

Université de Monastir

Institut Supérieur d'Informatique et Mathématique Monastir



Mémoire de Projet de Fin d'Année

Conception et Réalisation d'une plateforme E- Learning

Réalise par :

Bahri Chaher

Ben Hlel Adib

Arouay Oussama Dhiaa

Ben Lezreg Bilel

Bayoudh Anass

Jaziri Ahmed

Soutenu le 5/03/2024 devant le jury

Président : Mme Aljia BOUZIDI, ISIMM

Encadrant Pédagogique : Mme Aljia BOUZIDI, ISIMM

Introduction	1
1. Contexte générale du projet	1
1.1. Cadre du projet	1
1.2. Présentation de l'organisation d'accueil :	1
1.3. Objectif du projet	2
2. Etude de l'existant	2
2.1. Solutions existantes	3
2.1.1. Coursera	3
2.1.2. Kaggle	4
2.1.3. Udemy	5
2.1.4. Tableau comparative.....	6
2.2. Critique de l'existant.....	6
2.3. Solution retenue.....	6
3. Méthodologie de travail.....	7
3.1. Méthode 2TUP : Two Tracks Unified	7
3.2. Méthode agile Scrum.....	8
3.3. Méthodologie adoptée pour le projet de site web e-learning.....	9
Conclusion	9
Introduction	10
1. Identification des acteurs.....	10
2. Modèle informationnelle de contexte.....	10
2.1. Diagramme de contexte.....	11
3. Capture des besoins	11
3.1. Besoins fonctionnels.....	12
3.2. Besoins non fonctionnels	13
4. Spécification des besoins	13
4.1. Diagramme de cas d'utilisation général.....	13
4.2. Diagramme de cas d'utilisation détaillés	14
4.2.1. Diagramme de cas d'utilisation de l'acteur Utilisateur	14

4.2.2.	Diagramme de cas d'utilisation de l'acteur Admin	15
4.2.3.	Diagramme de cas d'utilisation de l'acteur Tuteur	16
4.2.4.	Diagramme de cas d'utilisation de l'acteur Etudiant	17
5.	Description des cas d'utilisation.....	17
5.1.	Cas d'utilisation « S'authentifier »	18
5.1.1.	Description textuelle.....	18
5.1.2.	Diagramme de séquence d'analyse	19
5.2.	Cas d'utilisation « Gérer ressources »	20
5.2.1.	Description textuelle.....	20
5.2.2.	Diagramme de séquence d'analyse	20
5.3.	Cas d'utilisation « Gérer certificats ».....	22
5.3.1.	Description textuelle.....	22
5.3.2.	Diagramme de séquence du cas d'utilisation "Gérer certificats"	23
5.4.	Cas d'utilisation « Se communiquer »	23
5.4.1.	Description textuelle.....	23
5.4.2.	Diagramme de séquence du cas d'utilisation "Se communiquer"	25
Conclusion	25
Introduction	26
1.	Architecture Applicative	26
2.	Conception de Base de Données.....	26
2.1.	Modèle Conceptuel de Données (MCD)	26
2.2.	Modèle Logique de Données (MLD)	28
3.	Conception logicielle	29
3.1.	Vue statique : Diagrammes de classes	29
3.1.1.	Diagramme de classe général.....	29
3.1.2.	Diagramme de classes de conception (MVC)	30
3.2.	Vue dynamique : Diagramme de séquence	32
4.	Conception graphique	32
4.1.	Interface du cas d'utilisation « Inscription »et « Connexion »	32
4.2.	Interface du cas d'utilisation « Gérer Progrès du cours »	34
Conclusion	35

Figure 1	Logo du "ISIMM"	1
Figure 2	Page d'accueil du site « Coursera »	3
Figure 3	Page d'accueil du site « Kaggle »	4
Figure 4	Page d'accueil du site « Udemy »	5
Figure 5	: Représentation de la méthodologie 2TUP	7
Figure 6	Représentation de la méthodologie SCRUM	8
Figure 7	Diagramme de contexte	11
Figure 8	Diagramme de cas d'utilisation général	14
Figure 9	Diagramme de cas d'utilisation de l'acteur Utilisateur	15
Figure 10	Diagramme de cas d'utilisation de l'acteur Admin	15
Figure 11	Diagramme de cas d'utilisation de l'acteur Tuteur	16
Figure 12	Diagramme de cas d'utilisation de l'acteur Etudiant	17
Figure 13	Diagramme de séquence du cas d'utilisation "s'authentifier"	19
Figure 14	Diagramme de séquence du cas d'utilisation "Gérer ressources"	21
Figure 15	Diagramme de séquence du cas d'utilisation "Gérer certificats"	23
Figure 16	Diagramme de séquence du cas d'utilisation "se communiquer"	25
Figure 17	Modèle Conceptuel de données (MCD)	26
Figure 18	Le Modèle Logique de Données (MLD)	29
Figure 19	Diagramme de classes générale	30
Figure 20	Diagramme de classes de conception (MVC)	31
Figure 21	Diagramme de séquence générale	32
Figure 22	Interface "Inscription"	33
Figure 23	Interface "Connexion"	33
Figure 24	Interface "Cas de Code d'accès oubliée"	34
Figure 25	Interface de l'étudiant pour gérer les cours	34

Tableau 1	Tableau comparatif entre les solutions existantes -----	6
Tableau 2	Description textuelle du cas d'utilisation "S'authentifier" -----	18
Tableau 3	Description textuelle du cas d'utilisation "Gérer ressources" -----	20
Tableau 4	Description textuelle du cas d'utilisation "Gérer certificats" -----	22
Tableau 5	Description textuelle du cas d'utilisation "Se communiquer" -----	23
Tableau 6	Les entités du modèle conceptuel de données -----	27

Introduction

Au cours de ce chapitre, nous allons introduire notre projet en étudiant son cadre général et la problématique. Ensuite nous allons procéder à l'étude de l'existant avec leurs critiques et nous introduirons notre solution proposée. Par la suite, nous allons aborder une étude sur les différentes méthodologies de travail existantes afin de dégager celle la plus adéquate à notre projet.

1. Contexte générale du projet

Dans cette partie, nous allons présenter le cadre général du projet et ses objectifs.

1.1. Cadre du projet

Le présent travail s'inscrit dans le cadre du la matière projet fédère pour un but de nous préparer et s'entrainer à commencer une petite expérience bien pratique : un projet réel un web site e-learning. Tout cela est une sorte d'application d'une méthode pédagogique assite par notre faculté « L'Institut Supérieur d'Informatique et de Mathématiques de Monastir ».

Le projet a été effectué au sein d'une équipe de 6 jeunes étudiant durant la période de trois mois du 15 janvier 2024 au 15 Avril 2024.

1.2. Présentation de l'organisation d'accueil :

L'ISIMM est un établissement d'enseignement supérieur public tunisien spécialisé dans les sciences et technologies. Il a été créé en 2002 et propose des formations allant de la licence au doctorat.



Figure 1 Logo du "ISIMM"

Formations : L'ISIMM propose des formations dans les domaines de l'informatique, des mathématiques et de l'électronique.

Objectifs : L'ISIMM vise à former des cadres qualifiés et innovants capables de répondre aux besoins du marché du travail.

Réputation : L'ISIMM est un établissement réputé pour la qualité de ses formations et la pertinence de ses programmes.

L'ISIMM est un établissement dynamique et en constante évolution. Il est bien connecté à son environnement socio-économique et industriel et favorise la mobilité nationale et internationale des étudiants.

1.3. Objectif du projet

Dans un secteur où les contraintes de formation en présentiel sont importantes (disponibilité, le frais de déplacement, etc.), le concept de la formation à distance offre des réponses complémentaires et une plus grande flexibilité à l'apprenant qui pourra se former à son rythme en fonction de ses disponibilités. Notre objectif consiste à créer une plateforme du e-Learning qui répond au besoin du tuteur et des apprenants à la fois et de manière à :

- Favoriser une pédagogie socioconstructiviste (collaboration, activités, réflexion critique, etc.).
- Garantir plus de souplesse pour la planification des formations à l'aide d'une interface ergonomique, légère, efficace, compatible -sans nécessité d'un navigateur récent- et simple à manipuler pour les différents utilisateurs,
- Faciliter l'accès aux ressources et services,
- Faciliter le partage, la production, la communication et la collaboration à distance,
- Offrir un accès à un Ai chatbot.

2. Etude de l'existant

L'étude de l'existant consiste à auditer les solutions existantes pour s'inspirer et pour raffiner de plus l'idée immergeant.

2.1. Solutions existantes

2.1.1. Coursera

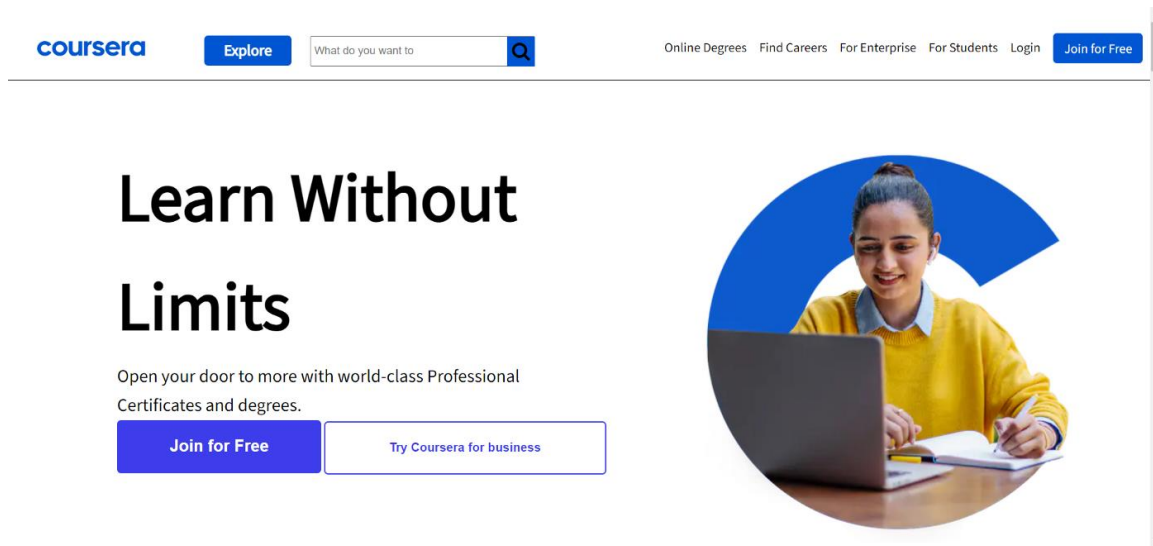


Figure 2 Page d'accueil du site « Coursera »

Les caractéristiques du site web Coursera comprennent plusieurs aspects qui contribuent à son succès en tant que plateforme d'apprentissage en ligne. Voici quelques-unes des caractéristiques clés de Coursera :

- Cours Universitaires (Coursera collabore avec des universités et des institutions éducatives)
- Certificats Vérifiés

Structure Pédagogique

- Projets Pratiques
- Flexibilité d'Apprentissage
- Accès à des Universités de Premier Plan
- Collaborations avec des universités et des entreprises

2.1.2. Kaggle

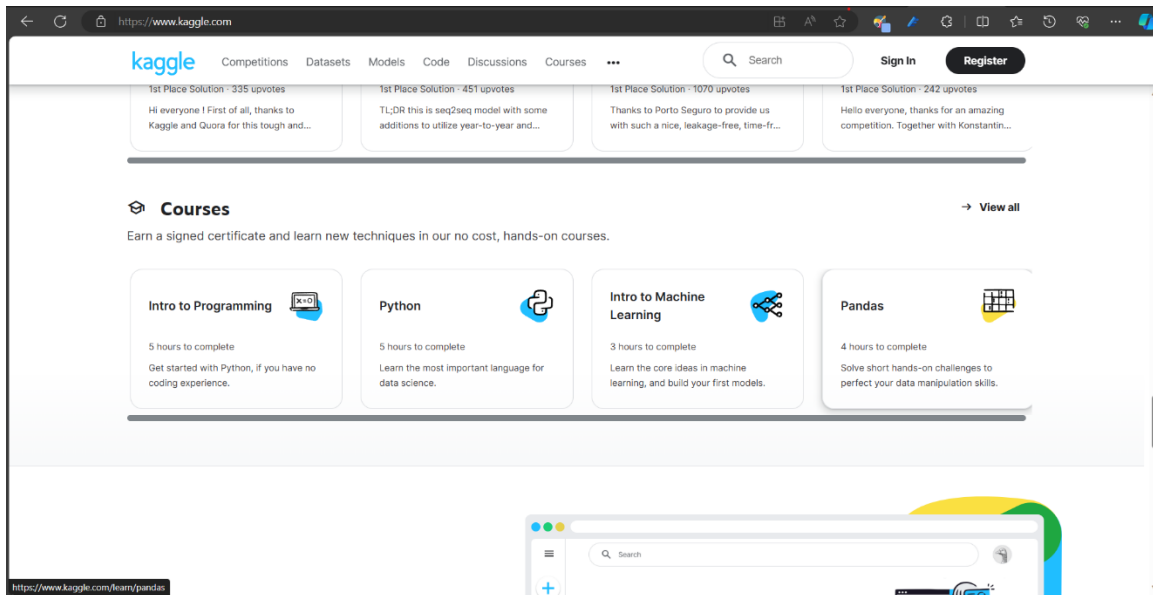


Figure 3 Page d'accueil du site « Kaggle »

Kaggle est une plateforme en ligne spécialisée dans les compétitions de science des données et l'apprentissage automatique. Voici quelques caractéristiques clés de Kaggle :

- Compétitions de Science des Données
- Ensembles de Données (vaste collection d'ensembles de données provenant de divers domaines)
- Notebooks (Les utilisateurs peuvent créer, partager et exécuter des notebooks interactifs directement sur la plateforme. Ces notebooks sont des environnements de programmation complets pour l'analyse de données et la création de modèles).
- Cours en Ligne (Kaggle propose des cours en ligne gratuits sur divers sujets liés à la science des données)
- Forums de Discussion (Kaggle propose des forums de discussion où les participants peuvent échanger des idées, poser des questions et discuter de problèmes spécifiques liés aux compétitions et aux projets).
- Points et Badges (Les utilisateurs peuvent gagner des points et des badges en participant activement à la communauté, en contribuant à des compétitions, en partageant des notebooks et en aidant d'autres membres).
- Plateforme Cloud (Kaggle propose une plateforme cloud qui permet aux utilisateurs d'exécuter des analyses et de construire des modèles directement en ligne).

