–––

**DEDICACE**

Nous dédions ce travail à toutes celles et ceux qui, par leur présence, leur soutien ou leur amour, ont contribué à rendre ce parcours possible et plus lumineux.

À nos chers parents, qui ont été notre premier pilier et notre plus grande force. Leur patience, leurs sacrifices, leur confiance en nous et leurs encouragements constants nous ont guidées tout au long de notre parcours académique. Que ce modeste travail soit le reflet de notre immense reconnaissance et de notre amour profond envers eux.

À nos frères et nos grands-parents, pour leur affection sincère, leurs prières silencieuses, et leur capacité à nous insuffler énergie et motivation même dans les moments les plus difficiles. Leur présence est un véritable cadeau.

À tous nos proches, amis et membres de la famille, qui nous ont tendu la main à un moment ou un autre, qui nous ont soutenues parfois sans même le savoir et qui ont su partager avec nous aussi bien les moments de doute que les instants de joie.

À nos amis fidèles, pour leur écoute, leur compréhension, leur bonne humeur, et surtout pour avoir toujours cru en nous, même quand nous doutions de nous-mêmes.

À vous tous, qui avez enrichi ce voyage par votre gentillesse, vos mots d’encouragement, et votre chaleur humaine : ce travail vous est aussi dédié. Vous avez rendu ce chemin plus beau, plus fort, et plus riche.

**Merci du fond du cœur.**

**ANAS, ZAKARIA**

***Remerciement***

Au terme de ce projet de fin d'études, qui marque l'aboutissement de notre parcours académique, nous tenons à exprimer nos vifs et sincères remerciements à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce travail.

Nos remerciements les plus profonds s'adressent en premier lieu à notre encadrante pédagogique, **M. CHAFIQ TARIK**. Nous la remercions pour la qualité de son

encadrement, ses précieux conseils, sa disponibilité constante et sa patience tout au long de ce projet. Sa rigueur scientifique et ses orientations pertinentes ont été un guide essentiel pour mener à bien notre mission.

Nous exprimons également notre profonde gratitude à notre co-encadrant, **M. Moussab BATAL**. Son expertise technique, son soutien et sa vision pragmatique nous ont permis de surmonter de nombreux défis et d'enrichir considérablement notre solution.

Nous adressons nos remerciements aux membres du jury pour l'honneur qu'ils nous font en acceptant d'évaluer notre travail et pour l'intérêt qu'ils y porteront.

Nous tenons également à remercier chaleureusement **Mme FAJR ROKIA**, coordinateur de la filière, pour sa disponibilité, sa bienveillance et son accompagnement tout au long de notre formation. Son engagement envers la réussite des étudiants a été pour nous une réelle source de motivation.

Nos remerciements s’étendent à l’ensemble du corps professoral et administratif de l’**école Supérieure Supemir de Casablanca**, pour la qualité de la formation qui nous a été dispensée durant notre cursus.

Enfin, nous tenons à remercier chaleureusement nos familles et nos amis pour leur soutien inconditionnel, leur patience et leurs encouragements durant les moments de doute et de travail intense. Leur confiance a été une source de motivation indispensable.

**Résumé**

**La Plateforme de Gestion Doctorale FSBM** est une solution numérique innovante dédiée à la gestion complète des dossiers de soutenance des doctorants au sein de la Faculté des Sciences Ben M’Sik. Elle vise à digitaliser, centraliser et automatiser l’ensemble des étapes nécessaires à la validation académique d’une thèse, depuis la soumission initiale du dossier par le doctorant jusqu’à la validation finale par les instances administratives et pédagogiques.  
  
Techniquement, elle repose sur **une architecture microservices** conteneurisée avec **Docker Compose,** combinant un frontend moderne développé avec **Next.js 15**, **React 19**, **Tailwind CSS** et **Zustand**, ainsi qu’un backend structuré en services spécialisés, développés en **Node.js** avec **Express**, couvrant les rôles d’authentification, doctorant, enseignant, assistant, administrateur et vice-doyen, le tout connecté à une base de données **MySQL 8.0**.

Le workflow de validation multi-niveaux permet une progression contrôlée entre les états brouillon → soumis → validé → accepté/rejeté, avec des permissions définies selon les rôles (doctorant, assistants, enseignants, vice-doyen). Chaque utilisateur dispose d’un tableau de bord personnalisé.  
  
Cette plateforme permet à l’institution de **gagner en efficacité, diminuer les erreurs humaines, assurer la transparence** du processus et **renforcer la sécurité des données.** Son architecture modulaire garantit une **scalabilité et une maintenabilité optimales,** positionnant cette solution comme un outil stratégique pour la **transformation numérique de la gestion des soutenances doctorales.**

**Table des matières**

**Table de figure**

**Liste des tableaux**

**Introduction Générale**

Selon le bulletin officiel N° 5674 du 16 Chaoual 1429 (16 octobre 2008) du Royaume du Maroc, le cycle doctoral est considéré comme une formation sanctionnée par l’obtention d’un diplôme de doctorat après une soutenance réussie d’une thèse devant des membres du jury désigné. Selon l’article 8 du décret N° 2-04-89, la préparation d’un doctorat au Maroc dure 3 ans, cette durée peut être prorogée d’un an ou de deux ans par le chef de l’établissement sur proposition du directeur du Centre d’Études Doctorales (CeDoc) après l’avis du directeur de thèse. En 2015, la durée de préparation du doctorat a été prolongée d’une troisième et dernière année, qui s’ajoute aux trois années réglementaires et aux deux années exceptionnelles (projet de décret N° 2-15-159 complétant le décret N° 2-04-89).

La formation doctorale est organisée au sein du CEDoc mis en place par les établissements d'enseignement conformément aux articles 19 et 24 de la loi no. 01-00. Le doctorant réalise ses travaux de recherche sous la direction de son encadrant (directeur de thèse) au sein de l’un des laboratoires de recherche fixé par le conseil de l’université. À part les activités de recherche, le doctorant est censé participer aux activités pédagogiques (Surveillance, TP, TD, Tutorat des étudiants…) et scientifiques (Colloques, Conférences, Journées doctorales, Module de formation en méthodologie…). Le volume horaire de la formation complémentaire dispensée durant la préparation du doctorat est de 200 heures.

L’accès au cycle doctoral au Maroc est ouvert aux titulaires d’un Master/Master spécialisé ou d’un diplôme reconnu équivalent qui répond aux critères d’admission prévus dans le descriptif de demande d’accréditation des formations doctorales du CEDoc concerné. Les conditions d'admission sont proposées par une équipe pédagogique du CEDoc, précisées dans la description et adoptées conformément aux dispositions de la loi N° 01.00.

Ce rapport a pour ambition **de présenter l’ensemble des étapes de conception, de développement et de validation de la plateforme.** Le **premier chapitre** exposera le contexte institutionnel, les limitations du système actuel et les objectifs poursuivis. Le **deuxième chapitre** détaillera les choix techniques, la modélisation du système et l’architecture adoptée. Le **troisième chapitre** décrira les étapes de réalisation concrète du projet, tant côté backend que frontend. Enfin, le **quatrième chapitre** présentera les tests menés, les résultats obtenus et une discussion sur les perspectives d’évolution.

**Chapitre 1 : Contexte, Étude de l'existant et objectifs**

**1.1 INTRODUCTION :**

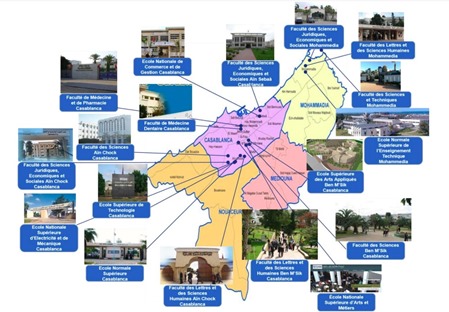
Ce chapitre inaugural a pour vocation de poser les fondations de notre étude en définissant le cadre général dans lequel s'inscrit ce projet. Nous débuterons par une présentation détaillée de notre organisme d'accueil, la Faculté des Sciences Ben M'Sik, afin de contextualiser l'environnement et ses enjeux. Par la suite, nous mènerons une analyse critique et approfondie du système d'information existant, en identifiant ses forces mais surtout ses limites en matière d'accessibilité et de centralisation des données. Cette étude de l'existant sera le socle sur lequel nous construirons une problématique claire et ciblée. En réponse à ce défi identifié, nous conclurons ce chapitre en formulant une série d'objectifs précis, quantifiables et mesurables, qui serviront de fil conducteur pour toute la phase de conception et de développement de notre solution.

|  |  |
| --- | --- |
| **1.2** | **Présentation de l’Université Hassan II :** |

C’est l’une des plus modernes structures d’enseignement supérieur du continent africain et du monde entier, située au sein de la capitale économique et industrielle du pays, "cette université a été toujours partie prenante de la construction de l’avenir qu’offre le pays l’évolution de son environnement régional et international et la dynamique de changement au sein de la société Marocaine"[2].

Après la fusion en 2014 de deux **universités Hassan II** Mohammedia et Casablanca, la vision et les axes de développement sont devenus conçus afin d’augmenter la croissance et l’attractivité de l’université à l’échelle nationale et internationale.

Sa mission est la production et la transmission du savoir à travers ses enseignants chercheurs et ses personnels administratifs et techniques aux étudiants, 18 établissements d’enseignement supérieurs distincts, répartis sur 6 champs situés dans les deux villes Casablanca et Mohammadia, l’**université Hassan II** dispose 8 établissements à accès ouvert et 10 établissements à accès régulé, elle offre une diversité des formations débouchant sur l’obtention des diplômes universitaires dans plusieurs champs disciplinaires.



***Figure 1 : Les 18 établissements de l’Université Hassan II***

|  |  |
| --- | --- |
| **1.3** | **Présentation de la Faculté des Sciences Ben M’Sick : Présentation de l’organisation d’accueil** |



***Figure 2 : La faculté des Sciences Ben M’Sick***

La Faculté des Sciences **Ben M’Scik FSBM** à Casablanca, fait partie de l’**Université Hassan II**. C’est un établissement public d’enseignement supérieur à accès libre dont le but est de développer des programmes d’enseignement et de recherche dans le domaine scientifique, et de préparer les étudiants au mieux à une entrée rapide dans la vie professionnelle et à favoriser leur insertion.

