



Université de la Manouba  
École Nationale des Sciences de l'Informatique  
**RAPPORT DU**

**PROJET DE CONCEPTION ET DE DÉVELOPPEMENT**

---

**Sujet : Conception et développement d'un  
site de cours de management**

---

*Auteurs :*

M. Hammam BEN YOUNES  
M. Iskander BEN AMOR  
M. Ahmed Amine RIHANI

*Encadrante :*

Mme. Houda GUERMAZI

---

Année Universitaire :2021 /2022

---

# Remerciements

---

Nos remerciements les plus excessives sont adressés, au terme travail, à tous ceux qui ont participé pour que ce dernier puisse être réalisé.

Notre gratitude s'adresse à notre encadrante madame GUERMAZI Houda pour ses conseils et sa grande disponibilité, aussi bien que pour le personnel de l'ENSI pour leurs conseils judicieux et leurs soutiens permanents.

Nous voudrions en faire part de notre amabilité à tous nos collègues de l'ENSI.

---

# Table des matières

---

<b>Remerciements</b>	<b>2</b>
<b>Introduction générale</b>	<b>6</b>
<b>1 Etude préalable</b>	<b>7</b>
1.1 Notion du E-Learning . . . . .	7
1.2 Notion de la formation à distance . . . . .	7
1.3 Étude de l'existant . . . . .	7
1.4 Problèmes dégagés . . . . .	10
1.5 Solutions promises . . . . .	10
1.6 Conclusion . . . . .	11
<b>2 Analyse et Spécification</b>	<b>12</b>
2.1 Analyse des besoins . . . . .	12
2.1.1 Besoins fonctionnels . . . . .	12
2.1.2 Besoins non fonctionnels . . . . .	13
2.2 Spécification des acteurs et cas d'utilisation : . . . . .	13
2.2.1 Définition des acteurs . . . . .	13
2.2.2 Diagramme de cas d'utilisation . . . . .	13
2.2.3 Description détaillée des cas d'utilisation . . . . .	15
<b>3 Conception</b>	<b>18</b>
3.1 Conception Générale . . . . .	18
3.1.1 Choix de l'architecture de l'application . . . . .	18
3.1.2 Architecture du site . . . . .	21
3.1.3 Choix Technologiques . . . . .	21
3.2 Conception Détaillée . . . . .	22
3.2.1 Le Diagramme de séquences « S'authentifier » : . . . . .	22
3.2.2 Le Diagramme de séquences « Créer un profil » : . . . . .	23
3.2.3 Le diagramme de séquences « Consulter un cours » : . . . . .	25
3.2.4 Le Diagramme de Classe de l'Application . . . . .	25
3.3 Conception de la Base de Données . . . . .	27
3.3.1 Diagramme entité-association . . . . .	27
3.3.2 Description des associations . . . . .	29
3.3.3 Description des tables de la Base de données . . . . .	30
<b>4 Réalisation</b>	<b>31</b>
4.1 Environnement de travail . . . . .	31
4.1.1 Configuration matérielle . . . . .	31
4.1.2 Configuration logicielle . . . . .	31
4.2 Captures d'Ecran . . . . .	32
4.2.1 Pages Communes aux utilisateurs du site . . . . .	32
4.2.2 Les pages accessibles par l'administrateur . . . . .	34

4.2.3	Les pages accessibles par l'étudiant . . . . .	37
4.3	Chronogramme . . . . .	40
<b>Conclusion et perspectives</b>		<b>42</b>
<b>Netographie</b>		<b>43</b>

---

# Table des figures

---

1.1	logo de SAP Litmos [n3]	8
1.2	page login du site SUP'COM e-learning [n4]	8
1.3	logo de Udemty [n5]	9
1.4	logo de Study.tn [n6]	9
1.5	logo de AcurSchool [n7]	10
2.1	Diagramme de cas d'utilisation général.	14
2.2	Cas d'utilisation gérer cours.	15
2.3	Cas d'utilisation créer et gérer un profil.	16
2.4	Cas d'utilisation gérer les étudiants.	17
3.1	Architecture MVC [n8].	19
3.2	Architecture 4 couches [n9].	19
3.3	Architecture 4-tiers [n10]	20
3.4	Structure du code source	21
3.5	Diagramme de séquences « S'authentifier »	23
3.6	Diagramme de séquences « Créer un profil »	24
3.7	Diagramme de séquences « Consulter un cours »	25
3.8	Diagramme de Classes	26
3.9	Diagramme E/A de la base de données.	28
4.1	Page d'authenfication	32
4.2	Page d'inscription	33
4.3	Tableau de bord de l'admin	34
4.4	Page des notes	35
4.5	Page d'ajout des cours	36
4.6	Page d'ajout d'événements	36
4.7	Tableau de bord de l'étudiant	37
4.8	Page de cours	37
4.9	Page du contenu de cours	38
4.10	Page de calendrier	38
4.11	Page de groupes	39
4.12	Page de messagerie	39

---

# Introduction générale

---

La pandémie covid 19 aura changé nombres de nos habitudes dont nous citons notre façon de travailler qui a pris un tournant inattendu basé sur le «télétravail» et la formation à distance basée sur les technologies de l'information et de la communication.

L'enseignement à distance, également appelé formation à distance, enseignement en ligne ou e-learning, renvoie à une modalité d'enseignement qui «permet à une personne d'apprendre de façon relativement autonome, avec des contraintes minimales d'horaire et de déplacement, et avec le soutien à distance de personnes-ressources. En plus, apprentissage à distance est accessible où que l'on se trouve, en présence des autres participants ou non. L'éducation à distance s'est toujours développée en fonction des innovations technologiques [n1].

C'est dans ce cadre général que s'inscrit notre projet de conception et de développement qui consiste à l'implémentation d'un site web dynamique pour enseignement à distance. Afin de comprendre la démarche que nous avons utilisée pour mener ce projet à son terme, notre rapport se structure de la façon suivante :

Tout d'abord, dans un premier chapitre intitulée " Etude préalable " nous présentons les concepts ainsi que l'étude de l'existant et le cadre général de notre projet.

Ensuite, le deuxième chapitre intitulé " Analyse et spécification " permet de dégager les besoins de notre site web, qu'ils soient fonctionnels ou non, et de les modéliser par des diagrammes de cas d'utilisation suivant la méthode UML.

Le troisième chapitre intitulé " Conception " aura pour but de présenter les fonctionnalités du site et son architecture.

Le dernier chapitre intitulé " Réalisation ", précise l'environnement du travail et présente les principales interfaces du site.

Enfin, nous présentons la conclusion générale ainsi que les perspectives d'amélioration de notre application.

# CHAPITRE 1

---

## Etude préalable

---

### Introduction

Avant d'entamer le développement de notre projet "Site Web de cours de management", une bonne étude de la notion du E-learning, d'une étude profonde de l'existant ainsi qu'une solide étude sur les technologies Web sont la bienvenue pour maintenir notre clarté vis-à-vis le présent projet.

### 1.1 Notion du E-Learning

Le e-learning est un type de formation en ligne et à distance qui utilise l'Internet et les nouvelles technologies digitales, dans le but d'améliorer les processus d'apprentissage. Ce type d'enseignement permet aux apprenants de pouvoir se former à distance, quand ils le souhaitent et d'avoir accès à différents contenus pédagogiques 24h/24, 7J/7 et peu importe le lieu où ils se trouvent. Ils peuvent ainsi se former dans le domaine qui les intéresse, et ce, toujours à leur rythme.

### 1.2 Notion de la formation à distance

La formation à distance désigne l'ensemble des dispositifs de formation à distance, soit les cours par correspondance, les MOOC ou les formations en ligne. Elle s'applique tant à la formation continue qu'à la formation initiale, tant de manière individuelle que collective. Ces formations utilisent de plus en plus les outils numériques et les moyens de l'Internet.

Les formations à distance peuvent se caractériser par leur flexibilité qui permet de les désigner comme « ouvertes », dans le sens où elles ne requièrent aucun prérequis autre que technique. L'apprenant peut alors gérer, de manière autonome, le temps consacré à son apprentissage et peut choisir d'entrer ou de sortir librement d'un dispositif[n2].

### 1.3 Étude de l'existant

Il existe plusieurs plateformes d'enseignement en ligne sur le marché, nous proposons de présenter quelque plateforme mais la liste n'est pas exhaustive.

## SAP Litmos :



FIGURE 1.1 – logo de SAP Litmos [n3]

Le système de gestion de l'apprentissage SAP Litmos, généralement appelé LMS, est une plate-forme qui alimente la formation pour l'entreprise intelligente d'aujourd'hui. Elle est créée en 2007 et utilisée par plus de 30 millions de personnes dans 150 pays, dans 35 langues.. C'est le moteur de certains des programmes de formation les plus sophistiqués au monde avec la capacité de s'adapter à des millions d'utilisateurs, un ensemble de fonctionnalités de nouvelle génération et une interface connue pour sa simplicité simplifiée et son incroyable facilité d'utilisation.

Cette plateforme est facile à utiliser, comporte un système d'évaluation bien établi et fournit des solutions robustes pour automatiser les tâches difficiles. Parmi les points faibles de cette plateforme, le prix qui est très élevé et l'environnement est destiné pour les entreprises uniquement.

## SUP'COM e-learning :

FIGURE 1.2 – page login du site SUP'COM e-learning [n4]

SUP'COM e-learning est une solution proposée par l'école supérieure des communications de Tunis ( Sup'com), réalisé avec Moodle. Elle est une plate-forme d'apprentissage en ligne qui permet de créer des communautés s'instruisant autour de contenus et d'activités destinées exclusivement aux étudiants de Sup'com.

Cette plateforme est personnalisable et flexible, facile à apprendre et à naviguer.

Mais d'autre part, l'interface et les rapports sont basiques et nécessitent de nombreuses configurations manuelles entre différents groupes d'apprentissage.



**Udemy :**



FIGURE 1.3 – logo de Udemy [n5]

Udemy est un site internet de formation en ligne à destination des adultes et des étudiants. Il est en ligne depuis mai 2010. En janvier 2020, le site compte plus de 50 millions d'étudiants et 57 000 cours en 65 langues.

L'entreprise est basée à San Francisco.

Udemy se décrit comme « le principal marché mondial de l'apprentissage et de l'enseignement ». Ils donnent accès à des milliers de cours produits par des instructeurs indépendants à des millions d'étudiants. Elle donne l'accès à des cours de formation pour les particuliers, les entreprises et fournit une plate-forme permettant aux instructeurs et aux experts d'héberger leur propre contenu de cours.

Malgré son succès, le site présente des lacunes comme le manque d'interactivité entre l'enseignant et les étudiants et l'absence d'outils de gestion des étudiants.

**Studytn :**



FIGURE 1.4 – logo de Study.tn [n6]

Studytn est une plate-forme en ligne installée en Tunisie permettant les instructeurs de déposer leur contenu et aux étudiants à apprendre des compétences dans des plusieurs domaines (Marketing, Business, Programmation).

Le site comporte des problèmes comme l'absence d'interactivité entre l'enseignant et les étudiants, le regroupement des étudiants n'est pas considéré et les quiz sont mal désignés.

## AcurSchool :



FIGURE 1.5 – logo de AcurSchool [n7]

AcurSchool est un logiciel de gestion scolaire tunisien multiplateformes (Web, Mobile, Desktop) et multi-utilisateurs (gestionnaire, enseignant, étudiant, parent). Composé d'un système de gestion et d'une plateforme d'enseignement en ligne.

### Tableau récapitulatif

Fonctionnalités	SAP Litmos	SUP'COM e-learning	Udemy	Studytn	AcurSchool
Accès aux cours	✓	✓	✓	✓	✓
Affichage des notes	✗	✓	✓	✗	✗
Quiz en ligne	✗	✗	✓	✓	✗
Messagerie	✗	✓	✗	✗	✗
Examens en ligne	✓	✗	✗	✗	✓

## 1.4 Problèmes dégagés

L'étude préalable a mis en évidence des lacunes dont souffrent les plateformes mentionnées ci-dessus. Parmi les lacunes nous citons le manque d'une messagerie dans certaines plateformes, pas d'affichage de notes et de la possibilité de passer des examens en ligne selon le cours choisi... Ainsi nous avons opté à créer notre propre solution qui répondra à tous ces besoins.

## 1.5 Solutions promises

Dans cette perspective, la réalisation d'une plateforme en ligne qui sert à bien former les jeunes étudiants dans le domaine de management et suivre leur progression tout au long le processus de formation est donc indispensable pour répondre aux besoins de notre problématique. En effet, notre travail reposera sur les résultats d'une étude comparative et évaluative bien détaillée des plateformes. Cela nous aidera à illustrer notre projet, que nous envisageons de mettre en œuvre pour faciliter l'accès aux formations. Nous avons intégré les différentes

fonctionnalités dans notre application comme la possibilité de passer des examens en ligne, la messagerie, organiser les groupes d'apprentissage et les classer selon leur niveau, la possibilité de proposer un sondage...