2.1.3. Udemy

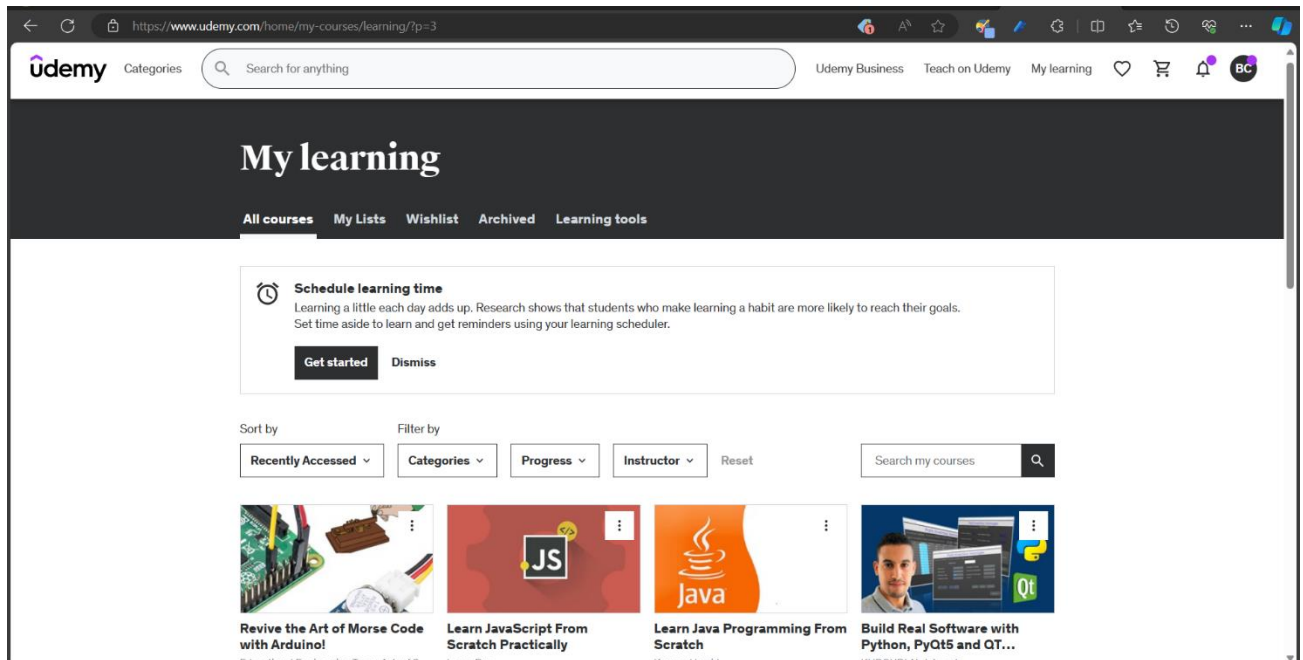


Figure 4 Page d'accueil du site « Udemy »

Udemy est une plateforme d'apprentissage en ligne qui propose une vaste gamme de cours sur divers sujets. Voici quelques-unes des caractéristiques clés d'Udemy :

- Variété de Cours
- Cours à la Demande
- Certificats de Complétion
- Instructeurs Indépendants (Les cours sur Udemy sont souvent créés et enseignés par des instructeurs indépendants)
- Prix Abordables
- Forums de Discussion (Chaque cours a son propre forum de discussion)
- Garantie de Remboursement
- Cours en Plusieurs Langues
- Évaluations et Commentaires (Les apprenants peuvent laisser des évaluations et des commentaires sur les cours)

2.1.4. Tableau comparative

Tableau 1 Tableau comparatif entre les solutions existantes

Critère	Coursera	Kaggle	Udemy
Objectif et Contenu	Cours universitaires diversifiés	Science des données, Machine Learning, Compétitions	Large gamme de sujets, y compris la programmation, la science des données, etc.
Approche pédagogique	Structuré avec modules hebdomadaires, devoirs, évaluations	Compétitions pratiques, cours interactifs	Cours à la demande, suivez à votre rythme
Certification	Certificats vérifiés par des institutions éducatives	Badges pour compétitions, mais pas de certifications officielles pour les cours	Certificats de complétion, moins reconnus que les diplômes universitaires
Communauté et Interactivité	Forums de discussion, projets de groupe	Forte communauté, partage de notebooks, discussions	Forums de cours, interaction avec instructeurs et apprenants
Coût	Certains cours gratuits, frais pour les certificats	Fonctionnalités de base gratuites, certains cours payants	Cours payants, prix variables en fonction du contenu et de l'instructeur
Expérience Utilisateur (UX)	Interface conviviale, navigation facile	Interface intuitive pour compétitions et cours, mais moins axée sur l'apprentissage traditionnel	Interface utilisateur conviviale, système de notation et de commentaires
Crédibilité des Cours	Cours dispensés par des institutions éducatives renommées	Axé sur les compétitions pratiques, moins de cours traditionnels	Cours créés par des instructeurs indépendants, diversité des approches pédagogiques

2.2. Critique de l'existant

Bien que les plateformes d'apprentissage en ligne aient considérablement transformé l'éducation, il subsiste des défis liés à la qualité, à la reconnaissance des certifications et à l'accessibilité. Chaque site mentionné précédemment ne donne pas l'opportunité aux utilisateurs de partager leurs cours ou ressources éducatifs et n'aide pas les gens qui n'ont pas une large connaissance comment créer un simple compte ou chercher facilement une cour.

2.3. Solution retenue

En s'appuyant sur d'autres solutions, et pour améliorer l'expérience utilisateur, notre plateforme e-learning doit offrir des cours sur une variété de sujets (mathématiques/ web dev/ python/ AI/ électrostatique...) avec exercices et examens avec des mises à jour en temps réel pour tenir les étudiants au courant de leurs cours, elle doit aussi offrir une interface simple mais conviviale dont on peut y accéder via site web ou via une application.

Notre plateforme doit aussi offrir un espace de partage de ressources éducatifs (des drives, des fiches de révision ,.....) et un espace de discussion qui permet aux étudiants de poser des questions ou de télécharger des ressources supplémentaires utiles avec un chatbot qui peut résoudre plusieurs problèmes (comment créer un compte, quel est le best-seller,...) pour mieux améliorer l'expérience d'utilisateur .

3. Méthodologie de travail

Avant de débiter le développement de chaque projet, le choix d'une méthodologie de travail est crucial pour garantir la création d'un site web éducatif fiable, adaptable et efficace. Pour orienter cette décision, nous avons comparé deux approches : la méthode 2TUP et la méthodologie agile Scrum.

3.1. Méthode 2TUP : Two Tracks Unified

La méthode 2 Tracks Unified est un processus de développement logiciel qui s'appuie sur la méthode du processus unifié, utilisant notamment UML, adoptant une approche itérative, se concentrant sur l'architecture et guidée par les cas d'utilisation.

Le 2TUP propose un cycle de développement en Y, séparant les aspects techniques des aspects fonctionnels. Il débute par une étude préliminaire identifiant les acteurs interagissant avec le système, les échanges de messages entre acteurs et système, la production du cahier des charges et la modélisation du contexte.

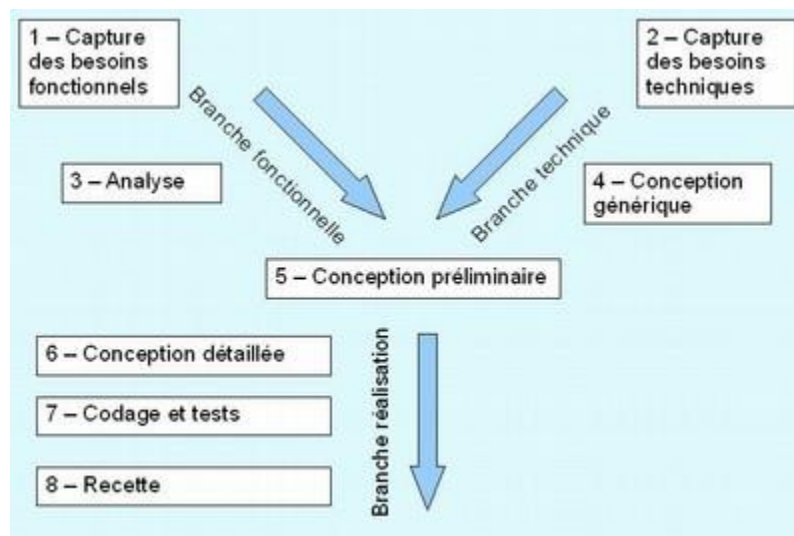


Figure 5 : Représentation de la méthodologie 2TUP

Le processus se déroule ensuite en trois phases essentielles, comme illustré dans la figure 5 :

- **Une branche fonctionnelle** : capturant les besoins fonctionnels, focalisée sur le métier des utilisateurs finaux.

- **Une branche technique** : capitalisant sur le savoir-faire technique indépendamment des fonctions à réaliser.
- **Une phase de réalisation** : réunissant les deux branches pour permettre la conception applicative et la livraison d'une solution adaptée aux besoins.

3.2. Méthode agile Scrum

La méthodologie agile Scrum, présentée dans la figure 6, découpe le projet en itérations appelées sprints, d'une durée de 2 à 4 semaines. Chaque sprint commence par une réunion de planification ("Sprint planning") pour définir ses objectifs et décomposer les besoins en plusieurs tâches. Le sprint se clôture par une démonstration du travail réalisé, évalué et validé.

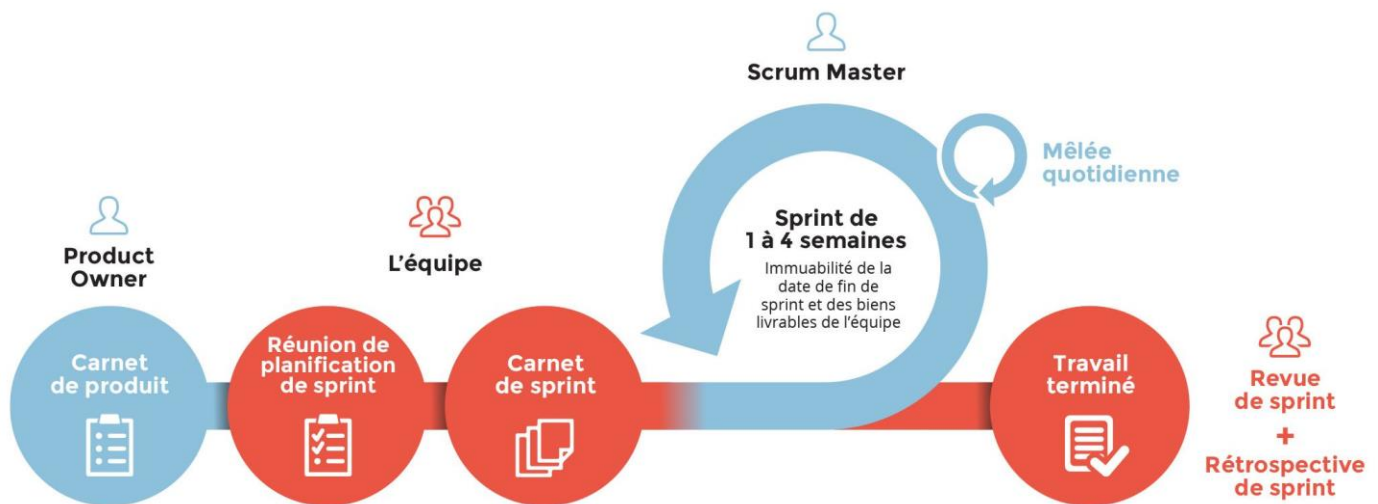


Figure 6 Représentation de la méthodologie SCRUM

Dans le contexte Scrum :

- **Responsable Produit (Product Owner)** : Représente les clients, détermine ce qui doit être réalisé.
- **Scrum Master** : Responsable du déroulement du processus, gestion du carnet du produit, garantit la motivation de l'équipe et l'efficacité de la collaboration.
- **Équipe projet** : S'organise pour produire les exigences prioritaires, livrant un incrément à la fin de chaque sprint.
- **User story** : Description d'un besoin fonctionnel du client.

