

Mémoire de Projet de Fin d'Etudes

Présenté en vue de l'obtention du

Diplôme National de Licence en Science
d'Informatique

Author Name

Date

ABSTRACT

A brief summary of the project.

DÉDICATION

To my parents.

TABLE DES MATIÈRES

1	Étude Préliminaire et Contextualisation du Projet	1
1	Contexte général et cadre académique du projet	2
1.1	Cadre académique du projet	2
1.2	Présentation de l'entité d'accueil	2
1.3	Identification de la problématique	4
1.4	Idée générale du projet	4
1.5	Concepts de base liés à notre projet	5
2	Analyse de l'existant et perspectives critiques	6
2.1	Étude des solutions existantes	6
2	Fondements de l'Intelligence Artificielle et de l'Infrastructure Cloud dans notre Plateforme	9
1	L'intelligence artificielle dans notre plateforme	11
1.1	Définitions	11
1.2	Explication des concepts clés	12
1.3	Algorithme d'Apprentissage Adopté	14
2	Le Cloud dans notre plateforme	16
2.1	Présentation du cloud computing	17
2.2	Type de cloud	17
2.3	Avantages de l'adoption du cloud dans notre projet	18

TABLE DES MATIÈRES

2.4	Service cloud adoptés	18
3	Analyse et spécification des besoins	20
1	Identification des acteurs	21
2	Modèle informationnel de contexte	22
3	Recueil des besoins	23
3.1	Capture des besoins fonctionnels	23
3.2	Capture des besoins non fonctionnels	25
4	Spécification des besoins	26
4.1	Diagramme de cas d'utilisation global	26
4.2	Diagrammes de cas d'utilisation détaillés	27
4.3	Analyse des besoins	30
5	Capture des besoins techniques	43
5.1	Framework frontend	43
5.2	Framework backend	44
5.3	Base de données	45

TABLE DES FIGURES

1.1	Logo du « Globe Services Informatique (GSI) »	3
1.2	Logo de « Google Classroom »	6
1.3	Logo de « Piazza »	7
1.4	Logo de « ChatGPT »	7
1.5	Logo de « poe »	8
2.1	Le déroulement et les étapes de Dynamic RAG	14
2.2	Le déroulement de l'algorithme Sparse Mixture of Experts	15
3.1	Diagramme de contexte dynamique de notre plateforme	23
3.2	Diagramme de cas d'utilisation global	27
3.3	Diagramme de cas d'utilisation détaillé de l'acteur « Etudiant » . .	28
3.4	Diagramme de cas d'utilisation détaillé de l'acteur « Enseignant »	29
3.5	Diagramme de cas d'utilisation détaillé de l'acteur «Modèle MoE»	30
3.6	Diagramme de séquence « S'identifier »	34
3.7	Diagramme de séquence « Créer classroom »	38
3.8	Diagramme de séquence « Démarrer chat »	42
3.9	Les statistiques de la satisfaction des utilisateurs à l'usage des frameworks et bibliothèques dans le domaine du développement web	44

3.10 Les statistiques des options de base de données les plus populaires	46
--	----

LISTE DES TABLEAUX

2.1	Descriptions des services de cloud computing	19
3.1	Description textuelle du cas d'utilisation « s'identifier »	31
3.2	Description textuelle du cas d'utilisation « Gérer classroom » . . .	35
3.3	Description textuelle du cas d'utilisation « Démarrer chat »	39

Introduction

A long introduction.

CHAPITRE 1

ÉTUDE PRÉLIMINAIRE ET CONTEXTUALISATION DU PROJET

Contents

1	Contexte général et cadre académique du projet	2
1.1	Cadre académique du projet	2
1.2	Présentation de l'entité d'accueil	2
1.3	Identification de la problématique	4
1.4	Idée générale du projet	4
1.5	Concepts de base liés à notre projet	5
2	Analyse de l'existant et perspectives critiques	6
2.1	Étude des solutions existantes	6

Introduction

Ce chapitre vise à exposer l'étude préliminaire de notre projet. Tout d'abord, nous présenterons le contexte général et l'entité qui nous a accueilli pour faire un stage de fin d'études. Puis, nous aborderons la problématique et l'idée générale de notre projet. Ensuite, nous entreprendrons une analyse approfondie de l'existant, mettant en évidence les avantages et les limites des solutions similaires présentées sur le marché, afin de nous en inspirer et de retenir une solution plus raffinée. Enfin, nous conclurons ce chapitre en exposant la méthodologie de développement la plus adaptée à la réalisation de notre projet, et le diagramme de Gantt illustrant le planning général de celui-ci.

1 Contexte général et cadre académique du projet

Dans cette partie, nous représenterons le contexte de notre projet, qui inclut le cadre académique, la présentation de l'organisme d'accueil, la problématique et l'idée générale de notre projet.

1.1 Cadre académique du projet

Notre projet, intitulé « *Conception et développement d'une plateforme E-learning intégrant un système intelligent d'élucidation intelligente des ressources pédagogiques* », s'inscrit dans le cadre de la préparation d'un projet de fin d'études en vue de l'obtention du diplôme national de Licence en Sciences d'Informatique à « **l'Institut Supérieur d'Informatique et de Mathématiques de Monastir (ISIMM)** » pour l'année universitaire 2023/2024. Le stage a été effectué au sein de la société **Globe Services Informatique GSI** durant une période de 4 mois.

1.2 Présentation de l'entité d'accueil

L'idée initiale de notre projet est proposée par nous-même et elle est adoptée par **STE Globe Services Informatique (GSI)**, une entreprise établie à Houmet

Souk, Monastir, en Tunisie. Fondée le 20 mai 2000 en tant que société à responsabilité limitée (SARL), **GSI** se concentre principalement sur le commerce et la maintenance de matériel informatique, ainsi que sur le développement web.



FIGURE 1.1 – Logo du « Globe Services Informatique (GSI) »

Les secteurs d'activités de GSI sont :

- La vente et la maintenance de matériel informatique.
- Le développement web et mobile.
- La formation en informatique.
- La vente de logiciels.

Identité de l'organisation :

- **Nom** : Globe Services Informatique (GSI).
- **Adresse** : 5 Rue de la République, Houmet Souk, Monastir, Tunisie.
- **Téléphone** : +(216) 73 447 447.
- **Fax** : (+216) 73 447 447.
- **E-mail** : commercial@gsi.com.tn.
- **Site web** : www.gsi.com.tn.

1.3 Identification de la problématique

Dans le quotidien étudiantin et dans le déroulement ordinaire d'un cours, où l'enseignant anime la classe et les étudiants s'investissent pleinement dans l'acquisition et la compréhension des connaissances, des perturbations diverses sont fréquemment rencontrées. En effet, les étudiants se heurtent souvent à des obstacles dans la compréhension de leurs cours ce qui entrave leur progression académique.

De plus, les supports pédagogiques fournis par les enseignants tels que les cours, exercices, corrections et examens peuvent parfois se révéler insuffisants pour une assimilation complète. Par conséquent, de nombreux étudiants sont amenés à chercher d'autres ressources par eux-mêmes. Néanmoins, l'accès à ces ressources présente un autre défi incontournable. Cette situation limite considérablement leur capacité à acquérir efficacement des connaissances.

Exemple : Le coût élevé de l'accès à certains documents et ressources pédagogiques d'intérêt constitue un obstacle financier majeur qui prive les étudiants d'outils essentiels fondamentaux à leur épanouissement éducatif.

1.4 Idée générale du projet

À l'origine de notre projet, nous avons puisé notre inspiration dans notre expérience de la vie étudiantine. Ainsi, nous cherchons à relever les défis auxquels nos pairs sont confrontés, tels que la compréhension de leurs cours, l'accès aux ressources pédagogiques adaptées, et le besoin de soutien académique personnalisé.

Notre idée consiste à concevoir et mettre en œuvre une plateforme interactive qui révolutionne l'expérience d'apprentissage des étudiants et des enseignants en fournissant des contenus multimédias et des outils d'intelligence artificielle. De plus, cette plateforme favorise l'apprentissage personnalisé et collaboratif, tout en demeurant une solution riche en services pertinents, intelligents et open-source.

Nous visons à exploiter des technologies avancées de l'intelligence artificiel (IA) et

du cloud pour mettre en place une solution robuste.

Avant de nous plonger dans l'analyse et la critique des solutions existantes sur le marché, il convient de préciser que notre projet est basé sur une idée préliminaire. Toutefois, cette idée peut être affinée et améliorée en s'inspirant des meilleures pratiques et des fonctionnalités offertes par les plateformes similaires. Dans un premier temps, nous allons donc expliquer les concepts de base liés à notre domaine de travail, afin de poser les fondations nécessaires à la compréhension de notre projet.

1.5 Concepts de base liés à notre projet

Notre projet s'appuie sur deux piliers fondamentaux qui sont le cloud computing et l'intelligence artificielle (IA). Ces concepts jouent un rôle central de notre ambition de révolutionner l'expérience d'apprentissage des étudiants et des enseignants, en offrant des solutions innovantes et performantes.

- **Cloud computing** : Le cloud computing révolutionne la manière dont les services informatiques sont fournis et consommés. En exploitant le cloud, notre projet peut tirer parti d'une infrastructure évolutive et flexible, permettant un accès rapide et sécurisé aux ressources informatiques. Cette approche offre également une réduction des coûts opérationnels et une amélioration de l'efficacité grâce à la mise à l'échelle automatique des ressources en fonction des besoins.
- **Intelligence artificielle (IA)** : L'intelligence artificielle constitue le cœur de notre projet, offrant des fonctionnalités avancées telles que l'analyse de documents, la recommandation de contenu personnalisé et l'assistance virtuelle. Grâce à l'IA, notre plateforme peut comprendre, interpréter et répondre aux besoins des utilisateurs de manière intelligente, offrant ainsi une expérience d'apprentissage plus personnalisée et efficace.