C'est pour cela notre solution représentera une alternative des plateformes e-learning existantes vu qu'elle contient plusieurs fonctionnalités que les autres en manquent.

## 1.6 Conclusion

Ce chapitre englobe l'étude préalable de notre projet. Nous avons en premier lieu consulté les différentes plateformes de e-learning déjà disponibles, ensuite relevé leurs problèmes et enfin nous avons proposé notre solution. Objectifs éclaircis, nous spécifierons dans le chapitre suivant clairement les fonctionnalités de notre système.

# CHAPITRE 2

---

## Analyse et Spécification

---

La phase d'analyse et spécification des besoins est nécessaire pour bien réaliser la conception.

En effet une description détaillée des besoins sert à faciliter la phase de développement du travail et la communication entre le développeur et le client.

### 2.1 Analyse des besoins

Dans cette étape on va déterminer les différents besoins que notre application doit satisfaire.

#### 2.1.1 Besoins fonctionnels

L'application doit obligatoirement répondre à l'ensemble des besoins fonctionnels suivants :

*Enseignant :*

- Gérer les documents d'enseignements (Cours pdf, lien vidéo)
- Consulter la liste et les statistiques d'étudiants
- Attribuer des examens et des quiz aux étudiants
- Noter les examens
- Marquer les présences et les absences
- Créer des sondages
- Publier des nouveautés / évènements

*Étudiant :*

- Consulter les supports de cours
- Passer des examens, faire des quiz et consulter ses notes
- Consulter les nouveautés / évènements
- Répondre aux sondages
- Échanger des messages

### 2.1.2 Besoins non fonctionnels

*Notre application doit répondre aux besoins suivants :*

**Facilité d'usage :** Temps d'entraînement court.

**Portabilité :** Fonctionne sur navigateur web pour la plupart des systèmes d'exploitation.

**Ergonomie et convivialité :** L'application doit offrir aux différents utilisateurs (administrateur, étudiants) une interface conviviale.

**Sécurité :** Afin de sécuriser l'accès à l'application à chaque session d'affectation de modules, un ensemble de couples (login, mot de passe) est généré pour être par la suite distribué aux étudiants.

**Fiabilité :** Taux d'échecs très bas : Le serveur d'application doit être capable de gérer un grand nombre d'accès et de requêtes simultanées.

**Contrainte d'évolution :** Notre application doit permettre une maintenance facile et être susceptible à évoluer.

## 2.2 Spécification des acteurs et cas d'utilisation :

Dans cette étape, nous allons, au moyen de différents diagrammes, modéliser notre application moyennant les cas d'utilisation qui permettent de structurer les interactions entre les utilisateurs, appelés acteurs et le système. De plus, ces diagrammes permettent de relier les actions faites par un utilisateur avec les réponses attendues du système.

### 2.2.1 Définition des acteurs

Les acteurs de notre application sont :

**L'enseignant :** Le responsable de la formation.

**L'étudiant :** Toute personne voulant participer à la formation en management.

### 2.2.2 Diagramme de cas d'utilisation

Dans ce qui suit, nous présentons les diagrammes de cas d'utilisation qui sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

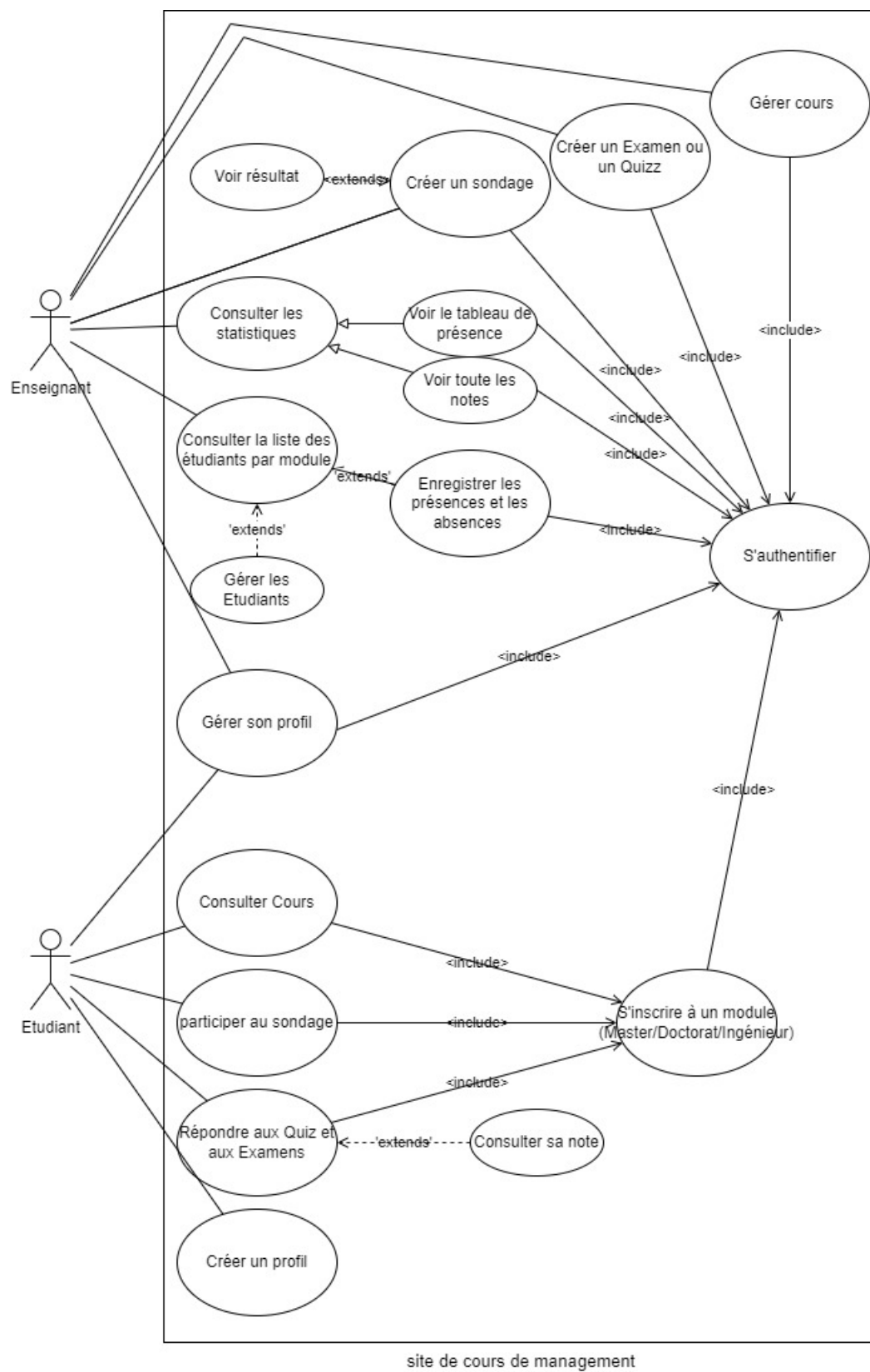


FIGURE 2.1 – Diagramme de cas d'utilisation général.

### 2.2.3 Description détaillée des cas d'utilisation

Dans cette partie nous détaillons quelques cas d'utilisation figurant dans le diagramme général.

#### Cas d'utilisation Gérer cours :

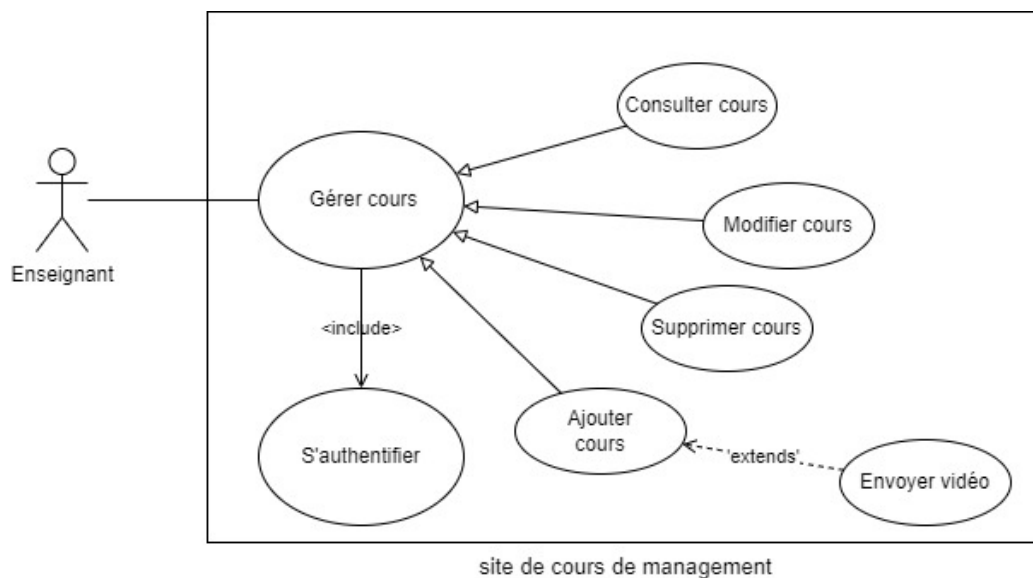


FIGURE 2.2 – Cas d'utilisation gérer cours.

Sommaire d'identification	
Titre	Gérer cours
But	L'Enseignant peut gérer ses cours
Résumé	L'Enseignant fournit un cours à un groupe et peut le modifier ou le supprimer
Acteur	Enseignant
Description des enchaînements	
Pré condition	Post condition
L'enseignant est authentifié	Gérer cours

## Cas de Gérer son profil et créer un compte :

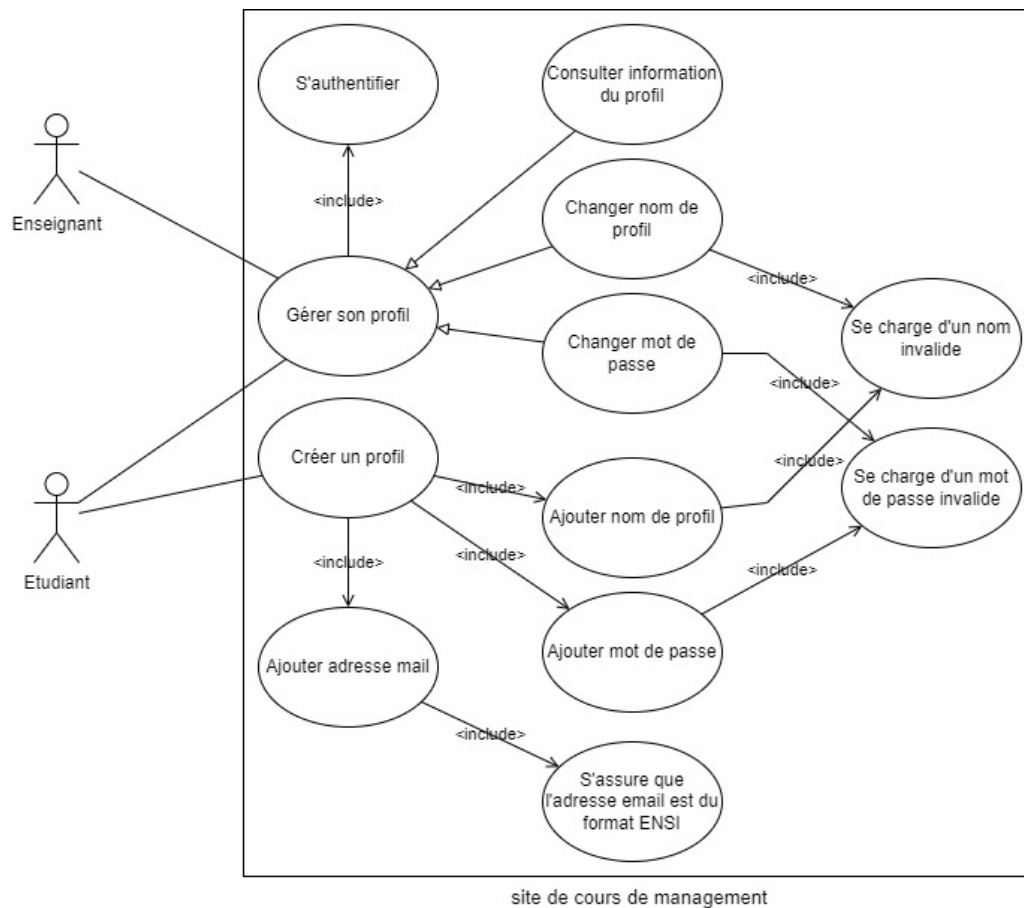


FIGURE 2.3 – Cas d'utilisation créer et gérer un profil.

