

Ministère de l'Enseignement Supérieur  
Et de la Recherche Scientifique  
Université de la Manouba  
Ecole nationale des Sciences de l'informatique



# RAPPORT DE PROJET DE CONCEPTION ET DE DEVELOPPEMENT

Sarra Ben Abdelmalek

Oussama Henia

---

**Mise en Place d'un workflow de  
gestion des processus  
administratifs de l'ENSI**

---

Encadrant académique : Mme. Leila Ben Ayed

Année universitaire : 2020-2021

J'autorise l'étudiant à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant académique, Leila Ben Ayed

**Signature**

**Mon père,** A qui je dois toute ma réussite. C'est grâce à toi que j'ai pu faire mes premiers pas dans la vie et réaliser mes ambitions. Ta pureté, ta droiture et ton honnêteté me serviront de modèle. Ce travail est le fruit de tes innombrables sacrifices. J'en serai éternellement reconnaissante.

**Maman,** Tu t'es toujours sacrifiée pour nous. Ton affection et ta tendresse n'ont d'égal que mon amour et mon estime pour toi. Puisse ce modeste travail témoigner de l'admiration que j'ai toujours eue pour toi.

**Ma sœur, mon frère et mes amis,** Je vous ai toujours trouvés proches de moi pour m'écouter, me soutenir et me conseiller. J'en suis très reconnaissante. Que la vie ne vous réserve que de la réussite.

**Sarra Ben Abdelmalek**

J'écris ces mots avec amour pour dédier aujourd'hui ma réussite à :

**Mon père**, tous les mots ne sauraient exprimer la gratitude, l'amour, le respect et la reconnaissance. Jamais je ne saurais m'exprimer quant aux sacrifices et aux dévouements que vous me consacrez.

**Maman**, ma vie, ma source de bonheur et d'affection, pour le temps sacrifié pour moi durant toutes ces années, Que tu sois toujours fière de ton petit fils.

**Ma grande famille**, je vous remercie pour l'estime et l'affection que vous m'avez accordé.

**Mes chers amis**, que j'aime et qui m'étaient une source de force et avec qui j'ai tout partagé ces dernières années.

Je tiens à vous remercier tous sans exception pour vos encouragements et pour votre précieuse aide qui m'a toujours placée sur la bonne route.

**Oussama Henia**

## REMERCIEMENTS

Au terme de ce travail, nous adressons nos vifs remerciements à **Mme Leila Ben Ayed**, notre encadrante pédagogique pour son suivi, sa disponibilité, son aide précieuse et ses conseils qui nous ont été d'une utilité indéniable.

Nous exprimons également toute notre gratitude à tous nos enseignants de l'ENSI pour la formation qu'ils nous ont fourni. Nous témoignons ici à tous les membres du jury, toute notre reconnaissance et notre respect pour eux d'avoir accepté d'évaluer notre travail.

Enfin, nous remercions ceux et celles qui ont participé de près ou de loin à l'élaboration du présent travail et principalement pour leur service et pour leur soutien moral tout au long de la préparation de ce projet de conception et de développement.

**SARRA BEN ABDELMALEK, Oussama Henia**

## TABLE DES MATIÈRES

Table des figures	ix
Liste des tableaux	xii
Introduction générale	1
<b>I Cadre général du projet</b>	<b>2</b>
Cadre général du projet	2
Introduction . . . . .	2
I Contexte général du Projet . . . . .	2
II Problématique . . . . .	2
III Travail à réaliser . . . . .	3
IV Méthodologie de travail . . . . .	3
IV.1 Comparatif des différents modèles . . . . .	3
IV.2 Choix du modèle Scrum . . . . .	6
IV.3 Processus Scrum . . . . .	6
Conclusion . . . . .	7
<b>II Sprint 0 : Planification et architecture</b>	<b>8</b>
Introduction . . . . .	8

I	Analyse et spécification des besoins . . . . .	8
I.1	Identification des acteurs . . . . .	8
I.2	Identification des besoins . . . . .	9
I.2.1	Besoins fonctionnels . . . . .	9
I.2.2	Besoins non fonctionnels . . . . .	10
I.2.3	Besoins du domaine . . . . .	10
I.3	Pilotage du projet avec Scrum . . . . .	10
I.3.1	Les fonctionnalités du backlog produit . . . . .	11
I.3.2	Planification des sprints . . . . .	13
I.4	Environnement de Travail . . . . .	13
I.4.1	Environnement matériel . . . . .	13
I.4.2	Environnement de développement . . . . .	14
I.4.3	Environnement logiciel . . . . .	14
I.4.4	Justification des choix . . . . .	15
I.5	Architecture générale de l'application . . . . .	16
I.5.1	Présentation de quelques styles architecturaux logiques . . . . .	16
I.5.2	Architecture logique . . . . .	18
I.5.3	Fonctionnement de l'architecture . . . . .	19
I.6	Diagramme de déploiement . . . . .	19
	Conclusion . . . . .	20
<b>III</b>	<b>Étude et réalisation du Sprint 1</b>	<b>21</b>
	Introduction . . . . .	21
I	Backlog du sprint . . . . .	21
II	Spécification fonctionnelle . . . . .	22
II.1	Diagrammes de cas d'utilisation . . . . .	22
II.2	Descriptions textuelles des cas d'utilisation . . . . .	26
III	Conception détaillée . . . . .	34

III.1	Patrons de conception . . . . .	34
III.2	Diagramme de classes du sprint 1 . . . . .	34
III.3	Diagrammes de séquence détaillés . . . . .	36
IV	Réalisation . . . . .	41
V	Test . . . . .	55
	Conclusion . . . . .	56
<b>IV</b>	<b>Étude et réalisation du Sprint 2</b>	<b>57</b>
	Introduction . . . . .	57
I	Backlog du sprint . . . . .	57
II	Spécification fonctionnelle . . . . .	58
II.1	Diagrammes de cas d'utilisation . . . . .	58
II.2	Descriptions textuelles des cas d'utilisation . . . . .	60
III	Conception détaillée . . . . .	63
III.1	Diagramme de classes final . . . . .	63
III.2	Modèle entité-association . . . . .	64
III.3	Diagrammes de séquence détaillés . . . . .	66
IV	Réalisation . . . . .	69
V	Tests . . . . .	73
	Conclusion . . . . .	74
	<b>Conclusion</b>	<b>75</b>



## TABLE DES FIGURES

I.1	Le processus SCRUM [1] . . . . .	7
II.1	Architecture logique de l'application [7] . . . . .	19
II.2	Diagramme de déploiement . . . . .	20
III.1	Diagramme de cas d'utilisation global du sprint 1 . . . . .	22
III.2	Diagramme de cas d'utilisation de l'opération s'authentifier . . . . .	23
III.3	Diagramme de cas d'utilisation de l'opération gérer matière . . . . .	23
III.4	Diagramme de cas d'utilisation de l'opération gérer profil étudiant . . . . .	24
III.5	Diagramme de cas d'utilisation de l'opération gérer profil enseignant . . . . .	25
III.6	Diagramme de cas d'utilisation de l'opération gérer inscriptions . . . . .	25
III.7	Diagramme de classes du sprint 1 . . . . .	35
III.8	Diagramme de séquence du cas d'utilisation s'authentifier . . . . .	36
III.9	Diagramme de séquence du cas d'utilisation ajouter matière . . . . .	37
III.10	Diagramme de séquence du cas d'utilisation modifier matière . . . . .	38
III.11	Diagramme de séquence du cas d'utilisation supprimer matière . . . . .	39
III.12	Diagramme de séquence du cas d'utilisation consulter matière . . . . .	40
III.13	Interface d'authentification . . . . .	41
III.14	Interface d'ajout de matière . . . . .	42
III.15	Interface de modification de matière . . . . .	43

III.16	Interface de consultation de matière . . . . .	44
III.17	Interface d'ajout d'un étudiant . . . . .	45
III.18	Interface de modification d'un étudiant . . . . .	46
III.19	Interface de suppression d'un étudiant . . . . .	47
III.20	Interface de consultation de la liste des étudiants . . . . .	48
III.21	Interface d'ajout d'un enseignant . . . . .	49
III.22	Interface de suppression d'un enseignant . . . . .	50
III.23	Interface de consultation de la liste des enseignants . . . . .	51
III.24	Interface d'ajout d'une inscription . . . . .	52
III.25	Interface de modification d'une inscription . . . . .	53
III.26	Interface de suppression d'une inscription . . . . .	54
III.27	Interface de consultation des inscriptions . . . . .	55
IV.1	Diagramme de cas d'utilisation global du sprint 2 . . . . .	58
IV.2	Diagramme de cas d'utilisation de l'opération gérer les emplois de temps. . . . .	59
IV.3	Diagramme de cas d'utilisation de l'opération gérer les notes. . . . .	59
IV.4	Diagramme de cas d'utilisation de l'opération gérer les absences . . . . .	60
IV.5	Diagramme de cas d'utilisation de l'opération visualiser les statistiques. . . . .	60
IV.6	Diagramme de classes global du système . . . . .	64
IV.7	Modèle entité-association . . . . .	65
IV.8	Diagramme de séquence du cas d'utilisation « Ajouter séance » . . . . .	66
IV.9	Diagramme de séquence du cas d'utilisation « Modifier séance » . . . . .	67
IV.10	Diagramme de séquence du cas d'utilisation « Supprimer séance » . . . . .	68
IV.11	Diagramme de séquence du cas d'utilisation « Consulter emploi de temps » . . . . .	69
IV.12	Interface de consultation des emploi de temps . . . . .	70
IV.13	Interface d'ajout d'une séance . . . . .	70
IV.14	Interface de modification d'une note . . . . .	71
IV.15	Interface de consultation des notes . . . . .	71

IV.16	Interface de gestion des absences . . . . .	72
IV.17	Interface de visualisation des statistiques . . . . .	72
IV.18	Structure d'un jeton JWT [9] . . . . .	79
IV.19	Requête d'obtention du jeton JWT . . . . .	80
IV.20	Réponse reçue par le serveur . . . . .	80
IV.21	Utilisation du modèle Data Transfer Object . . . . .	81

## LISTE DES TABLEAUX

II.1	Tableau du Backlog Produit . . . . .	12
III.1	Backlog du sprint 1 . . . . .	22
IV.1	Backlog du sprint 2 . . . . .	57

## INTRODUCTION GÉNÉRALE

Dans le cadre de la mondialisation et la globalisation, les établissements se dotent de divers moyens technologiques afin de gérer des flux d'informations qui leur permettent de rester à jour et gagner une part de compétitivité.

C'est dans ce contexte que plusieurs pays cherchent à automatiser leurs procédures et dans ce cadre, la plupart des universités internationales ont entamé ce domaine et ont fait une révolution technologique afin d'améliorer leur productivité et de faire face à quelques problèmes pénibles qui peuvent constituer un obstacle de progression. C'est dans ce cadre que s'inscrit notre projet qui vise à mettre en place une plateforme qui permet la gestion des services réalisés par l'administration au sein de l'ENSI.

Ce rapport est composé de plusieurs chapitres : un premier chapitre intitulé cadre général du projet qui sert à décrire le contexte du projet ainsi que la méthodologie de développement adoptée.

Ensuite, un deuxième chapitre intitulé sprint 0 est dédié à l'élaboration du Back log du produit, à la détermination des acteurs ainsi que la planification des différents sprints.

Les deux derniers chapitres qui sont le troisième et le quatrième seront dédiés à la conception, la présentation des captures d'écran relatives à la solution choisie et les tests des différents sprints.

# CHAPITRE I

## CADRE GÉNÉRAL DU PROJET

### Introduction

Dans ce chapitre, nous allons évoquer le cadre du projet. Nous présentons dans un premier lieu le contexte du projet. Puis, nous abordons la problématique, le travail à réaliser ainsi que la méthodologie de travail.

## I Contexte général du Projet

Notre projet vise à mettre en place une solution pour l'automatisation des procédures administratives associés accessibles à distance.

## II Problématique

Dans le cadre de notre démarche de suivi et d'amélioration continue au sein de notre service informatique et en étroite collaboration avec tous les autres départements de l'école, nous avons remarqué que les outils mis à disposition par l'ENSI ne répondent pas exactement aux besoins et sont dépassés.

Parmi ces problèmes on peut citer :

- Perte des données et des informations
- Problèmes d'estimation des coûts et des délais
- Une forte chance de commettre des erreurs
- Coût de mise à jour très élevé
- Absence d'un outil de gestion de documents
- Une absence de sécurité sur les informations
- L'utilisation intensive de l'outil Excel pour l'enregistrement des données des étudiants.
- La communication entre l'administration de l'ENSI et les étudiants est pratiquement difficile.

### III Travail à réaliser

L'objectif de ce travail sera d'assurer plus de fiabilité des données, d'automatiser les processus administratifs de l'ENSI et de remédier aux différents problèmes rencontrés. Cette solution permettra également de :

- Assurer une meilleure cohérence
- Gérer d'une façon complète tous les processus à savoir la gestion des étudiants, des absences, des filières, etc.
- Intégrer la plateforme au système d'information de l'ENSI.

## IV Méthodologie de travail

Dans cette partie, nous présentons quelques modèles de développement logiciel en établissant une comparaison entre eux, et nous choisissons un modèle que nous jugeons adéquat pour la conception et le développement de notre plateforme.

### IV.1 Comparatif des différents modèles

Modèle	Description	Points forts	Points faibles
Cascade	Chaque phase doit être définie précisément, c'est-à-dire, qu'elle correspond au cahier des charges, et possède une date d'échéance fixe.	Le planning est établi à l'avance et le chef de projet sait précisément ce qui va lui être livré et quand il pourra en prendre livraison	La très faible tolérance à l'erreur (les anomalies sont détectées tardivement) ce qui peut induire un coût important en cas d'anomalie.
Modèle en V	Est un modèle conceptuel de gestion de projet imaginé à la suite du problème de réactivité du modèle en cascade. Il permet, en cas d'anomalie, de limiter un retour aux étapes précédentes. Les phases de la partie montante doivent renvoyer de l'information sur les phases en vis-à-vis lorsque des défauts sont détectés, afin d'améliorer le logiciel.	Nécessite une formalisation des fonctionnalités du produit et de ce qui sera fait sur le projet. Il permet ainsi de bien réfléchir et de se poser les bonnes questions au début du projet, autant du côté client que du côté du prestataire.	Le manque de souplesse : chaque phase doit être terminée avant de passer à la suivante. Documentation importante : la formalisation de Chaque phase entraîne la production de documents en abondance voire en doublons.



RUP	Est un modèle utilisé pour les logiciels orientés objets. C'est une méthode générique, itérative et incrémentale	Itératif Spécifie le dialogue entre les différents intervenants du projet (les livrables, plannings et prototypes...) Propose des modèles de documents, et des canevas pour des projets types. Rôles bien définis, modélisation	Coûteux à personnaliser. Très axé processus, au détriment du développement (peu de place pour le code et la technologie). Lourd, largement étendu, il peut être difficile à mettre en œuvre.
SCRUM	Méthode Agile Projet décomposé en « Sprints ». Contient toutes les compétences nécessaires pour terminer le sprint. Equipe de travail composé de 5 à 9 personnes. Pas de chef projet.	Itératif et Incrémental. Facile à mettre en œuvre. La plus populaire des méthodes Agiles.	Ne couvre aucune technique d'ingénierie du logiciel.

Tableau récapitulatif de quelques modèles de développement logiciel

## IV.2 Choix du modèle Scrum

Au rugby, le terme Scrum signifie mêlée. Il exploite l'esprit et les valeurs du rugby et les adapte aux projets de développement.

Comme le pack lors d'un ballon porté au rugby, l'équipe chargée du développement travaille de façon collective, soudée vers un objectif précis. Le Scrum Master guide les membres de l'équipe, les repositionne dans la bonne direction et donne le tempo pour assurer la réussite du projet. Scrum utilise une approche itérative et incrémentale pour optimiser la prédictibilité et contrôler le risque.

Trois piliers soutiennent l'implémentation d'un contrôle empirique de processus : la transparence, l'inspection et l'adaptation.

- La transparence : requiert la définition d'un standard commun pour ces aspects afin que les observateurs partagent une compréhension commune de ce qui est observé.
- L'inspection : les utilisateurs de Scrum doivent fréquemment inspecter les artéfacts Scrum et l'état d'avancement par rapport à un objectif de Sprint (Sprint Goal) afin de détecter les écarts indésirables. La fréquence de ces inspections ne devrait pas gêner le travail en cours. Ces inspections sont bénéfiques lorsqu'elles sont effectuées de manière diligente sur les lieux du travail par les personnes qualifiées.
- L'adaptation : si un inspecteur détermine qu'un ou plusieurs aspects du processus dérivent hors des limites acceptables, et que le produit qui en résulte sera inacceptable, le processus ou le matériel utilisé par le processus doit être ajusté. Un ajustement doit être fait dès que possible afin de minimiser le risque d'autres dérives.

## IV.3 Processus Scrum

La figure ci-dessous décrit le principe de Scrum qui se résume comme suit :

- Propriétaire de produit (Le Product Owner) crée une liste des fonctionnalités prioritaires appelée Product Backlog.
- Pendant le Sprint Planning, l'équipe choisit une partie des fonctionnalités du Product Backlog pour préparer le Sprint Backlog et discuter la manière d'implémentation de ces fonctionnalités.
- L'équipe a une période de temps de deux à quatre semaines (Sprint) pour achever le travail, mais elle se réunit chaque jour pour évaluer le progrès (Daily Scrum).

- Tout au long de cette période, le Scrum Master maintient l'équipe qui est concentrée sur son objectif.
- À la fin de chaque sprint, le travail devrait être potentiellement livrable.
- Au début du Sprint suivant, l'équipe sélectionne une autre partie des fonctionnalités du Product Backlog et répète les étapes précédentes.
- Au-delà du Sprint, le cycle se répète jusqu'à ce que la liste des tâches du ProductBacklog soit achevée ou une date limite soit atteinte ou le budget du projet soit épuisé. Peu importe la cause de clôture, Scrum assure que le travail le plus important a été achevé lors de la fin du projet.



FIGURE I.1 – Le processus SCRUM [1]

## Conclusion

Dans ce chapitre, nous avons décrit le contexte général du projet. Nous avons commencé par aborder la problématique et le travail à réaliser. Ensuite, nous avons présenté les modèles de gestion de projet et enfin nous avons détaillé le modèle choisi.

Dans le chapitre suivant nous effectuons l'étude préalable de notre projet et nous présentons les exigences et les besoins.

## CHAPITRE II

# SPRINT 0 : PLANIFICATION ET ARCHITECTURE

## Introduction

Ce chapitre présente le sprint zéro dans la réalisation de notre projet. En premier lieu, nous identifions les acteurs de notre application. Ensuite, nous listons les besoins fonctionnels, non fonctionnels et du domaine de notre système et nous détaillons le travail par la méthodologie choisie dans le premier chapitre. Enfin, nous présentons l'architecture, le matériel de base, le langage de programmation ainsi que les technologies utilisées pour la mise en place de notre environnement de travail.

## I Analyse et spécification des besoins

Dans cette partie, nous présentons les acteurs, les besoins fonctionnels, non fonctionnels ainsi que les besoins du domaine.

### I.1 Identification des acteurs

Un acteur est une entité externe capable d'interagir avec le système. En réponse à des actions effectuées par un acteur, le système doit fournir des services qui répondent à ses besoins.

Les différents acteurs de notre application sont :

- ◆ Administrateur : Il est chargé de faire la mise à jour des profils des étudiants et des enseignants et la rectification. Il a pour mission de gérer toutes les données des étudiants et des enseignants. Il est chargé de consulter les absences des étudiants et leurs notes de chaque module selon les semestres. Il est chargé de gérer les matières, les emplois du temps, les inscriptions et de visualiser des statistiques.
- ◆ Etudiant : Il est chargé de visualiser les notes de chaque module selon les semestres, les absences ainsi que l'emploi de temps. Il peut consulter son profil.
- ◆ Enseignant : Il est chargé de gérer les absences des étudiants ainsi que leurs notes de chaque module selon les semestres. Il peut consulter son emploi de temps ainsi que son profil. Il peut aussi consulter la liste de ses étudiants par filière et par classe.

## **I.2 Identification des besoins**

Notre application doit satisfaire les exigences et les besoins de la totalité des utilisateurs de la plateforme. Nous présentons dans ce qui suit les besoins fonctionnels et non fonctionnels ainsi que les besoins du domaine des différents acteurs de notre application.

### **I.2.1 Besoins fonctionnels**

Les besoins fonctionnels sont ceux exprimés par le client. Ces besoins ont permis d'établir le cahier des charges. Les besoins fonctionnels relatifs à notre système sont :

- ◆ Module d'authentification : Chaque utilisateur doit pouvoir s'authentifier avec l'email et le mot de passe.
- ◆ Module de gestion des absences des étudiants : Il permet d'ajouter une absence à la liste des absences des étudiants, de modifier une absence ou de la supprimer. Il permet aussi de pouvoir consulter les absences.
- ◆ Module de gestion des notes : La gestion des notes consiste à affecter une note à un étudiant par matière et par semestre, de modifier une note ou d'ajouter une note. Il consiste aussi à pouvoir consulter les notes.
- ◆ Module de gestion des matières : La gestion des matières consiste à ajouter une matière pour chaque niveau et filière, de modifier une matière ou de la supprimer. Il consiste aussi à consulter les matières.
- ◆ Module de gestion des emplois du temps : La gestion des emplois du temps consiste à ajouter une séance à l'emploi de temps pour chaque niveau et filière, de modifier une séance dans un emploi ou de la supprimer. Il consiste aussi à consulter l'emploi de temps.

- ◆ Module de gestion des profils : Il permet d'ajouter un profil et de faire sa mise à jour. Il permet aussi de visualiser les profils.

### **I.2.2 Besoins non fonctionnels**

Ces besoins sont les fonctionnalités nécessaires afin de rendre le logiciel plus performant et qui se résument comme suit :

- ◆ Ergonomie : Les interfaces utilisateurs doivent être conviviales et faciles à utiliser.
- ◆ Fiabilité : L'application doit fonctionner correctement et ne doit pas contenir de bugs.
- ◆ Extensibilité : L'application doit permettre l'ajout de nouvelles fonctionnalités au moindre coût.
- ◆ Performance : Il s'agit de l'optimisation du temps de chargement des données ainsi que l'usage des bonnes pratiques de développement.
- ◆ Portabilité : L'application doit être accessible depuis n'importe quelle plateforme ou système d'exploitation.

### **I.2.3 Besoins du domaine**

Ils sont dérivés du domaine du système. Dans notre cas on s'intéresse essentiellement à l'information et à la sécurité de l'information. Pour cela on cite les besoins suivants :

- L'information peut être présentée quelle que soit sa forme de stockage, de traitement ou de transmission.
- Le mot de passe doit être individuel et ne doit pas être partagé.
- Chaque utilisateur a des droits d'accès spécifiques à lui.
- Le mot de passe doit être renouvelé chaque année.

## **I.3 Pilotage du projet avec Scrum**

Nous présentons dans cette partie l'ensemble des acteurs participants à la réalisation des différentes phases du projet.

L'équipe de développement est constituée, dans notre cas, de deux personnes chargées de la réalisation du projet allant de la conception au développement de l'application. Le Product Owner, le représentant du client et des utilisateurs, définit les fonctionnalités, les besoins et les priorités et dirige l'activité de l'équipe de développement. Le Scrum Master doit veiller au bon

déroulement du travail et à la bonne ambiance de l'équipe.

Dans le tableau suivant, nous présentons les rôles, les acteurs associés et leurs missions.

Rôle	Mission	Acteur
Scrum team	Conception, développement, tests et validation	Sarra Ben Abdelmalek Oussema henia
Product Owner	Définition des besoins et des fonctionnalités à développer	L'ENSI
Scrum master	Diriger l'équipe projet et veiller au bon déroulement de son travail	Mme Leila Ben Ayed

La définition des acteurs et de leurs rôles

### I.3.1 Les fonctionnalités du backlog produit

Le Backlog produit est un artéfact très important dans le processus Scrum. C'est l'ensemble des caractéristiques fonctionnelles ou techniques qui constituent le produit souhaité.

Le tableau suivant présente le backlog de notre produit :

- ID qui représente l'identifiant de la User Story,
- Thème qui sert à mieux ordonner les user stories, exemple : Gestion des ressources humaines
- User Story ou scénario comporte la description des user stories suivant la forme « En tant que . . . Je veux . . . »,

- La priorité de la User story selon la valeur métier et l'ordre de sa réalisation : la priorité est souvent exprimée par le Product Owner.