Monsieur **EL BOUARI ABDESLAM**, le doyen de la **FSBM** a assuré que «nous avons offert un espace de dynamisme et de promotion de la culture et de l’art au profit des étudiants en encadrant, développant et dynamisant les clubs créés/existants. Notre objectif est l’implication des étudiants dans la vie universitaire : offres d’emplois ou de stage, partage de compétences et d’expériences et l’amélioration du cadre de vie de l’étudiant au sein de l’établissement.»

Il a ajouté que "La Faculté des Sciences Ben M’Sick, en continuelle mutation dans la Ville de Casablanca depuis sa création en 1984, n’a cessé de contribuer à la diversification des spécialités et des circuits de formation et à la consolidation de la recherche scientifique au profit de ses étudiants, ainsi qu’au développement socio-économique de Casablanca"

L’administration de la faculté est dirigée par le doyen Mr. EL BOUARI ABDESLAM, qui est responsable de la qualité de l’ensemble du processus d’enseignement et de recherche.

La FSBM développe son activité autour de 6 **départements** :

• Département de Biologie.

• Département de Chimie.

• Département de Géologie.

• Département de Mathématiques et informatiques.

• Département de Physique.

• Département de Communication et Humanité.

Elle dispense de nombreuses formations de la licence au doctorat en passant par les licences professionnelles et les masters spécialisés ou de recherche, pour la licence fondamentale la faculté englobe actuellement 6 **parcours** :

• Sciences Mathématiques et Applications (SMA)

• Sciences Mathématiques et Informatiques (SMI)

• Sciences de la matière Chimies (SMC)

• Sciences de la Matière Physique (SMP)

• Sciences de la Terre et de l’Univers (STU)

• Sciences de la Vie (SVI)

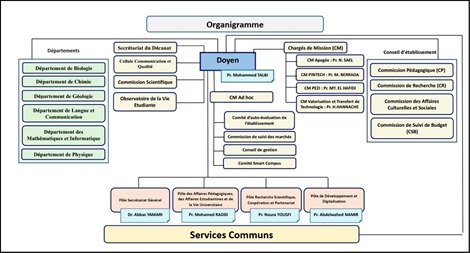
Concernant la licence professionnelle il y a deux **parcours** :

• Sciences et Technologies de Laboratoire Spécialité Biotechnologies (STLSB)

• Méthodes et Procédés d’Analyses (MPA)

Elle englobe notamment 18 **masters** :

***Figure 3 : Les Filières de Master à la FSBM***

**Organigramme de la Faculté des Sciences Ben M’Sick :**

***Figure 3 : Organigramme de la Faculté des Sciences Ben M’Sick***

**Les services communs de la Faculté des Sciences Ben M’Sick :**

1. **Service des ressources humaines :**

Ce service a pour objectif de gérer le personnel enseignant chercheur, gérer le personnel administratif et technique, ainsi que la gestion prévisionnelle de l’emploi et des compétences.

1. **Service des affaires économique :**

Ce service a pour principale mission, l’élaboration du budget et le suivi de son exécution en coordination avec les instances de la faculté, ainsi que la comptabilité, les marchés et les achats, tout en effectuant les paiements.

1. **Service des affaires générale et de planification :**

Le service des affaires générale est chargé de la planification, il assure la gestion du patrimoine mobilier de la faculté conformément aux règles, et faire de la maintenance et l’externalisation, et il gère le parc des véhicules.

1. **Service Informatique :**

Le Service Informatique assure le développement ou la sélection des logiciels selon les objectifs stratégiques et l’élaboration des plans d’informatisation de la faculté. Son rôle est d’établir les réseaux et sécurité, assurer l’organisation le suivi et la mise en œuvre de toutes l’infrastructure système et informatique. Sans oublier le développement et la maintenance des bases de données, la sélection et l’implantation des progiciels, la Transformation digitale. Ainsi que l’édition des diplômes et l’Apogée et Statistiques.

1. **Service des affaires estudiantines et de la vie universitaire :**

La direction des affaires estudiantines et de la vie universitaire offre une multitude des services au profit des étudiants. Elle favorise leur intégration et les accompagne dans les différentes étapes de leur parcours académique, depuis l’inscription jusqu’à l’insertion. Les missions du service sont, la formation initiale, la vérification et authentification des diplômes, et le suivi Apogée, ainsi que la formation continue.

1. **Service bibliothèque :**

Le service bibliothèque est accessible à tous les étudiants de la faculté, la bibliothèque est un lieu de travail, de recherche, de consultation et de prêt d’ouvrages, de thèses, ou de ressources électroniques qui sont en adéquation avec les formations universitaires dispensées, ...

1. **Service Recherche et Doctorat :**

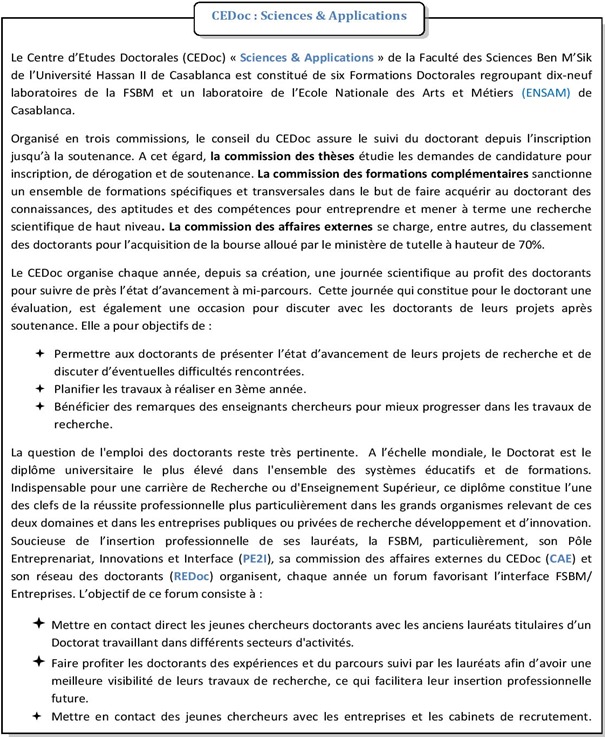
Le conseil est l’organe directeur de la Faculté. Ses activités sont coordonnées par le doyen et le président du conseil Monsieur Mohammed TALBI, ce conseil dé- finit la stratégie de développement, approuve les programmes, indique le nombre d’étudiants et les formes d’enseignement pour chaque spécialité, approuve-les disciplines pour les examens de licence, de master et de doctorat, valide les chefs des départements. Il est composé des représentants des départements, des élus et des étudiants.

Le rôle de ces représentants est d’assister régulièrement aux réunions tenues par le Conseil de la Faculté et de communiquer votre point de vue aux membres du Conseil. Cela est pris au sérieux par la faculté, car ils ne constituent pas moins d’un quart du nombre total des membres du conseil de la faculté, donc, aucune décision ne peut être prise sans les consulter.

Les étudiants élisent 3 représentants pour les représenter au sein du conseil (un pour le premier cycle, un pour le deuxième cycle et un pour le troisième cycle). Les décisions du conseil sont prises à la majorité des voix des membres présents.

|  |  |
| --- | --- |
| **1.4** | **Présentation du Centre d’Etude Doctoral de la FSBM :** |

**1.4.1 Présentation du CeDoc de la FSBM :**

* D’après le site web de la FSBM, le CeDoc de l’établissement consiste de :

***Figure 4 : Service CEDOC au seine de la faculté Ben M’Sick***

**1.4.** **Les structures de recherche de la FSBM :**

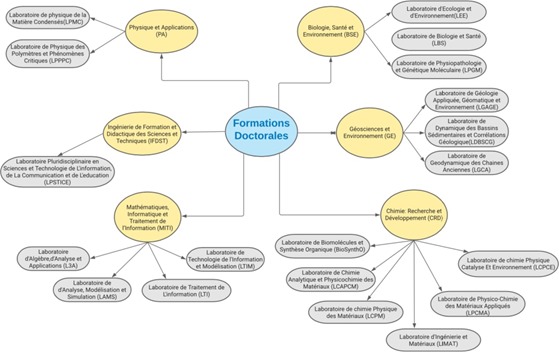
En ce qui concerne les structures de recherche, Il y a deux centres thématiques de recherche :

• Centre De Recherche Matériaux & Énergie (CRME)

• Centre De Recherche Santé & Biotechnologies (CRSB)

Avec un observatoire de recherche en didactique et pédagogie universitaire ORDIPU. Qui sert à promouvoir le développement des systèmes d’enseignement supérieur, développer des formations et une recherche dans le domaine de l’enseignement des sciences ainsi qu’organiser les colloques, tables rondes, Conférences... Alors l’observatoire représente une veille pédagogique non pas pour la faculté mais pour toute l’université.

Une plateforme Innovation et Technologies PInTech aide à assurer une re- cherche scientifique et une formation multidisciplinaire, élaborer des stratégies globales pour une gestion intégrée et optimale du matériel scientifique, et développer un partenariat entre l’université Hassan II de Casablanca et son environnement socio-économique.

En dernier lieu, la FSBM dispose 6 formations doctorales, avec 19 Laboratoires

***Figure 4 : Organigramme de la Faculté des Sciences Ben M’Sick***

Mr EL BOUARI ABDESLAM a poursuivi "Afin de renforcer la coopération, la Faculté des Sciences Ben M’Sick a noué des relations de coopération internationale à travers la signature de conventions et accords avec des universités, des institutions universitaires, des centres et des laboratoires de recherche au Maroc et à l’étranger. Elle est aussi engagée dans divers projets de coopération universitaire internationale"

|  |  |
| --- | --- |
| **1.5** | **ÉTUDE DE L'EXISTANT :** |

Avant de concevoir une solution numérique dédiée à la gestion des soutenances doctorales, il est essentiel d’analyser le système actuellement en place à la Faculté des Sciences Ben M’Sick. Cette étude permet de comprendre les limites du dispositif existant, d’identifier les points de friction, et de cibler les besoins réels des différents acteurs impliqués dans le processus doctoral.

