



UNIVERSITE DE FIANARANTSOA
ECOLE NATIONALE D'INFORMATIQUE

**MEMOIRE DE FIN D'ETUDES POUR L'OBTENTION DU DIPLOME
DE LICENCE PROFESSIONNELLE**

Mention : Informatique

Parcours : Informatique Générale

Intitulé :

**CONCEPTION ET REALISATION
D'UNE PLATEFORME WEB POUR LA
COMMUNICATION ET L'EDUCATION**

Présenté le 28 Novembre 2023

Par : Monsieur RASOLOFONIAINA TSIHEJE Marie Mickaelio

Membres du Jury :

Président : Monsieur MAHATODY Thomas, Docteur HDR

Examineur : Monsieur RANDRIANOMENJANAHARY Ferdinand, Doctorant en informatique

Rapporteurs : Monsieur BERTIN Andry, Doctorant en Informatique (Encadreur pédagogique)

Madame ARISOA Finaritra Ambroise, Ingénieur en Informatique (Encadreur professionnel)

Année Universitaire 2022-2023

CURRICULUM VITAE

RASOLOFONIAINA TSIHEJE Marie Mickaelio

Né le 13 Septembre 2002 à Bemasoandro Itaosy Antananarivo

Lot IG 067/3204 Tanambao Igaga Fianarantsoa

Adresse mail : tsihejem@gmail.com

Téléphone : +261342341566



FORMATION ET DIPLOME OBTENU

- 2022 - 2023 : Troisième année de formation en licence professionnel dans la parcours Informatique générale à l'Ecole Nationale d'Informatique à Fianarantsoa
- 2021 - 2022 : Deuxième année de formation en licence professionnel dans la parcours Informatique générale à l'Ecole Nationale d'Informatique à Fianarantsoa
- 2020 - 2021 : Première année de formation en licence professionnel dans la parcours Informatique générale à l'Ecole Nationale d'Informatique à Fianarantsoa
- 2018 - 2019 : Obtention du Baccalauréat série D à Lycée Rahevivelo Ramamonjy(LRR) à Fianarantsoa

STAGES ET EXPERIENCES PROFESSIONNELLES

- Conception et réalisation d'une plateforme web éducative en utilisant NodeJs, ExpressJs, ReactJs, Mysql cote SGBD, Stagiaire chez Faculté de Science Fianarantsoa.
- Création de mon portfolio personnel pour présenter mes projets et compétences en développement web, démonstration ma créativité et ma capacité à concevoir des interfaces attrayantes.
- Conception et réalisation d'une application de gestion de logement avec les Framework ReactJs, ExpressJs et NodeJs, permettant une gestion efficace des biens immobiliers.
- Développement d'une application web de gestion de bibliothèque avec le Framework ReactJs et PHP pour faciliter le suivi des livres et des emprunteurs.

COMPETENCES EN INFORMATIQUE

Outils bureautiques	Microsoft Word, Microsoft PowerPoint
Moyens de conception	UML, 2TUP
Langages de programmation	PHP, Python, Java
Technologies web	HTML, CSS, JavaScript
SGBD	MySQL, PostgreSQL, MongoDB
Maintenances	Matériel, Logiciel
Framework ou Bibliothèques ou CMS	Express Js, React Js, Laravel

COMPETENCES LINGUISTIQUES

Très bien : TB Bien : B Assez bien : AB

	Comprendre	Parler	Ecrire	Lire
Français	B	B	B	B
Anglais	AB	AB	AB	AB

DIVERTISSEMENTS

- Football,
- Jeux vidéos
- Lecture

SOMMAIRE GENERAL

CURRICULUM VITAE.....	I
SOMMAIRE GENERAL.....	III
REMERCIEMENTS.....	V
LISTE DES FIGURES.....	VI
LISTE DES TABLEAUX.....	VIII
LISTE DES ABREVIATIONS.....	IX
INTRODUCTION GENERALE.....	1
PARTIE I : PRESENTATION.....	2
Chapitre 1. Présentation de l'Ecole Nationale d'Informatique.....	3
1.1. Information d'ordre général.....	3
1.2. Missions et historiques.....	3
1.3. Organigramme institutionnel.....	5
1.4. Domaine de spécialisation.....	6
1.5. Architecture des formations pédagogiques.....	6
1.6. Relations de l'ENI avec l'organisme externes.....	8
1.7. Débouches professionnel et diplôme	9
1.8. Ressources humaines.....	10
Chapitre 2. Présentation de la faculté de Science Fianarantsoa.....	11
2.1. Information général.....	11
2.2. Historiques.....	11
2.3. Offre des formations.....	12
2.4. Les personnels administratifs et techniques.....	14
Chapitre 3. Description du projet.....	15
3.1. Formulation.....	15
3.2. Objectif et besoins de l'utilisateur.....	15
3.3. Moyens nécessaires à la réalisation du projet.....	16
3.4. Résultats attendus.....	17
3.5. Chronogramme de travail.....	18
PARTIE II : ANALYSE ET CONCEPTION.....	19
Chapitre 4. Analyse préalable.....	20
4.1. Analyse de l'existant.....	20
4.2. Critique de l'existant.....	20
4.3. Conception avant-projet.....	21
Chapitre 5. Analyse conceptuelle.....	29
5.1. Présentation de la méthode utilisée.....	29
5.2. Dictionnaire des données.....	29
5.3. Règles des gestions	30
5.4. Représentation et spécifications des besoins.....	32
5.5. Spécification des besoins techniques.....	50
5.6. Modélisation du domaine.....	51
Chapitre 6. Conception détaillée.....	52
6.1. Architecture de système	52
6.2. Diagramme de séquence de conception pour chaque cas d'utilisation.....	53

6.3. Diagramme de classe de conception pour chaque cas d'utilisation.....	60
6.4. Diagramme de classe de conception globale.....	64
6.5. Diagramme de paquetage.....	65
6.6. Diagramme de déploiement.....	65
PARTIE III : REALISATION.....	66
Chapitre 7. Mise en place de l'environnement de développement.....	67
7.1. Installation et configuration des outils.....	67
7.2. Architecture de l'application.....	72
Chapitre 8. Développement de l'application.....	73
8.1. Création de base de données.....	73
8.2. Codage de l'application.....	73
8.3. Présentation de l'application.....	77
CONCLUSION.....	85
BIBLIOGRAPHIE.....	X
WEBOGRAPHIE.....	XI
GLOSSAIRE.....	XII
TABLE DE MATIERE.....	XIII
RESUME.....	XV
ABSTRACT.....	XV

REMERCIEMENTS

Premièrement, nous tenons à rendre grâce à Dieu pour m’avoir donné la force, le courage et la santé durant la rédaction et la réalisation de ce mémoire.

Je tiens également à remercier :

- Monsieur **HAJALALAINA Aimé Richard**, Docteur HDR, Président de l’Université de Fianarantsoa, d’avoir assuré le bon fonctionnement de nos études à l’Université ;
- Monsieur **MAHATODY Thomas**, Docteur HDR, Directeur de l’Ecole Nationale d’Informatique de Fianarantsoa pour nous avoir donné l’opportunité de terminer notre étude en Troisième année de la licence professionnelle ;
- Monsieur **ANDRIAZAFIMHAZO Lahinirina Fridolin**, le Doyen de la FACULTE DE SCIENCE pour nous avoir accueillis au sein de son organisme pour effectuer mon stage ;
- Monsieur **RABETAFIKA Louis Haja**, Maître de conférence, responsable de la mention informatique à l’ENI ;
- Monsieur **GILANTE Gesazafy**, Assistant d’Enseignement Supérieur et de recherche, responsable du parcours Informatique générale à l’ENI
- Monsieur **BERTIN Andry**, Doctorant en informatique, mon encadreur pédagogique, pour sa constante disponibilité et son aide inconditionnelle dans la rédaction de ce mémoire ;
- Madame **ARISOA Finaritra Ambroise**, Ingénieur en Informatique, mon encadreur professionnel, pour son étroite collaboration dans l’accomplissement de ce travail ;
- Je tiens à exprimer ma sincère gratitude envers tous les membres de jury pour l’attention qu’ils m’accordent durant la présentation de ce mémoire ;
- Nous saisissons de cette occasion pour exprimer notre reconnaissance et notre gratitude envers tous nos professeurs et enseignants de l’ENI pour nous avoir transmis leurs connaissances durant toute l’année ;
- Enfin, nous tenons à adresser nos remerciements à nos familles et à nos amis, qui nous ont toujours soutenus et poussés à continuer nos études. Ce présent travail a pu voir le jour grâce à leur soutien.

LISTE DES FIGURES

Figure 1. Organigramme actuelle de l'école.....	5
Figure 2. Architecture des études correspondant au système LMD.....	7
Figure 3. Organigramme de la faculté de Science.....	14
Figure 4. Chronogramme des travaux.....	18
Figure 5. Phase de développement du processus unifié.....	29
Figure 6. Processus de développement en Y.....	30
Figure 7. Formalisme d'un cas d'utilisation.....	33
Figure 8. Diagramme de cas d'utilisation.....	35
Figure 9. Formalisme d'un diagramme de séquence.....	42
Figure 10. Diagramme de séquence de système : S'inscrire.....	43
Figure 11. Diagramme de séquence de système : S'authentifier.....	43
Figure 12. Diagramme de séquence de système : Créer compte Professeurs.....	44
Figure 13. Diagramme de séquence de système : Publier des annonces et informations.....	44
Figure 14. Diagramme de séquence de système : Création des cours.....	45
Figure 15. Diagramme de séquence de système : Création des devoirs.....	45
Figure 16. Diagramme de séquence de système : Consultation des annonces et informations....	46
Figure 17. Diagramme de séquence de système : Consultation des cours.....	46
Figure 18. Diagramme de séquence de système : Consultation des devoirs.....	47
Figure 19. Diagramme de séquence de système : Soumettre des devoirs.....	47
Figure 20. Diagramme de séquence de système : Gérer son profil.....	48
Figure 21. Diagramme de séquence de système : Notifier les Utilisateurs.....	48
Figure 22. Modèle de domaine	50
Figure 23. Architecture du système.....	51
Figure 24. Diagramme de séquence de conception : S'inscrire.....	52
Figure 25. Diagramme de séquence de conception : S'authentifier.....	53
Figure 26. Diagramme de séquence de conception : Créer compte Professeurs.....	53
Figure 27. Diagramme de séquence de conception : Publier des annonces et informations.....	54
Figure 28. Diagramme de séquence de conception : Création des cours.....	54
Figure 29. Diagramme de séquence de conception : Création des devoirs.....	55
Figure 30. Diagramme de séquence de conception : Consultation des annonces et informations..	55
Figure 31. Diagramme de séquence de conception : Consultation des cours.....	56
Figure 32. Diagramme de séquence de conception : Consultation des devoirs.....	56
Figure 33. Diagramme de séquence de conception : Soumettre des devoirs.....	57
Figure 34. Diagramme de séquence de conception : Gérer son profil.....	57
Figure 35. Diagramme de séquence de conception : Notifier les Utilisateurs.....	58
Figure 36. Diagramme de classe de conception : S'inscrire.....	58
Figure 37. Diagramme de classe de conception : S'authentifier.....	59
Figure 38. Diagramme de classe de conception : Créer compte Professeurs.....	59
Figure 39. Diagramme de classe de conception : Publier des annonces et informations.....	59
Figure 40. Diagramme de classe de conception : Création des cours.....	60
Figure 41. Diagramme de classe de conception : Création des devoirs.....	60
Figure 42. Diagramme de classe de conception : Consultation des annonces et informations....	61
Figure 43. Diagramme de classe de conception : Consultation des cours.....	61

Figure 44. Diagramme de classe de conception : Soumettre devoirs.....	61
Figure 45. Diagramme de classe de conception global.....	63
Figure 46. Diagramme de paquetage.....	64
Figure 47. Diagramme de déploiement.....	64
Figure 48. Installation de Visual Paradigm.....	66
Figure 49. Installation de Laragon.....	67
Figure 50. Lancement de Laragon.....	68
Figure 51. Installation de Visual Studio Code.....	69
Figure 52. Lancement de Visual Studio Code.....	69
Figure 53. Installation de NodeJs.....	70
Figure 54. Architecture de l'application.....	71
Figure 55. Connexion à la base de donnée.....	72
Figure 56. Routage en ExpressJs.....	73
Figure 57. Authentification en ExpressJs.....	74
Figure 58. Récupération des données vis à l'API.....	74
Figure 59. Page d'Authentification du cote Professeur et administrateur.....	75
Figure 60. Dashboard.....	75
Figure 61. Interface de création de comptes professeur.....	76
Figure 62. Liste des professeurs.....	76
Figure 63. Interface de publication des annonces.....	77
Figure 64. Liste des annonces publier.....	77
Figure 65. Interface de création des cours.....	78
Figure 66. Liste des cours.....	78
Figure 67. Interface de création des devoirs.....	79
Figure 68. Liste des devoirs.....	79
Figure 69. Page d'accueil.....	80
Figure 70. Interface d'Authentification des étudiants.....	80
Figure 71. Interface d'inscription des étudiants.....	81
Figure 72. Liste des annonces.....	81
Figure 73. Liste des cours.....	82
Figure 74. Liste des devoirs.....	82
Figure 75. Liste des professeurs.....	83
Figure 76. Interface de paramètre de compte.....	83
Figure 77. Interface de notification.....	84
Figure 78. Interface pour voir les annonces, cours et devoir.....	84

LISTE DES TABLEAUX

Tableau 1. Organisation du système de formation pédagogique de l'école.....	6
Tableau 2. Mention et parcours au sien de l'ENI.....	6
Tableau 3. Liste des formations existant à l'ENI.....	7
Tableau 4. Débouchés éventuels des jeunes diplômés.....	9
Tableau 5. Evolution de l'ouverture des formations à la faculté de Science.....	11
Tableau 6. Les différents mentions et parcours a la faculté de Science.....	13
Tableau 7. Les caractéristiques des matériels utiliser pendant le projet.....	17
Tableau 8. Moyens logiciels utilisés.....	17
Tableau 9. Critique de l'organisation actuelle.....	20
Tableau 10. Comparaison des solutions proposer.....	21
Tableau 11. Comparaison de 2TUP et Merise.....	22
Tableau 12. Comparaison des Framework.....	24
Tableau 13. Comparaison de NodeJs et PHP.....	24
Tableau 14. Comparaison des SGBD.....	25
Tableau 15. Comparaison entre Visual Studio Code et Webstrom.....	26
Tableau 16. Comparaison entre UML Designer et Visual Paradigm for UML.....	27
Tableau 17. Dictionnaires des données.....	30
Tableau 18. Priorisation des cas d'utilisations.....	42

LISTE DES ABREVIATIONS

2TUP: two Tracks Unified Process
API: Application Programming Interface
BM: Biologie Moléculaire
BT: Biologie Tropical
DTS : Diplôme de Technicien Supérieur
DUT : Diplôme Universitaire de Technicien
ENI: Ecole National d'Informatique
HDD: Hard Disk Drive.
HTML: HyperText Markup Language
HTTP: Portable Document Format.
JSON: JavaScript Object Notation.
LMD: Licence-Master-Doctorat
ME: Mathématique Elémentaire
MF: Mathématique Fondamental
MISS: Mathématique Informatique
MVC : Model-View-Contrôleur
PDF: Portable Document Format.
PHP: Hypertext Preprocessor.
RAM: Random Access Memory.
REST: Representational State.
RG: Règles de Gestion
SGBD: Système de Gestion de Base de Donnée
SQL: Structured Query Language.
UML: Unified Modeling Language.
XML : Extensible Markup Language.

INTRODUCTION GENERALE

L'éducation est un pilier fondamental de notre société, et son évolution a été marquée ces dernières décennies par une transformation numérique sans précédent. Les avancées technologiques ont ouvert la voie à de nouvelles opportunités d'apprentissage et de collaboration, transformant la manière dont les enseignants dispensent leurs cours et dont les étudiants accèdent aux ressources éducatives. Dans ce contexte de changement continu, la création d'une plateforme éducative novatrice devient impérative pour répondre aux besoins en constante évolution des enseignants et des étudiants.

Le présent mémoire a pour objectif de détailler la conception et le développement d'une plateforme éducative destinée à faciliter la communication et la collaboration entre les enseignants et les étudiants, tout en offrant un accès efficace aux cours et aux ressources pédagogiques. Cette plateforme vise à combler les lacunes observées dans les systèmes éducatifs traditionnels en offrant une expérience d'apprentissage plus interactive, flexible et accessible.

Ce mémoire explore en détail les différentes phases du processus de développement de la plateforme, en mettant l'accent sur les choix technologiques, les considérations de conception, les défis techniques, et les solutions adoptées pour assurer la sécurité des données et la convivialité de l'interface.

Enfin ce mémoire comporte trois parties, dans la première partie on va voir la présentation générale dans laquelle figure la présentation de l'ENI Fianarantsoa et de la FACULTE DE SCIENCE que j'ai faisais mon stage dans cette partie aussi qu'on va parler la description du projet en suite dans la deuxième partie on va voir l'analyse et conception dans laquelle figure l'analyse préalable, l'analyse de conceptuelle et la conception détaillée et dans la troisième partie on va voir la réalisation de projet dans laquelle figure la mise en place de l'environnement de développement et la développement de l'application.

PARTIE I : PRESENTATIONS

Chapitre 1. Présentation de l'Ecole Nationale d'Informatique

Dans ce chapitre, nous allons explorer en détail la présentation de l'Ecole Nationale d'Informatique (ENI).

1.1. Information d'ordre générale

L'Ecole Nationale d'Informatique, en abrégé ENI, est un établissement d'enseignement supérieur rattaché académiquement et administrativement à l'Université de Fianarantsoa. Le siège de l'Ecole se trouve à Tanambao-Antaninarenina à Fianarantsoa. L'adresse pour la prise de contact avec l'Ecole est la suivante : Ecole Nationale d'Informatique (ENI) Tanambao, Fianarantsoa. Le numéro de sa boîte postale est 1487 avec le code postal 301. Téléphone : 034 05 733 36 ou 032 15 204 28. Son adresse électronique est la suivante : **eni@eni.mg**. Il dispose également d'un site web : **www.eni.mg**

1.2. Missions et historiques

L'ENI se positionne sur l'échiquier socio-éducatif malgache comme étant le plus puissant secteur de diffusion et de vulgarisation des connaissances et des technologies informatiques.

Cette Ecole Supérieure peut être considérée aujourd'hui comme la vitrine et la pépinière des élites informaticiennes du pays.

De façon formelle, l'ENI était créée par le décret N° 83- 185 du 24 Mai 1983, comme étant le seul établissement Universitaire Professionnalisé au niveau national, destiné à former des techniciens et des Ingénieurs de haut niveau, aptes à répondre aux besoins et exigences d'Informatisation des entreprises, des sociétés et des organes implantés à Madagascar.

L'ENI a pour conséquent pour mission de former des spécialistes informaticiens compétents et opérationnels de différents niveaux notamment :

- En fournissant à des étudiants des connaissances de base en informatique ;
- En leur transmettant le savoir-faire requis, à travers la professionnalisation des formations dispensées et en essayant une meilleure adéquation des formations par rapport aux besoins évolutifs des sociétés et des entreprises ;
- En initiant les étudiants aux activités de recherche dans les différents domaines des Technologies de l'Information et de la Communication (TIC) ;

La filière de formation d'Analystes Programmeurs a été mise en place à l'Ecole en 1983, et a été gelée par la suite en 1996, tandis que la filière de formation d'ingénieurs a été ouverte à l'Ecole en 1986.

Une formation de troisième cycle a été ouverte à l'Ecole a été ouverte à l'Ecole depuis l'année 2003 – 2004 grâce à la coopération académique et scientifique entre l'Université de Fianarantsoa pour le compte de l'ENI et l'Université Paul Sabatier de Toulouse (UPST).

Cette filière avait pour objectif de former certains étudiants à la recherche dans les différents domaines de l'Informatique, et notamment pour préparer la relève des Enseignants-Chercheurs qui étaient en poste.

Pendant l'année 2007-2008, la formation en vue de l'obtention du diplôme de Licence Professionnelle en Informatique a été mise en place à l'ENI avec les deux parcours de formation :

- Génie Logiciel et base de Données.
- Administration des Système et réseaux.

La mise en place à l'Ecole de ces deux options de formation devait répondre au besoin de basculement vers le système Licence – Master – Doctorat (LMD).

En vue de surmonter les difficultés de limitation de l'effectif des étudiants accueillis à l'Ecole, notamment à cause du manque d'infrastructures, un système de « Formation Hybride » a été mise en place à partir de l'année 2010. Il s'agit en effet d'un système de formation semi présentielle et à distance avec l'utilisation de la visioconférence pour la formation à distance. Le système de formation hybride a été ainsi créé à Fianarantsoa ainsi qu'Université de Toliara. Cette formation est à l'origine du parcours Informatique Générale.

En 2023, une nouvelle mention Intelligence Artificielle (IA) a été ouvert au sein de l'Ecole pour répondre les besoins des entreprises. La formation est destinée aux étudiants titulaires du diplôme de licence (Bac +3) en Mathématiques ou en Statistiques ou en Informatique, etc. La mention IA comporte deux parcours:

- Gouvernance et Ingénierie de Données (GID),
- Objets connectés et Cybersécurité (OCC).

Le principe de l'enseignement pour le parcours GID offre aux l'étudiants des compétences scientifiques et techniques spécialisées en Science de données. Pour le parcours OCC, les étudiants octroient la double spécialité premièrement en internet des objets et deuxièmement en cyber

sécurité. La formation de master est axée sur l'ensemble d'applications de l'Intelligence Artificielle.

1.3. Organigramme institutionnel

L'organigramme de l'Ecole est inspiré des dispositions du décret N° 83-185 du 24 Mai 1983. L'ENI est administrée par un Conseil d'Ecole, et dirigée par un directeur nommé par un décret adopté en Conseil des Ministres. Le Collège des enseignants regroupant tous les enseignants-chercheurs permanents de l'Ecole est chargé de résoudre les problèmes liés à l'organisation pédagogique des enseignements. Le Conseil Scientifique propose les orientations pédagogiques et scientifiques de l'établissement, en tenant compte notamment de l'évolution du marché de travail et de l'adéquation des formations dispensées par rapport aux besoins des entreprises. La figure 1 représente l'organigramme actuel de l'ENI.

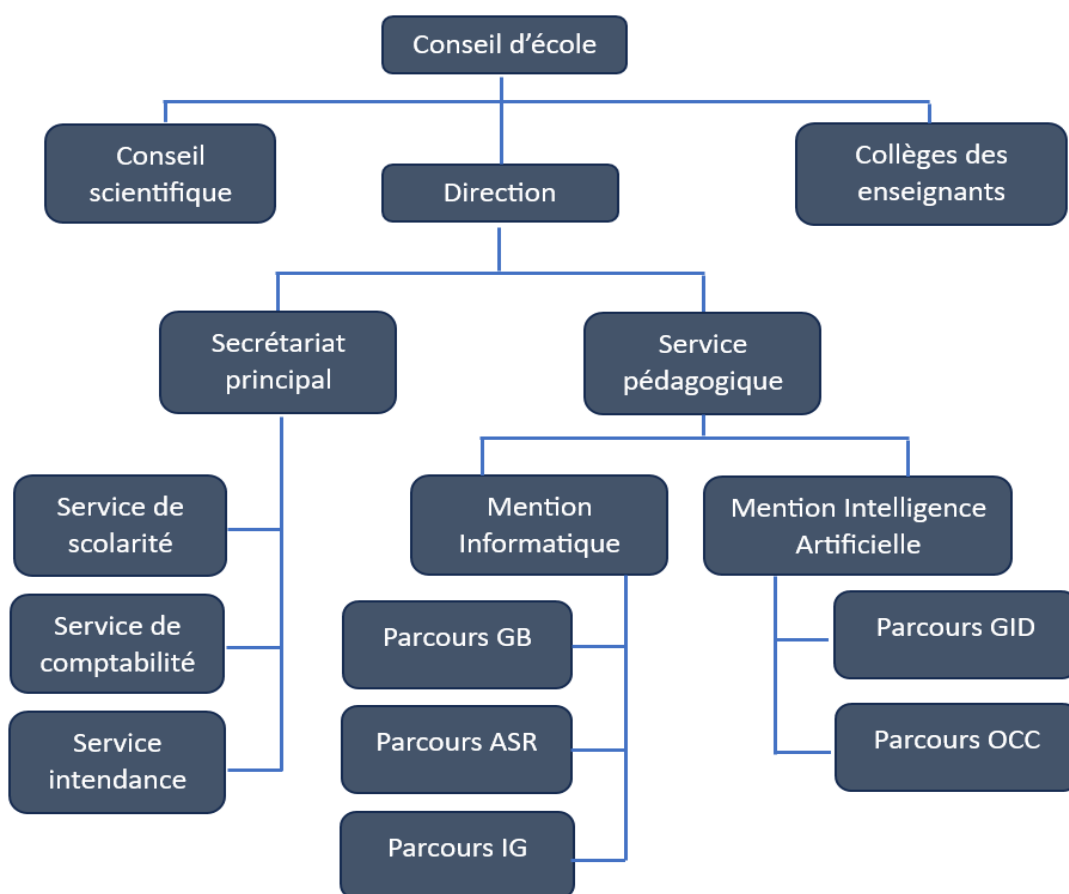


Figure 1. Organigramme actuel de l'Ecole

1.4. Domaine de spécialisation

Les activités de formation et de recherche organisées à l'ENI portent sur les domaines suivants :

- Génie logiciel et Base de Données ;
- Administration des Systèmes et Réseaux ;
- Informatique Générale ;
- Modélisation informatique et mathématique des Systèmes complexes ;
- Intelligence artificielle.

Le tableau 1 décrit l'organisation du système de formation pédagogique de l'Ecole.

Tableau 1. Organisation du système de formation pédagogique de l'Ecole

Formation Théorique	Formation Pratique
<ul style="list-style-type: none">- Enseignement théorique- Travaux dirigés- Travaux pratiques- Conférences	<ul style="list-style-type: none">- Etude de cas- Travaux de réalisation- Projets/ Projets tutorés- Voyages d'Etudes- Stages en entreprise

1.5. Architecture des formations pédagogiques

Le recrutement des étudiants à l'ENI se fait uniquement par voie de concours d'envergure nationale en première année. Les offres de formation organisées à l'Ecole ont été validées par la Commission Nationale d'Habilitation (CNH). Au sein de l'ENI, il existe deux mentions et cinq parcours. Le tableau 2 récapitule les mentions et les parcours au sein de l'Ecole :

Tableau 2. Mention et parcours au sein de l'ENI

Mention	Parcours
Informatique	Génie logiciel et Base de Données (GB)
	Administration des Systèmes et Réseaux (ASR)
	Informatique Générale (IG)
Intelligence Artificielle	Gouvernance et Ingénierie de Données (GID)
	Objets Connectés et Cyber sécurités (OCC)

La figure 2 représente l'architecture des études correspondant au système LMD.

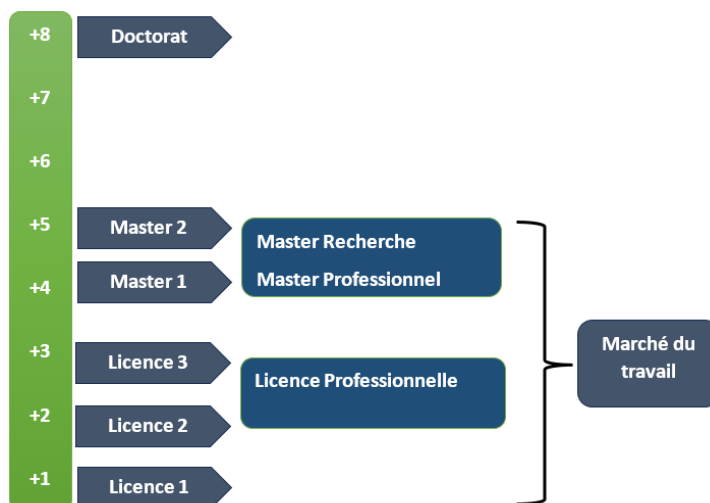


Figure 2. Architecture des études correspondant au système LMD

La licence peut avoir une vocation générale ou professionnelle. Le master peut avoir une vocation professionnelle ou de recherche. L'accès en première année de MASTER se fait automatiquement pour les étudiants de l'Ecole qui ont obtenu le diplôme de Licence Professionnelle. Le tableau 3 illustre la liste des formations existantes à l'ENI.

Tableau 3. Liste des formations existantes à l'ENI

	FORMATION	
	LICENCE PROFESSIONNELLE	MASTER
Condition admission	Par voie de concours	Par voie de concours pour la mention IA
Condition d'accès	Bac de série C, D ou Technique	Être titulaire de licence professionnelle
Durée de Formation	3 ans	2 ans
Diplôme délivré	Diplôme de Licence Professionnelle	Diplôme de Master Professionnel Diplôme de Master Recherche

Le Master Recherche permet à son titulaire de poursuivre directement des études en doctorat et de s'inscrire directement dans une Ecole Doctorale.

Les étudiants diplômés de l'Ecole sont plutôt bien accueillis dans les instituts universitaires étrangères (Canada, Suisse, France, ...)

1.6. Relation de l'ENI avec les organismes externes

Les stages effectués chaque année par les étudiants mettent l'Ecole en rapport permanent avec plus de 400 entreprises et organismes publics, semi-publics et privés, nationaux et internationaux. L'Ecole dispose ainsi d'un réseau d'entreprises, de sociétés et d'organismes publics et privés qui sont des partenaires par l'accueil en stage de ses étudiants, et éventuellement pour le recrutement après l'obtention des diplômes par ces derniers. Les compétences que l'Ecole cherche à développer chez ses étudiants sont l'adaptabilité, le sens de la responsabilité, du travail en équipe, le goût de l'expérimentation et l'innovation.

En effet, la vocation de l'ENI est de former des licenciés et des ingénieurs de niveau MASTER avec des qualités scientifiques, techniques et humaines reconnues, capables d'évoluer professionnellement dans des secteurs d'activité variés intégrant l'informatique. Les stages en milieu professionnel permettent de favoriser une meilleure adéquation entre les formations à l'Ecole et les besoins évolutifs du marché de l'emploi.

Parmi les sociétés, les entreprises et les organismes partenaires de l'Ecole, on peut citer : ACCENTURE Mauritius, AKATA Goavana, Air Madagascar, Ambre Associates, Airtel, Agence Universitaire de la Francophonie (AUF), AXIAN, B2B, Banque Centrale, , BIANCO, BlueLine, CNaPS, Bureau National de Gestion des Risques et des Catastrophes (BNGRC), CEDII-Fianarantsoa, Data Consulting, Central Test, Centre National Antiacridien, CNRE, COLAS, Direction Générale des Douanes, DLC, E-Tech Consulting, , FID, FIHARY Soft, FTM, GNOSYS, GENIUS AT WORK, Hello Tana, IBONIA, INGENOSIA, INSTAT, IOGA, JIRAMA, JOUVE, MADADEV, MAEP, MANAO, MEF, MEN, MESupRES, MFB, , MININTER, Min des Postes/Télécommunications et du Développement Numérique, NEOV MAD, Ny Havana, Madagascar National Parks, OMNITEC, ORANGE, OTME, PRACCESS, QMM Fort-Dauphin, SG Madagasikara SMMC, SMMEC, SNEDADRS Antsirabe, Sénat, Société d'Exploitation du Port de Toamasina (SEPT), SOFTWELL, Strategy Consulting, TELMA, VIVETEC, Société LAZAN'I BETSILEO, WWF, UGD, ARATO, MANAO, MNDPT, NG ACADEMY.NG, Relia ...

1.7. Débouchés professionnels et diplômés

Les formations proposées par l'Ecole permettent aux diplômés d'être immédiatement opérationnels sur le marché du travail avec la connaissance d'un métier complet lié à l'informatique aux TIC.

L'Ecole apporte à ses étudiants un savoir-faire et un savoir-être qui les accompagnent tout au long de leur vie professionnelle. Elle a une vocation professionnalisant. Les diplômés en LICENCE et en MASTER issus de l'ENI peuvent faire carrière dans différents secteurs.

L'Ecole bénéficie aujourd'hui de 40 années d'expériences pédagogiques et de reconnaissance auprès des sociétés, des entreprises et des organismes. C'est une Ecole Supérieure de référence en matière informatique.