<b>Id</b>	<b>En tant que</b>	<b>User story</b>	<b>Thème</b>	<b>Priorité</b>
1	Utilisateur	je veux m'authentifier pour accéder à l'application.	Authentification	Elevée
2	Etudiant	je veux consulter mon profil.	Gestion des pro- fils	Elevée
3	Enseignant	je veux consulter mon profil.	Gestion des pro- fils	Elevée
4	Administrateur	je veux faire la mise à jour des profils des étudiants et des enseignants et la rectification.	Gestion des pro- fils	Elevée
5	Etudiant	je veux visualiser les emplois du temps .	Gestion des em- plois de temps	Moyenne
6	Administrateur	je veux gérer les emplois de temps.	Gestion des em- plois de temps	Moyenne
7	Enseignant	je veux consulter mon emploi de temps.	Gestion des em- plois de temps	Moyenne
8	Administrateur	je veux gérer les matières.	Gestion des ma- tières	Moyenne
9	Etudiant	je veux visualiser les absences dans chaque module.	Gestion des ab- sences	Moyenne
10	Administrateur	je veux visualiser les absences des étudiants.	Gestion des ab- sences	Moyenne
11	Enseignant	je veux gérer les absences des étudiants	Gestion des ab- sences	Moyenne
12	Etudiant	je veux visualiser les notes de chaque module selon les semestres.	Gestion des notes	Moyenne
13	Administrateur	je veux consulter les notes des étudiants dans chaque module selon les semestres.	Gestion des notes	Moyenne
14	Enseignant	je veux gérer les notes des étudiants dans chaque module selon les semestres.	Gestion des notes	Moyenne
15	Administrateur	je veux gérer les inscriptions des étudiants.	Gestion des ins- criptions	Elevée
16	Administrateur	je veux visualiser des statistiques relatives aux données des étudiants et des enseignants.	Visualisation des statistiques	Faible

TABLE II.1 – Tableau du Backlog Produit



### I.3.2 Planification des sprints

Une fois nous avons élaboré le backlog produit, nous avons établi la planification des sprints. Le but de cette réunion de planification est de construire le backlog de sprint en se basant sur le backlog produit exprimé par le product owner.

A la fin, nous avons identifié, les durées prévisionnelles du travail à effectuer durant chaque sprint. Chaque sprint est d'une durée d'un mois.

Le tableau suivant montre la répartition des sprints relatifs à notre projet.

Plan du sprint 1	Plan du sprint 2
Estimation : du 15/02 au 15/03	Estimation : du 16/03 au 16/04
S'authentifier	Gestion des emplois de temps
Gestion des matières	Gestion des notes
Gestion des profils des enseignants	Gestion des absences
Gestion des profils des étudiants	Visualisation des statistiques
Gestion des inscriptions	

Planning des sprints

## I.4 Environnement de Travail

Dans cette partie, nous présentons l'environnement matériel, l'environnement de développement ainsi que l'environnement logiciel et nous terminons cette section par la justification des choix des technologies utilisées.

### I.4.1 Environnement matériel

Tout au long de notre projet, nous avons eu à notre disposition un ordinateur portable qui dispose de la configuration suivante :

- LENOVO : Intel R Core i5 @ 1.8GHz, Ram : 8, 00 Go,
- Système d'exploitation : Windows 10,

#### I.4.2 Environnement de développement

**Visual Studio code** : “ Est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS “.

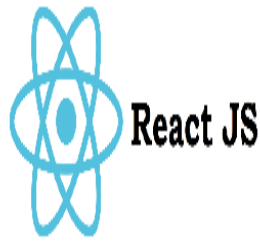


**Microsoft SQL SERVER** : “Microsoft SQL Server est un système de gestion de base de données en langage SQL incorporant entre autres un SGBDR développé et commercialisé par la société Microsoft“.



#### I.4.3 Environnement logiciel

**ReactJs** : “React est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'applications web mono page, via la création de composants dépendant d'un état et générant une page HTML à chaque changement d'état “.



[4]

**ASP.NET CORE :** “ ASP.NET Core est un Framework Web gratuit et open-source, développé par Microsoft et la communauté. Il est plus performant qu’ASP.NET. C’est un Framework modulaire qui fonctionne à la fois avec le Framework .NET, sous Windows et .NET Core en multiplate-forme. “



[5]

**Github :** “une plate-forme d’hébergement de code pour le contrôle de version et la collaboration“



[6]

#### **I.4.4 Justification des choix**

**ASP.NET CORE :** Ce Framework optimisé pour le cloud a pour origine ASP.NET qui est la solution web du .NET Framework. Apparu récemment (en 2016), il offre de nouveaux avantages en comparaison avec son ancêtre. ASP.NET Core peut être utilisé avec .NET Core

déployable sur tous les systèmes (Windows, OSX, Linux). Il est modulaire donc lors du déploiement, seules les composantes utilisées sont incluses dans le paquet, le rendant beaucoup plus léger.

Les principaux avantages du .NET Framework sont :

Language Integrated Query (LINQ) : Offre une syntaxe commune pour écrire des requêtes directement dans le code sur toute collection de données (listes, base de données...). Les erreurs sont attrapées à la compilation et LINQ est fortement typé.

async/await : Facilite le développement de programme multithread et les appels asynchrones.

C (en comparaison avec Java) : Le langage offre un support intégré de generic, lambda, variable dynamique (dynamic), types anonymes, getters et setters simplifié et autres.

Entity Framework Core (EF Core) : Évolution du mapping objet-relationnel (ORM). Il permet de créer la base de données et la contrôler à partir du code utilisé pour le backend (style Code First). Il intègre le principe de Repository Pattern pour manipuler les entités directement dans le code.

**REACT JS** : React JS propose une documentation détaillée.

Il est facile à apprendre car il présente une conception simple.

React permet de migrer entre les versions.

Les compétences acquises dans React peuvent être appliquées dans le développement natif.

## I.5 Architecture générale de l'application

Dans cette partie, nous allons définir l'architecture de l'application. Dans cette partie, nous nous intéressons à l'architecture logique.

### I.5.1 Présentation de quelques styles architecturaux logiques

Un style architectural est un patron décrivant une architecture logicielle permettant de résoudre un problème particulier. Le choix du style architectural se fait lors de la conception globale par l'architecte.

Style architectural	Description	Avantages	Inconvénients
Pipeline	Convient aux systèmes de traitement et de transformation de données. Composée d'un filtre qui reçoit les données des canaux d'entrée, les traite et les envoie vers les canaux de sortie et un canal unidirectionnel qui est l'intermédiaire entre le filtre producteur et le filtre consommateur.	<ul style="list-style-type: none"> <li>• Très flexible</li> <li>• Bon pour le traitement en lot</li> <li>• Décomposition fonctionnelle du système facilitée</li> </ul>	<ul style="list-style-type: none"> <li>• Mauvais pour le traitement interactif</li> </ul>
Avec référentiel de données	Utilisé lorsqu'il y a un partage de données entre les composants. Il nécessite deux types de composants qui sont un référentiel de données et un accesseur de données. Le référentiel présente le moyen de communication entre les accesseurs. Ce style nécessite aussi un connecteur qui relie l'accesseur au référentiel.	<ul style="list-style-type: none"> <li>• Compatible avec les applications contenant des tâches complexes sur les données et changeantes au cours du temps.</li> <li>• De nouveaux services peuvent être ajoutés.</li> </ul>	<ul style="list-style-type: none"> <li>• Le référentiel peut constituer un goulot d'étranglement.</li> </ul>
MVC	Composé de trois composants : Le modèle qui contient les classes dont les instances doivent être manipulées, la vue qui représente l'interface graphique et le contrôleur qui contient les fonctionnalités pour contrôler et gérer les interactions de l'utilisateur avec le modèle et la vue.	<ul style="list-style-type: none"> <li>• Adapté aux systèmes interactifs</li> <li>• La cohérence entre les données</li> </ul>	<ul style="list-style-type: none"> <li>• Goulot d'étranglement possible</li> </ul>
5 couches	Les composants : chaque composant réalise un service. Les connecteurs : dépendent du protocole d'interaction souhaité entre couches. La couche présentation représente l'interface graphique. La couche coordination gère les droits d'accès des utilisateurs.	<ul style="list-style-type: none"> <li>• Conception défensive : les API des couches constituent des endroits stratégiques pour insérer des assertions de vérification.</li> </ul>	<ul style="list-style-type: none"> <li>• Goulot d'étranglement possible</li> </ul>

	La couche services représente les services attendus par la plateforme. La couche domaine représente les différentes entités à gérer et à manipuler. La couche persistance gère la persistance des entités dans la base de données d'une façon persistante.	<ul style="list-style-type: none"> <li>• Il est facile d'ajouter de nouveaux services construits sur les services de plus bas niveau.</li> </ul>	
REST (Representational State Transfer)	Il permet de construire des applications (Web, Intranet, Web Services). C'est un ensemble de règles et de conventions à respecter. Il utilise le protocole HTTP.	<ul style="list-style-type: none"> <li>• Les WEB API sont compatibles avec divers Frontend.</li> </ul>	<ul style="list-style-type: none"> <li>• Architecture orientée ressources plutôt qu'orientée fonctions.</li> </ul>

Comparaison des styles architecturaux logiques

Dans notre cas, nous allons utiliser le style architectural REST étant donné qu'il présente divers avantages :

- ◆ Les API REST sont sans état donc le serveur ne sauvegarde pas les requêtes et les réponses des clients.
- ◆ Chaque API a une interface uniforme.
- ◆ Chaque composant utilisant REST n'a pas accès aux composants au-delà du composant avec lequel il interagit.
- ◆ Couplage faible entre le client et le serveur.

### I.5.2 Architecture logique

L'architecture logique d'un système décrit schématiquement les différents composants du système ainsi que les interactions qui s'effectuent entre eux. Cette architecture permet de développer des couches avec un couplage faible entre elles.

La figure suivante présente l'architecture logique de notre application.

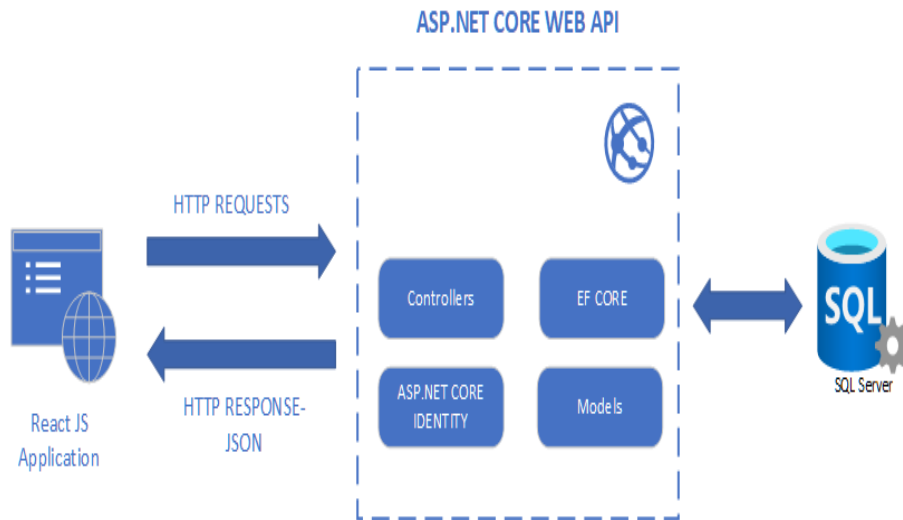


FIGURE II.1 – Architecture logique de l'application [7]

### I.5.3 Fonctionnement de l'architecture

**React JS Application :** est l'application chargée d'envoyer des requêtes http au composant ASP.NET CORE WEB API. Ces requêtes peuvent être de type GET, POST, DELETE ou PUT. Et la réponse est sous format JSON.

**ASP.NET CORE WEB API :** contient :

Les models qui sont les entités gérées,

Les controllers gèrent les requêtes HTTP entrantes et renvoient la réponse à l'appelant,

Entity Framework Core qui joue le rôle d'un ORM (Object relational Mapping) qui va faire la correspondance entre le modèle orienté objet et le modèle relationnel d'une part et la persistance des données d'autre part,

ASP.NET CORE IDENTITY qui est un API qui gère les utilisateurs, les mots de passe, les données de profil, les rôles, les revendications, les jetons, la confirmation par e-mail, etc.

**SQL SERVER :** est le système de gestion de base de données utilisé.

## I.6 Diagramme de déploiement

La figure suivante présente le diagramme de déploiement de notre application. Il montre l'architecture matérielle qui compose le système. Les ressources matérielles sont représentées sous forme de nœuds.

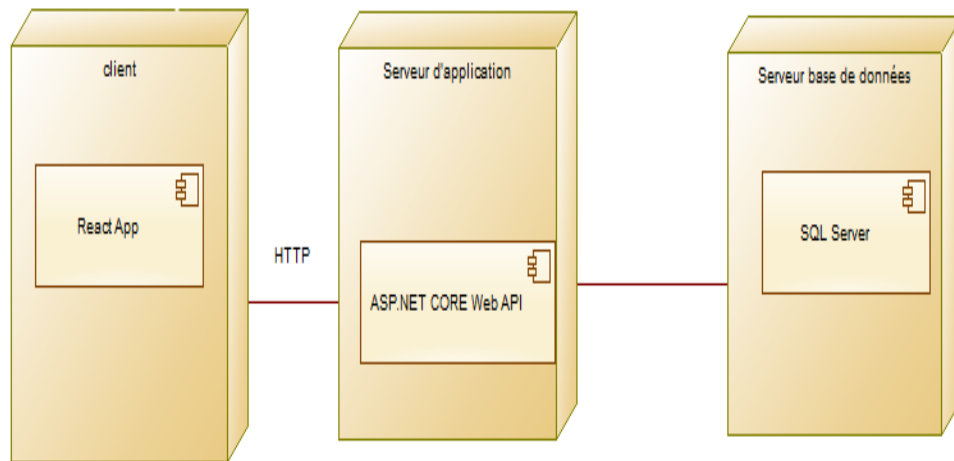


FIGURE II.2 – Diagramme de déploiement

## Conclusion

Au cours de ce chapitre nous avons identifié les besoins fonctionnels, non fonctionnels et du domaine de notre système ainsi que les acteurs. Ensuite, nous avons présenté la première étape de la méthodologie que nous avons choisie à savoir la réalisation du Backlog produit et des sprints et l'identification de l'équipe de travail. Puis , nous avons identifié l'environnement matériel et logiciel que nous utiliserons pour développer notre plateforme.

Dans le chapitre suivant nous entamons la réalisation du premier sprint.



## CHAPITRE III

# ÉTUDE ET RÉALISATION DU SPRINT 1

## Introduction

Dans ce chapitre, nous présentons la réalisation du premier sprint de notre projet, en divisant le travail en quatre phases principales qui sont l'analyse, la conception, la réalisation ainsi que les tests.

## I Backlog du sprint

Un sprint c'est le cœur du processus Scrum. Il s'agit d'une période de temps durant laquelle un incrément du produit sera réalisé. Chaque sprint a une durée constante bien déterminée, et chaque sprint ne doit démarrer qu'à condition que le précédent sprint soit achevé. Avant de commencer un sprint, l'équipe Scrum doit obligatoirement définir l'objectif de ce dernier sous la forme d'un tableau descriptif qui détermine la charge de travail pour chaque tâche contenue dans le sprint en termes de nombre de jours.

Le tableau suivant décrit la répartition de la charge de travail pour chaque fonctionnalité contenue dans le backlog du sprint.

ID	Histoire	Estimation
1	S'authentifier	3 jours
2	Gérer les matières	7 jours
3	Gérer le profil des enseignants	7 jours
4	Gérer le profil des étudiants	6 jours
5	Gérer les inscriptions	7 jours

TABLE III.1 – Backlog du sprint 1

## II Spécification fonctionnelle

Dans cette partie nous présentons la phase d'analyse du projet qui répond à la question « que fait le système ? ». La réponse à cette question se traduit par la présentation du diagramme des cas d'utilisation puis la description textuelle de chacun d'entre eux.

### II.1 Diagrammes de cas d'utilisation

La figure suivante décrit le diagramme de cas d'utilisation global du sprint 1.

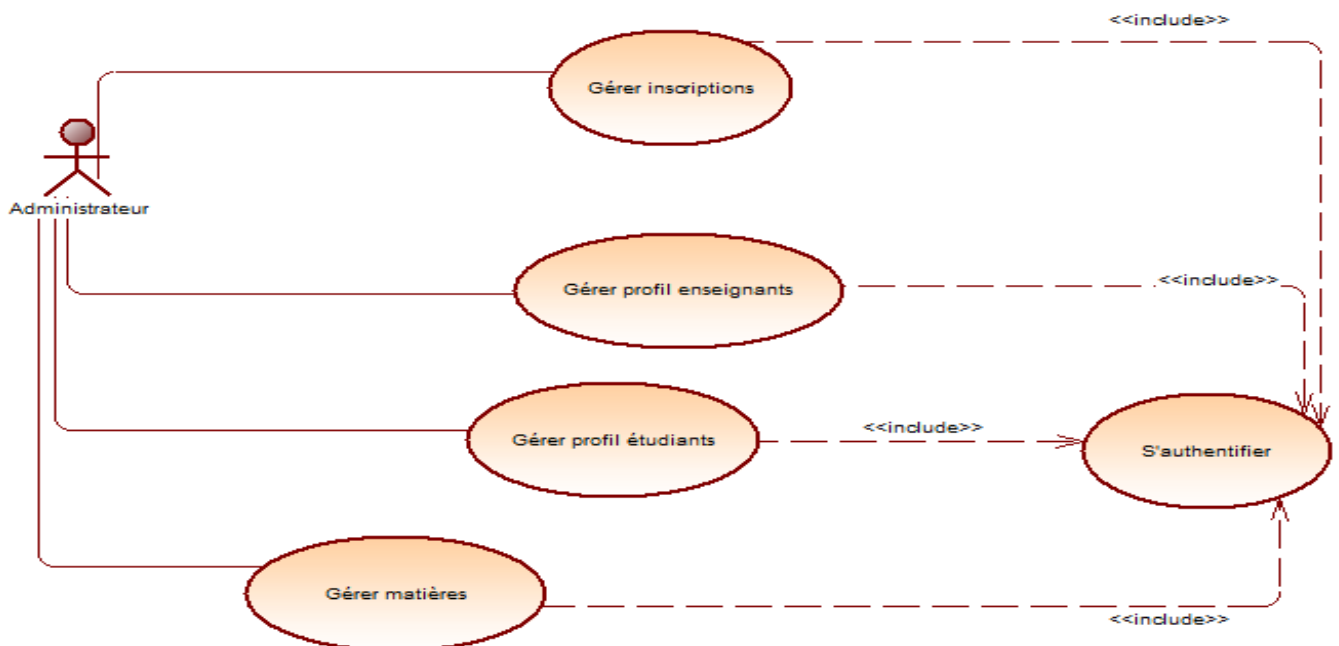


FIGURE III.1 – Diagramme de cas d'utilisation global du sprint 1

La figure suivante décrit le diagramme de cas d'utilisation de l'opération s'authentifier.

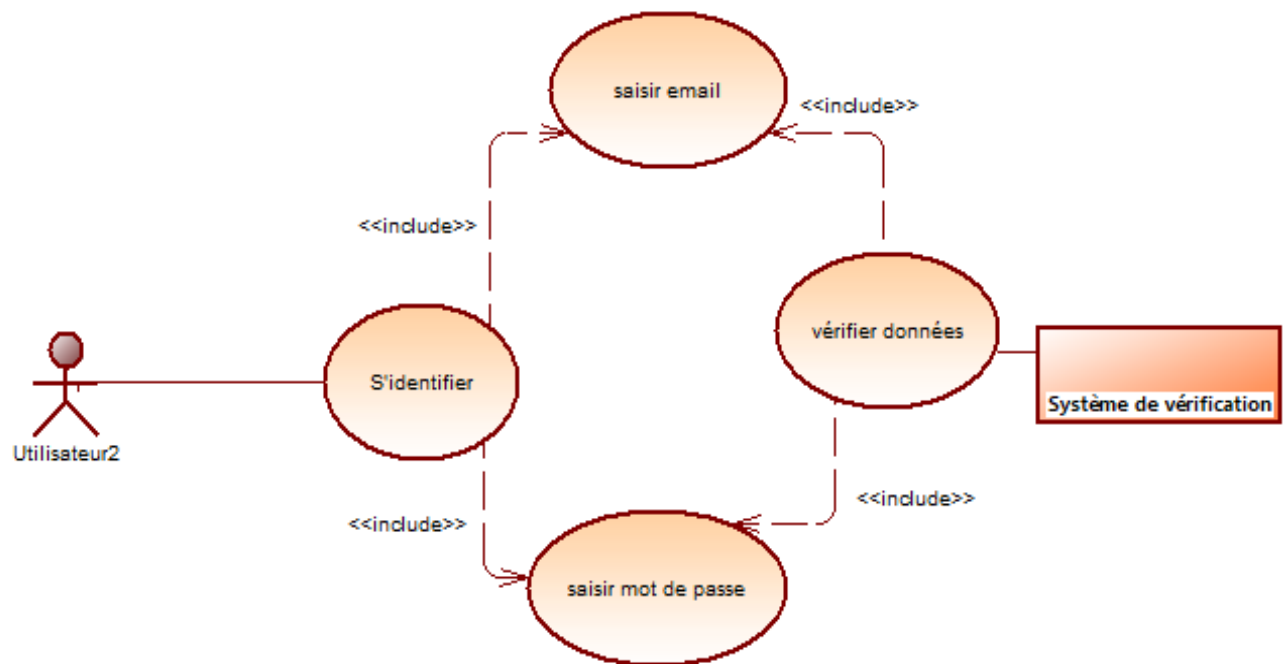


FIGURE III.2 – Diagramme de cas d'utilisation de l'opération s'authentifier

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer matière.

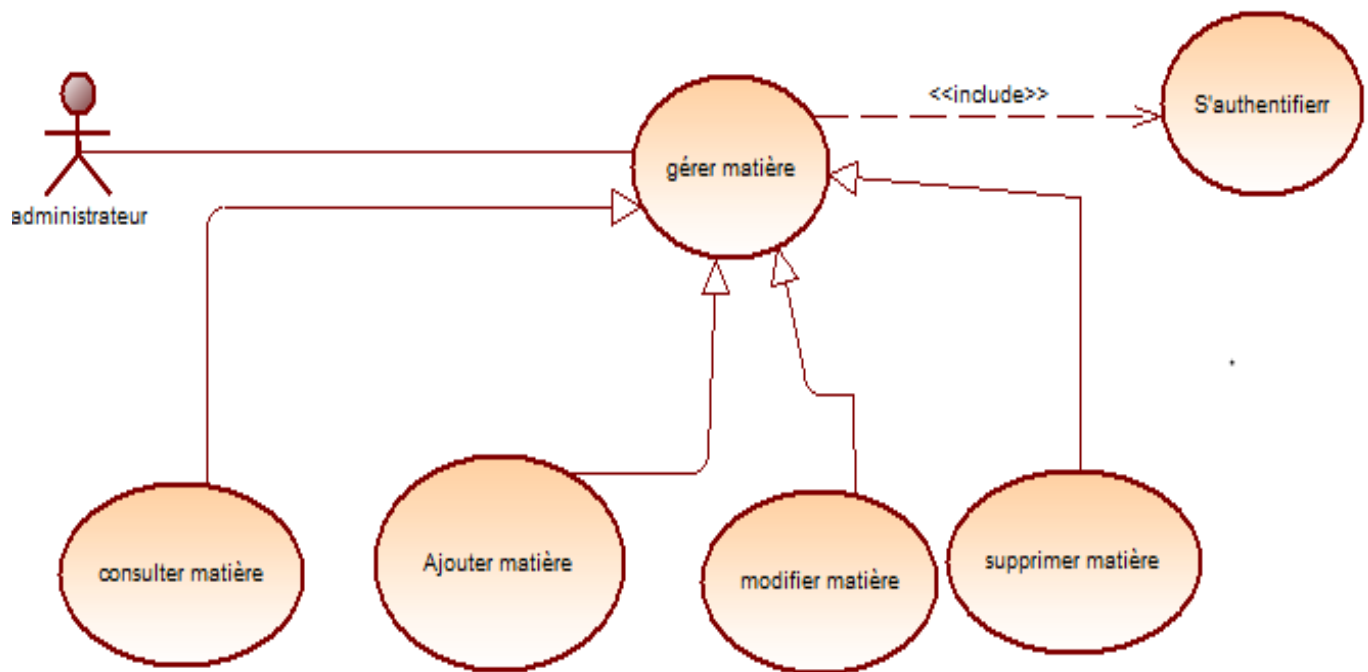


FIGURE III.3 – Diagramme de cas d'utilisation de l'opération gérer matière

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer profil étudiant.