Actuellement, la gestion des dossiers de soutenance repose encore largement sur des procédures manuelles, fragmentées et peu numérisées. Les doctorants, les encadrants, les assistants administratifs ainsi que les responsables pédagogiques doivent composer avec une multiplicité de supports et de canaux de communication :

* **Échanges par email :** les demandes de dépôt, les envois de documents ou les réponses aux requêtes sont souvent traités individuellement par email, ce qui entraîne une perte de temps considérable et une traçabilité limitée.
* **Documents papier :** les formulaires de soutenance, les autorisations, les CV, et autres pièces justificatives sont souvent imprimés, signés manuellement et remis physiquement, ce qui complique l’archivage et la consultation.
* **Systèmes bureautiques isolés :** les informations sont conservées dans des fichiers Excel, des dossiers locaux ou partagés entre services, sans réelle interconnexion ni base de données centralisée.
* **Communication informelle ou non standardisée :** certaines décisions ou étapes du processus sont communiquées verbalement ou par notes internes, sans enregistrement systématique, ce qui nuit à la cohérence du suivi.

Ce fonctionnement présente plusieurs limitations majeures :

* **Manque de centralisation :** aucune plateforme unique ne regroupe toutes les données, documents et étapes du processus de soutenance. Cela rend la consultation fastidieuse et le suivi difficile pour les doctorants comme pour les assistants.
* **Traçabilité faible :** il est complexe de connaître l’état d’avancement d’un dossier ou de retracer les actions et validations effectuées, surtout lorsqu’elles sont réparties entre différents acteurs ou non consignées.
* **Absence d’automatisation :** le traitement manuel des documents et des validations multiplie les erreurs potentielles, allonge les délais, et mobilise inutilement les ressources humaines.
* **Sécurité limitée :** le stockage non centralisé, parfois non sécurisé, expose les données à des risques de perte ou de fuite.
* **Charge administrative élevée :** le personnel est fortement sollicité pour des tâches répétitives ou à faible valeur ajoutée (relances, suivis de dossiers, vérifications documentaires), au détriment d’un encadrement de qualité.

Cette analyse met en lumière la nécessité d’une transformation numérique du processus de soutenance doctorale. Une plateforme web centralisée, sécurisée et accessible permettrait non seulement d’alléger les charges administratives, mais aussi d’améliorer la transparence, la réactivité et la qualité du service rendu à la communauté doctorale.

|  |  |
| --- | --- |
| **1.5** | **PRESENTATION DU PROJET :** |

**1.5.1 PROBLEMATIQUE :**

La gestion traditionnelle des soutenances doctorales au sein de la Faculté des Sciences Ben M’Sick repose encore en grande partie sur des processus manuels, papier et non centralisés, ce qui engendre de nombreuses limitations opérationnelles et difficultés d’organisation. Chaque étape, depuis la soumission des documents jusqu’à la validation finale, implique une multitude d’acteurs (doctorants, assistants, enseignants, administration) et s’effectue à travers des canaux dispersés, souvent peu adaptés aux exigences actuelles de rapidité, de fiabilité et de traçabilité.

Dans ce contexte, les doctorants rencontrent des obstacles fréquents : manque de visibilité sur l’état de leur dossier, retards liés à des validations successives non automatisées, besoin de déposer plusieurs fois les mêmes documents auprès d’acteurs différents. De leur côté, les assistants et responsables administratifs sont soumis à une charge de travail répétitive et chronophage, centrée sur des tâches à faible valeur ajoutée (vérifications manuelles, relances, classement papier), nuisant à la fluidité globale du processus.

La problématique centrale de ce projet est donc la suivante :

**Comment concevoir une plateforme numérique centralisée et sécurisée, capable de gérer efficacement l’ensemble du processus de soutenance doctorale, tout en assurant une traçabilité, une collaboration fluide entre les acteurs et une amélioration mesurable de la qualité du service ?**

Cette problématique soulève plusieurs enjeux critiques :

**Centralisation :** tous les documents, actions et validations doivent être regroupés au sein d’un espace unique et cohérent.

**Automatisation :** la gestion des transitions d’état (soumis, validé, accepté, rejeté) doit être fluide et encadrée par des règles claires.

**Suivi en temps réel :** chaque utilisateur doit pouvoir visualiser instantanément l’état d’avancement du dossier.

**Sécurité et contrôle d’accès :** les données doivent être protégées, avec des rôles clairement définis pour chaque intervenant.

**Expérience utilisateur :** l’interface doit être intuitive et adaptée aux profils variés (doctorants, assistants, enseignants, vice-doyen).

En résumé, cette problématique met en évidence la nécessité d’un système de gestion intelligent, modulaire et collaboratif, capable de numériser et d’optimiser le traitement des dossiers de soutenance, tout en respectant les contraintes administratives et pédagogiques propres au cadre doctoral.

**1.5.2 DEFINITION DES OBJECTIFS DU PROJET**

Face à la problématique identifiée, notre projet a pour ambition de développer une **plateforme web dédiée à la gestion complète, centralisée et sécurisée des soutenances doctorales**. Cette solution vise à répondre aux besoins spécifiques des doctorants, des assistants administratifs, des encadrants et de la direction académique. Pour encadrer cette démarche, **quatre objectifs principaux** ont été définis :

* **Objectif 1 : Centraliser l’ensemble du processus de soutenance dans un système unique**

L’objectif est de **regrouper toutes les étapes, documents et acteurs** au sein d’une même plateforme, depuis la soumission du dossier par le doctorant jusqu’à la validation finale par le vice-doyen. Cette centralisation permet de **réduire la dispersion des informations** et d’assurer une **meilleure visibilité et coordination** entre les services.

* **Objectif 2 : Automatiser le workflow de validation multi-niveaux**

La plateforme devra intégrer un **système de validation par étapes**, correspondant aux rôles attribués aux assistants (validation des dates, des documents, des checklists) et à la direction (acceptation finale). Cette automatisation garantira la **fluidité du processus**, la **réduction des délais de traitement**, et l’élimination des tâches répétitives à faible valeur ajoutée.

* **Objectif 3 : Assurer un accès sécurisé, personnalisé et en temps réel**

Chaque utilisateur (doctorant, assistant, enseignant, vice-doyen) disposera d’un **compte avec un tableau de bord personnalisé**, lui permettant d’accéder aux fonctionnalités correspondant à son rôle. L’accès sera protégé par une **authentification sécurisée** (JWT), et les informations seront consultables **24h/24** en toute confidentialité.

* **Objectif 4 : Faciliter le suivi, la collaboration et la traçabilité.**

La plateforme offrira un **suivi en temps réel de l’état d’avancement des dossiers**, des **notifications automatiques** à chaque changement d’état, ainsi qu’un **historique complet des actions** effectuées. Elle favorisera la **collaboration entre les différents acteurs** du processus doctoral et renforcera la **transparence des décisions**.

La mise en œuvre de ces objectifs permettra à la Faculté des Sciences Ben M’Sick de **répondre aux défis organisationnels actuels**, tout en s’inscrivant dans une **démarche de digitalisation durable** au service de l’excellence académique

|  |  |
| --- | --- |
| [**1.6**](https://profsamad.com/boite/resume-la-boite-chapitre-1/) | [**CONCLUSION**](https://profsamad.com/boite/resume-la-boite-chapitre-1/) |

Ce chapitre a permis de situer le **contexte institutionnel** et de mettre en évidence les **limites du système** actuel de gestion des soutenances doctorales à la Faculté des Sciences Ben M’Sick : procédures manuelles, absence de centralisation, manque de traçabilité et surcharge administrative.

Face à ces constats, notre projet propose le développement d’une **plateforme web** centralisée et sécurisée, intégrant un **workflow de validation automatisé**, une **interface adaptée à chaque utilisateur** et un suivi en **temps réel des dossiers**. Cette solution vise à **optimiser le processus de soutenance**, tout en renforçant l’efficacité et la transparence des échanges.

Le prochain chapitre présentera la phase **de conception et les choix techniques** qui ont structuré notre approche.

**Chapitre 2 : Conception et Analyse Détaillée**

|  |  |
| --- | --- |
| **2.1** | **INTRODUCTION :** |

Ce deuxième chapitre est consacré à la **phase de conception**, qui constitue un jalon essentiel du projet. C’est à ce stade que les **besoins fonctionnels et techniques identifiés** dans le chapitre précédent sont traduits en **une architecture logicielle claire** et en un plan de **développement structuré**.

Nous commencerons par définir **les spécifications fonctionnelles et non fonctionnelles** de la plateforme, afin d'établir **un cahier des charges précis**. Nous poursuivrons avec une **modélisation UML** (cas d’utilisation, diagrammes de séquence et de classes) pour visualiser les interactions, les rôles utilisateurs et la logique du système.

La méthodologie de gestion de projet adoptée est la **méthode agile Scrum**, choisie pour sa flexibilité et son approche itérative. Le développement a été organisé en **sprints** avec des revues régulières pour ajuster les priorités. L’outil **Jira** a été utilisé pour planifier les tâches, suivre l’avancement du projet et coordonner les différentes phases de réalisation.

Le cœur de ce chapitre détaillera ensuite les **choix techniques majeurs** : adoption d’une **architecture microservices**, utilisation de **Docker Compose**, développement backend en **Node.js avec Express**, et frontend en **Next.js 15** avec **React 19,** **Tailwind CSS** et **Zustand**. Nous expliquerons également la **gestion des rôles, du workflow de validation** et de la **sécurisation des accès**.

Ce chapitre constitue donc le **plan directeur** de la plateforme, sur lequel repose la mise en œuvre technique et la réussite du projet dans sa globalité.

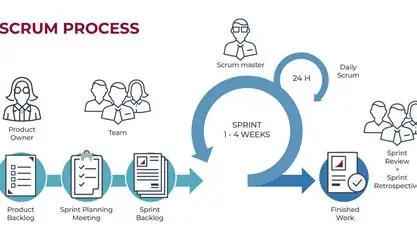
|  |  |
| --- | --- |
| **2.2** | **METHODOLOGIE ET PLANIFICATION DU PROJET** |

La réussite d’un projet informatique, même dans un cadre académique, repose sur une **méthodologie de travail rigoureuse** et une planification adaptée aux contraintes réelles du développement. Cette section présente **l’approche de gestion de projet** que nous avons adoptée pour mener à bien le développement de notre **plateforme de gestion des soutenances doctorales**, depuis l’analyse initiale jusqu’à la mise en œuvre technique.

