

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Master en Informatique

Option : Système d'Information et de Connaissances (S.I.C)

Thème

Conception et développement d'une application web pour la gestion des dossiers des doctorants

Réalisé par :

- BELASKRI MONCEF
- BAGHLI MOHAMMED EL-HADI

Présenté le 29 Juin 2022 devant le jury composé de MM.

- BENAMAR ABDELKRIM (Président)
- EL YEBDRI ZEYNEB (Encadreur)
- BENMANSOUR FAZILET (Examineur)

Remerciements

D'abord et avant tout, merci à Allah le tout miséricordieux, de nous avoir donné la force, la patience et le courage nécessaire pour finaliser toutes ces longues années d'études.

À toutes les personnes qui nous ont aidé durant la réalisation de ce travail, veuillez trouver dans ces lignes l'expression de notre reconnaissance et notre profonde gratitude.

Nous tenons à remercier aussi notre encadrante Mme Zeyneb ELYEBDRI qui nous a aidé à mener à bien notre travail grâce à tous ses conseils et son précieux soutien.

L'expression de nos remerciements les plus sincères vont aux membres de notre jury Mr BENAMAR Abdelkrim et Mme BENMANSOUR Fazilet pour avoir accepté de lire notre mémoire et d'évaluer notre travail.

Merci à tous les enseignants du département informatique qui ont fournis des efforts pour nous donner leurs savoir durant tout notre cursus universitaire.

Enfin, merci à nos parents et à tous les membres de nos familles respectives de nous avoir apporté leur soutien tout au long de cette démarche.

Dédicaces

En premier lieu, je remercie Dieu, le tout puissant de m'avoir donné la force pour réaliser ce modeste travail, que je dédie :

À mes chers parents qui n'ont jamais arrêté de m'encourager et de me soutenir durant toutes les épreuves que j'ai passé dans ma vie ainsi que pour tous les sacrifices qu'ils ont consentis pour ma réussite, que Dieu leur accorde santé, bonheur et prospérité.

À mes très chers soeurs pour leur aides et leur encouragements.

À tous les membres de ma famille qui se reconnaîtront dans ce message.

À tous mes amis ainsi qu'à tous les gens que j'ai côtoyé durant mon cursus scolaire.

BAGHLI Mohammed El-Hadi

Dédicaces

Tout d'abord, merci à Dieu de m'avoir aidé dans la réalisation de ce modeste travail, que je vais dédier :

À mes très chers parents qui m'ont soutenu et qui ont su m'aider à atteindre mes objectifs et pour tous les sacrifices qu'ils ont fait pour moi. Que ce travail soit le témoignage de ma plus profonde reconnaissance envers eux.

À mon frère et à ma soeur pour leurs encouragements.

À toute ma famille ainsi que mes amis pour les moments qu'on a passé ensemble.

À tous les gens que j'ai connu dans ma vie et qui m'ont aidé d'une quelconque façon.

BELASKRI Moncef

Abréviations

UABT	<i>Université Abou Bekr Belkaid Tlemcen</i>
DEUA	<i>Diplôme Universitaire d'Etudes Appliquées</i>
FS	<i>Faculté des Sciences</i>
UML	<i>Unified Modeling Language</i>
OOSE	<i>Object Oriented Software Engineering</i>
OMT	<i>Object Modeling Technique</i>
OMg	<i>Object Management Group</i>
UP	<i>Unified Process</i>
DCU	<i>Diagramme de Cas d'Utilisation</i>
UC	<i>Use case/ Cas d'utilisation</i>
DSS	<i>Diagramme de Séquence Système</i>
VSC	<i>Visual Studio Code</i>
HTML	<i>Hypertext Markup Language</i>
CSS	<i>Cascading Style Sheets</i>
JS	<i>JavaScript</i>
API	<i>Application Programming Interface</i>
MERN	<i>MONGO EXPRESS REACT NODE</i>
MEVN	<i>MONGO EXPRESS VUE NODE</i>
MEAN	<i>MONGO EXPRESS ANGULAR NODE</i>
Bdd	<i>Base de Données</i>
SQL	<i>Structured Query Language</i>
No-SQL	<i>Not only SQL</i>
JWT	<i>JavaScript Object Notation Web Token</i>
CRUD	<i>Create Read Update Delete</i>

Table des figures

1.1 Répartition des différents pôle de l'UABT	8
1.2 Organigramme UABT	10
1.3 Organigramme FS	11
2.1 Diagramme de Cas d'Utilisation du Visiteur	20
2.2 Diagramme de Cas d'Utilisation de la Secrétaire	21
2.3 Diagramme de Cas d'Utilisation du Doctorant	22
2.4 Diagramme de Cas d'Utilisation du Directeur/Co-directeur de thèse	22
2.5 Diagramme de Cas d'Utilisation de l'Adjoint de post-graduation	23
2.6 Diagrammes de Séquence Système pour contacter le support	24
2.7 Diagrammes de Séquence Système pour se connecter	24
2.8 Diagrammes de Séquence Système de la Secrétaire	25
2.9 Diagrammes de Séquence Système du Doctorant	25
2.10 Diagrammes de Séquence Système du Directeur/Co-directeur de thèse	26
2.11 Diagrammes de Séquence Système pour imprimer des documents	26
3.1 Logo des logiciels utilisés	29
3.2 la MERN Stack	31
3.3 la MERN Stack	31
3.4 Fonctionnement du JWT	33
3.5 Schéma Secrétaire et Adjoint de post-graduation	34
3.6 Schéma Enseignant	35
3.7 Schéma Doctorant	36
3.8 Schéma Avancement du Doctorant	37
3.9 Interface d'accueil	41
3.10 Interface de Connexion	42
3.11 Interface de la liste des doctorants	42
3.12 Interface pour ajouter un doctorant	43
3.13 Interface pour voir un doctorant	43
3.14 Interface pour modifier un doctorant	44
3.15 Interface pour supprimer un doctorant	44
3.16 Interface de la liste des enseignants	44
3.17 Interface pour ajouter un enseignant	45
3.18 Interface pour voir un enseignant	45

3.19	Interface pour modifier un enseignant	46
3.20	Interface pour supprimer un enseignant	46
3.21	Interface pour voir son profil	47
3.22	Interface pour modifier son avancement	47
3.23	Interface pour voir son avancement	48
3.24	Interface pour voir ses doctorants	48
3.25	Interface pour voir ou modifier l'avancement d'un de ses doctorants	49
3.26	Interface pour voir la liste des Doctorants activés	49
3.27	L'attestation d'inscription	50
3.28	La 1ère page de la fiche de réinscription	51
3.29	La 2ème page de la fiche de réinscription	52
3.30	Interface pour voir la liste des doctorants	53
3.31	Interface pour voir un doctorant	53
3.32	Interface pour modifier l'étape en cours d'un doctorant	54
3.33	Interface pour voir la liste des enseignants	54
3.34	Interface pour voir un enseignant	54
3.35	Interface pour voir les statistiques du département	55

Liste des tableaux

- 1.1 Comparaison avec les systèmes existants 14
- 1.2 Comparaison entre application bureau et web 14

Table des matières

Introduction Générale	4
1 Etude de l'Existant	6
1.1 Introduction	7
1.2 Présentation de l'Université Abou Bekr Belkaid Tlemcen(UABT)	7
1.2.1 Historique	7
1.2.2 Organes de l'UABT	8
1.2.3 Faculté des Sciences(FS)	11
1.2.4 Organigramme de la FS	11
1.3 Étude de l'existant	12
1.3.1 Objectif	12
1.3.2 Fonctionnement actuel	12
a) Types de doctorat	12
b) Étapes du doctorat	12
c) L'Inscription Universitaire	12
1.3.3 Méthodes utilisées pour la gestion des dossiers	13
a) Gestion manuelle	13
b) Utilisation de Google Forms	13
1.3.4 Systèmes existants	14
1.4 Besoins	15
1.4.1 Fonctionnels	15
a) Accès Visiteur	15
b) Accès secrétaire	15
c) Accès doctorants	15
d) Accès enseignants	15
e) Accès adjoint de post graduation	16
1.4.2 Non Fonctionnels	16
1.5 Conclusion	16
2 Conception détaillée de notre système DocUniT	17
2.1 Introduction	18
2.2 Le langage de modélisation UML	18
2.3 La méthode UP	19
2.4 Conception	19

2.4.1	Diagrammes de Cas d'Utilisation	20
a)	Pour le Visiteur	20
b)	Pour la Secrétaire	21
c)	Pour le Doctorant	22
d)	Pour le Directeur/Co-directeur de thèse	22
e)	Pour l'Adjoint de post-graduation	23
2.4.2	Diagrammes de Séquence Système	24
a)	Pour le Visiteur	24
b)	Pour la Secrétaire	25
c)	Pour le Doctorant	25
d)	Pour le Directeur/Co-directeur de thèse	26
e)	Pour l'Adjoint de post-graduation	26
2.4.3	Conclusion	27
3	Mise en œuvre de DocUniT	28
3.1	Introduction	29
3.2	Présentation des logiciels et outils de travail	29
3.2.1	Latex avec Overleaf	29
3.2.2	Adobe Photoshop	29
3.2.3	GitHub	30
3.2.4	Postman	30
3.2.5	Visual Studio Code (VSC)	30
3.3	Présentation des technologies pour la programmation	31
3.3.1	React pour le Front-end	32
3.3.2	Node et son framework Express pour le back-end	32
3.3.3	MongoDB pour la gestion de la base de données	32
3.3.4	Avantages et Inconvénients de la MERN Stack	32
3.4	Présentation des outils de sécurisation de données	33
3.4.1	Le JavaScript Object Notation Web Token	33
3.4.2	Fonction de Hachage	34
3.5	Réalisation	34
3.5.1	Création des schémas des collections MongoDB	34
3.5.2	Manipulation des données à travers les collections	37
a)	La création des données	37
b)	La mise à jour des données	38
c)	La suppression des données	39
3.5.3	Présentation des interfaces de notre application web DocUnit	40
a)	Interface du visiteur	40
b)	Les interfaces de la Secrétaire	42
c)	L'interface du Doctorant	47
d)	L'interface de l'Enseignant(Directeur/Co-directeur de thèse)	48

e) Les interfaces de l'Adjoint de Post-Graduation	49
3.6 Conclusion	55
Conclusion Générale	56
Bibliographie	57

Introduction Générale

De nos jours, le domaine de l'informatique a tellement pris de l'ampleur dans nos vies quotidiennes qu'on se doit de mettre à jour les systèmes utilisés dans nos services publics car ils sont devenus obsolètes.

Avant l'arrivée de l'ère informatique, les organismes publiques et plus particulièrement les administrations se basaient sur le recueil des informations de leurs clients grâce à des supports en papier. Cette méthode engendre une perte de temps considérable dans la saisie et la recherche d'informations ainsi que la dégradation des supports au fil du temps.

Dans le cadre de notre projet de fin d'étude, nous proposons une solution informatique pour la gestion des dossiers des doctorants de l'université de Tlemcen et plus particulièrement pour la faculté des sciences. Notre système baptisé DocUniT survient d'un besoin réel suite aux différents problèmes que rencontre les services concernés à savoir : les services de post-graduation des départements ainsi que le service de post-graduation de la faculté pour manipuler les dossiers des doctorants. Pour cela, notre système constitue une aide précieuse et utile pour les parties prenantes.

Le travail que nous avons réalisé a été réparti en plusieurs étapes :

En effet, dans le premier chapitre, nous avons commencé par une introduction sur l'université de Tlemcen et son histoire, ensuite nous avons présenté ses différentes facultés et plus particulièrement la faculté des sciences qui sera l'environnement de notre étude, puis nous avons fait une analyse de l'existant qui nous a permis de réaliser notre cahier de charges et de recenser tous les besoins de notre travail.

Ensuite, dans le deuxième chapitre, nous avons présenté la conception de notre système que nous avons réalisé grâce au langage de modélisation UML et la méthode UP. Dans cette dernière nous avons travaillé avec le diagramme de cas d'utilisation(DCU) pour citer tous les besoins de chaque utilisateur et le diagramme de séquence système(DSS) pour détailler chaque besoin présent dans le DCU.

Après, nous sommes passés à la réalisation de l'application web dans le 3ème chapitre en utilisant l'approche MERN qui est composée de quatre technologies à savoir : MongoDB, Express, React, Node. Cette approche nous a permis de mettre en place les interfaces de notre front-end avec React, de faire fonctionner correctement notre back-end grâce à Node et son framework Express qui interagit directement avec le React et de récupérer les données de notre base de données NoSQL à l'aide de Mongoose qui est en relation direct avec Express.

Enfin, nous clôturons ce mémoire par une conclusion générale et la présentation des perspectives futures pour l'amélioration de notre application web DocUnit.

Chapitre 1

Etude de l'Existant

1.1 Introduction

L'analyse de l'existant est la première étape du cycle de vie de notre système. Elle permet de prendre connaissance sur l'état actuel de l'environnement étudié en nous fournissant toutes les informations nécessaires sur le sujet afin de faire une bonne conception et de trouver la solution adéquate à notre problème. Dans ce premier chapitre, nous allons présenter le contexte de notre travail à savoir les différents services de l'université de Tlemcen et plus particulièrement, ceux qui s'impliquent dans la gestion des dossiers des doctorants dans la faculté des Sciences qui est notre centre d'intérêt.

Ensuite, nous établirons un cahier de charges qui aura pour but de présenter le contexte actuel et la finalité de notre étude et qui contiendra tous les besoins des différents acteurs qui prendront part au système étudié.

1.2 Présentation de l'Université Abou Bekr Belkaid Tlemcen(UABT)

1.2.1 Historique

L'université Abou Bekr Belkaid Tlemcen est une université située à Tlemcen en Algérie. Elle fut créée par le décret n° 89-138 du 01/08/1989, modifié et complété par le décret exécutif n° 95-205 du 05/08/1995, puis modifié par le décret exécutif n° 98-391 du 02/12/1998. Elle est issue d'une longue évolution progressive.

Au début, l'enseignement supérieur se concentrait sur un pôle universitaire qui comportait à l'origine (1974-1980) le seul tronc commun de Sciences Exactes et de Biologie.

Au fil du temps, cet enseignement s'est étendu de manière progressive à de nouvelles filières, couvrant chaque année, une série de stages et donnant aux étudiants la possibilité de poursuivre l'intégralité de leur cursus de fin d'études à Tlemcen.

En août 1984, l'établissement de la nouvelle carte universitaire et donc la création des instituts nationaux d'enseignement supérieur va permettre d'une part, à certains départements déjà existants d'acquérir le statut de véritables établissements et d'autre part, l'émergence de nouveaux secteurs.

Cette étape, engendre la création d'une formation de niveau 5 (Diplôme Universitaire d'Etudes Appliquées : DEUA) et le développement du premier post-diplôme dans la quasi-totalité des filières assurées à Tlemcen. Enfin, le lancement de la seconde post graduation verra le jour à partir de 1991-1992.

Finalement, après une période de quinze ans de gestation qu'apparaît l'Université d'Abou Bekr Belkaïd. Elle va être présentée comme étant une nouvelle entité, riche de cette longue période de maturation qui aura pour objectifs de réaliser de nouveaux défis.

L'université contient un rectorat, 8 facultés et un institut qui sont répartis sur 5 pôles universitaires ainsi qu'une annexe à Maghnia.

- Le pôle Imama qui contient la faculté des sciences économiques et de gestion et la faculté de droit et de sciences politiques.
- Le pôle Chetouane qui contient la faculté de technologie.
- Le pôle Kiffane qui contient l'institut des sciences et techniques appliquées(ISTA).
- Le pôle rocade appelé aussi le nouveau pôle qui contient la faculté des sciences de la nature, de la vie, de la terre et de l'univers, la faculté des sciences humaines et sociales, la faculté des sciences

et la faculté des langues et littérature.

- Le pôle caserne Miloud qui contient la faculté des sciences médicales. [1]

La figure suivante représente un récapitulatif des différents pôles de l'UABT.

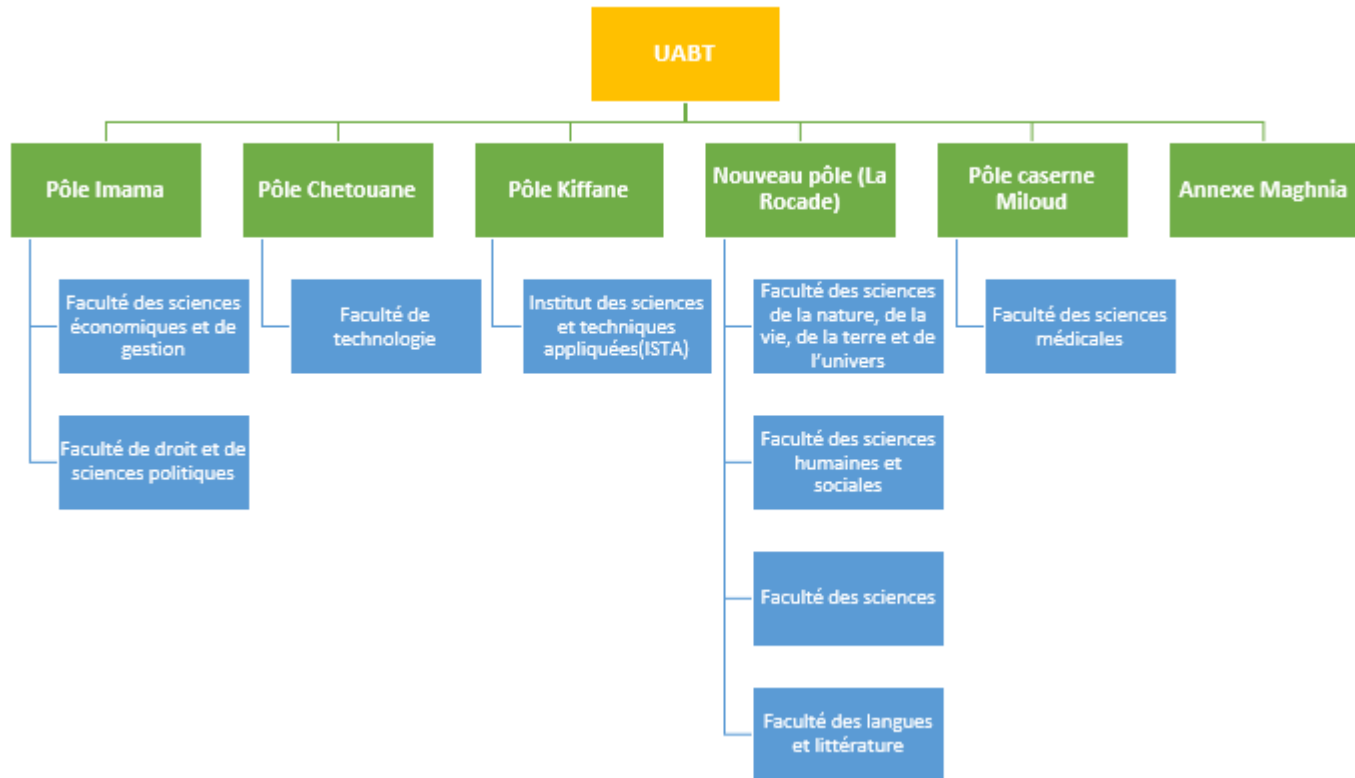


FIGURE 1.1 – Répartition des différents pôle de l'UABT

Nous nous focalisons dans ce qui suit sur les différents services qui sont assurés par l'UABT.

