**Département d’informatique**

**Licence fondamentale :**

**Sciences Mathématiques et Informatique (SMI)**

**Projet de Fin D’études**



**Conception et réalisation d’un site web d’enseignement à distance au profit des étudiants et des enseignants-chercheurs**

**Réalisé par : Encadré par :**

**BENAMMOU Abdessamad PR.** **AGHOUTANE Badraddine**

**HADDOUCHE Jamal**

Soutenu le 09-07-2021 devant le jury :

**AGHOUTANE Badraddine, Professeur à la Faculté des Sciences- Meknès**

**BOURRAY Hamid, Professeur à la Faculté des Sciences- Meknès**

**EL OUADGHIRI MLY Driss, Professeur à la Faculté des Sciences- Meknès**

 **Année Universitaire : 2020-2021**

**Dédicaces**

Ce projet de fin d’étude est dédié avec fierté à nos chers parents, source de vie, d’amour et d’affectation pour leurs soutien et leurs prières, leurs sacrifices et leurs encouragements qu’ils nous ont prodigués tout au long de nos scolarités.

C’est un moment de plaisir de dédier ce travail, à nos chers frères et sœurs, source de joie et de bonheur. À toutes nos familles, source d’espoir et de motivation. À tous nos amis de promotion 3 -ème année licence, toute personne qui occupe une place dans nos cœurs. Et finalement, à nos chers ami(e)s qui n’ont jamais cessé de nous soutenir.

**Remerciements**

Ce travail n’a pu être possible que du fait de personnes qui y ont aimablement consacrés leur temps. Nos remerciements vont à notre cher professeur et encadrant Badraddine AGHOUTANE pour son énorme soutien qu’il n’a pas cessé de nous accorder tout au long de la période du projet, merci pour sa bonne volonté, sa patience et ses précieux conseils ainsi que pour la pertinence de ses remarques.  
 Un remerciement particulier est adressé à nos professeurs pour la qualité de l’enseignement qu’ils nous ont abondé au cours de ces trois années passées à la faculté des sciences Meknès. Les remerciements seraient incomplets si nous ne citons pas les jurys, chacun par son nom pour avoir bien voulu examiner et juger notre travail.

**SOMMAIRE**

**Liste des figures**………………………………………………………………………………………………………….6

[**Introduction**](#introduction)…..….…………………………………………………………………..…………………..…………..…...7

[**Chapitre I :** **L'apprentissage à distance**](#partie1) **……………………………………….**…………….….9

**Ⅰ-1: Définition de l'apprentissage**……….**………….………………………………….**….……9

**Ⅰ-2: Les types d’apprentissage**……………**…………………………………….….**………...……9

**I-2-1 : Apprentissage en face à face**……………**……………………………….**…………9

**I-2-2 : Apprentissage en distanciel**………**…………………………….……….**………….9

I-2-2-1 : L’enseignement synchrone (ZOOM, TEAMS, etc.)…………………...9

I-2-2-2 : L’enseignement asynchrone (Moodle, Microsoft 365, etc.) …...10

**I-2-3 L’enseignement hybride**……………………………………………………….………11

**Chapitre II : Analyse et spécification des besoins ..……………….**……………… 13

**II-1 : Introduction …………………………………………………………….…………………….**……….13

**II-2- Les plates-formes pédagogiques d'apprentissage en ligne .**….……13

**II-3- Etude de l’existant ……………………………………………….………………….**…………….13

**II-3-1- Analyse de l’existant…………………………………….………………………….**…….13

II-3-1-1- Les MOOCs**………………………………………………………………..…….**………14

II-3-1-2- Les classes virtuelle (ZOOM, TEAMS, AdobeConnect, etc.) **.**…….14

II-3-1-3- Les Plateformes LMS………………………………………………………………..14

**II-3-2- Critique de l’existant**…………………………………………………………..…………15

**II-3-3- La solution proposée**……………………………………………………………………..16

**II-4- Spécification des exigences**………………………………………………………………….16

**II-4-1- Etude des besoins fonctionnels …………………………………………………….**16

**II-4-2- Etude des besoins non fonctionnels ……………………………………………..**16

**Chapitre III :** **Modélisation Conceptuelle et Organisationnelle**  ….…17

**III-1: Modélisation Conceptuelle**……………………………………………….……………..…17

**III-1-1- Introduction …………….…………………………………………………………..……**17

**III-1-2- Choix de la méthodologie de conception** …………………………..……....17

**III-1-3- Les acteurs système …**……………………………………………….………..…….…18

**III-1-4- Identification des cas d'utilisation ………………………………………….....**18

**III-1-5- Diagramme de paquetages ………………………………………………….….…**19

**III-1-6- Raffinement des cas d’utilisation ……………………………………………….**19

III-1-6-1. Cas d’utilisation « S’authentifier » ………………………………...19

III-1-6-2 Cas d’utilisation « Gérer le profil » …………………………………..20

III-1-6-3 Cas d’utilisation « Gérer les utilisateurs » ………………………..21

III-1-6-4 Cas d’utilisation « Gérer les cours »………………………..………. 22

III-1-6-5 Cas d’utilisation « Gérer les devoirs »………………………….…...23

III-1-6-6 Cas d’utilisation « Gérer les tests »………………………….….……24

**III-1-7- Diagramme de classes**………………………………………………………………….25

**III-2 : Modélisation organisationnelle**……………………………………..………….…….26

**III-3 : Conclusion**…………………………………………………………………………………………...28

**Chapitre IV :** **Réalisation**……………………………………………………………………………..……….29

**IV -1: Introduction**…………………………………………………………………………………..……….29

**IV -2: Choix des langages de développement, de SGBD et les …………….**29

**Outils de Développement**

**IV -3 : Architecture…………………………………………………………………………………………**30

**IV -4 : Présentation des interfaces**…………………………………………………………..…..32

**IV -5 : Apports**…………………………………………………………………………………………………….39

**IV -5 -1 : Apports au niveau des connaissances techniques** ………………………40

**IV -5 -2 : Apports au niveau de la conception et du développement** ….......40

**IV-6 : Conclusion**……………………………………………..……………………………………………….40

**Conclusion et perspectives** …………………………………………………………………………….41

**Bibliographie**………………………………………………..……………………….………………………………….42

**Webliographie** ………………………………………………………………………………………….…………….43

**Liste des figures**

Figure 1 : Diagramme de paquetages………………………………………………..…………………………...18

Figure 2 : Cas d’utilisation « S’authentifier » …………………………………………………………………..18

Figure 3 : Cas d’utilisation « Gérer le profil » ……………………………………………………..…………..19

Figure 4 : Cas d’utilisation « Gérer les utilisateurs »…………………………………………………20

Figure 5 :Diagramme de séquence « Ajouter utilisateur »………………………………………21

Figure 6 : Cas d’utilisation « Gérer les cours »………………………………………………………… 21

Figure 7 : Diagramme de séquence « Gérer les cours »………………………………………….. 22

Figure 8 : Cas d’utilisation « Gérer les devoirs »……………………………………………………… 22

Figure 9 : Cas d’utilisation « Gérer les tests »…………………………………………………………. 23

Figure 10 : Diagramme de classes……………………………………………………………………………24

Figure 11 : Architecture 3-tiers………………………………………………………………………………..30

Figure 12 : Page d’accueil………………………………………………………………………………………..31

Figure 13 : Page des cours publics……………………………………………………………………………32

Figure 14 : Page de connexion…………………………………………………………………………………32

Figure 15 : Page d’administrateur……………………………………………………………………………33

Figure 16 : Sidebar d’administrateur……………………………………………………………………….33

Figure 17 : Liste des enseignants……………………………………………………………………………..34

Figure 18 : Liste des apprenants………………………………………………………………………………34

Figure 19: Formulaires pour saisir les informations de l’enseignant ………………….…….34

et de l’apprenant

Figure 20 : Liste des Filières…………………………………………………………………….………………34

Figure 21 : Profil enseignant……………………………………………………………………………………35

Figure 22 : Liste des cours………………………………………………………………………………………35

Figure 23 : Liste des tests………………………………………………………………………………………..35

Figure 24 : Liste des devoirs…………………………………………………………………………………….35

Figure 25 : Liste des messages…………………………………………………………………………………36

Figure 26 : Carte enseignant……………………………………………………………………………………36

Figure 27 : Sidebare de l’apprenant…………………………………………………………………………37

Figure 28 : Interface de cours………………………………………………………………………………….37

Figure 29 : Formulaire message………………………………………………………………………………37

Figure 30 : Devoir…………………………………………………………………………………………………….38

**Introduction**

L'éducation est une nécessité légale pour tout le monde, mais avec ces circonstances de la pandémie de Corona virus tous les écoles et les facultés se trouve devant l'obligation de se fermer, d’où la nécessité d'intervenir entant que des étudiants de licence en mathématique et informatique et de mettre en place notre techniques et notre savoir accumulés tout au long de notre parcours éducatif.

Notre projet de fin d'étude est une réponse au problématique qui concerne la continuité des études en se basant sur la création d'une plateforme qui assure le suivi normal des études à distance, d'une manière permettant aux étudiants d’intervenir comme il veut, tout comme la mise en place de plusieurs fonctionnalités à savoir :

* Fournit pour l’administrateur un espace pour gérer les utilisateurs (ajouter, supprimer, rôle).
* Fournit pour l’enseignant un espace pour partager les cours, devoirs et test avec ses étudiants ainsi que la suivie de processus des devoirs et des tests.
* Fournit pour les étudiants un espace pour suivi leurs cours, et un espace pour contacter leurs enseignants.

Le présent rapport s’articule autour de quatre chapitres. Nous commencerons par le chapitre « L'apprentissage à distance » qui localise le contexte général du notre projet et met l’accent sur le champ d’étude de notre application qui est l’apprentissage à distance.

Le deuxième chapitre « Analyse et spécification des besoins » dédié à l’étude de l’existant où nous présenterons une synthèse des solutions existantes sur le marché en discutant les avantages et les inconvénients de chacune d’elles. Puis, nous formulerons les principales fonctionnalités de la solution à réaliser.

Le troisième chapitre « Modélisation Conceptuelle et Organisationnelle », dans  
lequel nous présenterons la notation de modélisation utilisée ainsi que l’ensemble des diagrammes conçus.

Dans le dernier chapitre « Réalisation », nous détaillerons notre environnement de travail suivis d’une présentation des différentes fonctionnalités de notre plateforme à travers des capture d’écran.

**Chapitre I : L'apprentissage à distance**

**I-1- Définition de l'apprentissage**

L'enseignement est une pratique, mise en œuvre par un enseignant, visant à transmettre des compétences à un élève, un étudiant ou tout autre public dans le cadre d’une institution éducative.

**I-2- Les types d’apprentissage**

**I**-**2**-**1** Apprentissage en face à face **:** c’est un enseignement où toutes les activités sont en présence, en personne, l’étudiant peut utiliser un appareil comme un ordinateur, une tablette, un téléphone mobile pour faire de la recherche sur des sites web, les enseignants se servant d’un projecteur et des diapos power point, mais la classe reste le principal moyen de prestation.

**I**-**2**-**2** Apprentissage en distanciel **:** L’enseignement à distance est l’ensemble des dispositifs fournissant une instruction par communication d'imprimés ou d'information électronique à des personnes engagées dans un apprentissage planifié à un lieu ou un moment différent de ceux auxquels interviennent le ou les instructeurs. Il y a deux principaux types de l'enseignement à distance :

**I**-**2**-**2**-**1** L’enseignement synchrone (ZOOM, TEAMS, etc.) **:**

Enseignement et apprentissage en ligne dans lesquels les participants interagissent en même temps et dans les mêmes espaces, les vidéoconférences ou les conférence web sont des exemples d’outils utilisés à des fins d’enseignement synchrone.