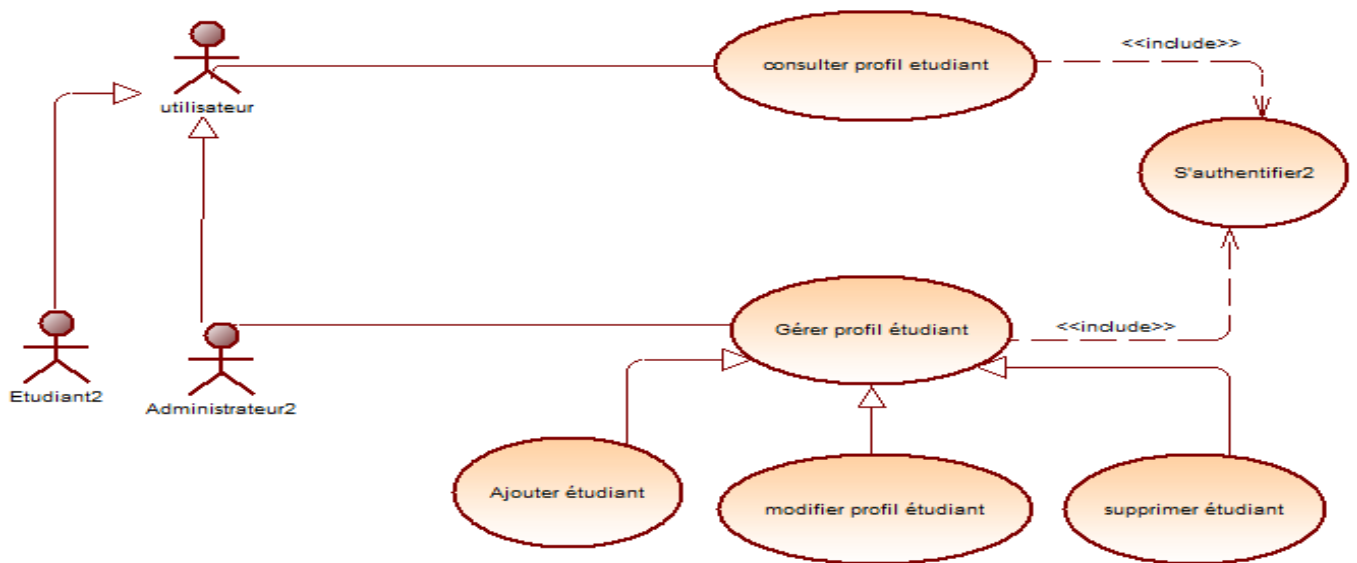


FIGURE III.4 – Diagramme de cas d'utilisation de l'opération gérer profil étudiant

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer profil enseignant.

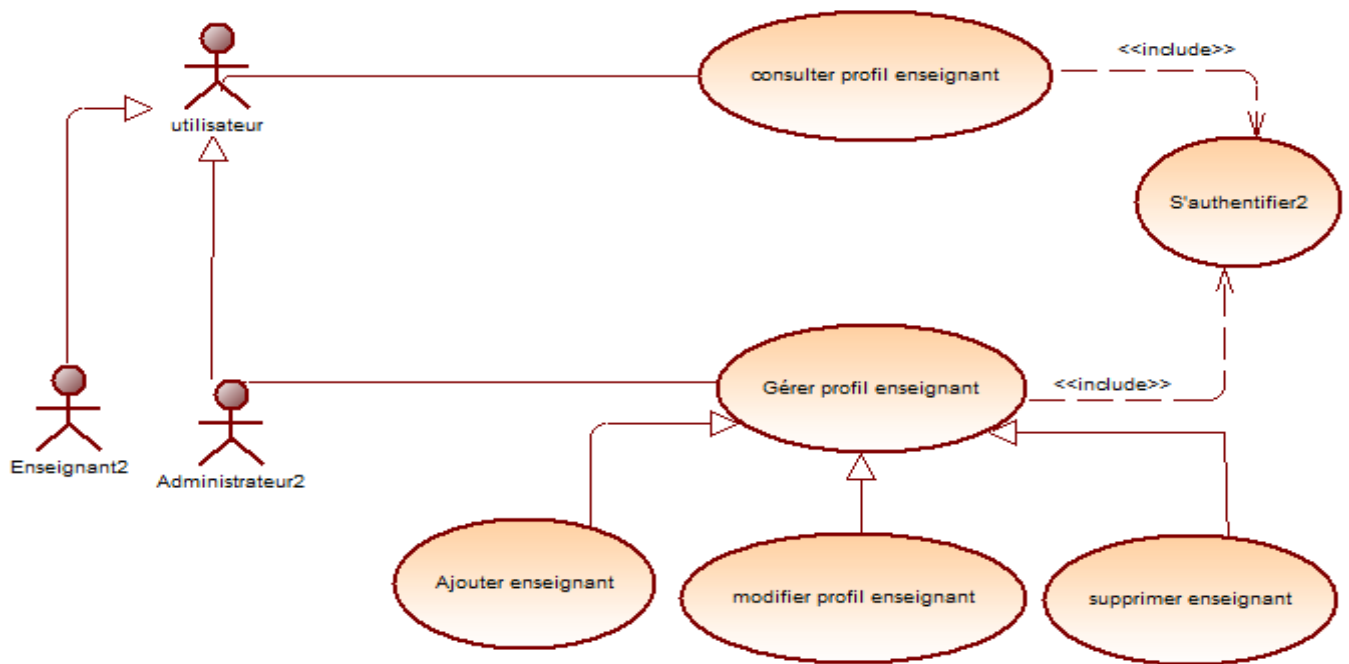


FIGURE III.5 – Diagramme de cas d'utilisation de l'opération gérer profil enseignant

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer inscriptions.

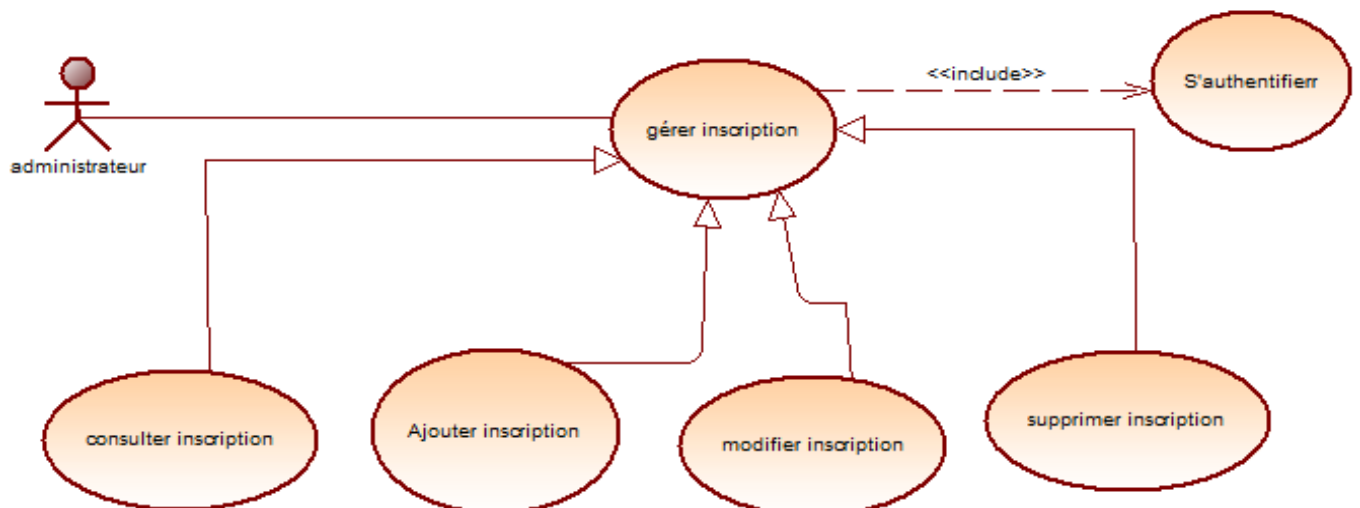


FIGURE III.6 – Diagramme de cas d'utilisation de l'opération gérer inscriptions

## II.2 Descriptions textuelles des cas d'utilisation

- Description textuelle du cas d'utilisation « s'authentifier »

Le tableau suivant décrit la description textuelle du cas d'utilisation « s'authentifier » .

Titre	S'authentifier
Acteur principal	Utilisateur
Résumé	A travers ce cas, l'utilisateur peut s'authentifier et par conséquent accéder à la plateforme en utilisant un email et un mot de passe.
Pré-condition	Utilisateur non authentifié.
Scénario principal	1.L'utilisateur saisit l'email. 2. L'utilisateur saisit le mot de passe. 3. L'utilisateur valide en cliquant sur le bouton se connecter.
Post-condition	Utilisateur authentifié
Exception	Réaffichage de l'interface d'authentification

Description textuelle du cas d'utilisation « s'authentifier »

- Raffinement de l'opération « Gérer matière »

Description textuelle du cas d'utilisation « ajouter matière »

Le tableau suivant décrit la description textuelle du cas d'utilisation « ajouter matière »

Titre	Ajouter matière
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut remplir les informations concernant la matière et les enregistrer.
Pré-conditions	Utilisateur authentifié, Matière inexistante.
Scénario principal	1. Cliquer sur ajouter matière 2. remplir les informations relatives à la matière 3. cliquer sur Ajouter
Post-condition	Matière ajoutée.
Exception	Un message d'erreur est affiché le cas échéant

Description textuelle du cas d'utilisation « ajouter matière »

Description textuelle du cas d'utilisation « Modifier matière »

Le tableau suivant décrit la description textuelle du cas d'utilisation « Modifier matière »

Titre	Modifier matière
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut modifier les informations d'une matière déjà créée.
Pré-conditions	Utilisateur authentifié, Matière non modifiée.
Scénario principal	1.choisir la matière à modifier 2. cliquer sur modifier 3. saisir les informations à modifier 4.sauvegarder les informations en cliquant sur enregistrer
Post-condition	Matière modifiée.
Exception	Un message d'erreur est affiché le cas échéant

Description textuelle du cas d'utilisation « Modifier matière »

Description textuelle du cas d'utilisation « supprimer matière »

Le tableau suivant décrit la description textuelle du cas d'utilisation « supprimer matière »

Titre	Supprimer matière
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut supprimer une matière si elle n'est plus enseignée.
Pré-conditions	Utilisateur authentifié, Matière non supprimée.
Scénario principal	1.choisir la matière à supprimer 2. cliquer sur supprimer
Post-condition	Matière supprimée.
Exception	Un message d'erreur est affiché le cas échéant

Description textuelle du cas d'utilisation « Supprimer matière »

Description textuelle du cas d'utilisation « consulter matière »

Le tableau suivant décrit la description textuelle du cas d'utilisation « Consulter matière »

Titre	Consulter matière
Acteur principal	étudiant
Résumé	A travers ce cas, l'étudiant peut consulter les informations d'une matière.
Pré-conditions	Utilisateur authentifié ,matière non consultée.
Scénario principal	1. Choisir la matière à consulter et cliquer dessus.
Post-condition	Matière consultée.

Description textuelle du cas d'utilisation « consulter matière »

- Raffinement de l'opération « Gérer profil étudiant »

Description textuelle du cas d'utilisation « ajouter étudiant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « ajouter étudiant ».

Titre	ajouter étudiant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut ajouter un étudiant à la liste des étudiants de l'école.
Pré-conditions	Utilisateur authentifié ,Etudiant non encore ajouté.
Scénario principal	1. Cliquer sur ajouter étudiant 2. remplir les informations relatives à l'étudiant 3. cliquer sur enregistrer
Post-condition	Etudiant ajouté.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « ajouter étudiant »

Description textuelle du cas d'utilisation « modifier étudiant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « modifier étudiant ».



Titre	modifier étudiant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut modifier les données d'un étudiant.
Pré-conditions	Utilisateur authentifié , Données de l'étudiant non modifiées.
Scénario principal	1.choisir l'étudiant à modifier ses données 2. cliquer sur modifier 3. saisir les informations à modifier 4.sauvegarder les informations en cliquant sur enregistrer
Post-condition	Données étudiant modifiées.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « modifier étudiant »

Description textuelle du cas d'utilisation « supprimer étudiant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « supprimer étudiant ».

Titre	supprimer étudiant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut supprimer un étudiant.
Pré-conditions	Utilisateur authentifié , Etudiant non supprimé.
Scénario principal	1. Choisir l'étudiant à supprimer 2. Cliquer sur supprimer
Post-condition	Etudiant supprimé.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « supprimer étudiant »

Description textuelle du cas d'utilisation « consulter liste étudiants »

Le tableau suivant décrit la description textuelle du cas d'utilisation « consulter liste étudiants ».

Titre	Consulter liste étudiants
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut consulter la liste des étudiants de l'école.
Pré-conditions	Utilisateur authentifié ,liste non consultée.
Scénario principal	1. choisir la rubrique Etudiants 2. choisir le groupe 3. visualiser la liste des étudiants
Post-condition	Liste consultée.

Description textuelle du cas d'utilisation « Consulter liste étudiants »

- Raffinement de l'opération « Gérer profil enseignant »

Description textuelle du cas d'utilisation « ajouter enseignant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « ajouter enseignant ».

Titre	ajouter enseignant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut ajouter un enseignant à la liste des enseignants de l'école.
Pré-conditions	Utilisateur authentifié ,Enseignant non encore ajouté.
Scénario principal	1. Cliquer sur ajouter enseignant 2. remplir les informations relatives à l'enseignant 3. cliquer sur enregistrer
Post-condition	Enseignant ajouté.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « ajouter enseignant »

Description textuelle du cas d'utilisation « modifier enseignant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « modifier enseignant ».

Titre	modifier enseignant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut modifier les données d'un enseignant.
Pré-conditions	Utilisateur authentifié ,Données de l'enseignant non modifiées.
Scénario principal	1.choisir l'enseignant à modifier ses données 2. cliquer sur modifier 3. saisir les informations à modifier 3.sauvegarder les informations en cliquant sur enregistrer
Post-condition	Données enseignant modifiées.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « modifier enseignant »

Description textuelle du cas d'utilisation « supprimer enseignant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « supprimer enseignant ».

Titre	Supprimer enseignant
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut supprimer un enseignant.
Pré-conditions	Utilisateur authentifié ,Enseignant non supprimé.
Scénario principal	1. Choisir l'enseignant à supprimer 2. Cliquer sur supprimer
Post-condition	Enseignant supprimé.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « supprimer enseignant »

Description textuelle du cas d'utilisation « consulter profil enseignant »

Le tableau suivant décrit la description textuelle du cas d'utilisation « consulter profil enseignant ».

Titre	consulter profil enseignant
Acteur principal	enseignant
Résumé	A travers ce cas, l'enseignant peut consulter son profil.
Pré-conditions	Utilisateur authentifié ,profil non consulté.
Scénario principal	1. Visualiser les données de son profil
Post-condition	Profil consulté.

Description textuelle du cas d'utilisation « Consulter profil enseignant »

- Raffinement de l'opération « Gérer inscriptions »

Description textuelle du cas d'utilisation « ajouter inscription »

Le tableau suivant décrit la description textuelle du cas d'utilisation « ajouter inscription »

Titre	ajouter inscription
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut ajouter une inscription d'un étudiant à la liste des inscriptions.
Pré-conditions	Utilisateur authentifié ,Inscription non encore ajoutée.
Scénario principal	1. Cliquer sur ajouter inscription 2. remplir les informations relatives à l'inscription 3. cliquer sur enregistrer
Post-condition	Inscription ajoutée.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « ajouter inscription »

Description textuelle du cas d'utilisation « modifier inscription »

Le tableau suivant décrit la description textuelle du cas d'utilisation « modifier inscription »

Titre	modifier inscription
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut modifier une inscription.
Pré-conditions	Utilisateur authentifié ,Inscription non mise à jour.
Scénario principal	1.choisir l'inscription à modifier 2. cliquer sur modifier 3. saisir les informations à modifier 4.sauvegarder les informations en cliquant sur enregistrer
Post-condition	Inscription modifiée.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « modifier inscription »

Description textuelle du cas d'utilisation « supprimer inscription »

Le tableau suivant décrit la description textuelle du cas d'utilisation « supprimer inscription »

Titre	supprimer inscription
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut supprimer une inscription.
Pré-conditions	Utilisateur authentifié ,Inscription non supprimée.
Scénario principal	1. Choisir l'inscription à supprimer 2. Cliquer sur supprimer
Post-condition	Inscription supprimée.
Exception	Un message d'erreur est affiché le cas échéant.

Description textuelle du cas d'utilisation « supprimer inscription »

Description textuelle du cas d'utilisation « consulter inscription »

Le tableau suivant décrit la description textuelle du cas d'utilisation « consulter inscription »

Titre	consulter inscription
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut consulter les inscriptions des étudiants.
Pré-conditions	Utilisateur authentifié , Inscription non consultée.
Scénario principal	1. Cliquer sur l'alternative inscriptions 2. Choisir l'inscription à consulter et cliquer dessus
Post-condition	Inscription consultée.

Description textuelle du cas d'utilisation « consulter inscription »

### III Conception détaillée

Dans cette section, nous présentons les patrons de conception utilisés, nous détaillons par la suite le diagramme de classes du premier sprint, ensuite nous abordons les diagrammes de séquence relatifs à ce sprint.

#### III.1 Patrons de conception

**Repository pattern :** Ce patron de conception est utilisé dans l'API afin de séparer la logique du domaine (business logic) de la logique faisant l'accès aux données, ainsi cette dernière peut être testée séparément. Nous pouvons aussi modifier la façon d'accéder aux données sans briser le reste du code.

**L'injection de dépendance :** Eviter de faire une dépendance directe entre deux classes d'une manière statique et veiller à ce que cette dépendance soit dynamique. Dans notre cas, pour diminuer le couplage, les classes controllers utilisent les classes interfaces qui se trouvent dans le package services et (son implémentation) permettra la communication avec la base de données en utilisant les DTO. (voir annexe)

#### III.2 Diagramme de classes du sprint 1

La figure ci-dessous présente le diagramme de classe des entités relatives au sprint 1 et qui va être construit au fur et à mesure tout au long des deux sprints.

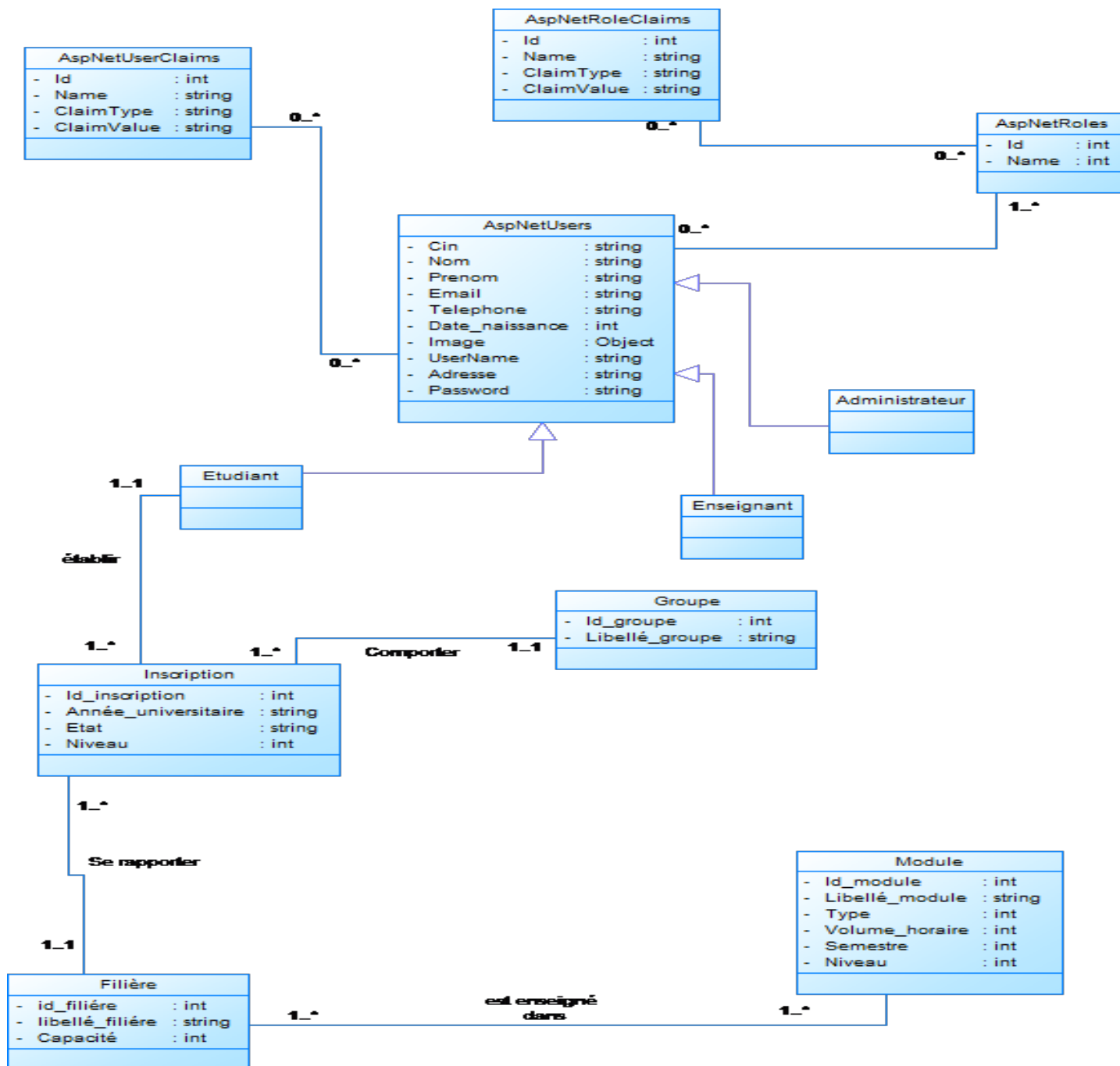


FIGURE III.7 – Diagramme de classes du sprint 1

AspNetUsers : C'est une classe prédéfinie qui comporte les données des utilisateurs

AspNetUserClaims : Classe prédéfinie. Utilisée par ASP.NET CORE IDENTITY pour stocker des informations supplémentaires que l'utilisateur possède.

AspNetRoles : C'est une classe prédéfinie qui comporte les rôles des utilisateurs.

AspNetRoleClaims : C'est une classe prédéfinie qui représente une revendication attribuée à tous les utilisateurs d'un rôle.

### III.3 Diagrammes de séquence détaillés

Un diagramme de séquence permet la représentation des différentes interactions entre les objets de notre système suivant un ordre chronologique.

Nous présentons dans ce qui suit les diagrammes des séquences détaillés des fonctionnalités du premier sprint.

Diagramme de séquence du cas d'utilisation « s'authentifier »

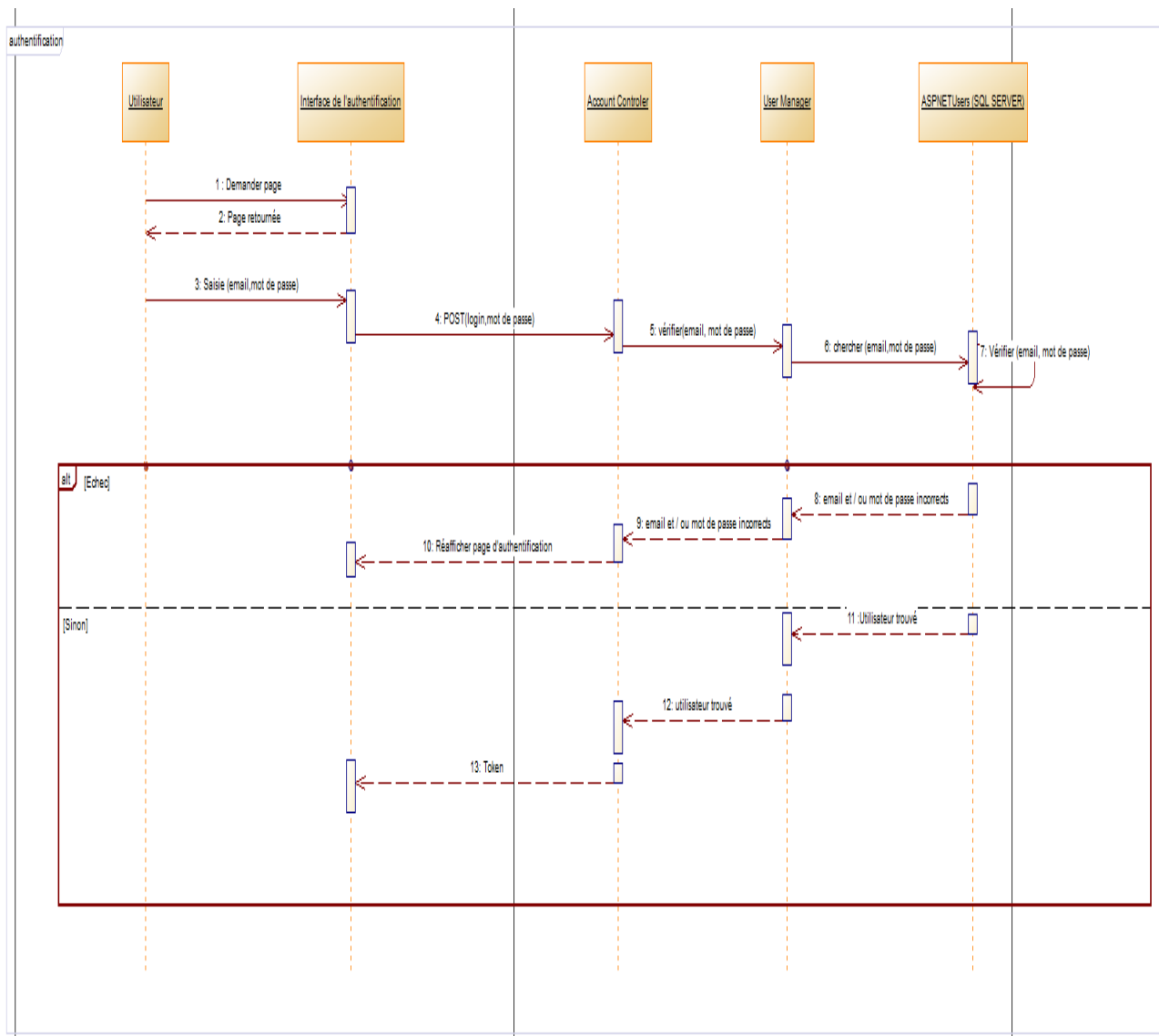


FIGURE III.8 – Diagramme de séquence du cas d'utilisation s'authentifier

Le scénario est le suivant :

- L'utilisateur demande l'accès à la page d'authentification
- L'interface d'authentification lui retourne la page demandée



- L'utilisateur saisit l'email et le mot de passe
- Après vérification de l'email et du mot de passe, s'ils sont corrects le contrôleur envoie un Token (voir annexe), sinon la page d'authentification réapparaît.

