

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Abderrahmane Mira de Bejaia  
Faculté des Sciences Exactes  
Département d'Informatique



**Mémoire De Fin De Cycle**  
**Pour l'obtention du diplôme de Master Professionnel en Informatique**  
**Spécialité : Administration et sécurité des réseaux**

## Thème

---

**Conception et réalisation d'une plateforme pour la gestion  
des activités de la faculté des sciences exactes de  
l'université de Béjaia**

---

Réalisé par : **MEGHAR Nassim**

Sous la supervision de : **Mme Houda EL BOUHISSI BRAHAMI**

**Soutenu le 1er juillet 2024 devant le jury composé de :**

- Mr. Khaled ALLEM : Président
- Mme Nadjat KHOULALLEN : Examinatrice

Année académique :2023-2024

## RÉSUMÉ

---

L'avancée technologique actuelle permet d'optimiser divers processus académiques, y compris ceux liés à la gestion des enseignants, des doctorants et des stages. Ce projet a pour but de concevoir une application web nommée "AcademicApp", destinée à la gestion des activités pédagogiques de la Faculté des Sciences Exactes de l'Université de Béjaïa.

Cette plateforme a pour rôle de faciliter le travail des secrétaires et du doyen de la faculté, en permettant l'affichage et la consultation des enseignants et doctorants de manière efficace et rapide. Elle permet aussi de gérer la coordination des stages et l'impression des attestations de manière centralisée.

Le projet suit le processus unifié (UP), en utilisant le langage de modélisation UML pour la conception et l'architecture du système, afin d'assurer une modélisation claire et cohérente des fonctionnalités de l'application. La réalisation a été effectuée à l'aide de plusieurs outils de développement tels que Visual Studio Code, Bootstrap, et le Framework Laravel pour PHP.

---

## ABSTRACT

---

Technological advancements today enable the optimization of various academic processes, including the management of teachers, doctoral students, and internships. This project aims to design a web application named "AcademicApp", intended for managing the academic activities of the Faculty of Exact Sciences at the University of Bejaia.

This platform's primary role is to assist secretaries and the dean by providing an efficient and quick way to display and consult teachers and doctoral students. It also allows for managing the coordination of internships and printing certificates, offering a centralized management solution.

The project follows the Unified Process (UP), using the UML modeling language for design to ensure clear and coherent modeling of the application's functionalities. The implementation was carried out using various development tools such as Visual Studio Code, Bootstrap, and the Laravel PHP framework.

---

## REMERCIEMENTS

---

Tout d'abord, nous tenons à exprimer notre profonde gratitude à Madame le Professeur El Bouhissi Houda, pour son soutien constant et ses encouragements tout au long de la réalisation de ce mémoire. Nous la remercions sincèrement pour le temps qu'elle nous a accordé, ses remarques pertinentes et ses retours constructifs, qui ont grandement contribué à l'aboutissement de ce travail. Sa rigueur académique et son souci du détail ont été une véritable source d'inspiration.

Nos remerciements vont également à tous les membres du jury pour avoir accepté d'examiner ce travail. Leurs observations et suggestions ont permis d'enrichir et d'améliorer la qualité de ce mémoire.

Enfin, nous adressons nos sincères remerciements à toutes les personnes qui ont, d'une manière ou d'une autre, contribué à la réalisation de ce projet, qu'il s'agisse de collègues, de chercheurs ou de toute personne ayant partagé son expertise et ses conseils.

---

## Table des matières

Table des figures.....	6
Liste des tableaux .....	8
Introduction générale.....	9
1 Introduction :.....	11
1.1 Définition de la faculté.....	11
1.2 Rôle de la Faculté dans une Université :.....	11
2 Services offerts par la Faculté .....	12
3 Tâches de la Faculté : .....	12
4 La gestion de la faculté :.....	12
5 Présentation du projet :.....	13
5.1 Processus de Conception .....	14
5.1.1 Analyse des Besoins :.....	14
5.1.2 Conception UML :.....	14
5.1.3 Implémentation :.....	14
5.1.4 Validation et Test : .....	14
5.2 Problématique :.....	15
5.2.1 État des lieux .....	15
5.2.2 Problèmes identifiés .....	15
5.2.3 Conséquences .....	15
5.2.4 Objectif.....	15
5.2.5 Solution.....	16
6 Méthodologie de développement .....	16
6.1 UML.....	16
6.2 Processus unifié.....	17
7 Conclusion .....	17
1 Introduction.....	18
2 Étude de l’Existant .....	18
2.1 Objectif de l’étude de l’existant .....	18
2.2 Analyse des méthodes actuelles .....	18
2.2.1 Méthodes manuelles .....	18
2.2.2 Outils existants .....	19
2.3 Problèmes identifiés .....	19
2.4 Justification de la solution.....	19

3	Analyse des besoins.....	20
3.1	Phase d’Inception .....	20
3.1.1	Expression des besoins .....	20
3.1.2	Intitulé du projet .....	20
3.1.3	<b>OBJECTIF DE L’APPLICATION</b> .....	20
3.1.4	Public visé .....	21
3.2	Identification des besoins .....	21
3.2.1	Besoins fonctionnels.....	21
3.2.2	Besoins non fonctionnels.....	22
3.3	Identification des cas d’utilisation.....	22
3.4	Description textuelle des cas d’utilisation .....	24
3.4.1	Cas d’utilisation « S’authentification » .....	24
3.4.2	Cas d’utilisation « Ajouter un enseignant » .....	25
3.4.3	Cas d’utilisation « Supprimer un enseignant ».....	26
3.4.4	Cas d’utilisation « Consulter la liste des enseignants ».....	27
3.4.5	Cas d’utilisation « Rechercher ».....	28
3.4.6	Cas d’utilisation « Imprimer attestation » .....	29
3.4.7	Cas d’utilisation « consulter les message » .....	30
3.5	Diagramme de séquence.....	30
3.5.1	Diagramme de séquence « S’authentification » .....	30
3.5.2	Diagramme de séquence « Ajouter un enseignant » .....	31
3.5.3	Diagramme de séquence « Supprimer un enseignant ».....	32
3.5.4	Diagramme de séquence « annuler un stage ».....	33
3.5.5	Diagramme de séquence « Imprimer une attestation » .....	34
3.5.6	Diagramme de séquence « rechercher » .....	35
3.6	Dictionnaire de données .....	36
3.6.1	Liste Des Règles De Gestion.....	38
3.7	Diagramme de classe de conception.....	38
3.8	Passage au modèle relationnel.....	40
3.8.1	Fondements.....	40
3.8.2	Les Règles De passage .....	40
3.8.3	Modèle Relationnel .....	41
4	Conclusion .....	42
1	Introduction.....	43

2	Outils de développement .....	43
2.1	Environnement machine .....	43
2.1.1	MacBook Air .....	43
2.1.2	Dell Latitude 7872.....	44
2.1.3	Acer Aspire.....	44
2.2	Environnement logiciel .....	44
2.2.1	XAMPP .....	44
2.2.2	Visual Code studio .....	46
2.2.3	La bibliothèque logicielle PHPMailer : .....	47
2.2.4	Composer.....	47
2.3	Environnement de programmation .....	48
2.3.1	PHP.....	48
2.3.2	CSS.....	49
2.3.3	Bootstrap.....	50
2.3.4	HTML.....	50
2.3.5	JavaScript .....	51
2.4	Schéma de navigation de l'application .....	52
2.4.1	Accueil.....	52
2.4.2	Register.....	53
2.4.3	Login.....	54
2.4.4	Page Administrateur .....	54
2.4.5	Page profile.....	55
2.4.6	Page liste des enseignants.....	56
2.4.7	Page liste des doctorants.....	56
2.4.8	Page liste des stages.....	57
2.4.9	Page contact.....	58
2.4.10	Ajouter Enseignant .....	58
2.4.11	Ajouter Doctorant.....	59
2.4.12	Planifier un stage .....	60
2.4.13	Supprimer enseignant .....	60
2.4.14	Supprimer doctorant.....	61
2.4.15	Annuler stage.....	62
2.4.16	Afficher plus de détails sur le stage.....	62
2.4.17	Imprimer attestation.....	63

2.4.18	Rechercher.....	64
2.5	Conclusion.....	64
	Conclusion générale .....	65
	Bibliographie .....	66

## Table des figures

Figure 1:	Schéma conceptuel de la gestion de la faculté.....	13
Figure 2:	diagramme de cas d'utilisation .....	23
Figure 3:	diagramme de séquence Authentification.....	31
Figure 4:	diagramme de séquence "ajouter un enseignant" .....	32
Figure 7:	Diagramme de séquence supprimer un enseignant.....	33
Figure 9:	Diagramme de séquence annuler un stage .....	34
Figure 10:	Diagramme de séquence « Imprimer l'attestation ».....	35
Figure 12:	diagramme de séquence rechercher .....	36
Figure 13:	diagramme de classe .....	39
Figure 14:	logo XAMPP.....	44
Figure 15:	logo d'APACHE.....	45
Figure 16:	logo MySQL .....	45
Figure 17:	Interface Visual Studio Code.....	46
Figure 18:	Logo PHPMailer .....	47
Figure 19:	logo composer.....	47
Figure 20:	Logo PHP.....	48
Figure 21:	logo Laravel .....	49
Figure 22:	logo Fortify laravel .....	49
Figure 23:	logo CSS .....	50
Figure 24:	logo Bootstrap.....	50
Figure 25:	logo HTML .....	51
Figure 26:	logo JavaScript.....	51
Figure 27:	Schéma de navigation de l'application .....	52
Figure 28:	page D'accueil.....	53

Figure 29:page Register.....	53
Figure 30:page login .....	54
Figure 31:profile administrateur.....	55
Figure 32:profile simple user.....	55
Figure 33:page liste des enseignants .....	56
Figure 34:page liste des doctorants .....	57
Figure 35:page liste des stages .....	58
Figure 36:page contact .....	58
Figure 37:ajouter enseignant .....	59
Figure 38:Ajouter doctorant .....	59
Figure 39:page planifier un stage .....	60
Figure 40:supprimer enseignant .....	61
Figure 41:supprimer doctorant .....	61
Figure 42:annuler stage .....	62
Figure 43:page plus de détails .....	62
Figure 44:imprimer attestation de stage .....	63
Figure 45:imprimer attestation de travail .....	63
Figure 46:la fonction rechercher .....	64

## Liste des tableaux

Tableau 1:Acteurs et leurs rôles .....	22
Tableau 2:Cas d'utilisation « Authentification ».....	24
Tableau 3 : Cas d'utilisation Ajouter un enseignant .....	25
Tableau 5:Cas d'utilisation supprimer un enseignant .....	26
Tableau 7:cas d'utilisation consulter la liste des enseignants .....	27
Tableau 10:Cas d'utilisation rechercher .....	28
Tableau 11 cad d'utilisation imprimer une attestation de travail.....	29
Tableau 15:cas d'utilisation consulter les messages .....	30
Tableau 16:Dictionnaire de données .....	38
Tableau 17:Fiche technique de Visual Studio Code.....	46

## Introduction générale

La Faculté des Sciences Exactes de l'Université de Béjaïa est un pilier fondamental dans la formation académique des futurs chercheurs, ingénieurs et professionnels dans des domaines variés tels que les mathématiques, la physique, l'informatique, la chimie et d'autres sciences exactes. Son rôle est essentiel pour le développement intellectuel et technologique de la société, en offrant une éducation de qualité, en menant des recherches scientifiques de pointe et en contribuant au développement socio-économique de la région et du pays. Toutefois, malgré son importance, la faculté fait face à des défis majeurs dans la gestion de ses activités pédagogiques et administratives, ce qui nuit à l'efficacité de ses processus académiques. Les méthodes traditionnelles de gestion, souvent basées sur des outils manuels ou bureautiques, sont insuffisantes pour gérer la complexité croissante des données académiques et des processus administratifs.

Dans ce contexte, ce mémoire présente le développement d'une application web, dénommée "AcademicApp", conçue pour améliorer la gestion des enseignants, des doctorants et des stages au sein de la Faculté des Sciences Exactes. Le principal objectif de ce projet est de faciliter le travail des secrétaires et des responsables administratifs en centralisant et automatisant plusieurs processus essentiels à la gestion académique, tels que l'enregistrement, la mise à jour et la consultation des informations des enseignants et doctorants, ainsi que la gestion des stages et des attestations liées à ces activités. En centralisant toutes ces données sur une plateforme unique et accessible, l'application vise à améliorer l'efficacité administrative et à réduire les risques d'erreurs humaines liés aux méthodes traditionnelles.

Le travail réalisé est structuré autour de trois chapitres principaux qui permettent de comprendre l'ensemble de la démarche de conception et de développement du système. Dans le Chapitre 1, nous détaillons le contexte et la problématique de la gestion actuelle au sein de la faculté. Ce chapitre met en lumière les défis rencontrés par les administrateurs, les enseignants et les doctorants dans l'exécution de leurs tâches quotidiennes. Nous analysons les limites des méthodes traditionnelles de gestion, telles que l'utilisation d'outils bureautiques et manuels, et la nécessité d'un système informatisé pour améliorer la gestion des activités académiques.

Dans le Chapitre 2, nous présentons la conception de l'application, en utilisant le processus unifié (UP) et le langage de modélisation UML. Cette phase de conception a impliqué la collecte des besoins fonctionnels et non fonctionnels de la faculté, ainsi que la modélisation du système à l'aide de diagrammes de cas d'utilisation, de diagrammes de séquence et de diagrammes de classes. Ces outils de modélisation ont permis de structurer de manière claire et précise les différentes fonctionnalités du système, en assurant que chaque besoin de gestion académique soit pris en compte. Ce chapitre décrit également les choix techniques effectués pour garantir une solution évolutive, robuste et adaptée aux spécificités de la faculté.

Dans le Chapitre 3, nous nous concentrons sur l'implémentation de l'application en détaillant les technologies et outils utilisés. L'application a été développée en utilisant le framework Laravel pour la partie backend, offrant ainsi une solution performante et sécurisée pour la gestion des données. Le frontend a été conçu avec Bootstrap pour garantir une interface utilisateur moderne, réactive et conviviale. Ce chapitre décrit également les principales fonctionnalités mises en œuvre, telles que l'ajout et la suppression d'enseignants et de doctorants, la gestion des stages, ainsi que la génération et l'impression des attestations. En outre, nous présentons les résultats obtenus grâce à l'implémentation de l'application, en soulignant l'impact positif de cette solution sur la gestion des activités pédagogiques de la faculté.

Ainsi, le projet "AcademicApp" constitue une réponse concrète et efficace aux défis de gestion rencontrés par la faculté, en offrant une plateforme centralisée et intuitive permettant aux administrateurs, enseignants et doctorants d'accéder rapidement aux informations et de gérer les processus administratifs de manière optimale. Les résultats obtenus montrent que l'application permet de simplifier et de sécuriser la gestion des activités académiques, réduisant ainsi les erreurs et améliorant la productivité de tous les acteurs impliqués.

# CHAPITRE 1

## CONTEXTE

### 1 Introduction :

L'avènement des nouvelles technologies a révolutionné de nombreux aspects de notre vie quotidienne, et l'éducation ne fait pas exception. Avec la montée en puissance des smartphones et de l'accessibilité à Internet, les applications web jouent désormais un rôle clé dans la gestion et la facilitation des processus académiques. Face à ces avancées, les universités, notamment les facultés des sciences exactes, doivent adapter leurs outils et infrastructures pour répondre aux attentes croissantes des étudiants, des enseignants et des administrateurs. Dans ce contexte, la conception et la réalisation d'une plateforme dédiée à la gestion pédagogique et administrative s'imposent comme une solution incontournable pour améliorer l'efficacité, réduire les erreurs et simplifier les processus.

Pour mieux comprendre les enjeux de ce projet, il est important de préciser le rôle et les responsabilités d'une faculté au sein d'une université.

#### 1.1 Définition de la faculté

Une faculté est une unité académique regroupant des enseignants, chercheurs et étudiants autour d'un domaine spécifique. Elle est responsable de dispenser des formations, de conduire des recherches scientifiques et de contribuer au développement des compétences académiques et professionnelles. Dans le cas de la Faculté des sciences exactes, cette mission s'accompagne de la gestion d'activités pédagogiques complexes, telles que la planification des cours, le suivi des doctorants et la coordination des stages. Ces responsabilités, de plus en plus exigeantes, nécessitent des outils modernes et efficaces pour garantir une gestion fluide et optimale.

#### 1.2 Rôle de la Faculté dans une Université :

La faculté joue un rôle central dans la mission éducative et de recherche de l'université. Elle est chargée de dispenser un enseignement de qualité, de mener des recherches avancées et de fournir des services essentiels à la communauté universitaire.