- **Backlog du produit** : Besoins priorisés par le product owner et estimés par l'équipe, évoluant et s'affinant.
- **Sprint** : Période de 2 à 4 semaines pendant laquelle une version terminée et utilisable du produit est réalisée.
- **Backlog du sprint** : Sélection de tâches issues du backlog du produit pour construire l'objectif du sprint.
- **Daily Meeting** : Réunion quotidienne pour mettre à jour sur les réalisations, les problèmes rencontrés et les objectifs du jour.
- **Démonstration du sprint** : Évaluation et validation du travail réalisé à la fin de chaque sprint.

3.3. Méthodologie adoptée pour le projet de site web e-learning

Nous avons décidé d'adopter le processus 2TUP pour notre projet de site web e-learning. Cette décision s'explique par le respect du cadre du projet, basé sur un processus de développement bien défini, débutant par la détermination des besoins fonctionnels du système jusqu'à la conception et l'implémentation. Cette méthodologie correspond également à l'envergure et à la durée du projet.

Conclusion

Pour conclure ce chapitre, nous avons d'abord positionné notre projet dans son contexte global. Par la suite, nous avons examiné les solutions déjà disponibles et avons suggéré notre alternative. Finalement, nous avons terminé en exposant la méthodologie de développement que nous avons choisie.

Introduction

L'étape d'analyse et spécification des besoins est une étape indispensable pour comprendre les fonctionnalités que le système doit fournir. Dans ce chapitre nous présenterons les fonctionnalités fournies et les acteurs concernés par notre système. Nous allons exprimer les besoins fonctionnels et non fonctionnels sous forme de diagrammes de cas d'utilisation et des diagrammes de séquence.

1. Identification des acteurs

Un acteur est une entité externe qui définit le rôle joué par un utilisateur, humain ou non humain, qui interagit avec un système interactif.

Notre système comporte les acteurs suivants :

- **Utilisateur** : C'est un simple visiteur du site, il peut consulter les nouveautés sur la page d'accueil. Il a aussi le droit de créer un profil et devenir un Etudiant ou un Tuteur dans notre plateforme.
- **Administrateur** : C'est un abonné au site qui a l'accès à l'espace administratif de la plateforme. Il peut donner l'accès au site pour les étudiants ou les tuteurs via un code bien défini et organiser les emplois du temps du matières.
- **Tuteur** : Le tuteur représente un prof enregistré dans la base de données qui a certaines fonctionnalités pour gérer les ressources (cours, exercices, tests...) et visualiser le progrès du chaque cours ainsi des étudiants.
- **Etudiant** : C'est un utilisateur qui possède un profil sur le site, il peut consulter, modifier son profil. Il peut également souscrire, consulter, annuler une cour comme il peut consulter ses notes, ses certificats, participer à des live, ajouter des commentaires, participer à l'espace de discussion « forum ».

2. Modèle informationnelle de contexte

Cette phase consiste à comprendre le contexte du système. Il s'agit de présenter les cas d'utilisation dans un digramme.

2.1. Diagramme de contexte

La figure 7 illustre le diagramme de contexte de notre système.

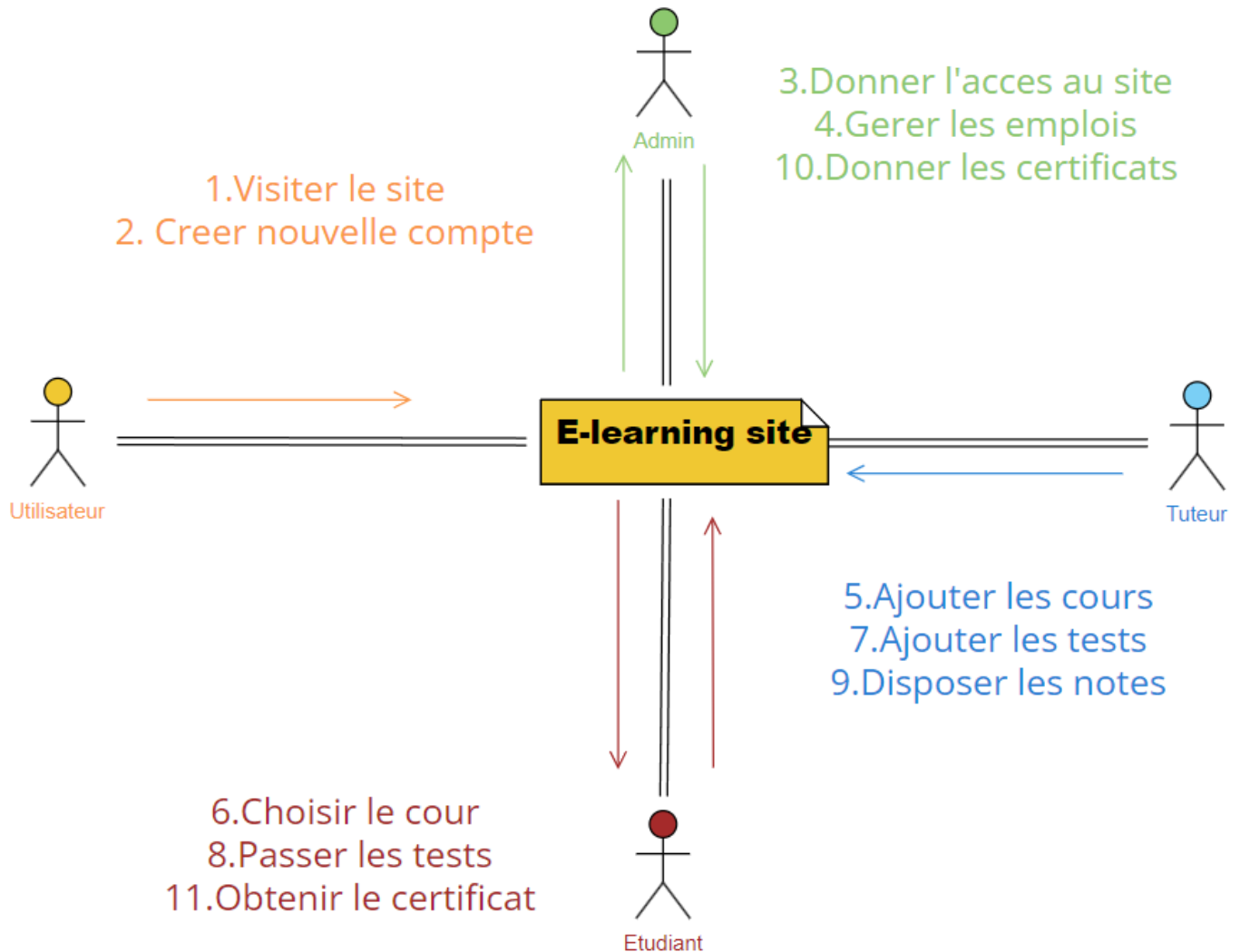


Figure 7 Diagramme de contexte

3. Capture des besoins

Dans cette partie, nous allons analyser les différents besoins fonctionnels et non fonctionnels que notre système cherche à satisfaire.

3.1. Besoins fonctionnels

Les besoins fonctionnels sont les fonctionnalités que notre système doit accomplir pour satisfaire les attentes de l'acteur de l'application web,

L'application doit permettre à l'**Administrateur** de :

- **Gérer des utilisateurs** : Création, modification et suppression des comptes utilisateur.
- **Gérer accès** : Donner ou retire l'accès au site (une code confidentiel « Physique » qui peut le personne inscrit au ISIMM soit Etudiant ou Tuteur l'avoir chez l'administration)
- **Gérer des notes et des certificats** : Enregistrement, publication des notes des étudiants, génération de rapports statistiques sur les performances et attribution des certificats.

L'application doit permettre à l'**Utilisateur (Visiteur)** de :

- **Accès aux informations publiques** : Consultation des actualités.
- **Accès aux ressources publiques** : Participation aux blogs : Lire et commenter les blogs publics.

L'application doit permettre à l'**Étudiant** de :

- **Gérer le profil et l'inscription** : Création et gestion du profil étudiant. Inscription aux cours.
- **Gérer les devoirs et les tests** : Soumission des devoirs, participation aux tests. Accéder aux notes : Consultation des notes attribuées, obtenir les certificats gagnés.
- **Contribuer des ressources** : permet de partager les fiches, cours personnelles avec les autres étudiants.
- **Communication et collaboration** : Communication avec les enseignants et les autres étudiants. Participation aux forums de discussion.
- **Accès au chatbot** : permet d'avoir de l'aide à partir un chatbot (Ce service sera disponible dans le futur).

L'application doit permettre au **Tuteur (Professeur)** de :

- **Gérer le profil** : Création et gestion du profil enseignant.
- **Gérer des cours attribués** : Attribution des devoirs et des tests, création et assignation des évaluations (Évaluation des devoirs et des tests), correction et notation des travaux.
- **Communication avec les étudiants** : Messagerie avec les étudiants. Participation aux forums de discussion.

3.2. Besoins non fonctionnels

Les besoins non-fonctionnels ont comme but de décrire les conditions requises qui permettent d'assurer le bon fonctionnement du système et optimiser les services qu'il fournit vis-à-vis l'utilisateur.

Notre application doit répondre aux critères suivants :

- **Sécurité** : L'application doit garantir à l'utilisateur connecté l'intégrité et la confidentialité de ses données.
La sécurité du système est assurée par l'authentification des clients par un login et un mot de passe crypté unique et réserver un vrai profil existant.
- **Performance** : Au niveau d'un système informatique la performance ne se définit pas uniquement par les temps de réponse résultants des applications aux utilisateurs, cette notion est plus vaste et comprend les aspects suivants :
 - Les temps de réponse
 - La disponibilité du système,
 - La faciliter de communiquer avec les professeurs,
 - La flexibilité d'interagir entre admin et professeur (disponibilité de professeur pour assister à un live)
 - 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 offre une interface conviviale et facile à utiliser,
 - La plateforme doit garantir la confidentialité, l'intégrité et la cohérence des données.
- **L'ergonomie** : On qualifie d'ergonomie un site web utile, accessible et qui donne satisfaction à l'utilisateur et qui a pour objectif d'améliorer les interactions homme-machine, la facilité d'utilisation.

4. Spécification des besoins

4.1. Diagramme de cas d'utilisation général

La figure 8 illustre le diagramme de cas d'utilisation général de notre système.

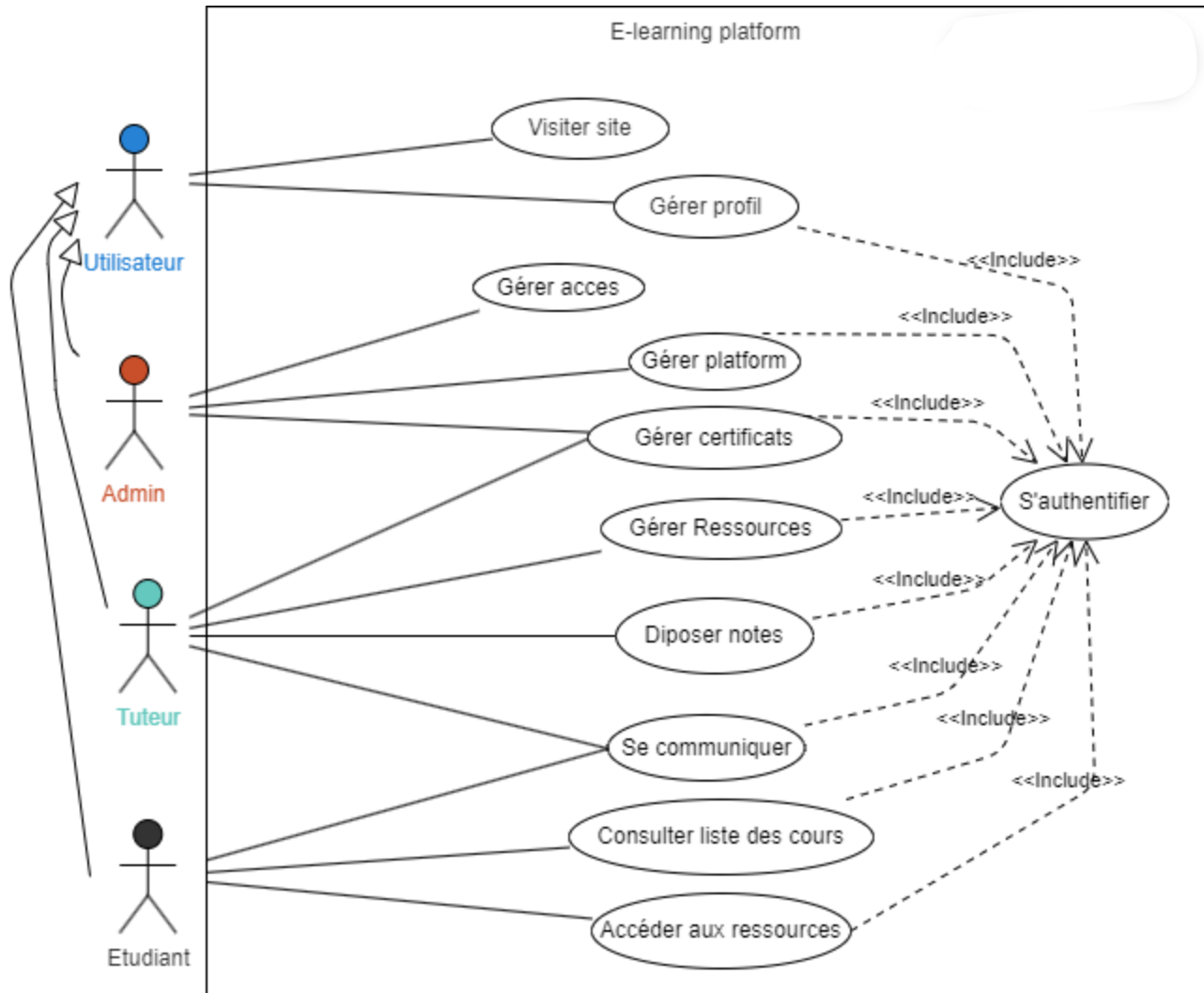


Figure 8 Diagramme de cas d'utilisation général

4.2. Diagramme de cas d'utilisation détaillés

Pour mieux comprendre les cas d'utilisation exprimés dans le diagramme de cas d'utilisation général, nous allons détailler dans ce qui suit les cas d'utilisation en fonction d'acteur.