D'une manière générale, les diplômés de l'ENI n'éprouvent pas de difficultés particulières à être recrutés au terme de leurs études. Cependant, l'ENI recommande à ses diplômés de promouvoir l'entrepreneuriat en TIC et de créer des cybercafés, des SSII ou des bureaux d'études. Le tableau 4 représente les débouchés éventuels des jeunes diplômés.

Tableau 4. Débouchés éventuels des jeunes diplômés

LICENCE	MASTER
<ul style="list-style-type: none">➤ Analyste - Programmeur➤ Administrateur de site web/de portail web➤ Assistant Informatique et internet➤ Chef de projet web ou multimédia➤ Développeur Informatique ou multimédia➤ Intégrateur web ou web designer➤ Hot liner/Hébergeur Internet➤ Agent de référencement➤ Technicien/Supérieur de help desk sur Informatique➤ Responsable de sécurité web➤ Administrateur de réseau	<ul style="list-style-type: none">➤ Administrateur de réseau et système➤ Architecture de système d'information➤ Développeur d'applications➤ Ingénieur réseau➤ Webmaster / Web Designer➤ Concepteur et réalisateur d'application➤ Directeur du système d'informations➤ Chef de projet informatique➤ Responsable de sécurité informatique➤ Consultant fonctionnel ou freelance

1.8. Ressources humaines

Les ressources humaines sont citées ci-dessous selon leurs responsabilités :

- Directeur de l'Ecole : Monsieur MAHATODY Thomas, Docteur HDR
- Responsable de la Mention « Informatique » : Monsieur RABETAFIKA Louis Haja, Maître de Conférences
- Responsable de la Mention « Intelligence Artificielle » : Monsieur DIMBISOA William Germain, Maître de Conférences
- Responsable du Parcours « Génie Logiciel et Base de Données » : Monsieur RALAIVAO Jean Christian, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Administration Systèmes et Réseaux » : Monsieur SIAKA, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Informatique Générale » : Monsieur GILANTE Gesazafy, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Gouvernance et Ingénierie de Données » : Madame RATIANANTITRA Volatiana Marielle, Maître de Conférences
- Responsable du Parcours « Objets Connectés et Cybersécurité » : Monsieur RAZAFIMAHATRATRA Hajarisena, Maître de Conférences

L'ENI compte quinze (15) enseignants permanents dont un (01) Professeur Titulaire, un (01) Professeur, un (01) Docteur HDR, huit (08) Maîtres de Conférences, quatre (04) Assistants d'Enseignement Supérieur et de Recherche, dix (10) enseignants vacataires, quarante un (41) personnel administratif.

Chapitre 2. Présentation de la faculté de Science

Au sien de ce chapitre, nous entreprendrons une exploration approfondie de la faculté de Science Fianarantsoa.

2.1. Information général

La faculté des Sciences est un établissement public de l'université de Fianarantsoa. Des formations de recherches et professionnelles y sont établies. Celles-ci distinguées dans plusieurs mentions et parcours. Le siège de la faculté de Science Fianarantsoa se trouve à Campus Universitaire Andrainjato Fianarantsoa. L'adresse pour la prise de contact avec faculté de Science est la suivante : Campus Universitaire Andrainjato Fianarantsoa. Le numéro de sa boîte postale est 1264 – 301 Fianarantsoa. Téléphone : 034 27 931 75 ou 034 46 620 41. Son adresse électronique est la suivante : Email : **facsciences@gmail.com**. Il dispose également d'un site web : **www.facsciences-fianara.mg**

2.2. Historiques

Le Centre Universitaire Régional (CUR) de Fianarantsoa commença effectivement à Exister durant l'année universitaire 1977/1978, et le 06 Juin 1977, avec la Première année d'une seule filière « Mathématiques » avec 126 étudiants. Comme il n'y avait que cette seule filière, le CUR fut confondu avec l'établissement abritant la filière « Mathématique » jusqu'à la création d'autres filières telles que la filière Droit en 1980. Depuis lors, l'Etablissement d'Enseignement Supérieur des Sciences fut créé et d'autres filières vinrent s'ajouter à la filière Mathématique selon le tableau ci-après. Puis depuis le changement du CUR en Université, l'Etablissement d'Enseignement Supérieur des Sciences fut remplacé par la Faculté des Sciences et les filières sont alors remplacées par des Départements.

Tableau 5. Evolution de l'Ouverture de Formation de la faculté de Science

Année	Ouverture
1977 – 1978	Centre Universitaire Régional – Filière : Mathématiques
1985 – 1986	Filière : Physique Chimie
1990 – 1991	Département de Chimie
1997 – 1998	Filière : Mathématique Informatique pour les Sciences Sociales (MISS)
1998 – 1999	Formation en Troisième Cycle de Chimie

2005 – 2006	Laboratoire de Physique Appliquée de l'Université de Fianarantsoa (LAPAU)
2008 – 2009	Département de Médecine
2015 – 2016	Mention Science de la Vie

Le 19 Août 2022, l'Université de Fianarantsoa, la Faculté des Sciences et le Gouverneur de la Région Atsimo Atsinanana ont signé une convention pour la mise en place de la Mention Sciences de la Vie au Centre Universitaire Régional Atsimo Antsinanana. Ceci fut dans l'optique de définir l'étroite collaboration ainsi que l'engagement des trois parties pour le bon fonctionnement de ladite Mention.

La faculté des sciences a fêté son 45^{ème} anniversaire (1977-2022) le 05 au 09 Décembre 2022 où il y aura beaucoup d'activités telle que le Basket Ball, Foot Ball, Semi-marathon, activités culturelles et série de conférence.

2.3. Offres des formations

Nous distinguons les offres de formation pendant l'ancien système et depuis le basculement vers le système LMD. Dans les deux cas, le recrutement des étudiants en première année se fait par sélection des dossiers et est ouvert aux titulaires d'un BAC SCIENTIFIQUE (Série C, D ou S) ou d'un BAC TECHNIQUE ET PROFESSIONNEL (Génie civil ou Génie Industriel) ou alors d'un BAC TECHNIQUE ET TECHNOLOGIQUE.

2.3.1. Ancien système

Selon le Tableau 5, il existait cinq Départements à la Faculté des Sciences :

- Département de Mathématiques, comprenant les filières Mathématique Fondamentale et Mathématique - Economie.
- Département de Physique
- Département de Chimie
- Département de Mathématique et Informatique pour les Sciences (M.I.S.S)
- Département de Médecine

La faculté à adopter cet ancien système jusqu'à l'année universitaire 2008 – 2009. Ensuite, depuis 2010, année de création de la Faculté de médecine, la Faculté des sciences possède quatre départements jusqu'à la transition vers le système LMD.

2.3.2. Système LMD

Depuis l'année universitaire 2012 - 2013, la Faculté des sciences bascule progressivement vers le système LMD (Licence, Master et Doctorat). Les différentes structures dans le système LMD étant le Domaine, la Mention et le Parcours, la Faculté a pour domaine les Sciences et Technologie comprend quatre mentions dans le grade licence et cinq mentions dans le grade master qui se résume dans le tableau 6:

Tableau 6. Les différents mentions et parcours de la Faculté des sciences

LICENCE				
Mathématique t				
Application	Physique Chimie	Physique et Application	Science de la Vie	
MF	Physique	EA2I	BM	
ME	Chimie généraliste		BT	
MISS	LP3E			
MASTER				
Mathématique et Application	Physique et Application	Chimie	Science de la Vie	GSEEM
MF	CAP	Chimie médical	BM	IEET
ME	MET	Géochimie	BT	GIRE
MISS	EMS	Vini-viticole		ER

2.4. Les Personnels Administratifs et Techniques

En tout et pour tout, huit (08) Agents furent chargés des tâches administratives lors de l'ouverture du Centre Universitaire Régional, et quinze (15) autres pour les tâches techniques.

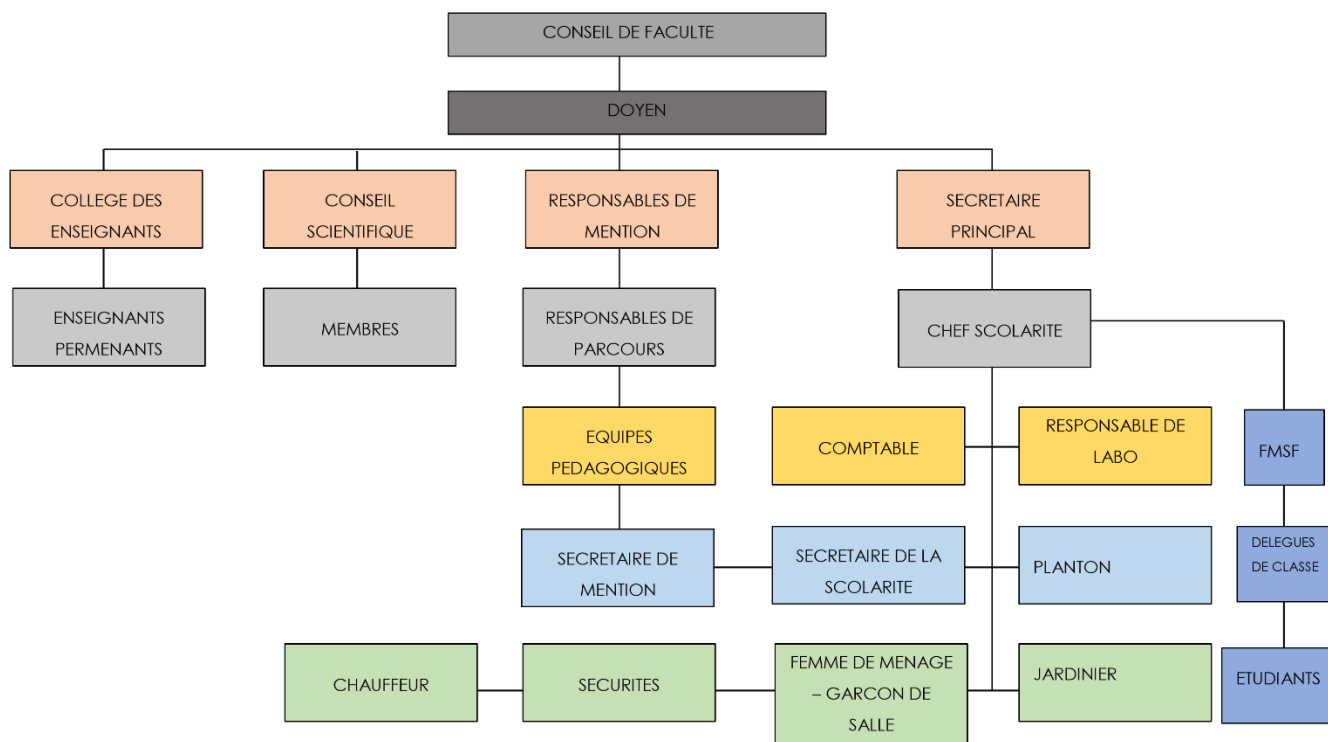


Figure 3. Organigramme de la Faculté de Science

Chapitre 3. Description de projet

Dans ce chapitre nous allons aborder une brève description du projet à effectuer, présenter les besoins de l'utilisateur, évoquer les moyens nécessaires à la réalisation du projet enfin préciser les résultats attendus du présent projet.

3.1. Formulation

L'éducation est un pilier fondamental de notre société, et son évolution a été marquée ces dernières décennies par une transformation numérique sans précédent. Les avancées technologiques ont ouvert la voie à de nouvelles opportunités d'apprentissage et de collaboration, transformant la manière dont les enseignants dispensent leurs cours et dont les étudiants accèdent aux ressources éducatives. Dans ce contexte de changement continu, la création d'une plateforme éducative novatrice devient impérative pour répondre aux besoins en constante évolution des enseignants et des étudiants.

3.2. Objectif et besoin d'utilisateur

Pour la concrétisation de ce projet, il faut d'abord définir les objectifs et les besoins des utilisateurs.

3.2.1. Objectif

L'objectif de la plateforme éducative en ligne est de créer un environnement interactif où les enseignants peuvent dispenser des cours, partager des ressources pédagogiques, publier des annonces pertinentes et évaluer les progrès des étudiants. Parallèlement, les étudiants peuvent accéder à ces ressources, consulter les annonces importantes, soumettre des devoirs et interagir avec leurs enseignants. Cette plateforme vise à favoriser l'apprentissage continu, en éliminant les barrières géographiques, en offrant un accès équitable à l'éducation et en favorisant une communication transparente au sein de la communauté éducative.

3.2.2. Besoin d'utilisateur

La plateforme envisagée doit satisfaire les besoins fonctionnels qui seront exécutés par le système et les besoins non fonctionnels qui perfectionnent la qualité logicielle du système.

➤ **Besoin fonctionnel :**

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter, il ne devient opérationnel que s'il les satisfait. Cette application doit couvrir principalement les besoins fonctionnels suivants :

- Mettre en ligne les cours et ressources pédagogiques en général
- Publier en ligne des annonces relatives aux études des étudiants
- Permettre aux étudiants de consulter et de télécharger les ressources

➤ **Besoins non fonctionnels :**

Pour pouvoir réaliser ces objectifs, on a besoin de :

- Une application web back-office qui comportera des formulaires pour les gestions de données et aussi la liste des différentes données, pour avoir une vue statistique ;
- Une application web Front Office qui comportera une interface publique pour la présentation dynamique des offres et les formulaires pour les réservations ;
- Avoir une interface simple, fluide et rapide
- Avoir des codes clairs pour permettre des futures améliorations
- Bien sécuriser l'application

3.3.Moyens nécessaires à la réalisation du projet

Pour la réalisation d'un projet informatique, le travail ne peut être effectué sans la présence de matériel et de logiciel. Donc, la nécessité de matériels est impérative.

3.3.1. Moyens humains

Les personnes qui contribuent au développement du projet sont :

- L'encadreur qui est l'initiateur ;
- Le stagiaire qui est à la fois concepteur et développeur (nous-même) ;

3.3.2. Moyens matériels

Pour la réalisation d'un projet informatique, le travail ne peut être effectué sans la présence de matériel et de logiciel. Donc, la nécessité de matériels est impérative.

Les moyens matériels que nous allons utiliser sont donc présentés dans cette partie. Pour notre part, nous aurons besoin de deux matériels qui seront détaillés par le tableau 7.

Tableau 7. Les caractéristiques des matériels utilisés pendant le projet

Machine	Processeur	Mémoire Vive(RAM)	Stockage	Carte Graphique
Ordinateur de développement	Intel Core i5-3337U @ 1, 80GHz	6Go	500 Go HDD	Intel® HD Graphics 4000

3.3.3. Moyens logiciels

Les logiciels utilisés lors de la réalisation de projet sont présentes dans le tableau 8.

Tableau 8. Moyens logiciels utilisés

LOGICIEL / LIBRAIRIE	VERSION	UTILITE
React	18.2.0	Une bibliothèque JavaScript, utilise pour la construction d'interface pour les application web
Node	18.17.1	Un environnement d'exécution cote serveur construit sur le moteur JavaScript V8 de Google chrome
Express	4.18.2	Un Framework pour simplifier le développement web cote back-end pour NodeJs
Mysql	8.0.30	Un système de base de données pour notre application
Visual Paradigm	8.0	Outils de modélisation
Visual Studio Code	1.84.2	Outils de développement
Git	2.42.0	Un système permet disposer notre code

3.4. Résultats attendus

- ✓ Conception et réalisation d'une plateforme de communication et éducation

Les attentes du projet sont donc :

- ✓ Une Plateforme Fonctionnelle : Une plateforme opérationnelle permettant aux enseignants de publier des cours, aux étudiants d'y accéder, de soumettre des devoirs, et de faciliter la communication.
- ✓ Amélioration de l'Expérience d'Apprentissage : Offrir un environnement en ligne interactif et accessible pour les étudiants, favorisant l'apprentissage continu.
- ✓ Communication Efficace : Faciliter la communication entre enseignants et étudiants via des annonces.
- ✓ Sécurité des Données : Assurer la sécurité et la confidentialité des informations partagées sur la plateforme.
- ✓ Adaptabilité et Scalabilité : Concevoir une plateforme évolutive pour répondre à un nombre croissant d'utilisateurs et de besoins éducatifs.

3.5 Chronogramme des travaux

La figure 4 représente le chronogramme des travaux.

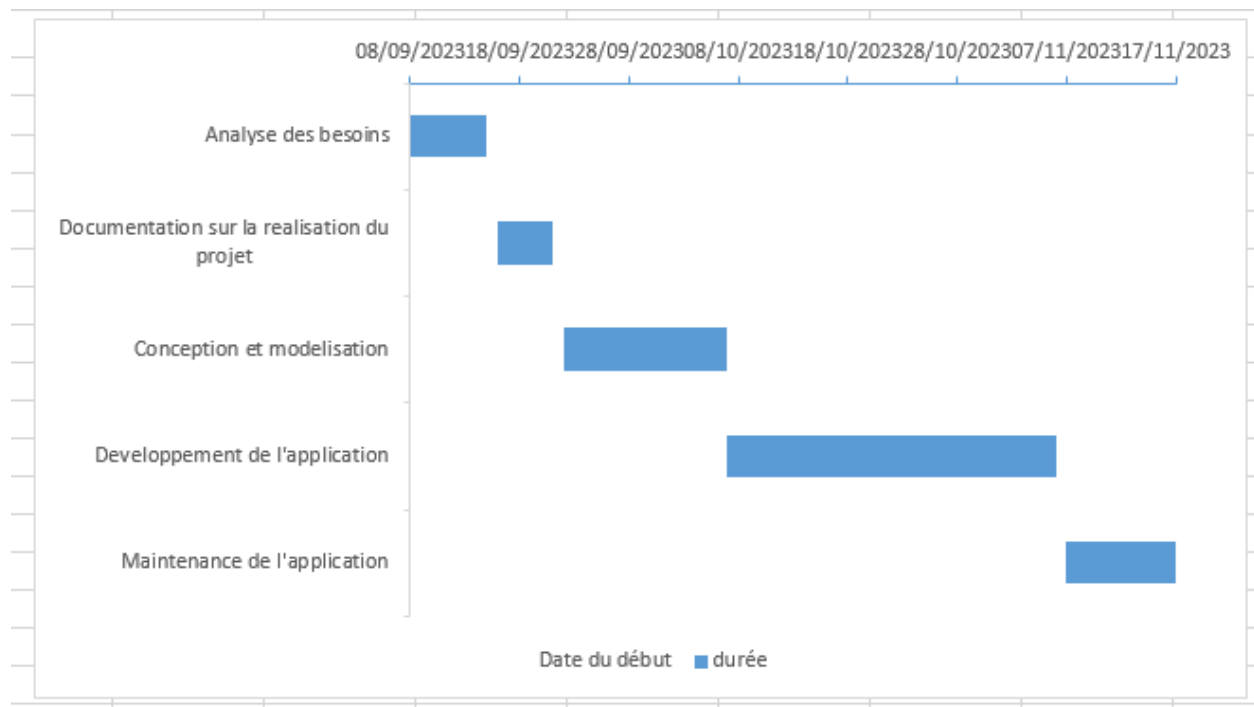


Figure 4. Chronogramme des travaux

PARTIE II: ANALYSE ET CONCEPTION

Chapitre 4. Analyse et conception

Une étape essentielle de tout cycle de développement logiciel consiste à effectuer une analyse préalable. Le but de cette phase est de savoir l'organisation actuel du système afin de dégager ses points forts ainsi que ses faiblesses et proposer des solutions pour améliorer le système actuel. Nous allons ensuite choisir parmi plusieurs outils qui y seront proposés les outils nécessaires à la création du nouveau système.

4.1. Analyse de l'existant

4.1.1. Organisation actuelle

Actuellement, les activités académique et administrative à la faculté de science sont gérées manuellement. Les enseignants distribuent les cours et les devoirs à le délégué de classe et puis le délégué de classe qui distribuent à tous les étudiants, pour soumettre des devoirs les étudiants donnent leur devoir aux délégué pour que le délégué donne aux enseignants. Les annonces importantes sont affichées sur des tableau d'affichage physiques ainsi que les résultats des examens. Les personnels implique sont les enseignants, les étudiants, le personnel administratif.

4.1.2. Inventaire des moyens matériels et logiciels

Bien que la faculté de Science n'ait pas de système préexistant, un inventaire des moyens matériels et logiciel est nécessaire pour la mise en place de la nouvelle plateforme, cela inclut les ordinateur et serveurs potentiels qui seront utilisés pour héberger la plateforme, ainsi les outils de développement, les langages de programmation, et les logiciels nécessaires pour le développement et la gestion de la plateforme.

4.2. Critique de l'existant

Après avoir analysé les organisations actuelles, nous allons maintenant définir les points faibles et forts de ce système dans le tableau 9.

Tableau 9. Critique de l'organisation actuelle

Points forts	Points faibles
<ul style="list-style-type: none">➤ La gestion manuelle actuelle offre une flexibilité essentielle, permettant une adaptation rapide au changement et aux situation imprévues	<ul style="list-style-type: none">➤ L'accessibilité aux informations académiques et administratives sont limité➤ La communication entre les enseignants, les étudiants et le

<ul style="list-style-type: none"> ➤ Pour ceux qui sont familiers avec les méthode actuelles, la simplicité d'utilisation peut être considéré comme un avantage 	<p>personnel administratif peut être entravée</p> <ul style="list-style-type: none"> ➤ La gestion manuelle des tâches administratives peut conduire à des erreurs humains, des retards et une inefficacité globale
--	---

4.3. Conception avant-projet

4.3.1. Proposition des solutions

Nous avons proposé deux solutions afin d'assurer une gestion efficace des informations académiques et des communications:

- Solution 1 : Choisir et acheter une plateforme web déjà existante qui offre des fonctionnalités prêtes à l'emploi.
- Solution 2 : Concevoir et développer une application web sur mesure, parfaitement adaptée aux exigences spécifiques de la plateforme.

le tableau 10 représente la comparaison des solutions

Tableau 10. Comparaison des solutions

SOLUTIONS	AVANTAGES	INCONVENIENTS
<u>Solution 1</u> : Choisir et acheter une plateforme web déjà existante qui offre des fonctionnalités prêtes à l'emploi.	<ul style="list-style-type: none"> ➤ La plateforme préexistante permet un déploiement plus rapide ➤ Economises des temps en utilisant des fonctionnalités déjà développées ➤ Moins de ressources nécessaire par rapport au 	<ul style="list-style-type: none"> ➤ Les fonctionnalités ne peuvent pas être adaptées à tous les besoins spécifiques ➤ Des couts initiaux peuvent être associés a l'acquisition de la plateforme

	développement personnalisé	
<u>Solution 2</u> : Concevoir et développer une application web sur mesure, parfaitement adaptée aux exigences spécifiques de la plateforme.	<ul style="list-style-type: none"> ➤ Contrôle total sur la conception pour répondre précisément aux besoins spécifiques ➤ Toutes les fonctionnalités peuvent être intégrées en fonction des exigences 	<ul style="list-style-type: none"> ➤ La conception et développement personnalisés peuvent prendre plus de temps.

Après avoir comparé ces deux solutions, on a opté pour la deuxième car la première nécessite des ressources humaines assez coûteuses.

4.3.2. Méthodes de conception et outils utilisés

4.3.2.1. Méthodes

En effet, la conception d'un système d'information n'est pas évidente parce qu'il faut réfléchir à l'ensemble de l'organisation que l'on doit mettre en place. La phase de conception nécessite des méthodes permettant de mettre en place un modèle sur lequel nous allons appuyer. La modélisation consiste à créer une représentation virtuelle d'une réalité de telle façon à faire ressortir les points auxquels nous nous intéressons. Ce type de méthode est appelé analyse. Il existe plusieurs méthodes d'analyse. Dans le tableau 11 la comparaison de quelques méthodes

Tableau 11. Comparaison de 2TUP et Merise

Méthodes	2TUP	Merise
Avantages	<ul style="list-style-type: none"> - Itératif - Centré sur l'architecture - Piloté par le cas d'utilisation d'UML - Architecture bidirectionnelle - Modélisation Objet - Indépendant d'un langage de 	<ul style="list-style-type: none"> - Structuration en étape et en point de contrôle - Approche conceptuelle, se focalisant sur les métiers et les besoins associés - Approche systématique

	programmation - Description dynamique du système - Association des traitements et des données qui facilite l'évolutivité du logiciel -Maitrisé	
Inconvénients	- La mise en pratique nécessite un apprentissage - L'intégration dans un processus n'est pas triviale	- Ne s'occupe pas de l'interface Utilisateur - Ne prend pas en compte la maintenance ni la deuxième informatisation - Méthode d'analyse spécialement conçue pour l'informatique de gestion - Etude séparées des données et des traitements

Après l'analyse de ces 2 méthodes de conception, la méthode 2TUP m'a paru le mieux adapté pour ce projet en vue de ces nombreux avantages. Et on a choisi UML comme langage de modélisation.

Présentation de 2TUP

2TUP signifie « 2 Track Unified Process ». C'est un processus UP. Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes. « 2 Track » signifie littéralement que le processus suit deux chemins. Il s'agit des chemins « fonctionnels » et « d'architecture technique », qui correspondent aux deux axes de changement imposés au système informatique.

4.3.2.2. Outils utilisés

Pour réaliser la solution qui a été adoptée précédemment, nous avons besoin des outils suivants :

- Un Framework de développement pour la partie interface de la plateforme.
- Un langage de programmation pour mettre en place la partie traitement.
- Un SGBD pour la gestion de la base de données.

- Un environnement de développement ou un éditeur.
- Un outil de modélisation.

- **Choix du Framework**

Il existe plusieurs Framework ou bibliothèque pour le développement d'application web. Une brève comparaison de quelques-uns de ces Framework est dressée dans le tableau 12.

Tableau 12. Comparaison des Framework

	Angular	React	Vue.js
Performance	moyen	haute	haute
Scalabilité	haute	haute	faible
Apprentissage	difficile	moyen	facile
Disponibilité des développeurs	haute	haute	faible
Communauté des développeurs	grande	très grande	petite
Acceptation et confiance	haute	très haute	faible

Cependant nous avons choisi d'utiliser le Framework **REACT** car il offre une architecture flexible et des fonctionnalités robustes qui correspondent à nos besoins spécifiques.

Présentation de React

React est une bibliothèque JavaScript développée par Facebook, se concentrant sur la construction d'interface utilisateur interactives et réactive. React met l'accent sur la création des composants réutilisable pour faciliter le développement modulaire. Souvent utilise de manière incrémentielle dans des projets existants, React peut être associée à un serveur NodeJs avec ExpressJs pour pile de développement performante.

- **Choix de langage de programmation**

Les langages de programmation (côté serveur) à choisir pour développer une application sont nombreuses. Le tableau 13 illustre une comparaison de quelques-uns de ces langages

Tableau 13. Comparaison entre NodeJs et PHP

LANGAGE	NodeJs	PHP
Avantages	<ul style="list-style-type: none">➤ Prise en charge de multithreading➤ Syntaxe unique pour cote clients et serveurs➤ Prise en charge intégré du serveur➤ Rapide	<ul style="list-style-type: none">➤ Facilité de développement➤ Prend en charge plus de Framework et CMS➤ Prise en charge par tous les services d'hébergements internet
Inconvénient	<ul style="list-style-type: none">➤ Requiert une bonne compréhension sur les fonctions de rappel et blocages➤ Configuration de l'environnement de développement difficile	<ul style="list-style-type: none">➤ Lenteur

Ainsi, nous avons choisi le langage **Node.js** en raison de sa rapidité.

Présentation de NodeJS

NodeJS est une plateforme de développement JavaScript. Ce n'est pas un serveur, ce n'est pas un Framework, c'est juste le langage JavaScript avec des bibliothèques permettant de réaliser des actions comme écrire sur la sortie standard, ouvrir/fermer des connections réseau ou encore créer un fichier. Il est souvent confondu avec un serveur car c'est son origine : Node.js a été créé par Ryan Dahl dans le but de pouvoir créer des applications temps réel où le serveur est capable de pousser de l'information au client. C'est dans ce but qu'il utilise la bibliothèque libuv pour réaliser son modèle d'entrée sortie non bloquante.

- **Choix de SGBD**

Une brève comparaison de quelques SGBD est dressée dans le tableau 14.

Tableau 14. Comparaison de SGBD

SGBD	MySQL	PostgreSQL	MongoDB
Avantages	<ul style="list-style-type: none"> ➤ Open source ➤ Très populaire ➤ Facile à utiliser, ➤ Il possède aussi une grande communauté. ➤ Déploiement facile 	<ul style="list-style-type: none"> ➤ Open source ➤ Base sur un modèle orienté objet ➤ Pas de point de défaillance unique 	<ul style="list-style-type: none"> ➤ Agile et évolutive ➤ Donne non structurer
Inconvénients	<ul style="list-style-type: none"> ➤ Base sur le modèle relationnel classique ➤ Trop lent 	<ul style="list-style-type: none"> ➤ Consomme beaucoup de ressources 	<ul style="list-style-type: none"> ➤ Donnée brute ➤ No SQL

Ainsi, nous avons choisi **MySQL** en raison de sa robustesse, de sa fiabilité et de sa capacité à gérer efficacement les bases de données relationnelles.

Représentation de MySQL

MySQL est un système de gestion de base de données relationnelles qui organise les données de manière tabulaire. Dans MySQL, une base de données peut contenir plusieurs tables, chaque table représentant une entité spécifique avec des colonnes décrivant les attributs de cette entité. Par exemple, une table "Utilisateurs" pourrait avoir des colonnes telles que "ID_utilisateur", "Nom", "Prénom", "Age", et "Email". Chaque ligne dans cette table représente un utilisateur spécifique avec des valeurs correspondantes pour chaque attribut. MySQL offre une solution robuste et bien établie pour la gestion efficace de bases de données relationnelles, avec une large adoption dans l'industrie en raison de sa fiabilité, de ses performances et de sa capacité à s'intégrer facilement à diverses applications.

- **Choix de l'environnement de développement(IDE)**

Le tableau 15 illustre une comparaison de quelques environnements de développement.

Tableau 15. Comparaison entre Visual Studio Code et Webstorm

ENVIRONNEMENT DE DEVELOPPEMENT	VISUAL STUDIO CODE	WEB STORM
Avantages	<ul style="list-style-type: none"> ➤ Rapide ➤ Leger 	<ul style="list-style-type: none"> ➤ Dispose de nombreux plugins installes ➤ Riche en fonctionnalités
Inconvénients	<ul style="list-style-type: none"> ➤ Nécessite l'installation de certains plugins ➤ Fonctionnalités limites 	<ul style="list-style-type: none"> ➤ Lent ➤ Consomme beaucoup de ressource

Ainsi, nous avons choisi **Visual Studio Code** en raison de sa rapidité à fonctionner.

Présentation de Visual Studio Code

Visual Studio Code (VSC par la suite) est un éditeur de code open-source, gratuit et multiplateforme (Windows, Mac et Linux), développé par Microsoft, à ne pas confondre avec Visual Studio, l'IDE propriétaire de Microsoft. VSC est développé avec Electron et exploite des fonctionnalités d'édition avancées du projet Monaco Editor. Principalement conçu pour le développement d'application avec JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.

- **Choix de l'outil de modélisation**

Le tableau 16 illustre une brève comparaison de quelques outils de modélisation.

Tableau 16 : Comparaison entre UML Designer et Visual Paradigm for UML

Nom	Editeur	Plateforme	Intégré ou autonome	Cout /licence	Ergonomie	Types des graphique prise en charges	autres
UML Designer	Obeo	Multiplate forme	Integrable dans ecilipse	Open source EPL	Ouverte	-cas d'utilisation -classe -collaboration	Possible de créer ses

						-déploiement -séquence -Etat -activité -composant	propres diagrammes
Visual Paradigm for UML	Visual Paradigm	Multiplate forme	Intégrable dans Eclipse, Netbeans, IntelliJ	Libre	Dispose de différentes fonctions facilitant la conception des diagrammes aux débutants	-Cas d'utilisation -Classe -Objet -Etat -Activité -Séquence -Communication -Composant -Déploiement -package -Structure Composite -chronogramme	Génération de code source à partir des diagrammes - Relation du mapping objet relationnel automatiquement à partir des diagrammes de classes

Nous avons alors choisi **Visual Paradigm for UML** parce que c'est un outil complet et d'une facilité d'utilisation.

Présentation de Visual Paradigm

Le logiciel de modélisation UML Visual Paradigm for UML est un bon outil pour réaliser des spécifications. Possédant de nombreuses fonctionnalités dans ses versions payantes, il possède une interface intuitive et est rapidement pris en main.