**2.2.1 MODELE EN CASCADE :**

La **méthode agile Scrum** a été adoptée comme cadre de gestion de notre projet. Cette approche itérative repose sur des **cycles de développement centrés sur des User Stories**, permettant de livrer progressivement des fonctionnalités clés, tout en ajustant les priorités en fonction des retours et des contraintes techniques.

Ce mode de gestion s’est avéré particulièrement adapté à notre projet, car il favorise la **flexibilité**, la **collaboration** entre les membres de l’équipe et une **répartition claire des responsabilités.** Chaque User Story a suivi un cycle de planification, développement, revue et validation.



*Figure 5 : Scrum processus*

**L’outil Jira a été utilisé pour **organiser les User Stories**, suivre leur avancement et documenter les décisions prises au fil du projet. Il a facilité la coordination globale et permis de garder une vision claire de la progression du développement.

*Figure 6 : Outil Jira*

Notre projet s'est articulé autour des six phases suivantes :

**Phase 1 : Analyse du besoin et du contexte**

Étude des problématiques liées à la gestion des soutenances doctorales à la FSBM, et recueil des attentes des utilisateurs (doctorants, assistants, administration).

**Phase 2 : Définition des spécifications et modélisation UML**

Rédaction des spécifications fonctionnelles et non fonctionnelles, élaboration des modèles de cas d’utilisation, de séquence et de classes.

**Phase 3 : Conception de l’architecture technique**

Mise en place de l’architecture microservices, structuration des conteneurs Docker, choix des technologies, et configuration de la base de données MySQL.

**Phase 4 : Développement des fonctionnalités principales**

Implémentation des services backend en **Node.js (Express)**, développement du frontend en **Next.js et React**, intégration de l’interface utilisateur, du système d’authentification et de la gestion des rôles.

**Phase 5 : Tests, validation et finalisation**

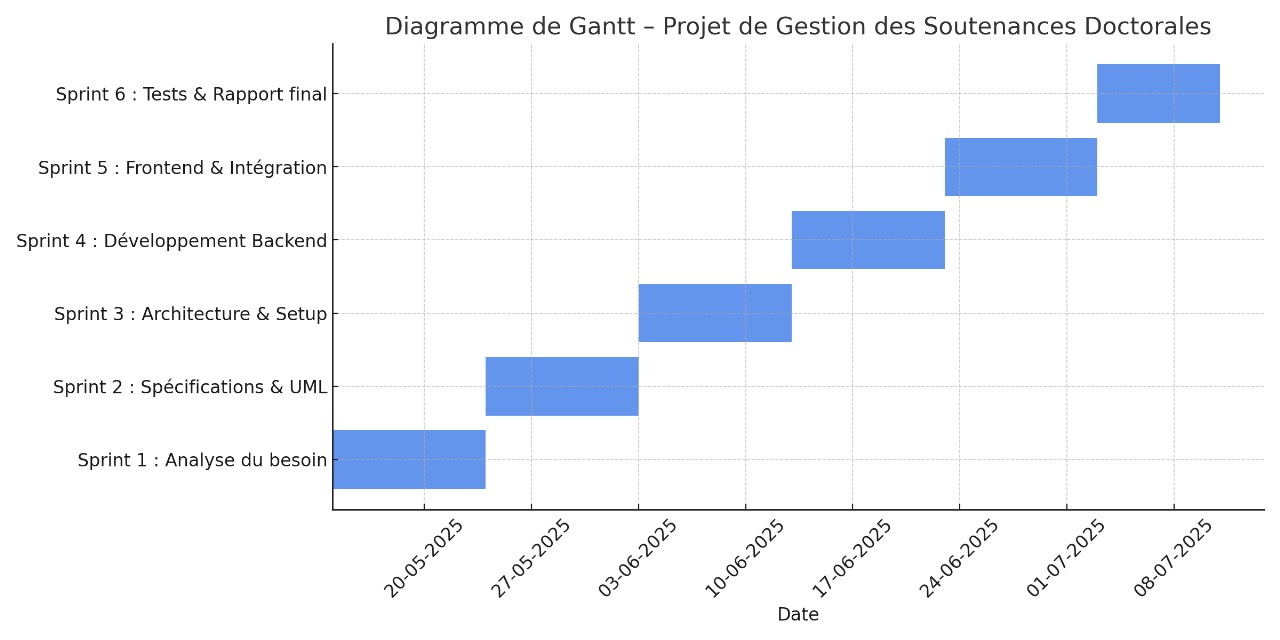
Réalisation de tests fonctionnels, correction des anomalies, collecte de feedback et finalisation des documents du projet.

**2.2.2 DIAGRAMME DE GANTT**

**Le diagramme de Gantt :** est un outil visuel de gestion de projet qui montre les tâches sur une échelle de temps, illustrant leur durée et séquence(Daubier & Daubier, 2022). Chaque

tâche est représentée par une barre horizontale, facilitant la planification et le suivi des délais.

Il permet de visualiser les dépendances et les chevauchements entre les tâches.



*Figure 7 : Diagramme de gantt*

Le diagramme de Gantt est un outil de gestion de projet permettant de visualiser les tâches dans le temps, leur durée, et leurs enchaînements. Il facilite la planification, le suivi de l’avancement et la coordination entre les différentes étapes du projet (Daubier & Daubier, 2022).

Dans notre projet, le diagramme de Gantt a été utilisé pour répartir les sprints sur plusieurs semaines et assurer une progression fluide entre :

* L’étude du besoin et l’analyse du contexte
* La rédaction des spécifications
* La conception et l’architecture technique
* L’implémentation du frontend et du backend
* La phase de tests et de validation finale

Cette planification structurée a permis de maîtriser les délais, d’anticiper les blocages, et d’assurer une réalisation progressive et contrôlée de la plateforme de gestion des soutenances doctorales.

|  |  |
| --- | --- |
| **2.3** | **SPECIFICATIONS FONCTIONNELLES ET NON- FONCTIONNELLES** |

Avant d’entamer le développement de la plateforme, il est indispensable de définir de manière claire et structurée les **exigences du système**. Celles-ci se répartissent en deux catégories complémentaires : **les spécifications fonctionnelles**, qui décrivent les fonctionnalités attendues, et les **spécifications non-fonctionnelles**, qui définissent les contraintes de qualité, de performance et de sécurité.

 **Les spécifications fonctionnelles :**

**Soumission de dossier par le doctorant :** Le doctorant doit pouvoir créer, modifier et soumettre un dossier de soutenance via une interface web dédiée.

**Upload sécurisé de documents:** Le système doit permettre l’import de fichiers (PDF, DOCX…) avec typage des documents requis (demande d’autorisation, CV, rapport, etc.) et vérification de leur format et taille.

**Workflow de validation multi-niveaux :** Le dossier suit un cycle structuré de validation impliquant plusieurs rôles : assistants (3 niveaux), enseignant encadrant, vice-doyen. Chaque acteur doit valider ou rejeter une étape.

**Gestion des rôles et des permissions :** Chaque utilisateur (doctorant, assistant, enseignant, administrateur, vice-doyen) a accès à un tableau de bord spécifique, avec des droits définis selon son rôle.

**Suivi de l’état du dossier :** Le système doit afficher l’état d’avancement du dossier (brouillon, soumis, validé, rejeté...) et son historique de traitement.

**Notifications automatiques :** À chaque changement d’état, les utilisateurs concernés doivent recevoir une notification (par interface ou email).

**Consultation et lecture seule pour les encadrants :** Les enseignants encadrants peuvent consulter les dossiers de leurs doctorants, sans possibilité de modification.

**Administration du système :** L’administrateur peut consulter les statistiques, corriger les erreurs, effectuer des retours arrière (rollback) et gérer les utilisateurs.

**1. Les spécifications non-fonctionnelles :**

**Performance / Temps de Réponse :** Les opérations critiques (soumission, validation, téléchargement) doivent s’exécuter en moins de 5 secondes dans des conditions normales d’utilisation.

**Ergonomie et Utilisabilité :** L’interface doit être responsive, simple d’utilisation, et compatible avec les navigateurs courants.

**Confidentialité et sécurité :** L’accès au système est protégé par authentification JWT. Les données sont transmises de façon sécurisée, et seules les personnes autorisées peuvent accéder aux dossiers.

**Traçabilité :** Un journal d’audit (logs) doit être maintenu pour enregistrer toutes les actions importantes (soumission, validation, suppression...).

**Disponibilité :** La plateforme doit être accessible 24h/24 et 7j/7, sauf en cas de maintenance programmée.

**Maintenabilité :** Le code source doit être modulaire, bien commenté, et suivre les bonnes pratiques du développement (notamment avec Node.js et Express côté backend, Next.js côté frontend), afin de faciliter les évolutions futures (ajout de nouveaux rôles, formats de fichiers, ou modules).

**2.4 ANALYSE ET CONCEPTION ORIENTEE OBJET (UML)**

L'approche Orientée Objet nous permet de modéliser notre système comme un ensemble d'objets en interaction, ce qui est particulièrement adapté à la structure de notre application. Pour représenter les aspects statiques et dynamiques de notre assistant conversationnel, nous utiliserons trois diagrammes UML clés : le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classes.

**2.4.1 DIAGRAMME DE CAS D’UTILISATION**

**Le diagramme de cas d'utilisation** : est un outil de modélisation visuelle utilisé en ingénierie logicielle pour décrire les interactions entre les utilisateurs et un système. Il représente les différentes actions que les utilisateurs peuvent effectuer et comment ces actions sont réalisées à travers les fonctionnalités du système. C'est une méthode efficace pour visualiser les exigences fonctionnelles d'un système logiciel.

