République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'hamed Bougara - Boumerdès



Faculté des Sciences
Département d'Informatique

Domaine : Mathématiques Informatique
Filière : Informatique
Spécialité : Systèmes d'informatique

N° de l'Arrêté d'habilitation de la spécialité : arrêté n °872 du 26/07/2016

Mémoire de fin d'études en vu de l'obtention du Diplôme de Licence Académique

Thème

Conception et réalisation d'une Application

Collaborative pour la gestion des examens de remplacement

Présenté par :
Attouche maha
Boumaza Imane
Mahmoudi Abderrahmane
Debbah mohamed salim

Soutenu le …../…./ ….. Devant le jury composé de
Mr. Mokrani Hocine : Examinateur
Mme. Lounas Razika : Encadreur

Remerciements

Nous remercions sincèrement **ALLAH, le Tout-Puissant**, pour la force, le courage et la patience accordés durant ce travail.

Nous remercions chaleureusement **Lounas Razika** pour son soutien et ses conseils tout au long de ce projet.

Nous exprimons également notre gratitude à **Mokrani Hocine** pour avoir accepté d'évaluer ce travail avec attention et bienveillance.

Nos remerciements sincères vont aussi à nos enseignants de la Faculté des Sciences Exactes de l'Université M'hamed Bougara-Boumerdès pour leur soutien tout au long de notre cursus.

Nous exprimons un merci tout particulier à nos parents, toujours présents à nos côtés, nous soutenant, nous guidant et partageant aussi bien nos joies que nos difficultés. Leur patience, leur amour et leurs sacrifices ont été une source précieuse de motivation.

Nous remercions également nos frères et sœurs, qui nous ont accompagnés avec encouragement et soutien constant tout au long de notre parcours académique.

À tous, nous adressons notre plus sincère gratitude, car sans leur présence et leur appui, ce projet n'aurait pas pu voir le jour.

Résumé

Dans le contexte de la transformation numérique que connaît l'enseignement supérieur, ce projet vise à moderniser le système de gestion des examens de remplacement à l'Université M'Hamed Bougara de Boumerdès, encore géré par des méthodes manuelles traditionnelles. L'objectif principal est de concevoir une application web collaborative permettant à l'étudiant de soumettre une demande d'absence électronique, examinée et validée par l'administrateur, qui envoie ensuite des notifications automatiques à l'enseignant et à l'étudiant, garantissant ainsi une coordination efficace entre toutes les parties. Le travail s'articule autour de trois étapes : l'étude du système existant et l'identification de ses lacunes, la conception à l'aide des diagrammes UML, puis le développement complet de la solution et son test. L'application assure une gestion centralisée et sécurisée du parcours des examens de remplacement, de la demande jusqu'à la notification. Les perspectives d'amélioration futures incluent la prise en charge des notifications par e-mail ou SMS et l'extension de la plateforme à l'ensemble des départements.

Abstract

In the context of digital transformation in higher education, this project aims to modernize the makeup examination management system at the University of M'Hamed Bougara of Boumerdès, which is still handled through traditional manual methods. The main goal is to design a collaborative web application that allows students to submit absence requests electronically, reviewed and approved by the administrator, who then automatically notifies both the teacher and the student — ensuring smooth coordination between all stakeholders. The work is structured in three stages : analyzing the existing system and identifying its limitations, UML-based design, and full development and testing of the solution. The application ensures centralized and secure management of the makeup exam process, from request submission to notification. Future improvement prospects include supporting email or SMS notifications and extending the platform to all departments.

الملخص

في ظل التحول الرقمي الذي يشهده قطاع التعليم العالي، يهدف هذا المشروع إلى تحديث نظام إدارة امتحانات الاستدراك في جامعة محمد بوقرة ببومرداس، والذي لا يزال يُسيّر بأساليب يدوية تقليدية. الهدف الرئيسي هو تصميم تطبيق ويب تشاركي يُتيح للطالب تقديم طلب غياب إلكترونياً، يُراجعه الإداري ويوافق عليه، لتُرسَل بعدها إشعارات تلقائية إلى الأستاذ والطالب معاً، مما يضمن تنسيقاً فعالاً بين جميع الأطراف.

يتكون العمل من ثلاث مراحل: دراسة الواقع الحالي وتحديد نقائصه، والتصميم بمخططات UML، وتطوير الحل الكامل واختباره. يضمن التطبيق إدارة مركزية وآمنة لمسار امتحانات الاستدراك، من الطلب إلى الإشعار.

وتشمل آفاق التحسين المستقبلية دعم الإشعارات عبر البريد الإلكتروني أو الرسائل النصية القصيرة، وتوسيع المنصة لتشمل جميع الأقسام.

Table des matières

Table des matières ii

Table des figures iv

Liste des tableaux v

Introduction Générale 1

1 Travail collaboratif et plateformes d'outils collaboratifs 2
- 1.1 Fondements du travail collaboratif . . . 2
- 1.2 Plateformes et outils de travail collaboratif . . . 3
  - 1.2.1 Outils de communication . . . 3
  - 1.2.2 Outils de partage, d'accès et de stockage . . . 3
  - 1.2.3 Outils de gestion des connaissances (Knowledge Management) . . . 3
  - 1.2.4 Outils de coordination et de synchronisation . . . 4
  - 1.2.5 Plateformes majeures et éditeurs . . . 4
  - 1.2.6 Logiciels libres et open source . . . 4
- 1.3 Types de travail collaboratif . . . 4
  - 1.3.1 Les plateformes collaboratives . . . 5
  - 1.3.2 Le collaboratif web (Web 2.0 et réseaux professionnels) . . . 5
  - 1.3.3 Intégration aux logiciels de groupware . . . 5
- 1.4 Le cycle de la collaboration . . . 5
- 1.5 Avantages et limites des plateformes collaboratives . . . 6
  - 1.5.1 Avantages . . . 6
  - 1.5.2 Limites et défis . . . 6

2 Etude de l'existant 7
- 2.1 Définition . . . 7
  - 2.1.1 Définition d'un examen . . . 7
  - 2.1.2 Définition d'un examen de remplacement . . . 7
- 2.2 L'intérêt de l'examen de remplacement . . . 8
  - 2.2.1 . . . 8
- 2.3 Fonctionnement actuel du système . . . 8
- 2.4 Méthodes et outils utilisés . . . 9

2.5 Limites et problèmes soulevés 9
2.6 Proposition d'une solution 10

3 Spécification et analyse des besoins 11
3.1 Présentation de langage UML 11
3.2 Analyse des besoins 12
3.2.1 Besoins fonctionnels 12
3.2.2 Besoins non-fonctionnels 12
3.3 Analyse fonctionnelle 12
3.3.1 Diagramme de cas d'utilisation 12
3.3.1.1 Identification des acteurs 12
3.3.1.2 Diagramme de cas d'utilisation général 14
3.3.1.3 Diagramme de cas d'utilisation d'un Étudiant 14
3.3.1.4 Diagramme de cas d'utilisation d'un administrateur 15
3.3.1.5 Diagramme de cas d'utilisation d'un Enseignant 15
3.3.2 Description textuelle globale des cas d'utilisation par acteur 15
3.3.3 Diagramme de séquence 17
3.3.3.1 Diagramme de Séquence d'Authentification 17
3.3.3.2 Diagramme de Séquence de l'admin 18
3.3.3.3 Diagramme de Séquence de l'etudiant 18
3.3.3.4 Diagramme de Séquence de l'Enseignant 19

4 Conception 20
4.1 Les diagrammes de classes 20
4.1.1 Définition 20
4.1.2 Dictionnaire de données 20
4.1.3 Composants du diagramme de classes 21
4.1.4 Diagramme de classes 22
4.1.5 Les règles de gestion 23
4.1.6 Modèle relationnel 23
4.2 Diagramme de séquence détaillé 24

5 Implémentation 25
5.1 Les outils Utilisés 25
5.1.1 Environnement hardware 25
5.1.2 Environnement software : 26
5.1.2.1 Outils de développement 26
5.1.2.2 Outils Back-End : 27
5.2 Présentation des interfaces graphique : 27

Conclusion générale 30
Bibliographie 31

Table des figures

3.1 Diagramme général de cas d'utilisation 14
3.2 Diagramme de cas d'utilisation d'un étudiant 14
3.3 Diagramme de cas d'utilisation d'un administrateur 15
3.4 Diagramme de cas d'utilisation d'un Enseignant 15
3.5 Diagramme de Séquence d'Authentification 17
3.6 Diagramme de Séquence de l'admin 18
3.7 Diagramme de Séquence de l'etudiant 18
3.8 Diagramme de Séquence de l'Enseignant 19

4.1 Diagramme de classes 22
4.2 Diagramme de séquence détaillé 24

Liste des tableaux

3.1 Description des cas d'utilisation 13
3.2 Description globale - Acteur Étudiant 16
3.3 Description globale - Acteur Enseignant 16
3.4 Description globale - Acteur Administrateur 16

4.1 Dictionnaire de données 21

Introduction Générale

Aujourd'hui, la transformation numérique touche tous les domaines, notamment l'enseignement supérieur, où les universités cherchent à améliorer leurs méthodes de gestion administrative et pédagogique. Parmi les processus importants, la gestion des examens de remplacement nécessite une bonne coordination entre l'administration, les enseignants et les étudiants. Cependant, cette gestion repose encore souvent sur des méthodes traditionnelles ou des outils simples comme les fichiers Excel, ce qui peut engendrer des erreurs et des retards. Dans ce contexte, certaines universités ont adopté des solutions numériques modernes, tandis que dans d'autres, comme l'université M'Hamed Bougara de Boumerdès (UMBB), les outils existants restent limités et ne répondent plus aux besoins actuels. Ainsi, notre projet vise à concevoir une application collaborative permettant de faciliter la gestion des examens de remplacement, en centralisant les informations et en améliorant la communication entre les utilisateurs. Pour cela, nous avons suivi une démarche basée sur l'analyse des besoins, la conception à l'aide des diagrammes UML, puis le développement et le test de l'application.