Chapitre 5. Analyse Conceptuel

Dans cette partie, nous allons établir la conception de base de l'application à développer.

5.1. Présentation de la méthode utilisée

Langage de modélisation :

La modélisation est une activité technique qui s'inscrit dans de nombreux processus d'ingénierie. Son but est de fournir une représentation approchée du système ou du produit que l'on veut analyser, concevoir ou fabriquer. En génie logiciel, nous modélisons les logiciels à partir de diagrammes et des modèles qui représentent l'architecture des aspects différents du logiciel à développer. Pour ce faire, il existe des méthodes de notation qu'on peut utiliser.

Pour notre part, nous allons utiliser UML.

Présentation de 2TUP :

« 2 Tracks Unified Process » ou 2TUP est un processus de développement logiciel construit sur UML et qui implémente le processus unifié. Le processus unifié permet de subdiviser en 4 phases (pré-étude, élaboration, construction et transition) le développement d'un logiciel éclairci dans la 05.

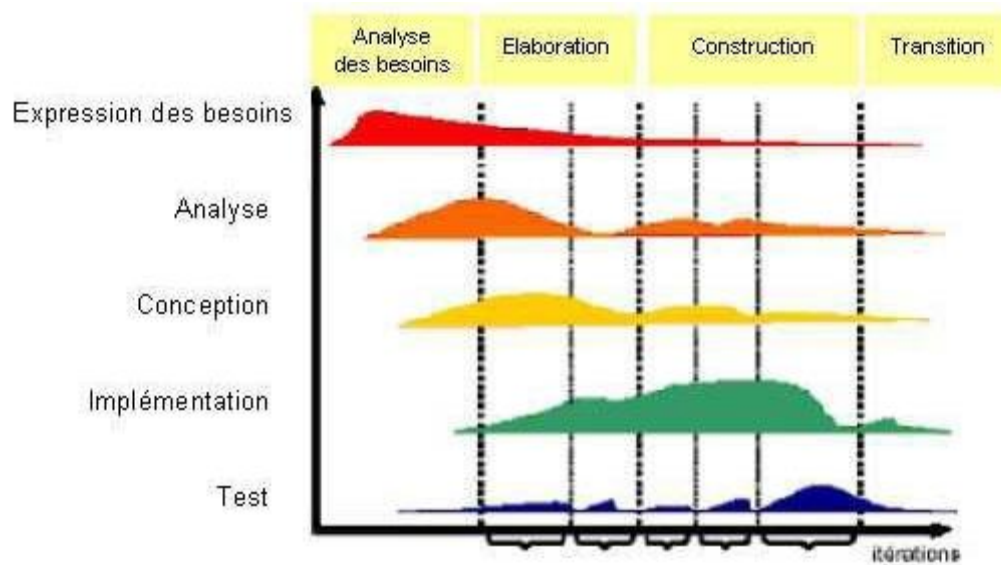


Figure 5. Phase de développement du processus unifié

Il est un processus de développement en forme de Y, comme illustré par la 06

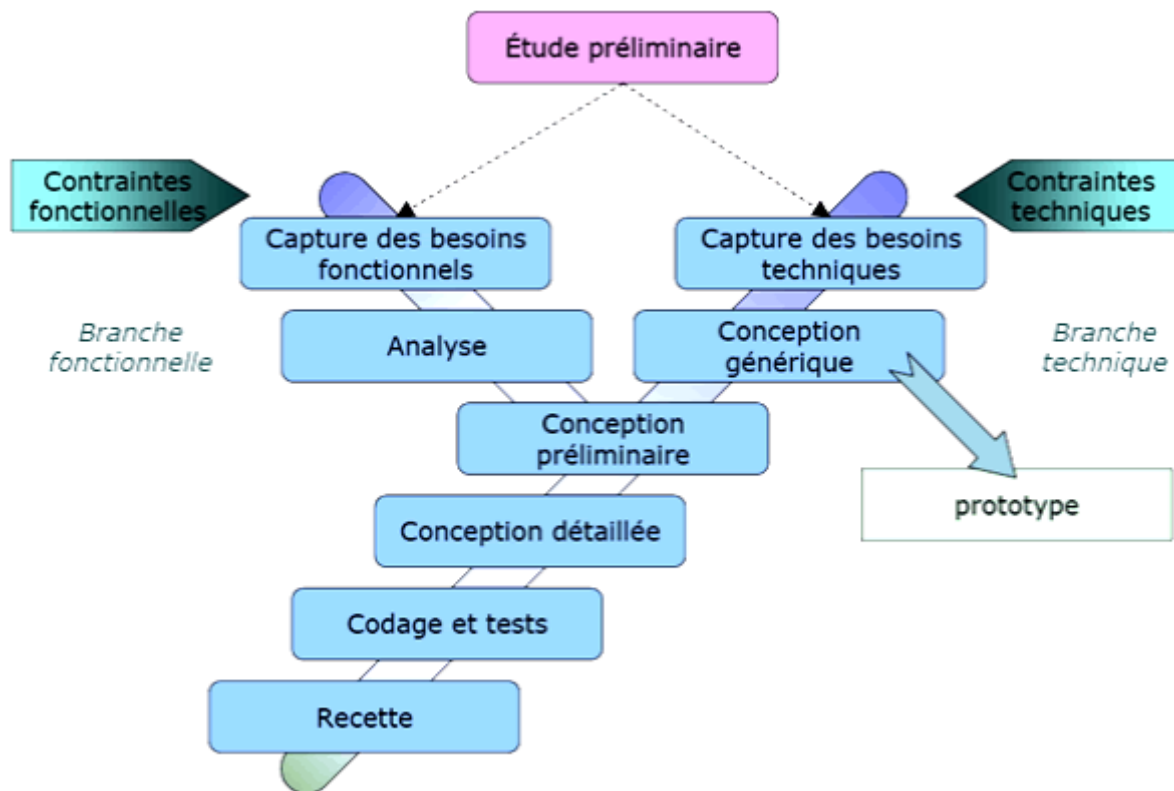


Figure 6. Processus de développement en Y

5.2. Dictionnaire des données

Le dictionnaire des données est la représentation détaillée de toutes les données élémentaires qui constitueront l'ensemble utilisé pour un projet informatique. Le dictionnaire est établi après l'analyse approfondie du domaine d'étude en recueillant chaque donnée qui sera utilisée pour le projet et en définissant la façon avec laquelle elle sera représentée dans le projet. Le tableau 7 représente ces dictionnaires des données.

Tableau 17. Dictionnaire des données

NOM	DESCRIPTION	TYPE	TAILLE	COMMENTAIRE
Annonce	Fichier contient les annonces	AN	255	
Cours	Fichier contient les cours	AN	255	
Date_de_publication	Le date de publication des annonces	D	10	jj-mm-aaaa
Date_de_creation	Le date de création des cours	D	10	jj-mm-aaaa

Date_de_devoirs	Le date de la publication des devoirs	D	10	jj-mm-aaaa
Date_de_notification	Le date de création des notification	D	10	jj-mm-aaaa
Date_de_soumission	Le date de soumission des devoirs	D	10	jj-mm-aaaa
Description	Description des annonces, des cours et des devoirs	AN	100	
Devoirs	Fichier qui contient les devoirs	AN	255	
Email	Email des utilisateur	AN	50	
Libelle	Libelle des cours ou des devoirs	AN	20	
Matricule	Matricule des étudiants	AN	10	
Mention	Mention des étudiants	AN	20	
Mot_de_passe	Mot de passe des utilisateurs	AN	150	
Niveau	Niveau des étudiants	AN	10	
Nom	Noms des utilisateurs	AN	50	
Parcours	Parcours des étudiants	AN	20	
Prenom	Prénoms des utilisateurs	AN	50	
Roles	Rôle des utilisateurs	AN	10	
Statut	La statut de la notification lu ou non lu	B	10	
Telephone	Numéro de téléphone des utilisateurs	AN	13	
Type	Type de la notification soit annonce, message, cours, ou devoir	AN	10	

AN : Alpha Numérique

N : Numérique

D : Date

B : Booléenne

5.3. Règles de gestions

Les règles de gestion sont des informations de commande qui définissent la suite des opérations effectuées pour passer des données de base au résultat. Ainsi, ils indiquent l'ensemble des liens existants sur les données manipulées, les actions effectuées par les postes de travail impliqués et les différentes formules de calcul. Les règles de gestion associées au domaine d'étude sont :

- RG1 : L'Administrateur créer les comptes des professeurs
- RG2 : L'Administrateur ou Professeur publier des annonces ou des informations importantes
- RG3 : L'Administrateur ou Professeurs peuvent annuler ou modifier les annonces ou l'information
- RG4 : Les Professeurs créer les cours pour les étudiants selon leur niveau
- RG5 : Les Professeurs peuvent annuler ou modifier les cours
- RG6 : Les Professeurs donne des devoirs pour les étudiants selon leur niveau
- RG7 : Les Professeurs peuvent annuler ou modifier les devoirs
- RG8 : Les étudiants créent son compte
- RG9 : Les étudiants peuvent voir ou télécharger les annonces ou les informations
- RG10 : Les étudiants peuvent voir ou télécharger les cours selon leur niveau
- RG11 : Les étudiants peuvent voir ou télécharger les devoirs selon leur niveau
- RG12 : L'Administrateur peuvent supprimer les comptes des professeurs ou étudiants.
- RG13 : Les utilisateurs notifier par le système.
- RG14 : Les étudiants soumettre des devoirs

5.4. Représentation et spécification des besoins

Maintenant, nous allons spécifier les besoins des utilisateurs. Pour cela, il faut que nous analysions cela par l'intermédiaire des diagrammes des cas d'utilisation.

5.4.1. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation donne une vision globale des communications acteurs / domaine. Il met en jeu les concepts suivants :

Acteur : Participant externe qui interagit avec le système, il représente les cas les plus importants du système en cours d'utilisation.

Cas d'utilisation : Il modélise une interaction entre le système informatique à développer et un utilisateur ou acteur interagissant avec le système. Plus précisément, un cas d'utilisation décrit une séquence d'un résultat observable pour un acteur.

Relation entre cas d'utilisation : La relation d'inclusion (**include**), quand le cas d'utilisation de base en incorpore explicitement un autre, de façon obligatoire ; Une relation d'extension (**extend**) d'un cas d'utilisation A par un cas d'utilisation B signifie qu'une instance de B peut être étendue par le comportement décrit dans A

La 07 indique le concept d'une représentation d'un cas d'utilisation.

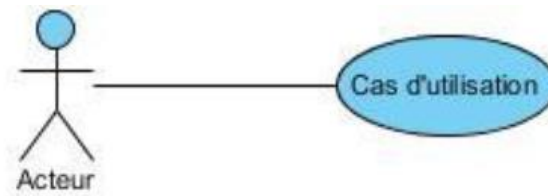


Figure 7. Formalisme d'un cas d'utilisation

➤ **S'authentifier**

L'authentification est une étape cruciale pour garantir la sécurité du système. Ce cas d'utilisation permet d'identifier l'utilisateur courant et de lui fournir les droits d'accès aux fonctionnalités qui lui sont attribués.

➤ **Publication des annonces et information**

Dans ce cas d'utilisation, les administrateurs et professeurs sont habilités à et diffuser diverses informations liées aux programmes d'études, aux cours et aux activités académiques des étudiants.

➤ **Création du compte des professeurs**

Dans ce cas d'utilisation, l'administrateur a pour responsabilité de créer les comptes des professeurs. Cette étape est essentielle pour permettre aux professeurs d'accéder à la plateforme d'interagir avec les étudiants, de gérer des cours et d'utiliser d'autres fonctionnalité liées à l'enseignement.

➤ **Création des devoirs**

Dans ce cas d'utilisation, les professeurs jouent un rôle essentiel en concevant et en proposant des cours sur mesure aux étudiants, en prenant compte de leur niveau études, leur mention académique ainsi que leurs parcours d'études.

➤ **Création des devoirs**

Dans ce cas d'utilisation, les professeurs attribuent des devoirs aux étudiants en tenant compte de leur niveau d'étude, de leur mention académique et de leur parcours éducatif.

➤ **Consultation des annonces et informations**

Dans ce cas d'utilisation, les étudiants ont la possibilité de consulter des annonces et des informations liées à leurs études. Permet aux étudiants d'accéder à des communications importantes et des informations pertinentes pour leurs parcours académiques.

➤ **Consultation des cours**

Dans ce cas d'utilisation, les étudiants ont la possibilité d'accéder aux cours en fonction de leur niveau d'études, de leur mention académique et de leur parcours d'études, cette approche vise à personnaliser l'expérience d'apprentissage des étudiants en leur permettant de suivre des cours qui correspondent à leur niveau.

➤ **Soumission des devoirs**

Dans ce cas d'utilisation, les étudiants ont la possibilité de soumettre leurs réponses aux devoirs qui leur sont assignés par les professeurs, cette étape est essentielle pour l'évaluation et la rétroaction des étudiants.

➤ **S'inscrire**

Dans ce cas d'utilisation, les étudiants s'inscrivent dans la plateforme afin de bénéficier de l'accès à l'ensemble des ressources qui y sont disponibles. Cette inscription leur permet de tirer pleinement parti des fonctionnalités et informations offertes par la plateforme.

➤ **Gérer profil**

Dans ce cas d'utilisation, les utilisateurs ont la possibilité de gérer leur profil personnel au sein de la plateforme. Cette fonctionnalité leur permet de personnaliser et de mettre à jour les informations relatives à leur compte.

➤ **Notifier les utilisateurs**

Dans ce cas d'utilisation, la fonction consiste à notifier les utilisateurs de toute information les concernant, tels que de nouveaux messages, des annonces, des cours, des devoirs ainsi que la date limite de remise des devoirs. Ceci vise à faciliter l'utilisation de la plateforme par les utilisateurs en les tenant informés des événements pertinents.

La figure 8 représente le diagramme de cas d'utilisation

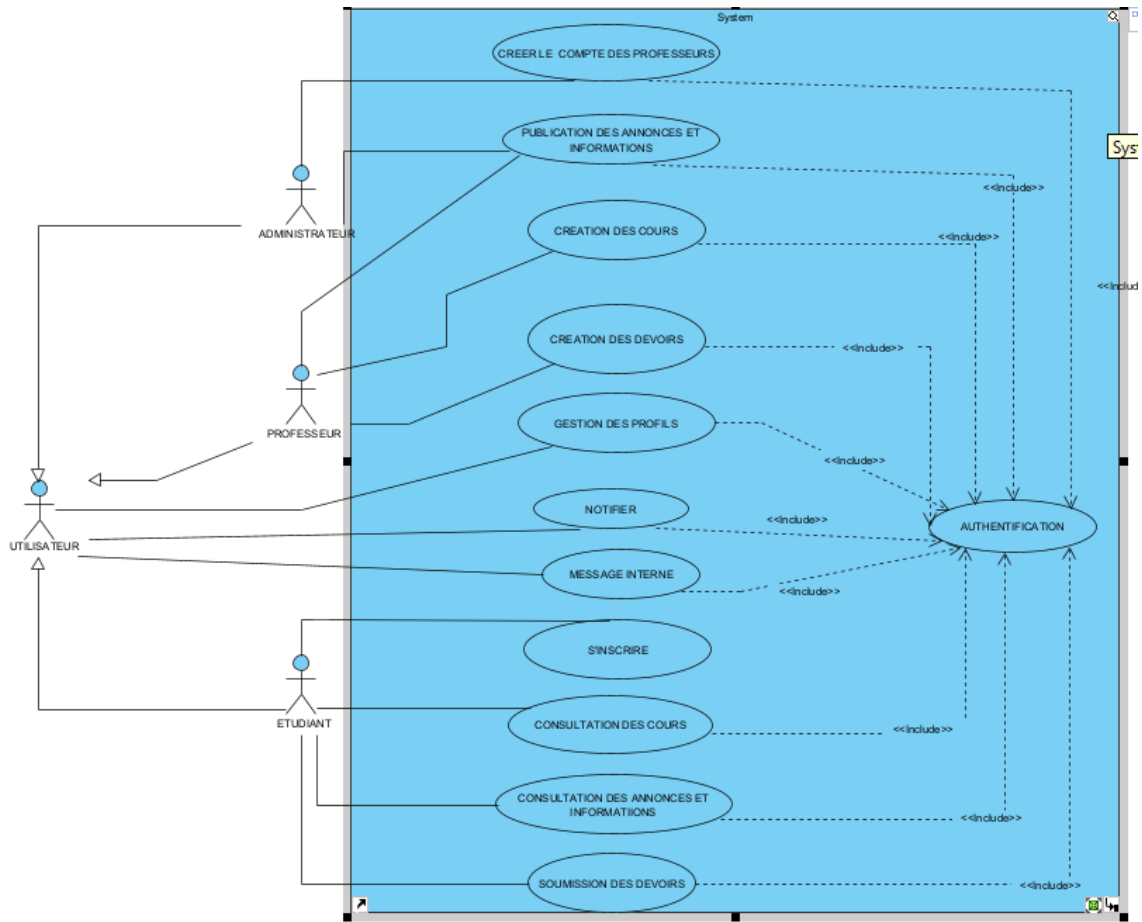


Figure 8. Diagramme de cas d'utilisation

5.4.2 Description textuelle pour chaque cas d'utilisation

Chaque cas d'utilisation doit être décrit sous forme textuelle afin de bien identifier les traitements à réaliser par le système en vue de la satisfaction du besoin exprimé par l'acteur. Nous allons cependant décrire textuellement chaque cas d'utilisation.

➤ Cas d'utilisation : S'authentifier

- **Acteur principaux :** Administrateur, Professeurs, Etudiants
- **Acteur secondaire :** Système
- **Précondition :** Les utilisateurs possèdent un compte enregistré
- **Début :** Les utilisateurs accèdent à la page de connexion du système.
- **Postcondition :** La validation de l'authentification confirme l'existence du compte utilisateur.

- **Fin** : Redirection vers la page d'accueil après authentification réussie.
- **Scenario nominal** :
 1. Les utilisateurs initient le processus d'authentification. Le système envoie la page d'authentification
 2. Le système répond en affichant la page d'authentification. Le système vérifie les données
 3. Les utilisateurs saisissent leurs identifiants de connexion.
 4. Le système authentifie les informations fournies.
 5. Confirmation de l'authentification par le système.

➤ **Cas d'utilisation : S'inscrire**

- **Acteur principaux** : Etudiants
- **Acteurs secondaire** : Système
- **Précondition** : Aucune inscription préalable n'est associée à l'étudiant.
- **Début** : L'étudiant accède à la page d'inscription ou au formulaire d'inscription du système.
- **Postcondition** : L'enregistrement des informations de l'étudiant crée un compte dans le système.
- **Fin** : Confirmation de l'inscription et accès accordé au système en tant qu'étudiant.
- **Scénario nominal** :
 1. L'étudiant accède à l'option d'inscription.
 2. Le système affiche le formulaire d'inscription.
 3. L'étudiant remplit les champs requis avec ses informations personnelles.
 4. L'étudiant soumet le formulaire d'inscription.
 5. Le système enregistre les données fournies et crée un compte pour l'étudiant.
 6. Confirmation de la réussite de l'inscription.

➤ **Cas d'utilisation : Créer compte professeur**

- **Acteur principaux** : Administrateur
- **Acteurs secondaire** : Système
- **Précondition** : L'administrateur est connecté au système et a les autorisations pour créer des comptes.

- **Début** : L'administrateur accède à la section de création de compte professeur.
- **Postcondition** : Le compte professeur est créé et associé au système.
- **Fin** : Confirmation de la création du compte professeur par l'administrateur
- **Scénario nominal** :
 1. L'administrateur accède à l'option de création de compte professeur.
 2. Le système affiche le formulaire de création de compte.
 3. L'administrateur saisit les détails requis du professeur (nom, identifiant, mot de passe, etc.).
 4. L'administrateur envoie les informations pour créer le compte.
 5. Le système enregistre les données fournies et crée le compte professeur.
 6. Confirmation de la création réussie du compte

➤ **Cas d'utilisation : Publier des annonces et informations**

- **Acteur principaux** : Administrateur, Professeurs
- **Acteurs secondaire** : Système
- **Précondition** : Les professeurs et l'administrateur sont connectés au système et ont les droits pour publier des annonces.
- **Début** : L'acteur (professeur ou administrateur) accède à l'interface de publication d'annonces.
- **Postcondition** : L'annonce est publiée et visible pour les utilisateurs du système.
- **Fin** : Confirmation de la publication réussie de l'annonce par l'acteur.
- **Scénario nominal** :
 1. L'acteur (professeur ou administrateur) accède à l'option de publication d'annonces.
 2. Le système affiche l'interface de publication.
 3. L'acteur rédige le contenu de l'annonce, y ajoute des détails et éventuellement des médias.
 4. L'acteur soumet l'annonce pour publication.
 5. Le système enregistre l'annonce dans la base de données du système.
 6. Confirmation de la publication réussie de l'annonce.

➤ **Cas d'utilisation : Création des cours**

- **Acteur principaux :** Professeurs
- **Acteurs secondaire :** Système
- **Précondition :** Le professeur est connecté au système et possède les autorisations nécessaires pour créer des cours.
- **Début :** Le professeur accède à l'interface de création de cours.
- **Postcondition :** Le cours est créé et répertorié dans le système.
- **Fin :** Confirmation de la création réussie du cours par le professeur.
- **Scénario nominal :**
 1. Le professeur accède à l'option de création de cours. Le système affiche le formulaire d'inscription.
 2. Le système affiche l'interface de création de cours.
 3. Le professeur renseigne les détails du cours : titre, description, objectifs, ressources, etc.
 4. Le professeur soumet les informations pour créer le cours.
 5. Le système enregistre les détails du cours dans la base de données.
 6. Confirmation de la création réussie du cours.

➤ **Cas d'utilisation : Création des devoirs**

- **Acteur principaux :** Professeurs
- **Acteurs secondaire :** Système
- **Précondition :** Le professeur est connecté au système et possède les autorisations nécessaires pour créer des devoirs.
- **Début :** Le professeur accède à l'interface de création de devoirs.
- **Postcondition :** Le devoir est créé et disponible pour les étudiants dans le système.
- **Fin :** Confirmation de la création réussie du devoir par le professeur.
- **Scénario nominal :**
 1. Le professeur accède à l'option de création de devoirs.
 2. Le système affiche l'interface de création de devoirs.
 3. Le professeur définit les détails du devoir : titre, description, date limite, consignes, etc.

4. Le professeur soumet les informations pour créer le devoir.
5. Le système enregistre les détails du devoir dans la base de données.
6. Confirmation de la création réussie du devoir.

➤ **Cas d'utilisation : Consultation des annonces et informations**

- **Acteur principaux :** Etudiants
- **Acteurs secondaire :** Système
- **Précondition :** L'étudiant est connecté au système et a l'autorisation de consulter les annonces.
- **Début :** L'étudiant accède à l'interface de consultation des annonces.
- **Postcondition :** L'étudiant visualise les annonces disponibles dans le système.
- **Fin :** Confirmation de la consultation réussie des annonces par l'étudiant.
- **Scénario nominal :**
 1. L'étudiants accède à la menu annonce.
 2. Le système affiche l'interface présentant les annonces disponibles.
 3. L'étudiant consulte les différentes annonces disponibles.
 4. L'étudiant peut lire les détails, les dates, les informations supplémentaires des annonces.

➤ **Cas d'utilisation : Consultation des cours**

- **Acteur principaux :** Etudiants
- **Acteurs secondaire :** Système
- **Précondition :** L'étudiant est connecté au système et a l'autorisation de consulter les cours selon différents critères.
- **Début :** L'étudiant accède à l'interface de consultation des cours selon le niveau, la mention ou le parcours.
- **Postcondition :** L'étudiant visualise les cours correspondant à son niveau, sa mention ou son parcours.
- **Fin :** Confirmation de la consultation réussie des cours par l'étudiant.
- **Scénario nominal :**
 1. L'étudiant accède au menu pour consulter les cours.

2. Le système affiche la liste des cours correspondant aux critères sélectionnés (niveau, mention, parcours).
3. L'étudiant peut consulter les détails des cours disponibles.

➤ **Cas d'utilisation : Soumission des devoirs**

- **Acteur principaux :** Etudiants
- **Acteurs secondaire :** Système
- **Précondition :** L'étudiant est connecté au système et a l'autorisation de consulter les devoirs selon différents critères après avoir consulté l'étudiant soumet la réponse des devoirs.
- **Début :** L'étudiant accède à l'interface de soumission des devoirs selon le niveau, la mention ou le parcours.
- **Postcondition :** L'étudiant visualise les devoirs correspondant à son niveau, sa mention ou son parcours.
- **Fin :** Confirmation de la soumission réussie des devoirs par l'étudiant.
- **Scénario nominal :**
 1. L'étudiant accède dans la devoir.
 2. Le système affiche la liste des devoirs
 3. L'étudiant soumet la réponse de devoirs.
 4. Le système affiche l'interface pour soumettre.
 5. Confirmation de la soumission de devoirs.

➤ **Cas d'utilisation : Gérer son profil**

- **Acteur principaux :** Utilisateurs (Étudiants, Professeurs, Administrateurs)
- **Acteurs secondaire :** Système
- **Précondition :** L'utilisateur est connecté au système et a accès à la fonction de gestion de profil.
- **Début :** L'utilisateur accède à l'option de gestion de profil.
- **Postcondition :** Les informations du profil sont mises à jour selon les actions de l'utilisateur.
- **Fin :** Confirmation de la mise à jour du profil par l'utilisateur.
- **Scénario nominal :**

1. L'utilisateur accède à l'option de gestion de profil dans la paramètre.
2. Le système affiche les informations du profil actuel de l'utilisateur.
3. L'utilisateur peut modifier des détails tels que ses informations personnelles, mot de passe, photo de profil, etc.
4. L'utilisateur effectue les modifications souhaitées.
5. Le système enregistre les mises à jour du profil de l'utilisateur.
6. Confirmation de la mise à jour réussie du profil.

➤ **Cas d'utilisation : Notifier les utilisateurs**

- **Acteur principaux :** Système
- **Acteurs secondaire :** Utilisateurs (Étudiants, Professeurs, Administrateurs)
- **Précondition :** Une nouvelle annonce est créée et prête à être diffusée.
- **Début :** Le système détecte la création d'une nouvelle annonce.
- **Postcondition :** Les utilisateurs reçoivent une notification.
- **Fin :** Envoi des notifications aux utilisateurs concernés.
- **Scénario nominal :**
 1. Le système détecte des informations (message, devoir remis, des annonces, des nouveaux cours ou devoirs).
 2. Le système identifie les utilisateurs concernés par la notification (tous les étudiants, par exemple).
 3. Le système envoie une notification à tous les utilisateurs appropriés.
 4. Les utilisateurs reçoivent la notification de la.
 5. Les utilisateurs peuvent accéder à l'annonce pour obtenir plus de détails.

5.4.3 Priorisation des cas d'utilisation

Puisque nous venons d'élaborer le diagramme des cas d'utilisations, nous allons maintenant déterminer la priorité de chacun de ces cas d'utilisation.

Tableau 18. Priorisation des cas d'utilisation

CAS D'UTILISATION	PRIORITE
S'authentifier	Élevée
S'inscrire	Élevée
Faire des annonces	Moyenne

Création des cours	Élevée
Création des devoirs	Élevée
Consultation des annonces	Moyenne
Accéder à des cours et devoirs	Élevée
Notifier les utilisateurs	Moyenne
Gérer son profil	Moyenne

5.4.4 Diagramme de séquence de système pour chaque cas d'utilisation

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.

L'ordre d'envoi d'un message est déterminé par sa position sur l'axe vertical du diagramme. La disposition des objets sur l'axe horizontal n'a pas de conséquence pour la sémantique du diagramme. Il existe un diagramme correspondant pour chaque cas d'utilisation.

ref : sous-séquence détaillée dans un autre diagramme de séquence

opt : sous-séquence optionnelle exécutée si condition de garde est vraie

loop ; Le fragment de séquence loop permet d'itérer un traitement un nombre maximum de fois jusqu'à une condition qui peut faire sortir de la boucle avant que le nombre de fois maximum ne soit atteint.

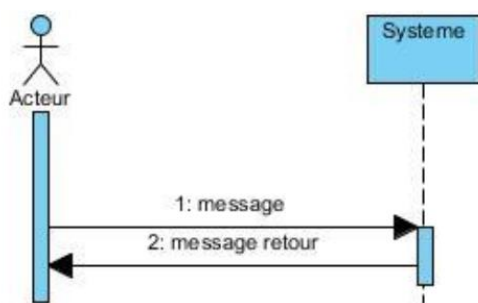


Figure 9. Formalisme d'un diagramme de séquence

➤ Cas d'utilisation : S'inscrire

La figure 10 représente le diagramme de séquence système pour le cas d'utilisation « S'inscrire ».

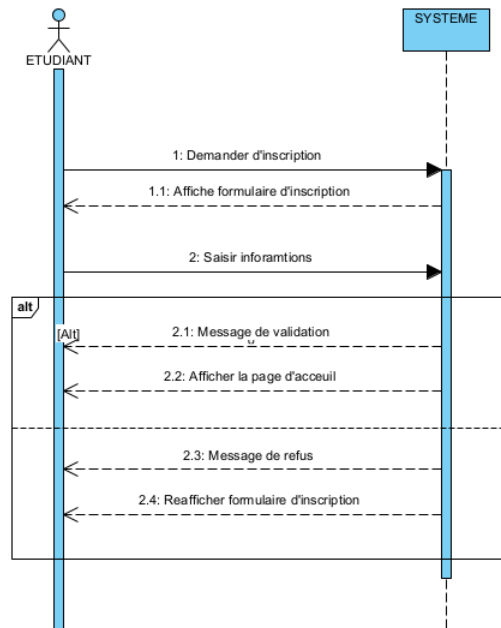


Figure 10. Diagramme de séquence de système : S'inscrire

➤ Cas d'utilisation : s'authentifier

La figure 11 représente le diagramme de séquence système pour le cas d'utilisation « S'authentifier ».

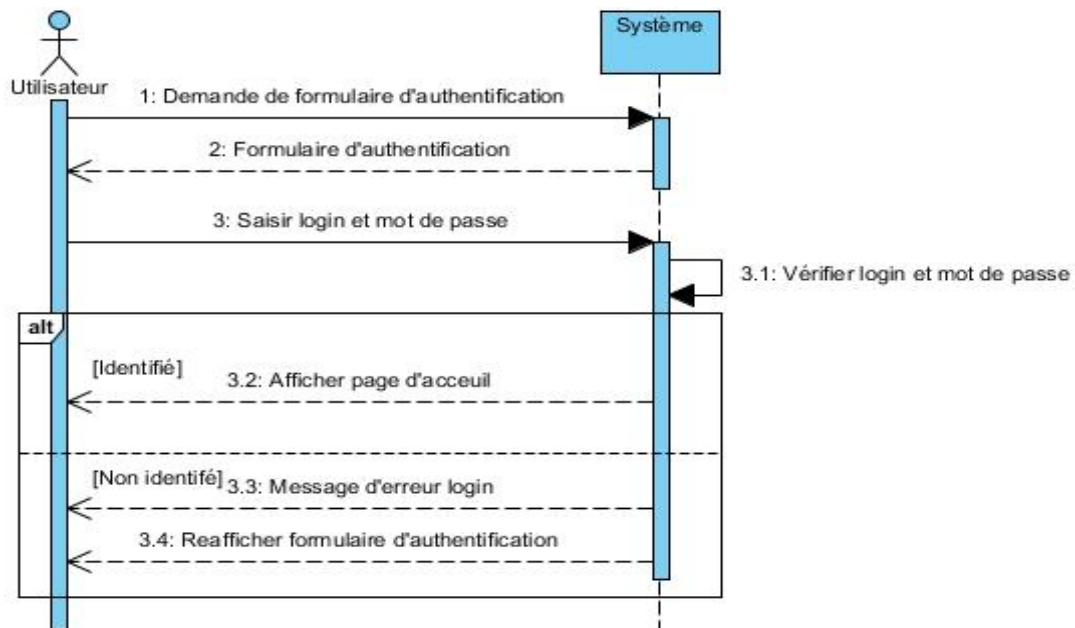


Figure 11. Diagramme de séquence système : S'authentifier

➤ **Cas d'utilisation : Créer comptes des professeurs**

La figure 12 représente le diagramme de séquence système pour le cas d'utilisation « Créer comptes des professeurs ».

