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 Informatiques
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 Plateforme Personnelle Académique pour les Enseignants.

Présenté par :
ALLIOUAT Hadjer
BOUDEGZDAME Nada Hafsa
CHAOUCHE Imen
SI ZIANI Serine

Stage Pratique réalisé au : /

Soutenu le 04/06/2026 Devant le jury composé de

HADJIDJ Drifa : Examinatrice
DJOUZI Kheyreddine : Encadreur

Résumé

Ce projet présente la conception et la réalisation d'une plateforme académique centralisée dédiée aux enseignants et aux étudiants. Dans un contexte de transformation numérique, la dispersion des outils pédagogiques et professionnels entrave la visibilité des travaux académiques et complexifie l'accès au savoir. Notre solution vise à unifier l'identité numérique de l'enseignant et la gestion pédagogique des cours au sein d'un espace unique, intuitif et sécurisé. Cette plateforme permet notamment la gestion structurée des ressources, le suivi des activités académiques, la valorisation des travaux de recherche ainsi qu'une communication fluide entre les différents acteurs. Le projet repose sur une architecture modulaire et évolutive, garantissant une gestion optimale des données et une amélioration significative de l'expérience utilisateur.

Mots-clés : Plateforme académique, Enseignant, Apprenant, Cours, Formation, CV.

Abstract

This project presents the design and development of a centralized academic platform dedicated to teachers and students. In the context of digital transformation, the dispersion of pedagogical and professional tools hinders the visibility of academic work and complicates access to knowledge. Our solution aims to unify the teacher's digital identity and pedagogical course management within a unique, intuitive, and secure space. This platform enables structured resource management, tracking of academic activities, showcasing of research work, and fluid communication between the various stakeholders. The project is based on a modular and scalable architecture, ensuring optimal data management and a significant improvement in user experience.

Keywords : Academic platform, Teacher, Learner, Course, Training, CV.

Avant-propos

Ce mémoire est réalisé dans le cadre de l'obtention du **Diplôme de Licence Académique** en Informatique, spécialité Systèmes Informatiques, au Département d'Informatique de la Faculté des Sciences de l'Université M'hamed Bougara de Boumerdès, année académique **2025-2026**.

Ce projet est né d'un constat concret : les enseignants universitaires manquent d'un espace numérique unifié pour gérer et valoriser leurs activités pédagogiques et académiques. C'est dans ce contexte que l'idée d'une plateforme personnelle académique centralisée a pris forme, s'appuyant sur des technologies modernes telles que Node.js, PostgreSQL, HTML5, CSS et JavaScript.

Ce mémoire s'articule autour de trois chapitres : le premier dresse un état de l'art des solutions existantes et pose la problématique ; le second couvre l'analyse des besoins et la conception UML du système ; le troisième présente l'environnement technique et la réalisation de la plateforme.

Nous tenons à exprimer notre profonde gratitude à notre encadreur **M. DJOUZI Kheyreddine** pour ses précieux conseils et son soutien, ainsi qu'à **Mme. HADJIDJ Drifa**, examinatrice, pour l'intérêt porté à notre travail. Nos remerciements vont également à l'ensemble du corps enseignant du Département d'Informatique pour la qualité de la formation dispensée.

Enfin, nous dédions ce travail à nos familles dont le soutien a été une source de motivation constante.

DÉDICACE

Nous dédions ce modeste travail à nous quatre, pour les efforts fournis, la persévérance, l'esprit d'équipe et tous les moments partagés durant la réalisation de ce projet de fin d'études.

Nous exprimons ensuite notre profonde gratitude à nos chers parents pour leur amour, leurs sacrifices, leurs encouragements et leur soutien inconditionnel tout au long de notre parcours universitaire.

Nos sincères remerciements vont également à nos familles, nos amis, ainsi qu'à tous ceux qui nous ont soutenus et motivés durant ces années d'études.

Nous remercions chaleureusement nos enseignants et particulièrement notre encadreur pour leurs précieux conseils, leur patience et leur accompagnement dans la réalisation de ce travail.

Enfin, à toutes les personnes ayant contribué, de près ou de loin, à la réussite de ce projet, nous adressons nos plus sincères remerciements.

Hadjer, Nada Hafsa, Imen, Serine

Table des matières

Avant-propos **ii**

Liste des abréviations **ix**

Introduction Générale **1**

I Contexte Général et État de l'Art **2**

I.1 Introduction 3

I.2 Contexte Général 3

I.2.A L'enseignant à l'ère du numérique 3

I.2.B Besoin de présence professionnelle (portfolio) 4

I.2.C Nécessité de gérer la pédagogie (plateforme académique) 4

I.2.D Évolution vers des solutions intégrées 4

I.3 État de l'art 4

I.4 Problématique 6

I.5 Suggestion des solutions futures 6

I.6 Conclusion 7

II Analyse et conception **8**

II.1 Introduction 9

II.2 Modélisation 9

II.2.A Définition de la modélisation 9
II.2.B Intérêt de la modélisation 9
II.2.C Présentation de quelques méthodes de modélisation 9
II.2.D Démarche de modélisation adoptée 10
II.3 Analyse 11
II.3.A Identification des acteurs 11
II.3.B Spécification des besoins 12
II.3.B.a Besoins Fonctionnels 12
II.3.B.b Besoins Non Fonctionnels 12
II.3.C Spécification des tâches 13
II.3.D Diagramme de cas d'utilisation 14
II.4 Conception 16
II.4.A Diagramme de Séquences (DS) 16
II.4.A.a DS : "S'authentifier" en tant qu'apprenant 16
II.4.A.b DS : "Modifier une publication" en tant qu'enseignant 17
II.4.A.c DS : "Télécharger les ressources publiques" en tant que visiteur 17
II.4.B Diagramme d'activités 18
II.4.B.a DA : "Envoyer un message" en tant qu'enseignant . . . 18
II.4.B.b DA : "S'inscrire" en tant que visiteur 19
II.4.C Diagramme de classe 19
II.4.D Diagramme de paquetages 21
II.4.E Diagramme de déploiement 21
II.5 Conclusion 22

III Réalisation **23**

III.1 Introduction 24
III.2 Présentation de l'Environnement Logiciel Utilisé 24

III.2.A Outils et Logiciels de Conception et de Développement 24
III.2.A.a Outils de Conception des Diagrammes 24
III.2.A.b Outils de Design d'Interfaces Graphiques 24
III.2.A.c Éditeur de Code 24
III.2.B Langages de Programmation et Frameworks Utilisés 24
III.2.B.a Backend 25
III.2.B.b Frontend 25
III.3 Présentation de l'environnement matériel utilisé 26
III.4 Conclusion 26

Conclusion Générale **27**

Table des figures

II.1 Démarche de modélisation adoptée 11
II.2 Diagramme de contexte de notre système 12
II.3 Diagramme de cas d'utilisation pour les acteurs "Visiteur" et "Apprenant" 14
II.4 Diagramme de cas d'utilisation pour l'acteur "Enseignant" 15
II.5 DS pour le cas d'utilisation "S'authentifier" en tant qu'apprenant . . . 16
II.6 DS pour le cas d'utilisation "Modifier une publication" en tant qu'enseignant 17
II.7 DS pour le cas "Télécharger les ressources publiques" en tant que visiteur 17
II.8 Diagramme d'activités pour le cas "Envoyer un message" en tant qu'enseignant 18
II.9 Diagramme d'activités pour le cas "S'inscrire" en tant que visiteur . . . 19
II.10 Diagramme de classe de notre système 20
II.11 Diagramme de paquetages 21
II.12 Diagramme de déploiement 22

Liste des tableaux

I.1 Étude comparative des solutions de portfolios existantes 5

II.1 Étude comparative des méthodes de modélisation 10

II.2 Répartition des tâches selon les utilisateurs du système 13

Liste des abréviations

BPMN	Business Process Model and Notation
CSS	Cascading Style Sheet
DA	Diagramme d'Activité
DC	Diagramme de Classe
DS	Diagramme de Sequence
HTML	Hyper Text Markup Language
LMS	Learning Management Software
UML	Unified Modeling Language
VS	Visual Studio

Introduction Générale

L'intégration des technologies de l'information et de la communication a radicalement transformé le paysage de l'enseignement supérieur. Le rôle de l'enseignant, autrefois centré sur la transmission du savoir, s'est élargi vers des missions d'ingénierie pédagogique et de médiation numérique. Toutefois, cette mutation s'accompagne d'une "dispersion numérique" : l'enseignant doit naviguer entre une multitude d'outils disparates pour gérer ses enseignements, ses publications et ses interactions, ce qui complexifie la visibilité de ses travaux et l'accès au contenu pour les étudiants.