Le travail présenté dans ce mémoire s'articule autour de trois chapitres principaux :

- **Chapitre 1 :** Ce chapitre présente le concept du travail collaboratif et les principaux outils utilisés, tout en montrant leur rôle dans l'amélioration de la communication et de la coordination.
- **Chapitre 2 :** Ce chapitre analyse le système existant, son fonctionnement et ses limites, afin de proposer une solution plus efficace et adaptée.
- **Chapitre 3 :** Ce chapitre est consacré à l'analyse des besoins fonctionnels et non fonctionnels du système, modélisés à l'aide des diagrammes UML.
- **Chapitre 4 :** Ce chapitre est dédié à la conception du système. La partie statique est basée sur le diagramme de classes, tandis que la partie dynamique s'appuie sur les diagrammes de séquence détaillés.
- **Chapitre 5 :** Ce chapitre présente les outils et langages utilisés, l'environnement de développement, ainsi que quelques exemples des interfaces principales de l'application.

Enfin, une conclusion générale synthétise nos apports et ouvre des perspectives d'avenir.

Chapitre 1

Travail collaboratif et plateformes d'outils collaboratifs

Introduction

À l'ère numérique, les technologies collaboratives sont devenues indispensables pour booster la productivité et connecter les équipes, peu importe leur distance géographique. Grâce aux TIC, ces plateformes permettent de communiquer en temps réel, de partage des documents et de gérer des projets efficacement. Ce chapitre analyse ce concept en présentant les principaux outils, leurs avantages et leurs limites.

1.1 Fondements du travail collaboratif

Le travail collaboratif est une organisation où plusieurs personnes unissent leurs compétences et partagent leurs connaissances pour atteindre un objectif commun. Axé sur une interaction continue et une responsabilité partagée, ce mode de travail s'appuie aujourd'hui sur les outils numériques pour permettre une coopération efficace à distance, de manière synchrone ou asynchrone[1].

1.2 Plateformes et outils de travail collaboratif

1.2.1 Outils de communication

Les outils de communication constituent la base du travail collaboratif. Ils permettent aux membres d'une équipe de rester connectés et d'échanger efficacement :

— **E-mails** : Permettent d'envoyer des messages électroniques à des destinataires précis, assurant un suivi écrit des échanges.

— **Messageries instantanées (IM)** : Offrent la possibilité de discuter en temps réel, partager des fichiers et créer des canaux thématiques (ex : Slack, Microsoft Teams).

— **Conférences en ligne** : Facilitent les réunions virtuelles, le partage d'écran et de contenus à distance (ex : Zoom, Google Meet).

1.2.2 Outils de partage, d'accès et de stockage

Ces outils permettent de travailler sur les mêmes documents ou applications, d'éviter les duplications et de centraliser les contributions en temps réel :

— **Édition conjointe** : Plusieurs participants peuvent rédiger, réviser et enrichir un document dans un espace virtuel partagé (ex : Google Docs, Microsoft Office Online).

— **Stockage et Cloud** : Des plateformes comme Google Drive ou OneDrive permettent de centraliser, sauvegarder et partager des documents de manière sécurisée.

— **Gestion documentaire** : Elle assure la synchronisation instantanée des fichiers et garantit l'accès aux versions les plus récentes.

1.2.3 Outils de gestion des connaissances (Knowledge Management)

Ces solutions structurent, organisent et diffusent les savoirs et compétences au sein de l'organisation :

— **Espaces de connaissances documentés** : Espaces web éditables pour enrichir collectivement le contenu et centraliser les procédures (ex : Notion, Confluence).

— **Wikis** : Outils collaboratifs permettant la co-rédaction rapide de wikis internes (ex : MediaWiki).

— **Cartographie des compétences** : Répertoire d'expertises permettant d'identifier rapidement les rôles de chacun ("qui fait quoi").

— **Moteurs de recherche et portails** : Points d'accès uniques donnant une vue structurée sur l'ensemble des ressources.

1.2.4 Outils de coordination et de synchronisation

Ces outils permettent de piloter un projet et de suivre l'avancement des tâches en temps réel ou de manière différée :

— **Agendas partagés** : Consultables par tous, ils facilitent l'organisation des réunions et la gestion des ressources communes (ex : Google Calendar).
— **Workflows** : Logiciels de gestion des flux qui organisent automatiquement les tâches et les activités d'un processus.

1.2.5 Plateformes majeures et éditeurs

Parmi les solutions intégrées les plus utilisées sur le marché actuel, on retrouve :

— **Microsoft 365** : Une suite complète (Teams, SharePoint, Outlook) facilitant la communication et la coordination globale en entreprise.
— **Google Workspace** : Offre un écosystème cloud puissant et fluide axé sur la collaboration en temps réel.
— **Écosystèmes modulaires** : Des plateformes modernes comme Notion ou Slack qui centralisent la communication et la documentation sur un seul espace.

1.2.6 Logiciels libres et open source

Des solutions souveraines et libres permettent également un travail collaboratif hautement personnalisable :

— **Nextcloud** : Une alternative open source complète à Google Drive, incluant le stockage, les agendas et la gestion des tâches.
— **Mattermost** : Une plateforme de communication d'équipe open source, alternative sécurisée aux messageries propriétaires.
— **Taiga et GitLab** : Outils de gestion de projets et de suivi de développement collaboratif adaptés aux équipes techniques [2].

1.3 Types de travail collaboratif

Le travail collaboratif a évolué fortement depuis les années 1990 grâce aux nouvelles technologies et à l'essor d'Internet. On peut distinguer plusieurs types et outils de collaboration selon leur usage et leur portée :

1.3.1 Les plateformes collaboratives

Ces espaces en ligne sont destinés à des groupes restreints et se concentrent surtout sur la communication et la coordination de projets fermés.

— **Exemples** : Espaces de travail privés sur Trello, Slack ou Google Workspace pour les PME.

— **Avantages** : Création rapide d'espaces de travail sécurisés, outils prêts à l'emploi, stockage et messagerie intégrés.

1.3.2 Le collaboratif web (Web 2.0 et réseaux professionnels)

Cette forme de collaboration implique un plus grand nombre d'utilisateurs via Internet et favorise la participation active et le réseautage.

— **Exemples** : Wikis publics (Wikipedia), plateformes de partage de code (GitHub) et réseaux sociaux professionnels (LinkedIn).

— **Avantages** : Versionnage automatique des documents, capitalisation massive des connaissances et facilité d'accès aux profils d'experts.

1.3.3 Intégration aux logiciels de groupware

Les outils collaboratifs ne se limitent pas au partage de documents, ils combinent plusieurs briques logicielles :

— **Gestion électronique de documents (GED)** : Pour la centralisation et l'archivage des fichiers.

— **Workflow** : Pour la gestion et l'automatisation des tâches.

— **Knowledge Management** : Pour capitaliser et structurer les connaissances.

— **Avantages** : Dynamisme, réactivité et proactivité des équipes, intégration des nouvelles technologies pour un usage optimal [3].

1.4 Le cycle de la collaboration

La collaboration se déroule selon un processus dynamique en quatre étapes clés :

— **Alignement et conception** : Définition des objectifs, planification du travail et organisation des ressources nécessaires.

— **Apprentissage et prise de décision** : Structuration de l'équipe à travers le choix des rôles, la clarification des responsabilités et l'établissement des buts communs.
— **Action et soutien** : Phase opérationnelle où les membres coordonnent leurs efforts, mobilisent leurs compétences et s'entraident pour surmonter les obstacles.
— **Réflexion et réajustement** : Évaluation du travail accompli pour analyser les points forts, identifier les améliorations et ajuster les plans pour le prochain cycle.

1.5 Avantages et limites des plateformes collaboratives

1.5.1 Avantages

Les plateformes collaboratives jouent un rôle essentiel dans les environnements numériques actuels. Elles facilitent la communication rapide à distance et favorisent le partage des ressources en temps réel, ce qui améliore la coordination et la flexibilité du travail en groupe tout en réduisant les déplacements. De plus, ces outils limitent les coûts liés aux infrastructures physiques, encouragent la participation active selon les compétences de chacun et structurent l'organisation des tâches pour atteindre les objectifs plus rapidement [4].

1.5.2 Limites et défis

Malgré leurs bénéfices, ces plateformes présentent certaines limites. Elles dépendent fortement d'une connexion Internet stable et peuvent engendrer une surcharge informationnelle (infobésité) pour les utilisateurs. De plus, la centralisation des données sur le Cloud soulève des défis majeurs liés à la sécurité, à la confidentialité des informations et à la résistance au changement de certains collaborateurs face aux nouveaux outils numériques.

Conclusion

Dans ce chapitre, nous avons présenté le concept du travail collaboratif qui offre de nombreux avantages et repose sur une variété d'outils utilisés au quotidien. Cependant, mobiliser plusieurs collaborateurs sur un même projet nécessite une coordination efficace et les bons outils. Dans le chapitre suivant, nous analyserons le système actuel et identifierons ses limites afin de préparer la modélisation et le développement de notre application collaborative.

Chapitre 2

Etude de l'existant

Introduction

Dans ce chapitre, nous allons présenter la gestion des examens de remplacement, en définissant son rôle au sein des établissements d'enseignement supérieur et en précisant les différents acteurs impliqués. Nous expliquerons pourquoi cette gestion est essentielle pour assurer le bon déroulement des sessions d'examen pour les étudiants absents pour des motifs légitimes, tels que la maladie, la force majeure ou d'autres justifications administratives. Ensuite, nous analyserons le système actuel afin de mettre en évidence ses limites et les difficultés rencontrées par les étudiants, les enseignants et le personnel administratif.

2.1 Définition

2.1.1 Définition d'un examen

Un examen est une évaluation officielle organisée au sein d'un établissement d'enseignement, dans le but de mesurer le niveau de connaissances et de compétences d'un étudiant dans une matière donnée, dans un cadre temporel et spatial défini, sous la supervision des enseignants et de l'administration.

