

كلية العلوم
+٥٢٤٧٥١١ | +٢٠٥٥٥٤٧
FACULTÉ DES SCIENCES



UNIVERSITE IBN ZOHR
FACULTE DES SCIENCES

Département Informatique
Filière Sciences Mathématiques et Informatique

PFE

Présenté par :

Rachida Tanassat

Hicham El Moudni

Pour l'obtention de la

Licence en Sciences Mathématiques et Informatique

**CONCEPTION ET DEVELOPPEMENT
D'UNE APPLICATION WEB D'AIDE
A LA GESTION DES ENSEIGNANTS**

Soutenu le 16 Juin 2022

Encadré par : Pr. Salwa Belaqziz

Année universitaire 2021-2022

« Page laissée intentionnellement vide »

Dédicace

*À nos chers parents, pour tous leurs soutiens, leurs sacrifices,
leur tendresse et leurs prières tout au long de nos études,
À nos chers frères, pour leur appui et leur encouragement,
À nos chères sœurs pour leur encouragement permanent, et leur
soutien moral,
À tous nos amis pour leur soutien tout au long de nos parcours
universitaire,
Que ce travail soit l'accomplissement de vos vœux tant allègues,
et le fruit de votre soutien infailible,
Merci d'être toujours là pour nous.*

Remerciements

En premier lieu, nous aimerions à remercier le Dieu de nous avoir donné le courage et la volonté de réaliser ce projet.

Nous tenons à exprimer également notre profonde reconnaissance à notre encadrante Mme. Salwa BELAQZIZ, enseignante à la Faculté des Sciences d'Agadir, pour son suivi et ses directives avisées tout au long de notre projet, ses précieuses remarques et conseils pertinents. Sa lecture minutieuse de ce document a beaucoup amélioré sa clarté. Puisse-t-elle trouver ici l'expression de notre profond respect et de toute notre reconnaissance.

Nous adressons également nos remerciements à l'équipe de l'administration de l'Ecole Nationale d'Architecture d'Agadir (ENAA) pour leurs accueils et nous remercions particulièrement Mme. Fatima Zohra Laababi qui était à notre disposition pour nous clarifier leur besoins et pour répondre à toutes nos questions durant notre projet.

Nous exprimons notre sincère gratitude aux membres du jury, qui ont accepté d'évaluer ce projet de fin d'études.

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis qui nous ont toujours soutenu et encouragé au cours de la réalisation de ce projet de fin d'études.

RESUME

L'éducation est le point de départ pour les pays en développement, pour améliorer ce domaine, nous devons faire appel aux technologies modernes.

Dans ce contexte, notre projet consiste à concevoir et développer une application web pour la gestion des enseignants d'un établissement universitaire. Cette application a été réalisée suite à un besoin exprimé par l'administration de l'École Nationale d'Architecture d'Agadir. Cette dernière a présenté sa satisfaction de l'application proposée et compte l'utiliser au sein de l'établissement.

L'application développée est basée sur l'utilisation de différentes technologies tels que le Framework Laravel, le langage PHP, la base de données MySQL etc...

Pour la bonne conception de notre application Merise est la méthode de conception qui a été mise en place, et comme langage, nous avons utilisé UML.

Mots clés : Application web, gestion des enseignants, Laravel, PHP, MYSQL, Merise, UML.

ABSTRACT

Education is the starting point for developing countries. To improve this area, we must make use of modern technologies.

In this context, our project consists in developing a web application for the management of teachers in a university establishment. This application is created following a need expressed by the administration of the National School of Architecture of Agadir. This latter presented its satisfaction with the proposed application, and intends to use it within the establishment.

Our developed application is based on the use of different technologies such as the Laravel Framework, the PHP language, the MySQL database etc...

On the other hand, we used Merise as the design method, and UML as the design language of our application.

Keywords: Web application web, teachers' management, Laravel, PHP, MYSQL, Merise, UML.

TABLE DES MATIERES

DEDICACE.....	ii
REMERCIEMENTS.....	iii
RESUME.....	iv
ABSTRACT	v
TABLE DES MATIERES	vi
LISTE DES FIGURES.....	viii
LISTE DES TABLEAUX	ix
LISTE DES ABREVIATIONS	x
INTRODUCTION GENERALE	1
CHAPITRE 1 : CONTEXTE GENERAL DU PROJET	3
1- CONTEXTE DU PROJET	3
2- PROBLEMATIQUE ET OBJECTIFS.....	5
3- DESCRIPTION DU PROJET.....	6
a. Présentation de l'ENA d'Agadir	6
b. La formation à l'ENA	6
c. Les formes d'enseignement à l'ENA.....	6
4- CONDUITE DU PROJET.....	7
5- PLANIFICATION DU PROJET	9
CHAPITRE 2 : ANALYSE ET CONCEPTION.....	11
1- SPECIFICATION DES BESOINS.....	11
a. Les besoins fonctionnels	11
b. Les besoins non fonctionnels	12
2- CONCEPTION.....	13
a. Méthode et logiciel de conception	13
b. Diagrammes de cas d'utilisation	14
c. Diagrammes de séquences	23
d. Diagramme MCD et classe	26
CHAPITRE 3 : ETUDE TECHNIQUE ET ENVIRONNEMENT	29
1- CAPTURE DES BESOINS TECHNIQUES.....	29
2- ARCHITECTURE ADOPTEE	29
3- CHOIX DES LANGAGES	31
4- FRAMEWORK UTILISE.....	32
5- ENVIRONNEMENTS MATERIELS, LOGICIELS, OUTILS.....	33
CHAPITRE 4 : REALISATIONS, INTERFACES, TESTS	36
CONCLUSION GENERALE	47
WEBOGRAPHIE.....	48

LISTE DES FIGURES

Figure 1 : Exemple de document qui contient les données des enseignants	4
Figure 2 : Exemple de contrat des enseignants	5
Figure 3 : Le processus 2TUP.....	8
Figure 4 : tableau du planning général suivis	9
Figure 5 : diagramme de Gantt de planification générale suivis	10
Figure 6 : Diagramme de cas d'utilisation globale	15
Figure 7 : Diagramme de cas d'utilisation de l'administrateur	16
Figure 8 : Diagramme de cas d'utilisation de l'enseignant	17
Figure 9 : Diagramme de cas d'utilisation de l'étudiant	17
Figure 10 : Diagramme de séquence d'authentification	23
Figure 11 : Diagramme de séquence d'ajout	24
Figure 12 : Diagramme de séquence de modification	24
Figure 13 : Diagramme de séquence de recherche	25
Figure 14 : Diagramme de séquence de suppression	26
Figure 15 : Diagramme MCD	27
Figure 16 : Diagramme de classe.....	28
Figure 17 : L'architecture MVC	29
Figure 18 : Interface d'authentification.....	36
Figure 19 : page d'accueil de l'administrateur	37
Figure 20 : ajouter un enseignant	37
Figure 21 : la liste des enseignants	38
Figure 22 : la liste des matières	39
Figure 23 : fiche d'un enseignant	39
Figure 24 : imprimer la convention de vacation	40
Figure 25 : voir le diplôme d'un enseignant	41
Figure 26 : la liste des diplômes	41
Figure 27 : publier une annonce	42
Figure 28 : Page d'accueil de l'enseignant	43
Figure 29 : le profil d'un enseignant	43
Figure 30 : publier un cours	44
Figure 31 : la liste des cours	44

Figure 32 : chercher un cours	45
Figure 33 : la liste des cours	46
Figure 34 : les emails “mailtrap”	46

LISTE DES TABLEAUX

Tableau 1 : la description du cas d'utilisation "S'inscrire et S'authentifier"	18
Tableau 2 : la description du cas d'utilisation "Ajouter un enseignant"	18
Tableau 3 : la description du cas d'utilisation "Lister les enseignants"	19
Tableau 4 : la description du cas d'utilisation " Modifier un enseignant"	19
Tableau 5 : la description du cas d'utilisation "Supprimer un enseignant"	20
Tableau 6 : la description du cas d'utilisation " Chercher un enseignant"	20
Tableau 7 : la description du cas d'utilisation " Gérer les annonces"	21
Tableau 8 : la description du cas d'utilisation " Gérer les cours"	21
Tableau 9 : la description du cas d'utilisation " Gérer le profile"	22
Tableau 10 : la description du cas d'utilisation " Rechercher des cours"	22
Tableau 11 : la description du cas d'utilisation " Visualiser les annonces"	23
Tableau 12 : les propriétés de l'appareil 1	33
Tableau 13 : les propriétés de l'appareil 2	33

LISTE DES ABREVIATIONS

ENAA : Ecole Nationale d'Architecture d'Agadir

2TUP : Two Tracks Unified Process

MVC : Model Vue Control

HTTP: HyperText Transfer Protocol

CSS : Cascading Style Sheets

HTML : Hypertext Markup Language

UML : Unified Modeling Language

TD : Travaux Dirigés

TP : Travaux Pratiques

INTRODUCTION GENERALE

Aujourd'hui, l'informatique joue un rôle prépondérant dans le domaine éducatif. Il s'agit d'une solution pédagogique multifacette et ses supports numériques qui peuvent être la clé pour motiver les étudiants à évaluer leurs niveaux. En effet, l'informatique peut rendre l'apprentissage plus facile et plus accessible, et permet également d'aider les enseignants à transférer l'information sans peine et sans trop d'efforts.

D'autre part, les administrateurs des établissements universitaires peuvent ainsi profiter de ses techniques modernes pour bien organiser leur travail.

Dans ce cadre, l'objectif de notre projet est de développer, voire même optimiser une application web permettant de gérer les enseignants et leurs cours, et qui est destinée principalement aux enseignants et administrateurs de l'Ecole Nationale d'Architecture d'Agadir (ENAA). Le développement de cette application a été effectué suite à un certain nombre de difficultés constatées, telles que :

- La perte du temps lors de la saisie des données manuellement ou lors de la recherche dans les archives.
- Le problème de sécurité et de la fiabilité des données.
- Absence d'une interface pour le partage des cours.
- La perte du temps du côté de l'étudiant lors de la recherche d'un cours.

Ce rapport est organisé de la manière suivante :

Dans le premier chapitre « Contexte général du projet », on fera d'abord une étude de l'existant afin de l'analyser et d'identifier le cahier des charges pour éclaircir les idées préconçues de notre application, puis on présentera l'objectif général et les principales solutions suggérées par notre projet en tenant compte de cette étude.

Dans le deuxième chapitre « Analyse et conception », nous présenterons la méthode suivie pour la conduite du projet et sa planification, et nous définissons les besoins fonctionnels et techniques des différents acteurs principaux de l'application.

Le troisième chapitre intitulé « Etude technique et environnement », on présentera le langage de programmation et les différents outils manipulés dans le développement de notre application.

Le quatrième chapitre intitulé « Réalisations » est consacré à la présentation des résultats de l'implémentation de l'application à travers les interfaces réalisées avec les descriptions.

Et nous terminons enfin, par une conclusion générale qui permettra de présenter une synthèse globale du travail.

CHAPITRE 1 : CONTEXTE GENERAL DU PROJET

1- Contexte du projet

Durant nos visites à l'Ecole Nationale d'Architecture d'Agadir, nous avons analysé les différentes étapes de la gestion des enseignants réalisée par les administrateurs et la gestion des cours réalisée par les enseignants.

Ces tâches sont principalement les suivantes :

- Un gestionnaire de la paie des enseignants joue un rôle très important dans la gestion des opérations financières telles que la préparation de la paie, les états des salaires. Pour effectuer ces tâches de façon parfaite, il est nécessaire de vérifier continuellement les données bancaires d'un employé. L'école ENAA a donc trouvé des problèmes concernant la gestion de paiement des enseignants suite à ces opérations qui se font manuellement. En effet, à chaque fois que le gestionnaire souhaite réaliser un paiement à un enseignant, il est obligé de vérifier les coordonnées de ce dernier (le relevé d'identité bancaire, Carte Nationale d'Identité ...) dans des archives (Figure 1), ce qui mène à une perte de temps et parfois d'argent en cas d'erreur.
- Un administrateur doit gérer les contrats des enseignants (Figure 2) lors de leur recrutement en tant qu'enseignants permanents ou vacataires. Cette tâche est aussi gérée d'une façon manuelle, c'est-à-dire qu'il n'existe pas de base de données pour le stockage et la gestion des différentes informations liées aux enseignants.
- Concernant les enseignants, on a constaté qu'il y a un besoin de gestion des cours et des travaux pratiques. En effet, il n'y a aucune plateforme mise à leur disposition où ils peuvent déposer les cours et les différents documents à leurs étudiants.

Royaume du Maroc
Ministère de l'Aménagement du Territoire
National, de l'Urbanisme, de l'Habitat
et de la Politique de la Ville

Aménagement du Territoire National et Urbanisme
Secrétariat Général

Ecole Nationale d'Architecture Agadir



المملكة المغربية
وزارة إعداد التراب الوطني والتعمير
والإسكان وسياسة المدينة

قطاع إعداد التراب الوطني والتعمير
الكتابة العامة

المدرسة الوطنية للهندسة المعمارية أكادير

**Etat Collectif des Indemnités de Vacation ENA AGADIR de
L'année Universitaire 2020/2021 Semestre (S2/S4/)**

Nom	Prénom	CIN	Type de Diplôme	Grade	Taux	Date d'obtention de Diplôme	Matière enseignés	L'année Universitaire	Semestre	RIB
XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX
XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX

Ecole Nationale d'Architecture Agadir



المدرسة الوطنية للهندسة المعمارية أكادير

Complexe Universitaire Ibn Zohr. Quartier Dakhla. BP 20732. Agadir Principale. 80000. Agadir. Tel/Fax : +212 528 22 50 41. www.enaagadir.ma