1.2.2 Organes de l'UABT

Le rectorat est composé d'un recteur qui est assisté dans ses tâches par quatre (4) vice-recteurs et un secrétaire général, il est organisé comme suit :

- Le Vice-recteur de la Formation Supérieure du Premier et Deuxième Cycles, de la formation Continue et les Diplômes et de la Formation Supérieure de Graduation. Il est responsable de :
 - Assurer le suivi des questions liées à l'enseignement et à la formation organisés par l'université.
 - Veiller à la cohérence des offres de formation présentées par les facultés et instituts avec le plan de développement universitaire.
 - Veiller au respect de la réglementation applicable en matière d'inscription, de réinscription, de contrôle des connaissances et de progression des étudiants.
 - Assurer le suivi des activités d'enseignement à distance proposées par l'université et promouvoir les activités de formation continue.
 - Veiller au respect des règles et procédures en vigueur pour la délivrance des diplômes et des équivalences.
 - Maintenir et mettre à jour le fichier des noms des étudiants.

Le Vice-Rectorat est composé du :

- Service de l'enseignement, de la formation et de l'évaluation.

- Service de la formation continue.
- Service des diplômes et des équivalences.
- Le Vice-recteur de la Formation Supérieure du Troisième Cycle, de l'Habilitation Universitaire et la Recherche Scientifique et de la Formation Supérieure de Post-Graduation. Il est responsable de :
 - Assurer le suivi des questions liées au déroulement des formations post-diplôme, post-diplôme spécialisé et d'accréditation académique et veiller au respect de la réglementation en la matière.
 - Veiller à la cohérence des offres de formation présentées par les facultés et instituts avec le plan de développement universitaire.
 - Suivre et rendre compte des activités de recherche des unités et laboratoires de recherche, en coordination avec les facultés et instituts.
 - Réaliser toute action de valorisation des résultats de la recherche.
 - Surveiller le fonctionnement du conseil scientifique de l'université et préserver ses archives
 - collecter et diffuser des informations sur les activités de recherche menées par l'université.
- Le Vice-Rectorat est composé du :
 - Service de formation post-diplôme et de post-diplôme spécialisé.
 - Service d'Accréditation Académique Universitaire.
 - Service de suivi des activités de recherche et de valorisation de ses résultats.
- Le Vice-recteur des Relations Extérieures, de la Coopération, de l'Animation et de la Communication et des Manifestations Scientifiques. Il est responsable de :
 - Promouvoir les relations de l'université avec son environnement socio-économique et initier des programmes de partenariat.
 - Initier toute action visant à favoriser les échanges et la coopération interuniversitaires dans les domaines de l'enseignement et de la recherche.
 - Mener des activités d'animation et de communication.
 - Organiser et promouvoir des événements scientifiques.
 - Surveiller et assurer la cohérence des programmes de perfectionnement des enseignants et de la formation.
- Le Vice-Rectorat est composé du :
 - Service des échanges, coopérations et partenariats inter-universitaires .
 - Service animation et communication et manifestations scientifiques.
- Le Vice-recteur au développement, à la Prospective et à l'Orientation. Il est responsable de :
 - Rassembler les éléments nécessaires à l'élaboration des projets de plans de développement de l'université.
 - Effectuer toute étude prospective sur les prévisions d'évolution de la communauté étudiante de l'université et proposer toutes mesures pour leur gestion, notamment, en termes de développement, d'encadrement pédagogique et administratif.
 - Maintenir le fichier statistique universitaire en assurant sa mise à jour périodique.
 - Élaborer tout matériel d'information sur les filières d'études dispensées par l'université et leurs débouchés professionnels.
 - Mettre à la disposition des étudiants toute information pouvant les aider dans leur choix d'orientation.

— Promouvoir les activités d'information. [2]

Pour résumer, la figure suivante présente toutes les parties du rectorat d'une manière générale.

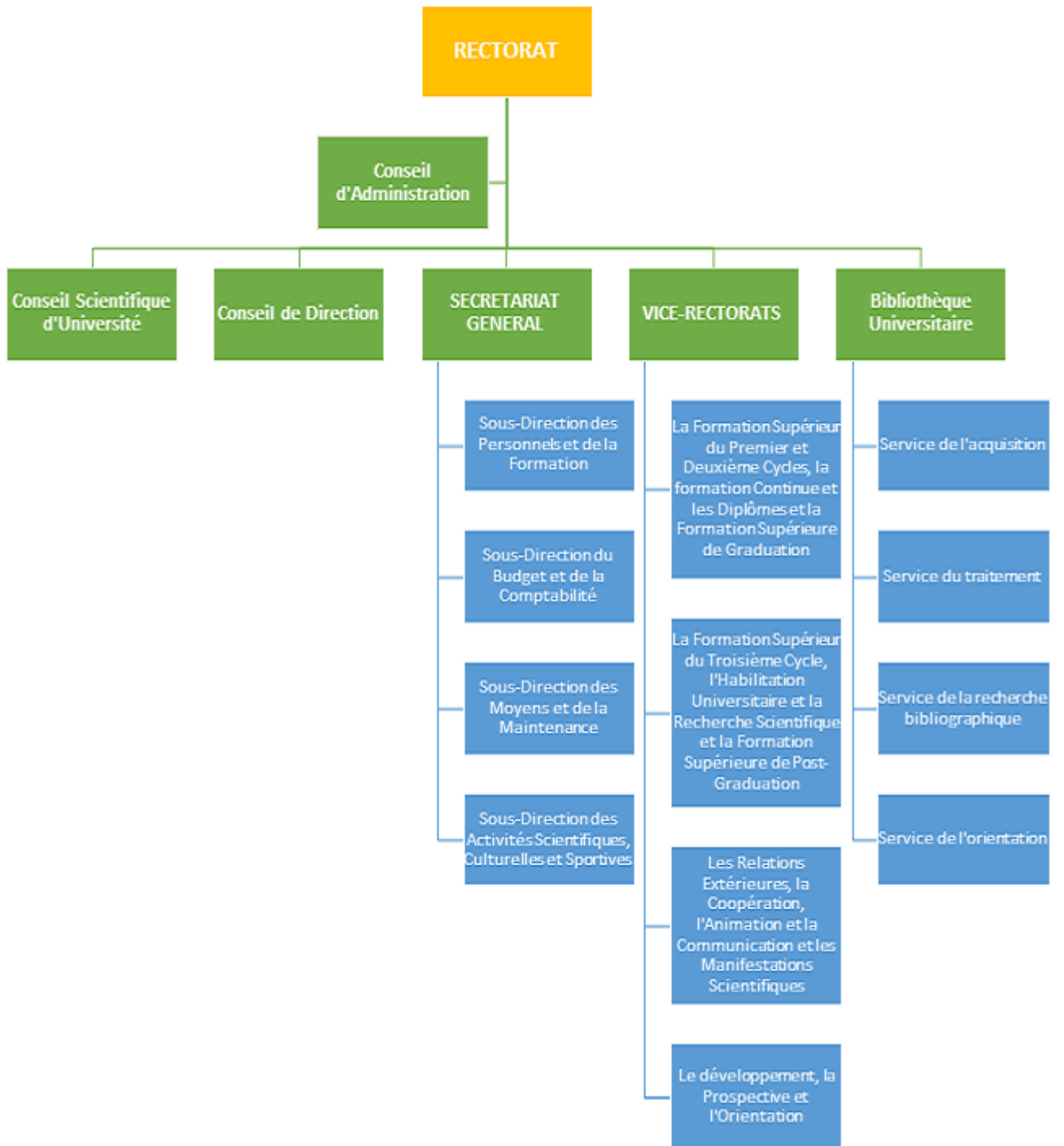


FIGURE 1.2 – Organigramme UABT

1.2.3 Faculté des Sciences(FS)

Parmi les facultés de l'UABT, on s'intéresse dans notre travail aux dossiers des doctorants de la faculté des Sciences. Cette dernière comporte quatre départements :

- Département d'Informatique.
- Département de Mathématique.
- Département de Physique.
- Département de Chimie.

1.2.4 Organigramme de la FS

L'organigramme qui suit (figure 1.3) présente de manière globale la hiérarchie de toutes les structures présentes au niveau de la faculté.

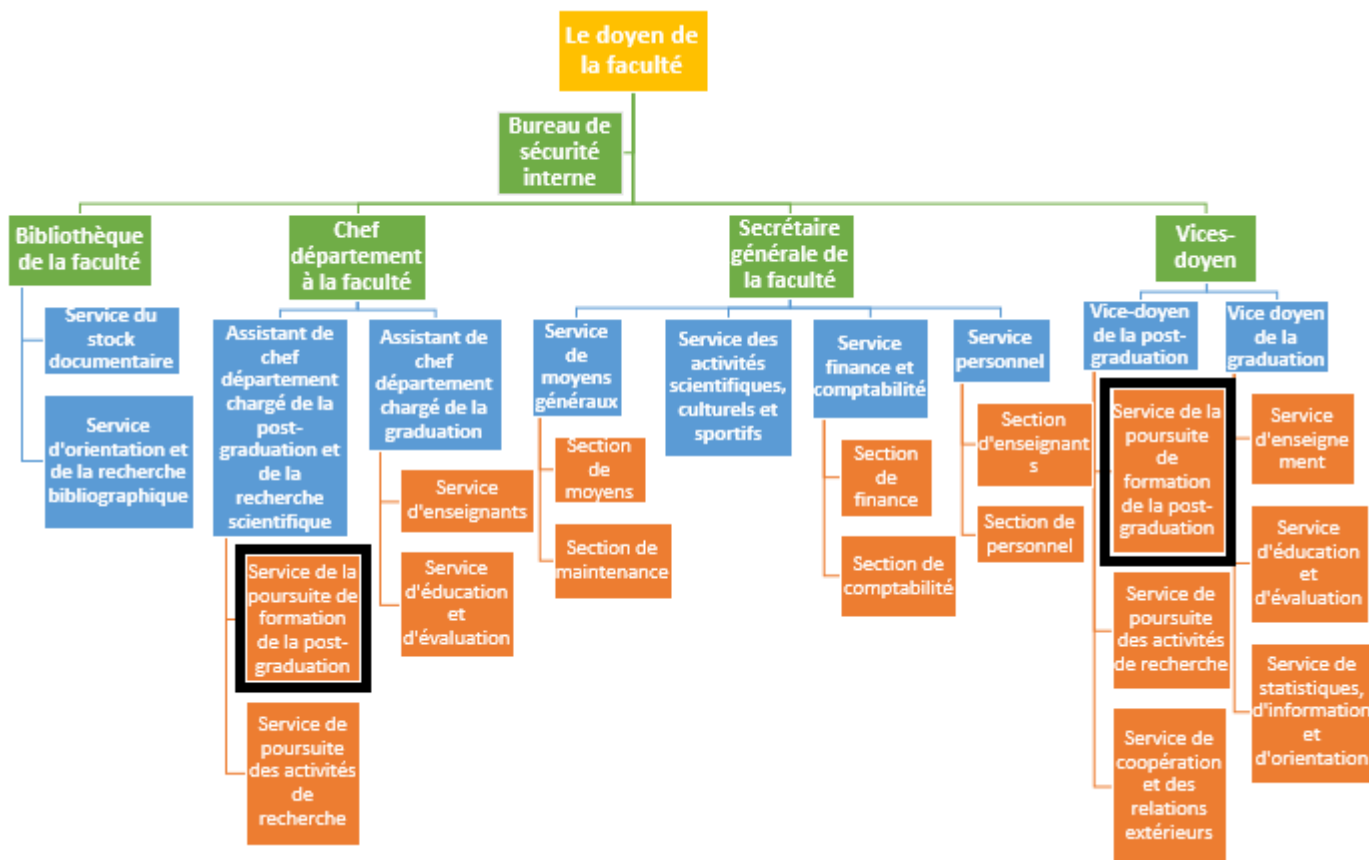


FIGURE 1.3 – Organigramme FS

Notre travail fait l'objet d'assurer une aide pour les deux services encadrés en noir : le service de la poursuite de formation de la post-graduation coté département et coté faculté. La section suivante est consacrée pour décrire la situation actuelle de la gestion des dossiers des doctorants au niveau du département d'informatique. [3]

1.3 Étude de l'existant

1.3.1 Objectif

Pour la gestion de dossiers des doctorants, notre université se base sur la récolte de dossiers avec des documents matérialisés ce qui implique une gestion approximative des dossiers, l'absence de sécurité et une présence physique obligatoire pour les doctorants.

Notre objectif va être d'informatiser ce système, qui de nos jours est devenu obsolète afin de faciliter l'inscription au doctorat ainsi que la gestion des données, et de sécuriser le système pour le protéger contre toutes menaces potentielles.

1.3.2 Fonctionnement actuel

a) Types de doctorat

Dans notre université, on a deux types de doctorat :

- Le doctorat classique : L'accès à la formation doctorale de l'ancien système se fait après l'obtention du diplôme de magister ou d'un titre équivalent.
- Le Doctorat du LMD : L'accès à la formation doctorale du système LMD se fait par voie de concours en finissant en haut du classement pour pouvoir être pris selon le nombre de places disponible pour chaque spécialité de chaque domaine.[4]

b) Étapes du doctorat

La période doctorale passe par 3 étapes principales :

- L'inscription pour la génération des fiches d'inscription et de réinscription.
- La soutenance pour l'évaluation du travail du doctorant.
- L'obtention du diplôme.[4]

c) L'Inscription Universitaire

- Après validation des résultats du concours par les comités et conseils scientifiques, les candidats admis au concours d'accès à une formation doctorale, doivent procéder à leur inscription au sein d'un seul établissement universitaire, dans un délai n'excédant pas les quinze (15) jours qui suivent la proclamation définitive des résultats.
- Le candidat doit déposer son dossier d'inscription auprès de son département qui le transmettra au service des inscriptions du Vice-Rectorat de la Post-Graduation. Il lui sera délivré une attestation d'inscription.
- Durée de préparation du Doctorat :
 - Pour l'ancien Doctorat : 05 Années maximum + 01 sur dérogation.
 - Pour le nouveau Doctorat : 03 Années maximum + 02 sur dérogation.L'année supplémentaire est accordée par le Chef d'établissement sur rapport motivé du directeur de thèse et après avis du conseil scientifique de l'établissement d'inscription.
- Dossier Administratif d'Inscription en Doctorat :
 - Inscription en 1ère année de Doctorat :

- Imprimé de demande d'inscription visé par le Président du Conseil Scientifique de la Faculté et par le Doyen.
- Procès verbal du Conseil Scientifique de la Faculté.
- Copie légalisée du baccalauréat.
- Copie légalisée du diplôme de graduation.
- Copie légalisée du diplôme de Magister (Pour L'ancien Système).
- Attestation de direction des études visée par le Président du Conseil Scientifique de la Faculté et par l'encadreur.
- Imprimé du CERIST.
- Extrait d'acte de naissance.
- 02 Photos d'identité.
- Frais d'inscription (200 DA).

- Réinscription en Doctorat :
 - Imprimés de demande de réinscription visé par le Président du Conseil Scientifique de la Faculté et le Doyen après l'avis de l'encadreur.
 - Frais d'inscription (200 DA).
 - État d'avancement de la thèse rédigé et visé par le directeur de thèse. [4]

1.3.3 Méthodes utilisées pour la gestion des dossiers

Nous avons interrogé le personnel universitaire sur les différentes méthodes mises en place pour la gestion des dossiers et après observation, nous avons retenu les méthodes suivantes :

a) Gestion manuelle

L'administration récupère les dossiers des doctorants et les met dans des classeurs selon la branche concernée et l'année d'inscription courante. Les problèmes de cette méthode sont :

- Absence de sécurité pour les informations personnelles des doctorants.
- Volume trop important de données à traiter ce qui peut engendrer la présence d'erreurs dans les dossiers des doctorants.
- Problème de stockage et détérioration de l'état des dossiers.
- Absence d'une application informatique.
- Aucune sauvegarde numérique des dossiers des doctorants.

b) Utilisation de Google Forms

Méthode informatisée qui permet aux doctorants de remplir plus facilement leurs données de réinscription selon un formulaire en ligne¹. Le problème de cette méthode est que le doctorant doit quand même remplir à nouveau sa fiche de réinscription sous format word et la déposer au niveau du département. Autre problème avec ce formulaire, c'est que le doctorant doit le remplir chaque année, ce qui veut dire que les informations statiques tel que le nom et le prénom doivent être ressaisies à chaque fois, ce qui

1. Formulaire d'inscription en ligne réalisé avec Google Forms <https://docs.google.com/forms/d/e/1FAIpQLSdoc-sXZ5cDKS0hqJaUo4YLDgovfoo3yAbJLiD76vnGeC1HPw/viewform>

crée un travail supplémentaire pour rien.