* **Les avantages de l’enseignement synchrone :**

|  |  |
| --- | --- |
| **Avantages pour l’enseignant :** | **Avantages pour l’étudiant :** |
| * Facilite les échanges avec les étudiants et permet de donner rapidement des indications. * Incite les étudiants à une meilleure disposition à prendre part aux activités collaboratives. * Implique de varier les stratégies pédagogiques utilisées de diversifier les approches d’enseignement et d’apprentissages afin de d’alimenter la motivation et la persévérance des étudiants. * Facilite les échanges avec les étudiants et permet de donner rapidement des indications. | * Développe le sentiment d’appartenance à un groupe, ce qui est motivant (offre la possibilité d’échanger avec les autres étudiants). * Si la séance de cours est enregistrée, l’étudiant a la possibilité de la consulter en différé (flexibilité plus grande qu’en classe). * Donnant des cours par vidéoconférence, ce qui permet aux élèves de participer à distance à partir de lieux géographiquement disparates dans leur autorité scolaire. |

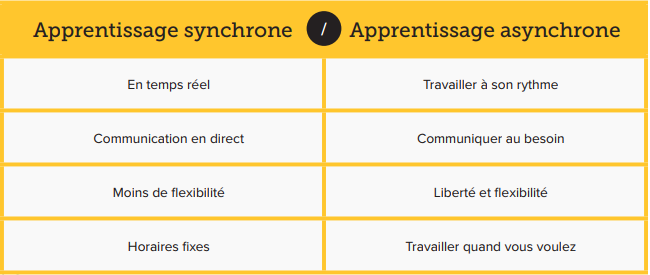
* **Les défis de l’enseignement synchrone :**
* Contrainte d’horaire précis : moments de pause difficiles.
* Il oblige tous les apprenants à avancer au même rythme, sans flexibilité quant au moment ou à la manière d’appréhender le contenu.
* Encadrement pointu de la séance : silences gênants, conversations simultanées et doutes sur la réaction aux idées.
* Nécessité d’un effort de concentration plus grand de la part de l’étudiant.

**I-2-2-2** L’enseignement asynchrone (Moodle, Microsoft 365, etc.) **:** L’enseignement asynchrone désigne la communication et les tâches exécutées à différents moments. La communication par courriel est un exemple de communication asynchrone. Dans l’enseignement à distance, les étudiantes ont généralement des ressources pour le travail asynchrone telles que des textes, vidéos, exercices autocorrigés, forums de discussion, etc. Les exercices dans un manuel ou les devoirs traditionnels à la maison sont des exemples d’activités asynchrones non médiatisées. L’asynchrone se produit donc en ligne (sur une plateforme numérique).

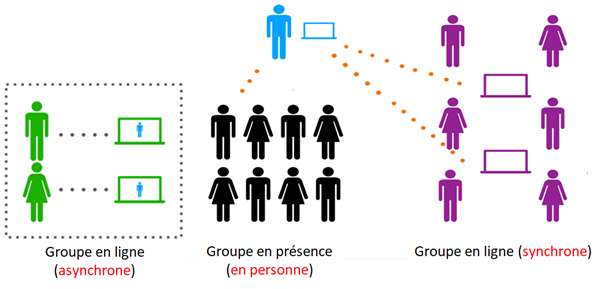
* **Les avantages de l’enseignement asynchrone :**

|  |  |
| --- | --- |
| **Avantages pour l’enseignant :** | **Avantages pour l’étudiant :** |
| * Offre la possibilité de concevoir des questionnaires rapides à corriger (à l’aide de raccourcis ou d’une banque de réponses). * Possibilité de modifier et d’optimiser le contenu en cas de besoin * Favorise la création d’activités interactives pour présenter du contenu de manière dynamique. * Contenu utilisable parallèlement à celui d’un cours traditionnel dans le cadre de méthodes d’apprentissage mixtes. * Contenu réutilisable à l’infini sans présentation en direct. | * Facilite la conciliation travail famille-études grâce au contenu accessible en tout temps. * Pousse les étudiants à interagir avec le contenu, les enseignants et les autres apprenants. * Chaque apprenant travaille à son propre rythme. Dans le cas où vous seriez plus rapide que les autres, il n’y a aucunement besoin de les attendre. L’inverse est également vrai : si vous êtes plus lent, vous pouvez prendre votre temps sans gêner qui que ce soit. Les formations peuvent également être façonnées sur mesure en fonction du niveau d’apprentissage. * Réduit les déplacements. * Octroie la possibilité de choisir le meilleur moment pour réaliser les activités (flexibilité de temps et d’espace). * Possibilité d’analyser ce qu’on a appris et d’y réfléchir avant de participer aux discussions ou activités du cours. |

* **Les défis de l’enseignement asynchrone :**
* Demande une planification rigoureuse de l’ensemble de la session (requiert plus de temps de préparation).
* Besoin d’autodiscipline des étudiantes, gestion du temps et des ressources.
* Si le contenu n’est pas clair, les étudiants tendent à prendre du retard.
* Les apprenants passent énormément de temps sur leur ordinateur et d’autres appareils de ce type ; ce qui peut entraîner de la fatigue visuelle.



**I**-**2**-**3** L’enseignement hybride: Un cours hybride est conçu de sorte que certaines heures de classe sont remplacées par des activités en ligne tout aussi importantes. Cela signifie que les parties en classe et en ligne d’un cours sont complémentaires et ont été combinées de manière réfléchie afin de bien répondre aux besoins de l’étudiant et aux objectifs du cours. Les composantes en ligne ne sont pas un ajout à une charge de cours complète, mais plutôt une substitution réfléchie pour certaines activités en classe.



**Chapitre II : Analyse et spécification des besoins**

**II-1- Introduction**

Ces dernières années ont vu Internet prendre de plus en plus d'importance et finalement devenir la plate-forme universelle de diffusion d'informations, avec un essor de la communication, de la collaboration et de l'enseignement à distance. La vue sur l'enseignement prend un nouveau visage, nous assistons à un nouveau besoin, de l'enseignement traditionnel dispensé dans des salles de cours, nous migrons vers une demande en temps réel de l'information**.**

**II-2-** **Les plates-formes pédagogiques** **d'****apprentissage en ligne**

Une plate-forme pédagogique d'apprentissage en ligne, appelée encore **LMS** (**Learning Management System)**, est un logiciel qui assiste la conduite des formations présentielles et à distance. Elle est basée sur des techniques de travail collaboratif et regroupe les outils nécessaires aux trois principaux acteurs de la formation : apprenant, formateur, administrateur. Elle fournit à chaque acteur un dispositif qui a pour première finalité l’accès à distance au contenu pédagogique, l’auto apprentissage, l’autoévaluation et la télé tutorat via l’utilisation des moyens de travail et de communication à plusieurs : visioconférence, e-mail, forums, chats, annotations, etc. Le but est donc de combler la perte de cohésion et de stimulation de la salle que peut sentir l’apprenant devant sa machine.

**II-3-** **Etude de l’existant**

Cette section a pour objectif d’étudier les solutions d’apprentissage en ligne (**E-Learning**) les plus connues sur le marché. Cette étude permet de dégager les points forts et les points faibles de chacune de ces solutions.  
Dans ce qui suit, nous présentons une analyse de l’existant, puis nous détaillons la critique de l’existant.

**II-3-1- Analyse de l’existant :**

La formation continue se fait actuellement de façon traditionnelle : cours, apprenants et formateurs sur place. Ce type de formation présente beaucoup d’inconvénients tels que :

✓ Contrainte du nombre de Places limitées ;

✓ Contrainte du nombre de salles réduites ;

✓ Charge élevée de la formation ;

Dans le but de résoudre ces inconvénients plusieurs outils ont étais créer à base des nouvelles technologies. Parmi lesquels nous pouvons citer :

**II-3-1-1- Les MOOCs**

Le MOOC (Massive Open online Course) est un nouveau format pédagogique qui consiste à mettre en libre accès des formations en ligne. Ces dernières sont la plupart du temps gratuites.

À l’instar d’une formation classique en e-learning, le MOOC est une technique d’apprentissage numérique qui met en scène un ou plusieurs intervenants professionnels. Ces formations sont cependant ouvertes à tous, contrairement à des formations à distance classiques qui sont limitées à un certain nombre d’apprenants. Les apprenants ont alors accès à plusieurs cours en ligne, mais aussi à différents contenus pédagogiques interactifs variés, comme des vidéos ou des quizz.

Les principales plateformes pour les MOOC sont **Coursera**, **edX**, **Udacity**, …

**II-3-1-2- Les classes virtuelle (ZOOM, TEAMS, AdobeConnect, etc.)**

Une classe virtuelle est comme son nom l’indique une formation qui réunit le formateur et les apprenants à distance en direct (on parle d’activité synchrone). Bien souvent, la classe virtuelle prend donc la forme d’une visioconférence : formateur et participants peuvent se voir échanger par audio ou tchat.

Mais elle offre en plus des fonctionnalités très utiles pour l’animation de formation : partage de documents et des affichages, sondage, tableau blanc, quiz, possibilité de répartir les participants en sous-groupes…

**II-3-1-3- Les Plateformes LMS**

Le terme LMS est l’acronyme de Learning Management System, pouvant se traduire en français par système de gestion de l’apprentissage. Plus concrètement ce terme désigne une plateforme de formation en ligne. Cette plateforme pédagogique permet de concevoir de structurer et de diffuser le contenu pédagogique auprès des apprenants. Un LMS comporte l’ensemble des outils nécessaires à l’enseignement et à l’apprentissage.

Les plateformes LMS comportent des fonctionnalités caractéristiques :

* Création de formations en ligne
* Les plateformes LMS permettent d’effectuer un suivi des apprenants (temps passé sur la plateforme, nombre de modules de cours suivis, résultats aux évaluations, etc.)
* Les LMS comportent des outils de communication plus ou moins poussés selon la plateforme utilisée.
* **Moodle :** terme « Moodle » était à l’origine un acronyme pour "***Modular Object Oriented Dynamic Learning Environment***".

Moodle est une plateforme d'apprentissage destinée à fournir aux enseignants, administrateurs et apprenants un système unique robuste, sûr et intégré pour créer des environnements d'apprentissages personnalisés.

Moodle présente de nombreuses caractéristiques : multilinguisme, forums, gestionnaire de ressources, tests et modules clé en main (Devoirs, Chat, Sondage, Glossaires, Journal, Etiquettes, Leçons, Wiki), etc.

**II-3-2- Critique de l’existant :**

Le tableau récapitule les avantages et les inconvénients de tous les outils traités.

|  |  |  |
| --- | --- | --- |
| **Outils** | **Avantages** | **Inconvénients** |
| **Les MOOCS** | * La plupart des MOOC sont gratuits * Un étudiant peut suivre une formation d ‘une université située à l’autre bout du monde sans avoir à effectuer de déplacement. * Le travail peut être partagé, regardé et critiqué par tous les participants. | * L’étudiant ne dispose pas d’encadrement et d’un suivi personnalisé tout au long de la formation. * Les MOOCS tendent à supprimer les interactions sociales entre les étudiants. * Les formations proposées peuvent avoir un caractère froid et impersonnel en raison du manque d’interaction. * Manque de motivation. |
| **Les classes virtuelle (ZOOM, TEAMS**  **, AdobeConnect, etc.)** | * La classe virtuelle offre également la possibilité d’être enregistrée ce qui permet à un étudiant absent de visualiser la classe à un autre moment. * Les apprenants n’ont pas besoin de se déplacer pour accéder à la formation. | * La nécessite d’être équipé d’un matériel adapté : une caméra, un micro de la qualité, d’une bonne connexion internet. * Les étudiants n’ont pas la chance de travailler en groupe. * La difficulté à capter et conserver l’attention des participants. |
| **Moodle** | * Open-source et gratuit quel que soit le nombre d’utilisateurs. * Possibilité de regrouper tous les cours d’un établissement sur une même plateforme. | * L’interface graphique n’est pas des plus intuitive. * Manque de motivation. * Défaut de contact direct avec le formateur. * Nécessite une personne pour paramétrer Moodle et assurer la maintenance. * Prévoir un budget également pour l’hébergement et les mises à jour, ainsi que l’assistance technique. |

**II-3-3- La solution proposée :**

L’étude l’existant nous a permis de dégager plusieurs anomalies que nous avons détaillées dans la section précédente. Pour faire recours à ces anomalies nous proposons de concevoir et d’implémenter une plateforme e-Learning pour poser et suivez les cours à distance.