4.2.1. Diagramme de cas d'utilisation de l'acteur Utilisateur

La figure 9 décrit le diagramme de cas d'utilisation de l'acteur Utilisateur

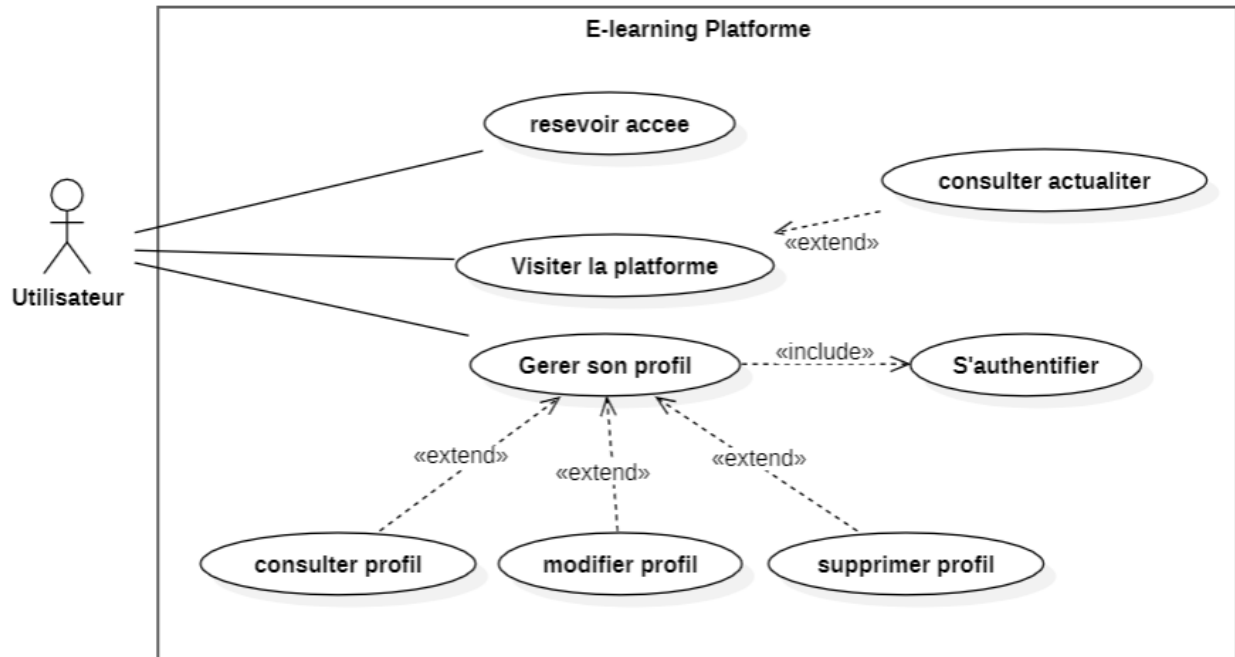


Figure 9 Diagramme de cas d'utilisation de l'acteur Utilisateur

4.2.2. Diagramme de cas d'utilisation de l'acteur Admin

La figure 10 décrit le diagramme de cas d'utilisation de l'acteur Admin

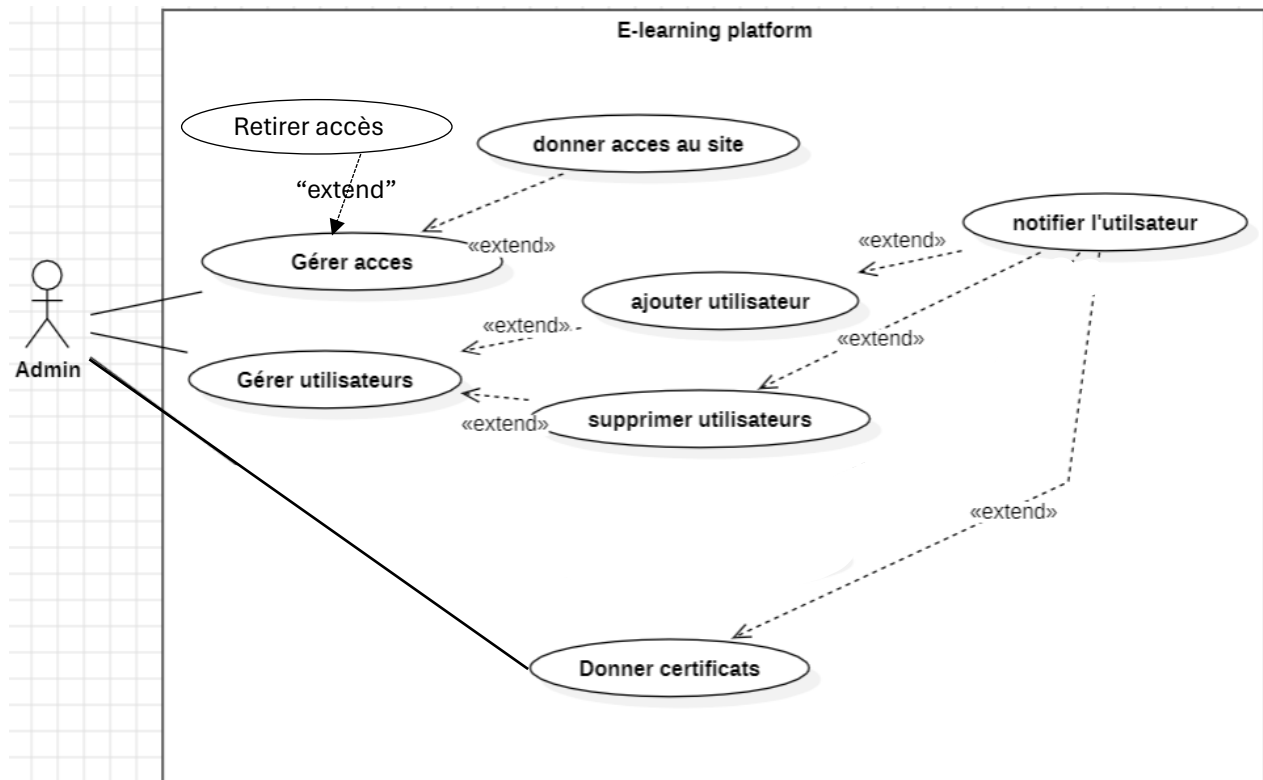


Figure 10 Diagramme de cas d'utilisation de l'acteur Admin

4.2.3. Diagramme de cas d'utilisation de l'acteur Tuteur

La figure 11 décrit le diagramme de cas d'utilisation de l'acteur Tuteur

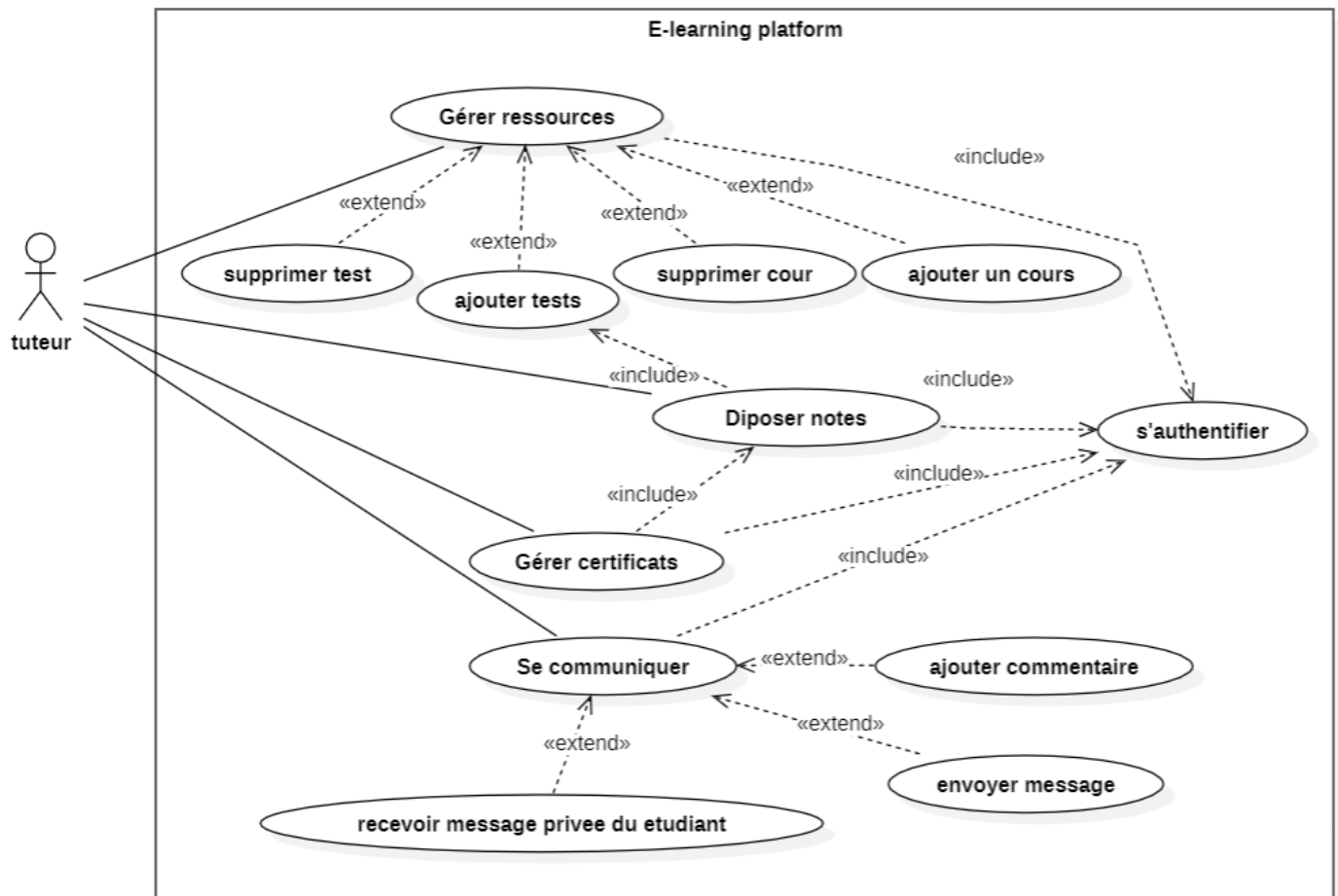


Figure 11 Diagramme de cas d'utilisation de l'acteur Tuteur

4.2.4. Diagramme de cas d'utilisation de l'acteur Etudiant

La figure 12 décrit le diagramme de cas d'utilisation de l'acteur Etudiant

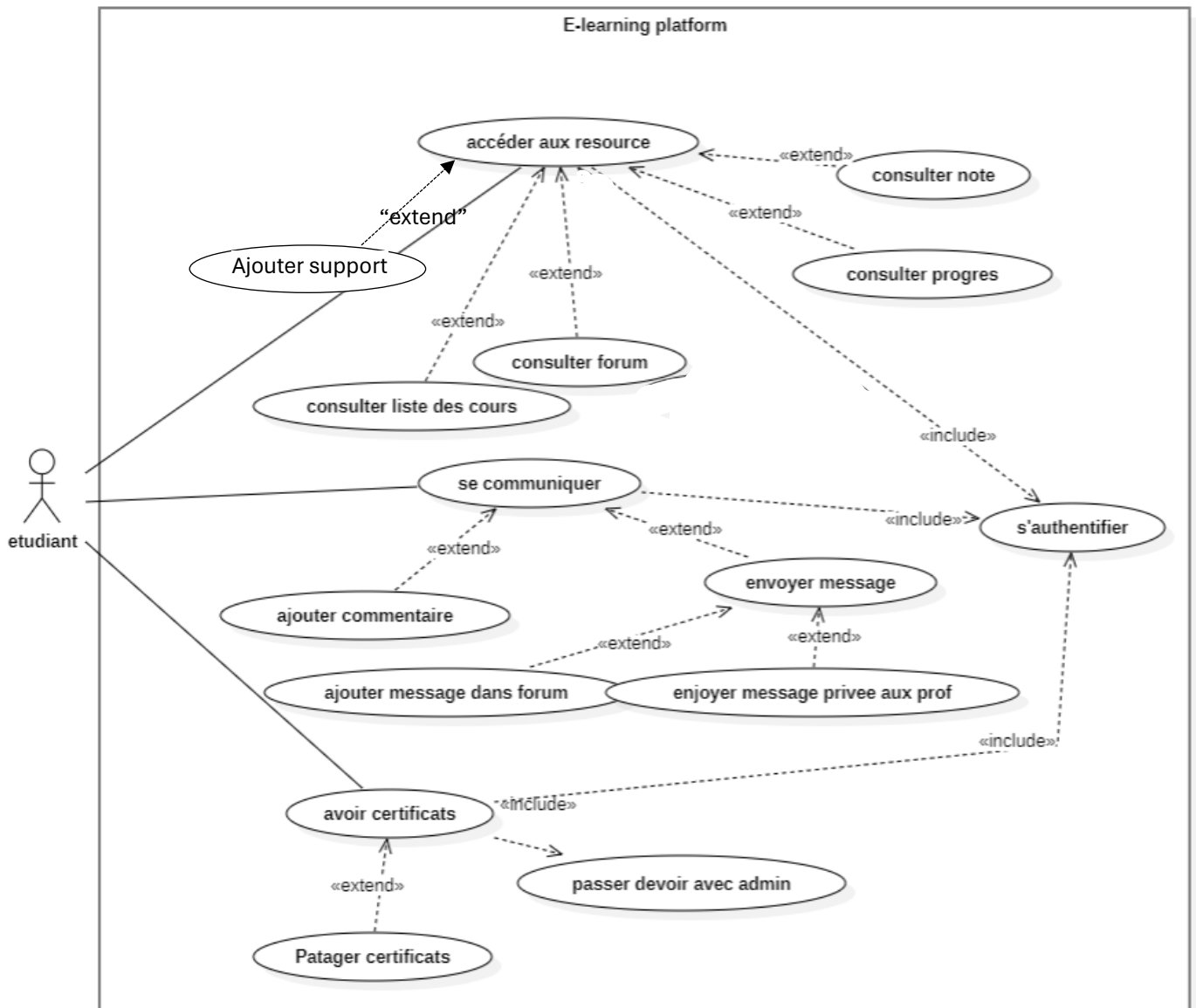


Figure 12 Diagramme de cas d'utilisation de l'acteur Etudiant

5. Description des cas d'utilisation

Dans cette partie nous allons citer la description textuelle de quelques cas d'utilisation principaux avec le diagramme de séquence qui nous permette de montrer les interactions des éléments du système entre eux et avec les acteurs.

5.1. Cas d'utilisation « S'authentifier »

5.1.1. Description textuelle

Tableau 2 Description textuelle du cas d'utilisation "S'authentifier"

Titre	S'authentifier
Acteur principal	Utilisateur, Administrateur, Tuteur, Étudiant
Description	Le système d'authentification permet aux différents utilisateurs de la plateforme d'accéder à leur compte de manière sécurisée ou de créer un nouveau compte.
Pre-condition	<ul style="list-style-type: none"> • L'utilisateur (Tuteur, Étudiant) doit avoir un code de sécurité pour créer un compte sur la plateforme. • L'utilisateur non authentifié ou l'utilisateur doit avoir un compte sur le site. L'utilisateur doit appartenir à l'ISIMM.
Scenario principal	<ol style="list-style-type: none"> 1. L'utilisateur saisit le code envoyé par l'administrateur (Tuteur, Étudiant). 2. L'utilisateur saisit son e-mail. 3. L'utilisateur saisit son mot de passe. 4. Le système valide les informations en cliquant sur le bouton Se connecter. 5. Le système vérifie la validité des champs. 6. L'utilisateur affiche la page d'accueil du site.
Scenario alternative	<ol style="list-style-type: none"> 1. L'utilisateur introduit une information incorrecte. 2. Le système affiche un message d'erreur et demande à l'utilisateur de vérifier l'ensemble des champs.
Post-condition	<ul style="list-style-type: none"> ➤ L'utilisateur est authentifié. ➤ La plateforme donne l'accès aux nouveaux champs.
Exception	<ul style="list-style-type: none"> ➤ Si les champs ne sont pas valides, le système affiche un message d'erreur "Donnée non valide". ➤ Si le système ne trouve pas l'e-mail et le mot de passe de l'utilisateur qui veut s'authentifier, il affiche un message d'erreur "L'utilisateur n'existe pas". ➤ Si le mot de passe ne correspond pas à l'e-mail saisi, le système affiche "Mot de passe incorrect". ➤ Si le code de sécurité est invalide, le système affiche "Code de sécurité invalide".

5.1.2. Diagramme de séquence d'analyse

Le diagramme de séquence représente dans la Figure 13 décrit le scénario d'authentification d'un utilisateur.

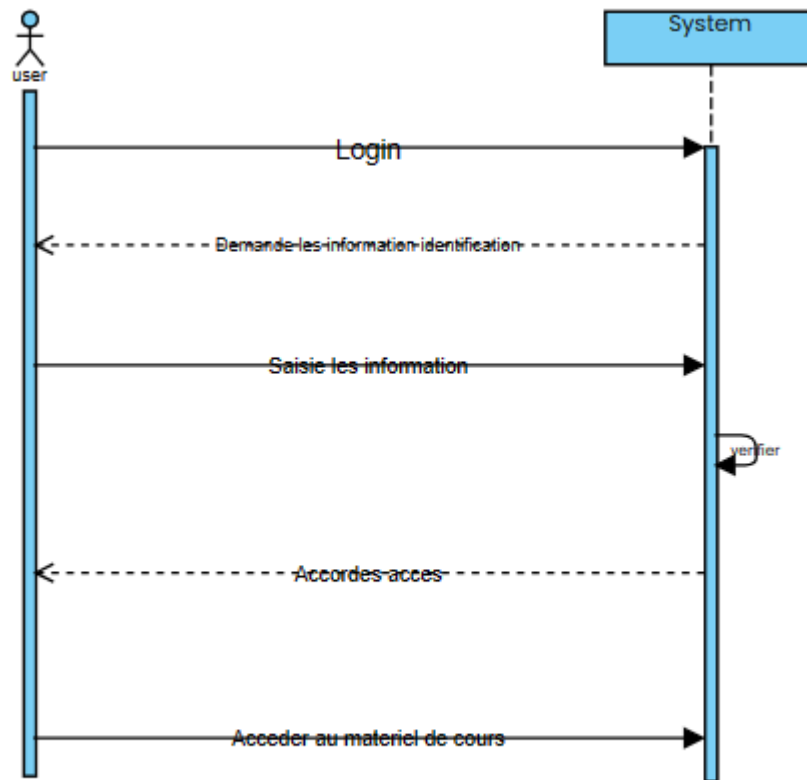


Figure 13 Diagramme de séquence du cas d'utilisation "s'authentifier"

5.2. Cas d'utilisation « Gérer ressources »

5.2.1. Description textuelle

Tableau 3 Description textuelle du cas d'utilisation "Gérer ressources"

Titre	Gérer ressources
Acteur principal	Tuteur
Description	Ce cas d'utilisation permet aux tuteurs d'ajouter, supprimer, modifier et accéder aux ressources (cours, exercices, examens)