## 2 Services offerts par la Faculté

Ces services, bien que fondamentaux, rencontrent actuellement des défis organisationnels et administratifs que notre plateforme vise à résoudre.

La faculté est chargée d'offrir divers services, notamment :

1. **Enseignement** : Organisation et suivi des cours dispensés aux étudiants.
2. **Recherche** : Supervision des projets de recherche et coordination des thèses doctorales.
3. **Encadrement** : Conseil académique pour les étudiants et doctorants.
4. **Gestion administrative** : Planification des emplois du temps, suivi des stages et gestion des attestations.

## 3 Tâches de la Faculté :

1. Dispenser des cours dans son domaine d'expertise.
2. Mener des recherches originales et publier des articles.
3. Encadrer et conseiller les étudiants dans leurs études et leur carrière.
4. Participer à des activités de sensibilisation communautaire.
5. Collaborer avec d'autres départements et institutions.
6. Évaluer les programmes académiques et participer à des processus d'accréditation.
7. Participer à des activités de développement professionnel.
8. Assumer des responsabilités administratives au sein de l'institution.

## 4 La gestion de la faculté :

La gestion d'une faculté implique la coordination de nombreuses activités académiques, administratives et organisationnelles, telles que :

- **Planification académique** : Élaboration des programmes, gestion des emplois du temps et suivi des cours.
- **Suivi des personnels académiques** : Gestion des informations des enseignants, doctorants et encadreurs de stages.
- **Coordination des stages** : Organisation, planification et suivi des stages pour les doctorants.
- **Production de documents administratifs** : Génération d'attestations de travail et de stage.



- Affichage de la liste des enseignants par spécialité, grade et département.
- Collecte des informations suivantes : nom, prénom, date de naissance, grade, spécialité, date de recrutement, dates d'obtention des grades.

## **2. Gestion des Doctorants :**

- Affichage de la liste des doctorants par spécialité, année d'inscription et département.
- Suivi des données essentielles : numéro d'inscription, nom, prénom, date de naissance, spécialité, année d'inscription, directeur de thèse.

## **3. Gestion des stages :**

- Enregistrement des détails des projets, y compris le code du projet, le chef de projet, les membres du projet et la date d'allée et de retour.

## 5.1 Processus de Conception

### 5.1.1 Analyse des Besoins :

- Identification des exigences fonctionnelles et non fonctionnelles de chaque module.
- Interaction avec les parties prenantes pour valider les besoins.

### 5.1.2 Conception UML :

- Modélisation des acteurs, des cas d'utilisation et des diagrammes de séquence pour chaque fonctionnalité.
- Conception des diagrammes de classes pour représenter la structure des données.

### 5.1.3 Implémentation :

- Développement des différentes fonctionnalités en utilisant les langages de programmation appropriés et en assurant l'intégration avec une base de données relationnelle.

### 5.1.4 Validation et Test :

- Vérification de la conformité des fonctionnalités développées avec les spécifications.
- Test exhaustif pour assurer la fiabilité de l'application.

## 5.2 Problématique :

La gestion pédagogique dans les universités, en particulier au sein de la Faculté des Sciences Exactes, présente de nombreux défis pour l'administration, les enseignants et les doctorants. Ces défis, souvent liés à l'utilisation de méthodes traditionnelles, entravent la fluidité et l'efficacité des processus académiques.

### 5.2.1 État des lieux

Actuellement, la faculté s'appuie sur des outils manuels ou des logiciels de bureautique non centralisés pour gérer ses activités. Ces outils, bien que fonctionnels pour des tâches simples, se révèlent insuffisants face à la complexité croissante des données académiques. Par exemple, la gestion des enseignants, des doctorants, des plannings et des stages exige des processus plus robustes et intégrés.

### 5.2.2 Problèmes identifiés

1. **Absence de centralisation** : Les informations sont dispersées, rendant difficile leur accès rapide et fiable.
2. **Gestion inefficace des processus** : Les modifications de plannings ou les mises à jour des données sont lentes et sujettes à des erreurs.
3. **Manque de transparence** : Les étudiants, enseignants et doctorants ne peuvent pas consulter facilement les informations essentielles, créant frustrations et retards.
4. **Risque d'erreurs humaines** : Les tâches manuelles augmentent la probabilité d'erreurs dans la gestion des données.

### 5.2.3 Conséquences

Ces limitations ont des répercussions significatives :

- **Perte de temps** pour les administrateurs et utilisateurs.
- **Dégradation de la qualité des services** académiques.
- **Insatisfaction générale** des acteurs impliqués dans les processus académiques.

### 5.2.4 Objectif

Face à ces défis, le projet propose de développer une application web nommée "AcademicApp". Cette plateforme vise à :

- **Centraliser les informations** : Offrir un accès unique à toutes les données académiques pertinentes.
- **Automatiser les tâches répétitives** : Réduire les erreurs humaines et améliorer l'efficacité des processus.
- **Améliorer l'expérience utilisateur** : Permettre aux enseignants, doctorants et administrateurs d'accéder rapidement aux données et d'effectuer leurs tâches avec facilité.
- **Augmenter la productivité** : Simplifier les flux de travail et offrir des outils modernes adaptés aux besoins spécifiques de la faculté.

Ainsi, "AcademicApp" répondra aux besoins urgents d'une gestion plus fluide, efficace et sécurisée des activités académiques de la faculté, tout en réduisant les tensions liées aux méthodes actuelles.

### 5.2.5 Solution

Afin de remédier aux problèmes cités ci-dessus, nous proposons la conception et la réalisation d'un outil de gestion.

Cet outil permettra à :

- L'administrateur d'ajouter et de supprimer des enseignants et des doctorants, de créer des stages.
- L'administrateur peut imprimer des attestations de stage et de travail.
- L'administrateur peut consulter les messages envoyés par des user.
- L'administrateur peut ajouter des administrateurs.
- User Connecté peut consulter la liste des enseignants et des doctorants et des stages.
- User Anonyme non connecté il a le droit que pour envoyer un message et contacter l'administrateur.

## 6 Méthodologie de développement

Pour développer notre application Web, notre choix s'est porté sur le processus unifié (UP) basé sur le langage de modélisation UML, une solution qui s'adapte à tous types de projets parce qu'il est guidé par les cas d'utilisations, aiguillé sur l'architecture et itératif et incrémental.

### 6.1 UML

Le sigle UML se lit en Anglais Unified Modeling Language, ce qui se traduit en Français par le langage de modélisation unifié, développé en réponse à l'appel à propositions lancé par l'OMG (Object Management Group) dans le but de définir la notation standard pour la modélisation des applications construites à l'aide d'objets. UML a vu le jour en novembre 1997, suite à la fusion de trois différentes méthodes, dont Object Oriented software Engineering (OOSE) de Ivar Jacobson, BOOCH de Grady Booch et Object Modeling Technique (OMT) de James Rumbaugh. Et depuis sa sortie il s'est imposé comme standard à utiliser en tant que langage de modélisation objet [3].

Cependant, UML n'est pas une méthode, mais seulement un langage graphique qui permet de modéliser et de communiquer les divers aspects d'un système d'information. N'ayant pas un cycle de vie propre, UML doit être associé à un processus de développement.

## 6.2 Processus unifié

Le processus unifié est un processus de développement logiciel associé à UML, il est tout d'abord guidé par les cas d'utilisation car la solution informatique et le système à construire va être défini avec les utilisateurs ce qui permet de capturer les besoins efficacement. UP est un processus centré sur l'architecture et orienté par la réduction des risques, et en étant itératif et incrémental il permet de :

- Avoir un feed-back des utilisateurs dès les premières itérations.
- Évaluer les risques.
- Effectuer des tests de manière continue.

Les caractéristiques du processus unifié lui permettent de s'adapter à une large classe de systèmes logiciels de différents domaines d'application, à différentes tailles d'entreprises de différents types.

## 7 Conclusion

Ce chapitre j'ai servi à vous présenter le cadre de mon projet. J'ai présenté l'organisme d'accueil de notre application Web, ses missions, ses besoins, ses objectifs ainsi que la méthodologie de développement optée. Dans le prochain chapitre je vais définir la partie de conception et analyse en présentant le cadre fonctionnel et le cadre dynamique de ce système.

# CHAPITRE 2

## CONCEPTION DE L'APPLICATION

### 1 Introduction

Ce chapitre présentera les étapes réalisées tout au long de notre projet suivant le processus UP (Unified Process). Il exposera donc toute la documentation technique et conceptuelle, en s'appuyant sur une analyse rigoureuse des besoins. Une étude de l'existant sera également incluse pour mieux comprendre les problèmes liés aux systèmes actuels et pour justifier la pertinence de la solution proposée.

Enfin, je réaliserai le modèle relationnel qui va permettre de présenter la structure statique du système d'information.

### 2 Étude de l'Existant

#### 2.1 Objectif de l'étude de l'existant

L'étude de l'existant permet d'identifier les systèmes ou les méthodes actuellement utilisés pour la gestion des activités académiques et administratives au sein de la Faculté des Sciences Exactes ou dans des contextes similaires. Cette analyse vise à comprendre les limites des approches actuelles et à mettre en évidence les besoins auxquels notre application, AcadimicApp, répondra.

#### 2.2 Analyse des méthodes actuelles

##### 2.2.1 Méthodes manuelles

Actuellement, les activités administratives telles que la gestion des enseignants, des doctorants, des stages et des plannings sont majoritairement effectuées à l'aide d'outils manuels ou bureautiques, tels que les fichiers Excel ou les documents papier. Ces méthodes présentent plusieurs limites :

- Fragmentation des informations : Les données sont souvent dispersées dans plusieurs fichiers, rendant difficile leur accès rapide et centralisé.
- Risque élevé d'erreurs humaines : La manipulation manuelle des données entraîne des risques de doublons, d'incohérences ou d'omissions.
- Temps de traitement long : Par exemple, la création d'un planning ou la recherche d'informations spécifiques peut prendre plusieurs heures.

### 2.2.2 Outils existants

Dans d'autres universités ou facultés, certains logiciels comme Moodle ou des systèmes de gestion spécialisés sont utilisés. Cependant, ces solutions sont souvent génériques et ne répondent pas toujours aux besoins spécifiques de la Faculté des Sciences Exactes. Les principales limitations observées dans ces systèmes sont :

- Une personnalisation limitée des fonctionnalités pour la gestion des stages ou des doctorants.
- Une courbe d'apprentissage parfois élevée pour les utilisateurs non technophiles.
- Des coûts élevés pour l'acquisition ou la maintenance.

### 2.3 Problèmes identifiés

Sur la base de cette analyse, plusieurs problèmes majeurs se dégagent :

1. **Absence de centralisation** : Les informations académiques sont dispersées, ce qui complique leur gestion.
2. **Faible ergonomie des outils actuels** : Les utilisateurs doivent naviguer entre différentes plateformes ou fichiers, augmentant les risques d'erreurs.
3. **Manque de transparence** : Les doctorants et enseignants n'ont pas un accès facile aux informations essentielles (plannings, stages, attestations, etc.).
4. Risque accru d'erreurs humaines : La gestion manuelle augmente la probabilité d'inexactitudes.

### 2.4 Justification de la solution

Face à ces problèmes, mon application web AcadimicApp propose une solution innovante et adaptée :

- **Centralisation des données** : Toutes les informations seront accessibles depuis une plateforme unique.
- **Automatisation des processus** : Réduction des tâches répétitives et fastidieuses pour améliorer l'efficacité.

- **Accessibilité accrue** : Une interface intuitive permettra aux utilisateurs (enseignants, doctorants, administrateurs) d'accéder facilement aux fonctionnalités.

- **Fiabilité des données** : La centralisation et l'automatisation minimiseront les risques d'erreurs.

Transition vers la conception

L'étude de l'existant a permis d'identifier les problèmes et les lacunes des systèmes actuels. Dans la section suivante, nous présenterons les besoins fonctionnels et non fonctionnels de l'application, ainsi que les modèles UML permettant de concevoir une solution adaptée.

### 3 Analyse des besoins

L'analyse des besoins représente la toute première étape de la conception de tout projet informatique, elle permet tout d'abord de prendre en compte les contraintes, les risques et tous les éléments pertinents du projet. Cette phase permet d'accéder à une compréhension des besoins et des exigences du client, et ceci afin de livrer les spécifications qui déterminent le choix de la conception de la solution.

#### 3.1 Phase d'Inception

##### 3.1.1 Expression des besoins

Dans cette partie, nous présentons les informations et les besoins que nous avons recueillis au sein de la faculté de notre université.

##### 3.1.2 Intitulé du projet

Le projet se résume en conception et réalisation d'une plateforme pour la gestion des activités de la faculté des sciences exactes sera nommée « AcadimicApp ».

##### 3.1.3 OBJECTIF DE L'APPLICATION

Elle a pour objectif de fournir une plateforme disposant d'un ensemble de fonctionnalités nécessaires qui permettent de faciliter la gestion et création des stages, la consultation des enseignants, des doctorants.

### 3.1.4 Public visé

L'application Web que je vais réaliser s'adresse essentiellement à l'administration, aux enseignants et doctorants.

## 3.2 Identification des besoins

L'identification des besoins a pour objectifs le recueil des besoins en utilisant le concept de cas d'utilisations proposé par UML, ensuite et pour chaque cas identifier toute classe candidate du modèle statique d'analyse.

### 3.2.1 Besoins fonctionnels

Un système ne devient opérationnel que s'il satisfait les besoins fonctionnels (besoins métiers). Principalement, notre application doit couvrir les besoins fonctionnels représentés dans le tableau 1 associés à leurs acteurs.

Acteurs	Rôles
<b>Administrateur</b>	<ul style="list-style-type: none"> <li>• Ajouter/supprimer un enseignant</li> <li>• Ajouter/supprimer un doctorant</li> <li>• Créer/supprimer un stage</li> <li>• Designner le responsable du stage</li> <li>• Designner les membre qui vont participer au stage.</li> <li>• Consulter les messages.</li> <li>• Imprimer des attestations.</li> <li>• Designner des administrateurs</li> </ul>

Enseignant, doctorant (User connecté)	<ul style="list-style-type: none"> <li>• Consulter la liste des stages</li> <li>• Consulter la liste des enseignants</li> <li>• Consulter la liste des doctorants</li> </ul>
--	--

Tableau 1: Acteurs et leurs rôles

### 3.2.2 Besoins non fonctionnels

Les besoins non fonctionnels sont des exigences qui portent sur le comportement du système, mais permettent d'identifier ses exigences internes et externes qui garantissent sa performance et le respect des exigences de l'utilisateur.

#### ❖ *PERFORMANCE :*

Notre application doit assurer un temps de réponse minimum tout en répondant aux besoins du manipulateur.

#### ❖ *LA SIMPLICITE :*

Chaque utilisateur pourra utiliser cette application d'une manière facile et claire.

#### ❖ *L'ERGONOMIE DE L'INTERFACE :*

Les interfaces doivent être simples et conviviales.

— La modularité de l'application : Avoir un code simple, facile à maintenir et à comprendre en cas de besoin.

### 3.3 Identification des cas d'utilisation

Un cas d'utilisation (use case) représente un ensemble de séquences d'actions réalisées par le système, et produisant un résultat observable intéressant pour un acteur particulier. Un cas d'utilisation modélise un service rendu par le système. Il exprime les interactions acteurs/système et apporte une valeur ajoutée « notable » à l'acteur concerné [4]. L'objectif principal est que l'ensemble des cas d'utilisation doit décrire exhaustivement les exigences fonctionnelles du système.

Nous présentons dans cette section le diagramme de cas d'utilisation puis la description de chacun d'eux.