Figure 1 : Exemple de document qui contient les données des enseignants

<p>ROYAUME DU MAROC Ministère de l'Enseignement Supérieur, de la Recherche Scientifique et de l'Éducation École Nationale d'Architecture AGADIR</p> <p>CONVENTION DE VACATION L'année universitaire 2020/2021</p> <p>Entre : L'École Nationale d'Architecture d'AGADIR (ENA-AGADIR), désignée ci-après par « ENA-AGADIR », siégeant au complexe universitaire d'El Zohir Hay Dakhla, représentée par son Directeur Mr TITTA Mohammed :</p> <p>D'une part</p> <p>Et : Monsieur : YOUNASS Diplôme : Architecte Identification fiscale ou Ordre et école : Adresse : 109, maison de l'Avocat, Avenue HANAN B - Agadir Tel : Bureau : 0528841637 Domicile : Mobile : 06000000000000 Fax : 2113 28841639 E-mail : 0000000000 CIN : 00000 Désigné ci-après par vacataire :</p> <p>D'autre part</p> <p>IL A ETE CONVENU CE QUI SUIT :</p> <p>ARTICLE 1 : Objet de la convention La présente convention a pour objet la réalisation de vacations semestrielles pour le compte de l'École Nationale d'Architecture d'AGADIR.</p> <p>ARTICLE 2 : Consistance des vacations Monsieur YOUNASS est sollicité pour assurer des vacations sous forme de cours, d'enseignements, de formation continue et de travaux divers, ayant trait entre autres, à la recherche.</p> <p>Page 2 sur 4</p>	<p>ARTICLE 3 : Modalités d'exécution des enseignements Le vacataire s'engage à remettre à la Direction le syllabus, annuel ou semestriel, au début de chaque année universitaire et à respecter les règlements de l'ENA-AGADIR. A ce titre, il doit respecter le planning des enseignements, assurer le contrôle continu des connaissances et remettre à la direction des études les notes d'évaluation des élèves après chaque contrôle et suivant les notes circulaires qui seront établies à cet effet. Le vacataire s'engage à remettre aux élèves les copies corrigées des contrôles de connaissance après chaque examen. Le vacataire s'engage à informer la Direction de l'ENA-AGADIR une semaine au moins à l'avance de toute indisponibilité ou absence programmée. Il doit proposer en même temps une date pour le rattrapage destiné à compenser cette absence. Pour toute autre absence imprévue le rattrapage est obligatoire. En outre, le vacataire s'engage à : - Fournir un CV, les syllabus des cours et / ou ateliers qu'il encadre. - Fournir une autorisation d'enseignement ou une patente. - Assurer aux étudiants de coordination et aux conseils de classe. - Encadrer les travaux de recherche, et assister aux jurys de diplôme en tant que représentant de l'administration et/ou enseignant de l'ENA-AGADIR. - Participer à l'élaboration des programmes d'enseignement et de formation. - Contribuer à l'actualisation des contenus et des méthodes d'enseignement. - Participer à l'amélioration de la vie scolaire.</p> <p>ARTICLE 4 : Appropriation des résultats Tous résultats des travaux du vacataire, quelque soit leur forme et particulièrement toute note, rapport, étude et autres documents produits dans le cadre de cette convention, restent la propriété exclusive de l'ENA-AGADIR.</p> <p>ARTICLE 5 : Indemnités L'École Nationale d'Architecture d'AGADIR (ENA-AGADIR) s'engage à régler les prestations fournies dans le cadre de la présente convention. Le montant de la rémunération globale, arrêté conformément aux taux en vigueur, sera calculé au prorata d'heures d'enseignement réellement fournies.</p> <p>ARTICLE 6 : Secret professionnel Le vacataire s'engage pendant la durée de la convention et après expiration à observer une discrétion absolue à l'égard de tout fait, information et document dont il a eu connaissance en</p> <p>Page 3 sur 4</p>	<p>raison de l'exercice de ses fonctions ou à l'occasion de celle-ci, il se déclare à cet égard obligé par la réglementation en vigueur en matière de secret professionnel.</p> <p>ARTICLE 7 : Réalisation Les deux parties peuvent résilier par écrit la présente convention à tout moment avec un mois de préavis, sans que cela puisse ouvrir droit à une quelconque indemnisation au profit du vacataire. L'administration peut prononcer la résiliation sans préavis, en cas de manquement grave aux règlements de l'ENA-AGADIR et/ou de cette convention.</p> <p>ARTICLE 8 : Litige Tous les litiges découlant de l'exécution de la présente convention seront réglés à l'amiable ou à défaut portés devant les tribunaux d'AGADIR.</p> <p>ARTICLE 9 : Validité La présente convention concerne le 2^{ème} semestre de l'année universitaire 2020/2021</p> <p>Date : Le vacataire (Lu et accepté en manuscrit)</p> <p>N.B : Prévoir parapher toutes les pages avant signature de la direction</p> <p>AGADIR, le Présenti par le Directeur de l'ENA-AGADIR</p> <p>Page 4 sur 4</p>
---	---	---

Figure 2 : Exemple de contrat des enseignants

2- Problématique et Objectifs

L'administration de l'ENAA souhaite utiliser des méthodes de nouvelles technologies et de nouvelles fonctionnalités pour faire face aux problèmes de gestion des enseignants et de leur paiement, ainsi que la gestion des cours par les enseignants. C'est dans ce contexte que nous sommes intervenus afin de concevoir une solution informatique d'aide à la gestion des enseignants pour pallier à ce problème.

Dans notre étude nous avons constaté que :

- Le comptable travaille manuellement sur Excel dans l'absence totale de sécurité et de bonne organisation.
- Pour chaque paiement, le comptable vérifie le relevé d'identité bancaire et d'autres informations de l'enseignant dans des archives, donc, plus le nombre des enseignants augmente plus la tâche sera difficile, et par conséquent, les possibilités d'erreur augmentent.
- Au début de chaque semestre, le service comptable doit générer manuellement des contrats pour les enseignants temporaires.

Afin de remédier aux défaillances et aux problèmes mentionnés ci-dessus, nous proposons une solution qui nous a paru adéquate et qui se manifeste par la mise en place d'une application Web.

Parmi les objectifs de cette application on cite :

- Stockage des données concernant chaque enseignant.
- Création d'une interface permettant aux administrateurs de publier les annonces.
- Facilitation de l'accès aux données lors des paiements.
- Génération automatique des contrats.
- Création d'une interface permettant aux enseignants de publier les cours des travaux dirigés et d'autres fichiers qui pourront être téléchargés de la part des étudiants.

3- Description du projet

a. Présentation de l'ENA d'Agadir

L'Ecole Nationale d'Architecture d'Agadir est un établissement public d'enseignement supérieur et de formation de cadres professionnels des secteurs publics et privés. Elle est sous la tutelle du Ministère de l'Urbanisme et de l'Aménagement du Territoire. Sa création fût confiée au Ministère de l'Habitat et de l'Aménagement du Territoire à Rabat, Division de l'Architecture. , qui prévoit l'inauguration de cinq écoles nationales d'architecture à travers les principales régions du Maroc (Fès, Marrakech, Casablanca, Oujda et Agadir).

De sa création récente, suite à un partenariat entre le Ministère de l'Urbanisme et l'Université Ibn Zohr, l'ENA d'Agadir se veut un établissement d'enseignement supérieur de référence dans les métiers liés à l'architecture et l'urbanisme à l'échelle nationale, régionale et internationale.

b. La formation à l'ENA

Le cycle architecte du Diplôme national d'architecte est de douze semestres. L'année académique du Cycle Architecte se compose de 2 semestres comprenant chacun 16 semaines d'enseignement et d'évaluation. Chaque semestre est composé de 6 à 8 modules avec un nombre horaire bien défini de face à face tout au long du semestre.

c. Les formes d'enseignement à l'ENA

L'enseignement est divisé en modules pédagogiques dispensés sous forme d'ateliers, de conférences, de séminaires, de travaux dirigés (TD), de devoirs (TP), et/ou d'autres activités pédagogiques.

4- Conduite du projet

Le cycle de vie d'un logiciel (software life cycle), désigne toutes les étapes du développement d'un logiciel, de sa conception à sa disparition.

Cette division a pour objectif de permettre la définition de jalons intermédiaires permettant de valider le développement logiciel.

Le cycle de vie comprend généralement au minimum les activités suivantes :

- **Définition des objectifs** : consistant à définir la finalité du projet et son inscription dans une stratégie globale.
- **Analyse des besoins et faisabilité** : c'est-à-dire l'expression, le recueil et la formalisation des besoins du demandeur et d'ensemble de contraintes.
- **Conception générale** : il s'agit de l'élaboration des spécifications de l'architecture générale du logiciel.
- **Conception détaillée** : consiste à définir précisément chaque sous-ensemble du logiciel.
- **Codage (implémentation ou programmation)** : la traduction dans un langage de programmation des fonctionnalités définies lors de phases de conception.
- **Tests unitaires** : permettant de vérifier individuellement que chaque sous-ensemble du logiciel est implémenté conformément aux spécifications.
- **Documentation** : visant à produire les informations nécessaires pour l'utilisation du logiciel aux spécifications initiales.
- **Mise en production**
- **Maintenance** : rassemble toutes les actions correctives et évolutives sur les logiciels.

Parmi les cycles de vie les plus utilisés on trouve : le modèle en cascade, le modèle en V, le modèle en spirale et le modèle par incrément. Cependant ces modèles ne permettent pas d'avoir une vision réelle sur le résultat, ils manquent de flexibilité aux modifications des spécifications ou des changements brusques, d'où le recours aux processus modernes et simplifiés, talque les méthodes Agiles ou les Processus Unifiés. Un processus définit une séquence d'étapes qui mènent à obtenir un système logiciel ou à évaluer un système existant. Deux types des processus de développement d'applications web sont possibles : on trouve d'une part les méthodes agiles qui sont actuellement en vogue et d'autre part on peut utiliser un Processus Unifié qui représente un cadre général et très complet de développement.

Concernant les méthodes agiles, elle sont adaptées pour les projets complexes ou de taille moyenne qui durent longtemps et qui nécessitent l'intervention d'une grande équipe de développement qui reste tout le long du projet en interaction directe avec le client pour résoudre les tâches complexes et réagir contre l'évolution permanente du contexte.

La méthodologie de développement que nous avons suivi pour la réalisation de ce projet est basée sur le processus 2TUP (le cycle en Y). Qu'est un processus unifié centré sur les cas d'utilisation et basé sur la modélisation avec le langage UML. Il consiste à fusionner les résultats des deux évolutions fonctionnelle et technique, ce qui conduit à un processus de développement en forme de caractérisé par trois branches comme l'illustre la figure 3 :

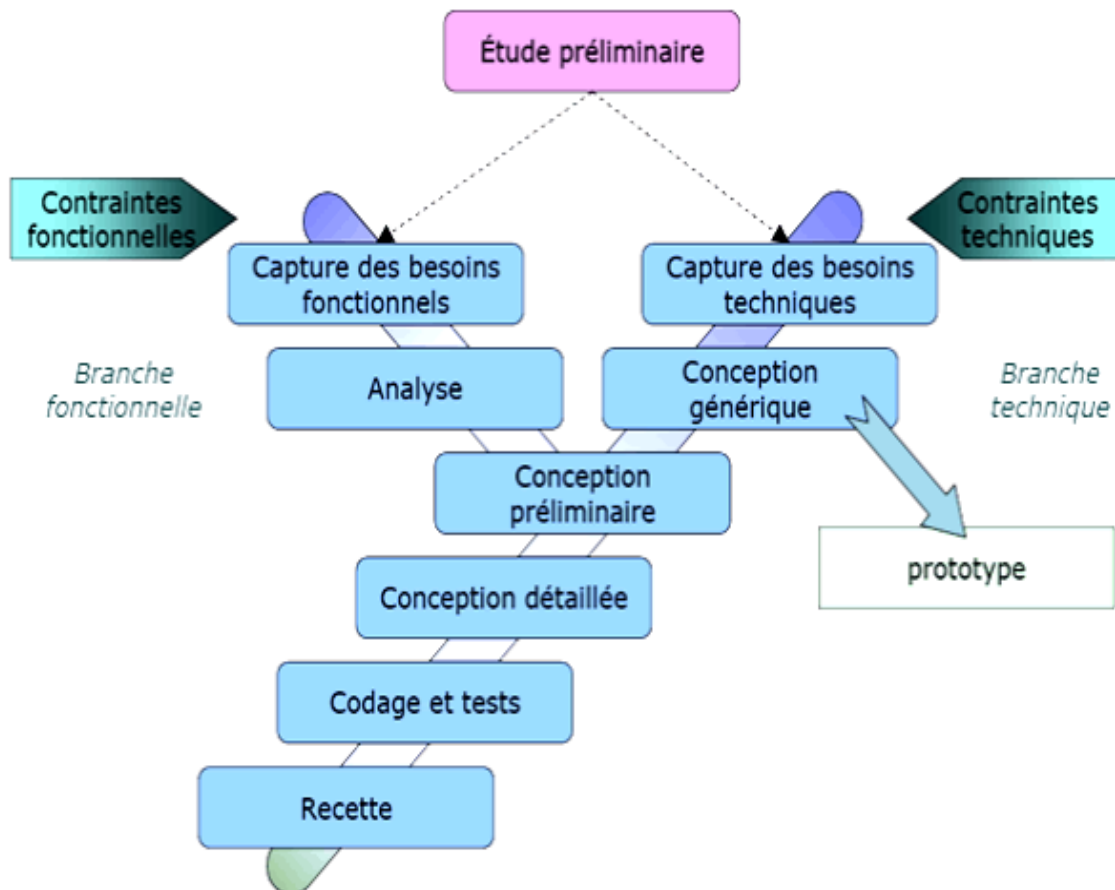



Figure 3: Le processus 2TUP

- ❖ **Branche fonctionnelle:** la capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, le maître d'ouvrage consolide les cahiers des charges et en vérifie la cohérence et l'exhaustivité.
- ❖ **Branche technique :** la capture des besoins techniques, qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés avec la prise en considération des contraintes d'intégration avec l'existant conditionnent généralement des prés requis d'architecture.
- ❖ **Branche réalisation :** comporte la conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.

5- Planification du projet

La planification est parmi les phases d'avant-projet les plus importantes. Elle consiste à déterminer et à ordonnancer les tâches du projet et à estimer leurs charges respectives. Pour planifier notre projet et suivre l'avancement, nous avons utilisé le diagramme de GANTT. Ce diagramme permet de planifier le projet et de rendre le suivi de son avancement plus simple. Il permet aussi de visualiser l'enchaînement et la durée des différentes tâches. La figure 5 représente le diagramme de GANTT des tâches principales que nous avons réalisées.