L'enregistrement des données par le doctorant via ce formulaire engendre une sauvegarde d'une ligne supplémentaire dans un fichiers excel dont seul le service de post graduation de la faculté aura accès. Ce dernier ne va servir que pour consultation, il l'utilise pour faire des statistiques ou tirer des informations. Et comme déjà mentionné, le doctorant doit déposer en main propre sa fiche de réinscription saisie sous word au niveau du département.

1.3.4 Systèmes existants

Dans le même contexte que le notre, c'est à dire la réalisation d'un système de gestion pour les dossiers des doctorants, nous avons trouvé deux travaux similaires au niveau de notre département qui ont été fait dans le cadre de projet de fin d'étude.

- Le premier a été réalisé par Bensidhoum Ayoub et Cherrak Soufiane.[5]
- Le deuxième quant à lui, est fait par Sakhi Walid Lahcen.[6]

Les deux sont des applications bureau faites en Pascal pour la première et en Java pour la deuxième, elles utilisent toutes les deux une bdd Sql afin de créer les dossiers des doctorants et générer les documents de ces derniers.

Le tableau suivant présente une comparaison générale entre les systèmes existants et notre système.

	Type D'application	Sécurité	Impression Attestation d'inscription	Impression Fiche de réinscription	Accès par rôle
Bensidhoum et Cherrak	Bureau	x	✓	✓	x
Sakhi	Bureau	x	✓	x	x
DocUniT	Web	✓	✓	✓	✓

TABLE 1.1 – Comparaison avec les systèmes existants

[7]L'aspect principal de différence est le type d'application(bureau ou web), ci-dessous un tableau comparatif entre ces deux types d'application :

	Besoin de connexion	Installation et mise à jour	Type de Platform	Moyen de déploiement	Performance	Sécurité
Application Bureau	Pas besoin	Pas d'installation avec mise à jour automatique	Une version pour chaque Platform	Nouvelle copie sous licence requise pour chaque système	Rapide et plus de fonctionnalités	Sécurisé
Application Web	Obligatoire	Installation et mise à jour manuelle	Cross Platform	Besoin seulement d'un navigateur et d'un compte	Peut-être lente selon sa complexité	Moins sûre

TABLE 1.2 – Comparaison entre application bureau et web

1.4 Besoins

Notre système devra prendre en compte les différents besoins fonctionnels et non fonctionnels de tous les acteurs qui vont interagir avec lui, à savoir :

- Le visiteur.
- La secrétaire.
- Le doctorant.
- L'enseignant(directeur/co-directeur de thèse).
- L'adjoint de post graduation.

1.4.1 Fonctionnels

Dans notre application web, il y aura plusieurs niveaux d'accès possible et chaque niveau accède à des fonctionnalités spécifiques.

a) Accès Visiteur

- Consulter le pdf qui contient le fonctionnement du doctorat post concours dans notre université, une description du site et des conseils concernant le déroulement du doctorat.
- Consulter la Foire aux Questions(FAQ).
- Contacter le support.
- Voir les informations concernant la faculté des sciences Tlemcen (mail et numéro de téléphone).

b) Accès secrétaire

- Se connecter grâce à un compte prédéfini.
- Remplir les informations des nouveaux doctorants.
- Créer un compte pour tous les doctorants et enseignants.
- Mettre à jour les données d'un doctorant ou d'un enseignant.
- Supprimer un doctorant ou un enseignant.

c) Accès doctorants

- Se connecter grâce à un compte qui va lui être attribuer par l'administration.
- Remplir son état et son pourcentage d'avancement.
- Voir son profil.

d) Accès enseignants

- Se connecter grâce à un compte qui va lui être attribuer par l'administration.
- Voir la liste de tous ses doctorants.
- Valider ou modifier l'état d'avancement de tous ses doctorants.
- Activer un doctorant après validation.

e) Accès adjoint de post graduation

- Se connecter grâce à un compte prédéfini.
- Voir la liste des doctorants valides.
- Accéder à la liste de tous les doctorants/directeurs de thèse.
- Télécharger et imprimer après vérification et validation des données les fiches de réinscription et les attestations d'inscription pour faire les signatures.
- Notifier les doctorants par mail lorsque leurs fiches et/ou attestations sont valides et signées pour les récupérer (éviter les déplacements inutiles des doctorants à l'administration).
- Notifier les doctorants par mail lorsque leurs diplômes sont prêt (après soutenance).
- Voir les statistiques des dossiers des doctorants du département (comme par exemple, le nombre total de doctorants par type de doctorat).

1.4.2 Non Fonctionnels

- Sécuriser au maximum les informations des doctorants.
- L'application doit offrir des interfaces simples pour faciliter l'interaction avec l'utilisateur.
- Fournir une option de tri par ordre alphabétique au niveau des listes des doctorants/enseignants afin de faciliter la recherche.
- Mettre à disposition des utilisateurs un bouton de recherche pour faciliter la navigation.
- Adapter l'interface de notre application web à tout type d'écrans (tablette, mobile, ordinateur).

1.5 Conclusion

Après avoir présenté le contexte de notre travail et plus particulièrement le service de post graduation de la faculté des sciences qui s'occupe des dossiers des doctorants depuis leurs inscriptions jusqu'à l'obtention de leurs diplômes, nous avons réaliser le cahier de charges. On va pouvoir faire une étude détaillée de ce système.

Chapitre 2

Conception détaillée de notre système

DocUniT

2.1 Introduction

La conception est la deuxième étape primordiale dans la réalisation de notre projet, elle permet de créer un processus qui répond aux besoins des utilisateurs en tenant compte des contraintes qu'ils nous imposent.

Dans ce chapitre, nous allons réaliser la conception de notre projet de gestion de dossiers des doctorants. Pour ce faire, nous avons choisi le langage de modélisation UML qui est destiné à faciliter la conception des diagrammes nécessaires au développement de notre application web.

2.2 Le langage de modélisation UML

UML (Unified Modeling Language, qui veut dire langage de modélisation unifié en français) est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existant auparavant, et est devenu désormais la référence en terme de modélisation objet.

La programmation orientée objet consiste à modéliser grâce à l'informatique un ensemble d'éléments d'une partie du monde réel (que l'on appelle domaine) en un ensemble d'entités informatiques. Ces entités sont appelées objets. Il s'agit de données informatiques regroupant les principales caractéristiques des éléments du monde réel. La difficulté de cette modélisation consiste à créer une représentation abstraite, sous forme d'objets, d'entités ayant une existence matérielle ou bien virtuelle.

La modélisation objet consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation (indépendant d'un langage de programmation). Il s'agit donc de déterminer les objets présents et d'isoler leurs données et les fonctions qui les utilisent. Pour cela des méthodes ont été mises au point. Entre 1970 et 1990, de nombreux analystes ont mis au point des approches orientées objets, si bien qu'en 1994 il existait plus de 50 méthodes objet. Toutefois seules 3 méthodes ont véritablement émergé :

- La méthode OMT de James Rumbaugh.
- La méthode BOOCH de Grady Booch.
- La méthode OOSE de Ivar Jacobson.

A partir de 1994, Rumbaugh et Booch (rejoints en 1995 par Jacobson) ont unis leurs efforts pour mettre au point la méthode unifiée (unified method 0.8), incorporant les avantages de chacune des méthodes précédentes.

En janvier 1997, La méthode unifiée 1.0 change de nom et devient la méthode UML, qui va s'imposer comme étant une notation universelle pour la modélisation objet.

Ainsi, il est nécessaire qu'une méthode objet soit définie de manière rigoureuse et unique afin de lever les ambiguïtés. De nombreuses méthodes objet ont été définies, mais aucune ne s'est imposé en raison du manque de standardisation. C'est pourquoi l'ensemble des acteurs du monde informatique a fondé en 1989 l'OMG, une organisation à but non lucratif, dont le but est de mettre au point des standards garantissant la compatibilité entre des applications programmées à l'aide de langages objet et fonctionnant sur des réseaux hétérogènes. Enfin, en juillet 2005, la version 2.0 d'UML a été adopté par l'OMG. [8]

2.3 La méthode UP

Unified Process(que l'on peut traduire par Processus Unifié) est un processus de développement logiciel conçu par les développeurs d'UML Jacobson, Booch, Rumbaugh en 1999, regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel.

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques. Il permet de répondre aux quatre préoccupations suivantes :

- QUI participe au projet ?
- QUOI,qu'est-ce qui est produit durant le projet ?
- COMMENT doit-il être réalisé ?
- QUAND est réalisé chaque livrable ?

Le processus de développement UP, associé à UML, met en œuvre plusieurs principes, parmi eux l'utilisation des diagrammes de cas d'utilisation, de séquence et de classe que l'on va essayer de réaliser pour notre projet.[9]

2.4 Conception

Dans notre système, on aura 5 différents acteurs et chacun d'entre eux représente une entité externe qui interagit directement avec le système, ces acteurs sont :

- Le visiteur du site web.
- Les secrétaires des 4 départements de la faculté des Sciences.
- Le doctorant qui sera l'acteur principal du système.
- Le directeur/co-directeur de thèse.
- Les adjoints de post-graduation des 4 départements de la faculté des Sciences.

On a choisi d'utiliser 2 diagrammes principaux d'UML qui nous ont semblé être les plus adéquats pour notre étude, à savoir :

- Le diagramme de cas d'utilisation qui sert à modéliser les besoins des utilisateurs.
- Le diagramme de séquence système pour détailler tous les scénarios possible des acteurs avec le système.

Enfin, pour la création des différents diagrammes qui vont suivre, on a utilisé :

- Modelio qui est un outil open source qui prend en charge la norme UML2.
- Draw.io qui est un logiciel de dessin graphique open source qui permet de créer des diagrammes UML.

2.4.1 Diagrammes de Cas d'Utilisation

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (Humain ou Machine) et un système.

Dans un diagramme de cas d'utilisation il existe des acteurs qui interagissent avec des cas d'utilisation UC.

Les UC permettent de structurer les besoins des utilisateurs et les objectifs du système.[10]

Nous allons présenter ci-dessous les différents diagrammes qu'on a réalisé.

a) Pour le Visiteur

Le visiteur peut : (voir la figure 2.1)

- Consulter le PDF qui contient tous ce qu'il faut savoir sur le doctorat post-concours.
- Consulter la FAQ si besoin.
- Contacter le support en cas de problème.
- Voir les informations principales de la faculté des Sciences en bas de la page d'accueil.

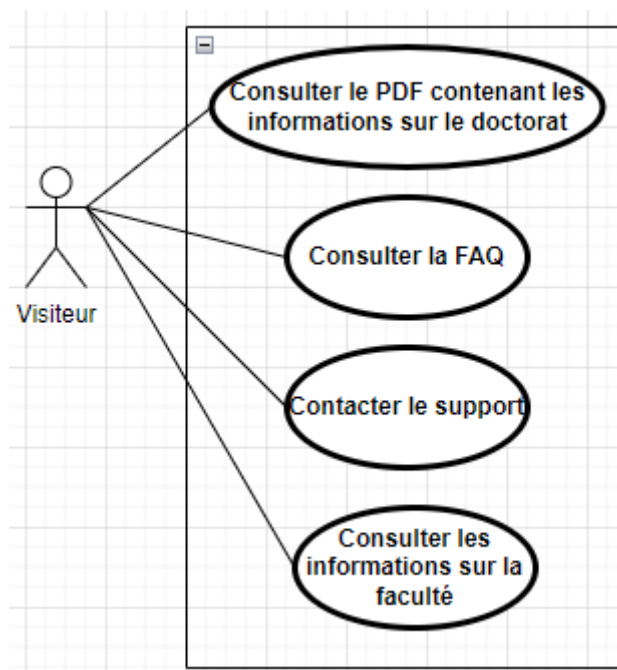


FIGURE 2.1 – Diagramme de Cas d'Utilisation du Visiteur

Pour tous les acteurs qui vont suivre, ils auront besoin de se connecter à notre application web. Pour cela, nous avons mis un include vers le UC se connecter pour chacun de leurs cas d'utilisation.

b) Pour la Secrétaire

La secrétaire peut : (voir la figure 2.2)

- Remplir les données statiques des doctorants(nom, prénom, ...) une et une seule fois lorsque son dossier sera valide par le comité de formation doctorale(sa 1ère année).

L'avantage de cette opération est que le doctorant ne va plus ressaisir ses données pendant tout son parcours, contrairement à la méthode existante. Il remplira pendant les autres années de réinscriptions seulement les informations concernant son avancement car une fiche y sera dédiée et cette dernière va être générée automatiquement pour impression.

- Attribuer des comptes pour les doctorants et les enseignants (directeur/co-directeur de thèse).
- Consulter, modifier ou supprimer les données qu'elle a créées.

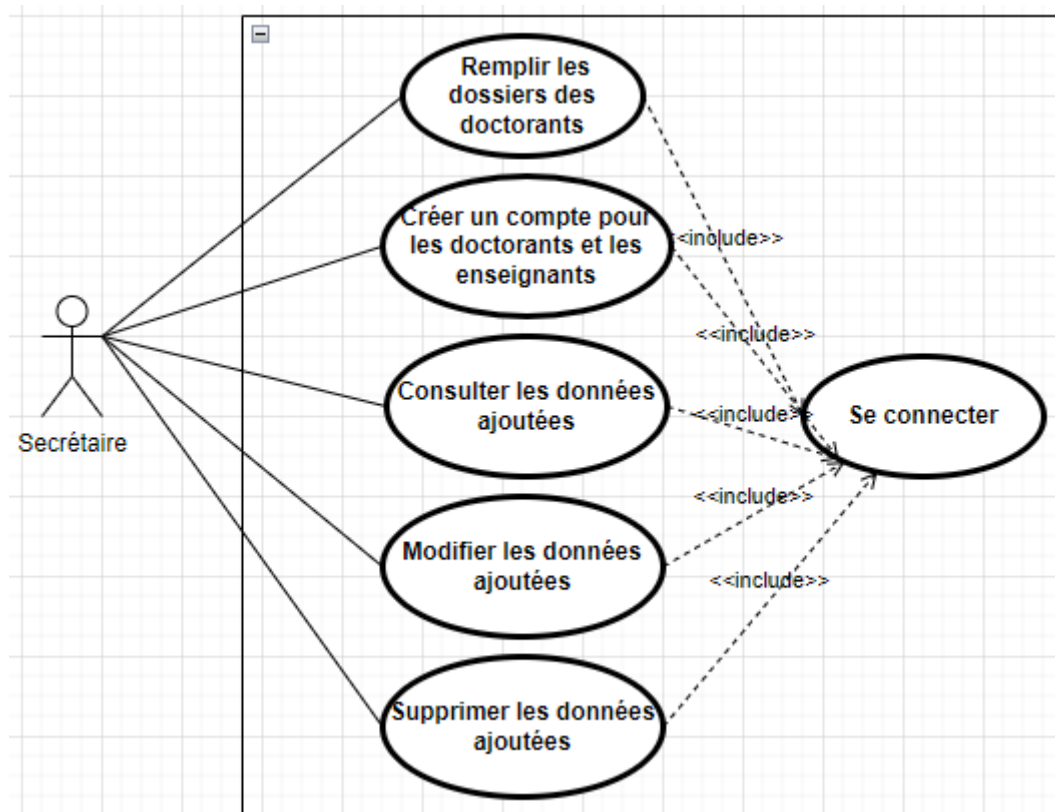


FIGURE 2.2 – Diagramme de Cas d'Utilisation de la Secrétaire

c) Pour le Doctorant

Le doctorant peut : (voir la figure 2.3)

- Renseigner sa date prévue de soutenance ainsi que son état et son pourcentage d'avancement.
- Consulter son profil.

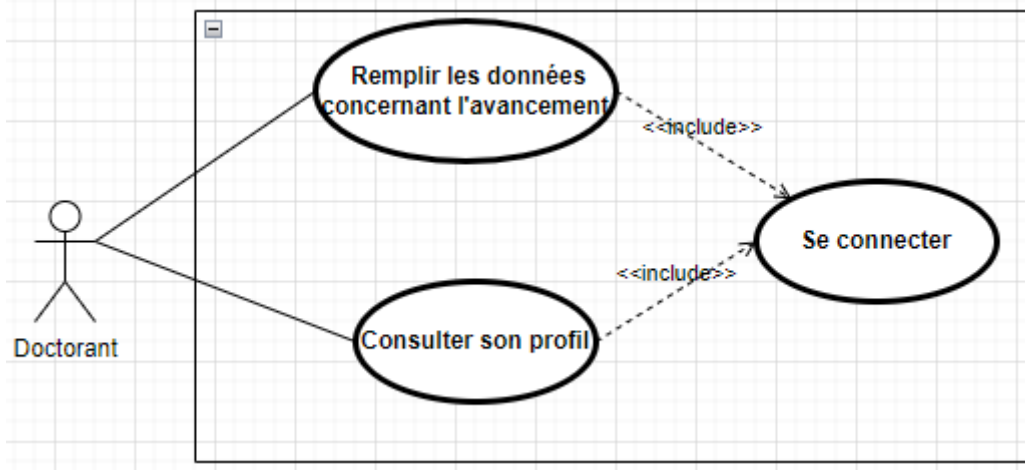


FIGURE 2.3 – Diagramme de Cas d'Utilisation du Doctorant

d) Pour le Directeur/Co-directeur de thèse

Le directeur/co-directeur de thèse peut : (voir la figure 2.4)

- Voir la liste complète de ses doctorants.
- Sélectionner un doctorant pour valider ou modifier les informations concernant son avancement.
- Sélectionner un doctorant pour l'activer après validation. Une fois qu'un doctorant sera activé, l'adjoint pourra imprimer sa fiche et/ou son attestation pour faire les signatures.