En combinant les capacités du cloud computing et de l'IA, notre projet vise à fournir une solution innovante et intelligente pour répondre aux défis de l'éducation contemporaine.

2 Analyse de l'existant et perspectives critiques

L'étude de l'existant est une étape primordiale qui permet de définir les points forts et les points faibles des systèmes similaires actuellement en place. Alors, cette section sera dédiée à faire une étude approfondie et critique des solutions existantes.

2.1 Étude des solutions existantes

Dans le domaine des technologies éducatives, plusieurs plateformes se démarquent par leur contribution à l'apprentissage. Nous étudierons ces solutions pour comprendre leurs forces. Nous identifierons également les opportunités d'amélioration que notre solution pourrait exploiter.

Étant donné qu'il n'existe pas de solutions tunisiennes similaires à notre projet, notre revue se concentre sur le marché international.

Ainsi, nous avons porté notre attention sur les solutions étrangères les plus connues qui s'alignent avec le contexte de notre projet. Notre étude se focalise, particulièrement sur *Google Classroom*, *Poe*, *ChatGPT* et *Piazza*, les quatre solutions les plus populaires et adaptées dans le monde entier.

- **Google Classroom** est une plateforme éducative qui permet aux enseignants de créer des salles de classe virtuelles pour leurs étudiants, partager des documents, des devoirs et communiquer avec les apprenants. Le logo de Google Classroom est illustré à la figure 1.2, permet de :



FIGURE 1.2 – Logo de « Google Classroom »

- Gérer les cours ainsi que les ressources pédagogiques pour les étudiants.

- Faire des discussions dans les classes.
- **Piazza** est une plateforme de collaboration en ligne conçue pour faciliter la communication entre les étudiants et les professeurs. Le logo de cette plateforme est affiché dans la figure 1.3, permet de :



FIGURE 1.3 – Logo de « Piazza »

- Stocker les documents.
- Organiser des discussions entre les étudiants et les enseignants.
- **ChatGPT** ou Chat Generative Pretrained Transformer, est un chatbot doté d'intelligence artificielle qui fournit des réponses textuelles instantanées aux questions des utilisateurs. Il se positionne comme un outil polyvalent répondant aux besoins diverse. Le logo de ChatGPT est illustré à la figure 1.4, permet de :



FIGURE 1.4 – Logo de « ChatGPT »

- Répondre aux questions et aux requêtes en se basant sur le contexte de la conversation.
- Fournir des informations sur divers sujets.
- **Poe** est un chatbot qui fonctionne à partir de textes, offrant des interactions conversationnelles pour répondre aux questions des utilisateurs et fournir une assistance. Le logo de Poe est illustré à la figure 1.5, permet de :
 - Résoudre des problèmes en décrivant votre contexte.
 - Obtenir des réponses personnalisées à vos besoins.

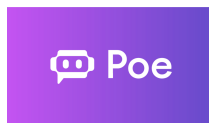


FIGURE 1.5 – Logo de « poe »

CHAPITRE 2

FONDEMENTS DE L'INTELLIGENCE ARTIFICIELLE ET DE L'INFRASTRUCTURE CLOUD DANS NOTRE PLATEFORME

Contents

1	Identification des acteurs	21
2	Modèle informationnel de contexte	22
3	Recueil des besoins	23
3.1	Capture des besoins fonctionnels	23
3.2	Capture des besoins non fonctionnels	25
4	Spécification des besoins	26
4.1	Diagramme de cas d'utilisation global	26
4.2	Diagrammes de cas d'utilisation détaillés	27
4.3	Analyse des besoins	30
5	Capture des besoins techniques	43
5.1	Framework frontend	43
5.2	Framework backend	44

Chapitre 2. Fondements de l’Intelligence Artificielle et de l’Infrastructure Cloud dans notre Plateforme

5.3	Base de données	45
-----	---------------------------	----

Introduction

Ce chapitre explore les fondements de l'Intelligence Artificielle (IA) et de l'infrastructure cloud dans notre plateforme. Nous aborderons des définitions et explications des concepts clés de l'AI et de l'apprentissage profond, notre choix de l'algorithme MoE et l'intégration du cloud. Cette compréhension nous permettra de saisir les capacités de notre plateforme.

1 L'intelligence artificielle dans notre plateforme

Cette section sera consacrée à la présentation des concepts de base sur l'intelligence artificielle.

1.1 Définitions

Dans cette partie, nous allons définir l'intelligence artificielle et l'apprentissage profond.

1.1.1 Intelligence artificielle

L'intelligence artificielle concerne la création de machines capables de penser et d'agir comme des êtres humains [X]. En d'autres termes, c'est la science qui vise à créer des programmes informatiques et des machines qui peuvent imiter le raisonnement humain, apprendre de l'expérience et accomplir des tâches variées de manière autonome.

1.1.2 Apprentissage profond

L'apprentissage profond, connu sous le nom de deep learning, est une branche de l'IA qui se concentre sur l'entraînement de réseaux de neurones artificiels [x]. Ces réseaux de neurones sont organisés en couches. Chaque couche transforme l'entrée qu'elle reçoit pour produire une sortie. Ces transformations sont ajustées

automatiquement pendant l'apprentissage pour améliorer les performances du système.

1.2 Explication des concepts clés

Dans cette section, nous allons clarifier quelques concepts fondamentaux.

1.2.1 Natural Language Model (NLP)

Les modèles de langage naturel (NLP) sont des composants de l'intelligence artificielle conçus pour comprendre et générer un langage humain naturel. Ils sont largement utilisés dans des applications telles que la traduction automatique, la génération de texte et l'analyse du sentiment. Ces modèles sont entraînés sur de grandes quantités de données textuelles afin d'apprendre les structures linguistiques et de capturer les nuances du langage.

1.2.2 Large Model Language (LLM)

Les modèles de langage de grande taille (LLM) sont des modèles d'intelligence artificielle qui ont été entraînés sur de vastes ensembles de données textuelles pour acquérir une compréhension approfondie du langage naturel. Ces modèles sont capables de générer du texte cohérent et de qualité et sont souvent utilisés pour une variété de tâches en NLP.

1.2.3 Tokenizers

Les tokenizers sont des outils essentiels en traitement automatique du NLP qui découpent le texte en unités plus petites appelées "tokens". Ces tokens peuvent être des mots, des sous-mots ou même des caractères individuels. Ils servent de points de départ pour l'analyse et le traitement du texte.

1.2.4 Retrieval Augmented Generation (RAG)

RAG est un concept utilisé dans le domaine de l'intelligence artificielle et le traitement du langage naturel. Il utilise des techniques de récupération d'informa-

tions pour améliorer la génération de texte par des modèles de langage.

En d'autres termes, au lieu de simplement générer du texte en fonction de ce que le modèle a appris pendant son entraînement, un système RAG va d'abord chercher dans une grande base de données de textes pour trouver des informations pertinentes. Il utilisera ensuite ces informations pour générer une réponse plus informée et précise.

Notons que cette base de données de textes peut être vectorielle. C'est-à-dire que les textes sont représentés sous forme de vecteurs dans un espace vectoriel. Ceci permet de les comparer et de les rechercher de manière efficace.

1.2.5 Dynamic Retrieval Augmented Generation (Dynamic RAG)

Dynamic RAG (Retrieval Augmented Generation) est une approche avancée de RAG qui permet d'adapter dynamiquement la récupération d'informations en fonction du contexte de la requête de l'utilisateur.

Dans un système RAG standard, la stratégie de récupération d'informations est généralement fixe et déterminée à l'avance. Par exemple, le système doit toujours chercher dans la même base de données de textes.

Alors que dans un système Dynamic RAG, le système peut adapter sa stratégie de récupération d'informations en fonction de la tâche à accomplir. Par exemple, pour une tâche de réponse à des questions, le système peut chercher dans une base de données de textes spécifique.

Cette approche permet au système d'être plus flexible et adaptable, ce qui lui permet d'améliorer ses performances sur une variété de tâches.

La Figure 2.1. synthétise les étapes et le déroulement de Dynamic RAG.

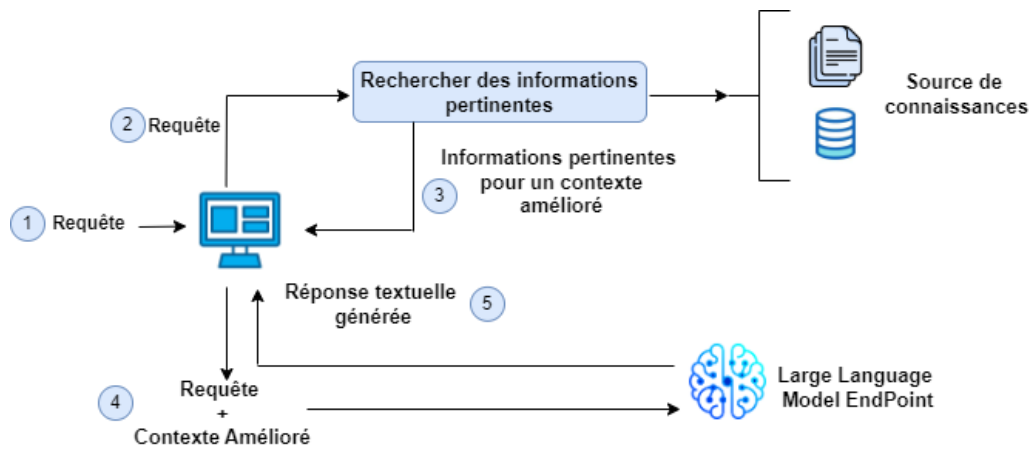


FIGURE 2.1 – Le déroulement et les étapes de Dynamic RAG

1.3 Algorithme d'Apprentissage Adopté

Dans cette section, nous expliquerons le Sparse Mixture of Experts (MoE), l'algorithme d'apprentissage adopté. Nous aborderons l'introduction à cet algorithme, la définition de la couche Sparse MoE et son architecture ainsi que le processus de formation et d'inférence associé.