Nom	Date de début	Date de fin
• Consigne/Sujet du PFE	16/02/2022	16/02/2022
• Reunion de demarrage	21/02/2022	21/02/2022
☐ • Analyse	22/02/2022	21/03/2022
• Etude fonctionnelle	22/02/2022	28/02/2022
• Etude de l'architecture logiciel	01/03/2022	07/03/2022
• Etude des Framework et technique	08/03/2022	21/03/2022
☐ • Conception	22/03/2022	07/04/2022
• Diagramme MCD et MLD	22/03/2022	29/03/2022
• Diagramme de séquence	28/03/2022	31/03/2022
• Diagramme de cas d'utilisation	01/04/2022	07/04/2022
☐ • Developpement	15/03/2022	19/05/2022
• Creation de la base de données	15/03/2022	23/03/2022
• Développement des interfaces	24/03/2022	20/04/2022
• Développement des fonctionnalités	25/03/2022	19/05/2022
☐ • Redaction du PFE	18/04/2022	13/05/2022
• Redaction du rapport	18/04/2022	13/05/2022
☐ • Tests et validation	20/05/2022	02/06/2022
• Tests et validation fonctionnelle	20/05/2022	02/06/2022

Figure 4 : tableau du planification général suivis

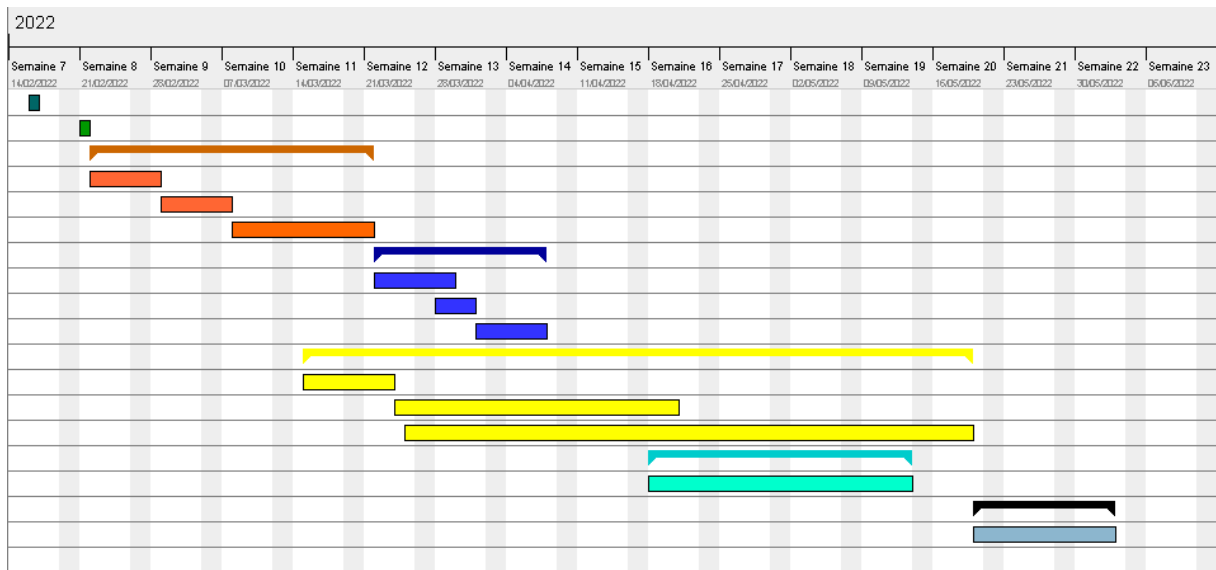


Figure 5 : diagramme de Gantt du planification général suivis

CHAPITRE 2 : ANALYSE ET CONCEPTION

Vu l'importance de l'étape de la conception dans le développement d'une application. Nous nous intéressons dans ce second chapitre à l'analyse et la conception de notre application développée.

1- Spécification des besoins :

L'application que nous avons développée regroupe les fonctionnalités suivantes :

a. Les besoins fonctionnels

i. Pour l'administrateur :

S'authentifier :

- L'administrateur peut s'authentifier à travers un email et un mot de passe. Le système vérifie les informations saisies pour valider l'accès.

Gestion des enseignants :

- L'administrateur peut ajouter, visualiser, modifier et supprimer des enseignants.
- Les données des enseignants supprimées sont sauvegardées dans la base de données pour que l'administrateur puisse les restaurer.
- L'administrateur peut trouver le plus rapidement possible un enseignant soit par son identifiant soit par son diplôme

Gestion des matières :

- L'administrateur peut ajouter, visualiser, modifier et supprimer des matières.

Gestion des annonces :

- L'administrateur peut ajouter, visualiser, modifier et supprimer des annonces.

Ajout des utilisateurs :

- L'administrateur peut ajouter des nouveaux utilisateurs avec la saisie des informations nécessaires.

Gestion du profil :

- L'administrateur peut modifier leurs informations personnelles.

ii. Pour l'enseignant :

S'authentifier :

- L'enseignant peut s'authentifier à travers un email et un mot de passe. Le système vérifie les informations saisies pour valider l'accès.

Gestion des cours :

- L'enseignant peut ajouter, modifier et supprimer des cours.
- Chaque enseignant a le droit de visualiser seulement ses cours.

Gestion des annonces :

- L'enseignant peut visualiser les annonces.

Gestion du profile:

- L'enseignant peut modifier ses informations personnelles.

iii. Pour l'étudiant :

S'authentifier :

- L'étudiant peut s'authentifier à travers un email et un mot de passe. Le système vérifie les informations saisies pour valider l'accès.

Recherche des cours :

- L'étudiant peut trouver le plus rapidement possible les cours par l'identifiant du semestre ou par le nom de la matière correspondant.

Gestion des annonces :

- L'étudiant peut visualiser les annonces.

Gestion du profile :

- L'étudiant peut modifier leurs informations personnelles.

b. Les besoins non fonctionnels

- **La performance :** l'application doit être performante, ce qui signifie que le système doit réagir dans un temps spécifié, indépendamment de l'action de l'utilisateur.
- **La sécurité :** l'application doit être hautement sécurisée, ce qui signifie que tous les services doivent être protégés contre les diverses menaces exploitant les vulnérabilités associées aux codes d'application.

- **La fiabilité** : l'application doit fonctionner de façon satisfaisante et cohérente sans erreurs, c'est-à-dire les données peuvent être récupérées en cas de panne de serveur.
- **L'extensibilité** : l'application pourra y avoir la possibilité d'ajouter ou modifier de nouvelles fonctionnalités.

2- Conception :

a. Méthode et logiciel de conception :

- **UML**



Est un langage standard, universel de modélisation graphique et textuelle, leur objectif est d'unifier les méthodes d'analyse et de conception des systèmes.

Analyse : recensement et définition des besoins.

Conception : définition d'une architecture « abstraite ».

Dans notre cas nous avons utilisé cette méthode pour avoir une vision globale de notre projet sous forme des diagrammes. Cette étape est très importante pour le développement des applications web.

- **Visual paradigm online**



Est un outil de création des diagrammes UML pour les applications web.

Dans notre cas, nous avons utilisé ce logiciel pour générer le diagramme de cas d'utilisation et de séquence.

- **powerAMC**



(SAP POWERDESIGNER) est un outil de modélisation de traitements et processus informatiques. Il permet de créer des plans pour visualiser les conséquences des changements technologiques avant l'implémentation réelle. Il permet de générer le modèle physique (MPD) et logique (MLD) correspondant au MCD. Dans notre cas, nous avons exploité ce logiciel pour générer MCD et classe.

b. Diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation est le premier diagramme à développer lors d'une modélisation UML, car il décrit le système du point de vue de l'utilisation.

Ce diagramme permet de classer les acteurs et de structurer les objectifs du système ainsi que d'identifier ses fonctionnalités.

Le diagramme de cas d'utilisation se compose de trois éléments principaux :

➤ **Un Acteur**

Un acteur est une entité qui définit le rôle joué par un utilisateur ou par un système qui interagit avec le système modélisé. Les acteurs apparaissent dans les diagrammes de cas d'utilisation. Les acteurs dans notre application sont :

Administrateur : C'est l'utilisateur qui possède le privilège du plus haut niveau. Cet acteur est capable de manipuler toutes les fonctionnalités de la partie de gestion des enseignants proposée par l'application notamment l'ajout des enseignants, afficher, modifier et supprimer les informations d'un enseignant.

Enseignant : Les enseignants qui manipulent quelques fonctionnalités notamment la consultation de profil et la mise à jour de leurs informations, l'ajout d'un cours, travaux pratiques, travaux dirigés ou tout autre document lié à la matière qu'il enseigne.

Etudiant : Les étudiants qui sont capables de télécharger les documents partagés par les enseignants.

➤ **Un cas d'utilisations :**

Il s'agit d'une image d'une fonctionnalité attendue, déclenchée en réponse à la stimulation d'un acteur.

➤ **Les relations :**

Trois types de relations sont pris en charge par la norme UML et sont graphiquement représentées par des types particuliers de ces les mêmes conditions d'exécution que le cas issu.

Une relation simple entre un acteur et une utilisation est un trait simple.