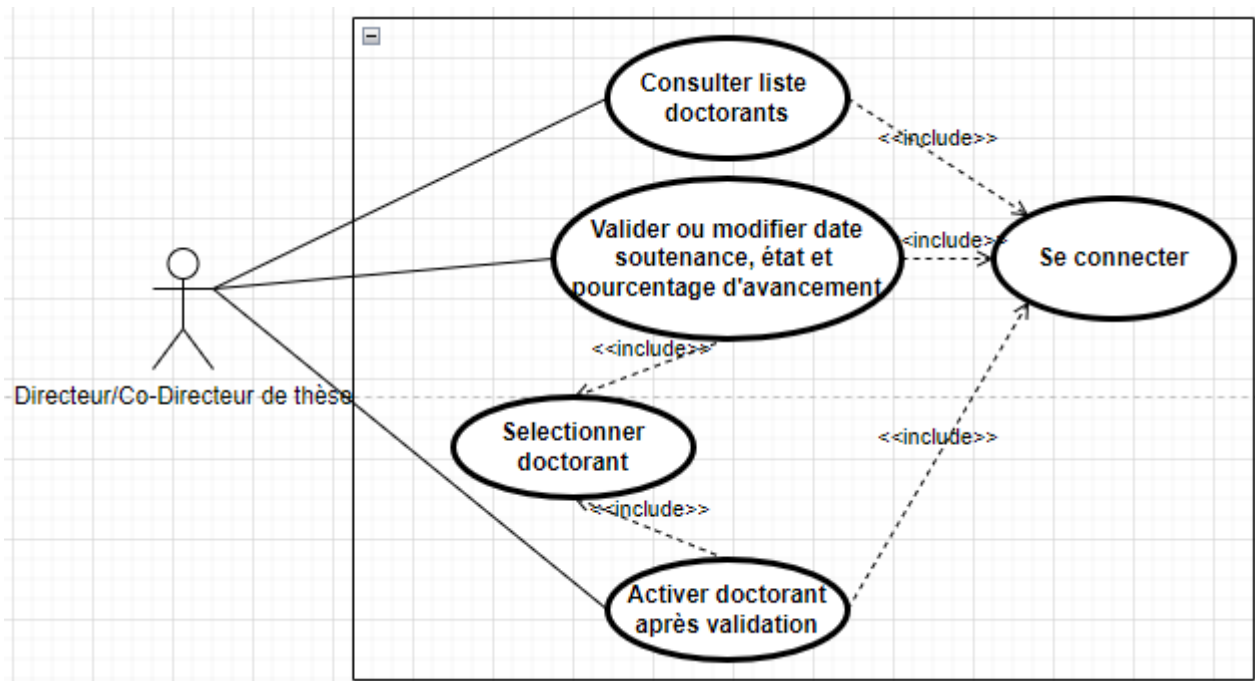


FIGURE 2.4 – Diagramme de Cas d'Utilisation du Directeur/Co-directeur de thèse

e) Pour l'Adjoint de post-graduation

L'adjoint de post-graduation peut : (voir la figure 2.5)

- Voir la liste complète des doctorants/enseignants.
- Voir la liste des doctorants valides.
- Sélectionner un doctorant valide pour imprimer son attestation d'inscription et sa fiche de réinscription.
- Consulter les statistiques.
- Sélectionner un doctorant et modifier son étape du doctorat pour l'informer si son dossier est validé par le CSD pour qu'il puisse préparer sa soutenance (étape soutenance) ou le notifier quand son diplôme est prêt (étape diplôme).

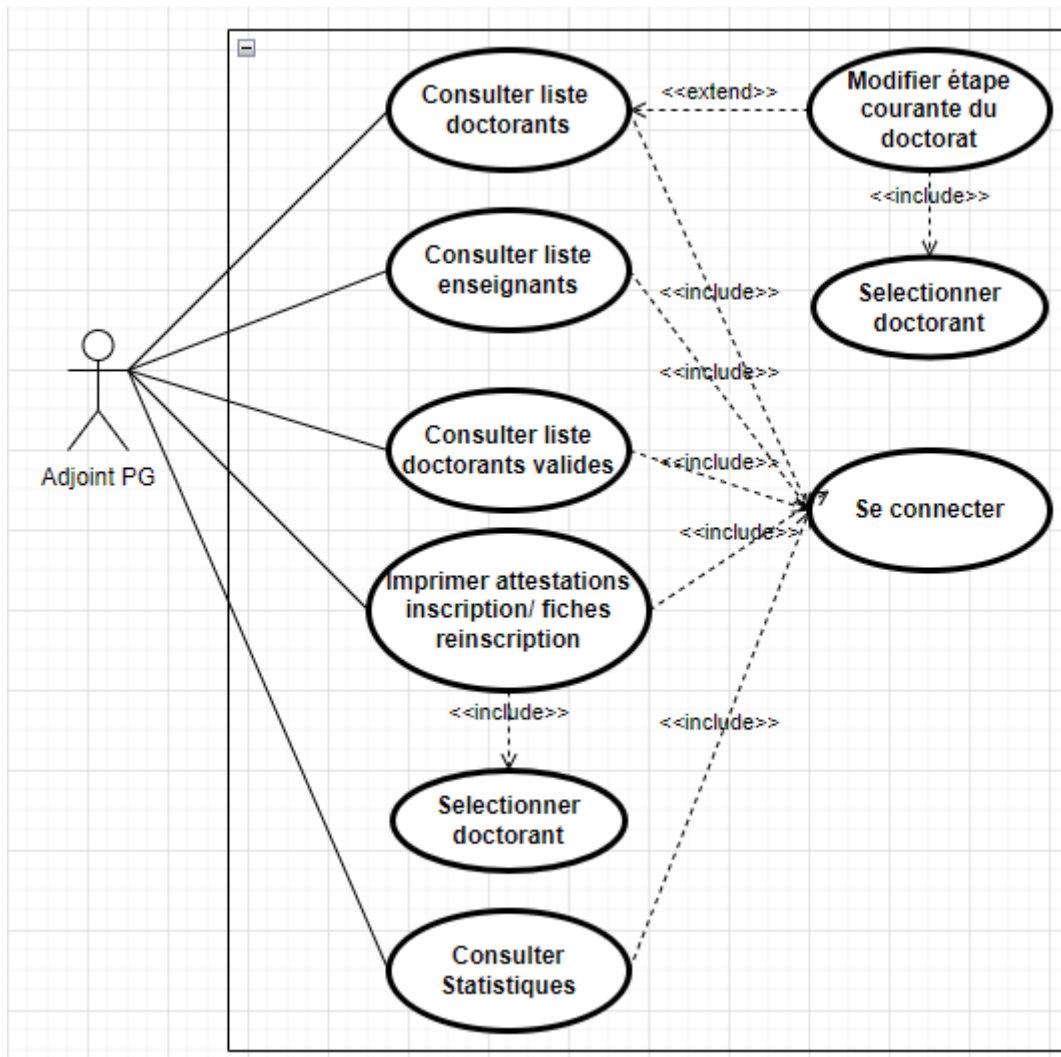


FIGURE 2.5 – Diagramme de Cas d'Utilisation de l'Adjoint de post-graduation

2.4.2 Diagrammes de Séquence Système

Les diagrammes de séquence, modélisent les interactions entre les objets dans un cas d'utilisation unique. Ils illustrent la manière dont les différentes parties d'un système interagissent entre elles pour exécuter une fonction, et l'ordre dans lequel les interactions se produisent lorsqu'un cas d'utilisation particulier est exécuté.

En termes plus simples, un diagramme de séquence montre les différentes parties d'un système qui travaillent dans une "séquence" pour obtenir quelque chose.[11]

Nous allons présenter les différents diagrammes qu'on a réalisé dans les lignes qui suivent.

a) Pour le Visiteur

Pour le cas d'utilisation contacter le support, il y aura la séquence suivante : (voir la figure 2.6)

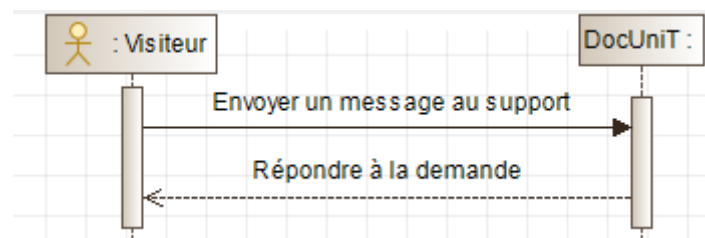


FIGURE 2.6 – Diagrammes de Séquence Système pour contacter le support

Pour tous les acteurs qui vont suivre, nous avons conçu la séquence suivante pour leurs permettre de se connecter. (voir figure 2.7)

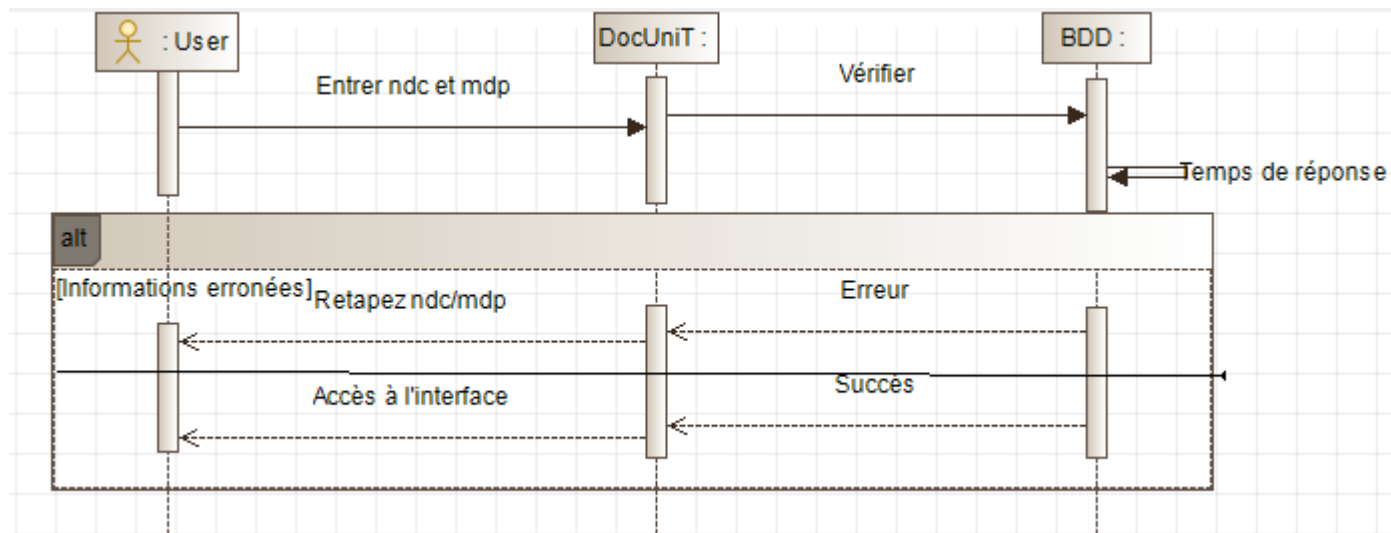


FIGURE 2.7 – Diagrammes de Séquence Système pour se connecter

b) Pour la Secrétaire

Dans la figure 2.8, nous avons établi le séquençement de tous les cas d'utilisation de la secrétaire.

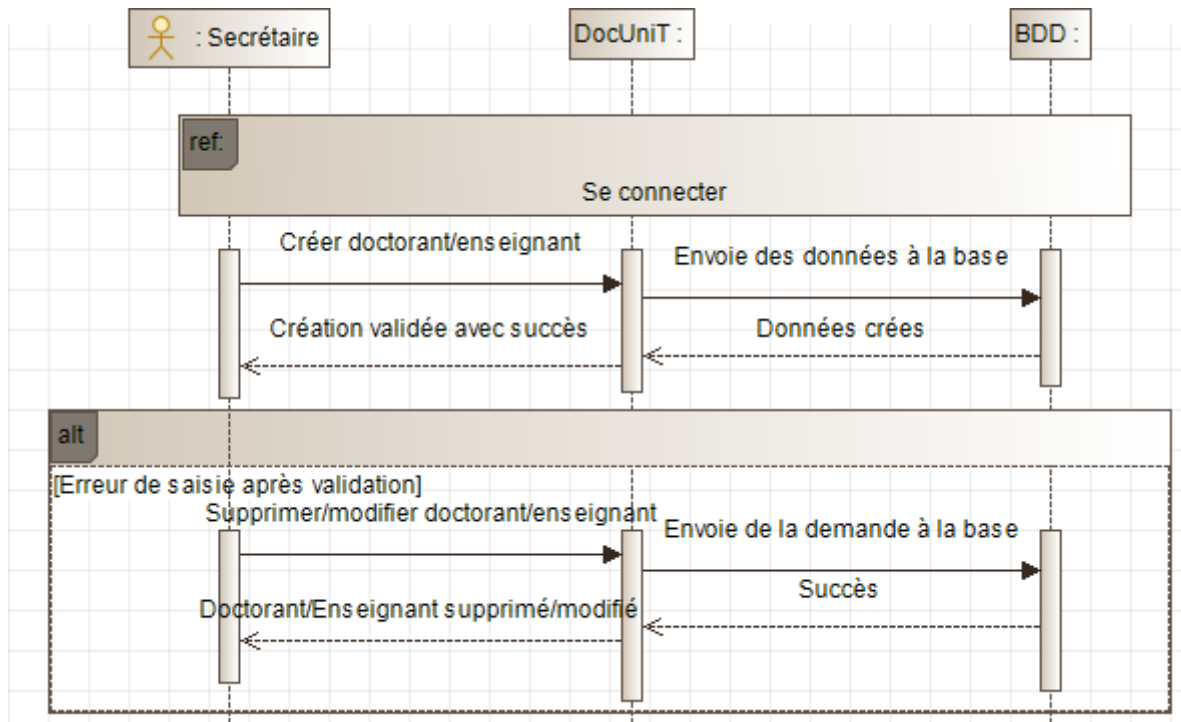


FIGURE 2.8 – Diagrammes de Séquence Système de la Secrétaire

c) Pour le Doctorant

Dans la figure 2.9, nous avons établi le séquençement de tous les cas d'utilisation du doctorant.

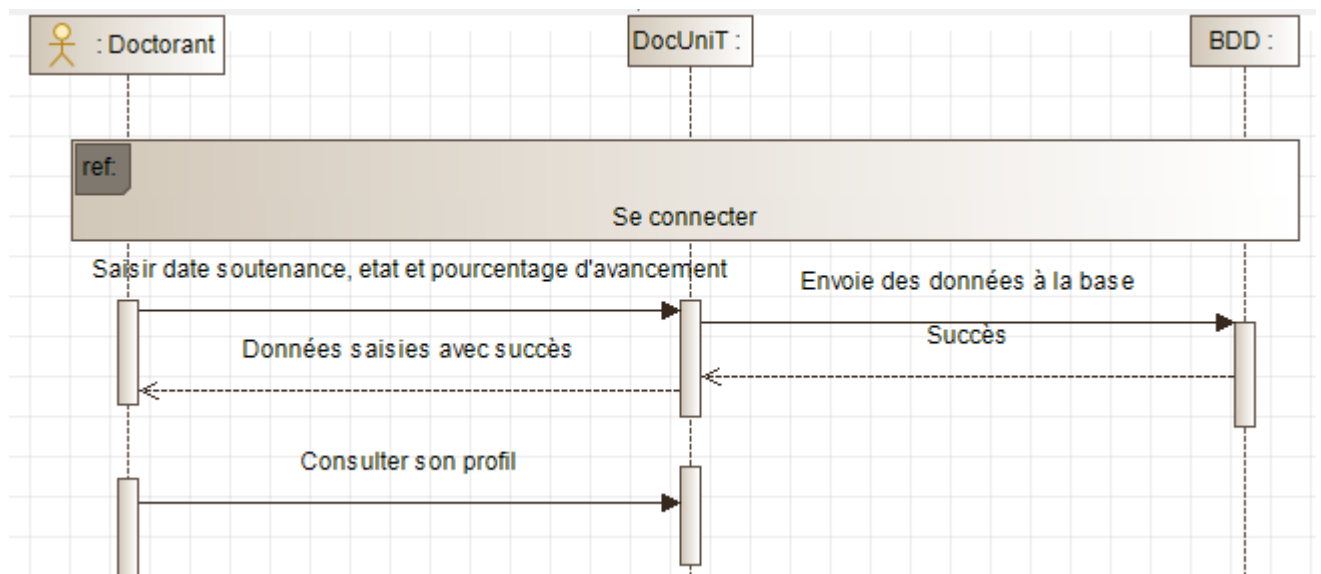


FIGURE 2.9 – Diagrammes de Séquence Système du Doctorant

d) Pour le Directeur/Co-directeur de thèse

Dans la figure 2.10, nous avons établi le séquençage de tous les cas d'utilisation de l'enseignant (directeur ou co-directeur de thèse).