Face à ces défis, il est impératif de moderniser l'organisation académique en s'appuyant sur des solutions intégrées. L'objectif de ce projet est de concevoir et de réaliser une plateforme académique centralisée. Cette plateforme vise à unifier les volets pédagogiques et professionnels, permettant ainsi une gestion homogène des activités de l'enseignant tout en offrant aux étudiants un point d'accès unique et intuitif au savoir.

Ce manuscrit est organisé en trois chapitres :

- **Chapitre I - Contexte Général et État de l'Art :** Ce chapitre analyse les limites des outils actuels et justifie la nécessité d'une centralisation des activités académiques.
- **Chapitre II - Analyse et Conception :** Nous y aborderons l'identification des besoins des utilisateurs, la modélisation des données et la conception de l'architecture du système via UML.
- **Chapitre III - Réalisation :** Ce chapitre présente les choix technologiques et les étapes de développement de la solution.

Enfin, le manuscrit se termine par une conclusion générale et une discussion sur les perspectives d'évolution de la plateforme.

Chapitre I

Contexte Général et État de l'Art

Résumé : *Ce chapitre explore la transformation numérique de l'enseignement, analysant le passage du rôle traditionnel de l'enseignant vers une posture d'intermédiaire numérique. Il dresse un état de l'art comparatif des solutions de portfolios existantes et pose la problématique de la centralisation des outils pédagogiques.*

Sommaire

I.1 Introduction **3**
I.2 Contexte Général **3**
I.3 État de l'art **4**
I.4 Problématique **6**
I.5 Suggestion des solutions futures **6**
I.6 Conclusion **7**

I.1 Introduction

Le paysage éducatif mondial traverse une transformation sans précédent, impulsée par l'omniprésence des technologies de l'information et de la communication (TIC). Ce premier chapitre vise à poser les jalons de notre réflexion en explorant le contexte global dans lequel évolue l'enseignement moderne. Nous aborderons d'abord les mutations du rôle de l'enseignant face au numérique, avant d'analyser les besoins croissants de visibilité professionnelle et de gestion pédagogique centralisée.

I.2 Contexte Général

L'évolution numérique redéfinit les frontières de l'espace académique classique.

I.2.A L'enseignant à l'ère du numérique

L'intégration du numérique dans le domaine éducatif ne se limite plus à l'utilisation sporadique d'outils, elle transforme profondément la définition même de la fonction de l'enseignant. À l'heure actuelle, l'enseignant s'est transformé en un intermédiaire devant naviguer entre la pédagogie traditionnelle et les environnements d'apprentissage virtuels. Cette évolution représente le changement de rôle d'un simple transmetteur de connaissances à celui de facilitateur et de médiateur. Dans un univers où l'information se trouve à portée de clic, le rôle de l'enseignant s'articule désormais autour de sa capacité à orienter l'élève dans la profusion d'informations disponibles, tout en cultivant chez lui une pensée critique et une compétence numérique robuste.

Le professeur contemporain est également tenu d'endosser le rôle d'ingénieur pédagogique. Il ne se limite plus à fournir du contenu, mais élabore des trajectoires d'apprentissage hybrides, combinant interactions face à face et ressources numériques. Cette combinaison nécessite une compétence technique de plus en plus pointue et une flexibilité continue face aux changements rapides des logiciels et des plateformes. Toutefois, la place croissante qu'occupe la technologie dans le quotidien de l'enseignant conduit à une fragmentation des tâches. Souvent, l'enseignant doit naviguer entre divers outils pour gérer les notes, stocker les fichiers et communiquer, ce qui engendre une lourde charge cognitive et administrative.

I.2.B Besoin de présence professionnelle (portfolio)

À l'ère du numérique, le profil d'un professeur dépasse la simple présentation de son Curriculum Vitae sur papier. Il est important d'avoir une présence en ligne pour montrer ses compétences et ses recherches via un portfolio numérique. Plus qu'une simple collection de documents, c'est un outil de pensée critique. Il permet à l'enseignant d'analyser son évolution pédagogique au fil du temps. En notant ses succès et ses tentatives, il transforme son expérience en savoir accessible.

Créer un portfolio sur sa propre plateforme est important pour la propriété intellectuelle et la durabilité. Contrairement aux réseaux sociaux ou aux serveurs temporaires, un espace personnel assure le contrôle des ressources et de leur diffusion. Cette présence numérique favorise les échanges avec des chercheurs du monde entier. Enfin, dans un milieu universitaire en constante évolution, ce portfolio assure une continuité de carrière en gardant une trace de son parcours, peu importe les changements d'établissements [1].

I.2.C Nécessité de gérer la pédagogie (plateforme académique)

L'établissement éducatif doit satisfaire une nécessité de structuration. L'usage d'une plateforme académique (LMS) devient essentiel pour regrouper les ressources et suivre le parcours des apprenants.

I.2.D Évolution vers des solutions intégrées

L'observation actuelle révèle une dissociation des outils. L'adoption de solutions unifiées reflète la combinaison de ces exigences, associant la gestion des identités professionnelles et la gestion pédagogique dans un environnement unique.

I.3 État de l'art

Dans cette section, nous analysons différentes solutions existantes afin d'identifier leurs forces, leurs limites et leurs descriptions.

- **M-MAMCHA et M-AISLI (Google Sites) :** Ces portfolios exploitent la flexibilité de l'outil Google Sites. M-MAMCHA se concentre sur une interface adaptable qui rend l'accès aux ressources pédagogiques plus simple. M-AISLI, lui, offre une structure plus remplie avec des options de téléchargement. Ces exemples montrent comment les outils No-Code peuvent servir à créer des portfolios numé-

riques pour aider à l'auto-évaluation [2][3].

- **A. Chambert-LOIR et Emanuel-RUSS (Sites Web personnels) :** Ces plateformes montrent l'approche du site web dédié. Le portfolio de A. Chambert-LOIR est divisé en trois parties pour aider l'utilisateur à s'y retrouver. Emanuel-RUSS préfère une navigation par recherche. Ces organisations montrent pourquoi l'architecture de l'information est importante pour présenter ses compétences professionnelles [4][5].
- **Moodle :** Ce système de gestion de l'apprentissage (LMS) open source est très modulable pour l'enseignement à distance. On peut le personnaliser facilement, ce qui en fait une référence mondiale pour ajouter des portfolios de type Mahara ou des portfolios intégrés [6].
- **Google Classroom :** Cette solution SaaS (Software as a Service) est pensée pour fonctionner avec Google Workspace. Elle rend la gestion des tâches pédagogiques plus facile en temps réel, mais elle offre moins de possibili]tés de personnalisation graphique qu'un site web dédié [7].

Le tableau I.1 présente un comparaison entre les différents portfolios étudies

Portfolios	**Avantages**	**Inconvénients**	**Type**
M-MAMCHA	Accès direct aux cours, ergonomie mobile.	Infos superflues.	Google Site
M-AISLI	Présentation complète, téléchargements.	Design daté, surchargé.	Google Site
A. Chambert-LOIR	Structure claire en 3 sections.	Trop textuel, non mobile.	Site Web
Emanuel-RUSS	Structure sobre, barre de recherche.	Pas de contact direct.	Site Web
Moodle	Très personnalisable, mondial.	Dépendance à la connexion.	Open Source
Google Classroom	Simple, intégré à Workspace.	Peu de fonctions avancées.	SaaS

TABLE I.1 – Étude comparative des solutions de portfolios existantes

I.4 Problématique

L'un des obstacles majeurs de l'écosystème universitaire actuel est la dispersion numérique. L'utilisation de plateformes disparates (Google Sites, blogs, sites personnels) crée une rupture dans l'accès au savoir et engendre trois (03) obstacles majeurs :

- **Visibilité limitée :** L'absence de profil rend l'expertise et les recherches des enseignants difficiles à identifier.
- **Accès complexe :** La multiplication des sites personnels fait perdre un temps précieux aux étudiants pour localiser les cours.
- **Navigation confuse :** L'incohérence des interfaces (UX) entre les différents sites désoriente les utilisateurs.