Notons que les informations de cette section sont issues de la référence suivante.[x]

1.3.1 Couche Sparse MoE et son architecture

La couche Sparse MoE est une composante clé des modèles de transformers remplaçant les couches traditionnelles de réseaux de neurones denses. Chaque couche Sparse MoE est constituée de multiples "experts" qui sont essentiellement des réseaux de neurones. Ces experts sont souvent des réseaux feed-forward (FFN) et peuvent aussi être plus complexes ou former un autre MoE permettant ainsi des MoEs hiérarchiques.

En plus des experts, une couche Sparse MoE inclut un réseau de portes ou "routeur". Ce routeur détermine comment les tokens sont distribués aux experts. Il est important de noter qu'un même token peut être attribué à plusieurs experts.

Le routeur est constitué de paramètres appris et entraîné avec le reste du réseau.

La Figure 2.2. synthétise l'architecture de l'algorithme Sparse Mixture of Experts.

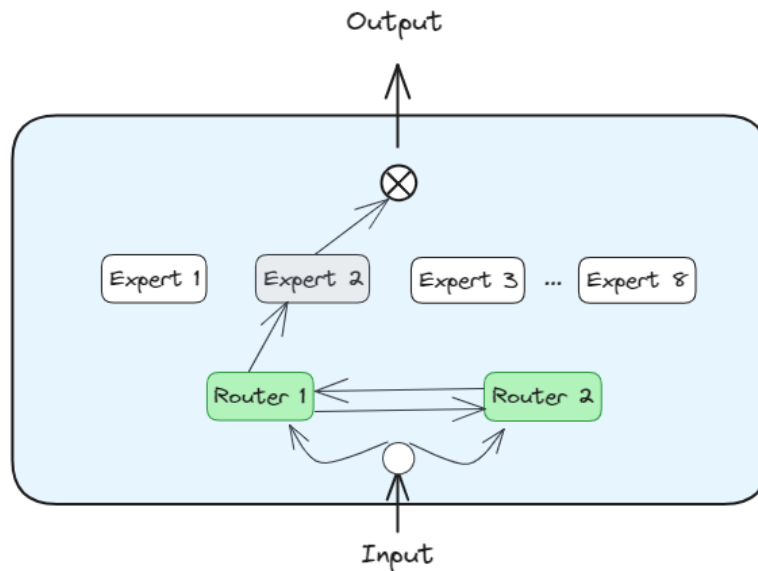


FIGURE 2.2 – Le déroulement de l’algorithme Sparse Mixture of Experts

1.3.2 Fonctionnement de l’algorithme Sparse Mixture of Experts

L’algorithme sparse Mixture of Experts est une méthode avancée qui combine les avantages des modèles experts spécialisés avec la flexibilité des modèles généralistes.

Contrairement aux réseaux neuronaux traditionnels qui utilisent une seule architecture de modèle pour traiter toutes les entrées, le MoE divise le processus d’apprentissage en plusieurs experts spécialisés. Ces experts sont ensuite combinés de manière pondérée pour générer une prédiction finale (voir Figure 2.2). Chaque expert est responsable de traiter une portion spécifique de l’entrée en apportant ainsi sa propre expertise à la résolution du problème. Ces experts peuvent être considérés comme des sous-modèles qui se spécialisent dans différentes parties de l’espace d’entrée.

L’approche sparse du MoE signifie que seuls quelques experts sont activés pour chaque exemple d’entrée. Elle permet une utilisation efficace des ressources computationnelles et une adaptation dynamique aux différentes conditions d’entrée.

Cette activation sélective des experts est déterminée par des routeurs qui dirigent l'entrée vers les experts les plus pertinents en fonction de son contenu.

1.3.3 Processus de formation et d'inférence

Le processus d'entraînement de Mixture of Experts (MoE) implique simultanément l'apprentissage des paramètres des experts et du routeur.

Pendant la formation, chaque jeton d'entrée est dirigé vers un ou plusieurs experts en fonction des poids appris par le routeur. Les experts traitent ces jetons et retournent leurs sorties respectives. Ces sorties sont combinées pour former la sortie finale du MoE utilisée pour calculer la perte et mettre à jour les paramètres du modèle. En phase d'inférence, seuls les experts les plus pertinents sont sélectionnés selon les poids du routeur en réduisant ainsi la complexité de calcul et améliorant l'efficacité du MoE. Bien que tous les paramètres doivent être chargés en mémoire, seuls ceux nécessaires sont utilisés pendant l'inférence.

1.3.4 Raisons du Choix de l'Algorithme Sparse Mixture of Experts

Le choix d'adopter l'algorithme Sparse Mixture of Experts (MoE) repose sur sa capacité à optimiser l'utilisation des ressources computationnelles en ne faisant fonctionner que les parties pertinentes du modèle. Cela permet d'augmenter l'efficacité de l'apprentissage et réduire la charge de calcul qui est essentiel pour des applications à grande échelle nécessitant des modèles complexes.

En intégrant le Sparse MoE dans notre plateforme, nous sommes en mesure de fournir des réponses plus précises et adaptées aux besoins spécifiques de chaque utilisateur en améliorant ainsi l'expérience globale d'apprentissage.

2 Le Cloud dans notre plateforme

Dans cette partie, nous allons présenter le cloud computing, les types de cloud ainsi que le type de cloud adopté dans notre plateforme.

2.1 Présentation du cloud computing

Le cloud computing est une technologie qui permet d'accéder à des ressources informatiques en utilisant l'internet.

L'infrastructure cloud joue un rôle crucial dans le bon fonctionnement et l'évolutivité de notre application. Elle nous permet de bénéficier de services performants et sécurisés en optimisant nos coûts d'exploitation.

2.2 Type de cloud

Dans cette section, nous présenterons les types de cloud, suivi par le type adopté.

2.2.1 Types d'infrastructures de cloud computing

Le cloud computing est divisé en trois types principaux : public, privé et hybride.

- **Cloud privé**

Un cloud privé est un environnement de cloud computing dédié à une seule organisation. Dans un cloud privé, toutes les ressources sont isolées et sous le contrôle d'une seule organisation. Ainsi, le cloud privé est également appelé cloud interne ou d'entreprise. [x]

- **Cloud publique**

Un cloud public est une infrastructure informatique dans laquelle un fournisseur de services met des ressources à la disposition du public via internet. Les ressources varient selon le fournisseur mais peuvent inclure des capacités de stockage, des applications ou des machines virtuelles. [x]

- **Cloud hybride**

Un cloud hybride est un environnement informatique mixte dans lequel des applications s'exécutent à l'aide d'une combinaison de ressources de calcul, de stockage et de services dans différents environnements (clouds publics et

clouds privés, y compris des centres de données sur site ou en périphérie).
[x]

2.2.2 Type de cloud adopté

Notre application repose sur une architecture Cloud hybride qui combine à la fois des services de Cloud public et de Cloud privé. Cette approche nous permet de tirer parti des avantages de chaque modèle en fonction des besoins spécifiques de notre application.

2.3 Avantages de l'adoption du cloud dans notre projet

L'infrastructure Cloud offre de nombreux avantages pour notre projet. Tout d'abord, elle nous permet de bénéficier d'une grande scalabilité en adaptant automatiquement les ressources allouées en fonction des besoins de notre application. De plus, elle garantit une haute disponibilité et une redondance des données grâce à la répartition géographique des serveurs. Enfin, l'infrastructure Cloud nous permet de réduire nos coûts d'exploitation.

2.4 Service cloud adoptés

Parmi les services Cloud intégrés dans notre plateforme, nous retrouvons le Runtime Edge, le Serverless et le CDN (Content Delivery Network). Les services cloud utilisés dans notre plateforme sont présentés dans le tableau 2.1.

TABLE 2.1 – Descriptions des services de cloud computing

Service	Description
Runtime Edge	Edge Runtime est idéal pour la diffusion de contenu dynamique et personnalisé avec une faible latence en utilisant de petites fonctions simples. Sa rapidité provient de l'utilisation minimale des ressources, mais cela peut être limité dans de nombreux scénarios. Dans notre application, Edge Runtime est utilisé pour optimiser la diffusion de contenu personnalisé pour garantir ainsi une expérience utilisateur rapide et réactive.
Serverless	Les architectures sans serveur, ou Serverless, permettent de déléguer la gestion des serveurs à un fournisseur de services Cloud. Cette approche offre plusieurs avantages tels qu'une réduction des coûts d'exploitation, une scalabilité automatique et une simplification de la maintenance. Dans notre application, le Serverless est utilisé pour gérer les fonctions backend telles que le traitement des données et la gestion des Application Programming Interface (API).
Content Delivery Network	Un Content Delivery Network (CDN) est un réseau de serveurs distribués géographiquement qui permet de distribuer du contenu à grande échelle. L'utilisation d'un CDN permet d'améliorer les performances de notre application en réduisant la latence et en optimisant la bande passante. Dans notre cas, le CDN est utilisé pour distribuer les ressources statiques de notre application telles que les images, les feuilles de style et les scripts.

Conclusion

Notre application exploite des techniques d'apprentissage profond et une infrastructure cloud avancée pour offrir une expérience utilisateur optimale et répondre aux exigences de performance et de fiabilité. Le chapitre suivant approfondira les exigences fonctionnelles et non fonctionnelles pour concevoir notre solution adaptée.