Diagramme de séquence du cas d'utilisation « Ajouter matière »

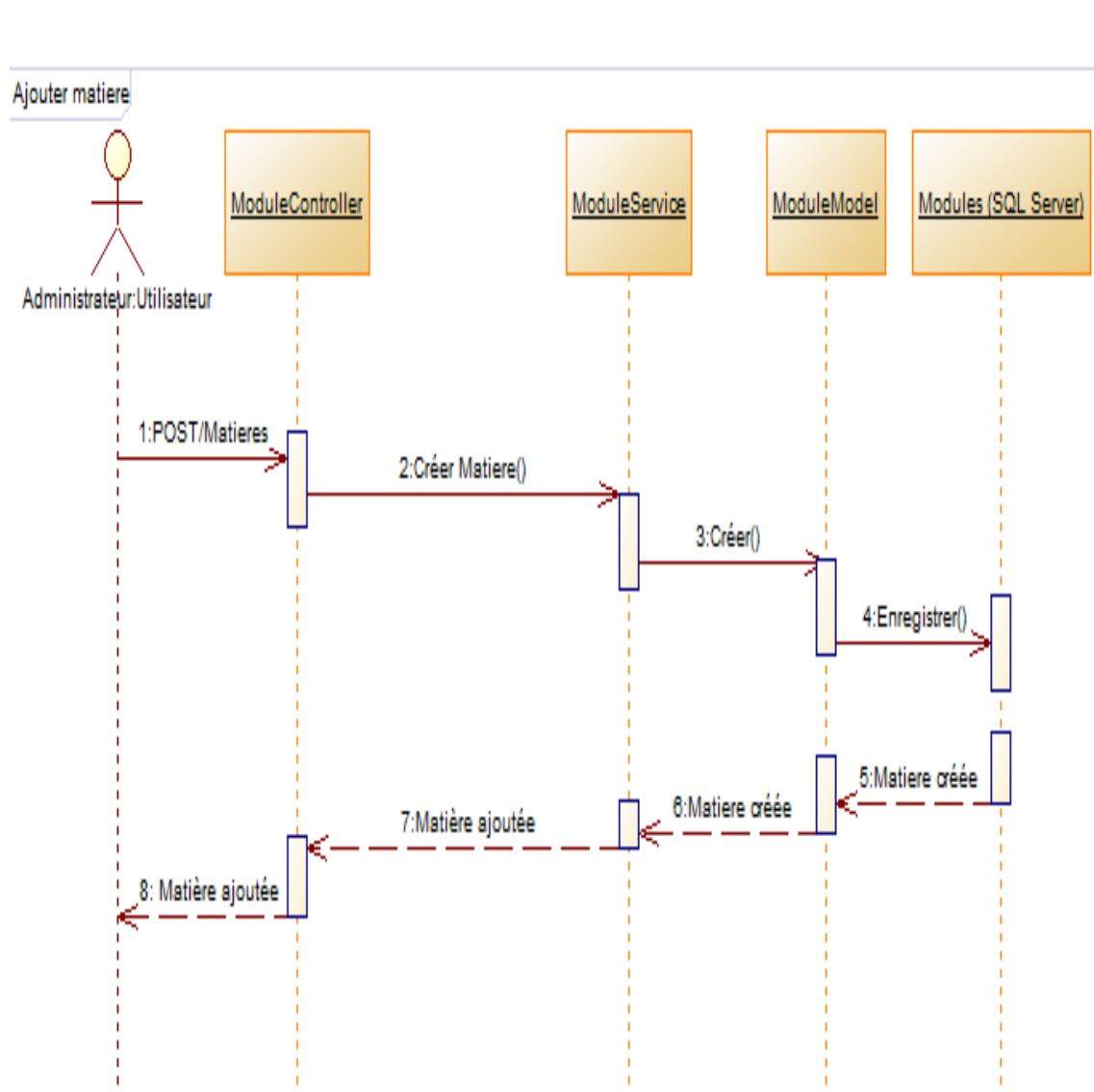


FIGURE III.9 – Diagramme de séquence du cas d'utilisation ajouter matière

Le scénario est le suivant :

- L'administrateur envoie une requête de type POST au contrôleur en cliquant sur le bouton ajouter une matière
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la création de la matière
- Les informations de la matière sont enregistrées automatiquement dans la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé

donc de l'ajout de la matière.

Diagramme de séquence du cas d'utilisation « Modifier Matière »

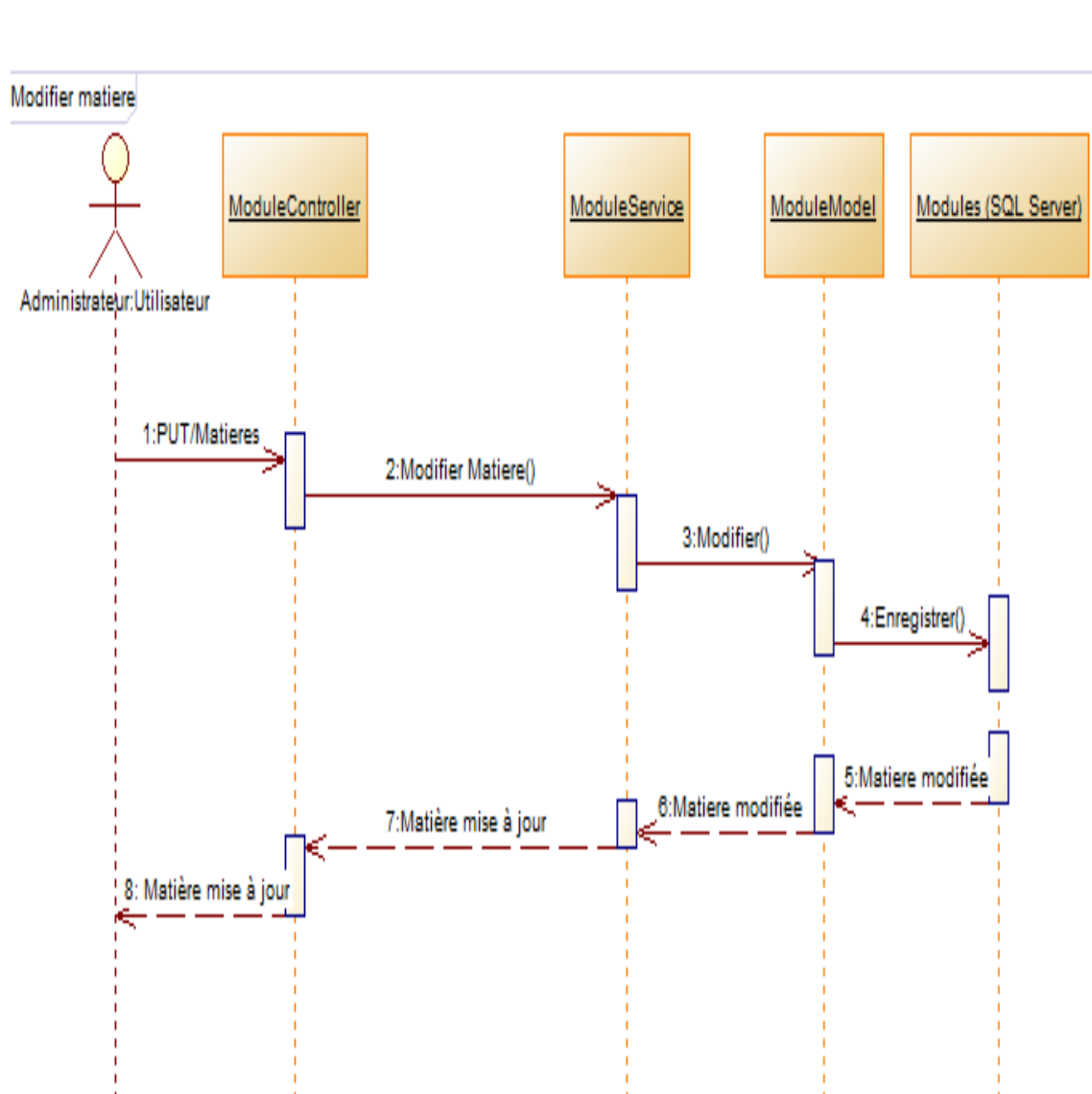


FIGURE III.10 – Diagramme de séquence du cas d'utilisation modifier matière

Le scénario est le suivant :

- L'administrateur envoie une requête de type PUT au contrôleur en cliquant sur l'icône modifier
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la modification des informations de la matière
- Les informations modifiées de la matière sont enregistrées automatiquement dans la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé

donc de la modification de la matière.

Diagramme de séquence du cas d'utilisation « supprimer matière »

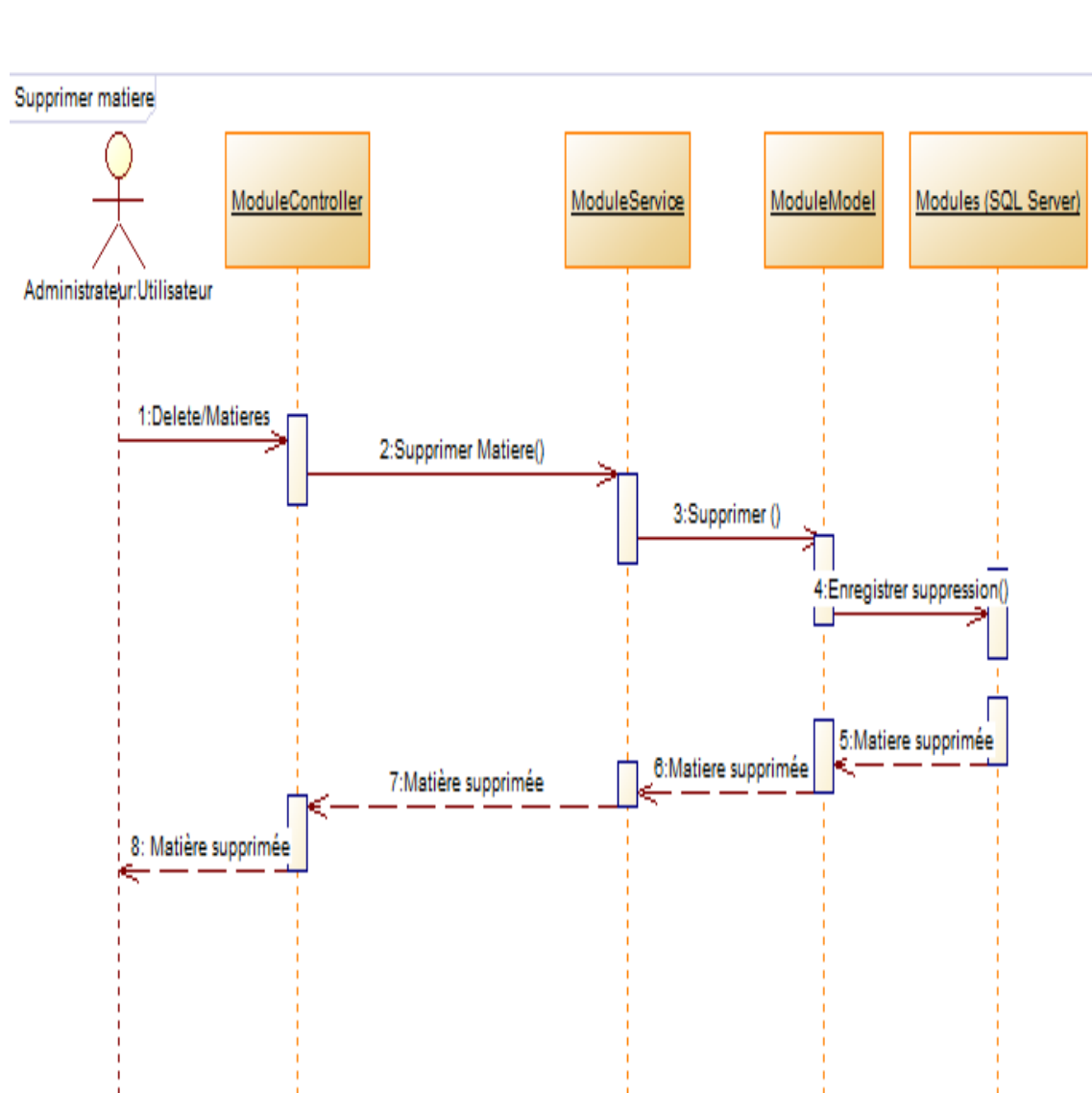


FIGURE III.11 – Diagramme de séquence du cas d'utilisation supprimer matière

Le scénario est le suivant :

- L'administrateur envoie une requête de type DELETE au contrôleur en cliquant sur l'icône supprimer
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la suppression de la matière
- La matière est supprimée automatiquement de la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé donc de la suppression de la matière.

Diagramme de séquence du cas d'utilisation « consulter matière »

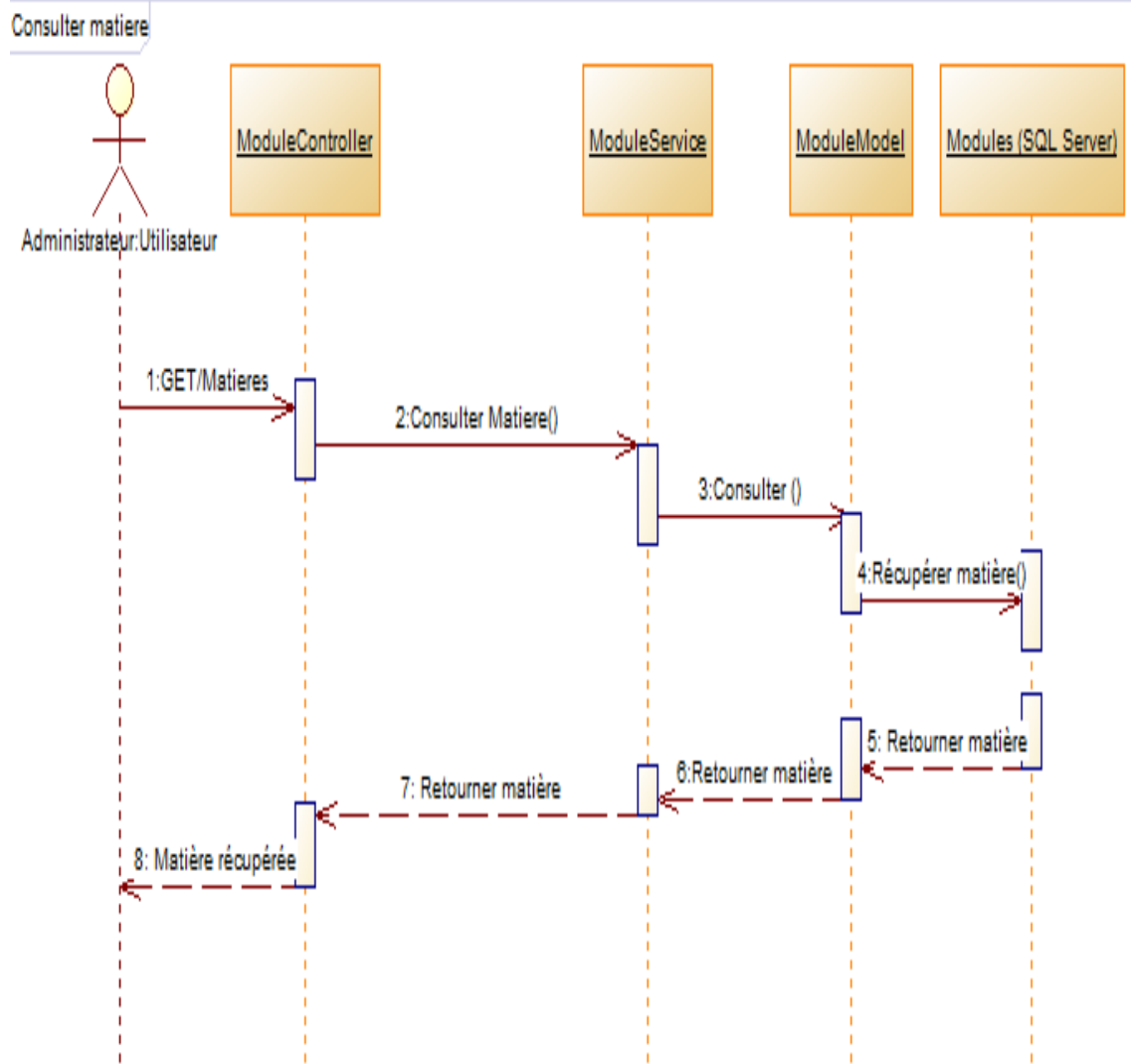


FIGURE III.12 – Diagramme de séquence du cas d'utilisation consulter matière

Le scénario est le suivant :

- L'administrateur envoie une requête de type GET au contrôleur en cliquant sur la matière à consulter
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la récupération de la matière
- La matière est récupérée de la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur peut donc consulter la matière.

**Remarque :**

Les diagrammes de séquences des opérations gérer profil étudiant, gérer profil enseignant et

gérer inscriptions ne sont pas présentés car ils sont semblables à l'opération gérer matières.

## IV Réalisation

Cette partie concerne la présentation du travail qui concerne le premier sprint à l'aide des captures d'écrans des différentes interfaces.