**II-4-** **Spécification des exigences**

Dans cette section nous identifions une liste d’exigences fonctionnelles et non fonctionnelles du système à concevoir. Certaines exigences sont ajoutées pour clarifier d’avantage les besoins des utilisateurs.

**II-4-1- Etude des besoins fonctionnels :**

Les besoins fonctionnels représentent les attentes de chaque acteur de la future plate-forme. Toute solution conceptuelle doit satisfaire, préalablement, à des besoins fonctionnels afin de délimiter le périmètre fonctionnel de l‘application et surveiller la traçabilité des besoins lors de la phase de développement.

La présente plateforme doit satisfaire les besoins fonctionnels suivants :

* La gestion des utilisateurs.
* La gestion des cours.
* La gestion des devoirs (travail à faire : TAF).
* La gestion des tests.
* La gestion des communications.

**II-4-2- Etude des besoins non fonctionnels :**

Un besoin non fonctionnel est une restriction ou une contrainte qui pèse sur un service du système, telle les contraintes liées à l’environnement et à l’implémentation et les exigences en matière de performances.

Les besoins non fonctionnels :

* Le code doit être clair pour permettre des futures évolutions ou améliorations.
* La plateforme doit fournir un accès rapide aux informations, et doit faire la mise à jour en temps réel.
* La plateforme doit être portable, extensible, réutilisable et fiable.
* La plateforme offre une interface conviviale et facile à utiliser.
* La plateforme doit garantir la confidentialité, l’intégrité et la cohérence des données.

**Chapitre III : Modélisation Conceptuelle et Organisationnelle**

**III-1 : Modélisation Conceptuelle**

**III -1-1- Introduction**

La modélisation est une activité qui vise à élaborer [des modèles](https://carnets2psycho.net/dico/sens-de-modele.html), c'est-à-dire à transcrire des données d'observations et à décrire des processus dans un langage approprié et formalisé.

Cette section, sera présentée comme suit : nous commençons par le choix de la méthodologie de conception. Ensuite nous identifions les acteurs et les diagrammes de système.

**III -1-2- Choix de la méthodologie de conception :**

Dans la cadre de notre projet, nous avons opté pour le langage UML comme un langage de conception. UML (Unified Modeling Language, traduisez « langage de modélisation objet unifié ») fournit un moyen astucieux permettant de représenter diverses projections d'une même représentation grâce aux vues.  
Une vue est constituée d'un ou plusieurs diagrammes. On distingue deux types de vues :

* **La vue statique, permettant de représenter le système physiquement :**
* Diagrammes de classes : représentent des collections d’éléments de modélisation statiques (classes, paquetages...), qui montrent la structure d'un modèle.
* Diagrammes de cas d’utilisation : identifient les utilisateurs du système (acteurs) et leurs interactions avec le système.
* **La vue dynamique, montrant le fonctionnement du système :**
* Diagrammes de collaboration : montrent des interactions entre objet (instances de classes et acteurs).
* Diagrammes de séquence : permettent de représenter des collaborations en objets selon un point de vue temporel, on y met l'accent sur la chronologie (envois de messages).
* Diagrammes d’états-transitions : permettent de décrire les changements d’états d’un objet ou d'un composant, en réponse aux interactions avec d'autres objets/composants ou avec des acteurs.
* Diagrammes d’activités : (une variante des diagrammes d’états-transitions) servent à représenter graphiquement le comportement d’une méthode ou déroulement d’un cas d'utilisation.

La conception de l’interface a été élaborée en suivant la démarche suivante :

* L'élaboration des diagrammes de cas d'utilisation. Cette étape a été réalisée suite à la spécification fonctionnelle de l’application.
* Recensement des classes candidates et élaboration du diagramme des classes.
* Dresser les diagrammes de séquences pour mettre en évidence les interactions entre les différents objets du système.

**III -1-3- Les acteurs système :**

Un acteur représente l’abstraction d’un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié.

Les acteurs interagissant avec notre système sont :

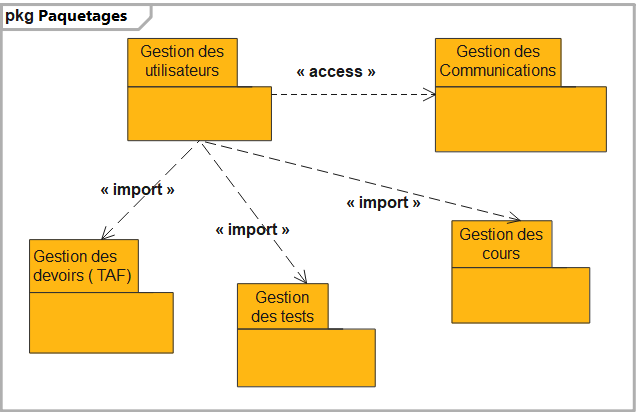
* + - **Administrateur :** c’est le responsable de l’administration du site**.**
    - **Enseignent :** élabore des cours, et propose des devoirs et des tests.
    - **Apprenant :** suivre un cours.

**III -1-4- Identification des cas d'utilisation :**

Nous décrivons pour chaque acteur les cas d’utilisation. On distingue les cas d’utilisation suivants :

|  |  |
| --- | --- |
| **Acteurs** | **Cas d’utilisation** |
| **Administrateur** | * Gérer son profil (mettre à jour ses informations.). * S’authentifier. * Gérer les utilisateurs (Ajouter, supprimer,etc.) * Gérer les filières (Ajouter,supprimer,etc.) |
| **Enseignent** | * Gérer son profil (mettre à jour ses informations, etc.). * S’authentifier. * Gérer ses cours (ajouter, modifier, gérer leur droit d’accès, etc.). * Gérer les devoirs (travaux à faire). * Gérer les tests (modifier, supprimer, consulter). * Se communiquer (répondre aux questions des apprenants). |
| **Apprenant** | * Gérer son profil (mettre à jour ses informations.). * S’authentifier. * Se communiquer (poser des questions). * Passer des tests. * Passer des devoirs. * Consulter les cours qui lui sont disponibles. * Télécharger un cours. |

**III -1-5- Diagramme de paquetages :**

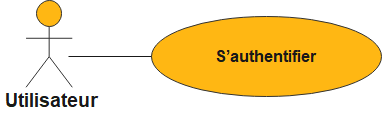


**Figure 1 :** **Diagramme de paquetages**

**III -1-6- Raffinement des cas d’utilisation :**

Dans le but de mieux comprendre notre système et les interactions avec les utilisateurs, dans cette partie nous allons détailler les principaux cas d’utilisation.

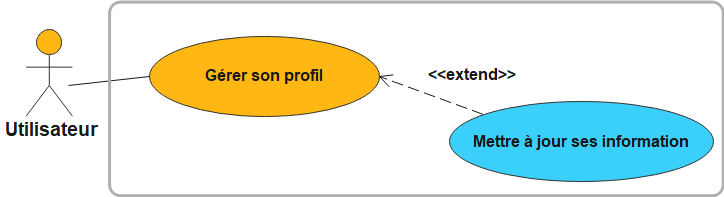
**III -1-6-1. Cas d’utilisation « S’authentifier »**

****

**Figure 2 :** **Cas d’utilisation « S’authentifier »**

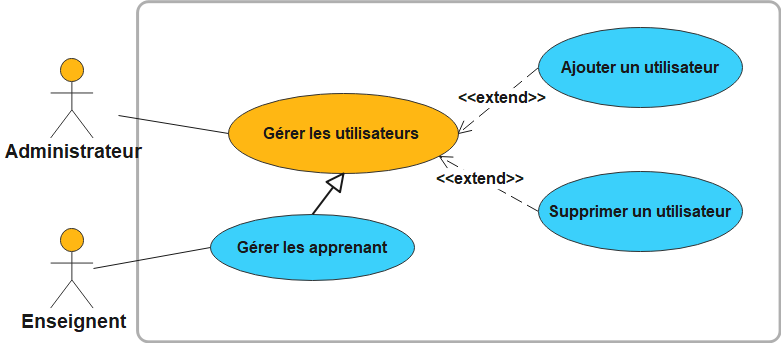
|  |  |
| --- | --- |
| **Description du cas <<S’authentifier>> :** | |
| **But** | Permet à l’utilisateur d’accéder à son espace personnel. |
| **Acteurs** | Administrateur, Enseignent, apprenant. |
| **Pré-condition** | L’acteur doit être un membre enregistré dans la plateforme. |
| **Scénario nominal** | Début  01 : Le système affiche un formulaire de connexion.  02 : L’acteur saisit son email ainsi que son mot de passe.  03 : Le système vérifie les paramètres.  04 : Le système ouvre l’espace de travail correspondant au profil.  Fin |
| **Scénario alternative** | Le login ou le mot de passe est incorrect : ce scénario commence au point 03 du scénario nominal.  01 : Le système informe l’acteur que les données saisies sont erronées et lui demande de saisir encore un fois les données. |

**III -1-6-2 Cas d’utilisation « Gérer le profil »**

 **Figure 3 :** **Cas d’utilisation « Gérer le profil »**

|  |  |
| --- | --- |
| **Description du cas <<Gérer le profil>>** | |
| **But** | Permet à l’acteur de de mettre à jour ses informations. |
| **Acteurs** | Administrateur, Enseignent, apprenant. |
| **Pré-condition** | L’acteur doit être un membre identifié. |
| **Scénario nominal** | Début  01 : Le système affiche le profil actuel de l’acteur.  02 : L’acteur met à jour ses informations.  03 : Le système vérifie la validité des informations saisies.  04 : Le système enregistre ces informations dans la base de données.  Fin |
| **Scénario alternative** | Les informations sont manquantes ou incorrectes : ce scénario commence au point 03 du scénario nominal.  01 : Le système informe l’acteur que les données saisies sont erronées, garde les informations saisies avant et le scénario reprend au point 02 du scénario nominal |

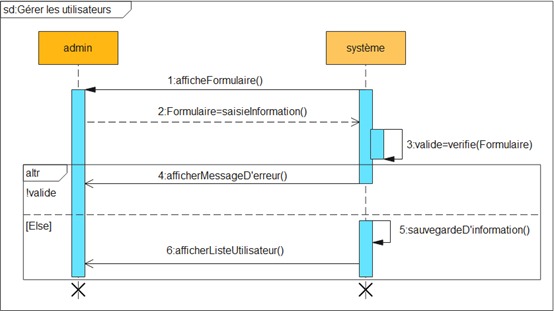
**III -1-6-3 Cas d’utilisation « Gérer les utilisateurs »**



**Figure 4 :** **Cas d’utilisation « Gérer les utilisateurs »**

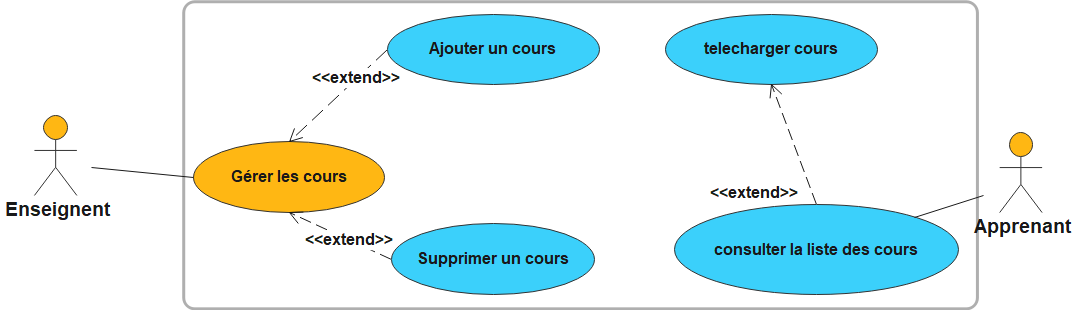
|  |  |
| --- | --- |
| **Description du cas<<gérer les utilisateurs>>** | |
| **But** | Permet à l’acteur d’ajouter et de supprimer un utilisateur. |
| **Acteurs** | Administrateur, Enseignent |
| **Pré-condition** | L’acteur doit être connecté. |
| **Scénario nominal** | Début  01 : Le système affiche un formulaire d’inscription à l’acteur.  02 : L’acteur saisit les informations du nouvel utilisateur et lui affecter un rôle.  03 : Le système vérifie la validité des informations saisies.  04 : Le système enregistre ces informations dans la base de données.  Fin |
| **Scénario alternative** | Les informations sont manquantes ou incorrectes : ce scénario commence au point 03 du scénario nominal.  01 : Le système informe l’acteur que les données saisies sont erronées et le scénario se reprend au point 02 du scénario nominal. |