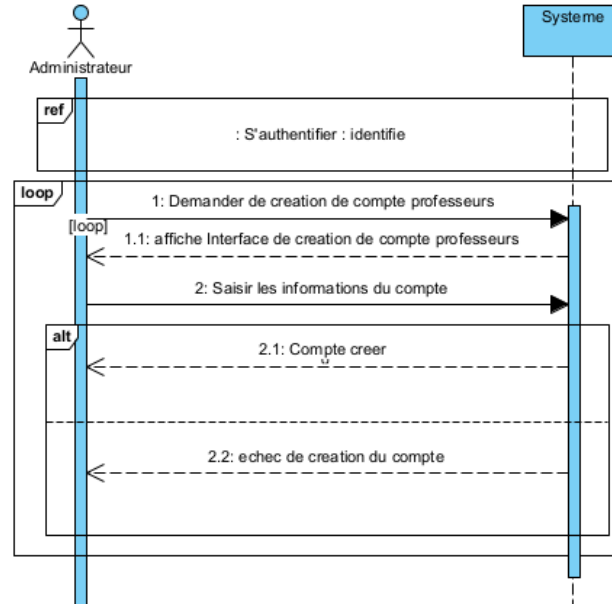


Figure 12. Diagramme de séquence de système : Créer compte Professeurs

➤ **Cas d'utilisation : Publier les annonces et informations**

La figure 13 représente le diagramme de séquence système pour le cas d'utilisation « Publier les annonces et informations ».

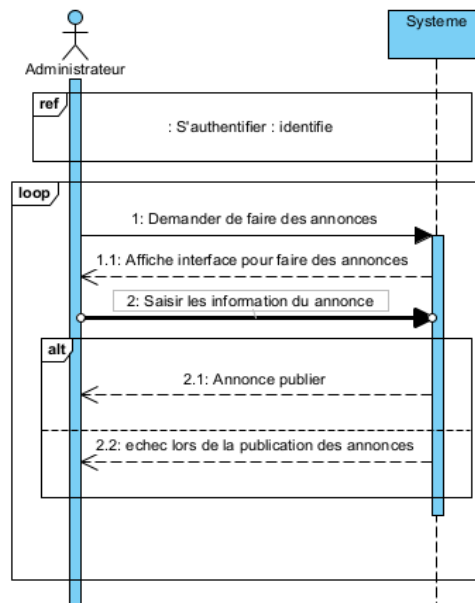


Figure 13. Diagramme de séquence de système : Publier annonces et informations

➤ **Cas d'utilisation : Création des cours**

La figure 14 représente le diagramme de séquence système pour le cas d'utilisation « Création des cours ».

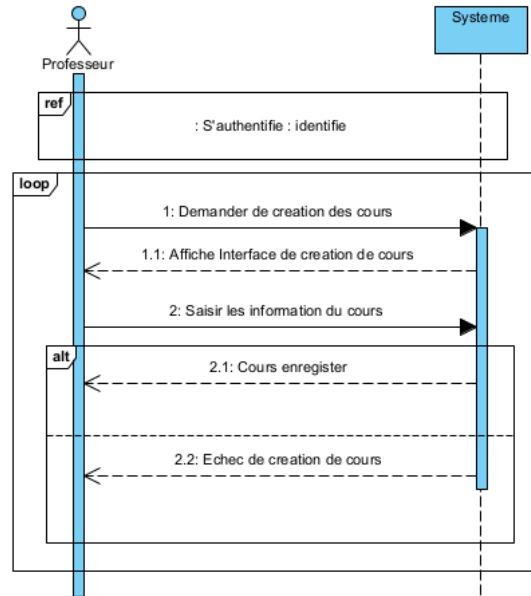


Figure 14. Diagramme de séquence de système : Création des cours

➤ **Cas d'utilisation : Création des devoirs**

La figure 15 représente le diagramme de séquence système pour le cas d'utilisation « Création des devoirs ».

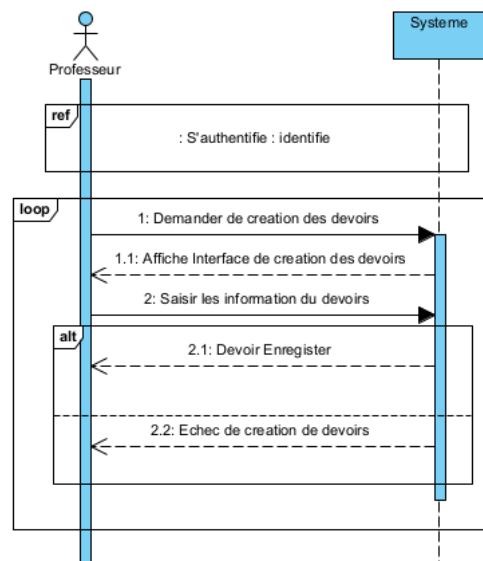


Figure 15. Diagramme de séquence de système : Création des devoirs

➤ Cas d'utilisation : Consulter les annonces et informations

La figure 16 représente le diagramme de séquence système pour le cas d'utilisation « Consulter les annonces et informations ».

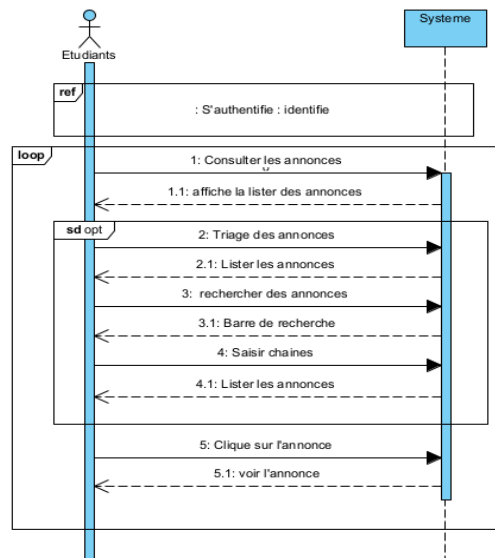


Figure 16. Diagramme de séquence de système : Consultation des annonces et informations

➤ Cas d'utilisation : Consultation des cours

La figure 17 représente le diagramme de séquence système pour le cas d'utilisation « Consultation des cours ».

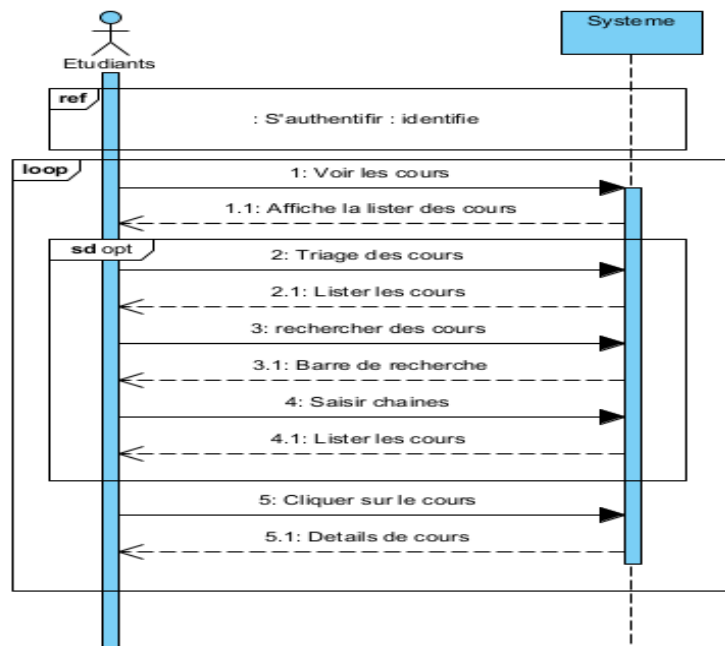


Figure 17. Diagramme de séquence de système : Consultation des cours

➤ **Cas d'utilisation : Consultation des devoirs**

La figure 18 représente le diagramme de séquence système pour le cas d'utilisation « Consultation des devoirs ».

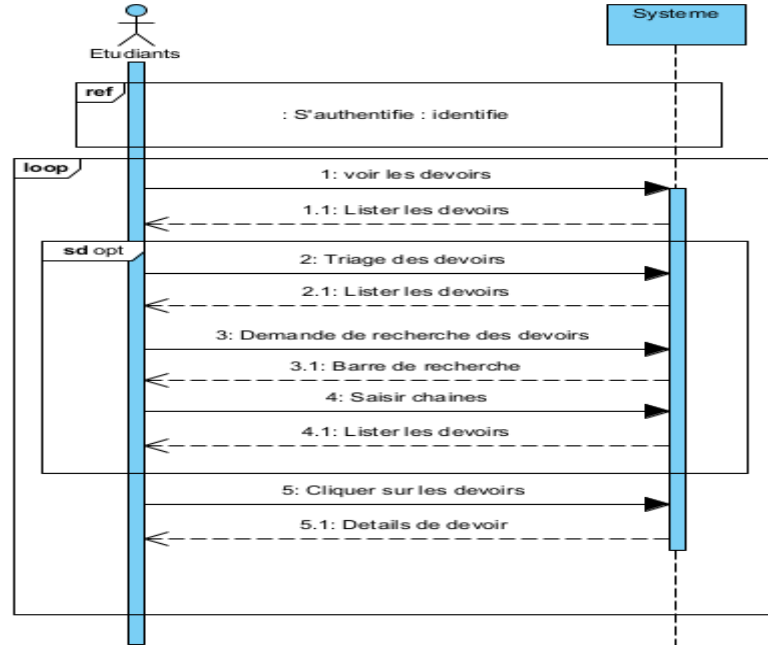


Figure 18. Diagramme de séquence de système : Consultation des devoirs

➤ **Cas d'utilisation : Soumettre les devoirs**

La figure 19 représente le diagramme de séquence système pour le cas d'utilisation « Soumettre les devoirs ».

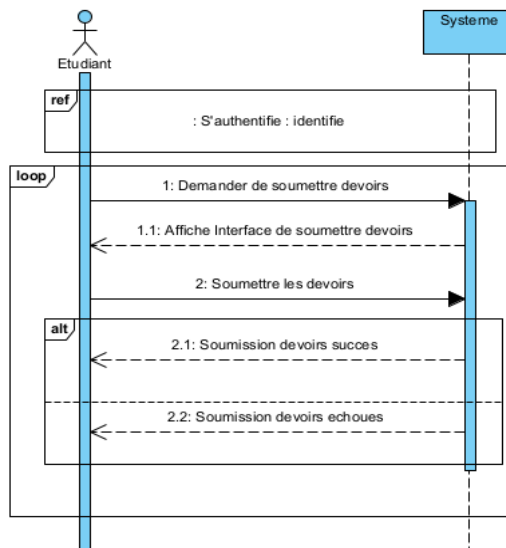


Figure 19. Diagramme de séquence de système : Soumettre des devoirs

➤ Cas d'utilisation : Gérer profile

La figure 20 représente le diagramme de séquence système pour le cas d'utilisation « Gérer profile ».

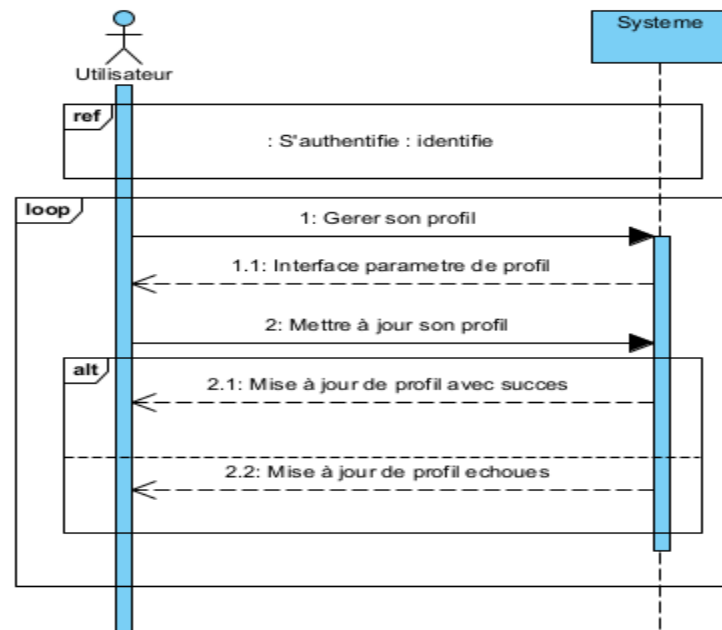


Figure 20. Diagramme de séquence de système : Gérer son profile

➤ Cas d'utilisation : Notifier les utilisateurs

La figure 21 représente le diagramme de séquence système pour le cas d'utilisation « Notifier les utilisateurs ».

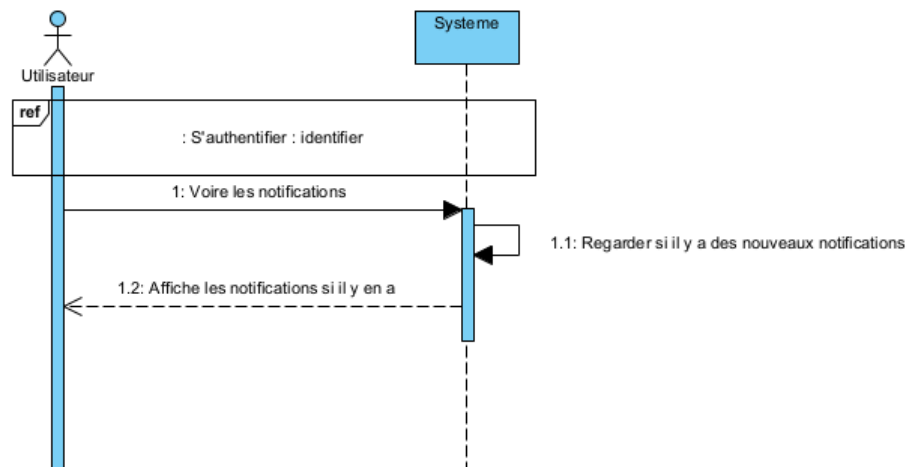


Figure 21. Diagramme de séquence de système : Notifier les utilisateurs

5.5. Spécification des besoins techniques

Les besoins techniques décrivent toutes les contraintes techniques, ergonomiques et esthétiques auxquelles est soumis le système pour sa réalisation et pour son bon fonctionnement. Et ce qui concerne notre application, nous avons dégagé les besoins suivants :

- **Sécurité de l'authentification** : mise en place d'un mécanisme d'authentification robuste pour assurer la sécurité des utilisateurs et leurs données.
- **Adaptabilité aux différents appareils** : la conception d'une interface utilisateur réactive et adaptive pour garantir une expérience utilisateur optimale sur divers appareils
- **Ergonomie et utilisabilité** : l'intégration de principe ergonomiques et d'une conception centrée sur l'utilisateur pour rendre l'application intuitive et conviviale.
- **Gestion des rôles et autorisations** : mise en place d'un système de gestion de rôles et autorisation pour définir les droit d'accès en fonction des utilisateurs.
- **Performances et temps de chargement** : l'optimisation des performances de l'application, y compris la réduction des temps de chargement, pour assurer une expérience fluide même avec des connexions internet moins rapides.
- **Interactivité en temps réel** : l'incorporation de fonctionnalités en temps réel, notamment pour les notifications, les mises à jours des cours ou des annonces.
- **Compatibilité avec les navigateurs** : la garantie de la compatibilité de l'application avec une gamme de navigateur populaires, tels que chrome, Firefox, safari et Edge.
- **Design esthétique et cohérent** : la création d'un design esthétique et cohérent, aligné avec l'identité visuelle de la faculté de science, pour offrir une expérience visuelle agréable aux utilisateurs.

5.6. Modélisation du domaine

La modélisation des besoins par des cas d'utilisation s'apparente à une analyse fonctionnelle classique. L'élaboration du modèle des classes du domaine permet d'opérer vers une véritable modélisation objet. La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classes appelée modèle du domaine. Ce modèle doit définir les classes qui modélisent les entités ou concepts présents dans le domaine (métier) de l'application. Il s'agit donc de produire un modèle des objets du monde réel dans un domaine donné.

La figure 22 représente le modèle de domaine du système

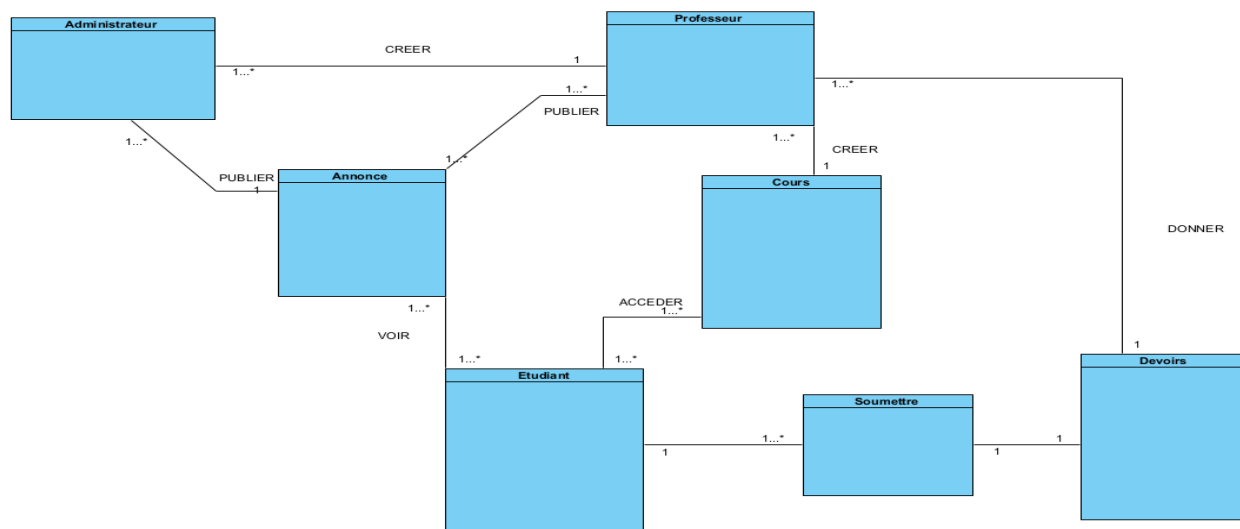


Figure 22. Modèle de domaine

Chapitre 6. Conception détaillée

Nous allons maintenant établir la conception détaillée du système dont nous avons mis en place.

6.1. Architecture du système

L'architecture précédente s'appuie sur MVC (Modèle-Vue-Contrôleur) [3] qui est une architecture destinée à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants en les regroupant par couches. Comme son nom l'indique, MVC regroupe les fonctions nécessaires en trois catégories : Modèle, Vue et Contrôleur.

La figure 23 représente la modèle MVC

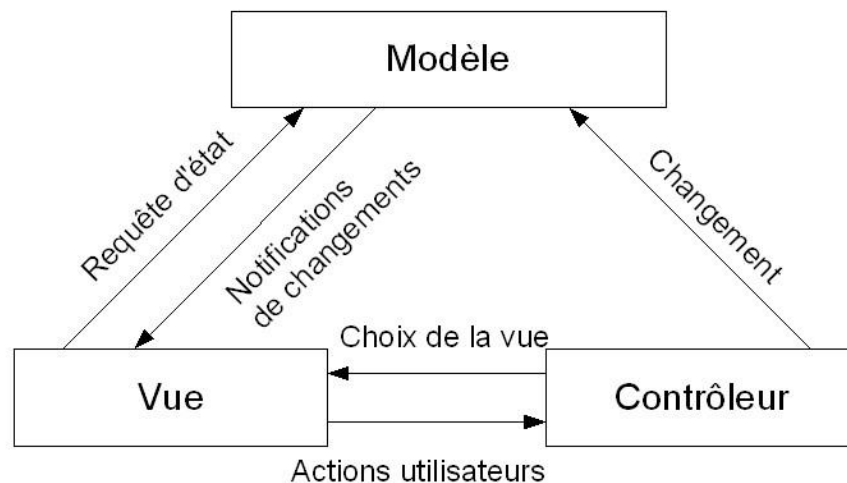


Figure 23. Architecture du système

- **Le Modèle (modèle de données):** cette partie gère les données de l'application. Son rôle est d'aller récupérer les informations « brutes » dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. On y trouve donc les requêtes SQL ou NoSQL.
- **La Vue (présentation, interface utilisateur):** cette partie se concentre sur l'affichage. Elle ne fait presque aucun traitement et se contente juste de récupérer des variables pour savoir ce qui doit être affiché. On y trouve essentiellement de code HTML.
- **Le Contrôleur** (logique de contrôle, gestion des événements, synchronisation): cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le Modèle et la Vue: le Contrôleur va demander au Modèle les données pour les analyser, prendre des décisions et renvoyer le texte à afficher à la Vue.

L'architecture MVC a été utilisée ici à cause de la clarté de l'architecture qu'elle impose. En effet, la modification des traitements ne change en rien la Vue. Par exemple on peut passer d'une base de données de type SQL à XML en changeant simplement les traitements d'interaction avec la base, et les vues ne s'en trouvent pas affectées.

La section est consacrée à l'implémentation des différentes fonctionnalités requises à l'authentification.

6.2. Diagramme de séquence de conception pour chaque cas d'utilisation

Les diagrammes de séquence de conception, dont des diagrammes qui décrivent des appels de méthodes et de la coopération entre instances et méthodes. Elles permettent de donner une vue en largeur du déroulement d'une opération : les objets impliqués, les méthodes des classes concernées, la source des paramètres. Elles sont utiles pour comprendre la mécanique d'ensemble et la répartition des responsabilités avant de rentrer dans le détail. Les diagrammes de séquence de conception sont utiles aussi bien comme outil de « conception » que comme « documentation ».

Pour chaque cas d'utilisation, il y a donc un diagramme de séquence de conception correspondant.

➤ Cas d'utilisation : S'inscrire

La figure 24 représente le diagramme de séquence de conception pour le cas d'utilisation « S'inscrire ».

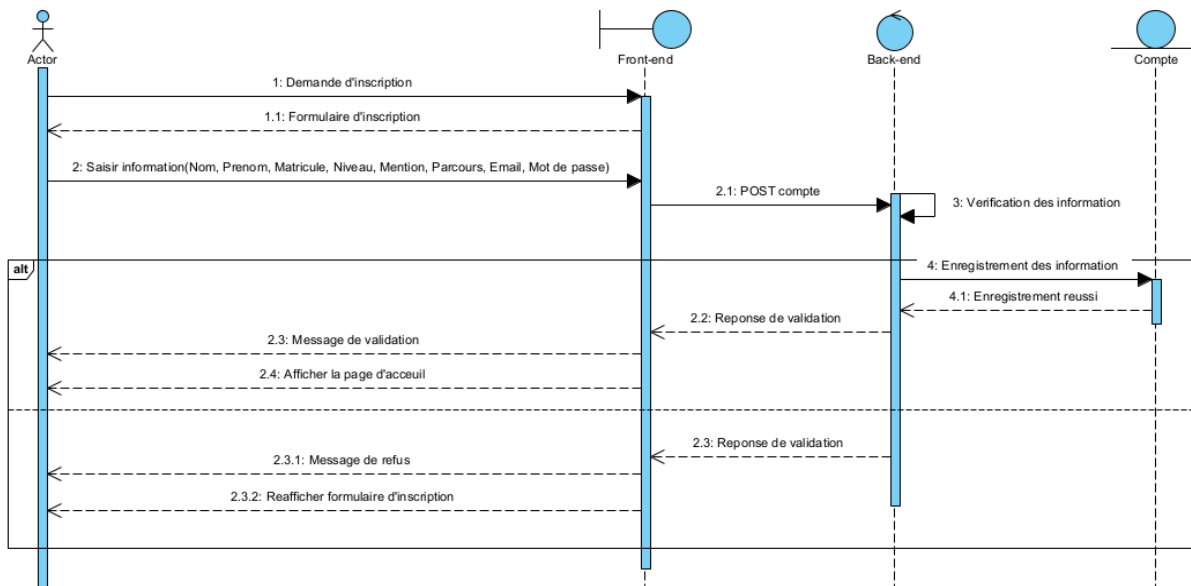


Figure 24. Diagramme de sequence de conception : S'inscrire

➤ Cas d'utilisation : S'authentifier

La figure 25 représente le diagramme de séquence de conception pour le cas d'utilisation « S'authentifier ».

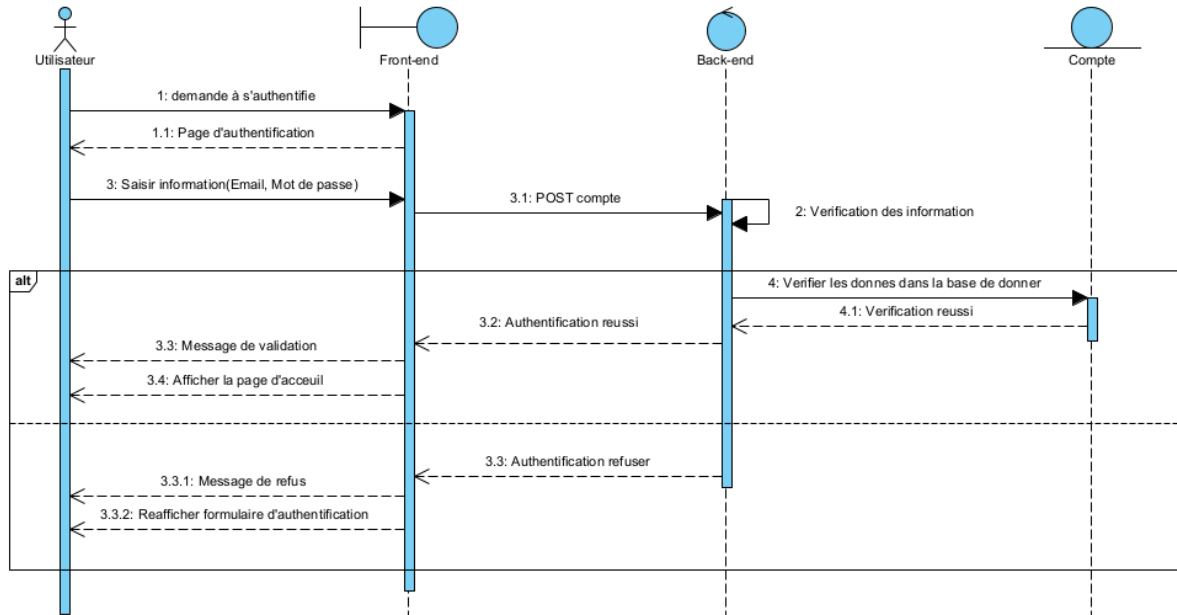


Figure 25. Diagramme de sequence de conception : S'authentifier

➤ Cas d'utilisation : Créer compte professeurs

La figure 26 représente le diagramme de séquence de conception pour le cas d'utilisation « Créer compte professeurs ».

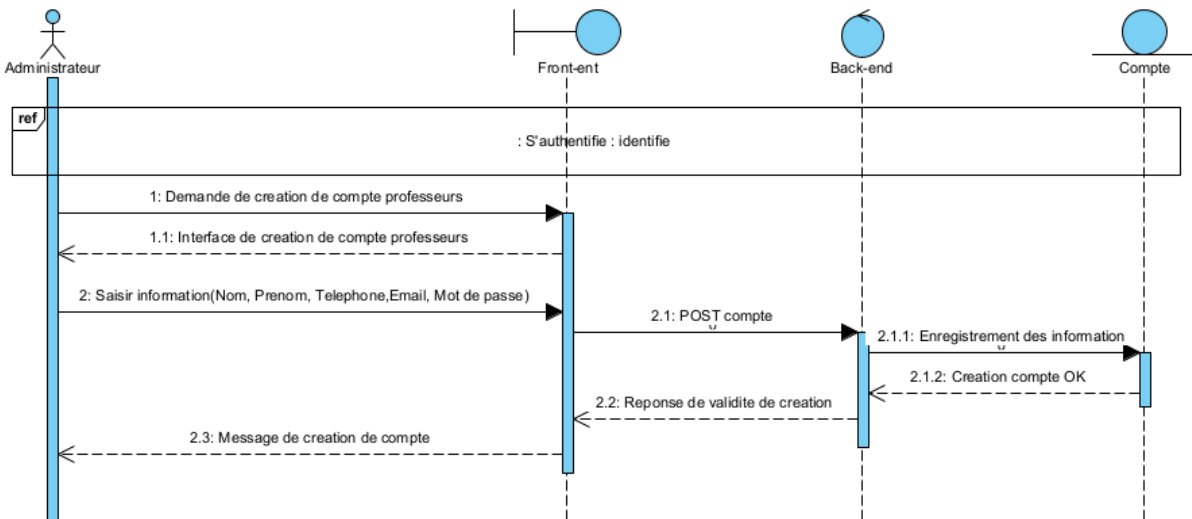


Figure 26. Diagramme de sequence de conception : Créer compte Professeurs

➤ Cas d'utilisation : Publier des annonces et informations

La figure 27 représente le diagramme de séquence de conception pour le cas d'utilisation « Publier des annonces et informations ».

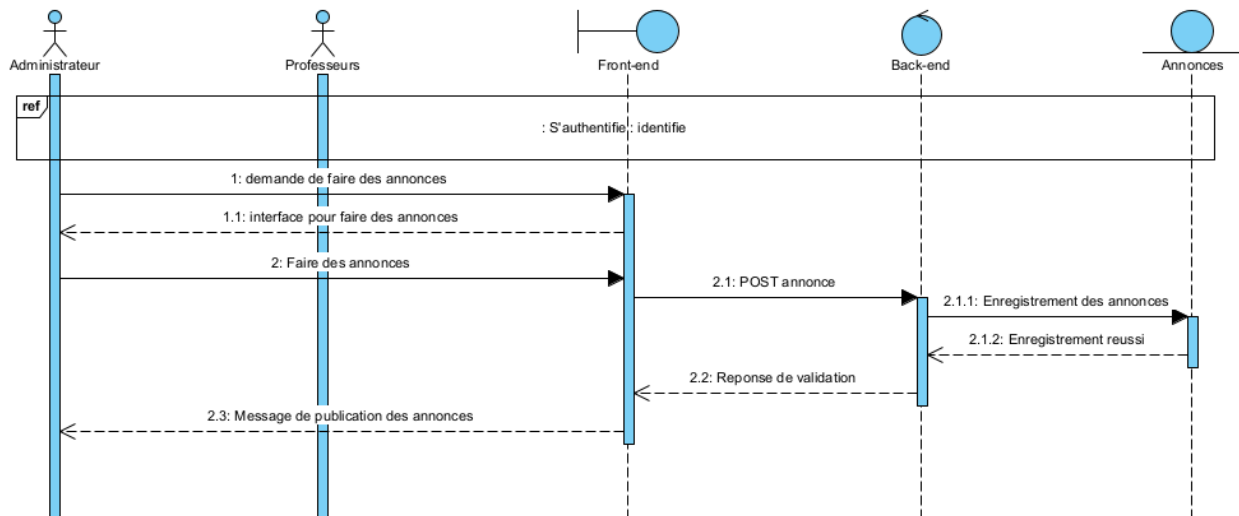


Figure 27. Diagramme de séquence de conception : Publier annonces et information

➤ Cas d'utilisation : Création des cours

La figure 28 représente le diagramme de séquence de conception pour le cas d'utilisation « Création des cours ».

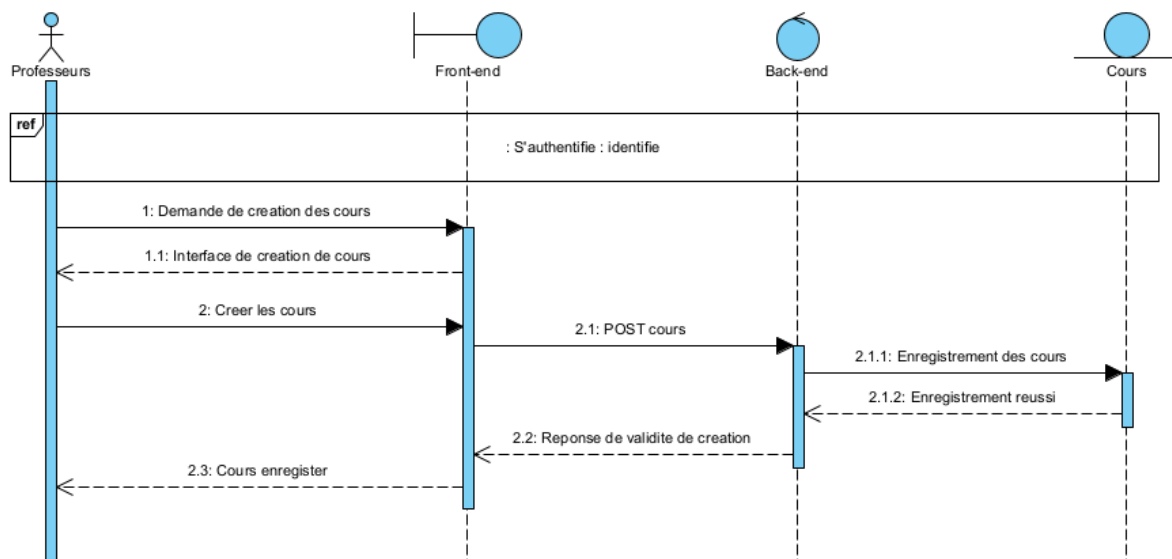


Figure 28. Diagramme de séquence de conception : Creation des cours

➤ Cas d'utilisation : Création des devoirs

La figure 29 représente le diagramme de séquence de conception pour le cas d'utilisation « Création des devoirs ».

