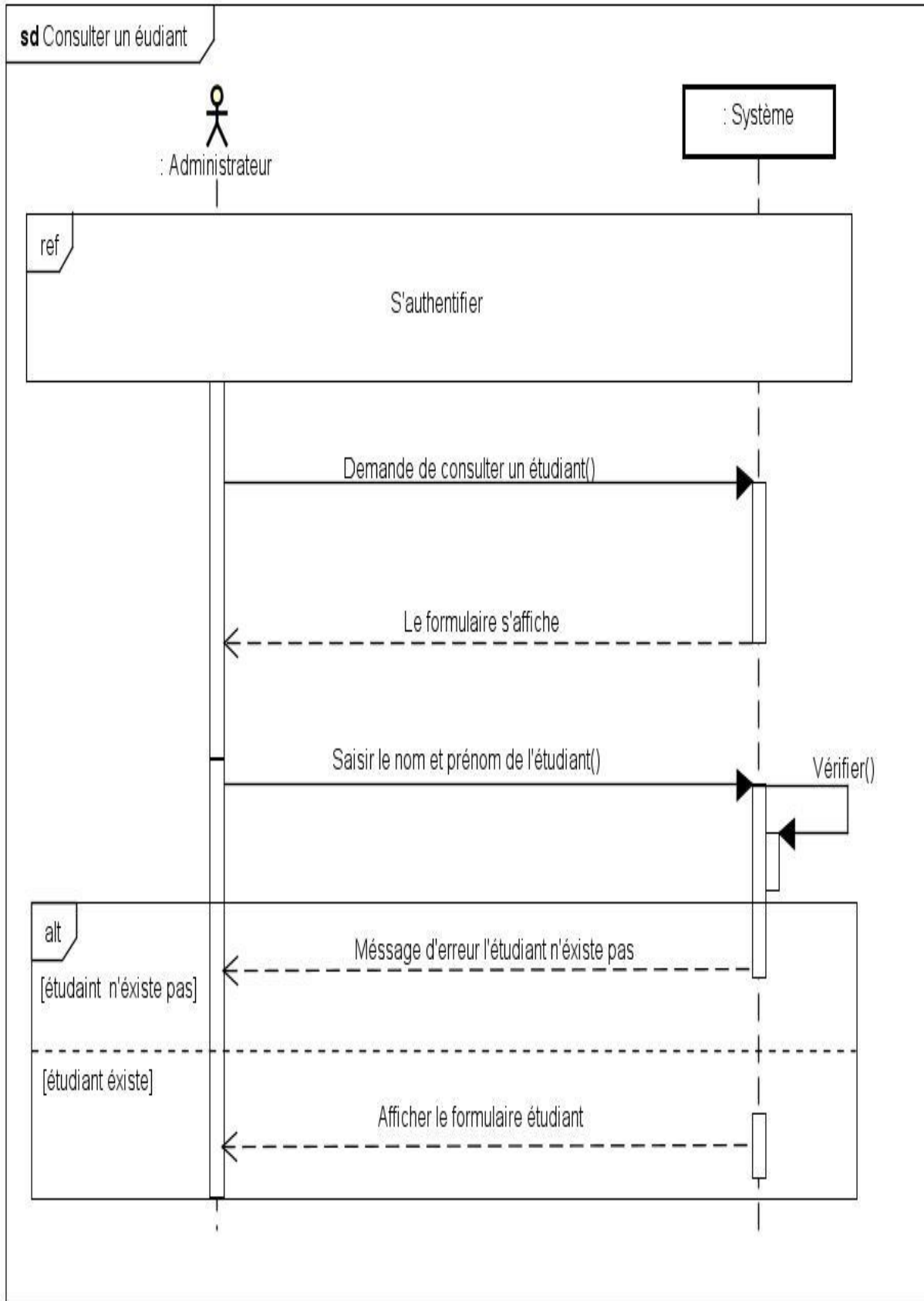


Figure II.13: Diagramme séquence système liée au cas d'utilisation "Consulter un étudiant"