I.5 Suggestion des solutions futures

Pour rendre les ressources des enseignants plus accessibles, nous retenons deux (02) solutions principales que nous allons détailler :

- **Réseaux sociaux professionnels :** L'utilisation de plateformes mondiales existantes comme LinkedIn et ResearchGate présente une visibilité internationale, des outils de réseautage déjà en place et la gratuité pour l'utilisateur. Mais cela reste insuffisant pour le contexte pédagogique spécifique, et les grands obstacles sont :
  - **Hétérogénéité des profils :** Le contenu académique est noyé parmi une multitude d'utilisateurs et de publications non pertinentes (publicités, profils commerciaux).
  - **Manque d'outils pédagogiques :** Ces plateformes ne sont pas conçues pour la gestion structurée des cours (absence de dossiers par module, de dépôt de supports de TD, etc.).
  - **Complexité de recherche :** Pour un étudiant, il est difficile de filtrer et de localiser rapidement une ressource spécifique au milieu d'un flux d'actualités global.
- **Une plateforme dédiée et centralisée :** La création d'un espace spécifique pour le corps enseignant est la solution la plus efficace. Elle permet de regrouper l'ensemble de l'écosystème académique en un seul point.
  - **Unification des activités :** Regrouper sur une même interface les activités pédagogiques (cours, TD, examens) et scientifiques (publications, domaines de recherche, conférences).

- **Identité numérique standardisée :** Chaque enseignant possède un profil structuré de la même manière, facilitant la lecture pour les visiteurs.
- **Accessibilité et contact :** Centraliser les coordonnées et les heures de réception pour rendre la communication entre étudiants et enseignants fluide et directe.
- **Zéro maintenance technique :** L'enseignant n'a plus à se soucier de l'hébergement ou du design, il se concentre uniquement sur son contenu.

I.6 Conclusion

Ce chapitre a posé les bases conceptuelles de notre projet en analysant l'évolution du rôle de l'enseignant à l'ère du numérique et en comparant les solutions de portfolios existantes. Cette étude comparative a mis en lumière les limites des outils actuels, marqués par une dispersion numérique qui nuit à la visibilité professionnelle et à l'accessibilité pédagogique. Ces constats justifient la nécessité d'une plateforme académique centralisée, capable de réunir en un seul espace l'identité numérique de l'enseignant et la gestion de ses ressources pédagogiques. Le chapitre suivant sera consacré à l'analyse et la conception de cette solution.

Chapitre II

Analyse et conception

Résumé : *Ce chapitre est consacré à l'analyse et à la conception de notre plateforme académique. Après avoir présenté les principales méthodes de modélisation (UML, Merise, BPMN) et justifié le choix d'UML, nous avons identifié les trois acteurs du système (Visiteur, Apprenant et Enseignant) et défini leurs besoins fonctionnels et non fonctionnels. Cette analyse a abouti à l'élaboration des diagrammes de cas d'utilisation, de séquence, d'activités et de classes, constituant ainsi une architecture logique et cohérente prête pour la phase d'implémentation.*

Sommaire

II.1 Introduction **9**
II.2 Modélisation **9**
II.3 Analyse **11**
II.4 Conception **16**
II.5 Conclusion **22**

II.1 Introduction

Le développement d'un système informatique passe par l'analyse et la conception. Elles permettent de mieux comprendre les besoins des utilisateurs, de classer les informations et de construire une architecture logique et cohérente avant de passer à la phase d'implémentation.

II.2 Modélisation

Avant de procéder à l'implémentation de notre application, il est indispensable d'en concevoir l'architecture. Nous présentons ici les fondements de la modélisation et la démarche adoptée pour ce projet.

II.2.A Définition de la modélisation

La modélisation est un processus qui consiste à représenter un système réel ou abstrait à l'aide de schémas, de diagrammes ou de modèles conceptuels. Elle permet de simplifier la réalité tout en conservant les informations essentielles nécessaires à la compréhension, à l'analyse et à la conception du système [8].

II.2.B Intérêt de la modélisation

La modélisation est essentielle pour tout projet en informatique. Elle permet une meilleure compréhension du système, renforce la communication entre les divers intervenants, diminue les incertitudes et constitue un socle solide pour les étapes de conception et de développement [9].

II.2.C Présentation de quelques méthodes de modélisation

Il existe plusieurs méthodes de modélisation comme l'UML, MERISE et le BPMN.

1. **UML :** UML (Unified Modeling Language) est un langage de modélisation graphique utilisé pour illustrer les systèmes logiciels à divers niveaux. Conçu dans les années 1990 par Booch, Rumbaugh et Jacobson, il est aujourd'hui la norme pour la conception des systèmes d'information [10].
2. **Merise :** Merise est une méthode française élaborée au début des années 1980, s'appuyant sur la distinction entre données et traitements à travers plusieurs niveaux de modélisation (conceptuel, logique et physique) [11].

3. **BPMN :** BPMN (Business Process Model and Notation) est une notation développée au début des années 2000 pour illustrer visuellement les processus métier, les flux d'informations et les interactions entre participants [12].

Le tableau II.1 présente une comparaison entre ces méthodes.

Critère	**UML**	**Merise**	**BPMN**
Type	Langage orienté objet	Modélisation des SI	Modélisation des processus
Objectif	Modéliser les systèmes logiciels	Concevoir les bases de données	Décrire les processus métier
Apparition	Années 1990	Début des années 1980	Début des années 2000
Diagrammes	Classes, séquence, activité...	MCD, MLD, MPD	Processus métier
Domaine	Développement logiciel	Bases de données	Gestion des processus
Avantages	Standard international	Bonne structuration	Très visuel

TABLE II.1 – Étude comparative des méthodes de modélisation

II.2.D Démarche de modélisation adoptée

Pour concevoir notre plateforme, nous avons adopté UML en raison de sa standardisation et de sa capacité à représenter des systèmes web complexes intégrant diverses interactions. La figure II.1 illustre notre démarche de modélisation.

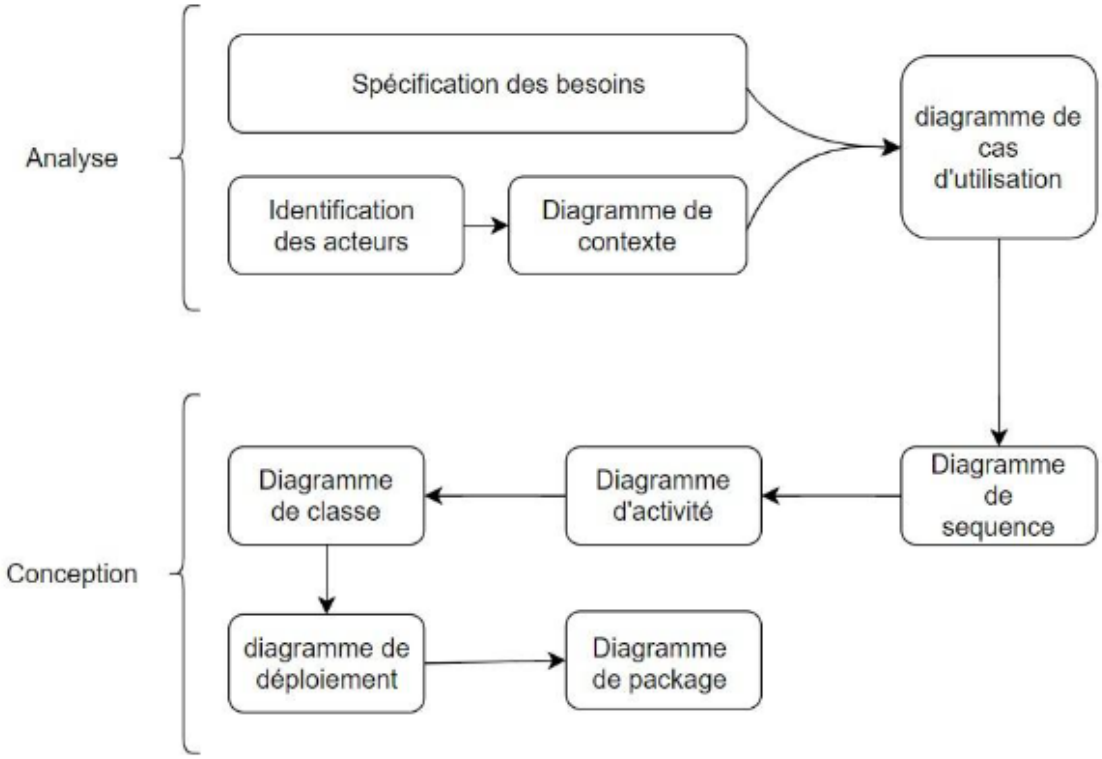


FIGURE II.1 – Démarche de modélisation adoptée

II.3 Analyse

Dans cette phase, les besoins seront clarifiés afin de comprendre les attentes des enseignants, identifier les parties prenantes et construire les diagrammes de cas d'utilisation définissant les fonctionnalités principales du système.