2.1.2 Définition d'un examen de remplacement

Un examen de remplacement est une seconde chance offerte à l'étudiant qui n'a pas pu se présenter à l'examen initial pour une raison valable, comme une maladie ou une situation

imprévue. Au lieu de le pénaliser, l'établissement lui fixe une nouvelle date pour qu'il puisse passer son examen normalement, sans que son parcours académique en soit affecté.

2.2 L'intérêt de l'examen de remplacement

- **Sur le plan pédagogique :** L'examen de remplacement garantit l'équité entre les étudiants. Il permet d'éviter que des absences involontaires, causées par la maladie ou la force majeure, n'impactent négativement le parcours académique de l'étudiant.
- **Sur le plan du cursus :** Cette procédure préserve le cheminement normal de l'étudiant, lui évitant le redoublement ou la perte d'une année universitaire pour des raisons indépendantes de sa volonté.
- **Sur le plan psychologique :** La mise en place d'une solution réglementaire réduit l'anxiété des étudiants face aux imprévus et renforce leur sentiment de sécurité au sein de l'établissement.
- **Sur le plan institutionnel :** Cela reflète une approche managériale moderne et flexible de la part de l'administration, privilégiant l'accompagnement des étudiants plutôt qu'une application rigide des règlements.

2.2.1

sectionLes acteurs intervenants Dans de nombreux établissements universitaires, la gestion des examens de remplacement se fait encore de manière traditionnelle ou semi-numérisée, sans véritable uniformisation. Le processus implique essentiellement trois acteurs principaux :

- **L'administration pédagogique :** Chargée de centraliser les demandes, de vérifier la validité des justificatifs d'absence et de coordonner.
- **Les enseignants :** Responsables de la préparation des sujets de remplacement, de l'organisation des sessions d'évaluation, ainsi que de la communication avec les étudiants concernés.
- **Les étudiants :** Qui soumettent leurs pièces justificatives dans les délais réglementaires et participent aux épreuves de remplacement.

2.3 Fonctionnement actuel du système

Au niveau du département d'informatique, le processus actuel se déroule généralement selon les étapes chronologiques suivantes :

1. L'étudiant absent à un examen officiel soumet une demande de remplacement, accompagnée d'une pièce justificative (certificat médical ou document administratif).
2. L'administration pédagogique procède à la vérification manuelle de la validité du justificatif déposé.
3. Les demandes validées sont enregistrées manuellement dans un fichier Excel ou consignées sur des registres papier.
4. La liste des étudiants retenus est transmise aux enseignants concernés par courrier électronique ou sous format imprimé.
5. L'organisation logistique de l'examen (choix de la date, de la salle et des surveillants) est planifiée manuellement, puis communiquée directement aux étudiants via leur courrier électronique académique ou à travers les canaux numériques du département (E-mail, Telegram).
6. Les notes et les résultats définitifs de ces examens sont ensuite transmis numériquement aux étudiants concernés et intégrés dans les données administratives.

2.4 Méthodes et outils utilisés

Actuellement, la gestion des examens de remplacement au sein du département s'appuie sur une combinaison d'outils traditionnels et de canaux numériques, à savoir :

- **Les fichiers Excel :** Utilisés principalement par l'administration pour le suivi interne, l'enregistrement des demandes et la centralisation des données des étudiants.
- **Les documents papier :** Destinés à la réception, l'archivage et la vérification des pièces justificatives d'absence déposées.
- **Le courrier électronique académique :** Privilégié pour les échanges officiels et directs entre l'administration, le corps enseignant et les étudiants.
- **La communication directe :** Les réunions de coordination et les contacts téléphoniques pour valider l'organisation logistique des épreuves.
- **Les canaux numériques du département :** Notamment Telegram et la page Facebook officielle, exploités pour la diffusion instantanée des plannings, des dates et des salles d'examen.

2.5 Limites et problèmes soulevés

L'analyse du système actuel révèle deux catégories de limites majeures :

Sur le plan technique :

- **Absence d'automatisation :** Le traitement manuel des demandes ralentit le flux de travail.
- **Pas de base de données :** L'information est fragmentée et difficile d'accès.
- **Insécurité des données :** Le recours à Excel ou au papier expose aux risques de perte et de modification.

Sur le plan organisationnel :

- **Défaut de traçabilité :** Impossible de suivre l'avancement des demandes en temps réel.
- **Redondance des tâches :** La ressaisie répétitive des informations augmente les risques d'erreurs.
- **Absence de reporting :** L'extraction de statistiques et de bilans semestriels est complexe.

En conclusion, ces défaillances alourdissent la charge administrative et compliquent la coordination.

2.6 Proposition d'une solution

Les limites soulevées mettent en évidence la nécessité de basculer vers un système moderne, structuré et collaboratif. Afin d'optimiser l'efficacité globale et de garantir la sécurité des données, nous proposons le développement d'une application web dédiée, reposant sur les piliers suivants :

- **Centralisation des données :** Implémentation d'une base de données unique et d'une interface web ergonomique pour simplifier la gestion quotidienne.
- **Gestion multi-utilisateurs :** Mise en place d'un système d'accès sécurisé avec des rôles et des permissions strictes pour chaque acteur (Administration, Enseignants, Étudiants).
- **Collaboration en temps réel :** Automatisation des notifications et des flux d'échanges sur la plateforme pour assurer une coordination fluide.
- **Sécurisation et traçabilité :** Intégration d'un mécanisme d'authentification rigoureux protégeant l'intégrité des informations contre toute modification non autorisée.

Conclusion

Ce chapitre a permis d'analyser le système actuel et d'en identifier les limites. Ce constat confirme la nécessité de déployer l'application web proposée pour automatiser ce processus.

Chapitre 3

Spécification et analyse des besoins

Introduction

Dans ce chapitre, nous présentons la spécification et l'analyse détaillée des besoins de notre système de gestion des examens de remplacement. À travers une modélisation en UML, nous commençons par définir l'architecture fonctionnelle de notre solution. Les diagrammes de cas d'utilisation permettent de représenter les interactions entre les différents acteurs du système, notamment l'administration pédagogique, les enseignants et les étudiants. Par ailleurs, les diagrammes de séquence décrivent le déroulement des opérations. Cette approche nous permet de mieux comprendre les besoins des utilisateurs et de proposer une solution adaptée et efficace pour la gestion des examens de remplacement.

3.1 Présentation de langage UML

Les diagrammes ne nécessitent pas toujours un langage strict pour communiquer des idées, cependant, l'UML (Unified Modeling Language) est un langage graphique standardisé qui permet de représenter un système, d'organiser les informations et de faciliter la conception. Les diagrammes peuvent servir à générer des idées, à structurer l'information ou à documenter le code. Une définition claire de l'Unified Modeling Language est importante pour que les modèles reflètent correctement l'intention de leurs créateurs.[7]

3.2 Analyse des besoins

3.2.1 Besoins fonctionnels

Les exigences fonctionnelles définissent l'ensemble des opérations et des fonctionnalités que le système doit intégrer pour répondre efficacement aux besoins des utilisateurs. Dans le cadre de notre application, ces exigences se traduisent par cinq fonctionnalités majeures : la gestion des demandes d'examens de remplacement, la gestion des différents profils d'utilisateurs (étudiants, enseignants et administration), la consultation et la modification des modules, ainsi que la planification et le suivi des examens. Enfin, un système de notifications est prévu pour alerter en temps réel les acteurs concernés.

3.2.2 Besoins non-fonctionnels

Les exigences non-fonctionnelles déterminent les critères de qualité, de performance et de sécurité indispensables au bon fonctionnement du système. Pour notre application, ces exigences se concentrent sur la protection rigoureuse des données sensibles, l'optimisation des performances afin de traiter rapidement un flux important de demandes d'examens de remplacement, ainsi que la portabilité pour garantir une accessibilité optimale sur différentes plateformes.

3.3 Analyse fonctionnelle

3.3.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation illustre les interactions entre les utilisateurs et le système à travers des actions principales. Il permet de représenter les besoins fonctionnels de façon claire, sans aborder les aspects techniques en détail [5] .

3.3.1.1 Identification des acteurs

Un acteur représente le rôle d'un utilisateur dans le système. Il peut s'agir d'une personne réelle ou d'un composant externe, matériel ou logiciel, qui interagit avec l'application. Un acteur peut également consulter et modifier l'état du système en envoyant et en recevant des informations.

notre application, les acteurs identifiés sont : l'Étudiant, l'Enseignant et l'Administrateur

Acteur	**Cas d'utilisation**	**Description**
Étudiant	S'authentifier	– Saisir email et mot de passe – Vérifier les identifiants – Accéder au système
	Gérer une demande	– Ajouter une demande – Modifier une demande – Annuler la demande – Déposer les justificatifs – Consulter ses demandes
	Consulter planification des examens	– Visualiser les dates des examens
Administrateur	S'authentifier	– Saisir les identifiants – Accéder aux fonctionnalités
	Gérer les demandes	– Consulter la liste des demandes et les justificatifs – Valider la demande – Rejeter la demande – Notifier l'étudiant
	Gérer les spécialités	– Ajouter une spécialité – Modifier une spécialité – Supprimer une spécialité – Consulter les spécialités
	Gérer les modules / niveaux	– Ajouter un module ou niveau – Modifier un module ou niveau – Supprimer un module ou niveau – Consulter les modules et niveaux
	Gérer les étudiants	– Ajouter un étudiant – Modifier les informations – Supprimer un étudiant – Consulter la liste des étudiants
	Planifier un examen	– Ajouter un examen – Supprimer un examen – Notifier un examen – Consulter un examen
	Envoyer des notifications	– Notifier les utilisateurs
Enseignant	Consulter notifications	– Recevoir les alertes de surveillance et de planification
	Consulter la liste des examens	– Visualiser le planning des examens à surveiller (date, heure, salle)
	Consulter la liste des étudiants	– Accéder à la liste des étudiants concernés par son propre module