Pré-condition	Le tuteur doit être authentifié
Scenario principal	<ol style="list-style-type: none"> 1. Le tuteur ajoute les ressources 2. Le tuteur supprime les ressources 3. Le tuteur modifie les ressources
Scenario alternative	<ol style="list-style-type: none"> 1. Ajouter ressources : <ul style="list-style-type: none"> • Le système vérifie que le tuteur est authentifié. • Le système vérifie que les ressources déposées n'existent pas déjà. • Type de ressources n'est pas compatible. 2. Supprimer ressources : <ul style="list-style-type: none"> • Le système vérifie que le tuteur est authentifié. • Le système vérifie que les ressources à supprimer existent. 3. Modifier ressources : <ul style="list-style-type: none"> • Le système vérifie que le tuteur est authentifié. • Le système vérifie que les ressources à modifier existent.
Post-condition	<ul style="list-style-type: none"> ➤ Les ressources sont bien ajoutées ➤ Les ressources sont bien modifiées ➤ Les ressources sont bien supprimées
Exception	<ul style="list-style-type: none"> ➤ Ressource existe déjà. ➤ Si la suppression ne se termine pas avec succès, le système affiche un message d'erreur "Erreur lors de suppression". ➤ Si la modification ne se termine pas avec succès, le système affiche un message d'erreur "Erreur lors de modification".

5.2.2. Diagramme de séquence d'analyse

Le diagramme de séquence représente dans la Figure 14 décrit le scénario de gérer les ressources.

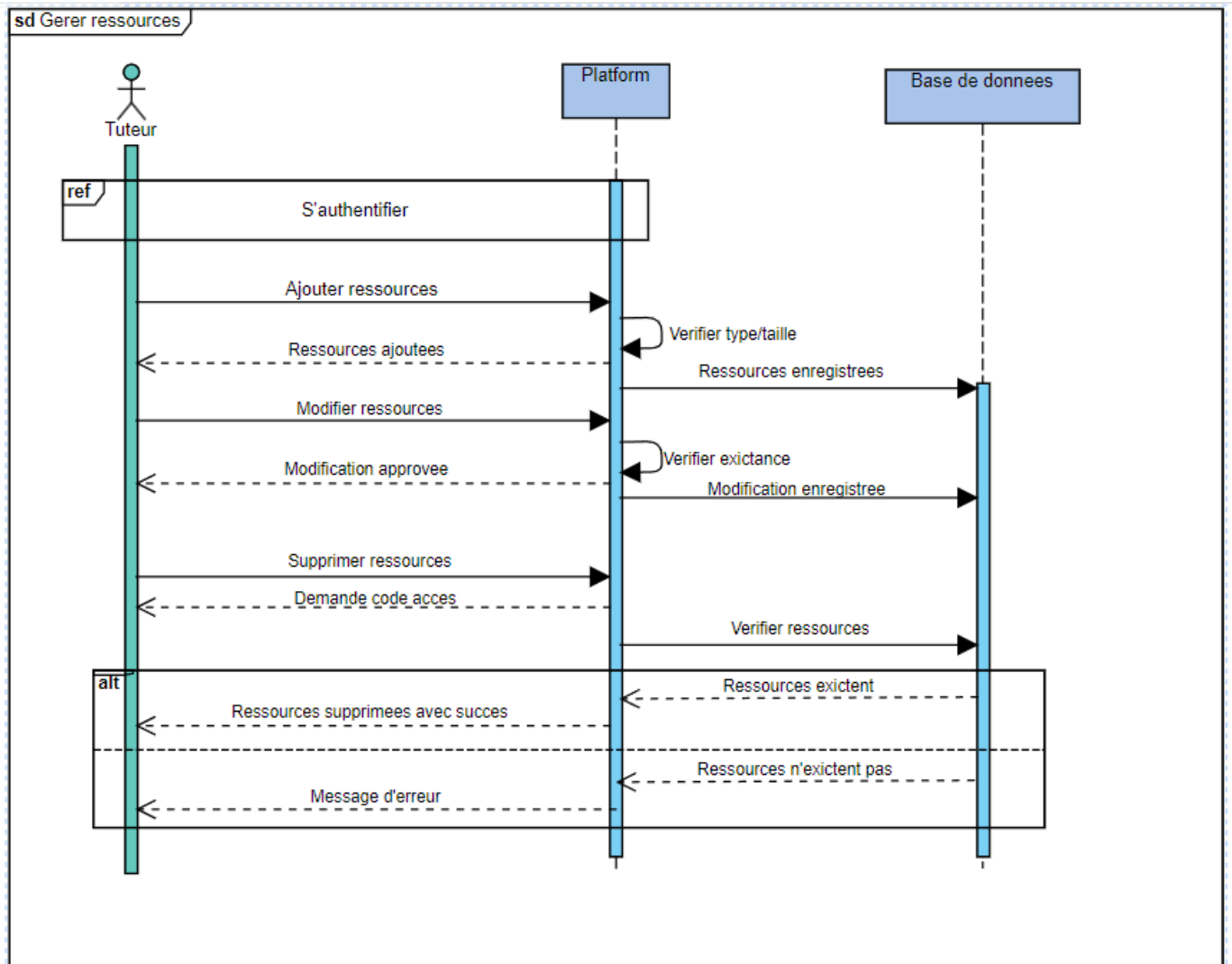


Figure 14 Diagramme de séquence du cas d'utilisation "Gérer ressources"

5.3. Cas d'utilisation « Gérer certificats »

5.3.1. Description textuelle

Tableau 4 Description textuelle du cas d'utilisation "Gérer certificats"

Titre	Gérer certificats
Acteur principal	Administrateur
Description	Ce cas d'utilisation permet à l'administrateur de donner des certificats aux étudiants dont la moyenne est supérieure à la moyenne.
Pré-condition	<ol style="list-style-type: none"> 1. Le tuteur dépose les notes 2. La moyenne des notes est supérieure ou égale à la moyenne de certification
Scénario principal	<ol style="list-style-type: none"> 1. Donner certificat : ce scénario débute quand l'administrateur veut donner un certificat aux étudiants après la fin de leurs examens, le processus suit les étapes suivantes : <ul style="list-style-type: none"> • L'administrateur vérifie que les notes sont déposées par les professeurs. • L'administrateur vérifie la liste des étudiants dont la moyenne est supérieure à la moyenne de certification. • L'administrateur envoie les certificats aux étudiants qui ont réussi.
Scénario alternative	<ol style="list-style-type: none"> 1. L'administrateur vérifie que les moyennes sont déposées : <ul style="list-style-type: none"> ➤ L'administrateur choisit d'afficher les moyennes. ➤ Les moyennes disponibles sont affichées dans la plateforme. ➤ Le tuteur ne donne pas les notes. 2. L'administrateur vérifie la liste des étudiants qui ont réussi : <ul style="list-style-type: none"> ➤ L'administrateur choisit d'afficher les moyennes. ➤ L'administrateur choisit les étudiants dont la moyenne est supérieure à la moyenne de certification. ➤ La cour n'a pas du certificat. 3. L'administrateur envoie les certificats : <ul style="list-style-type: none"> ➤ L'administrateur dépose les certificats. ➤ Le système envoie les certificats. ➤ Aucun étudiant à la moyenne.
Post-condition	Les certificats sont bien reçus par les étudiants.
Exception	<ul style="list-style-type: none"> ➤ Moyennes ne sont pas déposées. ➤ Si l'importation des certificats ne se termine pas avec succès, le système affiche un message d'erreur "Erreur lors de l'importation".