II.3.A Identification des acteurs

Nous avons identifié trois acteurs interagissant avec le système :

- **Visiteur :** Utilisateur anonyme dont les interactions sont limitées à la consultation et à l'inscription initiale.
- **Apprenant :** Utilisateur disposant d'un espace personnel sécurisé après authentification pour suivre les formations.
- **Enseignant :** Acteur principal gérant le contenu pédagogique et les étudiants avec des privilèges élevés.

La figure II.2 illustre le diagramme de contexte de notre système.



FIGURE II.2 – Diagramme de contexte de notre système

II.3.B Spécification des besoins

II.3.B.a Besoins Fonctionnels

Les besoins fonctionnels décrivent les services que le système doit fournir :

1. **Enseignant :** Inscription, authentification, gestion du profil, du CV, des manifestations et publications scientifiques, des cours, examens, formations, apprenants, messagerie et annonces.
2. **Visiteur :** Accès général, consultation des portfolios, annonces, ressources pédagogiques et inscription.
3. **Apprenant :** Authentification, gestion du profil, consultation et inscription aux formations, suivi, téléchargement des ressources et abandon de formation.

II.3.B.b Besoins Non Fonctionnels

Les besoins non fonctionnels définissent les contraintes techniques et les caractéristiques de qualité du système :

- **Sécurité :** Protection des données et authentification fiable.
- **Performance :** Temps de réponse rapide et optimisation des ressources.
- **Disponibilité :** Accès permanent via Internet (24h/24, 7j/7).
- **Ergonomie :** Interface intuitive et facile à prendre en main.
- **Compatibilité :** Fonctionnement optimal sur les principaux navigateurs et dif-

férents écrans.
- **Maintenabilité :** Conception modulaire pour faciliter les mises à jour futures.

II.3.C Spécification des tâches

ID	Tâche	Visiteur	Apprenant	Enseignant
1	Accéder à la plateforme / S'authentifier	X	X	X
2	Consulter les annonces, le portfolio et les ressources	X	X	X
3	Gérer son profil		X	
4	Gérer ses formations		X	
5	Gérer la messagerie			X
6	Gérer les annonces			X
7	Gérer le volet pédagogique			X
8	Gérer le volet académique			X
9	Affecter un apprenant à une formation			X
10	S'inscrire à une formation		X	
11	Accéder, étudier et suivre une formation		X	
12	Télécharger les ressources		X	
13	Consulter les cours et les TD	X	X	
14	Consulter les examens	X	X	
15	Abandonner une formation		X	
16	Gérer les cours			X
17	Gérer les apprenants			X
18	Gérer les examens			X
19	Gérer l'espace Moodle			X
20	Gérer son CV			X
21	Gérer ses recherches			X
22	Gérer ses manifestations scientifiques (MS)			X
23	Gérer ses publications scientifiques (PS)			X

TABLE II.2 – Répartition des tâches selon les utilisateurs du système

II.3.D Diagramme de cas d'utilisation

Les figures II.3 et II.4 illustrent les diagrammes de cas d'utilisation de notre système.

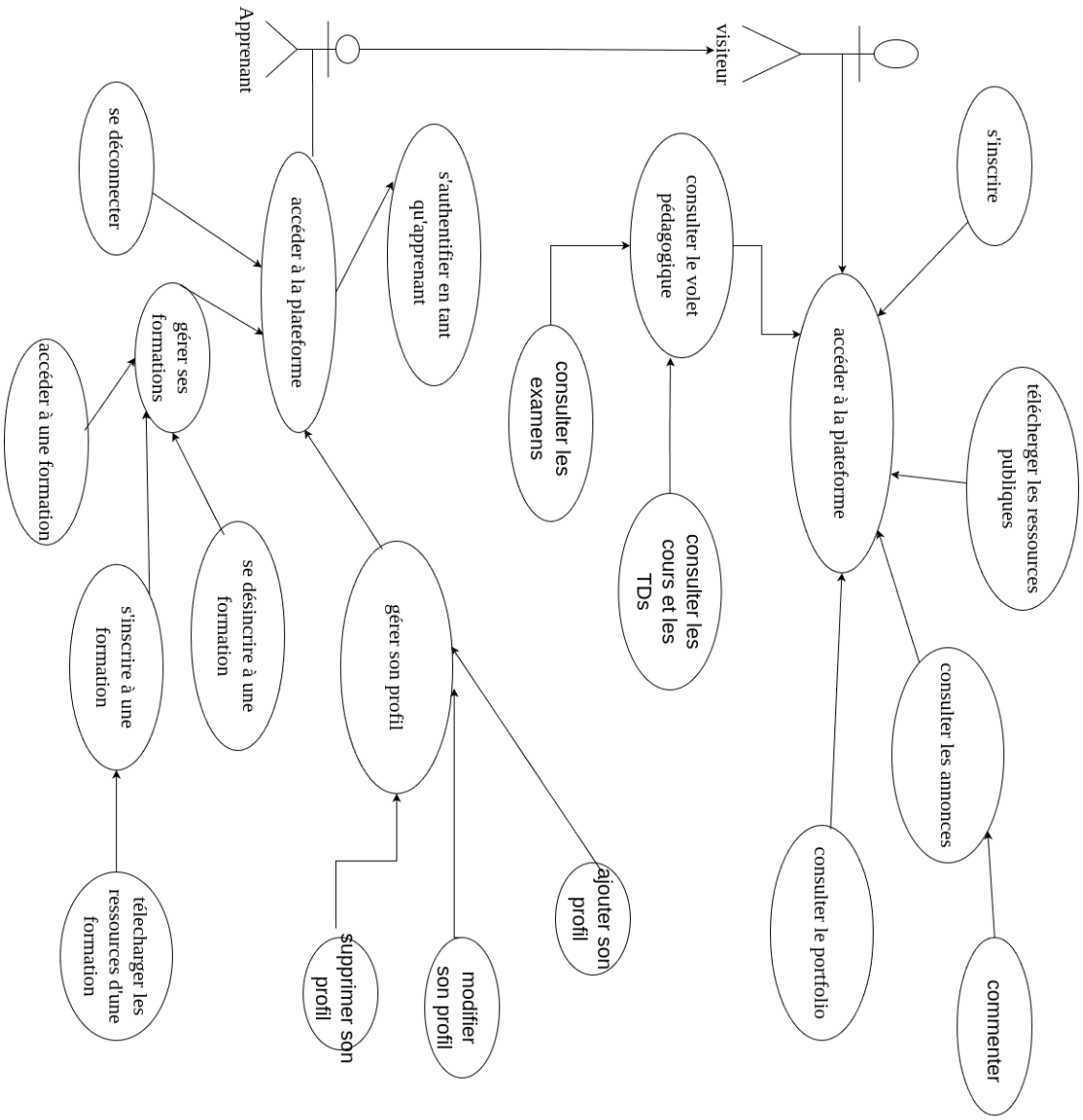


FIGURE II.3 – Diagramme de cas d'utilisation pour les acteurs "Visiteur" et "Apprenant"