TABLE 3.1 – Description des cas d'utilisation

3.3.1.2 Diagramme de cas d'utilisation général

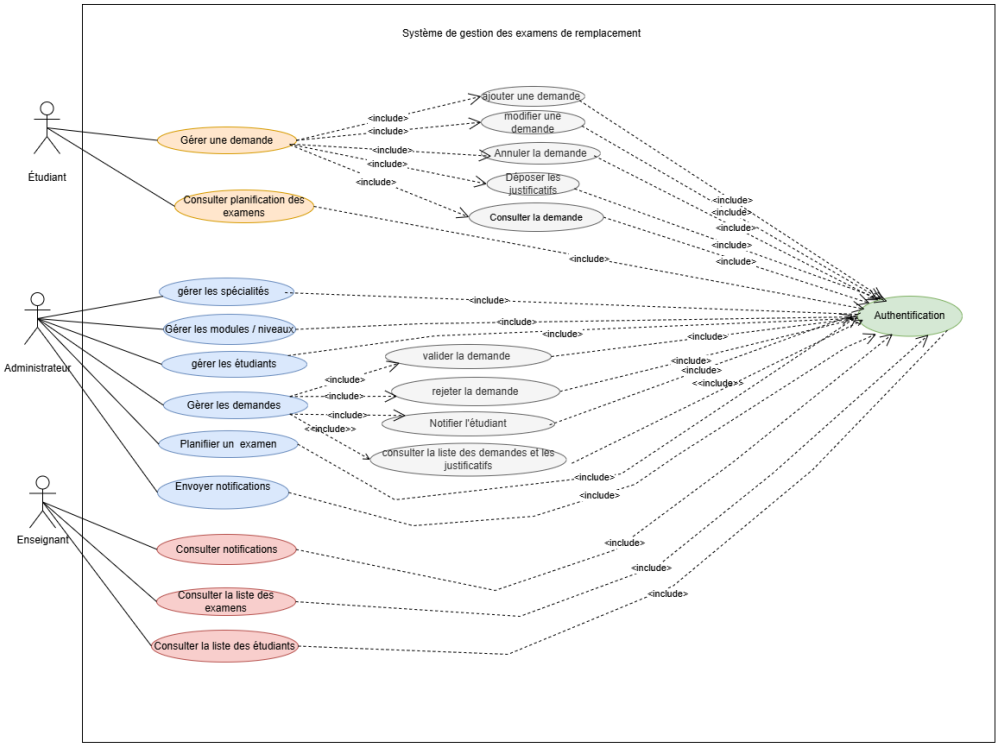


FIGURE 3.1 – Diagramme général de cas d'utilisation

3.3.1.3 Diagramme de cas d'utilisation d'un Étudiant

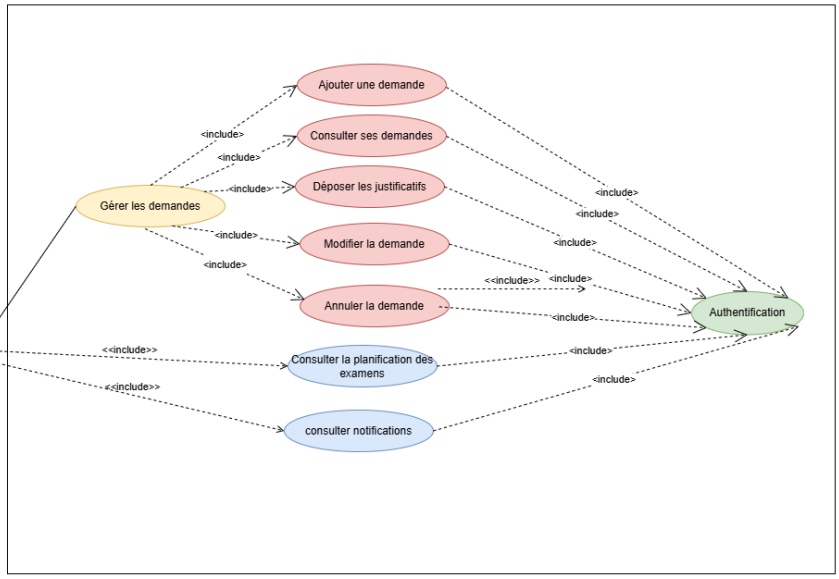


FIGURE 3.2 – Diagramme de cas d'utilisation d'un étudiant

3.3.1.4 Diagramme de cas d'utilisation d'un administrateur

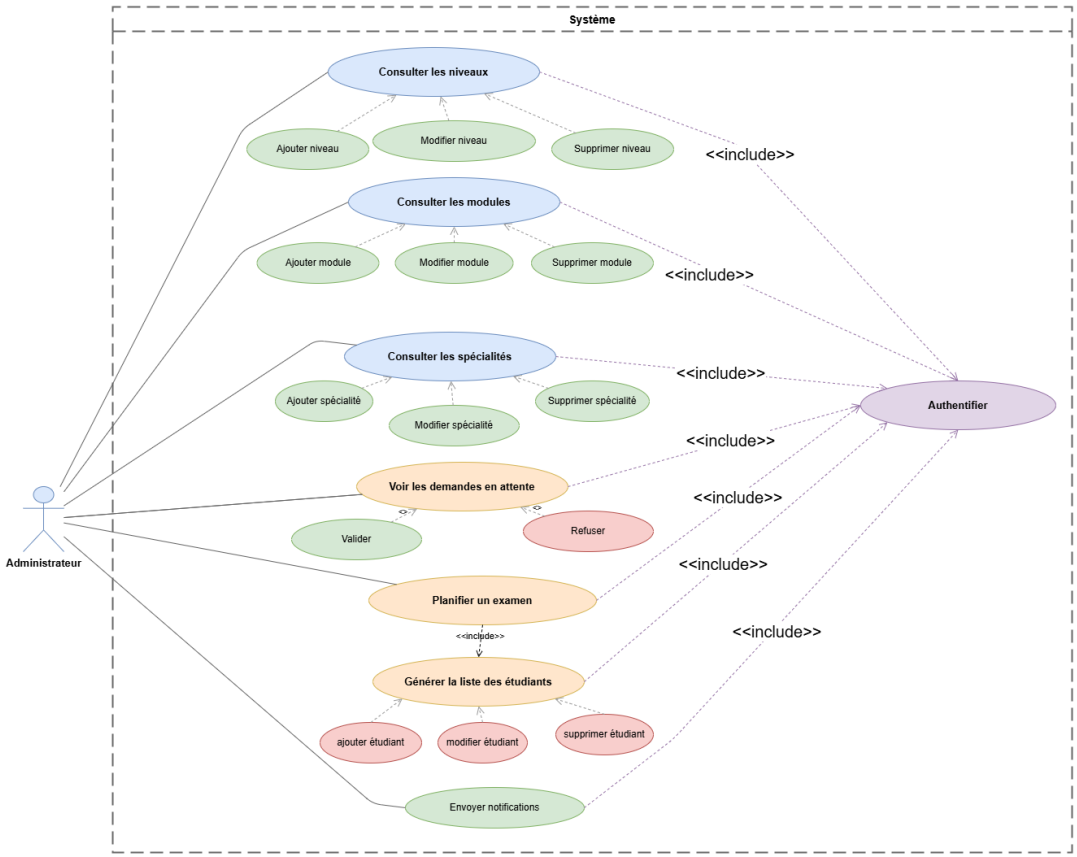


FIGURE 3.3 – Diagramme de cas d'utilisation d'un administrateur

3.3.1.5 Diagramme de cas d'utilisation d'un Enseignant

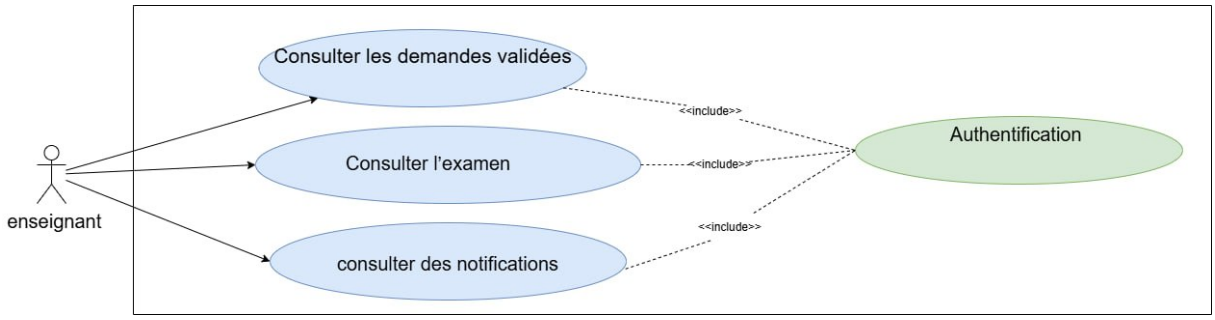


FIGURE 3.4 – Diagramme de cas d'utilisation d'un Enseignant

3.3.2 Description textuelle globale des cas d'utilisation par acteur

Nom de l'acteur	Étudiant
Description globale	S'authentifie pour gérer ses demandes, consulter le planning et recevoir les notifications.
Cas d'utilisation	S'authentifier, Ajouter/Consulter/Modifier/Annuler demande, Déposer justificatifs, Consulter planning et notifications.
Pré-conditions	Posséder un compte actif dans le système.
Flux principal	**1. Authentification :** Saisit ses identifiants et accède à son espace. **2. Ajouter demande :** Remplit le formulaire et soumet les justificatifs. **3. Gérer demandes :** Consulte, modifie ou annule sa demande en attente. **4. Consultation :** Consulte le calendrier des examens et ses notifications.
Flux alternatifs	Identifiants incorrects : message d'erreur ; Champs manquants : envoi bloqué.
Post-conditions	Demande enregistrée avec le statut "En attente".