### Interface d'authentification

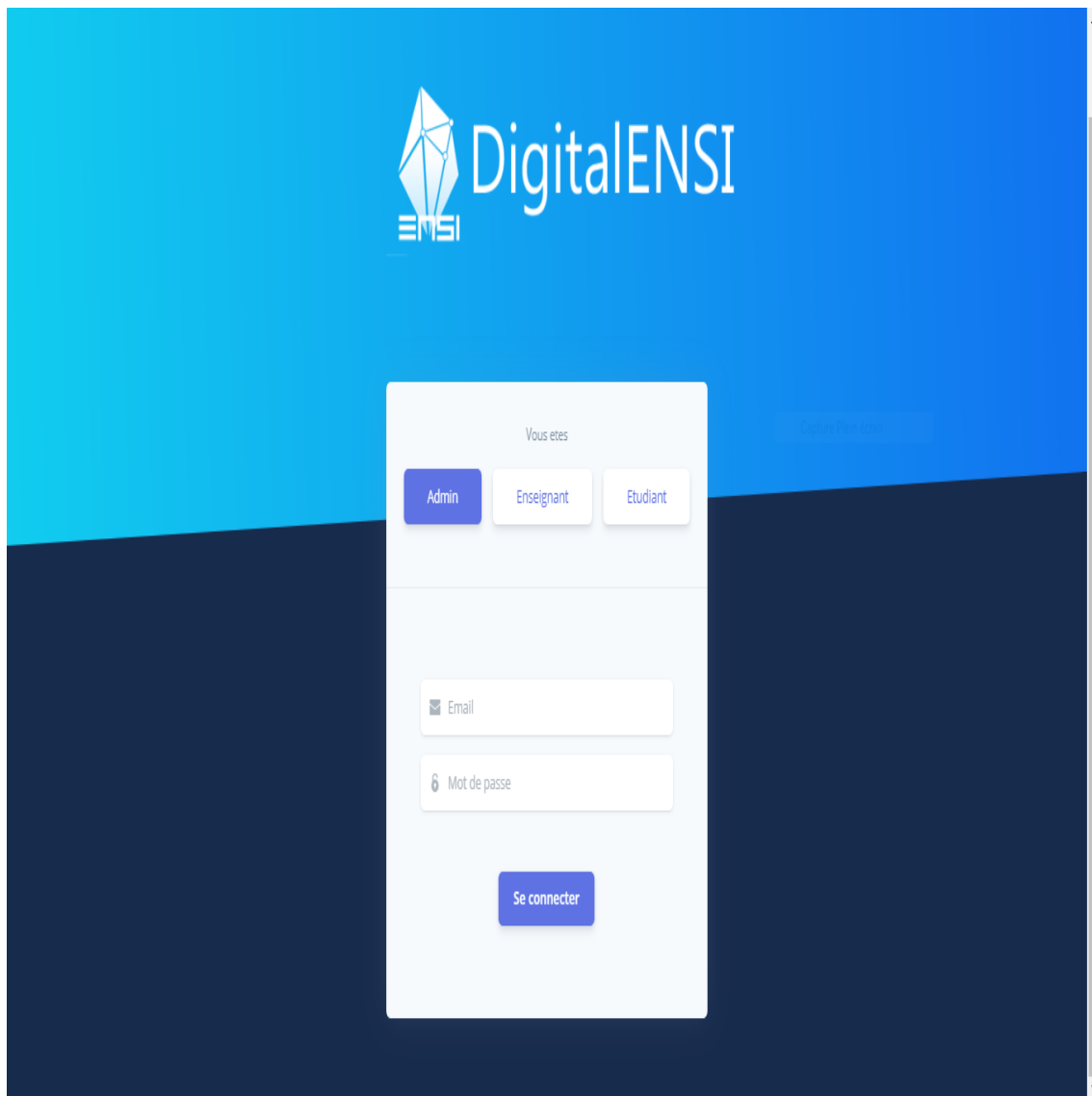


FIGURE III.13 – Interface d'authentification

Interface d'ajout de matière

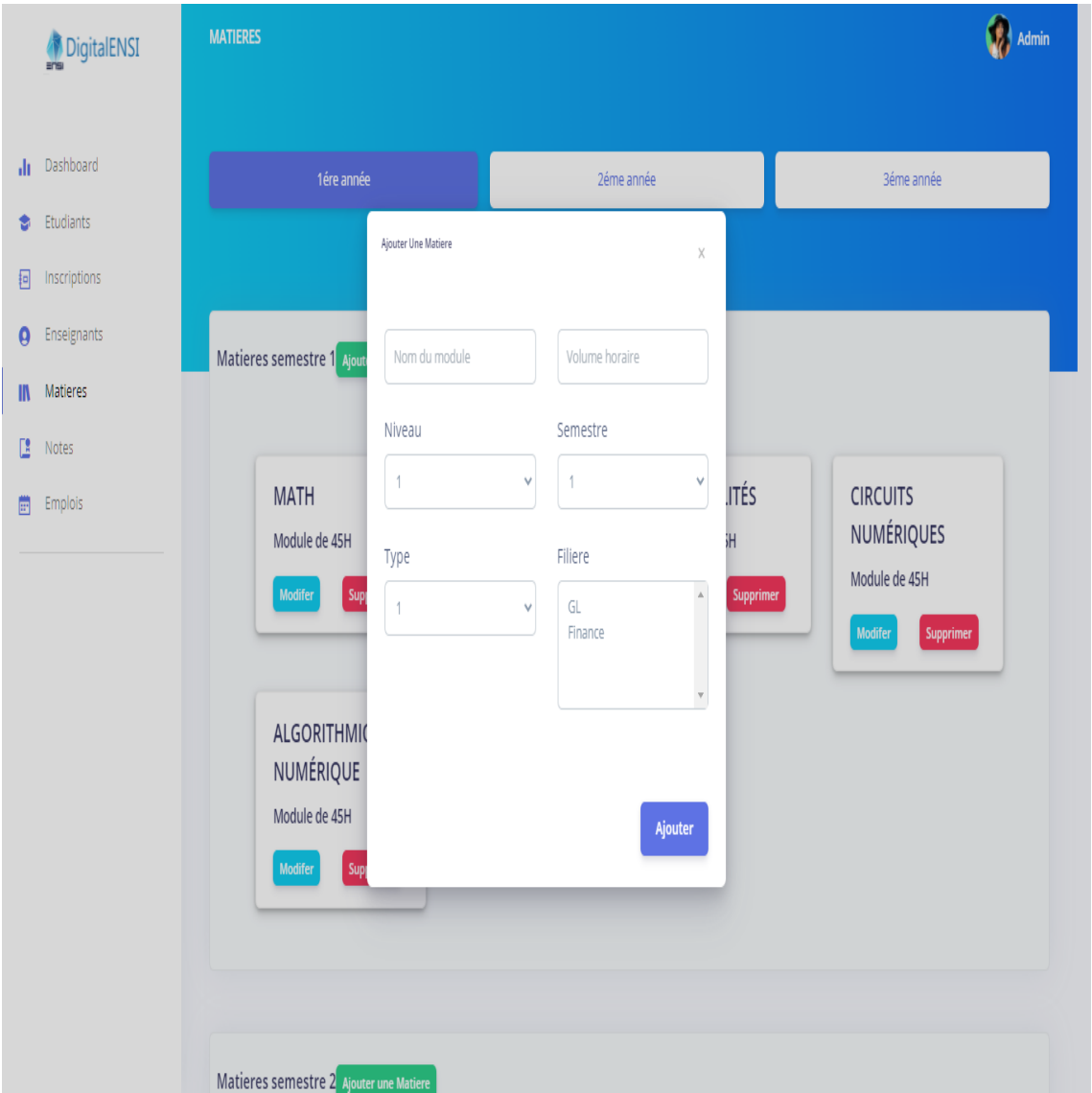


FIGURE III.14 – Interface d'ajout de matière

# Interface de modification de matière

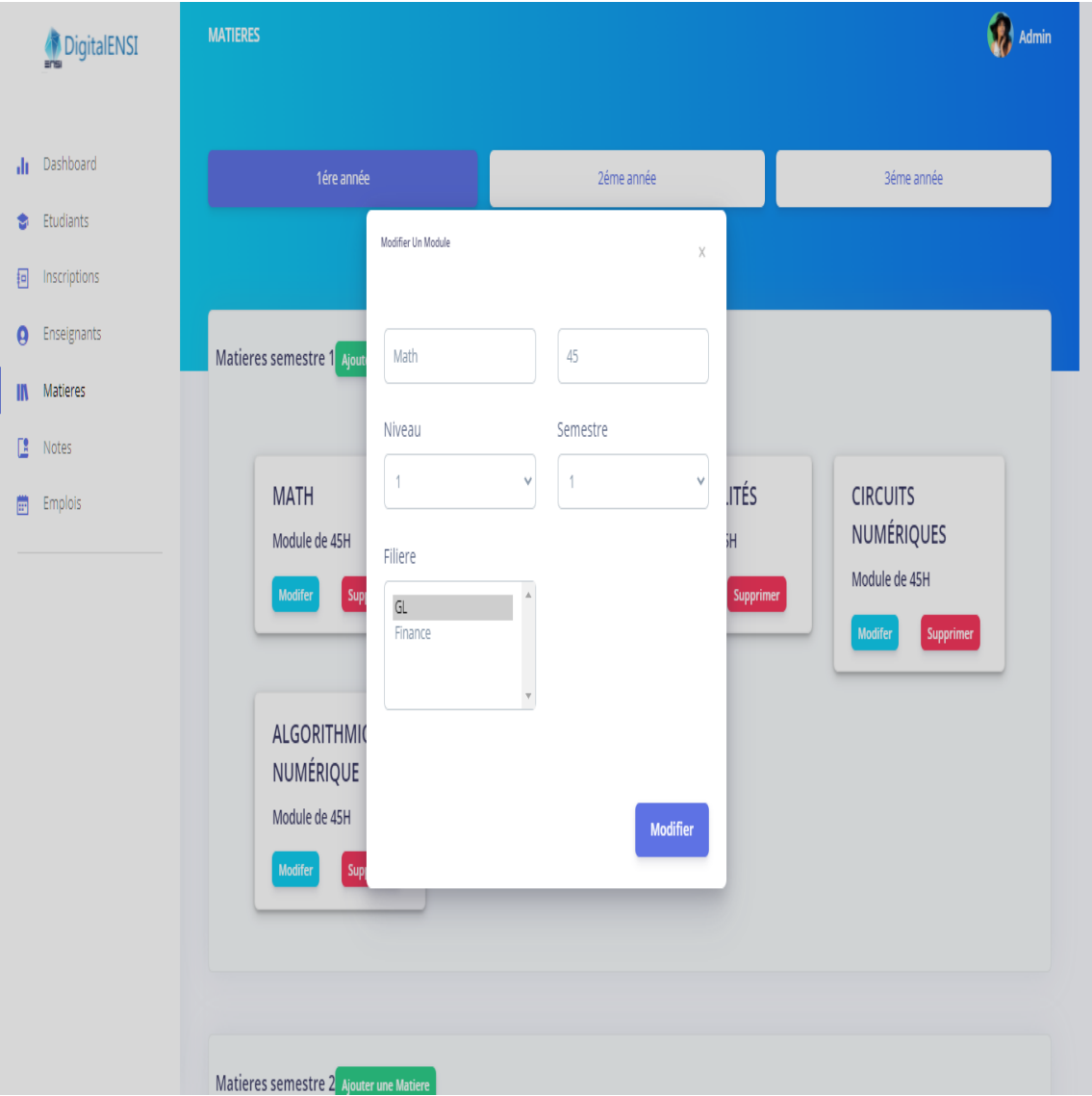


FIGURE III.15 – Interface de modification de matière

## Interface de consultation de matière

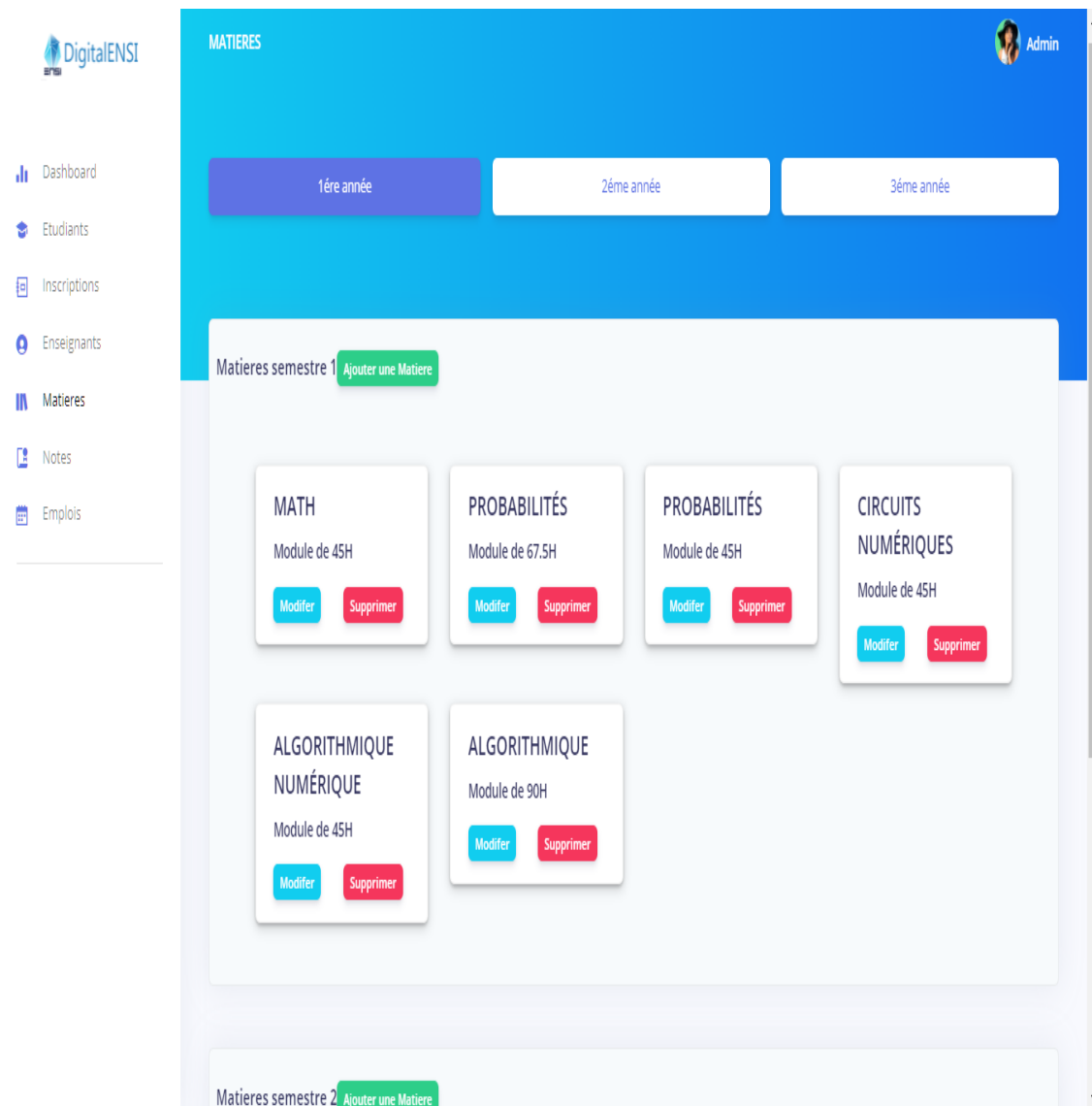


FIGURE III.16 – Interface de consultation de matière



# Interface d'ajout d'un étudiant

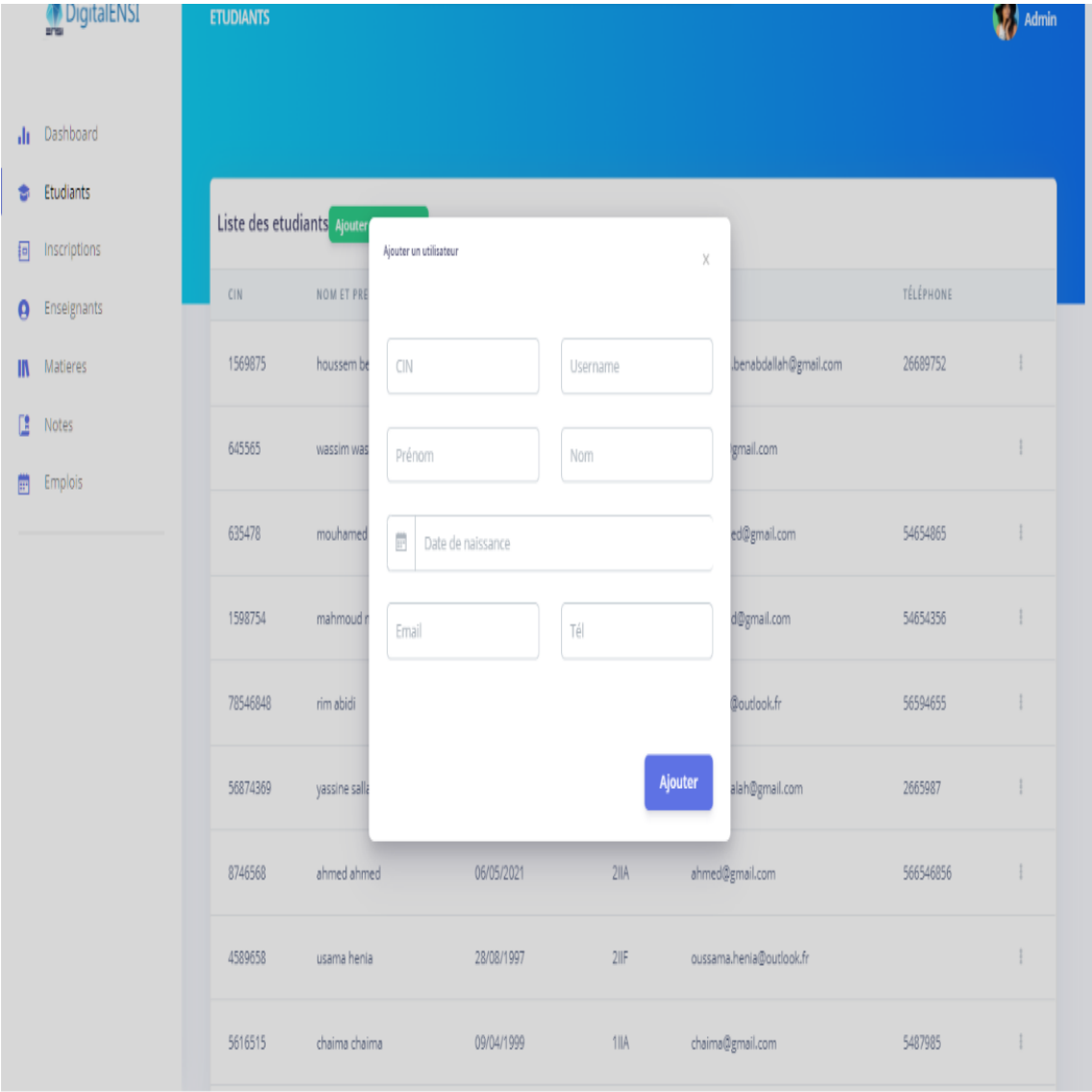


FIGURE III.17 – Interface d'ajout d'un étudiant

# Interface de modification d'un étudiant

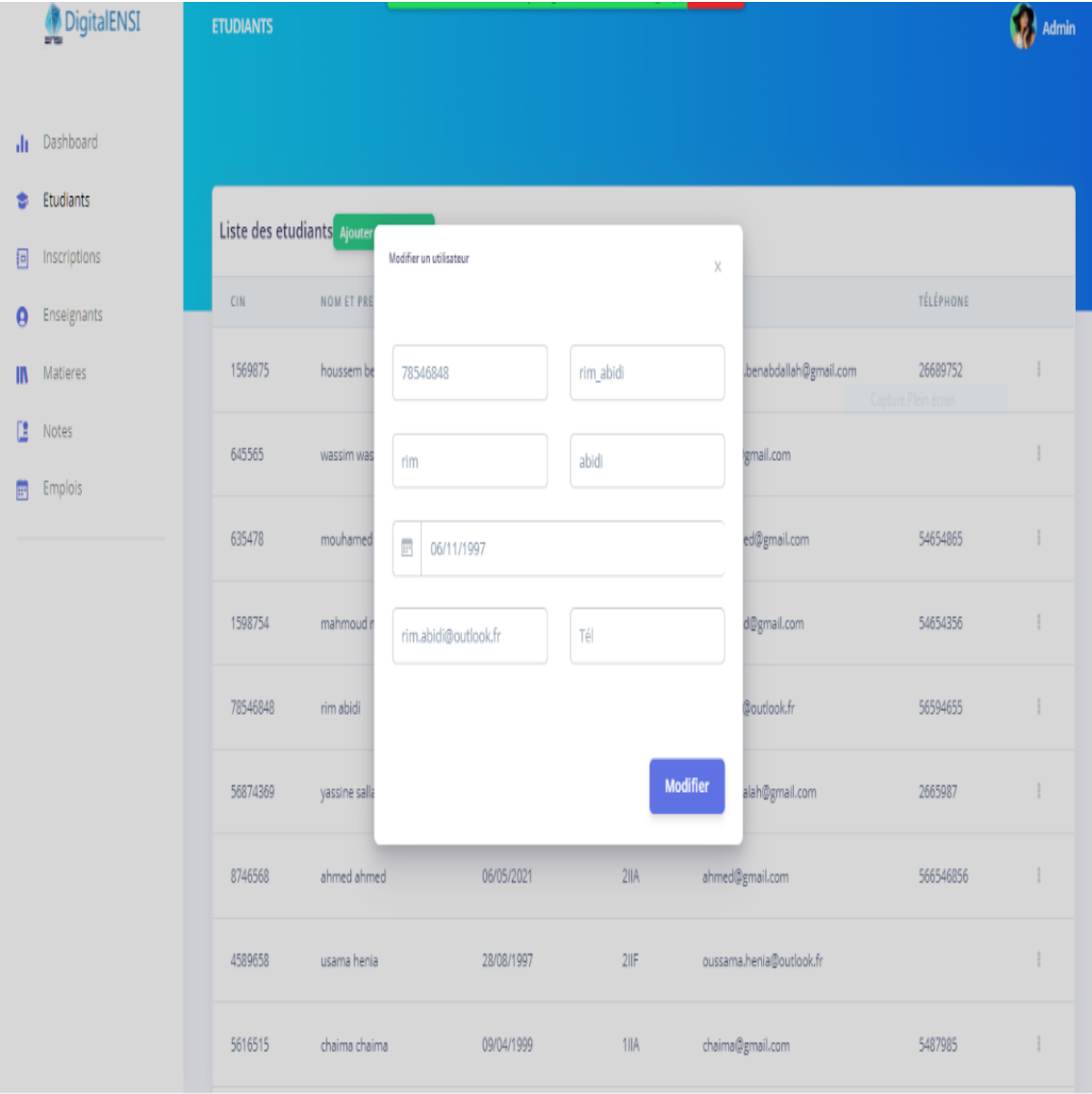


FIGURE III.18 – Interface de modification d'un étudiant

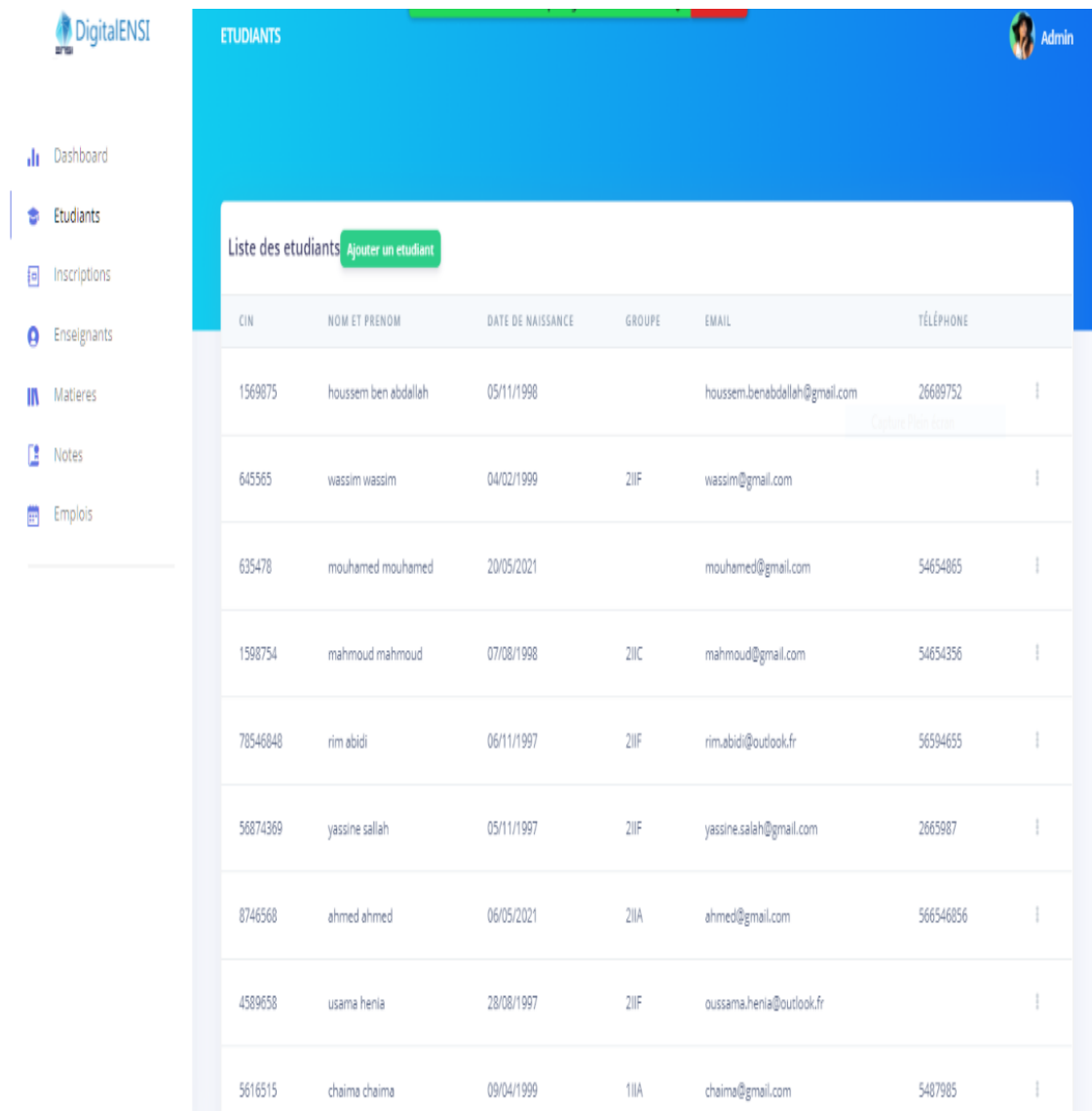
## Interface de suppression d'un étudiant

The screenshot displays the 'ETUDIANTS' management interface. On the left is a sidebar with navigation links: Dashboard, Etudiants (active), Inscriptions, Enseignants, Matieres, Notes, and Emplois. The main content area has a blue header with the 'DigitalENSI' logo and a user profile 'Admin'. Below the header, there's a section titled 'Liste des etudiants' with a green 'Ajouter un etudiant' button. A table lists ten students. A context menu is open over the second student, showing 'Modifier' and 'Supprimer' options.

CIN	NOM ET PRENOM	DATE DE NAISSANCE	GROUPE	EMAIL	TÉLÉPHONE
1569875	housem ben abdallah	05/11/1998		housem.benabdallah@gmail.com	26689752
645565	wassim wassim	04/02/1999	2IIF	wassim@gmail.com	
635478	mouhamed mouhamed	20/05/2021		mouhamed@gmail.com	54654865
1598754	mahmoud mahmoud	07/08/1998	2IIC	mahmoud@gmail.com	54654356
78546848	rim abidi	06/11/1997	2IIF	rim.abidi@outlook.fr	56594655
56874369	yassine salah	05/11/1997	2IIF	yassine.salah@gmail.com	2665987
8746568	ahmed ahmed	06/05/2021	2IIA	ahmed@gmail.com	566546856
4589658	usama henia	28/08/1997	2IIF	oussama.henia@outlook.fr	
5616515	chaima chaima	09/04/1999	1IIA	chaima@gmail.com	5487985

FIGURE III.19 – Interface de suppression d'un étudiant

## Interface de consultation de la liste des étudiants



The screenshot displays the 'ETUDIANTS' section of the DigitalENSI application. On the left, a sidebar contains navigation icons and labels: Dashboard, Etudiants (active), Inscriptions, Enseignants, Matieres, Notes, and Emplois. The main area features a blue header with the 'ETUDIANTS' title and an 'Admin' user profile. Below the header, a white box contains the title 'Liste des étudiants' and a green button labeled 'Ajouter un étudiant'. A table lists ten students with the following columns: CIN, NOM ET PRENOM, DATE DE NAISSANCE, GROUPE, EMAIL, and TÉLÉPHONE. Each row includes a vertical ellipsis icon for additional actions. A 'Capturer Photo' button is positioned above the first student's row.

CIN	NOM ET PRENOM	DATE DE NAISSANCE	GROUPE	EMAIL	TÉLÉPHONE
1569875	housem ben abdallah	05/11/1998		housem.benabdallah@gmail.com	26689752
645365	wassim wassim	04/02/1999	2IIF	wassim@gmail.com	
635478	mouhamed mouhamed	20/05/2021		mouhamed@gmail.com	54654865
1598754	mahmoud mahmoud	07/08/1998	2IIC	mahmoud@gmail.com	54654356
78546848	rim abidi	06/11/1997	2IIF	rim.abidi@outlook.fr	56594655
56874369	yassine salah	05/11/1997	2IIF	yassine.salah@gmail.com	2665987
8746568	ahmed ahmed	06/05/2021	2IIA	ahmed@gmail.com	566546856
4589658	usama henia	28/08/1997	2IIF	oussama.henia@outlook.fr	
5616515	chaima chaima	09/04/1999	1IIA	chaima@gmail.com	5487985

FIGURE III.20 – Interface de consultation de la liste des étudiants

Interface d’ajout d’un enseignant

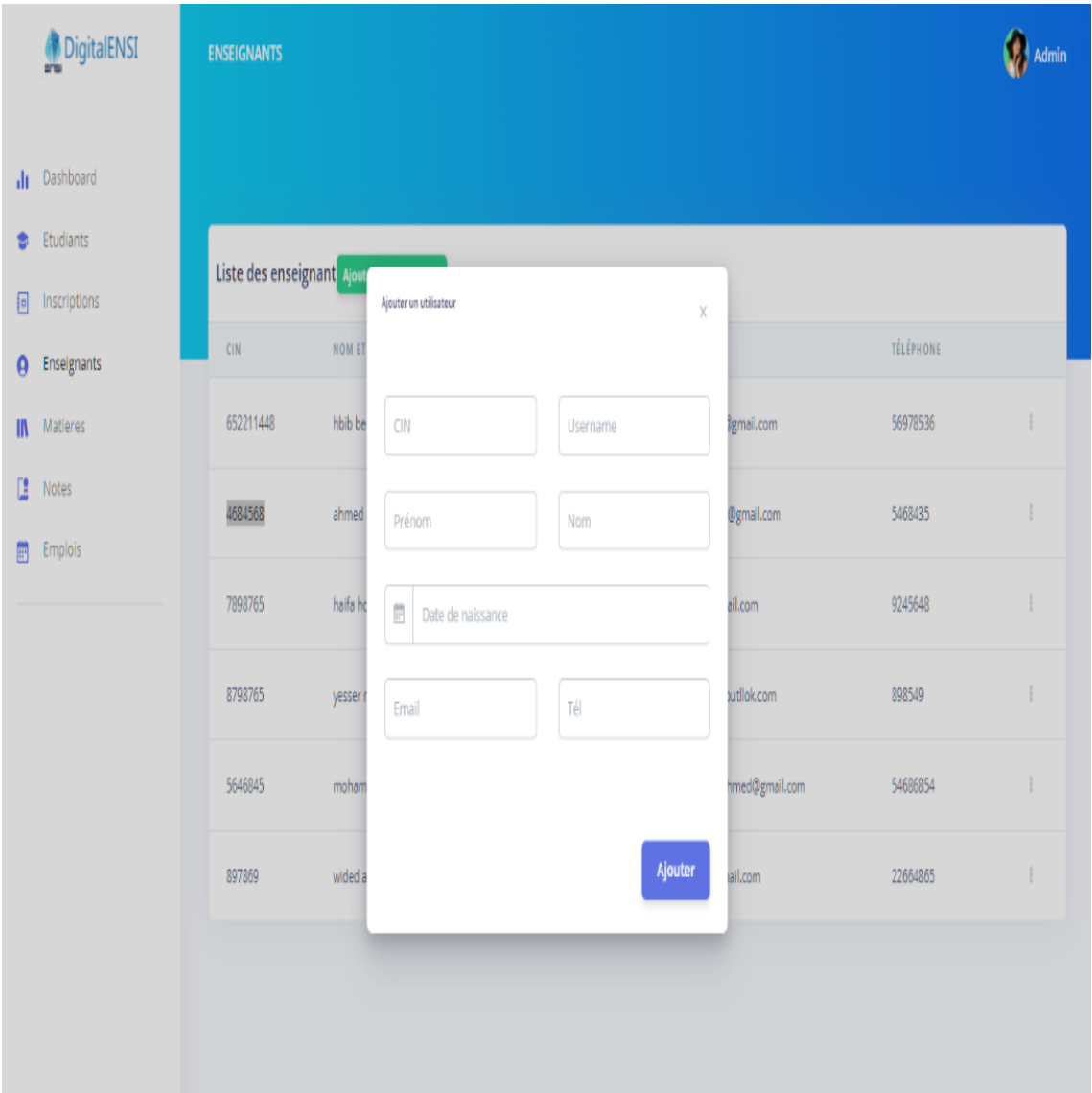


FIGURE III.21 – Interface d’ajout d’un enseignant

## Interface de suppression d'un enseignant

The screenshot displays the 'ENSEIGNANTS' management interface. On the left is a sidebar with navigation links: Dashboard, Etudiants, Inscriptions, Enseignants (active), Matieres, Notes, and Emplois. The main content area has a blue header with the 'ENSEIGNANTS' title and an 'Admin' user profile. Below the header, there's a section titled 'Liste des enseignants' with an 'Ajouter un enseignant' button. A table lists the following data:

CIN	NOM ET PRENOM	DATE DE NAISSANCE	EMAIL	TÉLÉPHONE
652211448	hibib ben hmida	03/06/1969	hibib.benhmda@gmail.com	56978536
4684568	ahmed nouisser	04/02/1960	ahmed.nouisser@gmail.com	54
7898765	haifa hosni	12/07/1975	haifa.hosni@gmail.com	9245648
8798765	yesser nasser	12/07/1975	yesser.nasser@outlook.com	898549
5646845	mohamed ben ahmed	14/07/1955	mohamed.benahmed@gmail.com	54686854
897869	wided ayari	16/01/1960	wided.ayari@gmail.com	22664865

A context menu is open over the second row (ahmed nouisser), showing two options: 'Modifier' and 'Supprimer'.