Sommaire d'identification				
Titre	Gérer cours			
But	L'utilisateur peut Gérer son profil			
Résumé	L'utilisateur peut modifier son nom de profil son mot de passe et consulter ses informations s'il le souhaite			
Acteur	Etudiant		Enseignant	
Description des enchainements				
	Pré condi- tion	Post condi- tion	Pré condi- tion	Post condi- tion
	L'Etudiant est authenti- fié	L'Etudiant est inscrit	L'Enseignant est authenti- fié	Gérer son profil



Sommaire d'identification	
Titre	Créer un profil
But	L'Etudiant peut s'inscrire au site
Résumé	L'Etudiant fournit les données nécessaires
Acteur	Etudiant
Description des enchainements	
Pré condition	Post condition
Accéder au site web	E-mail valide

### Cas de Gérer comptes d'Étudiants :

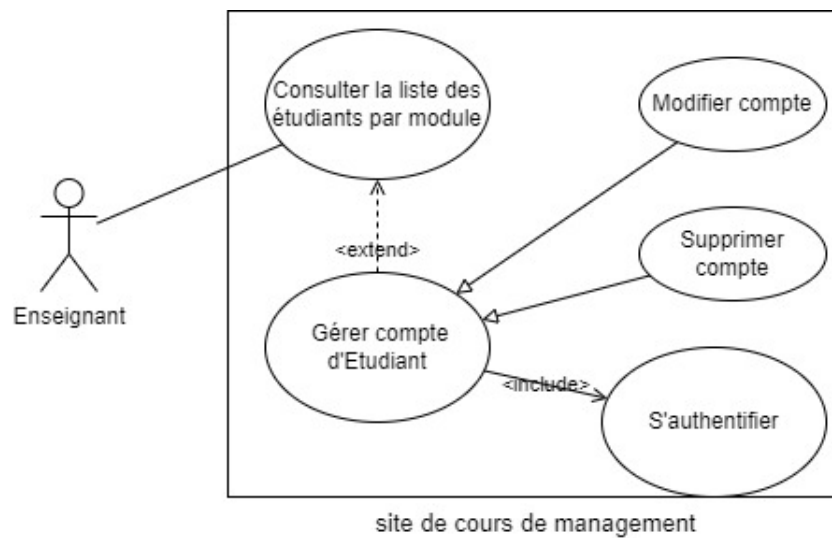


FIGURE 2.4 – Cas d'utilisation gérer les étudiants.

Sommaire d'identification	
Titre	Gérer compte d'Etudiant
But	L'Enseignant peut gérer les comptes inscrits
Résumé	L'Etudiant à l'aide d'une adresse e-mail ENSI fournit les données nécessaires
Acteur	Etudiant
Description des enchainements	
Pré condition	Post condition
L'enseignant est authentifié	L'utilisateur est inscrit

## Conclusion

Ce chapitre nous a permis de bien présenter les différents cas d'utilisations qui concernent l'administrateur et les étudiants et de définir les besoins fonctionnels et non fonctionnels à considérer afin de satisfaire les utilisateurs de notre application. L'analyse et spécification étant établies nous essayerons dans le chapitre suivant de élaborer avec détails l'architecture de notre système.

# CHAPITRE 3

---

## Conception

---

Les phases d'étude de l'existant et d'analyse ont abouti à une description complète du système étudié. A partir de l'analyse des exigences du projet, les besoins fonctionnels et non fonctionnels ont été fixés. Ces besoins décrivent les responsabilités que le système à développer doit satisfaire ainsi que les caractéristiques des fonctionnalités. A cette étape, il est nécessaire de concevoir et modéliser l'architecture générale de l'application à réaliser. C'est la phase de conception.

### 3.1 Conception Générale

Dans la suite, nous allons faire une description de l'architecture générale de l'application et une synthèse des différents modules de celle-ci.

#### 3.1.1 Choix de l'architecture de l'application

Pour notre application, le motif de conception à suivre est le motif MVC représenté dans la figure 3.1, c'est un schéma de programmation qui prend en compte toute l'architecture d'un programme et classe les différents types d'objets qui composent l'application dans 3 catégories :

- 1. Le modèle :** Le modèle représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il assure la gestion des données manipulées par l'application et garantit leur intégrité. Offre des méthodes pour mettre à jour ces données (insertion, suppression, changement de valeur). Il offre aussi des méthodes pour récupérer ces données. Les résultats renvoyés par le modèle sont dénués de toute présentation.
- 2. La 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...). Ces différents événements sont envoyés au contrôleur. La vue n'effectue aucun traitement.
- 3. Le 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 analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondante à la demande. 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 ensuite avertit la vue que les données ont changé pour qu'elle se mette à jour.

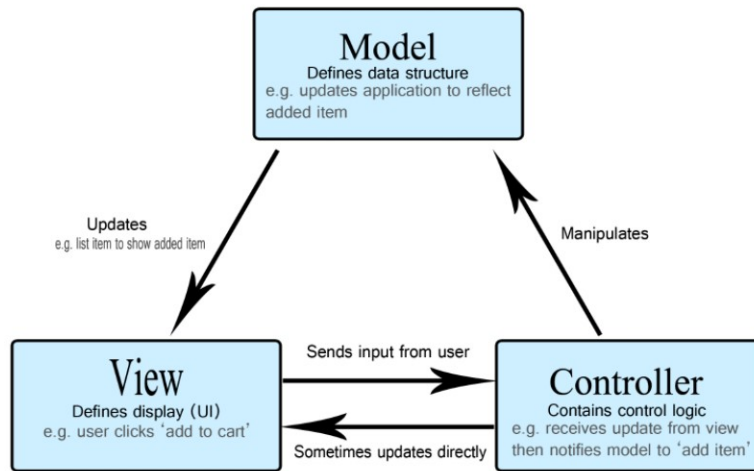


FIGURE 3.1 – Architecture MVC [n8].

Dans le cas de notre application, on compte 4 couches superposées les unes sur les autres, comme le montre la figure 3.2 et dont le rôle de chacune est défini comme suit :

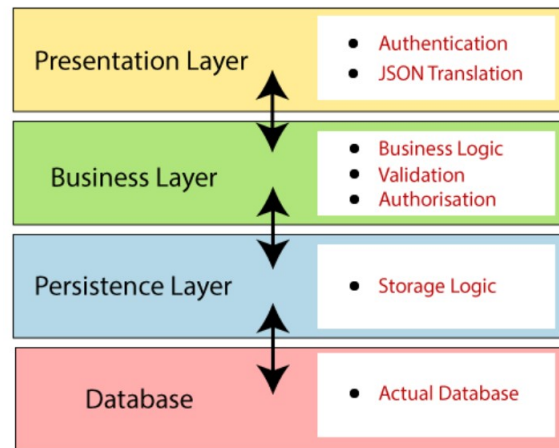


FIGURE 3.2 – Architecture 4 couches [n9].

**Couche de présentation :** la couche de présentation gère les requêtes HTTP, traduit le paramètre JSON en objet, authentifie la requête et la transfère à la couche métier. En bref, elle se compose de vues, c'est-à-dire d'une partie frontale.

**Couche métier :** la couche métier gère toute la logique métier. Elle se compose de classes de services et utilise des services fournis par des couches d'accès aux données. Elle effectue également l'autorisation et la validation.

**Couche de persistance :** la couche de persistance contient toute la logique de stockage et traduit les objets métier depuis et vers les lignes de la base de données.

**Couche de base de données :** dans la couche de base de données, les opérations CRUD (créer, récupérer, mettre à jour, supprimer) sont effectuées.

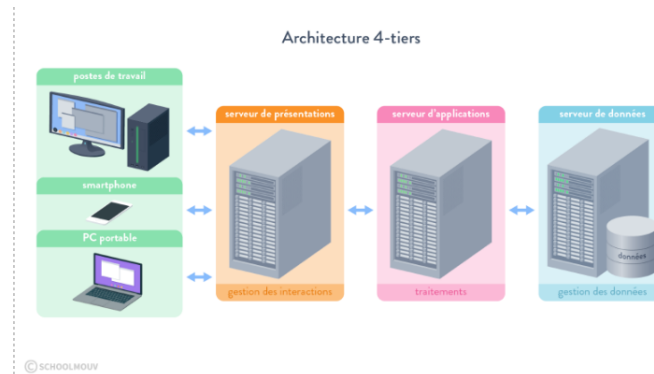


FIGURE 3.3 – Architecture 4-tiers [n10]

Notre application web d'apprentissage en ligne repose sur une architecture 4-tiers. Ce modèle d'architecture présente plusieurs avantages à savoir :

1. Les requêtes clients vers le serveur sont très flexibles.
2. Cette flexibilité permet à une entreprise d'envisager une grande souplesse pour l'introduction de toutes nouvelles technologies.
3. D'un point de vue développement, la séparation qui existe entre le client, le serveur et le SGBD permet une spécialisation des développeurs sur chaque tiers de l'architecture.
4. Plus de flexibilité dans l'allocation des ressources ; la portabilité du tier serveur permet d'envisager une allocation et ou modification dynamique au gré des besoins évolutifs au sein d'une entreprise.

*Dans le cas de notre application les 4-tiers sont :*

### Tier Client

**Tier Serveur d'application :** c'est un logiciel d'infrastructure offrant un contexte d'exécution pour des composants applicatifs.

**Tier Serveur de base de données :** c'est un serveur qui utilise une application de base de données qui fournit des services de base de données à d'autres programmes informatiques ou à des ordinateurs, tels que définis par le modèle client-serveur.

**Tier Serveur de messagerie :** c'est un logiciel serveur de courrier électronique. Il a pour vocation de transférer les messages électroniques d'un serveur à un autre.

### 3.1.2 Architecture du site

Ci-dessous une vue globale sur la structure du code source de la partie métier de notre site web.

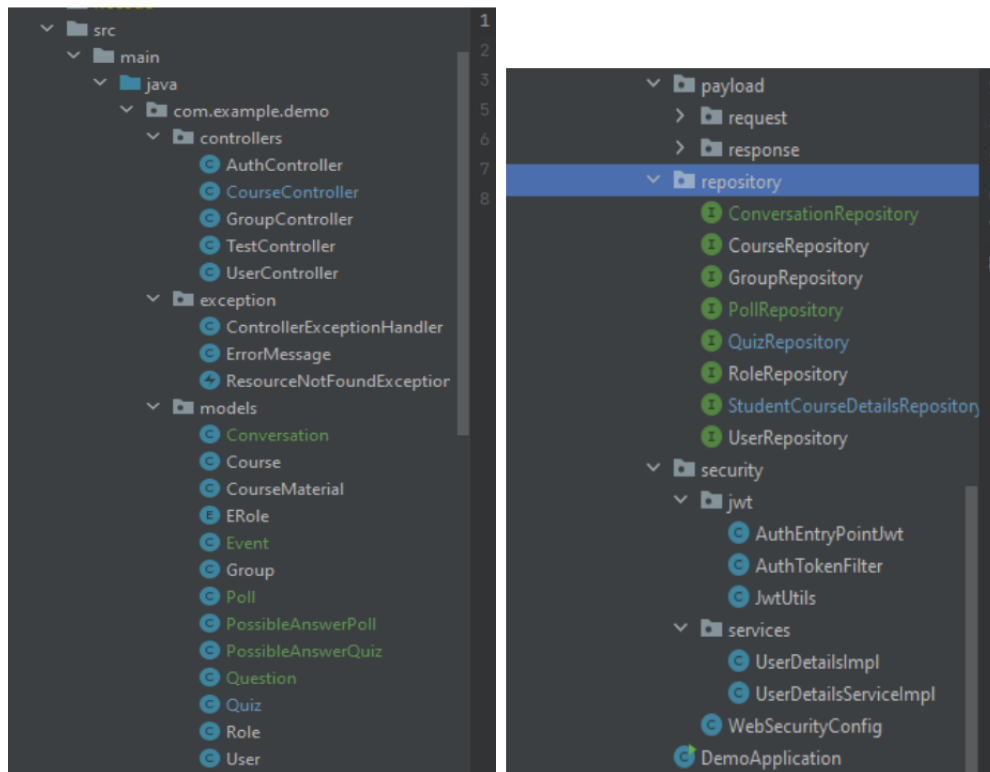


FIGURE 3.4 – Structure du code source

### 3.1.3 Choix Technologiques

Dans cette partie, nous présentons les différents choix technologiques utilisés dans ce projet.