TABLE 3.2 – Description globale - Acteur Étudiant

Nom de l'acteur	Enseignant
Description globale	S'authentifie pour suivre les demandes validées, consulter le planning et recevoir les notifications.
Cas d'utilisation	S'authentifier, Consulter demandes validées, Consulter examens, Consulter notifications.
Pré-conditions	Être enregistré dans la base de données du système.
Flux principal	**1. Authentification :** Saisit ses identifiants et accède à son tableau de bord. **2. Suivi des demandes :** Consulte les demandes validées pour organiser les examens. **3. Consultation examen :** Consulte les détails et la planification des examens programmés. **4. Notifications :** Consulte les alertes envoyées par le système.
Flux alternatifs	Authentification échouée : accès refusé, invitation à réessayer.
Post-conditions	Connecté avec accès aux listes et examens récupéré.

TABLE 3.3 – Description globale - Acteur Enseignant

Nom de l'acteur	Administrateur
Description globale	Gère les structures, valide les dossiers, planifie les examens et envoie les notifications.
Cas d'utilisation	S'authentifier, Gérer structures/étudiants (CRUD), Valider/Refuser dossiers, Planifier examens, Notifier.
Pré-conditions	Disposer de droits admin sur l'application.
Flux principal	**1. Authentification :** Se connecte à l'espace d'administration. **2. Gestion :** Met à jour étudiants, modules, spécialités et niveaux. **3. Dossiers :** Vérifie les justificatifs, valide ou refuse. **4. Planification :** Planifie l'examen et envoie les notifications.
Flux alternatifs	Identifiants erronés : accès refusé ; Justificatifs manquants : demande rejetée.
Post-conditions	Statuts mis à jour, examens planifiés, notifications transmises.

TABLE 3.4 – Description globale - Acteur Administrateur

3.3.3 Diagramme de séquence

Le diagramme de séquence a pour but d'illustrer les interactions entre les objets en mettant en évidence l'ordre temporel des échanges. Cette représentation peut être effectuée à partir des cas d'utilisation en tenant compte des différents scénarios possibles. [6] .