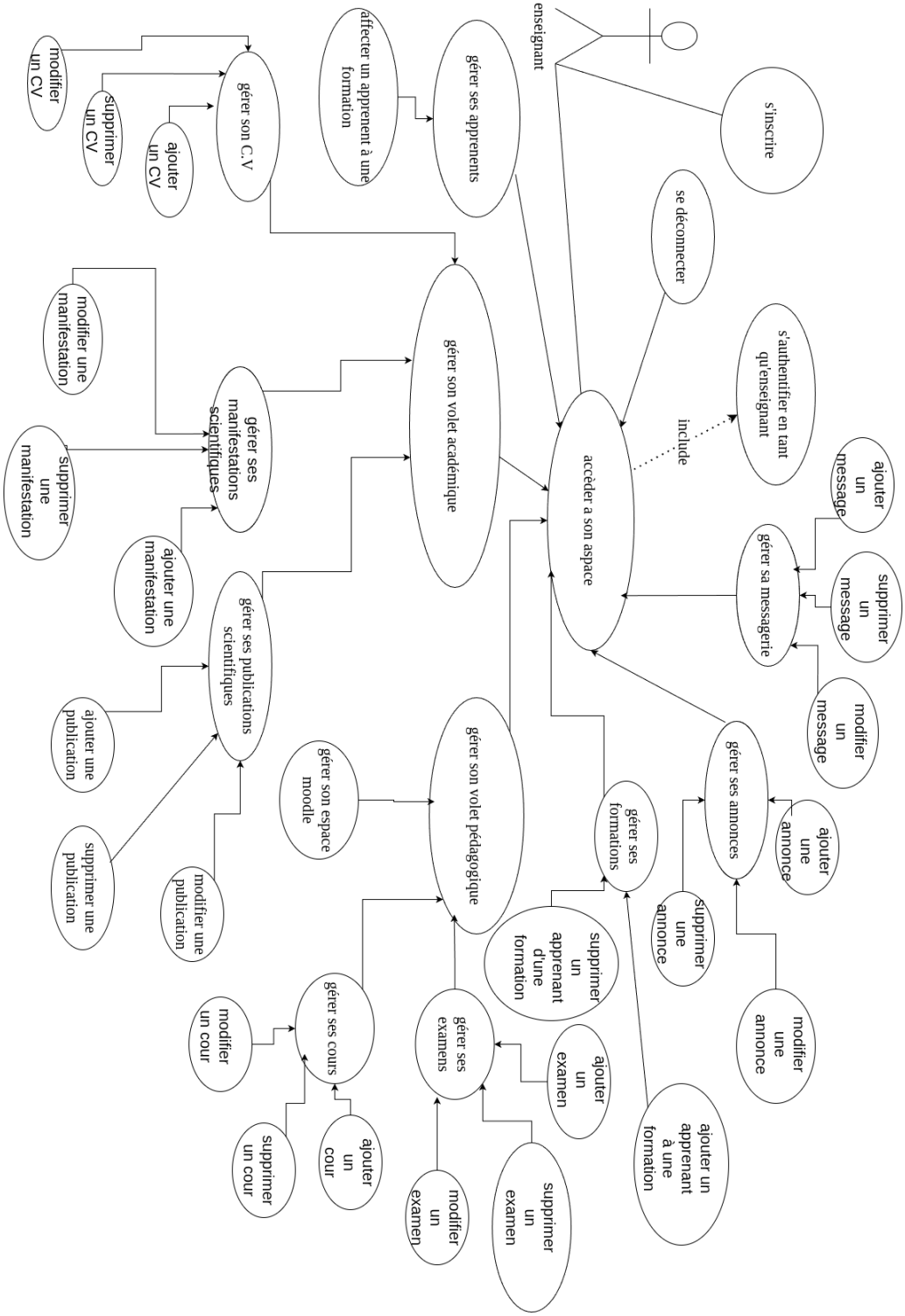


FIGURE II.4 – Diagramme de cas d'utilisation pour l'acteur "Enseignant"

II.4 Conception

Après la vérification des exigences, nous passerons à la conception technique à travers les diagrammes de séquence, d'activité et de classes.

II.4.A Diagramme de Séquences (DS)

Les diagrammes de séquence décrivent l'échange de messages entre les composants du système pour réaliser une fonctionnalité dans le temps.

II.4.A.a DS : "S'authentifier" en tant qu'apprenant

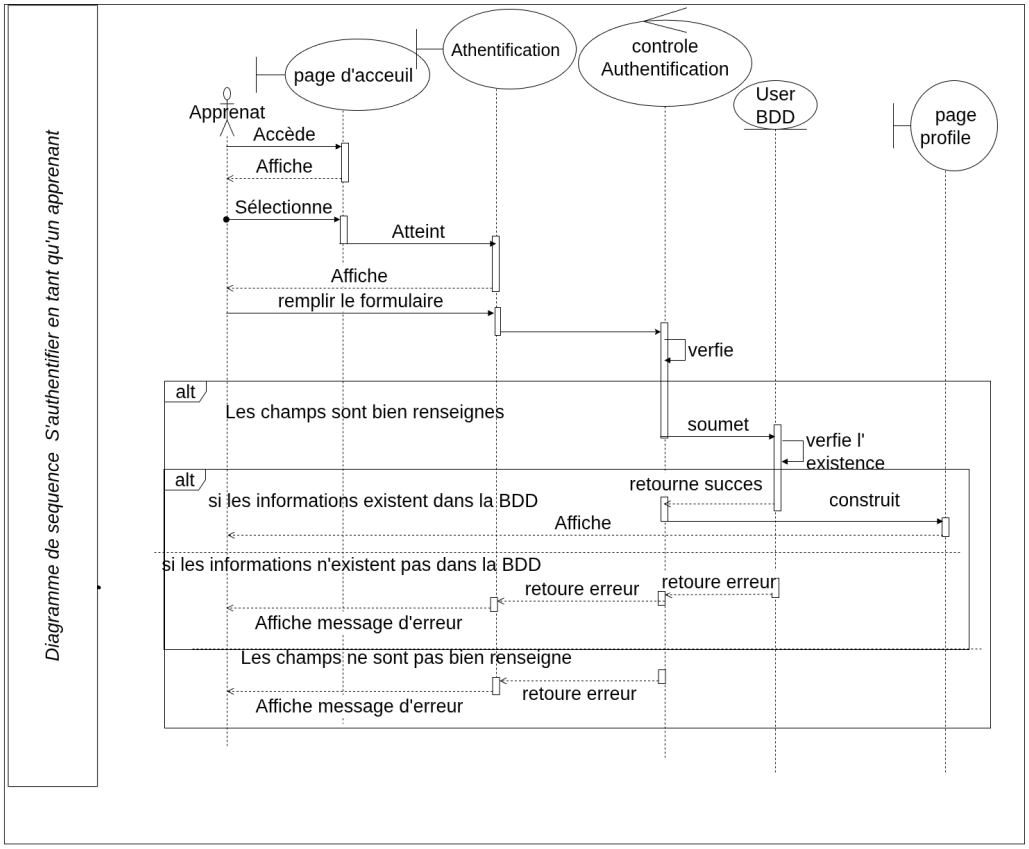


FIGURE II.5 – DS pour le cas d'utilisation "S'authentifier" en tant qu'apprenant

II.4.A.b DS : "Modifier une publication" en tant qu'enseignant

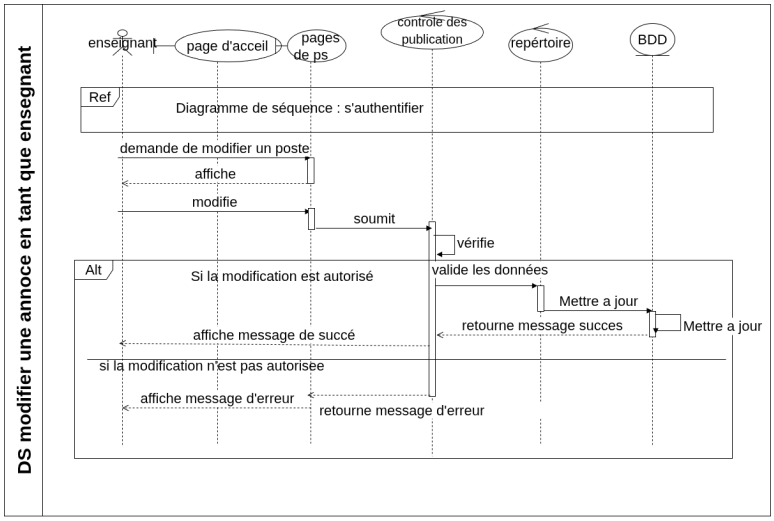


FIGURE II.6 – DS pour le cas d'utilisation "Modifier une publication" en tant qu'enseignant

II.4.A.c DS : "Télécharger les ressources publiques" en tant que visiteur

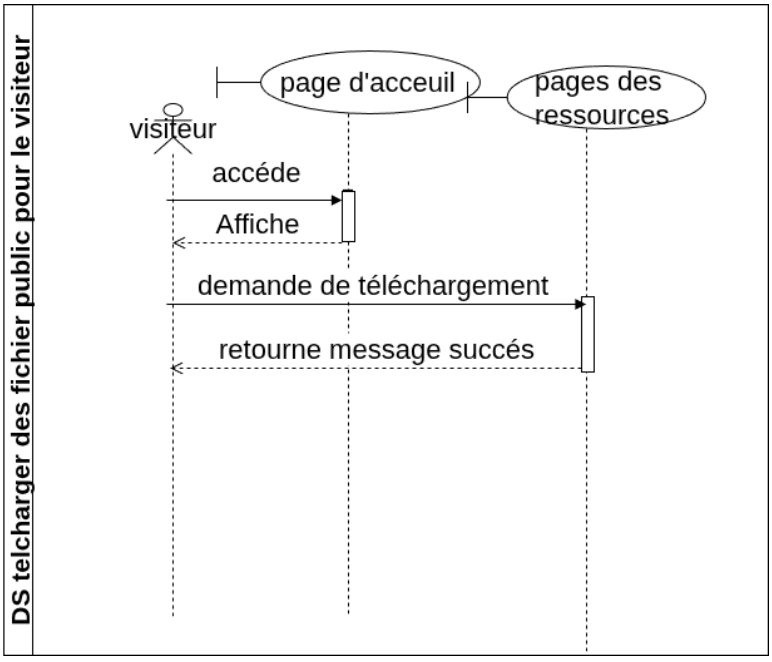


FIGURE II.7 – DS pour le cas "Télécharger les ressources publiques" en tant que visiteur

II.4.B Diagramme d'activités

Les diagrammes d'activités décrivent le déroulement logique des fonctionnalités de l'application et permettent de visualiser les étapes, les décisions et les actions effectuées par le système.