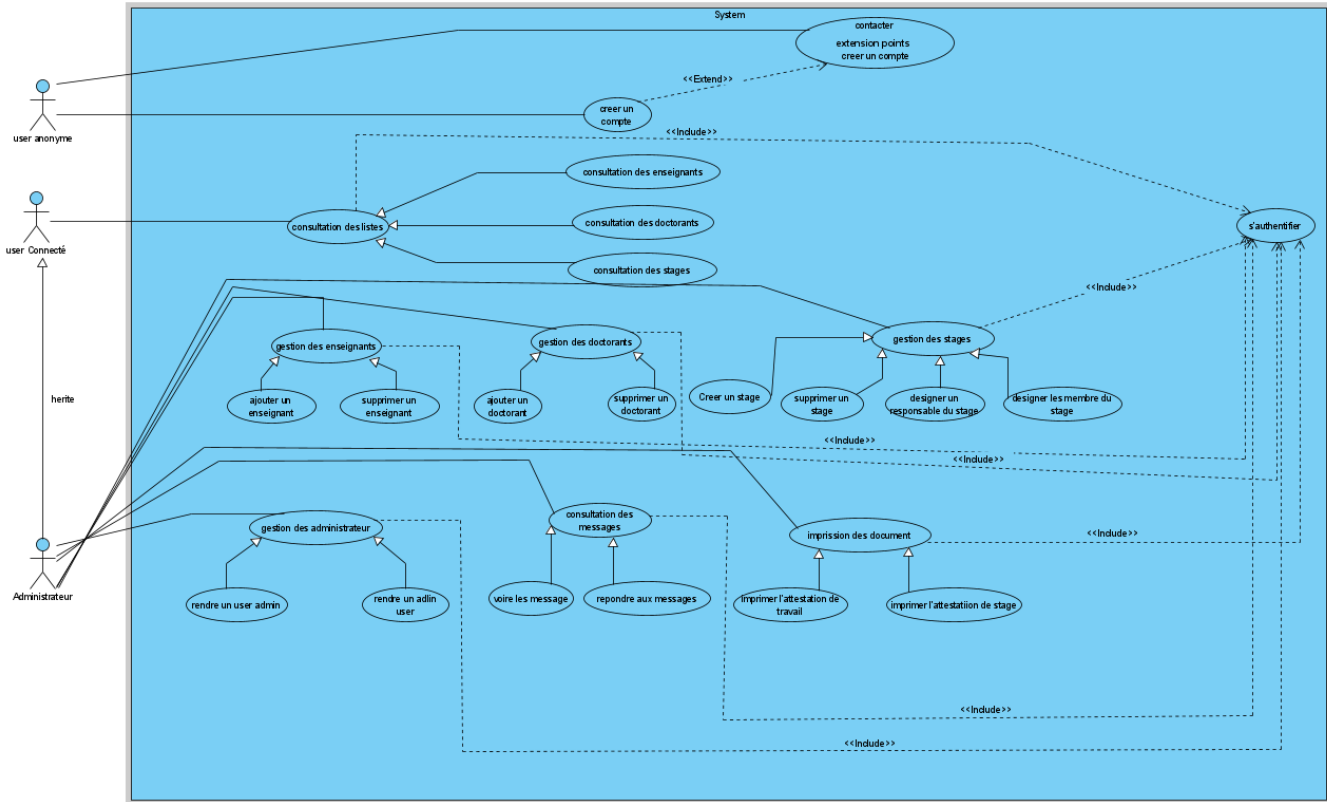


Figure 2:diagramme de cas d'utilisation

### 3.4 Description textuelle des cas d'utilisation

Dans cette section, nous présentons une description détaillée de chaque cas d'utilisation. Il s'agit de définir les scénarios possibles.

#### 3.4.1 Cas d'utilisation « S'authentification »

Nom	Authentification
<b>Acteur</b>	<ul style="list-style-type: none"> <li>• Enseignant</li> <li>• Doctorant</li> <li>• Administrateur</li> </ul>
<b>Scénario nominal</b>	<ul style="list-style-type: none"> <li>• L'utilisateur renseigne le formulaire d'authentification avec son login et mot de passe.</li> <li>• Le login et le mot de passe sont corrects.</li> </ul>
<b>Scénario alternatif</b>	<ul style="list-style-type: none"> <li>• L'utilisateur tente de s'authentifier en saisissant un login et un mot de passe.</li> <li>• Si le login ou/et le mot de passe sont incorrects:               <ul style="list-style-type: none"> <li>• Le système réaffiche le formulaire d'authentification.</li> <li>• L'utilisateur doit ressaisir son login et son mot de passe.</li> </ul> </li> </ul>

Tableau 2: Cas d'utilisation « Authentification »

## 3.4.2 Cas d'utilisation « Ajouter un enseignant »

Nom	Ajouter un enseignant
<b>Acteur</b>	<ul style="list-style-type: none"> <li>• Administrateur</li> </ul>
<b>Description</b>	<ul style="list-style-type: none"> <li>• Permet à l'administrateur d'ajouter un nouvel enseignant à la base de données.</li> </ul>
<b>Précondition</b>	<ul style="list-style-type: none"> <li>• Authentification.</li> </ul>
<b>Scénario nominal</b>	<ul style="list-style-type: none"> <li>• L'administrateur renseigne les champs du formulaire concernant l'enseignant.</li> <li>• Il valide et l'enseignant sera ajouté directement avec succès.</li> </ul>
<b>Scénario alternatif</b>	<ul style="list-style-type: none"> <li>• L'administrateur renseigne les champs du formulaire.</li> <li>• Matricule doit être un entier</li> <li>• Tous les champs doivent être remplis</li> <li>• Le système affiche un message d'erreur et réaffiche l'interface pour rectifier les informations saisies.</li> </ul>

Tableau 3 : Cas d'utilisation Ajouter un enseignant

**Remarque** : pour les cas d'utilisation « ajouter doctorant » et « créer un stage » suivent le même principe que « ajouter enseignant ».

## 3.4.3 Cas d'utilisation « Supprimer un enseignant »

Nom	Supprimer un enseignant
Acteur	<ul style="list-style-type: none"> <li>Administrateur</li> </ul>
Description	<ul style="list-style-type: none"> <li>Permet à l'administrateur de supprimer un enseignant déjà existant.</li> </ul>
Précondition	<ul style="list-style-type: none"> <li>Authentification.</li> </ul>
Scénario nominal	<ul style="list-style-type: none"> <li>L'administrateur accède à la liste des enseignants.</li> <li>Il sélectionne l'enseignant à supprimer.</li> <li>Un message de confirmation s'affiche.</li> <li>L'administrateur confirme la suppression.</li> <li>Le système supprime l'enseignant de la base de données.</li> <li>Un message de succès s'affiche indiquant que l'enseignant a été supprimé avec succès.</li> </ul>
Scénario alternatif	<ul style="list-style-type: none"> <li>L'administrateur tente de supprimer un enseignant.</li> <li>Si l'enseignant n'existe pas dans la base de données, le système affiche un message d'erreur.</li> <li>Si l'administrateur annule la suppression lors de la confirmation, l'opération est interrompue et aucune suppression n'est effectuée.</li> </ul>

Tableau 4: Cas d'utilisation supprimer un enseignant

**Remarque** : pour les cas d'utilisation « supprimer doctorant » et « annuler un stage » suivent le même principe que « supprimer enseignant ».

## 3.4.4 Cas d'utilisation « Consulter la liste des enseignants »

Nom	Consulter la liste des enseignants
<b>Acteur</b>	<ul style="list-style-type: none"> <li>• User connecté</li> </ul>
<b>Description</b>	<ul style="list-style-type: none"> <li>• Permet à un utilisateur authentifié de consulter la liste des enseignants enregistrés dans le système.</li> </ul>
<b>Précondition</b>	<ul style="list-style-type: none"> <li>• Authentification.</li> </ul>
<b>Scénario nominal</b>	<ul style="list-style-type: none"> <li>• L'utilisateur accède à la page de consultation des enseignants.</li> <li>• Le système affiche la liste complète des enseignants avec leurs informations principales (nom, prénom, spécialité, grade, etc.).</li> <li>• L'utilisateur peut parcourir la liste ou utiliser une barre de recherche pour filtrer les enseignants par critère (nom, spécialité, grade, etc.).</li> <li>• Si un critère de recherche est saisi, le système affiche les résultats correspondants.</li> </ul>
<b>Scénario alternatif</b>	<ul style="list-style-type: none"> <li>• L'utilisateur tente d'accéder à la liste sans être authentifié → Le système lui demande de s'authentifier.</li> <li>• L'utilisateur effectue une recherche avec un critère inexistant → Le système affiche un message indiquant qu'aucun enseignant ne correspond à la recherche.</li> </ul>

Tableau 5:cas d'utilisation consulter la liste des enseignants

**Remarque** : pour les cas d'utilisation « consulter liste doctorants » et « consulter liste stages » suivent le même principe que « consulter la liste des enseignants ».

## 3.4.5 Cas d'utilisation « Rechercher »

Nom	Rechercher
Acteur	<ul style="list-style-type: none"> <li>User connecté</li> </ul>
Description	<ul style="list-style-type: none"> <li>Permet à un utilisateur connecté d'effectuer une recherche parmi les listes utilisant des critères spécifiques.</li> </ul>
Précondition	<ul style="list-style-type: none"> <li>L'utilisateur doit être authentifié.</li> </ul>
Scénario nominal	<ul style="list-style-type: none"> <li>L'utilisateur accède à la page de consultation.</li> <li>Il saisit un ou plusieurs mots-clés dans la barre de recherche.</li> <li>L'utilisateur lance la recherche.</li> <li>Le système filtre et affiche les enseignants correspondant aux critères saisis.</li> <li>L'utilisateur consulte les résultats affichés.</li> </ul>
Scénario alternatif	<ul style="list-style-type: none"> <li>L'utilisateur tente d'effectuer une recherche sans être authentifié → Le système lui demande de s'authentifier.</li> <li>L'utilisateur saisit un critère de recherche inexistant → Le système affiche un message indiquant qu'aucun enseignant ne correspond à la recherche.</li> <li>L'utilisateur ne renseigne aucun critère de recherche → Le système affiche la liste complète des enseignants.</li> </ul>

Tableau 6: Cas d'utilisation rechercher

## 3.4.6 Cas d'utilisation « Imprimer attestation »

Nom	Imprimer l'attestation de travail
<b>Acteur</b>	<ul style="list-style-type: none"> <li>• L'administrateur</li> </ul>
<b>Description</b>	<ul style="list-style-type: none"> <li>• Permet à l'administrateur d'imprimer une attestation de travail ou de stage pour un enseignant ou un doctorant.</li> </ul>
<b>Précondition</b>	<ul style="list-style-type: none"> <li>• Authentification.</li> </ul>
<b>Scénario nominal</b>	<ul style="list-style-type: none"> <li>• L'administrateur accède à la page de des consultation des listes.</li> <li>• Il choisit l'enseignant ou le doctorant concerné dans la liste.</li> <li>• Le système génère automatiquement l'attestation avec les informations nécessaires.</li> <li>• Un aperçu de l'attestation s'affiche à l'écran.</li> <li>• L'administrateur valide l'impression.</li> <li>• Le système envoie le document à l'imprimante et affiche un message de confirmation.</li> </ul>
<b>Scénario alternatif</b>	<ul style="list-style-type: none"> <li>• L'administrateur tente d'accéder à la gestion des attestations sans être authentifié → Le système lui envoie un message d'erreur.</li> <li>• L'administrateur annule l'impression avant validation → L'opération est interrompue et aucune impression n'est effectuée.</li> </ul>

Tableau 7 cad d'utilisation imprimer une attestation de travail

### 3.4.7 Cas d'utilisation « consulter les message »

Nom	Consulter les messages
<b>Acteur</b>	<ul style="list-style-type: none"> <li>Administrateur</li> </ul>
<b>Description</b>	<ul style="list-style-type: none"> <li>Permet à l'administrateur de consulter les messages envoyés par les utilisateurs via la plateforme.</li> </ul>
<b>Précondition</b>	<ul style="list-style-type: none"> <li>Authentification.</li> </ul>
<b>Scénario nominal</b>	<ul style="list-style-type: none"> <li>L'administrateur accède à la section des messages dans le tableau de bord.</li> <li>Le système affiche la liste des messages reçus.</li> <li>L'administrateur sélectionne un message pour afficher son contenu en détail.</li> <li>L'administrateur peut répondre directement au message.</li> </ul>
<b>Scénario alternatif</b>	<ul style="list-style-type: none"> <li>L'administrateur tente d'accéder à la section des messages sans être authentifié → Le système lui demande de s'authentifier.</li> </ul>

Tableau 8:cas d'utilisation consulter les messages

## 3.5 Diagramme de séquence

L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés.

### 3.5.1 Diagramme de séquence « S'authentification »

Le diagramme de séquence représenté dans la figure 4 représente le scénario d'authentification d'un utilisateur.

Afin d'accéder aux fonctionnalités de notre application, tous les acteurs (Administrateur, enseignant et doctorant) doivent s'authentifier.

L'utilisateur doit tout d'abord remplir le formulaire afin de s'authentifier, et ceci en introduisant son login et son mot de passe puis valider l'envoi. Le système vérifie ensuite si les deux champs ont été bien saisis et bien conformes. Cependant, si l'utilisateur en oublie un champ ou qu'il est mal introduit, le système lui demande de ressaisir.

Finalement, et après avoir vérifié que l'utilisateur possède les droits d'accès, le système renvoie la page correspondante sinon il affiche un message d'erreur.

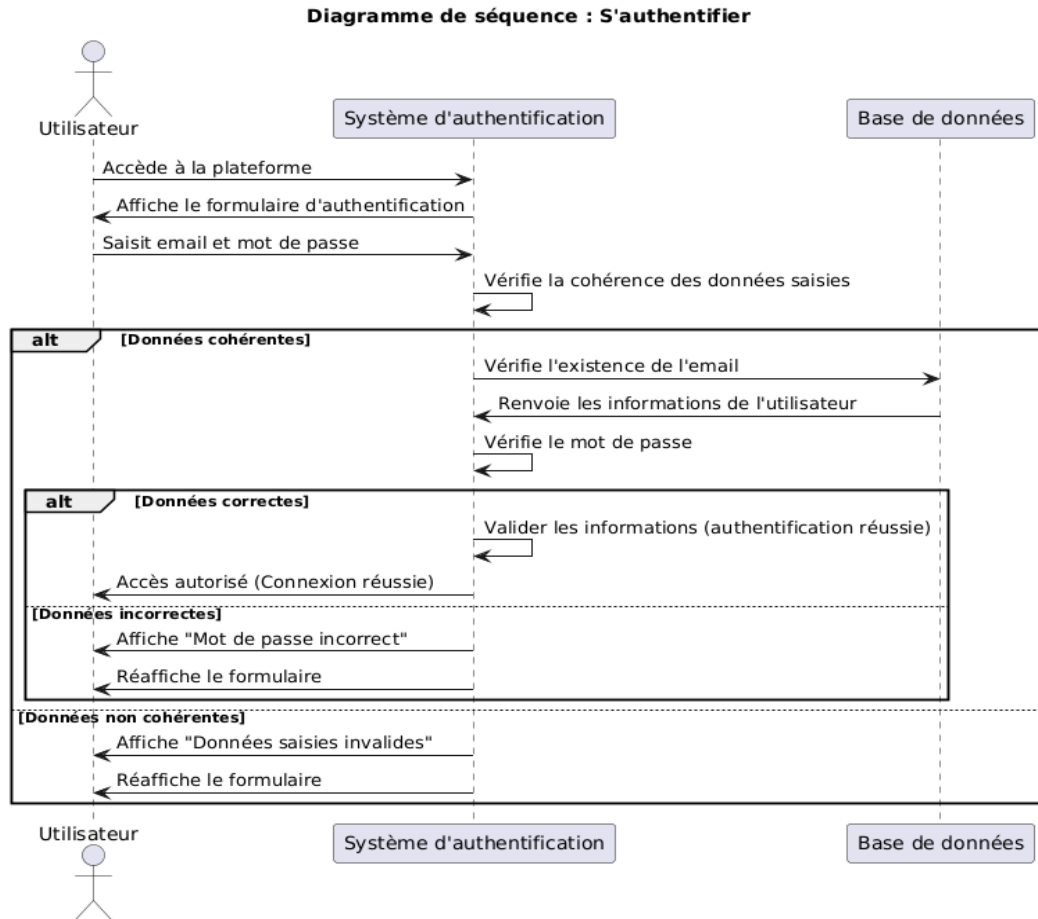


Figure 3:diagramme de séquence Authentification

### 3.5.2 Diagramme de séquence « Ajouter un enseignant »

Le diagramme de séquence représenté dans la figure 5 représente le scénario d'enregistrement d'un enseignant de la part de l'administrateur.

L'administrateur remplit le formulaire d'ajout d'un enseignant et le valide. Le système vérifie la conformité des champs. Et tant qu'ils ne sont pas sous format valide, il demandera de les introduire.

Finalement et si les données sont cohérentes, le système valide l'enregistrement met à jour la page liste des enseignants, sinon il renvoie un message d'erreur.