**Diagramme de séquence :**

****

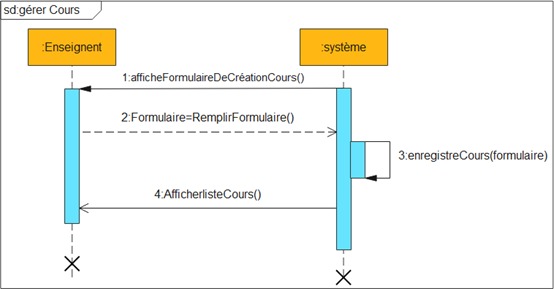
**Figure 5 :** **Diagramme de séquence « Ajouter utilisateur »**

**III -1-6-4 Cas d’utilisation « Gérer les cours »**

**Figure 6 :** **Cas d’utilisation « Gérer les cours »**

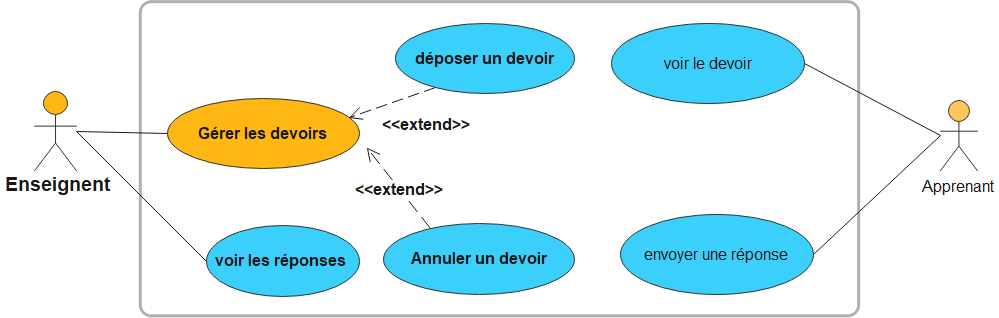
|  |  |
| --- | --- |
| **Description du cas :<<gérer les cours>>** | |
| **But** | Permet à l’acteur d’ajouter, d’annuler et modifier un cours. |
| **Acteurs** | Enseignent |
| **Pré-condition** | L’acteur doit être connecté. |
| **Scénario nominal** | Début  01 : Le système affiche un formulaire de création de cours.  02 : L’acteur saisie le contenu du cours  03 : L’acteur configure les droits d’accès au cours.  04 : L’acteur enregistre le cours dans la base de données.  Fin |

**Diagramme de séquence :**

****

**Figure 7 : Diagramme de séquence « Gérer les cours »**

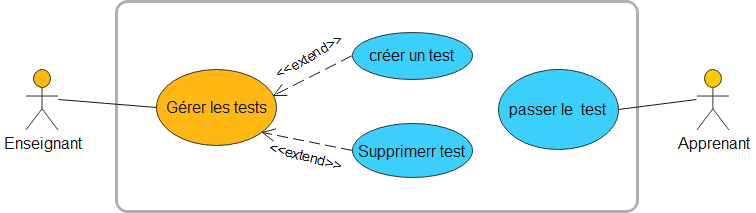
**III -1-6-5 Cas d’utilisation « Gérer les devoirs »**



**Figure 8 :** **Cas d’utilisation « Gérer les devoirs »**

|  |  |
| --- | --- |
| **Description du cas :<<gérer les devoirs>>** | |
| **But** | Permet à l’enseignant d’ajouter et supprimer des devoirs et permet à l’apprenant de voir les devoirs. |
| **Acteurs** | Enseignent, Apprenant |
| **Pré-condition** | L’acteur doit être connecté. |
| **Scénario nominal** | **Enseignant**  Début  01 : Le système affiche un formulaire de création de devoir.  02 : L’acteur saisie le contenu du devoir  03 : L’acteur enregistre le devoir dans la base de données.  Fin  **Apprenant**  01 : le système affiche la liste des devoirs.  02 : l’apprenant choisi un devoir.  03 : l’apprenant envoi la réponse.  Fin |

**III -1-6-6 Cas d’utilisation « Gérer les tests »**

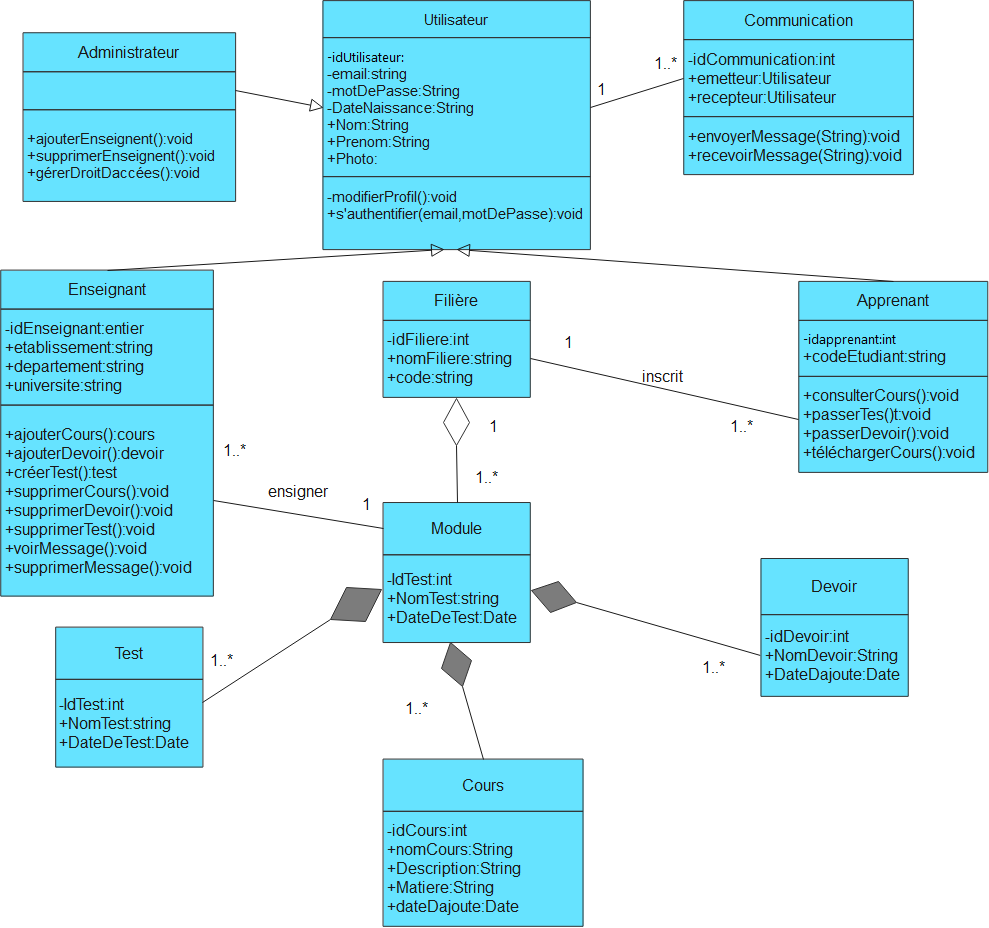


**Figure 9 :** **Cas d’utilisation « Gérer les tests »**

|  |  |
| --- | --- |
| **Description du cas :<<gérer les tests>>** | |
| **But** | Permet à l’acteur d’ajouter, d’annuler et modifier un test. |
| **Acteurs** | Enseignant |
| **Pré-condition** | L’acteur doit être connecté. |
| **Scénario nominal** | Début  01 L'acteur saisit le titre et lien de test (google forum ou autre plateforme)  02 : L’acteur enregistre le test dans la base de données.  Fin |

**III -1-7- Diagramme de classes :**

La figure ci-dessous représente le diagramme de classes qui contient toutes les informations telles que les classes, les méthodes, les associations et les propriétés.



**Figure 10 :** **Diagramme de classes**

**Représentation des associations entre les classes**

**Association simple :** les associations simples sont des liaisons logiques entre entités.  
Les cardinalités : précisent combien d’objets de classe considérée peuvent être liés à un objet de l’autre classe.

Le tableau suivant illustre les associations simples en indiquant leurs désignations, les classes participantes et leurs cardinalités

|  |  |  |
| --- | --- | --- |
| **Désignation** | **Classes participantes** | **Cardinalités** |
| Enseigner | Enseignant | Module | 1 |
| Inscrit | Apprenant | Filière | 1 |

**Généralisation/spécialisation :** C’est une relation d’héritage, dans laquelle les objets de l’élément spécialisé peuvent remplacer les objets de l’élément général.  
Le tableau suivant représente les associations de généralisation en indiquant la super classe et la sous classe.

|  |  |
| --- | --- |
| **Super-classe** | **Sous-classe** |
| Utilisateur | Enseignant |
| Utilisateur | Apprenant |
| Utilisateur | Administrateur |

**Agrégation :** Une agrégation est une association **non symétrique** dans laquelle l’une des deux classes joue un rôle prépondérant.

Le tableau suivant représente l’associations d’agrégation en indiquant l’agrégat et l’agrégé :

|  |  |
| --- | --- |
| **Agrégat** | **Agrégé** |
| Filière | Module |

**Composition :** La composition associe une classe composite et des classes parties, tel que tout objet partie appartient à un et un seul objet composite.

Le tableau suivant représente l’associations de composition en indiquant le composite et la classes partie :

|  |  |
| --- | --- |
| **Composite** | **Classe Partie** |
| Module | Test |
| Module | Cour |
| Module | Devoir |

**III-2 : Modélisation organisationnelle**

Dans ce qui suit,nous présentons le modèle de données optimisées.

**Utilisateur** (id\_utilisateu,nom,prenom,date\_ne,type,email,mot\_passe,photo,enligne)

**Filiere**(id\_filiere, code\_filiere,nom\_filiere)

**Enseignant**(id\_enseignant,description,site\_personnel,linkden ,departemment,

universite,etablissement,#id\_utilisateur)

**Apprenant**(id\_apprenant , code\_apprenant,#id\_utilisateur)

**Module**(id\_module, nom\_module,code\_module,#id\_filiere,#id\_enseignant)

**Suivre**(#id\_apprenant, #id\_module)

**Cours**(id\_cours,titre\_cours,fichier\_cours,video\_cours,type\_cours,date\_publication,

#id\_module)

**Devoir**(id\_devoirs ,titre\_devoirs,commentaire,fichier\_devoir,date\_finD,#id\_module)

**Test**(id\_test , titre\_test,date\_debut,date\_fin,lien\_test,#id\_module)

**Reponse**(#id\_apprenant, #id\_devoir,reponse)

**Messages**(id\_message,titre\_message,message,date\_message,vu,id\_recepteur,

id\_emetteur)

|  |  |  |
| --- | --- | --- |
| Nom Table | Liste des champs | Type et taille |
| **Utilisateur** | id\_utilisateur | int(10) |
| nom | varchar(50) |
| prenom | varchar(50) |
| date\_ne | date |
| type | tinitint(1) |
| email | varchar(320) |
| mot\_passe | varchar(100) |
| photo | lonblob |
| enligne | tinyint(1) |
| **Filieres** | id\_filiere | int(10) |
| code\_filiere | varchar(5) |
| nom\_filiere | varchar(500) |
| **Enseignant** | id\_enseignant | int(10) |
| description | text |
| site\_personnel | text |
| linkden | text |
| departemment | varchar(100) |
| universite | varchar(100) |
| etablissement | varchar(100) |
| #id\_utilisateur | int(10) |
| **Apprenant** | id\_apprenant | int(10) |
| code\_apprenant | char(10) |
| #id\_utilisateur | int(10) |
| **Module** | id\_module | int(10) |
| nom\_module | varchar(500) |
| code\_module | varchar(5) |
| #id\_enseignant | int(10) |
| #id\_filiere | int(10) |
| **Suivre** | #id\_apprenant | int(10) |
| #id\_module | int(10) |
| **Cours** | id\_cours | int(10) |
| titre\_cours | varchar(500) |
| fichier\_cours | longblob |
| video\_cours | longblob |
| type\_cours | enum(‘public’,’privé’) |
| date\_publication | int(10) |
| #id\_module | date |
| **Devoir** | id\_devoirs | int(10) |
| titre\_devoirs | Varchar(500) |
| commentaire | text |
| fichier\_devoir | longblob |
| date\_finD | date |
| #id\_module | int(10) |
| **Test** | id\_test | int(10) |
| titre\_test | Varchar(500) |
| date\_debut | datetime |
| date\_fin | datetime |
| lien\_test | text |
| #id\_module | int(10) |
| **Reponse** | #id\_apprenant | int(10) |
| #id\_devoir | int(10) |
| reponse | longblob |
| **Messages** | id\_message | int(10) |
| titre\_message | varchar(100) |
| message | text |
| date\_message | datetime |
| vu | Tinyint(4) |
| id\_emetteur | int(10) |
| id\_recepteur | int(10) |

**III-3 : Conclusion :**

Dans cette partie, nous avons réalisé la modélisation organisationnelle et logique de notre application. Cette modélisation nous a permis de dégager le modèle logique des données qui sera exploité lors de l’implémentation. Ce modèle sera transformé en modèle physique de données qui fera l’objet du chapitre suivant.