- 1. Serveur d'application :** Apache Tomcat ou simplement Tomcat est un serveur d'applications, plus précisément un conteneur web libre de servlets et JSP. Issu du projet Jakarta, c'est l'un des nombreux projets de l'Apache Software Foundation. Il implémente les spécifications des servlets et des JSP du Java Community Process[5], est paramétrable par des fichiers XML et des propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP[n11].
- 2. Serveur de données :** MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateurs. Cette caractéristique concorde avec la principale fonctionnalité de notre application qui consiste à consulter les vidéos de la base pour visionnage ou téléchargement[n12].

- 3. Angular :** Angular (communément appelé "Angular 2+" ou "Angular v2 et plus") est un framework côté client, open source, basé sur TypeScript, et co-dirigé par l'équipe du projet « Angular » à Google et par une communauté de particuliers et de sociétés. Angular est une réécriture complète d'AngularJS, cadriciel construit par la même équipe. Il permet la création d'applications Web et plus particulièrement de ce qu'on appelle des « Single Page Applications » : des applications web accessibles via une page web unique qui permet de fluidifier l'expérience utilisateur et d'éviter les chargements de pages à chaque nouvelle action. Le Framework est basé sur une architecture du type MVC et permet donc de séparer les données, le visuel et les actions pour une meilleure gestion des responsabilités. Un type d'architecture qui a largement fait ses preuves et qui permet une forte maintenabilité et une amélioration du travail collaboratif[n13].
- 4. Spring Boot :** Spring Boot est un nouveau framework créé par l'équipe de chez Pivotal, conçu pour simplifier le démarrage et le développement de nouvelles applications Spring. Le framework propose une approche dogmatique de la configuration, qui permet d'éviter aux développeurs de redéfinir la même configuration à plusieurs endroits du code. Dans ce sens, Boot se veut d'être un acteur majeur dans le secteur croissant du développement d'applications rapide [n14].

## 3.2 Conception Détaillée

Une fois l'architecture est définie, il faut concevoir le modèle des données convenable à l'application. Ce modèle est dégagé à partir de l'observation du système à concevoir. Il s'agit donc de dresser les diagrammes de séquences et de classes selon la notation UML.

### 3.2.1 Le Diagramme de séquences « S'authentifier » :

Le diagramme de séquence ci-dessous représente le cas d'utilisation « S'authentifier ». Après l'accès au site, il y'a une demande d'authentification de l'utilisateur.

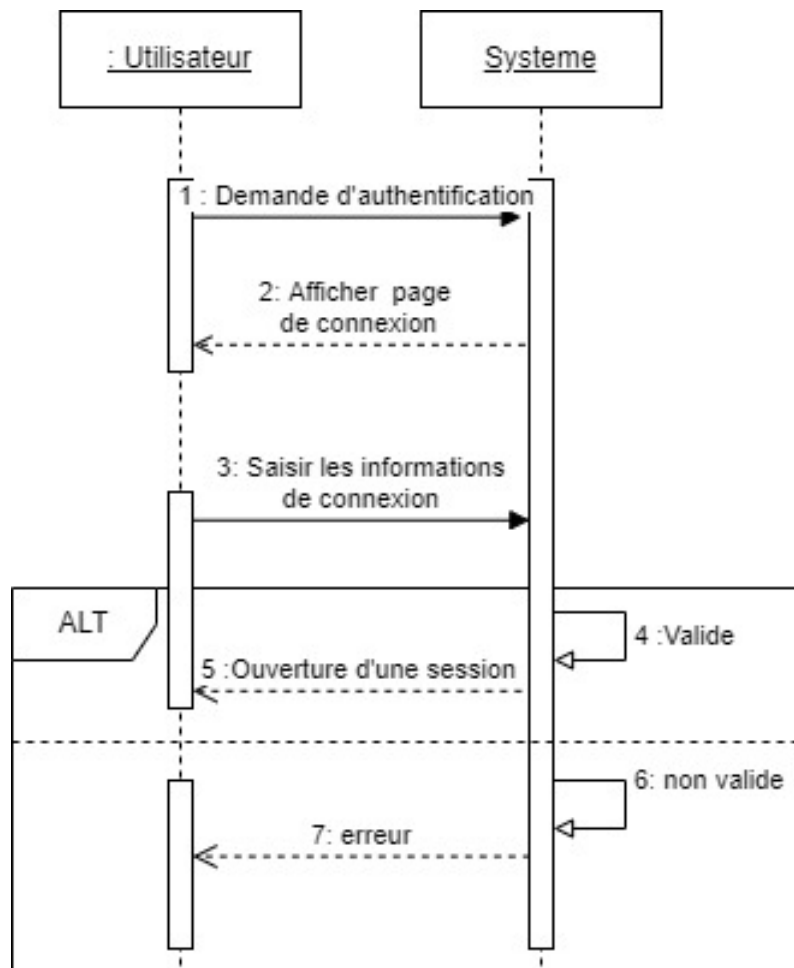


FIGURE 3.5 – Diagramme de séquences « S'authentifier »

Scenario : un Etudiant demande à s'identifier sur le site :

- L'Etudiant entre son adresse e-mail et son mot de passe sur la page de connexion.
- Après un certain temps, les informations sont considérées soit comme valides, auquel cas l'accès au site est accordé ou elles sont invalides, c'est-à-dire que l'on revient à l'écran de connexion.
- Si c'est un nouvel utilisateur il a l'option de s'inscrire.

### 3.2.2 Le Diagramme de séquences « Créer un profil » :

Le Diagramme de séquence ci-dessous représente le cas d'utilisation « Créer un profil ». Après l'accès au site, l'utilisateur a l'option de s'inscrire.

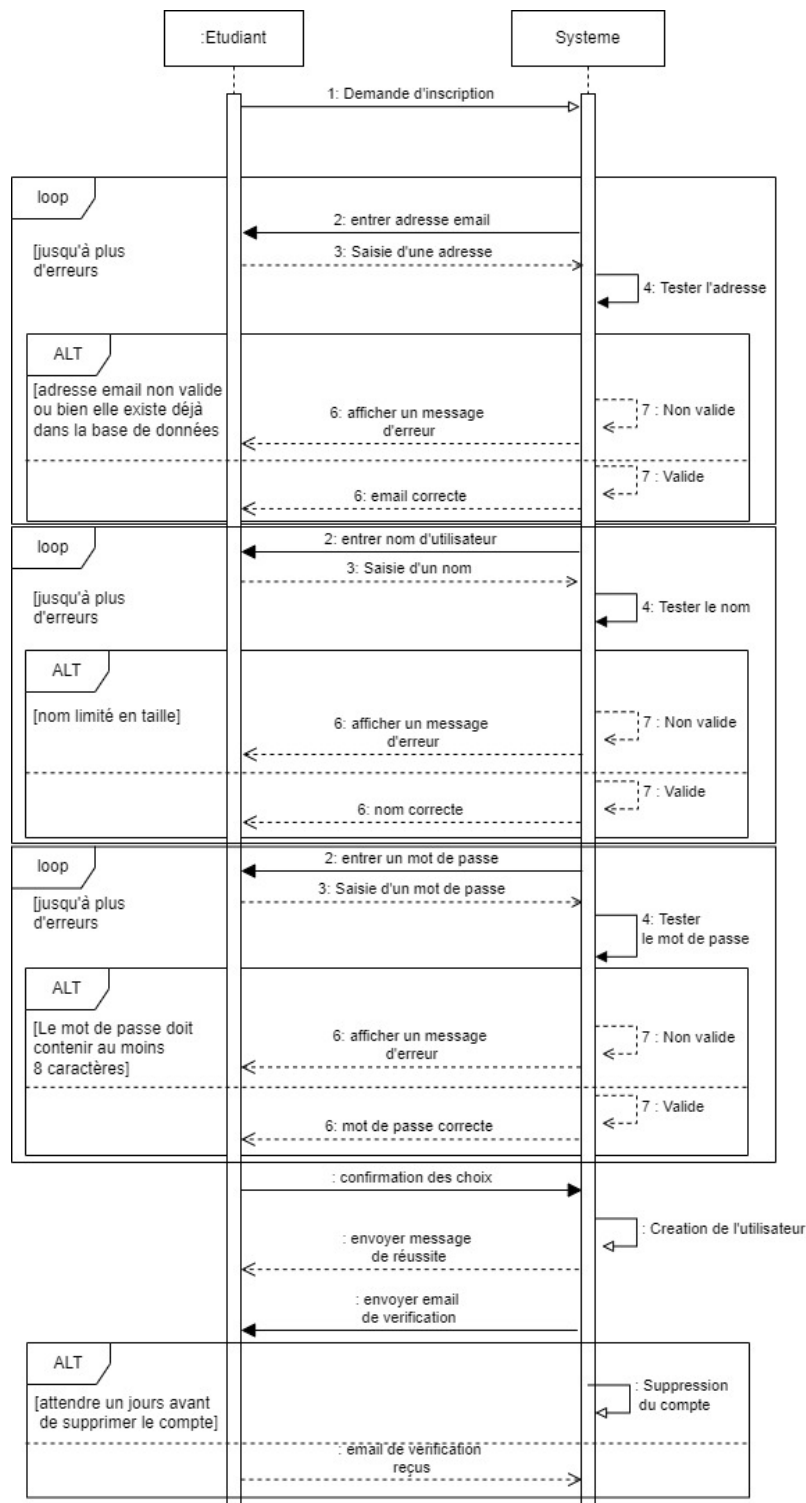


FIGURE 3.6 – Diagramme de séquences « Créer un profil »

Scenario : Un Etudiant sans compte au préalable veut s'inscrire au site :

- L'Etudiant accède à la page d'inscription depuis la page d'accueil (de connexion).
- L'Etudiant entre ses informations une à une et laisse le système traiter leurs justesse, Apres confirmation l'Etudiant et inscrit.
- Le système envoie un email de vérification afin d'affirmer l'appartenance de l'adresse mail à l'inscrit si ce n'est pas le cas, le compte sera supprimé après un jour d'attente.



### 3.2.3 Le diagramme de séquences « Consulter un cours » :

Le Diagramme de séquence ci-dessous représente le cas d'utilisation « Consulter un cours ». Scenario : Un Etudiant veut consulter un cours sur le site :

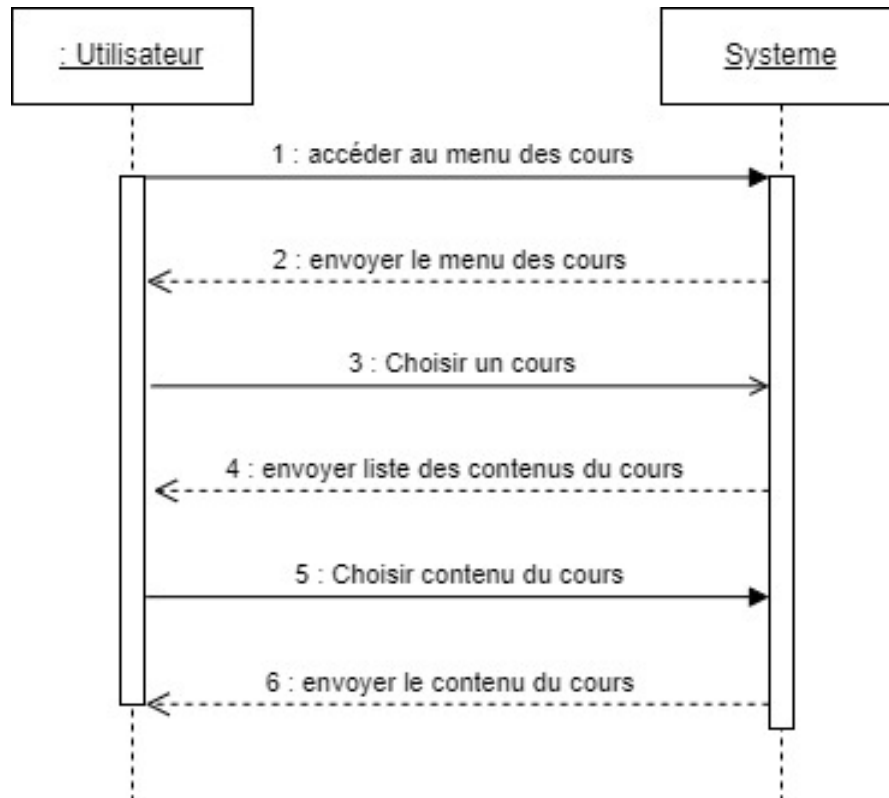


FIGURE 3.7 – Diagramme de séquences « Consulter un cours »

- L'Etudiant accède à la page des cours depuis le menu de gauche.
- L'Etudiant choisit le cours.
- Le système envoie la page cours choisit.