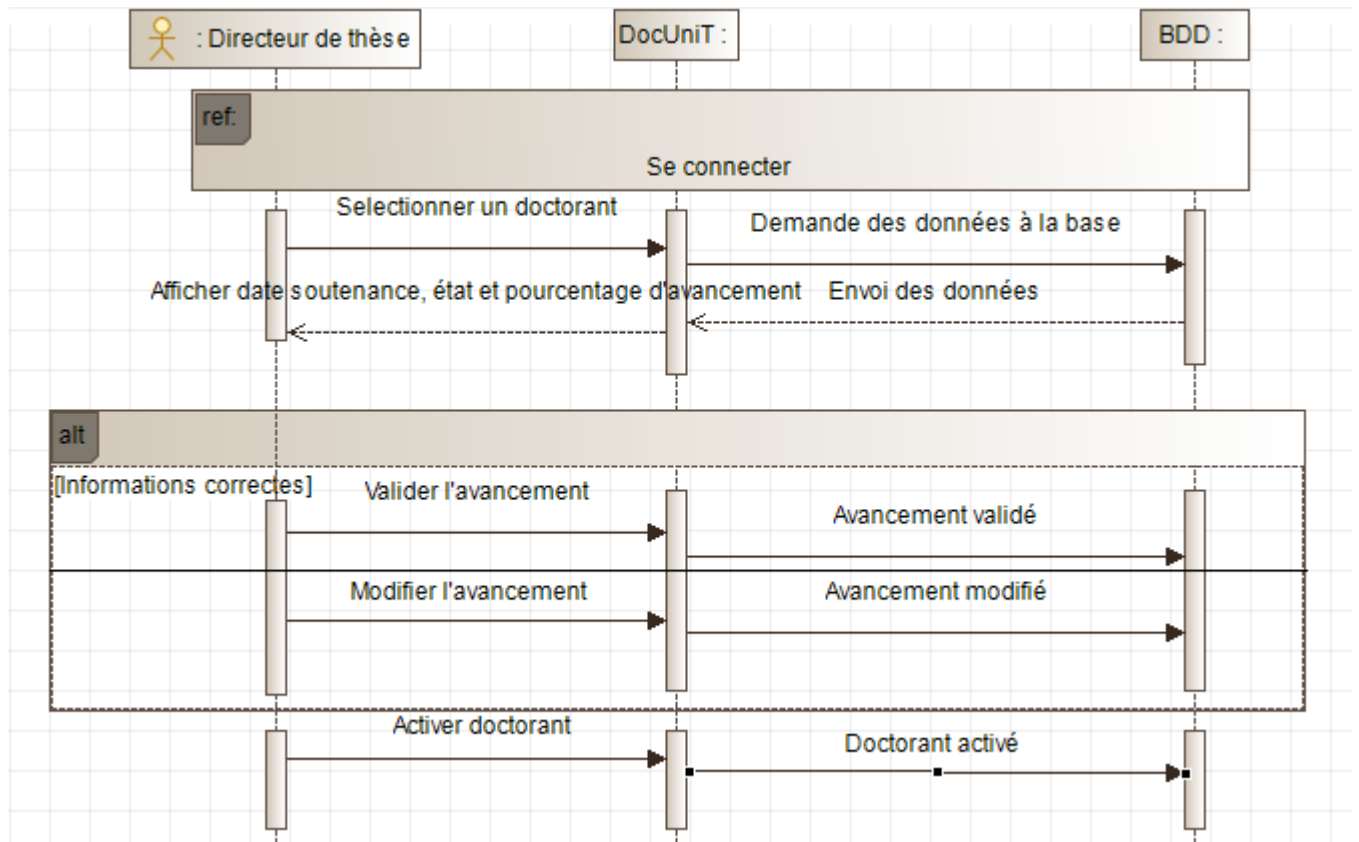


FIGURE 2.10 – Diagrammes de Séquence Système du Directeur/Co-directeur de thèse

e) Pour l'Adjoint de post-graduation

Dans la figure 2.11, nous avons établi le séquençage du cas d'utilisation : imprimer attestations inscription et fiches de réinscription de l'Adjoint de post-graduation.

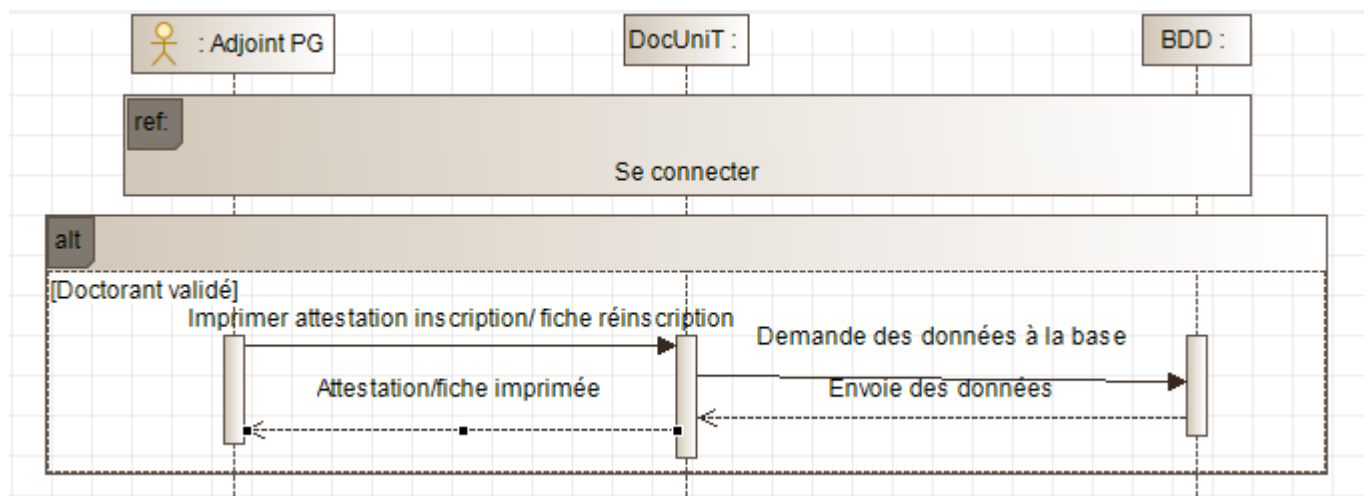


FIGURE 2.11 – Diagrammes de Séquence Système pour imprimer des documents

2.4.3 Conclusion

Dans ce chapitre, nous avons réalisé la conception détaillée du système de notre application web en utilisant le langage de modélisation UML et particulièrement les 2 diagrammes principaux de la méthode UP : le diagramme de cas d'utilisation et de séquence système.

Dans le chapitre suivant, nous allons aborder la dernière partie de notre travail qui représente la réalisation de notre application web.

Chapitre 3

Mise en œuvre de DocUnit

3.1 Introduction

La réalisation est la dernière étape de notre travail, elle est très importante car c'est grâce à elle que notre application web va voir le jour. Sa réussite dépend principalement du choix des outils choisis pour sa mise en place, afin de garantir une réponse optimale aux différents besoins des utilisateurs. Ce chapitre est donc dédié à cette étape où l'on va d'abord présenter tous les logiciels, outils et langages de programmation qu'on a utilisé pour ensuite enchaîner avec la création des schémas de notre bdd ainsi que la manipulation de leurs données et on finira avec les interfaces de l'application web.

3.2 Présentation des logiciels et outils de travail



FIGURE 3.1 – Logo des logiciels utilisés

3.2.1 Latex avec Overleaf

LaTeX est un système de composition de documents de haute qualité créée par Leslie Lamport, il comprend des fonctionnalités conçues pour la production de documents techniques et scientifiques. Il est disponible en tant que logiciel libre. [12]

Overleaf¹ est un éditeur Latex en ligne, collaboratif et en temps réel afin de pouvoir travailler sur un projet en simultané sur différents ordinateurs.

3.2.2 Adobe Photoshop

Adobe Photoshop est une application logicielle de retouche d'image et de retouche photo à utiliser sur les ordinateurs Windows ou MacOS. Il offre aux utilisateurs la possibilité de créer, d'améliorer ou de modifier des images, des photos ou des illustrations.

Changer d'arrière-plan, simuler une peinture réelle ou créer une vue alternative de l'univers est possible avec Adobe Photoshop. Il s'agit de l'outil logiciel le plus utilisé pour l'édition de photos, la manipulation d'images et la retouche pour de nombreux formats de fichiers d'images et de vidéos. Les outils de Photoshop permettent de modifier à la fois des images individuelles ainsi que de grands lots de photos.[13]

Nous l'avons utilisé pour réaliser le logo de notre application web ainsi que pour les images présentes dans nos interfaces.

1. Editeur Latex en ligne <https://fr.overleaf.com/>

3.2.3 GitHub

Git est un logiciel de gestion de version créé par Linus Torvalds, auteur du noyau Linux. Il permet d'enregistrer les différentes modifications effectuées sur un projet et de pouvoir retourner à une version précédente du projet.

Dans le langage des systèmes de gestion de version, la copie de l'intégralité des fichiers d'un projet et de leur version située sur le serveur central est appelé un dépôt. Git appelle également cela "repository".

GitHub est un service en ligne qui permet d'héberger des dépôts ou repository Git. C'est le plus grand hébergeur de dépôts Git du monde.

Une grande partie des dépôts hébergés sur GitHub sont publics, ce qui signifie que n'importe qui peut télécharger le code de ces dépôts et contribuer à leur développement en proposant de nouvelles fonctionnalités.

Pour récapituler, Git est un logiciel de gestion de version tandis que GitHub est un service en ligne d'hébergement de dépôts Git qui fait office de serveur central pour ces dépôts.[14]

Nous avons utilisé GitHub pour pouvoir travailler séparément et simultanément sur nos ordinateurs respectifs en s'échangeant nos versions à l'aide de la commande "PUSH" de GitHub.

3.2.4 Postman

Postman est un outil de développement d'API (interface de programmation d'applications) qui permet de créer, tester et modifier des API. Presque toutes les fonctionnalités qui pourraient être nécessaires à tout développeur sont encapsulées dans cet outil. Il est utilisé pour rendre les développements d'API faciles et simples. Il a la capacité de faire différents types de requests HTTP (GET, POST, PUT, PATCH), de sauvegarder des environnements pour une utilisation ultérieure, de convertir l'API en code pour différents langages (comme JavaScript, Python).[15]

On a décidé de travailler avec cet outil pour pouvoir tester le bon fonctionnement du back-end de notre application web pendant son développement.

3.2.5 Visual Studio Code (VSC)

VSC est un éditeur de code source léger mais puissant s'exécutant sur votre bureau et est disponible sur les systèmes d'exploitation Windows, macOS et Linux. Il a été développé par Microsoft. Il est livré avec un support intégré pour JavaScript, TypeScript, CSS et HTML et dispose d'un riche écosystème d'extensions pour d'autres langages via l'interface extension présente dans l'éditeur.[16]

Cet éditeur a été utilisé pour toute la programmation de l'application web.

3.3 Présentation des technologies pour la programmation



FIGURE 3.2 – la MERN Stack

MERN est un acronyme représentant un ensemble de technologies (à la fois côté serveur et côté client) permettant de développer des applications web full stack. Les 4 technologies utilisées sont :

- MongoDB
- Express.js
- React
- Node.js

La popularité de la MERN Stack vient du fait qu'elle rend le développement d'une application web full stack facile et rapide grâce à l'utilisation d'un seul langage de programmation : le Javascript.[17]

Pour mieux comprendre la mise en relation entre le coté client et le coté base de données en passant par le serveur, voici un schéma explicatif :[18]

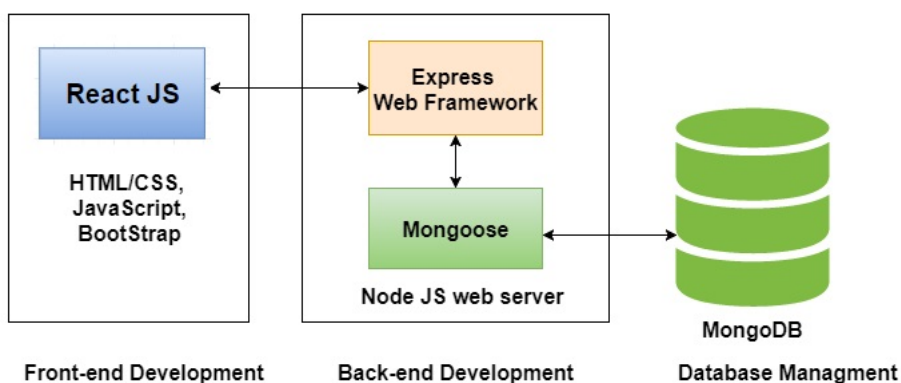


FIGURE 3.3 – la MERN Stack

3.3.1 React pour le Front-end

C'est un framework de développement front-end conçu par Facebook. Il permet de mettre en place la partie client de l'application web en allant consommer une API permettant de récupérer des données stockées en base de données. Ces données sont ensuite mises en places au sein des différentes pages de l'application web.[17]

3.3.2 Node et son framework Express pour le back-end

Node est un environnement d'exécution du langage JS hors d'un navigateur web, possédant quelques spécificités supplémentaires. Un serveur Node est donc simplement une application en JS côté serveur. Quant à lui, le framework Express est le framework de référence sur Node permettant de gérer des requêtes HTTP et ainsi de faire fonctionner correctement votre back-end.

Pour mettre en relation le framework Express et MongoDB, on utilise la bibliothèque Mongoose qui gère les relations entre les données, fournit une validation de schéma et fait la traduction entre les objets dans le code et la représentation de ces objets dans MongoDB.[17]

3.3.3 MongoDB pour la gestion de la base de données

La base de données MongoDB est une Bdd NoSQL basée sur des documents au sein de collections. À première vue, son format est plus simple à prendre en main qu'une Bdd relationnelle. Il n'y a donc pas de requête SQL pour récupérer des données. Les fonctions définies par MongoDB permettent de remplacer et simplifier ce processus.[17]

3.3.4 Avantages et Inconvénients de la MERN Stack

Parmi ses avantages, on peut citer :

- MERN est open-source et bénéficie du soutien de la communauté mondiale des développeurs, ce qui facilite la recherche pendant la programmation.
- Elle utilise la JS comme langage principal de programmation, ce qui la rend beaucoup plus efficace et moderne.
- Elle offre une expérience optimale aux utilisateurs grâce aux différents plugins qu'elle utilise pour créer des interfaces révolutionnaires plus fascinantes les unes que les autres.

Mais, comme toutes technologies de programmation, elle a quelques inconvénients :

- Elle n'est pas recommandée pour les applications de grande envergures.
- Elle possède des fonctionnalités de base qui sont limitées, ce qui implique que les développeurs doivent s'appuyer sur des services tiers. [19]

Il existe deux alternatives à la MERN stack qui sont :

- La MEAN stack : elle utilise AngularJS à la place de React. Les deux architectures sont presque similaires mais React est plus récent que AngularJS. On peut aussi dire que la MERN est une version mise à jour de la MEAN.
- La MEVN stack : elle remplace l'utilisation de React par Vue.js qui offre un développement rapide par rapport à React et Angular mais qui est limité au niveau des plugins disponibles et ne bénéficie

pas du total soutien de la communauté des développeurs car c'est un langage relativement nouveau. [19]

3.4 Présentation des outils de sécurisation de données

3.4.1 Le JavaScript Object Notation Web Token

Plus communément appelé JWT, est un jeton d'accès aux normes RFC 7519 qui permet d'échanger des données de manière sécurisée entre deux parties. Il contient toutes les informations importantes sur une entité, ce qui rend la consultation d'une bdd superflue et la session n'a pas besoin d'être stockée sur le serveur.

Les JWT sont particulièrement appréciés pour les opérations d'identification. Les messages courts peuvent être chiffrés et fournissent alors des informations sûres sur l'identité de l'expéditeur et si celui-ci dispose des droits d'accès requis. Les utilisateurs eux-mêmes ne sont qu'indirectement en contact avec les token, par exemple lorsqu'ils essaient de s'authentifier avec un nom d'utilisateur et un mot de passe. La véritable communication se fait entre les différentes applications du côté serveur et client. [20]

Le schéma suivant explique en détail le fonctionnement du JWT : [21]

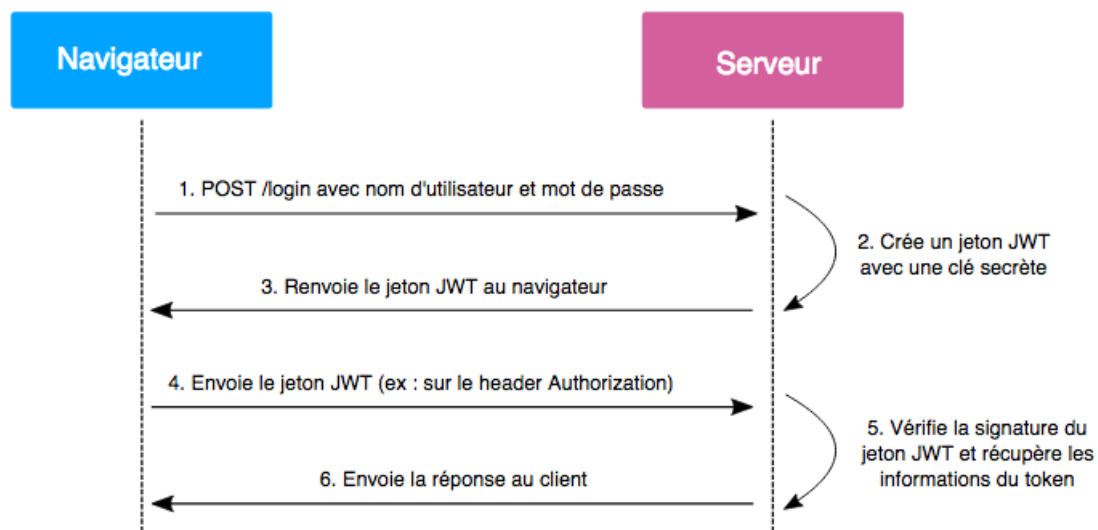


FIGURE 3.4 – Fonctionnement du JWT

En essayant de se connecter avec un nom d'utilisateur et un mot de passe, une clé secrète est déterminée avant l'utilisation du JWT. Dès qu'un utilisateur a entré avec succès ses données de connexion, le JWT est renvoyé avec la clé et stocké localement. Le transfert se fait par HTTPS afin de mieux protéger les données.