CHAPITRE 3

ANALYSE ET SPÉCIFICATION DES BESOINS

Contents

1	Identification des acteurs	21
2	Modèle informationnel de contexte	22
3	Recueil des besoins	23
3.1	Capture des besoins fonctionnels	23
3.2	Capture des besoins non fonctionnels	25
4	Spécification des besoins	26
4.1	Diagramme de cas d'utilisation global	26
4.2	Diagrammes de cas d'utilisation détaillés	27

Introduction

L'étape d'analyse et spécification des besoins est une étape indispensable pour comprendre les fonctionnalités que le système doit fournir. Ce chapitre sera consacré à détailler cette étape : Nous commencerons par l'identification des acteurs. Puis, nous élaborons le diagramme de contexte de notre système. Ensuite, nous allons faire un recueil sur les besoins fonctionnels et non fonctionnels de notre projet. Par la suite, nous allons spécifier et analyser les besoins identifiés en se basant sur le modèle de cas d'utilisation et le diagramme de séquence d'analyse du langage **Unified Modeling Language (UML)**. Enfin, un recueil sur les besoins techniques sera présenté tout en citant les frameworks choisis.

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. [x]

Notre système comporte les acteurs suivants :

- **Etudiant** : L'étudiant, en tant qu'utilisateur inscrit, est au cœur de notre système qui peut interagir avec notre chatbot Mixture of Experts et bénéficier de ses services intelligents. Ainsi, il peut accéder aux salons virtuels (Classrooms) et leurs contenus auxquels il est affilié c-à-d les ressources pédagogiques de ses enseignants. Ensuite, cet utilisateur peut partager des supports pédagogiques (de ses enseignants ou provenant de sources externes) dans un espace de chat instantané, sollicitant ainsi d'une gamme de services intelligents tels que l'explication, la traduction, la correction d'exercices, la génération de résumés, etc. Ces fonctionnalités sont conçues pour clarifier ses cours, assurant une assistance personnalisée tout au long de son apprentissage.
- **Enseignant** : L'acteur enseignant est un utilisateur qui admet un compte et joue un rôle actif dans l'expérience d'apprentissage en ligne en facilitant le partage de connaissances et en favorisant un environnement collaboratif

au sein de notre plateforme. Ainsi, il bénéficie d'outils conviviaux lui permettant de créer et de gérer des classrooms dédiés à ses cours et y partage ses ressources pédagogiques (cours, fascicule, corrections des fascicules, etc.). Il peut organiser les ressources pédagogiques de manière structurée, favorisant ainsi un accès facile et une interaction fluide avec les étudiants.

- **Modèle MoE (Modèle Mixture of Experts)** : Le modèle Mixture Of Experts est un composant système qui permet de répondre aux questions et requêtes des étudiants en se basant sur les ressources pédagogiques publiées sur la plateforme et ainsi de clarifier leurs lacunes. En analysant les matériaux pédagogiques, le modèle fournit des explications détaillées et des exemples pertinents pour clarifier les concepts difficiles offrant un soutien individualisé aux étudiants et améliorant leur compréhension du contenu éducatif.

2 Modèle informationnel de contexte

Le modèle informationnel de contexte donne un aperçu global de notre plateforme en identifiant les acteurs et leurs interactions avec le système. À travers le diagramme de contexte, nous illustrons ces échanges pour une meilleure compréhension. Cette représentation simplifiée du fonctionnement de notre application aide à saisir les flux d'informations et le rôle des différents intervenants.

La Figure 3.1 montre le diagramme de contexte et illustre bien le modèle informationnel de l'application.

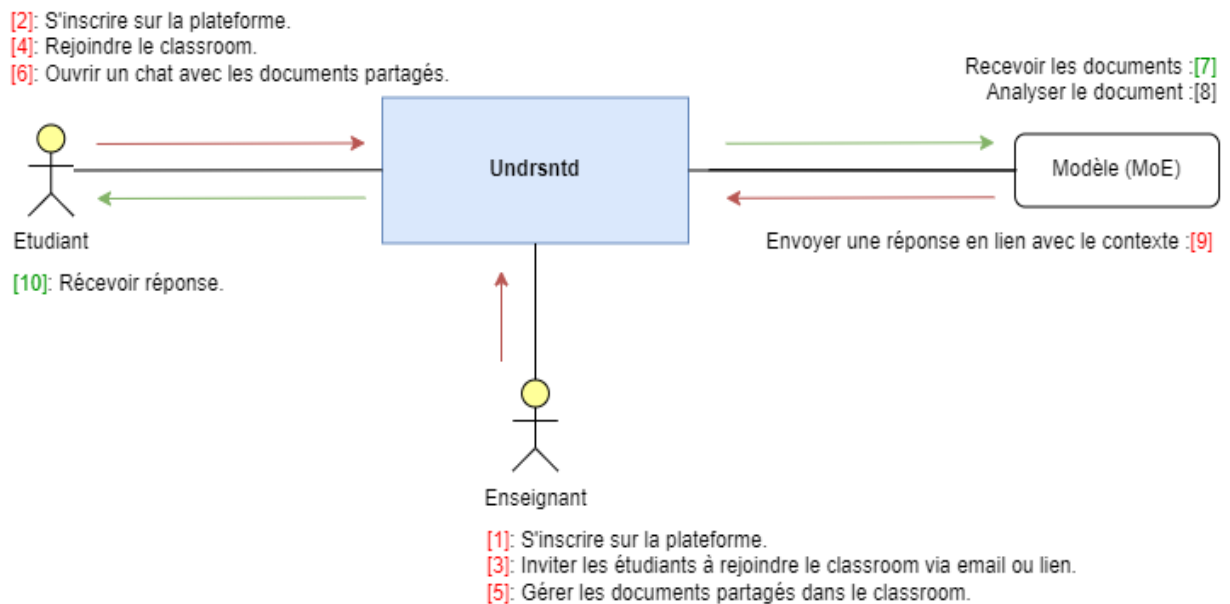


FIGURE 3.1 – Diagramme de contexte dynamique de notre plateforme

3 Recueil des besoins

Dans cette section, nous allons présenter les besoins fonctionnels et non fonctionnels de notre plateforme.

3.1 Capture des besoins fonctionnels

Les besoins fonctionnels définissent les actions que notre système doit accomplir pour répondre aux attentes des utilisateurs. Nous identifions ces besoins en fonction des différents acteurs impliqués. Ce capture nous permet de délimiter le périmètre fonctionnel de notre application et de garantir sa conformité aux exigences des utilisateurs.

- **Besoins fonctionnels de l'étudiant :**

- **S'inscrire :** Permettre à l'étudiant de créer un compte sur la plateforme en fournissant des informations nécessaires.

- **S'identifier** : Permettre à l'étudiant de se connecter à son compte pour accéder aux fonctionnalités de la plateforme.
- **Accéder au classroom** : Permettre à l'étudiant de rejoindre et quitter les classrooms, d'ouvrir ou télécharger les documents partagés, de démarrer un chat avec le modèle dans le contexte des documents partagés et d'interagir avec les publications.
- **Démarrer chat** : Permettre à l'étudiant de créer un espace de communication instantanée avec notre chatbot. Dans cette espace, l'étudiant peut non seulement démarrer une nouvelle conversation mais aussi consulter son historique et poursuivre la communication dans une conversation existante. Cette fonctionnalité lui permet d'envoyer une requête en se référant à un document spécifique ou sans faisant référence. Le premier cas est préconditionné par l'importation d'un document dans la conversation courante.
- **Gérer propre média** : Permettre à l'étudiant de consulter ses documents partagés ainsi que de les supprimer de la base de données.
- **Consulter historique** : Permettre à l'étudiant de consulter l'historique des classrooms, des commentaires ou des postes.
- **Gérer profil** : Permettre à l'étudiant de créer un compte sur la plateforme en fournissant des informations nécessaires.
- **Besoins fonctionnels de l'enseignant** :
 - **S'inscrire** : Permettre à l'enseignant de créer un compte sur la plateforme en fournissant des informations nécessaires.
 - **S'identifier** : Permettre à l'enseignant de se connecter à son compte pour accéder aux fonctionnalités de la plateforme.
 - **Gérer classroom** : Permettre à l'enseignant de créer, modifier, archiver, désarchiver et supprimer ses classrooms.
 - **Accéder au classroom** : Permettre à l'enseignant de gérer des publications en créant, modifiant et supprimant des publications, d'ouvrir ou télécharger des documents partagés, d'inviter des étudiants au class-

room et d'interagir sur les publications.

- **Consulter historique** : Permettre à l'enseignant de consulter l'historique des classrooms, des commentaires ou des postes.
 - **Gérer profil** : Permettre à l'enseignant de mettre à jour et de gérer les informations de son profil utilisateur telles que ses informations personnelles et ses préférences ainsi que de supprimer son profil.
- **Besoins fonctionnels du modèle MoE** :
 - **Répondre à une requête** : Permettre au modèle (MoE) de générer des réponses pertinentes en se référant à un document importé par l'étudiant ou à des ressources externes. Le modèle (MoE) est capable de traiter divers types de documents tels que des textes, des articles, des livres et même des codes sources. En fonction de cette analyse approfondie, il peut non seulement répondre aux questions de l'étudiant, mais, aussi, accomplir une multitude de tâches supplémentaires. Ainsi, il peut générer des résumés condensés des documents pour faciliter la compréhension, expliquer des concepts difficiles et donner des exemples concrets, corriger les exercices en fournissant des explications détaillées et bien plus encore. . .

3.2 Capture des besoins non fonctionnels

Les besoins non fonctionnels sont des critères de qualité qui décrivent les attentes non liées, directement, aux comportements fonctionnels.

Les besoins non fonctionnels de notre système sont :

- **La sécurité** : L'application doit garantir à l'utilisateur connecté l'intégrité et la confidentialité de ses données.
- **La convivialité** : La conception de l'interface utilisateur doit être simple et intuitive permettant aux utilisateurs de naviguer facilement dans le système et d'accomplir leurs tâches de manière efficace et agréable.

- **La performance** : Le système doit être en mesure de fournir des performances rapides et efficaces.
- **Fiabilité** : L'application doit fonctionner de façon cohérente pour les utilisateurs.
- **Rapidité** : Le système doit traiter les requêtes en temps réel et fournir des réponses instantanées aux étudiants pour répondre à leurs besoins.