II.4.B.a DA : "Envoyer un message" en tant qu'enseignant

Ce diagramme décrit les phases que l'enseignant traverse pour communiquer sur la plateforme : saisie du message, validation des informations, envoi et archivage dans la base de données.

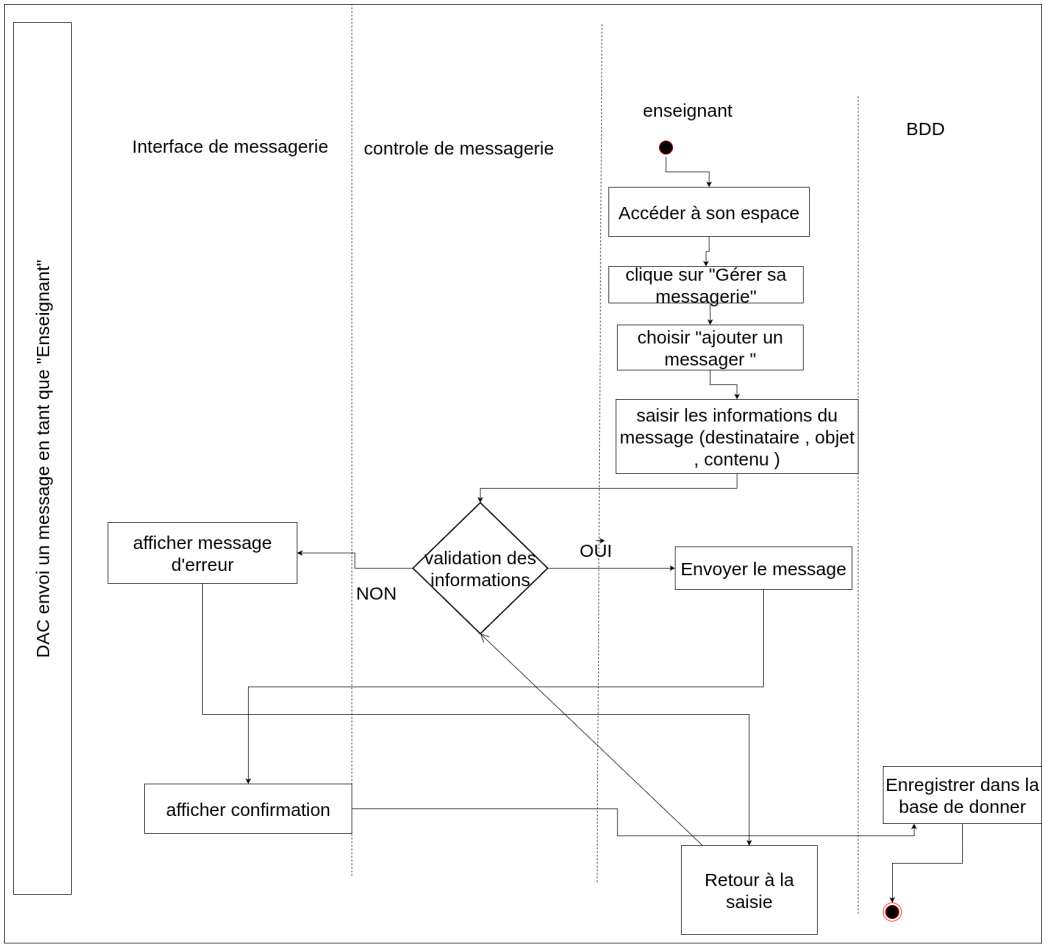


FIGURE II.8 – Diagramme d'activités pour le cas "Envoyer un message" en tant qu'enseignant

II.4.B.b DA : "S'inscrire" en tant que visiteur

Ce diagramme représente les étapes de création d'un compte : saisie et validation des informations, complétion du profil et enregistrement dans la base de données.

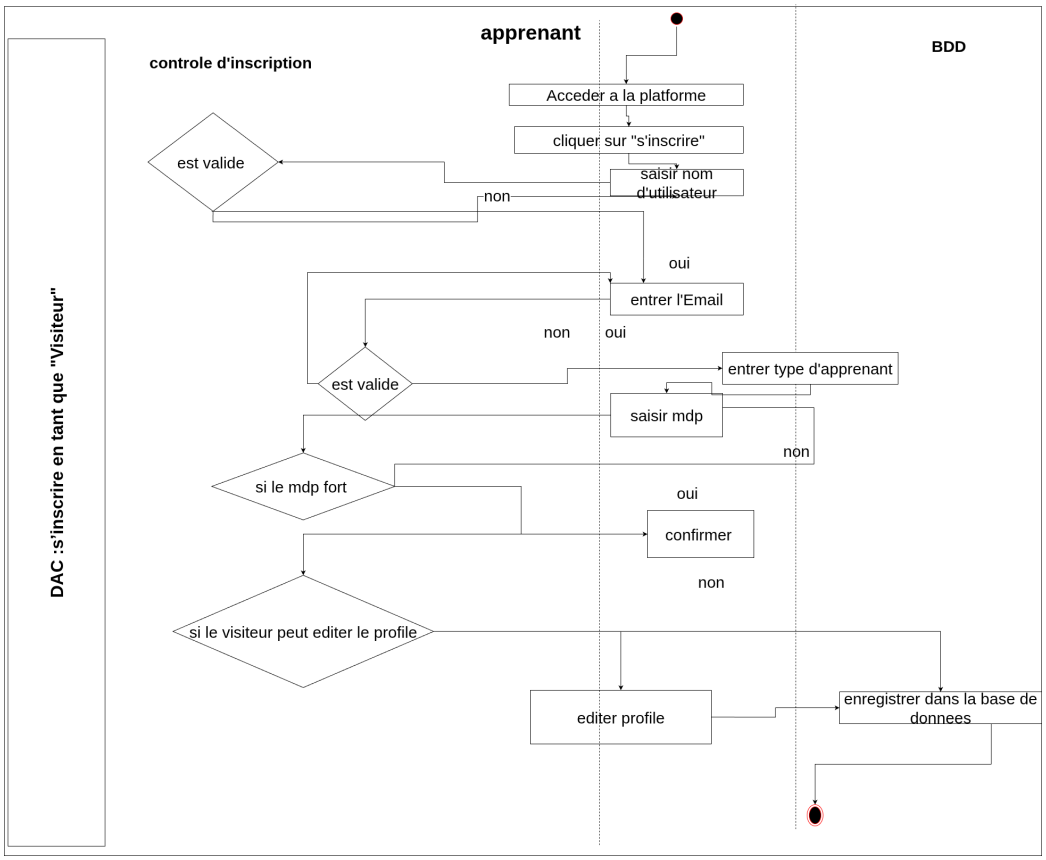


FIGURE II.9 – Diagramme d'activités pour le cas "S'inscrire" en tant que visiteur

II.4.C Diagramme de classe

Un diagramme de classe représente la structure statique d'un système informatique en illustrant les classes, leurs attributs, leurs méthodes et leurs relations. La figure II.10 illustre le diagramme de classe de notre système.