**2.4.2 IDENTIFICATION DES ACTEURS**

| Acteur | Description |
| --- | --- |
| Doctorant | Utilisateur principal qui gère son dossier de thèse et ses activités. |
| Assistant 1 | Personnel ayant le droit de rejeter les dossiers. |
| Assistant 2 | Responsable du suivi des rapporteurs et des doctorants. |
| Assistant 3 | Personnel chargé de valider les documents d’inscription. |
| Enseignat | Encadrant du doctorant, impliqué dans la validation des dossiers. |
| Vice-Doyen | Responsable de l’acceptation finale des dossiers. |
| Doyen | Supervise les statistiques, la gestion des utilisateurs. |
| Admin | Administrateur ayant accès global à tous les dossiers. |

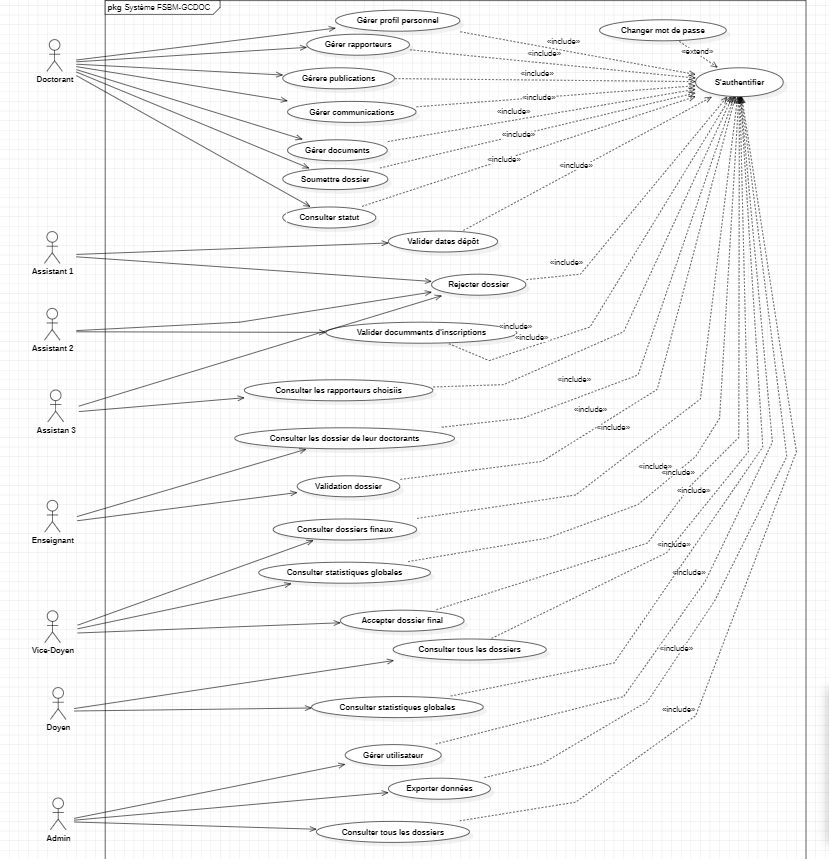
# Tableau 1:Identification des acteurs

**2.4.3 IDENTIFICATION DES CAS D’UTILISATION**

Voici les principales interactions associées à chaque acteur. Ces cas d'utilisation découlent directement des spécifications fonctionnelles (SF) définies précédemment.

| Acteur | Cas d’utilisation |
| --- | --- |
| Doctorant | * Gérer profile * Soumettre dossier * Consulter statut |
| Assistant 1 | * Valider dossier * Rejeter dossier |
| Assistant 2 | * Valider documents d’inscriptions |
| Assistant 3 | * Consulter les rapporteurs choisis |
| Enseignant | * Validation dossier * Suivre statut des dossiers |
| Vice-Doyen | * Consulter les dossiers * Accepter dossier final |
| Doyen | * Consulter statistiques globales |
| Admin | * Consulter les dossiers * Gérer les utilisateurs * Exporter données |

# Tableau 2:Identification des Cas d'Utilisation

 **2.4.4 REPRESENTATION VISUELLE DU DIAGRAME DE CAS D’UTILISATION:**

*Figure 4:Diagramme de cas d'utilisation*

**2.5 DIAGRAMME DE SEQUENCE :**

Ce diagramme décrit la chronologie des interactions entre les objets (ou composants) du système pour réaliser un cas d'utilisation spécifique. Nous allons illustrer le déroulement d'une requête de chat, depuis la saisie de l'utilisateur jusqu'à l'affichage de la réponse.

**2.5.1 IDENTIFICTION DES PARTICIPANTS**

Les principaux composants qui interagissent dans notre scénario **soumission de dossier** sont :

|  |  |
| --- | --- |
| Acteur | Description |
| Doctorant | La personne qui interagit avec le système. |
| InterfaceWeb (Frontend) | Application web utilisée par le doctorant  (TailwindCss/Next.js ). |
| Service Doctorant (Backend) | Couche métier qui gère la logique fonctionnelle. |
| Base de données | Stocke les informations du dossier. |
| Systéme de fichiers | Conserve les fichiers physiques soumis |

# Tableau 2:les composants du systeme

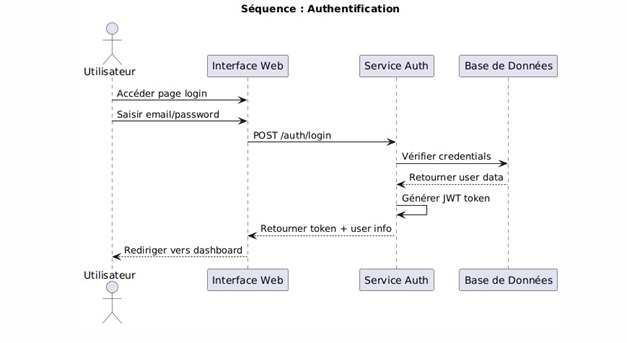
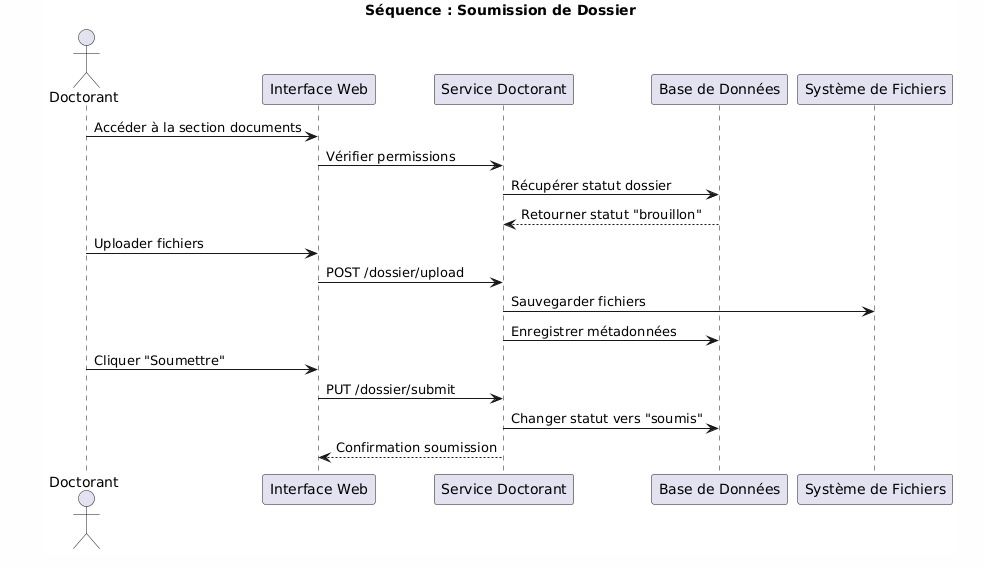
Les principaux composants qui interagissent dans notre scénario **soumission de dossier** sont :

|  |  |
| --- | --- |
| Acteur | Description |
| Utilisateur | Personne qui interagit avec l'application en saisissant ses identifiants pour se connecter. |
| InterfaceWeb (Frontend) | Interface front-end qui permet à l'utilisateur de saisir ses informations et affiche les résultats. |
| Service Auth  (Backend) | Service responsable de vérifier les identifiants, générer un token JWT et retourner les infos utilisateur.. |
| Base de données | Composant backend qui stocke les données d'identification de l'utilisateur utilisées pour l'authentification. |

# Tableau 3 : les composants du systeme

**2.5.2 CHRONOLOGIE DES MESSAGES :**

* **Pour Soumission de dossier** :
* Le **doctorant** accède à l'espace où il peut gérer ses fichiers.
* L’interface web interroge le **Service Doctorant** pour vérifier les droits de soumission de l’utilisateur, lequel consulte alors la **Base de Données** pour récupérer le statut du dossier, retourné ensuite comme étant « brouillon ».
* Le doctorant téléverse ses documents via **l’interface web**, ce qui déclenche une requête **POST** /dossier/upload envoyée au **Service Doctorant**, lequel sauvegarde les fichiers dans le **Système de Fichiers** et enregistre les métadonnées correspondantes dans la **Base de Données**.
* Lorsque le **doctorant** clique sur le bouton "Soumettre", l’interface envoie une requête **PUT** /dossier/submit au **Service Doctorant**, qui met alors à jour le statut du dossier à « soumis » dans la **Base de Données**.
* Le **doctorant** reçoit une confirmation que le dossier a été bien soumis.
* **Pour Authentification** :
* L’utilisateur ouvre la page de connexion.
* Il entre son email et son mot de passe.
* L’interface envoie ces informations au service d’authentification.
* Le service vérifie les identifiants dans la base de données.
* La base de données renvoie les informations de l’utilisateur.
* Le service crée un jeton d’authentification (JWT).
* Le service renvoie le jeton et les infos de l’utilisateur à l’interface.
* L’utilisateur est redirigé vers le tableau de bord.

 **2.5.3 REPRESENTATION VISUELLE DU DIAGRAMME DE SEQUENCE :**

*Figure 9 : Diagramme de séquence soumission de dossier* *Figure 9 : Diagramme de séquence d’authentification*

**2.6 DIAGRAMME DE CLASSE :**

Ce diagramme présente la structure statique du système, en particulier les principales classes qui composent notre backend. Il met en évidence les responsabilités de chaque classe et comment elles sont interconnectées, formant ainsi le blueprint de notre application.