Lorsque l'utilisateur veut accéder à des ressources protégées comme une API ou un chemin d'accès protégé, le JWT sera envoyé par l'agent utilisateur comme paramètre (par exemple « jwt » pour les GET-Requests) ou comme en-tête d'autorisation (pour POST, PUT, OPTIONS, DELETE). L'interlocuteur peut déchiffrer le JWT et si le contrôle est réussi, on va exécuter sa demande. [20]

3.4.2 Fonction de Hachage

Une fonction de hachage consiste à transformer une donnée quelconque en une donnée de taille fixée. Bien entendu, la donnée en question peut avoir plusieurs formes. Ce peut être du texte, une image, ... mais dans tous les cas la donnée sera transformée en un texte binaire avant qu'on lui applique la fonction de hachage qui va la rendre sous forme d'une suite de caractères.[22]

On l'a utilisé pour sécuriser les mots de passe des utilisateurs en les stockant hachées dans notre bdd à travers une fonction de hachage avec une clé secrète aléatoire pour que même un administrateur ne puisse pas le décoder.

3.5 Réalisation

3.5.1 Création des schémas des collections MongoDB

Pour la réalisation de notre application web, on a eu besoin de 4 collections différentes afin d'attribuer les rôles adéquats aux utilisateurs, de leurs créer un compte et de permettre la récupération des données pour générer la fiche de réinscription et l'attestation d'inscription pour chaque doctorant.

Le premier schéma (voir la figure 3.5) est pour la secrétaire et l'adjoint de post-graduation. Il permet de leur attribuer un compte par rapport à leur profile comme étant la secrétaire ou l'adjoint de post-graduation de leur département.

```
const SecSchema = new mongoose.Schema({
  username: {
    type: String,
    required: true,
  },
  password: {
    type: String,
    required: true,
  },
  role: {
    type: String,
    required: true,
  },
  dept: {
    type: String,
    required: true,
  }
});
```

FIGURE 3.5 – Schéma Secrétaire et Adjoint de post-graduation

Le deuxième schéma (voir la figure 3.6) est pour l'enseignant (directeur de thèse et co-directeur de thèse). Il permet de stocker ses informations personnelles, de lui attribuer un compte et le définit comme étant un enseignant dans son département.

```
const enseignantSchema = new mongoose.Schema({
  ensnom: {
    type: String,
    required: true,
  },
  ensprenom: {
    type: String,
    required: true,
  },
  ensgrade: {
    type: String,
    required: true,
  },
  ensetabori: {
    type: String,
    required: true,
  },
  enslaborata: {
    type: String,
    required: true,
  },
  ensnumtel: {
    type: String,
    required: true,
  },
  ensmail: {
    type: String,
    required: true,
  },
  ensusername: {
    type: String,
    required: true,
    unique: true,
  },
  enspassword: {
    type: String,
    required: true,
  },
  ensdept: {
    type: String,
    required: true,
  },
  role: {
    type: String,
    required: true,
  }
});
```

FIGURE 3.6 – Schéma Enseignant

Le troisième schéma (voir la figure 3.7) est pour le doctorant. Il permet de stocker ses informations personnelles, les informations principales de ses directeur et co-directeur de thèse, de lui attribuer un compte et le définit comme étant un doctorant dans son département.

```

const doctorantSchema = new mongoose.Schema({
  nom: {
    type: String,
    required: true,
  },
  prenom: {
    type: String,
    required: true,
  },
  username: {
    type: String,
    required: true,
    unique: true,
  },
  password: {
    type: String,
    required: true,
  },
  dateN: {
    type: String,
    required: true,
  },
  lieuN: {
    type: String,
    required: true,
  },
  adresse: {
    type: String,
    required: true,
  },
  numtel: {
    type: String,
    required: true,
  },
  mail: {
    type: String,
    required: true,
  },
  etapro: {
    type: String,
    required: true,
  },
  preci: {
    type: String,
  },
  anebac: {
    type: Number,
    required: true,
  },
  seribac: {
    type: String,
    required: true,
  },
  numbac: {
    type: Number,
    required: true,
  },
  dept: {
    type: String,
    required: true,
  },
  catdoc: {
    type: String,
    required: true,
  },
  derdip: {
    type: String,
    required: true,
  },
  precii: {
    type: String,
  },
  spederdip: {
    type: String,
    required: true,
  },
  datederdip: {
    type: String,
    required: true,
  },
  datepremdoc: {
    type: String,
    required: true,
  },
  spedoc: {
    type: String,
    required: true,
  },
  laborata: {
    type: String,
    required: true,
  },
  intithe: {
    type: String,
    required: true,
  },
  role: {
    type: String,
    required: true,
  },
  dirnom: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
  dirprenom: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
  dirgrade: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
  codirnom: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
  codirprenom: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
  codigrade: {
    type: String,
    required: true,
    ref: 'Enseignant',
  },
});

```

FIGURE 3.7 – Schéma Doctorant

Le quatrième schéma (voir la figure 3.8) est lui aussi pour le doctorant. Il permet de stocker toutes les informations nécessaires concernant son avancement durant la période du doctorat.

```
const avancementSchema = new mongoose.Schema({
  username: {
    type: String,
    required: true,
  },
  pctav: {
    type: String,
    required: true,
  },
  datesout: {
    type: String,
    required: true,
  },
  etav: {
    type: String,
    required: true,
  },
  aneactu: {
    type: Number,
    required: true,
  },
  status: {
    type: Boolean,
    required: true,
  },
});
```

FIGURE 3.8 – Schéma Avancement du Doctorant

3.5.2 Manipulation des données à travers les collections

Dans cette section, nous allons présenter un exemple pour montrer la façon dont nous avons procédé pour la manipulation des données de notre base à travers les 3 opérations suivantes : la création(create), la mise à jour(update) et la suppression(delete).

a) La création des données

On va prendre comme exemple la création d'un enseignant.

— Coté Back-end :

```
router.post("/registerens", async (req, res) => (
  const newEns = new Enseignant(
    ensnom : req.body.ensnom,
    ensprenom : req.body.ensprenom,
    ensgrade : req.body.ensgrade,
    ensetabori : req.body.ensetabori,
    enslaborata : req.body.enslaborata,
    ensnumtel : req.body.ensnumtel,
    ensmail : req.body.ensmail,
```

```
ensusername,  
enspassword : req.body.enspassword,  
ensdept : req.body.ensdept,  
role : 'ens'  
);  
const savedEns = await newEns.save();  
res.json(savedEns);
```

— Coté Front-end :

```
await axios.post("http://localhost:5000/users/registerens",  
(ensnom : EnsNom.current.value,  
ensprenom : EnsPrenom.current.value,  
ensgrade : EnsGrade.current.value,  
ensetabori : EnsEtabori.current.value,  
enslaborata : EnsLaborata.current.value,  
ensnumtel : EnsNumtel.current.value,  
ensmail : EnsMail.current.value,  
ensusername : EnsName.current.value,  
enspassword : EnsPassword.current.value,  
ensdept : userData.user.dept )
```

b) La mise à jour des données

On va prendre comme exemple la requête qui nous permet de mettre à jour l'avancement d'un doctorant.

— Coté Back-end :

```
router.put('/update/avncdoc/:username', async (req, res) => (  
let aneactu = req.body;  
const ExDoc = await Avancement.findOne(username : req.params.username , aneactu : aneactu);  
if (ExDoc)  
return res.status(400).json( msg : "avancement déjà validé" );  
await Avancement.updateOne( username : req.params.username ,  
set : (  
pctav : req.body.pctav,  
datesout : req.body.datesout,  
etav : req.body.etav,  
status : false,  
aneactu,  
)).
```

— Coté Front-end :

```
await axios.put("http://localhost:5000/users/update/avncdoc/" + userData.user.username,
  ( username: userData.user.username,
    pctav: Avancementpct.current.value,
    datesout: Avancementdatesout.current.value,
    etav: Avancementetav.current.value,
    aneactu: calculedate()+1
  )
)
```

c) La suppression des données

On va prendre comme exemple la suppression d'un doctorant.

— Coté Back-end :

```
router.delete("/deletedoc", indx, async (req, res) => (
  const deletedDoc = await Doctorant.findOneAndDelete(username: req.docun);
  res.json(deletedDoc);
));
```

— Coté Front-end :

```
(const doc = [...docs];
const index = doc.findIndex(
  (element) => element.id === deleteDocDialogRow.id );
axios.delete(
  "http://localhost:5000/users/deletedoc",
  ( headers: (
    "x-delete": doc[index].ndc ,
  ), ) ); doc.splice(index, 1);
setDocs(doc);)
```

3.5.3 Présentation des interfaces de notre application web DocUnit

a) Interface du visiteur

L'image suivante (figure 3.9) montre l'interface d'accueil de notre application web.



Features



Simplicité

Interfaçage clair et intuitif pour faciliter l'interaction avec l'utilisateur.



Connectivité

Finis l'obligation de déplacement à l'université, vous pouvez vous inscrire pour vous connecter quand vous voulez et où vous voulez.



Sécurité

Stockage des informations personnels dans une base de données dématérialisée, 100% fiable.



Rapidité

Fournir une interface qui se charge rapidement et qui délivre les contenus demandés en quelques secondes.



Facilité

Permettre aux doctorants de recevoir leurs attestations et aux encadrants de gérer à travers une interface l'ensemble des doctorants pour mieux voir leurs états d'avancement.



24/7 Support

Pour tout problème, l'application contient un FAQ pour répondre aux questions les plus récurrentes et une messagerie pour contacter le support en cas de découverte d'un nouveau problème.

Foire Aux Questions



FIGURE 3.9 – Interface d'accueil

La page d'accueil de notre application web contient 4 parties principales : la première, pour la présentation de son but et la consultation d'un PDF contenant des informations sur le déroulement du doctorat. La deuxième partie met en avant ses caractéristiques. La troisième contient un petit FAQ pour répondre aux questions les plus probables. Pour finir, la quatrième partie, qui elle contient des informations sur la faculté des sciences ainsi qu'un moyen de contacter le support de l'application web en cas de problème.

En haut à droite de la page d'accueil, il y a un bouton connexion qui permet d'ouvrir l'interface suivante (figure 3.10) dans la même page :

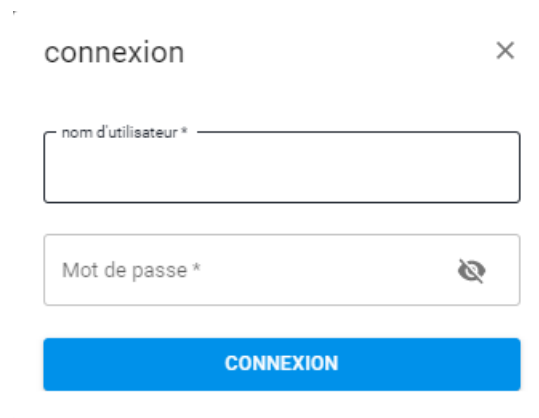
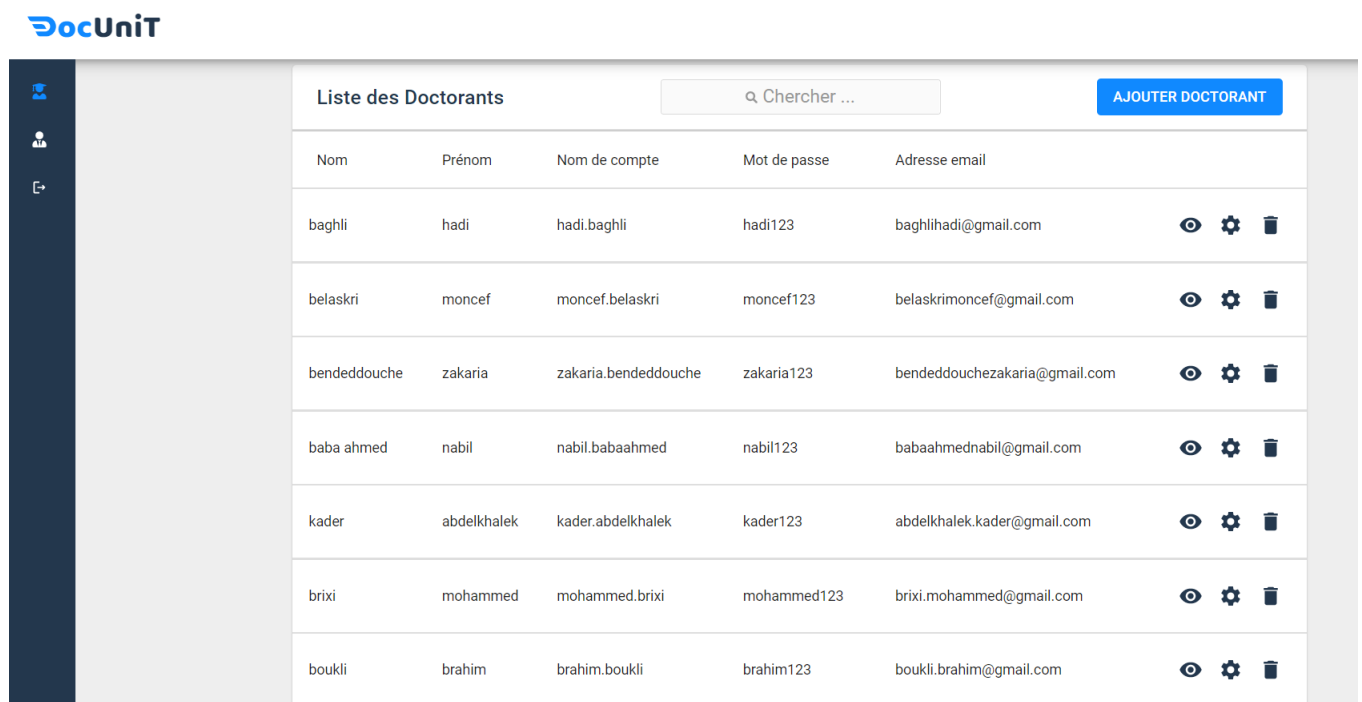


FIGURE 3.10 – Interface de Connexion

b) Les interfaces de la Secrétaire



Nom	Prénom	Nom de compte	Mot de passe	Adresse email	
baghli	hadi	hadi.baghli	hadi123	baghlihadi@gmail.com	👁 ⚙ 🗑
belaskri	moncef	moncef.belaskri	moncef123	belaskrimoncef@gmail.com	👁 ⚙ 🗑
bendeddouche	zakaria	zakaria.bendeddouche	zakaria123	bendeddouchezakaria@gmail.com	👁 ⚙ 🗑
baba ahmed	nabil	nabil.babaahmed	nabil123	babaahmednabil@gmail.com	👁 ⚙ 🗑
kader	abdelkhalek	kader.abdelkhalek	kader123	abdelkhalek.kader@gmail.com	👁 ⚙ 🗑
brixi	mohammed	mohammed.brixi	mohammed123	brixi.mohammed@gmail.com	👁 ⚙ 🗑
boukli	brahim	brahim.boukli	brahim123	boukli.brahim@gmail.com	👁 ⚙ 🗑

FIGURE 3.11 – Interface de la liste des doctorants

Grâce à cette interface, la secrétaire peut voir la liste des doctorants de son département, ajouter un doctorant à l'aide du bouton en haut à droite qui permet d'ouvrir l'interface d'ajout (figure 3.12), voir un doctorant à l'aide du premier bouton à gauche parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de consultation (figure 3.13), modifier un doctorant grâce au bouton au milieu parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de modification (figure 3.14), supprimer un doctorant avec le dernier bouton à droite parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de suppression (figure 3.15). Elle peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un doctorant grâce au bouton de recherche au milieu.

Création d'un Doctorant

Informations Personnelle

Nom *	Prénom *		
Né(e) le * jj/mm/aaaa	à *	Adresse *	

Contact

N° de téléphone *	Email *
-------------------	---------

Informations Professionnelle

Etat professionnel *		
Année d'obtention du BAC *	Série du BAC *	N° du BAC *

RETOUR VALIDER

FIGURE 3.12 – Interface pour ajouter un doctorant

Données d'un Doctorant

Informations Personnelle

Nom *	Prénom *		
belaskri	moncef		
Né(e) le *	à *	Adresse *	
25/05/1998	tln	oujlida	

Contact

N° de téléphone *	Email *
0555555555	belaskrimoncef@gmail.com

Informations Professionnelle

Etat professionnel *		
nonsal		
Année d'obtention du BAC *	Série du BAC *	N° du BAC *
2016	mi	2
Fiche de reinscription en *	Dernier diplôme obtenu *	
sci	mas	
Spécialité dernier diplôme obtenu *	Date de son obtention *	

FIGURE 3.13 – Interface pour voir un doctorant

Modification d'un Doctorant

Informations Personnelle

Nom * Prénom *

Né(e) le * Adresse *

Contact

N° de téléphone * Email *

Informations Professionnelle

Etat professionnel *

Année d'obtention du BAC * Série du BAC * N° du BAC *

RETOUR

FIGURE 3.14 – Interface pour modifier un doctorant

Confirmation

Voulez vous vraiment supprimer le doctorant **baghli hadi** de votre liste?