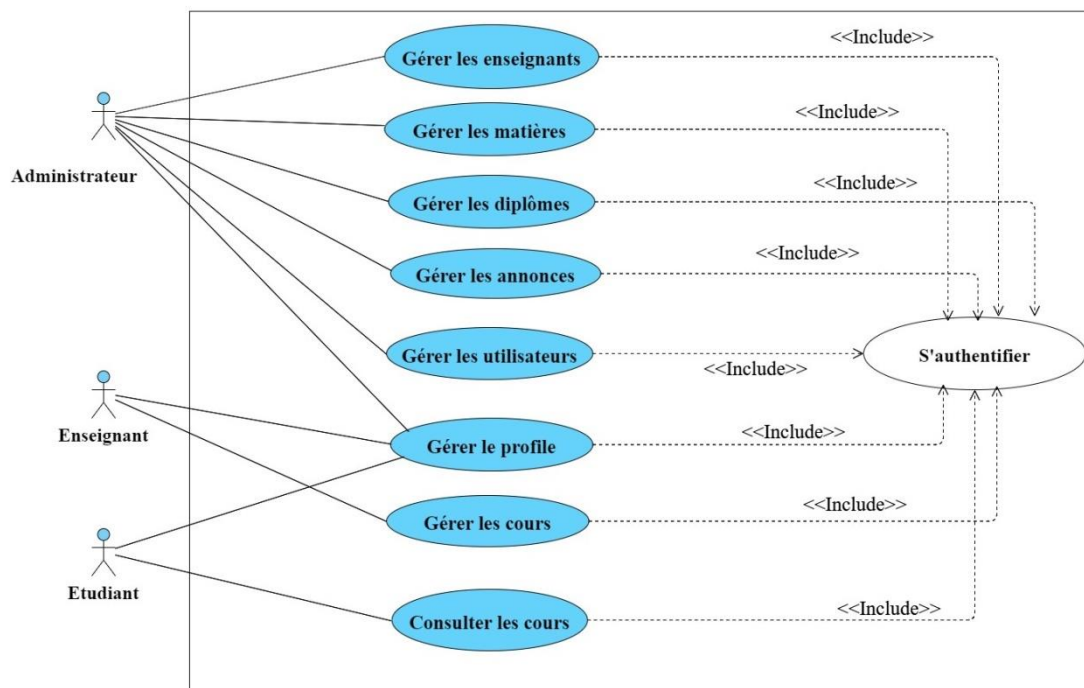


Figure 6 : Diagramme de cas d'utilisation globale

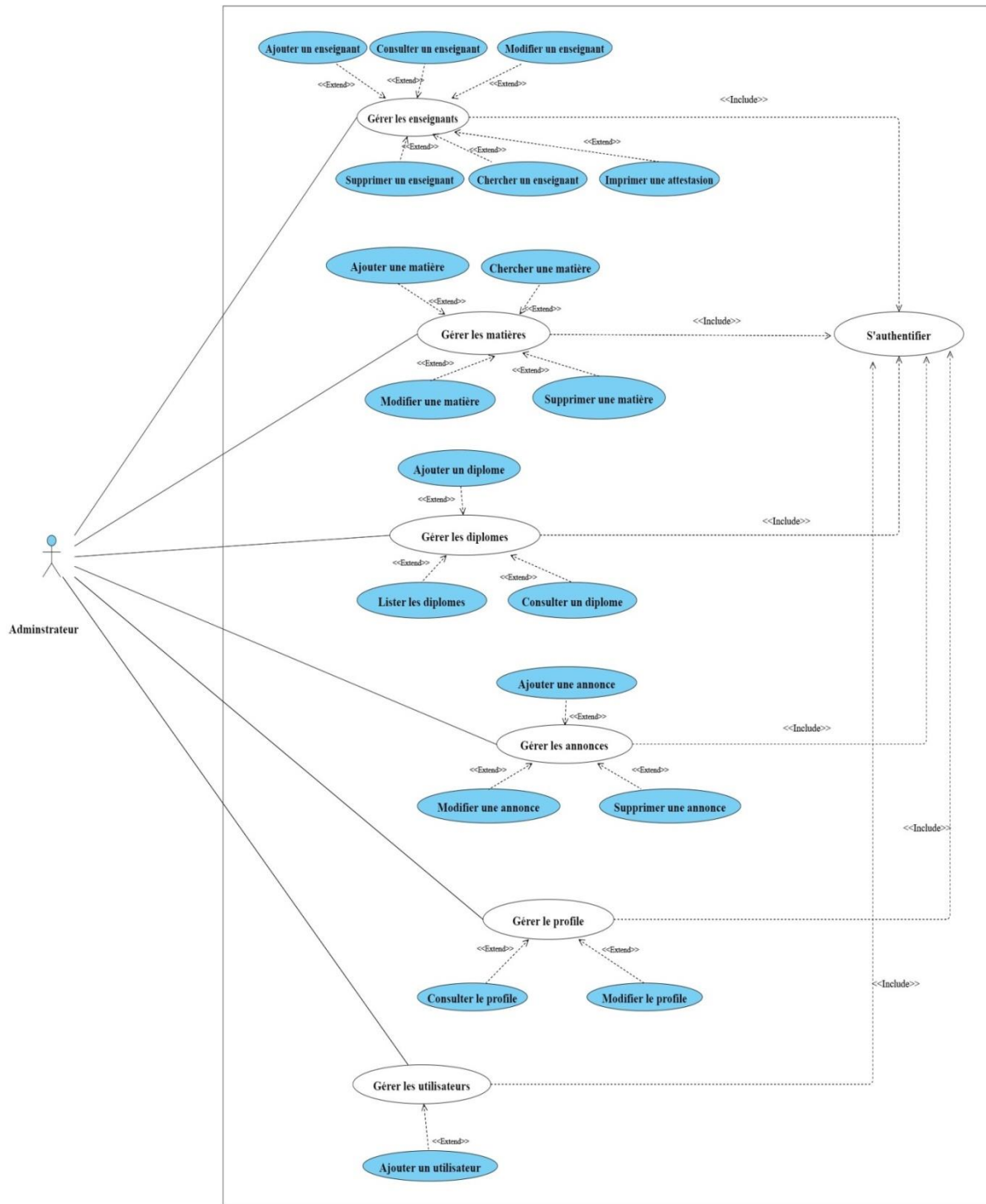


Figure 7 : Diagramme de cas d'utilisation de l'administrateur

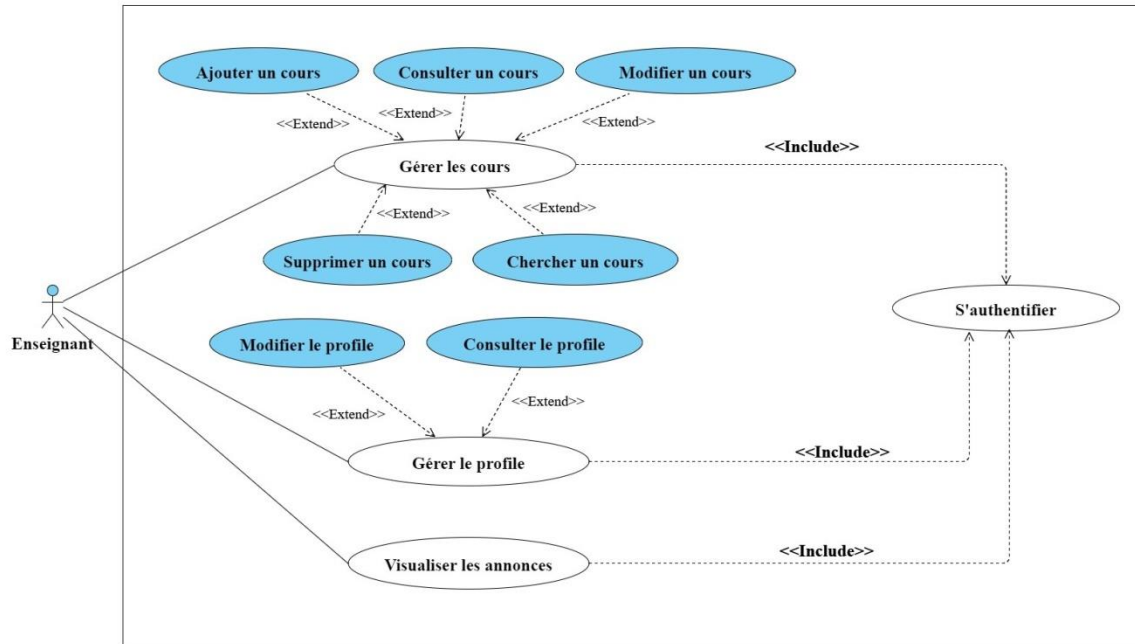


Figure 8 : Diagramme de cas d'utilisation de l'enseignant

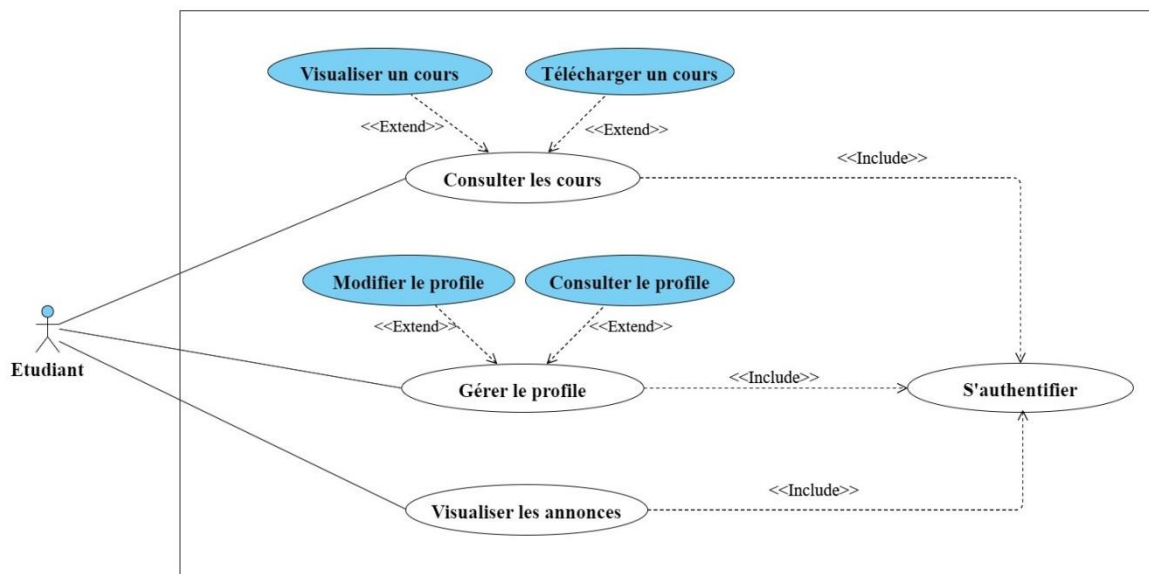


Figure 9 : Diagramme de cas d'utilisation de l'étudiant

- **description du cas d'utilisations**

Titre	S'inscrire et S'authentifier
Acteurs	<ul style="list-style-type: none"> ▪ Administrateur, Enseignant, Etudiant.
Description	<ul style="list-style-type: none"> ▪ Enregistrer l'utilisateur pour pouvoir accéder à l'application via le nom d'utilisateur et son mot de passe.
Préconditions	<ul style="list-style-type: none"> ▪ Serveur accessible. ▪ Navigateur en bon état de fonctionnement.
Post-conditions	<ul style="list-style-type: none"> ▪ Utilisateur inscrit et authentifié par un nom d'utilisateur et un mot de passe.
Scénarios	<ul style="list-style-type: none"> ▪ Saisir les données => Vérification => Accès à l'application. ▪ Saisir les données => Vérification => Affichage du message d'erreur.

Tableau 1 : la description du cas d'utilisation "S'inscrire et S'authentifier"

Titre	Ajouter un enseignant
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ Ajout d'un nouvel enseignant
Scénarios	<ul style="list-style-type: none"> ▪ Demande de formulaire => Traitement => Envoi de formulaire => Saisie des données => Vérification => Ajout avec succès et envoi du message de validation. ▪ Demande de formulaire => Traitement => Envoi de formulaire => Saisie des données => Vérification => Echech d'ajout et envoi du message d'erreur.

Tableau 2 : la description du cas d'utilisation "Ajouter un enseignant"

Titre	Lister les enseignants
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ La liste des enseignants est affichée sur l'écran
Scénarios	<ul style="list-style-type: none"> ▪ Affichage de la liste avec succès ▪ La liste est vide

Tableau 3 : la description du cas d'utilisation "Lister les enseignants"

Titre	Modifier un enseignant
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable ▪ Existence d'enseignant
Post-conditions	<ul style="list-style-type: none"> ▪ Mises à jour des informations
Scénarios	<ul style="list-style-type: none"> ▪ Demande de formulaire à modifier => Traitement => Envoi de formulaire => Modification des données => Vérification => Modification des données avec succès et envoi de message de validation. ▪ Demande de formulaire à modifier => Traitement => Envoi de formulaire => Modification des données => Vérification => Echech de modification et envoi de message d'erreur.

Tableau 4 : la description du cas d'utilisation" Modifier un enseignant"

Titre	Supprimer un enseignant
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable ▪ Existence d'enseignant
Post-conditions	<ul style="list-style-type: none"> ▪ Suppression d'enseignant
Scénarios	<ul style="list-style-type: none"> ▪ Demande la suppression d'un enseignant => Vérification => Suppression validée. ▪ Demande la suppression d'un enseignant => Vérification => Enseignant inscrit dans un cours => Echec de la suppression.

Tableau 5 : la description du cas d'utilisation "Supprimer un enseignant"

Titre	Chercher un enseignant
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable ▪ Formulaire de recherche disponible
Post-conditions	<ul style="list-style-type: none"> ▪ Les informations d'un enseignant affichent sur l'écran
Scénarios	<ul style="list-style-type: none"> ▪ Administrateur fait une recherche avec CIN ou diplôme => Traitement => Affichage la liste des enseignants. ▪ Administrateur fait une recherche avec CIN ou diplôme=> Traitement => Aucune résultat.

Tableau 6 : la description du cas d'utilisation" Chercher un enseignant"

Titre	Gérer les annonces
Acteur	<ul style="list-style-type: none"> ▪ Administrateur
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ La gestion des annonces
Scénarios	<ul style="list-style-type: none"> ▪ L'administrateur visualise les annonces qui ont été déjà publiés ▪ L'administrateur demande le formulaire d'ajout, de modification ou la suppression d'une annonce. ▪ Envoie de formulaire s'il y a une demande d'ajout ou de modification. ▪ Remplissage du formulaire. ▪ La vérification par le système puis la validation ou l'échec.

Tableau 7 : la description du cas d'utilisation" Gérer les annonces"

Titre	Gérer les cours
Acteur	<ul style="list-style-type: none"> ▪ Enseignant
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ La gestion des cours
Scénarios	<ul style="list-style-type: none"> ▪ L'enseignant visualise les cours qui sont déjà publiés ▪ L'enseignant demande le formulaire d'ajout, de modification ou la suppression d'un cours. ▪ Envoie de formulaire s'il y a une demande d'ajout ou de modification. ▪ Remplissage du formulaire. ▪ La vérification par le système puis la validation ou l'échec.

Tableau 8 : la description du cas d'utilisation" Gérer les cours"

Titre	Gérer le profile
Acteur	<ul style="list-style-type: none"> ▪ Administrateur, enseignant, étudiant
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ La gestion du profile
Scénarios	<ul style="list-style-type: none"> ▪ L'utilisateur peut visualiser leurs informations. ▪ L'utilisateur peut demander le formulaire de modification. ▪ Envoie de formulaire par le système. ▪ Remplissage du formulaire. ▪ La vérification par le système puis la validation ou l'échec.

Tableau 9 : la description du cas d'utilisation" Gérer le profile"

Titre	Recherche des cours
Acteur	<ul style="list-style-type: none"> ▪ Etudiant
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable ▪ Existence du cours
Post-conditions	<ul style="list-style-type: none"> ▪ La liste des cours affiche sur l'écran
Scénarios	<ul style="list-style-type: none"> ▪ Etudiant fait une recherche par le semestre et la matière correspondant à ce cours. ▪ La vérification par le système. ▪ L'affichage des cours sinon la liste vide.

Tableau 10 : la description du cas d'utilisation" Rechercher des cours"

Titre	Visualise les annonces
Acteur	<ul style="list-style-type: none"> ▪ Enseignant, Etudiant
Préconditions	<ul style="list-style-type: none"> ▪ Authentification préalable
Post-conditions	<ul style="list-style-type: none"> ▪ Affichage de la liste des annonces
Scénarios	<ul style="list-style-type: none"> ▪ visualise les annonces qui sont publiés par l'administrateur.

Tableau 11 : la description du cas d'utilisation" Visualiser les annonces"

c. Diagramme de séquence :

Le diagramme de séquence est une illustration des cas d'utilisation, il décrit l'ordre des interactions entre les objets qui composent le système pour accomplir une tâche.

La représentation se concentrant sur la séquence des interactions d'un point de vue temporel.

La modélisation de ces interactions l'indentification des scénarios possible des différents cas d'utilisation.