**2.6.1 IDENTIFICATION DES CLASSES PRINCIPALES**

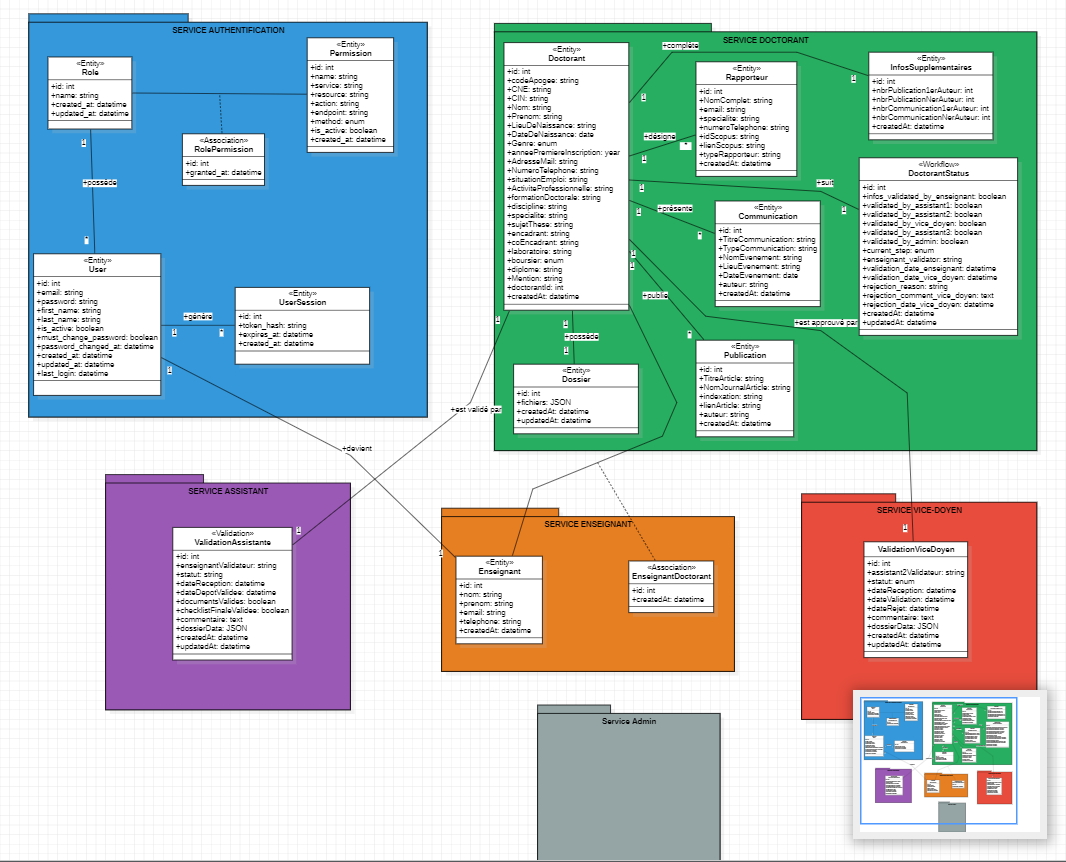
En analysant notre architecture, nous identifions les classes et entités logiques suivantes :

| *Service* | *classe* |
| --- | --- |
| SERVICE AUTHENTIFICATION | * Role * Permission * RolePermission * User * userSession |
| SERVICE DOCTORANT | * Doctorant * Rapporteur * Communication * Publication * Dossier * DoctorantStatut * InfosSuppl |
| SERVICE ENSEIGNANT | * Enseignant * EnseignantDoctorant |
| SERVICE ASSISTANT | * ValidationAssistant |
| SERVICE VICE-DOYEN | * ValidationVice-Doyen |
| Admin | L’Admin ne possède pas de classe dédiée car il utilise les fonctionnalités des autres services existants |
| Doyen | Doyen ne possède pas de classe dédiée car il utilise les fonctionnalités des autres services existants |

# Tableau 4 : Identification des classes principales

**2.6.2 ATTRIBUTS, METHODES ET RELATIONS :**

* Utilisateur est lié à un Rôle et à une Authentification.
* Rôle est associé à plusieurs Permissions.
* DossierDoctorant est lié à :
  + un InfosDoctorant
  + plusieurs Fichiers
  + plusieurs Publications
  + plusieurs Communications
  + plusieurs Rapporteurs
  + une InformationsSuppl
* Fichier, Publication, Communication sont tous rattachés à un DossierDoctorant
* Les services (Assistant, Enseignant, Vice-Doyen) interagissent avec les DossiersDoctorants selon leurs responsabilités.

** **2.6.3 REPRESENTATION VISUELLE DE DIAGRAMME DE CLASSE**

*Figure 6:Diagramme de classe*

|  |
| --- |
| **2.7 CHOIX DE L'ARCHITECTURE TECHNIQUE : LE MODELE RAG (RETRIEVAL-AUGMENTED**  **GENERATION)** |

Le projet repose sur une architecture microservices découplée, combinée à une séparation stricte entre le frontend et le backend. Ce choix a été motivé par les besoins de modularité, de scalabilité et de maintenabilité d’un système de gestion complexe, comme celui d’une plateforme doctorale.

Chaque fonctionnalité principale (authentification, gestion doctorants, administration, validation, etc.) est isolée dans un microservice indépendant. Ces microservices sont conteneurisés via Docker et orchestrés avec Docker Compose, ce qui facilite le déploiement, les mises à jour indépendantes, et l’isolation des erreurs.

Le frontend est développé séparément avec Next.js, ce qui permet de rendre l’interface très réactive tout en conservant la flexibilité côté serveur pour la gestion des accès et des permissions.

*Figure 7: Architecture Microservice*

* 1. **LES AVANTAGES:**

L’approche choisie présente plusieurs avantages techniques et organisationnels :

* **Scalabilité :** chaque service peut être déployé ou mis à l’échelle indépendamment selon la charge.
* **Modularité :** les services étant autonomes, ils peuvent être développés et testés séparément.
* **Testabilité :** chaque microservice peut être testé de manière unitaire ou intégrée.
* **Sécurité :** grâce à JWT, CORS, Helmet, et une stricte validation des entrées.
* **Performance :** utilisation du cache frontend, pagination, et base de données optimisée (index, triggers).
* **Flexibilité :** intégration facilitée de futurs services ou rôles (ex. ajout d’un coordinateur ou d’un nouveau validateur).

* 1. **PRESENTATION DES TECHNOLOGIES ET OUTILS**

Le développement de la plateforme s’appuie sur un ensemble cohérent de technologies modernes, choisies pour leur stabilité, leur popularité et leur capacité à répondre aux exigences d’un système distribué.

**2.9.1 LANGAGE ET FRAMEWORKS :**

| *Technologie* | *Rôle dans le projet* | *Justification du choix* |
| --- | --- | --- |
| *Creating a Next.js Project with Monorepo using Lerna: A Complete ...* | Langage de programmation FrontEnd | Permet un rendu hybride (SSR/CSR),et offre une excellente intégration avec React |
| The first alpha of Tailwind CSS v3 is now released - Laravel News | Framework CSS utilitaire | Permet de construire des interfaces modernes rapidement, avec une grande cohérence visuelle et sans surcharge CSS inutile. |
| How To Use Radix UI Themes | Composants UI accessibles | Offre des composants bas-niveau bien structurés, accessibles (a11y) et personnalisables avec Tailwind. |
| Zustand：简化状态管理的现代React状态库-CSDN博客 | Gestion d’état côté frontend | Léger, simple à intégrer, et performant pour gérer des états complexes sans surcharger l’application. |
| TypeScript Docs - Desktop App for Mac, Windows (PC), Linux - WebCatalog | Langage de développement frontend (et potentiellement backend) | Offre un typage statique, améliore la fiabilité du code, et facilite la maintenance à long terme. |
| Node.js: o que é, quando, como e por que usar | Runtime JavaScript côté serveur | Performant, non bloquant, et parfaitement adapté aux API REST ; il permet d’unifier la stack (JS fullstack) |
| 11 Best Node Js Frameworks To Be In 2023 Tech Trends | Framework backend pour les microservices | Minimaliste, rapide à mettre en place, très répandu, et facilement extensible avec des middlewares. |
| الفرق بين كل من MySQL و SQL Server مع أبرز الميزات لكل منهما | Base de données | pour sa fiabilité et sa compatibilité avec les microservices Node.js |
| Docker Logo, symbol, meaning, history, PNG, brand | Conteneurisation des microservices et de la base de données | Permet l’isolation, la portabilité, la reproductibilité des environnements de développement et de production. |
| Getting Started · Jest | Outil de tests unitaires frontend | Framework de test moderne pour JavaScript/TypeScript, simple à configurer avec Next.js et très rapide. |

# Tableau 5 : Langage et framework de programmation

|  |  |
| --- | --- |
| **2.10** | **CONCLUSION** |

Ce choix d’architecture technique répond parfaitement aux exigences de modularité, de sécurité, de performance et de maintenabilité propres à une plateforme de gestion académique à fort impact comme celle de la FSBM. Grâce à cette approche, le système est prêt à évoluer, à intégrer de nouveaux rôles ou services, et à s’adapter à de futurs besoins institutionnels.

**Chapitre 3 : Mise en Œuvre et Développement**

**3.1 INTRODUCTON**

Ce chapitre présente les étapes de réalisation du projet, depuis la mise en place de l’environnement de développement jusqu’à l’implémentation concrète des fonctionnalités. Il décrit également les choix techniques, l’organisation du code, l’intégration des microservices et les stratégies de test utilisées.

**3.2 Mise en Place de l’Environnement de Développement**

Pour garantir une homogénéité entre les environnements de développement, de test et de production, l’ensemble des services du projet a été conteneurisé avec Docker. Le fichier docker-compose.yml orchestre les microservices backend, la base de données MySQL ainsi que les volumes partagés pour les fichiers uploadés.

Principaux éléments :

* Docker Compose pour le déploiement multi-conteneurs
* Volumes Docker pour la persistance des données (dbdata, uploads)
* Réseau interne Docker (backend) pour la communication interservices

**3.3 Architecture Modulaire**

La plateforme suit une architecture microservices, chaque fonctionnalité métier étant encapsulée dans un service indépendant. Cette organisation favorise la séparation des responsabilités et la maintenabilité du projet.

Microservices implémentés :

|  |  |  |
| --- | --- | --- |
| Service | Port | Fonction |
| Auth-service | 5000 | Authentification, gestion des rôles et génération de JWT |
| Admin-service | 5001 | Gestion des utilisateurs et administration |
| Doctorant-service | 5002 | Gestion des dossiers doctorants et fichiers |
| Enseignant-service | 5003 | Interface des enseignants (consultation, remarques) |
| Assistant-service | 5004 | Validation intermédiaire des dossiers |
| Vice-doyen-service | 5005 | Validation finale des dossiers soumis |

# Tableau 6 : Langage et framework de programmation

Chaque service possède la même structure standard : controllers, routes, models, validators, middlewares, facilitant l’industrialisation du code.