4 Spécification des besoins

Pour une meilleure documentation de notre plateforme, nous allons formaliser, dans cette section, les fonctionnalités offertes par notre application en utilisant le diagramme de cas d'utilisation de l'UML.

4.1 Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation est le mécanisme le plus populaire pour la spécification des exigences. Il illustre les interactions entre les acteurs et le système. En outre, il donne une vue d'ensemble des fonctionnalités que le système doit offrir pour répondre aux besoins des utilisateurs.

La Figure 3.2 illustre le diagramme de cas d'utilisation global de notre application.

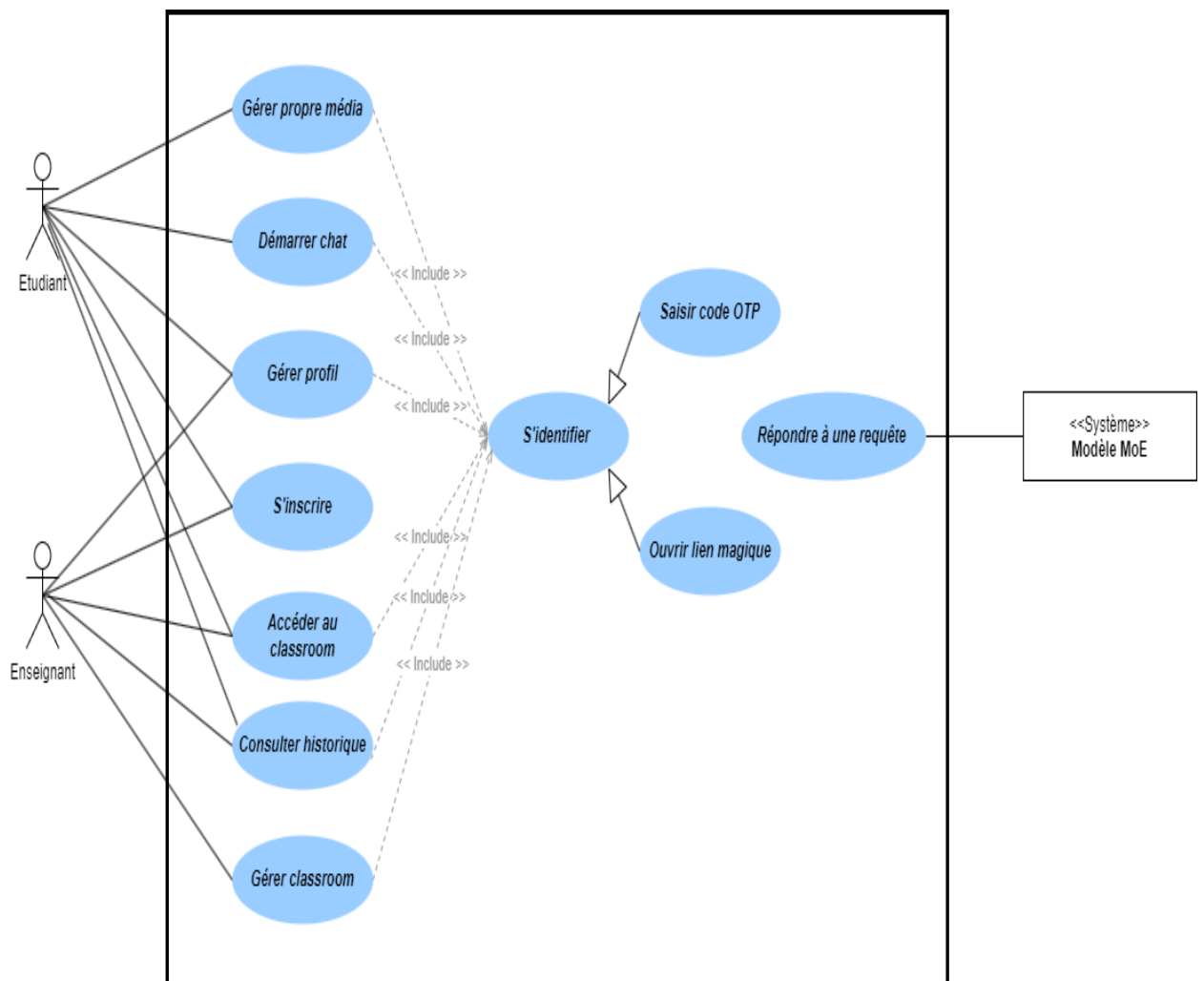


FIGURE 3.2 – Diagramme de cas d'utilisation global

4.2 Diagrammes de cas d'utilisation détaillés

Dans cette partie, nous allons raffiner les cas d'utilisation présentés dans la Figure 3.2. Le raffinement sera représenté par un diagramme de cas d'utilisation détaillé de chaque acteur.

4.2.1 Diagramme de cas d'utilisation détaillé de l'acteur « Etudiant »

La Figure 3.3 illustre le diagramme de cas d'utilisation détaillé de l'acteur étudiant après avoir s'identifier.



FIGURE 3.3 – Diagramme de cas d'utilisation détaillé de l'acteur « Etudiant »

4.2.2 Diagramme de cas d'utilisation détaillé de l'acteur « Enseignant »

La Figure 3.4 illustre le diagramme de cas d'utilisation détaillé de l'acteur enseignant après avoir s'identifier.

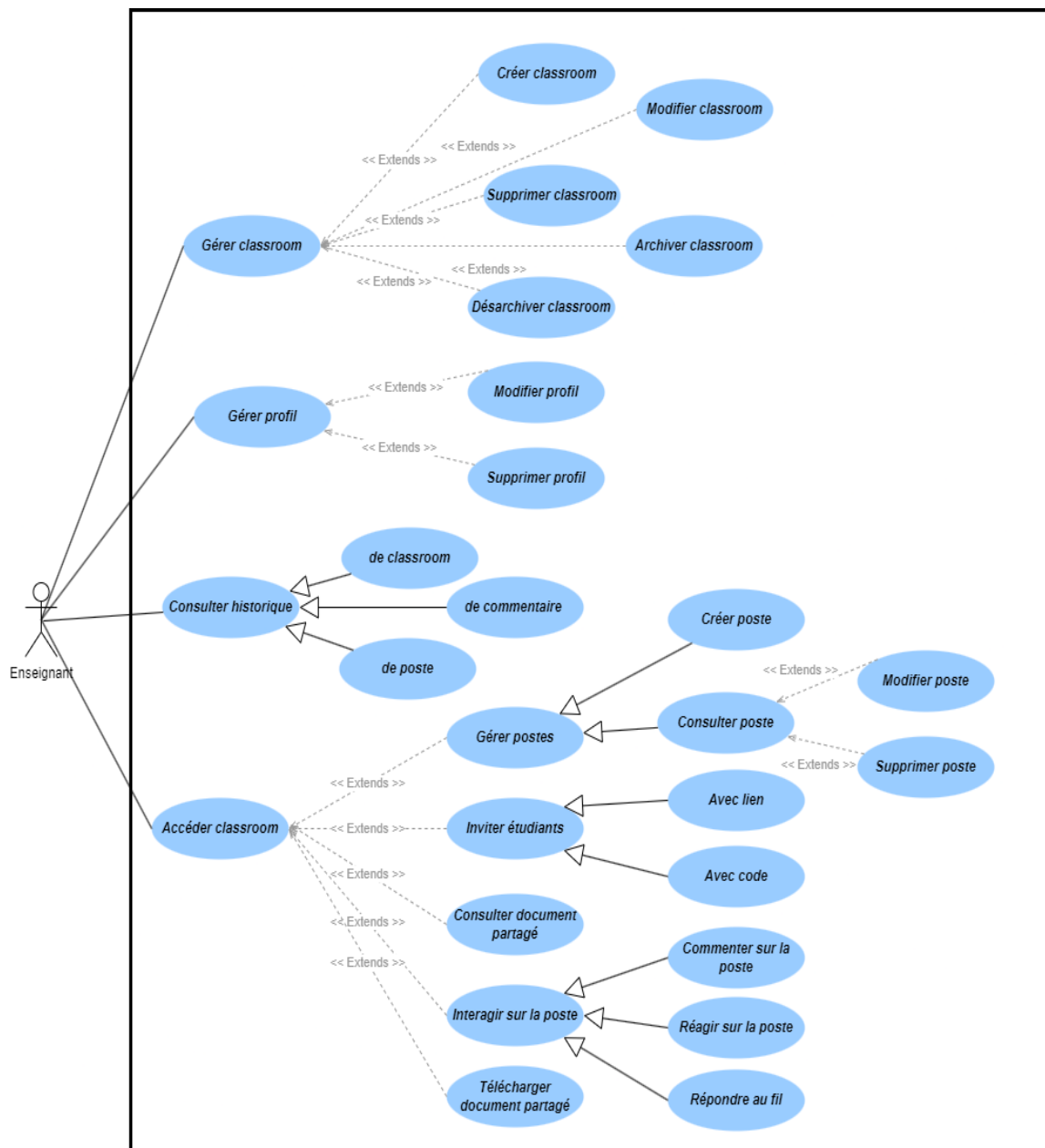


FIGURE 3.4 – Diagramme de cas d'utilisation détaillé de l'acteur « Enseignant »

4.2.3 Diagramme de cas d'utilisation détaillé de l'acteur « Modèle MoE »

La Figure 3.5 illustre le diagramme de cas d'utilisation détaillé de l'acteur modèle MoE.