F. Diagramme de séquence imprimer un rapport

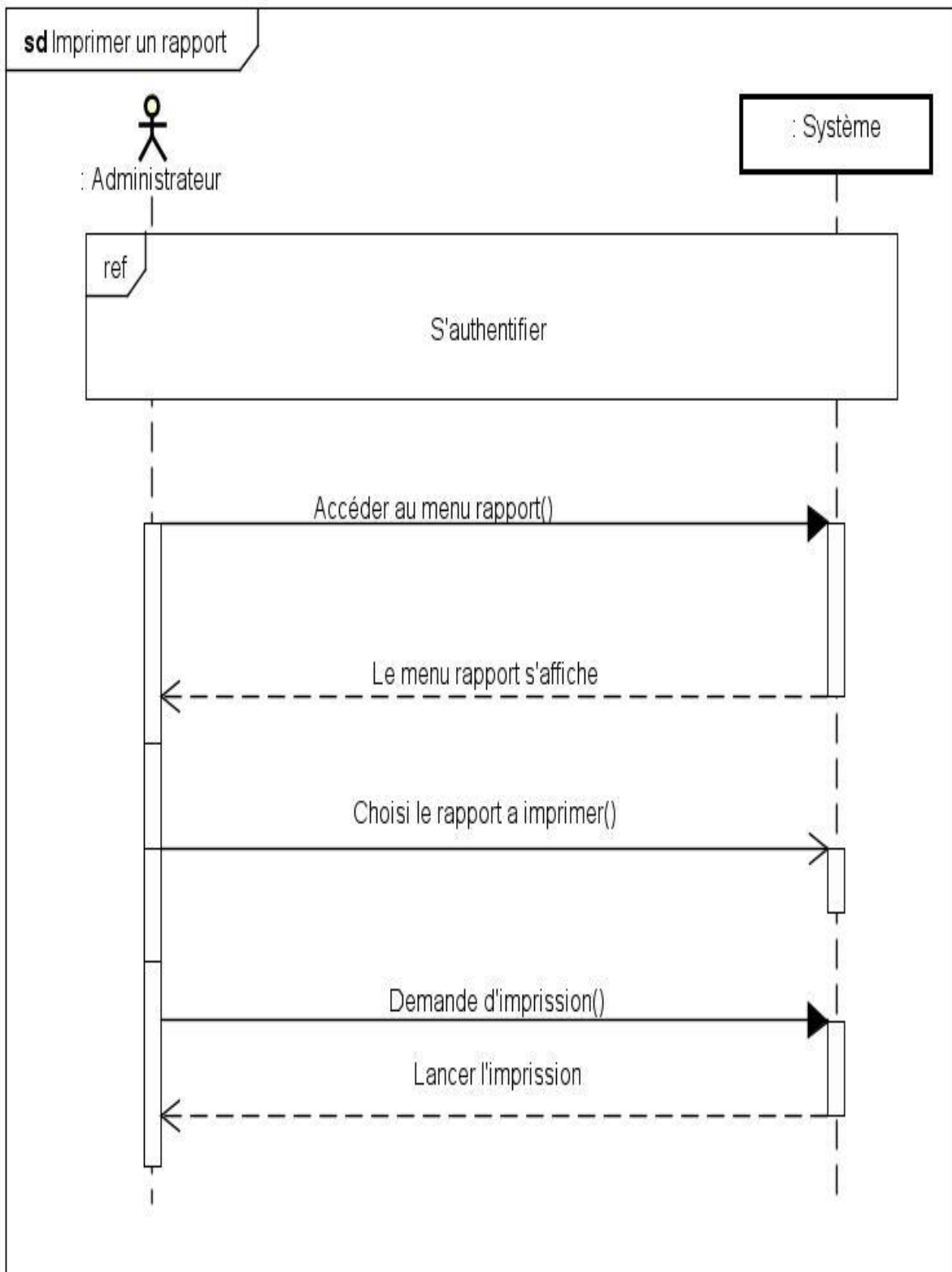


Figure II.14: Diagramme séquence système liée au cas d'utilisation "Imprimer un rapport"

Conclusion

Dans la première partie de ce chapitre, j'ai décrit le contexte de mon projet puis j'ai posé la problématique et tracé quelques objectifs.

Dans la deuxième partie, je me suis intéressé à analyser les besoins de mon application et à spécifier les différents cas d'utilisations en abordant le diagramme des cas d'utilisations et les diagrammes des séquences.

Dans ce qui suit j'entamerai la conception en présentant les diagrammes des interactions et le diagramme des classes.

III.1 introduction

Dans le cycle de vie de mon projet, la conception représente une phase primordiale et déterminante pour produire une application de haute qualité.

Dans ce qui suit je présente les diagrammes d'interactions, puis je passerons à la description de diagramme des classes, et je construire le modèle relationnel à partir de diagramme des classes de l'application, et sa en suivant les règles du passage au relationnel.

III.2 Diagramme d'interaction

Le diagramme d'interaction est un diagramme de séquence détaillé.

Je présente dans cette section les diagrammes d'interactions, en remplaçant le système représenté par une boîte noire dans le diagramme de séquences par les trois types de classes suivantes :

- L'objet interface (dialogue)(<D>) : représente l'interface entre l'acteur et le système,
- L'objet contrôle (<Ctrl>) :représente un traitement du système déclenché par un acteur,
- L'objet entité (<E>) :représente des objets décrits dans le cas d'utilisation.

Je respecte également les règles que je vais fixées sur les relations entre les classes d'analyses, mais je m'intéresse cette fois-ci aux interactions dynamiques entre objets :

- Les acteurs ne peuvent interagir (envoyer des messages) qu'avec les dialogues ;
- Les dialogues peuvent interagir avec les contrôles ;
- Les contrôles peuvent interagir avec les dialogues, les entités, ou d'autres contrôles;
- Les entités ne peuvent interagir qu'entre elles.

Les figures (III.1) jusqu'_a (III.6) présenterons respectivement les diagrammes d'interactions