3.3.3.1 Diagramme de Séquence d'Authentification

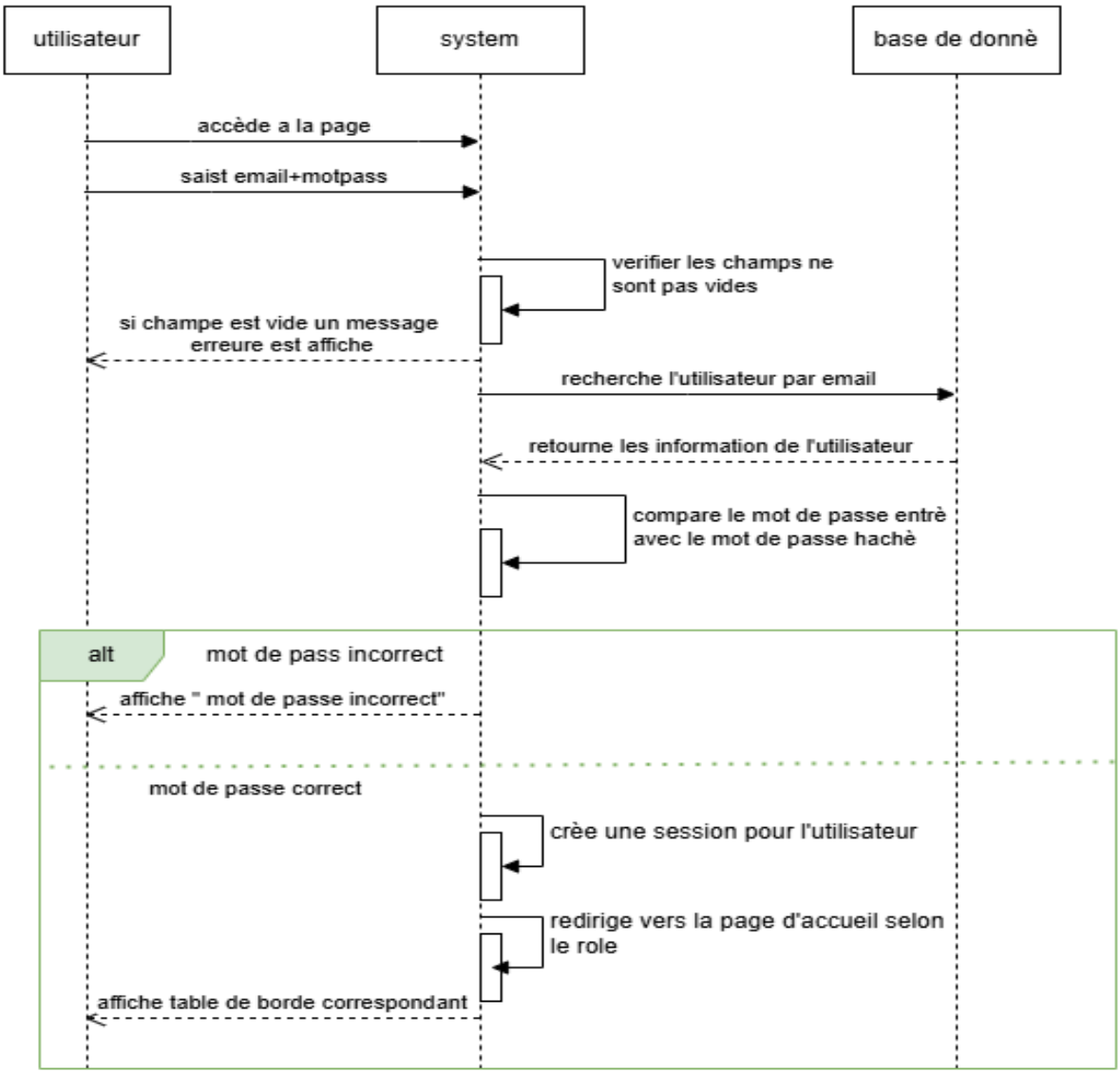


FIGURE 3.5 – Diagramme de Séquence d'Authentification

3.3.3.2 Diagramme de Séquence de l'admin

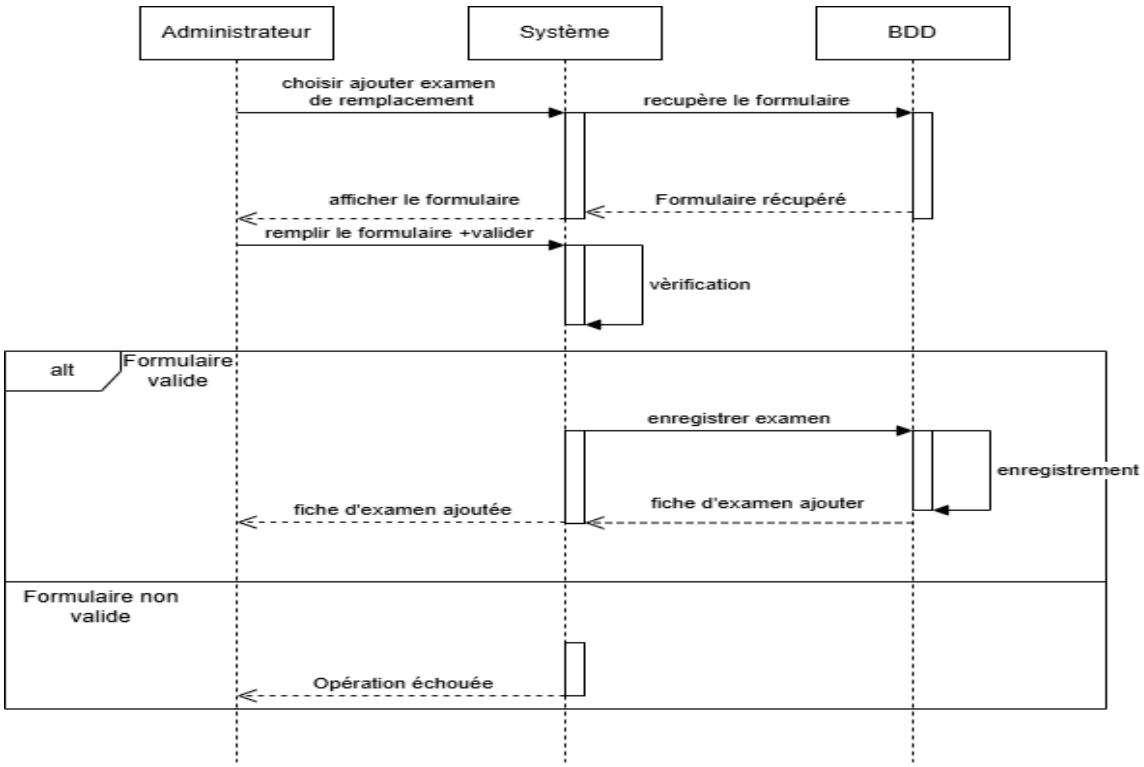


FIGURE 3.6 – Diagramme de Séquence de l'admin

3.3.3.3 Diagramme de Séquence de l'etudiant

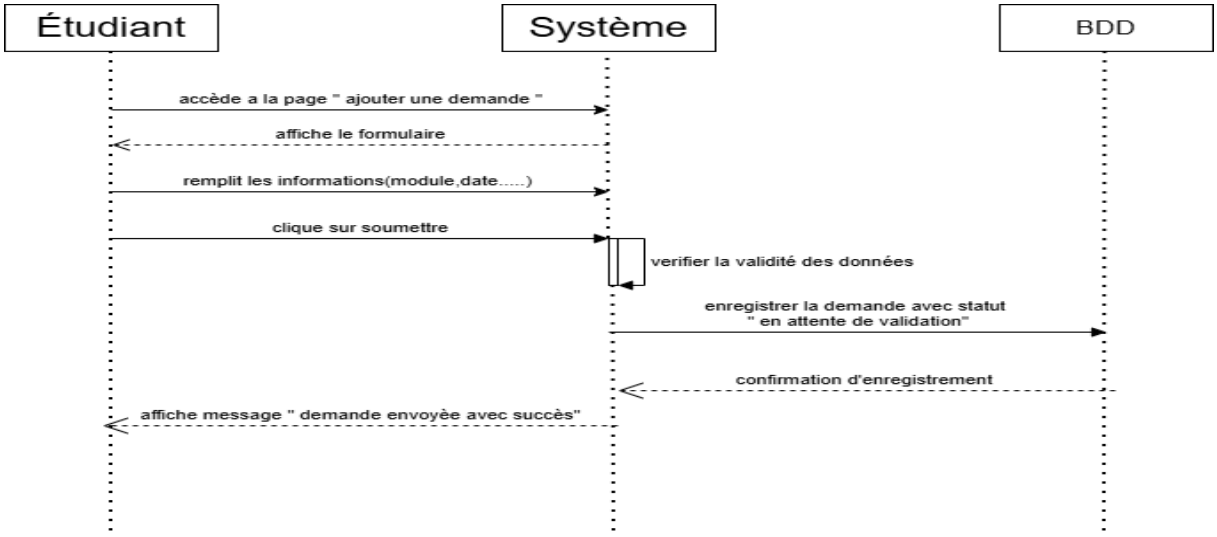


FIGURE 3.7 – Diagramme de Séquence de l'etudiant

3.3.3.4 Diagramme de Séquence de l'Enseignant

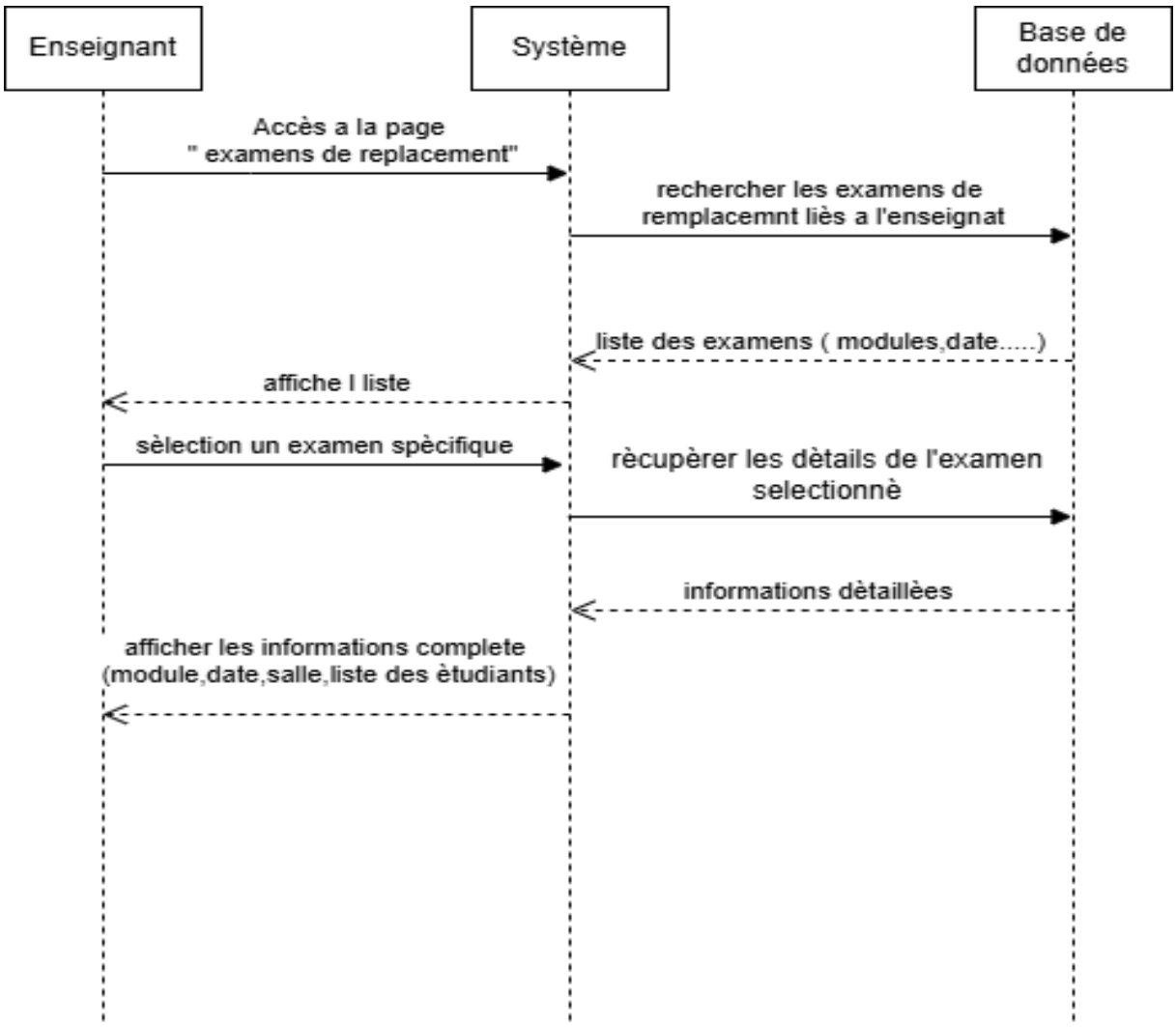


FIGURE 3.8 – Diagramme de Séquence de l'Enseignant

conclusion

En somme, ce chapitre nous a permis de définir les exigences du système et de modéliser les interactions des utilisateurs via le langage UML. Cette étape essentielle pose des bases claires pour la conception, nous garantissant ainsi le développement d'une application fluide, sécurisée et adaptée aux besoins.

Chapitre 4

Conception

Introduction

Ce chapitre décrit la conception détaillée de notre système. La partie statique est illustrée par le diagramme de classes et son dictionnaire de données associé, tandis que la partie dynamique s'appuie sur le diagramme de séquence détaillé.

4.1 Les diagrammes de classes

4.1.1 Définition

Le diagramme de classes en UML est un schéma qui permet de représenter la structure statique d'un système. Il met en évidence les différentes classes, leurs attributs, leurs méthodes ainsi que les relations qui les lient[7].

4.1.2 Dictionnaire de données

Le dictionnaire de données est un document de spécification qui recense et décrit toutes les données d'un système d'information (signification, type, codification et valeurs possibles). En lien direct avec le diagramme de classes, il constitue le socle descriptif de chaque attribut[8].

Désignation	Code attribut	Type	Taille
Identifiant de l'utilisateur	id_utilisateur	N	11
Identifiant de l'étudiant	id_etudiant	N	11
Identifiant de l'administrateur	id_administrateur	N	11
Identifiant de l'enseignant	id_enseignant	N	11
Identifiant de la demande	id_demande	N	11
Identifiant de l'examen	id_examen	N	11
Identifiant de la notification	id_notif	N	11
Identifiant de la spécialité	id_specialite	N	11
Identifiant du module	id_module	N	11
Identifiant du niveau	id_niveau	N	11
Nom de l'utilisateur	nom	A	255
Email de l'utilisateur	email	AN	255
Mot de passe de l'utilisateur	mot_passe	AN	255
Niveau d'étude	niveau	AN	50
Spécialité	specialite	AN	100
Statut de la demande	statut	AN	50
Justification de la demande	justification	AN	255
Intitulé du module	intitule	AN	150
Libellé du niveau	libelle	AN	50
Message de notification	message	AN	255
Date de la demande	date_demande	Date	10
Date de l'examen	date_examen	Date	10
Date de la notification	date_notif	Date	10
Heure de l'examen	heure	N	4

TABLE 4.1 – Dictionnaire de données

4.1.3 Composants du diagramme de classes

Le diagramme de classes utilisé dans notre conception est constitué des éléments suivants :

— **La classe :** Une abstraction qui regroupe des objets ayant des propriétés (attributs) et des comportements (méthodes) communs.

— **L'attribut :** Une propriété descriptive d'une classe. Chaque attribut possède un nom, un type de données et une **visibilité** (Privé - ou Public +).

— **La méthode :** Une opération ou une action qu'une classe peut réaliser, définissant ainsi

son comportement.

— **L'association :** Une relation sémantique entre deux ou plusieurs classes, caractérisée par des **multiplicités** (ex : 0..∗, 1) qui indiquent le nombre d'occurrences possibles.

— **La généralisation / L'héritage :** Une relation qui permet de définir une classe générale (super-classe comme utilisateur) et des classes spécialisées (sous-classes comme étudiant, enseignant) qui héritent de ses attributs et méthodes.