FERMER

FIGURE 3.15 – Interface pour supprimer un doctorant

Liste des Enseignants					<input type="text" value="Search..."/>	<input type="button" value="AJOUTER ENSEIGNANT"/>
Nom	Prénom	Nom de compte	Mot de passe	Adresse email		
elyebdri	zeyneb	zeyneb.elyebdri	zeyneb123	elyebdrizeyneb@gmail.com	<input type="checkbox"/>	<input type="checkbox"/>
benammar	abdelkrim	abdelkrim.benammar	abdelkrim123	benammarabdelkrim@gmail.com	<input type="checkbox"/>	<input type="checkbox"/>
matallah	houcine	houcine.matallah	houcine123	matallahhoucine@gmail.com	<input type="checkbox"/>	<input type="checkbox"/>
benmansour	fazilet	fazilet.benmansour	fazilet123	benmansour.fazilet@gmail.com	<input type="checkbox"/>	<input type="checkbox"/>

1-4 of 4

FIGURE 3.16 – Interface de la liste des enseignants

Grâce à cette interface, la secrétaire peut voir la liste des enseignants de son département, ajouter

un enseignant à l'aide du bouton en haut à droite qui permet d'ouvrir l'interface d'ajout (figure 3.17), voir un enseignant à l'aide du premier bouton à gauche parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de consultation (figure 3.18), modifier un enseignant grâce au bouton au milieu parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de modification (figure 3.19), supprimer un enseignant avec le dernier bouton à droite parmi les trois boutons à droite de la liste qui permet d'ouvrir l'interface de suppression (figure 3.20). Elle peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un enseignant grâce au bouton de recherche au milieu.

Création d'un Enseignant

Informations Personnelles

Nom * Prénom *

Contact

N° de téléphone * Email *

Informations Professionnelle

Grade * Etablissement d'origine * Laboratoire de rattachement *

Identifiants

RETOUR VALIDER

FIGURE 3.17 – Interface pour ajouter un enseignant

Données d'un Enseignant

Informations Personnelles

Nom * elyebdri Prénom * zeyneb

Contact

N° de téléphone * 055555555 Email * elyebdrizeyneb@gmail.com

Informations Professionnelle

Grade * maca Etablissement d'origine * tim Laboratoire de rattachement * tim

Identifiants

Nom de compte * zeyneb.elyebdri Mot de passe *

RETOUR

FIGURE 3.18 – Interface pour voir un enseignant

Modification d'un Enseignant

Informations Personnelle

Nom * eleybdri Prénom * zeyneb

Contact

N° de téléphone * 0555555555 Email * eleybdrizeyneb@gmail.com

Informations Professionnelle

Grade * maca Etablissement d'origine * tim Laboratoire de rattachement * tim

Identifiants

Nom de compte * Mot de passe *

RETOUR VALIDER

FIGURE 3.19 – Interface pour modifier un enseignant

Confirmation

Voulez vous vraiment supprimer l'enseignant
eleybdri zeyneb de votre liste?

FERMER

CONFIRMER

FIGURE 3.20 – Interface pour supprimer un enseignant

c) L'interface du Doctorant

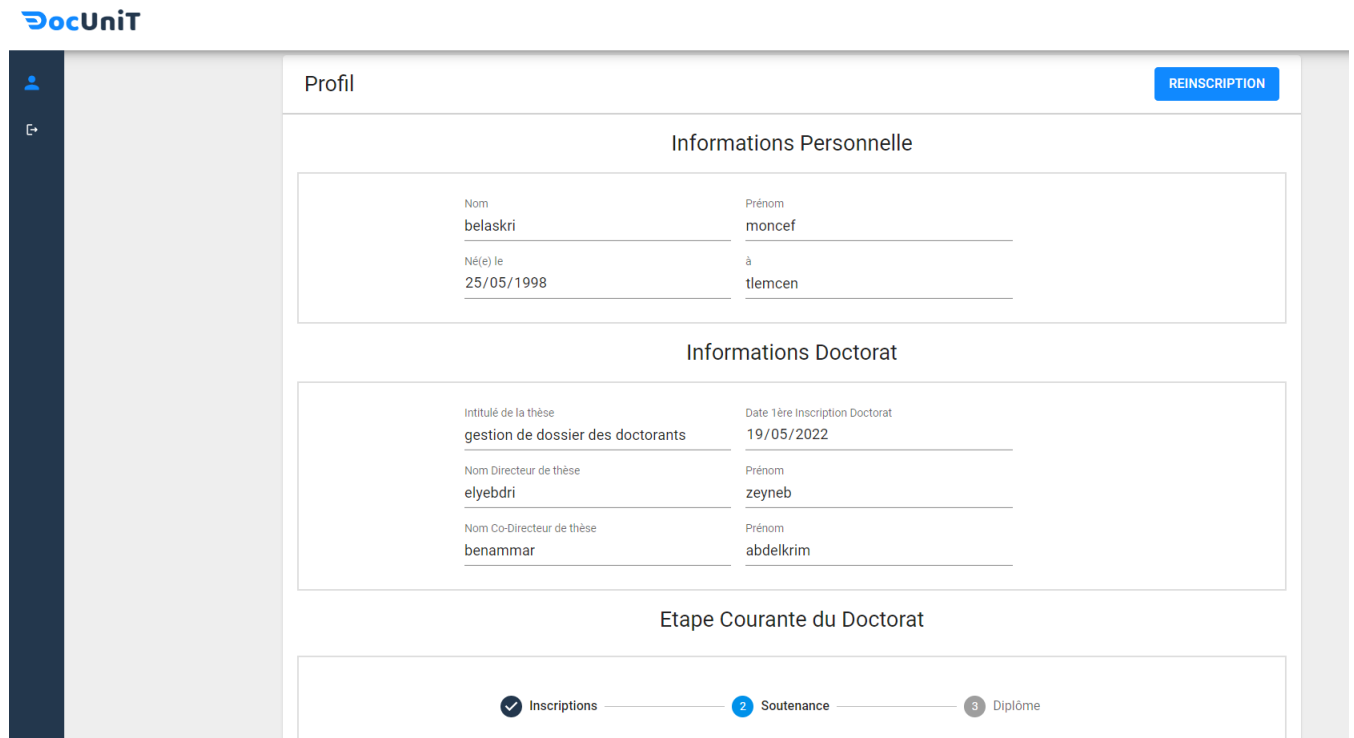


FIGURE 3.21 – Interface pour voir son profil

Pour faire sa réinscription, le doctorant doit appuyer sur le bouton en haut à droite et renseigner les informations demandées. S'il vient de s'inscrire pour la première fois, il verra l'interface suivante (voir la figure 3.22), sinon il aura celle d'après (voir la figure 3.23) avec la possibilité de la modifier une fois par an.

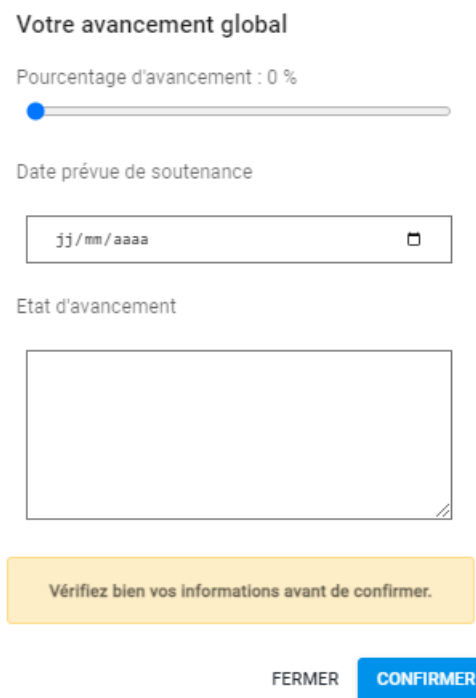



FIGURE 3.22 – Interface pour modifier son avancement

Votre avancement global

Pourcentage d'avancement : 35 %



Date prévue de soutenance

18/06/2022

Etat d'avancement

Pour mon état d'avancement, je suis entrain de rediger le mémoire. Actuellement, dans le chapitre 3 entrain de faire la réalisation de l'application web.

Vérifiez bien vos informations avant de confirmer.

FERMER CONFIRMER

FIGURE 3.23 – Interface pour voir son avancement

d) L'interface de l'Enseignant(Directeur/Co-directeur de thèse)

DocUniT

Liste des Doctorants

Q Search...

Nom	Prénom	Intitulé de la Thèse	Pourcentage d'avancement	Année courante	
baghli	hadi	gestion de dossier des doctorants	36 %	4	⚙️ 🔵
belaskri	moncef	gestion de dossier des doctorants	54 %	3	⚙️ 🔵
bendeddouche	zakaria	Réalisation d'une plateforme de gestion d'une salle de sport	70 %	1	⚙️ 🔵
baba ahmed	nabil	Gestion d'une compétition de football	23 %	5	⚙️ 🔵

1-4 of 4 < >

FIGURE 3.24 – Interface pour voir ses doctorants

Grâce à cette interface, l'enseignant peut voir la liste de ses doctorants, voir ou modifier l'avancement d'un doctorant à l'aide du premier bouton à gauche parmi les deux boutons à droite de la liste qui permet d'ouvrir l'interface d'avancement (voir la figure 3.25), activer un doctorant après validation de son avancement à l'aide du deuxième bouton à droite parmi les deux boutons à droite de la liste. Il peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un doctorant grâce au bouton de recherche en haut à droite.

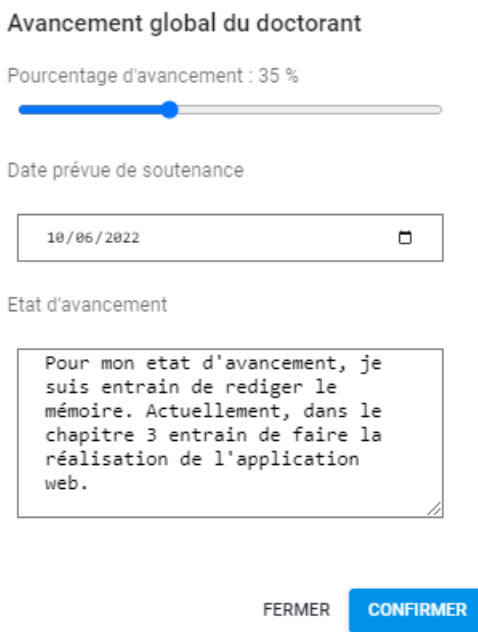


FIGURE 3.25 – Interface pour voir ou modifier l’avancement d’un de ses doctorants

e) Les interfaces de l’Adjoint de Post-Graduation

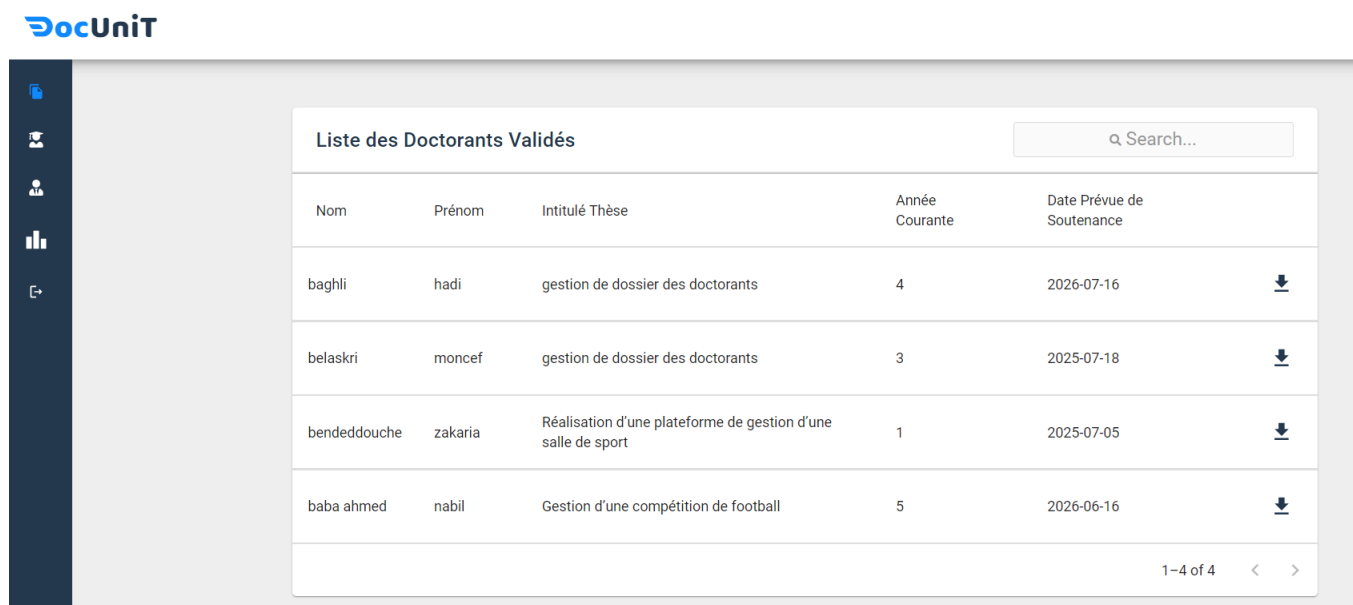


FIGURE 3.26 – Interface pour voir la liste des Doctorants activés

Grâce à cette interface, l’adjoint peut voir la liste des doctorants de son département qui sont activés, télécharger leurs attestations d’inscription et fiches de réinscription grâce au bouton à droite de la liste qui génère l’attestation d’inscription (voir les figure 3.27) et la fiche de réinscription (voir les figure 3.28, 3.29). Il peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un doctorant grâce au bouton de recherche en haut à droite.

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي و البحث العلمي

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Aboubekr BELKAID
TLEMCEM
Faculté des Sciences
Service de la Post Graduation
De la Recherche Scientifique et des Relations
Extérieures
Tel. & Fax: 043 21 63 72
Tel. & Fax: 043 21 63 68



جامعة أبو بكر بلقايد
تلمسان
كلية العلوم
مصلحة ما بعد التدرج
البحث العلمي والعلاقات
الخارجية

ATTESTATION D'INSCRIPTION

Je soussigné, Doyen de la faculté des sciences, atteste que:

L'étudiant(e): belaskri moncef

Né(e) le: 1998-05-25

à: tlm

Est régulièrement inscrit(e) sous le numéro: 2

En préparation de l'année: 1ère année doctorat

Pour l'année universitaire: 2021-2022

Option: sic

Département: Informatique

Tlemcen le:

N'est remis qu'un seul exemplaire à la personne concernée

FIGURE 3.27 – L'attestation d'inscription

ETAT D'AVANCEMENT :

Pour mon etat d'avancement, je suis entrain de rediger le mémoire. Actuellement, dans le chapitre 3 entrain de faire la réalisation de l'application web.

*** Pourcentage d'avancement : 35** *** Date prévue de soutenance : 2022-06-10**

*Avis et Visa du C.FD	Avis et Visa du C.S.D
<i>NB :</i> *Avis et visa du CFD : sont demandés uniquement pour les doctorants inscrits en Doctorat LMD.	
Avis et Visa du C.S.F	Avis et Visa du Doyen

NB : -La fiche doit être remplie par micro-ordinateur et imprimé sur une seule feuille (recto/verso).
 -Toute fiche remplie au stylo sera rejetée et l'attestation de réinscription ne sera pas délivrée.
Faculté des Sciences - Tidjani HADDAM : Tél : 043 21 63 70 / Tél & Fax : 043 21 63 68 / 043 21 63 71
Site Web: www.fs.univ-tlemcen.dz / Email: vdrpg.facsience@gmail.com

FIGURE 3.29 – La 2ème page de la fiche de réinscription

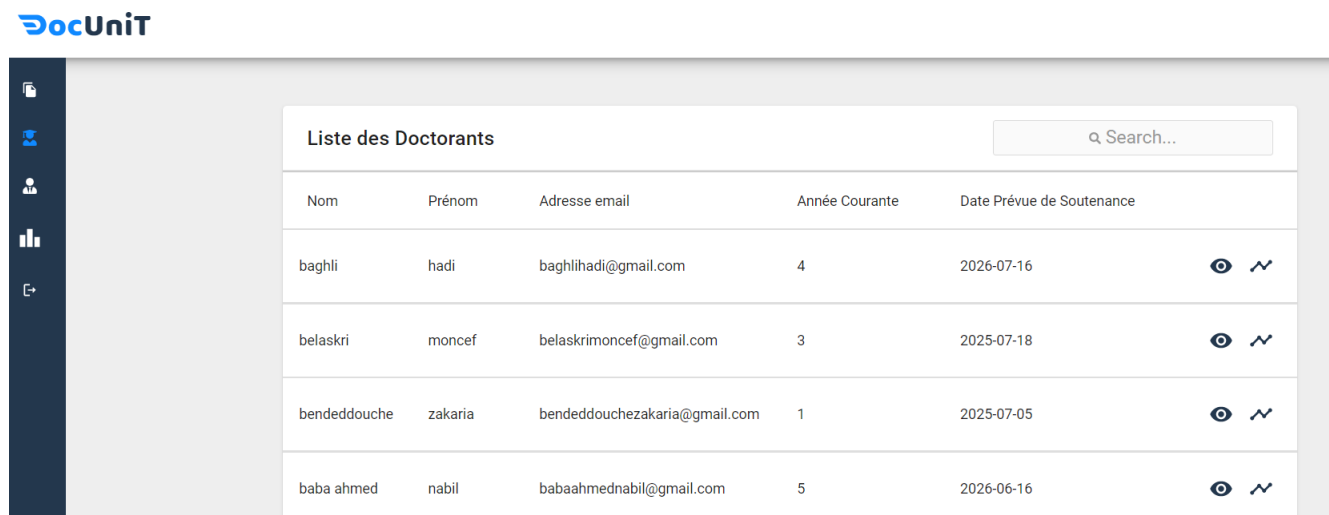


FIGURE 3.30 – Interface pour voir la liste des doctorants

Grâce à cette interface, l’adjoint peut voir la liste de tous les doctorants de son département, consulter leurs informations à l’aide du premier bouton à gauche parmi les boutons à droite de la liste qui ouvre l’interface de consultation (voir la figure 3.31) et choisir leur étape en cours grâce au bouton le plus à droite de la liste qui permet d’ouvrir l’interface de modification de l’étape (voir la figure 3.32). Il peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un doctorant grâce au bouton de recherche en haut à droite.