**Chapitre** **IV : Réalisation**

**IV-1- Introduction**

Après avoir élaboré la conception de notre application, nous abordons dans ce chapitre le dernier volet de ce rapport, qui a pour objectif d'exposer la phase de réalisation. Nous menons tout d’abord une étude technique où nous décrivons les ressources logicielles utilisées dans le développement de notre projet.

**IV-2- Choix des langages de développement, de SGBD et les Outils de développement**

**Java** est un langage de programmation informatique orienté objet. Java permet de développer des applications client-serveur. Côté client, les applets sont à l’origine de la notoriété du langage. C’est surtout côté serveur que Java s’est imposé dans le milieu de l’entreprise grâce aux servlets, et aux JSP





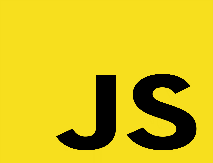
**JSP** est l’acronyme de Java Server Page. C’est une technologie java qui permet la génération des pages web dynamiques. La technologie JSP permet de séparer la présentation sous forme de code HTML et les traitements sous formes de classes.

La technologie JSP possède plusieurs avantages dont nous pouvons situer :

✓L'utilisation de Java par les JSP permet une indépendance de la plate-forme d'exécution mais aussi du serveur web utilisé.

✓La séparation des traitements et de la présentation : la page web peut être écrite par un designer et les tags Java peuvent être ajoutés ensuite par le développeur. Les traitements peuvent être réalisés par des composants réutilisables (des Java beans).  
✓Les JSP sont basées sur les servlets : tout ce qui est fait par une servlet pour la génération de pages dynamiques peut être fait avec une JSP.

Le **JavaScript** est un langage informatique utilisé dans le développement des pages web. Ce langage a la particularité de s'activer sur le poste client, Autrement dit, c’est votre ordinateur qui va recevoir le code et qui devra L’exécuter.



C'est en opposition à d'autres langages qui sont activé côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Firefox ou Internet Explorer.



**CSS** est l’acronyme de **C**ascading **S**tyle **S**heets, est un langage de feuille de style utilisé pour décrire la mise en forme d'un document écrit avec un langage de balisage. Il permet aux concepteurs de contrôler l’apparence et la disposition de leurs

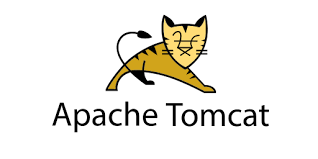
pages web.



**Eclipse** est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la fondation Eclipse visant à développer un environnement de production de logiciels libre qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java.

**MySQL** (My Structured Query Langage | Langage de requêtes structuré) est un système de gestion de bases de données relationnelles dédiées Open source. Il est très rapide, fiable et facile à utiliser et gratuit.





**Apache** Tomcat est un conteneur libre de servlets et JSP Java EE. Issu du projet Jakarta, c'est un projet principal de l’Apache Software Foundation. Il implémente les spécifications des

servlets et des JSP du Java Community Process1, est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.



**Microsoft** Word est un logiciel de traitement de texte couvre deux notions, assez différentes en pratique : Un éditeur de textes interactif et un compilateur pour un langage de mise en forme de textes.

**IV-3- Architecture**

L’architecture de notre application est de type client-serveur, où un ordinateur interagit avec d’autres sur Internet. Dans la réalisation de notre projet, nous avons opté pour une architecture MVC afin de garantir une assurance de la maintenabilité, la modularité de l’application et la rapidité de développement.

**MVC** littéralement Modèle Vue Contrôleur est une architecture qui organise l'interface Homme-Machine d'une manière à ce que le développement puisse se faire en couches indépendantes. Il impose la séparation entre les données, la présentation et les traitements, ce qui donne trois parties fondamentales dans l'application finale : le modèle de données, le contrôleur et la vue.  
**Couche Modèle**

Permet d'enregistrer les données, de les récupérer, de les lister, de les supprimer et de les mettre à jour.

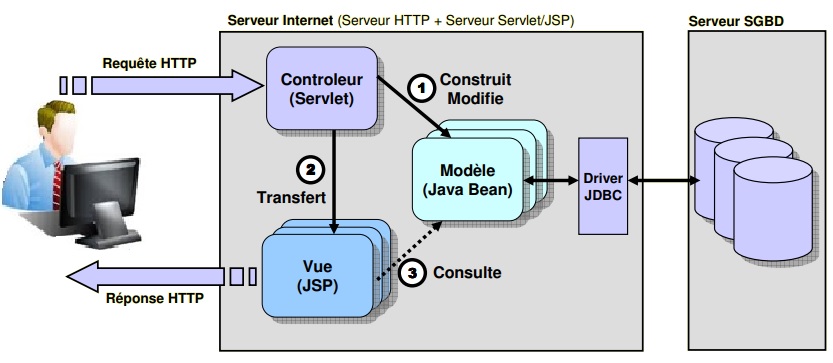
**Couche Vue**

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons, etc.). Ces différents événements sont envoyés au contrôleur, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur.

**Couche contrôleur**

Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle, et ce dernier notifie la vue que les données ont changée pour qu'elle se mette à jour.

La figure suivante représente l’architecture de notre application.



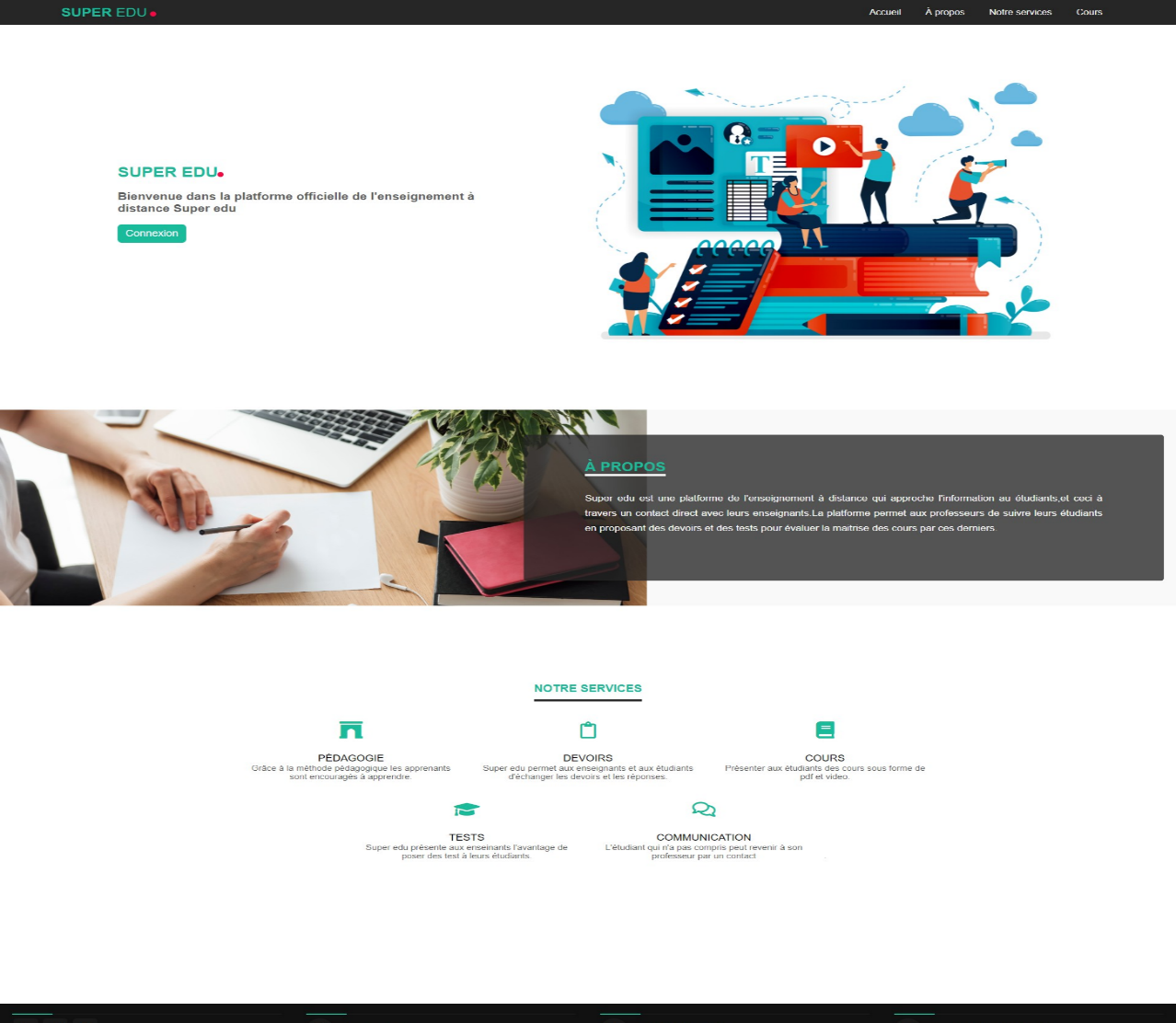
**Figure 11:** **Architecture 3-tiers**

⮚ **Couche Modele:** Les javaBeans permettent d'enregistrer les données, de les récupérer, de les lister, de les supprimer et de les mettre à jour.  
⮚ **Couche Vue :** Cette couche est en relation directe avec l’utilisateur (navigateur), elle est représentée par les pages JSP. Elle se met en relation direct avec la couche model sans ou avec l'intermédiaire de la couche contrôler.

⮚ **Couche Contrôler** : Les servlets jouent le rôle de la couche du contrôle.  
la servlet communique avec la couche modèle, pour remplir les demandes via des requêtes de l'utilisateur. Selon la requête, la servlet « aiguille » sur quel vue (JSP) est orienté l'utilisateur et avec quels attributs. Les attributs sont contenus dans les requêtes | réponses qui eux même contient des données prétraitées sous forme de JavaBeans (instances d'objet java), ou de type simple.