### 3.2.4 Le Diagramme de Classe de l'Application

Dans ce paragraphe, nous allons détailler les classes de notre application en précisant les différentes fonctionnalités.

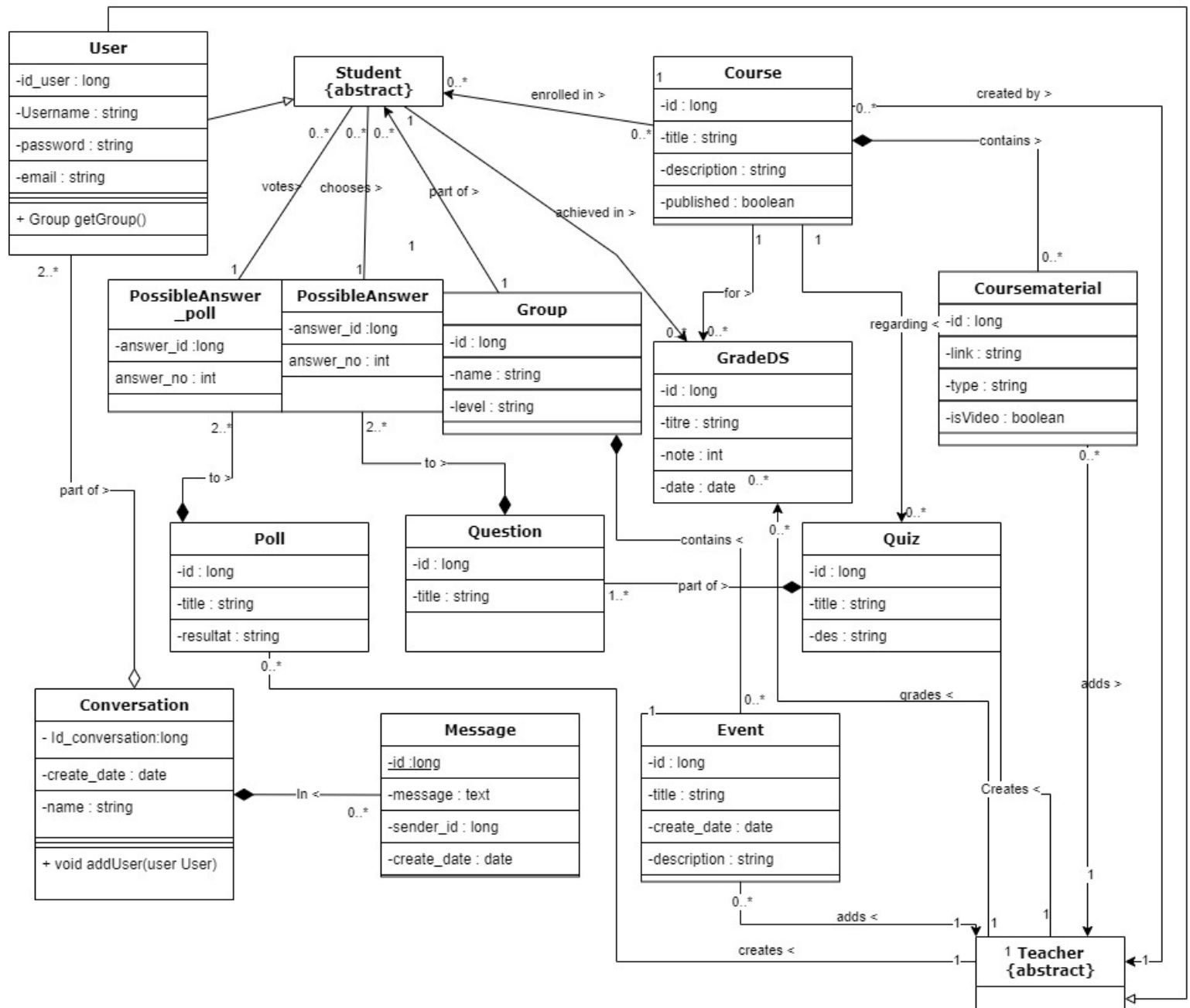


FIGURE 3.8 – Diagramme de Classes

## 3.3 Conception de la Base de Données

Afin de stocker les données traitées, l'application nécessite une base de données relationnelle. Pour la concevoir, il faut, tout d'abord, procéder par l'élaboration du schéma conceptuel à travers un modèle Entité-Association.

### 3.3.1 Diagramme entité-association

Le diagramme Entité-Association a pour but d'écrire de façon formelle les données qui seront utilisées par le système. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système à l'aide d'entités. A partir des besoins dégagés, nous avons pu distinguer les entités nécessaires pour le bon fonctionnement de cette application :

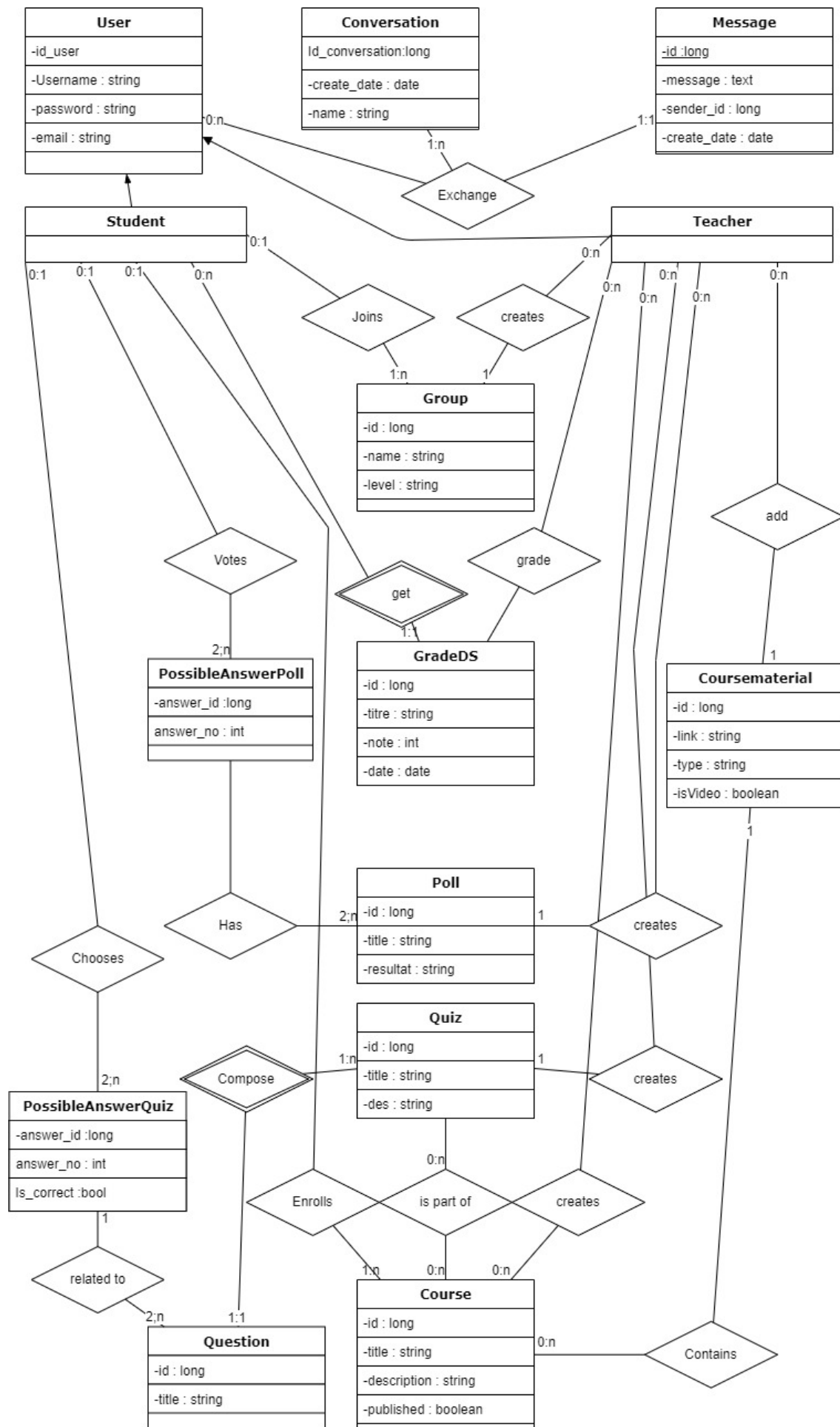


FIGURE 3.9 – Diagramme E/A de la base de données.

### 3.3.2 Description des associations

Le tableau ci-dessous présente une description plus détaillée des associations figurant dans le diagramme E/A de la base de données.

N°	Nom de l'association	Entités impliqués	type	description
1	Exchange	User, Message, Conversation	0,1,N	Une «Conversation» a au moins deux «User» et enregistre plusieurs «Message».
2	Joins	Student, Group	1,N	Un «Student» ne rejoint qu'un seul groupe et un groupe contient plusieurs «Student».
3	Enroll	Student, Courses	N,M	Un «Student» peut rejoindre plusieurs «Course» et un «Course» est suivi par plusieurs «Student».
4	Achieved	Student, GradeDS	N,1	Un «student» possède plusieurs notes d'évaluation et une note est unique pour chaque «student».
5	Contains	Courses, Coursematerial	0,1	Un «Course» possède des «Coursematerial» et un «Coursematerial» est unique à un «Course».
6	Chooses	Student, PossibleAnswerQuiz	0,N	Un «Student» ne choisit qu'une seule «PossibleAnswerQuiz» parmi celles proposées.
7	Has	PossibleAnswerPoll, Poll	0,N	Une «Question» composé d'une question et possède au moins deux «PossibleAnswerPoll» avec une seule qui est compté comme correcte.
8	compose	Question, Quiz	1,N	Un «Quiz» est un ensemble de «Question».

9	Answer to	PossibleAnswerQuiz, Question	0,N	Une «Question» composé d'une question et possède au moins deux «PossibleAnswerQuiz» avec une seule qui est compté comme correcte.
---	-----------	---------------------------------	-----	---

### 3.3.3 Description des tables de la Base de données

Le tableau ci-dessous présente une description plus détaillée des entités figurant dans le diagramme E/A de la base de données.

N°	Entités	description
1	User	Cette table contient les informations nécessaires pour identifier tous les enregistrés sur ce site web.
2	Course	Cette table identifie les Cours possibles à donner selon leurs niveaux.
3	CourseMaterial	Une table qui sert à identifier le contenu (liens vidéo, documents etc) pour un certain cours.
4	Group	Cette table identifie les Groupe d'étudiants selon leurs classes.
5	GradeDS	Cette table identifie un message envoyé à un utilisateur par un autre.
6	Conversation	Réunis les messages et leurs destinataires.
7	Quiz	Contient le titre et la description d'un quiz crée.
8	Poll	Contient le titre d'un sondage et son résultat.
9	GradeDS	Sert à identifier la note d'un DS et à qui elle est affectée.

## Conclusion

La phase de conception a permis de concevoir une architecture 4-tiers bénéficiant de la puissance du motif de conception MVC. En outre, ce chapitre a abouti à un modèle logique de données déduit à partir d'un modèle entité-association. Le chapitre suivant fera l'objet de la phase de réalisation, cette dernière permettra d'implémenter l'architecture précédemment détaillée.

# CHAPITRE 4

---

## Réalisation

---

Dans cette partie, il est question de décrire l'aspect implémentation. Nous commençons par présenter l'environnement matériel et logiciel supportant l'application.

Ensuite, nous allons passer en revue les différentes tâches réalisées pour terminer par un chronogramme qui décrit toutes les étapes de mise en œuvre de notre site Web.

### 4.1 Environnement de travail

Nous présentons dans cette section l'environnement matériel ainsi que celui logiciel utilisés pour le développement de notre application.

#### 4.1.1 Configuration matérielle

Pour la réalisation de cette application, nous avons à notre disposition les trois unités suivantes :

- Unité 1 :
  - Ordinateur : ASUS X540LJ
  - Processeur : Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz
  - Mémoire physique (RAM) installée : 8,00 Go
- Unité 2 :
  - Ordinateur : DELL
  - Processeur : Intel(R) Core(TM) i7-3537U CPU @ 2.00GHz
  - Mémoire physique (RAM) installée : 12,00 Go
- Unité 3 :
  - Ordinateur : HP OMEN
  - Processeur : Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz
  - Mémoire physique (RAM) installée : 12,00 Go

#### 4.1.2 Configuration logicielle

Pour réaliser notre projet, nous avons eu recours aux logiciels énumérés ci-dessous :

**Langages de programmation :** Java, SQL, JavaScript/TypeScript, HTML/CSS.

**Librairies particulières ou frameworks :** Spring Boot, Angular, MySql.

**nvironnements de développement intégré** (EDI ou Outils RAD pour interfaces utilisateur etc) MicroSoft Visual Studio Code, IntelliJ idea.