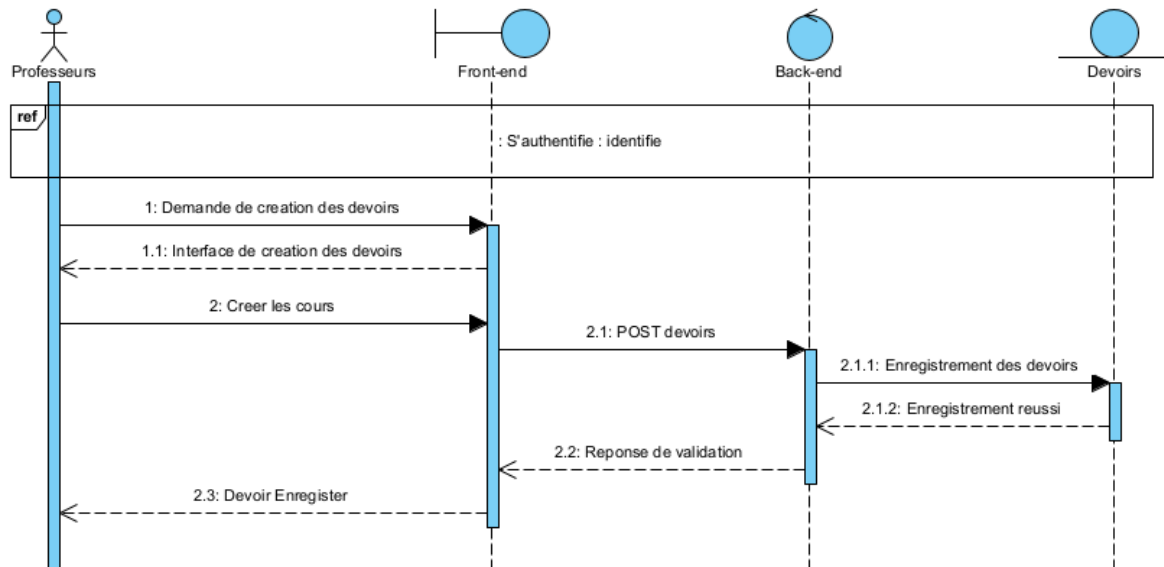


Figure 29. Diagramme de sequence de conception : Creation des devoirs

➤ Cas d'utilisation : Consultation des annonces et informations

La figure 30 représente le diagramme de séquence de conception pour le cas d'utilisation « Consultation des annonces et information ».

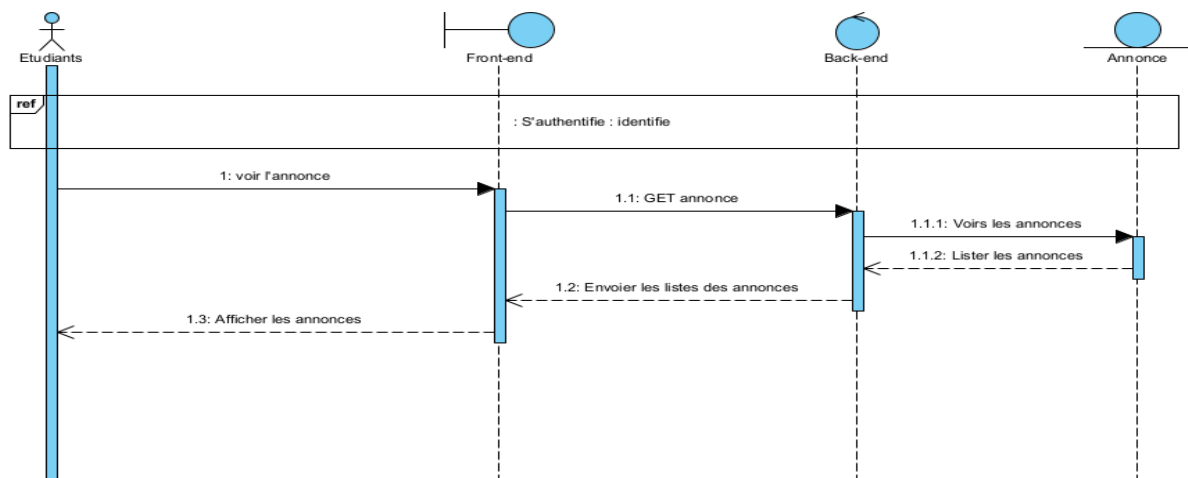


Figure 30. Diagramme de sequence de conception : Consultation des annonces et informations

➤ **Cas d'utilisation : Consultation des cours**

La figure 31 représente le diagramme de séquence de conception pour le cas d'utilisation « Consultation des cours ».

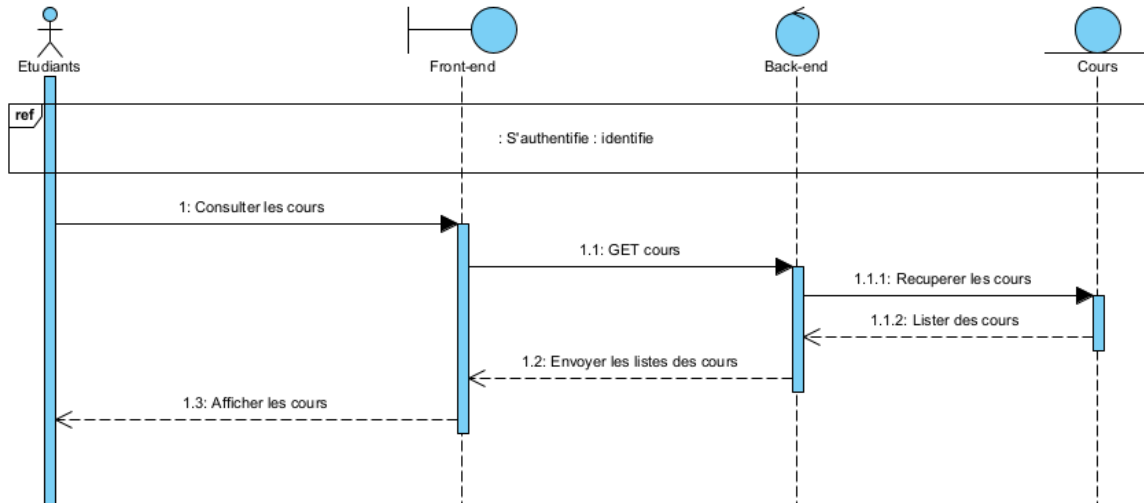


Figure 31. Diagramme de séquence de conception : Consultation des cours

➤ **Cas d'utilisation : Consultation des devoirs**

La figure 32 représente le diagramme de séquence de conception pour le cas d'utilisation « Consultation des devoirs ».

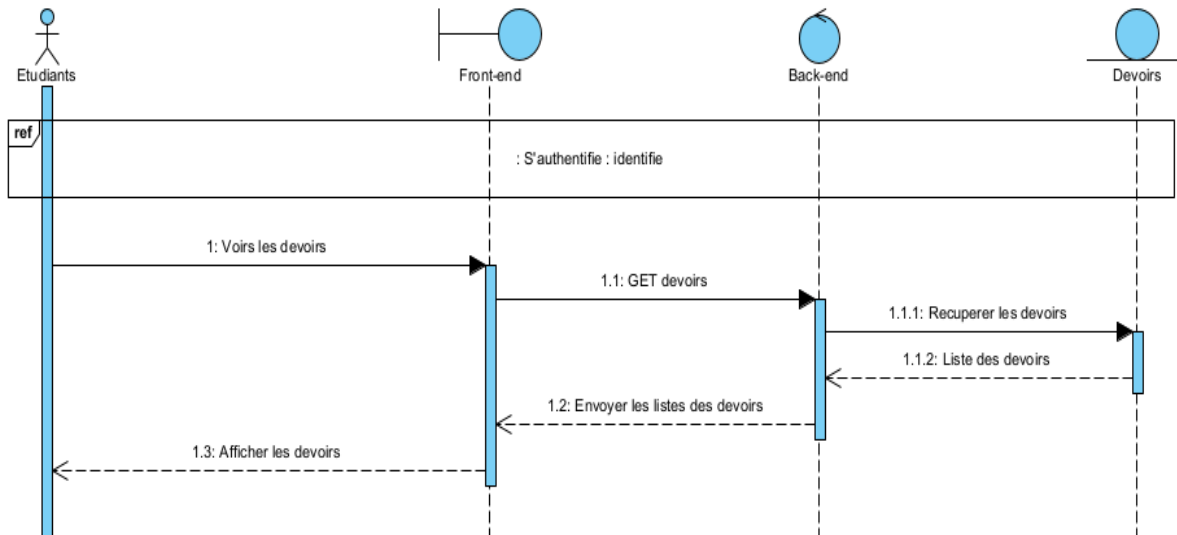


Figure 32. Diagramme de séquence de conception : Consultation des devoirs

➤ Cas d'utilisation : Soumettre des devoirs

La figure 33 représente le diagramme de séquence de conception pour le cas d'utilisation « Soumettre de devoirs ».

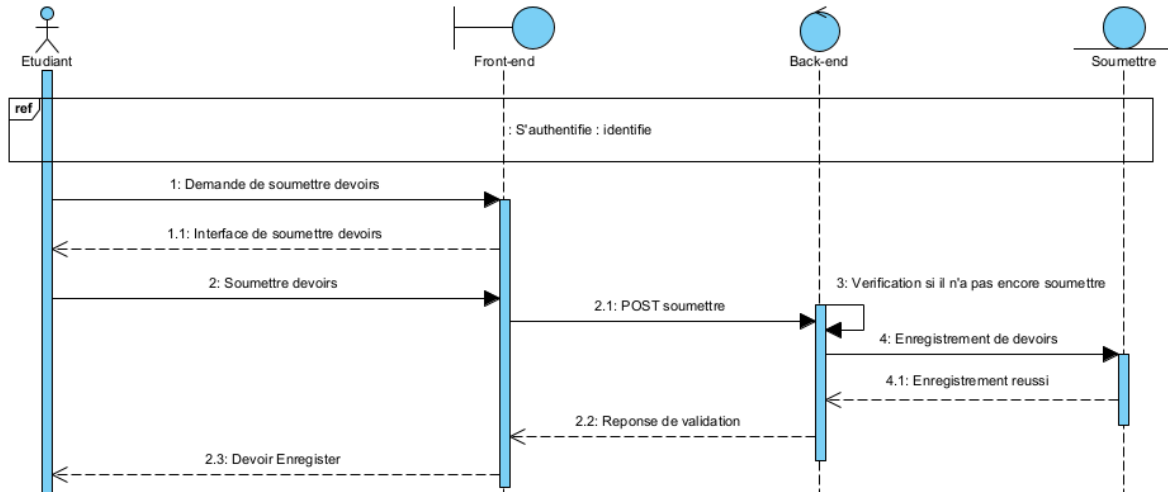


Figure 33. Diagramme de sequence de conception : Soumettre des devoirs

➤ Cas d'utilisation : Gérer son profile

La figure 34 représente le diagramme de séquence de conception pour le cas d'utilisation « Gérer son profile ».

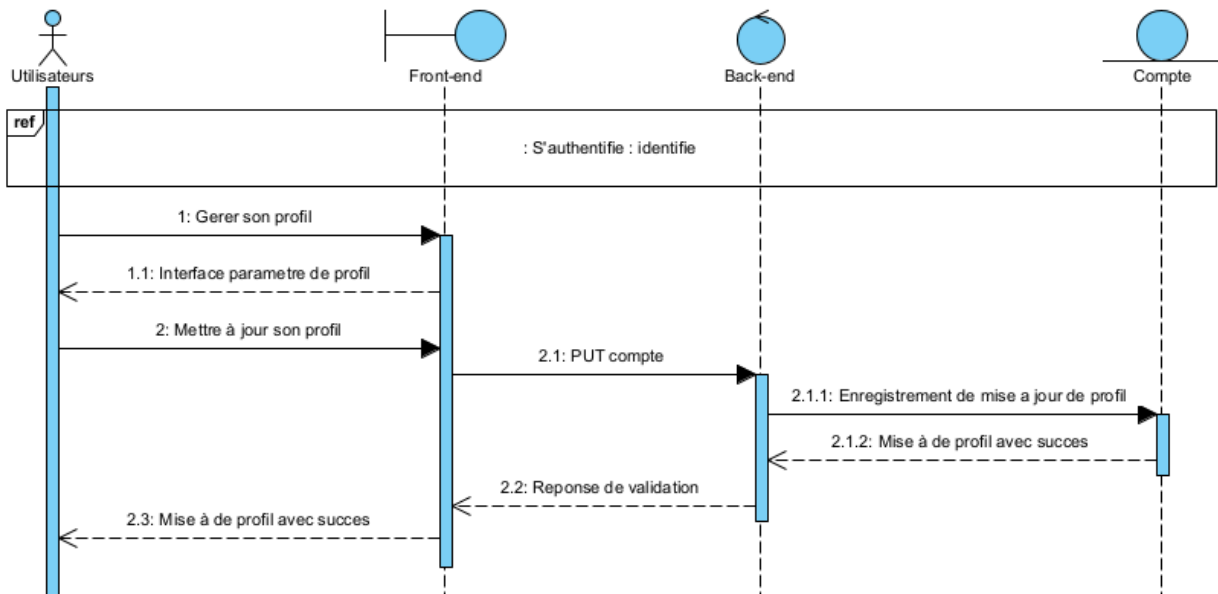


Figure 34. Diagramme de sequence de conception : Gerer son profile

➤ Cas d'utilisation : Notifier les utilisateurs

La figure 35 représente le diagramme de séquence de conception pour le cas d'utilisation « Notifier les utilisateurs ».

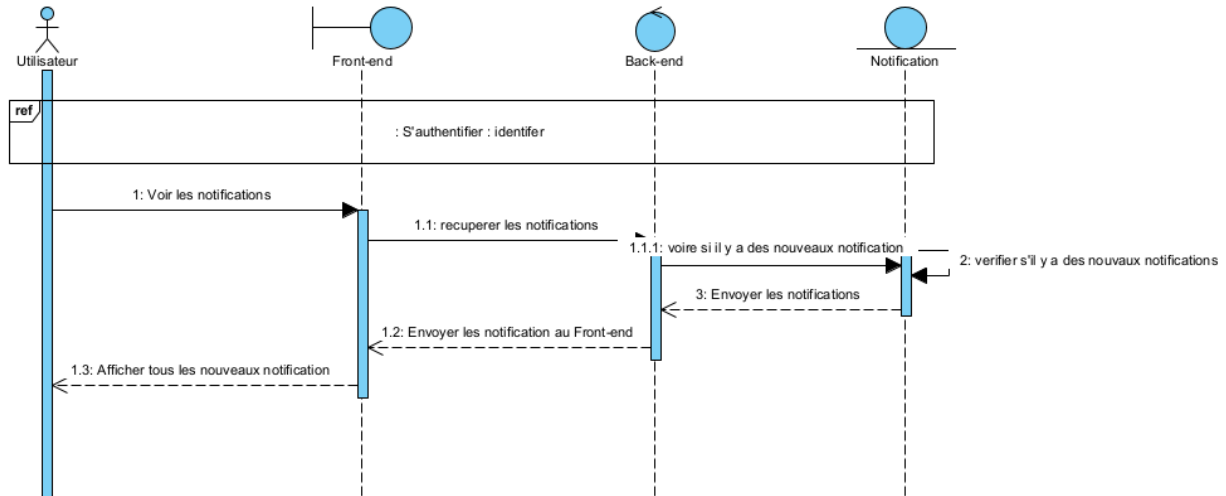


Figure 35. Diagramme de sequence de conception : Notifier les Utilisateurs

6.3. Diagramme de classe de conception pour chaque cas d'utilisation

Avant d'établir le diagramme de classe de conception global, analysons les diagrammes de classe pour chaque cas d'utilisation.

Diagramme de classe de conception pour le cas d'utilisation : S'inscrire

La figure 36 représente le diagramme de classe de conception pour le cas d'utilisation « S'inscrire ».

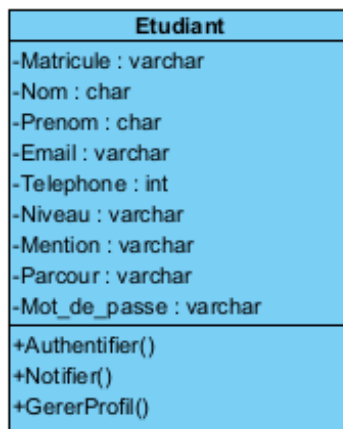


Figure 36. Diagramme de classe de conception : S'inscrire

Diagramme de classe de conception pour le cas d'utilisation : S'authentifier

La figure 37 représente le diagramme de classe de conception pour le cas d'utilisation « S'authentifier ».

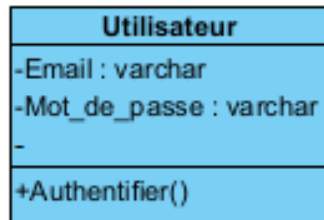


Figure 37. Diagramme de classe de conception : S'authentifier

Diagramme de classe de conception pour le cas d'utilisation : Créer compte des professeurs

La figure 38 représente le diagramme de classe de conception pour le cas d'utilisation « créer compte des professeurs ».

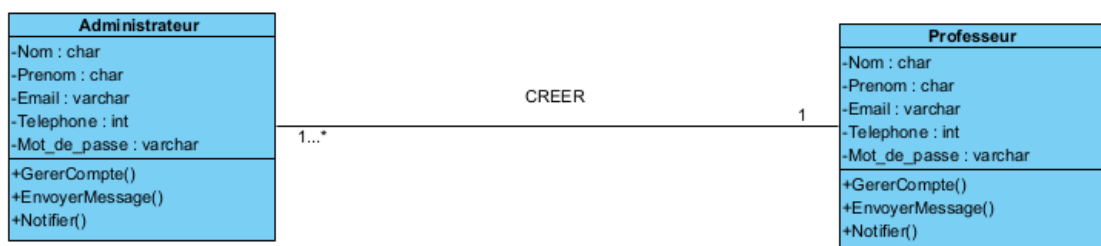


Figure 38. Diagramme de classe de conception : Créer compte des professeurs

Diagramme de classe de conception pour le cas d'utilisation : Publier des annonces et informations

La figure 39 représente le diagramme de classe de conception pour le cas d'utilisation « Publier des annonces et informations ».

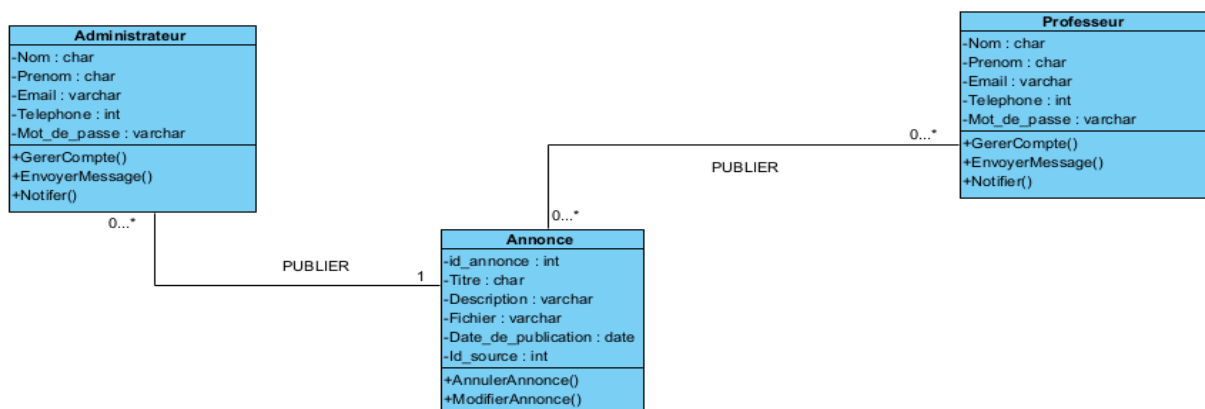


Figure 39. Diagramme de classe de conception : Publier des annonces et informations

Diagramme de classe de conception pour le cas d'utilisation : Création des cours

La figure 40 représente le diagramme de classe de conception pour le cas d'utilisation « Création des cours ».

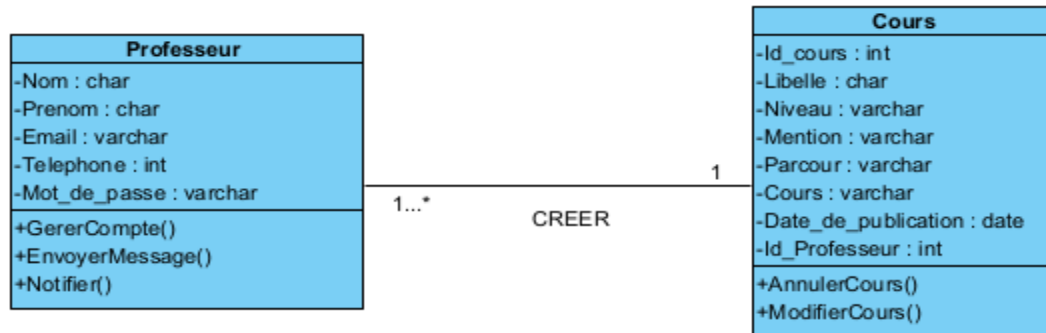


Figure 40. Diagramme de classe de conception : Creation des cours

Diagramme de classe de conception pour le cas d'utilisation : Création des devoirs

La figure 41 représente le diagramme de classe de conception pour le cas d'utilisation « Création des devoirs ».

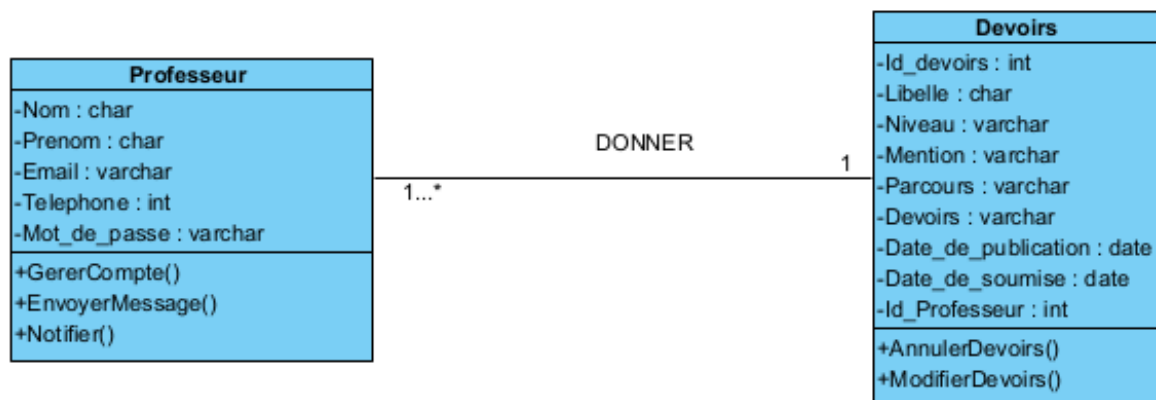


Figure 41. Diagramme de classe de conception : Creation des devoirs

Diagramme de classe de conception pour le cas d'utilisation : Consultation des annonces et informations

La figure 42 représente le diagramme de classe de conception pour le cas d'utilisation « Consultation des annonces et informations ».

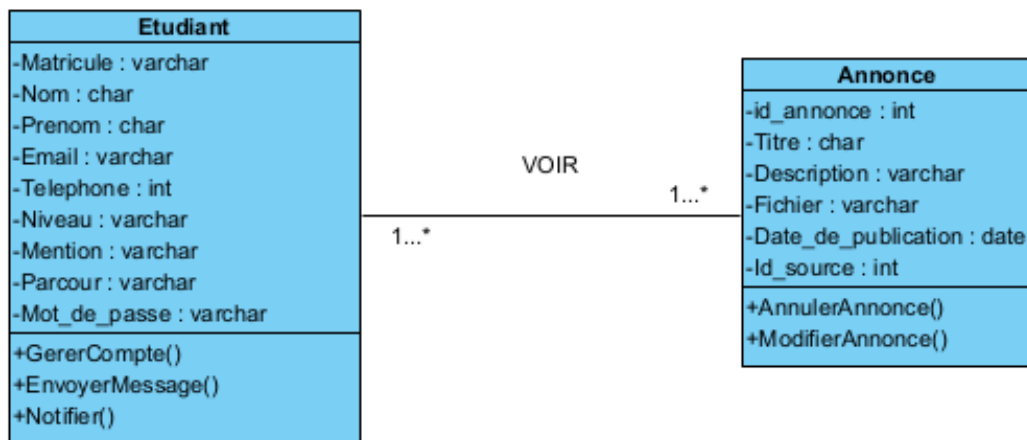


Figure 42. Diagramme de classe de conception : Consultation des annonces et informations

Diagramme de classe de conception pour le cas d'utilisation : Consultation des cours

La figure 43 représente le diagramme de classe de conception pour le cas d'utilisation « Consultation des cours ».

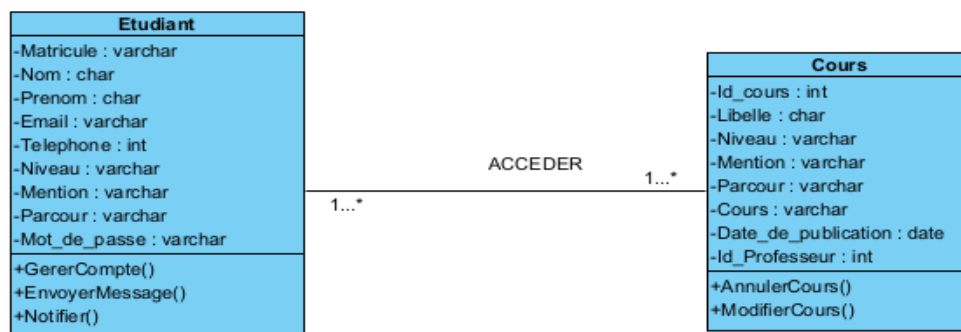


Figure 43. Diagramme de classe de conception : consultation des cours

Diagramme de classe de conception pour le cas d'utilisation : Soumettre des devoirs

La figure 44 représente le diagramme de classe de conception pour le cas d'utilisation « Soumettre des devoirs ».

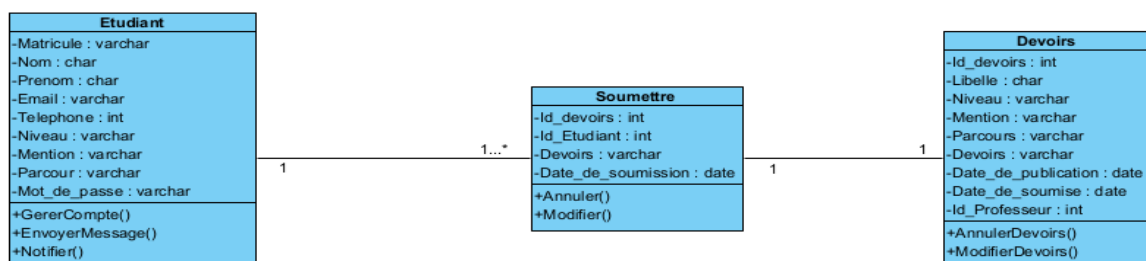


Figure 44. Diagramme de classe de conception : Soumettre des devoirs

6.4. Diagramme de classe de conception global

La description du diagramme des classes est fondée sur : le concept d'objet, le concept de classe comprenant les attributs et les opérations, les différents types d'association entre classes.

➤ Objet

Un objet est un concept, une abstraction ou une chose qui a un sens dans le contexte du système à modéliser. Chaque objet a une identité et peut être distingué des autres sans considérer à priori les valeurs de ses propriétés. Un objet est une instance d'une classe.

➤ Attributs

Un attribut est une donnée élémentaire d'une classe servant à caractériser les classes et les relations.

➤ Classe

Une classe décrit un groupe d'objets ayant les mêmes attributs, une même opération, et un même domaine de définition. La classe représente l'abstraction de ses objets.

➤ Opération

Une opération est une fonction applicable aux objets d'une classe. Elle permet de décrire le comportement d'un objet.

➤ Association

Une association entre classes permet de relier une classe à plusieurs autres classes. Chaque association peut être identifiée par son nom.

➤ Multiplicité

La multiplicité indique un domaine de valeurs pour exprimer le nombre minimum et maximum d'objets d'une classe qui peuvent être reliés à des objets d'une autre classe.

La figure 45 représente le diagramme de classe de conception global.

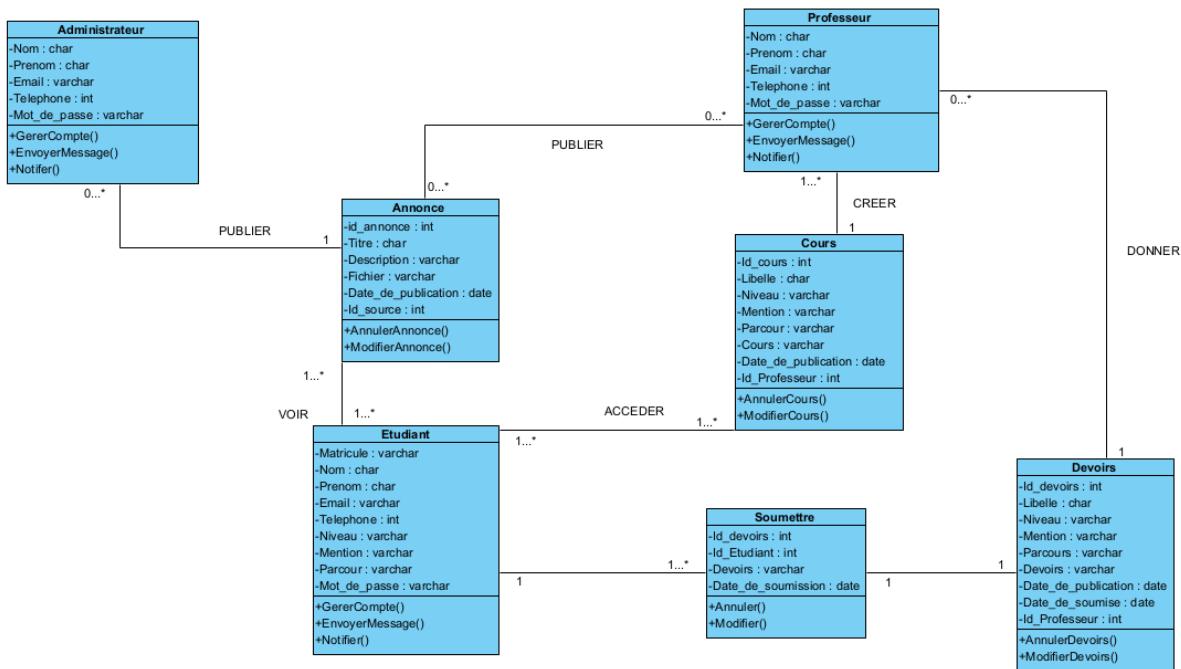


Figure 45. Diagramme de classe de conception global

6.5. Diagramme de paquets

Les diagrammes de package (ou diagramme de paquets) sont des diagrammes structurels utilisés pour représenter l'organisation et la disposition de divers éléments modélisés sous forme de paquets.

Dépendance d'importation : indique que la fonctionnalité a été importée d'un package à un autre. Notre diagramme de package qui est représenté par la figure 46 est inspiré de l'architecture du système pour favoriser la réutilisation des packages.