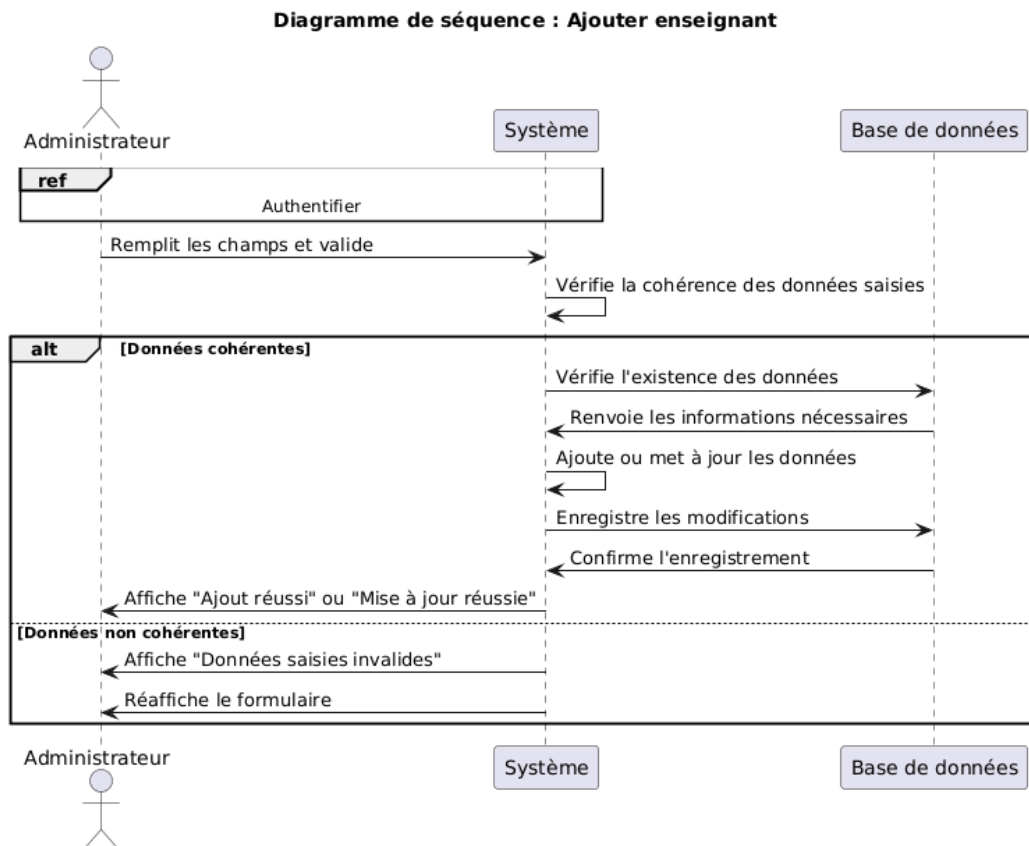


Figure 4: diagramme de séquence "ajouter un enseignant"

**Remarque :** Le diagramme de séquence « ajouter un doctorant » et « créer un stage » suivent le même principe que celui d'ajout des enseignants.

### 3.5.3 Diagramme de séquence « Supprimer un enseignant »

Ce diagramme de séquence Figure 8 représente le scénario de suppression d'un enseignant de la part de l'administrateur.

Après authentification, l'administrateur peut bel et bien supprimer un enseignant. L'administrateur fait une recherche sur un enseignant, ensuite clique sur l'Icon supprimer afin de confirmer la suppression. Finalement, le système met à jour le liste après le succès de la suppression.

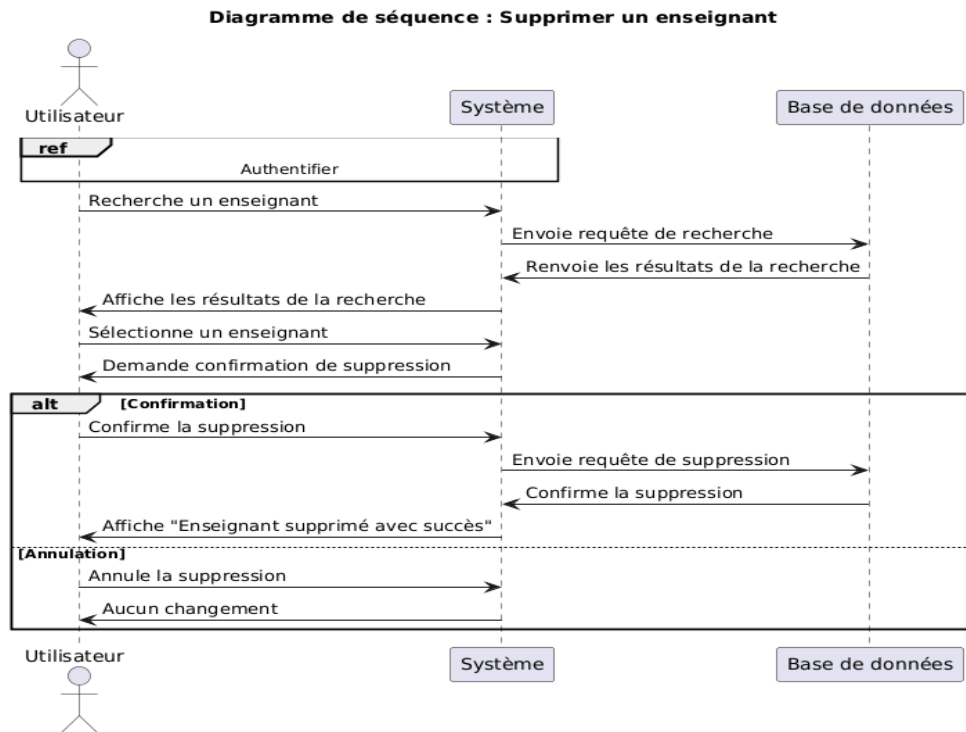


Figure 5: Diagramme de séquence supprimer un enseignant

**Remarque :** Le diagramme de séquence « supprimer un doctorant » suit le même principe que celui « supprimer un enseignant ».

### 3.5.4 Diagramme de séquence « annuler un stage »

Ce diagramme de séquence Figure 10 représente le scénario d'annulation d'un stage de la part de l'administrateur.

Après authentification, l'administrateur peut annuler un stage. Il clique sur l'Icon supprimer afin de confirmer la suppression. Finalement, le système met à jour le liste après le succès de la suppression.

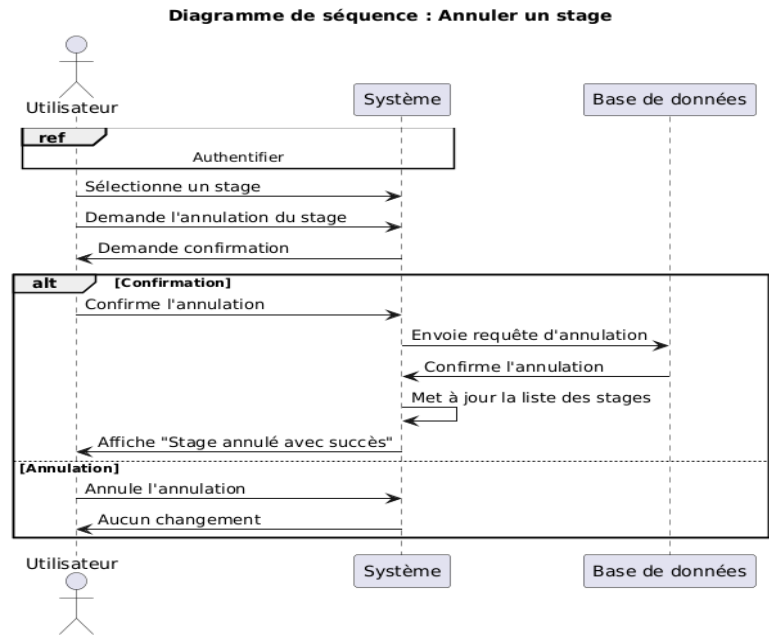


Figure 6: Diagramme de séquence annuler un stage

### 3.5.5 Diagramme de séquence « Imprimer une attestation »

Ce diagramme de séquence figure 11 représente le scénario d'impression d'attestation de travail de la part de l'administrateur.

Après authentification, L'administrateur pourra imprimer l'attestation et ceci en cliquant sur l'Icon d'impression. Ensuite, le système répond en affichant l'interface d'impression correspondante au navigateur. Enfin, il valide par la touche imprimer après d'avoir choisi les types d'impression.

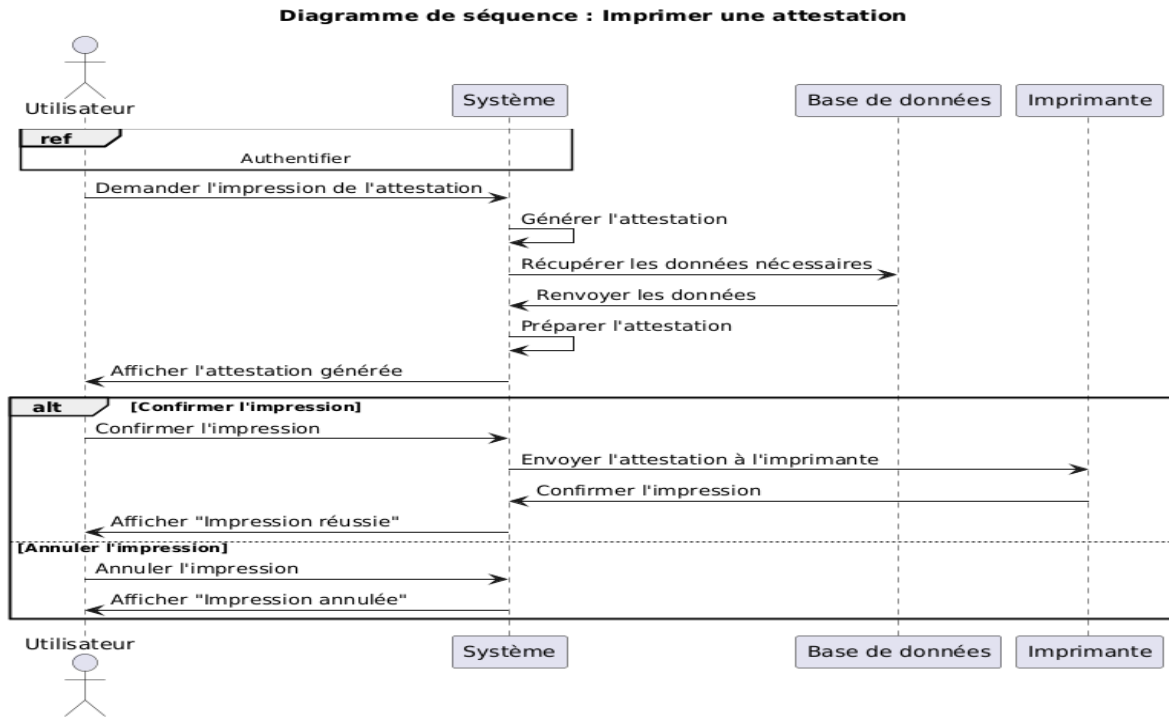


Figure 7: Diagramme de séquence « Imprimer l'attestation »

### 3.5.6 Diagramme de séquence « rechercher »

Ce diagramme de séquence figure 13 représente le scénario de recherche de la part d'User.

Après l'authentification. L'utilisateur saisit les critères de recherche. Il soumet la requête de recherche.

Le système vérifie les critères de recherche et exécute la recherche dans la base de données. Le système renvoie les résultats de la recherche à l'utilisateur. L'utilisateur consulte les résultats.

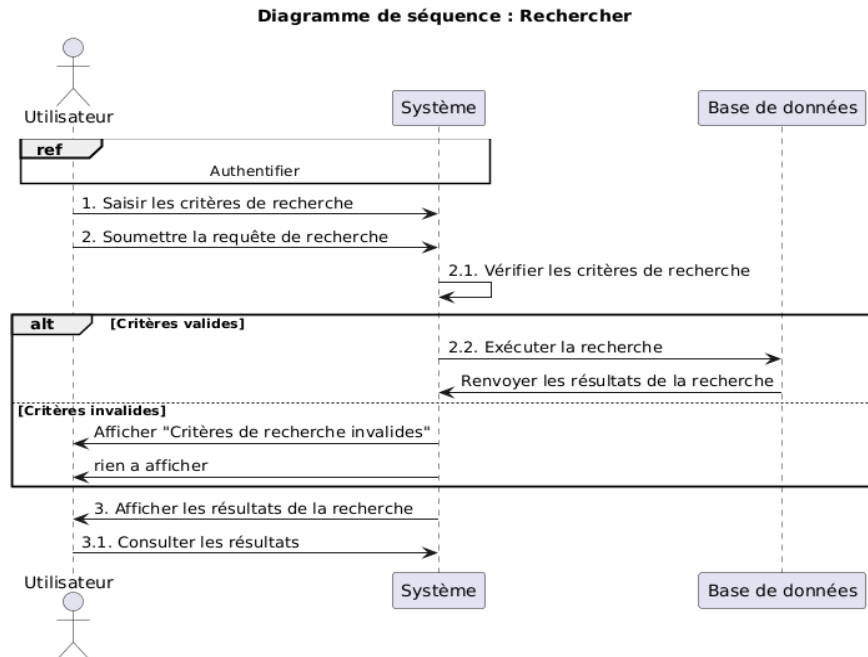


Figure 8:diagramme de séquence rechercher

### 3.6 Dictionnaire de données

Le dictionnaire des données représenté dans le tableau 16 Le dictionnaire des données est un document qui regroupe toutes les données recensés, spécifiées et recueillies, et que nous aurons à conserver dans notre base de données (et qui figureront donc dans le diagramme de classe).

Nom de la classe	Codification	Désignation	Type	Taille
<b>User</b>	Id	Identifiant de l'utilisateur	Integer	20
	Name	Nom et prénom	Varchar	255
	Email	Email de l'utilisateur	Varchar	255
	Password	Mot de passe de l'utilisateur	Varchar	255
<b>Enseignant</b>	Id	Identifiant de l'enseignant	Integer	20
	Nom	Nom de l'enseignant	Varchar	255
	Prenom	Prénom de l'enseignant	Varchar	255
	Matricule	Matricule de l'enseignant	Integer	11
	Date_naissance		Date	

	Adresse	Date de naissance de	Varchar	255
	Departement	l'enseignant	Varchar	255
	Specialite	Adresse de l'enseignant	Varchar	255
	Email	Département affilié	Varchar	255
	Sexe	Spécialité affiliée	Char	10
	Grade	Email de l'enseignant	Varchar	255
		Le sexe soit M ou F		
		Grade de l'enseignant		
<b>Doctorant</b>	Id	Identifiant du doctorant	Integer	20
	Nom	Nom du doctorant	Varchar	255
	Prenom	Prénom du doctorant	Varchar	255
	Matricule	Matricule du doctorant	Integer	11
	Date_naissance	Date de naissance du doctorant	Date	
	Adresse	Adresse du doctorant	Varchar	255
	Departement	Département affilié	Varchar	255
	Specialite	Spécialité affilie	Varchar	255
	Email	Email du doctorant	Varchar	255
	Sexe	Sexe soit M ou F	Char	255
	These	These du doctorant	Varchar	255
	Directeur_these	Directeur qui encadre le	Varchar	255
		doctorant	Varchar	255
	Faculte	Faculte affilie	year	4
	Annee_inscription	Annee d'inscription du doctorant		
<b>Members</b>	Id	Identifiant membre	Integer	20
	Nom	Nom du membre réfère à un	Varchar	255
		enseignant ou un doctorant		
	Type	Soit enseignant ou doctorant		
<b>Stages</b>	Id	Identifiant stage	Integer	20
	Intitule	L'intitule du stage	Varchar	255
	Date_aller	Date d'aller (début)	Date	
	Date_retour	Date retour (fin)	Date	

	Duree	La durée du stage	Integer	11
	Responsable_id	Responsable qui réfère a id enseignant	Integer	20
<b>Stage_membre</b>	Id	Identifiant du membre stage	Integer	20
	Stage_id	Identifiant du stage qui réfère a id dans stage	Integer	20
	Member_id	Identifiant du membre qui réfère à id dans members	Integer	20
<b>contact</b>	Id	Identifiant du message	Integer	20
	Nom	Nom de l'émetteur	Varchar	255
	Email	Email de l'émetteur	Varchar	255
	Message	Message envoie par l'émetteur	text	

Tableau 9: Dictionnaire de données

### 3.6.1 Liste Des Règles De Gestion

1. Un enseignant peut être responsable pour plusieurs stages.
2. Un enseignant peut participer à un stage.
3. Un doctorant peut participer à un stage.
4. Un stage peut être fait par plusieurs membre.
5. Un administrateur peut voire plusieurs messages
6. Un stage peut être planifié que pour un seul responsable
7. Un doctorant peut faire qu'une seule thèse
8. Un enseignant peut être directeur de thèse pour plusieurs doctorants
9. Un doctorant peut être dirigé par un seul enseignant

## 3.7 Diagramme de classe de conception

Le diagramme de classes présenté dans la figure 14 de notre solution informatique, chaque classe reflète directement une entité du monde réel avec les informations que contient chaque entité (attributs) et les relations entre elles. Nous avons également introduit quelques méthodes à

certaines de ces classes pour effectuer les traitements nécessaires. La figure 14 représente le diagramme de classes de notre solution informatique.