**IV- 4 - Présentation des interfaces**

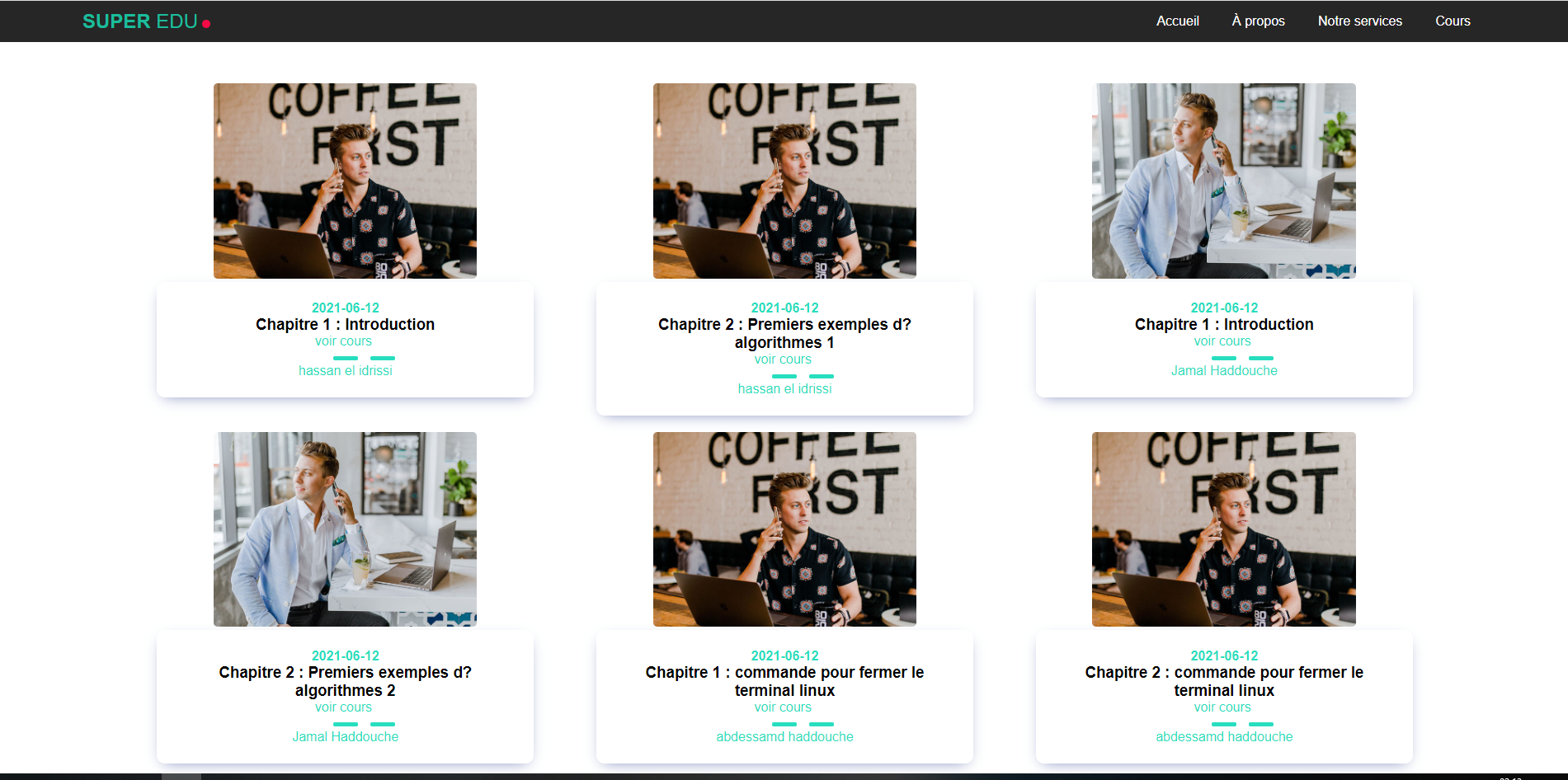
Après les phases d’étude de l’existant, la conception et la modélisation fonctionnelle et organisationnelle, nous avons développé les interfaces de notre application.  
Nous proposons dans cette section des interfaces graphiques de notre plateforme.



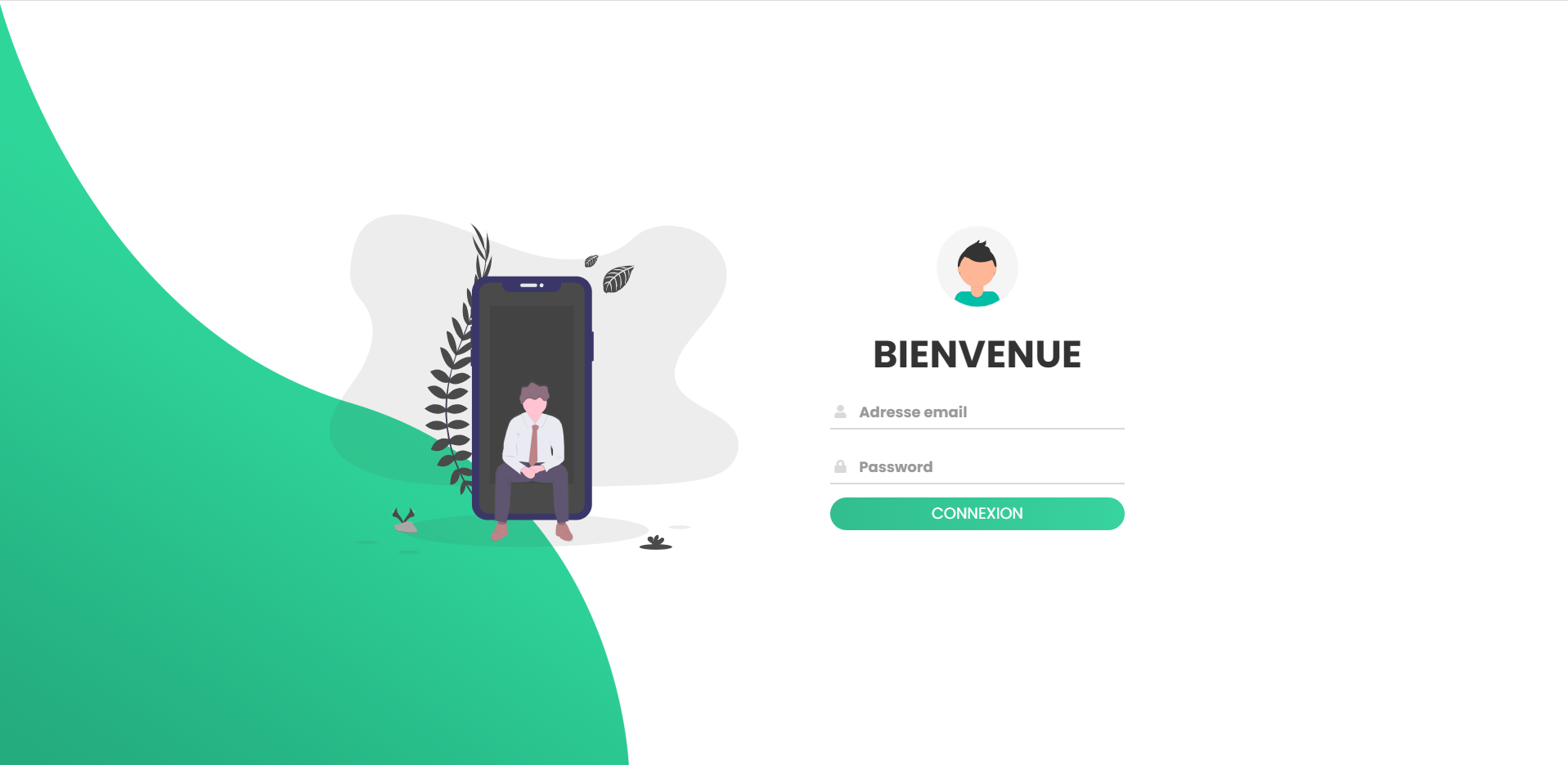
**Figure 12 :** **Page d’accueil**

**Accueil** : qui amène l’utilisateur à la page d’accueil de notre application.  
**COURS** : qui amène l’utilisateur à la liste des cours public proposés par les formateurs (tous les visiteurs du site peuvent voir ces cours).

**Connexion** : qui amène l’utilisateur à la page de connexion du site.



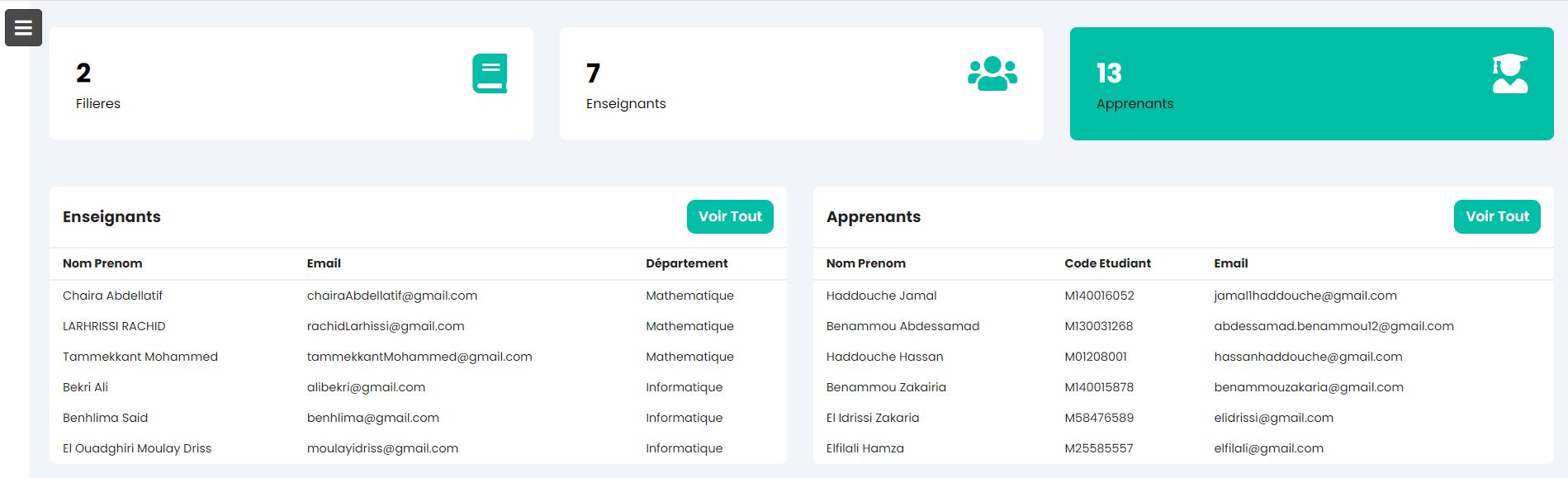
**Figure 13 :** **Page des cours publics**



**Figure 14 :** **Page de connexion**

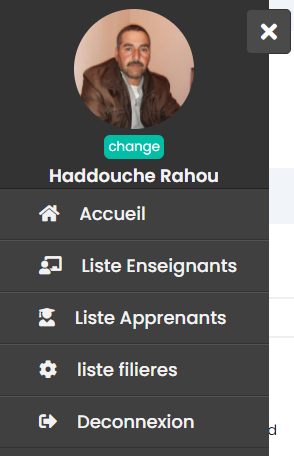
À partir de cette interface, un utilisateur (administrateur, enseignant et apprenant) peut se connecter. Il suffit d’entrer un email et un mot de passe et cliquer sur le bouton **« connexion »** pour ouvrir sa session. Lorsque ces informations de connexion sont validées, l’utilisateur sera redirigé vers l’interface de son profil.

**Profil Administrateur :**



**Figure 15 :** **Page d’administrateur**

Les deux tableaux affichent successivement les listes des enseignants et des apprenants inscrits dans le site avec leurs informations.



**Accueil** : qui amène l’administrateur à sa page d’accueil.

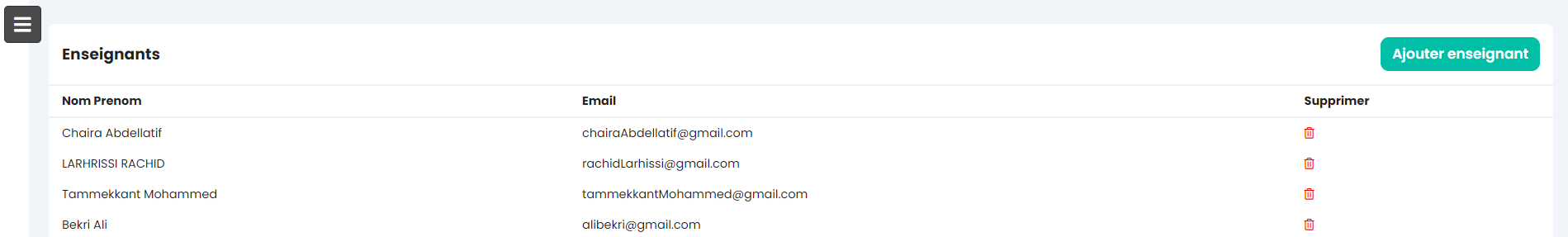
**Liste Enseignants** : affiche la liste des enseignants inscrits dans la plateforme.