**3.3 Développement du Frontend :**

Le frontend de la plateforme a été développé avec Next.js 15 (App Router) et React 19, en utilisant TypeScript pour garantir la robustesse du code.

Structure de base :

*Figure 8 : Structure Frontend*

**3.4 Gestion de la Base de Données**

Une base MySQL 8.0 centralise toutes les données. Elle est partagée entre les microservices, chacun accédant aux tables nécessaires.

Tables principales :

* users, infos\_doctorant, dossiers, publications, communications
* dossier\_files, rapporteurs, validation\_comments, infos\_supplementaires

Particularités :

* Utilisation d’enums pour les rôles (doctorant, assistant1, etc.)
* Enum statut\_validation pour suivre l’état du dossier tout au long du processus
* Index sur les colonnes fréquemment utilisées pour optimiser les requêtes
* Triggers pour le calcul automatique de certaines statistiques

**3.5 Workflow de validation**

**3.6 Sécurité et Contrôle d’Accès**

L’authentification et les autorisations sont gérées par le auth-service, qui délivre des JWT après vérification des identifiants.

Mesures de sécurité mises en œuvre :

* JWT pour l’auth stateless
* Helmet.js pour la protection des headers http
* CORS configuré finement selon l’origine des requêtes
* Rate Limiting pour limiter les abus (100 requêtes / 15 min)
* Express Validator pour empêcher les injections et erreurs utilisateur
  1. **Communication entre Composants**
* **Frontend ↔ Backend :**
  + Les appels sont effectués via des services REST
  + Les headers incluent le token JWT pour l’accès protégé
  + Les réponses sont uniformisées au format JSON
* **Backend ↔ Backend (Microservices) :**
  + Communication HTTP via noms de conteneurs Docker (auth-service, admin-service, etc.)
  + Les dépendances sont gérées automatiquement par Docker Compose (depends\_on)
  1. **Conclusion :**

La phase de mise en œuvre s’est appuyée sur une architecture robuste et modulaire, facilitant le développement collaboratif, la réutilisation du code, et la montée en charge du système. Chaque couche – frontend, backend, base de données – a été développée avec des technologies modernes, permettant d’assurer à la plateforme performance, sécurité et évolutivité.

**Chapitre 4 : Tests et Résultats**

* 1. **INTRODUCTION :**

Ce chapitre expose de manière détaillée la stratégie de test mise en place pour garantir la fiabilité, la sécurité et la stabilité de la plateforme de gestion doctorale. Il couvre l’ensemble des tests réalisés à différents niveaux de l’application, notamment les tests frontend (interface et interactions utilisateur), les tests backend (API, logique métier, microservices), ainsi que la validation fonctionnelle du processus de traitement des dossiers.

L'accent est également mis sur les tests de sécurité, la gestion des rôles via JWT, le contrôle des accès, ainsi que la conformité du workflow de validation. Enfin, les résultats obtenus sont analysés pour évaluer la qualité globale de la solution, en identifiant les points forts et les ajustements réalisés au cours de la phase de développement.

* 1. **Objectifs des Tests :**

Les objectifs principaux des tests étaient les suivants :

* Vérifier la stabilité et la cohérence du système dans différents scénarios utilisateurs.
* Détecter et corriger les erreurs fonctionnelles ou techniques.
* Valider le workflow de validation doctorale dans toutes ses étapes.
* Garantir la sécurité des accès et la gestion des rôles via les tokens JWT.
* Assurer une bonne expérience utilisateur (temps de réponse, navigation, affichage des erreurs).
  1. **Environnement de Test:**

Les tests ont été réalisés dans un environnement de développement identique à celui de production grâce à Docker. Cela a permis de valider le comportement de l’ensemble de la plateforme dans un contexte proche de la réalité.

Outils utilisés :

* Jest pour les tests unitaires frontend (Next.js + Zustand).
* Postman pour tester manuellement les API REST backend.
* Docker Compose pour orchestrer les services et reproduire les interactions réelles.
* MySQL avec des jeux de données factices pour tester tous les cas de figure (succès, erreurs, rejets).
  1. **Types de Tests Réalisés :**
* **Tests Fonctionnels (End-to-End manuels) :**

**Les scénarios suivants ont été testés :**

* + - Création et soumission de dossier par un doctorant.
    - Validation séquentielle par Enseignant → Assistant1 → Assistant2 → Vice-Doyen → Assistant3.
    - Validation finale par Admin puis Doyen.
    - Cas de rejet à différents niveaux (Assistant1, Admin, etc.).
    - Gestion des fichiers uploadés (vérification de l’accessibilité et de la persistance).
* **Tests de Sécurité :**
  + Authentification avec JWT (validation des rôles, accès restreints par middleware).
  + Simulation d’attaques basiques : requêtes sans token, requêtes avec mauvais rôle, injection de données non valides.
  + Validation des entrées utilisateur via Express Validator.
* **TestsUnitaires Frontend :**
  + Gestion de l’état avec Zustand
  + Affichage conditionnel des composants (ex. dashboard selon rôle)
  + Validations de formulaire avec les utilitaires custom (utils/validation.ts)
  + Résultats des appels à l’API simulés (mockés)
  1. **Résultats Obtenus :**

Les résultats des tests ont confirmé que la plateforme est fonctionnelle, stable et sécurisée. Voici un résumé des points principaux validés :

|  |  |
| --- | --- |
| Test | Résultat |
| Création/soumission d’un dossier | ✅ Réussi |
| Workflow de validation complet | ✅ Réussi |
| Authentification JWT | ✅ Réussi |
| Tests unitaires Zustand | ✅ Réussi |
| Téléversement de fichiers | ✅ Réussi |
| Rejets intermédiaires | ✅ Réussi |
| Sécurité API (middleware) | ✅ Réussi |
| Performance interface | ✅ Réussi |

# Tableau 7 : Résultats des tests

* 1. **Résultats Obtenus :**

Les tests montrent que la plateforme est prête pour une mise en production pilote. Les objectifs fonctionnels sont atteints, le système est sécurisé, et l’ergonomie est fluide pour tous les rôles.

La séparation en microservices a également permis d’isoler rapidement les bugs, de déployer des correctifs ciblés et de maintenir une bonne vitesse de développement.

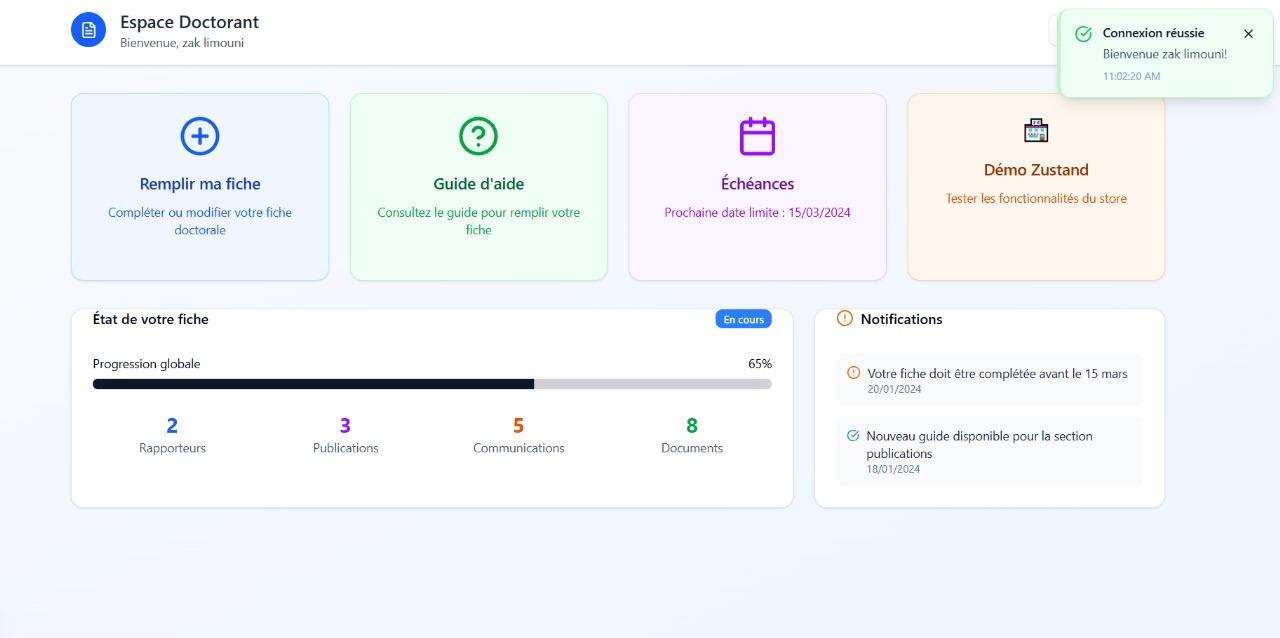
* 1. **Présentation des Interfaces Utilisateur :**

**Dashboard authentification :**

*Figure 8: Page login*

**L’interface d’accueil propose une page de connexion sécurisée :**

* Champs : email, mot de passe
* Contrôle des erreurs (mot de passe incorrect, utilisateur inconnu)
* Redirection automatique selon le rôle (admin, doctorant, assistant, etc.)
* Authentification via JWT