A. Diagramme d'interaction authentification d'un utilisateur

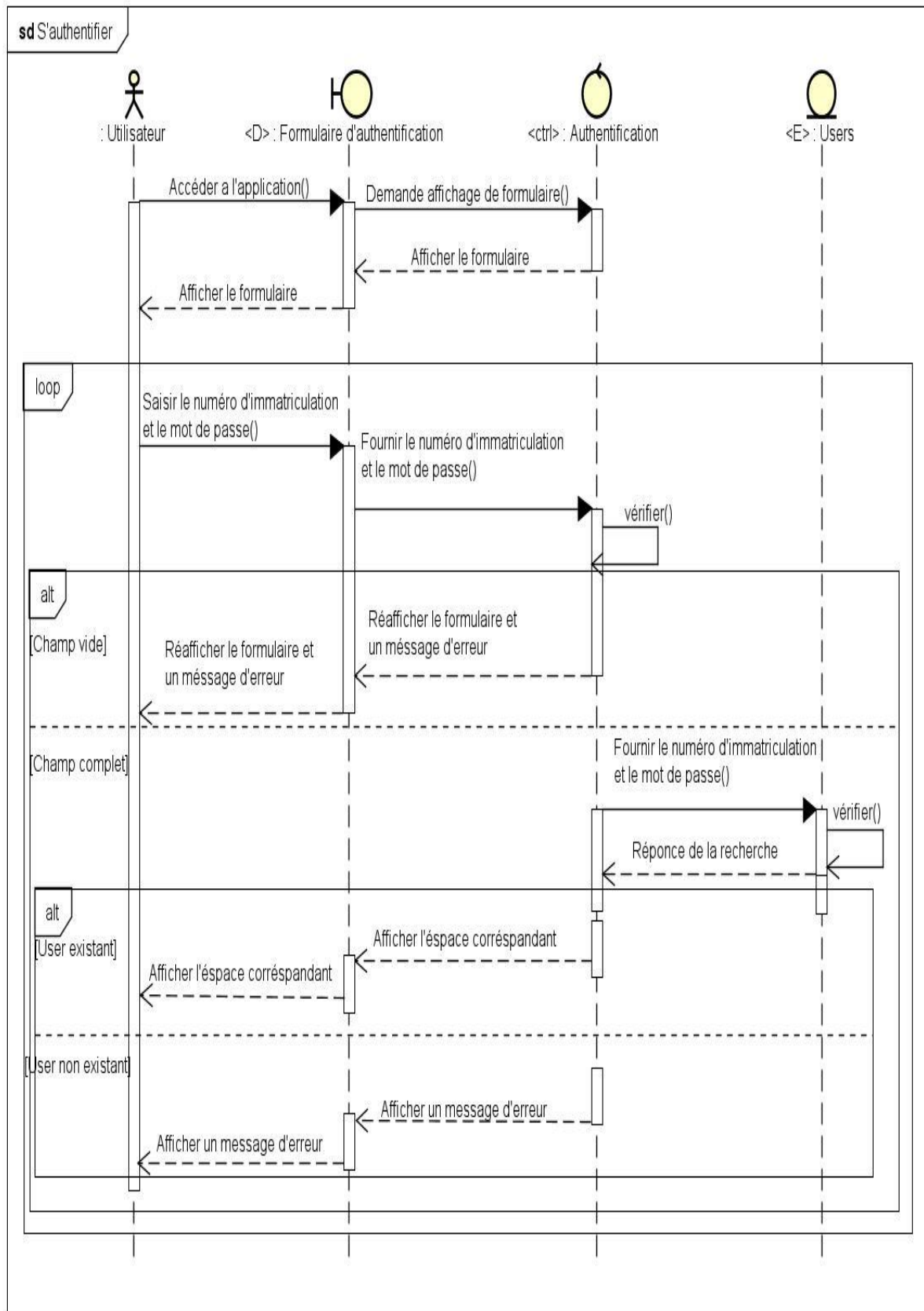


Figure III.1: Diagramme d'interaction authentification d'un utilisateur

B. Diagramme d'interaction ajouter un rapport

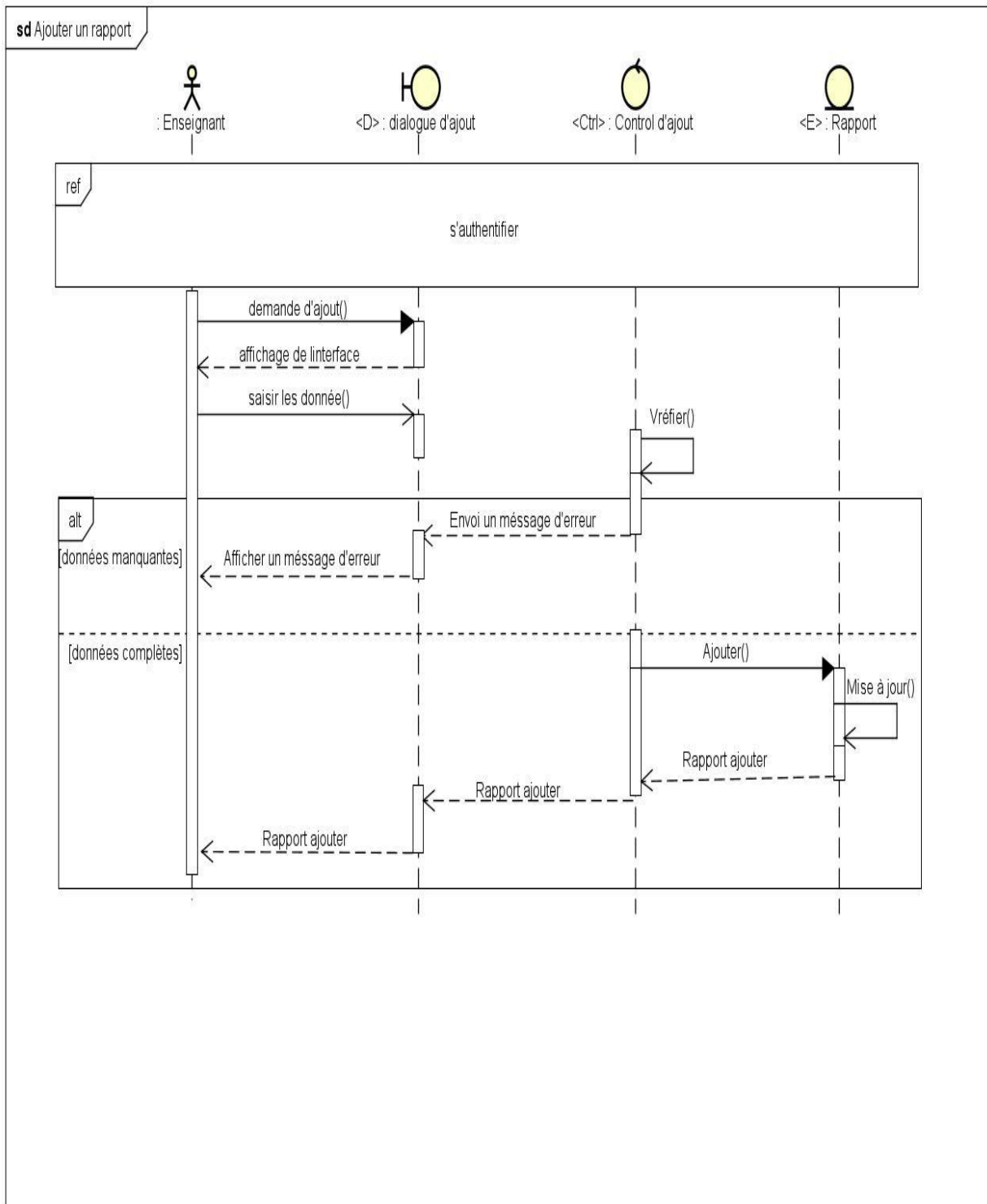


Figure III.2: Diagramme d'interaction ajouter un rapport

C. Diagramme d'interaction modifier un rapport

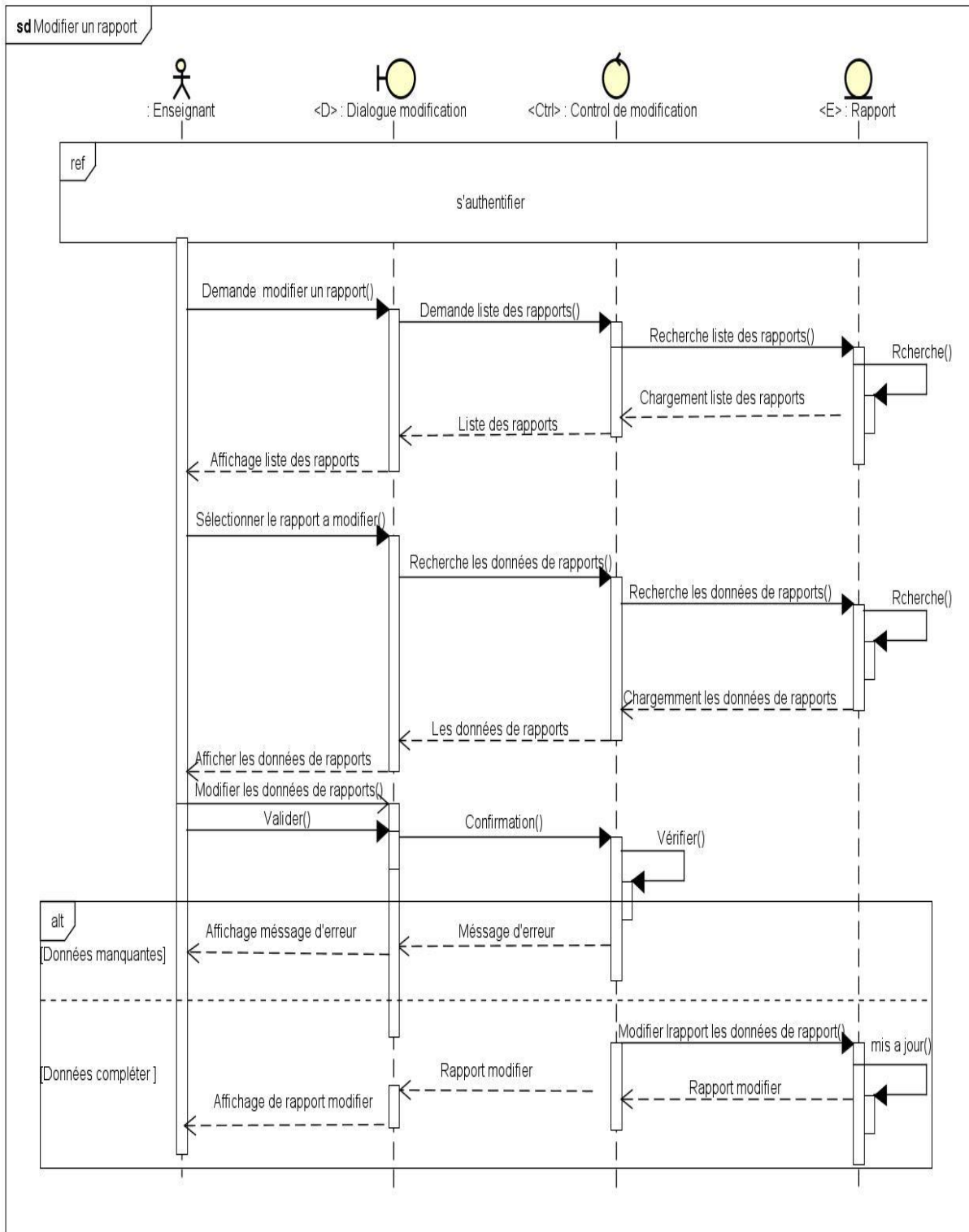


Figure III.3: Diagramme d'interaction modifier un rapport

D. Diagramme d'interaction supprimer un rapport

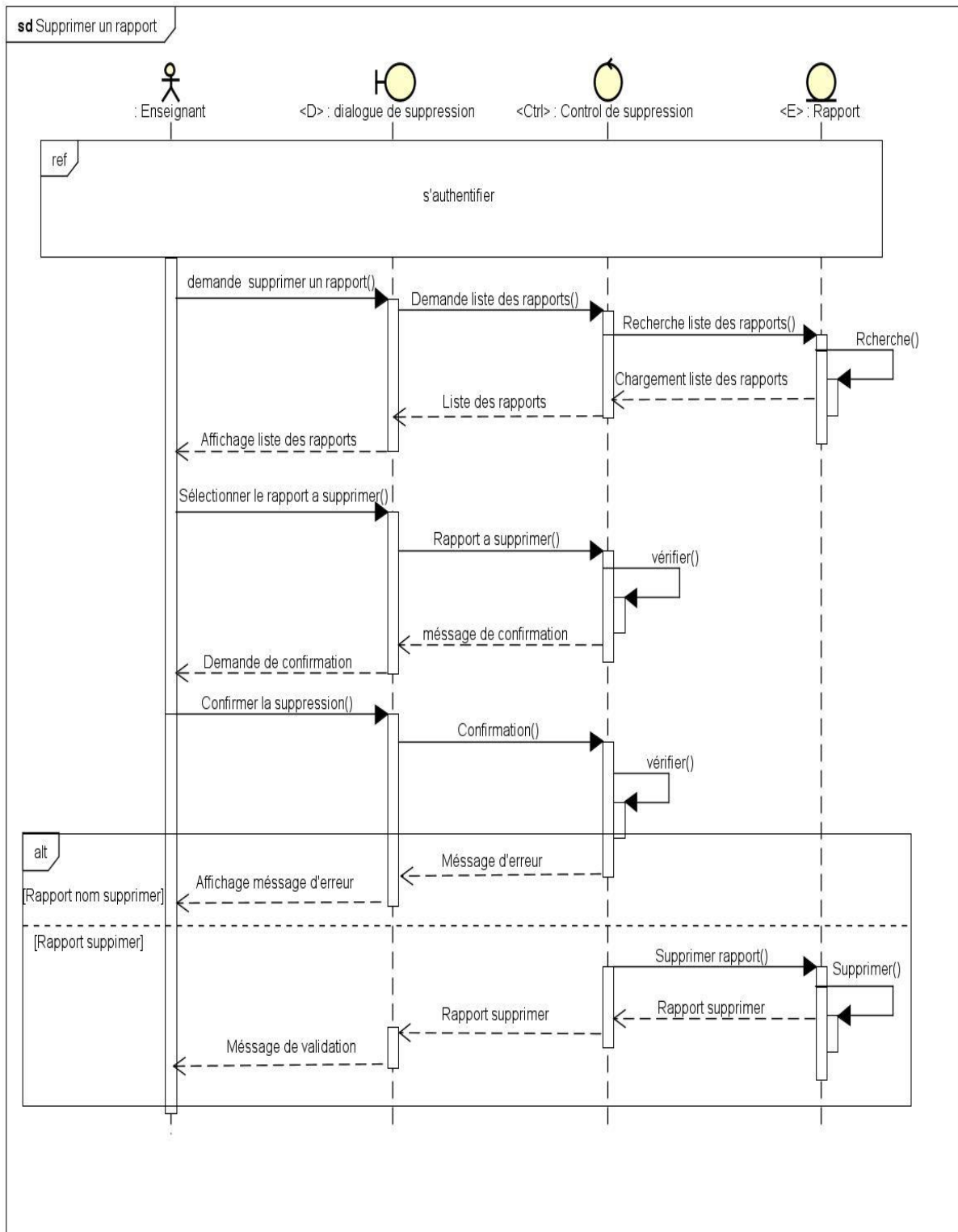


Figure III.4: Diagramme d'interaction supprimer un rapport

E. Diagramme d'interaction imprimer un rapport

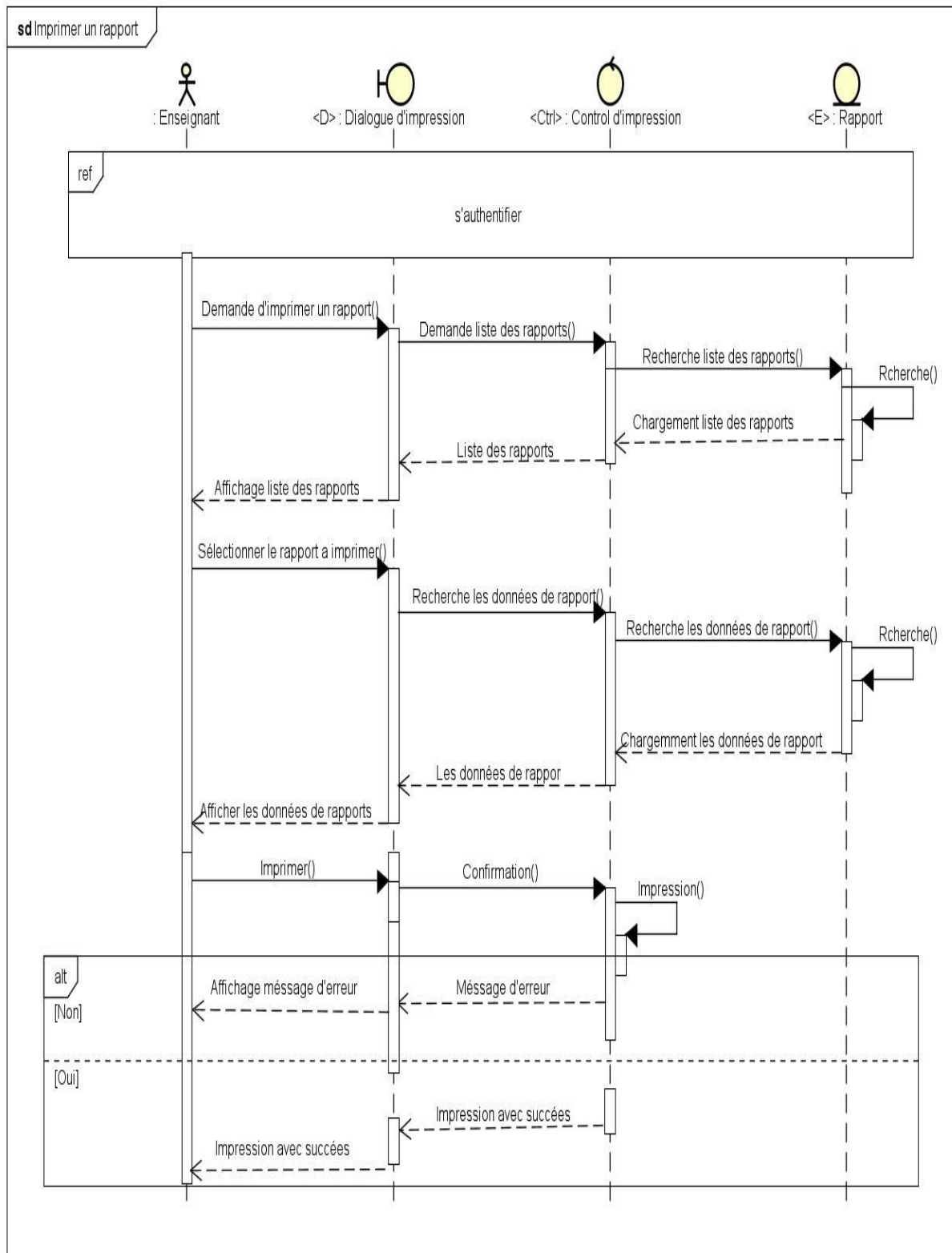


Figure III.5: Diagramme d'interaction imprimer un rapport