**SA COMPOSITION :**

- **Classe** : représente une description abstraite (à l'aide d'un rectangle) d'un groupe d'objets ayant les mêmes caractéristiques, tel que les classes utilisateur, enseignant, stage...etc.
- **Attributs** : est un type d'information tel (id\_stage) contenu dans une classe. Chaque attribut est caractérisé par son nom, sa visibilité, son type, sa valeur initiale et ses propriétés.
- **Opérations** : une opération est une fonction applicable aux objets d'une classe. Une opération permet de décrire le comportement d'un objet, et une méthode est l'implémentation d'une opération. Par exemple ajouter\_enseignant() dans la classe administrateur.

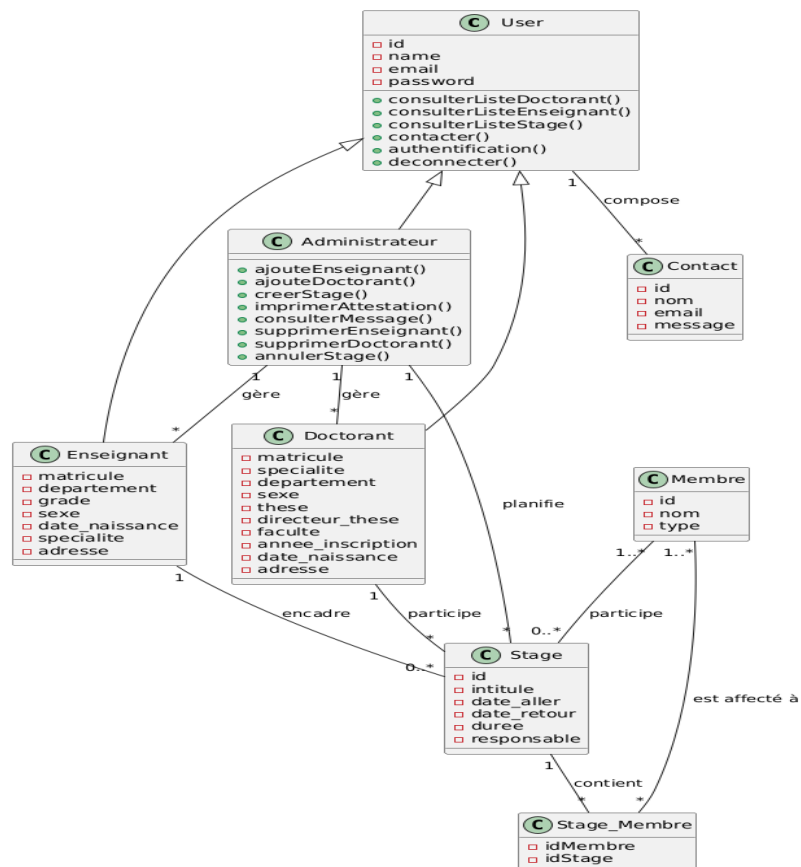


Figure 9:diagramme de classe

### 3.8 Passage au modèle relationnel

Nous appelons modèle relationnel un ensemble de concepts permettant de formaliser logiquement la description d'articles de fichiers plats qui sont des fichiers de données contenant un enregistrement par ligne, indépendamment de la façon dont ils sont physiquement stockés dans une mémoire numérique. Le modèle relationnel inclut des concepts pour la description de données, ainsi que des concepts pour la manipulation de données.

#### 3.8.1 Fondements

Le modèle relationnel permet de représenter les données que l'on va gérer à l'aide d'un très petit nombre de concepts très simples :

**LES DOMAINES DE VALEURS** : représente l'ensemble des valeurs d'un attribut.

**LES RELATIONS OU TABLES** : tableau de deux dimensions où les colonnes représentent les domaines et où les lignes contiennent les tuples.

**LES CLES** : certaines cases doivent avoir des valeurs uniques et non nulles, tel une notion d'identifiant.

**LES CLES ETRANGERES** : attribut qui est représenté en une clé primaire dans une autre entité et ceci afin de gérer les relations entre plusieurs tables.

#### 3.8.2 Les Règles De passage

Le passage du diagramme de classe au modèle relationnel ne se fait pas au hasard, il existe un certain nombre de règles qui permettent de réaliser cette opération. C'est d'ailleurs sur ces règles que s'appuient les outils de modélisation afin de réaliser ces opérations.

**LES OBJETS :**

- Chaque objet devient une table.
- Propriété de l'objet devient un attribut de la table.
- Identifiant de l'objet devient une clé primaire de la table.

**ASSOCIATION SANS PROPRIETES PROPRES :**

- Cardinalités (0,1) ou (1,1) vers (0, n) ou (1, n) : la clé de la relation avec la cardinalité (0,1) ou (1,1) migre vers la relation à la cardinalité (0,n) ou (1,n), et l'association disparaît. La clé créée est appelé "clé étrangère".
- Cardinalités (0, n) ou (1, n) vers (0, n) ou (1, n) : l'association se transforme en une relation avec comme clé la concaténation des clés des 2 relations.
- Associations avec propriétés propres : l'association se transforme en une relation ayant comme clé la concaténation des clés des relations associées à chaque individu, les attributs de cette nouvelle relation sont les propriétés propres de l'association précédente.

**TRANSFORMATION DE L'HERITAGE :**

Trois décompositions sont possibles pour traduire une association d'héritage en fonction des contraintes existantes :

- Décomposition par distinction : Il faut transformer chaque sous-classe en une relation. La clé primaire de la surclasse migre dans la (les) relation(s) issue(s) de la (des) sous-classe(s) et devient à la fois clé primaire et clé étrangère.
- Décomposition descendante (push-down) : Dans le cas contraire, il faut faire migrer tous ses attributs dans la ou les relation(s) issue(s) de la (des) sous-classe(s) dans la (les) relation(s) issue(s) de la (des) sous-classe(s).
- Décomposition ascendante (push-up) : Il faut supprimer la (les) relation(s) issue(s) de la (des) sous-classe(s) et faire migrer les attributs dans la relation issue de la surclasse.

**3.8.3 Modèle Relationnel**

En appliquant les règles de transformation d'un diagramme de classe vers un modèle relationnel citées précédemment, nous avons abouti au schéma relationnel suivant :

**User** ( id, name,password, email)

**Enseignant**(Id,nom,prenom,matricule,date\_naissance ,adresse ,departement ,specialite,email,sexe ,grade)

**Doctorant**(Id,nom,prenom,matricule,date\_naissance ,adresse ,departement ,specialite,email,sexe ,these ,#id\_enseignant,faculte,annee\_inscription)

**Stage**(id ,intitule,date\_aller ,date\_retour,duree,#responsable\_id)

**Membre**(id,#nom,type)

**Stage\_membre**(id,#id\_stage,#id\_membre)

**Contact**(id ,nom,email,message)

## 4 Conclusion

Dans ce chapitre, nous avons commencé par l'analyse des besoins, ensuite nous avons présenté les cas d'utilisation, les diagrammes de séquence qui leurs correspondent ainsi que le diagramme de classes. Enfin, nous avons terminé par le modèle relationnel de données qui nous permet d'avoir le schéma de la base de données de l'application. Dans le chapitre suivant, nous allons présenter l'implémentations de notre application et les outils et les environnements utilisés. Des copies d'écran sont ajoutés pour montrer les fonctionnalités de notre application.

# CHAPITRE 3

# IMPLÉMENTATION

## 1 Introduction

Tout développement de projet informatique nécessite le choix des technologies adéquates à son implémentation. Et c'est en définissant les outils de développement que nous débutons ce dernier chapitre. Par la suite, nous allons présenter les interfaces de notre application Web afin de mettre en évidence leurs aspect pratique et intuitive qui nous ont été l'un de nos principaux objectifs. Rappelons que notre projet consiste à réaliser un outil de planification et de gestion académique , ce type d'application repose sur une architecture client-serveur. Dans notre cas, le client est le navigateur Web, et le programme exécutant sur un ordinateur distant représente le serveur.

## 2 Outils de développement

### 2.1 Environnement machine

#### 2.1.1 MacBook Air

Système d'exploitation : Macintosh.

Processeur : Intel(R) Core (TM) i5-5250U CPU @ 1.6GHz.

Mémoire : 4 GB.

### 2.1.2 Dell Latitude 7872

Système d'exploitation : kali Linux 2023.

Processeur : Intel(R) Core (TM) i7-6600U CPU @ 2.6GHz.

Mémoire : 16 GB.

### 2.1.3 Acer Aspire

Système d'exploitation : Windows 11 Professionnel.

Processeur : Intel(R) Core (TM) i5-5000U CPU @ 2.6GHz.

Mémoire : 8 GB.

## 2.2 Environnement logiciel

Tout projet informatique nécessite l'utilisation de technologies performantes afin d'assurer une bonne implémentation des besoins définis dans les phases précédentes. Et pour le développement de cette application, nous avons utilisé les outils logiciels suivants :

### 2.2.1 XAMPP

XAMPP, acronyme de Cross-Platform (X), Apache (A), MySQL (M), PHP (P) et Perl (P) est un ensemble de logiciels libres, open source et multi-plateforme développé par des amis Apache. Il dispose des distributions pour le serveur Apache, MariaDB, PHP et Perl. Grâce à la souplesse de son utilisation, XAMPP permet de développer des applications de script côté serveur basées sur Hypertext Preprocessor (PHP) et Perl sans devoir utiliser un serveur Web distant.



Figure 10:logo XAMPP

Parmi les outils principaux de XAMPP, on trouve :

❖ *APACHE*

Apache est un logiciel de serveur Web gratuit, open source et maintenu par Apache Software Foundation, apparu il y a 25 ans, il a su garder la première place en question de fiabilité. Il permet principalement aux propriétaires de sites Web de servir du contenu à travers le Web. Si quelqu'un demande des fichiers, des images ou des documents en utilisant son navigateur, il servira ces fichiers aux clients utilisant des serveurs 'Hypertext Transfer Protocol (HTTP).



*Figure 11:logo d'APACHE*

❖ *MYSQL :*

Développé par une société suédoise "MySQL AB" en 1995, il est aujourd'hui le système de gestion de base de données relationnelles (SGBD) le plus répandu de par sa robustesse et sa sûreté sur le Web. Il permet la gestion de base de données relationnelle (SGBDR), une organisation facile, efficace et rapide des informations. MySQL est un système Open source, multiutilisateurs, multiplateformes et s'intègre facilement à plusieurs serveurs Web.

Le SQL dans "MySQL" signifie "Structured Query Language" qui le langage standard pour les traitements de bases de données.



*Figure 12:logo MySQL*

### 2.2.2 Visual Code studio

Première sortie	29 avril 2015
Développé par	Microsoft
Système d'exploitation	Windows, Mac, Linux
Langue	Multilingues
Dernière version	1.90 en Mai 2024

Tableau 10:Fiche technique de Visual Studio Code

Visual Studio Code est un éditeur de code source gratuit et open source, léger et puissant. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et possède un riche écosystème d'extensions pour d'autres langages (tels que C ++, C #, Java, Python, PHP, Go et plus encore) et des runtimes (tels que .NET et Unity).

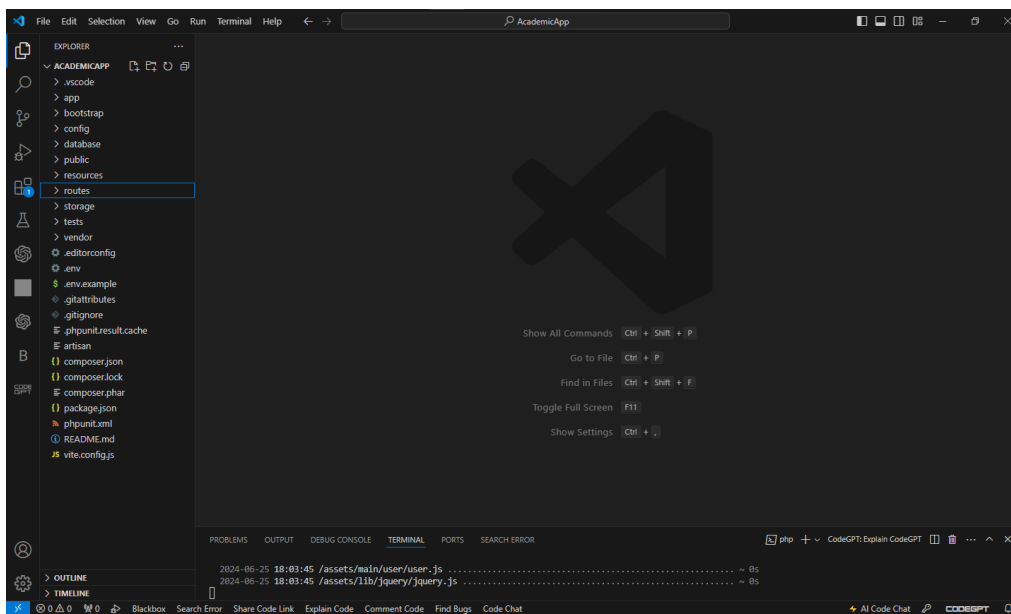


Figure 13: Interface Visual Studio Code

### 2.2.3 La bibliothèque logicielle PHPMailer :

La bibliothèque logicielle PHPMailer compte plus de 9 millions d'utilisateurs dans le monde aujourd'hui. Initialement créée en 2001 par Brent R. Matzelle dans le cadre d'un projet SourceForge. Ensuite, en 2004 il à été repris par Marcus Boiton et Andy Prevost et est devenu un projet d'incubateur pour Apache en 2010. Enfin, et en 2013 PHPMailer passe à l'organisation PHPMailer de GitHub.



Figure 14:Logo PHPMailer

### 2.2.4 Composer

Composer est un outil indispensable pour tout développeur PHP souhaitant gérer efficacement les dépendances de ses projets. Grâce à ses nombreuses fonctionnalités et à sa simplicité d'utilisation, Composer a révolutionné la manière dont les développeurs PHP gèrent leurs bibliothèques et packages.



Figure 15:logo composer

## 2.3 Environnement de programmation

### 2.3.1 PHP

PHP acronyme de Hypertext Processor, est un langage de scripts généraliste, gratuit et open source, conçu principalement pour le développement d'applications web. Sa syntaxe est inspiré du C, du Java et de Perl, Il est peu typé, et il permet de communiquer avec d'autres protocoles comme Lightweight Directory Access Protocol (LDAP), Internet Message Access Protocol (IMAP), Simple Network Management Protocol (SNMP), Network News Transfer Protocol (NNTP), Post Office Protocol 3 (POP3) ou encore HTTP. PHP est un langage de script côté serveur, ce qui fait qu'il est capable de réaliser tout ce qu'un script Common Gateway Interface (CGI) quelconque peut faire, comme collecter des données de formulaire, générer du contenu dynamique, ou gérer des cookies.



*Figure 16:Logo PHP*

PHP permet aussi de :

- Supporter un grand nombre de bases de données comme que le MySQL.
- Programmer de manières différentes, à savoir en orienter objet ou en procédural.
- Utiliser des objets Java tels des objets PHP de manière transparente.

### ❖ *Laravel*

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet, créé par Taylor Otwell en juin 2011, Laravel est distribué sous licence MIT.



*Figure 17:logo Laravel*

### *Fortify :*

**Laravel Fortify** est une implémentation backend d'authentification indépendante du frontend pour Laravel. Fortify enregistre les routes et les contrôleurs nécessaires à la mise en œuvre de toutes les fonctionnalités d'authentification de Laravel, notamment la connexion, l'enregistrement, la réinitialisation du mot de passe, la vérification des e-mails, etc.



*Figure 18:logo Fortify laravel*

### 2.3.2 CSS

Cascading Style Sheets (CSS) est un langage de conception qui vise à simplifier la représentation des pages Web, à savoir les couleurs, les tailles, les polices de caractères . . . etc. On peut modifier l'apparence de notre site sans avoir à toucher sa structure et ceci grâce aux règles CSS, ce qui facilite la construction ainsi que la maintenance des applications web. Les standards définissant le code CSS sont publiés par le World Wide Web Consortium (W3C).



*Figure 19:logo CSS*

### 2.3.3 Bootstrap

Initialement créé par un designer et un développeur sur Twitter, Bootstrap est devenu l'un des Framework frontend et des projets open source les plus populaires pour le développement des projets mobile-first et responsives sur le Web, il intègre HTML, CSS et JavaScript. Pour des typographies, des boutons et des interfaces de navigations, Bootstrap fournit des outils avec des styles déjà en place. La dernière version est sortie en Mai 2020.



*Figure 20:logo Bootstrap*

### 2.3.4 HTML

HTML est un langage de balise permettant le codage des pages Web, la structuration sémantique, la mise en forme des interfaces des sites et l'inclusion des ressources multimédias telles que les images, les formulaires de saisie, et les programmes informatiques. Il permet de créer des

documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation et des formats de présentation (feuilles de style en cascade).