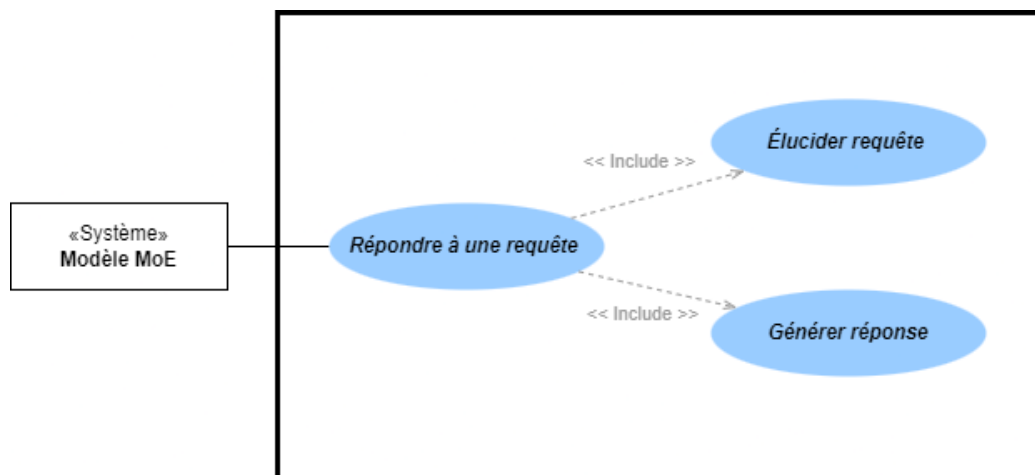


FIGURE 3.5 – Diagramme de cas d'utilisation détaillé de l'acteur «Modèle MoE»

4.3 Analyse des besoins

L'analyse des besoins est une étape importante permettant de représenter le comportement du système au fil du temps. Dans le contexte de la modélisation orienté objet l'analyse des besoins peut se faire par les descriptions textuelles et les diagrammes de séquence d'analyse. Une description textuelle est description des scénarios nominaux, d'extension et d'exception d'un cas d'utilisation. Un diagramme de séquence d'analyse, dite aussi acteur-système, est une représentation graphique utilisée pour montrer l'interaction entre les acteurs et le système où le système est représenté sous d'une boîte noire. Dans cette section, nous présenterons les descriptions textuelles et les diagrammes de séquence d'analyse relatifs aux principaux cas d'utilisation de notre plateforme.

4.3.1 Cas d'utilisation « S'identifier »

- **Description textuelle**

Le tableau 3.1 montre la description textuelle du cas d'utilisation « S'iden-

tifier ».

TABLE 3.1 – Description textuelle du cas d'utilisation « s'identifier »

Titre de cas d'utilisation	S'identifier
Acteurs principaux	Utilisateur (Enseignant, Étudiant)
Description	Grâce à ce cas d'utilisation, un utilisateur peut s'identifier et accéder à notre plateforme.
Précondition	L'utilisateur doit avoir un email universitaire.
Postcondition	L'utilisateur doit avoir un compte sur la plateforme.

Scénarios nominaux	<p>Scénario nominal 1 : Se connecter</p> <ol style="list-style-type: none">1. L'utilisateur accède à la page de connexion.2. L'utilisateur saisit son email universitaire.3. L'utilisateur essaie de se connecter en appuyant sur le bouton '<i>Se connecter</i>'.4. Si le champ mail est vide, l'<i>Exception 1</i> se déclenche.5. Le système envoie un email One Time Password (OTP) à l'email universitaire propre de l'utilisateur avec un code OTP et un lien magique.6. L'utilisateur choisit une méthode de connexion :<ul style="list-style-type: none">– Via un lien magique : L'utilisateur clique sur le lien magique dans l'email.– Via la saisie du code OTP : L'utilisateur entre le code OTP dans le champ de la saisie du code OTP. Si le code est erroné, l'<i>Exception 2</i> se déclenche.7. Le système affiche la page principale de la plateforme. <p>Scénario nominal 2 : Se déconnecter</p> <ol style="list-style-type: none">1. L'utilisateur clique sur son avatar.2. L'utilisateur se déconnecte en cliquant sur le bouton '<i>Se déconnecter</i>'.3. Le système affiche la page de connexion.
---------------------------	---

Scénarios alternatifs	<ol style="list-style-type: none">1. L'utilisateur accède à la page de connexion.2. L'utilisateur essaie de se connecter en appuyant sur le bouton <i>'Github'</i>.3. Le système affiche la page principale de la plateforme.
Exceptions	<ul style="list-style-type: none">– Exception 1 : Si l'utilisateur ne remplit pas le champ email, un message d'erreur « <i>Le champ email est requis</i> » est affiché.– Exception 2 : Si l'utilisateur saisit un code erroné, un message d'erreur « <i>Le code est invalide</i> » est affiché.

- **Diagramme de séquence acteur-système**

Pour ce cas d'utilisation, l'utilisateur peut être soit un étudiant, soit un enseignant lors de l'identification.

La Figure 3.6 illustre le diagramme de séquence « **S'identifier** ».

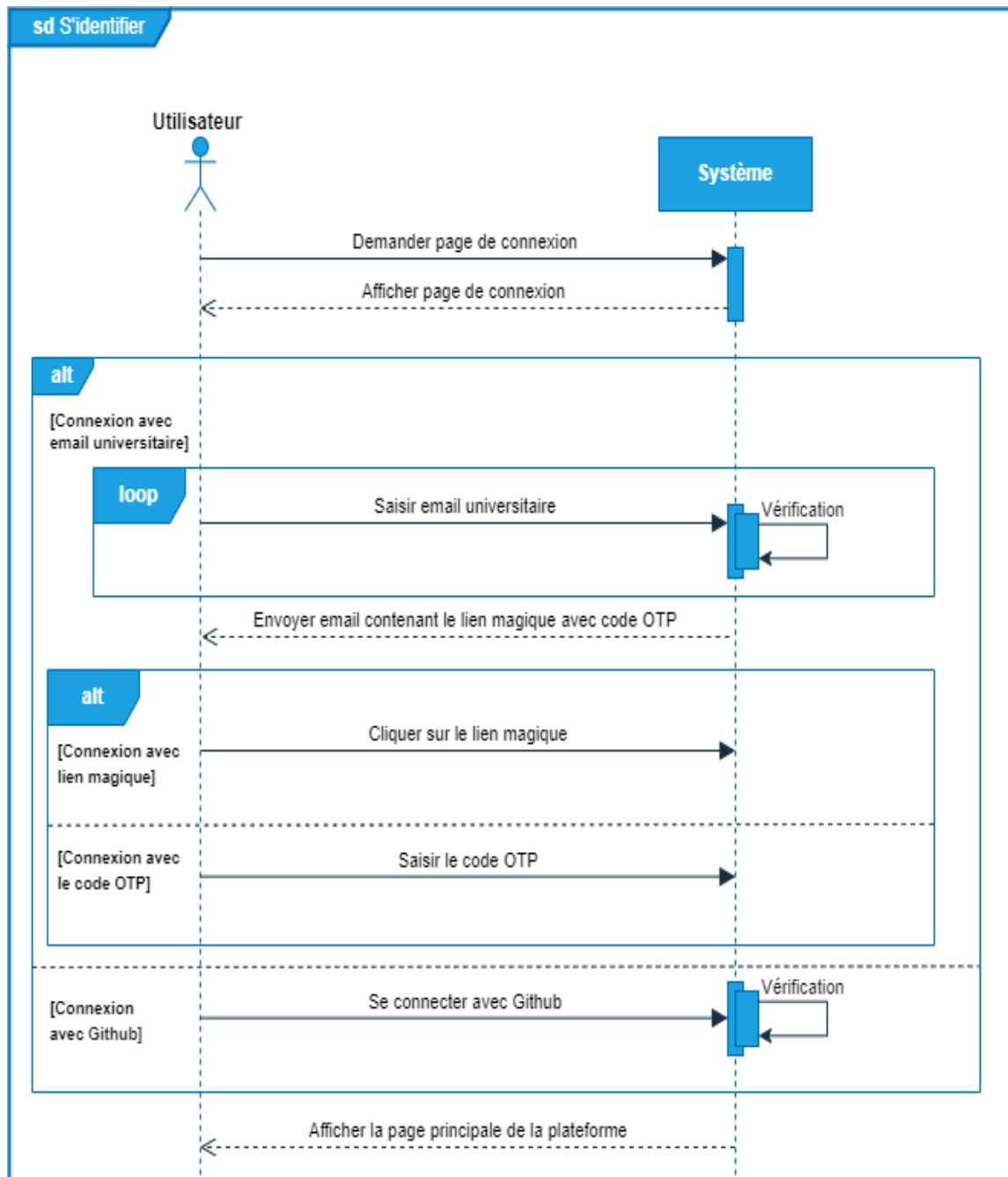


FIGURE 3.6 – Diagramme de séquence « S'identifier »


4.3.2 Cas d'utilisation « Gérer classroom »

- **Description textuelle**

Le tableau 3.2 montre la description textuelle du cas d'utilisation « **Gérer classroom** ».

TABLE 3.2 – Description textuelle du cas d'utilisation « Gérer classroom »

Titre de cas d'utilisation	Gérer classroom
Acteurs principaux	Enseignant
Description	Grâce à ce cas d'utilisation, un enseignant peut créer, modifier, supprimer, archiver et désarchiver un classroom.
Précondition	Enseignant identifié.
Postcondition	La liste des classrooms est mise à jour avec les modifications apportées (création, modification, suppression, archivage du classroom).
Scénarios nominaux	<p>Scénario nominal : Créer classroom</p> <ol style="list-style-type: none"> 1. L'enseignant demande la page de classrooms. 2. Le système affiche la liste des classrooms pour l'enseignant. 3. L'enseignant clique sur le bouton '<i>Créer classroom</i>'. 4. Le système affiche le formulaire d'ajout d'un classroom. 5. L'enseignant remplit le formulaire de la création du nouveau classroom. 6. (a) S'il existe un champ vide, l'<i>Exception 1</i> s'affiche. (b) Si un champ ne vérifie pas sa contrainte, l'<i>Exception 2</i> s'affiche. 7. Le système ajoute le classroom dans la base de données.