FIGURE III.22 – Interface de suppression d'un enseignant

## Interface de consultation de la liste des enseignants

The screenshot displays the DigitalENSI web application interface. On the left is a sidebar with navigation links: Dashboard, Etudiants, Inscriptions, Enseignants (selected), Matieres, Notes, and Emplois. The main content area has a blue header with the 'ENSEIGNANTS' title and an 'Admin' user profile. Below the header, there's a section titled 'Liste des enseignants' with an 'Ajouter un enseignant' button. A table lists the following teachers:

CIN	NOM ET PRENOM	DATE DE NAISSANCE	EMAIL	TÉLÉPHONE	
652211448	hbib ben hmida	03/06/1969	hbib.benhmda@gmail.com	56978536	⋮
4684568	ahmed nouisser	04/02/1960	ahmed.nouisser@gmail.com	5468435	⋮
7898765	haifa hosni	12/07/1975	haifa.hosni@gmail.com	9245648	⋮
8798765	yesser nasser	12/07/1975	yesser.nasser@outlook.com	898549	⋮
5646845	mohamed ben ahmed	14/07/1955	mohamed.benahmed@gmail.com	54686854	⋮
897869	wided ayari	16/01/1960	wided.ayari@gmail.com	22664865	⋮

FIGURE III.23 – Interface de consultation de la liste des enseignants

Interface d’ajout d’une inscription

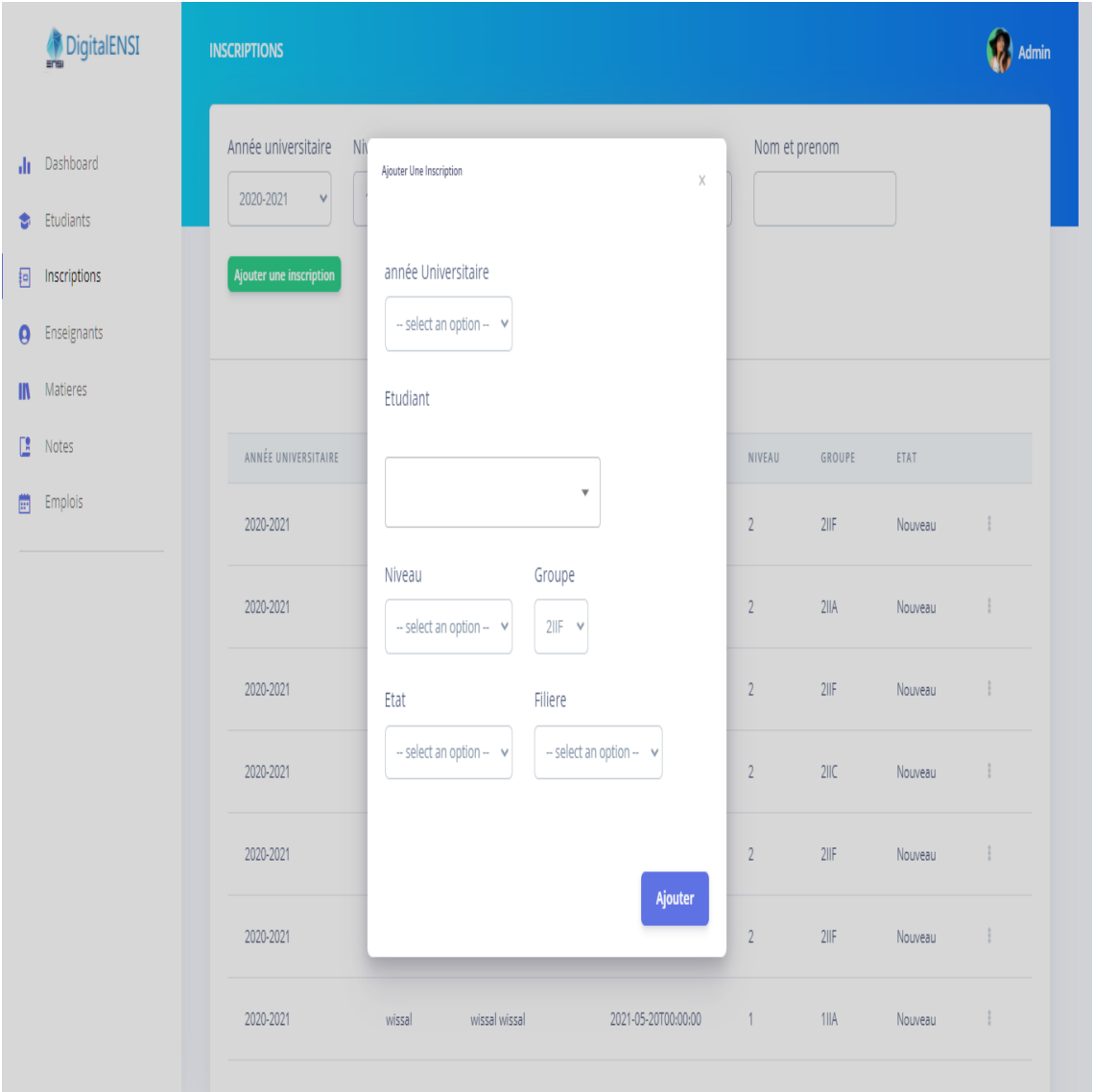


FIGURE III.24 – Interface d’ajout d’une inscription



## Interface de modification d'une inscription

The screenshot displays the 'DigitalENSI' application interface. The top navigation bar is blue with the 'DigitalENSI' logo on the left and an 'Admin' user profile on the right. A sidebar on the left lists various menu items: Dashboard, Etudiants, Inscriptions (highlighted), Enseignants, Matieres, Notes, and Emplois. The main content area is titled 'INSCRIPTIONS' and features a table of student registrations. Overlaid on this is a 'Modifier une inscription' modal form. The modal form contains the following fields:

- année Universitaire:** A dropdown menu currently showing '2020-2021'.
- Etudiant:** A search bar with the text 'rim abidi' entered.
- Niveau:** A dropdown menu currently showing '2'.
- Groupe:** A dropdown menu currently showing '2IIF'.
- Etat:** A dropdown menu currently showing 'Nouveau'.
- Filiere:** A dropdown menu currently showing '-- select an option --'.

A blue 'Modifier' button is located at the bottom right of the modal form. The background table shows columns for 'ANNÉE UNIVERSITAIRE', 'NIVEAU', 'GROUPE', 'ETAT', 'CIN', and 'Nom et prenom'. The table contains several rows of data, including one for 'rim abidi' with CIN '1997-11-06T00:00:00'.

FIGURE III.25 – Interface de modification d'une inscription

## Interface de suppression d'une inscription

**DigitalENSI**

**INSCRIPTIONS** Admin

Année universitaire Niveau Groupe Etat CIN Nom et prenom

2020-2021 1 1 Nouveau

Ajouter une inscription

ANNÉE UNIVERSITAIRE	CIN	NOM ET PRENOM	DATE DE NAISSANCE	NIVEAU	GROUPE	ETAT	
2020-2021	4589658	usama henia	1997-08-28T00:00:00	2	2IIF	Nouveau	⋮
2020-2021	8746568	ahmed ahmed	2021-05-06T00:00:00	2	2IIA		Modifier Supprimer
2020-2021	645565	wassim wassim	1999-02-04T00:00:00	2	2IIF	Nouveau	⋮
2020-2021	1598754	mahmoud mahmoud	1998-08-07T00:00:00	2	2IIC	Nouveau	⋮
2020-2021	56874369	yassine salah	1997-11-05T00:00:00	2	2IIF	Nouveau	⋮
2020-2021	78546848	rim abidi	1997-11-06T00:00:00	2	2IIF	Nouveau	⋮
2020-2021	wissal	wissal wissal	2021-05-20T00:00:00	1	1IIA	Nouveau	⋮

FIGURE III.26 – Interface de suppression d'une inscription

## Interface de consultation des inscriptions

ANNÉE UNIVERSITAIRE	CIN	NOM ET PRENOM	DATE DE NAISSANCE	NIVEAU	GROUPE	ETAT	
2020-2021	4589658	usama henia	1997-08-28T00:00:00	2	2IIF	Nouveau	
2020-2021	8746568	ahmed ahmed	2021-05-06T00:00:00	2	2IIA	Nouveau	
2020-2021	645565	wassim wassim	1999-02-04T00:00:00	2	2IIF	Nouveau	
2020-2021	1598754	mahmoud mahmoud	1998-08-07T00:00:00	2	2IIC	Nouveau	
2020-2021	56874369	yassine salah	1997-11-05T00:00:00	2	2IIF	Nouveau	
2020-2021	78546848	rim abidi	1997-11-06T00:00:00	2	2IIF	Nouveau	
2020-2021	wissal	wissal wissal	2021-05-20T00:00:00	1	1IIA	Nouveau	

FIGURE III.27 – Interface de consultation des inscriptions

## V Test

Le test d'un logiciel est un processus qui vise à assurer un bon fonctionnement du système avec une comparaison entre les comportements attendus et les résultats obtenus.

Avant de finir un Sprint nous avons opté pour le test des fonctionnalités du module.

Ensuite nous avons passé à la validation des fonctionnalités avec le Product Owner. Pour cela, nous présentons dans le tableau suivant l'ensemble des cas de tests relatifs au sprint 1.

Cas de test	Comportement attendu	Résultats
Test de l'authentification	-L'utilisateur saisit l'email et le mot de passe et clique sur le bouton se connecter pour accéder à la plateforme.	Conforme
Test de gestion des matières.	-L'administrateur a la possibilité d'ajouter une matière, de la consulter, de la modifier ou de la supprimer.	Conforme
Test de gestion des profils des étudiants.	-L'administrateur peut ajouter un étudiant, consulter ses données, modifier ses données ou supprimer un étudiant. L'étudiant peut consulter son profil.	Conforme
Test de gestion des profils des enseignants.	-L'administrateur peut ajouter un enseignant, consulter ses données, modifier ses données ou supprimer un enseignant. L'enseignant peut consulter son profil.	Conforme
Test de gestion des inscriptions.	-L'administrateur a la possibilité d'ajouter une inscription relative à un étudiant, de la consulter, de la modifier ou de la supprimer.	Conforme

Tests du sprint 1

## Conclusion

Dans ce chapitre, nous avons opté pour la présentation du premier sprint. Pour cela, nous sommes passés par quatre étapes qui sont l'analyse, la conception ainsi que la réalisation et les tests. Dans le chapitre qui suit nous présentons le deuxième sprint.

## CHAPITRE IV

# ÉTUDE ET RÉALISATION DU SPRINT 2

### Introduction

Après avoir terminé le premier sprint de notre système, nous passons maintenant au deuxième sprint. En effet, les méthodes agiles, et particulièrement Scrum, sont toujours caractérisées par un rythme qui est régulier. Au cours de ce chapitre qui représente l'étude et la réalisation du deuxième sprint, nous allons détailler l'application en présentant les différents cas d'utilisation proposés dans le backlog du sprint deux et qui est présenté dans ce chapitre.

Ensuite nous présentons la conception et nous achevons par la partie réalisation et tests.

### I Backlog du sprint

Le tableau suivant présente le backlog du deuxième sprint.

ID	Histoire	Estimation
6	Gérer les emplois de temps	7 jours
7	Gérer les notes des étudiants	8 jours
8	Gérer les absences	8 jours
9	Visualiser les statistiques	7 jours

TABLE IV.1 – Backlog du sprint 2

## II Spécification fonctionnelle

Dans cette section, nous présentons les diagrammes de cas d'utilisation et la description textuelle de chacun d'entre eux.

### II.1 Diagrammes de cas d'utilisation

La figure suivante représente le diagramme de cas d'utilisation global du sprint 2.

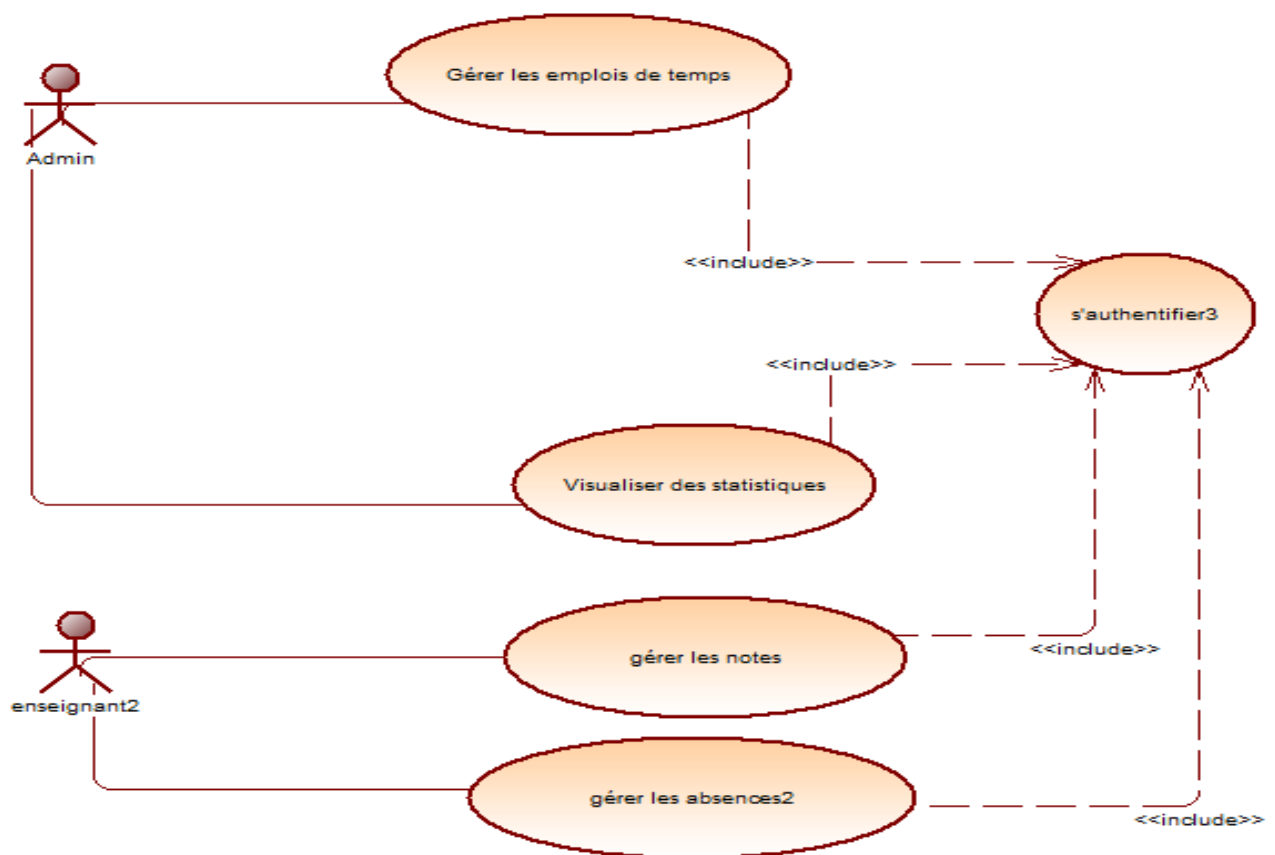


FIGURE IV.1 – Diagramme de cas d'utilisation global du sprint 2

La figure 4.2 décrit le diagramme de cas d'utilisation de l'opération gérer les emplois de temps.

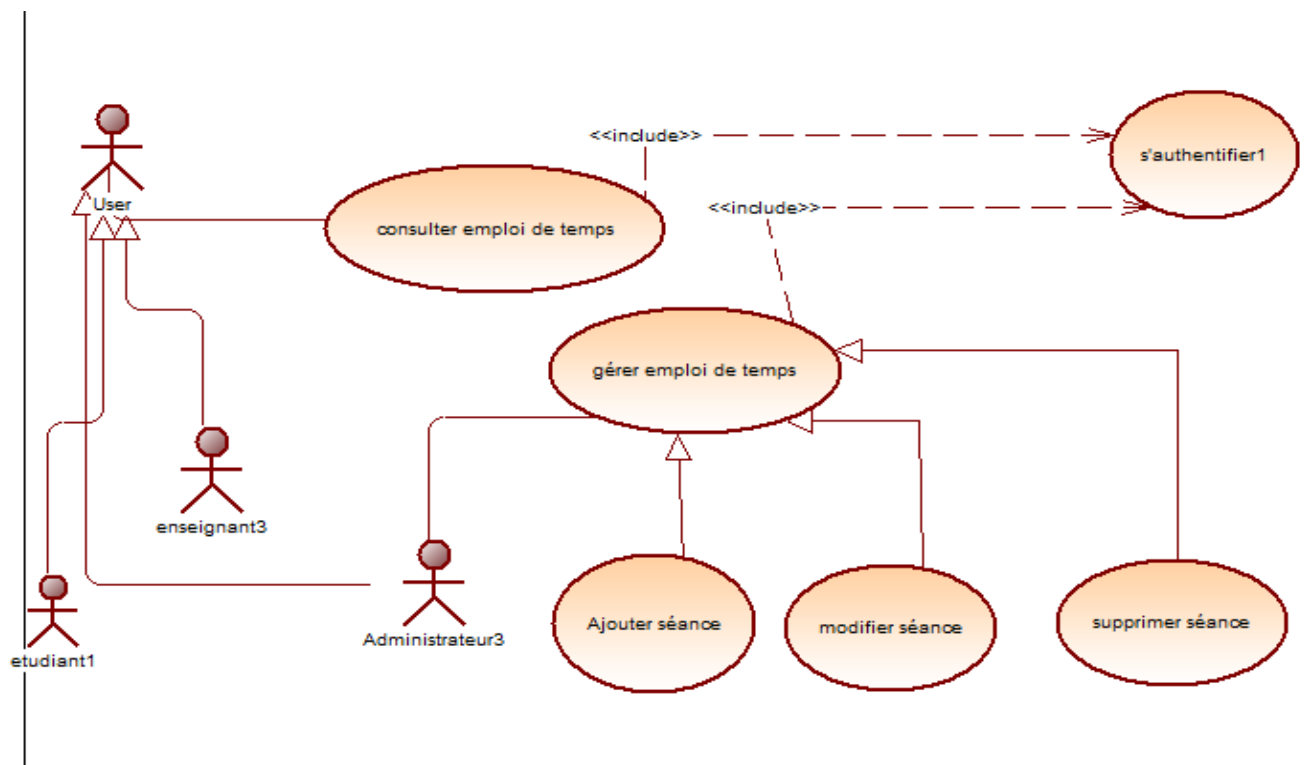


FIGURE IV.2 – Diagramme de cas d'utilisation de l'opération gérer les emplois de temps.

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer les notes.

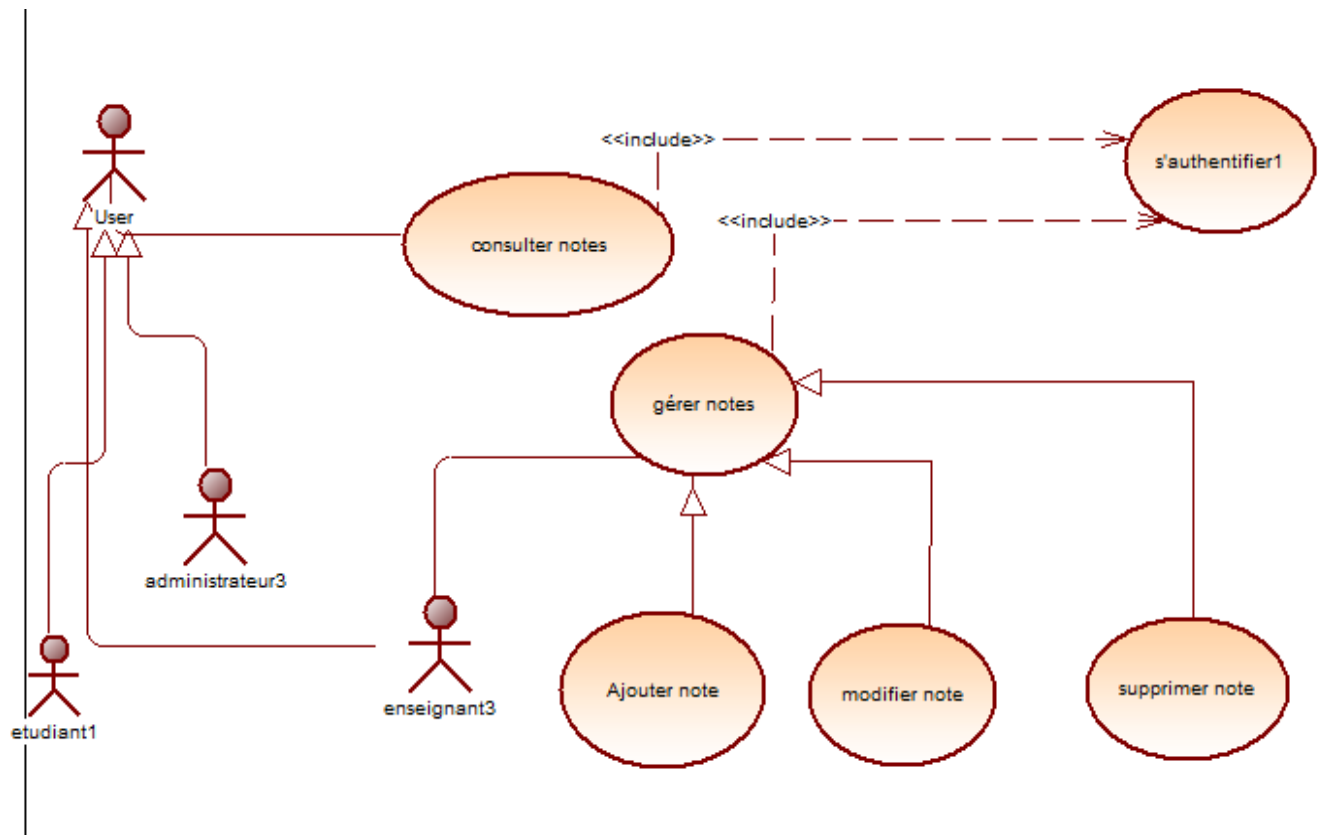


FIGURE IV.3 – Diagramme de cas d'utilisation de l'opération gérer les notes.

La figure suivante décrit le diagramme de cas d'utilisation de l'opération gérer les absences.