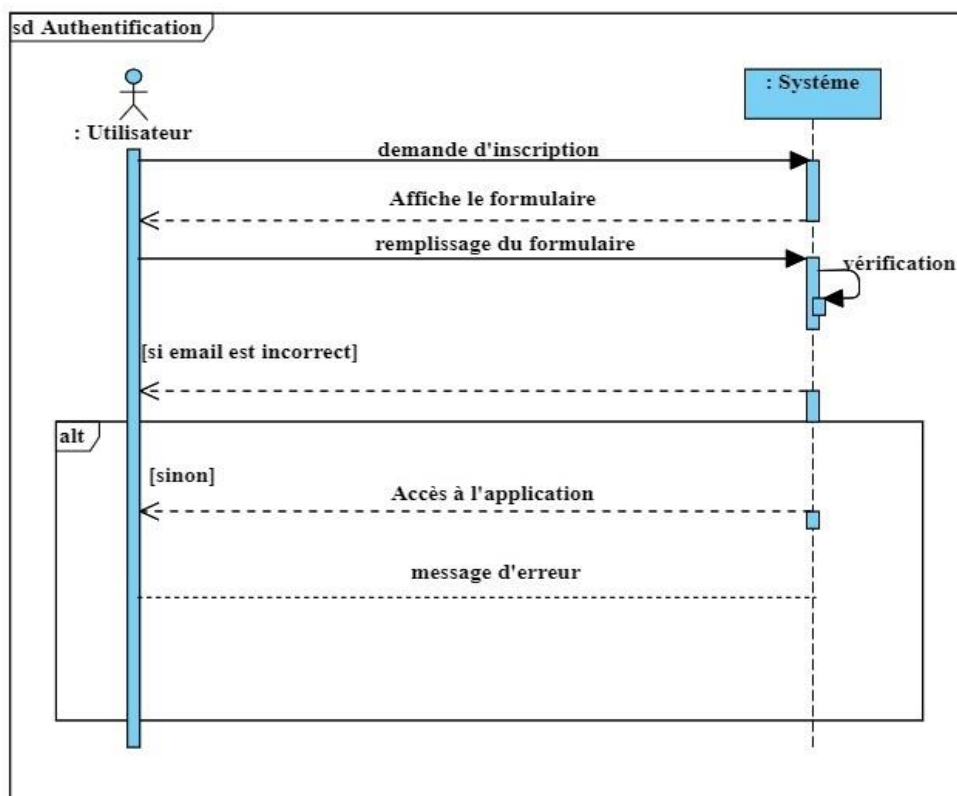


Figure 10 : Diagramme de séquence d'authentification

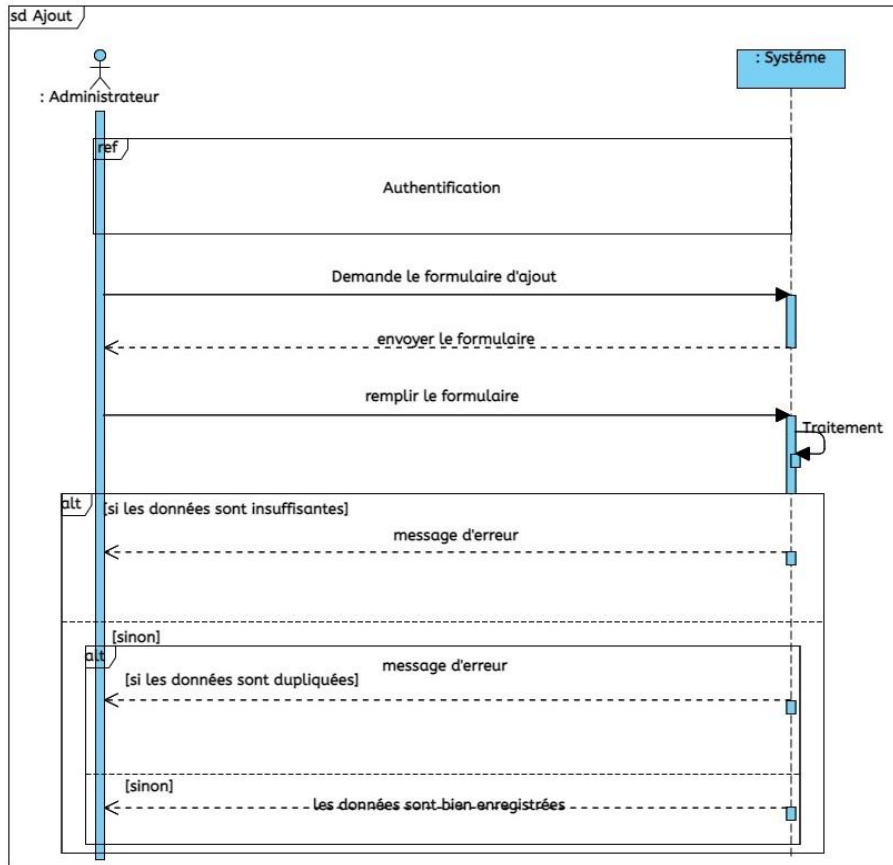


Figure 11 : Diagramme de séquence d'ajout

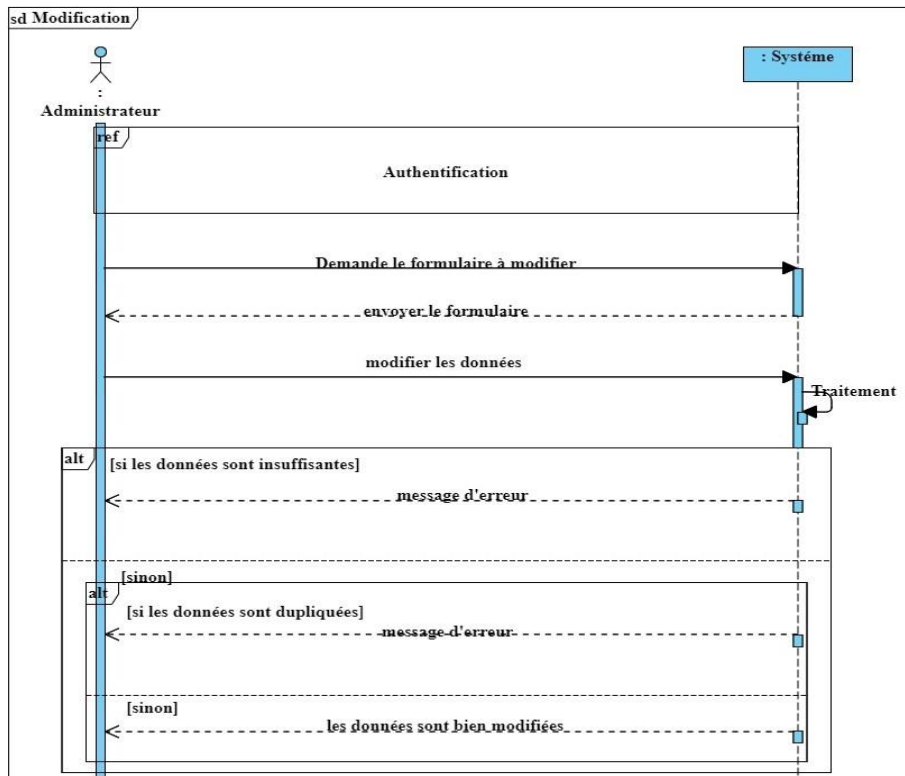


Figure 12 : Diagramme de séquence de modification

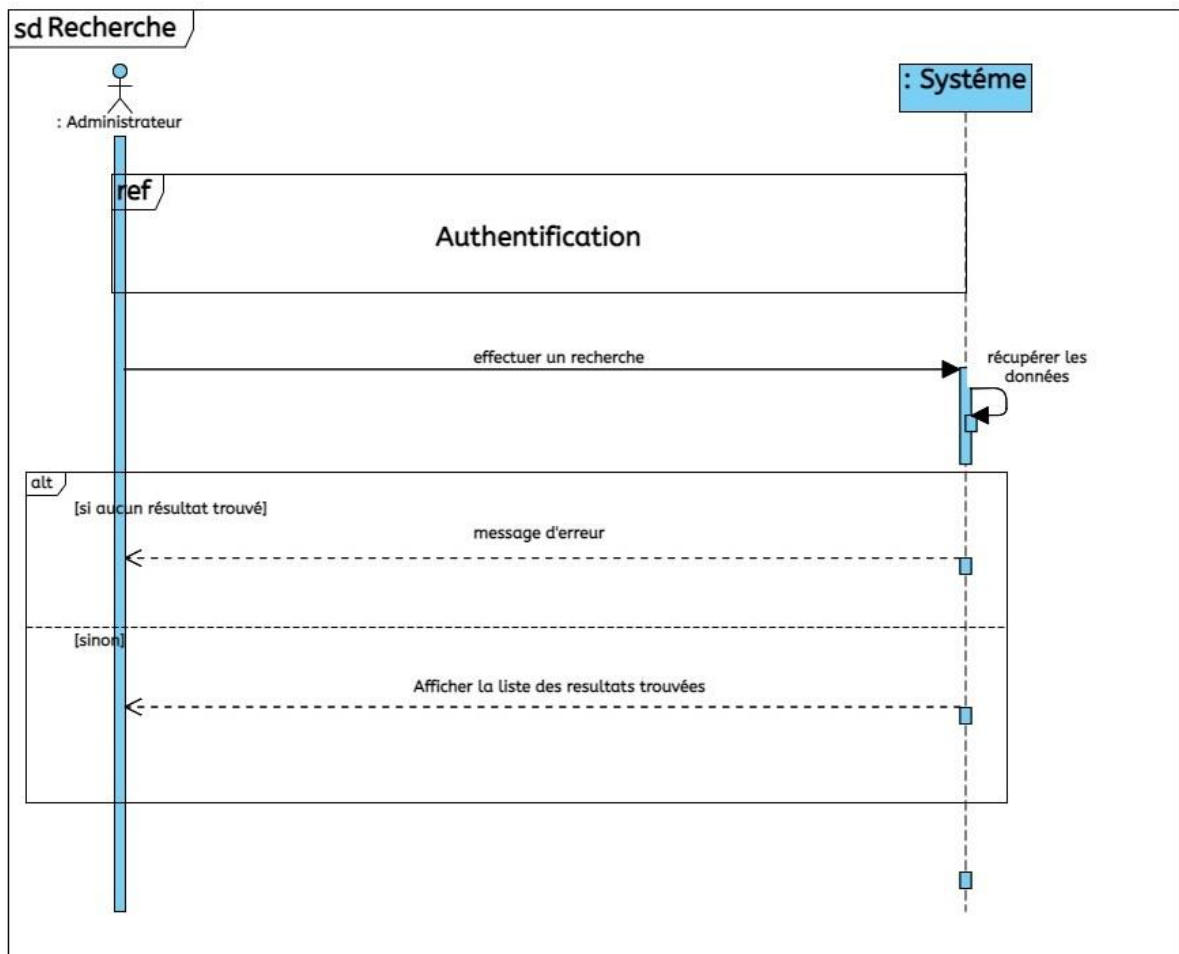


Figure 13 : Diagramme de séquence de recherche

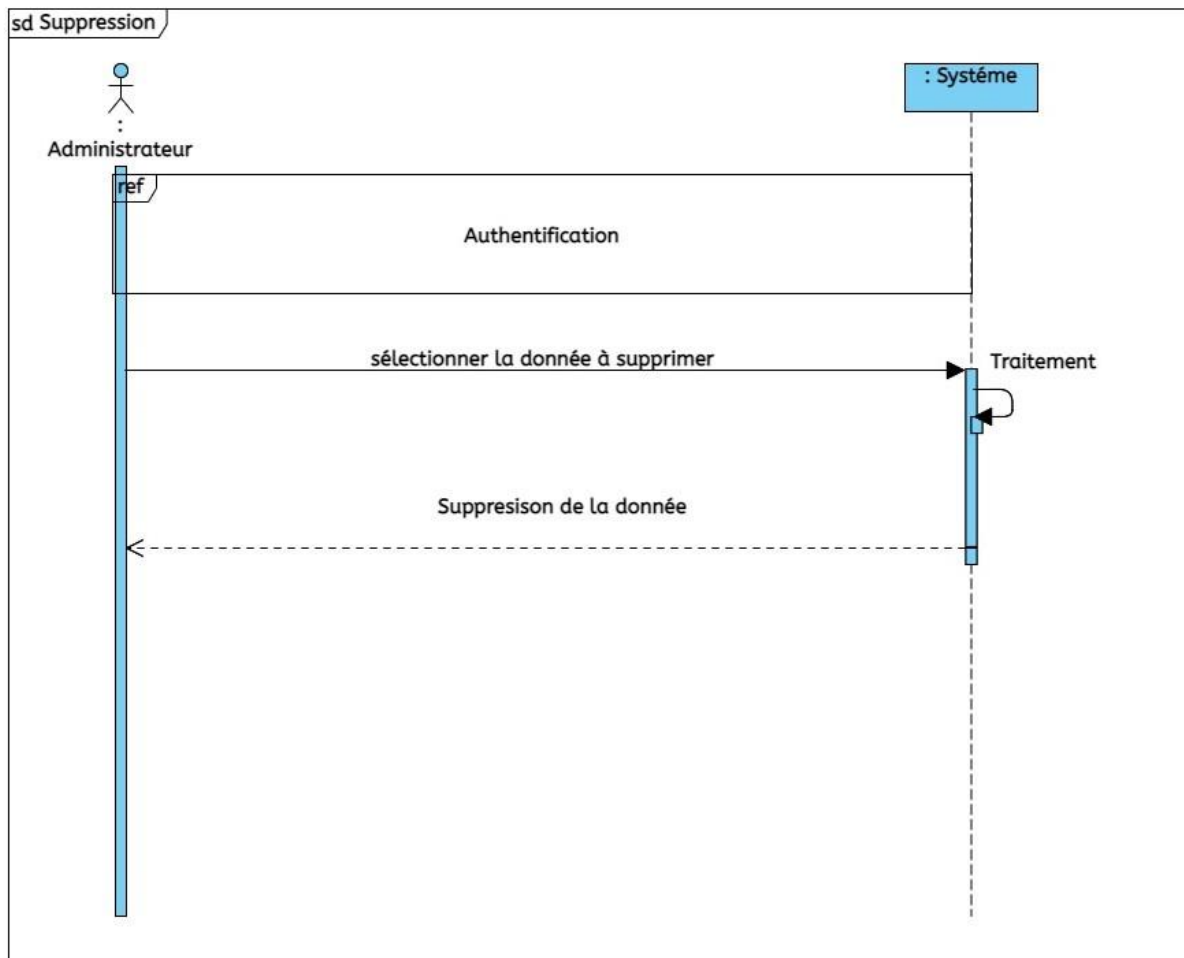


Figure 14 : Diagramme de séquence de suppression

d. Diagramme MCD et classe

i. Modèle conceptuel de données :

Le modèle conceptuel de données MCD permet de décrire la façon formelle des données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données facilement compréhensibles, permettant de décrire le système d'information à l'aide des entités.

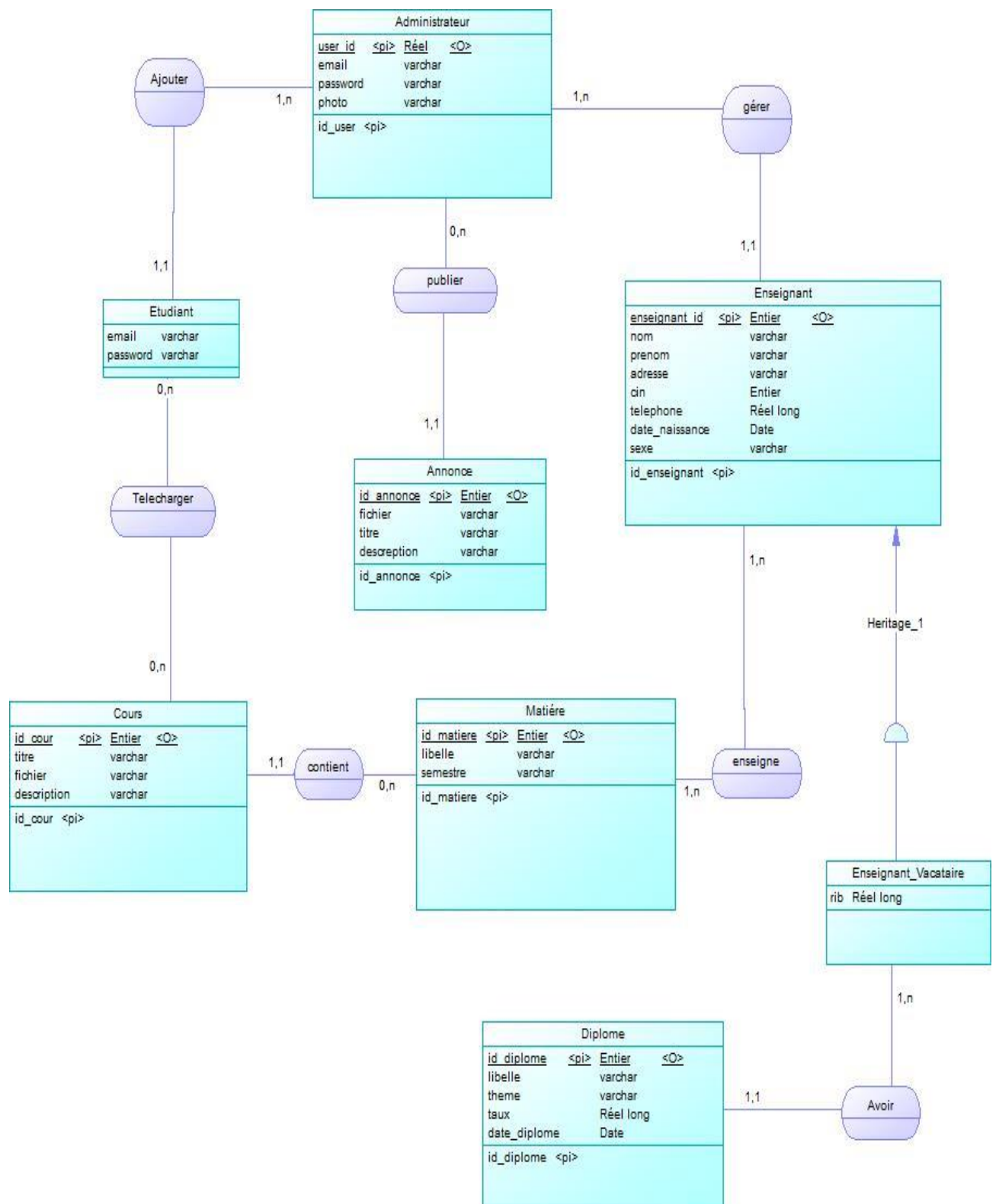


Figure 15 : Diagramme MCD

ii. Diagramme de classe

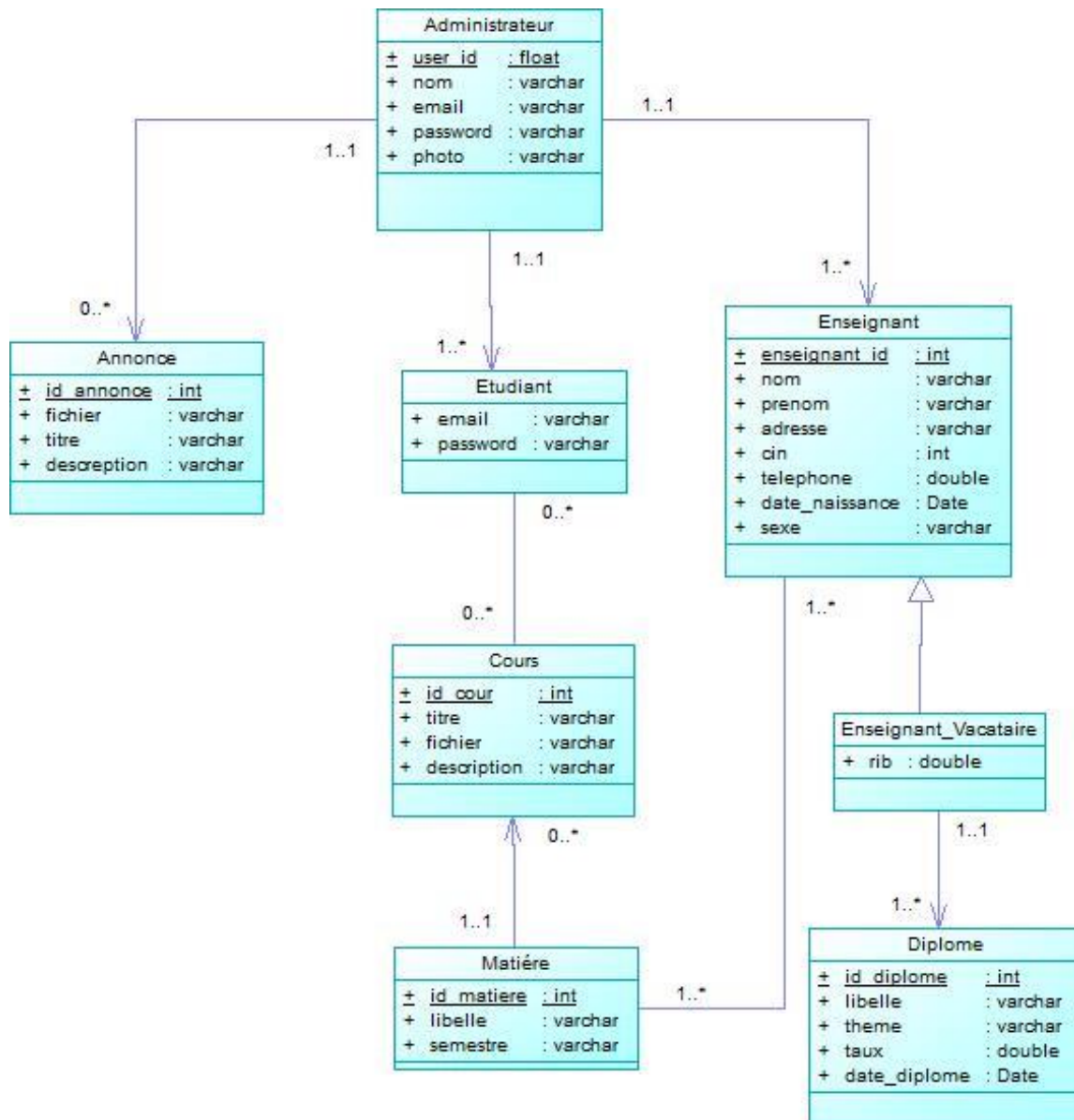


Figure 16 : Diagramme de classe

CHAPITRE 3 : ETUDE TECHNIQUE ET ENVIRONNEMENT

1- Capture des besoins techniques

Ce chapitre présente les différents outils et matériaux que nous avons utilisé lors de la réalisation de notre application web.