## 4.2 Captures d’Ecran

Dans cette partie, on présentera quelques pages de notre site. Ces pages peuvent donner une idée plus claire sur les résultats de notre travail. Ils sont en fait, le fruit des phases d’analyse, de conception et de codage.

### 4.2.1 Pages Communes aux utilisateurs du site

Voici la première page rencontrée par l’utilisateur. Ici l’utilisateur doit saisir son “username” et son mot de passe ou bien s’il n’en a pas il a l’option de créer un nouveau compte. Avoir un compte est nécessaire pour l’accès et l’utilisation du site web.

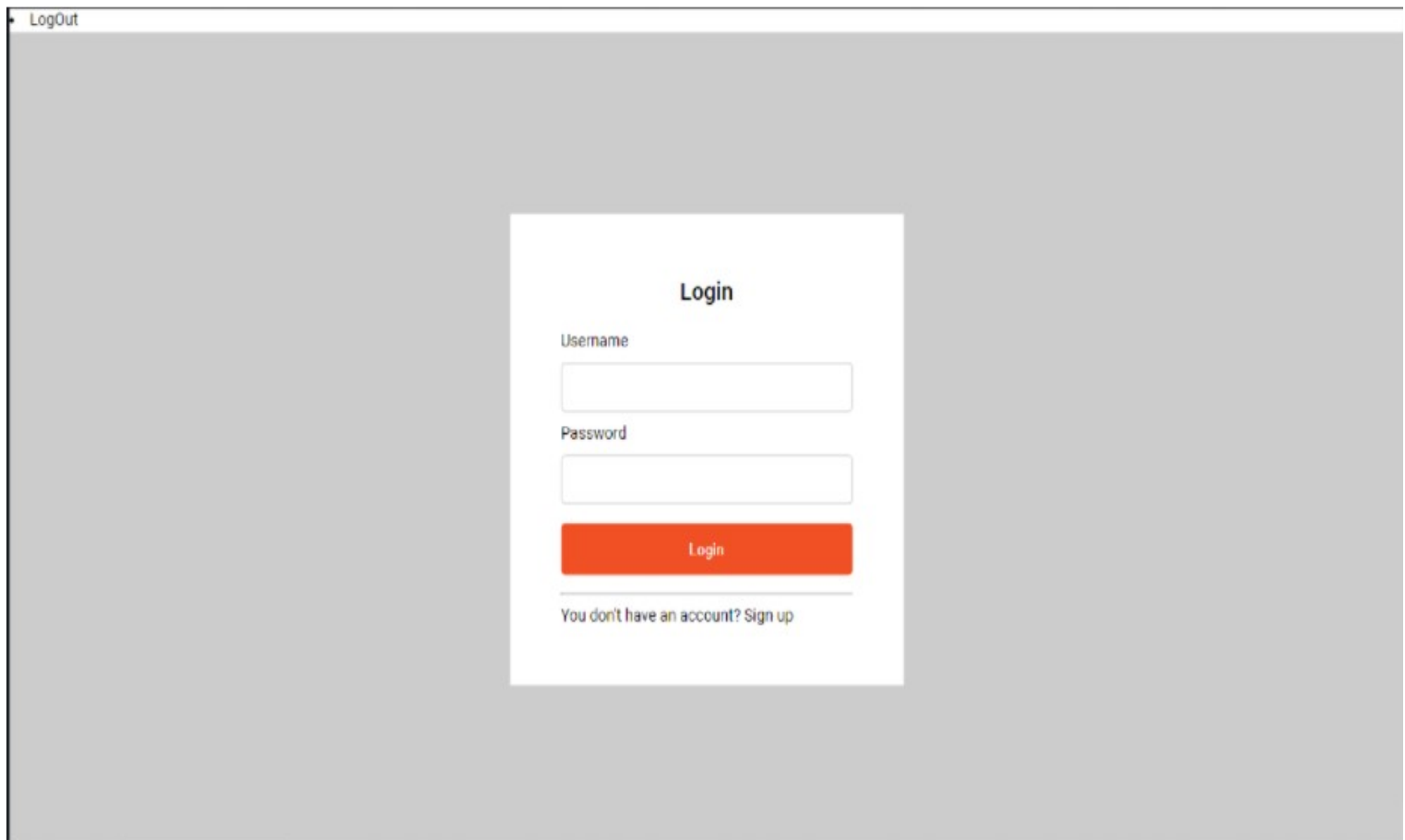
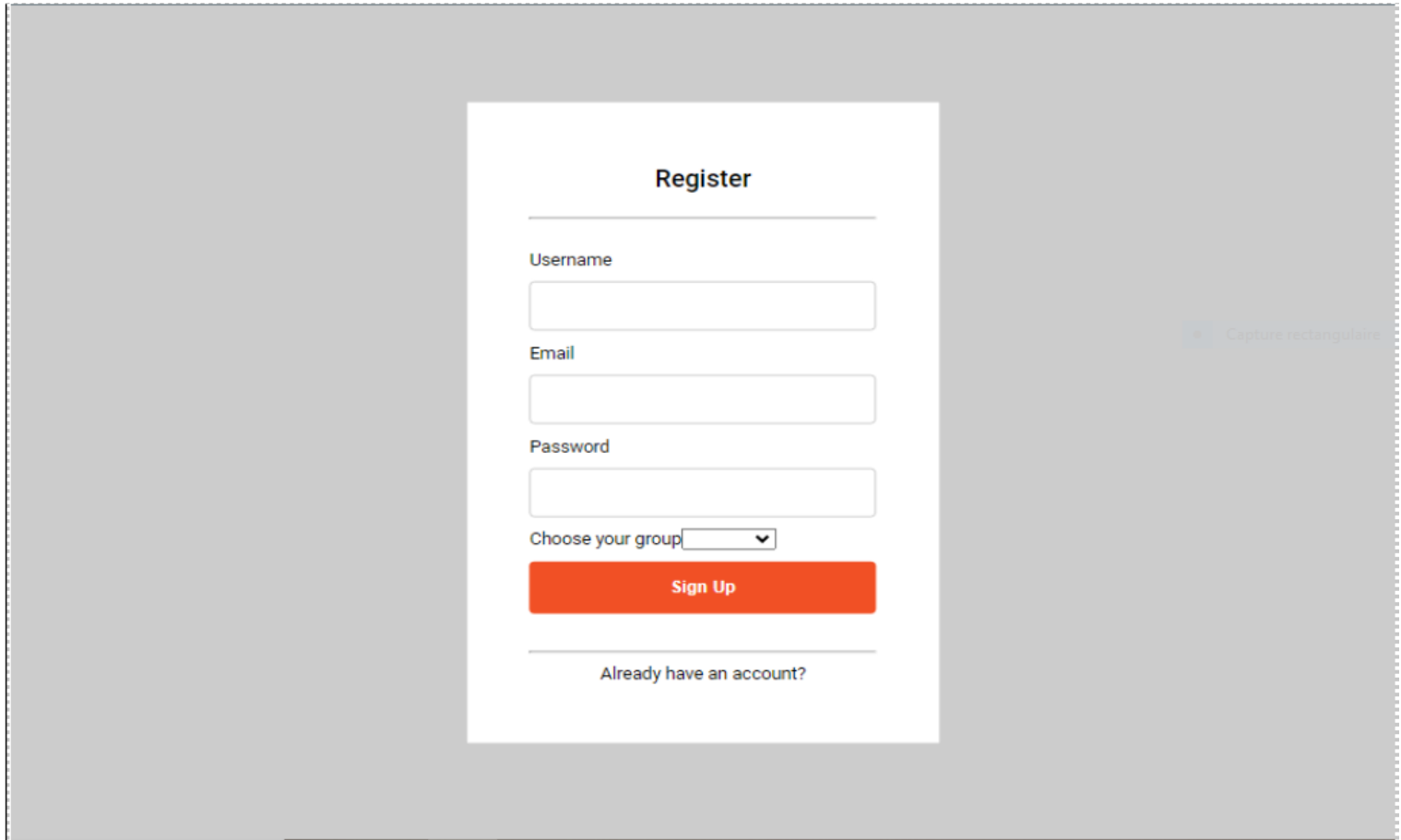


FIGURE 4.1 – Page d’authentification



Comme mentionné dans le paragraphe précédent, l'utilisateur a l'option de créer un nouveau compte. Cette tâche sera faite dans cette page après avoir choisi l'option "Sign up" dans la page précédente.

A screenshot of a web registration form titled "Register". The form is centered on a light gray background. It contains the following elements: a title "Register" in bold black text; a horizontal line; a label "Username" above a text input field; a label "Email" above a text input field; a label "Password" above a text input field; a label "Choose your group" followed by a dropdown menu; a red "Sign Up" button; and a link "Already have an account?" at the bottom. A faint watermark "Capture rectangulaire" is visible on the right side of the image.

**Register**

Username

Email

Password

Choose your group

**Sign Up**

[Already have an account?](#)

FIGURE 4.2 – Page d'inscription

### 4.2.2 Les pages accessibles par l'administrateur

Le tableau de bord de l'admin est désigné pour afficher quelques informations à propos des étudiants inscrits.

**ManageMe**

**Admin dashboard**

**Top Students**

Rank	Student	Level	Grade
1	Ali	Master 1	19
2	Salma	Ingénieur 2	18.5
3	Mariam	Ingénieur 3	18

**Groups**

Name	Group's name
Name	Group 1
Name	Group 1
Name	Group 1

[View Group](#)

**Calendar**

MAY 2022

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

**Upcoming**

Today

FIGURE 4.3 – Tableau de bord de l'admin

La page des notes permet à l'enseignant de consulter les groupes d'étudiants selon leurs modules et d'afficher et modifier leurs notes d'examens.

The screenshot shows the ManageMe application interface. On the left is a sidebar with navigation links: Dashboard, Courses, Schedule, Groups, Grades (highlighted), Messages, and Settings. At the bottom left is a 'Log out' button. The main content area displays a hierarchical view of the data:

- Module 1**: Instructeur 1, with a 'Show Groups' button.
- Group 1**: Niveau, Nombre de participants, with a 'Show DS' button.
- DS 1**: Date, Nombre de participants, with a 'Show Grades' button.

Below this hierarchy is a table of student grades and attendance:

Id	Name	Grade	Présent
2	Ahmed	14	yes
1	Hammam	16	yes
3	iskander	15	yes
2	Ahmed	14	yes
1	Hammam	0	no
3	iskander	15	yes
2	Ahmed	14	yes
1	Hammam	16	yes
3	iskander	15	yes
2	Ahmed	0	no

At the bottom of the table, there is a pagination control: 'Items per page: 10' and '1 - 10 of 12'.

On the right side of the interface, there is a 'Calendar' section showing 'MAY 2022' with a calendar grid. The date '11' is highlighted. Below the calendar, it says 'Selected date: Wed May 11 2022 12:57:21 GMT+0100 (West Africa Standard Time)'. There is also an 'Upcoming' section with a 'Today' button and a dropdown arrow.

FIGURE 4.4 – Page des notes

La page d'ajout des cours permet à l'enseignant de créer un nouveau cours, de lui attribuer un titre, une description ainsi que d'ajouter son contenu (documents, video ..).

FIGURE 4.5 – Page d'ajout des cours

Cette page présentée par la figure ci-dessous permet à l'enseignant d'ajouter un événement au calendrier, de lui affecter un nom, une description et une date.

FIGURE 4.6 – Page d'ajout d'événements

### 4.2.3 Les pages accessibles par l'étudiant

Il s'agit du tableau de bord de l'étudiant. Dans cette page l'étudiant trouvera un résumé des informations d'intérêt (les événements à venir, la moyenne de ses notes, les cours auxquels il est inscrit ...).