*Figure 21:logo HTML*

### 2.3.5 JavaScript

JavaScript est un langage de programmation léger qui ne nécessite aucune compilation, utilisé principalement pour l'interactivité des pages Web, par ailleurs il n'implémente pas la notion d'héritage et possède une notion de classe très simpliste. Conçu par Brendan Eich en décembre 2015. Il est profondément intégré à HTML, à tel point que sans HTML, il n'a pas vraiment de raison d'être. En JavaScript, on se contente de donner un peu de vie à HTML.



*Figure 22:logo JavaScript*

## 2.4 Schéma de navigation de l'application

Le schéma de navigation d'un site représente les différentes pages de celui-ci, organisées logiquement et hiérarchiquement sous forme d'un arbre. La première page doit être la page d'accueil (la racine), et les autres pages apparaissent en suivant un ordre logique. L'arborescence permet aux utilisateurs de mieux comprendre et bien se situer dans la plateforme, ce qui permet la facilité et l'efficacité de l'utilisation. Après l'analyse des besoins des utilisateurs, nous avons structuré notre plateforme et pu distinguer ses différentes sections et pages tel illustré dans la Figure 28.

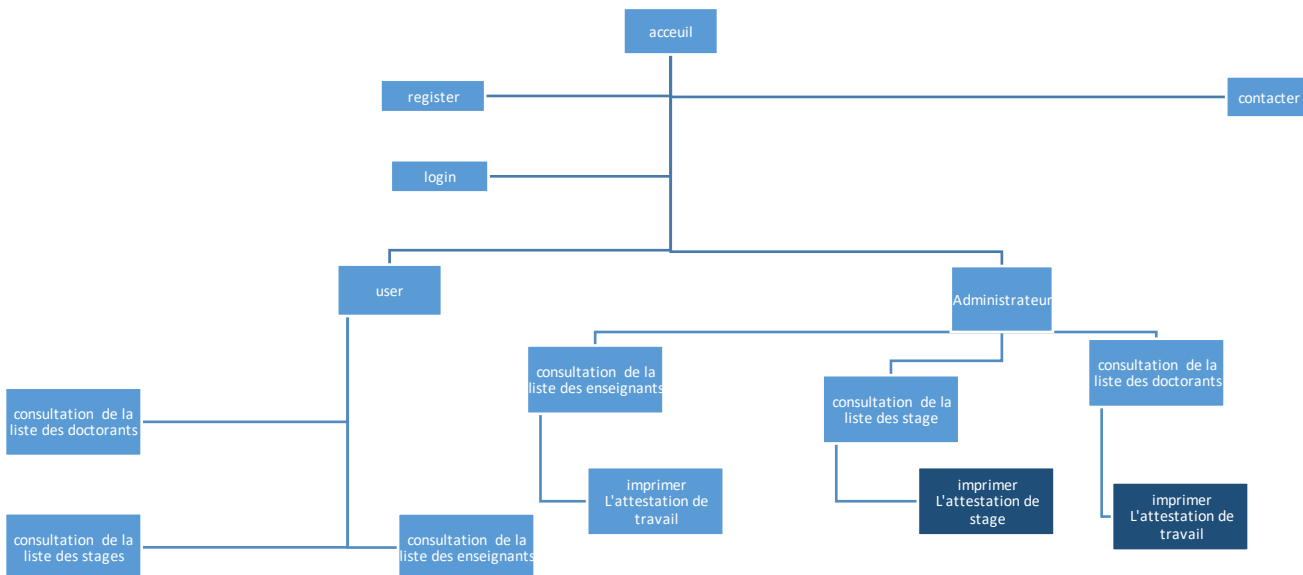


Figure 23: Schéma de navigation de l'application

### 2.4.1 Accueil

Cette page est un élément primordial de l'application, elle affiche à l'ouverture de l'application. Elle est composée d'un slideshow. Chaque slide a sa fonctionnalité, les fonctionnalités sont présentées par un login, register et contact.

La figure suivante représente l'interface d'accueil de la plateforme.

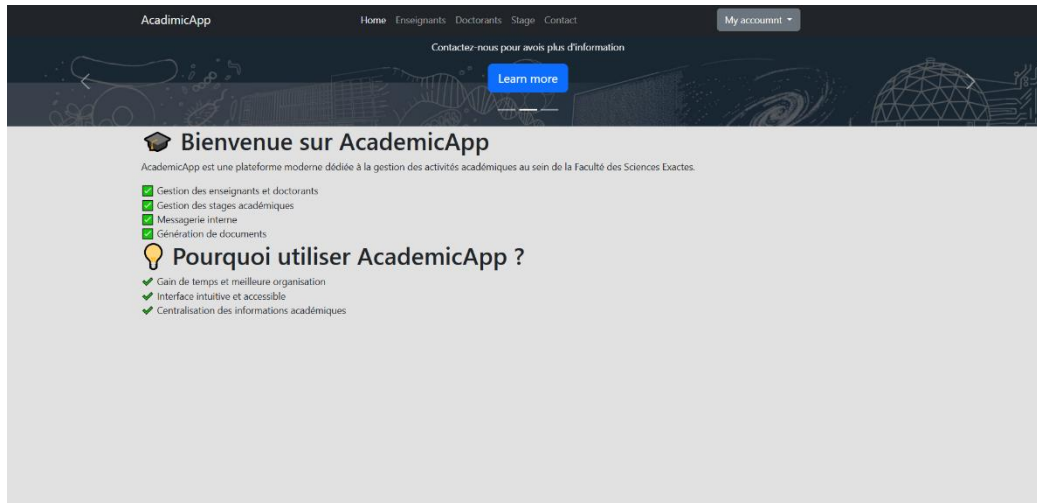


Figure 24:page D'accueil

## 2.4.2 Register

Cette page est la page d'enregistrement des Utilisateurs, L'utilisateur introduit ses données clique sur « s'inscrire ». Si les informations saisies sont selon les règles il sera inscrit et dirigé vers la page Dashboard, sinon il affiche un message d'erreur.

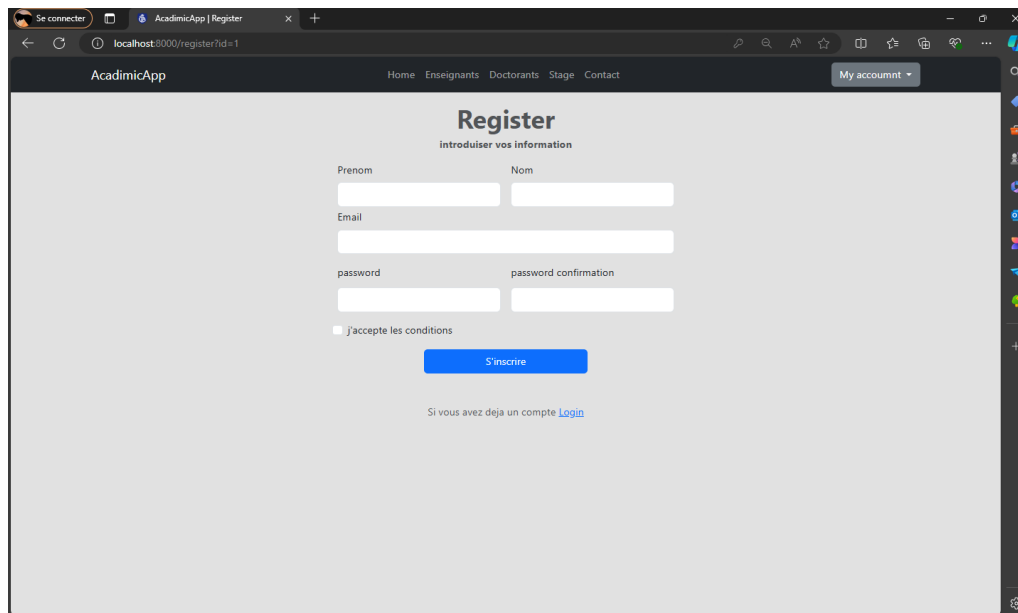


Figure 25:page Register

### 2.4.3 Login

Cette page est la page d'authentification de l'application, L'utilisateur introduit ses données clique sur « se connecter ». Si les informations saisies sont selon les règles il sera authentifié et dirigé vers la page Dashboard, sinon il affiche un message d'erreur.

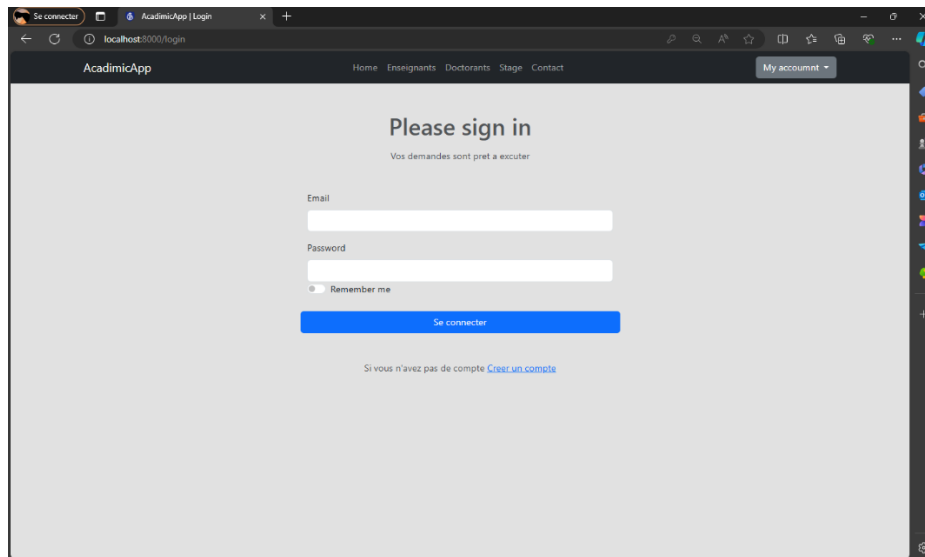


Figure 26:page login

### 2.4.4 Page Administrateur

Le premier utilisateur à se connecter à la plateforme est l'administrateur qui a le rôle est la gestion la liste des enseignants, des doctorants et des stages, et sa page du profile est illustrée dans la figure 32.

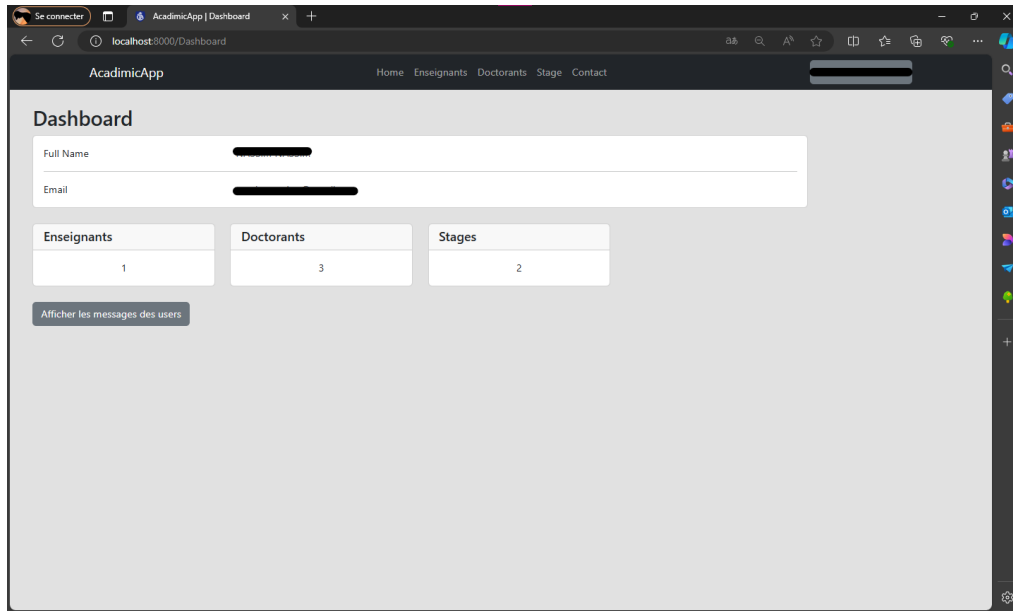


Figure 27:profile administrateur

### 2.4.5 Page profile

Cette page affiche les informations de l'utilisateur authentifié, l'utilisateur peut voir le nombre total des enseignant, doctorants et des stages.

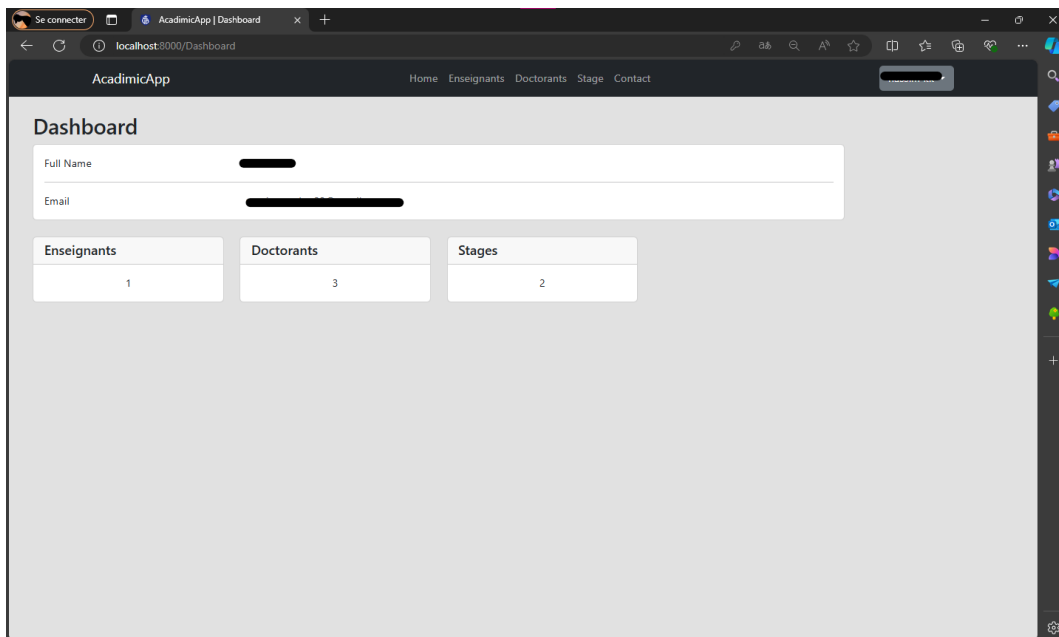


Figure 28:profile simple user

Dans la barre de navigation on trouve des liens :

- **Home** : pour aller à la page d'accueil
- **Enseignants** : pour aller à la page liste des enseignants
- **Doctorants** : pour aller à la page liste des doctorants
- **Stages** : pour aller à la page liste des stages
- **Contact** : pour aller à la page contact

### 2.4.6 Page liste des enseignants

Cette page affiche la liste des enseignants dans un tableau, on trouve aussi une barre de recherche .la liste s'affiche en fonction du mot tapé. L'administrateur peut ajouter un enseignant en cliquant sur le bouton « ajouter enseignant », et il peut supprimer en cliquant sur l'icône corbeille, et pour imprimer une attestation de travail il faut cliquer sur l'icône fichier.un message s'affiche à annonçant le succès de l'opération effectuée.

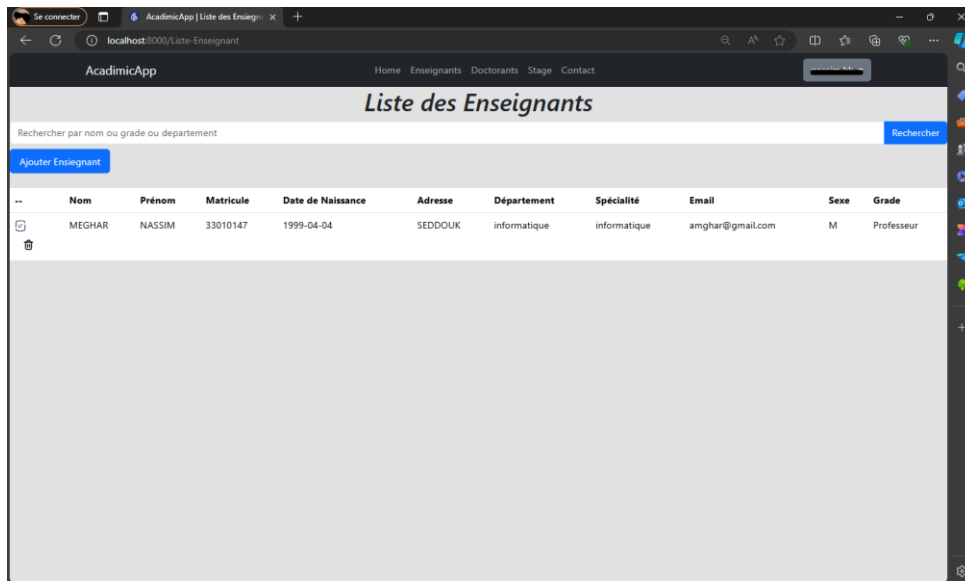


Figure 29:page liste des enseignants

### 2.4.7 Page liste des doctorants

Cette page affiche la liste des doctorants dans un tableau, on trouve aussi une barre de recherche .la liste s'affiche en fonction du mot tapé. L'administrateur peut ajouter un doctorant en cliquant sur le bouton « ajouter doctorants », et il peut supprimer en cliquant sur l'icône

corbeille, et pour imprimer une attestation de travail il faut cliquer sur l'icône fichier ,un message s'affiche à annonçant le succès de l'opération effectuée.