5.3.2. Diagramme de séquence du cas d'utilisation "Gérer certificats"

Le diagramme de séquence représente dans la Figure 15 décrit le scénario d’obtenir un certificat par l’étudiant (Utilisateur).

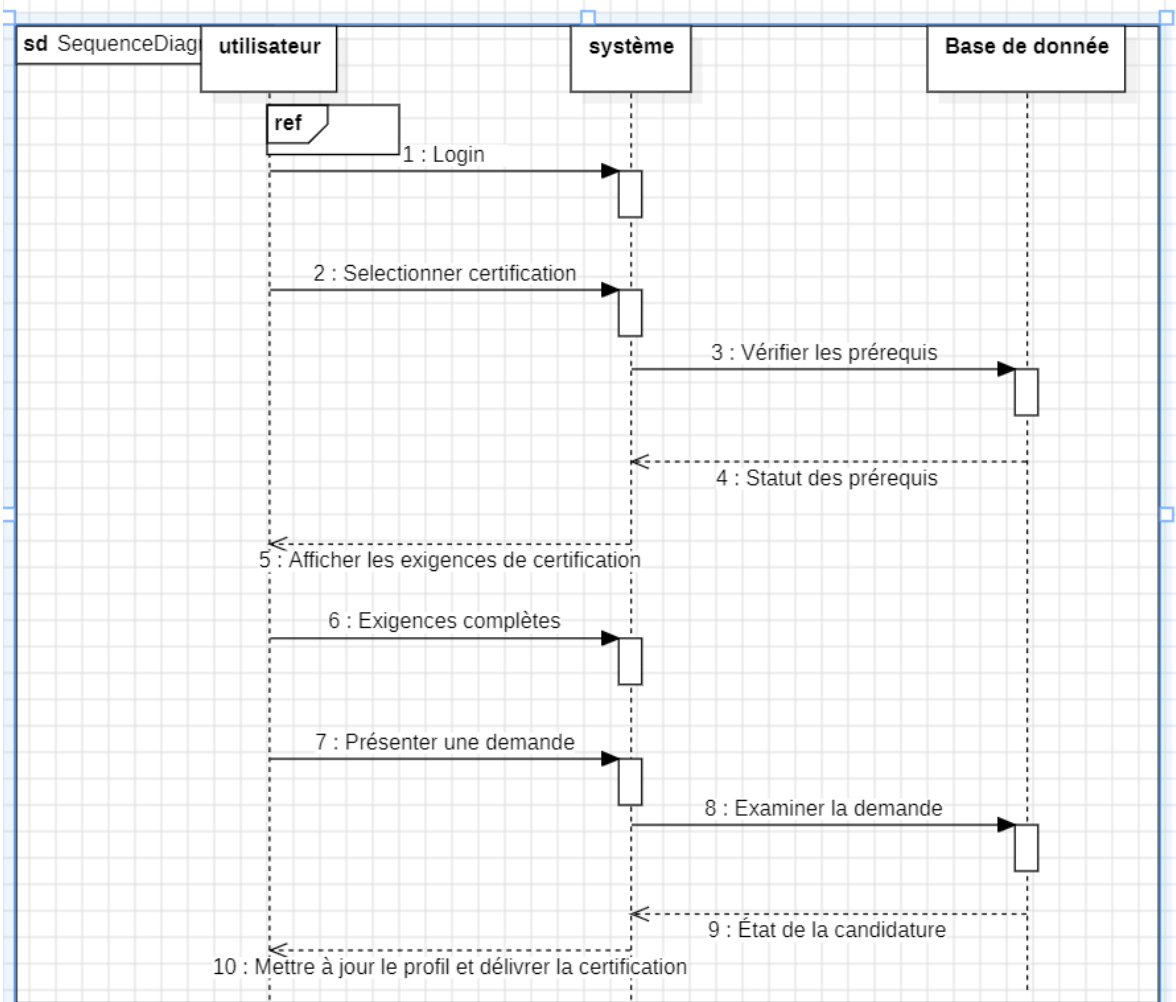


Figure 15 Diagramme de séquence du cas d'utilisation "Gérer certificats"

5.4. Cas d’utilisation « Se communiquer »

5.4.1. Description textuelle

Tableau 5 Description textuelle du cas d'utilisation "Se communiquer"

Titre	Se communiquer
Acteur principal	Tuteur, Étudiant
Description	Le système de communication permet aux tuteurs et aux étudiants d’échanger des informations, des documents, des feedbacks et des questions sur les cours et les activités pédagogiques. Le système offre

	différents modes de communication, tels que le chat, le forum, le courriel et la visioconférence.
Pre-condition	<ul style="list-style-type: none"> ➤ Le tuteur et l'étudiant doivent être authentifiés sur la plateforme. ➤ Le tuteur et l'étudiant doivent être inscrits au même cours ou à la même activité pédagogique.
Scenario principal	<ol style="list-style-type: none"> 1. Le tuteur ou l'étudiant choisit le mode de communication qu'il souhaite utiliser (chat, forum, courriel ou visioconférence). 2. Le système affiche la liste des destinataires possibles selon le mode choisi. 3. Le tuteur ou l'étudiant sélectionne le ou les destinataires de son message. 4. Le système ouvre une fenêtre de communication avec le ou les destinataires. 5. Le tuteur ou l'étudiant rédige et envoie son message, en y joignant éventuellement des documents. 6. Le système transmet le message au ou aux destinataires, en respectant les règles de confidentialité et de sécurité. 7. Le ou les destinataires reçoivent le message et peuvent y répondre selon le mode de communication utilisé.
Scenario alternative	<ol style="list-style-type: none"> 1. L'utilisateur se connecter et rencontre un problème avec le système de communication. 2. Le système affiche un message d'erreur et propose d'autres modes de communication disponibles.
Post-condition	<ul style="list-style-type: none"> ➤ Un nouveau chat est créé. ➤ Espace de discussion privée ouvert.
Exception	<ul style="list-style-type: none"> ➤ Si le tuteur ou l'étudiant n'est pas authentifié sur la plateforme, le système affiche le message d'erreur "Vous devez vous connecter pour accéder au système de communication". ➤ Si le tuteur ou l'étudiant n'est pas inscrit au même cours ou à la même activité pédagogique que le destinataire, le système affiche le message d'erreur "Vous ne pouvez pas communiquer avec ce destinataire". ➤ Si le mode de communication choisi n'est pas disponible ou compatible avec le destinataire, le système affiche le message d'erreur "Ce mode de communication n'est pas possible avec ce destinataire". ➤ Si le message ou le document envoyé est trop volumineux, le système affiche le message d'erreur "La taille du message ou du document dépasse la limite autorisée". ➤ Si le message ou le document envoyé contient des contenus inappropriés, le système affiche le message d'erreur "Le message ou le document contient des contenus interdits".

5.4.2. Diagramme de séquence du cas d'utilisation "Se communiquer"

Le diagramme de séquence représente dans la Figure 16 décrit le scenario de communication entre une groupe d'utilisateur (Tuteur/Etudiant) et deux utilisateurs séparément.

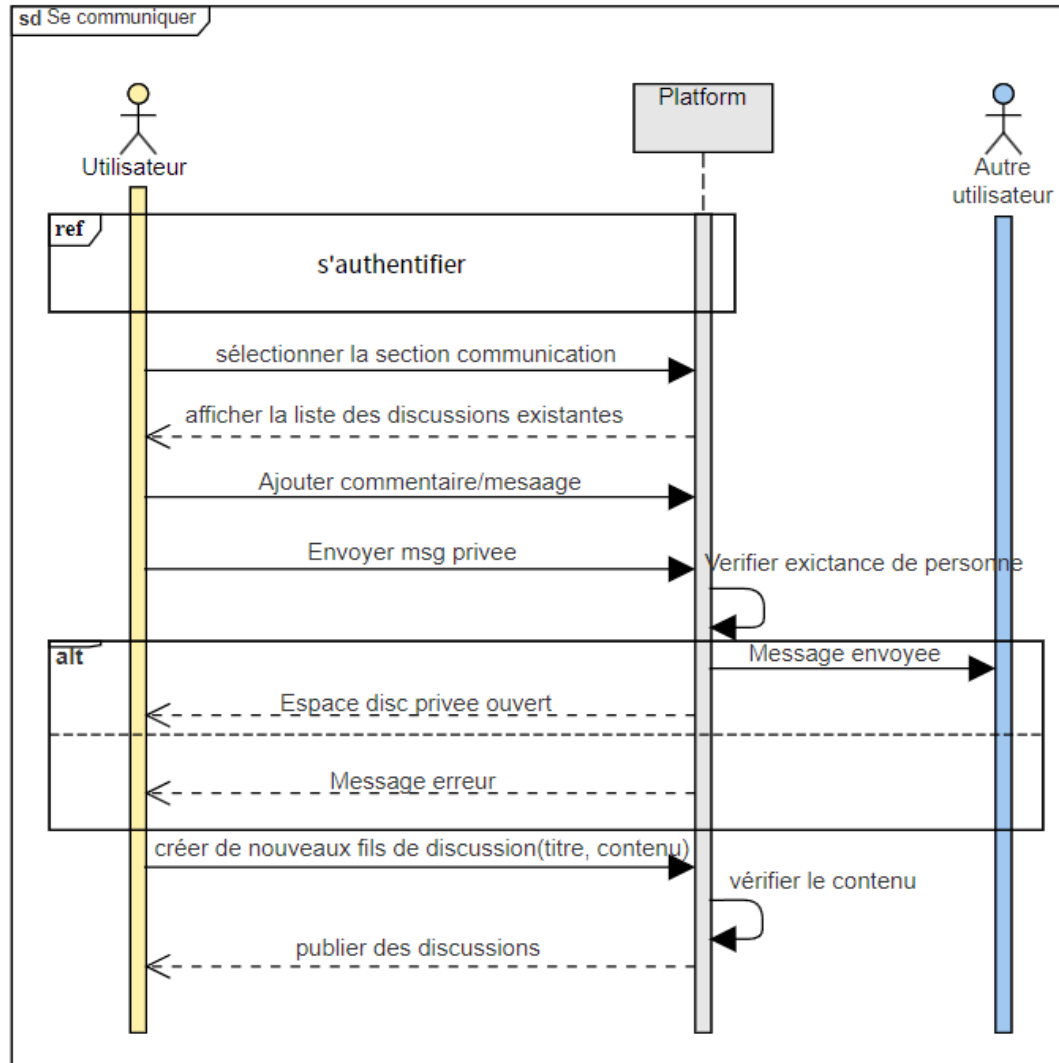


Figure 16 Diagramme de séquence du cas d'utilisation "se communiquer"

Conclusion

Dans ce chapitre, nous avons identifié les acteurs et les besoins fonctionnels et non fonctionnels pour notre plateforme. Puis, nous avons présente et détaille les différents cas d'utilisation. Et enfin nous avons présente quelques diagrammes de séquences acteur-système.

Introduction

Dans ce chapitre nous allons entamer une partie importante du développement de l'application qui constitue un pont entre la spécification et la réalisation. Nous commencerons par la présentation d'architecture générale de notre application ensuite la conception générale puis la conception détaillée comprenant les vues statiques via les diagrammes de classes. En fin nous clôturerons ce chapitre par quelques fonctionnalités de l'application à l'aide des maquettes.

1. Architecture Applicative

2. Conception de Base de Données

La conception d'une base de données est l'organisation des données selon un modèle.

Donc, Nous commençons par présenter le modèle conceptuel de données (MCD) et par la suite le modèle Logique de données (MLD).

2.1. Modèle Conceptuel de Données (MCD)

Un MCD décrit les différentes entités ainsi que les relations qui existent entre elles. La figure 17 illustre le modèle conceptuel de notre base de données. Le tableau 3.1 donne les détails de description de chaque entité.