III.3 Diagramme des classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que leurs relations. Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux aspects temporels et dynamiques.[10]

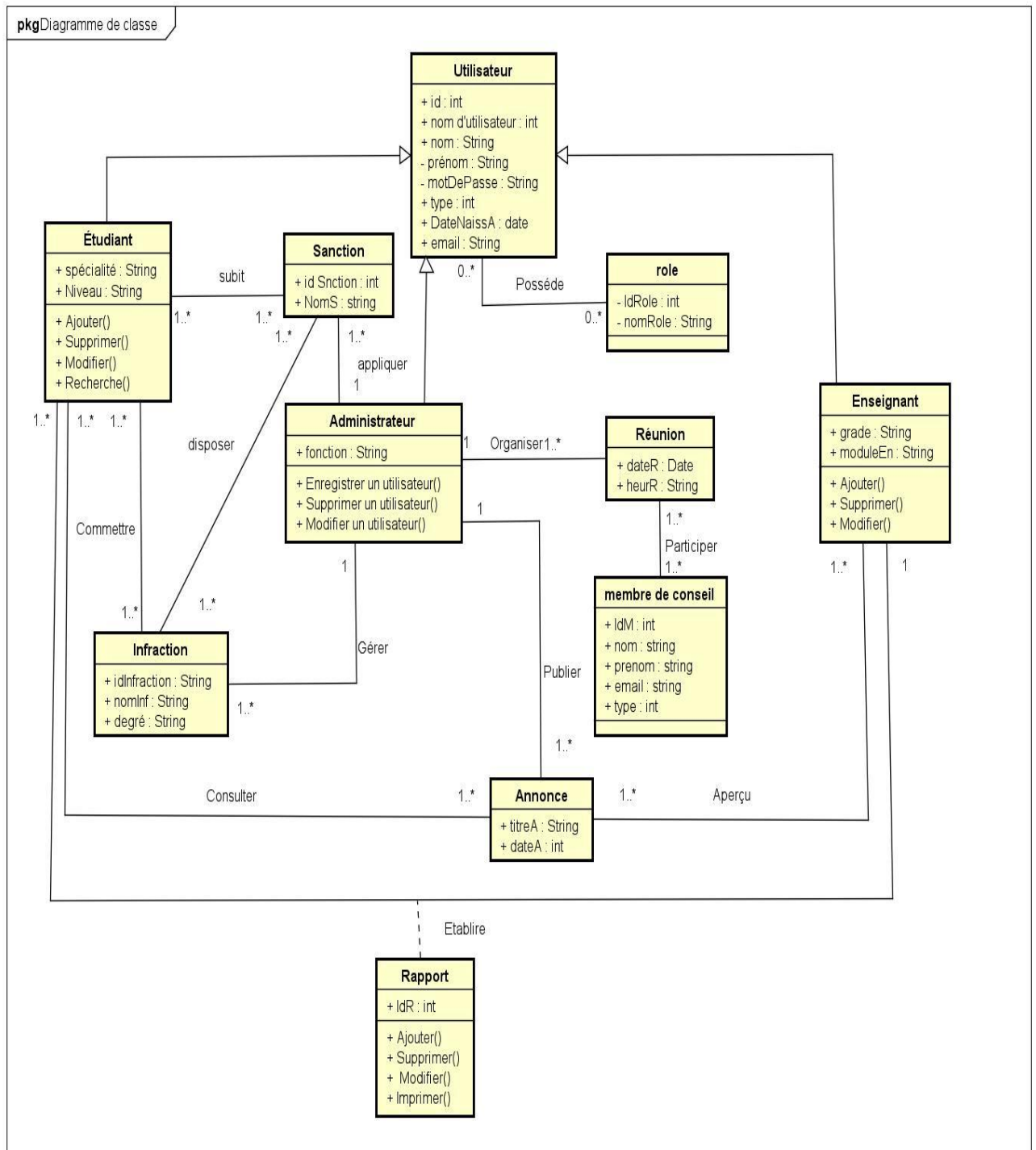


Figure III.6: Diagramme des classes

III.3.1 Dictionnaire de données :

Classe	Attributs	Signification	Type	Taille
Utilisateur	-id -nom d'utilisateur -nom - prénom -DateNaissA -email - motDePasse -Type	-Identifiant de l'utilisateur -nom d'utilisateur -nom -prénom -Date de naissance de l'utilisateur -email d'utilisateur - Mot de passe de l'utilisateur -Administrateur/Enseignant/Étudiant	-Entier -Entier -Caractère -Caractère -Date -Caractère -Caractère -Caractère	40
Administrateur	- fonction	- Fonction de l'administrateur	-Caractère	20
Enseignant	-grade - moduleEn	-Le grade de l'enseignant - Les modules enseignés par Enseignant	-Caractère -Caractère	20
Etudiant	- spécialité - Niveau	- Spécialité de l'étudiant - Niveau d'étude	-Caractère -Caractère	20
Infraction	-idInfraction -nomInf - degré	-Identifiant de l'infraction -nom de l'infraction -1er/2eme/3eme degré	-Entier -Caractère -Caractère	20
Annonce	-titreA -dateA	-Titre de l'annonce - Date de parution de l'annonce	-Caractère -Date	20
Rapport	-IdR	-Identifiant de Rapport	-Entier	20
Réunion	-dateR -heurR	- date de la réunion -heur de la réunion	-Date -Caractère	10
Sanction	-id Sanction - NomS	-Identifiant de la Sanction -Nom de la Sanction	-Caractère -Caractère	20

Tableau 3.1: Représentatif du dictionnaire de données.