4.1.4 Diagramme de classes

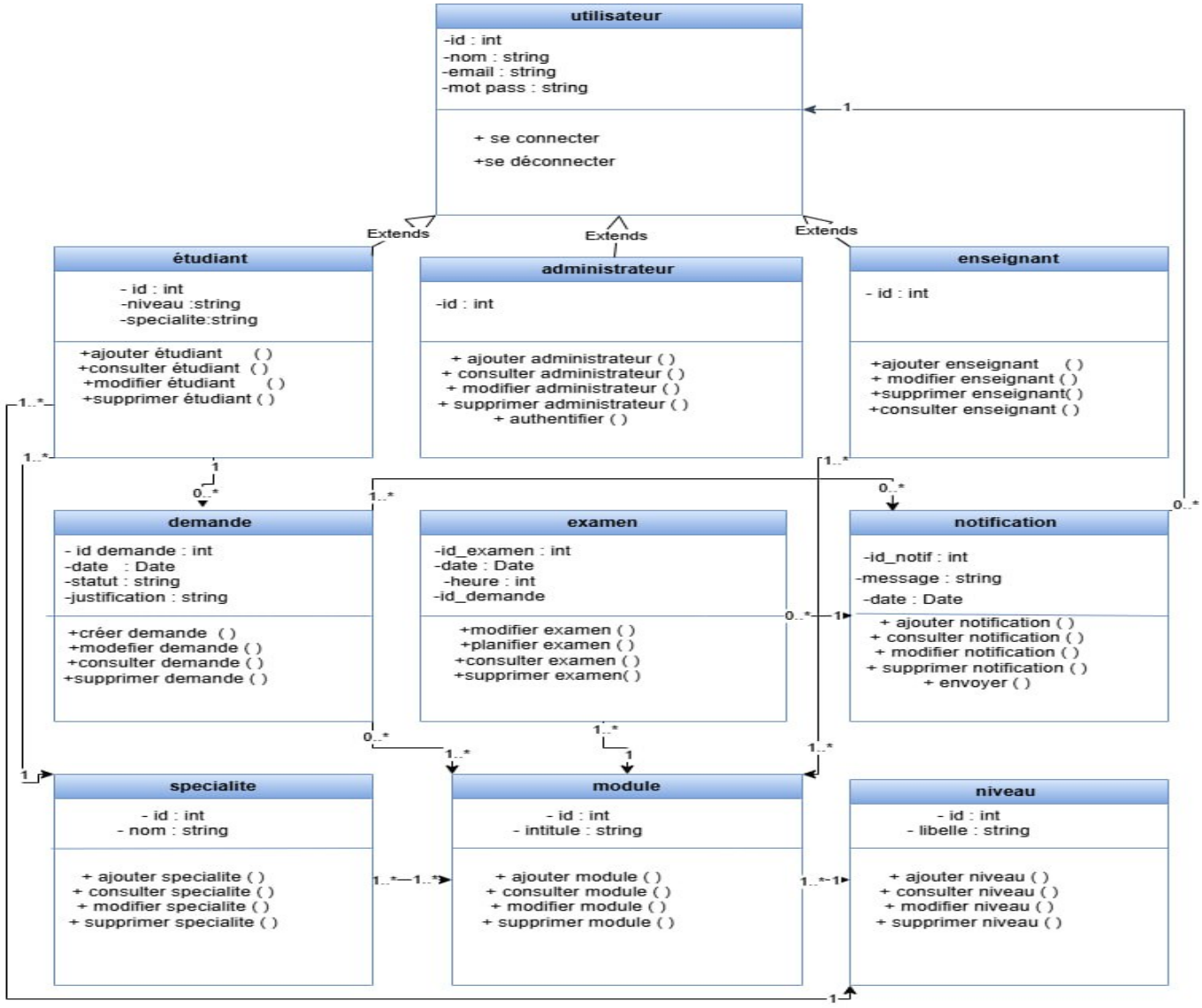


FIGURE 4.1 – Diagramme de classes

4.1.5 Les règles de gestion

- Une demande peut générer zéro ou plusieurs (0..∗) examens, et un examen peut concerner une ou plusieurs (1..∗) demandes.
- Un étudiant peut déposer zéro ou plusieurs (0..∗) demandes, et une demande peut être déposée par un et un seul (1) étudiant.
- Un examen concerne un ou plusieurs (1..∗) modules, et un module fait l'objet d'un et un seul (1) examen de remplacement.
- Une spécialité comporte un ou plusieurs (1..∗) modules, et un module peut appartenir à une ou plusieurs (1..∗) spécialités.
- Un niveau comporte un ou plusieurs (1..∗) modules, et un module appartient à un et un seul (1) niveau.
- Un étudiant est inscrit dans une et une seule (1) spécialité, et une spécialité peut contenir un ou plusieurs (1..∗) étudiants.
- Un utilisateur peut recevoir zéro ou plusieurs (0..∗) notifications, et une notification est adressée à un et un seul (1) utilisateur.

4.1.6 Modèle relationnel

Le modèle relationnel est une méthode de conception de bases de données où les informations sont organisées sous forme de tables. Chaque table est composée de lignes (qui représentent les enregistrements) et de colonnes (qui représentent les attributs). Ce modèle est basé sur des notions mathématiques comme les ensembles et les relations, et il est manipulé grâce au langage SQL.

— **Utilisateur**(id_utilisateur, nom, email, mot_passe)
— **Etudiant**(id_utilisateur#, niveau, specialite, id_specialite#)
— **Administrateur**(id_utilisateur#)
— **Enseignant**(id_utilisateur#)
— **Demande**(id_demande, date_demande, statut, justification, id_utilisateur#)
— **Examen**(id_examen, date_examen, heure, id_module#)
— **Concerner**(id_examen#, id_demande#)
— **Notification**(id_notif, message, date_notif, id_utilisateur#)
— **Module**(id_module, intitule, id_specialite#, id_niveau#)
— **Specialite**(id_specialite, nom, id_niveau#)
— **Niveau**(id_niveau, libelle)

4.2 Diagramme de séquence détaillé

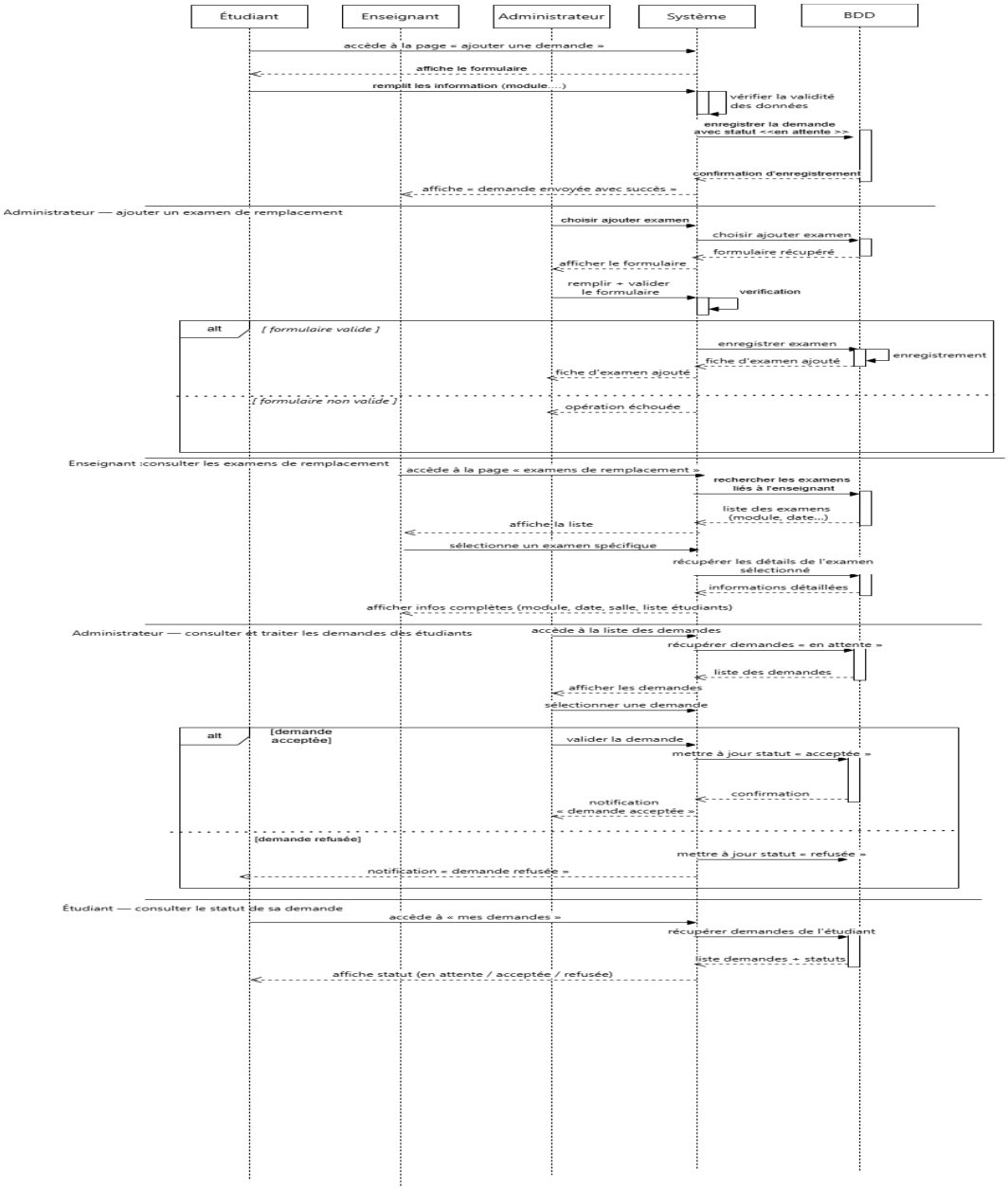


FIGURE 4.2 – Diagramme de séquence détaillé

Conclusion

Ce chapitre a présenté la conception du système à travers le langage UML, en mettant en évidence les principales entités ainsi que leurs relations. Cette étape de conception permet de mieux comprendre l'organisation et le fonctionnement du système. Elle constitue également une base importante pour la phase d'implémentation, en facilitant le développement et en assurant une meilleure structuration du travail.

Chapitre 5

Implémentation

Introduction

Dans ce chapitre, nous présenterons l'environnement matériel et logiciel utilisé, ainsi que les principaux choix techniques, puis les différentes interfaces de l'application.

5.1 Les outils Utilisés

5.1.1 Environnement hardware

- **Processeur :** AMD Ryzen 5 PRO 5650U with Radeon Graphics (2.30 GHz)
- **Mémoire RAM installée :**16 Go.
- **Type de système :** Système d'exploitation 64 bits.
- **Système d'exploitation :** : Windows 11, 64 bits (x64-based processor)

5.1.2 Environnement software :

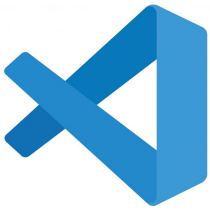
5.1.2.1 Outils de développement



MySQL :
est un système de gestion de bases de données relationnelles open-source, reconnu pour ses hautes performances, sa robustesse et sa simplicité d'intégration dans les applications web.[9]



Xampp :
est un logiciel gratuit qui fournit un environnement complet pour développer et tester des sites web localement sur un ordinateur. Il contient des outils essentiels tels que : Apache, PHP, MySQL, phpMyAdmin. Il est utilisé pour exécuter et programmer des sites web sans avoir besoin d'un véritable serveur Internet [10].