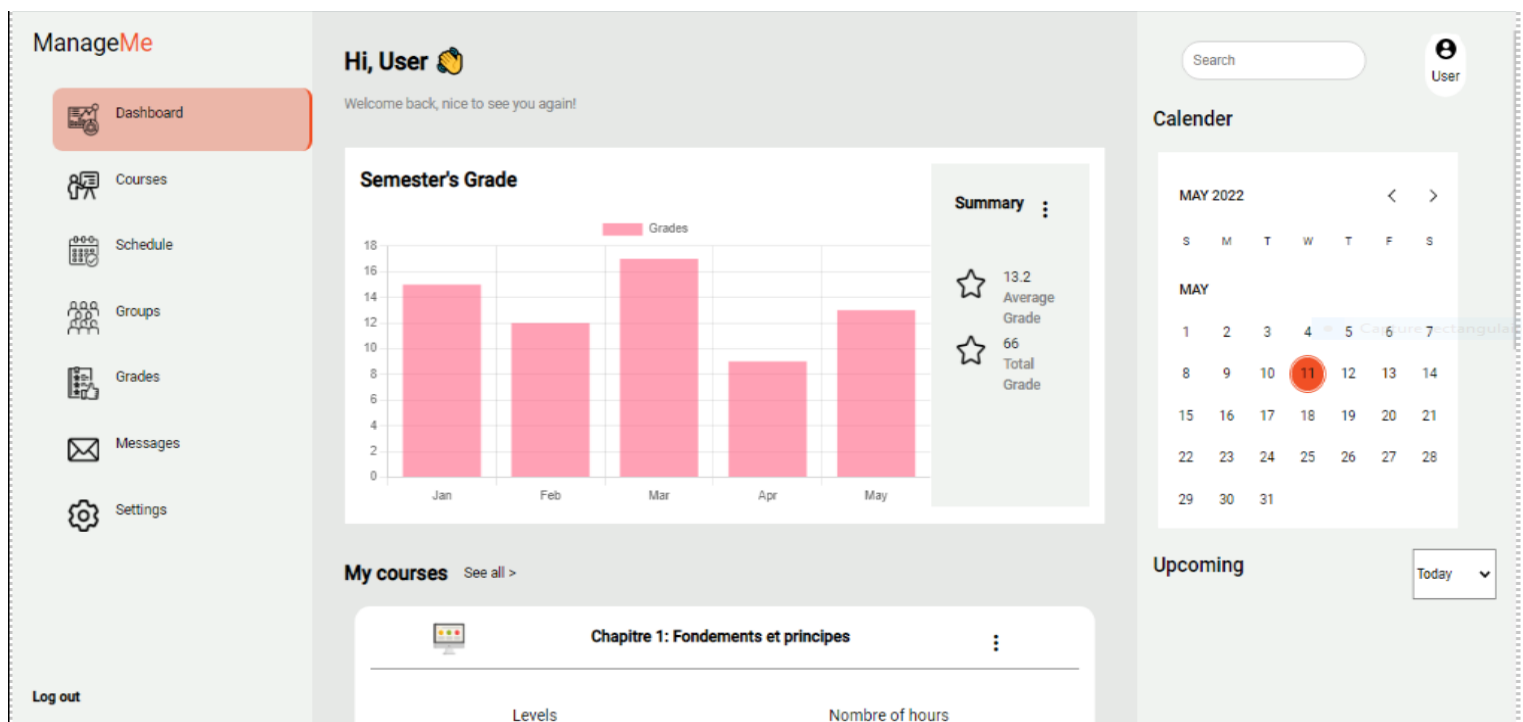


FIGURE 4.7 – Tableau de bord de l'étudiant

La page des cours permet à l'étudiant de consulter les cours auxquels il est inscrit ainsi que leurs contenus.

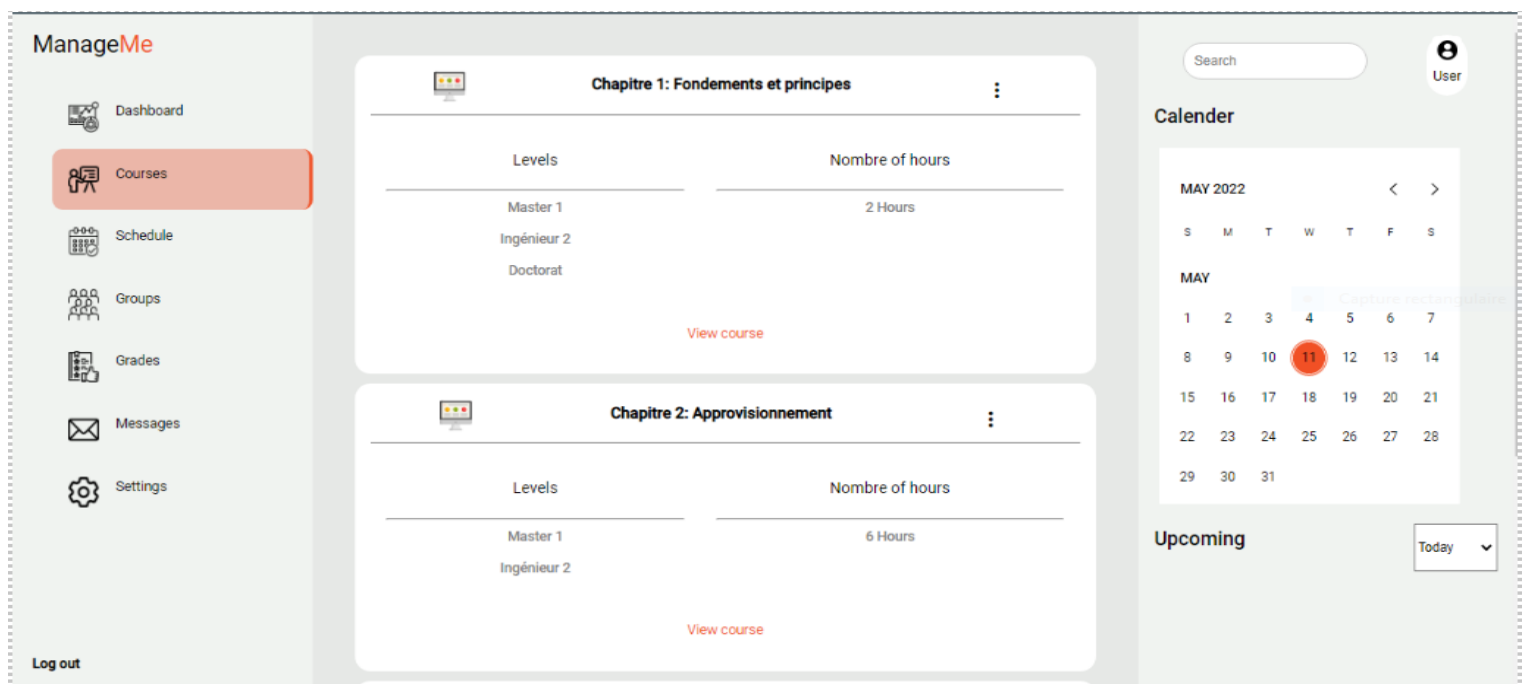


FIGURE 4.8 – Page de cours

Après avoir choisi “view course”, l’étudiant aura une description entière du cours concerné ainsi que son contenu, comme le montre la figure suivante.

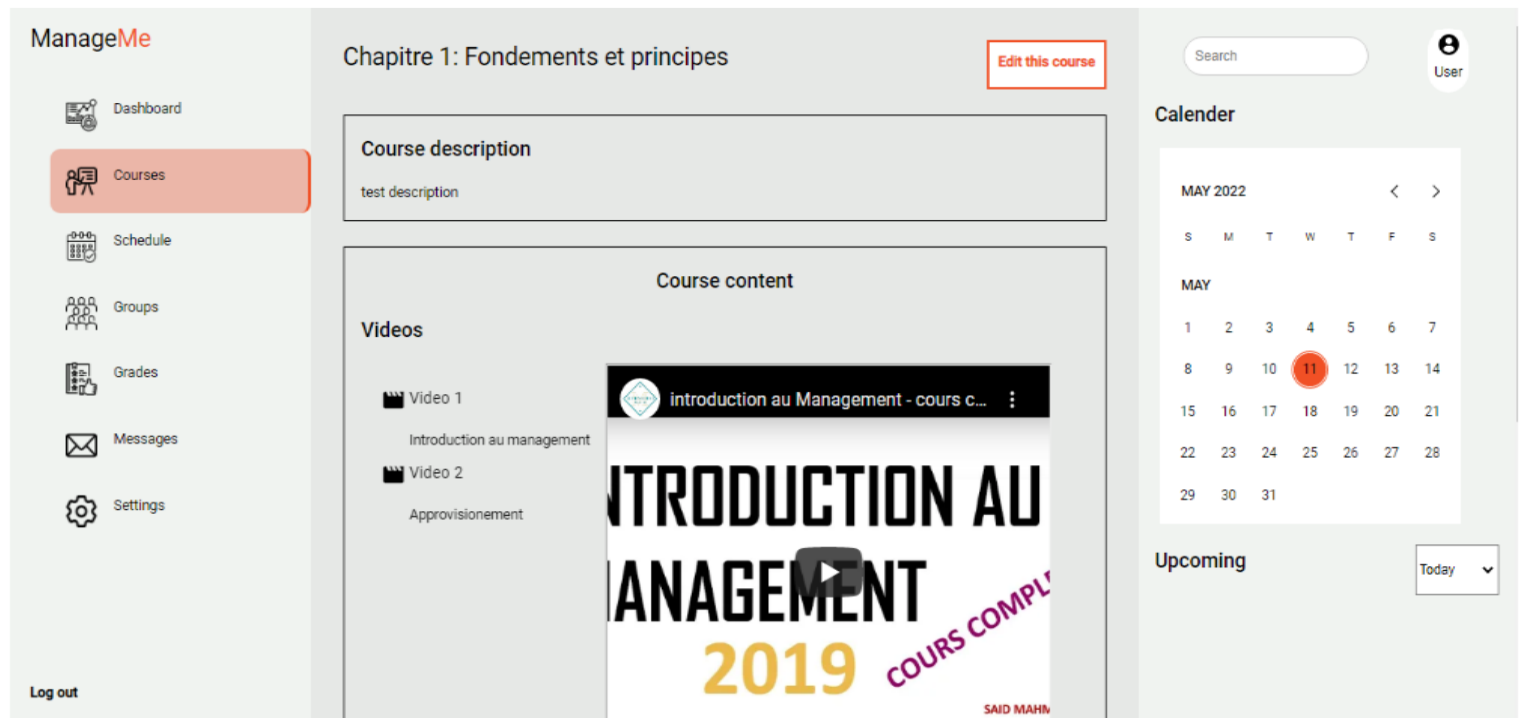


FIGURE 4.9 – Page du contenu de cours

La page du calendrier offre à l’étudiant une vue détaillée de son calendrier ainsi que les événements à venir.

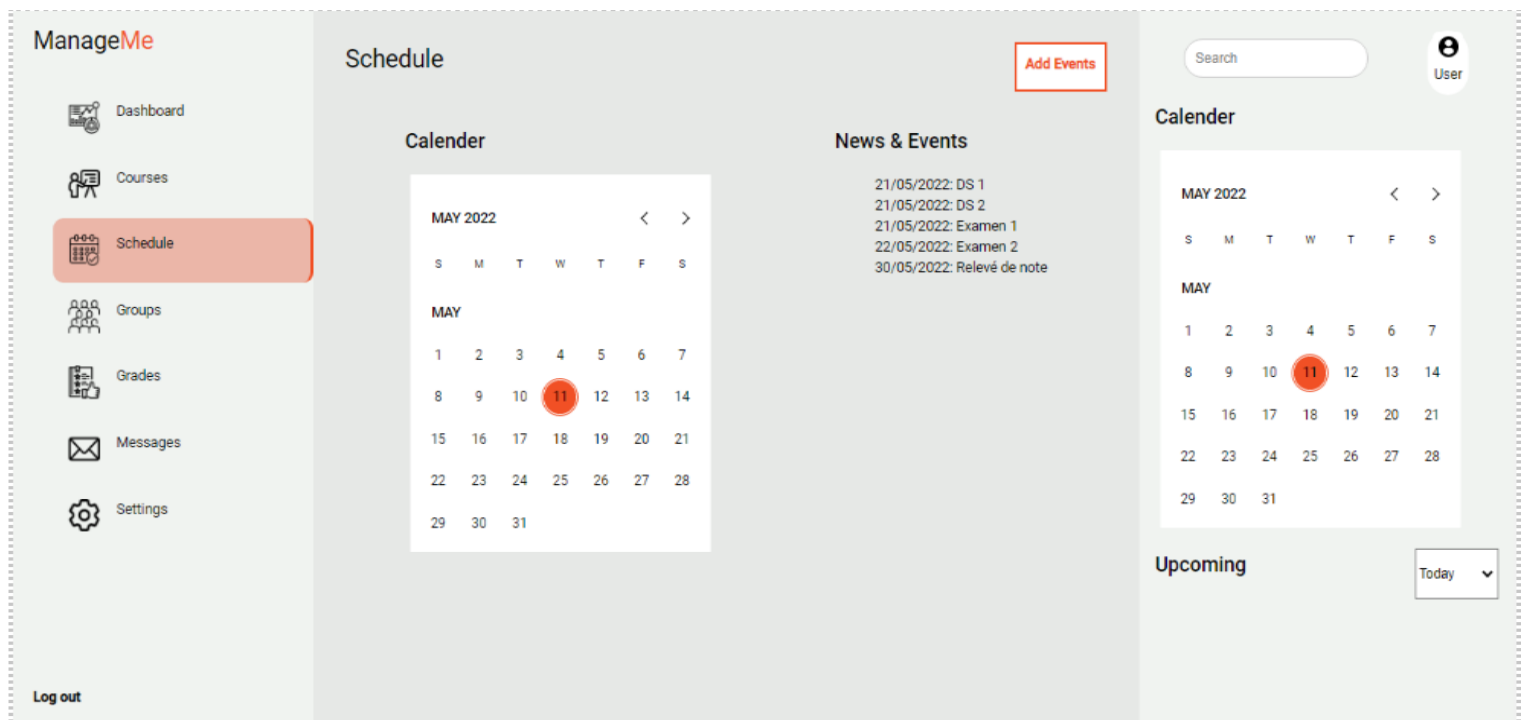


FIGURE 4.10 – Page de calendrier

Cette page permet à l'étudiant de consulter son groupe (liste de camarades ...).