<p>Scénarios alternatifs</p>	<p>Scénario alternatif 1 : Modifier classroom</p> <ol style="list-style-type: none"> 1. L'enseignant demande la page de classrooms. 2. Le système affiche la liste des classrooms pour l'enseignant. 3. L'enseignant clique sur le bouton de la modification du classroom spécifié. 4. Le système affiche le formulaire de modification du classroom. 5. L'enseignant modifie les informations du classroom à modifier. 6. (a) S'il existe un champ vide, l'<i>Exception 1</i> s'affiche. (b) Si un champ ne vérifie pas sa contrainte, l'<i>Exception 2</i> s'affiche. 7. Le Système modifie les informations du classroom dans la base de données. <p>Scénario alternatif 2 : Supprimer classroom</p> <ol style="list-style-type: none"> 1. L'enseignant demande la page de classrooms. 2. Le système affiche la liste des classrooms pour l'enseignant. 3. L'enseignant clique sur le bouton de suppression de classroom spécifié. 4. Le système affiche une fenêtre superposée pour confirmer la suppression. 5. L'enseignant clique sur le bouton '<i>Confirmer</i>'. 6. Le système retire le classroom de la base de données. <p>Scénario alternatif 3 : Archiver classroom</p> <ol style="list-style-type: none"> 1. L'enseignant demande la page de classrooms. 2. Le système affiche la liste des classrooms pour l'enseignant. 3. L'enseignant clique sur le bouton d'archivage de classroom spécifié.
<p>36</p>	<p> Undrstnd</p> <ol style="list-style-type: none"> 4. Le système affiche une fenêtre superposée pour confirmer l'archivage. 5. L'enseignant clique sur le bouton '<i>Confirmer</i>'. 6. Le système archive le classroom. <p>Scénario alternatif 4 : Désarchiver classroom</p>

Exceptions	<ul style="list-style-type: none">– Exception 1 : Si l’enseignant ne remplit pas un ou plusieurs champs du formulaire, un message d’erreur spécifique est affiché pour chaque champ vide.– Exception 2 : Si l’enseignant remplit un ou plusieurs champs du formulaire mais que les contraintes ne sont pas vérifiées, un message d’erreur spécifique est affiché pour chaque champ invalide.
-------------------	---

- **Diagramme de séquence acteur-système**

La Figure 3.7 illustre le diagramme de séquence « **Créer classroom** ».

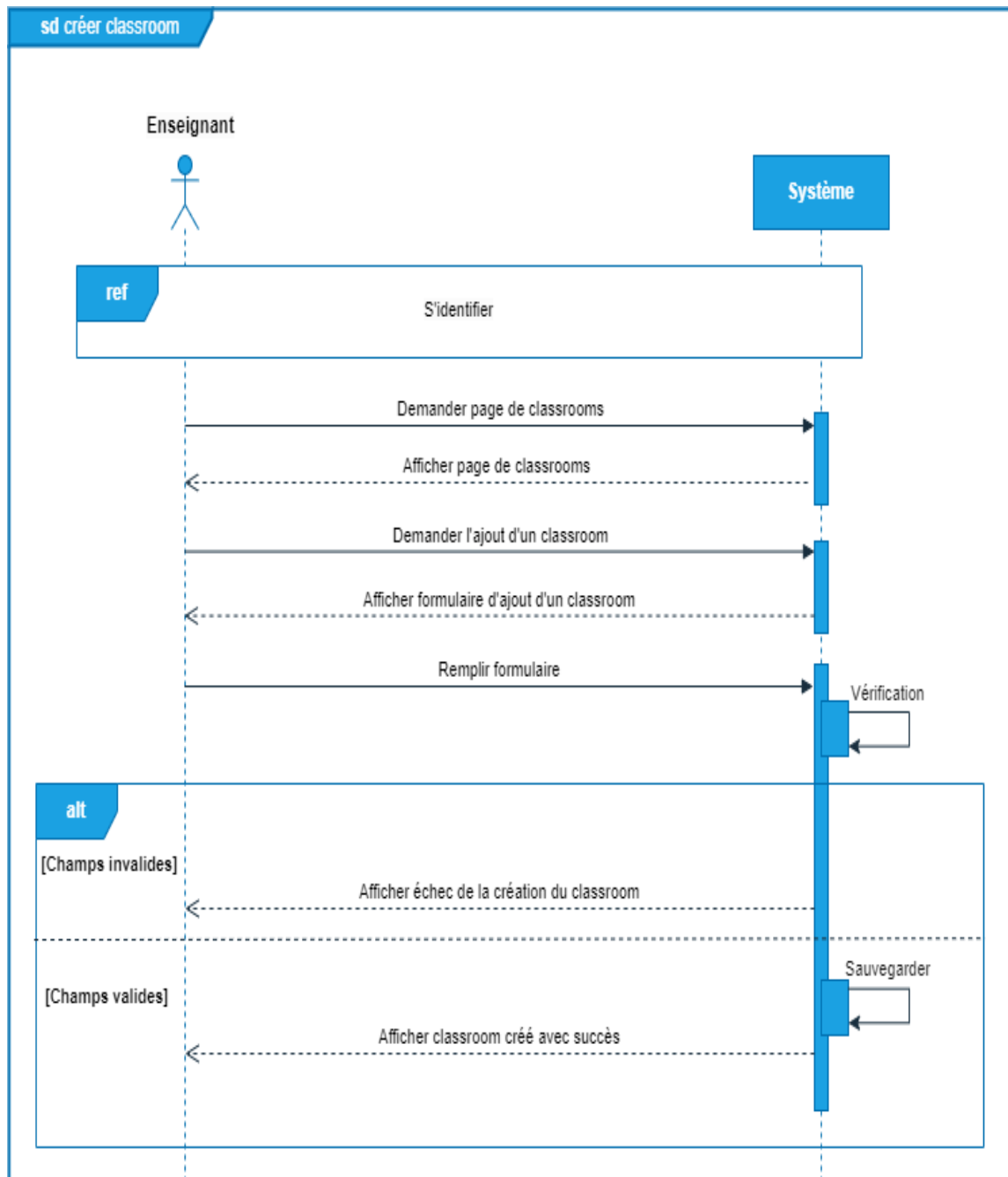


FIGURE 3.7 – Diagramme de séquence « Créer classroom »

4.3.3 Cas d'utilisation « Démarrer chat »

- **Description textuelle**

Le tableau 3.3 montre la description textuelle du cas d'utilisation « **Démarrer chat** ».

TABLE 3.3 – Description textuelle du cas d'utilisation « Démarrer chat »

Titre de cas d'utilisation	Démarrer chat
Acteurs principaux	Etudiant
Description	Grâce à ce cas d'utilisation, un étudiant peut créer une conversation avec le modèle MoE dans notre plateforme. Il peut envoyer des questions, des requêtes (demande d'explication, traduction, synthèse, etc.) sur un document partagé avec le système.
Précondition	Étudiant identifié.
Postcondition	Conversation créée.

Scénarios nominaux	<ol style="list-style-type: none">1. L'étudiant demande la page de chat avec le modèle MoE.2. Le système affiche au étudiant la page de chat.3. L'étudiant clique sur le bouton '<i>conversations</i>'.4. Le système affiche liste des conversations précédentes.5. L'étudiant choisit la manière de communication :<ul style="list-style-type: none">– Conversation en tenant les conversations précédentes :<ol style="list-style-type: none">5.1. L'étudiant sélectionne une conversation (conversation x)– L'étudiant veut démarrer une nouvelle conversation :<ol style="list-style-type: none">5.1. L'étudiant remplit le formulaire d'une nouvelle conversation.6. L'étudiant communique avec le modèle.7. Le système élucide les demandes de l'étudiant :<ul style="list-style-type: none">– Réponse en se référant aux ressources externes : Le modèle génère les réponses de l'étudiant en utilisant les informations provenant de ressources externes mentionnées.– Réponse en se référant aux ressources internes : Le modèle génère les réponses de l'étudiant en se basant sur les informations et le contexte mentionnés dans le document partagé, tout en permettant une explication plus approfondie en se référant à des ressources externes.8. Le système envoie les réponses.
---------------------------	--

Exceptions	<ul style="list-style-type: none">– Exception 1 : Si l'étudiant importe un document de format invalide, le système affiche un message d'erreur « <i>Fichier invalide, veuillez vérifier le format de fichier</i> ».– Exception 2 : Si la taille de document dépasse 25 Méga octets, le système affiche un message d'erreur « <i>Veuillez réduire la taille du document pour continuer</i> ».
-------------------	---

- **Diagramme de séquence acteur-système**

La Figure 3.8 illustre le diagramme de séquence « **Démarrer chat** ».