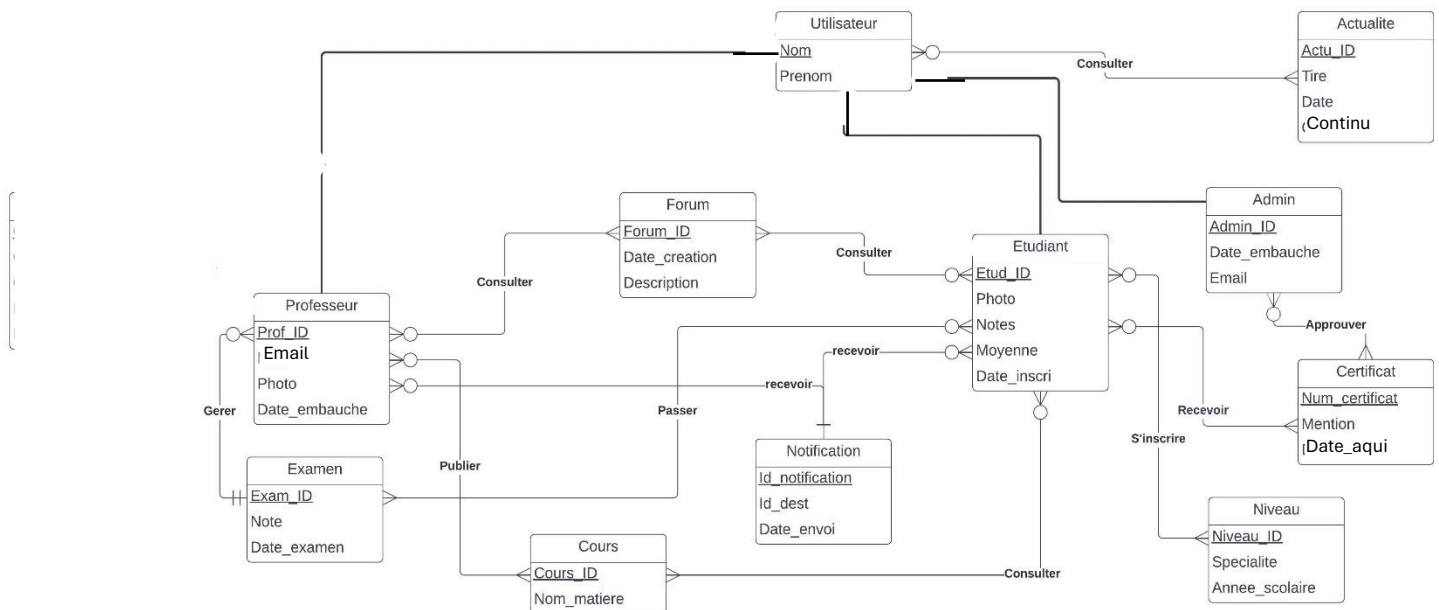


Figure 17 Modèle Conceptuel de données (MCD)

Tableau 6 Les entités du modèle conceptuel de données

Entités	Description	Attributs et type
Utilisateur	Tout utilisateur de la plateforme, ainsi les simples visiteurs.	- Nom : String - Prénom : String
Admin	Elle hérite la classe « Utilisateur » et contient les supplémentaires attributs relatifs à l'admin. L'admin a seulement le droit de donner les codes d'accès aux Etudiant et Tuteur.	- Admin_ID : Int - Date_Embauche : Date - Email : email
Tuteur (Professeur)	Elle hérite la classe « Utilisateur » et contient les supplémentaires attributs relatifs au Tuteur.	- Prof_ID : Int (code d'accès attribuer par l'admin) - Email : email - Photo : Binary - Date_Embauche : Date
Etudiant	Elle hérite la classe « Utilisateur » et contient les supplémentaires attributs relatifs à l'Etudiant.	- Etud_ID : Int (code d'accès attribuer par l'admin) - Photo : Binary - Notes : Float - Moyenne : Float - Date_inscription : Date
Actualité	Offre aux utilisateurs un accès rapide et efficace aux dernières informations, mises à jour et événements pertinents.	- Actualite_ID : Int - Titre : String - Date_publication : date - Continu : String
Certificat	Offre aux Etudiants la possibilité de valider leurs compétences et leurs réalisations grâce à des certificats officiels.	- Cetificat_ID : Int - Nom_certificat : String - Mention : String - Date_aqui : Date
Niveau	Permet aux Etudiants de suivre leur progression et d'avancer dans différents niveaux de compétence ou de maîtrise dans leurs domaines d'étude.	- Niveau_ID :int - Spécialité : String - Année_scolaire :Date
Cour	Contient les ressources (TD, TP, Fiche,) pour chaque Etudiants.	- Cour_id : Int - Nom_cour : String
Examen	Publier par Tuteur et obligatoire de le passer pour chaque obtenir pour procéder son parcours et obtient un certificat.	- Exam_ID : Int - Note : Float - Date_examen : Date
Notification	Elle dépend du chaque Utilisateur et permet lui informer du contenu d'un acte ou d'une action.	- ID_notification :Int - ID_destinataire :Int - Date_envoi : Date
Forum	Environnement pour les Etudiants et Tuteur pour se communiquer ensemble.	- Forum_ID : Int - Date_creation : Date - Description : String

2.2. Modèle Logique de Données (MLD)

Un modèle MCD peut être organisé selon différents modèles logiques de données. Dans notre projet, nous avons opté pour le modèle relationnel qui répond bien à nos besoins non fonctionnels.

En appliquant les règles de transformations du modèle E/A vers un modèle relationnel, nous obtenons le schéma relationnel suivant, illustré dans la figure 18.

Nous distinguons 14 relations, où 8 relations correspondent aux 8 entités du modèle conceptuel.

Utilisateur (Nom, Prénom)

Professeur (Prof_ID, Nom, Prénom, Email, Date_embauche, Photo)

Etudiant (Etud_ID, Nom, Prénom, Date_inscription, Moyenne, #Niveau_ID)

Admin (Admin_ID, Date_embauche, Email)

Niveau (Niveau_ID, Spécialité, Année_scolaire)

Cours (Cours_ID, Nom_matière, #Prof_ID)

Examen (Exam_ID, Note, Date_examen, #Prof_ID)

Forum (Forum_ID, Date_creation, Description)

Certificat (Num_certificat, Mention, Date_aqui, #Etud_ID, #Admin_ID)

Actualité (Actu_ID, Titre, Date, Continu)

Notification (ID_Notification, Date_envoi, #ID_Dest) « ID Dest=soit Etudiant ou Tuteur »

EtudExam (Etud_ID, Exam_ID)

EtudCour (Etud_ID, Cours_ID)

ProfForum (Prof_ID, Forum_ID)

EtudForum (Etud_ID, Forum_ID)

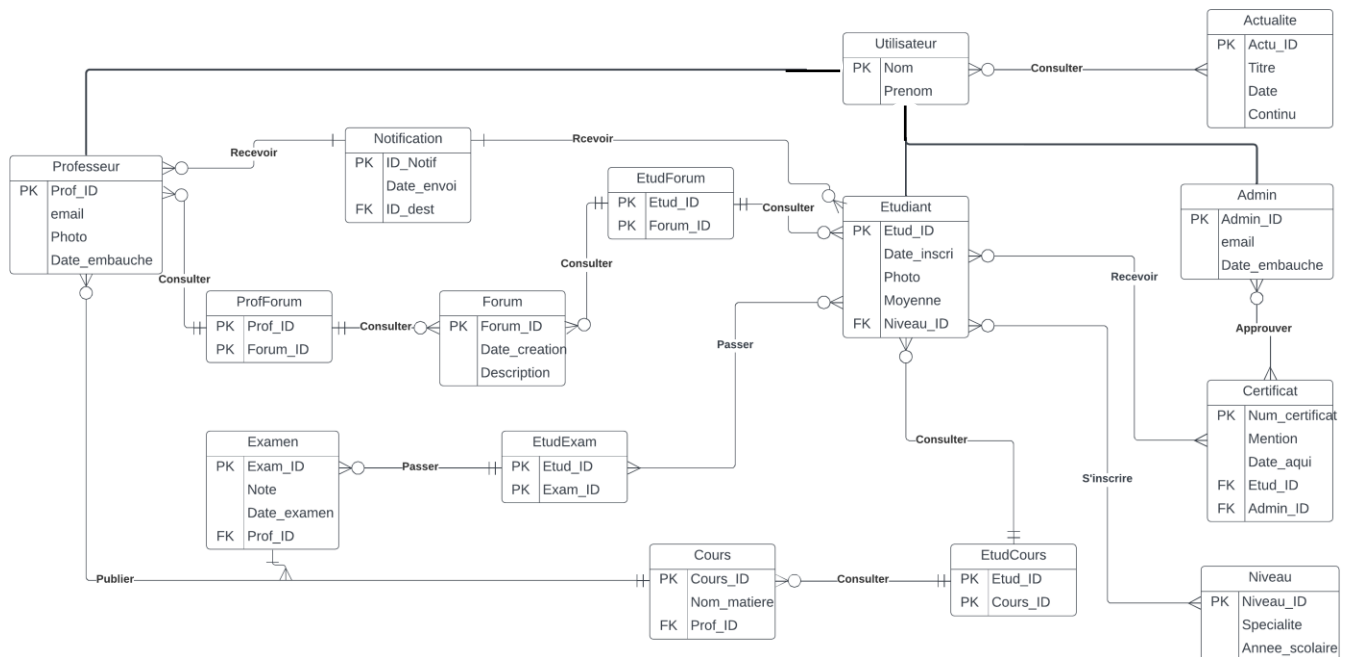


Figure 18 Le Modèle Logique de Données (MLD)

3. Conception logicielle

3.1. Vue statique : Diagrammes de classes

3.1.1. Diagramme de classe général

La figure 19 présente un diagramme de classes générale du notre plateforme qui contient toutes les informations telles que les classes, les méthodes, les associations et les propriétés.

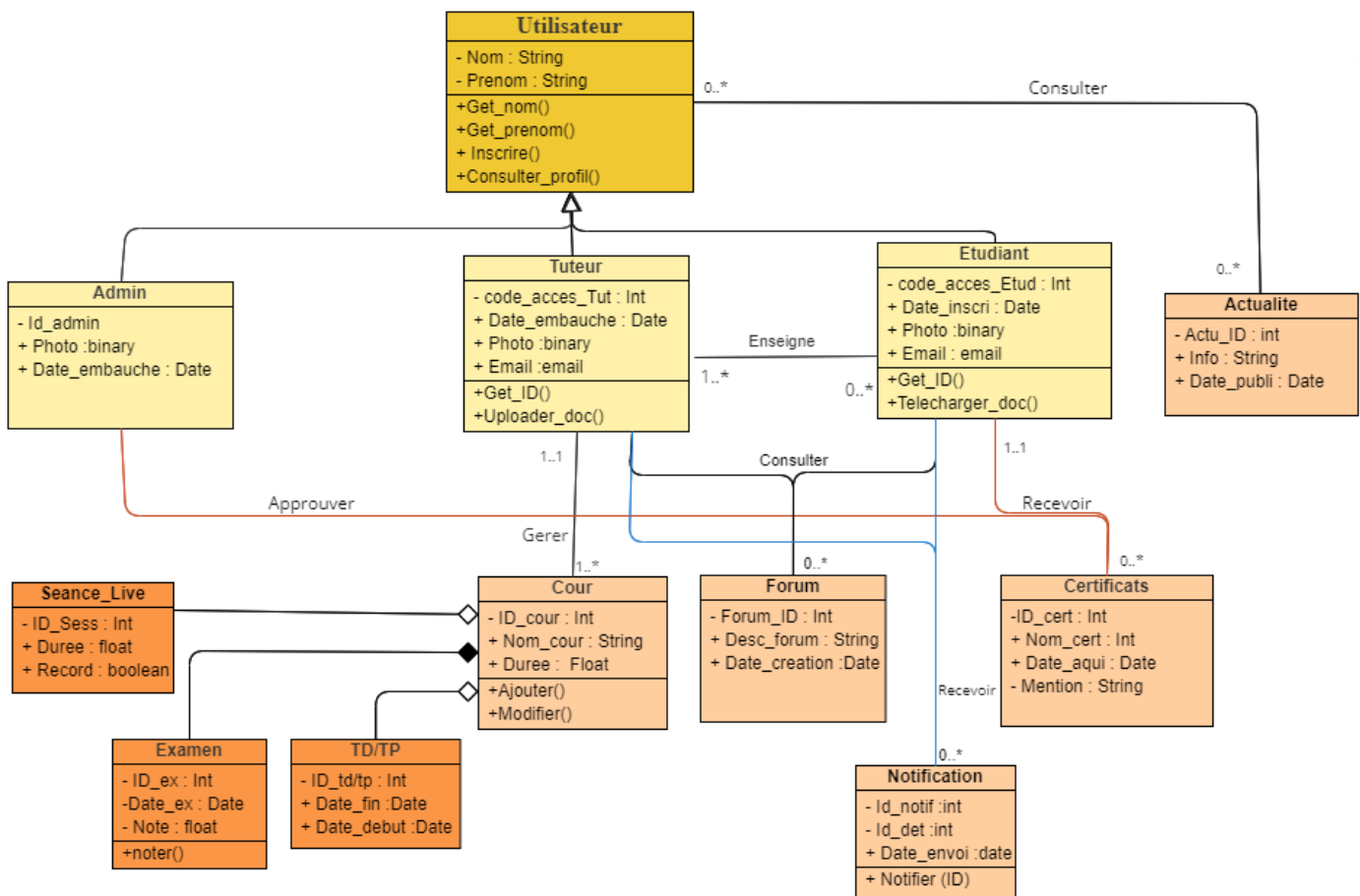


Figure 19 Diagramme de classes générale

3.1.2. Diagramme de classes de conception (MVC)

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.