La capture des besoins techniques collationne toutes les contraintes qui concernent les composants du projet et les choix dimensionnant la conception du système, à savoir les langages de programmation, le Framework et les environnements du travail, ainsi méthode et architecture du développement.

2- Architecture adoptée

Architecture MVC :

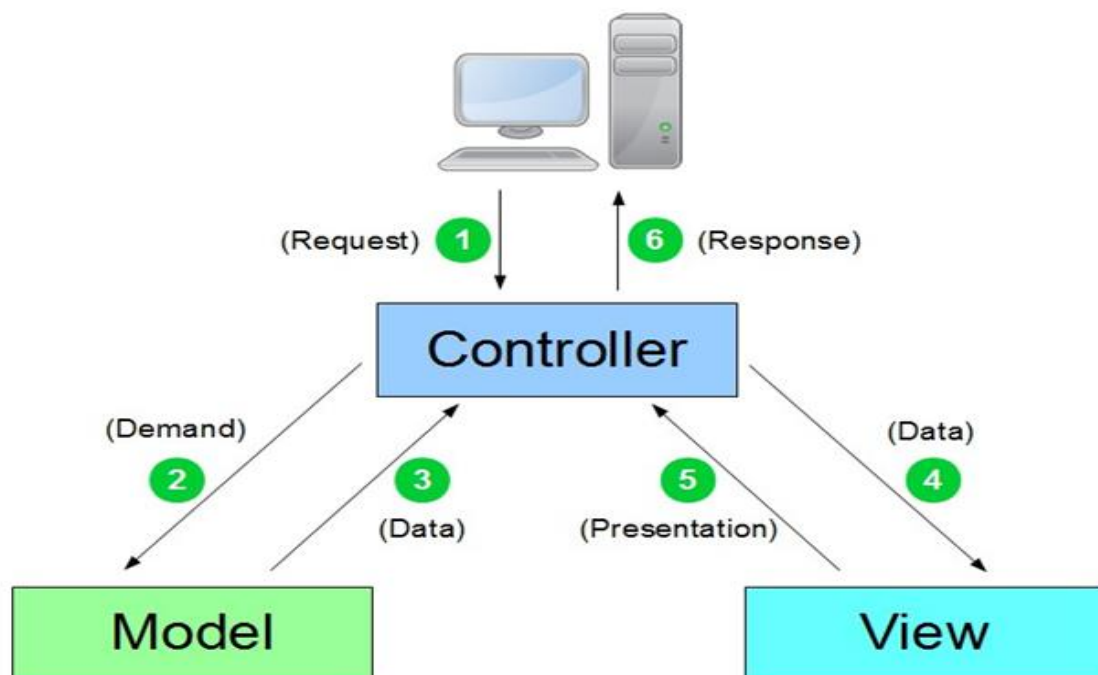


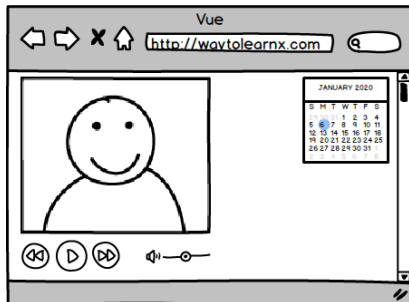
Figure17 : L'architecture MVC

Pour obtenir une application à la fois souple et solide nous avons adopté cette manière d'organisation des interfaces graphiques. Son intérêt est la séparation des données (modèle),

d'affichage (vue) et des actions (contrôleur), qui représentent les principaux composants logiques d'une application, chacun de ces composants est construit pour gérer les aspects de développement spécifiques.

L'architecture module-vue-contrôleur (figure ci-dessus) est une architecture standard la plus fréquemment utilisée dans le développement des projets extensibles et évolutifs.

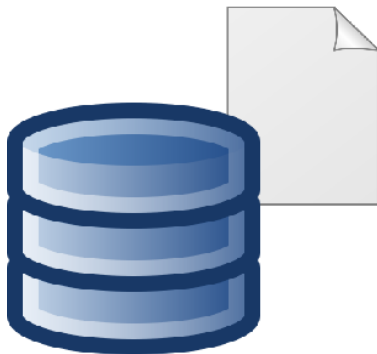
- **La vue :**



La vue est une interface homme machine (IHM), par exemple l'affichage d'une fenêtre, des boutons et des textes, on y trouve essentiellement le code html.

C'est le moyen qui s'intéresse aux interactions avec l'utilisateur, par l'affichage des données traitées par le modèle.

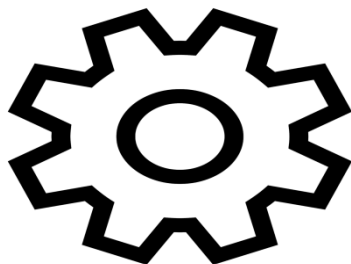
- **Modèle :**



Le modèle représente le cœur de l'application, il contient les données, les algorithmes et la configuration du programme.

Il s'intéresse aux interactions effectuées avec la base de données. Les résultats renvoyés par le modèle ne s'occupent pas de la présentation, le modèle ne contient aucun lien direct vers la vue.

- **Le contrôleur :**



Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle.

Il s'intéresse à la récupération et l'analyse des requêtes,

Par la demande des données au modèle pour les analyser puis prendre la décision et renvoyer le texte à afficher dans la vue.

Donc on peut dire que le contrôleur joue le rôle de l'intermédiaire entre la vue et le modèle.

3- Choix des langages

- **PHP**



HyperText Preprocessor est un langage de programmation ou langage de scripts généraliste et open source. Utilisé principalement pour produire des pages Web dynamiques grâce à un serveur locale HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété. Il est un langage impératif orienté objet. Il dérive du C et du Perl dont il reprend la syntaxe. Il est extensible grâce à de nombreux modules et son code source est ouvert.

- **HTML:**



Le HyperText Markup Language, généralement abrégé HTML ou dans sa dernière version HTML5, est le langage de balisage conçu pour représenter les pages web. HTML permet aussi de structurer sémantiquement et de mettre en forme le contenu des pages d'inclure des ressources multimédias dont des images, des formulaires.

- **CSS**



Cascading Style Sheets est un langage qui permet de gérer la présentation des documents HTML dans les pages web.

CSS concerne tous ce qui est en relation avec la mise en page et l'aspect décoratif « le style », tout en se basant sur les feuilles de style en cascade.

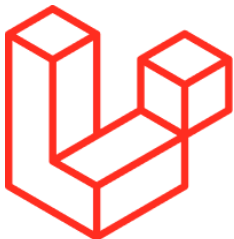
- **JavaScript**



JavaScript est un langage de programmation de scripts son rôle est de travailler l'ergonomie et l'interactivité des pages web pour apporter une meilleur expérience utilisateur.

4- Frameworks utilisés

- **Laravel**



Laravel est un Framework open-source écrit en PHP, créé par Tylor Otwel, respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orienté objet.

Ce Framework regroupe tous les meilleures bibliothèques utiles pour créer un site web.

- **Bootstrap**



Bootstrap est un Framework open-source développé par l'équipe du réseau social Twitter. Il génère les outils nécessaires pour la création du design (graphisme, animation et interactions avec la page dans le navigateur) de sites et applications web. C'est un ensemble qui contient des codes HTML et le CSS qui sont basé principalement sur des formulaires, cartes, boutons et navigation est dès partie JavaScript qui est optionnelle. On appelle ce type de Framework un « Front-End Framework ».

5- Environnements matériels, logiciels, outils

a. Environnements matériels

Durant le développement de notre application web nous avons recouru à 2 ordinateurs portables qui possèdent les caractéristiques suivantes (tableaux 12 et 13) :

Appareil	1
Type	PC portable
Marque	HP
Processeur	Intel(R) Core(TM) i5-5300U
Ram	8 Go
Disque Dur	256 Go
Système d'exploitation	Windows 10

Tableau 12 : les propriétés de l'appareil 1

Appareil	2
Type	PC Portable
Marque	HP
Processeur	Intel i5
Ram	8 GO
Disque Dur	256GO
Système d'exploitation	Windows 10

Tableau 13 : les propriétés de l'appareil 2

b. Environnements Logiciel

- **VS Code**



Visual Studio Code est un éditeur de code open-source développé par Microsoft. Supportant un très grand nombre de langages grâce à des extensions. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code et la refactorisation du code.

- **Google Chrome**



Chrome est un navigateur web propriétaire développé par Google depuis 2008 dans le cadre du projet libre Chromium.

Il fonctionne sous Windows, Mac, Linux, Android et iOS, Il permet le fonctionnement et l'exécution de tout code écrit par des langages attachés au web comme html et CSS ainsi JavaScript et PHP.

- **XAMPP**



Le nom XAMPP signifie Apache, MySQL, PHP et Perl tandis que la lettre « X » est conçue comme un logiciel qui peut fonctionner sur quatre OS principaux comme Windows, Mac OS, Linux et Solaris. Ce terme est souvent appelé multiplateforme (logiciel multi OS). C'est un programme qui comporte un ensemble des logiciels il permet de mettre en place un serveur Web local et un serveur FTP.

- **Gantt Project**



Gantt est un outil utilisé dans la planification et la gestion de projets qui permet de visualiser le temps des différentes tâches qui composent un projet. C'est un programme gratuit et libre qui permet de créer des diagrammes de Gantt et des réseaux PERT, en fonction du calcul du chemin critique.

- **MySQL**



MySQL est un Logiciel en système de gestion de bases de données relationnelles (SGBDR) permettant de manipuler des instructions adressées à la base de données sous forme de requêtes SQL. Il est distribué sous une double licence GPL et propriétaire.

- **Mailtrap**



Mailtrap est un service qui fournit aux développeurs une solution pour gérer les e-mails envoyés par les applications dans un environnement de test.

CHAPITRE 4 : REALISATIONS, INTERFACES, TESTS

Dans ce chapitre nous présentons les scénarios d'exécution de notre application d'aide à la gestion des enseignants. Les interfaces implémentées se divisent initialement en 3 parties (administrateur, enseignant et étudiant).

- **Interface d'authentification**

Cette interface (figure 18) permet à l'utilisateur de s'identifier. Dès la connexion, la page d'authentification s'affiche. Elle est composée de deux champs, 'username' et 'Password', l'utilisateur accède à l'interface dédiée selon son rôle administrateur, enseignant ou étudiant.

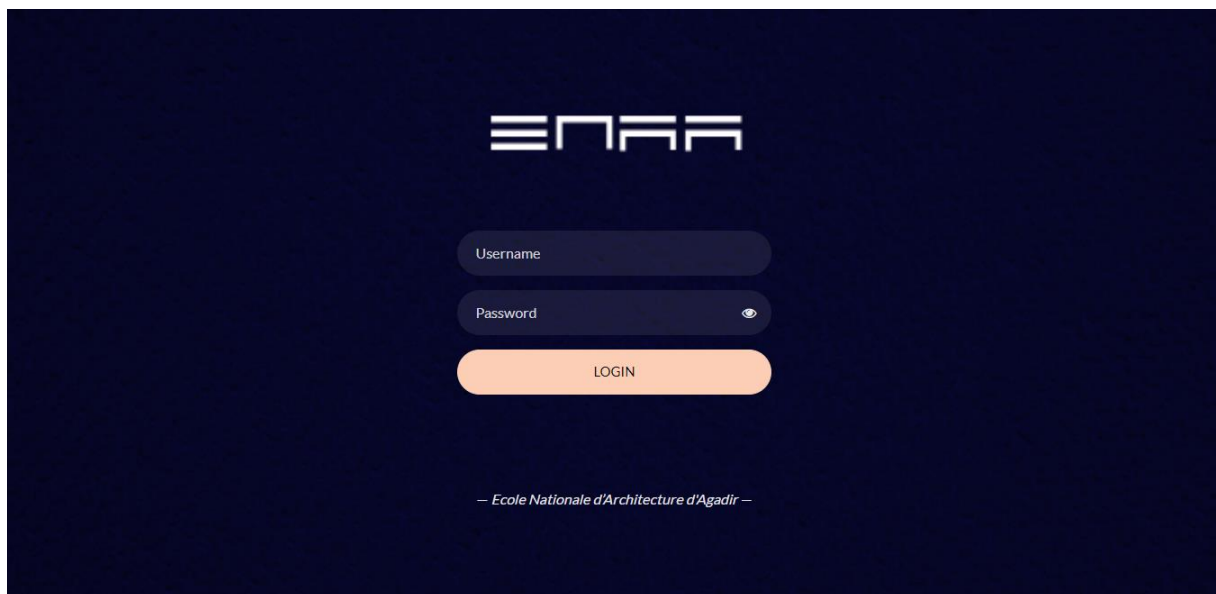


Figure 18 : interface d'authentification

- **Partie Administrateur :**

La figure ci-dessous représente la page d'accueil d'administrateur, il contient le menu général de l'application.