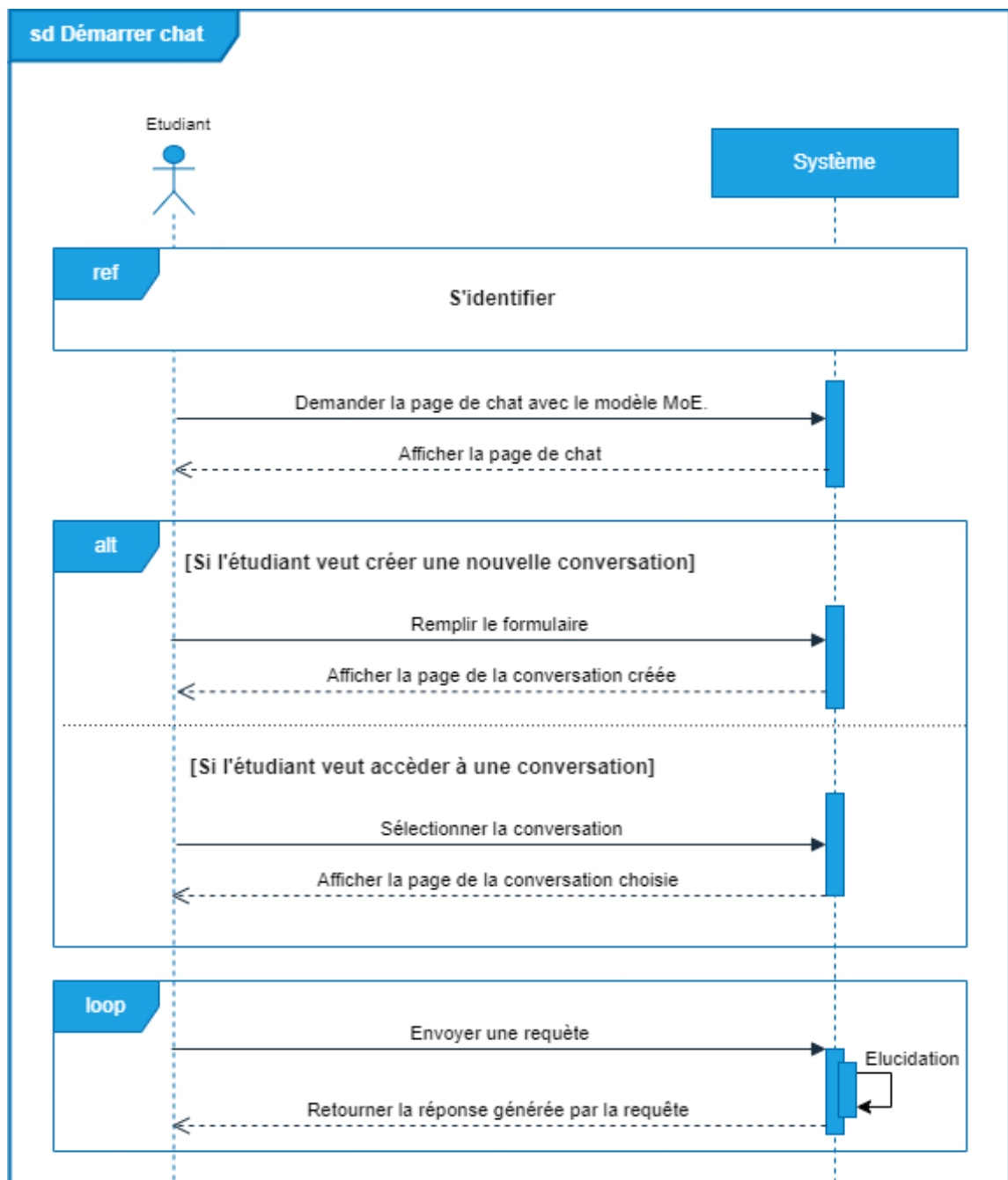


FIGURE 3.8 – Diagramme de séquence « Démarrer chat »

5 Capture des besoins techniques

Dans cette section, nous allons présenter les logiciels et outils utilisés pour développer et mettre en œuvre notre plateforme. Nous allons mettre en valeur le Framework utilisé pour le développement frontend et backend ainsi que la base de donnée choisie pour notre projet.

5.1 Framework frontend

Pour la partie frontend, nous allons utiliser le framework Next.js.

- **Nextjs**

Next.js est un framework gratuit et open source s'appuyant sur la bibliothèque javascript React. Next.js se distingue par sa capacité à offrir une expérience utilisateur performante et réactive. Grâce à sa structure basée sur React, Next.js permet de construire des interfaces utilisateur dynamiques et interactives avec une facilité. Son approche basée sur le rendu côté serveur garantit des performances optimales. Il offre aussi ses fonctionnalités avancées telles que le pré rendu des pages et le routage dynamique facilitant la navigation fluide et la gestion efficace des données.

La Figure 3.9 présente les statistiques de la satisfaction des utilisateurs à l'usage des frameworks et bibliothèques dans le domaine du développement web.

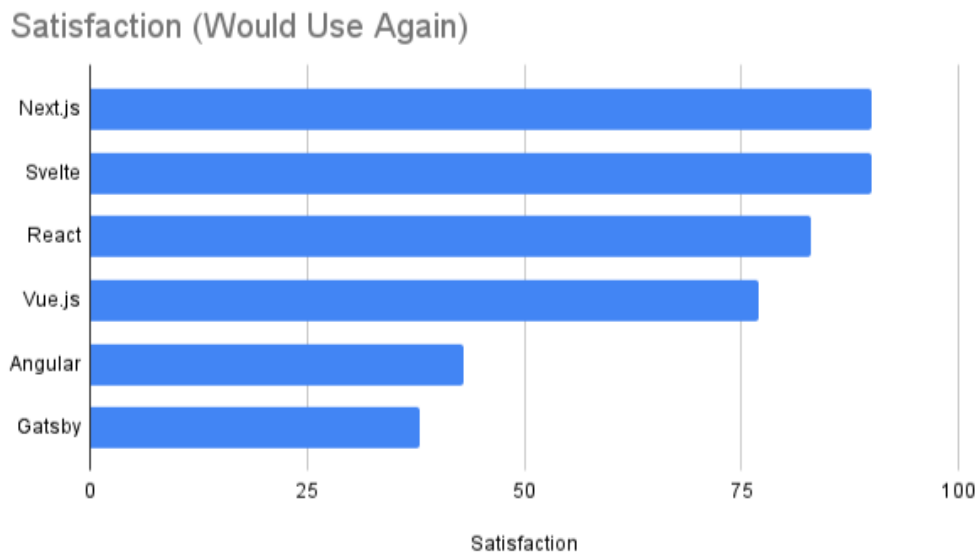


FIGURE 3.9 – Les statistiques de la satisfaction des utilisateurs à l’usage des frameworks et bibliothèques dans le domaine du développement web [6]

5.2 Framework backend

Next.js est souvent associé au développement frontend en raison de sa capacité à créer des interfaces utilisateur réactives et performantes. Il offre également des fonctionnalités robustes pour la partie backend. Grâce à sa compatibilité avec Node.js, Next.js permet de développer facilement des applications web full-stack en combinant efficacement le frontend et le backend. L’utilisation de Next.js pour la partie backend offre plusieurs avantages comme la possibilité de créer des API RESTful, la gestion des bases de données et la mise en œuvre de la logique métier côté serveur. Avec son architecture flexible et sa facilité d’utilisation, Next.js permet aux développeurs de créer des applications web complètes et cohérentes en offrant ainsi une expérience utilisateur fluide et satisfaisante pour la partie frontend et aussi la partie backend.

5.3 Base de données

Pour notre système de gestion de base de données, nous avons opté pour une approche SQL en utilisant PostgreSQL. Cette décision découle des nombreux avantages offerts par les bases de données SQL telles que la robustesse des transactions et la gestion efficace des données structurées.

- **PostgreSQL**

PostgreSQL est un système de gestion de base de données relationnelle open source qui offre une performance robuste, une fiabilité élevée et une extensibilité exceptionnelle. Connu pour sa conformité aux normes SQL, PostgreSQL prend en charge une large gamme de fonctionnalités avancées telles que les jointures complexes, les transactions ACID (Atomicité, Cohérence, Isolation, Durabilité) et la réplication asynchrone. Il est préféré pour être un choix pour les développeurs et les entreprises qui cherchent une solution de base de données puissante et fiable pour leurs applications.

La Figure 3.10 présente les statistiques des options de base de données les plus populaires.

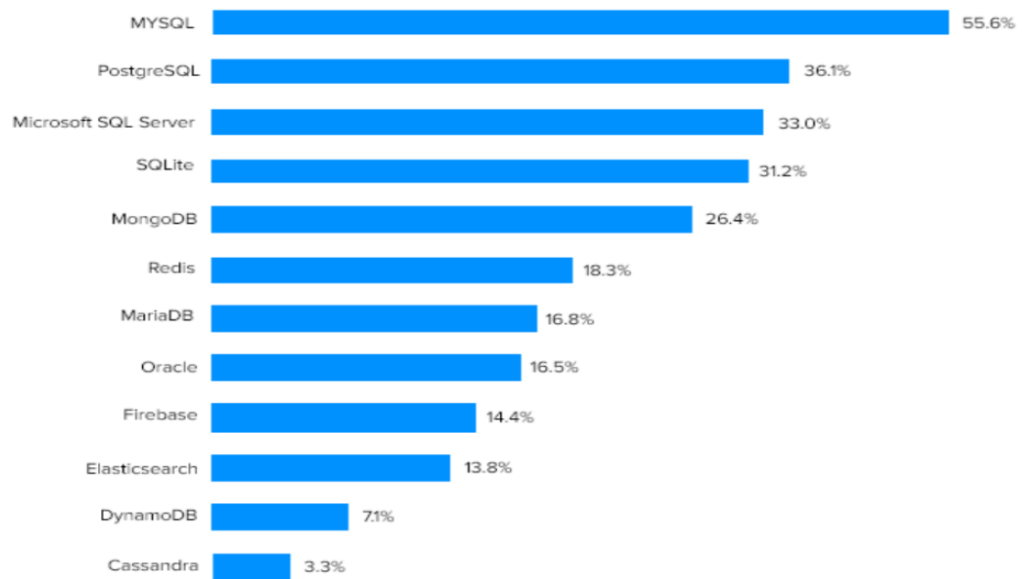


FIGURE 3.10 – Les statistiques des options de base de données les plus populaires [7]

Conclusion

Dans ce chapitre, nous avons mis en œuvre les branches fonctionnelle et technique de la méthode 2TUP se utilisant le modèle de cas d'utilisation (diagramme et description textuelle) et le diagramme de séquence d'analyse. Dans le chapitre suivant, nous présenterons la conception préliminaire et détaillée de notre application en appliquant la branche de la réalisation de la méthode 2TUP.