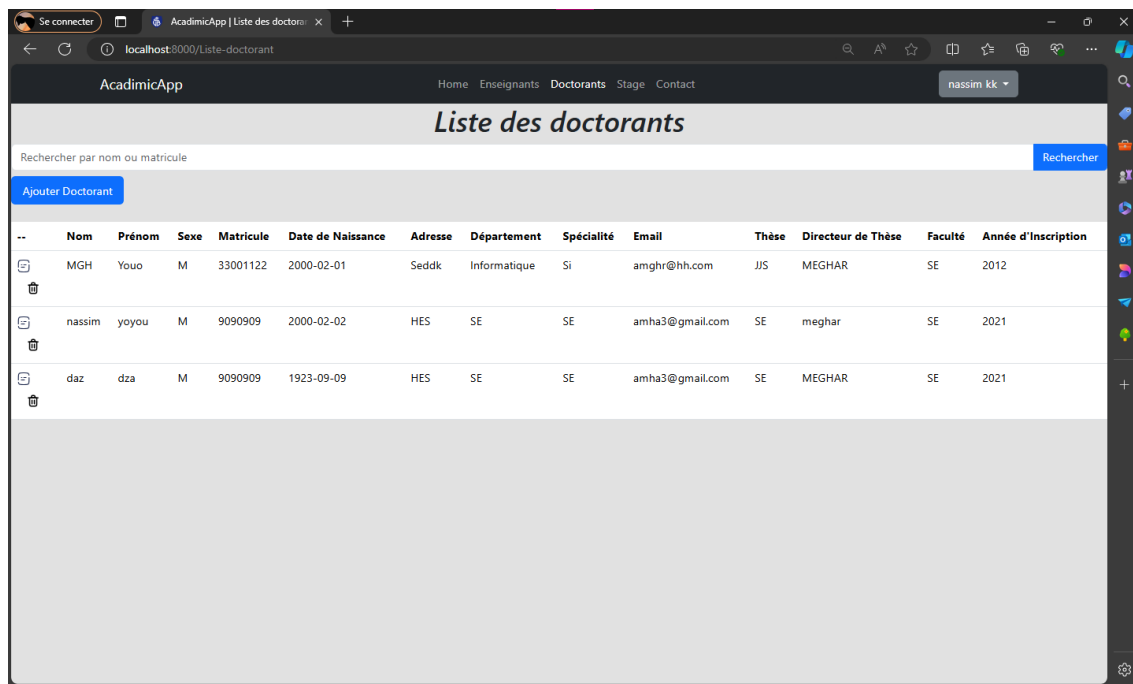
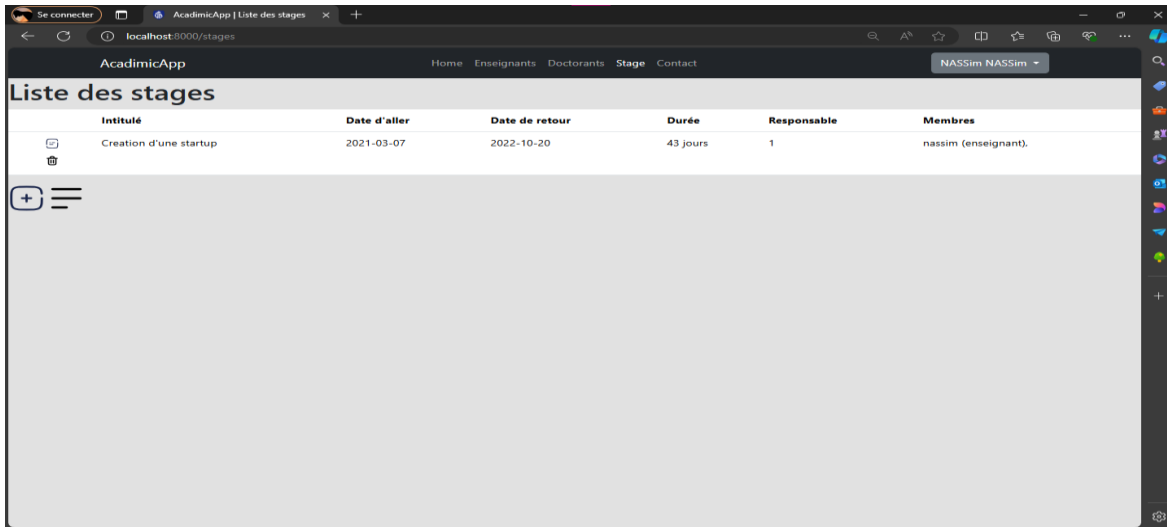


Figure 30:page liste des doctorants

#### 2.4.8 Page liste des stages

Cette page affiche la liste des stages dans un tableau. L'administrateur peut planifier un stage en cliquant sur l'icône [+], et il peut supprimer en cliquant sur l'icône corbeille, et pour imprimer une attestation de stage il faut cliquer sur l'icône fichier. Pour afficher les enseignant responsable l'utilisateur peut cliquer sur l'icône détails. un message s'affiche à annonçant le succès de l'opération effectuée.

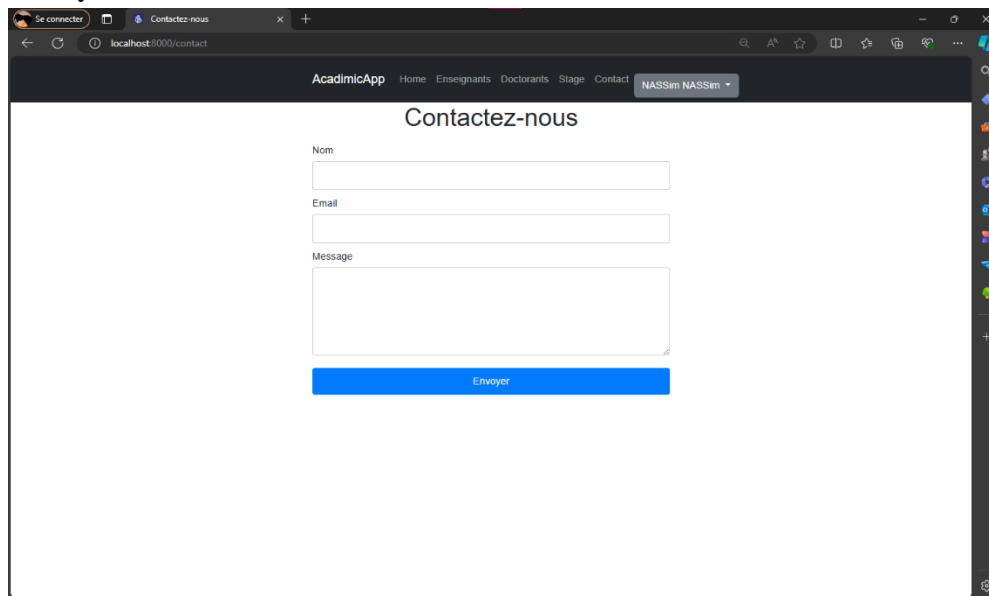


Intitulé	Date d'aller	Date de retour	Durée	Responsable	Membres
Creation d'une startup	2021-03-07	2022-10-20	43 jours	1	nassim (enseignant).

Figure 31:page liste des stages

### 2.4.9 Page contact

Cette page est une page accessible par tout le monde même les utilisateurs anonymes. Elle consiste à envoyer un message à l'administrateur. L'utilisateur introduit ses informations et le message à envoyer.



AcadimicApp Home Enseignants Doctorants Stage Contact NASSim NASSim

### Contactez-nous

Nom

Email

Message

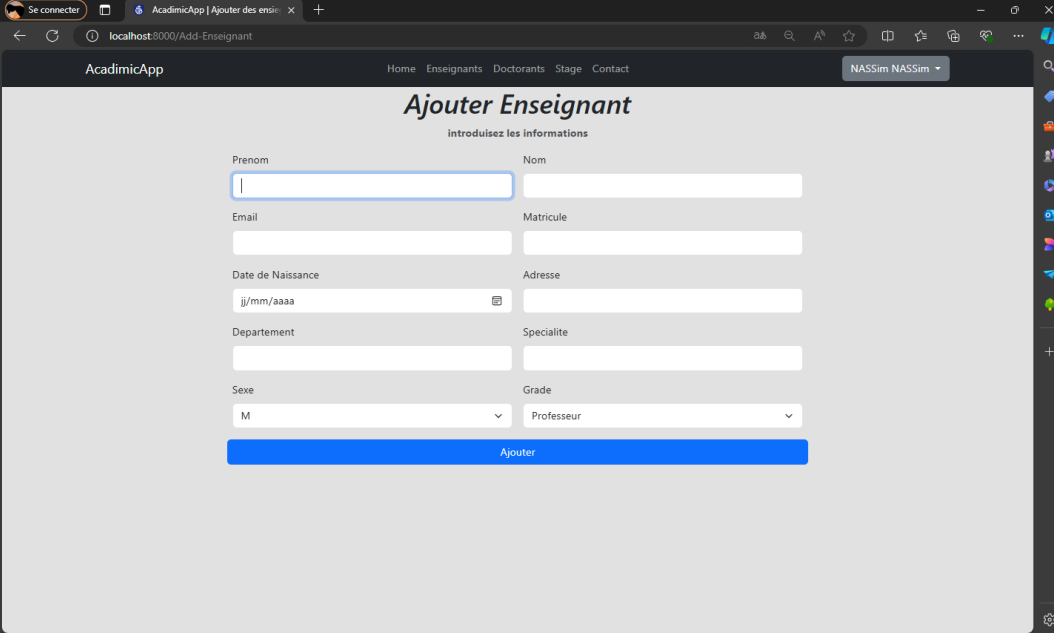
Envoyer

Figure 32:page contact

### 2.4.10 Ajouter Enseignant

La page « ajouter enseignant » permet à l'administrateur d'enregistrer un enseignant, sauf l'administrateur a accès à cette page. Il suffit remplir les champs demandés.

Un message s'affiche à la suite de validation pour signaler le succès de l'opération



The screenshot shows a web browser window displaying the 'Ajouter Enseignant' form. The form is titled 'Ajouter Enseignant' and 'introduisez les informations'. It contains the following fields:

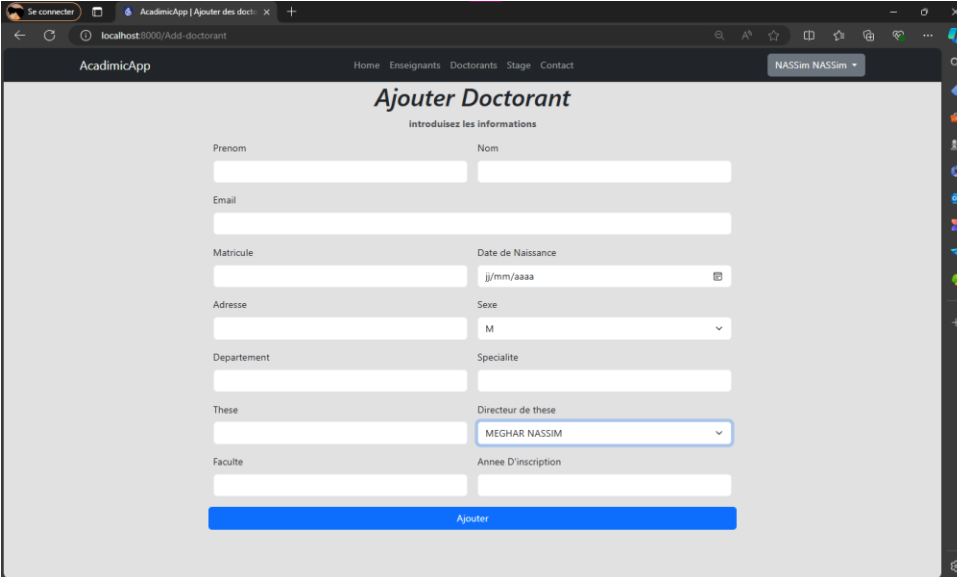
- Prenom:
- Nom:
- Email:
- Matricule:
- Date de Naissance:
- Adresse:
- Departement:
- Specialite:
- Sexe:
- Grade:

The 'Ajouter' button is highlighted in blue.

Figure 33:ajouter enseignant

#### 2.4.11 Ajouter Doctorant

La page « ajouter doctorant » permet à l'administrateur d'enregistrer un doctorant, sauf l'administrateur a accès à cette page. Il suffit de remplir les champs demandés. Un message s'affiche à la suite de validation pour signaler le succès de l'opération



The screenshot shows a web browser window displaying the 'Ajouter Doctorant' form. The form is titled 'Ajouter Doctorant' and 'introduisez les informations'. It contains the following fields:

- Prenom:
- Nom:
- Email:
- Matricule:
- Date de Naissance:
- Adresse:
- Sexe:
- Departement:
- These:
- Directeur de these:
- Faculte:
- Annee D'inscription:

The 'Ajouter' button is highlighted in blue.

Figure 34:Ajouter doctorant

### 2.4.12 Planifier un stage

La page « planifier stage » permet à l'administrateur d'créer un stage, sauf l'administrateur a accès à cette page. Il suffit remplir les champs demandés

Un message s'affiche à la suite de validation pour signaler le succès de l'opération.

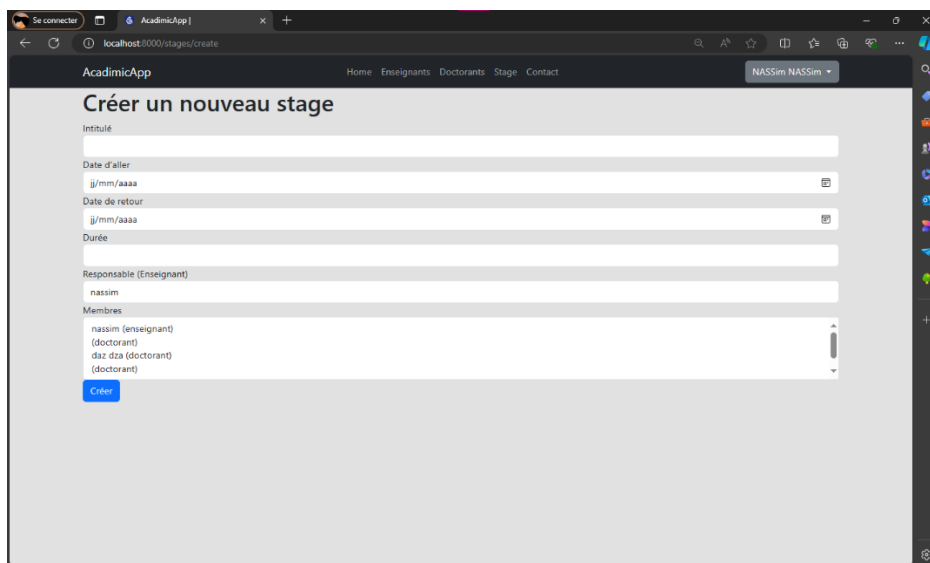


Figure 35:page planifier un stage

### 2.4.13 Supprimer enseignant

Sauf l'administrateur qui a accès à cette page. C'est la page de suppression des enseignant.

En cliquant sur l'illustration corbeille, une fenêtre s'affiche et demande s'il veut confirmer ou annuler la suppression. Ensuite un message s'affiche et annonce le succès de l'opération.