Comme illustré dans la figure 20, nous présentons les classes de notre application selon les couches suivantes :

- **Vue** : cette couche comporte les composants « Components » qui correspondent aux interfaces graphiques via lesquelles l'utilisateur interagit avec le système et provoque des événements.
- **Controller** : cette couche comporte les contrôleurs qui permettent de recevoir tous les événements de l'utilisateur et enclenche les actions à effectuer.

- **Model** : cette couche représente les entités métiers gérés par notre application. Il y a une analogie entre les entités de cette couche et les relations/tableaux de la base de données.

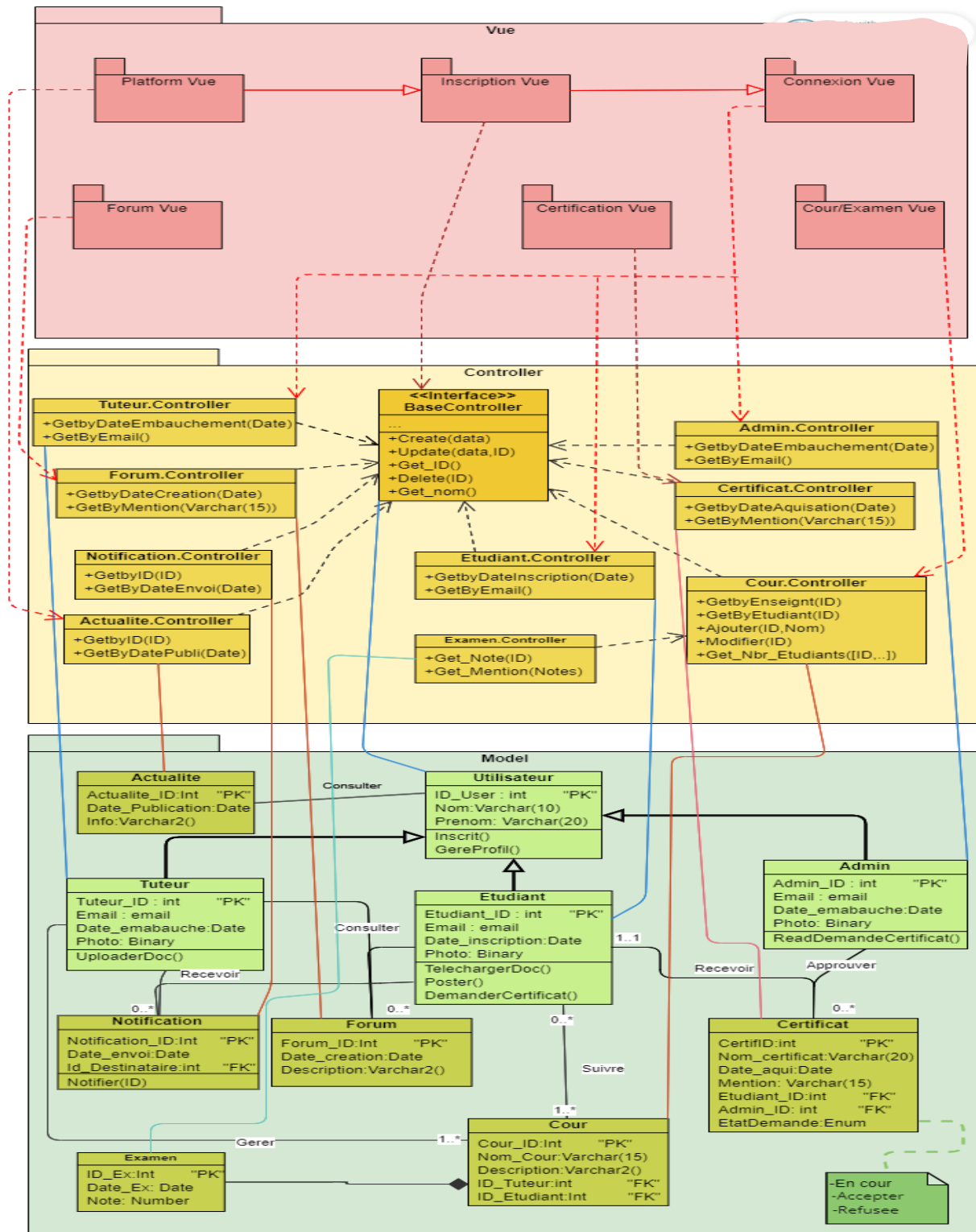


Figure 20 Diagramme de classes de conception (MVC)

3.2. Vue dynamique : Diagramme de séquence

Dans ce qui suit, nous présentons une vue générale des interactions entre les différentes couches logicielles de notre application. Comme illustré dans la figure 21, (le diagramme illustre tous les cas possibles de tous les actions d'acteur) le rôle d'un composant est de capter une requête envoyée par l'utilisateur et de provoquer la route pour requête vers le contrôleur approprié. Pour interagir avec la base de données le contrôleur sollicite la couche model.

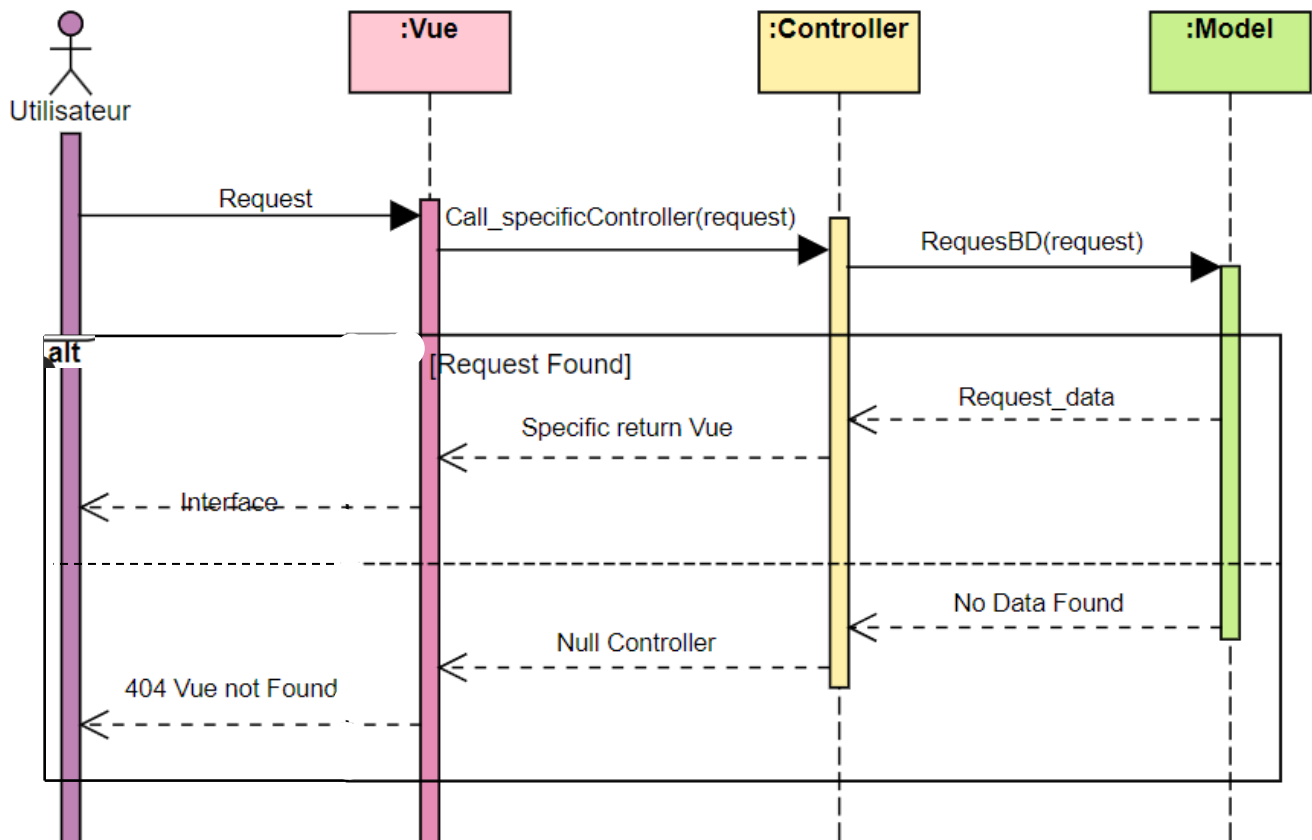


Figure 21 Diagramme de séquence générale


4. Conception graphique

La conception graphique est une étape indispensable, il permet de donner une idée sur le site et sur les différentes interfaces sous forme de schéma. Dans ce qui suit, nous allons illustrer quelques maquettes de notre application :

4.1. Interface du cas d'utilisation « Inscription » et « Connexion »

L'utilisateur est permis de créer un nouveau compte ou de connecter si il est déjà inscrit.

ooo



Creer un Compte

C'est déjà? Et alors Connectez-vous.

NAME

PRENOM


EMAIL

CODE D'ACCÈS

Inscription

Figure 22 Interface "Inscription"

ooo



Connexion

Connecter pour continuer.

EMAIL

CODE D'ACCÈS

Connecter

Figure 24 Interface "Connexion"

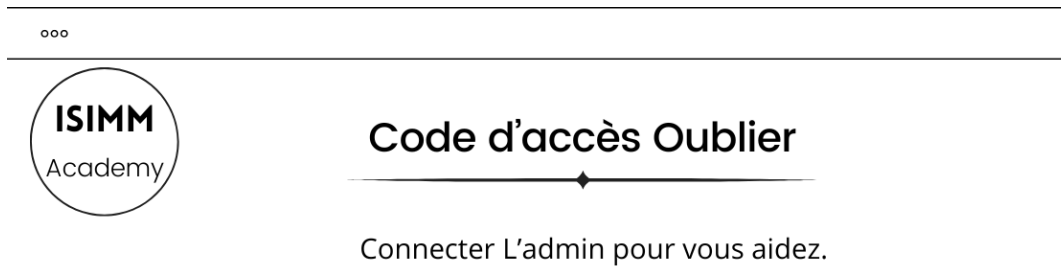


Figure 26 Interface "Cas de Code d'accès oubliée"

4.2. Interface du cas d'utilisation « Gérer Progrès du cours »

L'étudiant est permis de vérifier leurs progrès dans les cours inscrits et gérer soit télécharger ressources, contacter l'enseignant, créer ou commenter dans un forum spécial a cette cour, ...

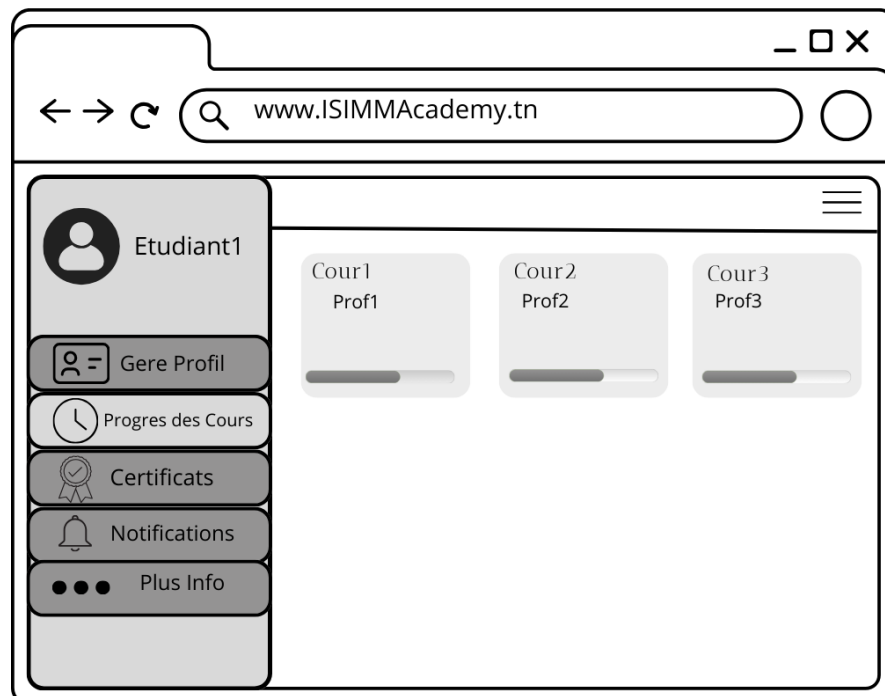


Figure 28 Interface de l'étudiant pour gérer les cours

Conclusion

Dans ce chapitre, nous avons détaillé les principales étapes de la conception de notre projet. En effet, nous avons commencé par présenter l'architecture générale de notre application. Ensuite, nous avons détaillé la conception de la base de données via le modèle conceptuel de données et le modèle relationnel. Puis nous avons passé à la conception logicielle en donnant le diagramme de classe qui détaille la vue statique et le diagramme de séquence qui détaillent la vue dynamique de notre système. Enfin nous avons présenté la conception graphique des quelques interfaces utilisateur.