**Dashboard Doctorant :**

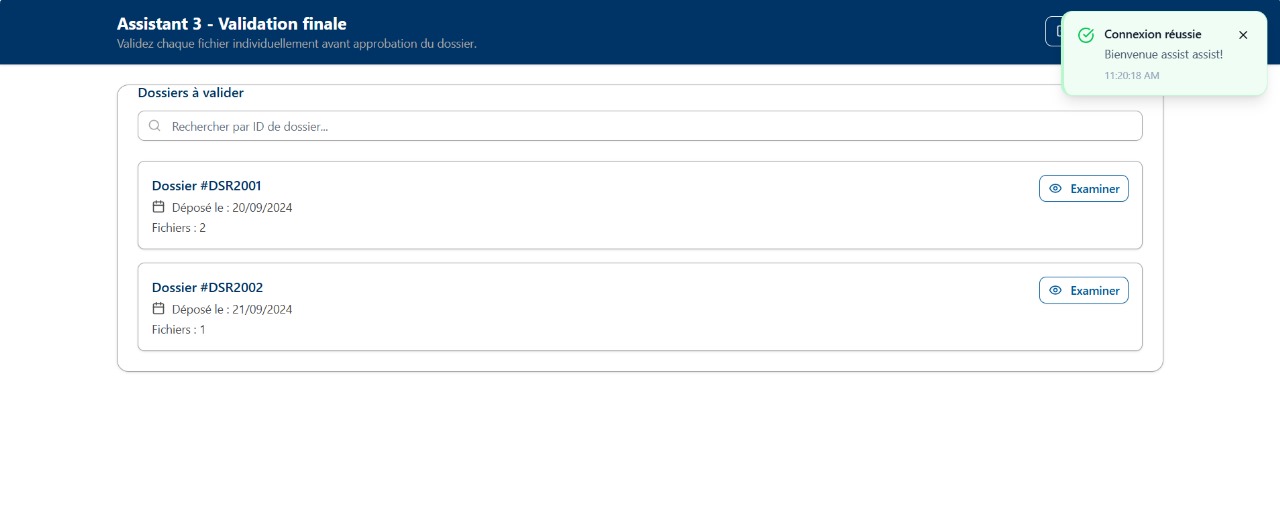
*Figure 9 : Dashboard Doctorant*

Le doctorant dispose d’un espace personnel depuis lequel il peut :

* Créer un nouveau dossier doctoral
* Modifier ou compléter un dossier en brouillon
* Télécharger des fichiers justificatifs
* Soumettre son dossier pour validation
* Suivre l’avancement du processus de validation (timeline, statut)

Éléments visibles :

* Carte d’état du dossier
* Historique des actions
* Bouton "Soumettre"
* Section "Commentaires de validation" en cas de rejet

**Dashboard Assistant :**

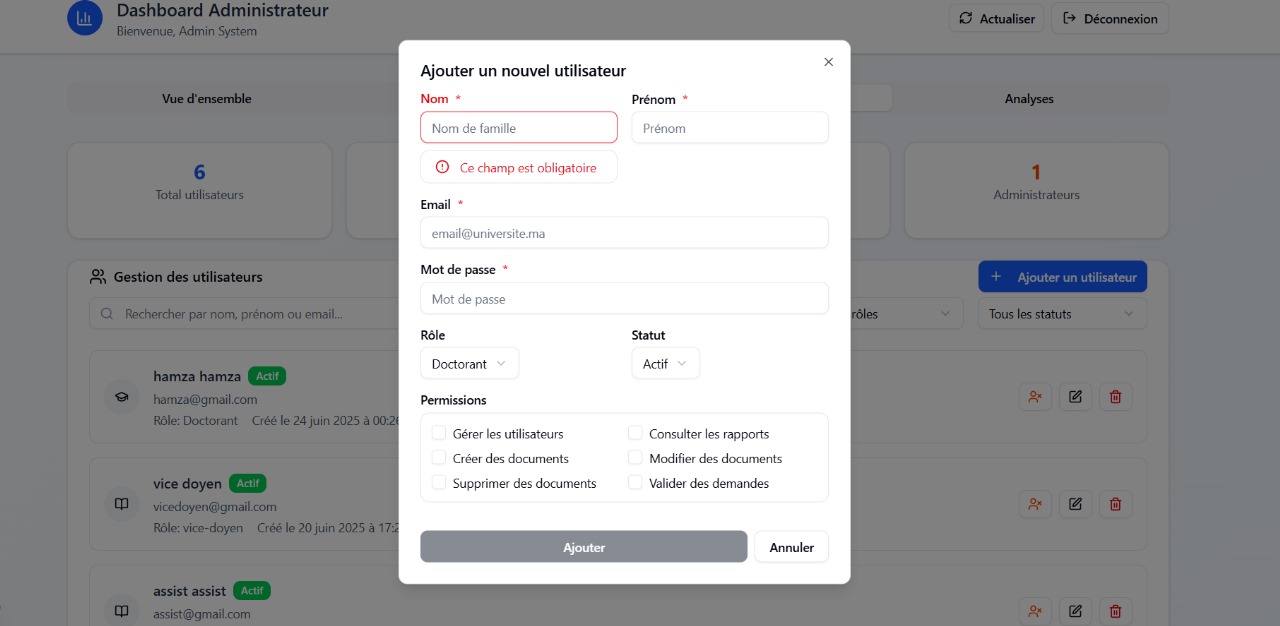
*Figure 9 : Dashboard Assistant*

**Trois niveaux d’assistants (Assistant1 à Assistant3) ont un dashboard commun, avec des fonctions spécifiques :**

**Liste des dossiers en attente de validation**

* Consultation des détails de chaque dossier
* Ajout de commentaires de validation
* Boutons Valider ou Rejeter
* Historique des validations déjà effectuées

**Dashboard Administrateur**

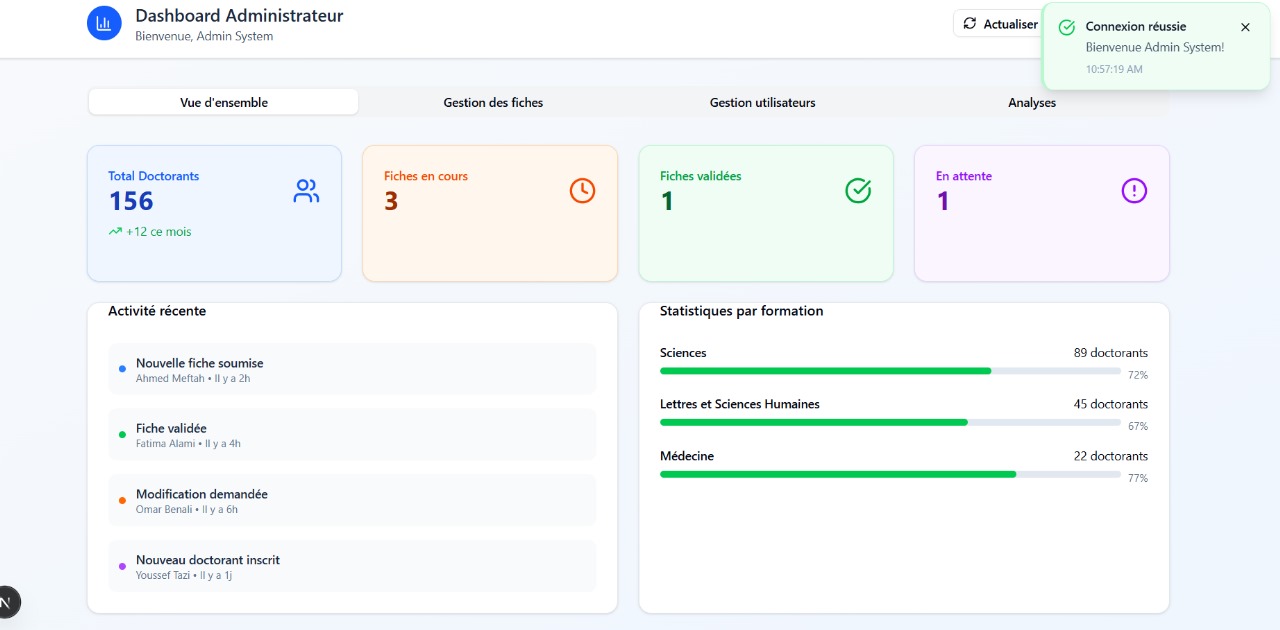
****

*Figure 9 : Dashboard Administrateur*

L’interface admin permet :

* La gestion des utilisateurs (création, modification, suppression)
* L’accès à tous les dossiers en cours ou archivés
* La validation administrative ou le rejet avec commentaire
* L’attribution manuelle de rôles

**Dashboard Doyen**



*Figure 10 : Dashboard Administrateur*

|  |  |
| --- | --- |
| **4.2** | **CONCLUSION** |

La phase de test a permis de valider en profondeur le bon fonctionnement de la plateforme, tant au niveau technique que fonctionnel. Les différents types de tests (unitaires, fonctionnels, de sécurité et de validation du workflow) ont confirmé que le système répond aux exigences attendues en termes de fiabilité, de performance et d’expérience utilisateur.

L’approche modulaire par microservices, couplée à une interface frontend réactive et bien testée, a facilité la détection et la correction rapide des anomalies. Les tests ont également démontré l’efficacité des mécanismes de sécurité, notamment l’authentification par JWT, la gestion fine des rôles, et la protection des routes critiques.

En résumé, les résultats obtenus montrent que la plateforme est prête à être déployée dans un environnement réel, avec la robustesse nécessaire pour gérer le processus doctoral de manière fluide, sécurisée et évolutive.

**CONCLUSION GENERALE:**

Ce projet de développement d'une plateforme de gestion doctorale pour la FSBM s'inscrit dans une volonté d'optimisation des processus académiques, en particulier ceux liés au suivi, à la validation et à la traçabilité des dossiers doctorants. Grâce à une architecture technique moderne, basée sur les microservices et les technologies web récentes, la solution proposée permet de répondre efficacement aux besoins spécifiques des différents acteurs du parcours doctoral (doctorants, assistants, administration, doyens, etc.).

Tout au long de ce travail, nous avons adopté une démarche structurée, depuis la conception de l’architecture jusqu’à la mise en œuvre, le développement, les tests et la validation finale. L’utilisation de technologies comme Next.js, Node.js, Docker, MySQL, Tailwind CSS et JWT a permis de garantir un haut niveau de modularité, de sécurité, de performance et de maintenabilité.

Les tests réalisés ont démontré la stabilité et la fiabilité du système, ainsi que sa capacité à gérer des flux complexes de validation tout en assurant une expérience utilisateur fluide et cohérente. Cette plateforme constitue ainsi une base solide pour une dématérialisation complète du processus doctoral, tout en restant évolutive pour intégrer de futurs besoins fonctionnels ou réglementaires.

En définitive, ce projet témoigne de l’importance d’une approche méthodique et centrée utilisateur dans la conception de solutions numériques institutionnelles, en assurant un équilibre entre exigences techniques, simplicité d’utilisation et robustesse métier.

**Bibliographie :**