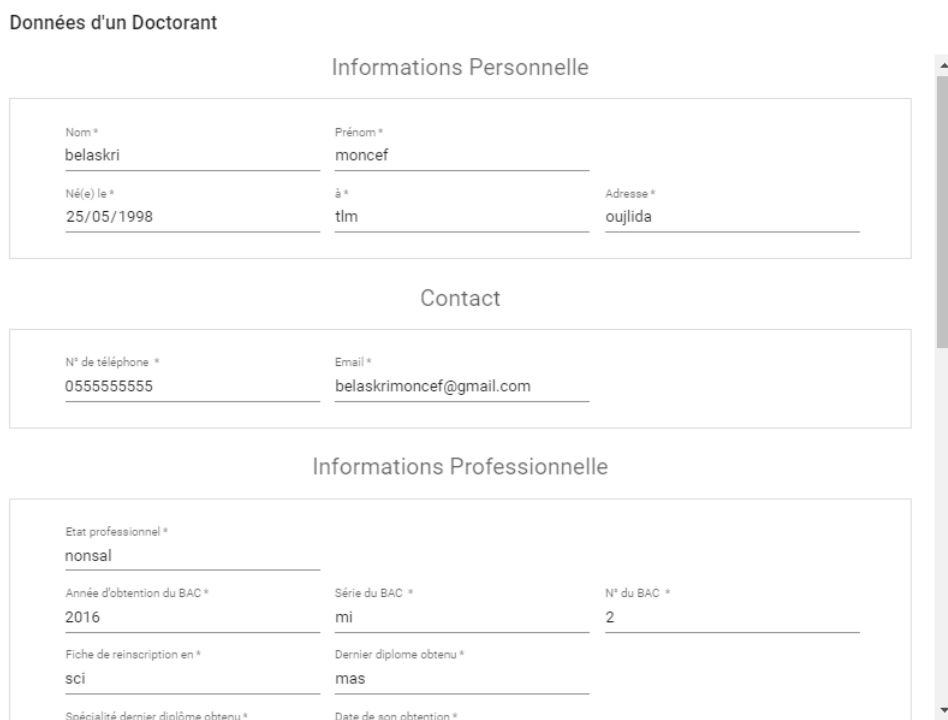


FIGURE 3.31 – Interface pour voir un doctorant

Dans l’interface ci-dessous, l’adjoint pourra modifier l’étape courante du doctorat pour un doctorant choisi. L’étape d’inscriptions est coché par défaut car le doctorant est déjà présent dans notre application. S’il coche l’étape soutenance, sa veut dire que le CSD a validé le dossier de ce doctorant et ce dernier va recevoir un email lui disant qu’il peut préparer sa soutenance. S’il coche l’étape diplôme, le doctorant va

être notifier via sa boîte email que son diplôme est prêt.



FIGURE 3.32 – Interface pour modifier l'étape en cours d'un doctorant

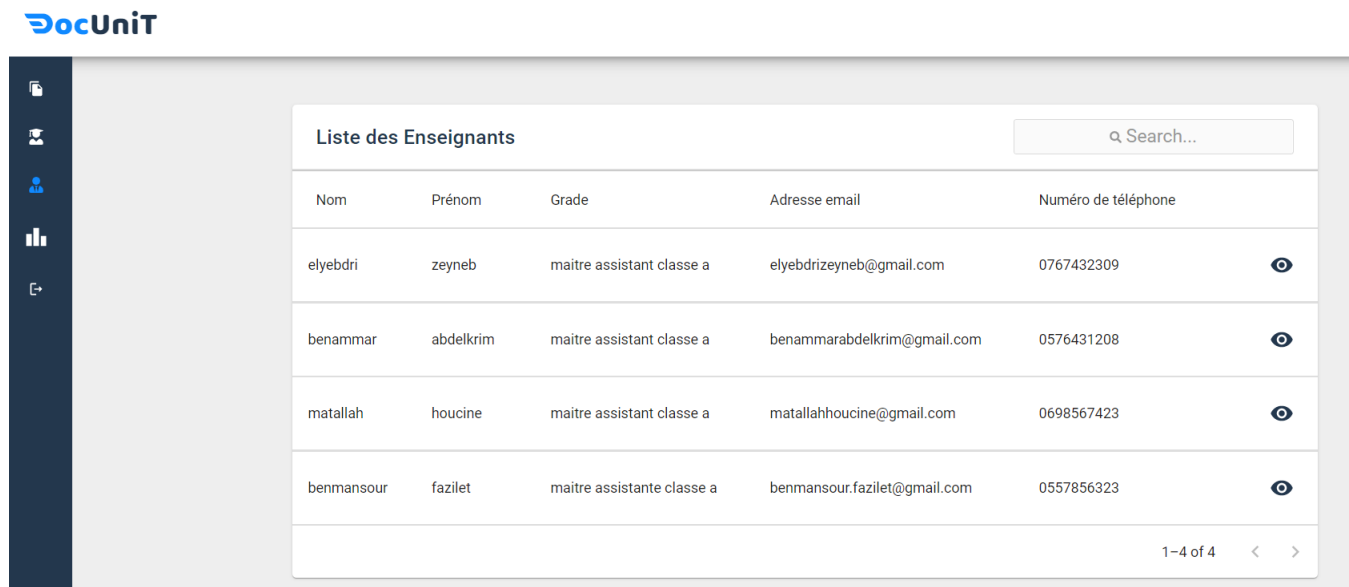


FIGURE 3.33 – Interface pour voir la liste des enseignants

Grâce à cette interface, l'adjoint peut voir la liste de tous les enseignants de son département, consulter leurs informations à l'aide du bouton à droite de la liste qui ouvre l'interface de consultation (voir la figure 3.34). Il peut aussi faire un tri par colonne et effectuer une recherche pour trouver plus facilement un enseignant grâce au bouton de recherche en haut à droite.

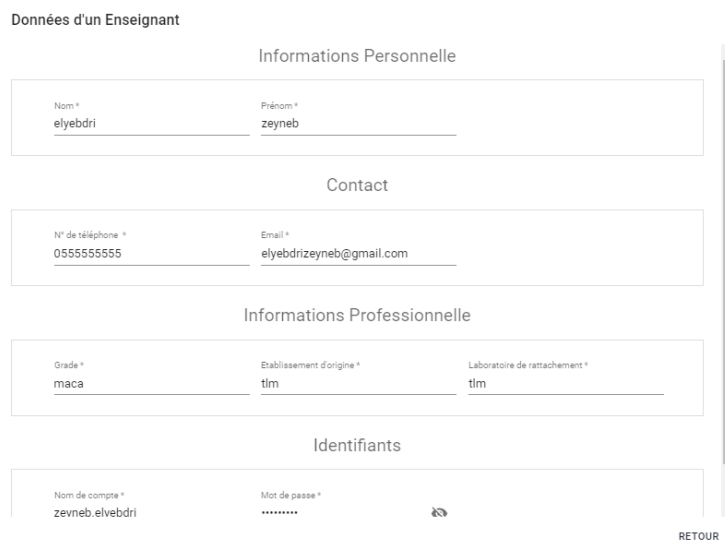


FIGURE 3.34 – Interface pour voir un enseignant

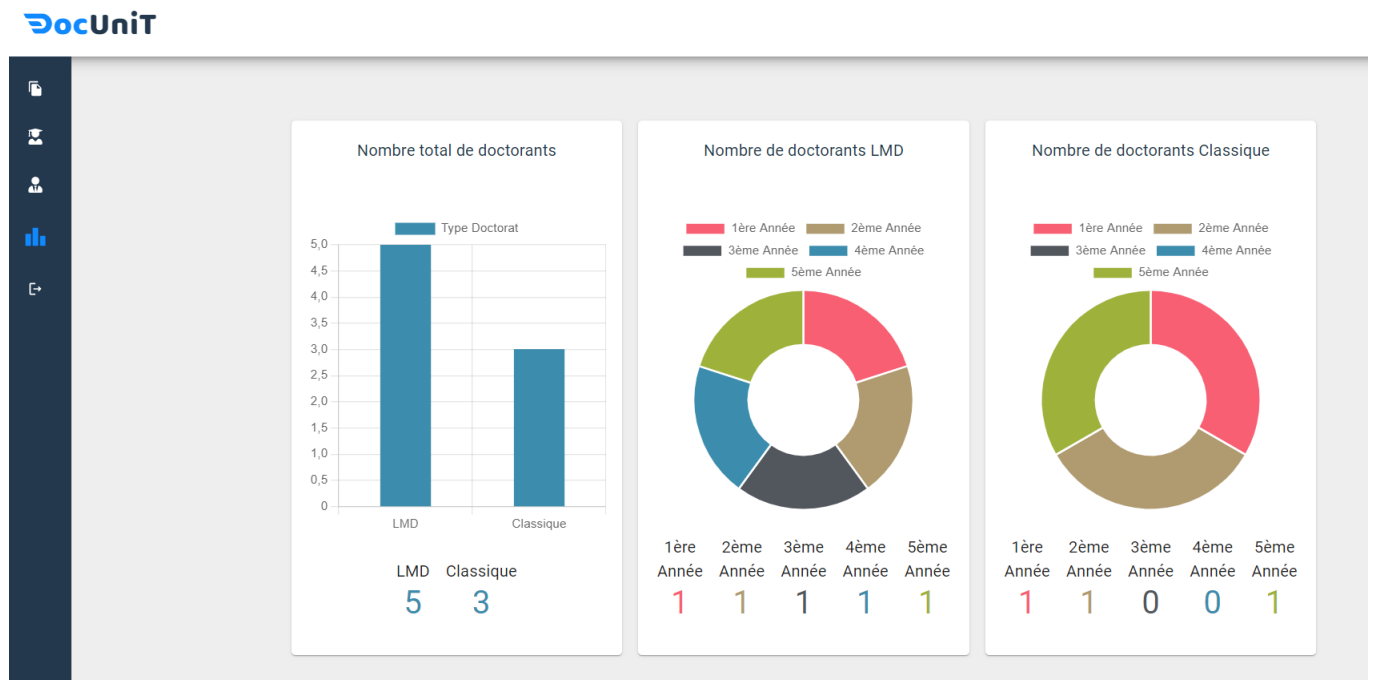


FIGURE 3.35 – Interface pour voir les statistiques du département

L’interface ci-dessus, permet à l’adjoint de consulter les statistiques globales de son département. Nous avons choisi de recenser le nombre total de doctorants pour chaque type de doctorat (lmd ou classique) et pour chaque type on affiche le nombre de doctorant par année.

3.6 Conclusion

Dans ce dernier chapitre, nous avons présenté tous les outils qui nous ont aidé pour mener à bien notre projet, puis on a élaboré les schémas des différentes collections de notre bdd. Enfin, on a exposé toutes les interfaces de notre application web avec leurs fonctionnalités respectives. Pour finir, on va faire une conclusion générale qui englobe toutes les étapes du déroulement de notre travail.

Conclusion Générale

Tout au long de notre mémoire, nous avons mis en avant l'aspect conception et réalisation du projet mais pour y parvenir, nous avons commencé par faire une étude de l'existant dans notre environnement qui est l'université Abou Bekr Belkaid de Tlemcen et plus particulièrement la faculté des sciences.

Nous sommes passés ensuite au cahier des charges où nous avons spécifié tous les besoins fonctionnels et non fonctionnels des utilisateurs de notre application web.

Afin de satisfaire les besoins recensés, nous avons commencé par utiliser UML2 ainsi que la méthode UP pour la conception des diagrammes de cas d'utilisation et de séquence système.

Puis, nous avons créé les collections de notre base de données NoSQL orientée document MongoDB pour pouvoir émettre les requêtes qui vont nous aider à mettre en relation la bdd avec l'application.

Cette dernière baptisée DocUniT est une application web pour la gestion des dossiers des doctorants. Elle a été réalisée grâce aux différents outils cités plus haut et particulièrement l'approche de programmation MERN Stack avec son environnement JavaScript.

Pour conclure, ce projet nous a été très bénéfique et fructueux car il nous a permis d'avoir une meilleure maîtrise dans le domaine de la programmation et de nous initier au domaine du travail. Cependant, il existe des possibilités d'amélioration pour notre travail et surtout au niveau des deux points qui vont suivre : l'amélioration du suivi des deux étapes restantes du processus du doctorat (la soutenance et le diplôme) et le raffinement du système de recherche en introduisant une recherche multicritères.

Bibliographie

- [1] "Présentation université de tlemcen." <https://www.univ-tlemcen.dz/pages/87/presentation>.
- [2] "Hiérarchie université de tlemcen." <https://www.univ-tlemcen.dz/pages/88/bodies-of-uabt>.
- [3] "Présentation de la faculté des sciences de tlemcen." <https://fs.univ-tlemcen.dz/pages/11/presentation-of-the-faculty>.
- [4] "Guide du doctorant université d'oran." https://www.univ-oran1.dz/maison_doctorants/guide.html.
- [5] A. BENSIDHOUM and S. CHERRAK, *Gestion des inscriptions doctorats, Mémoire présenté pour l'obtention du diplôme de Licence en Informatique dans l'université de Tlemcen*. 2013.
- [6] W. L. SAKHI, *GESTION DES INSCRIPTIONS DES DOCTORANTS, Mémoire présenté pour l'obtention du diplôme de Licence en Informatique dans l'université de Tlemcen*. 2015.
- [7] "Comparaison entre application bureau et web." <https://digitalskynet.com/blog/Desktop-App-vs-Web-App-Comparative-Analysis>.
- [8] "Introduction uml." <https://web.maths.unsw.edu.au/~lafaye/CCM/uml/umlintro.htm>.
- [9] "Processus unifié." https://www.academia.edu/39598956/Processus_Unifi%C3%A9_UP_and_2TUP.
- [10] "Tutoriel diagramme de cas d'utilisation." https://atefsd.weebly.com/uploads/5/0/3/6/503639/csi_02_chap02.pdf.
- [11] "Tutoriel diagramme de séquence système." <https://creately.com/blog/fr/uncategorized-fr/tutoriel-sur-le-diagramme-de-sequence/>.
- [12] "Présentation latex." <https://www.latex-project.org/>.
- [13] "Présentation photoshop." <https://www.campussupworld.com/blog/presentation-photoshop>.
- [14] "Présentation github." <https://www.pierre-giraud.com/git-github-apprendre-cours/presentation-git-github/>.
- [15] "Présentation postman." <https://fr.acervolima.com/introduction-a-postman/>.
- [16] "Présentation visual studio code." <https://www.gladir.com/SOFTWARE/VISUALSTUDIOCODE/presentation.htm>.
- [17] "Définition de la pile de logiciel mern stack." <https://www.kicklox.com/blog-talent/mern-stack-definition-enjeux-avantages/>.
- [18] "Schéma explicatif du fonctionnement de la mern stack." <https://www.bocasay.com/fr/comment-fonctionne-mern-stack/>.

- [19] “Comparaison entre mern, mevn, mean.” <https://www.arkasoftwares.com/blog/mean-vs-mern-vs-mevn-vs-lamp-stack-for-development/>.
- [20] “Définition jwt.” <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/json-web-token-jwt/>.
- [21] “Schéma explicatif du fonctionnement du jwt.” <https://www.vaadata.com/blog/fr/jetons-jwt-et-securite-principes-et-cas-dutilisation/>.
- [22] “Principe de la fonction de hashage.” <https://zestedesavoir.com/tutoriels/1895/les-fonctions-de-hachage-cryptographiques/#2-quest-ce-quune-fonction-de-hachage>.

Résumé

Le doctorat est la dernière et la plus élogieuse étape du cycle LMD. De nos jours, au vue de son importance et de sa complexité, l'utilisation d'un simple système de gestion manuelle pour les dossiers des doctorants depuis l'étape d'inscription jusqu'à l'obtention du diplôme ne suffit plus. C'est là que nous intervenons pour créer une application web qui permet de faciliter cette gestion pour toutes les parties, que ce soit l'administration ou le doctorant. Pour ce faire, nous avons utilisé le langage de modélisation UML pour la conception et l'approche MERN Stack qui prend en compte MongoDB pour la gestion de la base de données NoSQL, le framework react pour la création des interfaces et l'environnement Node avec son framework Express pour le coté serveur.

Mot clé : doctorat, inscription, gestion de dossier, application web, UML , MERN Stack.

Abstract

The doctorate is the last and most commendable step in the LMD cycle. Nowadays, in view of its importance and complexity, the use of a simple manual management system for the registration files of doctoral students is no longer sufficient. This is where we intervene to create a web application that allows to facilitate this management for all the parts, for the administration or the doctoral student. We used UML modeling and the MERN Stack approach which takes into account MongoDB for the management of the NoSQL database, the react framework for the creation of interfaces and the Node environment with its Express framework for the server side.

Keyword : doctorate, registration, files management, Web application, UML , MERN Stack.

ملخص

الدكتوراه هي المرحلة الأخيرة والأكثر استحقاقاً في دورة LMD. في الوقت الحاضر ، ونظراً لأهميتها وتعقيدها ، فإن استخدام نظام إدارة يدوي بسيط لملفات طلاب الدكتوراه من مرحلة التسجيل حتى التخرج لم يعد كافياً. لذلك قمنا بإنشاء تطبيق ويب يسهل هذه الإدارة لجميع الأطراف ، سواء كانت الإدارة أو طالب الدكتوراه. للقيام بذلك ، استخدمنا لغة نمذجة UML للتصميم ونهج MERN Stack الذي يأخذ في الاعتبار MongoDB لإدارة قاعدة بيانات NoSQL ، وإطار React لإنشاء الواجهات وبيئة Node مع إطار عمل Express الخاص بها.

الكلمات المفتاحية: دكتوراه، تسجيل، إدارة الملفات، تطبيق الويب، UML، MERN Stack.