III.4 Le modèle relationnel :

Le concepteur d'une base de données relationnelle doit élaborer un schéma relationnel de la base de données.

Cette activité consiste à définir toutes les relations de la base de données et leurs attributs.

III.4.1 Règles de passage au modèle relationnel:

Les règles utilisées pour le passage du diagramme de classes de mon application web au modèle Relationnel sont les suivantes :

- Toute entité devient une relation ayant pour clé primaire son identifiant [11].
- Chaque propriété se transforme en attribut [11].
- Toute association non hiérarchique (de type [n, n] ou de dimension > 2) devient une relation. La clé primaire est formée par la concaténation (juxtaposition) de l'ensemble des identifiants des entités reliées. Toutes les propriétés éventuelles deviennent des attributs qui ne peuvent pas faire partie de la clé [11].

III.1 Modèle relationnel:

Après avoir appliqué les règles de passage cité précédemment, je suis abouti au schéma relationnel de la base de données suivant :

Utilisateur (id, nom d'utilisateur, nom, prénom, DateNaissA, _motDePasse, email, Type);
Administrateur (#id, #nom d'utilisateur, # nom, #prénom, #DateNaissA, #motDePasse, #email, fonction);
Enseignant (#id, #nom d'utilisateur, #nom, #prénom , #DateNaissA , #motDePasse, #email, grade, moduleEn);
Etudiant ((#id, #nom d'utilisateur, #nom, #prénom, DateNaissA, #motDePasse, #email, spécialité, Niveau);
Infraction (idInfraction, nomInf, degré, #nom d'utilisateur, #id Sanction);
Annonce(titreA, dateA, #id);
Rapport(IdR, #id);
Réunion(#id, dateR, heurR);
Sanction(id Sanction, NomS, #id);
membre(#IdM, nom , prénom, email);

Conclusion

Après avoir abordé les différents diagrammes, j'ai pu déterminer bien précisément le fonctionnement de mon application web et mettre en évidence l'interaction entre nos acteurs et l'application web.

En effet l'objectif de cette analyse et conception, c'est de concrétiser le système d'information du Conseil de Discipline du Département d'Informatique par la phase implémentation que je vais aborder dans le chapitre qui va suivre.

IV.1 Introduction

Après analyse et conception de mon application, je entame la phase de réalisation de cette application qui utilise des technologies spécifiques à ces besoins.

Dans ce chapitre je présenterai en premier lieu l'environnement de développement et les langages utilisés, je passerai par la suite a la présentation de différentes interfaces de mon application.

IV.2 Environnement et outils de développement

L'environnement de développement de mon application est basée sur : le système de gestion de base de données relationnelles (SGBDR) SQLite, langage de programmation laravel livewire, ainsi que l'outil de modélisation UML Astah UML. Dans ce qui suit je donnerai une petite description de ces derniers.

IV.2.1 Outils de développement

IV.2.1.1 Laravel

Laravel est un puissant Framework PHP créé par Taylor Otwell, basé sur le modèle de conception MVC, conçu pour les développeurs qui ont besoin d'une boîte à outils simple et élégante pour créer des applications web complètes. [13]

IV.2.1.2 Livewire

Livewire est un Framework full-stack pour Laravel qui simplifie la création d'interfaces dynamiques, sans quitter le confort de Laravel.[14]

IV.2.1.3 Visual Studio code

Visual Studio Code (VSC par la suite) est un éditeur de code open-source, gratuit et multiplateforme (Windows, Mac et Linux), développé par Microsoft, à ne pas confondre avec Visual Studio, l'IDE propriétaire de Microsoft. VSC est développé avec Electron et exploite des fonctionnalités d'édition avancées du projet Monaco Editor. Principalement conçu pour le développement d'application avec JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.[12]

IV.2.1.4 PHP

Le PHP, pour Hypertext Preprocessor, désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques. Il s'agit d'un langage de programmation sous licence libre qui peut donc être utilisé par n'importe qui de façon totalement gratuite.

Créé au début des années 1990 par le Canadien et Groenlandais Rasmus Lerdorf, le langage PHP est souvent associé au serveur de base de données MySQL et au serveur Apache. Avec le système d'exploitation Linux, il fait partie intégrante de la suite de logiciels libres LAMP.

Sur un plan technique, le PHP s'utilise la plupart du temps côté serveur. Il génère du code HTML, CSS ou encore XHTML, des données (en PNG, JPG, etc.) ou encore des fichiers PDF. Il fait, depuis de nombreuses années, l'objet d'un développement spécifique et jouit aujourd'hui une bonne réputation en matière de fiabilité et de performances.[15]

IV.2.1.5 Tailwind CSS

Tailwind est un framework CSS créé par Adam Wathan . Contrairement à d'autres frameworks, il n'est pas fourni avec des classes à ajouter aux balises HTML. Au lieu de cela, il utilise une approche différente. Il apporte un niveau de contrôle beaucoup plus faible en utilisant des classes basées sur des utilitaires.[17].

IV.2.1.6 Astah UML

Astah UML est un outil de création de diagrammes UML léger. Avec l'intégration du code source et une bibliothèque robuste de plug-ins.

Astah UML est parfait pour ceux qui commencent à créer des diagrammes avec des diagrammes UML. [16]

IV.2.1.7 MySQL

MySQL est un système de gestion de base de données relationnelle open source (SGBDR) basé sur le langage de requête structuré (SQL). MySQL fonctionne sur pratiquement toutes les plateformes, y compris Linux , UNIX et Windows . Bien qu'il puisse être utilisé dans un large éventail d'applications, MySQL est le plus souvent associé aux applications Web et à la publication en lignedes cellules.[15]

IV.2.1.8 XAMPP

xampp est un ensemble de logiciels servant à mettre en place aisément un serveur Web, un serveur FTP et un serveur de messagerie électronique. C'est une distribution de logiciels libres (X Apache MySQL Perl PHP) offrant une bonne souplesse d'utilisation, reconnue pour son installation simple et rapide. Ainsi, il est à la portée de la plupart de personnes dans la mesure où il ne requiert pas de connaissances spécifiques et fonctionne, qui plus est , sur les dispositifs d'exploitation les plus communs.[18]

V.3 Ergonomie

IV.3.1 Logo

Étant Donné l'université de Bejaia déjà un logo personnalisé Significatif, je l'ai utilisé et j'ai pris le soin de le mettre en évidence sur la partie gauche de chaque interface de l'application. (Figure IV.1)



Figure IV.1: Logo de l'université de Bejaia

IV.3.2 Interfaces de l'application

Dans ce qui suit, je vais vous présenter quelques interfaces de l'application de conseil discipline de département informatique de l'université de Bejaia

IV.3.2.1 Interface d'authentification

La **figure IV.2** représente l'interface d'authentification qui permet à l'utilisateur de s'authentifier.