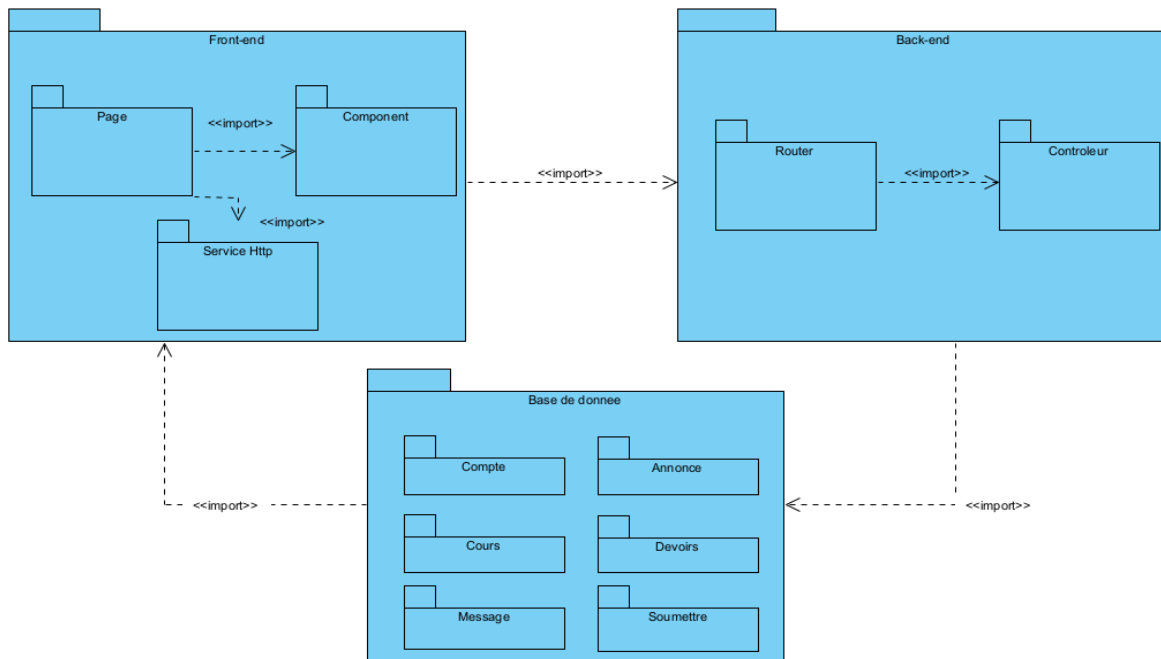


Figure 46. Diagramme de paquetage

6.6. Diagramme de déploiement

Le diagramme de déploiement montre la configuration des nœuds d'exécution et des composants qu'y résident. Il schématise les relations physiques et les nœuds entre les composants logiciels et matériels d'un système. La figure 47 représente le diagramme de déploiement de la phase conception.

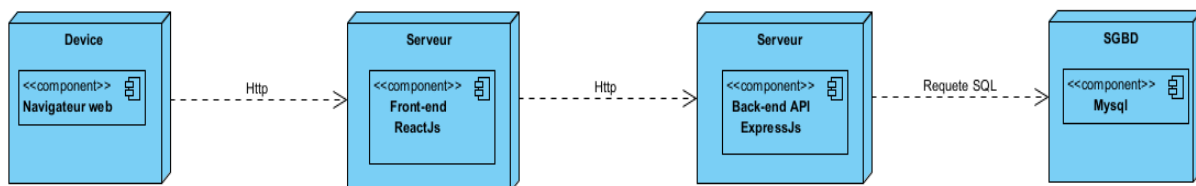


Figure 47. Diagramme de deploiment

PARTIE III : REALISATION

Chapitre 7. Mise en place de l'environnement et de développement

Dans ce chapitre nous allons aborder l'installation des outils nécessaires au projet ainsi que l'architecture de l'application.

7.1. Installation et configuration des outils

7.1.1. Visual Paradigm

Visual Paradigm est un logiciel de création de diagrammes dans le cadre d'une programmation. Tout en un, il possède plusieurs options permettant une large possibilité de modélisation en UML. On télécharge l'installateur de la Community Edition [2] pour un système Windows 10 64bit. La figure 48 montre le déroulement de son installation.

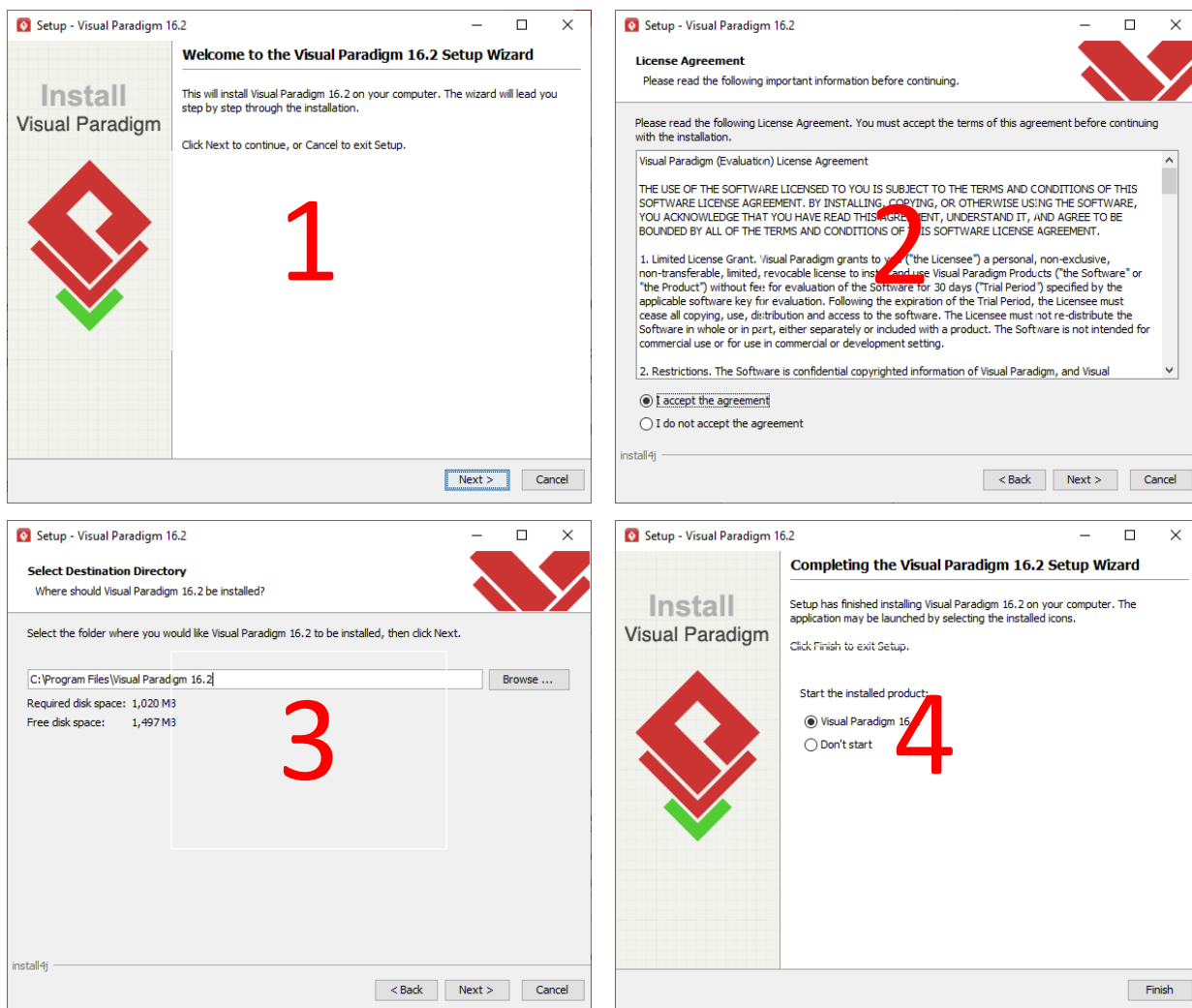


Figure 48. Installation de Visual Paradigm

7.1.2. Laragon

Laragon est un environnement de développement et un ensemble de serveurs web principalement conçus pour les systèmes d'exploitation Windows. Il offre une solution légère et facile à utiliser pour la mise en place d'un environnement de développement web local.

Une fois l'installateur de Laragon pour Windows téléchargé, on lance et on obtient une assistance d'installateur automatique. On accepte le contrat de licence utilisateur et suit les instructions. La figure 49 représente son installation.

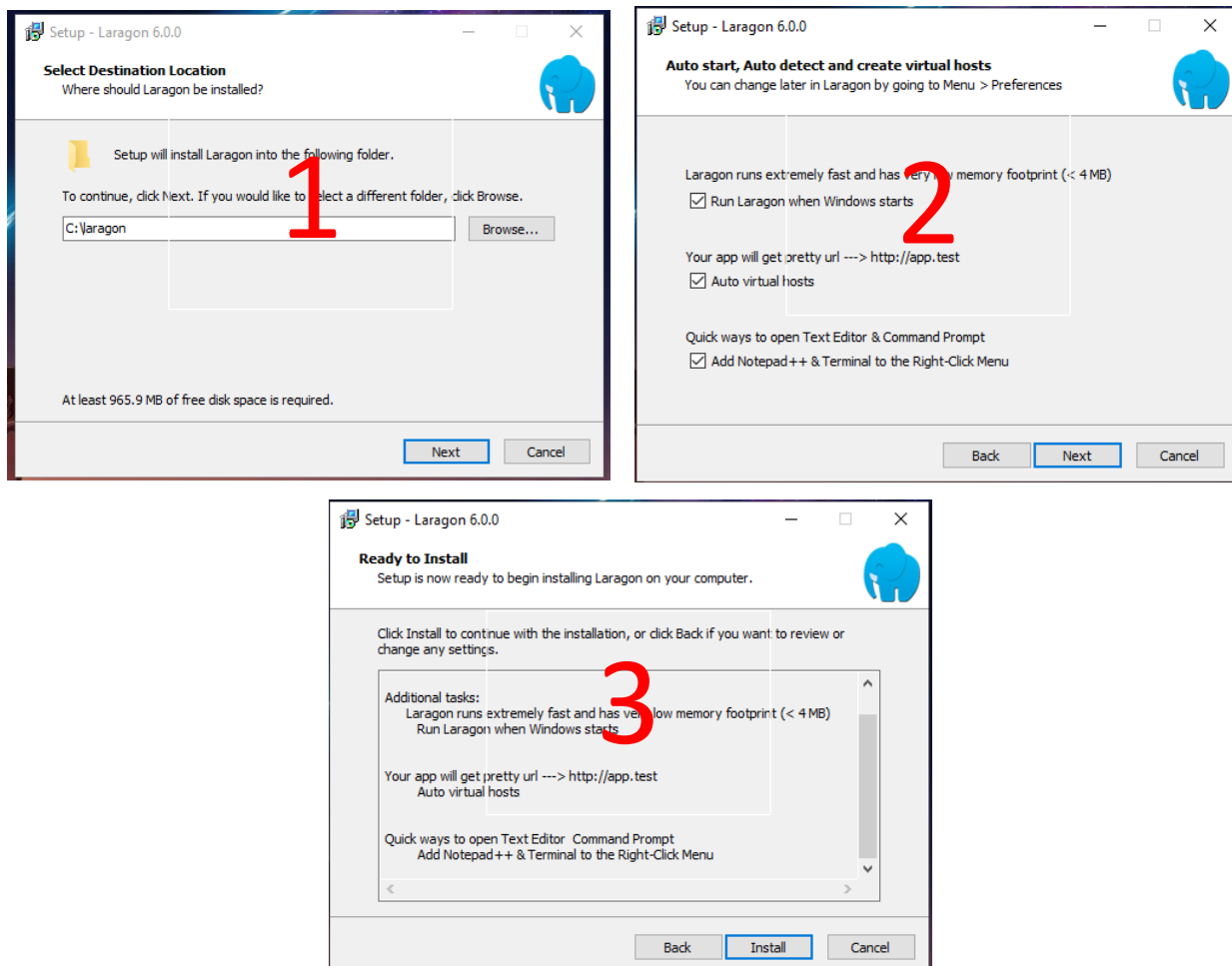


Figure 49. Installation de Laragon

Après le succès de l'installation, on lance le logiciel et on obtient comme celui de là La base de données MySQL fonctionne en connectant une chaîne de connexion. La figure 50 représente le lancement de Laragon.

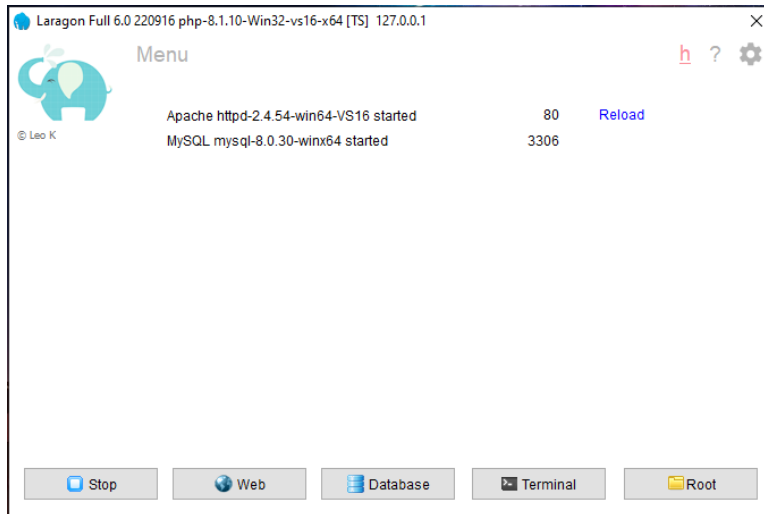
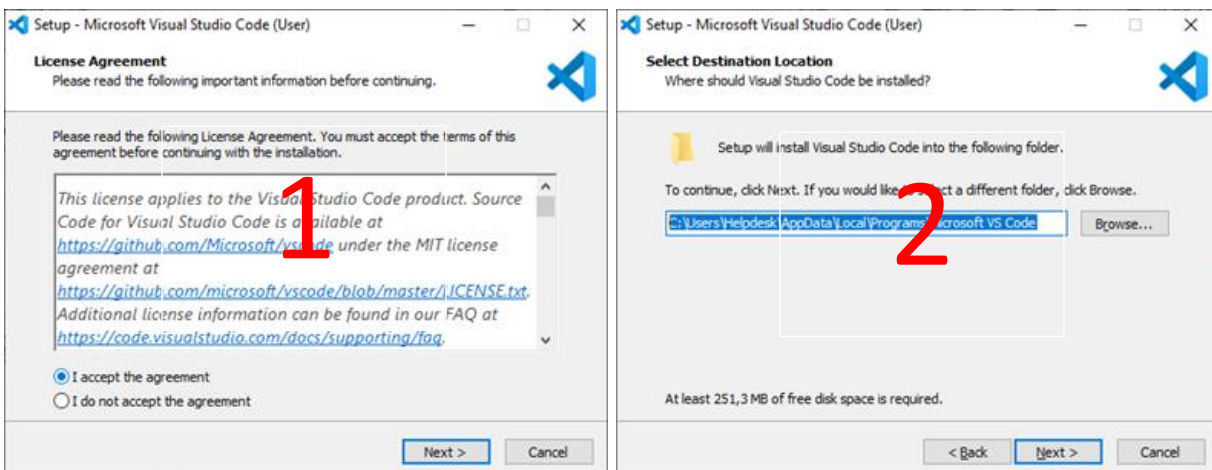


Figure 50. Lancement de Laragon

7.1.3. Visual Studio Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré.

Pour installer Visual Studio Code sous Windows, téléchargez l'exécutable à l'adresse On exécute le fichier d'installation .exe puis lis et accepte le contrat de licence et suivre les installations en gardant les paramètres par défaut comme l'indique la figure 51.



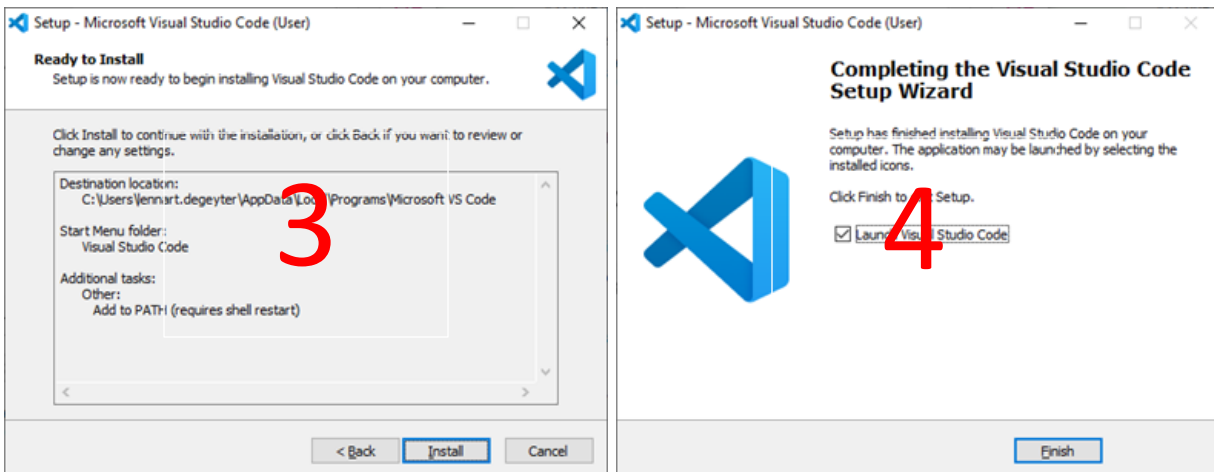


Figure 51. Installation de Visual Studio Code

Le logiciel est installé et prêt à l'emploi comme la figure 52 montre son premier démarrage.

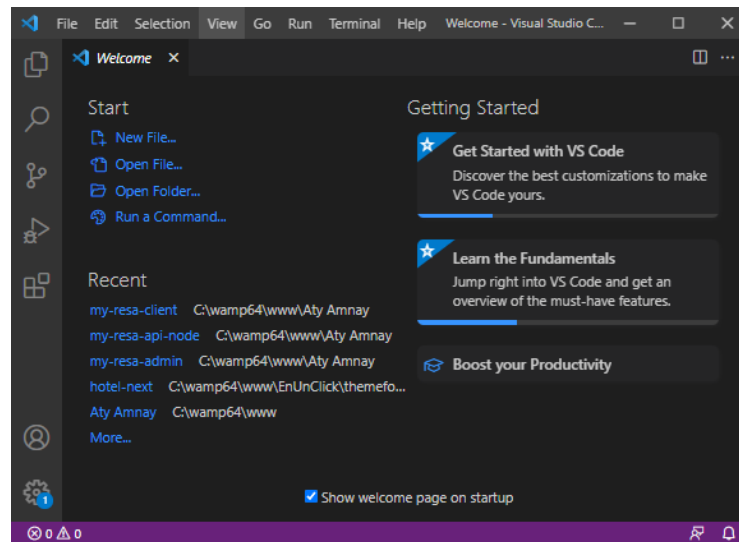


Figure 52. Lancement de Visual Studio Code

7.1.4. NodeJs

Node.js est une plateforme logicielle libre en JavaScript, orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge. Elle utilise la machine virtuelle V8, la librairie libuv pour sa boucle d'évènements, et implémente sous licence MIT les spécifications CommonJS.

npm est le gestionnaire de packages de la plate-forme JavaScript Node. Il met en place des modules pour que le nœud puisse les trouver et gère intelligemment les conflits de dépendances. Il est extrêmement configurable pour prendre en charge une grande variété de cas d'utilisation. Le

plus souvent, il est utilisé pour publier, découvrir, installer et développer des programmes de nœuds.

Afin de télécharger **NodeJS** et **npm** sur windows, on a besoin d'accéder à l'adresse de son téléchargement. L'installation de NodeJS sur Windows est très simple. Acceptez toutes des options par défaut et appuyez sur "Next...Next" jusqu'à la fin comme la montre dans la figure 53.

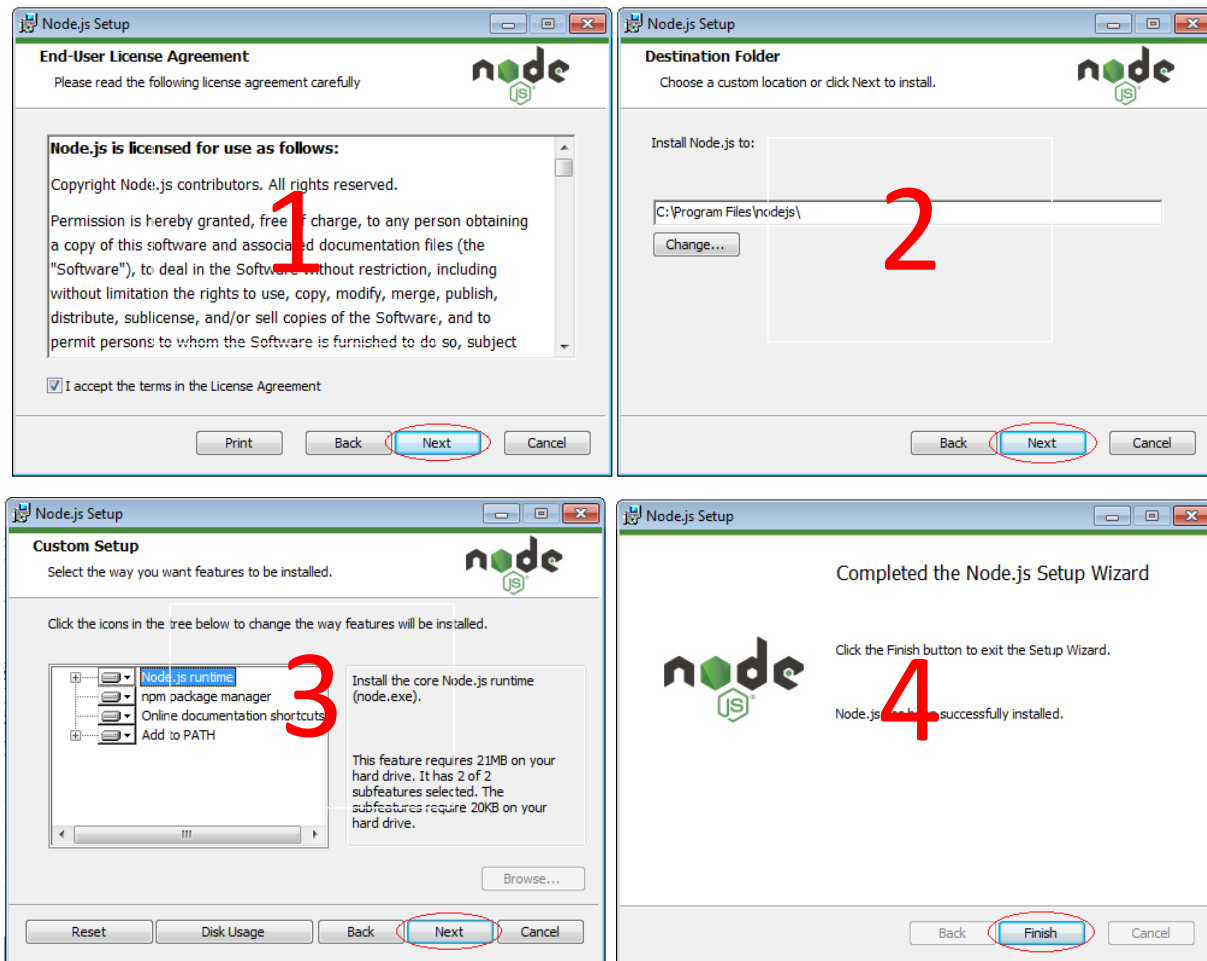


Figure 53. Installation de NodeJs

Après que l'installation de NodeJs s'est terminée avec succès. On peut vérifier sa version et celle du npm sur l'invite de commande Windows.

Yarn est un nouveau gestionnaire de packages qui remplace le flux de travail existant pour le client npm ou d'autres gestionnaires de packages tout en restant compatible avec le registre npm. Il dispose du même ensemble de fonctionnalités que les flux de travail existants tout en fonctionnant plus rapidement, de manière plus sécurisée et plus fiable. Il s'installe via npm par la commande : `npm install --global yarn`.

7.2. Architecture de l'application

Dans ce présent projet nous avons adopté l'architecture à trois tiers qui s'appuie sur trois niveaux dont :

- La vue (ou la présentation)
- La logique applicative (ou les traitements)
- La base de données

La figure 54 est la représentation exacte de l'architecture de l'application.

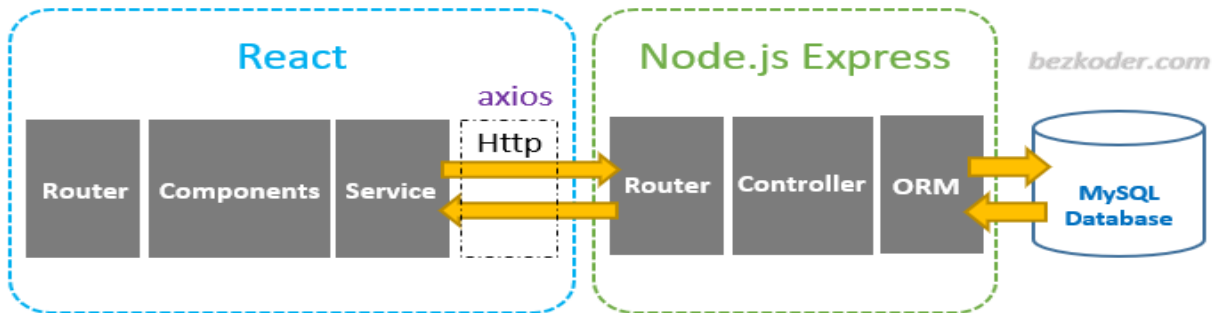


Figure 54. Architecture de l'application

Chapitre 8. Développement de l'application

Dans ce dernier chapitre nous allons présenter quelques morceaux de codes nécessaires pour la création de la base de données et pour le codage de l'application et enfin nous allons présenter quelques captures d'écran de l'application développée.

8.1. Création de la base de données

Une base de données permet de mettre des données à la disposition d'utilisateurs pour une consultation, une saisie ou bien une mise à jour, tout en s'assurant des droits accordés à ces derniers. Cela est d'autant plus utile que les données informatiques sont de plus en plus nombreuses.

La structure de **MySQL** a été amplement utilisée pour gérer et organiser les informations nécessaires à notre plateforme, offrant ainsi une solution fiable et efficiente pour la gestion de la base de données

Pour installer **MySQL** on tape la commande suivant **npm install mysql2**.

Une fois l'installation de **MySQL** terminée, nous procéderons à l'étape de connexion pour établir un lien entre notre plateforme et la base de donnée, permettant ainsi l'accès et la manipulation des données stockées. La figure 55 représente la connexion à la base de donnée.

```
1  const mysql = require('mysql2');
2  v const connection = mysql.createConnection({
3      host: 'localhost',
4      user: 'root',
5      password: '',
6      database: 'Fac_Science'
7  });
8
9  v connection.connect((err) => {
10 v   if (err) {
11       console.log('Erreur de connexion à MySQL :', err);
12       throw err;
13   }
14   console.log('Connecté à la base de données MySQL');
15   });
16
17  module.exports = connection;
```

Figure 55. Connexion à la base de donnée

8.2. Codage de l'application

8.2.1. Codage du côté Back-end

- **Création des routes du côté back-end**

ExpressJs, à l'instar de nombreux Framework web, agit comme un routeur où l'on définit les routes prises en charge par notre application, tous en spécifiant les actions à entreprendre dès que ces routes sont sollicitées. La figure 56 représente le routage en ExpressJs.

```
1  const express = require('express');
2  const cors = require('cors');
3  const app = express();
4  const port = 4000;
5
6  const authRoutes = require('./Authentification/Authentification');
7  const adminRoutes = require('./Administrateur/Administrateur');
8  const etudiantRoutes = require('./Etudiant/Etudiant');
9  const professeurRoutes = require('./Professeur/Professeur');
10 const messageRoutes = require('./Message/Message');
11
12 app.use(cors());
13
14 app.get('/', (req, res) => {
15   res.send('Bonjour à tous !');
16 });
17
18
19 app.use('/Authentification', authRoutes);
20 app.use('/Administrateur', adminRoutes);
21 app.use('/Etudiant', etudiantRoutes);
22 app.use('/Professeur', professeurRoutes);
23 app.use('/Message', messageRoutes);
24
25
26 app.listen(port, () => {
27   console.log(`Serveur en cours d'exécution sur le port ${port}`);
28 });
29
```

Figure 56. Routage ExpressJs

Voici un extrait de code utilisé pour l'authentification avec ExpressJs. La figure 57 représente l'authentification en ExpressJs.

```

16 router.post('/Login', async (req, res) => {
17   const { Email, Mot_de_passe } = req.body;
18   try {
19     const query = 'SELECT * FROM Compte WHERE Email = ?';
20     connection.query(query, [Email], async (err, rows) => {
21       if (err) {
22         console.error('Erreur lors de la recherche de l\'utilisateur :', err);
23         res.json({ error: 'Une erreur est survenue lors de la recherche de l\'utilisateur.' });
24         return;
25       }
26       console.log(rows);
27       if (rows.length === 0) {
28         res.status(404).json({ error: 'Utilisateur non trouvé.' });
29         return;
30       }
31       const user = rows[0];
32       const match = await bcrypt.compare(Mot_de_passe, user.Mot_de_passe);
33
34       if (match) {
35         const accessToken = generateAccessToken(user);
36         res.send(JSON.stringify(accessToken));
37       } else {
38         res.status(401).json({ error: 'Mot de passe incorrect.' });
39       }
40     });
41   } catch (error) {
42     console.error('Erreur lors de la connexion', error);
43     res.status(500).json({ error: 'Une erreur est survenue lors de la connexion' });
44   }
45 });

```

Figure 57. Authentification en ExpressJs

8.2.2. Codage du côté vu utilisateur

La figure 58 montre la fonctionnalité de récupération des données venant de l'API

```

28 useEffect(() => {
29
30   const getAllAnnonces = async () => {
31     try {
32       const response = await axios.get('http://localhost:4000/Administrateur/annonce');
33       setAnnouncements(response.data);
34       console.log(response.data);
35     } catch (error) {
36       console.error(error);
37     }
38   }
39   getAllAnnonces();
40
41 }, []);

```

Figure 58. Récupération des données vis à l'API

React Component : Les **composants** permettent de découper l'interface utilisateur en éléments indépendants et réutilisables, permettant ainsi de considérer chaque élément de manière isolée

8.3. Présentation de l'application

Nous allons maintenant visualiser l'interface de notre application que nous avons développée.

La figure 59 illustre la page d'authentification du côté des Professeurs et l'Administrateur de notre application.

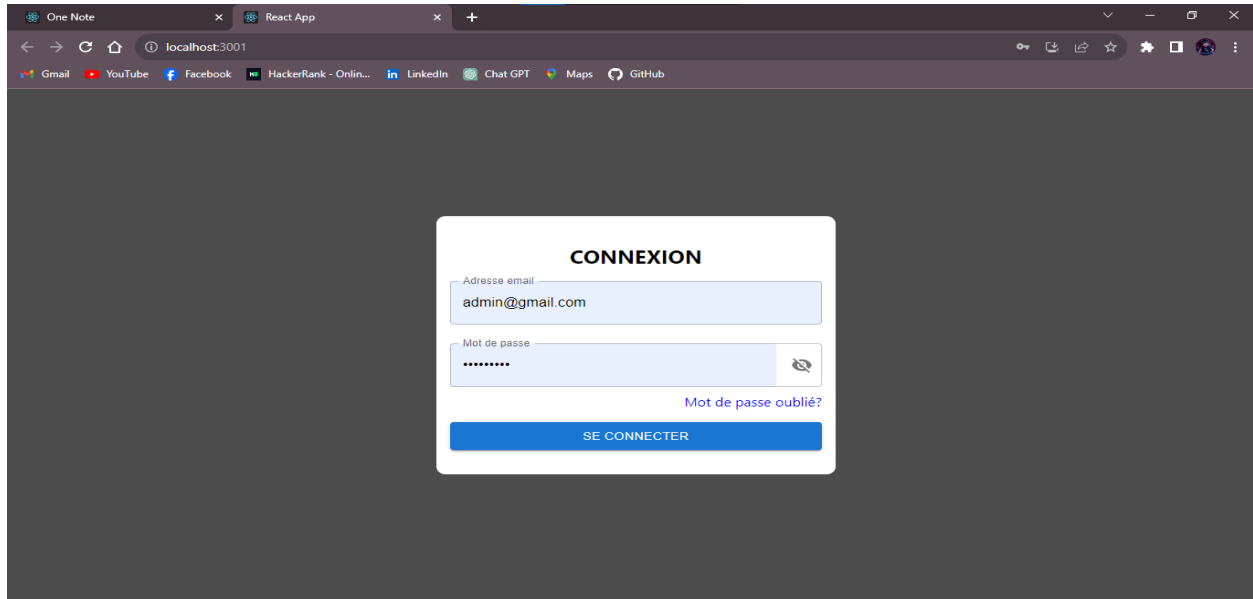


Figure 59. Page d'Authentification du côté Professeurs et Administrateur

La figure 60 illustre le tableau de bord (ou statistique) de l'application.