**Liste Apprenants**: affiche la liste des apprenants inscrit dans la plateforme.

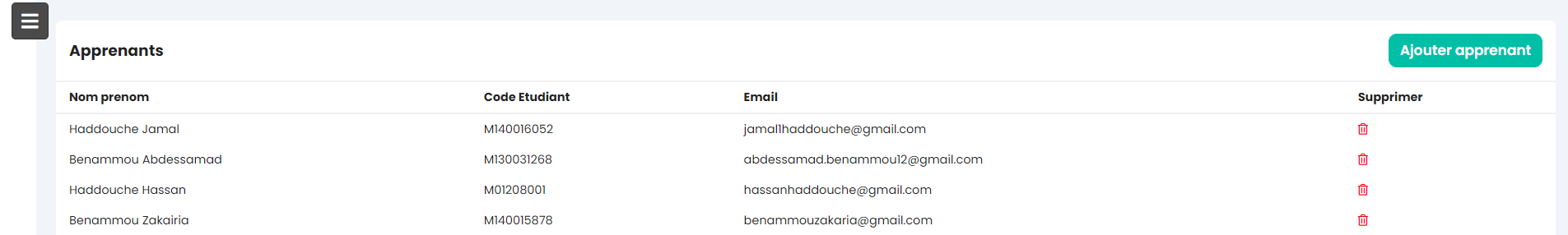
**Liste Filières :** affiche la liste des filières.

**Déconnexion** : permet de mettre fin à la session de l’administrateur.

**Figure 16 :** **Sidebar d’administrateur**



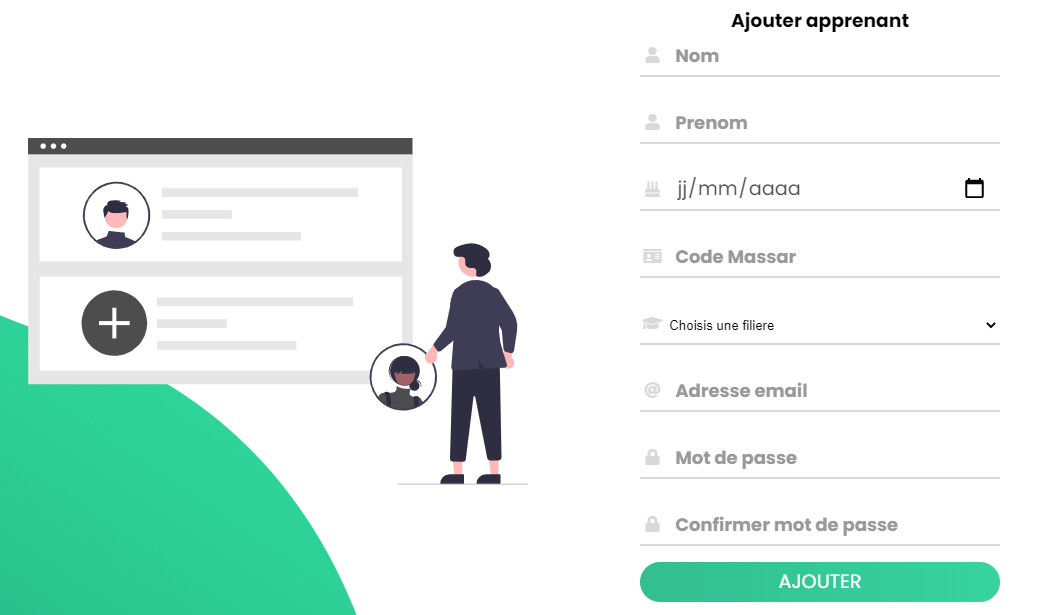
**Figure 17 :** **Liste des enseignants**



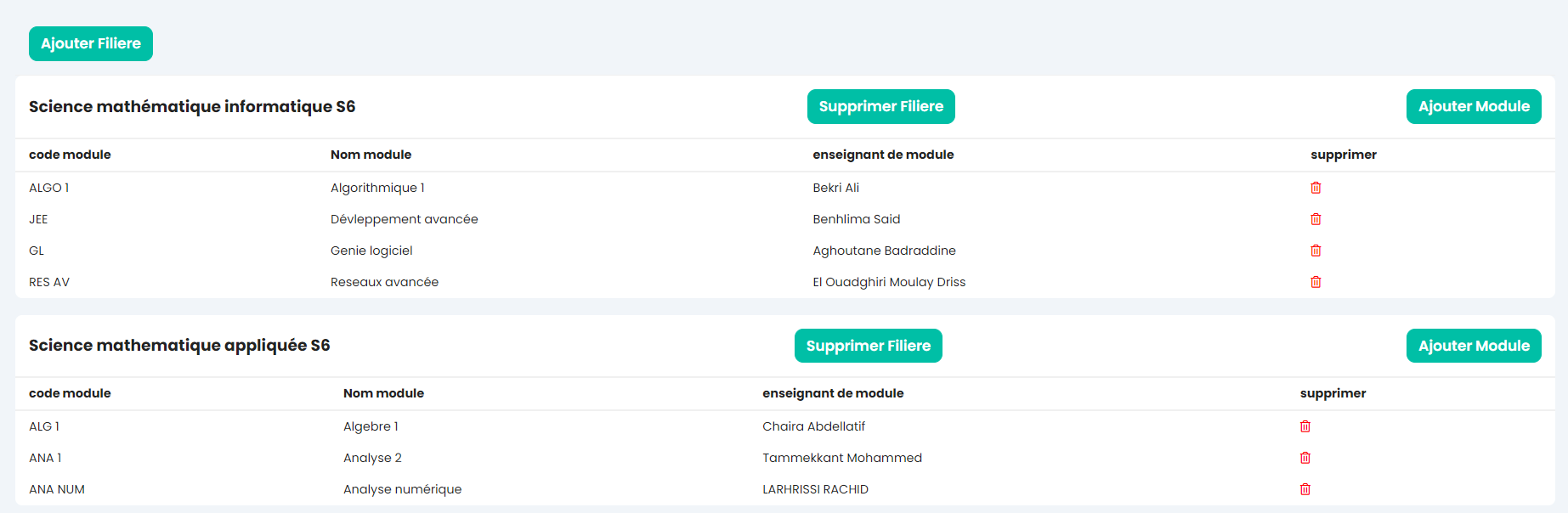
**Figure 18 :** **Liste des apprenants**

À partir de ces deux interfaces l’administrateur peut ajouter et supprimer des enseignants et des apprenants.





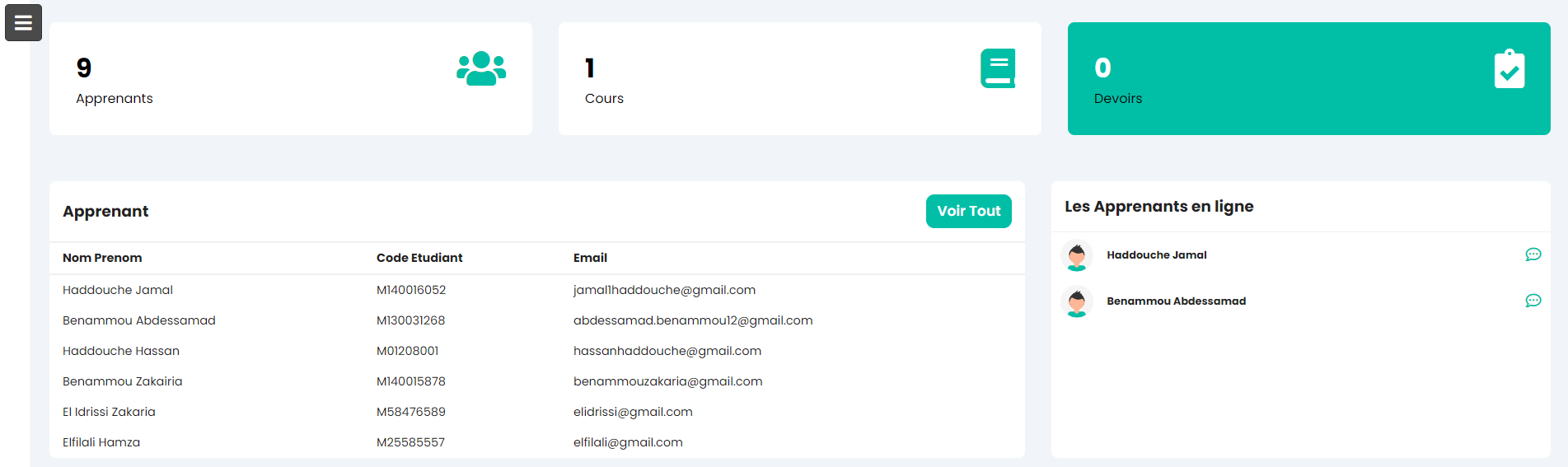
**Figure 19 :** **Formulaires pour saisir les informations de l’enseignant et de l’apprenant**



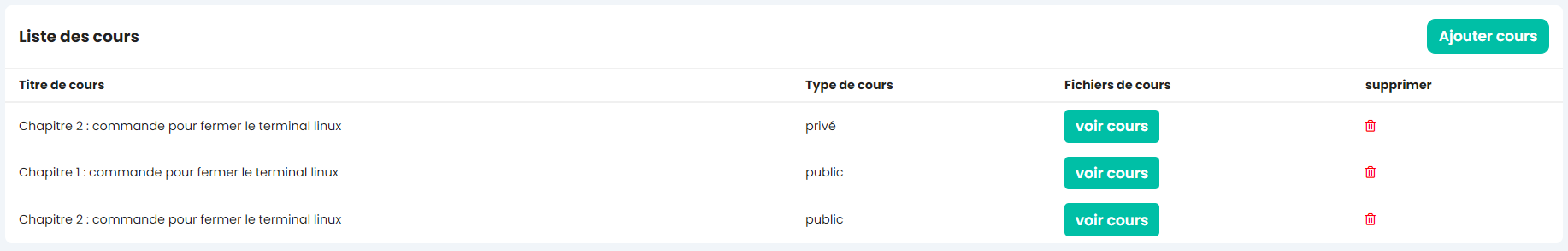
**Figure 20 :** **Liste des Filières**

À partir de ces deux interfaces l’administrateur peut ajouter et supprimer des filières et affecter des modules dans chaque filière.