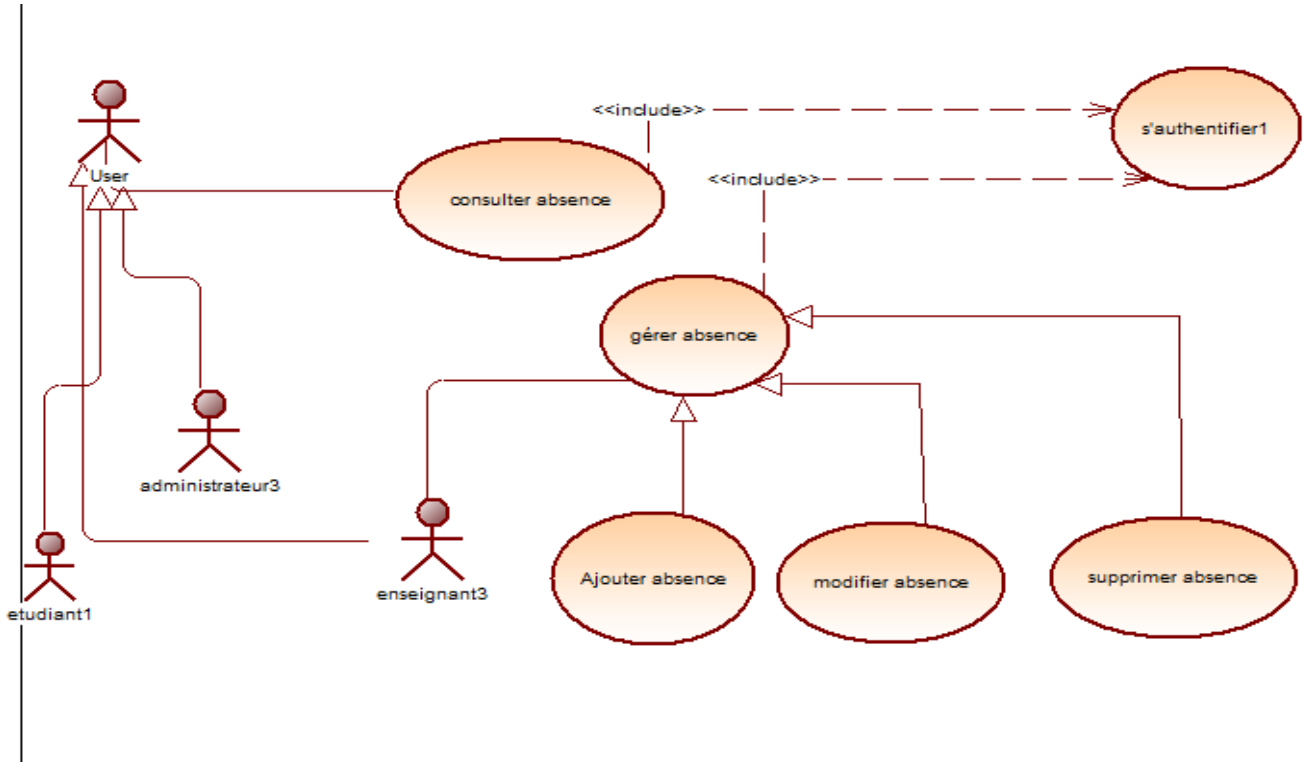


FIGURE IV.4 – Diagramme de cas d'utilisation de l'opération gérer les absences

La figure suivante décrit le diagramme de cas d'utilisation de l'opération visualiser les statistiques.

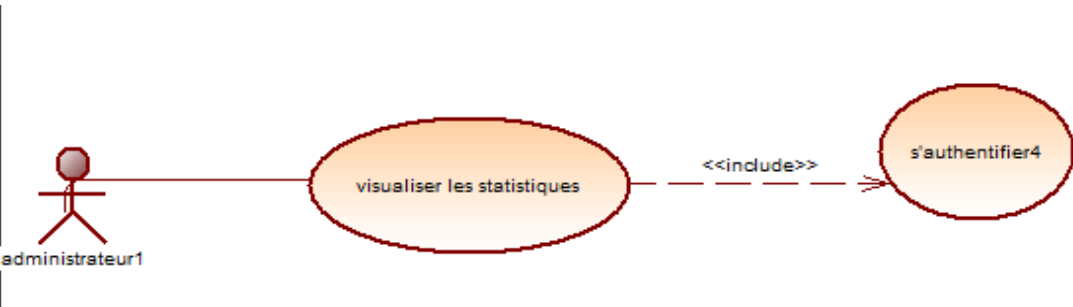


FIGURE IV.5 – Diagramme de cas d'utilisation de l'opération visualiser les statistiques.

## II.2 Descriptions textuelles des cas d'utilisation

- Raffinement de l'opération « Gérer les emplois de temps »

Description textuelle du cas d'utilisation « Ajouter séance »

Le tableau suivant décrit la description textuelle du cas d'utilisation « Ajouter séance » .



Titre	Ajouter séance
Acteur principal	Administrateur
Résumé	A travers ce cas, l'administrateur peut ajouter des séances à chaque emploi de temps par niveau et par filière.
Pré-condition	Utilisateur authentifié, Séance non ajoutée
Scénario principal	1. cliquer sur la rubrique emplois 2. choisir le groupe 3. choisir le module et l'enseignant de la séance de cours 4. choisir le créneau horaire relatif à la séance 5. Cliquer sur Ajouter
Post-condition	Séance ajoutée
Exception	Un message d'erreur est affiché si la séance n'a pas été ajoutée

Description textuelle du cas d'utilisation « Ajouter séance »

Description textuelle du cas d'utilisation « Modifier séance »

Le tableau suivant décrit la description textuelle du cas d'utilisation « Modifier séance »

Titre	Modifier séance
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut modifier les données d'une séance de cours dans l'emploi de temps.
Pré-conditions	Utilisateur authentifié, Séance non modifiée.
Scénario principal	1. choisir la séance à modifier 2. cliquer sur l'icône Modifier 3. saisir les données à mettre à jour 4. cliquer sur le bouton Modifier
Post-condition	Séance modifiée.
Exception	Un message d'erreur est affiché le cas échéant

Description textuelle du cas d'utilisation « Modifier séance »

Description textuelle du cas d'utilisation « Supprimer séance »

Le tableau suivant décrit la description textuelle du cas d'utilisation « supprimer séance »

Titre	supprimer séance
Acteur principal	administrateur
Résumé	A travers ce cas, l'administrateur peut supprimer une séance de cours dans l'emploi de temps.
Pré-conditions	Utilisateur authentifié, Séance non supprimée.
Scénario principal	1. Choisir la séance à supprimer 2. Cliquer sur l'icône supprimer 3. Cliquer sur le bouton supprimer
Post-condition	Séance supprimée.
Exception	Un message d'erreur est affiché le cas échéant

Description textuelle du cas d'utilisation « Supprimer séance »

Description textuelle du cas d'utilisation « consulter emploi de temps »

Le tableau suivant décrit la description textuelle du cas d'utilisation « consulter emploi de temps »

Titre	consulter emploi de temps
Acteur principal	Etudiant
Résumé	A travers ce cas, l'étudiant peut consulter son emploi de temps.
Pré-conditions	Utilisateur authentifié, Emploi de temps non consulté.
Scénario principal	1. Choisir la rubrique Emploi 2. Consulter l'emploi de temps
Post-condition	Emploi de temps consulté.

Description textuelle du cas d'utilisation « Consulter emploi de temps »

### Remarque :

Nous avons choisi de ne pas présenter les descriptions textuelles des opérations gestion des notes et gestion des absences car elles ressemblent à celles relatives à la gestion de l'emploi de temps. La description textuelle du cas d'utilisation visualiser les statistiques est semblable à celle de consulter emploi de temps, l'acteur principal est l'administrateur.

## III Conception détaillée

Dans cette partie, nous présentons le diagramme de classes final de tout le système , la conception de la base de données à travers un modèle entité-association ainsi que les diagrammes de séquences du deuxième sprint.

### III.1 Diagramme de classes final

Après avoir entamé le sprint 2, nous présentons le diagramme de classes final qui comporte toutes les entités de notre système.

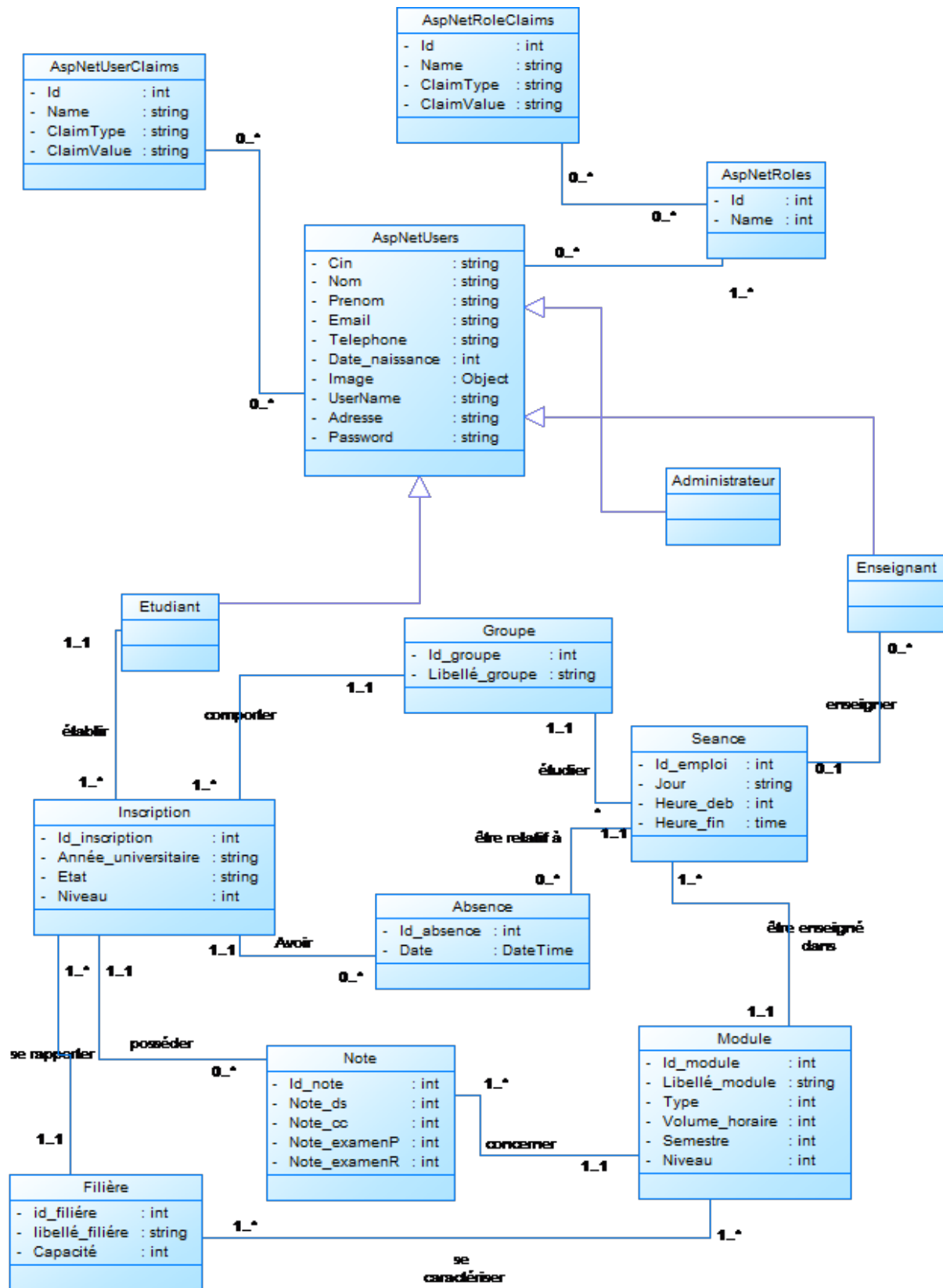


FIGURE IV.6 – Diagramme de classes global du système

## III.2 Modèle entité-association

Le modèle entité-association est une description graphique afin de représenter des modèles de données sous formes de diagramme contenant des entités et des associations.

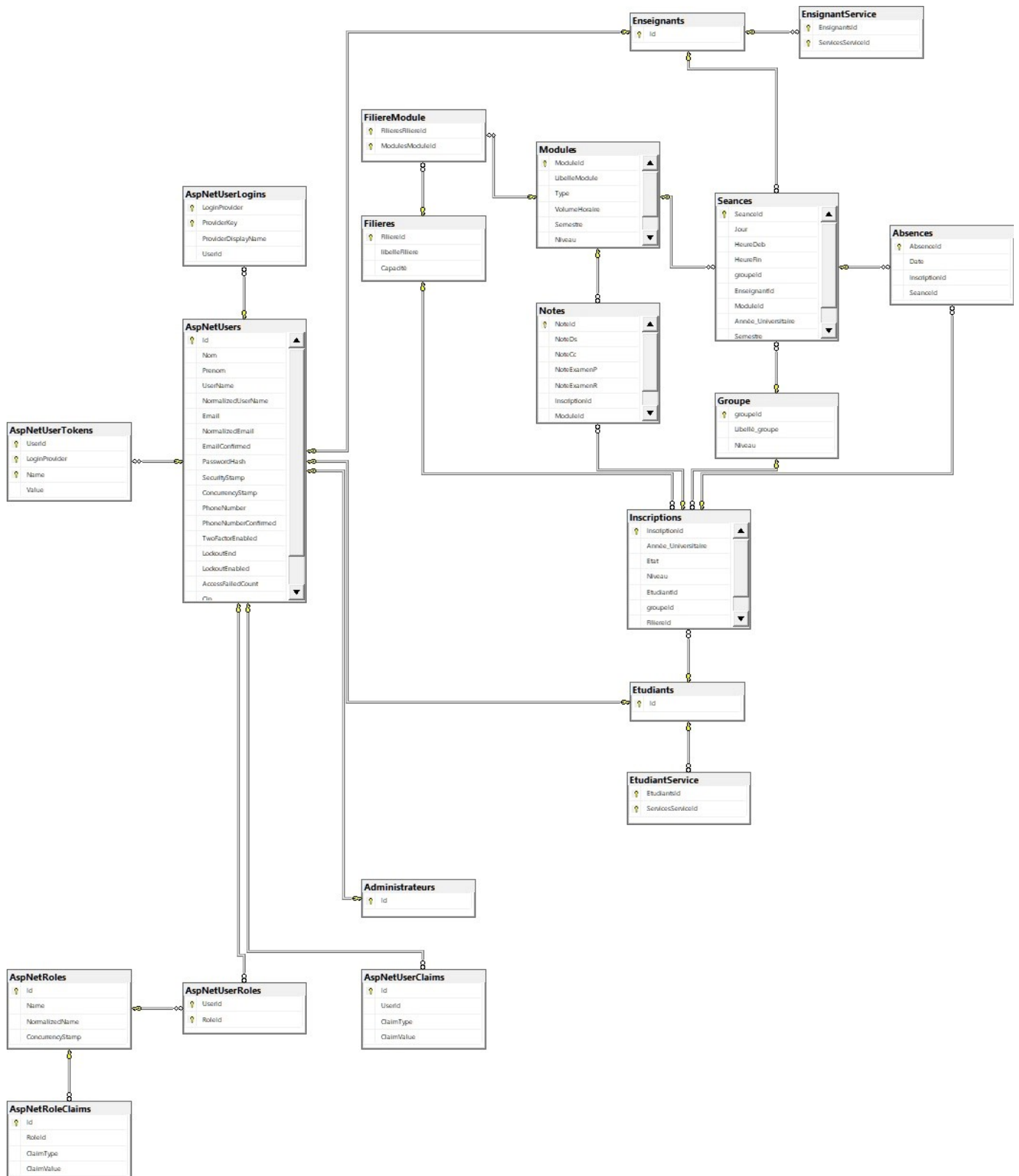


FIGURE IV.7 – Modèle entité-association

### III.3 Diagrammes de séquence détaillés

Après avoir terminé la description textuelle des cas d'utilisation relatifs au deuxième sprint, nous passons à la présentation des diagrammes de séquence relatifs à ce sprint.

Diagramme de séquence du cas d'utilisation « Ajouter séance »

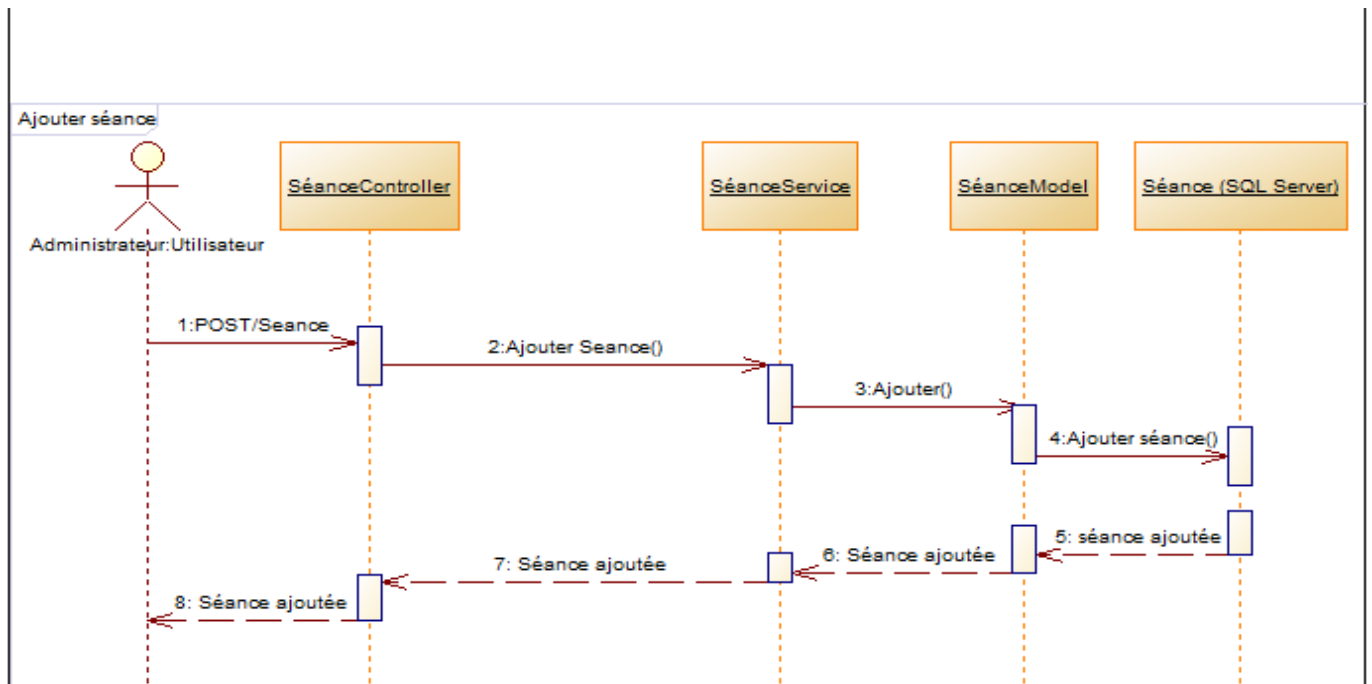


FIGURE IV.8 – Diagramme de séquence du cas d'utilisation « Ajouter séance »

Le scénario est le suivant :

- L'administrateur envoie une requête de type POST au contrôleur en cliquant sur le bouton Ajouter
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la création de la séance
- Les informations de la séance sont enregistrées automatiquement dans la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé donc de l'ajout de la séance dans l'emploi de temps.

Diagramme de séquence du cas d'utilisation « Modifier séance »

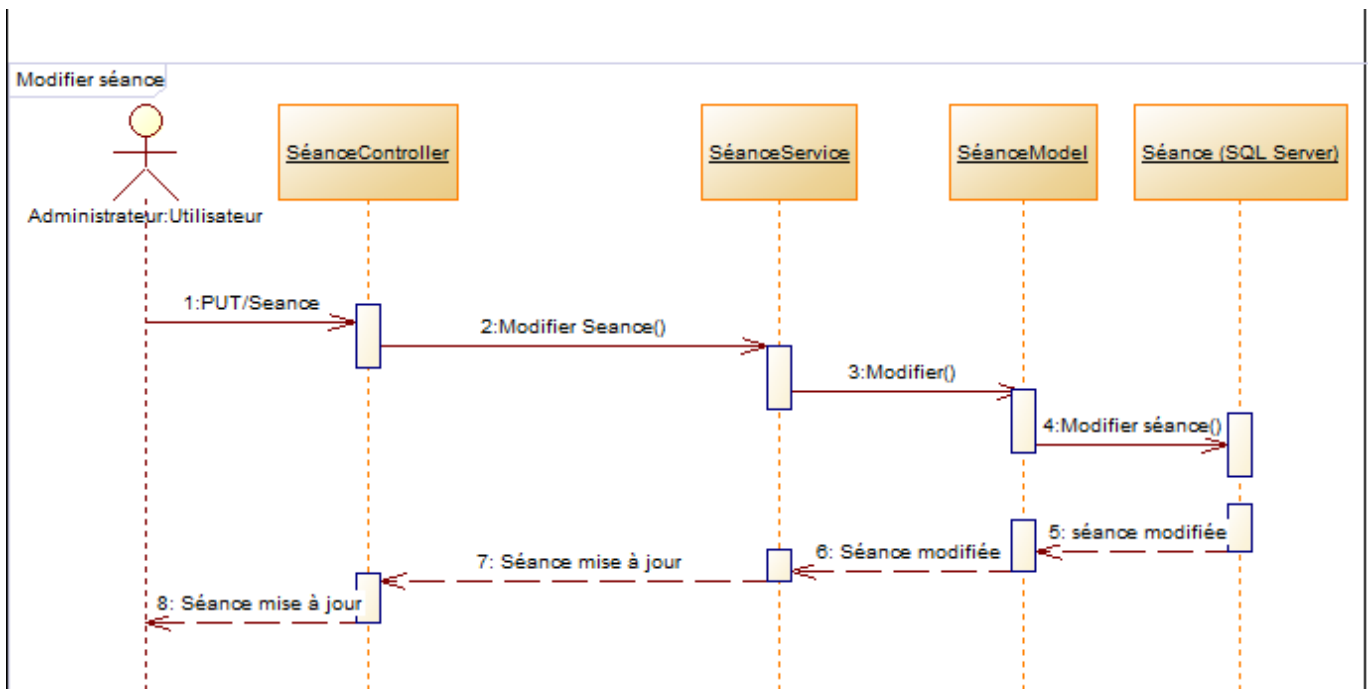


FIGURE IV.9 – Diagramme de séquence du cas d'utilisation « Modifier séance »

Le scénario est le suivant :

- L'administrateur envoie une requête de type PUT au contrôleur en cliquant sur le l'icône Modifier
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la modification des données de la séance
- Les informations de la séance sont enregistrées automatiquement dans la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé donc de la modification de la séance.

Diagramme de séquence du cas d'utilisation « Supprimer séance »

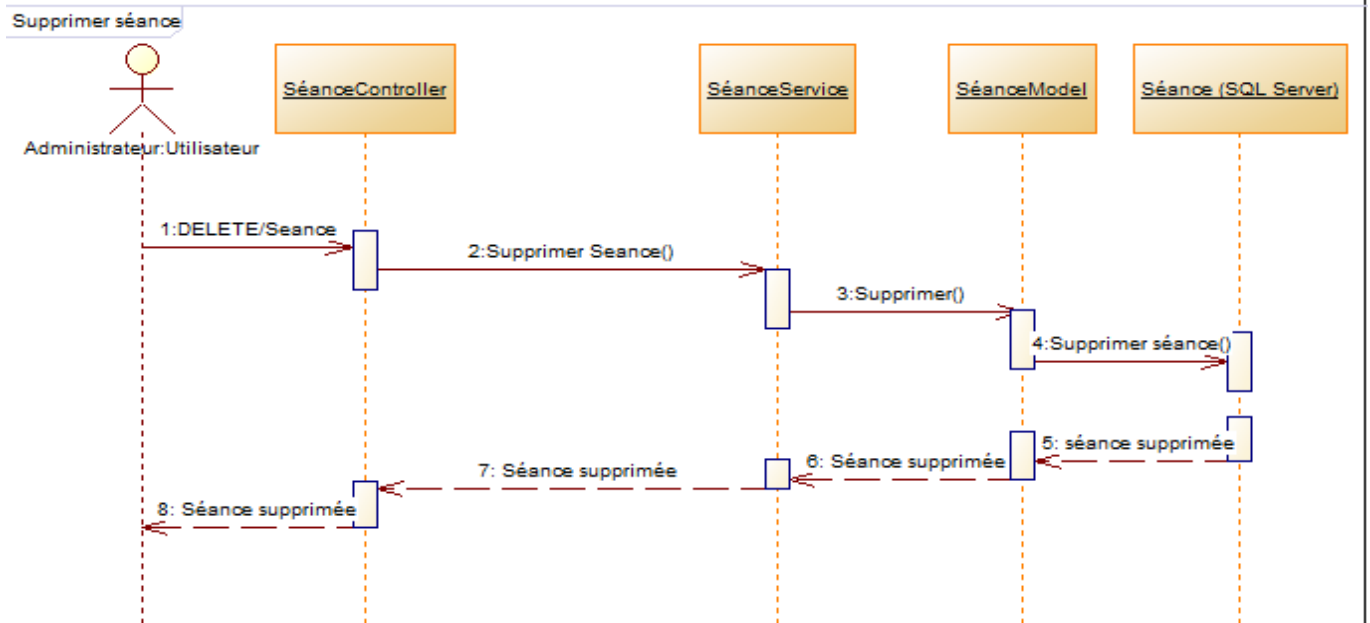


FIGURE IV.10 – Diagramme de séquence du cas d'utilisation « Supprimer séance »

Le scénario est le suivant :

- L'administrateur envoie une requête de type DELETE au contrôleur en cliquant sur le l'icône Supprimer
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la suppression de la séance
- La séance est supprimée de la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'administrateur est informé donc de la suppression de la séance.