Visual studio code :
est un éditeur de code développé par Microsoft. Il sert à écrire, modifier et tester des programmes et des sites web. Il est rapide, léger et prend en charge plusieurs langages de programmation comme PHP, Python et JavaScript. [11]



HTML
HyperText Markup Language (HTML) signifie « HyperText Markup Language », que l'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé pour créer et structurer le contenu d'une page web.[12]



CSS :
est un langage utilisé pour contrôler le style et la mise en page des pages web, comme les couleurs, les polices, les tailles et la position des éléments. Il permet aussi de rendre les pages web plus belles et organisées .[13]

5.1.2.2 Outils Back-End :



PHP :
est un langage de programmation qui permet de créer des fonctions, des classes et des sites web dynamiques et interactifs. Il sert à organiser et exécuter des instructions de programmation.[14]



JavaScript :
est un langage de programmation utilisé pour rendre les pages web interactives et dynamiques en ajoutant des fonctionnalités supplémentaires.[15]



PhpMyAdmin :
est un outil écrit en PHP qui permet de gérer des bases de données MySQL via une interface web. Il permet de créer, supprimer et modifier des bases de données et des tables, exécuter des requêtes SQL et gérer les utilisateurs et les permissions.[16]

5.2 Présentation des interfaces graphique :

Dans cette partie, nous présentons les interfaces graphiques à travers des captures d'écran illustrant les différentes fonctionnalités de notre application :

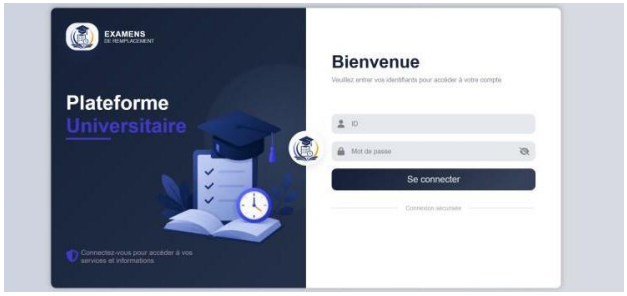


Figure 5.1 - Login page

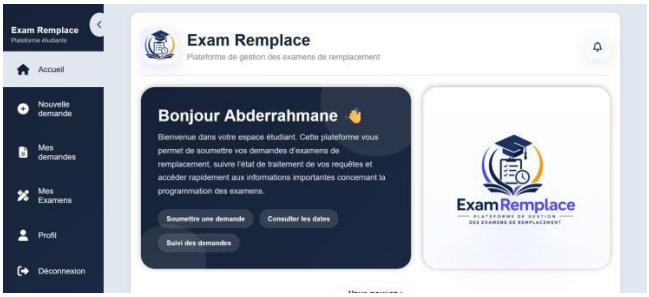


Figure 5.2 - interface etudiant



Figure 5.3 -soumettre une demande.



Figure 5.4 – Profil Étudiant

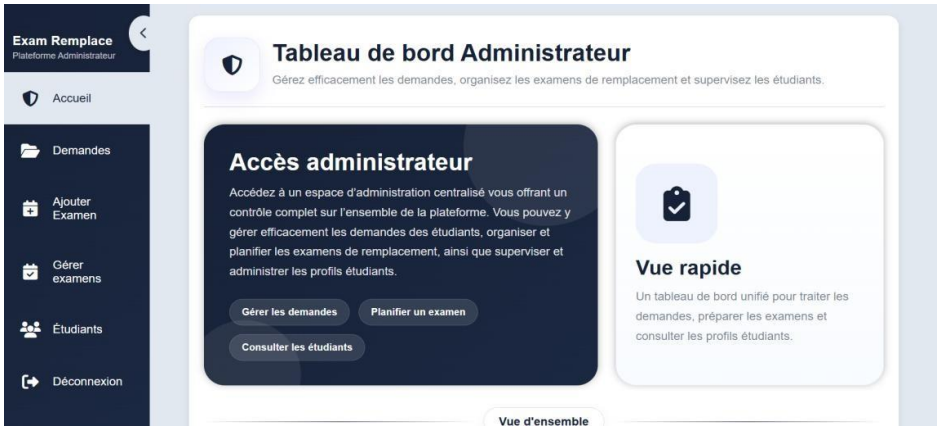


Figure 5.4 -Tableau de bord Administrateur



Figure 5.5 - Gérer les demandes

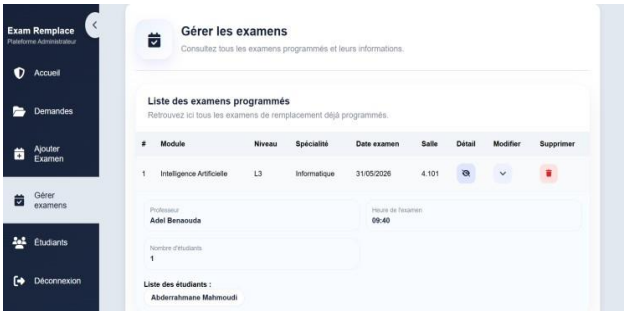


Figure 5.6 - Nouvel examen



Figure 5.7 -Tableau de bord Enseignant

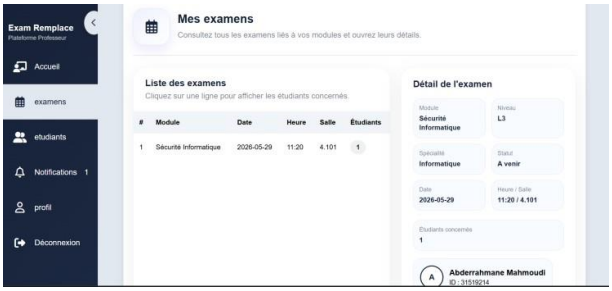


Figure 5.8 - Mes examens

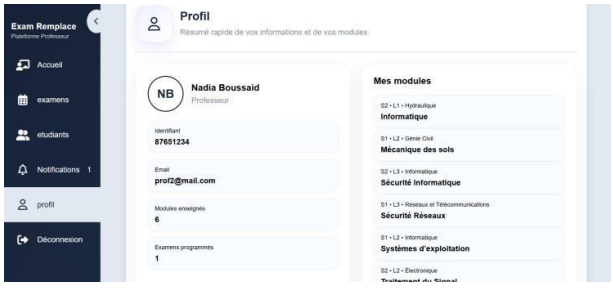


Figure 5.9 – Profil Enseignant

Conclusion :

Dans ce dernier chapitre, nous présentons l'environnement logiciel et les outils techniques utilisés pour développer notre application, puis nous exposons les différentes interfaces à travers des captures d'écran.

Conclusion générale

Notre projet de fin d'études avait pour objectif de concevoir et développer une application collaborative dédiée à la gestion des examens de remplacement au sein de l'Université M'hamed Bougara de Boumerdès, visant à faciliter l'organisation et le suivi des demandes tout en améliorant la communication entre les étudiants, les enseignants et l'administration.

Nous avons conçu une plateforme web centralisée, simple et intuitive, permettant aux étudiants de soumettre et suivre leurs demandes en ligne, à l'administration de les valider et d'organiser les examens, et aux enseignants de consulter les listes concernées. La conception s'est appuyée sur le langage UML à travers les diagrammes de cas d'utilisation, de séquence et de classes, en suivant une démarche structurée allant de l'analyse des besoins jusqu'au développement final.

Pour la réalisation, nous avons mobilisé des technologies web éprouvées HTML, CSS, JavaScript, PHP et MySQL , garantissant une application performante, sécurisée et évolutive. Ce projet nous a permis de renforcer nos compétences en développement web, en modélisation UML et en gestion de projet en équipe.

En perspective, l'application pourra être enrichie par la génération automatique des convocations, un système d'archivage des demandes, ainsi que l'amélioration des tableaux de bord statistiques. Nous espérons que ce travail contribuera à moderniser la gestion administrative des examens de remplacement et servira de base pour de futures évolutions.

Bibliographie

[1] Gangloff-Ziegler, C. (2009). *Les freins au travail collaboratif*. Marché et organisations, 10(3), 95-112. https://shs.cairn.info/revue-marche-et-organisations-2009-3-page-95?lang=fr

[2] Anis, B. (2012). *Conception et réalisation d'une plateforme pour le travail collaboratif cas : SADAREP Tizi-Ouzou* (Doctoral dissertation, Université Mouloud Mammeri). http://hasp.ummto.dz/bitstream/handle/ummto/1063/ANIS%20BOUALEM.pdf

[3] Chaumette, C., et Desbiens, D. (2008). *Typologie des systèmes d'information : Le travail collaboratif*. Université de Lille. http://observatoire.tic.free.fr/documents/travail_collaboratif.pdf

[4] Compain, G., Eynaud, P., Morel, L., et Vercher-Chaptal, C. (2019). *Les plateformes collaboratives : Éléments de caractérisation et stratégies de développement*. HAL SHS. https://shs.hal.science/halshs-02117564

[5] Roques, P., et Vallée, F. (2007). *UML 2 en action*. Éditions Eyrolles.

[6] Gabay, J., et Gabay, D. (2008). *UML 2 Analyse et conception : Mise en œuvre guidée avec études de cas*. Dunod.

[7] O'Keefe, G. (2006). Improving the Definition of UML. In *International Conference on Model Driven Engineering Languages and Systems* (pp. 42-56). Berlin, Heidelberg : Springer Berlin Heidelberg.

[8] Vinet, C. (2023). *Dictionnaire de données du SINP V1.0*. PatriNat (OFB-MNHN-CNRS-IRD).

[9] ORACLE CORPORATION. (2024). *MySQL Documentation*. Disponible [Lien].

[10] Dvorski, D. D. (2007). *Installing, configuring, and developing with Xampp*. Skills Canada, 492. http://dalibor.dvorski.net

[11] Visual Studio Code. (2019). *Visual Studio Code*. Disponible [Lien].

[12] W3Schools. *HTML Tutorial*. Disponible [Lien] [Dernière consultation le 15 mars 2025].

[13] Duckett, J., & Schlüter, J. (2011). *HTML & CSS : Design and Build Websites*. New York : John Wiley & Sons.

[14] Heurtel, O. (2002). *PHP 8*. Éditions ENI.

[15] Sidik, B. (2011). *JavaScript*. Informatika.

[16] Carbonell, M. (2015). Phpmyadmin. *INTRODUCCION A LA CREACION DE BASES DE DATOS*. Disponible [Lien].