**Profil enseignant :**



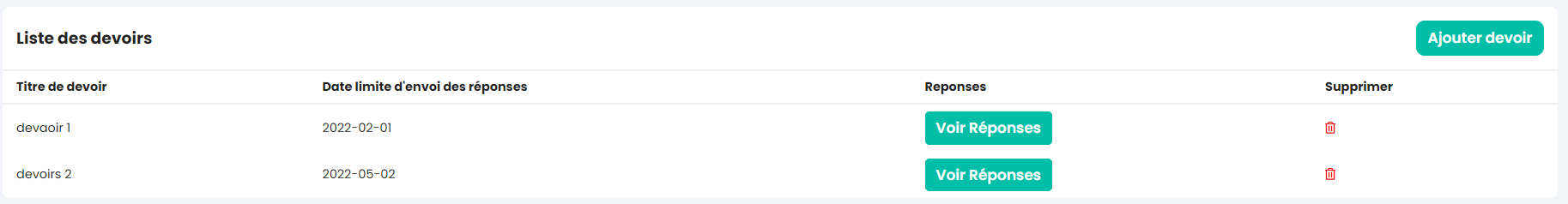
**Figure 21 :** **Profil enseignant**



**Figure 22 :** **liste des cours**

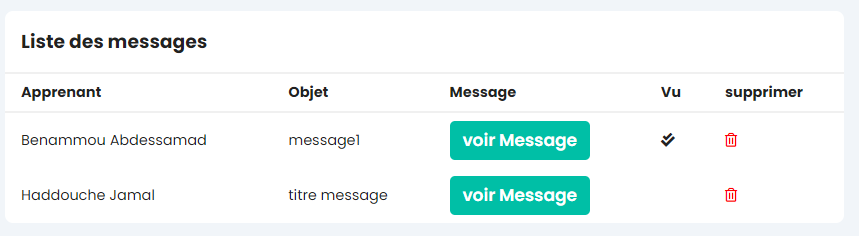


**Figure 23 :** **liste des tests**



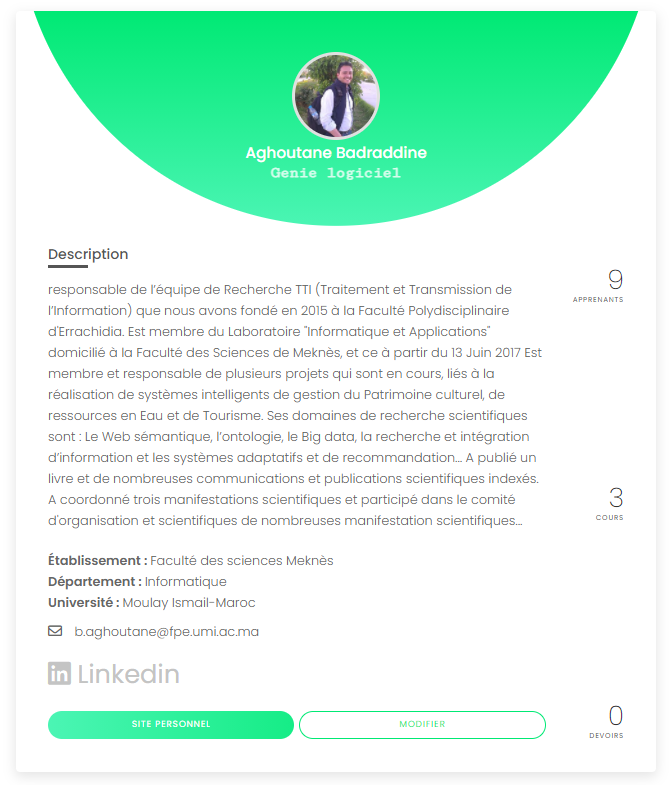
**Figure 24:** **liste des devoirs**

À partir de ces trois dernières interfaces l’enseignant peut ajouter, voir et supprimer des cours, des devoirs et des tests.



**Figure 25 :** **liste des messages**

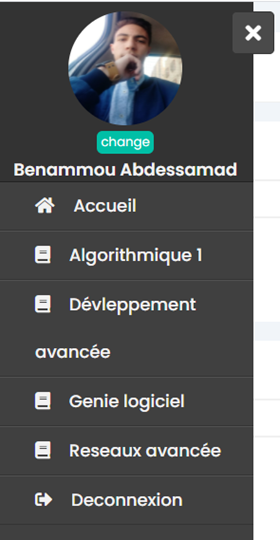
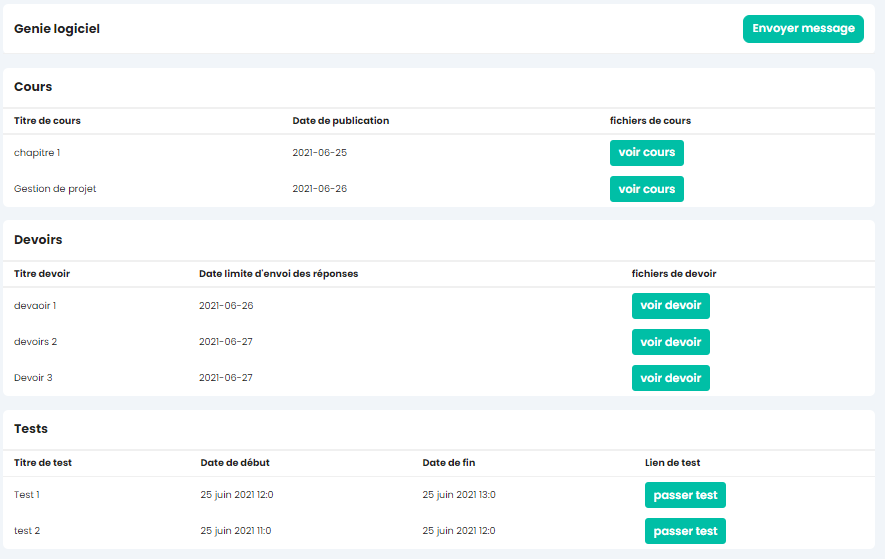
À partir de cette interface l’enseignant peut voir, répondre et supprimer les messages envoyés par ses apprenants.



**Figure 26 :** **carte enseignant**

À partir de cette interface l’enseignant peut voir et modifier ses informations personnelles (description, LinkedIn, site personnel, établissement, département, université).

**Profil apprenant :**

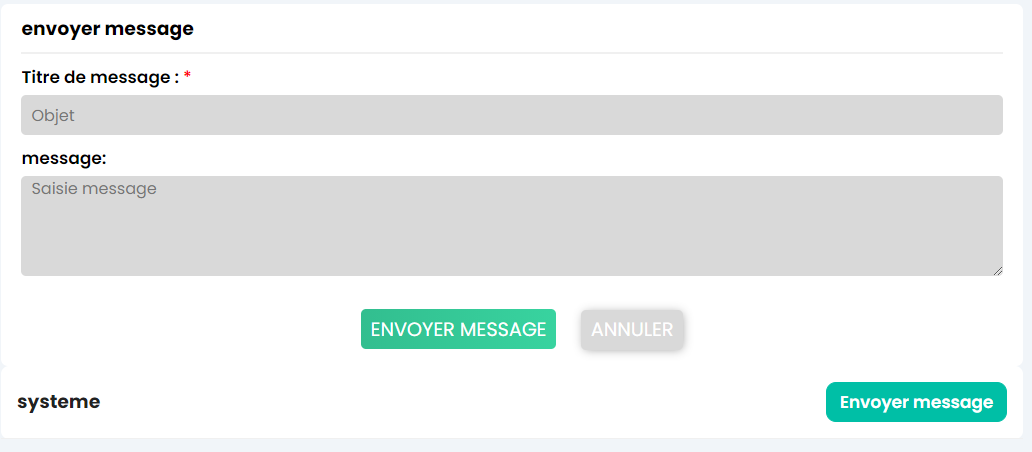
**** 

**Figure 27 :** **sidebare de l’apprenant Figure 28 :** **interface de cours**

À partir de cette « sidebare » l’apprenant peut changer sa photo de profil et de consulter la liste des modules. Lorsque l’apprenant choisit un module, l’interface à droite s’affiche.

Cette dernière se compose de quatre champs sont :

* **Le bouton « envoyer message »** : lorsque l’étudiant clique sur le Bouton, le formulaire ci-dessous s’affiche.



**Figure 29 :** **formulaire message**

* **Le tableau cours** : contient tous les cours (fichier PDF et vidéo) avec les dates de leurs publications.
* **Le tableau devoirs** : à partir de ce tableau l’apprenant peut voir ses devoirs et le dernier délai d’envoi des réponses. Par une clique sur Le bouton **«**voir devoir », l’interface ci-dessous s’affiche. À partir de cette interface il peut lire, télécharger et envoyer sa réponse.



**Figure 30 : Devoir**

* **Le tableau test** : à partir de ce tableau l’apprenant peut voir les dates des tests organisés par son enseignant et aussi accéder à ces tests.

**IV- 5 - Apports**

Ce projet de fin d’études nous a permis de se familiariser avec un certain nombre de  
concepts, tout en se basant sur nos connaissances acquises au cours de notre formation au sein de notre établissement.

Notre projet de fin d’étude a été une occasion, pour développer et exercer nos capacités d’observation, d’analyse, de conception, de développement et de rédaction.

**IV-5-1 : Apports au niveau des connaissances techniques**

Ce projet de fin d’études nous a permis de :

✓ Mettre en œuvre les notions et les connaissances acquises au sein des modules  
« Méthodologie de conception des bases de données ».

✓ Manipuler les logiciels MySQL et Eclipse au cours du développement de l’application.  
✓ Améliorer nos connaissances théoriques concernant la communication client / serveur.  
✓ Apprendre à gérer un projet.

**IV-5-2 : Apports au niveau de la conception et du développement**

Au niveau de la conception et du développement, cette étude nous a permis de :  
✓ Mener une conception orientée objet représentée avec le langage UML.

✓ Apprendre à maitriser la méthodologie de conception et de développement des  
applications clientes.

✓ Maitriser les étapes de développement et de conception d’une application Web.

**IV-6 Conclusion :**

Dans ce chapitre, nous avons pu présenter l’environnement et le processus de développement d’une application web. Nous avons exposé ainsi le résultat de développement, à l’aide des aperçus écran.

**Conclusion et perspectives**

L'objectif de notre projet de fin d’étude, est de concevoir et développer un site web dynamique d’enseignement à distance au profit des étudiants et des enseignants-chercheurs. Le présent manuscrit détaille la démarche et les étapes que nous avons suivies pour réaliser ce projet. En premier lieu, on a effectué une phase d’étude des différents types d'apprentissages et outils existants. En deuxième lieu, on a exposé la phase de l'analyse et l'étude des besoins fonctionnels et non fonctionnels. En troisième lieu, on a détaillé la conception et la modélisation de notre application. En dernier lieu, on a justifié les choix technologiques, les étapes de la réalisation de l'application ainsi que les résultats obtenus.

Ce projet était une occasion pour mettre en évidence et déployer sur le plan pratique nos connaissances en informatique. Toutes les fonctionnalités décrites dans le cahier des charges fonctionnelles ont été développées et validées. Néanmoins, ce projet pourra être amélioré par l’ajout d’autres fonctionnalités ou modules comme :

* La mise en place d’un forum : un espace de discussion publique formé de

catégories et de sous-forums permettant aux utilisateurs de discuter sur des sujets variés et classés par catégories.

Bibliographie

[1] Marcelle Parr, M. Éd - « Document préparé pour le Réseau d’enseignement francophone à distance du Canada (REFAD) » Mars 2019**.**

[2] Roger Lavoie Sébastien Richard « GUIDE SUR L’APPRENTISSAGE À DISTANCE » Août 2015

[3] Marie-Claude Riverin « ENSEIGNEMENT À DISTANCE GUIDE PRATIQUE » la fabrique, Produit par l’APOP – Octobre 2019.

[4] Tony Bates Ph. D - Associé de recherche « Les 10 fondamentaux de l’enseignement en ligne pour le personnel enseignant et de formation » Septembre 2016.

[5] Ilmar Slaidins « Guide de bonnes pratiques à l’usage des acteurs du e-learning» Novembre 2004.

[6] Arnal, Philippune (JDNet) « Solutions de e-learning : voyage dans la jungle des platesformes » Novembre 2003.

[7] Jibril Touzi, « Les LCMS et leurs rôles dans le e-learning », Atelier régional, Année 2007.

[8] Fernand Gervais (Vice‐président Enseignement et apprentissage) et Éric Martel, l'Université Laval « Cours en ligne ouverts et massifs -Massive open online course(MOOC) »

[9] Michel Bernard, Willy MEHEUST « La classe virtuelle : Quels apports dans les dispositifs de formation en ligne ? » Mémoire de MASTER 2 -Année universitaire 2011/2012.

[10] Sid Ahmed BENRAOUANE « Guide pratique du e-Learning - Stratégie, pédagogie et conception avec le logiciel Moodle » Dunod, Paris, 2011.

[11] TechSmith « Pour une transition efficace vers l’enseignement à distance - Le guide pratique » 2020.

Webographie

[1] <https://fr.wikipedia.org/wiki/Learning_management_system>

[2] <https://apolearn.com/solutions/lms/definition-lms/>

[3] <https://journals.openedition.org/dms/1047#tocto1n2>

[4] <https://www.digiforma.com/definition/mooc/>

[5] <https://www.digiforma.com/definition/classe-virtuelle/>

[6] <https://fr.wikipedia.org/wiki/Moodle>

[7] [https://openclassrooms.com/fr/courses/626954-creez-votre-application-web- avec-java-ee](https://openclassrooms.com/fr/courses/626954-creez-votre-application-web-%20avec-java-ee)