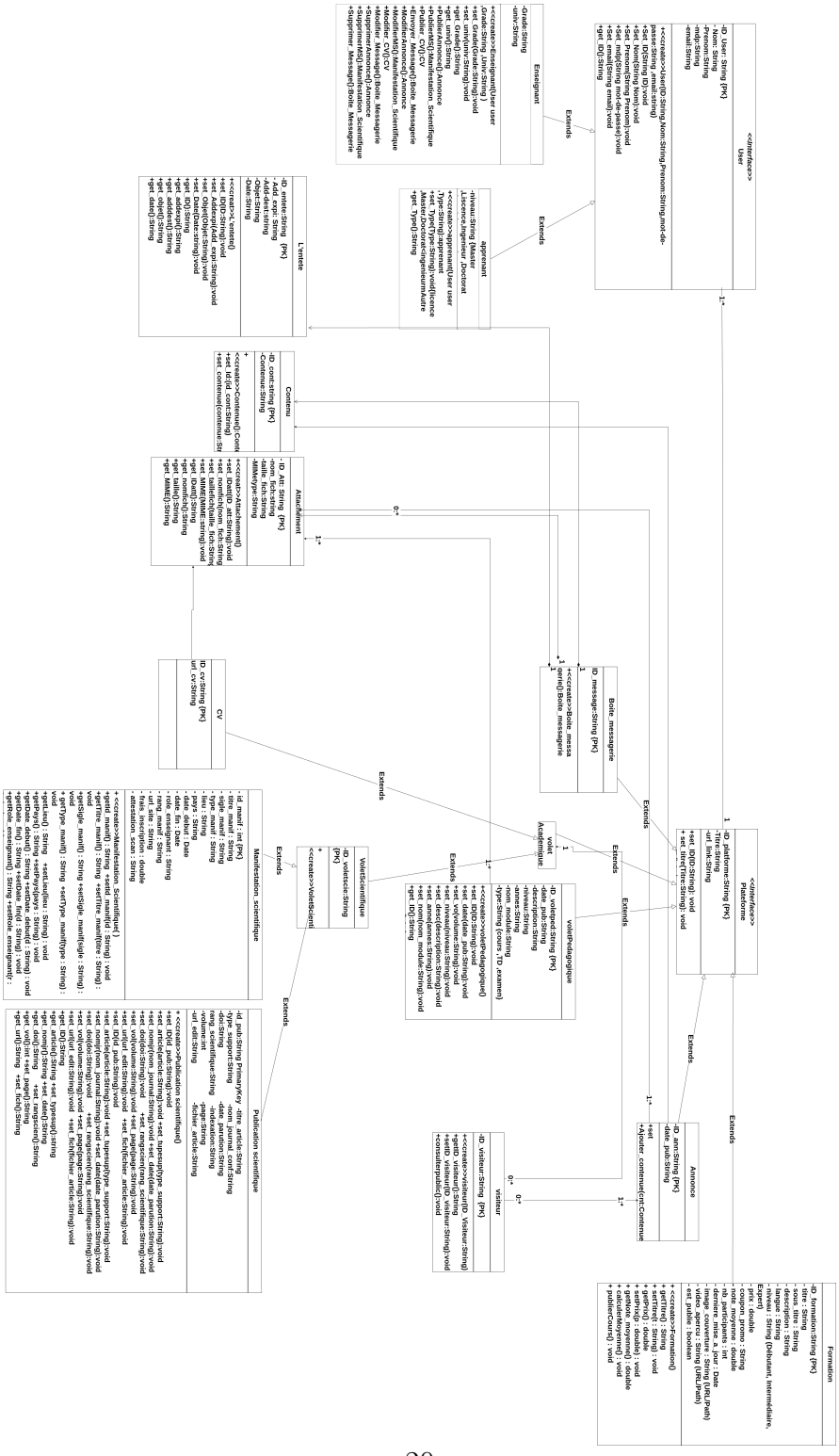


FIGURE II.10 – Diagramme de classe de notre système

II.4.D Diagramme de paquetages

Un diagramme de paquetages permet de structurer et d'organiser le système en regroupant les classes apparentées en modules logiques appelés paquetages. Il met en évidence la modularité de l'architecture ainsi que les relations de dépendance et d'importation entre les différents domaines métiers (Utilisateur, Plateforme et Académique). La figure II.11 illustre le diagramme de paquetages de notre système.

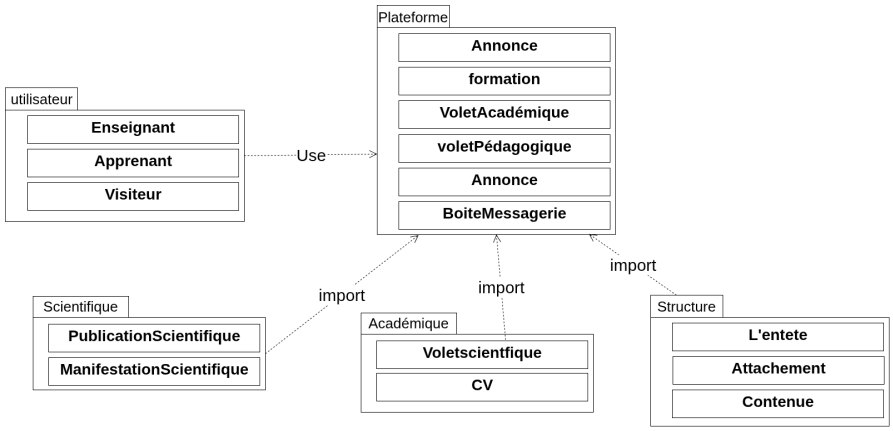


FIGURE II.11 – Diagramme de paquetages

II.4.E Diagramme de déploiement

Un diagramme de déploiement représente l'architecture physique du système en illustrant la disposition matérielle des nœuds (serveurs, terminaux clients), les composants logiciels qui y sont exécutés, ainsi que les protocoles de communication qui les relient. La figure II.12 illustre le diagramme de déploiement de notre système.

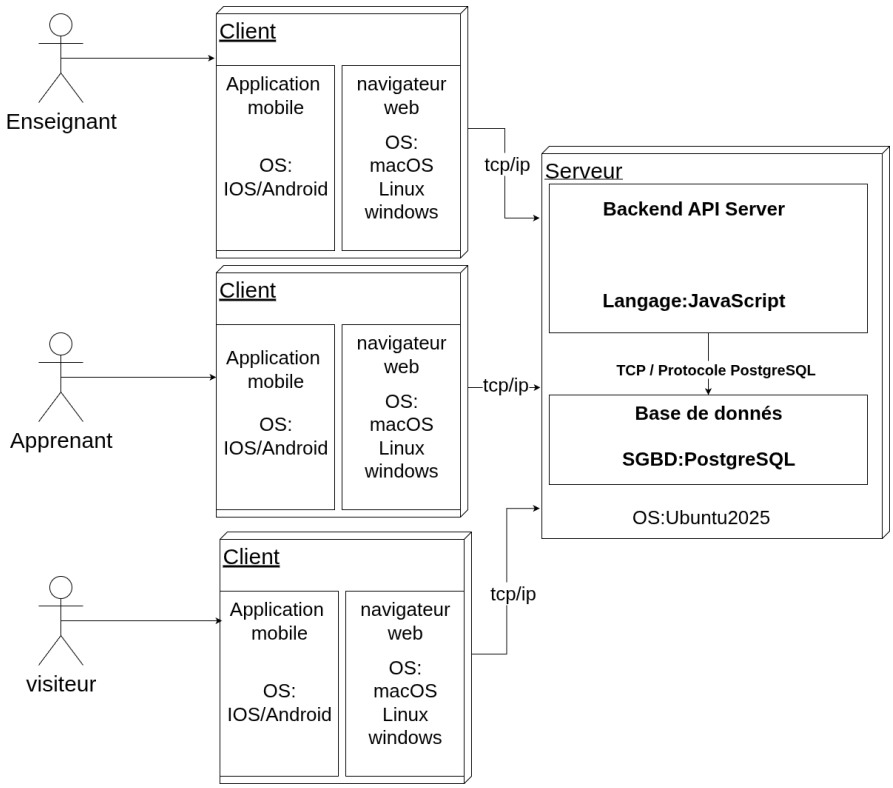


FIGURE II.12 – Diagramme de déploiement

II.5 Conclusion

Ce chapitre a posé les fondements analytiques et conceptuels de notre plateforme académique. L'étude comparative des méthodes de modélisation nous a conduits à adopter UML comme langage de référence. L'identification des trois acteurs principaux et la spécification de leurs besoins ont été formalisées à travers une série de diagrammes UML couvrant les interactions, les flux d'activités et la structure statique des données, constituant une base solide pour la phase de réalisation.