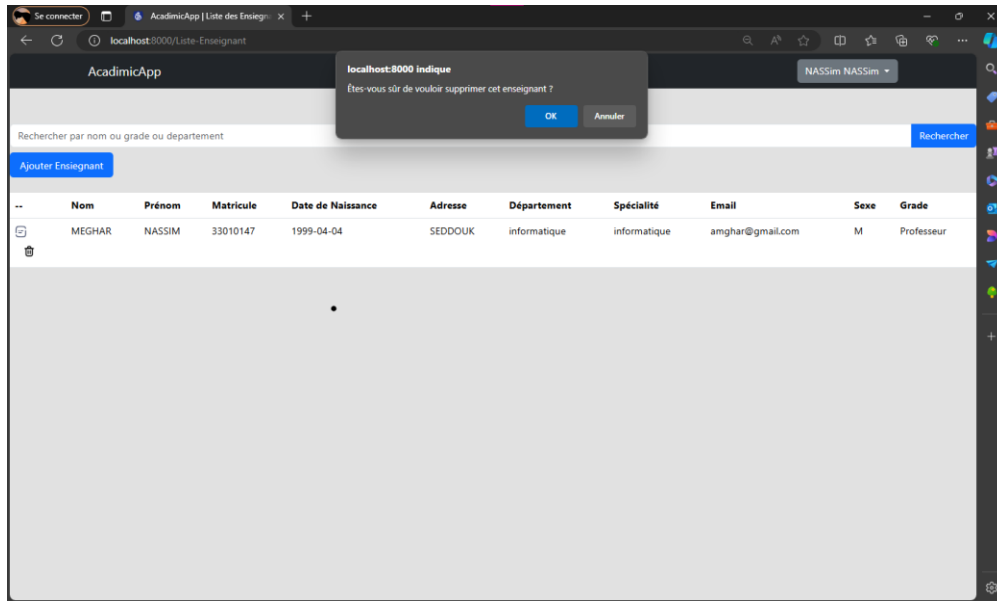


Figure 36:supprimer enseignant

### 2.4.14 Supprimer doctorant

Sauf l’administrateur qui a accès à cette page. C’est la page de suppression des doctorants.

En cliquant sur l’illustration corbeille, une fenêtre s’affiche et demande s’il veut confirmer ou annuler la suppression. Ensuite un message s’affiche et annonce le succès de l’opération.

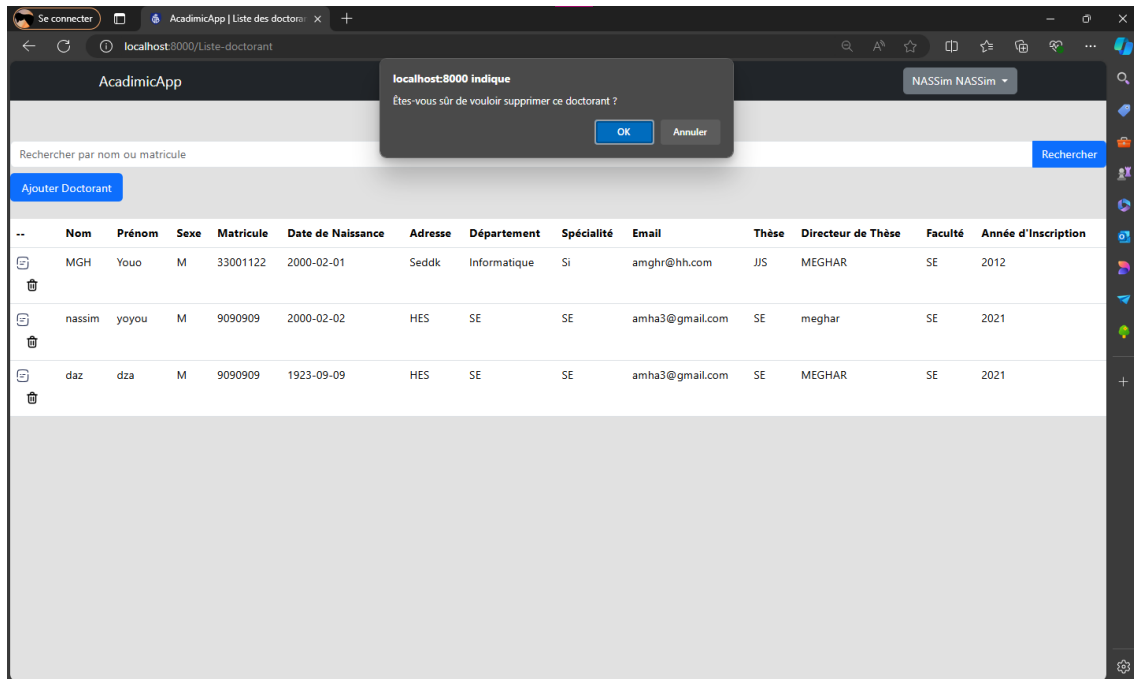


Figure 37:supprimer doctorant

### 2.4.15 Annuler stage

Sauf l'administrateur qui a accès à cette page. C'est la page d'annulation des stages.

En cliquant sur l'illustration corbeille, une fenêtre s'affiche et demande s'il veut confirmer ou annuler la suppression. Ensuite un message s'affiche et annonce le succès de l'opération.

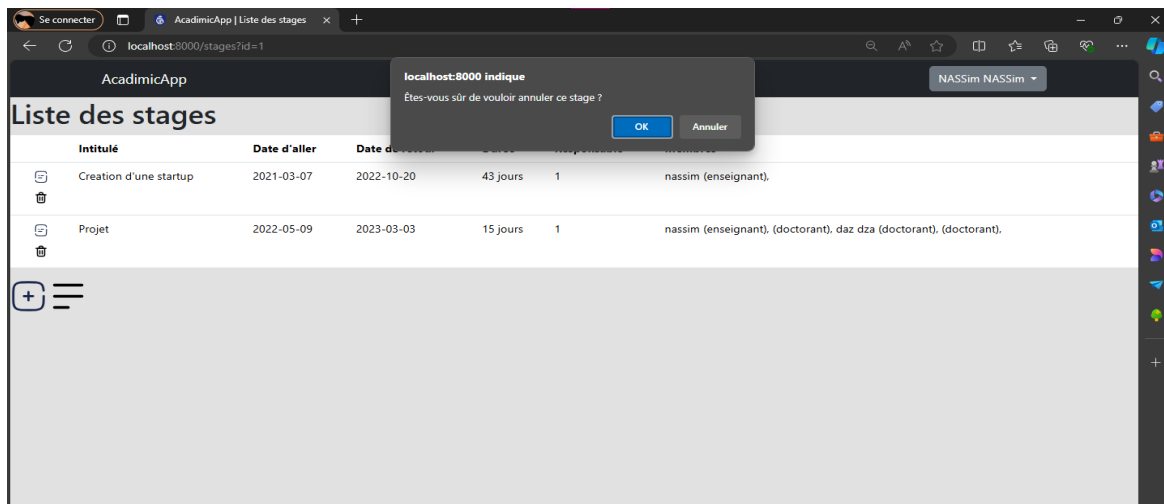


Figure 38:annuler stage

### 2.4.16 Afficher plus de détails sur le stage

En cliquant sur l'icône détails dans la page liste stage il affiche une autre page plus détaillée.

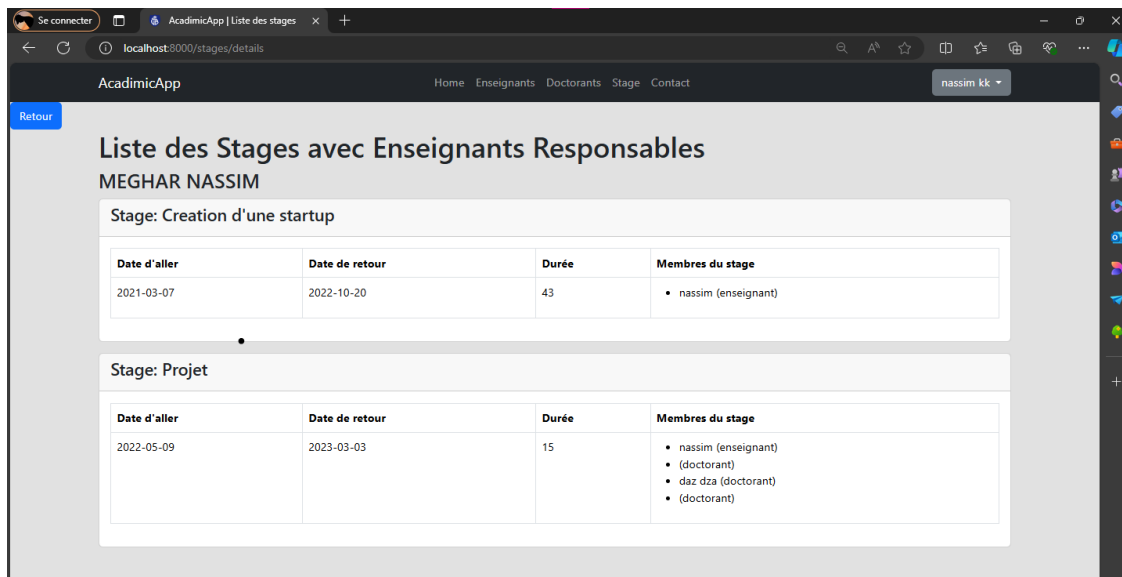


Figure 39:page plus de détails

### 2.4.17 Imprimer attestation

Sauf l'administrateur qui a accès à ces pages d'impression. La figure 45 et 46 représenté une capture d'une opération d'impression.

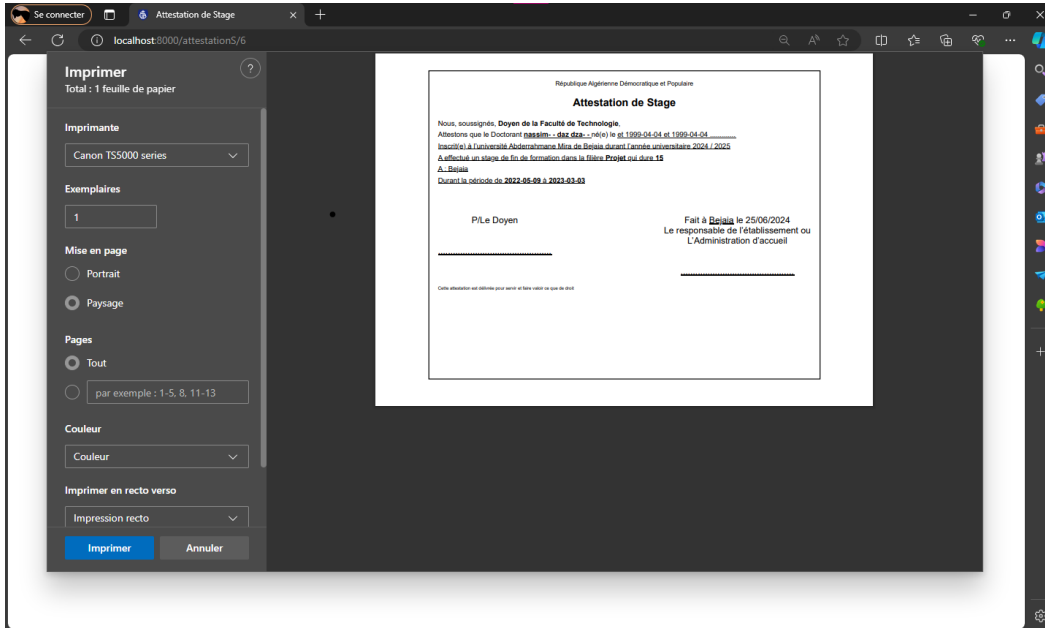


Figure 40:imprimer attestation de stage

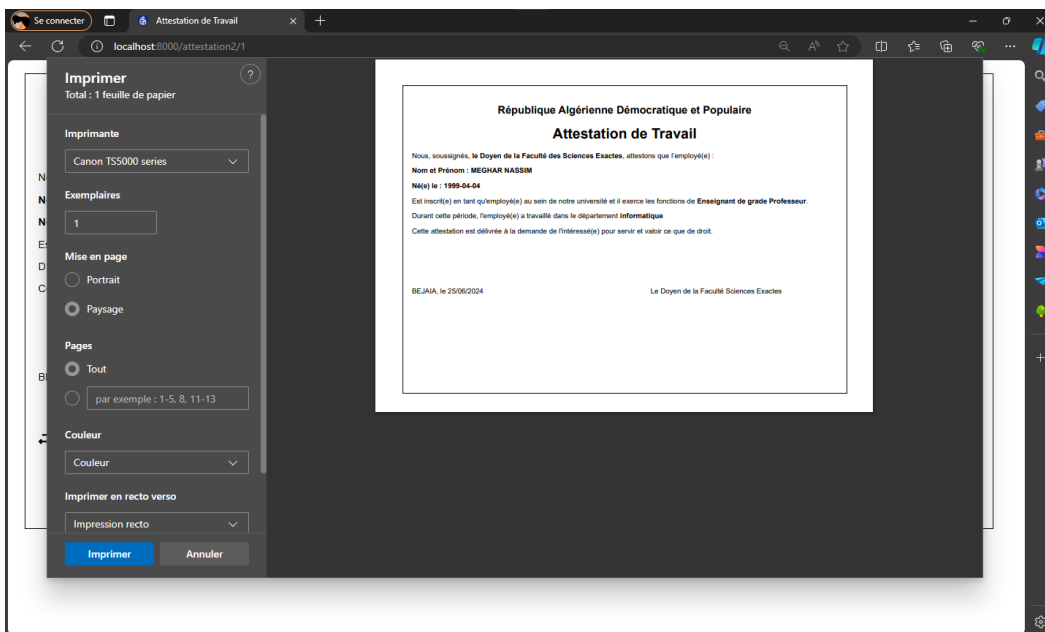


Figure 41:imprimer attestation de travail

### 2.4.18 Rechercher

Cette fonctionnalité est vraiment efficace parce que il consiste à viser directement dans la base de données. La figure 47 présente le code de la fonction.

```
public function liste_Doctorant(Request $request)
{
    $query = Doctorant::query();

    if ($request->has('search')) {
        $search = $request->input('search');
        $query->where('nom', 'LIKE', "%{$search}%")
            ->orWhere('prenom', 'LIKE', "%{$search}%")
            ->orWhere('departement', 'LIKE', "%{$search}%")
            ->orWhere('specialite', 'LIKE', "%{$search}%")
            ->orWhere('sexe', 'LIKE', "%{$search}%")
        ;
    }
}
```

Figure 42: la fonction rechercher

## 2.5 Conclusion

Ce dernier chapitre décrit tout d'abord les outils de développement que nous avons utilisés pour la réalisation de notre projet. Ensuite, nous avons présenté les interfaces les plus essentielles de notre application. En dernier lieu, nous avons défini la fonctionnalité de recherche approfondie et son processus.

Nous pensons que cette application répond aux besoins définis dans le premier chapitre et aux exigences principales des utilisateurs, qui consistent en la réalisation d'un outil permettant la planification des créneaux de soutenances d'une méthode plus optimale au sein de la faculté de sciences exactes de l'université de Béjaïa.

## Conclusion générale

Ce projet de fin cycle avait comme objectif la conception et le développement d'une plateforme de gestion des enseignant et des doctorant et planification des stage pour la faculté des sciences exactes de l'université de Bejaia.

Nous avons dressé un état de l'existant pour fonder une bonne analyse grâce auquel nous avons pu capturer tous les besoins des futurs utilisateurs de la plateforme, qui nous ont permis de fixer la structure globale de l'application à partir la phase de conception.

L'accomplissement de cette mission, nous a aidé à approfondir et élargir nos connaissances acquises durant notre cursus universitaire, à pratiquer et à utiliser les toutes nouvelles technologies de développement, et ceci en suivant des méthodes de modélisation et de conception fondamentales.

Enfin, nous espérons que ce modeste travail pourra servir de référence à tous ceux qui voudront s'engager dans un travail similaire et homologue.

## Bibliographie

1. Leila El HAJ. Fiches pratiques : Moodle pour les enseignants, 2009
2. Joseph Gabay and David Gabay. UML 2 Analyse et conception : Mise en œuvre guidée avec études de cas. Dunod, 2008.
3. Conrad Bock. Uml 2 activity and action models. J. Object Technol., 2(4) :43–53, 2003.
4. Christian Soutou. De UML à SQL - La conception de base de données. Eyrolles edition, 2002.
5. Site officiel de xampp. <https://www.apachefriends.org/fr/index.html>
6. Site officiel de composer <https://getcomposer.org/> consulté le 26/06/2024
7. Forum d'apprentissage <https://stackoverflow.com/> consulté le 21/04/2024
8. Site officiel d'Apache. <https://www.apache.org/> consulté le 24/12/2023
9. Site officiel de MySQL. <https://dev.mysql.com/doc/> consulté le 24/12/2023
10. Visual Studio Code. <https://code.visualstudio.com/docs> consulté le 24/12/2023
11. PHPMailer. <https://github.com/PHPMailer/PHPMailer#readme> consulté le 03/02/2024
12. PHP manuel. <https://www.php.net/manual/fr/index.php> consulté le 13/04/2024
13. Site officiel de Bootstrap. <https://getbootstrap.com/docs> consulté le 24/06/2024
14. Charnay Daniel and Chaléat Philippe. HTML et Javascript. Eyrolles, 1998
15. Site officiel de Laravel <https://laravel.com/docs/> consulté le 26/06/2024