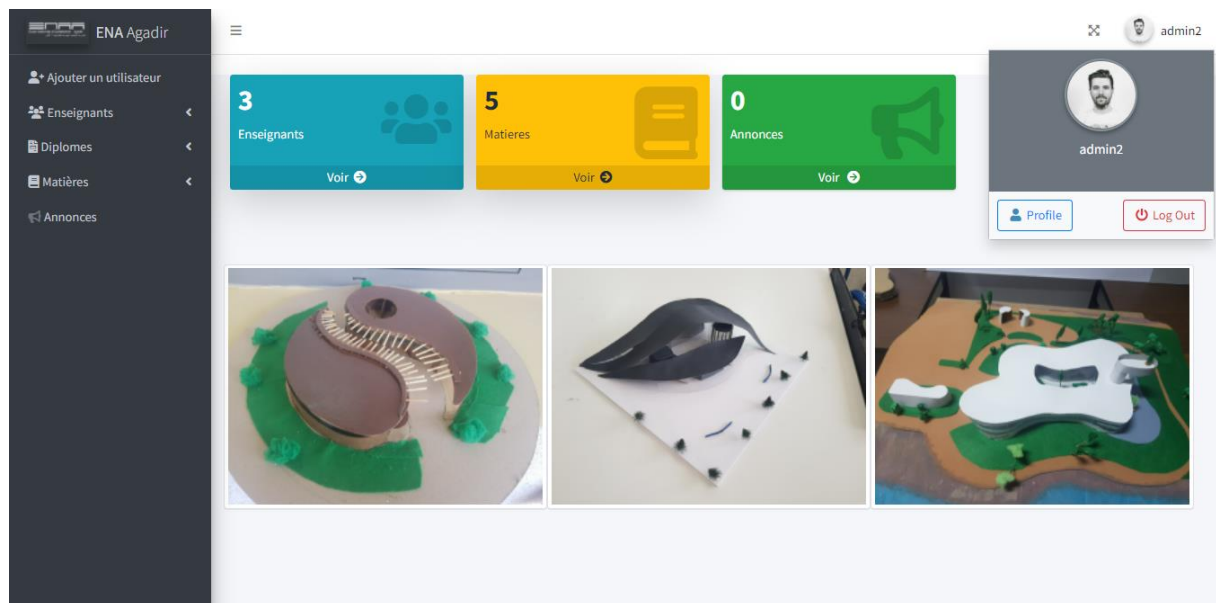


Figure 19 : page d'accueil de l'administrateur

Ce formulaire permet à l'administrateur d'ajouter des nouveaux enseignants en remplissant les champs nécessaires à remplir, la figure suivante (figure 20) présente ces champs notamment le nom, le prénom et l'image.

Par conséquent l'enseignant ajouté devient un utilisateur (partie enseignant), avec un mot de passe composé de son nom et son CIN.

Figure 20 : ajouter un enseignant

L'interface de la figure 21 représente les tâches que l'administrateur peut faire :

- ✓ consulter la liste des enseignants.
- ✓ ajouter des enseignants.
- ✓ visualiser les informations des enseignants.
- ✓ modifier et supprimer un enseignant.

The screenshot shows the 'La liste des enseignants' page in the ENA Agadir application. The sidebar on the left contains navigation options: 'Ajouter un utilisateur', 'Enseignants' (selected), 'Liste des enseignants', 'Ajouter un enseignant', 'Enseignants supprimés', 'Diplomes', 'Matières', and 'Annonces'. The main content area displays '3 ENSEIGNANTS' and a table with the following data:

Numéro d'ordre	Photo	Nom	Prénom	CIN	Téléphone	Email	Actions
1		El moudni	Hicham	JD123456	0636578490	hichamlmodni@gmail.com	
2		Tanassat	Rachida	JD00000	+212678965432	rachidatanassat@gmail.com	
3		Bougrn	Hafid	JE1111	+212643897653	Hafid@gmail.com	

Below the table, it says 'Showing 1 to 3 of 3 entries' and there are 'Previous', '1', and 'Next' navigation buttons.

Figure 21 : la liste des enseignants

L'application permet notamment la gestion des matières :

- ✓ consulter la liste des matières.
- ✓ ajouter des matières.
- ✓ modifier ou supprimer une matière.

Cela est représenté dans la figure 22.

ENA Agadir

admin2

La liste des matieres

Copy CSV PDF Print Column visibility

Search:

Numéro	Libelle	Semestre	Actions
1	Art et design	S1	
2	Commerce et vente	S2	
3	Informatique et systeme	S3	
4	La mode	S5	
5	La gestion et l'organisation	S6	

Showing 1 to 5 of 5 entries

Previous 1 Next

Figure 22 : la liste des matières

L'administrateur peut aussi visualiser les informations d'un enseignant ainsi ses diplômes, les matières qu'il enseigne, comme il peut imprimer une attestation de vacation.

ENA Agadir

admin2

Afficher l'enseignant

EL MOUDNI HICHAM

[Matiere](#) [Diplome](#) [CONVENTION DE VACATION](#)

Nom
El moudni

Prénom
Hicham

CIN
JD123456

Date de naissance
2002-05-24

Sexe

Figure 23 : fiche d'un enseignant

L'attestation de vacation s'affiche comme le montre la figure ci-dessous (figure 24), on récupère des données nécessaires depuis la base de données (le nom, le prénom, l'année scolaire, la matière et le diplôme).

The screenshot shows a web application interface for 'ENA Agadir'. On the left is a dark sidebar with navigation options: 'Ajouter un utilisateur', 'Enseignants', 'Diplomes', 'Matières', and 'Annonces'. The main content area is titled 'Imprimer la convention' and displays a document titled 'CONVENTION DE VACATION' for the academic year 2021/2022. The document header includes the logo of the 'ROYAUME DU MAROC' and the 'Ecole Nationale d'Architecture AGADIR'. The document text includes sections for 'Entre:', 'D'une part', 'ARTICLE 8 : Litige', and 'ARTICLE 9 : Validité'. At the bottom, there is a date field and a date stamp 'AGADIR, le 2022-05-07'. Two small icons (a red one and a blue one) are visible at the bottom right of the document area.

Figure 24 : imprimer la convention de vacation

La figure ci-dessous représente un exemple d’affichage d’un diplôme.



Figure 25 : voir le diplôme d’un enseignant

Cette interface (figure 26) rassemble les enseignants avec leurs diplômes pour faciliter la recherche d’un enseignant en se basant sur son diplôme.

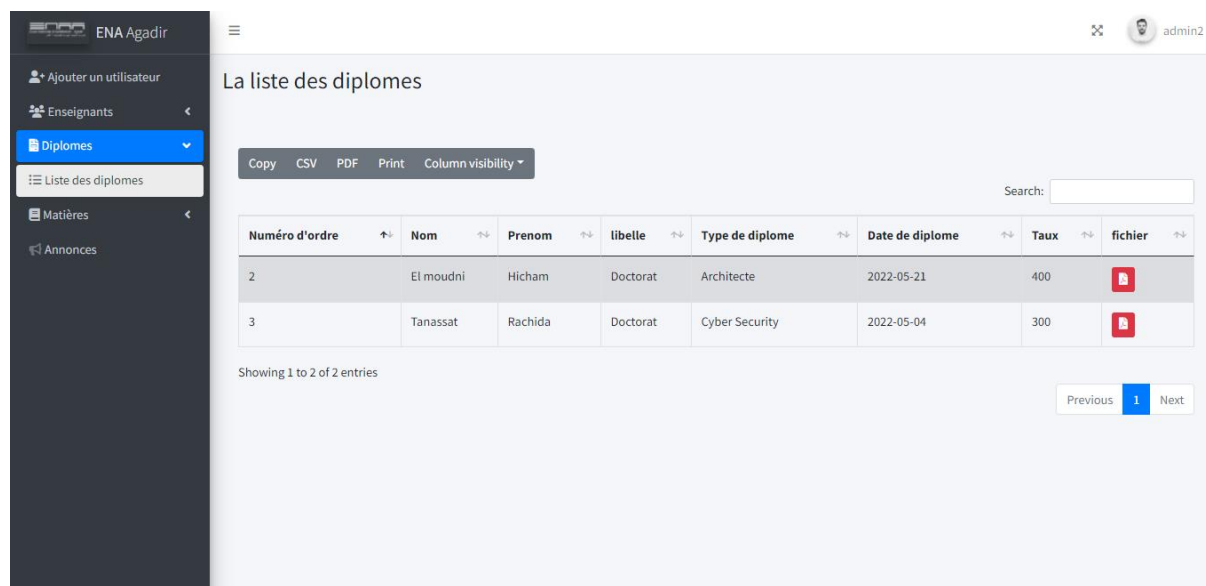


Figure 26 : la liste des diplômes

Notre application permet à l'administrateur de publier des annonces qui seront affichées par la suite aux enseignants et aux étudiants (figure 27).

Une fois l'administrateur publie une annonce, une notification sera envoyée aux enseignants et aux étudiants par email.

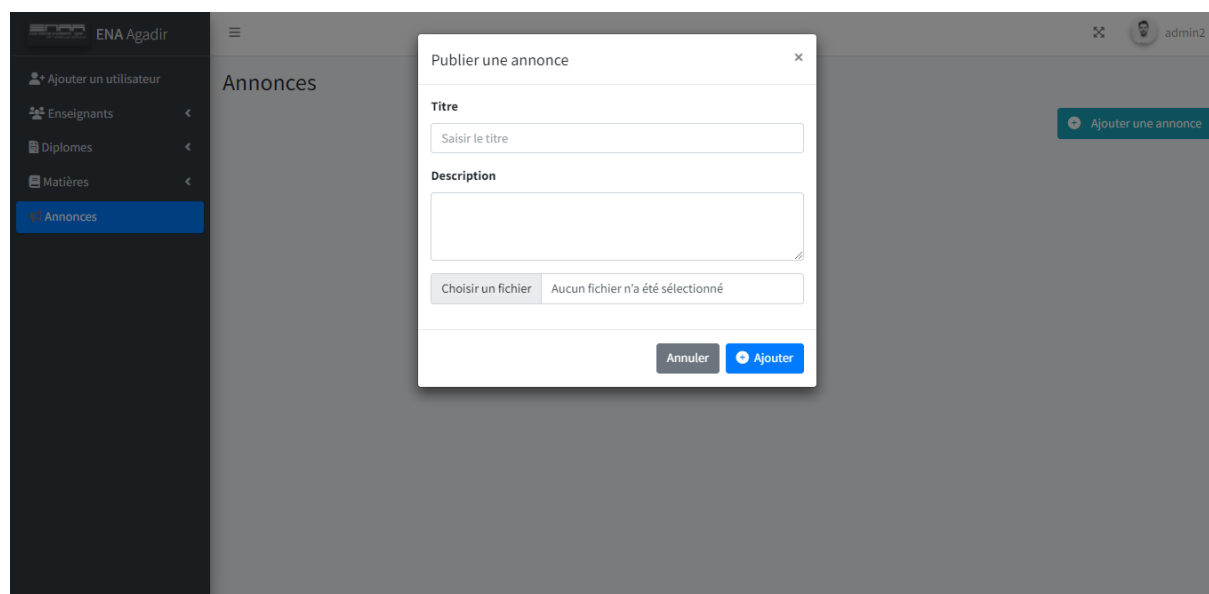


Figure 27 : publier une annonce

- **Partie enseignant**

L'interface suivante (figure 28) présente la page d'accueil de l'enseignant, qui se compose initialement du menu général et des annonces publiées par l'administrateur.

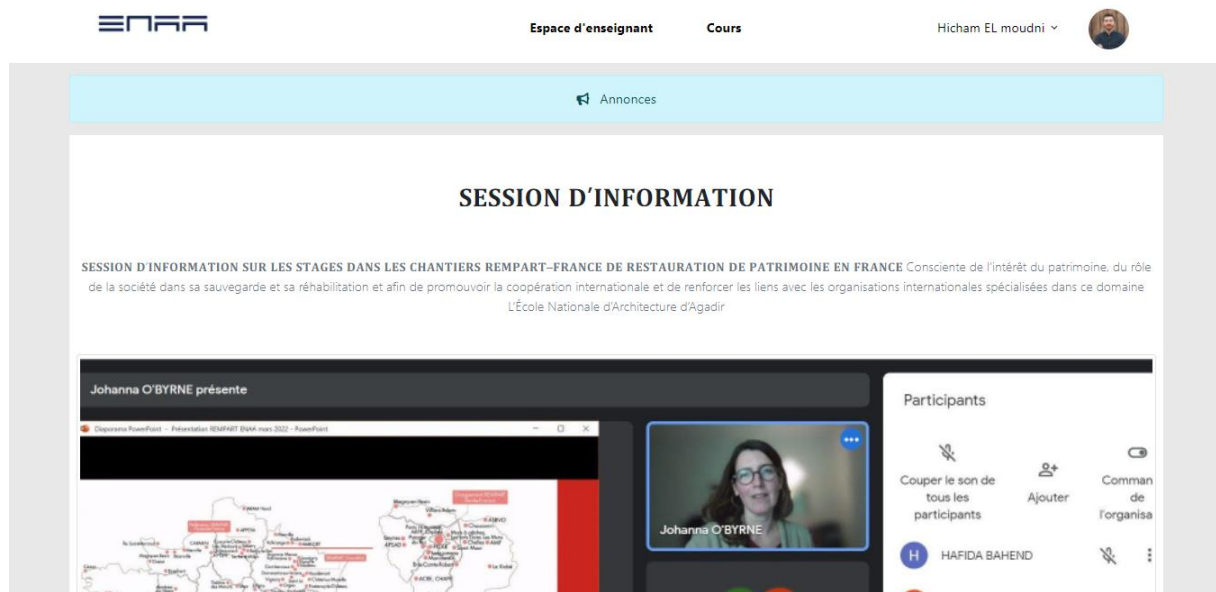


Figure 28 : Page d'accueil de l'enseignant

L'interface ci-dessous (figure 29) représente le profil d'un enseignant connecté. Ce dernier peut modifier son nom et son mot de passe.

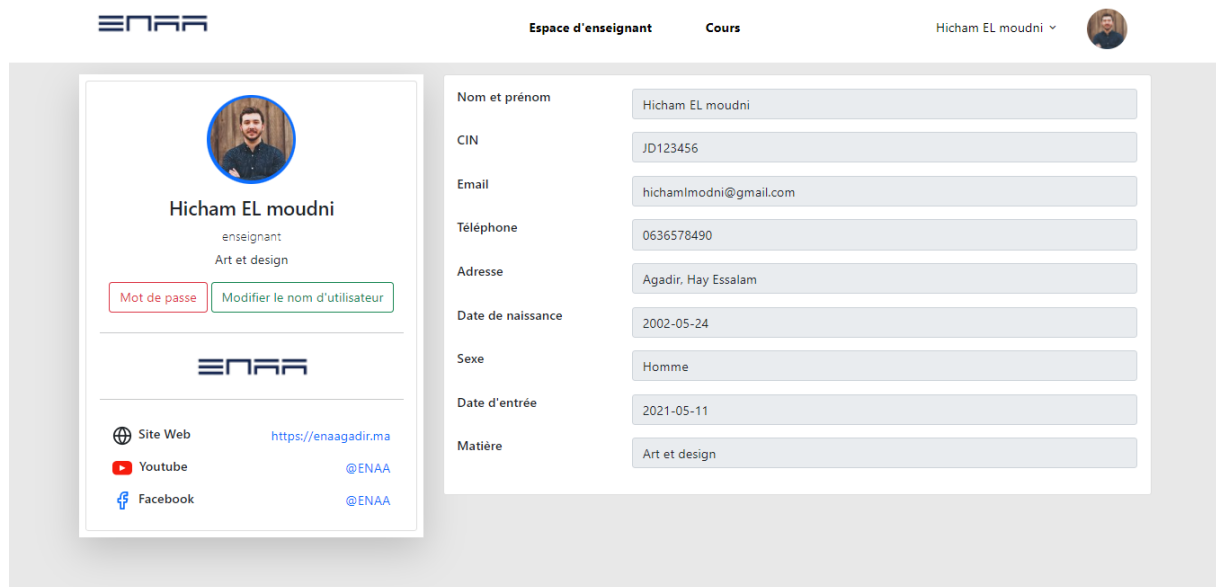


Figure 29 : le profil d'un enseignant

Chaque enseignant peut publier des cours. Après cela est fait tous les étudiants de l'application reçoivent une notification par email (figure 30).