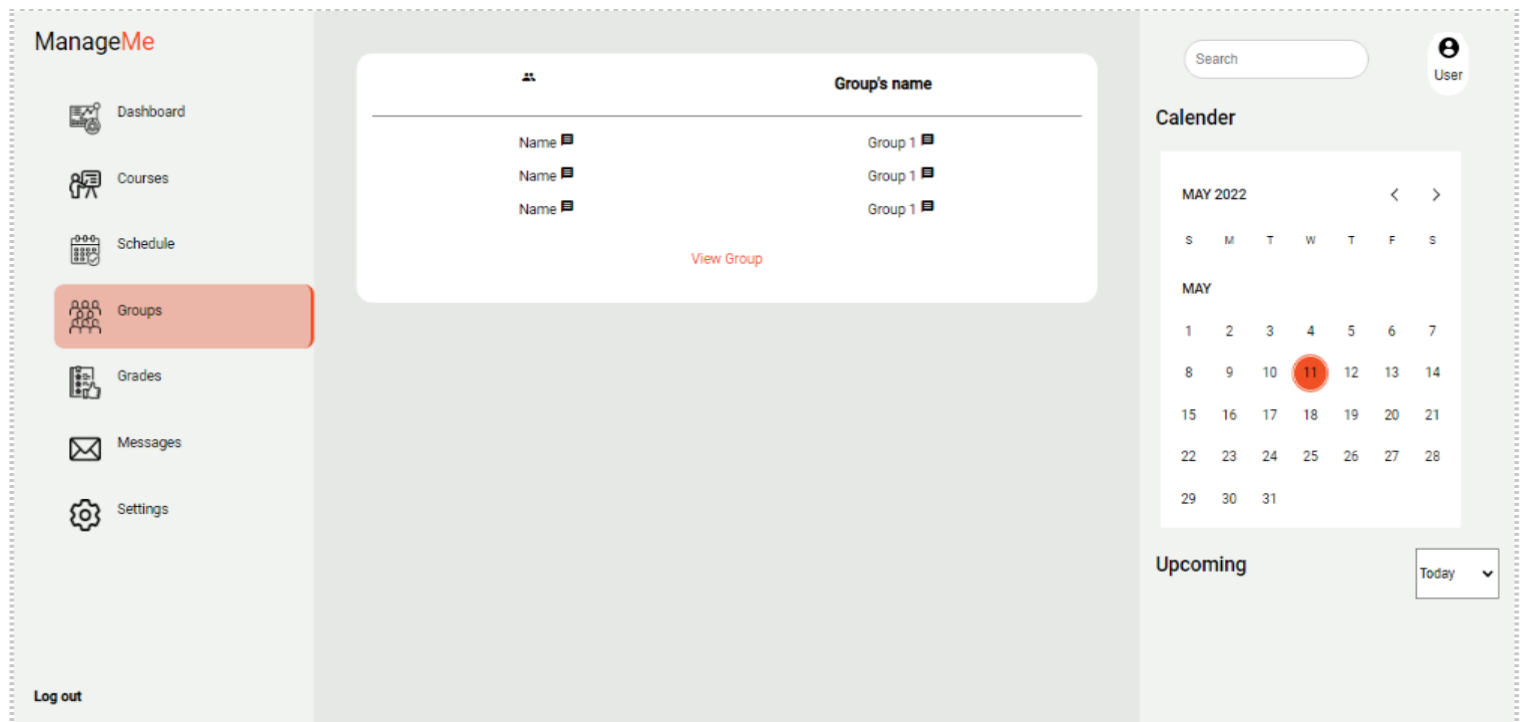


FIGURE 4.11 – Page de groupes

Dans cette page l'étudiant peut consulter les messages envoyés et reçus ainsi que communiquer avec d'autres étudiants et/ou l'enseignant.

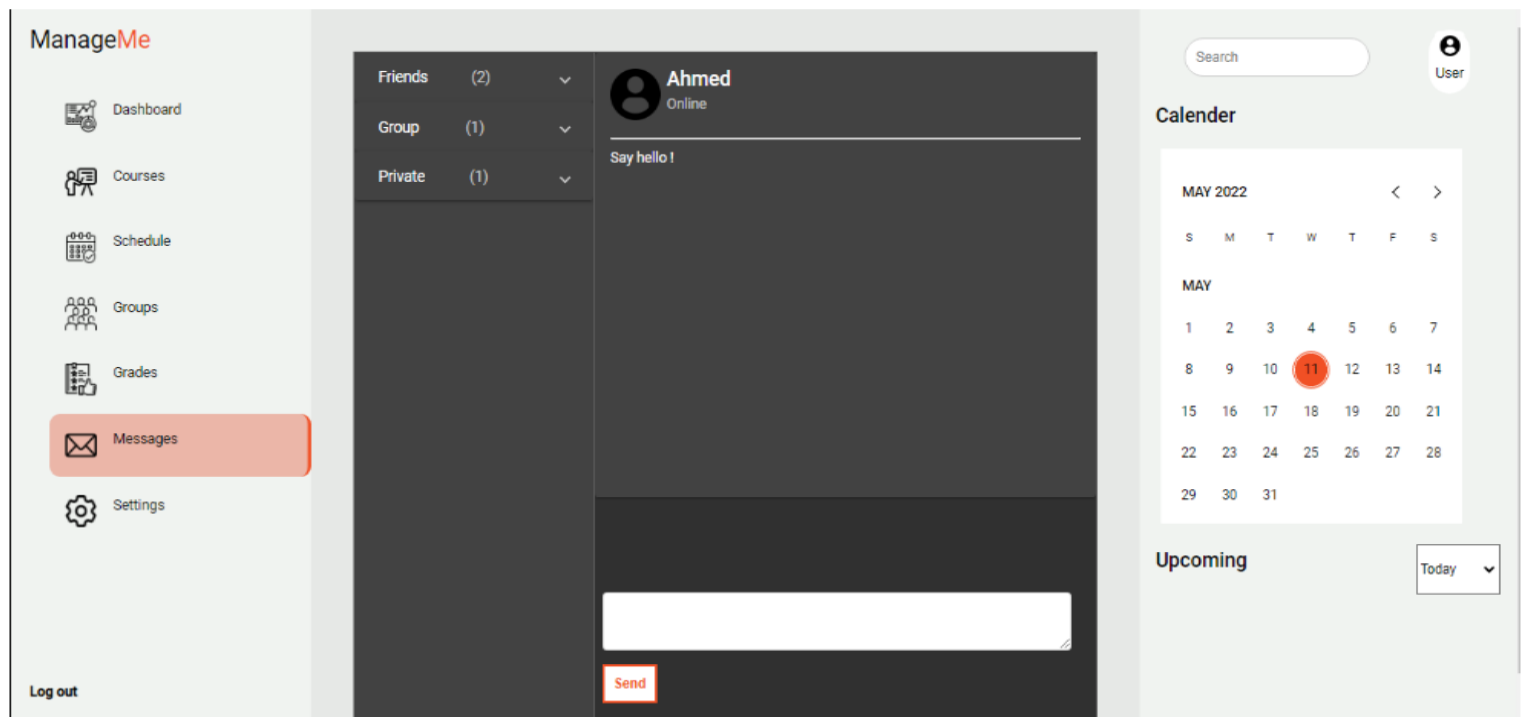


FIGURE 4.12 – Page de messagerie





## Conclusion :

Dans ce chapitre nous avons présenté l'environnement de travail ainsi que quelques interfaces du site afin de visualiser les résultats obtenus. Nous avons ensuite présenté les différentes étapes par un chronogramme des tâches dans le but de clarifier la répartition temporelle de ces derniers, tout au long de ce projet.

---

# Conclusion et perspectives

---

De nos jours, la communication web est la plus favorisée en termes de temps et en termes de fiabilité. D'où l'importance du E-learning ; il permet d'améliorer les processus d'apprentissage à travers la formation à distance et d'avoir accès à différents contenus pédagogiques sans contraintes de temps et de lieu.

Dans ce projet, nous avons essayé de concevoir et développer une application web pour enseignement à distance. Pour cela, nous avons élaboré une étude préalable sur les concepts du e-learning et de la formation à distance. Puis nous avons consulté les différentes plateformes de e-learning déjà disponibles en mettant l'accent sur leurs limites.

Pour faire la conception détaillée de notre application, on a adopté l'UML qui nous a aidé à élaborer la représentation visuelle des objets, des états et des processus dans notre logiciel. Pour implémenter notre solution, nous avons utilisé les motifs architecturaux 4-tiers et 4 couches pour notre plateforme. Pour la partie "Frontend", nous avons utilisé le Framework Angular qui est basé sur le langage TypeScript et considéré comme un Framework lourd et utilisé pour les PC mobiles et de bureau. Pour la partie "Backend", on a utilisé Spring Boot. Afin de connecter les parties frontend et backend, on a exploité Apache Tomcat comme serveur d'application.

Nous avons également conçu une base de données en utilisant un système de gestion de base de données relationnelle MySQL pour bien structurer les informations de notre système.

Cependant notre plateforme peut certainement être améliorée sur différents plans. À titre d'exemple on pourrait enrichir cette application pour prendre en charge des "voice-chats" ou "video-chats", ou alors dans le but de perfectionner notre système on pourrait envisager l'utilisation des Services Web afin d'alléger la couche métier de l'application.

---

# Netographie

---

[n1] Enseignement à distance : <https://www.bienenseigner.com/lenseignement-a-distance-avantages-et-inconvenients/#:text=D%C3%A9finition%20de,l%E2%80%99enseignement%20%C3%A0%20distance>, dernière consultation le 23/03/2022

[n2] Pierre Mœglin, Les industries éducatives, Paris, Puf, coll. « Que sais-je ? », 2010 , dernière consultation le 25/03/2022.

[n3] SAP Litmos : <https://www.litmos.com/> , dernière consultation le 15/04/2022

[n4] SUP'COM e-learning : <https://e-learning.supcom.tn/moodle/> , dernière consultation le 16/04/2022

[n5] Udemy : <https://www.udemy.com/> , dernière consultation le 19/04/2022

[n6] Studytn : <https://www.studytn.com/> , dernière consultation le 19/04/2022

[n7] AcurSchool : <https://acurschool.com/> , dernière consultation le 19/04/2022

[n8] Architecture MVC : <https://developer.mozilla.org/en-US/docs/Glossary/MVC>, dernière consultation le 18/04/2022

[n9] Architecture 4 couches : <https://www.javatpoint.com/spring-boot-architecture>, dernière consultation le 18/04/2022

[n10] Architecture 4-tiers : <https://www.schoolmouv.fr/cours/systeme-de-gest>

ion-de-bases-de-donnees-sgbd-/fiche-de-cours, dernière consultation le 18/04/2022

[n11] Apache Tomcat : Site officiel <https://tomcat.apache.org/>, dernière consultation le 12/05/2022

[n12] MySQL : NetBeans. Creating a Simple Web Application Using a MySQL Database. [Http://www.netbeans.org](http://www.netbeans.org), dernière consultation le 12/05/2022

[n13] Angular : Manjunath M, « AngularJS and Angular 2+ : a Detailed Comparison — SitePoint » [archive], sur <https://www.Sitepoint.com>, 6 avril 2018, dernière consultation le 12/05/2022

[n14] Spring boot : <https://www.infoq.com/fr/articles/microframeworks1-spring-boot/>, dernière consultation le 10/05/2022