Chapitre III

Réalisation

Résumé : *Ce chapitre est consacré à la phase de réalisation de notre plateforme académique. Nous présentons d'abord l'environnement logiciel utilisé, comprenant les outils de conception et de développement, ainsi que les langages de programmation et frameworks adoptés. Nous décrivons ensuite l'environnement matériel ayant supporté ces travaux. Enfin, nous illustrons les interfaces graphiques finales de la plateforme, concrétisant les fonctionnalités développées pour les différents profils utilisateurs.*

Sommaire

III.1 Introduction **24**
III.2 Présentation de l'Environnement Logiciel Utilisé **24**
III.3 Présentation de l'environnement matériel utilisé **26**
III.4 Conclusion **26**

III.1 Introduction

Après avoir franchi les étapes cruciales de l'analyse des besoins et de la conception de l'architecture du système, ce troisième chapitre marque le passage à la phase de mise en œuvre concrète de la plateforme académique. Nous présenterons l'environnement logiciel de développement, incluant les outils de conception, les éditeurs de code, ainsi que les langages et frameworks utilisés, puis nous décrirons l'environnement matériel mobilisé.

III.2 Présentation de l'Environnement Logiciel Utilisé

Cette section décrit les outils, technologies et langages ayant servi de support technique pour la conception et le développement de notre plateforme.

III.2.A Outils et Logiciels de Conception et de Développement

Cette sous-section présente les outils utilisés pour la conception et le développement de notre plateforme.

III.2.A.a Outils de Conception des Diagrammes

Draw.io est un outil gratuit et open source de création de diagrammes UML, organigrammes et schémas de bases de données [13].

III.2.A.b Outils de Design d'Interfaces Graphiques

Figma est un outil de conception d'interfaces basé sur le cloud, permettant la collaboration en temps réel pour la création de maquettes et prototypes [14].

III.2.A.c Éditeur de Code

Visual Studio Code est un éditeur gratuit et open source de Microsoft, intégrant la complétion de code, le débogage et la gestion native de Git [15]

III.2.B Langages de Programmation et Frameworks Utilisés

Cette sous-section présente les langages et frameworks retenus pour le développement du backend et du frontend de notre plateforme académique.

III.2.B.a Backend

Dans cette partie, nous présentons les technologies utilisées côté serveur pour assurer le bon fonctionnement de notre plateforme.

1. Node.js



Node.js est un environnement d'exécution JavaScript gratuit, open-source et multiplateforme, permettant de créer des serveurs, des applications web, des outils en ligne de commande et des scripts [16].

2. API REST



Une API REST est un style architectural standard pour concevoir des applications en réseau, fournissant des règles permettant de créer des services web simples, évolutifs et faciles à intégrer [17].

3. PostgreSQL



PostgreSQL est un SGBD objet-relationnel puissant et open source, avec plus de 35 ans de développement actif, reconnu pour sa fiabilité, sa robustesse et ses performances [18].

III.2.B.b Frontend

Cette sous-section présente les technologies utilisées pour concevoir l'interface utilisateur de notre plateforme académique.

1. HTML5



HTML est le langage de balisage standard pour la création de pages web, permettant de structurer en-têtes, texte, images, tableaux, listes et lecteurs vidéo [19].

2. CSS



CSS décrit l'affichage des éléments HTML à l'écran ou sur papier. Il permet de contrôler la mise en page de plusieurs pages web simultanément via des fichiers CSS externes [20].

3. JavaScript



JavaScript (JS) est un langage léger, interprété et multiparadigme (impératif, fonctionnel, orienté objet), utilisé aussi bien pour les pages web que dans des environnements hors navigateur comme Node.js [19].

4. Kotlin



Kotlin est un langage de programmation moderne, statiquement typé et entièrement interopérable avec Java, officiellement recommandé par Google pour le développement Android. Reconnu pour sa concision, sa sécurité face aux erreurs de pointeurs nuls et sa syntaxe expressive.[21]

III.3 Présentation de l'environnement matériel utilisé

Dans le cadre de la réalisation de ce projet, nous avons utilisé un ordinateur portable dont les principales caractéristiques techniques sont présentées ci-dessous.

- **Modèle :** HP Elitebook 850 G7
- **Processeur :** Intel Core i5-10310U cadencé à 1.70 GHz
- **Mémoire RAM :** 16 Go
- **Stockage :** 500 Go (dont 163 Go utilisés)
- **Carte graphique :** Intel UHD Graphics
- **Système d'exploitation :** Windows 11 64 bits.

III.4 Conclusion

Ce chapitre a présenté l'environnement technique complet de notre plateforme académique. Les outils (Figma, Draw.io, VS Code), les langages et technologies (HTML5, CSS3, JavaScript, Node.js, API REST, PostgreSQL) forment un écosystème moderne et cohérent, garantissant une plateforme fiable, évolutive et maintenable, prête à répondre aux besoins des enseignants et des apprenants.

Conclusion générale

Au terme de ce projet, nous avons pu démontrer la pertinence d'une approche centralisée pour la gestion des activités académiques. L'analyse initiale a permis d'identifier que la dispersion numérique constitue un frein majeur, tant pour la valorisation du profil de l'enseignant que pour l'efficacité du parcours d'apprentissage. La phase de conception a abouti à une architecture robuste, capable de répondre aux besoins spécifiques de nos utilisateurs cibles. La réalisation technique, s'appuyant sur des technologies modernes, a permis de concrétiser une plateforme fonctionnelle, ergonomique et évolutive. Ce travail constitue une base solide pour la transformation numérique de l'environnement pédagogique.

En perspectives, cette plateforme pourra s'enrichir par l'intégration de fonctionnalités de communication en temps réel, le développement d'une application mobile dédiée ou l'ajout d'outils d'analyse statistique de la progression des apprenants. En définitive, cette solution représente un levier efficace pour moderniser l'organisation académique tout en valorisant l'expertise et les ressources des enseignants.

Bibliographie

[1] Kenneth Zeichner and Susan Wray. The teaching portfolio in us teacher education programs : What we know and what we need to know. *Teaching and Teacher Education*, 17(5) :613–621, 2001.

[2] M. Mamchaoui. Portfolio enseignant sur google sites, 2026.

[3] A.Aisli. Portfolio enseignant sur google sites.

[4] E.Russ. Google site portfolio.

[5] C.Loir. Site web portfolio.

[6] Moodle. Moodle learning management system.

[7] Google. Google classroom. `https://classroom.google.com`.

[8] Jean-Claude Bignon. *Architecture et modélisation : Les maquettes numériques*. Presses universitaires de Nancy, Nancy, 2014.

[9] Pascal Roques. *UML 2 par la pratique : Cours et exercices corrigés*. Eyrolles, Paris, 7e edition, 2011.

[10] D.Gaceb. Cours de génie logiciel et modélisation. Support de cours, Université de Boumerdès (UMBB), 2026.

[11] M.MAICH. Cours de merise mehthodology. Support de cours, Université de Amar Tledji laghouat, 2025.

[12] Jakob Freund and Bernd Rücker. *Real-Life BPMN : Using BPMN, CMMN and DMN to Analyze, Improve, and Automate Business Processes*. CreateSpace Independent Publishing Platform, 4e edition, 2019.

[13] JGraph Ltd. Draw.io — diagram software and flowchart maker. `https://www.drawio.com`, 2026. Consulté le 20 mai 2026.

[14] Figma, Inc. Figma : The collaborative interface design tool. `https://www.figma.com`, 2026. Consulté le 20 mai 2026.

[15] Microsoft Corporation. Visual studio code — code editing. redefined. `https://code.visualstudio.com`, 2026. Consulté le 20 mai 2026.

[16] OpenJS Foundation. Node.js — run javascript everywhere. `https://nodejs.org`, 2026. Consulté le 20 mai 2026.

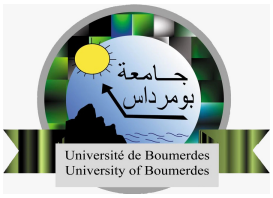
[17] Roy Thomas Fielding. Architectural styles and the design of network-based software architectures. `https://ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm`, 2026. Consulté le 20 mai 2026.

[18] The PostgreSQL Global Development Group. Postgresql : The world's most advanced open source relational database. `https://www.postgresql.org`, 2026. Consulté le 20 mai 2026.

[19] Mozilla Foundation. Mdn web docs. `https://developer.mozilla.org`, 2026. Consulté le 20 mai 2026.

[20] W3Schools. W3schools online web tutorials. `https://www.w3schools.com`, 2026. Consulté le 20 mai 2026.

[21] Google Android Developers. Learn kotlin for android. `https://developer.android.com/kotlin/learn`, 2026. Consulté le 27 mai 2026.





Département d'Informatique

Faculté des Sciences

Université M'hamed BOUGARA
BOUMERDÈS

Année académique : 2025-2026