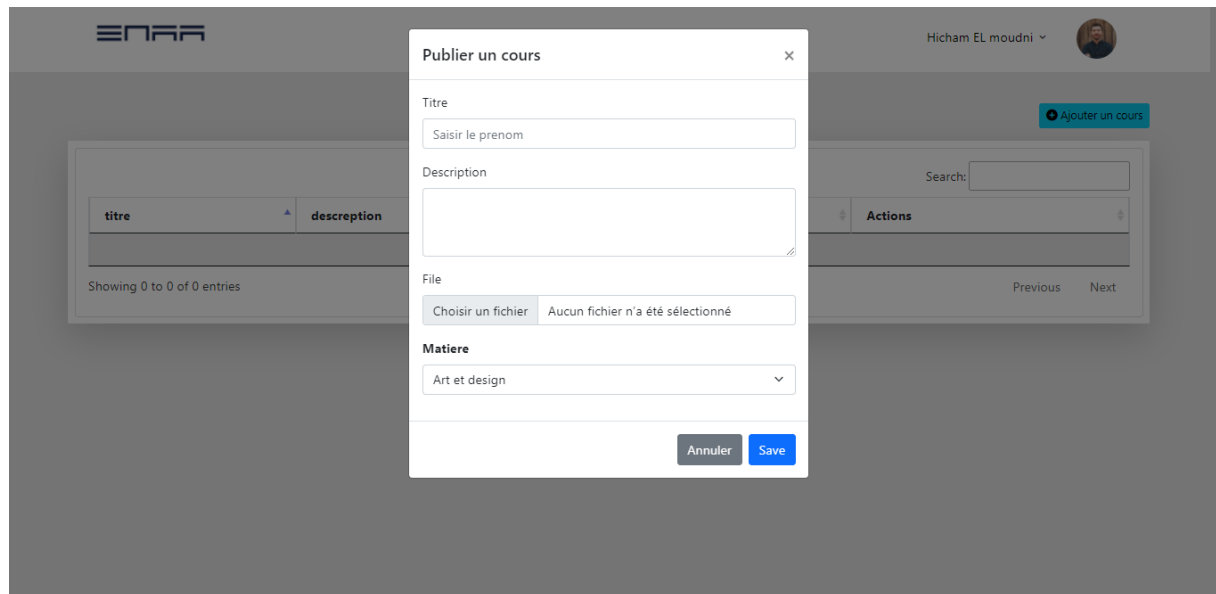


Figure 30 : publier un cours

Lorsqu'un cours est ajouté, une redirection est faite vers cette interface (figure 31) qui affiche tous les cours publiés afin que l'enseignant puisse les modifier ou les supprimer.

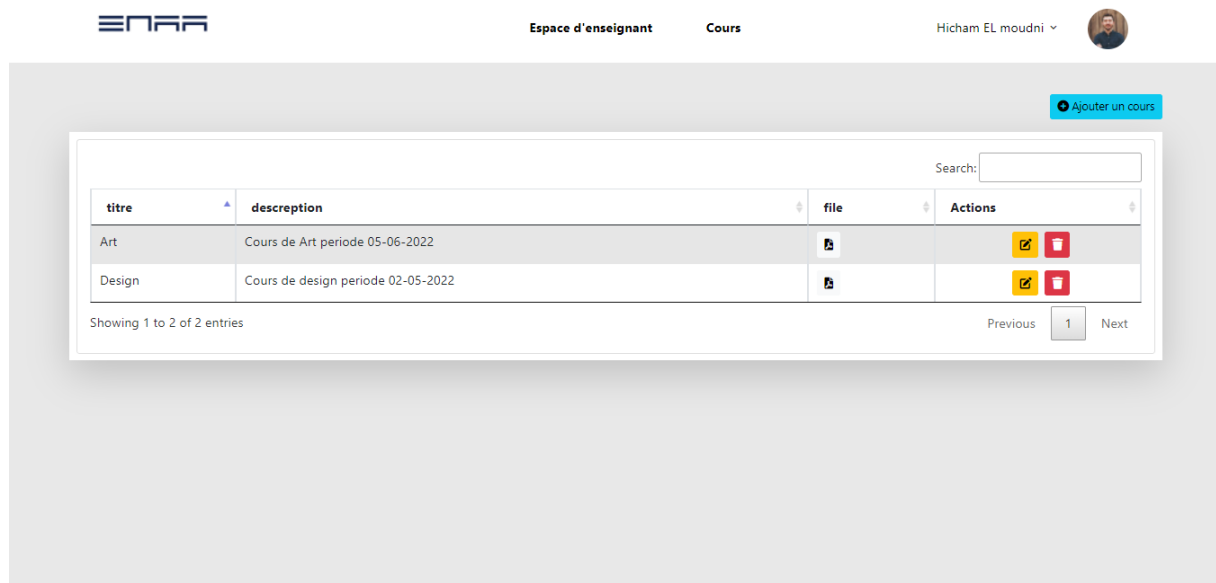


Figure 31 : la liste des cours

- **Partie étudiant**

L'interface ci-dessous permet aux étudiants la recherche d'un cours, en spécifiant le semestre. Une liste des matières liées à ce semestre sera disponible pour la sélection. En appuyant sur le bouton de recherche, La liste des cours associée à la matière choisie sera affichée (figure 33).

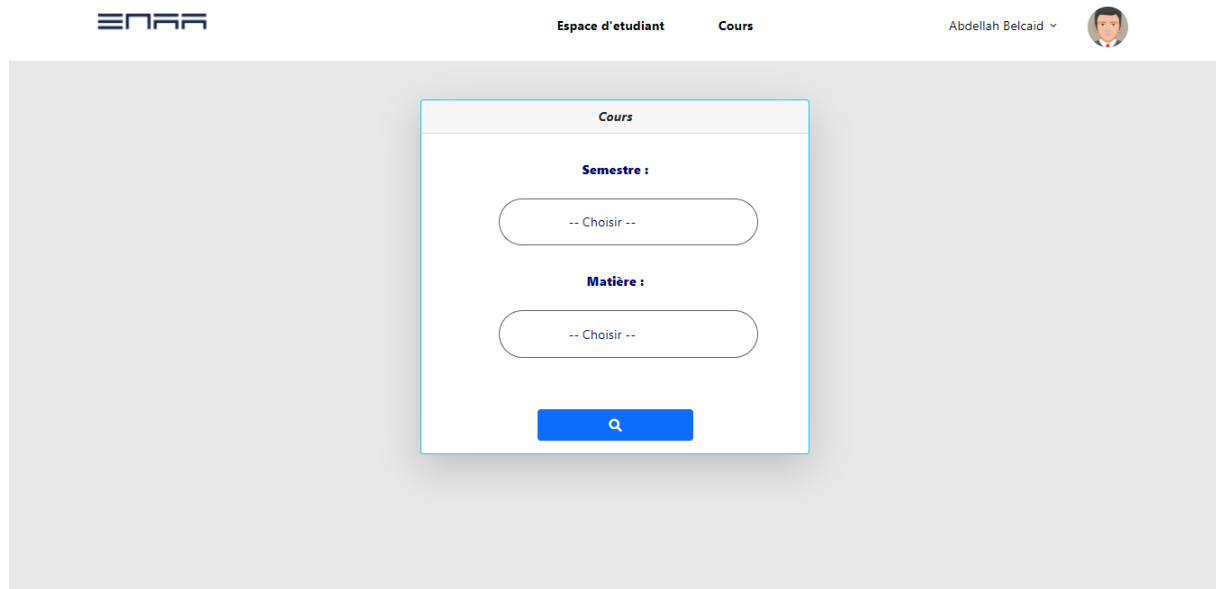


Figure 32 : chercher un cours

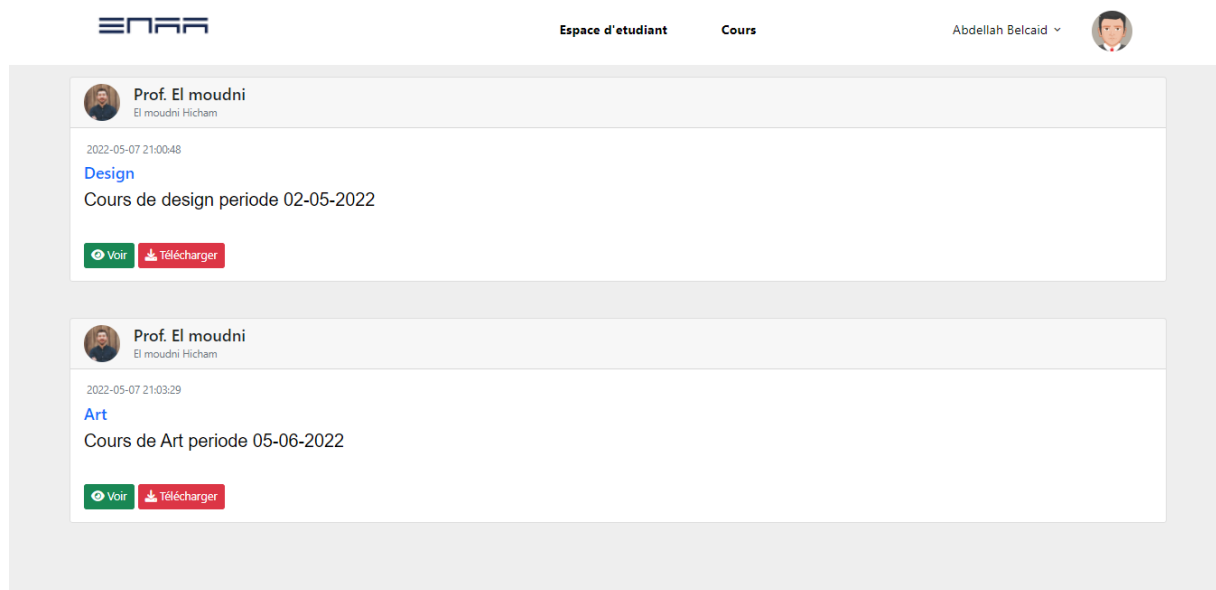


Figure 33 : la liste des cours

Pour tester les notifications, on a opté pour l'utilisation de « Mailtrap ».

La figure suivante montre les notifications envoyées par l'application après chaque publication d'une annonce ou d'un cours.

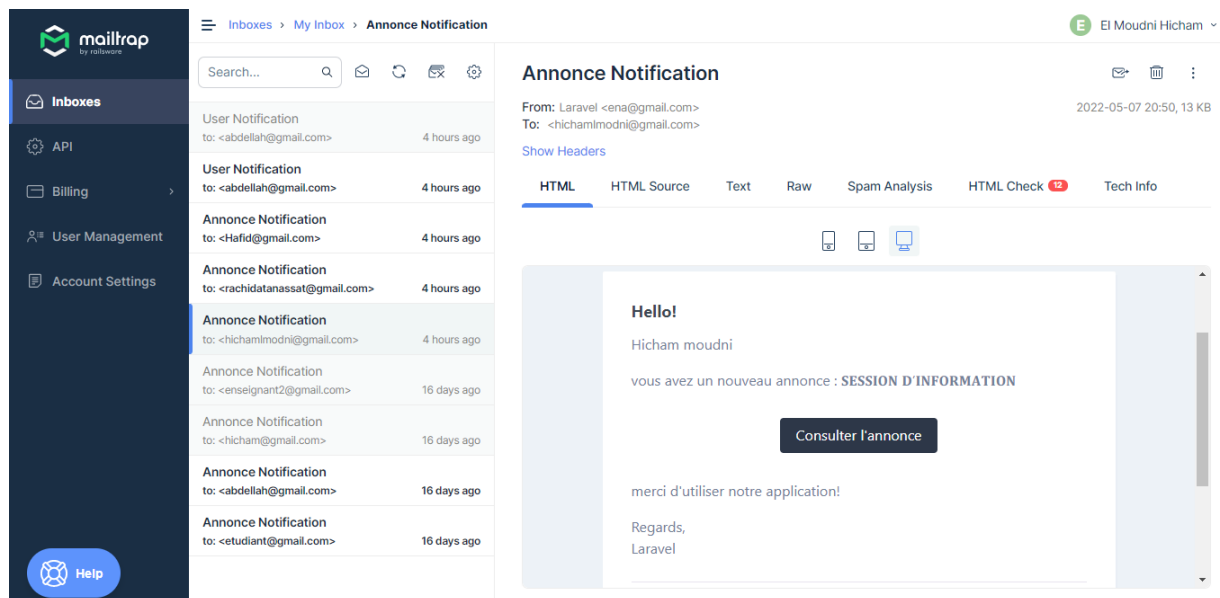


Figure 34 : les emails “mailtrap”

CONCLUSION GENERALE

Notre projet se résume à la conception et la réalisation d'une application web permettant de gérer les enseignants de l'École Nationale d'architecture Agadir. Tout au long de ce rapport, nous avons présenté la démarche que nous avons suivie pour mettre au point cette application.

En ce qui concerne la démarche poursuite, nous avons commencé dans un premier lieu par comprendre le contexte général de notre application et prendre des notes de toutes les difficultés qui rencontre l'ENAA, en détaillant le cahier des charges à l'aide des diagrammes UML. Nous avons préparé par la suite la stratégie du travail en respectant les besoins fonctionnels et non fonctionnels et finalement pour la phase de la programmation, on est supervisé par l'architecture MVC.

Durant la réalisation de notre projet, nous avons pu mettre en œuvre nos connaissances acquises tout le long de notre formation et nous étions mené à acquérir de nouvelles connaissances tel que la programmation PHP, Laravel et d'aller plus en profondeur au niveau de la pratique. Ce projet nous a permis ainsi d'évaluer nos compétences et d'acquérir une réelle expérience aussi bien méthodologique que technique.

WEBOGRAPHIE

Laravel : <https://laravel.com/>

MySQL : <http://dev.mysql.com/doc/>

HTML: <http://www.w3schools.com/html/default.asp>

CSS: <http://www.w3schools.com/css/default.asp>

JavaScript : <http://www.w3schools.com/js/default.asp>