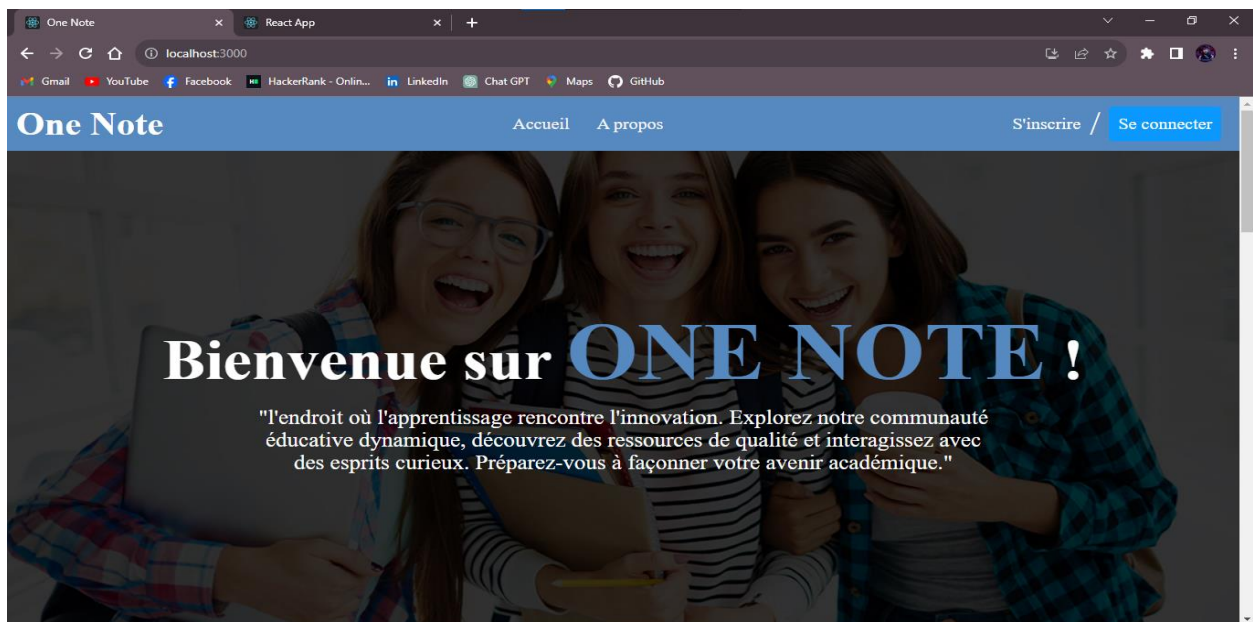


Figure 60. Dashboard

La figure 61 illustre l'interface de création de compte des professeurs

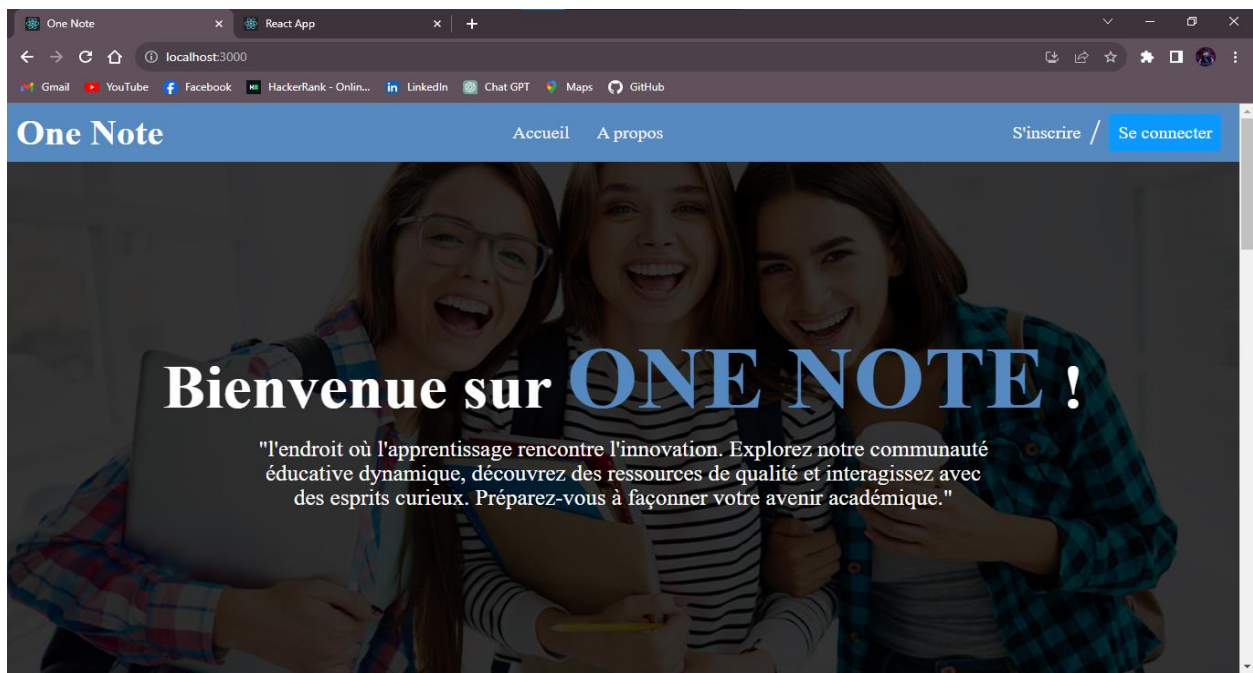


Figure 61. Interface de création de comptes professeur

La figure 62 illustre la liste des professeurs qui ont des comptes

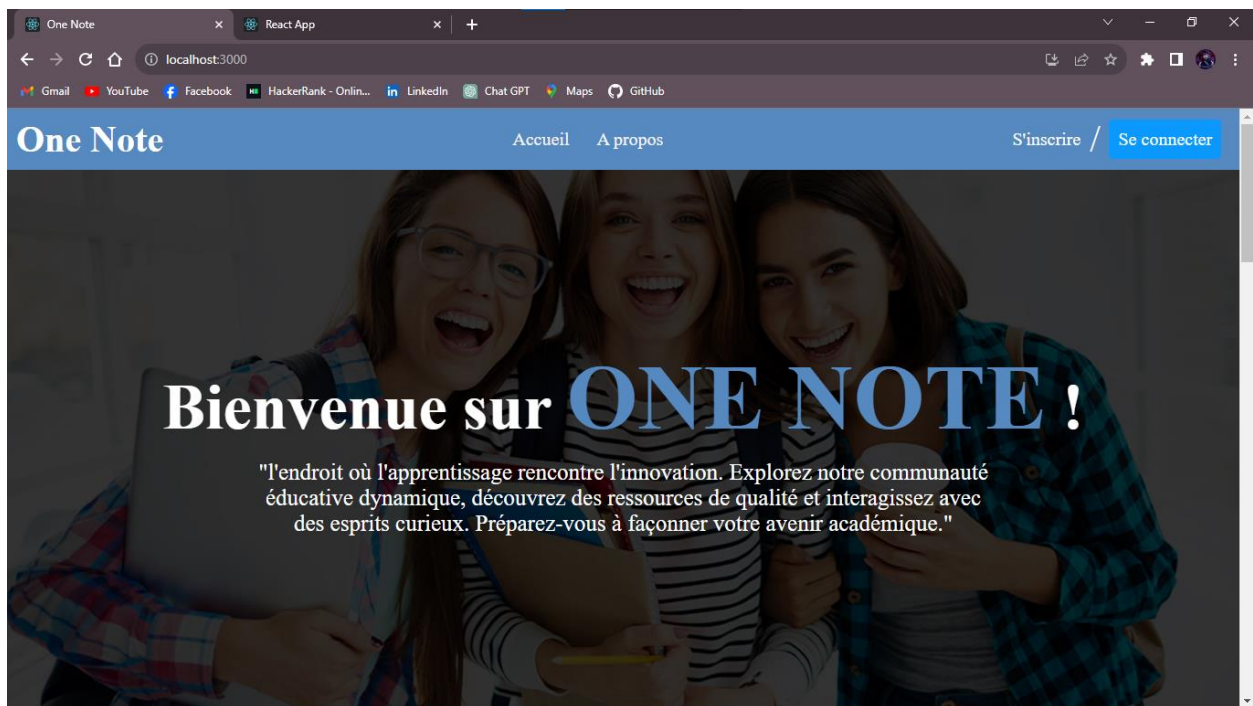


Figure 62. Liste des Professeurs

La figure 63 illustre l'interface de publication des annonces

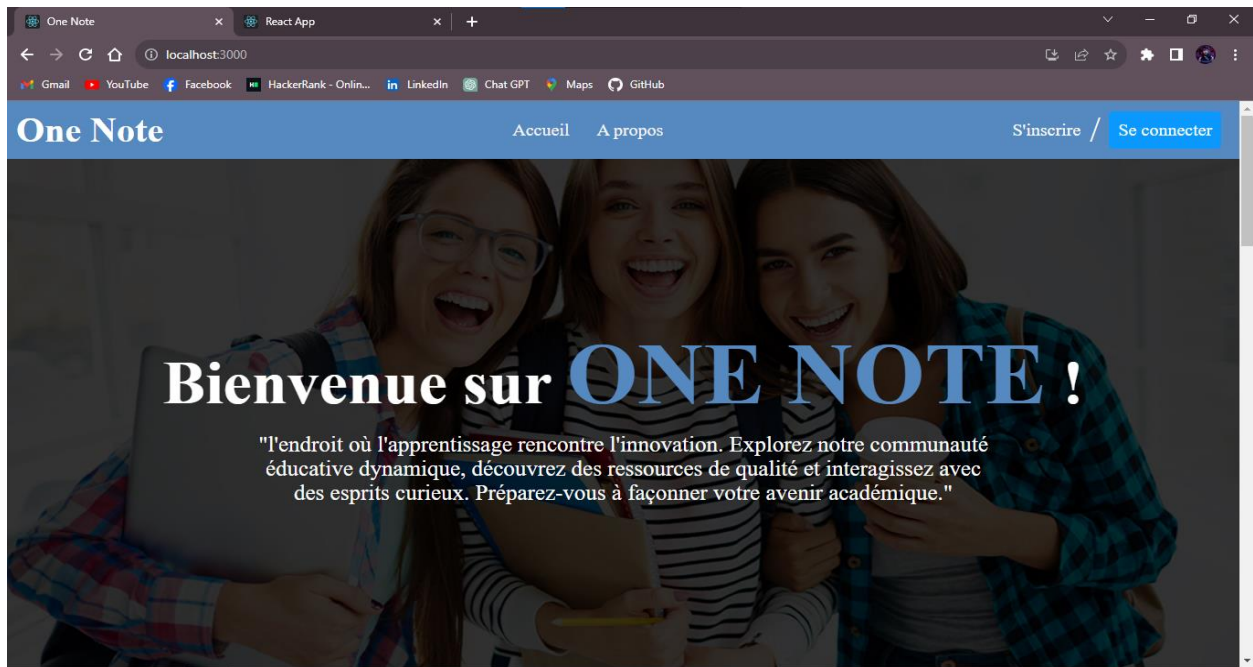


Figure 63. Interface de publication des annonces

La figure 64 illustre la liste des annonces publier

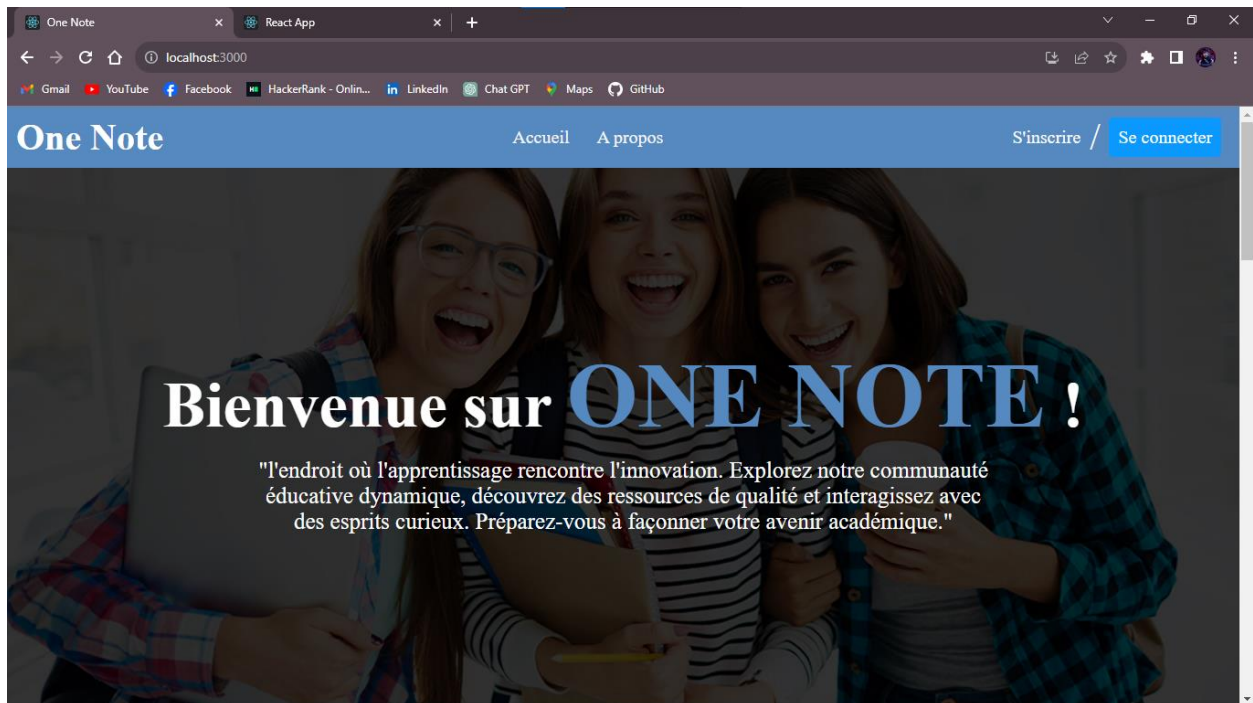


Figure 64. Liste des annonces publier

La figure 65 illustre l'interface de création des cours

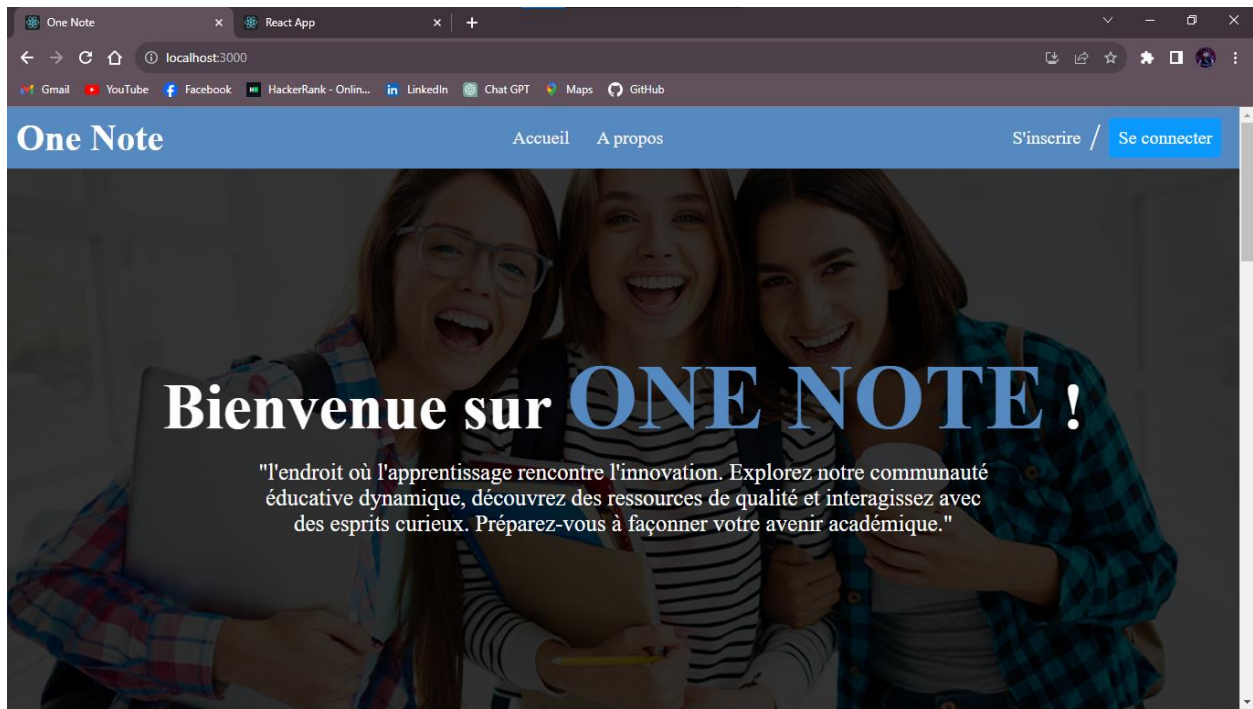


Figure 65. Interface de Création des cours

La figure 66 illustre la liste des cours

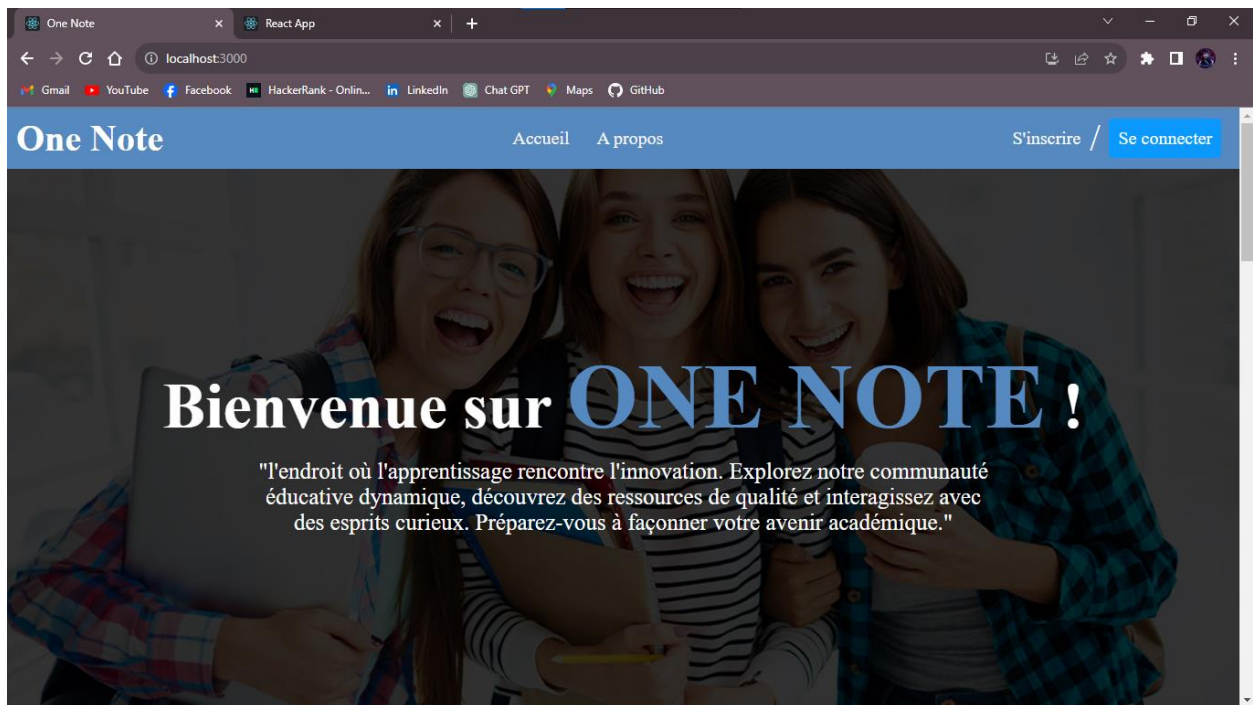


Figure 66. Liste des cours

La figure 67 illustre l'interface de création des devoirs

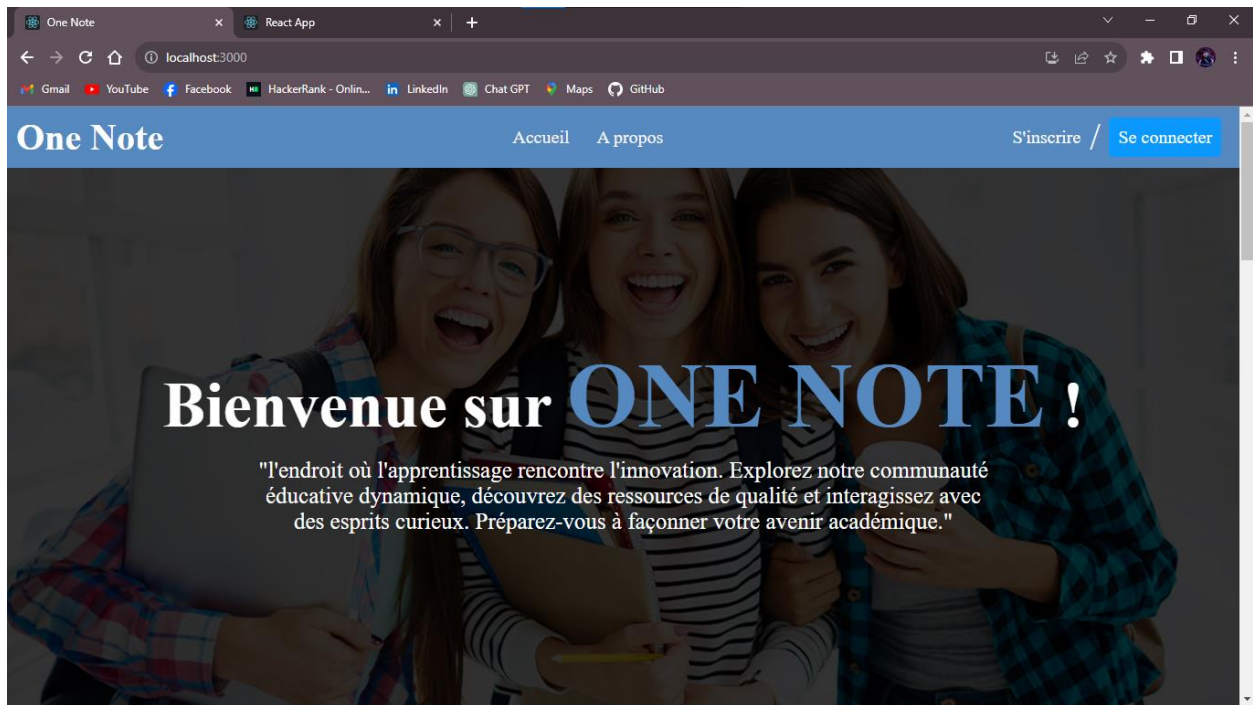


Figure 67. Interface de Création des devoirs

La figure 68 illustre la liste des devoirs

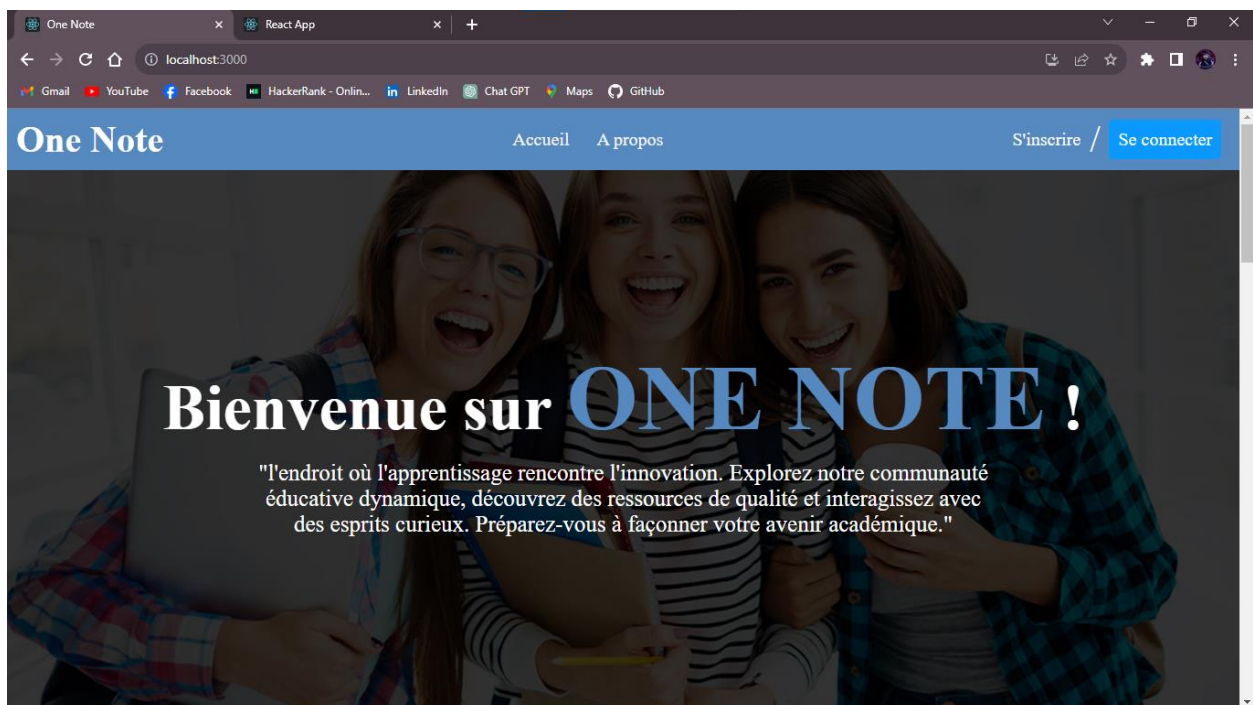


Figure 68. Liste des devoirs

La figure 69 illustre la page d'accueil

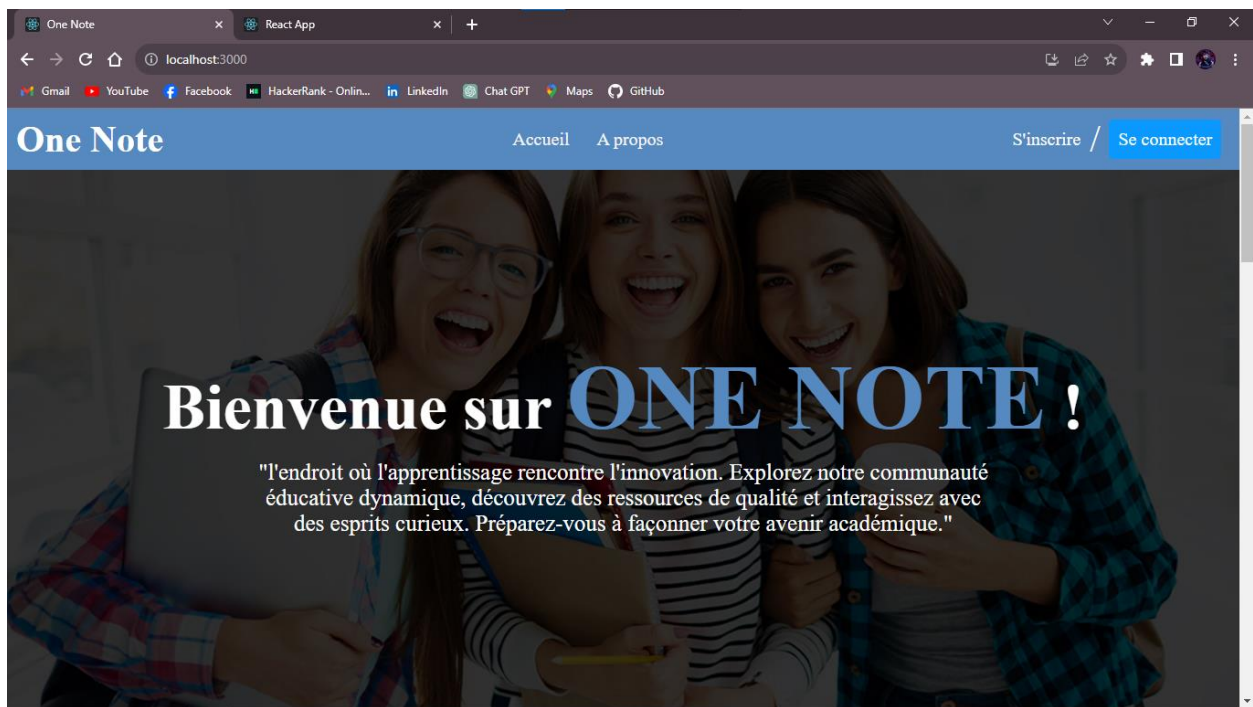


Figure 69. Page d'accueil

La figure 70 illustre l'interface d'authentification des étudiants

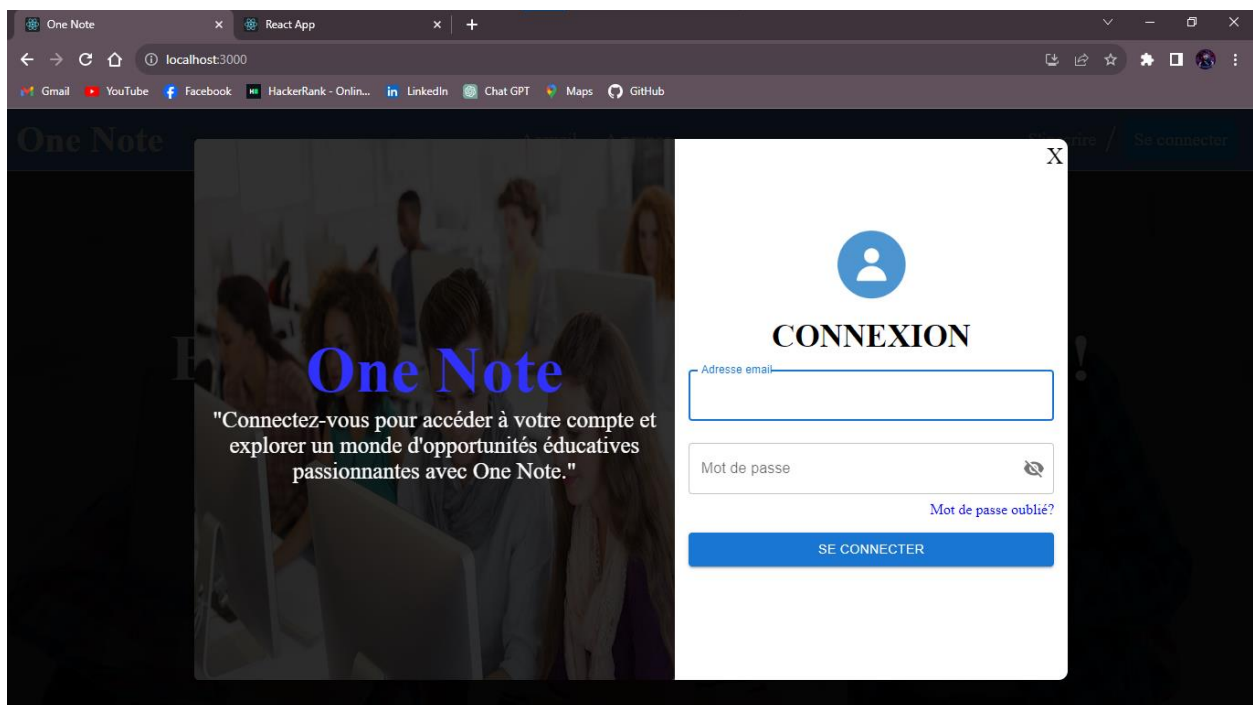
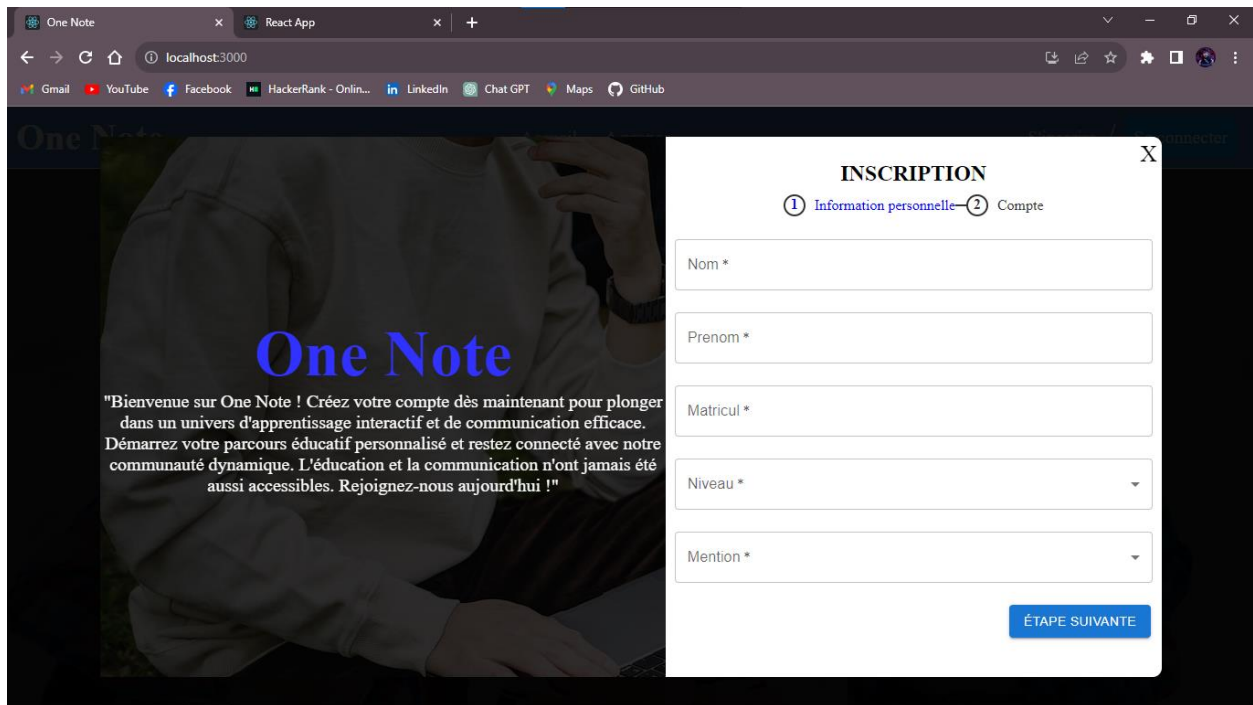


Figure 70. Interface d'Authentification des étudiants

La figure 71 illustre l'interface d'inscription des étudiants sur l'application



The screenshot shows a web browser window with the URL 'localhost:3000'. The page features a dark background with a person's hands typing on a laptop. On the right, a white modal titled 'INSCRIPTION' is open. It has two steps: '1 Information personnelle' (active) and '2 Compte'. The form includes fields for 'Nom *', 'Prenom *', 'Matricul *', 'Niveau *' (a dropdown menu), and 'Mention *' (a dropdown menu). A blue button labeled 'ÉTAPE SUIVANTE' is at the bottom right of the modal. The background text on the left reads: 'Bienvenue sur One Note ! Créez votre compte dès maintenant pour plonger dans un univers d'apprentissage interactif et de communication efficace. Démarrez votre parcours éducatif personnalisé et restez connecté avec notre communauté dynamique. L'éducation et la communication n'ont jamais été aussi accessibles. Rejoignez-nous aujourd'hui !'