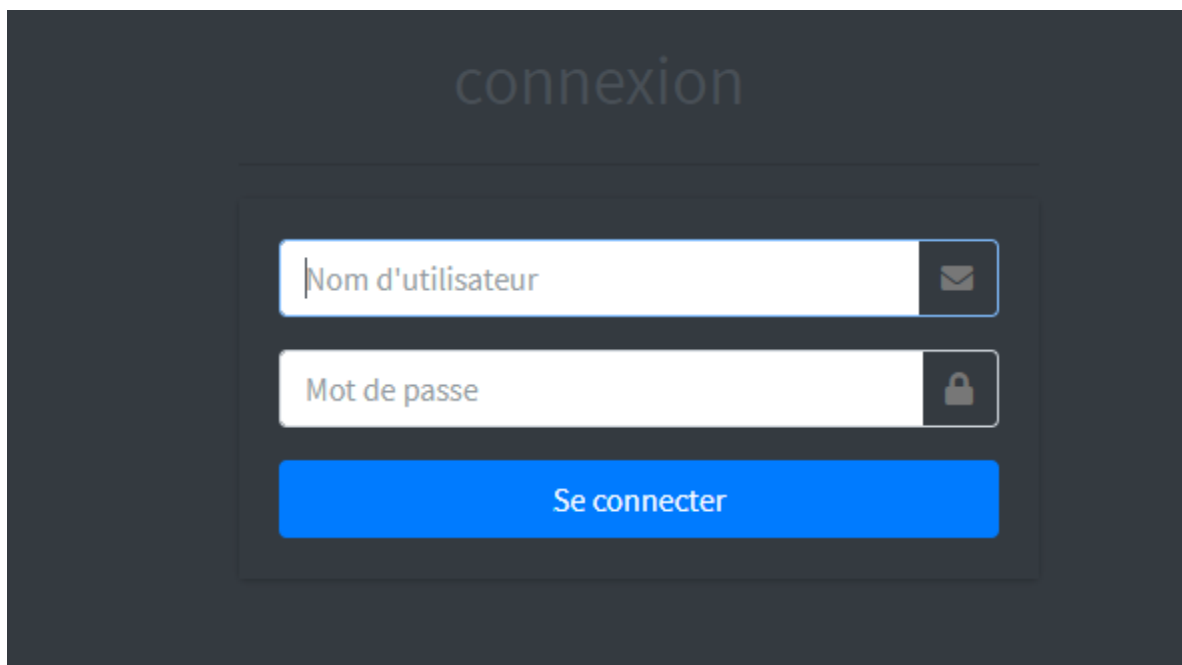


Figure IV.2: L'interface d'authentification.

IV.3.2.2 Page d'Accueil

Cette page permet à l'utilisateur d'accéder aux différentes fonctionnalités de l'application Gestion de conseil.

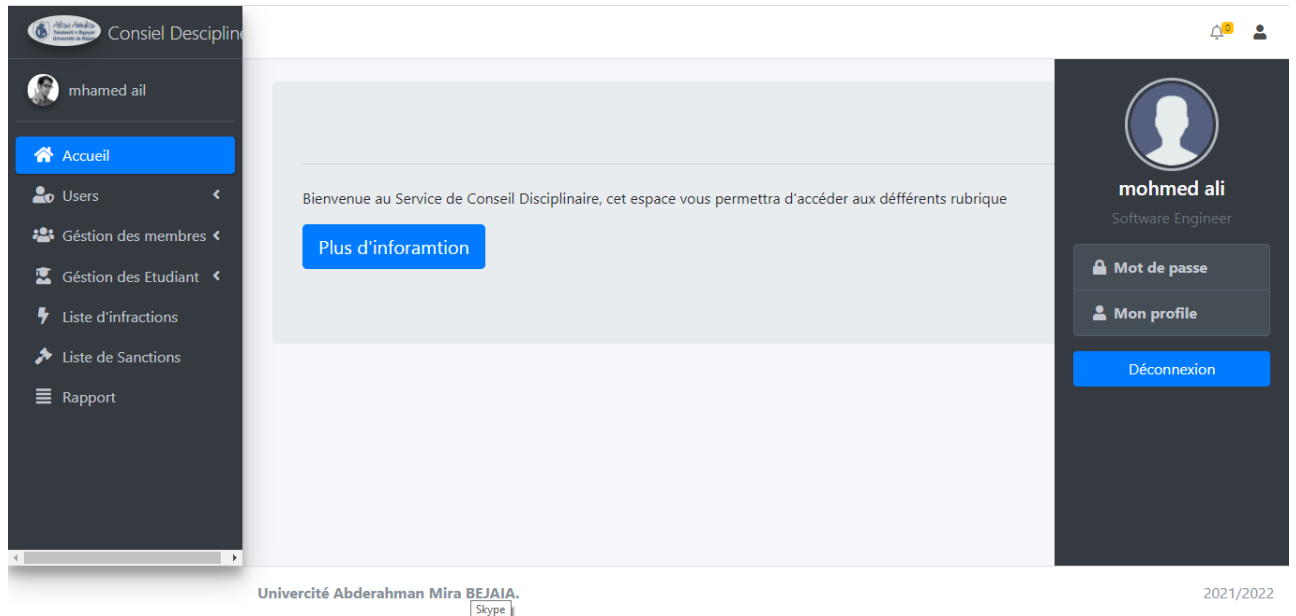


Figure IV.3: Page d'accueil

IV.3.2.3 Page de gestion des utilisateurs

Cette page permet à l'administrateur de gérer les différents utilisateurs de l'application.