Diagramme de séquence du cas d'utilisation « Consulter emploi de temps »



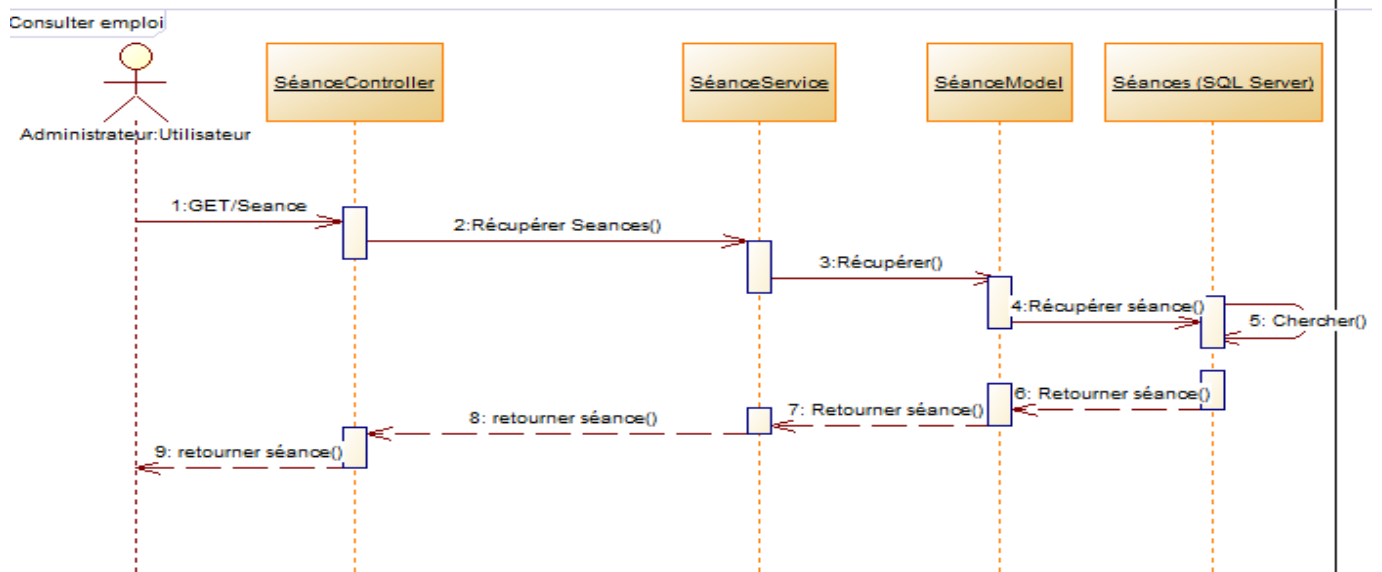


FIGURE IV.11 – Diagramme de séquence du cas d'utilisation « Consulter emploi de temps »

Le scénario est le suivant :

- L'administrateur envoie une requête de type GET au contrôleur en cliquant sur la rubrique Emploi
- Le contrôleur envoie la requête à la couche services
- La couche services présente le service attendu qui est la récupération de la séance
- La séance est récupérée de la base de données
- En suivant l'ordre des réponses entre les différentes couches, l'étudiant récupère les séances de cours et peut donc consulter son emploi de temps.

**Remarque :** Nous avons choisi de ne pas présenter les diagrammes de séquence relatifs à la gestion des notes, des absences car ils ressemblent au diagramme de séquence de l'opération de gestion de l'emploi de temps. Le diagramme de séquence de l'opération de visualisation des statistiques est semblable à celui de consultation de l'emploi de temps.

## IV Réalisation

Dans cette partie, nous présentons des captures d'écran des interfaces relatives au sprint 2.

## Interface de consultation des emploi de temps

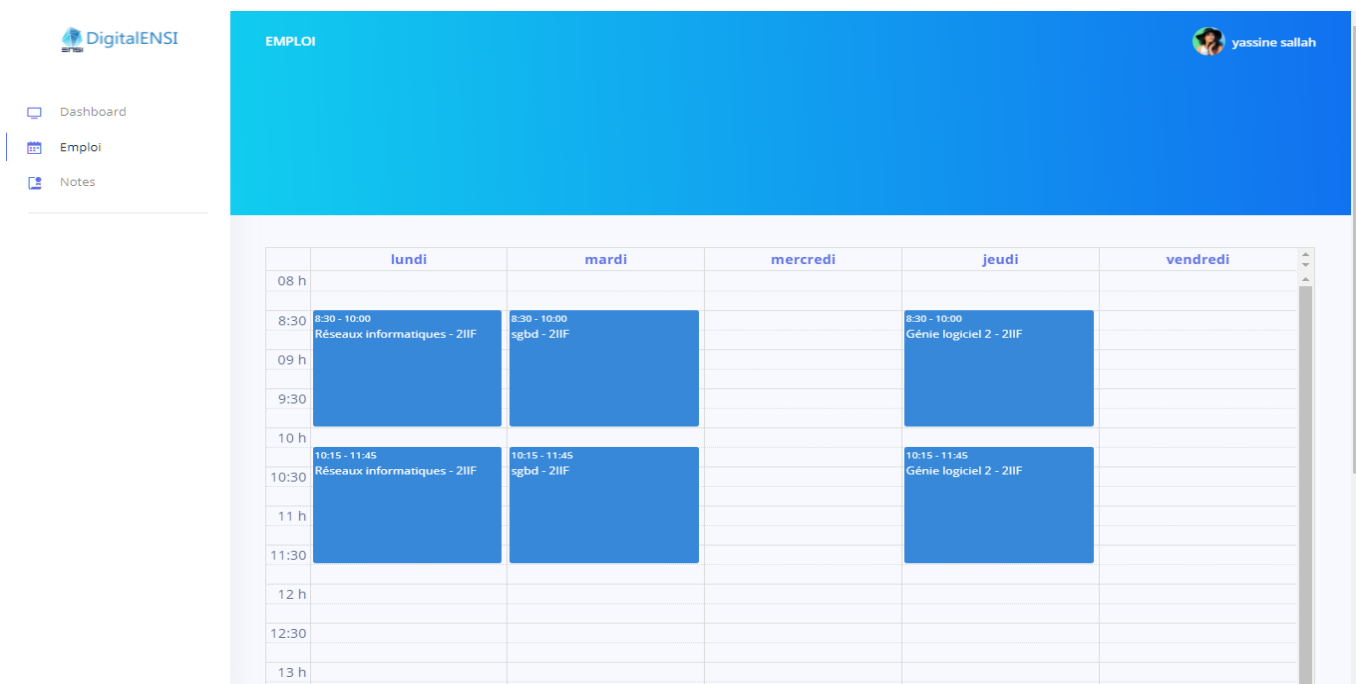


FIGURE IV.12 – Interface de consultation des emploi de temps

## Interface d'ajout d'une séance

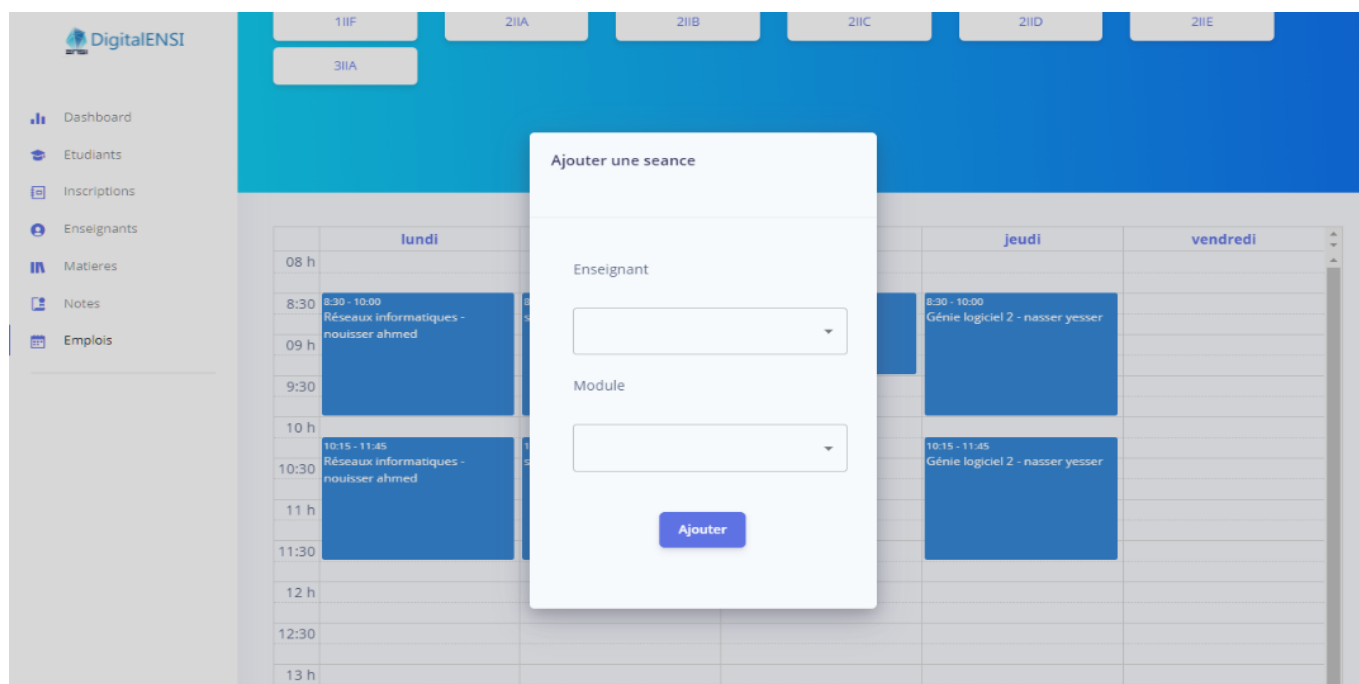


FIGURE IV.13 – Interface d'ajout d'une séance

## Interface de modification d'une note

Semestre: 2

Matiere:

Groupe:

Ajouter une note

Note DS: 0

Note CC: 0

NOTE EXAMEN PRINCIPAL: 0

NOTE EXAMEN PRINCIPAL: MOYENNE RATRAPPAGE

Modifier

NOM ET PRENOM	NOTE DS	NOTE CC	NOTE EXAMEN PRINCIPAL	MOYENNE PRINCIPAL	NOTE EXAMEN RATRAPPAGE	MOYENNE RATRAPPAGE
henia usama	0					
wassim wassim	0					
sallah yassine	0					
abidi rim	0					

FIGURE IV.14 – Interface de modification d'une note

## Interface de consultation des notes

Semestre: 2

Matiere: Réseaux informatiques

Groupe: 2IIF

NOM ET PRENOM	NOTE DS	NOTE CC	NOTE EXAMEN PRINCIPAL	MOYENNE PRINCIPAL	NOTE EXAMEN RATRAPPAGE	MOYENNE RATRAPPAGE
henia usama	0	0	0	0		
wassim wassim	0	0	0	0		
sallah yassine	0	0	0	0		
abidi rim	0	0	0	0		

FIGURE IV.15 – Interface de consultation des notes

## Interface de gestion des absences

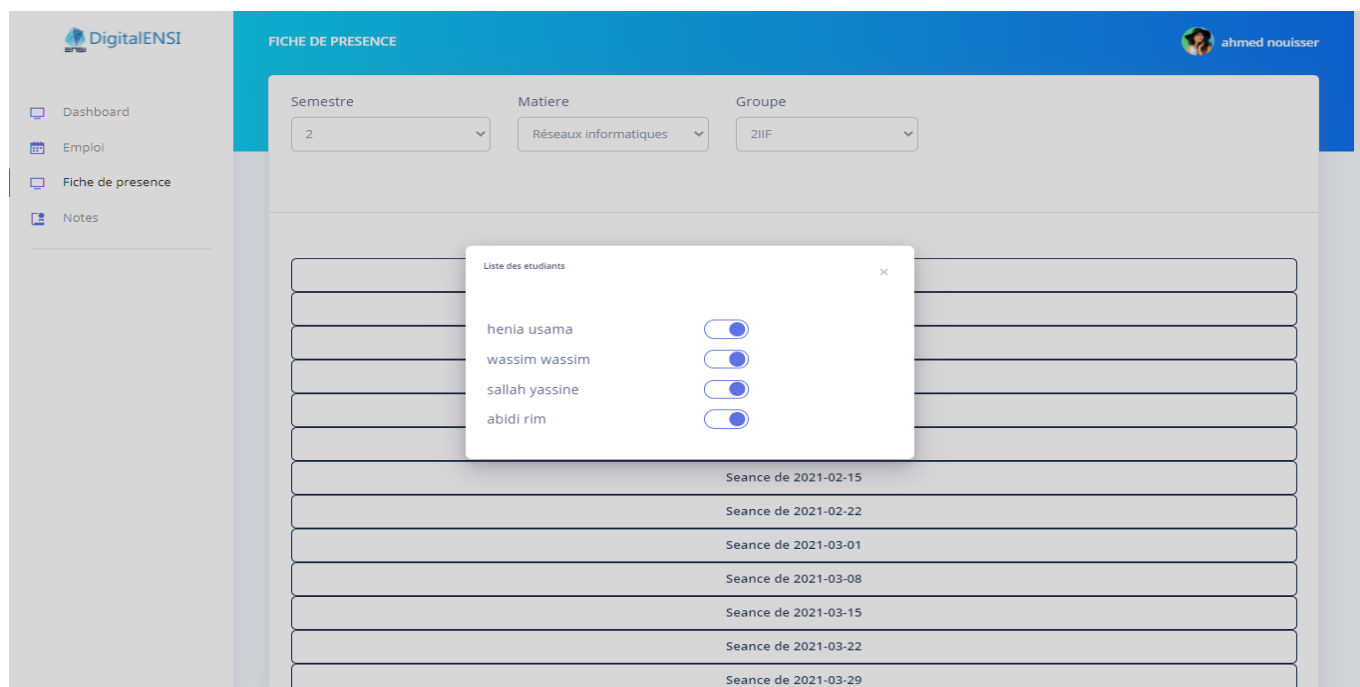


FIGURE IV.16 – Interface de gestion des absences

## Interface de visualisation des statistiques



FIGURE IV.17 – Interface de visualisation des statistiques

## V Tests

Le tableau suivant présente les cas de tests relatifs au deuxième sprint.

Cas de test	Comportement attendu	Résultats
Test de gestion des emplois de temps	-L'administrateur peut ajouter, modifier, supprimer des séances de cours. Il peut aussi visualiser n'importe quel emploi de temps. Chaque étudiant peut visualiser son emploi de temps relatif à son groupe. Chaque enseignant peut de même consulter son emploi de temps.	Conforme
Test de gestion des notes.	-L'enseignant peut ajouter, supprimer, modifier et consulter les notes de ses étudiants. L'étudiant peut consulter ses notes sans pouvoir consulter les notes des autres étudiants. L'administrateur peut consulter n'importe quelle note de n'importe quel étudiant.	Conforme
Test de gestion des absences	-L'enseignant peut ajouter, modifier, supprimer ou consulter une absence d'un étudiant qui appartient au groupe qu'il enseigne. L'étudiant peut consulter la liste de ses absences. L'administrateur peut consulter la liste des absences de n'importe quel étudiant de l'école.	Conforme
Test de visualisation des statistiques.	-L'administrateur peut visualiser les statistiques relatives aux taux d'absence, notes des étudiants, etc.	Conforme

Tests du sprint 2

## Conclusion

A ce niveau, nous avons réussi à développer le dernier sprint de notre système pour arriver enfin à un produit fonctionnel et complet. Notre application est prête à être exploitée en offrant aux utilisateurs la possibilité de gérer les modules de la plateforme.

## CONCLUSION GÉNÉRALE ET PERSPECTIVES

Ce rapport s'inscrit dans le cadre de notre projet de conception et de développement réalisé à l'école nationale des sciences de l'informatique.

Nous avons mis en place un workflow pour la gestion des processus administratives de l'école.

Nous avons commencé en premier lieu par présenter le cadre général du projet. Dans cette partie, nous avons entamé une étude préliminaire pour présenter la problématique et l'objectif du travail.

Par la suite, nous avons entamé l'analyse et la spécification des besoins.

Dans ce chapitre, nous avons identifié les besoins fonctionnels, non fonctionnels et du domaine et les différents acteurs qui vont interagir avec le système. Ensuite, nous avons présenté le style architectural utilisé et l'environnement de travail.

Puis, nous avons entamé les sprints. En effet chaque sprint comporte la spécification des besoins à l'aide des diagrammes de cas d'utilisation du langage UML, la conception globale, la conception détaillée qui comporte les diagrammes de séquence objet et les diagrammes de classes et enfin la réalisation et les tests.

Pour conclure, ce projet nous a permis d'acquérir une expérience enrichissante et de mettre en pratique les concepts théoriques acquis durant notre cursus académique. Il présente de même

une opportunité pour maîtriser de nouvelles technologies intéressantes.

Bien évidemment, ceci n'est qu'une phase d'un travail qui va continuer à progresser en enrichissant les fonctionnalités mises à disposition des utilisateurs par d'autres comme la demande de service, l'ajout et le téléchargement de fichiers d'actualité, etc. Nous pouvons aussi développer une application mobile qui permet de gérer les notifications.



# Netographie

- [1] [En ligne]. Available : <https://blog.myagilepartner.fr/index.php/2018/08/17/ceremonies-sprint-scrum/> [Accès le 15 février 2021].
- [2] Available : <https://fr.wikipedia.org/wiki/VisualStudioCode> [Accès le 16 février 2021].
- [3] [En ligne]. Available : <https://fr.wikipedia.org/wiki/MicrosoftSQLserver> [Accès le 1 mars 2021].
- [4] [En ligne]. Available : <https://fr.wikipedia.org/wiki/React> [Accès le 10 mars 2021].
- [5] [En ligne]. Available : <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0> [Accès le 15 mars 2021]
- [6] [En ligne]. Available : <https://docs.github.com/en> [Accès le 16 mars 2021]
- [7] [En ligne]. Available : <https://www.codingame.com/playgrounds/35462/creating-web-api-in-asp-net-core-2-0/part-1---web-api> [Accès le 17 mars 2021]
- [8] [En ligne]. Available : Jetons JWT et sécurité .Principes et cas d' utilisation. <https://www.vaadata.com/blog/frjetons-jwt-et-securite-principes-et-cas-dutilisation> [Accès le 10 avril 2021]
- [9] [En ligne]. Available : Token-Based Authentication With AngularJS NodeJS. <https://code.tutsplus.com/tutorials/token-based-authentication-with-angularjs-nodejs--cms-22543> [Accès le 10 avril 2021]

# Annexe

## Authentification à base de Token

### Principe

Lors de plusieurs accès simultanés à la base de données, le serveur peut recevoir des requêtes simultanées. De ce fait, chaque utilisateur doit disposer d'un jeton qui l'identifie. Chaque utilisateur qui est authentifié avec succès, reçoit un jeton qui est une chaîne de caractères et d'une durée de vie spécifique.

### JWT : JSON Web Token

Les JWT sont des jetons générés par un serveur lors de l'authentification d'un utilisateur sur une application web, et qui sont ensuite transmis au client [8]

### Pourquoi JWT

Dans le cas de notre application, nous avons utilisé des API REST.

Le transfert de données se fait à travers le protocole HTTP en envoyant des requêtes et sous format JSON.

De cette façon, nous pouvons faire l'authentification avec JWT.

### Structure d'un jeton JWT

Un jeton JWT se compose de 3 champs selon la norme RFC 7519.

La figure ci-dessous représente les différents champs d'un jeton JWT.



## Requête d'obtention du Token JWT

Ci-dessous la figure qui présente la requête pour l'obtention du Token JWT.

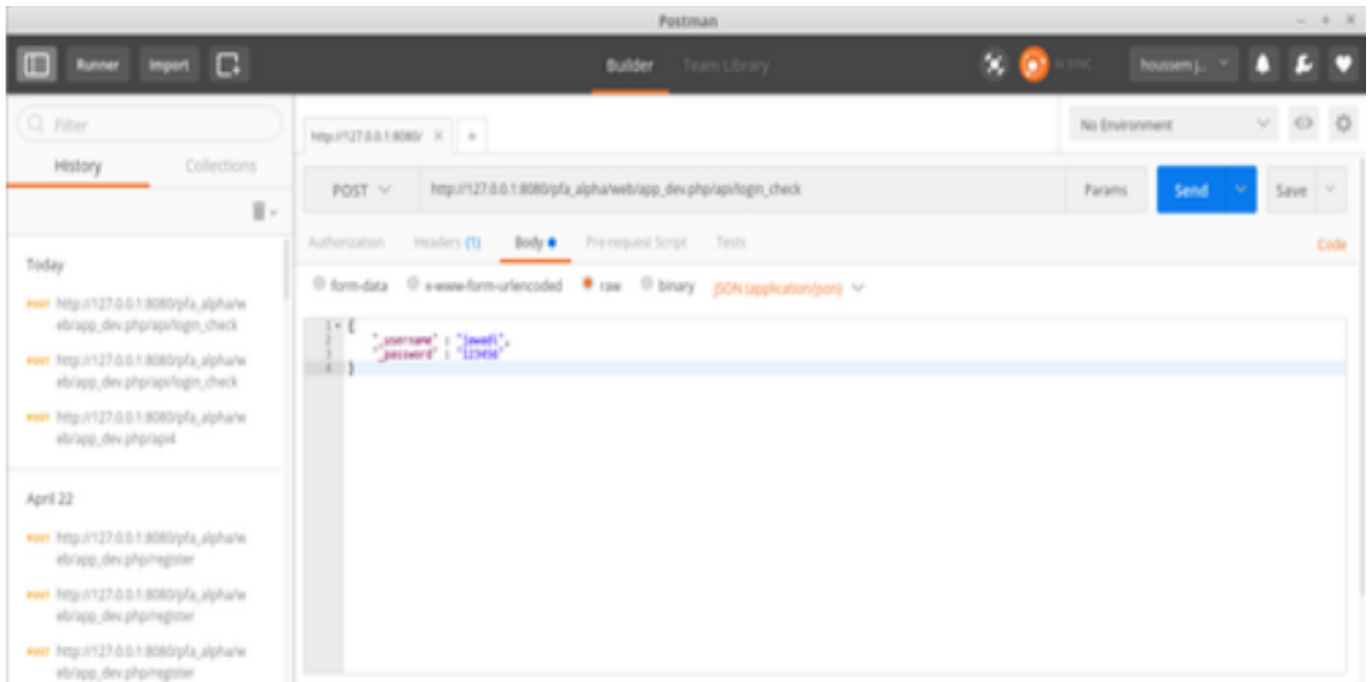


FIGURE IV.19 – Requête d’obtention du jeton JWT

Ci-dessous la figure qui présente la réponse envoyée suite à la requête. Dans le cas d'un échec, on reçoit un message d'erreur.



FIGURE IV.20 – Réponse reçue par le serveur

## DATA TRANSFER OBJECT (DTO)

Il permet de composer les réponses de l'API ainsi que ses paramètres.

Ce modèle est créé avec un objectif bien défini : transférer des données vers des interfaces distantes , tout comme les services Web.

Ceci limite les données acceptées par les endpoints permettant un meilleur contrôle des données à modifier.

La figure suivante montre l'utilisation de l'interface `etudiantService` afin de supprimer un étudiant de la base de données en utilisant les DTO.

```
[HttpDelete("api/Etudiant/{cin}")]
public async Task<IActionResult> DeleteEtudiant(string cin)
{
    var etudiant =await _etudiantService.GetEtudiantByCinAsync(cin);
    if(etudiant.Inscriptions==null|| etudiant.Inscriptions.Count==0){
        await _etudiantManager.DeleteAsync(etudiant);
        return Ok();
    }

    return NoContent();
}
```

FIGURE IV.21 – Utilisation du modèle Data Transfer Object

# Résumé

Dans le présent travail,nous avons conçu et réalisé dans le cadre de notre projet de développement et de conception, une plateforme pour la gestion des processus administratifs de l'ENSI.

En effet,la plateforme sert à automatiser quelques processus de travail dans l'administration,auxquels les étudiants de l'école nationale des sciences de l'informatique peuvent accéder à travers des interfaces graphiques ergonomiques et conviviales dans le but de gagner du temps.

## ABSTRACT

In the present work,we created and implemented as a part of our development and design project at our National School of Computer Science, a platform for administration and students.Indeed,this platform serves to automate the entire work process in the administration and to which students can access to their information through ergonomic and user-friendly graphical interfaces in order to save time.

## ملخص

في العمل الحالي قمنا بتصميم و تنفيذ كجزء من مشروعنا داخل المدرسة الوطنية لعلوم الاعلامية بمنوبة تونس  
تطبيقا تساهم في تقريب المعلومة من الطالب و لاسيما في تسهيل المعاملات بين الإدارة و الطلبة و هذا يندرج ضمن  
رقمته الإدارة .