Figure 71. Interface d'inscription des étudiants

La figure 72 illustre la liste des annonces publiés

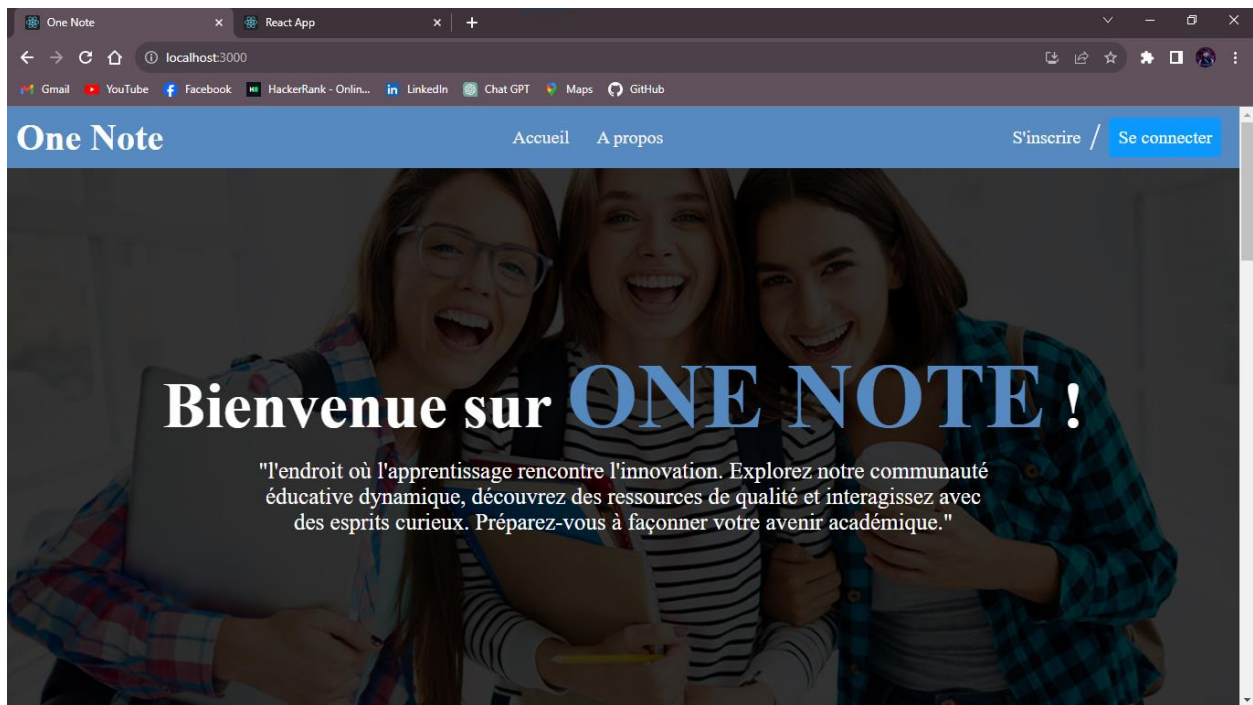


Figure 72. List des annonces

La figure 73 illustre la liste des cours

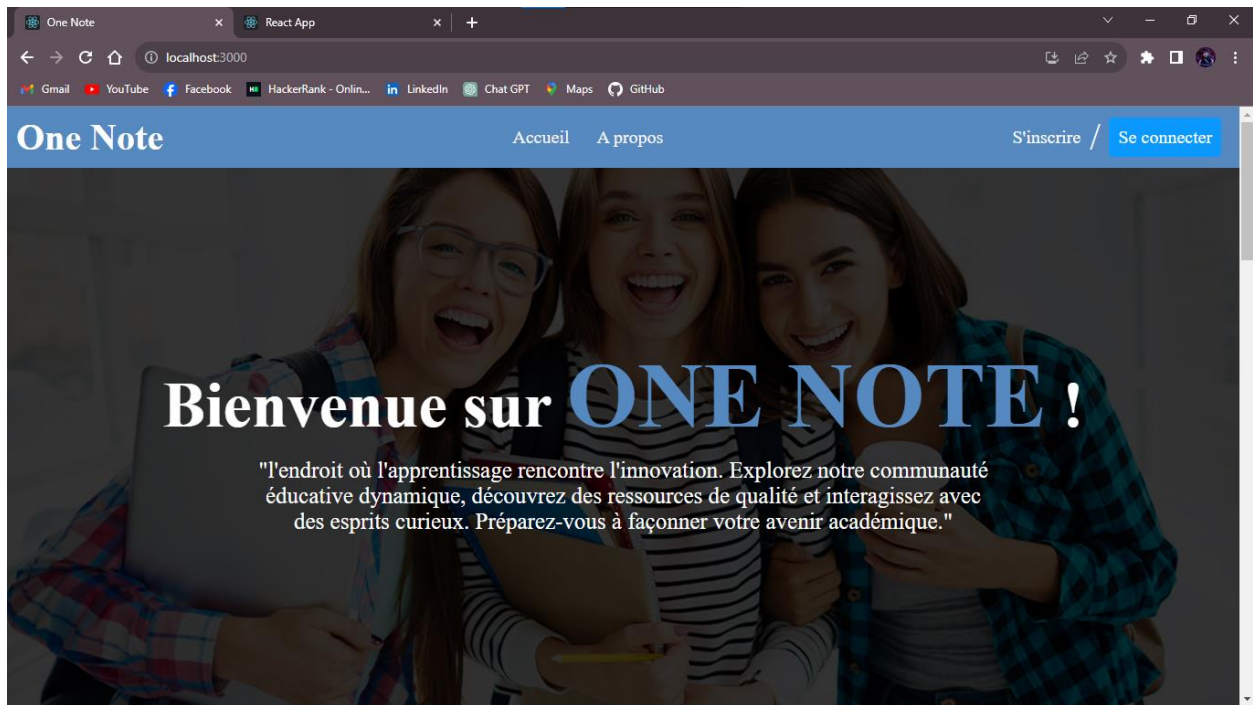


Figure 73. Liste des cours

La figure 74 illustre la liste des devoirs

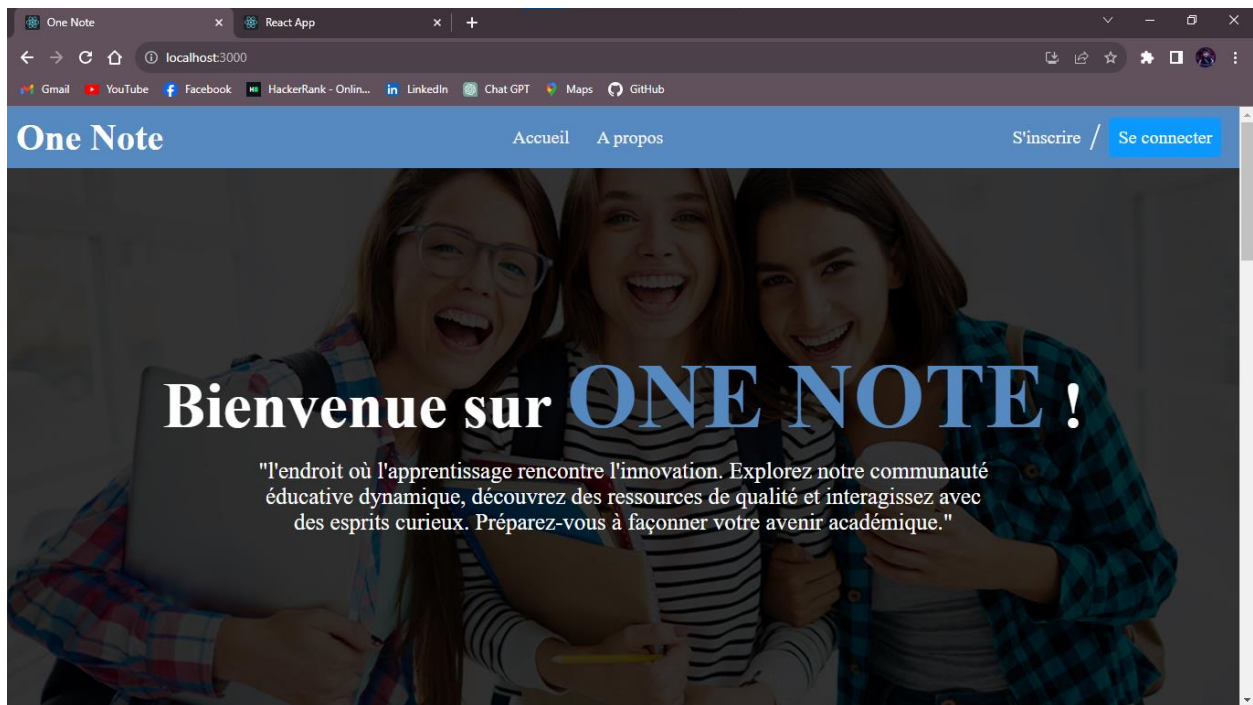


Figure 74. Liste des devoirs

La figure 75 illustre la liste des professeurs

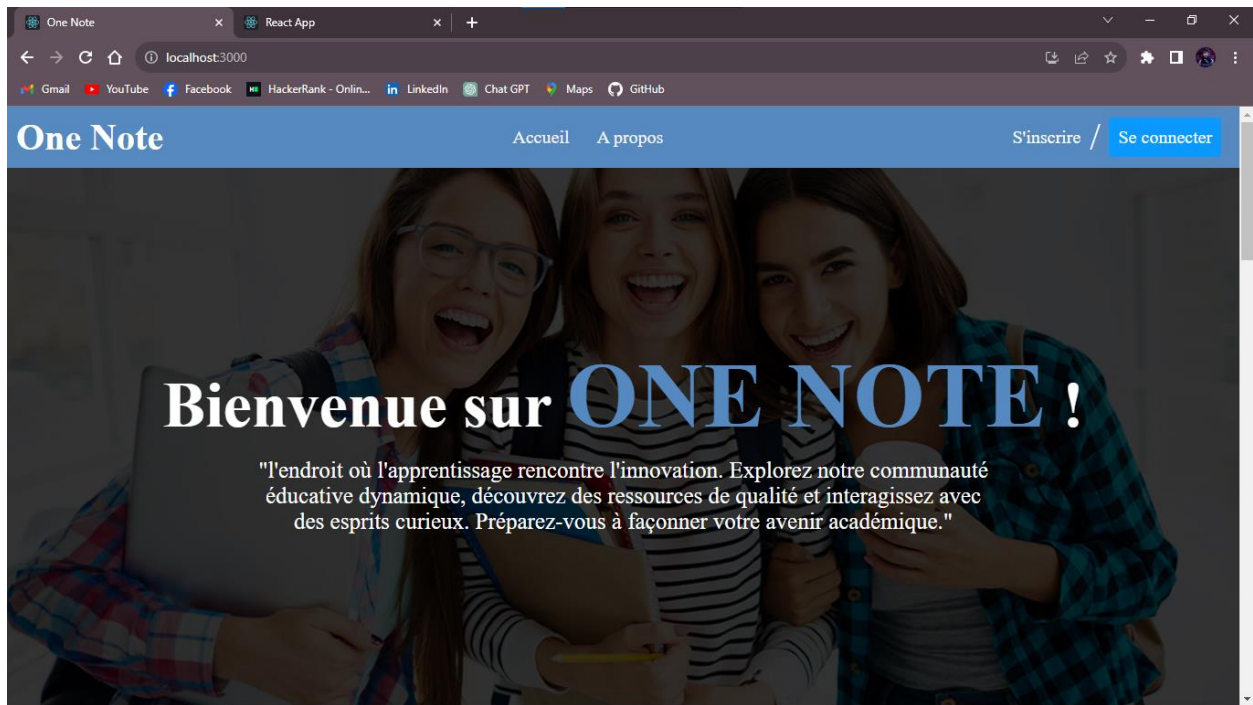


Figure 75. Liste des professeurs

La figure 76 illustre l'interface de paramètre des comptes étudiants pour modifier son profil

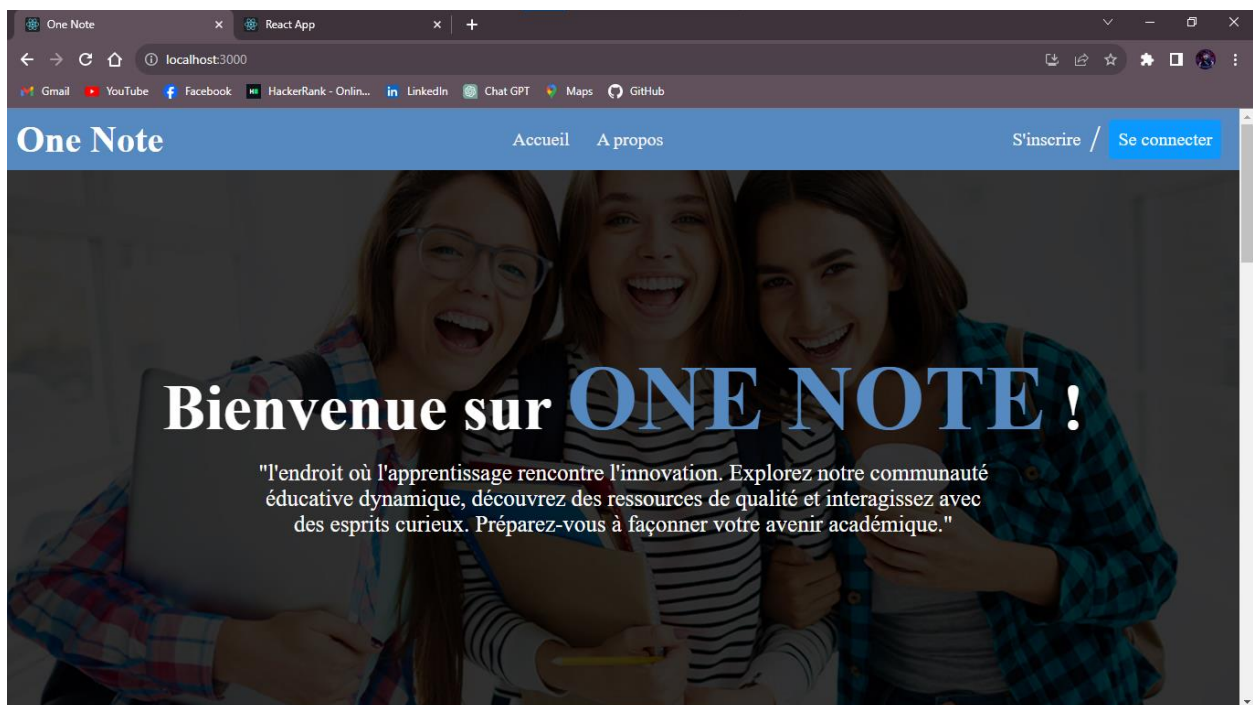


Figure 76. Interface de paramètre de compte

La figure 77 illustre l'interface de notification

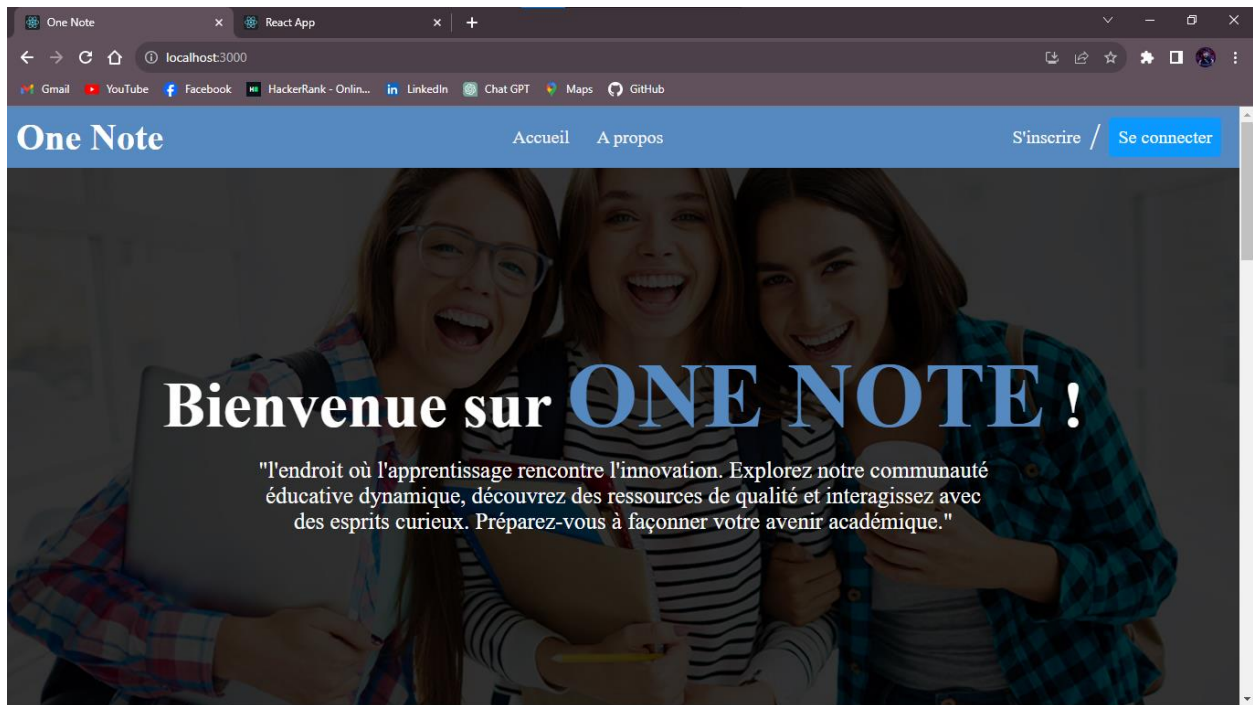


Figure 77. Interface de notification

La figure 78 illustre l'interface pour voir un cours, un devoirs et annonces

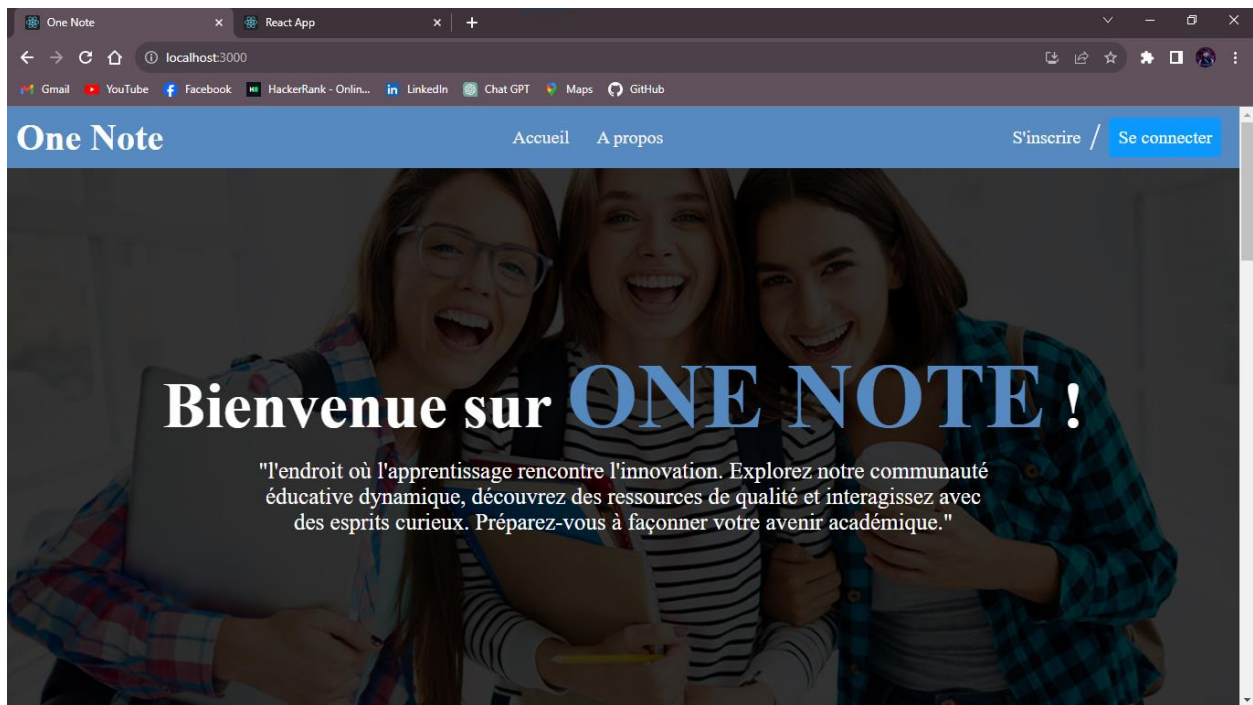


Figure 78. Interface pour regarder les ressource dans l'application

CONCLUSION

En conclusion, la réalisation d'un projet exige une solide conception et une organisation méticuleuse, particulièrement dans le cadre de l'alternance. Notre projet, axé sur la création d'une plateforme éducative en ligne, a été guidé par une attention constante aux besoins des acteurs clés : l'administrateur, les enseignants et les étudiants. L'application a réussi à répondre de manière complète à ces besoins fonctionnels, avec la transformation de chaque spécification de l'analyse en module au sein de l'application, ces modules étant accessibles en fonction des droits d'accès correspondants.

Pour mener à bien ce projet, nous avons adopté la méthode 2TUP de conception, utilisant le langage de modélisation UML. Le développement s'est appuyé sur la technologie Node.js et React.js et le système de gestion de base de données MySQL. Notre outil de modélisation, VP-UML, a également joué un rôle crucial dans la création de la plateforme. Grâce à cette initiative, nous confirmons notre engagement envers l'amélioration de l'accès à l'éducation, l'amélioration de la qualité de l'apprentissage en ligne, et la promotion de l'efficacité pédagogique.

Néanmoins, il convient de noter que la plateforme n'est pas encore accessible au public. Bien qu'opérationnelle, de nombreuses améliorations restent envisageables, notamment le développement d'une application mobile pour accroître sa disponibilité. En fin de compte, cette expérience d'alternance a constitué une étape enrichissante. Elle a permis, d'une part, l'application pratique et l'approfondissement des connaissances acquises au fil de ma formation à l'ENI, et, d'autre part, la familiarisation avec la gestion de projets informatiques, ouvrant ainsi la voie à une exploration plus poussée du milieu professionnel.

BIBLIOGRAPHIE

- [1] Joseph Gabay et David Gabay, **UML2 analyse et conception**, Edition Eyrolles, Paris 2008, 242 pages.
- [2] Pascal Roques, **2TUP**, Edition Eyrolles 2015, 31 pages.
- [3] Claus T. Jensen, **Les API POUR LES NULS**, Edition limitée IBM 2015, 53 pages.
- [4] Sébastien Castiel, **Redécouvrir Javascript avec Node.js**, Edition Développer 2017, 34 pages.

WEBOGRAPHIE

- [5] <https://www.ambient-it.net/reactjs-vs-angular-vs-vuejs-que-choisir-en-2023/>, Que choisir entre ReactJs, VueJs et Angular, consulté le
- [6] <https://practicalprogramming.fr/express-js>, Express.Js, consulté le
- [7] <https://nodejs.org/en/download>., NodeJs, consulté le
- [8] <https://Reactjs.org/en/download>., site officiel pour la documentation de ReactJs, consulté le
- [9] <https://MaterialUI.com>, site officiel de Material UI pour ReactJs, consulte le

GLOSSAIRE

TABLE DES MATIERES

CURRICULUM VITAE.....	I
SOMMAIRE GENERAL.....	III
REMERCIEMENTS.....	V
LISTE DES FIGURES.....	VI
LISTE DES TABLEAUX.....	VIII
LISTE DES ABREVIATIONS.....	IX
INTRODUCTION GENERALE.....	1
PARTIE I : PRESENTATION.....	2
Chapitre 1. Présentation de l'Ecole Nationale d'Informatique.....	3
1.1. Information d'ordre général.....	3
1.2. Missions et historiques.....	3
1.3. Organigramme institutionnel.....	5
1.4. Domaine de spécialisation.....	6
1.5. Architecture des formations pédagogiques.....	6
1.6. Relations de l'ENI avec l'organisme externes.....	8
1.7. Débouches professionnel et diplôme	9
1.8. Ressources humaines.....	10
Chapitre 2. Présentation de la faculté de Science Fianarantsoa.....	11
2.1. Information général.....	11
2.2. Historiques.....	11
2.3. Offre des formations.....	12
2.3.1. Ancien système.....	12
2.3.2. Système LMD.....	13
2.4. Les personnels administratifs et techniques.....	14
Chapitre 3. Description du projet.....	15
3.1. Formulation.....	15
3.2. Objectif et besoins de l'utilisateur.....	15
3.2.1. Objectif.....	15
3.2.2. Besoin d'utilisateur.....	15
3.3. Moyens nécessaires à la réalisation du projet.....	16
3.3.1. Moyens Humains.....	16
3.3.2. Moyens Matériels.....	16
3.3.3. Moyens Logiciels.....	17
3.4. Résultats attendus.....	17
3.5. Chronogramme de travail.....	18
PARTIE II : ANALYSE ET CONCEPTION.....	19
Chapitre 4. Analyse préalable.....	20
4.1. Analyse de l'existant.....	20
4.1.1 Organisations actuelle.....	20
4.1.2 Inventaire des moyens matériels et logiciels.....	20
4.2. Critique de l'existant.....	20
4.3. Conception avant-projet.....	21
4.3.1 Proposition des solutions.....	21
4.3.2 Méthodes de conception et outils utilisés.....	22

Chapitre 5. Analyse conceptuelle.....	29
5.1. Présentation de la méthode utilisée.....	29
5.2. Dictionnaire des données.....	29
5.3. Règles des gestions	30
5.4. Représentation et spécifications des besoins.....	32
5.4.1. Diagramme des cas d'utilisation.....	32
5.4.2. Description textuelle de chaque cas d'utilisation.....	35
5.4.2. Priorisation des cas d'utilisation.....	42
5.4.3. Diagramme de séquence système pour chaque cas d'utilisation.....	42
5.5. Spécification des besoins techniques.....	50
5.6. Modélisation du domaine.....	51
Chapitre 6. Conception détaillée.....	52
6.1. Architecture de système	52
6.2. Diagramme de séquence de conception pour chaque cas d'utilisation.....	53
6.3. Diagramme de classe de conception pour chaque cas d'utilisation.....	60
6.4. Diagramme de classe de conception globale.....	64
6.5. Diagramme de paquetage.....	65
6.6. Diagramme de déploiement.....	65
PARTIE III : REALISATION.....	66
Chapitre 7. Mise en place de l'environnement de développement.....	67
7.1. Installation et configuration des outils.....	67
7.1.1. Visual Paradigm.....	67
7.1.2. Laragon.....	68
7.1.3. Visual Studio Code.....	69
7.1.4. NodeJs.....	70
7.2. Architecture de l'application.....	72
Chapitre 8. Développement de l'application.....	73
8.1. Création de base de données.....	73
8.2. Codage de l'application.....	73
8.2.1. Codage du cote Back-end.....	73
8.2.2. Codage du cote vue utilisateur.....	75
8.3. Présentation de l'application.....	77
CONCLUSION.....	85
BIBLIOGRAPHIE.....	X
WEBOGRAPHIE.....	XI
GLOSSAIRE.....	XII
TABLE DE MATIERE.....	XIII
RESUME.....	XV
ABSTRACT.....	XV

RESUME

Notre projet ambitieux consiste à concevoir et à développer une plateforme éducative en ligne novatrice, visant à révolutionner l'apprentissage en offrant aux enseignants un espace pour publier des cours interactifs, des ressources pédagogiques variées (allant de documents PDF à des vidéos), et la possibilité de créer des devoirs. Du côté des étudiants, la plateforme leur offre la possibilité de s'inscrire aux cours, d'accéder aisément aux contenus pédagogiques associés, de soumettre leurs devoirs en ligne, et de favoriser la communication avec les enseignants pour une expérience d'apprentissage en ligne enrichissante et collaborative. Ce projet est guidé par un profond désir d'améliorer l'accès à l'éducation et de créer un environnement éducatif innovant et efficace, avec une priorité sur la simplicité d'utilisation, la sécurité des données et la qualité pédagogique.

Mots clés : Conception - Communication - Education - Plateforme web – Réalisation

ABSTRACT

Our ambitious project aims to design and develop an innovative online educational platform that seeks to revolutionize learning. It provides teachers with a space to publish interactive courses, diverse pedagogical resources (ranging from PDF documents to videos), and the ability to create assignments. On the student side, the platform allows them to enroll in courses, easily access associated educational content, submit assignments online, and facilitate communication with teachers for a rich and collaborative online learning experience. This project is guided by a deep desire to improve access to education and create an innovative and effective educational environment, with a focus on user-friendliness, data security, and pedagogical quality.

Keywords: Design – Communication – Education - Realization – Web platform