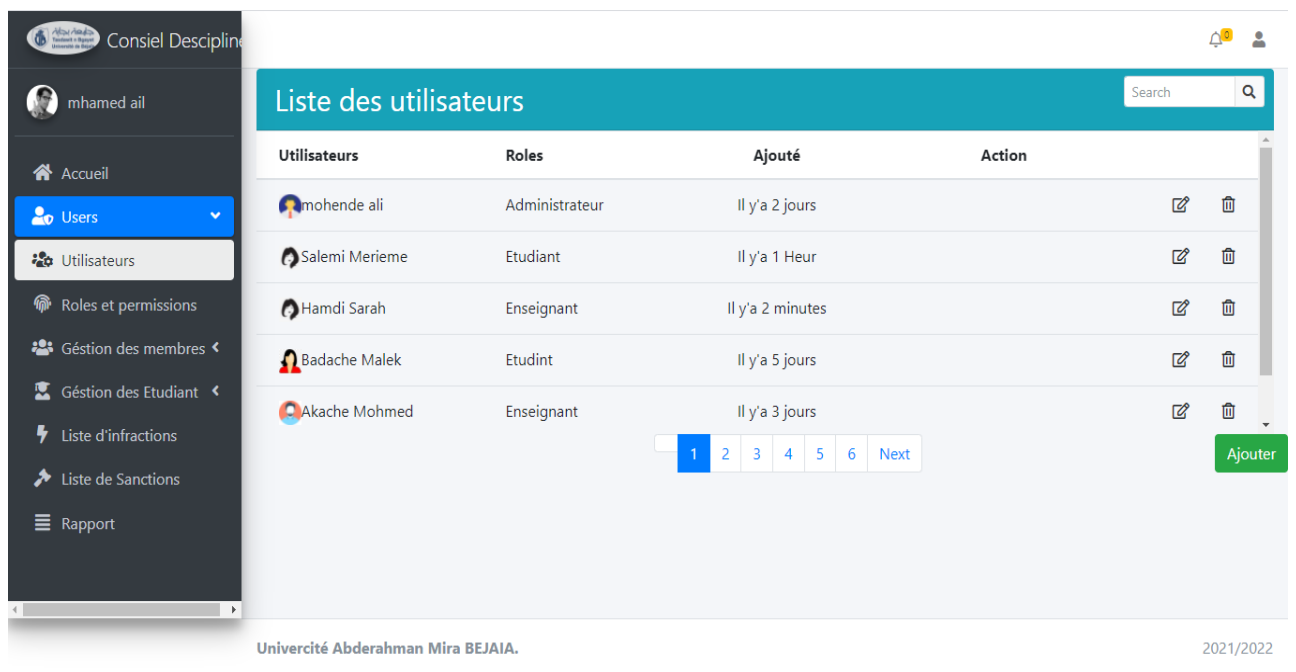


Figure IV.4: Page gestion des utilisateurs

IV.2.3.4 Page de gestion des étudiants

Cette page permet à l'administrateur de gérer les étudiants.

Etudiant	Date de naissance	Nom d'utilisateur	Spécialité	Niveux	Infraction	Décision
Hamadeche Arezki	12-10-1995	161633225250	GL	M1	tentative de triche	0 dans le module
Badache Hafit	05-03-1994	161633223131	ASR	M2		
Harone Mssilia	20-05-1999	161633225258	Informatique	L1	Détruire les meubles publique	Avertissement
Keddad milisa	06-03-1999	161633225274	MI	L2		
Chawech Aris	16-03-1993	161633225246	IA	M2	Insulte un personnel universitaire	Avertissement

Figure IV.5: Page de gestion des étudiants

IV.2.3.5 Capture d'écran d'une annonce

L'administrateur peut publier les nouvelles annonces dans cette espace

Bienvenue au Service de Conseil Disciplinaire, cet espace vous permettra d'accéder aux différents rubrique

Aujourd'hui se déroulera une réunion de conseil discipline au niveau de département à 13:00h

Objet:
tentative de triche d'un étudiant M1 GL

[Plus d'information](#)

Figure IV.6: Capture d'écran d'une annonce

Conclusion Générale

L'objectif de mon projet de fin d'étude était de concevoir et développer une application web permettant la gestion automatisée du conseil de discipline du département informatique de la faculté des sciences exactes (université Abderrahmane mira Bejaia).

Pour mener à bien ma mission, j'ai mis en œuvre toutes mes connaissances acquises durant notre cursus universitaire, ajouté aux précieuses informations que j'ai eu au sein du département informatique de la faculté des sciences exactes (université Abderrahmane mira Bejaia).

j'ai fondé une bonne analyse de l'existant suivie d'une conception minutieuse de ma base de données et de mes interfaces, ce qui m'a permis de capturer tous les besoins du service administratif du département afin de réaliser le travail demandé.

En outre, ce projet m'a offert la possibilité de revoir et de suivre avec beaucoup d'attention les méthodes fondamentales de la modélisation et de la conception des systèmes d'informations grâce auxquelles l'application a été mise au point.

j'ai la certitude que le nouveau système développé sera bénéfique et apportera des améliorations aux membres du conseil de discipline du département.

je suis d'autant plus satisfait que la réalisation de ce projet représente une opportunité pour moi de découvrir le monde professionnel, ses réalités, ses difficultés et ses engagements imposés.

Enfin, j'espère que ce modeste travail servira de référence et de base à tous ceux qui voudront s'engager dans un travail homologue.

Bibliographie:

1. glossaire. *syloe.com*. [En ligne] 22 05 2021. [Citation : 26 05 2006.] <https://www.syloe.com/glossaire/systeme-dinformation/>.
2. digitalguide. *ionos.fr/*. [En ligne] [Citation : 26 05 2021.] <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/uml-un-langage-de-modelisation-pour-la-programmation-orientee-objet/>.
3. **roques, pascal**. *les cahiers du programmeur-UML2-Modéliser une application web*. s.l. : eyrolles, 2006. 4^{ème} édition.
4. [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique))
5. [https://fr.wiktionary.org/wiki/conseil_de_discipline\(14/08/2021\)](https://fr.wiktionary.org/wiki/conseil_de_discipline(14/08/2021))
6. https://www.univ-usto.dz/wp-content/uploads/2021/05/REGLEMENT_INTERIEUR_DE_LUSTO_Avril_2016-1.pdf (13/08/2021).
7. <http://prive.iutenligne.net/iuRxM0CThIXDIjzG/informatique/langages/kettaf/UML/04modeleconceptuel/0302acteurs.html>(18/08/2021)
8. [https://fr.wikipedia.org/wiki/Diagramme_de_profils\(18/08/2021\)](https://fr.wikipedia.org/wiki/Diagramme_de_profils(18/08/2021))
9. J.CONALLEN. *Concevoir des applications web avec UML*. Eyrolles, 2000.
10. https://fr.wikipedia.org/wiki/Diagramme_de_classes (03/08/2021)
11. <http://stephanie.laporte.pagesperso-orange.fr/Pdf/passageMCDMR.pdf> Consulter (20/08/2021).
12. https://edutechwiki.unige.ch/fr/Visual_studio_code (03/08/2021)
13. didaquest. *didaquest.org*. [En ligne] 18 05 2021. [Citation : 24 5 2021.] <https://didaquest.org/wiki/Laravel>.
14. livewire. *laravel-livewire*. [En ligne] [Citation : 21 05 2021.] <https://laravel-livewire.com/>
15. [https://searchoracle.techtarget.com/definition/MySQL\(24/05/2021\)](https://searchoracle.techtarget.com/definition/MySQL(24/05/2021))
16. [https://astah.net/products/free-student-license/\(05/06/2021\)](https://astah.net/products/free-student-license/(05/06/2021))
17. [https://www.section.io/engineering-education/introduction-to-tailwind-css/\(07/09/2021\)](https://www.section.io/engineering-education/introduction-to-tailwind-css/(07/09/2021))
18. [http://www.standard-du-web.com/xampp.php\(24/05/2021\)](http://www.standard-du-web.com/xampp.php(24/05/2021))